**Introduction**

Many social programs have a hard time making sure the right people are given enough aid. It’s especially tricky when a program focuses on the poorest segment of the population. The worlds poorest typically can’t provide the necessary income and expense records to prove that they qualify.

In Latin America, one popular method uses an algorithm to verify income qualification. It’s called the Proxy Means Test (or PMT). With PMT, agencies use a model that considers a family’s observable household attributes like the material of their walls and ceiling, or the assets found in the home to classify them and predict their level of need.

While this is an improvement, accuracy remains a problem as the region’s population grows and poverty declines.

Beyond Costa Rica, many countries face this same problem of inaccurately assessing social need.

IDB (Inter-American Development Bank) which is the largest source of development financing for Latin America and the Caribbean has taken an initiative to improve this method of classification through the Kaggle community by providing us with the data of Costa Rican households classified in one of four poverty levels i.e. extreme, moderate, vulnerable and non-vulnerable poverty levels.

**Aim**

Our aim in this study/competition is to analyze this dataset, generate insights that would help us in our task of training a machine learning algorithm that would be able to classify each household to the correct poverty level and generalize well on unseen data.

**Background Research**

Background Research or domain knowledge about the problem and the data available to us is crucial for developing a thorough understanding and incorporate important findings or improve upon the work already done or attempted by others working on the same or similar problem.

A fair amount of background research was done on this subject as well which was quite helpful in developing an understanding of the data at hand and incorporating the findings and research done on the economic condition of the people of Costa Rica and defining characteristics among the people from different poverty levels across different regions.

One specific source that proved the most helpful was a book titled **‘Costa Rica’s Development by Ana Maria Oviedo, Susana M. Sanchez, Kathy A. Lindert, and J. Humberto Lopez’** which provided a comprehensive understanding of the causes influencing poverty in the various regions of Costa Rica and different indicators of poverty among the various levels.

The knowledge derived from this research was incorporated into the data in the Feature Engineering and Construction phase of our Data Science pipeline which could help the deployed classification algorithms in discriminating among the various poverty levels in our dataset.

**Data Description**

The complete dataset description is given in the description.txt file which contains all the column names with their descriptions.

**Training** Dataset: **9557** rows **142** columns

**Testing** Dataset: **23856** rows **141** columns

**Number of classes** or poverty levels: **4 (Extreme, Moderate, Vulnerable and Non-Vulnerable)**

**Class Distribution: Non-Vulnerable: 1953, Vulnerable: 355, Moderate: 442, Extreme: 222**

The most interesting and important aspect of this dataset is the structure of the dataset which cannot be overlooked and needs to be taken into account in all the stages of the Data Science pipeline.

Each row in the dataset pertains to an individual of a household who has participated in a survey and his/her responses have been recorded in the given columns. One of the individuals would be the head of the household and his/her poverty level would be used to assess whether our classification algorithm has correctly classified the household.

The dataset consists of features/attributes pertaining to an individual such as age, education etc. as well as for the household such as the presence of a ceiling, total members in the house etc.

Therefore, we have to consider attributes of the individual as well as the attributes pertaining to a household given in the dataset and the attributes that we can construct or aggregate from the individuals of the household.

One more important aspect that we would need to consider would be the high class imbalance wherein we have high number of Non-vulnerable class compared to other classes.

**Data Cleaning**

1. **Checking for Null Values**

Null Values found in 3 columns:-v2a1 (monthly rent), v18q1 (number of tablets owned), rezesc (years behind in school), meaneduc (mean education of 18+ individuals of a household in years).

**Steps Performed:**

* 1. Dropping rez\_esc (Years behind in School) column as 80% of the column has null values and we have no way of knowing or inferring these values from other columns.
  2. V18q1 had null values where the individual did not own a tablet which was given in v18q (binary) column which indicates if the owner has one or more tablets. Hence, replaced the null values with 0 where v18q column had 0 as a value.
  3. V2a1 column had 70% null values. Most of the null values were for the individuals who owned the house and did not have to pay any rent. This information was given in the tipovivi (ownership status) columns. Hence, substituted these null values with 0.A small percentage of null values where for the individuals whose ownership status was unknown or miscellaneous. Hence, imputed the null values with the mean rent.
  4. Meaneduc column had null values as well. The values were calculated by averaging the years of schooling of individuals over 18 from the escolari column.

1. **Cleaning the dependency Column**

Dependency = (number of members of the household younger than 19 or older than 64)/ (number of member of household between 19 and 64)

The dependency column is given by the above formula which would give a floating point number. However, inconsistent entries were found in the column e.g. string values (‘no’, ‘yes’), incorrect values not consistent with the values given by the formula.

Therefore, the dependency values were recalculated using the formula thereby cleaning the column of incorrect entries.

