|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Variables | Description | Domain Type | Entries | Missing Values |
| 1 | RowNumber | The row numbers of each entry | Ratio | 10000 | 0 |
| 2 | CustomerId | An identifier for each customer | Ratio | 10000 | 0 |
| 3 | Surname | The surname of each customer | Nominal | 10000 | 0 |
| 4 | CreditScore | Measure of individual creditworthiness, derived from financial history | Ratio | 10000 | 0 |
| 5 | Geography | Location of customer | Nominal | 10000 | 0 |
| 6 | Gender | Sex of each customer (Male or Female) | Nominal | 10000 | 0 |
| 7 | Age | Each customer’s age | Ratio | 10000 | 0 |
| 8 | Tenure | Duration of customer with bank | Ratio | 10000 | 0 |
| 9 | Balance | Current account balance of customer | Ratio | 10000 | 0 |
| 10 | NumOfProducts | Customer purchases made with the bank | Ratio | 10000 | 0 |
| 11 | HasCrCard | Customer has a credit card in his name | Ratio | 10000 | 0 |
| 12 | IsActiveMember | Customer is an active member of the bank | Ratio | 10000 | 0 |
| 13 | EstimatedSalary | The customers estimated salary | Ratio | 10000 | 0 |
| 14 | Exited | Customer churns or not | Ratio | 10000 | 0 |
| 15 | Complain | Customer makes a complaint or not | Ratio | 10000 | 0 |
| 16 | Satisfaction Score | Satisfaction of customer with the bank’s services | Ratio | 10000 | 0 |
| 17 | Card Type | Type of credit card customer utilizes | Ordinal | 10000 | 0 |
| 18 | Points Earned | Points earned by the customer for using credit card | Ratio | 10000 | 0 |
|  | TOTAL |  |  |  | 0 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Check | Findings | Impact on Solutions |
| 1 | Accuracy and Precision | No issues detected | None |
| 2 | Correctness by Entry | a) We observe some strange values for people’s surnames. Values such as H?, L?, Hs?, Y?, Y?an, K?, and Hs?eh, do not represent real-world names.  b) Estimated salary reports some customers earning less than 10,000 to as low as 11.58. This is strange as many of these reported customers have a bank balance above 100,000. | a) These issues question how reliable the rows with these values are towards our model’s prediction.  b) These errors question how reliable our trained model will be. Setting a threshold below which we consider unreasonable could be one possible solution for this problem. |
| 3 | Completeness | No issues detected | None |
| 4 | Consistency (validity and integrity) | No issues detected | None |
| 5 | Data Source Reliability | While the dataset is gotten from Kaggle, the authenticity of the bank where this dataset is gotten from is not specified. | Difficulty relying on results and findings from dataset as a case for modelling a real-world phenomenon. |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Variables | Mean | Std | Min | Median | Max |
| 1 | CreditScore | 650.53 | 96.65 | 350 | 652 | 850 |
| 2 | Age | 35.92 | 10.49 | 18 | 37 | 92 |
| 3 | Tenure | 5.013 | 2.89 | 0 | 5 | 10 |
| 4 | Balance | 76485.89 | 62397.41 | 0 | 97198.54 | 250898.09 |
| 5 | NumOfProducts | 1.53 | 0.58 | 1 | 1 | 4 |
| 6 | HasCrCard | 0.71 | 0.46 | 0 | 1 | 1 |
| 7 | IsActiveMember | 0.52 | 0.5 | 0 | 1 | 1 |
| 8 | EstimatedSalary | 100090.24 | 57510.49 | 11.58 | 100193.92 | 199992.48 |
| 9 | Exited | 0.2 | 0.4 | 0 | 0 | 1 |
| 10 | Complain | 0.2 | 0.4 | 0 | 0 | 1 |
| 11 | Satisfaction Score | 3.014 | 1.41 | 1 | 3 | 5 |
| 12 | Points Earned | 606.515 | 225.93 | 119 | 605 | 1000 |

Figure 3.1: Card Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Diamond | Platinum | Gold | Silver |
| COUNT | 2507 | 2495 | 2502 | 2496 |

Figure 3.2: Geography

|  |  |  |  |
| --- | --- | --- | --- |
|  | France | Spain | Germany |
| COUNT | 5014 | 2477 | 2509 |

