# Predicting US Distress Scores*

Ariana Schindler
*DATA1030*
*Data Science Initiative*
*Brown University, Providence, RI, 02912*
*https://github.com/arianaschindler/data1030_final*

(Dated: December 9, 2022)

## I. INTRODUCTION

### A. BACKGROUND

This data comes from the Economic Innovation Group [1] and was sourced from the US Census Bureau over the years 2015-2019 and examines economic well-being at the zip code level in order to provide a structured record of the divided landscape of American prosperity. The DCI's collection of geographic, economic, and demographic features offers a comparative view of the spatial distribution of U.S. economic well-being and is a tool through which we can evaluate for where and for whom America's promise of opportunity applies.

### B. DATA OVERVIEW

The target variable of this dataset is Distress Score, which is a continuous feature. The `Distress Score` target variable has values which range from $0 - 100$, with a score of 0 indicating prosperity and a score of 100 indicating distress. The dataset is composed of 27 features and 25766 observations. The categorical features all consider geographic identifiers of the observations and the remaining features are made up of continuous economic and demographic observations.

One of the most intriguing publications I read which analyzed the DCI dataset was published in the Annals of Surgery journal [2] and combined the DCI dataset with data on risk outcomes after surgery. They hypothesized that the DCI could predict a risk-adjusted outcome after surgery and after their research, they found with a p-value of $\alpha = 0.005$ that the DCI could successfully predict these risk scores.

Another interesting study which used the DCI dataset was published in ScienceDirect and addressed socio-economic status and COVID-19-related cases and fatalities [3]. The team's goal was to quantify the relationship between COVID-19 cases and fatalities and the socio-economic status of the person afflicted. Their results found that the socio-economic features most strongly correlated with COVID-19 cases and fatalities are `Percent of Adults w/o a High School Degree`, and `Black or African-American Percent of Population`.

## II. EXPLORATORY DATA ANALYSIS

### A. INITIAL LOOK AT DATA

I began looking at the data by checking descriptive statistics. I noticed that the quartiles for the target variable were evenly distributed, with a minimum of 0, a maximum of 100, and 25% = 25.0, 50% = 50.0, and 75% = 75.0. This made me suspicious that `Distress Score` might have a uniform distribution, so my next step was to create a histogram of the target variable to check the distribution.

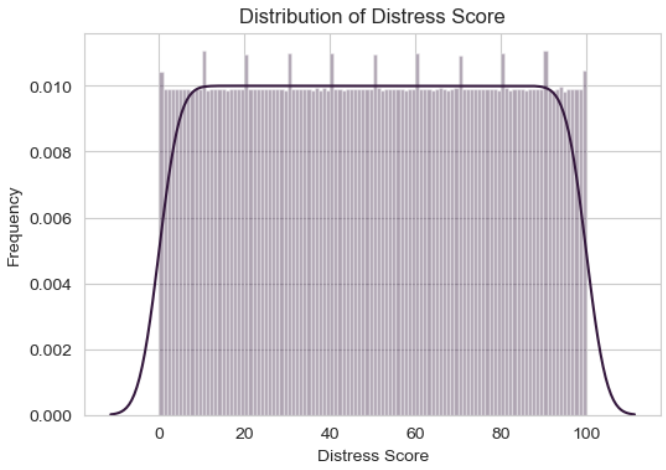


FIG. 1: Distribution of Distress Score

### B. DISTRIBUTIONS AND CORRELATIONS

After finding that the target variable was indeed uniformly distributed, I wanted to visualize the target variable against other features in the dataset. After creating a correlation heatmap, I found that the three strongest feature correlations were `Poverty Rate`, `Median Income Ratio`, and `Percent of Adults w/o a High School Degree`.

* ariana_schindler@brown.edu

The first relationship I explored further was the moderately strong positive correlation between `Distress Score` and `Poverty Rate` which had a correlation coefficient of 0.745. I visualized this with a scatterplot with the target variable on the x-axis and `Poverty Rate` on the y-axis.

