Predictive Analysis for Identifying Poverty Risk Factors

Problem Formulation

In our project, we aim to develop a predictive model that identifies households at risk of falling into poverty, specifically, utilizing demographic characteristics, educational attainment, housing material, and infrastructure as predictors. By analyzing a comprehensive dataset, we seek to answer the following questions:

1. Can demographic characteristics such as age, gender, marital status, and relationship to household head serve as reliable indicators to identify households at risk of poverty?
2. To what extent does educational attainment, measured by years of schooling, years behind in school, and level of education contribute to predicting the likelihood of a household falling into poverty?
3. Does housing ownership, including availability of basic amenities such as toilets, bathroom, electronics, and number of rooms influence economic vulnerability of households?
4. How do housing material and infrastructure variables, such as wall, floor, and roof materials, water provision, electricity source, toilet facilities, and rubbish disposal method, contribute to the economic vulnerability of households?

We find this problem to be non-trivial. As even in today's age, poverty continues to be a problem. As “even though the country has enjoyed a healthy growth rate for over 25 years, the proportion of Costa Ricans living below the poverty line remains pretty much the same as it did in 1994 at around 20 percent, while income inequality is on the rise" (Hidalgo, 5)

Data Acquisition

For the dataset, we are using Kaggle. The dataset consists of Costa Rican households and the level of poverty based on over 140 data fields as the columns. The dataset has an overwhelming amount of data, such as demographics, employment, education history, and area of residence. The data was deemed relevant to our group, since we could generalize some columns and attempt to predict the causes of poverty.

Data Preprocessing

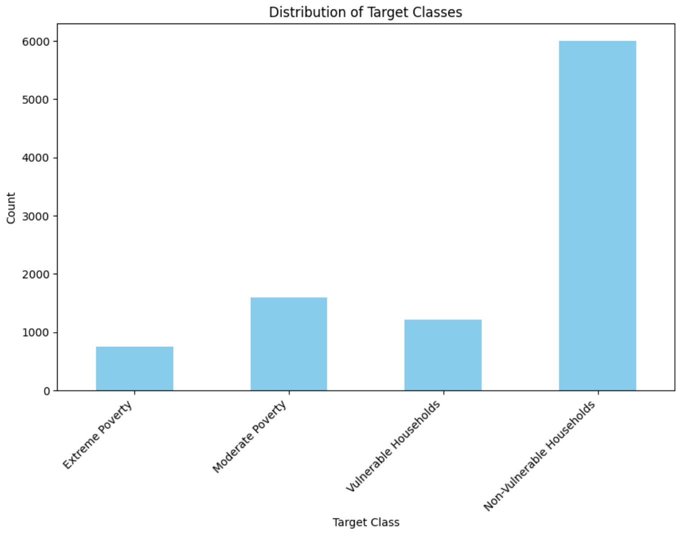
During our data preprocessing and analysis, we found that our training and test dataset contained some missing values around 3.5%. NaN values appeared in the columns like monthly payment, number of tablets owned and education levels of households. For monthly payment, we decided to remove these rows since these NaN values were detrimental to our dataset. In our dataset, there was an indicator column that identifies whether the household owned tablets or not. It seemed that this next column of number of tablets were NaN values when the indicator column was a 0. Therefore, we changed each NaN value of this column to 0. Next, the education levels were given NaN values as a replacement to unknown. After trying a removal, since there was a significant amount of these variables that also removed relevant data rows, we decided that it was best to make the NaN values to the average education value of the columns to keep the rows for relevant data. Moving on to duplicate values, surprisingly we found that there were no duplicates. Since our data contains a large amount of variables, which totalled around 140. We found that many variables can be structured into a single category. We remapped the columns, to reduce the amount of data. For example, the categorical columns of education had 9 different columns to represent 9 different education levels. Instead, we rearranged them into one column called Education, numbered them from 0 to 8 which represents no level to postgraduate.

After remapping multiple columns into one respective column, we had these new fields added to the dataset: Marital Status, Family Role, House Material, Utilities, Child/elder ratio, House Payment Method, Region, Electronics.

