

**Midterm - Credit Card Fraud Detection Based on Data Mining Techniques**

**Prepared by:** Amirkhan Serikbay

**ID:** 21B030716

**Faculty: SITE**

**Major:** Computer Systems and Software

Almaty, 2024

**Content**

[Introduction 3](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.3znysh7)

[1. Data Preprocessing](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.tyjcwt) 5

[1.1. Data loading and basic structure analysis](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.3dy6vkm) 5

[1.2. Checking and handling missing values](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.3dy6vkm) 6

[1.3. Checking and removing duplicates](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.3dy6vkm) 6

[1.4. Scaling and standardization of features](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.3dy6vkm) 7

[1.5. Balancing classes](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.3dy6vkm) 8

[2. Exploratory Data Analysis (EDA)](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.4d34og8) 8

[2.1 Construction of histograms of feature distribution 9](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.17dp8vu)

[2.2 Boxplot for class comparison 10](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.17dp8vu)

[2.3 Correlation matrix](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.17dp8vu) 10

[2.4 Pairplot distributions of key features 11](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.17dp8vu)

[2.5 Visualization of anomalies and outliers 12](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.17dp8vu)

[3. Feature selection and construction of a basic logistic regression model using RFE](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.4d34og8) 13

[3.1 Feature Selection with RFE](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.17dp8vu) 13

[3.2 Building a Basic Logistic Regression Model](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.17dp8vu) 14

[3.3 Interpretation of the coefficients of the features](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.17dp8vu) 14

[[4. Comparison of basic classification models](#_heading=h.4d34og8)](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.lnxbz9) 15

4.1 DecisionTreeClassifier 15

4.2 RandomForestClassifier 16

4.3 Conclusions 17

[[5. Logistic Regression Optimization](#_heading=h.4d34og8)](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.lnxbz9) 19

[[6. Advanced classification methods and their comparison with baseline models](#_heading=h.4d34og8)](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.lnxbz9) 19

6.1 Neural network (MLPClassifier) 19

6.2 RandomForestClassifier 20

6.3 Conclusion 20

[Conclusion](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.35nkun2) 21

[References](file:///C:\Users\SERIKAMI\Downloads\отчет%20по%20практике2%20(1).docx#_heading=h.1ksv4uv) 22

**Introduction**

**Brief description:** The project aims to refine and optimize machine learning models to detect fraudulent transactions based on proprietary credit card transaction data. The project will implement a full cycle of data analysis: from pre-processing and visualization to building and evaluating various classification algorithms. The main objective is to build a construction model that can predict fraud despite the requirement for class imbalance.

**The goal** of this project is to detect fraudulent transactions based on credit card data. We will use machine learning methods to build a model that can recognize fraudulent transactions and minimize financial losses for the company conducting such transactions.

**Description of the dataset:**

We work with a classic credit card fraud dataset that contains transaction records with different characteristics. Here are the main features of the dataset:

* Number of rows: 284,807 transactions.
* Number of features: 30 (including the target variable).
* Target variable:
  1. Class — 0 means a legitimate transaction, 1 — fraud.
* Features in the dataset:
  1. Time — transaction time from the first transaction in the dataset (seconds).
  2. V1–V28 — anonymized features obtained using PCA (Principal Component Analysis).
  3. Amount — transaction amount (in euros).
  4. Class — target variable (fraud or not).

**Dataset Issues:**

Unrecognizable Features: Features V1, V2, ..., V28 are the result of PCA and have no obvious interpretation. This makes them difficult to use in traditional data analysis.

Unbalanced Classes: The dataset is highly unbalanced. There are many more legitimate transactions than fraudulent ones. This means that conventional classification methods may be biased towards the dominant class (legitimate transactions).

Feature Scaling: The Amount feature has a wide range of values ​​and requires scaling to bring it to the same level as the other features.

**PCA (Principal Component Analysis)** is a dimensionality reduction and data analysis technique used to reduce the number of features in a data set while preserving as much information about the data as possible. It helps identify the principal components (the main axes) that best explain the variation in the original data.

**Why is this important?**

Financial Losses: Credit card fraud costs financial institutions millions of dollars every year.

Customer Trust: The ability of a bank or company to detect suspicious transactions helps maintain a high level of customer trust.

Minimize False Positives: It is important not only to find all cases of fraud, but also to reduce the number of false positives (false accusations of fraud) so as not to disturb customers.

