Multiple Linear Regression

Multicollinearity Problem and Normality of Errors

Problem of Multicollinearity

Multicollinearity exists if there is strong linear relationship among the independent variables

Multicollinearity has two serious consequences:

1. Highly Unstable Model Parameters

As standard errors of their estimates are inflated

2. Model Fails to Accurately Predict for Out of Sample Data

Therefore, it is important to check for Multicollinearity in regression analysis

Detecting Multicollinearity Through VIF

VIF (Variance Inflation Factor) Method:

Dependent Variable : Y

Independent variables : X1, X2, X3, X4

Dependent Variable	Independent Variables	R^2	$1 - R^2$ = Tolerance	**VIF = 1/(Tolerance)**
X1	X2, X3, X4			
X2	X1, X3, X4			
X3	X1, X2, X4			
X4	X1, X2, X3			

Any VIF > 5, indicates presence of Multicollinearity

Detecting Multicollinearity in R

```
#Importing the Data, Fitting Linear Model
```

```r
perindex<-read.csv("Performance Index.csv", header=TRUE)
jpimodel<-lm(jpi~aptitude+tol+technical+general, data=perindex)
```

```
#Variance Inflation Factor
#Install and load package "car".
```

```r
install.packages("car")
library(car)
```

car stands for Companion to Applied Regression and consists of several useful functions for advance regression analysis.

```r
vif(jpimodel)
```

vif() in package car calculates VIFs.

* Continuing with same dataset "Performance Index"

Detecting Multicollinearity in R

```
# Output
```

```
aptitude       tol technical   general
1.179906  1.328205  2.073907  2.024968
```

Interpretation :
All VIFs are less than 5, Multicollinearity is not present.

Multicollinearity – Remedial Measures

The problem of Multicollinearity can be solved by different approaches:

- Drop one of the independent variables, which is explained by others
- Use Principal Component Regression in case of severe Multicollinearity
- Use Ridge Regression

* Dropping a variable may not be a good idea if many VIFs are large.
Principal Component Method will be discussed in detail under Data Reduction and Segmentation

Case Study - Modelling Resale Price of Cars

Background

- A car garage has old cars for resale. They keep records for different models of cars and their specifications.

Objective

- To predict the resale price based on the information available about the engine size, horse power, weight and years of use of the cars

Available Information

- Records -26
- Independent Variables: ENGINE SIZE, HORSE POWER, WEIGHT AND YEARS
- Dependent Variable: RESALE PRICE

Data Snapshot

Dependent variable

Independent variables

MODEL	RESALE PRICE	ENGINE SIZE	HORSE POWER	WEIGHT	YEARS
Daihatsu Cuore	3870	846	32	650	2.9
Suzuki Swift 1.0 GL	4163	993	39	790	2.9
Fiat Panda Mambo L	3490	899	29	730	3.1

Observatio

Columns	Description	Type	Measurement	Possible values
MODEL	Model of the car	character	-	-
RESALE PRICE	Resale price	numeric	Euro	positive values
ENGINE SIZE	Size of the engine	numeric	cc	positive values
HORSE POWER	Power of the engine	numeric	kW	positive values
WEIGHT	Weight of the car	numeric	kg	positive values
YEARS	Number of years in use	numeric	-	positive values

Correlation Matrix

```
# Importing the Data
```

```
cardata<-read.csv(”car price data.csv", header=TRUE)
```

```
# Graphical representation of data
# Install and load package “GGally”
```