1. **Dropping one column between tamhog and hhsize**

It was found upon investigation that these two columns were exactly similar to each other, both of these columns indicating the total members in the household. Therefore, we need to drop one of these columns as one would be redundant and collinear with the other. Hence, dropping tamhog and proceeding with hhsize.

1. **Investigating the categorical columns to check for consistency**

We have a number of categorical columns pertaining to the attributes of the household.

For e.g. We have 'eviv1','eviv2' and 'eviv3' which are binary columns indicating the floor quality of the house which could be bad, regular or good, each column representing a specific value.

We need to check that at least one of these columns has a value of 1, representing a specific value.

Therefore, we investigated these categorical columns and found that columns representing source of electricity and material used for roof had inconsistent entries where none of the columns had a 1.

**Steps performed:**

* 1. This issue was addressed by introducing new columns which had a 1 indicating that source of electricity was other than specified in case of source of electricity and material used is other than specified values for roof material.
  2. Columns indicating education level of an individual had 3 rows where the level was unknown. Therefore, checked the escolari (Years of schooling) column to estimate the level of education and substitute a 1 in the respective column.

1. **Dropping the squared attributes**

The dataset contains columns containing squared values of the continuous features/columns. These squared features would be highly correlated with these continuous features. Therefore, I would be dropping these features to avoid redundancy. Deriving features from the existing ones would be taken care of in the feature engineering stage.

1. **Correcting the poverty levels/Target labels**

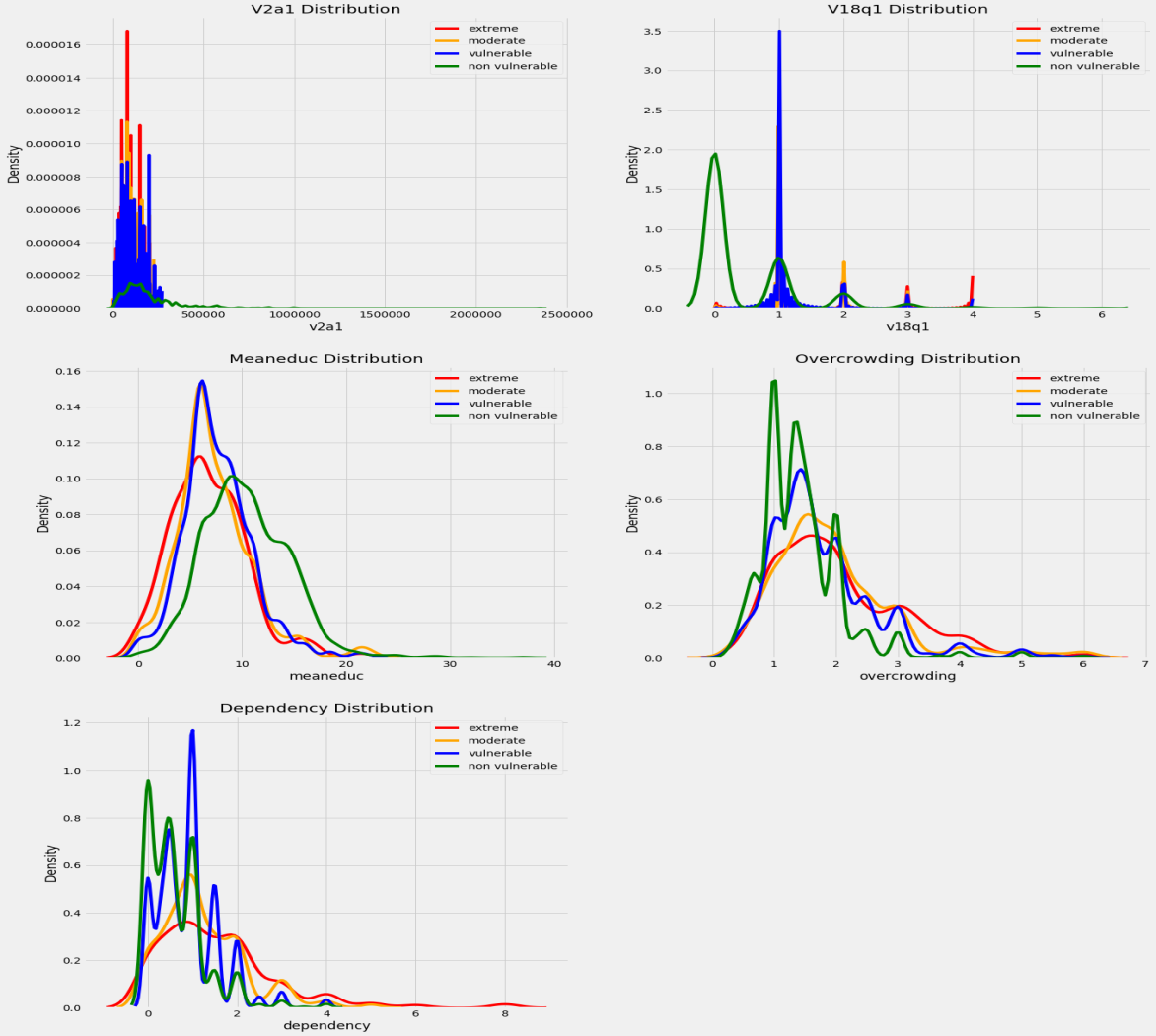
There were rows in the dataset where the individuals of a household had different poverty levels compared to the poverty level of the head of the household as only the labels of the head of the household would be used for scoring.