**Metrics for evaluating the quality of models:**

Since the problem is highly imbalanced, conventional metrics (e.g., accuracy) may not be suitable. Important metrics:

* Recall: A metric that shows the proportion of fraudulent transactions that are correctly predicted.
* Precision: The proportion of true positives among all transactions that are labeled as fraudulent.
* AUC-ROC: The area under the error curve, showing the tradeoff between recall and precision.

**1. Data Preprocessing**

Data preprocessing is an important step in any machine learning project. It involves preparing the initial dataset for further analysis and model building. Proper data preprocessing helps improve the quality and accuracy of models, as well as avoid errors and distortions in the results.

The main steps in the preprocessing step are:

* Data loading and basic structure analysis.
* Checking and handling missing values.
* Checking and removing duplicates.
* Feature scaling and standardization.
* Class balancing to address the imbalance issue.
  1. **Data loading and basic structure analysis**

In this step, we load the data into the analysis environment and examine its main characteristics. To do this, we perform the following steps:

* Data import: Data is loaded from the source file (in our case, a CSV file). We used the pandas library to load the data as a DataFrame, which is convenient for further analysis.
* Viewing the first rows of data: Using the head() method to print the first 5-10 rows of the dataset to ensure that the data has loaded correctly and matches the expected structure.
* Printing information about the dataset: The info() method allows you to quickly get information about the data types, the number of non-zero values ​​in each column, and the total amount of memory occupied by the dataset.
* Analyzing the dimensions of the data: shape prints the dimensions of the dataframe in the format (number of rows, number of columns). This helps to estimate how much data we have and how many features are available for analysis.
* Basic statistical characteristics of the data: The describe() method prints the basic statistical parameters (mean, standard deviation, minimum and maximum values, etc.) for all numeric features. This allows you to understand how the feature values ​​are distributed and identify possible outliers or abnormal values.

**Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание**

**Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание**

**1.2. Checking and Handling Missing Values**

Missing values ​​can negatively affect the quality and stability of models. Therefore, it is important to identify them and take appropriate measures:

* Checking for missing values: The isnull().sum() method can be used to count the number of missing values ​​in each column. In our case, some features (V7 - V28, Amount) and the target variable (Class) had one missing value each.
* Choosing a strategy for handling missing values: Removing rows: Applicable if missing values ​​are a small part of the total number of rows.
* Replacing with mean or median value: Applicable for numeric features if there are many missing values. In this project, due to the very small number of missing values ​​(less than 0.01% of the total data), it was decided to remove rows with missing values.

**1.3. Checking and removing duplicates**

Duplicate rows can distort the results of the analysis, especially if they belong to rare classes. At this stage:

* Checking for duplicates: We use the duplicated().sum() method to count the number of duplicate rows. In our data set, 153 duplicate rows were found.
* Removing duplicates: We use the drop\_duplicates() method. After removing duplicates, we always check the size of the data set and the distribution of classes.

**Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание**

**1.4. Feature Scaling and Standardization**

Scaling is the process of bringing numeric features to the same scale. This is necessary when features have different ranges of values, as most machine learning models are sensitive to the scale of the input data.

* Scaling problem: The Amount feature (transaction amount) has a wide range of values, while features V1 - V28 are already standardized. To bring Amount to the same scale, we use StandardScaler from sklearn.
* Standardization: Transforming the Amount feature to a standardized form (z-score), where values ​​are expressed as deviations from the mean.
* Updating the dataset: The scaled Amount feature is replaced in the original dataset.

**Изображение выглядит как текст, снимок экрана, Шрифт, веб-страница

Автоматически созданное описание**

**1.5. Class Balancing**

Class imbalance (e.g. 99% of records are legitimate and 1% are fraudulent) causes the model to ignore the minority and focus on predicting the majority. This leads to poor results despite high overall accuracy.

