R PRACTICE REPORT

PROBABILITY AND STATISTICS ASSIGNMENT

WEEK 6
MODULE 6

By : **HARSHIT GAUR**
MASTER OF PROFESSIONAL STUDIES IN ANALYTICS
ALY 6010 : PROBABILITY THEORY AND INTRODUCTORY STATISTICS
DECEMBER 17, 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.

Regression Analysis is a reliable method of identifying which variables have an impact on a topic of interest. The process of performing a regression allows us to confidently determine which factors matter most, which factors can be ignored, and how these factors influence each other. It is basically performed to determine the effects on target variables.

Dummy variable or Indicator Variable is an artificial variable created to represent an attribute or feature with 2 or more distinct categories/levels.

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 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 on that data set.

I grew up watching my all-time favourite TV Show (Animated Series) - 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 in 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.

```
> # Print the structure of 'pokemons dataset.csv' data set
> str(pokemon_dataset)
'data.frame':   1045 obs. of  11 variables:
 $ Name          : chr  "Bulbasaur" "Ivysaur" "Venusaur" "Venusaur" ...
 $ Name2         : chr  "" "" "" "Mega Venusaur" ...
 $ Primary.Type  : chr  "GRASS" "GRASS" "GRASS" "GRASS" ...
 $ Secondary.type: chr  "POISON" "POISON" "POISON" "POISON" ...
 $ Attack        : int  49 62 82 100 52 64 84 130 104 48 ...
 $ Defense       : int  49 63 83 123 43 58 78 111 78 65 ...
 $ HP            : int  45 60 80 80 39 58 78 78 78 44 ...
 $ Sp.Attack     : int  65 80 100 122 60 80 109 130 159 50 ...
 $ Sp.Defense    : int  65 80 100 120 50 65 85 85 115 64 ...
 $ Speed         : int  45 60 80 80 65 80 100 100 100 43 ...
 $ Total         : int  318 405 525 625 309 405 534 634 634 314 ...
```

Figure 1: Structure of the data set.

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

	Name	Name2	Primary.Type	Secondary.type	Attack	Defense	HP	Sp.Attack	Sp.Defense	Speed	Total
1	Bulbasaur		GRASS	POISON	49	49	45	65	65	45	318
2	Ivysaur		GRASS	POISON	62	63	60	80	80	60	405
3	Venusaur		GRASS	POISON	82	83	80	100	100	80	525
4	Venusaur	Mega Venusaur	GRASS	POISON	100	123	80	122	120	80	625
5	Charmander		FIRE		52	43	39	60	50	65	309
6	Charmeleon		FIRE		64	58	58	80	65	80	405
7	Charizard		FIRE	FLYING	84	78	78	109	85	100	534
8	Charizard	Mega Charizard X	FIRE	DRAGON	130	111	78	130	85	100	634
9	Charizard	Mega Charizard Y	FIRE	FLYING	104	78	78	159	115	100	634
10	Squirtle		WATER		48	65	44	50	64	43	314
11	Wartortle		WATER		63	80	59	65	80	58	405

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 :

a. 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.

```
> # To check inconsistencies in the 'Primary Type - Character' feature of the data set.
> unique(pokemon_dataset$Primary.Type)
 [1] "GRASS"    "FIRE"     "WATER"    "BUG"      "NORMAL"   "DARK"     "POISON"   "ELECTRIC" "GROUND"
[10] "ICE"      "FAIRY"    "STEEL"    "FIGHTING" "PSYCHIC"  "ROCK"     "GHOST"    "DRAGON"   "FLYING"
```

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

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

```
> # To check NaN, NULL values in the data set.
> sum(is.na(pokemon_dataset))
[1] 0
> sum(is.null(pokemon_dataset))
[1] 0
```

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

c. 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()***.

```
# 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)
```

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

d. 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.

```
# Check the duplicate values in a combination of 2 features.
duplicated(pokemon_dataset[,1:2])
 [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[12] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[23] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[34] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
[45] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[56] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

Figure 5: Check for Duplication in the combination of features from the data set.

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

```
> # Retrieving the duplicated records from the data set.
> pokemon_dataset[which(duplicated(pokemon_dataset[,1:2])),]
       Name   Name2 Primary.Type Secondary.type Attack Defense HP Sp.Attack Sp.Defense Speed Total
44 Nidoran♀ Unknown       POISON        Unknown     57      40 46        40         40    50   273
```

Figure 6: Retrieving the duplicate records from the data set.

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

```
# Eliminating the duplicated records using the indexes provided by the above step.
pokemon_dataset <- pokemon_dataset %>% filter( !row_number() %in% 44)
```

Figure 7: Eliminating the duplicate records from the data set.

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

```
> # Removing 'NA, Missing Values' from the data set.
> dataSet <- na.omit(dataSet)
```

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

h. 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 -

	vars	n	mean	sd	min	max	range	se	IQR	Q0.25	Q0.75
Name*	1	1045	447.99617	258.555303	1	897	896	7.9982561	437	226	663
Name2*	2	1045	14.95311	36.707379	1	165	164	1.1355212	0	1	1
Primary.Type*	3	1045	10.40191	5.520876	1	18	17	0.1707851	9	6	15
Secondary.type*	4	1045	6.04689	5.835443	1	19	18	0.1805160	9	1	10
Attack	5	1045	80.46699	32.413665	5	190	185	1.0026977	45	55	100
Defense	6	1045	74.66124	31.237903	5	250	245	0.9663261	40	50	90
HP	7	1045	70.06794	26.671411	1	255	254	0.8250644	32	50	82
Sp.Attack	8	1045	73.02201	32.724797	10	194	184	1.0123223	45	50	95
Sp.Defense	9	1045	72.28900	28.074148	20	250	230	0.8684572	40	50	90
Speed	10	1045	68.80766	30.210094	5	200	195	0.9345315	45	45	90
Total	11	1045	439.31483	121.970701	175	1125	950	3.7730918	185	330	515

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 -

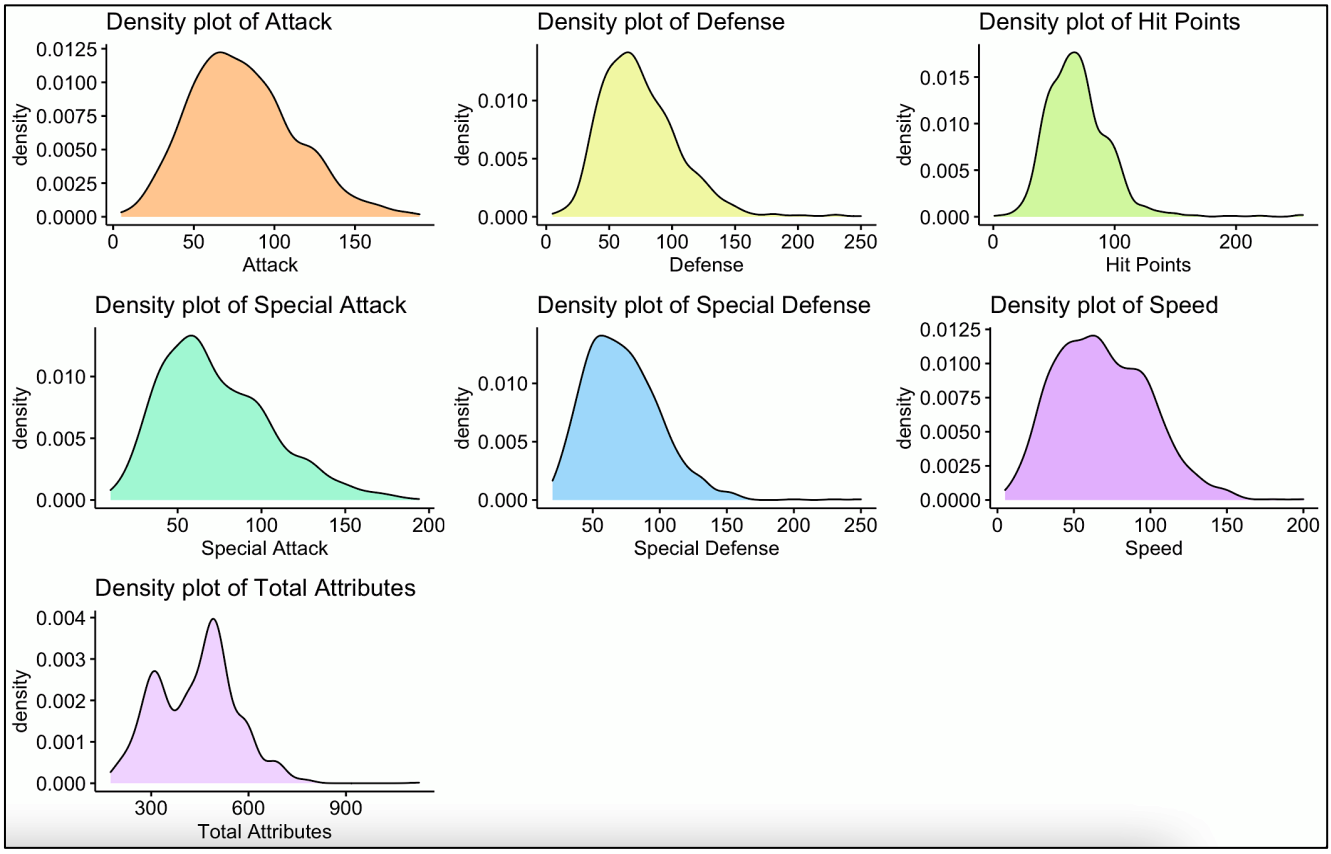

Figure 10: Density Plot of features of the data set.

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.

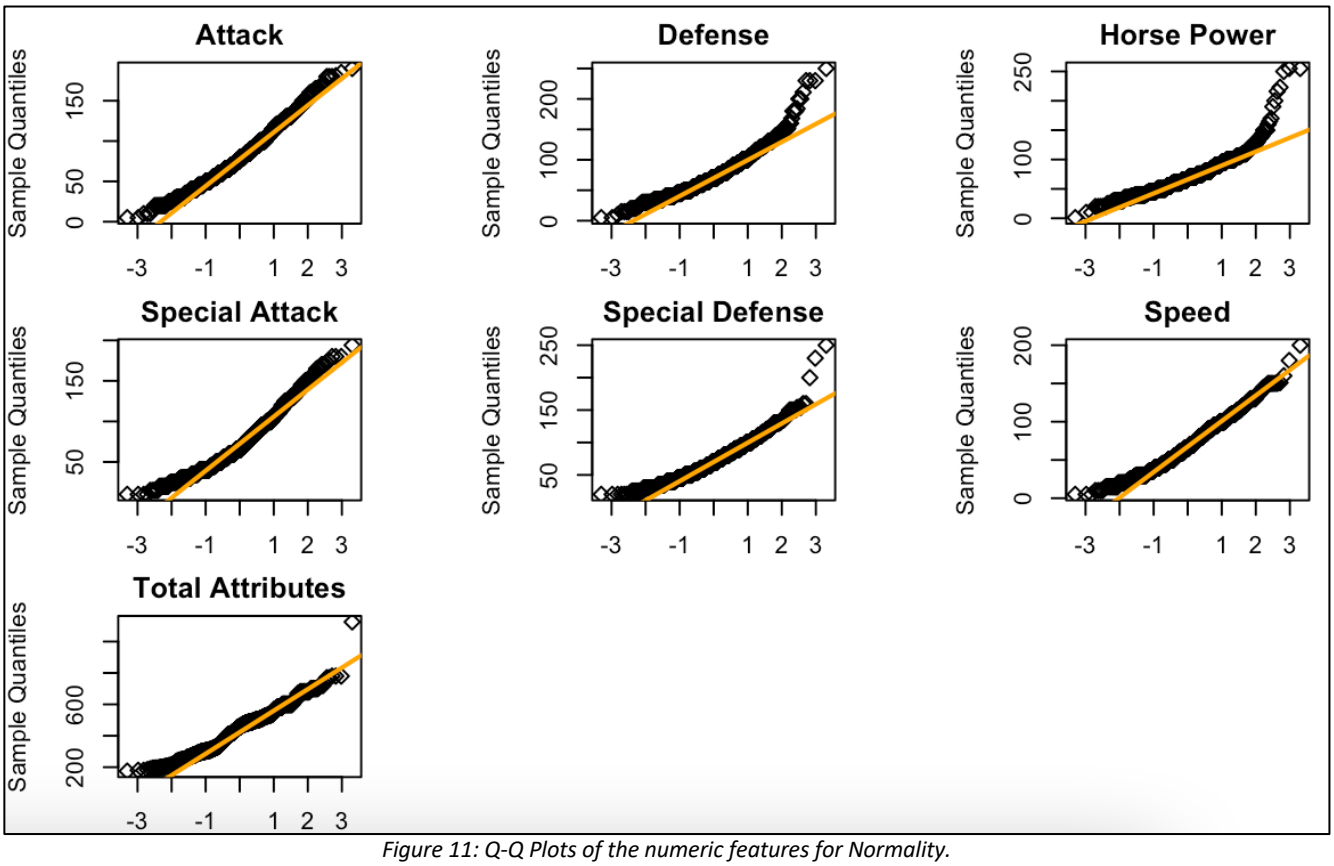

Figure 11: Q-Q Plots of the numeric features for Normality.

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.

