EasyVisa Project

Ensemble Technique and Model Tuning Project

Contents

Business Problem Overview and Solution Approach

The Problem

- U.S. businesses face strong demand for talent but struggle to find and attract the right individuals needed to stay competitive.
- The Immigration and Nationality Act allows foreign workers to work in the U.S. while protecting U.S. workers by ensuring employers follow strict hiring regulations, overseen by the Office of Foreign Labor Certification.
- The OFLC certifies job applications for foreign workers only when employers prove there are not enough qualified U.S. workers available at prevailing local wages.
- In FY 2016, the OFLC saw a 9% rise in applications, processing over 775,000 requests for nearly 1.7 million positions, making case reviews increasingly challenging due to growing demand.
- Due to the rising number of applicants each year, the OFLC has hired EasyVisa to implement a machine learning-based solution for shortlisting candidates with higher chances of visa approval.
- Data has been supplied to EasyVisa for analysis and inference from which data-driven decisions can be made.

Solution Approach

- The approach is to draw insight from the data provided, perform exploratory data analysis, make important observations, and build a classification model that can help facilitate visa approval process and predict visa outcomes

Data Background and Content

The data contains the different attributes of employee and employer. The data has 25480 rows and 12 columns. The columns types are integer, object and float (2 int, 9 object and 1 float). The data has the following columns:

- 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
- 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

There are no duplicates in the data.

There are no missing values in the dataset

EDA Results

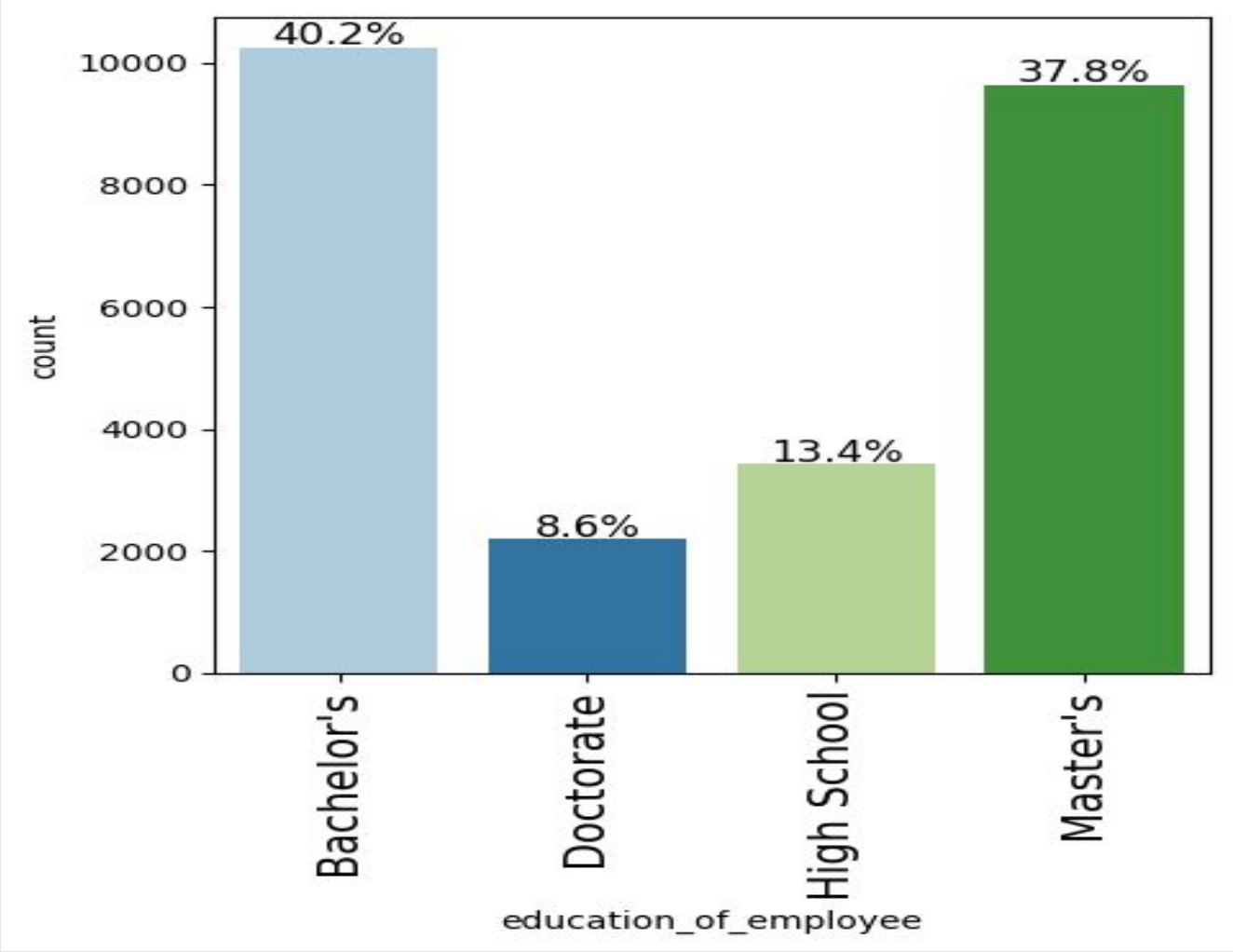
- The minimum number of employees is -26 which does not really make sense.
- The year of establishment of companies seeking visa approval is between 1800 to 2016.
- The average prevailing wage is 74,455.81, and the large gap between the 75th percentile and the maximum suggests potential outliers.
- We will fix the negative values in the number of employees by taking absolute values of the column. There are about 33 negative values.
- According to the data, most of the employees are from Asia (16861) followed by Europe (3732). Employees from Oceania are the least.
- Of all the employees, the majority hold a Bachelor's degree, followed by those with a Master's degree. High School graduates make up a smaller portion, while the fewest have a Doctorate.
- There are more employees with job experience(14802) than those without job experience (10678).
- There are more employees requiring job training(22525) than those who do not (2955).
- Among all regions of employment, the Northeast has the highest number of employees (7,195), closely followed by the South (7,017) and the West (6,586). The Midwest has fewer employees (4,307), while the Island region has the least (375).
- The majority of employees are paid on a yearly basis (22,962), followed by those paid hourly (2,157). A smaller number are paid weekly (272), and the fewest receive monthly wages (89).
- Most positions are full-time (22,773), while other positions are non-full-time (2,707).
- Most of the applications for visa were certified (17018) while others were denied (8462).

EDA Results

- The number of unique values in the case_id column is 25480, this means that all the values are unique. So, we will drop this column as this gives no special information about the data

EDA Results

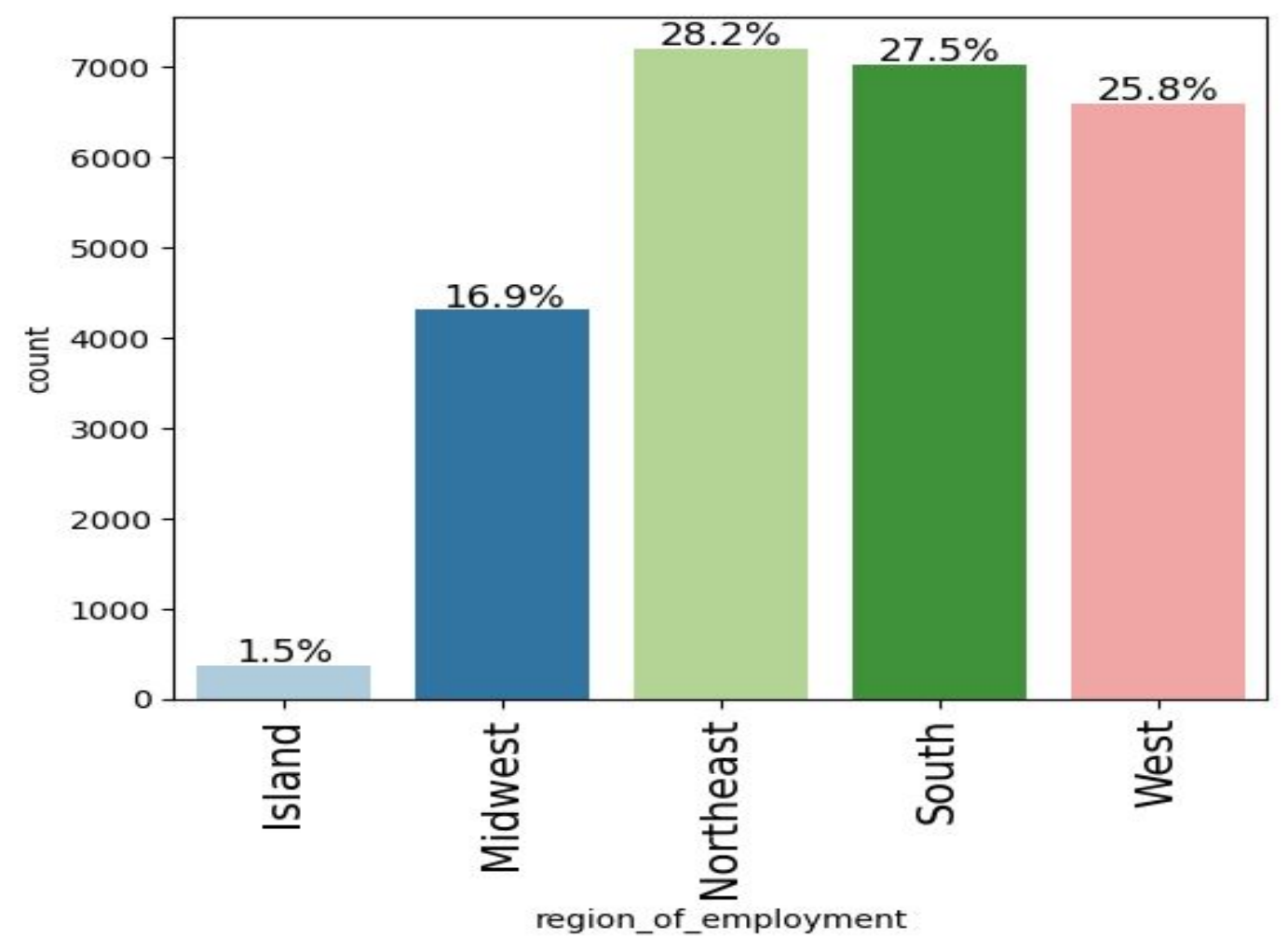
Univariate analysis: Observation on Education of Employee



- The majority of applicants hold a Bachelor's degree (40.2%), followed by those with a Master's degree (37.8%).
- A smaller portion of applicants have a High School Certificate while the least number of employee possess a Doctorate degree (8.6%).

EDA Results

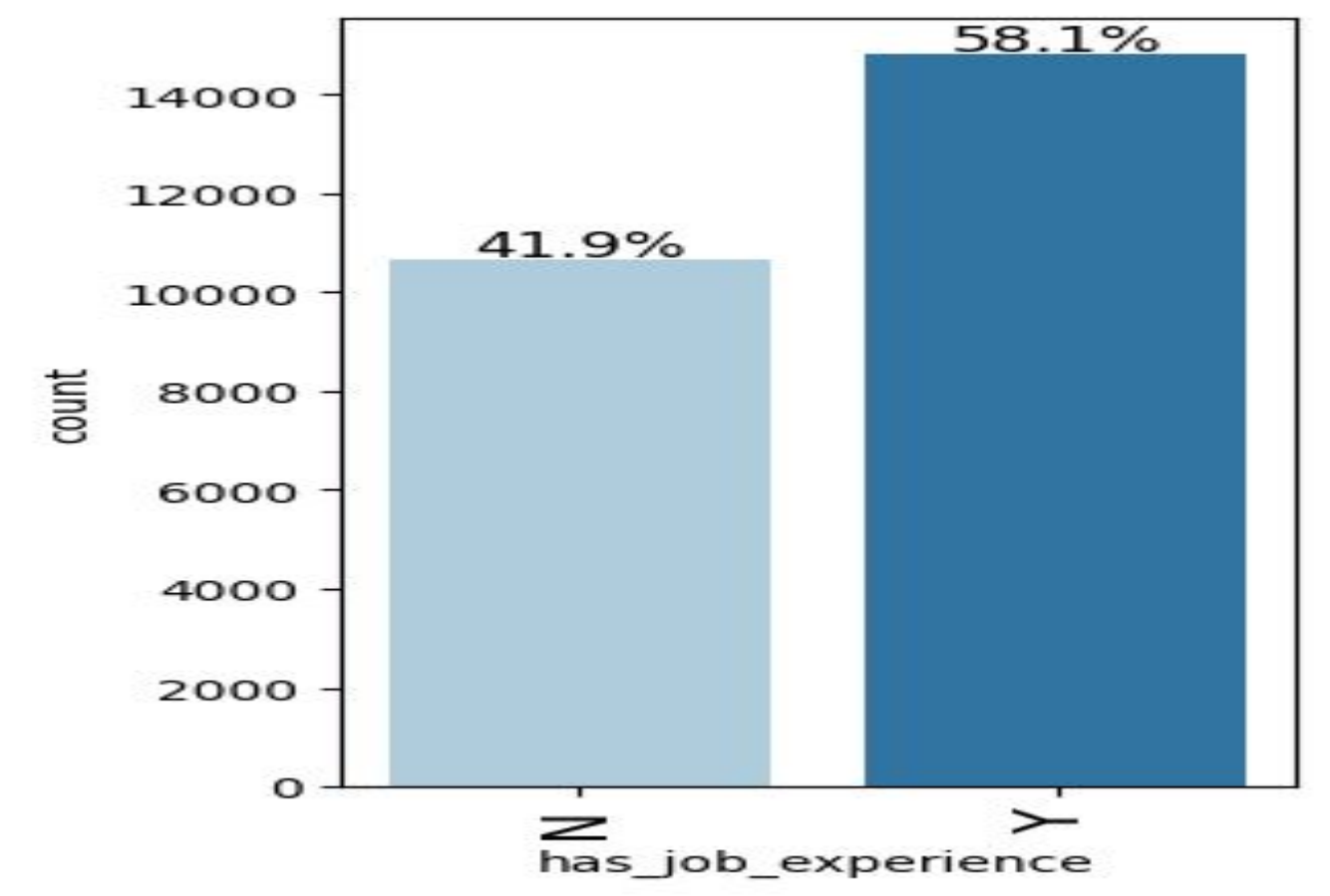
Univariate analysis: Observation on Region of Employment



- The Northeast region accounts for the highest percentage of applicants (28.5%), followed closely by the South (27.5%) and the West (25.8%). The Midwest follows with 16.9%, while the Island region has the smallest share (1.5%).