* Problem: In our case, legitimate transactions (0.0) are significantly predominant (37,618 records), while fraudulent transactions (1.0) only number 103.
* Balancing techniques: Undersampling: Reducing the number of class 0 records. Oversampling: Increasing the number of class 1 records (e.g. duplicating existing ones or creating synthetic data).
* SMOTE (Synthetic Minority Over-sampling Technique): Generate synthetic examples for the minority class based on nearest neighbor interpolation.
* Choosing a balancing method: We chose to reduce the number of legitimate transactions (Undersampling) so that the number of legitimate and fraudulent transactions became equal. This method is suitable for the initial analysis stage, as it reduces the size of the data, making it more manageable for visualization and basic models.
* Implementation of the method: Legal transactions (Class = 0.0) are reduced to the number of fraudulent ones (Class = 1.0), i.e. to 103 records. As a result, we have a balanced dataset in which 50% of legitimate and 50% of fraudulent transactions.
* Checking the distribution of classes after balancing: The distribution of classes is checked using the value\_counts() method to ensure that the classes are balanced.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

1. **Exploratory Data Analysis (EDA)**

During the Exploratory Data Analysis (EDA) phase, we performed a comprehensive feature analysis to understand the structure of the data, identify patterns, and identify key features for further building machine learning models. This step allows us to form an idea of ​​the dataset and make decisions about feature selection and processing. Let's describe in detail what was done at this stage.

The main goals of EDA are:

* Evaluate the distribution of features and understand their structure.
* Check how features differ depending on the target class.
* Identify relationships between features (correlations).
* Find key features that best separate classes.
* Identify anomalies and outliers that may affect the quality of models.

**2.1. Building Feature Distribution Histograms**

Goal: To study the distribution of each feature and determine how much the feature values ​​differ between legitimate and fraudulent transactions.

* What was done: Distribution histograms were built for all features in the dataset (V1 - V28, Time, Amount). Peaks, outliers, and distribution symmetry were analyzed.
* Key observations: Most features have an asymmetric distribution with a high peak in a certain area (e.g. V1, V2, V3). Features V4, V11, and Amount show significant outliers that may indicate anomalous transactions. Feature Amount is concentrated in the area of ​​small values, but some transactions have very high amounts, which is important to consider.
* Conclusions: Features V1, V2, V3, V4, and V11 require detailed analysis, as their distribution differs from other features. Feature Amount can be useful for identifying suspicious transactions with large amounts.

Изображение выглядит как диаграмма

Автоматически созданное описание

**2.2. Boxplot for class comparison**

Goal: Understand how feature values ​​are distributed across classes (Class = 0 or 1).

* What was done: Boxplots were plotted for each feature across the target variable (Class). Compared medians, quartiles, and outliers for legitimate (0) and fraudulent transactions (1).
* Key observations: Features V2, V4, V11, V14, and V17 show significant differences in distribution between classes. Features V3, V10, V16, and V21 also show differences, but to a lesser extent. Feature Amount does not show significant differences in medians, but outliers indicate important transactions.
* Conclusions: Features V2, V4, V11, V14, and V17 may be key for classification, as the differences between classes are clear. The Amount feature should be taken into account when analyzing outliers, since fraudulent transactions can have high values.

Изображение выглядит как текст, диаграмма, План, снимок экрана

Автоматически созданное описание

**2.3. Correlation Matrix**

Goal: Understand the relationships between features and identify highly correlated features that may be redundant.

* What was done: A correlation matrix was built for all features, including the target variable (Class). A heatmap was built to visualize the strength of the correlations.
* Key observations: Features V12, V14, and V17 have a high cross-correlation, indicating redundancy. Feature Amount has an independent distribution and does not correlate with other features. Features V11, V13, and V18 also have a high correlation with each other, which should be taken into account when selecting features.
* Conclusions: Features with high correlation (V12, V14, V17) can be combined or removed to prevent overfitting. Features that do not correlate with others (Amount) can be useful for classification.

Изображение выглядит как снимок экрана, шаблон, Красочность, линия

Автоматически созданное описание

**2.4. Pairplot distributions of key features**

Goal: Understand how classes (legitimate and fraudulent transactions) are separated in the space of several key features.

* What was done: Pairplots were constructed for several key features (V1, V2, V3, Amount). Scatter plots showed how points are distributed across classes (legitimate and fraudulent transactions).
* Key observations: Features V1, V2, and V3 demonstrate good separation of classes, especially in areas with extreme values. Feature Amount does not have a clear separation, but outliers may indicate anomalous transactions.
* Conclusions: Features V1, V2, and V3 have high information content and should be included in models. Feature Amount plays a supporting role, especially in case of outliers.

