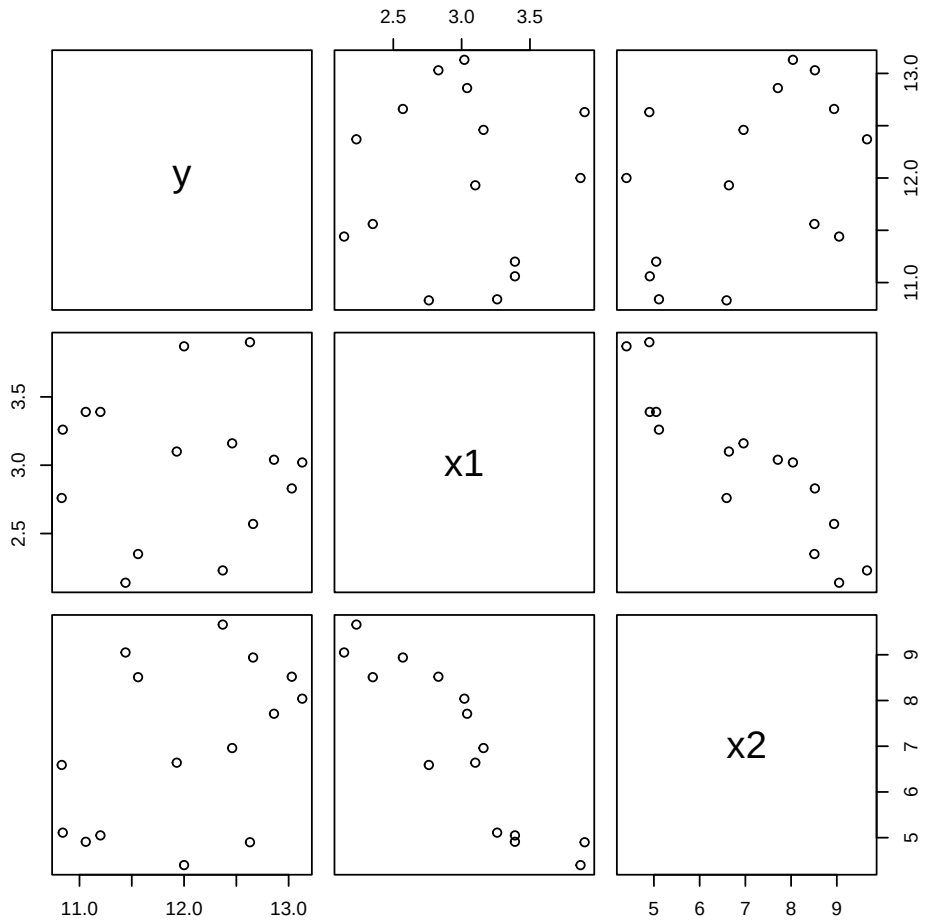
Multicollearity - section 3.10 and chapter 9

```
>   data=read.csv("http://chase.mathstat.dal.ca/~bsmith/stat3340/Data/
>   attach(data)
>   data[1:2,]

      y   x1   x2
1 12.37 2.23 9.66
2 12.66 2.57 8.94

>   pairs(data)
```



- There appears to be little relationship between Y and X_1 or Y and X_2.
- There does seen to be collinearity between X_1 and X_2.

