Problem Set 2[*][†]

Sergio Sandoval[‡] Maria Fernanda Blanco
Juan Gutierrez Juan Fernando Barberi

"Wars of nations are fought to change maps. But wars of poverty are fought to map change"

– M. Ali

1 Introduction

The accurate measurement of poverty is an ongoing challenge for economists and policymakers, as it is crucial for the design of effective policies to alleviate poverty. Traditional data collection methods, such as household surveys, are resource-intensive and time-consuming, often resulting in significant delays before actionable insights can be obtained. In response to this challenge, organizations like the World Bank have proposed the use of predictive analytics and machine learning techniques to improve the efficiency and accuracy of poverty measurement. This approach aims to not only make the process faster and more cost-effective but also to provide more precise targeting of interventions for populations most in need. Competitions like the Pover-T Test: Predicting Poverty reflect this shift towards leveraging advanced analytics to improve our understanding and measurement of poverty, with the ultimate goal of supporting the development of better public policies.

The focus of this project is on predicting poverty at the household level in Colombia, using data from the National Administrative Department of Statistics (DANE) and the "Empalme de las Series de Empleo, Pobreza y Desigualdad" (MESE) mission. The primary objective is to build a model capable of accurately classifying households as either poor or non-poor, with a particular emphasis on using the fewest number of variables possible without sacrificing predictive accuracy. The dataset provides rich socio-economic information at both the household and individual levels, allowing for the construction of additional features that capture household composition and the characteristics of individual members, such as education level, employment status, and demographic variables. In this case, poverty is defined as a binary classification problem, where a household is considered poor if its income falls below a specified poverty line.

Over the past several years, numerous studies have explored the application of machine learning techniques for poverty prediction. One notable example is the work of Jean et al. (2016), who used high-resolution satellite images combined with machine learning to predict poverty levels in sub-Saharan Africa. This study highlights the potential of predictive models to serve as alternatives to traditional survey methods, especially in regions where data collection is expensive or infeasible. In Latin America, a study by Muneton and Manrique (2023) employed machine learning algorithms such as Random Forest and CatBoost to predict multidimensional poverty in Medellin, Colombia. This study, which utilized spatial data from open-source platforms like OpenStreetMap, demonstrates the ability of machine learning models to predict poverty at a granular level, emphasizing the importance of spatial factors in the analysis of socio-economic conditions.

The dataset used in this analysis offers a unique opportunity to test the efficacy of predictive models in classifying households by poverty status. It consists of both training and test sets, which include a variety of features at both the household and individual levels. For the purposes of this exercise, the models are tasked with predicting whether a household falls below the poverty line, without access to direct income information in the test set. Instead, the models rely on socio-demographic indicators such as education, employment, household composition, and housing conditions to make their predictions. This scenario mirrors real-world situations where policymakers must make decisions with incomplete or indirect information, underscoring the relevance of the dataset and the models to the broader goal of improving poverty measurement.

[*] Big Data and Machine Learning, Universidad de los Andes.

[†] Excluding tables and figures, the document is less than 10 pages.

[‡] Link of the repository: https://github.com/setosandoval/BDML_Team1_2024.git

The results of the analysis indicate that three models stood out as top performers: Logit, Random Forest, and XGBoost. After rigorous cross-validation and parameter tuning, these models were submitted to Kaggle for evaluation. The XGBoost model emerged as the best overall performer, achieving an F1-score of 0.675 on Kaggle and a score of 0.67 in the sub-testing set. Random Forest followed closely with an F1-score of 0.659, while Logit achieved a respectable score of 0.637. Each model benefited from the application of SMOTE (Synthetic Minority Over-sampling Technique) to address class imbalance in the training data, as well as the optimization of classification thresholds to maximize the F1-score.

Several factors contributed to the strong performance of XGBoost. Its sequential tree-building process allowed it to iteratively correct classification errors, enabling it to capture complex interactions between variables. The model's ability to adjust the classification threshold further enhanced its predictive power, particularly by improving recall, which is critical for minimizing false negatives in poverty prediction. This was especially important in the context of this analysis, where the goal was to identify as many poor households as possible to ensure targeted interventions. Random Forest, while performing slightly worse than XGBoost, also demonstrated strong generalization across different datasets. Its ensemble method, which constructs multiple decision trees and averages their predictions, made it robust against overfitting and capable of handling the high-dimensional data used in this analysis. However, the independent voting mechanism of Random Forest limited its ability to capture the same level of complexity as XGBoost, leading to slightly lower overall performance. Logit, though outperformed by the ensemble methods, still provided valuable insights into the relationships between individual features and poverty status. Its interpretability and simplicity make it an attractive baseline model, even though it struggled to capture the complex interactions that were well-handled by the more sophisticated models.

In conclusion, this analysis demonstrates the potential of machine learning techniques for poverty prediction, with XGBoost emerging as the top-performing model. The results suggest that models capable of capturing non-linear relationships and interactions between variables, such as XGBoost and Random Forest, are particularly well-suited for complex socio-economic datasets. Moving forward, future research could focus on further improving these models through enhanced feature engineering, hyperparameter tuning, and the application of more advanced ensemble techniques. Additionally, exploring the integration of alternative data sources, such as satellite imagery or mobile phone data, could offer new avenues for improving the accuracy and timeliness of poverty predictions, helping to better inform public policies aimed at reducing poverty in Colombia and beyond.

