**R PRACTICE REPORT**

**PROBABILITY AND STATISTICS ASSIGNMENT**

**WEEK 3**

**MODULE 3**

By : **HARSHIT GAUR**

MASTER OF PROFESSIONAL STUDIES IN ANALYTICS

ALY 6010 : PROBABILITY THEORY AND INTRODUCTORY STATISTICS

NOVEMBER 19, 2021

To : **PROF. AMIN KARIMPOUR**

# **ABSTRACT**

**Data analytics** is a discipline focused on extracting insights from data. It comprises of the processes, tools and techniques of data analysis and management, including the collection, organization, and storage of data. The chief aim of data analytics is to apply statistical analysis and technologies on data in order to find trends and solve problems.

**Big data** can be analysed for insights that lead to better decisions and strategic business moves. As data sets grow bigger and more complex, it is important to extract valuable insight from your data.

Analysis should focus on improvement and developing a strategy for improving the pattern recognition and findings for better efficiency.

It provides insights about the top performing and underperforming products/services/entitles in the data set, the patterns in their abilities and resonance of their types.

# **INTRODUCTION**

It is strongly encouraged to find and choose a data set in an area where one is more interested and is personally motivated to explore about.

Initially, I decided to go with a dataset that will help me hone the skills that I have learned so far in this course and go even beyond it by acquiring more knowledge and broadening my prowess of the analytical skills and R programming language. Then, I came across a dataset which not only interested me very much but also gave me various ideas to implement using that data set.   
  
 I grew up watching my all-time favourite TV Show (Anime) - Pokémon and the dataset which I chose is about the different types of Pokémon present along with their attributes. The dataset contains various numerical and categorical data. It has 1,045 data points with 11 features related to the primary types, attack attributes, defence attributes, hit points, special attributes and more. I decided to put my soft spot for the series to better use and up-skill myself in the analytical, visualization, and programmatical aspects of the domain.

The features available in the data set are -

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Feature** | **Dictionary** |
| 1. | Name | Name of the Pokémon |
| 2. | Name2 | Secondary name of the Pokémon |
| 3. | Primary.Type | Primary type |
| 4. | Secondary.Type | Secondary type to which Pokémon belongs to |
| 5. | Attack | Attack attribute |
| 6. | Defense | Defence Quality |
| 7. | HP | Hit Points |
| 8. | Sp.Attack | Special Attack attribute |
| 9. | Sp.Defense | Special Defense attribute |
| 10. | Speed | Speed of the Pokémon |
| 11. | Total | Total Qualities (Summation) |

*Table 1: Features of the data set with their dictionary.*

The data set was obtained from the below URL and will be referred in the bibliography as well :

*The World of Pokémons*. (2021, September 29). Kaggle. https://www.kaggle.com/hamdallak/the-world-of-pokemons

From the structure of the data set, the features, their types, and values can be determined.

|  |
| --- |
| *Figure 1: Structure of the data set.* |

Some of the data points from the summary of the data set are present below.

|  |
| --- |
| *Figure 2: Summary of the data set.* |

# **DATA PRE-PROCESSING AND CLEANING**

The first blush of the data set portrayed the data as not clean. After an **Initial Data Analysis (IDA)** process assisted with some graph visualizations as well, I found out that the data set requires pre-processing and cleaning in it.

Data cleaning is an important & necessary factor in the data analysis process. As the famous quote says - *"Garbage In, Garbage Out."*

I proceeded with the needed & necessary step of data cleaning using RStudio application on this data set to wrangle and eliminate all the garbage values which consisted of :

* Missing values
* Duplicate values

Some of the actions performed for data cleaning are :

* 1. The categorical features in the data set were checked for inconsistencies in them. If there were some found, we would need to normalize them using *if..elseif..else* condition with data manipulation.

|  |
| --- |
| *Figure 2: Inconsistencies Check on the feature 'Primary Type' in the data set.* |

* 1. We checked the data set for NA and NULL values in its features using the functions which can be referred from the below snapshot.

|  |
| --- |
| *Figure 3: Checking records with NA, NULL values from the data set.* |

* 1. There were many data points with missing or blank values in the features 'Name2' and 'Secondary.Type'. Removing these data points from the data set made no sense as it would have lost us a majority of the data. Instead, the missing or empty values were replaced by the word "NoName & NoType" using a function ***GSUB()***.

|  |
| --- |
| *Figure 4: Eliminating records with NA, missing values from the data set.* |