```
# 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))
```

Figure 12: Describe the data set for descriptive statistics..

	vars	n	mean	sd	min	max	range	se	IQR	Q0.25	Q0.75
Name*	1	1044	447.92241	258.668222	1	897	896	8.0055805	437.50	225.75	663.25
Name2*	2	1044	121.38123	25.931732	1	165	164	0.8025670	0.00	129.00	129.00
Primary.Type*	3	1044	10.39847	5.522397	1	18	17	0.1709139	9.00	6.00	15.00
Secondary.type*	4	1044	11.77969	4.200321	1	19	18	0.1299967	6.00	8.00	14.00
Attack	5	1044	80.48946	32.421051	5	190	185	1.0034063	45.00	55.00	100.00
Defense	6	1044	74.69444	31.234423	5	250	245	0.9666811	40.00	50.00	90.00
HP	7	1044	70.09100	26.673775	1	255	254	0.8255326	32.25	50.00	82.25
Sp.Attack	8	1044	73.05364	32.724496	10	194	184	1.0127977	45.00	50.00	95.00
Sp.Defense	9	1044	72.31992	28.069786	20	250	230	0.8687381	40.00	50.00	90.00
Speed	10	1044	68.82567	30.218956	5	200	195	0.9352532	45.00	45.00	90.00
Total	11	1044	439.47414	121.920342	175	1125	950	3.7733399	185.00	330.00	515.00

Figure 13: Descriptive Statistics after cleaning the data set.

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

DESCRIPTIVE STATISTICS (BY GROUPING)

```
# 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)
```

Figure 14: Describe the data set with GROUPINGS (based on features)

```
 Descriptive statistics by group
group: BUG
                vars  n   mean     sd median trimmed    mad min max range  skew kurtosis    se
Name*              1 81  38.49  22.07     39   38.57  28.17   1  75    74 -0.02    -1.25  2.45
Name2*             2 81   4.95   0.74      5    5.00   0.00   1   8     7 -1.75    15.12  0.08
Primary.Type*      3 81   1.00   0.00      1    1.00   0.00   1   1     0   NaN      NaN  0.00
Secondary.type*    4 81   7.95   3.45      9    8.05   2.97   1  14    13 -0.24    -0.77  0.38
Attack             5 81  71.07  37.55     65   67.77  37.06  10 185   175  0.75     0.00  4.17
Defense            6 81  71.80  34.41     60   67.82  29.65  20 230   210  1.49     3.83  3.82
HP                 7 81  57.02  17.34     60   57.65  14.83   1 107   106 -0.32     0.60  1.93
Sp.Attack          8 81  56.38  29.37     53   53.22  25.20  10 145   135  0.98     0.64  3.26
Sp.Defense         9 81  65.07  31.40     60   62.98  29.65  20 230   210  1.74     7.24  3.49
Speed             10 81  63.26  33.91     60   60.46  35.58   5 160   155  0.73     0.11  3.77
Total             11 81 384.62 120.57    395  384.08 148.26 180 600   420 -0.09    -1.14 13.40
------------------------------------------------------------------------------
```

Figure 15: Descriptive Statistics by GROUPS (based on Primary Attack feature)

