Signal Processing for Medical Imaging

Charles L. Byrne
Department of Mathematical Sciences
University of Massachusetts Lowell
Lowell, MA 01854

August 12, 2008

(The most recent version is available as a pdf file at
http://faculty.uml.edu/cbyrne/cbyrne.html)

Contents

Part I

Preliminaries

Chapter 1

Preface

The term *image* is used here to denote any single- or multi-dimensional representation of a distribution of interest. The term *signal processing* is also used broadly to denote the extraction of information from measured data, usually obtained through some mode of remote sensing. This is not a survey of the ever-growing field of medical imaging, nor is it a summary of the history of the subject. The emphasis here is on mathematical tools that feature prominently in medical imaging. Several areas of applications, such as transmission and emission tomography, magnetic-resonance imaging (MRI), and intensity-modulated radiation therapy, are described in some detail, both to illustrate the importance of mathematical tools such as the Fourier transform, iterative optimization and statistical parameter estimation, and to provide concrete examples of medical applications.

The reader interested in learning more about computerized tomography should consult the classical books by Kak and Slaney [140], Natterer [170], and those edited by Herman [127] and by Herman and Natterer [128]. More recent volumes, such as [171] and [210], should also be *required reading*.

Helpful introductory articles on emerging applications have appeared in recent issues of the IEEE Signal Processing Magazine, specifically the January 1997, November 2001, and May 2006 issues. The January 1997 issue, described as a *special issue on medical imaging modalities*, includes articles on electrical heart imaging [26], positron-emission tomography (PET) [172], MRI [212], and ultrasound [186]. Each of these topics was fairly well established by 1997. In contrast, the January 2001 issue, describing *emerging medical imaging technologies*, looks at such newer techniques as electromagnetic brain mapping [7], electrical impedance tomography [190], heart strain imaging [163], and diffuse optical tomography [19]. A more recent issue, in May 2006, surveys the imaging being done now at the cellular and molecular level, with articles on fluoresence microscopy [189], molecular bioimaging [165], electron microscopy [104], cryo-electron tomography

[155], and several other topics (see also [209, 220, 206, 219]).

Books on subjects such as tomographic imaging necessarily contain material on signal processing, but their treatment is often inadequate. The main reason for this, I believe, is that the concepts and problems of signal processing are best presented to students through the use of physical examples; often the best examples do not fall within the subject area of the book and the authors hesitate to include such apparently tangential material. In contrast, I have included in these notes what I consider to be the best real-world examples that illustrate the main ideas of signal processing, without regard to subject area. As a result, the reader will find discussions of solar radio-emission problems, sonar and radar imaging, ocean acoustic tomography, and the like.

These notes are designed to be used either for a one-semester course on signal processing in medical imaging, or a two-semester course that also includes an in-depth treatment of iterative reconstruction methods. Topics from the appendices should be included as needed.

Most of my referenced articles, as well as several others, are available as pdf files at http://faculty.uml.edu/cbyrne/cbyrne.html. If you find any typographical errors, please email me.

Chapter 2

Topics for Research Papers

This course is not intended as an overview of medical imaging, or even an overview of tomography; the emphasis here is on the mathematical aspects of medical image reconstruction, and there are numerous mathematical exercises throughout the text that the student is encouraged to attempt. Nevertheless, students taking this course may wish to develop a broader understanding of the various aspects of medical tomography. For that reason, I suggest here several topics for research papers.

- **1.** In this course we discuss four main types of scanning: x-ray transmission tomography (CAT); positron emission tomography (PET); single photon computed emission tomography (SPECT); and magnetic resonance imaging (MRI). Each of these modalities has its particular place in medical diagnosis, although there may be areas of overlap. Investigate the uses of these different modalities. In those areas in which more than one of these modalities are feasible, what factors are considered in making the choice?
- **2.** Select one of the four modalities listed in the previous problem and investigate the issues currently being discussed by researchers working on that modality. What are the current problems that they are trying to overcome? What are the possibilities for that modality in the future?
- **3.** Sometimes, information obtained from one type of scan can be used in another. Investigate such use of dual-modality scanning. What are the main issues involved?
- **4.** Hardware plays an important role in medical imaging. Investigate the state-of-the art in hardware for the various modalities.

- **5.** The goal in medical imaging is accurate diagnosis, not nice pictures. Investigate the ways in which this goal is included in the development of reconstruction methods.

- **6.** As new medical technologies are developed and medical costs continue to rise, there will be efforts made to weigh the benefits of the new technologies against the economic costs and potential health risks. On Sunday, June 29, 2008, the New York Times carried a front-page article, "Weighing the Costs of a Look Inside the Heart" , dealing with the benefits and costs of CT angiograms, that is, x-ray tomographic imaging of the interiors of arteries. Increased use of scanning devices obviously benefits the owners of these devices, which, increasingly, are the doctors themselves. But, are these new technologies always better than the cheaper methods they replace, and thereby worth the added cost and potential health risks?

Chapter 3

Introduction

3.1 Overview

Before we yield to the temptation to introduce mathematical notation, it is a good idea to survey the topics to be covered.

3.1.1 Topics

Our focus in this book is on several problems in what we shall loosely call medical imaging, and on the various mathematical techniques currently used to solve these problems. Specifically, we shall concentrate on problems arising in transmission tomography, emission tomography (single-photon (SPECT) and positron (PET)), magnetic resonance imaging (MRI) and intensity-modulated radiation therapy (IMRT). The mathematical techniques we shall consider include the Fourier transform, frequency-domain filtering, the fast Fourier transform (FFT), super-position models and discretization, large systems of linear equations, statistical maximum-likelihood parameter estimation, constrained optimization, iterative algorithms, randomness, sensitivity to noise, regularization methods, projection onto convex sets (POCS), and statistical detection and decision-making.

3.1.2 Organization

The approach we shall follow throughout this book is to begin with a chapter describing a particular area of medical imaging, for example, transmission tomography, with emphasis on the mathematical formulation of the problem, in this example, reconstruction from line integrals. We then investigate the relevant mathematical tools, which, in this example, include the Fourier transform, frequency-domain filtering, the fast Fourier transform and certain iterative algebraic reconstruction techniques. Once we

have completed our discussion of the mathematical tools, we may return, if needed, to the solution of the original medical imaging problem. These chapters will constitute a single part of the text. We then repeat this format, as we proceed to consider several other problem areas in successive parts of the book. Background material is included in appendices.

3.2 Transmission Tomography

Part of the text deals with transmission tomography. Previously, when people spoke of a "CAT scan" they usually meant transmission tomography, although the term is now used by lay people to describe any of several scanning modalities, including single-photon emission computed tomography (SPECT), positron emission tomography (PET), ultrasound, and magnetic resonance imaging (MRI).

3.2.1 Brief Description

Computer-assisted tomography (CAT) scans have revolutionized medical practice. One example of CAT is transmission tomography. The goal here is to image the spatial distribution of various matter within the body, by estimating the distribution of radiation attenuation. At least in theory, the data are line integrals of the function of interest.

In transmission tomography, radiation, usually x-ray, is transmitted through the object being scanned. The object of interest need not be a living human being; King Tut has received a CAT-scan and industrial uses of transmission scanning are common. Recent work [192] has shown the practicality of using cosmic rays to scan cargo for hidden nuclear material; tomographic reconstruction of the scattering ability of the contents can reveal the presence of shielding.

In the simplest formulation of transmission tomography, the beams are assumed to travel along straight lines through the object, the initial intensity of the beams is known and the intensity of the beams, as they exit the object, is measured for each line. The goal is to estimate and image the x-ray attenuation function, which correlates closely with the spatial distribution of attenuating material within the object. Unexpected absence of attenuation can indicate a broken bone, for example.

As the x-ray beam travels along its line through the body, it is weakened by the attenuating material it encounters. The reduced intensity of the exiting beam provides a measure of how much attenuation the x-ray encountered as it traveled along the line, but gives no indication of where along that line it encountered the attenuation; in theory, what we have learned is the integral of the attenuation function along the line. It is only by repeating the process with other beams along other lines that we can

begin to localize the attenuation and reconstruct an image of this non-negative attenuation function. In some approaches, the lines are all in the same plane and a reconstruction of a single slice through the object is the goal; in other cases, a fully three-dimensional scanning occurs. The word "tomography" itself comes from the Greek "tomos" , meaning part or slice; the word "atom"was coined to describe something supposed to be "without parts".

3.2.2 The Theoretical Problem

In theory, we will have the integral of the attenuation function along every line through the object. The *Radon Transform* is the operator that assigns to each attenuation function its integrals over every line. The mathematical problem is then to invert the Radon Transform, that is, to recapture the attenuation function from its line integrals. Is it always possible to determine the attenuation function from its line integrals? Yes. One way to show this is to use the Fourier transform to prove what is called the *Central Slice Theorem*. The reconstruction is then inversion of the Fourier transform; various methods for such inversion rely on frequency-domain filtering and back-projection.

3.2.3 The Practical Problem

Practise, of course, is never quite the same as theory. The problem, as we have described it, is an over-simplification in several respects, the main one being that we never have all the line integrals. Ultimately, we will construct a discrete image, made up of finitely many pixels. Consequently, it is reasonable to assume, from the start, that the attenuation function to be estimated is well approximated by a function that is constant across small squares (or cubes), called pixels (or voxels), and that the goal is to determine these finitely many pixel values.

3.2.4 The Discretized Problem

When the problem is discretized in this way, different mathematics begins to play a role. The line integrals are replaced by finite sums, and the problem can be viewed as one of solving a large number of linear equations, subject to side constraints, such as the non-negativity of the pixel values. The Fourier transform and the Central Slice Theorem are still relevant, but in discrete form, with the fast Fourier transform (FFT) playing a major role in discrete filtered back-projection methods. This approach provides fast reconstruction, but is limited in other ways. Alternatively, we can turn to iterative algorithms for solving large systems of linear equations, subject to constraints. This approach allows for greater inclusion of the

physics into the reconstruction, but can be slow; accelerating these iterative reconstruction algorithms is a major concern, as is controlling sensitivity to noise in the data.

3.2.5 Mathematical Tools

As we just saw, Fourier transformation in one and two dimensions, and frequency-domain filtering are important tools that we need to discuss in some detail. In the discretized formulation of the problem, periodic convolution of finite vectors and its implementation using the fast Fourier transform play major roles. Because actual data is always finite, we consider the issue of under-determined problems that allow for more than one answer, and the need to include prior information to obtain reasonable reconstructions. Under-determined problems are often solved using optimization, such as maximizing the entropy or minimizing the norm of the image, subject to the data as constraints. Constraints are often described mathematically using the notion of convex sets. Finding an image satisfying several sets of constraints can often be viewed as finding a vector in the intersection of convex sets, the so-called *convex feasibility problem* (CFP).

3.3 Emission Tomography

A second part of the text deals with emission tomography. Unlike transmission tomography, emission tomography (ET) is used only with living beings, principally humans and small animals. Although this modality was initially used to uncover pathologies, it is now used to study normal functioning, as well. In emission tomography, which includes positron emission tomography (PET) and single photon emission tomography (SPECT), the patient inhales, swallows, or is injected with, chemicals to which radioactive material has been chemically attached [210]. The chemicals are designed to accumulate in that specific region of the body we wish to image. For example, we may be looking for tumors in the abdomen, weakness in the heart wall, or evidence of brain activity in a selected region. In some cases, the chemicals are designed to accumulate more in healthy regions, and less so, or not at all, in unhealthy ones. The opposite may also be the case; tumors may exhibit greater avidity for certain chemicals. The patient is placed on a table surrounded by detectors that count the number of emitted photons. On the basis of where the various counts were obtained, we wish to determine the concentration of radioactivity at various locations throughout the region of interest within the patient.

Although PET and SPECT share some applications, their uses are generally determined by the nature of the chemicals that have been designed for this purpose, as well as by the half-life of the radionuclides employed.

Those radioactive isotopes used in PET generally have half-lives on the order of minutes and must be manufactured on site, adding to the expense of PET. The isotopes used in SPECT have half-lives on the order of many hours, or even days, so can be manufactured off-site and can also be used in scanning procedures that extend over some appreciable period of time.

3.3.1 Coincidence-Detection PET

In PET the radionuclide emits individual positrons, which travel, on average, between 4 mm and 2.5 cm (depending on their kinetic energy) before encountering an electron. The resulting annihilation releases two gamma-ray photons that then proceed in essentially opposite directions. Detection in the PET case means the recording of two photons at nearly the same time at two different detectors. The locations of these two detectors then provide the end points of the line segment passing, more or less, through the site of the original positron emission. Therefore, each possible pair of detectors determines a *line of response* (LOR). When a LOR is recorded, it is assumed that a positron was emitted somewhere along that line. The PET data consists of a chronological list of LOR that are recorded. Because the two photons detected at either end of the LOR are not detected at exactly the same time, the time difference can be used in *time-of-flight* PET to further localize the site of the emission to a smaller segment of perhaps 8 cm in length.

3.3.2 Single-Photon Emission Tomography

Single-photon emission tomography (SPECT) is similar to PET and has the same objective: to image the distribution of a radionuclide within the body of the patient. In SPECT the radionuclide emits single photons, which then travel through the body of the patient and, in some fraction of the cases, are detected. Detections in SPECT correspond to individual sensor locations outside the body. The data in SPECT are the photon counts at each of the finitely many detector locations. Unlike PET, in SPECT lead collimators are placed in front of the gamma-camera detectors to eliminate photons arriving at oblique angles. While this helps us narrow down the possible sources of detected photons, it also reduces the number of detected photons and thereby decreases the signal-to-noise ratio.

3.3.3 The Line-Integral Model for PET and SPECT

To solve the reconstruction problem we need a model that relates the count data to the radionuclide density function. A somewhat unsophisticated, but computationally attractive, model is taken from transmission tomography: to view the count at a particular detector as the line integral of the

radionuclide density function along the line from the detector that is perpendicular to the camera face. The count data then provide many such line integrals and the reconstruction problem becomes the familiar one of estimating a function from noisy measurements of line integrals. Viewing the data as line integrals allows us to use the Fourier transform in reconstruction. The resulting *filtered back-projection* (FBP) algorithm is a commonly used method for medical imaging in clinical settings.

The line-integral model for PET assumes a fixed set of possible LOR, with most LOR recording many emissions. Another approach is *list-mode* PET, in which detections are recording as they occur by listing the two end points of the associated LOR. The number of potential LOR is much higher in list-mode, with most of the possible LOR being recording only once, or not at all [137, 176, 53].

3.3.4 Problems with the Line-Integral Model

It is not really accurate, however, to view the photon counts at the detectors as line integrals. Consequently, applying filtered back-projection to the counts at each detector can lead to distorted reconstructions. There are at least three degradations that need to be corrected before FBP can be successfully applied [143]: attenuation, scatter, and spatially dependent resolution.

In the SPECT case, as in most such inverse problems, there is a trade-off to be made between careful modeling of the physical situation and computational tractability. The FBP method slights the physics in favor of computational simplicity and speed. In recent years, iterative methods, such as the *algebraic reconstruction technique* (ART), its multiplicative variant, MART, the expectation maximization maximum likelihood (MLEM or EMML) method, and the rescaled block-iterative EMML (RBI-EMML), that incorporate more of the physics have become competitive.

3.3.5 The Stochastic Model: Discrete Poisson Emitters

In iterative reconstruction we begin by *discretizing* the problem; that is, we imagine the region of interest within the patient to consist of finitely many tiny squares, called *pixels* for two-dimensional processing or cubes, called *voxels* for three-dimensional processing. We imagine that each pixel has its own level of concentration of radioactivity and these concentration levels are what we want to determine. Proportional to these concentration levels are the average rates of emission of photons. To achieve our goal we must construct a model that relates the measured counts to these concentration levels at the pixels. The standard way to do this is to adopt the model of *independent Poisson emitters*. Any Poisson-distributed random variable

has a mean equal to its variance. The *signal-to-noise ratio* (SNR) is usually taken to be the ratio of the mean to the standard deviation, which, in the Poisson case, is then the square root of the mean. Consequently, the Poisson SNR increases as the mean value increases, which points to the desirability (at least, statistically speaking) of higher dosages to the patient.

3.3.6 Reconstruction as Parameter Estimation

The goal is to reconstruct the distribution of radionuclide intensity by estimating the pixel concentration levels. The pixel concentration levels can be viewed as parameters and the data are instances of random variables, so the problem looks like a fairly standard parameter estimation problem of the sort studied in beginning statistics. One of the basic tools for statistical parameter estimation is likelihood maximization, which is playing an increasingly important role in medical imaging. There are several problems, however.

One problem is that the number of parameters is quite large, as large as the number of data values, in most cases. Standard statistical parameter estimation usually deals with the estimation of a handful of parameters. Another problem is that we do not quite know the relationship between the pixel concentration levels and the count data. The reason for this is that the probability that a photon emitted from a given pixel will be detected at a given detector will vary from one patient to the next, since whether or not a photon makes it from a given pixel to a given detector depends on the geometric relationship between detector and pixel, as well as what is in the patient's body between these two locations. If there are ribs or skull getting in the way, the probability of making it goes down. If there are just lungs, the probability goes up. These probabilities can change during the scanning process, when the patient moves. Some motion is unavoidable, such as breathing and the beating of the heart. Determining good values of the probabilities in the absence of motion, and correcting for the effects of motion, are important parts of SPECT image reconstruction.

3.3.7 X-Ray Fluorescence Computed Tomography

X-ray fluorescence computed tomography (XFCT) is a form of emission tomography that seeks to reconstruct the spatial distribution of elements of interest within the body [152]. Unlike SPECT and PET, these elements need not be radioactive. Beams of synchrotron radiation are used to stimulate the emission of fluorescence x-rays from the atoms of the elements of interest. These fluorescence x-rays can then be detected and the distribution of the elements estimated and imaged. As with SPECT, attenuation is a problem; making things worse is the lack of information about the distribution of attenuators at the various fluorescence energies.

3.4 Magnetic Resonance Imaging

In a third part of the text, we focus on magnetic resonance imaging. Protons have *spin*, which, for our purposes here, can be viewed as a charge distribution in the nucleus revolving around an axis. Associated with the resulting current is a *magnetic dipole moment* collinear with the axis of the spin. In elements with an odd number of protons, such as hydrogen, the nucleus itself will have a net magnetic moment. The objective in *magnetic resonance imaging* (MRI) is to determine the density of such elements in a volume of interest within the body. This is achieved by forcing the individual spinning nuclei to emit signals that, while too weak to be detected alone, are detectable in the aggregate. The signals are generated by the precession that results when the axes of the magnetic dipole moments are first aligned and then perturbed.

In much of MRI, it is the distribution of hydrogen in water molecules that is the object of interest, although the imaging of phosphorus to study energy transfer in biological processing is also important. There is ongoing work using tracers containing fluorine, to target specific areas of the body and avoid background resonance.

3.4.1 Alignment

In the absence of an external magnetic field, the axes of these magnetic dipole moments have random orientation, dictated mainly by thermal effects. When an external magnetic field is introduced, it induces a small fraction, about one in 10^5, of the dipole moments to begin to align their axes with that of the external magnetic field. Only because the number of protons per unit of volume is so large do we get a significant number of moments aligned in this way. A strong external magnetic field, about $20,000$ times that of the earth's, is required to produce enough alignment to generate a detectable signal.

3.4.2 Precession

When the axes of the aligned magnetic dipole moments are perturbed, they begin to precess, like a spinning top, around the axis of the external magnetic field, at the *Larmor frequency*, which is proportional to the intensity of the external magnetic field. If the magnetic field intensity varies spatially, then so does the Larmor frequency. Each precessing magnetic dipole moment generates a signal; taken together, they contain information about the density of the element at the various locations within the body. As we shall see, when the external magnetic field is appropriately chosen, a Fourier relationship can be established between the information extracted from the received signal and this density function.

3.4.3 Slice Isolation

When the external magnetic field is the *static field*, then the Larmor frequency is the same everywhere. If, instead, we impose an external magnetic field that varies spatially, then the Larmor frequency is also spatially varying. This external field is now said to include a *gradient field*.

3.4.4 Tipping

When a magnetic dipole moment is given a component out of its axis of alignment, it begins to precess around its axis of alignment, with frequency equal to its Larmor frequency. To create this off-axis component, we apply a *radio-frequency field* (rf field) for a short time. The effect of imposing this rf field is to tip the aligned magnetic dipole moment axes away from the axis of alignment, initiating precession. The dipoles that have been tipped ninety degrees out of their axis of alignment generate the strongest signal.

3.4.5 Imaging

The information we seek about the proton density function is contained within the received signal. By carefully adding gradient fields to the external field, we can make the Larmor frequency spatially varying, so that each frequency component of the received signal contains a piece of the information we seek. The proton density function is then obtained through Fourier transformations. Fourier-transform estimation and extrapolation techniques play a major role in this rapidly expanding field [124].

3.4.6 The Line-Integral Approach

By appropriately selecting the gradient field and the radio-frequency field, it is possible to create a situation in which the received signal comes primarily from dipoles along a given line in a preselected plane. Performing an FFT of the received signal gives us line integrals of the density function along lines in that plane. In this way, we obtain the three-dimensional Radon transform of the desired density function. The Central Slice Theorem for this case tells us that, in theory, we have the Fourier transform of the density function.

3.4.7 Phase Encoding

In the line-integral approach, the line-integral data is used to obtain values of the Fourier transform of the density function along lines through the origin in Fourier space. It would be more convenient for the FFT if we have Fourier-transform values on the points of a rectangular grid. We can obtain this by selecting the gradient fields to achieve *phase encoding*.

3.5 Intensity Modulated Radiation Therapy

Next, we consider *intensity modulated radiation therapy* (IMRT). Although it is not actually an imaging problem, intensity modulated radiation therapy is an emerging field that involves some of the same mathematical techniques used to solve the medical imaging problems discusses previously, particularly methods for solving the convex feasibility problem.

3.5.1 Brief Description

In IMRT beamlets of radiation with different intensities are transmitted into the body of the patient. Each voxel within the patient will then absorb a certain dose of radiation from each beamlet. The goal of IMRT is to direct a sufficient dosage to those regions requiring the radiation, those that are designated *planned target volumes* (PTV), while limiting the dosage received by the other regions, the so-called *organs at risk* (OAR).

3.5.2 The Problem and the Constraints

The intensities and dosages are obviously non-negative quantities. In addition, there are *implementation constraints*; the available treatment machine will impose its own requirements, such as a limit on the difference in intensities between adjacent beamlets. In dosage space, there will be a lower bound on the acceptable dosage delivered to those regions designated as the PTV, and an upper bound on the acceptable dosage delivered to those regions designated as the OAR. The problem is to determine the intensities of the various beamlets to achieve these somewhat conflicting goals.

3.5.3 Convex Feasibility and IMRT

The CQ algorithm [54, 55] is an iterative algorithm for solving the convex feasibility problem. Because it is particularly simple to implement in many cases, it has become the focus of recent work in IMRT. In [68] Censor *et al.* extend the CQ algorithm to solve what they call the *multiple-set split feasibility problem* (MSSFP) . In the sequel [69] it is shown that the constraints in IMRT can be modeled as inclusion in convex sets and the extended CQ algorithm is used to determine dose intensities for IMRT that satisfy both dose constraints and radiation-source constraints.

3.6 A Word about Prior Information

An important point to keep in mind when doing signal processing is that, while the data is usually limited, the information we seek may not be lost. Although processing the data in a reasonable way may suggest otherwise,

other processing methods may reveal that the desired information is still available in the data. Figure 3.1 illustrates this point.

The original image on the upper right of Figure 3.1 is a discrete rectangular array of intensity values simulating a slice of a head. The data was obtained by taking the two-dimensional discrete Fourier transform of the original image, and then discarding, that is, setting to zero, all these spatial frequency values, except for those in a smaller rectangular region around the origin. The problem then is under-determined. A minimum-norm solution would seem to be a reasonable reconstruction method.

The minimum-norm solution is shown on the lower right. It is calculated simply by performing an inverse discrete Fourier transform on the array of modified discrete Fourier transform values. The original image has relatively large values where the skull is located, but the minimum-norm reconstruction does not want such high values; the norm involves the sum of squares of intensities, and high values contribute disproportionately to the norm. Consequently, the minimum-norm reconstruction chooses instead to conform to the measured data by spreading what should be the skull intensities throughout the interior of the skull. The minimum-norm reconstruction does tell us something about the original; it tells us about the existence of the skull itself, which, of course, is indeed a prominent feature of the original. However, in all likelihood, we would already know about the skull; it would be the interior that we want to know about.

Using our knowledge of the presence of a skull, which we might have obtained from the minimum-norm reconstruction itself, we construct the prior estimate shown in the upper left. Now we use the same data as before, and calculate a minimum-weighted-norm reconstruction, using as the weight vector the reciprocals of the values of the prior image. This minimum-weighted-norm reconstruction is shown on the lower left; it is clearly almost the same as the original image. The calculation of the minimum-weighted norm solution can be done iteratively using the ART algorithm [194].

When we weight the skull area with the inverse of the prior image, we allow the reconstruction to place higher values there without having much of an effect on the overall weighted norm. In addition, the reciprocal weighting in the interior makes spreading intensity into that region costly, so the interior remains relatively clear, allowing us to see what is really present there.

When we try to reconstruct an image from limited data, it is easy to assume that the information we seek has been lost, particularly when a reasonable reconstruction method fails to reveal what we want to know. As this example, and many others, show, the information we seek is often still in the data, but needs to be brought out in a more subtle way.

3.7 Broader Issues

On Sunday, June 29, 2008, the New York Times carried a front-page article, "Weighing the Costs of a Look Inside the Heart" , dealing with the benefits and costs of CT angiograms, that is, x-ray tomographic imaging of the interiors of arteries. As the article points out, there is often financial incentive for doctors to use this new technology, particularly if they own the scanner, but so far there have been no large medical studies that have shown CT angiograms to be better than the older, cheaper tests. The higher cost of the CT angiograms, the exposure to radiation (the equivalent of up to several hundred chest x-rays), and the possibility of allergic reaction to and kidney damage from the contrast agents used, have led some to begin to question the increased use of this newer technology. Attempts by the administrators of Medicare not to pay for CT angiograms until better studies of their effectiveness have been carried out were successfully defeated by lobbyists for the cardiologists. The absence of clear guidelines for the use of scans continues to be troubling.

One point skeptics often make is that CT angiograms can reveal arterial blockage due to plaque, which may or may not indicate a medical problem, depending on the degree of blockage, but cannot reveal if and when some of the plaque will break away and cause a clot, which is the more serious medical problem. Often CT angiograms are performed in conjunction with emission tomographic scans designed to study blood flow, thereby increasing the overall cost and exposure to radiation.

These issues are not limited to CT angiograms and cardiology. In the Sunday, July 6, 2008 issue of Parade Magazine, the article "The Danger of Too Many Tests" explored the more general issue of risks and expense associated with the increased use of scans. The cost of diagnostic imaging approaches $100 billion dollars annually in the USA. Some five million scans are performed on children, who are ten times more sensitive to radiation than adults. Scans are commonly ordered for patients complaining of headaches, while most headaches are not indicators of a more serious condition, unless accompanied by other symptoms. Full-body scans as virtual physicals have been heavily marketed to perfectly healthy individuals, although most experts agree that this is a bad idea for people without symptoms; false positives and incidental findings often lead to more imaging and risky invasive procedures, including surgery. Even older technologies, such as chest x-rays, can be problematic when used as a general screening device, due to false positives and increased exposure to radiation. X-rays for back pain are commonplace, although most back pain goes away within a few weeks.

Magnetic-resonance imaging (MRI) is an excellent method for imaging soft tissue, and is useful for brain and cancer imaging. It does not involve radiation, but is much more expensive than ordinary x-rays, which are

often adequate.

The financial benefits that accrue to the owners of the scanners is not the whole story for the rapid increase in the use of scans. Doctors have less time to spend with patients these days, so scans provide a means for getting quick answers. Doctors and hospitals fear malpractice suits if they miss a serious condition because they failed to use all the available technology. Patients often expect the "best" treatment, which, to them, usually means the latest technology. A negative finding on a scan is an easy way to reassure the patient and reduce anxiety.

As medical costs grow rapidly and the expense is increasingly shared with the patient, there will be more calls for assessment of the relative costs and benefits of modern medical technology.

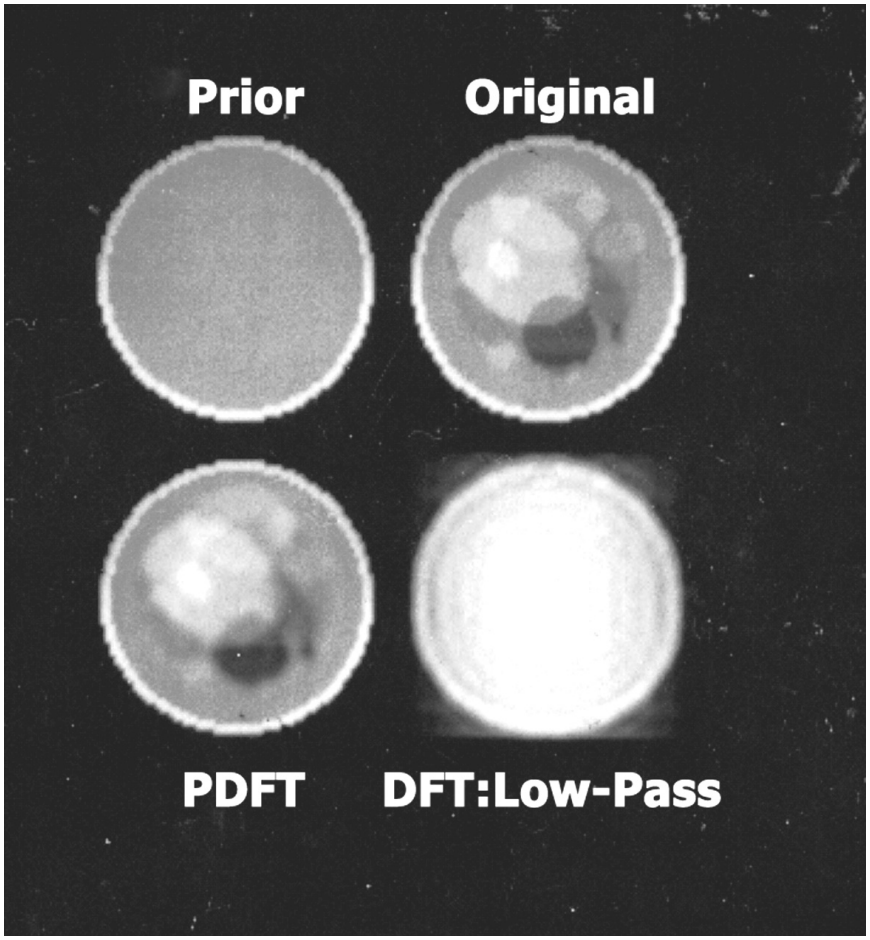

Figure 3.1: Extracting information in image reconstruction.

Part II

Transmission Tomography

Chapter 4

Transmission Tomography I

In this part of the text we focus on transmission tomography. This chapter will provide a detailed description of how the data is gathered, the mathematical model of the scanning process, and the problem to be solved. In subsequent chapters we shall study the various mathematical techniques needed to solve this problem and the manner in which these techniques are applied.

4.1 X-ray Transmission Tomography

Although transmission tomography is not limited to scanning living beings, we shall concentrate here on the use of x-ray tomography in medical diagnosis and the issues that concern us in that application. The mathematical formulation will, of course, apply more generally.

In x-ray tomography, x-rays are transmitted through the body along many lines. In some, but not all, cases, the lines will all lie in the same plane. The strength of the x-rays upon entering the body is assumed known, and the strength upon leaving the body is measured. This data can then be used to estimate the amount of attenuation the x-ray encountered along that line, which is taken to be the integral, along that line, of the attenuation function. On the basis of these line integrals, we estimate the attenuation function. This estimate is presented to the physician as one or more two-dimensional images.

4.2 The Exponential-Decay Model

As an x-ray beam passes through the body, it encounters various types of matter, such as soft tissue, bone, ligaments, air, each weakening the beam to a greater or lesser extent. If the intensity of the beam upon entry is I_{in} and I_{out} is its lower intensity after passing through the body, then

$$I_{out} = I_{in} e^{-\int_L f},$$

where $f = f(x, y) \geq 0$ is the *attenuation function* describing the two-dimensional distribution of matter within the slice of the body being scanned and $\int_L f$ is the integral of the function f over the line L along which the x-ray beam has passed. To see why this is the case, imagine the line L parameterized by the variable s and consider the intensity function $I(s)$ as a function of s. For small $\Delta s > 0$, the drop in intensity from the start to the end of the interval $[s, s + \Delta s]$ is approximately proportional to the intensity $I(s)$, to the attenuation $f(s)$ and to Δs, the length of the interval; that is,

$$I(s) - I(s + \Delta s) \approx f(s) I(s) \Delta s.$$

Dividing by Δs and letting Δs approach zero, we get

$$I'(s) = -f(s) I(s).$$

Exercise 4.1 *Show that the solution to this differential equation is*

$$I(s) = I(0) \exp(-\int_{u=0}^{u=s} f(u) du).$$

Hint: Use an integrating factor.

From knowledge of I_{in} and I_{out}, we can determine $\int_L f$. If we know $\int_L f$ for every line in the x, y-plane we can reconstruct the attenuation function f. In the real world we know line integrals only approximately and only for finitely many lines. The goal in x-ray transmission tomography is to estimate the attenuation function $f(x, y)$ in the slice, from finitely many noisy measurements of the line integrals. We usually have prior information about the values that $f(x, y)$ can take on. We also expect to find sharp boundaries separating regions where the function $f(x, y)$ varies only slightly. Therefore, we need algorithms capable of providing such images.

4.3 Difficulties to be Overcome

There are several problems associated with this model. X-ray beams are not exactly straight lines; the beams tend to spread out. The x-rays are not monochromatic, and their various frequency components are attenuated at

different rates, resulting in *beam hardening*, that is, changes in the spectrum of the beam as it passes through the object (see the appendix on the Laplace transform). The beams consist of photons obeying statistical laws, so our algorithms probably should be based on these laws. How we choose the line segments is determined by the nature of the problem; in certain cases we are somewhat limited in our choice of these segments. Patients move; they breathe, their hearts beat, and, occasionally, they shift position during the scan. Compensating for these motions is an important, and difficult, aspect of the image reconstruction process. Finally, to be practical in a clinical setting, the processing that leads to the reconstructed image must be completed in a short time, usually around fifteen minutes. This time constraint is what motivates viewing the three-dimensional attenuation function in terms of its two-dimensional slices.

As we shall see, the Fourier transform and the associated theory of convolution filters play important roles in the reconstruction of transmission tomographic images.

The data we actually obtain at the detectors are counts of detected photons. These counts are not the line integrals; they are random quantities whose means, or expected values, are related to the line integrals. The Fourier inversion methods for solving the problem ignore its statistical aspects; in contrast, other methods, such as likelihood maximization, are based on a statistical model that involves Poisson-distributed emissions.

4.4 Reconstruction from Line Integrals

We turn now to the underlying problem of reconstructing attenuation functions from line-integral data.

4.4.1 The Radon Transform

Our goal is to reconstruct the function $f(x, y) \geq 0$ from line-integral data. Let θ be a fixed angle in the interval $[0, \pi)$. Form the t, s-axis system with the positive t-axis making the angle θ with the positive x-axis, as shown in Figure 4.1. Each point (x, y) in the original coordinate system has coordinates (t, s) in the second system, where the t and s are given by

$$t = x \cos\theta + y \sin\theta,$$

and

$$s = -x \sin\theta + y \cos\theta.$$

If we have the new coordinates (t, s) of a point, the old coordinates are (x, y) given by

$$x = t \cos\theta - s \sin\theta,$$

and

$$y = t\sin\theta + s\cos\theta.$$

We can then write the function f as a function of the variables t and s. For each fixed value of t, we compute the integral

$$\int_L f(x,y)ds = \int f(t\cos\theta - s\sin\theta, t\sin\theta + s\cos\theta)ds$$

along the single line L corresponding to the fixed values of θ and t. We repeat this process for every value of t and then change the angle θ and repeat again. In this way we obtain the integrals of f over every line L in the plane. We denote by $r_f(\theta,t)$ the integral

$$r_f(\theta,t) = \int_L f(x,y)ds.$$

The function $r_f(\theta,t)$ is called the *Radon transform* of f.

4.4.2 The Central Slice Theorem

For fixed θ the function $r_f(\theta,t)$ is a function of the single real variable t; let $R_f(\theta,\omega)$ be its Fourier transform. Then

$$R_f(\theta,\omega) = \int r_f(\theta,t)e^{i\omega t}dt$$

$$= \int\int f(t\cos\theta - s\sin\theta, t\sin\theta + s\cos\theta)e^{i\omega t}dsdt$$

$$= \int\int f(x,y)e^{i\omega(x\cos\theta + y\sin\theta)}dxdy = F(\omega\cos\theta, \omega\sin\theta),$$

where $F(\omega\cos\theta, \omega\sin\theta)$ is the two-dimensional Fourier transform of the function $f(x,y)$, evaluated at the point $(\omega\cos\theta, \omega\sin\theta)$; this relationship is called the *Central Slice Theorem*. For fixed θ, as we change the value of ω, we obtain the values of the function F along the points of the line making the angle θ with the horizontal axis. As θ varies in $[0,\pi)$, we get all the values of the function F. Once we have F, we can obtain f using the formula for the two-dimensional inverse Fourier transform. We conclude that we are able to determine f from its line integrals. As we shall see, inverting the Fourier transform can be implemented by combinations of frequency-domain filtering and back-projection.

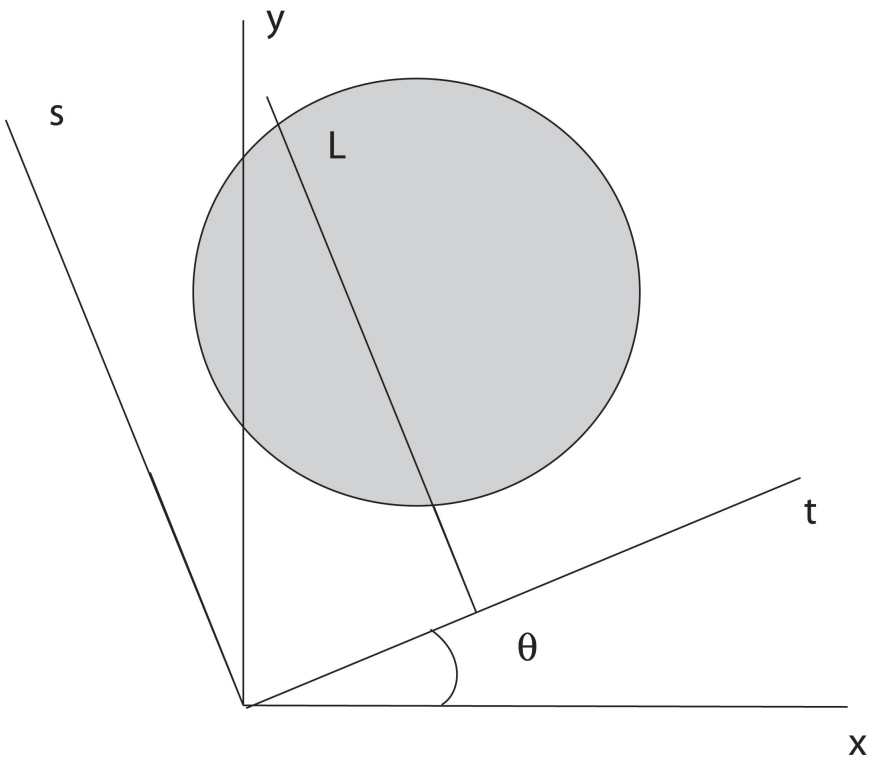

Figure 4.1: The Radon transform of f at (t, θ) is the line integral of f along line L.

Chapter 5

Complex Exponentials

The most important signals considered in signal processing are *sinusoids*, that is, sine or cosine functions. A *complex sinusoid* is a function of the real variable t having the form

$$f(t) = \cos \omega t + i \sin \omega t, \tag{5.1}$$

for some real frequency ω. Complex sinusoids are also called *complex exponential functions*.

5.1 Why "Exponential"?

Complex exponential functions have the property $f(t + u) = f(t)f(u)$, which is characteristic of exponential functions. This property can be easily verified for $f(t)$ using trigonometric identities.

Exponential functions in calculus take the form $g(t) = a^t$, for some positive constant a; the most famous of these is $g(t) = e^t$. The function $f(t)$ in Equation (5.1) has complex values, so cannot be $f(t) = a^t$ for any positive a. But, what if we let a be complex? If it is the case that $f(t) = a^t$ for some complex a, then, setting $t = 1$, we would have $a = f(1) = \cos \omega + i \sin \omega$. This is the complex number denoted e^i; to see why we consider Taylor series expansions.

5.2 Taylor-series expansions

The Taylor series expansion for the exponential function $g(t) = e^t$ is

$$e^t = 1 + t + \frac{1}{2!}t^2 + \frac{1}{3!}t^3 + ... \tag{5.2}$$

If we replace t with $i\omega$, where $i = \sqrt{-1}$, we obtain

$$e^{i\omega} = (1 - \frac{1}{2!}\omega^2 + \frac{1}{4!}\omega^4 - ...) + i(\omega - \frac{1}{3!}\omega^3 + \frac{1}{5!}\omega^5 - ...). \tag{5.3}$$

We recognize the two series in Equation (5.3) as the Taylor-series expansions for $\cos\omega$ and $\sin\omega$, respectively, so we can write

$$e^{i\omega} = \cos\omega + i\sin\omega.$$

Therefore the complex exponential function in Equation (5.1) can be written

$$f(t) = (e^{i\omega})^t = e^{i\omega t}.$$

If $A = |A|e^{i\theta}$, then the signal $h(t) = Ae^{i\omega t}$ can be written

$$h(t) = |A|e^{i(\omega t+\theta)};$$

here A is called the *complex amplitude* of the signal $h(t)$, with positive amplitude $|A|$ and phase θ.

5.3 Basic Properties

The laws of exponents apply to the complex exponential functions, so, for example, we can write

$$e^{i\omega t}e^{i\omega u} = e^{i\omega(t+u)}.$$

Note also that the complex conjugate of $e^{i\omega t}$ is

$$\overline{e^{i\omega t}} = e^{-i\omega t}$$

It follows directly from the definition of $e^{i\omega t}$ that

$$\sin(\omega t) = \frac{1}{2i}[e^{i\omega t} - e^{-i\omega t}],$$

and

$$\cos(\omega t) = \frac{1}{2}[e^{i\omega t} + e^{-i\omega t}].$$

Exercise 5.1 *Show that*

$$e^{ia} + e^{ib} = e^{i\frac{a+b}{2}}[e^{i\frac{a-b}{2}} + e^{-i\frac{a-b}{2}}] = 2e^{i\frac{a+b}{2}}\cos(\frac{a-b}{2}),$$

and

$$e^{ia} - e^{ib} = e^{i\frac{a+b}{2}}[e^{i\frac{a-b}{2}} - e^{-i\frac{a-b}{2}}] = 2ie^{i\frac{a+b}{2}}\sin(\frac{a-b}{2}).$$

Exercise 5.2 *Use the formula for the sum of a geometric progression,*

$$1 + r + r^2 + ... + r^k = (1 - r^{k+1})/(1 - r),$$

to show that

$$\sum_{n=M}^{N} e^{i\omega n} = e^{i\frac{M+N}{2}} \frac{\sin(\omega\frac{N-M+1}{2})}{\sin(\frac{\omega}{2})}. \tag{5.4}$$

Exercise 5.3 *Express the result in the previous exercise in terms of real and imaginary parts to show that*

$$\sum_{n=M}^{N} \cos(\omega n) = \cos(\frac{M+N}{2}) \frac{\sin(\omega\frac{N-M+1}{2})}{\sin(\frac{\omega}{2})},$$

and

$$\sum_{n=M}^{N} \sin(\omega n) = \sin(\frac{M+N}{2}) \frac{\sin(\omega\frac{N-M+1}{2})}{\sin(\frac{\omega}{2})}.$$

Chapter 6

The Fourier Transform

As we noted previously, the Fourier transform in one and two dimensions plays an important role in transmission tomographic image reconstruction, both in the theoretical formulation and in the practical implementation. In fact, in many areas of remote sensing, including MRI, what we want is related by the Fourier transform to what we can measure.

In this chapter we review the basic properties of the Fourier transform.

6.1 Fourier-Transform Pairs

Let $f(x)$ be defined for the real variable x in $(-\infty, \infty)$. The *Fourier transform* of $f(x)$ is the function of the real variable γ given by

$$F(\gamma) = \int_{-\infty}^{\infty} f(x) e^{i\gamma x} dx. \tag{6.1}$$

Precisely how we interpret the infinite integrals that arise in the discussion of the Fourier transform will depend on the properties of the function $f(x)$. A detailed treatment of this issue, which is beyond the scope of this book, can be found in almost any text on the Fourier transform (see, for example, [113]).

6.1.1 The Issue of Units

When we write $\cos \pi = -1$, it is with the understanding that π is a measure of angle, in radians; the function cos will always have an independent variable in units of radians. By extension, the same is true of the complex exponential functions. Therefore, when we write $e^{ix\gamma}$, we understand the product $x\gamma$ to be in units of radians. If x is measured in seconds, then γ is in units of radians per second; if x is in meters, then γ is in units of

radians per meter. When x is in seconds, we sometimes use the variable $\frac{\gamma}{2\pi}$; since 2π is then in units of radians per cycle, the variable $\frac{\gamma}{2\pi}$ is in units of cycles per second, or Hertz. When we sample $f(x)$ at values of x spaced Δ apart, the Δ is in units of x-units per sample, and the reciprocal, $\frac{1}{\Delta}$, which is called the *sampling frequency*, is in units of samples per x-units. If x is in seconds, then Δ is in units of seconds per sample, and $\frac{1}{\Delta}$ is in units of samples per second.

6.1.2 Reconstructing from Fourier-Transform Data

Our goal is often to reconstruct the function $f(x)$ from measurements of its Fourier transform $F(\gamma)$. But, how?

If we have $F(\gamma)$ for all real γ, then we can recover the function $f(x)$ using the *Fourier Inversion Formula*:

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\gamma) e^{-i\gamma x} d\gamma. \tag{6.2}$$

The functions $f(x)$ and $F(\gamma)$ are called a *Fourier-transform pair*. Once again, the proper interpretation of Equation (6.2) will depend on the properties of the functions involved. If both $f(x)$ and $F(\gamma)$ are measurable and absolutely integrable then both functions are continuous. In the next chapter, we prove the Fourier Inversion Formula for the functions in the Schwartz class [113].

6.1.3 An Example

Consider the function $f(x) = \frac{1}{2A}$, for $|x| \leq A$, and $f(x) = 0$, otherwise. The Fourier transform of this $f(x)$ is

$$F(\gamma) = \frac{\sin(A\gamma)}{A\gamma},$$

for all real $\gamma \neq 0$, and $F(0) = 1$. Note that $F(\gamma)$ is nonzero throughout the real line, except for isolated zeros, but that it goes to zero as we go to the infinities. This is typical behavior. Notice also that the smaller the A, the slower $F(\gamma)$ dies out; the first zeros of $F(\gamma)$ are at $|\gamma| = \frac{\pi}{A}$, so the main lobe widens as A goes to zero. The function $f(x)$ is not continuous, so its Fourier transform cannot be absolutely integrable. In this case, the Fourier Inversion Formula must be interpreted as involving convergence in the L^2 norm.

It may seem paradoxical that when A is larger, its Fourier transform dies off more quickly. The Fourier transform $F(\gamma)$ goes to zero faster for larger A because of destructive interference. Because of differences in their complex phases as x varies, the magnitude of the sum of the complex exponential

functions $e^{i\gamma x}$ is much smaller than we might expect, especially when A is large. For smaller A the x are more similar to one another and so the complex exponential functions are much more *in phase* with one another; consequently, the magnitude of the sum remains large. A more quantitative statement of this phenomenon is provided by the *uncertainty principle* (see [56]).

6.1.4 The Dirac Delta

Consider what happens in the limit, as $A \to 0$. Then we have an infinitely high point source at $x = 0$; we denote this by $\delta(x)$, the *Dirac delta*. The Fourier transform approaches the constant function with value 1, for all γ; the Fourier transform of $f(x) = \delta(x)$ is the constant function $F(\gamma) = 1$, for all γ. The Dirac delta $\delta(x)$ has the *sifting property*:

$$\int h(x)\delta(x)dx = h(0),$$

for each function $h(x)$ that is continuous at $x = 0$.

Because the Fourier transform of $\delta(x)$ is the function $F(\gamma) = 1$, the Fourier inversion formula tells us that

$$\delta(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-i\gamma x}d\gamma. \tag{6.3}$$

Obviously, this integral cannot be understood in the usual way. The integral in Equation (6.3) is a symbolic way of saying that

$$\int h(x)(\frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-i\gamma x}d\gamma)dx = \int h(x)\delta(x)dx = h(0), \tag{6.4}$$

for all $h(x)$ that are continuous at $x = 0$; that is, the integral in Equation (6.3) has the sifting property, so it acts like $\delta(x)$. Interchanging the order of integration in Equation (6.4), we obtain

$$\int h(x)(\frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-i\gamma x}d\gamma)dx = \frac{1}{2\pi}\int_{-\infty}^{\infty}(\int h(x)e^{-i\gamma x}dx)d\gamma$$

$$= \frac{1}{2\pi}\int_{-\infty}^{\infty} H(-\gamma)d\gamma = \frac{1}{2\pi}\int_{-\infty}^{\infty} H(\gamma)d\gamma = h(0).$$

We shall return to the Dirac delta when we consider farfield point sources.

6.2 Practical Limitations

In actual remote-sensing problems, arrays of sensors cannot be of infinite extent. In digital signal processing, moreover, there are only finitely many

sensors. We never measure the entire Fourier transform $F(\gamma)$, but, at best, just part of it; in the direct transmission problem we measure $F(\gamma)$ only for $\gamma = k$, with $|k| \leq \frac{\omega}{c}$. In fact, the data we are able to measure is almost never exact values of $F(\gamma)$, but rather, values of some distorted or blurred version. To describe such situations, we usually resort to *convolution-filter* models.

6.3 Convolution Filtering

Imagine that what we measure are not values of $F(\gamma)$, but of $F(\gamma)H(\gamma)$, where $H(\gamma)$ is a function that describes the limitations and distorting effects of the measuring process, including any blurring due to the medium through which the signals have passed, such as refraction of light as it passes through the atmosphere. If we apply the Fourier Inversion Formula to $F(\gamma)H(\gamma)$, instead of to $F(\gamma)$, we get

$$g(x) = \frac{1}{2\pi} \int F(\gamma)H(\gamma)e^{-i\gamma x} dx. \tag{6.5}$$

The function $g(x)$ that results is $g(x) = (f * h)(x)$, the *convolution* of the functions $f(x)$ and $h(x)$, with the latter given by

$$h(x) = \frac{1}{2\pi} \int H(\gamma)e^{-i\gamma x} dx.$$

Note that, if $f(x) = \delta(x)$, then $g(x) = h(x)$; that is, our reconstruction of the object from distorted data is the function $h(x)$ itself. For that reason, the function $h(x)$ is called the *point-spread function* of the imaging system.

Convolution filtering refers to the process of converting any given function, say $f(x)$, into a different function, say $g(x)$, by convolving $f(x)$ with a fixed function $h(x)$. Since this process can be achieved by multiplying $F(\gamma)$ by $H(\gamma)$ and then inverse Fourier transforming, such convolution filters are studied in terms of the properties of the function $H(\gamma)$, known in this context as the *system transfer function*, or the *optical transfer function* (OTF); when γ is a frequency, rather than a spatial frequency, $H(\gamma)$ is called the *frequency-response function* of the filter. The function $|H(\gamma)|$, the magnitude of $H(\gamma)$, is called the *modulation transfer function* (MTF). The study of convolution filters is a major part of signal processing. Such filters provide both reasonable models for the degradation signals undergo, and useful tools for reconstruction.

Let us rewrite Equation (6.5), replacing $F(\gamma)$ and $H(\gamma)$ with their definitions, as given by Equation (6.1). Then we have

$$g(x) = \frac{1}{2\pi} \int (\int f(t)e^{i\gamma t} dt)(\int h(s)e^{i\gamma s} ds)e^{-i\gamma x} d\gamma.$$

Interchanging the order of integration, we get

$$g(x) = \frac{1}{2\pi} \int \int f(t)h(s)(\int e^{i\gamma(x-(t+s))} d\gamma)dsdt.$$

Now using Equation (6.3) to replace the inner integral with $2\pi\delta(x-(t+s))$, the next integral becomes

$$2\pi \int h(s)\delta(x-(t+s))ds = 2\pi h(x-t).$$

Finally, we have

$$g(x) = \int f(t)h(x-t)dt; \tag{6.6}$$

this is the definition of the convolution of the functions f and h.

6.4 Low-Pass Filtering

A major problem in image reconstruction is the removal of blurring, which is often modeled using the notion of convolution filtering. In the one-dimensional case, we describe blurring by saying that we have available measurements not of $F(\gamma)$, but of $F(\gamma)H(\gamma)$, where $H(\gamma)$ is the frequency-response function describing the blurring. If we know the nature of the blurring, then we know $H(\gamma)$, at least to some degree of precision. We can try to remove the blurring by taking measurements of $F(\gamma)H(\gamma)$, dividing these numbers by the value of $H(\gamma)$, and then inverse Fourier transforming. The problem is that our measurements are always noisy, and typical functions $H(\gamma)$ have many zeros and small values, making division by $H(\gamma)$ dangerous, except where the values of $H(\gamma)$ are not too small. These values of γ tend to be the smaller ones, centered around zero, so that we end up with estimates of $F(\gamma)$ itself only for the smaller values of γ. The result is a *low-pass filtering* of the object $f(x)$.

To investigate such low-pass filtering, we suppose that $H(\gamma) = 1$, for $|\gamma| \leq \Gamma$, and is zero, otherwise. Then the filter is called the ideal Γ-lowpass filter. In the farfield propagation model, the variable x is spatial, and the variable γ is spatial frequency, related to how the function $f(x)$ changes spatially, as we move x. Rapid changes in $f(x)$ are associated with values of $F(\gamma)$ for large γ. For the case in which the variable x is time, the variable γ becomes frequency, and the effect of the low-pass filter on $f(x)$ is to remove its higher-frequency components.

One effect of low-pass filtering in image processing is to smooth out the more rapidly changing features of an image. This can be useful if these features are simply unwanted oscillations, but if they are important detail, the smoothing presents a problem. Restoring such wanted detail is

often viewed as removing the unwanted effects of the low-pass filtering; in other words, we try to recapture the missing high-spatial-frequency values that have been zeroed out. Such an approach to image restoration is called *frequency-domain extrapolation* . How can we hope to recover these missing spatial frequencies, when they could have been anything? To have some chance of estimating these missing values we need to have some prior information about the image being reconstructed.

6.5 Two-Dimensional Fourier Transforms

More generally, we consider a function $f(x, y)$ of two real variables. Its Fourier transformation is

$$F(\alpha, \beta) = \int \int f(x, y) e^{i(x\alpha + y\beta)} dx dy. \tag{6.7}$$

For example, suppose that $f(x, y) = 1$ for $\sqrt{x^2 + y^2} \leq R$, and zero, otherwise. Then we have

$$F(\alpha, \beta) = \int_{-\pi}^{\pi} \int_{0}^{R} e^{-i(\alpha r \cos\theta + \beta r \sin\theta)} r dr d\theta.$$

In polar coordinates, with $\alpha = \rho \cos\phi$ and $\beta = \rho \sin\phi$, we have

$$F(\rho, \phi) = \int_{0}^{R} \int_{-\pi}^{\pi} e^{ir\rho\cos(\theta - \phi)} d\theta r dr.$$

The inner integral is well known;

$$\int_{-\pi}^{\pi} e^{ir\rho\cos(\theta - \phi)} d\theta = 2\pi J_0(r\rho),$$

where J_0 denotes the 0th order Bessel function. Using the identity

$$\int_{0}^{z} t^n J_{n-1}(t) dt = z^n J_n(z),$$

we have

$$F(\rho, \phi) = \frac{2\pi R}{\rho} J_1(\rho R).$$

Notice that, since $f(x, y)$ is a radial function, that is, dependent only on the distance from $(0, 0)$ to (x, y), its Fourier transform is also radial.

The first positive zero of $J_1(t)$ is around $t = 4$, so when we measure F at various locations and find $F(\rho, \phi) = 0$ for a particular (ρ, ϕ), we can estimate $R \approx 4/\rho$. So, even when a distant spherical object, like a star, is too far away to be imaged well, we can sometimes estimate its size by finding where the intensity of the received signal is zero [145].

6.5.1 Two-Dimensional Fourier Inversion

Just as in the one-dimensional case, the Fourier transformation that produced $F(\alpha, \beta)$ can be inverted to recover the original $f(x, y)$. The Fourier Inversion Formula in this case is

$$f(x,y) = \frac{1}{4\pi^2} \int \int F(\alpha, \beta) e^{-i(\alpha x + \beta y)} d\alpha d\beta. \tag{6.8}$$

It is important to note that this procedure can be viewed as two one-dimensional Fourier inversions: first, we invert $F(\alpha, \beta)$, as a function of, say, β only, to get the function of α and y

$$g(\alpha, y) = \frac{1}{2\pi} \int F(\alpha, \beta) e^{-i\beta y} d\beta;$$

second, we invert $g(\alpha, y)$, as a function of α, to get

$$f(x,y) = \frac{1}{2\pi} \int g(\alpha, y) e^{-i\alpha x} d\alpha.$$

If we write the functions $f(x, y)$ and $F(\alpha, \beta)$ in polar coordinates, we obtain alternative ways to implement the two-dimensional Fourier inversion. We shall consider these other ways when we discuss the tomography problem of reconstructing a function $f(x, y)$ from line-integral data.

6.6 Fourier Series

Students typically encounter Fourier series before they see Fourier transforms. Suppose that $F(\gamma)$ is zero outside of the interval $[-\Gamma, \Gamma]$. For integers n and $\Delta = \frac{\pi}{\Gamma}$, the complex exponential functions $e^{i\gamma n\Delta}$ are 2Γ-periodic, and mutually orthogonal; that is, for $m \neq n$, we have

$$\int_{-\Gamma}^{\Gamma} e^{i\gamma n\Delta} e^{-i\gamma m\Delta} d\Gamma = 0.$$

The objective in Fourier series is to express the function $F(\gamma)$, for γ in $[-\Gamma, \Gamma]$, as a sum of these complex exponential functions,

$$F(\gamma) = \sum_{n=-\infty}^{\infty} a_n e^{i\gamma n\Delta}, \tag{6.9}$$

for some choice of the coefficients a_n.

Multiplying both sides of Equation (6.9) by $e^{-i\gamma m\Delta}$ and integrating from $-\Gamma$ to Γ, we find that

$$\int_{-\Gamma}^{\Gamma} F(\gamma) e^{-i\gamma m\Delta} d\gamma = 2\Gamma a_m.$$

Notice that

$$\int_{-\Gamma}^{\Gamma} F(\gamma)e^{-i\gamma m\Delta}d\gamma = 2\pi f(m\Delta)$$

also. Consequently, we have

$$a_m = \Delta f(m\Delta).$$

This gives us the important result that whenever $F(\gamma)$ is zero outside an interval $[-\Gamma, \Gamma]$, we can recover $F(\gamma)$, and thereby $f(x)$ also, from the infinite discrete set of samples $f(m\Delta)$, for $\Delta = \frac{\pi}{\Gamma}$. In signal processing this result is called *Shannon's Sampling Theorem.*

If $G(\gamma)$ is also zero for $|\gamma| > \Gamma$, then it follows from the orthogonality of the complex exponential functions $e^{i\gamma n\Delta}$ that

$$\frac{1}{2\pi}\int_{-\Gamma}^{\Gamma} F(\gamma)\overline{G(\gamma)}d\gamma = \Delta \sum_{n=-\infty}^{\infty} f(n\Delta)\overline{g(n\Delta)};$$

this is Parseval's Equation.

Note that if $F(\gamma) = 0$ for $|\gamma| > \Gamma$, then the same is true if we replace Γ with any larger value. It follows that in Shannon's Sampling Theorem we need only that $\Delta \leq \frac{\pi}{\Gamma}$.

6.7 The Discrete Fourier Transform

Suppose again that $F(\gamma)$ is zero for $|\gamma| > \Gamma$ and let $\Delta = \frac{\pi}{\Gamma}$. In real applications we never have the entire infinite set of samples $\{f(n\Delta)\}$; at best, we would have a finite subset of these, say for $n = 1$ to $n = N$. If our goal is to estimate $F(\gamma)$, we might choose the *discrete Fourier transform* (DFT) estimate

$$F_{DFT}(\gamma) = \Delta \sum_{n=1}^{N} f(n\Delta)e^{in\Delta\gamma}.$$

The DFT estimate $F_{DFT}(\gamma)$ is data consistent; its inverse Fourier-transform value at $x = n\Delta$ is $f(n\Delta)$ for $n = 1, ..., N$. The DFT is sometimes used in a slightly more general context in which the coefficients are not necessarily viewed as samples of a function $f(x)$.

Once we have decided to use the DFT estimate for the function $F(\gamma)$, we would want to evaluate this estimate at some number of values of γ, so that, for example, we could plot this function. When N is not large (say, several hundred), this poses no problem. But in many applications, especially image processing, N is in the thousands or more, and evaluating the DFT estimate at that many points without a fast algorithm is too costly and time-consuming. The *fast Fourier transform* is an algorithm for performing this calculation quickly.

