DSO110 - Final Group Project - Lottery

Alberta "Albi" Kovatcheva and Barbra Treston

Background

Albi and Barbra have chosen the “Mega Millions Winning Numbers” dataset because the lottery is something that is familiar and accessible to a wide range of people worldwide; it would be difficult to find someone who hasn’t dreamed of hitting the jackpot and changing their life forever. However, it is also widely accepted that the lottery is not set up to favor the player. In the case of Mega Millions, although there is a 1 in 24 chance of winning something, the odds of choosing all 6 numbers correctly to win the jackpot is 1 in 302,575,350 - a fact that is posted openly on both the New York Lottery and Mega Millions websites. By analyzing the winning numbers data as well as complementary datasets on lottery retailers, lottery aid to local school districts, and monies recouped from the lottery winnings of public aid recipients, Albi and Barbra hope to glean insight to make actionable suggestions on how lottery players can get the best return on their investment as well as to demonstrate for the average person whether the lottery serves any societal good or whether it may be best to abstain from playing altogether.

Data Wrangling

The data must be wrangled/formatted to be suitable for analysis.

Tasks:

1. From 'Draw Date', extract the month.
2. From 'Draw Date', extract the day.
3. From 'Draw Date', extract the year.
4. From 'Draw Date', extract the weekday.
5. From 'Draw Date', extract the quarter.
6. Separate the 'Winning Numbers' column, into 5 columns, with each containing one of the winning numbers in their corresponding order of being drawn.

Import data.

In [1]:

```
import pandas as pd
import seaborn as sns
import matplotlib.mlab as mlab
import matplotlib.pyplot as plt
import math
import numpy as np
from numpy import nan
import datetime as dt
from datetime import date
```

In [2]:

```
Winning_Numbers = pd.read_csv("C:/Users/albi/Downloads/lottery/Lottery_Mega_M

pd.set_option("display.max_columns", None)

Winning_Numbers.head()
```

Out[2]:

	Draw Date	Winning Numbers	Mega Ball	Multiplier
0	09/25/2020	20 36 37 48 67	16	2.0
1	09/29/2020	14 39 43 44 67	19	3.0
2	10/02/2020	09 38 47 49 68	25	2.0
3	10/06/2020	15 16 18 39 59	17	3.0
4	10/09/2020	05 11 25 27 64	13	2.0

Tasks 1-5: Extract month, day, year, weekday, and quarter from 'Draw Date'.

1. Extract month from 'Draw Date'.

In [3]:

```
Winning_Numbers['month'] = pd.DatetimeIndex(Winning_Numbers['Draw Date']).mor
Winning_Numbers.head()
```

Out[3]:

	Draw Date	Winning Numbers	Mega Ball	Multiplier	month
0	09/25/2020	20 36 37 48 67	16	2.0	9
1	09/29/2020	14 39 43 44 67	19	3.0	9
2	10/02/2020	09 38 47 49 68	25	2.0	10
3	10/06/2020	15 16 18 39 59	17	3.0	10
4	10/09/2020	05 11 25 27 64	13	2.0	10

2. Extract day from 'Draw Date'.

In [4]:

```
Winning_Numbers['day'] = pd.DatetimeIndex(Winning_Numbers['Draw Date']).day
Winning_Numbers.head()
```

Out[4]:

	Draw Date	Winning Numbers	Mega Ball	Multiplier	month	day
0	09/25/2020	20 36 37 48 67	16	2.0	9	25
1	09/29/2020	14 39 43 44 67	19	3.0	9	29
2	10/02/2020	09 38 47 49 68	25	2.0	10	2
3	10/06/2020	15 16 18 39 59	17	3.0	10	6
4	10/09/2020	05 11 25 27 64	13	2.0	10	9

3. Extract year from 'Draw Date'.

In [5]:

```python
Winning_Numbers['year'] = pd.DatetimeIndex(Winning_Numbers['Draw Date']).year
Winning_Numbers.head()
```

Out[5]:

	Draw Date	Winning Numbers	Mega Ball	Multiplier	month	day	year
0	09/25/2020	20 36 37 48 67	16	2.0	9	25	2020
1	09/29/2020	14 39 43 44 67	19	3.0	9	29	2020
2	10/02/2020	09 38 47 49 68	25	2.0	10	2	2020
3	10/06/2020	15 16 18 39 59	17	3.0	10	6	2020
4	10/09/2020	05 11 25 27 64	13	2.0	10	9	2020

4. Extract weekday from 'Draw Date'.

In [6]:

```python
Winning_Numbers['weekday'] = pd.DatetimeIndex(Winning_Numbers['Draw Date']).c
Winning_Numbers.head()
```

Out[6]:

	Draw Date	Winning Numbers	Mega Ball	Multiplier	month	day	year	weekday
0	09/25/2020	20 36 37 48 67	16	2.0	9	25	2020	4
1	09/29/2020	14 39 43 44 67	19	3.0	9	29	2020	1
2	10/02/2020	09 38 47 49 68	25	2.0	10	2	2020	4
3	10/06/2020	15 16 18 39 59	17	3.0	10	6	2020	1
4	10/09/2020	05 11 25 27 64	13	2.0	10	9	2020	4

