Regression models for predicting rail transit ridership at the station level

Daniel Hartig

1 Introduction

The United States is undergoing a rail boom. Since 2010, new light rail lines have opened in Dallas, Los Angeles, Salt Lake City, Denver, Minneapolis, Houston, Seattle and more. A new heavy rail line opened in Washington DC, and a commuter rail system in Orlando. As transit expands in cities in the United States, there is an opportunity to validate predictive rail ridership models.

A survey of transit agencies [1] conducted by the Transit Cooperative Research Program showed that relatively few agencies are using quantitative models when forecasting ridership for new lines, extensions or stations under consideration for funding. Of the 35 agencies that responded to the survey, 29 use professional judgment and 28 use rules of thumb among one or more techniques used to generate ridership forecasts. Another method used by 22 agencies is service elasticity; a set of general transit demand response curves for changing transportation options, published by the Transportation Research Board [2].

For quantitative methods, the most commonly used technique–by 18 of 35 surveyed agencies–is the four-step travel demand model [3], introduced by Mannheim and Florian [4, 5]. The Mannheim-Florian model's four steps are trip generation, trip distribution, mode choice and route choice. In the trip generation phase, trip endpoints are created with production and attraction ends. In the trip distribution step, these endpoints are paired up to generate trips; for example a residence is paired with a job location for one trip, or a hotel is paired with a tourist attraction. In the mode choice step, trips are assigned to various transportation modes, such as personal vehicle, bus, or walking. Finally, in the route choice step, a route using that mode of transportation is chosen.

An implementation of the Mannheim-Florian model can be seen in the Seattle's Sound Transit Ridership Forecasting Methodology Report [6, 7]. The Sound Transit 3 (ST3) was a ballot measure that passed in 2016 for a $54 billion expansion of the local light rail system involving 100 km of new tracks and 37 new stations. The ridership forecasting methodology report explained how the project's official ridership projections were developed. The regional area is divided into 785 Alternatives Analysis Zones and for each of these zones transit surveys and recorded ridership on local bus routes were used to complete the trip generation and trip distribution steps. The mode choice and route choice is done using an incremental logit model to predict changes in transit mode based on changes in transit mode availability.

A significant weakness of the four-step method is that it relies on already known transit ridership [8]. For a proposed light rail system, for example, existing bus routes would be used to complete the trip generation and trip distribution steps.

A regression based ridership model could provide an alternative estimate of ridership without depending on previously known transit ridership counts. Only seven of the 35 surveyed agencies used regression models to predict future transit ridership. This thesis proposes a regression-based model using data from the United States Census Bureau at the zip code level. The model will be trained on the zip code characteristics and ridership data from existing light and heavy rail transit systems. The resulting model will be used to predict ridership on other rail transit systems.

1.1 Original Contributions

There are two main objectives of this thesis. The first objective is to investigate the utility of various data from the US Census Bureau as predictor variables in a regression model for urban rail ridership. The second objective is to determine what generalized linear regression methods–in terms of the choice of loss function and link function–are best suited to modeling urban rail ridership given the available data.

This thesis diverges from previous regression modeling by investigating an expanded set of potential features. The US Census Bureau provides extensive data on a per-zip code bases as detailed in Section 2.2: there are over a thousand possible features available. This data is provided as counts per zip code. To translate this zip code data into features for a regression model, this thesis presents a novel geographic sampling method in Section 2.4.

With features for regression analysis in hand, this thesis benchmarks several different regression methods. Previous regression based analyses of urban rail ridership have exclusively used ordinary least squares regression. This thesis expands the model base to include other loss functions and link functions and reports accuracy metrics for each model.

2 Data Sources and Feature Generation

2.1 Selection of Transit Systems

The response variable for the regression analysis is average weekday ridership over a period of at least one year. Ridership data for agencies that publish annual ridership reports is used to validate the model (for ridership data sources, see Appendix A.1). Six cities were selected for this study: Boston, Chicago, Los Angeles, Atlanta, Dallas, and Denver. Several cities were eliminated from analysis for various reasons. The author was unable to locate ridership information for several cities such as Houston and Miami. A limitation of the Census dataset is that it does not include government employment. While state level employment is significant in all potential cities, state employment levels are relatively constant from city to city. Federal employment varies greatly, however; Washington DC and San Diego with its military installations had to be eliminated due to the large impact of un-recorded federal employment. San Francisco and Philadelphia were eliminated because they have multiple rail systems without integrated fares. New York City was eliminated because its subway has higher ridership than all other intra-urban rail systems in the country combined.

Ridership data is captured by different methods in different cities, but the same underlying data is counted by each case. In Chicago and Boston, the transit agency measures paid station entries. Boston additionally reports transfers between lines at each transfer station. Therefore, daily ridership is measured as the total number of entrances per station per day. This results in one count per one trip; and two counts per one daily commute. Dallas, Denver and Los Angeles report boarding and alighting by train car; so daily ridership is measured as the total number of boardings per stations per day. For transfer stations, boarding is 'double counted' since a boarding at station A and a transfer at station B would result in two counts per one trip. Therefore, for these three networks, the transfer stations and their unusually high ridership counts are excluded from the analysis. For Atlanta, the ridership reporting definition is unknown, so we eliminate the one transfer station as a precaution. For Boston and Chicago, ridership is counted by extracting data from the fare system for paid station entries. In Dallas and Denver, passenger counts are measured at the train car by Automated Personnel Counters. In Los Angeles and Atlanta the counting methodology is unknown.

The data closest to 2015 is used when possible to get an accurate relation between ridership and census data. The census data as well as Chicago, Dallas, and Denver's ridership statistics are from 2015. Boston's ridership is from 2014, Los Angeles' is from 2013-2014, and Atlanta's is from 2010-2013.

2.2 Data Sources for Predictor Variables

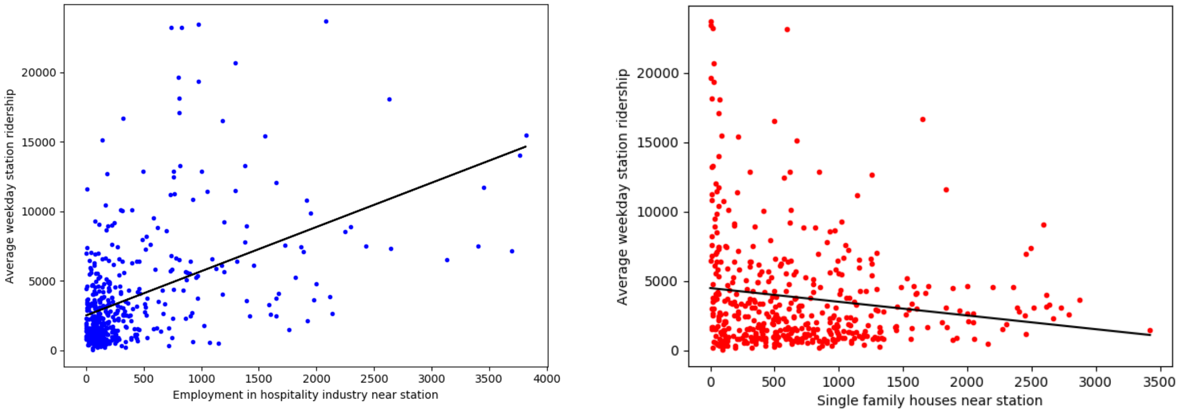
The zip code level data for feature generation comes from the US Census Bureau and is available at factfinder.census.gov. There over one thousand potential data sets available. We refer to the data available for each zip code as characteristics of the zip code. Selection of features is guided by Kuby [9], Taylor [10], and Currie [11], who demonstrate the significance of characteristics such as employment, population, universities, poverty, airports, park and ride stations, and rental housing units. The characteristics can be divided into two categories: counting characteristics and dummy variables.

This model emphasizes using only features that have 'real' units, as opposed to binary features. The only binary dummy variable this model uses as a feature the presence of park-and-ride parking spaces at a transit station. Instead of using measures of land use mix as proposed in other models [12, 13], or binary dummy variables for universities and central business districts, the equivalent information is provided naturally as counting data in the feature set. Counts of housing types (such as large apartment buildings versus single family homes) replace land use mix; the number of jobs at universities or in financial jobs provide more information than dummy variables for presence of universities or central business districts.

Population related	**Employment related**
Total population	Total Employment pay
Population over age 65	Employment in finance industry
Population with bachelor's degree	Employment at hospitals
Number of residents employed full time	
Number of housing units built after 2000	

Table 1: Examples of characteristics associated with zipcodes

The selected characteristics are generally related either to the population of the zip code or the number of jobs of the zip code. The captured characteristics of the built environment, such as the quantity, age, and building type of the housing stock, are generally related to population. For example, the total number of housing units is expected to be highly correlated with the population of a zip code. Examples of characteristics are provided in Table 1. Some characteristics, such as total population, are expected to have a positive relationship with ridership; others, such as population under the age of 18, are expected to have a negative relationship. Examples of associations between characteristics and ridership are provided in Figure 1. A summary of all the selected characteristics is provided in Appendix A.2.



(a) An example of a positive association

(b) An example of a negative association

Figure 1: Single covariates against the response variable with ordinary least squares (OLS) line of best fit.

In Yao [14], a distinction is made between 'Need Index' features, a series of features that depend on the characteristics around the station and are independent of the transit network; and 'Transit Network' features, which do depend on the travel time between stations of the transit network. To model network features for each station, the sum of each characteristic for every other station within 15 and 30 minutes transit time is included as a feature of the original station, as described in Section 2.6. This provides us a quantitative way to express the 'centrality' dummy variable that is provided as a flag in many models [9, 12]. Centrality could be proportional to the count of population or jobs within 30 minutes of a station, for example.

We translate zip code level data into transit station specific data by sampling each zip code's geographic area to determine proximity to a transit station. For each zip code near the transit network, a set of random points within that zip code is generated using rejection sampling. For each of the those points, one or more closest stations are determined. Each point is assigned to one or more station within walking distance, using the method defined in Section 2.3. Counts for the characteristics of each zip code, such as population or employment, are then assigned to each station proportional to the number of points assigned to each station.

