# Analysis of Poverty in the USA

Jesse Wolfe, January 2018

## Executive Summary

This document presents an analysis of poverty rates from different counties throughout the USA. The data set used consists of 3198 samples, each representing a county. Every sample has 33 features (columns) that can be examined when analyzing its poverty rate. These features include many different economic, demographic, and health related statistics. After exploring the data, a boosted decision tree model was made to predict the poverty rate of a county, given its features.

From the analysis, the author finds the following conclusions:

There are many factors that contribute to poverty in the United States. However, the author found the following features to be the most indicative of a county's poverty rate.

- *econ__pct_civilian_labor*
- *econ__pct_unemployment*
- *econ__pct_uninsured_adults*
- *demo__pct_adults_less_than_a_high_school_diploma*
- *demo__pct_adults_bachelors_or_higher*

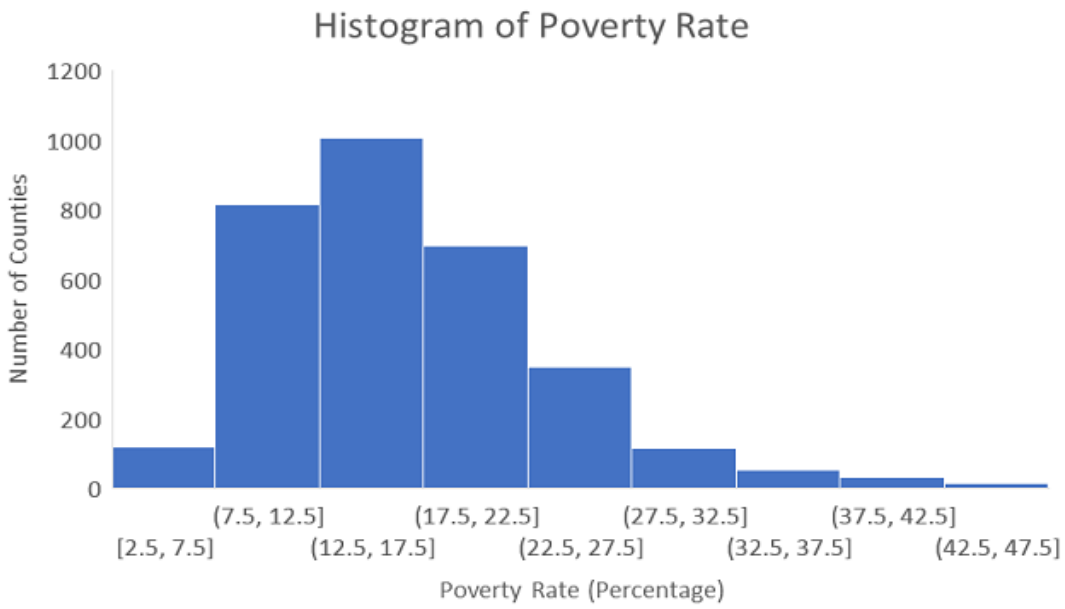
(Features above represent yearly percentages of a county's population)

## Initial Data Exploration

The analysis began by evaluating summary statistics for each feature in the data set. The **poverty_rate** feature is shown first, as it is the feature that will be predicted.

| *poverty_rate* | |
|---|---|
| | |
| Mean | 16.81713571 |
| Standard Error | 0.118441495 |
| Median | 15.8 |
| Mode | 14.3 |
| Standard Deviation | 6.697968659 |
| Sample Variance | 44.86278416 |
| Kurtosis | 1.666646236 |
| Skewness | 1.048357214 |
| Range | 44.9 |
| Minimum | 2.5 |
| Maximum | 47.4 |
| Sum | 53781.2 |
| Count | 3198 |

On the left you can see the summary statistics for the poverty rate of all counties involved in the data. There are counties with as little as 2.5% of the population in poverty; whereas others have as much as 47.4%. A Standard deviation of 6.7% and a mean of almost 17%.

