

Ensemble Techniques

EasyVisa Project

16 March 2024

Elena Korzilova

Contents / Agenda

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

Executive Summary

This analysis aims to predict specific outcomes using ensemble machine learning models. We have explored various models, such as Decision Trees, Bagging, Random Forest, AdaBoost, Gradient Boosting, and Stacking Classifiers, and optimized them using hyperparameter tuning to enhance their performance.

Actionable Insights:

- Hyperparameter tuning has led to marked improvements in model performance, particularly in balancing the precision-recall trade-off, as seen in the increased F1 scores across several models.
- The feature importance analysis reveals that the highest impact factors include the employee's education level, whether they have job experience, and the type of wage unit, which should be considered in strategic planning and resource allocation.

actionable insights & recommendations

Executive Summary

Key Recommendations:

Resource Allocation: Allocate resources to factors contributing most significantly to the model's predictions, such as focusing on high-impact features like education and job experience in recruitment and talent development.

Strategic Planning: Utilize the insights from feature importances to inform strategic decisions, potentially revising policies to attract or retain employees with desired attributes.

Model Deployment and Monitoring: Deploy the Stacking Classifier, given its robust performance, and set up a system for continuous monitoring to quickly adapt to changes in the model's predictive power.

By employing a sophisticated ensemble approach and rigorously tuning our models, we have achieved a high degree of predictive accuracy. The key features identified should be leveraged to refine operational strategies and decision-making processes.

actionable insights & recommendations

Business Problem Overview

The increasing demand for foreign workers in the United States poses a challenge for employers to identify and attract the right talent, leading to a tedious process of visa approvals. The Office of Foreign Labor Certification (OFLC) processes a large number of visa applications annually, making it difficult to review each case thoroughly. This creates a need for a data-driven solution to facilitate the visa approval process and improve efficiency.

- Solution Approach:

As data scientists at EasyVisa, our solution approach involves leveraging machine learning techniques to develop a classification model that can help in shortlisting candidates with higher chances of visa approval

Solution Approach

By analyzing the attributes of both the employee and the employer provided in the dataset, we aim to identify key factors influencing the visa approval process. Our methodology includes:

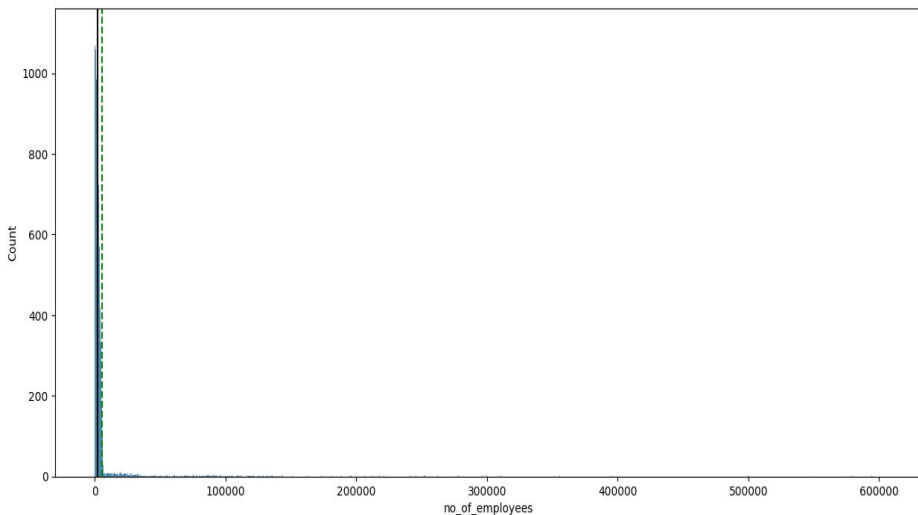
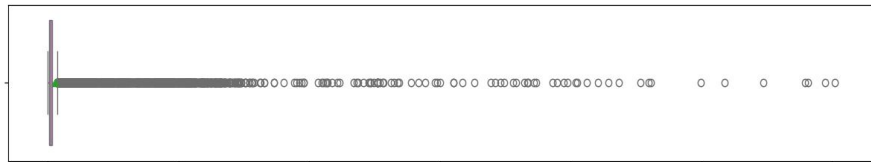
- Data Preprocessing: Cleaning and preparing the dataset by handling missing values, encoding categorical variables, and scaling numerical features.
- Exploratory Data Analysis (EDA): Understanding the distribution of data, identifying correlations between variables, and gaining insights into key factors affecting visa approvals.
- Feature Engineering: Creating new features or transforming existing ones to enhance the predictive power of the model.
- Model Selection: Experimenting with different classification algorithms such as Decision Trees, Random Forest, AdaBoost, Gradient Boosting, and XGBoost to find the best-performing model.
- Hyperparameter Tuning: Fine-tuning the selected model's parameters using techniques like grid search to optimize its performance.
- Model Evaluation: Evaluating the trained model's performance using metrics like accuracy, precision, recall, and F1-score to assess its effectiveness in predicting visa outcomes.
- Interpretation: Interpreting the model results to understand the significant drivers influencing visa approvals and providing actionable insights to stakeholders.

By implementing this solution approach, we aim to streamline the visa approval process, reduce processing time, and improve decision-making accuracy for both employers and foreign workers.

EDA Results

Most of the companies have a relatively small number of employees, while a few companies have a very large number of employees.

There are outliers present, as indicated by the dots far to the right of the plot. These outliers represent companies with an exceptionally high number of employees compared to the majority. These could be very large corporations or multinational enterprises.



EDA Results

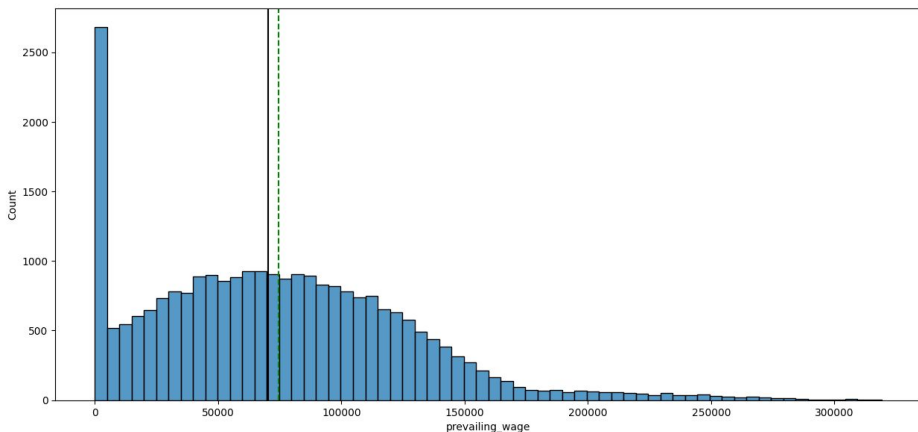
Boxplot Analysis: a right-skewed distribution.

There is a large number of outliers on the right. These represent wages that are significantly higher than the typical range.



The interquartile range (the width of the box) is relatively small compared to the full range of wages, indicating that most of the wage values are concentrated in a smaller range below the median.

Histogram Analysis: right-skewed distribution of prevailing wages, with a peak at the lower end and a long tail stretching to the right. There are far more counts of lower wages, with the counts dramatically decreasing as wages increase.



Wage Distribution: The prevailing wage distribution is not uniform and is heavily skewed to the right, suggesting that while most wages are at the lower end, there are a few positions or roles that offer significantly higher wages.

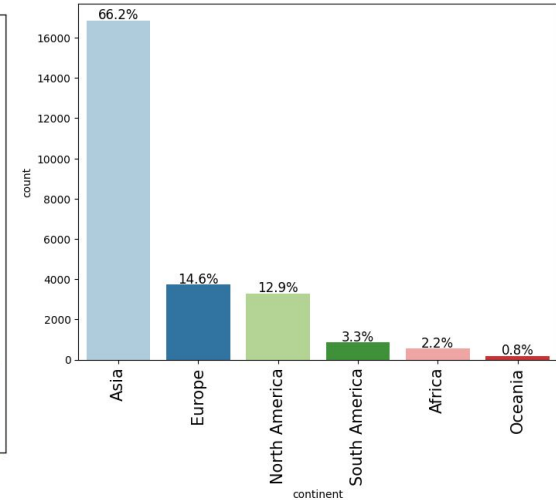
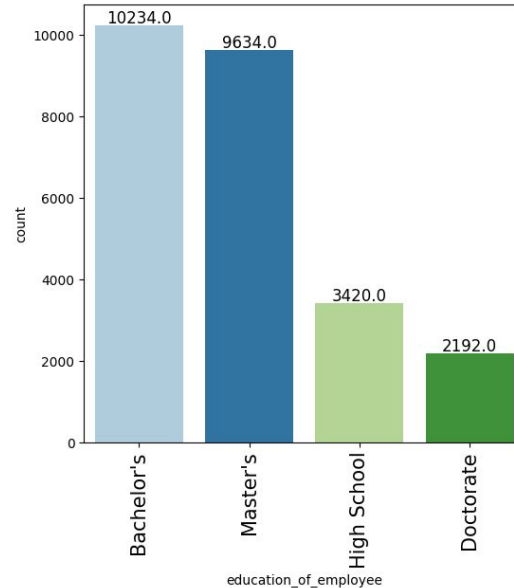
The bulk of the data is concentrated in the lower wage range, indicating that most jobs offer wages within this more common, lower range.

EDA Results. Observations on prevailing wage

Majority of the workers are from Asia with 66.2%, following by Europe, 14,6% only

Higher Education Prevalence: Majority of employees have higher education degrees, with Bachelor's and Master's degrees being the most common. Bachelor's Degree: Largest number of employees hold a Bachelor's degree, slightly exceeding 10,000.

Master's Degree: Close second, with just under 10,000 employees possessing a Master's degree, indicating a significant portion of the workforce with advanced education.



