H1-B Visa Status Prediction Analysis

Questions

- Can we predict H1B visa status using applicant information?
- Which employers send the most H1B visa applications?
- What is the percentage ratio of rejections and acceptance?
- What are the year-wise trends for H1B visa applications and employers?
- Which factors play a significant role in the acceptance of the H1B visa lottery system?
- How does wage impact the H1B visa case status?

Purpose

- The purpose of this project is to develop a prediction model that can accurately predict the outcome of an H1B visa application.
- The model will use machine learning algorithms to **analyze data** related to previous H1B visa applications and **predict** the likelihood of **approval** or **denial** for new applications.

Dataset Used:

- Dataset has data based on 6 year H1-B application data.
- It has 3002458 rows and 11 columns
- Columns used are:
- CASE_STATUS (Target Column), EMPLOYER_NAME, SOC_NAME, JOB_TITLE, FULL_TIME_POSITION, PREVAILING_WAGE, YEAR, WORKSITE, lon, lat

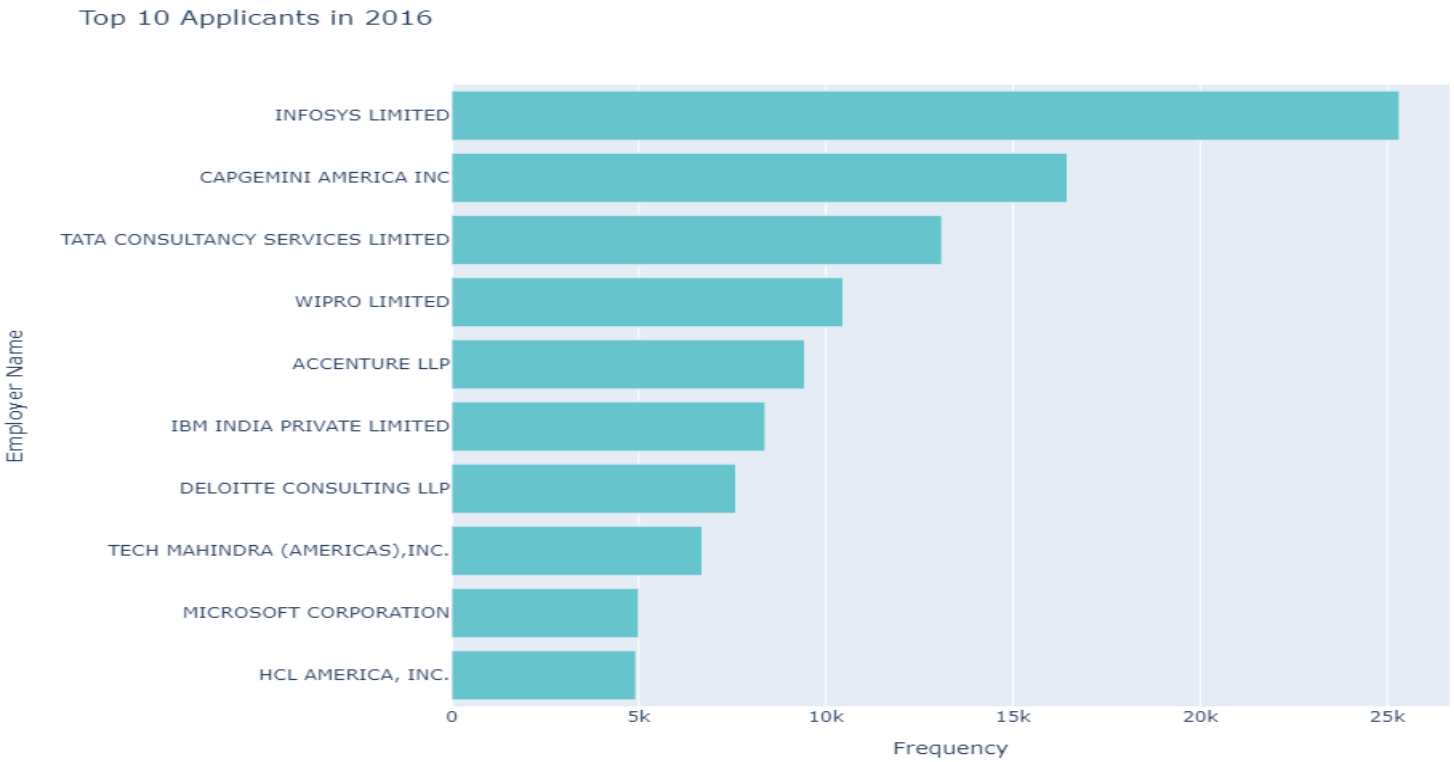
Target column Data

CERTIFIED	2615623
CERTIFIED-WITHDRAWN	202659
DENIED	94346
WITHDRAWN	89799
PENDING QUALITY AND COMPLIANCE REVIEW - UNASSIGNED	15
REJECTED	2
INVALIDATED	1