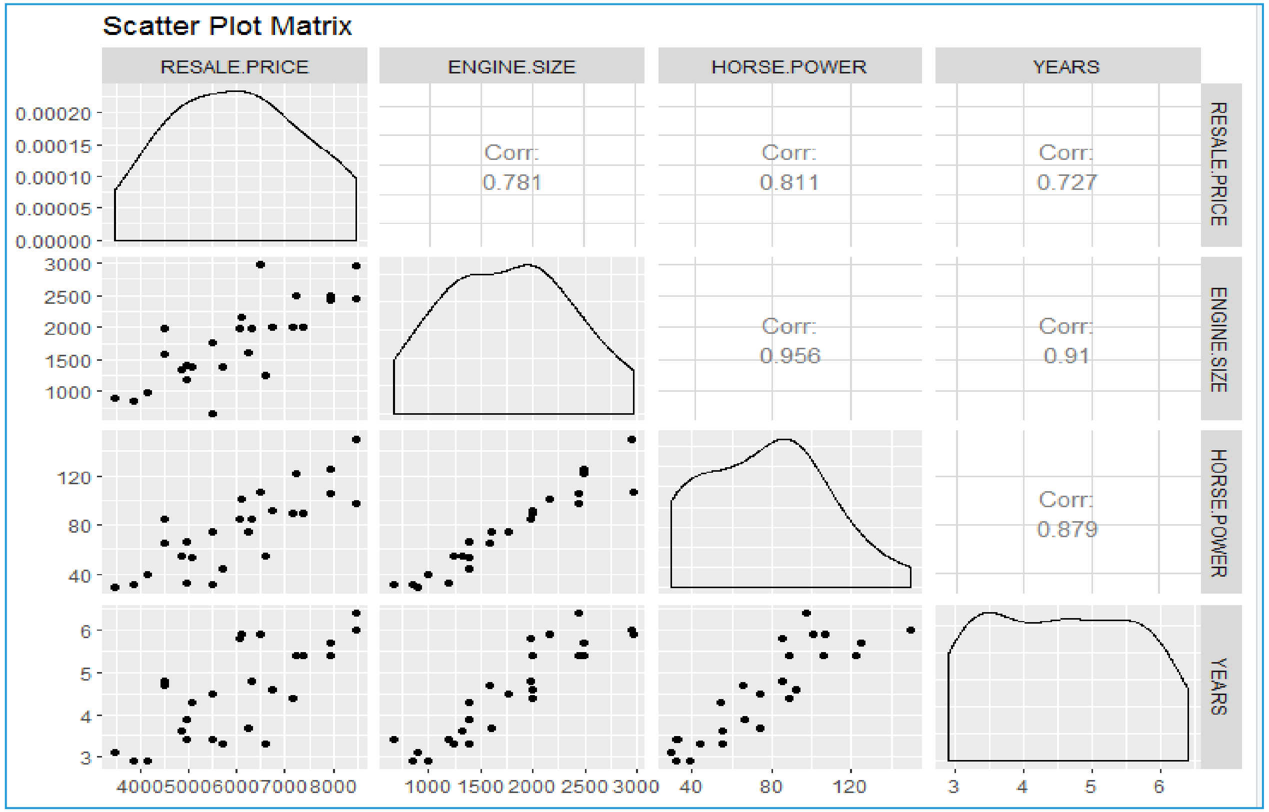
```
install.packages("GGally")
library(GGally)

ggpairs(cardata[,c("RESALE.PRICE","ENGINE.SIZE","HORSE.POWER","YEARS")
],title="Scatter Plot Matrix", columnLabels = c("RESALE.PRICE",
"ENGINE.SIZE","HORSE.POWER","YEARS"))
```

- **ggpairs()** in the package GGally is used to plot the scatter plot matrix.

Correlation Matrix

`# Output`



Interpretation :

➢ The independent variables have high positive correlation among themselves .

Detecting Multicollinearity in R

```
#Fitting Linear Model
```

```r
model<-lm(RESALE.PRICE ~ ENGINE.SIZE +HORSE.POWER +WEIGHT+ YEARS,
data=cardata)
```

```
#Variance Inflation Factor
```

```r
library(car)
vif(model)
```

```
# Output
```

```
ENGINE.SIZE HORSE.POWER      WEIGHT       YEARS
  15.759113   12.046734    9.113045   13.978640
```

Interpretation

VIF values for all the variables are greater than 5, hence we can conclude that there exist Multicollinearity between the independent variables.