* 1. The data set was checked for duplicate values in the combination of 'Name' and 'Name2' because of their uniqueness when combined. We found some duplicate values from the output below where "TRUE" is written.

|  |
| --- |
| Text  Description automatically generated  *Figure 5: Check for Duplication in the combination of features from the data set.* |

* 1. The duplicate records were retrieved from the data set using a combination of functions '**WHICH()'** and '**DUPLICATED()'**.

|  |
| --- |
| *Figure 6: Retrieving the duplicate records from the data set.* |

* 1. The duplicate values found in the above step were eliminated from the data set using **'FILTER()'** function of the ***DPLYR*** library.

|  |
| --- |
| *Figure 7: Eliminating the duplicate records from the data set.* |

* 1. After performing the previous steps, the data set was omitted for NA values if present using the below function **'NA.OMIT()'**.

|  |
| --- |
| *Figure 8: Removing the NA, missing values from the data set.* |

* 1. The data set do not contain any kind of unbalanced features.

# **EXPLORATORY DATA ANALYSIS**

The descriptive statistics of the features of the data set can be summarised to calculate the statistics -

|  |
| --- |
| *Figure 9: Summary of the data set.* |

We used the **'DESCRIBE()**' function from the package **'PSYCH'** to find out the descriptive statistics of features of the data set. The following observations can be made using the statistics found in summary of the data set -

1. The **mean** of the attribute *Attack* is around **80.47** with a **standard deviation** of **32.41** and **quartiles value (Lower Quartile - 55, Higher Quartile - 100).**   
   From the observations, we can calculate that the data points around **maximum value (190)** can be *outliers* to the feature.
2. The **mean** of the attribute *Defense* is around **74.66** with a **standard deviation** of **31.24** and **quartiles value (Lower Quartile - 50, Higher Quartile - 90).**   
   From the observations, we can calculate that the data points around **maximum value (250)** can be *outliers* to the feature.
3. The **mean** of the attribute *HP (Hit Point)* is around **70.07** with a **standard deviation** of **26.67** and **quartiles value (Lower Quartile - 50, Higher Quartile - 82).**   
   From the observations, we can calculate that the data points around **maximum value (255)** can be *outliers* to the feature.
4. The **mean** of the attribute *Sp.Attack (Special Attack)* is around **73.02** with a **standard deviation** of **32.73** and **quartiles value (Lower Quartile - 50, Higher Quartile - 95).**   
   From the observations, we can calculate that the data points around **maximum value (194)** can be *outliers* to the feature.
5. The **mean** of the attribute *Sp.Defense (Special Defense)* is around **72.29** with a **standard deviation** of **28.07** and **quartiles value (Lower Quartile - 50, Higher Quartile - 90).**   
   We can calculate that data points around **maximum value (250)** can be *outliers* in it.
6. The **mean** of the attribute *Speed* is around **68.80** with a **standard deviation** of **30.21** and **quartiles value (Lower Quartile - 45, Higher Quartile - 90).**   
   From the observations, we can calculate that the data points around **maximum value (200)** can be *outliers* to the feature.
7. The **mean** of the attribute *Total (Sum of all attributes)* is around **439.32** with a **standard deviation** of **121.97** and **quartiles value (Lower Quartile - 330, Higher Quartile - 515).**   
   From the observations, we can calculate that the data points around **maximum value (1125)** can be *outliers* to the feature.

# NORMALITY

The normality of all the features of the data set was checked in order to understand the type of distribution -

|  |
| --- |
| *Figure 10: Density Plot of features of the data set.* |
| |  |  |  | | --- | --- | --- | | **S. No.** | **Feature** | **Shapiro Wilks Test (P-value)** | | 1. | Attack | 0.00701233 | | 2. | Defense | 0.0000001781662 | | 3. | HP | 0.0000000303332 | | 4. | Sp.Attack | 0.0004299911 | | 5. | Sp.Defense | 0.0003624276 | | 6. | Speed | 0.03911546 | | 7. | Total | 0.00133075 |   *Table 2: Shapiro-Wilks Test of the data set.* |
| *Figure 11: Q-Q Plots of the numeric features for Normality.* |

The density plots, Shapiro-Wilks Test, and the Quantiles-Quantiles plots of the features gave out the below results about the data set -

1. The density graphs of all the features show that they can be considered as normally distributed.
2. **Shapiro-Wilks Test** was used to verify the normality of a feature of the data set. But, since the p-value was less than 5%, we can reject the normality of the features.
3. The features *Attack, Sp.Attack, Speed* have the same line for Sample and Theoretical Quantiles. They can be considered to be normally distributed.
4. The other features will need to be treated to make them normally distributed.

