Financial Inclusion in East Africa's 4 Countries

Group 7 project final report

IST 707 Applied Machine Learning

By

Kartik Kaul | kkaul01@syr.edu

Jaineel Parmar | japarmar@syr.edu

Shan Jiang | sjiang41@syr.edu

Chuan Tse Tsai | ctsai04@syr.edu

Introduction

The lack of financial inclusion is a major obstacle to economic and human development in some countries of East Africa. Only a small percentage of adults in Kenya, Rwanda, Tanzania, and Uganda have access to or use commercial bank accounts. Although mobile money and fintech solutions have expanded, banks are still important for providing financial services, including savings, payments, and credit. To promote long-term economic growth, it is crucial to develop machine learning models that can predict who is most likely to have or use a bank account. These models can provide insights into the current state of financial inclusion and the factors that drive financial stability.

Scope: 4 countries of East Africa - Kenya, Rwanda, Tanzania, and Uganda

Objective:

The goal of this project is to develop a machine learning model that can identify those people who are most likely to possess or use a bank account. The model will provide insights into factors affecting financial security and measure financial inclusion in Kenya, Rwanda, Tanzania, and Uganda. The project uses Python and classification techniques such as Support Vector Machines (SVMs), Random Forest, and kNN.

Research Questions:

Research Question 1: What are the demographic and socioeconomic factors that are most predictive of bank account ownership and usage in Kenya, Rwanda, Tanzania, and Uganda?

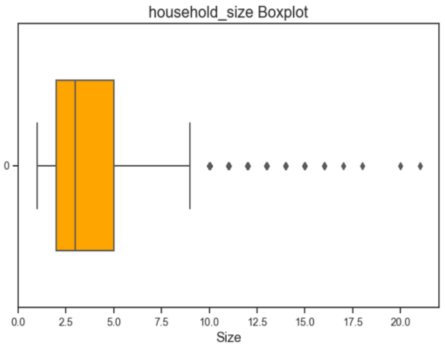
Research Question 2: How well do various machine learning algorithms, including Support Vector Machines (SVMs), Random Forest, and kNN, perform in predicting bank account ownership and usage in East Africa?

Data Description:

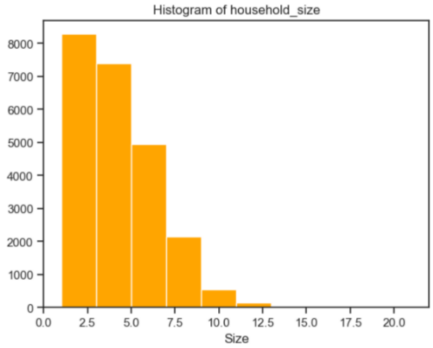
The main dataset contains demographic information and what financial services are used by approximately 23524 individuals across 4 countries of East Africa. This data was extracted from various Finscope surveys in 2016-2018. After checking, there is no missing value in the dataset.

1. Checking outlier for numeric value

The box plot in Graph 1 shows outliers in the household_size variable with values above 10 but removing them is not feasible as they could be individuals with big or joint families. The histogram in Graph 2 shows that most individuals (over 5000) have a family size of 2 or 3. The data is right skewed with the peak on the left side and mean household_size being greater than the median household_size. Therefore, we plan to include outliers and evaluate their impact on machine learning performance.

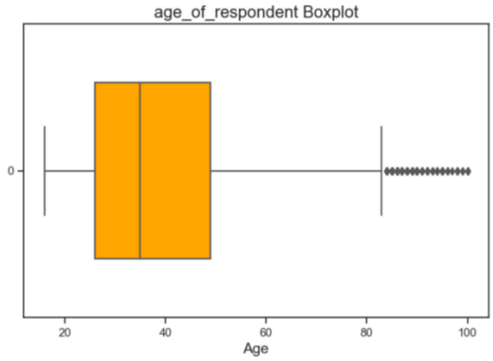


Graph 1: household_size Boxplot

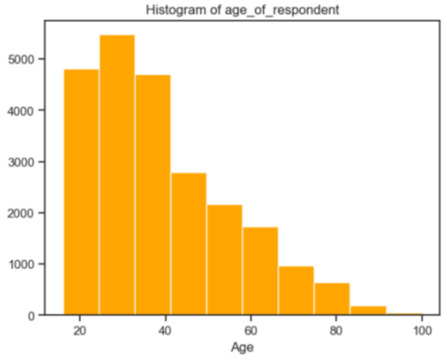


Graph 2: household_size

Based on the Graph 3 and Graph 4, the age_of_respondent variable has outliers with values above 80, but they can't be removed as they could represent elderly respondents. Most individuals have an age between 35-40, and the data is right skewed. The inclusion of outliers is necessary to avoid skewness, and their impact on machine learning performance needs to be evaluated.



