**Introduction**

EDA consists of univariate (1-variable) and bivariate (2-variables) analysis.  
In this post we will review some functions that lead us to the analysis of the first case.

* Step 1 - First approach to data
* Step 2 - Analyzing categorical variables
* Step 3 - Analyzing numerical variables
* Step 4 - Analyzing numerical and categorical at the same time

Covering some key points in a basic EDA:

* Data types
* Outliers
* Missing values
* Distributions (numerically and graphically) for both, numerical and categorical variables.

**Type of analysis results**

They can be two: informative or operative.

* **Informative** - For example plots or any long variable summary. We cannot filter data from it, but give us a lot of information at once. Most used on the **EDA** stage.
* **Operative** - The results can be used to take an action directly on the data workflow (for example, selecting any variables whose percentage of missing values are below 20%). Most used in the **Data Preparation** stage.

Practical In R

About data set

The datasets consist of several medical predictor (independent) variables and one target (dependent) variable, **Outcome**. Independent variables include the number of pregnancies the patient has had, their BMI, insulin level, age, and so on.

Data Description

* Pregnancies:Number of times pregnant
* Glucose:Plasma glucose concentration a 2 hours in an oral glucose tolerance test
* BloodPressure: Diastolic blood pressure (mm Hg)
* SkinThickness: Triceps skin fold thickness (mm)
* Insulin: 2-Hour serum insulin (mu U/ml)
* BMI:Body mass index (weight in kg/(height in m)^2)
* DiabetesPedigreeFunction: Diabetes pedigree function
* Age:Age (years)
* Outcome: Class variable (0 or 1)

1 indicates diabetes is present

**Simple Data Understanding in R**

**Exploratory Data Analysis Part 1**

1. To Read data from CSV file to R

diabet<-read.csv(file.choose(),header = TRUE,sep = ",")

* Diabet: is the variable we are creating to store the csv file in form of data frame
* Read.csv :is used as we are reading csv file
* File.choose () :wil open the browser to select desired csv file
* Header=TRUE: will treat first row as a header
* Sep: as we have csv that comma separated file , we use “,”

1. To extract first few lines of data set

**head(diabet)**

* This will produce first few , by default 6 lines of the dataset

1. To check data type of every variable in dataset

**str(diabet)**

1. To check summary of entire data frame object

**summary(diabet)**

1. To check first 10 rows of the dataset

**diabet[1:10,]**

1. To check only 2 columns of the dataset

**diabet[,1:2]**

1. To display first 10 rows and only 2 columns of the dataset

**diabet[1:10,1:2]**

NOTE: when we want to fetch rows we mention datasetname[no\_of\_rows,]

When we want to fetch columns we mention datasetname[,no\_of\_cols]

When we want to fetch both we write datasetname[no\_of\_rows,no\_of\_cols]

1. To display observations having no of pregnancies =1

**newdata1<-subset(diabet,diabet$Outcome=="1")**

**newdata1**

Here we created new variable called as newdata1 and we are storing the subset of diabet data in newdata1. It is mandatory to write again newdata1 in order to view the output on console as we are creating new variable to store result of subsetting

1. To display multiple conditions for subsetting

**newdata2<-subset(diabet,diabet$Pregnancies=="1" & diabet$Outcome=="0")**

**newdata2**

1. To display multiple conditions for subsetting as well as selecting only required colums from the subset

**newdata3<-subset(diabet,diabet$Pregnancies=="1" | diabet$Outcome=="0",select=c(1,2))**

**newdata3**

1. To sort the data of a column in ascending order

**newdata4<-diabet[order(diabet$BMI), ]# we are sorting on all rows hence we are not writting anything after ,**

**newdata4**

1. To sort the data of a column in descending order

**newdata5<-diabet[order(-diabet$BMI),]**

**newdata5**

1. To aggregate the value based on one column

**newdata6<-aggregate(BMI~Outcome,data=diabet,FUN=mean)**

**newdata6**

this will calculate mean of BMI with respect to Outcome variable from diabet dataset