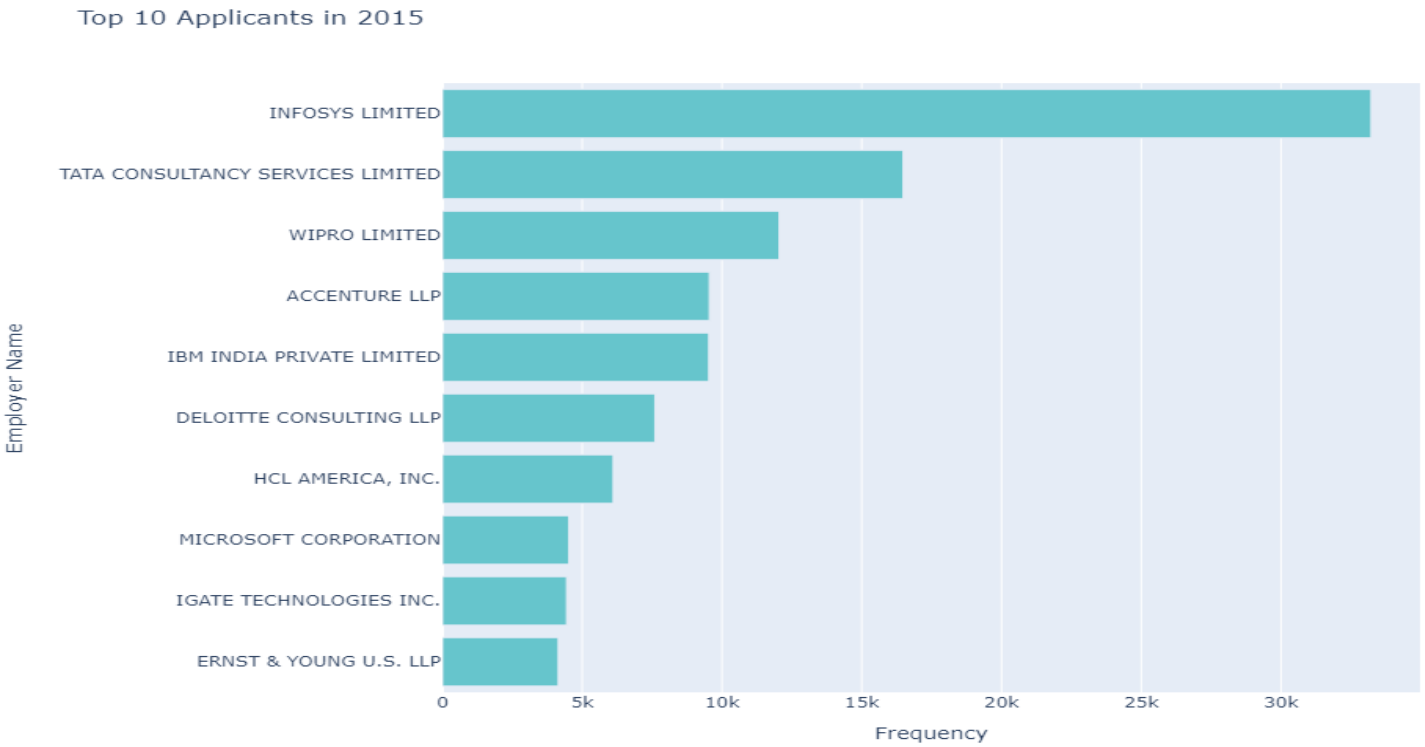
Data Cleaning Methods

- **Imputation** data cleaning method is used to replace the NaN values with a suitable value such as the mean, median, or mode of the column.
- For string columns, same method is used to replace them with a string value such as "**unknown**" or "**missing**".
- Used **string matching** to clean up the 'SOC_NAME' column in the given dataframe replace the values with the department names such as IT, Agriculture, Management, Marketing etc.