2 Data

2.1 Description and Purpose

The data used in this analysis comes from the mission for the "Empalme de las Series de Empleo, Pobreza y Desigualdad" (MESE), a collaboration between DANE and the National Planning Department (DNP) of Colombia. This initiative was established to address inconsistencies in labor market and poverty statistics caused by the transition from the Continuous Household Survey (ECH) to the Gran Encuesta Integrada de Hogares (GEIH). As part of its broader mandate, MESE developed a new methodology for measuring monetary poverty in Colombia, improving the comparability of income data and poverty estimates across time.

The data employed in this study is divided into training and testing sets, both at the household and individual levels. While the training data contains detailed socio-economic characteristics such as income, household composition, and access to basic services, the testing data is intentionally limited, with missing values across several variables and the complete absence of income-related variables. Additionally, the testing data does not include households from BogotÃ¡, adding another layer of complexity to the prediction task. This constraint was designed to make poverty prediction more challenging, requiring the model to rely on other socio-demographic features to make accurate predictions. At the household level, the data includes attributes that capture aspects of housing conditions, family size, and household structure, from which new variables can be constructed.

Similarly, individual-level data provides insights into characteristics such as education, social security coverage, employment status, and age distribution. These individual attributes can be aggregated or expressed as proportions within the household, offering flexible ways to model household-level outcomes. Moreover, the head of household's characteristics can be isolated to create additional predictive indicators. Both datasets also contain information on geographical location, which can be used to construct variables that capture regional effects on poverty.

The ability to derive such variables from the available data, despite the absence of direct income information in the testing set, underscores the relevance of this dataset for addressing the problem posed: predicting household poverty. Since the goal is to estimate the poverty status of households, it is important to understand how this status is constructed. In this case, a household is classified as poor if its per capita income, adjusted for household size and composition, falls below a threshold known as the poverty line. This threshold varies by household and is influenced by factors such as the number of individuals in the household and their geographic location. As a result, it is crucial to construct variables that approximate these factors, such as housing strata in Colombia, which serves as a proxy for socio-economic status. By exploiting a range of socio-economic and demographic indicators, it is possible to develop robust models that not only classify households by poverty status but also provide deeper insights into the factors driving these outcomes, allowing for accurate prediction and comprehensive analysis.

2.2 Data Cleaning

The dataset consists of four main data sources: two for training and two for testing, at both the household and individual levels. The primary focus of this analysis is on the household datasets, as the objective is to predict poverty at the household level. Variables from the household datasets are directly used in the model, while individual-level variables are aggregated at the household level. Additionally, variables specific to the head of the household are included to capture relevant household dynamics. New variables were constructed based on these aggregations, and in some cases, missing values were imputed to ensure data completeness. It is important to note that individual-level data contains a high proportion of missing values, which was considered when deciding which variables to include. Only variables common to both the training and testing datasets were selected, as the model cannot be trained on variables that are absent in the testing set. After this process, the household datasets were reduced to 63 variables, while the individual datasets were reduced to 15 variables. The training household dataset contains one additional variable -poverty status- indicating whether a household is classified as poor or not. All datasets contain an ID, which allows for merging individual-level information into the household datasets. The final household training dataset contains 164,960 observations, while the household testing dataset has 66,168 observations. Although the individual datasets contain many more observations, the primary interest lies in the household datasets, as they are used for the poverty prediction task.

At the end of the data cleaning process, two household datasets were constructed for training and testing, each containing 34 variables and with the same number of observations as mentioned before. The variables are summarized in *Table 1.* Importantly, after the data cleaning and imputation processes, no missing values remained in any of the variables. Now, first we focus on the household-level variables used for the prediction task. One of the key variables is the domain, which indicates whether a household is located in a major city or a rural area-an essential factor for understanding regional poverty dynamics. Household size, measured both by the total number of individuals and as consumption units, was also included, as larger households may face greater financial strain. Overcrowding was captured by the number of people per room, a common measure of housing conditions. Additionally, the poverty line was incorporated for each household, providing a crucial threshold for classifying poverty status. Expansion factors at both the departmental and household levels were also included to ensure proper weighting in the analysis. A categorical variable describing housing tenure was added, distinguishing between ownership, mortgage, rental, and other arrangements. Among these, rental status is particularly relevant, as it often correlates with financial vulnerability. Importantly, none of these household-level variables contained missing values. Finally, a key constructed variable was rental price. While the dataset initially included a variable for actual rent, it had missing values, particularly for households that were not renting. To address this, the hypothetical rent households would pay if they rented their home was used to fill gaps. A final rent variable was created by combining actual rent for those renting and hypothetical rent for non-renting households. This final variable had no missing values and was crucial for capturing housing-related financial strain.