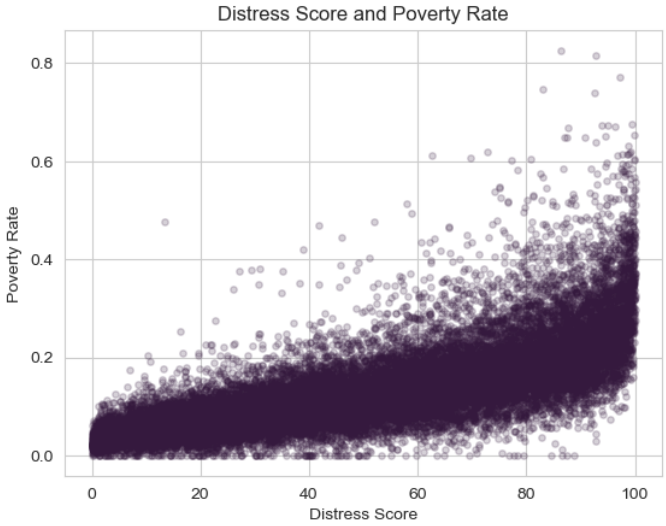


FIG. 2: Distress Score and Poverty Rate

Next I visualized the relationship between `Distress Score` and `Median Income Ratio` with a line graph. To this graph, I also added a confidence interval outside of the line to visualize the variation in the data. In this figure, we can see there is a negative correlation, though this line is not linear and suggests that we should explore nonlinear regression models after preprocessing.

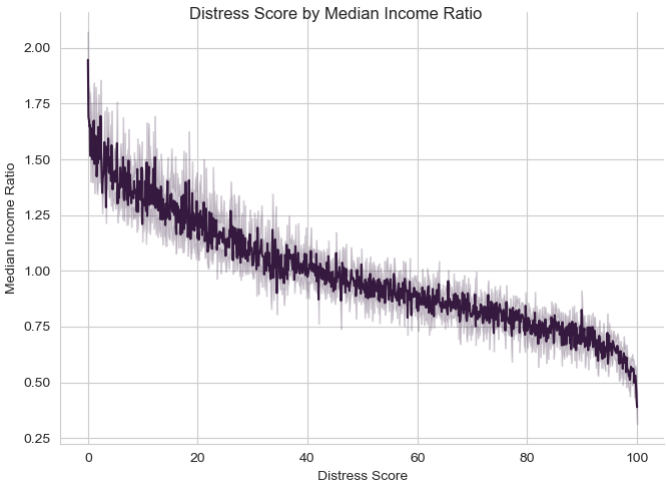


FIG. 3: Distress Score and Median Income Ratio

The final relationship I wanted to visualize in greater detail than the correlation heatmap was between `Distress Score` and `Percent of Adults w/o a High School Degree`. For this visualization, I transformed the target variable into a categorical one by evenly splitting the uniformly distributed observations into quintiles, with 1 indicating prosperity and 5 indicating distress. After transforming the target, I visualized the correlation with a stacked bar graph with `Percent of Adults w/o a High School Degree` on the x-axis, a count on the y-axis, and the quintiles as different hues of stacked bars.

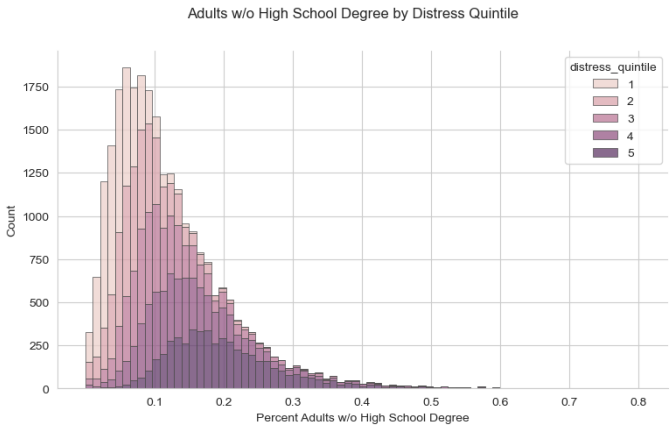


FIG. 4: Distress Score and Percent of Adults w/o High School Degree or equivalent