2.3 Definition of 'Walking Distance'

The area within walking distance of a station is called its 'catchment'. To generate a feature for a transit station, the total count of some characteristic within the catchment of that stations is considered. The standard transit catchment distance in literature for rail stations is one half mile (800 m). Guerra [15] suggests that one half mile is more appropriate for population as a feature while one quarter mile (400 m) is better for employment as a feature. A case study [16] from a 2003 Montreal transit riders origin-destination survey concluded that approximately 50% of riders of the city's urban rail transit walk less than 500 meters to their stations, while 90% walk less than 1000 meters. The maximum walking distance is approximately 1500 meters. In another regression-based study of catchment sizes, Gutierrez [13] found the optimal distance for for assigning population and employment to a station was between 600 and 900 meters in straight line distance. A straight line distance is measured as the crow flies as opposed to a distance following pedestrian walkways; this distance measurement is consistent with the distances used in this analysis.

A significant concern when tabulating catchments is that in many of the densest regions of cities, there are many stations within a few hundred meters of each other. For example, from the Massachusetts State House in downtown Boston, there are seven urban rail stations on four different train lines within a 10 minute walk, according to Google Maps. Any of those stations could be the 'best' place to board and disembark the train for a commute to work at the State House, depending on the direction and train line by which the commuter was arriving in downtown. It is important that we allow some 'overlap' in catchment areas; people at one geographic location could use more than one station for various transit needs.

Given this information, we choose cutoffs of 500 meters and 1000 meters for calculating station distances. For any location that has residents, jobs, or other desired countable characteristics, all stations within 500 meters will be considered equally likely to capture a share of that resident or job's transit demand. If there are no stations within 500 meters; then all stations within 1000 meters will be considered.

2.4 Rejection Sampling of Zip Code Shapefiles

We translate zip code level data into transit station specific data by using a Monte Carlo method to estimate feature counts near transit stations. Sample points are generated within each zip code near the transit network. Those points are assigned to whichever stations are within walking distance of the station, as defined in Section 2.3. If there are multiple stations within walking distance, then equal fractions of that one point are assigned to each station. The ratio of points assigned to each station to total points generated for each zip code is used to assign feature counts to each station, such that

$$\text{count}(station_i) = \sum_j w_{ij} \cdot \text{count}(zipcode_j)$$

where w_{ij} is the proportion of the points generated in zipcode j that are assigned, wholly or partially, to station i. Not all areas within a zip code j will be within walking distance of any station i, so

$$\sum_j w_{ij} \leq 1$$

for all j. An algorithm for rejection sampling a single zipcode is provided in Algorithm 1.

Algorithm 1 Algorithm for estimating characeristic counts that are near transit stations

```
 Given zipcode is a single zipcode near the transit network
 Let zipcode.latrange and zipcode.lonrange be maximum and minimum latitude or longitude for zipcode
shapefile
 $n \leftarrow$ max(zipcode.area in hectares, 1000)
 randomPoints$\leftarrow$ {}
while len(randompoints) $< n$ do                          ▷ Generate n random points within zipcode
    lon $\leftarrow$ random number $\in$ zipcode.lonrange; lat$\leftarrow$ random number $\in$ zipcode.latrange
    point $\leftarrow$ (lat, lon)
    if point is inside zipcode.shapefile and point is not inside exclusion areas then
        randomPoints $\leftarrow$ point
for point $\in$ randomPoints do                                ▷ Assignment of points to stations
    $n_{0.5}$ $\leftarrow$ number of stations within 0.5 km of point
    $n_1$ $\leftarrow$ number of stations within 1 km of point
    if $n_{0.5} > 0$ then
        for station within 0.5 km of point do
            station.characteristicValue += zipcode.characteristicValue / $n_{0.5}$
    else if $n_1 > 0$ then
        for station within 1 km of point do
            station.characteristicValue += zipcode.characteristicValue / $n_1$
```

The US Census Bureau provides TIGER/Line shapefiles of each zip code tabulation area (ZCTA) in the United States at `https://www.census.gov/geo/maps-data/data/tiger-line.html`. Random points are generated in a rectangular box drawn around the extremities of each zipcode's shape; these random points are accepted if they are within the shapefile or rejected if they are outside it. Those points that are inside the shapefile are tested against author-created exclusion zones. These zones are shapes within the zip code's shapefile area that are known to not have any population, employment, or other countable characteristics. The exclusion zones are mostly drawn over water areas and parks. Those points that are inside the exclusion zones are also rejected. This creates a set of points randomly drawn from the zip code's land area, not counting unoccupied regions like parks.

The set of points is tested for their distance to any transit stations to determine which station catchments they fall in, as described in Section 2.3. The characteristic counts associated with each tested point are divided between all stations within 500 meters. If there are no stations within 500 meters, then the point is divided

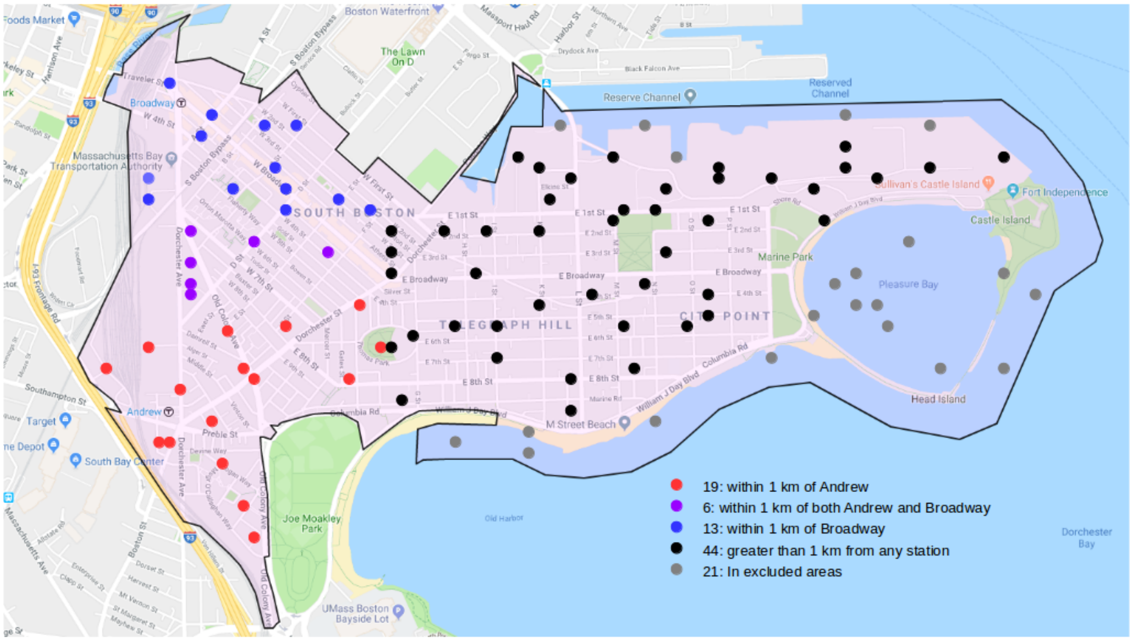


Figure 2: Illustration of nearest zip code estimation for zip code 02127.

between all stations within 1000 meters. If no stations are within 1000 meters, that point is not assigned to any station. The total sum of points and fractional points assigned to each station is divided by the total points available to get the fraction of each of the zip code's characteristic data counts is assigned to that transit station.

An example using zip code 02127, the South Boston neighborhood of Boston, illustrates the sampling method (Figure 2). 100 random points are selected within the area of the shapefile. Of these, 21 points indicated in gray are rejected due to exclusion areas based on water area, parks and abandoned port facilities. Of the remaining 79 points, 8 are within 500 meters of Andrew station, while 6 are within 500 meters of Broadway station. Moving out to the 1000 meter radius, 11 are within 1 km of Andrew station, for a total of 19 that are close to Andrew; 7 are within 1 km of Broadway station for a total of 13 that are close to Broadway; and 6 are within 1 km of both. The 6 points within 1 km of both stations are divided evenly between the two. The total population of South Boston is 36494. Therefore,

$$\begin{aligned}
\text{Counts assigned to station} &= \frac{\#\text{ points for one station} \cdot \sum \frac{\#\text{ points for multiple stations}}{\#\text{ stations for each point}}}{\#\text{ points for zipcode}} \cdot \text{characteristic counts} \\
&= \frac{19 + \frac{6}{2}}{79} \cdot 36494 \text{ people} \\
&= 10163 \text{ people}
\end{aligned}$$

are assigned to Andrew station. Similarly, 7391 people are assigned to Broadway station. Summed over all zip codes near Andrew and Broadway stations, this shows how the total population within walking distance of the station is estimated. This calculation is performed for all countable features and all zip codes and summed total counts for each characteristic are used as a feature for each transit station.

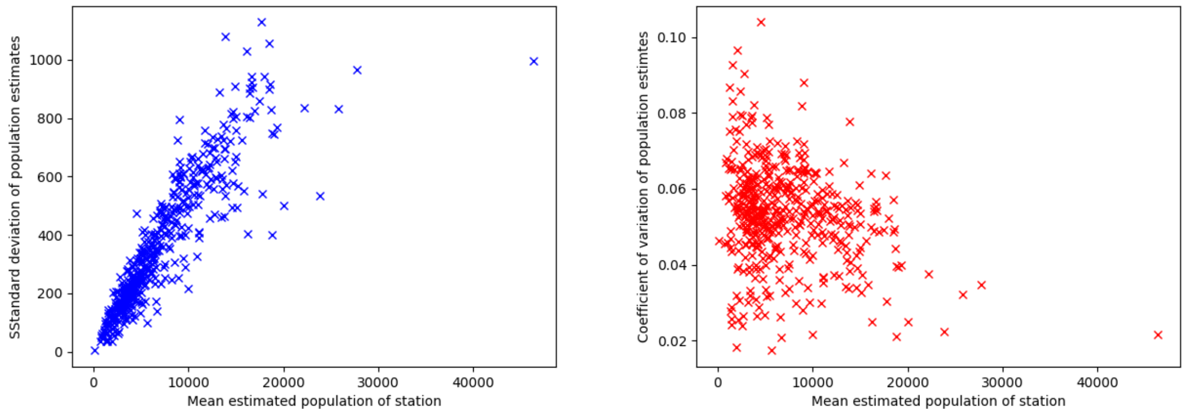
2.5 Variance of Monte Carlo estimates

With any Monte Carlo method for estimation, there is variance in the estimates generated. In this case, the variance comes primarily from the random locations of the points. It is possible that in different Monte Carlo trials, a station may get significantly more or less points nearby it. This is especially significant in areas of high population density; one extra or missing point could be worth hundreds or thousands of jobs in the densest areas of downtown Chicago or Boston. To keep variance to an acceptably low level, we must generate enough sample points that variation between trials is minimal.