Figure 3.2: Gender

|  |  |  |
| --- | --- | --- |
|  | Female | Male |
| COUNT | 4543 | 5457 |

Figure 3.1: Tenure

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| COUNT | 413 | 1035 | 1048 | 1009 | 989 | 1012 | 967 | 1028 | 1025 | 984 | 490 |

Figure 3.2: Satisfaction Score

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 |
| COUNT | 1932 | 2014 | 2042 | 2008 | 2004 |

Figure 3.2: NumOfProducts

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 |
| COUNT | 5084 | 4590 | 266 | 60 |

Figure 3.2: IsActiveMember

|  |  |  |
| --- | --- | --- |
|  | 1 | 0 |
| COUNT | 5151 | 4849 |

Figure 3.1: HasCrCard

|  |  |  |
| --- | --- | --- |
|  | 1 | 0 |
| COUNT | 7055 | 2945 |

Figure 3.2: Exited

|  |  |  |
| --- | --- | --- |
|  | 1 | 0 |
| COUNT | 2038 | 7962 |

Figure 3.2: Complain

|  |  |  |
| --- | --- | --- |
|  | 1 | 0 |
| COUNT | 2044 | 7956 |

# Some potential things to consider

"""

1) What is the impact of our analysis from a specific geographical location?

i.e Does having an imbalanced geographical class impact our analysis

2) Can we create new features that capture the relationship between customers

and churn.

3) Create data binning to help with skewed distributions.

4) When binning the Card Type, it's worth considering if the column is truly Nominal

or should be represented as Ordinal data type. This should be considered, as within

the bank, there is some level of inherent hierarchy among the card type a user holds

and as such, our transformation of the card type column needs to reflect this.

"""

# EDA Findings

"""

After exploratory data analysis was conducted, we make the following findings:

- Without needing to preprocess the data, we highlight 3 features which won't be relevant

for our analysis and drop them. These features include - "RowNumber", "CustomerId", "Surname".

\* Row number here is just an identifier and will not play any role in contributing to the

understanding of why bank customers leave the bank.

\* CustomerID similar to the row number is an identifier and will not contribute to our models

predictive capacity.

\* Surname refers to the lastname of our customers. This doesn't add value to our prediction

as we cannot say in any literature that given a person's surname, we can tell if he or she

will leave the bank. The idea of churn has nothing to do with the customer’s surname. Hence,

we drop this column.

- For our analysis on bank customer churn, our dataset makes use of 5457 males and 4543 females.

- Our data has no duplicate values across each row

- No missing values were found

- The bank customers ages range from 18 to 92

- The data collected spans 3 geographical locations - France, Germany, and Spain.

With the majority of the customers in our analysis coming from France. France has

5014 customers analysed, Germany has 2509 customers, while Spain has 2477 customers.

- Customers within the bank have 4 different possible card types they can hold. These

include - Diamond cards, Gold cards, Silver cards, and Platinum cards.

- When looking at the value counts for our customers age, it is seen that the larger

majority of customers in our dataset are in their 30's.

- Interestingly, we see a 0.996 strong positive correlation between the complain column

and the exited column. This strong linear relationship between the complain feature and

our target exited, suggests that we can use this singular complain column to predict

customer churn in banks. The impact of this feature on our analysis needs to be explored

further.

- No strong linear correlation among other variables outside of the exited and complain

features.

- The Age feature has a left-skewed distribution as seen in our histogram. While

creditscore is slightly skewed to the right. We observe a uniform distribution among the

following features - EstimatedSalary, IsActiveMember, and SatisfactionScore. All other

features have an unbalanced class distribution.

"""

# Data Quality Checks

"""

For our data quality checks, we consider the following checks:

1) Accuracy and Precision Check:

e.g. currency exchange between £ and ¥: £1 ↔ ¥9.10 or £1 ↔ ¥9.056187

2) Correctness by Entry Check:

e.g. entered 9.056287 instead of 9.056187

3) Completeness Check:

e.g. no date of birth is given

4) Consistency (validity and integrity) Check:

e.g. book borrowing date: 02/02/2019, date of return: 03/01/2019

5) Redundancy (unnecessary redundancy) Check:

e.g. simply collecting together data on replication servers

6) Data Source Reliability Checks

"""

"""

HOW IS IT POSSIBLE THAT PEOPLE HAVE ESTIMATED SALARY LESS THAN BALANCE

"""

