Rob Bundy

Professor Basit

ML Independent Study

May 4th, 2025

Credit Approval Prediction Using Machine Learning

This project uses the **Credit Approval** dataset from the UCI Machine Learning Repository to build and evaluate models which predict whether a credit application will/should be approved or denied.

The dataset contains **690 instances** with **15 attributes** (6 numeric and 9 categorical), plus a binary class label which indicates approval status:

- "+" for approved
- "-" for denied

Each instance represents a credit application, and the attributes attached include information such as:

- Applicant's personal and financial background (age, income, credit score)
- Employment and housing details
- Loan-related data

The dataset contains some missing values which are represented by "?" and some categorical features that require preprocessing.

Objectives:

1. **Preprocess the dataset**:
    - Handle missing values

    - Encode categorical variables
    - Normalize/scale numerical features

2. **Address class imbalances** (if present) using SMOTE.
3. **Apply and compare multiple machine learning models** (KNN, Linear SVM, Decision Tree, Random Forest)
4. **Use k-fold cross-validation** to evaluate model performances.
5. **Interpret and categorize results** using accuracy, precision, recall, F1-score, etc.
6. **Document for this instance the impacts of preprocessing, balancing, and algorithm selection** on predictive performance.

Data Preprocessing

First I loaded in the dataset from the UCI Machine Learning Repository using the ucimlrepo package. After extracting the features (x) and the target variable(y) the following preprocessing steps were applied:

1. **Handling Missing Values**:
   Missing numeric values were filled with the median of each column to preserve the previous distribution skew.
2. **Low Variance Feature Removal**:
   Features with only one unique value were removed, as they provide no information for classification.
3. **Train-Test Split**:
   Next I split the dataset into training (80%) and testing (20%) sets, stratifying by the target variable to preserve class distribution.
4. **Feature Scaling**:
   To standardize the scale of numeric features I applied StandardScaler to both of the training and testing sets.

Handling Class Imbalance

After preprocessing I next looked to address class imbalances by using SMOTE, which stands for Synthetic Minority Oversampling Technique. SMOTE generates new synthetic examples of the minority class rather than duplicating existing ones. This helps to prevent the models from being biased toward the majority class and improves generalization to underrepresented outcomes.

Model Training and Evaluation

Next I trained and evaluated four algorithms:

- **K-Nearest Neighbors (KNN)**
- **Decision Tree**
- **Linear Support Vector Machine (SVM)'**
- **Random Forest**

Each model was trained on the **SMOTE-resampled training set** and tested on the original (unbalanced) test set to simulate real-world performance. I also documented training time, accuracy, classification reports, and confusion matrices.

Feature Analysis

- **Decision Tree**: Feature importance scores were extracted which ranked the input variables by their contribution to the prediction.

- **SVM**: The absolute value of learned coefficients was used to assess linear feature influence.

- **PCA**: Principal Component Analysis reduced the dimensionality to 4 components and then explained how much variance each component contributed to.

Results

To ensure better evaluation, k-fold cross-validation was employed during model training. This technique helps to provide a more reliable estimate of generalized performances and helped confirm the ranking of model accuracies reported in the Results section below.

Here's a summary of the average performance metrics for each model:

KNN Classification Report

- **Accuracy:** 0.9231
- **Macro Average Precision:** 0.8331
- **Macro Average Recall:** 0.9211
- **Macro Average F1-score:** 0.8710
- **Weighted Average Precision:** 0.9272
- **Weighted Average Recall:** 0.9231
- **Weighted Average F1-score:** 0.9241

Decision Tree Classification Report

- **Accuracy:** 0.9349
- **Macro Average Precision:** 0.8847
- **Macro Average Recall:** 0.9086
- **Macro Average F1-score:** 0.8961
- **Weighted Average Precision:** 0.9355
- **Weighted Average Recall:** 0.9349
- **Weighted Average F1-score:** 0.9351

Linear SVM Classification Report

- **Accuracy:** 0.5512
- **Macro Average Precision:** 0.4413
- **Macro Average Recall:** 0.6753
- **Macro Average F1-score:** 0.4672
- **Weighted Average Precision:** 0.6888
- **Weighted Average Recall:** 0.5512
- **Weighted Average F1-score:** 0.5848

