Lsn 16

Clark

Admin

Up to this point we have been using statistical models of the form:

Key to this has been the assumption that our explanatory variable, or independent variable, is categorical. This was nice in that we could think of each of our observations as coming from a group with seperate means. For instance we can think of H_0 and H_a from above as:

However, in some studies our explained variation comes from a variable that has a natural ordering to it. In fact we've seen this a bit already (recall Pistachio study).

Grape Seeds:

Note here our researchers are interested in explaining the variation in the amount of proanthocyanidin (PC) in a grape seed and the sources of the explained variation might be thought of as the percentage of ethanol.

```r
grape<-read.table("http://www.isi-stats.com/isi2/data/Polyphenols.txt",header=T)
grape <- grape %>% mutate(Ethanol=Ethanol.) %>% select(-Ethanol.)
grape <- grape %>% mutate(Time.hrs=Time.hrs.)%>% select(-Time.hrs.)
```

Note that the null model here is:

Which can be fit as:

```r
null.lm<-lm(PC~1,data=grape)
summary(null.lm)
```

```
##
## Call:
## lm(formula = PC ~ 1, data = grape)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -89.20 -20.25   12.40   27.60   68.90
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   237.90      11.49   20.71  6.7e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 44.49 on 14 degrees of freedom
```

Perhaps we fit a seperate means model:

Which is done through:

```
sep.means<-lm(PC~0+as.factor(Ethanol),data=grape)
summary(sep.means)
```

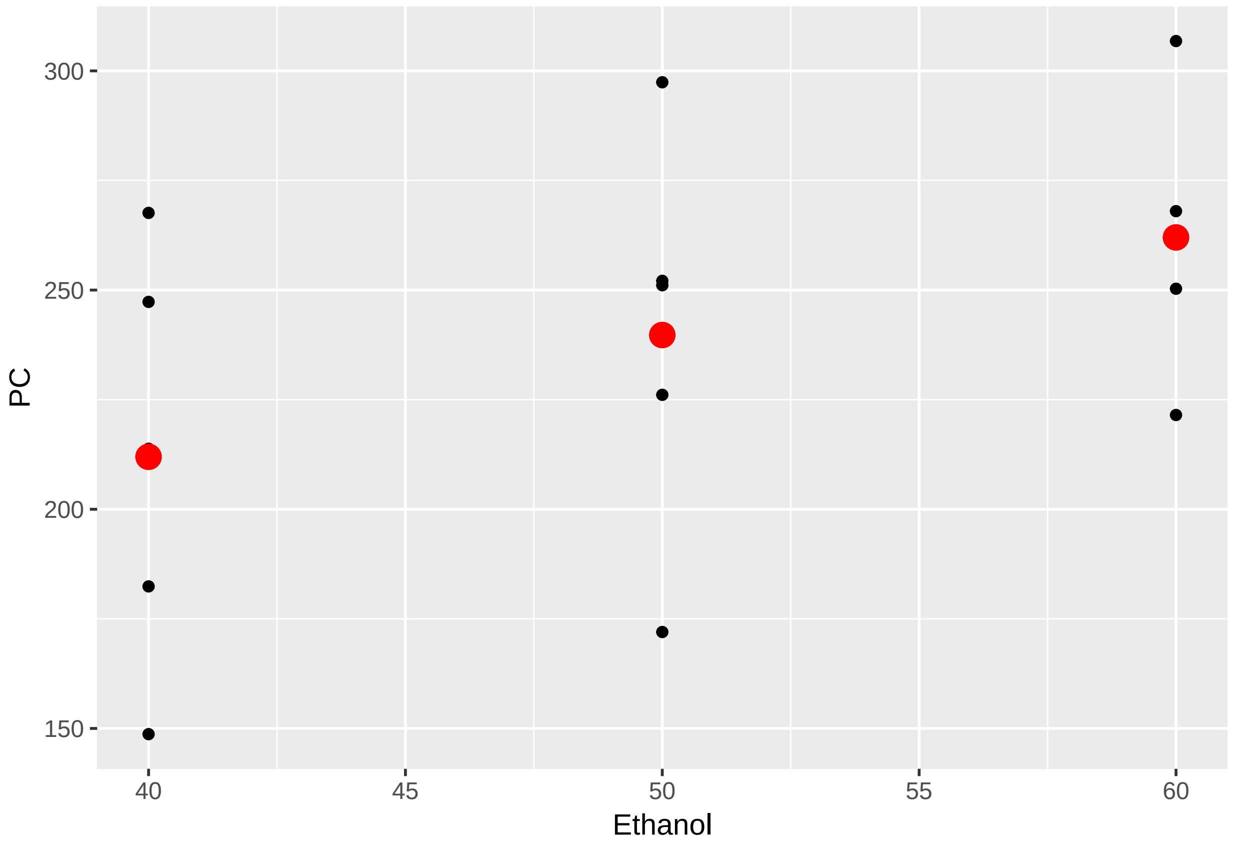
```
##
## Call:
## lm(formula = PC ~ 0 + as.factor(Ethanol), data = grape)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -67.74 -21.60   1.84  23.85  57.66
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## as.factor(Ethanol)40    212.0       18.9   11.22 1.02e-07 ***
## as.factor(Ethanol)50    239.7       18.9   12.69 2.60e-08 ***
## as.factor(Ethanol)60    262.0       18.9   13.86 9.53e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.26 on 12 degrees of freedom
## Multiple R-squared:  0.9756, Adjusted R-squared:  0.9694
## F-statistic: 159.7 on 3 and 12 DF,  p-value: 6.187e-10
```