## III. METHODS

### A. SPLITTING

This data is iid, does not have group structure, and is not dependent on time. Because of these reasons, I performed a basic split on the dataset. To split the data, the first thing I did was remove the target variable from the rest of the columns and identify $X$ and $y$. Then I performed a basic split with 60% train, 20% validation, and 20% test. This method best mimics future use because the target variable is iid and uniformly distributed so we do not need to adjust for skewed data.

### B. PREPROCESSING

The data did not contain group structure and none of the categorical variables were ordered, so the preprocessing step was fairly straight-forward. Since many of the geographic features were simply larger-scale extensions of each other, I chose to keep only 3 categorical features: `States`, `Census Regions`, and `land_designations`, which identifies the location of the zip code as 'suburban', 'rural', 'small town', or 'urban'. For these categorical variables, I used one hot encoding. The remaining features were continuous so to preprocess these, I transformed them using the standard scaler. After preprocessing, we have 75 features in the dataset with

15459 observations in the training set and 5154 observations in the test set.

### C. ML PIPELINE

For the ML Pipeline, I developed an algorithm which would loop over 5 random states and then split and preprocess according to our predetermined strategies. After fitting and transforming the training data and transforming the validation and testing sets, I calculated the baseline root mean squared error for each model. I chose to evaluate the model performances with the RMSE so that I could receive the average test scores for each model in the same unit as our target variable. Since our target is conveniently a score between $1 - 100$, this makes it straightforward to compare our models' average test score to the baseline.

After calculating the baselines, a for-loop inserts the model-specific parameters into the models using ParameterGrid. The model is then fit with the training data and predictions for each models for each random state are then produced using the test set. The RMSE is then calculated from the original testing target values and the predicted testing target values. All of these scores are stored and returned.

#### 1. *MODELS AND PARAMETERS*

Since our target is numeric, I selected regression models for the pipeline. The linear models that I included in my analysis were Lasso, Ridge, and ElasticNet, all of which I tuned the parameter *alpha* for. Along with *alpha*, for the ElasticNet model, I also tuned the *l1_ratio*. In my EDA, I noticed that the target variable appeared to possibly have nonlinear relationships with other features, so I also included nonlinear models in my pipeline. For the RandomForestRegressor, I turned the *max_depth* and *max_features* parameters, for the KNeighborRegressor model I tuned the *n_neighbors* parameter, and for the SVR model I tuned the parameters *gamma* and *C*.

## IV. RESULTS

### A. MODEL PERFORMANCE

Across the random states, the variability of the different models was apparent in the standard deviation in their test scores. As suspected, the nonlinear models had smaller standard deviations while the linear models had larger standard deviations over the random states. The Random Forest model had an extremely low standard deviation of 0.056, while the next smallest was the KNN model with a standard deviation of 0.104. The Lasso model had the largest standard deviation of 0.455.

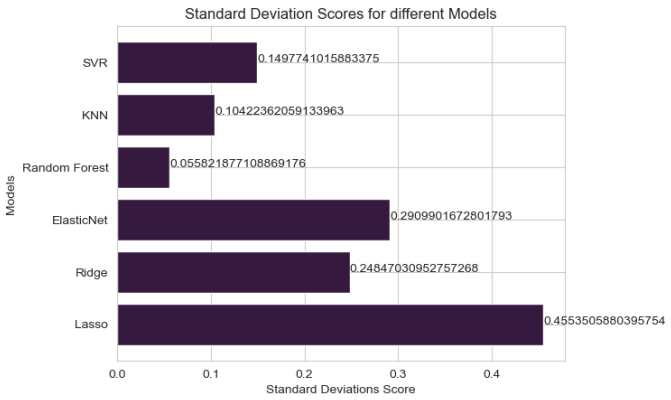


FIG. 5: Standard Deviations of Model Test Scores

To check out how a sample of the model's performance, I took 100 random samples from the testing data and their respective predictions and visualized them on the same plot with an error bar between the true and predicted value of the difference between the two. Analyzing larger and larger samples up to the full dataset size, I found that the smallest amount of statistically significant outliers in the errors between the true and predicted values is around 3%.

