

### QQI

### HIGHER DIPLOMA IN SCIENCE IN DATA ANALYTICS

### Solution For FINAL EXAMINATIONS

*Module Code:* **B8IT109**

*Module Description:* **Advanced Data Analytics**

*Examiner:* **Dr Shahram Azizi Sazi**

**INSTRUCTIONS TO CANDIDATES**

* *This is an open book- material exam; students are allowed to use lecture notes, code, and different websites to respond the questions.*
* *Please select four questions out of five questions. Explicitly specify your selected questions on the top of exam paper.*
* *R code and necessary outputs (i.e. graphs/plots/curves) need to be saved in word format and submit to Moodle.*

### Question 1

Use **mtcars** dataset and consider **mpg** and **vs** as the attributes of interest.

1. Use the appropriate probability models to quantify the uncertainty in mpg and vs.

**(5 Marks)**

1. Estimate the parameters of your proposed models using the dataset.

**(5 Marks)**

1. Predict the future values of mpg and vs using (a) and (b).

**(10 Marks)**

1. Using (a), (b), find P(mpg > 90). **(5 Marks)**

**(TOTAL: 25 Marks)**

Solution:

X=mtcars$mpg

# model x using N(mu, sigma), mu and sigma are parameters

mu=mean(X); sigma=sd(X)

# prediction

D=rnorm(1000, mu, sigma)

#################

pred=mean(D)

pred

[1] 19.92489

###########################################################################

y=mtcars$vs

p=sum(y)/length(y)

p

[1] 0.4375

D=rbinom(1000,1,p)

table(D)

D

0 1

544 456

pred=0

#############################################################

prob=1-pnorm(10,mu,sigma)

prob

[1] 0.9529594

**Question 2**

In regression analysis, the **Boston** dataset is analysed in R and its output is as follows.

Call:

lm(formula = medv ~ crim + zn + indus + chas + nox + rm, data = Boston)

Residuals:

Min 1Q Median 3Q Max

-21.016 -3.420 -0.684 2.506 39.467

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -17.95464 3.21376 -5.587 3.81e-08 \*\*\*

crim -0.17691 0.03459 -5.114 4.50e-07 \*\*\*

zn 0.02128 0.01385 1.537 0.1249

indus -0.14365 0.06394 -2.247 0.0251 \*

chas 4.78468 1.05909 4.518 7.81e-06 \*\*\*

nox -7.18489 3.69353 -1.945 0.0523 .

rm 7.34159 0.41720 17.597 < 2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.943 on 499 degrees of freedom

Multiple R-squared: 0.5874, Adjusted R-squared: 0.5824

F-statistic: 118.4 on 6 and 499 DF, p-value: < 2.2e-16

1. Using this output, specify the response and independent variables. **(5 Marks)**
2. Based on the output, which type of GLM is proposed for this analysis.

**(5 Marks)**

1. List the assumptions for your proposed regression model. **(5 Marks)**
2. Specify the significant independent variables on the response variable at the level of 𝛼 = 0.05. **(5 Marks)**
3. Using the output, find the optimal predictive model for the response variable.

**(5 Marks)**

**Solution:**

Response variable: **medv**

Independent variables: **crim , zn , indus , chas , nox , rm**

1. Multiple linear regression
2. The error terms (i.e. ) needs to be **Normally distributed**, with the **mean zero** and **constant variance**. Also**, are independent**.
3. Variables **crim, indus , chas , rm**, are significant because the corresponding p-value for these variables is less than 𝛼 = 0.05.

### Question 3

Loading the package **‘datasets’,** use the dataset **‘readingSkills’** and

consider **nativeSpeaker** as the output variable.