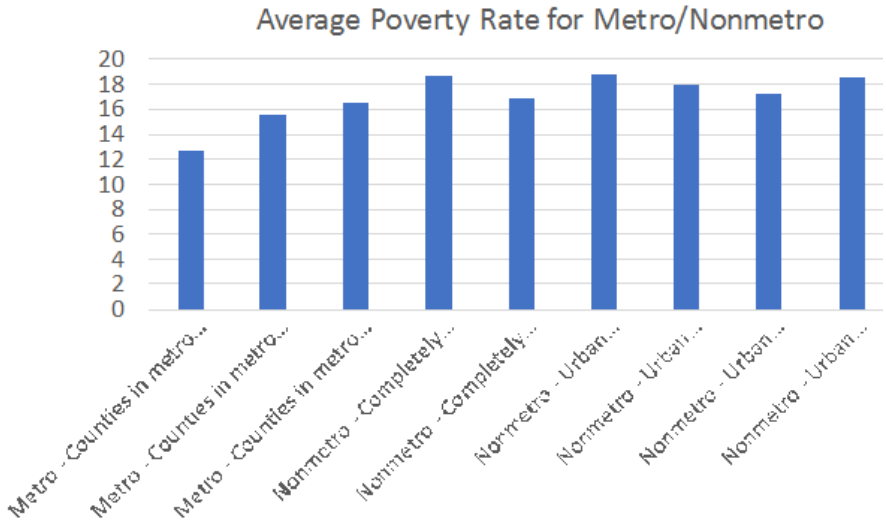



A histogram of poverty rate shows that the distribution is right-skewed, meaning most counties have poverty rates that are on the lower end of the spectrum.

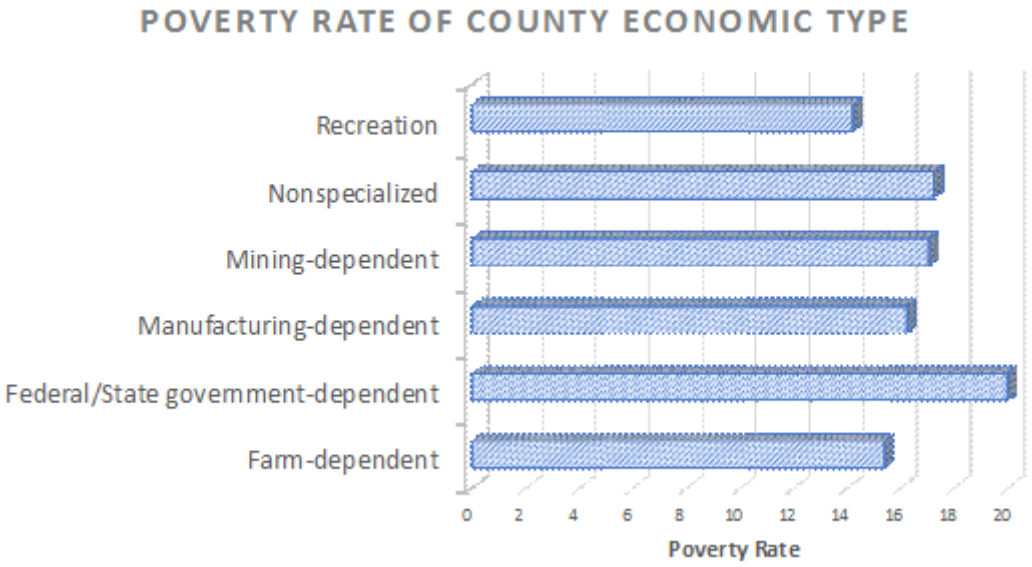
| *health__homicides_per_100k* | | *health__motor_vehicle_crash_deaths_per_100k* | |
|---|---|---|---|
| | | | |
| Mean | 5.95074736 | Mean | 21.11607695 |
| Standard Error | 0.144314845 | Standard Error | 0.199430112 |
| Median | 4.84 | Median | 19.63 |
| Mode | 3.3 | Mode | 21.84 |
| Standard Deviation | 5.063374298 | Standard Deviation | 10.51698434 |
| Sample Variance | 25.63775928 | Sample Variance | 110.6069597 |
| Kurtosis | 13.77948185 | Kurtosis | 4.054096859 |
| Skewness | 2.736955972 | Skewness | 1.302389088 |
| Range | 51.88 | Range | 107.36 |
| Minimum | -0.39 | Minimum | 3.09 |
| Maximum | 51.49 | Maximum | 110.45 |
| Sum | 7325.37 | Sum | 58723.81 |
| Count | 1231 | Count | 2781 |

