Regression Toyota Corolla Data

W. Grossmann

```
#============================================================
# Regression example: Prices of used Toyota Corolla
# Source: Ledolter: Data Mining and Business Analytics with R
#============================================================
#Packages
library(car)     # Used for scatterplot matrices
library(leaps)   # For subset regression
```

```
## Warning: package 'leaps' was built under R version 3.4.4
```

```
library(MASS)    # Support functions and data sets
library(DAAG)    # For cross validation in regression
```

```
## Warning: package 'DAAG' was built under R version 3.4.4
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'DAAG'
```

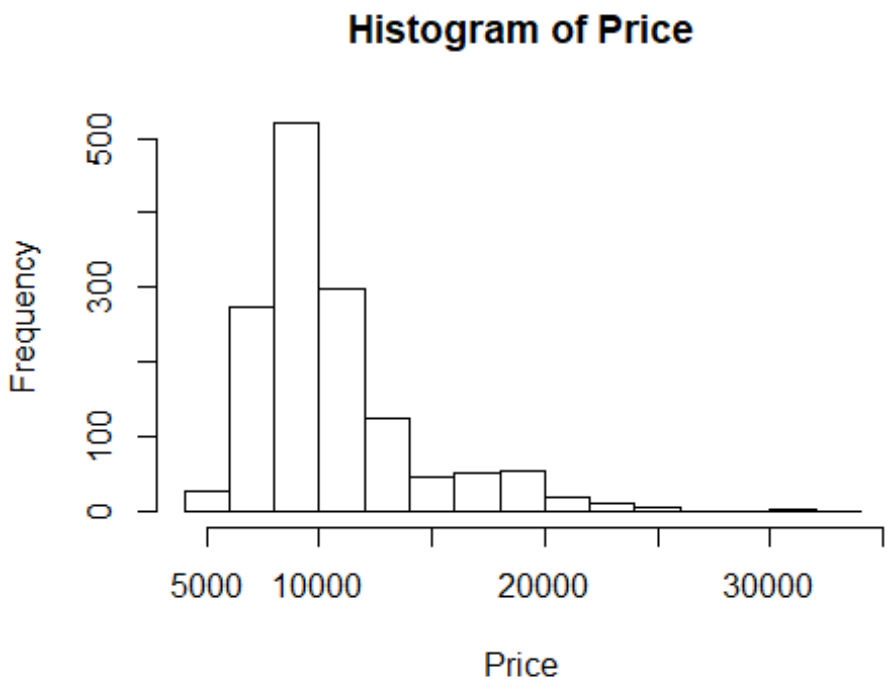
```
## The following object is masked from 'package:MASS':
##
##     hills
```

```
## The following object is masked from 'package:car':
##
##     vif
```

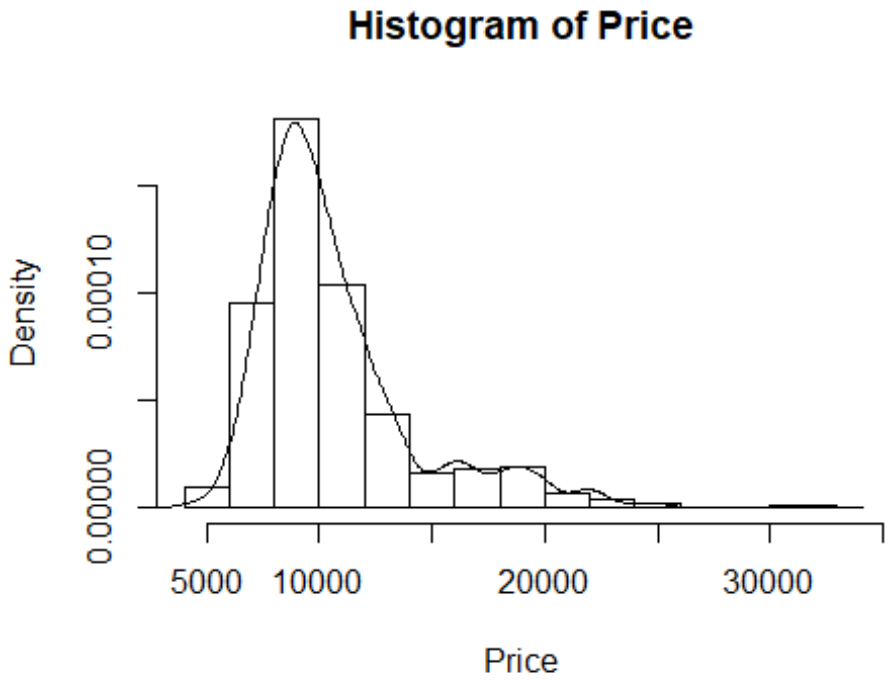
```
#=======================================================
#Data
#=======================================================
ToyotaCorolla1 <-
read.csv("ToyotaCorolla.csv", sep=";")
toyota<-as.data.frame(ToyotaCorolla1)
attach(toyota)
#=======================================================
# Summary Statistics and Data Preparation
#=======================================================

summary(toyota)
```

```
##      Price            Age              KM           FuelType
##  Min.   : 4350   Min.   : 1.00   Min.   :     1   CNG   :  17
##  1st Qu.: 8450   1st Qu.:44.00   1st Qu.: 43000   Diesel: 155
##  Median : 9900   Median :61.00   Median : 63451   Petrol:1263
##  Mean   :10733   Mean   :55.93   Mean   : 68581
```