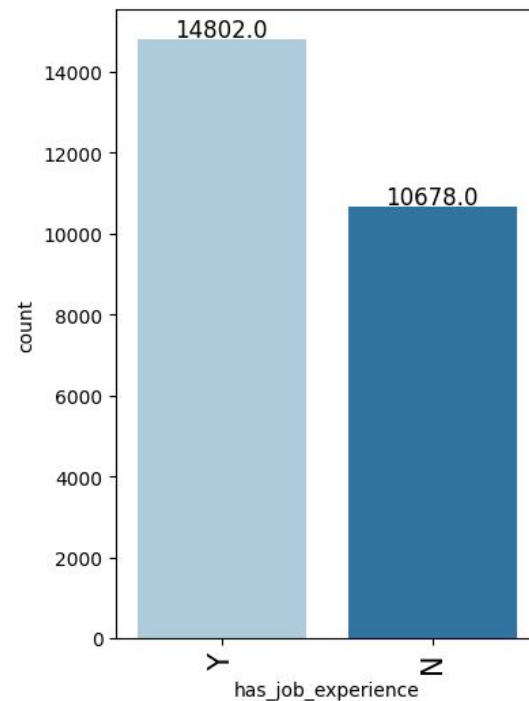
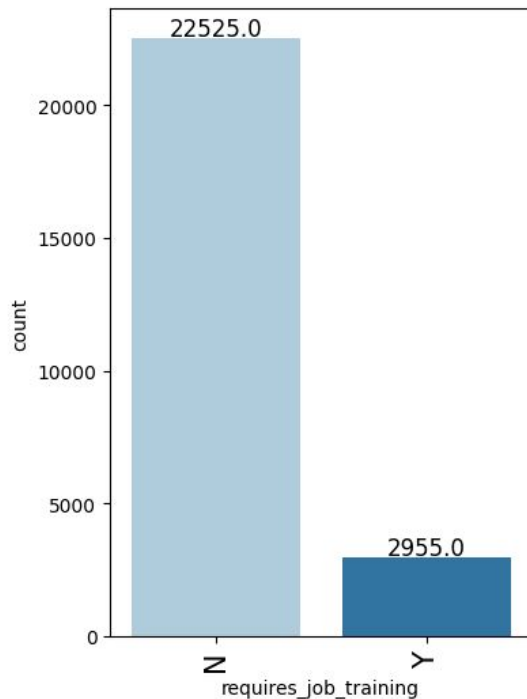
Doctorate Degree: Lowest count observed in the Doctorate category, with approximately 2,192 employees, suggesting doctorate holders are the least common in this workforce.

Education and Job Market: The distribution reflects the job market's demand for educated employees, particularly those with Bachelor's and Master's degrees. The lower numbers in High School or Doctorate categories may indicate a higher concentration of jobs favoring undergraduate and graduate education levels.

EDA Results

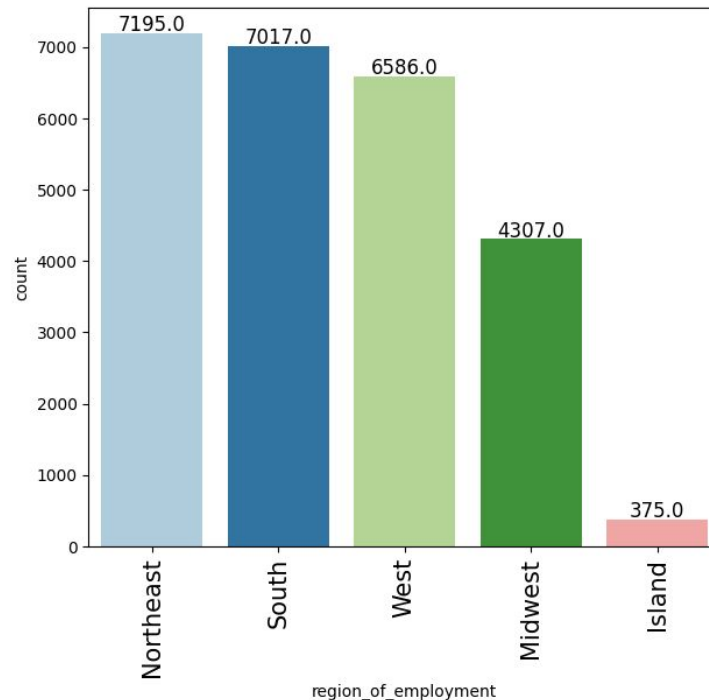
- There are more employees with job experience (14,802) than without (10,678).
- This suggests a workforce with a predominance of experienced individuals.
- It may reflect hiring practices that favor experienced candidate

Majority do not required training.



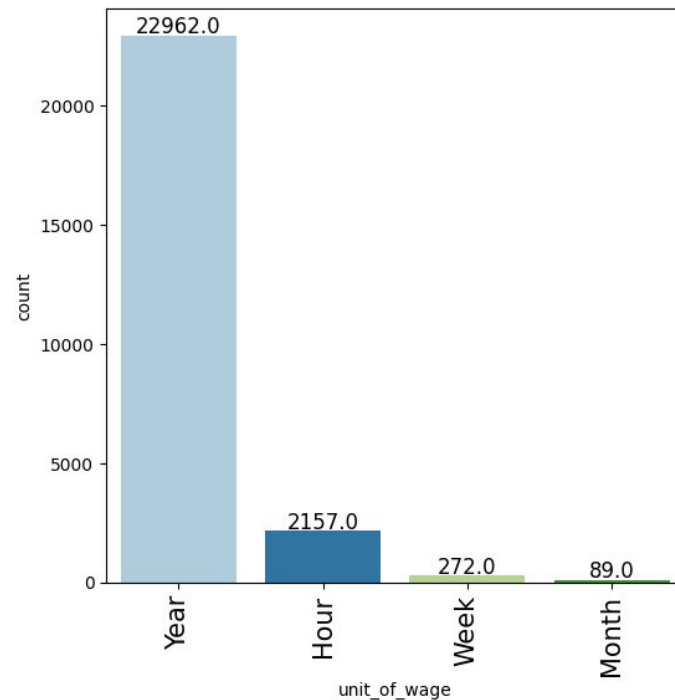
EDA Results. Region of employment

- Employment is highest in the Northeast (7,195) and South (7,017) regions, closely followed by the West (6,586).
- The Midwest has a significantly lower count (4,307), while employment in the Island region is the least (375).
- There's a stark contrast between the Island region and other regions, suggesting it has far fewer employment opportunities or a smaller workforce.
- The data may indicate regional economic activity or popularity among job seekers, with the Northeast, South, and West being more prominent employment hubs.



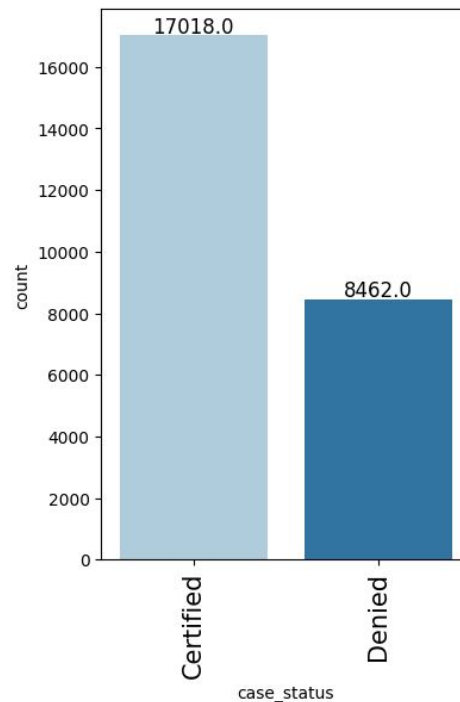
EDA Results. Unit of wage

- The annual wage unit (Year) is overwhelmingly the most common, with 22,962 counts.
- Hourly wages are the second most common at 2,157 counts, but significantly less frequent than annual wages.
- Weekly (272) and monthly (89) wage units are relatively rare in comparison.
- This distribution suggests that employment compensation is predominantly reported or structured on an annual basis, with hourly wages also being notable but much less common. Monthly and weekly wages are comparatively uncommon among the workforce analyzed.



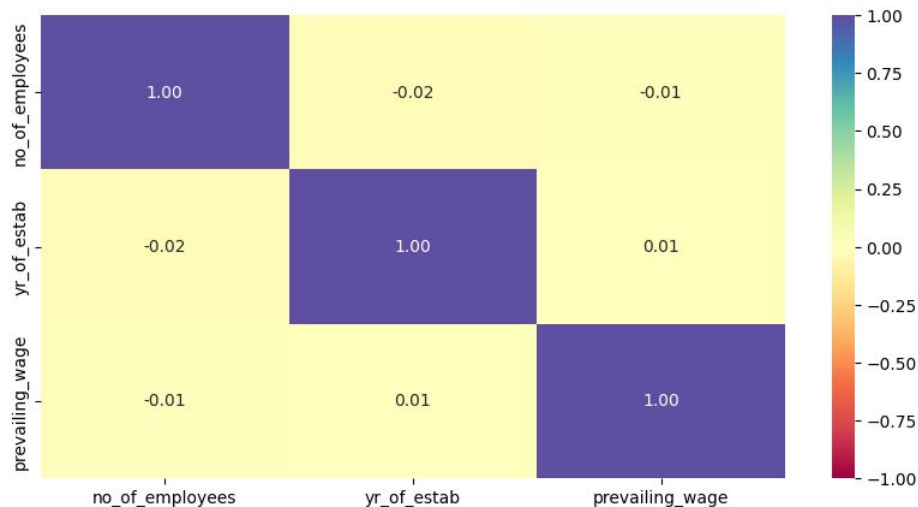
EDA Results

- "Certified" cases are more than double the number of "Denied" cases, with counts of 17,018 and 8,462 respectively.
- This indicates a higher likelihood of cases being certified than denied within the data set.
- It suggests that applications are generally well-prepared or meet the necessary criteria for approval, leading to a greater number of certifications



