TREELEAF TECHNOLOGIES PVT. LTD. MACHINE LEARNING INTERNSHIP QUALIFICATION TASK

Submitted by:

Shovit Nepal

Task1: Bank loan classification

Problem Statement:

I was given a dataset of the bank loans consisting of 15 columns and a corresponding target column. The task is to build a machine-learning model that can accurately classify whether the personal loan was accepted or not based on the information provided.

Approach:

1. Initially, I loaded the dataset and checked for any missing values. I found missing values in the 'Gender', 'Income', 'Home Ownership', and 'Online' columns.
2. For the 'Income' column, I performed mean imputation, replacing the missing values with the mean of the existing values in the column.
3. I also observed some invalid placeholders ('#' and '-') in the 'Gender' column, and I replaced them with NaN to mark them as missing values.
4. For the 'Gender', 'Home Ownership', and 'Online' columns, I performed mode imputation, replacing the missing values with the mode (most frequent value) of each respective column.
5. I examined the unique values of the 'Personal Loan' column and found that it contains two classes: 0 (indicating no personal loan) and 1 (indicating accepting a personal loan). To ensure consistency, I replaced any empty spaces in this column with the value 0.
6. After data preprocessing, I built five machine learning models: SVM, Random Forest Classifier, Adaptive boosting, Gradient Boosting Classifier and XG Boost among which XG Boost gives best accuracy (98.90%).

Key Findings:

1. SVM Model:

Accuracy: **94.28%**

The SVM model showed good accuracy but struggled to predict class 1 (accepting personal loan), resulting in lower precision, recall, and F1-score for this class.

The model performed well for class 0 (not accepting personal loan) with high precision, recall, and F1-score.

- Classification Report:

	precision	recall	f1-score	support
0	0.95	0.99	0.97	902
1	0.85	0.48	0.62	95
accuracy			0.94	997
macro avg	0.90	0.74	0.79	997
weighted avg	0.94	0.94	0.94	997

- Confusion Matrix:

```
[[894   8]
 [ 49  46]]
```

2. Random Forest Classifier:

Accuracy: **98.29%**

The Random Forest model outperformed the SVM model in terms of accuracy.

The model showed high precision, recall, and F1-score for both classes, indicating better performance in predicting both classes.

- Random Forest Classification Report:

	precision	recall	f1-score	support
0	0.98	1.00	0.99	902
1	0.98	0.84	0.90	95
accuracy			0.98	997
macro avg	0.98	0.92	0.95	997
weighted avg	0.98	0.98	0.98	997

- Random Forest Confusion Matrix:

```
[[900   2]
 [ 15  80]]
```

3. Gradient Boosting Classifier:

Accuracy: **98.70%**

The Gradient Boosting model performed even better than the Random Forest model, achieving higher accuracy.

Similar to the Random Forest model, the Gradient Boosting model exhibited high precision, recall, and F1-score for both classes.

- Gradient Boosting Classification Report:

	precision	recall	f1-score	support
0	0.99	1.00	0.99	898
1	0.98	0.89	0.93	100
accuracy			0.99	998
macro avg	0.98	0.94	0.96	998
weighted avg	0.99	0.99	0.99	998

- Gradient Boosting Confusion Matrix:

```
[[896   2]
 [ 11  89]]
```

4. XG Boost

The XGBoost model achieved an accuracy of 98.90%, which is very high. From the classification report and confusion matrix, we can see that the XGBoost model performs exceptionally well. It shows high precision, recall, and F1-score for both classes. The accuracy is also impressive, and the confusion matrix confirms that the model is making accurate predictions.

- XGBoost Classification Report:

	precision	recall	f1-score	support
0	0.99	1.00	0.99	898
1	0.96	0.93	0.94	100
accuracy			0.99	998
macro avg	0.98	0.96	0.97	998
weighted avg	0.99	0.99	0.99	998

- XGBoost Confusion Matrix:

```
[[894   4]
[  7  93]]
```

Overall, the XGBoost model seems to be an excellent fit for this dataset.

Insights and Observations:

The XG Boost, Random Forest and Gradient Boosting models demonstrated superior performance compared to the SVM model, indicating that ensemble methods can be more effective for this dataset.
The dataset appears to have a class imbalance, with class 1 (accepting personal loan) having significantly fewer samples than class 0 (not accepting personal loan). This imbalance might have impacted the SVM model's performance in predicting class 1.
Further hyper parameter tuning and feature engineering could potentially improve model performance even more.

Based on the findings, I prefer either the XG Boost or the Gradient Boosting Classifier for predicting whether a customer will accept a personal loan. Both models have demonstrated high accuracy and balanced performance in predicting both classes.