Solutions Chapter 1

Data Set Up

```
library(datasets)
data(airquality)
#remove any rows with NAs omitted.
airquality=na.omit(airquality)
#make Month a factor (i.e., the Month number is a name rather than a number)
airquality$Month=as.factor(airquality$Month)
#add a region factor
airquality$region = rep(c("north","south"),60)[1:111]
#Only use 5 data points for the homework so you can show the matrices easily
homeworkdat = airquality[1:5,]
```

Problem 1

Using Form 1 $\mathbf{y} = \mathbf{Zx} + \mathbf{e}$, write the model being fit by this command

```
fit=lm(Ozone ~ Wind + Temp, data=homeworkdat)
```

The model is $Ozone_i = \alpha + \beta_w Wind_i + \beta_t Temp_i + e_i$. Form 1 for this model is:

$$\begin{bmatrix} Ozone_1 \\ Ozone_2 \\ Ozone_3 \\ Ozone_4 \\ Ozone_5 \end{bmatrix} = \begin{bmatrix} 1 & Wind_1 & Temp_1 \\ 1 & Wind_2 & Temp_2 \\ 1 & Wind_3 & Temp_3 \\ 1 & Wind_4 & Temp_4 \\ 1 & Wind_5 & Temp_5 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta_w \\ \beta_t \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{bmatrix}$$

Problem 2

Build the y and Z matrices for the above model in R and solve for x (the parameters). Show that they match what you get from the 'lm' call.

Here are the **y** and **Z**. You can see they match the **y** and **Z** in the equation above:

```
y=matrix(homeworkdat$Ozone, ncol=1)
Z=cbind(1, homeworkdat$Wind, homeworkdat$Temp)
y
```

```
      [,1]
[1,]    41
[2,]    36
[3,]    12
[4,]    18
[5,]    23
```

```
 Z
```

```
     [,1] [,2] [,3]
[1,]    1  7.4   67
[2,]    1  8.0   72
[3,]    1 12.6   74
[4,]    1 11.5   62
[5,]    1  8.6   65
```

Next we solve for **x** and show it matches what we get from 'lm'. This uses the code in section 1.1.1

```
 solve(t(Z)%*%Z)%*%t(Z)%*%y
```

```
           [,1]
[1,] 56.6219201
[2,] -4.9776707
[3,]  0.2538717
```

```
 coef(lm(Ozone ~ Wind + Temp, data=homeworkdat))
```

```
(Intercept)         Wind        Temp
 56.6219201   -4.9776707   0.2538717
```

Problem 3

If you added -1 to your 'lm' call in question 1, what changes in your model?

First run the 'lm' call and see what changed. The intercept (α) is dropped.

```
 fit=lm(Ozone ~ -1 + Wind + Temp, data=homeworkdat)
 fit
```

```
Call:
lm(formula = Ozone ~ -1 + Wind + Temp, data = homeworkdat)

Coefficients:
  Wind    Temp
-4.650   1.037
```

The model is now $Ozone_i = \beta_w Wind_i + \beta_t Temp_i + e_i$. To get rid of the α we drop the 1s column:

$$\begin{bmatrix} Ozone_1 \\ Ozone_2 \\ Ozone_3 \\ Ozone_4 \\ Ozone_5 \end{bmatrix} = \begin{bmatrix} Wind_1 \ Temp_1 \\ Wind_2 \ Temp_2 \\ Wind_3 \ Temp_3 \\ Wind_4 \ Temp_4 \\ Wind_5 \ Temp_5 \end{bmatrix} \begin{bmatrix} \beta_w \\ \beta_t \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{bmatrix}$$

Problem 4, part 1

Write the model for question 1 in Form 2.

In Form 2, $\mathbf{y} = \mathbf{Zx} + \mathbf{e}$ and the explanatory variables appear in the $\mathbf{x}$ as a column vector (a matrix with one column). So $\mathbf{x}$ looks like this

$$\begin{bmatrix} 1 \\ Wind_1 \\ \dots \\ Wind_5 \\ Temp_1 \\ \dots \\ Temp_5 \end{bmatrix}$$

Once you get that, then you know that $\mathbf{Z}$ is a $5 \times (1 + 5 + 5)$ matrix. The first column of $\mathbf{Z}$ is the α. The next 5 columns of $\mathbf{Z}$ is a 5×5 diagonal matrix with β_w on the diagonal, just like in Equation 1.12. For the next explanatory variable, Temperature, we tack on another 5×5 diagonal matrix; this time with β_t on the diagonal. That's all you needed to say for homework; just to show that you figured out what the form of the $\mathbf{y}$, $\mathbf{Z}$ and $\mathbf{x}$ look like.

