DATA CLEANING AND MANIPULATION IN R STUDIO

To apply any statistical model or operation on data first we need to clean our data with different parameters so that we would be able to extract correct information from data. A very usual example of this cleaning is detection and removal of outliers. Having outliers in data make our results biased, a very common example in this context is average of income distribution. If we try to find out the average income of people live in Mumbai then it would contain the income of Ambani too and our average income would be too high then the actual average income of an individual lives in Mumbai. To get a true(nearer) result we should remove Ambani income from the group. Not only outliers in data create problems with result interpretation but even some values prevent us from getting results like the missing values or string value in numerical columns. Sometimes it would be very difficult to get rid of from these problems and get the correct result.

This paper will guide you through the methods of dealing with all such problems that can occur to you at the time of data analysis, this entire paper is based on R Studio software and all the analysis is included in the R Script that can only be executed in R Studio(I have used R version 3.5.1). The data that I've used for this analysis is 'Survey' data that is available in 'MASS' package of R and this can be installed from R repository.

- Install.package("MASS", dependencies = T, repos = 'https://cran.r-project.org/index.html')
- library('MASS')

Above command will install 'MASS' package in your system and enable it for your usage(until R Studio is running).

Now, let's call the dataset that we require

- data(survey)
- View(survey)

These commands will import the survey data into R Studio and present it in R Script section.

Here you can see that in this data set there are many values which is missing(named as 'na') and will create problem if you will try to run a statistical or mathematical operation on it. As if you will try to compute the mean of a row that contains an 'NA' value it will return you the result as 'NA'(Since R won't be able to calculate with string value).There are plenty of ways to deal with this problem in one case we can use 'na.omit' command that will remove the entire row from the data. But sometime we won't have enough data and we can't remove the entire row(one's responses) just because a single value in that row is missing.

Our first job here is to find whether our data or columns in the data have missing values, if yes than how many. So that we would be able to decide what treatment can we adopt to deal with missing values.

To check whether my data column contains missing values apply:

- Is.na(data$column name) # It will return you "T" at all places where it finds an NA value and "F" where it will find non "NA".

But here comes a problem, this function only detects if the value is “NA” but what if we will have a blank value? For that we first have to change all blank values to “NA”(same operation will work with any other value).

Change blank values to “NA”

- Data[Column name==""] <- NA # It will change all blank values to “NA” and then we can easily remove “NAs”.

Now you can apply previous formula to see how many NAs are there in your data. One question still remains, I have to find out how many missing values are actually there(Count of missing value).

To find count of missing values:

- sum(is.na(data$column name)) or
- table(is,na(data$column name)),

Best way to detect number of missing values in each row is :

- colsums(is.na(data)) # It will give you the count of all missing values in each columns.

Also, if we have enough data and fewer missing values we can remove those rows by:

- na.omit(data)) # or,
- data[complete.cases(data),] # it will keep only those rows that have no missing values.

So far we have seen the scenarios in which we can remove the rows containing missing values but sometimes removing rows is not an option and we have to generate those missing values. In this scenario of generating new values in dataset we have to keep in mind few things about data type, data distribution and skewness.

First scenario in which my data column that contains missing value is normally distributed in that case best suitable value to fill for missing value is the mean value of rest of the values. But, how do we decide whether that data is normally distributed or not?

The best way to check whether the data is normally distributed or not is plot and the best plot for this is histogram.

To plot histogram in R:

- hist(data$column_name)

This will show you the graph of your column and now you have to check whether the distribution of your data is normal or not, if you find out higher bars on the center and adjacent bars symmetric then your data is normally distributed and we can fill up missing values with mean value. To insert mean value first we have to compute it, and the best way to compute the summary of your dataset is through summary command.

- Summary(data) # it will give all the details(statistical) about your dataset.

Now since you have the mean value,

- Data[column name == “”] -> mean

What if my data is not normally distributed and skewed towards a side, in that case we replace missing values with median value,

- Data[column name == “”] -> median

So far we have discussed about the continuous data but what if my data is categorical? In thecase of categorical data best value to replace with missing value is mode of that table.

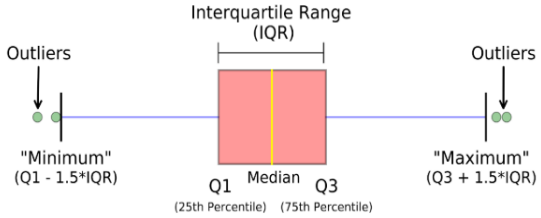
- Data[column name == “”] -> mode

Now we have dealt with missing values in our data, next step is to detect outliers. Outliers are extreme values that change your results by a significant amount and lies in the extent of data.

The best(Conventional) method to detect outliers in dataset is plotting boxplot(whisker plot) and you can see the dots outside from the plots which is to be considered as outliers. In the analysis of data or computation of any statistical result from data first we have to remove these values.

To plot boxplot:

- Boxplot(data[column name])



In the boxplot all the outliers would come out as dots from the plot(Kind of brick with two tails), As shown in this picture.

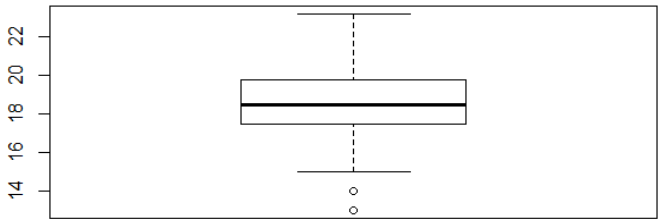
The brick represent entire data and both tails are the limits for which we are allowed to take the values from. These tails represent the range of both side as maximum value and minimum value till then we can accept our data value.

Values	Meaning	Relevance
Minimum Value	The minimum range till we can expect a value to be included in analysis.	Any value lesser than this should consider as outlier.
Q_1	First 25^{th} percentile of the data.	First quarter of the data.
Median	The mid value of the data	Half of the data.
Q_3	75^{th} percentile of the data	$3/4^{th}$ of the data.
Maximum Value	The maximum range till we can expect a value to be included in analysis.	Any value greater than this should consider as outlier.

The minimum value that we can consider as to be the part of our data should be greater than $Q_1 * 1.5$ IQR and the maximum value $Q_3 * 1.5$IQR, here is Inter quartile range and this is calculated by $Q_3 – Q_1$. The line in the mid is the median of the data.

As if we plot a boxplot for the column ‘Wr.Hnd’ from survey data we see that there will be two dots below the lower tail(value).These two dots should be considered as outliers when we do analysis and keeping this column as the part of analysis.

- Boxplot(survey$ Wr.Hnd)



Next step is to remove these outliers from data while conducting analysis. Or exclude these values.

Note: This is an early relies paper and I will update it soon with next steps in data cleaning and references.