Analysis: 35 /40
Narrative: 30/35
Predictions: 10/10

STAT443 Final Project

Maggie Wesolowski, Maggie Grethel, Jayden Koenig, Sarah Rodriguez

2024-12-15

Introduction

The goal of this project is to classify households into one of four poverty levels based on a range of socioeconomic and demographic variables. The dataset provides both household and individual level variables. Some example variables include: the status of the flooring of a house, number of individuals living in a house, house location region, and electricity and toilet status in a house. We will be focusing on the heads of household to develop a predictive model that balances interpretability with predictive power. This classification can help identify vulnerable groups and guide social policies. Key considerations include handling missing data and ensuring appropriate feature representation.

Exploring Data

During exploration, we identified missing values in several variables, such as v2a1 (monthly rent) and rez_esc (years behind in school). Missing values in v2a1 often occur when households own their homes and do not pay rent, suggesting that these missing values can reasonably be imputed as 0. For other variables with missing data, we need to determine if the data is missing at random. Depending on the nature of the missingness, strategies such as mean/mode imputation or introducing a new “unknown” category will be applied.

Exploring Missing Values

- If the value of v18q is 0, then the value in v18q1 is NA. Therefore, we convert NAs in v18q1 to be 0, meaning the number of tablets a household owns in 0.
- The missing values in v2a1 (monthly rent) are because the house is owned and fully paid. When tipovivi1 is 1, v2a1 is NA.

Fixing Coding Error

- ‘Edjefe’ is supposed to be the years of education of male head of household. The variable should only contain numeric values for male heads of household. However, it contains ‘yes’ and ‘no’. With ‘no’ meaning that person is neither a male nor the head of household. And ‘yes’ being that person is male and head of household. This is the same for the variable ‘edjefa’.
- There is 120 cases where ‘edeje’ and ‘edjefa’ were coded incorrectly. To fix this we check where that inconsistency is and change it. For instance, if there is a ‘yes’ in ‘edjefe’ but that person is neither male nor head of household, the ‘yes’ is

change to a ‘no’. And we will do the same for the ‘edjefa’ variable to ensure that the data is consistent.

- Now that the variables have been corrected and are consistent, the next thing we need to do is replace the ‘yes’ in each column with the years of education from the ‘escolari’ variable. If there is a ‘no’ then the value will be NA. In this case, it makes sense to have a missing value because it is not relevant to know the years of education if that person does not meet the requirements for that variable.

Fixing Redundencies

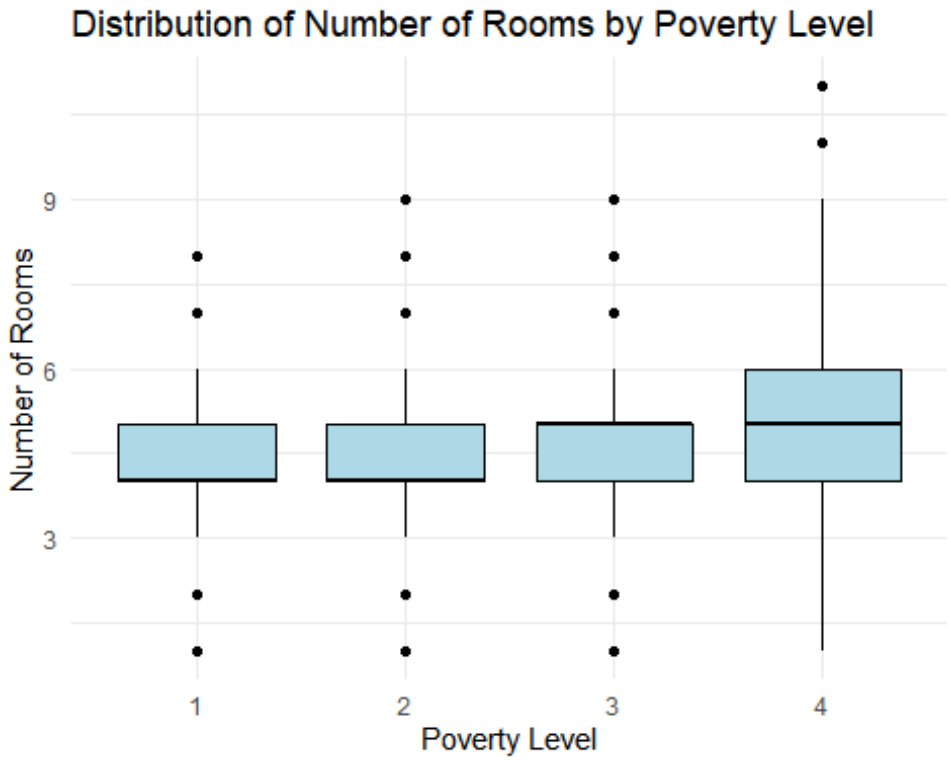
Having separate columns for female and male is redundant, so we made one column for gender with male = 1 and 0 = female.

