Statische Modellen & Data-analyse

Homework 2

Andreas Put Li Quan

May 23, 2011

1. Introduction

We analyze the *autos* dataset of the UCI Machine Learning Repository which can be found at `http://archive.ics.uci.edu/ml/machine-learning-databases/autos/`. We first give a short overview of the dataset and than discuss our analysis of the dataset.

2. Description dataset

First some background information and context about the dataset:

- Creator/Donor: Jeffrey C. Schlimmer (`Jeffrey.Schlimmer@a.gp.cs.cmu.edu`)
- Date: May 19, 1987
- Sources:
  1. 1985 Model Import Car and Truck Specifications, 1985 Ward's Automotive Yearbook.
  2. Personal Auto Manuals, Insurance Services Office, 160 Water Street, New York, NY 10038.
  3. Insurance Collision Report, Insurance Institute for Highway Safety, Watergate 600, Washington, DC 20037.

The data set consists of three types of entities: (a) the specification of an auto in terms of various characteristics (a.o. length, width, city and highway fuel mileage), (b) its assigned insurance risk rating, and (c) its normalized losses in use as compared to other cars.

The risk rating corresponds to the degree to which the auto is more risky than its price indicates. Cars are initially assigned a risk factor symbol associated with its price. Then, if it is more risky (or less), this symbol is adjusted by moving it up (or down) the scale. Actuarians call this process "symboling". A value of $+3$ indicates that the auto is risky, -3 that it is probably pretty safe.

The relative average loss payment per insured vehicle year is a normalized value for all autos within a particular size classification (two-door small, station wagons, sports/speciality, etc.), and represents the average loss per car per year.

All the attributes including their type and range can be found in Table 1. The units used are the US customary units[1], e.g. prices are given in US$, masses in pounds and lengths in inches.

3. Dataset exploration and preprocessing

The dataset contains $p = 26$ attributes, and $n = 206$ observations. It also contains some instances with missing attribute values, especially for normalized losses. Because we have a large amount of data ($n > 5p$), we chose to simply omit observations with any missing attribute values. If we do this, we still have plenty of data to use—more specifically $n = 159$ observations—which is why we chose this method instead of replacing them with for instance variable means or interpolations [1].

We have an *exploratory observational* dataset. A tool such as ggobi[2] allows us to visualize the data in different ways to get familiar with the dataset. This way, we interactively searched for extreme outliers and screened explanatory variables.

We found out that the variable 'engine-location' only had 3 instances of 'rear' (0 after removing instances with missing values) which is why we can easily discard this variable. Furthermore, as we have such a highly dimensional dataset, for our analysis we will mostly only consider the numerical attributes and one categorical variable (that is, 'drive-wheels'). We will focus on regression models where the response variable is the price (this was also the goal in [3] where an instance-based machine learning model was used). Some plots of the response variable are shown in Figure 1. Clearly, this is still quite some data to process and to analyze.

Figure 2 shows the visualization of the correlation matrix: it is clear that many variables are highly correlated together, for instance, we have an almost perfect correlation between 'city.mpg' and 'highway.mpg', and 'length' and 'width'. So this means we have to pay extra attention for multicollinearity effects.

Figure 4 shows that the price variable should be transformed. We can test this formally using the Shapiro-Wilk normality test on the price.

```
Shapiro-Wilk normality test
data:  autos$price
W = 0.829, p-value = 2.351e-12
```

After a (common) logarithmic transformation,

```
Shapiro-Wilk normality test
data:  autos2$price
W = 0.9411, p-value = 3.576e-06
```

we get a better but still a rather poor result. Using the (more general) BoxCox transformation, where the parameter $\lambda = -0.4$ was found using maximization of the log-likelihood (Figure 3), we get a much better result:

[1] http://en.wikipedia.org/wiki/United_States_customary_units
[2] http://www.ggobi.org/

```
Shapiro-Wilk normality test
data:  autosNum$price
W = 0.9634, p-value = 0.0003285
```

We also see that the boxplot of drive-wheels versus price (Figure 5) has improved using this transformation.

4. PCA and PCR/PLS

Because our dataset is highly dimensional, Principal Component Analysis (PCA) can be used for data reduction. Using this method, we can convert a set of possible correlated variables to a set of uncorrelated (principal) components.

For non-numerical attributes we could do PCA introducing binary variables (which would attain comparable results to Multiple Correspondence Analysis (MCA)). However, for betters results we would have to to find a suitable way to represent distances between variable categories and individuals in the factorial space. This can be done for instance using Gifi Methods for Optimal Scaling [2], implemented in the R package 'homals'. This seems however way out of scope so for our PCA we will simply limit our dataset to the 15 numerical attributes.

A Partial Least Squares Regression will also be conducted, and we will determine the best method for this dataset.

4.1. Principal Component Regression

The first step in the PCR is the PCA itself (Figure 6b). Using the screeplot in Figure 6a we can use about 4–6 PC's to account for 80–90% of the variance:

```
                          PC1    PC2    PC3     PC4     PC5     PC6
Standard deviation     2.6478 1.5063 1.2006 0.95939 0.83978 0.65437
Proportion of Variance 0.5008 0.1621 0.1029 0.06574 0.05037 0.03059
Cumulative Proportion  0.5008 0.6629 0.7658 0.83154 0.88192 0.91250
```

For our analysis we use the 4 first PC's (see Figure 7a). The resulting model has a RMSE of 2745. The results of the regression analysis are shown below:

```
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       11445.73     217.69  52.579  < 2e-16 ***
autos2.pca$x[, 1] -1936.92      82.47 -23.485  < 2e-16 ***
autos2.pca$x[, 2]    32.99     144.97   0.228    0.820
autos2.pca$x[, 3]   783.43     181.89   4.307 2.93e-05 ***
autos2.pca$x[, 4]  -132.65     227.62  -0.583    0.561
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 2745 on 154 degrees of freedom
Multiple R-squared: 0.7874,Adjusted R-squared: 0.7819
F-statistic: 142.6 on 4 and 154 DF,  p-value: < 2.2e-16
```

After applying the Box-Cox transformation on the price, the resulting model has an RMSE of 2822.113. This is greater then the first model, but when we study the results (see Figure 7b), we see that 1 price-estimate is responsible for this bad RMSE.[3]

```
Coefficients:
                    Estimate Std. Error  t value Pr(>|t|)
(Intercept)        2.4369983  0.0003124 7799.794  < 2e-16 ***
autos2.pca$x[, 1] -0.0036713  0.0001184  -31.014  < 2e-16 ***
autos2.pca$x[, 2]  0.0003439  0.0002081    1.653  0.10044
autos2.pca$x[, 3]  0.0007255  0.0002611    2.779  0.00613 **
autos2.pca$x[, 4]  0.0000775  0.0003267    0.237  0.81280
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.00394 on 154 degrees of freedom
Multiple R-squared: 0.8633,Adjusted R-squared: 0.8597
F-statistic: 243.1 on 4 and 154 DF,  p-value: < 2.2e-16
```

We can see that this model is quite better than the first model: the R^2 value is slightly better, although the RMSE value is larger than the first one. The summary of these models are located at Summary 1. We can see that the mean of this model is worse than the mean of the first model, which explains the worse RMSE. However, the variance of this model is very close to the real variance, which can be seen by the median and quantiles. This explains the better R^2 value for the PCR with Box-Cox.