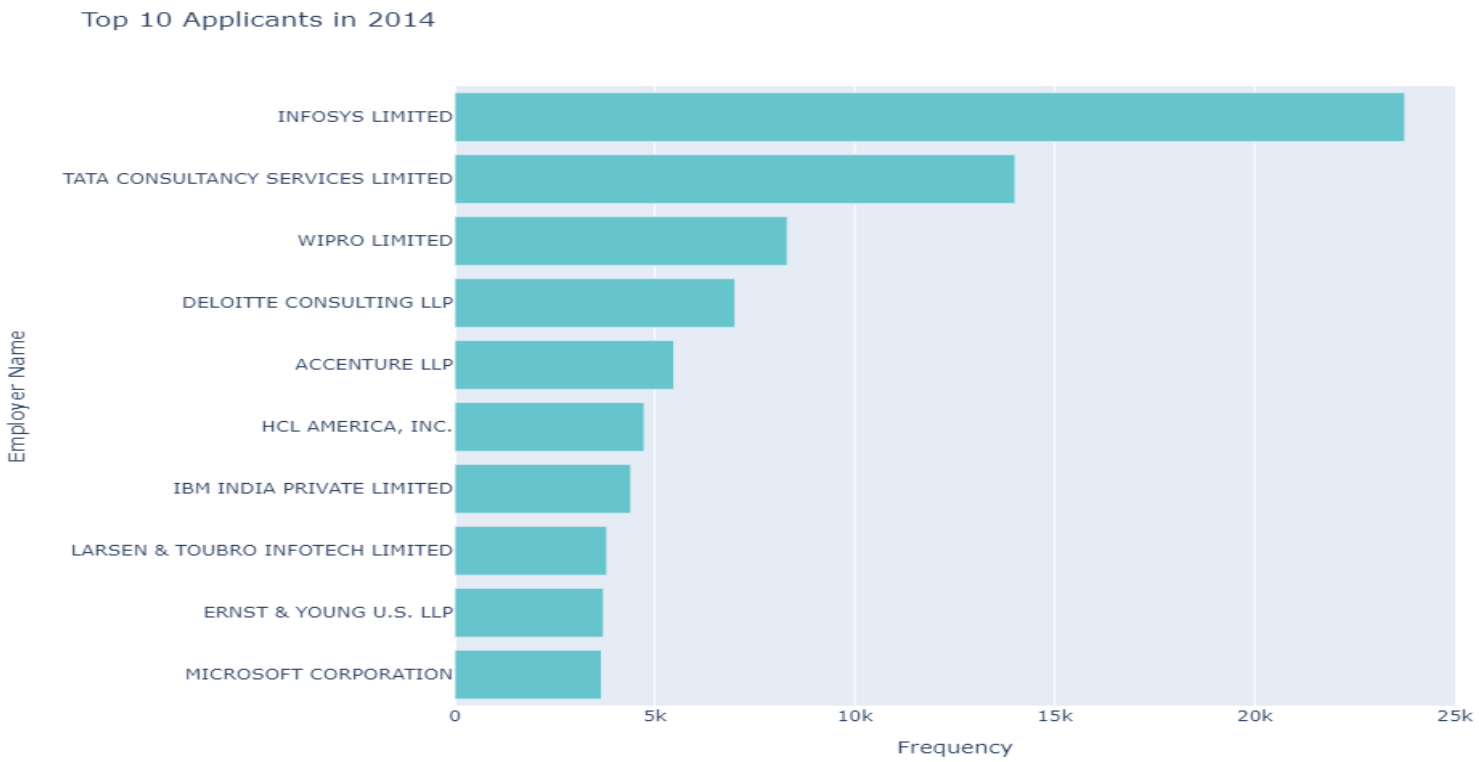
2016 Top 10 Employers



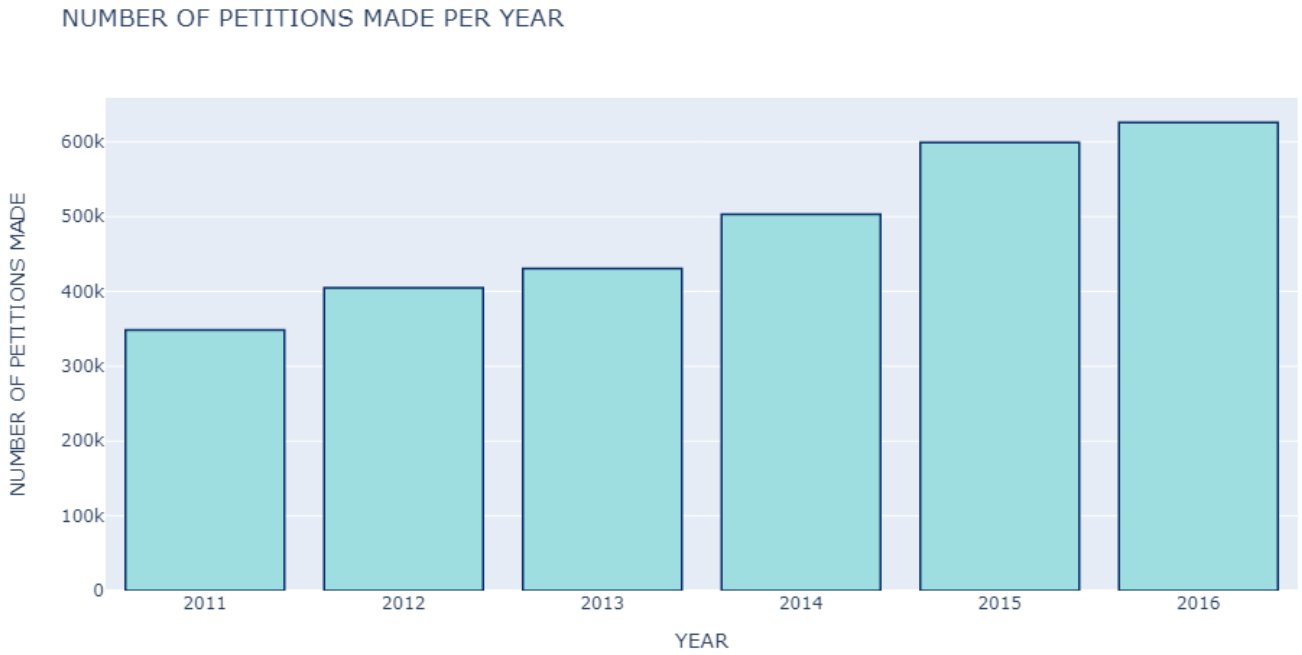
2015 Top 10 Employers