6.8 The Fast Fourier Transform

A fundamental problem in signal processing is to estimate finitely many values of the function $F(\gamma)$ from finitely many values of its (inverse) Fourier transform, $f(x)$. As we shall see, the DFT arises in several ways in that estimation effort. The *fast Fourier transform* (FFT), discovered in 1965 by Cooley and Tukey, is an important and efficient algorithm for calculating the vector DFT [82]. John Tukey has been quoted as saying that his main contribution to this discovery was the firm and often voiced belief that such an algorithm must exist.

6.8.1 Evaluating a Polynomial

To illustrate the main idea underlying the FFT, consider the problem of evaluating a real polynomial $P(x)$ at a point, say $x = c$. Let the polynomial be

$$P(x) = a_0 + a_1 x + a_2 x^2 + ... + a_{2K} x^{2K},$$

where a_{2K} might be zero. Performing the evaluation efficiently by Horner's method,

$$P(c) = (((a_{2K} c + a_{2K-1})c + a_{2K-2})c + a_{2K-3})c + ...,$$

requires $2K$ multiplications, so the complexity is on the order of the degree of the polynomial being evaluated. But suppose we also want $P(-c)$. We can write

$$P(x) = (a_0 + a_2 x^2 + ... + a_{2K} x^{2K}) + x(a_1 + a_3 x^2 + ... + a_{2K-1} x^{2K-2})$$

or

$$P(x) = Q(x^2) + xR(x^2).$$

Therefore, we have $P(c) = Q(c^2) + cR(c^2)$ and $P(-c) = Q(c^2) - cR(c^2)$. If we evaluate $P(c)$ by evaluating $Q(c^2)$ and $R(c^2)$ separately, one more multiplication gives us $P(-c)$ as well. The FFT is based on repeated use of this idea, which turns out to be more powerful when we are using complex exponentials, because of their periodicity.

6.8.2 The DFT and the Vector DFT

Given the complex N-dimensional column vector $\mathbf{f} = (f_0, f_1, ..., f_{N-1})^T$, define the DFT of vector $\mathbf{f}$ to be the function $DFT_{\mathbf{f}}(\gamma)$, defined for γ in $[0, 2\pi)$, given by

$$DFT_{\mathbf{f}}(\gamma) = \sum_{n=0}^{N-1} f_n e^{in\gamma}.$$

Let $\mathbf{F}$ be the complex N-dimensional vector $\mathbf{F} = (F_0, F_1, ..., F_{N-1})^T$, where $F_k = DFT_{\mathbf{f}}(2\pi k/N), k = 0, 1, ..., N-1$. So the vector $\mathbf{F}$ consists of N values of the function $DFT_{\mathbf{f}}$, taken at N equi-spaced points $2\pi/N$ apart in $[0, 2\pi)$.

From the formula for $DFT_{\mathbf{f}}$ we have, for $k = 0, 1, ..., N-1$,

$$F_k = F(2\pi k/N) = \sum_{n=0}^{N-1} f_n e^{2\pi i n k/N}. \tag{6.10}$$

To calculate a single F_k requires N multiplications; it would seem that to calculate all N of them would require N^2 multiplications. However, using the FFT algorithm, we can calculate vector $\mathbf{F}$ in approximately $N \log_2(N)$ multiplications.

6.8.3 Exploiting Redundancy

Suppose that $N = 2M$ is even. We can rewrite Equation (6.10) as follows:

$$F_k = \sum_{m=0}^{M-1} f_{2m} e^{2\pi i (2m) k/N} + \sum_{m=0}^{M-1} f_{2m+1} e^{2\pi i (2m+1) k/N},$$

or, equivalently,

$$F_k = \sum_{m=0}^{M-1} f_{2m} e^{2\pi i m k/M} + e^{2\pi i k/N} \sum_{m=0}^{M-1} f_{2m+1} e^{2\pi i m k/M}. \tag{6.11}$$

Note that if $0 \leq k \leq M-1$ then

$$F_{k+M} = \sum_{m=0}^{M-1} f_{2m} e^{2\pi i m k/M} - e^{2\pi i k/N} \sum_{m=0}^{M-1} f_{2m+1} e^{2\pi i m k/M}, \tag{6.12}$$

so there is no additional computational cost in calculating the second half of the entries of $\mathbf{F}$, once we have calculated the first half. The FFT is the algorithm that results when we take full advantage of the savings obtainable by splitting a DFT calculating into two similar calculations of half the size.

We assume now that $N = 2^L$. Notice that if we use Equations (6.11) and (6.12) to calculate vector $\mathbf{F}$, the problem reduces to the calculation of two similar DFT evaluations, both involving half as many entries, followed by one multiplication for each of the k between 0 and $M-1$. We can split these in half as well. The FFT algorithm involves repeated splitting of the calculations of DFTs at each step into two similar DFTs, but with half the number of entries, followed by as many multiplications as there are entries in either one of these smaller DFTs. We use recursion to calculate the cost

$C(N)$ of computing $\mathbf{F}$ using this FFT method. From Equation (6.11) we see that $C(N) = 2C(N/2) + (N/2)$. Applying the same reasoning to get $C(N/2) = 2C(N/4) + (N/4)$, we obtain

$$C(N) = 2C(N/2) + (N/2) = 4C(N/4) + 2(N/2) = ...$$

$$= 2^L C(N/2^L) + L(N/2) = N + L(N/2).$$

Therefore, the cost required to calculate $\mathbf{F}$ is approximately $N \log_2 N$.

The FFT can be used to calculate the periodic convolution (or even the nonperiodic convolution) of finite length vectors.

6.8.4 Estimating the Fourier Transform

Finally, let's return to the original context of estimating the Fourier transform $F(\gamma)$ of function $f(x)$ from finitely many samples of $f(x)$. If we have N equi-spaced samples, we can use them to form the vector $\mathbf{f}$ and perform the FFT algorithm to get vector $\mathbf{F}$ consisting of N values of the DFT estimate of $F(\omega)$. It may happen that we wish to calculate more than N values of the DFT estimate, perhaps to produce a smooth looking graph. We can still use the FFT, but we must trick it into thinking we have more data that the N samples we really have. We do this by *zero-padding*. Instead of creating the N-dimensional vector $\mathbf{f}$, we make a longer vector by appending, say, J zeros to the data, to make a vector that has dimension $N + J$. The DFT estimate is still the same function of γ, since we have only included new zero coefficients as fake data; but, the FFT thinks we have $N + J$ data values, so it returns $N + J$ values of the DFT, at $N + J$ equi-spaced values of γ in $[0, 2\pi)$.

6.8.5 The Two-Dimensional Case

Suppose now that we have the data $\{f(m\Delta_x, n\Delta_y)\}$, for $m = 1, ..., M$ and $n = 1, ..., N$, where $\Delta_x > 0$ and $\Delta_y > 0$ are the sample spacings in the x and y directions, respectively. The DFT of this data is the function $F_{DFT}(\alpha, \beta)$ defined by

$$F_{DFT}(\alpha, \beta) = \Delta_x \Delta_y \sum_{m=1}^{M} \sum_{n=1}^{N} f(m\Delta_x, n\Delta_y) e^{i(\alpha m \Delta_x + \beta n \Delta_y)},$$

for $|\alpha| \leq \pi/\Delta_x$ and $|\beta| \leq \pi/\Delta_y$. The two-dimensional FFT produces MN values of $F_{DFT}(\alpha, \beta)$ on a rectangular grid of M equi-spaced values of α and N equi-spaced values of β. This calculation proceeds as follows. First, for each fixed value of n, a FFT of the M data points $\{f(m\Delta_x, n\Delta_y)\}, m = 1, ..., M$ is calculated, producing a function, say $G(\alpha_m, n\Delta_y)$, of M equi-spaced values of α and the N equi-spaced values $n\Delta_y$. Then, for each

of the M equi-spaced values of α, the FFT is applied to the N values $G(\alpha_m, n\Delta_y), n = 1, ..., N$, to produce the final result.

Chapter 7

Properties of the Fourier Transform

In this chapter we review the basic properties of the Fourier transform.

7.1 Fourier-Transform Pairs

As we saw previously, the functions $f(x)$ and $F(\gamma)$ form a Fourier-transform pair, in which the Fourier transform (FT) of $f(x)$ is given by

$$F(\gamma) = \int_{-\infty}^{\infty} f(x) e^{i\gamma x} dx, \tag{7.1}$$

and the inverse Fourier transform (IFT) of $F(\gamma)$ is

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\gamma) e^{-i\gamma x} d\gamma. \tag{7.2}$$

Note that the definitions of the FT and IFT just given may differ slightly from the ones found elsewhere; our definitions are those of Bochner and Chandrasekharan [21] and Twomey [205]. The differences are minor and involve only the placement of the quantity 2π and of the minus sign in the exponent. One sometimes sees the Fourier transform of the function f denoted $\hat{f}$; here we shall reserve the symbol $\hat{f}$ for estimates of the function f.

7.1.1 Decomposing $f(x)$

One way to view Equation (7.2) is that it shows us the function $f(x)$ as a superposition of complex exponential functions $e^{-i\gamma x}$, where γ runs over

the entire real line. The use of the minus sign here is simply for notational convenience later. Viewed in this way, we are decomposing $f(x)$ into the complex exponential functions that make it up. For each fixed value of γ, the complex number $F(\gamma) = |F(\gamma)|e^{i\theta(\gamma)}$ tells us that the amount of $e^{-i\gamma x}$ in $f(x)$ is $|F(\gamma)|$, and that $e^{i\gamma x}$ involves a phase shift by $\theta(\gamma)$. When the function $f(x)$ corresponds to something physical, we must be careful not to assume that each of the complex exponential functions also corresponds to a physical quantity.

For example, suppose that the function $f(x)$ is simply the function that is one for $|x| \leq A$ and zero otherwise. Such a function may correspond to a physical process that is off prior to time $x = -A$, then is on until $x = A$, when it is turned off again. We can represent this function as a superposition of all the complex exponential functions $e^{i\gamma x}$, for all real γ, but no single complex exponential function corresponds to anything physical. We need the destructive interference created by these infinitely many complex exponential functions in order to make $f(x)$ zero outside $[-A, A]$.

7.2 Basic Properties of the Fourier Transform

In this section we present the basic properties of the Fourier transform. Proofs of these assertions are left as exercises.

Exercise 7.1 *Let $F(\gamma)$ be the FT of the function $f(x)$. Use the definitions of the FT and IFT given above to establish the following basic properties of the Fourier transform operation:*

- **Symmetry:** The FT of the function $F(x)$ is $2\pi f(-\gamma)$. For example, the FT of the function $f(x) = \frac{\sin(\Gamma x)}{\pi x}$ is $\chi_\Gamma(\gamma)$, so the FT of $g(x) = \chi_\Gamma(x)$ is $G(\gamma) = 2\pi\frac{\sin(\Gamma\gamma)}{\pi\gamma}$.
- **Conjugation:** The FT of $\overline{f(x)}$ is $\overline{F(-\gamma)}$.
- **Scaling:** The FT of $f(ax)$ is $\frac{1}{|a|}F(\frac{\gamma}{a})$ for any nonzero constant a.
- **Shifting:** The FT of $f(x-a)$ is $e^{ia\gamma}F(\gamma)$.
- **Modulation:** The FT of $f(x)\cos(\gamma_0 x)$ is $\frac{1}{2}[F(\gamma+\gamma_0) + F(\gamma-\gamma_0)]$.
- **Differentiation:** The FT of the nth derivative, $f^{(n)}(x)$ is $(-i\gamma)^n F(\gamma)$. The IFT of $F^{(n)}(\gamma)$ is $(ix)^n f(x)$.

- **Convolution in x:** Let f, F, g, G and h, H be FT pairs, with

$$h(x) = \int f(y)g(x-y)dy,$$

so that $h(x) = (f * g)(x)$ is the convolution of $f(x)$ and $g(x)$. Then $H(\gamma) = F(\gamma)G(\gamma)$. For example, if we take $g(x) = \overline{f(-x)}$, then

$$h(x) = \int f(x+y)\overline{f(y)}dy = \int f(y)\overline{f(y-x)}dy = r_f(x)$$

is the *autocorrelation function* associated with $f(x)$ and

$$H(\gamma) = |F(\gamma)|^2 = R_f(\gamma) \geq 0$$

is the *power spectrum* of $f(x)$.

- **Convolution in γ:** Let f, F, g, G and h, H be FT pairs, with $h(x) = f(x)g(x)$. Then $H(\gamma) = \frac{1}{2\pi}(F * G)(\gamma)$.

Exercise 7.2 *Let T be a linear, time-invariant operator. Show that T is a convolution operator by showing that, for each input function f, the output function $h = Tf$ is the convolution of f with g, where $g(t)$ is the inverse FT of the function $G(\gamma)$.*

7.3 Some Fourier-Transform Pairs

In this section we present several Fourier-transform pairs.

Exercise 7.3 *Show that the Fourier transform of $f(x) = e^{-\alpha^2 x^2}$ is $F(\gamma) = \frac{\sqrt{\pi}}{\alpha} e^{-(\frac{\gamma}{2\alpha})^2}$.*

Hint: Calculate the derivative $F'(\gamma)$ by differentiating under the integral sign in the definition of F and integrating by parts. Then solve the resulting differential equation.

Let $u(x)$ be the *Heaviside function* that is $+1$ if $x \geq 0$ and 0 otherwise. Let $\chi_X(x)$ be the *characteristic function* of the interval $[-X, X]$ that is $+1$ for x in $[-X, X]$ and 0 otherwise. Let $\mathrm{sgn}(x)$ be the *sign function* that is $+1$ if $x > 0$, -1 if $x < 0$ and zero for $x = 0$.

Exercise 7.4 *Show that the FT of the function $f(x) = u(x)e^{-ax}$ is $F(\gamma) = \frac{1}{a - i\gamma}$, for every positive constant a.*

Exercise 7.5 *Show that the FT of $f(x) = \chi_X(x)$ is $F(\gamma) = 2\frac{\sin(X\gamma)}{\gamma}$.*

Exercise 7.6 *Show that the IFT of the function* $F(\gamma) = 2i/\gamma$ *is* $f(x) = \text{sgn}(x)$.

Hints: Write the formula for the inverse Fourier transform of $F(\gamma)$ as

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{+\infty}\frac{2i}{\gamma}\cos\gamma x d\gamma - \frac{i}{2\pi}\int_{-\infty}^{+\infty}\frac{2i}{\gamma}\sin\gamma x d\gamma,$$

which reduces to

$$f(x) = \frac{1}{\pi}\int_{-\infty}^{+\infty}\frac{1}{\gamma}\sin\gamma x d\gamma,$$

since the integrand of the first integral is odd. For $x > 0$ consider the Fourier transform of the function $\chi_x(t)$. For $x < 0$ perform the change of variables $u = -x$.

We saw earlier that the $F(\gamma) = \chi_\Gamma(\gamma)$ has for its inverse Fourier transform the function $f(x) = \frac{\sin\Gamma x}{\pi x}$; note that $f(0) = \frac{\Gamma}{\pi}$ and $f(x) = 0$ for the first time when $\Gamma x = \pi$ or $x = \frac{\pi}{\Gamma}$. For any Γ-band-limited function $g(x)$ we have $G(\gamma) = G(\gamma)\chi_\Gamma(\gamma)$, so that, for any x_0, we have

$$g(x_0) = \int_{-\infty}^{\infty} g(x)\frac{\sin\Gamma(x - x_0)}{\pi(x - x_0)}dx.$$

We describe this by saying that the function $f(x) = \frac{\sin\Gamma x}{\pi x}$ has the *sifting property* for all Γ-band-limited functions $g(x)$.

As Γ grows larger, $f(0)$ approaches $+\infty$, while $f(x)$ goes to zero for $x \neq 0$. The limit is therefore not a function; it is a *generalized function* called *the Dirac delta function at zero*, denoted $\delta(x)$. For this reason the function $f(x) = \frac{\sin\Gamma x}{\pi x}$ is called an *approximate delta function*. The FT of $\delta(x)$ is the function $F(\gamma) = 1$ for all γ. The Dirac delta function $\delta(x)$ enjoys the *sifting property* for all $g(x)$; that is,

$$g(x_0) = \int_{-\infty}^{\infty} g(x)\delta(x - x_0)dx.$$

It follows from the sifting and shifting properties that the FT of $\delta(x - x_0)$ is the function $e^{ix_0\gamma}$.

The formula for the inverse FT now says

$$\delta(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-ix\gamma}d\gamma. \tag{7.3}$$

If we try to make sense of this integral according to the rules of calculus we get stuck quickly. The problem is that the integral formula doesn't mean

quite what it does ordinarily and the $\delta(x)$ is not really a function, but an operator on functions; it is sometimes called a *distribution*. The Dirac deltas are mathematical fictions, not in the bad sense of being lies or fakes, but in the sense of being made up for some purpose. They provide helpful descriptions of impulsive forces, probability densities in which a discrete point has nonzero probability, or, in array processing, objects far enough away to be viewed as occupying a discrete point in space.

We shall treat the relationship expressed by Equation (7.3) as a formal statement, rather than attempt to explain the use of the integral in what is surely an unconventional manner.

If we move the discussion into the γ domain and define the Dirac delta function $\delta(\gamma)$ to be the FT of the function that has the value $\frac{1}{2\pi}$ for all x, then the FT of the complex exponential function $\frac{1}{2\pi}e^{-i\gamma_0 x}$ is $\delta(\gamma-\gamma_0)$, visualized as a "spike" at γ_0, that is, a generalized function that has the value $+\infty$ at $\gamma=\gamma_0$ and zero elsewhere. This is a useful result, in that it provides the motivation for considering the Fourier transform of a signal $s(t)$ containing hidden periodicities. If $s(t)$ is a sum of complex exponentials with frequencies $-\gamma_n$, then its Fourier transform will consist of Dirac delta functions $\delta(\gamma-\gamma_n)$. If we then estimate the Fourier transform of $s(t)$ from sampled data, we are looking for the peaks in the Fourier transform that approximate the infinitely high spikes of these delta functions.

Exercise 7.7 *Use the fact that* $\mathrm{sgn}(x)=2u(x)-1$ *and the previous exercise to show that* $f(x)=u(x)$ *has the FT* $F(\gamma)=i/\gamma+\pi\delta(\gamma)$.

Generally, the functions $f(x)$ and $F(\gamma)$ are complex-valued, so that we may speak about their real and imaginary parts. The next exercise explores the connections that hold among these real-valued functions.

Exercise 7.8 *Let* $f(x)$ *be arbitrary and* $F(\gamma)$ *its Fourier transform. Let* $F(\gamma)=R(\gamma)+iX(\gamma)$, *where* R *and* X *are real-valued functions, and similarly, let* $f(x)=f_1(x)+if_2(x)$, *where* f_1 *and* f_2 *are real-valued. Find relationships between the pairs* R,X *and* f_1,f_2.

Exercise 7.9 *Let* f, F *be a FT pair. Let* $g(x)=\int_{-\infty}^{x} f(y)dy$. *Show that the FT of* $g(x)$ *is* $G(\gamma)=\pi F(0)\delta(\gamma)+\frac{iF(\gamma)}{\gamma}$.

Hint: For $u(x)$ the Heaviside function we have

$$\int_{-\infty}^{x} f(y)dy=\int_{-\infty}^{\infty} f(y)u(x-y)dy.$$

We can use properties of the Dirac delta functions to extend the Parseval equation to Fourier transforms, where it is usually called the *Parseval-Plancherel* equation.

Exercise 7.10 *Let $f(x), F(\gamma)$ and $g(x), G(\gamma)$ be Fourier transform pairs. Use Equation (7.3) to establish the Parseval-Plancherel equation*

$$\langle f, g\rangle = \int f(x)\overline{g(x)}dx = \frac{1}{2\pi}\int F(\gamma)\overline{G(\gamma)}d\gamma,$$

from which it follows that

$$||f||^2 = \langle f, f\rangle = \int |f(x)|^2 dx = \frac{1}{2\pi}\int |F(\gamma)|^2 d\gamma.$$

Exercise 7.11 *We define the even part of $f(x)$ to be the function*

$$f_e(x) = \frac{f(x) + f(-x)}{2},$$

and the odd part of $f(x)$ to be

$$f_o(x) = \frac{f(x) - f(-x)}{2};$$

define F_e and F_o similarly for F the FT of f. Let $F(\gamma) = R(\gamma) + iX(\gamma)$ be the decomposition of F into its real and imaginary parts. We say that f is a causal function if $f(x) = 0$ for all $x < 0$. Show that, if f is causal, then R and X are related; specifically, show that X is the Hilbert transform of R, that is,

$$X(\gamma) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{R(\alpha)}{\gamma - \alpha}d\alpha.$$

Hint: If $f(x) = 0$ for $x < 0$ then $f(x)\text{sgn}(x) = f(x)$. Apply the convolution theorem, then compare real and imaginary parts.

Exercise 7.12 *The one-sided Laplace transform (LT) of f is $\mathcal{F}$ given by*

$$\mathcal{F}(z) = \int_0^{\infty} f(x)e^{-zx}dx.$$

Compute $\mathcal{F}(z)$ for $f(x) = u(x)$, the Heaviside function. Compare $\mathcal{F}(-i\gamma)$ with the FT of u.

7.4 Functions in the Schwartz Class

As we have already seen, the integrals in the formulas relating the two functions of a Fourier-transform pair sometimes need to be interpreted with care, depending on the properties of the two functions involved. One class of functions for which we can establish the formulas is the Schwartz class. A function $f(x)$ is said to be in the *Schwartz class*, or to be a *Schwartz function* if $f(x)$ is infinitely differentiable and

$$|x|^m f^{(n)}(x) \to 0$$

as x goes to $-\infty$ and $+\infty$. Here $f^{(n)}(x)$ denotes the nth derivative of $f(x)$. An example of a Schwartz function is $f(x) = e^{-x^2}$, with Fourier transform $F(\gamma) = \sqrt{\pi}e^{-\gamma^2/4}$. If $f(x)$ is a Schwartz function, then so is its Fourier transform.

To prove the Fourier Inversion Formula it is sufficient to show that

$$f(0) = \int_{-\infty}^{\infty} F(\gamma)d\gamma/2\pi.$$

Write

$$f(x) = f(0)e^{-x^2} + (f(x) - f(0)e^{-x^2}) = f(0)e^{-x^2} + g(x). \tag{7.4}$$

Then $g(0) = 0$, so $g(x) = xh(x)$. Then the Fourier transform of $g(x)$ is the derivative of the Fourier transform of $h(x)$; that is,

$$G(\gamma) = H'(\gamma).$$

The function $H(\gamma)$ is a Schwartz function, so it goes to zero at the infinities. Computing the Fourier transform of both sides of Equation (7.4), we obtain

$$F(\gamma) = f(0)\sqrt{\pi}e^{-\gamma^2/4} + H'(\gamma). \tag{7.5}$$

Therefore,

$$\int_{-\infty}^{\infty} F(\gamma)d\gamma = 2\pi f(0) + H(+\infty) - H(-\infty) = 2\pi f(0).$$

To prove the Fourier Inversion Formula, we let $K(\gamma) = F(\gamma)e^{-ix_0\gamma}$, for fixed x_0. Then the inverse Fourier transform of $K(\gamma)$ is $k(x) = f(x + x_0)$, and therefore

$$\int_{-\infty}^{\infty} K(\gamma)d\gamma = 2\pi k(0) = 2\pi f(x_0).$$

Chapter 8

Using Prior Knowledge

A basic problem in signal processing is the estimation of the function $F(\gamma)$ from finitely many values of its inverse Fourier transform $f(x)$. The DFT is one such estimator. As we shall see in this section, there are other estimators that are able to make better use of prior information about $F(\gamma)$ and thereby provide a better estimate.

8.1 Over-sampling

We assume, for the moment, that $F(\gamma) = 0$ for $|\gamma| > \Gamma$ and that $\Delta = \frac{\pi}{\Gamma}$. In Figure 8.1 below, we show the DFT estimate for $F(\gamma)$ for a case in which $\Gamma = \frac{\pi}{30}$. This would tell us that the proper sampling spacing is $\Delta = 30$. However, it is not uncommon to have situations in which x is time and we can take as many samples of $f(x)$ as we wish, but must take the samples at points x within some limited time interval, say $[0, A]$. In the case considered in the figure, $A = 130$. If we had used $\Delta = 30$, we would have obtained only four data points, which is not sufficient information. Instead, we used $\Delta = 1$ and took $N = 129$ data points; we *over-sampled*. There is a price to be paid for over-sampling, however.

The DFT estimation procedure does not "know" about the true value of Γ; it only "sees" Δ. It "assumes" incorrectly that Γ must be π, since $\Delta = 1$. Consequently, it "thinks" that we want it to estimate $F(\gamma)$ on the interval $[-\pi, \pi]$. It doesn't "know" that we know that $F(\gamma)$ is zero on most of this interval. Therefore, the DFT spends a lot of its energy trying to describe the part of the graph of $F(\gamma)$ where it is zero, and relatively little of its energy describing what is happening within the interval $[-\Gamma, \Gamma]$, which is all that we are interested in. This is why the bottom graph in the figure shows the DFT to be poor within $[-\Gamma, \Gamma]$. There is a second graph in the figure. It looks quite a bit better. How was that graph obtained?

We know that $F(\gamma) = 0$ outside the interval $[-\Gamma, \Gamma]$. Can we somehow let the estimation process know that we know this, so that it doesn't waste its energy outside this interval? Yes, we can.

The *characteristic function* of the interval $[-\Gamma, \Gamma]$ is

$$\chi_\Gamma(\gamma) = \begin{cases} 1, \text{if } |\gamma| \leq \Gamma\,; \\ 0, \text{if } |\gamma| > \Gamma\,. \end{cases}$$

We take as our estimator of $F(\gamma)$ a function called the *modified* DFT, (MDFT) having the form

$$MDFT(\gamma) = \chi_\Gamma(\gamma) \sum_{m=0}^{N-1} a_m e^{im\Delta\gamma}. \tag{8.1}$$

We determine the coefficients a_m by making $MDFT(\gamma)$ consistent with the data. Inserting $MDFT(\gamma)$ into the integral in Equation (7.2) and setting $x = n\Delta$, for each $n = 0, 1, ..., N-1$, in turn, we find that we must have

$$f(n\Delta) = \frac{1}{2\pi} \sum_{m=0}^{N-1} a_m \int_{-\Gamma}^{\Gamma} e^{i(m-n)\Delta\gamma} d\gamma.$$

Performing the integration, we find that we need

$$f(n\Delta) = \sum_{m=0}^{N-1} a_m \frac{\sin(\Gamma(n-m)\Delta)}{\pi(n-m)\Delta}, \tag{8.2}$$

for $n = 0, 1, ..., N-1$. We solve for the a_m and insert these coefficients into the formula for the MDFT. The graph of the MDFT is the top graph in the figure.

The main idea in the MDFT is to use a form of the estimator that already includes whatever important features of $F(\gamma)$ we may know a priori. In the case of the MDFT, we knew that $F(\gamma) = 0$ outside the interval $[-\Gamma, \Gamma]$, so we introduced a factor of $\chi_\Gamma(\gamma)$ in the estimator. Now, whatever coefficients we use, any estimator of the form given in Equation (8.1) will automatically be zero outside $[-\Gamma, \Gamma]$. We are then free to select the coefficients so as to make the MDFT consistent with the data. This involves solving the system of linear equations in (8.2).

8.2 Using Other Prior Information

The approach that led to the MDFT estimate suggests that we can introduce other prior information besides the support of $F(\gamma)$. For example, if we have some idea of the overall shape of the function $F(\gamma)$, we could

choose $P(\gamma) > 0$ to indicate this shape and use it instead of $\chi_\Gamma(\gamma)$ in our estimator. This leads to the PDFT estimator, which has the form

$$PDFT(\gamma) = P(\gamma) \sum_{n=0}^{N-1} b_m e^{im\Delta\gamma}. \tag{8.3}$$

Now we find the b_m by forcing the right side of Equation (8.3) to be consistent with the data. Inserting the function $PDFT(\gamma)$ into the integral in Equation (7.2), we find that we must have

$$f(n\Delta) = \frac{1}{2\pi} \sum_{m=0}^{N-1} b_m \int_{-\infty}^{\infty} P(\gamma) e^{i(m-n)\Delta\gamma} d\gamma. \tag{8.4}$$

Using $p(x)$, the inverse Fourier transform of $P(\gamma)$, given by

$$p(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} P(\gamma) e^{-ix\gamma} d\gamma,$$

we find that we must have

$$f(n\Delta) = \sum_{m=0}^{N-1} b_m p((n-m)\Delta), \tag{8.5}$$

for $n = 0, 1, ..., N-1$. We solve this system of equations for the b_m and insert them into the PDFT estimator in Equation (8.3).

In Figure 8.2 we have the function $F(\gamma)$ in the upper left corner. It consists of one large bump in the center and one smaller bump toward the right side. The DFT on the upper right side gives only slight indication that the smaller bump exists. The data here is somewhat over-sampled, so we can try the MDFT. The prior for the MDFT is $P(\gamma) = \chi_\Gamma(\gamma)$, which is pictured in the center left frame; it is shown only over $[-\Gamma, \Gamma]$, where it is just one. The MDFT estimate is in the center right frame; it shows only slight improvement over the DFT. Now, suppose we know that there is a large bump in the center. Both the DFT and the MDFT tell us clearly that this is the case, so even if we did not know it at the start, we know it now. Let's select as our prior a function $P(\gamma)$ that includes the big bump in the center, as shown in the lower left. The PDFT on the lower right now shows the smaller bump more clearly.

A more dramatic illustration of the use of the PDFT is shown in Figure 8.3. The function $F(\gamma)$ is a function of two variables simulating a slice of a head. It has been approximated by a discrete image, called here the "original" . The data was obtained by taking the two-dimensional vector DFT of the discrete image and replacing most of its values with zeros. When we formed the inverse vector DFT, we obtained the estimate in the lower

right. This is essentially the DFT estimate, and it tells us nothing about the inside of the head. From prior information, or even from the DFT estimate itself, we know that the true $F(\gamma)$ includes a skull. We therefore select as our prior the (discretized) function of two variables shown in the upper left. The PDFT estimate is the image in the lower left. The important point to remember here is that the same data was used to generate both pictures.

We saw previously how the MDFT can improve the estimate of $F(\gamma)$, by incorporating the prior information about its support. Precisely why the improvement occurs is the subject of the next section.

8.3 Analysis of the MDFT

Let our data be $f(x_m)$, $m = 1, ..., M$, where the x_m are arbitrary values of the variable x. If $F(\gamma)$ is zero outside $[-\Gamma, \Gamma]$, then minimizing the energy over $[-\Gamma, \Gamma]$ subject to data consistency produces an estimate of the form

$$F_\Gamma(\gamma) = \chi_\Gamma(\gamma) \sum_{m=1}^{M} b_m \exp(ix_m\gamma),$$

with the b_m satisfying the equations

$$f(x_n) = \sum_{m=1}^{M} b_m \frac{\sin(\Gamma(x_m - x_n))}{\pi(x_m - x_n)},$$

for $n = 1, ..., M$. The matrix S_Γ with entries $\frac{\sin(\Gamma(x_m - x_n))}{\pi(x_m - x_n)}$ we call a *sinc* matrix.

8.3.1 Eigenvector Analysis of the MDFT

Although it seems reasonable that incorporating the additional information about the support of $F(\gamma)$ should improve the estimation, it would be more convincing if we had a more mathematical argument to make. For that we turn to an analysis of the eigenvectors of the sinc matrix. Throughout this subsection we make the simplification that $x_n = n$.

Exercise 8.1 *The purpose of this exercise is to show that, for an Hermitian nonnegative-definite M by M matrix Q, a norm-one eigenvector $\mathbf{u}^1$ of Q associated with its largest eigenvalue, λ_1, maximizes the quadratic form $\mathbf{a}^\dagger Q\mathbf{a}$ over all vectors $\mathbf{a}$ with norm one. Let $Q = ULU^\dagger$ be the eigenvector decomposition of Q, where the columns of U are mutually orthogonal eigenvectors $\mathbf{u}^n$ with norms equal to one, so that $U^\dagger U = I$, and*

$L = diag\{\lambda_1, ..., \lambda_M\}$ is the diagonal matrix with the eigenvalues of Q as its entries along the main diagonal. Assume that $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_M$. Then maximize

$$\mathbf{a}^\dagger Q\mathbf{a} = \sum_{n=1}^{M} \lambda_n \, |\mathbf{a}^\dagger \mathbf{u}^n|^2,$$

subject to the constraint

$$\mathbf{a}^\dagger \mathbf{a} = \mathbf{a}^\dagger U^\dagger U\mathbf{a} = \sum_{n=1}^{M} |\mathbf{a}^\dagger \mathbf{u}^n|^2 = 1.$$

Hint: Show $\mathbf{a}^\dagger Q\mathbf{a}$ is a convex combination of the eigenvalues of Q.

Exercise 8.2 *Show that, for the sinc matrix $Q = S_\Gamma$, the quadratic form $\mathbf{a}^\dagger Q\mathbf{a}$ in the previous exercise becomes*

$$\mathbf{a}^\dagger S_\Gamma \mathbf{a} = \frac{1}{2\pi} \int_{-\Gamma}^{\Gamma} |\sum_{n=1}^{M} a_n e^{in\gamma}|^2 d\gamma.$$

Show that the norm of the vector $\mathbf{a}$ is the integral

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} |\sum_{n=1}^{M} a_n e^{in\gamma}|^2 d\gamma.$$

Exercise 8.3 *For $M = 30$ compute the eigenvalues of the matrix S_Γ for various choices of Γ, such as $\Gamma = \frac{\pi}{k}$, for $k = 2, 3, ..., 10$. For each k arrange the set of eigenvalues in decreasing order and note the proportion of them that are not near zero. The set of eigenvalues of a matrix is sometimes called its eigenspectrum and the nonnegative function $\chi_\Gamma(\gamma)$ is a power spectrum; here is one time in which different notions of a spectrum are related.*

8.3.2 The Eigenfunctions of S_Γ

Suppose that the vector $\mathbf{u}^1 = (u_1^1, ..., u_M^1)^T$ is an eigenvector of S_Γ corresponding to the largest eigenvalue, λ_1. Associate with $\mathbf{u}^1$ the *eigenfunction*

$$U^1(\gamma) = \sum_{n=1}^{M} u_n^1 e^{in\gamma}.$$

Then

$$\lambda_1 = \int_{-\Gamma}^{\Gamma} |U^1(\gamma)|^2 d\gamma / \int_{-\pi}^{\pi} |U^1(\gamma)|^2 d\gamma$$

and $U^1(\gamma)$ is the function of its form that is most concentrated within the interval $[-\Gamma, \Gamma]$.

Similarly, if $\mathbf{u}^M$ is an eigenvector of S_Γ associated with the smallest eigenvalue λ_M, then the corresponding eigenfunction $U^M(\gamma)$ is the function of its form least concentrated in the interval $[-\Gamma, \Gamma]$.

Exercise 8.4 *Plot for $|\gamma| \leq \pi$ the functions $|U^m(\gamma)|$ corresponding to each of the eigenvectors of the sinc matrix S_Γ. Pay particular attention to the places where each of these functions is zero.*

The eigenvectors of S_Γ corresponding to different eigenvalues are orthogonal, that is $(\mathbf{u}^m)^\dagger \mathbf{u}^n = 0$ if m is not n. We can write this in terms of integrals:

$$\int_{-\pi}^{\pi} U^n(\gamma) \overline{U^m(\gamma)} d\gamma = 0$$

if m is not n. The mutual orthogonality of these eigenfunctions is related to the locations of their roots, which were studied in the previous exercise.

Any Hermitian matrix Q is invertible if and only if none of its eigenvalues is zero. With λ_m and $\mathbf{u}^m$, $m = 1, ..., M$, the eigenvalues and eigenvectors of Q, the inverse of Q can then be written as

$$Q^{-1} = (1/\lambda_1)\mathbf{u}^1(\mathbf{u}^1)^\dagger + ... + (1/\lambda_M)\mathbf{u}^M(\mathbf{u}^M)^\dagger.$$

Exercise 8.5 *Show that the MDFT estimator given by Equation (8.1) $F_\Gamma(\gamma)$ can be written as*

$$F_\Gamma(\gamma) = \chi_\Gamma(\gamma) \sum_{m=1}^{M} \frac{1}{\lambda_m} (\mathbf{u}^m)^\dagger \mathbf{d}\, U^m(\gamma),$$

where $\mathbf{d} = (f(1), f(2), ..., f(M))^T$ is the data vector.

Exercise 8.6 *Show that the DFT estimate of $F(\gamma)$, restricted to the interval $[-\Gamma, \Gamma]$, is*

$$F_{DFT}(\gamma) = \chi_\Gamma(\gamma) \sum_{m=1}^{M} (\mathbf{u}^m)^\dagger \mathbf{d}\, U^m(\gamma).$$

From these two exercises we can learn why it is that the estimate $F_\Gamma(\gamma)$ resolves better than the DFT. The former makes more use of the eigenfunctions $U^m(\gamma)$ for higher values of m, since these are the ones for which λ_m is closer to zero. Since those eigenfunctions are the ones having most of their roots within the interval $[-\Gamma, \Gamma]$, they have the most flexibility within that region and are better able to describe those features in $F(\gamma)$ that are not resolved by the DFT.

8.4 The Discrete PDFT (DPDFT)

The derivation of the PDFT assumes a function $f(x)$ of one or more continuous real variables, with the data obtained from $f(x)$ by integration. The discrete PDFT (DPDFT) begins with $f(x)$ replaced by a finite vector $f = (f_1, ..., f_J)^T$ that is a discretization of $f(x)$; say that $f_j = f(x_j)$ for some point x_j. The integrals that describe the Fourier transform data can be replaced by finite sums,

$$F(\gamma_n) = \sum_{j=1}^{J} f_j E_{nj}, \tag{8.6}$$

where $E_{nj} = e^{ix_j\gamma_n}$. We have used a Riemann-sum approximation of the integrals here, but other choices are also available. The problem then is to solve this system of equations for the f_j.

Since the N is fixed, but the J is under our control, we select $J > N$, so that the system becomes under-determined. Now we can use minimum-norm and minimum-weighted-norms solutions of the finite-dimensional problem to obtain an approximate, discretized PDFT solution.

Since the PDFT is a minimum-weighted norm solution in the continous-variable formulation, it is reasonable to let the DPDFT be the corresponding minimum-weighted-norm solution obtained with the positive-definite matrix Q the diagonal matrix having for its jth diagonal entry

$$Q_{jj} = 1/p(x_j), \tag{8.7}$$

if $p(x_j) > 0$, and zero, otherwise.

8.4.1 Calculating the DPDFT

The DPDFT is a minimum-weighted-norm solution, which can be calculated using, say, the ART algorithm. We know that, in the under-determined case, the ART provides the the solution closest to the starting vector, in the sense of the Euclidean distance. We therefore reformulate the system, so that the minimum-weighted norm solution becomes a minimum-norm solution, as we did earlier, and then begin the ART iteration with zero. For recent work involving the DPDFT se [195, 194, 196].

8.4.2 Regularization

We noted earlier that one of the principles guiding the estimation of $f(x)$ from Fourier transform data should be that we do not want to overfit the estimate to noisy data. In the PDFT, this can be avoided by adding a small positive quantity to the main diagonal of the matrix P. In the DPDFT, sensitivity to noise is reduced by using the iterative regularized ART [58].

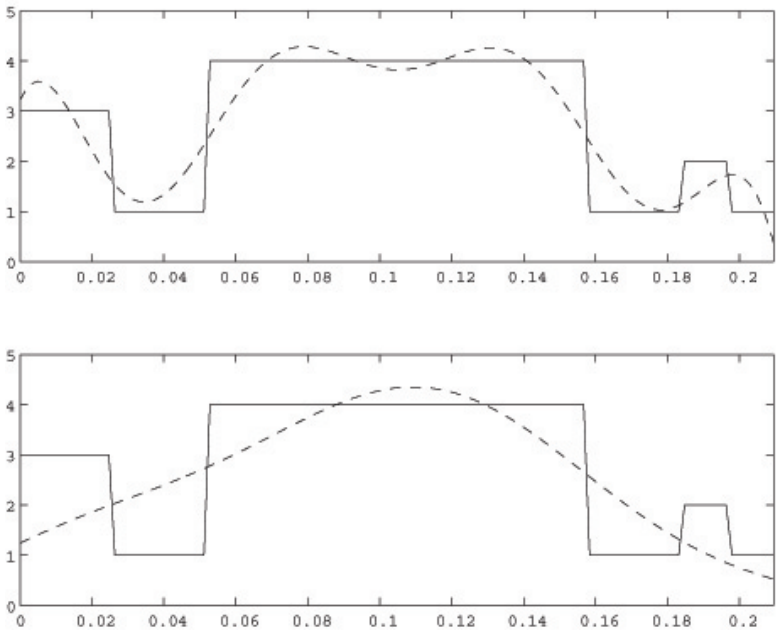

Figure 8.1: The non-iterative band-limited extrapolation method (MDFT) (top) and the DFT (bottom) for $N = 129$, $\Delta = 1$ and $\Gamma = \pi/30$.

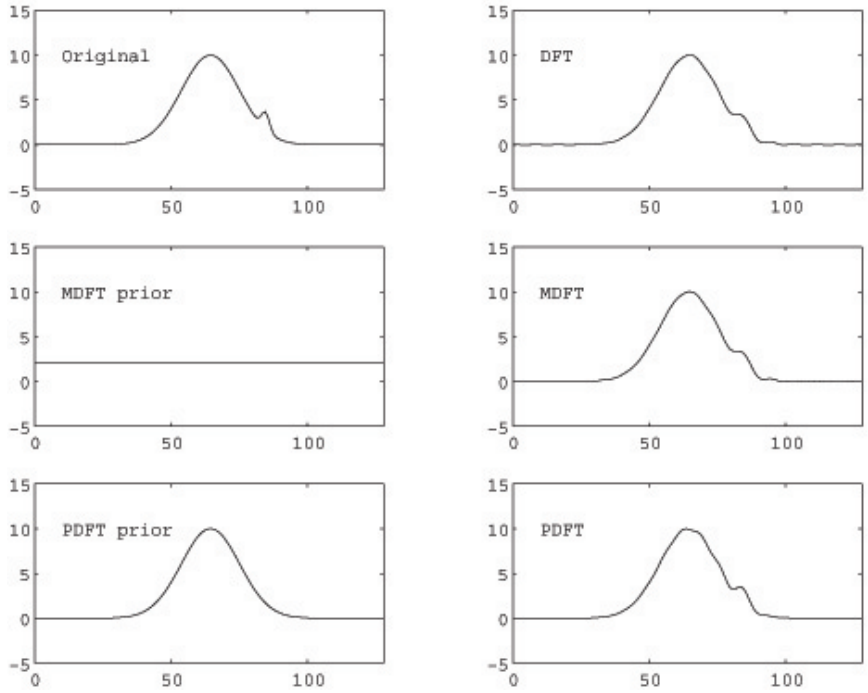

Figure 8.2: The DFT, the MDFT, and the PDFT.

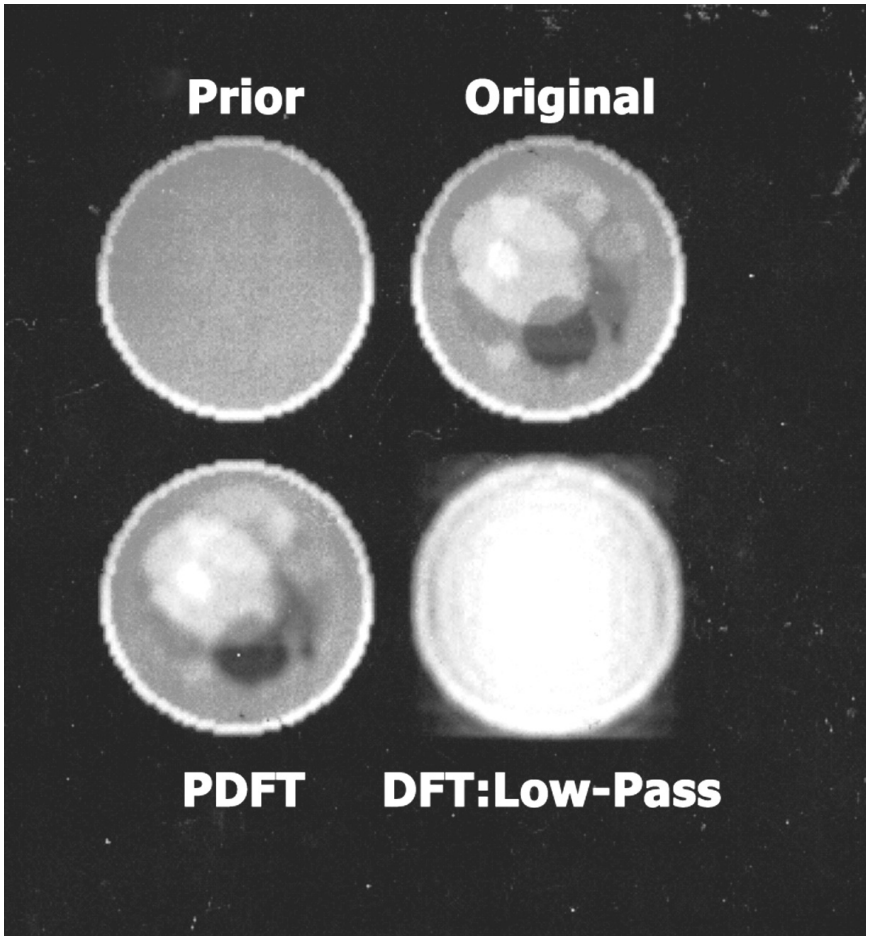

Figure 8.3: The PDFT in image reconstruction.

Chapter 9

ART and MART

The *algebraic reconstruction technique* (ART) was introduced by Gordon, Bender and Herman [120] as a method for discrete image reconstruction in transmission tomography. It was noticed somewhat later that the ART is a special case of Kaczmarz's algorithm [139].

9.1 The ART in Tomography

For $i = 1, ..., I$, let L_i be the set of pixel indices j for which the j-th pixel intersects the i-th line segment, and let $|L_i|$ be the cardinality of the set L_i. Let $A_{ij} = 1$ for j in L_i, and $A_{ij} = 0$ otherwise. With $i = k(\mathrm{mod}\, I) + 1$, the iterative step of the ART algorithm is

$$x_j^{k+1} = x_j^k + \frac{1}{|L_i|}(b_i - (Ax^k)_i), \tag{9.1}$$

for j in L_i, and

$$x_j^{k+1} = x_j^k, \tag{9.2}$$

if j is not in L_i. In each step of ART, we take the error, $b_i - (Ax^k)_i$, associated with the current x^k and the i-th equation, and distribute it equally over each of the pixels that intersects L_i.

A somewhat more sophisticated version of ART allows A_{ij} to include the length of the i-th line segment that lies within the j-th pixel; A_{ij} is taken to be the ratio of this length to the length of the diagonal of the j-pixel.

More generally, ART can be viewed as an iterative method for solving an arbitrary system of linear equations, $Ax = b$.

9.2 The ART in the General Case

Let A be a complex matrix with I rows and J columns, and let b be a member of C^I. We want to solve the system $Ax = b$.

For each index value i, let H_i be the hyperplane of J-dimensional vectors given by

$$H_i = \{x|(Ax)_i = b_i\}, \tag{9.3}$$

and P_i the orthogonal projection operator onto H_i. Let x^0 be arbitrary and, for each nonnegative integer k, let $i(k) = k(\mathrm{mod}\, I) + 1$. The iterative step of the ART is

$$x^{k+1} = P_{i(k)}x^k. \tag{9.4}$$

Because the ART uses only a single equation at each step, it has been called a *row-action* method .

9.2.1 Calculating the ART

Given any vector z the vector in H_i closest to z, in the sense of the Euclidean distance, has the entries

$$x_j = z_j + \overline{A_{ij}}(b_i - (Az)_i)/\sum_{m=1}^{J} |A_{im}|^2. \tag{9.5}$$

To simplify our calculations, we shall assume, throughout this chapter, that the rows of A have been rescaled to have Euclidean length one; that is

$$\sum_{j=1}^{J} |A_{ij}|^2 = 1, \tag{9.6}$$

for each $i = 1, ..., I$, and that the entries of b have been rescaled accordingly, to preserve the equations $Ax = b$. The ART is then the following: begin with an arbitrary vector x^0; for each nonnegative integer k, having found x^k, the next iterate x^{k+1} has entries

$$x_j^{k+1} = x_j^k + \overline{A_{ij}}(b_i - (Ax^k)_i). \tag{9.7}$$

When the system $Ax = b$ has exact solutions the ART converges to the solution closest to x^0, in the 2-norm. How fast the algorithm converges will depend on the ordering of the equations and on whether or not we use relaxation. In selecting the equation ordering, the important thing is to avoid particularly bad orderings, in which the hyperplanes H_i and H_{i+1} are nearly parallel.

9.2.2 Full-cycle ART

We also consider the *full-cycle* ART, with iterative step $z^{k+1} = Tz^k$, for

$$T = P_I P_{I-1} \cdots P_2 P_1. \tag{9.8}$$

When the system $Ax = b$ has solutions, the fixed points of T are solutions. When there are no solutions of $Ax = b$, the operator T will still have fixed points, but they will no longer be exact solutions.

9.2.3 Relaxed ART

The ART employs orthogonal projections onto the individual hyperplanes. If we permit the next iterate to fall short of the hyperplane, or somewhat beyond it, we get a relaxed version of ART.The relaxed ART algorithm is as follows:

Algorithm 9.1 (Relaxed ART) *With $\omega \in (0, 2)$, x^0 arbitrary, and $i = k(\mathrm{mod}\, I) + 1$, let*

$$x_j^{k+1} = x_j^k + \omega \overline{A_{ij}}(b_i - (Ax^k)_i)). \tag{9.9}$$

The relaxed ART converges to the solution closest to x^0, in the consistent case. In the inconsistent case, it does not converge, but subsequences associated with the same i converge to distinct vectors, forming a limit cycle.

9.2.4 Constrained ART

Let C be a closed, nonempty convex subset of C^J and $P_C x$ the orthogonal projection of x onto C. If there are solutions of $Ax = b$ that lie within C, we can find them using the constrained ART algorithm:

Algorithm 9.2 (Constrained ART) *With x^0 arbitrary and $i = k(\mathrm{mod}\, I)+1$, let*

$$x_j^{k+1} = P_C(x_j^k + \overline{A_{ij}}(b_i - (Ax^k)_i)). \tag{9.10}$$

For example, if A and b are real and we seek a nonnegative solution to $Ax = b$, we can use

Algorithm 9.3 (Non-negative ART) *With x^0 arbitrary and $i = k(\mathrm{mod}\, I)+1$, let*

$$x_j^{k+1} = (x_j^k + A_{ij}(b_i - (Ax^k)_i))_+, \tag{9.11}$$

where, for any real number a, $a_+ = \max\{a, 0\}$.

The constrained ART converges to a solution of $Ax = b$ within C, whenever such solutions exist.

Noise in the data can manifest itself in a variety of ways; we have seen what can happen when we impose positivity on the calculated least-squares solution, that is, when we minimize $||Ax-b||_2$ over all non-negative vectors x. Theorem 9.1 tells us that when $J > I$, but $Ax = b$ has no non-negative solutions, the non-negatively constrained least-squares solution can have at most $I-1$ non-zero entries, regardless of how large J is. This phenomenon also occurs with several other approximate methods, such as those that minimize the cross-entropy distance.

Definition 9.1 *The matrix A has the full-rank property if A and every matrix Q obtained from A by deleting columns have full rank.*

Theorem 9.1 *Let A have the full-rank property. Suppose there is no non-negative solution to the system of equations $Ax = b$. Then there is a subset S of the set $\{j = 1, 2, ..., J\}$, with cardinality at most $I-1$, such that, if $\hat{x}$ is any minimizer of $||Ax - b||_2$ subject to $x \geq 0$, then $\hat{x}_j = 0$ for j not in S. Therefore, $\hat{x}$ is unique.*

For a proof, see [58].

9.2.5 Convergence of ART

For the consistent case, in which the system $Ax = b$ has exact solutions, we have the following result.

Theorem 9.2 *Let $A\hat{x} = b$ and let x^0 be arbitrary. Let $\{x^k\}$ be generated by Equation (9.7). Then the sequence $\{||\hat{x} - x^k||_2\}$ is decreasing and $\{x^k\}$ converges to the solution of $Ax = b$ closest to x^0.*

9.3 The MART

The *multiplicative* ART (MART) [120] is an iterative algorithm closely related to the ART. It also was devised to obtain tomographic images, but, like ART, applies more generally; MART applies to systems of linear equations $Ax = b$ for which the b_i are positive, the A_{ij} are nonnegative, and the solution x we seek is to have nonnegative entries. It is not so easy to see the relation between ART and MART if we look at the most general formulation of MART. For that reason, we begin with a simpler case, transmission tomographic imaging, in which the relation is most clearly visible.

9.3.1 A Special Case of MART

We begin by considering the application of MART to the transmission tomography problem. For $i = 1, ..., I$, let L_i be the set of pixel indices j for which the j-th pixel intersects the i-th line segment, and let $|L_i|$ be the cardinality of the set L_i. Let $A_{ij} = 1$ for j in L_i, and $A_{ij} = 0$ otherwise. With $i = k(\mathrm{mod}\, I) + 1$, the iterative step of the ART algorithm is

$$x_j^{k+1} = x_j^k + \frac{1}{|L_i|}(b_i - (Ax^k)_i), \tag{9.12}$$

for j in L_i, and

$$x_j^{k+1} = x_j^k, \tag{9.13}$$

if j is not in L_i. In each step of ART, we take the error, $b_i - (Ax^k)_i$, associated with the current x^k and the i-th equation, and distribute it equally over each of the pixels that intersects L_i.

Suppose, now, that each b_i is positive, and we know in advance that the desired image we wish to reconstruct must be nonnegative. We can begin with $x^0 > 0$, but as we compute the ART steps, we may lose nonnegativity. One way to avoid this loss is to correct the current x^k multiplicatively, rather than additively, as in ART. This leads to the *multiplicative* ART (MART).

The MART, in this case, has the iterative step

$$x_j^{k+1} = x_j^k\Big(\frac{b_i}{(Ax^k)_i}\Big), \tag{9.14}$$

for those j in L_i, and

$$x_j^{k+1} = x_j^k, \tag{9.15}$$

otherwise. Therefore, we can write the iterative step as

$$x_j^{k+1} = x_j^k\Big(\frac{b_i}{(Ax^k)_i}\Big)^{A_{ij}}. \tag{9.16}$$

9.3.2 The MART in the General Case

Taking the entries of the matrix A to be either one or zero, depending on whether or not the j-th pixel is in the set L_i, is too crude. The line L_i may just clip a corner of one pixel, but pass through the center of another. Surely, it makes more sense to let A_{ij} be the length of the intersection of line L_i with the j-th pixel, or, perhaps, this length divided by the length of the diagonal of the pixel. It may also be more realistic to consider a strip, instead of a line. Other modifications to A_{ij} may made made, in order to

better describe the physics of the situation. Finally, all we can be sure of is that A_{ij} will be nonnegative, for each i and j. In such cases, what is the proper form for the MART?

The MART, which can be applied only to nonnegative systems, is a sequential, or row-action, method that uses one equation only at each step of the iteration.

Algorithm 9.4 (MART) *Let x^0 be any positive vector, and $i = k(\mathrm{mod}\, I)+1$. Having found x^k for positive integer k, define x^{k+1} by*

$$x_j^{k+1} = x_j^k \Big(\frac{b_i}{(Ax^k)_i}\Big)^{m_i^{-1}A_{ij}}, \tag{9.17}$$

where $m_i = \max\{A_{ij} \,|j = 1, 2, ..., J\}$.

Some treatments of MART leave out the m_i, but require only that the entries of A have been rescaled so that $A_{ij} \leq 1$ for all i and j. The m_i is important, however, in accelerating the convergence of MART.

The MART can be accelerated by relaxation, as well.

Algorithm 9.5 (Relaxed MART) *Let x^0 be any positive vector, and $i = k(\mathrm{mod}\, I) + 1$. Having found x^k for positive integer k, define x^{k+1} by*

$$x_j^{k+1} = x_j^k \Big(\frac{b_i}{(Ax^k)_i}\Big)^{\gamma_i m_i^{-1}A_{ij}}, \tag{9.18}$$

where γ_i is in the interval $(0, 1)$.

As with ART, finding the best relaxation parameters is a bit of an art.

9.3.3 Cross-Entropy

For $a > 0$ and $b > 0$, let the cross-entropy or Kullback-Leibler distance from a to b be

$$KL(a, b) = a \log \frac{a}{b} + b - a, \tag{9.19}$$

$KL(a, 0) = +\infty$, and $KL(0, b) = b$. Extend to nonnegative vectors coordinate-wise, so that

$$KL(x, z) = \sum_{j=1}^{J} KL(x_j, z_j). \tag{9.20}$$

Unlike the Euclidean distance, the KL distance is not symmetric; $KL(Ax, b)$ and $KL(b, Ax)$ are distinct, and we can obtain different approximate solutions of $Ax = b$ by minimizing these two distances with respect to nonnegative x.

9.3.4 Convergence of MART

In the consistent case, by which we mean that $Ax = b$ has nonnegative solutions, we have the following convergence theorem for MART.

Theorem 9.3 *In the consistent case, the MART converges to the unique nonnegative solution of $b = Ax$ for which the distance $\sum_{j=1}^{J} KL(x_j, x_j^0)$ is minimized.*

If the starting vector x^0 is the vector whose entries are all one, then the MART converges to the solution that maximizes the Shannon entropy,

$$SE(x) = \sum_{j=1}^{J} x_j \log x_j - x_j. \tag{9.21}$$

As with ART, the speed of convergence is greatly affected by the ordering of the equations, converging most slowly when consecutive equations correspond to nearly parallel hyperplanes.

Open Question: When there are no nonnegative solutions, MART does not converge to a single vector, but, like ART, is always observed to produce a limit cycle of vectors. Unlike ART, there is no proof of the existence of a limit cycle for MART.

Chapter 10

Transmission Tomography II

According to the Central Slice Theorem, if we have all the line integrals through the attenuation function $f(x, y)$ then we have the two-dimensional Fourier transform of $f(x, y)$. To get $f(x, y)$ we need to invert the two-dimensional Fourier transform.

10.1 Inverting the Fourier Transform

The Fourier-transform inversion formula for two-dimensional functions tells us that the function $f(x, y)$ can be obtained as

$$f(x, y) = \frac{1}{4\pi^2} \int \int F(u, v) e^{-i(xu+yv)} du dv. \tag{10.1}$$

We now derive alternative inversion formulas.

10.1.1 Ramp Filter, then Back-project

Expressing the double integral in Equation (10.1) in polar coordinates (ω, θ), with $\omega \geq 0$, $u = \omega \cos\theta$, and $v = \omega \sin\theta$, we get

$$f(x, y) = \frac{1}{4\pi^2} \int_0^{2\pi} \int_0^{\infty} F(u, v) e^{-i(xu+yv)} \omega d\omega d\theta,$$

or

$$f(x, y) = \frac{1}{4\pi^2} \int_0^{\pi} \int_{-\infty}^{\infty} F(u, v) e^{-i(xu+yv)} |\omega| d\omega d\theta.$$

Now write

$$F(u, v) = F(\omega \cos\theta, \omega \sin\theta) = R_f(\theta, \omega),$$

where $R_f(\theta,\omega)$ is the FT with respect to t of $r_f(\theta,t)$, so that

$$\int_{-\infty}^{\infty} F(u,v)e^{-i(xu+yv)}|\omega|d\omega = \int_{-\infty}^{\infty} R_f(\theta,\omega)|\omega|e^{-i\omega t}d\omega.$$

The function $g_f(\theta,t)$ defined for $t = x\cos\theta + y\sin\theta$ by

$$g_f(\theta, x\cos\theta + y\sin\theta) = \frac{1}{2\pi}\int_{-\infty}^{\infty} R_f(\theta,\omega)|\omega|e^{-i\omega t}d\omega \tag{10.2}$$

is the result of a linear filtering of $r_f(\theta,t)$ using a *ramp filter* with transfer function $H(\omega) = |\omega|$. Then,

$$f(x,y) = \frac{1}{2\pi}\int_0^{\pi} g_f(\theta, x\cos\theta + y\sin\theta)d\theta \tag{10.3}$$

gives $f(x,y)$ as the result of a *back-projection operator*; for every fixed value of (θ,t) add $g_f(\theta,t)$ to the current value at the point (x,y) for all (x,y) lying on the straight line determined by θ and t by $t = x\cos\theta + y\sin\theta$. The final value at a fixed point (x,y) is then the average of all the values $g_f(\theta,t)$ for those (θ,t) for which (x,y) is on the line $t = x\cos\theta + y\sin\theta$. It is therefore said that $f(x,y)$ can be obtained by *filtered back-projection* (FBP) of the line-integral data.

Knowing that $f(x,y)$ is related to the complete set of line integrals by filtered back-projection suggests that, when only finitely many line integrals are available, a similar ramp filtering and back-projection can be used to estimate $f(x,y)$; in the clinic this is the most widely used method for the reconstruction of tomographic images.