```
------------------------------------------------------------------------
group: DARK
                vars  n   mean     sd median trimmed    mad min max range  skew kurtosis    se
Name*              1 46  21.41  12.74   21.5   21.37  16.31   1  43    42  0.01    -1.30  1.88
Name2*             2 46   9.80   2.66   11.0   10.45   0.00   1  11    10 -2.07     3.00  0.39
Primary.Type*      3 46   1.00   0.00    1.0    1.00   0.00   1   1     0   NaN      NaN  0.00
Secondary.type*    4 46   7.04   3.44    8.5    7.21   3.71   1  12    11 -0.41    -1.30  0.51
Attack             5 46  81.93  28.36   85.0   81.45  29.65  28 150   122  0.26    -0.53  4.18
Defense            6 46  66.39  25.32   60.5   65.16  25.20  28 125    97  0.44    -0.81  3.73
HP                 7 46  70.07  31.63   65.0   66.21  22.24  30 223   193  2.48     9.45  4.66
Sp.Attack          8 46  71.09  30.72   65.0   68.92  29.65  25 140   115  0.58    -0.68  4.53
Sp.Defense         9 46  68.37  24.73   65.0   66.26  22.98  30 130   100  0.62    -0.33  3.65
Speed             10 46  76.70  25.82   71.5   76.63  30.39  20 125   105  0.09    -0.88  3.81
Total             11 46 434.54 115.73  443.0  435.55 111.19 220 680   460 -0.09    -0.94 17.06
------------------------------------------------------------------------
group: DRAGON
                vars  n   mean     sd median trimmed    mad min max range  skew kurtosis    se
Name*              1 41  17.32   9.24     19   17.58  10.38   1  31    30 -0.23    -1.20  1.44
Name2*             2 41   9.68   2.81     11   10.36   0.00   1  12    11 -1.86     2.14  0.44
Primary.Type*      3 41   1.00   0.00      1    1.00   0.00   1   1     0   NaN      NaN  0.00
Secondary.type*    4 41   7.10   2.34      7    7.33   2.97   1  10     9 -0.75    -0.34  0.37
Attack             5 41 107.02  32.84    100  105.73  37.06  50 180   130  0.32    -0.73  5.13
Defense            6 41  83.93  26.04     90   84.52  29.65  30 130   100 -0.17    -0.89  4.07
HP                 7 41  85.24  37.13     80   80.91  29.65  28 216   188  1.59     3.45  5.80
Sp.Attack          8 41  91.73  39.98     91   89.42  45.96  30 180   150  0.36    -0.92  6.24
Sp.Defense         9 41  84.10  30.01     90   83.42  29.65  30 150   120  0.09    -0.57  4.69
Speed             10 41  84.34  24.15     85   84.58  25.20  40 142   102 -0.10    -0.63  3.77
Total             11 41 536.37 147.31    600  543.42 148.26 270 780   510 -0.36    -1.21 23.01
------------------------------------------------------------------------
```

Figure 16: Descriptive Statistics by GROUPS (based on Primary Attack & Secondary Attack features)

a. The descriptive statistics of the data set was calculated after applying **GROUPINGs** based on *Primary Type* and *Secondary Type*.
b. There are 18 groups for which these descriptive statistics were calculated.
c. 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

d. 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

e. 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.

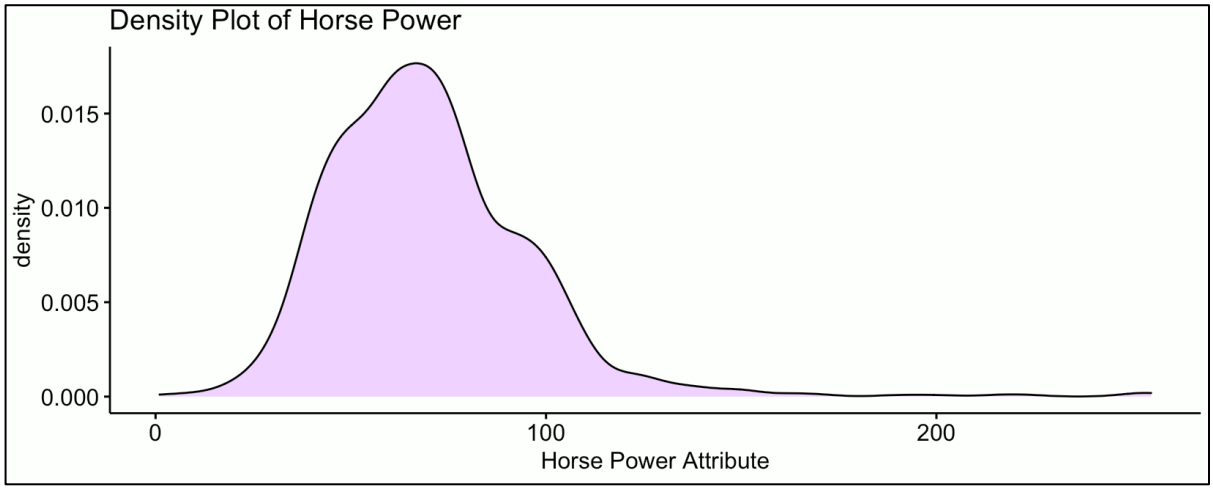

Figure 17: Density Plot of Horse Power attribute

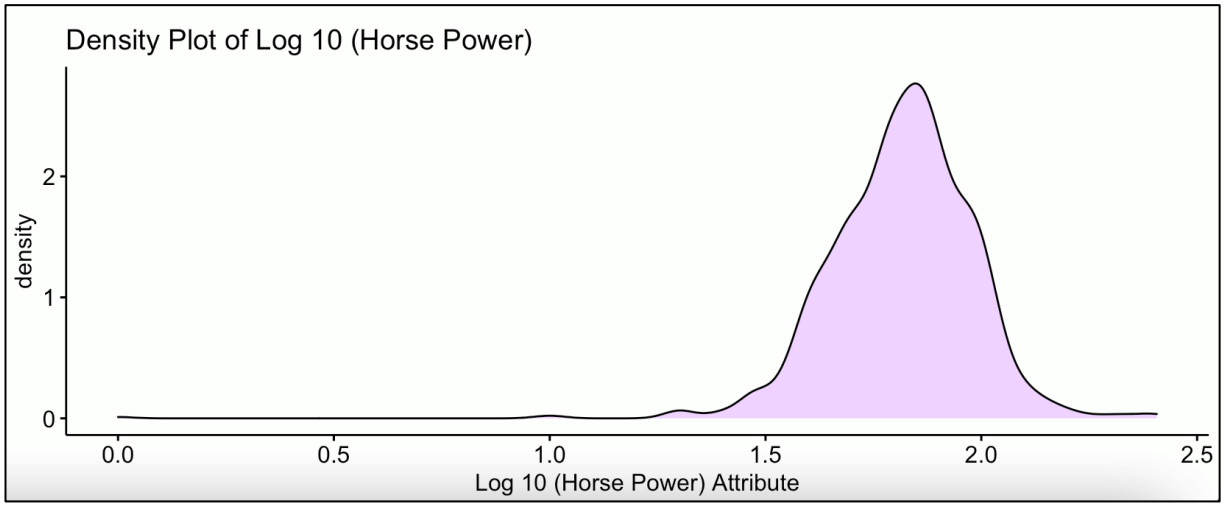

Figure 18: Density Plot of Logarithm 10 of Horse Power attribute

a. 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.
b. 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.

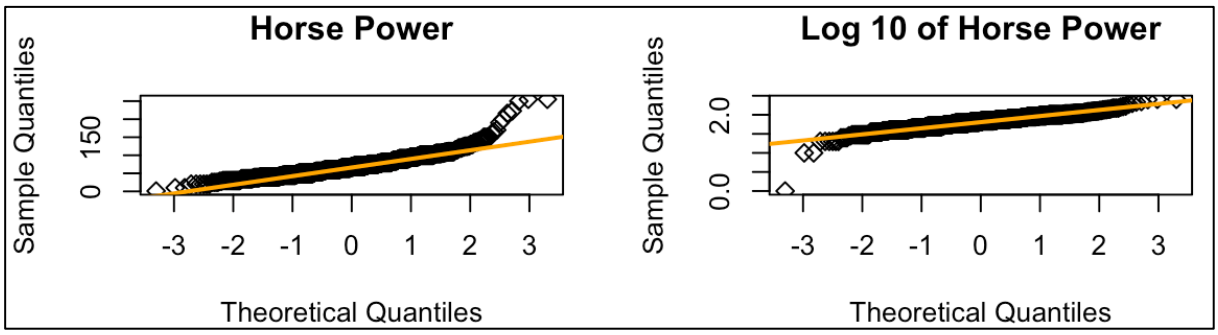

Figure 19: Q-Q Plots of normal and Logarithm 10 of Horse Power attribute

Correlation Chart and Table of the Attributes of Pokémons

We have plotted the Correlation Chart (Correlation Matrix) and a Correlation Table between the 6 attributes of Pokémons. The attributes are *Horse Power (HP), Attack, Sp.Attack, Defense, Sp.Defense, Speed*.

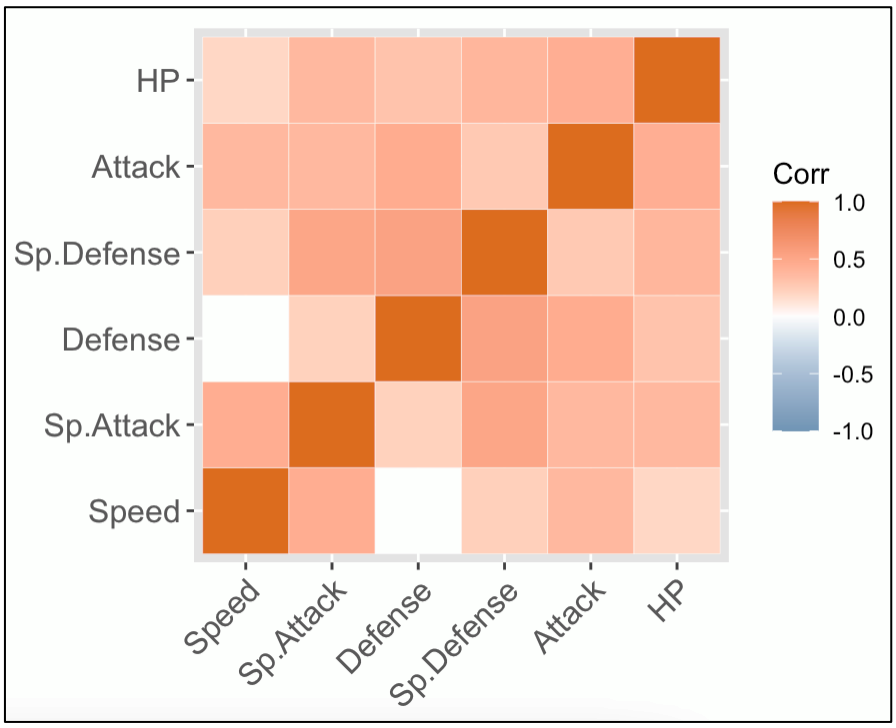

Figure 20: Correlation Chart (Matrix) of all 6 attributes of Pokémons.

	Attack	Defense	Speed	HP	Sp.Attack	Sp.Defense
Attack	1.0000000	0.457285137	0.373749403	0.4437831	0.3683233	0.2670709
Defense	0.4572851	1.000000000	0.004097065	0.2996335	0.2226480	0.5436334
Speed	0.3737494	0.004097065	1.000000000	0.1892755	0.4453380	0.2263604
HP	0.4437831	0.299633505	0.189275512	1.0000000	0.3709869	0.3927184
Sp.Attack	0.3683233	0.222648004	0.445337996	0.3709869	1.0000000	0.5114408
Sp.Defense	0.2670709	0.543633398	0.226360429	0.3927184	0.5114408	1.0000000

Figure 21: Correlation Table of all 6 attributes of Pokémons.

a. From the correlation chart and correlation table as well, we can figure out that all the 6 attributes belonging to the Pokémon world are **positively correlated to each other.**

b. The correlation between ***Attack and Horse Power (HP)*** is 0.44, which indicates that they are positively correlated and more Horse Power (HP) a Pokémon possess, the more Attack prowess it will have.

c. The correlation between **Defense and Special Defense (Sp. Defense)** is 0.54, which indicates a strong positive correlation between them and states that the more defensive power a Pokémon has, the more Special Defensive ability it will have.

REGRESSION ANALYSIS

We have plotted a scatterplot between the attributes *Defense & Special Defense (Sp.Defense)* to check the pattern of distribution along with a Linear Regression Model line in the infographic. The regression line has been plotted using **GEOM_SMOOTH()** function.

Afterwards, we have performed regression analysis between these attributes with *Special Defense* as the dependent variable (responding variable) and *Defense* as the independent variable.

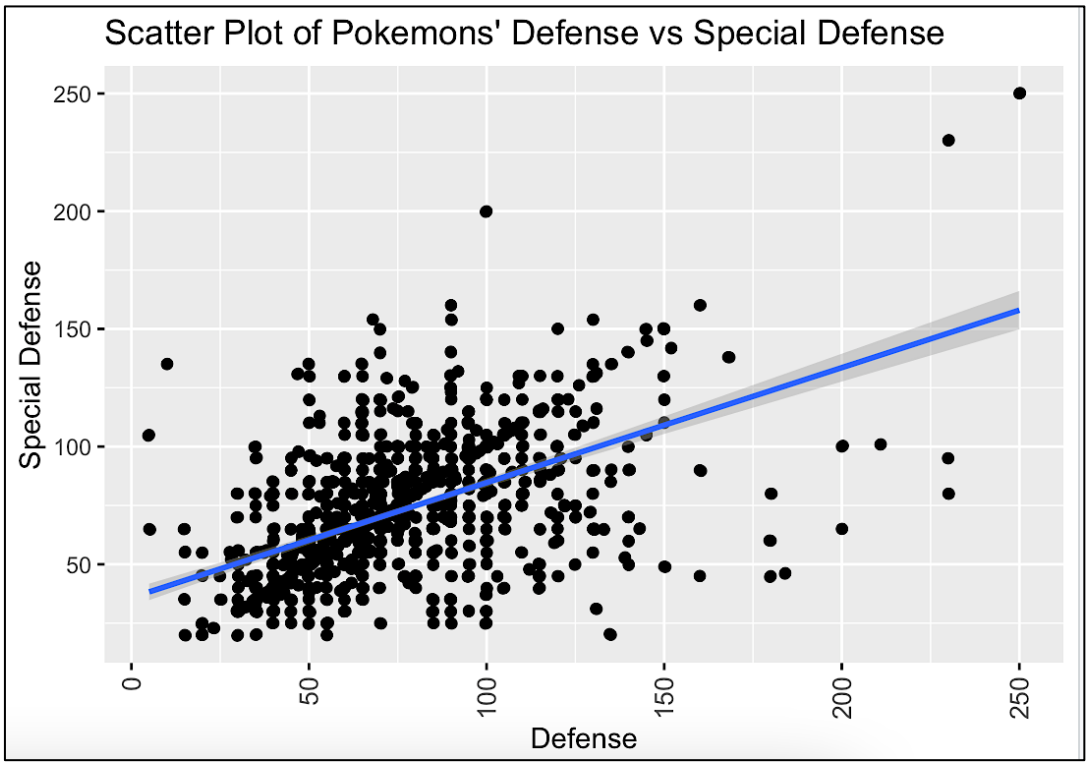

Figure 34: Scatterplot with regression line between 'Defense' & 'Special Defense' attributes of Pokémons.