EDA Results

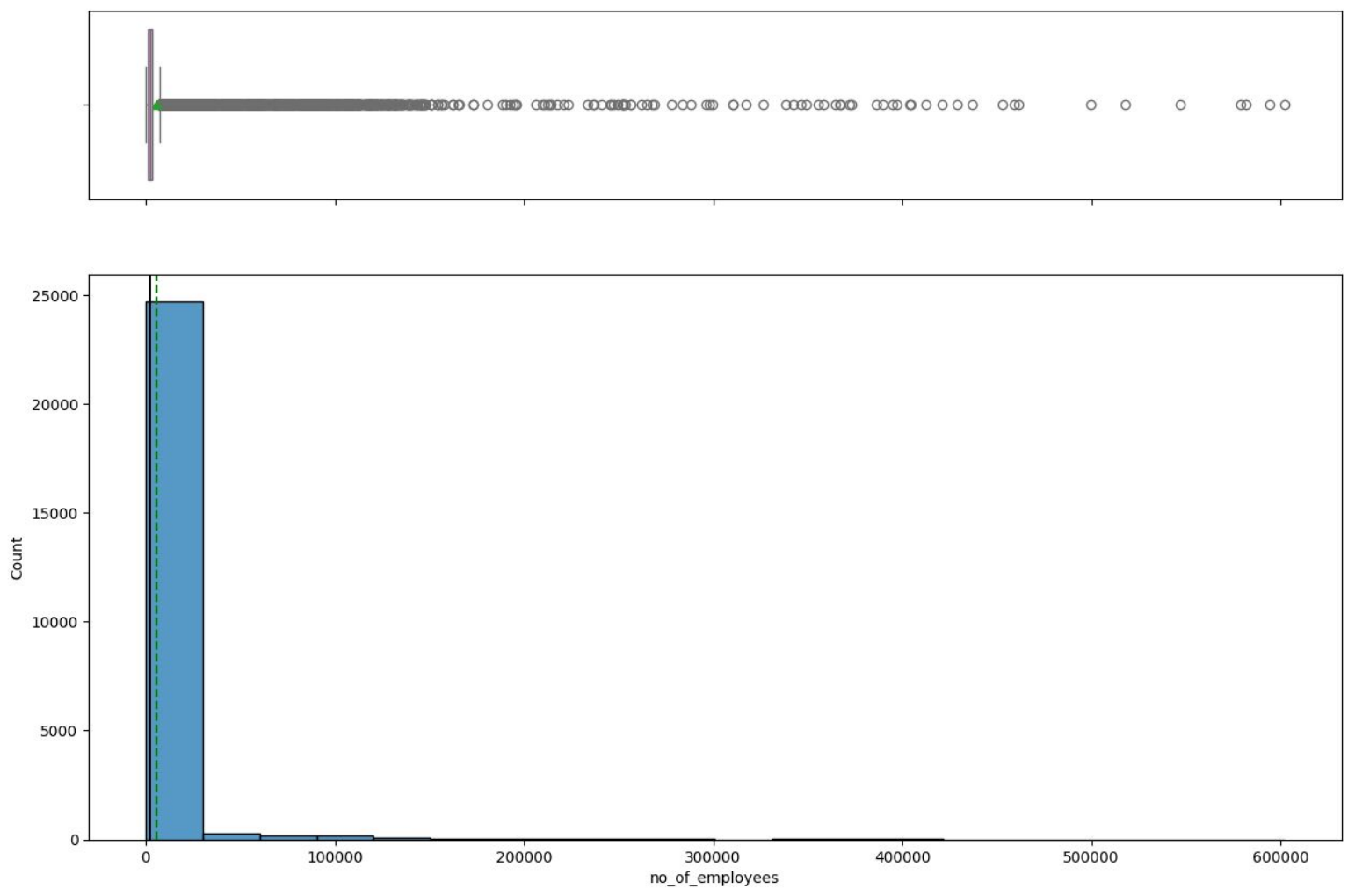
Univariate analysis: Observation on Job Experience



- More applicants have job experience (58.1%) while others (41.9%) do not have job experience.

EDA Results

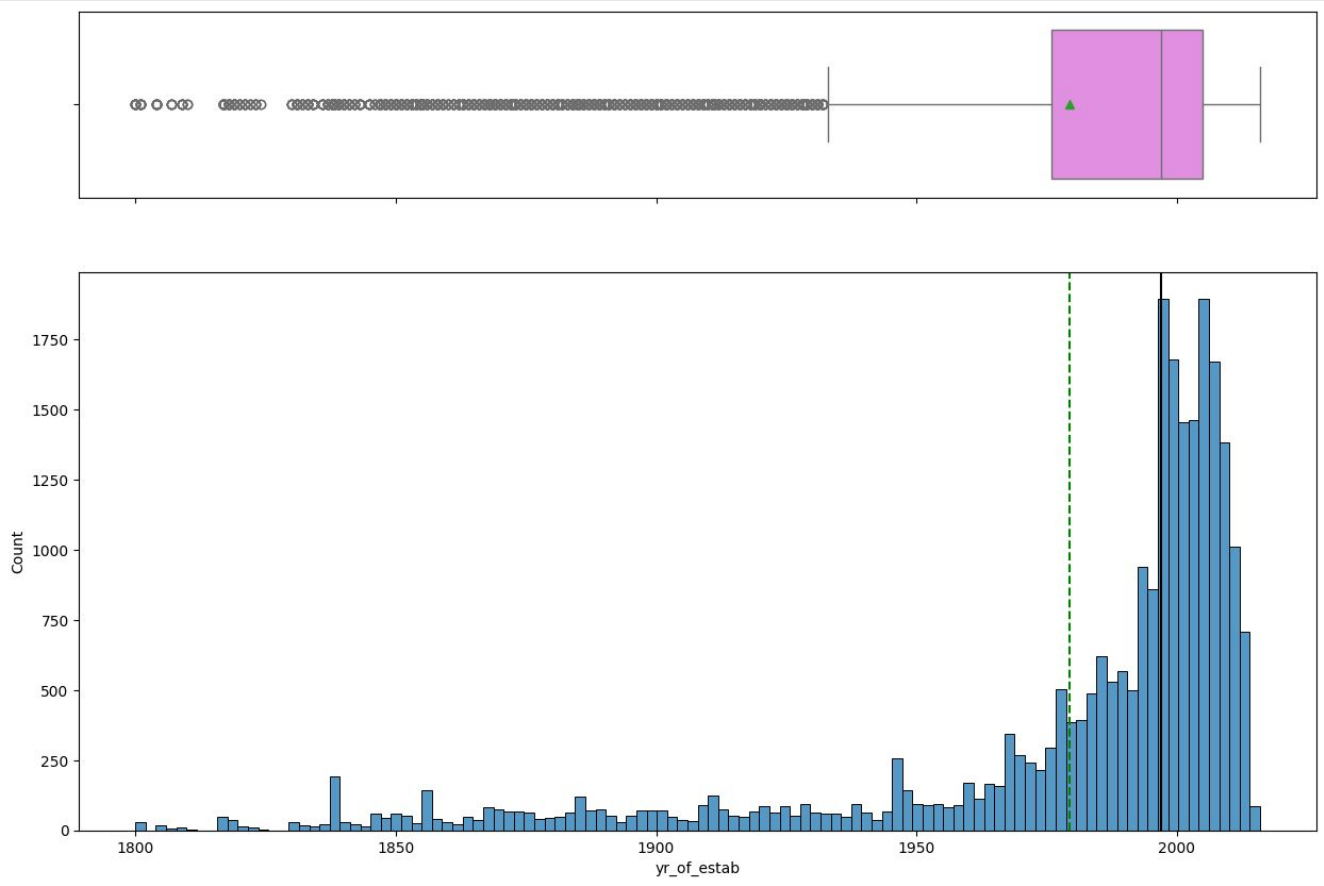
Univariate analysis: Observation on Number of Employees



- The number of employees is heavily right-skewed, indicating that most companies have relatively small workforces.
- A few companies have over 400,000 employees. These large organizations likely operate multiple offices globally.

EDA Results

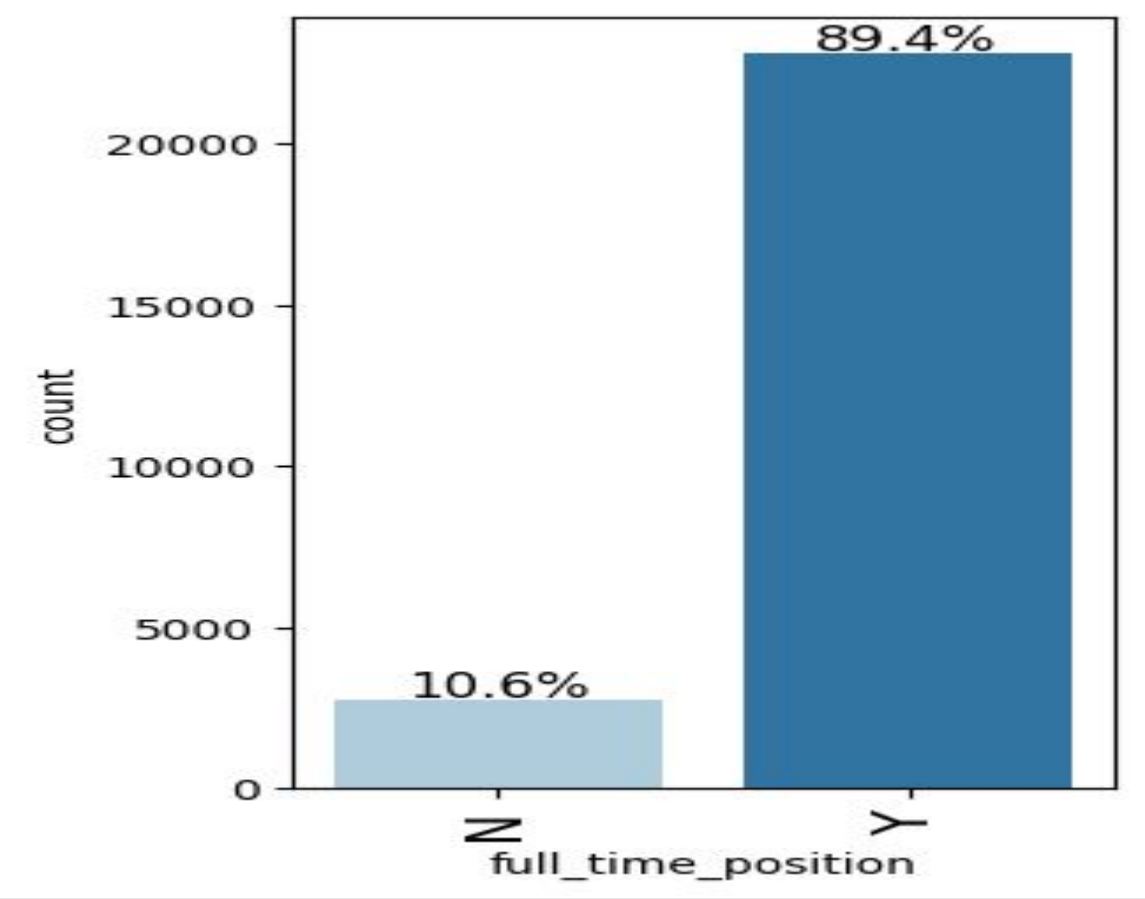
Univariate analysis: Observation on Year of Establishment



- The distribution of year of establishment is skewed to the left.
- There are few organizations that were established before 1850.
- The boxplot shows the presence of outliers.
- The surge in company formations in recent decades likely reflects increased demand for skilled migrant labor, driving higher volumes of employment-based visa applications.

EDA Results

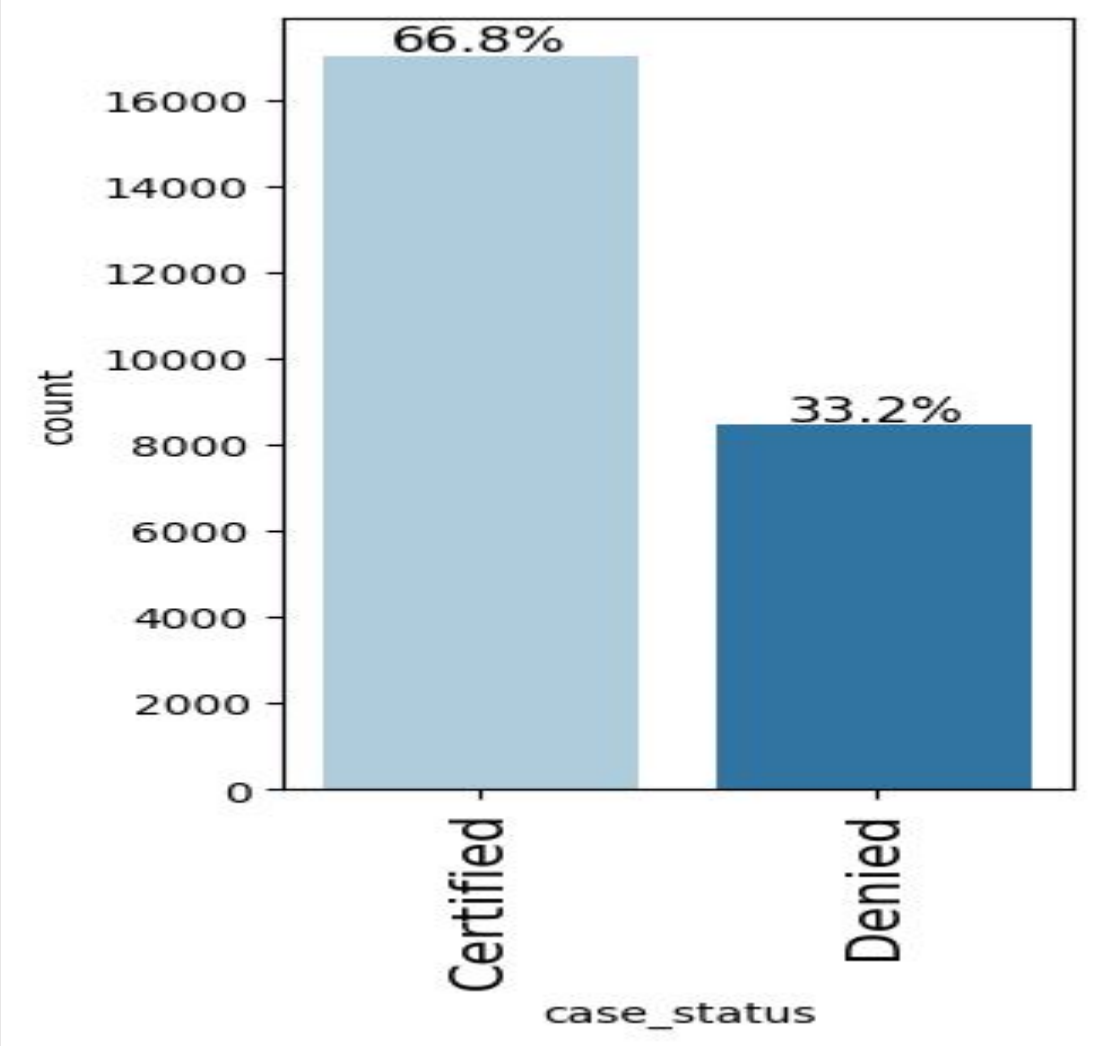
Univariate analysis: Observation on Full Time Position



- Here is a barplot for the full time position column.
- It shows that 89.4% of the positions are full time while 10.6% are part-time.

