Packages

Many of the statistical functions in R has been implemented in certain packages already. We need to install these packages and then load them to make use of them.

```
# Install package - (install once per machine)
install.packages("")

# Load package - (load once per script)
library()
```

Formula Syntax

Most of the statistical functions accept the data in a certain syntax known as “Formula”

```
# Includes y vs x1 variable
lm(y ~ x1, data=df)
# Includes y vs (x1 and x2) variables
lm(y ~ x1 + x2, data=df)
# Includes y vs all the variables in the data except y
lm(y ~ ., data=df)
# Includes y vs all the remaining variables except x1
lm(y ~ . -x1, data=df)
```

Important packages required for linear regression

```
library(dplyr)
library(tidyverse)
library(ggplot2)
library(reshape2)
```

Data

To load the data

```
# Read the csv file into dataframe df
df = read.csv("filename.csv")

# Read the csv file that has no header row and
# is separated by comma
df = read.csv("filename.csv", header=FALSE, sep = ",")
```

To get quick information about the data

```
nrow(); ncol()  # data dimensions
dim()           # dimensions
head()          # extract first part of data
tail()          # extract last part of data
colnames()      # column names
rownames()      # row names
summary()       # summary of the dataframe
```

To modify/transform the data

```
subset()        # subset data by condition
factor()        # create grouping variable
relevel()       # change reference level
cut()           # cut numeric into intervals
round()         # rounding numbers
c()             # concatenate numerics
seq()           # create sequence
margin.table()  # sum table entries
```

View/change data type

```
# To check
is.numeric()
is.character()
is.data.frame()
# To convert
as.numeric()
as.character()
as.data.frame()
```

Sort Dataframe

```
# Ascending
df[order(x1),]
# Descending
df[order(-x1)]
# Sort by multiple variables
df[order(x1, -x2)]
```

Treating missing values

```
# Check for NA values in a list
is.na()
# Check for NA values in a column of a dataframe
is.na(df$x1)

# Dropping based on all rows in the dataframe
drop_na(df)
# Dropping based on specific rows in the dataframe
df %>% drop_na(x1, x2)
```

Dummy variable encoding

```
# Note 'dcast' is a function of reshape2 package
newdata <- dcast(data = data, Outcome ~ Variable, length)
```

Linear Regression

```
model <- lm()           # Fit Linear model
summary(model)          # Summary of the model
coef(model)             # Estimated parameters
confint(model)          # CI for estimates
anova(model)            # Anova test for estimates
plotModel(model)        # Plot regression lines
```

Hypothesis tests

```
t.test()        # t-test
binom.test()    # binomial (exact) test
prop.test()     # approximate test
fisher.test()   # Fisher's exact test
cor.test()      # correlation test
chisq.test()    # chi-square test
```

Plots

```
hist()                  # Histogram
pairs()                 # Correlation matrix plot
corrplot(cor(data))     # Correlation plot
# Boxplot
ggplot(data, aes(x=x, y=y))+
  geom_boxplot()
# Scatter plot
ggplot(data, aes(x=x, y=y))+
  geom_point()
# Scatter plot with regression line
ggplot(data, aes(x=x, y=y)) +
  geom_point()+
  geom_smooth(method=lm)
```