4.2. Partial Least Squares

Instead of finding hyperplanes of maximum variance between the response and regressions, PLS finds a linear regression model by projecting the predicted variables and the observable variables to a new space. So again, this is a nice method for our high dimensional dataset.

After applying the partial least squares (Figure 8) the following results are visible:

```
       (Intercept)   1 comps   2 comps   3 comps   4 comps
CV         0.01055  0.004031  0.003889  0.003855  0.003877
adjCV      0.01055  0.004025  0.003876  0.003842  0.003861

TRAINING: % variance explained
                1 comps  2 comps  3 comps  4 comps
X                 50.04    60.22    70.75    80.19
autos2.bcprice    86.14    87.92    88.42    88.60
```

[3] While studying this model, we have to keep in mind that the price values are transformed.

With this PLS regression, a cross validation check is conducted. The details of the predicted price values are shown in Summary 1.

Summary 1 Summary of the PCR and PLS models

```
> summary - Price
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   5120    7370    9230   11400   14700   35100
> summary - PCR price
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  -1892    7121   10870   11450   15320   28730
> summary - PCRboxcox price
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   4200    7310    9570   11300   13900   53800
> summary - PLSR price
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  -2110    7020   10900   11400   15500   27000
```

We see that the PLSR predictions have a better estimate with regard to the mean, but when we look at the quantiles, the PCR with the Box-Cox transformation still wins. Even the PCR without the Box-Cox transformation yiels better values for the variance and median. We can conclude for this dataset that a PCR method is preferable to a PLSR method.

5. Linear models

First we build a linear model using all numeric variables to predict the price. We see that quite a lot hypotheses $H_0 : \beta = 0$ are not rejected. The overall F-test indicates clearly that there is a regression relation between our response variable and the predictors ($p < 2.2 \times 10^{-16}$).

```
Call:
lm(formula = price ~ ., data = autosWorking)

Residuals:
       Min         1Q     Median         3Q        Max
-0.0051952 -0.0014309 -0.0002276  0.0013448  0.0053776

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        2.119e+00  1.555e-02 136.330  < 2e-16 ***
normalized.losses  1.022e-05  6.956e-06   1.469 0.143919
```

```
wheel.base          4.007e-05  9.640e-05   0.416 0.678232
length              8.733e-05  4.624e-05   1.889 0.060982 .
width               5.090e-04  2.290e-04   2.223 0.027810 *
height              1.080e-04  1.319e-04   0.819 0.414314
curb.weight         2.964e-06  1.671e-06   1.774 0.078204 .
engine.size        -2.233e-07  1.834e-05  -0.012 0.990305
bore               -7.929e-04  1.117e-03  -0.710 0.478778
stroke             -2.472e-04  7.598e-04  -0.325 0.745422
compression.ratio   2.038e-04  7.406e-05   2.752 0.006692 **
horsepower          5.511e-05  1.605e-05   3.433 0.000783 ***
peak.rpm            2.369e-07  5.563e-07   0.426 0.670845
city.mpg           -3.737e-04  1.543e-04  -2.421 0.016737 *
highway.mpg         2.008e-04  1.410e-04   1.424 0.156651
drive.wheelsfwd    -1.399e-03  1.090e-03  -1.284 0.201288
drive.wheelsrwd     2.010e-04  1.166e-03   0.172 0.863341
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.002272 on 142 degrees of freedom
Multiple R-squared: 0.8937,Adjusted R-squared: 0.8818
F-statistic: 74.64 on 16 and 142 DF,  p-value: < 2.2e-16
```

Figure 9 shows some diagnostic plots of this model: these indicate that in general our model is appropriate, but there are some small issues.

Of course, in general we want a simpler model. This dataset has too many variables to properly determine the best regressors for linear regression by hand. That is why we use an automated technique for variable selection: stepwise regression (using the Akaike information criterion). The variables that are selected with this method are thus: curb-weight, horsepower, length, normalized-losses, width, drive-wheels, compression-ratio and city-mpg. Figure 10 shows some diagnostic plots of this model, which performs comparable to the full model.

```
Call:
lm(formula = price ~ curb.weight + horsepower + length + normalized.losses +
    width + compression.ratio + city.mpg + drive.wheels, data = autosWorking)

Residuals:
      Min        1Q    Median        3Q       Max
-0.005600 -0.001402 -0.000225  0.001359  0.005609

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        2.129e+00  1.222e-02 174.190  < 2e-16 ***
curb.weight        2.939e-06  1.317e-06   2.231  0.02716 *
```

```
horsepower         4.797e-05  1.337e-05   3.589  0.00045 ***
length             1.165e-04  3.631e-05   3.208  0.00164 **
normalized.losses  1.071e-05  5.792e-06   1.848  0.06654 .
width              4.343e-04  2.063e-04   2.105  0.03699 *
compression.ratio  1.967e-04  7.166e-05   2.746  0.00678 **
city.mpg          -1.887e-04  7.429e-05  -2.541  0.01209 *
drive.wheelsfwd   -9.538e-04  9.334e-04  -1.022  0.30848
drive.wheelsrwd    4.415e-04  1.005e-03   0.440  0.66091
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.00225 on 149 degrees of freedom
Multiple R-squared: 0.8906,Adjusted R-squared: 0.884
F-statistic: 134.8 on 9 and 149 DF,  p-value: < 2.2e-16
```

6. ANOVA

Consider the difference in price between autos based on the form of engine/transmission layout used in the motor vehicles—where the engine either drives only the front wheels (fwd), rear wheels (rwd) or both (4wd). The boxplot in Figure 5 suggests that there are differences amongst them.

First we check the normality assumption (Figure 11): the Q-Q plot does not give a strong impression that the prices for the different wheel-drives are not normal distributed. The Shapiro-Wilk test is used to confirm this. Summary 2 shows that the front and rear wheel-drive have a low p-value. They are probably not exactly normal, but it is close enough on the decision boundary with $\alpha = 0.025$.