Household vs. Individual-Level Variables

The data includes variables at two levels:

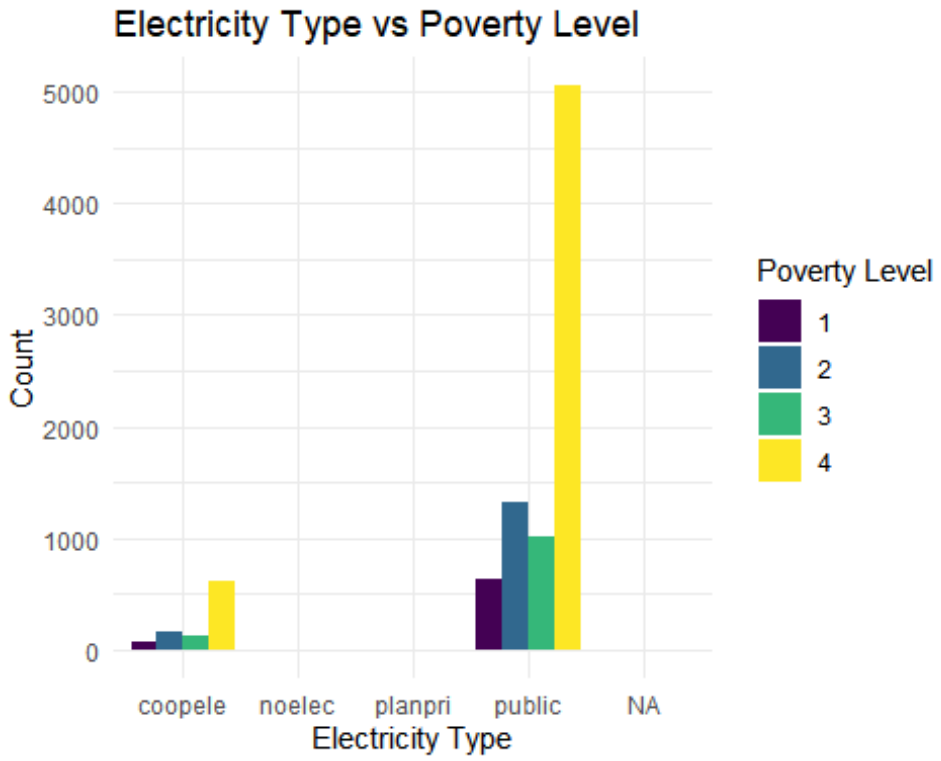
- Household-level: Variables like rooms, electricity type, and toilet type describe the collective living conditions of the household.
- Individual-level: Variables like age, gender, and years of education provide details about each household member.
  Since the focus is on heads of household, individual-level data for non-heads was aggregated to derive household-level insights.
    - For example: The number of children, adults, and elderly members in a household was calculated.
- Average years of schooling for adults was used as a measure of household educational attainment.