10.1.2 Back-project, then Ramp Filter

There is a second way to recover $f(x,y)$ using back-projection and filtering, this time in the reverse order; that is, we back-project the Radon transform and then ramp filter the resulting function of two variables. We begin again with the relation

$$f(x,y) = \frac{1}{4\pi^2}\int_0^{2\pi}\int_0^{\infty} F(u,v)e^{-i(xu+yv)}\omega d\omega d\theta,$$

which we write as

$$f(x,y) = \frac{1}{4\pi^2}\int_0^{2\pi}\int_0^{\infty} \frac{F(u,v)}{\sqrt{u^2+v^2}}\sqrt{u^2+v^2}e^{-i(xu+yv)}\omega d\omega d\theta$$

$$= \frac{1}{4\pi^2}\int_0^{2\pi}\int_0^{\infty} G(u,v)\sqrt{u^2+v^2}e^{-i(xu+yv)}\omega d\omega d\theta, \tag{10.4}$$

using

$$G(u,v) = \frac{F(u,v)}{\sqrt{u^2+v^2}}$$

for $(u,v) \neq (0,0)$. Equation (10.4) expresses $f(x,y)$ as the result of performing a two-dimensional ramp filtering of $g(x,y)$, the inverse Fourier transform of $G(u,v)$. We show now that $g(x,y)$ is the back-projection of the function $r_f(\omega,t)$; that is, we show that

$$g(x,y) = \frac{1}{2\pi}\int_0^\pi r_f(\theta, x\cos\theta + y\sin\theta)d\theta.$$

We have

$$\begin{aligned} g(x,y) &= \frac{1}{4\pi^2}\int_0^\pi\int_{-\infty}^{\infty} G(\omega\cos\theta,\omega\sin\theta)|\omega|e^{-i\omega(x\cos\theta+y\sin\theta)}d\omega d\theta \\ &= \frac{1}{4\pi^2}\int_0^\pi\int_{-\infty}^{\infty} F(\omega\cos\theta,\omega\sin\theta)e^{-i\omega(x\cos\theta+y\sin\theta)}d\omega d\theta \\ &= \frac{1}{4\pi^2}\int_0^\pi\int_{-\infty}^{\infty} R_f(\theta,\omega)e^{-i\omega(x\cos\theta+y\sin\theta)}d\omega d\theta \\ &= \frac{1}{2\pi}\int_0^\pi r_f(\theta, x\cos\theta + y\sin\theta)d\theta, \end{aligned}$$

as required.

10.1.3 Radon's Inversion Formula

To get Radon's inversion formula, we need two basic properties of the Fourier transform. First, if $f(x)$ has Fourier transform $F(\gamma)$ then the derivative $f'(x)$ has Fourier transform $-i\gamma F(\gamma)$. Second, if $F(\gamma) = \text{sgn}(\gamma)$, the function that is $\frac{\gamma}{|\gamma|}$ for $\gamma \neq 0$, and equal to zero for $\gamma = 0$, then its inverse Fourier transform is $f(x) = \frac{1}{i\pi x}$.

Writing equation (10.2) as

$$g_f(\theta,t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \omega R_f(\theta,\omega)\text{sgn}(\omega)e^{-i\omega t}d\omega,$$

we see that g_f is the inverse Fourier transform of the product of the two functions $\omega R_f(\theta,\omega)$ and $\text{sgn}(\omega)$. Consequently, g_f is the convolution of their individual inverse Fourier transforms, $i\frac{\partial}{\partial t}r_f(\theta,t)$ and $\frac{1}{i\pi t}$; that is,

$$g_f(\theta,t) = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{\partial}{\partial t}r_f(\theta,s)\frac{1}{t-s}ds,$$

which is the Hilbert transform of the function $\frac{\partial}{\partial t}r_f(\theta,t)$, with respect to the variable t. Radon's inversion formula is then

$$f(x,y) = \frac{1}{2\pi}\int_0^\pi HT(\frac{\partial}{\partial t}r_f(\theta,t))d\theta.$$

10.1.4 Practical Issues

Of course, we never have the Radon transform $r_f(\theta, t)$ for all values of its variables. Only finitely many angles θ are used, and, for each θ, we will have (approximate) values of line integrals for only finitely many t. Therefore, taking the Fourier transform of $r_f(\theta, t)$, as a function of the single variable t, is not something we can actually do. At best, we can approximate $R_f(\theta, \omega)$ for finitely many θ. From the Central Slice Theorem, we can then say that we have approximate values of $F(\omega\cos\theta, \omega\sin\theta)$, for finitely many θ. This means that we have (approximate) Fourier transform values for $f(x, y)$ along finitely many lines through the origin, like the spokes of a wheel. The farther from the origin we get, the fewer values we have, so the *coverage* in Fourier space is quite uneven. The low-spatial-frequencies are much better estimated than higher ones, meaning that we have a low-pass version of the desired $f(x, y)$. The filtered back-projection approaches we have just discussed both involve ramp filtering, in which the higher frequencies are increased, relative to the lower ones. This too can only be implemented approximately, since the data is noisy and careless ramp filtering will cause the reconstructed image to be unacceptably noisy.

10.2 Summary

We have seen how the problem of reconstructing a function from line integrals arises in transmission tomography. The Central Slice Theorem connects the line integrals and the Radon transform to the Fourier transform of the desired attenuation function. Various approaches to implementing the Fourier Inversion Formula lead to filtered back-projection algorithms for the reconstruction. In x-ray tomography, as well as in PET, viewing the data as line integrals ignores the statistical aspects of the problem, and in SPECT, it ignores, as well, the important physical effects of attenuation. To incorporate more of the physics of the problem, iterative algorithms based on statistical models have been developed. We shall consider some of these algorithms later.

Part III

Emission Tomography

Chapter 11

Emission Tomography I

In this next part of the text we take up the subject of emission tomography. In this chapter we describe the two modalities of emission tomography, *positron emission tomography* (PET) and *single photon emission computed tomography* (SPECT), and introduce the basic mathematical models for both.

11.1 Positron Emission Tomography

As we noted previously, detection in the PET case means the recording of two photons at nearly the same time at two different detectors. The locations of these two detectors then provide the end points of the line segment passing, more or less, through the site of the original positron emission. Therefore, each possible pair of detectors determines a *line of response* (LOR). When a LOR is recorded, it is assumed that a positron was emitted somewhere along that line. The PET data consists of a chronological list of LOR that are recorded.

Let the LOR be parameterized by the variable s, with $s = 0$ and $s = c$ denoting the two ends, and c the distance from one end to the other. For a fixed value $s = s_0$, let $P(s)$ be the probability of reaching s for a photon resulting from an emission at s_0. For small $\Delta s > 0$ the probability that a photon that reached s is absorbed in the interval $[s, s+\Delta s]$ is approximately $\mu(s)\Delta s$, where $\mu(s) \geq 0$ is the photon attenuation density at s. Then $P(s + \Delta s) \approx P(s)[1 - \mu(s)\Delta s]$, so that

$$P(s + \Delta s) - P(s) \approx -P(s)\mu(s)\Delta s.$$

Dividing by Δs and letting Δs go to zero, we get

$$P'(s) = -P(s)\mu(s).$$

It follows that

$$P(s) = e^{-\int_{s_0}^{s} \mu(t)dt}.$$

The probability that the photon will reach $s = c$ and be detected is then

$$P(c) = e^{-\int_{s_0}^{c} \mu(t)dt}.$$

Similarly, we find that the probability that a photon will succeed in reaching $s = 0$ from s_0 is

$$P(0) = e^{-\int_{0}^{s_0} \mu(t)dt}.$$

Since having one photon reach $s = 0$ and the other reach $s = c$ are independent events, their probabilities multiply, so that the probability that both photons reach their destinations and a coincident detection is recorded for this LOR is

$$e^{-\int_{0}^{c} \mu(t)dt}.$$

The expected number of coincident detections along the LOR is then proportional to

$$\int_0^c f(s)e^{-\int_0^c \mu(t)dt}ds = e^{-\int_0^c \mu(t)dt} \int_0^c f(s)ds, \tag{11.1}$$

where $f(s)$ is the intensity of radionuclide at s. Assuming we know the attenuation function $\mu(s)$, we can estimate the line integral $\int_0^c f(s)ds$ from the number of coincident detections recorded along the LOR. So, once again, we have line-integral data pertaining to the function of interest.

11.2 Single-Photon Emission Tomography

We remarked earlier that there are at least three degradations that need to be corrected before the line-integral model and FBP can be successfully applied in the SPECT case [143]: attenuation, scatter, and spatially dependent resolution. Some photons never reach the detectors because they are absorbed in the body. As in the PET case, correcting for attenuation requires knowledge of the patient's body; this knowledge can be obtained by performing a transmission scan at the same time. In contrast to the PET case, the attenuation due to absorption is more difficult to correct, since it does not involve merely the line integral of the attenuation function, but a half-line integral that depends on the distribution of matter between each photon source and each detector.

As in the PET case previously discussed, the probability that a photon emitted at the point on the line corresponding to the variable $s = s_0$ will reach $s = c$ and be detected is then

$$P(s_0) = e^{-\int_{s_0}^{c} \mu(t)dt}.$$

If $f(s)$ is the expected number of photons emitted from point s during the scanning, then the expected number of photons detected at c is proportional to

$$\int_0^c f(s)e^{-\int_s^c \mu(t)dt}ds. \tag{11.2}$$

Notice the difference between the integral in Equation (11.2) and the one in Equation (15.1).

The integral in Equation (11.2) varies with the line being considered; the resulting function of lines is called the *attenuated Radon transform*. If the attenuation function μ is constant, then the attenuated Radon transform is called the *exponential Radon transform*.

While some photons are absorbed within the body, others are first deflected and then detected; this is called *scatter*. Consequently, some of the detected photons do not come from where they seem to come from. The scattered photons often have reduced energy, compared to *primary*, or non-scattered, photons, and scatter correction can be based on this energy difference; see [143].

Finally, even if there were no attenuation and no scatter, it would be incorrect to view the detected photons as having originated along a straight line from the detector. The detectors have a cone of acceptance that widens as it recedes from the detector. This results in spatially varying resolution. There are mathematical ways to correct for both spatially varying resolution and uniform attenuation [199]. Correcting for the more realistic non-uniform and patient-specific attenuation is more difficult and is the subject of on-going research.

Spatially varying resolution complicates the quantitation problem, which is the effort to determine the exact amount of radionuclide present within a given region of the body, by introducing the *partial volume effect* and *spill-over* (see [210]). To a large extent, these problems are shortcomings of reconstruction based on the line-integral model. If we assume that all photons detected at a particular detector came from points within a narrow strip perpendicular to the camera face, and we reconstruct the image using this assumption, then photons coming from locations outside this strip will be incorrectly attributed to locations within the strip (spill-over), and therefore not correctly attributed to their true source location. If the true source location also has its counts raised by spill-over, the net effect may not be significant; if, however, the true source is a hot spot surrounded by cold background, it gets no spill-over from its neighbors and its true intensity value is underestimated, resulting in the partial-volume effect. The term "partial volume" indicates that the hot spot is smaller than the region that the line-integral model offers as the source of the emitted photons. One way to counter these effects is to introduce a description of the spatially dependent blur into the reconstruction, which is then performed

by iterative methods [180].

In the SPECT case, as in most such inverse problems, there is a trade-off to be made between careful modeling of the physical situation and computational tractability. The FBP method slights the physics in favor of computational simplicity and speed. In recent years, iterative methods that incorporate more of the physics have become competitive.

11.2.1 The Discrete Model

In iterative reconstruction we begin by *discretizing* the problem; that is, we imagine the region of interest within the patient to consist of finitely many tiny squares, called *pixels* for two-dimensional processing or cubes, called *voxels* for three-dimensional processing. In what follows we shall not distinguish the two cases, but as a linguistic shorthand, we shall refer to 'pixels' indexed by $j = 1, ..., J$. The detectors are indexed by $i = 1, ..., I$, the count obtained at detector i is denoted y_i, and the vector $\mathbf{y} = (y_1, ..., y_I)^T$ is our data. In practice, for the fully three-dimensional case, I and J can be several hundred thousand.

We imagine that each pixel j has its own level of concentration of radioactivity and these concentration levels are what we want to determine. Proportional to these concentration levels are the average rates of emission of photons; the average rate for j we denote by x_j. The goal is to determine the vector $\mathbf{x} = (x_1, ..., x_J)^T$ from $\mathbf{y}$.

11.2.2 Discrete Attenuated Radon Transform

To achieve our goal we must construct a model that relates $\mathbf{y}$ to $\mathbf{x}$. One way to do that is to discretize the attenuated Radon Transform [123, 204].

The objective is to describe the contribution to the count data from the intensity x_j at the jth pixel. We assume, for the moment, that all the radionuclide is concentrated within the jth pixel, and we compute the resulting attenuated Radon Transform. Following [123, 204], we adopt a ray model for detection, which means that corresponding to each detector is a line of acceptance and that all the counts recorded at that detector came from pixels that intersect this line. This is a simplification, of course, since each detector has a solid angle of acceptance, which leads to depth-dependent blur.

For notational simplicity, we suppose that the line of acceptance associated with the ith detector is parameterized by arc-length $s \geq 0$, with $s = c > 0$ corresponding to the point closest to the detector, within the body, $s = 0$ corresponding to the point farthest from the detector, at which the line leaves the body, $s = b < c$ the point closest to the detector within the jth pixel, and $s = a < b$ the point farthest from the detector at which

the line leaves the jth pixel. The length of the intersection of the jth pixel with the line is then $d_{ij} = b - a$.

We are assuming that all the radionuclide is within the jth pixel, with intensity distribution (proportional to) x_j, so the value at detector i of the attenuated Radon Transform is

$$A_{ij} = \int_a^b x_j e^{-\int_s^c \mu(t)dt} ds. \tag{11.3}$$

We assume that the attenuation is uniformly equal to $\mu_j \geq 0$ within the jth pixel, so we can write

$$A_{ij} = \int_a^b x_j e^{-\int_s^b \mu_j dt - \int_b^c \mu(t)dt} ds,$$

or

$$A_{ij} = x_j e^{-\int_b^c \mu(t)dt} \int_a^b e^{(s-b)\mu_j} ds.$$

If $\mu_j = 0$, then we have

$$A_{ij} = x_j e^{-\int_b^c \mu(t)dt} d_{ij},$$

while if $\mu_k > 0$ we have

$$A_{ij} = \left(x_j e^{-\int_b^c \mu(t)dt} d_{ij} \right) S_{ij},$$

where

$$S_{ij} = \frac{1}{d_{ij}} \int_a^b e^{(b-s)\mu_j} ds = \frac{1}{\mu_j d_{ij}} (1 - e^{-\mu_j d_{ij}}).$$

We can then write

$$A_{ij} = x_j W_{ij},$$

for each j and i.

Since the function

$$g(t) = \frac{1}{t}(1 - e^{-t})$$

is positive for positive t, $g(0) = 1$, and $g(+\infty) = 0$, it is reasonable to view S_{ij} as the survival proportion associated with the jth pixel and the line from the ith detector. Expanding the exponential in S_{ij} in a power series, we find that

$$S_{ij} = \frac{1}{\mu_j d_{ij}} (1 - e^{-\mu_j d_{ij}}) \approx 1 - \frac{1}{2} \mu_j d_{ij},$$

so that the loss proportion is approximately $\frac{1}{2}\mu_j d_{ij}$. If we were to adopt the decaying exponential model for a photon surviving its passage through the jth pixel, and assume all the radionuclide was initially at the far side

of the jth pixel, we would replace S_{ij} with $e^{-\mu_j d_{ij}}$, which is approximately $1-\mu_j d_{ij}$, so that the loss proportion is approximately $\mu_j d_{ij}$. This is twice the loss proportion that we got using the other model, and is larger because we are assuming that all the radionuclide in the jth pixel has to attempt to travel through the entire jth pixel, whereas, due to the spreading of the radionuclide throughout the pixel, the average journey through the pixel is only half of the length d_{ij}.

Having found the values W_{ij}, we form the matrix W having these entries and then find a non-negative solution of the system of equations $Wx = y$, using one of a number of iterative algorithms, including the EMML. Contrary to what is stated in [204], it may not be appropriate to consider W_{ij} as the probability that a photon emitted at the jth pixel is detected at the ith detector, even though $0 \leq W_{ij} \leq 1$ for each i and j. If viewed that way, it would be the case that

$$\sum_{i=1}^{I} W_{ij}$$

would be the probability of detecting a photon emitted from the jth pixel; we have no guarantee, however, that this sum is not greater than one.

It is significant that the authors in [204] realize that the EMML iterative algorithm can be used to find a non-negative solution of $Wx = y$, even though no stochastic model for the data is assumed in their derivation. Their development involves discretizing the attenuated Radon Transform, which involves no randomness, and viewing the count data as approximate values of this discrete function.

There is another approach that can be used to relate the count data to the intensity levels x_j. This other approach is based on a stochastic model, as we describe next.

11.2.3 A Stochastic Model

Another way to relate the count data to the intensities x_j is to adopt the model of *independent Poisson emitters*. For $i = 1, ..., I$ and $j = 1, ..., J$, denote by Z_{ij} the random variable whose value is to be the number of photons emitted from pixel j, and detected at detector i, during the scanning time. We assume that the members of the collection $\{Z_{ij} | i = 1, ..., I, j = 1, ..., J\}$ are independent. In keeping with standard practice in modeling radioactivity, we also assume that the Z_{ij} are Poisson-distributed.

Generally, the signal-to-noise ratio (SNR) is the ratio of the mean of a distribution to its standard deviation (the square root of the variance). In the case of the Poisson distribution, the variance and the mean are the same, so the SNR is the square root of the mean; therefore, the higher the mean the higher the SNR.

We assume that Z_{ij} is a Poisson random variable whose mean value (and variance) is $\lambda_{ij} = P_{ij}x_j$. Here the $x_j \geq 0$ is the average rate of emission from pixel j, as discussed previously, and $P_{ij} \geq 0$ is the probability that a photon emitted from pixel j will be detected at detector i. The calculation of the P_{ij} can be quite similar to the derivation of the W_{ij} in the previous subsection, with the exception that we do need to have

$$\sum_{i=1}^{I} P_{ij} \leq 1.$$

We then define the random variables $Y_i = \sum_{j=1}^{J} Z_{ij}$, the total counts to be recorded at detector i; our actual count y_i is then the observed value of the random variable Y_i. Note that the actual values of the individual Z_{ij} are not observable.

Any Poisson-distributed random variable has a mean equal to its variance. The *signal-to-noise ratio* (SNR) is usually taken to be the ratio of the mean to the standard deviation, which, in the Poisson case, is then the square root of the mean. Consequently, the Poisson SNR increases as the mean value increases, which points to the desirability (at least, statistically speaking) of higher dosages to the patient.

Having found the P_{ij}, we take P to be the matrix with these entries. Since Px is the vector of expected counts at the various detectors, and y is the vector of actual counts, trying to find a non-negative solution of the system $y = Px$ may not seem completely reasonable. However, this is what several well known iterative algorithms do, even ones such as the EMML that were not originally designed for this purpose.

11.2.4 Reconstruction as Parameter Estimation

The goal is to estimate the distribution of radionuclide intensity by calculating the vector $\mathbf{x}$. The entries of $\mathbf{x}$ are parameters and the data are instances of random variables, so the problem looks like a fairly standard parameter estimation problem of the sort studied in beginning statistics. One of the basic tools for statistical parameter estimation is likelihood maximization, which is playing an increasingly important role in medical imaging. There are several problems, however. One is that the number of parameters is quite large, as large as the number of data values, in most cases. Standard statistical parameter estimation usually deals with the estimation of a handful of parameters. Another problem is that we do not know what the P_{ij} are. These values will vary from one patient to the next, since whether or not a photon makes it from a given pixel to a given detector depends on the geometric relationship between detector i and pixel j, as well as what is in the patient's body between these two locations. If there are ribs or skull getting in the way, the probability of making it goes

down. If there are just lungs, the probability goes up. These values can change during the scanning process, when the patient moves. Some motion is unavoidable, such as breathing and the beating of the heart. Determining good values of the P_{ij} in the absence of motion, and correcting for the effects of motion, are important parts of SPECT image reconstruction.

Chapter 12

Urn Models for Tomography

There seems to be a tradition in physics of using simple models or examples involving urns and marbles to illustrate important principles. In keeping with that tradition, we give an urn model to illustrate various aspects of remote sensing, and apply the model to tomography.

12.1 The Urn Model for Remote Sensing

Suppose that we have J urns numbered $j = 1, ..., J$, each containing marbles of various colors. Suppose that there are I colors, numbered $i = 1, ..., I$. Suppose also that there is a box containing N small pieces of paper, and on each piece is written the number of one of the J urns. Assume that N is much larger than J. Assume that I know the precise contents of each urn. My objective is to determine the precise contents of the box, that is, to estimate the number of pieces of paper corresponding to each of the numbers $j = 1, ..., J$.

Out of my view, my assistant removes one piece of paper from the box, takes one marble from the indicated urn, announces to me the color of the marble, and then replaces both the piece of paper and the marble. This action is repeated many times, at the end of which I have a long list of colors. This list is my data, from which I must determine the contents of the box.

This is a form of remote sensing; what we have access to is not what we are really interested in, but only related to it in some way. Sometimes such data is called "incomplete data" , in contrast to the "complete data" , which would be the list of the actual urn numbers drawn from the box.

If all the marbles of one color are in a single urn, the problem is trivial;

when I hear a color, I know immediately which urn contained that marble. My list of colors is then a list of urn numbers; I have the complete data now. My best estimate of the number of pieces of paper containing the urn number j is then simply N times the proportion of draws that resulted in urn j being selected.

At the other extreme, suppose two urns had identical contents. Then I could not distinguish one urn from the other and would be unable to estimate more than the total number of pieces of paper containing either of the two urn numbers.

Generally, the more the contents of the urns differ, the easier the task of estimating the contents of the box. In remote sensing applications, these issues affect our ability to resolve individual components contributing to the data.

To introduce some mathematics, let us denote by x_j the proportion of the pieces of paper that have the number j written on them. Let P_{ij} be the proportion of the marbles in urn j that have the color i. Let y_i be the proportion of times the color i occurs on the list of colors. The expected proportion of times i occurs on the list is $E(y_i) = \sum_{j=1}^{J} P_{ij}x_j = (Px)_i$, where P is the I by J matrix with entries P_{ij} and x is the J by 1 column vector with entries x_j. A reasonable way to estimate x is to replace $E(y_i)$ with the actual y_i and solve the system of linear equations $y_i = \sum_{j=1}^{J} P_{ij}x_j$, $i = 1, ..., I$. Of course, we require that the x_j be nonnegative and sum to one, so special algorithms may be needed to find such solutions. If there are two urns, j_1 and j_2, such that P_{ij_1} and P_{ij_2} are nearly equal for all i, then we will have a hard time distinguishing x_{j_1} and x_{j_2}.

In a number of applications that fit this model, such as medical tomography, the values x_j are taken to be parameters, the data y_i are statistics, and the x_j are estimated by adopting a probabilistic model and maximizing the likelihood function. iterative algorithms, such as the expectation maximization (EMML) algorithm are often used for such problems.

12.2 The Urn Model in Tomography

Now we apply this simple model to transmission and emission tomography.

12.2.1 The Case of SPECT

In the SPECT case, let there be J pixels or voxels, numbered $j = 1, ..., J$ and I detectors, numbered $i = 1, ..., I$. Let P_{ij} be the probability that a photon emitted at pixel j will be detected at detector i; we assume these probabilities are known to us. Let y_i be the proportion of the total photon count that was recorded at the ith detector. Denote by x_j the (unknown) proportion of the total photon count that was emitted from

pixel j. Selecting an urn randomly is analogous to selecting which pixel will be the next to emit a photon. Learning the color of the marble is analogous to learning where the photon was detected; for simplicity we are assuming that all emitted photons are detected, but this is not essential. The data we have, the counts at each detector, constitute the "incomplete data" ; the "complete data" would be the counts of emissions from each of the J pixels.

If the pixels numbered j_1 and j_2 are neighbors, then we would expect P_{ij_1} and P_{ij_2} to be almost equal, for every i. This makes it difficult to estimate accurately the separate quantities x_{j_1} and x_{j_2}, which is a resolution problem.

We can determine the x_j by finding nonnegative solutions of the system $y_i = \sum_{j=1}^{J} P_{ij}x_j$; this is what the various iterative algorithms, such as MART, EMML and RBI-EMML, seek to do.

12.2.2 The Case of PET

In the PET case, let there be J pixels or voxels, numbered $j = 1, ..., J$ and I lines of response (LOR), numbered $i = 1, ..., I$. Let P_{ij} be the probability that a positron emitted at pixel j will result in a coincidence detection associated with LOR i; we assume these probabilities are known to us. Let y_i be the proportion of the total detections that was associated with the ith LOR. Denote by x_j the (unknown) proportion of the total count that was due to a positron emitted from pixel j. Selecting an urn randomly is analogous to selecting which pixel will be the next to emit a positron. Learning the color of the marble is analogous to learning which LOR was detected; again, for simplicity we are assuming that all emitted positrons are detected, but this is not essential. As in the SPECT case, we can determine the x_j by finding nonnegative solutions of the system $y_i = \sum_{j=1}^{J} P_{ij}x_j$.

12.2.3 The Case of Transmission Tomography

Assume that x-ray beams are sent along I line segments, numbered $i = 1, ..., I$, and that the initial strength of each beam is known. By measuring the final strength, we determine the drop in intensity due to absorption along the ith line segment. Associated with each line segment we then have the proportion of transmitted photons that were absorbed, but we do not know where along the line segment the absorption took place. The proportion of absorbed photons for each line is our data, and corresponds to the proportion of each color in the list. The rate of change of the intensity of the x-ray beam as it passes through the jth pixel is proportional to the intensity itself, to P_{ij}, the length of the ith segment that is within the jth pixel, and to x_j, the amount of attenuating material present in the jth

pixel. Therefore, the intensity of the x-ray beam leaving the jth pixel is the product of the intensity of the beam upon entering the jth pixel and the decay term, $e^{-P_{ij}x_j}$.

The "complete data" is the proportion of photons entering the jth pixel that were absorbed within it; the "incomplete data" is the proportion of photons sent along each line segment that were absorbed. Selecting the jth urn is analogous to having an absorption occurring at the jth pixel. Knowing that an absorption has occurred along the ith line segment does tell us that an absorption occurred at one of the pixels that intersections that line segment, but that is analogous to knowing that there are certain urns that are the only ones that contain the ith color.

The (measured) intensity of the beam at the end of the ith line segment is $e^{-(Px)_i}$ times the (known) intensity of the beam when it began its journey along the ith line segment. Taking logs, we obtain a system of linear equations which we can solve for the x_j.

12.3 Hidden Markov Models

Hidden Markov models (HMM) are increasingly important in speech processing, optical character recognition and DNA sequence analysis. In this section we illustrate HMM using a modification of the urn model.

Suppose, once again, that we have J urns, indexed by $j = 1, ..., J$ and I colors of marbles, indexed by $i = 1, ..., I$. Associated with each of the J urns is a box, containing a large number of pieces of paper, with the number of one urn written on each piece. My assistant selects one box, say the j_0th box, to start the experiment. He draws a piece of paper from that box, reads the number written on it, call it j_1, goes to the urn with the number j_1 and draws out a marble. He then announces the color. He then draws a piece of paper from box number j_1, reads the next number, say j_2, proceeds to urn number j_2, etc. After N marbles have been drawn, the only data I have is a list of colors, $\mathbf{c} = \{c_1, c_2, ..., c_N\}$.

According to the hidden Markov model, the probability that my assistant will proceed from the urn numbered k to the urn numbered j is b_{jk}, with $\sum_{j=1}^{J} b_{jk} = 1$ for all k, and the probability that the color c_i will be drawn from the urn numbered j is a_{ij}, with $\sum_{i=1}^{I} a_{ij} = 1.$ for all j. The colors announced are the *visible states*, while the unannounced urn numbers are the *hidden states*.

There are several distinct objectives one can have, when using HMM. We assume that the data is the list of colors, $\mathbf{c}$.

- **Evaluation:** For given probabilities a_{ij} and b_{jk}, what is the probability that the list $\mathbf{c}$ was generated according to the HMM? Here, the objective is to see if the model is a good description of the data.

- **Decoding:** Given the model, the probabilities and the list **c**, what list $\mathbf{j} = \{j_1, j_2, ..., j_N\}$ of potential visited urns is the most likely? Now, we want to infer the hidden states from the visible ones.

- **Learning:** We are told that there are J urns and I colors, but are not told the probabilities a_{ij} and b_{jk}. We are given several data vectors **c** generated by the HMM; these are the *training sets*. The objective is to learn the probabilities.

Once again, the EMML algorithm can play a role in solving these problems [96].

Chapter 13

Block-Iterative Methods

Image reconstruction problems in tomography are often formulated as statistical likelihood maximization problems in which the pixel values of the desired image play the role of parameters. Iterative algorithms based on cross-entropy minimization, such as the *expectation maximization maximum likelihood* (EMML) method and the *simultaneous multiplicative algebraic reconstruction technique* (SMART) can be used to solve such problems. Because the EMML and SMART are slow to converge for large amounts of data typical in imaging problems, acceleration of the algorithms using blocks of data or ordered subsets has become popular. There are a number of different ways to formulate these block-iterative versions of EMML and SMART, involving the choice of certain normalization and regularization parameters. These methods are not faster merely because they are block-iterative; the correct choice of the parameters is crucial. The purpose of this chapter is to discuss these different formulations in detail sufficient to reveal the precise roles played by the parameters and to guide the user in choosing them.

13.1 Overview

The algorithms we discuss here have interesting histories, which we sketch in this section.

13.1.1 The SMART and its variants

Like the ART, the MART has a simultaneous version, called the SMART. Like MART, SMART applies only to nonnegative systems of equations $Ax = b$. Unlike MART, SMART is a simultaneous algorithm that uses all equations in each step of the iteration. The SMART was discovered in 1972, independently, by Darroch and Ratcliff, working in statistics, [87]

and by Schmidlin [191] in medical imaging; neither work makes reference to MART. Darroch and Ratcliff do consider block-iterative versions of their algorithm, in which only some of the equations are used at each step, but their convergence proof involves unnecessary restrictions on the system matrix. Censor and Segman [71] seem to be the first to present the SMART and its block-iterative variants explicitly as generalizations of MART.

13.1.2 The EMML and its variants

The *expectation maximization maximum likelihood* (EMML) method turns out to be closely related to the SMART, although it has quite a different history. The EMML algorithm we discuss here is actually a special case of a more general approach to likelihood maximization, usually called the EM algorithm [89]; the book by McLachnan and Krishnan [162] is a good source for the history of this more general algorithm.

It was noticed by Rockmore and Macovski [188] that the image reconstruction problems posed by medical tomography could be formulated as statistical parameter estimation problems. Following up on this idea, Shepp and Vardi [193] suggested the use of the EM algorithm for solving the reconstruction problem in emission tomography. In [150], Lange and Carson presented an EM-type iterative method for transmission tomographic image reconstruction, and pointed out a gap in the convergence proof given in [193] for the emission case. In [208], Vardi, Shepp and Kaufman repaired the earlier proof, relying on techniques due to Csiszár and Tusnády [85]. In [151] Lange, Bahn and Little improve the transmission and emission algorithms, by including regularization to reduce the effects of noise. The question of uniqueness of the solution in the inconsistent case was resolved in [43].

The MART and SMART were initially designed to apply to consistent systems of equations. Darroch and Ratcliff did not consider what happens in the inconsistent case, in which the system of equations has no non-negative solutions; this issue was resolved in [43], where it was shown that the SMART converges to a non-negative minimizer of the Kullback-Leibler distance $KL(Ax, b)$. The EMML, as a statistical parameter estimation technique, was not originally thought to be connected to any system of linear equations. In [43] it was shown that the EMML leads to a non-negative minimizer of the Kullback-Leibler distance $KL(b, Ax)$, thereby exhibiting a close connection between the SMART and the EMML methods. Consequently, when the non-negative system of linear equations $Ax = b$ has a non-negative solution, the EMML converges to such a solution.

13.1.3 Block-iterative Versions of SMART and EMML

As we have seen, Darroch and Ratcliff included what are now called block-iterative versions of SMART in their original paper [87]. Censor and Segman [71] viewed SMART and its block-iterative versions as natural extension of the MART. Consequently, block-iterative variants of SMART have been around for some time. The story with the EMML is quite different.

The paper of Holte, Schmidlin, *et al.* [134] compares the performance of Schmidlin's method of [191] with the EMML algorithm. Almost as an aside, they notice the accelerating effect of what they call *projection interleaving*, that is, the use of blocks. This paper contains no explicit formulas, however, and presents no theory, so one can only make educated guesses as to the precise iterative methods employed. Somewhat later, Hudson, Hutton and Larkin [135, 136] observed that the EMML can be significantly accelerated if, at each step, one employs only some of the data. They referred to this approach as the *ordered subset* EM method (OSEM). They gave a proof of convergence of the OSEM, for the consistent case. The proof relied on a fairly restrictive relationship between the matrix A and the choice of blocks, called *subset balance*. In [46] a revised version of the OSEM, called the *rescaled block-iterative* EMML (RBI-EMML), was shown to converge, in the consistent case, regardless of the choice of blocks.

13.1.4 Basic assumptions

Methods based on cross-entropy, such as the MART, SMART, EMML and all block-iterative versions of these algorithms apply to nonnegative systems that we denote by $Ax = b$, where b is a vector of positive entries, A is a matrix with entries $A_{ij} \geq 0$ such that for each j the sum $s_j = \sum_{i=1}^{I} A_{ij}$ is positive and we seek a solution x with nonnegative entries. If no nonnegative x satisfies $b = Ax$ we say the system is *inconsistent*.

Simultaneous iterative algorithms employ all of the equations at each step of the iteration; block-iterative methods do not. For the latter methods we assume that the index set $\{i = 1, ..., I\}$ is the (not necessarily disjoint) union of the N sets or *blocks* B_n, $n = 1, ..., N$. We shall require that $s_{nj} = \sum_{i \in B_n} A_{ij} > 0$ for each n and each j. Block-iterative methods like ART and MART for which each block consists of precisely one element are called *row-action* or *sequential* methods. We begin our discussion with the SMART and the EMML method.

13.2 The SMART and the EMML method

Both the SMART and the EMML method provide a solution of $b = Ax$ when such exist and (distinct) approximate solutions in the inconsistent case. The SMART algorithm is the following:

Algorithm 13.1 (SMART) *Let x^0 be an arbitrary positive vector. For $k = 0, 1, ...$ let*

$$x_j^{k+1} = x_j^k \exp\Big(s_j^{-1} \sum_{i=1}^{I} A_{ij} \log \frac{b_i}{(Ax^k)_i}\Big). \tag{13.1}$$

The exponential and logarithm in the SMART iterative step are computationally expensive. The EMML method is similar to the SMART, but somewhat less costly to compute.

Algorithm 13.2 (EMML) *Let x^0 be an arbitrary positive vector. For $k = 0, 1, ...$ let*

$$x_j^{k+1} = x_j^k s_j^{-1} \sum_{i=1}^{I} A_{ij} \frac{b_i}{(Ax^k)_i}. \tag{13.2}$$

The main results concerning the SMART are given by the following theorem.

Theorem 13.1 *In the consistent case the SMART converges to the unique nonnegative solution of $b = Ax$ for which the distance $\sum_{j=1}^{J} s_j KL(x_j, x_j^0)$ is minimized. In the inconsistent case it converges to the unique nonnegative minimizer of the distance $KL(Ax, y)$ for which $\sum_{j=1}^{J} s_j KL(x_j, x_j^0)$ is minimized; if A and every matrix derived from A by deleting columns has full rank then there is a unique nonnegative minimizer of $KL(Ax, y)$ and at most $I - 1$ of its entries are nonzero.*

For the EMML method the main results are the following.

Theorem 13.2 *In the consistent case the EMML algorithm converges to nonnegative solution of $b = Ax$. In the inconsistent case it converges to a nonnegative minimizer of the distance $KL(y, Ax)$; if A and every matrix derived from A by deleting columns has full rank then there is a unique nonnegative minimizer of $KL(y, Ax)$ and at most $I - 1$ of its entries are nonzero.*

In the consistent case there may be multiple nonnegative solutions and the one obtained by the EMML algorithm will depend on the starting vector x^0; how it depends on x^0 is an open question.

These theorems are special cases of more general results on block-iterative methods that we shall prove later in this chapter.

Both the EMML and SMART are related to likelihood maximization. Minimizing the function $KL(y, Ax)$ is equivalent to maximizing the likelihood when the b_i are taken to be measurements of independent Poisson random variables having means $(Ax)_i$. The entries of x are the parameters

to be determined. This situation arises in emission tomography. So the EMML is a likelihood maximizer, as its name suggests.

The connection between SMART and likelihood maximization is a bit more convoluted. Suppose that $s_j = 1$ for each j. The solution of $b = Ax$ for which $KL(x, x^0)$ is minimized necessarily has the form

$$x_j = x_j^0 \exp\Big(\sum_{i=1}^{I} A_{ij}\lambda_i\Big) \tag{13.3}$$

for some vector λ with entries λ_i. This *log linear* form also arises in transmission tomography, where it is natural to assume that $s_j = 1$ for each j and $\lambda_i \leq 0$ for each i. We have the following lemma that helps to connect the SMART algorithm with the transmission tomography problem:

Lemma 13.1 *Minimizing $KL(d, x)$ over x as in Equation (13.3) is equivalent to minimizing $KL(x, x^0)$, subject to $Ax = Ad$.*

The solution to the latter problem can be obtained using the SMART.

With $x_+ = \sum_{j=1}^{J} x_j$ the vector A with entries $p_j = x_j/x_+$ is a probability vector. Let $d = (d_1, ..., d_J)^T$ be a vector whose entries are nonnegative integers, with $K = \sum_{j=1}^{J} d_j$. Suppose that, for each j, p_j is the probability of index j and d_j is the number of times index j was chosen in K trials. The likelihood function of the parameters λ_i is

$$L(\lambda) = \prod_{j=1}^{J} p_j^{d_j} \tag{13.4}$$

so that the log-likelihood function is

$$LL(\lambda) = \sum_{j=1}^{J} d_j \log p_j. \tag{13.5}$$

Since A is a probability vector, maximizing $L(\lambda)$ is equivalent to minimizing $KL(d, p)$ with respect to λ, which, according to the lemma above, can be solved using SMART. In fact, since all of the block-iterative versions of SMART have the same limit whenever they have the same starting vector, any of these methods can be used to solve this maximum likelihood problem. In the case of transmission tomography the λ_i must be nonpositive, so if SMART is to be used, some modification is needed to obtain such a solution.

Those who have used the SMART or the EMML on sizable problems have certainly noticed that they are both slow to converge. An important issue, therefore, is how to accelerate convergence. One popular method is through the use of *block-iterative* (or *ordered subset*) methods.

13.3 Ordered-Subset Versions

To illustrate block-iterative methods and to motivate our subsequent discussion we consider now the *ordered subset* EM algorithm (OSEM), which is a popular technique in some areas of medical imaging, as well as an analogous version of SMART, which we shall call here the OSSMART. The OSEM is now used quite frequently in tomographic image reconstruction, where it is acknowledged to produce usable images significantly faster then EMML. From a theoretical perspective both OSEM and OSSMART are incorrect. How to correct them is the subject of much that follows here.

The idea behind the OSEM (OSSMART) is simple: the iteration looks very much like the EMML (SMART), but at each step of the iteration the summations are taken only over the current block. The blocks are processed cyclically.

The OSEM iteration is the following: for $k = 0, 1, ...$ and $n = k(\mathrm{mod}\, N)+1$, having found x^k let

OSEM:

$$x_j^{k+1} = x_j^k s_{nj}^{-1} \sum_{i \in B_n} A_{ij} \frac{b_i}{(Ax^k)_i}. \tag{13.6}$$

The OSSMART has the following iterative step:

OSSMART

$$x_j^{k+1} = x_j^k \exp\Big(s_{nj}^{-1} \sum_{i \in B_n} A_{ij} \log \frac{b_i}{(Ax^k)_i}\Big). \tag{13.7}$$

In general we do not expect block-iterative algorithms to converge in the inconsistent case, but to exhibit *subsequential convergence* to a *limit cycle*, as we shall discuss later. We do, however, want them to converge to a solution in the consistent case; the OSEM and OSSMART fail to do this except when the matrix A and the set of blocks $\{B_n, n = 1, ..., N\}$ satisfy the condition known as *subset balance*, which means that the sums s_{nj} depend only on j and not on n. While this may be approximately valid in some special cases, it is overly restrictive, eliminating, for example, almost every set of blocks whose cardinalities are not all the same. When the OSEM does well in practice in medical imaging it is probably because the N is not large and only a few iterations are carried out.

The experience with the OSEM was encouraging, however, and strongly suggested that an equally fast, but mathematically correct, block-iterative version of EMML was to be had; this is the *rescaled block-iterative* EMML (RBI-EMML). Both RBI-EMML and an analogous corrected version of OSSMART, the RBI-SMART, provide fast convergence to a solution in the consistent case, for any choice of blocks.

13.4 The RBI-SMART

We turn next to the block-iterative versions of the SMART, which we shall denote BI-SMART. These methods were known prior to the discovery of RBI-EMML and played an important role in that discovery; the importance of rescaling for acceleration was apparently not appreciated, however.

We start by considering a formulation of BI-SMART that is general enough to include all of the variants we wish to discuss. As we shall see, this formulation is too general and will need to be restricted in certain ways to obtain convergence. Let the iterative step be

$$x_j^{k+1} = x_j^k \exp\Big(\beta_{nj} \sum_{i \in B_n} \alpha_{ni} A_{ij} \log\Big(\frac{b_i}{(Ax^k)_i}\Big)\Big), \tag{13.8}$$

for $j = 1, 2, ..., J$, $n = k(\mathrm{mod}\, N) + 1$ and β_{nj} and α_{ni} positive. As we shall see, our convergence proof will require that β_{nj} be separable, that is, $b_{nj} = \gamma_j \delta_n$ for each j and n and that

$$\gamma_j \delta_n \sigma_{nj} \leq 1, \tag{13.9}$$

for $\sigma_{nj} = \sum_{i \in B_n} \alpha_{ni} A_{ij}$. With these conditions satisfied we have the following result.

Theorem 13.3 *Let x be a nonnegative solution of $b = Ax$. For any positive vector x^0 and any collection of blocks $\{B_n, n = 1, ..., N\}$ the sequence $\{x^k\}$ given by Equation (13.8) converges to the unique solution of $b = Ax$ for which the weighted cross-entropy $\sum_{j=1}^J \gamma_j^{-1} KL(x_j, x_j^0)$ is minimized.*

The inequality in the following lemma is the basis for the convergence proof.

Lemma 13.2 *Let $b = Ax$ for some nonnegative x. Then for $\{x^k\}$ as in Equation (13.8) we have*

$$\sum_{j=1}^J \gamma_j^{-1} KL(x_j, x_j^k) - \sum_{j=1}^J \gamma_j^{-1} KL(x_j, x_j^{k+1}) \geq \tag{13.10}$$

$$\delta_n \sum_{i \in B_n} \alpha_{ni} KL(b_i, (Ax^k)_i). \tag{13.11}$$

Proof: First note that

$$x_j^{k+1} = x_j^k \exp\Big(\gamma_j \delta_n \sum_{i \in B_n} \alpha_{ni} A_{ij} \log\Big(\frac{b_i}{(Ax^k)_i}\Big)\Big), \tag{13.12}$$

and

$$\exp\Big(\gamma_j \delta_n \sum_{i \in B_n} \alpha_{ni} A_{ij} \log\Big(\frac{b_i}{(Ax^k)_i}\Big)\Big) \tag{13.13}$$

can be written as

$$\exp\Big((1-\gamma_j\delta_n\sigma_{nj})\log 1+\gamma_j\delta_n\sum_{i\in B_n}\alpha_{ni}A_{ij}\log\Big(\frac{b_i}{(Ax^k)_i}\Big)\Big), \tag{13.14}$$

which, by the convexity of the exponential function, is not greater than

$$(1-\gamma_j\delta_n\sigma_{nj})+\gamma_j\delta_n\sum_{i\in B_n}\alpha_{ni}A_{ij}\frac{b_i}{(Ax^k)_i}. \tag{13.15}$$

It follows that

$$\sum_{j=1}^{J}\gamma_j^{-1}(x_j^k-x_j^{k+1})\geq\delta_n\sum_{i\in B_n}\alpha_{ni}((Ax^k)_i-b_i). \tag{13.16}$$

We also have

$$\log(x_j^{k+1}/x_j^k)=\gamma_j\delta_n\sum_{i\in B_n}\alpha_{ni}A_{ij}\log\frac{b_i}{(Ax^k)_i}. \tag{13.17}$$

Therefore

$$\sum_{j=1}^{J}\gamma_j^{-1}KL(x_j,x_j^k)-\sum_{j=1}^{J}\gamma_j^{-1}KL(x_j,x_j^{k+1}) \tag{13.18}$$

$$=\sum_{j=1}^{J}\gamma_j^{-1}(x_j\log(x_j^{k+1}/x_j^k)+x_j^k-x_j^{k+1}) \tag{13.19}$$

$$=\sum_{j=1}^{J}x_j\delta_n\sum_{i\in B_n}\alpha_{ni}A_{ij}\log\frac{b_i}{(Ax^k)_i}+\sum_{j=1}^{J}\gamma_j^{-1}(x_j^k-x_j^{k+1}) \tag{13.20}$$

$$=\delta_n\sum_{i\in B_n}\alpha_{ni}(\sum_{j=1}^{J}x_jA_{ij})\log\frac{b_i}{(Ax^k)_i}+\sum_{j=1}^{J}\gamma_j^{-1}(x_j^k-x_j^{k+1}) \tag{13.21}$$

$$\geq\delta_n\Big(\sum_{i\in B_n}\alpha_{ni}(b_i\log\frac{b_i}{(Ax^k)_i}+(Ax^k)_i-b_i)\Big)=\delta_n\sum_{i\in B_n}\alpha_{ni}KL(b_i,(Ax^k)_i). \tag{13.22}$$

This completes the proof of the lemma. ∎

From the inequality (13.11) we conclude that the sequence

$$\{\sum_{j=1}^{J}\gamma_j^{-1}KL(x_j,x_j^k)\} \tag{13.23}$$

is decreasing, that $\{x^k\}$ is therefore bounded and the sequence

$$\{\sum_{i\in B_n}\alpha_{ni}KL(b_i,(Ax^k)_i)\} \tag{13.24}$$

is converging to zero. Let x^* be any cluster point of the sequence $\{x^k\}$. Then it is not difficult to show that $b = Ax^*$. Replacing x with x^* we have that the sequence $\{\sum_{j=1}^{J}\gamma_j^{-1}KL(x_j^*,x_j^k)\}$ is decreasing; since a subsequence converges to zero, so does the whole sequence. Therefore x^* is the limit of the sequence $\{x^k\}$. This proves that the algorithm produces a solution of $b = Ax$. To conclude further that the solution is the one for which the quantity $\sum_{j=1}^{J}\gamma_j^{-1}KL(x_j,x_j^0)$ is minimized requires further work to replace the inequality (13.11) with an equation in which the right side is independent of the particular solution x chosen; see the final section of this chapter for the details.

We see from the theorem that how we select the γ_j is determined by how we wish to weight the terms in the sum $\sum_{j=1}^{J}\gamma_j^{-1}KL(x_j,x_j^0)$. In some cases we want to minimize the cross-entropy $KL(x,x^0)$ subject to $b = Ax$; in this case we would select $\gamma_j = 1$. In other cases we may have some prior knowledge as to the relative sizes of the x_j and wish to emphasize the smaller values more; then we may choose γ_j proportional to our prior estimate of the size of x_j. Having selected the γ_j, we see from the inequality (13.11) that convergence will be accelerated if we select δ_n as large as permitted by the condition $\gamma_j\delta_n\sigma_{nj} \leq 1$. This suggests that we take

$$\delta_n = 1/\min\{\sigma_{nj}\gamma_j,\ j = 1,...,J\}. \tag{13.25}$$

The *rescaled* BI-SMART (RBI-SMART) as presented in [45, 47, 48] uses this choice, but with $\alpha_{ni} = 1$ for each n and i. For each $n = 1,...,N$ let

$$m_n = \max\{s_{nj}s_j^{-1}|j = 1,...,J\}. \tag{13.26}$$

The original RBI-SMART is as follows:

Algorithm 13.3 (RBI-SMART) *Let x^0 be an arbitrary positive vector. For $k = 0, 1, ...$, let $n = k(\mathrm{mod}\, N) + 1$. Then let*

$$x_j^{k+1} = x_j^k \exp\Big(m_n^{-1}s_j^{-1}\sum_{i\in B_n}A_{ij}\log\Big(\frac{b_i}{(Ax^k)_i}\Big)\Big). \tag{13.27}$$

Notice that Equation (13.27) can be written as

$$\log x_j^{k+1} = (1 - m_n^{-1} s_j^{-1} s_{nj}) \log x_j^k + m_n^{-1} s_j^{-1} \sum_{i \in B_n} A_{ij} \log \Big(x_j^k \frac{b_i}{(Ax^k)_i} \Big), \tag{13.28}$$

from which we see that x_j^{k+1} is a weighted geometric mean of x_j^k and the terms

$$(Q_i x^k)_j = x_j^k \Big(\frac{b_i}{(Ax^k)_i} \Big),$$

for $i \in B_n$. This will be helpful in deriving block-iterative versions of the EMML algorithm. The vectors $Q_i(x^k)$ are sometimes called weighted KL projections.

Let's look now at some of the other choices for these parameters that have been considered in the literature.

First, we notice that the OSSMART does not generally satisfy the requirements, since in (13.7) the choices are $\alpha_{ni} = 1$ and $\beta_{nj} = s_{nj}^{-1}$; the only times this is acceptable is if the s_{nj} are separable; that is, $s_{nj} = r_j t_n$ for some r_j and t_n. This is slightly more general than the condition of subset balance and is sufficient for convergence of OSSMART.

In [71] Censor and Segman make the choices $\beta_{nj} = 1$ and $\alpha_{ni} > 0$ such that $\sigma_{nj} \leq 1$ for all n and j. In those cases in which σ_{nj} is much less than 1 for each n and j their iterative scheme is probably excessively relaxed; it is hard to see how one might improve the rate of convergence by altering only the weights α_{ni}, however. Limiting the choice to $\gamma_j \delta_n = 1$ reduces our ability to accelerate this algorithm.

The original SMART in Equation (13.1) uses $N = 1$, $\gamma_j = s_j^{-1}$ and $\alpha_{ni} = \alpha_i = 1$. Clearly the inequality (13.9) is satisfied; in fact it becomes an equality now.

For the row-action version of SMART, the *multiplicative* ART (MART), due to Gordon, Bender and Herman [120], we take $N = I$ and $B_n = B_i = \{i\}$ for $i = 1, ..., I$. The MART has the iterative

$$x_j^{k+1} = x_j^k \Big(\frac{b_i}{(Ax^k)_i} \Big)^{m_i^{-1} A_{ij}}, \tag{13.29}$$

for $j = 1, 2, ..., J$, $i = k (\mathrm{mod}\, I) + 1$ and $m_i > 0$ chosen so that $m_i^{-1} A_{ij} \leq 1$ for all j. The smaller m_i is the faster the convergence, so a good choice is $m_i = \max\{A_{ij}|,\ j = 1, ..., J\}$. Although this particular choice for m_i is not explicitly mentioned in the various discussions of MART I have seen, it was used in implementations of MART from the beginning [130].

Darroch and Ratcliff included a discussion of a block-iterative version of SMART in their 1972 paper [87]. Close inspection of their version reveals that they require that $s_{nj} = \sum_{i \in B_n} A_{ij} = 1$ for all j. Since this is unlikely

to be the case initially, we might try to rescale the equations or unknowns to obtain this condition. However, unless $s_{nj} = \sum_{i \in B_n} A_{ij}$ depends only on j and not on n, which is the *subset balance* property used in [136], we cannot redefine the unknowns in a way that is independent of n.

The MART fails to converge in the inconsistent case. What is always observed, but for which no proof exists, is that, for each fixed $i = 1, 2, ..., I$, as $m \to +\infty$, the MART subsequences $\{x^{mI+i}\}$ converge to separate limit vectors, say $x^{\infty,i}$. This *limit cycle* LC $= \{x^{\infty,i} | i = 1, ..., I\}$ reduces to a single vector whenever there is a nonnegative solution of $b = Ax$. The greater the minimum value of $KL(Ax, y)$ the more distinct from one another the vectors of the limit cycle are. An analogous result is observed for BI-SMART.

13.5 The RBI-EMML

As we did with SMART, we consider now a formulation of BI-EMML that is general enough to include all of the variants we wish to discuss. Once again, the formulation is too general and will need to be restricted in certain ways to obtain convergence. Let the iterative step be

$$x_j^{k+1} = x_j^k(1 - \beta_{nj}\sigma_{nj}) + x_j^k \beta_{nj} \sum_{i \in B_n} \alpha_{ni} A_{ij} \frac{b_i}{(Ax^k)_i}, \tag{13.30}$$

for $j = 1, 2, ..., J$, $n = k(\mathrm{mod}\, N)+1$ and β_{nj} and α_{ni} positive. As in the case of BI-SMART, our convergence proof will require that β_{nj} be separable, that is,

$$b_{nj} = \gamma_j \delta_n \tag{13.31}$$

for each j and n and that the inequality (13.9) hold. With these conditions satisfied we have the following result.

Theorem 13.4 *Let x be a nonnegative solution of $b = Ax$. For any positive vector x^0 and any collection of blocks $\{B_n, n = 1, ..., N\}$ the sequence $\{x^k\}$ given by Equation (13.8) converges to a nonnegative solution of $b = Ax$.*

When there are multiple nonnegative solutions of $b = Ax$ the solution obtained by BI-EMML will depend on the starting point x^0, but precisely how it depends on x^0 is an open question. Also, in contrast to the case of BI-SMART, the solution can depend on the particular choice of the blocks. The inequality in the following lemma is the basis for the convergence proof.

Lemma 13.3 *Let $b = Ax$ for some nonnegative x. Then for $\{x^k\}$ as in*

Equation (13.30) we have

$$\sum_{j=1}^{J} \gamma_j^{-1} KL(x_j, x_j^k) - \sum_{j=1}^{J} \gamma_j^{-1} KL(x_j, x_j^{k+1}) \geq \tag{13.32}$$

$$\delta_n \sum_{i \in B_n} \alpha_{ni} KL(b_i, (Ax^k)_i). \tag{13.33}$$

Proof: From the iterative step

$$x_j^{k+1} = x_j^k (1 - \gamma_j \delta_n \sigma_{nj}) + x_j^k \gamma_j \delta_n \sum_{i \in B_n} \alpha_{ni} A_{ij} \frac{b_i}{(Ax^k)_i} \tag{13.34}$$

we have

$$\log(x_j^{k+1}/x_j^k) = \log\Big((1 - \gamma_j \delta_n \sigma_{nj}) + \gamma_j \delta_n \sum_{i \in B_n} \alpha_{ni} A_{ij} \frac{b_i}{(Ax^k)_i}\Big). \tag{13.35}$$

By the concavity of the logarithm we obtain the inequality

$$\log(x_j^{k+1}/x_j^k) \geq \Big((1 - \gamma_j \delta_n \sigma_{nj}) \log 1 + \gamma_j \delta_n \sum_{i \in B_n} \alpha_{ni} A_{ij} \log \frac{b_i}{(Ax^k)_i}\Big), \tag{13.36}$$

or

$$\log(x_j^{k+1}/x_j^k) \geq \gamma_j \delta_n \sum_{i \in B_n} \alpha_{ni} A_{ij} \log \frac{b_i}{(Ax^k)_i}. \tag{13.37}$$

Therefore

$$\sum_{j=1}^{J} \gamma_j^{-1} x_j \log(x_j^{k+1}/x_j^k) \geq \delta_n \sum_{i \in B_n} \alpha_{ni} (\sum_{j=1}^{J} x_j A_{ij}) \log \frac{b_i}{(Ax^k)_i}. \tag{13.38}$$

Note that it is at this step that we used the separability of the β_{nj}. Also

$$\sum_{j=1}^{J} \gamma_j^{-1} (x_j^{k+1} - x_j^k) = \delta_n \sum_{i \in B_n} ((Ax^k)_i - b_i). \tag{13.39}$$

This concludes the proof of the lemma. ∎

From the inequality in (13.33) we conclude, as we did in the BI-SMART case, that the sequence $\{\sum_{j=1}^{J} \gamma_j^{-1} KL(x_j, x_j^k)\}$ is decreasing, that $\{x^k\}$ is therefore bounded and the sequence $\{\sum_{i \in B_n} \alpha_{ni} KL(b_i, (Ax^k)_i)\}$ is converging to zero. Let x^* be any cluster point of the sequence $\{x^\}$. Then it is

not difficult to show that $b = Ax^*$. Replacing x with x^* we have that the sequence $\{\sum_{j=1}^J \gamma_j^{-1} KL(x_j^*, x_j^k)\}$ is decreasing; since a subsequence converges to zero, so does the whole sequence. Therefore x^* is the limit of the sequence $\{x^k\}$. This proves that the algorithm produces a nonnegative solution of $b = Ax$. So far, we have been unable to replace the inequality in (13.33) with an equation in which the right side is independent of the particular solution x chosen.

Having selected the γ_j, we see from the inequality in (13.33) that convergence will be accelerated if we select δ_n as large as permitted by the condition $\gamma_j \delta_n \sigma_{nj} \leq 1$. This suggests that once again we take

$$\delta_n = 1/\min\{\sigma_{nj}\gamma_j,\, j = 1, ..., J\}. \tag{13.40}$$

The *rescaled* BI-EMML (RBI-EMML) as presented in [45, 47, 48] uses this choice, but with $\alpha_{ni} = 1$ for each n and i. The original motivation for the RBI-EMML came from consideration of Equation (13.28), replacing the geometric means with arithmetic means. This RBI-EMML is as follows:

Algorithm 13.4 (RBI-EMML) *Let x^0 be an arbitrary positive vector. For $k = 0, 1, ...$, let $n = k(\mathrm{mod}\, N) + 1$. Then let*

$$x_j^{k+1} = (1 - m_n^{-1} s_j^{-1} s_{nj}) x_j^k + m_n^{-1} s_j^{-1} x_j^k \sum_{i \in B_n} (A_{ij} \frac{b_i}{(Ax^k)_i}). \tag{13.41}$$

Let's look now at some of the other choices for these parameters that have been considered in the literature.

First, we notice that the OSEM does not generally satisfy the requirements, since in (13.6) the choices are $\alpha_{ni} = 1$ and $\beta_{nj} = s_{nj}^{-1}$; the only times this is acceptable is if the s_{nj} are separable; that is, $s_{nj} = r_j t_n$ for some r_j and t_n. This is slightly more general than the condition of subset balance and is sufficient for convergence of OSEM.

The original EMML in Equation (13.2) uses $N = 1$, $\gamma_j = s_j^{-1}$ and $\alpha_{ni} = \alpha_i = 1$. Clearly the inequality (13.9) is satisfied; in fact it becomes an equality now.

Notice that the calculations required to perform the BI-SMART are somewhat more complicated than those needed in BI-EMML. Because the MART converges rapidly in most cases there is considerable interest in the row-action version of EMML. It was clear from the outset that using the OSEM in a row-action mode does not work. We see from the formula for BI-EMML that the proper row-action version of EMML, which we call the EM-MART, is the following:

Algorithm 13.5 (EM-MART) *Let x^0 be an arbitrary positive vector and $i = k(\mathrm{mod}\, I) + 1$. Then let*

$$x_j^{k+1} = (1 - \delta_i \gamma_j \alpha_{ii} A_{ij}) x_j^k + \delta_i \gamma_j \alpha_{ii} A_{ij} \frac{b_i}{(Ax^k)_i}, \tag{13.42}$$

with

$$\gamma_j \delta_i \alpha_{ii} A_{ij} \leq 1 \tag{13.43}$$

for all i and j.

The optimal choice would seem to be to take $\delta_i \alpha_{ii}$ as large as possible; that is, to select $\delta_i \alpha_{ii} = 1/\max\{\gamma_j A_{ij}, j = 1, ..., J\}$. With this choice the EM-MART is called the *rescaled* EM-MART (REM-MART).

The EM-MART fails to converge in the inconsistent case. What is always observed, but for which no proof exists, is that, for each fixed $i = 1, 2, ..., I$, as $m \to +\infty$, the EM-MART subsequences $\{x^{mI+i}\}$ converge to separate limit vectors, say $x^{\infty,i}$. This *limit cycle* $\text{LC} = \{x^{\infty,i} | i = 1, ..., I\}$ reduces to a single vector whenever there is a nonnegative solution of $b = Ax$. The greater the minimum value of $KL(y, Ax)$ the more distinct from one another the vectors of the limit cycle are. An analogous result is observed for BI-EMML.

We must mention a method that closely resembles the REM-MART, the *row-action maximum likelihood algorithm* (RAMLA), which was discovered independently by Browne and De Pierro [27]. The RAMLA avoids the limit cycle in the inconsistent case by using strong underrelaxation involving a decreasing sequence of relaxation parameters λ_k. The RAMLA is the following:

Algorithm 13.6 (RAMLA) *Let x^0 be an arbitrary positive vector, and $n = k(\text{mod}\, N) + 1$. Let the positive relaxation parameters λ_k be chosen to converge to zero and $\sum_{k=0}^{+\infty} \lambda_k = +\infty$. Then,*

$$x_j^{k+1} = (1 - \lambda_k \sum_{i \in B_n} A_{ij}) x_j^k + \lambda_k x_j^k \sum_{i \in B_n} A_{ij} \Big(\frac{b_i}{(Ax^k)_i} \Big), \tag{13.44}$$

13.6 RBI-SMART and Entropy Maximization

As we stated earlier, in the consistent case the sequence $\{x^k\}$ generated by the BI-SMART algorithm and given by Equation (13.12) converges to the unique solution of $b = Ax$ for which the distance $\sum_{j=1}^{J} \gamma_j^{-1} KL(x_j, x_j^0)$ is minimized. In this section we sketch the proof of this result as a sequence of lemmas, each of which is easily established.

Lemma 13.4 *For any nonnegative vectors a and b with $a_+ = \sum_{m=1}^{M} a_m$ and $b_+ = \sum_{m=1}^{M} b_m > 0$ we have*

$$KL(a, b) = KL(a_+, b_+) + KL(a_+, \frac{a_+}{b_+} b). \tag{13.45}$$

For nonnegative vectors x and z let

$$G_n(x,z) = \sum_{j=1}^{J} \gamma_j^{-1} KL(x_j, z_j) \tag{13.46}$$

$$+\delta_n \sum_{i \in B_n} \alpha_{ni} [KL((Ax)_i, b_i) - KL((Ax)_i, (Az)_i)]. \tag{13.47}$$

It follows from Lemma 13.45 and the inequality

$$\gamma_j^{-1} - \delta_n \sigma_{nj} \geq 1 \tag{13.48}$$

that $G_n(x,z) \geq 0$ in all cases.