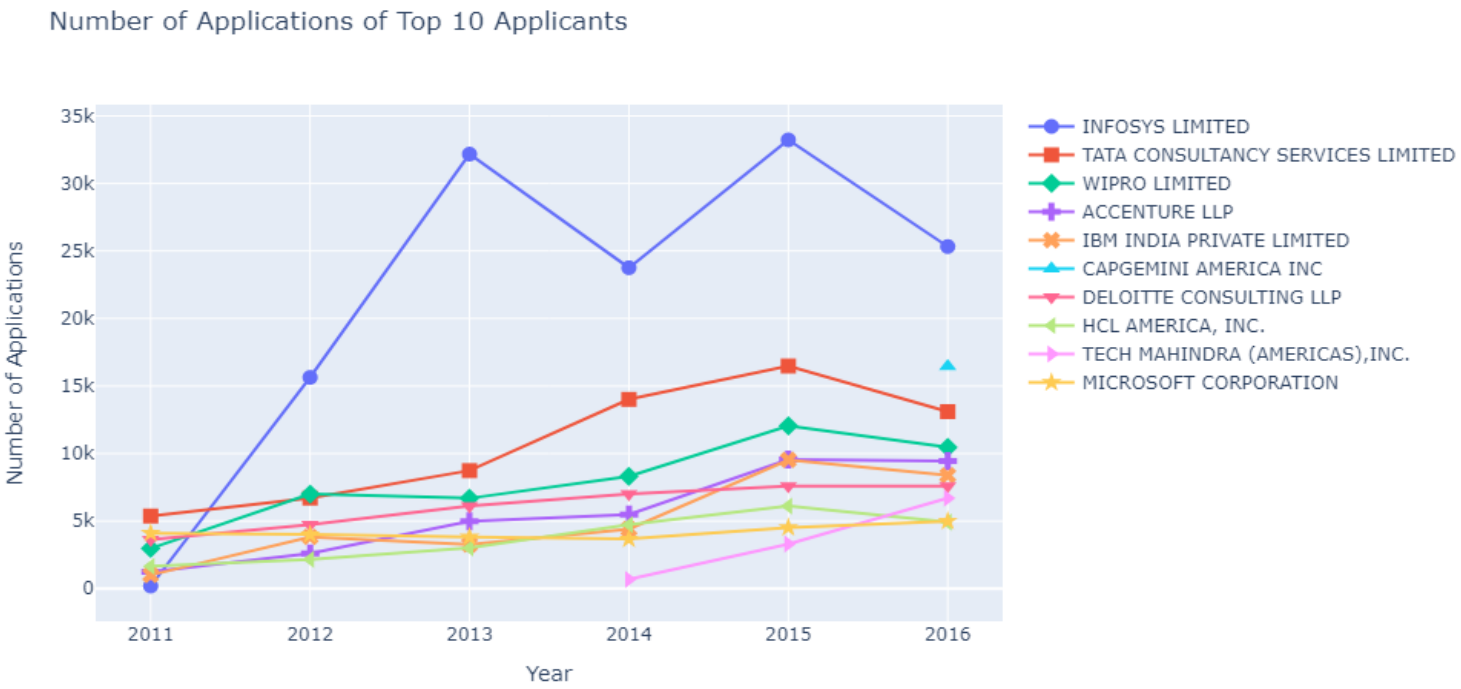
2014 Top 10 Employers



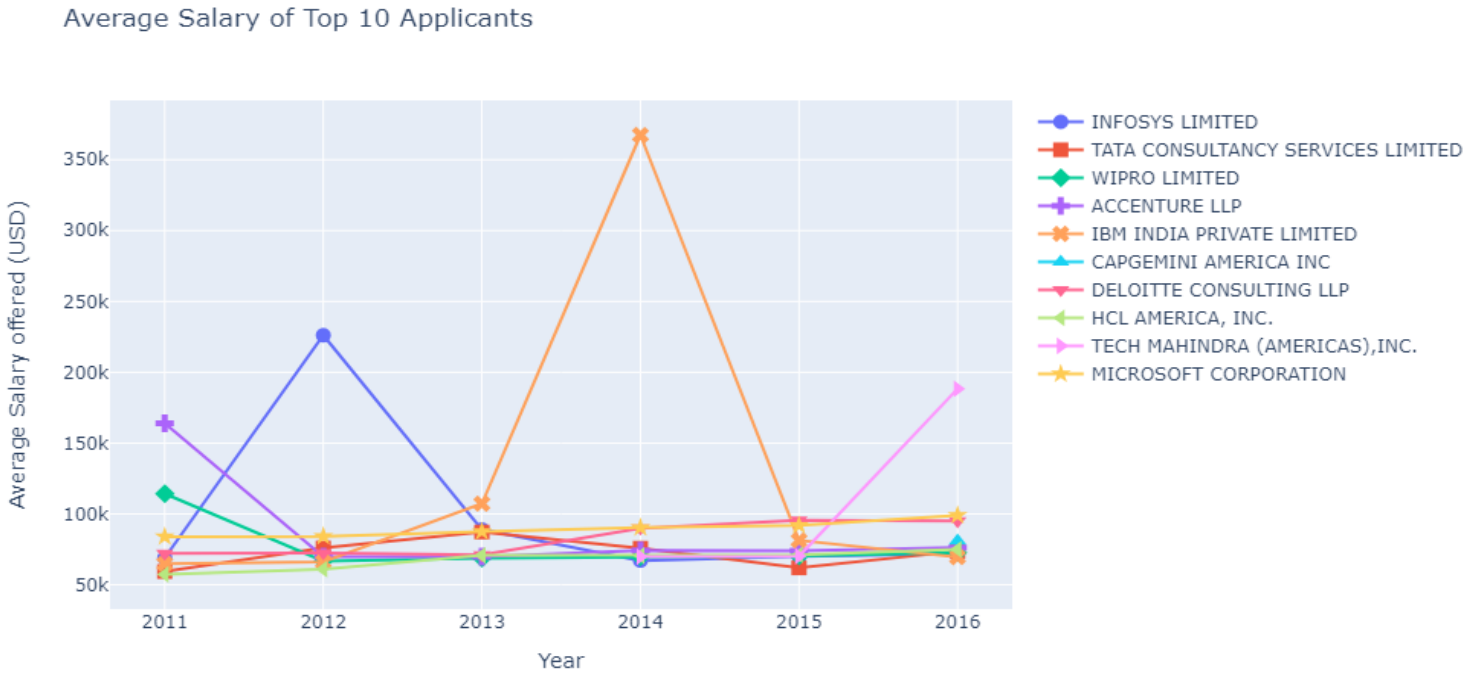