Household-Level



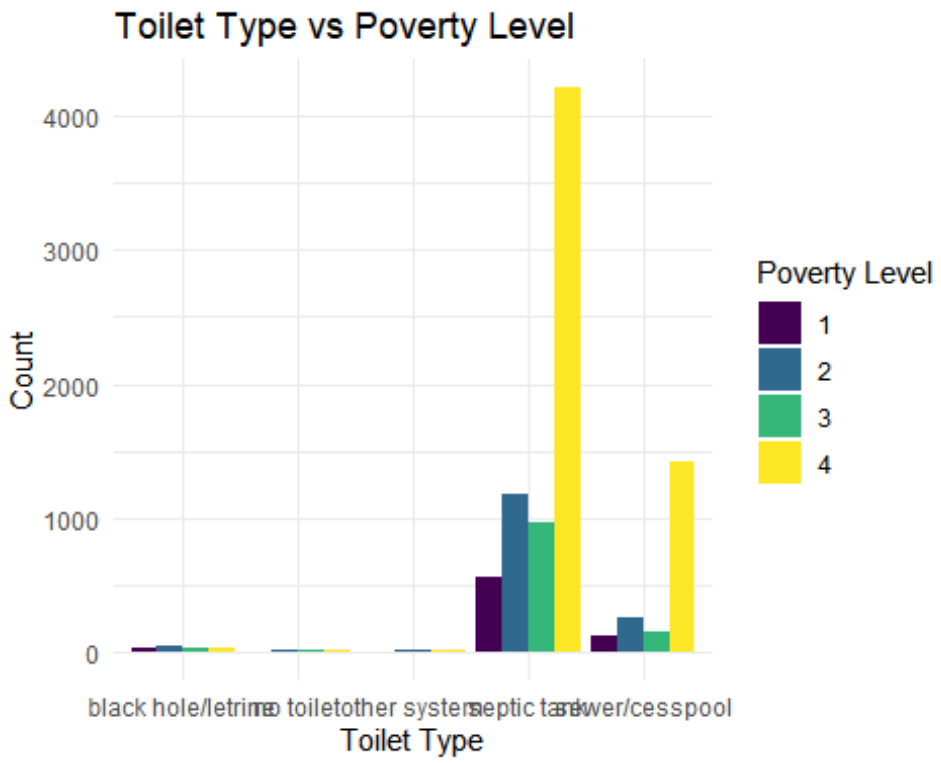
Poverty levels 3 and 4 had a higher average number of rooms being in being closer to 5 and levels 1 and 2 being 4.

We created a new variable electricity_type for the following variables: public, noelec, planpri, and coopele. Now we can see the distribution of for each electricity over the different poverty levels.



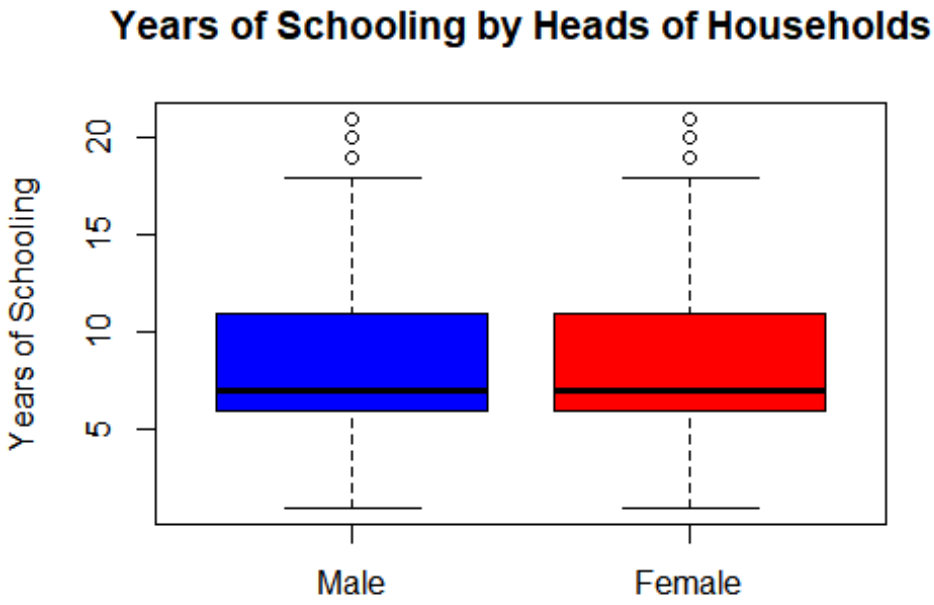
Cooperative	No Electricity	Private Plant
991	20	3

The most common type of electricity is public and based off the graph, it is more likely to be classified as a non-vulnerable household if you have public electricity and electricity from cooperative. There is not enough data for no electricity or private plant electricity to make an assumption.



