Problem 1

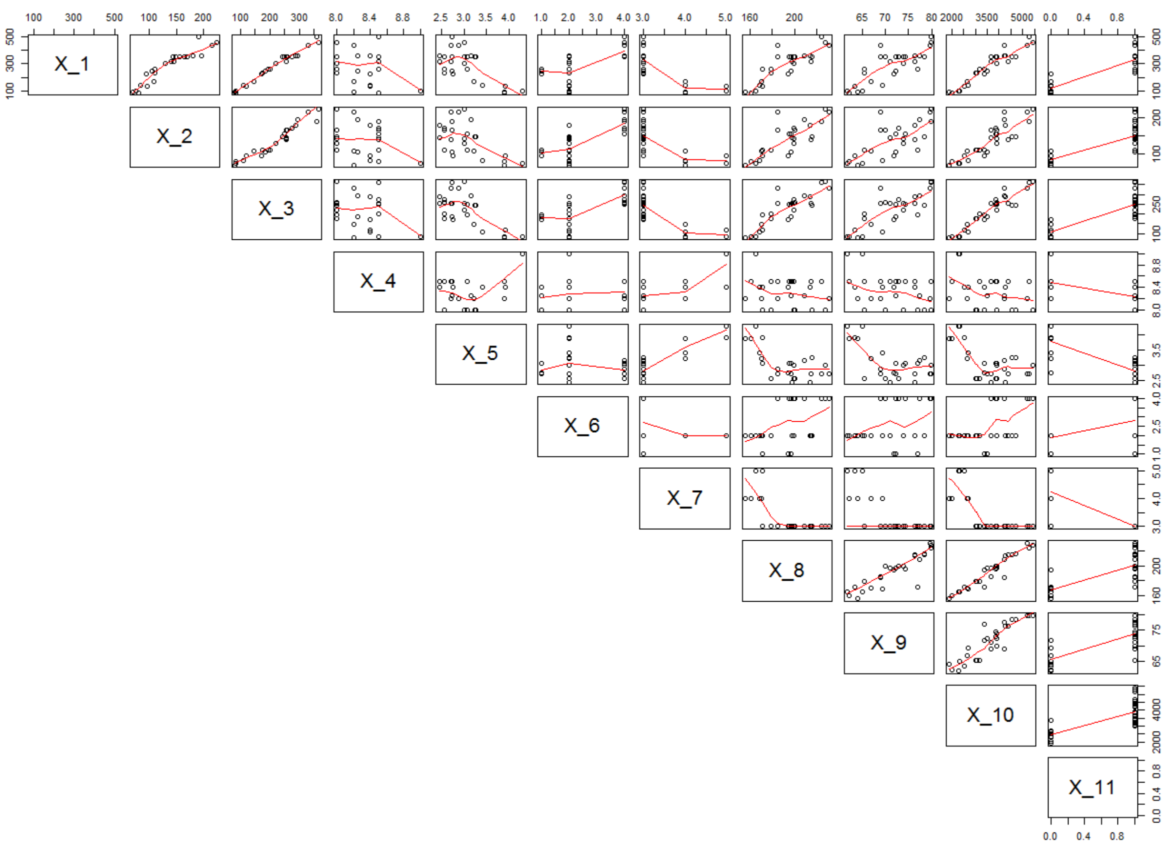
Part a.

```
> corr = round(cor(mydata[-1]),2)
> >
> pairs(mydata[,-1], main = "Correlation coeffficients matrix and scatter plot",
+       pch = 21, lower.panel = NULL, panel = panel.smooth, cex.labels=2)
>
> # Alternative
> corrplot.mixed(corr, upper = "ellipse", lower = "number")

> # The pairwise correlation coefficients of the predictor variables
> # and the corresponding scatter plots show strong linear relationships
> # among pairs of predictors variables, suggesting collinearity.
> # For example, x_1 has |correlations| of greater than |0.6| with all
> # other predictors except x_4. Looking at the scatter plot, a clear
> # linear relationship between x_1 and x_2, x_3, x_8, x_9 and x_10
> # can be seen.
> # Other notable correlations from the scatter plots are:
> # x_2 with x_3, x_8, x_9 and x_10
> # x_3 with x_8, x_9 and x_10
> # x_8 with x_9 and x_10
> # x_9 with x_10
> # Other predictors also exhibit collinearity based on the correlation
> # coefficient (e.g. x7 and x11 have a value of -0.85)
> # For x_8, x_9 and x_10, the correlations with the other
> # predictors seems to follow a similar patter across
> # x_8, x_9 and x_10 for each of the other predictors.
```

Part b.

```
> fit = lm(Y~.,mydata)
> summary(fit)
lm(formula = Y ~ ., data = mydata)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 17.773204  30.508775   0.583   0.5674
X_1         -0.077946   0.058607  -1.330   0.2001
X_2         -0.073399   0.088924  -0.825   0.4199
X_3          0.121115   0.091353   1.326   0.2015
X_4          1.329034   3.099535   0.429   0.6732
X_5          5.975989   3.158647   1.892   0.0747 .
X_6          0.304178   1.289094   0.236   0.8161
X_7         -3.198576   3.105435  -1.030   0.3167
X_8          0.185362   0.129252   1.434   0.1687
X_9         -0.399146   0.323812  -1.233   0.2336
X_10        -0.005193   0.005893  -0.881   0.3898
X_11         0.598655   3.020681   0.198   0.8451
---
> # Compute VIF
> library(car)
> vif(fit)
       X_1        X_2        X_3        X_4        X_5        X_6        X_7        X_8
128.834832  43.921063 160.436093   2.057834   7.780750   5.326714  11.735038  20.585810
       X_9       X_10       X_11
  9.419449  85.675755   5.142547
>
> # Determine VIF > 10
> names(vif(fit))[vif(fit)>10]
[1] "X_1"  "X_2"  "X_3"  "X_7"  "X_8"  "X_10"
> # It appears that X1, X2, X3, X7, X8 and X10 are affected by the
> # presence of collinearity because VIF > 10.
```