Summary 2 Shapiro-Wilk test: Prices wheel drives

```
data:  autosWorking.4wd
W = 0.888, p-value = 0.224
data:  autosWorking.fwd
W = 0.9737, p-value = 0.03496
data:  autosWorking.rwd
W = 0.9444, p-value = 0.0286
```

There is little evidence to doubt the homoskedasticity assumption as shown by Levene's test ($p = 0.74$):

```
Levene's Test for Homogeneity of Variance (center = median)
       Df F value Pr(>F)
group   2  0.2954 0.7446
      156
```

We test this using ANOVA and confirm that there exists a difference among them.

```
              Df    Sum Sq   Mean Sq F value    Pr(>F)
drive.wheels   2 0.0082212 0.0041106  69.231 < 2.2e-16 ***
Residuals    156 0.0092624 0.0000594
```

Using the Tukey-HSD test we search which groups differ. The result is that the rear wheel drive systems are significantly more expensive.

```
Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = price ~ drive.wheels, data = autosWorking)

$drive.wheels
                diff          lwr         upr     p adj
fwd-4wd -0.002790591 -0.006999128 0.001417946 0.2621226
rwd-4wd  0.007253770  0.002858308 0.011649232 0.0004095
rwd-fwd  0.010044360  0.008015527 0.012073194 0.0000000
```

7. Conclusion

Because of the high dimension of our dataset (even after our simplification), it was quite difficult to find a good regression model (curse of dimensionality). Techniques like PCA, PCR and PLS are good techniques for this problem and are reasonably efficient to compute. A major disadvantage is the interpretation of the transformed regressors.

For the general linear models, the variable selection was a problem. This was resolved using the stepwise regression method, which automatically selected a set of suitable regressors. This made the model easier to work with and to understand.

The data preprocessing step was very important: the non-normality of the data was solved using a power transformation (more specifically, a BoxCox transformation). This resulted in better models, but this made the data more difficult to understand.

A. Tables and Figures

List of Tables

List of Figures

Attribute	**Attribute Type**	**Attribute Range**
symboling	ordinal	-3, -2, -1, 0, 1, 2, 3
normalized-losses	numerical	continuous from 65 to 256
make	categorical	alfa-romero, audi, bmw, chevrolet, dodge, honda, isuzu, jaguar, mazda, mercedes-benz, mercury, mitsubishi, nissan, peugot, plymouth, porsche, renault, saab, subaru, toyota, volkswagen, volvo
fuel-type	categorical	diesel, gas
aspiration	categorical	std, turbo
num-of-doors	ordinal	two, four
body-style	categorical	hardtop, wagon, sedan, hatchback, convertible
drive-wheels	categorical	4wd, fwd, rwd
engine-location	categorical	front, rear
wheel-base	numerical	continuous from 86.6 120.9
length	numerical	continuous from 141.1 to 208.1
width	numerical	continuous from 60.3 to 72.3
height	numerical	continuous from 47.8 to 59.8
curb-weight	numerical	continuous from 1488 to 4066
engine-type	categorical	dohc, dohcv, l, ohc, ohcf, ohcv, rotor
num-of-cylinders	ordinal	two, three, four, five, six, eight, twelve
engine-size	numerical	continuous from 61 to 326
fuel-system	categorical	1bbl, 2bbl, 4bbl, idi, mfi, mpfi, spdi, spfi
bore	numerical	continuous from 2.54 to 3.94
stroke	numerical	continuous from 2.07 to 4.17
compression-ratio	numerical	continuous from 7 to 23
horsepower	numerical	continuous from 48 to 288
peak-rpm	numerical	continuous from 4150 to 6600
city-mpg	numerical	continuous from 13 to 49
highway-mpg	numerical	continuous from 16 to 54
price	numerical	continuous from 5118 to 45400

Table 1: Description of *autos* dataset.

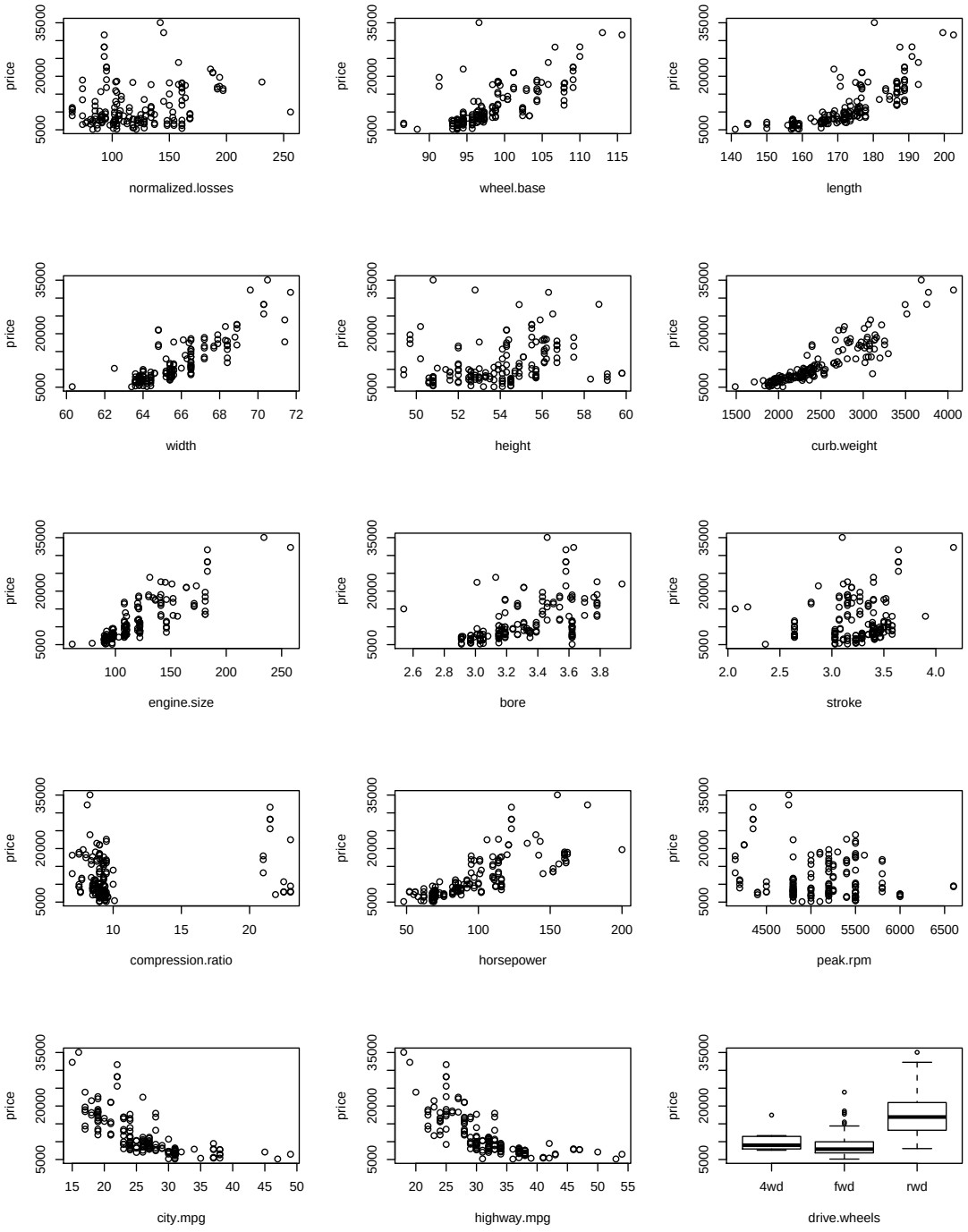


Figure 1: Response variable 'price' versus regressors.

