Phase 4: Data Exploration

Team Name: La Casa de Papel

Team Members:- Smit Kakadiya, Bhargav Patel, Saurav Aich

UNH Emails:-

Saurav Aich:- saich1@unh.newhaven.edu

Smit Mansukhbhai Kakadiya:- skaka3@unh.newhaven.edu

Bhargav Prakashchandra Patel:- bpate21@unh.newhaven.edu

Student ID:-

Saurav Aich:- 00718242

Smit Mansukhbhai Kakadiya:- 00703186

Bhargav Prakashchandra Patel:- 00711864

Team Head:-

Saurav Aich

Selected dataset and Research Question : -

Our dataset has been taken from Webmd which contains data from the year 1990 to 2012 where people across different age groups along with their food habits and vital information. These data have been collected to answer the question which is What makes the US population susceptible to diabetes?

Data Exploration : -

Data Exploration in simple words can be explained as the technique to understand various aspects of the data. Dataset just provides the data but what information we are trying to fetch out and how we can achieve it. Data Exploration also helps us to relate variables with each other and plot it in a map which gives us a clear view about the data. Every dataset has discrepancies like null values or redundancy which need to be cleaned. The cleaner the data the better view about the data can be gained.

Data exploration is usually performed in 3 major steps: -

1. Understanding the data.
2. Cleaning the data.
3. Relationship between the variables.

Tool Used : - Jupyter Notebook a python framework has been used to carry out the data exploration.

Understanding the data : -

For data exploration certain libraries need to be called among them pandas, numpy and seaborn. These libraries help to break down the data and then visualize it in different forms.

The first step was to upload the csv file to jupyter notebook.

```
In [2]: data = pd.read_csv("diabetes_dataset.csv")
```

The above command helped to upload the csv file to the python framework.

```
In [3]: print(data.head())
```

The above command prints the head of the table which takes the first 4 rows of the database and displays the information.

```
  Class Class Language   Age  Year Gender Insurance Category  \
0   APH        English  47.0  2016      F  Private insurance
1  PCHW        Spanish  35.0  2015      F              Other
2  ARCF        English  58.0  2015      F           MediCARE
3  PCHW        Spanish  41.0  2015      F               None
4  ARCF        English  56.0  2015      M               None

      Medical Home Category          Race/Ethnicity Education Level  \
0           Doctor's Office         American Indian          College
1                       NaN         Hispanic/Latino              NaN
2                       NaN  Black/African American              1-8
3  No regular place of care         Hispanic/Latino              NaN
4           Emergency Room                     NaN          College

  Diabetes Status (Yes/No)  ... Fruits & Vegetable Consumption  \
0                      Yes  ...                             3-4
1                       No  ...                             1-2
2                      NaN  ...                             1-2
3                       No  ...                             1-2
4                       No  ...                             1-2

  Sugar-Sweetened Beverage Consumption  Food Measurement  \
0                                    0            0 days
1                                    2  I don't know how
2                                  NaN               NaN
3                                    2  I don't know how
4                                    1            0 days
```

After it is seen that the data was able to be fetched out of the database. The next step was to know how many rows and columns are there in the dataset.

```
In [5]: print(data.shape)
```

```
(1688, 25)
```

The data.shape provided us the answer where it was seen that there are a total of 1688 rows and 25 columns.

Every Dataset has unique values for example the Gender column contains 2 unique values which are male and female. Now every column in the dataset has their own unique values and therefore it's essential to figure it out and through python this can be achieved.

```
In [8]: print(data.nunique())
```

This is the above command which fetches all the unique values from all the columns.

```
Class                                          4
Class Language                                 3
Age                                           75
Year                                           3
Gender                                         3
Insurance Category                             6
Medical Home Category                          9
Race/Ethnicity                                 7
Education Level                                8
Diabetes Status (Yes/No)                       3
Heart Disease (Yes/No)                         2
High Blood Pressure (Yes/No)                   2
Tobacco Use (Yes/No)                           2
Previous Diabetes Education (Yes/No)           2
Diabetes Knowledge                             3
Fruits & Vegetable Consumption                 5
Sugar-Sweetened Beverage Consumption           5
Food Measurement                               6
Carbohydrate Counting                          5
Exercise                                       7
Problem Area in Diabetes (PAID) Scale Score   80
ZIP code (address)                             0
ZIP code (city)                                0
ZIP code (state)                               0
ZIP code (zip)                                86
dtype: int64
```

And if we wanted to know the details of particular column that could also be performed via the command listed below : -

```
In [9]: print(data['Gender'].unique())
```

```
['F' 'M' nan 'f']
```

```
In [10]: print(data['Race/Ethnicity'].unique())
```

```
['American Indian' 'Hispanic/Latino' 'Black/African American' nan 'Asian'
 'White' 'Other' 'Unknown']
```

These above commands helped us in understanding the data, how much data has been stored, what is the count, the unique values these helped us to gain more knowledge about the data. It gave us the picture and insightful information about the dataset.