```
> summary(lm(data = pokemon_dataset, Sp.Defense ~ Defense))

Call:
lm(formula = Sp.Defense ~ Defense, data = pokemon_dataset)

Residuals:
    Min      1Q  Median      3Q     Max
-81.782 -12.956  -2.244  12.645 115.317

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 35.82772    1.89170   18.94   <2e-16 ***
Defense      0.48855    0.02337   20.91   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 23.57 on 1042 degrees of freedom
Multiple R-squared:  0.2955,    Adjusted R-squared:  0.2949
F-statistic: 437.1 on 1 and 1042 DF,  p-value: < 2.2e-16
```

Figure 35: Regression Analysis with 'Special Defense' as responding variable.

a. From the Jittered scatterplot, we can figure out that both of these attributes are **positively correlated to each other.**

b. The **Coefficient of Determination (R-squared or R^2) is around 0.3, which signifies that the fit between these attributes is not very good.**

c. The R^2 value of 0.3 also indicates that only around 30% of the points pass through the line.

d. The Regression Model Equation for the above regression would become :
SPECIAL DEFENSE = 35.83 + (DEFENSE * 0.49)

e. In our case, the response variable (Special Defense) cannot be explained properly using the independent variable (Defense). Only 30% of the response variable can be explained using the attribute Defense.

f. We can also figure out from this regression analysis that a Pokémon will possess a Special Defensive ability at around 35 even if the defense prowess is 0 for it.

Attack vs Horse Power (HP)

We have plotted a scatterplot between the attributes *Attack & Horse Power (HP)* to check the pattern of distribution along with a Linear Regression Model line in the infographic using **GEOM_SMOOTH()** function. Also, we have performed regression analysis between these attributes with *Attack* as the dependent variable (responding variable) and *HP* as the independent variable.

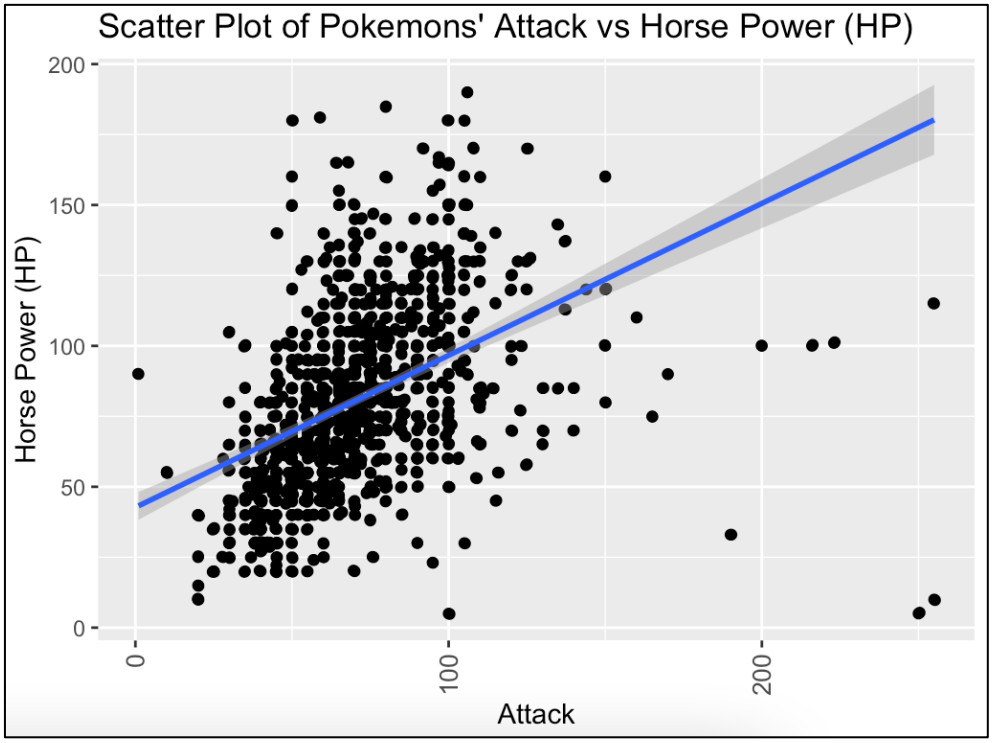

Figure 36: Scatterplot with regression line between 'Attack' & 'Horse Power' attributes of Pokémons.

```
> # Regression Analysis
> summary(lm(data = pokemon_dataset, Attack ~ HP))

Call:
lm(formula = Attack ~ HP, data = pokemon_dataset)

Residuals:
     Min       1Q   Median       3Q      Max
-172.533  -19.258   -3.137   16.863  110.348

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 42.68217    2.53039   16.87   <2e-16 ***
HP           0.53940    0.03374   15.99   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 29.07 on 1042 degrees of freedom
Multiple R-squared:  0.1969,    Adjusted R-squared:  0.1962
F-statistic: 255.5 on 1 and 1042 DF,  p-value: < 2.2e-16
```

Figure 37: Regression Analysis with 'Attack' as responding variable.

a. From the Jittered scatterplot, we can figure out that both of these attributes are **positively correlated to each other.**

b. The **Coefficient of Determination (R-squared or R^2) is around 0.2, which signifies that the fit between these attributes is not very good.**

c. The R^2 value of 0.2 also indicates that only around 20% of the points pass through the line.

d. The Regression Model Equation for the above regression would become :
ATTACK = 42.68 + (HORSE-POWER * 0.54)

e. In our case, the response variable (Attack) cannot be explained properly using the independent variable (HP). Only 20% of the response variable can be explained using the attribute HP.

f. We can also figure out from this regression analysis that a Pokémon will possess the Attack ability at around 43 even if it's not having any horse power in it.

MULTIPLE LINEAR REGRESSION ANALYSIS

We have plotted a scatterplot between the attributes *Defense & Special Defense (Sp.Defense)* with Multiple Linear Regression Lines based on *Primary Type* of Pokémons to check the pattern of distribution in the infographic. The regression line has been plotted using **GEOM_SMOOTH()** function.

We performed regression analysis on data set with *Special Defense* as the dependent variable (responding variable) and *Defense* as the independent variable with *Primary Type* grouping.

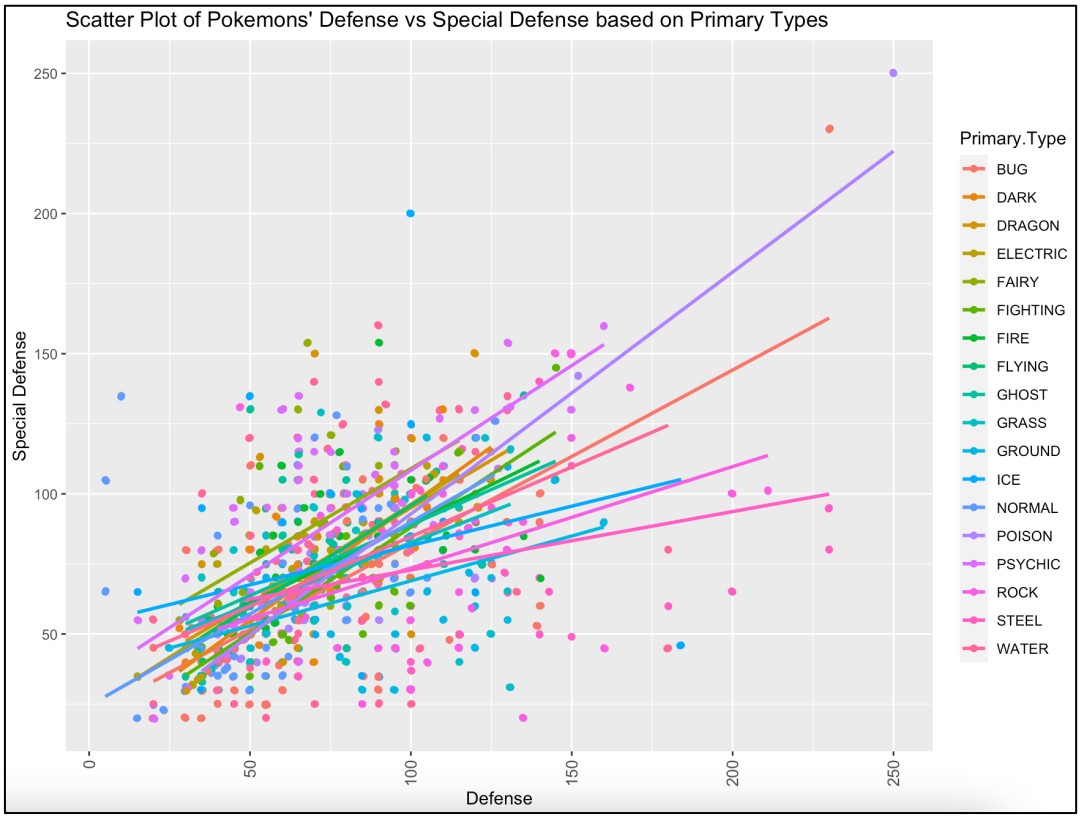

Figure 38: Scatterplot with regression line between 'Defense' & 'Special Defense' attributes of Pokémons.