Therefore, the correct label was substituted for individuals of a household having a different poverty level than the head of the household.

**Exploratory Data Analysis**

Exploratory data analysis is an important part of our analysis as it helps us form a landscape of our dataset and enables us to look at our dataset from different perspectives and uncover hidden insights and make inferences from our data that would help us in the later stages of our pipeline.

**1. Investigating the Continuous features**

****

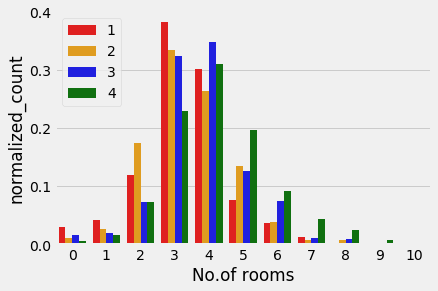
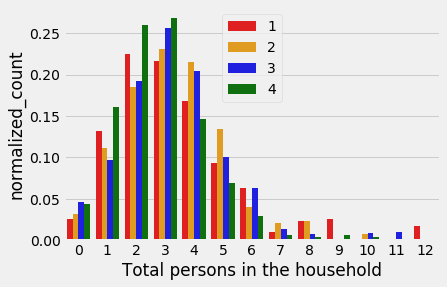
**Observations:**

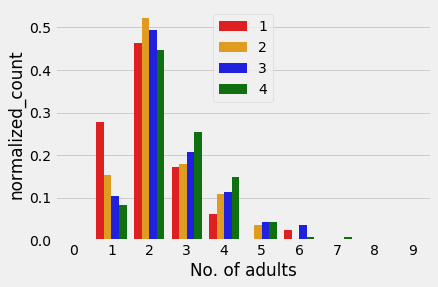
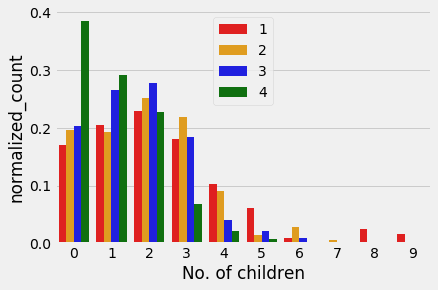
1. We can see a clear difference in the v18q1 column, where vulnerable households usually own a single tablet, whereas non-vulnerable households have a range of values. We can also see a small spike for extreme households at 4, which is quite unexpected for the number of tablets owned.

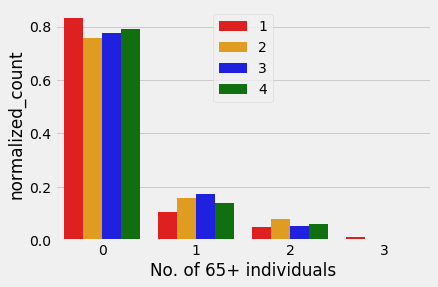
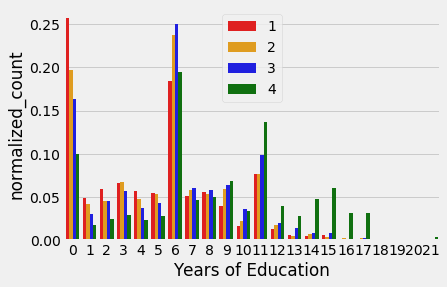
2. We can also see a difference in the overcrowding column where non-vulnerable households have spikes between 1 and 3 before its distribution tapers off, whereas we see that much of the area of the distribution covers values from 1 to 4 before tapering off.

3. We see that the non-vulnerable distribution in the meaneduc column is slightly shifted to the right than the other distributions, indicating a higher mean education for these households.