"""

Do column to column visualization of categorical variables

"""

"""

REMEMBEER YOU DID NORMAL CORRELATION BETWEEN VARIABLES. FOR BINARY TARGET, YOU NEED TO USE

POINT-BISERIAL CORRELATION

"""

# Drop the COMPLAIN COLUMN

"""

During our first iteration in our model building phase, the built model is able to predict accuratly

100% for test data across 5 different algorithms. While at first glance, it may seem

like the best way to go, however, when things are looking too good to be true, it is probably

too good to be true. This means when we have successfully created

a model that predicts the likelihood a bank customer with am accuracy of 100%, the results should be

considered too good to be true and due to chance as predicting accurately a 100% accuracy means

this model never fails.

The possible reason for this behaviour in our model could be the relationship between the complain featrure

and the target exited. After removing outliers using the mahalanobis multivariate outlier technique, which

was able to capture two groups of people between the complain feature and exited. These groups are:

- Those who made complaints but didn't churn

- Those who churned but didn't complain

Upon removing these rows highlighted by our mahalanobis algorithm, correlation between the complain

feature and exited is 1 indicating a perfect positive correlation between the variables. This relationship

therefore crates the illusion that complain and exited are the same. Theoritically, this is misleading while

trying to model the real-world as there exists no perfect correlation between these two variables. This proxy

variable complain is the feature our algorithm captures and uses as the most important feature to learn the

realationship between bank customers and why they churn.

The approach taken to mitigate this was to drop the complain column our dataset and build our predictive model

around the other features.

"""  
  
FOR FEATURE ENGINEERING, CONSIDER CREATING A NEW FEATURE THAT HELPS SOLVE THE ISSUE OF ZEROS. IT CAN BE A BINARY FEATURE THAT WAKENS WHEN THE ACCOUNT BALANCE IS ZERO AND SWITCHES OFF WHEN NOT.

|  |  |
| --- | --- |
| Variables | Reason For Dropping Variable |
| CustomerId | This is the unique identifier for each customer. The data here does not follow any pattern or trend that the model can learn from, making them non-informative for prediction purposes. |
| Surname | Surnames usually have high cardinality, meaning there are many unique values. This can lead to issues with certain algorithms, increasing model complexity and potentially degrading performance. |
| RowNumber | The same assumption holds from our discussion on CustomerId |
| Balance | As part of our data preparation step, this variable is removed and replaced with the log transformation of the column and a binary feature representing if customer has zero balance or a non-zero balance. |
| Complain | This column is dropped later in our analysis after first iteration in our model building phase. Many insights justify the dropping of this column. First, perfect correlation between this variable and target after multivariate outlier detection is done using mahalanobis algorithm. More than 5 algorithms achieve a 100% accuracy in precision, recall, accuracy. Doesn’t follow real-world assumptions for determining churn. These reasons and more prompt us to drop this variable. |

|  |  |
| --- | --- |
| Significance Level | 0.05 |
| Findings | From our chi-square distribution, with a significance level of 0.05, any value for the mahalanobis distance of each data point to its distribution greater than 26.3 is considered an outlier. We find 14 of such data points from our dataset of 10000 data points that are labelled outliers by the mahalanobis algorithm. |
| Insights | The analysis begins with the observation of a high correlation (0.96) between the 'complain' variable and the 'exited' variable, indicating that the predictor is almost perfectly correlated with the target. This high correlation is a problem.  Using the Mahalanobis distance algorithm to check for outliers in the multivariate space identifies 14 data points as outliers. These outliers fall into two distinct groups:  1. Customers who made complaints but did not leave the bank.  2. Customers who did not make complaints but left the bank.  Removing these 14 data points from the dataset, the correlation between the 'complain' and 'exited' variables becomes perfect (1.0), meaning that all remaining customers who made complaints left the bank, and those who did not complain stayed with the bank. |

|  |  |
| --- | --- |
| Derived Feature | Description |
| Age\_Group | Created from the Age variable using equal-width with 10 bins. |
| PointEarned\_Group | Created from the Point Earned variable using equal-frequency binning with 5 bins. |
| CreditScore\_Group | Created from CreditScore variable using equal-width binning with 10 bins. |
| Balance\_Zero | Used as part of a two-step feature process to fix the zero-inflated distribution issue found in the balance variable. The derived variable is a binary variable which sets a one when the balance is zero and vice-versa. |
| Balance\_Log | Used as part of a two-step feature process to fix the zero-inflated distribution issue found in the balance variable. The derived variable is a log transformation of the balance variable while setting zero balances to negative one. |
| Log\_Age | Logarithmic transformation of the age variable to fix left-skewness of the distribution. |
| Log\_CreditScore | Logarithmic transformation of the creditscore variable to fix right-skewness of the distribution. |