Correlation coeffficients matrix and scatter plot



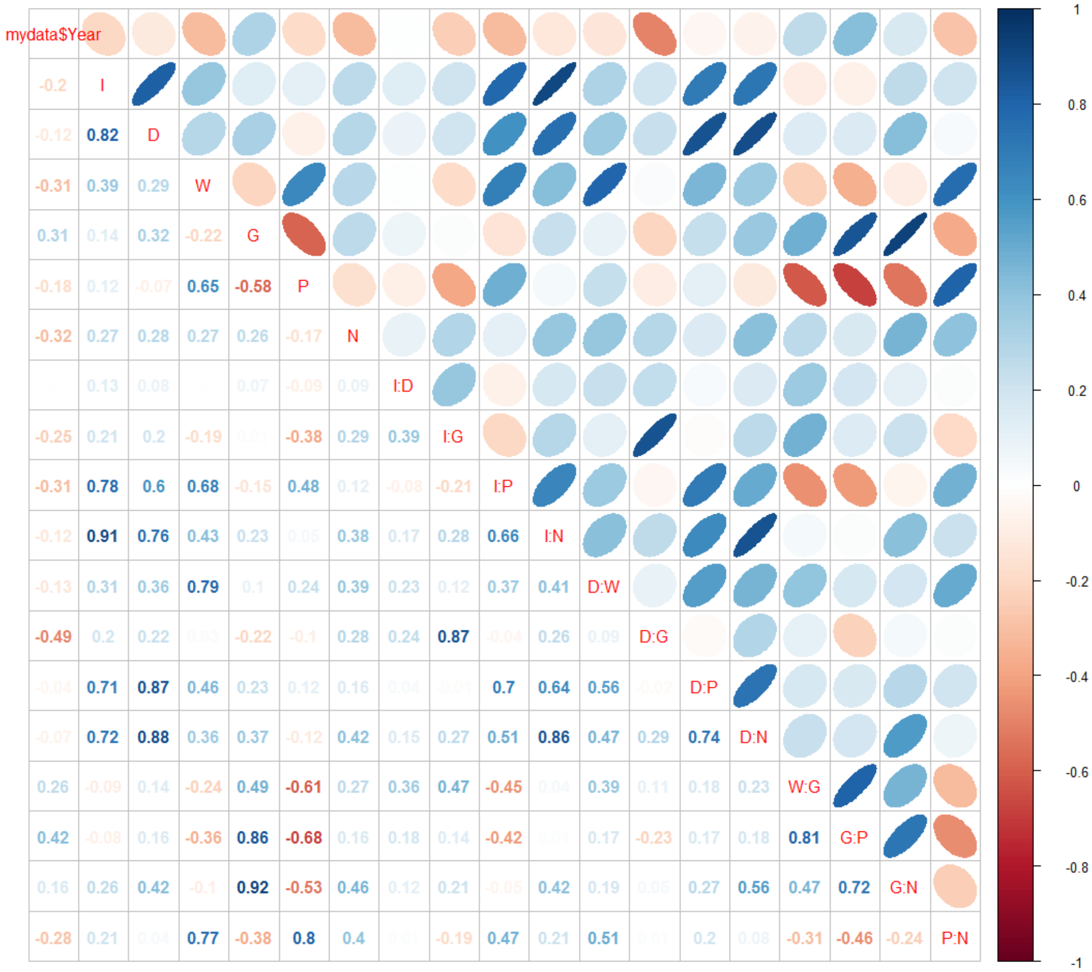
Problem 2

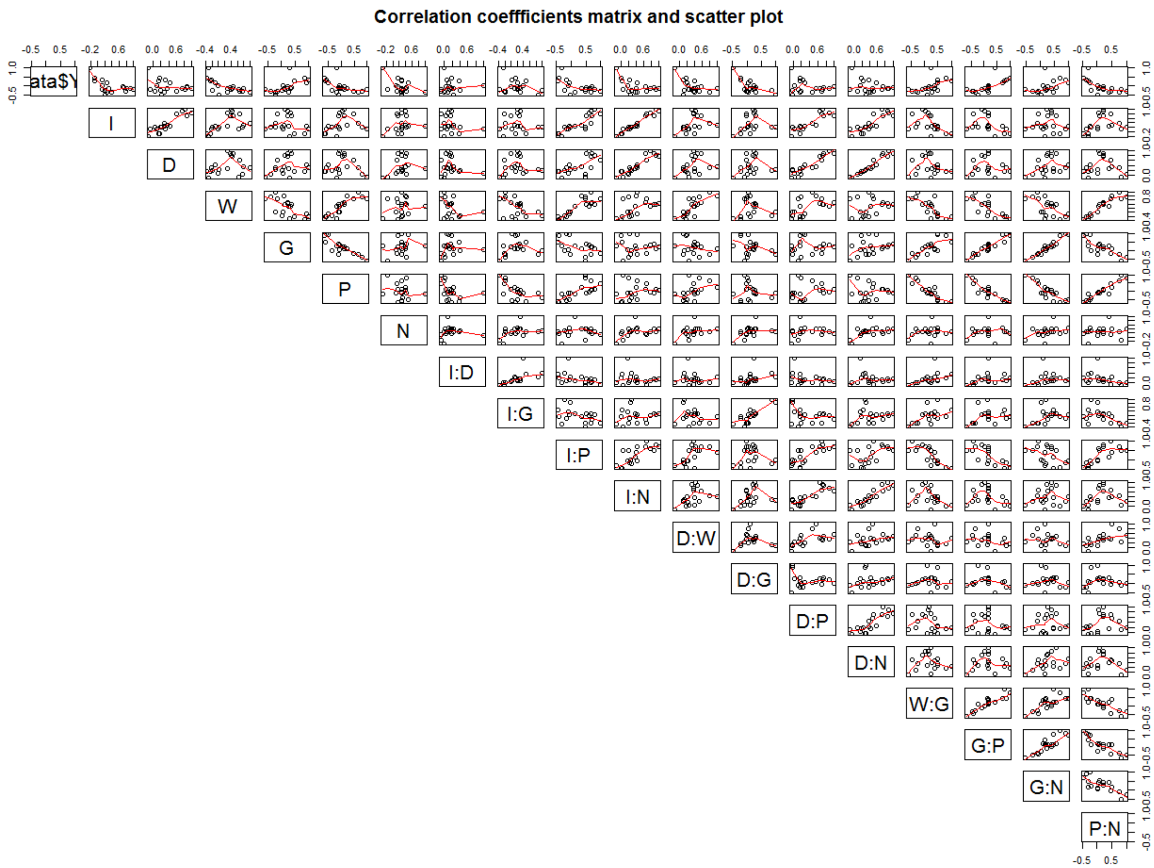
Part a.

```
> # Maximum number of terms (coefficients) in a linear regression with
> # n number of observations:
> # - equal to n will give 0 df (perfect fit) (n-1 predictors)
> # - equal to n-1 will give 1 df (n-2 predictors)
> # In this example, for a perfect fit, the max number of terms is 21
> # (20 predictors), and for at least 1 df so that inferences can be done, the max
> # number of terms is 20 (19 predictors).
> # 19 predictors were fitted for the next part
```