The number of points that will be accepted by rejection sampling is set beforehand. We generate random points, and use rejection sampling to see if they are inside the shape of the zipcode and outside of any exclusion zones. We continue generating points until we have reached the desired number of accepted points.

The land area of the zip codes near the studied transit networks vary in size from as small as 30 hectare in downtown Chicago to as much as 18100 hectares at the suburban end of transit lines in Dallas. To provide an appropriate balance between accuracy and processing speed, we use one accepted point per hectare, but with a minimum limit of 1000 accepted points per zip code. This effectively provides over 10 points per hectare in for the zip codes in the densest parts of the studied networks: the downtown areas of Chicago and Boston. These areas also have the densest network of transit stations, with many stations within a kilometer of each other. In these denser areas, it is important to have enough points that the division of points between stations does not result in too much variance.

We generate 100 sets of estimates for the a single feature (total population) for all stations in the six transit networks. Figure 3(a) shows a graph of means of total population estimates against standard deviation of the population estimates, while Figure 3(b) is means against coefficient of variation. The standard deviation shows an increasing linear relationship with the population mean. The coefficient of variation is never greater than 10.4% and generally decreases with increasing estimated mean population. The mean coefficient of variance over all studied stations is 5.3%. By this method, the relative accuracy of any single station estimate of population or employment relative to the mean of 100 estimates of the same feature ±10% .



(a) Standard deviation versus mean by transit station.

(b) Coefficient of variation versus mean by transit station.

Figure 3: Indications of variance for Monte Carlo estimates of the population feature.

2.6 Generation of network-dependent features

For each 'Need Index' type feature generated by rejection sampling, a set of corresponding network-based features are generated to represent the sum total of a certain characteristic (such as population or employment) within a given travel time of that station. An algorithm for calculating network characteristics is shown in Algorithm 2.

Algorithm 2 Algorithm for calculating network characteristic counts

```
for station∈ transit network do
    nearby15 ← all other stations within 15 minutes travel time of station
    nearby30← all other stations within 30 minutes travel time of station
    for every characteristic in the model do
        for otherStation ∈ nearby15 do
            station.characteristicValue15 + = otherStation.characteristicValue
        for otherStation∈ nearby30 do
            station.characteristicValue30 + = otherStation.characteristicValue
```

The transit network is laid out as a directed graph, where nodes represent the transit stations and edges are weighted by the travel time between the stations. Travel times between stations are available in the transit schedules published by the appropriate transit agencies. Travel times can be different in different directions,

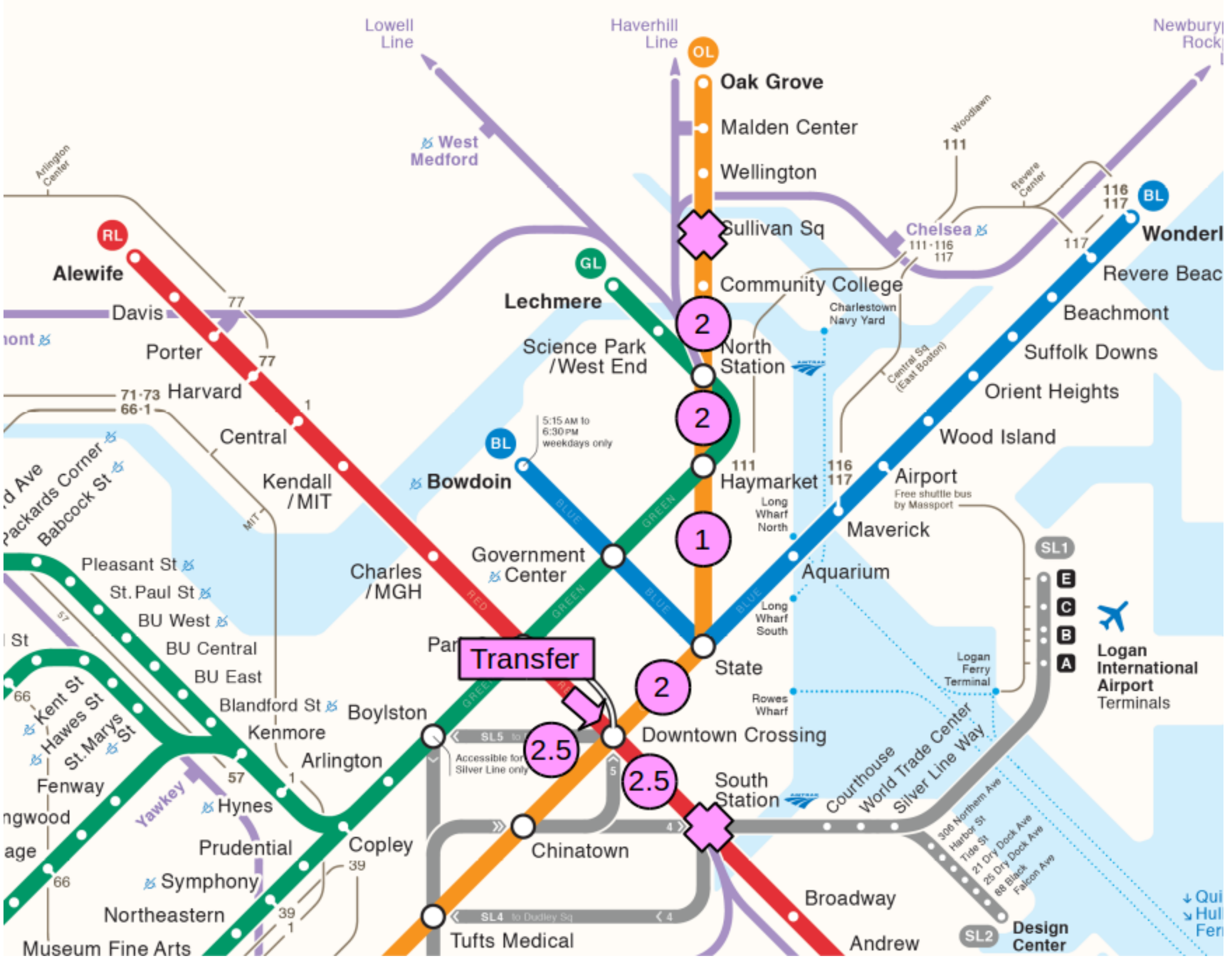


Figure 4: Illustration of travel time calculation for Sullivan Square to South Station, in Boston.

following the published schedules. At transfer points, there is a separate node for a single station on each line. The multiple nodes for the same station have edges between them weighted by the average wait time between trains. The wait time is estimated at half the time between trains at the station being transferred to. For example, if one train arrives every 10 minutes, then a transfer to a node on that line at any station will have an estimated travel time of 5 minutes. Many transit agencies align arrival times so that one line will depart a few minutes after another train arrives. No attempt is made to capture this more complicated arrangement of estimated transfer times.

An illustration of the calculation of travel time between Sullivan Square and South Station in Boston is provided in Figure 4. Starting at Sullivan Square on the Orange Line southbound, there are four edge traversals totaling seven minutes to get to Downtown Crossing. From there, there is a 2.5 minute wait until a Red Line (also Southbound) train arrives, and 2.5 more minutes of travel to South Station. The total travel time is thus twelve minutes.

For any given station, we find a set of all stations within aX distance by selecting an ego graph out of the entire transit network. An ego graph is a sub-graph containing all nodes that are within a specified distance of the central node.

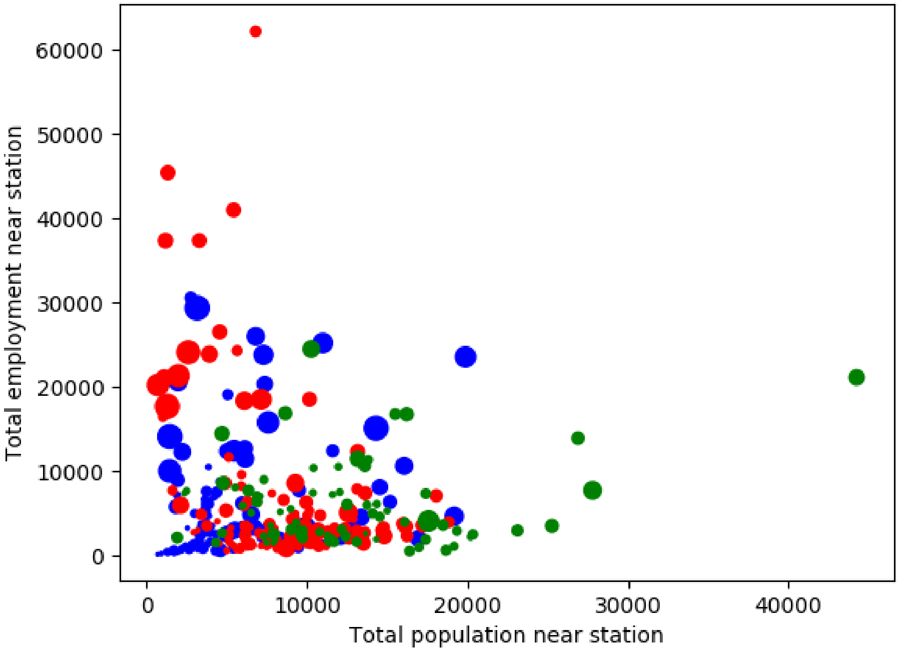
For each station A and for the set of all stations (S) within 15 minutes of A; the counts of each 'Need Index' type feature is summed over all stations of S. This is the count used for the corresponding network type feature of A. The same procedure is repeated for all stations within 30 minutes of A. Thus, for each 'Need Index' type feature associated with a station in the feature set, there are two additional features: one summing the counts of that feature within 15 minutes and one within 30 minutes.

