Predicting the Commutes of 1990 San Francisco Bay Area Drivers

Supervised Learning Regression with Lasso/Ridge & Random Forests

Luke Catalano

I. Introduction

This paper presents a machine learning approach to predicting the commute time of individual San Francisco Bay Area workers around the year 1990 utilising numerous identifying features of their daily drive. Tech companies Google and Apple have solved the challenge of trip time prediction through machine learning methods paired with comprehensive real-time data to provide their users with accurate trip time estimates. Having insight into commute length allows for optimal schedule planning and aids an employee in choosing how far away from home they should work. Offering a precise trip time prediction software to the public is an incredible product, and is one of the most widely consumed products of machine learning today.

The machine learning models for this paper were trained and tested on data from the 1990 Bay Area Travel Survey and lack a valid representation of the environment that current-day Bay Area commuters face. A prediction can be attempted with more recent data in SF and other major cities, however, the out-of-sample deviance may be vastly different than the results of this study due to changes in commuter habits, population density, and highway design.

II. Data

The data for this study comes from the 1990 Bay Area Travel Survey, which randomly sampled households throughout the San Fransico Bay Area. One randomly selected household representative logged their daily drives for a week, providing surveyors with their daily trips. The study resulted in 9,439 individuals from unique households in the Bay Area reporting 70,774 trips. We will be exclusively working with records where an individual was driving to work, the total trip time was greater than 0 minutes & less than 200 minutes, and the individual was less than 100 years old, leaving us with 24,632 total valid trips. We will not be limiting the mode of transportation the individual uses to get to work, allowing for bicycle/walking commutes.

There are 87 available features in this dataset, the most important for this prediction include departure and arrival times (in military time), the day of the week, the individual's trip number of the day, the mode of transportation and vehicle type, the major bridges crossed during the trip, the origin location purpose (Home, Work, etc), the destination location purpose, an indication of if the trip was a carpool, the person's home county and work county, and the person's occupation. Our outcome of interest is the time it took an individual to get from their origin location to their destination location, so we take the difference between the destination arrival time and the origin departure time to get our outcome variable of time. For this analysis, we will treat a difference of *X* between destination and origin time as *X* minutes, regardless of the original time format of origin/destination time. **Figure 1** presents the descriptive characteristics of the outcome variable of interest, time:

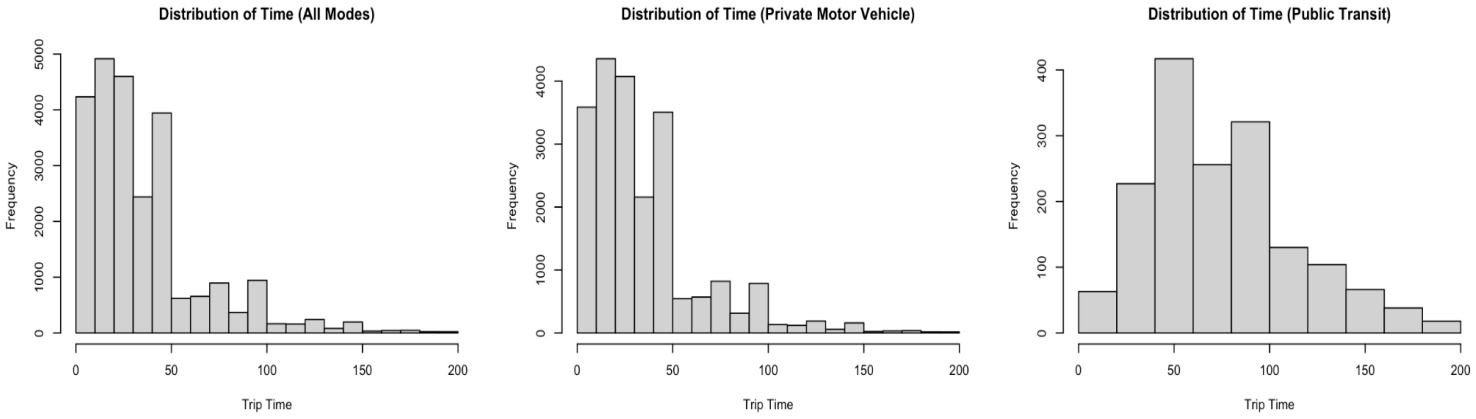
Figure 1: Descriptive Characteristics of Trip Time

Descriptive Characteristics - Outcome

Variable	Obs.	Mean	Std. Dev.	Min	Max
time	24632	36.63	30.9	1	198

A minimum trip time of 1 minute likely indicates someone lives on top of or next to where they work, which is possible in a dense city environment. A maximum of 198 minutes likely indicates a trip to an out-of-city office, a result of a major traffic collision, or delays in public transportation. The average commute was 36.63 minutes, with a standard deviation of 30.09 minutes, suggesting significant variation in the trip times per individual, which can lead to bias in our machine learning prediction. A large amount of this variation in our dependent variable is likely attributed to outlier trips with commute times over 2.5 hours. Many of the outlier trips involved public transportation, which is worth exploring. **Figure 2** presents the distributions of our outcome variable commute time for different methods of transportation.