|  |  |  |
| --- | --- | --- |
|  | Data Preparation Step | Step Explanation |
| 1 | Dropping Irrelevant Columns | Here we drop the variables – RowNumber, CustomerId, and Surname. During the second iteration in our modelling, we drop the complain variable |
| 2 | Check for Outliers | Multivariate outlier detection using Mahalanobis distance of data points from the distribution |
| 3 | Handling Zero-Inflated Distribution | The balance variable has a very high number of zeros |
| 4 | Data Transformation | The variables – Geography, Card Type, and Gender are categorical and therefore need to be encoded |
| 5 | Feature Engineering | New features are created using domain knowledge and logarithmic transformations |
| 6 | Splitting the Dataset | Here, we use the structure defined by the CRISP-DM, splitting the dataset 80% for training and 20% for testing. From the 80% used in training, we use 33% for validation and refinement of model parameters and the rest for training |
| 7 | Handling Class Imbalance | This is done only on the training data using the ADASYN oversampling technique as an improvement on the SMOTE algorithm. |
| 8 | Scaling the Dataset | Using the MinMaxScaler, we scale the training data then transform the test and validation data |
| 9 | Feature Selection | Features are selected based on their p-value calculated using f-statistics ANOVA. Here, 18 features are selected |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Variable | Scores | P-Value | Selected/Not Selected |
| 1 | Log\_Age | 1472.51 | 0.0 |  |
| 2 | IsActiveMember | 1471.38 | 0.0 |  |
| 3 | Age | 1244.12 | 0.0 |  |
| 4 | Balance\_Zero | 844.52 | 0.0 |  |
| 5 | Gender\_Male | 810.85 | 0.0 |  |
| 6 | Age\_Group | 644.09 | 0.0 |  |
| 7 | Geography\_Spain | 479.64 | 0.0 |  |
| 8 | NumOfProducts | 436.75 | 0.0 |  |
| 9 | HasCrCard | 273.96 | 0.0 |  |
| 10 | Balance\_Log | 168.35 | 0.0 |  |
| 11 | CreditScore\_Group | 163.88 | 0.0 |  |
| 12 | Satisfaction Score | 150.16 | 0.0 |  |
| 13 | PointEarned\_Group | 95.99 | 0.0 |  |
| 14 | Card Type | 88.25 | 0.0 |  |
| 15 | Tenure | 33.29 | 0.0 |  |
| 16 | Log\_CreditScore | 22.68 | 0.0 |  |
| 17 | CreditScore | 20.65 | 0.0 |  |
| 18 | EstimatedSalary | 5.05 | 0.02 |  |
| 19 | Point Earned | 0.25 | 0.62 |  |
| 20 | Geography\_Germany | 0.17 | 0.68 |  |

Figure 10.1: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) for Algorithms as seen from Confusion Matrix

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Algorithm | TP | TN | FP | FN | Team Member |
| 0 | Logistic Regression | 407 | 1591 | 0 | 0 | Leonard |
| 1 | LightGBM Classifier | 407 | 1591 | 0 | 0 | Leonard |
| 2 | Gradient Boosting | 408 | 1592 | 0 | 0 | Chidera |
| 3 | Random Forest Classifier | 408 | 1592 | 0 | 0 | Chidera |
| 4 | Decision Tree Classifier | 407 | 1591 | 0 | 0 | Leonard |
| 5 | SVC | 407 | 1591 | 0 | 0 | Leonard |
| 6 | XGBoost Classifier | 407 | 1591 | 0 | 0 | Leonard |