Part b.

```
> # Fit a model with year, and as many two-way interaction terms (including the
> # main effects) as possible
> fit = lm(V~mydata$Year+.*.,mydata[-1])
> fit = update(fit,.~.-I:W)
> fit = update(fit,.~.-W:P)
> fit = update(fit,.~.-W:N)
> summary(fit)

Call:
lm(formula = V ~ mydata$Year + I + D + W + G + P + N + I:D +
    I:G + I:P + I:N + D:W + D:G + D:P + D:N + W:G + G:P + G:N +
    P:N, data = mydata[-1])


Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.3774121  2.4630195  -0.559    0.675
mydata$Year  0.0009159  0.0011568   0.792    0.574
I           -0.3891546  0.4365911  -0.891    0.537
D            0.3918810  0.4388925   0.893    0.536
W           -1.0662691  1.9082105  -0.559    0.676
G           -0.0178859  0.0268871  -0.665    0.626
P            0.0218266  0.0638922   0.342    0.790
N            0.0011055  0.0385081   0.029    0.982
I:D         -0.0002413  0.0419984  -0.006    0.996
I:G         -0.0149306  0.0269669  -0.554    0.678
I:P          0.0512403  0.0504937   1.015    0.495
I:N          0.0477215  0.0674762   0.707    0.608
D:W          1.0520223  2.2268291   0.472    0.719
D:G          0.0323322  0.0322257   1.003    0.499
D:P         -0.0647882  0.0547079  -1.184    0.446
D:N         -0.0279259  0.0672911  -0.415    0.750
W:G         -0.0494660  0.1535681  -0.322    0.802
G:P          0.0050588  0.0053601   0.944    0.518
G:N         -0.0009287  0.0022219  -0.418    0.748
P:N         -0.0006583  0.0101308  -0.065    0.959

Residual standard error: 0.04703 on 1 degrees of freedom
Multiple R-squared: 0.9804,    Adjusted R-squared: 0.6075
F-statistic: 2.629 on 19 and 1 DF,  p-value: 0.4553

> # Correlation matrix of predictor variables
> corr = cor(model.matrix(fit)[,-1]) # corr for all predictors (e.g. interaction)
> # corr = round(cor(mydata[-2]),2)  # corr for variables in dataset
>
> corrplot.mixed(corr, upper = "ellipse", lower = "number")
>
> pairs(corr, main = "Correlation coeffficients matrix and scatter plot",
+       pch = 21, lower.panel = NULL, panel = panel.smooth, cex.labels=2)
```