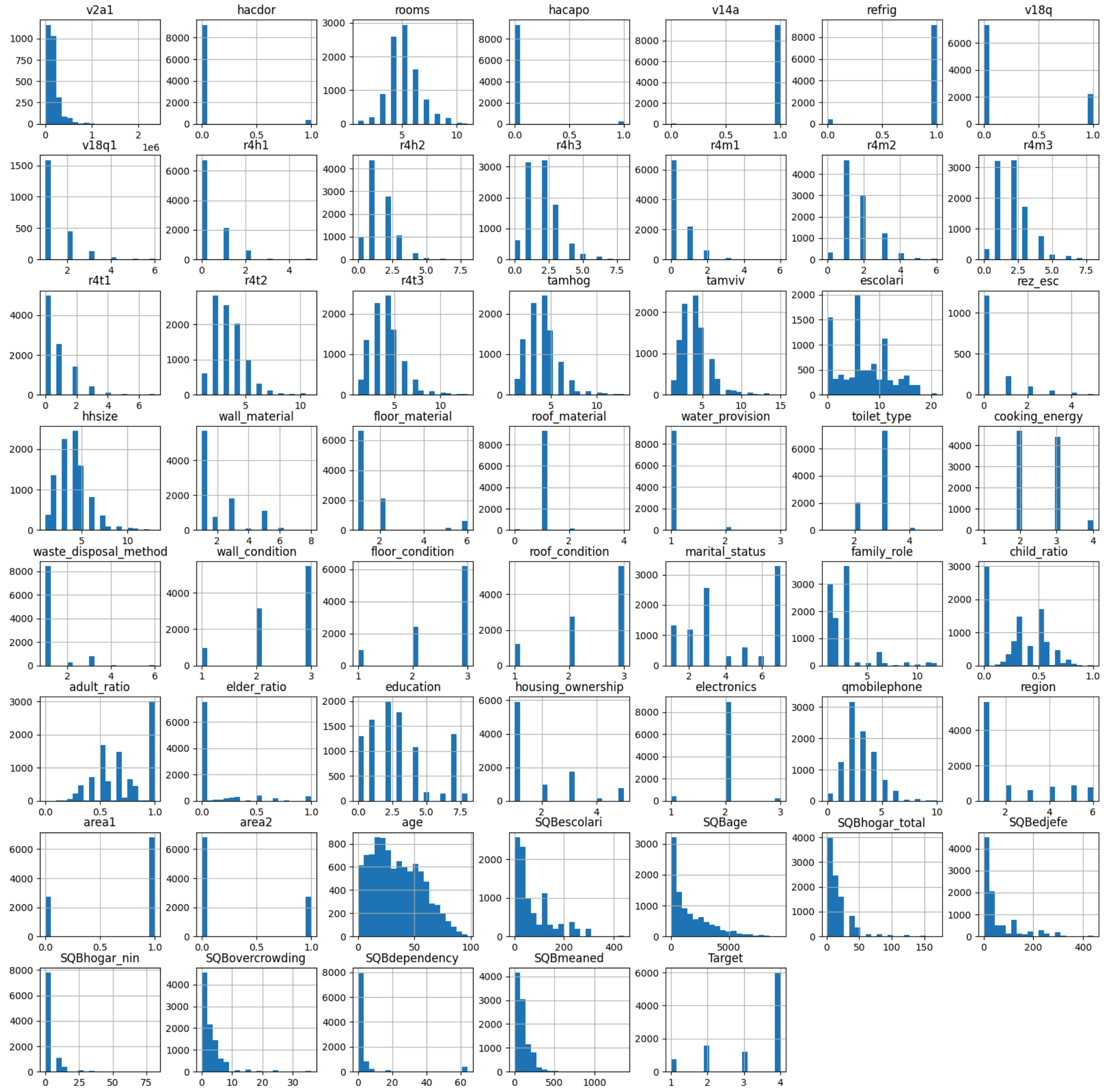
We saved the newly cleaned data into a new csv file, then started our EDA portion.

Exploratory Data Analysis

First let's look at our overall target distribution.