Lemma 13.5 *For every x we have*

$$G_n(x,x) = \delta_n \sum_{i \in B_n} \alpha_{ni} KL((Ax)_i, b_i) \tag{13.49}$$

so that

$$G_n(x,z) = G_n(x,x) + \sum_{j=1}^{J} \gamma_j^{-1} KL(x_j, z_j) \tag{13.50}$$

$$-\delta_n \sum_{i \in B_n} \alpha_{ni} KL((Ax)_i, (Az)_i). \tag{13.51}$$

Therefore the distance $G_n(x,z)$ is minimized, as a function of z, by $z = x$. Now we minimize $G_n(x,z)$ as a function of x. The following lemma shows that the answer is

$$x_j = z_j' = z_j \exp\Big(\gamma_j \delta_n \sum_{i \in B_n} \alpha_{ni} A_{ij} \log \frac{b_i}{(Az)_i}\Big). \tag{13.52}$$

Lemma 13.6 *For each x and z we have*

$$G_n(x,z) = G_n(z',z) + \sum_{j=1}^{J} \gamma_j^{-1} KL(x_j, z_j'). \tag{13.53}$$

It is clear that $(x^k)' = x^{k+1}$ for all k.

Now let $b = Pu$ for some nonnegative vector u. We calculate $G_n(u, x^k)$ in two ways: using the definition we have

$$G_n(u,x^k) = \sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^k) - \delta_n \sum_{i \in B_n} \alpha_{ni} KL(b_i, (Ax^k)_i), \tag{13.54}$$

while using Lemma 13.53 we find that

$$G_n(u, x^k) = G_n(x^{k+1}, x^k) + \sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^{k+1}). \tag{13.55}$$

Therefore

$$\sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^k) - \sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^{k+1}) \tag{13.56}$$

$$= G_n(x^{k+1}, x^k) + \delta_n \sum_{i \in B_n} \alpha_{ni} KL(b_i, (Ax^k)_i). \tag{13.57}$$

We conclude several things from this.

First, the sequence $\{\sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^k)\}$ is decreasing, so that the sequences $\{G_n(x^{k+1}, x^k)\}$ and $\{\delta_n \sum_{i \in B_n} \alpha_{ni} KL(b_i, (Ax^k)_i)\}$ converge to zero. Therefore the sequence $\{x^k\}$ is bounded and we may select an arbitrary cluster point x^*. It follows that $b = Ax^*$. We may therefore replace the generic solution u with x^* to find that $\{\sum_{j=1}^{J} \gamma_j^{-1} KL(x_j^*, x_j^k)\}$ is a decreasing sequence; but since a subsequence converges to zero, the entire sequence must converge to zero. Therefore $\{x^k\}$ converges to the solution x^*.

Finally, since the right side of Equation (13.57) does not depend on the particular choice of solution we made, neither does the left side. By *telescoping* we conclude that

$$\sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^0) - \sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^*) \tag{13.58}$$

is also independent of the choice of u. Consequently, minimizing the function $\sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^0)$ over all solutions u is equivalent to minimizing $\sum_{j=1}^{J} \gamma_j^{-1} KL(u_j, x_j^*)$ over all solutions u; but the solution to the latter problem is obviously $u = x^*$. This completes the proof.

Chapter 14

Regularization

When we use an iterative algorithm, we want it to solve our problem. We also want the solution in a reasonable amount of time, and we want slight errors in the measurements to cause only slight perturbations in the calculated answer. We have already discussed the use of block-iterative methods to accelerate convergence. Now we turn to regularization as a means of reducing sensitivity to noise. Because a number of regularization methods can be derived using a Bayesian *maximum a posteriori* approach, regularization is sometimes treated under the heading of MAP methods; see, for example, [167, 184] and the discussion in [57]. Penalty functions are also used for regularization [105, 2, 3].

14.1 Where Does Sensitivity Come From?

We illustrate the sensitivity problem that can arise when the inconsistent system $Ax = b$ has more equations than unknowns. We take A to be I by J and we calculate the least-squares solution,

$$x_{LS} = (A^\dagger A)^{-1} A^\dagger b, \tag{14.1}$$

assuming that the J by J Hermitian, nonnegative-definite matrix $Q = (A^\dagger A)$ is invertible, and therefore positive-definite.

The matrix Q has the eigenvalue/eigenvector decomposition

$$Q = \lambda_1 u_1 u_1^\dagger + \cdots + \lambda_J u_J u_J^\dagger, \tag{14.2}$$

where the (necessarily positive) eigenvalues of Q are

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_J > 0, \tag{14.3}$$

and the vectors u_j are the corresponding orthonormal eigenvectors.

14.1.1 The Singular-Value Decomposition of A

The square roots $\sqrt{\lambda_j}$ are called the *singular values* of A. The *singular-value decomposition* (SVD) of A is similar to the eigenvalue/eigenvector decomposition of Q: we have

$$A = \sqrt{\lambda_1} u_1 v_1^\dagger + \cdots + \sqrt{\lambda_I} u_J v_J^\dagger, \tag{14.4}$$

where the v_j are particular eigenvectors of $AA^\dagger$. We see from the SVD that the quantities $\sqrt{\lambda_j}$ determine the relative importance of each term $u_j v_j^\dagger$.

The SVD is commonly used for compressing transmitted or stored images. In such cases, the rectangular matrix A is a discretized image. It is not uncommon for many of the lowest singular values of A to be nearly zero, and to be essentially insignificant in the reconstruction of A. Only those terms in the SVD for which the singular values are significant need to be transmitted or stored. The resulting images may be slightly blurred, but can be restored later, as needed.

When the matrix A is a finite model of a linear imaging system, there will necessarily be model error in the selection of A. Getting the dominant terms in the SVD nearly correct is much more important (and usually much easier) than getting the smaller ones correct. The problems arise when we try to invert the system, to solve $Ax = b$ for x.

14.1.2 The Inverse of $Q = A^\dagger A$

The inverse of Q can then be written

$$Q^{-1} = \lambda_1^{-1} u_1 u_1^\dagger + \cdots + \lambda_J^{-1} u_J u_J^\dagger, \tag{14.5}$$

so that, with $A^\dagger b = c$, we have

$$x_{LS} = \lambda_1^{-1} (u_1^\dagger c) u_1 + \cdots + \lambda_J^{-1} (u_J^\dagger c) u_J. \tag{14.6}$$

Because the eigenvectors are orthonormal, we can express $||A^\dagger b||_2^2 = ||c||_2^2$ as

$$||c||_2^2 = |u_1^\dagger c|^2 + \cdots + |u_J^\dagger c|^2, \tag{14.7}$$

and $||x_{LS}||_2^2$ as

$$||x_{LS}||_2^2 = \lambda_1^{-1} |u_1^\dagger c|^2 + \cdots + \lambda_J^{-1} |u_J^\dagger c|^2. \tag{14.8}$$

It is not uncommon for the eigenvalues of Q to be quite distinct, with some of them much larger than the others. When this is the case, we see that $||x_{LS}||_2$ can be much larger than $||c||_2$, because of the presence of the terms involving the reciprocals of the small eigenvalues. When the measurements

b are essentially noise-free, we may have $|u_j^\dagger c|$ relatively small, for the indices near J, keeping the product $\lambda_j^{-1}|u_j^\dagger c|^2$ reasonable in size, but when the b becomes noisy, this may no longer be the case. The result is that those terms corresponding to the reciprocals of the smallest eigenvalues dominate the sum for x_{LS} and the norm of x_{LS} becomes quite large. The least-squares solution we have computed is essentially all noise and useless.

In our discussion of the ART, we saw that when we impose a non-negativity constraint on the solution, noise in the data can manifest itself in a different way. When A has more columns than rows, but $Ax = b$ has no non-negative solution, then, at least for those A having the *full-rank property*, the non-negatively constrained least-squares solution has at most $I - 1$ non-zero entries. This happens also with the EMML and SMART solutions. As with the ART, regularization can eliminate the problem.

14.1.3 Reducing the Sensitivity to Noise

As we just saw, the presence of small eigenvalues for Q and noise in b can cause $||x_{LS}||_2$ to be much larger than $||A^\dagger b||_2$, with the result that x_{LS} is useless. In this case, even though x_{LS} minimizes $||Ax - b||_2$, it does so by overfitting to the noisy b. To reduce the sensitivity to noise and thereby obtain a more useful approximate solution, we can *regularize* the problem.

It often happens in applications that, even when there is an exact solution of $Ax = b$, noise in the vector b makes such as exact solution undesirable; in such cases a *regularized solution* is usually used instead. Select $\epsilon > 0$ and a vector p that is a prior estimate of the desired solution. Define

$$F_\epsilon(x) = (1 - \epsilon)\|Ax - b\|_2^2 + \epsilon\|x - p\|_2^2. \tag{14.9}$$

Lemma 14.1 *The function F_ϵ always has a unique minimizer $\hat{x}_\epsilon$, given by*

$$\hat{x}_\epsilon = ((1 - \epsilon)A^\dagger A + \epsilon I)^{-1}((1 - \epsilon)A^\dagger b + \epsilon p); \tag{14.10}$$

this is a regularized solution of $Ax = b$. Here, p is a prior estimate of the desired solution. Note that the inverse above always exists.

Note that, if $p = 0$, then

$$\hat{x}_\epsilon = (A^\dagger A + \gamma^2 I)^{-1} A^\dagger b, \tag{14.11}$$

for $\gamma^2 = \frac{\epsilon}{1-\epsilon}$. The regularized solution has been obtained by modifying the formula for x_{LS}, replacing the inverse of the matrix $Q = A^\dagger A$ with the inverse of $Q + \gamma^2 I$. When ϵ is near zero, so is γ^2, and the matrices

Q and $Q + \gamma^2 I$ are nearly equal. What is different is that the eigenvalues of $Q + \gamma^2 I$ are $\lambda_i + \gamma^2$, so that, when the eigenvalues are inverted, the reciprocal eigenvalues are no larger than $1/\gamma^2$, which prevents the norm of x_ϵ from being too large, and decreases the sensitivity to noise.

Lemma 14.2 *Let ϵ be in $(0, 1)$, and let I be the identity matrix whose dimensions are understood from the context. Then*

$$((1-\epsilon)AA^\dagger + \epsilon I)^{-1}A = A((1-\epsilon)A^\dagger A + \epsilon I)^{-1}, \tag{14.12}$$

and, taking conjugate transposes,

$$A^\dagger((1-\epsilon)AA^\dagger + \epsilon I)^{-1} = ((1-\epsilon)A^\dagger A + \epsilon I)^{-1}A^\dagger. \tag{14.13}$$

Proof: Use the identity

$$A((1-\epsilon)A^\dagger A + \epsilon I) = ((1-\epsilon)AA^\dagger + \epsilon I)A. \tag{14.14}$$

∎

Lemma 14.3 *Any vector p in R^J can be written as $p = A^\dagger q + r$, where $Ar = 0$.*

What happens to $\hat{x}_\epsilon$ as ϵ goes to zero? This will depend on which case we are in:

Case 1: $J \leq I$, and we assume that $A^\dagger A$ is invertible; or

Case 2: $J > I$, and we assume that $AA^\dagger$ is invertible.

Lemma 14.4 *In Case 1, taking limits as $\epsilon \to 0$ on both sides of the expression for $\hat{x}_\epsilon$ gives $\hat{x}_\epsilon \to (A^\dagger A)^{-1}A^\dagger b$, the least squares solution of $Ax = b$.*

We consider Case 2 now. Write $p = A^\dagger q + r$, with $Ar = 0$. Then

$$\hat{x}_\epsilon = A^\dagger((1-\epsilon)AA^\dagger + \epsilon I)^{-1}((1-\epsilon)b + \epsilon q) + ((1-\epsilon)A^\dagger A + \epsilon I)^{-1}(\epsilon r). \tag{14.15}$$

Lemma 14.5 (a) *We have*

$$((1-\epsilon)A^\dagger A + \epsilon I)^{-1}(\epsilon r) = r, \tag{14.16}$$

for all $\epsilon \in (0, 1)$. **(b)** *Taking the limit of $\hat{x}_\epsilon$, as $\epsilon \to 0$, we get $\hat{x}_\epsilon \to A^\dagger(AA^\dagger)^{-1}b + r$. This is the solution of $Ax = b$ closest to p.*

Proof: For part (a) let

$$t_\epsilon = ((1-\epsilon)A^\dagger A + \epsilon I)^{-1}(\epsilon r). \tag{14.17}$$

Then, multiplying by A gives

$$At_\epsilon = A((1-\epsilon)A^\dagger A + \epsilon I)^{-1}(\epsilon r). \tag{14.18}$$

Now show that $At_\epsilon = 0$. For part (b) draw a diagram for the case of one equation in two unknowns. ∎

14.2 Iterative Regularization

It is often the case that the entries of the vector b in the system $Ax = b$ come from measurements, so are usually noisy. If the entries of b are noisy but the system $Ax = b$ remains consistent (which can easily happen in the under-determined case, with $J > I$), the ART begun at $x^0 = 0$ converges to the solution having minimum norm, but this norm can be quite large. The resulting solution is probably useless. Instead of solving $Ax = b$, we *regularize* by minimizing, for example, the function $F_\epsilon(x)$ given in Equation (14.9). For the case of $p = 0$, the solution to this problem is the vector $\hat{x}_\epsilon$ in Equation (14.11). However, we do not want to calculate $A^\dagger A + \gamma^2 I$, in order to solve

$$(A^\dagger A + \gamma^2 I)x = A^\dagger b, \tag{14.19}$$

when the matrix A is large. Fortunately, there are ways to find $\hat{x}_\epsilon$, using only the matrix A. We saw previously how this might be accomplished using the ART; now we show how the Landweber algorithm can be used to calculate this regularized solution.

14.2.1 Iterative Regularization with Landweber's Algorithm

Our goal is to minimize the function in Equation (14.9), with $p = 0$. Notice that this is equivalent to minimizing the function

$$F(x) = ||Bx - c||_2^2, \tag{14.20}$$

for

$$B = \begin{bmatrix} A \\ \gamma I \end{bmatrix}, \tag{14.21}$$

and

$$c = \begin{bmatrix} b \\ 0 \end{bmatrix}, \tag{14.22}$$

where 0 denotes a column vector with all entries equal to zero nd $\gamma = \frac{\epsilon}{1-\epsilon}$.. The Landweber iteration for the problem $Bx = c$ is

$$x^{k+1} = x^k + \alpha B^T(c - Bx^k), \tag{14.23}$$

for $0 < \alpha < 2/\rho(B^T B)$, where $\rho(B^T B)$ is the spectral radius of $B^T B$. Equation (14.23) can be written as

$$x^{k+1} = (1 - \alpha\gamma^2)x^k + \alpha A^T(b - Ax^k). \tag{14.24}$$

We see from Equation (14.24) that the Landweber algorithm for solving the regularized least squares problem amounts to a relaxed version of the Landweber algorithm applied to the original least squares problem.

14.3 A Bayesian View of Reconstruction

The EMML iterative algorithm maximizes the likelihood function for the case in which the entries of the data vector $b = (b_1, ..., b_I)^T$ are assumed to be samples of independent Poisson random variables with mean values $(Ax)_i$; here, A is an I by J matrix with nonnegative entries and $x = (x_1, ..., x_J)^T$ is the vector of nonnegative parameters to be estimated. Equivalently, it minimizes the Kullback-Leibler distance $KL(b, Ax)$. This situation arises in single photon emission tomography, where the b_i are the number of photons counted at each detector i, x is the vectorized image to be reconstructed and its entries x_j are (proportional to) the radionuclide intensity levels at each voxel j. When the signal-to-noise ratio is low, which is almost always the case in medical applications, maximizing likelihood can lead to unacceptably noisy reconstructions, particularly when J is larger than I. One way to remedy this problem is simply to halt the EMML algorithm after a few iterations, to avoid over-fitting the x to the noisy data. A more mathematically sophisticated remedy is to employ a penalized-likelihood or Bayesian approach and seek a maximum *a posteriori* (MAP) estimate of x.

In the Bayesian approach we view x as an instance of a random vector having a probability density function $f(x)$. Instead of maximizing the likelihood given the data, we now maximize the posterior likelihood, given both the data and the prior distribution for x. This is equivalent to minimizing

$$F(x) = KL(b, Ax) - \log f(x). \tag{14.25}$$

The EMML algorithm is an example of an optimization method based on alternating minimization of a function $H(x, z) > 0$ of two vector variables. The alternating minimization works this way: let x and z be vector variables and $H(x, z) > 0$. If we fix z and minimize $H(x, z)$ with respect to x, we find that the solution is $x = z$, the vector we fixed; that is,

$$H(x, z) \geq H(z, z) \tag{14.26}$$

always. If we fix x and minimize $H(x, z)$ with respect to z, we get something new; call it Tx. The EMML algorithm has the iterative step $x^{k+1} = Tx^k$.

Obviously, we can't use an arbitrary function H; it must be related to the function $KL(b, Ax)$ that we wish to minimize, and we must be able to obtain each intermediate optimizer in closed form. The clever step is to select $H(x, z)$ so that $H(x, x) = KL(b, Ax)$, for any x. Now see what we have so far:

$$KL(b, Ax^k) = H(x^k, x^k) \geq H(x^k, x^{k+1}) \tag{14.27}$$

$$\geq H(x^{k+1}, x^{k+1}) = KL(b, Ax^{k+1}). \tag{14.28}$$

That tells us that the algorithm makes $KL(b, Ax^k)$ decrease with each iteration. The proof doesn't stop here, but at least it is now plausible that the EMML iteration could minimize $KL(b, Ax)$.

The function $H(x, z)$ used in the EMML case is the KL distance

$$H(x, z) = KL(r(x), q(z)) = \sum_{i=1}^{I} \sum_{j=i}^{J} KL(r(x)_{ij}, q(z)_{ij}); \tag{14.29}$$

we define, for each nonnegative vector x for which $(Ax)_i = \sum_{j=1}^{J} A_{ij}x_j > 0$, the arrays $r(x) = \{r(x)_{ij}\}$ and $q(x) = \{q(x)_{ij}\}$ with entries

$$r(x)_{ij} = x_j A_{ij} \frac{b_i}{(Ax)_i} \tag{14.30}$$

and

$$q(x)_{ij} = x_j A_{ij}. \tag{14.31}$$

With $x = x^k$ fixed, we minimize with respect to z to obtain the next EMML iterate x^{k+1}. Having selected the prior pdf $f(x)$, we want an iterative algorithm to minimize the function $F(x)$ in Equation (14.25). It would be a great help if we could mimic the alternating minimization formulation and obtain x^{k+1} by minimizing

$$KL(r(x^k), q(z)) - \log f(z) \tag{14.32}$$

with respect to z. Unfortunately, to be able to express each new x^{k+1} in closed form, we need to choose $f(x)$ carefully.

14.4 The Gamma Prior Distribution for x

In [151] Lange *et al.* suggest viewing the entries x_j as samples of independent gamma-distributed random variables. A gamma-distributed random variable x takes positive values and has for its pdf the *gamma distribution* defined for positive x by

$$\gamma(x) = \frac{1}{\Gamma(\alpha)} (\frac{\alpha}{\beta})^{\alpha} x^{\alpha-1} e^{-\alpha x/\beta}, \tag{14.33}$$

where α and β are positive parameters and Γ denotes the gamma function. The mean of such a gamma-distributed random variable is then $\mu = \beta$ and the variance is $\sigma^2 = \beta^2/\alpha$.

Lemma 14.6 *If the entries z_j of z are viewed as independent and gamma-distributed with means μ_j and variances σ_j^2, then minimizing the function in line (14.32) with respect to z is equivalent to minimizing the function*

$$KL(r(x^k), q(z)) + \sum_{j=1}^{J} \delta_j KL(\gamma_j, z_j), \tag{14.34}$$

for

$$\delta_j = \frac{\mu_j}{\sigma_j^2},\ \gamma_j = \frac{\mu_j^2 - \sigma_j^2}{\mu_j}, \tag{14.35}$$

under the assumption that the latter term is positive.

The resulting regularized EMML algorithm is the following:

Algorithm 14.1 (γ-prior Regularized EMML) *Let x^0 be an arbitrary positive vector. Then let*

$$x_j^{k+1} = \frac{\delta_j}{\delta_j + s_j}\gamma_j + \frac{1}{\delta_j + s_j} x_j^k \sum_{i=1}^{I} A_{ij} b_i / (Ax^k)_i, \tag{14.36}$$

where $s_j = \sum_{i=1}^{I} A_{ij}$.

We see from Equation (14.36) that the MAP iteration using the gamma priors generates a sequence of estimates each entry of which is a convex combination or weighted arithmetic mean of the result of one EMML step and the prior estimate γ_j. Convergence of the resulting iterative sequence is established by Lange, Bahn and Little in [151]; see also [43].

14.5 The One-Step-Late Alternative

It may well happen that we do not wish to use the gamma priors model and prefer some other $f(x)$. Because we will not be able to find a closed form expression for the z minimizing the function in line (14.32), we need some other way to proceed with the alternating minimization. Green [121] has offered the *one-step-late* (OSL) alternative.

When we try to minimize the function in line (14.32) by setting the gradient to zero we replace the variable z that occurs in the gradient of the term $-\log f(z)$ with x^k, the previously calculated iterate. Then, we can solve for z in closed form to obtain the new x^{k+1}. Unfortunately, negative entries can result and convergence is not guaranteed. There is a sizable literature on the use of MAP methods for this problem. In [52] an interior point algorithm (IPA) is presented that avoids the OSL issue. In [168] the IPA is used to regularize transmission tomographic images.

14.6 Regularizing the SMART

The SMART algorithm is not derived as a maximum likelihood method, so regularized versions do not take the form of MAP algorithms. Nevertheless, in the presence of noisy data, the SMART algorithm suffers from the same problem that afflicts the EMML, overfitting to noisy data resulting in an unacceptably noisy image. As we saw earlier, there is a close connection between the EMML and SMART algorithms. This suggests that a regularization method for SMART can be developed along the lines of the MAP with gamma priors used for EMML. Since the SMART is obtained by minimizing the function $KL(q(z), r(x^k))$ with respect to z to obtain x^{k+1}, it seems reasonable to attempt to derive a regularized SMART iterative scheme by minimizing

$$KL(q(z), r(x^k)) + \sum_{j=1}^{J} \delta_j KL(z_j, \gamma_j), \tag{14.37}$$

as a function of z, for selected positive parameters δ_j and γ_j. This leads to the following algorithm:

Algorithm 14.2 (Regularized SMART) *Let x^0 be an arbitrary positive vector. Then let*

$$\log x_j^{k+1} = \frac{\delta_j}{\delta_j + s_j} \log \gamma_j + \frac{1}{\delta_j + s_j} x_j^k \sum_{i=1}^{I} A_{ij} \log[b_i/(Ax^k)_i]. \tag{14.38}$$

In [43] it was shown that this iterative sequence converges to a minimizer of the function

$$KL(Ax, y) + \sum_{j=1}^{J} \delta_j KL(x_j, \gamma_j). \tag{14.39}$$

It is useful to note that, although it may be possible to rederive this minimization problem within the framework of Bayesian MAP estimation by carefully selecting a prior pdf for the vector x, we have not done so. The MAP approach is a special case of regularization through the use of penalty functions. These penalty functions need not arise through a Bayesian formulation of the parameter-estimation problem.

14.7 De Pierro's Surrogate-Function Method

In [90] De Pierro presents a modified EMML algorithm that includes regularization in the form of a penalty function. His objective is the same as ours was in the case of regularized SMART: to embed the penalty term

in the alternating minimization framework in such a way as to make it possible to obtain the next iterate in closed form. Because his *surrogate function* method has been used subsequently by others to obtain penalized likelihood algorithms [73], we consider his approach in some detail.

Let x and z be vector variables and $H(x,z) > 0$. Mimicking the behavior of the function $H(x,z)$ used in Equation (14.29), we require that if we fix z and minimize $H(x,z)$ with respect to x, the solution should be $x = z$, the vector we fixed; that is, $H(x,z) \geq H(z,z)$ always. If we fix x and minimize $H(x,z)$ with respect to z, we should get something new; call it Tx. As with the EMML, the algorithm will have the iterative step $x^{k+1} = Tx^k$.

Summarizing, we see that we need a function $H(x,z)$ with the properties (1) $H(x,z) \geq H(z,z)$ for all x and z; (2) $H(x,x)$ is the function $F(x)$ we wish to minimize; and (3) minimizing $H(x,z)$ with respect to z for fixed x is easy.

The function to be minimized is

$$F(x) = KL(b, Ax) + g(x), \tag{14.40}$$

where $g(x) \geq 0$ is some penalty function. De Pierro uses penalty functions $g(x)$ of the form

$$g(x) = \sum_{l=1}^{p} f_l(\langle s_l, x\rangle). \tag{14.41}$$

Let us define the matrix S to have for its lth row the vector s_l^T. Then $\langle s_l, x\rangle = (Sx)_l$, the lth entry of the vector Sx. Therefore,

$$g(x) = \sum_{l=1}^{p} f_l((Sx)_l). \tag{14.42}$$

Let $\lambda_{lj} > 0$ with $\sum_{j=1}^{J} \lambda_{lj} = 1$, for each l.

Assume that the functions f_l are convex. Therefore, for each l, we have

$$f_l((Sx)_l) = f_l(\sum_{j=1}^{J} S_{lj}x_j) = f_l(\sum_{j=1}^{J} \lambda_{lj}(S_{lj}/\lambda_{lj})x_j) \tag{14.43}$$

$$\leq \sum_{j=1}^{J} \lambda_{lj} f_l((S_{lj}/\lambda_{lj})x_j). \tag{14.44}$$

Therefore,

$$g(x) \leq \sum_{l=1}^{p}\sum_{j=1}^{J} \lambda_{lj} f_l((S_{lj}/\lambda_{lj})x_j). \tag{14.45}$$

So we have replaced $g(x)$ with a related function in which the x_j occur separately, rather than just in the combinations $(Sx)_l$. But we aren't quite done yet.

We would like to take for De Pierro's $H(x,z)$ the function used in the EMML algorithm, plus the function

$$\sum_{l=1}^{p}\sum_{j=1}^{J}\lambda_{lj}f_l((S_{lj}/\lambda_{lj})z_j). \tag{14.46}$$

But there is one slight problem: we need $H(z,z) = F(z)$, which we don't have yet.

De Pierro's clever trick is to replace $f_l((S_{lj}/\lambda_{lj})z_j)$ with

$$f_l\Big((S_{lj}/\lambda_{lj})z_j - (S_{lj}/\lambda_{lj})x_j\Big) + f_l((Sx)_l). \tag{14.47}$$

So, De Pierro's function $H(x,z)$ is the sum of the $H(x,z)$ used in the EMML case and the function

$$\sum_{l=1}^{p}\sum_{j=1}^{J}\lambda_{lj}f_l\Big((S_{lj}/\lambda_{lj})z_j - (S_{lj}/\lambda_{lj})x_j\Big) + \sum_{l=1}^{p}f_l((Sx)_l). \tag{14.48}$$

Now he has the three properties he needs. Once he has computed x^k, he minimizes $H(x^k,z)$ by taking the gradient and solving the equations for the correct $z = Tx^k = x^{k+1}$. For the choices of f_l he discusses, these intermediate calculations can either be done in closed form (the quadratic case) or with a simple Newton-Raphson iteration (the logcosh case).

14.8 Block-Iterative Regularization

We saw previously that it is possible to obtain a regularized least-squares solution $\hat{x}_\epsilon$, and thereby avoid the limit cycle, using only the matrix A and the ART algorithm. This prompts us to ask if it is possible to find regularized SMART solutions using block-iterative variants of SMART. Similarly, we wonder if it is possible to do the same for EMML.

Open Question: Can we use the MART to find the minimizer of the function

$$KL(Ax,b) + \epsilon KL(x,p)? \tag{14.49}$$

More generally, can we obtain the minimizer using RBI-SMART?

Open Question: Can we use the RBI-EMML methods to obtain the minimizer of the function

$$KL(b,Ax) + \epsilon KL(p,x)? \tag{14.50}$$

There have been various attempts to include regularization in block-iterative methods, to reduce noise sensitivity and avoid limit cycles; the paper by Ahn and Fessler [2] is a good source, as is [3]. Most of these approaches have been *ad hoc*, with little or no theoretical basis. Typically, they simply modify each iterative step by including an additional term that appears to be related to the regularizing penalty function. The case of the ART is instructive, however. In that case, we obtained the desired iterative algorithm by using an augmented set of variables, not simply by modifying each step of the original ART algorithm. How to do this for the MART and the other block-iterative algorithms is not obvious.

Recall that the RAMLA method in Equation (13.44) is similar to the RBI-EMML algorithm, but employs a sequence of decreasing relaxation parameters, which, if properly chosen, will cause the iterates to converge to the minimizer of $KL(b, Ax)$, thereby avoiding the limit cycle. In [92] De Pierro and Yamaguchi present a regularized version of RAMLA, but without guaranteed convergence.

Chapter 15

List-Mode Reconstruction in PET

15.1 Why List-Mode Processing?

In PET the radionuclide emits individual positrons, which travel, on average, between 4 mm and 2.5 cm (depending on their kinetic energy) before encountering an electron. The resulting annihilation releases two gamma-ray photons that then proceed in essentially opposite directions. Detection in the PET case means the recording of two photons at nearly the same time at two different detectors. The locations of these two detectors then provide the end points of the line segment passing, more or less, through the site of the original positron emission. Therefore, each possible pair of detectors determines a line of response. When a LOR is recorded, it is assumed that a positron was emitted somewhere along that line.

In modern PET scanners the number of pairs of detectors, and therefore, the number of potential LOR, often exceeds the number of detections; the count recorded at any single i is typically one or zero. It makes sense, therefore, to record the data as a list of those LOR corresponding to a detection; this is list-mode data.

15.2 Correcting for Attenuation in PET

In SPECT attenuation correction is performed by modifying the probabilities P_{ij}. In PET the situation is at once simpler and more involved.

Let a given LOR be parameterized by the variable s, with $s = 0$ and $s = c$ denoting the two ends, and c the distance from one end to the other. For a fixed value $s = s_0$, let $P(s)$ be the probability of reaching s for a photon resulting from an emission at s_0. For small $\Delta s > 0$ the probability

that a photon that reached s is absorbed in the interval $[s, s + \Delta s]$ is approximately $\mu(s)\Delta s$, where $\mu(s) \geq 0$ is the photon attenuation density at s. Then $P(s + \Delta s) \approx P(s)[1 - \mu(s)\Delta s]$, so that

$$P(s + \Delta s) - P(s) \approx -P(s)\mu(s)\Delta s.$$

Dividing by Δs and letting Δs go to zero, we get

$$P'(s) = -P(s)\mu(s).$$

It follows that

$$P(s) = e^{-\int_{s_0}^{s} \mu(t)dt}.$$

The probability that the photon will reach $s = c$ and be detected is then

$$P(c) = e^{-\int_{s_0}^{c} \mu(t)dt}.$$

Similarly, we find that the probability that a photon will succeed in reaching $s = 0$ from s_0 is

$$P(0) = e^{-\int_{0}^{s_0} \mu(t)dt}.$$

Since having one photon reach $s = 0$ and the other reach $s = c$ are independent events, their probabilities multiply, so that the probability that both photons reach their destinations and a coincident detection is recorded for this LOR is

$$e^{-\int_{0}^{c} \mu(t)dt}.$$

The expected number of coincident detections along the LOR is then proportional to

$$\int_0^c f(s)e^{-\int_0^c \mu(t)dt}ds = e^{-\int_0^c \mu(t)dt} \int_0^c f(s)ds, \tag{15.1}$$

where $f(s)$ is the intensity of radionuclide at s.

For each LOR i and each pixel or voxel j, let A_{ij} be the *geometric probability* that an emission at j will result in two photons traveling along the LOR i. The probability A_{ij} is unrelated to the attenuation presented by the body of the patient. Then the probability that an emission at j will result in the LOR i being added to the list is

$$P_{ij} = a_i A_{ij},$$

where

$$a_i = e^{-\int_i \mu(s)ds},$$

and the integral is the line integral along the line segment associated with the LOR i. We then perform attenuation correction by using the probabilities P_{ij} in the reconstruction.

Note that, if the number I of potential LOR is not too large and the entries of the data vector y are not simply zero or one, we might correct for attenuation by replacing each y_i with y_i/a_i, which is approximately the count we would have seen for the LOR i if there had been no attenuation. However, in the more typical case of large I and zero or one values for the y_i, this approach does not make much sense. The effect of attenuation now is to prevent certain i from being recorded, not to diminish the values of the positive y_i of the LOR that were recorded. Therefore, at least in theory, it makes more sense to correct for attenuation by using the P_{ij}. There is an additional complication, though.

In list-mode processing, I, the number of potential LOR, is much larger than the size of the list. To employ the EMML algorithm or one of its block-iterative variants, we need to calculate the probabilities associated with those LOR on the list, but it is costly to do this for all the potential LOR; we do need to compute the sensitivities, or probabilities of detection, for each pixel, however. If we consider only the geometry of the scanner, calculating the sensitivities for each pixel is not difficult and can be done once and used repeatedly; it is much more problematic if we must include the patient-specific attenuation. For this reason, it makes sense, practically speaking, to correct for attenuation in list-mode PET by replacing y_i with y_i/a_i for those y_i equal to one. The reconstruction is probably much the same, either way.

15.3 Modeling the Possible LOR

We can model the potential LOR simply as pairs of detectors, so that I, the number of potential LOR, is very large, but finite, and finite probability vectors, rather than probability density functions, suffice in forming the likelihood function. The EMML algorithm applies directly to this list-mode model. This is the approach adopted by Huesman *et al.* [**?**].

Alternatively, one can assume that the end-point coordinates form a continuum, so that the set of potential LOR is uncountably infinite. Now we need probability density functions to form the likelihood function. This method, adopted by Parra and Barrett [176], makes the application of the EMML algorithm more complicated, as discussed in [53].

15.4 EMML: The Finite LOR Model

In this section we discuss the EMML iterative algorithm for list-mode reconstruction based on the finite model.

Let the list of recorded LOR be $\{i_1, ..., i_M\}$ and let

$$Q_{mj} = P_{i_m,j},$$

for $m = 1, ..., M$. Since the values of the y_i are typically zero or one, the i_m are typically distinct, but this is not essential here. The EMML iteration becomes

$$x_j^{k+1} = x_j^k s_j^{-1} \sum_{m=1}^{M} Q_{mj} \Big(\frac{1}{(Qx^k)_m} \Big). \tag{15.2}$$

Note that we still need to use the sensitivity values

$$s_j = \sum_{i=1}^{I} P_{ij},$$

which are the probabilities of detection. However, for imaging the radionuclide we do not need to calculate the s_j by first determining each of the P_{ij}; we need only that the $s_j > \sum_{m=1}^{M} Q_{mj}$ for each j and that the relative values of the various s_j be reasonably accurate. For quantitation, though, accurate absolute values of the s_j are needed.

15.5 List-mode RBI-EMML

We turn now to the block-iterative versions of EMML. For $n = 1, ..., N$ let C_n consist of all indices m such that the LOR i_m on the list is also in B_n. The list-mode BI-EMML (LMBI-EMML) has the iterative step

$$x_j^k = (1 - \gamma_n \delta_j s_{nj}) x_j^{k-1} + x_j^k \gamma_n \delta_j \sum_{m \in C_n} P_{ij} \Big(\frac{1}{(Qx^k)_m} \Big), \tag{15.3}$$

with $\gamma > 0$ chosen so that

$$s_{nj} \delta_j \gamma_n \leq 1.$$

When we select $\delta_j = s_j^{-1}$, we must then have $\gamma_n \leq \mu_n^{-1}$. When we have $\delta_j = 1$, we need $\gamma_n \leq m_n^{-1}$. Generally speaking, the larger the γ_n the faster the convergence. The *rescaled* LMBI-EMML (LMRBI-EMML) uses the largest values of γ_n consistent with these constraints.

Note that, as previously, we need s_j and now we also need s_{nj}. As before, though, we do not need to specify each of the P_{ij} to obtain reasonable choices for these values.

15.6 The Row-action LMRBI-EMML: LMEMART

The row-action or *event-by-event* version of the RBI-EMML algorithm, the LMEMART, is a special case of the LMRBI-EMML in which, for $m = 1, ..., M$, each LOR i_m on the list forms its own block or subset, denoted

C_m. Another way to say this is that we choose the original blocks B_n so that no B_n contains more than one i_m. For clarity, we shall assume that the blocks B_n are chosen so that $B_m = \{i_m\}$ and $C_m = \{m\}$, for $m = 1, ..., M$. We then let B_{M+1} consist of all the i not equal to some I_m on the list, and $N = M + 1$. Therefore, for $n = 1, ..., M$, we have

$$s_{nj} = Q_{nj}.$$

In the LMEMART each iteration employs a single member of the list and we cycle through the list repeatedly. The iteration index is now $m = 1, ..., M$, with $m = m(k) = k(\mathrm{mod}\, M) + 1$.

The LMEMART has the iterative step

$$x_j^{k+1} = (1 - \gamma_m \delta_j Q_{mj}) x_j^k + x_j^k \gamma_m \delta_j Q_{mj} \Big(\frac{1}{(Qx^k)_m} \Big), \tag{15.4}$$

with $Q_{mj}\delta_j\gamma_m \leq 1$.

15.7 EMML: The Continuous LOR Model

When the end points of the potential LOR are allowed to take on values in a continuum, the likelihood function involves probability density functions, rather than finite probabilities. This poses a difficulty, in that the values of probability density functions can be any non-negative real number; only their integrals are required to be one. As a result, the convergence theory for the EMML algorithm and its various block-iterative versions does not apply unchanged.

For each pixel index j, let $f_j(\cdot)$ be the probability density function (pdf) whose domain is the (uncountably infinite) set of potential LOR with the property that the probability that an emission at j results in an LOR from the set S being recorded is the integral of f_j over S. With x_j the expected number of emissions from j during the scanning time, and

$$x_+ = \sum_{j=1}^{J} x_j,$$

the probability that an emission came from j, given that an emission has happened, is x_j/x_+. Therefore, the probability that an LOR in the set S will be recorded, given that an emission has happened, is the integral over S of the pdf

$$f(\cdot) = \frac{1}{x_+} \sum_{j=1}^{J} x_j f_j(\cdot).$$

For each j let d_j be the probability that an emission from j will be detected, and let

$$d = \frac{1}{x_+} \sum_{j=1}^{J} x_j d_j$$

be the probability that an emission will be detected.

The number of items on the list, M, is also a random variable, which we model as having a Poisson distribution with mean value dx_+. Therefore, the probability of M is

$$p(M) = \exp(-x_+ d)(x_+ d)^M / M!.$$

Given the list of recorded LOR, the likelihood function is then

$$L(x) = p(M) \prod_{m=1}^{M} f(i_m),$$

and the log likelihood function to be maximized is

$$LL(x) = -x_+ d + \sum_{m=1}^{M} \log(Px)_m,$$

where the matrix P has entries

$$P_{mj} = f_j(i_m).$$

Note that

$$(Px)_m = \sum_{j=1}^{J} P_{mj} x_j,$$

so that

$$\sum_{m=1}^{M} (Px)_m = \sum_{j=1}^{J} \Big(\sum_{m=1}^{M} P_{mj} \Big) x_j = \sum_{j=1}^{J} c_j x_j,$$

for

$$c_j = \sum_{m=1}^{M} P_{mj}.$$

Maximizing the log likelihood function is equivalent to minimizing

$$KL(u, Px) - \sum_{m=1}^{M} (Px)_m + x_+ d + \text{constants},$$

where u is the vector whose entries are all one, and therefore equivalent to minimizing

$$F(x) = KL(u, Px) + \sum_{j=1}^{J} (d_j - c_j) x_j.$$

The EMML algorithm itself will minimize only $KL(u, Px)$. The basic problem now is that we have values of probability density functions and the quantities c_j, which can be any positive real numbers, are unrelated to the detectability or sensitivity d_j.

It was shown in [53] that the EMML algorithm can be modified to provide a convergent iterative method for minimizing $F(x)$. This modified EMML algorithm has the iterative step

$$x_j^{k+1} = x_j^k d_j^{-1} \sum_{m=1}^{M} \Big(\frac{1}{(Px^k)_m} \Big).$$

For the finite model, as in [**?**], this is just the usual EMML and convergence follows from known results, but for the continuous model, as in [176], this iterative scheme falls outside the EMML framework and convergence needed to be established, as in [53].

Just as the EMML algorithm must be modified before it can be applied to the continuous model, we must adapt the block-iterative versions as well; see [53] for details.

Part IV

Magnetic Resonance Imaging

Chapter 16

Magnetic Resonance Imaging

In elements with an odd number of protons, such as hydrogen, the nucleus itself will have a net magnetic moment. The objective in *magnetic resonance imaging* (MRI) is to determine the density of such elements in a volume of interest within the body. This is achieved by forcing the individual spinning nuclei to emit signals that, while too weak to be detected alone, are detectable in the aggregate. Fourier-transform estimation and extrapolation techniques play a major role in the rapidly expanding field of magnetic resonance imaging [124].

16.1 Slice Isolation

When the external magnetic field is the *static field* $B_0\mathbf{k}$, that is, the magnetic field has strength B_0 and axis $\mathbf{k} = (0, 0, 1)$, then the Larmor frequency is the same everywhere and equals $\omega_0 = \gamma B_0$, where γ is the gyromagnetic constant. If, instead, we impose an external magnetic field $(B_0 + G_z(z - z_0))\mathbf{k}$, for some constant G_z, then the Larmor frequency is ω_0 only within the plane $z = z_0$. This external field now includes a *gradient field*.

16.2 Tipping

When a magnetic dipole moment that is aligned with $\mathbf{k}$ is given a component in the x, y-plane, it begins to precess around the z-axis, with frequency equal to its Larmor frequency. To create this x, y-plane component, we ap-

ply a *radio-frequency field* (rf field)

$$H_1(t)(\cos(\omega t)\mathbf{i} + \sin(\omega t)\mathbf{j}).$$

The function $H_1(t)$ typically lasts only for a short while, and the effect of imposing this rf field is to tip the aligned magnetic dipole moment axes away from the z-axis, initiating precession. Those dipole axes that tip most are those whose Larmor frequency is ω. Therefore, if we first isolate the slice $z = z_0$ and then choose $\omega = \omega_0$, we tip primarily those dipole axes within the plane $z = z_0$. The dipoles that have been tipped ninety degrees into the x, y-plane generate the strongest signal. How much tipping occurs also depends on $H_1(t)$, so it is common to select $H_1(t)$ to be constant over the time interval $[0, \tau]$, and zero elsewhere, with integral $\frac{\pi}{2\gamma}$. This $H_1(t)$ is called a $\frac{\pi}{2}$-pulse, and tips those axes with Larmor frequency ω_0 into the x, y-plane.

16.3 Imaging

The information we seek about the proton density function is contained within the received signal. By carefully adding gradient fields to the external field, we can make the Larmor frequency spatially varying, so that each frequency component of the received signal contains a piece of the information we seek. The proton density function is then obtained through Fourier transformations.

16.3.1 The Line-Integral Approach

Suppose that we have isolated the plane $z = z_0$ and tipped the aligned axes using a $\frac{\pi}{2}$-pulse. After the tipping has been completed, we introduce an external field $(B_0 + G_x x)\mathbf{k}$, so that now the Larmor frequency of dipoles within the plane $z = z_0$ is $\omega(x) = \omega_0 + \gamma G_x x$, which depends on the x-coordinate of the point. The result is that the component of the received signal associated with the frequency $\omega(x)$ is due solely to those dipoles having that x coordinate. Performing an FFT of the received signal gives us line integrals of the density function along lines in the x, y-plane having fixed x-coordinate.

More generally, if we introduce an external field $(B_0 + G_x x + G_y y)\mathbf{k}$, the Larmor frequency is constant at $\omega(x, y) = \omega_0 + \gamma(G_x x + G_y y) = \omega_0 + \gamma s$ along lines in the x, y-plane with equation

$$G_x x + G_y y = s.$$

Again performing an FFT on the received signal, we obtain the integral of the density function along these lines. In this way, we obtain the three-dimensional Radon transform of the desired density function. The central

slice theorem for this case tells us that we can obtain the Fourier transform of the density function by performing a one-dimensional Fourier transform with respect to the variable s. For each fixed (G_x, G_y) we obtain this Fourier transform along a ray through the origin. By varying the (G_x, G_y) we get the entire Fourier transform. The desired density function is then obtained by Fourier inversion.

16.3.2 Phase Encoding

In the line-integral approach, the line-integral data is used to obtain values of the Fourier transform of the density function along lines through the origin in Fourier space. It would be more convenient to have Fourier-transform values on the points of a rectangular grid. We can obtain this by selecting the gradient fields to achieve *phase encoding*.

Suppose that, after the tipping has been performed, we impose the external field $(B_0+G_y y)\mathbf{k}$ for T seconds. The effect is to alter the precession frequency from ω_0 to $\omega(y) = \omega_0 + \gamma G_y y$. A harmonic $e^{i\omega_0 t}$ is changed to

$$e^{i\omega_0 t} e^{i\gamma G_y y t},$$

so that, after T seconds,we have

$$e^{i\omega_0 T} e^{i\gamma G_y y T}.$$

For $t \geq T$, the harmonic $e^{i\omega_0 t}$ returns, but now it is

$$e^{i\omega_0 t} e^{i\gamma G_y y T}.$$

The effect is to introduce a phase shift of $\gamma G_y y T$. Each point with the same y-coordinate has the same phase shift.

After time T, when this gradient field is turned off, we impose a second external field, $(B_0 + G_x x)\mathbf{k}$. Because this gradient field alters the Larmor frequencies, at times $t \geq T$ the harmonic $e^{i\omega_0 t} e^{i\gamma G_y y T}$ is transformed into

$$e^{i\omega_0 t} e^{i\gamma G_y y T} e^{i\gamma G_x x t}.$$

The received signal is now

$$S(t) = e^{i\omega_0 t} \int\int \rho(x, y) e^{i\gamma G_y y T} e^{i\gamma G_x x t} dx dy,$$

where $\rho(x, y)$ is the value of the proton density function at $x, y)$. Removing the $e^{i\omega_0 t}$ factor, we have

$$\int\int \rho(x, y) e^{i\gamma G_y y T} e^{i\gamma G_x x t} dx dy,$$

which is the Fourier transform of $\rho(x, y)$ at the point $(\gamma G_x t, \gamma G_y T)$. By selecting equi-spaced values of t and altering the G_y, we can get the Fourier transform values on a rectangular grid.

16.4 The General Formulation

The external magnetic field generated in the MRI scanner is generally described by

$$H(r,t) = (H_0 + \mathbf{G}(t)\cdot\mathbf{r})\mathbf{k} + H_1(t)(\cos(\omega t)\mathbf{i} + \sin(\omega t)\mathbf{j}). \tag{16.1}$$

The vectors $\mathbf{i}, \mathbf{j}$, and $\mathbf{k}$ are the unit vectors along the coordinate axes, and $\mathbf{r} = (x, y, z)$. The vector-valued function $\mathbf{G}(t) = (G_x(t), G_y(t), G_z(t))$ produces the *gradient field*

$$\mathbf{G}(t)\cdot\mathbf{r}.$$

The magnetic field component in the x, y plane is the *radio frequency* (rf) field.

If $\mathbf{G}(t) = 0$, then the Larmor frequency is ω_0 everywhere. Using $\omega = \omega_0$ in the rf field, with a $\frac{\pi}{2}$-pulse, will then tip the aligned axes into the x, y-plane and initiate precession. If $\mathbf{G}(t) = \theta$, for some direction vector θ, then the Larmor frequency is constant on planes $\theta\cdot\mathbf{r} = s$. Using an rf field with frequency $\omega = \gamma(H_0 + s)$ and a $\frac{\pi}{2}$-pulse will then tip the axes in this plane into the x, y-plane. The strength of the received signal will then be proportional to the integral, over this plane, of the proton density function. Therefore, the measured data will be values of the three-dimensional Radon transform of the proton density function, which is related to its three-dimensional Fourier transform by the Central Slice Theorem. Later, we shall consider two more widely used examples of $\mathbf{G}(t)$.

16.5 The Received Signal

We assume now that the function $H_1(t)$ is a *short* $\frac{\pi}{2}$*-pulse*, that is, it has constant value over a short time interval $[0, \tau]$ and has integral $\frac{\pi}{2\gamma}$. The received signal produced by the precessing magnetic dipole moments is approximately

$$S(t) = \int_{R^3} \rho(\mathbf{r}) \exp(-i\gamma(\int_0^t \mathbf{G}(s)ds)\cdot\mathbf{r}) \exp(-t/T_2)d\mathbf{r}, \tag{16.2}$$

where $\rho(\mathbf{r})$ is the proton density function, and T_2 is the *transverse* or *spin-spin* relaxation time. The vector integral in the exponent is

$$\int_0^t \mathbf{G}(s)ds = (\int_0^t G_x(s)ds, \int_0^t G_y(s)ds, \int_0^t G_z(s)ds).$$

Now imagine approximating the function $G_x(s)$ over the interval $[0, t]$ by a step function that is constant over small subintervals, that is, $G_x(s)$ is approximately $G_x(n\Delta)$ for s in the interval $[n\Delta, (n+1)\Delta)$, with $n =$

$1, ..., N$ and $\Delta = \frac{t}{N}$. During the interval $[n\Delta, (n+1)\Delta)$, the presence of this gradient field component causes the phase to change by the amount $x\gamma G_x(n\Delta)\Delta$, so that by the time we reach $s = t$ the phase has changed by

$$x \sum_{n=1}^{N} G_x(n\Delta)\Delta,$$

which is approximately $x \int_0^t G_x(s)ds$.

16.5.1 An Example of $\mathbf{G}(t)$

Suppose now that $g > 0$ and θ is an arbitrary direction vector. Let

$$\mathbf{G}(t) = g\theta, \text{ for } \tau \leq t, \tag{16.3}$$

and $\mathbf{G}(t) = 0$ otherwise. Then the received signal $S(t)$ is

$$S(t) = \int_{R^3} \rho(\mathbf{r}) \exp(-i\gamma g(t-\tau)\theta \cdot \mathbf{r}) d\mathbf{r}$$

$$= (2\pi)^{3/2} \hat{\rho}(\gamma g(t-\tau)\theta), \tag{16.4}$$

for $\tau \leq t << T_2$, where $\hat{\rho}$ denotes the three-dimensional Fourier transform of the function $\rho(\mathbf{r})$.

From Equation (16.4) we see that, by selecting different direction vectors and by sampling the received signal $S(t)$ at various times, we can obtain values of the Fourier transform of ρ along lines through the origin in the Fourier domain, called *k-space*. If we had these values for all θ and for all t we would be able to determine $\rho(\mathbf{r})$ exactly. Instead, we have much the same problem as in transmission tomography; only finitely many θ and only finitely many samples of $S(t)$. Noise is also a problem, because the resonance signal is not strong, even though the external magnetic field is.

We may wish to avoid having to estimate the function $\rho(\mathbf{r})$ from finitely many noisy values of its Fourier transform. We can do this by selecting the gradient field $\mathbf{G}(t)$ differently.

16.5.2 Another Example of $\mathbf{G}(t)$

The vector-valued function $\mathbf{G}(t)$ can be written as

$$\mathbf{G}(t) = (G_1(t), G_2(t), G_3(t)).$$

Now we let

$$G_2(t) = g_2,$$

and

$$G_3(t) = g_3,$$

for $0 \leq t \leq \tau$, and zero otherwise, and

$$G_1(t) = g_1,$$

for $\tau \leq t$, and zero otherwise. This means that only $H_0\mathbf{k}$ and the rf field are present up to time τ, and then the rf field is shut off and the gradient field is turned on. Then, for $t \geq \tau$, we have

$$S(t) = (2\pi)^{3/2}\hat{M}_0(\gamma(t-\tau)g_1, \gamma\tau g_2, \gamma\tau g_3).$$

By selecting

$$t_n = n\Delta t + \tau, \text{for } n = 1, ..., N,$$

$$g_{2k} = k\Delta g,$$

and

$$g_{3i} = i\Delta g,$$

for $i, k = -m, ..., m$ we have values of the Fourier transform, $\hat{M}_0$, on a Cartesian grid in three-dimensional k-space. The proton density function, ρ, can then be approximated using the fast Fourier transform.

Although the reconstruction employs the FFT, obtaining the Fourier-transform values on the Cartesian grid can take time. An abdominal scan can last for a couple of hours, during which the patient is confined, motionless and required to hold his or her breath repeatedly. Recent work on *compressed sensing* is being applied to reduce the number of Fourier-transform values that need to be collected, and thereby reduce the scan time [215, 160].

16.6 Compressed Sensing in Image reconstruction

As we have seen, the data one obtains from the scanning process can often be interpreted as values of the Fourier transform of the desired image; this is precisely the case in magnetic-resonance imaging, and approximately true for x-ray transmission tomography, positron-emission tomography (PET) and single-photon emission tomography (SPECT). The images one encounters in medical diagnosis are often approximately locally constant, so the associated array of discrete partial derivatives will be sparse. If this sparse derivative array can be recovered from relatively few Fourier-transform values, then the scanning time can be reduced.

16.6.1 Incoherent Bases

The objective in CS is exploit sparseness to reconstruct a vector f in R^J from relatively few linear functional measurements [95].

Let $U = \{u^1, u^2, ..., u^J\}$ and $V = \{v^1, v^2, ..., v^J\}$ be two orthonormal bases for R^J, with all members of R^J represented as column vectors. For $i = 1, 2, ..., J$, let

$$\mu_i = \max_{1 \leq j \leq J} \{|\langle u^i, v^j \rangle|\}$$

and

$$\mu(U, V) = \max_{i=1}^{I} \mu_i.$$

We know from Cauchy's Inequality that

$$|\langle u^i, v^j \rangle| \leq 1,$$

and from Parseval's Equation

$$\sum_{j=1}^{J} |\langle u^i, v^j \rangle|^2 = ||u^i||^2 = 1.$$

Therefore, we have

$$\frac{1}{\sqrt{J}} \leq \mu(U, V) \leq 1.$$

The quantity $\mu(U, V)$ is the *coherence* measure of the two bases; the closer $\mu(U, V)$ is to the lower bound of $\frac{1}{\sqrt{J}}$, the more *incoherent* the two bases are.

Let f be a fixed member of R^J; we expand f in the V basis as

$$f = x_1 v^1 + x_2 v^2 + ... + x_J v^J.$$

We say that the coefficient vector $x = (x_1, ..., x_J)$ is S-sparse if S is the number of non-zero x_j.

16.6.2 Exploiting Sparseness

If S is small, most of the x_j are zero, but since we do not know which ones these are, we would have to compute all the linear functional values

$$x_j = \langle f, v^j \rangle$$

to recover f exactly. In fact, the smaller S is, the harder it would be to learn anything from randomly selected x_j, since most would be zero. The idea in CS is to obtain measurements of f with members of a different orthonormal basis, which we call the U basis. If the members of U are very

much like the members of V, then nothing is gained. But, if the members of U are quite unlike the members of V, then each inner product measurement

$$y_i = \langle f, u^i \rangle = f^T u^i$$

should tell us something about f. If the two bases are sufficiently incoherent, then relatively few y_i values should tell us quite a bit about f. Specifically, we have the following result due to Candès and Romberg [60]: suppose the coefficient vector x for representing f in the V basis is S-sparse. Select uniformly randomly $M \leq J$ members of the U basis and compute the measurements $y_i = \langle f, u^i \rangle$. Then, if M is sufficiently large, it is highly probable that $z = x$ also solves the problem of minimizing the one-norm

$$||z||_1 = |z_1| + |z_2| + ... + |z_J|,$$

subject to the conditions

$$y_i = \langle g, a^i \rangle = g^T u^i,$$

for those M randomly selected u^i, where

$$g = z_1 v^1 + z_2 v^2 + ... + z_J v^J.$$

This can be formulated as a linear programming problem. The smaller $\mu(U, V)$ is, the smaller the M is permitted to be without reducing the probability of perfect reconstruction.

Part V

Intensity Modulated Radiation Therapy

Chapter 17

Intensity Modulated Radiation Therapy

In *intensity modulated radiation therapy* (IMRT) beamlets of radiation with different intensities are transmitted into the body of the patient. Each voxel within the patient will then absorb a certain dose of radiation from each beamlet. The goal of IMRT is to direct a sufficient dosage to those regions requiring the radiation, those that are designated *planned target volumes* (PTV), while limiting the dosage received by the other regions, the so-called *organs at risk* (OAR). In our discussion here we follow Censor et al. [69].

17.1 The Forward and Inverse Problems

The *forward problem* is to calculate the radiation dose absorbed in the irradiated tissue based on a given distribution of the beamlet intensities. The *inverse problem* is to find a distribution of beamlet intensities, the radiation intensity map, that will result in a clinically acceptable dose distribution. One important constraint is that the radiation intensity map must be implementable, that is, it is physically possible to produce such an intensity map, given the machine's design. There will be limits on the change in intensity between two adjacent beamlets, for example.

17.2 Equivalent Uniform Dosage

The *equivalent uniform dose* (EUD) for tumors is the biologically equivalent dose which, if given uniformly, will lead to the same cell-kill within the tumor volume as the actual non-uniform dose.

17.3 Constraints

Constraints on the EUD received by each voxel of the body are described in *dose space*, the space of vectors whose entries are the doses received at each voxel. Constraints on the deliverable radiation intensities of the beamlets are best described in *intensity space*, the space of vectors whose entries are the intensity levels associated with each of the beamlets. The constraints in dose space will be upper bounds on the dosage received by the OAR and lower bounds on the dosage received by the PTV. The constraints in intensity space are limits on the complexity of the intensity map and on the delivery time, and, obviously, that the intensities be non-negative. Because the constraints operate in two different domains, it is convenient to formulate the problem using these two domains. This leads to a split-feasibility problem.

17.4 The Multi-Set Split-Feasibilty-Problem Model

The *split feasibility problem* (SFP) is to find an x in a given closed convex subset C of R^J such that Ax is in a given closed convex subset Q of R^I, where A is a given real I by J matrix. Because the constraints are best described in terms of several sets in dose space and several sets in intensity space, the SFP model needs to be expanded into the *multi-set* SFP (MSSFP) [68].

It is not uncommon to find that, once the various constraints have been specified, there is no intensity map that satisfies them all. In such cases, it is desirable to find an intensity map that comes as close as possible to satisfying all the constraints. One way to do this, as we shall see, is to minimize a *proximity function*.

17.5 Formulating the Proximity Function

For $i = 1, ..., I$, and $j = 1, ..., J$, let $h_i \geq 0$ be the dose absorbed by the i-th voxel of the patient's body, $x_j \geq 0$ be the intensity of the j-th beamlet of radiation, and $D_{ij} \geq 0$ be the dose absorbed at the i-th voxel due to a unit intensity of radiation at the j-th beamlet. The non-negative matrix D with entries D_{ij} is the *dose influence matrix*.

In intensity space, we have the obvious constraints that $x_j \geq 0$. In addition, there are *implementation constraints*; the available treatment machine will impose its own requirements, such as a limit on the difference in intensities between adjacent beamlets. In dosage space, there will be a lower bound on the dosage delivered to those regions designated as *planned tar-*

get volumes (PTV), and an upper bound on the dosage delivered to those regions designated as *organs at risk* (OAR).

17.6 Equivalent Uniform Dosage Functions

Suppose that S_t is either a PTV or a OAR, and suppose that S_t contains N_t voxels. For each dosage vector $h = (h_1, ..., h_I)^T$ define the *equivalent uniform dosage function* (EUD-function) $e_t(h)$ by

$$e_t(h) = (\frac{1}{N_t} \sum_{i \in S_t} (h_i)^\alpha)^{1/\alpha}, \tag{17.1}$$

where $0 < \alpha < 1$ if S_t is a PTV, and $\alpha > 1$ if S_t is an OAR. The function $e_t(h)$ is convex, for h nonnegative, when S_t is an OAR, and $-e_t(h)$ is convex, when S_t is a PTV. The constraints in dosage space take the form

$$e_t(h) \leq a_t,$$

when S_t is an OAR, and

$$-e_t(h) \leq b_t,$$

when S_t is a PTV. Therefore, we require that $h = Dx$ lie within the intersection of these convex sets.

Chapter 18

Convex Sets

Convex sets and convex functions play important roles in optimization. In this chapter we survey the basic facts concerning the geometry of convex sets. We begin with the geometry of R^J.

18.1 The Geometry of Real Euclidean Space

We denote by R^J the real Euclidean space consisting of all J-dimensional column vectors $x = (x_1, ..., x_J)^T$ with real entries x_j; here the superscript T denotes the transpose of the 1 by J matrix (or, row vector) $(x_1, ..., x_J)$.

18.1.1 Inner Products

For $x = (x_1, ..., x_J)^T$ and $y = (y_1, ..., y_J)^T$ in R^J, the dot product $x \cdot y$ is defined to be

$$x \cdot y = \sum_{j=1}^{J} x_j y_j. \tag{18.1}$$

Note that we can write

$$x \cdot y = y^T x = x^T y, \tag{18.2}$$

where juxtaposition indicates matrix multiplication. The 2-norm, or *Euclidean norm*, or *Euclidean length*, of x is

$$||x||_2 = \sqrt{x \cdot x} = \sqrt{x^T x}. \tag{18.3}$$

The *Euclidean distance* between two vectors x and y in R^J is $||x - y||_2$.

The space R^J, along with its dot product, is an example of a finite-dimensional Hilbert space.