It was also noted that features like *health__homicides_per_100k* and *health__motor_vehicle_crash_deaths_per_100k* were missing many values. To remedy this missing data, we will fill null values; which will be discussed later. Important categorical variables in the data set include:

- *area__rucc* — Rural-Urban Continuum Codes – Categorizes a county by its size and proximity to metros
- *area__urban_influence* — Urban Influence Codes – Categorizes a county by its population size and relation to cities. Similar to *area__rucc*
- *econ__economic_typology* – Classifies a county by its dependent economic type
- *yr* – Values include “a” and “b”. The data has been accumulated over 2 years.



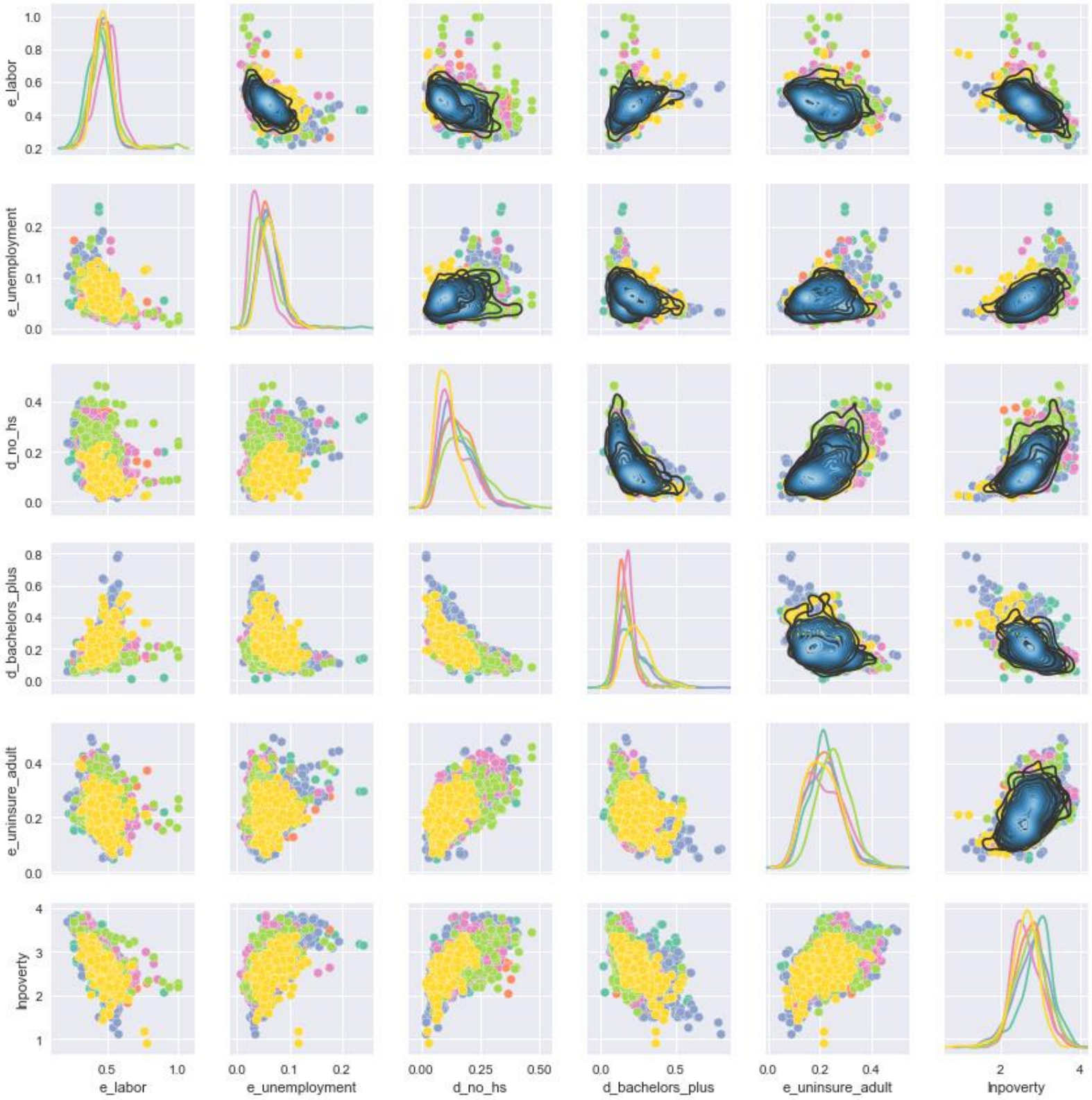

It can be seen in the above bar graph that all Nonmetro counties have higher poverty rates than Metro counties.