```
> # Regression Analysis
> summary(lm(data = pokemon_dataset, Sp.Defense ~ Defense + Primary.Type))

Call:
lm(formula = Sp.Defense ~ Defense + Primary.Type, data = pokemon_dataset)

Residuals:
    Min      1Q  Median      3Q     Max
-88.337 -13.140  -2.203  12.087 111.231

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          26.12301    3.02677   8.631  < 2e-16 ***
Defense               0.54248    0.02396  22.644  < 2e-16 ***
Primary.TypeDARK      6.23091    4.14019   1.505  0.13264
Primary.TypeDRAGON   12.44632    4.30589   2.891  0.00393 **
Primary.TypeELECTRIC  9.60367    3.78465   2.538  0.01131 *
Primary.TypeFAIRY    24.78964    5.38888   4.600 4.75e-06 ***
```

Figure 39A: Regression Analysis with 'Special Defense' as responding variable.

```
Primary.TypeFLYING      8.95514    8.30872   1.078  0.28138
Primary.TypeGHOST       9.31432    4.26646   2.183  0.02925 *
Primary.TypeGRASS       4.96910    3.42395   1.451  0.14701
Primary.TypeGROUND     -8.16773    4.30853  -1.896  0.05828 .
Primary.TypeICE         8.39846    4.36864   1.922  0.05483 .
Primary.TypeNORMAL      5.17861    3.26130   1.588  0.11262
Primary.TypePOISON      5.30379    4.33385   1.224  0.22131
Primary.TypePSYCHIC    22.31691    3.54433   6.297 4.51e-10 ***
Primary.TypeROCK       -6.46750    3.86896  -1.672  0.09490 .
Primary.TypeSTEEL     -13.23622    4.61913  -2.866  0.00425 **
Primary.TypeWATER       5.60704    3.15481   1.777  0.07582 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 22.41 on 1025 degrees of freedom
Multiple R-squared:  0.3734,    Adjusted R-squared:  0.3624
F-statistic: 33.93 on 18 and 1025 DF,  p-value: < 2.2e-16
```

Figure 39B: Regression Analysis with 'Special Defense' as responding variable.

a. From the scatterplot, we can figure out that both of these attributes are **positively correlated to each other based on the Primary Types.**

b. The **Coefficient of Determination (R-squared or R^2) is around 0.37, which signifies that the fit between these attributes is not very good.**

c. The R^2 value of 0.37 also indicates that only around 37% of the points pass through the line.

d. The Regression Model Equation for the above regression would become :
SPECIAL DEFENSE = 26.12 + (DEFENSE * 0.54) + (Primary.Type_DARK * 6.23) + (Primary.Type_DRAGON * 12.44) + (Primary.Type_ELECTRIC * 9.60) + ...

e. In our case, the response variable (Special Defense) cannot be explained properly using the independent variable (Defense). Only 37% of the response variable can be explained using the attribute Defense.

f. We can also figure out from this regression analysis that a Pokémon will possess a Special Defensive ability at around 26 even if the defense prowess is 0 for it.

g. Since, there are 18 different levels of Primary Types, we wanted to first create dummy variables of this feature and perform the regression based on them. The total number of dummy variables will be *18 - 1 = 17.*

DUMMY VARIABLE & ANALYSIS USING THEM

Since we have 18 levels of categories in the attribute *Primary Type*, we have created Dummy Variables or Treatment Variables of this feature. Later, we have plotted a scatterplot between the attributes *Defense & Special Defense (Sp.Defense)* with a single Linear Regression Line based on *Primary Type - DARK* of Pokémons to check the pattern of distribution in the infographic. The dummy variables are created using **RECIPES** library.

We performed regression analysis on data set with *Special Defense* as the dependent variable and *Defense* as the independent variable with *Primary Type - DARK* grouped as factors.

```
# ----------- 1st Instance of creation of Dummy Variables ----------- #
pokemon_dataset_dummy <- pokemon_dataset %>% recipe(Total ~ .) %>%
          step_dummy(Primary.Type, one_hot = FALSE) %>%
          prep() %>% bake(pokemon_dataset)
```

Figure 40: Creation of dummy variables of attribute Primary.Type of Pokémons.

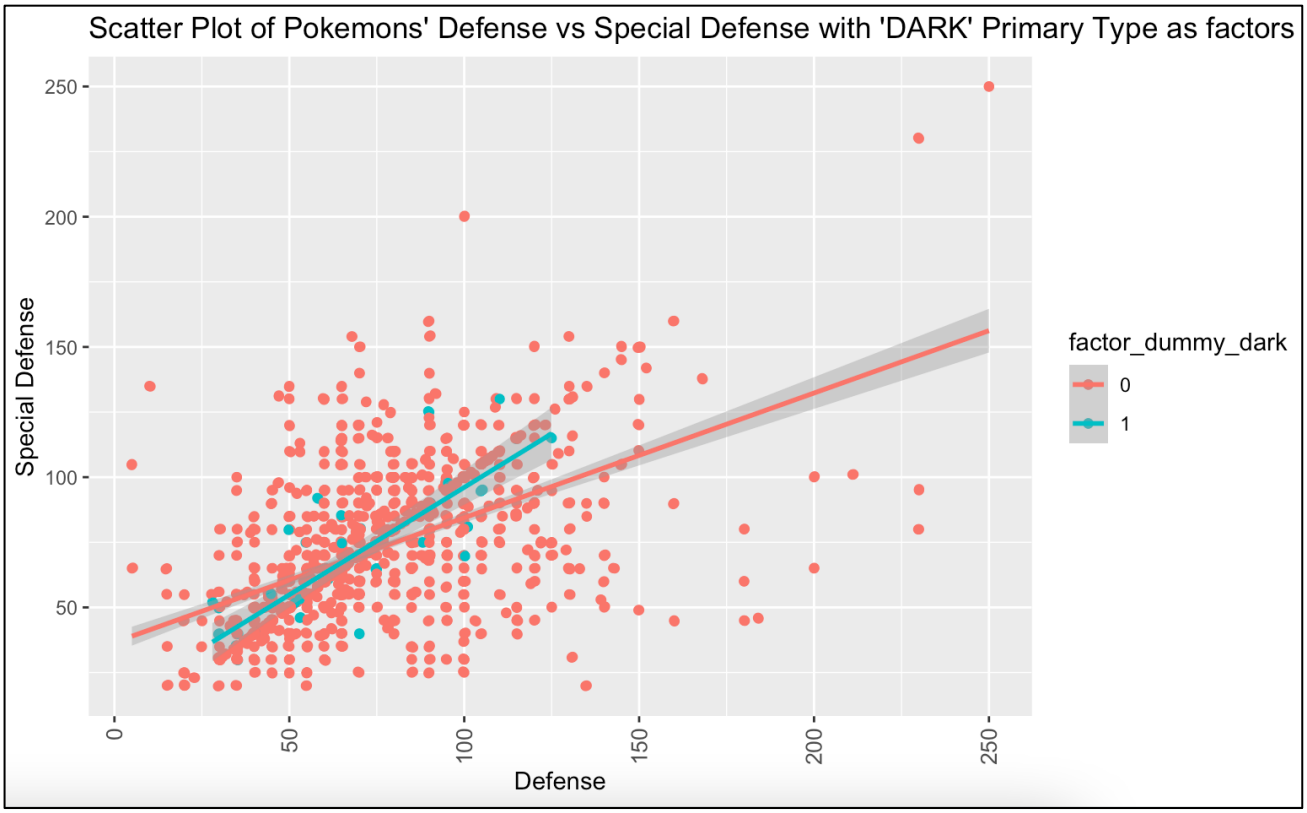

Figure 41: Scatterplot with regression line between 'Defense' & 'Special Defense' of Pokémons factored on DARK-Type.

```
lm(formula = Sp.Defense ~ Defense + Primary.Type_DARK + Primary.Type_DRAGON +
    Primary.Type_ELECTRIC + Primary.Type_FAIRY + Primary.Type_FIGHTING +
    Primary.Type_FIRE + Primary.Type_FLYING + Primary.Type_GHOST +
    Primary.Type_GRASS + Primary.Type_GROUND + Primary.Type_ICE +
    Primary.Type_NORMAL + Primary.Type_POISON + Primary.Type_PSYCHIC +
    Primary.Type_ROCK + Primary.Type_STEEL + Primary.Type_WATER,
    data = pokemon_dataset_dummy)
```

Figure 42A: Regression Analysis with 'Special Defense' along with dummy variables.

