Arş. Gör. Kübra ADALI BLM3590 Statistical Data Analysis

**Deadline: 08.01.2023 23:59**

# Statistical Data Analysis Project

OĞUZHAN TOPALOĞLU
Ç19052025 – Group 1

*Computer Engineering,*
*Faculty of Electrical-Electronics,*
*Yıldız Technical University*

Istanbul, 2023

## 1. General Information

In this project, I have written 2 files (main.py, my_functions.py) to do the requirements of the project. my_functions.py file contains all of the statistical functions and main.py file uses these functions to calculate, plot and print the results. main.py file is made of 6 sections which are named as section 0,1,2,3,4 and 5. In this report I will explain the details of these functions and give the plotted graphs in the appendix.

## 2. Section 0 & 1

Section 0, simply prints the project's and my name to the console along with my student ID. Section 1, imports the dataset that was downloaded from https://www.kaggle.com/datasets/abcsds/pokemon and named "Pokemons.csv". I also remove all rows that contain null values in any column as preprocessing. This means that I will be only investigating relationships between pokemons that have only 1 type rather that 2.

## 3. Section 2

In this section I create 4 plots.

The first plot is a scatter plot and it shows the relationship between the Total and HP attributes of the pokemons. It seems like it has a positive linear relationship but there's a lot of noise and few outliers.

The second plot is a bar plot that shows the count of Pokemons in each generation. It seems like generation 1,3 and 5 has the most number of pokemons in video games while generation 6 is having the least number of pokemons.

The third plot is a line plot that shows the Total attribute of pokemons as the generations change. It seems like pokemons are the strongest when they're generation 4 and the weakest when they're generation 2. And it also seems like the smallest change it Total attribute happens between generation 5 and 6.

The fourth plot is box plot showing the minimum, maximum, quartile 1-3 and interquartile range of the Total attribute.

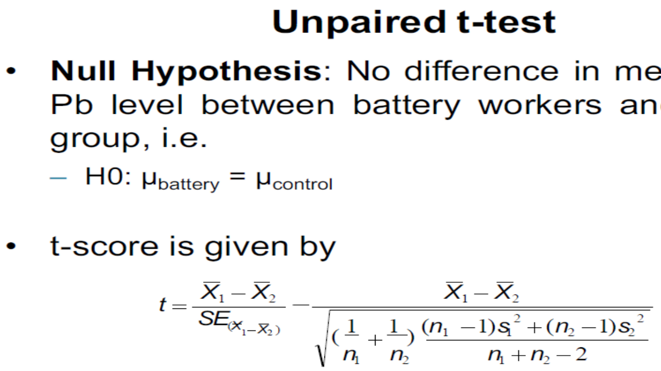
## 4. Section 3

In this section I calculate the descriptive statistics (min, max, count, range, mean, mode, standard deviation, coefficient of variance, quartile 1-3 and the interquartile range) of each numerical column and print them to the console using a package called tabular to make it look better. While calculating the quartiles, I also check if the quartile index is an integer or not. For example, if I get the index as 3.62, that means that the quartile should be equal to "$x_3 + (x_4 - x_3) * 0.62$" rather than just being equal to $x_3$. This is called linear interpolation and it’s necessary to get more accurate results while calculating quartiles.

## 5. Section 4

In this section I do 3 t tests using the do_t_test_for function. The first test shows that the difference in means between the Total stat of Generation 1 and Generation 3 Pokemons is not statistically significant. The second test shows that the difference in means between the Defense stat of Fire-type and Water-type pokemons is not statistically significant. The third test shows that the difference in means between the Attack stat of Generation 1 and Generation 2 Pokemons is not statistically significant.

In the do_t_test_for function I used this formula we learnt during class:



**Unpaired t-test**

- **Null Hypothesis**: No difference in mean blood Pb level between battery workers and control group, i.e.
  - H0: $\mu_{battery} = \mu_{control}$
- t-score is given by

$$t = \frac{\overline{X}_1 - \overline{X}_2}{SE_{(\overline{X}_1 - \overline{X}_2)}} = \frac{\overline{X}_1 - \overline{X}_2}{\sqrt{(\frac{1}{n_1} + \frac{1}{n_2})\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}}}$$