Source	**Variable**	**Description**
Household	dom	City or rural classification of the household
	p_room	Number of persons per room in the household
	kind	Type of housing ownership: owned, rented, no property
	rent	Monthly rent price, or hypothetical rent if missing
	num_per	Total number of people in the household
	num_per_u	Number of people in the expenditure unit
	pl	Poverty line indicator for the household
	fex_c	Household expansion factor
	fex_dpto	Expansion factor at the department level
Individual	num_female	Total number of females in the household
	mean_age	Average age of household members
	num_minor	Total number of minors in the household
	num_old	Total number of elderly people in the household
	num_s_sec	Number of people with social security in the household
	max_educ	Maximum education level in the household -
	subs	Total number of subsidies received by the household
	num_unemp	Total number of unemployed individuals
	num_inact	Total number of inactive individuals in the household
	sum_fex_c_p	Sum of expansion factor for each individual in the household
	head_female	1 if the household head is female
	head_age	Age of the household head
	head_old	1 if the household head is over 60 years old
	head_s_sec	1 if the household head has social security
	head_educ	Maximum education level of the household head -
	head_unemp	1 if the household head is unemployed
	head_inact	1 if the household head is inactive
	head_sub	1 if the household head receives subsidies
	prop_female	Proportion of females in the household
	prop_minor	Proportion of minors in the household
	prop_old	Proportion of elderly people in the household
	prop_s_sec	Proportion of people with social security
	prop_subs	Proportion of individuals receiving subsidies
	prop_unemp	Proportion of unemployed individuals in the household
	prop_inact	Proportion of inactive individuals in the household

Table 1. Variables Description.

To incorporate individual-level information into the household dataset, the person-level data was first modified to identify key characteristics. Specifically, we created indicators for whether a person was the head of the household, a woman, under 15 years old, over 60 years old, had social security coverage, their employment status (unemployed or inactive), and their highest level of education. Missing values for social security were imputed as no coverage, and for education, if missing, it was assumed the individual had no formal education. Additionally, we summed the number of subsidies each person received using four subsidy-related variables, assuming a value of zero if missing. The expansion factor at the individual level was also included to ensure proper weighting in the aggregation process. These variables capture key socio-demographic characteristics essential for poverty prediction, as aspects like social security coverage, education level, and employment status are directly linked to household economic well-being. The imputations made for missing values were justified based on their logical assumptions-for example, assuming no social security or education when missing reflects likely conditions in low-income populations.

Once these individual-level variables were constructed, they were aggregated to the household level to derive summary statistics. These included the total number of women, children (under 15), elderly (over 60), individuals with social security coverage, unemployed, inactive individuals, and the total number of subsidies received by the household. The maximum education level in the household, the household's average age, and the total expansion factor were also computed. Additionally, proportions were calculated for key groups, such as the proportion of women, children, elderly, subsidy recipients, unemployed, and inactive individuals within the household. These proportions allow for a more nuanced analysis, as they highlight the household's internal composition relative to its size. Finally, variables specific to the head of the household, whether they were a woman, elderly, had social security, received subsidies, were unemployed or inactive, and their education level-were also included, as the characteristics of the head often play a significant role in household economic decisions. By incorporating these variables, both in absolute and proportional terms, the model gains a more comprehensive understanding of household dynamics, which is crucial for predicting poverty status.

2.3 Descriptive Statistics

In this analysis, not all variables are included in the descriptive statistics section for several reasons. The most important reason is that only the variables essential for building the classification models are selected, as they form the main objective of the study: predicting poverty at the household level. This approach allows for a clearer and more precise understanding of the patterns and relationships that directly influence the outcomes of the models, avoiding distractions from less relevant variables. Additionally, there are practical reasons that justify this decision. First, the space limitations of the document make it impossible to perform a comprehensive analysis of all variables. Second, including all variables would not provide significant value to the study, as many do not contribute to the prediction objective and their inclusion could overwhelm the reader with unnecessary information, complicating the overall interpretation of the analysis. Therefore, an efficient and precise selection of variables was made, respecting space constraints and ensuring that the chosen variables are relevant to the predictive analysis of poverty.

The analysis shows that 20.02% of the observations in the training set correspond to individuals living in poverty, while 79.98% are not. This indicates that although poverty is a relevant factor, the majority of households in the sample are above the poverty threshold, which is an important point to consider when understanding the distribution and variability of the data. As mentioned earlier, *Figure 1* visually shows the distribution of the population classified as poor and not poor. In this figure, the poverty line is used as the criterion for determining poverty, based on the total income of the observations. The poverty line is set at $271,522 Colombian pesos, which is used as the threshold to define whether a household is considered poor or not. The key variable used for this figure is total income, obtained from the database provided by DANE.

The number of unemployed individuals also plays a fundamental role in determining poverty. The analysis shows that, on average, there are 1.46 unemployed individuals per household in the sample, which is significant since, as the number of unemployed people in a household increases, the likelihood of that household falling below the poverty line also increases. When analyzing the proportion of unemployed individuals in households, it is observed that 25% of households have less than 33% of their members unemployed, while 75% of households have less than 75% of their members unemployed. These figures reflect the unequal distribution of unemployment among households, a factor that directly contributes to differences in poverty levels.