Cleaning of Data : -

This is the second stage of data exploration. This step includes identifying null values across the dataset and if any column is not required can be dropped via the cleaning of data. When data is clean the analysis of the data becomes easier as without the discrepancies it becomes easier to visualize and create relationships which is the third phase of data exploration.

First step was to calculate the null values.

```
In [11]: print(data.isnull().sum())
```

This above command gave a total number of null values present across the dataset. It calculates all the null values in every column and gives the result as to how many are there across the dataset.

```
Class                                              0
Class Language                                     0
Age                                               32
Year                                               0
Gender                                            37
Insurance Category                               113
Medical Home Category                             91
Race/Ethnicity                                    37
Education Level                                  289
Diabetes Status (Yes/No)                          30
Heart Disease (Yes/No)                           100
High Blood Pressure (Yes/No)                      93
Tobacco Use (Yes/No)                             124
Previous Diabetes Education (Yes/No)             125
Diabetes Knowledge                               154
Fruits & Vegetable Consumption                    52
Sugar-Sweetened Beverage Consumption              53
Food Measurement                                  63
Carbohydrate Counting                             66
Exercise                                          78
Problem Area in Diabetes (PAID) Scale Score     1056
ZIP code (address)                              1688
ZIP code (city)                                 1688
ZIP code (state)                                1688
ZIP code (zip)                                   175
dtype: int64
```

After studying the result it was observed that there are some columns which are not required and can be dropped from the dataset as it would not provide any insightful information on that topic.

In [12]:
```
student = data.drop(['ZIP code (state)'], axis=1)
student = student.drop(['ZIP code (zip)'], axis=1)
student = student.drop(['ZIP code (city)'], axis=1)
student = student.drop(['ZIP code (address)'], axis=1)
```

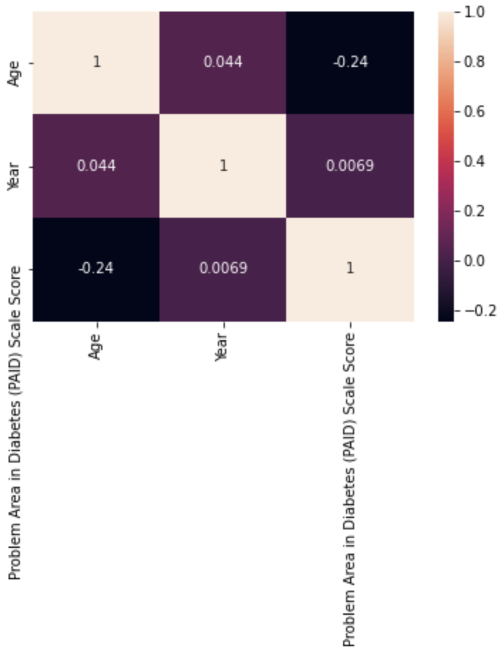
Analysis and Visualizing the data : -

After the completion of the cleaning the next step remains is analyzing and visualizing the variables which can be achieved building a relationship among them.

In [15]:
```
sns.heatmap(corelation, xticklabels = corelation.columns, yticklabels = corel
```

This command helps to visualize the correlation between the variables.

