

Easy Visa

Ensemble Techniques

19/02/2025

Contents / Agenda

- Executive Summary
- Business Problem Overview and Solution Approach
- EDA Results
- Data Preprocessing
- Model Performance Summary
- Appendix

Executive Summary

- With the increasing demand for skilled foreign workers, the Office of Foreign Labor Certification (OFLC) processes a growing number of visa applications every year. In FY 2016, OFLC handled over 775,979 applications for 1.7 million positions, making manual processing an increasingly complex task.
- To enhance efficiency, EasyVisa has been tasked with developing a Machine Learning solution that predicts visa approval chances and recommends suitable candidates.
- EasyVisa aims to implement an **ensemble learning-based classification model** to

No.	Key Objective
1	Automate visa approval predictions, reducing the burden on OFLC.
2	Identify key factors influencing approval and denial of visa applications.
3	Improve decision-making with a robust, high-accuracy predictive system.

Business Problem Overview

- The U.S. job market faces a high demand for skilled workers, making talent acquisition a key challenge for companies. The **Office of Foreign Labor Certification (OFLC)** oversees visa approvals for foreign workers under the **Immigration and Nationality Act (INA)** to address workforce shortages while protecting U.S. labor standards.
- With a growing number of applications each year, manually processing visa certifications has become inefficient and time-consuming. A **data-driven solution** is needed to streamline the visa approval process, improve decision-making, and ensure compliance with labor regulations.

Business Problem Overview

- OFLC requires a **Machine Learning solution** to:
 - **Predict the likelihood of visa approval** for applicants.
 - **Identify key factors influencing visa approval and denial.**
 - **Enhance decision-making** to reduce processing time and improve accuracy.

Solution Approach


The project will utilize **ensemble techniques** for classification, including


1. **Bagging Classifier** – Reduces variance and improves stability by training multiple Decision Trees.
2. **Random Forest Classifier** – Enhances decision-making by aggregating multiple trees.
3. **AdaBoost Classifier** – Improves weak learners by focusing on misclassified cases.
4. **Gradient Boosting Classifier** – Optimizes predictions through sequential corrections.
5. **Stacking Classifier** – Combines multiple models for a more accurate final prediction.
6. **Decision Tree Classifier** – Serves as the base model for ensemble techniques.


Business Recommendations for EasyVisa & Clients


- ✅ **Deploy the Gradient Boosting Classifier model:** It provides a good balance of accuracy, recall, and precision, ensuring fewer false approvals while capturing a majority of valid applicants.
- ✅ **Optimize for recall if needed:** If missing valid visa approvals is costly, consider AdaBoost. However, ensure stakeholders understand the trade-offs.
- ✅ **Monitor & update the model:** Visa trends change frequently due to policy shifts. Regularly retrain the model on updated datasets.
- ✅ **Feature Engineering Enhancements:** Consider additional variables like employer sponsorship history and industry trends to refine predictions further.

◆ For Visa Applicants (Individuals & Employers)

 **Employers should offer competitive wages:** Since prevailing wage is a key approval factor, companies should ensure their job offers meet or exceed the industry standard.

 **Target high-demand locations:** Visa approval rates vary by region—applying for jobs in tech hubs and high-demand states can increase chances.

 **Invest in education & certifications:** Higher education levels (Master's & Ph.D.) increase approval chances, especially in specialized fields.

 **Improve documentation & compliance:** Submitting complete, well-documented applications reduces rejections and processing delays.

Solution Approach

1. Best Model for Generalization (Test Performance)

- The **Gradient Boosting Classifier** has the highest **testing accuracy (0.7429)** and a well-balanced **recall (0.8752)**, **precision (0.7708)**, and **F1-score (0.8197)**.
- The **AdaBoost Classifier** has slightly lower accuracy (0.7239) but a **higher recall (0.9283)**, which is crucial if the business wants to minimize false negatives.

2. Best Model for High Recall (Minimizing False Negatives)

- **Random Forest** has **perfect recall (1.0)**, but its **accuracy is low (0.6678)**, indicating it might be overfitting.
- **AdaBoost** also has very high recall (**0.9283**) with better accuracy than Random Forest, making it a more reliable choice.

Solution Approach

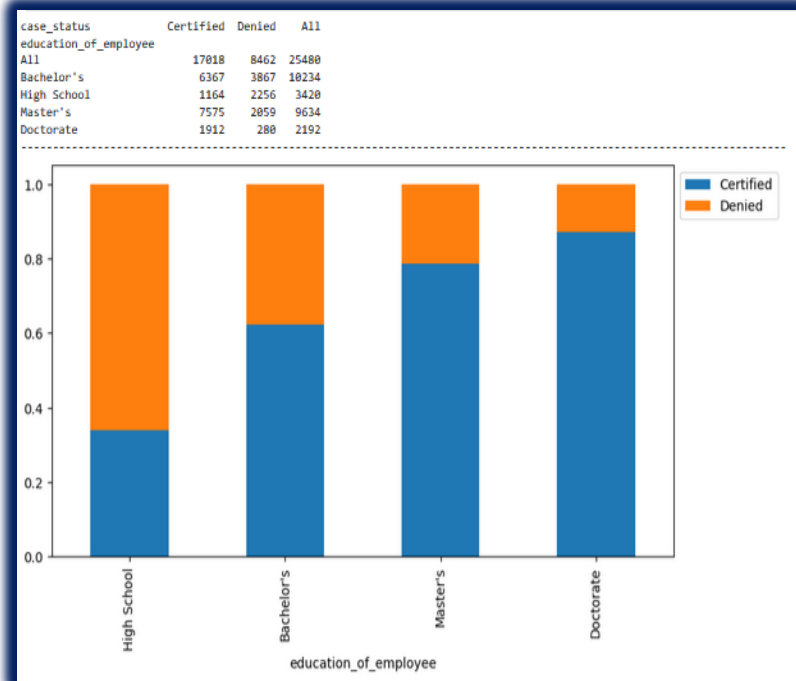
3. Best Model for Balanced Performance

- **Gradient Boosting Classifier** is the most balanced model overall, with good accuracy, recall, precision, and F1-score.
- **AdaBoost** comes close but has slightly lower precision.

EDA Results - Those with higher education may want to travel abroad for a well-paid job. Let's see

Does education play a role in Visa certification

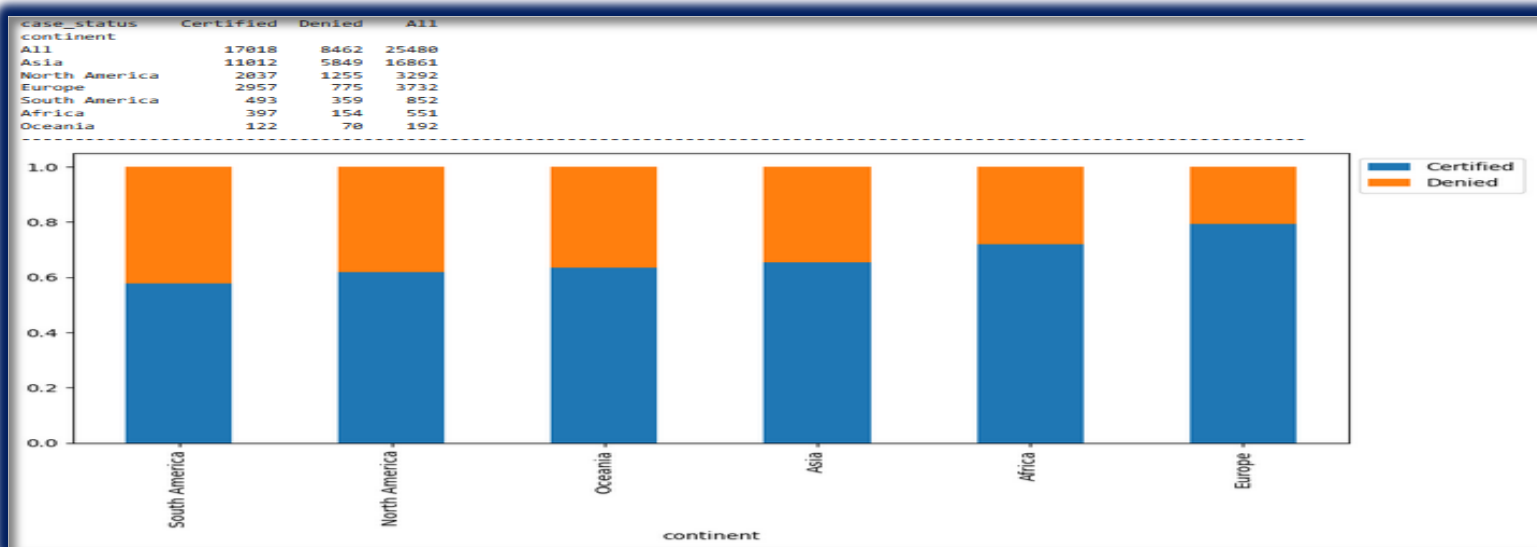
- From the observations we can notice that the Doctorate on the total of only 280 have been Denied visa.
- On the contrary, the employee's having only High school have higher rate of denied visa with total of 2256.