1. Split the dataset into 80% as the train-set and 20% as the test-set. (use set.seed(104))

**(2.5 Marks)**

1. Apply Random Forest (RF) algorithm to train the classifier using train-set with 20 trees. **(5 Marks)**
2. Predict the test-set using the trained model of classifier. **(2.5 Marks)**
3. Provide the confusion matrix and obtain the accuracy. **(5 Marks)**

1. Redo parts b-d to apply either Naïve Bayes or Decision Tree. Which model does provide the higher accuracy? **(10 Marks)**

**(TOTAL: 25 Marks)**

Solution:

1. Please see R code.
2. Please see R code.
3. Please see R code.
4. Confusion matrix

pred no yes

no 21 0

yes 0 19

> acc\_rf

[1] 1

R code for this question part (a)-(d):

set.seed(104)

n=nrow(dataset)

indexes = sample(n,n\*(80/100))

trainset = dataset[indexes,]

testset = dataset[-indexes,]

rf <- randomForest(nativeSpeaker ~., data = readingSkills, ntree=20)

pred= predict(rf, testset)

tab=table(pred,testset[,1]) # confusion matrix

acc\_rf=sum(tab[row(tab)==col(tab)])/sum(tab)##################################

R code for this question part (e)

dataset=readingSkills

vect=rep(0,2)

for(i in 1:1000) {

n=nrow(dataset)

indexes = sample(n,n\*(80/100))

trainset = dataset[indexes,]

testset = dataset[-indexes,]

rf <- randomForest(nativeSpeaker ~., data = readingSkills, ntree=20)

pred= predict(rf, testset)

tab=table(pred,testset[,1]) # confusion matrix

acc\_rf=sum(tab[row(tab)==col(tab)])/sum(tab)

# NB

nb <- naiveBayes(nativeSpeaker ~., data = readingSkills)

pred\_nb= predict(nb, testset)

tab\_nb=table(pred\_nb,testset[,1]) # confusion matrix

acc\_nb=sum(tab\_nb[row(tab\_nb)==col(tab\_nb)])/sum(tab\_nb)

accu\_vect=c(acc\_rf,acc\_nb)

vect=vect+(1/1000)\*accu\_vect

}

vect

[1] 0.999375 0.623525

Accuracy of random forest is higher.

### Question 4

Use **data('EuStockMarkets')** to load the in-built dataset 'EuStockMarkets' in R, consider ***DAX*** as your time series variable:

1. Validate the assumptions using graphical visualization.

**(5 Marks)**

***Answer***: ***wide sense stationary and normality should be checked.***

***data()***

***data('EuStockMarkets')***

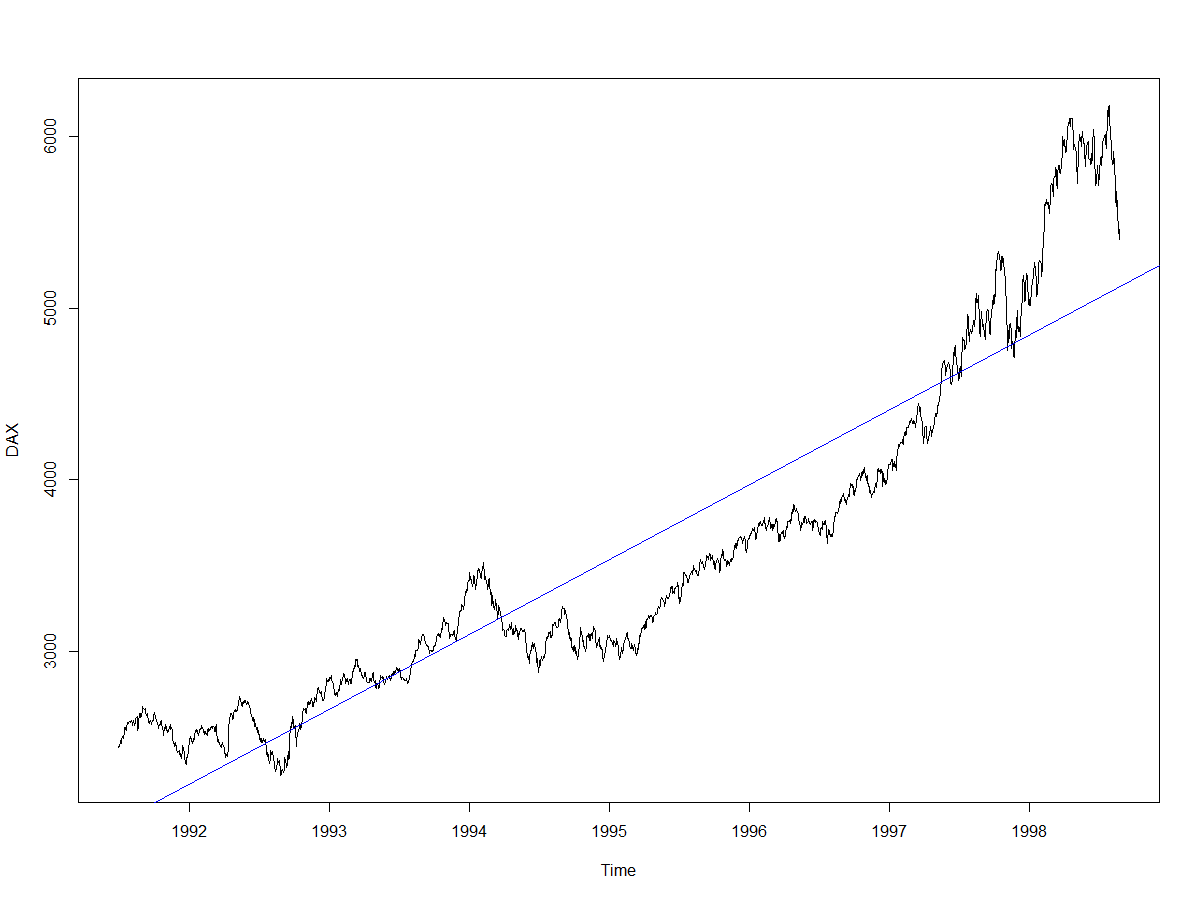
***DAX=EuStockMarkets[,4]***

***plot(DAX) # time series***

***abline(reg=lm(DAX~time(DAX)), col="blue") # line