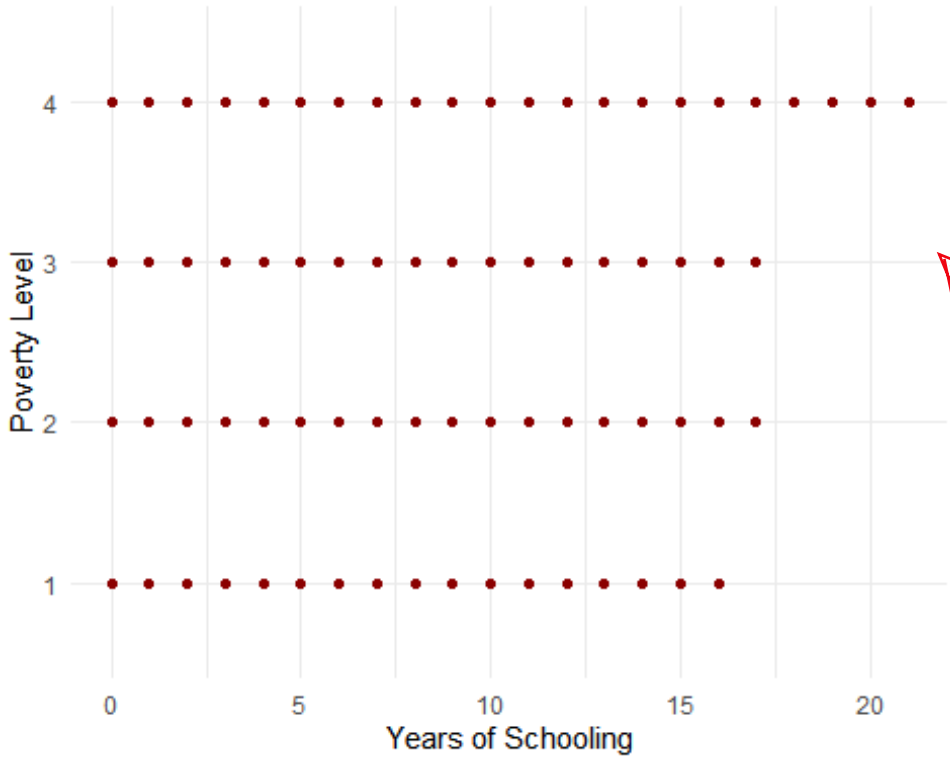
The most common type of toilet is septic tank and sewer. Based off the graph, it is more likely to be classified as a non-vulnerable household if you have those two types of toilets. There is not enough data for other types of toilet.

Individual Level



Statistic	Female	Male
Minimum	25	40
1st Quartile	30	30
Median	50	10
Mean	8.412	8.505
3rd Quartile	11	11
Maximum	21	21
NA’s	5957	3626

This shows that the distribution of years of schooling is relatively the same for both male and female head of households.



It does not appear that years of schooling has a major impact on the classification of one's poverty level.

Correlated Variables

Exploration revealed correlations among some variables. For instance:

- Rooms and bedrooms are highly correlated, as both describe the household's living space.
- tamviv (household size) and hogar_total (total individuals in the household) are similar and provide redundant information.
  To handle these correlations, we selected one representative variable.

