Data 100: Final Project Modeling Report - Traffic Dataset

AJ Kumar, Raj Bhanushali, Cary Kuang, David Gomez Siu

Problem

The problem we attempted to solve through modeling is predicting the travel time from 300 Hayes St to any location in the Bay Area, and whether specific destinations were affected more significantly than others. Specifically, we wanted to look at office commutes and how travel times to corporate areas were affected by the pandemic and lockdowns. Our hypothesis is that areas surrounding large corporate offices experienced the greatest drop in travel times after the lockdown, as the implementation of work-from-home policies greatly reduced the amount of workers commuting to their offices. We were able to test this hypothesis using an outside dataset of major Bay Area office locations and the provided traffic dataset.

Answer

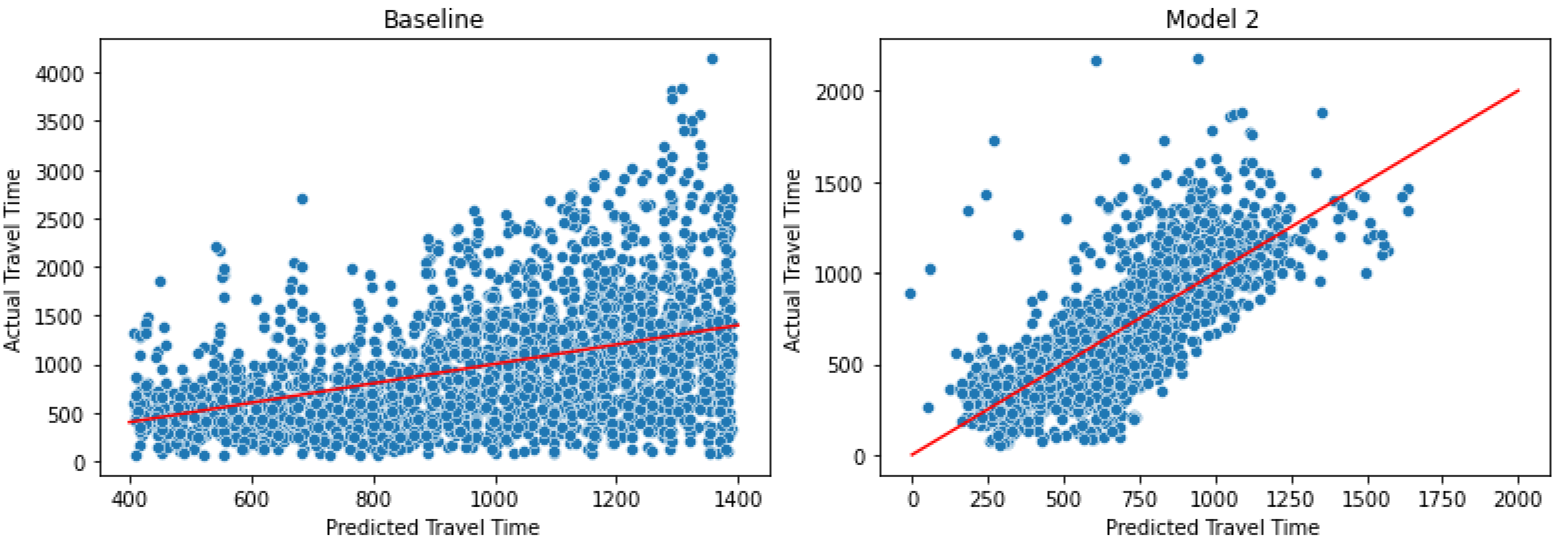
By using separate models from before and after the lockdown, we were able to measure the impact of office proximity on travel time from 300 Hayes St. For the pre-lockdown model, the near-office coefficient was -135.9977, while for the post lockdown model, the near-office coefficient went down to -60.23854873511756, with both models posting a similar R^2 value of 0.8075 and 0.8151, respectively. This meant that our hypothesis was incorrect; we predicted that destinations near office buildings would have the greatest speed up in travel time after the lockdown, but the model shows that the post-lockdown coefficient was less negative compared to the pre-lockdown coefficient. Since we are predicting mean travel time, this means that a less negative coefficient is slower compared to a more negative coefficient for the mean travel time. In other words, destinations that were near the major office buildings actually got slower after the lockdown compared to before the lockdown in terms of travel time. Since we know from our EDA that the average travel time decreased after the lockdown, we can reject our hypothesis that near-office destinations experienced the greatest speed up in travel times after the lockdown.