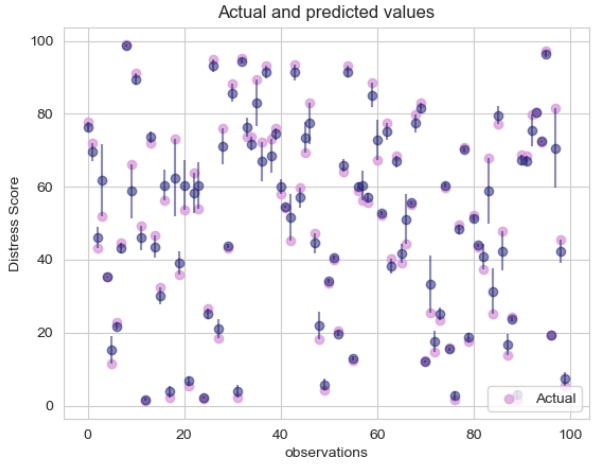


FIG. 6: Sample of Error Between True and Predicted

#### 1. *BASELINE ANALYSIS*

The baseline root mean squared error of our dataset was 28.969 and every model analyzed in the ML pipeline had a mean test score smaller than the baseline. Just as we saw with the standard deviations, the mean test scores of the nonlinear models indicated that the nonlinear models were better predictors of the target variable than the linear models. The Lasso model had the largest mean test score, with an RMSE of 11.104. Moving on to the nonlinear models, the Random Forest Regressor had the

lowest RMSE with a score of 3.139. Using the best Random Forest Regressor model to predict `Distress Score`, we would likely be within about 3 units of predicting the correct distress score.

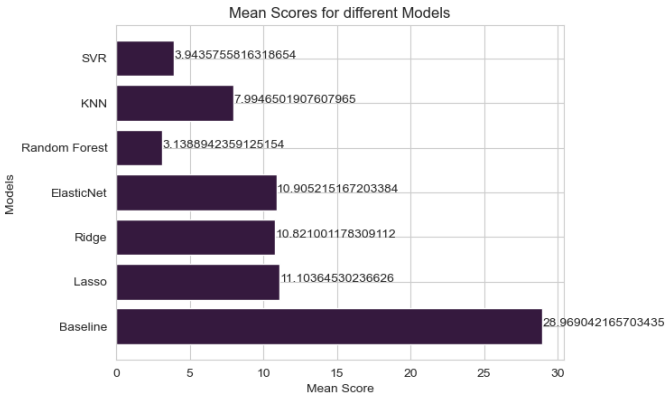


FIG. 7: Means of Model Test Scores

To further the baseline analysis, I calculated how many standard deviations away from the baseline each model's average test score was. This resulted in more evidence to support what we have already discovered: that the linear models had a smaller standard deviation between their scores and the baseline, and the nonlinear models had a larger standard deviation between their scores and the baseline, with the Random Forest model having the largest standard deviation of 0.892.