Figure 10.2: Evaluation Metrics for Algorithm Test Data

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Algorithm | Accuracy | Precision | Recall | F1-Score | TPR | TNR | PC1 | PC0 |
| 0 | Logistic Regression | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1 | LightGBM Classifier | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 2 | Gradient Boosting | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 3 | Random Forest Classifier | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 4 | Decision Tree Classifier | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 5 | SVC | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 6 | XGBoost Classifier | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Algorithm | TP | TN | FP | FN | Team Member |
| 1 | Gradient Boosting | 250 | 1377 | 215 | 158 | Chidera |
| 2 | Random Forest Classifier | 263 | 1340 | 252 | 145 | Chidera |
| 3 | XGBoost | 219 | 1475 | 116 | 188 | Leonard |
| 4 | SVC | 195 | 1414 | 177 | 212 | Leonard |
| 5 | Logistic Regression | 181 | 1360 | 231 | 226 | Leonard |
| 6 | LightGBM | 205 | 1475 | 116 | 202 | Leonard |
| 7 | Decision Tree | 243 | 1321 | 270 | 164 | Leonard |

Figure 11.2: Evaluation Metrics for Algorithm Test Data

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Algorithm | Accuracy | Model Precision | Model Recall | F1-Score | TPR | TNR | Test Precision (C\_1) | Test Precision (C\_0) |
| 1 | Gradient Boosting | 0.81 | 0.82 | 0.81 | 0.82 | 0.62 | 0.86 | 0.54 | 0.90 |
| 2 | Random Forest | 0.80 | 0.82 | 0.80 | 0.81 | 0.64 | 0.84 | 0.51 | 0.90 |
| 3 | Decision Tree | 0.78 | 0.80 | 0.78 | 0.79 | 0.60 | 0.83 | 0.47 | 0.89 |
| 4 | LightGBM | 0.84 | 0.83 | 0.84 | 0.83 | 0.50 | 0.93 | 0.64 | 0.88 |
| 5 | XGBoost | 0.85 | 0.84 | 0.85 | 0.84 | 0.54 | 0.93 | 0.65 | 0.89 |
| 6 | Logistic Regression | 0.77 | 0.77 | 0.77 | 0.77 | 0.44 | 0.85 | 0.44 | 0.86 |
| 7 | SVC | 0.81 | 0.80 | 0.81 | 0.80 | 0.48 | 0.89 | 0.52 | 0.87 |

* **Training Data Results without Complain Variable and Classes Balanced**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Algorithm | Accuracy | Model Precision | Model Recall | F1-Score | TPR | TNR | Test Precision (C\_1) | Test Precision (C\_0) |
| 1 | Logistic Regression | 0.83 | 0.83 | 0.83 | 0.83 | 0.82 | 0.85 | 0.84 | 0.82 |
| 2 | LightGBM Classifier | 0.94 | 0.94 | 0.94 | 0.94 | 0.92 | 0.97 | 0.96 | 0.92 |
| 3 | Gradient Boosting | 0.95 | 0.95 | 0.95 | 0.95 | 0.93 | 0.97 | 0.97 | 0.94 |
| 4 | Random Forest | 0.84 | 0.84 | 0.84 | 0.84 | 0.82 | 0.86 | 0.86 | 0.83 |
| 5 | Decision Tree | 0.81 | 0.81 | 0.81 | 0.81 | 0.82 | 0.80 | 0.80 | 0.81 |
| 6 | SVC | 0.87 | 0.87 | 0.87 | 0.87 | 0.85 | 0.90 | 0.89 | 0.85 |
| 7 | XGBoost Classifier | 0.98 | 0.98 | 0.98 | 0.98 | 0.96 | 0.99 | 0.99 | 0.96 |

* **Validation Data Results without Complain Variable and Classes Balanced**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Algorithm | Accuracy | Model Precision | Model Recall | F1-Score | TPR | TNR | Test Precision (C\_1) | Test Precision (C\_0) |
| 1 | Logistic Regression | 0.77 | 0.77 | 0.77 | 0.77 | 0.47 | 0.85 | 0.44 | 0.87 |
| 2 | LightGBM Classifier | 0.84 | 0.83 | 0.84 | 0.83 | 0.54 | 0.92 | 0.61 | 0.89 |
| 3 | Gradient Boosting | 0.84 | 0.82 | 0.84 | 0.83 | 0.49 | 0.93 | 0.62 | 0.88 |
| 4 | Random Forest | 0.81 | 0.82 | 0.81 | 0.81 | 0.59 | 0.87 | 0.52 | 0.90 |
| 5 | Decision Tree | 0.76 | 0.79 | 0.76 | 0.77 | 0.59 | 0.81 | 0.44 | 0.89 |
| 6 | SVC | 0.79 | 0.78 | 0.79 | 0.78 | 0.44 | 0.88 | 0.48 | 0.87 |
| 7 | XGBoost Classifier | 0.84 | 0.82 | 0.84 | 0.83 | 0.49 | 0.93 | 0.62 | 0.88 |