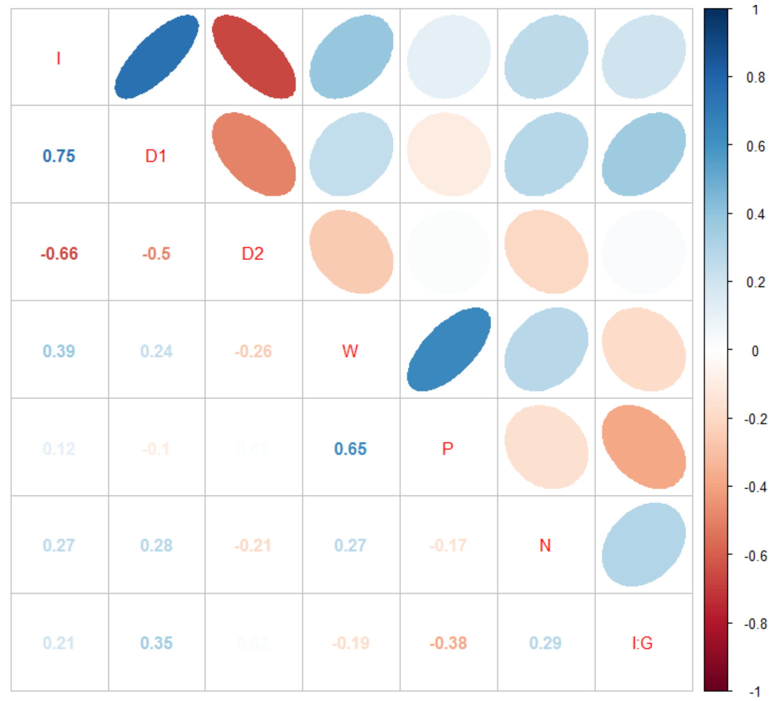

mydata$Year
I
D
W
G
P
N
I:D
I:G
I:P
I:N
D:W
D:G
D:P
D:N
W:G
G:P
G:N
P:N

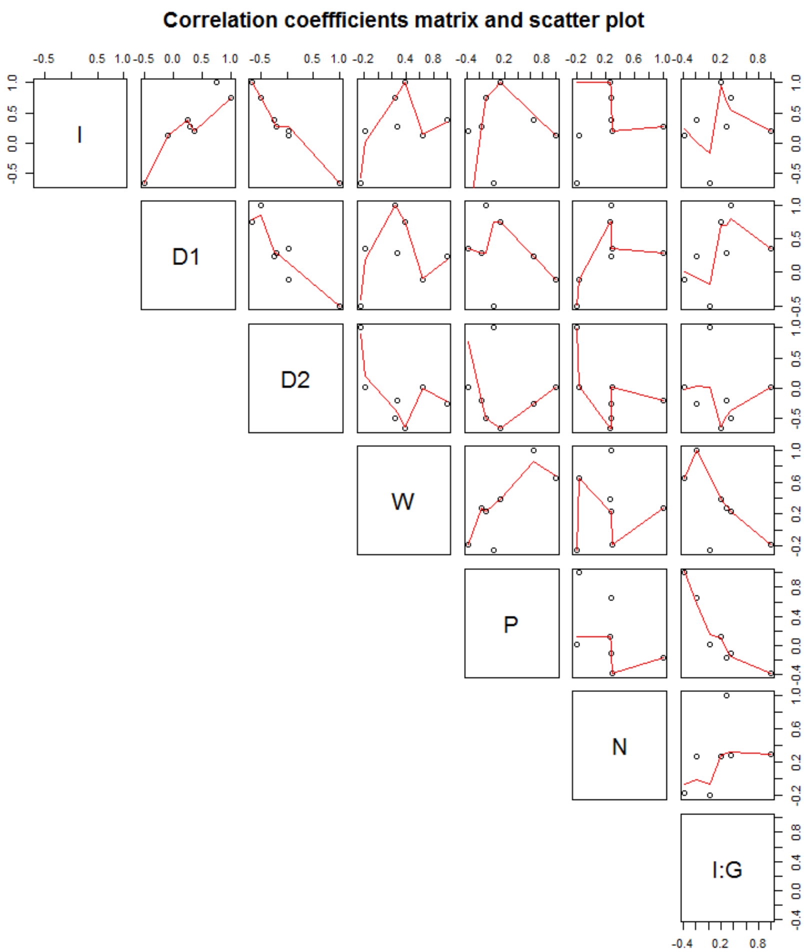



```
> # Correlation matrix shows strong evidence of collinearity among some
> # of the predicors (look at high magnitudes for correlation coefficient
> # in conjuction for a trend in the scatter plot)
> # For example, I has strong correlation with D, IN, IP, DP, DN>
> # Compute VIF
> library(car)
> vif(fit)
mydata$Year            I            D            W            G            P            N          I:D
   7.453621  1805.678696 1202.686309 4233.352772  230.237231  508.450542  115.188575     3.721595
        I:G          I:P          I:N          D:W          D:G          D:P          D:N          W:G
 237.292589   810.735449 1912.283560 4056.911272  228.421710  472.319766 1399.933140 1603.080713
        G:P          G:N          P:N
 677.137902    64.464644   497.538723
>
> # Determine VIF > 10
> names(vif(fit))[vif(fit)>10]
 [1] "I"   "D"   "W"   "G"   "P"   "N"   "I:G" "I:P" "I:N" "D:W" "D:G" "D:P" "D:N" "W:G" "G:P"
[16] "G:N" "P:N"
>
> # It appears that the predictor variables above are affected by the
> # presence of collinearity because VIF > 10. The only predictors
> # that don't exhibit multicollinearity problems are Year and I:D
> # Note that also the VIF is very large,  so there is a severe
> # multicollinearity problem.

> # Condition number
> sqrt(kappa(corr,exact=TRUE))
[1] 264.0279
> sqrt(max(eigen(corr)$values)/min(eigen(corr)$values))
[1] 264.0279
> # A large condition number indicates evidence of strong collinearity
> # In this case condition number > 15, so there are
> # harmful effects of collinearity in the data
```

Problem 3

```
> fit = lm(V~ I + D1 + D2 + W + G:I + P + N, data = mydata)
> summary(fit)

Call:
lm(formula = V ~ I + D1 + D2 + W + G:I + P + N, data = mydata)

Residuals:
      Min        1Q    Median        3Q       Max
-0.044201 -0.022728 -0.002548  0.011671  0.084681

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.5054760  0.0364190  13.879 3.58e-09 ***
I           -0.0205982  0.0174858  -1.178 0.259912
D1           0.0633485  0.0312177   2.029 0.063423 .
D2          -0.0469714  0.0291912  -1.609 0.131600
W            0.0123948  0.0436938   0.284 0.781127
P           -0.0006963  0.0041333  -0.168 0.868808
N           -0.0051083  0.0039349  -1.298 0.216773
I:G          0.0094222  0.0019580   4.812 0.000339 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04245 on 13 degrees of freedom
Multiple R-squared: 0.7922,	Adjusted R-squared: 0.6802
F-statistic: 7.078 on 7 and 13 DF,  p-value: 0.001307

>
> # Correlation matrix of predictor variables
> corr = cor(model.matrix(fit)[,-1]) # corr for all predictors (e.g. interaction)
> # corr = round(cor(mydata[-2]),2)  # corr for variables in dataset
> corrplot.mixed(corr, upper = "ellipse", lower = "number")
>
> pairs(corr, main = "Correlation coeffficients matrix and scatter plot",
+       pch = 21, lower.panel = NULL, panel = panel.smooth, cex.labels=2)
>
> # Correlation matrix shows some evidence (not strong) of collinearity among some
> # of the predictors (look at high magnitudes for correlation coefficient
> # in conjunction for a trend in the scatter plot)
> # For example, I has moderate correlation with D1 and D2, D1 with D2, and
> # W with P
>
> # Compute VIF
> library(car)
> vif(fit)
       I       D1       D2        W        P        N      I:G
3.555492 2.678628 2.026857 2.724643 2.612056 1.476388 1.535663
>
> # Determine VIF > 10
> names(vif(fit))[vif(fit)>10]
character(0)
>
> # It appears that the none of the predictor variables above are
> # affected by the presence of collinearity because VIF < 10.
> # Thus, there is no multicollinearity problem.
>
> # Condition number
> sqrt(max(eigen(corr)$values)/min(eigen(corr)$values))
[1] 4.030008
> sqrt(kappa(corr,exact=TRUE))
[1] 4.030008
> # A large condition number indicates evidence of strong collinearity
> # In this case condition number < 15, so there are no
> # harmful effects of collinearity in the data
```