# DESCRIPTIVE STATISTICS

After the process of data pre-processing and cleaning, I have calculated the descriptive statistics of all the features of the data set to check the difference between the previous set and this new set of descriptive statistics. The function **DESCRIBE()** of the package **{PSYCH}**was used to calculate them. I have included *Quantiles* and *Inter-Quartile Range* columns as well.

|  |
| --- |
| *Figure 12: Describe the data set for descriptive statistics..* |
| *Figure 13: Descriptive Statistics after cleaning the data set.* |

1. We have eliminated some duplicate records from the data set. This changed the descriptive statistics values.
2. The **mean** of the attribute *Attack* is around **80.49** instead of **80.47** (earlier) with changes in the *standard deviation.*
3. The **mean** of the attribute *Defense* is around **74.70** instead of **74.66** (earlier) with a slight change of 0.02 in **standard .**
4. Few other statistics have been changed which can be figured out from the table above.

# DESCRIPTIVE STATISTICS (by Grouping)

|  |
| --- |
| *Figure 14: Describe the data set with GROUPINGS (based on features)* |
| *Table  Description automatically generated*  *Figure 15: Descriptive Statistics by GROUPS (based on Primary Attack feature)* |
| *A screenshot of a computer  Description automatically generated with low confidence*  *Figure 16: Descriptive Statistics by GROUPS (based on Primary Attack & Secondary Attack features)* |

1. The descriptive statistics of the data set was calculated after applying **GROUPINGs** based on *Primary Type* and *Secondary Type*.
2. There are 18 groups for which these descriptive statistics were calculated.
3. The mean of the *Attack* feature from the groups above shows that :

|  |  |  |
| --- | --- | --- |
| **GROUP** | **MEAN (*ATTACK*)** | **RANKING (attack based)** |
| BUG | 71.07 | 3RD |
| DARK | 81.93 | 2ND |
| DRAGON | 107.02 | 1ST |

1. The standard deviation of the *HP (Hit Power)* feature from the groups above shows that :

|  |  |  |
| --- | --- | --- |
| **GROUP** | **S.D. (*Hit Power*)** | **RANKING (HP based)** |
| BUG | 17.34 | 3RD |
| DARK | 31.63 | 2ND |
| DRAGON | 37.13 | 1ST |

1. Similar type of observations can be made from the statistics table above.

**Density Plot of Horse Power Attribute of Pokémons**

We have plotted the Density plots of *Horse Power* attribute of the Pokémons attribute which represents the distribution of the horse power (HP) feature of all the Pokémons in the data set. The plots belong to the original values and logarithm of these values.

|  |
| --- |
| *Figure 17: Density Plot of Horse Power attribute* |
| *Figure 18: Density Plot of Logarithm 10 of Horse Power attribute* |

1. From the density plot of Horse Power attribute (Figure 19), we can figure out that the feature can be anticipated to be normally distributed in the data set.
2. If we apply the logarithmic function to the data set of the feature, the infographic depicts the normality of the feature. The feature can be considered to be normally distributed.

|  |
| --- |
| **Chart, line chart  Description automatically generated**  *Figure 19: Q-Q Plots of normal and Logarithm 10 of Horse Power attribute* |

**One Sample T-Tests**

The data set has been grouped according to '*Primary Type*' attribute for the statistical hypothesis testing. We applied One Sample t-test 3 times varied by the feature attributes or types of the Pokémons.

# ONE SAMPLE T-TEST ~ WATER Pokémons on MEAN OF ATTACK feature

|  |
| --- |
| *Figure 20: One Sample t-Test of Water type Pokémons with Mean of Attack feature* |
| *Figure 21: Result of One Sample t-Test of Water type Pokémons with Mean of Attack feature* |
| *Figure 22: P-Value of One Sample t-Test of Water type Pokémons with Mean of Attack feature* |

**NULL HYPOTHESIS, H0** : True Mean is less than 74.99254

**ALTERNATE HYPOTHESIS, H1** : True Mean is greater than 74.99254

**ALTERNATIVE** : Greater

**DEGREE OF FREEDOM** : 1043

**P-VALUE** : 0.00000002691689

**Observation** : Since, the p-value of our One Sample t-test is 0.00000002691689, which is **less than alpha = 0.05, we reject the Null Hypothesis of the test**. The **T.TEST()** function in R takes the confidence interval as 95% (0.95) by default.