Graph 3: age_of_respondent Boxplot



Graph 4: Histogram age_of_respondent

2. Checking multicollinearity problem

The dataset has 13 variables as shown in Table 1. It includes different categories of employment status such as self-employed, dependent (with subcategories of government-dependent and remittance-dependent), formally employed (with subcategories of government and private), informally employed, other income, and no income. The income may vary depending on factors such as the success of their business, government benefits, or income from family members living abroad. The dataset also includes variables such as household size, age, gender, and education level.

Table 1: Data Description

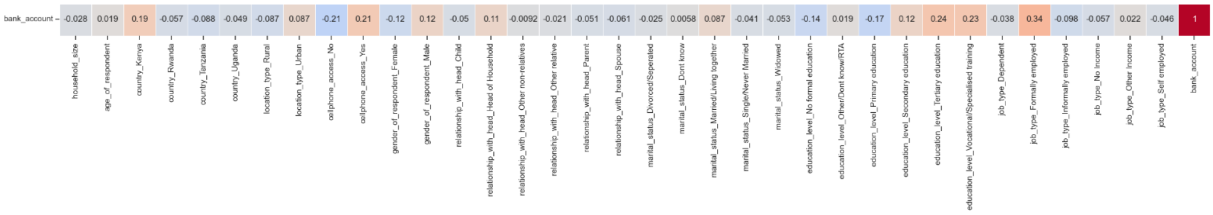
Variable Definitions	
country	Country interviewee is in.
year	Year survey was done in.
uniqueid	Unique identifier for each interviewee
location_type	Type of location: Rural, Urban
cellphone_access	If interviewee has access to cellphone: Yes, No
Household_size	Number of people living in one house

age_of_respondent	The age of the interviewee
gender_of_respondent	Gender of interviewee: Male, Female
Relationship_with_head	The interviewee's relationship with the head of house: Head of Household, Spouse, Child, Parent, Other relative, Other non-relatives, Don't know
marital_status	The martial status of the interviewee: Married/Living together, Divorced/Separated, Widowed, Single/Never Married, Don't Know
education_level	Highest level of education: No formal education, Primary education, Secondary education, Vocational/Specialized training, Tertiary education, Other/Don't know/RTA
job_type	Type of job interviewee has: Self employed, Dependent, Formally employed, Informally employed, Other Income, No Income

Data Preparation

1. Removing unwanted columns
To prepare the data for the machine learning model, the input features are separated from the target variable 'bank_account'. The input features are stored in a variable called X, while the target variable is stored in a variable called y. To make sure that the input features are relevant, three columns, namely 'bank_account', 'uniqueid', and 'year', are removed from X. This is because these columns do not provide any useful information for predicting the target variable.
2. Preparing train and test dataset
We divided the original data into two parts: the train dataset with 75% of the data and the test dataset with 25% of the data. The train dataset has 17643 examples with 5 features, while the test dataset has 5881 records.
3. Handling imbalanced classes (resampling):
We have performed resampling using resample method in sklearn.utils package as the classes are imbalanced. So, to avoid bias we will make records representing both the classes equally.

