Team- STAR

(Phase 05: Modeling Data.)

Team Member 1**: Sridevi Jaidi** *[Head of the team]*
Email: sjaid1@unh.newhaven.edu

Team Member 2: **Shyam sunder Reddy Beeram**
Email: sbeer4@unh.newhaven.edu

Team Member 3: **Priyanka Nandigam**
Email: pnand3@unh.newhaven.edu

Content:

1.Introduction

About the dataset:

Dataset link: https://www.kaggle.com/datasets/nsharan/h-1b-visa

This dataset contains five years' worth of H-1B petition data, with approximately 3 million records overall. The columns in the dataset include case status, employer name, worksite coordinates, job title, prevailing wage, occupation code, and year filed.

In detail:
H-1B visas are a category of employment-based, non-immigrant visas for temporary foreign workers in the United States. For a foreign national to apply for H1-B visa, a US employer must offer them a job and submit a petition for a H-1B visa to the US immigration department. This is also the most common visa status applied for and held by international students once they complete college or higher education and begin working in a full-time position.

Research question:

Can we predict visa status of the applicants, by feeding the model with the dataset which contains the required fields by which the machine can classify the visa status as certified or denied.?

Data Mining Techniques:

There are several data mining techniques to be used depending on the data you are going to process such as images, sound, text, and numerical values. The algorithms can be chosen according to the objective. As the dataset which we are using is a **Classification dataset** we can use the following algorithms,

- Logistic Regression
- Random Forest Regression / Classification
- Decision Tree Regression / Classification
- K-Nearest Neighbors
- Support Vector Machine

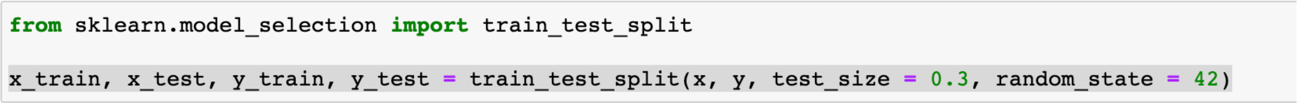
We will need to train the datasets to run smoothly and see an incremental improvement in the prediction rate. The techniques we used in this project were Logistic regression, Random Forest classifier, decision tree classifier.

Parameters:

Model Parameters: These are the parameters in the model that must be determined using the training data set. These are the fitted parameters.

Hyperparameters: These are adjustable parameters that must be tuned in order to obtain a model with optimal performance.

The parameters in our project were:



```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.3, random_state = 42)
```

Hardware:

The Machine we are using to perform the project is:

Hardware Overview:

Model Name:	MacBook Pro
Model Identifier:	MacBookPro18,3
Chip:	Apple M1 Pro
Total Number of Cores:	8 (6 performance and 2 efficiency)
Memory:	16 GB

In order to develop this project, we need to install the following software/packages:

Step 1:

Anaconda Navigator :

Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning related applications. It can be installed on Windows, Linux, and macOS.Conda is an open-source, cross-platform, package management system. Anaconda comes with great tools like JupyterLab, Jupyter Notebook, QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code.

For this project, we will be using **Jupyter** notebook and **Spyder**

Step 2:

To build Machine learning models you must require the following packages

Sklearn: Scikit-learn is a library in Python that provides many unsupervised and supervised learning algorithms.

NumPy: NumPy is a Python package that stands for 'Numerical Python'. It is the core library for scientific computing, which contains a powerful n-dimensional array object

Pandas: pandas is a fast, powerful, flexible, and easy to use open-source data analysis and manipulation tool, built on top of the Python programming language.

Matplotlib: It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits

Flask: Web framework used for building Web applications.

