R Codes for Chapter-8

1. Polynomial Regression in R

The function **"lm"** can be used to perform polynomial regression in R and much of the syntax is the same as that used for fitting other regression models. To perform second order polynomial regression with **2** explanatory variables use the command:

lm(response ~ explanatory_1 + explanatory_2 + explanatory_1^2 + explanatory_2^2+ explanatory_1*+ explanatory_2)

Here the terms response and explanatory in the function should be replaced by the names of the response and explanatory variables, respectively, used in the analysis.

Ex. Power cells example: Y= number of cycles, X1=charge rate, X2=Temperature.

The following program reads in the data.

```
> # Data:
> Y=c(150,86,49,288,157,131,184,109,279,235,224)
> X1=c(.6,1,1.4,.6,1,1,1,1.4,.6,1,1.4)
> X2=c(10,10,10,20,20,20,20,20,30,30,30)
```

Centralizing the predictors:

Here we scale the data instead of centralize.

```
> Xbar1=mean(X1)
> Xbar2=mean(X2)
> x1=(X1-Xbar1)/.4
> x2=(X2-Xbar2)/10
```

The following shows the difference between the correlations of X and X^2 for original variables and for scaled variables.

```
> cor(X1,X1^2)
[1] 0.9910312
> cor(x1,x1^2)
[1] -4.042173e-16
>
> cor(X2,X2^2)
[1] 0.9860911
> cor(x2,x2^2)
[1] 0
```

Note that the correlations for the scaled data (usually centralized data) are small (here those are zero because of the symmetric distribution of the values.)

Fitting the model (Second order polynomial Regression Model):

First define square and product terms.

```
> x1_2=x1^2
> x2_2=x2^2
> x12=x1*x2
```

Then fit the model.

```
> prm=lm(Y~x1+x2+x1_2+x2_2+x12)
> prm

Call:
lm(formula = Y ~ x1 + x2 + x1_2 + x2_2 + x12)

Coefficients:
(Intercept)           x1           x2         x1_2         x2_2          x12
     162.84       -55.83        75.50        27.39       -10.61        11.50
```

Summary output:

```
> summary(prm)

Call:
lm(formula = Y ~ x1 + x2 + x1_2 + x2_2 + x12)

Residuals:
      1       2       3       4       5       6       7       8       9      
10      11
-21.465   9.263  12.202  41.930  -5.842 -31.842  21.158 -25.404 -20.465   7.2
63  13.202

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   162.84      16.61   9.805 0.000188 ***
x1            -55.83      13.22  -4.224 0.008292 ** 
x2             75.50      13.22   5.712 0.002297 ** 
x1_2           27.39      20.34   1.347 0.235856    
x2_2          -10.61      20.34  -0.521 0.624352    
x12            11.50      16.19   0.710 0.509184    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 32.37 on 5 degrees of freedom
Multiple R-squared:  0.9135,	Adjusted R-squared:  0.8271 
F-statistic: 10.57 on 5 and 5 DF,  p-value: 0.01086
```

Lack of Fit Test:

```
> fit<-lm(Y~x1+x2+x1_2+x2_2+x12)
> exfactor = factor( c(seq(-4,-1), rep(0,3),seq(1,4)) )
> #fit full model
> anova( fit, lm(Y ~ exfactor))
```

```
Analysis of Variance Table

Model 1: Y ~ x1 + x2 + x1_2 + x2_2 + x12
Model 2: Y ~ exfactor
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1      5 5240.4
2      2 1404.7  3    3835.8 1.8205 0.3738
```

Here note that p-value for lack of fit is 0.3738 (> alpha). So Conclude H0. That is, the second order polynomial function is a good fit.

Extended ANOVA Table:

```
> anova(prm)
Analysis of Variance Table

Response: Y
          Df Sum Sq Mean Sq F value   Pr(>F)
x1         1  18704   18704 17.8460 0.008292 **
x2         1  34202   34202 32.6323 0.002297 **
x1_2       1   1646    1646  1.5704 0.265552
x2_2       1    285     285  0.2719 0.624352
x12        1    529     529  0.5047 0.509184
Residuals  5   5240    1048
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Partial F Test:

Here we test whether first order model is sufficient or not. Note that full model is prm. Then we fit the reduced model.

