Project Part 1

Due date : Friday 4th March by 11pm. This part is worth 25% of the overall.

You will be docked 5% for each day you are late with submission, in accordance with IT Sligo Marks and Standards. There are three files for this assignment.

1. This pdf : `Projectp1.pdf`
2. The data set Premier League Matches 2019/20 : `matches.csv`.
3. Description of Premier League data set : `columns.txt`.
4. Lyrics of Taylor Swift Songs split per album : `Taylorlyrics.zip`.

Ensure when you are uploading your work you only upload your own work and **not** any dataset. **Just upload the ipynb file, renamed to have your name and Student Number**.

This is **NOT** a group assignment. You must complete the assignment **individually**. If you share your work, your marks will be divided. Some students may be asked to demonstrate their knowledge of the submitted work. Ensure you read the complete document.

When you are asked to answer a question. Put answers in markdown cells. The interpretation of information and results is at least as important as the building of models.

Probability Problem

1 Rolling a Dice (8 Marks)

You wish to find the expected value $E(X)$ of rolling a fair k-sided dice (6,12,20) N times by simulating the rolling of the dice (like flipping a coin). You can recall from notes that rolling a 6-sided dice has $E(X) = 3.5$.

1. You should write a function that has the attributes k and N, k being the number of sides to a dice and N being the number of times to roll the dice.
2. This function will return the mean value of the N rolls of the dice.
3. Run the trials with 6, 12 and 20 sided dice, a number of times printing each value (do it in steps up to 1000)
4. Comment (in a markdown cell) what your estimate for $E(X)$ is based on these.
5. Make a plot for each of the three different types of dice.
    - Plot the number of trials vs the mean value in that number of trials.
    - Do 10000 trials.
    - Print a line with your estimate for $E(X)$.
    - Style the plot with informative labels and axis.
6. Using code (numpy array and function if possible), calculate the $E(X)$ for the three different types of dice using what you know about basic probability. Does this match your earlier estimate?

2 Data Exploration Football (9 Marks)

The file `matches.csv` contains detailed information about all the football matches in the Premier League in 2019/20. The file `columns.txt` gives the meaning of each of the columns. If you do not follow football at all, this is just data to analyse.

You want to analyse the data using Pandas

1. Find out the following bits of information
   (a) The mean number of home goals per game
   (b) The mean number of away goals per game
   (c) The mean number of goals per game
   (d) The mean number of goals in the first half
   (e) The mean number of goals in the second half
   (f) The mean number of goals in any half
   (g) The mean number of yellow cards in a match
   (h) The mean number of red cards in a match

   Storing all of these in variables.
2. Plot on a bar chart, the number of home wins, the number of away wins and the number of draws.
3. Make box plots for home goals, aways goals, home team fouls and away team fouls - all on the same figure.
4. Plot on a histogram the number of matches on each date.
   - Why do you think there is a big gap between some dates? Write it in a markdown cell
5. Pick any team in the league at random, the one you support if you want.
   - The groupby method may come in handy `https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.groupby.html`

   Get the following information about your team
   (a) The mean number of goals scored per game
   (b) The mean number of goals conceded per game

   Using this information make a model to predict
   (a) The Probability of scoring 2 or more goals in the next game
   (b) The Probability of conceding 2 or more goals in the next game

3 Analysing Taylor Swift Lyrics (8 marks)

You have lyric information for 9 albums. You wish to read in all these albums and find out how many times each word appears per song.

Unzip the songs into a new folder called `lyrics` in the same place as your ipynb file.

```
import glob
allalbums = glob.glob("lyrics/*.csv")
```

will give you a list of all csv files in that folder.

Loop through that list of files, creating a dataframe for each. For each dataframe you can get the number of tracks by

```
maxtrack = df.track_n.max()
```

You can then get the lyrics for every track with

```
for i in range(1,maxtrack+1):
    tracklyrics = " ".join(df[df.track_n==i]["lyric"].values).replace("\"", "")
```

Then create a dictionary that stores the number of times each word appears in that song (I'm sure this has been done before in class for something similar).

In the end you need a list of dictionaries. Each entry in the list contains the count of every word per song, for example (the song Tim McGraw)

```
{"i'm": 2, "weren't": 1, 'not': 1, 'from': 1, 'bittersweet': 1,
'your': 6, 'black': 3, "lookin'": 1, 'tears': 1, 'shame': 2,
'letter': 2, 'it': 2, 'gone': 1, 'shined': 2, 'time': 2,
'left': 1, 'tim': 6, 'put': 2, 'one': 2, 'mmmm': 1, 'never': 1,
'someday': 1, 'had': 1, 'long': 3, 'stars': 2, 'but': 2,
'of': 10, "that's": 2, 'when': 9, 'like': 3, 'little': 4,
```

Now let's pick a particular word ("your"), if we want to see how many times the word appears in a particular song we can get it by

```
trackdict[i].get("your", 0)
```

This searches the dictionary for track indexed by `i` for the word "your" and returns how many times it appears, it gives 0 if it does not occur. If I want to know the total number of times "your" appears in all the songs, I loop over all the tracks and add them all up together. I can then get the average amount of times "your" appears.

1. Having said all of that, your task is to pick two words you think she may use often.
2. Find the average number of times those words are used per song.
3. Find the probability of a song having the words sung twice.
4. Find the probability of the words being sung at most two times.

Suspected Cheating & Plagiarism

Attention is drawn to IT Sligo's definition of plagiarism at `https://vle.itsligo.ie/course/view.php?id=2206`. Student plagiarism occurs when a student presents the work of another as their own work, without appropriate acknowledgement. It can include:

- Presenting work which has been copied from the Internet, books, journals or other sources;
- Presenting work which paraphrases the writings of other authors, without acknowledgement;
- Presenting work which has been written by somebody else, such as another student or a family member;
- Presenting work which has been purchased from an Internet site or other source and submitting as own work;
- Presenting work which has been produced collaboratively as one's own individual work;
- Student plagiarism can also occur where a student submits the same piece of their own work for a number of different assignments;

Plagiarism by students can be deliberate, where a student intentionally attempts to pass off the work of another as their own work. It can also be accidental, where a student fails to use appropriate citation and referencing in their work or is unaware of what constitutes plagiarism.