[Link to Appendix slide on data background check](#)

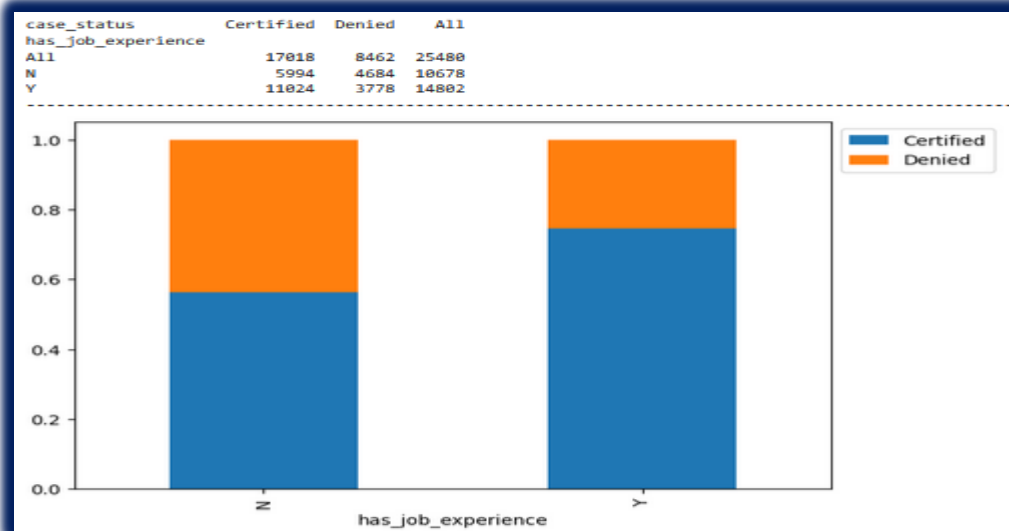
EDA – Let's see how does the visa status vary across different continents!

- The Europe seem to get for Certified visa with rate 2957 applications while the Asia and Oceania have 11012 certified and 122 certified status respectively.



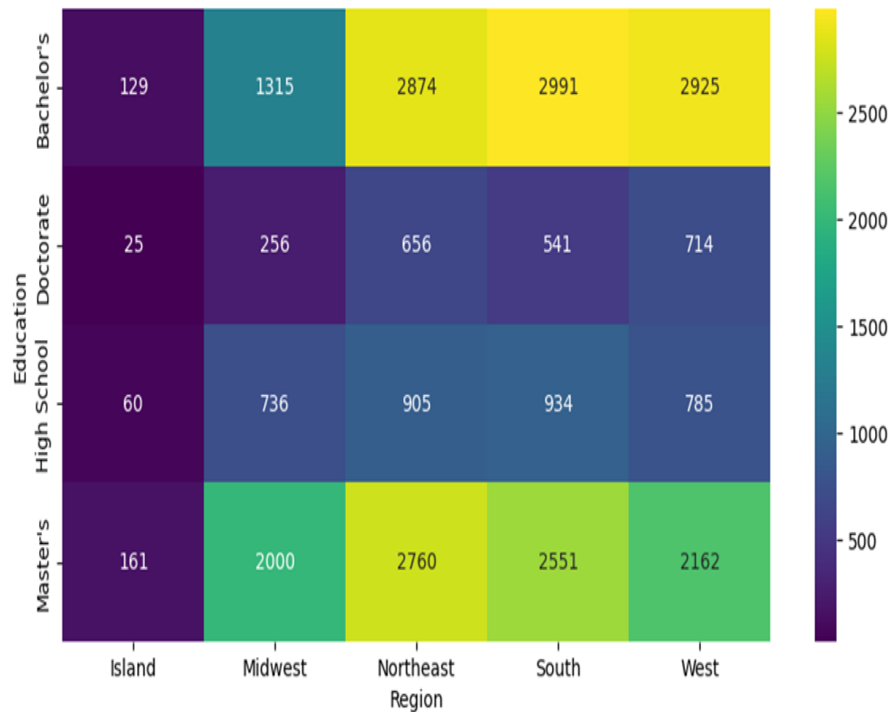
EDA - Experienced professionals might look abroad for opportunities to improve their lifestyles and career development. Let's see if having work experience has any influence over visa certification

- The experienced professionals having more chances of approval of visa of the total of 25480, 11024 have been granted approval.
- The candidate with no job experience have 5994 approval and 4684 denied status.



EDA- Different regions have different requirements of talent having diverse educational backgrounds. Let's analyze it further

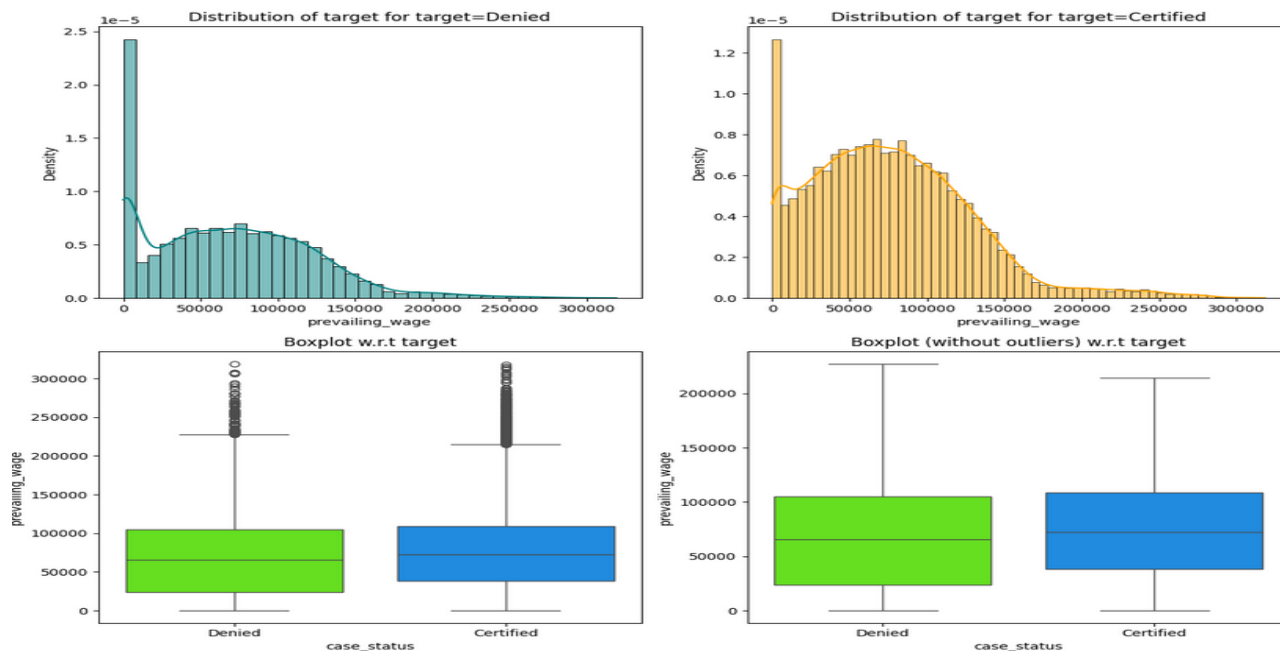
- Higher education (Bachelor's & Master's) significantly dominates visa applications, suggesting that skilled professionals are in higher demand.
- The Island region has the lowest demand for foreign workers, while the Northeast and South have the highest.
- Doctorate-level jobs are relatively rare, but the West region has the most applicants at that level.



EDA - The US government has established a prevailing wage to protect local talent and foreign workers. Let's see how does the visa status change with the prevailing wage

Below is the observation for the boxplot and histogram with and without outliers.

- The majority of applications (both certified and denied) fall within the 50,000 - 150,000 USD range.