Variable Importance

Key variables likely to influence poverty classification include:

- Living conditions: Number of rooms, overcrowding rate, and toilet type.
- Assets: Presence of a refrigerator, computer, or mobile phone.
- Education: Average years of schooling for adults and the education level of the head of household.
- Location: Urban/rural classification and region of residence.

```
#model using the key variables found during data exploration

model = glm(target ~ toilet_type + rooms + computer + overcrowding+ refrig + mobilephone + meaneduc + escolari + lugar1 + lugar2 + lugar3 + lugar4 + lugar5 + area1 + area2, family = binomial,data=p.tr)
```

Practical Insights

Understanding the socioeconomic conditions of non-head household members was valuable in summarizing household dynamics. For example, the number of dependents (children and elderly) relative to working-age adults provides a clearer picture of household dependency, which is a critical factor in poverty classification.
This data exploration phase lays the groundwork for creating a predictive model by addressing missing data, selecting relevant features, and ensuring the dataset is well-prepared for training and testing.

Model Selection

The model selection process considered the nature of the data, the ordinal structure of the target variable, and the project's goals of interpretability and accuracy.

Models Considered

The primary considerations for selecting the best model included:

- **Predictive Performance:** Measured using accuracy, F1 score, and metrics that account for the ordinal nature of the target variable.
- **Interpretability:** Priority was given to models that provide insight into the relationships between features and poverty levels, aiding practical decision-making.
- **Handling of Missing Data and Multicollinearity:** Models were assessed based on their ability to handle missing values and highly correlated features.

We first used multinomial logistic regression for the multi-class classification task. Our accuracy was 0.679 and a low F1 score of 0.178 revealed that it would be useful to look at the class imbalance. Class counts revealed there are significantly more people in non-vulnerable households than in vulnerable households or poverty. We also noticed that our model predicted a 4 for an outcome when it was really 1,2, or 3. It was predicting a non-vulnerable household when in reality it was a vulnerable or poverty household. Our model also often predicted a 2 when it was really a 1,3, or 4.

```
set.seed(234)
#multinom model for training dataset
model1 <- multinom(target ~ toilet_type + rooms + computer+ hogar_total + overcrowding + refrig  + mobilephone + escolari + lugar1 + lugar2 + lugar3 + lugar4 + lugar5+area1+area2+v2a1+r4h3+r4m3+r4t3+paredblolad+paredzocalo+paredpreb+paredmad+pisomoscer+pisocemento+techocane+cielorazo+etecho1+eviv1+dependency , data=p.tr)
```

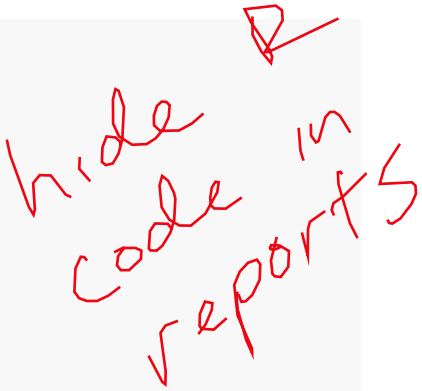
```r
#testing data
probabilities1 <- predict(model1,  newdata = p.te, type = "probs")
predictions1 <- predict(model1,newdata=p.te)
head(predictions1)
accuracy1 <- mean(predictions1 == p.te$target)
accuracy1

f1_1=F1_Score(predictions1,p.te$target)
f1_1

#class counts
length(which(p.tr$target==4))
length(which(p.tr$target==3))
length(which(p.tr$target==2))
length(which(p.tr$target==1))

conf_matrix1 <- table(predictions1, p.te$target)
print(conf_matrix1)

##
## predictions1    1    2    3    4
##            1   18    7    4    8
##            2   57  141   59   61
##            3    5   14   22    8
##            4   85  231  188 1357
```



Once the multinomial model revealed for us that we should take class imbalance into account, we moved on to explore other models:

- Logistic Regression: A baseline model with high interpretability.
- Random Forests: An ensemble method that reduces overfitting and captures complex interactions between variables while providing feature importance metrics.

```r
set.seed(234)

model2 <- randomForest(target ~ toilet_type + rooms + computer+ hogar_total + overcrowding + refrig + mobilephone + escolari + lugar1 + lugar2 + lugar3 + lugar4 + lugar5+area1+area2+v2a1+r4h3+r4m3+r4t3+paredblolad+paredzocalo+paredpreb+paredmad+pisomoscer+pisocemento+techocane+cielorazo+etecho1+eviv1+dependency, data = p.tr, ntree=50)

#Predictions on the test dataset
predictions2 <- predict(model2, newdata = p.te, type = "class")

#confusion matrix
conf_matrix2 <- confusionMatrix(predictions2, p.te$target)
as.matrix(conf_matrix2,what="classes")
```

```
##                          1         2         3         4
## Sensitivity          0.70909091 0.7099237 0.71428571 0.9790795
## Specificity          0.99428571 0.9818376 0.98493976 0.7665463
## Pos Pred Value       0.90697674 0.8913738 0.86666667 0.8785982
## Neg Pred Value       0.97752809 0.9415984 0.96176471 0.9550225
## Precision            0.90697674 0.8913738 0.86666667 0.8785982
## Recall               0.70909091 0.7099237 0.71428571 0.9790795
## F1                   0.79591837 0.7903683 0.78313253 0.9261214
## Prevalence           0.07284768 0.1735099 0.12052980 0.6331126
## Detection Rate       0.05165563 0.1231788 0.08609272 0.6198675
## Detection Prevalence 0.05695364 0.1381898 0.09933775 0.7055188
## Balanced Accuracy    0.85168831 0.8458806 0.84961274 0.8728129
```

```
accuracy2 <- mean(predictions2 == p.te$target)
accuracy2
```

```
## [1] 0.8807947
```

```
#our F1 score with random forest increases from 0.1782178 on our multinomial model to 0.7959184 for random forest
f1_2=F1_Score(predictions2,p.te$target)
f1_2
```

```
## [1] 0.7959184
```

Random Forest Model for Training Dataset

We found that adding the total number of individuals in the household and monthly rent payment improved our accuracy of our model. Adding total males in household, total females in household, and total people in the household also raised our accuracy. The material of wall, floor, and roof helped increase our model's accuracy. Our F1 score with random forest increases from our multinomial model to 0.796 for Random Forest.

```
set.seed(234)
#Handling class weights
#https://medium.com/@ravi.abhinav4/improving-class-imbalance-with-class-weights-in-machine-learning-af072fdd4aa4
class_weights= c(6792/(4*4254), 6792/(4*879), 6792/(4*1109), 6792/(4*550))
model3 <- randomForest(target ~ toilet_type + rooms + computer+ hogar_total + overcrowding + refrig + mobilephone + escolari + lugar1 + lugar2 + lugar3 + lugar4 + lugar5+area1+area2+v2a1+r4h3+r4m3+r4t3+paredblolad+paredzocalo+paredpreb+paredmad+pisomoscer+pisocemento+techocane+cielorazo+etecho1+eviv1+dependency, data = p.tr, classwt=class_weights)

#Predictions on the test dataset
predictions3 <- predict(model3, newdata = p.te, type = "class")

#confusion matrix
conf_matrix3 = confusionMatrix(predictions3, p.te$target)
conf_matrix3
```

```
## Confusion Matrix and Statistics
##
##          Reference
## Prediction    1    2    3    4
##          1  113    2    0    1
##          2   16  304    7   17
##          3    5   11  209   15
##          4   31   76   57 1401
##
## Overall Statistics
##
##               Accuracy : 0.8949
##                 95% CI : (0.8816, 0.9073)
##    No Information Rate : 0.6331
##    P-Value [Acc > NIR] : < 2.2e-16
##
##                  Kappa : 0.7978
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3 Class: 4
## Sensitivity           0.68485   0.7735  0.76557   0.9770
## Specificity           0.99857   0.9786  0.98444   0.8026
## Pos Pred Value        0.97414   0.8837  0.87083   0.8952
## Neg Pred Value        0.97580   0.9537  0.96840   0.9529
## Prevalence            0.07285   0.1735  0.12053   0.6331
## Detection Rate        0.04989   0.1342  0.09227   0.6185
## Detection Prevalence  0.05121   0.1519  0.10596   0.6909
## Balanced Accuracy     0.84171   0.8761  0.87500   0.8898
```

```
#find the f1 score for each target value
as.matrix(conf_matrix3,what="classes")
```

```
##                           1         2         3         4
## Sensitivity          0.68484848 0.7735369 0.76556777 0.9769874
## Specificity          0.99857143 0.9786325 0.98443775 0.8026474
## Pos Pred Value       0.97413793 0.8837209 0.87083333 0.8952077
## Neg Pred Value       0.97580270 0.9536700 0.96839506 0.9528571
## Precision            0.97413793 0.8837209 0.87083333 0.8952077
## Recall               0.68484848 0.7735369 0.76556777 0.9769874
## F1                   0.80427046 0.8249661 0.81481481 0.9343114
## Prevalence           0.07284768 0.1735099 0.12052980 0.6331126
## Detection Rate       0.04988962 0.1342163 0.09227373 0.6185430
## Detection Prevalence 0.05121413 0.1518764 0.10596026 0.6909492
## Balanced Accuracy    0.84170996 0.8760847 0.87500276 0.8898174
```

```
accuracy3 <- mean(predictions3 == p.te$target)
accuracy3
```

```
## [1] 0.8949227
```

```
f1_3=F1_Score(predictions3,p.te$target)
f1_3

## [1] 0.8042705
```

