STAT 486: Final Report

Madison Wozniak

Table of Contents

Introduction

Problem Statement

Food affordability in the Unites States has reached an all-time low in recent years, and is a struggle that many individuals and families are no stranger to. This is a critical issue for single mothers especially, as it impacts their household's stability, health, and overall quality of life. In California, the average cost of living remains just barely achievable for many, and women-headed households face unique economic challenges because of it. An analysis of the food affordability for single-mother families in California can serve as a starting point for economic understanding, and opportunities for political intervention.

Aim of this Report

This report aims to understand if food affordability for single mothers in California can be accurately explained and predicted by socioeconomic and demographic factors. By looking at data that includes information on food costs, annual income, regional demographics and economic indicators, this analysis seeks to accomplish the following:

1. Understand the relationship between income, location, ethnic group and food affordability.
2. Develop predictive machine learning models on food affordability to income ratios.
3. Address how these insights can influence policy changes and improve economic outcomes in the future.

Exploratory Data Analysis

To fully understand the important factors that may influence affordability, we must first look at the raw data. This section gives an overview of the key variables by identifying relevant distributions and variations within the data. These summary statistics act as a foundation for deciding what the potential predictors of affordability could be.

The dataset includes the following features:

- `affordability_ratio` (numeric - target variable) Ratio of food cost to household income
- `median_income` (numeric) Median household income
- `race_eth_name` (string) Name of race/ethnic group
- `geotype` (string) Type of geographic unit place (PL), county (CO), region (RE), state (CA)
- `geoname` (string) Name of geographic unit
- `county_name` (string) Name of county the geotype is in
- `region_code` (string) Metropolitan Planning Organization (MPO)- based region code
- `cost_yr` (numeric) Annual food costs
- `ave_fam_size` (numeric) Average family size - combined by race

With the variables defined above, the next step is to delve into the statistical properties of each, which includes their underlying distributions and variations. It is important to be aware of any potential outliers or missing data before selecting predictive features for future models.

	region_code	cost_yr	median_income	affordability_ratio	ave_fam_size

	region_code	cost_yr	median_income	affordability_ratio	ave_fam_size
count	295362	264366	101836	101836	280593
mean	11.388	7602.632	38038.998	0.325	3.294
standard deviation	3.341	1435.062	27050.591	0.42	0.636
minimum	1	3095.425	2500	0.021	1.36
Q2	9	6667.128	22417	0.153	2.88
median	14	7460.84	33103	0.227	3.25
Q4	14	8325.618	46418	0.349	3.61
maximum	14	16872.05	250000	4.852	7.2