Number of applications each year



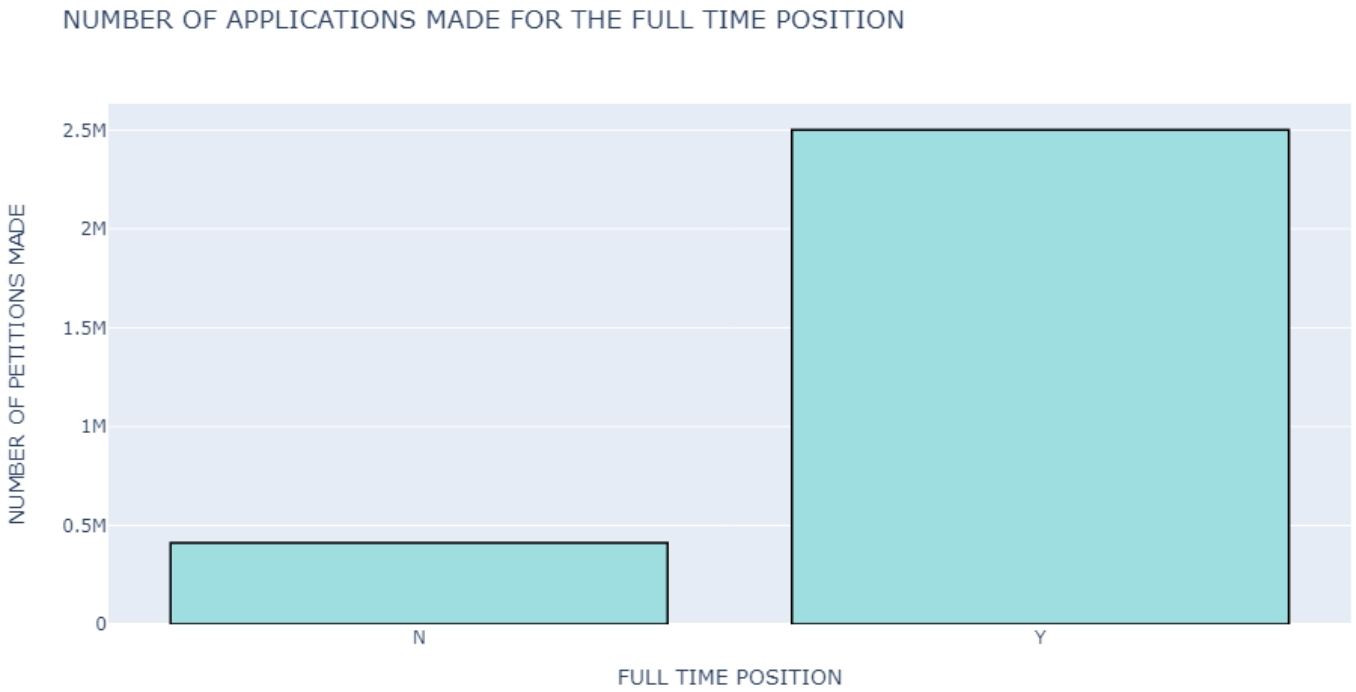
Number of Applications of Top 10 Applicants