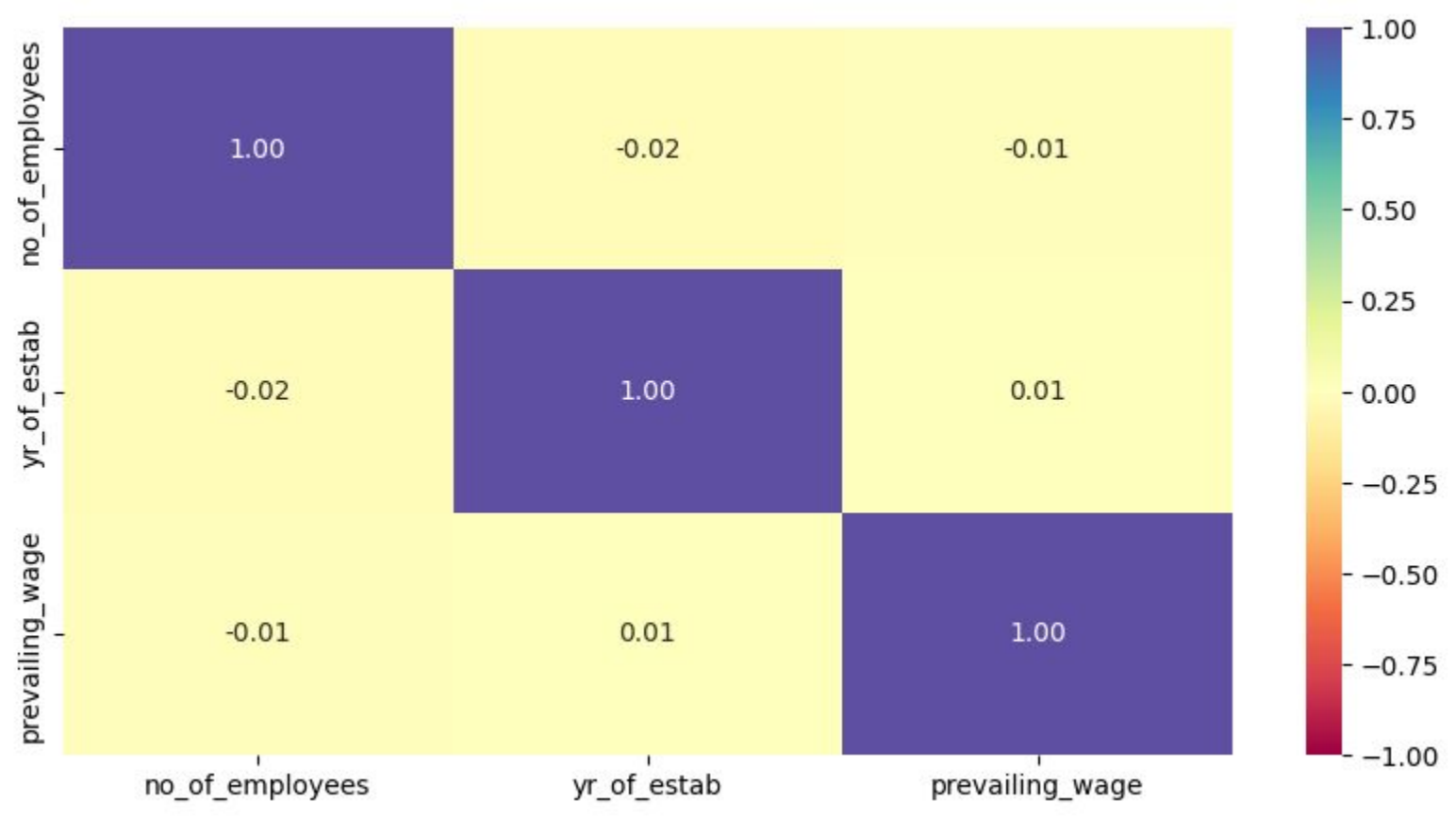
EDA Results

Univariate analysis: Observation on Case Status



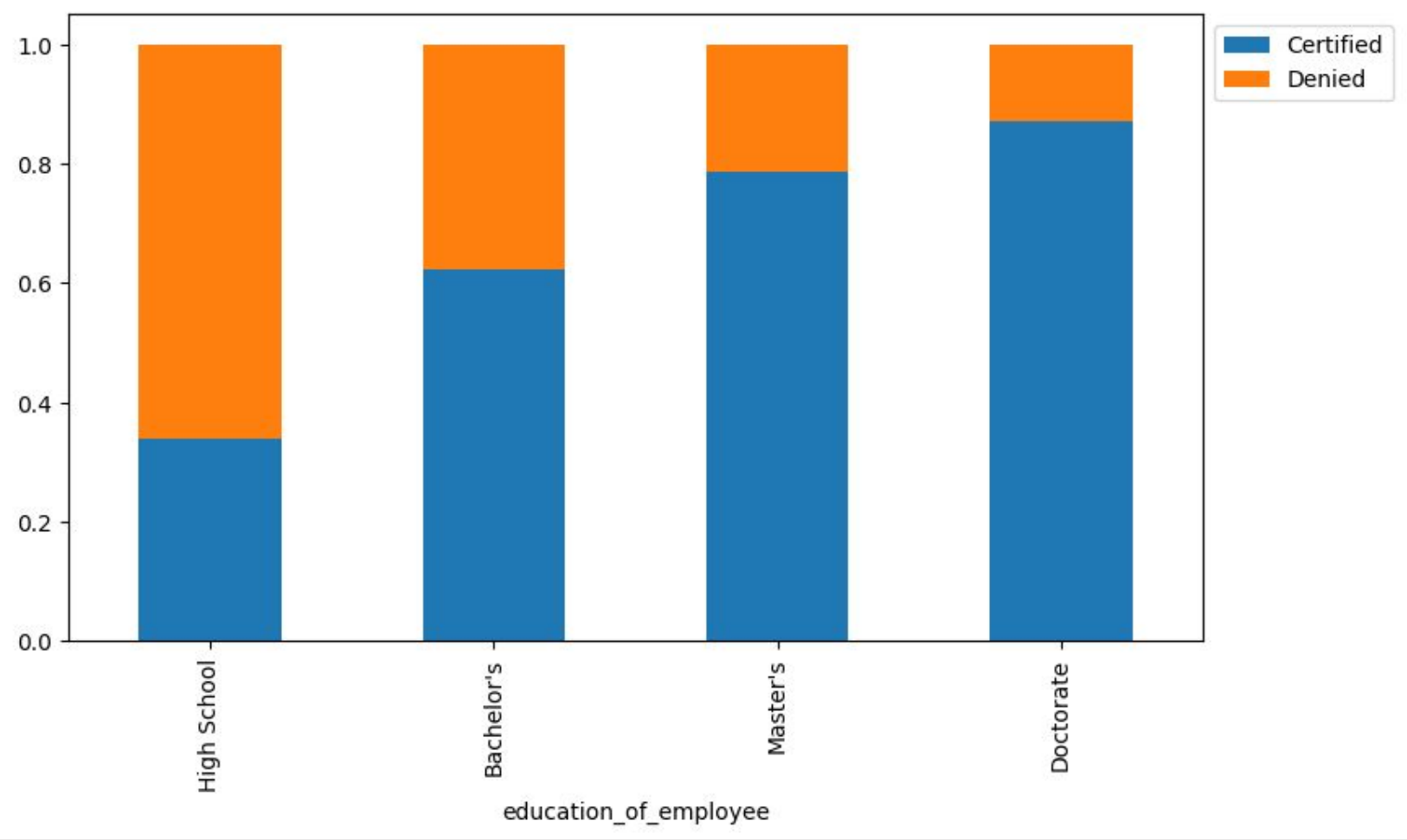
- Here is a bar plot showing the outcome of visa application. Some applications were granted while some were denied.
- It shows that 66.8% of the cases were certified while 33.2% were denied.

EDA Results
Bivariate analysis



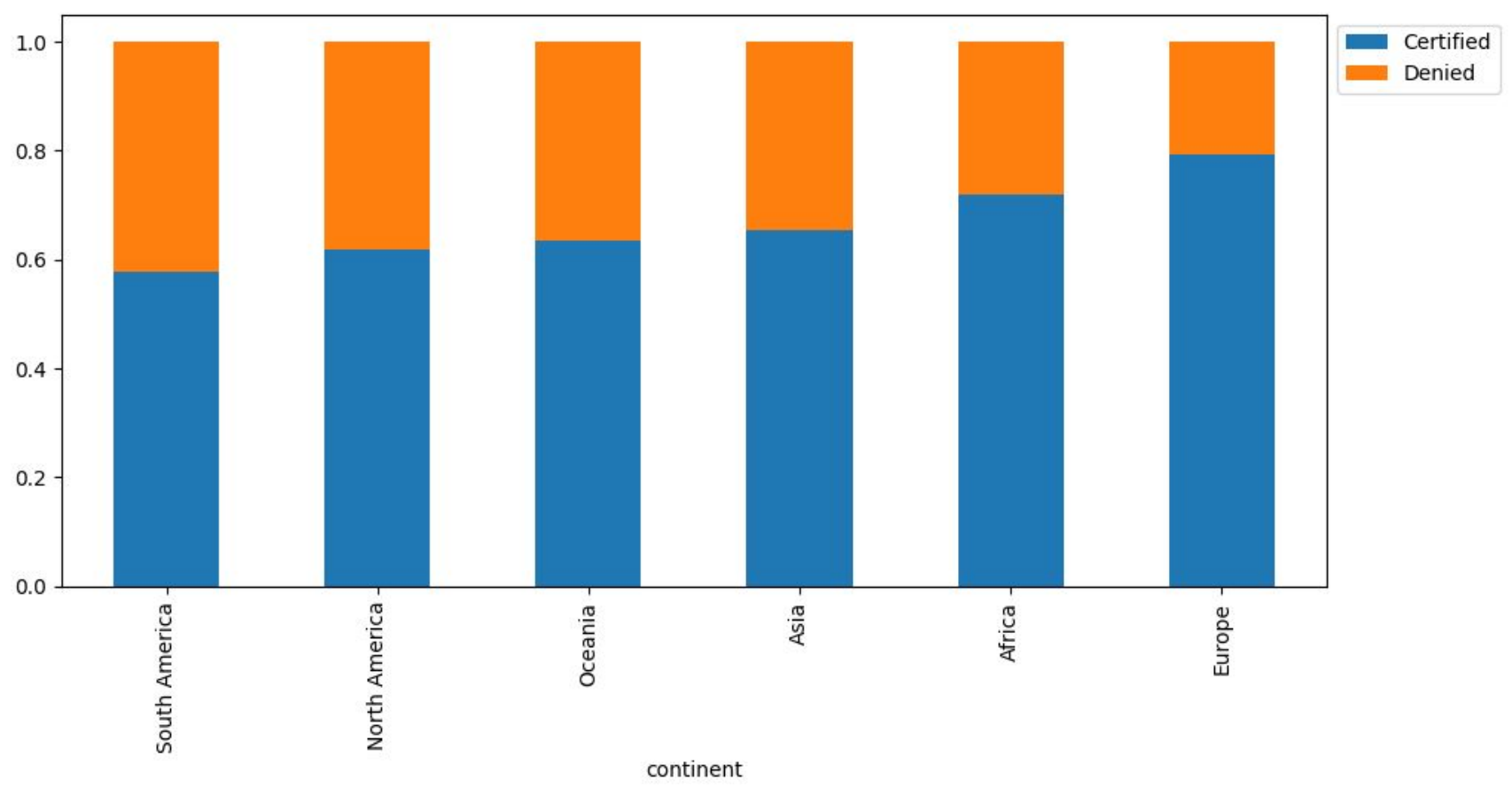
- There is no correlation between any two of the following variables: no_of_employees, yr_of_estab and prevailing_wage
- We will need further analysis to determine which variable significantly influence the case status.

EDA Results
Bivariate analysis: Education vs Case Status



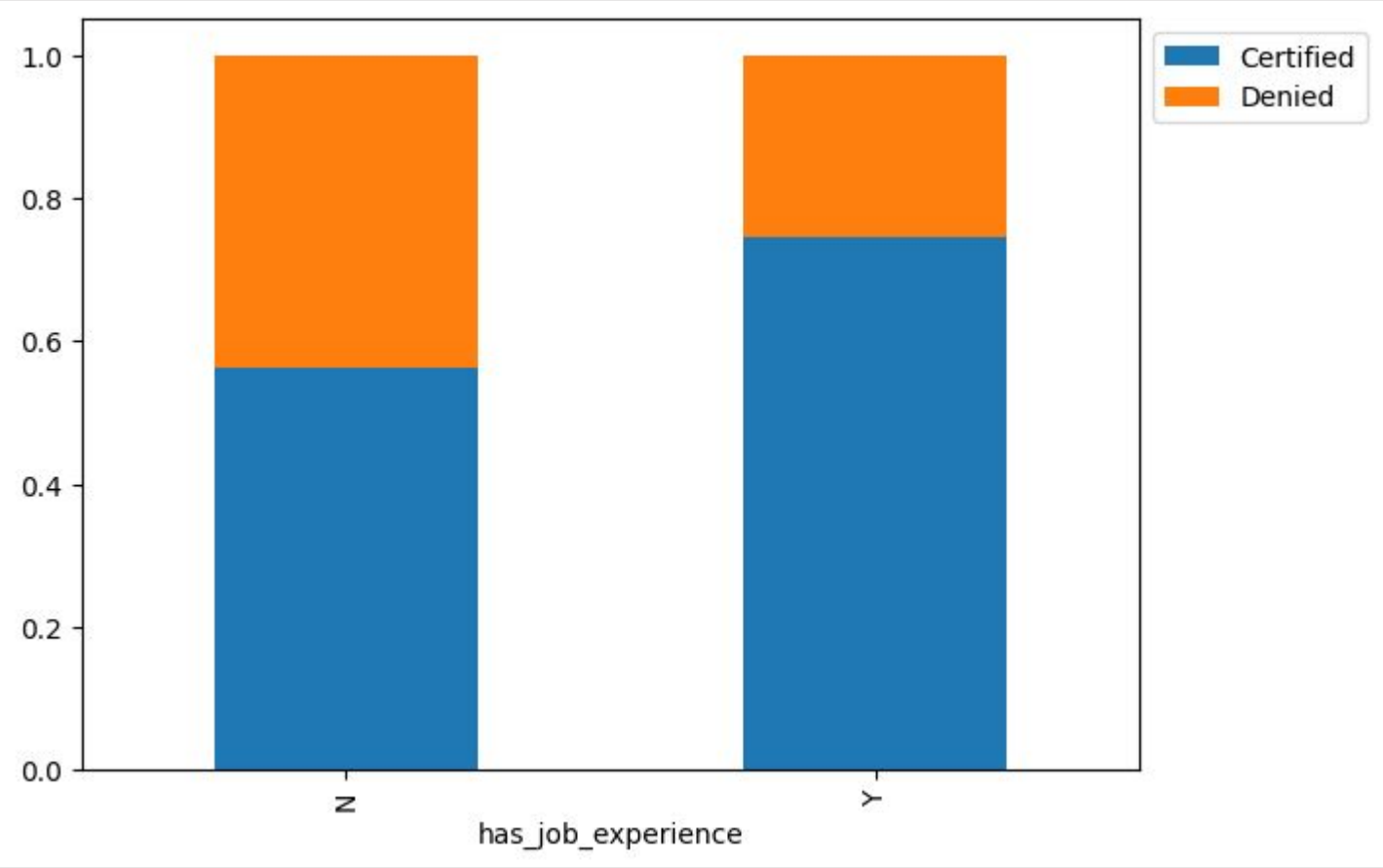
- Applicants who hold a doctorate degree are the most likely to be certified, followed by those with a master's degree, and then by those with bachelor's degree and those with high school certificate are the least to be certified.
- About 60% of applicants with Bachelor's degree got certified.
- Applicants with only high school certificate are highly likely to be denied.
- It is not surprising to see that the barplot suggests that higher level of education increase the likelihood of being certified.

EDA Results
Bivariate analysis: Continent vs Case Status



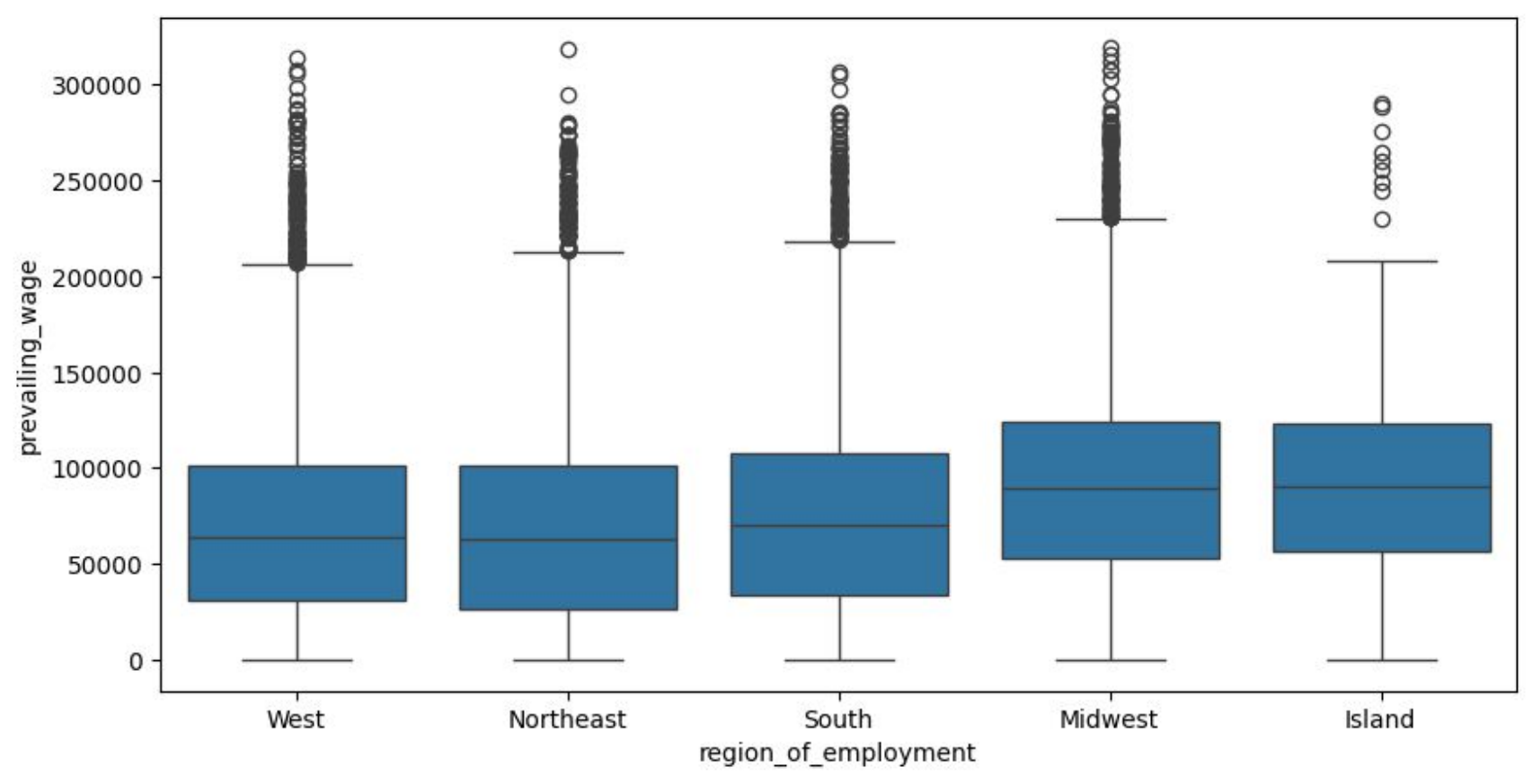
- Barplot shows that applicants who are from Europe and Africa are the most likely to be certified.
- About 60% of applicants from North America were certified.
- South American applicants were the most likely to be denied.
- Asia has the highest number of applications but Asian applicants are the third most likely to be certified.