Table 2: Numeric Data Summary Satistics - rounded for readability

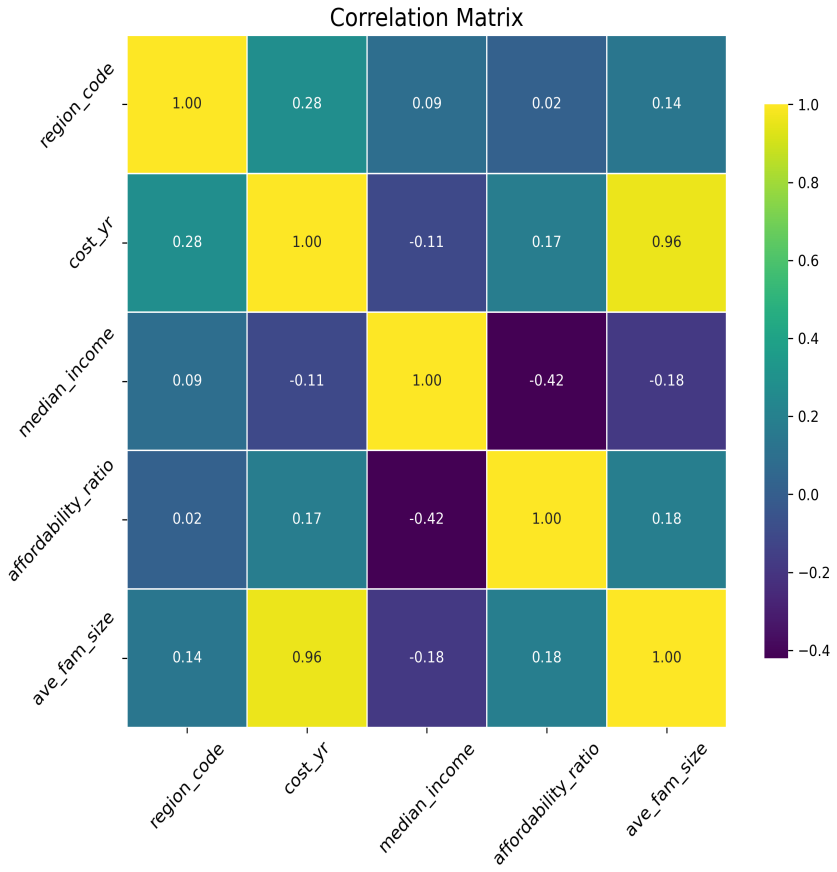

Figure 1: Correlation Matrix for Numeric Features

It can be noted from Figure 1 that there is a strong, almost perfect correlation between several variables, namely `cost_yr` and `ave_fam_size`.

While not overwhelmingly apparent in the correlation matrix, there is also an important relationship between `median_income` and `ave_fam_size` that Figure 2 below illustrates.

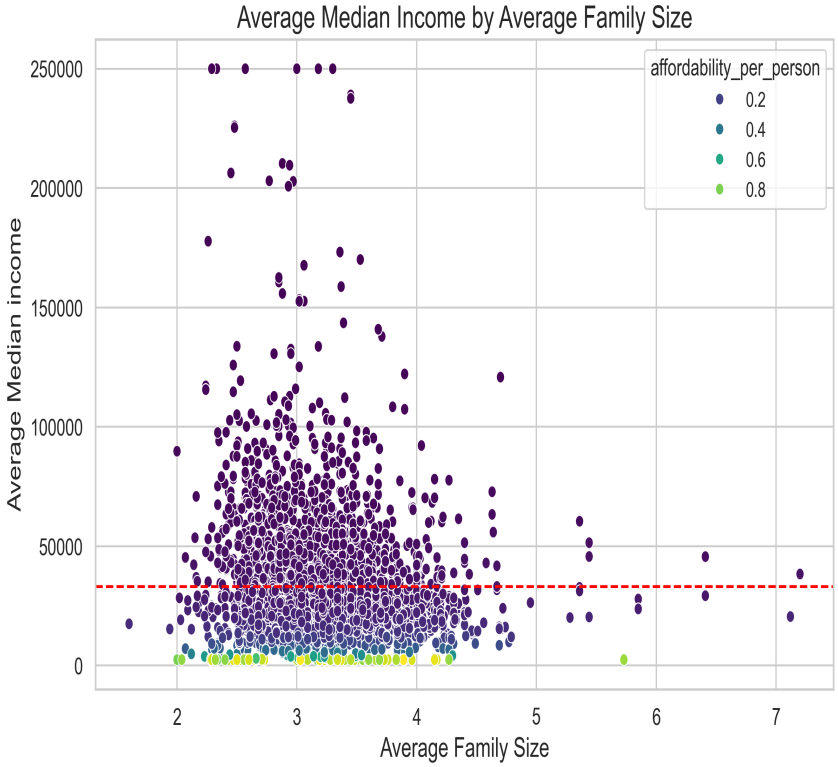

Figure 2: Scatterplot of Income Against Family Size

Figure 2 reveals a particularly interesting trend: while the spread of the points seems to suggest that most women-headed households earn enough to afford food for their families, the red line superimposed on the graph highlights the unfortunate reality. The median annual income in this dataset is $33,103.00. This is significantly below the median annual household income in the United States which stood at $80,610.00 as of 2022.

Although a majority of the data in this report is numeric, there are some categorical features that can still be examined. Table 3 below includes some summary statistics on these features.

	race_eth_name	geotype	geoname	county_name
count	295371	295371	295371	295236
unique	9	4	1581	58
top	AIAN	PL	Franklin CDP	Los Angeles
frequency	32819	298431	990	123966

Table 3: Categorical Data Summary Satistics

By combining the numeric and categorical features together for EDA some key relationships not captured by the correlaiton matrix become apparent in the graphs below.