If you wanted to write it out in math form, you could show Z as:

$$\begin{bmatrix} \text{column of} & 5 \times 5 \text{ diagonal matrix} & 5 \times 5 \text{ diagonal matrix} \\ \alpha & \text{with } \beta_w \text{ on the diagonal} & \text{with } \beta_t \text{ on the diagonal} \end{bmatrix}$$

or

$$\begin{bmatrix} \alpha & \beta_w & \dots & 0 & \beta_t & \dots & 0 \\ \dots & \dots & \ddots & \dots & \dots & \ddots & \dots \\ \alpha & 0 & \dots & \beta_w & 0 & \dots & \beta_t \end{bmatrix}$$

Problem 4, part 2

Solve for the parameters.

To do the 2nd part, you adapt the code from subsection 1.2.3 to solve for the parameters. You will need to contruct new **Z**, **y** and **x** in the code.

The **y** and **x** are easy:

```
y=matrix(homeworkdat$Ozone, ncol=1)
x=matrix(c(1, homeworkdat$Wind, homeworkdat$Temp), ncol=1)
```

We adapt the code for making **Z** from 1.2.3:

```
#we know that Z is a 5 x (1+5+5) matrix
#n is the number of data points
n=5
#nrows = n; what about ncol?,  ncol is 1 (alpha) + n (Wind) + n (Temp)
Z=matrix(list(0),n,1+n+n)
#the first column is alpha
Z[,1]="alpha"
#columns 2:112 are a diagonal matrix with betaw on the diagonal
diag(Z[,2:(n+1)])="betaw"
#columns 113:223 are a diagonal matrix with betat on the diagonal
diag(Z[,(n+2):(2*n+1)])="betat"
```

Now we can solve for Z:

```
require(MARSS)
P=MARSS:::convert.model.mat(Z)$free[,,1]
M=kronecker(t(x),diag(n))%*%P
solve(t(M)%*%M)%*%t(M)%*%y
```

```
            [,1]
alpha 56.6219201
betaw -4.9776707
betat  0.2538717
```

```
 coef(lm(Ozone ~ Wind + Temp, data=homeworkdat))
```

```
(Intercept)         Wind        Temp
 56.6219201   -4.9776707    0.2538717
```

Problem 5, part 1

Model the ozone data with only a region effect. Write this in Form 1b.

First make sure you understand what model is being fit by the 'lm' call:

```
fit=lm(Ozone ~ -1 + region, data=homeworkdat)
fit
```

```
Call:
lm(formula = Ozone ~ -1 + region, data = homeworkdat)

Coefficients:
regionnorth  regionsouth
      25.33        27.00
```

The model is $Ozone_i = \alpha_j + e_i$ where j is the region the measurement was taken in. This is an intercept (or level) only model where the intercept is determined by the region (north or south).

We want to write that model in matrix form using Form 1b, $\mathbf{y} = \mathbf{Dd} + \mathbf{e}$. Eqn 1.21 shows you how to do this for Form 1, except we do not have explanatory variables besides region so we do not have a column with something like 'air' in it. Form 1b is the transpose of Form 1.

Matrix $\mathbf{y}$ is:
$$\mathbf{y} = \begin{bmatrix} Ozone_1 \ Ozone_2 \ \ldots \ Ozone_5 \end{bmatrix}$$

Matrix $\mathbf{D}$ is:
$$\mathbf{D} = \begin{bmatrix} \alpha_n \ \alpha_s \end{bmatrix}$$

For $\mathbf{d}$, we need to know that each column of $\mathbf{d}$ is for a different data point i and tells us what region that data point is from. So $\mathbf{d}$ has 2 rows, one for each region. If there is a 1 in row 1, it means that data point came from the north. If there is a 1 in row 2, it means that data point came from the south.

Let's look at the regions

```
homeworkdat$region
```

```
[1] "north" "south" "north" "south" "north"
```

So the first measurement is from the north, next from south, then north, ... $\mathbf{Z}$ looks like this
$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

The easy way to figure out $\mathbf{d}$ is to remember that Form 1b is just the transpose of Form 1, so $\mathbf{d} = \mathbf{Z}^\top$. So let R show you what $\mathbf{d}$ is:

```
t(model.matrix(fit))
```

```
            1 2 3 4 7
regionnorth 1 0 1 0 1
regionsouth 0 1 0 1 0
attr(,"assign")
[1] 1 1
attr(,"contrasts")
attr(,"contrasts")$region
[1] "contr.treatment"
```

Problem 5, part 2

Show that you can solve for the parameters.