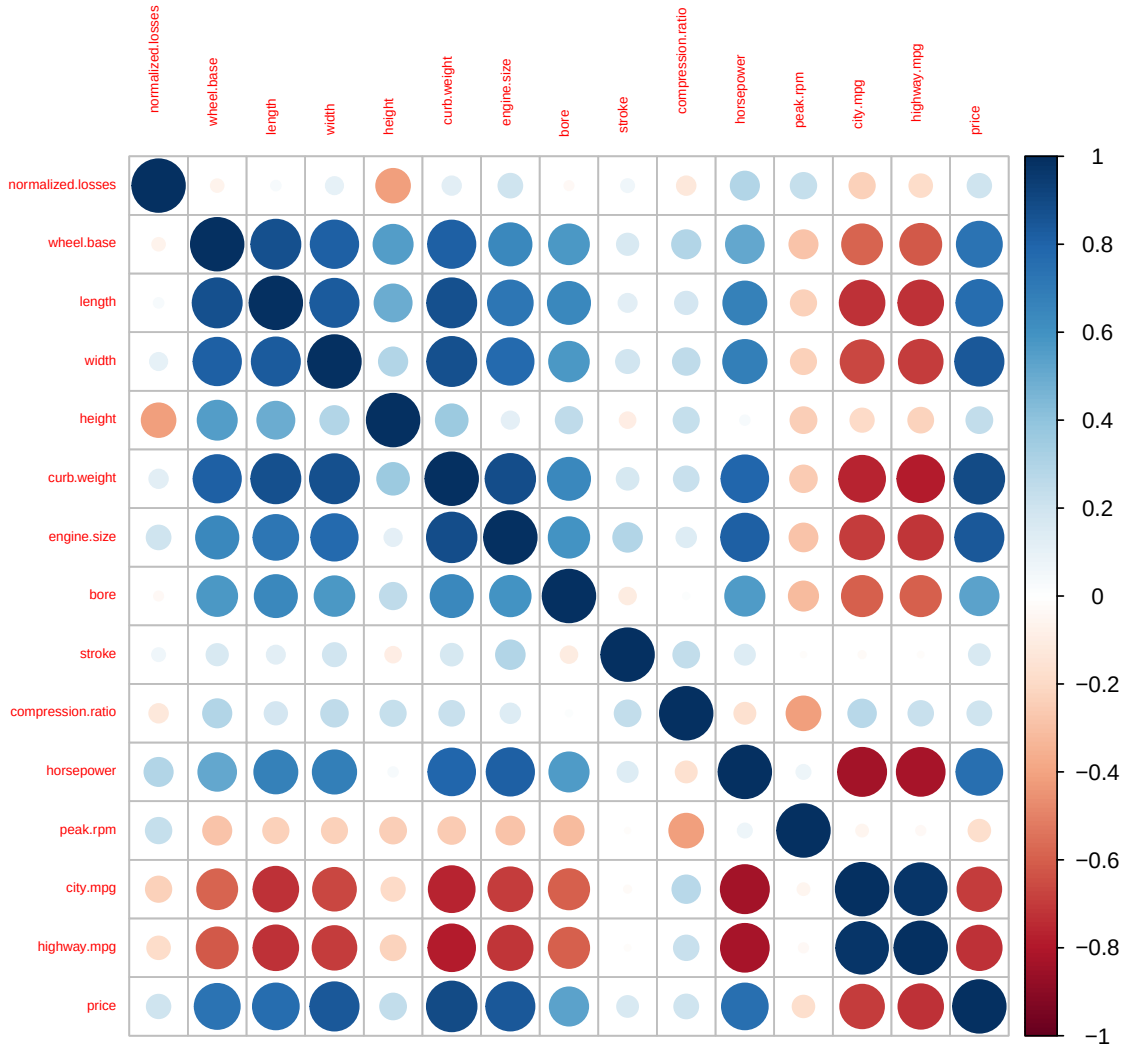


Figure 2: Correlation matrix plot.

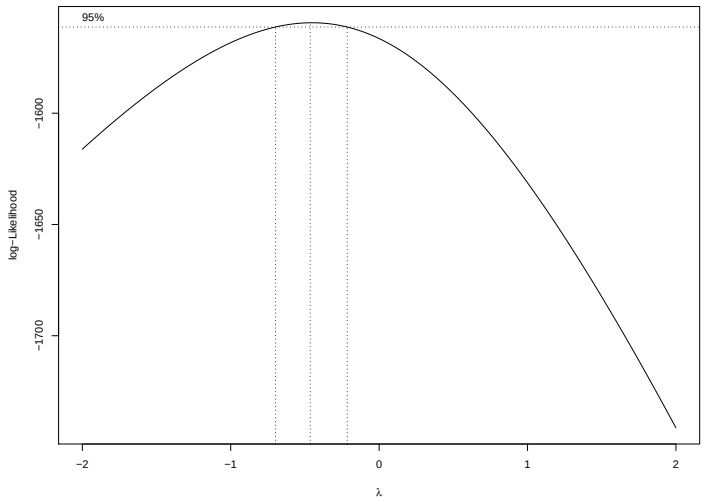


Figure 3: Maximum likelihood of BoxCox transformation: $\lambda \approx -0.45$.

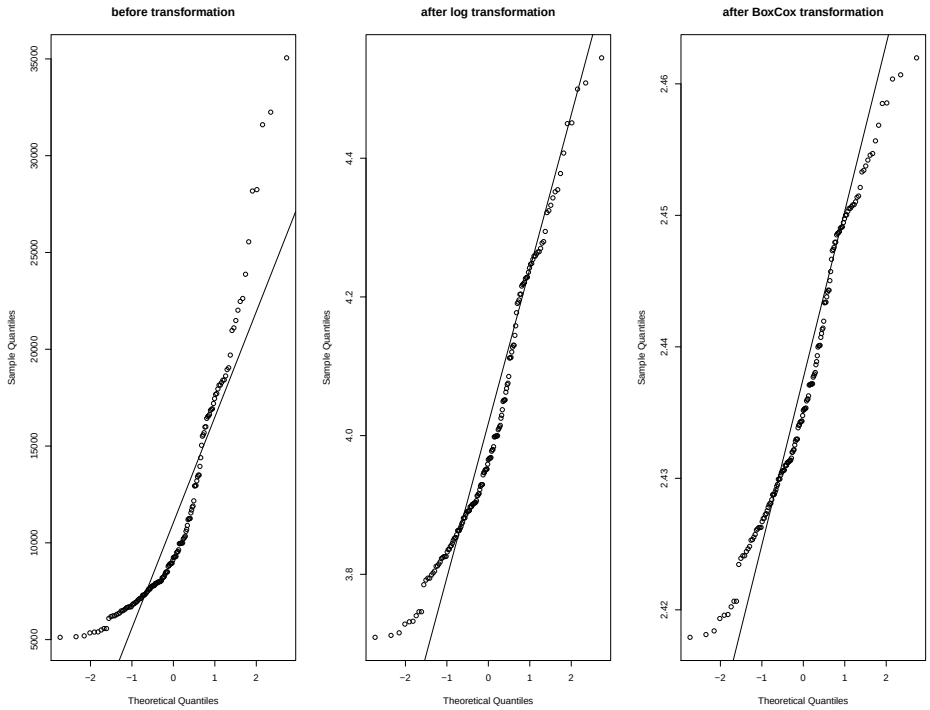


Figure 4: Transformation of the price improves normality.