for the mean of time series***

***qqnorm(DAX)***

***From the plot of time series data, there is a clear increasing trend which shows the non-stationarity of wage.***

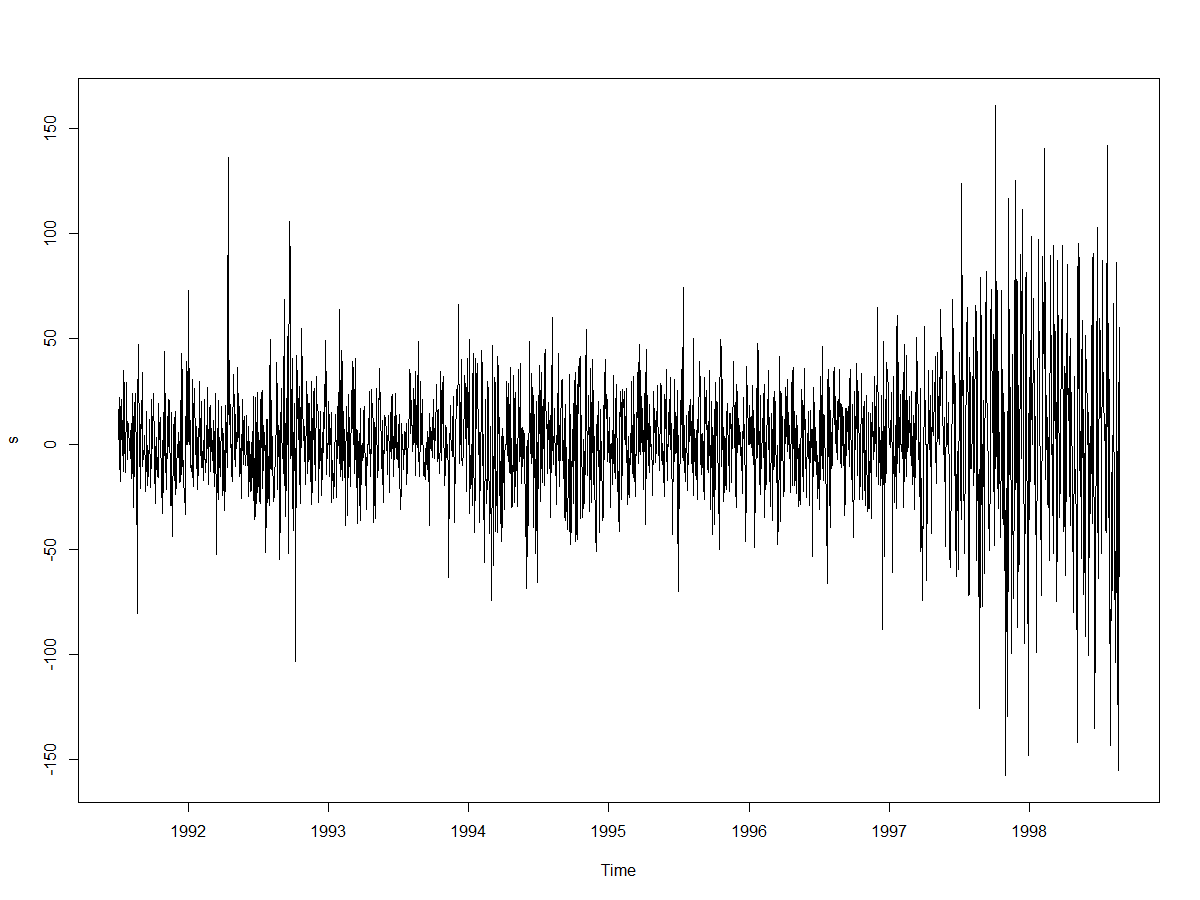
******

***In order to make ts stationary in mean, we use diff() and to make the ts stationary in variance we can use log() function.***