Figure 2: Distributions of Trip Time



The frequency distribution of trip times is skewed right for all three distributions, with a majority of the trips taking under 100 minutes to complete regardless of the mode of transportation. For those in private motor vehicles, a low proportion of commuters experience commutes longer than 60 minutes, and only a handful of observations experience commutes longer than 150 minutes. This changes greatly for public transportation, which sees a larger proportion of commutes taking more than 60 minutes to complete, and makes up the large majority of the outliers in our overall dataset. The average total time for trips using public transportation was 75.67, more than two times the average for all commutes. This implies that the mode of transportation is a vital feature to include in the prediction models as it greatly affects an individual's commute time.

III. Methodology

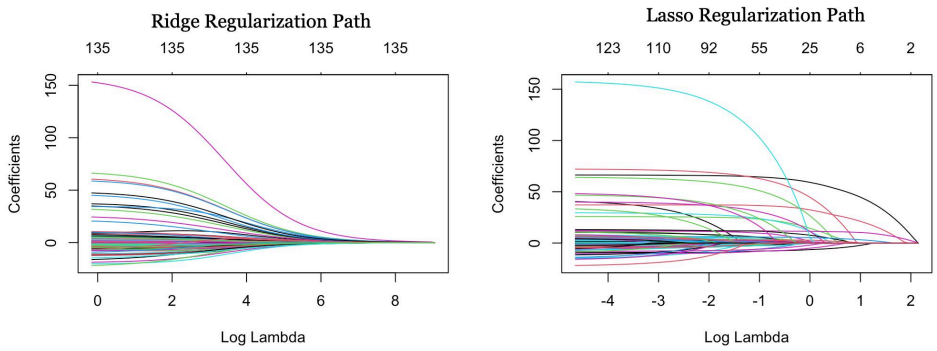
For this prediction problem, I will use lasso and ridge regression for regularization, along with random forests as a non-parametric alternative. The goal is to create a model that provides us with the most precise out-of-sample predictive performance, stemming from a low out-of-sample deviance. Rather than using OLS linear regression to estimate the total commute times by minimizing the model's in-sample deviance, I am opting to use lasso and ridge regression to

regularize my models and avoid overfitting by systematically decreasing the model's complexity. Overfitting occurs when a model fits current in sample data too well often due to high complexity (the inclusion of too many features), which is generally an indicator that the model is fit to the noise of our current data. An overfit model's prediction is not representative of the noise in future out-of-sample data, harming its out-of-sample predictive capabilities. Regularization prevents overfitting by reducing model complexity, making estimated feature coefficients smaller by applying a regularization penalty, and even shrinking them to zero in some cases.

The ridge model selects the estimated feature coefficient that minimizes the in-sample deviance, plus a squared penalty. This squared coefficient penalty prevents the value from being shrunk completely to zero, but often ridge shrinks the coefficient so close to zero that the feature has very little effect on the final prediction, decreasing complexity. The lasso model operates in nearly the same way, with a slight adjustment to the penalty. Rather than a squared penalty, lasso regression applies an absolute value penalty instead. This allows some coefficients to be shrunk to exactly zero, excluding unnecessary features completely. In both methods, the penalty term is multiplied by a penalty parameter λ chosen by the researcher.

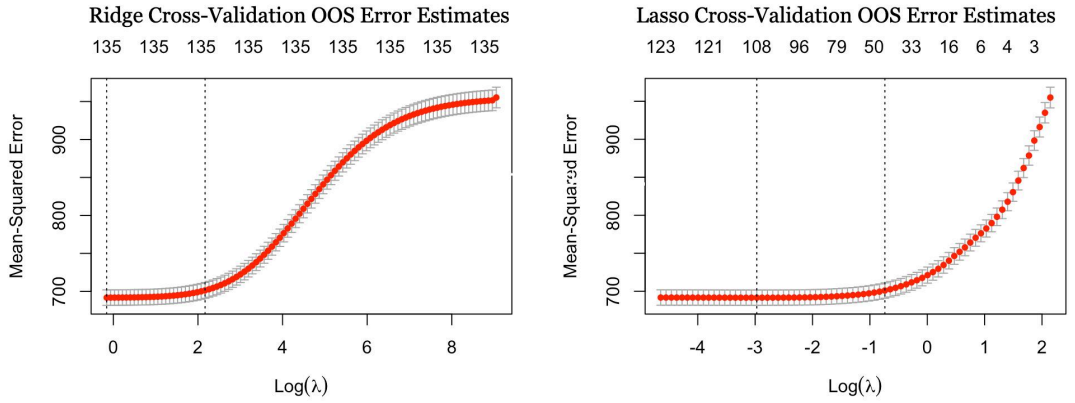
The λ penalty parameter in lasso and ridge regression is selected to optimize the balance between bias and variance, which have an inverse relationship known as the bias-variance tradeoff. As we increase model complexity by reducing regularization (lower λ), we decrease bias but increase variance as our model becomes fit to the noise of current data. Intuitively, as we decrease model complexity by increasing regularization strength (higher λ), we increase bias but decrease variance, as our model now contains fewer features to make predictions. **Figure 4** shows the regularization paths of the coefficients on all covariates in our lasso and ridge models for this prediction as the natural log value of the λ penalty parameter increases.