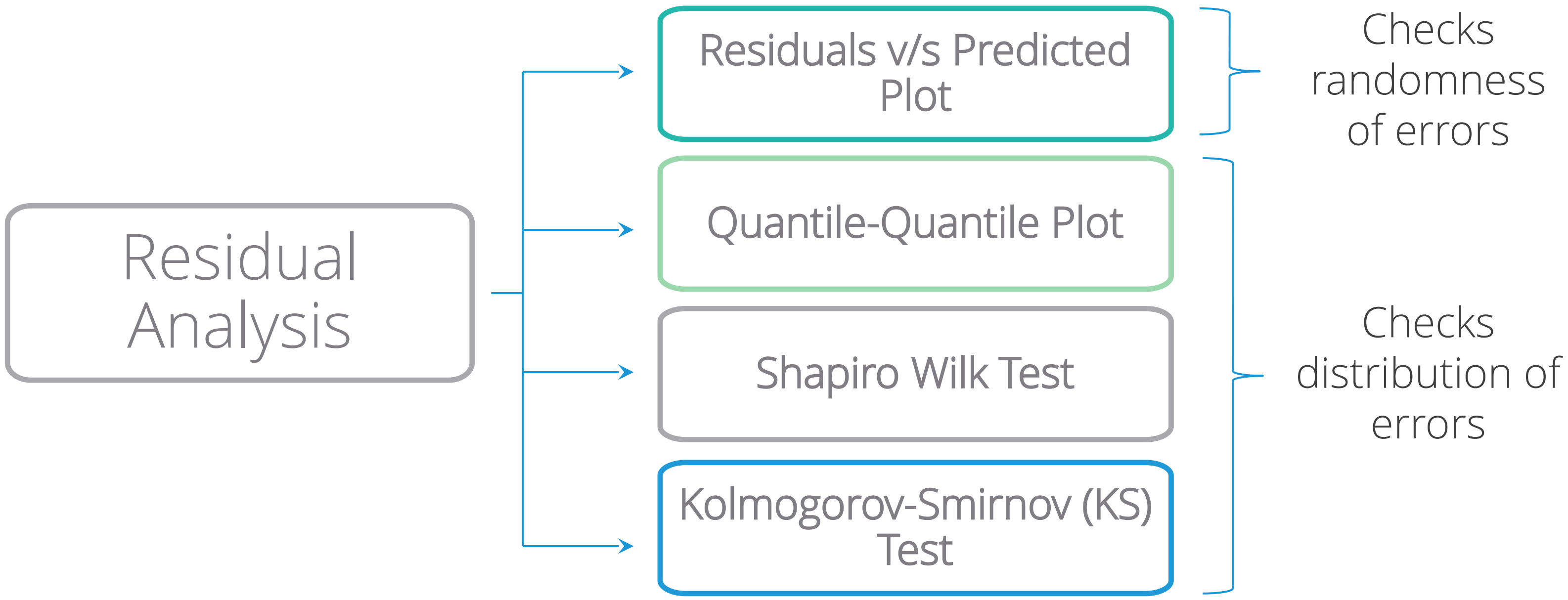
Normality of Errors

- The errors in Multiple Linear Regression are assumed to follow Normal Distribution.

- If Normality of Errors is not true then statistical tests and associated P values based on F and t distribution are not reliable.

Residual Analysis

Observed Value – Predicted value = Residual



Residual Analysis for Performance Index Data

Continuing with the "**Performance Index** " data,

- **Model** job performance index (**jpi**) based on aptitude score (**aptitude**), test of language (**tol**), technical knowledge (**technical**) and general information (**general**)
- Get the fitted values and thus the residuals.
- Analyse the distribution of residuals

Residual v/s Predicted Plot in R

```
#Importing the Data, Fitting Linear Model and Calculate Fitted Values
and Residuals
perindex<-read.csv("Performance Index.csv",header=TRUE)
jpimodel<-lm(jpi~aptitude+tol+technical+general, data=perindex)
perindex$pred<-fitted(jpimodel)
perindex$resi<-residuals(jpimodel)
```