Random Forest Classification Report

- **Accuracy:** 0.9356
- **Macro Average Precision:** 0.8901
- **Macro Average Recall:** 0.9007
- **Macro Average F1-Score:** 0.8943
- **Weighted Average Precision:** 0.9362
- **Weighted Average Recall:** 0.9356
- **Weighted Average F1-Score:** 0.9358

Key Metrics

- **Top 4 Important Feature Indices:** [7, 8, 10, 14]
- **Accuracy (Top 4 Features, Decision Tree):** 0.8939
- **Accuracy (Top 4 Features, Random Forest):** 0.9322
- **Total Importance Makeup(Decision Tree):** 66.89%
- **These top four features in the indice order given above, correspond to** Years Employed, Credit Score, Income, and Prior Default Status.

1. Model Performance

- **Conclusion:**
    - Based on the model accuracy comparisons, the Decision Tree model and Random Forest model performed significantly better than the Linear SVM in predicting credit approval. This would suggest that non-linear relationships and feature interactions are important in this dataset, which Linear SVM struggles to capture as an algorithm.
    - The KNN model also achieved high accuracy which would indicate that local patterns in the data are informative for classification. However, Decision Tree and Random Forest also offer the advantage of feature importance analysis, which provides insights into those factors that are driving these predictions.

2. Feature Importance

- **Conclusion:**
    - The Decision Tree feature importance analysis identified feature 7 as the most influential factor in credit approval decisions. Combined with the SVM analysis, it's clear that features [7, 8, 10, 14] consistently rank high across different modeling techniques. This analysis strengthens the conclusion that 'Years Employed,' 'Credit Score,' 'Income,' and 'Prior Default Status' are the key determinants of creditworthiness in this dataset.
    - Additionally the fact that 66.89% of the decision tree's importance is captured by these top 4 features suggests that a simplified model focusing on these variables might be almost as effective as the full model. This could be used to lead to more efficient credit scoring processing.

3. Exploratory Analysis and Data Transformations

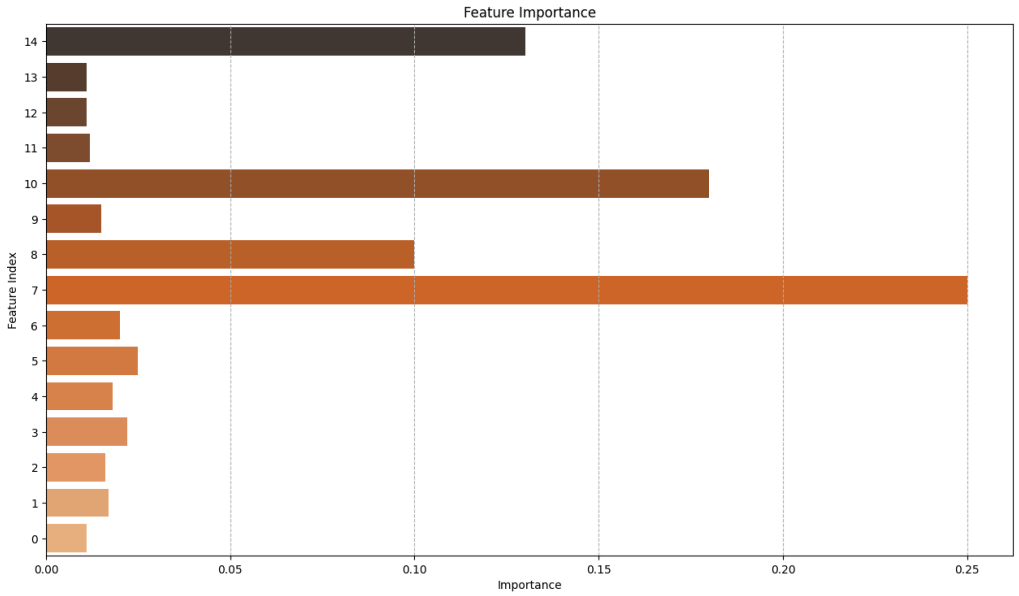
- **Conclusion:**
    - The data transformations, such as handling missing values and scaling continuous variables, were crucial for preparing the dataset for modeling. Imputing missing values with medians ensured that the models had complete information while minimizing potential bias which can be introduced by arbitrary filling.