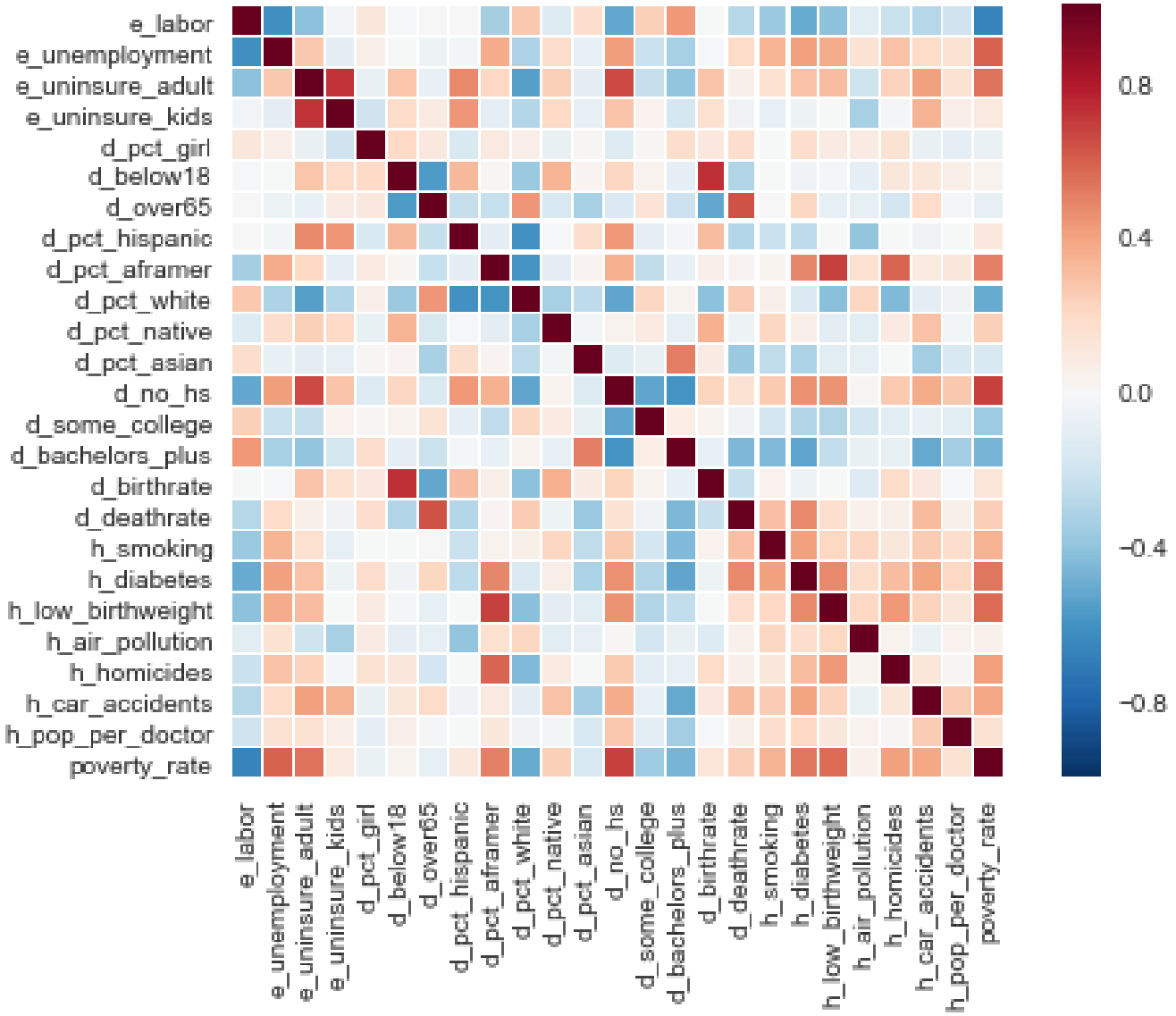

Additionally, we can see that counties that have Federal/State government dependent economies are by far the most impoverished.

## Correlation and Apparent Relationships

After analyzing individual features, and the different facets of the data set, we attempted to find any relationships, specifically between those features and **poverty rate**. In using a bit of Python, and the Seaborn graphing module, key relationships were discovered between those features summarized in the Executive Summary, and poverty rate. Visualizations of these relationships can be seen on the next two pages.




The five features the author found to be most significant are plotted against each other on the graph above. Their names have been shortened for readability. On the bottom row- the features are plotted against the log of *poverty_rate*, named *lnpoverty*. Notice that there are linear relationships between the columns selected and the poverty rate. This visualization is also multi-faceted in that it is colorized by the categorical variable, economic typology. It can also be noted that there are outliers in counties with Recreation and Federal/State government dependent economies.




