**Improved XGBoost Model**

Unsatisfied with our XGBoost model’s performance, we decided to fix its overfitting issue. Due to XGBoost’s abilities to be able to analyze large data sets efficiently, we felt that we still have to use XGBoost but with some major adjustments. We knew of two ways to fix the issue of overfitting: more aggressive hyperparameter tuning and a better cross-validation method.

**Hyperparameter Tuning & Cross-Validation**

We decided to tune more hyperparameters to find the most appropriate model for our data set. Initially, we only tuned n\_estimators, max\_depth, learning\_rate, and min\_child\_weight. To improve upon this, we tuned the same hyperparameters in addition to subsample, colsample\_bytree, colsample\_bylevel, and gamma.

We also improved on our cross-validation methods. Like before, we used the 5-fold cross-validation. This time, however, we made specific adjustments in this step to directly combat overfitting. We used XGBoost’s “early stopping” technique, a method used to stop the training process when the loss in the validation step starts to increase, by setting its parameter to 25 (around 10% of the training dataset).

**Coefficient Analysis**

After making these adjustments, we used XGBoost’s feature of displaying its most important coefficients to view the weight of every feature in the data set. Like before, we primarily used the weight of the coefficients for comparison.

This was our result:

Chart, bar chart

Description automatically generated

**1. Mobility’s Even Heavier Weight Than Before**

Surprisingly, our improved XGBoost model showed a significantly different weights of coefficients this time. All of the mobility features (putting the baseline differences aside) have shown to have more effect on the hospitalizations for COVID-19 than the influx within different areas marked by the zip code. On top of this, there is a much more consistent change in weight value as the feature importance value decreases.

**2.** **Zipcode Influx’s Decrease in Importance**

Most notably, the influx that different areas experienced shows to have a much smaller importance in this model than in the original XGBoost model. All of them are shown to have much lower importance than the mobility features except one: the area with the zip code 77843.

**Model Accuracy**

We used the same metrics as before to measure the model’s accuracy. Strictly from the training data set we divided for cross validation, we got these results:

Text

Description automatically generated

On the test data set, we achieved exceedingly better results compared to the original XGBoost model.

A screenshot of a computer

Description automatically generated with low confidence

**New Comparison of the Previous Best Model and the New Best Model**

Conclusion Table of Comparison

| **Model** | **Test RMSE** | **Test MSE** | **Test MAE** |
| --- | --- | --- | --- |
| Original XGBoost Model | 49.546824 | 2454.887798 | 37.391982 |
| Improved XGBoost Model | 8.669740 | 75.164398 | 3.271394 |

**Original XGBoost Model:**

1. Tuned 4 hyperparameters with 5-fold validation with no early stopping rounds implemented
2. Suffers from severe overfitting

**Improved XGBoost Model:**

1. Tuned 8 hyperparameters with 5-fold validation with early stopping rounds set to 25
2. Coefficients have more consistent values and shows visual pattern compared to original model
3. No overfitting, although the test results being better than the validation errors may indicate slight underfitting

We consider the new improved XGBoost Model to be the best model for predicting the number of hospitalizations based on the given COVID-19 dataset. When observing all relevant error metrics (RMSE, MSE, and MAE) the improved XGBoost model vastly outperforms the original XGBoost model.

In another way, this illustrates just how damaging overfitting can be to an analytical model.