These features are important for providing a measure of centrality to the network. Stations near the center of the network and at transfer points between lines will have higher counts of network features than peripheral stations. The other important function of the network features is to provide estimates of the total scale of system ridership. The more people, jobs, and other characteristics near transit stations, the higher the overall system ridership is expected to be. This is a key component of the model's portability between different city's transit networks.

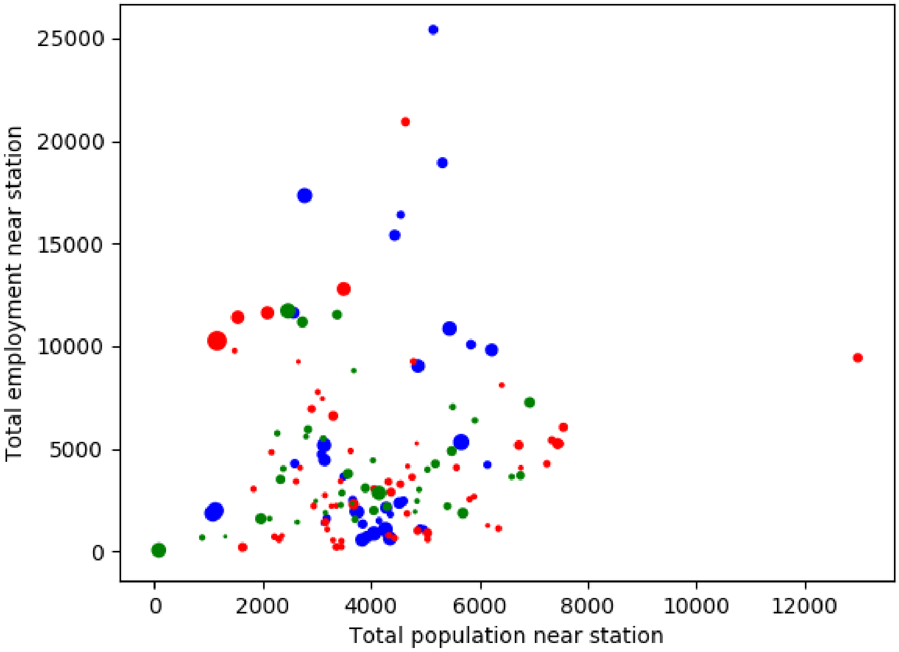
2.7 Analysis of zip code characteristics by city

The purpose of including multiple transit networks with varying levels of ridership is to ensure that the model captures a broad range of relationships between employment, population and other zip code characteristics on the one hand, and transit ridership on the other. Figure 5 shows the employment and ridership counts within walking distance assigned to each station on the transit networks. The ridership of each stations is proportional to the area of the marker for each station. We can see that there is a wide range of population and employment figures near transit stations. In our selection of six transit networks, some networks have much higher peaks of certain zip code characteristics than others, so the graphic is divided into two subfigures, one for the higher ridership networks and one for the lower.

Station remoteness has a significant impact on the characteristics of stations. For example, the five highest population transit stations are all in Los Angeles. It is not true that the densest areas of Los Angeles have a higher population density than Boston or Chicago. Rather, the Los Angeles transit network has lower station density than Boston or Chicago in the areas of highest population density. Therefore, some stations in high population density areas of Los Angeles will have a transit catchment of up to two square kilometers. For



(a) Boston - Blue; Chicago - Red; Los Angeles - Green



(b) Atlanta - Blue; Dallas - Red; Denver - Green

Figure 5: Employment against population with a 15 minute transit ride for sets of three transit networks. Area of marker represents ridership of the station

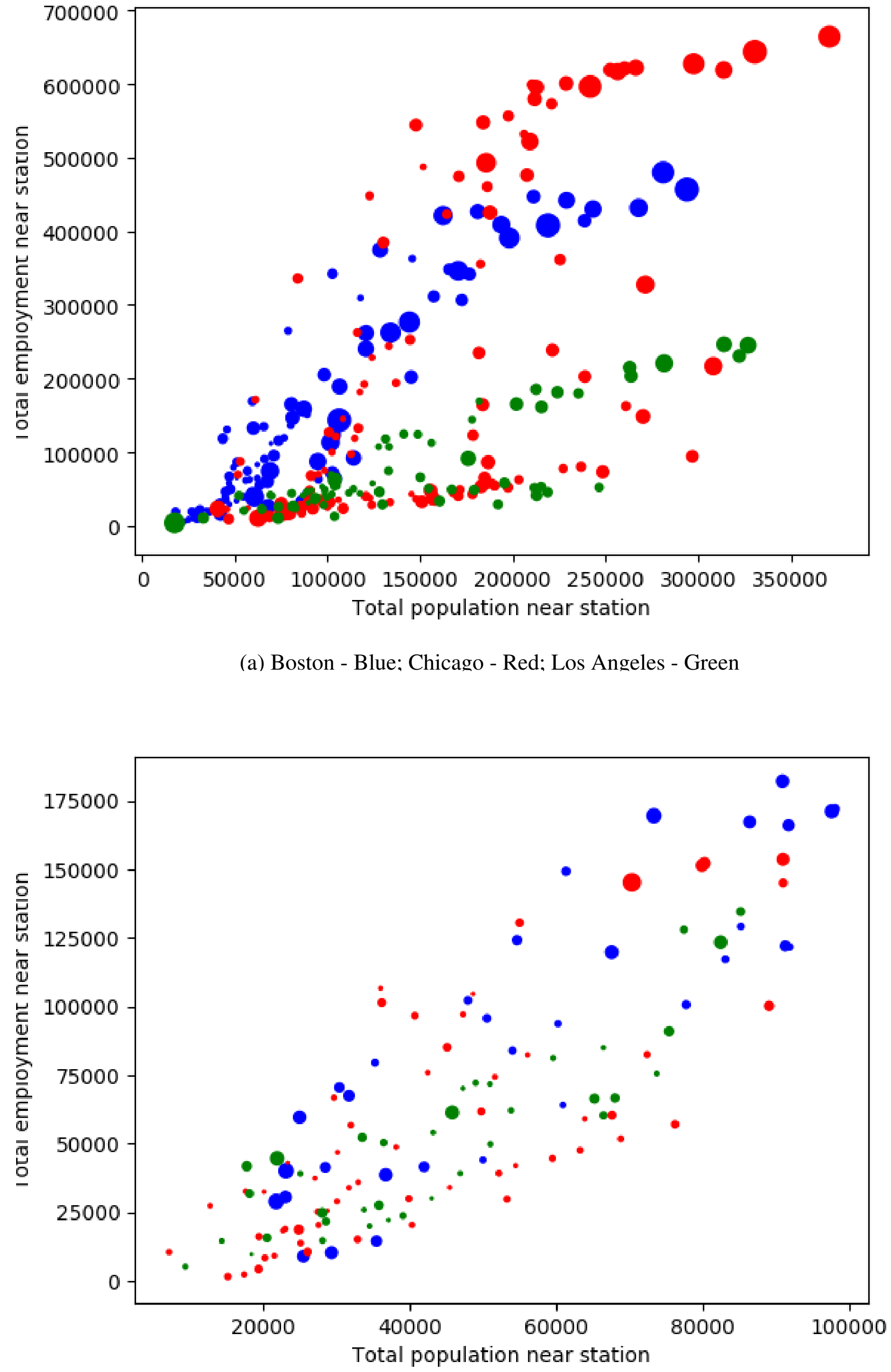
Network City	Total Population	Total Employment	Network length (km)	Avg Weekday Ridership
Chicago	1189454	872206	157	608472
Boston	622484	656375	98	591823
Los Angeles	935041	436846	169	254183
Atlanta	150171	207601	77	150237
Dallas	255092	261813	150	96069
Denver	161542	170631	149	75128

Table 2: Network wide characteristics for the six transit networks considered in this thesis

Boston or Chicago, with closer station spacing and multiple lines, any point in the areas of highest population density will have four or more stations within walking distance.

The overall characteristics of each transit network are found in Table 2. There is a positive linear relationship between both population and employment and ridership at the network-wide level. However, the effect of network length is also significant. For example, Atlanta's population and employment near its transit stations are both lower than that of Dallas, but Atlanta's ridership is fifty percent larger. Atlanta's network length is about half that of Dallas, suggesting that Atlanta's transit accessible population and employment are compressed into a much smaller area. This density appears to be a significant factor in Atlanta's higher ridership. However, looking at Figure 5(b), where Atlanta's stations are in the blue and Dallas' in red, it is not clear that Atlanta's stations have any individual advantage in population and employment.

This shows the necessity of the network-dependent features. In Figure 6, we see the sum of population and employment within a 15 minute transit rider of each station, instead of within walking distance of each station. In Figure 6(b) we see that Atlanta's smaller network and more tightly spaced stations means that stations generally have more other stations within a given travel distance. Therefore, many of Atlanta's stations have higher network population and employment counts than their corresponding stations in Dallas. The inclusion of the network features will allow the regression model to more accurately predict the higher ridership for stations in Atlanta.



(a) Boston - Blue; Chicago - Red; Los Angeles - Green

(b) Atlanta - Blue; Dallas - Red; Denver - Green

Figure 6: Employment against population with walking distance for sets of three transit networks. Area of marker represents ridership of the station.

3 Model generation and Error Metrics

3.1 Metrics for assessing accuracy of predictions

In general, the projected ridership forecasts of new transit infrastructure investments significantly overestimates transit ridership. The pioneering study in this field by Pickrell in 1989 [17] found that for ten rail projects completed between 1977 and 1985, and assessed between 1986 and 1989, the actual ridership was between 28% and 85% lower than projected. A 2006 study [18] of 25 major passenger rail projects in 14 nations found that 21 of the projects had actual ridership below projections, with the average true system ridership 48% below the projection. Accurate assessment of projection accuracy is imperative for creating ridership estimates that serve the public interest.

