Problem Set 2: Predicting Poverty
"Wars of nations are fought to change maps. But wars of poverty are fought to map change" M. Ali

Due Date: September 30 at 6pm in Bloque Neón

1 Introduction

This problem set was inspired by a recent competition hosted by the world bank: Pover-T Tests: Predicting Poverty. The idea is to predict poverty in Colombia. As the competition states, *"measuring poverty is hard, time consuming, and expensive. By building better models, we can run surveys with fewer, more targeted questions that rapidly and cheaply measure the effectiveness of new policies and interventions. The more accurate our models, the more accurately we can target interventions and iterate on policies, maximizing the impact and cost-effectiveness of these strategies."*

The objective is to predict poverty at the household level. Data, however, are provided at the household and individual levels. You can use individual-level information to build extra variables to improve your prediction. You can use the variable `id` to merge households with individuals.

The data folder contains four data sets from the GEIH; training and testing data sets at the household and individual levels. You will note that some variables are missing in the testing data sets; this is designed to make things a bit more challenging.

A document describing the variables is available in the data folder. You can also check them on the DANE website. An essential dimension for policymakers is they can *rapidly and cheaply* measure poverty. When building your model, aim for a model that uses the minimum number of variables.

There are two expected outputs:

1. A `.pdf` document.

2. And a `.csv` file with your predictions.

1.1 General Instructions

The main objective is to construct a predictive model of household poverty. Note that a household is classified as

$$Poor = I(Inc < Pl) \tag{1}$$

where I is an indicator function that takes one if the family income is below a certain poverty line.

This suggests two ways to go about predicting poverty. First, approach it as a classification problem: predict zeros (no poor), and ones (poor). Second, as an income prediction problem. With the predicted income, you can use the poverty line and get the classification. You will explore both routes in this problem set.

The document must contain the following sections:

- Introduction. The introduction briefly states the problem and if there are any antecedents. It describes the data and its suitability for the issue at hand. It contains a preview of the results and main takeaways.
- Data[1]. Treat this section as an opportunity to present a compelling narrative to justify or defend your data choices and help the reader understand your data and its variation. Use your professional knowledge to add value to this section. Do not present it as a "dry" list of ingredients.
- Models and Results. This section presents the specifications and models used for the predictive tasks. Since the approach it's two-fold, you must include two subsections:

  1. Classification models. This subsection describes the classification approach, that is, your attempt o directly predict zeros (no poor), and ones (poor)
  2. Income regression models.

  These subsections must include (not necessarily in this order):

  - A detailed explanation of the final chosen model. The explanation must include how the model was trained, hyper-parameters selection, and other relevant information.
  - A comparison to at least 5 other specifications/models for each approach. That is, compare your best classification specification/model to at least 5 (five) others, and the same for your regression specifications. Compare them in terms of predictive power for the task at hand.

[1] This section is located here so the reader can understand your work, but it should probably be the last section you write. Why? Because you are going to make data choices in the estimated models. And all variables included in these models should be described here.

    - A description the variables used in the model and discuss their relative importance in the prediction.
    - A description of any sub-sampling strategy used to address class imbalances.

- Conclusions and recommendations. In this section, you state the main takeaways of your work.

2 Additional Guidelines

I expect the following things from the problem set, omission of any of these guidelines will be penalized.

- Turn a `.pdf` document in Bloque Neón. The document should not be longer than 6 (six) pages and include at most 6 (six) exhibits (tables and/or figures). Bibliography and exhibits don't count towards the page limit. You are welcome to add an appendix, but the main document must be self-contained. Specifically, a reader should be able to follow the analysis in the paper and be convinced it is correct and coherent from the main text alone, without consulting the appendix.

- Turn a `.csv` file in Bloque Neón. An example of how the submission file should look like is in the data folder: `submission_template.csv`. This file includes three columns, one with the variable *id*, one with your prediction using a classification model, and one with a prediction using an income model. **Do not change the name of the columns**.

- I will judge predictions based on false-positive rates, false-negative rates, and the model's sparsity. In the final score, false negatives will have more weight (75%), i.e. poor families classified as non-poor, and the more variables you use, the lower your score.

- Name your `.csv` file with the following convention: the file name must include the name `predictions`, followed by your teammates' last names, and the number of variables you used for each model's prediction, all separated by underscores. For example, `predictions_gomez_matinez_sarmiento_c4_r5.csv`, where `c4` indicates the number of variables in the classification final model, and `r5` indicates the number of variables in the regression final model. Transformations of the same variable do not count as extra variables. For example if you have Age, Age^2, $Age \times Gender$, that only counts as two variables, since you are only using Age and $Gender$.

- I will assign bonus points based on relative rankings.

- Tables, figures, and writing must be as neat as possible. Label all the variables included. If you have something in your figures or tables, I expect they are addressed in the text.

- The document must include a link to your GitHub Repository.
  - The repository must follow the template.
  - The README should help the reader navigate your repository. A good README helps your project stand out from other projects and is the first file a person sees when they come across your repository. Therefore, this file should be detailed enough to focus on your project and how it does it, but not so long that it loses the reader's attention. For example, Project Awesome has a curated list of interesting READMEs.
  - Include brief instructions to fully replicate the work.
  - The main repository branch should show at least five (5) substantial contributions from each team member.
- The code has to be:
  - Fully reproducible.
  - Readable and include comments. In coding, like in writing, a good coding style is critical. I encourage you to follow the tidyverse style guide.