Descriptive Analysis



Graph 5: Mixed Correlation Matrix Heatmap with bank account

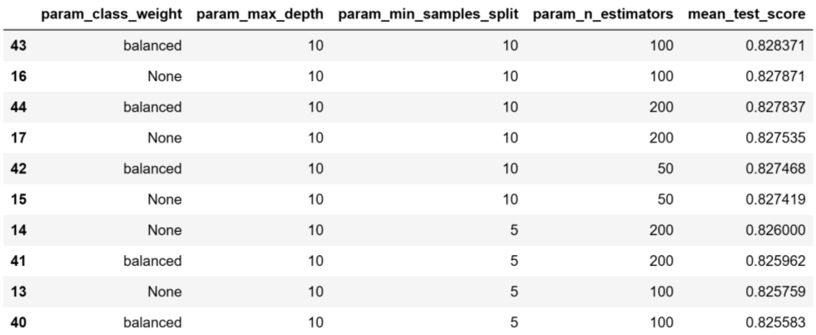
The Mixed Correlation Matrix Heatmap (Graph 5) shows that the job_type_Formally employed has the highest correlation with the bank account, with a correlation value of 0.34. This may be because people with formal employment have a steady income, and their wages are usually paid directly into a bank account. On the other hand, there is a negative correlation of -0.21 between the person who does not have cellphone access and bank account. Nowadays, mobile phones are essential for daily life, and people without access to them may have poor financial situations, which means they are less likely to have a bank account.

Modeling:

1. Random Forest

Table 2 displays the results of a hyperparameter tuning process for a random forest classifier model. The hyperparameters that were tuned are class_weight, max_depth, min_samples_split, and n_estimators. The table lists the top 10 parameter combinations that achieved the highest accuracy scores.

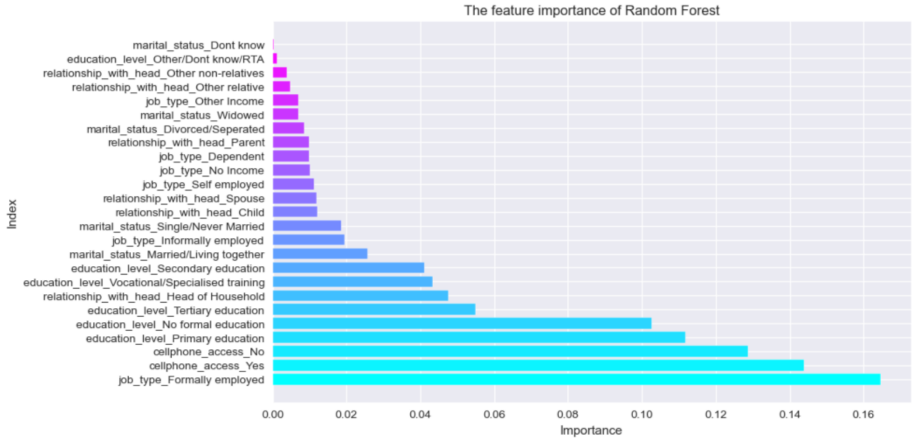
Table 2: Top 10 parameters based on accuracy of Random Forest



	param_class_weight	param_max_depth	param_min_samples_split	param_n_estimators	mean_test_score
43	balanced	10	10	100	0.828371
16	None	10	10	100	0.827871
44	balanced	10	10	200	0.827837
17	None	10	10	200	0.827535
42	balanced	10	10	50	0.827468
15	None	10	10	50	0.827419
14	None	10	5	200	0.826000
41	balanced	10	5	200	0.825962
13	None	10	5	100	0.825759
40	balanced	10	5	100	0.825583

In Table 2, the best hyperparameter combination is identified as index 43, with a "balanced" class_weight, max_depth of 10, min_samples_split of 10, and 100 estimators, resulting in a mean cross-validated score of 0.828371, the highest among all combinations tested.

The feature importance of Random Forest graph (Graph 6) shows the most important feature for predicting the target variable is 'job_type_Formally employed' with an importance score of 0.164583, followed by 'cellphone_access_Yes' with a score of 0.143942 and 'cellphone_access_No' with a score of 0.128659.