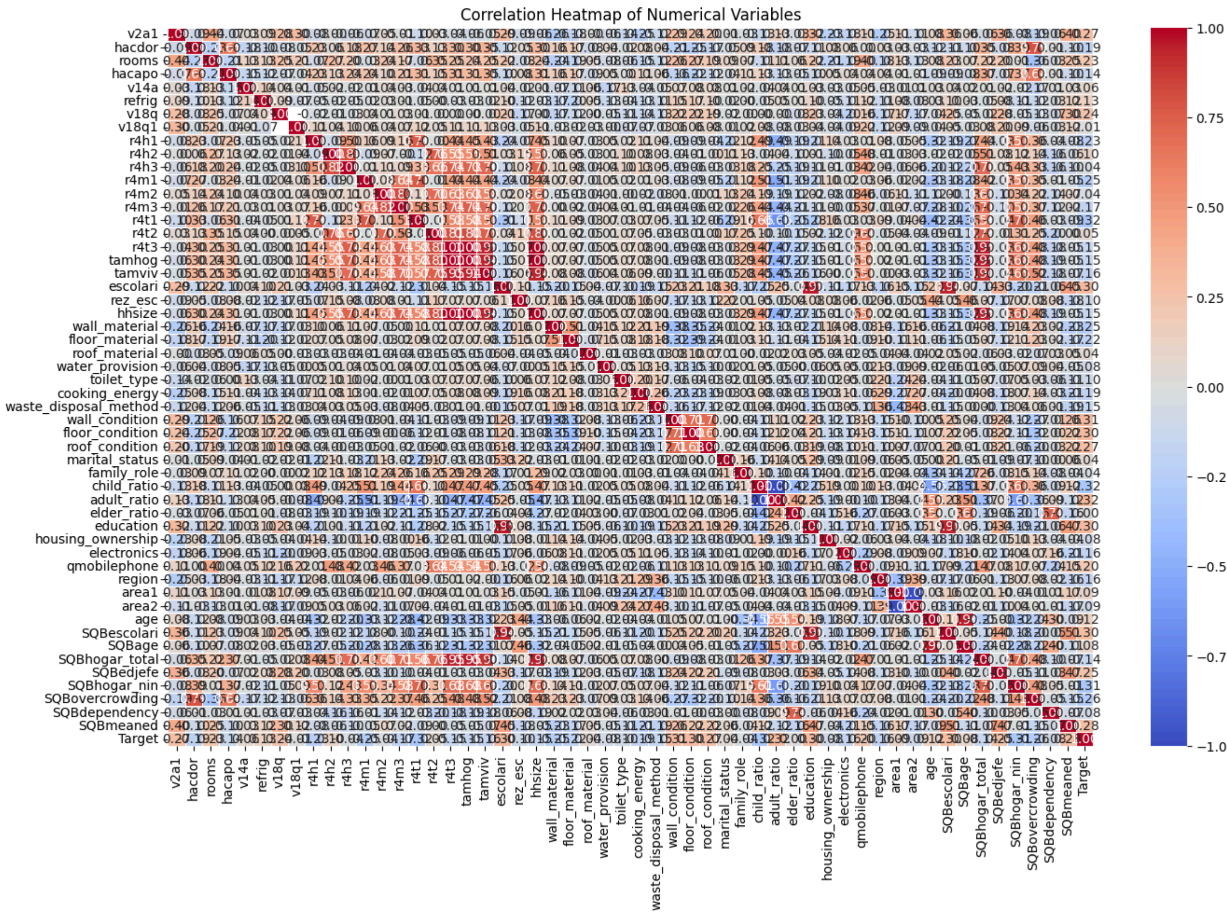


Most of our households in our dataset are non-vulnerable to poverty. However, we can see after cleaning, there are more than 3000 households that may be experiencing financial struggles. Next, we looked at the distributions of all relevant columns.