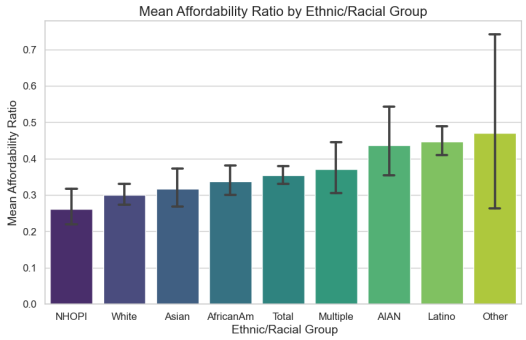

Figure 3: Affordability by Ethnic Group

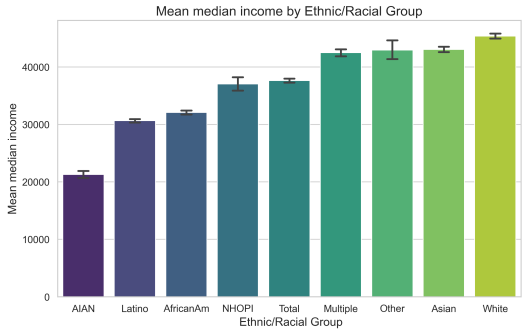

Figure 4: Income by Ethnic Group

It appears that there is almost a perfect inverse relationship between affordability and income when accounting for ethnic group.

EDA Key Findings

The Exploratory Data Analysis presented in this section was useful to begin understanding basic structures, patterns, and relationships within this dataset. The following is a brief summary of the most important EDA insights:

- There is a strong positive correlation between `cost_yr` and `ave_fam_size` which makes sense, larger families tend to require a greater amount spent on groceries.
- This data contains a wide range of annual income ($2,500 - $250,000), but is highly skewed.
- There appears to be an inverse relationship between `affordability_ratio` and `median_income` when grouping by ethnic group.

These key findings from the EDA uphold that there is a complex relationship between income, family size, and food affordability among women-headed households in California. Moving forward, these insights will act as a foundation for finding a predictive model that best estimates affordability ratios based on economic and demographic factors.

Methods

Feature Engineering

Effective feature engineering for this dataset includes transforming the raw data obtained into meaningful, usable representations that can be fed into a model for accurate predictions. This section will identify the steps taken to engineer features effectively - specifically the process of enhancing relationships and providing more location information.

1. Add Latitude and Longitude

The original dataset used for this report did not include coordinate information, but it can be helpful to include these features to visualize the regions this exploration focuses on. To gather this information, I imported a csv file from GitHub that contains all of the major counties, zipcodes, and coordinates in California. Using a list of counties in the original data, I mapped those names to the counties in the csv file and merged their corresponding latitude and longitude values.

2. Food Cost Index

The Food Cost Index is a measurement from the Consumer Price Index (CPI) used to understand the yearly costs of food in the United States. To assess where the women in this dataset stand in regards to the national average, the `food_cost_index` variable was calculated by dividing `cost_yr` by the national average cost of food. This value was calculated by averaging the median cost of food for high, medium, and low income households. (national average source)

	latitude	longitude	food_cost_index
count	291717	291717	264366
mean	35.218	-116.352	72.955
standard deviation	2.655	8.745	13.771
minimum	26.13	-123.73	29.704
Q2	33.97	-119.67	63.978
median	34.06	-118.26	71.594
Q4	36.75	-117.32	79.893
maximum	48.92	-71.35	161.904

Table 4: Additional Numeric Data Summary Statistics

Preprocessing

4. Handling Categorical Features

All categorical features must be encoded into numeric formats that the computer can process when using modelling techniques. To do so, the `OneHotEncoder` package from the `sklearn` module is imported and fit to the predictors after they have been split into training and testing sets. This encoder works by creating binary columns for every unique category in all the categorical features contained in the data.

5. Handling Numeric Features

Similarly to the categorical features, numeric features also need some preprocessing before getting passed into relevant models. Since this analysis deals with features that are likely very skewed, the predictors need to be scaled using the `StandardScaler` package also from `sklearn`.

