Exam 2 Solution

1. (a) (ii) False. There's only one residual plot since there is only one set of residuals, from

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + e_i \quad \text{and} \quad \hat{e}_i = y_i - \hat{y}_i$$

   (b) (iv) None of the above. It is $n - p - 1$ for all coefficients.

   (c) (i) We will get the same estimates for C and D in the two models. This follows from arguments in Section 4.1.3 (see `BGSgirls` problem)

   (d) (i) The slopes of the added variable plots are the same as the slopes of multiple regression, but they can change signs when variables are added/deleted (see textbook, page 80, Table 4.2, Models 2 and 3).

   (e) (iv) None of the above. There is no restriction on the effects plot (there are for added variable plots).

2. (a) This is done by the following commands

```
m2<-lm(log(muniUse)~year+muniPrecip, data=MinnWater)
predict(m2, newdata=data.frame(year=2015, muniPrecip=20),interval="prediction",
        level=.98)
```

   giving us the answer

```
       fit      lwr      upr
1 5.003487 4.873301 5.133674
```

   (b) If we assume that have computed n, $\mathbf{X}$, $\hat{\boldsymbol{\beta}}$, $\hat{\sigma}^2$, as `n`, `X`, `bhat`, `sigma2hat` in R, respectively, then we just need

```
xstar<-c(1, 2015, 20)
ystar<-t(xstar)%*%bhat
sepred<-sqrt(sigma2hat*(1+t(xstar)%*%solve(t(X)%*%X)%*%xstar))
ystar
ystar-qt(0.01,n-2-1,lower.tail=F)*sepred
ystar+qt(0.01,n-2-1,lower.tail=F)*sepred
```

   giving us the matching answer

```
> ystar
         [,1]
[1,] 5.003487
> ystar-qt(0.01,n-2-1,lower.tail=F)*sepred
         [,1]
[1,] 4.873301
> ystar+qt(0.01,n-2-1,lower.tail=F)*sepred
         [,1]
[1,] 5.133674
```

3. (a) The easiest thing to do is write

```
y<-c(5,3,4)
x1<-c(-1,0,1)
x2<-c(-1,2,-1)
X<-matrix(c(rep(1,3),x1,x2),ncol=3)
bhat<-solve(t(X)%*%X)%*%t(X)%*%y
bhat[3]
```

(giving us the answer -0.5).

(b) Be careful that the usual sequence of codes will not work, because $n-p-1 = 3-2-1 = 0$, so we must use the information $\hat{\sigma}^2 = 3$ in the code. Assuming that we have already computed matrix `X` in R, then you may do

```
sigma2hat<-3
varhatbhat<-sigma2hat*solve(t(X)%*%X)
sebhat<-sqrt(diag(varhatbhat))
sebhat[3]
```

(giving us the answer 0.7071068).

(c) (BONUS) First, since

$$\mathbf{X}'\mathbf{X} = \begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & 2 & -1 \end{pmatrix} \begin{pmatrix} 1 & -1 & -1 \\ 1 & 0 & 2 \\ 1 & 1 & -1 \end{pmatrix} = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 6 \end{pmatrix}$$

and $\widehat{\text{Var}}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}^2(\mathbf{X}'\mathbf{X})^{-1}$, with

$$(\mathbf{X}'\mathbf{X})^{-1} = \begin{pmatrix} 1/3 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/6 \end{pmatrix}$$

we obtain (with $\hat{\sigma}^2 = 3$)

$$\widehat{\text{Var}}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}^2(\mathbf{X}'\mathbf{X})^{-1} = 3 \begin{pmatrix} 1/3 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3/2 & 0 \\ 0 & 0 & 1/2 \end{pmatrix}$$

Now, $\text{se}(\hat{\beta}_0)$ is simply a square root of the $[3,3]$ element of $\widehat{\text{Var}}(\hat{\boldsymbol{\beta}})$, which is $\sqrt{1/2}$, so that we have

$$\text{se}(\hat{\beta}_2) = \sqrt{1/2} = 0.7071068.$$

4. Since

$$\text{Var}(\mathbf{1}_n'\widehat{\mathbf{Y}}) = \text{Var}(\mathbf{1}_n'\mathbf{X}\hat{\boldsymbol{\beta}}) = \mathbf{1}_n'\mathbf{X}\text{Var}(\hat{\boldsymbol{\beta}})\mathbf{X}'\mathbf{1}_n = \mathbf{1}_n'\mathbf{X}\sigma^2(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{1}_n = \sigma^2\mathbf{1}_n'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{1}_n$$

we see that it has the form $\text{Var}(\mathbf{1}_n'\widehat{\mathbf{Y}}) = \sigma^2 \cdot c$ with $c = \mathbf{1}_n'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{1}_n$. (Note that c is a scalar...).