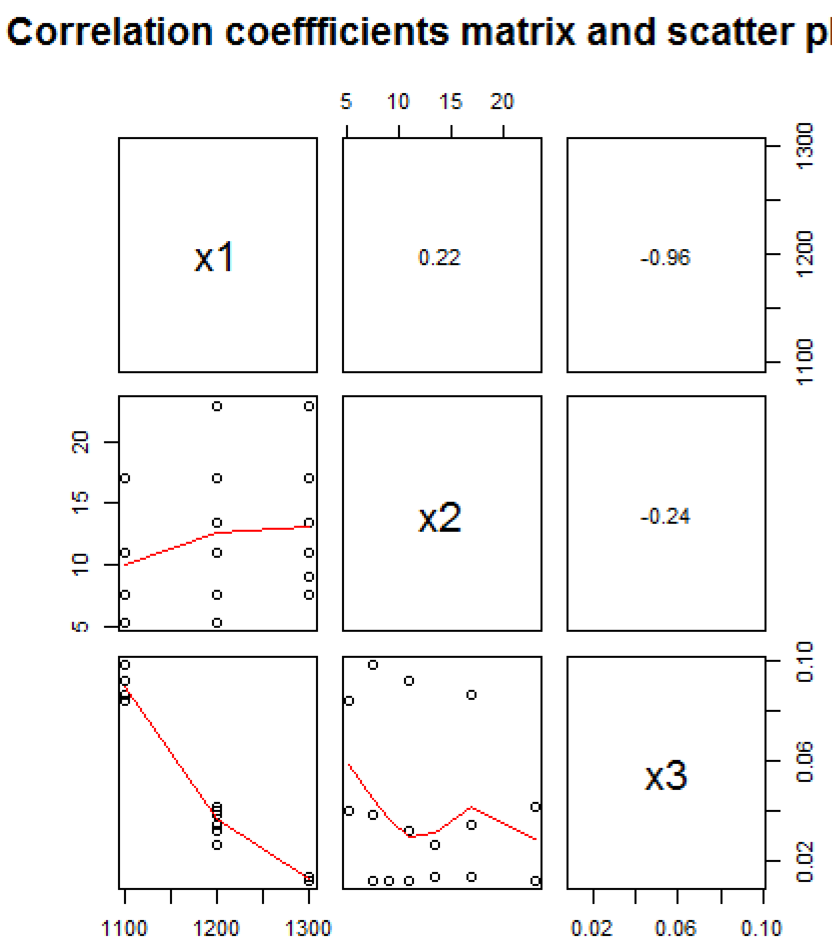

I
D1
D2
W
P
N
I:G
0.75
-0.66
-0.5
0.39
0.24
-0.26
0.12
-0.1
0.02
0.65
0.27
0.28
-0.21
0.27
-0.17
0.21
0.35
0.02
-0.19
-0.38
0.29


Correlation coeffficients matrix and scatter plot
I
D1
D2
W
P
N
I:G

Problem 4

Part a.

```
> panel.pearson <- function(x, y, ...) {
+   horizontal <- (par("usr")[1] + par("usr")[2]) / 2;
+   vertical <- (par("usr")[3] + par("usr")[4]) / 2;
+   text(horizontal, vertical, format(cor(x,y), digits=2))
+ }
>
> pairs(mydata[,1:3], main = "Correlation coefffficients matrix and scatter plot",
+         pch = 21, upper.panel=panel.pearson,lower.panel = panel.smooth)
```




```
> Note the problem is asking to plot only the three predictors variables, not
> all the terms of the second degree model (which might be more appropriate).

> # There is a sign of collinearity between x1 and x3,
> # where it looks like a linear relationship with a strong correlation
> # coefficient -0.96. However, as professor mentioned, large bivariate
> # correlations indicate multicollinearity but they don't capture multivariate
> # linear dependence relationships. So they are not reliable indicators of
> # multicollinearity. Further tests are needed (e.g. VIF).
```

Part b.

```
> fit = lm(y~ x1 + x2 + x3 + x1*x2 + x1*x3 + x2*x3 + I(x1^2) + I(x2^2) + I(x3^2),
mydata)
> summary(fit)

Call:
lm(formula = y ~ x1 + x2 + x3 + x1 * x2 + x1 * x3 + x2 * x3 +
    I(x1^2) + I(x2^2) + I(x3^2), data = mydata)

Residuals:
    Min      1Q  Median      3Q     Max
-1.3499 -0.3411  0.1297  0.5011  0.6720

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.617e+03  3.136e+03  -1.153  0.29260
x1           5.324e+00  4.879e+00   1.091  0.31706
x2           1.924e+01  4.303e+00   4.472  0.00423 **
x3           1.377e+04  1.045e+04   1.318  0.23572
I(x1^2)     -1.927e-03  1.896e-03  -1.016  0.34874
I(x2^2)     -3.034e-02  1.168e-02  -2.597  0.04084 *
I(x3^2)     -1.158e+04  7.699e+03  -1.504  0.18318
x1:x2       -1.414e-02  3.212e-03  -4.404  0.00455 **
x1:x3       -1.058e+01  8.241e+00  -1.283  0.24666
x2:x3       -2.103e+01  9.241e+00  -2.276  0.06312 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9014 on 6 degrees of freedom
Multiple R-squared: 0.9977,	Adjusted R-squared: 0.9943
F-statistic: 289.7 on 9 and 6 DF,  p-value: 3.225e-07

> library(car)
> vif(fit)
          x1           x2           x3      I(x1^2)      I(x2^2)      I(x3^2)        x1:x2
2.856749e+06 1.095614e+04 2.017163e+06 2.501945e+06 6.573359e+01 1.266710e+04 9.802903e+03
       x1:x3        x2:x3
1.428092e+06 2.403594e+02
>
> vif(fit)>10
     x1      x2      x3 I(x1^2) I(x2^2) I(x3^2)   x1:x2   x1:x3   x2:x3
   TRUE    TRUE    TRUE    TRUE    TRUE    TRUE    TRUE    TRUE    TRUE
>
> # The VIF > 10 for all terms in the model and are very large, so there is a
> # multicollinearity problem.
>
```