To assess the predictions from regression analysis, there should be two metrics used: one for the total system ridership, and another for station level ridership. The economic justification for new construction of a transit system is based on projections of total system ridership. The projected cost of operating the system must be proportional to the projected revenue from ridership. The placement of stations should be guided by an evaluation of station level ridership. Decisions on whether to fund infill stations along existing transit lines, or line extensions to previously unserved areas depend on analysis of ridership at the station level. Therefore, it is important to create a ridership model that optimizes both system and station level ridership.

Following Pickrell, the metric for system-wide projection accuracy is standard percentage error in total system ridership, which we will refer to as system error. This error is

$$E_{system} = \frac{\left| \sum\limits_{i \in \text{stations}} y_{i,proj} - \sum\limits_{i \in \text{stations}} y_{i,true} \right|}{\sum\limits_{i \in \text{stations}} y_{i,true}},$$

where y_{proj} is the projected ridership and y_{true} the true ridership for each station, and each is summed over all stations in a transit network.

Hardy [19] extends Pickrell's analysis by including absolute station error for stations on newly added sections of an existing transit network. Following Hardy, a measure of station level error on a network is the summed absolute error of all station projections. The rail networks in this study vary widely in total ridership; therefore, to allow network-to-network comparison, this summed absolute station error can divided by total system ridership. The resulting metric for station error given a projected (y_{proj}) and actual ridership

(y_{true}) is

$$E_{station} = \frac{\sum_{i \in \text{stations}} \left| y_{i,proj} - y_{i,true} \right|}{\sum_{i \in \text{stations}} y_{i,true}}.$$

We refer to this metric as station error.

3.2 Data distribution and regression selection

Regression models for station level ridership have used ordinary least squares regression [9, 10, 11, 12, 13] to generate predictions. We investigate the applicability of more complex regression models, and whether or not these other regression types can provide better modeling accuracy.

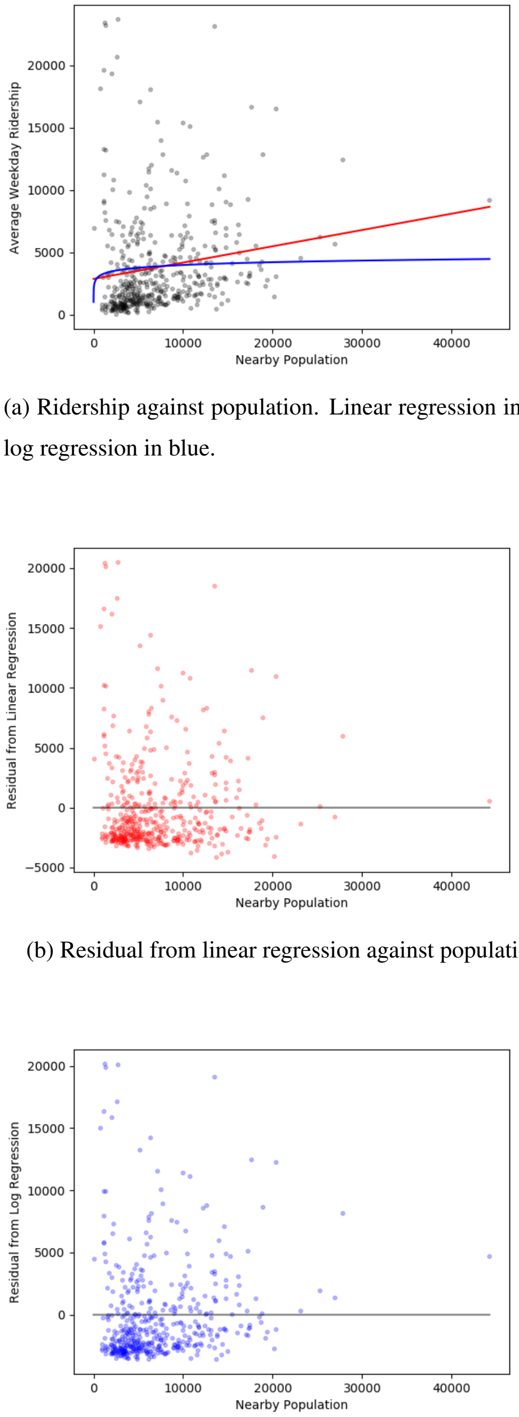
The dependent variable, average weekday ridership at a station, has range $[0, \infty)$. The exponential function has this range and so it is reasonable to assume that there may be a logarithmic relationship between characteristics and ridership. We can compare a linear least squares line of best fit with a line of best fit using the logarithm of the ridership as the response variable.

There are generally two types of feature included in this survey: those that are related to population and those that are related to employment. A plot of all stations' population versus ridership appears in Figure 7 with linear and logarithmic regression lines and residuals. A similar treatment for employment versus ridership appears in Figure 8. In Figures 7(a) and 8(a), a linear and logarithmic best fit line are fitted to the data. The residuals from the linear regression line are plotted in (b) and from the logarithmic regression in (c) of each figure.

The coefficient of determination for the two regression types for both population and employment are shown in Table 3. Judging from the coefficients of determinations and the distributions of the residuals, there does not appear to be any clear modeling advantage from using either a linear or log transformed response variable.

Variable	Linear R^2	Log R^2
Population	0.0267	0.0034
Employment	0.1948	0.1878

Table 3: Regression of Population and Employment against ridership

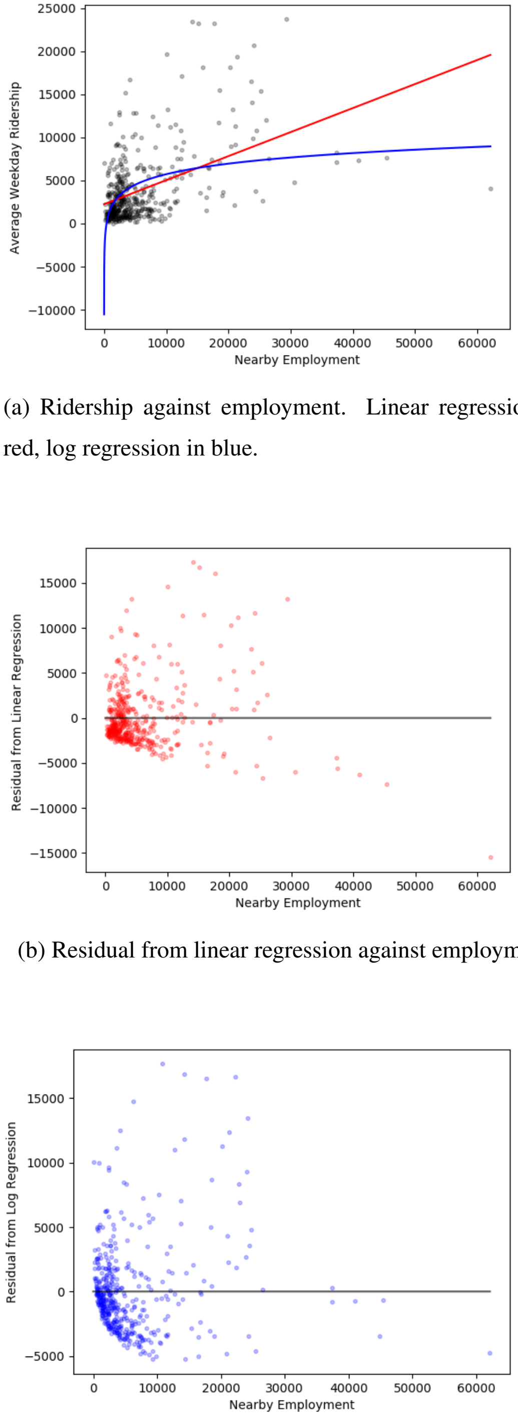


(a) Ridership against population. Linear regression in red, log regression in blue.

(b) Residual from linear regression against population

(c) Residual from log regression against population

Figure 7: Analysis of population regression



(a) Ridership against employment. Linear regression in red, log regression in blue.

(b) Residual from linear regression against employment

(c) Residual from log regression against employment

Figure 8: Analysis of employment regression

The metrics for projection accuracy introduced in Section 3.1 depend on the absolute difference between actual and predicted ridership. This suggests that regression using the Least Absolute Deviation (LAD) (3.4.4) loss function is appropriate for this problem. Since the response variable (ridership) has a value of integer counts, and the variance of ridership increases with increasing employment, we will also investigate the use of a Poisson model. Finally, we will use ordinary least squares (OLS) regression as a baseline comparison, to see if the other methods have any performance advantage compared to the currently used regression type. To test a variety of transformed relationships between the features and response variable, we will test the Poisson regression with both the log and identity link functions, and least squares regression with both a linear response variable and a log-transformed response variable.

Upon performing any regression analysis, it becomes immediately apparent that one set of features is different from the others. Estimating station ridership using only the number of post-secondary (college-level) students within a 15 or 30 minute transit trip of each station provides a very accurate measurement of system level ridership. These variables are represented in the model as `15net_students` and `30net_students`, respectively. We show the results of a single variable OLS regression of the one variable against ridership. The scores are the average of the six way cross-validation across the six transit networks in the study; each of the six networks is used as the test set in one case, while the other five transit networks are used as the training set. The ‘best’ scoring other feature (`15net_hunits_old`; the number of housing units built before 1940 within a 15 minute transit ride) is shown for comparison.

Variable	System Error	Station Error
30net_students	0.1016	0.5961
15net_students	0.0946	0.6197
15net_hunits_old	0.2854	0.6700

Table 4: Error for single variable OLS for selected features

The system error scores for `30net_students` and `15net_students` are much lower than for any other variable, while the station error for these features are also lower than any other features. As we will see, the single-feature OLS of either of these features produces a model that is approximately as good as any other model we will develop. This raises questions about the relationship between this feature and the response variable. It is possible that the population of students within walking distance of a transit station is driven by the availability of local transit, and not the other way around; that is, students may choose their housing locations based on availability of transit. In that case, the number of students is not a valid explanatory variable. Since the relationship is unclear and these features are outliers, we will remove all features derived from number of post-secondary students from the model.