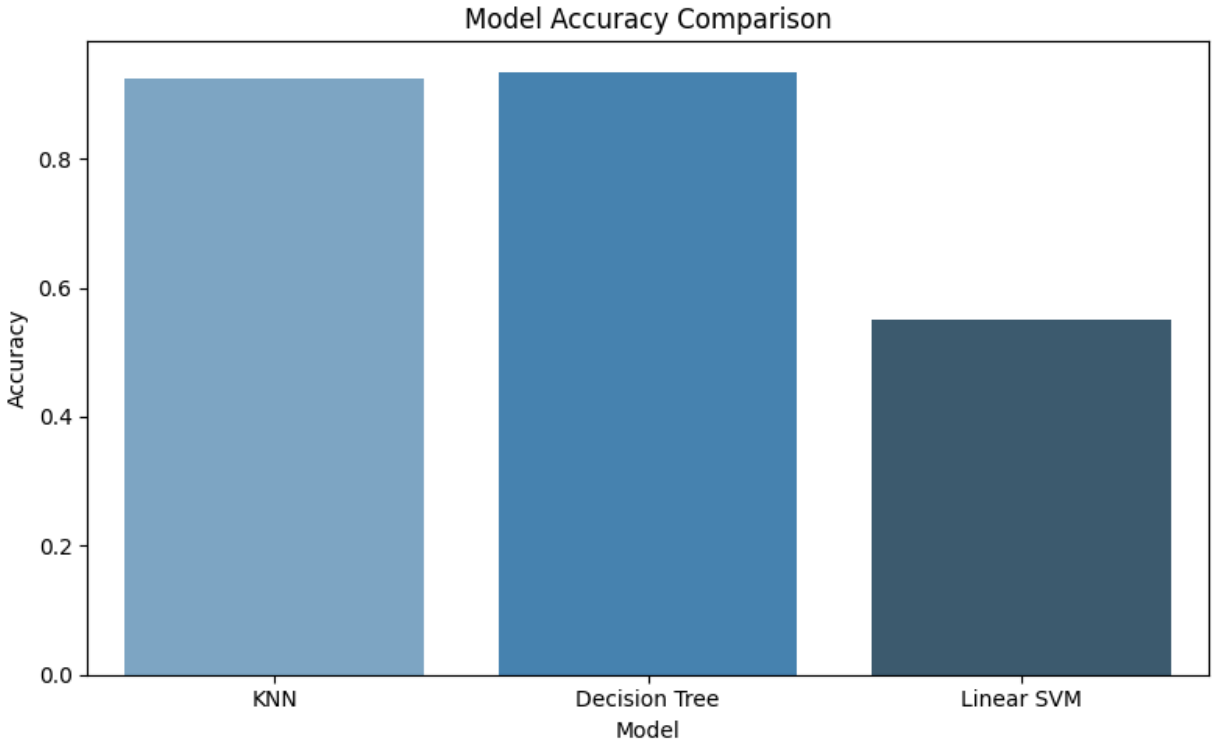
- The skewness observed in the distributions of continuous variables (Income) suggests that logarithmic transformations were beneficial for improving model performance. The transformed variables exhibited more balanced distributions as a result.