Figure 4: Commute Time Model Regularization Paths



The value of the λ penalty parameter is chosen such that the cross-validation OOS error estimate from K-Fold cross-validation is minimized. K-Fold cross-validation is the process of systematically splitting the data to train a model and estimate its out-of-sample predictive performance. For each fold of data in *K* total folds, we train a model on all but the current fold, setting the current fold as our test fold. We predict the values of the test fold using the training model, which allows us to estimate an OOS deviance as the test set is not in the training sample. This process is repeated K times, iterating on the values of K until we have K total OOS deviances. We then take the average of these OOS deviances to arrive at the cross-validated estimate of the OOS deviance. **Figure 5** is a plot of the cross-validated OOS deviance error estimates, showing the estimated range of OOS deviance of our commute models for each log λ penalty.

Figure 5: Commute Model's Cross-Validation OOS Error Estimates



The minimum cross-validated OOS deviance in each of these graphs is indicated by the far-left line in this example. The final λ penalty for each model is the exponentiated log λ penalty at this far-left minimum CV OOS deviance line. This is a crucial step in optimizing the model's predictive performance on unseen data, as the model with the lowest cross-validated estimate of out-of-sample deviance will likely perform the best on any new out-of-sample data fed into the model, assuming the data the model was trained on is representative of the new unseen data. This cross-validated OOS deviance is then compared to the null deviance (out-of-sample deviance when predicting the sample mean of our outcome variable for each observation) to produce an OOS R-squared goodness of fit estimate.

I will incorporate an interaction term into the lasso and ridge models to fully capture the effects of driving between the home county and work county. Individuals are more likely to start their trip in their home county and are guaranteed to end their trip in their work county. Interacting home county with work county opens up the model to capture the effects of distance due to the difference in the average time for multiple-county trips vs. in-county trips, which does not exist by default in any of the features available to us. Ideally, we would be able to compare the mean of cross-country-border trips with that of inter-county trips, however, one of the limitations of our data and this interaction term is that trips don't always begin in the individual's home county. A trip back to work from the doctor's office for individual A is logged with the same home county as a trip from individual A's home. Regardless, cross-validated OOS deviance increased when this interaction term was included in the ridge and lasso models, as it is still likely to capture even the distance effect for those travelling from their home counties to work.

Lasso and ridge regression are forms of parametric estimation, depending on model parameter inputs that could be unrepresentative of the data distribution. By assuming a linear

functional form for each covariate, we risk leaving the model misspecified if we train the model on the wrong functional form. Due to my uncertainty about the distribution of each included covariate in my parametric models, I will include a Random Forests model in this study to compare the predictive performance, attempting to make the best predictions possible.

Random Forests provides a non-parametric estimation constructed on the concept of decision trees (specifically regression trees), a nested series of if-else statements that the observations are filtered through to output predictions, with each node (split) in the tree dividing the range of covariates into two halves and sorting each observation into a new node. As a result, we end up with a non-parametric estimate of commute times, where each individual's commute time estimate is the average time of the bin that the individual was sorted into. The Random Forest model produces an outcome equivalent to the average estimate of many optimally classified regression trees, where the split in each node of an optimized tree is determined such that the in-sample deviance of the observations in that leaf node (a subset of nodes that do not have children nodes) is minimized. Random forests rely on in-sample resampling, generating a new random resample for each optimally classified regression tree to reduce the variance of our estimate, which could otherwise be very high due to the decrease in bias that comes from a large regression tree. The "min.node.size" argument declares the minimum amount of observations allowed in a node for it to be considered for further splitting, limiting the growth of our tree to our desired size. We specify the number of optimized trees we'd like to generate through the "num.tree' parameter, limiting our random forest model size to our desired value. The final non-parametric estimation is the average of all optimally classified regression trees for each terminal bin. We then can input our data into this model to generate non-parametric estimates of commute time for individuals.