Part c.

```
> centered = apply(mydata[1:3],2,function(x) x-mean(x))
> centered = cbind(centered,mydata$y)
> colnames(centered)[4]="y"
> centered = data.frame(centered)
>
> fit = lm(y~ x1 + x2 + x3 + x1*x2 + x1*x3 + x2*x3 + I(x1^2) + I(x2^2) + I(x3^2),
centered)
> summary(fit)

Call:
lm(formula = y ~ x1 + x2 + x3 + x1 * x2 + x1 * x3 + x2 * x3 +
    I(x1^2) + I(x2^2) + I(x3^2), data = centered)

Residuals:
    Min      1Q  Median      3Q     Max
-1.3499 -0.3411  0.1297  0.5011  0.6720
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.590e+01  1.092e+00  32.884 5.26e-08 ***
x1           4.966e-02  5.592e-02   0.888 0.408719    
x2           4.907e-01  5.423e-02   9.048 0.000102 ***
x3          -2.542e+02  1.919e+02  -1.325 0.233461    
I(x1^2)     -1.927e-03  1.896e-03  -1.016 0.348741    
I(x2^2)     -3.034e-02  1.168e-02  -2.597 0.040844 *  
I(x3^2)     -1.158e+04  7.699e+03  -1.504 0.183182    
x1:x2       -1.414e-02  3.212e-03  -4.404 0.004547 ** 
x1:x3       -1.058e+01  8.241e+00  -1.283 0.246663    
x2:x3       -2.103e+01  9.241e+00  -2.276 0.063116 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9014 on 6 degrees of freedom
Multiple R-squared: 0.9977,	Adjusted R-squared: 0.9943 
F-statistic: 289.7 on 9 and 6 DF,  p-value: 3.225e-07

> library(car)
> vif(fit)
         x1          x2          x3     I(x1^2)     I(x2^2)     I(x3^2)       x1:x2       x1:x3 
 375.247759    1.740631  680.280039 1762.575365    3.164318 1156.766284   31.037059 6563.345193 
      x2:x3 
  35.611286 
> 
> vif(fit)>10
     x1      x2      x3 I(x1^2) I(x2^2) I(x3^2)   x1:x2   x1:x3   x2:x3 
   TRUE   FALSE    TRUE    TRUE   FALSE    TRUE    TRUE    TRUE    TRUE 
> 
> # The VIF > 10 for some terms shown above, so there is still multicollinearity
> # but it is less severe (all vif's are lower, and X2 and X2^2 don't exhibit
> # multicollinearity)
> 
```

Part d.

```
> # The assumption that the predictors are linearly independent is violated
> # because there is a multicollinearity problem. Thus, the model is not
> # valid since the assumption does not hold, and therefore the model cannot
> # give reliable results. For example, if two predictors are highly correlated
> # the estimated effects are not reliable because it is difficult to separate
> # apart their effects.
```

Problem 5

Part a.

Complete the table using the formulas:

- $p = \#\ of\ predictors\ in\ model$
- $Error\ df = n - (p + 1)$
- $MSE_p = SSE_p / error\ df$
- $R^2_{adj,p} = 1 - MSE_p / (SST/(n-1))$
  - $SST = SSR_p + SSE_p \rightarrow \sum(y-\bar{y})^2 = \sum(\hat{y}-\bar{y})^2 + \sum(y-\hat{y})^2$
  - For intercept only model, $\hat{y} = \bar{y} \rightarrow \sum(y-\bar{y})^2 = 0 + \sum(y-\bar{y})^2 \rightarrow SST = SSE_0$
- $C_p = SSE_p / MSE_{full} + 2(p+1) - n$

Variables in Model	SSEp	p	Error df	MSE p	Adj Rp ^2	Cp
None	950	0	19	50	0	20
x1	720	1	18	40	0.2	12.8
x2	630	1	18	35	0.3	9.2
x3	540	1	18	30	0.4	5.6
x1,x2	595	2	17	35	0.3	9.8
x1,x3	425	2	17	25	0.5	3
x2,x3	510	2	17	30	0.4	6.4
x1,x2,x3	400	3	16	25	0.5	4

n	20
SST	950
MSE (full)	25

Part b.

The adjusted R^2criteria says the best model is the one with the highest adjusted R^2, which in this case is (x1,x3) and (x1,x2,x3). The C_p criteria says the best model is the one with minimum C_p, which in this case is (x1,x3). So using both adjusted R^2 and C_p criteria, the best model is (x1,x3), which maximizes adjusted R^2 and minimizes C_p.

Part c.

Compute F-to-enter values:

- $F_1 = \frac{[SSE(None)-SSE(x_1)]/1}{SSE(x_1)/[n-(p+1)]} = \frac{[950-720]/1}{720/[20-(1+1)]} = 5.75$
- $F_2 = \frac{[SSE(None)-SSE(x_2)]/1}{SSE(x_2)/[n-(p+1)]} = \frac{[950-630]/1}{630/[20-(1+1)]} = 9.14$
- $F_3 = \frac{[SSE(None)-SSE(x_3)]/1}{SSE(x_3)/[n-(p+1)]} = \frac{[950-540]/1}{540/[20-(1+1)]} = 13.67$

Max $F_i = F_3 = 13.67 > f_{in} = 4.0$ → Enter x_3 into the equation

Part d.