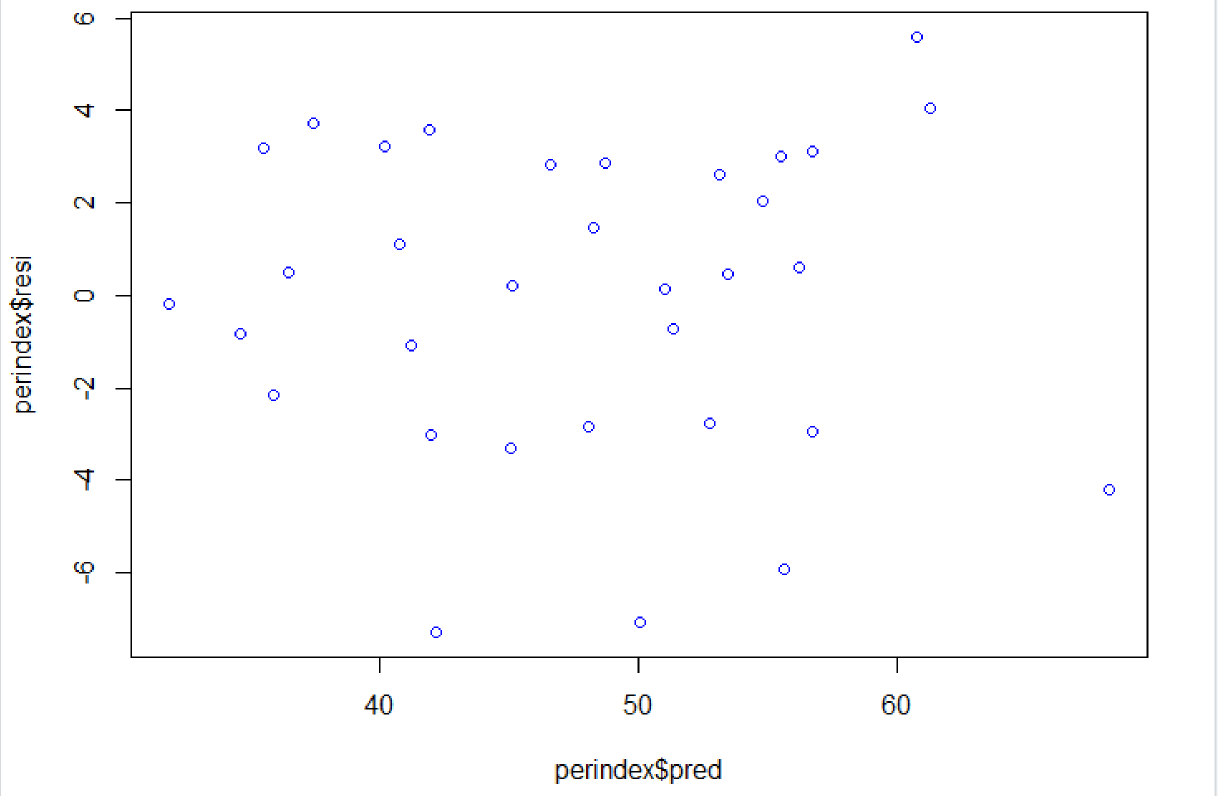
- ❑ **lm()** fits a linear regression.
- ❑ **fitted()** and **residuals()** fetch fitted values and residuals respectively.

```
#Residuals v/s Predicted Plot
plot(perindex$pred,perindex$resi,col="blue")
```

plot() is used to plot predicted values against residuals.

Residual v/s Predicted Plot in R

```
# Output
```



Interpretation:

- Residuals in our model are randomly distributed which indicates presence of Homoscedasticity

QQ Plot in R

- **The Quantile-Quantile (QQ) Plot** is a powerful graphical tool for assessing normality.
- **Quantiles are calculated using sample data and plotted against expected quantiles under Normal distribution.**

High Correlation between Sample Quantiles and Theoretical Quantiles → **Normality**

- If the data are truly sampled from a Gaussian (Normal) distribution, the **QQ plot will be linear.**