**2. Investigating the Numerical Columns**

** **

**** 

**Observations:**

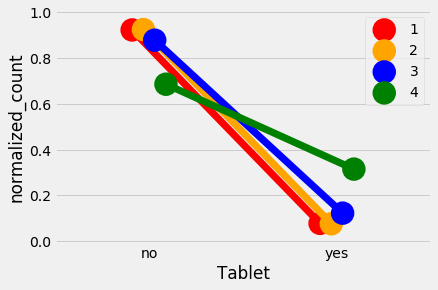
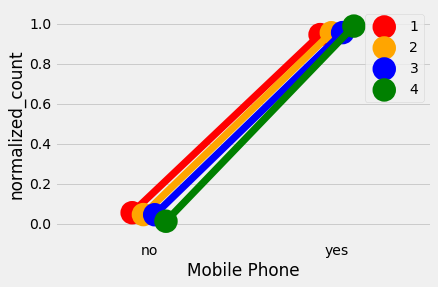
1. Non-Vulnerable households have a much larger range of rooms ranging from 0 to 11 whereas the other households have 3-4 rooms and 5-6 in some cases.

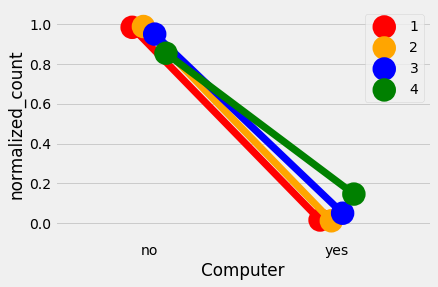
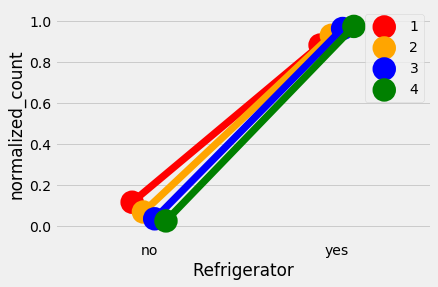
2. There are more number of children in extreme and moderate poverty households compared to the other poverty levels, sometimes 8 or 9 in some cases. Vulnerable and non-vulnerable poverty level households have up to 3 children after which the counts taper off sharply from 4.

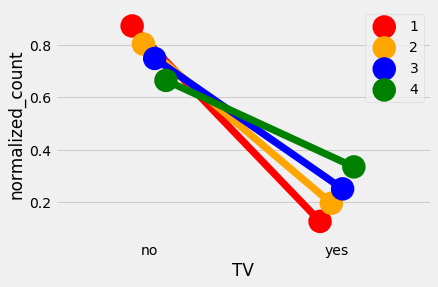
3. We observe three prominent spikes in the figure at year 0, 6 and 11 for all the poverty levels. We can also observe the fact that extreme poverty households have the highest count at year 0 followed by other poverty level households in descending order of the level. Only non-vulnerable households appear to pursue higher education from year 12 onwards.

**3. Investigating Refrigerator, TV, bathroom, tablet, mobile phone, computer Columns**

**Extreme, Moderate, Vulnerable, Non-Vulnerable**

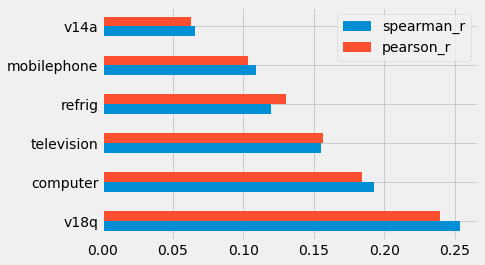
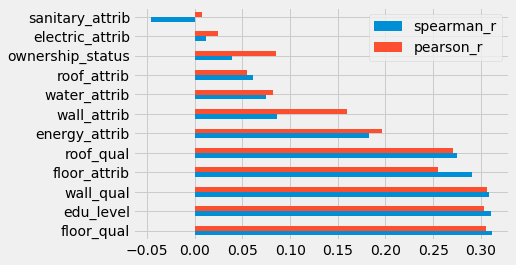
 

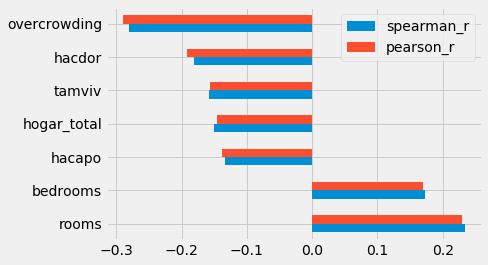
**Observation:**

1. We see a difference in the number of tablets owned by individuals belonging to different poverty levels. Non-vulnerable individuals own more tablets than other poverty levels.

2. We also see a difference in the TVs owned across poverty levels which could be a discriminating factor for classifying poverty levels on unseen data.