Is this better? Well, perhaps. To test this we can run:

```
grape<-grape %>% mutate(Ethanolf=as.factor(Ethanol))
contrasts(grape$Ethanolf)<-contr.sum
effects.mod<-lm(PC~Ethanolf,data=grape)
anova(effects.mod)
```

```
## Analysis of Variance Table
##
## Response: PC
##           Df  Sum Sq Mean Sq F value Pr(>F)
## Ethanolf   2  6285.4  3142.7    1.76 0.2137
## Residuals 12 21427.1  1785.6
```

```
gr.means <- grape %>% group_by(Ethanolf)%>%summarize(means=mean(PC))%>%
  mutate(Ethanol=as.numeric(as.character(Ethanolf)))
grape %>% ggplot(aes(x=Ethanol,y=PC))+geom_point()+geom_point(aes(x=Ethanol,y=means),color="red",size=4
```



If our means have a linear relationship, perhaps our model could be:

Which is of course the model for a regression. Note that this is the statistical model. The fitted model or the predicted model is:

Our book uses b_0 and b_1 instead of $\hat{\beta}_0$ and $\hat{\beta}_1$. I prefer the hats, but use what you'd like.

To fit this model we simply run:

```
reg.lm<-lm(PC~Ethanol,data=grape)
summary(reg.lm)
```

```
## 
## Call:
## lm(formula = PC ~ Ethanol, data = grape)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
## -65.90 -21.55   0.92  24.31  59.50 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)  
## (Intercept)  112.800     65.081   1.733   0.1067  
## Ethanol        2.502      1.285   1.948   0.0734 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 40.62 on 13 degrees of freedom
## Multiple R-squared:  0.2259, Adjusted R-squared:  0.1663 
## F-statistic: 3.793 on 1 and 13 DF,  p-value: 0.07338
```

Note that we obtain $\hat{\beta}$ through the method of least squares:

The interpretation of $\hat{\beta}_0$ and $\hat{\beta}_1$ are:

One often overlooked aspect of using regression vs ANOVA is that the linear regression model is actually more restrictive than the seperate means model:

```
anova(reg.lm)
```