EDA Results
Bivariate analysis: Job Experience vs Case Status



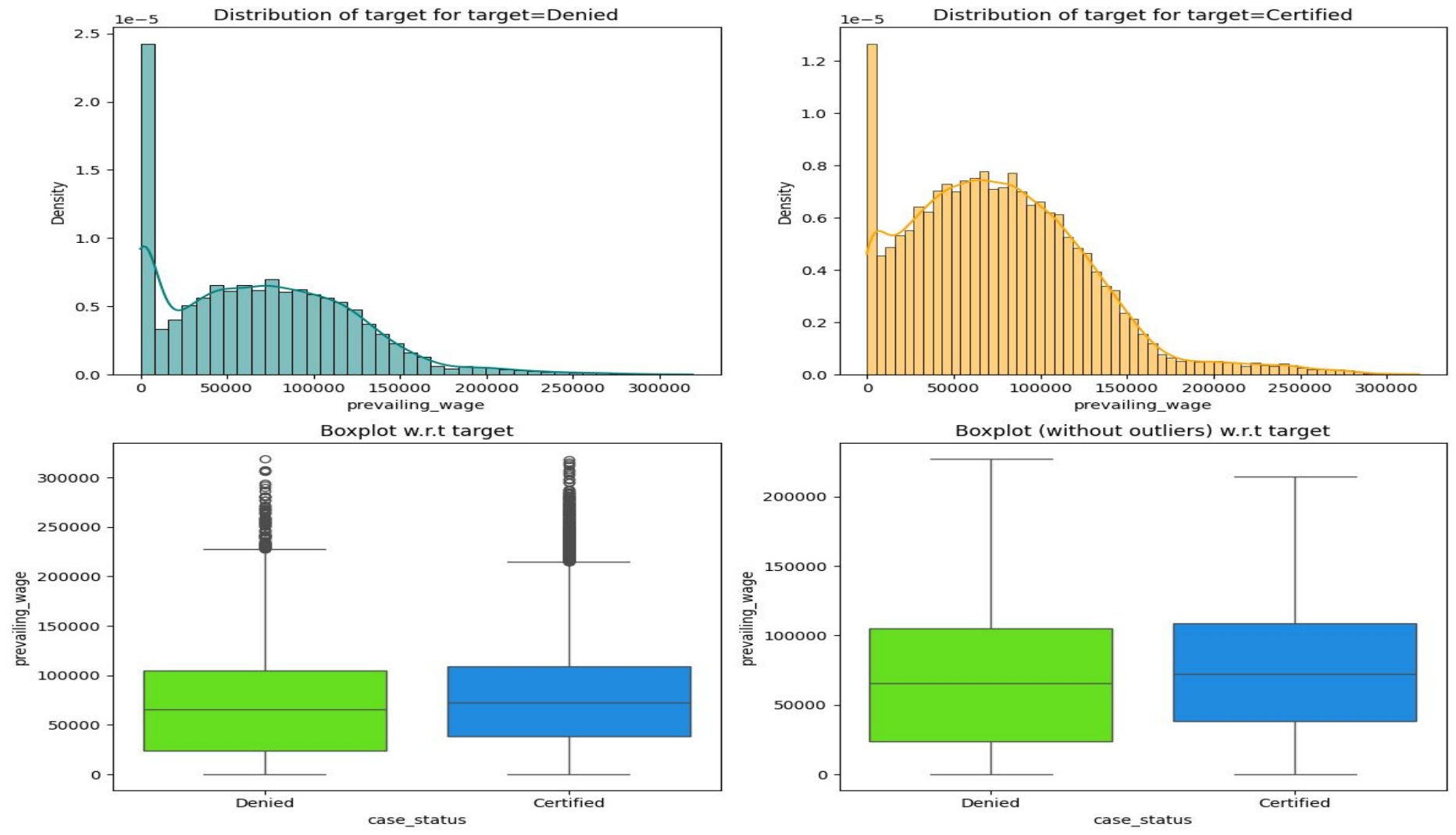
- Barplot shows that applicants with job experience are more likely to be certified.
- Less than 60% of applicants with no job experience got certified.
- This suggests that having job experience may be key to being certified.

EDA Results
Bivariate analysis: Prevailing wage vs Region of Employment



- Prevailing wage seems to be higher in Midwest and Island regions.
- All the regions have outlier prevailing wages. There are people in each region who earn significantly higher than other people working in the same region.
- Prevailing wage in the West and Northeast regions seem to be the same, and slightly lower than the prevailing wage in the South.

EDA Results
Bivariate analysis: Prevailing Wage vs Case Status

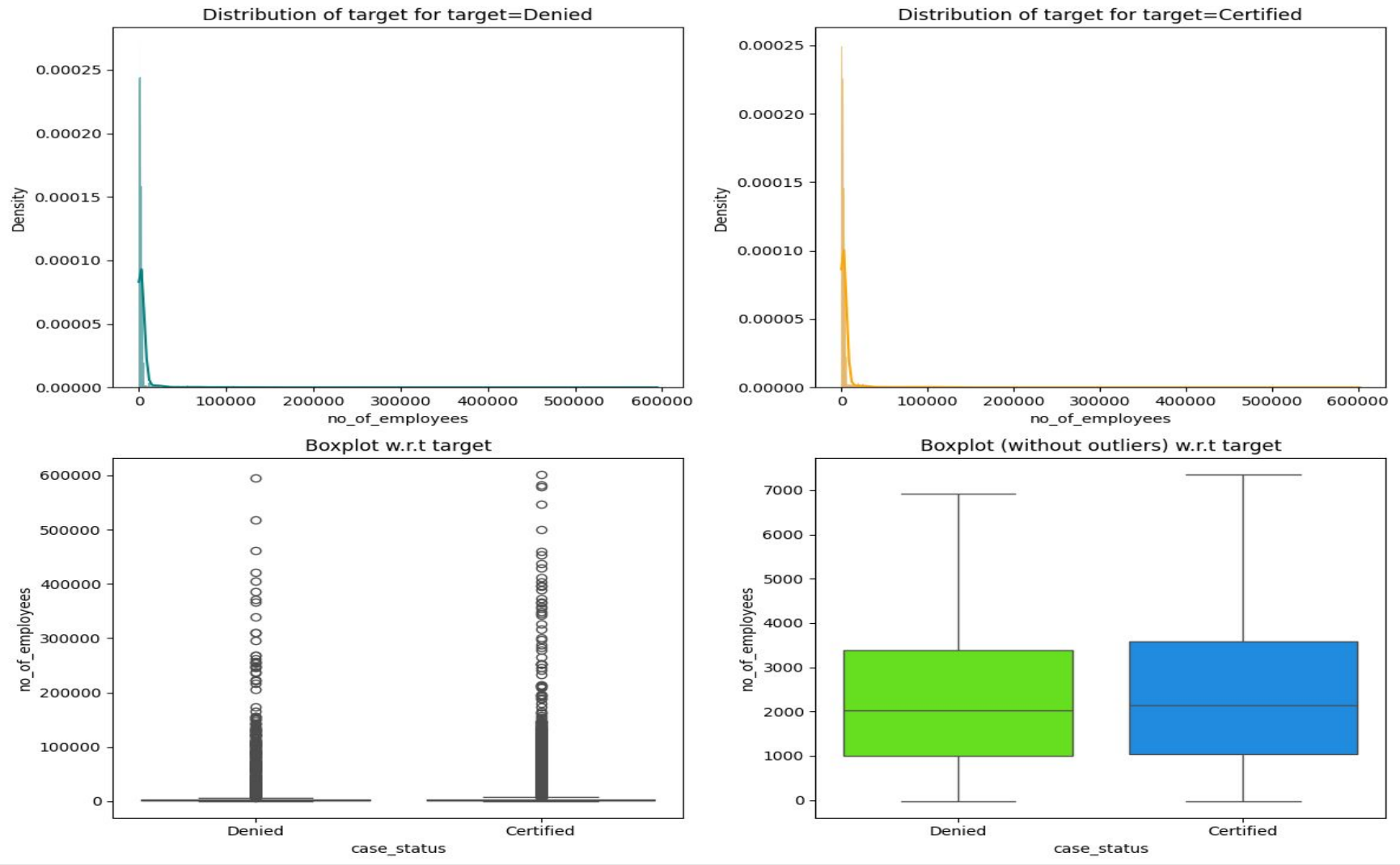


EDA Results
Bivariate analysis: Prevailing Wage vs Case Status

- For applicants who were not certified, the prevailing wage seem to vary.

- The second plot shows the distribution of prevailing wage for applicants who were certified
- This distribution is rightly skewed, there are some applicants with very high prevailing wages
- There are two Boxes of boxplots: boxplot w.r.t target and boxplot (without outliers) w.r.t target
- For the first Box of boxplot, there is a clear difference between the median. While applicants who were denied are likely to be associated with lower prevailing wage, there are some applicants who were associated significantly higher prevailing wages. For leads who were certified, the median prevailing wage is much higher, and the IQR is shorter, indicating less variability in the prevailing wages associated with applicants.
- For the second Box of boxplots, the outliers have been removed. We can see that some applicants who were certified were associated with slightly lower prevailing wage than those that were denied.
- The median prevailing wage for the certified applications is slightly higher than the median prevailing wage for denied applications.

EDA Results
Bivariate analysis: Number of Employees vs Case Status



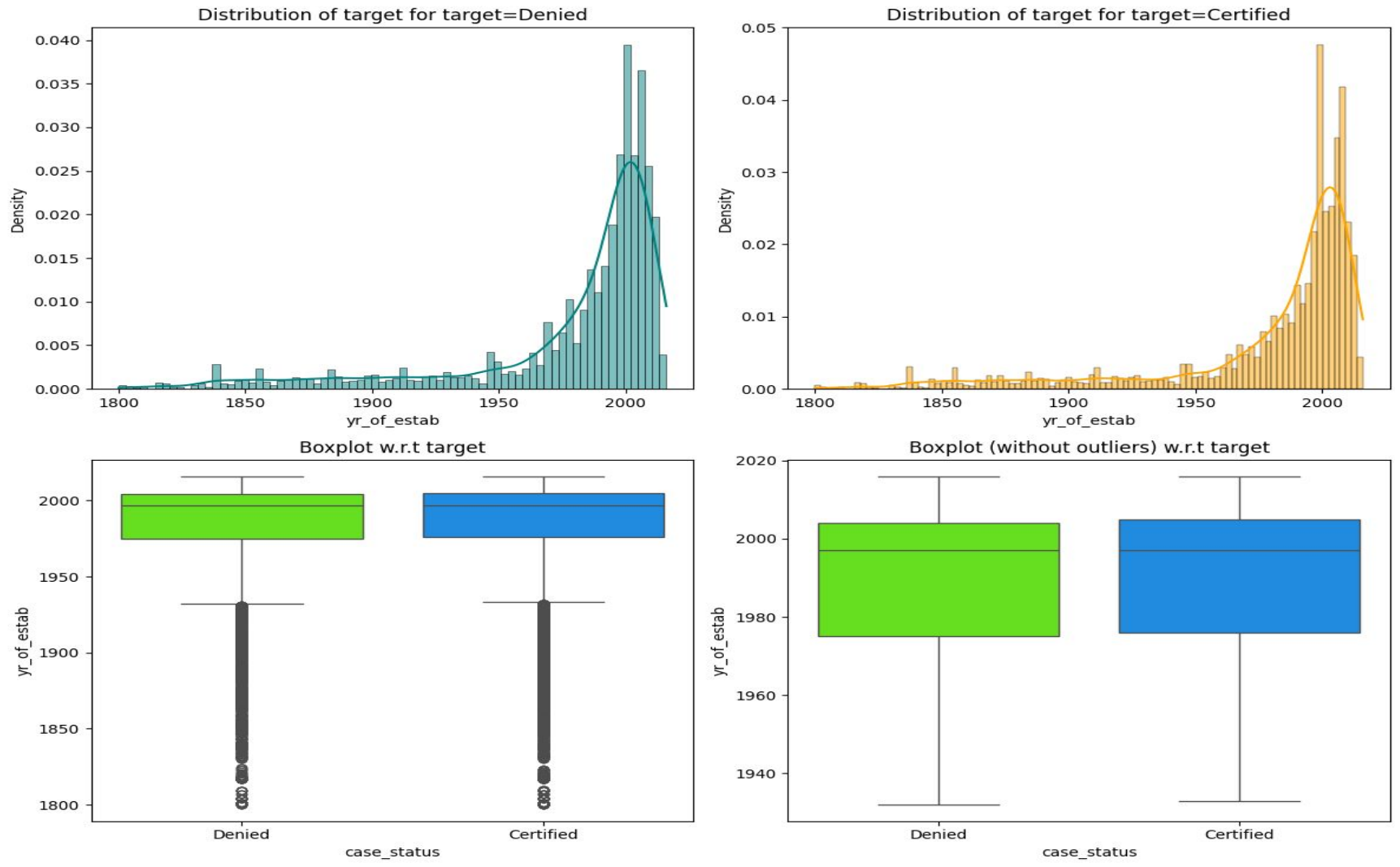
EDA Results
Bivariate analysis: Number of Employees vs Case Status

- For applicants who were denied, the number of employees in the employer's company seem to vary.

- The second plot displays the employee distribution in companies where applicants were certified.
- This distribution is rightly skewed, some applications were from companies with significantly high number of employees.
- There are two Boxes of boxplots: boxplot w.r.t target and boxplot (without outliers) w.r.t target
- For the first Box of boxplot, there isn't much difference between the median. There are outliers, there are some companies that employed lots of people. This is also true for certified applicants.
- For the second Box of boxplots, the outliers have been removed.