Compute F-to-enter values:

- $F_1 = \frac{[SSE(x_3)-SSE(x_1,x_3)]/1}{SSE(x_1,x_3)/[n-(p+1)]} = \frac{[540-425]/1}{425/[20-(2+1)]} = 4.6$
- $F_2 = \frac{[SSE(x_3)-SSE(x_2,x_3)]/1}{SSE(x_2,x_3)/[n-(p+1)]} = \frac{[540-510]/1}{510/[20-(2+1)]} = 1$

Max $F_i = F_1 = 4.6 > f_{in} = 4.0$ → Enter x_1 into the equation (already including x_3) as the second variable

Compute partial regression coefficient with respect to y controlling for the first variable (x_3) that entered the model:

- ${r^2}_{yx_1|x_3} = \frac{SSE(x_3)-SSE(x_1,x_3)}{SSE(x_3)} = \frac{540-425}{540} = 0.213 \rightarrow r_{yx_1|x_3} = 0.461$

Part e.

Conduct a partial F-test to decide whether the first variable (x_3) that entered the model should be removed upon the entry of the second variable (x_1).

- $F_3 = \frac{[SSE(x_1)-SSE(x_1,x_3)]/1}{SSE(x_1,x_3)/[n-(p+1)]} = \frac{[720-425]/1}{425/[20-(2+1)]} = 11.8$
- $F_3 > f_{out} = 4.0$ so x_3 should not be removed from the model according to stepwise algorithm
- The F-test says $F_3 = 11.8 > f_{1,17,0.05} = 4.45$ so conclude x_3 is significant at level 0.05 (reject null that is equal to zero) and thus it should not be removed from the model upon entry of x_1.

Part f.

- $F_2 = \frac{[SSE(x_1,x_3)-SSE(x_1,x_3,x_2)]/1}{SSE(x_1,x_3,x_2)/[n-(p+1)]} = \frac{[425-400]/1}{400/[20-(3+1)]} = 1$
- $F_2 = 1 \leq f_{in} = 4.0$ so x_2 should not enter the model according to stepwise algorithm
- The F-test says $F_2 = 1 < f_{1,16,0.05} = 4.49$ so conclude x_2 is insignificant at level 0.05 (do not reject null that is equal to zero) and thus it should not enter the model given x_1 and x_3 are already in.