Modeling

To test our hypothesis, we decided to build a multivariate linear regression model to predict travel times from 300 Hayes St to another location in the Bay Area. The features we used in our final model were: indicator variables for the day of the month, indicator variables for the general region (to encode spatial information), the distance in miles between 300 Hayes St and the point we are testing, the latitude and longitude of the input point, the mean traffic speed collected from the Uber dataset at the specified date and location, and an indicator variable of whether the location is within a 1 mile radius of a major corporate office. The output of this model is a predicted travel time in seconds between 300 Hayes St, and our inputted destination. We chose a linear model because we saw that in the guided modeling section, we were able to get great accuracy on our speed predictions. In addition, it was important for us to use and interpret the coefficients in our model to test our hypothesis (specifically the office proximity variable). We settled on these features after much trial and error because we found that there was a delicate balance between improving the accuracy of the model by adding additional features and overfitting on these additional features. For example, we attempted to encode the spatial data by having an indicator variable for each census tract, but this resulted in our model being incredibly overfit and our test accuracy wildly fluctuating based on the randomness of our training-test split.

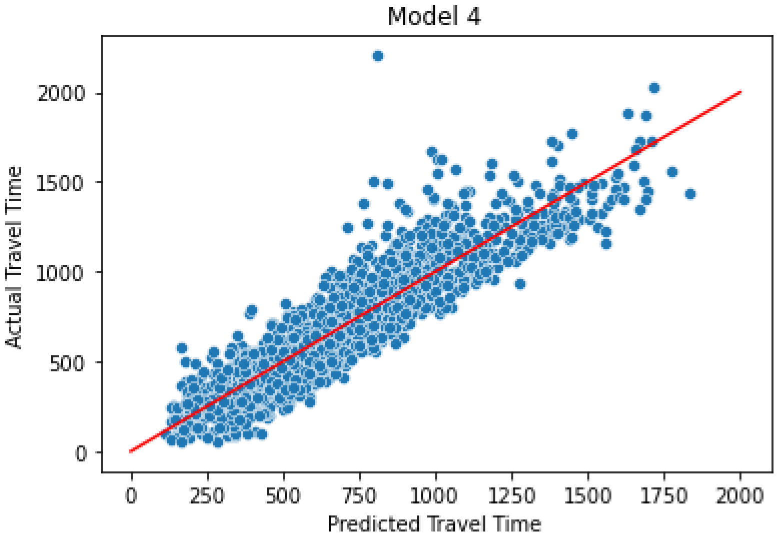
Model Evaluation/Analysis

For our first model improvement (Model 2 added longitude/latitude/speed), we saw a significant improvement in our accuracy, resulting in an increase of 30-40% for our model score.



From the graphs above, you can quickly see that the baseline model has a rather high MSE, with a large number of points varying above and below the Y=X line. Although you could also say this for model 2, the overall MSE for model 2 is much lower because the errors for predictions are much lower. This huge improvement is due to our new features, which are often correlated with travel time(destination coordinates,speed), especially since our base model only includes the destination movement ids and days, which are not enough to accurately predict travel times.

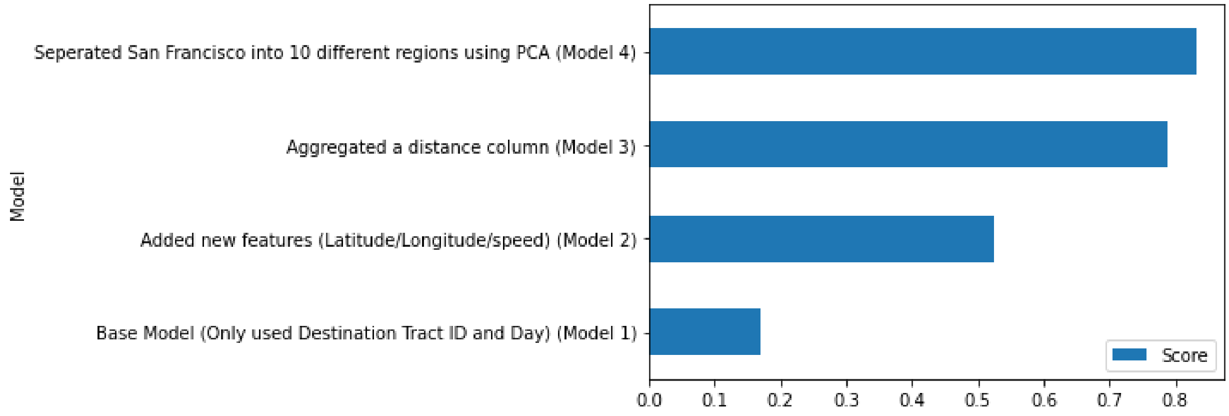
When looking at making the next model, we also paid attention to the fact that the second model has a significant number of outliers with low predicted travel times but high actual travel times. We also noted that the shape of the data seems to be slightly skewed counterclockwise, with a more positive slope. Our next model sought to incorporate features that would be able to account for the variability that we saw in the second model.



