Credit Score Classification (Multi-Class): Optimizing Financial Decision-Making

Deep Learning Techniques: Artificial Neural Network (ANN) Model

---

Duygu Jones | Data Scientist | Sep 2024
Follow me: duygujones.com | Linkedin | GitHub | Kaggle | Medium | Tableau

Project Overview: Credit Score Classification Using ANN

This project focuses on building a robust **Artificial Neural Network (ANN)** model to predict customer credit scores based on various attributes, including demographic, financial, and behavioral data.
The objective is to classify customers into three categories: **Good**, **Standard**, and **Poor** credit scores, assisting financial institutions in making informed lending decisions.

Importance of Credit Score in Finance

A **credit score** reflects a customer's creditworthiness, indicating how likely they are to repay debts on time.
Financial institutions use credit scores to assess risk, set interest rates, and decide loan terms.
Customers with **Good** credit scores are seen as low-risk and typically receive better financial terms, while those with **Poor** credit scores are considered high-risk, potentially facing higher interest rates or loan rejections.
Categorizing customers based on their credit scores helps lenders make faster, data-driven decisions.

Key Steps in the Project

Before training the **ANN model**, the following steps are performed to prepare the dataset:

1. **Handling missing and unusual values**: Addressing gaps and anomalies in the dataset through imputation and cleaning techniques.
2. **Encoding categorical features**: Converting categorical data, such as payment behavior, into numerical values for model compatibility.
3. **Scaling numerical features**: Ensuring features like **Annual Income** and **Outstanding Debt** are normalized to improve model performance.
4. **Data Cleaning**: Removing unnecessary columns, such as **ID** and **Name**, that do not contribute predictive value.

5. **Model Development (ANN)**: The **ANN** captures complex, non-linear relationships between features like **Outstanding Debt**, **Loan Amount**, and **Payment Behavior** to predict the customer's **Credit_Score**.
6. **Business Application**: By accurately predicting credit scores, the model supports financial institutions in making better lending decisions, setting appropriate interest rates, and segmenting customers for targeted financial products.

Why ANN for Credit Score Prediction?

Artificial Neural Networks are ideal for this task due to their ability to identify patterns and relationships in complex data. Credit scoring involves multiple interconnected financial factors, and an ANN can capture these interactions to deliver accurate credit score predictions.

Business Application

- **Loan Approvals**: By predicting a customer's credit score, the ANN model helps financial institutions assess the risk of lending to an individual.
- **Interest Rates**: Customers with higher credit scores receive more favorable loan terms, while those with lower scores are considered higher risk.
- **Risk Management**: Accurate credit score predictions enable banks to reduce financial risks and make better lending decisions, ultimately leading to better customer segmentation and optimized loan offerings.

Conclusion

This project highlights the crucial role that **credit scores** play in both financial decision-making and risk assessment. The use of an **ANN model** enables the accurate classification of customers based on their financial behavior, supporting real-world applications such as loan approval processes and interest rate determination. By leveraging deep learning techniques, we aim to create a highly effective model that can assist financial institutions in managing risk and improving customer service.

About the Dataset

The dataset consists of customer data, including demographic information, financial history, and payment behavior, which are important factors in determining credit scores. This data will be processed, cleaned, and engineered to extract the most important features for training the ANN model.
The target feature is the **Credit_Score**, which classifies customers into three categories based on their financial history:

1. **Good**: Low-risk customers with strong financial management.
2. **Standard**: Average-risk customers with moderate financial reliability.
3. **Poor**: High-risk customers who may struggle with debt repayments.

- **Dataset:** Credit Score Classification Dataset
- **Content:** Customer demographic, financial, and credit history details.
- **Number of Rows:** 100.000
- **Number of Columns:** 28

No	INPUTS	Description
1	**ID**	Unique identifier for each record.
2	**Customer_ID**	Unique identifier for each customer.
3	**Month**	Month of the transaction or record.
4	**Name**	Customer's name.
5	**Age**	The customer's age.
6	**SSN**	Customer's social security number.
7	**Occupation**	The customer's occupation.
8	**Annual_Income**	The customer's annual income.
9	**Monthly_Inhand_Salary**	The customer's monthly in-hand salary.