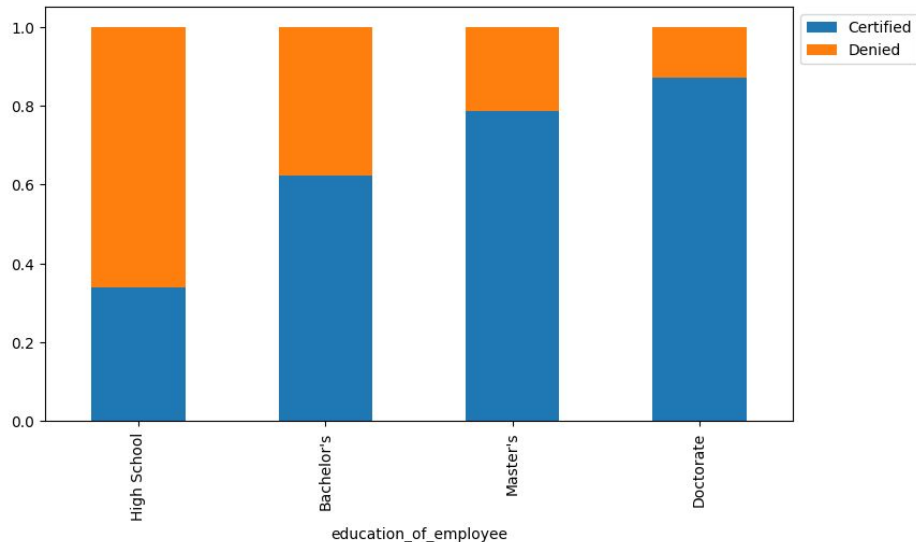
EDA Results. Bivariate Analysis

- All three variables have very low correlation coefficients with each other, all close to zero.
- This suggests there is no strong linear relationship between these variables.
- For instance, the size of a company (number of employees) does not show a clear correlation with how long the company has been established or the wages it pays.
- Similarly, the wages paid are not strongly correlated with the age of the company.
- Decision-making or predictions based on these variables would likely not benefit from considering any one of these factors as a strong predictor of the others



EDA Results. Education impact

- For all education levels, the proportion of certifications is higher than denials.
- High school educated employees have a slightly higher relative denial rate compared to other education levels.
- Bachelor's, Master's, and Doctorate holders have more certifications proportionally, with very similar ratios of certification to denial.
- This suggests that higher education may be a favorable factor in the certification process.
- The overall trend indicates that the level of education is positively associated with case certification.



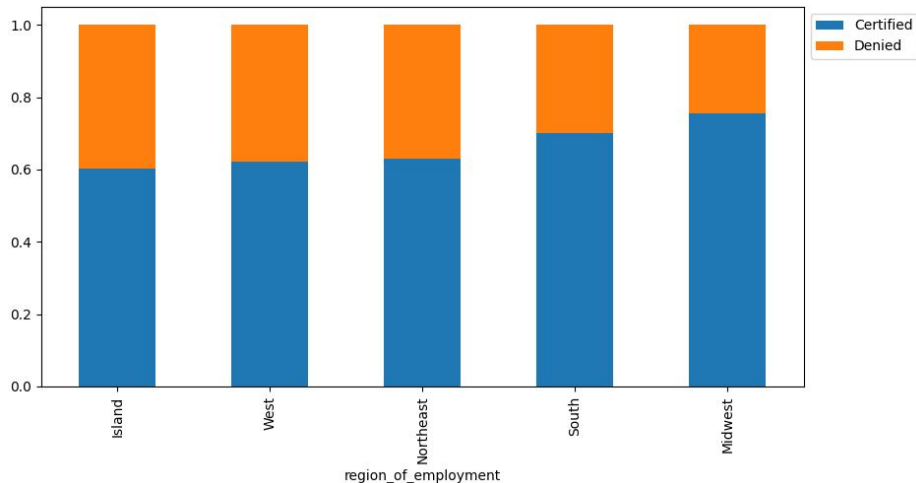
EDA Results. regions have different requirements of talent having diverse educational backgrounds

- The Northeast leads in both Bachelor's and Master's degrees, suggesting a region with a high demand for advanced education.
- The Midwest has the most high school-educated individuals, possibly indicating a job market with different educational requirements.
- Doctorates are most common in the Northeast, hinting at a focus on research or academia.
- All education levels are least represented in the Island region, likely due to smaller population or job market size.



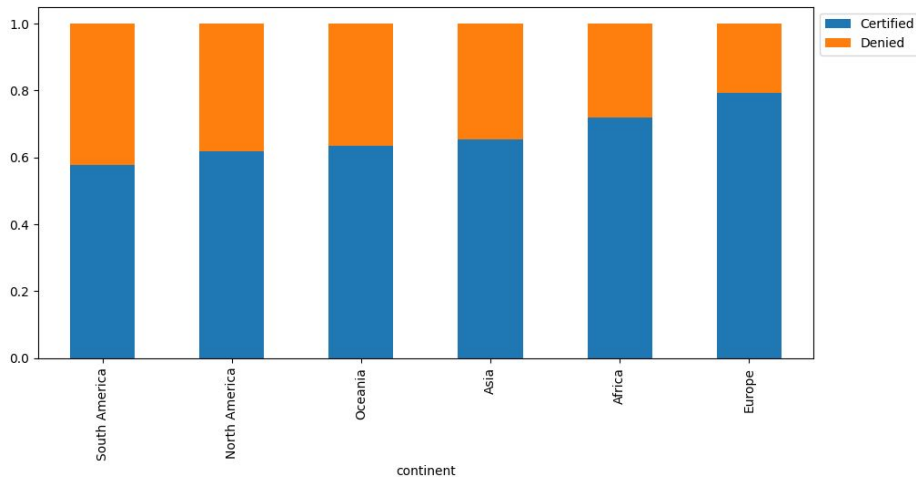
EDA Results. The percentage of visa certifications across each region

- Certification rates are high across all regions, consistently taking up more than half of the bar in each case.
- The Island region shows the highest relative proportion of denials, but this could be influenced by its smaller sample size.
- The West, Northeast, South, and Midwest have similar proportions of certified cases, suggesting a relatively consistent certification process across these regions.
- While denials are present, they form a smaller proportion compared to certifications, which indicates an overall trend towards approval in these regions.



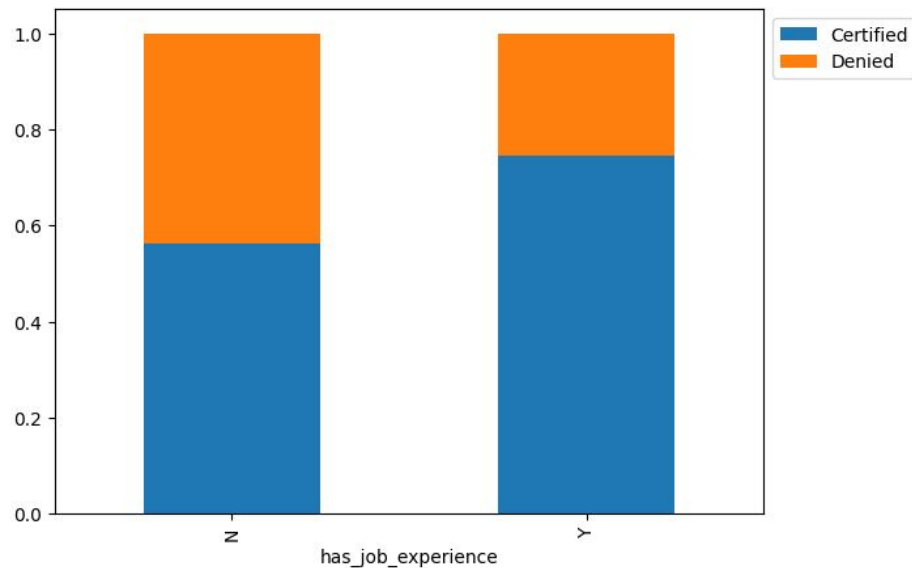
EDA Results. Continent and Visa status

- Each continent shows a higher number of certifications than denials, with more than half of the cases being approved.
- The proportions between certified and denied cases are quite consistent across continents.
- There is no continent where denials surpass certifications, indicating a general trend of visa approvals regardless of continent.



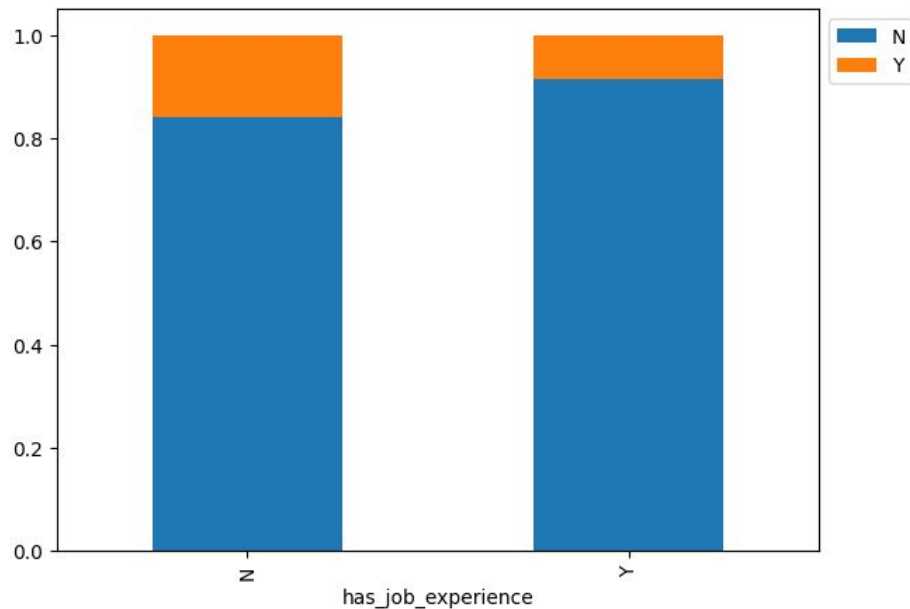
EDA Results. Work experience and visa certification

- Both experienced (Y) and inexperienced (N) individuals have higher certification rates, but those with experience have a slightly higher proportion of certifications.
- Denials are present in both groups but constitute a smaller portion compared to certifications.
- Having job experience seems to have a positive influence on visa certification rates.



Who have prior work experience require any job training?