```
Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)           26.12301    3.02677   8.631  < 2e-16 ***
Defense                0.54248    0.02396  22.644  < 2e-16 ***
Primary.Type_DARK      6.23091    4.14019   1.505  0.13264
Primary.Type_DRAGON   12.44632    4.30589   2.891  0.00393 **
Primary.Type_ELECTRIC  9.60367    3.78465   2.538  0.01131 *
Primary.Type_FAIRY    24.78964    5.38888   4.600 4.75e-06 ***
Primary.Type_FIGHTING  2.12893    4.26222   0.499  0.61754
Primary.Type_FIRE      8.22256    3.73359   2.202  0.02786 *
Primary.Type_FLYING    8.95514    8.30872   1.078  0.28138
Primary.Type_GHOST     9.31432    4.26646   2.183  0.02925 *
Primary.Type_GRASS     4.96910    3.42395   1.451  0.14701
Primary.Type_GROUND   -8.16773    4.30853  -1.896  0.05828 .
Primary.Type_ICE       8.39846    4.36864   1.922  0.05483 .
Primary.Type_NORMAL    5.17861    3.26130   1.588  0.11262
Primary.Type_POISON    5.30379    4.33385   1.224  0.22131
Primary.Type_PSYCHIC  22.31691    3.54433   6.297 4.51e-10 ***
Primary.Type_ROCK     -6.46750    3.86896  -1.672  0.09490 .
Primary.Type_STEEL   -13.23622    4.61913  -2.866  0.00425 **
Primary.Type_WATER     5.60704    3.15481   1.777  0.07582 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 22.41 on 1025 degrees of freedom
Multiple R-squared:  0.3734,    Adjusted R-squared:  0.3624
F-statistic: 33.93 on 18 and 1025 DF,  p-value: < 2.2e-16
```

Figure 42B: Regression Analysis with 'Special Defense' and dummy variables.

a. From the scatterplot, we can figure out that both of these attributes are **positively correlated to each other based on the factors of Primary Type - DARK.**

b. The **Coefficient of Determination (R-squared or R^2) is around 0.37, which signifies that the fit between these attributes is not very good here as well and which corresponds to the Coefficient of Determination (R-squared) when taken the original feature as whole.**

c. The R^2 value of 0.37 also indicates that only around 37% of the points pass through the line. Moreover, only 37% of the response variable can be explained using the attribute Defense.

d. We could learn from here that R automatically factorises the categorical variables in regression.

e. The regression line corresponding to the factor of 1 is more positively correlated to attributes.

f. We can also figure out from this regression analysis that a Pokémon will possess a Special Defensive ability at around 26 even if the defense prowess is 0 for it.

Correlation Chart and Table of the Dummy Variables of Pokémons

We have plotted the Correlation Chart (Correlation Matrix) between the 5 attributes and Dummy variables of *Primary Type* attribute of Pokémons.

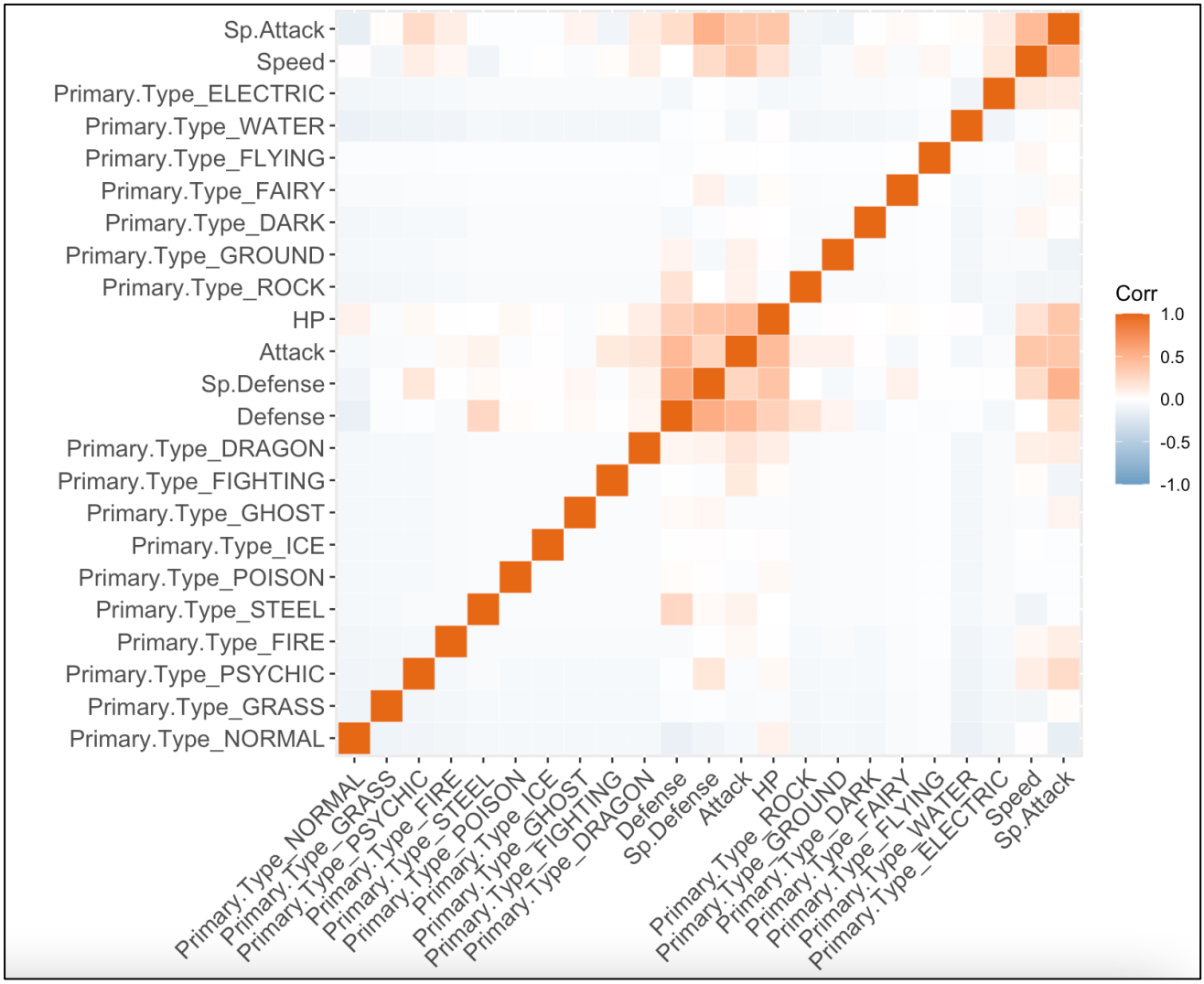

Figure 43: Correlation Chart (Matrix) of all 6 attributes of Pokémons.

a. From the correlation chart, we can figure out that all the 5 main (original) attributes belonging to the Pokémon world are **positively correlated to each other.**

b. But, some of the dummy variables of *Primary Type* features are either negatively correlated or close to being neutral. Rest of the dummy variables are positively correlated to other variables.

c. If we analyse further into this, we can figure out the **BEST SUBSET** of dummy variables which define the model properly and provides the best fit for our model.

SUBSET OF VARIABLE & CREATING DUMMY OF THIS SUBSET

Since we have 18 levels of categories in the attribute *Primary Type*, to take in consideration the conciseness of report and proper analysis, we have taken the Primary Types of Pokémons belonging to the **3 out of 4 Elements of Nature (Water, Fire, Earth).** We have created Dummy Variables or Treatment Variables of this subset of feature and later have plotted a scatterplot between the attributes *Defense & Special Defense (Sp.Defense)* with multiple Linear Regression Lines based on factors of these subsetted dummy variables.

```
# ------------- Subset Creation (Based on 3 out of 4 Elements of Matter) ------------ #
subset_pokemon_dataset_water <- subset(pokemon_dataset,
       Primary.Type == "WATER" | Primary.Type == "ICE")
subset_pokemon_dataset_fire <- subset(pokemon_dataset,
      Primary.Type == "FIRE" | Primary.Type == "DRAGON")
subset_pokemon_dataset_earth <- subset(pokemon_dataset,
      Primary.Type == "GROUND" | Primary.Type == "GRASS" | Primary.Type == "ROCK")
```

Figure 44: Creation of dummy variables of attribute Primary.Type of Pokémons.

```
# ------------- Dummy Variable Creation (Based on Subsetted 3 out of 4 Elements of Matter) ------------ #
subset_pokemon_dataset$FIRE_TYPE <- ifelse(
  subset_pokemon_dataset$Primary.Type == "FIRE" | subset_pokemon_dataset$Primary.Type == "DRAGON", 1, 0)