EDA Results

Bivariate analysis: Year of Establishment vs Case Status

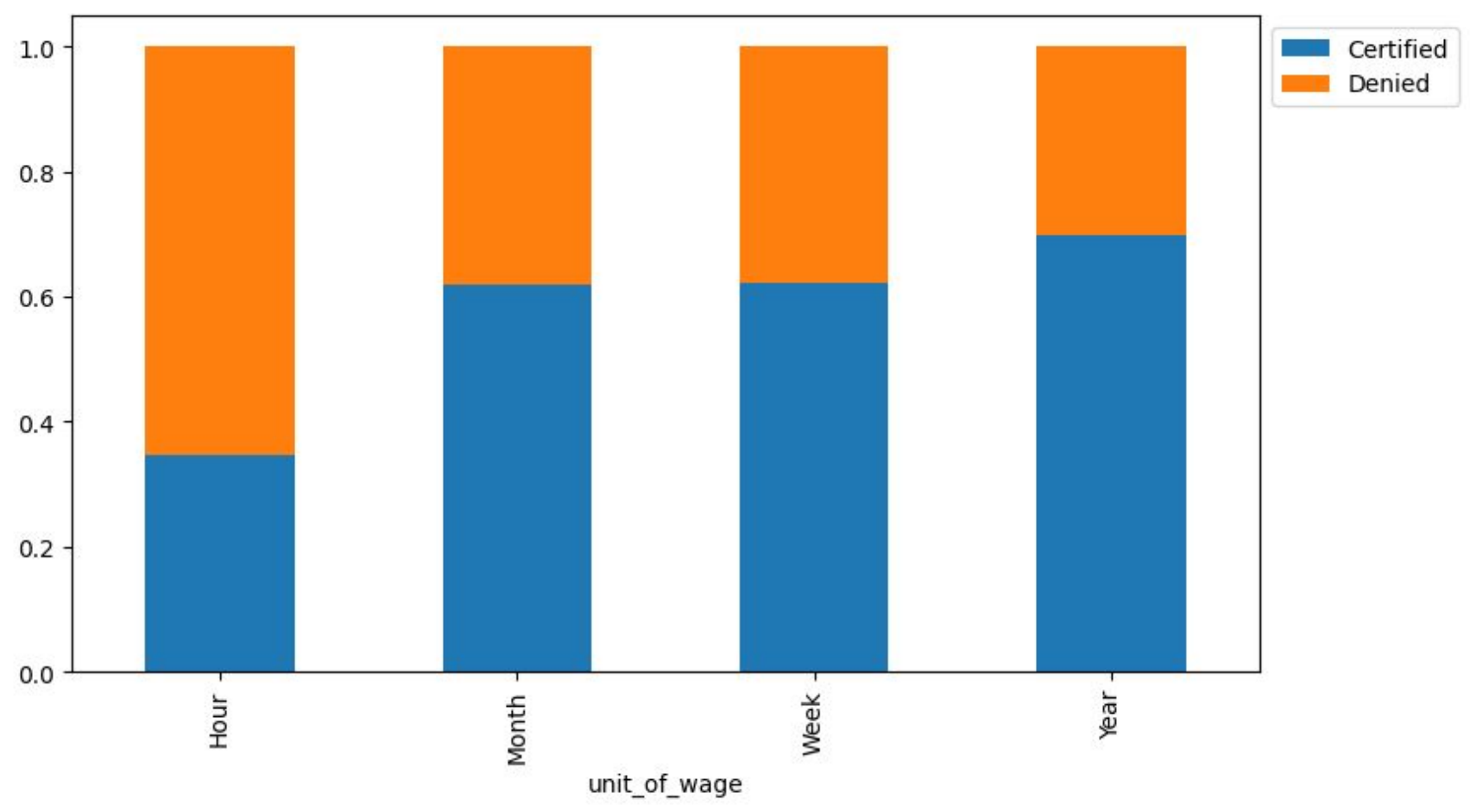


EDA Results
Bivariate analysis: Year of Establishment vs Case Status

- For applications that were eventually denied, the year of establishment of the companies appears to vary. Among these, the majority came from recently established companies.

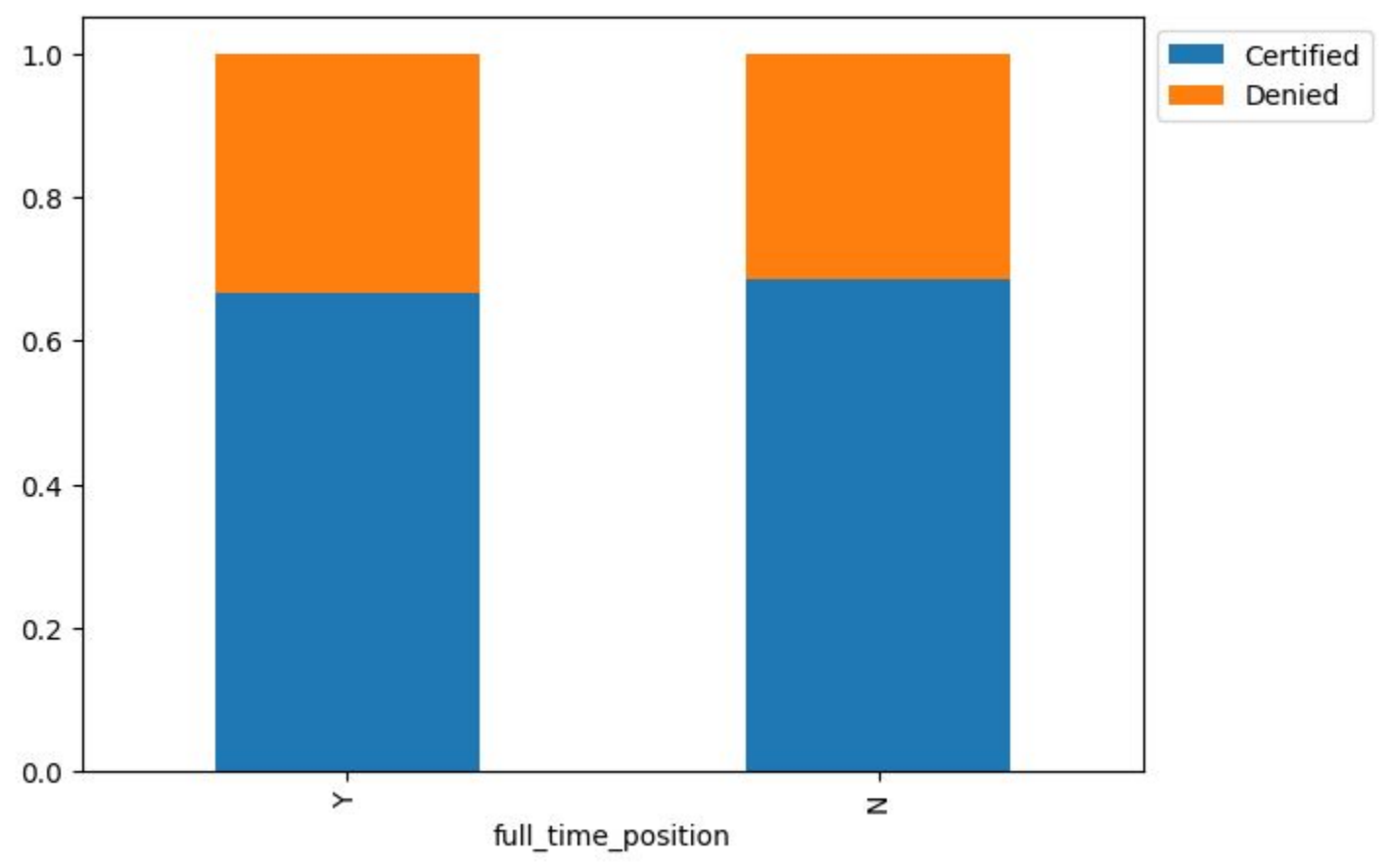
- The second plot shows the distribution of year of establishment for certified applications.
- This distribution is left skewed, some certified applications were associated with companies established as far back as year 1800. But, majority of certified applications are associated with recently established companies.
- There are two Boxes of boxplots: boxplot w.r.t target and boxplot (without outliers) w.r.t target
- For the first Box of boxplot, there isn't much difference between the median. There are outliers, there are some denied applications from old companies. This is also true for certified applications.
- For the second Box of boxplots, the outliers have been removed.

EDA Results
Bivariate analysis: Unit of Wage vs Case Status



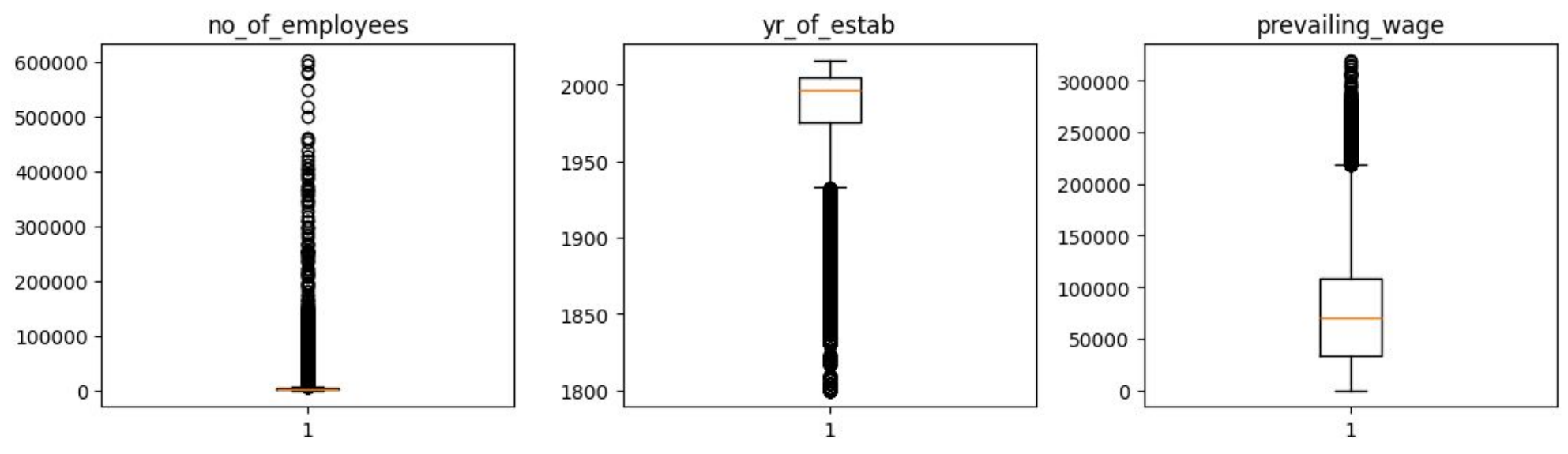
- Applicants who would be paid yearly were mostly certified.
- About 60% of applicants who would be paid weekly and monthly were certified.
- Less than 40% of hourly paid applicants got certified.
- The plot suggests that higher unit of wage could influence whether an applicant is certified.

EDA Results
Bivariate analysis: Full Time Position vs Case Status



- Applicants seeking full time positions are almost equally likely to get certified as those seeking part-time position.
- There are more applications from those seeking full time position.
- More than 60% of the applications from full-time applicants were certified. The same is true for part-time applications.

Data Preprocessing: Outlier check



- There are three boxplots in the image above.

- All the boxplots show the presence of outliers. Some companies have an unusually large number of employees. Some companies are exceptionally old. Some applications are linked to significantly high prevailing wages.

Data Preparation for Modelling

- The shape of training set is 15288 rows, 21 columns
- The shape of validation set is 5096 rows, 21 columns
- The shape of test set is 15288 rows, 21 columns
- Percentage of classes in training set: denied (0.332156), certified (0.667844)
- Percentage of classes in validation set: denied (0.332025), certified (0.667975)
- Percentage of classes in test set: denied (0.332025), certified (0.667975)

Model Building (Original Data)

We would like to minimize wrong predictions where our model predicts that the visa application will

1. get certified but in reality, the visa application should get denied.
2. not get certified but in reality, the visa application should get certified.

To do this, we find the F1 scores for each of the models

These are the F1 scores.

Training Performance:

Bagging: 0.989150179193873
Random Forest: 1.0
Decision Tree: 1.0
AdaBoost: 0.8204269947530306
Gradient Boosting: 0.8291218182658106
XGBoost: 0.8963229453814886

VValidation Performance:

Bagging: 0.7736516357206012
Random Forest: 0.8049978941457251
Decision Tree: 0.7486033519553073
AdaBoost: 0.8180081855388813
Gradient Boosting: 0.826637008202419
XGBoost: 0.8079119654547987

- *I decided to build 6 models to see how all of them actually perform.*
- On training performance, Decision Tree and Random Forest have a perfect F1 score while Bagging is close to perfect, both suggesting overfitting. Other models have moderately high F1 scores.
- On validation performance, the score for Bagging has dropped significantly when compared to training performance. Same is true for Random Forest and Decision Tree. This is clearly overfitting. AdaBoost and Gradient Boosting are doing so well, followed by XGBoost. Both models have good balance between train/validation.

Model Building (Oversampled Data)

Before Oversampling, counts of label 'Certified': 10210
Before Oversampling, counts of label 'Denied': 5078

After Oversampling, counts of label 'Certified': 10210
After Oversampling, counts of label 'Denied': 10210

After Oversampling, the shape of train_X: (20420, 21)
After Oversampling, the shape of train_y: (20420,)

Before oversampling, the dataset was imbalanced, with 10,210 certified applications and only 5,078 denied ones. This imbalance could negatively affect model performance, especially for predicting the Denied applications.

After applying SMOTE, the number of Denied cases was increased to match the Certified ones, resulting in balanced classes with 10,210 each.

The training dataset now contains 20,420 samples with 21 features, and the corresponding target labels (train_y) also reflect this balanced size.

Model Building (Oversampled Data)

Training Performance:

Bagging: 0.9875473741201949
Random Forest: 0.9999510260051913
Decision Tree: 1.0
AdaBoost: 0.800549840368925 2
Gradient Boosting: 0.8072434234901815
XGBoost: 0.8708686342053813

Validation Performance:

Bagging: 0.7665171898355755
Random Forest: 0.7965442764578834
Decision Tree: 0.7320006012325266
AdaBoost: 0.8195334879279771
Gradient Boosting: 0.8173049645390071
XGBoost: 0.8129304286718201

- On training performance, Decision Tree has a perfect F1 score, Random Forest and Bagging are close to a perfect F1 score, all these suggesting overfitting. Other models have moderately high F1 scores.
- On validation performance, the score for Decision Tree, Bagging and Random Forest have dropped significantly when compared to training performance. This is clearly overfitting. AdaBoost and Gradient Boosting are doing so well, followed by XGBoost. Both models have good balance between train/validation.

Model Building (Undersampled Data)

Before Under Sampling, counts of label 'Certified': 10210
Before Under Sampling, counts of label 'Denied': 5078

After Under Sampling, counts of label 'Certified': 5078
After Under Sampling, counts of label 'Denied': 5078

After Under Sampling, the shape of train_X: (10156, 21)
After Under Sampling, the shape of train_y: (10156,)

- Before undersampling, the dataset was imbalanced, with 10,210 certified applications and only 5,078 denied ones. This imbalance could negatively affect model performance, especially for predicting the Denied applications.

- After applying random undersampling, the number of Certified cases was decreased to match the Denied ones, resulting in balanced classes with 5078 each.

- The training dataset now contains 10,156 samples with 21 features, and the corresponding target labels (train_y) also reflect this balanced size.

Model Building (Undersampled Data)

Training Performance:

Bagging: 0.980368709516691 5
Random Forest: 1.0
Decision Tree: 1.0
AdaBoost: 0.701534304738010 3
Gradient Boosting: 0.728144171779141 1
XGBoost: 0.872035139092240 1

Validation Performance:

Bagging: 0.705704697986577 2
Random Forest: 0.741721854304635 8
Decision Tree: 0.695581809354264 4
AdaBoost: 0.765990884802766
Gradient Boosting: 0.776595744680851
XGBoost: 0.745930418129588 3

- On training performance, Decision Tree and Random Forest has a perfect F1 score while Bagging is close to perfect, both suggesting overfitting. XGBoost has moderately high F1 score which looks good while AdaBoost and Gradient Boosting have lower scores, they may benefit from hyperparameter tuning.
- On validation performance, the score for Decision Tree and Bagging have dropped significantly when compared to training performance. Same is true for Random Forest. This is clearly overfitting. AdaBoost and Gradient Boosting are doing so well, followed by XGBoost. Both models have good balance between train/validation.