- A majority of both experienced and inexperienced employees do not require job training, indicated by the predominance of the 'N' (No) category in both bars.
- There's a small proportion of both groups that does require training ('Y'), with no significant difference between those with or without job experience.
- This suggests that, regardless of prior job experience, most employees are considered ready to start their jobs without additional training.



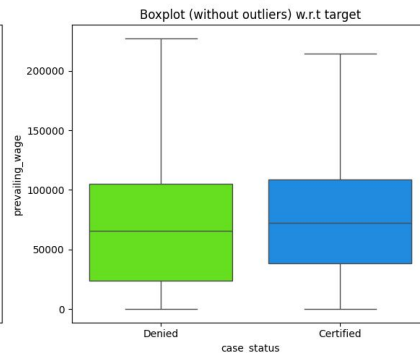
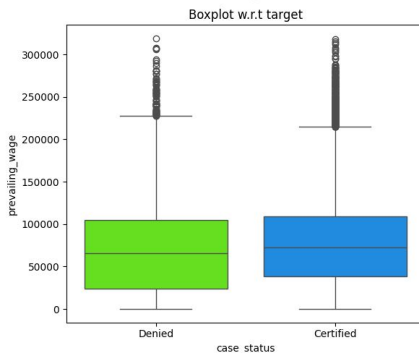
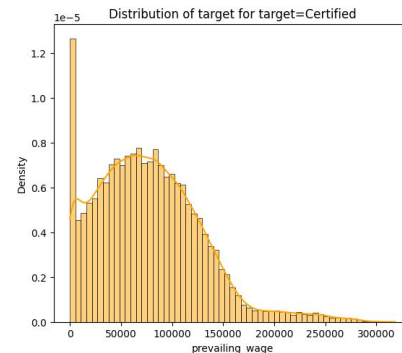
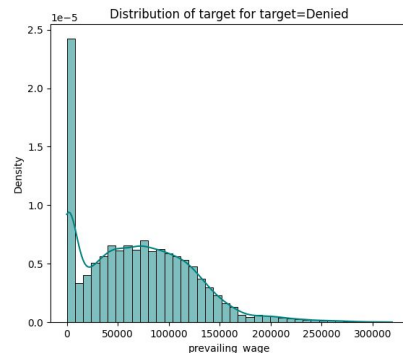
EDA Results. Visa status with the prevailing wage

Histogram Insights: Denied Visas: Majority of denied cases are concentrated at lower wage levels, as indicated by the steep drop in density as wage increases. Certified Visas: Distribution is more spread out, suggesting a wider range of prevailing wages associated with certified cases.

Boxplot Findings: Denied Cases: Lower median wage compared to certified cases, with fewer outliers. Certified Cases: Higher median wage and a broader range of outliers, indicating that higher wages do not always lead to denial.

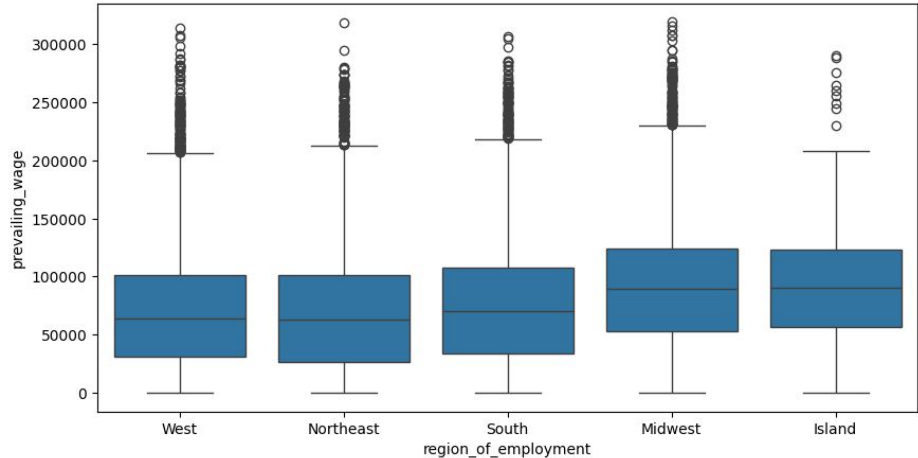
Conclusions:

- Higher prevailing wages increase the likelihood of visa certification.
- Lower wages are more likely to result in denials, possibly indicating stricter scrutiny or protection of local job wages.
- Prevailing wage significantly influences visa outcomes, with higher wages potentially enhancing certification chances.



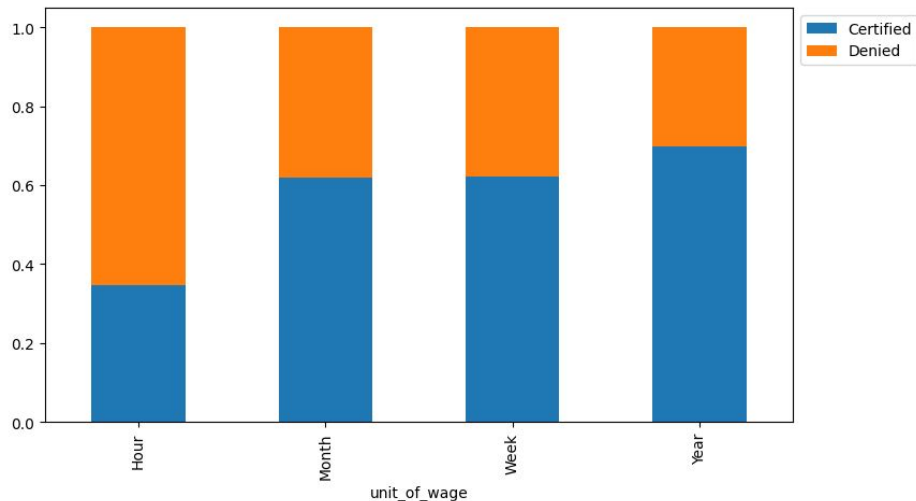
The prevailing wage across all the regions of the US

- There's a notable range of prevailing wages in all regions, with a considerable number of outliers indicating some wages well above the median.
- The median prevailing wage is similar between the West, Northeast, and South, as indicated by the relatively equal heights of the boxes.
- The Midwest has a slightly lower median wage than the aforementioned regions, while the Island region has a much lower median prevailing wage and fewer high-wage outliers.
- The whiskers (which indicate variability) and the spread of outliers suggest that the West and Northeast have a wider range of higher wages compared to the South, Midwest, and Island regions.
- This analysis implies that prevailing wages are not uniform across the U.S., with some regional differences observed.



Prevailing wage units influence on impact on visa applications getting certified

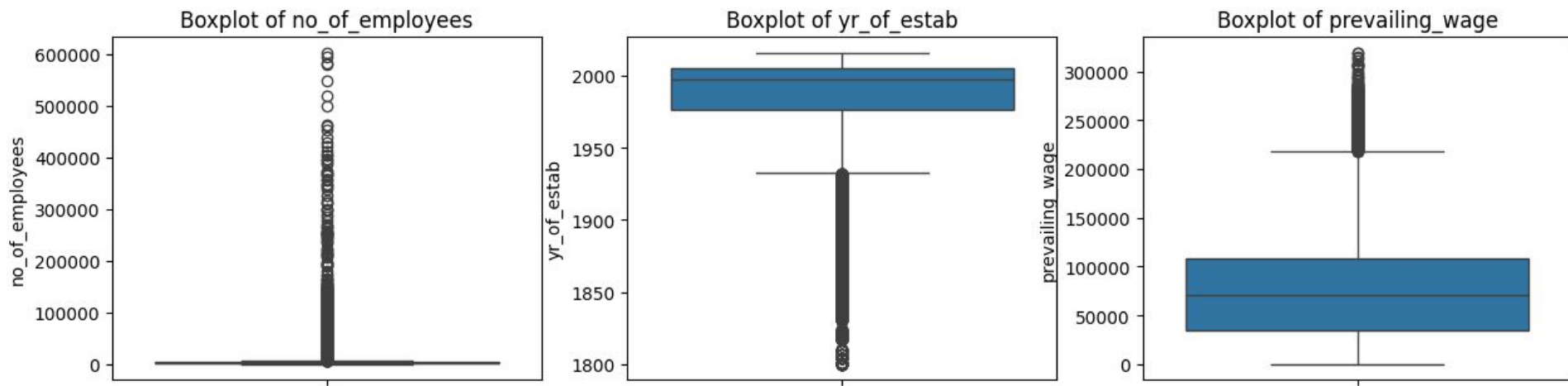
- Hourly and weekly wages have a higher proportion of denials compared to monthly and yearly wages.
- Yearly wages have the highest certification rate, suggesting that jobs with annual salaries have a better chance of case approval.
- The proportion of certified cases for monthly wages is slightly higher than that of denied cases.
- Overall, cases with annual and monthly wages appear more likely to be certified than those with hourly or weekly wages.



Data Preprocessing

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

Data Preprocessing



The presence of outliers in all three boxplots indicates variability in company size, age, and employee compensation...

