Applying lm()

Vaibhav Sherkar (BS2041)

Now, we fit the following model to the data.

$$l_{ij} = \alpha_i + \beta w_{ij} + \epsilon_{ij}$$

Above model assess the effect of variables lab and weight on length. Note the absence of intercept term in the model. Now, we fit this model to our data using lm() function in R. α_i's correspond to the variable 'lab' in the dataset. β corresponds to the variable 'weight'. Intercept term is absent in the model. Hence, we write the following code in R.

```
> fit=lm(length ~ lab+weight-1,data=alllab)
```

Note the presence of -1 inside the lm function due to absence of intercept term in the model. Now we look at the estimates of coefficients. We type fit in R. Output is as follows:

```
> fit

Call:
lm(formula = length ~ lab + weight - 1, data = alllab)

Coefficients:
lab1    lab2  weight
3.276   5.173   2.013
```

The estimates are $\hat{\alpha_1} = 3.276$, $\hat{\alpha_2} = 5.173$, $\hat{\beta} = 2.013$

We now look at the design matrix which is used to fit this model. We know that in the data, first 6 observations come from lab 1 and last 5 observations from lab 2. We are estimating the coefficients $\alpha_1, \alpha_2, \beta$. Hence the design matrix looks like

$$\begin{bmatrix} 1 & 0 & w_{11} \\ 1 & 0 & w_{12} \\ 1 & 0 & w_{13} \\ 1 & 0 & w_{14} \\ 1 & 0 & w_{15} \\ 1 & 0 & w_{16} \\ 0 & 1 & w_{21} \\ 0 & 1 & w_{22} \\ 0 & 1 & w_{23} \\ 0 & 1 & w_{24} \\ 0 & 1 & w_{25} \end{bmatrix}$$

Now, we look at the design matrix used by R to fit the model 'fit' to the data using model.matrix() function.

```
> model.matrix(fit)
   lab1 lab2 weight
1     1    0    1.0
2     1    0    1.5
3     1    0    2.0
4     1    0    2.5
5     1    0    3.0
6     1    0    3.5
7     0    1    1.2
8     0    1    1.5
9     0    1    1.8
10    0    1    2.1
11    0    1    2.1
attr(,"assign")
[1] 1 1 2
attr(,"contrasts")
attr(,"contrasts")$lab
[1] "contr.treatment"
```

The design matrix used by R for fitting the model is exactly same as expected design matrix.