These models rely on a crucial assumption that the data we train our model on is representative of the data we'd like to make predictions on. We have data from the SF Bay Area Travel Survey from the year 1990, so it would not be possible to predict the commute times for drivers in 2025 due to various changes in the commuting environment, such as changes in population, highway construction and renovation that decrease (or increase) commute times, changes in the types of vehicles, and changes to the percentage of the population that utilizes public transit. It may still be possible to make predictions on new data if it comes from trips that occurred in the 1990 Bay Area. To make predictions on the commutes of SF Bay drivers in 1990, the types of drives would need to be representative of the population of the SF Bay area in 1990. **Figure 3** showcases the different kinds of drives our data contains, drilling down the origin of all 24,632 drives to work into their levels.

Figure 3: Origin Level Counts

Levels Count - Origin Purpose

Origin Purpose	Count	Percent of Total
Work Related	7615	30.90%
Child Care	2780	11.20%
Medical / Dental	2655	10.70%
Home	1	0.00%
Other	9456	38.30%

We find that participants in the 1990 SF Bay Area Travel Study did not select their home as their origin location, despite the option in the survey. We can infer that this is not representative of the true distribution of trip types in the SF Bay Area, as the vast majority of employees drive to work from their homes. This leads us to conclude that the representativeness assumption does not hold, and we will be unable to make accurate predictions on data that aligns with the true distribution, regardless of any low OOS deviance or high OOS R-squared value.

IV. Main Results

Our Ridge and Lasso models use the same features to predict the commute time of SF Bay Area residents in 1990. It's important to remember these predictions are not representative of the true population's commute times in the 1990 Bay Area, and we are simply trying to maximize out-of-sample predictive performance on our chosen dataset. Our models will include indicators for the mode of travel, origin purpose, travel day, carpool, first bridge crossed, second bridge crossed, business type, occupation type, vehicle type, an interaction term between indicators for work county & home county, and individual's trip num. of the day. We set the number of folds to 20 to provide a cross-validated estimate of the OOS deviance and R-squared and will explore the model performance where the λ penalty is optimized.

The lasso and ridge models had nearly identical OOS R-square values of .2757 and .2758 respectively, a measure of the goodness of fit to the OOS data with 1.00 being a perfect fit. The OOS deviance for the ridge and lasso models are 691.76 and 691.55 respectively, which is 264 less than the null deviance for each model. The null deviance in this context is simply a prediction of the sample mean commute time for each observation. Our model is likely suffering due to the parametric nature of our estimation, as we do not know the functional form of each feature on total commute time.

The top predictors for commute time in the lasso model were mostly modes of travel, including an indicator for an Amtrak commute ($\widehat{\beta}_{Amtrack} = 70.75$), an indicator for a CalTrain commute ($\widehat{\beta}_{CalTrain} = 65.82$) and an indicator for a Motorcycle commute ($\widehat{\beta}_{motorcycle} = 62.24$). All else held constant, taking the Amtrack to work was associated with an average increase of 70.75 minutes in commute time, and taking the CalTrain was associated with an average increase of 65.82 minutes. This is consistent with our findings from **Figure 2**, where we identified that the average commute time for public transportation users was more than double that of private

vehicle users, suggesting that public transport significantly impacts the total commute time. The coefficient on the indicator for driving a motorcycle to work surprised me, as I would imagine being able to split lanes would cut down commute time significantly. Driving a motorcycle to work was associated with an average increase in commute time of 62.24 minutes, holding all else constant. The ridge model produced very similar results, with the top three most predictive features being an indicator for Amtrak commute ($\widehat{\beta}_{Amtrack}$ = 66.57), an indicator for CalTrain commute ($\widehat{\beta}_{CalTrain}$ = 60.80), and an indicator for Motorcycle commute ($\widehat{\beta}_{motorcycle}$ = 58.95). With OOS goodness of fit values so close to each other, it is not surprising that these models have similar coefficients on their most predictive features. One feature I'm surprised is not very predictive of commute time is the indicator of a trip being a carpool. I'd imagine coordinating pickups and dropoffs would make a trip more complicated and lengthy, but a trip being a carpool was only associated with a 3.45-minute increase in the ridge model, and a 3.64-minute increase in the lasso model, holding all else constant.

The parametric approach of lasso and ridge has a relatively low OOS goodness of fit for both models, performing only moderately better than the null model. Our non-parametric random forests approach assumes nothing about the functional form of each included feature. We will generate 500 optimally generated trees, with a minimum node size of 10. We have included all of the features from the lasso and ridge models, removing the interaction term and turning it back into two separate features of work county and home county. Our Random Forest model has an OOS R-squared goodness of fit value of 0.3329, performing about 5% better on out-of-sample data than our parametric lasso and ridge models. This suggests there are certain interactions in the data that we are not properly specifying in the parametric models, or the form of the true data distribution is not linear, limiting our linear model's OOS predictive performance.