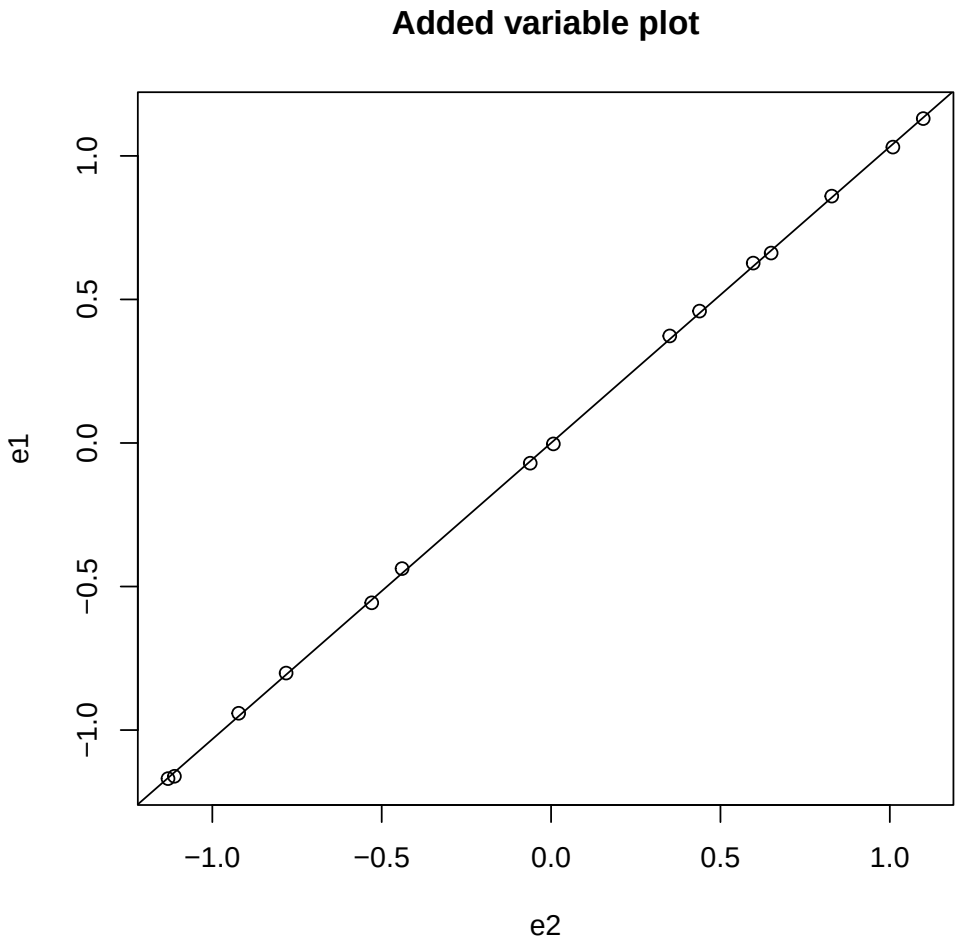
- but the fit of the linear model is perfect!

```
> lm.out=lm(y~x1+x2,data=data)
> summary(lm.out)

Call:
lm(formula = y ~ x1 + x2, data = data)

Residuals:
      Min        1Q    Median        3Q       Max
-0.013632 -0.009451 -0.002279  0.008630  0.016325

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -4.515414   0.061142  -73.85   <2e-16 ***
x1           3.097008   0.012274  252.31   <2e-16 ***
x2           1.031859   0.003684  280.08   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01072 on 12 degrees of freedom
Multiple R-squared:  0.9998,        Adjusted R-squared:  0.9998
F-statistic: 3.922e+04 on 2 and 12 DF,  p-value: < 2.2e-16

> anova(lm.out)

Analysis of Variance Table

Response: y
          Df Sum Sq Mean Sq    F value Pr(>F)
x1         1 0.0001  0.0001     0.4896 0.4975
x2         1 9.0072  9.0072 78444.1973 <2e-16 ***
Residuals 12 0.0014  0.0001
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

- This is a surprizing consequence of the strong correlation between x_1 and x_2.
- It means we can't completely trust scatterplots to reveal relationships among variables.

Let's look at the added variable plot for X_2.

```
> e1=residuals(lm(y~x1,data=data))
> e2=residuals(lm(x2~x1,data=data))
> plot(e2,e1,main="Added variable plot")
> abline(lm(e1~e2))
> summary(lm(e1~e2))

Call:
lm(formula = e1 ~ e2)

Residuals:
      Min        1Q    Median        3Q       Max
-0.013632 -0.009451 -0.002279  0.008630  0.016325