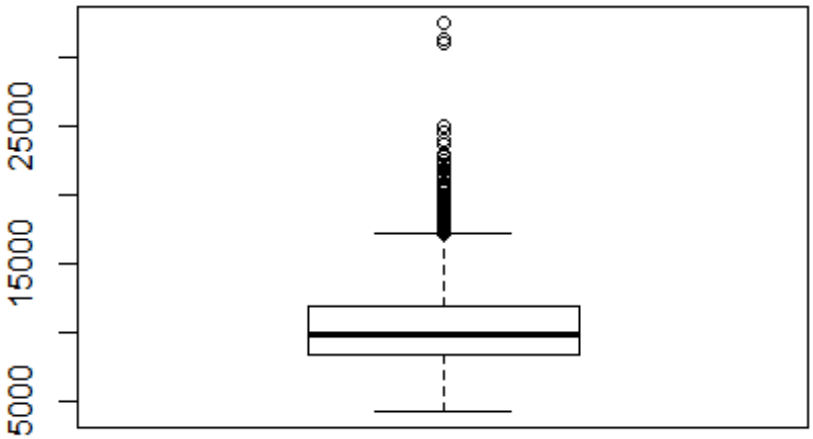
```
##  3rd Qu.:11950   3rd Qu.:70.00   3rd Qu.: 87042
##  Max.   :32500   Max.   :80.00   Max.   :243000
##        HP           MetColor        Automatic             CC
##  Min.   : 69.0   Min.   :0.0000   Min.   :0.00000   Min.   :1300
##  1st Qu.: 90.0   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:1400
##  Median :110.0   Median :1.0000   Median :0.00000   Median :1600
##  Mean   :101.5   Mean   :0.6753   Mean   :0.05575   Mean   :1567
##  3rd Qu.:110.0   3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:1600
##  Max.   :192.0   Max.   :1.0000   Max.   :1.00000   Max.   :2000
##      Doors           Weight
##  Min.   :2.000   Min.   :1000
##  1st Qu.:3.000   1st Qu.:1040
##  Median :4.000   Median :1070
##  Mean   :4.033   Mean   :1072
##  3rd Qu.:5.000   3rd Qu.:1085
##  Max.   :5.000   Max.   :1615

#-----------------------------------------------------
#Definition of Factors
#-----------------------------------------------------
metallic<-as.factor(MetColor)
metallic1<- factor(MetColor, labels =c("no","yes"))
summary(metallic1)

##  no yes
## 466 969

summary(metallic)

##   0   1
## 466 969

automatic<- as.factor(Automatic)
automatic1<- factor(Automatic, labels =c("no","yes"))
summary(automatic1)

##   no  yes
## 1355   80

#-----------------------------------------------------


#=====================================================
# Visualization of quantitative Variables
#=====================================================
hist(Price)
```



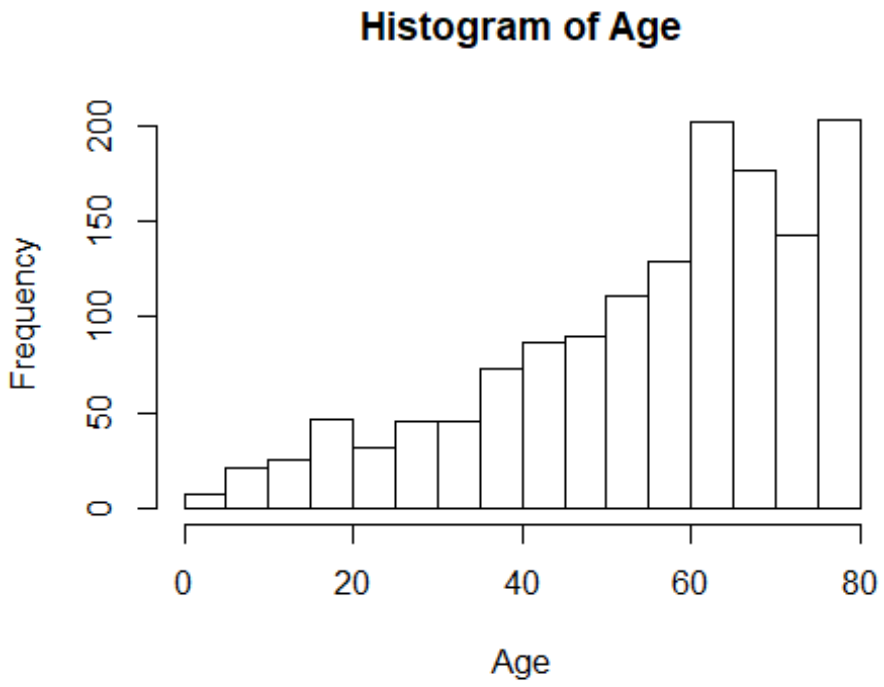

```
hist(Price, freq = FALSE)
lines(density(Price))
```