In our final model, our resulting model score increased from the previous model score by about five percentage points. We found that by including the following features in conjunction, we were able to get the highest scoring model that did not overfit the data: indicator variables for the day of the month, indicator variables for the general region, the distance in miles between 300 Hayes St and the testing point, the latitude and longitude of the input point, the mean traffic speed at the specified date and location, and an indicator variable of whether the location is within a 1 mile radius of a major corporate office.

The spread of points in this model seems rather uniformly distributed about the red line, indicating that there is much lower MSE for this model in comparison to the previous two. In addition, there seem to be many fewer outliers when using this model, which indicates to us that the inclusion of these features seems to explain some of the variability that we were able to see in previous models. Even if the model score seems to indicate only a small increase, visual analysis of the data spread on the last model provides a more tangible evaluation of the accuracy of our model. It should give us confidence that there is a lack of outliers. It is also possible that the outlier where the model predicted around 750 seconds when the actual travel time was around 2250 seconds is skewing the model score slightly. If we were to do further analysis on this model and this data set, we would want to pay particular attention to this data point to ensure that it is not an incorrectly inputted value.

Model Improvement



As a baseline, the first model we tested was a basic linear regression model that just took in the day and the census block (destination movement ID) as features, and predicted mean travel time. This resulted in an extremely low accuracy of around 18%, but we knew that this was just a starting point for us to experiment with what features we added and how the spatial information was encoded.

The first improvement we made to the model was to treat the destination movement ID as a categorical variable instead of a quantitative variable. Initially, we tried doing this by creating dummy variables for each ID, but this resulted in an extremely large number of variables and overfitting on the data. To counteract this, we began researching different spatial regression methods that would help us best represent location data in our model. Finally, we decided to use PCA on the latitude and longitude to split it up into ten different regions. We then used the region column as an input to our linear regression model, which improved the accuracy of our model.

The second improvement we made to the model was to find additional features that would give the linear model more information to work with without being overly specific. To do this, we figured that the latitude and longitude of the destination would give us some additional information. Moreover, this could be treated as quantitative data so there was no processing that needed to be done to it before feeding it into our model. Another feature we felt would be greatly useful was the location in miles between our starting point (300 Hayes) and our destination. To do this, we added another column to our DataFrame that calculated the distance between the two locations using a library called geopy, which gave us a precise estimation using the latitudes and longitudes of the two points.

The last improvement we wanted to make was to build the office proximity variable that would be used to test our hypothesis. To do so, we initially searched for a comprehensive dataset online of corporate offices in the Bay Area. Our hopes were that we would be able to filter out the biggest offices based on numbers of employees, and then use the latitudes and longitudes of these locations to mark points that were near large office buildings. However, the information we were looking for was not publicly available and was behind paywalls, so we decided to build our own dataset to serve as a proof-of-concept. To do this, we compiled the locations of some big companies in the Bay Area and thought of offices that were spread across different regions (e.g Santa Clara, San Francisco, East Bay, etc). We then added the office proximity variable by iterating through each row in our new datatable, and returning a 1 if we found an office that was within 1 mile of our input location. This serves as an easy way to mark each location's proximity to an office location, but requires iterating through the entire dataset for every point that we would like to predict on. If we had been able to acquire a larger dataset, we likely would have used geopandas or some other GIS software to perform this calculation much more efficiently.

Future Work

For future directions in our modeling, we believe that it would be better to have more parameters to work with and to also potentially improve our existing parameters. While there exists a more comprehensive dataset regarding Bay Area office locations, we were unable to access it because it was not available to the public. We could build on this concept and also try to encode other types of locations (restaurants, malls, sports arenas, bars, etc.) that are typically common destinations for uber rides within the Bay Area. Some other parameters that might be useful to investigate include weather, time of day, holidays, accident data, lane closures, and major events (i.e. parades, big games, etc.).

We also could expand on our model use, as we mainly focused on multiple linear regression. While we were able to create effective models (above 0.80 R^2 values for the predicted vs. actual values), we believe that we could get higher R^2 values by using more advanced models such as Random Forests or deep neural networks to make more robust predictions. If we were to combine this with more parameters, we could potentially reveal more complex interactions between traffic speeds and the environment around them, allowing us to potentially generalize beyond just the Bay Area.