6. Handling Missing Values

Referring back to Tables 2.1, 2.2, and 3, there are a handful of missing values that need to be handled before running any models. Using Scikit-Learn's `SimpleImputer` package, the missing numeric values can be filled with the average value of the given feature, and the most frequent value can be filled in categorical features.

Supervised Learning Models

Several supervised learning models were tested on this data to test the relationships between features and the target variable. Each model was assessed based on the performance metric root mean squared error (rMSE),

computational time, and challenges that occurred with the process. The table below includes the summary on these models and metrics.

Model Name	Description	Hyperparameters	rMSE	Time	Challenges
K Nearest Neighbors Regressor	Predicting target by finding the average of the *k* number of neighbors surrounding a data point.	`n_neighbors`: 5 `weights`: 'distance'	0.0321	12m	Computationally expensive
Ridge Regression	A linear regression model that uses Ridge regularization - a penalty term added to the cost function that pushes all coefficients towards zero.	`alpha`: 10	0.2015	26.1s	Poor rMSE score compared to other models
Decision Trees	Starting from the root node, branches move down with nodes split based on different patterns in the most important features until we reach the terminal node (leaves).	`max_depth`: 7 `min_samples_leaf`: 1 `min_samples_split`: 10	0.0158	2m32s	Prone to overfitting
XGBoost	Combines weak learning trees into strong learners by combining residuals and pruning.	`learning_rate`: 0.2 `max_depth`: 7 `n_estimators`: 100 `subsample`: 1.0	0.0153	2m20s	Best available model
Deep Neural Network	Using Tensorflow, a DNN is an artificial neural network that includes multiple layers that can look for different patterns in the data.	`epochs`: 20 `batch_size`: 32 `validation_split`: 0.2	0.0628	1m57s	Not a very strong model for this data

Table 5: Supervised Learning Models

Discussion on Model Selection

Notable Patterns Across Models

1. Tree-Based Models Outperform Linear Models

As recorded in Table 5, the Decision Tree and XGBoost models outperformed Ridge Regression - indicating that there likely is not a strong linear relationship between the features.

2. Single Model Vs. Ensemble Method

While both the Decision Trees and XGBoost performed well, XGBoost's built-in regularization made it robust to overfitting, and more accurate. The DNN was flexible and handled the patterns in the data well, but it failed to outperform XGBoost, indicating that it may not be complex enough. While more layers and complexity could be added to this model, it is currently more computationally expensive than XGBoost, and adding more dimensions would only add to this time disparity.

3. Computational Costs

K Nearest Neighbors (KNN) yieled a high computational cost because it relies heavily on time-consuming distance calculations for each prediction. XGBoost was not the fastest model, but offered a good trade-off between runtime and predictive power.

Detailed Discussion on Best Model

After weighing all of the benefits and disadvantages of each of the models explored, the XGBoost model has proven to be the *best* based on its rMSE score, total time, and inherent ability to generalize well to new data.

XGBoost Hyperparameter Tuning

To build the most optimized version of the XGBoost Regressor, hyperparameter tuning was conducted using the grid search approach. The goal of this method was to provide many possible combinations of hyperparameters and identify the ones that minimize the model's prediction error on the training data, while also balancing the model's ability to generalize to unseen data. The following hyperparameters were tested:

- `n_estimators`: Controls the number of trees in the model
- `learning_rate`: Controls the step size for updating model weights
- `max_depth`: Specify the maximum depth of a tree
- `subsample`: Fraction of observations randomly sampled for each tree

With these hyperparameter options, `GridSearchCV` tested all combinations and evaluated them using 5-fold cross validation, which splits the data into 4 training subsets and one validation. This tuning process was effective in ensuring that the model was fit properly, and contained hyperparameters that control the model's over/under-fitting tendencies.

SHAP for Feature Importance

Shapley Additive Explanation (SHAP) values are a strong tool that can help interpret the information generated from the XGBoost Regressor's predictions. They act as a general framework that can be appplied to models, and explain their outputs by uniformly quantifying feature contributions.

I. Model Performance Evaluation

One thing SHAP can be used for is to see how predicted values compare to actual values to evaluate model performance. From these comparisons, we can gather examples of true positives/negatives and false positives/negatives from the model's output.