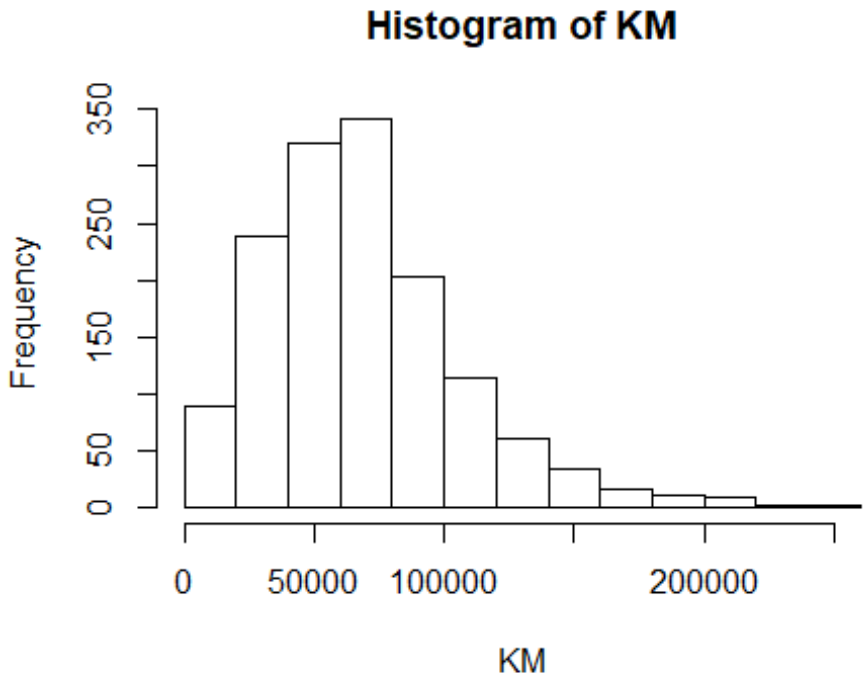

```
boxplot(Price)
```



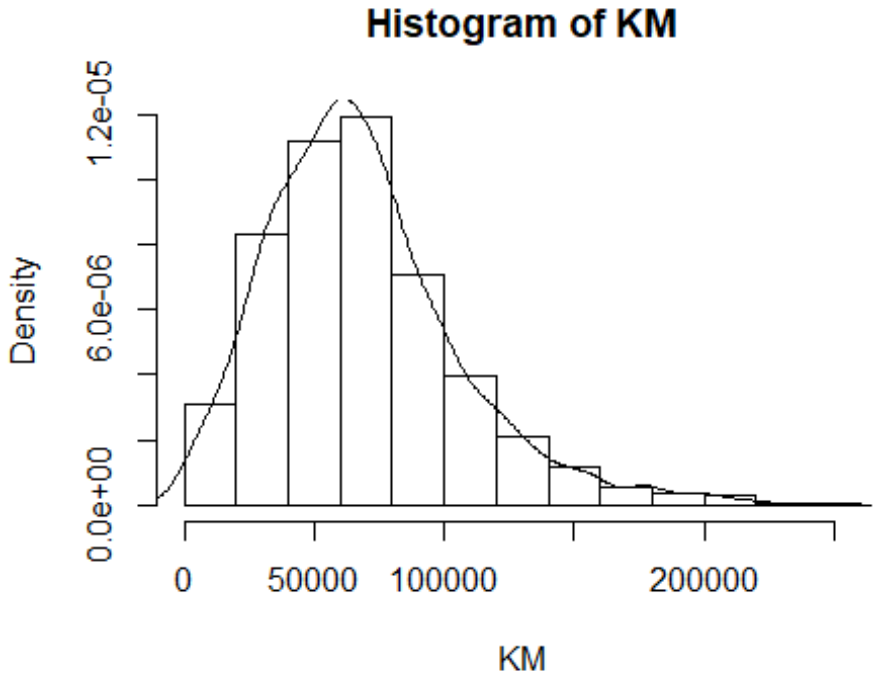

```
hist(Age)
```



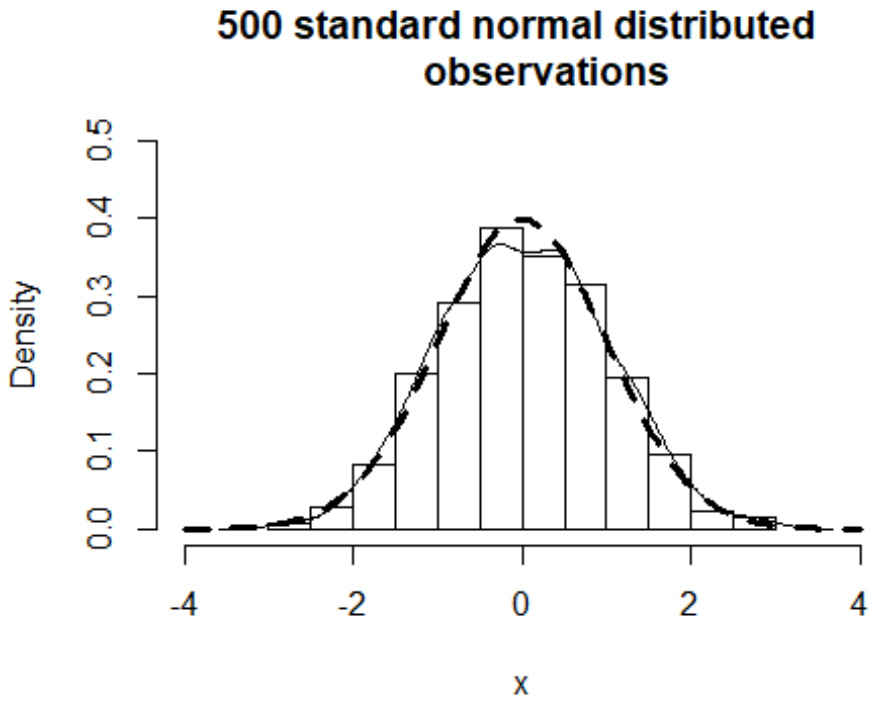

```
hist(KM)
```



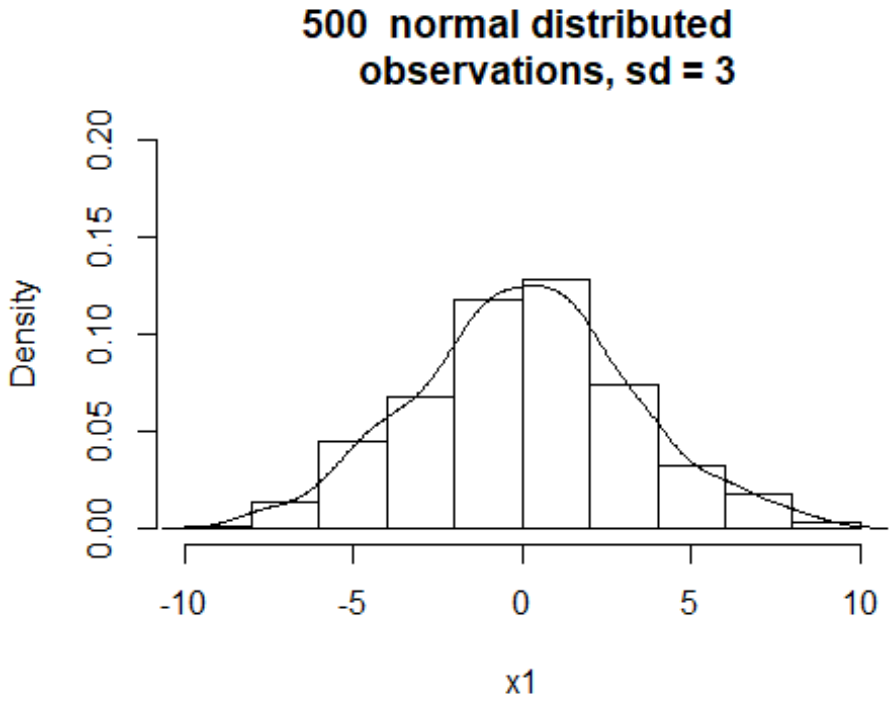

```
hist(KM,freq=FALSE)
lines(density(KM))
```



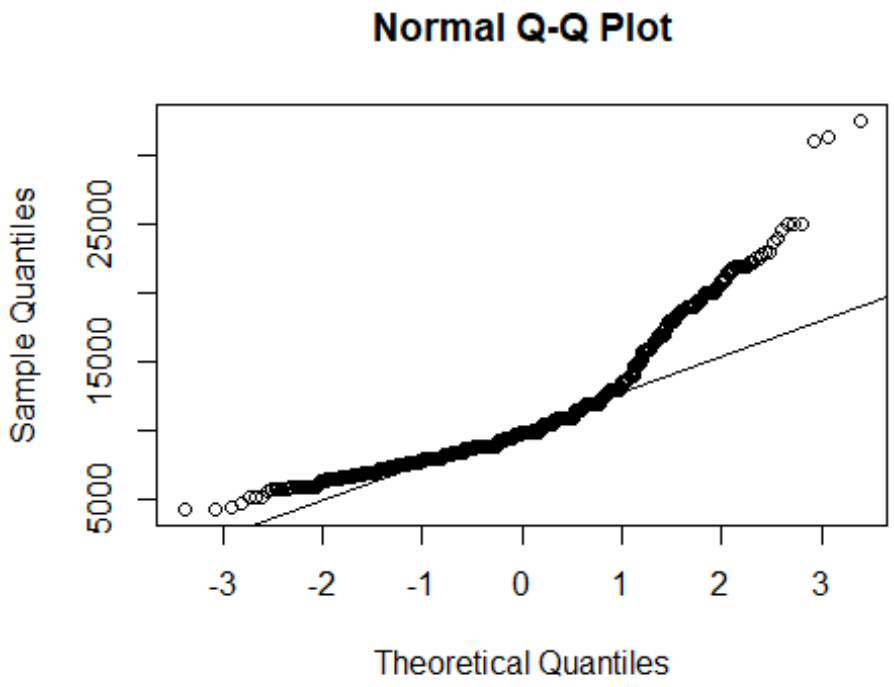
```r
#=====================================================
# Comparison with Normal distribution
#=====================================================

x<- rnorm(500, mean = 0, sd = 1)
x1<-rnorm(500, mean = 0, sd = 3)


hist(x, freq = FALSE, main="500 standard normal distributed
     observations", xlim = c(-4,4), ylim = c(0,0.5))
lines(density(x))
curve(dnorm, from=-4,to = 4, add = TRUE, lwd = 3, lty = 2)
```