Education, on the other hand, plays a crucial role in determining household income. In our sample, the most common education level is university: at least one person in 80,385 households reached this education level. In contrast, 2,446 households reported no level of education, 25 households reported preschool as the highest level, 16,492 households reported primary school, and 18,111 households reported high school as the highest level of education achieved. These figures highlight the importance of education as a key factor in economic mobility and the likelihood of overcoming poverty.

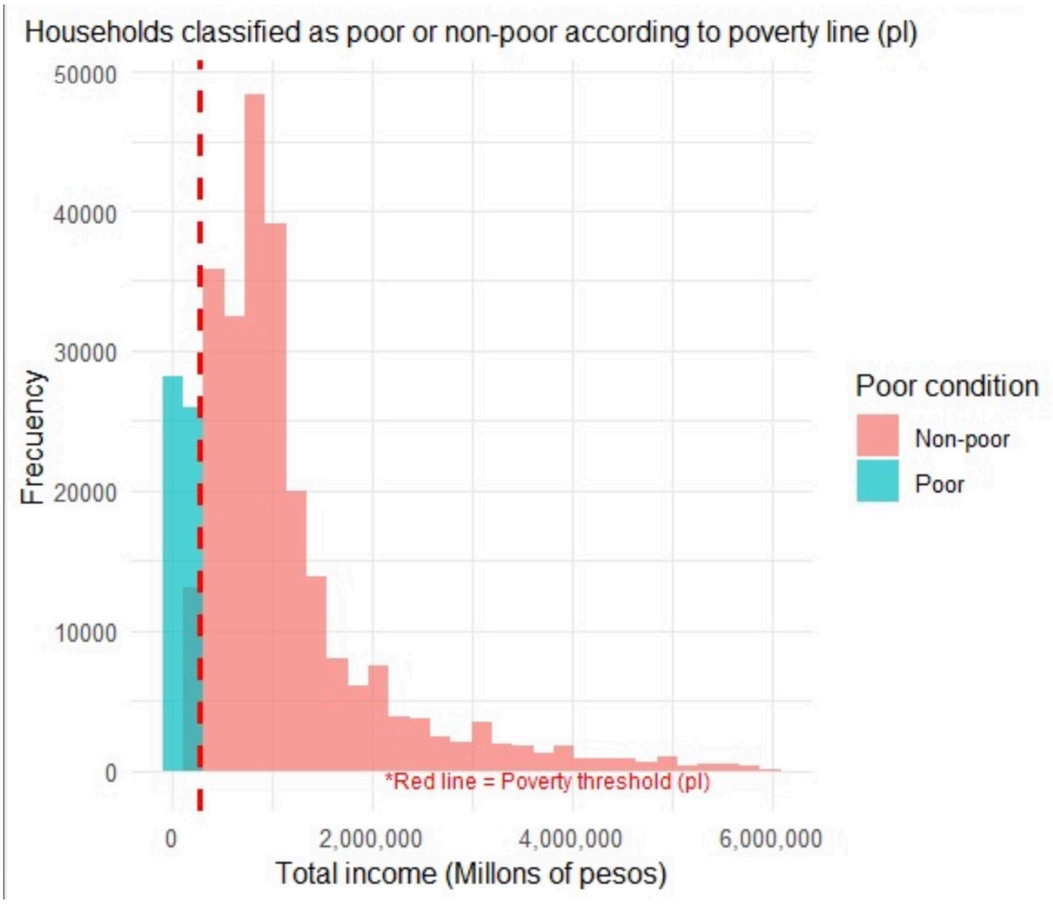

Figure 1. Distribution of Total Income Based on the Poverty Threshold (pl)

3 Training Models

3.1 Predictive Strategy

In this problem set, the goal is to predict whether a household is poor (1) or not poor (0), based on the available socio-economic and demographic variables. The classification of poverty is determined using the poverty line, which varies across households depending on their composition and location, as previously explained. All models use the full set of 34 variables that were derived from both the household and individual datasets, ensuring that a comprehensive set of household characteristics is considered in the predictions. These variables include demographic indicators such as household size, the number of children, elderly, and working members, as well as housing conditions like overcrowding and rental status. Additionally, educational attainment, access to social security, and employment status of household members are critical variables that reflect the economic stability of the household. By incorporating these variables into the models, we aim to capture the multifaceted nature of poverty and improve the accuracy of the predictions.

For this problem set, we employed a range of models to predict household poverty, each chosen for its ability to handle different aspects of the data and provide diverse perspectives on the classification task. We began with the Logit model, a widely used method for binary classification that is well-suited for the task of predicting poverty status (1 = poor, 0 = not poor). The simplicity and interpretability of Logit make it an excellent baseline model, allowing us to understand how the predictors directly influence the probability of a household being classified as poor. In addition, we implemented Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA), both of which are useful for classification when the predictors follow a normal distribution. LDA assumes a common covariance structure across classes, making it efficient for problems where this assumption holds, while QDA relaxes this assumption, allowing for more flexibility in cases where the covariance differs between poor and non-poor households.