1. True Positives/Negatives (TP/TN):

   A predicted affordability ratio of 0.3246 for an individual compared to the actual ratio of 0.3247 has an error of essentially zero (1.034459e-04) which is insignificant at the chosen 0.01 level, thus suggesting that the model is accurate in this instance.

2. False Positives/Negatives (FP/FN):

   A predicted affordability ratio of 1.3403 compared to the actual value of 1.4491 has an error of ~0.11 which is significant at the chosen 0.01 level. Since the model incorrectly classified this individual higher than their actual affordability ratio, this is a **False Positive** overprediction. An opposite instance with negative error would represent a **False Negative** underprediction.

II. Feature Contributions

To continue interpreting the XGBoost model predictions, SHAP can be used to quanitfy and visualize feature contributions. This can be done on a global scale by identifying the most influential features across the dataset, and locally to visualize individual predictions.

A SHAP summary plot helps to identify features by ranking them based on their average absolute SHAP values.

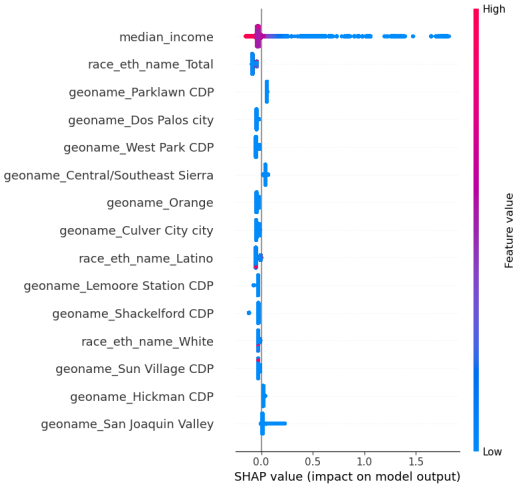

Figure 4: SHAP Global Feature Contributions

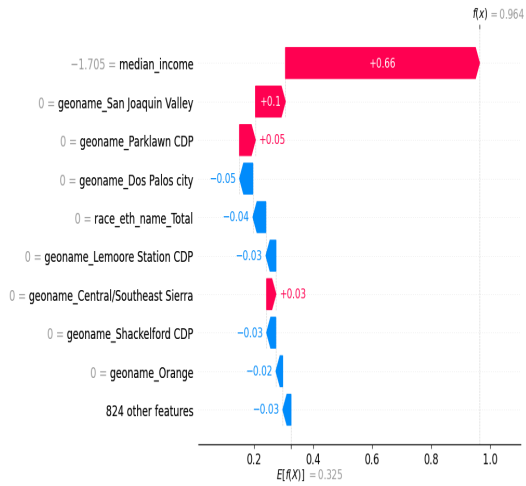

Figure 5: SHAP Local Feature Contributions

As seen in Figure 5, the feature `median_income` is a substantially influential variable in the model. When `median_income` is high, overall predictions for `affordability_ratio` decrease slightly, and when `median_income` is low, `affordability_ratio` increases substantially. Similar relationships can be interpreted for the other features on the SHAP global feature contributions plot.

To see the feature impacts on a specific prediction, the waterfall plot in Figure 5 shows that `median_income` contributes +0.66 to `affordability_ratio`, pushing the prediction higher than the baseline expected prediction of 0.325. `geoname_San Joaquin Valley` contributes +0.1, also pulling the prediction for affordability up. These features in combination with the others captured in Figure 5 contribute to pushing the final prediction for this particular individual to a predicted affordability ratio of 0.964.

SHAP is a powerful tool used to interpret the XGBoost predictions by qunatifying the contributions of each feature to the model's output. Key global features that are particularly influential were identified by SHAP's global analysis, and local analyses from the waterfall plot provide insight into individual predictions. These

interpretations allow models to be applied to real-world implications by ensuring that the predictions are easily explainable.

Anomaly Detection

The next step in understanding the data and chosen model is to find observations or patterns that deviate significantly from the norm. This may include underpredictions, overpredictions, or unexpected feature combinations.