Изображение выглядит как текст, диаграмма, карта, линия

Автоматически созданное описание

**2.5. Visualizing Anomalies and Outliers**

Goal: Determine which features contain anomalies and outliers that may affect the quality of the models.

* What was done: Boxplots were plotted for the Amount feature depending on the class. Transactions with abnormally high amounts were identified.
* Key observations: For class 1 (fraud), outliers have high values, unlike class 0. This indicates anomalous transactions that may be a sign of fraud.
* Conclusions: It is necessary to pay attention to the treatment of outliers and anomalies for the Amount feature when building a model.

Изображение выглядит как текст, снимок экрана, диаграмма, Прямоугольник

Автоматически созданное описание

**3. Feature selection and building a baseline logistic regression model using RFE**

In this step, we focused on selecting the most significant features and applying the baseline logistic regression model. The goal of this step is to select key features that have the greatest impact on predicting the target variable (fraud) and evaluate their contribution to the overall performance of the model.

The main tasks were:

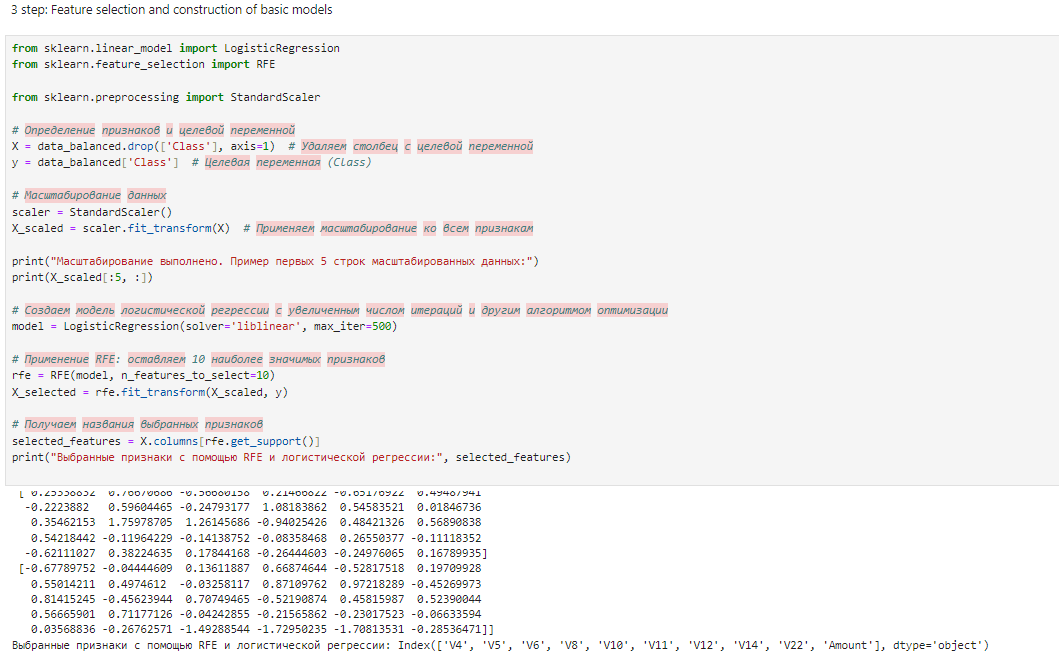
* Using recursive feature elimination (RFE) to select the most significant features.
* Building the baseline logistic regression model and evaluating its performance.
* Analyzing the feature coefficients to determine their impact on the probability of fraud.

**3.1. Feature Selection with RFE**

The Recursive Feature Elimination (RFE) method was used for feature selection. RFE uses a model (in our case, logistic regression) to estimate the importance of each feature and recursively eliminates the least important features until a specified number (10 features) remains.

Why RFE:

* RFE allows us to take into account the interaction of features and their contribution to the model, which is important when there are a large number of features.
* Logistic regression was used as the base model for evaluation, as it is well interpretable and provides an understanding of the influence of each feature.



**3.2. Building a Baseline Logistic Regression Model**

After feature selection, a baseline logistic regression model was built. We used logistic regression on the selected features and evaluated its performance on the test dataset.

* Goal: To evaluate how well the model classifies legitimate and fraudulent transactions based on the selected features.
* Evaluation metrics: Accuracy, AUC-ROC, and a detailed classification report (classification\_report).