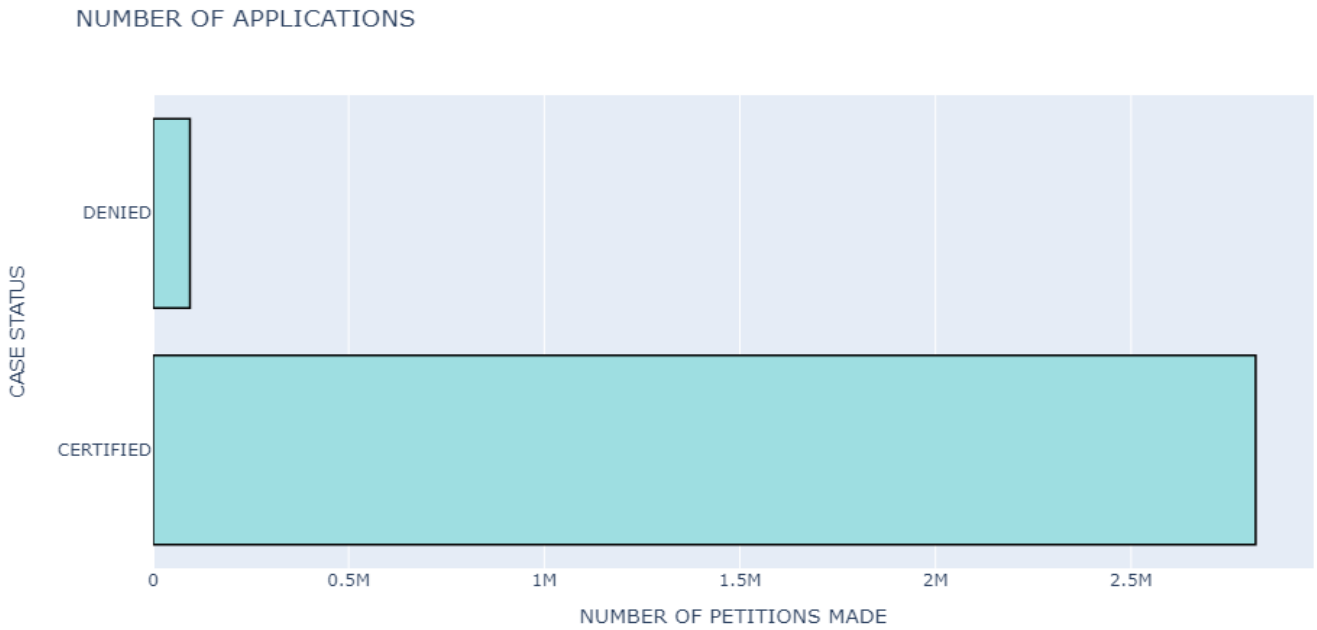
Average Salary of Top 10 Applicants



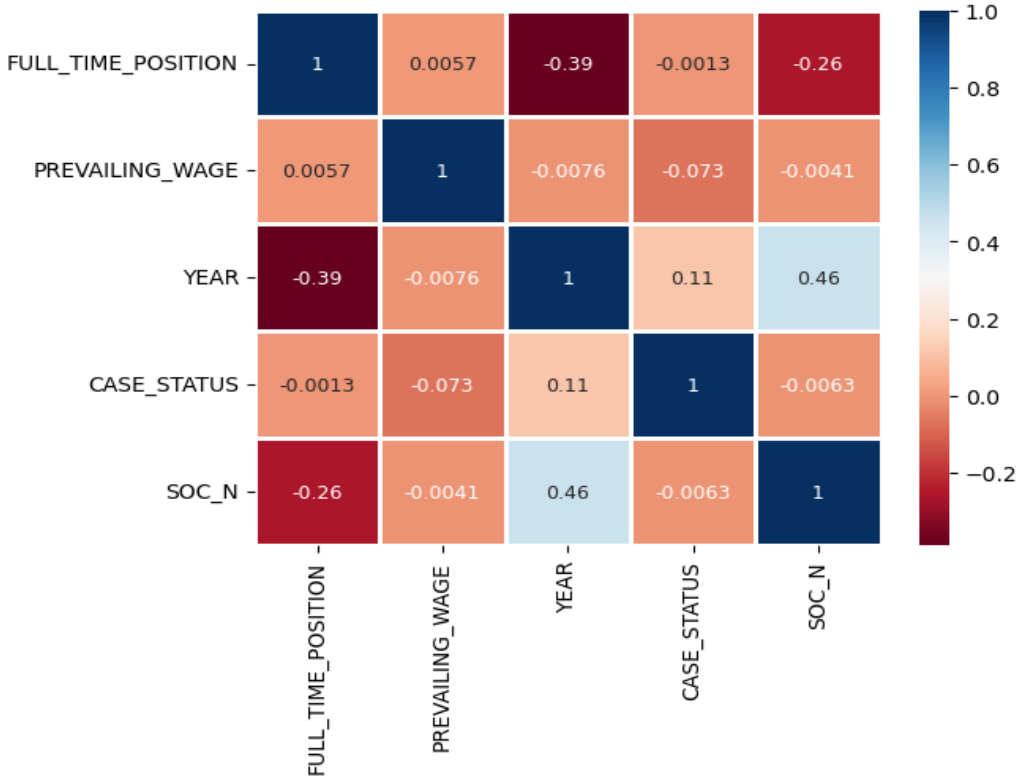
Full Time Positions: **'N'** and **'Y'**



Certified Vs Denied



Seaborn Heat Map to understand the positive and negative correlation



Removing Outliers

Method Used: Z-score

Before applying : 0.9679485418826776
After Applying: 0.9685945068204025

There is increase of approximately 0.001% accuracy with Logistic Regression

Data is highly imbalanced

- Initial Confusion Matrix:

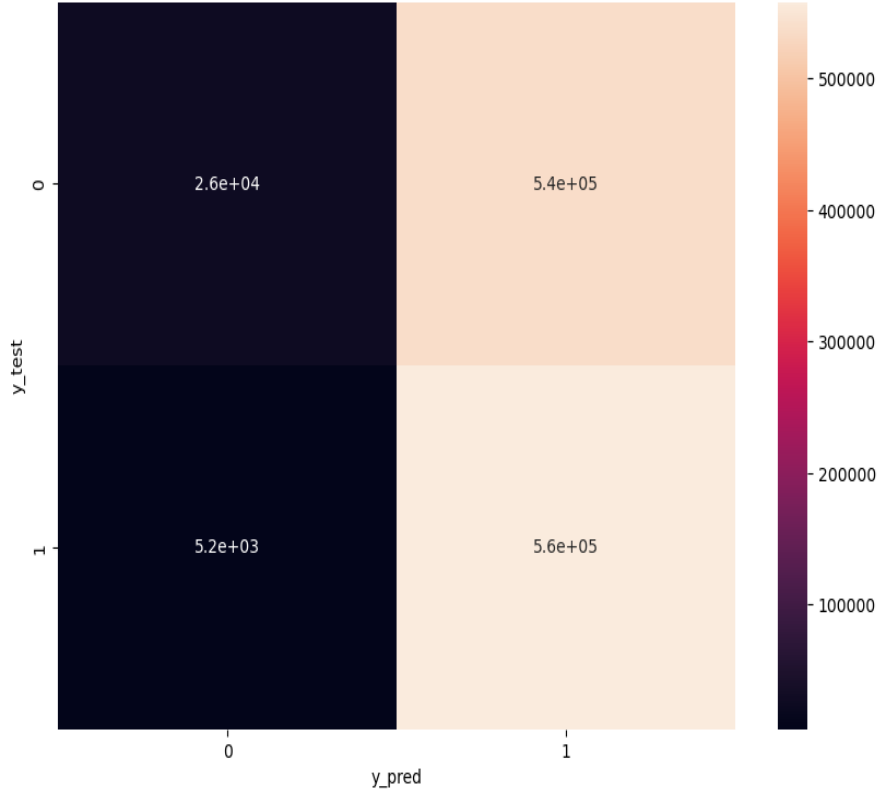
```
array([[    42,  18064],
       [     0, 557080]], dtype=int64)
```

- Later Confusion Matrix:

```
array([[ 26376, 537496],
       [  5163, 558283]], dtype=int64)
```

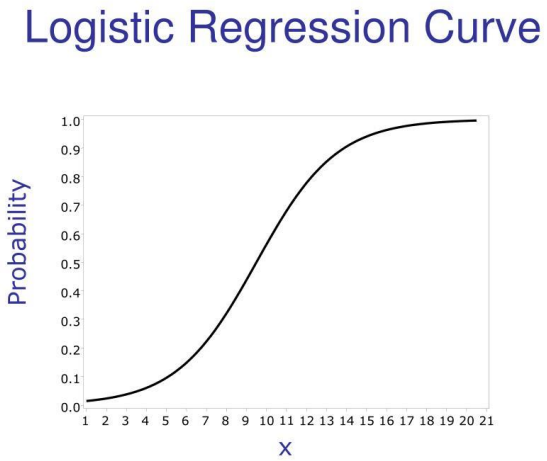
- After Upsampling:

	precision	recall	f1-score	support
0	0.84	0.05	0.09	563872
1	0.51	0.99	0.67	563446