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 6.007e-17  2.658e-03     0.0        1
e2          1.032e+00  3.540e-03   291.5   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0103 on 13 degrees of freedom
Multiple R-squared:  0.9998,        Adjusted R-squared:  0.9998
F-statistic: 8.498e+04 on 1 and 13 DF,  p-value: < 2.2e-16
```


Added variable plot
e1
e2
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0

Multicollinearity between the predictor variables means that the columns of $\boldsymbol{X}$ are nearly linearly dependent.

In this case the matrix $\boldsymbol{X}^T\boldsymbol{X}$ is ill conditioned, in which case

- $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ will typically have large diagonal elements
- meaning that the standard errors of the $\hat{\beta}_j$ are large
- and as described on page 289-290, the least squares estimates of the β_j's will tend to be too large in absolute value.
- where W is like X, but having columns normalized to have unit variance, the diagonal elements of $(\boldsymbol{W}^T\boldsymbol{W})^{-1}$ are called **variance inflation factors**, also known as VIF's.
- Can show that the VIF for the j'th regesson coefficient can be written as

$$VIF_j = \frac{1}{1 - R_j^2}$$

  where R_j^2 is the coefficient of determination from regressing X_j on the other predictor variables.
- **Rule of thumb:** A VIF larger than 5 or 10 implies serious problems with multicollinearity.
- When there is multicollinearity the estimated β's are very sensitive to minor changes in the data, as are the predicted values of future y's.

What to do in the presence of multicolinearity?

- Try some new combinations of the predictors which might be closer to orthogonal. In the acetylene example, better to center the predictor variables.
- Ridge regression - replaces $\boldsymbol{X}^T\boldsymbol{X}$ by $\boldsymbol{X}^T\boldsymbol{X} + k\boldsymbol{I}$. Gives reduced variance to the resulting estimator, but generates a biased estimator.
- Principal components regression - uses new variables (principal components) which are transformed versions of the predictor variables, and are orthogonal. Principal components are a topic in Stat4350.
- Variable elimination. (In the acetylene example below , it may be better not to include both the linear and quadratic terms.)

The following codes a functon vif to calculate the variance inflation factors. Two examples are used, the soft drink delivery time data from Chapter 3, and the uncentered acetylene data from chapter 9.

```
> scale=function(x)(x-mean(x))/(sqrt(var(x)*(length(x)-1))) #scaling fu
> vif=function(lmout){  #function to calculation VIFs
+ X=model.matrix(lmout)
+ W=X[,-1] #remove the column of 1s
+ #scale the columns of W
+ for (j in (1:dim(W)[2])){
+    W[,j]=scale(W[,j])}
+    wpw=t(W)%*%W
+    print("correlataions of the predictor variables")
+    print(round(wpw,2))
+    vifs=diag(solve(t(W)%*%W))
+    print("variance inflation factors")
+    print(round(vifs,2))
+  return(NULL)}
```

Variance inflation factors for the delivery time data in chapter 3.

```
> data=read.csv("http://chase.mathstat.dal.ca/~bsmith/stat3340/Data/del
> data[1:2,]

      y x1  x2
1 16.68  7 560
2 11.50  3 220

> #fit a linear model and calculate VIFs
> vif(lm(y~x1+x2,data=data))

[1] "correlataions of the predictor variables"
     x1   x2
x1 1.00 0.82
x2 0.82 1.00
[1] "variance inflation factors"
  x1   x2
3.12 3.12
NULL
```

Variance inflation factors for the acetylene data in chapter 9.

```
> data=read.csv("http://chase.mathstat.dal.ca/~bsmith/stat3340/Data/ac
> data[1:2,]

     y   x1  x2    x3
1 49.0 1300 7.5 0.012
2 50.2 1300 9.0 0.012

> #fit a full quadratic model, and calulate VIFs
> vif(lm(y~x1+x2+x3+x1*x2+x1*x3+x2*x3+I(x1^2)+I(x2^2)+I(x3^2),data=dat