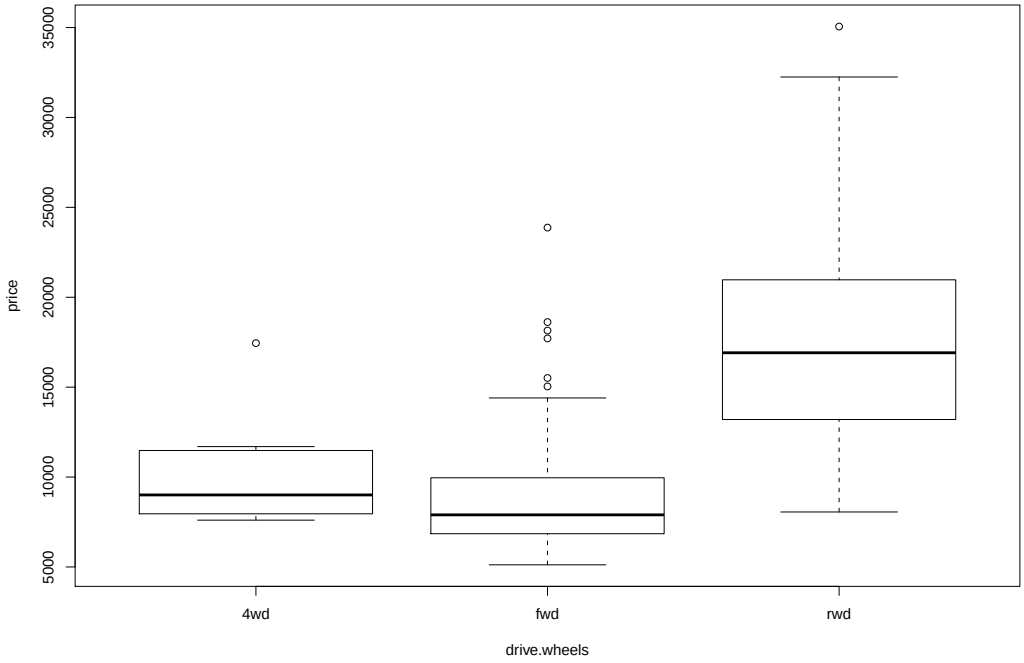
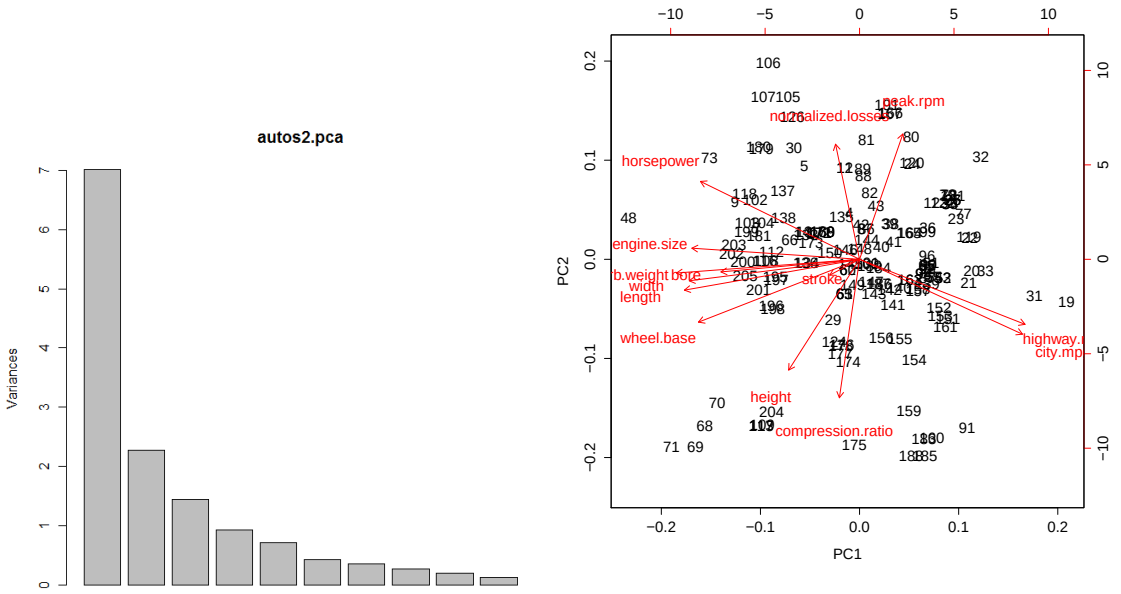
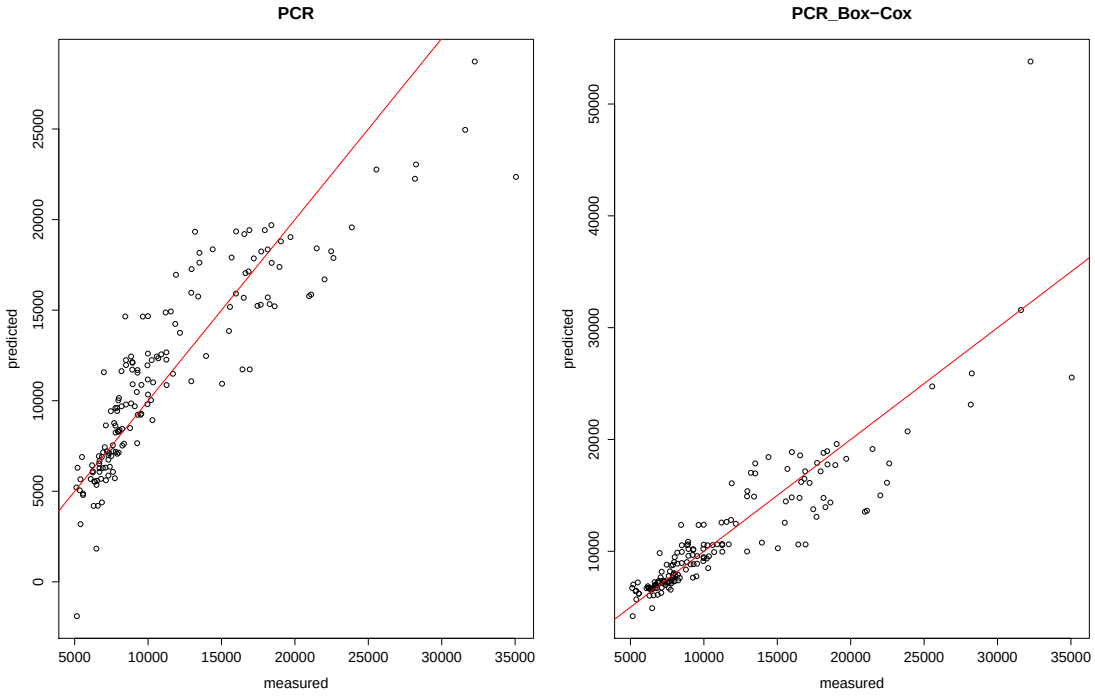


Figure 5: Boxplot of drive-wheels versus price—before and after BoxCox transformation.