**4. Checking Correlation of columns with the Target**

** **

****

**Observation:**

1. Floor, education level, wall, roof and floor quality have a slightly positive correlation with the poverty level.

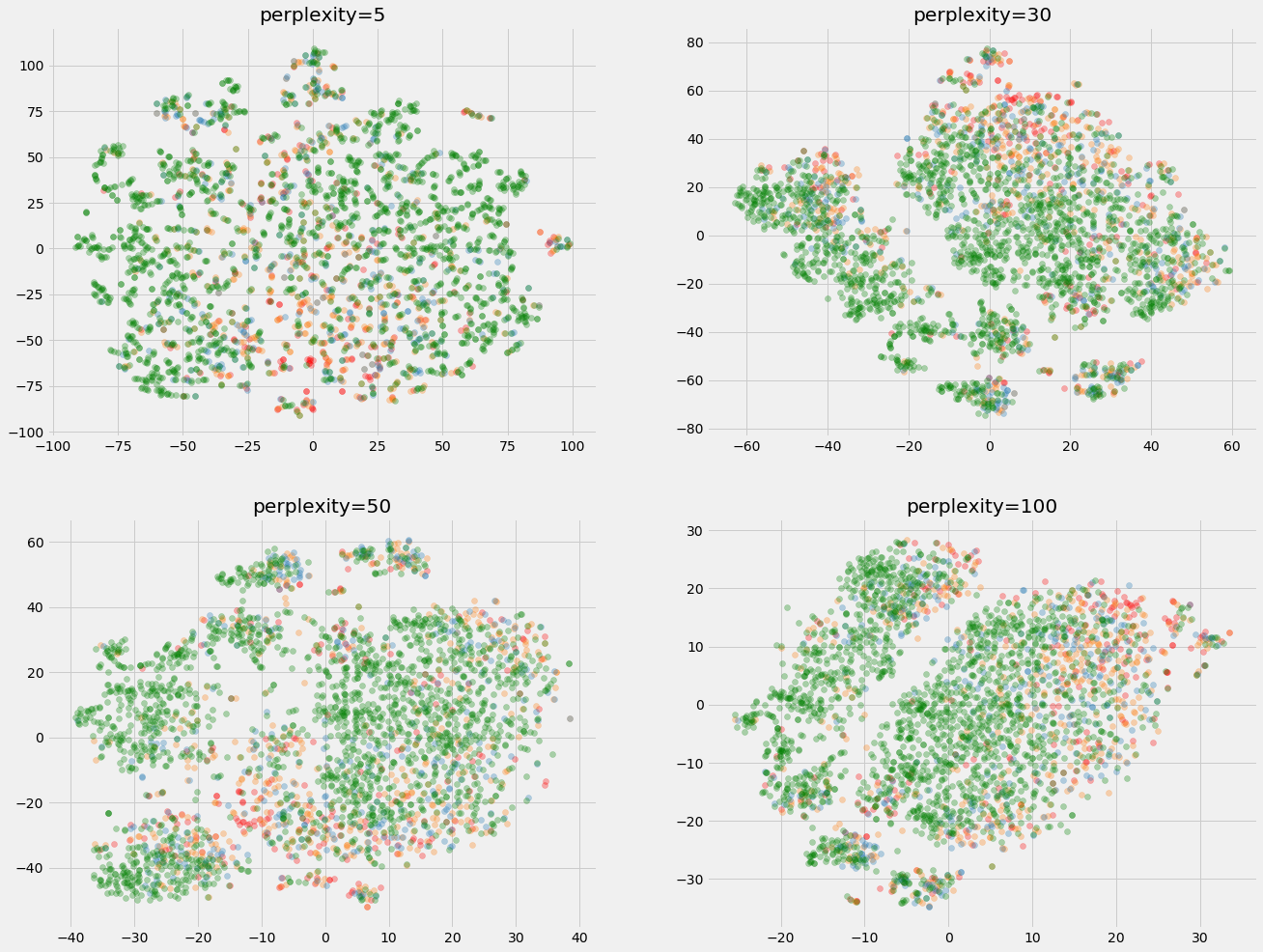
2. Overcrowding, hacdor, tamviv, hogar\_total, hacapo have a negative correlation with poverty level. This is understandable as these columns represent overcrowding, household size, persons living in each household etc. We can see that as the poverty levels begin to decrease from non-vulnerable to extreme, these features begin to increase.

Eg.extreme poverty levels tend to have more overcrowding whereas non-vulnerable households tend to have less overcrowding

3. Amenities like refrigerator, television, mobile phone etc. have a positive correlation with the poverty level. As the poverty level increases from extreme to non-vulnerable, we tend to see an increase in the households/individuals holding these amenities.

**Data Visualization**

Before proceeding to the feature engineering stage, we should try to visualize our dataset to help us understand the structure of our dataset, gauge if the classes look separable with the existing features and also observe the class imbalance in our dataset.



These visualizations were prepared using TSNE-visualization technique which embeds high dimensional data into a lower dimensional subspace making it easier for us to visually observe our data.

Perplexity value controls the tradeoff between dominance of local or global structure in our data. Generally, a value of 50 is a default when using TSNE for visualization; however it’s better if we try other values as well to compare the difference with different perplexity values.