I. Anomaly Detection on Affordability

To detect affordability anomalies, the dataset can be clustered into 3 groups based on affordability. This clustering techniques helps with uncovering general patterns in the distribution of the target, while outlining extreme cases to be aware of.

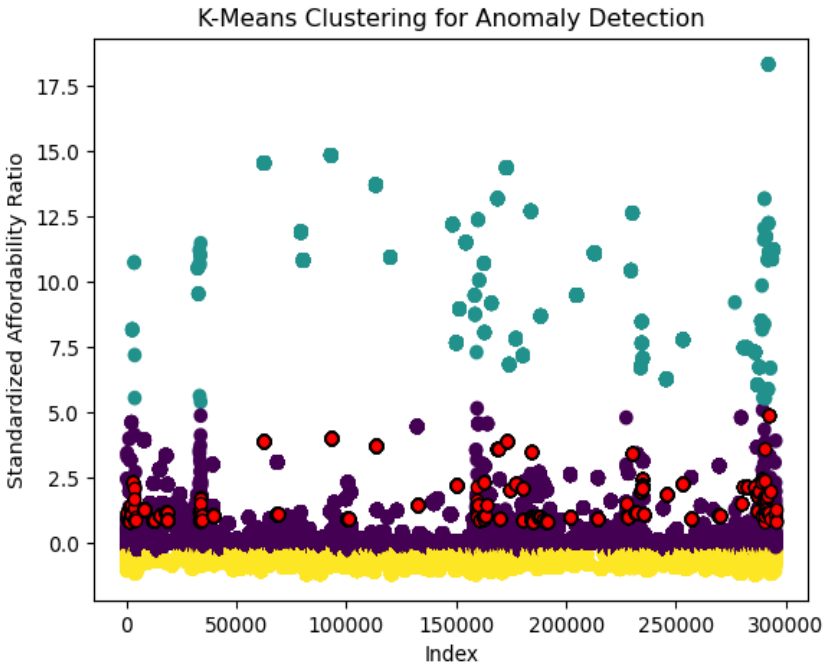

Figure 6

In Figure 6, anomalies are highlighted in red. These were calculated by selecting the top 1% of affordability ratios farthest from their respective cluster centroids. Such large distances indicate these individuals likely have unusually high ratios given the surrounding averages.

II. Anomaly Detection on Affordability, Family Size, and Ethnic Group

Anomaly detection can also be expanded to tracking multiple features. In this case, looking at family size and ethnicity can uncover important affordability subgroups.

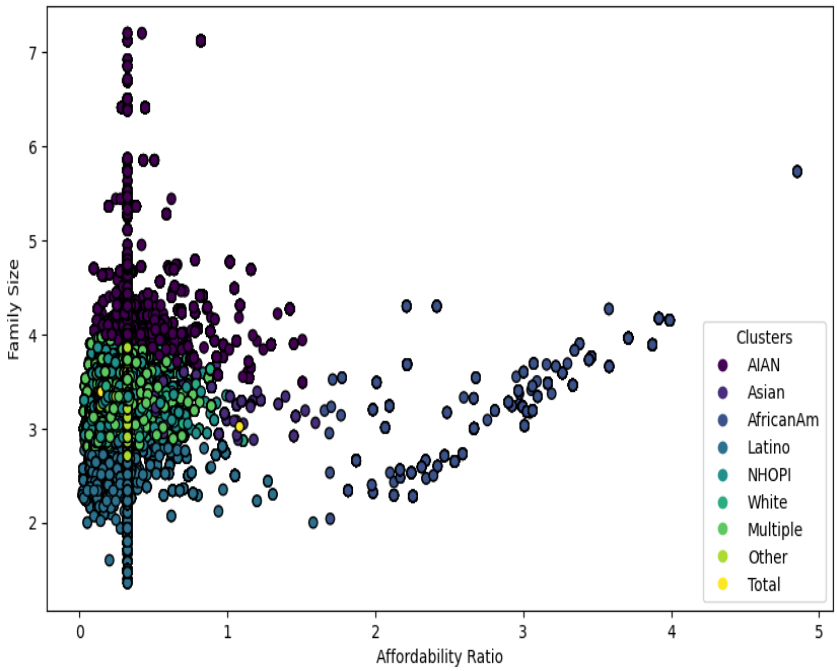

Figure 6

In Figure 6, the cluster groupings are within close proximity of each other, but there is a clear visual difference from one ethnic group to the next. The anomalies in this figure are any individuals that appear far from their cluster centroids. examples of these could be high or low earners clustered in a prodominently low or high affordability or family size group.