Definition 18.1 *Let V be a real vector space. The scalar-valued function $\langle u, v\rangle$ is called an inner product on V if the following four properties hold, for all u, w, and v in V, and al real c:*

$$\langle u + w, v\rangle = \langle u, v\rangle + \langle w, v\rangle; \tag{18.4}$$

$$\langle cu, v\rangle = c\langle u, v\rangle; \tag{18.5}$$

$$\langle v, u\rangle = \langle u, v\rangle; \tag{18.6}$$

and

$$\langle u, u\rangle \geq 0, \tag{18.7}$$

with equality in Inequality (18.7) if and only if $u = 0$.

The dot product of vectors is an example of an inner product. The properties of an inner product are precisely the ones needed to prove Cauchy's Inequality, which then holds for any inner product. We shall favor the dot product notation $u \cdot v$ for the inner product of vectors, although we shall occasionally use the matrix multiplication form, $v^T u$ or the inner product notation $\langle u, v\rangle$.

18.1.2 Cauchy's Inequality

Cauchy's Inequality, also called the Cauchy-Schwarz Inequality, tells us that

$$|\langle x, y\rangle| \leq ||x||_2 ||y||_2, \tag{18.8}$$

with equality if and only if $y = \alpha x$, for some scalar α. The Cauchy-Schwarz Inequality holds for any inner product.

A simple application of Cauchy's inequality gives us

$$||x + y||_2 \leq ||x||_2 + ||y||_2; \tag{18.9}$$

this is called the *Triangle Inequality*. We say that the vectors x and y are *mutually orthogonal* if $\langle x, y\rangle = 0$.

The *Parallelogram Law* is an easy consequence of the definition of the 2-norm:

$$||x + y||_2^2 + ||x - y||_2^2 = 2||x||_2^2 + 2||y||_2^2. \tag{18.10}$$

It is important to remember that Cauchy's Inequality and the Parallelogram Law hold only for the 2-norm.

18.2 A Bit of Topology

Having the norm allows us to define the distance between two points x and y in R^J as $||x-y||$. Being able to talk about how close points are to each other enables us to define continuity of functions on R^J and to consider topological notions of closed set, open set, interior of a set and boundary of a set.

Definition 18.2 *A subset B of R^J is closed if, whenever x^k is in B for each non-negative integer k and $||x-x^k|| \to 0$, as $k \to +\infty$, then x is in B.*

For example, $B = [0,1]$ is closed as a subset of R, but $B = (0,1)$ is not.

Definition 18.3 *We say that $d \geq 0$ is the distance from the point x to the set B if, for every $\epsilon > 0$, there is b_ϵ in B, with $||x - b_\epsilon||_2 < d + \epsilon$, and no b in B with $||x-b||_2 < d$.*

The distance from the point 0 in R to the set $(0,1)$ is zero, while its distance to the set $(1,2)$ is one. It follows easily from the definitions that, if B is closed and $d = 0$, then x is in B.

Definition 18.4 *The closure of a set B is the set of all points x whose distance from B is zero.*

The closure of the interval $B = (0,1)$ is $[0,1]$.

Definition 18.5 *A subset U of R^J is open if its complement, the set of all points not in U, is closed.*

Definition 18.6 *Let C be a subset of R^J. A point x in C is said to be an interior point of set C if there is $\epsilon > 0$ such that every point z with $||x-z|| < \epsilon$ is in C. The interior of the set C, written int(C), is the set of all interior points of C. It is also the largest open set contained within C.*

For example, the open interval $(0,1)$ is the interior of the intervals $(0,1]$ and $[0,1]$. A set C is open if and only if $C = \text{int}(C)$.

Definition 18.7 *A point x in R^J is said to be a boundary point of set C if, for every $\epsilon > 0$, there are points y_ϵ in C and z_ϵ not in C, both depending on the choice of ϵ, with $||x - y_\epsilon|| < \epsilon$ and $||x - z_\epsilon|| < \epsilon$. The boundary of C is the set of all boundary points of C. It is also the intersection of the closure of C with the closure of its complement.*

For example, the points $x = 0$ and $x = 1$ are boundary points of the set $(0,1]$.

Definition 18.8 *For $k = 0, 1, 2, ...$, let x^k be a vector in R^J. The sequence of vectors $\{x^k\}$ is said to converge to the vector z if, given any $\epsilon > 0$, there is positive integer n, usually depending on ϵ, such that, for every $k > n$, we have $||z - x^k|| \leq \epsilon$. Then we say that z is the limit of the sequence.*

For example, the sequence $\{x^k = \frac{1}{k+1}\}$ in R converges to $z = 0$. The sequence $\{(-1)^k\}$ alternates between 1 and -1, so does not converge. However, the subsequence associated with odd k converges to $z = -1$, while the subsequence associated with even k converges to $z = 1$. The values $z = -1$ and $z = 1$ are called *subsequential limit points*, or, sometimes, *cluster points* of the sequence.

Definition 18.9 *A sequence $\{x^k\}$ of vectors in R^J is said to be bounded if there is a constant $b > 0$, such that $||x^k|| \leq b$, for all k.*

A fundamental result in analysis is the following.

Proposition 18.1 *Every convergent sequence of vectors in R^J is bounded. Every bounded sequence of vectors in R^J has at least one convergent subsequence, therefore, has at least one cluster point.*

18.3 Convex Sets in R^J

In preparation for our discussion of linear and nonlinear programming, we consider some of the basic concepts from the geometry of convex sets.

18.3.1 Basic Definitions

We begin with the basic definitions.

Definition 18.10 *A vector z is said to be a convex combination of the vectors x and y if there is α in the interval $[0, 1]$ such that $z = (1-\alpha)x+\alpha y$.*

Definition 18.11 *A nonempty set C in R^J is said to be convex if, for any distinct points x and y in C, and for any real number α in the interval $(0, 1)$, the point $(1 - \alpha)x + \alpha y$ is also in C; that is, C is closed to convex combinations.*

For example, the unit ball B in R^J, consisting of all x with $||x||_2 \leq 1$, is convex, while the surface of the ball, the set of all x with $||x||_2 = 1$, is not convex.

Definition 18.12 *The convex hull of a set S, denoted conv(S), is the smallest convex set containing S.*

Proposition 18.2 *The convex hull of a set S is the set C of all convex combinations of members of S.*

Definition 18.13 *A subset S of R^J is a subspace if, for every x and y in S and scalars α and β, the linear combination $\alpha x + \beta y$ is again in S.*

A subspace is necessarily a convex set.

Definition 18.14 *The orthogonal complement of a subspace S is the set*

$$S^{\perp} = \{u | u^T s = 0, \text{for every } s \in S\}, \tag{18.11}$$

the set of all vectors u in R^J that are orthogonal to every member of S.

For example, in R^3, the x, y-plane is a subspace and has for its orthogonal complement the z-axis.

Definition 18.15 *A subset M of R^J is a linear manifold if there is a subspace S and a vector b such that*

$$M = S + b = \{x | x = s + b, \text{for some } s \text{ in } S\}.$$

Any linear manifold is convex.

Definition 18.16 *For a fixed column vector a with Euclidean length one and a fixed scalar γ the hyperplane determined by a and γ is the set*

$$H(a, \gamma) = \{z | \langle a, z \rangle = \gamma\}.$$

The hyperplanes $H(a, \gamma)$ are linear manifolds, and the hyperplanes $H(a, 0)$ are subspaces.

Definition 18.17 *Given a subset C of R^J, the affine hull of C, denoted aff(C), is the smallest linear manifold containing C.*

For example, let C be the line segment connecting the two points $(0, 1)$ and $(1, 2)$ in R^2. The affine hull of C is the straight line whose equation is $y = x + 1$.

Definition 18.18 *The dimension of a subset of R^J is the dimension of its affine hull, which is the dimension of the subspace of which it is a translate.*

The set C above has dimension one. A set containing only one point is its own affine hull, since it is a translate of the subspace $\{0\}$.

In R^2, the line segment connecting the points $(0, 1)$ and $(1, 2)$ has no interior; it is a one-dimensional subset of a two-dimensional space and can contain no two-dimensional ball. But, the part of this set without its two end points is a sort of interior, called the *relative interior*.

Definition 18.19 *The relative interior of a subset C of R^J, denoted $ri(C)$, is the interior of C, as defined by considering C as a subset of its affine hull.*

Since a set consisting of a single point is its own affine hull, it is its own relative interior.

Definition 18.20 *A point x in a convex set C is said to be an extreme point of C if the set obtained by removing x from C remains convex.*

Said another way, $x \in C$ is an extreme point of C if x cannot be written as

$$x = (1 - \alpha)y + \alpha z, \tag{18.12}$$

for $y, z \neq x$ and $\alpha \in (0, 1)$. For example, the point $x = 1$ is an extreme point of the convex set $C = [0, 1]$. Every point on the boundary of a sphere in R^J is an extreme point of the sphere. The set of all extreme points of a convex set is denoted $\text{Ext}(C)$.

Definition 18.21 *A non-zero vector d is said to be a direction of unboundedness of a convex set C if, for all x in C and all $\gamma \geq 0$, the vector $x + \gamma d$ is in C.*

For example, if C is the non-negative orthant in R^J, then any non-negative vector d is a direction of unboundedness.

Definition 18.22 *A vector a is normal to a convex set C at the point s in C if*

$$\langle a, c - s \rangle \leq 0, \tag{18.13}$$

for all c in C.

Definition 18.23 *Let C be convex and s in C. The normal cone to C at s, denoted $N_C(s)$, is the set of all vectors a that are normal to C at s.*

18.3.2 Orthogonal Projection onto Convex Sets

The following proposition is fundamental in the study of convexity and can be found in most books on the subject; see, for example, the text by Goebel and Reich [118].

Proposition 18.3 *Given any nonempty closed convex set C and an arbitrary vector x in R^J, there is a unique member of C closest to x, denoted $P_C x$, the orthogonal (or metric) projection of x onto C.*

Proof: If x is in C, then $P_C x = x$, so assume that x is not in C. Then $d > 0$, where d is the distance from x to C. For each positive integer n, select c_n in C with $||x - c_n||_2 < d + \frac{1}{n}$, and $||x - c_n||_2 < ||x - c_{n-1}||_2$. Then the sequence $\{c_n\}$ is bounded; let c^* be any cluster point. It follows easily that $||x - c^*||_2 = d$ and that c^* is in C. If there is any other member c of C with $||x - c||_2 = d$, then, by the Parallelogram Law, we would have $||x - (c^* + c)/2||_2 < d$, which is a contradiction. Therefore, c^* is $P_C x$. ∎

For example, if $C = U$, the unit ball, then $P_C x = x/||x||_2$, for all x such that $||x||_2 > 1$, and $P_C x = x$ otherwise. If C is R^J_+, the nonnegative cone of R^J, consisting of all vectors x with $x_j \geq 0$, for each j, then $P_C x = x_+$, the vector whose entries are $\max(x_j, 0)$. For any closed, convex set C, the distance from x to C is $||x - P_C x||$.

If a nonempty set S is not convex, then the orthogonal projection of a vector x onto S need not be well defined; there may be more than one vector in S closest to x. In fact, it is known that a set S is convex if and only if, for every x not in S, there is a unique point in S closest to x. Note that there may well be some x for which there is a unique closest point in S, but if S is not convex, then there must be at least one point without a unique closest point in S.

Lemma 18.1 *For $H = H(a, \gamma)$, $z = P_H x$ is the vector*

$$z = P_H x = x + (\gamma - \langle a, x \rangle)a. \tag{18.14}$$

We shall use this fact in our discussion of the ART algorithm.

For an arbitrary nonempty closed convex set C in R^J, the orthogonal projection $T = P_C$ is a nonlinear operator, unless, of course, C is a subspace. We may not be able to describe $P_C x$ explicitly, but we do know a useful property of $P_C x$.

Proposition 18.4 *For a given x, a vector z in C is $P_C x$ if and only if*

$$\langle c - z, z - x \rangle \geq 0, \tag{18.15}$$

for all c in the set C.

Proof: Let c be arbitrary in C and α in $(0, 1)$. Then

$$||x - P_C x||_2^2 \leq ||x - (1 - \alpha)P_C x - \alpha c||_2^2 = ||x - P_C x + \alpha(P_C x - c)||_2^2$$

$$= ||x - P_C x||_2^2 - 2\alpha\langle x - P_C x, c - P_C x \rangle + \alpha^2 ||P_C x - c||_2^2. \tag{18.16}$$

Therefore,

$$-2\alpha\langle x - P_C x, c - P_C x \rangle + \alpha^2 ||P_C x - c||_2^2 \geq 0, \tag{18.17}$$

so that

$$2\langle x - P_C x, c - P_C x\rangle \leq \alpha ||P_C x - c||_2^2. \tag{18.18}$$

Taking the limit, as $\alpha \to 0$, we conclude that

$$\langle c - P_C x, P_C x - x\rangle \geq 0. \tag{18.19}$$

If z is a member of C that also has the property

$$\langle c - z, z - x\rangle \geq 0, \tag{18.20}$$

for all c in C, then we have both

$$\langle z - P_C x, P_C x - x\rangle \geq 0, \tag{18.21}$$

and

$$\langle z - P_C x, x - z\rangle \geq 0. \tag{18.22}$$

Adding on both sides of these two inequalities lead to

$$\langle z - P_C x, P_C x - z\rangle \geq 0. \tag{18.23}$$

But,

$$\langle z - P_C x, P_C x - z\rangle = -||z - P_C x||_2^2, \tag{18.24}$$

so it must be the case that $z = P_C x$. This completes the proof. ∎

18.4 Some Results on Projections

The characterization of the orthogonal projection operator P_C given by Proposition 18.4 has a number of important consequences.

Corollary 18.1 *Let S be any subspace of R^J. Then, for any x in R^J and s in S, we have*

$$\langle P_S x - x, s\rangle = 0. \tag{18.25}$$

Proof: Since S is a subspace, $s + P_S x$ is again in S, for all s, as is cs, for every scalar c. ∎

This corollary enables us to prove the Decomposition Theorem.

Theorem 18.1 *Let S be any subspace of R^J and x any member of R^J. Then there are unique vectors s in S and u in $S^\perp$ such that $x = s + u$. The vector s is $P_S x$ and the vector u is $P_{S^\perp} x$.*

Proof: For the given x we take $s = P_S x$ and $u = x - P_S x$. Corollary 18.1 assures us that u is in $S^\perp$. Now we need to show that this decomposition is unique. To that end, suppose that we can write $x = s_1 + u_1$, with s_1 in S and u_1 in $S^\perp$. Then Proposition 18.4 tells us that, since $s_1 - x$ is orthogonal to every member of S, s_1 must be $P_S x$. ∎

This theorem is often presented in a slightly different manner.

Theorem 18.2 *Let A be a real I by J matrix. Then every vector b in R^I can be written uniquely as $b = Ax + w$, where $A^T w = 0$.*

To derive Theorem 18.2 from Theorem 18.1, we simply let $S = \{Ax | x \in R^J\}$. Then $S^\perp$ is the set of all w such that $A^T w = 0$. It follows that w is the member of the null space of A^T closest to b.

Here are additional consequences of Proposition 18.4.

Corollary 18.2 *Let S be any subspace of R^J, d a fixed vector, and V the linear manifold $V = S + d = \{v = s + d | s \in S\}$, obtained by translating the members of S by the vector d. Then, for every x in R^J and every v in V, we have*

$$\langle P_V x - x, v - P_V x \rangle = 0. \tag{18.26}$$

Proof: Since v and $P_V x$ are in V, they have the form $v = s + d$, and $P_V x = \hat{s} + d$, for some s and $\hat{s}$ in S. Then $v - P_V x = s - \hat{s}$. ∎

Corollary 18.3 *Let H be the hyperplane $H(a, \gamma)$. Then, for every x, and every h in H, we have*

$$\langle P_H x - x, h - P_H x \rangle = 0. \tag{18.27}$$

Corollary 18.4 *Let S be a subspace of R^J. Then $(S^\perp)^\perp = S$.*

Proof: Every x in R^J has the form $x = s + u$, with s in S and u in $S^\perp$. Suppose x is in $(S^\perp)^\perp$. Then $u = 0$. ∎

Chapter 19

The Split Feasibility Problem

The *split feasibility problem* (SFP) [67] is to find $c \in C$ with $Ac \in Q$, if such points exist, where A is a real I by J matrix and C and Q are nonempty, closed convex sets in R^J and R^I, respectively. In this chapter we discuss the CQ algorithm for solving the SFP, as well as recent extensions and applications.

19.1 The CQ Algorithm

In [54] the CQ algorithm for solving the SFP was presented, for the real case. It has the iterative step

$$x^{k+1} = P_C(x^k - \gamma A^T(I - P_Q)Ax^k), \tag{19.1}$$

where I is the identity operator and $\gamma \in (0, 2/\rho(A^TA))$, for $\rho(A^TA)$ the spectral radius of the matrix A^TA, which is also its largest eigenvalue. The CQ algorithm can be extended to the complex case, in which the matrix A has complex entries, and the sets C and Q are in C^J and C^I, respectively. The iterative step of the extended CQ algorithm is then

$$x^{k+1} = P_C(x^k - \gamma A^\dagger(I - P_Q)Ax^k). \tag{19.2}$$

The CQ algorithm converges to a solution of the SFP, for any starting vector x^0, whenever the SFP has solutions. When the SFP has no solutions, the CQ algorithm converges to a minimizer of the function

$$f(x) = \frac{1}{2}||P_QAx - Ax||_2^2$$

over the set C, provided such constrained minimizers exist [55]. The CQ algorithm employs the relaxation parameter γ in the interval $(0, 2/L)$, where L is the largest eigenvalue of the matrix A^TA. Choosing the best relaxation parameter in any algorithm is a nontrivial procedure. Generally speaking, we want to select γ near to $1/L$. If A is normalized so that each row has length one, then the spectral radius of A^TA does not exceed the maximum number of nonzero elements in any column of A. A similar upper bound on $\rho(A^TA)$ can be obtained for non-normalized, ϵ-sparse A.

19.2 Particular Cases of the CQ Algorithm

It is easy to find important examples of the SFP: if $C \subseteq R^J$ and $Q = \{b\}$ then solving the SFP amounts to solving the linear system of equations $Ax = b$; if C is a proper subset of R^J, such as the nonnegative cone, then we seek solutions of $Ax = b$ that lie within C, if there are any. Generally, we cannot solve the SFP in closed form and iterative methods are needed.

A number of well known iterative algorithms, such as the Landweber [148] and projected Landweber methods (see [15]), are particular cases of the CQ algorithm.

19.2.1 The Landweber algorithm

With x^0 arbitrary and $k = 0, 1, ...$ let

$$x^{k+1} = x^k + \gamma A^T(b - Ax^k). \tag{19.3}$$

This is the Landweber algorithm.

19.2.2 The Projected Landweber Algorithm

For a general nonempty closed convex C, x^0 arbitrary, and $k = 0, 1, ...$, the projected Landweber method for finding a solution of $Ax = b$ in C has the iterative step

$$x^{k+1} = P_C(x^k + \gamma A^T(b - Ax^k)). \tag{19.4}$$

19.2.3 Convergence of the Landweber Algorithms

From the convergence theorem for the CQ algorithm it follows that the Landweber algorithm converges to a solution of $Ax = b$ and the projected Landweber algorithm converges to a solution of $Ax = b$ in C, whenever such solutions exist. When there are no solutions of the desired type, the Landweber algorithm converges to a least squares approximate solution

of $Ax = b$, while the projected Landweber algorithm will converge to a minimizer, over the set C, of the function $||b - Ax||_2$, whenever such a minimizer exists.

19.2.4 The Simultaneous ART (SART)

Another example of the CQ algorithm is the *simultaneous algebraic reconstruction technique* (SART) [4] for solving $Ax = b$, for nonnegative matrix A. Let A be an I by J matrix with nonnegative entries. Let $A_{i+} > 0$ be the sum of the entries in the ith row of A and $A_{+j} > 0$ be the sum of the entries in the jth column of A. Consider the (possibly inconsistent) system $Ax = b$. The SART algorithm has the following iterative step:

$$x_j^{k+1} = x_j^k + \frac{1}{A_{+j}} \sum\nolimits_{i=1}^{I} A_{ij}(b_i - (Ax^k)_i)/A_{i+}.$$

We make the following changes of variables:

$$B_{ij} = A_{ij}/(A_{i+})^{1/2}(A_{+j})^{1/2},$$

$$z_j = x_j(A_{+j})^{1/2},$$

and

$$c_i = b_i/(A_{i+})^{1/2}.$$

Then the SART iterative step can be written as

$$z^{k+1} = z^k + B^T(c - Bz^k).$$

This is a particular case of the Landweber algorithm, with $\gamma = 1$. The convergence of SART follows from that of the CQ algorithm, once we know that the largest eigenvalue of B^TB is less than two; in fact, we show that it is one [54].

If B^TB had an eigenvalue greater than one and some of the entries of A are zero, then, replacing these zero entries with very small positive entries, we could obtain a new A whose associated B^TB also had an eigenvalue greater than one. Therefore, we assume, without loss of generality, that A has all positive entries. Since the new B^TB also has only positive entries, this matrix is irreducible and the Perron-Frobenius Theorem applies. We shall use this to complete the proof.

Let $u = (u_1, ..., u_J)^T$ with $u_j = (A_{+j})^{1/2}$ and $v = (v_1, ..., v_I)^T$, with $v_i = (A_{i+})^{1/2}$. Then we have $Bu = v$ and $B^Tv = u$; that is, u is an eigenvector of B^TB with associated eigenvalue equal to one, and all the entries of u are positive, by assumption. The Perron-Frobenius theorem applies and tells us that the eigenvector associated with the largest eigenvalue has all positive entries. Since the matrix B^TB is symmetric its eigenvectors are orthogonal; therefore u itself must be an eigenvector associated with the largest eigenvalue of B^TB. The convergence of SART follows.

19.2.5 Application of the CQ Algorithm in Dynamic ET

To illustrate how an image reconstruction problem can be formulated as a SFP, we consider briefly *emission computed tomography* (ET) image reconstruction. The objective in ET is to reconstruct the internal spatial distribution of intensity of a radionuclide from counts of photons detected outside the patient. In static ET the intensity distribution is assumed constant over the scanning time. Our data are photon counts at the detectors, forming the positive vector b and we have a matrix A of detection probabilities; our model is $Ax = b$, for x a nonnegative vector. We could then take $Q = \{b\}$ and $C = R^N_+$, the nonnegative cone in R^N.

In *dynamic* ET [103] the intensity levels at each voxel may vary with time. The observation time is subdivided into, say, T intervals and one static image, call it x^t, is associated with the time interval denoted by t, for $t = 1, ..., T$. The vector x is the concatenation of these T image vectors x^t. The discrete time interval at which each data value is collected is also recorded and the problem is to reconstruct this succession of images.

Because the data associated with a single time interval is insufficient, by itself, to generate a useful image, one often uses prior information concerning the time history at each fixed voxel to devise a model of the behavior of the intensity levels at each voxel, as functions of time. One may, for example, assume that the radionuclide intensities at a fixed voxel are increasing with time, or are concave (or convex) with time. The problem then is to find $x \geq 0$ with $Ax = b$ and $Dx \geq 0$, where D is a matrix chosen to describe this additional prior information. For example, we may wish to require that, for each fixed voxel, the intensity is an increasing function of (discrete) time; then we want

$$x_j^{t+1} - x_j^t \geq 0,$$

for each t and each voxel index j. Or, we may wish to require that the intensity at each voxel describes a concave function of time, in which case nonnegative second differences would be imposed:

$$(x_j^{t+1} - x_j^t) - (x_j^{t+2} - x_j^{t+1}) \geq 0.$$

In either case, the matrix D can be selected to include the left sides of these inequalities, while the set Q can include the nonnegative cone as one factor.

19.2.6 More on the CQ Algorithm

One of the obvious drawbacks to the use of the CQ algorithm is that we would need the projections P_C and P_Q to be easily calculated. Several

authors have offered remedies for that problem, using approximations of the convex sets by the intersection of hyperplanes and orthogonal projections onto those hyperplanes [214].

Part VI

Appendices

Chapter 20

Appendix: Some Probability Theory

In this chapter we review a few important results from the theory of probability.

20.1 Independent Random Variables

Let $X_1, ..., X_N$ be N independent real random variables with the same mean (that is, expected value) μ and same variance σ^2. The main consequence of independence is that $E(X_i X_j) = E(X_i)E(X_j) = \mu^2$ for $i \neq j$. Then, it is easily shown that the *sample average*

$$\bar{X} = N^{-1} \sum_{n=1}^{N} X_n$$

has μ for its mean and σ^2/N for its variance.

Exercise 20.1 *Prove these two assertions.*

20.2 Maximum Likelihood Parameter Estimation

Suppose that the random variable X has a probability density function $p(x;\theta)$, where θ is an unknown parameter. A common problem in statistics is to estimate θ from independently sampled values of X, say $x_1, ..., x_N$. A

frequently used approach is to maximize the function of θ given by

$$L(\theta) = L(\theta; x_1, ..., x_N) = \prod_{n=1}^{N} p(x_n; \theta).$$

The function $L(\theta)$ is the *likelihood function* and a value of θ maximizing $L(\theta)$ is a *maximum likelihood estimate*. We give two examples of maximum likelihood (ML) estimation.

20.2.1 An Example: The Bias of a Coin

Let θ in the interval $[0, 1]$ be the unknown probability of success on one trial of a binomial distribution (a coin flip, for example), so that the probability of k successes in N trials is $L(\theta; k, N) = \frac{N!}{k!(N-k)!}\theta^k(1-\theta)^{N-k}$, for $k = 0, 1, ..., N$. If we have observed N trials and have recorded k successes, we can estimate θ by selecting that $\hat{\theta}$ for which $L(\theta, k, N)$ is maximized as a function of θ.

Exercise 20.2 *Show that, for the binomial case described above, the maximum likelihood estimate of θ is $\hat{\theta} = k/N$.*

20.2.2 Estimating a Poisson Mean

A random variable X taking on only nonnegative integer values is said to have the *Poisson distribution* with parameter $\lambda > 0$ if, for each nonnegative integer k, the probability p_k that X will take on the value k is given by

$$p_k = e^{-\lambda}\lambda^k/k!.$$

Exercise 20.3 *Show that the sequence $\{p_k\}_{k=0}^{\infty}$ sums to one.*

Exercise 20.4 *Show that the expected value $E(X)$ is λ, where the expected value in this case is*

$$E(X) = \sum_{k=0}^{\infty} kp_k.$$

Exercise 20.5 *Show that the variance of X is also λ, where the variance of X in this case is*

$$\text{var}(X) = \sum_{k=0}^{\infty}(k-\lambda)^2 p_k.$$

Exercise 20.6 *Show that the ML estimate of λ based on N independent samples is the sample mean.*

20.3 Independent Poisson Random Variables

Let $Z_1, ..., Z_N$ be independent Poisson random variables with expected value $E(Z_n) = \lambda_n$. Let $\mathbf{Z}$ be the random vector with Z_n as its entries, λ the vector whose entries are the λ_n, and $\lambda_+ = \sum_{n=1}^N \lambda_n$. Then the probability function for $\mathbf{Z}$ is

$$f(\mathbf{Z}|\lambda) = \prod_{n=1}^N \lambda_n^{z_n} \exp(-\lambda_n)/z_n! = \exp(-\lambda_+) \prod_{n=1}^N \lambda_n^{z_n}/z_n! \,. \tag{20.1}$$

Now let $Y = \sum_{n=1}^N Z_n$. Then, the probability function for Y is

$$\begin{aligned} \text{Prob}(Y = y) &= \text{Prob}(Z_1 + ... + Z_N = y) \\ &= \sum_{z_1+...z_N=y} \exp(-\lambda_+) \prod_{n=1}^N \lambda_n^{z_n}/z_n! \,. \end{aligned} \tag{20.2}$$

But, as we shall see shortly, we have

$$\sum_{z_1+...z_N=y} \exp(-\lambda_+) \prod_{n=1}^N \lambda_n^{z_n}/z_n! = \exp(-\lambda_+)\lambda_+^y/y! \,. \tag{20.3}$$

Therefore, Y is a Poisson random variable with $E(Y) = \lambda_+$.

When we observe an instance of y, we can consider the conditional distribution $f(\mathbf{Z}|\lambda, y)$ of $\{Z_1, ..., Z_N\}$, subject to $y = Z_1 + ... + Z_N$. We have

$$f(\mathbf{Z}|\lambda, y) = \frac{y!}{z_1!...z_N!}(\frac{\lambda_1}{\lambda_+})^{z_1}...(\frac{\lambda_N}{\lambda_+})^{z_N}. \tag{20.4}$$

This is a *multinomial distribution.* Given y and λ, the conditional expected value of Z_n is then $E(Z_n|\lambda, y) = y\lambda_n/\lambda_+$. To see why Equation (20.3) is true, we discuss the multinomial distribution.

20.4 The Multinomial Distribution

When we expand the quantity $(a_1 + ... + a_N)^y$, we obtain a sum of terms, each of the form $a_1^{z_1}...a_N^{z_N}$, with $z_1 + ... + z_N = y$. How many terms of the same form are there? There are N variables. We are to select z_n of type n, for each $n = 1, ..., N$, to get $y = z_1 + ... + z_N$ factors. Imagine y blank spaces, to be filled in by various factor types as we do the selection. We select z_1 of these blanks and mark them a_1, for type one. We can do that in $\binom{y}{z_1}$ ways. We then select z_2 of the remaining blank spaces and enter

a_2 in them; we can do this in $\binom{y-z_1}{z_2}$ ways. Continuing in this way, we find that we can select the N factor types in

$$\binom{y}{z_1}\binom{y-z_1}{z_2}...\binom{y-(z_1+...+z_{N-2})}{z_{N-1}} \tag{20.5}$$

ways, or in

$$\frac{y!}{z_1!(y-z_1)!}...\frac{(y-(z_1+...+z_{N-2}))!}{z_{N-1}!(y-(z_1+...+z_{N-1}))!} = \frac{y!}{z_1!...z_N!}. \tag{20.6}$$

This tells us in how many different sequences the factor types can be selected. Applying this, we get the multinomial theorem:

$$(a_1+...+a_N)^y = \sum_{z_1+...+z_N=y} \frac{y!}{z_1!...z_N!} a_1^{z_1}...a_N^{z_N}. \tag{20.7}$$

Select $a_n = \lambda_n/\lambda_+$. Then,

$$\begin{aligned} 1 = 1^y &= (\frac{\lambda_1}{\lambda_+}+...+\frac{\lambda_N}{\lambda_+})^y \\ &= \sum_{z_1+...+z_N=y} \frac{y!}{z_1!...z_N!}(\frac{\lambda_1}{\lambda_+})^{z_1}...(\frac{\lambda_N}{\lambda_+})^{z_N}. \end{aligned} \tag{20.8}$$

From this we get

$$\sum_{z_1+...z_N=y} \exp(-\lambda_+)\prod_{n=1}^{N}\lambda_n^{z_n}/z_n! = \exp(-\lambda_+)\lambda_+^y/y!\ . \tag{20.9}$$

20.5 Characteristic Functions

The Fourier transform shows up in probability theory in the guise of the *characteristic function* of a random variable. The characteristic function is related to, but more general than, the moment-generating function and serves much the same purposes.

A real-valued random variable X is said to have the probability density function (pdf) $f(x)$ if, for any interval $[a,b]$, the probability that X takes its value within this interval is given by the integral $\int_a^b f(x)dx$. To be a pdf, $f(x)$ must be nonnegative and $\int_{-\infty}^{\infty} f(x)dx = 1$. The *characteristic function* of X is then

$$F(\omega) = \int_{-\infty}^{\infty} f(x)e^{ix\omega}dx.$$

The formulas for differentiating the Fourier transform are quite useful in determining the moments of a random variable.

The *expected value* of X is

$$E(X) = \int_{-\infty}^{\infty} xf(x)dx,$$

and for any real-valued function $g(x)$ the expected value of the random variable $g(X)$ is

$$E(g(X)) = \int_{-\infty}^{\infty} g(x)f(x)dx.$$

The nth moment of X is

$$E(X^n) = \int_{-\infty}^{\infty} x^n f(x)dx;$$

the *variance* of X is then $\text{var}(X) = E(X^2) - E(X)^2$. It follows, therefore, that the nth moment of the random variable X is given by

$$E(X^n) = (i)^n F^{(n)}(0).$$

If we have N real-valued random variables $X_1, ..., X_N$, their *joint probability density function* is $f(x_1, ..., x_N) \geq 0$ having the property that, for any intervals $[a_1, b_1], ..., [a_N, b_N]$, the probability that X_n takes its value within $[a_n, b_n]$, for each n, is given by the multiple integral

$$\int_{a_1}^{b_1} \cdots \int_{a_N}^{b_N} f(x_1, ..., x_N)dx_1 \cdots dx_N.$$

The joint moments are then

$$E(X_1^{m_1} \cdots X_N^{m_N}) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} x_1^{m_1} \cdots x_N^{m_N} f(x_1, ..., x_N)dx_1 \cdots dx_N.$$

The joint moments can be calculated by evaluating at zero the partial derivatives of the characteristic function of the joint pdf.

The random variables are said to be *independent* if

$$f(x_1, ..., x_N) = f(x_1) \cdots f(x_N),$$

where, in keeping with the convention used in the probability literature, $f(x_n)$ denotes the pdf of the random variable X_n.

If X and Y are independent random variables with probability density functions $f(x)$ and $g(y)$, then the probability density function for the random variable $Z = X + Y$ is $(f * g)(z)$, the convolution of f and g. To see this, we first calculate the cumulative distribution function

$$H(z) = \text{Prob}\,(X + Y \leq z),$$

which is

$$H(z) = \int_{x=-\infty}^{+\infty} \int_{y=-\infty}^{z-x} f(x)g(y)dydx.$$

Using the change of variable $t = x + y$, we get

$$H(z) = \int_{x=-\infty}^{+\infty} \int_{t=-\infty}^{z} f(x)g(t-x)dtdx.$$

The pdf for the random variable Z is $h(z) = H'(z)$, the derivative of $H(z)$. Differentiating the inner integral with respect to z, we obtain

$$h(z) = \int_{x=-\infty}^{+\infty} f(x)g(z-x)dx;$$

therefore, $h(z) = (f * g)(z)$. It follows that the characteristic function for the random variable $Z = X + Y$ is the product of the characteristic functions for X and Y.

20.6 Gaussian Random Variables

A real-valued random variable X is called *Gaussian* or *normal* with mean μ and variance σ^2 if its probabilty density function (pdf) is

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{(x-\mu)^2}{2\sigma^2}). \tag{20.10}$$

In the statistical literature a normal random variable is *standard* if its mean is $\mu = 0$ and its variance is $\sigma^2 = 1$.

20.6.1 Gaussian Random Vectors

Suppose now that $Z_1, ..., Z_N$ are independent standard normal random variables. Then, their joint pdf is the function

$$f(z_1, ..., z_N) = \prod_{n=1}^{N} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}z_n^2) = \frac{1}{(\sqrt{2\pi})^N} \exp(-\frac{1}{2}(z_1^2 + ... + z_N^2)).$$

By taking linear combinations of these random variables, we can obtain a new set of normal random variables that are no longer independent. For each $m = 1, ..., M$ let

$$X_m = \sum_{n=1}^{N} A_{mn} Z_n.$$

Then $E(X_m) = 0$.

The *covariance matrix* associated with the X_m is the matrix R with entries $R_{mn} = E(X_m X_n)$, $m, n = 1, 2, ..., M$. We have

$$E(X_m X_n) = \sum_{k=1}^{N} A_{mk} \sum_{j=1}^{N} A_{nj} E(Z_k Z_j).$$

Since the Z_n are independent with mean zero, we have $E(Z_k Z_j) = 0$ for $k \neq j$ and $E(Z_k^2) = 1$. Therefore,

$$E(X_m X_n) = \sum_{k=1}^{N} A_{mk} A_{nk},$$

and the covariance matrix is $R = AA^T$.

Writing $\mathbf{X} = (X_1, ..., X_M)^T$ and $\mathbf{Z} = (Z_1, ..., Z_N)^T$, we have $\mathbf{X} = A\mathbf{Z}$, where A is the M by N matrix with entries A_{mn}. Using the standard formulas for changing variables, we find that the joint pdf for the random variables $X_1, ..., X_M$ is

$$f(x_1, ..., x_M) = \frac{1}{\sqrt{\det(R)}} \frac{1}{(\sqrt{2\pi})^N} \exp(-\frac{1}{2}\mathbf{x}^T R^{-1}\mathbf{x}),$$

with $\mathbf{x} = (x_1, ..., x_N)^T$. For the remainder of this chapter, we limit the discussion to the case of $M = N = 2$ and use the notation $X_1 = X$, $X_2 = Y$ and $f(x_1, x_2) = f(x, y)$. We also let $\rho = E(XY)/\sigma_1\sigma_2$.

The two-dimensional FT of the function $f(x, y)$, the characteristic function of the Gaussian random vector $\mathbf{X}$, is

$$F(\alpha, \beta) = \exp(-\frac{1}{2}(\sigma_1^2\alpha^2 + \sigma_2^2\beta^2 + 2\sigma_1\sigma_2\rho\alpha\beta)).$$

Exercise 20.7 *Use partial derivatives of $F(\alpha, \beta)$ to show that $E(X^2Y^2) = 2\sigma_1^2\sigma_2^2\rho^2$.*

Exercise 20.8 *Show that $E(X^2Y^2) = E(X^2)E(Y^2) + 2E(XY)^2$.*

20.6.2 Complex Gaussian Random Variables

Let X and Y be independent real Gaussian random variables with means μ_x and μ_y, respectively, and common variance σ^2. Then $W = X + iY$ is a *complex Gaussian random variable* with mean $\mu_w = E(W) = \mu_x + i\mu_y$ and variance $\sigma_w^2 = 2\sigma^2$.

The results of Exercise 20.7 extend to complex Gaussian random variables W and V. In the complex case we have

$$E(|V|^2|W|^2) = E(|V|^2)E(|W|^2) + |E(V\overline{W})|^2.$$

This is important in optical image processing, where it is called the *Hanbury-Brown Twiss effect* and provides the basis for intensity interferometry [112]. The main point is that we can obtain magnitude information about $E(V\overline{W})$, but not phase information, by measuring the correlation between the magnitudes of V and W; that is, we learn something about $E(V\overline{W})$ from intensity measurements. Since we have only the magnitude of $E(V\overline{W})$, we then have a *phase problem*.

Chapter 21

Appendix: Bayesian Methods

21.1 Using *A Priori* Information

We know that to get information out we need to put information in; but how to do it is the problem. One approach that is quite popular within the image-reconstruction community is the use of statistical Bayesian methods and maximum *a posteriori* (MAP) estimation.

21.2 Conditional Probabilities and Bayes' Rule

Suppose that A and B are two events with positive probabilities $P(A)$ and $P(B)$, respectively. The *conditional probability* of B, given A, is defined to be $P(B|A) = P(A \cap B)/P(A)$. It follows that Bayes' Rule holds:

$$P(A|B) = P(B|A)P(A)/P(B).$$

To illustrate the use of this rule, we consider the following example.

21.2.1 An Example of Bayes' Rule

Suppose that, in a certain town, 10 percent of the adults over 50 have diabetes. The town doctor correctly diagnoses those with diabetes as having the disease 95 percent of the time. In two percent of the cases he incorrectly diagnoses those not having the disease as having it. Let D mean that the patient has diabetes, N that the patient does not have the disease, A that a diagnosis of diabetes is made, and B that a diagnosis of diabetes is

not made. The probability that he will diagnose a given adult as having diabetes is given by the rule of total probability:

$$P(A) = P(A|D)P(D) + P(A|N)P(N).$$

In this example, we obtain $P(A) = 0.113$. Now suppose a patient receives a diagnosis of diabetes. What is the probability that this diagnosis is correct? In other words, what is $P(D|A)$? For this we use Bayes' Rule:

$$P(D|A) = P(A|D)P(D)/P(A),$$

which turns out to be 0.84.

21.2.2 Using Prior Probabilities

So far nothing is controversial. The fun begins when we attempt to broaden the use of Bayes' Rule to ascribe *a priori* probabilities to quantities that are not random. The example used originally by Thomas Bayes in the eighteenth century is as follows. Imagine a billiard table with a line drawn across it parallel to its shorter side, cutting the table into two rectangular regions, the nearer called A and the farther B. Balls are tossed on to the table, coming to rest in either of the two regions. Suppose that we are told only that after N such tosses n of the balls ended up in region A. What is the probability that the next ball will end up in region A?

At first it would seem that we cannot answer this question unless we are told the probability of any ball ending up in region A; Bayes argues differently, however. Let A be the event that a ball comes to rest in region A, and let $P(A) = x$ be the unknown probability of coming to rest in region A; we may consider x to be the relative area of region A, although this is not necessary. Let D be the event that n out of N balls end up in A. Then,

$$P(D|x) = \binom{N}{n} x^n (1-x)^{N-n}.$$

Bayes then adopts the view that the horizontal line on the table was randomly positioned so that the unknown x can be treated as a random variable. Using Bayes' Rule, we have

$$P(x|D) = P(D|x)P(x)/P(D),$$

where $P(x)$ is the probability density function (pdf) of the random variable x, which Bayes takes to be uniform over the interval $[0, 1]$. Therefore, we have

$$P(x|D) = c\binom{N}{n} x^n (1-x)^{N-n},$$

where c is chosen so as to make $P(x|D)$ a pdf.

Exercise 21.1 *Use integration by parts, or look up facts about the Beta function, to show that*

$$\binom{N}{n}\int_0^1 x^n(1-x)^{N-n}dx = 1/(N+1),$$

and

$$\binom{N+1}{n+1}\int_0^1 x^{n+1}(1-x)^{N-n}dx = 1/(N+2)$$

for $n = 0, 1, ..., N$.

From the exercise we can conclude that $c = N + 1$; therefore we have the pdf $P(x|D)$. Now we want to estimate x itself. One way to do this is to calculate the expected value of this pdf, which, according to the exercise, is $(n+1)/(N+2)$. So even though we do not know x, we can reasonably say $(n+1)/(N+2)$ is the probability that the next ball will end up in region A, given the behavior of the previous N balls.

There is a second way to estimate x; we can find the value of x for which the pdf reaches its maximum. A quick calculation shows this value to be n/N. This estimate of x is not the same as the one we calculated using the expected value but they are close for large N.

What is controversial here is the decision to treat the positioning of the line as a random act, with the resulting probability x a random variable, as well as the specification of the pdf governing x. Even if x were a random variable, we do not necessarily know its pdf. Bayes takes the pdf to be uniform over $[0, 1]$ more as an expression of ignorance than of knowledge. It is this broader use of prior probabilities that is generally known as *Bayesian methods* and not the use of Bayes' Rule itself.

21.3 Maximum *A Posteriori* Estimation

Bayesian methods provide us with an alternative to maximum likelihood parameter estimation. Suppose that a random variable (or vector) Z has the pdf $f(z;\theta)$, where θ is a parameter. When this pdf is viewed as a function of θ, not of z, it is called the *likelihood function*. Having observed an instance of Z, call it z, we can estimate the parameter θ by selecting that value for which the likelihood function $f(z;\theta)$ has its maximum. This is the *maximum likelihood* (ML) estimator. Alternatively, suppose that we treat θ itself as one value of a random variable Θ having its own pdf, say $g(\theta)$. Then, Bayes' Rule says that the conditional pdf of Θ, given z, is

$$g(\theta|z) = f(z;\theta)g(\theta)/f(z),$$

where

$$f(z) = \int f(z;\theta)g(\theta)d\theta.$$

The maximum *a posteriori* (MAP) estimate of θ is the one for which the function $g(\theta|z)$ is maximized. Taking logs and ignoring terms that do not involve θ, we find that the MAP estimate of θ maximizes the function $\log f(z;\theta) + \log g(\theta)$.

Because the ML estimate maximizes $\log f(z;\theta)$, the MAP estimate is viewed as involving a *penalty term* $\log g(\theta)$ missing in the ML approach. This penalty function is based on the prior pdf $g(\theta)$. We choose $g(\theta)$ in a way that expresses our prior knowledge of the parameter θ.

21.4 MAP Reconstruction of Images

In emission tomography the parameter θ is actually a vectorized image that we wish to reconstruct and the observed data constitute z. Our prior knowledge about θ may be that the true image is near some prior estimate, say ρ, of the correct answer, in which case $g(\theta)$ is selected to peak at ρ [151]. Frequently our prior knowledge of θ is that the image it represents is nearly constant locally, except for edges. Then $g(\theta)$ is designed to weight more heavily the locally-constant images and less heavily the others [115, 121, 153, 126, 157].

21.5 Penalty Function Methods

The so-called *penalty function* that appears in the MAP approach comes from a prior pdf for θ. This suggests more general methods that involve a penalty function term that does not necessarily emerge from Bayes' Rule [43]. Such methods are well-known in optimization. We are free to estimate θ as the maximizer of a suitable objective function whether or not that function is a posterior probability. Using penalty function methods permits us to avoid the controversies that accompany Bayesian methods.

Chapter 22

Appendix: Discrete Signal Processing

Although we usually model real-world distributions as functions of continuous variables, while the data we actually obtain are finite, it is standard practice to develop signal processing fundamentals within the context of infinite sequences, or functions of discrete variables. Infinite sequences arise when we sample functions of continuous variables, or when we extend finite data. Within the context of discrete signal processing, Fourier series replace Fourier transforms as the key mathematical tool. The Shannon sampling theorem provides the link between these two branches of Fourier analysis.

22.1 Discrete Signals

A discrete signal is a function $x = \{x(n)\}$ defined for all integers n. In signal processing, such discrete signals are often the result of *sampling* a function of a continuous variable. In our discussion of farfield propagation, we saw that the data gathered at each sensor effected a sampling of the Fourier transform, $F(\gamma)$, of the distant distribution $f(x)$. In the theoretical situation in which we had available an infinite discrete set of sensors, we would have an infinite sequence, obtained by sampling the function $F(\gamma)$. In many applications, the function that is being sampled is a function of time, say $f(t)$; we shall use this example in our discussion here.

In the most common case, that of equispaced sampling, we have $x(n) = f(n\Delta)$, where $\Delta > 0$ is the sampling interval. Generally, such discrete signals are neither a realistic model of the physical situation nor an accurate description of what we have actually obtained through measurement. Nevertheless, discrete signals provide the most convenient framework within

which to study the the basic tools of signal processing coming from Fourier analysis.

22.2 Notation

It is common practice to denote functions of a discrete variable by the letters x, y or z, as well as f, g or h. So we speak of the discrete signals $x = \{x(n) = 2n - 1, -\infty < n < \infty\}$ or $y = \{y(n) = -n^3 + n, -\infty < n < \infty\}$. For convenience, we often just say $x(n) = 2n - 1$ or $y(n) = n^3 + n$ when we mean the whole function x or y. However, if k is regarded as a fixed, but unspecified, integer, $x(k)$ means the value of the function x at k. This is really the same thing that we do in calculus, when we define a function $f(x) = ax^2 + bx + c$; the x is a variable, while the a, b, and c are parameters that do not change during the discussion of this function. Now n is a variable, while k is a parameter.

There are two special discrete signals with *reserved names*, δ and u: $\delta(0) = 1$ and $\delta(n) = 0$, for $n \neq 0$; $u(n) = 1$, for $n \geq 0$ and $u(n) = 0$ for $n < 0$. When we say that their names are reserved we mean that whenever you see these names you can (usually) assume that they refer to the same functions as just defined; in calculus e^x and $\sin x$ are reserved names, while in signal processing δ and u are reserved names.

22.3 Operations on Discrete Signals

Because discrete signals are functions, we can perform on them many of the operations we perform on functions of a continuous variable. For instance, we can add discrete signals x and y, to get the discrete signal $x + y$, we can multiply x by a real number c to get the discrete signal cx, we can multiply x and y to get xy, and so on. We can *shift* x to the right k units to get y with $y(n) = x(n - k)$. Notice that, if we shift $x = \delta$ to the right k units, we have y with $y(k) = 1$ and $y(n) = 0$ for $n \neq k$; we call this function δ_k, so we sometimes say that $\delta = \delta_0$.

In general, an operation, or, to use the official word, an *operator*, T works on a discrete signal x to produce another discrete signal y; we describe this situation by writing $y = T(x)$. For example, the operator $T = S_k$ shifts any x to the right by k units; for example, $S_3(\delta) = \delta_3$. We are particularly interested in operators that possess certain nice properties.

22.3.1 Linear Operators

An operator T is called *linear* if, for any x and z and numbers a and b we have $T(ax + bz) = aT(x) + bT(z)$; for example, the operator $T = S_k$ is linear.

Exercise 22.1 *Which of the following operators are linear?*

a. $T(x)(n) = x(n-1) + x(n)$;

b. $T(x)(n) = nx(n)$;

c. $T(x)(n) = x(n)^2$.

22.3.2 Shift-invariant Operators

Notice that operators are also functions, although not the sort that we usually study; their domains and ranges consist of functions. We have seen such operator-type functions in calculus class- the operator that transforms a function into its derivative is an operator-type function. Therefore we can combine operators using composition, in the same way we compose functions. The composition of operators T and S is the operator that first performs S and then performs T on the result; that is, the composition of T and S begins with x and ends with $y = T(S(x))$. Notice that, just as with ordinary functions, the order of the operators in the composition matters; $T(S(x))$ and $S(T(x))$ need not be the same discrete signal. We say that operators T and S *commute* if $T(S(x)) = S(T(x))$, for all x; in that case we write $TS = ST$.

An operator T is said to be *shift-invariant* if $TS_k = S_kT$ for all integers k. This means that if y is the output of the system described by T when the input is x, then when we shift the input by k, from x to S_kx, all that happens to the output is that the y is also shifted by k, from y to S_ky. For example, suppose that T is the squaring operator, defined by $T(x) = y$ with $y(n) = x(n)^2$. Then T is shift-invariant. On the other hand, the operator T with $y = T(x)$ such that $y(n) = x(-n)$ is not shift-invariant.

Exercise 22.2 *Which of the following operators are shift-invariant?*

a. $T(x)(n) = x(0) + x(n)$;

b. $T(x)(n) = x(n) + x(-n)$;

c. $T(x)(n) = \sum_{k=-2}^{2} x(n+k)$.

We are most interested in operators T that are both linear and shift-invariant; these are called LSI operators. An LSI operator T is often viewed as a linear system having inputs called x and outputs called y, where $y = T(x)$, and we speak of a LSI system.

22.3.3 Convolution Operators

Let h be a fixed discrete signal. For any discrete signal x define $y = T(x)$ by

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k),$$

for any integer n. We then say that y is the *convolution* of x with h and write $y = x * h$. Notice that $x * h = h * x$; that is,

$$\sum_{k=-\infty}^{\infty} h(k)x(n-k) = \sum_{k=-\infty}^{\infty} x(k)h(n-k).$$

The operator T is then the *convolution with* h operator. Any such T is linear.

22.3.4 LSI Filters are Convolutions

The operator T that is convolution with h is linear and shift-invariant. The most important fact in signal processing is that every T that is *linear and shift-invariant* (LSI) must be convolution with h, for some fixed discrete signal h.

Because of the importance of this result we give a proof now. First, we must find the h. To do this we let $x = \delta$; the h we seek is then the output $h = T(\delta)$. Now we must show that, for any other input x, we have $T(x) = x * h$. Note that for any k we have $\delta_k = S_k(\delta)$, so that

$$T(\delta_k) = T(S_k(\delta)) = S_k(T(\delta)) = S_k(h),$$

and so

$$T(\delta_k)(n) = S_k(h)(n) = h(n-k).$$

We can write an arbitrary x in terms of the δ_k as

$$x = \sum_{k=-\infty}^{\infty} x(k)\delta_k.$$

Then

$$T(x)(n) = T(\sum_{k=-\infty}^{\infty} x(k)\delta_k)(n) = \sum_{k=-\infty}^{\infty} x(k)T(\delta_k)(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k).$$

Therefore, $T(x) = x * h$, as we claimed. Because the h associated with the operator T is $h = T(\delta)$, the discrete signal h is called the *impulse-response function* of the system.

22.4 Special Types of Discrete Signals

Some of our calculations, such as convolution, involve infinite sums. In order for these sums to make sense we would need to impose certain restrictions on the discrete signals involved. Some books consider only discrete

signals x that are *absolutely summable*, that is, for which

$$\sum_{n=-\infty}^{\infty} |x(n)| < \infty,$$

or, at least, x that are *bounded*, which means that there is a positive constant $b > 0$ with $|x(n)| \leq b$ for all n. Sometimes the condition of absolute summability is imposed only on discrete functions h that are to be associated with LSI operators. Operators T whose h is absolutely summable have the desirable property of *stability*; that is, if the input function x is bounded, so is the output function $y = T(x)$. This property is also called the *bounded in, bounded out* (BIBO) property.

Exercise 22.3 *Show that the operator T is a stable operator if and only if its associated h is absolutely summable. Hint: If h is not absolutely summable, consider the input sequence with $x(n) = \overline{h(-n)}/|h(n)|$.*

In order to make use of the full power of Fourier methods some texts require that discrete signals x be *absolutely square-summable*, that is,

$$\sum_{n=-\infty}^{\infty} |x(n)|^2 < \infty.$$

Exercise 22.4 *Show that the discrete signal $x(n) = \frac{1}{|n|+1}$ is absolutely square-summable, but not absolutely summable.*

Our approach will be to avoid discussing specific requirements, with the understanding that some requirements will usually be needed to make the mathematics rigorous.

22.5 The Frequency-Response Function

Just as sine and cosine functions play important roles in calculus, so do their discrete counterparts in signal processing. The discrete sine function with frequency ω is the discrete signal $\sin_\omega$ with

$$\sin_\omega(n) = \sin(\omega n),$$

for each integer n. Similarly, the discrete cosine function with frequency ω is $\cos_\omega$ with

$$\cos_\omega(n) = \cos(\omega n).$$

It is convenient to include in the discussion the complex exponential e_ω defined by

$$e_\omega(n) = \cos_\omega(n) + i \sin_\omega(n) = e^{i\omega n}.$$

Since these discrete signals are the same for ω and $\omega + 2\pi$ we assume that ω lies in the interval $[-\pi, \pi)$.

22.5.1 The Response of a LSI System to $x = e_\omega$

Let T denote a LSI system and let ω be fixed. We show now that

$$T(e_\omega) = He_\omega,$$

for some constant H. Since the H can vary as we change ω it is really a function of ω, so we denote it $H = H(\omega)$.

Let $v = \{v(n)\}$ be the signal $v = e_\omega - S_1(e_\omega)$. Then we have

$$v(n) = e^{in\omega} - e^{i(n-1)\omega} = (1 - e^{-i\omega})e^{in\omega}.$$

Therefore, we can write

$$v = (1 - e^{-i\omega})e_\omega,$$

from which it follows that

$$T(v) = (1 - e^{-i\omega})T(e_\omega). \tag{22.1}$$

But we also have

$$T(v) = T(e_\omega - S_1(e_\omega)) = T(e_\omega) - TS_1(e_\omega),$$

and, since T is shift-invariant, $TS_1 = S_1T$, we know that

$$T(v) = T(e_\omega) - S_1T(e_\omega). \tag{22.2}$$

Combining Equations (22.1) and (22.2), we get

$$(1 - e^{-i\omega})T(e_\omega) = T(e_\omega) - S_1T(e_\omega).$$

Therefore,

$$S_1T(e_\omega) = e^{-i\omega}T(e_\omega),$$

or

$$T(e_\omega)(n-1) = S_1T(e_\omega)(n) = e^{-i\omega}T(e_\omega)(n).$$

We conclude from this that

$$e^{in\omega}T(e_\omega)(0) = T(e_\omega)(n),$$

for all n. Finally, we let $H(\omega) = T(e_\omega)(0)$.

It is useful to note that we did not use here the fact that T is a convolution operator. However, since we do know that $T(x) = x * h$, for $h = T(\delta)$, we can relate the function $H(\omega)$ to h.

22.5.2 Relating $H(\omega)$ to $h = T(\delta)$

Since T is a LSI operator, T operates by convolving with $h = T(\delta)$. Consider what happens when we select for the input the discrete signal $x = e_\omega$. Then the output is $y = T(e_\omega)$ with

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)e^{i\omega(n-k)} = H(e^{i\omega})e^{i\omega n},$$

where

$$H(e^{i\omega}) = \sum_{k=-\infty}^{\infty} h(k)e^{-i\omega k} \tag{22.3}$$

is the value, at ω, of the *frequency-response function* of T. The point here is that when the input is $x = e_\omega$ the output is a multiple of e_ω, the multiplier being the (possibly complex) number $H(e^{i\omega})$. Linear, shift-invariant systems T do not alter the frequency of the input, but just change its amplitude and/or phase. The constant $H(e^{i\omega})$ is the same as $H(\omega)$ obtained earlier; having two different notations for the same function is an unfortunate, but common, occurrence in the signal-processing literature.

It is important to note that the infinite sum in Equation (22.3) need not converge for arbitrary $h = \{h(k)\}$. It does converge, obviously, whenever h is finitely nonzero; it will also converge for infinitely nonzero sequences that are suitably restricted.

A common problem in signal processing is to design a LSI filter with a desired frequency-response function $H(e^{i\omega})$. To determine $h(m)$, given $H(e^{i\omega})$, we multiply both sides of Equation (22.3) by $e^{i\omega m}$, multiply by $\frac{1}{2\pi}$, integrate over the interval $[-\pi, \pi]$, and use the helpful fact that

$$\int_{-\pi}^{\pi} e^{i(m-k)\omega} d\omega = 0,$$

for $m \neq k$. The result is

$$h(m) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(e^{i\omega})e^{i\omega m} d\omega. \tag{22.4}$$

It is useful to extend the definition of $H(e^{i\omega})$ to permit $e^{i\omega}$ to be replaced by any complex number z. Then we get the *z-transform* of h, given by

$$H(z) = \sum_{k=-\infty}^{\infty} h(k)z^{-k}.$$

We can study the working of the system T on more general inputs x by representing x as a sum of complex-exponential discrete signals e_ω.

The representation, in Equation (22.4), of the infinite sequence $h = \{h(k)\}$ as a superposition of complex-exponential discrete signals suggests the possibility that such a representation is available for general infinite discrete signals, a notion we take up in the next section.

22.6 The Discrete Fourier Transform

A common theme running through mathematics is the representation of complicated objects in terms of simpler ones. Taylor-series expansion enables us to view quite general functions as infinite versions of polynomials by representing them as infinite sums of the power functions. Fourier-series expansions give representations of quite general functions as infinite sums of sines and cosines. Here we obtain similar representation for discrete signals, as infinite sums of the complex exponentials, e_ω, for ω in $[-\pi, \pi)$.

Our goal is to represent a general discrete signal x as a sum of the e_ω, for ω in the interval $[-\pi, \pi)$. Such a sum is, in general, an integral over ω. So we seek to represent x as

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) e^{i\omega n} d\omega, \tag{22.5}$$

where $X(\omega)$ is a function to be determined. As we shall see, the function we seek is the *discrete Fourier transform* (DFT) of x, defined by

$$X(\omega) = \sum_{m=-\infty}^{\infty} x(m) e^{-i\omega m}. \tag{22.6}$$

This follows from the discussion leading up to Equation (22.4). Notice that in the case $x = h$ the function $H(\omega)$ is the same as the frequency-response function $H(e^{i\omega})$ defined earlier. For this reason the notation $X(\omega)$ and $X(e^{i\omega})$ are used interchangably. The DFT of the discrete signal x is sometimes called the *discrete-time Fourier transform* (DTFT).

The sum in Equation (22.6) is the *Fourier-series expansion* for the function $X(\omega)$, over the interval $[-\pi, \pi)$; the $x(n)$ are its *Fourier coefficients*.

The infinite series in Equation (22.4) that is used to define $X(\omega)$ may not converge. For example, suppose that x is an exponential signal, with $x(n) = e^{i\omega_0 n}$. Then the infinite sum would be

$$\sum_{m=-\infty}^{\infty} e^{i(\omega_0 - \omega)m},$$

which obviously does not converge, at least in any ordinary sense. Consider, though, what happens when we put this sum inside an integral and reverse

the order of integration and summation. Specifically, consider

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} F(\omega)\sum_{m=-\infty}^{\infty} e^{i(\omega_0-\omega)m}d\omega,$$

$$= \sum_{m=-\infty}^{\infty}(\frac{1}{2\pi}\int_{-\pi}^{\pi} F(\omega)e^{i(\omega_0-\omega)m}d\omega),$$

$$= \sum_{m=-\infty}^{\infty} e^{i\omega_0 m}f(m) = F(\omega_0).$$

So, the infinite sum acts like the Dirac delta $\delta(\omega-\omega_0)$. This motivates the following definition of the infinite sum:

$$\sum_{m=-\infty}^{\infty} e^{i(\omega_0-\omega)m} = \delta(\omega-\omega_0). \tag{22.7}$$

A different approach to the infinite sum is to consider

$$\lim_{N\to+\infty}\frac{1}{2N+1}\sum_{m=-N}^{N} e^{i(\omega_0-\omega)m}.$$

According to Equation (5.4), we have

$$\sum_{n=-N}^{N} e^{i\omega n} = \frac{\sin(\omega(N+\frac{1}{2}))}{\sin(\frac{\omega}{2})}.$$

Therefore,

$$\lim_{N\to+\infty}\frac{1}{2N+1}\sum_{m=-N}^{N} e^{i(\omega_0-\omega)m} = 1, \tag{22.8}$$

for $\omega=\omega_0$, and zero, otherwise.

22.7 The Convolution Theorem

Once again, let $y=T(x)$, where T is a LSI operator with associated filter $h=\{h(k)\}$. Because we can write

$$x(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)e_\omega(n)d\omega,$$

or, in shorthand, leaving out the n, as

$$x = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)e_\omega d\omega,$$

we have

$$y = T(x) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)T(e_\omega)d\omega,$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)H(\omega)e_\omega d\omega,$$

or

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)H(\omega)e_\omega(n)d\omega.$$

But we also have

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} Y(\omega)e_\omega(n)d\omega,$$

from which we conclude that

$$Y(\omega) = X(\omega)H(\omega), \tag{22.9}$$

for each ω in $[-\pi, \pi)$.