This means that we have sufficient evidence to say that **Population True Mean** of the *Attack* feature of all the Pokémons in the data set is greater than the **True Mean** of *Attack* feature of Pokémons of **Water** type.

# ONE SAMPLE T-TEST ~ WATER Pokémons on MEAN OF HORSE POWER feature

|  |
| --- |
| *Figure 23: One Sample t-Test of Water type Pokémons with Mean of Horse Power feature* |
| *Figure 24: Result of One Sample t-Test of Water type Pokémons with Mean of Horse Power feature* |
| *Figure 25: P-Value of One Sample t-Test of Water type Pokémons with Mean of Horse Power feature* |

**NULL HYPOTHESIS, H0** : True Mean is less than 70.091

**ALTERNATE HYPOTHESIS, H1** : True Mean is greater than 70.091

**ALTERNATIVE** : Greater

**DEGREE OF FREEDOM** : 1043

**P-VALUE** : 0.8327445

**Observation** : Since, the p-value of our One Sample t-test is 0.8327445, which is **greater than alpha = 0.05, we cannot reject the Null Hypothesis of the test**. The **T.TEST()** function in R takes the confidence interval as 95% (0.95) by default.

This means that we do not have sufficient evidence to say that **Population True Mean** of the *Horse Power* feature of all the Pokémons in the data set is greater than the **True Mean** of *Horse Power* feature of Pokémons of **Water** type.

# ONE SAMPLE T-TEST ~ PSYCHIC Pokémons on MEAN OF HORSE POWER feature

|  |
| --- |
| *Figure 26: One Sample t-Test of Psychic type Pokémons with Mean of Horse Power feature* |
| *Figure 27: Result of One Sample t-Test of Psychic type Pokémons with Mean of Horse Power feature* |
| *Figure 28: P-Value of One Sample t-Test of Psychic type Pokémons with Mean of Horse Power feature* |

**NULL HYPOTHESIS, H0** : True Mean is greater than 73.67089

**ALTERNATE HYPOTHESIS, H1** : True Mean is less than 73.67089

**ALTERNATIVE** : Less

**DEGREE OF FREEDOM** : 1043

**P-VALUE** : 0.000007941574

**Observation** : Since, the p-value of our One Sample t-test is 0.000007941574, which is **less than alpha = 0.05, we can reject the Null Hypothesis of the test**. The **T.TEST()** function in R takes the confidence interval as 95% (0.95) by default.

This means that we have sufficient evidence to say that **Population True Mean** of the *Horse Power* feature of all the Pokémons in the data set is greater than the **True Mean** of *Horse Power* feature of Pokémons of **Psychic** type.

**Two Sample T-Tests**

The data set has been grouped according to '*Primary Type*' attribute for the statistical hypothesis testing.

# WATER vs PSYCHIC Pokémons on MEAN OF ATTACK feature

|  |
| --- |
| *Figure 29: Two Sample t-Test of Water vs Psychic type Pokémons with Mean of Attack feature* |
| *Figure 30: Result of Two Sample t-Test of Water vs Psychic type Pokémons with Mean of Attack feature* |
| *Figure 31: P-Value of Two Sample t-Test of Water vs Psychic type Pokémons with Mean of Attack feature* |

**NULL HYPOTHESIS, H0** : True Difference in Means of both Populations (Water & Psychic) is equal to 0.

**ALTERNATE HYPOTHESIS, H1** : True Difference in Means of both Populations (Water & Psychic) is not equal to 0.

**ALTERNATIVE** : Two. Sided

**DEGREE OF FREEDOM** : 1043

**P-VALUE** : 0.8139599

**Observation** : Since, the p-value of our Two Sample t-test is 0.8139599, which is **greater than alpha = 0.05, we cannot reject the Null Hypothesis of the test.** The **T.TEST()** function in R takes the confidence interval as 95% (0.95) by default.

This means that we do not have sufficient evidence to say that **True Difference in Both Population Means** of the *Water & Psychic* Pokémons in the data set is not equal to 0.