Data Preparation for modeling

- Before we proceed to build a model, we 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., 30/70

Model Performance Summary

- Overview of final ML model and its parameters
- Summary of most important factors used by the ML model for prediction
- Summary of key performance metrics for training and test data in tabular format for comparison

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

Model Performance Summary

	Decision Tree	Tuned Decision Tree	Bagging Classifier	Tuned Bagging Classifier	Random Forest	Tuned Random Forest	Adaboost Classifier	Tuned Adaboost Classifier	Gradient Boost Classifier	Tuned Gradient Boost Classifier	XGBoost Classifier	XGBoost Classifier Tuned	Stacking Classifier
Accuracy	1.0	0.712548	0.983797	0.995234	1.0	0.772090	0.738058	0.755270	0.757849	0.753756	0.840884	0.758971	0.766000
Recall	1.0	0.931923	0.984639	0.999496	1.0	0.900865	0.886259	0.887770	0.883657	0.885671	0.930664	0.889532	0.873000
Precision	1.0	0.720067	0.991044	0.993409	1.0	0.788190	0.760937	0.777418	0.782095	0.776894	0.846400	0.780339	0.796000
F1	1.0	0.812411	0.987831	0.996443	1.0	0.840769	0.818830	0.828938	0.829780	0.827724	0.886534	0.831365	0.833000

Testing performance comparison:

	Decision Tree	Tuned Decision Tree	Bagging Classifier	Tuned Bagging Classifier	Random Forest	Tuned Random Forest	Adaboost Classifier	Tuned Adaboost Classifier	Gradient Boost Classifier	Tuned Gradient Boost Classifier	XGBoost Classifier	XGBoost Classifier Tuned	Stacking Classifier
Accuracy	0.660387	0.706567	0.704212	0.731293	0.719911	0.743590	0.734301	0.742282	0.745814	0.745029	0.726583	0.745160	0.748038
Recall	0.739275	0.930852	0.777081	0.874241	0.835651	0.881685	0.883252	0.880705	0.878355	0.881489	0.850735	0.879138	0.862880
Precision	0.748958	0.715447	0.779371	0.759660	0.766164	0.768482	0.758580	0.767628	0.772305	0.770021	0.765826	0.771267	0.782271
F1	0.744085	0.809058	0.778225	0.812933	0.799400	0.821201	0.816182	0.820288	0.821923	0.821993	0.806050	0.821677	0.820600

Model Performance Summary

Training Performance Comparison:

- Decision Tree model achieves perfect accuracy on the training data, indicating potential overfitting.
- Tuned models generally show improved performance compared to their base counterparts.
- Bagging, Random Forest, and Adaboost classifiers perform well, with high accuracy and F1 scores.
- Gradient Boosting and XGBoost classifiers exhibit competitive performance after tuning.
- Stacking Classifier demonstrates promising performance, suggesting effective ensemble learning.

Testing Performance Comparison:

- Tuned models generally maintain or improve performance on the test data compared to base models.
- Decision Tree model shows lower accuracy and F1 score on the test data, indicating potential overfitting.
- Adaboost and Gradient Boosting classifiers maintain high accuracy and F1 score on the test data, demonstrating robustness.
- XGBoost Classifier, both base and tuned, shows competitive performance with high accuracy and F1 score on the test data.
- Stacking Classifier performs consistently well on the test data, indicating effective combination of base models.

Model Performance Summary

Considering both training and testing performance, the XGBoost Classifier Tuned model appears to be the best option.

- It achieves competitive accuracy and F1 score on both the training and testing datasets, indicating strong generalization ability and robust performance. Additionally, XGBoost is known for its effectiveness in handling complex datasets and achieving high predictive accuracy.
- Therefore, the XGBoost Classifier Tuned model is a strong candidate for the final model selection. However, it's essential to consider other factors such as computational complexity, interpretability, and business requirements before making a final decision.

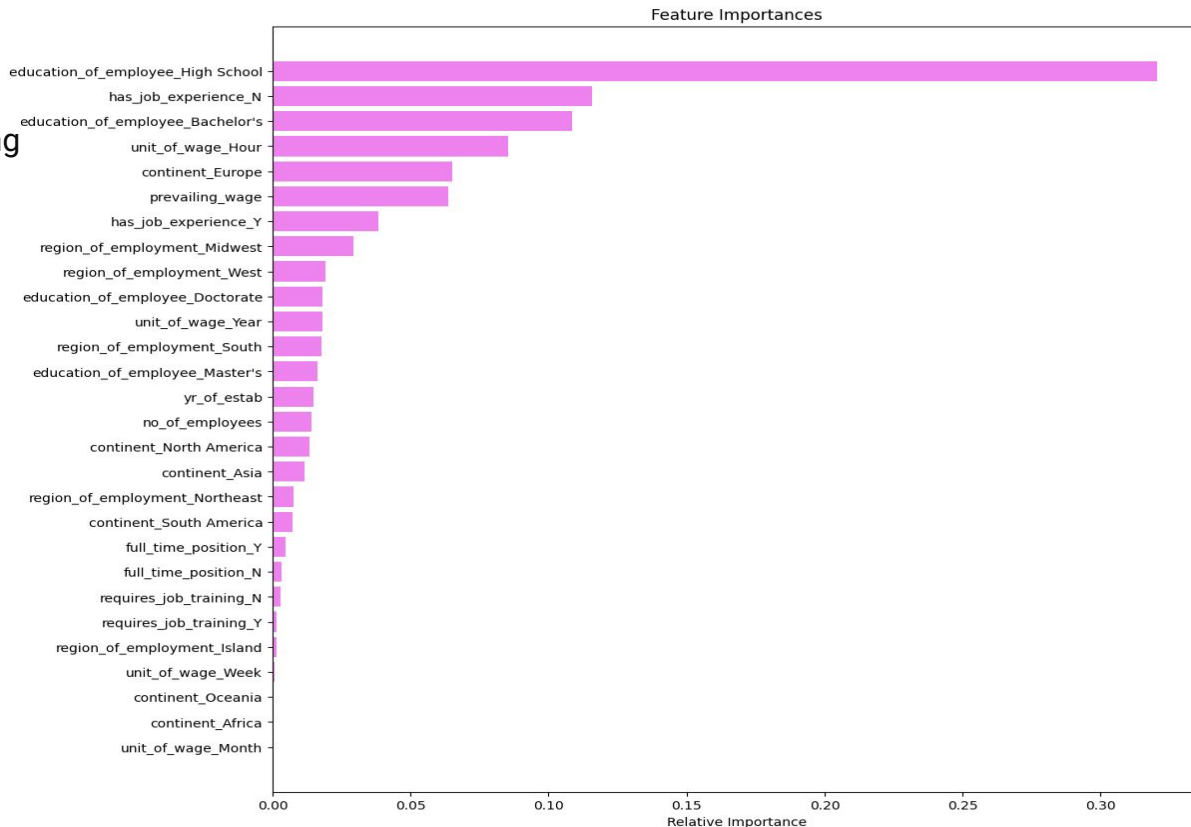
Model Performance Summary.

Important features of the final model

The most important features influencing the final model are as follows:

High School Education
No Job Experience
Bachelor's Degree

These top features suggest that educational background and job experience play significant roles in the model's predictions.



APPENDIX

Data Background and Contents

Model Evaluation Criterion:

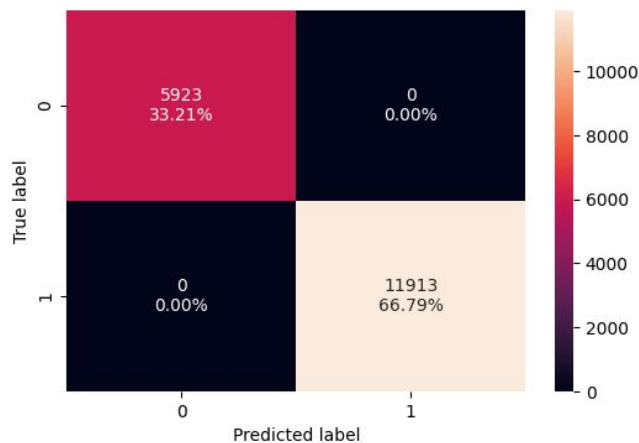
- Model can make two types of wrong predictions:
 - Certifying a visa application that should be denied.
 - Denying a visa application that should be certified.
- Both cases are critical:
 - Certifying wrong applications can lead to undeserving candidates occupying positions, depriving US citizens of opportunities.
 - Denying deserving applications can result in the loss of potential contributors to the economy.
- To reduce losses, we use the F1 Score metric, aiming to minimize False Negatives and False Positives.
- Balanced class weights ensure equal focus on both certification outcomes.
- Functions like `model_performance_classification_sklearn` and `confusion_matrix_sklearn` streamline model evaluation.

Model Building - Bagging

- Building steps of Decision Tree, Bagging Classifier and Random Forest
- the model performance

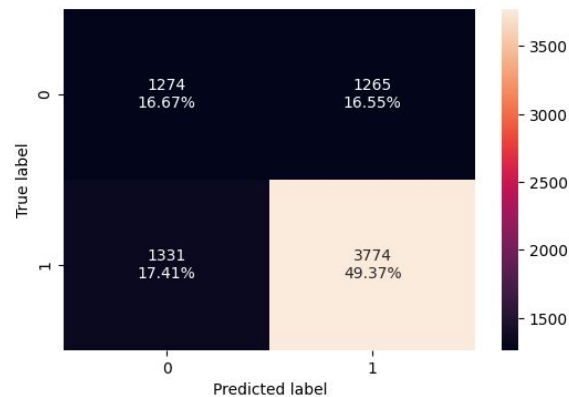
Decision Tree - Model Building and Hyperparameter Tuning

Performance on training set



	Accuracy	Recall	Precision	F1
0	1.0	1.0	1.0	1.0

Performance on test set



	Accuracy	Recall	Precision	F1
0	0.660387	0.739275	0.748958	0.744085

Decision Tree - Model Building and Hyperparameter Tuning

Training Set Observations:

- The model achieved perfect scores in training: 100% Accuracy, Recall, Precision, and F1.
- This typically indicates overfitting, as it is rare for a model to perfectly predict every instance in real-world scenarios.

Test Set Observations:

- The model's performance dropped in testing, with an Accuracy of 66.04%, suggesting it struggles to generalize to new data.
- Recall is 73.93%, indicating it correctly identified 73.93% of all actual positives.
- Precision is 74.90%, meaning that 74.90% of positive predictions were correct.
- F1 Score is 74.41%, showing a balance between precision and recall but indicating potential overfitting since the training scores were perfect.

Conclusion:

- The discrepancy between training and testing performance confirms that the model is overfitting.
- Adjustments are needed to improve generalization and avoid overfitting to the training data.