(a) Screeplot **(b)** Biplot

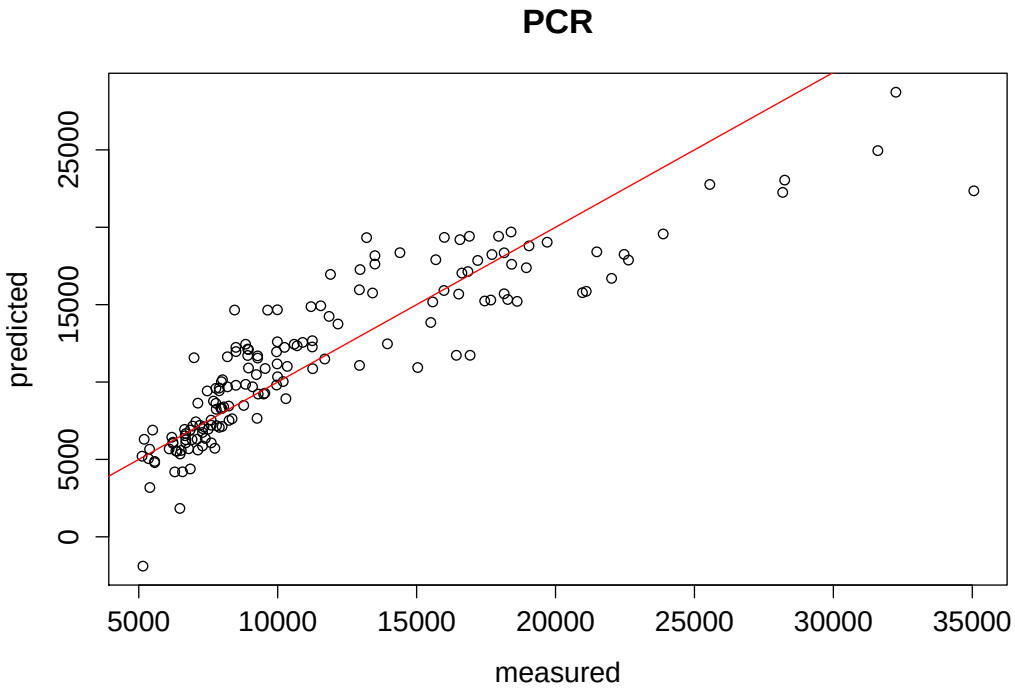
Figure 6: PCA.



(a) PCR — normal

(b) PCR — Box-Cox

Figure 7: PCR



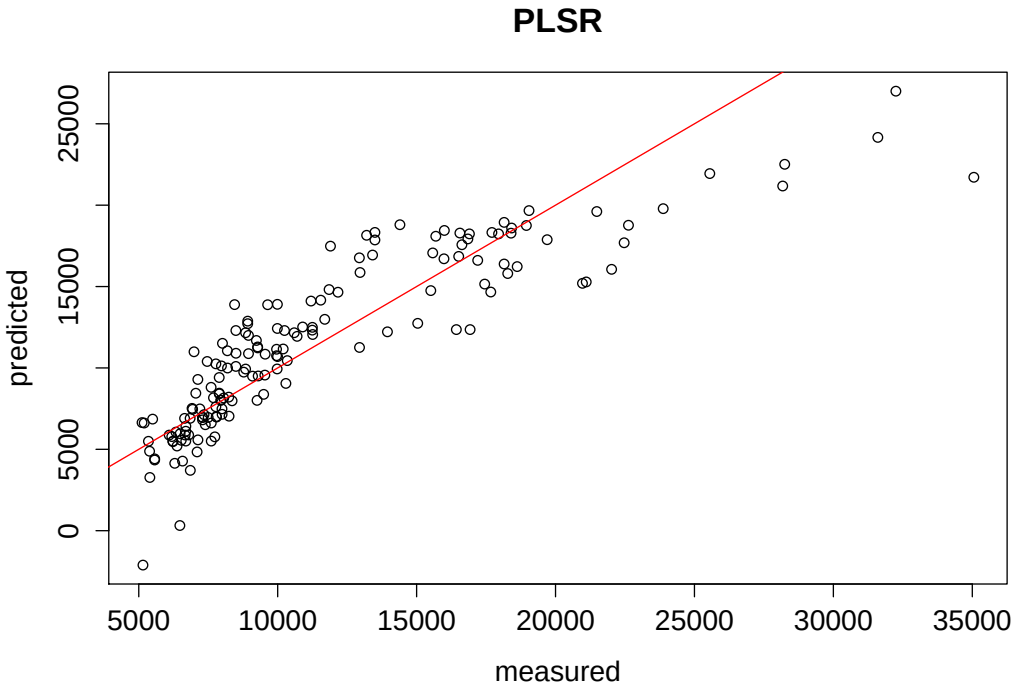
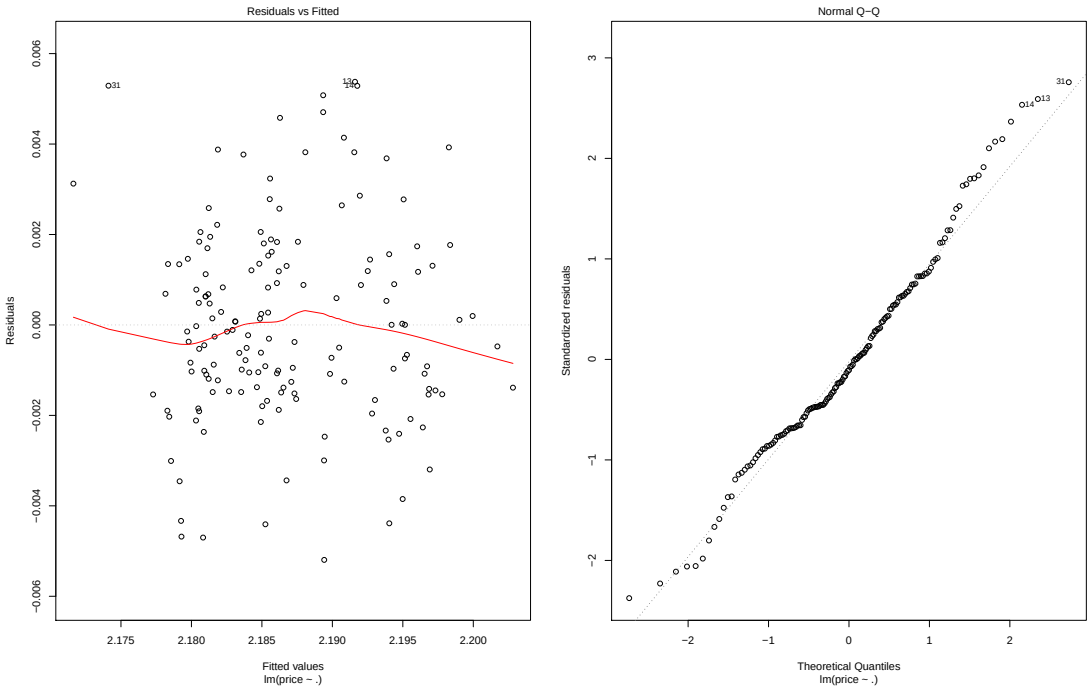
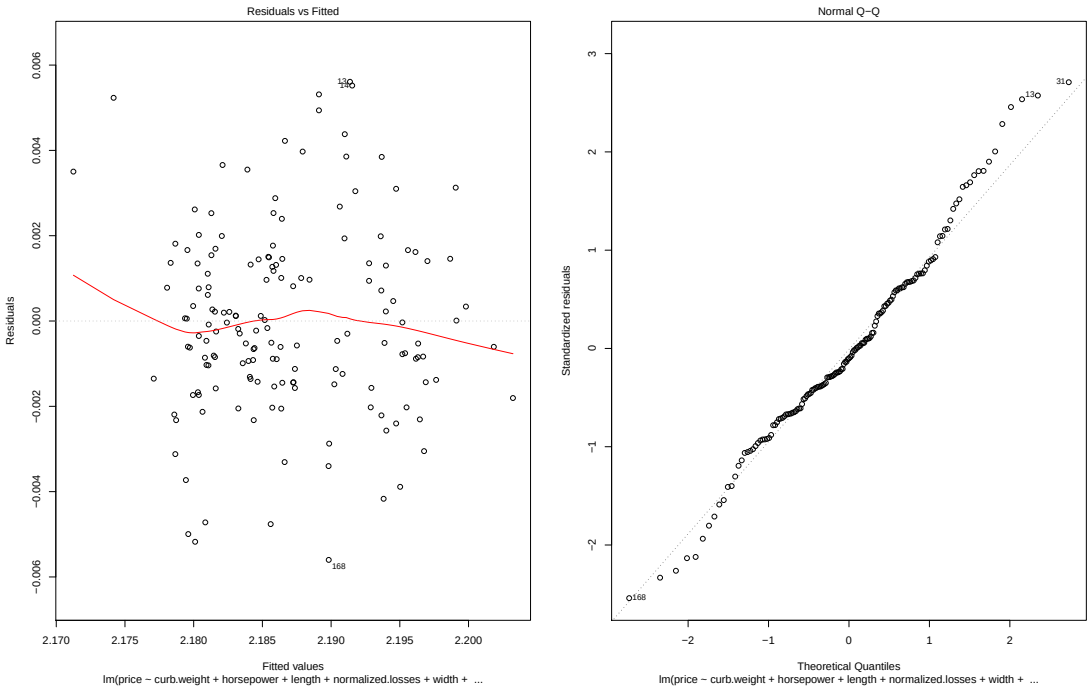