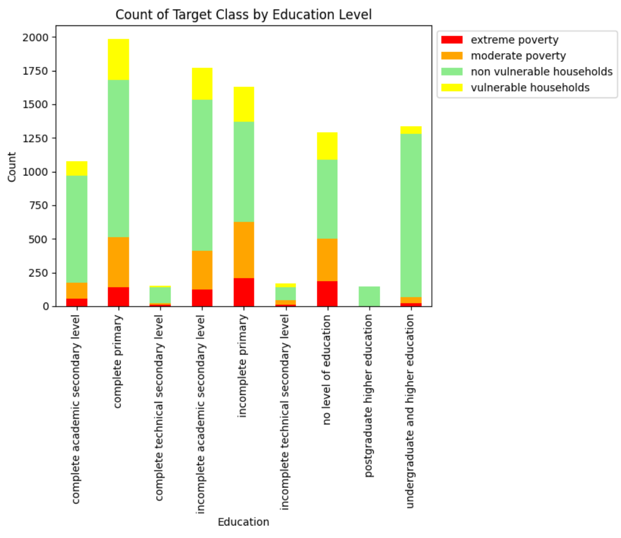


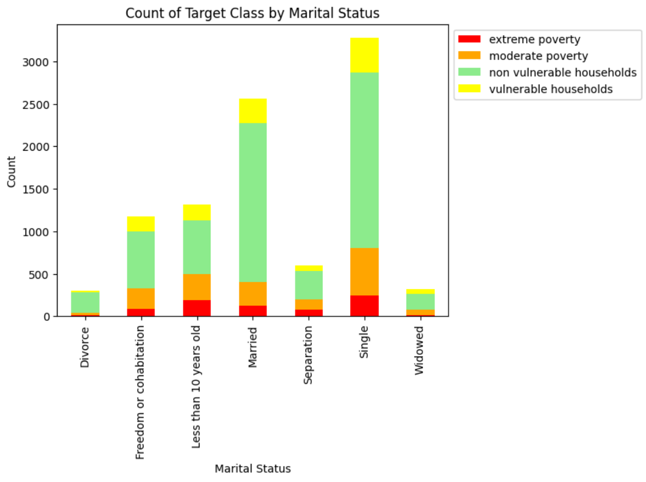
From these columns, we can see that there are many different types of data, including some indicator variables such as hadcor, hacapo, v14a, refrig, etc.

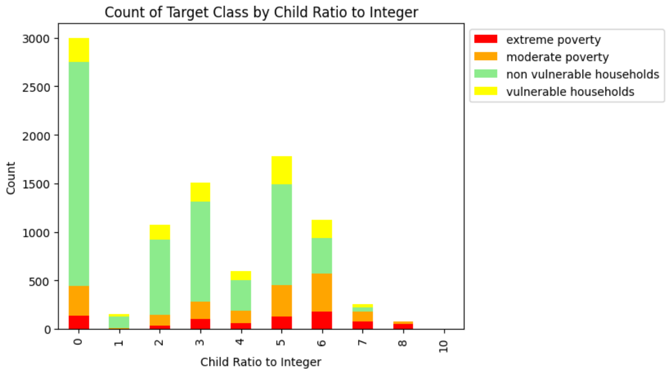
There isn't a concrete pattern to any one of these columns, and we decided to go in depth to their correlations for our non-visual insight.

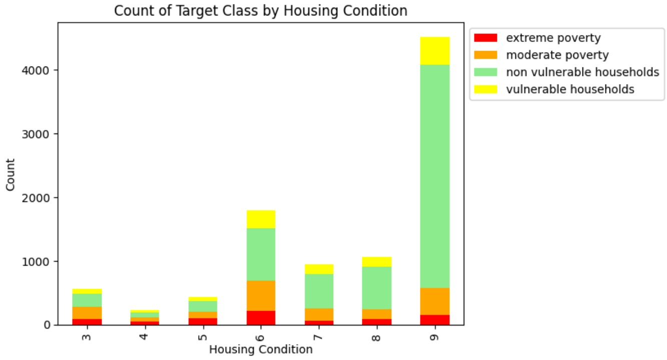


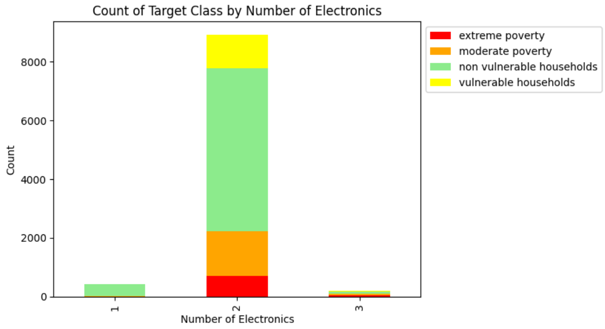
This correlation heatmap tells us the relationships of all of our relevant variables to our target. Unfortunately, the heatmap does not tell us much, as the biggest correlation values within our variables are 0.32, and -0.32. We will now explore further into specific independent variables, specifically: Marital Status, Education, Child Ratio to integer, Housing Condition, Electronics, Region. These were our findings:

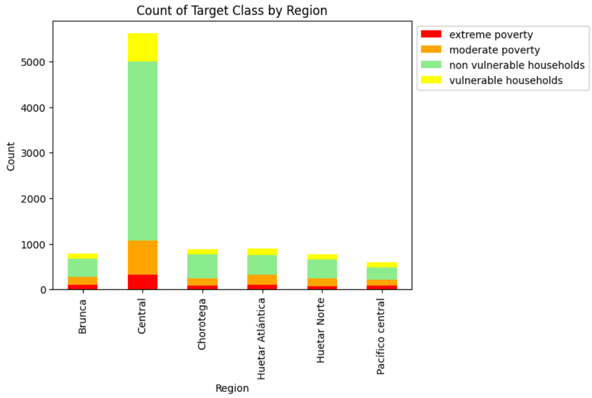












Examining various variables, it becomes evident that several are inconsequential. For instance, households exhibit a significant clustering around owning two electronics, making it challenging to pick out any contribution to increase of poverty. These graphical representations offer some insights to our original questions in the problem formation. Revisiting question 3, it is shown that the presence or absence of electronics does not augment the economic susceptibility of households.