Above is a correlation matrix between all features used in the decision tree model. The colors represent the strength of the correlation coefficient between features. The coefficient can be any value between -1 and 1, and it tells us how close the compared features are to a perfect linear relationship. White represents no correlation, red is a strong positive correlation, and blue a strong negative correlation. Again, notice the bottom row, all columns are plotted against our target column *poverty_rate*. Columns *e_labor* and *d_bachelors_plus* have a strong negative linear relationship with *poverty_rate*. Whereas *e_unemployment*, *e_uninsure_adult*, and *d_no_hs* have a strong positive linear relationship with *poverty_rate*.

Correlation Coefficients:

- *econ__pct_civilian_labor* **- 0.670416908**
- *econ__pct_unemployment* **0.592021544**
- *econ__pct_uninsured_adults* **0.541711966**
- *demo__pct_adults_less_than_a_high_school_diploma* **0.680360284**
- *demo__pct_adults_bachelors_or_higher* **- 0.467133735**

# Predicting Poverty

After getting a sense of the data through analysis and visualization, a **Boosted Decision Tree** model was created to predict the *poverty_rate* of other counties data. Before this could be done, it was necessary to clean the data set. Missing values were replaced with the mode of the respective column. The mode of the column was chosen because it proved the most rewarding for our model. It was then necessary to normalize the data by taking the LogNormal of all numerical features (except categorical). This would ensure a more "normalized" scale of the data and more accurate predictions.