Section 1.1.4 shows you how to solve for the parameters when the model is in form 1b. We need to make **y** and **d** in R.

```
y=matrix(homeworkdat$Ozone, nrow=1)
```

We could form **d** like so

```
ndatapoints=5
d=matrix(0,2,ndatapoints)
for(i in 1:ndatapoints) d[ifelse(airquality$region[i]=="north",1,2),i]=1
```

or this

```
d=rbind(
   as.numeric(airquality$region=="north"),
   as.numeric(airquality$region=="south")
)
```

or just use the output from our 'lm' call since R's 'lm' is forming the **d** matrix too:

```
d=t(model.matrix(fit))
```

Now we solve for the parameters using the code in section 1.1.4:

```
y%*%t(d)%*%solve(d%*%t(d))
```

```
     regionnorth regionsouth
[1,]    25.33333          27
```

```
coef(lm(Ozone ~ -1 + region, data=homeworkdat))
```

```
regionnorth regionsouth
   25.33333    27.00000
```

Problem 6, part 1

Write the model in question 5 in Form 2

The reason we had to use a matrix with 1s and 0s to tell our matrix math what α to use is that Form 1 and Form 1b have the parameters appearing once in a column vector or a row vector. So we need a matrix with 1s and 0s to say 'where' to put the αs.

In Form 2, we can have the parameters just repeat in our matrix. This is how we'd write the model on the whiteboard. We'd just have the α (intercept) column have multiple αs in it. Section 1.3.2 shows you how to have different intercepts in Form 2. The model in question 5 is just like in section 1.3.2, but without the 'air'. It just has different monthly intercepts.

The model is:

$$\begin{bmatrix} Ozone_1 \\ Ozone_2 \\ Ozone_3 \\ Ozone_4 \\ Ozone_5 \end{bmatrix} = \begin{bmatrix} \alpha_n \\ \alpha_s \\ \alpha_n \\ \alpha_s \\ \alpha_n \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{bmatrix} = \mathbf{Zx} + \mathbf{e}$$

Problem 6, part 2

Solve for the parameters Solve for the parameters

To do this we write $\mathbf{y}$, $\mathbf{Z}$, and $\mathbf{x}$ in R and use the code in section 1.2.3.

```
#the number of data points
n=5
y=matrix(homeworkdat$Ozone, ncol=1)
x=matrix(1)
Z=matrix(paste("alpha",homeworkdat$region, sep="."),ncol=1)
```

Then we solve for the parameters:

```
require(MARSS)
P=MARSS:::convert.model.mat(Z)$free[,,1]
M=kronecker(t(x),diag(n))%*%P
solve(t(M)%*%M)%*%t(M)%*%y
```

```
               [,1]
alpha.north 25.33333
alpha.south 27.00000
```

```
 coef(lm(Ozone ~ -1 + region, data=homeworkdat))
```

```
regionnorth regionsouth
   25.33333    27.00000
```

Problem 7

Write the model below in Form 2 and solve for the parameters

```
 fit=lm(Ozone ~ Temp:region, data=homeworkdat)
```

The first step is to write out what model is being fit. The model is

$$Ozone_i = \alpha + \beta_n Temp_i + e_i$$

if i is from the north and

$$Ozone_i = \alpha + \beta_s Temp_i + e_i$$

if i is from the south. So each region has the same intercept α but we are including a linear temperature effect that is different for each region.

This is just like Equation 1.26 but with one α:

$$\begin{bmatrix} Ozone_1 \\ Ozone_2 \\ Ozone_3 \\ Ozone_4 \\ Ozone_5 \end{bmatrix} = \begin{bmatrix} \alpha & \beta_n & 0 & 0 & 0 & 0 \\ \alpha & 0 & \beta_s & 0 & 0 & 0 \\ \alpha & 0 & 0 & \beta_n & 0 & 0 \\ \alpha & 0 & 0 & 0 & \beta_s & 0 \\ \alpha & 0 & 0 & 0 & 0 & \beta_n \end{bmatrix} \begin{bmatrix} 1 \\ Temp_1 \\ Temp_2 \\ Temp_3 \\ Temp_4 \\ Temp_5 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \dots \\ e_{111} \end{bmatrix} = \mathbf{Zx} + \mathbf{e}$$