# WATER vs PSYCHIC Pokémons on MEAN OF SPECIAL ATTACK feature

|  |
| --- |
| *Figure 32: Two Sample t-Test of Water vs Psychic type Pokémons with Mean of Special Attack feature* |
| *Figure 33: Result of Two Sample t-Test of Water vs Psychic type Pokémons with Mean of Special Attack feature* |
| *Figure 34: P-Value of Two Sample t-Test of Water vs Psychic type Pokémons with Mean of Special Attack feature* |

**NULL HYPOTHESIS, H0** : True Difference in Means of both Populations (Water & Psychic) is equal to 0.

**ALTERNATE HYPOTHESIS, H1** : True Difference in Means of both Populations (Water & Psychic) is not equal to 0.

**ALTERNATIVE** : Two. Sided

**DEGREE OF FREEDOM** : 1043

**P-VALUE** : 0.000005909999

**Observation** : Since, the p-value of our Two Sample t-test is 0.000005909999, which is **less than alpha = 0.05, we can reject the Null Hypothesis of the test.** The **T.TEST()** function in R takes the confidence interval as 95% (0.95) by default.

This means that we have sufficient evidence to say that **True Difference in Both Population Means** of the *Water & Psychic* Pokémons in the data set is not equal to 0.

**Hypothesis Testing (P-Test)**

The data set has been grouped according to '*Primary Type*' attribute for the statistical hypothesis testing (P-test). We have taken the **T-Score as Test Statistic** in this test**.** We are applying this test on *the Mean of Attack feature of Water type* *Pokémons*.

|  |
| --- |
| *Figure 35: P-Test of Pokémons with Mean of Attack feature of Water Pokémons.* |
| *Figure 36: Result of P-Test of Pokémons with Mean of Attack feature of Water Pokémons* |
| *Figure 37: Area to the right of Test Statistic using P-Test of Pokémons with Mean of Attack feature of Water Pokémons* |

**NULL HYPOTHESIS, H0** : The True Mean of Attack feature in the whole Populations is greater than or equal to mean of attack feature of Water type Pokémons.

**ALTERNATE HYPOTHESIS, H1** : The True Mean of Attack feature in the whole Populations is less than mean of attack feature of Water type Pokémons.

**LOWER.TAIL** : False

**DEGREE OF FREEDOM** : 1043

**P-VALUE** : 0.00000002691689

**Observation** : Since, the p-value of our Hypothesis Testing using p-test is 0.00000002691689, which is **less than alpha = 0.05, we can reject the Null Hypothesis of the test.** The **PT()** function in R takes the confidence interval as 95% (0.95) by default.

This means that we have sufficient evidence to say that **True Mean of Attack feature in the whole Population is less than Mean of Attack feature in Water type Pokémons.**

# **CONCLUSION**

The dataset of 'The World of Pokémons' has provided various insights about the types, abilities of the Pokémons and the patterns between them as well. We performed initial data analysis, exploratory data analysis, calculated various statistics, plotted a few visualisation graphs in order to understand the analysis properly, and performed normality tests and various hypothesis tests. The below points can be inferred from the analysis :

* We used density plots to determine the normality of the features visually and found out that we can anticipate few features to be normally distributed.
* The Shapiro-Wilks Tests and Q-Q plots provided better information about the normality of the features. **Attack, Speed, Sp.Attack** can be considered to be normally distributed when used in original or logarithmic format.
* The application of One Sample t-test on various features varied with types of Pokémons helped in rejecting or not the null hypothesis in each cases and provided the better understanding of the true populations.
* We used Two Sample t-tests as well to determine the relationship between two features of the data set.
* Hypothesis Testing using P-Test is also performed on the data set to determine if the true mean of the population for their Attack feature is greater than or equal to the mean of Attack feature of Water type Pokémons.
* It can also be concluded that the **most powerful species** in the Pokémon world is **DRAGON**. The total mean value of abilities is around **536**.*In most cases, dragon species Pokémons would come out as winners in the fights.*
* **After 125, the correlation becomes negative for PSYCHIC type Pokémons.** This means that PSYCHIC type Pokémons tend to lose their defense abilities when they acquire more attacking prowess.

# **BIBLIOGRAPHY**

1. *The World of Pokemons*. (2021, September 29). Kaggle. https://www.kaggle.com/hamdallak/the-world-of-pokemons
2. *Guide To Data Cleaning: Definition, Benefits, Components, And How To Clean Your Data*. (2021). Tableau. https://www.tableau.com/learn/articles/what-is-data-cleaning
3. Burns, E. (2021, January 16). *Data Cleaning in R Made Simple - Towards Data Science*. Medium. https://towardsdatascience.com/data-cleaning-in-r-made-simple-1b77303b0b17
4. *Normality Test in R - Easy Guides - Wiki - STHDA*. (2021). STHDA. http://www.sthda.com/english/wiki/normality-test-in-r
5. *A Grammar of Data Manipulation*. (2021). Dplyr. https://dplyr.tidyverse.org/
6. *Home - RDocumentation*. (2021). Functions in R - Documentation. https://www.rdocumentation.org/
7. *Chapter 5 Probability and Statistics in R | Applied Statistics with R*. (2021, July 23). Probability and Statistics in R. https://daviddalpiaz.github.io/appliedstats/probability-and-statistics-in-r.html