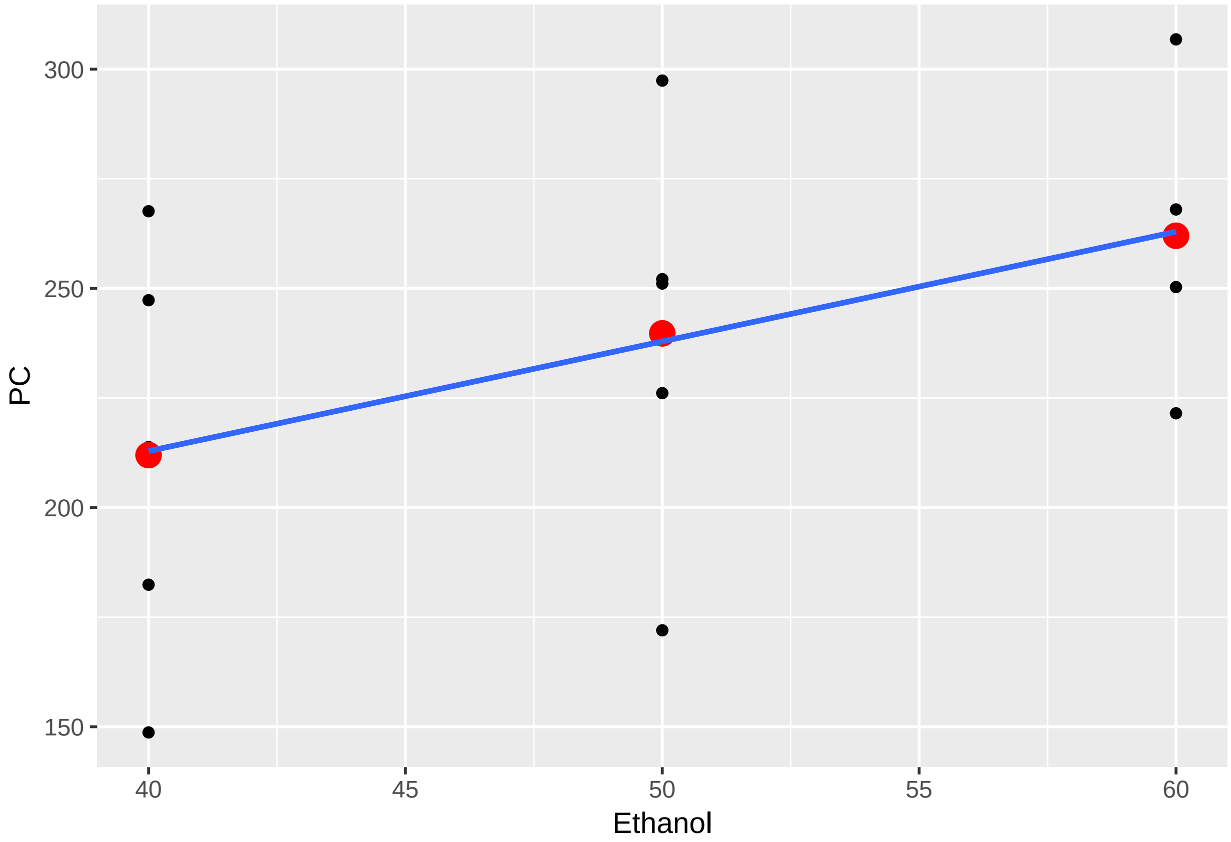
```
## Analysis of Variance Table
## 
## Response: PC
##           Df Sum Sq Mean Sq F value  Pr(>F)  
## Ethanol    1   6260  6260.0  3.7935 0.07338 .
## Residuals 13  21453  1650.2                  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that the sums of squares are higher under this model. Why?

However, the SE of the residuals is smaller. When we use a regression model vs a seperate means model we are making a tradeoff. We are actually building a simpler model, but hoping that the additional complexity a multiple means model provides minimal difference in explaining variability. **All things being equal we always prefer a simpler model**. If we have two models that have similar SSError values, choose the model that uses fewer degrees of freedom if you don't have science to save you.

To put this another way, previously if we had μ, α_1 and α_2 we automatically knew α_3 (why?). Now, if we know one of our means and β_0 we know all of our other means.