Figure 8: PCR and PLS.



(a) Residuals versus fitted **(b)** Normal Q-Q plot of standardized residuals

Figure 9: Diagnostic plots for linear model using all variables as regressors.



(a) Residuals versus fitted **(b)** Normal Q-Q plot of standardized residuals

Figure 10: Diagnostic plots for linear model obtained by stepwise regression (AIC).

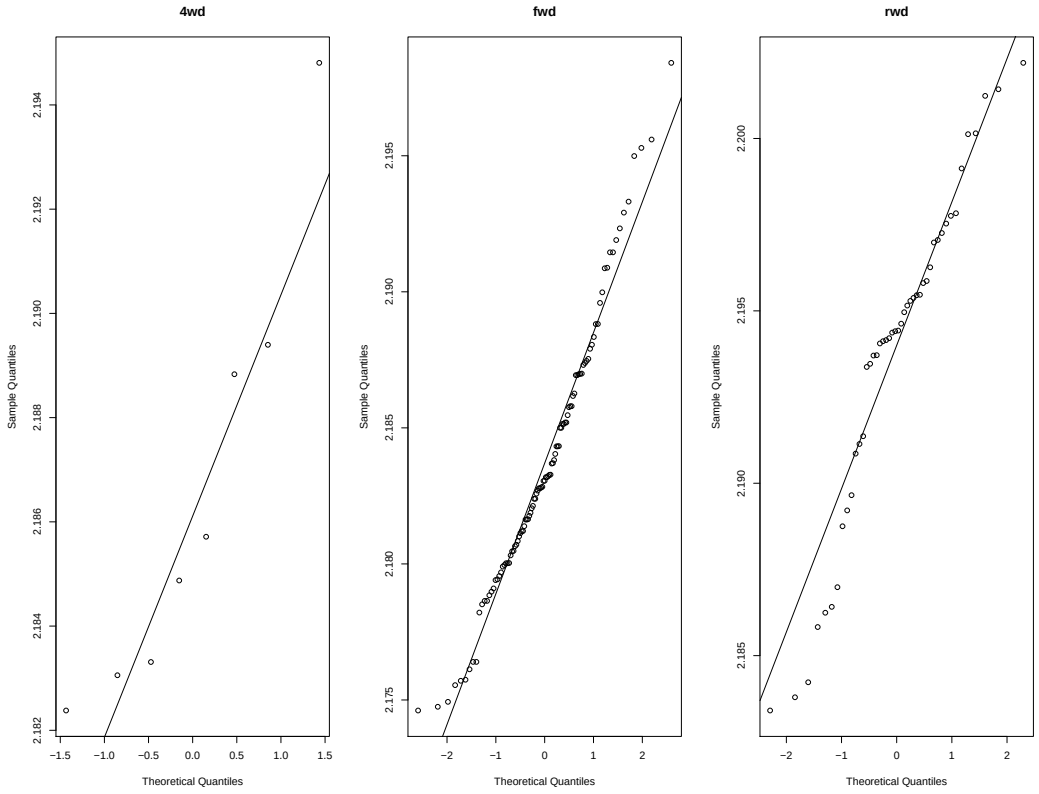


Figure 11: Normal Q-Q plots of drive-wheels.