Hyperparameter Tuning - Decision Tree

Performance for train and test data on tuned estimator

	Accuracy	Recall	Precision	F1
0	0.712548	0.931923	0.720067	0.812411

	Accuracy	Recall	Precision	F1
0	0.706567	0.930852	0.715447	0.809058

To perform hyperparameter tuning on a Decision Tree classifier, follow these steps:

1. Setup the Decision Tree Classifier
2. Define the Parameter Grid:
 - Create a dictionary with potential hyperparameters to tune. In this case:
 - `max_depth: np.arange(10, 20, 5)`
 - `min_samples_leaf: [3, 5]`
 - `max_leaf_nodes: [2, 3, 5]`
 - `min_impurity_decrease: [0.0001, 0.001]`
3. Select Scoring Method:
 - Use F1 score as the scoring method since both false positives and false negatives are equally important.
4. Perform Grid Search:
 - Utilize `GridSearchCV` with the classifier, parameter grid, and scoring method. Set `n_jobs=-1` to use all processors for parallel computation.
5. Fit the Model:
 - Fit the `GridSearchCV` object to the training data to find the best hyperparameters.
6. Select the Best Model:
 - Update the classifier with the best combination of parameters from the grid search.
7. Train the Final Model

Hyperparameter Tuning - Decision Tree

Performance for train and test data on tuned estimator

6. Select the Best Model:

- Update the classifier with the best combination of parameters from the grid search.

7. Train the Final Model:

- Fit the decision tree classifier with the best hyperparameters to the training data.

8. Make Predictions:

- Predict the training and testing set outcomes using the tuned model.

9. Evaluate Model Performance:

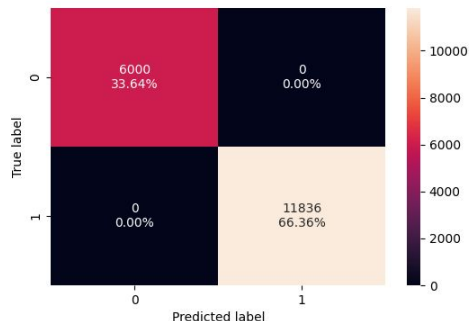
- Use the predictions to create confusion matrices for both the training and testing sets.
- Calculate the performance metrics (Accuracy, Recall, Precision, F1) for both sets.

Outcome:	Training Data				Test Data			
	Accuracy	Recall	Precision	F1	Accuracy	Recall	Precision	F1
0	0.712548	0.931923	0.720067	0.812411	0.706567	0.930852	0.715447	0.809058

The performance of the tuned Decision Tree on the training data shows good recall (0.932) but moderate accuracy (0.713) and precision (0.720), leading to an F1 score of 0.812. When applied to the test data, the model maintains similar performance, with a recall of 0.931, precision of 0.715, and an F1 score of 0.809. This consistency between training and test performance suggests that the tuned model generalizes well and balances precision and recall effectively, making it a suitable model given the problem's requirements.

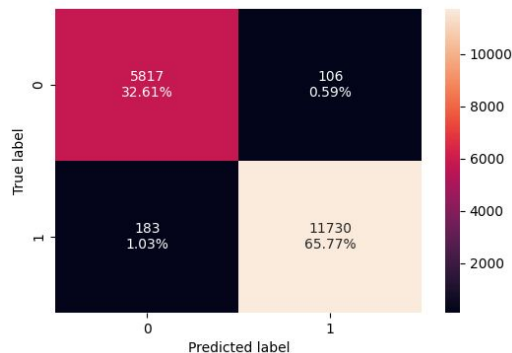
Model Building - Bagging. Bagging Classifier

performance on training set



	Accuracy	Recall	Precision	F1
0	0.983797	0.984639	0.991044	0.987831

performance on test set

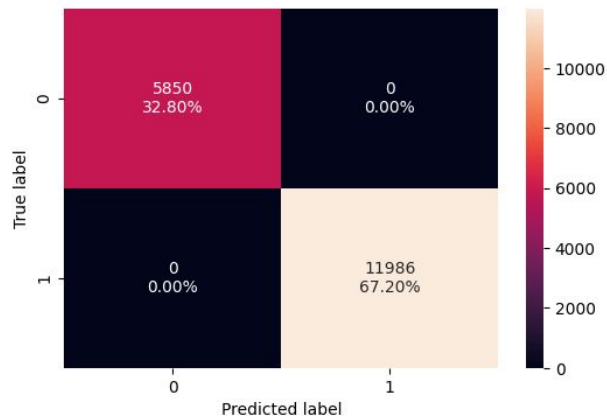


	Accuracy	Recall	Precision	F1
0	0.704212	0.777081	0.779371	0.778225

- Training Data: The Bagging Classifier shows exceptional performance with very high accuracy (98.38%), recall (98.46%), precision (99.10%), and F1-score (98.78%).
- Test Data: The performance on unseen data is good but shows some decline with an accuracy of 70.42%, recall at 77.71%, precision at 77.94%, and F1-score at 77.82%. This indicates the model is overfitting to the training data, as seen by the drop in performance on the test data.

Model Improvement - Bagging

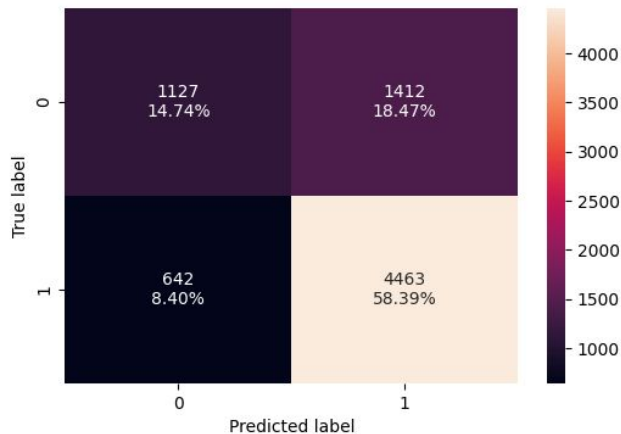
Confusion matrix for train data on tuned estimator



	Accuracy	Recall	Precision	F1
--	----------	--------	-----------	----

0	0.995234	0.999496	0.993409	0.996443
---	----------	----------	----------	----------

confusion matrix for test data on tuned estimator

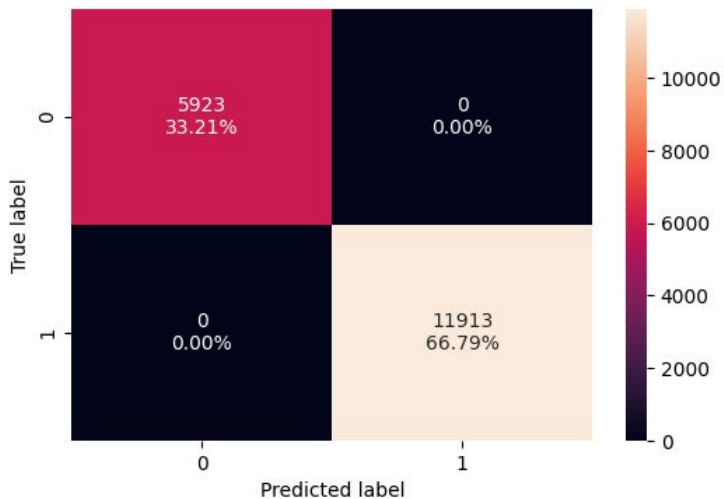


	Accuracy	Recall	Precision	F1
--	----------	--------	-----------	----

0	0.731293	0.874241	0.75966	0.812933
---	----------	----------	---------	----------

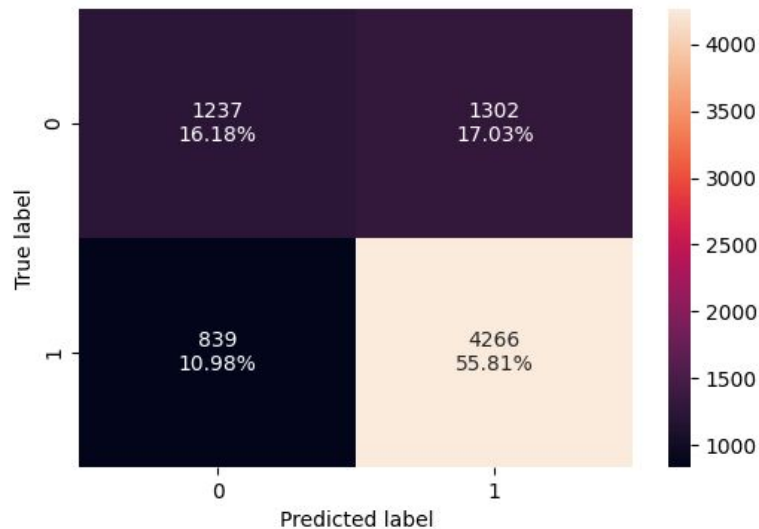
- Post-tuning, the training set performance is still nearly perfect, with an accuracy of 99.52%, recall of 99.95%, precision of 99.34%, and an F1 score of 99.64%.
- On the test data, there's a significant improvement in the model's performance: accuracy is now 73.13%, recall has increased to 87.42%, precision is at 75.97%, and the F1 score has improved to 81.29%.

Model Improvement - Bagging vs Random Forest




	Accuracy	Recall	Precision	F1
0	1.0	1.0	1.0	1.0





	Accuracy	Recall	Precision	F1
0	0.719911	0.835651	0.766164	0.7994



Model Improvement - Bagging. Random Forest.

Train data VS test data

Define the Random Forest classifier with balanced class weights and a random state.