# **APPENDIX**

#---------------------- Week\_2\_Module\_2\_R-Script ----------------------#

print("Author : Harshit Gaur")

print("Week 2 Assignment - Module 2 R Pratice")

# Importing the packages.

listOfPackages <- c(

"dplyr", "tidyr", "plyr", "tidyverse", "RColorBrewer", "plotrix", "scales", "ggplot2",

"data.table", "reshape", "gridExtra", "vtable", "moments", "ggpubr", "psych", "GGally"

)

for (package in listOfPackages) {

if (package %in% rownames(installed.packages()) == FALSE)

{ install.packages(package) }

# Importing the package.

library(package, character.only = TRUE)

}

# STEP 2: Import data set

# Note: Change the working directory as per the file's location.

setwd("/Users/HarshitGaur/Documents/Northeastern University/MPS Analytics/ALY 6010/Class 2/R Practice Assignment/")

pokemon\_dataset <- read.csv("pokemons dataset.csv", header = TRUE)

# Display the data set.

View(pokemon\_dataset)

# Print the structure of 'pokemons dataset.csv' data set

str(pokemon\_dataset)

# Print the summary of 'pokemons dataset.csv' data set

summary(pokemon\_dataset)

st(pokemon\_dataset)

# Describe the summary of the data set by providing Descriptive Statistics.

View(describe(pokemon\_dataset, skew = FALSE, quant = c(0.25, 0.75), IQR = TRUE))

#------------------- Data Cleaning -------------------#

# To check inconsistencies in the 'Primary Type - Character' feature of the data set.

unique(pokemon\_dataset$Primary.Type)

# To check NA, NULL values in the data set.

sum(is.na(pokemon\_dataset))

sum(is.null(pokemon\_dataset))

# Replacing Empty Values in the features with the word 'NoName & NoType'

pokemon\_dataset$Name2 <- gsub('^$', 'NoName', pokemon\_dataset$Name2)

pokemon\_dataset$Secondary.type <- gsub('^$', 'NoType', pokemon\_dataset$Secondary.type)

# Check the duplicate values in a combination of 2 features.

duplicated(pokemon\_dataset[,1:2])

# Retrieving the duplicated records from the data set.

pokemon\_dataset[which(duplicated(pokemon\_dataset[,1:2])),]

# Eliminating the duplicated records using the indexes provided by the above step.

pokemon\_dataset <- pokemon\_dataset %>% filter( !row\_number() %in% 44)

# Removing 'NA, Missing Values' from the data set.

pokemon\_dataset <- na.omit(pokemon\_dataset)

#------------------- Exploratory Data Analysis -------------------#

# Check Normality using Density Graphs of all the univariates.

normality\_attack <- ggdensity(pokemon\_dataset$Attack, main = "Density plot of Attack", xlab = "Attack", fill = "#ffa514")

normality\_defense <- ggdensity(pokemon\_dataset$Defense, main = "Density plot of Defense", xlab = "Defense", fill = "#edf759")

normality\_hp <- ggdensity(pokemon\_dataset$HP, main = "Density plot of Hit Points", xlab = "Hit Points", fill = "#baf54c")

normality\_spAttack <- ggdensity(pokemon\_dataset$Sp.Attack, main = "Density plot of Special Attack", xlab = "Special Attack", fill = "#4cf5bd")

normality\_spDefense <- ggdensity(pokemon\_dataset$Sp.Defense, main = "Density plot of Special Defense", xlab = "Special Defense", fill = "#40c7f7")

normality\_speed <- ggdensity(pokemon\_dataset$Speed, main = "Density plot of Speed", xlab = "Speed", fill = "#d27afa")

normality\_total <- ggdensity(pokemon\_dataset$Total, main = "Density plot of Total Attributes", xlab = "Total Attributes", fill = "#eabdff")

grid.arrange(normality\_attack,normality\_defense,normality\_hp,normality\_spAttack,normality\_spDefense,normality\_speed,normality\_total)

# Check Normality using Shapiro-Wilks Test

shapiro.test(pokemon\_dataset$Attack)

# Check Normality using Q-Q Plot of all the numeric features.

# Function to plot graph

qq\_plot <- function(numeric\_feature, mainTitle) {

qqnorm(numeric\_feature, pch = 5, frame = TRUE, main = mainTitle)

qqline(numeric\_feature, col = "#ffa514", lwd = 2)