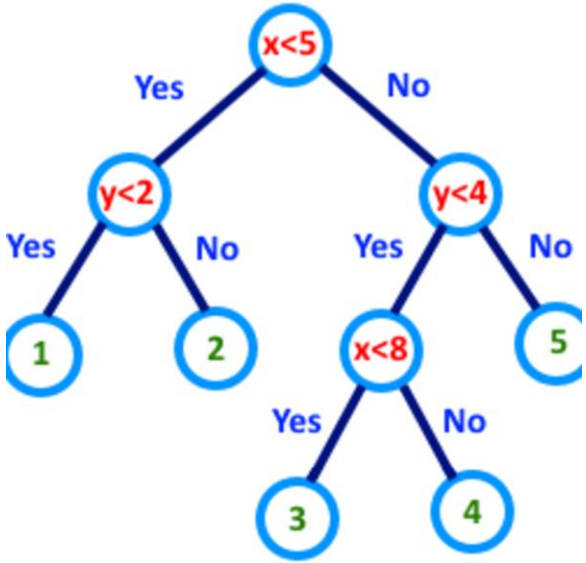
Logistic Regression

	precision	recall	f1-score	support
0	0.84	0.05	0.09	563872
1	0.51	0.99	0.67	563446
accuracy			0.52	1127318
macro avg	0.67	0.52	0.38	1127318
weighted avg	0.67	0.52	0.38	1127318



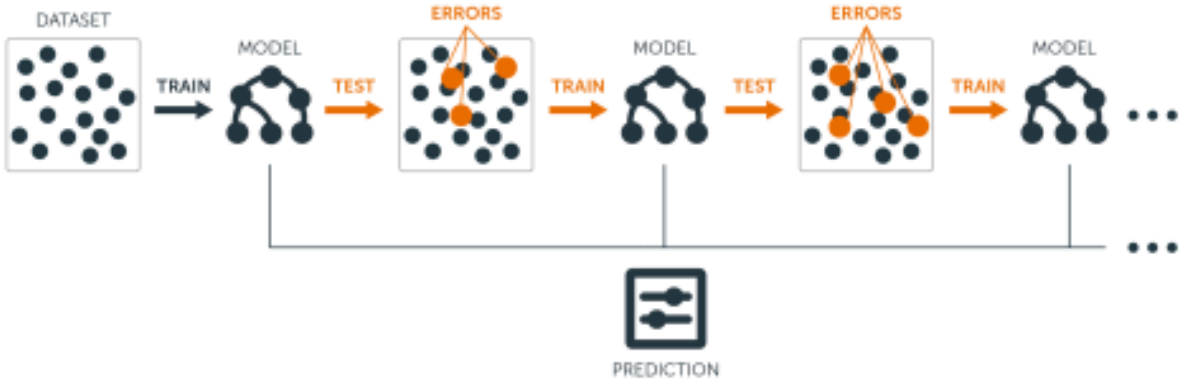
Decision Tree Classifier

	precision	recall	f1-score	support
0	0.81	0.80	0.80	563872
1	0.80	0.81	0.81	563446
accuracy			0.81	1127318
macro avg	0.81	0.81	0.81	1127318
weighted avg	0.81	0.81	0.81	1127318



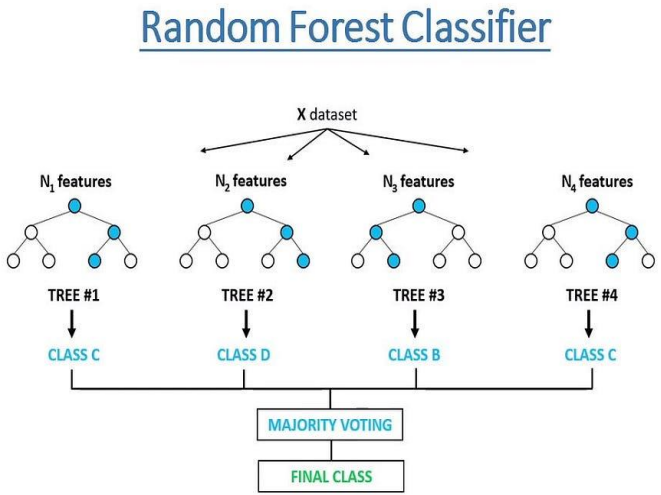
Gradient Boosting Classifier

	precision	recall	f1-score	support
0	0.71	0.61	0.66	563872
1	0.66	0.75	0.70	563446
accuracy			0.68	1127318
macro avg	0.69	0.68	0.68	1127318
weighted avg	0.69	0.68	0.68	1127318



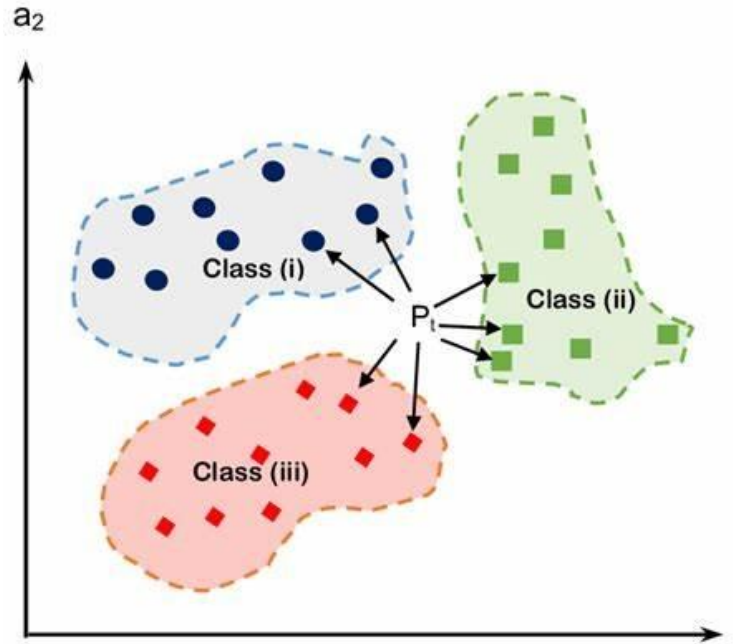
Random Forest

	precision	recall	f1-score	support
0	0.81	0.80	0.80	563872
1	0.80	0.81	0.81	563446
accuracy			0.81	1127318
macro avg	0.81	0.81	0.81	1127318
weighted avg	0.81	0.81	0.81	1127318