There are many iterations of this type of subgroup clustering that would provide valuable insights regarding various factors. Providing this type of analysis can be used to inform certain political interventions to help subgroups that would benefit from government action.

Dimension Reduction

The topic used to understand this dataset is dimension reduction. This step is included to simplify the complexity of the data, making it easier to analyze. Since some of the features in this dataset were One-Hot Encoded, it has high dimensionality from the numerous binary features. This section will focus on applying Principle COmponent ANalysis (PCA) and UMAP to transform the data.

I. PCA

The first step is to fit PCA to the training predictors. SInce the goal is to find a linear transformation of the original data that mximizes the variance of the data, the graph below helps to visualize what this process looks like for our case.

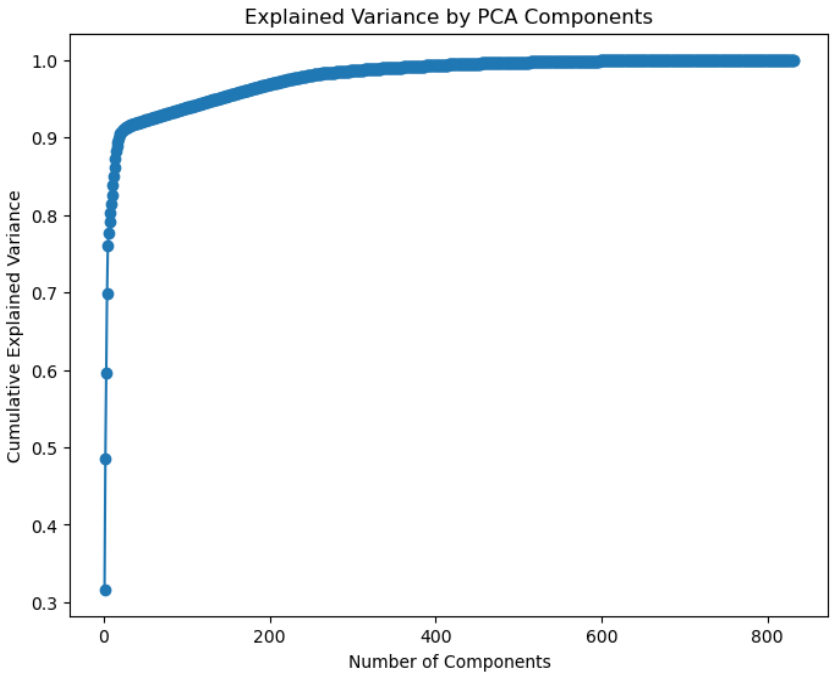

Figure 8: PCA Maximize Variance Plot

From this, we know that to retain 95% of the variance, we need 138 components. The next step is to find the hyperplane that preserves the largest amount of the variance, and project the data onto that hyperplane.