Model Results:

* Test Set Precision: 0.9355
* AUC-ROC: 0.9375

Classification Report:

* The model demonstrated high precision and recall for both classes.
* The precision for fraudulent transactions (class 1) was 0.88, indicating that the model has a high ability to detect fraud.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

**3.3. Interpreting feature coefficients**

Logistic regression coefficients help us understand how each feature contributes to predicting the target variable:

* Positive coefficients: Increase the likelihood that a transaction is fraudulent.
* Negative coefficients: Decrease the likelihood of fraud.

Изображение выглядит как текст, снимок экрана, число

Автоматически созданное описание

**4. Comparison of Base Classification Models**

In this stage of the project, we focused on building and comparing different base classification models to identify the most effective strategy. The main goal is to evaluate which model performs better in classification based on the selected features and current data. We built three models: logistic regression, decision tree, and random forest.

Problem statement:

* Compare base models on a balanced data set.
* Evaluate the performance of the model’s using accuracy and AUC-ROC metrics.
* Determine which model best classifies transactions as legitimate or fraudulent.
* Based on the results, select the optimal model for further optimization or advanced methods.

For each of the three models, we used the same data that was selected in the previous steps using the recursive feature elimination (RFE) method. The 10 most significant features were selected: ['V4', 'V5', 'V6', 'V8', 'V10', 'V11', 'V12', 'V14', 'V22', 'Amount']

**4.1. DecisionTreeClassifier**

The decision tree model (DecisionTreeClassifier) ​​was applied with default settings. Training was performed on the selected features to evaluate the influence of these features on the decision tree.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

**4.2. RandomForestClassifier**

A random forest model (RandomForestClassifier) ​​consisting of several decision trees is used. The goal is to evaluate how combining decision trees improves performance.

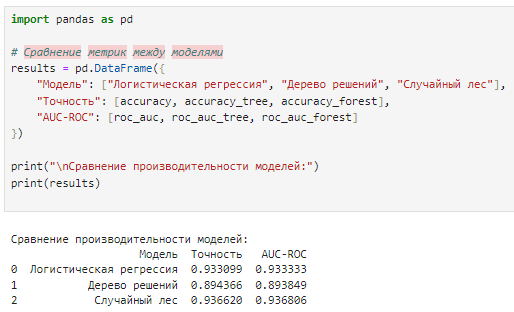
Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

**4.3. Conclusion**

Metrics used to evaluate the models:

* Accuracy — the proportion of correctly predicted examples among all observations.
* AUC-ROC (Area Under Curve — Receiver Operating Characteristic) — a metric that takes into account the ratio of true positive and false positive predictions, which is especially important for unbalanced classes.
* Classification report (Precision, Recall, F1-score) — metrics that allow you to evaluate the quality of classification for each class separately.



Logistic regression performed best in terms of both metrics (accuracy and AUC-ROC).

Decision tree and random forest performed similarly, which may indicate overfitting issues or insufficient complexity of the models.

Logistic regression performed better on the selected features, indicating the linear nature of the dependencies in the data.

Logistic regression is the best model among the basic algorithms considered. Now we can move on to the next step - optimizing hyperparameters or using more complex algorithms to further improve the classification quality.

**5. Logistic Regression Optimization**

At this stage, we performed the optimization of the logistic regression hyperparameters in order to improve the classification quality and the performance of the model on the test dataset.

Optimization goal:

* Find the best logistic regression hyperparameters to maximize the AUC-ROC metric value and improve the generalization ability of the model.
* Determine how much the hyperparameter optimization will improve the performance compared to the baseline model.

Optimization method: The GridSearchCV method was used to find the optimal parameters, which performs a complete enumeration of all combinations of hyperparameters based on cross-validation. The parameters that were optimized:

* C — regularization coefficient (controls the degree of penalty for large feature weights). Values: [0.01, 0.1, 1, 10, 100].
* penalty — regularization type (l1, l2).
* solver — optimization algorithm (liblinear, saga).

Optimization process:

* Initialize logistic regression and tune the hyperparameter grid.
* Use 5-fold cross-validation to evaluate each parameter combination.
* Find the best parameter combination that yields the highest AUC-ROC metric.