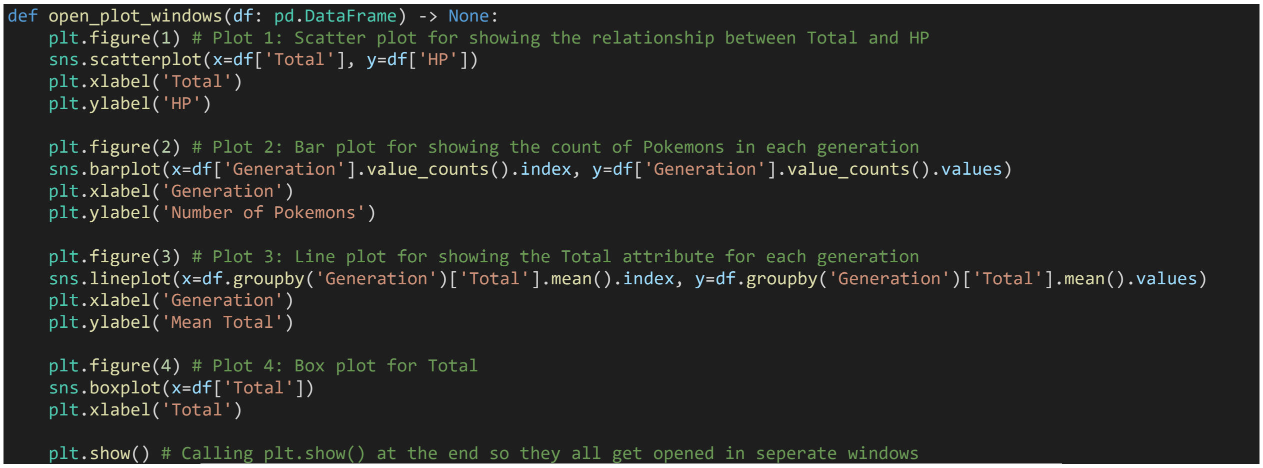
## 6. Section 5

The Pearson correlation coefficient is a measure of the linear relationship between two continuous variables. It can range from -1 (perfect negative correlation) to 1 (perfect positive correlation), with a value of 0 indicating no correlation. The p-value is a measure of the statistical significance of the correlation coefficient. A small p-value (generally less than 0.05) indicates that the correlation is statistically significant, while a large p-value (greater than or equal to 0.05) indicates that the correlation is not statistically significant. The calculate_pearson_corr_for function calculates the Pearson correlation coefficient by looping through the elements of the two input data series, calculates the t-statistic based on the correlation coefficient, and then calculates the p-value based on the t-statistic using the error function (erf). It then prints the Pearson correlation coefficient and p-value to the console.

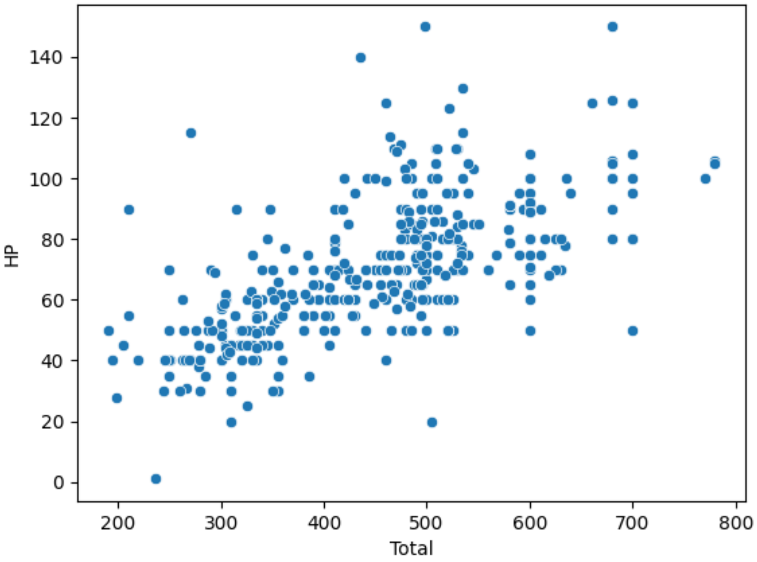
I use this calculate_pearson_corr_for function to do 3 tests:

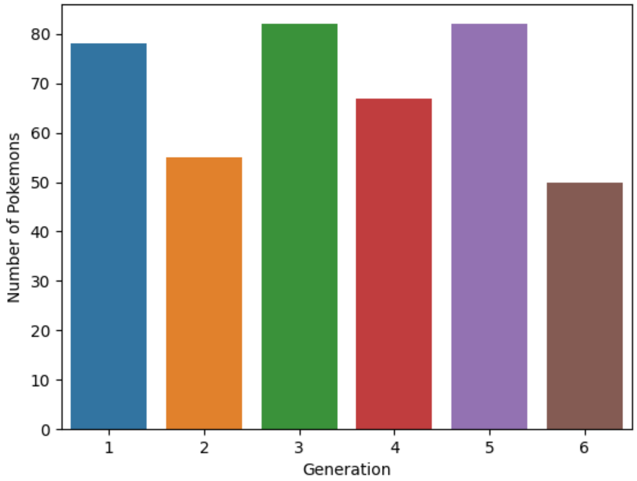
- Total stat and Generation number (there’s no significant relationship between them)
- Sp. Atk and Sp. Def stats (there is a significant relationship between the two variables)
- Attack and Defense stats (there is a significant relationship between the two variables)