```
> redm=lm(Y~x1+x2)
> redm

Call:
lm(formula = Y ~ x1 + x2)

Coefficients:
(Intercept)           x1           x2
     172.00       -55.83        75.50
```

The ANOVA table for two models.

```
> anova(redm,prm)

Analysis of Variance Table

Model 1: Y ~ x1 + x2
Model 2: Y ~ x1 + x2 + x1_2 + x2_2 + x12
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1      8 7700.3
2      5 5240.4  3    2459.9 0.7823 0.5527
```

Here note that the p-value for partial F test is 0.5527 (>alpha). So we conclude H0: beta3=beta4 =beta5=0. That is, square terms and the interaction term can be dropped from the model (the linear model is sufficient).

Estimation of Regression Coefficients:

Here we calculate 90% family confidence coefficients by the Bonferroni method.

```
> g = length(coef(lm(Y ~ X1+X2)) ) - 1
> alpha=0.1  #for 90% confidence intervals.
> confint( firom, level = 1-(alpha/g) )
                  2.5 %    97.5 %
(Intercept)   64.617930 256.54874
X1          -212.602056 -66.56461
X2             4.629251  10.47075
```

So 90% Bonferroni family confidence intervals for beta1 and beta2 are (-212.602, -66.565) and (4.629, 10.471) respectively.

2. Regression Models with Qualitative Predictors in R

Here also the function "**lm**" can be used to fit a regression model with qualitative predictors in R.

<u>Ex.</u> Insurance Innovation Example: Y- # of months elapsed, X1- size of firm, X2- type of firm (stock company, mutual company).

The following command imports the data into R.

```
> data=read.table("R:\\Teaching\\2016\\MA 542\\Class preperation\\Insurance.c
sv",header = FALSE)
```

The following command adds names "Y","X1" and "X2" to corresponding columns.

```
> colnames(data)<-c("Y","X1","X2")
> attach(data)
```

Fitting the Model:

Here corresponding indicator variables need to be defined before fit the model. Then fit the model as follows.

```
> fit<-lm(Y~X1 +X2)
> fit

Call:
lm(formula = Y ~ X1 + X2)

Coefficients:
(Intercept)           X1           X2
     160.58      -139.58         7.55
```

```
> summary(fit)

Call:
lm(formula = Y ~ X1 + X2)

Residuals:
    Min      1Q  Median      3Q     Max 
-41.000 -13.750  -7.167  10.167  60.167 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  160.583     41.615   3.859 0.004817 ** 
X1          -139.583     31.665  -4.408 0.002262 ** 
X2             7.550      1.267   5.961 0.000338 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 31.02 on 8 degrees of freedom
Multiple R-squared:  0.8729,   Adjusted R-squared:  0.8412 
F-statistic: 27.48 on 2 and 8 DF,  p-value: 0.0002606
```

ANOVA Table:

```
> anova(fit)
Analysis of Variance Table

Response: Y
          Df Sum Sq Mean Sq F value    Pr(>F)    
X1         1  18704   18704  19.432 0.0022619 ** 
X2         1  34201   34201  35.532 0.0003378 ***
Residuals  8   7700     963                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
```

Fitting a Model with Interaction Term:

```
> intfit=lm(Y~X1+X2+X1*X2)
> intfit

Call:
lm(formula = Y ~ X1 + X2 + X1 * X2)

Coefficients:
(Intercept)           X1           X2        X1:X2  
    218.083     -197.083        4.675        2.875  


> summary(intfit)

Call:
lm(formula = Y ~ X1 + X2 + X1 * X2)

Residuals:
   Min     1Q Median     3Q    Max 
-41.00 -13.33 -10.50  15.92  60.17 
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)  
(Intercept)  218.083     90.809   2.402   0.0474 *
X1          -197.083     86.430  -2.280   0.0566 .
X2             4.675      4.209   1.111   0.3034  
X1:X2          2.875      4.001   0.719   0.4957  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 32.01 on 7 degrees of freedom
Multiple R-squared:  0.8817,  Adjusted R-squared:  0.831 
F-statistic: 17.39 on 3 and 7 DF,  p-value: 0.001264
```