General Linear F Test & Multicollinearity Tutorial

For this tutorial, we will learn how conduct the general linear F test as well as to detect the presence of multicollinearity in MLR. We will continue to use the "nfl.txt" dataset. The data are on NFL team performance from the 1976 season. The variables are:

- y: Games won (out of 14 games)
- x_1: Rushing yards (season)
- x_2: Passing yards (season)
- x_3: Punting average (yards/punt)
- x_4: Field goal percentage (FGs made/FGs attempted)
- x_5: Turnover differential (turnovers acquired minus turnovers lost)
- x_6: Penalty yards (season)
- x_7: Percent rushing (rushing plays/total plays)
- x_8: Opponents' rushing yards (season)
- x_9: Opponents' passing yards (season)

We want to assess how the number of games won may be predicted and related to these predictors.

Download the data file and read the data in.

```
Data<-read.table("nfl.txt", header=TRUE)
```

There are a number of strategies on how to start building a multiple linear regression (MLR) model. One possible strategy is to build an initial model based on what appear to be predictors that are most related to the number of wins. Let us create a correlation matrix of the variables:

```
round(cor(Data),3)
```

```
##          y     x1     x2     x3     x4     x5     x6     x7     x8     x9
## y   1.000  0.593  0.483 -0.081  0.258  0.513  0.224  0.545 -0.738 -0.304
## x1  0.593  1.000 -0.037  0.212  0.070  0.600  0.253  0.837 -0.659 -0.111
## x2  0.483 -0.037  1.000 -0.069  0.302  0.135 -0.193 -0.197 -0.051  0.146
## x3 -0.081  0.212 -0.069  1.000 -0.413  0.115 -0.003  0.163  0.290  0.088
## x4  0.258  0.070  0.302 -0.413  1.000  0.149 -0.128 -0.101 -0.164  0.059
## x5  0.513  0.600  0.135  0.115  0.149  1.000  0.259  0.610 -0.470 -0.090
## x6  0.224  0.253 -0.193 -0.003 -0.128  0.259  1.000  0.367 -0.352 -0.173
## x7  0.545  0.837 -0.197  0.163 -0.101  0.610  0.367  1.000 -0.685 -0.203
## x8 -0.738 -0.659 -0.051  0.290 -0.164 -0.470 -0.352 -0.685  1.000  0.417
## x9 -0.304 -0.111  0.146  0.088  0.059 -0.090 -0.173 -0.203  0.417  1.000
```

We use the `round()` function so we can limit the number of decimal places the output uses, which in this case is three.

1 General Linear F Test

It appears from the correlation matrix that x_1, x_5, x_7, x_8 have strong linear associations with the number of wins. So we start with these four predictors for our MLR:

```
##fit MLR
result<-lm(y~x1+x5+x7+x8, data=Data)
summary(result)
```

```
##
## Call:
## lm(formula = y ~ x1 + x5 + x7 + x8, data = Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4923 -1.7750  0.0165  1.4748  5.1252
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 22.787348  10.240559   2.225  0.03616 *
## x1           0.001951   0.002286   0.853  0.40226
## x5           0.068083   0.057204   1.190  0.24612
## x7          -0.124625   0.169953  -0.733  0.47079
## x8          -0.006015   0.001779  -3.382  0.00257 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.402 on 23 degrees of freedom
## Multiple R-squared:  0.5941, Adjusted R-squared:  0.5235
## F-statistic: 8.417 on 4 and 23 DF,  p-value: 0.0002456
```

Based on the t tests, we consider dropping x_1, x_5, x_7 from the model. So we perform a general linear F test with the full model using predictors x_1, x_5, x_7, x_8 and the reduced model only using x_8. The null and alternative hypotheses are:

$H_0 : \beta_1 = \beta_5 = \beta_7 = 0,$

H_a : at least one of the coefficients in H_0 is not 0.

In words, the null hypothesis supports going with the reduced model by dropping x_1, x_5, x_7, whereas the alternative hypothesis supports the full model by not dropping x_1, x_5, x_7.

We explore two approaches to conducting this general linear F test.

1.1 Directly comparing the full and reduced models

In this approach, we fit the reduced model, and then use the `anova()` function to compare the reduced model with the full model:

```r
reduced<-lm(y~x8, data=Data)

##general linear F test to compare reduced model with full model
anova(reduced, result)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ x8
## Model 2: y ~ x1 + x5 + x7 + x8
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     26 148.87
## 2     23 132.70  3    16.169 0.9341 0.4402
```

The F statistic from this test is 0.9341, with a p-value of 0.4402. So we fail to reject the null hypothesis, so there is little evidence of supporting the full model. We go with the reduced model over the full model.

1.2 Sequential sums of squares

In this other approach, we use the `anova()` function on the full model to obtain the **sequential sums of squares** associated with the full model:

```
anova(result) ##output doesn't give us needed info
```

```
## Analysis of Variance Table
##
## Response: y
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## x1         1 115.068 115.068 19.9435 0.0001763 ***
## x5         1  12.637  12.637  2.1902 0.1524627
## x7         1   0.578   0.578  0.1002 0.7544524
## x8         1  65.978  65.978 11.4352 0.0025706 **
## Residuals 23 132.703   5.770
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The values under the column "Sum Sq" give the sequential SS_Rs. Notice how the information is provided: in the order in which the predictors were entered into `lm()`. Recall our question is whether we can drop x_1, x_5, x_7 and leave x_8 in. So we need $SS_R(x_1, x_5, x_7|x_8)$ but this output does not give us the needed info.

We need to rearrange the order in which the predictors are entered into `lm()`:

```
##rearrange. put predictors to drop last in lm()
full<-lm(y~x8+x1+x5+x7, data=Data)

anova(full)
```

```
## Analysis of Variance Table
##
## Response: y
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## x8         1 178.092 178.092 30.8668 1.188e-05 ***
## x1         1   6.636   6.636  1.1502    0.2946
## x5         1   6.430   6.430  1.1144    0.3021
## x7         1   3.102   3.102  0.5377    0.4708
## Residuals 23 132.703   5.770
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The partial F statistic is

$$\begin{aligned}
F &= \frac{[SS_R(F) - SS_R(R)]/r}{SS_{res}(F)/(n-p)} \\
&= \frac{[SS_R(x_1, x_5, x_7, x_8) - SS_R(x_8)]/3}{SS_{res}(x_1, x_5, x_7, x_8)/(28-5)} \\
&= \frac{SS_R(x_1, x_5, x_7|x_8)/3}{SS_{res}(x_1, x_5, x_7, x_8)/(28-5)} \\
&= \frac{(6.636 + 6.430 + 3.102)/3}{132.703/23} \\
&= 0.9340758
\end{aligned} \tag{1}$$

which is similar to the value found in approach 1 (discrepancy due to rounding off in intermediate steps).

The corresponding p-value is

```
1-pf(0.9340758,3,23)
```

```
## [1] 0.4402025
```

and the critical value is

```
qf(0.95,3,23)
```

```
## [1] 3.027998
```

So we fail to reject the null hypothesis and go with the reduced model.

2 Multicollinearity

With the presence of multiple predictors, it is often tempting to start by including all the predictors in the model.

```
##fit MLR with all predictors
all<-lm(y~., data=Data)
```

There are a few ways to detect the presence of multicollinearity in our model.

2.1 t tests and ANOVA F test

The presence of a lot of insignificant t tests for the regression coefficients, along a highly significant ANOVA F test is an indication that multicollinearity is present:

```
##look at t tests, and F test
summary(all)
```

```
##
## Call:
## lm(formula = y ~ ., data = Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0408 -0.6802 -0.1131  0.9835  2.9785
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.292e+00  1.281e+01  -0.569 0.576312
## x1           8.124e-04  2.006e-03   0.405 0.690329
## x2           3.631e-03  8.410e-04   4.318 0.000414 ***
## x3           1.222e-01  2.590e-01   0.472 0.642750
## x4           3.189e-02  4.160e-02   0.767 0.453289
## x5           1.511e-05  4.684e-02   0.000 0.999746
## x6           1.590e-03  3.248e-03   0.490 0.630338
## x7           1.544e-01  1.521e-01   1.015 0.323547
## x8          -3.895e-03  2.052e-03  -1.898 0.073793 .
## x9          -1.791e-03  1.417e-03  -1.264 0.222490
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.83 on 18 degrees of freedom
## Multiple R-squared:  0.8156, Adjusted R-squared:  0.7234
## F-statistic: 8.846 on 9 and 18 DF,  p-value: 5.303e-05
```

Notice how almost all the t tests are insignificant, but the ANOVA F is highly significant. So we have evidence of multicollinearity.

2.2 Standard errors of estimated coefficients

Looking at the output from `summary()`, we do not see any standard errors that are large. If we have strong multicollinearity, standard errors should be large.

2.3 Correlation between pairs of predictors

We can also look at the pairwise correlations among predictors:

```
##correlation matrix, round to 3 decimal
round(cor(Data[,-1]),3)
```

```
##        x1     x2     x3     x4     x5     x6     x7     x8     x9
## x1  1.000 -0.037  0.212  0.070  0.600  0.253  0.837 -0.659 -0.111
## x2 -0.037  1.000 -0.069  0.302  0.135 -0.193 -0.197 -0.051  0.146
## x3  0.212 -0.069  1.000 -0.413  0.115 -0.003  0.163  0.290  0.088
## x4  0.070  0.302 -0.413  1.000  0.149 -0.128 -0.101 -0.164  0.059
## x5  0.600  0.135  0.115  0.149  1.000  0.259  0.610 -0.470 -0.090
## x6  0.253 -0.193 -0.003 -0.128  0.259  1.000  0.367 -0.352 -0.173
## x7  0.837 -0.197  0.163 -0.101  0.610  0.367  1.000 -0.685 -0.203
## x8 -0.659 -0.051  0.290 -0.164 -0.470 -0.352 -0.685  1.000  0.417
## x9 -0.111  0.146  0.088  0.059 -0.090 -0.173 -0.203  0.417  1.000
```

Looking at this matrix, we notice that pairs of predictors involving x_1, x_5, x_7, x_8 have high correlations. For pairs involving other predictors, the correlations are a lot weaker. So there is some degree of multicollinearity.

2.4 VIFs

High VIFs are an indication of multicollinearity.

```
##VIFs
library(faraway)
faraway::vif(all)
```

```
##       x1       x2       x3       x4       x5       x6       x7       x8
## 4.827645 1.420161 2.126597 1.566107 1.924035 1.275979 5.414572 4.535643
##       x9
## 1.423390
```

The largest VIF belongs to β_7, which is 5.414572. VIFs above 5 indicate a moderate degree of multicollinearity, while VIFs above 10 indicate a strong degree of multicollinearity.

To summarize what we have seen:

- The ANOVA F test is significant, but a lot of the t tests are insignificant.
- We don't see huge standard errors for the estimated coefficients.
- 4 of the predictors have high pairwise correlations, x_1, x_5, x_7, x_8.
- The largest VIF is 5.414572.

Collectively, there is some degree of multicollinearity in this model.

2.5 Next steps

We have identified that predictors x_1, x_5, x_7, x_8 are the ones that are most likely to be causing multicollinearity. A solution will be to use a subset of these predictors and not all of them.

Using subject matter knowledge can help with this decision.