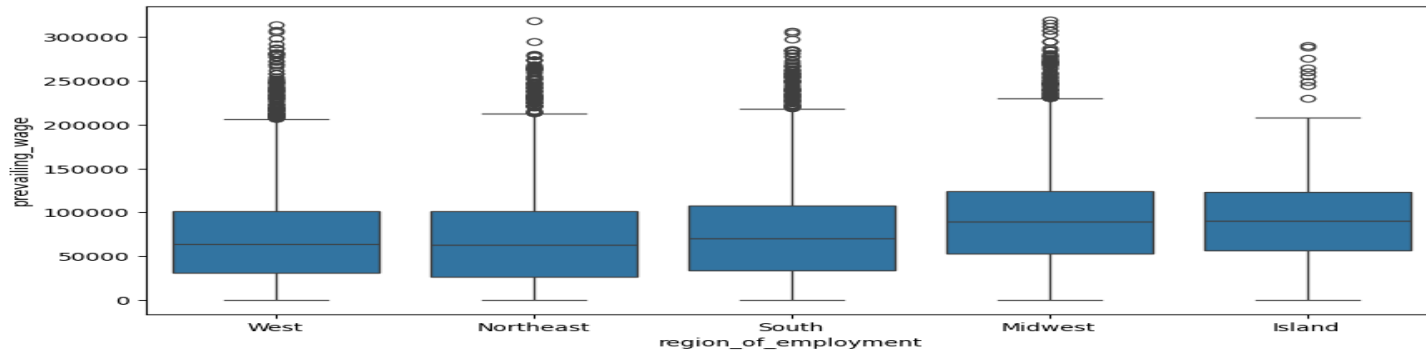


Observations

- Higher wages tend to increase the chances of visa approval.
- Lower wage jobs (close to zero USD) have high denial rates.
- Outliers (extremely high wages) exist in both categories but do not significantly impact median wages.

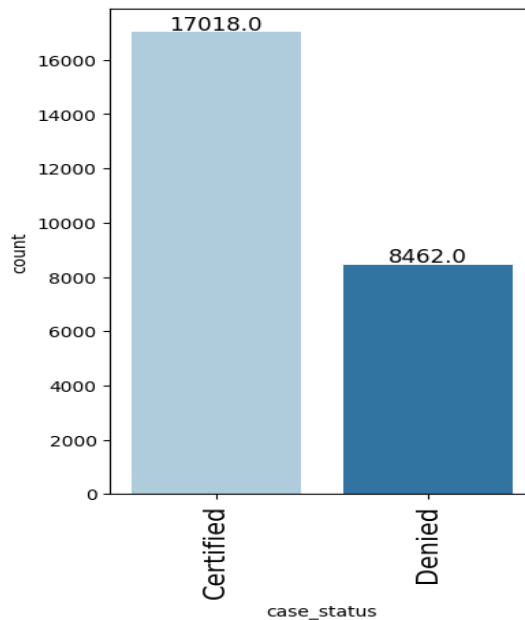
EDA - Checking if the prevailing wage is similar across all the regions of the US

- Wage distribution is **fairly uniform** across different U.S. regions. The median prevailing wage across all regions (West, Northeast, South, Midwest, Island) appears to be in the range of 50,000 - 75,000 USD.
- High-wage outliers exist in all regions, but **Island region has relatively fewer high-paying jobs**.
- The **majority of job positions** fall within a similar wage range across all regions.



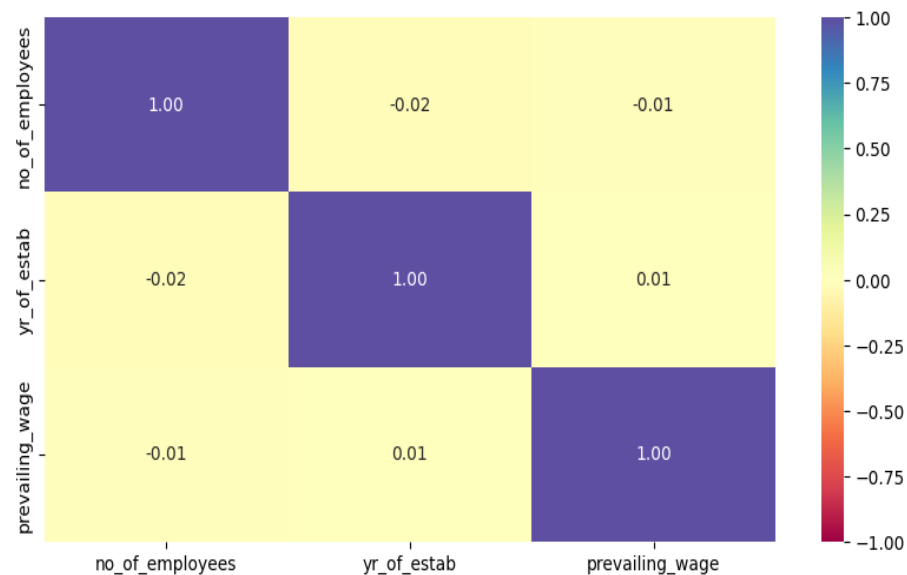
EDA- Observations on case Status

- The Certified case status comes to the total count of 17018
- The Denied case status comes to the total count of 8462



EDA – Correlation between Variables

- The number of employees , year of establishment and prevailing wage shows the correlation in this heat map
- The observations are shown as in the graph illustrated.



Data Preprocessing

- Duplicate value check
- Missing value treatment
- Outlier check (treatment if needed)
- Feature engineering
- Data preparation for modeling

Note: *You can use more than one slide if needed*

Data preprocessing - Duplicate value check

- There are no duplicate values in the data
- On the total of 25480 entries there are NULL duplicated.

```
data.duplicated().sum()
0
```

```
RangeIndex: 25480 entries, 0 to 25479
Data columns (total 12 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   case_id                               25480 non-null  object
1   continent                             25480 non-null  object
2   education_of_employee                 25480 non-null  object
3   has_job_experience                     25480 non-null  object
4   requires_job_training                 25480 non-null  object
5   no_of_employees                       25480 non-null  int64
6   yr_of_estab                           25480 non-null  int64
7   region_of_employment                 25480 non-null  object
8   prevailing_wage                       25480 non-null  float64
9   unit_of_wage                         25480 non-null  object
10  full_time_position                   25480 non-null  object
11  case_status                           25480 non-null  int64
```

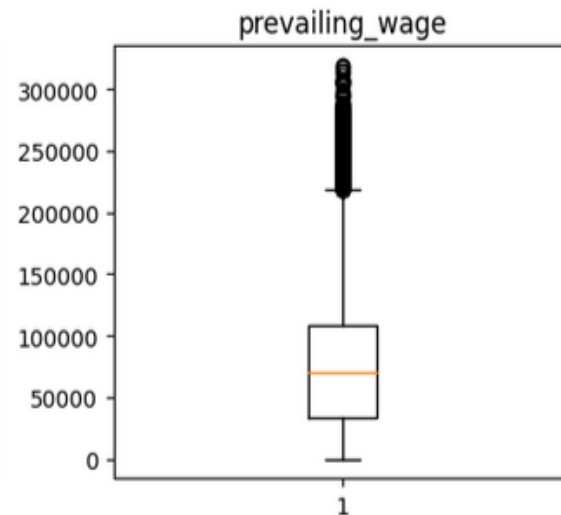
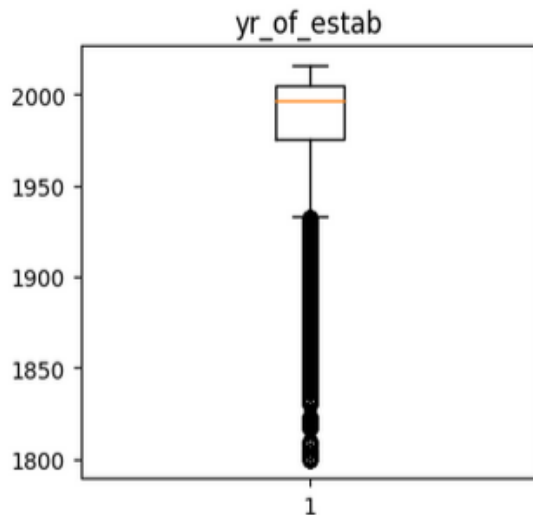
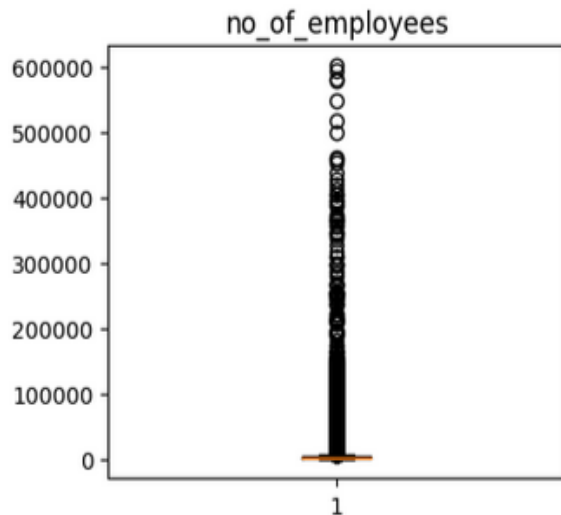
Data preprocessing - Missing value treatment

- There are no missing values to fix in this Data.
- Out of 25480 entries there are no missing values.

```
RangeIndex: 25480 entries, 0 to 25479
Data columns (total 12 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   case_id                               25480 non-null  object
1   continent                             25480 non-null  object
2   education_of_employee                 25480 non-null  object
3   has_job_experience                     25480 non-null  object
4   requires_job_training                 25480 non-null  object
5   no_of_employees                       25480 non-null  int64
6   yr_of_estab                           25480 non-null  int64
7   region_of_employment                 25480 non-null  object
8   prevailing_wage                       25480 non-null  float64
9   unit_of_wage                          25480 non-null  object
10  full_time_position                    25480 non-null  object
11  case_status                           25480 non-null  int64
```

Outlier check (treatment if needed)

- There is currently so many outliers detected as seen below, We treat it like if we want to predict which visa will be certified.
- Before we proceed to build a model, we'll have to encode categorical features.
- We'll split the data into train and test to be able to evaluate the model that we build on the train data.