QQ Plot in R

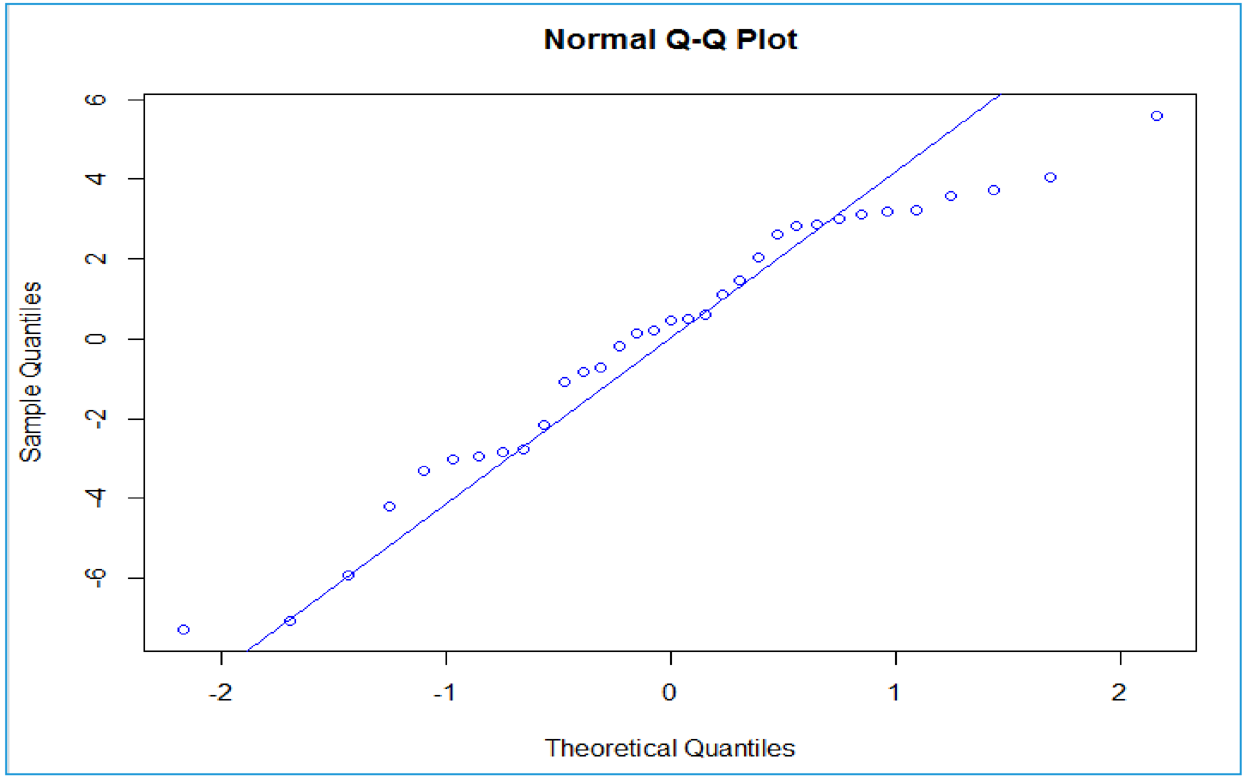
```r
#QQ Plot
qqnorm(perindex$resi,col="blue")

qqline(perindex$resi,col="blue")
```

- ❑ **qqnorm()** produces a plot with theoretical quantiles on x axis against the sample quantiles on y axis.
- ❑ Column for which normality is being tested is specified in the first argument.
- ❑ **qqline()** adds a line which passes through the first and third quartiles.

QQ Plot in R

```
# Output
```



Interpretation:

- ➢ Most of these points are close to the line except few values indicating no serious deviation from Normality.

Absence of Normality – Remedial Measure

Mathematical Transformation of the dependent variable is used as a remedial measure in case of serious departure from Normality.

Typically Log Transformation is used. However, there is general transformation called as Box Cox Transformation given as :

- Box Cox transformation

$$Y^* = \frac{Y^\lambda - 1}{\lambda} \qquad \lambda \neq 0$$

$$= \log Y \qquad \lambda = 0$$

Where Y is the response variable

- R can automatically detect the optimum λ using **`boxcox()`** in package MASS