R-Code

```
 #### Homework 5
#1. Exercise 9.3 from text book (only do parts (a) and (d)).
#2. Exercise 9.4 from text book (only do parts (a) and (b)).
#3. Exercise 9.5 from text book (only do part (a)).

#### Setup ##################################################

# Install packages if needed
# install.packages("ggplot2")
# install.packages("grid")
# install.packages("gridExtra")
# install.packages("XLConnect")
# install.packages("corrplot")
# install.packages("Hmisc")
# install.packages("car")

# Load packages
library(ggplot2)
library(grid)
library(gridExtra)
library(XLConnect)
library(corrplot)
library(Hmisc)
library(car)
library(MASS)


# My PC
main = "C:/Users/Steven/Documents/Academics/3_Graduate School/2014-2015 ~ NU/"

# Aginity
#main = "\\\\nas1/labuser169"

course = "MSIA_401_Statistical Methods for Data Mining"
datafolder = "Data"
setwd(file.path(main,course, datafolder))

#### Problem 1 ##################################################
# Import data
filename = "P256.txt"
mydata = read.table(filename,header = T)

# Look at data
names(mydata)
head(mydata)
nrow(mydata)
```

```
summary(mydata)

# Fix names
names(mydata)[c(11,12)]=c("X_10","X_11")

#### Part a

# Plot separate

corr = round(cor(mydata[-1]),2)

corrplot(corr,method="number", type="upper")

pairs(mydata[,-1], main = "Correlation coeffficients matrix and scatter plot",
     pch = 21, lower.panel = NULL, panel = panel.smooth, cex.labels=2)

# Alternative
corrplot.mixed(corr, upper = "ellipse", lower = "number")


# Plot combine correlation coefficients matrix and scatter plot
# http://www2.warwick.ac.uk/fac/sci/moac/people/students/peter_cock/r/iris_plots/
panel.pearson <- function(x, y, ...) {
  horizontal <- (par("usr")[1] + par("usr")[2]) / 2;
  vertical <- (par("usr")[3] + par("usr")[4]) / 2;
  text(horizontal, vertical, format(cor(x,y), digits=2))
}

pairs(mydata[,2:length(mydata)], main = "Correlation coeffficients matrix and scatter plot",
     pch = 21, upper.panel=panel.pearson,lower.panel = panel.smooth)

# The pairwise correlation coefficeints of the predictor vairables
# and the corresponding scatter plots show strong linear relationships
# among pairs of predictors variables, suggesting collinearity.
# (look at high magnitudes for correlation coefficient
# in conjuction for a trend in the scatter plot)
# For example, x_1 has |correlations| of greater than |0.6| with all
# other predictors except x_4. Looking at the scatte plot, a clear
# linear relationship between x_1 and x_2, x_3, x_8, x_9 and x_10
# can be seen.
# Other notable correlations from the scatter plots are:
# x_2 with x_3, x_8, x_9 and x_10
# x_3 with x_8, x_9 and x_10
# x_8 with x_9 and x_10
# x_9 with x_10
# For x_8, x_9 and x_10, the correlations with the other
# predictors seems to follow a similar patter across
# x_8, x_9 and x_10 for each of the other predictors.
```

```
#### Part b

fit = lm(Y~.,mydata)
summary(fit)

# Compute VIF
library(car)
vif(fit)

# Determine VIF > 10
names(vif(fit))[vif(fit)>10]

# It appears that X1, X2, X3, X7, X8 and X10 are affected by the
# presence of collinearity because VIF > 10. Thus, there is
# a multicollinearity problem


#### Problem 2 ################################################
# Import data
filename = "P160.txt"
mydata = read.table(filename,header = T)
# Look at data
names(mydata)
head(mydata)
nrow(mydata)
summary(mydata)

#### part a

# Maximum number of terms (coefficients) in a linear regression
# is the number number of observations in the data.
# So for for this example, you can fit at most 21 terms (including
# intercept, so 20 predictors) since you need df :
# n-(p+1)>=0, n=21 -> p+1 =< 21 (or 20? need df at least 1)

#### part b

fit = lm(V~mydata$Year+.*.,mydata[-1])
#fit = lm(V~mydata$Year+.^3,mydata[-1])
fit = update(fit,.~.-I:W)
fit = update(fit,.~.-W:P)
fit = update(fit,.~.-W:N)
summary(fit)

# Correlation matrix of predictor variables
corr = cor(model.matrix(fit)[,-1]) # corr for all predictors (e.g. interaction)
# corr = round(cor(mydata[-2]),2)  # corr for variables in dataset
```

```
corrplot.mixed(corr, upper = "ellipse", lower = "number")

pairs(corr, main = "Correlation coeffficients matrix and scatter plot",
    pch = 21, lower.panel = NULL, panel = panel.smooth, cex.labels=2)

# Correlation matrix shows strong evidence of collinearity among some
# of the predicors (look at high magnitudes for correlation coefficient
# in conjuction for a trend in the scatter plot)
# For example, I has strong correlation with D, IN, IP, DP, DN

# Compute VIF
library(car)
vif(fit)

# Determine VIF > 10
names(vif(fit))[vif(fit)>10]

# It appears that the predicor variables above are affected by the
# presence of collinearity because VIF > 10. The only predictors
# that don't exhibit multicollinearity problems are Year and I:D
# Note that also the VIF is very large,  so there is a severe
# multicollinearity problem.

# Condition number
sqrt(kappa(corr,exact=TRUE))
sqrt(max(eigen(corr)$values)/min(eigen(corr)$values))

# A large condition number indicates evidence of strong collinearity
# In this case condition number > 15, so there are
# harmful effects of collinearity in the data


#### Problem 3 ###############################################
# Import data
filename = "P160.txt"
mydata = read.table(filename,header = T)
# Look at data
names(mydata)
head(mydata)
nrow(mydata)
summary(mydata)

# Dummy variable
mydata$D1 = (mydata$D==1)*1
mydata$D2 = (mydata$D==-1)*1

fit = lm(V~ I + D1 + D2 + W + G:I + P + N, data = mydata)
```

```
summary(fit)

# Correlation matrix of predictor variables
corr = cor(model.matrix(fit)[,-1]) # corr for all predictors (e.g. interaction)
# corr = round(cor(mydata[-2]),2)  # corr for variables in dataset

corrplot.mixed(corr, upper = "ellipse", lower = "number")

pairs(corr, main = "Correlation coeffficients matrix and scatter plot",
    pch = 21, lower.panel = NULL, panel = panel.smooth, cex.labels=2)

# Correlation matrix shows some evidence of collinearity among some
# of the predicors (look at high magnitudes for correlation coefficient
# in conjuction for a trend in the scatter plot)
# For example, I has moderate correlation with D1 and D2, D1 with D2, and
# W with P

# Compute VIF
library(car)
vif(fit)

# Determine VIF > 10
names(vif(fit))[vif(fit)>10]

# It appears that the none of the predicor variables above are
# affected by the presense of collinearity because VIF < 10.
# Thus, there is no multicollinearity problem.

# Condition number
sqrt(kappa(corr,exact=TRUE))
sqrt(max(eigen(corr)$values)/min(eigen(corr)$values))

#kappa(fit)

# A large condition number indicates evidence of strong collinearity
# In this case condition number < 15, so there are no
# harmful effects of collinearity in the data

#### Problem 4 ################################################
# Import data
filename = "acetylene.csv"
mydata = read.csv(filename,header = T)

# Look at data
names(mydata)
head(mydata)
nrow(mydata)
summary(mydata)
```

```
### Part a

corr = round(cor(mydata[1:3]),2)
corr

# Plot combine correlation coefficients matrix and scatter plot
# http://www2.warwick.ac.uk/fac/sci/moac/people/students/peter_cock/r/iris_plots/
panel.pearson <- function(x, y, ...) {
  horizontal <- (par("usr")[1] + par("usr")[2]) / 2;
  vertical <- (par("usr")[3] + par("usr")[4]) / 2;
  text(horizontal, vertical, format(cor(x,y), digits=2))
}

pairs(mydata[,1:3], main = "Correlation coeffficients matrix and scatter plot",
      pch = 21, upper.panel=panel.pearson,lower.panel = panel.smooth)

# Yes, there is a sign of multicollinearity between x1 and x3,
# where it looks like a linear relationship with a strong correlation
# coefficient -0.96.

### Part b

fit = lm(y~ x1 + x2 + x3 + x1*x2 + x1*x3 + x2*x3 + I(x1^2) + I(x2^2) + I(x3^2), mydata)
summary(fit)
library(car)
vif(fit)

vif(fit)>10

# The VIF > 10 for all terms in the model and are very large, so there is a
# multicollinearity problem.

### Part c

centered = apply(mydata[1:3],2,function(x) x-mean(x))
centered = cbind(centered,mydata$y)
colnames(centered)[4]="y"
centered = data.frame(centered)

fit = lm(y~ x1 + x2 + x3 + x1*x2 + x1*x3 + x2*x3 + I(x1^2) + I(x2^2) + I(x3^2), centered)
summary(fit)
library(car)
vif(fit)

vif(fit)>10

# The VIF > 10 for some terms, so there is still multicollinearity but it is
```

less severe (all vif's are lower, and X2 and X2^2 don't exhibit
multicollinearity)

Part d

The assumption that the predictors are linearly independent is violated
because there is a multicollinearity problem. Thus, the model is not
valid since the assumption does not hold, and therefore the model cannot
give reliable results.

exact = TRUE, sqrt (kappa)