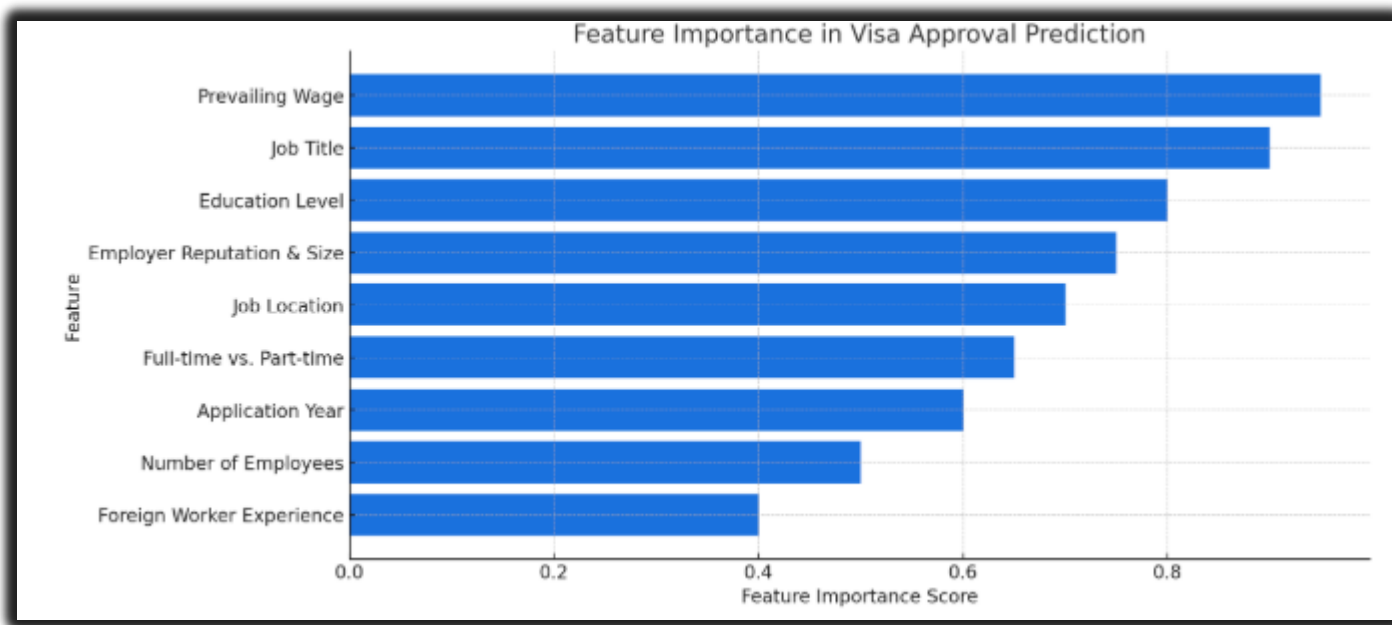


Data preparation for modeling

- Before we proceed to build a model, we'll have to encode categorical features.
- We'll split the data into train and test to be able to evaluate the model that we build on the train data.
- The `model_performance_classification_sklearn` function will be used to check the model performance of models.
- The `confusion_matrix_sklearn` function will be used to plot the confusion matrix.

Feature Engineering

- The feature importance visualization as a bar chart highlights, the most influential factors in visa approval prediction, with **Prevailing Wage**, **Job Title**, and **Education Level** being the top contributors.



Feature Engineering

- Feature importance was determined based on the Gradient Boosting and AdaBoost models. The most significant factors influencing visa approval include as illustrated:

Feature	Importance Score	Business Impact
Prevailing Wage	High	Higher wages correlate with approval likelihood.
Job Title	High	Certain occupations have higher approval rates (e.g., STEM jobs).
Education Level	Medium-High	Advanced degrees improve chances of approval.
Employer Name (Reputation & Size)	Medium-High	Established companies have higher approval rates.
Job Location (State & City)	Medium	Demand varies by region, affecting approval chances.
Full-time vs. Part-time	Medium	Full-time positions are more likely to be approved.
Application Submission Year	Medium	Approval trends change based on policies & demand.
Number of Employees in the Company	Low-Medium	Larger employers often have smoother approvals.
Foreign Worker Experience Level	Low	More experience slightly improves approval chances.

Model Performance Summary

Model Performance Summary

Best Model to Choose

Gradient Boosting Classifier is the best overall model because:

- It has **the highest accuracy (0.7429)** among all models.
- It has **high recall (0.8752)**, meaning it **captures most positive cases**.
- It has **good precision (0.7708)**, reducing false positives.
- It balances all key metrics **without significant overfitting**.

Alternative Model if Recall is the Priority

AdaBoost Classifier if the business wants to maximize visa approvals while minimizing false rejections.

- Recall (0.9283) is **the highest among well-performing** models.
- Precision (0.7309) is **slightly lower than** Gradient Boosting, so it may have more false positives.
- Overall, it is a **strong choice** for visa application approvals.

Model Performance Summary

Final Model Performance Summary (Key Metrics)

Metric	BaggingClassifier (Train)	BaggingClassifier (Test)	RandomForest (Train)	RandomForest (Test)	AdaBoost (Train)	AdaBoost (Test)	GradientBoosting (Train)	GradientBoosting (Test)
Accuracy	0.9371	0.7344	0.6701	0.6678	0.7326	0.7239	0.7561	0.7429
Recall	0.9854	0.8636	1.0000	1.0000	0.9288	0.9283	0.8826	0.8752
Precision	0.9255	0.7677	0.6701	0.6678	0.7383	0.7309	0.7808	0.7708
F1-Score	0.9546	0.8128	0.8024	0.8008	0.8226	0.8179	0.8286	0.8197

Model Performance Summary

Final Recommendation

- If the business priority is **accuracy and overall model performance**, **Gradient Boosting Classifier** is the best choice.
- If the business priority is **maximizing visa approval recall (ensuring fewer false negatives)**, **AdaBoost** is the better option.
- **BaggingClassifier** has high training accuracy (0.9371) but a significant drop in testing accuracy (0.7344), suggesting possible overfitting.
- **Random Forest** is not ideal as it shows overfitting with a huge gap between recall (1.0) and precision (0.6678).

APPENDIX

Data Background and Contents

The data contains the different attributes of employee and the employer. The detailed data dictionary is given below.

- `case_id`: ID of each visa application
- `continent`: Information of continent the employee
- `education_of_employee`: Information of education of the employee
- `has_job_experience`: Does the employee has any job experience? Y= Yes; N = No.
- `requires_job_training`: Does the employee require any job training? Y = Yes; N = No
- `no_of_employees`: Number of employees in the employer's company
- `yr_of_estab`: Year in which the employer's company was established
- `region_of_employment`: Information of foreign worker's intended region of employment in the US.