There are 94 remaining features, while some transit networks have as few as 38 stations. Feature selection is necessary to prevent the model from being over-specified. We use two methods of feature selection for each of the five regression types: LASSO regression described in Section 3.3.1 and a stepwise forward selection method described in Section 3.3.2.

3.3 Feature Selection

3.3.1 Feature selection by LASSO regularization

Least Absolute Shrinkage and Selection Operator (LASSO) regression can be used to perform feature selection for regression analysis. The LASSO method adds a regularization term to an objective function to penalize regression coefficients. By forcing the sum of the absolute values of all regression coefficients to be less than a fixed value, some regression coefficients are set to zero, thereby eliminating them from the model.

Because LASSO constrains the magnitude of the coefficients, it is important that all coefficients are on the same scale. Therefore, for all LASSO feature selection methods, the features are normalized by z-score standardization. Each entry $x_{i,j}$ in the covariate matrix is scaled by

$$x_{i,j,scaled} = \frac{x_{i,j} - \mu_i}{\sigma_i}$$

where μ_i and σ_i are the mean and standard deviation of the i-th column of the covariate matrix.

An important part of the LASSO solution is selection of λ, the penalty coefficient. The derivation of the LASSO optimization equations are described in each subsecetion of Section 3.4. The penalty in a LASSO regression is divided into two parts: a loss function on the error of the predicted values and a penalty on the regression coefficients. For a sufficiently large value of λ, the optimum solution is for all regression coefficients to be zero, thus eliminating the regression coefficient penalty. Solving for the optimum λ thus requires identifying the minimum λ value for which the solution is all zero coefficients, a solution with zero degrees of freedom. This is typically done by a grid search; this is the method used by the `glmnet` package [20]. From `glmnet`, we use the cross-validation methods that will automatically optimize λ. The minimum value of λ where the solution has zero degrees of freedom is identified by grid search, and then cyclical coordinate descent is used to find the optimum λ value.

LAD LASSO is performed using the `flare` package in R; this package does not have a built in cross-validation algorithm to optimize λ. Instead, the author created a λ grid search algorithm comparable to that performed by `glmnet`. The Poisson LASSO with identity link is performed by author supplied code, where λ is solved concurrently with the regression parameters (θ) by an interior point solver, as detailed in section 3.4.2.

LASSO regression is performed six times for each LASSO regression type. A trial is made with each city as the test set and all other cities as the training set. Therefore, we identify six sets of selected features for each LASSO regression type. To choose a single set of features from these six sets, we choose all features that are selected by four or more of the six cross-validation trials. This is the feature set that will be reported in the results section.

3.3.2 Feature selection by stepwise forward selection

Since the data set for this problem is small–with only 466 total stations–we can validate the LASSO results using a stepwise forward selection approach. This method is a greedy search of all possible features to find the best combination of features that minimizes our error metrics, performed in a multi-step process. This process is performed for each of the five regression types defined in Table 5.

In the first step, for all features, we perform a single-covariate regression against the response variable: ridership. Each feature is evaluated in six-way cross validation, where each of the six transit networks is used as the test set once, while the other five are used for the training. Since there are two types of error defined in Section 3.1, station and system error, we use the arithmetic mean of the two error metrics to score each feature. We then take the arithmetic mean of the score from all six trial runs, and select the single feature with the lowest error score. If k represents one of the six systems that is evaluated as the test set, then the error score for each feature is

$$E_{feature} = \frac{\sum\limits_{k \in 6\ systems} \dfrac{E_{k,system} + E_{k,station}}{2}}{6}.$$

After choosing a feature in the first step, in the second step we perform a two covariate regression against ridership. The feature selected in the first round is used, and we test all other features as the second covariate. The second feature that yields the lowest error score is then selected. The subsequent steps continue with multiple regression using all of the already chosen features.

Algorithm 3 Algorithm for choosing variables by stepwise forward selection

```
Let $F$ be set of all features
$R = \emptyset$
for $i \leftarrow 1$ to 25 do
    for $f \in F$ do
        Predict test set using regression with features $f \bigcup R$ of the training set
        Calculate error score for $f$
    $r = \min(f)$; $r$ is feature $f \subset F$ with lowest error score
    $R = R \bigcup r$
    $F = F - r$
```

For each regression type, we perform 25 steps of forward selection, identifying 25 features, in order, according to Algorithm 3. We then plot the error scores against number of features to find number of features that yields the lowest error score; and example of these graphs is Figure 9. The set of features that produces the lowest error scores for this regression type is included in the results in Table 6.

3.4 Derivation of Regression methods

For the five regression types described, the methodology and packages used are described below. A chart of packages used for implementation is presented in Table 5.

Regression Method	Type	package
Least Squares	Linear	LASSO: `glmnet` in Python
		Forward Selection: `statsmodels` in Python
	Log Transform	LASSO: `glmnet` in Python
		Forward Selection: `statsmodels` in Python
LAD	Linear	LASSO: `flare` in R
		Forward Selection: `statsmodels` in Python
Poisson	Log Link	LASSO: `glmnet` in Python
		Forward Selection: `statsmodels` in Python
	Identity Link	LASSO: Author-created in Python
		Forward Selection: `statsmodels` in Python

Table 5: Regression types and packages used in analysis.

3.4.1 Least squares regression

Ordinary Least Squares (OLS) regression is a model where the ith of m response variables y_i is a linear function of the regressors,

$$y_i = x_i^\top \theta + \epsilon_i$$

where θ is a parameterization. The error component ϵ_i is assumed to be normally distributed. The expected value of any element of Y is the corresponding element of $X^\top \theta$:

$$E(y_i \,|\, x_i; \theta) = x_i^\top \theta$$

Here, y_i is one of m response variables, $\theta \in \mathbb{R}^p$ is a length p vector of parameters and x_i is one of m length p vectors of covariates. An intercept is implemented by pre-pending a feature with constant value 1 to each vector x_i. For this model, the $p = 95$ parameters correspond to the 94 implemented features and one intercept column. The response variables are the ridership of each individual station, so m is the number of stations included in any model fitting.

For each response variable y_i, the residual is $y_i - X^\top \theta$. For OLS, the measure of best fit is the sum of squared residuals, so we optimize the parameters by

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^p}{\arg\min} \sum_{i=1}^{m} \left(y_i - x_i^\top \theta \right)^2$$

to obtain a least squares best fit solution $\hat{\theta}$.

For LASSO regularization [21], an additional constraint on the parameters is introduced to limit the absolute magnitude of the sum of the parameters:

$$\sum_{j=1}^{p} |\theta_j| \leq t.$$

This has the practical effect of selecting only a subset of the provided features. We apply this constraint to OLS to solve for parameters such that

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^p}{\arg\min} \frac{1}{m} \sum_{i=1}^{m} \left(y_i - x_i^\top \theta \right)^2 + \sum_{j=1}^{p} \lambda \left| \theta_j \right|.$$

The tuning parameter λ is proportional to the strength of the regularization penalty. The larger the λ, the fewer features will be selected. The LASSO solver used for both Least Squares and Poisson in this study is the `glmnet` package. This package identifies the optimal λ by dividing the data set for cross-validation, then using a grid search and cyclical coordinate descent to identify the λ associated with minimum error.

For the least squares regression using the log transform, we transform the dependent variable Y to $\log Y$ and perform the same OLS or LASSO regression.

3.4.2 Poisson regression

Poisson regression is case of the generalized linear model (GLM) [22]. GLM generalizes linear regression by relating the expected value of the response variable to the linear predictor through a link function.

$$E(\mathbf{Y}) = g^{-1}(\mathbf{X}\beta)$$

The distribution of the response variable follows a probability distribution from the exponential family. For Poisson regression, the variance follows a Poisson distribution. We perform Poisson regression with both its canonical log link and identity link. For the log link, the mean of the predicted Poisson distribution is given by

$$E(y_i \mid x_i; \theta) = e^{x_i^\top \theta},$$

where x, y, and θ are as defined in the last section. Using this mean as the parameter of a Poisson probability mass function, the joint distribution is

$$p(y_1, ..., y_m | x_1, ..., x_m; \theta) = \prod_{i=1}^{m} \frac{e^{y_i x_i \theta} e^{-e^{x_i \theta}}}{y_i!}.$$

This represents the probability of any set of ridership numbers given a feature set and parameterization. The optimal parameterization is obtained using the negative log-likelihood expression

$$-\mathcal{L}(\theta | X, Y) = \sum_{i=1}^{m} e^{x_i^\top \theta} - y_i x_i^\top \theta.$$

In this expression, we ignore the constant factorial term which falls out in differentiation. This convex function is the objective function for our optimization problem and is minimized over $\theta \in \mathbb{R}^p$.

For LASSO regression, we minimize the parameters [23] over

$$\hat{\theta}(\lambda) = \underset{\theta \in \mathbb{R}^p}{\arg\min} \frac{1}{m} \sum_{i=1}^{m} e^{x_i^\top \theta} - y_i x_i^\top \theta + \sum_{j=1}^{p} \lambda \left| \theta_j \right|.$$

3.4.3 Poisson regression with Identity Link

For the identity link, the mean of the predicted Poisson distribution is given by

$$E(y_i \,|\, x_i; \theta) = x_i^\top \theta$$

which yields a joint distribution

$$p(y_1, ..., y_m | x_1, ..., x_m; \theta) = \prod_{i=1}^{m} \frac{\left(x_i^\top \theta\right)^{y_i} e^{-x_i^\top \theta}}{y_i!}.$$

The negative log likelihood of this distribution is the objective function

$$-\mathcal{L}(\theta \,|\, X, Y) = \sum_{i=1}^{m} -y_i \log\left(x_i^\top \theta\right) + x_i^\top \theta. \tag{1}$$

This function is convex and so the minimum can be obtained using convex optimization methods.

With no suitable package available to perform LASSO regression using Poisson regression and the identity link, we implement a solution using Python. We use the primal-dual interior point method as outlined in Boyd and Vandenberghe [24]. We will set up a problem of the form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, ..., m. \end{array}$$

There are no equality constraints of the form $Ax = b$ in this problem setup; there are only inequality constraints related to LASSO.

