BigDataAnalyticsAssignment1

Parth Patel

February 22, 2016

1 : What is your independent variable, what are your dependent variables given this analysis goal?

Solution : Since we are predicting the mpg, our dependent Variable is mpg.

Everything else (as follows) becomes independent Variables : 1. cylinders
2. displacement
3. horsepower
4. weight
5. acceleration
6. model year
7. origin
8. car name

2 : Describe the data by reporting means and standard deviation of each variable; plot pairs of variables (in a plot matrix) and report observations from the plot.

Solution :

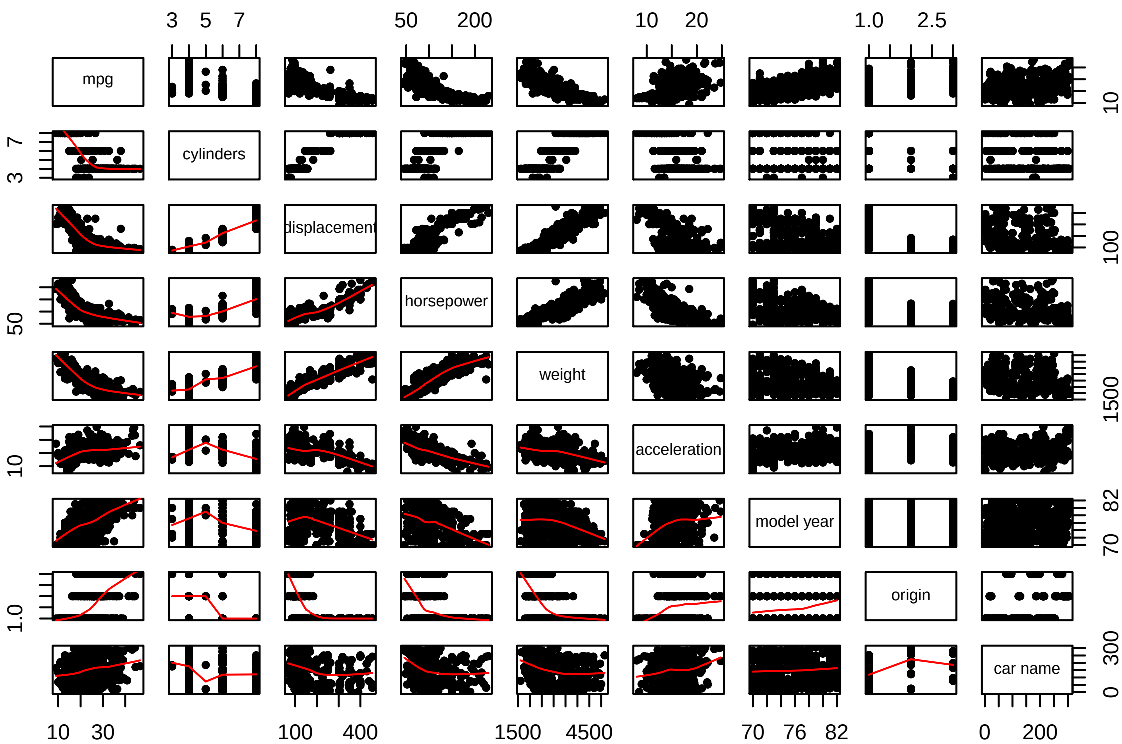
```
mpg_data<-read.table("/Users/Parth/Desktop/Z604BigDataLab/auto-mpg/auto-mpg.data",header = FALSE,na.stri
colnames(mpg_data)<-c( "mpg","cylinders","displacement" ,"horsepower","weight",
                  "acceleration","model year","origin","car name")
mpg_data = na.omit(mpg_data)
summary(mpg_data)
```

```
##       mpg          cylinders      displacement     horsepower
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0
##
##      weight      acceleration     model year        origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000
##  Median :2804   Median :15.50   Median :76.00   Median :1.000
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
##
##               car name
##  amc matador       :  5
##  ford pinto        :  5
##  toyota corolla    :  5
##  amc gremlin       :  4
##  amc hornet        :  4
##  chevrolet chevette:  4
##  (Other)           :365
```

```r
sapply(mpg_data[-9], sd)
```

```
##          mpg    cylinders displacement   horsepower       weight
##    7.8050075    1.7057832  104.6440039   38.4911599  849.4025600
## acceleration   model year       origin
##    2.7588641    3.6837365    0.8055182
```

```r
pairs(mpg_data,lower.panel = panel.smooth,pch = 20)
```



From the above plot we can infer that the following pairs of attributes show linear corelation :

1. Displacement ~ Horsepower. (positive corelation)
2. Horsepower ~ Weight. (positive corelation)
3. Acceleration ~ Horsepower (weak negative corealtion).
4. Horsepower ~ mpg (weak negative corealtion)
5. Weight ~ mpg (negative corealtion)
6. Weight ~ Displacement (positive corelation)

Thus, pairs plot appear to be good for determining rough linear correlations between continuous variables.But not the same for looking at discrete variables.