**Observation:**

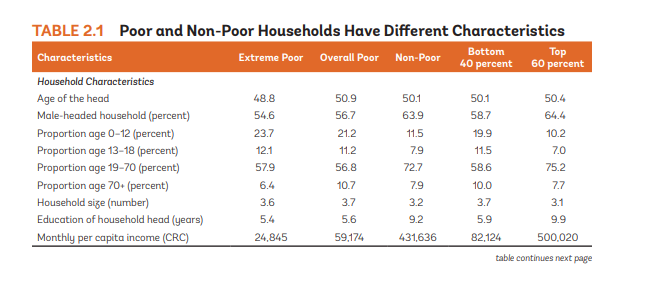
We can see from the visualizations that the classes don’t look separable in two dimensions and the minority classes look highly under-represented. This class imbalance needs to be addressed if we want our algorithms to better differentiate between the classes.

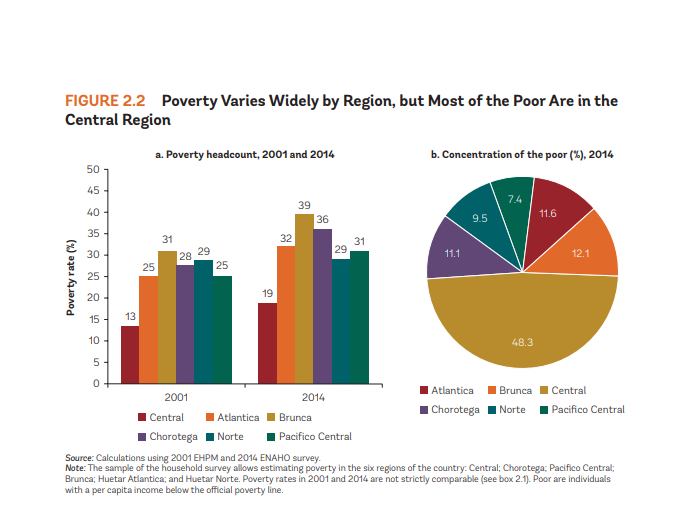
**Feature Engineering**

The next step in our pipeline would be to engineer features which would make our models better discriminate between our classes.

Therefore, to achieve this objective, I wanted to incorporate the knowledge that we discovered during our research.

Some of the key insights from the study that I think could prove useful are given in these images:-





We can observe that this would give us a good head start by creating these features and build up on them to create more informative features.

**Some of the new features created during feature engineering stage:-**

1. If the household head is female.

2. Proportion of children under 12, Proportion of adults, Proportion of elderly

3. Rent-per-person, Rent-per-room.

4. Proportion of males and females in the household.

5. Mean education of children under 12, Mean education of children 12-18

7. Standard of living score.

Score formula = **1/4(1/4(electricity) + 1/2(water) + 1/3(energy) + 1/5(sanitation) + 1/6(refrig + TV + mobile + tablet + computer + bathroom))**

This score would consider if the household has electricity, water, energy used for cooking, sanitation and amenities like refrigerator, TV, mobile phone etc. It would have a low score 0 and a max score of 1.

8. House quality score.

Score formula = **1/4(1/2(wall qual) x 1/6(wall\_attrib) + 1/2(floor qual) x 1/5(floor\_attrib) + 1/2(roof qual) x 1/4(roof\_attrib) + ceiling)**

This score would consider the floor, wall, roof quality and the material used along with the presence of ceiling in the house.