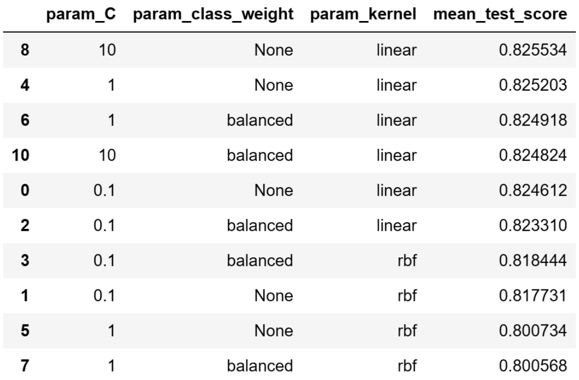


Graph 6: The feature Importance of Random Forest

2. SVM

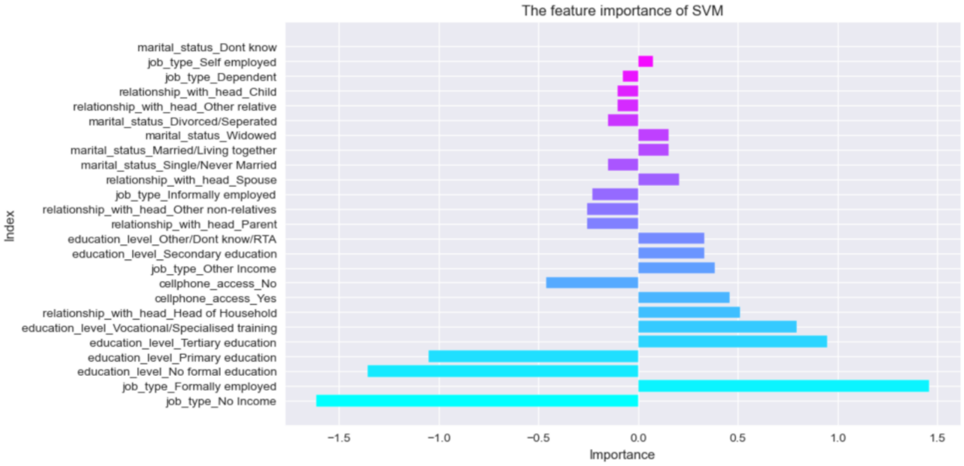
Table 6 displays the outcomes of a hyperparameter tuning process for a classification model. The evaluated parameters include param_C, param_class_weight, and param_kernel. The mean_test_score column represents the mean score of the model on the test set for each parameter combination. The top 10 parameters are listed in the table based on accuracy.

Table 3: Top 10 parameters based on accuracy of SVM



	param_C	param_class_weight	param_kernel	mean_test_score
8	10	None	linear	0.825534
4	1	None	linear	0.825203
6	1	balanced	linear	0.824918
10	10	balanced	linear	0.824824
0	0.1	None	linear	0.824612
2	0.1	balanced	linear	0.823310
3	0.1	balanced	rbf	0.818444
1	0.1	None	rbf	0.817731
5	1	None	rbf	0.800734
7	1	balanced	rbf	0.800568

To represent feature importance in SVM model, we use coefficient values, which can be seen in Graph 7. The important features for predicting the target variable are job_type_Formally employed, education_level_No formal education, and education_level_Primary education, while marital_status_Dont know has no impact on the target variable with a coefficient value of zero.