Data Mining Techniques Outcomes:

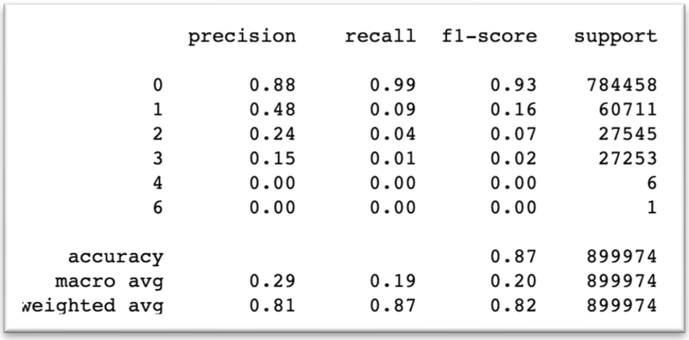
The algorithms can be chosen according to the objective. As the dataset which we are using is a **Classification dataset** we can use the following algorithms,

- Logistic Regression
- Random Forest Regression / Classification
- Decision Tree Regression / Classification
- K-Nearest Neighbors
- Support Vector Machine

Performance Metrics:

Performance metrics using different data mining algorithms based on accuracy of various researchers:

- ○ Logistic Regression: 85%
- ○ K nearest neighbors: 83%
- ○ Support vector Machine: 82%
- ○ Decision tree:77%
- ○ Random Forest Classifier: 89%



```
              precision    recall  f1-score   support

           0       0.88      0.99      0.93    784458
           1       0.48      0.09      0.16     60711
           2       0.24      0.04      0.07     27545
           3       0.15      0.01      0.02     27253
           4       0.00      0.00      0.00         6
           6       0.00      0.00      0.00         1

    accuracy                           0.87    899974
   macro avg       0.29      0.19      0.20    899974
weighted avg       0.81      0.87      0.82    899974
```

Visualization techniques and Outcomes:

The list of exploration techniques (statistical and visualization techniques) used in this project are:

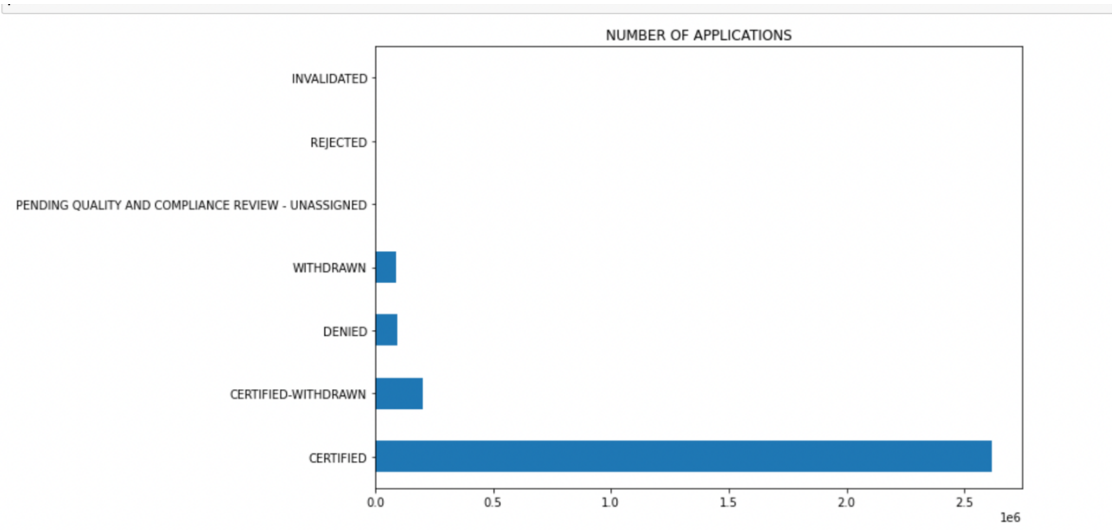
- Bar graph
- Histogram
- White grid graph
- Drawing plot
- Co-relation & Heatmap

After displaying five rows in the dataset, we understand how the dataset and its attributes. And also, we print information about data frame including the index type and columns non null values and memory usage. We also code to get to get a Series containing counts of unique values.

Then after cleaning the data, we start visualizing data in different perspectives.

Bar graph(i):

This is for the analysis of the case status of the applicants.