K Neighbors Classifier

	precision	recall	f1-score	support
0	0.77	0.79	0.78	563872
1	0.78	0.76	0.77	563446
accuracy			0.77	1127318
macro avg	0.77	0.77	0.77	1127318
weighted avg	0.77	0.77	0.77	1127318

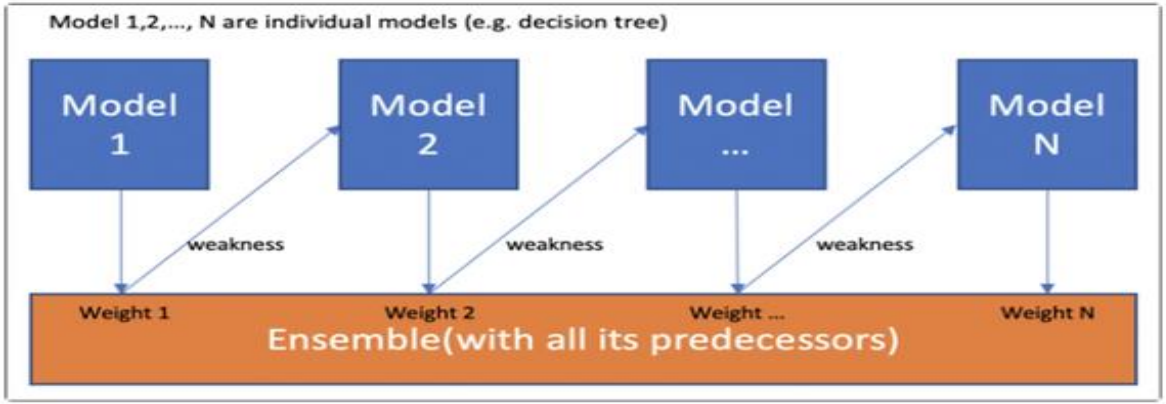


Gaussian NB

	precision	recall	f1-score	support
0	1.00	0.02	0.04	563872
1	0.50	1.00	0.67	563446
accuracy			0.51	1127318
macro avg	0.75	0.51	0.35	1127318
weighted avg	0.75	0.51	0.35	1127318

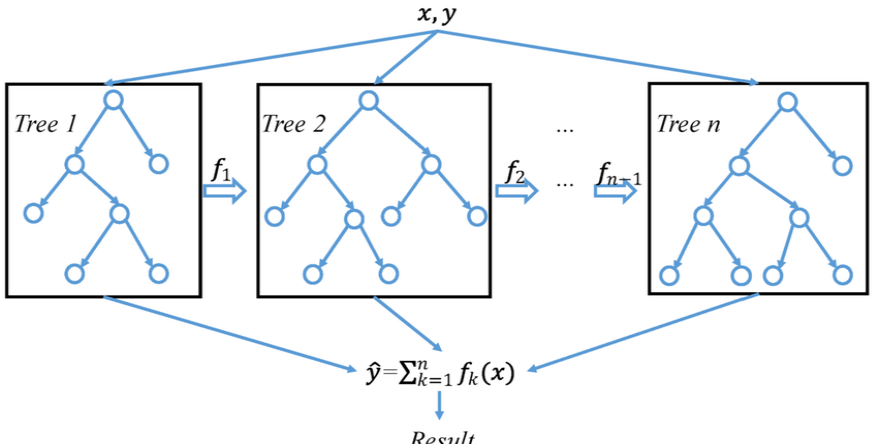
AdaBoost Classifier

	precision	recall	f1-score	support
0	0.71	0.61	0.66	563872
1	0.66	0.74	0.70	563446
accuracy			0.68	1127318
macro avg	0.68	0.68	0.68	1127318
weighted avg	0.68	0.68	0.68	1127318



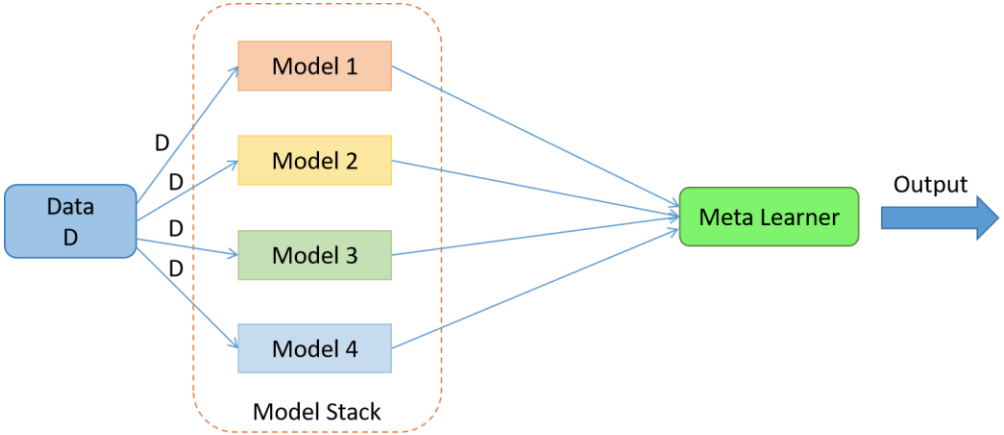
XGBoost Classifier

	precision	recall	f1-score	support
0	0.71	0.66	0.68	563872
1	0.68	0.73	0.70	563446
accuracy			0.69	1127318
macro avg	0.69	0.69	0.69	1127318
weighted avg	0.69	0.69	0.69	1127318



Stacking

- The key idea behind Stacking is to leverage the **strengths of each individual model** and to reduce their **weaknesses** by combining their predictions. This can lead to a more accurate and robust predictive model than any single model on its own.
- **Models considered:** Decision Tree, Random Forest and XGBoost
- Accuracy: 67%



Thank You!

To all the international students here,

Good Luck with your H1-B lottery in future.