The function to be minimized, f_0, is the negative log likelihood in Equation 1. The inequality constraint for LASSO is that the ℓ_1 norm of the parameter vector is less than some constant t, $||\boldsymbol{\theta}||_1 < t$, or $||\boldsymbol{\theta}||_1 - t \leq 0$. The ℓ_1 norm is the sum of the absolute values of the parameters. Both f_0 and f_1 must be convex and twice continuously differentiable, but the LASSO constraint is a non-differentiable function. Therefore we split the absolute value constraint into two parts,

$$\theta_j = \theta_j^+ - \theta_j^- \tag{2}$$

where

$$\theta_j^+ = \begin{cases} \theta_j, & \theta_j > 0 \\ 0, & \theta_j \leq 0 \end{cases} \qquad \qquad \theta_j^- = \begin{cases} \theta_j, & \theta_j < 0 \\ 0, & \theta_j \geq 0 \end{cases} \tag{3}$$

so that

$$|\theta_j| = \theta_j^+ + \theta_j^-.$$

There are p entries in $\boldsymbol{\theta}$; and, for each entry, both terms from Equation (2) must be non-negative. Therefore, the LASSO constraint is expanded to $2p$ separate terms that must all be non-negative. All the terms associated with either θ_j^+ or θ_j^- are grouped into two separate constraint equations, f_1 or f_2, respectively.

Therefore, we choose f_0, f_1, and f_2 with the additional constraint that the parameters must be non-negative for f_1 to be equivalent to the LASSO constraint. The selected functions, plus their derivatives and gradients are:

$$f_0(\theta) = \sum_{i=1}^{m} -y_i \log\left(x_i^\top \theta\right) + x_i\theta \quad \nabla f_0(\theta) = \sum_{i=1}^{m} \begin{bmatrix} \dfrac{x_i\left(-y_i + x_i^\top \theta\right)}{x_i^\top \theta} \\ \dfrac{x_i\left(y_i + x_i^\top \theta\right)}{x_i^\top \theta} \end{bmatrix} \quad \nabla^2 f_0(\theta) = \sum_{i=1}^{m} \begin{bmatrix} \dfrac{y_i x_i^2}{\left(x_i^\top \theta\right)^2} & \dfrac{-y_i x_i^2}{\left(x_i^\top \theta\right)^2} \\ \dfrac{-y_i x_i^2}{\left(x_i^\top \theta\right)^2} & \dfrac{y_i x_i^2}{\left(x_i^\top \theta\right)^2} \end{bmatrix} \tag{4}$$

$$f_1(\theta^+) = -\,\theta_j^+ \quad \forall j \in \{1...p\} \qquad \nabla f_1(\theta^+) = \begin{bmatrix} -1 & \dots & -1 \\ 0 & \dots & 0 \end{bmatrix} \qquad \nabla^2 f_1(\theta^+) = 0 \tag{5}$$

$$f_2(\theta^-) = -\,\theta_j^- \quad \forall j \in \{1...p\} \qquad \nabla f_2(\theta^-) = \begin{bmatrix} 0 & \dots & 0 \\ -1 & \dots & -1 \end{bmatrix} \qquad \nabla^2 f_2(\theta^-) = 0 \tag{6}$$

The constraints f_1 and $_2$ are such that all elements in the vectors $\boldsymbol{\theta}^+$ and $\boldsymbol{\theta}^-$ must be less than or equal to zero. Note that $\theta_j = \theta_j^+ - \theta_j^-$ is retained in the definitions of f_0. The constraints f_1 and f_2 combine to

$$-\theta_j^+ - \theta_j^- = -|\theta_j|$$

which ensures that the LASSO constraint is also met by the non-negativity requirement of f_1 and f_2.

We establish the modified Karush-Kuhn-Tucker (KKT) equations for solving this problem using a Newton method. There is no equality constraint, which takes the form $Ax = b$ in the usual KKT conditions; therefore these terms are not present in our equations. Given that there are p features and m data points in our problem, then $f_0, f_1, f_2 : \mathbb{R}^p \to \mathbb{R}$.

To solve the multiple constraints, we define

$$f(\theta^+, \theta^-) = \begin{bmatrix} f_1(\theta^+) \\ f_2(\theta^-) \end{bmatrix} = \begin{bmatrix} \theta_1^+ \\ \vdots \\ \theta_p^+ \\ \theta_1^- \\ \vdots \\ \theta_p^- \end{bmatrix}, \qquad D\, f(\theta^+, \theta^-) = \begin{bmatrix} \nabla f_1(\theta^+)^\top \\ \nabla f_2(\theta^-)^\top \end{bmatrix} = \begin{bmatrix} -1 & \dots & -1 & 0 & \dots & 0 \\ 0 & \dots & 0 & -1 & \dots & -1 \end{bmatrix}$$

The Newton steps for solving the modified KKT equations are given by

$$r_\gamma(\theta, \lambda) = \begin{bmatrix} \nabla f_0(\theta^+, \theta^-) + \lambda D\, f(\theta^+, \theta^-) \\ -\text{diag}(\lambda) f(\theta^+, \theta^-) - \frac{1}{\gamma}\mathbb{I} \end{bmatrix}.$$

For a current point $y = (\theta, \lambda)$, the next Newton step will be $\Delta y = (\Delta\theta, \Delta\lambda)$. The Newton step is characterized by the linear equation

$$r_\gamma(y + \Delta y) \approx r_t(y) + \nabla r_t(y) \Delta y = 0$$

or

$$\begin{bmatrix} \nabla^2 f_0(\theta^+, \theta^-) + \lambda \nabla^2 f_1(\theta^+) + \lambda \nabla^2 f_2(\theta^-) & D\, f(\theta^+, \theta^-)^\top \\ -\text{diag}(\lambda) D\, f(\theta^+, \theta^-) & -\text{diag}\left(f(\theta^+, \theta^-)\right) \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\lambda \end{bmatrix} = - \begin{bmatrix} r_{dual} \\ r_{cent} \end{bmatrix}.$$

We retain the objective function f_0 and its gradient and Hessian as defined in (4) but we plug in values for the constraint functions from (5) and (6) to get a set of two linear equations. The identity of θ allows us to eliminate θ^+ and θ^-.

$$\nabla^2 f_0(\theta) \Delta\theta - \Delta\lambda = -r_{dual} \tag{7}$$

$$\text{diag}(\lambda) \Delta\theta + \text{diag}(\theta) \Delta\lambda = -r_{cent} \tag{8}$$

We solve (8) for $\Delta\lambda$ in terms of $\Delta\theta$,

$$\Delta\lambda = \frac{-r_{cent} - \text{diag}(\lambda)\Delta\theta}{\theta}, \tag{9}$$

and plug into (7) to get

$$\nabla^2 f_0(\theta) \Delta\theta + \frac{r_{cent}}{\theta} + \frac{\text{diag}(\lambda)\Delta\theta}{\theta} = -r_{dual}$$

$$\left(\nabla^2 f_0(\theta) + \frac{\text{diag}(\lambda)}{\theta} \right) \Delta\theta = -r_{dual} - \frac{r_{cent}}{\theta}. \tag{10}$$

Algorithm 4 Algorithm for solving interior point primal-dual problem

```
Given $\epsilon > 0$

while $||r_{dual}||_2 < \epsilon$ and $\eta \leq \epsilon$ do
    $\gamma \leftarrow p\mu/\eta$
    Compute $\Delta\theta$ and $\Delta\lambda$ using (9) and (10).
    Determine step length $s > 0$
    $\theta \leftarrow \theta + s\Delta\theta$
    $\lambda \leftarrow \lambda + s\Delta\lambda$
```

The value of $\Delta\theta$ is solved from the linear equation in (10) with r_{dual} and r_{cent} as defined in (7) and (8), respectively.

Minimization over θ and λ is obtained by by iteration, as shown in Algorithm 4. The duality gap, the difference between the primal and dual solutions, is not necessarily feasible during each iteration of the interior point method. Therefore, a surrogate duality gap is $\eta(\theta, \lambda) = -f_1(\theta)^\top \lambda = \theta\lambda$. A constant scaling constant for γ is μ; p is the number of features in the covariate matrix, or the length of θ and λ. During each iteration, we update $\Delta\theta$ in accordance with (10) and $\Delta\lambda$ in accordance with (9). The iterations repeat until both the surrogate duality gap and the dual residual are below a certain threshold (ϵ).

3.4.4 Least Absolute Deviations Regression

Least absolute deviations regression is similar to OLS (defined in Section 3.4.1), except that instead of minimizing the sum of squared residuals loss function, it minimizes the sum of the absolute value of the residuals

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^p}{\arg\min} \sum_{i=1}^{m} |\, y_i - \theta x_i \,| \,.$$

We could add a weight factor to LAD regression so that the optimization takes the form

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^p}{\arg\min} \sum_{i=1}^{m} w_0 \,|\, y_i - \theta x_i \,| \,.$$

if we choose $w_0 = 1/y_i$, then the LAD loss function is the same as the station error ($E_{station}$) defined in Section 3.1:

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^p}{\arg\min} \frac{\sum\limits_{i=1}^{m} |\, y_i - \theta x_i \,|}{\sum\limits_{i=1}^{m} y_i}.$$

Therefore, this regression should minimize one of the two error metrics we have defined for this model.

For LASSO regularization, the the constraint on parameter absolute magnitude is included to get

$$\hat{\theta} = \underset{\theta\in\mathbb{R}^p}{\arg\min} \sum_{i=1}^{m} w_0 \, | \, y_i - \theta x_i \, | + \sum_{j=1}^{p} \lambda \left|\theta_j\right| ,$$

where the weight factor is the same as above.