Figure 11.3: Mean - Cross Validation (Train and Test), Fit Time, and Score Time

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Fit time | Score time | CV Test score | CV Train score |
| Decision Tree Classifier | 0.04 | 0.0 | 0.86 | 0.85 |
| Gradient Boosting Classifier | 18.18 | 0.016 | 0.84 | 0.94 |
| LGBM Classifier | 0.31 | 0.006 | 0.86 | 0.94 |
| Logistic Regression | 1.54 | 0.001 | 0.82 | 0.82 |
| Random Forest Classifier | 0.94 | 0.024 | 0.85 | 0.86 |
| SVC | 7.26 | 0.26 | 0.82 | 0.84 |
| XGBoost | 1.30 | 0.007 | 0.85 | 0.99 |

Figure 11.4: Standard Deviation - Cross Validation (Train and Test), Fit Time, and Score Time

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Fit time | Score time | CV Test score | CV Train score |
| Decision Tree Classifier | 0.006 | 0.001 | 0.01 | 0.001 |
| Gradient Boosting Classifier | 1.404 | 0.01 | 0.013 | 0.002 |
| LGBM Classifier | 0.062 | 0.002 | 0.02 | 0.001 |
| Logistic Regression | 0.1 | 0.0 | 0.009 | 0.001 |
| Random Forest Classifier | 0.113 | 0.006 | 0.012 | 0.002 |
| SVC | 0.821 | 0.04 | 0.008 | 0.001 |
| XGBoost | 0.338 | 0.009 | 0.012 | 0.002 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Classifier | Tuning Technique | Changes | Validation For Selection |
| 1 | Random Forest | Domain Understanding | max\_depth = 5  max\_features = None  random\_state = 0  warm\_start = True | Domain knowledge suggests limiting the depth to prevent overfitting and no limit on features to consider all variables. The random\_state ensures reproducibility, and warm\_start allows for incremental training. |
| 2 | Gradient Boosting | Domain Understanding | learning\_rate = 0.01  n\_estimators = 300  max\_depth = 5  random\_state = 0  n\_iter\_no\_change = 10 | A low learning\_rate with many n\_estimators help in fine-tuning the model. Limiting max\_depth prevents overfitting, and n\_iter\_no\_change stops training early if no improvement, avoiding overfitting and saving time. |
| 3 | Logistic Regression | Grid Search | C = 0.01  l1\_ratio = 0.1  penalty = None  solver = 'newton-cg' | Grid Search results indicate these parameters provide the best balance between bias and variance, with C controlling regularization, l1\_ratio mixing L1 and L2 regularization, and newton-cg solver suitable for small datasets. |
| 4 | Decision Tree | Grid Search | criterion = 'entropy'  max\_depth = 10  min\_samples\_leaf = 10  min\_samples\_split = 10 | criterion = 'entropy' improves information gain; setting max\_depth, min\_samples\_leaf, and min\_samples\_split helps in reducing overfitting and ensures robustness of the tree. |
| 5 | SVC | Domain Understanding | kernel = 'rbf'  gamma="auto"  probability = True  random\_state = 0  C = 1.0 | The RBF kernel captures non-linearities well. gamma = 'auto' sets a default value based on features, C = 1.0 balances margin and misclassification, and probability = True provides probability estimates. |
| 6 | XGBoost | Grid Search | random\_state = 0  learning\_rate = 0.1  max\_depth = 5  min\_child\_weight = 5  n\_estimators = 300  subsample = 0.9 | Grid Search optimizes these hyperparameters to prevent overfitting and enhance generalization. learning\_rate, max\_depth, min\_child\_weight, and subsample are critical in managing bias-variance trade-off. |
| 7 | LightGBM | Grid Search | max\_depth = 6  min\_child\_samples = 30  n\_estimators = 200  subsample = 0.6 | Grid Search results show these parameters prevent overfitting while maintaining performance. max\_depth and min\_child\_samples handle complexity and overfitting; subsample ensures diversity in the boosted trees. |