Model Performance Improvement

Hyperparameter Tuning

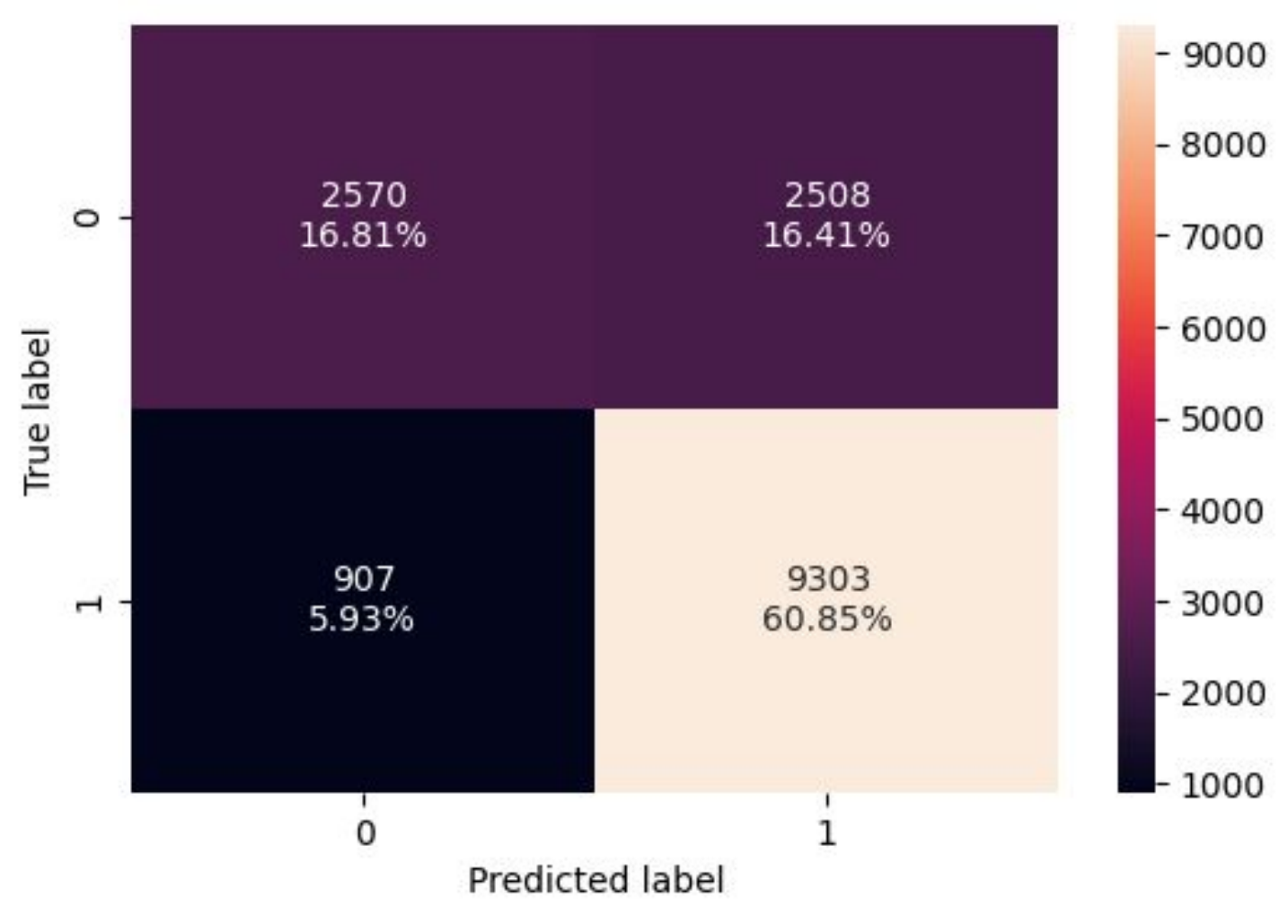
RandomForestClassifier

RandomForestClassifier(max_depth=np.int64(10), min_samples_split=5,
n_estimators=np.int64(30),
oob_score=True,
random_state=1)

This creates a moderately sized random forest (30 trees, max depth 10), with some constraints to avoid overfitting, and uses OOB samples to estimate model performance without needing a separate validation set.

Model Performance Improvement

Checking model performance on Training (Random Forest)



- The model correctly predicted 16.81% of the application as denied.
- The model correctly predicted 60.85% of the applications as certified.
- The model incorrectly predicted 16.41% of the applications as certified.
- The model incorrectly predicted 5.93% of the applications as denied.

Model Performance Improvement

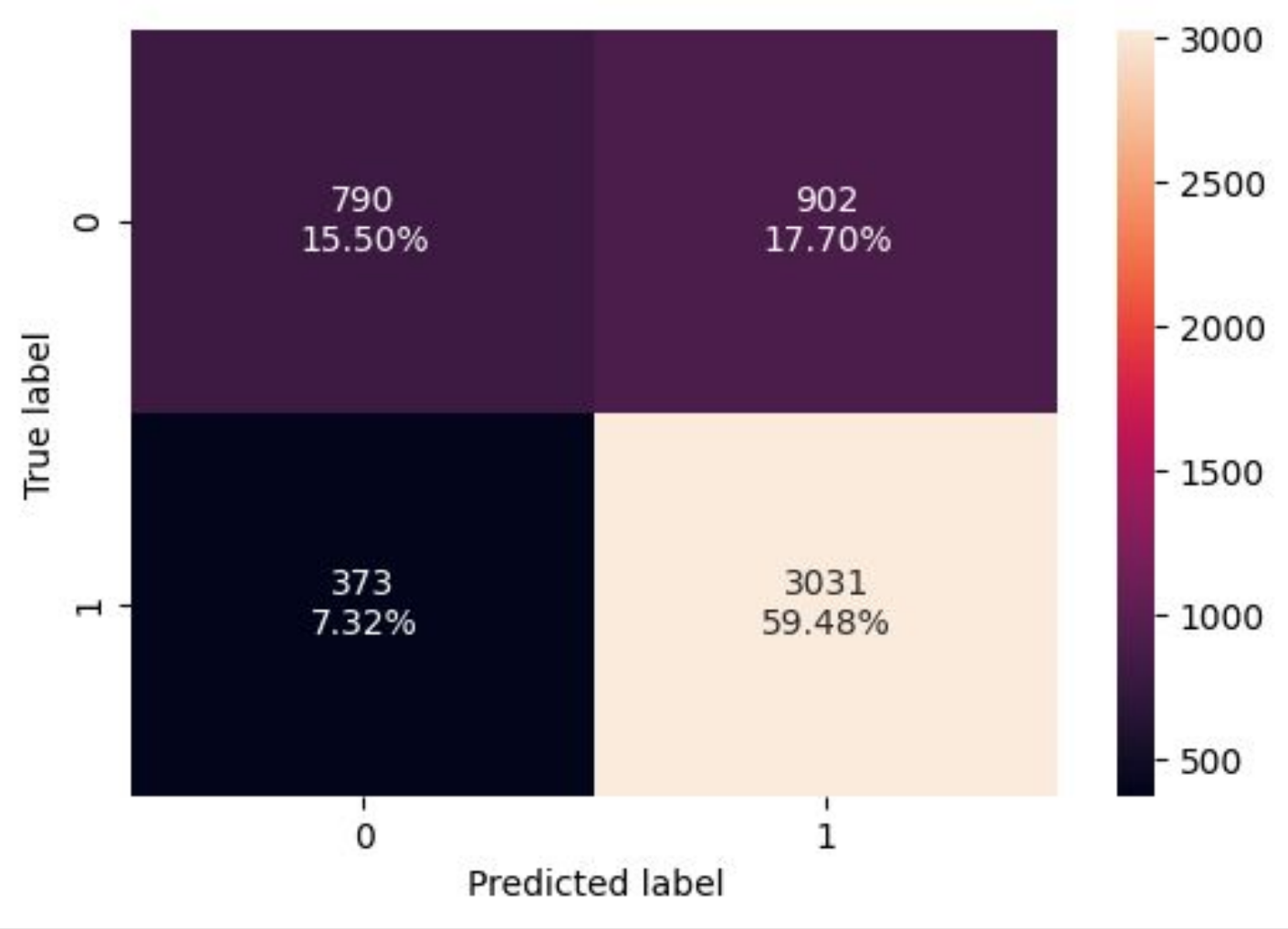
Checking model performance on Training (Random Forest)

	Accuracy	Recall	Precision	F1
0	0.776622	0.911166	0.787656	0.844921

- The model is highly sensitive to Certified cases, because it has a high recall, meaning it is good at not missing approved applications.
- The precision is lower, so it is also predicting some Denied applications as Certified (false positives).
- The F1 score (0.84) reflects a good overall balance, leaning toward recall.
- The overall accuracy (~77.7%) is decent but could be misleading since the dataset is imbalanced, which is why our F1 and recall are especially useful here.
- The model seems to over-predict approved cases, but the high F1 score suggests it is performing well overall.

Model Performance Improvement

Checking model performance on Validation (Random Forest)



- The model correctly predicted 15.50% of the application as denied.
- The model correctly predicted 59.48% of the applications as certified.
- The model incorrectly predicted 17.70% of the applications as certified.
- The model incorrectly predicted 7.32% of the applications as denied.
- This model is not yet great at predicting denied cases.

Model Performance Improvement

Checking model performance on Validation (Random Forest)

	Accuracy	Recall	Precision	F1
0	0.749804	0.890423	0.770659	0.826223

- The model does a great job at not missing Certified applications, it is very sensitive.
- Precision of 0.770659 shows the model makes a fair number of false positive predictions, i.e., some Denied cases are being misclassified as Certified.
- An F1 score of 0.826223 shows a strong balance between recall and precision, slightly lower than the training F1, which is expected and actually good (less overfitting).
- An accuracy score of 0.749804 is acceptable overall.
- On the validation set, the model generalizes well with strong recall and F1 score, meaning it is reliably identifying most Certified applications with reasonable precision. It is slightly less precise than on training data, but overall performs consistently and effectively.

Model Performance Improvement
Hyperparameter Tuning - AdaBoost Classifier

AdaBoostClassifier

AdaBoostClassifier(estimator=DecisionTreeClassifier(max_depth=3,

random_state=1),

learning_rate=np.float64(0.060000000000000005),
n_estimators=np.int64(100),
random_state=1)

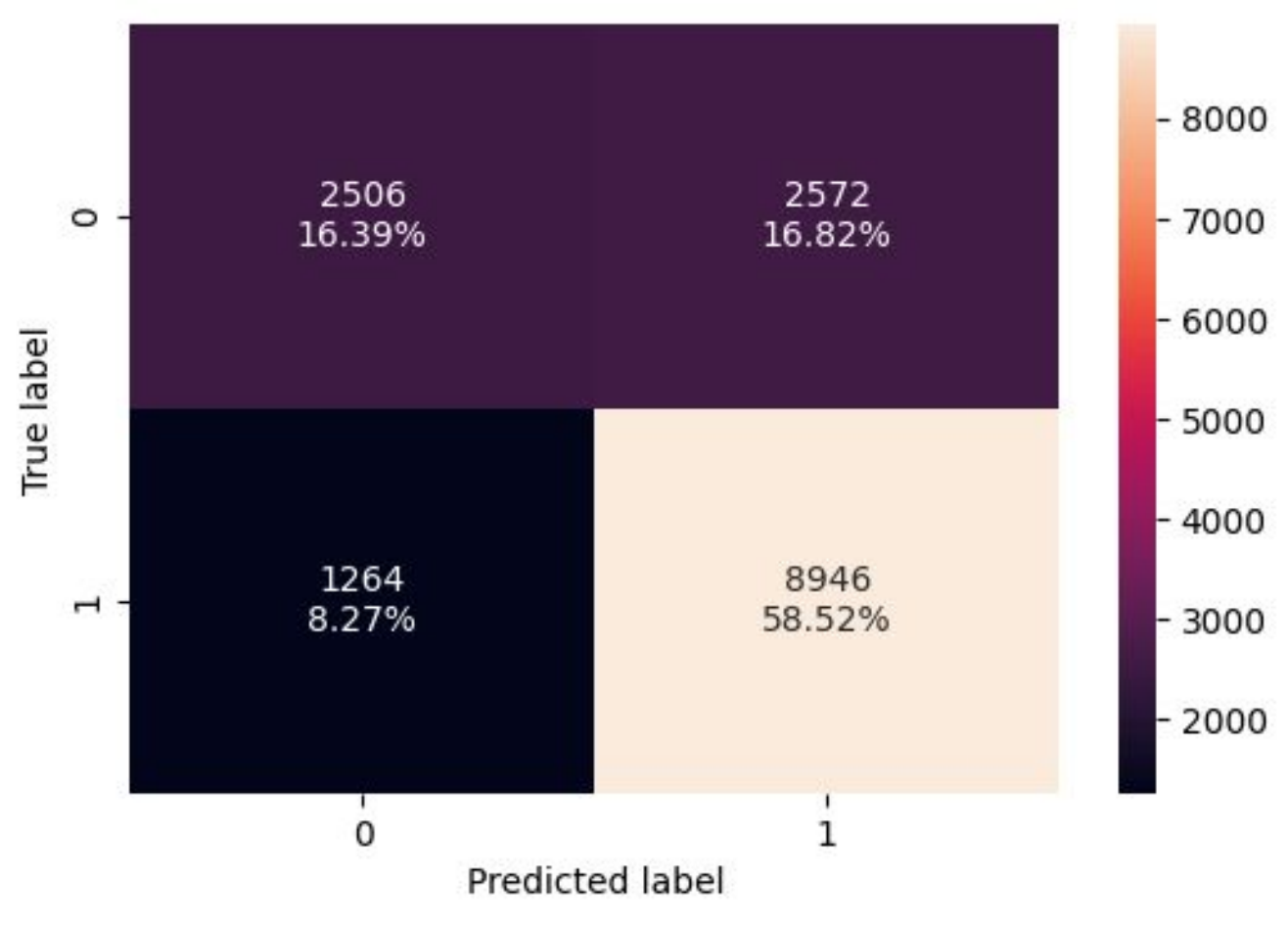
estimator: DecisionTreeClassifier
DecisionTreeClassifier(max_depth=3,
random_state=1)

- Our AdaBoost uses 100 small decision trees (depth 3), learns slowly (learning rate 0.06), and is designed to gradually build a strong model by focusing more on previously misclassified data. We are working on imbalance data.

Model Performance Improvement

Hyperparameter Tuning - AdaBoost Classifier

Checking model performance on training set



- The model correctly predicted 16.39% of the application as denied.
- The model correctly predicted 59.52% of the applications as certified.
- The model incorrectly predicted 16.82% of the applications as certified.
- The model incorrectly predicted 8.27% of the applications as denied.

Model Performance Improvement

Hyperparameter Tuning - AdaBoost Classifier

Checking model performance on training set

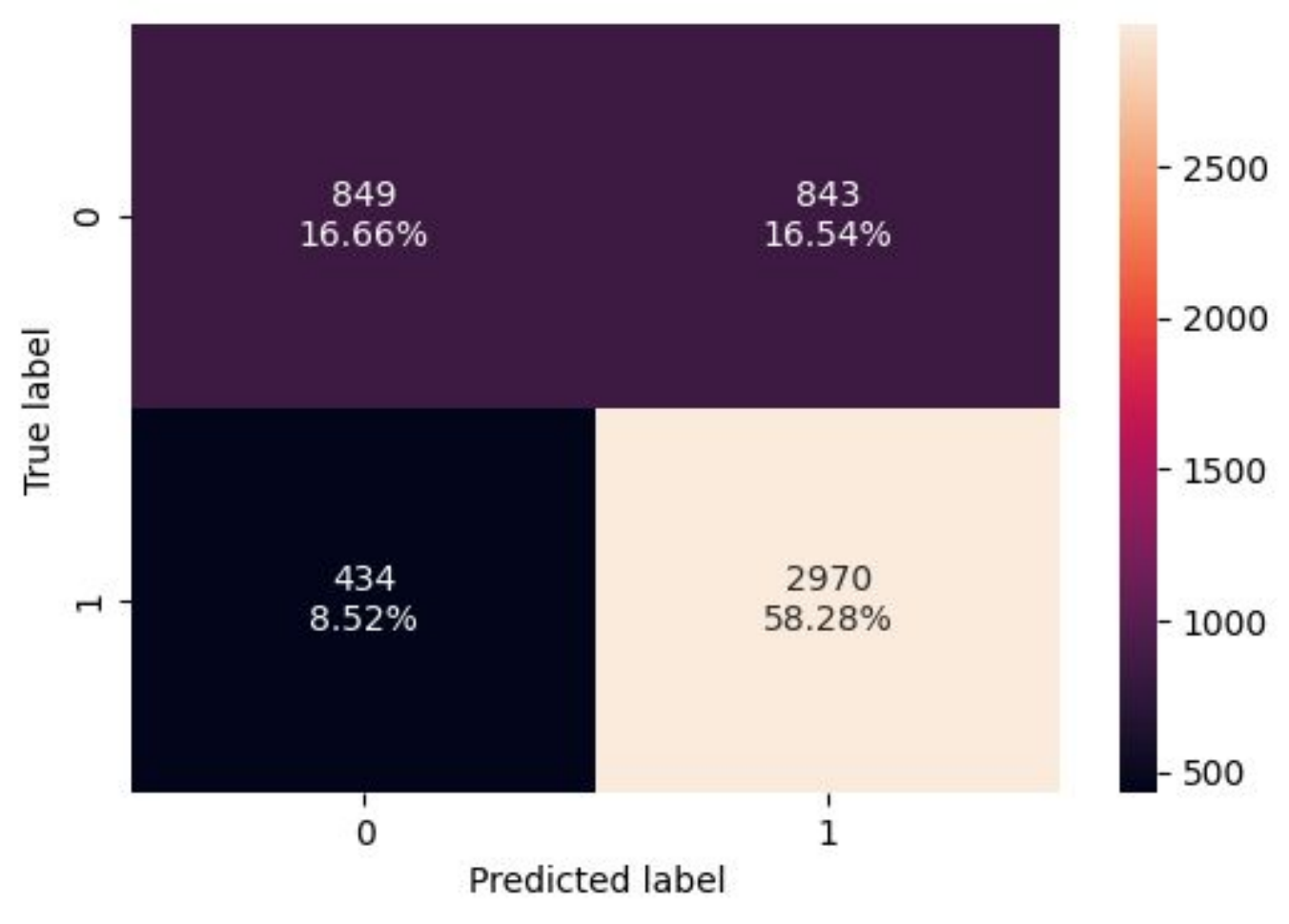
	Accuracy	Recall	Precision	F1
0	0.749084	0.8762	0.776697	0.823454

- The model does a great job at not missing Certified applications, it is very sensitive.
- Precision of 0.776697 shows the model makes a fair number of false positive predictions, i.e., some Denied cases are being misclassified as Certified.
- An F1 score of 0.823454 shows a strong balance between recall and precision, indicating the model performs well on the training set.
- An accuracy score of 0.749084 is acceptable overall.
- On the training set, the model demonstrates reliable classification of Certified applications with a solid trade-off between precision and recall, indicating that it has learned well without overfitting.