1. To check if any column contains missing observation

**colSums(is.na(diabet))**

1. To plot Histogram of a particular column in dataset

**hist(diabet$BMI)**

1. To plot boxplot of a particular column in dataset

**boxplot(diabet$BMI)**

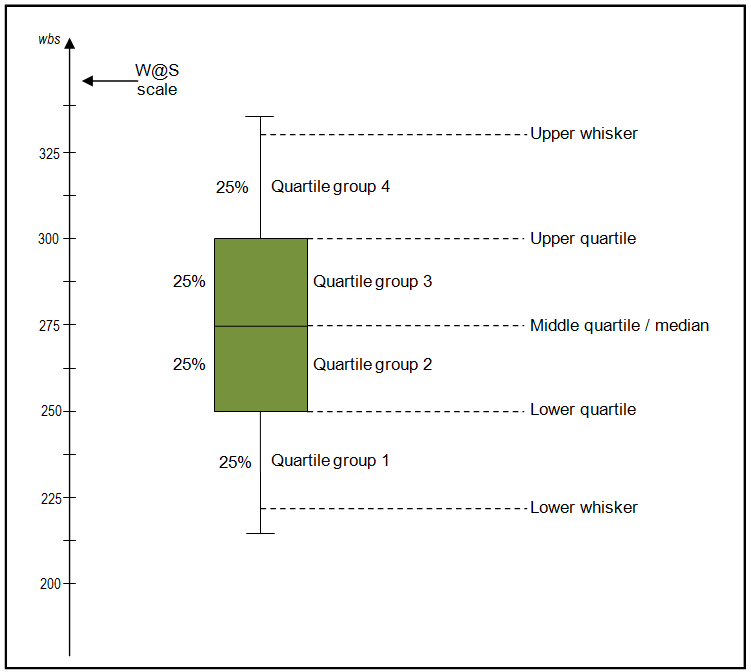
1. To view properties of particular column of data

**mean(diabet$BMI)**

**median(diabet$BMI)**

**max(diabet$BMI)**

**min(diabet$BMI)**

NOTE

Boxplot

To begin with, scores are sorted. Then four equal sized groups are made from the ordered scores. That is, 25% of all scores are placed in each group. The lines dividing the groups are called quartiles, and the groups are referred to as quartile groups. Usually we label these groups 1 to 4 starting at the bottom.

## Definitions

**Median**  
The median (middle quartile) marks the mid-point of the data and is shown by the line that divides the box into two parts. Half the scores are greater than or equal to this value and half are less.

**Inter-quartile range**  
The middle “box” represents the middle 50% of scores for the group. The range of scores from lower to upper quartile is referred to as the inter-quartile range. The middle 50% of scores fall within the inter-quartile range.

**Upper quartile**  
Seventy-five percent of the scores fall below the upper quartile.

**Lower quartile**  
Twenty-five percent of scores fall below the lower quartile.

**Whiskers**  
The upper and lower whiskers represent scores outside the middle 50%. Whiskers often (but not always) stretch over a wider range of scores than the middle quartile groups.

### 

### Interpreting box plots/Box plots in general

Box plots are used to show overall patterns of response for a group. They provide a useful way to visualize the range and other characteristics of responses for a large group.

Comprehensive code in r

|  |
| --- |
| diabet<-read.csv(file.choose(),header = T,sep = ",")  head(diabet)  str(diabet)  summary(diabet)  diabet[1:10,]  diabet[,1:2]  diabet[1:10,1:2]  newdata1<-subset(diabet,diabet$Outcome=="1")  newdata1  newdata2<-subset(diabet,diabet$Pregnancies=="1" & diabet$Outcome=="0")  newdata2  newdata3<-subset(diabet,diabet$Pregnancies=="1" | diabet$Outcome=="0",select=c(1,2))  newdata3  newdata4<-diabet[order(diabet$BMI), ]# we are sorting on all rows hence we are not writting anything after ,  newdata4  newdata5<-diabet[order(-diabet$BMI),]  newdata5  names(diabet)  newdata6<-aggregate(BMI~Outcome,data=diabet,FUN=mean)  newdata6  colSums(is.na(diabet))  hist(diabet$BMI)  boxplot(diabet$BMI)  mean(diabet$BMI)  median(diabet$BMI)  max(diabet$BMI)  min(diabet$BMI) |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* HAPPY LEARNING\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*