The model was trained on 85% of the data and tested on the remaining 15%. A series of nested for loops was used to tune the Boosted Decision Tree's parameters; a portion of the code can be seen here:

```
for est in range(50, 100, 5):
    for samp in range(30, 50, 2):
        for depth in range(25, 35, 1):
            regr = GradientBoostingRegressor(learning_rate=0.109, n_estimators=est,
                                             min_samples_leaf=samp,max_depth=depth)
            regr.fit(x_train, y_train.values.ravel())
            y_2 = regr.predict(x_test)
            rmse = metrics.mean_squared_error(y_test, y_2)
            if not lowest_found:
```

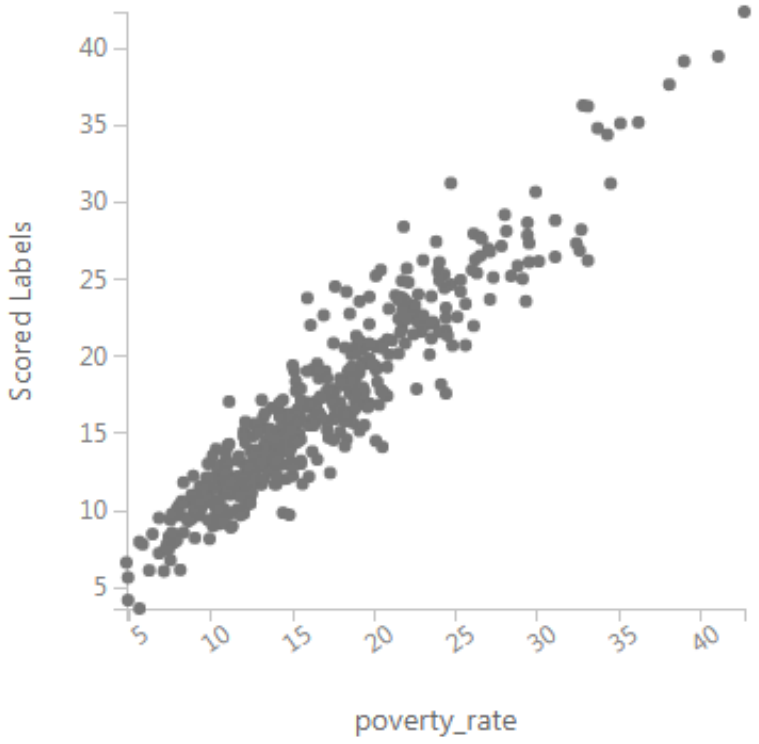
Python was also used in choosing which features would be used in the model:

```
def optimize_feature_space(potential_dropped_features):
    last_rmse = 0
    lowest_rmse = 0
    lowest_found = False
    dropped_features_that_improved_model = []
    for combo in range(0, len(potential_dropped_features)+1):
        for subset in itertools.combinations(potential_dropped_features, combo):
            x_train, x_test, y_train, y_test = train_test_split(training_set2.drop(list(subset), axis=1),
                                                                train_labels2, train_size=0.85,
                                                                test_size=0.15, random_state=7)

            regr = GradientBoostingRegressor(n_estimators=754, learning_rate=0.109,
                                             min_samples_leaf=35, max_depth=28)
            regr.fit(x_train, y_train.values.ravel())
            y_2 = regr.predict(x_test)
            rmse = metrics.mean_squared_error(y_test, y_2)
```

Interestingly, 5 features were found to be a hinderance to our model and were thus removed. They were, *health__pct_adult_obesity, health__pop_per_dentist, health__pct_excessive_drinking, health__pct_physical_inacticity, and demo__pct_adults_with_high_school_diploma.* Removing these features increased the model's predictive performance.

After the model's parameters were tuned and features selected- it was run, and that 15% of our data left over for testing was tested. A graph of our model's predictions (Scored Labels) vs the actual poverty rates can be seen here:




If the model's predictions were perfect, the scatter plot above would be a perfectly straight line, having a slope of 1 (and we would have most likely "overfit" our data). This is not the case, but you can see that the model's predictions are generally accurate. The Boosted Decision Tree model achieved a Root Mean Squared Error (RMSE) of 2.165522. This means that on average, our predictions for a county's poverty rate are off by only 2.17%.

## Conclusion

The analysis of poverty rates in the USA has shown that those poverty rates can be accurately predicted, given enough data. The more data we can compile on these counties, the more accurately we will be able to predict poverty rates in the future. Specifically, things such as civilian labor percentages, unemployment rates, education, and uninsured adults have a significant influence on poverty rates.