Modeling

Our problem is a classification problem, so we decided to use both sklearn’s neural network in “MLPClassifier”, and “GradientBoostingClassifer”. Both MLPClassifier and GradientBoostingClassifer need a large dataset, but we find that GradientBoostingClassifer gave an overall better representation to our accuracy.

MLPClassifiers can capture complex nonlinear relationships between features and the target variable, making them suitable for tasks where simpler models may not suffice. Since MLPs have many hyperparameters to tune, including the number of layers, it is not only computationally expensive, but it is also prone to overfitting. GradientBoosting builds an ensemble to weak learners sequentially, improving upon the eros of previous models, and this often leads to high accuracy. We find that using MLPClassifier gave a slightly less accurate result as we were not able to test the best hyperparameters. Since GradientBoosting gives a robustness to overfitting, it seemed to fit our mode of computation and dataset best. In order to have the best understanding of the columns in our dataset, we gathered together the variables which were most highly correlated in the heatmap from our EDA, and used those metrics in our new test data. Our dependent variable/target was the poverty level specified in order of “1: 'Extreme Poverty', 2: 'Moderate Poverty', 3: 'Vulnerable Households, and 4: 'Non-Vulnerable Households”. After carefully choosing GradientBoosting, we used a cross-validation strategy for testing. The cross-validation method was used to collect 5 different folds then tested on each. Then, we took the 5 models, and used the one with the best accuracy/least error, and we got approximately 87.9% accuracy, which is a very good model.

Model Summary:

$F(x) = \sum^{M}_{m=1} \gamma_m h_m(x)$

Where:

$F(x)$→ Final prediction function

M → Number of weak learners (trees) in the ensemble

γ_m → Learning rates associated with each weak learner

h_m→ Individual weak learners

Estimations for these parameters, is M = 100, y_m = 0.1, h_m = 100

So the equation is $\sum^{100}_{m=1} 0.1 h_m(x)$

To fit the model, we used an appropriate size of 80/20 split to test the data, which was a ratio of 1725:424 in training and testing.

Model Evaluation

After testing the model on separate test data, we see that the test data does not perform as well as the training data. Although it is decently accurate, it falls down to 75.93%, which most likely means that my model is most likely overfitting. Two distinct metrics were chosen for validation purposes. Accuracy measures the proportion of correctly predicted instances out of the total instances in the test set. It is a common metric used for classification tasks and provides an overall indication of model performance. The F1 score is the harmonic mean of precision and recall, providing a balance between the two metrics. It is particularly useful when dealing with imbalanced datasets, as it considers both false positives and false negatives. The model's performance on the training set was not directly evaluated in the provided code snippet. However, typically, a comparison between training and testing performance can reveal insights into the model's ability to generalize to unseen data. If the model performs significantly better on the training set compared to the test set, it may indicate overfitting, where the model has learned to memorize the training data rather than generalize to new data. Conversely, if the model performs similarly on both sets, it suggests good generalization ability.

Conclusion

In modeling the target variable of identifying households at risk of poverty, our approach yielded promising outcomes. Leveraging a dataset encompassing various demographic, educational, and housing characteristics, we employed GradientBoostingClassifier for its robustness in handling complex relationships and achieving high accuracy. Through extensive preprocessing, including handling missing values and feature engineering, we derived meaningful insights. Notably, variables like marital status, education, and housing conditions exhibited notable correlations with poverty risk, providing valuable indicators for identifying vulnerable households. However, a notable limitation surfaced during model evaluation, wherein the test data performance lagged behind the training data, indicating potential overfitting and suggesting avenues for further refinement. Despite this drawback, the model demonstrated a commendable strength in its predictive capabilities, with an accuracy of approximately 87.9% on the training set. Moving forward, future researchers embarking on similar endeavors should prioritize robust validation techniques to mitigate overfitting and consider incorporating additional data sources or refining feature selection strategies to enhance model generalization. Additionally, fostering collaboration and knowledge exchange within the research community could facilitate the adoption of best practices and foster innovation in poverty prediction methodologies.

Bibliography

“Costa Rican Household Poverty Prediction." *Kaggle*, Kaggle Inc., www.kaggle.com/competitions/costa-rican-household-poverty-prediction. Accessed[12 Apr. 2024]

Hidalgo, Juan, Growth Without Poverty Reduction: The Case of Costa Rica (January 23, 2014).

Cato Institute Economic Development Bulletin No. 18, Available at SSRN:

https://ssrn.com/abstract=2499826