5. Extract quarter from 'Draw Date'.

In [7]:

```python
Winning_Numbers['quarter'] = pd.DatetimeIndex(Winning_Numbers['Draw Date']).c
Winning_Numbers.head()
```

Out[7]:

	Draw Date	Winning Numbers	Mega Ball	Multiplier	month	day	year	weekday	quarter
0	09/25/2020	20 36 37 48 67	16	2.0	9	25	2020	4	3
1	09/29/2020	14 39 43 44 67	19	3.0	9	29	2020	1	3
2	10/02/2020	09 38 47 49 68	25	2.0	10	2	2020	4	4
3	10/06/2020	15 16 18 39 59	17	3.0	10	6	2020	1	4
4	10/09/2020	05 11 25 27 64	13	2.0	10	9	2020	4	4

Task 6: Convert 'Winning Numbers' to string and then separate terms into individual columns (5).

Determine the data types in the "Winning_Numbers" data frame. To accomplish this task, the 'Winning Numbers' data must be of the string type.

In [8]:

```
Winning_Numbers.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2036 entries, 0 to 2035
Data columns (total 9 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Draw Date        2036 non-null   object 
 1   Winning Numbers  2036 non-null   object 
 2   Mega Ball        2036 non-null   int64  
 3   Multiplier       1133 non-null   float64
 4   month            2036 non-null   int64  
 5   day              2036 non-null   int64  
 6   year             2036 non-null   int64  
 7   weekday          2036 non-null   int64  
 8   quarter          2036 non-null   int64  
dtypes: float64(1), int64(6), object(2)
memory usage: 143.3+ KB
```

Winning_Numbers.info() shows that the 'Winning Numbers' data is of the object type. Below, it is converted to string type.

In [9]:

```
Winning_Numbers["Winning Numbers"]= Winning_Numbers["Winning Numbers"].astype
```

After the 'Winning Numbers' data is converted to string type, it is split into individual columns.

In [10]:

```
Winning_Numbers1 = Winning_Numbers['Winning Numbers'].str.split(' ', expand=T
```

Below is the output of this operation.

In [11]:

```
Winning_Numbers1.head()
```

Out[11]:

	0	1	2	3	4
0	20	36	37	48	67
1	14	39	43	44	67
2	09	38	47	49	68
3	15	16	18	39	59
4	05	11	25	27	64

Winning_Numbers.info() is used to verify the data types in the "Winning_Numbers1" data frame.

In [12]:

```
Winning_Numbers1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2036 entries, 0 to 2035
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   0       2036 non-null   object
 1   1       2036 non-null   object
 2   2       2036 non-null   object
 3   3       2036 non-null   object
 4   4       2036 non-null   object
dtypes: object(5)
memory usage: 79.7+ KB
```

To prevent a NaN value, the winning numbers are converted into the integer data type.

In [22]:

```
Winning_Numbers1[0]= Winning_Numbers1[0].astype(int)
Winning_Numbers1[1]= Winning_Numbers1[1].astype(int)
Winning_Numbers1[2]= Winning_Numbers1[2].astype(int)
Winning_Numbers1[3]= Winning_Numbers1[3].astype(int)
Winning_Numbers1[4]= Winning_Numbers1[4].astype(int)
Winning_Numbers1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2036 entries, 0 to 2035
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   0       2036 non-null   int32
 1   1       2036 non-null   int32
 2   2       2036 non-null   int32
 3   3       2036 non-null   int32
 4   4       2036 non-null   int32
dtypes: int32(5)
memory usage: 39.9 KB
```

Here, the Winning_Numbers & Winning_Numbers1 dataframes are concatenated together.

In [33]:

```
result = pd.concat([Winning_Numbers1, Winning_Numbers], axis=1)

#frames = [Winning_Numbers, Winning_Numbers1]
```

In [34]:
```
result.head()
```

Out[34]:

	0	1	2	3	4	Draw Date	Winning Numbers	Mega Ball	Multiplier	month	day	year	weekday	c
0	20	36	37	48	67	09/25/2020	20 36 37 48 67	16	2.0	9	25	2020	4	
1	14	39	43	44	67	09/29/2020	14 39 43 44 67	19	3.0	9	29	2020	1	
2	9	38	47	49	68	10/02/2020	09 38 47 49 68	25	2.0	10	2	2020	4	
3	15	16	18	39	59	10/06/2020	15 16 18 39 59	17	3.0	10	6	2020	1	
4	5	11	25	27	64	10/09/2020	05 11 25 27 64	13	2.0	10	9	2020	4	

Export data to excel file.

In [35]:
```
result.to_excel("Winning_Numbers_Wrangled.xlsx")
```

In [38]:
```
import os
os. getcwd()
```

Out[38]:
```
'C:\\Users\\albi'
```

Now that this data is wrangled, it is ready for use in analysis.

In [ ]:
```

```