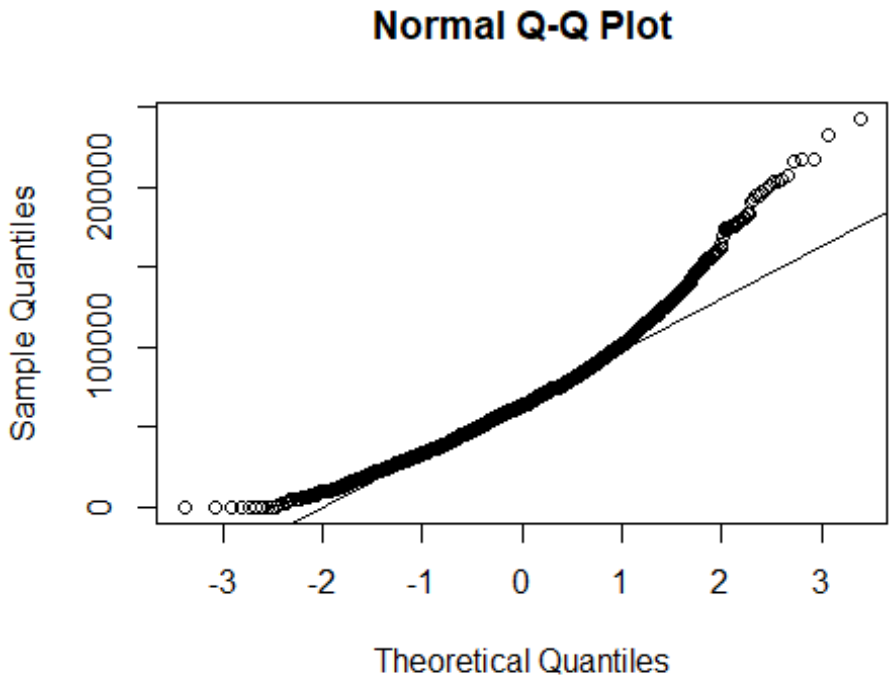

```r
hist(x1, freq = FALSE, main="500  normal distributed
     observations, sd = 3", xlim = c(-10,10), ylim = c(0,0.2))
lines(density(x1))
```



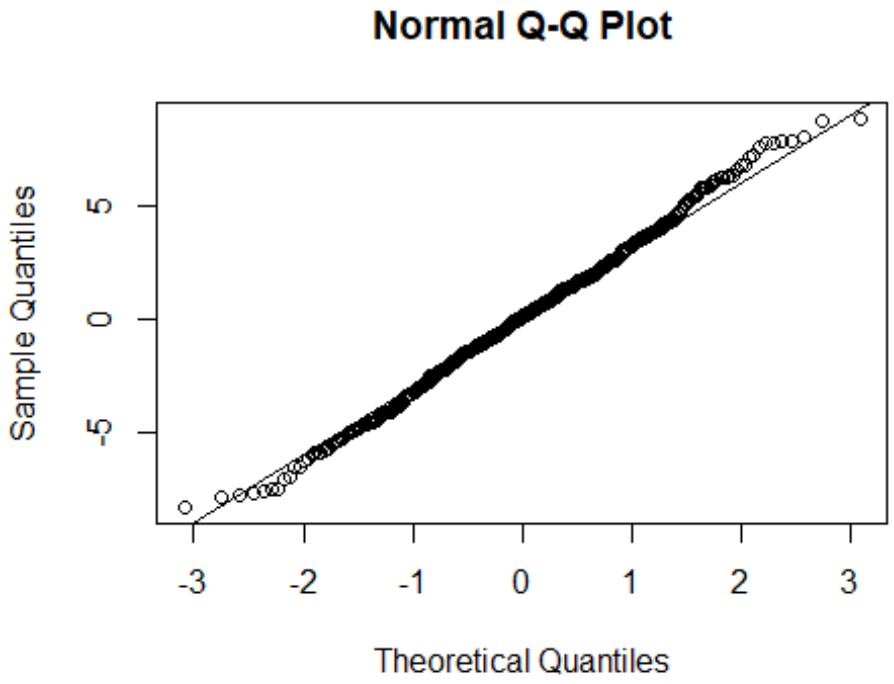

```
#=========================================================
#Assessing normal probability with q-q plots
#=========================================================
qqnorm(Price)
qqline(Price)
```