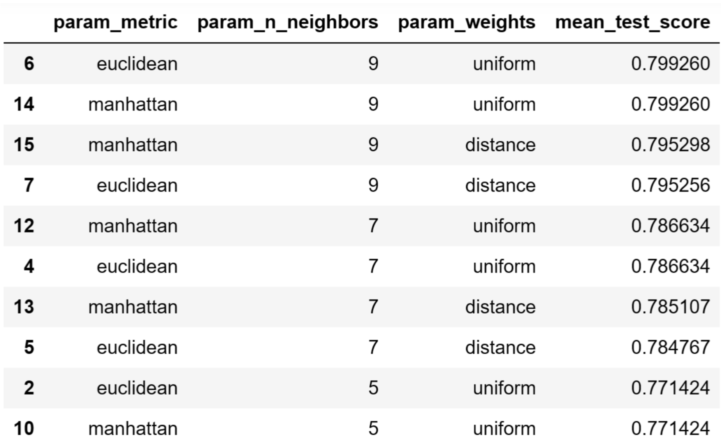


Graph 7: The feature Importance of SVM

3. kNN

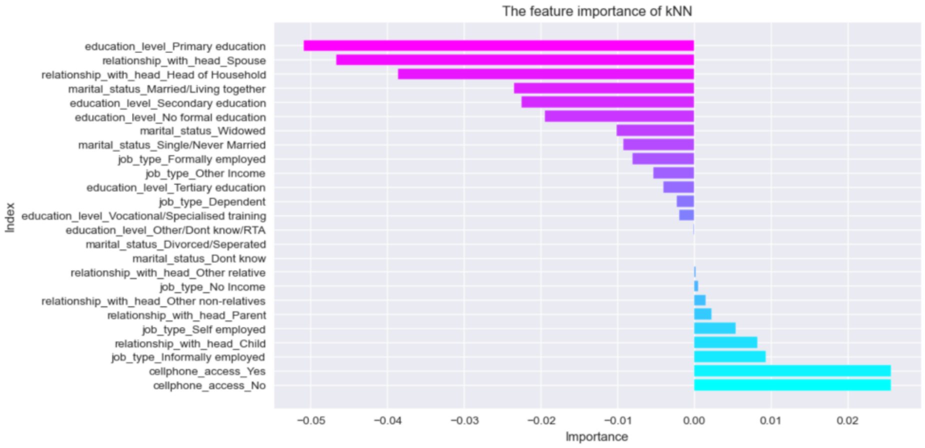
Table 4 shows the performance of a K-nearest neighbors (KNN) model with different hyperparameters. The model was evaluated based on mean test score, which measures its accuracy on unseen data. The hyperparameters that were adjusted are param_metric, param_n_neighbors, and param_weights. The table lists the top 10 parameters based on accuracy.

Table 4: Top 10 parameters based on accuracy of kNN



	param_metric	param_n_neighbors	param_weights	mean_test_score
6	euclidean	9	uniform	0.799260
14	manhattan	9	uniform	0.799260
15	manhattan	9	distance	0.795298
7	euclidean	9	distance	0.795256
12	manhattan	7	uniform	0.786634
4	euclidean	7	uniform	0.786634
13	manhattan	7	distance	0.785107
5	euclidean	7	distance	0.784767
2	euclidean	5	uniform	0.771424
10	manhattan	5	uniform	0.771424

The KNN model results suggest that the best hyperparameter combination is using either the Euclidean or Manhattan metric with 9 neighbors and uniform or distance weights. The importance scores in Graph 8 indicate that cellphone access is the most important feature for predicting the target variable, while features like 'education_level_Primary education' have negative importance scores, meaning they are less important for prediction. However, it's worth noting that all features contribute to the model's performance, and the importance scores are a relative measure of each feature's contribution, not an absolute measure of their predictive power.

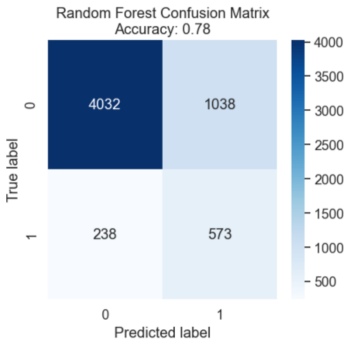


Graph 8: The feature Importance of kNN

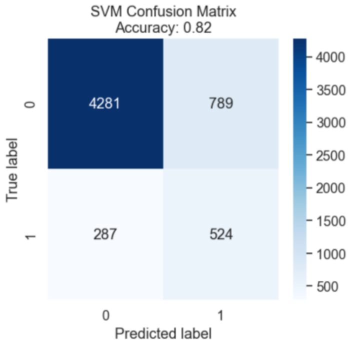
Prediction Performance Comparison:

1. With all features:

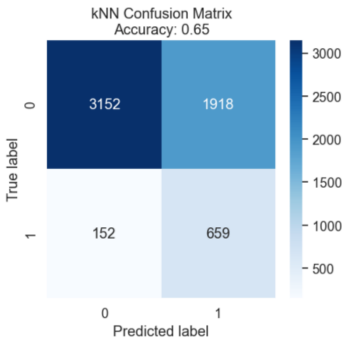
Three models are applied to predict the test data to identify which person has bank account. SVM has the highest prediction performance than other two models with 0.82 of the accuracy.



Graph 9: Random Forest Confusion Matrix



Graph 10: SVM Confusion Matrix



Graph 11: kNN Confusion Matrix

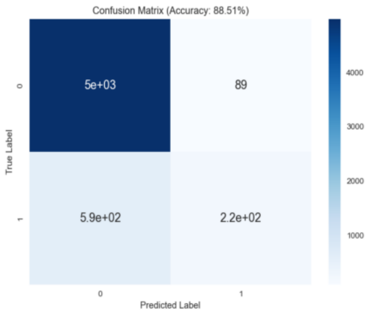
The Table 5 indicates that all models perform better at predicting class 0 than class 1. The Random Forest and SVM models have high precision scores for class 0 but lower recall scores for class 1. The kNN model has a high precision score for class 1 but a low recall score for class 0. Therefore, the Random Forest and SVM models may be better at predicting class 0, while the kNN model may be more appropriate for predicting class 1. Overall, based on these results, the SVM model has the highest prediction performance.