Model Performance Improvement

Hyperparameter Tuning - AdaBoost Classifier

Checking model performance on Validation set



- The model correctly predicted 16.66% of the application as denied.
- The model correctly predicted 58.28% of the applications as certified.
- The model incorrectly predicted 16.54% of the applications as certified.
- The model incorrectly predicted 8.52% of the applications as denied.
- This shows that while the model is effective at recognizing Certified applications, it still confuses a fair number of Denied cases as Certified

Model Performance Improvement

Hyperparameter Tuning - AdaBoost Classifier

Checking model performance on Validation set

	Accuracy	Recall	Precision	F1
0	0.749411	0.872503	0.778914	0.823057

- The model does a great job at not missing Certified applications, it is very sensitive.
- Precision of 0.778914 shows the model makes a fair number of false positive predictions, i.e., some Denied cases are being misclassified as Certified.
- An F1 score of 0.823057 shows a strong balance between recall and precision, slightly lower than the training F1, which is expected and actually good (less overfitting).
- An accuracy score of 0.749411 is acceptable overall.
- On the validation set, the model generalizes well with strong recall and F1 score, meaning it is reliably identifying most Certified applications with reasonable precision. It is slightly less precise than on training data, but overall performs consistently and effectively.

Model Performance Improvement
Hyperparameter Tuning - Gradient Boosting Classifier

GradientBoostingClassifier

```
GradientBoostingClassifier(init=AdaBoost
Classifier(random_state=1),
                           learning_rate=0.05,
max_features=0.7,

n_estimators=np.int64(50),
random_state=1,
                           subsample=0.7)
```

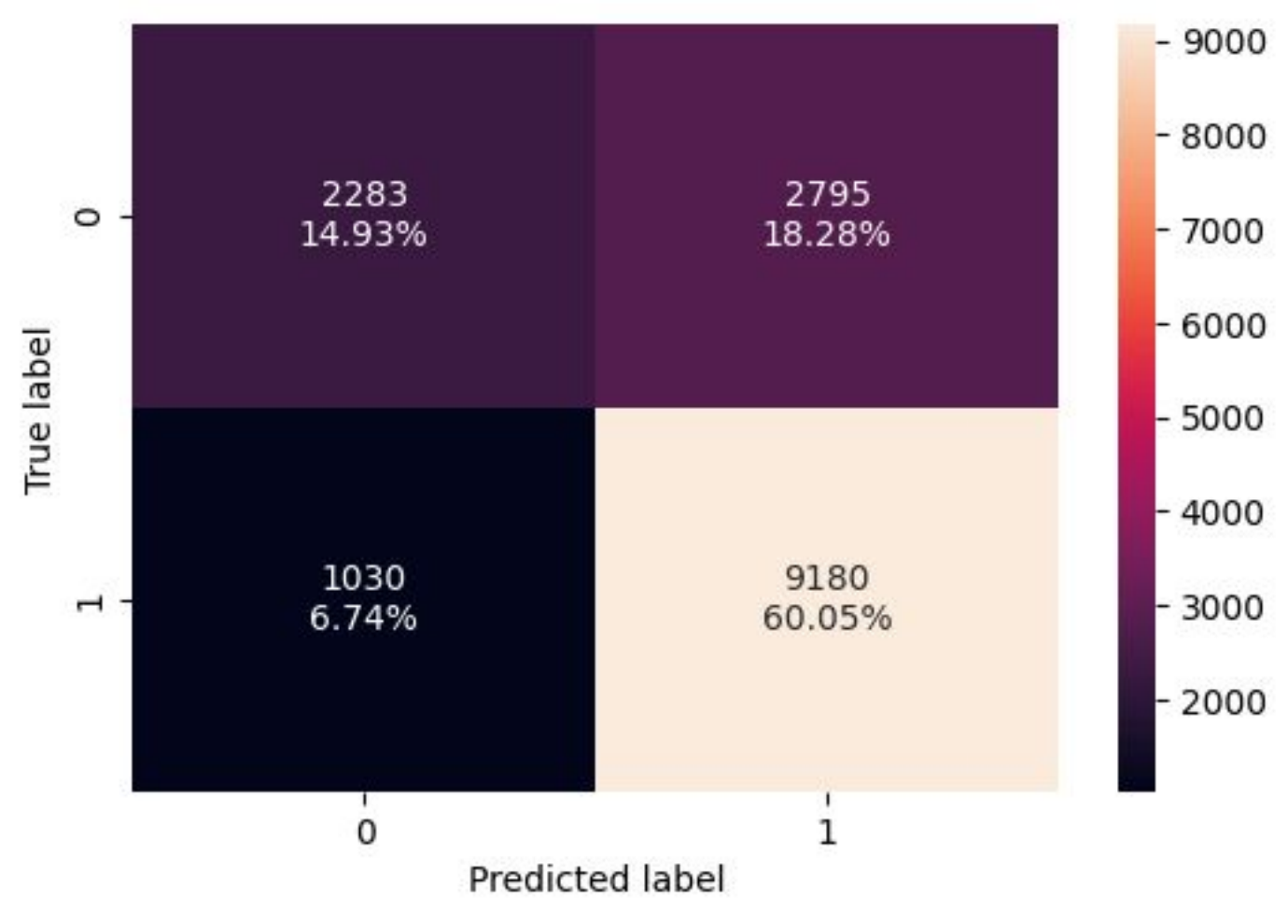
init: AdaBoostClassifier
AdaBoostClassifier(random_state=1)

Our GradientBoostingClassifier is initialized with an AdaBoost model, with the hope that this will make the model more powerful and not overfit.

Model Performance Improvement

Hyperparameter Tuning - Gradient Boosting Classifier

Checking model performance on training set



- The model correctly predicted 14.93% of the application as denied.
- The model correctly predicted 60.05% of the applications as certified.
- The model incorrectly predicted 18.28% of the applications as certified.
- The model incorrectly predicted 6.74% of the applications as denied.
- The Gradient Boosting model performs well in identifying Certified cases but indicates overconfidence in approvals. This could be risky in sensitive applications like visa decisions and may require further tuning.

Model Performance Improvement

Hyperparameter Tuning - Gradient Boosting Classifier

Checking model performance on training set

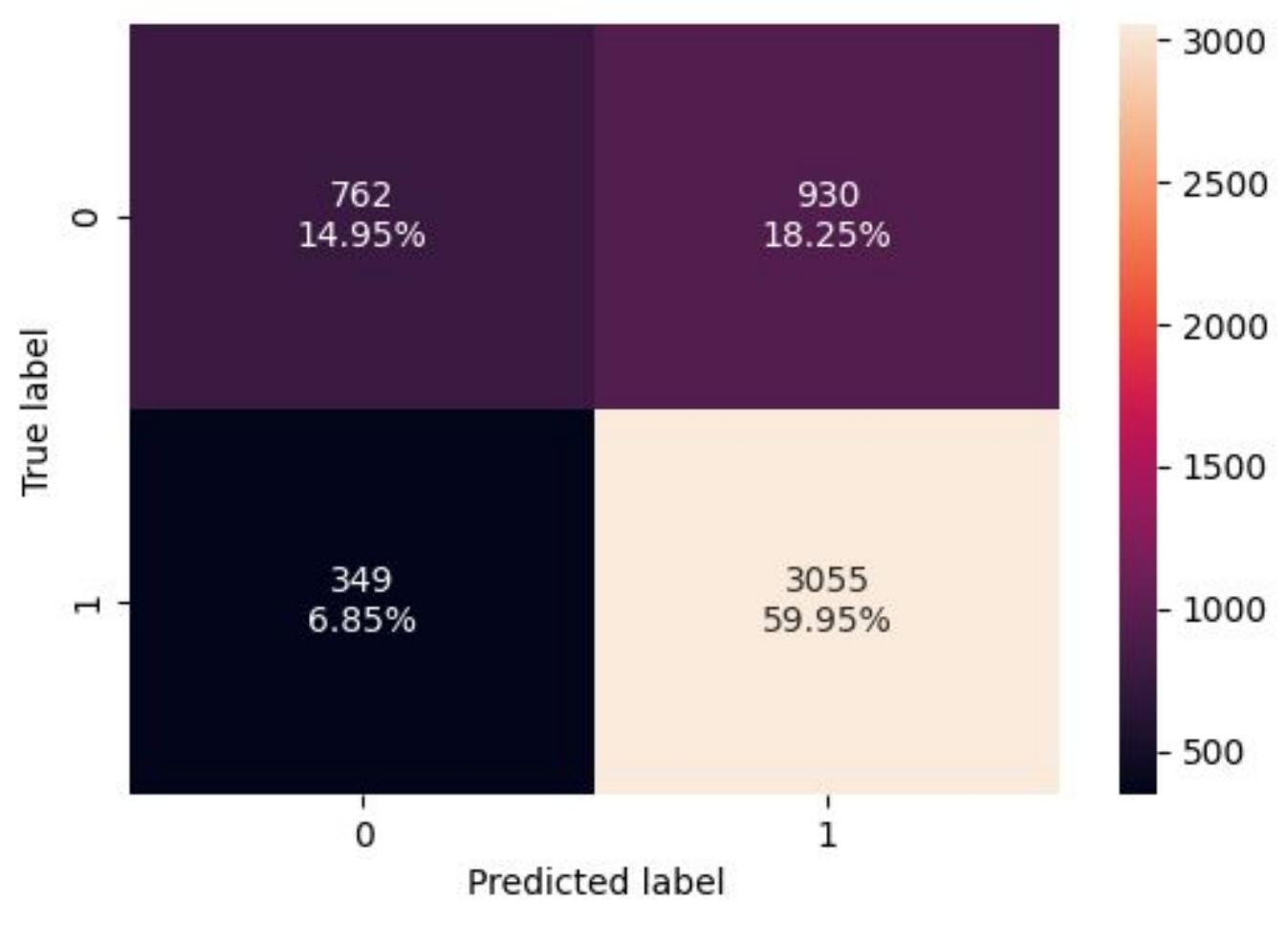
	Accuracy	Recall	Precision	F1
0	0.749804	0.899119	0.766597	0.827586

- The model does a great job at not missing certified applications, it is very sensitive.
- Precision of 0.766597 shows the model makes a fair number of false positive predictions, i.e., some denied cases are being misclassified as certified.
- An F1 score of 0.827586 shows a strong balance between recall and precision, indicating solid learning on the training data.
- An accuracy score of 0.749804 is acceptable overall.
- On the training set, the model effectively identifies most Certified applications while maintaining a good trade-off between recall and precision.

Model Performance Improvement

Hyperparameter Tuning - Gradient Boosting Classifier

Checking model performance on Validation set



- The model correctly predicted 14.95% of the application as denied.
- The model correctly predicted 59.95% of the applications as certified.
- The model incorrectly predicted 18.25% of the applications as certified.
- The model incorrectly predicted 6.85% of the applications as denied.

Model Performance Improvement

Hyperparameter Tuning - Gradient Boosting Classifier

Checking model performance on Validation set

	Accuracy	Recall	Precision	F1
0	0.749019	0.897474	0.766625	0.826905

- The Gradient Boosting model shows excellent generalization from training to validation.
- The differences (from training to validation) in all scores/metrics (accuracy, recall, precision, F1) are negligible
- It seems Gradient Boosting is well-tuned and robust, and could be a reliable choice for predicting visa certification outcomes without signs of overfitting or underfitting.

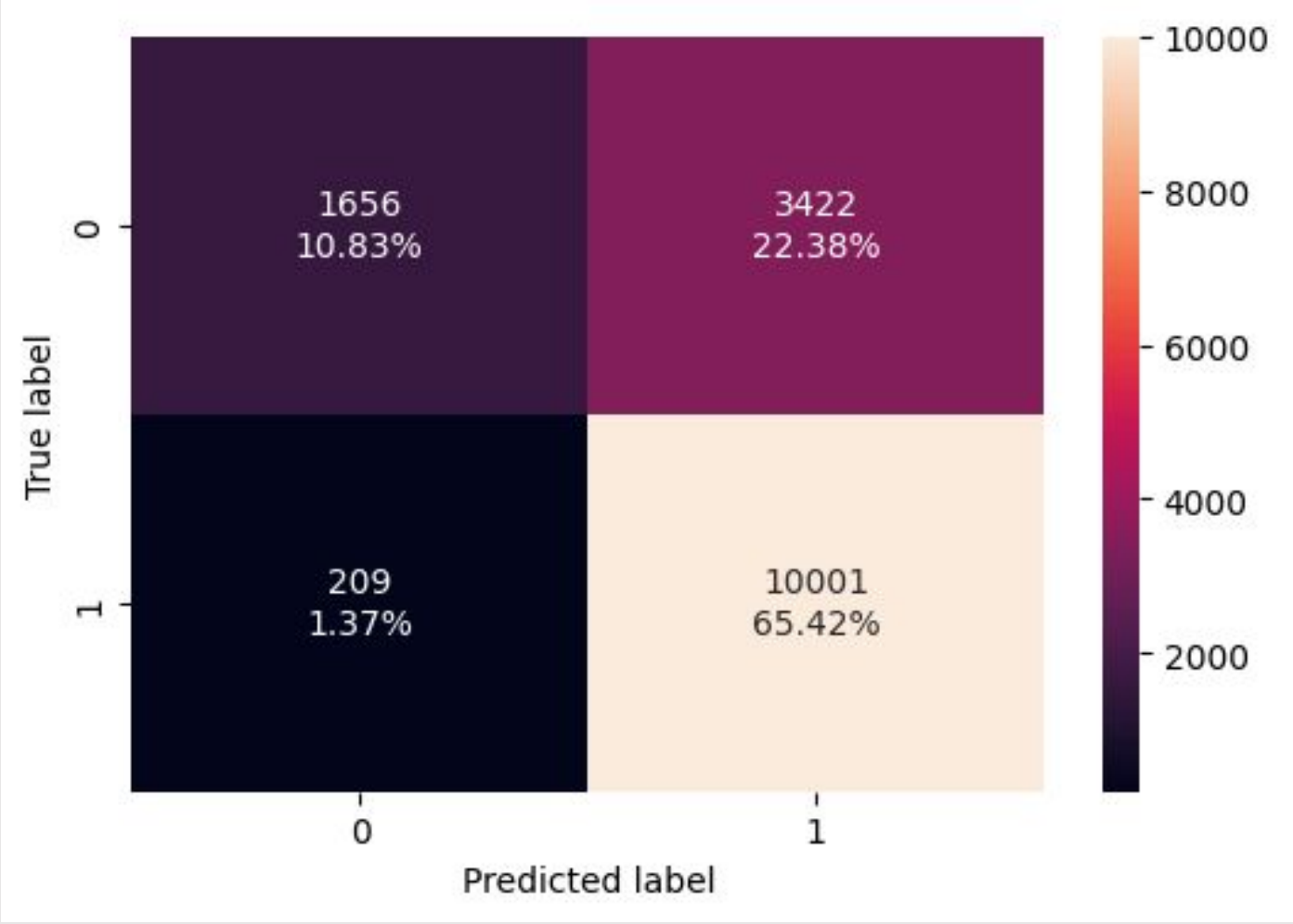
Model Performance Improvement
Hyperparameter Tuning - XGBoost Classifier

XGBClassifier

```
XGBClassifier(base_score=None, booster=None, callbacks=None,
              colsample_bylevel=None, colsample_bynode=None,
              colsample_bytree=None, device=None,
early_stopping_rounds=None,
              enable_categorical=False, eval_metric='logloss',
              feature_types=None, feature_weights=None, gamma=1,
              grow_policy=None, importance_type=None,
              interaction_constraints=None, learning_rate=0.1,
max_bin=None,
              max_cat_threshold=None, max_cat_to_onehot=None,
              max_delta_step=None, max_depth=None,
max_leaves=None,
              min_child_weight=None, missing=nan,
monotone_constraints=None,
              multi_strategy=None, n_estimators=np.int64(100),
n_jobs=None,
              num_parallel_tree=None, ...)
```

- Our XGBoost is moderately tuned.
- It learns at a standard pace, and we are using log loss as its evaluation metric.
- It has some overfitting protection built in via gamma.