Equation (22.9) is the most important equation in signal processing. It describes the activity of an LSI system by telling us that the system simply multiplies the DFT of the input x by the DFT of the h, the frequency-response function of the system, to produce the DFT of the output y. Since $y = x * h$ it also tells us that whenever y is formed by convolving two discrete signals x and h, its DFT is the product of the DFT of x and the DFT of h.

22.8 Sampling and Aliasing

The term *sampling* refers to the transition from a function $f(t)$ of a continuous variable to a discrete signal x, defined by $x(n) = f(n\Delta)$, where $\Delta > 0$ is the *sample spacing*. For example, suppose that $f(t) = \sin(\gamma t)$ for some frequency $\gamma > 0$. Then $x(n) = \sin(\gamma n\Delta) = \sin(\omega n)$, where $\omega = \gamma\Delta$. We define $X(\omega)$, the DFT of the discrete signal x, for $|\omega| \leq \pi$, so we need $|\gamma|\Delta \leq \pi$. This means we must select Δ so that $\Delta \leq \pi/|\gamma|$. In general, if the function $f(t)$ has sinusoidal components with frequencies γ such that $|\gamma| \leq \Gamma$ then we should select $\Delta \leq \pi/\Gamma$.

If we select Δ too large, then a frequency component of $f(t)$ corresponding to $|\gamma| > \pi/\Delta$ will be mistaken for a frequency with smaller magnitude. This is *aliasing*. For example, if $f(t) = \sin(3t)$, but $\Delta = \pi/2$, then the frequency $\gamma = 3$ is mistaken for the frequency $\gamma = -1$, which lies in $[-2, 2]$. When we sample we get

$$x(n) = \sin(3\Delta n) = \sin(-\Delta n + 4\Delta n) = \sin(-\Delta n + 2\pi n) = \sin(-\Delta n),$$

for each n.

Chapter 23

Appendix: Randomness in Signal Processing

We treat noise in our data using the probabilistic concept of *random variable*. The term is not self-explanatory, so we begin by explaining what a random variable is.

23.1 Random Variables as Models

When we use mathematical tools, such as differential equations, probability, or systems of linear equations, to describe a real-world situation, we say that we are employing a *mathematical model*. Such models must be sufficiently sophisticated to capture the essential features of the situation, while remaining computationally manageable. In this chapter we are interested in one particular type of mathematical model, the *random variable*.

Imagine that you are holding a baseball four feet off the ground. If you drop it, it will land on the ground directly below where you held it. The height of the ball at any time during the fall is described by the function $h(t)$ satisfying the ordinary differential equation $h''(t) = -32\frac{\text{ft}}{\text{sec}^2}$. Solving this differential equation with the initial conditions $h(0) = 4$ ft , $h'(0) = 0\frac{\text{ft}}{\text{sec}}$, we find that $h(t) = 4 - 16t^2$. Solving $h(T) = 0$ for T we find the elapsed time T until impact is $T = 0.5$ sec.. The velocity of the ball at impact is $h'(T) = -32T = -16\frac{\text{ft}}{\text{sec}}$.

Now imagine that, instead of a baseball, you are holding a feather. The feather and the baseball are both subject to the same laws of gravity, but now other aspects of the situation, which we could safely ignore in the case of the baseball, become important in the case of the feather. Like the baseball, the feather is subjected to air resistance and to whatever fluctuations in air currents may be present during its fall. Unlike the baseball, however,

the effects of the air matter to the flight of the feather; in fact, they become the dominant factors. When we designed our differential-equation model for the falling baseball we performed no experiments to help us understand its behavior. We simply ignored all other aspects of the situation, and included only gravity in our mathematical model. Even the modeling of gravity was slightly simplified, in that we assumed a constant gravitational acceleration, even though Newton's Laws tell us that it increases as we approach the center of the earth. When we drop the ball and find that our model is accurate we feel no need to change it. When we drop the feather we discover immediately that a new model is needed; but what?

The first thing we observe is that the feather falls in a manner that is impossible to predict with accuracy. Dropping it once again, we notice that it behaves differently this time, landing in a different place and, perhaps, taking longer to land. How are we to model such a situation, in which repeated experiments produce different results? Can we say nothing useful about what will happen when we drop the feather the next time?

As we continue to drop the feather, we notice that, while the feather usually does not fall directly beneath the point of release, it does not fall too far away. Suppose we draw a grid of horizontal and vertical lines on the ground, dividing the ground into a pattern of squares of equal area. Now we repeatedly drop the feather and record the proportion of times the feather is (mainly) contained within each square; we also record the elapsed time. As we are about to drop the feather the next time, we may well assume that the outcome will be consistent with the behavior we have observed during the previous drops. While we cannot say for certain where the feather will fall, nor what the elapsed time will be, we feel comfortable making a *probabilistic statement* about the likelihood that the feather will land in any given square and about the elapsed time.

The squares into which the feather may land are finite, or, if we insist on creating an infinite grid, discretely infinite, while the elapsed time can be any positive real number. Let us number the squares as $n = 1, 2, 3, ...$ and let p_n be the proportion of drops that resulted in the feather landing mainly in square n. Then $p_n \geq 0$ and $\sum_{n=1}^{\infty} p_n = 1$. The sequence $p = \{p_n | n = 1, 2, ...\}$ is then a *discrete probability sequence* (dps), or a *probability sequence*, or a *discrete probability*. Now let N be the number of the square that will contain the feather on the next drop. All we can say about N is that, according to our model, the probability that N will equal n is p_n. We call N a *discrete random variable* with *probability sequence* p.

It is difficult to be more precise about what probability really means. When we say that the probability is p_n that the feather will land in square n on the next drop, where does that probability reside? Do we believe that the numbers p_n are *in the feather* somehow? Do these numbers simply describe our own ignorance, so are *in our heads*? Are they a combination of the two, in our heads as a result of our having experienced what the

feather did previously? Perhaps it is best simply to view probablity as a type of mathematical model that we choose to adopt in certain situations.

Now let T be the elapsed time for the next feather to hit the ground. What can we say about T? Based on our prior experience, we are willing to say that, for any interval $[a, b]$ within $(0, \infty)$, the probability that T will take on a value within $[a, b]$ is the proportion of prior drops in which the elapsed time was between a and b. Then T is a *continuous random variable*, in that the values it may take on (in theory, at least) lie in a continuum. To help us calculate the probabilities associated with T we use our prior experience to specify a function $f_T(t)$, called the *probability density function* (pdf) of T, having the property that the probability that T will lie between a and b can be calculated as $\int_a^b f_T(t)dt$. Such $f_T(t)$ will have the properties $f_T(t) \geq 0$ for all positive t and $\int_0^\infty f_T(t)dt = 1$.

In the case of the falling feather we had to perform experiments to determine appropriate ps p and pdf $f_T(t)$. In practice, we often describe our random variables using a ps or pdf from a well-studied parametric family of such mathematical objects. Popular examples of such ps and pdf, such as Poisson probabilities and Gaussian pdf, are discussed early in most courses in probability theory.

It is simplest to discuss the main points of random signal processing within the context of discrete signals, so we return there now.

23.2 Discrete Random Signal Processing

Previously, we have encountered specific discrete functions, such as δ_k, u, e_ω, whose values at each integer n are given by an exact formula. In signal processing we must also concern ourselves with discrete functions whose values are not given by such formulas, but rather, seem to obey only probabilistic laws. We shall need such discrete functions to model noise. For example, imagine that, at each time n, a fair coin is tossed and $x(n) = 1$ if the coin shows heads, $x(n) = -1$ if the coin shows tails. We cannot determine the value of $x(n)$ from any formula; we must simply toss the coins. Given any discrete function x with values $x(n)$ that are either 1 or -1, we cannot say if x was generated by such a coin-flipping manner. In fact, any such x could have been the result of coin flips. All we can say is how likely it is that a particular x was so generated. For example, if $x(n) = 1$ for n even and $x(n) = -1$ for n odd, we feel, intuitively, that it is highly unlikely that such an x came from random coin tossing. What bothers us, of course, is that the values $x(n)$ seem so predictable; randomness seems to require some degree of unpredictability. If we were given two such sequences, the first being the one described above, with 1 and -1 alternating, and the second exhibiting no obvious pattern, and asked to select the one generated by independent random coin tossing, we

would clearly choose the second one. There is a subtle point here, however. When we say that we are "given an infinite sequence" what do we really mean? Because the issue here is not the infinite nature of the sequences, let us reformulate the discussion in terms of finite vectors of length, say, 100, with entries 1 or -1. If we are shown a print-out of two such vectors, the first with alternating 1 and -1, and the second vector exhibiting no obvious pattern, we would immediately say that it was the second one that was generated by the coin-flipping procedure, even though the two vectors are equally likely to have been so generated. The point is that we associate randomness with the absence of a pattern, more than with probability. When there is a pattern, the vector can be described in a way that is significantly shorter than simply listing its entries. Indeed, it has been suggested that a vector is random if it cannot be described in a manner shorter than simply listing its members.

23.2.1 The Simplest Random Sequence

We say that a sequence $x = \{x(n)\}$ is a *random sequence* or a *discrete random process* if $x(n)$ is a random variable for each integer n. A simple, yet remarkably useful, example is the random-coin-flip sequence, which we shall denote by $c = \{c(n)\}$. In this model a coin is flipped for each n and $c(n) = 1$ if the coin comes up heads, with $c(n) = -1$ if the coin comes up tails. It will be convenient to allow for the coin to be *biased*, that is, for the probabilities of heads and tails to be unequal. We denote by p the probability that heads occurs and $1 - p$ the probability of tails; the coin is called *unbiased* or *fair* if $p = 1/2$. To find the *expected value* of $c(n)$, written $E(c(n))$, we multiply each possible value of $c(n)$ by its probability and sum; that is,

$$E(c(n)) = (+1)p + (-1)(1 - p) = 2p - 1.$$

If the coin is fair then $E(c(n)) = 0$. The variance of the random variable $c(n)$, measuring its tendency to deviate from its expected value, is $var(c(n)) = E([c(n) - E(c(n))]^2)$. We have

$$var(c(n)) = [+1 - (2p - 1)]^2 p + [-1 - (2p - 1)]^2 (1 - p) = 4p - 4p^2.$$

If the coin is fair then $var(c(n)) = 1$. It is important to note that we do not change the coin at any time during the generation of the random sequence c; in particular, the p does not depend on n.

The random-coin-flip sequence c is the simplest example of a discrete random process or a random discrete function. It is important to remember that a random discrete function is not any one particular discrete function, but rather a probabilistic model chosen to allow us to talk about the probabilities associated with the values of the $x(n)$. In the next section we

shall use this discrete random process to generate a wide class of discrete random processes, obtained by viewing $c = c(n)$ as the input into a linear, shift-invariant (LSI) filter.

23.3 Random Discrete Functions or Discrete Random Processes

A linear, shift-invariant (LSI) operator T with impulse response function $h = \{h(k)\}$ operates on any input sequence $x = \{x(n)\}$ to produce the output sequence $y = \{y(n)\}$ according to the convolution formula

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) = \sum_{k=-\infty}^{\infty} x(k)h(n-k). \tag{23.1}$$

We learn more about the system that T represents when we select as input sinusoids at fixed frequencies. Let ω be a fixed frequency in the interval $[-\pi, \pi)$ and let $x = e_\omega$, so that $x(n) = e^{in\omega}$ for each integer n. Then Equation (23.1) shows us that the output is

$$y(n) = H(e^{i\omega})x(n),$$

where

$$H(e^{i\omega}) = \sum_{k=-\infty}^{\infty} h(k)e^{-ik\omega}. \tag{23.2}$$

This function of ω is called the *frequency-response function* of the system. We can learn even more about the system by selecting as input the sequence $x(n) = z^n$, where z is an arbitrary complex number. Then Equation (23.1) gives the output as

$$y(n) = H(z)x(n),$$

where

$$H(z) = \sum_{k=-\infty}^{\infty} h(k)z^{-k}. \tag{23.3}$$

Note that if we select $z = e^{i\omega}$ then $H(z) = H(e^{i\omega})$ as given by Equation (23.2). The function $H(z)$ of the complex variable z is the z-transform of the sequence h and also the *transfer function* of the system determined by h.

Now we take this approach one step further. Let us select as our input $x = \{x(n)\}$ the random-coin-flip sequence $c = \{c(n)\}$, with $p = 0.5$. It is important to note that such an x is not one specific discrete function,

but a random model for such functions. The output $y = \{y(n)\}$ is again a random sequence, with

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)c(n-k). \tag{23.4}$$

Clearly, in order for the infinite sum to converge we would need to place restrictions on the sequence h; if $h(k)$ is zero except for finitely many values of k then we have no problem. We shall put off discussion of convergence issues and focus on statistical properties of the output random sequence y.

Let u and v be (possibly complex-valued) random variables with expected values $E(u)$ and $E(v)$, respectively. The covariance between u and v is defined to be

$$cov(u,v) = E([u - E(u))\overline{(v - E(v))}]),$$

and the cross-correlation between u and v is

$$corr(u,v) = E(u\overline{v}).$$

It is easily shown that $cov(u,v) = corr(u,v) - E(u)\overline{E(v)}$. When $u = v$ we get $cov(u,u) = var(u)$ and $corr(u,u) = E(|u|^2)$. If $E(u) = E(v) = 0$ then $cov(u,v) = corr(u,v)$.

To illustrate, let $u = c(n)$ and $v = c(n-m)$. Then, since the coin is fair, $E(c(n)) = E(c(n-m)) = 0$ and

$$cov(c(n), c(n-m)) = corr(c(n), c(n-m)) = E(c(n)\overline{c(n-m)}).$$

Because the $c(n)$ are independent, $E(c(n)\overline{c(n-m)}) = 0$ for m not equal to 0, and $E(|c(n)|^2) = var(c(n)) = 1$. Therefore

$$cov(c(n), c(n-m)) = corr(c(n), c(n-m)) = 0, \text{for } m \neq 0,$$

and

$$cov(c(n), c(n)) = corr(c(n), c(n)) = 1.$$

Returning now to the output sequence $y = \{y(n)\}$ we compute the correlation $corr(y(n), y(n-m)) = E(y(n)\overline{y(n-m)})$. Using the convolution formula Equation (23.4) we find that

$$corr(y(n), y(n-m)) = \sum_{k=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} h(k)\overline{h(j)} corr(c(n-k), c(n-m-j)).$$

Since

$$corr(c(n-k), c(n-m-j)) = 0, \text{for } k \neq m+j,$$

we have

$$corr(y(n), y(n-m)) = \sum_{k=-\infty}^{\infty} h(k)\overline{h(k-m)}. \tag{23.5}$$

The expression of the right side of Equation (23.5) is the definition of the *autocorrelaton* of the sequence h, denoted $\rho_h(m)$; that is,

$$\rho_h(m) = \sum_{k=-\infty}^{\infty} h(k)\overline{h(k-m)}. \tag{23.6}$$

It is important to note that the expected value of $y(n)$ is

$$E(y(n)) = \sum_{k=-\infty}^{\infty} h(k)E(c(n-k)) = 0$$

and the correlation $corr(y(n), y(n-m))$ depends only on m; neither quantity depends on n and the sequence y is therefore called *weak-sense stationary*. Let's consider an example.

Take $h(0) = h(1) = 0.5$ and $h(k) = 0$ otherwise. Then the system is the two-point moving-average, with

$$y(n) = 0.5x(n) + 0.5x(n-1).$$

With $x(n) = c(n)$ we have

$$y(n) = 0.5c(n) + 0.5c(n-1).$$

In the case of the random-coin-flip sequence c each $c(n)$ is unrelated to any other $c(m)$; the coin flips are independent. This is no longer the case for the $y(n)$; one effect of the filter h is to introduce correlation into the output. To illustrate, since $y(0)$ and $y(1)$ both depend, to some degree, on the value $c(0)$, they are related. Using Equation (23.6) we have

$$\rho_h(0) = h(0)h(0) + h(1)h(1) = 0.25 + 0.25 = 0.5,$$

$$\rho_h(-1) = h(0)h(1) = 0.25,$$

$$\rho_h(+1) = h(1)h(0) = 0.25,$$

and

$$\rho_h(m) = 0, \text{otherwise}.$$

So we see that $y(n)$ and $y(n-m)$ are related, for $m = -1, 0, +1$, but not otherwise.

23.4 Correlation Functions and Power Spectra

As we have seen, any nonrandom sequence $h = \{h(k)\}$ has its autocorrelation function defined, for each integer m, by

$$\rho_h(m) = \sum_{k=-\infty}^{\infty} h(k)\overline{h(k-m)}.$$

For a random sequence $y(n)$ that is wide-sense stationary, its correlation function is defined to be

$$\rho_y(m) = E(y(n)\overline{y(n-m)}).$$

The *power spectrum* of h is defined for ω in $[-\pi, \pi]$ by

$$S_h(\omega) = \sum_{m=-\infty}^{\infty} \rho_h(m)e^{-im\omega}.$$

It is easy to see that

$$S_h(\omega) = |H(e^{i\omega})|^2,$$

so that $S_h(\omega) \geq 0$. The power spectrum of the random sequence $y = \{y(n)\}$ is defined as

$$S_y(\omega) = \sum_{m=-\infty}^{\infty} \rho_y(m)e^{-im\omega}.$$

Although it is not immediately obvious, we also have $S_y(\omega) \geq 0$. One way to see this is to consider

$$Y(e^{i\omega}) = \sum_{n=-\infty}^{\infty} y(n)e^{-in\omega}$$

and to calculate

$$E(|Y(e^{i\omega})|^2) = \sum_{m=-\infty}^{\infty} E(y(n)\overline{y(n-m)})e^{-im\omega} = S_y(\omega).$$

Given any power spectrum $S_y(\omega)$ we can construct $H(e^{i\omega})$ by selecting an arbitrary phase angle θ and letting

$$H(e^{i\omega}) = \sqrt{S_y(\omega)}e^{i\theta}.$$

We then obtain the nonrandom sequence h associated with $H(e^{i\omega})$ using

$$h(n) = \int_{-\pi}^{\pi} H(e^{i\omega})e^{in\omega}d\omega/2\pi.$$

It follows that $\rho_h(m) = \rho_y(m)$ for each m and $S_h(\omega) = S_y(\omega)$ for each ω.

What we have discovered is that, when the input to the system is the random-coin-flip sequence c, the output sequence y has a correlation function $\rho_y(m)$ that is equal to the autocorrelation of the sequence h. As we just saw, for any weak-sense stationary random sequence y with expected value $E(y(n))$ constant and correlation function $corr(y(n), y(n-m))$ independent of n, there is a LSI system h with $\rho_h(m) = \rho_y(m)$ for each m. Therefore, any weak-sense stationary random sequence y can be viewed as the output of an LSI system, when the input is the random-coin-flip sequence $c = \{c(n)\}$.

23.5 Random Sinusoidal Sequences

If $A = |A|e^{i\theta}$, with amplitude $|A|$ a positive-valued random variable and phase angle θ a random variable taking values in the interval $[-\pi, \pi]$ then A is a complex-valued random variable. For a fixed frequency ω_0 we define a random sinusoidal sequence $s = \{s(n)\}$ by $s(n) = Ae^{in\omega_0}$. We assume that θ has the uniform distribution over $[-\pi, \pi]$ so that the expected value of $s(n)$ is zero. The correlation function for s is

$$\rho_s(m) = E(s(n)\overline{s(n-m)}) = E(|A|^2)e^{im\omega_0}$$

and the power spectrum of s is

$$S_s(\omega) = E(|A|^2) \sum_{m=-\infty}^{\infty} e^{im(\omega_0 - \omega)},$$

so that, by Equation (22.7), we have

$$S_s(\omega) = E(|A|^2)\delta(\omega - \omega_0).$$

We generalize this example to the case of multiple independent sinusoids. Suppose that, for $j = 1, ..., J$, we have fixed frequencies ω_j and independent complex-valued random variables A_j. We let our random sequence be defined by

$$s(n) = \sum_{j=1}^{J} A_j e^{in\omega_j}.$$

Then the correlation function for x is

$$\rho_s(m) = \sum_{j=1}^{J} E(|A_j|^2)e^{im\omega_j}$$

and the power spectrum for s is

$$S_s(\omega) = \sum_{j=1}^{J} E(|A_j|^2)\delta(\omega - \omega_j).$$

A commonly used model in signal processing is that of independent sinusoids in additive noise.

Let $q = \{q(n)\}$ be an arbitrary weak-sense stationary discrete random sequence, with correlation function $\rho_q(m)$ and power spectrum $S_q(\omega)$. We say that q is white noise if $\rho_q(m) = 0$ for m not equal to zero, or, equivalently, if the power spectrum $S_q(\omega)$ is constant over the interval $[-\pi, \pi]$. The *independent sinusoids in additive noise* model is a random sequence of the form

$$x(n) = \sum_{j=1}^{J} A_j e^{in\omega_j} + q(n).$$

The *signal power* is defined to be $\rho_s(0)$, which is the sum of the $E(|A_j|^2)$, while the noise power is $\rho_q(0)$. The *signal-to-noise ratio* (SNR) is the ratio of signal power to noise power.

It is often the case that the SNR is quite low and it is desirable to process the x to enhance this ratio. The data we have is typically finitely many values of $x(n)$, say for $n = 1, 2, ..., N$. One way to process the data is to estimate $\rho_x(m)$ for some small number of integers m around zero, using, for example, the *lag products* estimate

$$\hat{\rho}_x(m) = \frac{1}{N-m} \sum_{n=1}^{N-m} x(n)\overline{x(n-m)},$$

for $m = 0, 1, ..., M < N$ and $\hat{\rho}_x(-m) = \overline{\hat{\rho}_x(m)}$. Because $\rho_q(m) = 0$ for m not equal to zero, we will have $\hat{\rho}_x(m)$ approximating $\rho_s(m)$ for nonzero values of m, thereby reducing the effect of the noise.

The additive noise is said to be *correlated* or *non-white* if it is not the case that $\rho_x(m) = 0$ for all nonzero m. In this case the noise power spectrum is not constant, and so may be concentrated in certain regions of the interval $[-\pi, \pi]$.

23.6 Spread-Spectrum Communication

In this section we return to the random-coin-flip model, this time allowing the coin to be biased, that is, p need not be 0.5. Let $s = \{s(n)\}$ be a random sequence, such as $s(n) = Ae^{in\omega_0}$, with $E(s(n)) = \mu$ and correlation function $\rho_s(m)$. Define a second random sequence x by

$$x(n) = s(n)c(n).$$

The random sequence x is generated from the random signal s by randomly changing its signs. We can show that

$$E(x(n)) = \mu(2p-1)$$

and, for m not equal to zero,

$$\rho_x(m) = \rho_s(m)(2p-1)^2,$$

with $\rho_x(0) = \rho_s(0) + 4p(1-p)\mu^2$. Therefore, if $p = 1$ or $p = 0$ we get $\rho_x(m) = \rho_s(m)$ for all m, but for $p = 0.5$ we get $\rho_x(m) = 0$ for m not equal to zero. If the coin is unbiased, then the random sign changes convert the original signal s into white noise. Generally, we have

$$S_x(\omega) = (2p-1)^2 S_s(\omega) + (1-(2p-1)^2)(\mu^2 + \rho_s(0)),$$

which says that the power spectrum of x is a combination of the signal power spectrum and a white-noise power spectrum, approaching the white-noise power spectrum as p approaches 0.5. If the original signal power spectrum is concentrated within a small interval, then the effect of the random sign changes is to spread that spectrum. Once we know what the sequence c is we can recapture the original signal from $s(n) = x(n)c(n)$. The use of such a spread spectrum permits the sending of multiple narrow-band signals, without confusion, as well as protecting against any narrow-band additive interference.

23.7 Stochastic Difference Equations

The ordinary first-order differential equation $y'(t) + ay(t) = f(t)$, with initial condition $y(0) = 0$, has for its solution $y(t) = e^{-at}\int_0^t e^{as} f(s)ds$. One way to look at such differential equations is to consider $f(t)$ to be the input to a system having $y(t)$ as its output. The system determines which terms will occur on the left side of the differential equation. In many applications the input $f(t)$ is viewed as random noise and the output is then a continuous-time random process. Here we want to consider the discrete analog of such differential equations.

We replace the first derivative with the first difference, $y(n+1) - y(n)$ and we replace the input with the random-coin-flip sequence $c = \{c(n)\}$, to obtain the random difference equation

$$y(n+1) - y(n) + ay(n) = c(n). \tag{23.7}$$

With $b = 1 - a$ and $0 < b < 1$ we have

$$y(n+1) - by(n) = c(n). \tag{23.8}$$

The solution is $y = \{y(n)\}$ given by

$$y(n) = b^n \sum_{k=-\infty}^{n} b^{-k} c(k). \tag{23.9}$$

Comparing this with the solution of the differential equation, we see that the term b^n plays the role of $e^{-at} = (e^{-a})^t$, so that $b = 1 - a$ is substituting for e^{-a}. The infinite sum replaces the infinite integral, with $b^{-k}c(k)$ replacing the integrand $e^{as}f(s)$.

The solution sequence y given by Equation (23.9) is a weak-sense stationary random sequence and its correlation function is

$$\rho_y(m) = b^m/(1 - b^2).$$

Since

$$b^n \sum_{k=-\infty}^{n} b^{-k} = 1 - b$$

the random sequence $(1 - b)^{-1}y(n)$ is an infinite *moving-average* random sequence formed from the random sequence c.

We can derive the solution in Equation (23.9) using z-transforms. The expression $y(n) - by(n - 1)$ can be viewed as the output of a LSI system with $h(0) = 1$ and $h(1) = -b$. Then $H(z) = 1 - bz^{-1} = (z - b)/z$ and the inverse $H(z)^{-1} = z/(z - b)$ describes the inverse system. Since

$$H(z)^{-1} = z/(z - b) = 1/(1 - bz^{-1}) = 1 + bz^{-1} + b^2z^{-2} + ...$$

the inverse system applied to input $c = \{c(n)\}$ is

$$y(n) = c(n) + bc(n - 1) + b^2c(n - 2) + ... = b^n \sum_{k=-\infty}^{n} b^{-k} c(k).$$

23.8 Random Vectors and Correlation Matrices

In estimation and detection theory, the task is to distinguish *signal vectors* from *noise vectors*. In order to perform such a task, we need to know how signal vectors differ from noise vectors. Most frequently, what we have is statistical information. The signal vectors of interest, which we denote by $s = (s_1, ..., s_N)^T$, typically exhibit some patterns of behavior among their entries. For example, a constant signal, such as $s = (1, 1, ..., 1)^T$, has all its entries identical. A sinusoidal signal, such as $s = (1, -1, 1, -1, ..., 1, -1)^T$, exhibits a periodicity in its entries. If the signal is a vectorization of a two-dimensional image, then the patterns will be more difficult to describe, but

will be there, nevertheless. In contrast, a typical noise vector, denoted $q = (q_1, ..., q_N)^T$, may have entries that are unrelated to each other, as in white noise. Of course, what is signal and what is noise depends on the context; unwanted interference in radio may be viewed as noise, even though it may be a weather report or a song.

To deal with these notions mathematically, we adopt statistical models. The entries of s and q are taken to be random variables, so that s and q are random vectors. Often we assume that the mean values, $E(s)$ and $E(q)$, are zero. Then patterns that may exist among the entries of these vectors are described in terms of *correlations*. The *noise covariance matrix*, which we denote by Q, has for its entries $Q_{mn} = E((q_m - E(q_m))\overline{(q_n - E(q_n))})$, for $m, n = 1, ..., N$. The signal covariance matrix is defined similarly. If $E(q_n) = 0$ and $E(|q_n|^2) = 1$ for each n, then Q is the *noise correlation matrix*. Such matrices Q are Hermitian and non-negative definite, that is, $x^\dagger Q x$ is non-negative, for every vector x. If Q is a positive multiple of the identity matrix, then the noise is said to be *white noise*.

Chapter 24

Appendix: Detection and Classification

In some applications of remote sensing, our goal is simply to see what is "out there"; in sonar mapping of the sea floor, the data are the acoustic signals as reflected from the bottom, from which the changes in depth can be inferred. Such problems are *estimation* problems.

In other applications, such as sonar target detection or medical diagnostic imaging, we are looking for certain things, evidence of a surface vessel or submarine, in the sonar case, or a tumor or other abnormality in the medical case. These are *detection* problems. In the sonar case, the data may be used directly in the detection task, or may be processed in some way, perhaps frequency-filtered, prior to being used for detection. In the medical case, or in synthetic-aperture radar (SAR), the data is usually used to construct an image, which is then used for the detection task. In estimation, the goal can be to determine how much of something is present; detection is then a special case, in which we want to decide if the amount present is zero or not.

The detection problem is also a special case of *discrimination*, in which the goal is to decide which of two possibilities is true; in detection the possibilities are simply the presence or absence of the sought-for signal.

More generally, in *classification* or *identification*, the objective is to decide, on the basis of measured data, which of several possibilities is true.

24.1 Estimation

We consider only estimates that are linear in the data, that is, estimates of the form

$$\hat{\gamma} = b^\dagger x = \sum_{n=1}^{N} \overline{b_n} x_n, \tag{24.1}$$

where $x = (x_1, ..., x_N)^T$ is the vector of data and $b^\dagger$ denotes the conjugate transpose of the vector $b = (b_1, ..., b_N)^T$. The vector b that we use will be the *best linear unbiased estimator* (BLUE) [57] for the particular estimation problem.

24.1.1 The simplest case: a constant in noise

We begin with the simplest case, estimating the value of a constant, given several instances of the constant in additive noise. Our data are $x_n = \gamma + q_n$, for $n = 1, ..., N$, where γ is the constant to be estimated, and the q_n are noises. For convenience, we write

$$x = \gamma u + q, \tag{24.2}$$

where $x = (x_1, ..., x_N)^T$, $q = (q_1, ..., q_N)^T$, $u = (1, ..., 1)^T$, the expected value of the random vector q is $E(q) = 0$, and the covariance matrix of q is $E(qq^T) = Q$. The BLUE employs the vector

$$b = \frac{1}{u^\dagger Q^{-1} u} Q^{-1} u. \tag{24.3}$$

The BLUE estimate of γ is

$$\hat{\gamma} = \frac{1}{u^\dagger Q^{-1} u} u^\dagger Q^{-1} x. \tag{24.4}$$

If $Q = \sigma^2 I$, for some $\sigma > 0$, with I the identity matrix, then the noise q is said to be *white*. In this case, the BLUE estimate of γ is simply the average of the x_n.

24.1.2 A known signal vector in noise

Generalizing somewhat, we consider the case in which the data vector x has the form

$$x = \gamma s + q, \tag{24.5}$$

where $s = (s_1, ..., s_N)^T$ is a known signal vector. The BLUE estimator is

$$b = \frac{1}{s^\dagger Q^{-1} s} Q^{-1} s \tag{24.6}$$

and the BLUE estimate of γ is now

$$\hat{\gamma} = \frac{1}{s^\dagger Q^{-1} s} s^\dagger Q^{-1} x. \tag{24.7}$$

In numerous applications of signal processing, the signal vectors take the form of sampled sinusoids; that is, $s = e_\theta$, with

$$e_\theta = \frac{1}{\sqrt{N}}(e^{-i\theta}, e^{-2i\theta}, ..., e^{-Ni\theta})^T, \tag{24.8}$$

where θ is a frequency in the interval $[0, 2\pi)$. If the noise is white, then the BLUE estimate of γ is

$$\hat{\gamma} = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} x_n e^{in\theta}, \tag{24.9}$$

which is the *discrete Fourier transform*(DFT) of the data, evaluated at the frequency θ.

24.1.3 Multiple signals in noise

Suppose now that the data values are

$$x_n = \sum_{m=1}^{M} \gamma_m s_n^m + q_n, \tag{24.10}$$

where the signal vectors $s^m = (s_1^m, ..., s_N^m)^T$ are known and we want to estimate the γ_m. We write this in matrix-vector notation as

$$x = Sc + q, \tag{24.11}$$

where S is the matrix with entries $S_{nm} = s_n^m$, and our goal is to find $c = (\gamma_1, ..., \gamma_N)^T$, the vector of coefficients. The BLUE estimate of the vector c is

$$\hat{c} = (S^\dagger Q^{-1} S)^{-1} S^\dagger Q^{-1} x, \tag{24.12}$$

assuming that the matrix $S^\dagger Q^{-1} S$ is invertible, in which case we must have $M \leq N$.

If the signals s^m are mutually orthogonal and have length one, then $S^\dagger S = I$; if, in addition, the noise is white, the BLUE estimate of c is $\hat{c} = S^\dagger x$, so that

$$\hat{c}_m = \sum_{n=1}^{N} x_n \overline{s_n^m}. \tag{24.13}$$

This case arises when the signals are $s^m = e_{\theta_m}$, for $\theta_m = 2\pi m/M$, for $m = 1, ..., M$, in which case the BLUE estimate of c_m is

$$\hat{c}_m = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} x_n e^{2\pi i m n/M}, \tag{24.14}$$

the DFT of the data, evaluated at the frequency θ_m. Note that when the frequencies θ_m are not these, the matrix $S^\dagger S$ is not I, and the BLUE estimate is not obtained from the DFT of the data.

24.2 Detection

As we noted previously, the detection problem is a special case of estimation. Detecting the known signal s in noise is equivalent to deciding if the coefficient γ is zero or not. The procedure is to calculate $\hat{\gamma}$, the BLUE estimate of γ, and say that s has been detected if $|\hat{\gamma}|$ exceeds a certain threshold. In the case of multiple known signals, we calculate $\hat{c}$, the BLUE estimate of the coefficient vector c, and base our decisions on the magnitudes of each entry of $\hat{c}$.

24.2.1 Parametrized signal

It is sometimes the case that we know that the signal s we seek to detect is a member of a parametrized family, $\{s_\theta | \theta \in \Theta\}$, of potential signal vectors, but we do not know the value of the parameter θ. For example, we may be trying to detect a sinusoidal signal, $s = e_\theta$, where θ is an unknown frequency in the interval $[0, 2\pi)$. In sonar direction-of-arrival estimation, we seek to detect a farfield point source of acoustic energy, but do not know the direction of the source. The BLUE estimator can be extended to these cases, as well [57]. For each fixed value of the parameter θ, we estimate γ using the BLUE, obtaining the estimate

$$\hat{\gamma}(\theta) = \frac{1}{s_\theta^\dagger Q^{-1} s_\theta} s_\theta^\dagger Q^{-1} x, \tag{24.15}$$

which is then a function of θ. If the maximum of the magnitude of this function exceeds a specified threshold, then we may say that there is a signal present corresponding to that value of θ.

Another approach would be to extend the model of multiple signals to include a continuum of possibilities, replacing the finite sum with an integral. Then the model of the data becomes

$$x = \int_{\theta \in \Theta} \gamma(\theta) s_\theta d\theta + q. \tag{24.16}$$

Let S now denote the integral operator

$$S(\gamma) = \int_{\theta \in \Theta} \gamma(\theta) s_\theta d\theta \tag{24.17}$$

that transforms a function γ of the variable θ into a vector. The adjoint operator, $S^\dagger$, transforms any N-vector v into a function of θ, according to

$$S^\dagger(v)(\theta) = \sum_{n=1}^{N} v_n \overline{(s_\theta)_n} = s_\theta^\dagger v \,. \tag{24.18}$$

Consequently, $S^\dagger Q^{-1} S$ is the function of θ given by

$$g(\theta) = (S^\dagger Q^{-1} S)(\theta) = \sum_{n=1}^{N} \sum_{j=1}^{N} Q_{nj}^{-1} (s_\theta)_j \overline{(s_\theta)_n}, \tag{24.19}$$

so

$$g(\theta) = s_\theta^\dagger Q^{-1} s_\theta. \tag{24.20}$$

The generalized BLUE estimate of $\gamma(\theta)$ is then

$$\hat{\gamma}(\theta) = \frac{1}{g(\theta)} \sum_{j=1}^{N} a_j \overline{(s_\theta)_j} = \frac{1}{g(\theta)} s_\theta^\dagger a \,, \tag{24.21}$$

where $x = Qa$ or

$$x_n = \sum_{j=1}^{N} a_j Q_{nj}, \tag{24.22}$$

for $j = 1, ..., N$, and so $a = Q^{-1}x$. This is the same estimate we obtained in the previous paragraph. The only difference is that, in the first case, we assume that there is only one signal active, and apply the BLUE for each fixed θ, looking for the one most likely to be active. In the second case, we choose to view the data as a noisy superposition of a continuum of the s_θ, not just one. The resulting estimate of $\gamma(\theta)$ describes how each of the individual signal vectors s_θ contribute to the data vector x. Nevertheless, the calculations we perform are the same.

If the noise is white, we have $a_j = x_j$ for each j. The function $g(\theta)$ becomes

$$g(\theta) = \sum_{n=1}^{N} |(s_\theta)_n|^2, \tag{24.23}$$

which is simply the square of the length of the vector s_θ. If, in addition, the signal vectors all have length one, then the estimate of the function $\gamma(\theta)$ becomes

$$\hat{\gamma}(\theta) = \sum_{n=1}^{N} x_n \overline{(s_\theta)_n} = s_\theta^\dagger x. \tag{24.24}$$

Finally, if the signals are sinusoids $s_\theta = e_\theta$, then

$$\hat{\gamma}(\theta) = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} x_n e^{in\theta}, \tag{24.25}$$

again, the DFT of the data vector.

24.3 Discrimination

The problem now is to decide if the data is $x = s^1 + q$ or $x = s^2 + q$, where s^1 and s^2 are known vectors. This problem can be converted into a detection problem: Do we have $x - s^1 = q$ or $x - s^1 = s^2 - s^1 + q$? Then the BLUE involves the vector $Q^{-1}(s^2 - s^1)$ and the discrimination is made based on the quantity $(s^2 - s^1)^\dagger Q^{-1} x$. If this quantity is near enough to zero we say that the signal is s^1; otherwise, we say that it is s^2. The BLUE in this case is sometimes called the *Hotelling linear discriminant*, and a procedure that uses this method to perform medical diagnostics is called a *Hotelling observer*.

More generally, suppose we want to decide if a given vector x comes from class C_1 or from class C_2. If we can find a vector b such that $b^T x > a$ for every x that comes from C_1, and $b^T x < a$ for every x that comes from C_2, then the vector b is a linear discriminant for deciding between the classes C_1 and C_2.

24.3.1 Channelized Observers

The N by N matrix Q can be quite large, particularly when x and q are vectorizations of two-dimensional images. If, in additional, the matrix Q is obtained from K observed instances of the random vector q, then for Q to be invertible, we need $K \geq N$. To avoid these and other difficulties, the *channelized* Hotelling linear discriminant is often used. The idea here is to replace the data vector x with Ux for an appropriately chosen J by N matrix U, with J much smaller than N; the value $J = 3$ is used in [117], with the channels chosen to capture image information within selected frequency bands.

24.3.2 An Example of Discrimination

Suppose that there are two groups of students, the first group denoted G_1, the second G_2. The math SAT score for the students in G_1 is always above 500, while their verbal scores are always below 500. For the students in G_2 the opposite is true; the math scores are below 500, the verbal above. For each student we create the two-dimensional vector $x = (x_1, x_2)^T$ of SAT scores, with x_1 the math score, x_2 the verbal score. Let $b = (1, -1)^T$. Then for every student in G_1 we have $b^T x > 0$, while for those in G_2, we have $b^T x < 0$. Therefore, the vector b provides a linear discriminant.

Suppose we have a third group, G_3, whose math scores and verbal scores are both below 500. To discriminate between members of G_1 and G_3 we can use the vector $b = (1, 0)^T$ and $a = 500$. To discriminate between the groups G_2 and G_3, we can use the vector $b = (0, 1)^T$ and $a = 500$.

Now suppose that we want to decide from which of the three groups the vector x comes; this is classification.

24.4 Classification

The classification problem is to determine to which of several classes of vectors a given vector x belongs. For simplicity, we assume all vectors are real. The simplest approach to solving this problem is to seek linear discriminant functions; that is, for each class we want to have a vector b with the property that $b^T x > 0$ if and only if x is in the class. If the vectors x are randomly distributed according to one of the parametrized family of probability density functions (pdf) $p(x; \omega)$ and the ith class corresponds to the parameter value ω_i then we can often determine the discriminant vectors b^i from these pdf. In many cases, however, we do not have the pdf and the b^i must be estimated through a learning or training step before they are used on as yet unclassified data vectors. In the discussion that follows we focus on obtaining b for one class, suppressing the index i.

24.4.1 The Training Stage

In the training stage a candidate for b is tested on vectors whose class membership is known, say $\{x^1, ..., x^M\}$. First, we replace each vector x^m that is not in the class with its negative. Then we seek b such that $b^T x^m > 0$ for all m. With A the matrix whose mth row is $(x^m)^T$ we can write the problem as $Ab > 0$. If the b we obtain has some entries very close to zero it might not work well enough on actual data; it is often better, then, to take a vector ϵ with small positive entries and require $Ab \geq \epsilon$. When we have found b for each class we then have the machinery to perform the classification task.

There are several problems to be overcome, obviously. The main one is that there may not be a vector b for each class; the problem $Ab \geq \epsilon$ need not have a solution. In classification this is described by saying that the vectors x^m are not linearly separable [96]. The second problem is finding the b for each class; we need an algorithm to solve $Ab \geq \epsilon$.

One approach to designing an algorithm for finding b is the following: for arbitrary b let $f(b)$ be the number of the x^m misclassified by vector b. Then minimize $f(b)$ with respect to b. Alternatively, we can minimize the function $g(b)$ defined to be the sum of the values $-b^T x^m$, taken over all the x^m that are misclassified; the $g(b)$ has the advantage of being continuously valued. The batch Perceptron algorithm [96] uses gradient descent methods to minimize $g(b)$. Another approach is to use the Agmon-Motzkin-Schoenberg (AMS) algorithm to solve the system of linear inequalities $Ab \geq \epsilon$ [57].

When the training set of vectors is linearly separable, the batch Perceptron and the AMS algorithms converge to a solution, for each class. When the training vectors are not linearly separable there will be a class for which the problem $Ab \geq \epsilon$ will have no solution. Iterative algorithms in this case cannot converge to a solution. Instead, they may converge to an approximate solution or, as with the AMS algorithm, converge subsequentially to a limit cycle of more than one vector.

24.4.2 Our Example Again

We return to the example given earlier, involving the three groups of students and their SAT scores. To be consistent with the conventions of this section, we define $x = (x_1, x_2)^T$ differently now. Let x_1 be the math SAT score, minus 500, and x_2 be the verbal SAT score, minus 500. The vector $b = (1, 0)^T$ has the property that $b^T x > 0$ for each x coming from G_1, but $b^T x < 0$ for each x not coming from G_1. Similarly, the vector $b = (0, 1)^T$ has the property that $b^T x > 0$ for all x coming from G_2, while $b^T x < 0$ for all x not coming from G_2. However, there is no vector b with the property that $b^T x > 0$ for x coming from G_3, but $b^T x < 0$ for all x not coming from G_3; the group G_3 is not linearly separable from the others. Notice, however, that if we perform our classification sequentially, we can employ linear classifiers. First, we use the vector $b = (1, 0)^T$ to decide if the vector x comes from G_1 or not. If it does, fine; if not, then use vector $b = (0, 1)^T$ to decide if it comes from G_2 or G_3.

24.5 More realistic models

In many important estimation and detection problems, the signal vector s is not known precisely. In medical diagnostics, we may be trying to detect a lesion, and may know it when we see it, but may not be able to describe it

using a single vector s, which now would be a vectorized image. Similarly, in discrimination or classification problems, we may have several examples of each type we wish to identify, but will be unable to reduce these types to single representative vectors. We now have to derive an analog of the BLUE that is optimal with respect to the examples that have been presented for training. The linear procedure we seek will be one that has performed best, with respect to a training set of examples. The *Fisher linear discriminant* is an example of such a procedure.

24.5.1 The Fisher linear discriminant

Suppose that we have available for training K vectors $x^1, ..., x^K$ in R^N, with vectors $x^1, ..., x^J$ in the class A, and the remaining $K - J$ vectors in the class B. Let w be an arbitrary vector of length one, and for each k let $y_k = w^T x^k$ be the projected data. The numbers y_k, $k = 1, ..., J$, form the set Y_A, the remaining ones the set Y_B. Let

$$\mu_A = \frac{1}{J} \sum_{k=1}^{J} x^k, \tag{24.26}$$

$$\mu_B = \frac{1}{K-J} \sum_{k=J+1}^{K} x^k, \tag{24.27}$$

$$m_A = \frac{1}{J} \sum_{k=1}^{J} y_k = w^T \mu_A, \tag{24.28}$$

and

$$m_B = \frac{1}{K-J} \sum_{k=J+1}^{K} y_k = w^T \mu_B. \tag{24.29}$$

Let

$$\sigma_A^2 = \sum_{k=1}^{J} (y_k - m_A)^2, \tag{24.30}$$

and

$$\sigma_B^2 = \sum_{k=J+1}^{K} (y_k - m_B)^2. \tag{24.31}$$

The quantity $\sigma^2 = \sigma_A^2 + \sigma_B^2$ is the *total within-class scatter* of the projected data. Define the function $F(w)$ to be

$$F(w) = \frac{(m_A - m_B)^2}{\sigma^2}. \tag{24.32}$$

The *Fisher linear discriminant* is the vector w for which $F(w)$ achieves its maximum.

Define the scatter matrices S_A and S_B as follows:

$$S_A = \sum_{k=1}^{J} (x^k - \mu_A)(x^k - \mu_A)^T, \tag{24.33}$$

and

$$S_B = \sum_{k=J+1}^{K} (x^k - \mu_B)(x^k - \mu_B)^T. \tag{24.34}$$

Then

$$S_{within} = S_A + S_B \tag{24.35}$$

is the *within-class scatter matrix* and

$$S_{between} = (\mu_A - \mu_B)(\mu_A - \mu_B)^T \tag{24.36}$$

is the *between-class scatter matrix*. The function $F(w)$ can then be written as

$$F(w) = w^T S_{between} w / w^T S_{within} w. \tag{24.37}$$

The w for which $F(w)$ achieves its maximum value is then

$$w = S_{within}^{-1}(\mu_A - \mu_B). \tag{24.38}$$

This vector w is the Fisher linear discriminant. When a new data vector x is obtained, we decide to which of the two classes it belongs by calculating $w^T x$.

24.6 A more general estimation problem

It is often the case, in practice, that the object of interest is a function of one or several continuous variables, and our data consists of finitely many linear functional values. For example, suppose that our object of interest is the function of two real variables $f(u, v)$, and that our data are the values

$$x_n = \int\int f(u, v) h_n(u, v) du dv + q_n, \tag{24.39}$$

for noise q_n and known functions $h_n(u,v)$, $n = 1, ..., N$. Our goal may be to reconstruct the function $f(u,v)$ itself, or, more modestly, to estimate some other linear functional value, $\int\int f(u,v)g(u,v)dudv$, such as the integral of $f(u,v)$ over some two-dimensional set A. We consider only estimates that are linear in the data x. Unfortunately, we can obtain an unbiased estimate of $\int\int f(u,v)g(u,v)dudv$ only if we can calculate $\int\int f(u,v)g(u,v)dudv$ from noise-free data, for any $f(u,v)$, which can be done only if the function $g(u,v)$ has the form

$$g(u,v) = \sum_{n=1}^{N} a_n h_n(u,v), \tag{24.40}$$

for some constants a_n. This rather negative result suggests that the information about $f(u,v)$ that we can expect to extract from the data is quite limited. On the other hand, if we should know, in advance, that $f(u,v)$ is a member of a parametrized family of functions and if the data is sufficient to calculate the parameter, then not only can we estimate $\int\int f(u,v)g(u,v)dudv$ from the data, for every $g(u,v)$, but we can determine $f(u,v)$ itself.

To investigate this problem further, we assume that f and the h_n are members of a Hilbert space X, such as $L^2(R)$ or $L^2(R^2)$. Since the problem of obtaining an unbiased linear estimate is equivalent to that of achieving perfect reconstruction from noise-free data, we assume that the data we have are

$$x_n = \langle f, h_n \rangle, \tag{24.41}$$

where $\langle a, b \rangle$ denotes the inner product in the space X. For $X = L^2(R^2)$ we have

$$\langle a, b \rangle = \int\int a(u,v)\overline{b(u,v)}dudv. \tag{24.42}$$

The goal is to reconstruct the linear functional $\langle f, g \rangle$ as a linear combination of the entries of the data vector x.

Each g in X can be written in the form

$$g = \sum_{n=1}^{N} c_n h_n + z, \tag{24.43}$$

for some choice of constants c_n and some z with the property that

$$\langle z, h_n \rangle = 0, \tag{24.44}$$

for each n. Then we have

$$\langle f, g \rangle = \sum_{n=1}^{N} c_n \langle f, h_n \rangle + \langle f, z \rangle = \sum_{n=1}^{N} c_n x_n + \langle f, z \rangle. \tag{24.45}$$

The problem then is that we cannot determine the quantity $\langle f, z\rangle$ from the data, in general.

However, if it should be the case that f is a linear combination of the h_n, that is, there are constants a_n so that

$$f = \sum_{n=1}^{N} a_n h_n, \tag{24.46}$$

then $\langle f, z\rangle = 0$. But why should it be the case?

Notice that the data we have measured exists prior to the specification of the Hilbert space X. By choosing different Hilbert spaces, the data can be represented in different ways, using different inner products and different h_n. To make this somewhat abstract statement more concrete, consider the example of Fourier-transform data.

24.6.1 An Example: Fourier-Transform Data

Suppose that the object of interest is $f(r)$, a function of the single real variable r. Suppose that our data values are

$$x_n = F(\omega_n) = \int f(r) e^{-i\omega_n r} dr, \tag{24.47}$$

for $n = 1, ..., N$, and ω_n arbitrary frequencies. With $X = L^2(R)$, we can write

$$x_n = F(\omega_n) = \langle f, h_n\rangle, \tag{24.48}$$

for

$$h_n(r) = e^{i\omega_n r}. \tag{24.49}$$

Then we will have f in the span of the h_n if f can be written

$$f(r) = \sum_{n=1}^{N} a_n e^{i\omega_n r}, \tag{24.50}$$

for some constants a_n. However, unless N is very large, or the $h_n(r)$ have been carefully chosen, f will probably not be well described by such a sum.

But we should not give up! We can also write

$$x_n = \int f(r) p(r) e^{-i\omega_n r} p(r)^{-1} dr, \tag{24.51}$$

where $p(r) > 0$. If we define X now to be the Hilbert space with

$$\langle s, t\rangle = \int s(r)\overline{t(r)} p(r)^{-1} dr, \tag{24.52}$$

then

$$h_n(r) = p(r)e^{i\omega_n r}. \tag{24.53}$$

Now we will have f in the span of the h_n if

$$f(r) = p(r)\sum_{n=1}^{N} a_n e^{i\omega_n r}, \tag{24.54}$$

for some a_n. If we have prior knowledge about $f(r)$, or, more precisely, about $|f(r)|$, such as its support, or any prominent components that it may have, we can include them in a prior estimate $p(r)$ of $|f(r)|$, making it much more likely that f lies in the span of the h_n, or, at least, can be well approximately by members of this span.

This approach was developed for image reconstruction from Fourier data in [34, 35, 41]. In those papers it was called the PDFT estimator. See the appendix for more discussion of Fourier-transform estimation.

24.6.2 More Generally

In general, if we want to make it plausible that f lies in the span of the h_n, we can alter the ambient Hilbert space, and its inner product, so that the h_n that represent the data also have a good chance of capturing the desired f within their span. This freedom to tailor the Hilbert space to the f, using prior knowledge of f, is the *way out* that we need to overcome the negative result we saw early on.

24.7 Conclusions

We always have finite data. In the absence of additional knowledge about f, we can say little, unless the data set is large. But, in most reconstruction problems we do have additional information, often qualitiative, about the object f to be recovered. We may, for instance, be willing to say that f is well-approximated by a finite sum of pixels, voxels, or blobs. Finite data, if there is enough of it, will then suffice to recover f, at least approximately, from which we can calculate any desired linear-functional value. The example above, involving Fourier data, shows how we can use prior knowledge to tailor the ambient Hilbert space, to get beyond the negative earlier result. The negative result reinforces the point that there is no *one-size-fits-all* method that will work for all f, but for each individual f, if we have prior knowledge about it, all is not lost. There have been a great many papers stressing the importance of prior information in reconstruction from limited data [38, 86].

Chapter 25

Appendix: Planewave Propagation

In this chapter we demonstrate how the Fourier transform arises naturally as we study the signals received in the farfield from an array of tranmitters or reflectors. We restrict our attention to single-frequency, or narrowband, signals.

25.1 Transmission and Remote-Sensing

For pedagogical reasons, we shall discuss separately what we shall call the transmission and the remote-sensing problems, although the two problems are opposite sides of the same coin, in a sense. In the one-dimensional transmission problem, it is convenient to imagine the transmitters located at points $(x, 0)$ within a bounded interval $[-A, A]$ of the x-axis, and the measurements taken at points P lying on a circle of radius D, centered at the origin. The radius D is large, with respect to A. It may well be the case that no actual sensing is to be performed, but rather, we are simply interested in what the received signal pattern is at points P distant from the transmitters. Such would be the case, for example, if we were analyzing or constructing a transmission pattern of radio broadcasts. In the remote-sensing problem, in contrast, we imagine, in the one-dimensional case, that our sensors occupy a bounded interval of the x-axis, and the transmitters or reflectors are points of a circle whose radius is large, with respect to the size of the bounded interval. The actual size of the radius does not matter and we are interested in determining the amplitudes of the transmitted or reflected signals, as a function of angle only. Such is the case in astronomy, farfield sonar or radar, and the like. Both the transmission and remote-sensing problems illustrate the important role played by the

Fourier transform.

25.2 The Transmission Problem

We identify two distinct transmission problems: the direct problem and the inverse problem. In the direct transmission problem, we wish to determine the farfield pattern, given the complex amplitudes of the transmitted signals. In the inverse transmission problem, the array of transmitters or reflectors is the object of interest; we are given, or we measure, the farfield pattern and wish to determine the amplitudes. For simplicity, we consider only single-frequency signals.

We suppose that each point x in the interval $[-A, A]$ transmits the signal $f(x)e^{i\omega t}$, where $f(x)$ is the complex amplitude of the signal and $\omega > 0$ is the common fixed frequency of the signals. Let $D > 0$ be large, with respect to A, and consider the signal received at each point P given in polar coordinates by $P = (D, \theta)$. The distance from $(x, 0)$ to P is approximately $D - x\cos\theta$, so that, at time t, the point P receives from $(x, 0)$ the signal $f(x)e^{i\omega(t-(D-x\cos\theta)/c)}$, where c is the propagation speed. Therefore, the combined signal received at P is

$$B(P,t) = e^{i\omega t}e^{-i\omega D/c}\int_{-A}^{A} f(x)e^{ix\frac{\omega\cos\theta}{c}}dx. \tag{25.1}$$

The integral term, which gives the farfield pattern of the transmission, is

$$F(\frac{\omega\cos\theta}{c}) = \int_{-A}^{A} f(x)e^{ix\frac{\omega\cos\theta}{c}}dx, \tag{25.2}$$

where $F(\gamma)$ is the Fourier transform of $f(x)$, given by

$$F(\gamma) = \int_{-A}^{A} f(x)e^{ix\gamma}dx. \tag{25.3}$$

How $F(\frac{\omega\cos\theta}{c})$ behaves, as a function of θ, as we change A and ω, is discussed in some detail in the chapter on direct transmission.

Consider, for example, the function $f(x) = 1$, for $|x| \leq A$, and $f(x) = 0$, otherwise. The Fourier transform of $f(x)$ is

$$F(\gamma) = 2A\text{sinc}(A\gamma), \tag{25.4}$$

where $\text{sinc}(t)$ is defined to be

$$\text{sinc}(t) = \frac{\sin(t)}{t}, \tag{25.5}$$

for $t \neq 0$, and $\text{sinc}(0) = 1$. Then $F(\frac{\omega \cos\theta}{c}) = 2A$ when $\cos\theta = 0$, so when $\theta = \frac{\pi}{2}$ and $\theta = \frac{3\pi}{2}$. We will have $F(\frac{\omega \cos\theta}{c}) = 0$ when $A\frac{\omega \cos\theta}{c} = \pi$, or $\cos\theta = \frac{\pi c}{A\omega}$. Therefore, the transmission pattern has no nulls if $\frac{\pi c}{A\omega} > 1$. In order for the transmission pattern to have nulls, we need $A > \frac{\lambda}{2}$, where $\lambda = \frac{2\pi c}{\omega}$ is the wavelength. This rather counterintuitive fact, namely that we need more signals transmitted in order to receive less at certain locations, illustrates the phenomenon of destructive interference.

25.3 Reciprocity

For certain remote-sensing applications, such as sonar and radar array processing and astronomy, it is convenient to switch the roles of sender and receiver. Imagine that superimposed planewave fields are sensed at points within some bounded region of the interior of the sphere, having been transmitted or reflected from the points P on the surface of a sphere whose radius D is large with respect to the bounded region. The *reciprocity principle* tells us that the same mathematical relation holds between points P and $(x, 0)$, regardless of which is the sender and which the receiver. Consequently, the data obtained at the points $(x, 0)$ are then values of the inverse Fourier transform of the function describing the amplitude of the signal sent from each point P.

25.4 Remote Sensing

A basic problem in remote sensing is to determine the nature of a distant object by measuring signals transmitted by or reflected from that object. If the object of interest is sufficiently remote, that is, is in the *farfield*, the data we obtain by sampling the propagating spatio-temporal field is related, approximately, to what we want by *Fourier transformation*. The problem is then to estimate a function from finitely many (usually noisy) values of its *Fourier transform*. The application we consider here is a common one of remote-sensing of transmitted or reflected waves propagating from distant sources. Examples include optical imaging of planets and asteroids using reflected sunlight, radio-astronomy imaging of distant sources of radio waves, active and passive sonar, and radar imaging.

25.5 The Wave Equation

In many areas of remote sensing, what we measure are the fluctuations in time of an electromagnetic or acoustic field. Such fields are described mathematically as solutions of certain partial differential equations, such

as the *wave equation*. A function $u(x, y, z, t)$ is said to satisfy the *three-dimensional wave equation* if

$$u_{tt} = c^2(u_{xx} + u_{yy} + u_{zz}) = c^2\nabla^2 u, \tag{25.6}$$

where u_{tt} denotes the second partial derivative of u with respect to the time variable t twice and $c > 0$ is the (constant) speed of propagation. More complicated versions of the wave equation permit the speed of propagation c to vary with the spatial variables x, y, z, but we shall not consider that here.

We use the method of *separation of variables* at this point, to get some idea about the nature of solutions of the wave equation. Assume, for the moment, that the solution $u(t, x, y, z)$ has the simple form

$$u(t, x, y, z) = f(t)g(x, y, z). \tag{25.7}$$

Inserting this separated form into the wave equation, we get

$$f''(t)g(x, y, z) = c^2 f(t)\nabla^2 g(x, y, z) \tag{25.8}$$

or

$$f''(t)/f(t) = c^2\nabla^2 g(x, y, z)/g(x, y, z). \tag{25.9}$$

The function on the left is independent of the spatial variables, while the one on the right is independent of the time variable; consequently, they must both equal the same constant, which we denote $-\omega^2$. From this we have two separate equations,

$$f''(t) + \omega^2 f(t) = 0, \tag{25.10}$$

and

$$\nabla^2 g(x, y, z) + \frac{\omega^2}{c^2} g(x, y, z) = 0. \tag{25.11}$$

Equation (25.11) is the *Helmholtz equation*.

Equation (25.10) has for its solutions the functions $f(t) = \cos(\omega t)$ and $\sin(\omega t)$, or, in complex form, the complex exponential functions $f(t) = e^{i\omega t}$ and $f(t) = e^{-i\omega t}$. Functions $u(t, x, y, z) = f(t)g(x, y, z)$ with such time dependence are called *time-harmonic* solutions.

25.6 Planewave Solutions

Suppose that, beginning at time $t = 0$, there is a localized disturbance. As time passes, that disturbance spreads out spherically. When the radius

of the sphere is very large, the surface of the sphere appears planar, to an observer on that surface, who is said then to be in the *far field*. This motivates the study of solutions of the wave equation that are constant on planes; the so-called *planewave solutions*.

Let $\mathbf{s} = (x, y, z)$ and $u(\mathbf{s}, t) = u(x, y, z, t) = e^{i\omega t}e^{i\mathbf{k}\cdot\mathbf{s}}$. Then we can show that u satisfies the wave equation $u_{tt} = c^2\nabla^2 u$ for any real vector $\mathbf{k}$, so long as $||\mathbf{k}||^2 = \omega^2/c^2$. This solution is a planewave associated with frequency ω and *wavevector* $\mathbf{k}$; at any fixed time the function $u(\mathbf{s}, t)$ is constant on any plane in three-dimensional space having $\mathbf{k}$ as a normal vector.

In radar and sonar, the field $u(\mathbf{s}, t)$ being sampled is usually viewed as a discrete or continuous superposition of planewave solutions with various amplitudes, frequencies, and wavevectors. We sample the field at various spatial locations $\mathbf{s}$, for various times t. Here we simplify the situation a bit by assuming that all the planewave solutions are associated with the same frequency, ω. If not, we can perform an FFT on the functions of time received at each sensor location $\mathbf{s}$ and keep only the value associated with the desired frequency ω.