[1] "correlataions of the predictor variables"
           x1    x2    x3 I(x1^2) I(x2^2) I(x3^2) x1:x2 x1:x3 x2:x3
x1       1.00  0.22 -0.96    1.00    0.20   -0.89  0.35 -0.96 -0.76
x2       0.22  1.00 -0.24    0.22    0.98   -0.25  0.99 -0.24  0.33
x3      -0.96 -0.24  1.00   -0.95   -0.21    0.98 -0.35  1.00  0.76
I(x1^2)  1.00  0.22 -0.95    1.00    0.20   -0.88  0.34 -0.96 -0.75
I(x2^2)  0.20  0.98 -0.21    0.20    1.00   -0.23  0.97 -0.21  0.32
I(x3^2) -0.89 -0.25  0.98   -0.88   -0.23    1.00 -0.35  0.98  0.72
x1:x2    0.35  0.99 -0.35    0.34    0.97   -0.35  1.00 -0.35  0.20
x1:x3   -0.96 -0.24  1.00   -0.96   -0.21    0.98 -0.35  1.00  0.76
x2:x3   -0.76  0.33  0.76   -0.75    0.32    0.72  0.20  0.76  1.00
[1] "variance inflation factors"
        x1         x2         x3    I(x1^2)    I(x2^2)    I(x3^2)      x
2856748.97   10956.14 2017162.54 2501944.63      65.73   12667.10    980
     x1:x3      x2:x3
1428091.89     240.36
NULL
```

- When using polynomial terms, it is useful, and often essential, to subract the mean from the predictor variables, in order to reduce the correlation between linear, quadratic, and higher order terms.
- In the following we see that correlations and VIF's are reduced just by subtracting the means from x_1, x_2 and x_3.
- but the VIF's are still very large.

```
> data2=data
> data2$x1=data2$x1-mean(data2$x1)
> data2$x2=data2$x2-mean(data2$x2)
> data2$x3=data2$x3-mean(data2$x3)
> data2.lm=lm(y~x1+x2+x3+x1*x2+x1*x3+x2*x3+I(x1^2)+I(x2^2)+I(x3^2),da
> vif(data2.lm)

[1] "correlataions of the predictor variables"
           x1    x2    x3 I(x1^2) I(x2^2) I(x3^2) x1:x2 x1:x3 x2:x3
x1       1.00  0.22 -0.96   -0.27    0.03   -0.58 -0.13  0.44  0.21
x2       0.22  1.00 -0.24   -0.15    0.50   -0.22  0.04  0.19 -0.02
x3      -0.96 -0.24  1.00    0.50   -0.02    0.77  0.19 -0.66 -0.27
I(x1^2) -0.27 -0.15  0.50    1.00   -0.12    0.89  0.25 -0.97 -0.28
I(x2^2)  0.03  0.50 -0.02   -0.12    1.00   -0.16  0.40  0.13 -0.37
I(x3^2) -0.58 -0.22  0.77    0.89   -0.16    1.00  0.27 -0.97 -0.36
x1:x2   -0.13  0.04  0.19    0.25    0.40    0.27  1.00 -0.26 -0.97
x1:x3    0.44  0.19 -0.66   -0.97    0.13   -0.97 -0.26  1.00  0.32
x2:x3    0.21 -0.02 -0.27   -0.28   -0.37   -0.36 -0.97  0.32  1.00
[1] "variance inflation factors"
     x1      x2      x3 I(x1^2) I(x2^2) I(x3^2)   x1:x2   x1:x3   x2:x3
 375.25    1.74  680.28 1762.58    3.16 1156.77   31.04 6563.35   35.61
NULL
```

- It may be that the collinearity between x_1 and x_3 is what is generating the large VIFs for those variables.
- Try removing all terms involving x_3

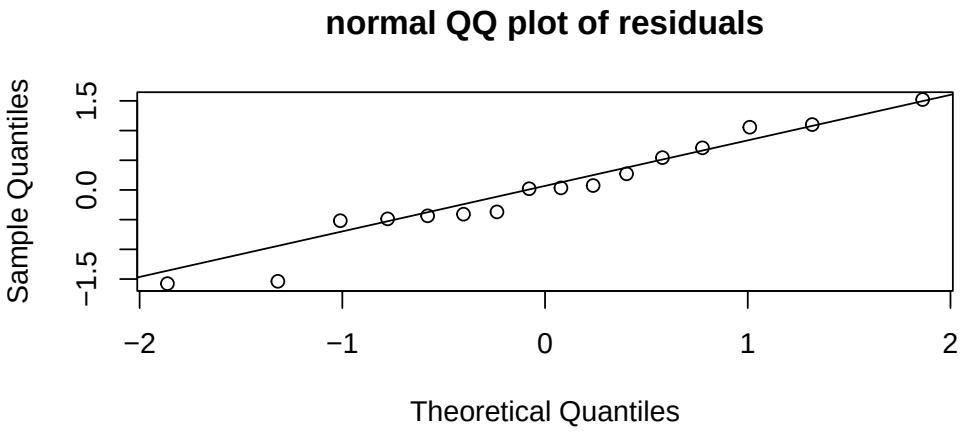
```
>  data2.lm2=lm(y~x1+x2+x1*x2+I(x1^2)+I(x2^2),data=data2)
>  vif(data2.lm2)