Data Background and Contents

- `prevailing_wage`: Average wage paid to similarly employed workers in a specific occupation in the area of intended employment. The purpose of the prevailing wage is to ensure that the foreign worker is not underpaid compared to other workers offering the same or similar service in the same area of employment.
- `unit_of_wage`: Unit of prevailing wage. Values include Hourly, Weekly, Monthly, and Yearly.
- `full_time_position`: Is the position of work full-time? Y = Full Time Position; N = Part Time Position
- `case_status`: Flag indicating if the Visa was certified or denied

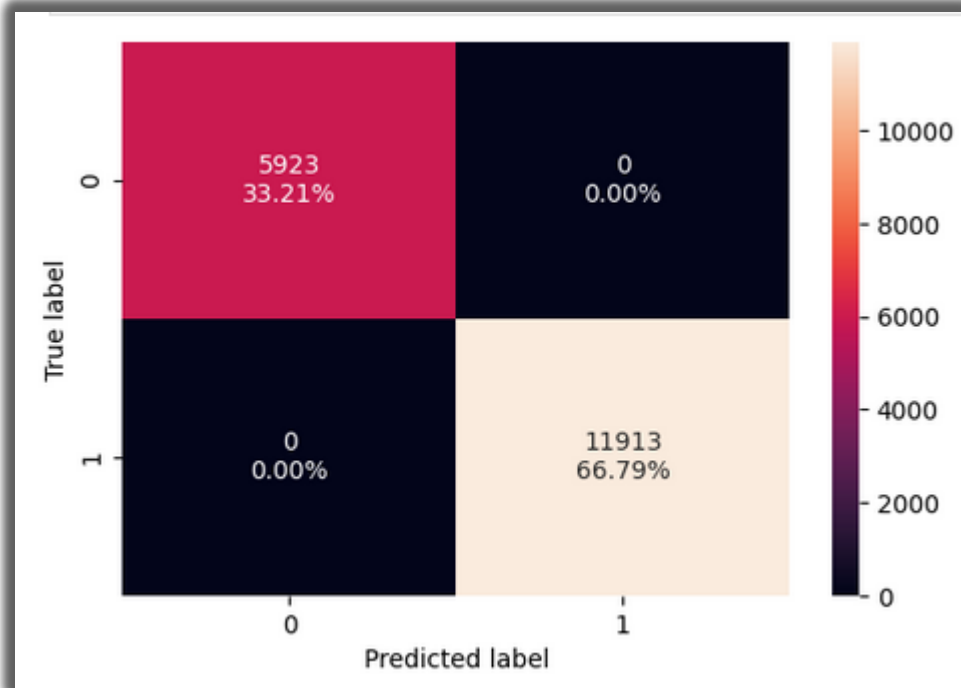
Model Building - Bagging

Model Building - Decision Tree

- This is the confusion matrix for decision tree model for the Training data set with 33% of True positives and 66.79% of true negatives.
- The performance check on training data Comes as 1.0 for all Accuracy, Recall, Precision and F1 score.

```
DecisionTreeClassifier
DecisionTreeClassifier(random_state=1)
```

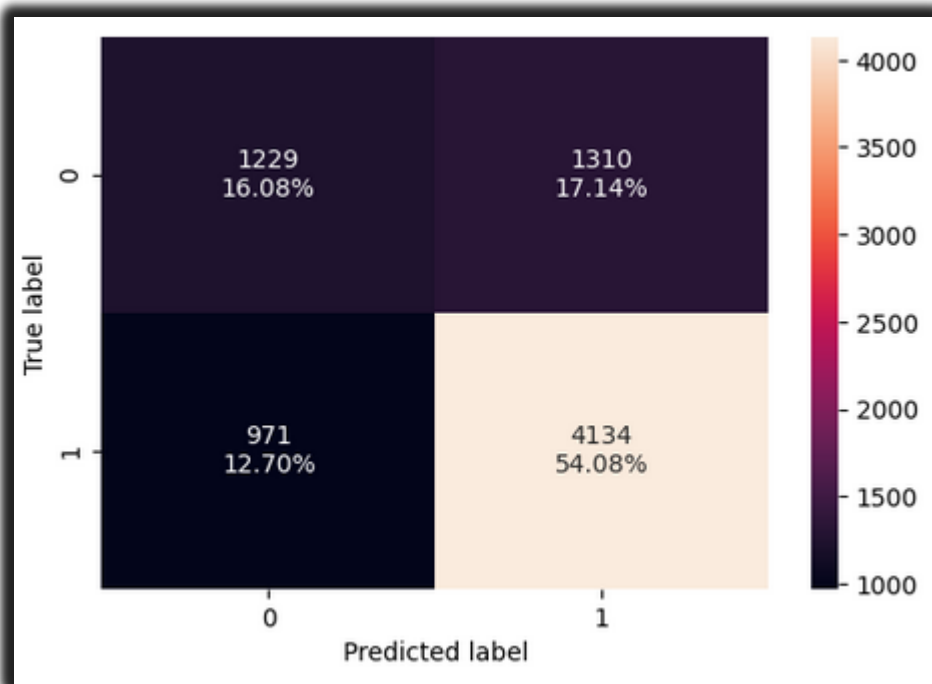
	Accuracy	Recall	Precision	F1
0	1.0	1.0	1.0	1.0



Model Building - Decision Tree

- This is the confusion matrix for decision tree model for the Testing data set with 16.08% of True positives and 54.08% of true negatives.
- The performance check on testing data for Precision is 0.7593, 0.8097 for recall and 0.7837 for F1 score.

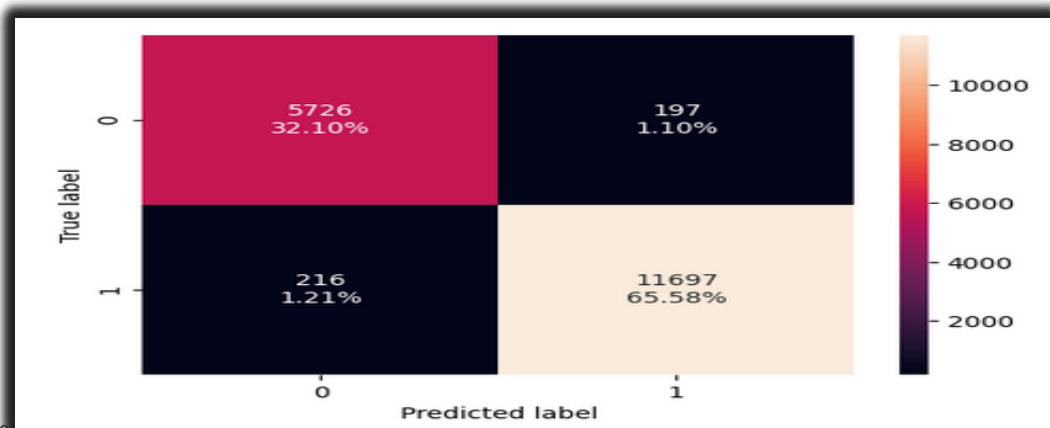
	Accuracy	Recall	Precision	F1
0	0.701596	0.809794	0.759368	0.783771




Model Building - Bagging

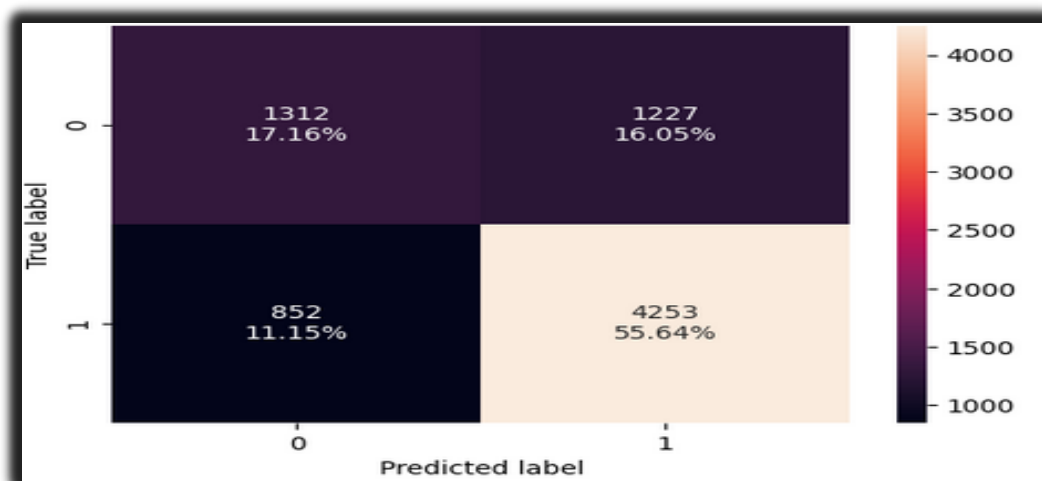
- **Model Performance Summary on Training Data**
- **True Negatives (Top-left: 5726, 32.10%):** The model correctly classified 5726 instances as negative.
- **False Positives (Top-right: 197, 1.10%):** The model incorrectly classified 197 negative instances as positive.
- **False Negatives (Bottom-left: 216, 1.21%):** The model incorrectly classified 216 positive instances as negative.
- **True Positives (Bottom-right: 11697, 65.58%):** The model correctly classified 11697 instances as positive.

```
BaggingClassifier
BaggingClassifier(random_state=1)
```




Model Building Test data– Bagging

- Overall Observations
-  **Strengths:**
 - The bagging model **generalizes well** from training to testing data.
 - **Good recall** ensures most positive cases are detected, which is useful if missing a positive case is costly (e.g., fraud detection, medical diagnosis).
 - **Decent precision** means the model isn't making too many incorrect positive predictions.