9. If the household has safe waste disposal facilities

10. If the head of household is single, divorced or married

11. Number of gadgets, gadgets per person

12. Aggregations of above features (mean, min, max, sum, std, range)

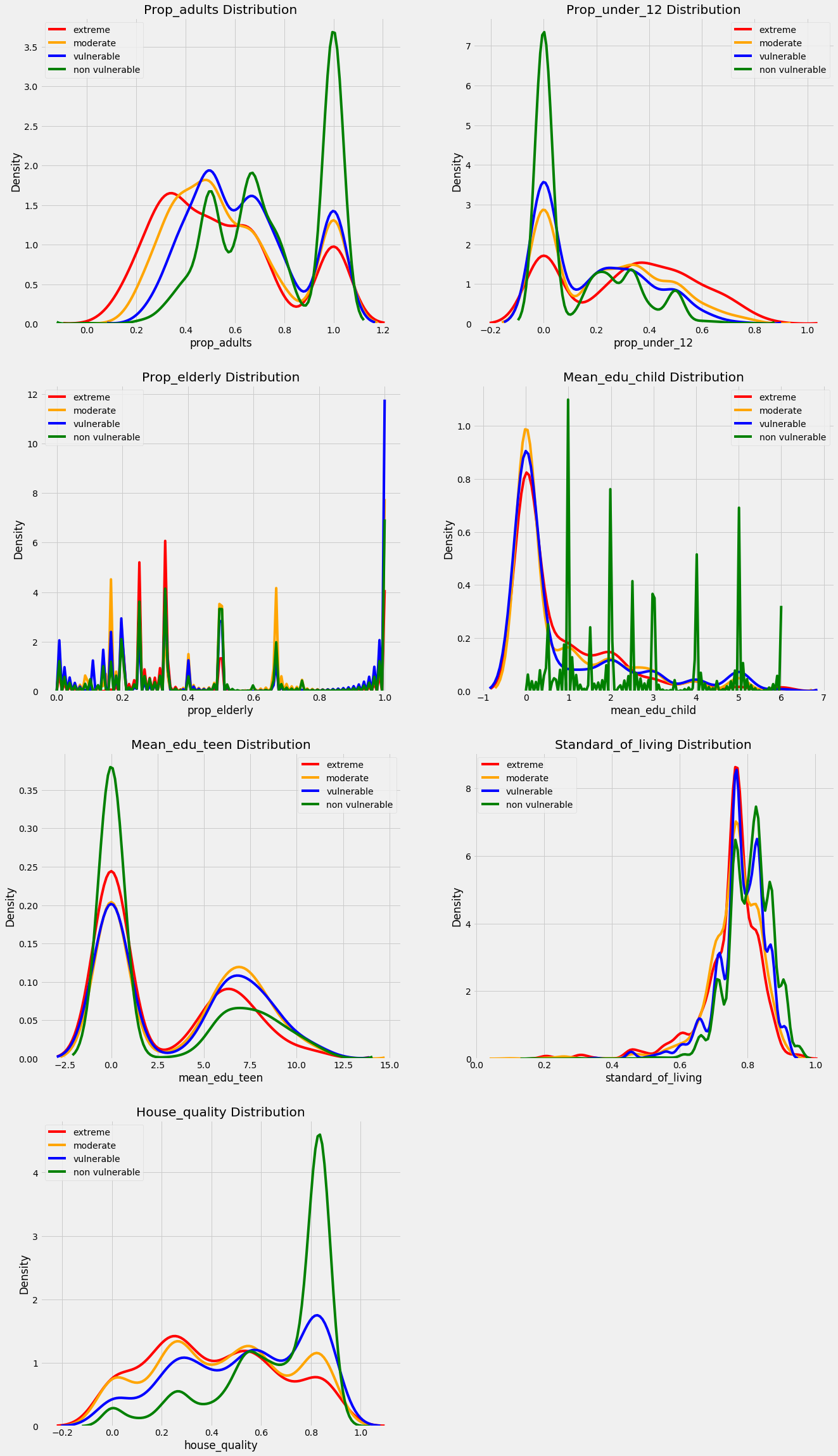
**Training and Test Data after feature engineering**

**Training Data: 9557** rows **115 columns**

**Test Data: 23856** rows **114 columns**

**Data Analysis post Feature Engineering**

We performed data analysis post feature engineering to analyze the new features and try to investigate if these features look informative and would be able to help differentiate between classes.

**Investigating engineered features**

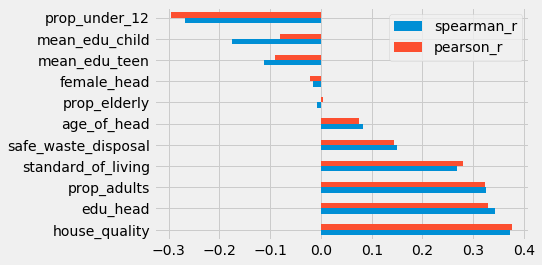
**Observations:**

1. If we observe the proportion of adults distribution across poverty levels, non-vulnerable households have a high density of adults compared to children and elderly members in the family which could mean lower dependence or financial pressure on the adult/working members of the family.

2. We can observe a distinct pattern in the proportion of children density plot where most of the non-vulnerable households have less number of children per family, followed by vulnerable, moderate and poor households.

3.We see a similar pattern in the house quality plot as well where the non-vulnerable households have a higher score followed by vulnerable, moderate and poor households.

**Correlation with the Target:**

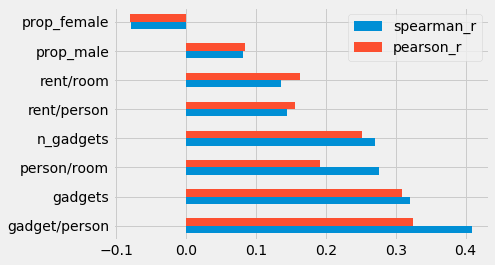
****

**Observation:**

1.We can observe that features like safe waste disposal, standard of living, proportion of adults, house quality etc. have a positive correlation with the target which means that we tend to see an increase in these attributes of a household as we move from a poor to a non-vulnerable household.

