This project is aimed to predict the poverty level of Costa Rican households by using the data records of information about households and individuals. We also want to find out the most important factors that influence the judgment of poverty levels. So, the data mining task includes data preprocessing, building different classification models by using SPSS Modeler and Python and model evaluation.

Here is the part of SPSS. We start from data preprocessing. The first thing we did is drop some data. We have a feature named ‘parentesco1’, which means head of this household. The poverty level of the whole household is decided by the poverty level of this header. So we can ignore family members who are not headers and only deal with data who are the head of the household. Next is missing value and mix type. We have two features with missing value, monthly rent payment and number of tablets household owns, we fill these two features with number 0. With the mix type, we fill ‘no’ with number 0, and ‘yes’ with number 1.

Then comes feature selection. We let SPSS do the selection first, we drop features with importance lower than 0.95. also, we selected the feature by hand, to make the features more logical.

A screenshot of a cell phone

Description automatically generatedThe last thing of data preprocessing is about imbalanced data. From the table above, we can see the original data is very imbalanced, so we tried two methods, SMOTE and oversample. We can see SMOTE did not performed well, so we decided to build the model with original data and oversampled data.

The second part is model building. We build four models, named ‘Decision Tree’, ‘Random Forest’, ‘XGBoost Tree’ and ‘ANN’. We apply original data and oversampled data to the models separately and do the comparison of these two datasets.

From the models, we can see what the important features are. No matter in models with original data or models with oversampled data, ‘average years of education for adults’, ‘number of mobile phones’, ‘number of persons’ and ‘dependency rate’ turn to be the most important ones. Based on this, if some charity organizations need to do the poverty research, they can focus on features like these and ignore features like where the electricity come from or how the household deal with rubbish.

Then we do the comparison. We introduced three measurements to evaluate the performance of the models, which are accuracy, weighted-F1 and macro-F1. Marco-F1 is the average F1 of all class, and weighted-F1 is macro-F1 with different weights. F1 measure is a combination of precision and recall, so it’s a more balanced measure. Here is the comparison table.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Models with Oversampled Data | | | Models with Original Data | | |
|  | Accuracy | Weighted-F1 | Macro-F1 | Accuracy | Weighted-F1 | Macro-F1 |
| C5.0 | 52.35% | 53.64% | 33.17% | 65.81% | 61.15% | 39.38% |
| Random Forest | 64.74% | 61.74% | 40.06% | 66.45% | 60.31% | 37.02% |
| XGBoost | 60.26% | 58.31% | 36.57% | 67.31% | 60.93% | 38.90% |
| ANN | 50.43% | 71.02% | 54.50% | 65.60% | 71.02% | 54.5% |

From this table, we can see models with oversampled data didn’t performed better than models with original data. So we decided to check the models with original data. From accuracy, we can see XGBoost performed best, with the accuracy of 67.31%. However, when we turn to weighted-F1 and Macro-F1, we can see ANN performed best. So we decide to choose ANN as our final model.