Solving for this is a matter of writing $\mathbf{y}$, $\mathbf{Z}$, and $\mathbf{x}$ in R.

```
n=5 #the number of data points
y=matrix(homeworkdat$Ozone, ncol=1)
x=matrix(c(1, homeworkdat$Temp),ncol=1)
#Set up Z; it is 5 x 6
Z=matrix(list(0),n,n+1)
#make the alpha column
Z[,1]="alpha"
#make the diagonal of columns 2:6 equal to the betas
diag(Z[,2:(1+n)])=paste("beta",homeworkdat$region, sep=".")
```

Then we solve for the parameters:

```
require(MARSS)
P=MARSS:::convert.model.mat(Z)$free[,,1]
M=kronecker(t(x),diag(n))%*%P
solve(t(M)%*%M)%*%t(M)%*%y
```

```
                [,1]
alpha       23.86230424
beta.north  0.01510679
beta.south  0.05654090
```

```
 coef(lm(Ozone ~ Temp:region, data=homeworkdat))
```

```
    (Intercept) Temp:regionnorth Temp:regionsouth
    23.86230424       0.01510679       0.05654090
```

Problem 8**

Using the airquality dataset with 111 data points, write the model below in Form 2 and solve for the parameters.

```
 fit=lm(Ozone ~ -1 + Temp:region + Month, data=airquality)
```

The first step is to write out what model is being fit.

$$Ozone_i = \alpha_j + \beta_k Temp_i + e_i$$

If i is from the j-th month, the intercept is α_j. If i is from the north, $beta_k$ is β_n. If it is from the south, $beta_k$ is β_s.

Next, let's look at region and Month.

```
airquality$region[1:10]
```

```
[1] "north" "south" "north" "south" "north" "south" "north"
[8] "south" "north" "south"
```

```
airquality$Month
```

```
  [1] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6 6
 [28] 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
 [55] 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
 [82] 8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
[109] 9 9 9
Levels: 5 6 7 8 9
```

Region is alternating north/south and month is grouped.

So in matrix form our model looks like

$$
\begin{bmatrix} Ozone_1 \\ Ozone_2 \\ \dots \\ Ozone_{111} \end{bmatrix} =
\begin{bmatrix}
\alpha_5 & \beta_n & 0 & \dots & 0 & 0 \\
\alpha_5 & 0 & \beta_s & \dots & 0 & 0 \\
\dots & \dots & \dots & \ddots & \dots & 0 \\
\alpha_9 & 0 & 0 & \dots & \beta_s & 0 \\
\alpha_9 & 0 & 0 & \dots & 0 & \beta_n
\end{bmatrix}
\begin{bmatrix} 1 \\ Temp_1 \\ Temp_2 \\ \dots \\ Temp_{111} \end{bmatrix} +
\begin{bmatrix} e_1 \\ e_2 \\ \dots \\ e_{111} \end{bmatrix} = \mathbf{Z}\mathbf{x} + \mathbf{e}
$$

Solving for this is a matter of writing $\mathbf{y}$, $\mathbf{Z}$, and $\mathbf{x}$ in R. We don't have to create character strings for our parameter names. We just use the text in our data.frame and use the `paste()` function.

```
n=111 #the number of data points
y=matrix(airquality$Ozone, ncol=1)
x=matrix(c(1, airquality$Temp),ncol=1)
#Set up Z; it is 111 x 112
Z=matrix(list(0),n,n+1)
#make the alpha column
Z[,1]=paste("alpha",airquality$Month, sep="")
#make the diagonal of columns 2:112 equal to the betas
diag(Z[,2:(1+n)])=paste("beta",airquality$region, sep=".")
```

Then we solve for the parameters:

```
require(MARSS)
P=MARSS:::convert.model.mat(Z)$free[,,1]
M=kronecker(t(x),diag(n))%*%P
solve(t(M)%*%M)%*%t(M)%*%y
```

```
                 [,1]
alpha5      -160.254743
alpha6      -187.708278
alpha7      -173.611725
alpha8      -172.152400
alpha9      -181.929911
beta.north     2.790616
beta.south     2.758379
```

```
 coef(lm(Ozone ~ -1 + Temp:region + Month, data=airquality))
```

```
        Month5            Month6            Month7
    -160.254743      -187.708278      -173.611725
        Month8            Month9 Temp:regionnorth
    -172.152400      -181.929911         2.790616
Temp:regionsouth
       2.758379
```