Assignment 05–Essential Software–Spring2023

Problem1

In this problem, you will simulate expected outcome of the upcoming general election in Pakistan. In Pakistan, in the National Assembly, we have total of 342 seats and one has to secure 172 seats to establish majority and thereby form the National Government. Though there are several political parties in Pakistan, I will list only some of them with their past 2018 scores without naming them with their true names. Consider a dictionary $\{'A' : 149,'B' : 82,'C' : 54,'D' : 15,'E' : 3,'F' : 1,'G' : 7,'H' : 5,'I' : 5,'J' : 4,'K' : 1,'L' : 1,'M' : 13,'N' : 2\}$, which has hypothetical names of all political parties and actual NA seats won in the 2018 elections. Construct a DataFrame which consists of two columns such that the first column has names of the political parties and the second column has corresponding NA seats as given above. Label the columns with appropriate titles. For simulation purpose, define a function that generates random number representing number of seats won by each political party in such a way that the total sum of all the seats won by all political parties equals 342. The above 2018 data suggests that for any party, it is less likely to score seats above 100 and comparatively more chances to score between 50-100 and even more to score less than 50 seats. Based on these observations, we assign probabilities to each randomly generated number of seats. Define a function that assigns probability to each score with following details: if score is greater than 100 its probability is 0.08, if the score is between 50-100 the probability is 0.15 and if the score is less than 50 the probability is 0.77. Append two more columns of randomly generated scores and their corresponding probabilities respectively to the previously defined DataFrame. Repeat the process of allocating random seats and assigning probabilities to each party 100 times. Subsequently, calculate the averages of the seats and probabilities for each party over 100 runs. Represent 2018 and your averaged results over 100 runs with some appropriate graphs. The following itemized list shows the required items in your code.

- There should be a DataFrame object by the name of *df* where you store and work with all of your data.
- There must be a function by the name of *generate_seats()* which allocates random seats to each party such that the sum is 342.
- There must be a function by the name of *assign_prob()* which assigns probabilities to randomly generated seats as per description given above.
- Generate a figure and divide it into four axes such that first axis shows graph of data from 2018, second graph shows that average data over 100 runs, third graphs shows the average probabilities and fourth graph combines previous three graphs. For each graph, on x axis always plot name of the political party. You may decide any type of the graph such as line, bar, etc. Each graph should have proper labels, markers, markersize, legend, etc.

Problem 2

Download Titanic data from www.kaggle.com and do the following.

- Bring the data into appropriate form, for example, fill/remove the missing values, encode an object column to numerical if required, etc.
- Decide the appropriate feature and target matrices.
- Draw the pair plots of all features and calculate the correlation matrix
- Based on correlation matrix, if needed, further refine the data.
- Split the data into training and test data sets.
- Fit the LinearRegression, and Gaussian Naive Bayes models to the data.
- Check the accuracy of both methods using test data.
- Compare the two methods.

Submission: It is due on Thursday 11,2023 by 11:55PM.