}

# Changing Plot Matrix Size to 3x3.

par(mfrow = c(3,3))

# Check Normality using Q-Q Plot of 'Attack' Feature.

qq\_plot(pokemon\_dataset$Attack, "Attack")

# Check Normality using Q-Q Plot of 'Defense' Feature.

qq\_plot(pokemon\_dataset$Defense, "Defense")

# Check Normality using Q-Q Plot of 'HP' Feature.

qq\_plot(pokemon\_dataset$HP, "Horse Power")

# Check Normality using Q-Q Plot of 'Special Attack' Feature.

qq\_plot(pokemon\_dataset$Sp.Attack, "Special Attack")

# Check Normality using Q-Q Plot of 'Special Defense' Feature.

qq\_plot(pokemon\_dataset$Sp.Defense, "Special Defense")

# Check Normality using Q-Q Plot of 'Speed' Feature.

qq\_plot(pokemon\_dataset$Speed, "Speed")

# Check Normality using Q-Q Plot of 'Total Attributes' Feature.

qq\_plot(pokemon\_dataset$Total, "Total Attributes")

# Resetting Plot Matrix Size to 1x1.

par(mfrow = c(1,1))

# Check Skewness of the features

skewness(pokemon\_dataset$Attack)

skewness(pokemon\_dataset$Defense)

skewness(pokemon\_dataset$HP)

skewness(pokemon\_dataset$Sp.Attack)

skewness(pokemon\_dataset$Sp.Defense)

skewness(pokemon\_dataset$Speed)

skewness(pokemon\_dataset$Total)

# Check Skewness of the features

kurtosis(pokemon\_dataset$Attack)

kurtosis(pokemon\_dataset$Defense)

kurtosis(pokemon\_dataset$HP)

kurtosis(pokemon\_dataset$Sp.Attack)

kurtosis(pokemon\_dataset$Sp.Defense)

kurtosis(pokemon\_dataset$Speed)

kurtosis(pokemon\_dataset$Total)

# Describe the summary of the data set by providing Descriptive Statistics.

View(describe(pokemon\_dataset, skew = FALSE, quant = c(0.25, 0.75), IQR = TRUE))

# Describe the summary of the data set by grouping

describeBy(pokemon\_dataset, group = pokemon\_dataset$Primary.Type)

describeBy(pokemon\_dataset, group = pokemon\_dataset$Secondary.type)

# Plot an Grouped Line graph for attack using 'ggplot'

# ----------- Plot 1: Primary Type vs Summation of Attributes ----------- #

# Summation of 'Attributes' according to Primary Type feature

summationTable <- pokemon\_dataset %>% group\_by(pokemon\_dataset$Primary.Type) %>% dplyr::summarise(

sum\_attack = sum(Attack),

sum\_defense = sum(Defense),

sum\_hp = sum(HP),

sum\_spAttack = sum(Sp.Attack),

sum\_spDefense = sum(Sp.Defense),

sum\_speed = sum(Speed),

sum\_total = sum(Total),

)

# Converting the summation Table into a Data Frame.

summationTable <- data.frame((summationTable))

par(mar = c(4, 10, 8, 4) + 0.1)

ggplot(summationTable, aes(x = pokemon\_dataset.Primary.Type, group = 1)) +

geom\_line(aes(y = sum\_attack, color="Attack"), linetype="dotted", size=1) +

geom\_line(aes(y = sum\_defense, color="Defense"), linetype="longdash", size=1) +

geom\_line(aes(y = sum\_hp, color="HP"), linetype="F1", size=1) +

geom\_line(aes(y = sum\_spAttack, color="Special Attack"), linetype="F1", size=1) +

geom\_line(aes(y = sum\_spDefense, color="Special Attack"), linetype="solid", size=1) +

geom\_line(aes(y = sum\_speed, color="Speed"), linetype="twodash", size=1) +

scale\_color\_manual(

name = "Legends",

values = c("Attack" = "red", "Defense" = "blue", "HP" = "yellow", "Special Attack" = "orange", "Special Defense" = "Darkgreen", "Speed" = "magenta")) +

theme(

panel.background = element\_rect("#c9f5e4"),

axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1, size = 12)

) +

labs(

title = "Plot 2: Primary Type vs Summation of Attributes",

x = "Primary Type",

y = "Summation of Attributes"