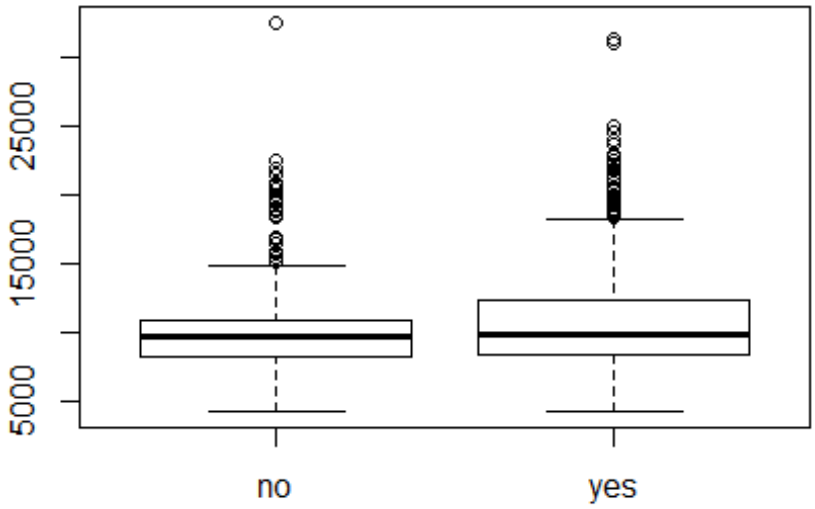

```
qqnorm(KM)
qqline(KM)
```



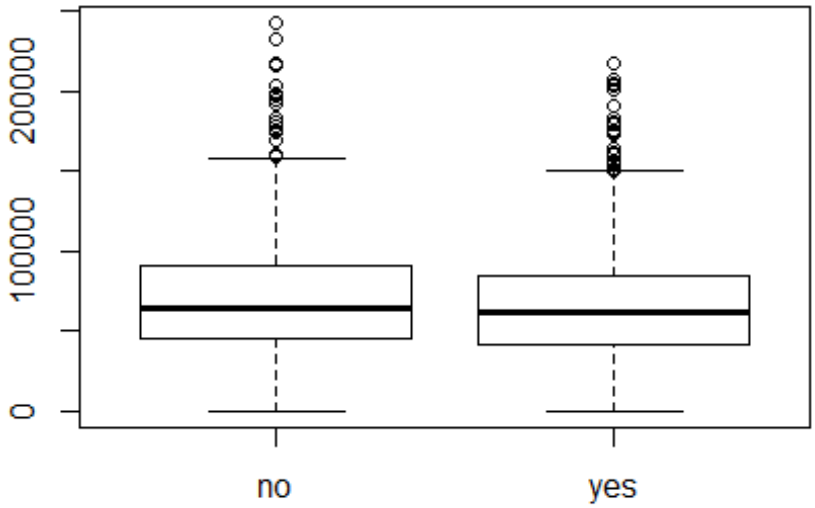
```r
qqnorm(x1)
qqline(x1)
```




```r
#=========================================================
#Distribution in groups
#=========================================================
boxplot(Price~metallic1)
```




```
boxplot(KM~metallic1)
```

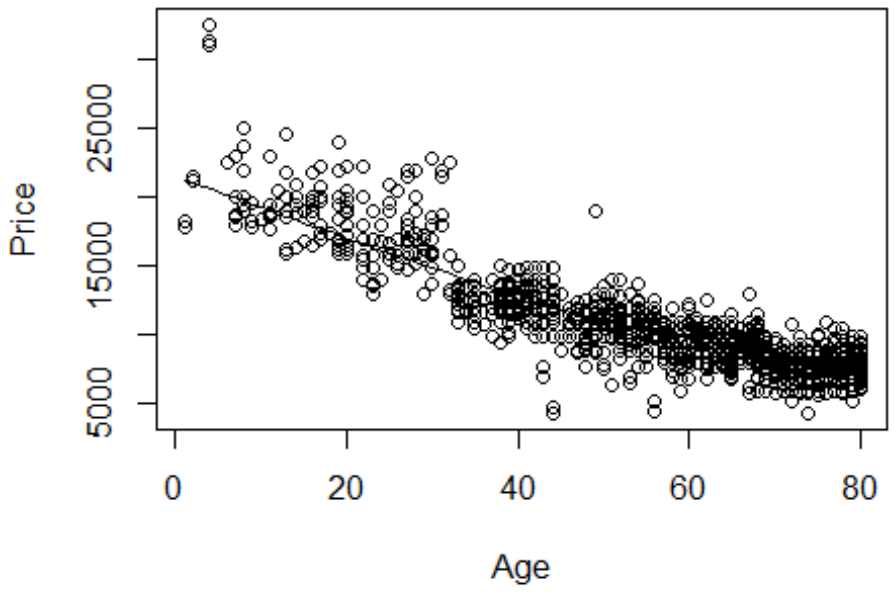



```
#=======================================================
# Comparison of means in Groups
```

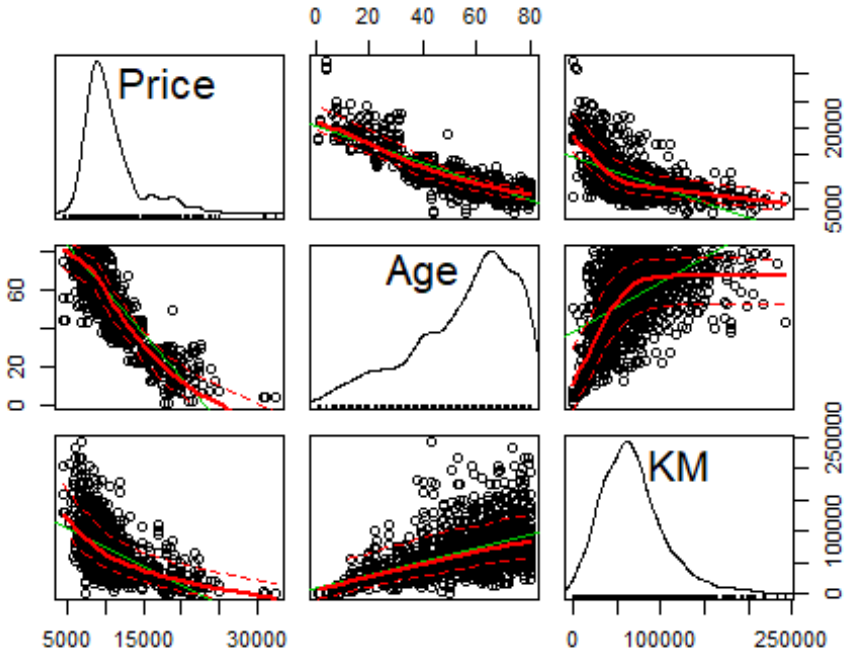
```r
#=====================================================
t.test(Price~metallic1)

##
##  Welch Two Sample t-test
##
## data:  Price by metallic1
## t = -4.3593, df = 1069.1, p-value = 1.43e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1212.2942  -459.7084
## sample estimates:
##  mean in group no mean in group yes
##          10168.94          11004.94

t.test(KM~metallic1)

##
##  Welch Two Sample t-test
##
## data:  KM by metallic1
## t = 3.0102, df = 830.14, p-value = 0.00269
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   2296.294 10902.861
## sample estimates:
##  mean in group no mean in group yes
##          73037.46          66437.88

wilcox.test(Price~metallic1)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Price by metallic1
## W = 199430, p-value = 0.0003358
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(KM~metallic1)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  KM by metallic1
## W = 242960, p-value = 0.01944
## alternative hypothesis: true location shift is not equal to 0

#=================================================
# Correlation and Scatterplots
#=================================================
data1<-cbind(Price, Age, KM, HP, Weight)
cor(data1)
```

```
##              Price        Age           KM           HP      Weight
## Price   1.0000000 -0.8764941 -0.57217147  0.31555715  0.58211818
## Age    -0.8764941  1.0000000  0.50784308 -0.15712980 -0.47113971
## KM     -0.5721715  0.5078431  1.00000000 -0.33324067 -0.02763113
## HP      0.3155572 -0.1571298 -0.33324067  1.00000000  0.08933133
## Weight  0.5821182 -0.4711397 -0.02763113  0.08933133  1.00000000
```