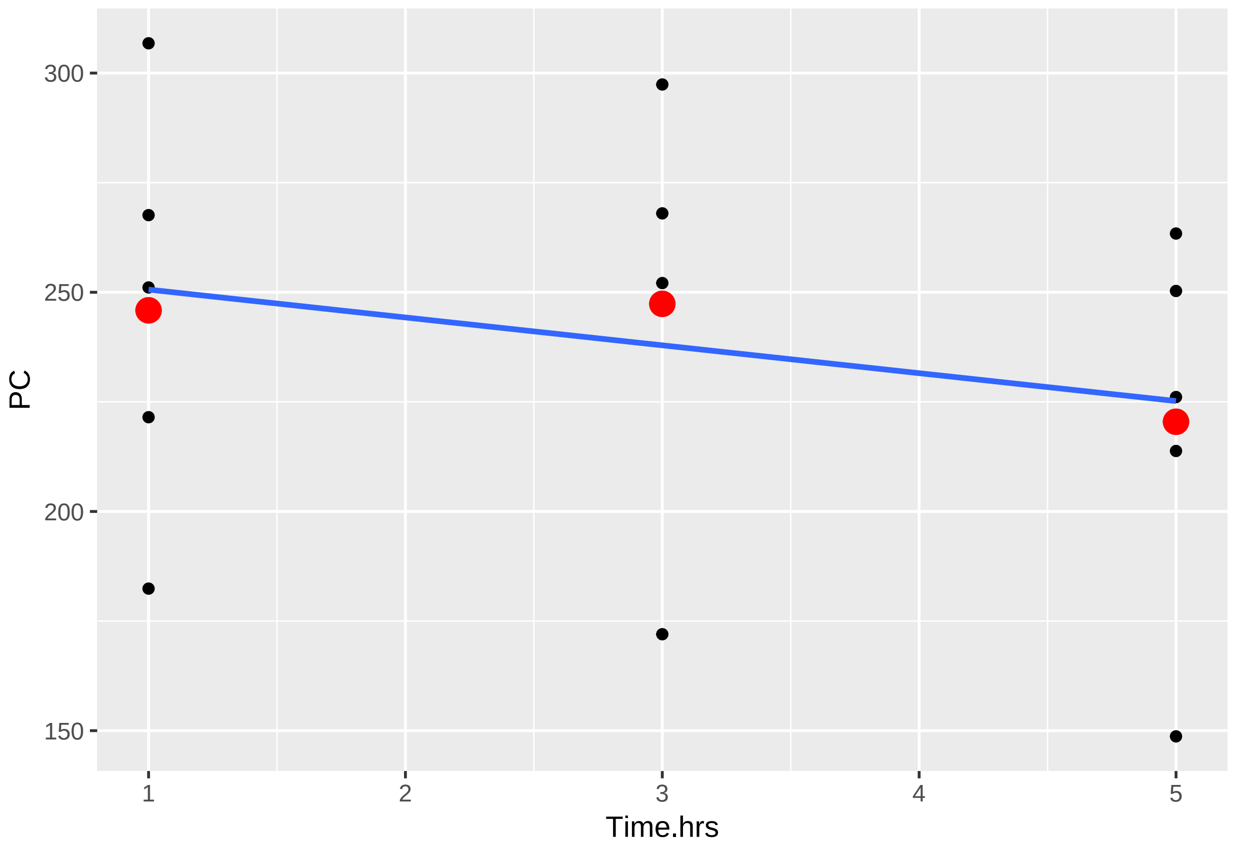
```
grape %>% ggplot(aes(x=Ethanol,y=PC))+geom_point()+
  geom_point(aes(x=Ethanol,y=means),color="red",size=4,data=gr.means)+
  stat_smooth(method="lm", se=FALSE)
```



In later lessons we'll explore the statistical properties of $\hat{\beta}$ and the linear regression model. The key point here is that a linear regression model is appropriate if we believe ther is a linear relationship between our explanatory variable and our response. It isn't always clear whether a seperate means model outperforms a linear regresison model though:

```
grape<-grape %>% mutate(Timef=as.factor(Time.hrs))
reg.lm<-lm(PC~Time.hrs,data=grape)
sep.means<-lm(PC~Timef,data=grape)
```

```
gr.means <- grape %>% group_by(Timef)%>%summarize(means=mean(PC))%>%
  mutate(Time=as.numeric(as.character(Timef)))
grape %>% ggplot(aes(x=Time.hrs,y=PC))+geom_point()+
  geom_point(aes(x=Time,y=means),color="red",size=4,data=gr.means)+
  stat_smooth(method="lm",se=FALSE)
```


300
250
200
150
PC
1
2
3
4
5
Time.hrs