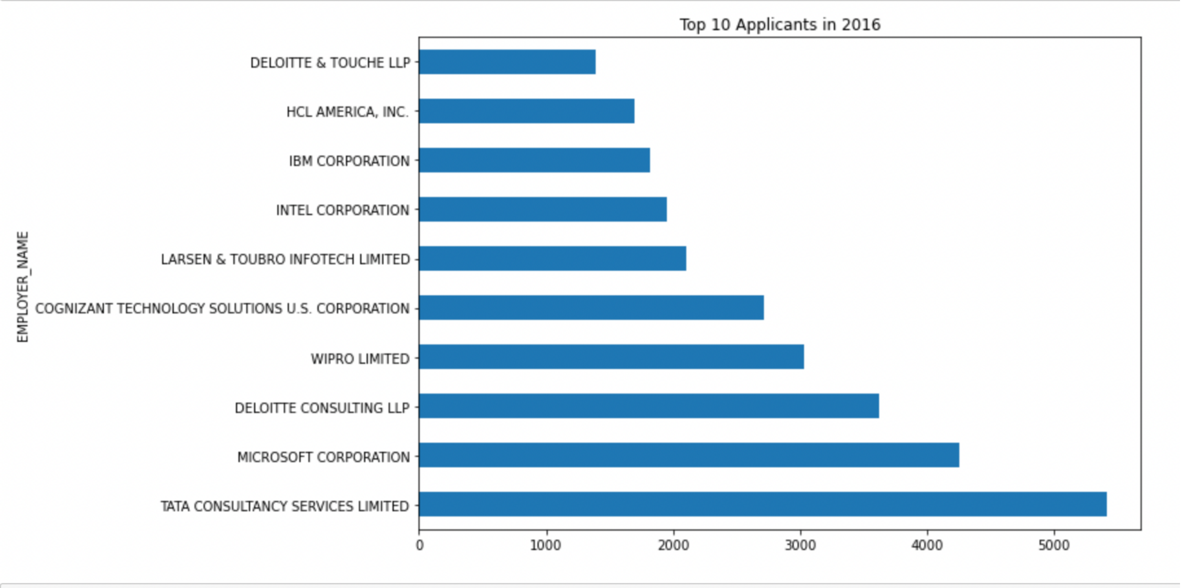


```
Number of Applications per year
```

The graph which we plotted here is a bar graph to know the case status of the applicants per year.

Bar graph (ii)

This is for the analysis of applicants who belong to top 10 companies in 2016.

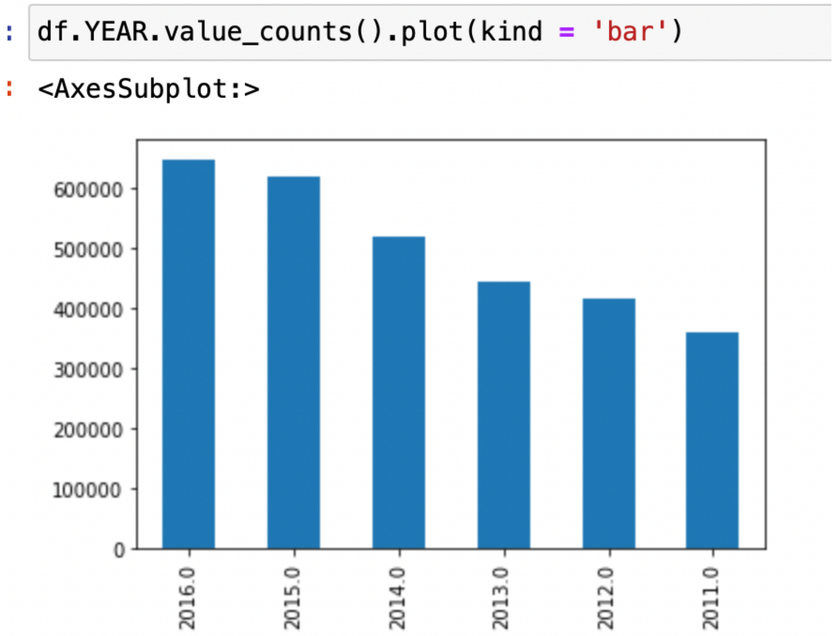


The graph which we plotted here is a bar graph to know the employer who has highest no of applicants in 2016.

In the project there were few other bar graphs to know the status and employer who has good number of applicants.

Histogram:

This is for the analysis of no of applicants per year.