Model Performance Improvement
Hyperparameter Tuning - XGBoost Classifier
Checking model performance on training set



- The model correctly predicted 10.83% of the application as denied.
- The model correctly predicted 65.42% of the applications as certified.
- The model incorrectly predicted 22.38% of the applications as certified.
- The model incorrectly predicted 1.37% of the applications as denied.
- The model confidently identifies Certified applications. However, it misclassifies many Denied cases as Certified, indicating a bias toward the Certified class due to class imbalance.

Model Performance Improvement

Hyperparameter Tuning - XGBoost Classifier

Checking model performance on training set

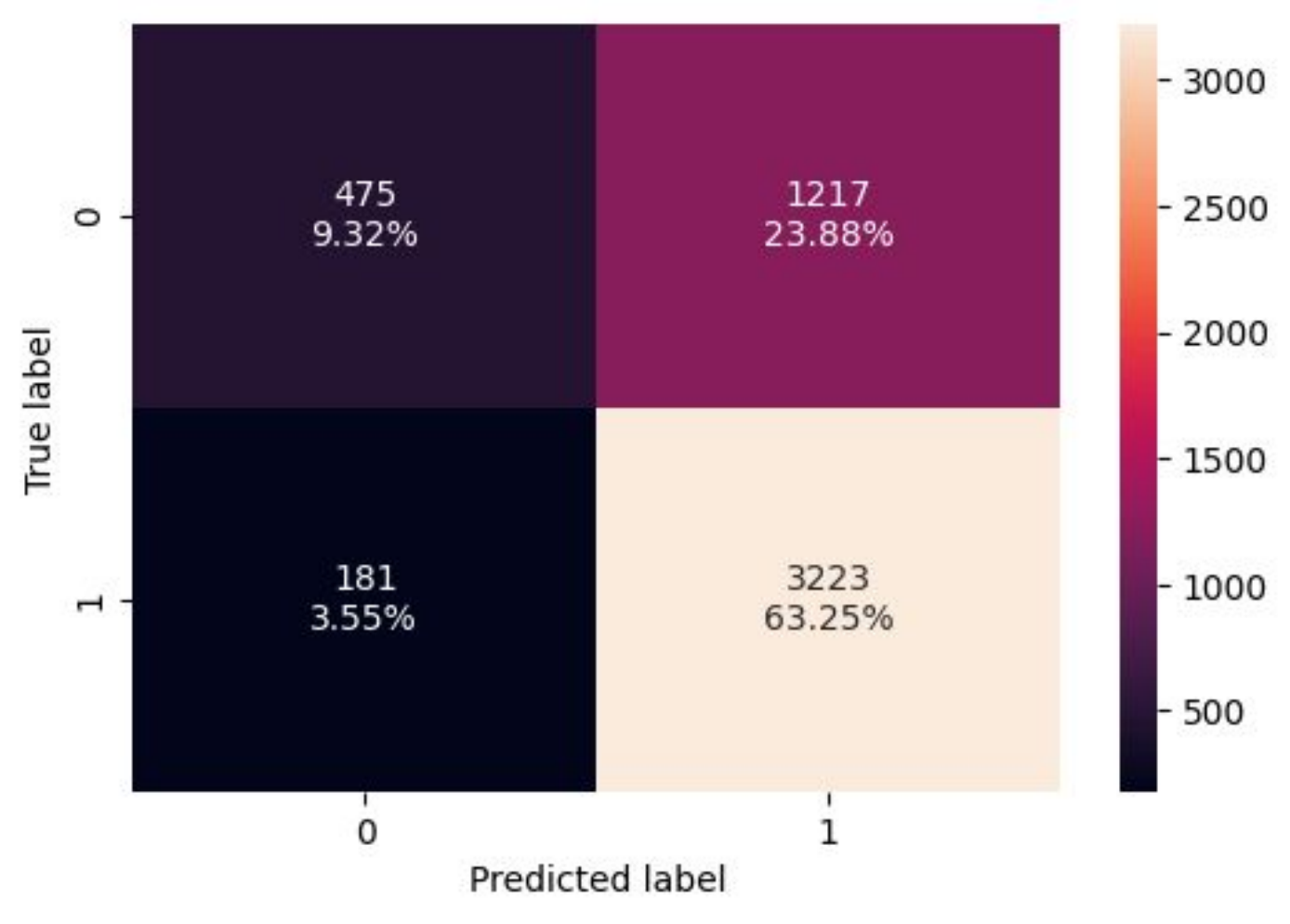
	Accuracy	Recall	Precision	F1
0	0.762493	0.97953	0.745064	0.846359

- The XGBoost model shows high recall on the training set, with a strong F1 score and good accuracy.
- Its lower precision suggests it could overpredicts certified cases, indicating a potential bias toward the positive class.
- We will compare these scores with the validation to see if we are overfitting.

Model Performance Improvement

Hyperparameter Tuning - XGBoost Classifier

Checking model performance on Validation set



- The model correctly predicted 9.32% of the application as denied.
- The model correctly predicted 63.25% of the applications as certified.
- The model incorrectly predicted 23.88% of the applications as certified.
- The model incorrectly predicted 3.55% of the applications as denied.

Model Performance Improvement

Hyperparameter Tuning - XGBoost Classifier

Checking model performance on Validation set

	Accuracy	Recall	Precision	F1
0	0.725667	0.946827	0.725901	0.821775

- All four metrics decrease slightly from training to validation, which is okay. The model still maintains strong recall and F1 score.
- The drop in accuracy and recall suggests mild overfitting. The model performs slightly better on training data, likely capturing some specific patterns.
- It seems that the model would be able to generalize well.
- High recall and slightly lower precision on both training and validation sets suggest the model is still slightly biased toward predicting certified.

Model Comparison and Final Model Selection

Comparing all models on Training set

Training performance comparison:

	TRF	TAC	TGBC	XGBCT
Accuracy	0.776622	0.749084	0.749804	0.762493
Recall	0.911166	0.876200	0.899119	0.979530
Precision	0.787656	0.776697	0.766597	0.745064
F1	0.844921	0.823454	0.827586	0.846359

Tuned Random Forest= TRF
Tuned Adaboost Classifier=TAC
Tuned Gradient Boost Classifier=TGBC
XGBoost Classifier Tuned=XGBCT

Model Comparison and Final Model Selection

Comparing all models on Training set

- XGBoost is slightly higher than Random Forest on F1 score, making it the best overall performer in terms of F1 score.
- XGBoost has the highest recall, capturing nearly all Certified applications.
- Random Forest has the best precision, meaning it is more cautious and produces fewer false positives than the others, this is ideal when false certifications are risky.
- Random Forest shows a strong balance across all metrics (good recall, highest precision, and great F1 score), making it a reliable.
- Despite top scores, XGBoost's lower precision compared to its recall suggests it may be bias towards predicting Certified cases.
- Depending what is important, we might later choose XGBoost when recall is critical, or Random Forest when precision and balanced performance in all metrics are preferred.
- Let us see how the models compare with each other on validation set. This is shown on the next slide.

Model Comparison and Final Model Selection

Comparing all models on Validation set

Training performance comparison:

	TRF	TAC	TGBC	XGBCT
Accuracy	0.749804	0.749411	0.749019	0.725667
Recall	0.890423	0.872503	0.897474	0.946827
Precision	0.770659	0.778914	0.766625	0.725901
F1	0.826223	0.823057	0.826905	0.821775

Tuned Random Forest= TRF
Tuned Adaboost Classifier=TAC
Tuned Gradient Boost Classifier=TGBC
XGBoost Classifier Tuned=XGBCT

Model Comparison and Final Model Selection

Comparing all models on Validation set

- Gradient Boosting slightly outperforms the others on F1 score, making it the best overall performer on the validation set in terms of F1 score.
- XGBoost again leads in recall, meaning it captures almost all Certified applications. This makes it ideal when the cost of missing a Certified case is high.
- AdaBoost has the highest precision, suggesting it is more valuable when false certifications carry risks.
- While XGBoost maintains excellent recall, its lower precision compared to the others again indicates a bias toward predicting Certified.
- Depending on what is more important, maximizing approvals (recall) or minimizing errors (precision), XGBoost and AdaBoost remain top contenders.
- Gradient Boosting stands out here for its best F1 performance, signaling robust generalization.
- XGBoost and Random Forest might be slightly overfitting as they have the highest drop in accuracy scores from training to validation.
- Gradient Boosting also has the least drop in almost all the metrics, thus might be the best for generalization.

To prevent data leakage and ensure fair evaluation, the dataset is split into training, validation, and test sets.

The training set is used to fit the model, the validation set is used for hyperparameter tuning and model selection, and the test set is held back until the end to simulate real-world data and assess the final model's performance. The test set must not influence training or tuning.

Model Comparison and Final Model Selection

Comparing all models on TEST set

Training performance comparison:

	TRF	TAC	TGBC	XGBCT
Accuracy	0.749804	0.749411	0.749019	0.725667
Recall	0.890423	0.872503	0.897474	0.946827
Precision	0.770659	0.778914	0.766625	0.725901
F1	0.826223	0.823057	0.826905	0.821775

Tuned Random Forest= TRF
Tuned Adaboost Classifier=TAC
Tuned Gradient Boost Classifier=TGBC
XGBoost Classifier Tuned=XGBCT

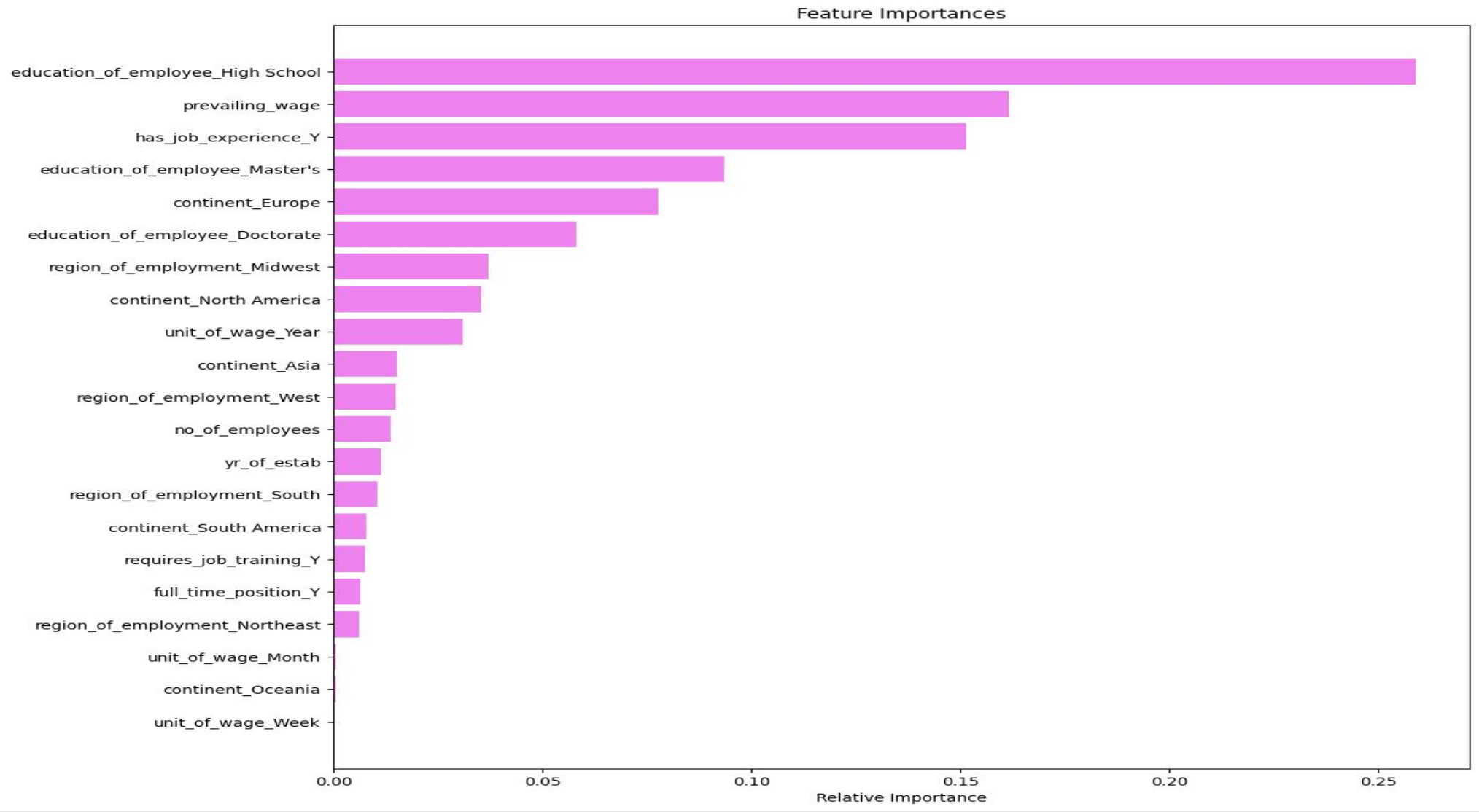
Model Comparison and Final Model Selection

Comparing all models on TEST set

- If recall is most important then we can choose XGBoost.
- We can choose AdaBoost if precision is most important, that is when false certifications are costly.
- We want to be able to correctly predict certified application with balanced performance on all the metrics, so Tuned Gradient Boosting is the best performing model on F1 scores with balanced predictions.

Model Comparison and Final Model Selection

Important features of the final model



Model Comparison and Final Model Selection
Important features of the final model

Considering the feature importance of the selected model (Gradient Boost), the following are the five most important for determining whether a visa application is certified:

- Having a high school certificate
- Prevailing wage
- Having a job experience
- Having a master’s degree
- Being a European

Actionable Insights and Key Take away

- Education of employees: prioritizing educated visa applicants can drive economic growth, fill skill gaps, and foster innovation, contributing to the country’s ability to sustain long-term development and global competitiveness.
- Prevailing wage: ensuring visa applicants meet the prevailing wage standards helps protect local labor markets by preventing underpayment. It ensures fair compensation for foreign workers, promotes workforce equity, and maintains competitive wage levels, benefiting both the economy and local workers.
- Job experience: visas should be granted to applicants with job experience because they can adapt quickly, fill skill gaps, boost productivity, reduce hiring risks, and enhance global competitiveness.
- Continet_europe: The applicant's continent should not be a key factor in visa approval, although our analysis shows otherwise. Skills, experience, and education should take precedence, ensuring a diverse, talented workforce that drives innovation and economic growth.