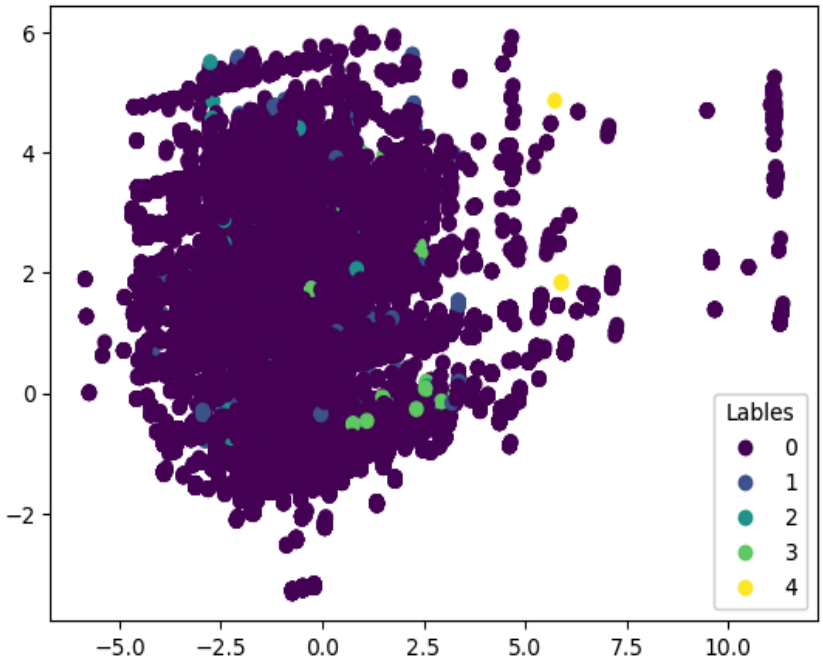

Figure 8: PCA Labels

The new `X_train_pc` and `X_test_pc` could be used in the XGBoost model for improved performance. Fitting the optimized model to these new training and testing sets yieled a rMSE of 0.0202 which is higher than the original model's result (0.0153). This could be due to the fact that the dimenion reduction values are not complex enough to explain the behavior in the data well.

II. UMAP

UMAP is an example of a manifold learning method to see how a non-linear dimension reduction algorithm would perform on this data. There are multiple manifold learning options, but UMAP is generally faster than tSNE, another popular choice, and balances global versus local relationships better than tSNE as well.

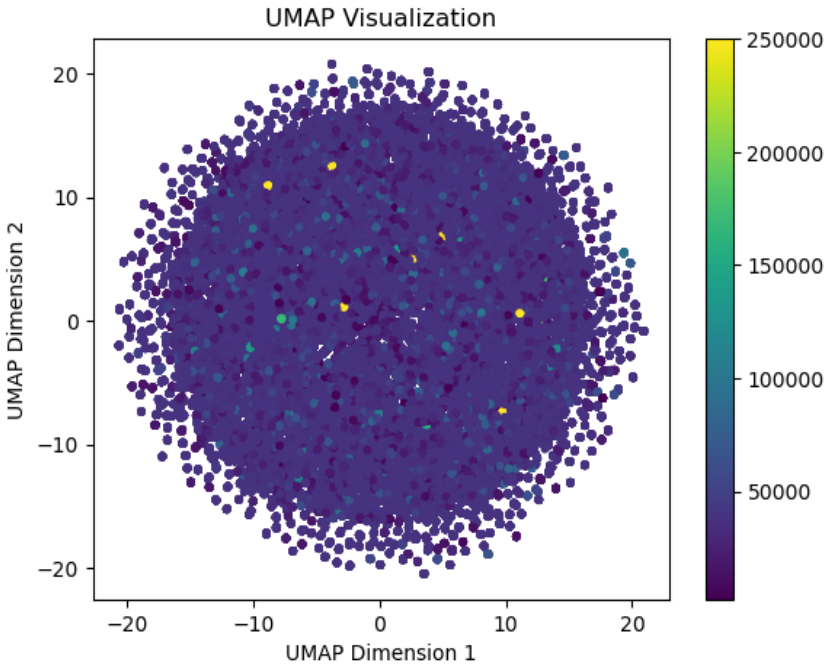

Figure 9: UMAP Labels

Since UMAP is being used for visualization purposes in this report, a 2D representation of the dat is not extremely informative or useful. Additionally, UMAP took significantly longer than PCA which adds to the disadvantage of using it as the key dimension reduction technique for this analysis.

Conclusion and Next Steps

This report used machine learning techniques to analyze and predict the nature of affordality ratios among women-headed households in California based on income and additional demographic information. After contrasting multiple different kinds of models, XGBoost proved to be the most effective as it captured the complexity of the data and provided reliable predictions. An analysis of feature importance with SHAP reinforced the strong impact of income and geographic location on affordability. Anomaly detection helped highlight outliers and unique cases of individuals who are likely higher earners or spend less on food than the majority of the women sampled.

In the future, there could be stronger integration of geographic information that could be used for a more thorough spatial analysis of affordability considering it was such an influential predictor. It may also be useful to test additional ensemble approaches, and refine the anomaly detection process to find more interesting patterns and subgroup interactions.

include some suggestions