***s=diff(DAX)***

***plot(s) # time series***

***Applying diff() leads to the following plot:***

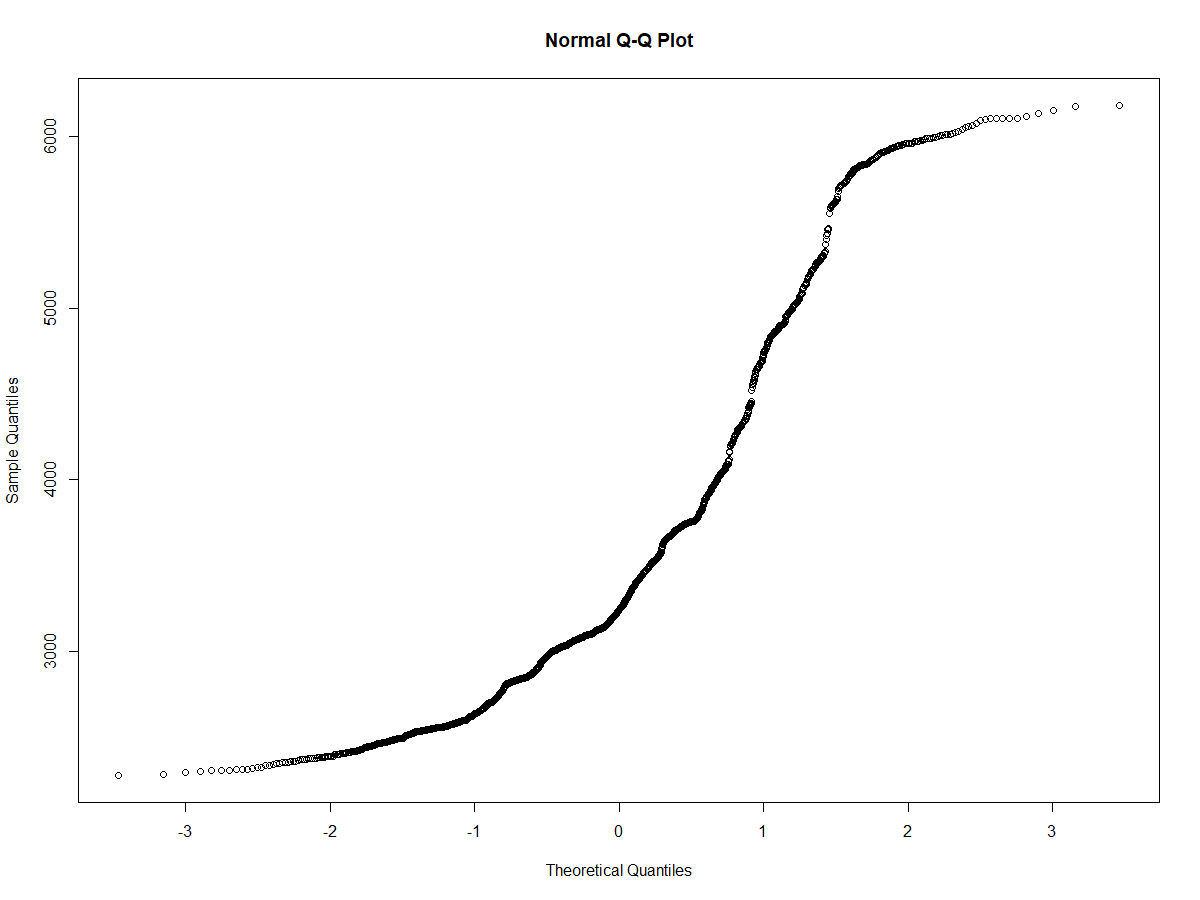
******

***The above plot shows stationary in mean but not stationary in variance, therefore we need to take log() as follows:***

***ls=diff(log(DAX))***

***plot(ls)***

***Also using the QQ-plot, wage does not seem to be normal.***

******

***To solve these issues, we use auto.arima model. Please see R code.***

1. Fit the optimized ARIMA model for ***DAX*** and provide the coefficient estimates for the fitted model. **(10 Marks)**

***Answer:*** *we need to apply auto.arima with both seasonal=T and F*

***auto.fit<-auto.arima(DAX, seasonal=T)***

***auto.fit***

Series: training

ARIMA(1,1,0)(1,1,0)[12]

Coefficients:

ar1 sar1

-0.2434 -0.2127

s.e. 0.1046 0.1072

sigma^2 estimated as 91.28: log likelihood=-319.1

AIC=644.2 AICc=644.49 BIC=651.6

***auto.fit<-auto.arima(DAX, seasonal=F)***

***auto.fit***

Series: DAX

ARIMA(0,1,1) with drift

Coefficients:

ma1 drift

0.1276 1.6206

s.e. 0.0232 0.7944

sigma^2 estimated as 923.8: log likelihood=-8983.91

AIC=17973.83 AICc=17973.84 BIC=17990.41

***Based the output,*** *We can see that AIC for seasonal ts is much lower than that of non-seasonal ts. Therefore, the estimation of coefficients based on seasonal auto.arima is*

Coefficients:

ar1 sar1

-0.2434 -0.2127

s.e. 0.1046 0.1072

1. What is the estimated order for AR and MA?

**(5 Marks)**

***Answer:*** *using the above output (*ARIMA(1,1,0)(1,1,0)[12] *), the order estimation for non-seasonal term is p=1, d=1, q=0 and for seasonal part*