Table 5: Classification Report with all features

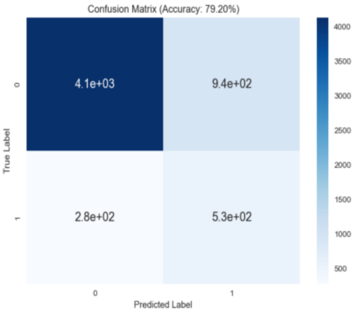
	Precision(0)	Recall(0)	Precision(1)	Recall(1)
Random Forest	0.94	0.80	0.36	0.71
SVM	0.94	0.84	0.40	0.65
kNN	0.95	0.62	0.26	0.81

2. **With the top 15 features:**

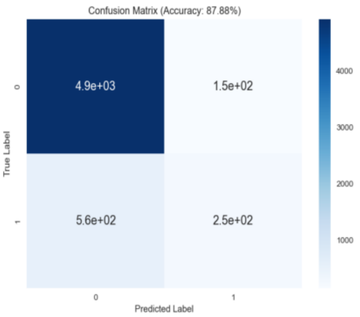
Based on the feature importance, we found that many variables affect a model's performance, so we selected the top 15 features for each model to improve it. After this process, the accuracy of the Random Forest and kNN models increased significantly, but for the SVM model, the prediction performance declined, and it did not show much improvement compared to before.



Graph12: Random Forest Confusion Matrix with top 15 features



Graph13: SVM Confusion Matrix with top 15 features



Graph14: kNN Confusion Matrix with top 15 features

The table 6 shows that the kNN model with the top 15 features achieved high precision scores of 0.90 and 0.62 for class 0 and class 1, respectively, as well as high recall scores of 0.97 and 0.31 for the same classes. Therefore, it can be concluded that the kNN model is the most suitable for the classification task using the top 15 features.

Table 6: Classification Report with top 15 features

	Precision(0)	Recall(0)	Precision(1)	Recall(1)
Random Forest	0.89	0.98	0.72	0.28
SVM	0.94	0.81	0.36	0.65
kNN	0.90	0.97	0.62	0.31

Concluding remarks

Based on this report, we can conclude the following:

- The most important features for predicting poverty levels are cellphone access and job type, while the least important features are education level and relationship with head of household. Cellphone access is particularly important because it reflects the use of mobile banking, which implies that the individual has a bank account.
- Comparing the model performance using all features and top 15 features, we observed that the Random Forest and kNN models showed a significant increase in accuracy. However, the performance of the SVM model declined and remained similar to before. Therefore, it can be concluded that selecting important features can improve the performance of some models significantly, but not all. The selection of feature selection technique and the number of features to be selected may vary depending on the model and dataset.
- In summary, the kNN model with top 15 features outperforms the other two models, as it has a higher recall score for non-poor individuals and relatively high precision scores for both non-poor and poor individuals.

References

Rao, A. (2015, February 19). Which one to use: Random Forest vs. SVM vs. KNN. Analytics Vidhya.
https://discuss.analyticsvidhya.com/t/which-one-to-use-randomforest-vs-svm-vs-knn/2897

scikit-learn: Machine Learning in Python. (n.d.). scikit-learn: Machine Learning in Python — scikit-learn 1.0 documentation. Retrieved April 28, 2023, from
https://scikit-learn.org/stable/

Zindi Competition:

Zindi. “Financial Inclusion in Africa.” Zindi, 2021,
https://zindi.africa/competitions/financial-inclusion-in-africa/data.

Analytics Vidhya Discussion:

Analytics Vidhya. “Which One to Use – RandomForest vs SVM vs KNN?” Analytics Vidhya, 2016,
https://discuss.analyticsvidhya.com/t/which-one-to-use-randomforest-vs-svm-vs-knn/2897.