) +

theme(plot.title = element\_text(hjust = 0.5, colour = "#7F3D17", face = "bold"))

# Plot an Scatter Plot Matrix graph for 'Attributes' using 'pairs'

# ----------- Plot 2: Scatter Plot Matrix between Abilities (Attributes) ----------- #

pairs(~Attack + Defense + HP + Sp.Attack + Sp.Defense + Speed, pokemon\_dataset, col=c("YELLOW", "RED"), pch = 5)

# Plot an Scatter Chart graph for 'Primary Type vs Attack' using 'ggplot'

# ----------- Plot 3: Scatter Plot of Primary Type vs Attack ----------- #

par(mar = c(2,4,2,4))

ggplot(data = pokemon\_dataset, aes(x = Primary.Type, y = Attack, color = Primary.Type)) +

geom\_point() +

labs(title = 'Scatter Plot of Primary Type vs Attack', x = 'Primary Type', y = 'Attack Attribute') +

theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1, size = 10))

# Plot an Jitter Chart graph for 'Primary Type vs Attack' using 'ggplot'

# ----------- Plot 4: Jittered Scatter Plot of Primary Type vs Attack ----------- #

par(mar = c(2,4,2,4))

ggplot(data = pokemon\_dataset, aes(x = Primary.Type, y = Attack, color = Primary.Type)) +

geom\_jitter(position = position\_jitterdodge()) +

labs(title = 'Jitter Plot of Primary Type vs Attack', x = 'Primary Type', y = 'Attack Attribute') +

theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1, size = 10))

# Plot an Scatter Chart graph for 'Attack vs Defense (Grouped in 3 Types)' using 'ggplot'

# ----------- Plot 5: Scatter Plot of Attack vs Defense (Grouped in 3 Types) ----------- #

dgw\_group <- filter(pokemon\_dataset, Primary.Type %in% c("DRAGON", "WATER", "PSYCHIC"))

par(mar = c(2,4,2,4))

ggplot(data = dgw\_group, aes(x = Attack, y = Defense, color = Primary.Type)) +

geom\_point() +

labs(title = 'Scatter Plot of Attack vs Defense', x = 'Attack Attribute', y = 'Defense Attribute') +

theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1, size = 10)) +

geom\_smooth(se = FALSE)

# Plot an Scatter Chart graph for 'Primary Type vs Attack varied with Defense' using 'ggplot'

# ----------- Plot 6: Scatter Plot of Primary Type vs Attack ----------- #

par(mar = c(2,4,2,4))

ggplot(data = pokemon\_dataset, aes(x = Primary.Type, y = Attack, color = Defense)) +

geom\_point() +

labs(title = 'Scatter Plot of Primary Type vs Attack varied with Defense', x = 'Primary Type', y = 'Attack Attribute') +

theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1, size = 10))

# Plot an Jitter Chart graph for 'Primary Type vs Attack varied with Defense' using 'ggplot'

# ----------- Plot 7: Jitter Plot of Primary Type vs Attack ----------- #

par(mar = c(2,4,2,4))

ggplot(data = pokemon\_dataset, aes(x = Primary.Type, y = Attack, color = Defense)) +

geom\_jitter(position = position\_jitterdodge()) +

labs(title = 'Jitter Plot of Primary Type vs Attack varied with Defense', x = 'Primary Type', y = 'Attack Attribute') +

theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1, size = 10))

# Plot an Box Plot for 'Primary Type vs Attack' using 'ggplot'

# ----------- Plot 8: Box Plot of Primary Type vs Attack ----------- #

par(mar = c(2,4,2,4))

ggplot(data = pokemon\_dataset, aes(x = Primary.Type, y = Attack, color = Primary.Type)) +

geom\_boxplot(outlier.size = 3.5) +

labs(title = 'Box Plot of Primary Type vs Attack varied with Defense', x = 'Primary Type', y = 'Attack Attribute') +

theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1, size = 10))

# Plot an Jitter Chart graph for 'Primary Type vs Attack varied with Defense' using 'ggplot'

# ----------- Plot 9: Jittered Box Plot of Primary Type vs Attack ----------- #

par(mar = c(2,4,2,4))

ggplot(data = pokemon\_dataset, aes(x = Primary.Type, y = Attack, color = Primary.Type)) +

geom\_boxplot(outlier.size = 3.5, outlier.stroke = 2) +

geom\_jitter(position = position\_jitterdodge()) +

labs(title = 'Jitter Plot of Primary Type vs Attack', x = 'Primary Type', y = 'Attack Attribute') +

theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1, size = 10))