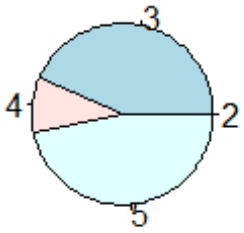
```
data2<-cbind(Price,Age, KM)
scatter.smooth(Age,Price)
```

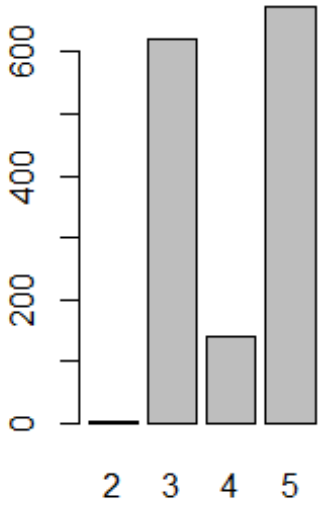



```
scatterplotMatrix(data2)
```

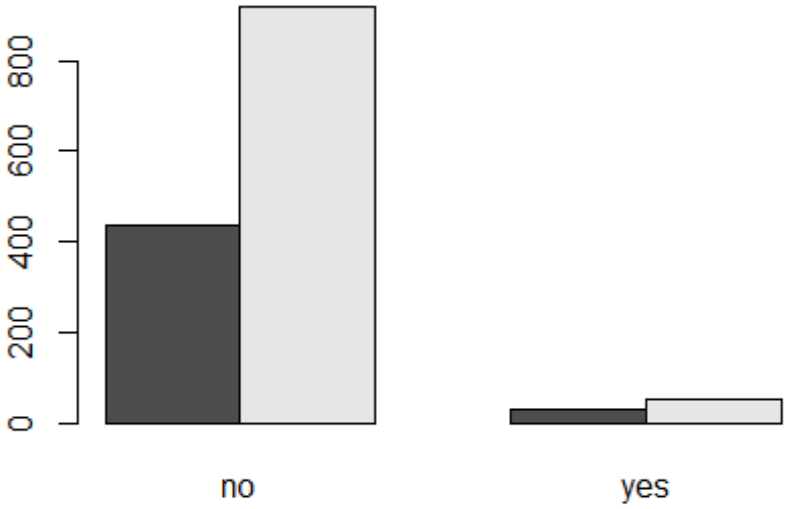



```
#=================================================
# Qualitative Variables
#=================================================
FreqDoor<-table(Doors)
table(Doors)

## Doors
##   2   3   4   5
##   2 622 138 673

pic1<-par(mfrow = c(1,2)) #Display with two plots in a row
pie(FreqDoor)
barplot(FreqDoor)
```

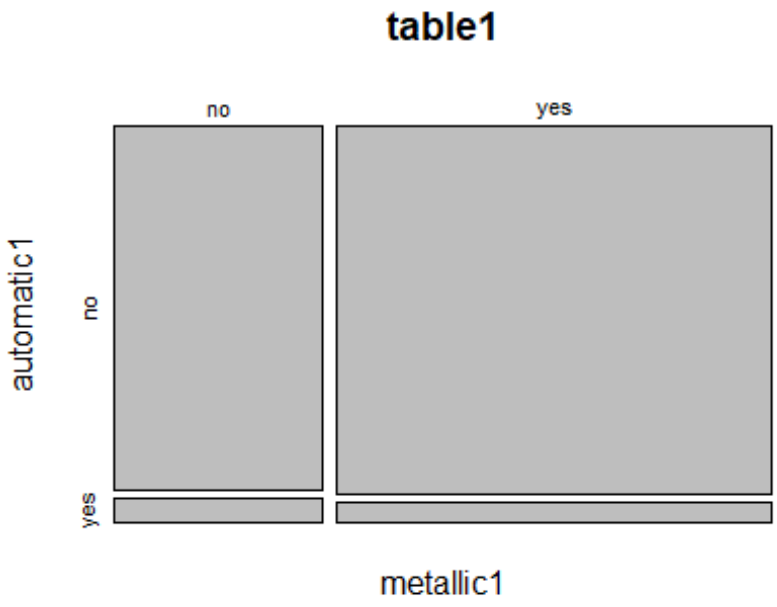





```
par(pic1) # Return to standarddisplay

table(FuelType)

## FuelType
##    CNG Diesel Petrol
##     17    155   1263

table1<-table(metallic1,automatic1 )
table1

##          automatic1
## metallic1  no yes
##       no  437  29
##       yes 918  51

barplot(table1, beside=TRUE)
```