3 : **a :** *

Build a linear regression model, and report its summary.

Solution :

We can use the trial and error method to try out the different combinations of attributes and generate a model for each one. Later we can use anova to find out the best of the lot.

```r
model1<-lm(mpg~factor(mpg_data$cylinders),data = mpg_data)
model2<-lm(mpg~factor(mpg_data$cylinders)+ weight,data = mpg_data)
model3<-lm(mpg~factor(mpg_data$cylinders) + weight + horsepower,data = mpg_data)
model4<-lm(mpg~factor(mpg_data$cylinders) + weight + horsepower +acceleration,data = mpg_data)
model5<-lm(mpg~factor(mpg_data$cylinders) + weight + horsepower+factor(origin),data = mpg_data)
model6<-lm(mpg~factor(mpg_data$cylinders) + weight + horsepower +acceleration +displacement,data = mpg_
model7 <-lm(mpg~factor(mpg_data$cylinders) + weight + horsepower + acceleration +displacement+`model yea
model8 <-lm(mpg~factor(mpg_data$cylinders) + weight + horsepower + displacement,data = mpg_data)
model9<-lm(mpg~displacement + weight + horsepower,data = mpg_data)

model10 <- lm(mpg~factor(mpg_data$cylinders) + weight + horsepower+factor(mpg_data$`model year`),data =
anova(model1,model2,model3,model4,model5,model6,model7,model8,model9,model10)
```

```
## Analysis of Variance Table
##
## Model  1: mpg ~ factor(mpg_data$cylinders)
## Model  2: mpg ~ factor(mpg_data$cylinders) + weight
## Model  3: mpg ~ factor(mpg_data$cylinders) + weight + horsepower
## Model  4: mpg ~ factor(mpg_data$cylinders) + weight + horsepower + acceleration
## Model  5: mpg ~ factor(mpg_data$cylinders) + weight + horsepower + factor(origin)
## Model  6: mpg ~ factor(mpg_data$cylinders) + weight + horsepower + acceleration +
##     displacement
## Model  7: mpg ~ factor(mpg_data$cylinders) + weight + horsepower + acceleration +
##     displacement + `model year` + origin + `car name`
## Model  8: mpg ~ factor(mpg_data$cylinders) + weight + horsepower + displacement
## Model  9: mpg ~ displacement + weight + horsepower
## Model 10: mpg ~ factor(mpg_data$cylinders) + weight + horsepower + factor(mpg_data$`model year`)
##    Res.Df    RSS   Df Sum of Sq        F    Pr(>F)
## 1     387 8544.5
## 2     386 6552.5    1    1992.0 387.2046 < 2.2e-16 ***
## 3     385 6143.4    1     409.1  79.5237 8.694e-14 ***
## 4     384 6135.9    1       7.5   1.4624    0.2299
## 5     383 5842.1    1     293.8  57.1026 4.646e-11 ***
## 6     383 6135.7    0    -293.6
## 7      84  432.1  299    5703.6   3.7079 3.240e-11 ***
## 8     384 6143.4 -300   -5711.2   3.7005 3.406e-11 ***
## 9     388 6980.0   -4    -836.6  40.6568 < 2.2e-16 ***
## 10    373 3175.7   15    3804.3  49.2990 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

*** b : *** And what does the hypothesis testing (i.e. t-test results) tell you about the linear model coefficients?

*** Solution : ****

From the anova result we can see that the ***model10***, ***model2*** *and* **model9*** can gives us one of the best model since the t-test p-values for these models are the least, which shows that these models are the most significant. We can use the summary() function on each model to further check the R square value and compare all three to the base model statistics.

```r
base.model=lm(mpg ~ 1,data=mpg_data)
#Summary of base model
summary(base.model)
```

```
##
## Call:
## lm(formula = mpg ~ 1, data = mpg_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.4459  -6.4459  -0.6959   5.5541  23.1541
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  23.4459     0.3942   59.48   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.805 on 391 degrees of freedom
```

```r
m_base.forward <- step(base.model, scope=~factor(cylinders) + weight + horsepower +acceleration +displac
```