No	INPUTS	Description
10	**Num_Bank_Accounts**	Number of bank accounts owned by the customer.
11	**Num_Credit_Card**	Number of credit cards owned by the customer.
12	**Interest_Rate**	The interest rate applied to loans or credit.
13	**Num_of_Loan**	Number of loans taken by the customer.
14	**Type_of_Loan**	Type of loan taken by the customer.
15	**Delay_from_due_date**	The delay in payment from the due date.
16	**Num_of_Delayed_Payment**	Number of delayed payments made by the customer.
17	**Changed_Credit_Limit**	Changes made to the customer's credit limit.
18	**Num_Credit_Inquiries**	Number of credit inquiries made.
19	**Credit_Mix**	The mix of credit types the customer uses (e.g., loans, credit cards).
20	**Outstanding_Debt**	Total outstanding debt the customer has.
21	**Credit_Utilization_Ratio**	The ratio of credit used to the total credit limit.
22	**Credit_History_Age**	The length of the customer's credit history.
23	**Payment_of_Min_Amount**	Whether the customer pays the minimum amount required each month.
24	**Total_EMI_per_month**	The total EMI (Equated Monthly Installment) the customer pays each month.
25	**Amount_invested_monthly**	The amount invested by the customer each month.
26	**Payment_Behaviour**	The payment behavior of the customer.
27	**Monthly_Balance**	The customer's remaining balance at the end of each month.
28	**Credit_Score**	The customer's credit score (target variable: "Good," "Poor," "Standard").

- > *This dataset is ideal for developing credit risk assessment models, allowing for a detailed analysis of the factors that impact an individual's credit score.*
- > *Original dataset is available on Kaggle:* Credit score classification

---

Further Summary of the Project:

- **Comprehensive EDA**: Addressed missing and unusual values while retaining outliers to maintain the data's natural characteristics.
- **Preprocessing**: Meticulously encoded categorical columns and employed pipelines to prevent data leakage, ensuring data integrity.
- **Data Scaling**: Appropriately scaled the data to prepare for modeling.
- **Handling Imbalanced Classes**: Applied both SMOTE and class weighting techniques, with SMOTE proving more effective.
- **Model Optimization**: Iteratively optimized the ANN model architecture across eight configurations.
- **Final Model Selection**: Chose the ANN-7 model with SMOTE for its robust performance.
- **Model Testing**: Made predictions on the test dataset to validate the model's effectiveness.

For additional details about each model tested in this project, please visit the repository on my GitHub profile.

Table of Contents:

Setup and Initialization

Import the Libraries

In [1]:
```
#!pip install missingno
```

In [2]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import missingno
import matplotlib.pyplot as plt

%matplotlib inline
# %matplotlib notebook
plt.rcParams["figure.figsize"] = (12, 6)
# plt.rcParams['figure.dpi'] = 100
sns.set_style("whitegrid")
import warnings

warnings.filterwarnings("ignore")
warnings.warn("this will not show")
pd.set_option('display.float_format', lambda x: '%.3f' % x)
```

In [3]:
```
# Pre-Processing
from sklearn.preprocessing import OrdinalEncoder, OneHotEncoder, MinMaxScaler
from sklearn.model_selection import train_test_split, GridSearchCV, cross_val_score, cross_validate
# Metrics
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.metrics import roc_auc_score, roc_curve, precision_recall_curve, average_precision_score

# Model relavant libraries
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Activation, Dropout, BatchNormalization
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras.saving import save_model
from keras.optimizers import Adam
from keras.regularizers import l2
```

Load the Dataset

In [4]:
```
train0 = pd.read_csv('train.csv')
train = train0.copy()

train.head(3)
```