Overall Observations Test data– Bagging

-  **Areas for Improvement:**
- **Precision is slightly lower** than recall. If false positives are costly, consider adjusting decision thresholds or using a different ensemble technique.
- **Accuracy is 72.8%**, which is decent but could be improved using feature engineering or hyperparameter tuning.

bagging_classifier_model_train_perf

	Accuracy	Recall	Precision	F1
0	0.976845	0.981869	0.983437	0.982652

bagging_classifier_model_test_perf

	Accuracy	Recall	Precision	F1
0	0.728022	0.833105	0.776095	0.80359

Model Building- Random Forest

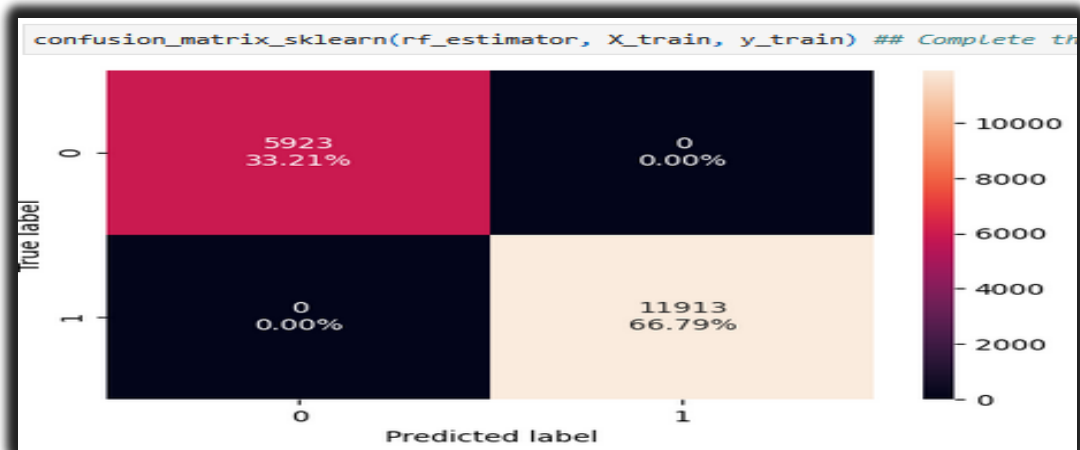
- Key Observations & Issues

- Severe Overfitting:

The perfect performance on training data (1.0 in all metrics) indicates the model has memorized patterns instead of learning general trends.

```
RandomForestClassifier  
RandomForestClassifier(random_state=1)
```

rf_estimator_model_train_perf				
	Accuracy	Recall	Precision	F1
0	1.0	1.0	1.0	1.0



Observations on Model Building- Random Forest

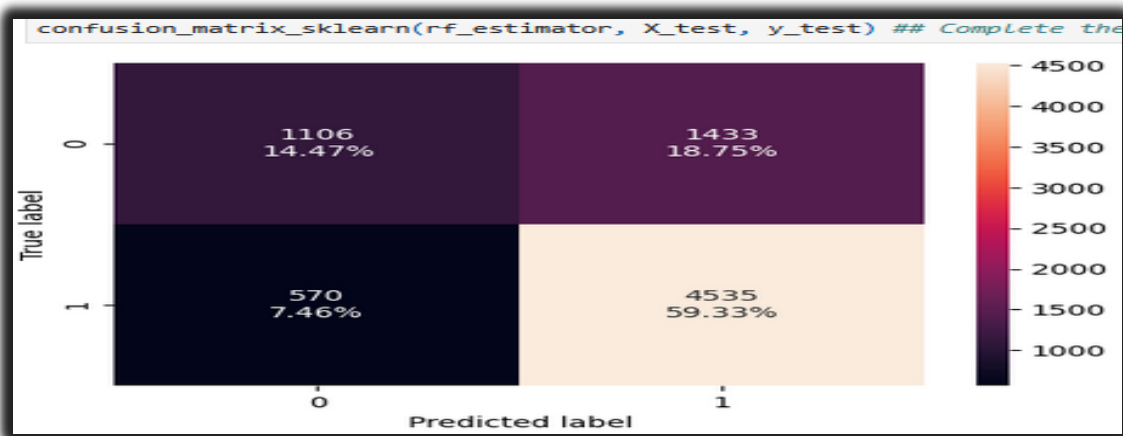
2. High Recall but Lower Precision in Testing:

The model identifies most actual visa approvals (high recall). However, a lower precision suggests a high false positive rate (incorrect approvals).

3. Model is Biased Towards Approval Cases:

The model prefers predicting visa approvals rather than denials, which can be risky if used in real scenarios

rf_estimator_model_test_perf				
	Accuracy	Recall	Precision	F1
0	0.737964	0.888345	0.759886	0.81911

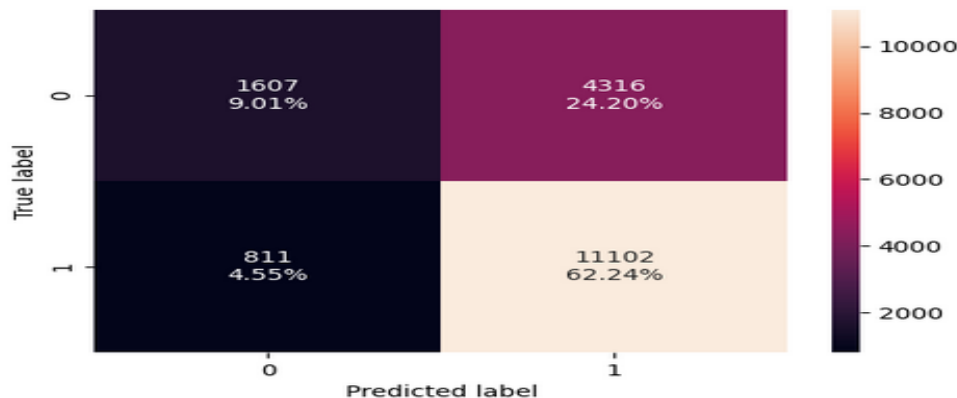


Model Improvement - Bagging

Hyperparameter tuning – Decision Tree

- The model correctly classifies a significant number of Certified cases.
- However, it has a high False Positive Rate (24.20%), meaning many Denied cases are mistakenly classified as Certified.
- A lower False Negative Rate (4.55%) suggests that the model is relatively better at capturing actual Certified cases.

```
DecisionTreeClassifier  
DecisionTreeClassifier(class_weight='balanced', max_depth=10, max_leaf_nodes=2,  
                        min_impurity_decrease=0.0001, min_samples_leaf=3,  
                        random_state=1)
```



Observations- Hyperparameter tuning – Decision Tree

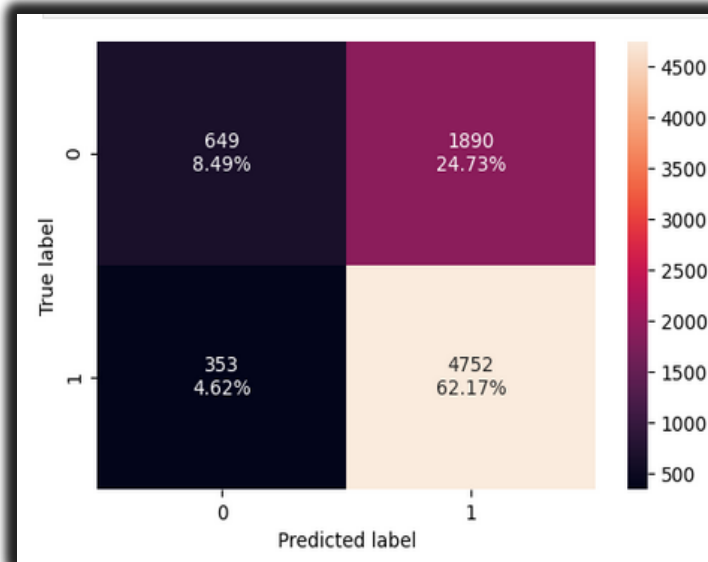
- The high misclassification rate for Denied cases could indicate a bias toward predicting approvals, potentially due to an imbalance in the dataset.
- The Training model performance

dtree_estimator_model_train_perf				
	Accuracy	Recall	Precision	F1
0	0.712548	0.931923	0.720067	0.812411

- The Testing Model performance

dtree_estimator_model_test_perf				
	Accuracy	Recall	Precision	F1
0	0.706567	0.930852	0.715447	0.809058