```
## Start:  AIC=1611.93
## mpg ~ 1
##
##                        Df Sum of Sq     RSS    AIC
## + weight                1   16497.8  7321.2 1151.5
## + displacement          1   15440.2  8378.8 1204.4
## + factor(cylinders)     4   15274.5  8544.5 1218.1
## + horsepower            1   14433.1  9385.9 1248.9
## + factor(`model year`) 12   10236.3 13582.7 1415.8
## + origin                1    7609.2 16209.8 1463.1
## + acceleration          1    4268.5 19550.5 1536.5
## <none>                              23819.0 1611.9
##
## Step:  AIC=1151.49
## mpg ~ weight
##
##                        Df Sum of Sq    RSS     AIC
## + factor(`model year`) 12    3558.5 3762.8  914.56
## + factor(cylinders)     4     768.7 6552.5 1116.01
## + horsepower            1     327.4 6993.8 1135.56
## + origin                1     222.2 7099.0 1141.41
## + acceleration          1     168.3 7152.9 1144.37
## + displacement          1     150.9 7170.3 1145.33
## <none>                              7321.2 1151.49
##
## Step:  AIC=914.56
## mpg ~ weight + factor(`model year`)
##
##                     Df Sum of Sq    RSS    AIC
## + factor(cylinders)  4    517.65 3245.1 864.55
## + origin             1    158.63 3604.1 899.68
## <none>                           3762.8 914.56
## + acceleration       1     16.84 3745.9 914.81
## + horsepower         1     15.71 3747.0 914.92
## + displacement       1      0.76 3762.0 916.49
##
```

```
## Step:  AIC=864.55
## mpg ~ weight + factor(`model year`) + factor(cylinders)
##
##                  Df Sum of Sq    RSS    AIC
## + origin          1   127.208 3117.9 850.87
## + horsepower      1    69.381 3175.7 858.08
## + acceleration    1    32.430 3212.7 862.61
## <none>                        3245.1 864.55
## + displacement    1     5.896 3239.2 865.83
##
## Step:  AIC=850.87
## mpg ~ weight + factor(`model year`) + factor(cylinders) + origin
##
##                  Df Sum of Sq    RSS    AIC
## + horsepower      1    95.075 3022.8 840.73
## + acceleration    1    35.354 3082.5 848.40
## <none>                        3117.9 850.87
## + displacement    1     0.063 3117.8 852.86
##
## Step:  AIC=840.73
## mpg ~ weight + factor(`model year`) + factor(cylinders) + origin +
##     horsepower
##
##                  Df Sum of Sq    RSS    AIC
## + displacement    1   18.6797 3004.1 840.30
## <none>                        3022.8 840.73
## + acceleration    1    0.1973 3022.6 842.71
##
## Step:  AIC=840.3
## mpg ~ weight + factor(`model year`) + factor(cylinders) + origin +
##     horsepower + displacement
##
##                  Df  Sum of Sq    RSS   AIC
## <none>                          3004.1 840.3
## + acceleration    1 0.00091246 3004.1 842.3
```

```r
#Summary of forward base model
summary(m_base.forward)
```

```
##
## Call:
## lm(formula = mpg ~ weight + factor(`model year`) + factor(cylinders) +
##     origin + horsepower + displacement, data = mpg_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.5269 -1.7124 -0.0611  1.4069 12.0049
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              29.756277   2.050630  14.511  < 2e-16 ***
## weight                   -0.005052   0.000536  -9.426  < 2e-16 ***
## factor(`model year`)71    0.824821   0.804472   1.025 0.305892
## factor(`model year`)72   -0.583990   0.798521  -0.731 0.465033
```

```
## factor(`model year`)73 -0.589425   0.718502  -0.820 0.412542
## factor(`model year`)74  1.138720   0.843731   1.350 0.177960
## factor(`model year`)75  0.789374   0.825832   0.956 0.339769
## factor(`model year`)76  1.426182   0.792917   1.799 0.072886 .
## factor(`model year`)77  2.876218   0.808342   3.558 0.000422 ***
## factor(`model year`)78  2.846777   0.767539   3.709 0.000240 ***
## factor(`model year`)79  4.773889   0.813082   5.871 9.60e-09 ***
## factor(`model year`)80  8.930309   0.864098  10.335  < 2e-16 ***
## factor(`model year`)81  6.266911   0.839388   7.466 5.95e-13 ***
## factor(`model year`)82  7.606291   0.822669   9.246  < 2e-16 ***
## factor(cylinders)4      7.224348   1.503146   4.806 2.24e-06 ***
## factor(cylinders)5      7.274167   2.268089   3.207 0.001457 **
## factor(cylinders)6      4.499643   1.686855   2.667 0.007977 **
## factor(cylinders)8      6.617667   1.952581   3.389 0.000776 ***
## origin                  1.140958   0.248004   4.601 5.79e-06 ***
## horsepower             -0.039217   0.010466  -3.747 0.000207 ***
## displacement            0.009961   0.006558   1.519 0.129655
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.846 on 371 degrees of freedom
## Multiple R-squared:  0.8739, Adjusted R-squared:  0.8671
## F-statistic: 128.5 on 20 and 371 DF,  p-value: < 2.2e-16
```