Out[15]: <AxesSubplot:>

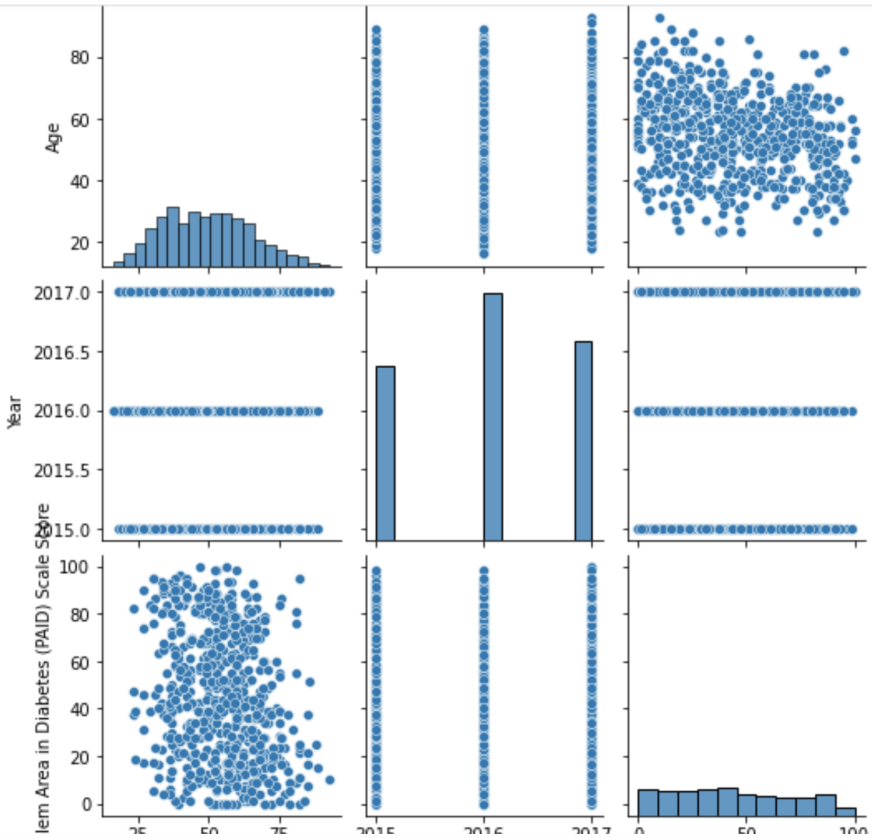


In the above image a heatmap is generated which is a correlation matrix which gives a wider perspective of the dataset for advance analysis. In this image it is observed that year, age, and problems in diabetes scale score are the variables which can be further considered for an advanced analysis.

Heatmap is usually used for integer values as it does not take any categorical values into account which are the string values and for that there is another command which can get a better view of the categorical data.

The pairplot on the other hand takes 2 variables into account and the variables can be continuous categorical or boolean as well and pairplot is a group of plots for the variables in the dataset.

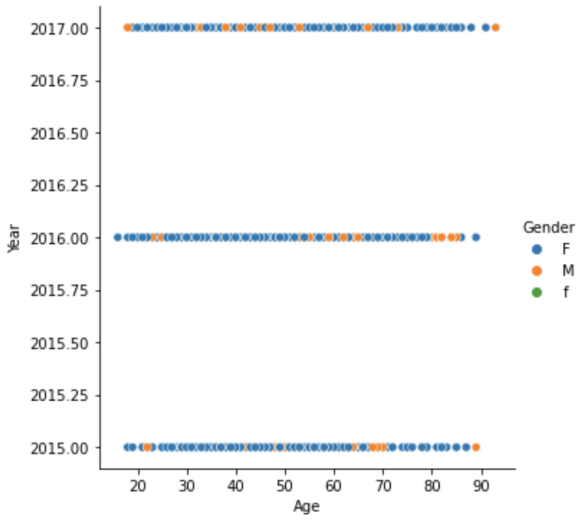
In [16]:
```
sns.pairplot(student)
```



The next step is to plot data on the relationship between two numeric variables when compared with one categorical data variable. The scatter plot helps in achieving this goal.

In [18]:
```
sns.relplot(x= 'Age', y= 'Year', hue= 'Gender', data= student)
```

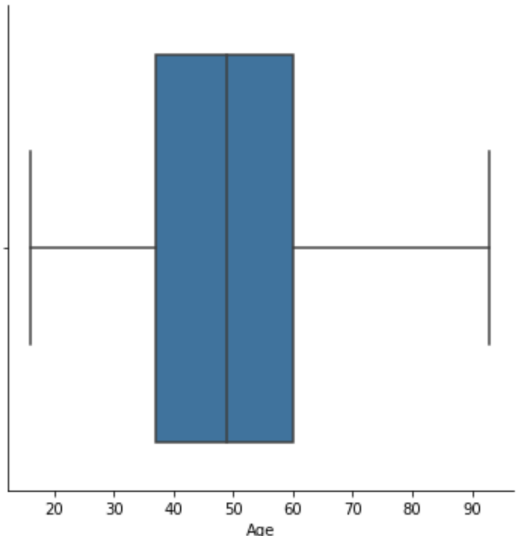
Out[18]:
```
<seaborn.axisgrid.FacetGrid at 0x7fe69b293f10>
```



The last plotting which was used to visualize the data was the categorical plot where the distribution of the variable across the dataset can be visualized.

```
In [29]: sns.catplot(x='Age', kind= 'box', data= student)

Out[29]: <seaborn.axisgrid.FacetGrid at 0x7fe6b8f94c40>
```



Github Repository Link:

https://github.com/Saurav-Aich/Phase-4-Data_Exploration