25.7 Superposition and the Fourier Transform

In the continuous superposition model, the field is

$$u(\mathbf{s}, t) = e^{i\omega t}\int F(\mathbf{k})e^{i\mathbf{k}\cdot\mathbf{s}}d\mathbf{k}. \tag{25.12}$$

Our measurements at the sensor locations $\mathbf{s}$ give us the values

$$f(\mathbf{s}) = \int F(\mathbf{k})e^{i\mathbf{k}\cdot\mathbf{s}}d\mathbf{k}. \tag{25.13}$$

The data are then Fourier transform values of the complex function $F(\mathbf{k})$; $F(\mathbf{k})$ is defined for all three-dimensional real vectors $\mathbf{k}$, but is zero, in theory, at least, for those $\mathbf{k}$ whose squared length $||\mathbf{k}||^2$ is not equal to ω^2/c^2. Our goal is then to estimate $F(\mathbf{k})$ from measured values of its Fourier transform. Since each $\mathbf{k}$ is a normal vector for its planewave field component, determining the value of $F(\mathbf{k})$ will tell us the strength of the planewave component coming from the direction $\mathbf{k}$.

25.7.1 The Spherical Model

We can imagine that the sources of the planewave fields are the points P that lie on the surface of a large sphere centered at the origin. For each P, the ray from the origin to P is parallel to some wavevector $\mathbf{k}$. The function $F(\mathbf{k})$ can then be viewed as a function $F(P)$ of the points P. Our measurements will be taken at points $\mathbf{s}$ inside this sphere. The radius of

the sphere is assumed to be orders of magnitude larger than the distance between sensors. The situation is that of astronomical observation of the heavens using ground-based antennas. The sources of the optical or electromagnetic signals reaching the antennas are viewed as lying on a large sphere surrounding the earth. Distance to the sources is not considered now, and all we are interested in are the amplitudes $F(\mathbf{k})$ of the fields associated with each direction $\mathbf{k}$.

25.8 Sensor Arrays

In some applications the sensor locations are essentially arbitrary, while in others their locations are carefully chosen. Sometimes, the sensors are collinear, as in sonar towed arrays.

25.8.1 The Two-Dimensional Array

Suppose now that the sensors are in locations $\mathbf{s} = (x, y, 0)$, for various x and y; then we have a *planar array* of sensors. Then the dot product $\mathbf{s} \cdot \mathbf{k}$ that occurs in Equation (25.13) is

$$\mathbf{s} \cdot \mathbf{k} = xk_1 + yk_2; \tag{25.14}$$

we cannot *see* the third component, k_3. However, since we know the size of the vector $\mathbf{k}$, we can determine $|k_3|$. The only ambiguity that remains is that we cannot distinguish sources on the upper hemisphere from those on the lower one. In most cases, such as astronomy, it is obvious in which hemisphere the sources lie, so the ambiguity is resolved.

The function $F(\mathbf{k})$ can then be viewed as $F(k_1, k_2)$, a function of the two variables k_1 and k_2. Our measurements give us values of $f(x, y)$, the two-dimensional Fourier transform of $F(k_1, k_2)$. Because of the limitation $||\mathbf{k}|| = \frac{\omega}{c}$, the function $F(k_1, k_2)$ has bounded support. Consequently, its Fourier transform cannot have bounded support. As a result, we can never have all the values of $f(x, y)$, and so cannot hope to reconstruct $F(k_1, k_2)$ exactly, even for noise-free data.

25.8.2 The One-Dimensional Array

If the sensors are located at points $\mathbf{s}$ having the form $\mathbf{s} = (x, 0, 0)$, then we have a *line array* of sensors. The dot product in Equation (25.13) becomes

$$\mathbf{s} \cdot \mathbf{k} = xk_1. \tag{25.15}$$

Now the ambiguity is greater than in the planar array case. Once we have k_1, we know that

$$k_2^2 + k_3^2 = (\frac{\omega}{c})^2 - k_1^2, \tag{25.16}$$

which describes points P lying on a circle on the surface of the distant sphere, with the vector $(k_1, 0, 0)$ pointing at the center of the circle. It is said then that we have a *cone of ambiguity*. One way to resolve the situation is to assume $k_3 = 0$; then $|k_2|$ can be determined and we have remaining only the ambiguity involving the sign of k_2. Once again, in many applications, this remaining ambiguity can be resolved by other means.

Once we have resolved any ambiguity, we can view the function $F(\mathbf{k})$ as $F(k_1)$, a function of the single variable k_1. Our measurements give us values of $f(x)$, the Fourier transform of $F(k_1)$. As in the two-dimensional case, the restriction on the size of the vectors $\mathbf{k}$ means that the function $F(k_1)$ has bounded support. Consequently, its Fourier transform, $f(x)$, cannot have bounded support. Therefore, we shall never have all of $f(x)$, and so cannot hope to reconstruct $F(k_1)$ exactly, even for noise-free data.

25.8.3 Limited Aperture

In both the one- and two-dimensional problems, the sensors will be placed within some bounded region, such as $|x| \leq A$, $|y| \leq B$ for the two-dimensional problem, or $|x| \leq A$ for the one-dimensional case. These bounded regions are the *apertures* of the arrays. The larger these apertures are, in units of the wavelength, the better the resolution of the reconstructions.

In digital array processing there are only finitely many sensors, which then places added limitations on our ability to reconstruction the field amplitude function $F(\mathbf{k})$.

25.9 The Remote-Sensing Problem

We shall begin our discussion of the remote-sensing problem by considering an extended object transmitting or reflecting a single-frequency, or *narrowband*, signal. The narrowband, extended-object case is a good place to begin, since a point object is simply a limiting case of an extended object, and broadband received signals can always be filtered to reduce their frequency band.

25.9.1 The Solar-Emission Problem

In [23] Bracewell discusses the *solar-emission* problem. In 1942, it was observed that radio-wave emissions in the one-meter wavelength range were arriving from the sun. Were they coming from the entire disk of the sun or were the sources more localized, in sunspots, for example? The problem then was to view each location on the sun's surface as a potential source of these radio waves and to determine the intensity of emission corresponding to each location.

For electromagnetic waves the propagation speed is the speed of light in a vacuum, which we shall take here to be $c = 3 \times 10^8$ meters per second. The wavelength λ for gamma rays is around one Angstrom, which is 10^{-10} meters; for x-rays it is about one millimicron, or 10^{-9} meters. The visible spectrum has wavelengths that are a little less than one micron, that is, 10^{-6} meters. Shortwave radio has a wavelength around one millimeter; microwaves have wavelengths between one centimeter and one meter. Broadcast radio has a λ running from about 10 meters to 1000 meters, while the so-called long radio waves can have wavelengths several thousand meters long.

The sun has an angular diameter of 30 min. of arc, or one-half of a degree, when viewed from earth, but the needed resolution was more like 3 min. of arc. As we shall see shortly, such resolution requires a radio telescope 1000 wavelengths across, which means a diameter of 1km at a wavelength of 1 meter; in 1942 the largest military radar antennas were less than 5 meters across. A solution was found, using the method of reconstructing an object from line-integral data, a technique that surfaced again in tomography. The problem here is inherently two-dimensional, but, for simplicity, we shall begin with the one-dimensional case.

25.10 Sampling

In the one-dimensional case, the signal received at the point $(x, 0, 0)$ is essentially the inverse Fourier transform $f(x)$ of the function $F(k_1)$; for notational simplicity, we write $k = k_1$. The $F(k)$ supported on a bounded interval $|k| \leq \frac{\omega}{c}$, so $f(x)$ cannot have bounded support. As we noted earlier, to determine $F(k)$ exactly, we would need measurements of $f(x)$ on an unbounded set. But, which unbounded set?

Because the function $F(k)$ is zero outside the interval $[-\frac{\omega}{c}, \frac{\omega}{c}]$, the function $f(x)$ is *band-limited*. The *Nyquist spacing* in the variable x is therefore

$$\Delta_x = \frac{\pi c}{\omega}. \tag{25.17}$$

The wavelength λ associated with the frequency ω is defined to be

$$\lambda = \frac{2\pi c}{\omega}, \tag{25.18}$$

so that

$$\Delta_x = \frac{\lambda}{2}. \tag{25.19}$$

The significance of the Nyquist spacing comes from *Shannon's Sampling Theorem*, which says that if we have the values $f(m\Delta_x)$, for all integers m, then we have enough information to recover $F(k)$ exactly. In practice, of course, this is never the case.

25.11 The Limited-Aperture Problem

In the remote-sensing problem, our measurements at points $(x, 0, 0)$ in the farfield give us the values $f(x)$. Suppose now that we are able to take measurements only for limited values of x, say for $|x| \leq A$; then $2A$ is the *aperture* of our antenna or array of sensors. We describe this by saying that we have available measurements of $f(x)h(x)$, where $h(x) = \chi_A(x) = 1$, for $|x| \leq A$, and zero otherwise. So, in addition to describing blurring and low-pass filtering, the convolution-filter model can also be used to model the limited-aperture problem. As in the low-pass case, the limited-aperture problem can be attacked using extrapolation, but with the same sort of risks described for the low-pass case. A much different approach is to increase the aperture by physically moving the array of sensors, as in *synthetic aperture radar* (SAR).

Returning to the farfield remote-sensing model, if we have Fourier transform data only for $|x| \leq A$, then we have $f(x)$ for $|x| \leq A$. Using $h(x) = \chi_A(x)$ to describe the limited aperture of the system, the point-spread function is $H(\gamma) = 2A\text{sinc}(\gamma A)$, the Fourier transform of $h(x)$. The first zeros of the numerator occur at $|\gamma| = \frac{\pi}{A}$, so the main lobe of the point-spread function has width $\frac{2\pi}{A}$. For this reason, the resolution of such a limited-aperture imaging system is said to be on the order of $\frac{1}{A}$. Since $|k| \leq \frac{\omega}{c}$, we can write $k = \frac{\omega}{c}\cos\theta$, where θ denotes the angle between the positive x-axis and the vector $\mathbf{k} = (k_1, k_2, 0)$; that is, θ points in the direction of the point P associated with the wavevector $\mathbf{k}$. The resolution, as measured by the width of the main lobe of the point-spread function $H(\gamma)$, in units of k, is $\frac{2\pi}{A}$, but, the angular resolution will depend also on the frequency ω. Since $k = \frac{2\pi}{\lambda}\cos\theta$, a distance of one unit in k may correspond to a large change in θ when ω is small, but only to a relatively small change in θ when ω is large. For this reason, the aperture of the array is usually measured in units of the wavelength; an aperture of $A = 5$ meters may be acceptable if the frequency is high, so that the wavelength is small, but not if the radiation is in the one-meter-wavelength range.

25.12 Resolution

If $F(k) = \delta(k)$ and $h(x) = \chi_A(x)$ describes the aperture-limitation of the imaging system, then the point-spread function is $H(\gamma) = 2A\text{sinc}(\gamma A)$. The maximum of $H(\gamma)$ still occurs at $\gamma = 0$, but the main lobe of $H(\gamma)$ extends from $-\frac{\pi}{A}$ to $\frac{\pi}{A}$; the point source has been spread out. If the point-source object shifts, so that $F(k) = \delta(k - a)$, then the reconstructed image of the object is $H(k - a)$, so the peak is still in the proper place. If we know *a priori* that the object is a single point source, but we do not know its location, the spreading of the point poses no problem; we simply look for

the maximum in the reconstructed image. Problems arise when the object contains several point sources, or when we do not know *a priori* what we are looking at, or when the object contains no point sources, but is just a continuous distribution.

Suppose that $F(k) = \delta(k-a) + \delta(k-b)$; that is, the object consists of two point sources. Then Fourier transformation of the aperture-limited data leads to the reconstructed image

$$R(k) = 2A\Big(\text{sinc}(A(k-a)) + \text{sinc}(A(k-b))\Big). \tag{25.20}$$

If $|b-a|$ is large enough, $R(k)$ will have two distinct maxima, at approximately $k=a$ and $k=b$, respectively. For this to happen, we need π/A, the width of the main lobe of the function $\text{sinc}(Ak)$, to be less than $|b-a|$. In other words, to resolve the two point sources a distance $|b-a|$ apart, we need $A \geq \pi/|b-a|$. However, if $|b-a|$ is too small, the distinct maxima merge into one, at $k = \frac{a+b}{2}$ and resolution will be lost. How small is too small will depend on both A and ω.

Suppose now that $F(k) = \delta(k-a)$, but we do not know *a priori* that the object is a single point source. We calculate

$$R(k) = H(k-a) = 2A\text{sinc}(A(k-a)) \tag{25.21}$$

and use this function as our reconstructed image of the object, for all k. What we see when we look at $R(k)$ for some $k = b \neq a$ is $R(b)$, which is the same thing we see when the point source is at $k=b$ and we look at $k=a$. Point-spreading is, therefore, more than a cosmetic problem. When the object is a point source at $k=a$, but we do not know *a priori* that it is a point source, the spreading of the point causes us to believe that the object function $F(k)$ is nonzero at values of k other than $k=a$. When we look at, say, $k=b$, we see a nonzero value that is caused by the presence of the point source at $k=a$.

Suppose now that the object function $F(k)$ contains no point sources, but is simply an ordinary function of k. If the aperture A is very small, then the function $H(k)$ is nearly constant over the entire extent of the object. The convolution of $F(k)$ and $H(k)$ is essentially the integral of $F(k)$, so the reconstructed object is $R(k) = \int F(k)dk$, for all k.

Let's see what this means for the solar-emission problem discussed earlier.

25.12.1 The Solar-Emission Problem Revisited

The wavelength of the radiation is $\lambda = 1$ meter. Therefore, $\frac{\omega}{c} = 2\pi$, and k in the interval $[-2\pi, 2\pi]$ corresponds to the angle θ in $[0, \pi]$. The sun has an angular diameter of 30 minutes of arc, which is about 10^{-2} radians. Therefore, the sun subtends the angles θ in $[\frac{\pi}{2} - (0.5)\cdot 10^{-2}, \frac{\pi}{2} + (0.5)\cdot 10^{-2}]$,

which corresponds roughly to the variable k in the interval $[-3 \cdot 10^{-2}, 3 \cdot 10^{-2}]$. Resolution of 3 minutes of arc means resolution in the variable k of $3 \cdot 10^{-3}$. If the aperture is $2A$, then to achieve this resolution, we need

$$\frac{\pi}{A} \leq 3 \cdot 10^{-3}, \tag{25.22}$$

or

$$A \geq \frac{\pi}{3} \cdot 10^{3} \tag{25.23}$$

meters, or A not less than about 1000 meters.

The radio-wave signals emitted by the sun are focused, using a parabolic radio-telescope. The telescope is pointed at the center of the sun. Because the sun is a great distance from the earth and the subtended arc is small (30 min.), the signals from each point on the sun's surface arrive at the parabola nearly head-on, that is, parallel to the line from the vertex to the focal point, and are reflected to the receiver located at the focal point of the parabola. The effect of the parabolic antenna is not to discriminate against signals coming from other directions, since there are none, but to effect a summation of the signals received at points $(x, 0, 0)$, for $|x| \leq A$, where $2A$ is the diameter of the parabola. When the aperture is large, the function $h(x)$ is nearly one for all x and the signal received at the focal point is essentially

$$\int f(x)dx = F(0); \tag{25.24}$$

we are now able to distinguish between $F(0)$ and other values $F(k)$. When the aperture is small, $h(x)$ is essentially $\delta(x)$ and the signal received at the focal point is essentially

$$\int f(x)\delta(x)dx = f(0) = \int F(k)dk; \tag{25.25}$$

now all we get is the contribution from all the k, superimposed, and all resolution is lost.

Since the solar emission problem is clearly two-dimensional, and we need 3 min. resolution in both dimensions, it would seem that we would need a circular antenna with a diameter of about one kilometer, or a rectangular antenna roughly one kilometer on a side. We shall return to this problem later, once when we discuss multi-dimensional Fourier transforms, and then again when we consider tomographic reconstruction of images from line integrals.

25.13 Discrete Data

A familiar topic in signal processing is the passage from functions of continuous variables to discrete sequences. This transition is achieved by *sam-*

pling, that is, extracting values of the continuous-variable function at discrete points in its domain. Our example of farfield propagation can be used to explore some of the issues involved in sampling.

Imagine an infinite *uniform line array* of sensors formed by placing receivers at the points $(n\Delta, 0, 0)$, for some $\Delta > 0$ and all integers n. Then our data are the values $f(n\Delta)$. Because we defined $k = \frac{\omega}{c}\cos\theta$, it is clear that the function $F(k)$ is zero for k outside the interval $[-\frac{\omega}{c}, \frac{\omega}{c}]$.

Our discrete array of sensors cannot distinguish between the signal arriving from θ and a signal with the same amplitude, coming from an angle α with

$$\frac{\omega}{c}\cos\alpha = \frac{\omega}{c}\cos\theta + \frac{2\pi}{\Delta}m, \tag{25.26}$$

where m is an integer. To resolve this ambiguity, we select $\Delta > 0$ so that

$$-\frac{\omega}{c} + \frac{2\pi}{\Delta} \geq \frac{\omega}{c}, \tag{25.27}$$

or

$$\Delta \leq \frac{\pi c}{\omega} = \frac{\lambda}{2}. \tag{25.28}$$

The sensor spacing $\Delta_s = \frac{\lambda}{2}$ is the *Nyquist spacing*.

In the sunspot example, the object function $F(k)$ is zero for k outside of an interval much smaller than $[-\frac{\omega}{c}, \frac{\omega}{c}]$. Knowing that $F(k) = 0$ for $|k| > K$, for some $0 < K < \frac{\omega}{c}$, we can accept ambiguities that confuse θ with another angle that lies outside the angular diameter of the object. Consequently, we can redefine the Nyquist spacing to be

$$\Delta_s = \frac{\pi}{K}. \tag{25.29}$$

This tells us that when we are imaging a distant object with a small angular diameter, the Nyquist spacing is greater than $\frac{\lambda}{2}$. If our sensor spacing has been chosen to be $\frac{\lambda}{2}$, then we have *oversampled*. In the oversampled case, band-limited extrapolation methods can be used to improve resolution .

25.13.1 Reconstruction from Samples

From the data gathered at our infinite array we have extracted the Fourier transform values $f(n\Delta)$, for all integers n. The obvious question is whether or not the data is sufficient to reconstruct $F(k)$. We know that, to avoid ambiguity, we must have $\Delta \leq \frac{\pi c}{\omega}$. The good news is that, provided this condition holds, $F(k)$ is uniquely determined by this data and formulas exist for reconstructing $F(k)$ from the data; this is the content of the *Shannon's Sampling Theorem*. Of course, this is only of theoretical interest, since we never have infinite data. Nevertheless, a considerable amount of traditional signal-processing exposition makes use of this infinite-sequence model. The real problem, of course, is that our data is always finite.

25.14 The Finite-Data Problem

Suppose that we build a *uniform line array* of sensors by placing receivers at the points $(n\Delta, 0, 0)$, for some $\Delta > 0$ and $n = -N, ..., N$. Then our data are the values $f(n\Delta)$, for $n = -N, ..., N$. Suppose, as previously, that the object of interest, the function $F(k)$, is nonzero only for values of k in the interval $[-K, K]$, for some $0 < K < \frac{\omega}{c}$. Once again, we must have $\Delta \leq \frac{\pi c}{\omega}$ to avoid ambiguity; but this is not enough, now. The finite Fourier data is no longer sufficient to determine a unique $F(k)$. The best we can hope to do is to estimate the true $F(k)$, using both our measured Fourier data and whatever prior knowledge we may have about the function $F(k)$, such as where it is nonzero, if it consists of Dirac delta point sources, or if it is nonnegative. The data is also noisy, and that must be accounted for in the reconstruction process.

In certain applications, such as sonar array processing, the sensors are not necessarily arrayed at equal intervals along a line, or even at the grid points of a rectangle, but in an essentially arbitrary pattern in two, or even three, dimensions. In such cases, we have values of the Fourier transform of the object function, but at essentially arbitrary values of the variable. How best to reconstruct the object function in such cases is not obvious.

25.15 Functions of Several Variables

Fourier transformation applies, as well, to functions of several variables. As in the one-dimensional case, we can motivate the multi-dimensional Fourier transform using the farfield propagation model. As we noted earlier, the solar emission problem is inherently a two-dimensional problem.

25.15.1 Two-Dimensional Farfield Object

Assume that our sensors are located at points $\mathbf{s} = (x, y, 0)$ in the x,y-plane. As discussed previously, we assume that the function $F(\mathbf{k})$ can be viewed as a function $F(k_1, k_2)$. Since, in most applications, the distant object has a small angular diameter when viewed from a great distance - the sun's is only 30 minutes of arc - the function $F(k_1, k_2)$ will be supported on a small subset of vectors (k_1, k_2).

25.15.2 Limited Apertures in Two Dimensions

Suppose we have the values of the Fourier transform, $f(x, y)$, for $|x| \leq A$ and $|y| \leq A$. We describe this limited-data problem using the function $h(x, y)$ that is one for $|x| \leq A$, and $|y| \leq A$, and zero, otherwise. Then the

point-spread function is the Fourier transform of this $h(x, y)$, given by

$$H(\alpha, \beta) = 4AB\text{sinc}(A\alpha)\text{sinc}(B\beta). \tag{25.30}$$

The resolution in the horizontal (x) direction is on the order of $\frac{1}{A}$, and $\frac{1}{B}$ in the vertical, where, as in the one-dimensional case, aperture is best measured in units of wavelength.

Suppose our aperture is circular, with radius A. Then we have Fourier transform values $f(x, y)$ for $\sqrt{x^2 + y^2} \leq A$. Let $h(x, y)$ equal one, for $\sqrt{x^2 + y^2} \leq A$, and zero, otherwise. Then the point-spread function of this limited-aperture system is the Fourier transform of $h(x, y)$, given by $H(\alpha, \beta) = \frac{2\pi A}{r} J_1(rA)$, with $r = \sqrt{\alpha^2 + \beta^2}$. The resolution of this system is roughly the distance from the origin to the first null of the function $J_1(rA)$, which means that $rA = 4$, roughly.

For the solar emission problem, this says that we would need a circular aperture with radius approximately one kilometer to achieve 3 minutes of arc resolution. But this holds only if the antenna is stationary; a moving antenna is different! The solar emission problem was solved by using a rectangular antenna with a large A, but a small B, and exploiting the rotation of the earth. The resolution is then good in the horizontal, but bad in the vertical, so that the imaging system discriminates well between two distinct vertical lines, but cannot resolve sources within the same vertical line. Because B is small, what we end up with is essentially the integral of the function $f(x, z)$ along each vertical line. By tilting the antenna, and waiting for the earth to rotate enough, we can get these integrals along any set of parallel lines. The problem then is to reconstruct $F(k_1, k_2)$ from such line integrals. This is also the main problem in tomography.

25.16 Broadband Signals

We have spent considerable time discussing the case of a distant point source or an extended object transmitting or reflecting a single-frequency signal. If the signal consists of many frequencies, the so-called broadband case, we can still analyze the received signals at the sensors in terms of time delays, but we cannot easily convert the delays to phase differences, and thereby make good use of the Fourier transform. One approach is to filter each received signal, to remove components at all but a single frequency, and then to proceed as previously discussed. In this way we can process one frequency at a time. The object now is described in terms of a function of both $\mathbf{k}$ and ω, with $F(\mathbf{k}, \omega)$ the complex amplitude associated with the wave vector $\mathbf{k}$ and the frequency ω. In the case of radar, the function $F(\mathbf{k}, \omega)$ tells us how the material at P reflects the radio waves at the various frequencies ω, and thereby gives information about the nature of the material making up the object near the point P.

There are times, of course, when we do not want to decompose a broadband signal into single-frequency components. A satellite reflecting a TV signal is a broadband point source. All we are interested in is receiving the broadband signal clearly, free of any other interfering sources. The direction of the satellite is known and the antenna is turned to face the satellite. Each location on the parabolic dish reflects the same signal. Because of its parabolic shape, the signals reflected off the dish and picked up at the focal point have exactly the same travel time from the satellite, so they combine coherently, to give us the desired TV signal.

Chapter 26

Appendix: Conjugate-Direction Methods

Finding the least-squares solution of a possibly inconsistent system of linear equations $Ax = b$ is equivalent to minimizing the quadratic function $f(x) = \frac{1}{2}||Ax - b||_2^2$ and so can be viewed within the framework of optimization. Iterative optimization methods can then be used to provide, or at least suggest, algorithms for obtaining the least-squares solution. The *conjugate gradient method* is one such method.

26.1 Iterative Minimization

Iterative methods for minimizing a real-valued function $f(x)$ over the vector variable x usually take the following form: having obtained x^{k-1}, a new direction vector d^k is selected, an appropriate scalar $\alpha_k > 0$ is determined and the next member of the iterative sequence is given by

$$x^k = x^{k-1} + \alpha_k d^k. \tag{26.1}$$

Ideally, one would choose the α_k to be the value of α for which the function $f(x^{k-1} + \alpha d^k)$ is minimized. It is assumed that the direction d^k is a *descent direction*; that is, for small positive α the function $f(x^{k-1} + \alpha d^k)$ is strictly decreasing. Finding the optimal value of α at each step of the iteration is difficult, if not impossible, in most cases, and approximate methods, using line searches, are commonly used.

Exercise 26.1 *Differentiate the function $f(x^{k-1}+\alpha d^k)$ with respect to the variable α to show that*

$$\nabla f(x^k) \cdot d^k = 0. \tag{26.2}$$

Since the gradient $\nabla f(x^k)$ is orthogonal to the previous direction vector d^k and also because $-\nabla f(x)$ is the direction of greatest decrease of $f(x)$, the choice of $d^{k+1} = -\nabla f(x^k)$ as the next direction vector is a reasonable one. With this choice we obtain Cauchy's *steepest descent method* [159]:

$$x^{k+1} = x^k - \alpha_{k+1} \nabla f(x^k).$$

The steepest descent method need not converge in general and even when it does, it can do so slowly, suggesting that there may be better choices for the direction vectors. For example, the Newton-Raphson method [169] employs the following iteration:

$$x^{k+1} = x^k - \nabla^2 f(x^k)^{-1} \nabla f(x^k),$$

where $\nabla^2 f(x)$ is the Hessian matrix for $f(x)$ at x. To investigate further the issues associated with the selection of the direction vectors, we consider the more tractable special case of quadratic optimization.

26.2 Quadratic Optimization

Let A be an arbitrary real I by J matrix. The linear system of equations $Ax = b$ need not have any solutions, and we may wish to find a least-squares solution $x = \hat{x}$ that minimizes

$$f(x) = \frac{1}{2}||b - Ax||_2^2. \tag{26.3}$$

The vector b can be written

$$b = A\hat{x} + \hat{w},$$

where $A^T\hat{w} = 0$ and a least squares solution is an exact solution of the linear system $Qx = c$, with $Q = A^TA$ and $c = A^Tb$. We shall assume that Q is invertible and there is a unique least squares solution; this is the typical case.

We consider now the iterative scheme described by Equation (26.1) for $f(x)$ as in Equation (26.3). For this $f(x)$ the gradient becomes

$$\nabla f(x) = Qx - c.$$

The optimal α_k for the iteration can be obtained in closed form.

Exercise 26.2 *Show that the optimal α_k is*

$$\alpha_k = \frac{r^k \cdot d^k}{d^k \cdot Qd^k}, \tag{26.4}$$

where $r^k = c - Qx^{k-1}$.

Exercise 26.3 *Let $||x||_Q^2 = x \cdot Qx$ denote the square of the Q-norm of x. Show that*

$$||\hat{x} - x^{k-1}||_Q^2 - ||\hat{x} - x^k||_Q^2 = (r^k \cdot d^k)^2 / d^k \cdot Qd^k \geq 0$$

for any direction vectors d^k.

If the sequence of direction vectors $\{d^k\}$ is completely general, the iterative sequence need not converge. However, if the set of direction vectors is finite and spans R^J and we employ them cyclically, convergence follows.

Theorem 26.1 *Let $\{d^1, ..., d^J\}$ be any finite set whose span is all of R^J. Let α_k be chosen according to Equation (26.4). Then, for $k = 0, 1, ...$, $j = k(\mathrm{mod}\, J) + 1$, and any x^0, the sequence defined by*

$$x^k = x^{k-1} + \alpha_k d^j$$

converges to the least squares solution.

Proof: The sequence $\{||\hat{x} - x^k||_Q^2\}$ is decreasing and, therefore, the sequence $\{(r^k \cdot d^k)^2 / d^k \cdot Qd^k$ must converge to zero. Therefore, the vectors x^k are bounded, and for each $j = 1, ..., J$, the subsequences $\{x^{mJ+j}$, $m = 0, 1, ...\}$ have cluster points, say $x^{*,j}$ with

$$x^{*,j} = x^{*,j-1} + \frac{(c - Qx^{*,j-1}) \cdot d^j}{d^j \cdot Qd^j} d^j.$$

Since

$$r^{mJ+j} \cdot d^j \to 0,$$

it follows that, for each $j = 1, ..., J$,

$$(c - Qx^{*,j}) \cdot d^j = 0.$$

Therefore,

$$x^{*,1} = ... = x^{*,J} = x^*$$

with $Qx^* = c$. Consequently, x^* is the least squares solution and the sequence $\{||x^* - x^k||_Q\}$ is decreasing. But a subsequence converges to zero; therefore, $\{||x^* - x^k||_Q\} \to 0$. This completes the proof. ∎

There is an interesting corollary to this theorem that pertains to a modified version of the ART algorithm. For $k = 0, 1, ...$ and $i = k(\text{mod } M) + 1$ and with the rows of A normalized to have length one, the ART iterative step is

$$x^{k+1} = x^k + (b_i - (Ax^k)_i)a^i,$$

where a^i is the ith column of A^T. When $Ax = b$ has no solutions, the ART algorithm does not converge to the least-squares solution; rather, it exhibits subsequential convergence to a limit cycle. However, using the previous theorem, we can show that the following modification of the ART, which we shall call the *least squares ART* (LS-ART), converges to the least-squares solution for every x^0:

$$x^{k+1} = x^k + \frac{r^{k+1} \cdot a^i}{a^i \cdot Qa^i} a^i.$$

In the quadratic case the steepest descent iteration has the form

$$x^k = x^{k-1} + \frac{r^k \cdot r^k}{r^k \cdot Qr^k} r^k.$$

We have the following result.

Theorem 26.2 *The steepest descent method converges to the least-squares solution.*

Proof: As in the proof of the previous theorem, we have

$$||\hat{x} - x^{k-1}||_Q^2 - ||\hat{x} - x^k||_Q^2 = (r^k \cdot d^k)^2 / d^k \cdot Qd^k \geq 0,$$

where now the direction vectors are $d^k = r^k$. So, the sequence $\{||\hat{x} - x^k||_Q^2\}$ is decreasing, and therefore the sequence $\{(r^k \cdot r^k)^2 / r^k \cdot Qr^k\}$ must converge to zero. The sequence $\{x^k\}$ is bounded; let x^* be a cluster point. It follows that $c - Qx^* = 0$, so that x^* is the least-squares solution $\hat{x}$. The rest of the proof follows as in the proof of the previous theorem. ∎

26.3 Conjugate Bases for R^J

If the set $\{v^1, ..., v^J\}$ is a basis for R^J, then any vector x in R^J can be expressed as a linear combination of the basis vectors; that is, there are real numbers $a_1, ..., a_J$ for which

$$x = a_1 v^1 + a_2 v^2 + ... + a_J v^J.$$

For each x the coefficients a_j are unique. To determine the a_j we write

$$x \cdot v^m = a_1 v^1 \cdot v^m + a_2 v^2 \cdot v^m + ... + a_J v^J \cdot v^m,$$

for $m = 1, ..., M$. Having calculated the quantities $x \cdot v^m$ and $v^j \cdot v^m$, we solve the resulting system of linear equations for the a_j.

If the set $\{u^1, ..., u^M\}$ is an orthogonal basis, that is, then $u^j \cdot u^m = 0$, unless $j = m$, then the system of linear equations is now trivial to solve. The solution is $a_j = x \cdot u^j / u^j \cdot u^j$, for each j. Of course, we still need to compute the quantities $x \cdot u^j$.

The least-squares solution of the linear system of equations $Ax = b$ is

$$\hat{x} = (A^T A)^{-1} A^T b = Q^{-1} c.$$

To express $\hat{x}$ as a linear combination of the members of an orthogonal basis $\{u^1, ..., u^J\}$ we need the quantities $\hat{x} \cdot u^j$, which usually means that we need to know $\hat{x}$ first. For a special kind of basis, a *Q-conjugate basis*, knowing $\hat{x}$ ahead of time is not necessary; we need only know Q and c. Therefore, we can use such a basis to find $\hat{x}$. This is the essence of the *conjugate gradient method* (CGM), in which we calculate a conjugate basis and, in the process, determine $\hat{x}$.

26.3.1 Conjugate Directions

From Equation (26.2) we have

$$(c - Qx^{k+1}) \cdot d^k = 0,$$

which can be expressed as

$$(\hat{x} - x^{k+1}) \cdot Qd^k = (\hat{x} - x^{k+1})^T Qd^k = 0.$$

Two vectors x and y are said to be *Q-orthogonal* (or *Q-conjugate*, or just *conjugate*), if $x \cdot Qy = 0$. So, the least-squares solution that we seek lies in a direction from x^{k+1} that is Q-orthogonal to d^k. This suggests that we can do better than steepest descent if we take the next direction to be Q-orthogonal to the previous one, rather than just orthogonal. This leads us to *conjugate direction methods*.

Exercise 26.4 *Say that the set $\{p^1, ..., p^n\}$ is a conjugate set for R^J if $p^i \cdot Qp^j = 0$ for $i \neq j$. Prove that a conjugate set that does not contain zero is linearly independent. Show that if $p^n \neq 0$ for $n = 1, ..., J$, then the least-squares vector $\hat{x}$ can be written as*

$$\hat{x} = a_1 p^1 + ... + a_J p^J,$$

with $a_j = c \cdot p^j / p^j \cdot Qp^j$ for each j. Hint: use the Q-inner product $\langle x, y \rangle_Q = x \cdot Qy$.

Therefore, once we have a conjugate basis, computing the least squares solution is trivial. Generating a conjugate basis can obviously be done using the standard Gram-Schmidt approach.

26.3.2 The Gram-Schmidt Method

Let $\{v^1, ..., v^J\}$ be a linearly independent set of vectors in the space R^M, where $J \leq M$. The Gram-Schmidt method uses the v^j to create an orthogonal basis $\{u^1, ..., u^J\}$ for the span of the v^j. Begin by taking $u^1 = v^1$. For $j = 2, ..., J$, let

$$u^j = v^j - \frac{u^1 \cdot v^j}{u^1 \cdot u^1} u^1 - ... - \frac{u^{j-1} \cdot v^j}{u^{j-1} \cdot u^{j-1}} u^{j-1}.$$

To apply this approach to obtain a conjugate basis, we would simply replace the dot products $u^k \cdot v^j$ and $u^k \cdot u^k$ with the Q-inner products, that is,

$$p^j = v^j - \frac{p^1 \cdot Qv^j}{p^1 \cdot Qp^1} p^1 - ... - \frac{p^{j-1} \cdot Qv^j}{p^{j-1} \cdot Qp^{j-1}} p^{j-1}. \tag{26.5}$$

Even though the Q-inner products can always be written as $x \cdot Qy = Ax \cdot Ay$, so that we need not compute the matrix Q, calculating a conjugate basis using Gram-Schmidt is not practical for large J. There is a way out, fortunately.

If we take $p^1 = v^1$ and $v^j = Qp^{j-1}$, we have a much more efficient mechanism for generating a conjugate basis, namely a three-term recursion formula [159]. The set $\{p^1, Qp^1, ..., Qp^{J-1}\}$ need not be a linearly independent set, in general, but, if our goal is to find $\hat{x}$, and not really to calculate a full conjugate basis, this does not matter, as we shall see.

Theorem 26.3 *Let $p^1 \neq 0$ be arbitrary. Let p^2 be given by*

$$p^2 = Qp^1 - \frac{Qp^1 \cdot Qp^1}{p^1 \cdot Qp^1} p^1,$$

so that $p^2 \cdot Qp^1 = 0$. Then, for $n \geq 2$, let p^{n+1} be given by

$$p^{n+1} = Qp^n - \frac{Qp^n \cdot Qp^n}{p^n \cdot Qp^n} p^n - \frac{Qp^{n-1} \cdot Qp^n}{p^{n-1} \cdot Qp^{n-1}} p^{n-1}. \tag{26.6}$$

Then, the set $\{p^1, ..., p^J\}$ is a conjugate set for R^J. If $p^n \neq 0$ for each n, then the set is a conjugate basis for R^J.

Proof: We consider the induction step of the proof. Assume that $\{p^1, ..., p^n\}$ is a Q-orthogonal set of vectors; we then show that $\{p^1, ..., p^{n+1}\}$ is also, provided that $n \leq J - 1$. It is clear from Equation (26.6) that

$$p^{n+1} \cdot Qp^n = p^{n+1} \cdot Qp^{n-1} = 0.$$

For $j \leq n-2$, we have

$$p^{n+1} \cdot Qp^j = p^j \cdot Qp^{n+1} = p^j \cdot Q^2 p^n - ap^j \cdot Qp^n - bp^j \cdot Qp^{n-1},$$

for constants a and b. The second and third terms on the right side are then zero because of the induction hypothesis. The first term is also zero since

$$p^j \cdot Q^2 p^n = (Qp^j) \cdot Qp^n = 0$$

because Qp^j is in the span of $\{p^1, ..., p^{j+1}\}$, and so is Q-orthogonal to p^n. ∎

The calculations in the three-term recursion formula Equation (26.6) also occur in the Gram-Schmidt approach in Equation (26.5); the point is that Equation (26.6) uses only the first three terms, in every case.

26.4 The Conjugate Gradient Method

The main idea in the *conjugate gradient method* (CGM) is to build the conjugate set as we calculate the least squares solution using the iterative algorithm

$$x^n = x^{n-1} + \alpha_n p^n. \tag{26.7}$$

The α_n is chosen so as to minimize the function of α defined by $f(x^{n-1} + \alpha p^n)$, and so we have

$$\alpha_n = \frac{r^n \cdot p^n}{p^n \cdot Qp^n},$$

where $r^n = c - Qx^{n-1}$. Since the function $f(x) = \frac{1}{2}||Ax - b||_2^2$ has for its gradient $\nabla f(x) = A^T(Ax - b) = Qx - c$, the residual vector $r^n = c - Qx^{n-1}$ is the direction of steepest descent from the point $x = x^{n-1}$. The CGM combines the use of the negative gradient directions from the steepest descent method with the use of a conjugate basis of directions, by using the r^{n+1} to construct the next direction p^{n+1} in such a way as to form a conjugate set $\{p_1, ..., p^J\}$.

As before, there is an efficient recursive formula that provides the next direction: let $p^1 = r^1 = (c - Qx^0)$ and

$$p^{n+1} = r^{n+1} - \frac{r^{n+1} \cdot Qp^n}{p^n \cdot Qp^n} p^n. \tag{26.8}$$

Since the α_n is the optimal choice and

$$r^{n+1} = -\nabla f(x^n),$$

we have, according to Equation (26.2),

$$r^{n+1} \cdot p^n = 0.$$

Exercise 26.5 *Prove that $r^{n+1} = 0$ whenever $p^{n+1} = 0$, in which case we have $c = Qx^n$, so that x^n is the least-squares solution.*

In theory, the CGM converges to the least squares solution in finitely many steps, since we either reach $p^{n+1} = 0$ or $n + 1 = J$. In practice, the CGM can be employed as a fully iterative method by cycling back through the previously used directions.

An induction proof similar to the one used to prove Theorem 26.3 establishes that the set $\{p^1, ..., p^J\}$ is a conjugate set [159, 169]. In fact, we can say more.

Theorem 26.4 *For $n = 1, 2, ..., J$ and $j = 1, ..., n-1$ we have a) $r^n \cdot r^j = 0$; b) $r^n \cdot p^j = 0$; and c) $p^n \cdot Qp^j = 0$.*

The proof presented here through a series of exercises is based on that given in [169].

The proof uses induction on the number n. Throughout the following exercises assume that the statements in the theorem hold for some $n < J$. We prove that they hold also for $n + 1$.

Exercise 26.6 *Use the fact that*

$$r^{j+1} = r^j - \alpha_j Qp^j,$$

to show that Qp^j is in the span of the vectors r^j and r^{j+1}.

Exercise 26.7 *Show that $r^{n+1} \cdot r^n = 0$. Hint: establish that*

$$\alpha_n = \frac{r^n \cdot r^n}{p^n \cdot Qp^n}.$$

Exercise 26.8 *Show that $r^{n+1} \cdot r^j = 0$, for $j = 1, ..., n - 1$. Hint: use the induction hypothesis.*

Exercise 26.9 *Show that $r^{n+1} \cdot p^j = 0$, for $j = 1, ..., n$. Hint: first, establish that*

$$p^j = r^j - \beta_{j-1} p^{j-1},$$

where

$$\beta_{j-1} = \frac{r^j \cdot Qp^{j-1}}{p^{j-1} \cdot Qp^{j-1}},$$

and

$$r^{n+1} = r^n - \alpha_n Qp^n.$$

Exercise 26.10 *Show that $p^{n+1} \cdot Qp^j = 0$, for $j = 1, ..., n - 1$. Hint: use*

$$Qp^j = \alpha_j^{-1}(r^j - r^{j+1}).$$

The final step in the proof is contained in the following exercise.

Exercise 26.11 *Show that* $p^{n+1} \cdot Qp^n = 0$. *Hint: establish that*

$$\beta_n = -\frac{r^{n+1} \cdot r^{n+1}}{r^n \cdot r^n}.$$

The convergence rate of the CGM depends on the condition number of the matrix Q, which is the ratio of its largest to its smallest eigenvalues. When the condition number is much greater than one convergence can be accelerated by *preconditioning* the matrix Q; this means replacing Q with $P^{-1/2}QP^{-1/2}$, for some positive-definite approximation P of Q (see [6]).

There are versions of the CGM for the minimization of nonquadratic functions. In the quadratic case the next conjugate direction p^{n+1} is built from the residual r^{n+1} and p^n. Since, in that case, $r^{n+1} = -\nabla f(x^n)$, this suggests that in the nonquadratic case we build p^{n+1} from $-\nabla f(x^n)$ and p^n. This leads to the Fletcher-Reeves method. Other similar algorithms, such as the Polak-Ribiere and the Hestenes-Stiefel methods, perform better on certain problems [169].

Chapter 27

Appendix: Matrix Theory

27.1 Matrix Inverses

A square matrix A is said to have inverse A^{-1} provided that

$$AA^{-1} = A^{-1}A = I,$$

where I is the identity matrix. The 2 by 2 matrix $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ has an inverse

$$A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

whenever the *determinant* of A, $\det(A) = ad - bc$ is not zero. More generally, associated with every complex square matrix is the complex number called its determinant, which is obtained from the entries of the matrix using formulas that can be found in any text on linear algebra. The significance of the determinant is that the matrix is invertible if and only if its determinant is not zero. This is of more theoretical than practical importance, since no computer can tell when a number is precisely zero. A matrix A that is not square cannot have an inverse, but does have a *pseudo-inverse*, which is found using the singular-value decomposition.

27.2 Basic Linear Algebra

In this section we discuss systems of linear equations, Gaussian elimination, and the notions of basic and non-basic variables.

27.2.1 Bases and Dimension

The notions of a basis and of linear independence are fundamental in linear algebra. Let $\mathcal{V}$ be a vector space.

Definition 27.1 *A collection of vectors $\{u^1, ..., u^N\}$ in $\mathcal{V}$ is linearly independent if there is no choice of scalars $\alpha_1, ..., \alpha_N$, not all zero, such that*

$$0 = \alpha_1 u^1 + ... + \alpha_N u^N. \tag{27.1}$$

Definition 27.2 *The span of a collection of vectors $\{u^1, ..., u^N\}$ in $\mathcal{V}$ is the set of all vectors x that can be written as linear combinations of the u^n; that is, for which there are scalars $c_1, ..., c_N$, such that*

$$x = c_1 u^1 + ... + c_N u^N. \tag{27.2}$$

Definition 27.3 *A collection of vectors $\{w^1, ..., w^N\}$ in $\mathcal{V}$ is called a spanning set for a subspace S if the set S is their span.*

Definition 27.4 *A collection of vectors $\{u^1, ..., u^N\}$ in $\mathcal{V}$ is called a basis for a subspace S if the collection is linearly independent and S is their span.*

Definition 27.5 *A collection of vectors $\{u^1, ..., u^N\}$ in $\mathcal{V}$ is called orthonormal if $||u^n||_2 = 1$, for all n, and $\langle u^m, u^n \rangle = 0$, for $m \neq n$.*

Suppose that S is a subspace of $\mathcal{V}$, that $\{w^1, ..., w^N\}$ is a spanning set for S, and $\{u^1, ..., u^M\}$ is a linearly independent subset of S. Beginning with w_1, we augment the set $\{u^1, ..., u^M\}$ with w_j if w_j is not in the span of the u_m and the w_k previously included. At the end of this process, we have a linearly independent spanning set, and therefore, a basis, for S (Why?). Similarly, beginning with w_1, we remove w_j from the set $\{w^1, ..., w^N\}$ if w_j is a linear combination of the w_k, $k = 1, ..., j-1$. In this way we obtain a linearly independent set that spans S, hence another basis for S. The following lemma will allow us to prove that all bases for a subspace S have the same number of elements.

Lemma 27.1 *Let $W = \{w^1, ..., w^N\}$ be a spanning set for a subspace S in R^I, and $V = \{v^1, ..., v^M\}$ a linearly independent subset of S. Then $M \leq N$.*

Proof: Suppose that $M > N$. Let $B_0 = \{w^1, ..., w^N\}$. To obtain the set B_1, form the set $C_1 = \{v_1, w_1, ..., w_N\}$ and remove the first member of C_1 that is a linear combination of members of C_1 that occur to its left in the listing; since v_1 has no members to its left, it is not removed. Since W is a spanning set, v_1 is a linear combination of the members of W, so that some member of W is a linear combination of v_1 and the remaining members of W; remove the first member of W for which this is true.

We note that the set B_1 is a spanning set for S and has N members. Having obtained the spanning set B_k, with N members and whose first k members are $v_k, ..., v_1$, we form the set $C_{k+1} = B_k \cup \{v_{k+1}\}$, listing the members so that the first $k+1$ of them are $\{v_{k+1}, v_k, ..., v_1\}$. To get the set

B_{k+1} we remove the first member of C_{k+1} that is a linear combination of the members to its left; there must be one, since B_k is a spanning set, and so v_{k+1} is a linear combination of the members of B_k. Since the set V is linearly independent, the member removed is from the set W. Continuing in this fashion, we obtain a sequence of spanning sets $B_1, ..., B_N$, each with N members. The set B_N is $B_N = \{v_1, ..., v_N\}$ and v_{N+1} must then be a linear combination of the members of B_N, which contradicts the linear independence of V. ∎

Corollary 27.1 *Every basis for a subspace S has the same number of elements.*

Definition 27.6 *The dimension of a subspace S is the number of elements in any basis.*

Lemma 27.2 *For any matrix A, the number of linearly independent rows equals the number of linearly independent columns.*

Proof: See Exercise 27.2.

Definition 27.7 *The rank of A is the number of linearly independent rows or of linearly independent columns of A.*

Exercise 27.1 *Let $W = \{w^1, ..., w^N\}$ be a spanning set for a subspace S in R^I, and $V = \{v^1, ..., v^M\}$ a linearly independent subset of S. Then, according to Lemma 27.1, $M \leq N$. Let A be the I by M matrix whose columns are the vectors v_m and B the I by N matrix whose columns are the w_n. Since W is a spanning set for S, there is an N by M matrix C such that $A = BC$. Prove Lemma 27.1 by considering the space of solutions of the system $Ax = 0$.*

Exercise 27.2 *Prove Lemma 27.2. Hints: Suppose that A is an I by J matrix, and that the row space of A, that is, the subspace $RS(A)$ of R^I spanned by the columns of A, has dimension K, for some $K \leq J$. Show that there is an I by K matrix U and a K by J matrix M such that $A = UM$. Use $A^T = M^T U^T$ to show that the column space of A, the subspace $CS(A)$ of R^J spanned by the rows of A, has a spanning set with K members. Conclude that the dimensions of $RS(A)$ and $CS(A)$ are the same; this number is the rank of A.*

27.2.2 Systems of Linear Equations

Consider the system of three linear equations in five unknowns given by

$$\begin{array}{rrrrrl} x_1 & +2x_2 & & +2x_4 & +x_5 & = 0 \\ -x_1 & -x_2 & +x_3 & +x_4 & & = 0\,. \\ x_1 & +2x_2 & -3x_3 & -x_4 & -2x_5 & = 0 \end{array} \tag{27.3}$$

This system can be written in matrix form as $Ax = 0$, with A the coefficient matrix

$$A = \begin{bmatrix} 1 & 2 & 0 & 2 & 1 \\ -1 & -1 & 1 & 1 & 0 \\ 1 & 2 & -3 & -1 & -2 \end{bmatrix}, \tag{27.4}$$

and $x = (x_1, x_2, x_3, x_4, x_5)^T$. Applying Gaussian elimination to this system, we obtain a second, simpler, system with the same solutions:

$$\begin{array}{rrrrrr} x_1 & & & -2x_4 & +x_5 & = 0 \\ & x_2 & & +2x_4 & & = 0\,. \\ & & x_3 & +x_4 & +x_5 & = 0 \end{array} \tag{27.5}$$

From this simpler system we see that the variables x_4 and x_5 can be freely chosen, with the other three variables then determined by this system of equations. The variables x_4 and x_5 are then independent, the others dependent. The variables x_1, x_2 and x_3 are then called *basic variables*. To obtain a basis of solutions we can let $x_4 = 1$ and $x_5 = 0$, obtaining the solution $x = (2, -2, -1, 1, 0)^T$, and then choose $x_4 = 0$ and $x_5 = 1$ to get the solution $x = (-1, 0, -1, 0, 1)^T$. Every solution to $Ax = 0$ is then a linear combination of these two solutions. Notice that which variables are basic and which are non-basic is somewhat arbitrary, in that we could have chosen as the non-basic variables any two whose columns are independent.

Having decided that x_4 and x_5 are the non-basic variables, we can write the original matrix A as $A = [\,B \quad N\,]$, where B is the square invertible matrix

$$B = \begin{bmatrix} 1 & 2 & 0 \\ -1 & -1 & 1 \\ 1 & 2 & -3 \end{bmatrix}, \tag{27.6}$$

and N is the matrix

$$N = \begin{bmatrix} 2 & 1 \\ 1 & 0 \\ -1 & -2 \end{bmatrix}. \tag{27.7}$$

With $x_B = (x_1, x_2, x_3)^T$ and $x_N = (x_4, x_5)^T$ we can write

$$Ax = Bx_B + Nx_N = 0, \tag{27.8}$$

so that

$$x_B = -B^{-1}Nx_N. \tag{27.9}$$

27.2.3 Real and Complex Systems of Linear Equations

A system $Ax = b$ of linear equations is called a *complex system*, or a *real system* if the entries of A, x and b are complex, or real, respectively. For any matrix A, we denote by A^T and $A^\dagger$ the transpose and conjugate transpose of A, respectively.

Any complex system can be converted to a real system in the following way. A complex matrix A can be written as $A = A_1 + iA_2$, where A_1 and A_2 are real matrices and $i = \sqrt{-1}$. Similarly, $x = x^1 + ix^2$ and $b = b^1 + ib^2$, where x^1, x^2, b^1 and b^2 are real vectors. Denote by $\tilde{A}$ the real matrix

$$\tilde{A} = \begin{bmatrix} A_1 & -A_2 \\ A_2 & A_1 \end{bmatrix}, \tag{27.10}$$

by $\tilde{x}$ the real vector

$$\tilde{x} = \begin{bmatrix} x^1 \\ x^2 \end{bmatrix}, \tag{27.11}$$

and by $\tilde{b}$ the real vector

$$\tilde{b} = \begin{bmatrix} b^1 \\ b^2 \end{bmatrix}. \tag{27.12}$$

Then x satisfies the system $Ax = b$ if and only if $\tilde{x}$ satisfies the system $\tilde{A}\tilde{x} = \tilde{b}$.

Definition 27.8 *A square matrix A is symmetric if $A^T = A$ and Hermitian if $A^\dagger = A$.*

Definition 27.9 *A non-zero vector x is said to be an eigenvector of the square matrix A if there is a scalar λ such that $Ax = \lambda x$. Then λ is said to be an eigenvalue of A.*

If x is an eigenvector of A with eigenvalue λ, then the matrix $A - \lambda I$ has no inverse, so its determinant is zero; here I is the identity matrix with ones on the main diagonal and zeros elsewhere. Solving for the roots of the determinant is one way to calculate the eigenvalues of A. For example, the eigenvalues of the Hermitian matrix

$$B = \begin{bmatrix} 1 & 2+i \\ 2-i & 1 \end{bmatrix} \tag{27.13}$$

are $\lambda = 1 + \sqrt{5}$ and $\lambda = 1 - \sqrt{5}$, with corresponding eigenvectors $u = (\sqrt{5}, 2 - i)^T$ and $v = (\sqrt{5}, i - 2)^T$, respectively. Then $\tilde{B}$ has the same eigenvalues, but both with multiplicity two. Finally, the associated eigenvectors of $\tilde{B}$ are

$$\begin{bmatrix} u^1 \\ u^2 \end{bmatrix}, \tag{27.14}$$

and

$$\begin{bmatrix} -u^2 \\ u^1 \end{bmatrix}, \tag{27.15}$$

for $\lambda = 1 + \sqrt{5}$, and

$$\begin{bmatrix} v^1 \\ v^2 \end{bmatrix}, \tag{27.16}$$

and

$$\begin{bmatrix} -v^2 \\ v^1 \end{bmatrix}, \tag{27.17}$$

for $\lambda = 1 - \sqrt{5}$.

27.3 Solutions of Under-determined Systems of Linear Equations

Suppose that $A\mathbf{x} = \mathbf{b}$ is a consistent linear system of M equations in N unknowns, where $M < N$. Then there are infinitely many solutions. A standard procedure in such cases is to find that solution $\mathbf{x}$ having the smallest norm

$$||\mathbf{x}|| = \sqrt{\sum_{n=1}^{N} |x_n|^2}.$$

As we shall see shortly, the *minimum norm* solution of $A\mathbf{x} = \mathbf{b}$ is a vector of the form $\mathbf{x} = A^\dagger \mathbf{z}$, where $A^\dagger$ denotes the conjugate transpose of the matrix A. Then $A\mathbf{x} = \mathbf{b}$ becomes $AA^\dagger \mathbf{z} = \mathbf{b}$. Typically, $(AA^\dagger)^{-1}$ will exist, and we get $\mathbf{z} = (AA^\dagger)^{-1}\mathbf{b}$, from which it follows that the minimum norm solution is $\mathbf{x} = A^\dagger(AA^\dagger)^{-1}\mathbf{b}$. When M and N are not too large, forming the matrix $AA^\dagger$ and solving for $\mathbf{z}$ is not prohibitively expensive and time-consuming. However, in image processing the vector $\mathbf{x}$ is often a vectorization of a two-dimensional (or even three-dimensional) image and M and N can be on the order of tens of thousands or more. The ART algorithm gives us a fast method for finding the minimum norm solution without computing $AA^\dagger$.

We begin by proving that the minimum norm solution of $A\mathbf{x} = \mathbf{b}$ has the form $\mathbf{x} = A^\dagger \mathbf{z}$ for some M-dimensional complex vector $\mathbf{z}$.

Let the *null space* of the matrix A be all N-dimensional complex vectors $\mathbf{w}$ with $A\mathbf{w} = \mathbf{0}$. If $A\mathbf{x} = \mathbf{b}$ then $A(\mathbf{x} + \mathbf{w}) = \mathbf{b}$ for all $\mathbf{w}$ in the null space of A. If $\mathbf{x} = A^\dagger \mathbf{z}$ and $\mathbf{w}$ is in the null space of A, then

$$||\mathbf{x} + \mathbf{w}||^2 = ||A^\dagger \mathbf{z} + \mathbf{w}||^2 = (A^\dagger \mathbf{z} + \mathbf{w})^\dagger (A^\dagger \mathbf{z} + \mathbf{w})$$

$$= (A^\dagger \mathbf{z})^\dagger (A^\dagger \mathbf{z}) + (A^\dagger \mathbf{z})^\dagger \mathbf{w} + \mathbf{w}^\dagger (A^\dagger \mathbf{z}) + \mathbf{w}^\dagger \mathbf{w}$$
$$= ||A^\dagger \mathbf{z}||^2 + (A^\dagger \mathbf{z})^\dagger \mathbf{w} + \mathbf{w}^\dagger (A^\dagger \mathbf{z}) + ||\mathbf{w}||^2$$
$$= ||A^\dagger \mathbf{z}||^2 + ||\mathbf{w}||^2,$$

since

$$\mathbf{w}^\dagger (A^\dagger \mathbf{z}) = (A\mathbf{w})^\dagger \mathbf{z} = \mathbf{0}^\dagger \mathbf{z} = 0$$

and

$$(A^\dagger \mathbf{z})^\dagger \mathbf{w} = \mathbf{z}^\dagger A\mathbf{w} = \mathbf{z}^\dagger \mathbf{0} = 0.$$

Therefore, $||\mathbf{x} + \mathbf{w}|| = ||A^\dagger \mathbf{z} + \mathbf{w}|| > ||A^\dagger \mathbf{z}|| = ||\mathbf{x}||$ unless $\mathbf{w} = \mathbf{0}$. This completes the proof.

Exercise 27.3 *Show that if* $\mathbf{z} = (z_1, ..., z_N)^T$ *is a column vector with complex entries and* $H = H^\dagger$ *is an* N *by* N *Hermitian matrix with complex entries then the quadratic form* $\mathbf{z}^\dagger H\mathbf{z}$ *is a real number. Show that the quadratic form* $\mathbf{z}^\dagger H\mathbf{z}$ *can be calculated using only real numbers. Let* $\mathbf{z} = \mathbf{x} + i\mathbf{y}$*, with* $\mathbf{x}$ *and* $\mathbf{y}$ *real vectors and let* $H = A + iB$*, where* A *and* B *are real matrices. Then show that* $A^T = A$*,* $B^T = -B$*,* $\mathbf{x}^T B\mathbf{x} = 0$ *and finally,*

$$\mathbf{z}^\dagger H\mathbf{z} = [\,\mathbf{x}^T \quad \mathbf{y}^T\,] \begin{bmatrix} A & -B \\ B & A \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}.$$

Use the fact that $\mathbf{z}^\dagger H\mathbf{z}$ *is real for every vector* $\mathbf{z}$ *to conclude that the eigenvalues of* H *are real.*

27.4 Eigenvalues and Eigenvectors

Given N by N complex matrix A, we say that a complex number λ is an *eigenvalue* of A if there is a nonzero vector $\mathbf{u}$ with $A\mathbf{u} = \lambda\mathbf{u}$. The column vector $\mathbf{u}$ is then called an *eigenvector* of A associated with eigenvalue λ; clearly, if $\mathbf{u}$ is an eigenvector of A, then so is $c\mathbf{u}$, for any constant $c \neq 0$. If λ is an eigenvalue of A, then the matrix $A - \lambda I$ fails to have an inverse, since $(A - \lambda I)\mathbf{u} = \mathbf{0}$ but $\mathbf{u} \neq \mathbf{0}$. If we treat λ as a variable and compute the determinant of $A - \lambda I$, we obtain a polynomial of degree N in λ. Its roots $\lambda_1, ..., \lambda_N$ are then the eigenvalues of A. If $||\mathbf{u}||^2 = \mathbf{u}^\dagger \mathbf{u} = 1$ then $\mathbf{u}^\dagger A\mathbf{u} = \lambda\mathbf{u}^\dagger \mathbf{u} = \lambda$.

It can be shown that it is possible to find a set of N mutually orthogonal eigenvectors of the Hermitian matrix H; call them $\{\mathbf{u}^1, ..., \mathbf{u}^N\}$. The matrix H can then be written as

$$H = \sum_{n=1}^{N} \lambda_n \mathbf{u}^n (\mathbf{u}^n)^\dagger,$$

a linear superposition of the *dyad* matrices $\mathbf{u}^n(\mathbf{u}^n)^\dagger$. We can also write $H = ULU^\dagger$, where U is the matrix whose nth column is the column vector $\mathbf{u}^n$ and L is the diagonal matrix with the eigenvalues down the main diagonal and zero elsewhere.

The matrix H is invertible if and only if none of the λ are zero and its inverse is

$$H^{-1} = \sum_{n=1}^{N} \lambda_n^{-1} \mathbf{u}^n (\mathbf{u}^n)^\dagger.$$

We also have $H^{-1} = UL^{-1}U^\dagger$.

A Hermitian matrix Q is said to be nonnegative-definite (positive-definite) if all the eigenvalues of Q are nonnegative (positive). The matrix Q is a nonnegative-definite matrix if and only if there is another matrix C such that $Q = C^\dagger C$. Since the eigenvalues of Q are nonnegative, the diagonal matrix L has a square root, $\sqrt{L}$. Using the fact that $U^\dagger U = I$, we have

$$Q = ULU^\dagger = U\sqrt{L}U^\dagger U\sqrt{L}U^\dagger;$$

we then take $C = U\sqrt{L}U^\dagger$, so $C^\dagger = C$. Then $\mathbf{z}^\dagger Q\mathbf{z} = \mathbf{z}^\dagger C^\dagger C\mathbf{z} = ||C\mathbf{z}||^2$, so that Q is positive-definite if and only if C is invertible.

Exercise 27.4 *Let A be an M by N matrix with complex entries. View A as a linear function with domain C^N, the space of all N-dimensional complex column vectors, and range contained within C^M, via the expression $A(\mathbf{x}) = A\mathbf{x}$. Suppose that $M > N$. The range of A, denoted $R(A)$, cannot be all of C^M. Show that every vector $\mathbf{z}$ in C^M can be written uniquely in the form $\mathbf{z} = A\mathbf{x} + \mathbf{w}$, where $A^\dagger\mathbf{w} = \mathbf{0}$. Show that $\|\mathbf{z}\|^2 = \|A\mathbf{x}\|^2 + \|\mathbf{w}\|^2$, where $\|\mathbf{z}\|^2$ denotes the square of the norm of $\mathbf{z}$.*

Hint: If $\mathbf{z} = A\mathbf{x} + \mathbf{w}$ then consider $A^\dagger\mathbf{z}$. Assume $A^\dagger A$ is invertible.