To introduce regularization and handle multicollinearity in the dataset, we used Elastic Net, which combines the penalties of both Lasso and Ridge regression. This model is particularly helpful when dealing with a large number of predictors, as it performs variable selection while also controlling for overfitting. CART (Classification and Regression Trees) was also applied to model non-linear relationships between the variables, offering a decision-tree structure that is intuitive and can capture interactions between different predictors. Moving beyond simple trees, we included Random Forest, an ensemble method that builds multiple decision trees and averages their predictions. Random Forest is robust against overfitting and performs well with complex, high-dimensional datasets like ours. Finally, we employed XGBoosting, a boosting method that builds trees sequentially to correct the errors of previous trees. XGBoost is known for its efficiency and accuracy in classification tasks, particularly when dealing with imbalanced datasets and complex interactions. While other boosting methods could have been considered, XGBoost was chosen for its performance and flexibility in this type of prediction problem.

For the methods that require hyperparameter tuning, we tested several combinations based on commonly recommended values for each model. These combinations were selected to optimize performance while avoiding overfitting. Additionally, given the class imbalance in the data, where the number of non-poor households significantly outweighs the number of poor households, we applied SMOTE (Synthetic Minority Over-sampling Technique) to balance the training data. Using the R implementation of SMOTE, the minority class (poor households) was oversampled to achieve a 50-50 balance, ensuring that the models were trained on a more representative dataset and improving their ability to accurately predict poverty status. In all cases, we implemented 5-fold cross-validation (cv = 5) to further enhance the robustness of the models and prevent overfitting. This method divides the data into five subsets, using four for training and one for validation in each fold, providing a reliable estimate of model performance across different data splits. The performance of each model is evaluated using the F1-score, as proposed in the problem set. The F1-score is the harmonic mean of precision and recall, which is particularly valuable when dealing with imbalanced datasets. Maximizing the F1-score helps to minimize false negatives (i.e., misclassifying poor households as non-poor) while maintaining high precision, which is critical in ensuring accurate poverty prediction.

3.2 Models Performance

Model	**F1**	**F1 SMOTE**	**Hyperparameters**
Logistic Regression (Logit)	0.60	0.63	-
Linear Discriminant Analysis (LDA)	0.56	0.60	-
Quadratic Discriminant Analysis (QDA)	0.54	0.54	-
Elastic Net (EN)	0.55	0.58	$\alpha = 0.1, \lambda = 0.001$
Classification Tree (CART)	0.58	0.59	cp = 0.001
Random Forest (RF)	0.60	0.62	$n = 200, \text{mtry} = 5, \text{min\_node\_size} = 20$
XGBoost	0.64	0.64	$N = 200, \eta = 0.2, \gamma = 1, \text{max\_depth} = 4$

Table 2. Performance metrics with and without SMOTE and corresponding hyperparameters

In this analysis, we evaluated seven models in total, each tested in two versions: one trained on the original dataset and another trained with the dataset balanced using SMOTE, resulting in 14 different model setups. To assess model performance, the training dataset was split into sub-training and sub-testing sets, where models were trained on the sub-training set and evaluated on the sub-testing set using the F1-score. The results for all models are summarized in *Table 2*. Beginning with the models that do not require hyperparameter tuning, the Logit model performed reasonably well, achieving an F1-score of 0.60 without SMOTE and improving to 0.63 with SMOTE. This improvement can be attributed to the fact that Logit models often struggle with imbalanced datasets, as they can disproportionately favor the majority class. SMOTE helped balance the dataset, allowing the Logit model to better identify poor households without overly skewing towards the non-poor class. The LDA model followed a similar pattern, with a score of 0.56 without SMOTE and 0.60 with SMOTE. This is expected, as LDA assumes equal covariance matrices across classes, which can make it sensitive to class imbalances. By balancing the classes with SMOTE, LDA was able to make more accurate distinctions between poor and non-poor households. On the other hand, QDA, which relaxes the assumption of equal covariance, did not benefit from SMOTE, maintaining an F1-score of 0.54 in both cases. QDA's sensitivity to small sample sizes within the minority class likely limited its performance, and even with SMOTE, it could not effectively improve predictions for poor households.

For the CART model, the primary hyperparameter tested was the complexity parameter (cp), which controls the size of the tree and helps to avoid overfitting. We tested values of cp ranging from 0.001 to 0.1 in increments of 0.01, as this range is commonly used to fine-tune the model's balance between complexity and generalization. The results show that CART achieved an F1-score of 0.58 without SMOTE and 0.59 with SMOTE. The modest improvement with SMOTE can be attributed to CART's inherent sensitivity to class imbalances, as it tends to split based on majority class dominance. By oversampling the minority class (poor households), SMOTE allowed the tree to grow in a way that better represented the minority class, albeit with a small gain in performance. The slight improvement reflects CART's limitation in handling more complex interactions within the data compared to ensemble methods. Moving to Random Forest, we explored multiple hyperparameter configurations to optimize the model's performance. Specifically, we tested values for the number of variables randomly selected at each split, trying 5, 6, and 8 variables, as well as the minimum node size, which was tested with values of 10 and 20 to control the depth of trees. The split rule used was the default "gini" index, which is standard for classification tasks. Random Forest performed well, with an F1-score of 0.60 without SMOTE, improving to 0.62 with SMOTE. The improvement from SMOTE is more pronounced here, as Random Forest benefits significantly from balanced datasets. The oversampling of the minority class allowed the ensemble of trees to correct for the class imbalance, resulting in more accurate predictions of poor households. The robustness of Random Forest, along with its ability to handle high-dimensional data and complex interactions, contributed to its strong performance.