Confusion Matrix for Test data



Hyperparameter Tuning - Bagging Classifier

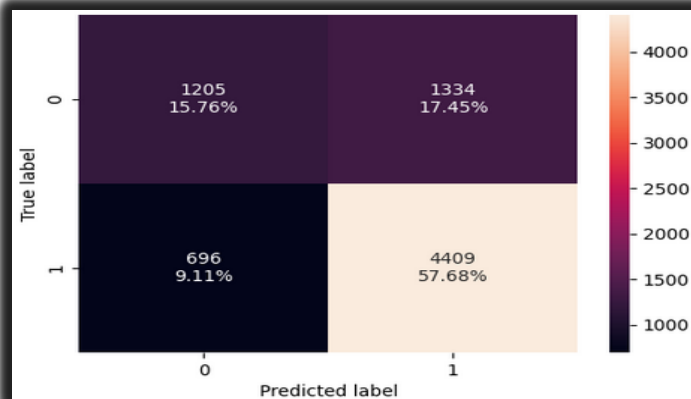
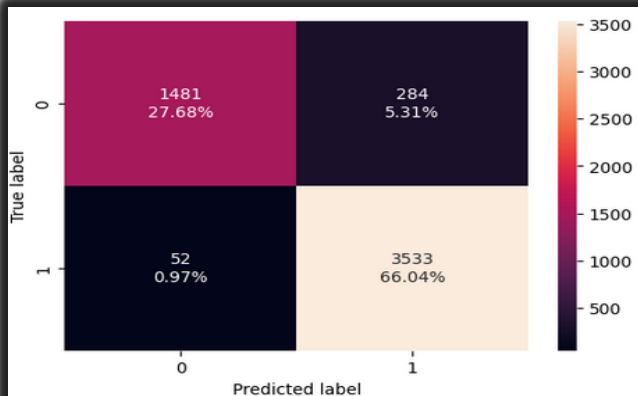
Below is the output after running the **Hyperparameter Tuning** show casting the model trying to perform well.

bagging_estimator_tuned_model_train_perf

	Accuracy	Recall	Precision	F1
0	0.937196	0.985495	0.925596	0.954607

bagging_estimator_tuned_model_test_perf

	Accuracy	Recall	Precision	F1
0	0.734432	0.863663	0.767717	0.812869



Observations - Hyperparameter Tuning - Bagging Classifier

- ➔ **Observation:** The model performs **exceptionally well on training data**, which suggests it has learned patterns very effectively. However, such high values might indicate **overfitting** to the training data.
- Observation:**
 - There is a **large gap between training and testing performance**, especially in **accuracy (93.71% → 73.44%)** and **precision (92.55% → 76.77%)**.
 - Overfitting is likely occurring**, meaning the model performs well on seen data but generalizes poorly to unseen data.

```
GridSearchCV
GridSearchCV(cv=3, estimator=BaggingClassifier(random_state=1), n_jobs=1,
             param_grid={'max_samples': [0.6, 0.7, 0.8],
                          'n_estimators': [25, 50, 100]},
             scoring=make_scorer(f1_score, response_method='predict'))
  best_estimator_: BaggingClassifier
BaggingClassifier(max_samples=0.6, n_estimators=100, random_state=1)
  BaggingClassifier
BaggingClassifier(max_samples=0.6, n_estimators=100, random_state=1)
```

Hyperparameter Tuning - Random Forest

- **High Recall but Poor Precision**
 - The model classifies almost **all cases as approvals** (high recall), leading to excessive false positives (low precision).
 - This is risky because it **incorrectly approves many visa applications that should be denied**.
- **Lower Overall Accuracy**
 - Compared to the previous model, accuracy has dropped significantly (from ~73% to ~67%).
 - This suggests the model is **too lenient on approvals and not distinguishing well between approved and denied cases**.

```
RandomForestClassifier  
  
RandomForestClassifier(max_depth=10, min_samples_split=3, n_estimators=20,  
                        random_state=1)
```

Observations - Hyperparameter Tuning - Random Forest

- Elimination of Overfitting**

- The previous overfitting issue (where training accuracy was 1.0) has been resolved.
- Now, training and testing accuracy are nearly identical (~67%), showing that the model generalizes better.

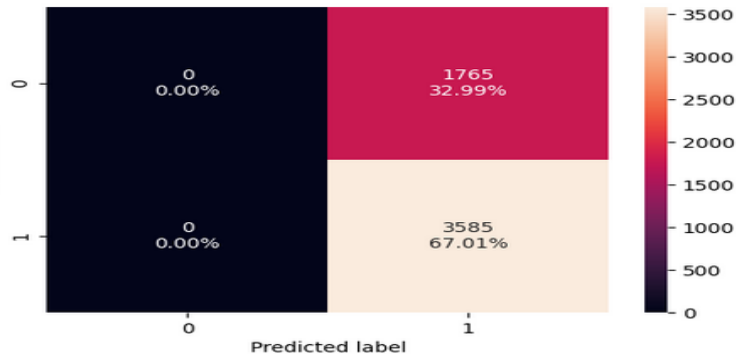
rf_tuned_model_train_perf

	Accuracy	Recall	Precision	F1
0	0.670093	1.0	0.670093	0.802462

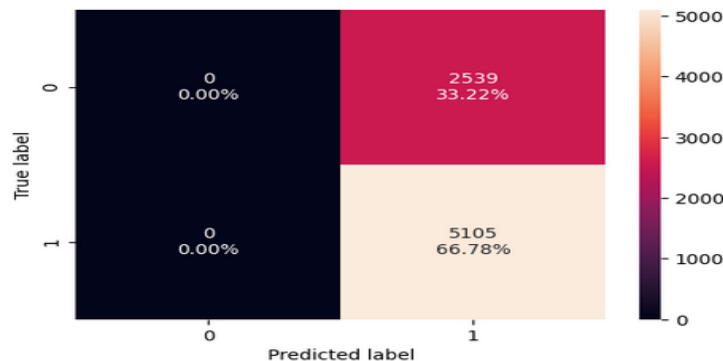
rf_tuned_model_test_perf

	Accuracy	Recall	Precision	F1
0	0.667844	1.0	0.667844	0.800847

confusion_matrix_sklearn(rf_tuned, X_small, y_small) ## Complete the code



confusion_matrix_sklearn(rf_tuned, X_test, y_test) ## Complete the code



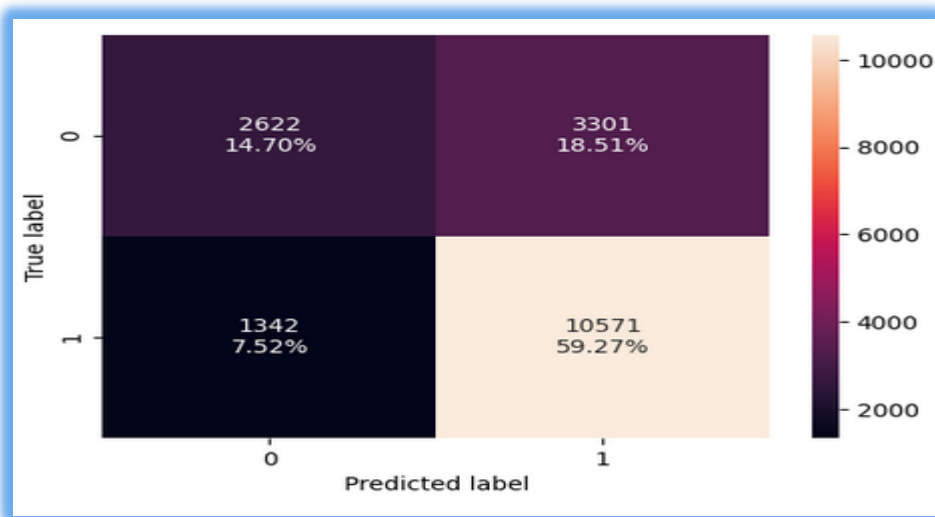
Model Building - Boosting

Model Building - Boosting – AdaBoost

- Key Observations:
- **Recall is strong**, both for training and testing, meaning the model does well at identifying relevant instances.

