Project Documentation: Down-sampling with RandomForestClassifier

Project Overview

This project focuses on addressing class imbalance in a dataset through **downsampling**. After balancing the classes, a **RandomForestClassifier** (with default settings) was trained. The model's performance was evaluated using **Accuracy**, **Recall**, and **F1 Score**.

Workflow Summary

1. **Data Preprocessing**
    - Loaded the dataset.
    - Identified class imbalance.
    - Performed **random downsampling** on the majority class.
2. **Model Training**
    - Used **RandomForestClassifier** from sklearn.ensemble.
    - Trained the model on the **downsampled** (balanced) dataset.
    - **No hyperparameter tuning** was performed — default parameters were used.
3. **Model Evaluation**
    - Evaluated the model using **Accuracy**, **Recall**, and **F1 Score**.

Downsampling Details

- **Technique**: Random downsampling of the majority class.
- **Goal**: Balance the number of samples between the majority and minority classes to prevent bias.

Model Details

- **Model Used**: RandomForestClassifier

- **Library**: scikit-learn
- **Hyperparameters**: Default values used.

Final Results

Metric	Score
Accuracy Score	0.9340
Recall Score	0.9337
F1 Score	0.9337

Conclusion

- **Downsampling** successfully balanced the dataset and improved model fairness.
- The **RandomForestClassifier** achieved strong performance without any hyperparameter tuning:
    - **Accuracy**: 93.40%
    - **Recall**: 93.37%
    - **F1 Score**: 93.37%
- Future work could explore advanced techniques like oversampling (SMOTE) or tuning the RandomForest parameters to push the performance even further.