Optimization results: best model parameters:

* C: 0.1
* penalty: l1
* solver: liblinear
* Best AUC-ROC on cross-validation: 0.9886

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

|  |  |  |
| --- | --- | --- |
| Model | Accuracy | AUC-ROC |
| Basic logistic regression | 0.933099 | 0.9333 |
| Optimized Logistic Regression | 0.9401 | 0.9403 |

The optimized model's performance on the test set is slightly lower than that of the baseline logistic regression. This may indicate overfitting of the model on the training set or sensitivity to regularization.

Optimization resulted in significant improvement in the cross-validation step (AUC-ROC = 0.9821), but the performance on the test set remained at the level of the baseline model.

The fitted parameters (C=0.1, penalty='l2') indicate that L2 regularization performs better on this dataset

**6. Advanced classification methods and their comparison with baseline models**

In this step, we explore advanced classification methods to evaluate their performance and compare them with previously built base models. Advanced methods include random forest, support vector machines (SVM), and neural networks. These algorithms often have high power to detect complex dependencies and can improve the quality of classification.

Main objective:

* Apply advanced algorithms to the classification task.
* Evaluate their performance using accuracy and AUC-ROC metrics.
* Compare the results with the base models to determine whether the advanced methods should be used as the final model.

Neural network (MLPClassifier)

Neural networks are capable of modeling complex nonlinear relationships and can be useful for deeper analysis. We will use the Multilayer Perceptron (MLPClassifier) ​​from the scikit-learn library.

**6.1 Neural network (MLPClassifier)**

Neural networks are capable of modeling complex nonlinear relationships and can be useful for deeper analysis. We will use the Multilayer Perceptron (MLPClassifier) ​​from the scikit-learn library.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

**6.2. Support Vector Machines (SVMs)**

Support Vector Machines are a powerful algorithm that finds the hyperplane that best separates classes. It works well on small datasets and can find complex separating boundaries.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

**6.3. Conclusion**

SVM and neural network showed the highest accuracy and AUC-ROC values ​​among all models:

* SVM: AUC-ROC = 0.9439, Accuracy = 0.9437.
* Neural network: AUC-ROC = 0.9437, Accuracy = 0.9437.

Logistic regression (optimized) showed slightly lower results:

* AUC-ROC = 0.9403, Accuracy = 0.9401.

Despite this, logistic regression is still a good interpretable model with high accuracy.

We looked at all the key classification algorithms and concluded that SVM and Neural Network are the best candidates for the final model. Depending on your preference, you can use any of them or build an ensemble model to improve accuracy and robustness.

**Conclusion**

The goal of the project was to build and optimize a model for detecting fraudulent transactions using credit card data. We performed a full cycle of analysis and model building, starting with data preprocessing and exploratory data analysis (EDA) and ending with building, optimizing, and comparing different machine learning algorithms. The main focus was on the accuracy and capability of the models, which differ by class, since the fraud problem is very relevant and requires high accuracy with a minimum number of errors.

Project Steps Data Preprocessing:

1. Cleaning the data and eliminating missing data.

* Feature Scaling.
* Balancing the data using the undersampling method to obtain an equal number of includes classes "fraud" and "not fraud".

1. Exploratory Data Analysis (EDA):

* Identifying key features that demonstrate significant differences between classes.
* Building visualizations (histograms, boxplots, and correlation matrix) to understand the relationships between the components and their impact on the target variable.

1. Selection Criteria:

* Applying criterion selection methods such as SelectKBest and Recursive Feature Extraction (RFE).
* Selecting the top 10 features that have the greatest impact on predicting the fraud class.

1. Model Building:

* Building a logistic regression and evaluating its performance.
* Obtaining metric accuracy and AUC-ROC for the evaluation models on the test dataset.

1. Logistic Regression Optimization:

* Applying GridSearchCV to find the best hyperparameters.
* Obtaining the best parameters for the logistic regression and improving the quality of the models.

1. Building and Comparing Advanced Models:

* Applying advanced classification algorithms such as Random Forest, SVM, and Neural Network.
* Comparing all models and determining the most effective one.

The optimal model for this task is SVM with the highest AUC-ROC value (0.9439). This model has high accuracy and efficiency in distinguishing between the fraud and non-fraud classes, making it suitable for final use.

**References**

* <https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud>
* <https://www.w3schools.com/python/python_ml_train_test.asp>
* <https://www.w3schools.com/python/pandas/default.asp>
* <https://numpy.org/doc/stable/>
* https://scikit-learn.org/stable/