B. R-code

Used libraries:

- MASS
- car
- rggobi
- corrplot
- pls
- FitAR

```
# Andreas Put & Li Quan
# Homework 2: Statistische Modellen & Data-analyse
# May 23, 2011
##############################################################################
library(MASS)
library(car)

autos<-read.table("autos.txt",header=T,sep=",",na.strings="?")
autos$symboling <- factor(autos$symboling,ordered=T)
autos$num.of.doors <- factor(autos$num.of.doors,ordered=T,levels=c("two","four"))
autos$num.of.cylinders <- factor(autos$num.of.cylinders,ordered=T,levels=c("two","
    three","four","five","six","eight","twelve"))

#exploratory analysis
#library(rggobi)## for interactive data exploration with ggobi, type ggobi(autos)
summary(autos)
autos<-na.omit(autos) #just remove the instances with some missing attribute value
summary(autos)

autosNum <- autos[sapply(autos,is.numeric)] #consider only numerical attributes
summary(autosNum)

library(corrplot)
autosNum.cor <- cor(autosNum)
corrplot::corrplot(autosNum.cor) #avoids collision with corrplot of pls library

autosWorking <- autosNum          #this will be our main working dataset
autosWorking$drive.wheels <- autos$drive.wheels
par(mfrow=c(5,3))
plot(price~.,autosWorking)
par(mfrow=c(1,1))

#data transformation is needed for price
shapiro.test(autosWorking$price)
#we can perform logarithmic transformation on price
autosLog <- autosWorking
autosLog$price <- log10(autosWorking$price)
#lets use the more general boxcox transformation for better results
library(FitAR)
autosWorking.lm <- lm(price~.,autosWorking); summary(autosWorking.lm)
boxcox(autosWorking.lm)
lambda <- -0.45
autosWorking$price <- bxcx(autosWorking$price, lambda, InverseQ = F, type = "
    BoxCox")
shapiro.test(autosWorking$price)

par(mfrow=c(1,3))
qqnorm(autos$price,main="before transformation")
qqline(autos$price)
```

```
qqnorm(autosLog$price,main="after log transformation")
qqline(autosLog$price)
qqnorm(autosWorking$price,main="after BoxCox transformation")
qqline(autosWorking$price)

autosNum$price <- autosWorking$price

#now make again a plot of response variable versus regressors
par(mfrow=c(5,3))
plot(price~.,autosWorking)
par(mfrow=c(1,1))

par(mfrow=c(1,2))
boxplot(price~drive.wheels,autos)
boxplot(price~drive.wheels,autosWorking)
par(mfrow=c(1,1))

#PCA
library(pls)
autosNum.pca <- prcomp(autosNum[1:14],scale=T,center=T)
plot(autosNum.pca)
summary(autosNum.pca)
biplot(autosNum.pca)
#qqnorm(autosNum.pca$x[,1]);qqline(autosNum.pca$x[,1])
#scatterplot(autosNum.pca$x[,1], autosNum.pca$x[,2])
npc <- 4

#PCR
autosNum.pcr <- lm(autos$price~autosNum.pca$x[,1]+autosNum.pca$x[,2]+autosNum.pca$
    x[,3]+autosNum.pca$x[,4])
summary(autosNum.pcr)
plot(autos$price,autosNum.pcr$fitted.values,main="PCR",xlab = 'measured',ylab = '
    predicted')
lines(c(1:40000),c(1:40000),col='red')

#PCR With Box-Cox
boxcox(autos$price~autosNum.pca$x[,1]+autosNum.pca$x[,2]+autosNum.pca$x[,3]+
    autosNum.pca$x[,4])
lambda2 <- -0.4
autosNum.bcprice <- bxcx(autos$price,lambda2,InverseQ=F,type="BoxCox")
shapiro.test(autosNum.bcprice)
autosNum.bcpcr <- lm(autosNum.bcprice~autosNum.pca$x[,1]+autosNum.pca$x[,2]+
    autosNum.pca$x[,3]+autosNum.pca$x[,4])
summary(autosNum.bcpcr)
autosNum.rbcprice <- bxcx(autosNum.bcpcr$fitted.values,lambda2,InverseQ=T,type="
    BoxCox")
plot(autos$price,autosNum.rbcprice,main="PCR_Box-Cox",xlab='measured',ylab='
    predicted')
lines(c(1:40000),c(1:40000),col='red')

#PLSR
par(mfrow=c(2,1))
autosNum.plsr <- plsr(price~.-price,data=autosNum,ncomp=npc,validation="CV",scale=
    T)
summary(autosNum.plsr)
plot(autos$price,autosNum.pcr$fitted.values,main="PCR",xlab ='measured',ylab='
    predicted')
lines(c(1:40000),c(1:40000),col='red')

plot(autos$price,autosNum.plsr$fitted.values[,1,1],main="PLSR",xlab='measured',
    ylab='predicted')
lines(c(1:40000),c(1:40000),col='red')
```

```
par(mfrow=c(1,1))

# linear model using all attributes
autosWorking.lm <- lm(price~.,autosWorking)
summary(autosWorking.lm)

#some diagnostic plots
plot(autosWorking.lm)

#variable selection by AIC in stepwise algorithm
autosWorking.aic <- stepAIC(lm(price~1,autosWorking),
                        list(upper=~normalized.losses+wheel.base+length+width+
                            height+
                                curb.weight+engine.size+bore+stroke+compression.
                                    ratio+
                                horsepower+peak.rpm+city.mpg+highway.mpg+drive.
                                    wheels,
                             lower=~1),
                        direction="both")
autosWorking.lm2 <- lm(price ~ curb.weight + horsepower + length + normalized.
    losses +
                        width + compression.ratio + city.mpg + drive.wheels,
                       data=autosWorking)
summary(autosWorking.lm2)
plot(autosWorking.lm2)

#aov
par(mfrow=c(1,3))
autosWorking.4wd <- autosWorking$price[autosWorking$drive.wheels == "4wd"]
qqnorm(autosWorking.4wd,main="4wd")
qqline(autosWorking.4wd)
shapiro.test(autosWorking.4wd)

autosWorking.fwd <- autosWorking$price[autosWorking$drive.wheels == "fwd"]
qqnorm(autosWorking.fwd,main="fwd")
qqline(autosWorking.fwd)
shapiro.test(autosWorking.fwd)

autosWorking.rwd <- autosWorking$price[autosWorking$drive.wheels == "rwd"]
qqnorm(autosWorking.rwd,main="rwd")
qqline(autosWorking.rwd)
shapiro.test(autosWorking.rwd)
par(mfrow=c(1,1))

leveneTest(price~drive.wheels, data=autosWorking)
autosWorking.aov <- aov(price ~ drive.wheels, autosWorking)
summary(autosWorking.aov)

autosWorking.HSD <- TukeyHSD(autosWorking.aov); autosWorking.HSD
```

References

[1] E. Acuñ and C. Rodriguez. The treatment of missing values and its effect in the classifier accuracy. *Classification, Clustering and Data Mining Applications*, pages 639–648, 2004.

[2] J. de Leeuw and P. Mair. Gifi methods for optimal scaling in R: The package homals. *Journal of Statistical Software*, 31(4):1–21, 2009.

[3] D. Kibler, D. W. Aha, and M. K. Albert. Instance-based prediction of real-valued attributes. *Computational Intelligence*, 5(2):51–57, 1989.