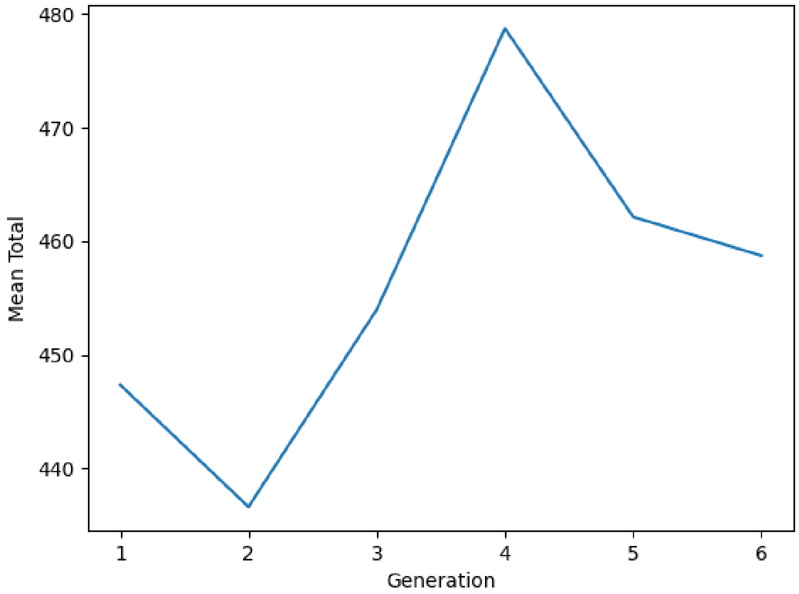
# 7. Appendix (Chart Code & Graphs from Section 2)

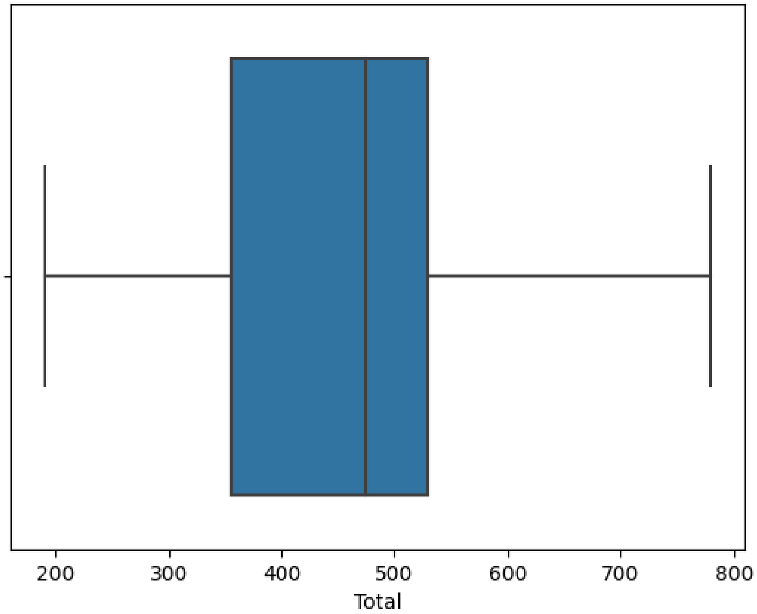


```python
def open_plot_windows(df: pd.DataFrame) -> None:
    plt.figure(1) # Plot 1: Scatter plot for showing the relationship between Total and HP
    sns.scatterplot(x=df['Total'], y=df['HP'])
    plt.xlabel('Total')
    plt.ylabel('HP')

    plt.figure(2) # Plot 2: Bar plot for showing the count of Pokemons in each generation
    sns.barplot(x=df['Generation'].value_counts().index, y=df['Generation'].value_counts().values)
    plt.xlabel('Generation')
    plt.ylabel('Number of Pokemons')

    plt.figure(3) # Plot 3: Line plot for showing the Total attribute for each generation
    sns.lineplot(x=df.groupby('Generation')['Total'].mean().index, y=df.groupby('Generation')['Total'].mean().values)
    plt.xlabel('Generation')
    plt.ylabel('Mean Total')

    plt.figure(4) # Plot 4: Box plot for Total
    sns.boxplot(x=df['Total'])
    plt.xlabel('Total')

    plt.show() # Calling plt.show() at the end so they all get opened in seperate windows
```









**NOTE: rest of the code can be found in the “src” folder that’s inside the ZIP file. I couldn’t fit all of it in here since there’s a max page limit which is equal to 5. (I sent an email about this but didn’t get an answer...)**