For XGBoost, a powerful boosting algorithm known for its performance in classification tasks, we tested several key hyperparameters to optimize the model. The number of boosting iterations was set to 200 to ensure the model had enough rounds to converge. We experimented with the maximum tree depth, testing values of 4 and 6, to control the complexity of the model and prevent overfitting. The learning rate, which determines the size of the step taken at each boosting iteration, was varied between 0.05 and 0.2 to balance between fast convergence and model stability. Additionally, the regularization parameter (gamma), which dictates the minimum loss reduction needed for further splitting, was tested with values of 0 and 1 to enhance the model's generalization capabilities. Finally, the subsample ratio was set to 0.8 to reduce overfitting by ensuring that each tree was trained on a random subset of the data. XGBoost delivered the highest performance among all models, achieving an F1-score of 0.64 both with and without SMOTE. This result is expected, as XGBoost is known for its ability to handle imbalanced datasets effectively, even without oversampling. The boosting mechanism of XGBoost, which builds trees sequentially to correct for the errors of previous trees, inherently helps mitigate class imbalance. Therefore, while SMOTE did not further improve its performance, the model still outperformed others due to its ability to capture complex interactions and its robustness in handling high-dimensional data. These results highlight XGBoost's flexibility and strength in classification tasks, particularly in complex and imbalanced datasets like this one.

4 Final Models

4.1 Models Performance

Model	**Optimal Threshold**	**F1 (Out-of-sample)**	**F1 (Kaggle)**
Logistic Regression (Logit)	0.62	0.64	0.637
Random Forest (RF)	0.39	0.65	0.659
XGBoost	0.34	0.67	0.675

Table 3. Optimal threshold and F1 scores for out-of-sample test and Kaggle

For the final models submitted to Kaggle, we selected three models: Logit, Random Forest, and XGBoost, all trained with the dataset balanced using SMOTE, as explained in the last section. These models were chosen because they achieved the highest F1-scores during the initial evaluation, as detailed in the previous section. The hyperparameters for each model were the ones that had previously maximized the F1-score during cross-validation. The primary innovation at this stage was the use of an optimal threshold. Instead of relying on the default 0.5 threshold for classification, we determined the threshold that maximized the F1-score on the sub-testing set for each model. This threshold was then applied to the predictions on the full test set, which were subsequently submitted to Kaggle for evaluation.

The results from Kaggle, summarized in *Table 3*, indicate that XGBoost with SMOTE was the top-performing model, achieving an F1-score of 0.675. This model also recorded the highest F1-score in the sub-testing set, with 0.67. The primary strength of XGBoost lies in its boosting mechanism, which allows it to sequentially correct errors made by previous trees. This capability, combined with its ability to model complex, non-linear relationships between the variables, makes XGBoost particularly effective in this dataset, which contains both household-level and individual-level characteristics. The optimal threshold for XGBoost was set at 0.34, significantly lower than the thresholds of the other models. This lower threshold greatly contributed to its superior performance by improving recall, allowing the model to capture a larger proportion of true positives-households classified as poor. In a task like poverty prediction, recall is especially important, as missing true cases of poverty could lead to ineffective policy interventions. The ability of XGBoost to balance recall without overly sacrificing precision (thus achieving a high F1-score) highlights its robustness and adaptability, especially when working with imbalanced data.

Random Forest with SMOTE, while not as performant as XGBoost, also demonstrated strong results, with an F1-score of 0.659 on Kaggle and 0.65 in the sub-testing set. Random Forest is known for its resilience against overfitting and its ability to handle high-dimensional datasets through its ensemble approach. The fact that Random Forest performed well across both out-of-sample data and the Kaggle test indicates that it was effective at generalizing across different samples. However, its performance fell slightly behind XGBoost due to the differences in their methods: while Random Forest relies on independent trees voting, XGBoost builds each tree sequentially, which allows it to iteratively refine and correct mistakes, leading to slightly better performance. Random Forest's optimal threshold of 0.39 was also lower than the Logit model's threshold, indicating that it too favored recall, though not to the same extent as XGBoost. The overall balance between precision and recall in Random Forest was good, but its inability to capture more complex patterns in the data may have limited its predictive capacity compared to XGBoost.

Finally, the Logit model with SMOTE achieved an F1-score of 0.637 on Kaggle and 0.64 on the sub-testing set. While this result is respectable, it falls behind the two ensemble methods. Logit models tend to work well in simpler classification problems but struggle when the relationships between variables are more complex, as was the case with this dataset. The optimal threshold for Logit was set at 0.62, much higher than those of Random Forest and XGBoost, which indicates that the Logit model prioritized precision over recall. This higher threshold means that the model was more conservative in classifying households as poor, leading to fewer false positives but also a lower recall-resulting in more missed cases of poverty. This trade-off between precision and recall explains why Logit did not perform as well as the ensemble methods in terms of overall F1-score. While precision is important, in this context, missing too many poor households can be a significant limitation, as the goal is to correctly identify as many households in need as possible.