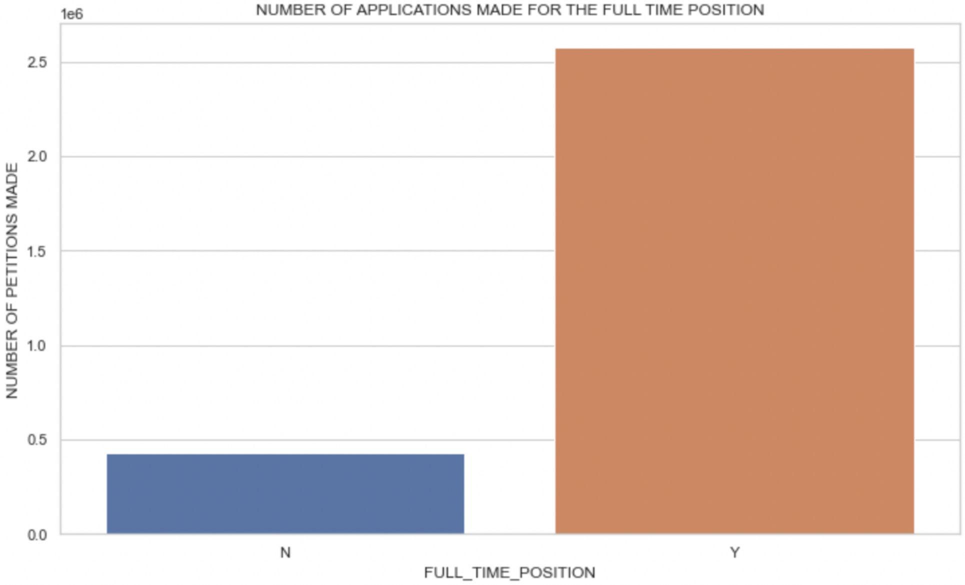
```
: df.YEAR.value_counts().plot(kind = 'bar')
```

```
: <AxesSubplot:>
```

The histogram shows the data of no of applicants filed visa in that respective year.

White grid graph:

This is for the analysis of Number of applicants made for the full time position.

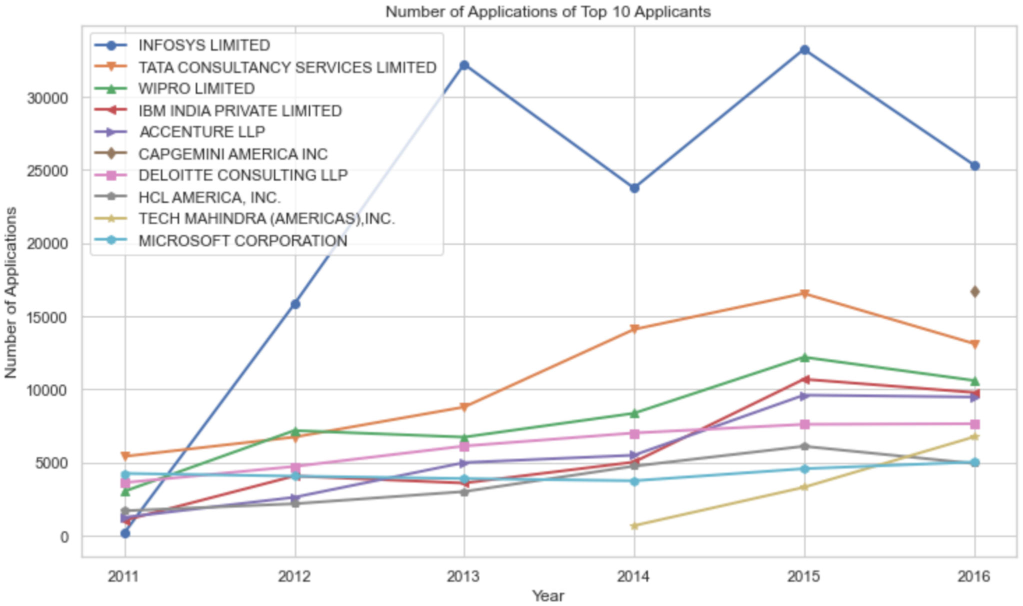


g = sns.countplot(x = 'FULL_TIME_POSITION', data = df)

Here, by using sns we are counting the full time positions and here X-axis is denoted for full time positions.

Drawing Plot:

This is for the analysis of number of applicants of the top 10 companies.