[1] "correlataions of the predictor variables"
           x1    x2 I(x1^2) I(x2^2) x1:x2
x1       1.00  0.22   -0.27    0.03 -0.13
x2       0.22  1.00   -0.15    0.50  0.04
I(x1^2) -0.27 -0.15    1.00   -0.12  0.25
I(x2^2)  0.03  0.50   -0.12    1.00  0.40
x1:x2   -0.13  0.04    0.25    0.40  1.00
[1] "variance inflation factors"
     x1      x2 I(x1^2) I(x2^2)   x1:x2
   1.14    1.45    1.20    1.72    1.37
NULL
```

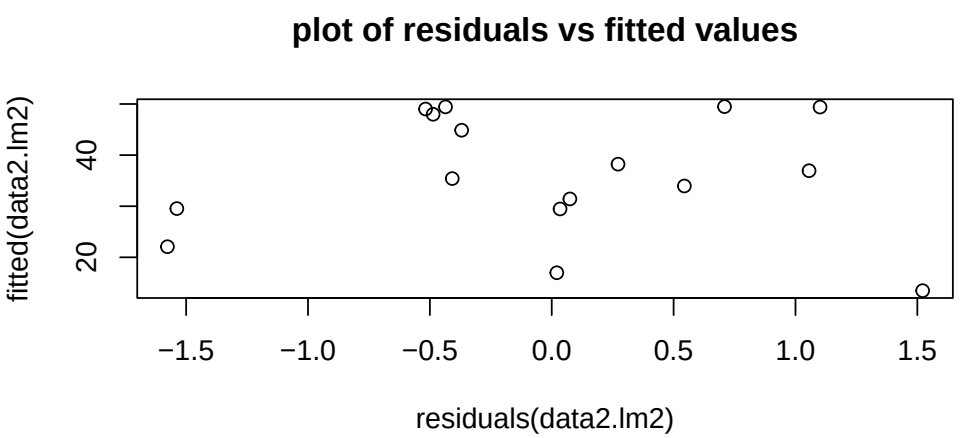
- Do the terms in x_3 improve the fit of the model?

```
> anova(data2.lm,data2.lm2)

Analysis of Variance Table


Model 1: y ~ x1 + x2 + x3 + x1 * x2 + x1 * x3 + x2 * x3 + I(x1^2) + I
    I(x3^2)
Model 2: y ~ x1 + x2 + x1 * x2 + I(x1^2) + I(x2^2)
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1      6  4.8756
2     10 11.3721 -4   -6.4966 1.9987 0.2139
```

- Does this reduced model look satisfactory?

```
> par(mfrow=c(2,1))
> qqnorm(residuals(data2.lm2),main="normal QQ plot of residuals")
> qqline(residuals(data2.lm2))
> plot(residuals(data2.lm2),fitted(data2.lm2),main="plot of residua
```


normal QQ plot of residuals
Sample Quantiles
−1.5
0.0
1.5
−2
−1
0
1
2
Theoretical Quantiles


plot of residuals vs fitted values
fitted(data2.lm2)
20
40
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
residuals(data2.lm2)