Data Narrative Assignment 3

Dewansh Kumar
Computer Science and Engineering
Indian Institute of Technology Gandhinagar
Gandhinagar, India
dewansh.kumar@iitgn.ac.in

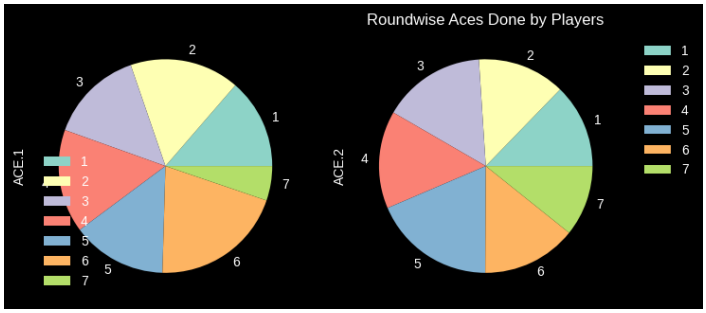
***Abstract*— This report is a Data Narrative on the dataset given to us. It contains 8 questions related to the details given in the dataset with their solutions, codes, plots, and graphs along with observations from the plots and graphs.**

I. Overview of the Dataset

This dataset is a collection of 8 files containing the match statistics for both women and men at the four major tennis tournaments of the year 2013. Each file has 42 columns and a minimum of 76 rows.

II. Scientific Questions and Answers

1) Create a pie chart of the round-wise number of aces done by player 1 and player 2.
Ans:



Observation: We can clearly see from the above pie chart that since there was only one match of round 7 (final match) still there are many aces done. This clearly shows that in final matches many aces are scored and this is a totally practical observation also.

2) Find what is the probability of winning a whole match if a player loses the first two rounds continuously.
Ans:

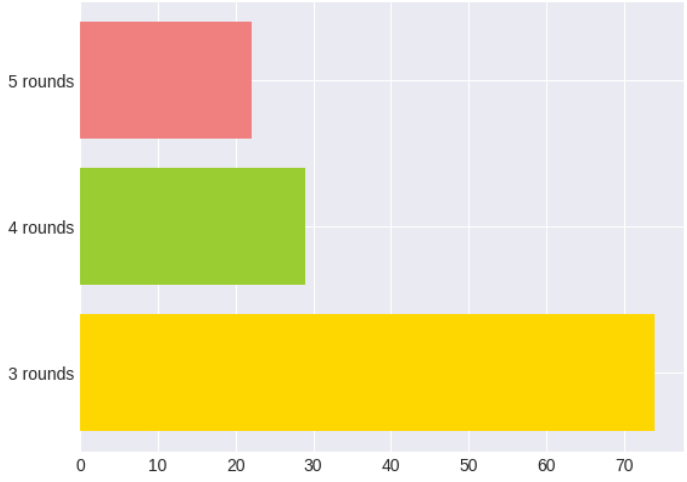
```
df=pd.read_csv('AusOpen-women-2013.csv')
set1=(df['ST1.1']-df['ST1.2']).tolist()
set2=(df['ST2.1']-df['ST2.2']).tolist()
set3=(df['ST3.1']-df['ST3.2']).tolist()
result=df['Result'].tolist()
losefirsttwo=0
flag=0; ind=[]
for i in range(len(df)):
  if set1[i]<0 and set2[i]<0:
    losefirsttwo+=1
for i in range(len(df)):
  if set1[i]<0 and set2[i]<0 and result[i]==1:
    flag+=1
    ind.append(i)
print(flag,losefirsttwo)
print(flag/losefirsttwo)
```

```
0 41
0.0
```

Observation: We can see that in this championship there are a total of 41 matches in which a player loses the first two matches continuously and then also loses the third time also every-time. Hence it can be concluded that the first two matches decide the result of the match almost completely.