```r
#Summary of  model2
summary(model2)
```

```
##
## Call:
## lm(formula = mpg ~ factor(mpg_data$cylinders) + weight, data = mpg_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.2540  -2.5350  -0.2333   1.9110  16.8831
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 35.2062017  2.4646251  14.285  < 2e-16 ***
## factor(mpg_data$cylinders)4  8.1632569  2.0813281   3.922 0.000104 ***
## factor(mpg_data$cylinders)5 11.1236000  3.1718117   3.507 0.000506 ***
## factor(mpg_data$cylinders)6  4.3340730  2.1572818   2.009 0.045228 *
## factor(mpg_data$cylinders)8  4.9001794  2.3121157   2.119 0.034699 *
## weight                      -0.0061106  0.0005641 -10.833  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.12 on 386 degrees of freedom
## Multiple R-squared:  0.7249, Adjusted R-squared:  0.7213
## F-statistic: 203.4 on 5 and 386 DF,  p-value: < 2.2e-16
```

```r
#Summary of  model9
summary(model9)
```

```
##
```

```
## Call:
## lm(formula = mpg ~ displacement + weight + horsepower, data = mpg_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3347  -2.8028  -0.3402   2.2037  16.2409
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  44.8559357  1.1959200  37.507  < 2e-16 ***
## displacement -0.0057688  0.0065819  -0.876  0.38132
## weight       -0.0053516  0.0007124  -7.513 4.04e-13 ***
## horsepower   -0.0416741  0.0128139  -3.252  0.00125 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.241 on 388 degrees of freedom
## Multiple R-squared:  0.707,  Adjusted R-squared:  0.7047
## F-statistic:   312 on 3 and 388 DF,  p-value: < 2.2e-16
```

```r
#Summary of  model10
summary(model10)
```

```
##
## Call:
## lm(formula = mpg ~ factor(mpg_data$cylinders) + weight + horsepower +
##     factor(mpg_data$`model year`), data = mpg_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.0240 -1.7624 -0.0319  1.6177 12.1261
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     33.2434734  1.9030906  17.468  < 2e-16 ***
## factor(mpg_data$cylinders)4      6.7116726  1.5001196   4.474 1.02e-05 ***
## factor(mpg_data$cylinders)5      7.2594684  2.2980568   3.159 0.001712 **
## factor(mpg_data$cylinders)6      4.1286314  1.5680799   2.633 0.008817 **
## factor(mpg_data$cylinders)8      6.9381993  1.6746196   4.143 4.24e-05 ***
## weight                          -0.0053065  0.0004865 -10.908  < 2e-16 ***
## horsepower                      -0.0280872  0.0098391  -2.855 0.004549 **
## factor(mpg_data$`model year`)71  0.9508122  0.8243561   1.153 0.249485
## factor(mpg_data$`model year`)72 -0.5498986  0.8141175  -0.675 0.499806
## factor(mpg_data$`model year`)73 -0.4762528  0.7356537  -0.647 0.517780
## factor(mpg_data$`model year`)74  1.2856174  0.8579690   1.498 0.134864
## factor(mpg_data$`model year`)75  0.9958222  0.8438772   1.180 0.238730
## factor(mpg_data$`model year`)76  1.4955797  0.8083427   1.850 0.065078 .
## factor(mpg_data$`model year`)77  2.9127501  0.8249641   3.531 0.000466 ***
## factor(mpg_data$`model year`)78  2.9003361  0.7810016   3.714 0.000236 ***
## factor(mpg_data$`model year`)79  4.6312039  0.8316267   5.569 4.91e-08 ***
## factor(mpg_data$`model year`)80  9.4007513  0.8769124  10.720  < 2e-16 ***
## factor(mpg_data$`model year`)81  6.5707143  0.8542335   7.692 1.30e-13 ***
## factor(mpg_data$`model year`)82  7.5082505  0.8429508   8.907  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.918 on 373 degrees of freedom
## Multiple R-squared:  0.8667, Adjusted R-squared:  0.8602
## F-statistic: 134.7 on 18 and 373 DF,  p-value: < 2.2e-16
```

c : What does R square of this model tell you ?

Solution : *R squared value **0.8667** for* **model10(mpg~cylinders + weight + horsepower + model year*** tells us that the model is good since higher R squared values signifies lower error.

d : * Can you reduce any independent variables to obtain a better model?

*** Solution : ***

Yes,I believe if we further break down the variables such as cylinders and model year into their respective factors,we can get a better regression model.