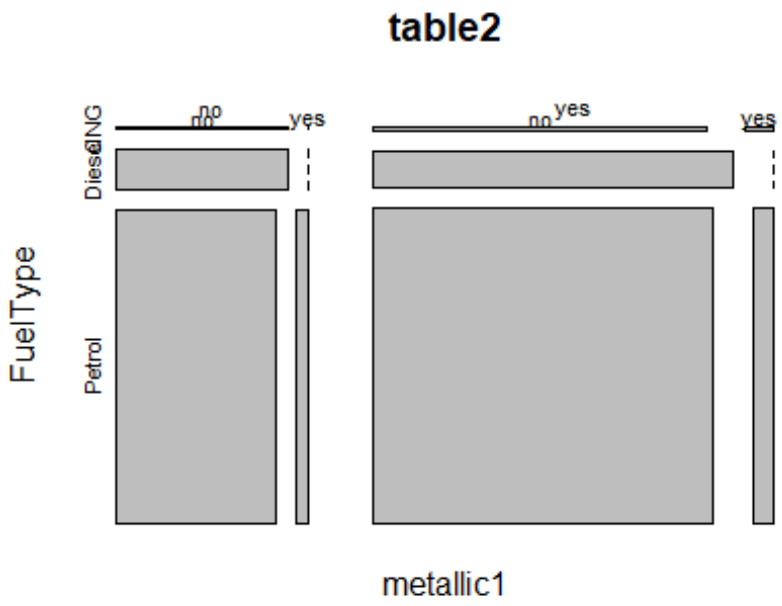

```
mosaicplot(table1)
```




```
table2<-table(metallic1,FuelType,automatic1 )
table2
```

```
## , , automatic1 = no
##
##           FuelType
## metallic1 CNG Diesel Petrol
##       no    4     53    380
##       yes  12    102    804
##
## , , automatic1 = yes
##
##           FuelType
## metallic1 CNG Diesel Petrol
##       no    0      0     29
##       yes   1      0     50
```

```
mosaicplot(table2)
```




```
# ============================================================
# Regression model with price as dependent variable (output)
#============================================================

auto<-cbind(Price, Age, KM, HP, Weight,metallic1)
auto<-as.data.frame(auto)
mod1<-lm(Price~., data = toyota)
summary(mod1)

##
## Call:
## lm(formula = Price ~ ., data = toyota)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10704.2   -743.0     -0.5    727.6   6452.2
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -3.948e+03  1.303e+03  -3.030 0.002486 **
## Age            -1.215e+02  2.606e+00 -46.605  < 2e-16 ***
## KM             -1.649e-02  1.315e-03 -12.543  < 2e-16 ***
## FuelTypeDiesel  3.363e+03  5.179e+02   6.494 1.15e-10 ***
## FuelTypePetrol  1.112e+03  3.317e+02   3.352 0.000822 ***
## HP              6.051e+01  5.746e+00  10.532  < 2e-16 ***
## MetColor        4.920e+01  7.486e+01   0.657 0.511118
## Automatic       3.204e+02  1.568e+02   2.043 0.041227 *
## CC             -4.147e+00  5.443e-01  -7.619 4.62e-14 ***
## Doors          -5.958e+00  3.999e+01  -0.149 0.881595
## Weight          2.013e+01  1.202e+00  16.752  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1313 on 1424 degrees of freedom
## Multiple R-squared:  0.8698, Adjusted R-squared:  0.8689
## F-statistic: 951.6 on 10 and 1424 DF,  p-value: < 2.2e-16
```

```
mod2<-lm(Price~., data = auto)
summary(mod2)
```

```
##
## Call:
## lm(formula = Price ~ ., data = auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10012.3   -764.4     -3.0    794.9   6236.1
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.176e+03  9.431e+02  -4.428 1.02e-05 ***
## Age         -1.218e+02  2.632e+00 -46.278  < 2e-16 ***
## KM          -1.993e-02  1.210e-03 -16.472  < 2e-16 ***
## HP           3.014e+01  2.536e+00  11.886  < 2e-16 ***
## Weight       1.867e+01  8.029e-01  23.251  < 2e-16 ***
## metallic1    5.601e+00  7.651e+01   0.073    0.942
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1348 on 1429 degrees of freedom
## Multiple R-squared:  0.8624, Adjusted R-squared:  0.8619
## F-statistic:  1791 on 5 and 1429 DF,  p-value: < 2.2e-16
```

```r
#---------------------------------------------
# Manual reduction of the model
#---------------------------------------------
anova(mod2)
```

```
## Analysis of Variance Table
##
## Response: Price
##             Df     Sum Sq    Mean Sq   F value Pr(>F)
## Age          1 1.4491e+10 1.4491e+10 7978.5956 <2e-16 ***
## KM           1 4.1030e+08 4.1030e+08  225.9010 <2e-16 ***
## HP           1 3.8362e+08 3.8362e+08  211.2150 <2e-16 ***
## Weight       1 9.8225e+08 9.8225e+08  540.8083 <2e-16 ***
## metallic1    1 9.7340e+03 9.7340e+03    0.0054 0.9417
## Residuals 1429 2.5955e+09 1.8163e+06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
drop1(mod2)
```

```
## Single term deletions
##
## Model:
## Price ~ Age + KM + HP + Weight + metallic1
##           Df  Sum of Sq        RSS   AIC
## <none>                  2595452433 20688
## Age        1 3889869541 6485321974 22000
## KM         1  492804373 3088256807 20935
## HP         1  256615867 2852068300 20821
## Weight     1  981919748 3577372181 21146
## metallic1  1       9734 2595462167 20686
```

```r
#---------------------------------------------
#automatic subset selection with package leap
#---------------------------------------------
X<-auto[,-1]
Y<-auto[,1]
out=summary(regsubsets(X,Y,nbest=2,nvmax=ncol(X)))
tab=cbind(out$which,out$rsq,out$adjr2,out$cp)
tab
```

```
##   (Intercept) Age KM HP Weight metallic1
## 1           1   1  0  0      0         0 0.7682419 0.7680802  975.929647
## 1           1   0  0  0      1         0 0.3388616 0.3384002 5435.269747
## 2           1   1  0  0      1         0 0.8050240 0.8047517  595.928265
## 2           1   1  0  1      0         0 0.8006673 0.8003889  641.174425
## 3           1   1  1  0      1         0 0.8487793 0.8484623  143.506391
## 3           1   1  0  1      1         0 0.8362572 0.8359139  273.555293
## 4           1   1  1  1      1         0 0.8624041 0.8620192    4.005359
## 4           1   1  1  0      1         1 0.8488004 0.8483775  145.287149
## 5           1   1  1  1      1         1 0.8624046 0.8619232    6.000000
```

```r
#Conclusion: A model without metallic, deleting HP also reasonable
auto1<-auto[, -6]
auto2<-auto[,-c(4,6)]
head(auto2)
```

```
##   Price Age    KM Weight
## 1 13500  23 46986   1165
## 2 13750  23 72937   1165
## 3 13950  24 41711   1165
## 4 14950  26 48000   1165
## 5 13750  30 38500   1170
## 6 12950  32 61000   1170
```

```r
#============================================================
# Selection with package mass
#============================================================
mod.sec <- lm(Price~.,data=toyota)
step <- stepAIC(mod.sec, direction="both")
```

```
## Start:  AIC=20618.03
## Price ~ Age + KM + FuelType + HP + MetColor + Automatic + CC +
##     Doors + Weight
##
##              Df  Sum of Sq        RSS   AIC
## - Doors       1      38268 2455428762 20616
## - MetColor    1     744884 2456135378 20617
## <none>                     2455390494 20618
## - Automatic   1    7197616 2462588110 20620
## - FuelType    2   72771536 2528162030 20656
## - CC          1  100105707 2555496201 20673
## - HP          1  191269800 2646660294 20724
## - KM          1  271298730 2726689225 20766
## - Weight      1  483874508 2939265002 20874
## - Age         1 3745212303 6200602798 21945
##
## Step:  AIC=20616.05
## Price ~ Age + KM + FuelType + HP + MetColor + Automatic + CC +
##     Weight
##
##              Df  Sum of Sq        RSS   AIC
## - MetColor    1     727044 2456155806 20615
## <none>                     2455428762 20616
## + Doors       1      38268 2455390494 20618
## - Automatic   1    7376542 2462805304 20618
## - FuelType    2   76184800 2531613562 20656
## - CC          1  103428111 2558856873 20673
## - HP          1  199958045 2655386807 20726
## - KM          1  272641992 2728070754 20765
## - Weight      1  548161657 3003590419 20903
## - Age         1 3745178112 6200606873 21943
##
```

```
## Step:  AIC=20614.47
## Price ~ Age + KM + FuelType + HP + Automatic + CC + Weight
##
##             Df  Sum of Sq        RSS   AIC
## <none>                     2456155806 20615
## + MetColor   1     727044 2455428762 20616
## + Doors      1      20427 2456135378 20617
## - Automatic  1    7264688 2463420494 20617
## - FuelType   2   75591718 2531747523 20654
## - CC         1  102768498 2558924303 20671
## - HP         1  199373638 2655529443 20725
## - KM         1  273991954 2730147760 20764
## - Weight     1  548885285 3005041090 20902
## - Age        1 3762860225 6219016030 21946
```

```
step$anova # display results
```

```
## Stepwise Model Path
## Analysis of Deviance Table
##
## Initial Model:
## Price ~ Age + KM + FuelType + HP + MetColor + Automatic + CC +
##     Doors + Weight
##
## Final Model:
## Price ~ Age + KM + FuelType + HP + Automatic + CC + Weight
##
##
##         Step Df  Deviance Resid. Df Resid. Dev      AIC
## 1                              1424 2455390494 20618.03
## 2    - Doors  1  38267.63      1425 2455428762 20616.05
## 3 - MetColor  1 727043.95      1426 2456155806 20614.47
```

```
#=========================================================
# Splitting the data for estimation of generalization error
#=========================================================
mod<-lm(Price~., data = auto1)



#Splitting randomly in  training and test data
set.seed(1041)
## fixing the seed for random selection useful for
# obtaining comparable results in case of different tests
n=length(auto$Price)
n1=1000
n2=n-n1
train=sample(1:n,n1)

#Regression for training data
```

```r
mod.train=lm(Price~.,data=auto1[train,])
summary(mod.train)
```