subset_pokemon_dataset$EARTH_TYPE <- ifelse(
  subset_pokemon_dataset$Primary.Type == "GROUND" | subset_pokemon_dataset$Primary.Type == "GRASS" |
  subset_pokemon_dataset$Primary.Type == "ROCK", 1, 0)
```

Figure 45: Scatterplot with regression line between 'Defense' & 'Special Defense' of Pokémons factored on DARK-Type.

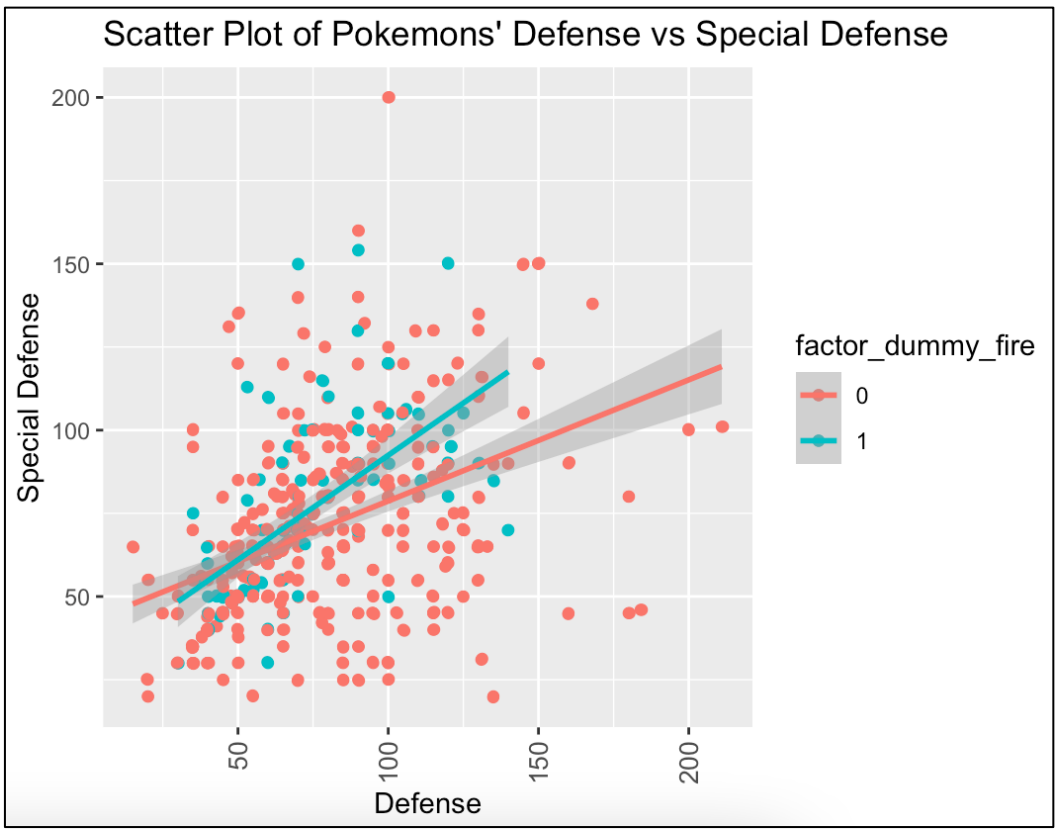

Figure 46: Regression Analysis with 'Special Defense' along with dummy variables.

```
Call:
lm(formula = Sp.Defense ~ Defense + FIRE_TYPE, data = subset_pokemon_dataset)

Residuals:
    Min      1Q  Median      3Q     Max
-73.920 -14.876  -1.561  13.589 120.351

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 38.87241    3.15916  12.305  < 2e-16 ***
Defense      0.40776    0.03696  11.031  < 2e-16 ***
FIRE_TYPE    7.10615    2.68981   2.642  0.00852 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 24.34 on 468 degrees of freedom
Multiple R-squared:  0.2119,    Adjusted R-squared:  0.2086
F-statistic: 62.93 on 2 and 468 DF,  p-value: < 2.2e-16
```

Figure 47B: Regression Analysis with 'Special Defense' and dummy variables.

```
Call:
lm(formula = Sp.Defense ~ Defense + EARTH_TYPE, data = subset_pokemon_dataset)

Residuals:
    Min      1Q  Median      3Q     Max
-74.221 -14.238  -1.782  11.929 115.218

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 42.59239    3.09102   13.78  < 2e-16 ***
Defense      0.42189    0.03716   11.35  < 2e-16 ***
EARTH_TYPE  -7.88256    2.29810   -3.43 0.000657 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 24.22 on 468 degrees of freedom
Multiple R-squared:  0.2198,    Adjusted R-squared:  0.2165
F-statistic: 65.92 on 2 and 468 DF,  p-value: < 2.2e-16
```

Figure 47B: Regression Analysis with 'Special Defense' and dummy variables.

a. From the scatterplot, we can figure out that both of these attributes are **positively correlated to each other based on the subset of FIRE - ELEMENT type of Pokémons.**

b. The **Coefficient of Determination (R-squared or R^2) is around 0.21 in case of FIRE-Element type and 0.22 in case of EARTH-Element type, which signifies that the fit between these attributes is not very good.**

c. The R^2 value of 0.21 also indicates that only around 21% of the points pass through the line. Moreover, only 21% of the response variable can be explained using the attributes Defense and FIRE_TYPE elements in the data set.

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.
- From the correlation chart and table, we can figure out that all the 6 attributes belonging to the Pokémon world are **positively correlated to each other.**
- We performed Regression Analysis to predict the value of dependent variable 'Attack' in correlation with its different independent variables (original, dummy, subsetted, or dummies of this subsetted variable)
- We created dummy variables of the attribute *Primary Type* which gave out 18 new dummy features for us and perform regression analysis using these dummy variables. We got the R^2 value of 0.37 indicates that only around 37% of the points pass through the line and only 37% of the response variable can be explained using the attribute Defense.
- A Correlation chart was plotted along with these dummy variables which will help us to figure out the **BEST SUBSET** of dummy variables that define the model properly and provides the best fit for our model.
- We made a subset of elements based on *Primary Type* attribute of the dataset. We took out only those Pokémons who belong to the **3 Elements out of 4 of Nature (Water, Fire, Earth)** and used them to perform Regression Analysis.
- The **Coefficient of Determination (R-squared or R^2) is around 0.21** in case of **FIRE-Element type** and **0.22** in case of **EARTH-Element type**, which signifies that the fit between these attributes is not very good.
- The Regression Model Equation for the above regression :
  SPECIAL DEFENSE = 26.12 + (DEFENSE * 0.54) + (Primary.Type_DARK * 6.23) + (Primary.Type_DRAGON * 12.44) + (Primary.Type_ELECTRIC * 9.60) + ...

BIBLIOGRAPHY

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

2. *A Grammar of Data Manipulation*. (2021). Dplyr. https://dplyr.tidyverse.org/

3. *Home - RDocumentation*. (2021). Functions in R - Documentation. https://www.rdocumentation.org/

4. Z. (2021e, February 2). *How to Create Dummy Variables in R (Step-by-Step)*. Statology. https://www.statology.org/dummy-variables-in-r/

5. *Quick-R: Subsetting Data*. (2021). Subsetting Data in R. https://www.statmethods.net/management/subset.html

6. Z. (2021e, May 18). *How to Read and Interpret a Regression Table*. Statology. https://www.statology.org/read-interpret-regression-table/

7. Marsja, E. (2021, April 15). *How to Create Dummy Variables in R (with Examples)*. Erik Marsja. https://www.marsja.se/create-dummy-variables-in-r/

APPENDIX

```
#---------------------- Week_6_Module_6_R-Script ----------------------#

print("Author : Harshit Gaur")
print("Week 6 Assignment - Module 6 R Pratice")

# Importing the packages.
listOfPackages <- c(
  "dplyr", "tidyr", "plyr", "tidyverse", "RColorBrewer", "plotrix", "scales", "ggplot2",
  "data.table", "reshape", "gridExtra", "vtable", "moments", "ggpubr", "psych", "GGally",
  "ggcorrplot",
  "recipes"
)

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 6/R Practice/")
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_tot
al)

# Check Normality using Shapiro-Wilks Test
format(shapiro.test(pokemon_dataset$Attack)$p.value, scientific = FALSE)
format(shapiro.test(pokemon_dataset$Defense)$p.value, scientific = FALSE)
format(shapiro.test(pokemon_dataset$HP)$p.value, scientific = FALSE)
format(shapiro.test(pokemon_dataset$Sp.Attack)$p.value, scientific = FALSE)
format(shapiro.test(pokemon_dataset$Sp.Defense)$p.value, scientific = FALSE)
format(shapiro.test(pokemon_dataset$Speed)$p.value, scientific = FALSE)
format(shapiro.test(pokemon_dataset$Total)$p.value, scientific = FALSE)
format(shapiro.test(log10(pokemon_dataset$Attack))$p.value, scientific = FALSE)

#-------- 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)


# ----------- Checking Normality of Features ----------- #

# Changing Plot Matrix Size to 3x3.
par(mfrow = c(2,2))

# Check Normality using Q-Q Plot of 'HP' and Log 10 of 'HP' Feature.
qq_plot(pokemon_dataset$HP, "Horse Power")

hp_log10 <- log10(pokemon_dataset$HP)
qq_plot(hp_log10, "Log 10 of Horse Power")

# Changing Plot Matrix Size to 3x3.
par(mfrow = c(1,1))

gd1 <- ggdensity(pokemon_dataset$HP, main = "Density Plot of Horse Power", xlab = "Horse Power Attribute", fill = "#eabdff")
gd2 <- ggdensity(hp_log10, main = "Density Plot of Log 10 (Horse Power)", xlab = "Log 10 (Horse Power) Attribute", fill = "#eabdff")

grid.arrange(gd1, gd2)


# ------------- Correlation ------------ #

data_corr <- pokemon_dataset[, c("Attack", "Defense", "Speed", "HP", "Sp.Attack", "Sp.Defense")]
corr <- round(cor(data_corr), 2)

ggcorrplot(corr, hc.order = TRUE, outline.col = "WHITE", ggtheme = ggplot2::theme_gray, colors = c("#6D9EC1", "WHITE",
"#E46726"))

# Correlation Matrix (Table) of 6 attributes of the data set.
View(cor(data_corr))


# ----------- One Sample Testing (t-Test) ----------- #

# Means of 'Attributes' according to Primary Type feature
groupPokemons <- pokemon_dataset %>% dplyr::group_by(pokemon_dataset$Primary.Type) %>% dplyr::summarise(
  mean_attack = mean(Attack),
  mean_defense = mean(Defense),
  mean_hp = mean(HP),
  mean_spAttack = mean(Sp.Attack),
  mean_spDefense = mean(Sp.Defense),
  mean_speed = mean(Speed),
  mean_total = mean(Total),
)

mean_Attack_WaterPokemon <- groupPokemons$mean_attack[groupPokemons$`pokemon_dataset$Primary.Type` == "WATER"]
mean_HP_WaterPokemon <- groupPokemons$mean_hp[groupPokemons$`pokemon_dataset$Primary.Type` == "WATER"]
mean_HP_PsychicPokemon <- groupPokemons$mean_hp[groupPokemons$`pokemon_dataset$Primary.Type` == "PSYCHIC"]
View(describe(groupPokemons))

