PROJECT: HOUSE LOAN DATA ANALYSIS (PREDICTING DEFAULT PAYMENTS)

DEEP LEARNING (TENSORFLOW WITH KERAS)

DESCRIPTION

For safe and secure lending experience, it's important to analyse the past data. In this project, you must build a deep learning model to predict the chance of default for future loans using the historical data. As you will see, this dataset is highly imbalanced and includes a lot of features that make this problem more challenging.

OBJECTIVE

Create a model that predicts whether or not an applicant will be able to repay a loan using historical data.

DOMAIN

Finance

ANALYSIS TO BE DONE

Perform data preprocessing and build a deep learning prediction model.

INTRODUCTION

The objective of this project is to develop a predictive model using TensorFlow Keras that can accurately classify whether a customer will default on their payment or not. This task is crucial for financial institutions to assess creditworthiness and make informed decisions regarding loan approvals and risk management.

STEPS PERFORMED

STEP-1: UNDERSTANDING THE BUSINESS PROBLEM/PROBLEM STATEMENT

In this initial step, we gained a clear understanding of the business problem, which is to predict whether a customer will default on their payment or not. This helped us establish the project's scope and define our objectives.

STEP-2: GETTING DATA (IMPORTING BY PANDAS)

We imported the dataset using the Pandas library, which allowed us to read and manipulate the data easily. The dataset contains various features related to customers' payment history, demographics, and financial behaviour.

STEP-3: UNDERSTANDING THE DATA

We examined the structure of the data, including the number of rows and columns, as well as the data types of each feature. This step helped us understand the dataset and identify potential issues such as missing values or imbalanced classes.

STEP-4: DATA CLEANING

We addressed any missing values or inconsistencies in the dataset. This involved techniques such as imputation, where missing values were filled with appropriate values, and data standardization to ensure consistency in the data format.

STEP-5: DATA VISUALIZATION

We performed data visualization techniques to gain insights into the dataset. This included creating various plots, such as histograms, bar charts, and scatter plots, to visualize the distribution and relationships between different features. These visualizations helped us identify any patterns or outliers in the data.

STEP-6: EXPLORATORY DATA ANALYSIS (EDA)

Through exploratory data analysis, we conducted statistical analysis and correlation analysis to gain deeper insights into the dataset. This involved calculating summary statistics, identifying correlations between features, and exploring relationships between variables. EDA helped us understand the data's characteristics and potential factors influencing default payments.

STEP-7: FEATURE ENGINEERING

As part of feature engineering, we calculated the percentage of default to payer for the TARGET column. This provided us with valuable information about the proportion of default payments in the dataset. This feature engineering step helped us derive additional insights and potentially improve model performance.

STEP-8: FEATURE SELECTION

We performed feature selection to identify the most relevant features that have a significant impact on predicting default payments. This involved techniques such as correlation analysis, feature importance ranking, or domain knowledge-based selection. By selecting the most informative features, we aimed to improve model performance and reduce overfitting.

STEP-9: SPLITTING THE DATA

To train and evaluate our predictive model, we split the dataset into training and testing sets. This allowed us to assess the model's performance on unseen data and avoid overfitting. The training set was used to train the model, while the testing set was used to evaluate its accuracy and generalization capability.

STEP-10: MODEL BUILDING

We built a predictive model using TensorFlow Keras, a high-level neural networks API. The model architecture included input layers, hidden layers with activation functions, and an output layer with sigmoid activation for binary classification. We used appropriate loss functions and optimizers to train the model on the training data.

STEP-11: PREDICTION AND ACCURACY

Using the trained model, we made predictions on the test data and evaluated the accuracy of the model's predictions. We assessed metrics such as accuracy, precision, recall, and F1-score to measure the model's performance in classifying default and non-default cases.

STEP-12: TUNING AND IMPROVING ACCURACY

In order to improve the accuracy of our model, we performed several tasks:

1.BALANCED THE DATASET IF IT WAS IMBALANCED.

This involved techniques such as oversampling or under sampling to address class imbalance.

2.PLOTTED THE BALANCED OR IMBALANCED DATA TO VISUALIZE

The distribution and understand the impact of balancing techniques.

3.CALCULATED SENSITIVITY AS A METRIC

To evaluate the model's ability to correctly identify positive cases (default payments).

4.CALCULATED THE AREA UNDER THE RECEIVER OPERATING CHARACTERISTICS CURVE (AUC-ROC)

To assess the overall performance of the model in distinguishing between default and non-default cases.

CONCLUSION

In conclusion, we successfully developed a predictive model for predicting default payments. We went through several steps