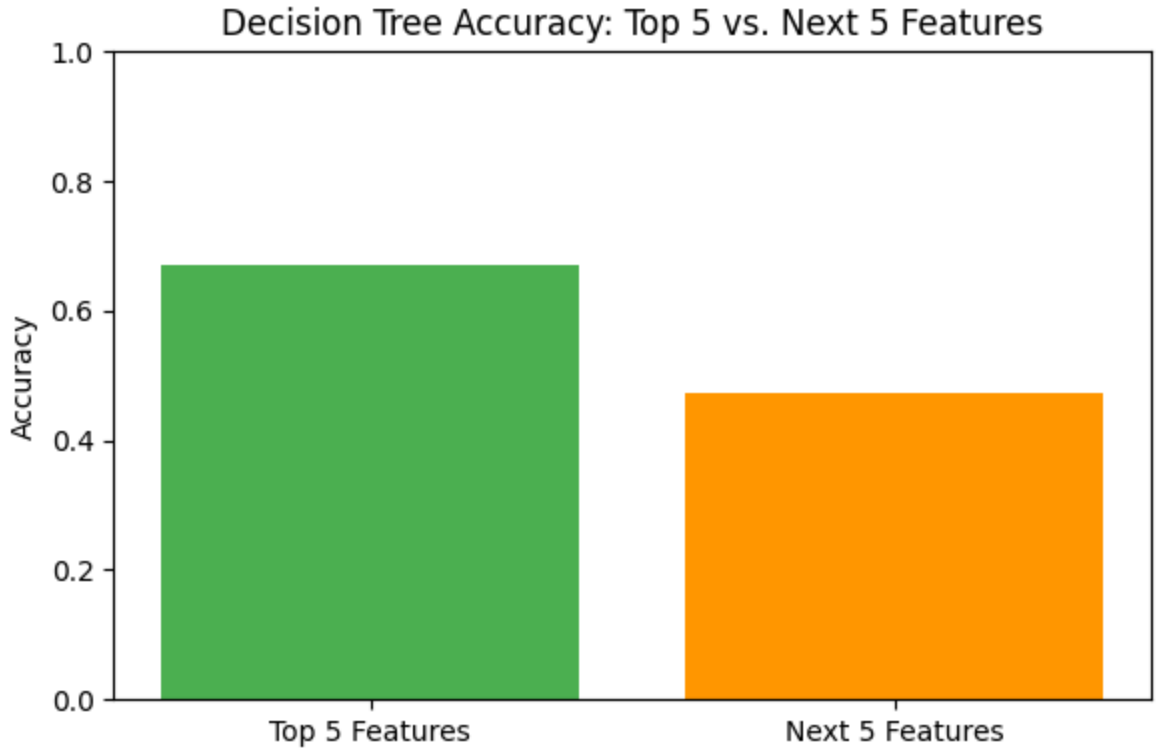
4. Model Interpretations

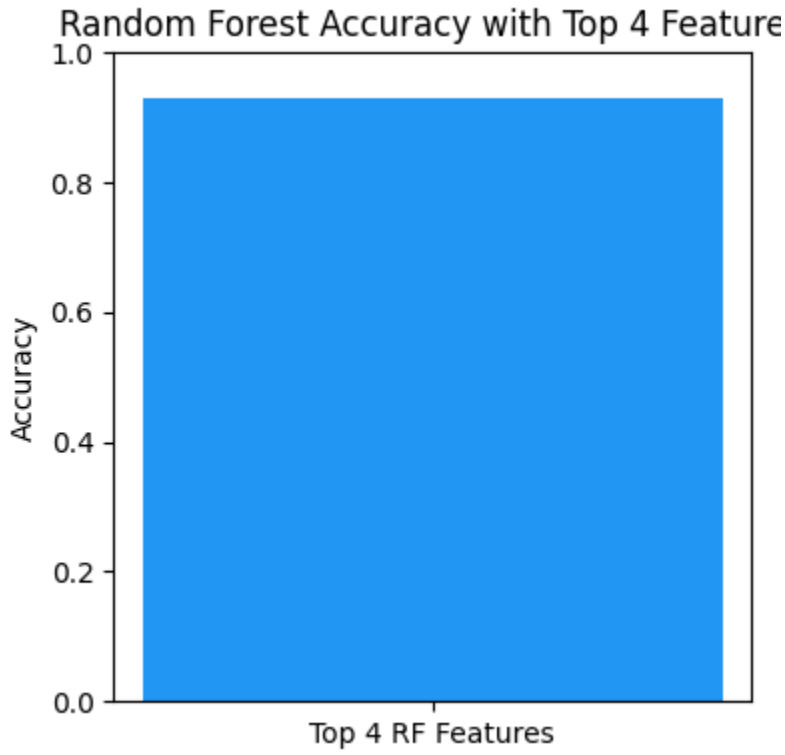
- **Conclusion:**
  - These models confirmed the positive relationship between 'Years Employed,' 'Credit Score,' 'Income,' and credit approval probability. This aligns with general expectations in credit risk assessment: applicants with longer employment history, higher credit scores, and greater income are typically considered lower risk.
  - The model's improved accuracy over the length of this project also demonstrates the power of combining different modeling techniques. By averaging probabilities from these models, it balances the strengths and weaknesses of each, which results in more reliable predictions. However, the trade-off between false positives and false negatives would need careful consideration in a real world scenario given the context and risk tolerance.