# One-Sample t-test for 'Mean of Attack of Water type Pokemon'
t.test(pokemon_dataset$Attack, mu = mean_Attack_WaterPokemon, alternative = "greater", conf.level = .95)

# One-Sample t-test for 'Mean of Horse Power of Water type Pokemon'
t.test(pokemon_dataset$HP, mu = mean_HP_WaterPokemon, alternative = "greater")

# One-Sample t-test for 'Mean of Horse Power of Psychic type Pokemon'
t.test(pokemon_dataset$HP, mu = mean_HP_PsychicPokemon, alternative = "less")
```

```
# ---------- Two Sample Testing (t-Test) ---------- #

# Two-Sample t-test for 'Mean of Attack of Water vs Psychic type Pokemon'
t.test(pokemon_dataset$Attack[pokemon_dataset$Primary.Type == "WATER"],
pokemon_dataset$Attack[pokemon_dataset$Primary.Type == "PSYCHIC"], alternative = "two.sided")

# Two-Sample t-test for 'Mean of Special Attack of Water vs Psychic type Pokemon'
t.test(pokemon_dataset$Sp.Attack[pokemon_dataset$Primary.Type == "WATER"],
pokemon_dataset$Sp.Attack[pokemon_dataset$Primary.Type == "PSYCHIC"], alternative = "two.sided")


# ---------- Hypothesis Testing (P-Test) ---------- #

# Degree of Freedom
deg_of_freedom <- nrow(pokemon_dataset) - 1

# Population Mean (X Bar)
population_mean <- mean(pokemon_dataset$Attack)

# Standard Deviation of Sample (SD)
sd <- sd(pokemon_dataset$Attack)

# Test Statistics (t-score)
t_score <- (population_mean - mean_Attack_WaterPokemon) / (sd / sqrt(nrow(pokemon_dataset)))

# Hypothesis Testing (p-test)
pt(t_score, deg_of_freedom, lower.tail = FALSE)


# ---------- Regression Analysis ---------- #

# Make a ggplot (Scatter plot) of variables 'Defense' and 'Special Defense'.
ggplot(data = pokemon_dataset, aes(x = Defense, y = Sp.Defense)) +
  geom_point() +
  geom_jitter() +
  labs(title = "Scatter Plot of Pokemons' Defense vs Special Defense", x = "Defense", y = 'Special Defense') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 10)) +
  geom_smooth(method = "lm")

# Regression Analysis
summary(lm(data = pokemon_dataset, Sp.Defense ~ Defense))


# Make a ggplot (Scatter plot) of variables 'Attack' and 'HP'.
ggplot(data = pokemon_dataset, aes(x = HP, y = Attack)) +
  geom_point() +
  geom_jitter() +
  labs(title = "Scatter Plot of Pokemons' Attack vs Horse Power (HP)", x = "Attack", y = 'Horse Power (HP)') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 10)) +
  geom_smooth(method = "lm")

# Regression Analysis
summary(lm(data = pokemon_dataset, Attack ~ HP))


# ---------- Concept of Dummy Variables / Treatment Variables ---------- #

# Make a ggplot (Scatter plot) of variables 'Defense' and 'Special Defense' Colored with 'Primary Type'
ggplot(data = pokemon_dataset, aes(x = Defense, y = Sp.Defense, color = Primary.Type)) +
  geom_point() +
  labs(title = "Scatter Plot of Pokemons' Defense vs Special Defense based on Primary Types", x = "Defense", y = 'Special Defense') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 10)) +
  geom_smooth(method = "lm", se = FALSE)

# Regression Analysis
summary(lm(data = pokemon_dataset, Sp.Defense ~ Defense + Primary.Type))


# ---------- 1st Instance of creation of Dummy Variables ---------- #
```

```
pokemon_dataset_dummy <- pokemon_dataset %>% recipe(Total ~ .) %>%
        step_dummy(Primary.Type, one_hot = FALSE) %>%
        prep() %>% bake(pokemon_dataset)

factor_dummy_dark <- as.factor(pokemon_dataset_dummy$Primary.Type_DARK)
# Make a ggplot (Scatter plot) of variables 'Defense' and 'Special Defense'.
ggplot(data = pokemon_dataset_dummy, aes(x = Defense, y = Sp.Defense, color = factor_dummy_dark)) +
  geom_point() +
  geom_jitter() +
  labs(title = "Scatter Plot of Pokemons' Defense vs Special Defense with 'DARK' Primary Type as factors", x = "Defense", y = 'Special
Defense') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 10)) +
  geom_smooth(method = "lm")

# Regression Analysis
summary(lm(data = pokemon_dataset_dummy, Sp.Defense ~ Defense + Primary.Type_DARK + Primary.Type_DRAGON
        + Primary.Type_ELECTRIC + Primary.Type_FAIRY + Primary.Type_FIGHTING + Primary.Type_FIRE
        + Primary.Type_FLYING + Primary.Type_GHOST + Primary.Type_GRASS + Primary.Type_GROUND
        + Primary.Type_ICE + Primary.Type_NORMAL + Primary.Type_POISON + Primary.Type_PSYCHIC
        + Primary.Type_ROCK + Primary.Type_STEEL + Primary.Type_WATER))


# ------------- Correlation ------------ #

data_corr <- pokemon_dataset_dummy[, c("Attack", "Defense", "Speed", "HP", "Sp.Attack", "Sp.Defense", "Primary.Type_DARK",
"Primary.Type_DRAGON",
                                       "Primary.Type_ELECTRIC", "Primary.Type_FAIRY", "Primary.Type_FIGHTING", "Primary.Type_FIRE",
                                       "Primary.Type_FLYING", "Primary.Type_GHOST", "Primary.Type_GRASS", "Primary.Type_GROUND",
                                       "Primary.Type_ICE", "Primary.Type_NORMAL", "Primary.Type_POISON", "Primary.Type_PSYCHIC",
                                       "Primary.Type_ROCK", "Primary.Type_STEEL", "Primary.Type_WATER")]
corr <- round(cor(data_corr), 2)

ggcorrplot(corr, hc.order = TRUE, outline.col = "WHITE", ggtheme = ggplot2::theme_gray, colors = c("#6D9EC1", "WHITE",
"#E46726"))

# Correlation Matrix (Table) of 6 attributes of the data set.
View(cor(data_corr))


# ------------- Subset Creation (Based on 3 out of 4 Elements of Matter) ------------ #
subset_pokemon_dataset_water <- subset(pokemon_dataset, Primary.Type == "WATER" | Primary.Type == "ICE")
subset_pokemon_dataset_fire <- subset(pokemon_dataset, Primary.Type == "FIRE" | Primary.Type == "DRAGON")
subset_pokemon_dataset_earth <- subset(pokemon_dataset, Primary.Type == "GROUND" | Primary.Type == "GRASS" | Primary.Type
== "ROCK")

subset_pokemon_dataset <- rbind(subset_pokemon_dataset_water, subset_pokemon_dataset_fire, subset_pokemon_dataset_earth)

# ------------- Dummy Variable Creation (Based on Subsetted 3 out of 4 Elements of Matter) ------------ #
subset_pokemon_dataset$FIRE_TYPE <- ifelse(subset_pokemon_dataset$Primary.Type == "FIRE" |
subset_pokemon_dataset$Primary.Type == "DRAGON", 1, 0)
subset_pokemon_dataset$EARTH_TYPE <- ifelse(subset_pokemon_dataset$Primary.Type == "GROUND" |
subset_pokemon_dataset$Primary.Type == "GRASS" | subset_pokemon_dataset$Primary.Type == "ROCK", 1, 0)

# Make a ggplot (Scatter plot) of variables 'Defense' and 'Special Defense'.
ggplot(data = subset_pokemon_dataset, aes(x = Defense, y = Sp.Defense, color = Primary.Type)) +
  geom_point() +
  geom_jitter() +
  labs(title = "Scatter Plot of Pokemons' Defense vs Special Defense", x = "Defense", y = 'Special Defense') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 10)) +
  geom_smooth(method = "lm", se = FALSE)

# Regression Analysis
summary(lm(data = subset_pokemon_dataset, Sp.Defense ~ Defense + FIRE_TYPE + EARTH_TYPE))


# ------------- Part 2 (Separate Subset Regression) ------------ #

factor_dummy_fire <- as.factor(subset_pokemon_dataset$FIRE_TYPE)
# Make a ggplot (Scatter plot) of variables 'Defense' and 'Special Defense'.
ggplot(data = subset_pokemon_dataset, aes(x = Defense, y = Sp.Defense, color = factor_dummy_fire)) +
  geom_point() +
```

```
  geom_jitter() +
  labs(title = "Scatter Plot of Pokemons' Defense vs Special Defense", x = "Defense", y = 'Special Defense') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 10)) +
  geom_smooth(method = "lm", se = TRUE)

# Regression Analysis
summary(lm(data = subset_pokemon_dataset, Sp.Defense ~ Defense + FIRE_TYPE))


factor_dummy_earth <- as.factor(subset_pokemon_dataset$EARTH_TYPE)
# Make a ggplot (Scatter plot) of variables 'Defense' and 'Special Defense' with 'EARTH TYPE' as grouping.
ggplot(data = subset_pokemon_dataset, aes(x = Defense, y = Sp.Defense, color = factor_dummy_earth)) +
  geom_point() +
  labs(title = "Scatter Plot of Pokemons' Defense vs Special Defense", x = "Defense", y = 'Special Defense') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 10)) +
  geom_smooth(method = "lm", se = FALSE)

# Regression Analysis
summary(lm(data = subset_pokemon_dataset, Sp.Defense ~ Defense + EARTH_TYPE))
```