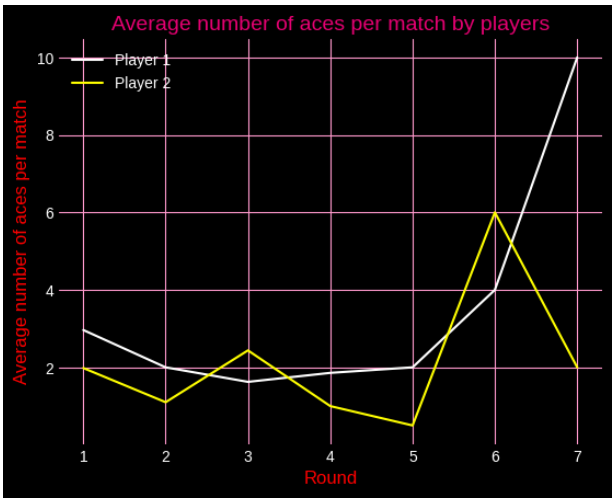
3) Find the number of matches in which the match ended in (a) 3 sets (b) 4 sets (c) 5 sets. Also, plot a bar graph for this.
Ans:

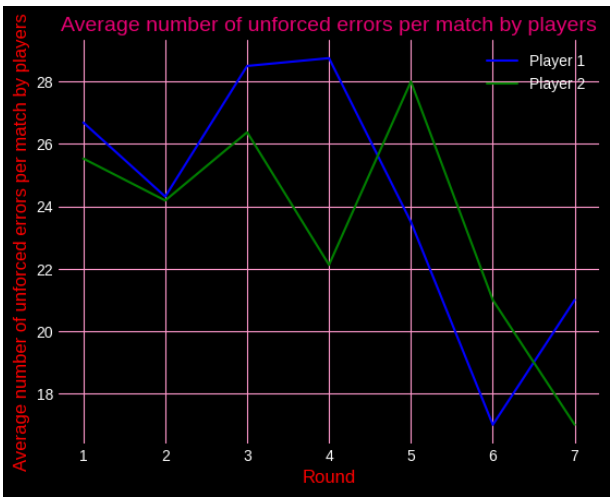
Total matches: 125
Matches in which 3 sets were played: 74
Matches in which 4 sets were played: 29
Matches in which 5 sets were played: 22

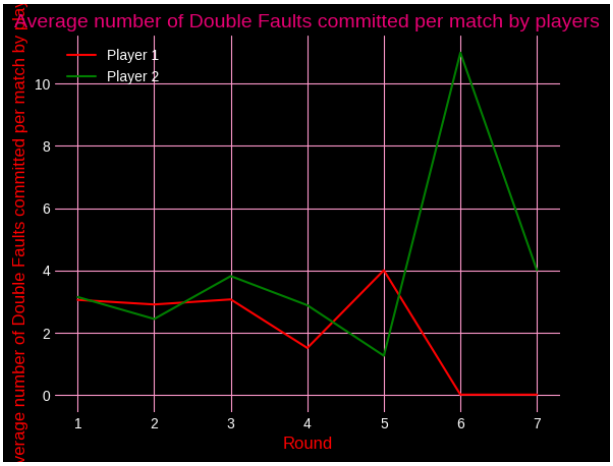


Observation: We can see that most of the matches ended in only 3 sets of games which is not very common.

4) Plot a line graph that shows the relationship between the average number of aces done by player1 and player2 and the round of the match. Similarly, also plot the line graph for the relationship between the average number of unforced errors and the average number of double faults vs the round of the match.
Ans:

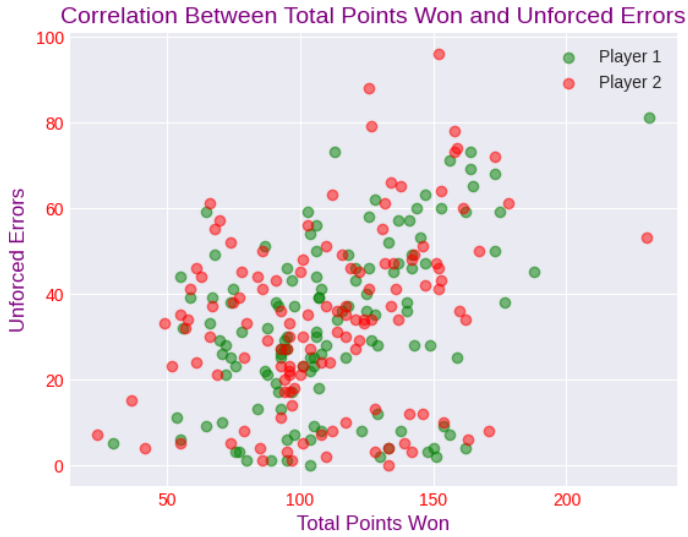






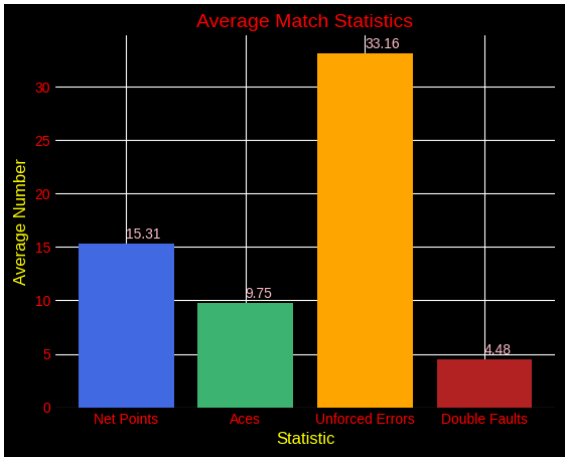
Observation: From the above line graph we can clearly see that as the round number increases the number of aces increase, which is not the case with the other two. The average number of unforced number errors decreases as the round number increases which is quite obvious since as the round increases there are better players in the game. But the average number of double faults doesn't show any fixed behavior, it is kind of constant and doesn't depend much on the round number.

5) Is there any correlation between total points won by a player and unforced errors made by the player. Plot a scatter graph to visualize any such relationship clearly.
Ans:



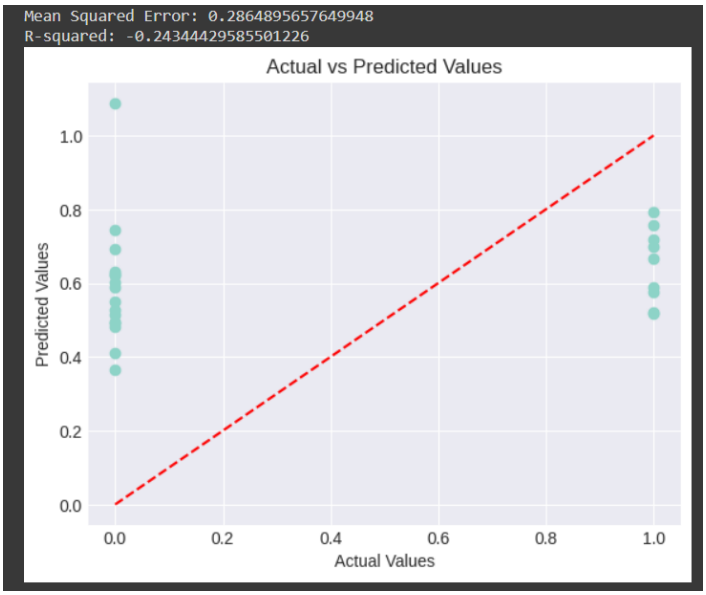
Observation: We can see some relationship between total points won and unforced errors from the above scatter plot. We can roughly say that as the unforced errors increase, the total points won also increase. But, this is not always true and hence there is not any good relationship between these two.

6) Find the average statistics of the championship like the average number of net points attempted per match by a player, the average number of aces per match by a player, the average number of unforced errors per match by a player, etc. Finally, plot a bar graph for this data.
Ans:



Observation: We can clearly see and visualize the statistics of the championship from the above-plotted bar graph.

7) Make a linear regression model and train it using the data available. Then predict the results of all the matches and calculate the mean-squared error. Also, plot a scatter plot for the actual result and predicted result.
Ans:

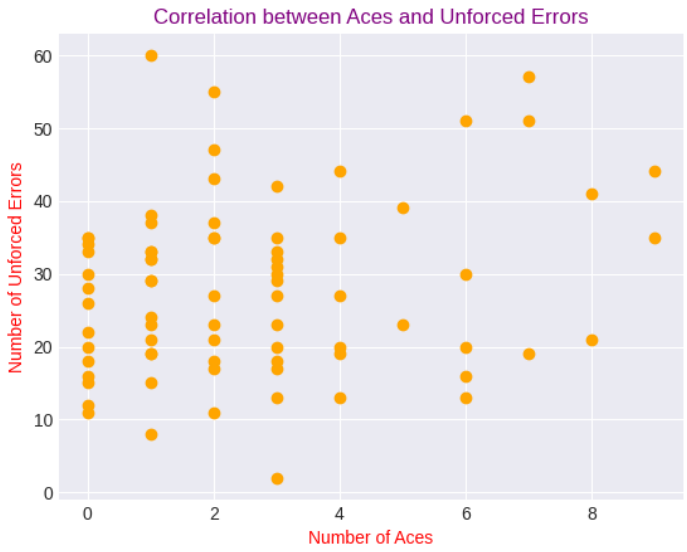


Observation: We can see that the mean squared error for this trained model is around 29%, which is not so a bad model but still can't be used to predict the result of a game clearly. We can conclude that linear regression can't be applied everywhere on any dataset and we can expect to get a very accurate prediction. We should use different types of machine-learning models to correctly predict the result of a game.

Code-Snippet:

```
df=pd.read_csv('Wimbledon-women-2013.csv')
from sklearn.linear_model import LinearRegression
tennis_data = df
X = tennis_data[['UFE.1', 'FSP.1', 'FSP.2', 'SSP.1', 'NPA.1', 'NPA.2']]
y = tennis_data['Result']
X.loc[:, 'UFE.1'] = X['UFE.1'].fillna(0)
X.loc[:, 'FSP.1'] = X['FSP.1'].fillna(0)
X.loc[:, 'FSP.2'] = X['FSP.2'].fillna(0)
X.loc[:, 'SSP.1'] = X['SSP.1'].fillna(0)
X.loc[:, 'NPA.1'] = X['NPA.1'].fillna(0)
X.loc[:, 'NPA.2'] = X['NPA.2'].fillna(0)
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

8) Is there any relationship between the number of aces and unforced errors? Plot a scatter graph to visualize any such relationship if exists.
Ans:



Observation: We can clearly see that there is no such relationship between the number of aces and unforced errors.

Code-Snippet:

```
tennis_data=pd.read_csv('USOpen-women-2013.csv')
x = tennis_data['ACE.1']
y = tennis_data['UFE.1']
plt.scatter(x, y,color='orange')
plt.title('Correlation between Aces and Unforced Errors',color='purple')
plt.xlabel('Number of Aces',color='red')
plt.ylabel('Number of Unforced Errors',color='red')
plt.show()
```

III. Details of Libraries and Functions

A) Libraries

1. Pandas: Pandas is a library used in Python programming language for data manipulation and analysis. It provides data structures and functions using which the program becomes fast and efficient. It makes the Python environment very powerful, fast, effective, and productive. There are many pros of using pandas like, any file object can be read and manipulated using it, different datasets can be joined easily, etc. [1].
2. Matplotlib: It is a data visualization library in Python used for 2D plotting. It is used for plotting graphs and plots of lines, histograms, bar graphs, pie charts, etc. Using Matplotlib, we can plot high-quality graphs with minimal effort [2].

B) Functions

1. '.read' function: It is a function of Pandas library used to read all types of file objects like data tables, CSV files, etc.
2. '.groupby' function: It is also a function of Pandas library used to group a dataset after splitting it, applying a function on it, and combining the results. It

can be used efficiently to group a large amount of data and compute operations on the groups.

3. '.count' function: It is also a function of the Pandas library used to count the values of different attributes of a dataset. We can use it to count the values of each row or each column of a data frame.
4. '.sum' function: It too is a function in Pandas library used to find the sum of values of specified axis. It is very useful in finding sums as it skips the missing value and computes the rest.
5. '.sort' function: It is also a function of Pandas library used to sort a data frame along either axis in ascending or descending order as specified.

IV. ACKNOWLEDGMENT

I would like to thank Prof. Shanmuganathan to give this wonderful opportunity to mine this dataset and frame some amazing questions. It was a very interesting thing and I learned a lot of basic data science concepts while doing this.
I would also like to thank my TA Mr. Akbar and other TAs who always were very happy to help me with any doubts during lab sessions.

V. REFERENCES

- [1] *Pandas documentation#* (no date) *pandas documentation - pandas 1.5.3 documentation*. Available at: https://pandas.pydata.org/docs/ (Accessed: April 23, 2023).

- [2] *Matplotlib 3.7.1 documentation#* (no date) *Matplotlib documentation - Matplotlib 3.7.1 documentation*. Available at: https://matplotlib.org/stable/index.html (Accessed: April 23, 2023).