Handling of Imbalanced Data

The dataset's target variable exhibited imbalanced class distributions, with some poverty levels being underrepresented. To address this:

- **Class Weights**: Class weights were calculated based on the inverse frequency of each class and applied during model training. We found that applying class weights only brought up the accuracy by about 1% where we took the inverse frequency of the counts to account for the class imbalance.
- **Evaluation Metrics**: Metrics such as the weighted F1 score and Cohen's kappa were prioritized to ensure fair assessment across all classes.
- **Avoidance of Undersampling**: Undersampling techniques were avoided to prevent loss of valuable information from the majority class.

The Random Forest model provided interpretable insights into the underlying patterns of poverty through feature importance rankings. Key findings include:

- **Education and Assets:** Households with fewer years of schooling and lower levels of asset ownership were more likely to be classified into higher poverty levels.
- **Household Structure:** Larger households with higher dependency ratios and overcrowding were strongly associated with higher poverty risk.
- **Geographical and Housing Conditions:** Features such as household location (urban vs. rural) and housing conditions (e.g., type of roof, floor materials) also contributed significantly to poverty classification.

```
#Predicting target values for the poverty test blinded dataset

predictions3_test <- predict(model3, newdata = poverty_test, type = "class")
poverty_test$target=predictions3_test
write.csv(poverty_test, "poverty-test-blinded2.csv", row.names = FALSE)
```

Results Summary

The goal of this analysis was to classify households into one of four poverty levels based on a set of socioeconomic and demographic variables. After evaluating multiple models and selecting the best one based on predictive performance and interpretability, our results follow:

Model Performance

The final model selected was Random Forest. We assessed its performance using multiple metrics, considering both accuracy and the ordinal nature of the target variable.

- **Accuracy**: The model achieved an overall accuracy of 0.8949 on the test set, indicating a strong ability to correctly classify households into the appropriate poverty categories.
- **Precision, Recall, and F1 Score:** These metrics were calculated for each of the four poverty levels to ensure the model performs well across all classes. The F1 score for the most underrepresented category (e.g., “extreme poverty”) was 0.8043, suggesting the model handles imbalanced classes effectively.
- **Confusion Matrix:** The confusion matrix revealed that the model was particularly effective at distinguishing between classes 2 and 4, which had high sensitivity (0.7735 and 0.9770, respectively) and balanced accuracy (0.8761 and 0.8898, respectively). However, class 1 had lower sensitivity (0.6848), indicating challenges in correctly identifying households in this category. Misclassifications in this class may be due to overlapping feature values with other categories.

Wow!

Model Interpretability

While the selected model provided strong predictive performance, it also offered insights into the underlying patterns of poverty through its feature importance rankings. For example, households with fewer years of schooling and lower levels of asset ownership were more likely to be classified into higher poverty levels. Similarly, larger households with higher dependency ratios and overcrowding were associated with higher poverty risk. These insights can help inform targeted social interventions aimed at improving education and access to resources.

Nicely done!