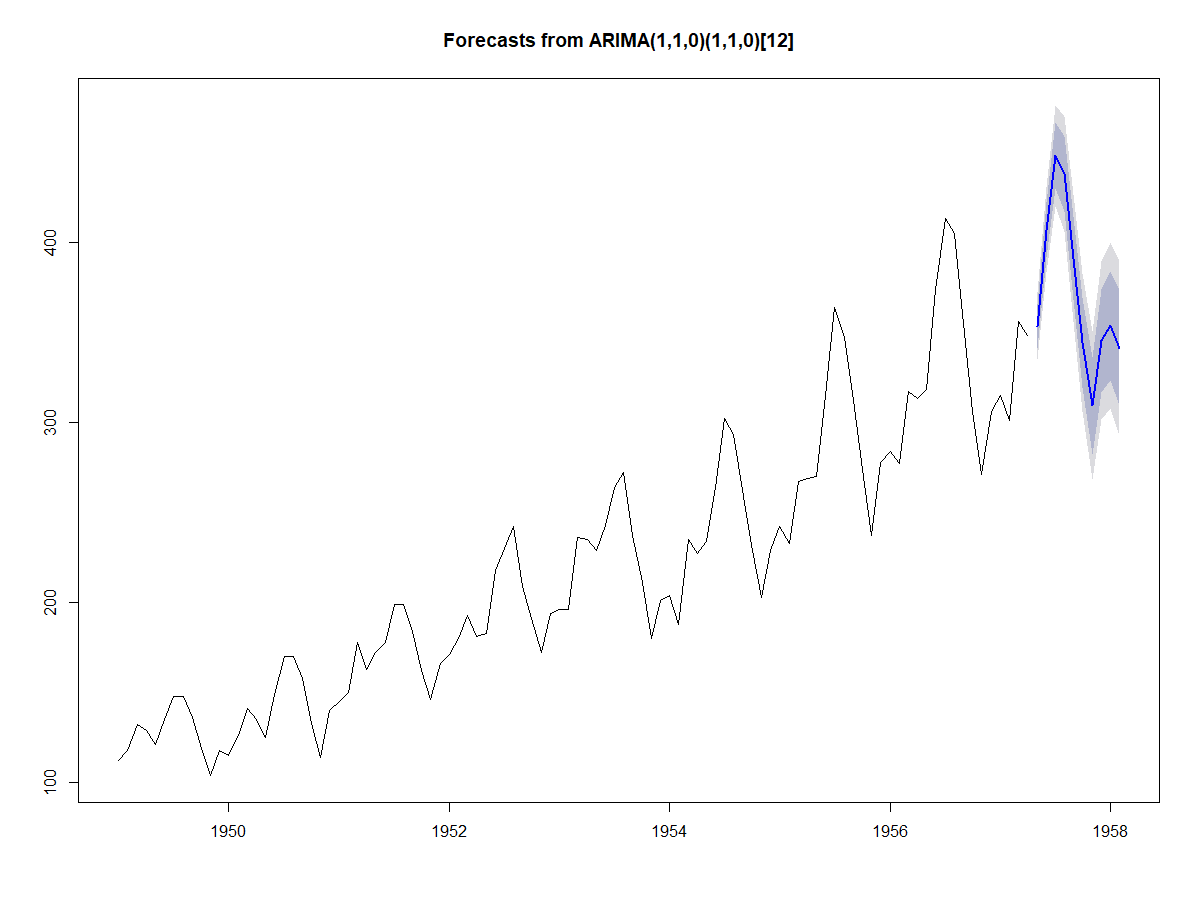
*are p=1, d=1, q=0.*

1. Forecast h=10 step ahead prediction of ***DAX*** on the plot of the original time series.

***Answer:***

*auto.fcast <- forecast(auto.fit, h=10)*

*plot(auto.fcast)*



**(5 Marks)**

**(Total: 25 Marks)**

### Question 5 : *multivariate analysis and unsupervised learning methods*

Use dataset available on <http://users.stat.ufl.edu/~winner/data/hybrid_reg.csv>

* 1. Use LDA to classify the dataset into few classes so that at least 85% of information of dataset is explained through new classification. (**Hint**: model the output variable “**carclass\_id”** to input variables “**msrp**”, “**accelrate**”, and “**mpg**”). How many LDs do you choose? Explain the reason. **(10 Marks)**
  2. Apply PCA to input variables, and identify the important principle components involving at least 90% of dataset variation. Explain your decision strategy? Plot principle components versus their variance (**Hint**: to sketch the plot use the Scree plot). **(5 Marks)**
  3. Use K-means clustering analysis to input variables and identify the most important classes. How many classes do you select? Why?

**(5 Marks)**

* 1. Split the dataset into two sets of variables so that **X**=( msrp, mpgmpge) and **Y**=( accelrate, mpg). Apply canonical correlation analysis to find the cross-correlation between **X** and **Y**. What is the correlation between ***msrp*** and ***mpg***?

**(5 Marks)**

**(Total: 25 Marks)**

Solution:

1. *using the output of lda,*

Coefficients of linear discriminants:

LD1 LD2 LD3

msrp 0.0000109007 5.761851e-05 4.799366e-05

accelrate 0.0399827796 5.046553e-02 -5.162886e-01

mpg -0.1150164945 8.534612e-02 -2.397826e-02

Proportion of trace:

LD1 LD2 LD3

0.6945 0.2002 0.1053

*LD1 and LD2 explain at least 85% of information of the dataset.*

1. *using the output of pca,*

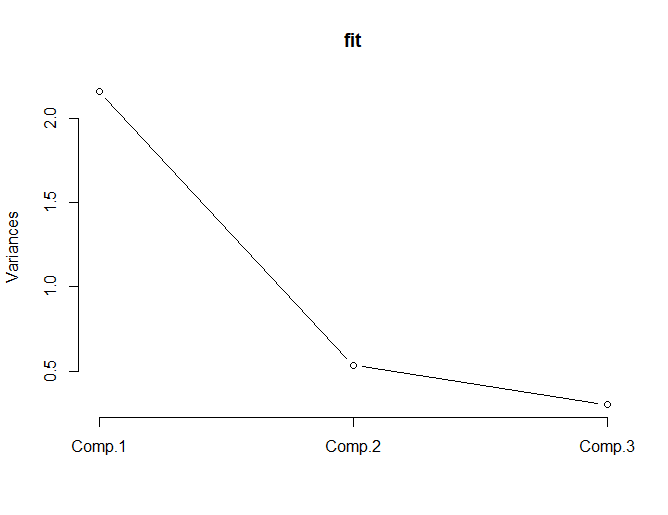
Importance of components:

Comp.1 Comp.2 Comp.3

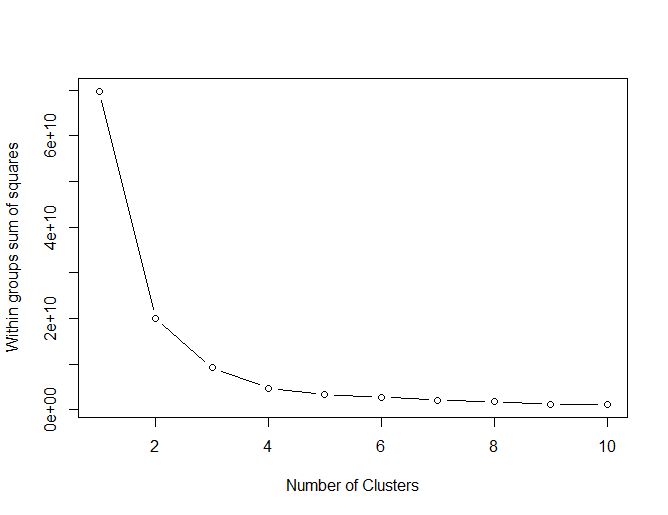
Standard deviation 1.4696872 0.7325764 0.5507734

Proportion of Variance 0.7199935 0.1788894 0.1011171

Cumulative Proportion 0.7199935 0.8988829 1.0000000

*All three components should be selected at level 90%. *

1. ***Answer:*** *please see R code.*
2. ***Answer:*** *using WSS plot, the elbow is on the fourth cluster. Therefore, the first four clusters are selected based on K-means analysis.*

**

R code:

# part (a)

filename="http://users.stat.ufl.edu/~winner/data/hybrid\_reg.csv"

data<-read.csv(filename,header=TRUE)

head(data)

library(MASS)

dataset.lda <- lda(carclass\_id~msrp+accelrate+mpg, data)

dataset.lda

##########################################################

# part (b)

dataset=cbind(data$msrp,data$accelrate,data$mpg)

fit <- princomp(dataset, cor=TRUE)

summary(fit)

loadings(fit) # pc loadings

plot(fit,type="lines") # scree plot

#########################################################

# part (c)

# K-Means

k.means.fit <- kmeans(dataset, 6) # k = 3 is the number of classes in type

attributes(k.means.fit)

# Centroids(arithmetic mean)

k.means.fit$centers

# Cluster size:

k.means.fit$size

wssplot <- function(dataset, nc=10, seed=1234){

wss <- (nrow(data)-1)\*sum(apply(dataset,2,var))

for (i in 2:nc){

set.seed(seed)

wss[i] <- sum(kmeans(dataset, centers=i)$withinss)}

plot(1:nc, wss, type="b", xlab="Number of Clusters",

ylab="Within groups sum of squares")}

wssplot(dataset, nc=10)

#################################################################

# part (d)

install.packages('CCA') # install GGally packages

library(CCA)

X <- cbind(data$msrp , data$mpg)

Y <- cbind(data$accelrate , data$mpgmpge)

cor(X,Y) # correlation between two set of variables

**End of Examination**