In comparing the models, the most notable observation is how the optimal threshold selection played a critical role in improving model performance. XGBoost benefited the most from a lower threshold, which allowed it to enhance recall without drastically reducing precision, contributing to its superior F1-score. Random Forest also saw gains from a relatively lower threshold, though not as pronounced as in XGBoost. On the other hand, Logit's higher threshold emphasized precision, but this came at the expense of recall, explaining why it lagged behind the other models. Overall, while all three models benefited from the use of SMOTE to handle class imbalance, the ensemble methods, particularly XGBoost, proved better equipped to handle the complexities of the dataset and maximize the F1-score through fine-tuned threshold optimization.

4.2 Variable Importance

From the variable importance plots for Random Forest and XGBoost (*Figure 2*), we can identify several common patterns across both models. In the XGBoost variable importance plot, the most critical variables are the number of unemployed individuals and the proportion of unemployed individuals in the household, followed by the maximum education level of university or higher attained by household members. Unemployment is a strong predictor of poverty, as households with unemployed members are more likely to face financial instability due to the lack of consistent income. The inclusion of education, particularly university-level education, reflects the well-established link between higher education and economic well-being. Households with members who have higher levels of education are generally more resilient to poverty due to better employment prospects and higher earning potential. Additionally, the rent variable is important, indicating that housing costs significantly affect poverty status. High rental expenses can place considerable strain on household finances, particularly in lower-income households. Other important variables in XGBoost include the number of minors, the number of subsidies received, and whether the head of the household is female. These variables are crucial as they capture demographic and gender dynamics within the household, offering insights into vulnerabilities related to family structure and dependency levels.

In the Random Forest model, similar variables are ranked highly. The proportion of unemployed individuals, the number of unemployed, and rent remain top predictors, reinforcing the importance of employment status and housing costs in poverty prediction. Additionally, variables like the number of minors and maximum education level (university) are again crucial, suggesting strong consistency across both models. One key difference is that the proportion of individuals with social security appears as a more important variable in Random Forest compared to XGBoost. This reflects the model's sensitivity to access to formal employment and social safety nets, which can be critical in reducing poverty vulnerability. Interestingly, the number of subsidies and the proportion of individuals who receive subsidies are important in both models, suggesting that government support programs are a key factor in determining household resilience to poverty. Other variables, such as number of inactive individuals, number of females, and household size (number of people per room), are significant in both models but vary slightly in rank, with Random Forest placing more weight on social and demographic proportions.