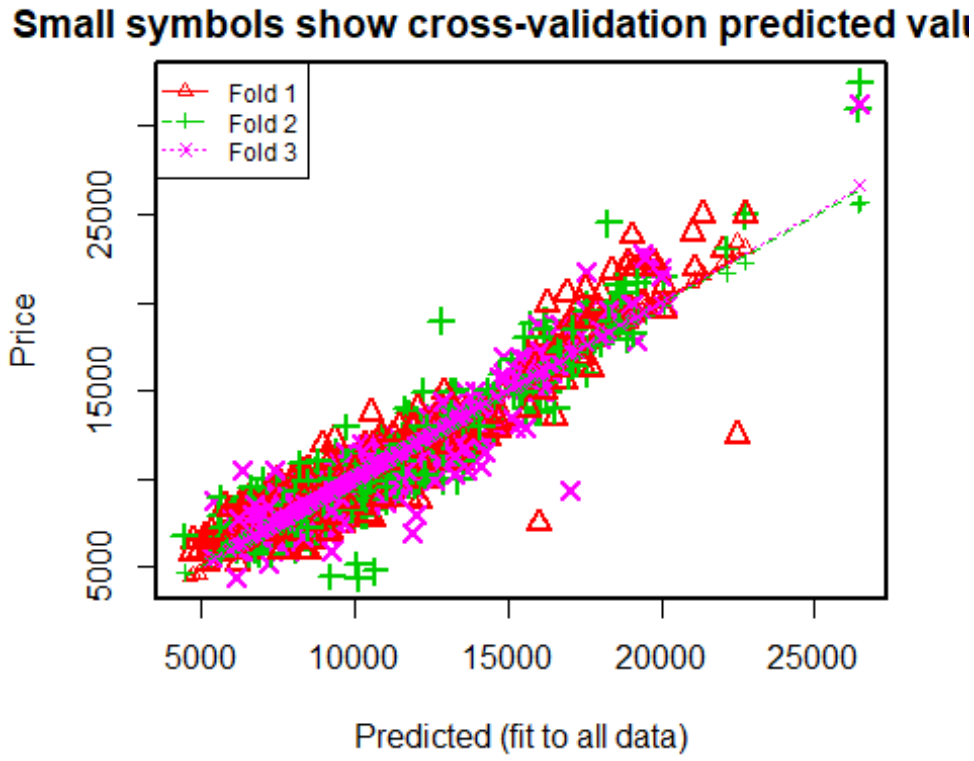
```
##
## Call:
## lm(formula = Price ~ ., data = auto1[train, ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8940.1  -760.4   -16.4   798.0  6778.7
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.753e+03  1.074e+03  -1.632    0.103
## Age         -1.271e+02  3.127e+00 -40.660   <2e-16 ***
## KM          -1.758e-02  1.447e-03 -12.146   <2e-16 ***
## HP           2.981e+01  3.102e+00   9.610   <2e-16 ***
## Weight       1.657e+01  9.079e-01  18.252   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1352 on 995 degrees of freedom
## Multiple R-squared:  0.8577, Adjusted R-squared:  0.8572
## F-statistic:  1500 on 4 and 995 DF,  p-value: < 2.2e-16
```

```r
#Prediction for test data
pred=predict(mod.train,newdat=auto1[-train,])

#Prediction error
obs=auto$Price[-train]
diff=obs-pred
percdiff=abs(diff)/obs
me=mean(diff)
rmse=sqrt(sum(diff**2)/n2)

mape=100*(mean(percdiff))

error<-cbind(me,rmse, mape)
colnames(error)<-
cbind("mean error",
"  root mean square error",
                    "  mean absolute percent error")
error
```

```
##      mean error   root mean square error   mean absolute percent error
## [1,]   22.26225                 1349.903                       10.08488
```

```r
#=================================================
#Cross validation using DAAG
#=================================================
```

```
cv.lm( auto1,    form.lm = formula(Price ~ Age + KM + HP + Weight ),
                m=3, dots = FALSE, seed=29, plotit=TRUE, printit=FALSE)

## Warning in cv.lm(auto1, form.lm = formula(Price ~ Age + KM + HP + Weight),
:
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```




```
#=====================================================
# Cross validation if one is left out
#=====================================================
n=length(auto1$Price)
diff=dim(n)
percdiff=dim(n)
for (k in 1:n) {
  train1=c(1:n)
  train=train1[train1!=k]
  m1=lm(Price~.,data=auto1[train,])
  pred=predict(m1,newdat=auto[-train,])
  obs=auto$Price[-train]
  diff[k]=obs-pred
  percdiff[k]=abs(diff[k])/obs
}
me1=mean(diff)
rmse1=sqrt(mean(diff**2))
```

```r
mape1=100*(mean(percdiff))
me1   # mean error
```

```
## [1] -0.04898206
```

```r
rmse1 # root mean square error
```

```
## [1] 1361.318
```

```r
mape1 # mean absolute percent error
```

```
## [1] 9.995245
```