27.5 Vectorization of a Matrix

When the complex M by N matrix A is stored in the computer it is usually *vectorized*; that is, the matrix

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1N} \\ A_{21} & A_{22} & \dots & A_{2N} \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ A_{M1} & A_{M2} & \dots & A_{MN} \end{bmatrix}$$

becomes

$$\mathbf{vec}(A) = (A_{11}, A_{21}, ..., A_{M1}, A_{12}, A_{22}, ..., A_{M2}, ..., A_{MN})^T.$$

Exercise 27.5 **(a)** *Show that the complex dot product* $\mathbf{vec}(A)\cdot\mathbf{vec}(B) = \mathbf{vec}(B)^\dagger\mathbf{vec}(A)$ *can be obtained by*

$$\mathbf{vec}(A)\cdot\mathbf{vec}(B) = \text{trace}\,(AB^\dagger) = tr(AB^\dagger),$$

where, for a square matrix C, $\text{trace}\,(C)$ *means the sum of the entries along the main diagonal of* C. *We can therefore use the trace to define an inner product between matrices:* $< A, B >= \text{trace}\,(AB^\dagger)$.

(b) *Show that* $\text{trace}\,(AA^\dagger) \geq 0$ *for all* A, *so that we can use the trace to define a norm on matrices:* $||A||^2 = \text{trace}\,(AA^\dagger)$.

Exercise 27.6 *Let* $B = ULD^\dagger$ *be an* M *by* N *matrix in diagonalized form; that is,* L *is an* M *by* N *diagonal matrix with entries* $\lambda_1, ..., \lambda_K$ *on its main diagonal, where* $K = \min(M, N)$, *and* U *and* V *are square matrices. Let the* n*-th column of* U *be denoted* $\mathbf{u}^n$ *and similarly for the columns of* V. *Such a diagonal decomposition occurs in the singular value decomposition (SVD). Show that we can write*

$$B = \lambda_1\mathbf{u}^1(\mathbf{v}^1)^\dagger + ... + \lambda_K\mathbf{u}^K(\mathbf{v}^K)^\dagger.$$

If B is an N by N Hermitian matrix, then we can take $U = V$ and $K = M = N$, with the columns of U the eigenvectors of B, normalized to have Euclidean norm equal to one, and the λ_n to be the eigenvalues of B. In this case we may also assume that U is a *unitary* matrix; that is, $UU^\dagger = U^\dagger U = I$, where I denotes the identity matrix.

27.6 The Singular Value Decomposition (SVD)

We have just seen that an N by N Hermitian matrix H can be written in terms of its eigenvalues and eigenvectors as $H = ULU^\dagger$ or as

$$H = \sum_{n=1}^{N} \lambda_n\mathbf{u}^n(\mathbf{u}^n)^\dagger.$$

The *singular value decomposition* (SVD) is a similar result that applies to any rectangular matrix. It is an important tool in image compression and pseudo-inversion.

Let C be any N by K complex matrix. In presenting the SVD of C we shall assume that $K \geq N$; the SVD of $C^\dagger$ will come from that of C. Let $A = C^\dagger C$ and $B = CC^\dagger$; we assume, reasonably, that B, the smaller of the two matrices, is invertible, so all the eigenvalues $\lambda_1, ..., \lambda_N$ of B are positive. Then, write the eigenvalue/eigenvector decomposition of B as $B = ULU^\dagger$.

Exercise 27.7 *Show that the nonzero eigenvalues of A and B are the same.*

Let V be the K by K matrix whose first N columns are those of the matrix $C^\dagger UL^{-1/2}$ and whose remaining $K - N$ columns are any mutually orthogonal norm-one vectors that are all orthogonal to each of the first N columns. Let M be the N by K matrix with diagonal entries $M_{nn} = \sqrt{\lambda_n}$ for $n = 1, ..., N$ and whose remaining entries are zero. The nonzero entries of M, $\sqrt{\lambda_n}$, are called the *singular values* of C. The *singular value decomposition* (SVD) of C is $C = UMV^\dagger$. The SVD of $C^\dagger$ is $C^\dagger = VM^TU^\dagger$.

Exercise 27.8 *Show that $UMV^\dagger$ equals C.*

Using the SVD of C we can write

$$C = \sum_{n=1}^{N} \sqrt{\lambda_n} \mathbf{u}^n (\mathbf{v}^n)^\dagger,$$

where $\mathbf{v}^n$ denotes the nth column of the matrix V.

In image processing, matrices such as C are used to represent discrete two-dimensional images, with the entries of C corresponding to the grey level or color at each pixel. It is common to find that most of the N singular values of C are nearly zero, so that C can be written approximately as a sum of far fewer than N dyads; this is SVD image compression.

If $N \neq K$ then C cannot have an inverse; it does, however, have a *pseudo-inverse*, $C^* = VM^*U^\dagger$, where M^* is the matrix obtained from M by taking the inverse of each of its nonzero entries and leaving the remaining zeros the same. The pseudo-inverse of $C^\dagger$ is

$$(C^\dagger)^* = (C^*)^\dagger = U(M^*)^T V^\dagger = U(M^\dagger)^* V^\dagger.$$

Some important properties of the pseudo-inverse are the following:

1. $CC^*C = C$,

2. $C^*CC^* = C^*$,

3. $(C^*C)^\dagger = C^*C$,

4. $(CC^*)^\dagger = CC^*$.

The pseudo-inverse of an arbitrary I by J matrix G can be used in much the same way as the inverse of nonsingular matrices to find approximate or exact solutions of systems of equations $G\mathbf{x} = \mathbf{d}$. The following examples illustrate this point.

Exercise 27.9 *If $I > J$ the system $G\mathbf{x} = \mathbf{d}$ probably has no exact solution. Show that whenever $G^\dagger G$ is invertible the pseudo-inverse of G is $G^* = (G^\dagger G)^{-1}G^\dagger$ so that the vector $\mathbf{x} = G^*\mathbf{d}$ is the least squares approximate solution.*

Exercise 27.10 *If $I < J$ the system $G\mathbf{x} = \mathbf{d}$ probably has infinitely many solutions. Show that whenever the matrix $GG^\dagger$ is invertible the pseudo-inverse of G is $G^* = G^\dagger(GG^\dagger)^{-1}$, so that the vector $\mathbf{x} = G^*\mathbf{d}$ is the exact solution of $G\mathbf{x} = \mathbf{d}$ closest to the origin; that is, it is the minimum norm solution.*

27.7 Singular Values of Sparse Matrices

In image reconstruction from projections the M by N matrix A is usually quite large and often ϵ-sparse; that is, most of its elements do not exceed ϵ in absolute value, where ϵ denotes a small positive quantity. In transmission tomography each column of A corresponds to a single pixel in the digitized image, while each row of A corresponds to a line segment through the object, along which an x-ray beam has traveled. The entries of a given row of A are nonzero only for those columns whose associated pixel lies on that line segment; clearly, most of the entries of any given row of A will then be zero. In emission tomography the I by J nonnegative matrix P has entries $P_{ij} \geq 0$; for each detector i and pixel j, P_{ij} is the probability that an emission at the jth pixel will be detected at the ith detector. When a detection is recorded at the ith detector, we want the likely source of the emission to be one of only a small number of pixels. For single photon emission tomography (SPECT), a lead collimator is used to permit detection of only those photons approaching the detector straight on. In positron emission tomography (PET), coincidence detection serves much the same purpose. In both cases the probabilities P_{ij} will be zero (or

nearly zero) for most combinations of i and j. Such matrices are called *sparse* (or *almost sparse*). We discuss now a convenient estimate for the largest singular value of an almost sparse matrix A, which, for notational convenience only, we take to be real.

In [54] it was shown that if A is normalized so that each row has length one, then the spectral radius of A^TA, which is the square of the largest singular value of A itself, does not exceed the maximum number of nonzero elements in any column of A. A similar upper bound on $\rho(A^TA)$ can be obtained for non-normalized, ϵ-sparse A.

Let A be an M by N matrix. For each $n = 1, ..., N$, let $s_n > 0$ be the number of nonzero entries in the nth column of A, and let s be the maximum of the s_n. Let G be the M by N matrix with entries

$$G_{mn} = A_{mn}/(\sum_{l=1}^{N} s_l A_{ml}^2)^{1/2}.$$

Lent has shown that the eigenvalues of the matrix G^TG do not exceed one [156]. This result suggested the following proposition, whose proof was given in [54].

Proposition 27.1 *Let A be an M by N matrix. For each $m = 1, ..., M$ let $\nu_m = \sum_{n=1}^{N} A_{mn}^2 > 0$. For each $n = 1, ..., N$ let $\sigma_n = \sum_{m=1}^{M} e_{mn}\nu_m$, where $e_{mn} = 1$ if $A_{mn} \neq 0$ and $e_{mn} = 0$ otherwise. Let σ denote the maximum of the σ_n. Then the eigenvalues of the matrix A^TA do not exceed σ. If A is normalized so that the Euclidean length of each of its rows is one, then the eigenvalues of A^TA do not exceed s, the maximum number of nonzero elements in any column of A.*

Proof: For simplicity, we consider only the normalized case; the proof for the more general case is similar.

Let $A^TA\mathbf{v} = c\mathbf{v}$ for some nonzero vector $\mathbf{v}$. We show that $c \leq s$. We have $AA^TA\mathbf{v} = cA\mathbf{v}$ and so $\mathbf{w}^TAA^T\mathbf{w} = \mathbf{v}^TA^TAA^TA\mathbf{v} = c\mathbf{v}^TA^TA\mathbf{v} = c\mathbf{w}^T\mathbf{w}$, for $\mathbf{w} = A\mathbf{v}$. Then, with $e_{mn} = 1$ if $A_{mn} \neq 0$ and $e_{mn} = 0$ otherwise, we have

$$(\sum_{m=1}^{M} A_{mn}w_m)^2 = (\sum_{m=1}^{M} A_{mn}e_{mn}w_m)^2$$

$$\leq (\sum_{m=1}^{M} A_{mn}^2 w_m^2)(\sum_{m=1}^{M} e_{mn}^2) =$$

$$(\sum_{m=1}^{M} A_{mn}^2 w_m^2)s_j \leq (\sum_{m=1}^{M} A_{mn}^2 w_m^2)s.$$

Therefore,

$$\mathbf{w}^T AA^T \mathbf{w} = \sum_{n=1}^{N} (\sum_{m=1}^{M} A_{mn} w_m)^2 \leq \sum_{n=1}^{N} (\sum_{m=1}^{M} A_{mn}^2 w_m^2) s,$$

and

$$\mathbf{w}^T AA^T \mathbf{w} = c \sum_{m=1}^{M} w_m^2 = c \sum_{m=1}^{M} w_m^2 (\sum_{n=1}^{N} A_{mn}^2)$$

$$= c \sum_{m=1}^{M} \sum_{n=1}^{N} w_m^2 A_{mn}^2.$$

The result follows immediately. ∎

If we normalize A so that its rows have length one, then the trace of the matrix AA^T is $\mathrm{tr}(AA^T) = M$, which is also the sum of the eigenvalues of $A^T A$. Consequently, the maximum eigenvalue of $A^T A$ does not exceed M; this result improves that upper bound considerably, if A is sparse and so $s << M$.

In image reconstruction from projection data that includes scattering we often encounter matrices A most of whose entries are small, if not exactly zero. A slight modification of the proof provides us with a useful upper bound for L, the largest eigenvalue of $A^T A$, in such cases. Assume that the rows of A have length one. For $\epsilon > 0$ let s be the largest number of entries in any column of A whose magnitudes exceed ϵ. Then we have

$$L \leq s + MN\epsilon^2 + 2\epsilon(MNs)^{1/2}.$$

The proof of this result is similar to that for Proposition 27.1.

Chapter 28

Appendix: Constrained Iteration Methods

The ART and its simultaneous and block-iterative versions are designed to solve general systems of linear equations $Ax = b$. The SMART, EMML and RBI methods require that the entries of A be nonnegative, those of b positive and produce nonnegative x. In this chapter we present variations of the SMART and EMML that impose the constraints $u_j \leq x_j \leq v_j$, where the u_j and v_j are selected lower and upper bounds on the individual entries x_j. These algorithms were used in [168] as a method for including in transmission tomographic reconstruction spatially varying upper and lower bounds on the x-ray attenuation.

28.1 Modifying the KL distance

The SMART, EMML and RBI methods are based on the Kullback-Leibler distance between nonnegative vectors. To impose more general constraints on the entries of x we derive algorithms based on shifted KL distances, also called Fermi-Dirac generalized entropies.

For a fixed real vector u, the shifted KL distance $KL(x - u, z - u)$ is defined for vectors x and z having $x_j \geq u_j$ and $z_j \geq u_j$. Similarly, the shifted distance $KL(v - x, v - z)$ applies only to those vectors x and z for which $x_j \leq v_j$ and $z_j \leq v_j$. For $u_j \leq v_j$, the combined distance

$$KL(x - u, z - u) + KL(v - x, v - z)$$

is restricted to those x and z whose entries x_j and z_j lie in the interval $[u_j, v_j]$. Our objective is to mimic the derivation of the SMART, EMML and RBI methods, replacing KL distances with shifted KL distances, to obtain algorithms that enforce the constraints $u_j \leq x_j \leq v_j$, for each j.

The algorithms that result are the ABMART and ABEMML block-iterative methods. These algorithms were originally presented in [49], in which the vectors u and v were called a and b, hence the names of the algorithms. Throughout this chapter we shall assume that the entries of the matrix A are nonnegative. We shall denote by B_n, $n = 1, ..., N$ a partition of the index set $\{i = 1, ..., I\}$ into blocks. For $k = 0, 1, ...$ let $n(k) = k(\mathrm{mod}\, N)+1$.

The projected Landweber algorithm can also be used to impose the restrictions $u_j \leq x_j \leq v_j$; however, the projection step in that algorithm is implemented by clipping, or setting equal to u_j or v_j values of x_j that would otherwise fall outside the desired range. The result is that the values u_j and v_j can occur more frequently than may be desired. One advantage of the AB methods is that the values u_j and v_j represent barriers that can only be reached in the limit and are never taken on at any step of the iteration.

28.2 The ABMART Algorithm

We assume that $(Au)_i \leq b_i \leq (Av)_i$ and seek a solution of $Ax = b$ with $u_j \leq x_j \leq v_j$, for each j. The algorithm begins with an initial vector x^0 satisfying $u_j \leq x_j^0 \leq v_j$, for each j. Having calculated x^k, we take

$$x_j^{k+1} = \alpha_j^k v_j + (1 - \alpha_j^k) u_j, \tag{28.1}$$

with $n = n(k)$,

$$\alpha_j^k = \frac{c_j^k \prod^n (d_i^k)^{A_{ij}}}{1 + c_j^k \prod^n (d_i^k)^{A_{ij}}}, \tag{28.2}$$

$$c_j^k = \frac{(x_j^k - u_j)}{(v_j - x_j^k)}, \tag{28.3}$$

and

$$d_j^k = \frac{(b_i - (Au)_i)((Av)_i - (Ax^k)_i)}{((Av)_i - b_i)((Ax^k)_i - (Au)_i)}, \tag{28.4}$$

where $\prod^n$ denotes the product over those indices i in $B_{n(k)}$. Notice that, at each step of the iteration, x_j^k is a convex combination of the endpoints u_j and v_j, so that x_j^k lies in the interval $[u_j, v_j]$.

We have the following theorem concerning the convergence of the ABMART algorithm:

Theorem 28.1 *If there is a soluton of the system $Ax = b$ that satisfies the constraints $u_j \leq x_j \leq v_j$ for each j, then, for any N and any choice of the*

blocks B_n, the ABMART sequence converges to that constrained solution of $Ax = b$ for which the Fermi-Dirac generalized entropic distance from x to x^0,

$$KL(x - u, x^0 - u) + KL(v - x, v - x^0),$$

is minimized. If there is no constrained solution of $Ax = b$, then, for $N = 1$, the ABMART sequence converges to the minimizer of

$$KL(Ax - Au, b - Au) + KL(Av - Ax, Av - b)$$

for which

$$KL(x - u, x^0 - u) + KL(v - x, v - x^0)$$

is minimized.

The proof is similar to that for RBI-SMART and is found in [49].

28.3 The ABEMML Algorithm

We make the same assumptions as in the previous section. The iterative step of the ABEMML algorithm is

$$x_j^{k+1} = \alpha_j^k v_j + (1 - \alpha_j^k) u_j, \tag{28.5}$$

where

$$\alpha_j^k = \gamma_j^k / d_j^k, \tag{28.6}$$

$$\gamma_j^k = (x_j^k - u_j) e_j^k, \tag{28.7}$$

$$\beta_j^k = (v_j - x_j^k) f_j^k, \tag{28.8}$$

$$d_j^k = \gamma_j^k + \beta_j^k, \tag{28.9}$$

$$e_j^k = \left(1 - \sum_{i \in B_n} A_{ij}\right) + \sum_{i \in B_n} A_{ij} \left(\frac{b_i - (Au)_i}{(Ax^k)_i - (Au)_i}\right), \tag{28.10}$$

and

$$f_j^k = \left(1 - \sum_{i \in B_n} A_{ij}\right) + \sum_{i \in B_n} A_{ij} \left(\frac{(Av)_i - b_i}{(Av)_i - (Ax^k)_i}\right). \tag{28.11}$$

We have the following theorem concerning the convergence of the ABEMML algorithm:

Theorem 28.2 *If there is a soluton of the system $Ax = b$ that satisfies the constraints $u_j \leq x_j \leq v_j$ for each j, then, for any N and any choice of the blocks B_n, the ABEMML sequence converges to such a constrained solution of $Ax = b$. If there is no constrained solution of $Ax = b$, then, for $N = 1$, the ABMART sequence converges to a constrained minimizer of*

$$KL(Ax - Au, b - Au) + KL(Av - Ax, Av - b).$$

The proof is similar to that for RBI-EMML and is to be found in [49]. In contrast to the ABMART theorem, this is all we can say about the limits of the ABEMML sequences.

Open Question: How does the limit of the ABEMML iterative sequence depend, in the consistent case, on the choice of blocks, and, in general, on the choice of x^0?

Chapter 29

Appendix: Inverse Problems and the Laplace Transform

In farfield propagation problems, the measured data are often related to the desired object function by a Fourier transformation. The image reconstruction problem then became one of estimating a function from finitely many noisy values of its Fourier transform. In this chapter we consider two inverse problems involving the Laplace transform.

29.1 The Laplace Transform and the Ozone Layer

The example is taken from Twomey's book [205].

29.1.1 The Laplace Transform

The Laplace transform of the function $f(x)$ defined for $0 \leq x < +\infty$ is the function

$$\mathcal{F}(s) = \int_0^{+\infty} f(x)e^{-sx}dx. \tag{29.1}$$

29.1.2 Scattering of Ultraviolet Radiation

The sun emits ultraviolet (UV) radiation that enters the Earth's atmosphere at an angle θ_0 that depends on the sun's position, and with intensity $I(0)$. Let the x-axis be vertical, with $x = 0$ at the top of the atmosphere

and x increasing as we move down to the Earth's surface, at $x = X$. The intensity at x is given by

$$I(x) = I(0)e^{-kx/\cos\theta_0}. \tag{29.2}$$

Within the ozone layer, the amount of UV radiation scattered in the direction θ is given by

$$S(\theta, \theta_0)I(0)e^{-kx/\cos\theta_0}\Delta p, \tag{29.3}$$

where $S(\theta, \theta_0)$ is a known parameter, and Δp is the change in the pressure of the ozone within the infinitesimal layer $[x, x+\Delta x]$, and so is proportional to the concentration of ozone within that layer.

29.1.3 Measuring the Scattered Intensity

The radiation scattered at the angle θ then travels to the ground, a distance of $X - x$, weakened along the way, and reaches the ground with intensity

$$S(\theta, \theta_0)I(0)e^{-kx/\cos\theta_0}e^{-k(X-x)/\cos\theta}\Delta p. \tag{29.4}$$

The total scattered intensity at angle θ is then a superposition of the intensities due to scattering at each of the thin layers, and is then

$$S(\theta, \theta_0)I(0)e^{-kX/\cos\theta_0}\int_0^X e^{-x\beta}dp, \tag{29.5}$$

where

$$\beta = k[\frac{1}{\cos\theta_0} - \frac{1}{\cos\theta}]. \tag{29.6}$$

This superposition of intensity can then be written as

$$S(\theta, \theta_0)I(0)e^{-kX/\cos\theta_0}\int_0^X e^{-x\beta}p'(x)dx. \tag{29.7}$$

29.1.4 The Laplace Transform Data

Using integration by parts, we get

$$\int_0^X e^{-x\beta}p'(x)dx = p(X)e^{-\beta X} - p(0) + \beta\int_0^X e^{-\beta x}p(x)dx. \tag{29.8}$$

Since $p(0) = 0$ and $p(X)$ can be measured, our data is then the Laplace transform value

$$\int_0^{+\infty} e^{-\beta x}p(x)dx; \tag{29.9}$$

note that we can replace the upper limit X with $+\infty$ if we extend $p(x)$ as zero beyond $x = X$.

The variable β depends on the two angles θ and θ_0. We can alter θ as we measure and θ_0 changes as the sun moves relative to the earth. In this way we get values of the Laplace transform of $p(x)$ for various values of β. The problem then is to recover $p(x)$ from these values. Because the Laplace transform involves a smoothing of the function $p(x)$, recovering $p(x)$ from its Laplace transform is more ill-conditioned than is the Fourier transform inversion problem.

29.2 The Laplace Transform and Energy Spectral Estimation

In x-ray transmission tomography, x-ray beams are sent through the object and the drop in intensity is measured. These measurements are then used to estimate the distribution of attenuating material within the object. A typical x-ray beam contains components with different energy levels. Because components at different energy levels will be attenuated differently, it is important to know the relative contribution of each energy level to the entering beam. The energy spectrum is the function $f(E)$ that describes the intensity of the components at each energy level $E > 0$.

29.2.1 The attenuation coefficient function

Each specific material, say aluminum, for example, is associated with attenuation coefficients, which is a function of energy, which we shall denote by $\mu(E)$. A beam with the single energy E passing through a thickness x of the material will be weakened by the factor $e^{-\mu(E)x}$. By passing the beam through various thicknesses x of aluminum and registering the intensity drops, one obtains values of the absorption function

$$R(x) = \int_0^\infty f(E)e^{-\mu(E)x}dE. \tag{29.10}$$

Using a change of variable, we can write $R(x)$ as a Laplace transform.

29.2.2 The absorption function as a Laplace transform

For each material, the attenuation function $\mu(E)$ is a strictly decreasing function of E, so $\mu(E)$ has an inverse, which we denote by g; that is, $g(t) = E$, for $t = \mu(E)$. Equation (29.10) can then be rewritten as

$$R(x) = \int_0^\infty f(g(t))e^{-tx}g'(t)dt. \tag{29.11}$$

We see then that $R(x)$ is the Laplace transform of the function $r(t) = f(g(t))g'(t)$. Our measurements of the intensity drops provide values of $R(x)$, for various values of x, from which we must estimate the functions $r(t)$, and, ultimately, $f(E)$.

Bibliography

[1] Agmon, S. (1954) "The relaxation method for linear inequalities." *Canadian Journal of Mathematics* **6**, pp. 382–392.

[2] Ahn, S., and Fessler, J. (2003) "Globally convergent image reconstruction for emission tomography using relaxed ordered subset algorithms." *IEEE Transactions on Medical Imaging*, **22(5)**, pp. 613–626.

[3] Ahn, S., Fessler, J., Blatt, D., and Hero, A. (2006) "Convergent incremental optimization transfer algorithms: application to tomography." *IEEE Transactions on Medical Imaging*, **25(3)**, pp. 283–296.

[4] Anderson, A. and Kak, A. (1984) "Simultaneous algebraic reconstruction technique (SART): a superior implementation of the ART algorithm." *Ultrasonic Imaging* **6**, pp. 81–94.

[5] Ash, R. and Gardner, M. (1975) *Topics in Stochastic Processes* Boston: Academic Press.

[6] Axelsson, O. (1994) *Iterative Solution Methods*. Cambridge, UK: Cambridge University Press.

[7] Baillet, S., Mosher, J., and Leahy, R. (2001) "Electromagnetic Brain Mapping" , *IEEE Signal Processing Magazine*, **18 (6)**, pp. 14–30.

[8] Barrett, H., White, T., and Parra, L. (1997) "List-mode likelihood." *J. Opt. Soc. Am. A* **14**, pp. 2914–2923.

[9] Bauschke, H. (1996) "The approximation of fixed points of compositions of nonexpansive mappings in Hilbert space," *Journal of Mathematical Analysis and Applications*, **202**, pp. 150–159.

[10] Bauschke, H. (2001) "Projection algorithms: results and open problems." in *Inherently Parallel Algorithms in Feasibility and Optimization and their Applications*, Butnariu, D., Censor, Y., and Reich, S., editors, Amsterdam: Elsevier Science. pp. 11–22.

[11] Bauschke, H. and Borwein, J. (1996) "On projection algorithms for solving convex feasibility problems."*SIAM Review* **38 (3)**, pp. 367–426.

[12] Bauschke, H., Borwein, J., and Lewis, A. (1997) "The method of cyclic projections for closed convex sets in Hilbert space."*Contemporary Mathematics: Recent Developments in Optimization Theory and Nonlinear Analysis* **204**, American Mathematical Society, pp. 1–38.

[13] Bauschke, H., and Lewis, A. (2000) "Dykstra's algorithm with Bregman projections: a convergence proof." *Optimization*, **48**, pp. 409–427.

[14] Bertero, M. (1992) "Sampling theory, resolution limits and inversion methods."in [16], pp. 71–94.

[15] Bertero, M., and Boccacci, P. (1998) *Introduction to Inverse Problems in Imaging* Bristol, UK: Institute of Physics Publishing.

[16] Bertero, M. and Pike, E.R., editors (1992) *Inverse Problems in Scattering and Imaging* Malvern Physics Series, Adam Hilger, IOP Publishing, London.

[17] Bertsekas, D.P. (1997) "A new class of incremental gradient methods for least squares problems."*SIAM J. Optim.* **7**, pp. 913–926.

[18] Blackman, R. and Tukey, J. (1959) *The Measurement of Power Spectra*. New York: Dover Publications.

[19] Boas, D., Brooks, D., MIller, E., DiMarzio, C., Kilmer, M., Gaudette, R., and Zhang, Q. (2001) "Imaging the Body with Diffuse Optical Tomography" , *IEEE Signal Processing Magazine*, **18 (6)**, pp. 57–75.

[20] Born, M. and Wolf, E. (1999) *Principles of Optics: 7th edition*. Cambridge, UK: Cambridge University Press.

[21] Bochner, S. and Chandrasekharan, K. (1949) *Fourier Transforms*, Annals of Mathematical Studies, No. 19. Princeton, NJ: Princeton University Press.

[22] Borwein, J. and Lewis, A. (2000) *Convex Analysis and Nonlinear Optimization*. Canadian Mathematical Society Books in Mathematics, New York: Springer-Verlag.

[23] Bracewell, R.C. (1979) Image Reconstruction in Radio Astronomy, in [127], pp. 81–104.

[24] Bregman, L.M. (1967) "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming." *USSR Computational Mathematics and Mathematical Physics* **7**, pp. 200–217.

[25] Bregman, L., Censor, Y., and Reich, S. (1999) "Dykstra's algorithm as the nonlinear extension of Bregman's optimization method." *Journal of Convex Analysis*, **6 (2)**, pp. 319–333.

[26] Brooks, D., and MacLeod, R. (1997) "Electrical Imaging of the Heart" *IEEE Signal Processing Magazine*, **14 (1)**, pp. 24–42.

[27] Browne, J. and A. DePierro, A. (1996) "A row-action alternative to the EM algorithm for maximizing likelihoods in emission tomography." *IEEE Trans. Med. Imag.* **15**, pp. 687–699.

[28] Bruyant, P., Sau, J., and Mallet, J.J. (1999) "Noise removal using factor analysis of dynamic structures: application to cardiac gated studies." *Journal of Nuclear Medicine* **40 (10)**, pp. 1676–1682.

[29] Budinger, T., Gullberg, G., and Huesman, R. (1979) "Emission Computed Tomography." in [127], pp. 147–246.

[30] Burg, J. (1967) "Maximum entropy spectral analysis." *paper presented at the 37th Annual SEG meeting, Oklahoma City, OK.*

[31] Burg, J. (1972) "The relationship between maximum entropy spectra and maximum likelihood spectra." *Geophysics* **37**, pp. 375–376.

[32] Burg, J. (1975) *Maximum Entropy Spectral Analysis*, Ph.D. dissertation, Stanford University.

[33] Byrne, C. and Fitzgerald, R. (1979) "A unifying model for spectrum estimation." in *Proceedings of the RADC Workshop on Spectrum Estimation- October 1979*, Griffiss AFB, Rome, NY.

[34] Byrne, C. and Fitzgerald, R. (1982) "Reconstruction from partial information, with applications to tomography." *SIAM J. Applied Math.* **42(4)**, pp. 933–940.

[35] Byrne, C., Fitzgerald, R., Fiddy, M., Hall, T. and Darling, A. (1983) "Image restoration and resolution enhancement." *J. Opt. Soc. Amer.* **73**, pp. 1481–1487.

[36] Byrne, C., and Wells, D. (1983) "Limit of continuous and discrete finite-band Gerchberg iterative spectrum extrapolation." *Optics Letters* **8 (10)**, pp. 526–527.

[37] Byrne, C. and Fitzgerald, R. (1984) "Spectral estimators that extend the maximum entropy and maximum likelihood methods." *SIAM J. Applied Math.* **44(2)**, pp. 425–442.

[38] Byrne, C., Levine, B.M., and Dainty, J.C. (1984) "Stable estimation of the probability density function of intensity from photon frequency counts." *JOSA Communications* **1(11)**, pp. 1132–1135.

[39] Byrne, C., and Wells, D. (1985) "Optimality of certain iterative and non-iterative data extrapolation procedures." *Journal of Mathematical Analysis and Applications* **111 (1)**, pp. 26–34.

[40] Byrne, C. and Fiddy, M. (1987) "Estimation of continuous object distributions from Fourier magnitude measurements." *JOSA A* **4**, pp. 412–417.

[41] Byrne, C. and Fiddy, M. (1988) "Images as power spectra; reconstruction as Wiener filter approximation." *Inverse Problems* **4**, pp. 399–409.

[42] Byrne, C., Haughton, D., and Jiang, T. (1993) "High-resolution inversion of the discrete Poisson and binomial transformations." *Inverse Problems* **9**, pp. 39–56.

[43] Byrne, C. (1993) "Iterative image reconstruction algorithms based on cross-entropy minimization." *IEEE Transactions on Image Processing* **IP-2**, pp. 96–103.

[44] Byrne, C. (1995) "Erratum and addendum to 'Iterative image reconstruction algorithms based on cross-entropy minimization'." *IEEE Transactions on Image Processing* **IP-4**, pp. 225–226.

[45] Byrne, C. (1996) "Iterative reconstruction algorithms based on cross-entropy minimization."in *Image Models (and their Speech Model Cousins)*, S.E. Levinson and L. Shepp, editors, IMA Volumes in Mathematics and its Applications, Volume 80, pp. 1–11. New York: Springer-Verlag.

[46] Byrne, C. (1996) "Block-iterative methods for image reconstruction from projections." *IEEE Transactions on Image Processing* **IP-5**, pp. 792–794.

[47] Byrne, C. (1997) "Convergent block-iterative algorithms for image reconstruction from inconsistent data." *IEEE Transactions on Image Processing* **IP-6**, pp. 1296–1304.

[48] Byrne, C. (1998) "Accelerating the EMML algorithm and related iterative algorithms by rescaled block-iterative (RBI) methods." *IEEE Transactions on Image Processing* **IP-7**, pp. 100–109.

[49] Byrne, C. (1998) "Iterative deconvolution and deblurring with constraints" , *Inverse Problems*, **14**, pp. 1455-1467.

[50] Byrne, C. (1999) "Iterative projection onto convex sets using multiple Bregman distances." *Inverse Problems* **15**, pp. 1295–1313.

[51] Byrne, C. (2000) "Block-iterative interior point optimization methods for image reconstruction from limited data." *Inverse Problems* **16**, pp. 1405–1419.

[52] Byrne, C. (2001) "Bregman-Legendre multidistance projection algorithms for convex feasibility and optimization." in *Inherently Parallel Algorithms in Feasibility and Optimization and their Applications*, Butnariu, D., Censor, Y., and Reich, S., editors, pp. 87–100. Amsterdam: Elsevier Publ.,

[53] Byrne, C. (2001) "Likelihood maximization for list-mode emission tomographic image reconstruction." *IEEE Transactions on Medical Imaging* **20(10)**, pp. 1084–1092.

[54] Byrne, C. (2002) "Iterative oblique projection onto convex sets and the split feasibility problem." *Inverse Problems* **18**, pp. 441–453.

[55] Byrne, C. (2004) "A unified treatment of some iterative algorithms in signal processing and image reconstruction." *Inverse Problems* **20**, pp. 103–120.

[56] Byrne, C. (2005) "Choosing parameters in block-iterative or ordered-subset reconstruction algorithms." *IEEE Transactions on Image Processing*, **14 (3)**, pp. 321–327.

[57] Byrne, C. (2005) *Signal Processing: A Mathematical Approach*, AK Peters, Publ., Wellesley, MA.

[58] Byrne, C. (2007) *Applied Iterative Methods*, AK Peters, Publ., Wellesley, MA.

[59] Byrne, C. and Censor, Y. (2001) "Proximity function minimization using multiple Bregman projections, with applications to split feasibility and Kullback-Leibler distance minimization." *Annals of Operations Research* **105**, pp. 77–98.

[60] Candès, E., and Romberg, J. (2007) "Sparsity and incoherence in compressive sampling" *Inverse Problems*, **23(3)**, pp. 969–985.

[61] Candès, E., Romberg, J., and Tao, T. (2006) "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information" *IEEE Transactions on Information Theory*, **52(2)**, pp. 489–509.

[62] Candès, E., Wakin, M., and Boyd, S. (2007) "Enhancing sparsity by reweighted l_1 minimization" preprint available at http://www.acm.caltech.edu/ emmanuel/publications.html .

[63] Candy, J. (1988) *Signal Processing: The Modern Approach* New York: McGraw-Hill Publ.

[64] Cederquist, J., Fienup, J., Wackerman, C., Robinson, S., and Kryskowski, D. (1989) "Wave-front phase estimation from Fourier intensity measurements." *Journal of the Optical Society of America A* **6(7)**, pp. 1020–1026.

[65] Censor, Y. (1981) "Row-action methods for huge and sparse systems and their applications." *SIAM Review*, **23**: 444–464.

[66] Censor, Y., Eggermont, P.P.B., and Gordon, D. (1983) "Strong underrelaxation in Kaczmarz's method for inconsistent systems." *Numerische Mathematik* **41**, pp. 83–92.

[67] Censor, Y. and Elfving, T. (1994) "A multiprojection algorithm using Bregman projections in a product space." *Numerical Algorithms* **8**, pp. 221–239.

[68] Censor, Y., Elfving, T., Kopf, N., and Bortfeld, T. (2006) "The multiple-sets split feasibility problem and its application for inverse problems." *Inverse Problems*, to appear.

[69] Censor, Y., Bortfeld, T., Martin, B., and Trofimov, A. (2006) "A unified approach for inversion problems in intensity-modulated radiation therapy." , to appear.

[70] Censor, Y., and Reich, S. (1998) "The Dykstra algorithm for Bregman projections." *Communications in Applied Analysis*, **2**, pp. 323–339.

[71] Censor, Y. and Segman, J. (1987) "On block-iterative maximization." *J. of Information and Optimization Sciences* **8**, pp. 275–291.

[72] Censor, Y. and Zenios, S.A. (1997) *Parallel Optimization: Theory, Algorithms and Applications*. New York: Oxford University Press.

[73] Chang, J.-H., Anderson, J.M.M., and Votaw, J.R. (2004) "Regularized image reconstruction algorithms for positron emission tomography." *IEEE Transactions on Medical Imaging* **23(9)**, pp. 1165–1175.

[74] Childers, D., editor (1978) *Modern Spectral Analysis*. New York:IEEE Press.

[75] Chui, C. and Chen, G. (1991) *Kalman Filtering*, second edition. Berlin: Springer-Verlag.

[76] Cimmino, G. (1938) "Calcolo approssimato per soluzioni die sistemi di equazioni lineari." *La Ricerca Scientifica XVI, Series II, Anno IX* **1**, pp. 326–333.

[77] Combettes, P. (1993) "The foundations of set theoretic estimation." *Proceedings of the IEEE* **81 (2)**, pp. 182–208.

[78] Combettes, P. (1996) "The convex feasibility problem in image recovery." *Advances in Imaging and Electron Physics* **95**, pp. 155–270.

[79] Combettes, P. (2000) "Fejér monotonicity in convex optimization."in *Encyclopedia of Optimization*, C.A. Floudas and P. M. Pardalos, editors, Boston: Kluwer Publ.

[80] Combettes, P., and Trussell, J. (1990) "Method of successive projections for finding a common point of sets in a metric space." *Journal of Optimization Theory and Applications* **67 (3)**, pp. 487–507.

[81] Combettes, P., and Wajs, V. (2005) Signal recovery by proximal forward-backward splitting, *Multiscale Modeling and Simulation*, **4(4)**, pp. 1168–1200.

[82] Cooley, J. and Tukey, J. (1965) "An algorithm for the machine calculation of complex Fourier series." *Math. Comp.*, **19**, pp. 297–301.

[83] Csiszár, I. (1989) "A geometric interpretation of Darroch and Ratcliff's generalized iterative scaling." *The Annals of Statistics* **17 (3)**, pp. 1409–1413.

[84] Csiszár, I. (1991) "Why least squares and maximum entropy? An axiomatic approach to inference for linear inverse problems." *The Annals of Statistics* **19 (4)**, pp. 2032–2066.

[85] Csiszár, I. and Tusnády, G. (1984) "Information geometry and alternating minimization procedures." *Statistics and Decisions* **Supp. 1**, pp. 205–237.

[86] Dainty, J. C. and Fiddy, M. (1984) "The essential role of prior knowledge in phase retrieval." *Optica Acta* **31**, pp. 325–330.

[87] Darroch, J. and Ratcliff, D. (1972) "Generalized iterative scaling for log-linear models." *Annals of Mathematical Statistics* **43**, pp. 1470–1480.

[88] Dax, A. (1990) "The convergence of linear stationary iterative processes for solving singular unstructured systems of linear equations," *SIAM Review*, **32**, pp. 611–635.

[89] Dempster, A.P., Laird, N.M. and Rubin, D.B. (1977) "Maximum likelihood from incomplete data via the EM algorithm." *Journal of the Royal Statistical Society, Series B* **37**, pp. 1–38.

[90] De Pierro, A. (1995) "A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography." *IEEE Transactions on Medical Imaging* **14**, pp. 132–137.

[91] De Pierro, A. and Iusem, A. (1990) "On the asymptotic behavior of some alternate smoothing series expansion iterative methods." *Linear Algebra and its Applications* **130**, pp. 3–24.

[92] De Pierro, A., and Yamaguchi, M. (2001) "Fast EM-like methods for maximum 'a posteriori' estimates in emission tomography" *Transactions on Medical Imaging*, **20 (4)**.

[93] Deutsch, F., and Yamada, I. (1998) "Minimizing certain convex functions over the intersection of the fixed point sets of nonexpansive mappings" , *Numerical Functional Analysis and Optimization*, **19**, pp. 33–56.

[94] Dhanantwari, A., Stergiopoulos, S., and Iakovidis, I. (2001) "Correcting organ motion artifacts in x-ray CT medical imaging systems by adaptive processing. I. Theory." *Med. Phys.* **28(8)**, pp. 1562–1576.

[95] Donoho, D. (2006) "Compressed sampling" *IEEE Transactions on Information Theory*, **52 (4)**. (download preprints at http://www.stat.stanford.edu/ donoho/Reports).

[96] Duda, R., Hart, P., and Stork, D. (2001) *Pattern Classification*, Wiley.

[97] Dugundji, J. (1970) *Topology* Boston: Allyn and Bacon, Inc.

[98] Dykstra, R. (1983) "An algorithm for restricted least squares regression" *J. Amer. Statist. Assoc.*, **78 (384)**, pp. 837–842.

[99] Eggermont, P.P.B., Herman, G.T., and Lent, A. (1981) "Iterative algorithms for large partitioned linear systems, with applications to image reconstruction." *Linear Algebra and its Applications* **40**, pp. 37–67.

[100] Elsner, L., Koltracht, L., and Neumann, M. (1992) "Convergence of sequential and asynchronous nonlinear paracontractions." *Numerische Mathematik*, **62**, pp. 305–319.

[101] Erdogan, H., and Fessler, J. (1999) "Fast monotonic algorithms for transmission tomography" *IEEE Transactions on Medical Imaging*, **18(9)**, pp. 801–814.

[102] Everitt, B. and Hand, D. (1981) *Finite Mixture Distributions* London: Chapman and Hall.

[103] Farncombe, T. (2000) "Functional dynamic SPECT imaging using a single slow camera rotation" , *Ph.D. thesis, Dept. of Physics, University of British Columbia*.

[104] Fernandez, J., Sorzano, C., Marabini, R., and Carazo, J-M. (2006) "Image Processing and 3-D Reconstruction in Electron Microscopy" , *IEEE Signal Processing Magazine*, **23 (3)**, pp. 84–94.

[105] Fessler, J., Ficaro, E., Clinthorne, N., and Lange, K. (1997) Grouped-coordinate ascent algorithms for penalized-likelihood transmission image reconstruction, *IEEE Transactions on Medical Imaging*, **16 (2)**, pp. 166–175.

[106] Feynman, R., Leighton, R., and Sands, M. (1963) *The Feynman Lectures on Physics, Vol. 1*. Boston: Addison-Wesley.

[107] Fiddy, M. (1983) "The phase retrieval problem."in *Inverse Optics*, SPIE Proceedings 413 (A.J. Devaney, editor), pp. 176–181.

[108] Fiddy, M. (2008) *private communication*.

[109] Fienup, J. (1979) "Space object imaging through the turbulent atmosphere." *Optical Engineering* **18**, pp. 529–534.

[110] Fienup, J. (1987) "Reconstruction of a complex-valued object from the modulus of its Fourier transform using a support constraint." *Journal of the Optical Society of America A* **4(1)**, pp. 118–123.

[111] Fleming, W. (1965) *Functions of Several Variables*, Addison-Wesley Publ., Reading, MA.

[112] Frieden, B. R. (1982) *Probability, Statistical Optics and Data Testing*. Berlin: Springer-Verlag.

[113] Gasquet, C. and Witomski, F. (1998) *Fourier Analysis and Applications*. Berlin: Springer-Verlag.

[114] Gelb, A., editor, (1974) *Applied Optimal Estimation*, written by the technical staff of The Analytic Sciences Corporation, MIT Press, Cambridge, MA.

[115] Geman, S., and Geman, D. (1984) "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images." *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-6**, pp. 721–741.

[116] Gerchberg, R. W. (1974) "Super-restoration through error energy reduction." *Optica Acta* **21**, pp. 709–720.

[117] Gifford, H., King, M., de Vries, D., and Soares, E. (2000) "Channelized Hotelling and human observer correlation for lesion detection in hepatic SPECT imaging" *Journal of Nuclear Medicine* **41(3)**, pp. 514–521.

[118] Goebel, K., and Reich, S. (1984) *Uniform Convexity, Hyperbolic Geometry, and Nonexpansive Mappings*, New York: Dekker.

[119] Golshtein, E., and Tretyakov, N. (1996) *Modified Lagrangians and Monotone Maps in Optimization*. New York: John Wiley and Sons, Inc.

[120] Gordon, R., Bender, R., and Herman, G.T. (1970) "Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography." *J. Theoret. Biol.* **29**, pp. 471–481.

[121] Green, P. (1990) "Bayesian reconstructions from emission tomography data using a modified EM algorithm." *IEEE Transactions on Medical Imaging* **9**, pp. 84–93.

[122] Gubin, L.G., Polyak, B.T. and Raik, E.V. (1967) "The method of projections for finding the common point of convex sets." *USSR Computational Mathematics and Mathematical Physics* **7**, pp. 1–24.

[123] Gullberg, G., Huesman, R., Malko, J., Pelc, N., and Budinger, T. (1986) "An attenuated projector-backprojector for iterative SPECT reconstruction." *Physics in Medicine and Biology*, **30**, pp. 799–816.

[124] Haacke, E., Brown, R., Thompson, M., and Venkatesan, R. (1999) *Magnetic Resonance Imaging*. New York: Wiley-Liss.

[125] Haykin, S. (1985) *Array Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall.

[126] Hebert, T. and Leahy, R. (1989) "A generalized EM algorithm for 3-D Bayesian reconstruction from Poisson data using Gibbs priors." *IEEE Transactions on Medical Imaging* **8**, pp. 194–202.

[127] Herman, G.T. (ed.) (1979) "Image Reconstruction from Projections" , *Topics in Applied Physics, Vol. 32*, Springer-Verlag, Berlin.

[128] Herman, G.T., and Natterer, F. (eds.) (1981) "Mathematical Aspects of Computerized Tomography" , *Lecture Notes in Medical Informatics, Vol. 8*, Springer-Verlag, Berlin.

[129] Herman, G.T., Censor, Y., Gordon, D., and Lewitt, R. (1985) Comment (on the paper [208]), *Journal of the American Statistical Association* **80**, pp. 22–25.

[130] Herman, G. T. (1999) *private communication.*

[131] Herman, G. T. and Meyer, L. (1993) "Algebraic reconstruction techniques can be made computationally efficient." *IEEE Transactions on Medical Imaging* **12**, pp. 600–609.

[132] Hildreth, C. (1957) "A quadratic programming procedure." *Naval Research Logistics Quarterly* **4**, pp. 79–85. Erratum, p. 361.

[133] Hogg, R. and Craig, A. (1978) *Introduction to Mathematical Statistics* MacMillan, New York.

[134] Holte, S., Schmidlin, P., Linden, A., Rosenqvist, G. and Eriksson, L. (1990) "Iterative image reconstruction for positron emission tomography: a study of convergence and quantitation problems." *IEEE Transactions on Nuclear Science* **37**, pp. 629–635.

[135] Hudson, M., Hutton, B., and Larkin, R. (1992) "Accelerated EM reconstruction using ordered subsets." *Journal of Nuclear Medicine*, **33**, p.960.

[136] Hudson, H.M. and Larkin, R.S. (1994) "Accelerated image reconstruction using ordered subsets of projection data." *IEEE Transactions on Medical Imaging* **13**, pp. 601–609.

[137] Huesman, R., Klein, G., Moses, W., Qi, J., Ruetter, B., and Virador, P. (2000) "List-mode maximum likelihood reconstruction applied to positron emission mammography (PEM) with irregular sampling." *IEEE Transactions on Medical Imaging* **19 (5)**, pp. 532–537.

[138] Hutton, B., Kyme, A., Lau, Y., Skerrett, D., and Fulton, R. (2002) "A hybrid 3-D reconstruction/registration algorithm for correction of head motion in emission tomography." *IEEE Transactions on Nuclear Science* **49 (1)**, pp. 188–194.

[139] Kaczmarz, S. (1937) "Angenäherte Auflösung von Systemen linearer Gleichungen." *Bulletin de l'Academie Polonaise des Sciences et Lettres* **A35**, pp. 355–357.

[140] Kak, A., and Slaney, M. (2001) "Principles of Computerized Tomographic Imaging" , SIAM, Philadelphia, PA.

[141] Kalman, R. (1960) "A new approach to linear filtering and prediction problems." *Trans. ASME, J. Basic Eng.* **82**, pp. 35–45.

[142] Katznelson, Y. (1983) *An Introduction to Harmonic Analysis*. New York: John Wiley and Sons, Inc.

[143] King, M., Glick, S., Pretorius, H., Wells, G., Gifford, H., Narayanan, M., and Farncombe, T. (2004) Attenuation, Scatter, and Spatial Resolution Compensation in SPECT, in [210], pp. 473–498.

[144] Koltracht, L., and Lancaster, P. (1990) "Constraining strategies for linear iterative processes." *IMA J. Numer. Anal.*, **10**, pp. 555–567.

[145] Körner, T. (1988) *Fourier Analysis*. Cambridge, UK: Cambridge University Press.

[146] Körner, T. (1996) *The Pleasures of Counting*. Cambridge, UK: Cambridge University Press.

[147] Kullback, S. and Leibler, R. (1951) "On information and sufficiency." *Annals of Mathematical Statistics* **22**, pp. 79–86.

[148] Landweber, L. (1951) "An iterative formula for Fredholm integral equations of the first kind." *Amer. J. of Math.* **73**, pp. 615–624.

[149] Lane, R. (1987) "Recovery of complex images from Fourier magnitude." *Optics Communications* **63(1)**, pp. 6–10.

[150] Lange, K. and Carson, R. (1984) "EM reconstruction algorithms for emission and transmission tomography." *Journal of Computer Assisted Tomography* **8**, pp. 306–316.

[151] Lange, K., Bahn, M. and Little, R. (1987) "A theoretical study of some maximum likelihood algorithms for emission and transmission tomography." *IEEE Trans. Med. Imag.* **MI-6(2)**, pp. 106–114.

[152] La Rivière, P., and Vargas, P. (2006) "Monotonic penalized-likelihood image reconstruction for x-ray fluorescence computed tomography." *IEEE Transactions on Medical Imaging* **25(9)**, pp. 1117–1129.

[153] Leahy, R., Hebert, T., and Lee, R. (1989) "Applications of Markov random field models in medical imaging." in *Proceedings of the Conference on Information Processing in Medical Imaging* Lawrence-Berkeley Laboratory, Berkeley, CA.

[154] Leahy, R. and Byrne, C. (2000) "Guest editorial: Recent development in iterative image reconstruction for PET and SPECT." *IEEE Trans. Med. Imag.* **19**, pp. 257–260.

[155] Leis, A., Beck, M., Gruska, M., Best, C., Hegerl, R., Baumeister, W., and Leis, J. (2006) "Cryo-electron tomography of biological specimens" , *IEEE Signal Processing Magazine*, **23 (3)**, pp. 95–103.

[156] Lent, A. (1998) *private communication*.

[157] Levitan, E. and Herman, G. (1987) "A maximum *a posteriori* probability expectation maximization algorithm for image reconstruction in emission tomography." *IEEE Transactions on Medical Imaging* **6**, pp. 185–192.

[158] Liao, C.-W., Fiddy, M., and Byrne, C. (1997) "Imaging from the zero locations of far-field intensity data." *Journal of the Optical Society of America -A* **14 (12)**, pp. 3155–3161.

[159] Luenberger, D. (1969) *Optimization by Vector Space Methods*. New York: John Wiley and Sons, Inc.

[160] Lustig, M., Donoho, D., and Pauly, J. (2008) *Magnetic Resonance in Medicine*, to appear.

[161] Mann, W. (1953) "Mean value methods in iteration." *Proc. Amer. Math. Soc.* **4**, pp. 506–510.

[162] McLachlan, G.J. and Krishnan, T. (1997) *The EM Algorithm and Extensions*. New York: John Wiley and Sons, Inc.

[163] McVeigh, E., and Ozturk, C. (2001) "Imaging Myocardial Strain" , *IEEE Signal Processing Magazine*, **18 (6)**, pp. 44–56.

[164] Meidunas, E. (2001) *Re-scaled Block Iterative Expectation Maximization Maximum Likelihood (RBI-EMML) Abundance Estimation and Sub-pixel Material Identification in Hyperspectral Imagery*, MS thesis, Department of Electrical Engineering, University of Massachusetts Lowell.

[165] Meijering, E., Smal, I., and Danuser, G. (2006) "Tracking in Molecular Bioimaging" , *IEEE Signal Processing Magazine*, **23 (3)**, pp. 46–53.

[166] Motzkin, T. and Schoenberg, I. (1954) "The relaxation method for linear inequalities." *Canadian Journal of Mathematics* **6**, pp. 393–404.

[167] Mumcuoglu, E., Leahy, R., and Cherry, S. (1996) "Bayesian reconstruction of PET images: Methodology and performance analysis." *Phys. Med. Biol.*, **41**, pp. 1777–1807.

[168] Narayanan, M., Byrne, C. and King, M. (2001) "An interior point iterative maximum-likelihood reconstruction algorithm incorporating upper and lower bounds with application to SPECT transmission imaging." *IEEE Transactions on Medical Imaging* **TMI-20 (4)**, pp. 342–353.

[169] Nash, S. and Sofer, A. (1996) *Linear and Nonlinear Programming*. New York: McGraw-Hill.

[170] Natterer, F. (1986) *Mathematics of Computed Tomography*. New York: John Wiley and Sons, Inc.

[171] Natterer, F., and Wübbeling, F. (2001) *Mathematical Methods in Image Reconstruction*. Philadelphia, PA: SIAM Publ.

[172] Ollinger, J., and Fessler, J. (1997) "Positron-Emission Tomography" , *IEEE Signal Processing Magazine*, **14 (1)**, pp. 43–55.

[173] Oppenheim, A. and Schafer, R. (1975) *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall.

[174] Papoulis, A. (1975) "A new algorithm in spectral analysis and band-limited extrapolation." *IEEE Transactions on Circuits and Systems* **22**, pp. 735–742.

[175] Papoulis, A. (1977) *Signal Analysis*. New York: McGraw-Hill.

[176] Parra, L. and Barrett, H. (1998) "List-mode likelihood: EM algorithm and image quality estimation demonstrated on 2-D PET." *IEEE Transactions on Medical Imaging* **17**, pp. 228–235.

[177] Paulraj, A., Roy, R., and Kailath, T. (1986) "A subspace rotation approach to signal parameter estimation." *Proceedings of the IEEE* **74**, pp. 1044–1045.

[178] Peressini, A., Sullivan, F., and Uhl, J. (1988) *The Mathematics of Nonlinear Programming*. Berlin: Springer-Verlag.

[179] Peters, T. (1981) "Resolution improvement to CT systems using aperture-function correction" , in [128], pp. 241–251.

[180] Pretorius, H., King, M., Pan, T-S, deVries, D., Glick, S., and Byrne, C. (1998) "Reducing the influence of the partial volume effect on SPECT activity quantitation with 3D modelling of spatial resolution in iterative reconstruction" , *Phys.Med. Biol.* **43**, pp. 407–420.

[181] Pižurica, A., Philips, W., Lemahieu, I., and Acheroy, M. (2003) "A versatile wavelet domain noise filtration technique for medical imaging." *IEEE Transactions on Medical Imaging: Special Issue on Wavelets in Medical Imaging* **22**, pp. 323–331.

[182] Poggio, T. and Smale, S. (2003) "The mathematics of learning: dealing with data."*Notices of the American Mathematical Society* **50 (5)**, pp. 537–544.

[183] Priestley, M. B. (1981) *Spectral Analysis and Time Series*. Boston: Academic Press.

[184] Qi, J., Leahy, R., Cherry, S., Chatziioannou, A., and Farquhar, T. (1998) "High resolution 3D Bayesian image reconstruction using the microPET small animal scanner. " *Phys. Med. Biol.*, **43 (4)**, pp. 1001–1013.

[185] Qian, H. (1990) "Inverse Poisson transformation and shot noise filtering."*Rev. Sci. Instrum.* **61**, pp. 2088–2091.

[186] Quistgaard, J. (1997) "Signal Acquisition and Processing in Medical Diagnostic Ultrasound" , *IEEE Signal processing Magazine*, **14 (1)**, pp. 67–74.

[187] Rockafellar, R. (1970) *Convex Analysis*. Princeton, NJ: Princeton University Press.

[188] Rockmore, A., and Macovski, A. (1976) "A maximum likelihood approach to emission image reconstruction from projections" , *IEEE Transactions on Nuclear Science*, **NS-23**, pp. 1428–1432.

[189] Sarder, P., and Nehorai, A. (2006) "Deconvolution Methods for 3-D Fluoresence Microscopy Images" , *IEEE Signal Processing Magazine*, **23 (3)**, pp. 32–45.

[190] Saulnier, G., Blue, R., Newell, J., Isaacson, D., and Edic, P. (2001) "Electrical Impedance Tomography" , *IEEE Signal Processing Magazine*, **18 (6)**, pp. 31–43.

[191] Schmidlin, P. (1972) "Iterative separation of sections in tomographic scintigrams." *Nucl. Med.* **15(1)**.

[192] Schultz, L., Blanpied, G., Borozdin, K., *et al.* (2007) "Statistical reconstruction for cosmic ray muon tomography." *IEEE Transactions on Image Processing*, **16(8)**, pp. 1985–1993.

[193] Shepp, L., and Vardi, Y. (1982) Maximum likelihood reconstruction for emission tomography, *IEEE Transactions on Medical Imaging*, **MI-1**, pp. 113–122.

[194] Shieh, M., Byrne, C., Testorf, M., and Fiddy, M. (2006) "Iterative image reconstruction using prior knowledge." *Journal of the Optical Society of America, A*, **23(6)**, pp. 1292–1300.

[195] Shieh, M., Byrne, C., and Fiddy, M. (2006) "Image reconstruction: a unifying model for resolution enhancement and data extrapolation: Tutorial." *Journal of the Optical Society of America, A*, **23(2)**, pp. 258–266.

[196] Shieh, M., and Byrne, C. (2006) "Image reconstruction from limited Fourier data." *Journal of the Optical Society of America, A*, **23(11)**.

[197] Smith, C. Ray and Grandy, W.T., editors (1985) *Maximum-Entropy and Bayesian Methods in Inverse Problems*. Dordrecht: Reidel Publ.

[198] Smith, C. Ray and Erickson, G., editors (1987) *Maximum-Entropy and Bayesian Spectral Analysis and Estimation Problems*. Dordrecht: Reidel Publ.

[199] Soares, E., Byrne, C., Glick, S., Appledorn, R., and King, M. (1993) Implementation and evaluation of an analytic solution to the photon attenuation and nonstationary resolution reconstruction problem in SPECT, *IEEE Transactions on Nuclear Science*, **40 (4)**, pp. 1231–1237.

[200] Stark, H. and Yang, Y. (1998) *Vector Space Projections: A Numerical Approach to Signal and Image Processing, Neural Nets and Optics*. New York: John Wiley and Sons, Inc.

[201] Strang, G. (1980) *Linear Algebra and its Applications*. New York: Academic Press.

[202] Tanabe, K. (1971) "Projection method for solving a singular system of linear equations and its applications." *Numer. Math.* **17**, pp. 203–214.

[203] Therrien, C. (1992) *Discrete Random Signals and Statistical Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall.

[204] Tsui, B., Gullberg, G., Edgerton, E., Ballard, J., Perry, J., McCartney, W., and Berg, J. (1989) "Correction of non-uniform attenuation in cardiac SPECT imaging." *Journal of Nuclear Medicine*, **30(4)**, pp. 497–507.

[205] Twomey, S. (1996) *Introduction to the Mathematics of Inversion in Remote Sensing and Indirect Measurement*. New York: Dover Publ.

[206] Udpa, L., Ayres, V., Fan, Y., Chen, Q., Kumar, S. (2006) "Deconvolution of Atomic Force Microscopy Data for Cellular and Molecular Imaging" , *IEEE Signal Processing Magazine*, **23 (3)**, pp. 73–83.

[207] Van Trees, H. (1968) *Detection, Estimation and Modulation Theory*. New York: John Wiley and Sons, Inc.

[208] Vardi, Y., Shepp, L.A. and Kaufman, L. (1985) "A statistical model for positron emission tomography." *Journal of the American Statistical Association* **80**, pp. 8–20.

[209] Vonesch, C., Aguet, F., Vonesch, J-L, and Unser, M. (2006) "The Colored Revolution in BioImaging" , *IEEE Signal Processing Magazine*, **23 (3)**, pp. 20–31.

[210] Wernick, M. and Aarsvold, J., editors (2004) *Emission Tomography: The Fundamentals of PET and SPECT*. San Diego: Elsevier Academic Press.

[211] Wiener, N. (1949) *Time Series*. Cambridge, MA: MIT Press.

[212] Wright, G.A. (1997) "Magnetic Resonance Imaging" , *IEEE Signal Processing Magazine*, **14 (1)**, pp. 56–66.

[213] Wright, W., Pridham, R., and Kay, S. (1981) "Digital signal processing for sonar." *Proc. IEEE* **69**, pp. 1451–1506.

[214] Yang, Q. (2004) "The relaxed CQ algorithm solving the split feasibility problem." *Inverse Problems*, **20**, pp. 1261–1266.

[215] Yin, W., and Zhang, Y. (2008) "Extracting salient features from less data via l_1-minimization." *SIAG/OPT Views-and-News*, **19(1)**, pp. 11–19.

[216] Youla, D. (1978) "Generalized image restoration by the method of alternating projections." *IEEE Transactions on Circuits and Systems* **CAS-25 (9)**, pp. 694–702.

[217] Youla, D.C. (1987) "Mathematical theory of image restoration by the method of convex projections." in *Image Recovery: Theory and Applications*, pp. 29–78, Stark, H., editor (1987) Orlando FL: Academic Press.

[218] Young, R. (1980) *An Introduction to Nonharmonic Fourier Analysis*. Boston: Academic Press.

[219] Zhou, X., and Wong, S. (2006) "Informatics challenges of high-throughput microscopy" , *IEEE Signal Processing Magazine*, **23 (3)**, pp. 63–72.

[220] Zimmer, C., Zhang, B., Dufour, A., Thébaud, A., Berlemont, S., Meas-Yedid, V., and Marin, J-C. (2006) "On the digital trail of mobile cells" , *IEEE Signal Processing Magazine*, **23 (3)**, pp. 54–62.

Index