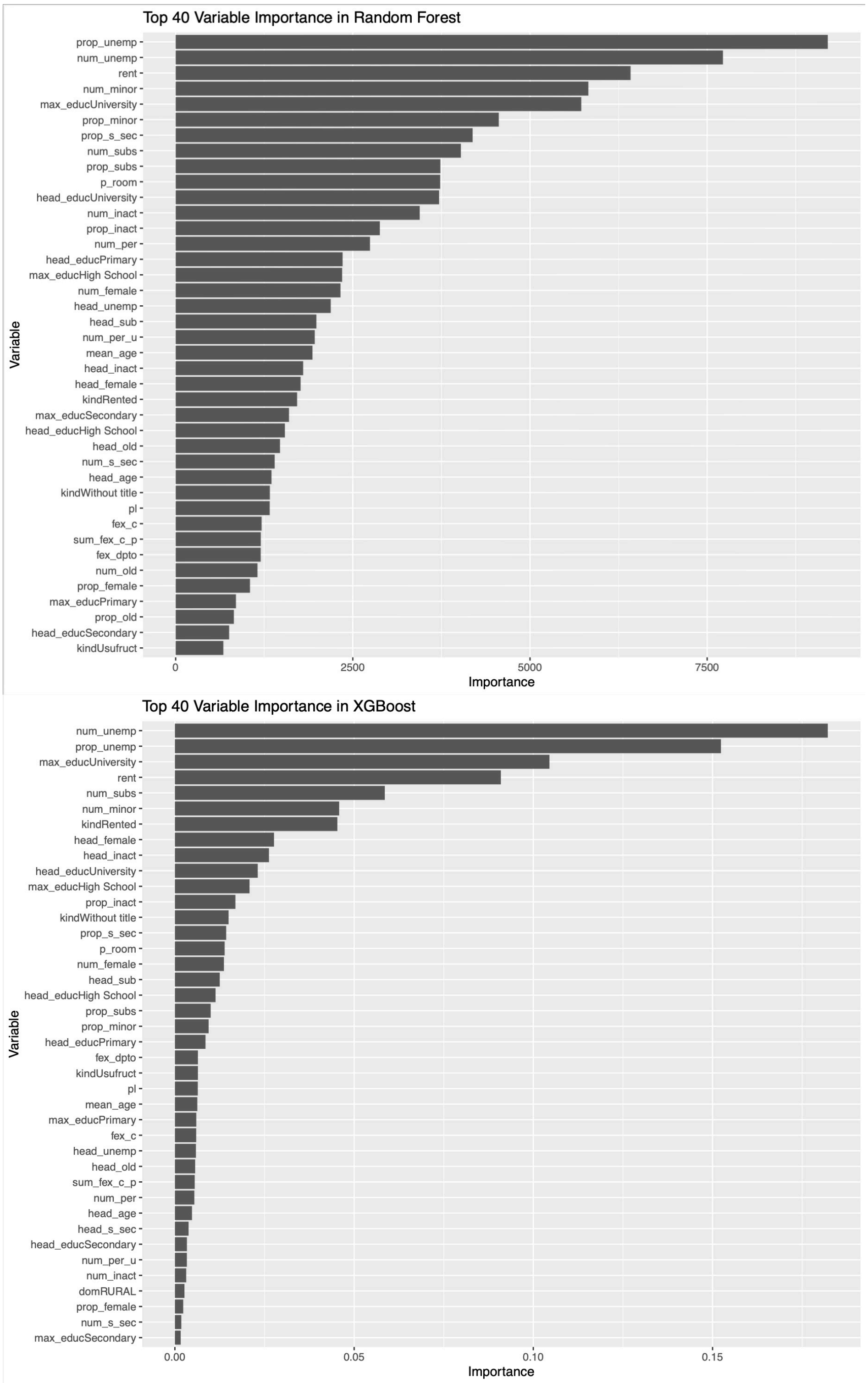

Figure 2. Variable Importance for Random Forest and XGBoost

Additionally, in XGBoost, variables like the education level of the household head and whether the head of household is inactive or unemployed also play a substantial role, reflecting the critical influence of the household head's characteristics on the overall economic well-being of the household. In Random Forest, the expansion factor at the departmental level and the poverty line emerge as important variables. These geographic and policy-related factors indicate that regional disparities and thresholds for poverty classification are key in understanding poverty across different contexts.

Although we do not have a specific variable importance plot for the Logit model, we can infer that similar variables, such as number of unemployed, proportion of unemployed, maximum education level, and rent, would be crucial due to their direct, interpretable relationships with poverty.

Variables related to social security, subsidies, and household size would likely also play important roles, though the linear nature of Logit may limit its ability to capture the complex interactions between these variables that are well handled by Random Forest and XGBoost.

In summary, while both XGBoost and Random Forest highlight the importance of unemployment, education, rent, and household composition, Random Forest places slightly more emphasis on social security access and expansion factors, whereas XGBoost tends to prioritize demographic proportions and household head characteristics. These differences reflect the strengths of each model in capturing different dimensions of household vulnerability to poverty. The Logit model would likely rely on the same key variables but may struggle to capture the more nuanced interactions that the ensemble methods handle, leading to its relatively lower performance.

5 Conclution

The primary objective of this analysis was to predict household poverty status based on a rich set of socio-economic and demographic variables from the MESE dataset. The models aimed to classify households as poor or non-poor, with an emphasis on optimizing predictive performance through F1-scores. After evaluating various models, three were selected for final submission to Kaggle: Logit, Random Forest, and XGBoost, each using the optimal threshold and SMOTE for handling class imbalance.

Among these, XGBoost proved to be the best-performing model, achieving the highest F1-score of 0.675 on Kaggle, as well as the highest score in the sub-test set. Its superior performance can be attributed to its boosting mechanism, which allows it to iteratively correct errors in previous trees and capture complex non-linear relationships within the data. Additionally, the use of a lower optimal threshold (0.34) enabled XGBoost to prioritize recall, which is critical in poverty prediction tasks where identifying as many poor households as possible is crucial for effective intervention. Random Forest also performed well, achieving an F1-score of 0.659, but its ensemble method of independent trees was less effective in correcting classification errors compared to XGBoost's sequential tree-building process. Lastly, the Logit model, while respectable with an F1-score of 0.637, struggled to match the complexity captured by the ensemble methods due to its simpler linear nature.

When examining the variables driving performance in these models, both Random Forest and XGBoost highlighted unemployment (number and proportion of unemployed individuals) and education levels (university degree or higher) as critical predictors of poverty. These variables underscore the direct relationship between employment status, educational attainment, and economic well-being, reflecting broader societal trends where higher education leads to better employment opportunities and a lower likelihood of poverty. Rent was also a key factor, as high housing costs disproportionately affect poorer households, limiting their disposable income and contributing to financial strain. Additionally, demographic variables like the number of minors and gender of the household head captured household composition dynamics, further illustrating the model's ability to identify vulnerable households.

Despite the strengths of these models, there are several potential paths for improving the results. Firstly, hyperparameter tuning could be expanded further, exploring a wider range of values for models like XGBoost and Random Forest to optimize their performance. Secondly, feature engineering could be enhanced by creating more interaction terms or exploring non-linear transformations of key variables, especially in models like Logit where these relationships are not automatically captured. Lastly, employing ensemble techniques that combine the strengths of different models could potentially yield better results by leveraging the unique advantages of each approach.

In conclusion, XGBoost emerged as the most effective model for predicting poverty in this dataset, driven by its ability to handle complex interactions and a lower classification threshold that maximized recall. Future improvements could focus on refining hyperparameters, exploring richer feature transformations, and integrating ensemble techniques to further enhance predictive accuracy and robustness.