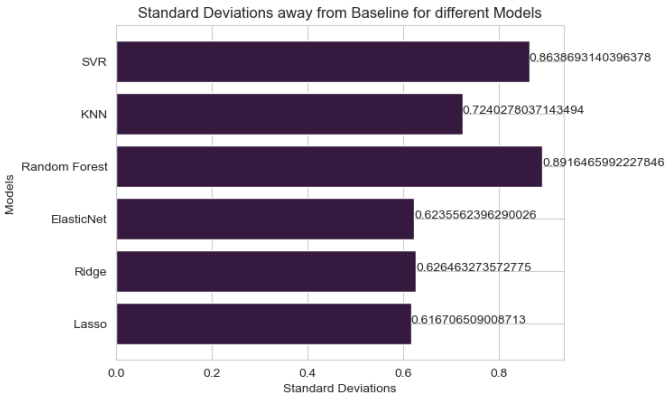


FIG. 8: Standard Deviations Away from Baseline

## B. FEATURE IMPORTANCE

To assess global feature importance, I used the built-in feature importance function for Random Forest Regressor, calculated SHAP values, and also used permutation. For all three methods, the two highest feature importances were `Poverty Rate` and `Median Income Ratio`. The third highest feature, however, differed among all three, with the built-in feature importance ranking `Adults without High School Degree` as 3rd most important, and with SHAP values ranking `Vacancy Rate` and Permutation ranking `Percent Change in Employment` in the 3rd place spot.

To further the feature importance analysis, I used the SHAP values to look at a few local feature importances. While the second-most important features (and so on) varied depending on where I looked locally, every local feature importance I checked prioritized `Poverty Rate` as the most important feature.

## C. IN CONTEXT OF THE DATA

As it pertains to our dataset, the feature importances tell us that the Poverty Rate of the area being examined is the most important factor in determining the distress score for that area. This makes sense, since the range of our target variable is $1 - 100$, with 1 being prosperous and 100 being distressed. The feature importance also revealed that the features such as location and race had an insignificant effect on distress score in comparison to economic data.

Further analysis showed that the top 5 features which appear in all feature importance methods include `Poverty Rate`, `Median Income Ratio`, `Adults without High School Degree`, `Vacancy Rate`, and `Percent Change in Employment`.

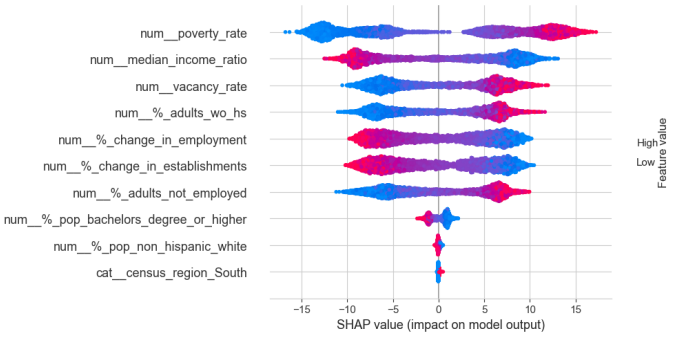


FIG. 9: SHAP Global Feature Importance

## V. OUTLOOK

The test score for the Random Forest Regressor was so small that this was not only my best model, but I also want to continue to tune the parameters to see if I can get the model even closer to predicting the true distress scores. Since the original distress scores were calculated using an undisclosed formula, I am fairly confident that we can tune the model to an even higher accuracy.

Another current model improvement could possibly be adding additional income data and local economic features since these seemed to have the greatest global and local feature importance.

The final technique I would like to use on this dataset would be to classify distress scores into quintiles and run classification models on the target variable as a categorical variable.

[1] Kesler, Patrick. "Distressed Communities - Economic Innovation Group." Economic Innovation Group, https://eig.org/distressed-communities/. Accessed 21 Oct. 2022.

[2] Mehaffey, J. Hunter, et al. "Socioeconomic 'Distressed Communities Index' Improves Surgical Risk-Adjustment." Annals of Surgery, no. 3, Ovid Technologies (Wolters Kluwer Health), Mar. 2020, pp. 470–74. Crossref, doi:10.1097/sla.0000000000002997.

[3] Hawkins, R. B., et al. "Socio-Economic Status and COVID-19–Related Cases and Fatalities." Public Health, Elsevier BV, Dec. 2020, pp. 129–34. Crossref, doi:10.1016/j.puhe.2020.09.016.