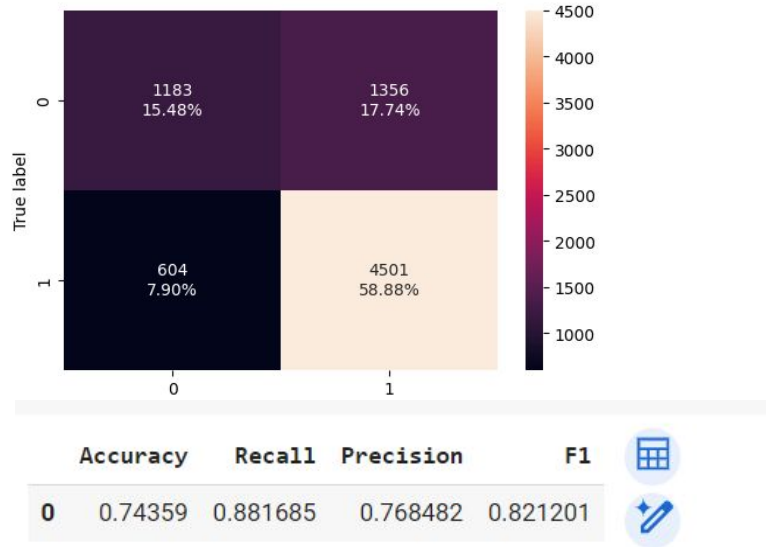
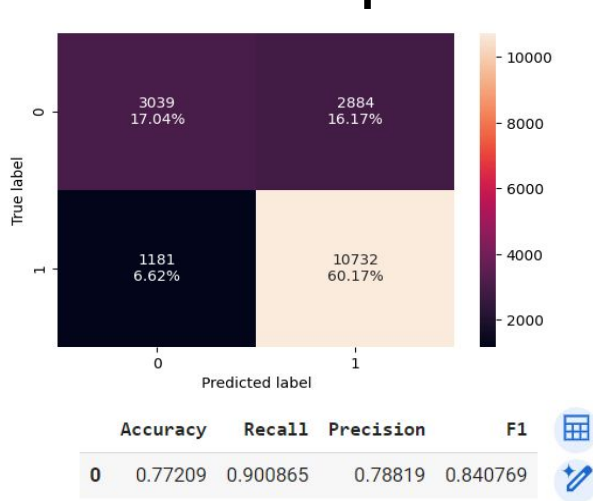
Fit the model on the training data.

Predict and evaluate the model's performance on both training and test sets.

Model Performance:

- Before Tuning:
 - Training: 100% Accuracy, Recall, Precision, F1
 - Test: 71.99% Accuracy, 83.57% Recall, 76.62% Precision, 79.94% F

Model Improvement - Baddina. Random Forest



Model Performance after Tuning:

- Training: 77.21% Accuracy, 90.09% Recall, 78.82% Precision, 84.08% F1
- Test: 74.36% Accuracy, 88.17% Recall, 76.85% Precision, 82.12% F1

Improvement Analysis: Tuning the Random Forest improved the test recall (from 83.57% to 88.17%) and F1 score (from 79.94% to 82.12%), indicating better generalization and a more balanced performance between precision and recall. The slight drop in training metrics indicates reduced overfitting.

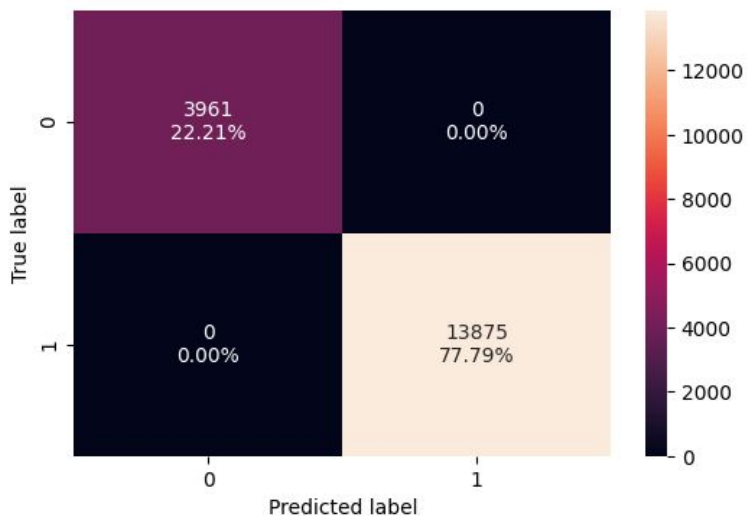
Model Building - Boosting

- Please mention regarding the model building steps of Adaboost and Gradient Boost
- Comment on the model performance

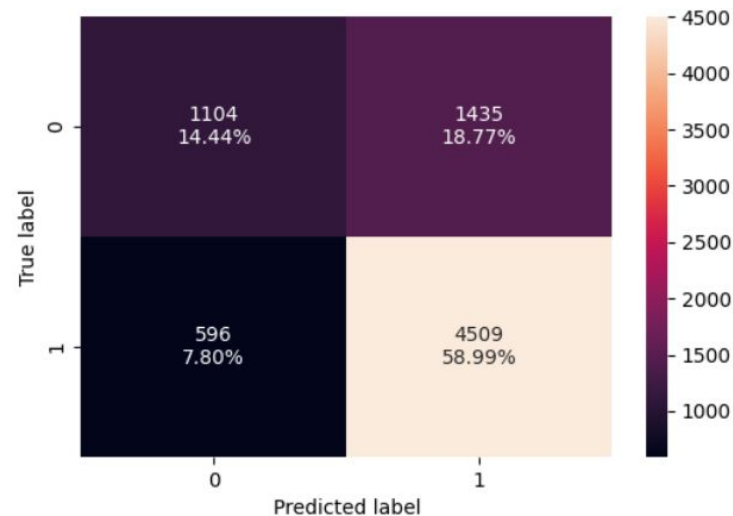
Note: *You can use more than one slide if needed*

Note: *Building XGBoost is optional*

Model Building - Boosting. Adaboost Train VS Test



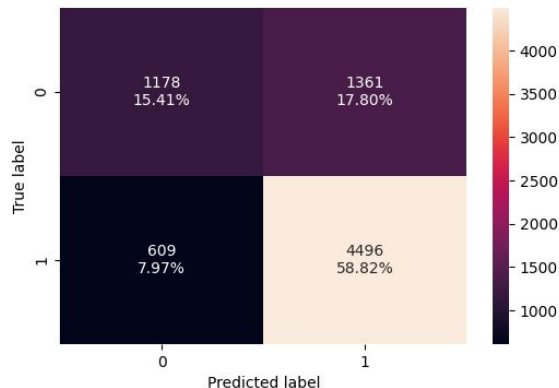
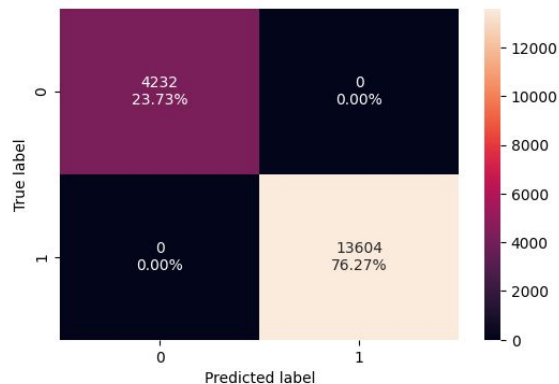
	Accuracy	Recall	Precision	F1
0	0.738058	0.886259	0.760937	0.81883



	Accuracy	Recall	Precision	F1
0	0.734301	0.883252	0.75858	0.816182



Adaboost. Confusion matrix for train data and test data on tuned estimator



	Accuracy	Recall	Precision	F1
0	0.75527	0.88777	0.777418	0.828938

Adaboost Classifier Model Building Steps:

Define the AdaBoost Classifier with a specified random state for reproducibility.
Fit the model to the training data.
Make predictions on the training set and evaluate the model using metrics such as accuracy, recall, precision, and F1 score.
Generate a confusion matrix to understand the model's prediction distribution.

Performance Overview:

- The initial AdaBoost model achieved an accuracy of 73.81%, recall of 88.33%, precision of 76.09%, and F1 score of 81.88% on the training set.
- On the test set, it achieved an accuracy of 73.43%, recall of 88.33%, precision of 75.86%, and F1 score of 81.62%.

	Accuracy	Recall	Precision	F1
0	0.742282	0.880705	0.767628	0.820288

Model Building - Boosting

Hyperparameter Tuning for AdaBoost:

Setup the AdaBoost Classifier with a range of hyperparameters for grid search.

Used GridSearchCV with cross-validation to find the best hyperparameters.

Evaluated the tuned model using the same metrics as before.

Performance Improvement:

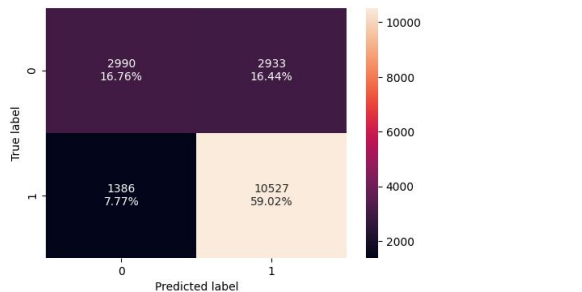
- The tuned AdaBoost model showed an increase in performance on the training set with an accuracy of 75.53%, recall of 88.78%, precision of 77.74%, and an F1 score of 82.89%.
- The test set performance also improved, showing an accuracy of 74.23%, recall of 88.07%, precision of 76.76%, and an F1 score of 82.03%.

Conclusion:

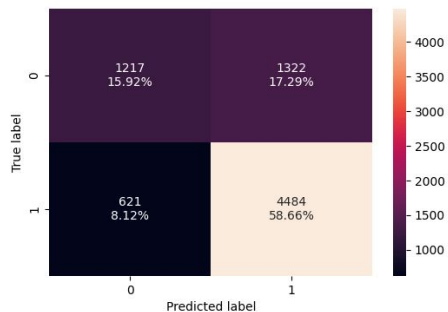
Hyperparameter tuning led to performance improvement for the AdaBoost Classifier, indicating that the adjustments made to the parameters better captured the underlying patterns in the data without overfitting. This is evidenced by increased accuracy and F1 score on both the training and test sets, as well as high recall, which is important when both false negatives and positives are equally important. The model is now more generalized and likely to perform better on unseen

Model Building - Boosting. Gradient Boost Classifier

Training VS Test performance

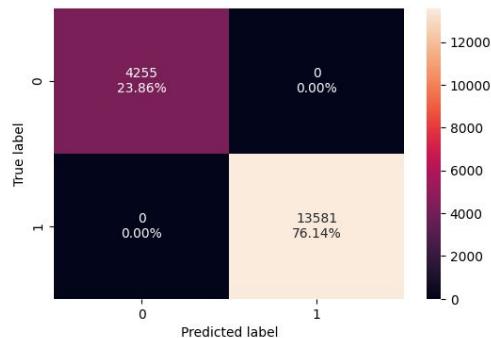


	Accuracy	Recall	Precision	F1
0	0.757849	0.883657	0.782095	0.82978

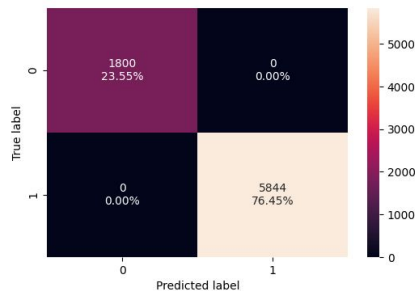


	Accuracy	Recall	Precision	F1
0	0.745814	0.878355	0.772305	0.821923

Confusion matrix for train and test data on tuned estimator



	Accuracy	Recall	Precision	F1
0	0.753756	0.885671	0.776894	0.827724



	Accuracy	Recall	Precision	F1
0	0.745029	0.881489	0.770021	0.821993

Model Building Boosting. Gradient Boost Classifier

Gradient Boosting Classifier Building Steps:

Initialize the Gradient Boosting Classifier with a fixed random state for consistency.