Here the graph tells about the total number of applications of the top 10 companies.
The plot of different color is for the respective company showed in menu graph.

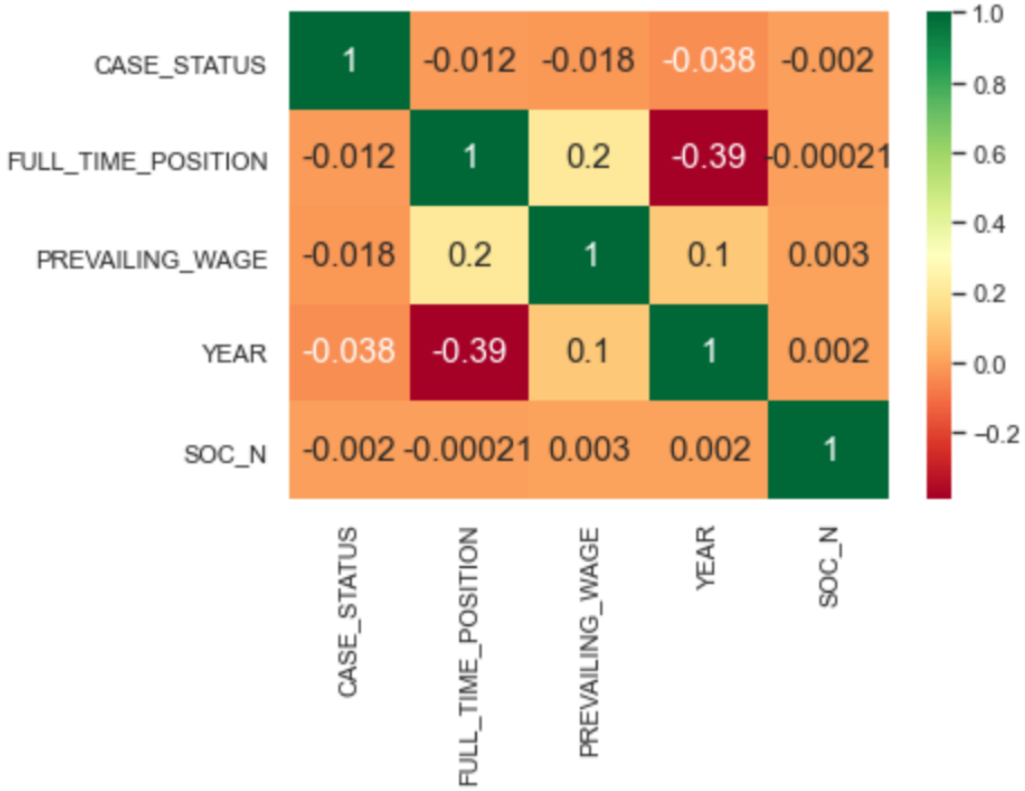
Co-relation & Heatmap:

Correlation is used for measuring the strength and direction of the linear relationship between two continuous random variables x and y. A positive correlation means the variables increase or decrease together. A negative correlation means if one variable increases then the other decrease.

Correlation values can be computed using the 'corr()' method of the Data Frame and rendered using heatmap.

According to our prediction we can drop columns which are not required or preferred.

```
<AxesSubplot:>
```



	CASE_STATUS	FULL_TIME_POSITION	PREVAILING_WAGE	YEAR	SOC_N
CASE_STATUS	1	-0.012	-0.018	-0.038	-0.002
FULL_TIME_POSITION	-0.012	1	0.2	-0.39	-0.00021
PREVAILING_WAGE	-0.018	0.2	1	0.1	0.003
YEAR	-0.038	-0.39	0.1	1	0.002
SOC_N	-0.002	-0.00021	0.003	0.002	1

Conclusion:

There are several Machine Learning algorithms to be used depending on the data you are going to process such as images, sound, text, and numerical values. The algorithms can be chosen according to the objective. As the dataset which we are using is a **Classification dataset** we are using random forest classifier through which we got the good accuracy. We can use Logistic Regression Decision Tree Regression / Classification K-Nearest Neighbors and Support Vector Machine models too but based on accuracy we decided to proceed with Random Forest model.

Our future work will be on optimization, finalizing the building an user friendly application.

Github:

https://github.com/shyambeeram/STAR