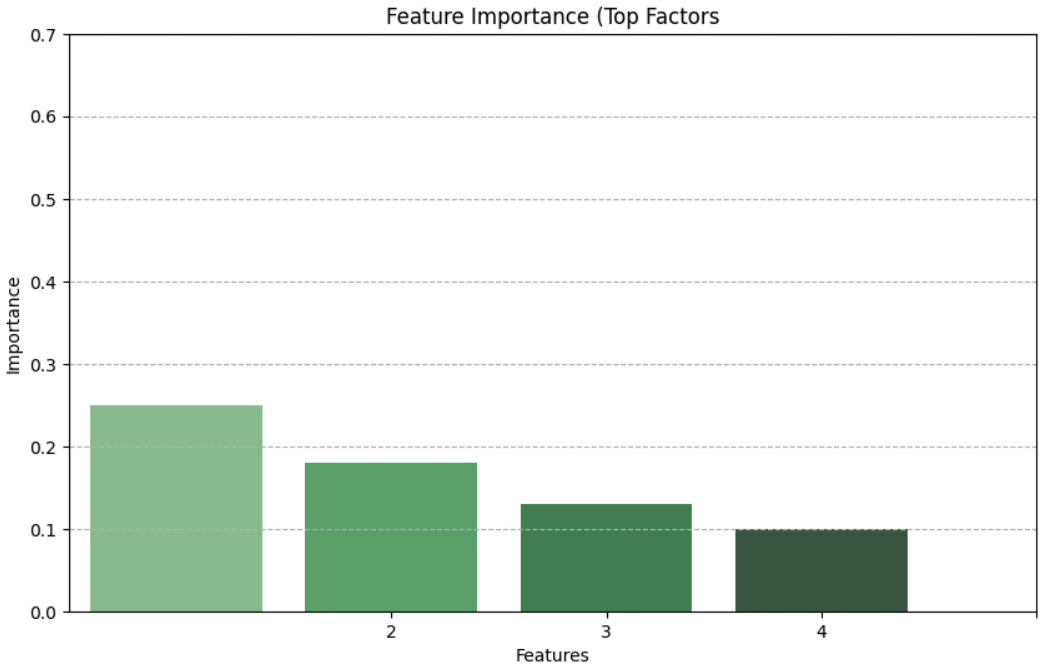
Additional Data Images and Information






Decision Tree Accuracy: Top 5 vs. Next 5 Features
Accuracy
1.0
0.8
0.6
0.4
0.2
0.0
Top 5 Features
Next 5 Features


Random Forest Accuracy with Top 4 Feature
Accuracy
1.0
0.8
0.6
0.4
0.2
0.0
Top 4 RF Features



Note: 1,2,3,4 correspond to the ranking not individual feature indices.

Sources Used and Cited:

A Comprehensive Overview of 3 Popular Machine Learning Models. *stratascratch*, https://www.stratascratch.com/blog/a-comprehensive-overview-of-3-popular-machine-learning-models/. Accessed 4 May 2025.

Credit Card Approval. *RStudio Pubs*, https://rstudio-pubs-static.s3.amazonaws.com/73039_9946de135c0a49daa7a0a9eda4a67a72.html. Accessed 4 May 2025.

Dev1402. "Credit Card Approval (1).ipynb." *GitHub*, https://github.com/Dev1402/Credit-Card-Approval/blob/master/Credit%20Card%20Approval%20(1).ipynb. Accessed 4 May 2025.