Fit the model to the training data to learn from it.

Predict on the training data and generate a confusion matrix to visualize the predictions.

Evaluate the model's training performance with accuracy, recall, precision, and F1 score metrics.

Model Performance Before Tuning:

- Training set: Accuracy of 75.78%, recall of 88.37%, precision of 78.21%, and F1 score of 82.98%.
- Test set: Accuracy of 74.58%, recall of 87.84%, precision of 77.23%, and F1 score of 82.19%.

Hyperparameter Tuning Steps:

Configure a range of hyperparameters for the Gradient Boosting Classifier.

Employ GridSearchCV with cross-validation to determine the best combination of parameters.

Fit the tuned model to the training data.

Performance Improvement Observation:

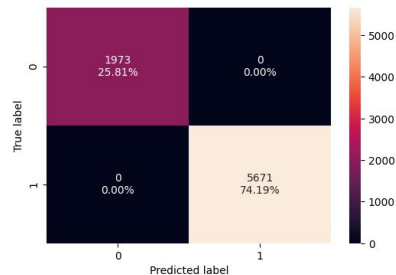
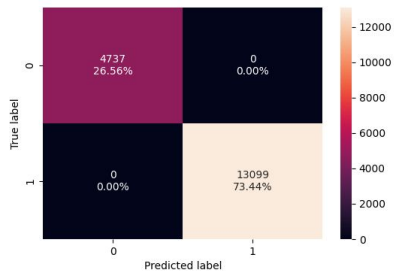
- The tuned Gradient Boosting Classifier's training performance slightly decreased, with an accuracy of 75.38%, recall of 88.57%, precision of 77.69%, and F1 score of 82.77%.
- The test performance remained relatively stable post-tuning, with accuracy at 74.50%, recall at 88.15%, precision at 77.00%, and F1 score at 82.20%.

The performance of the Gradient Boosting Classifier after hyperparameter tuning did not change significantly on the test set, indicating that the original model was already well-suited to the data. The slight decrease in training metrics after tuning suggests that the model may have become more generalized, reducing overfitting. The confusion matrices reinforce this, showing a balanced distribution of predictions. Overall, the Gradient Boosting Classifier, with or without tuning, provides robust predictions, as reflected in the consistently high recall and F1 score.

Model Building - Boosting. Gradient Boost Classifier

Training VS Test performance

Steps for XGBoost Classifier Model Building:



Define XGBoost Classifier with initial settings, including random state and evaluation metric.

Fit the classifier to the training data.

Make predictions on the training data and compute the confusion matrix.

Evaluate the model's training performance using accuracy, recall, precision, and F1 score metrics.

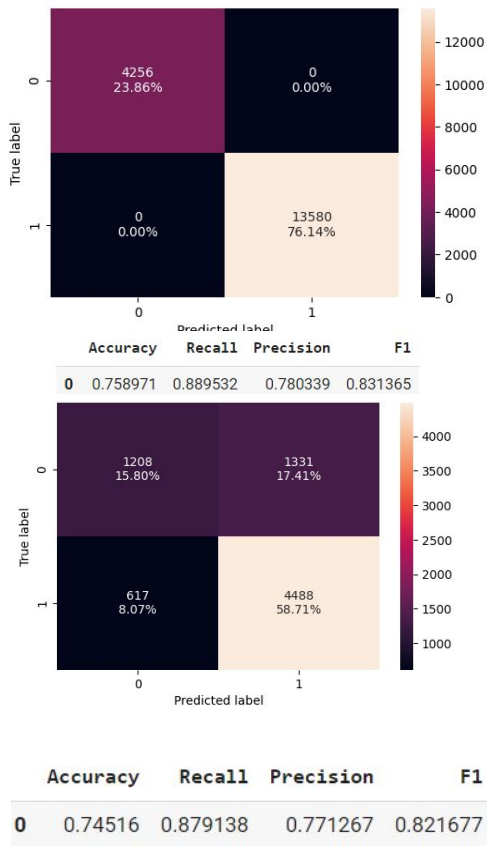
Repeat steps 3 and 4 for the test data to evaluate the model's performance on unseen data.

Performance Overview:

- Before tuning, the XGBoost model had a training accuracy of 84.09% and an F1 score of 88.65%, indicating a strong fit on the training data.
- On the test data, it achieved an accuracy of 72.66% and an F1 score of 80.61%, which is lower than the training performance, suggesting some overfitting.

Model Building - Boosting. Gradient Boost Classifier

Training VS Test performance on tuned estimator



Hyperparameter Tuning for XGBoost: Setup the XGBoost Classifier with a range of hyperparameters for grid search. Use GridSearchCV with cross-validation to find the optimal hyperparameters. Evaluate the tuned model using confusion matrices and performance metrics on both the training and test data.

Improvement Analysis:

- The tuned XGBoost model's training performance showed a slightly reduced accuracy of 75.90% and F1 score of 83.14%, which is closer to the test performance.
- The test set metrics showed an improved accuracy of 74.52% and F1 score of 82.17% after tuning.
- This suggests the tuning helped reduce overfitting and improve the model's ability to generalize to new data.

Hyperparameter tuning of the XGBoost model improved its generalization performance on the test data, as indicated by a higher F1 score and better balance between precision and recall. This suggests that the tuning process helped the model to make better predictions on unseen data, making it more reliable for practical applications. The accuracy and F1 score improvements also imply a more balanced trade-off between false positives and negatives, which is crucial when both have equal importance.

Model Improvement - Boosting

- Comment on the improvement in the model performance by hyperparameter tuning
- The model building steps of the Stacking Classifier

Model Improvement - Boosting

Improvement in Model Performance by Hyperparameter Tuning:

- Hyperparameter tuning of the XGBoost Classifier improved its generalization ability on unseen data.
- The tuned XGBoost model demonstrated better performance metrics (Accuracy, Recall, Precision, F1) compared to the untuned version.
- This improvement is particularly noted in the F1 score, which increased from 0.806 to 0.822, indicating a more balanced classifier.

Model Building Steps for the Stacking Classifier:

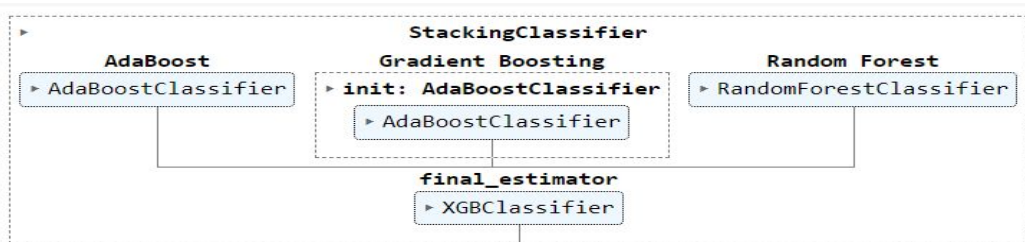
Assemble an array of estimators, including AdaBoost, Gradient Boosting, and Random Forest.

Define a final estimator, which is the tuned XGBoost model.

Create the Stacking Classifier with the array of estimators and the final estimator.

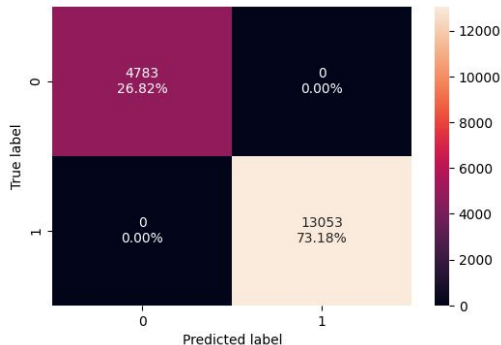
Fit the Stacking Classifier to the training data.

Evaluate the model's performance using accuracy, recall, precision, and F1 score on both the training and test sets.

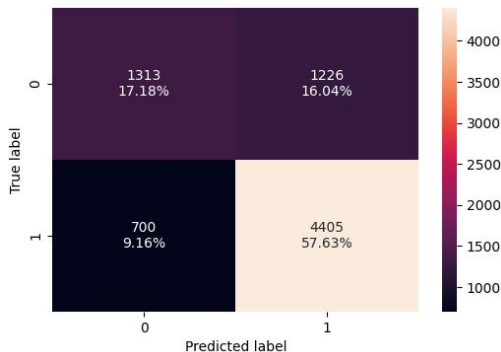


Model Improvement - Boosting

Training VS Test performance on tuned estimator



	Accuracy	Recall	Precision	F1
0	0.766764	0.873248	0.796982	0.833373



	Accuracy	Recall	Precision	F1
0	0.748038	0.86288	0.782277	0.820604

Model Performance and Conclusion:

- The Stacking Classifier achieved an accuracy of 74.80%, a recall of 86.29%, precision of 78.23%, and an F1 score of 82.06% on the test set.
- These results indicate that the Stacking Classifier, which combines the predictions of several base estimators, provides a boost in predictive performance compared to individual models.
- The use of a tuned XGBoost model as the final estimator in the stacking approach likely contributed to this improvement, optimizing the strengths of the individual models and leading to a better overall prediction on the test data.