2. We also see a negative correlation with certain attributes like proportion of children, education of children etc.

3. These correlations have significant p-values as well.

****

**Observation:**

1.All but one of the attributes have positive correlation with the Target(poverty levels).The highest correlated attribute appears to be the number of gadgets like mobile, computer, tablets owned per person in a household.

**Approach**

We have completed the feature engineering and analysis stage, and therefore would be moving on to the modelling stage where we would be performing the following steps:

**1) Selecting appropriate/relevant samples from data for training.**

**2) Preprocessing/scaling of data**

**3) Choosing a scoring metric for evaluation**

**4) Experimentation of different machine learning models with and without oversampling.**

**5) Consolidation, comparison and interpretation of results.**

**Selecting appropriate/relevant samples from data for training**

Due to the given structure of our dataset where we have two granularities to consider, individual data in each row and household data which can be grouped using household level identifier. Additionally, only heads of the household will be considered for scoring.

Since we have already engineered features at an individual as well as on the household level, hence, we would only be using heads of the household for training and validating our ML models

**Preprocessing/Scaling Training and Test data**

We would be preprocessing/scaling our data to have **0 mean and unit variance** to avoid some of the features dominating over other features by the sheer scale of their values and biasing the model towards this feature and providing incorrect weights or hurting the final score.

**Scoring metric/s for evaluation**

We would be using **macro F1-score** as our primary scoring metric for our machine learning models as it is the metric that would be used for scoring our test predictions when submitting to the Kaggle competition.

We would also be using other metrics such as **ROC-AUC curve, confusion matrix** to help us understand our models’ performance from various perspectives.

**Machine Learning Models**

We would be trying different classes of machine learning models:-

**Non-parametric models**: - SVM and K-nearest neighbors

**Parametric Models**:-Logistic Regression

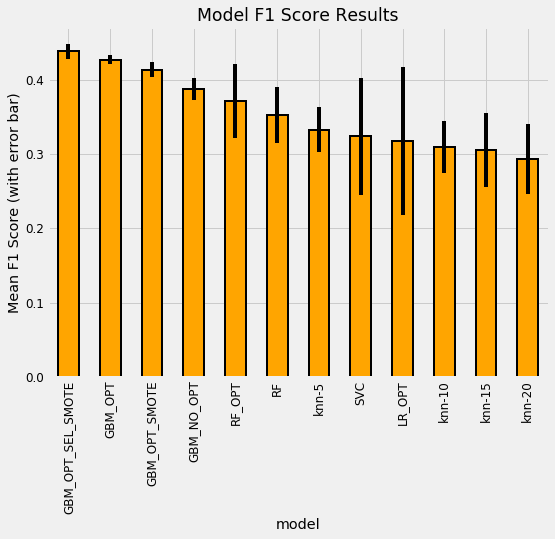
**Ensemble models**: - Random Forest and Extreme Gradient Boosting Trees

**Model Tuning using Bayesian Optimization**

We would be performing model optimization or tuning model hyperparameters using Bayesian optimization with cross-validation using hyperopt package.

We are using this optimization method over grid-search and random search methods as it uses observational history in searching the parameter space for deciding the best set of parameters that maximize or minimize the scoring metric being used.

**Results**

****

**Model results bar chart showing the cross validated macro f1 score of each model and their variation with vertical bars showing standard deviation/variability in scores.**

**Observation:**

1. Among the parametric models, KNN and SVC, KNN-5 gave the highest validation score, with less variability than the SVC model.
2. Logistic regression scored higher than the other KNN models, but with a lot of variability in scores across the folds.
3. Ensemble models were much better in performing this task, even without optimization which explains the superiority of ensemble models on well behaved and structured datasets.
4. We can also notice the gains obtained when we oversample our minority classes with Gradient boosting ensemble model.
5. We obtain a further improvement, a slight increase in our score when we perform feature selection based on the feature importances returned by our GBM model with oversampling. This is our final model which gave us the best score.

**Conclusion**

This project gave me a glimpse of how real world datasets could be, interesting, unique and quite messy requiring a careful understanding of the underlying structure and working our way up towards the other stages in the data pipeline.

I also understood that working on a project is an iterative process, where we get to the final stages of the pipeline, and based on the result, we iterate and go back to fix something or apply a new technique or start again altogether. For instance, I had to go back to the feature engineering stage and rethink ways to engineer new features to improve my score, keeping the public leaderboard scores as a reference. Although, this might not be the case in a real world scenario. Therefore, it is important to try several approaches to solve a problem, starting from simpler models and then work our way up to more sophisticated and complex models where the simpler models could serve as a benchmark when we are trying more complex models.

I learnt a new way of hyper parameter optimization called Bayesian optimization, which is a much more improved way of tuning the hyperparameters of our model using the Bayes’ theorem to converge on the best parameters for the model. One of the drawbacks could be overfitting, which could be reduced if we use cross validation.