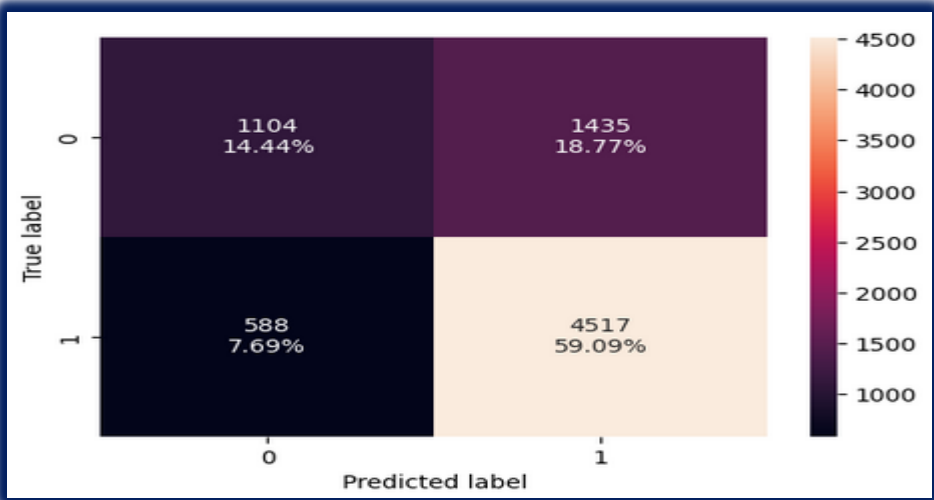
```
AdaBoostClassifier
AdaBoostClassifier(random_state=1)
```

ab_classifier_model_train_perf				
	Accuracy	Recall	Precision	F1
0	0.739684	0.88735	0.762039	0.819934



Observations - Boosting – AdaBoost - Model Building

- Precision has a slight drop from training to testing, suggesting more false positives in the testing set.
- The F1-score on testing data is lower than training, which may indicate slight overfitting.
- Overall, **the model is performing reasonably well**, but there may be a chance to fine-tune or explore techniques like regularization to improve generalization to unseen data.



ab_classifier_model_test_perf				
	Accuracy	Recall	Precision	F1
0	0.735348	0.884819	0.758905	0.817039

Boosting- Gradient Boost Classifier – Model Building

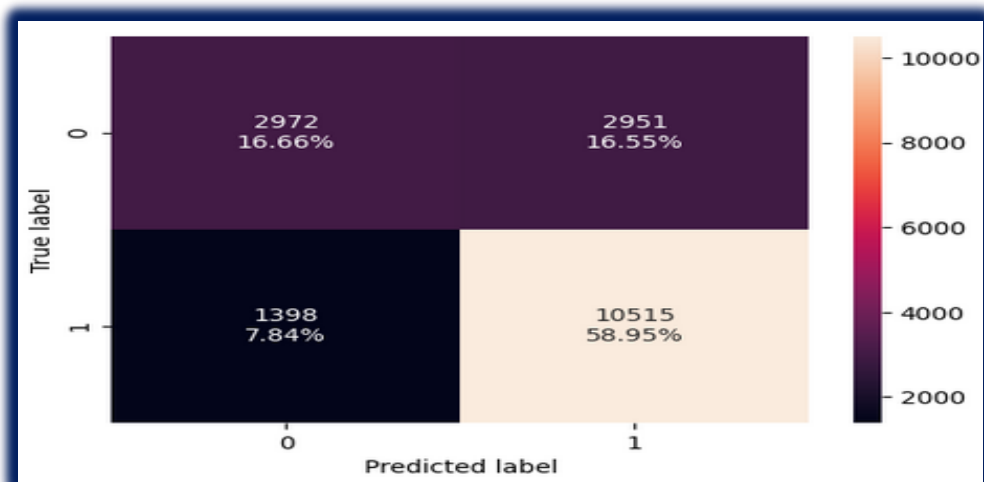
Key Observations:

- **Strong Recall:** The model performs excellently at identifying positive cases, both in training and testing.
- **Precision and Recall Balance:** There's a slight drop in precision when moving from training to testing, but it's still good. The F1-score reflects a strong balance between precision and recall.

```

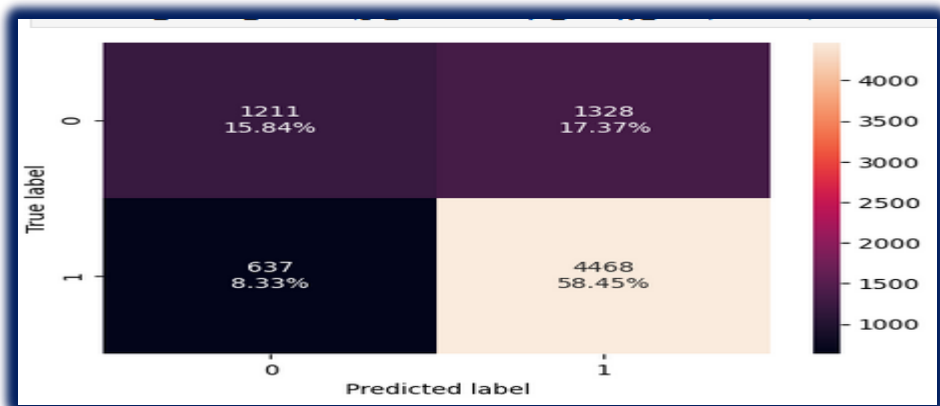
GradientBoostingClassifier
GradientBoostingClassifier(random_state=1)
    
```

gb_classifier_model_train_perf				
	Accuracy	Recall	Precision	F1
0	0.756167	0.882649	0.780855	0.828638



Observations - Boosting- Gradient Boost Classifier

- **Accuracy Drop:** The small drop in accuracy from training to testing indicates that the model has a good generalization ability, though there is still a little room for improvement.
- **Overall Performance:** The Gradient Boosting model is performing well, with no major issues with overfitting, and the drop in performance on the testing data is minimal.



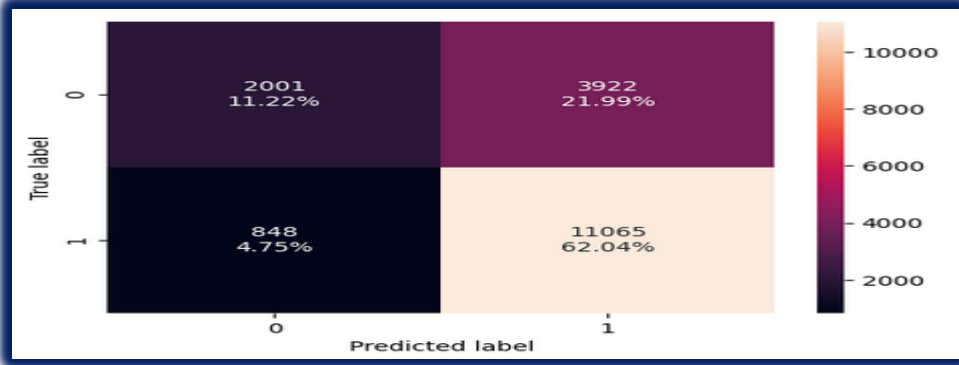
gb_classifier_model_test_perf

	Accuracy	Recall	Precision	F1
0	0.742936	0.87522	0.770876	0.819741

Model Improvement - Boosting

Boosting - Hyperparameter Tuning - AdaBoost Classifier

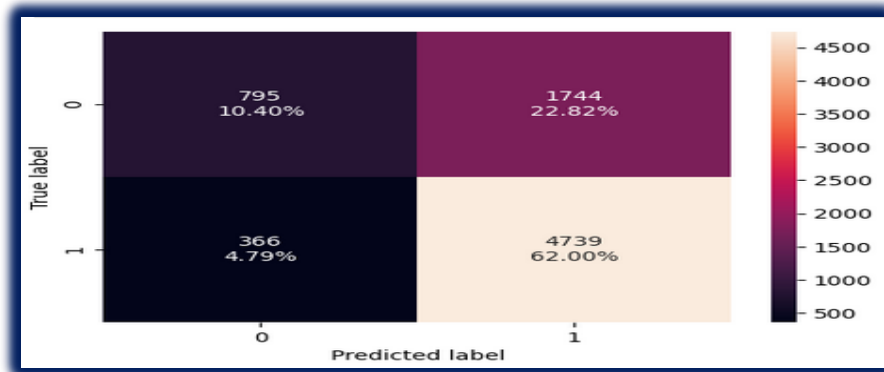
- **Key Observations:**
- **High Recall:** Both the training and testing data show very high recall, indicating that the model is excellent at identifying positive cases. This is great if you're more concerned about catching as many positive instances as possible (e.g., in fraud detection or medical diagnoses).



abc_tuned_model_train_perf				
	Accuracy	Recall	Precision	F1
0	0.732563	0.928817	0.738307	0.822677

Observations: Boosting - Hyperparameter Tuning - AdaBoost Classifier

- **Lower Precision:** There is a notable gap between recall and precision, especially on the testing data, suggesting that the model is generating a moderate number of false positives. Precision is slightly lower on the testing data than on the training data, which is typical as the model generalizes to unseen data.



abc_tuned_model_test_perf				
	Accuracy	Recall	Precision	F1
0	0.723967	0.928306	0.730989	0.817915

Observations: Boosting - Hyperparameter Tuning - AdaBoost Classifier

- **F1-Score:** The F1-scores are good on both training and testing data, indicating a solid balance between precision and recall. The small drop in the testing F1-score suggests that there's a slight loss of performance when generalizing to new data, but it's not significant.
- **F1-Score on Training data (0.822677):** The F1-score is quite good, suggesting that there's a reasonable balance between recall and precision. However, because precision is lower than recall, there's still some room to improve the precision without sacrificing too much recall.
- **Accuracy:** The drop in accuracy from training to testing is small, which is a good sign of generalization. The model is not overfitting significantly.



Happy Learning !