4 Results

Regression Type	Feat Select Type	Min System Err	Min Station Err	# Features Selected	Avg System Err	Avg Station Err
LstSq - Linear	Pop and Emp only	0.0451	0.5324	2	0.5172	0.8307
LstSq - Linear	Best Single Feat	0.0465	0.4731	1	0.2852	0.6700
LstSq - Linear	Random Forest	0.1181	0.4161	5	0.3214	0.6439
LstSq - Linear	Random Forest	0.2146	0.4545	10	0.3900	0.7205
LstSq - Linear	LASSO	0.0152	0.4717	9	0.3455	0.7057
LstSq - Log Trans	LASSO	0.0404	0.4501	19	0.4041	0.6773
Poisson - Log Link	LASSO	0.0892	0.4753	10	0.4857	0.7731
Poisson - Identity	LASSO	0.0721	0.4516	10	0.4214	0.7142
LAD	LASSO	0.0241	0.6417	12	0.4443	0.7061
LstSq - Linear	Stepwise	0.0216	0.4461	12	0.1919	0.6197
LstSq - Log Trans	Stepwise	0.0450	0.4997	16	0.1741	0.5789
Poisson - Log Link	Stepwise	0.0426	0.4861	15	0.2184	0.6218
Poisson - Identity	Stepwise	0.0009	0.4113	8	0.1936	0.6071
LAD	Stepwise	0.0005	0.3986	4	0.1792	0.5610

Table 6: Results of regression analysis, compared with some baseline measures

4.1 Results table and baseline comparisons

Table 6 shows the results of all the tested regression models. For each model, there are five result columns. Each model selects a set of features; feature selection by the LASSO method is described in section 3.3.1 while feature selection by forward selection is described in 3.3.2. For each of the six transit systems, a model of the appropriate regression type is made from the other five transit systems. The model uses the selected feature set and we calculate the system and station errors. Reported in the first two columns are the minimum system and station error for any of the six models. The last two columns are the average of the system and station errors across all six models.

In the top block of Table 6, we include a series of baseline comparison measures. The first is a simple, two-variable OLS regression taking the two variables that are most obviously relevant to transit ridership: population within walking distance, and employment within walking distance. The second baseline comparison is the 'best' single feature that we selected from our pool of 94 available features. The single best feature is the number of housing units built before 1940 within a 15 minute transit ride of the station of interest.

The third and fourth baseline comparisons use features selected by random forest regression [25]. A random forest model is implemented using the python programming language's `scikit-learn` package. The feature matrix is sampled randomly and distributed independently into a forest of 10 trees. Using least squares error, the random forest model predicts the response variables. However, we are interested in feature selection. The 'Gini importance' metric, which is the decreases in Gini impunity criterion for each feature summed over the entire forest [26], is used to score each feature by its relative importance.

We run 100 iterations of the random forest model, and sum the feature scores for each run. The top five and ten selected features are analyzed in the results table. The random forest feature sets outperform the population and employment based model, but does not outperform the best single feature.

An analysis of the accuracy of the methods shown in Table 6 is included in Section 5.

4.2 LASSO regularization results

The various LASSO regression types have a large variance in number of features chosen, both between transit networks and between regression types, shown in Table 7. Since the six way cross validation produces a different set of selected features for each transit network, we must develop a means for determining a 'best' feature set for a common comparison. To represent the results of the LASSO regression, all features that are selected by four or more of the six cross-validation trials are used.

Regression Type	Atlanta	Boston	Chicago	Dallas	Denver	Los Angeles	Average	Selected
LstSq - Linear	10	2	8	9	10	17	9.3	9
LstSq - Log Transform	26	12	17	22	28	46	25.2	19
Poisson - Log Link	11	5	12	26	27	29	18.3	10
Poisson - Identity Link	12	15	13	12	11	12	12.5	10
LAD	0	26	15	27	49	49	27.7	12

Table 7: Number of features selected by LASSO; by transit network and regression type

The LAD LASSO is unusual in its high variability between different trial runs. On average, the six trials selected 28 features, compared to only 12 features that were selected by four different trials. On the opposite side of the spectrum, linear least squares LASSO selected an average of 9 features per trial, and 9 features were selected by four or more trials. For linear least squares, the different trials substantially agreed when selecting variables.

Another way to compare the results of the different trials and methods is to measure the sparsity of the coefficients. Sparsity is a measure of the 'empty space' in the paramaterization of the model. We choose a non-normalized variant of the Hoyer measure for sparseness [27] as our metric:

$$\text{sparseness}(\beta) = \left(\frac{||\beta||_1}{||\beta||_2}\right)^2.$$

The Hoyer measure is proven to meet the intuitive attributes of a sparsity measure [28], but is normalized on the range $[0, 1]$. We desire our variant to have a range related to the number of features selected, to provide additional information about the quality of LASSO feature selection. For a vector with minimum sparseness, $\beta_{min} = [a\,0\,0\,...]$,

$$\text{sparseness}(\beta_{min}) = \left(\frac{|a|}{\sqrt{a^2}}\right)^2 = 1.$$

For a vector with maximum sparseness, $\beta_{max} = [a\,a\,a\,...]$,

$$\text{sparseness}(\beta_{max}) = \left(\frac{k|a|}{\sqrt{ka^2}}\right)^2 = k$$

where k is the length of vector β. The maximum possible sparseness score, for a parameter vector with equal coefficients for all features, would be the number of features. For our application, there are 94 features, so the range of the sparseness metric is $[1, 94]$.

Regression Type	Atlanta	Boston	Chicago	Dallas	Denver	Los Angeles
LstSq - Linear	3.5	1.7	2.8	3.4	4.2	6.0
LstSq - Log Transform	1.6	1.3	1.4	1.6	2.1	7.2
Poisson - Log Link	1.2	1.1	1.2	1.6	1.7	2.3
Poisson - Identity Link	4.1	3.6	4.2	3.8	4.0	4.3
LAD	NaN	12.9	8.5	9.6	15.9	15.7

Table 8: Number of features selected by LASSO; by transit network and regression type

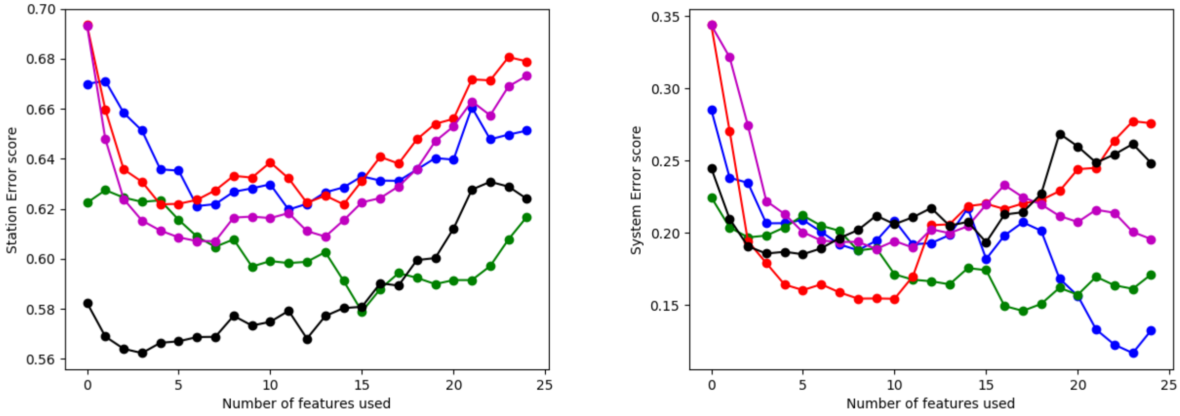
The LAD regression is notable both for the high sparseness of its parameterizations and for the case of Atlanta where no variables are selected. Also notable is that Los Angeles has a higher sparseness than any other city for four of the five regression methods; and has the second highest sparseness for the fifth.

For all regression types, the LASSO method of feature selection performs significantly worse than the forward selection method. The results in Table 6 allow a comparison of the LASSO regression against some baseline measures. None of the LASSO methods outperform the five variable random forest model or the best single variable, in either error metric.

4.3 Forward selection results

Using the method for feature selection described in Section 3.3.2, we select the fist 25 features and graph the resulting system and station error scores in Figure 9. The graphed data point for both system and station error is the average error from using each of the six transit systems as the test set.

In general, each of the error scores decreases with the addition of new features up to a certain number of covariates, and then increases again. The minima for the system and station errors do not coincide with each other for any of the five regression methods, so there is a range of features which produce very similar error scores. By using the arithmetic average of the system and station errors, we pick an optimum number of features for each regression type; this is reported in Table 6.



(a) Station Error against number of variables selected

(b) System Error against number of variables seleted

Figure 9: Error against number of features used for stepwise forward selection

Blue - Least Squares Linear; Green - Least Squares Log; Red - Poisson Log; Magenta - Poisson Identity; Black - LAD

The station error is identical to the Least Absolute Deviation loss function. Therefore, it is not surprising that this regression type performs the best on the Station Error metric, having the lowest station error for up to 15 features. An interesting result of LAD regression is that the station error curve has its minimum much earlier than the other regression types, which show constant decreasing station error for at least the first six features. In the system error metric, the LAD regression loses its advantage. Instead, both the Least Squares regression methods produce low system error scores with increasing numbers of variables.

The minimum scores from each regression type are recorded in Table 6 along with the average score over the range of good features. The stepwise regression significantly outperforms the LASSO regression in creating feature combinations that can accurately predict ridership in unknown transit networks. All the stepwise regression methods outperform all of the baseline methods showing the first block of the results table, in both error metrics.

4.4 Analysis of selected features

A summary of selected features is given in Table 9. There are ten total regression methods used for feature selection; five LASSO methods and five forward selection methods. The count columns in Table 9 shows the number of times each feature was selected by each regression method. For example, the `near_hospitality` feature was selected by all ten methods, five LASSO and five forward selection.

The highly rated features display a mixture of potential correlation and causation with respect to transit ridership. For example, a major university is likely to drive transit ridership both in the immediately adjacent stations, and in nearby areas where students especially will live. On the other hand, the number of hospital jobs (`30net_medical`) within 30 minutes of transit is more likely to be a reflection of population density rather than a driver of transit ridership. Some features, like the ubiquitously selected `near_hospitality` are a mixture of both. Restaurants and hotels are likely to be located in areas of good transit accessibility, while they themselves will drive transit ridership to take advantage of their services.

Because the forward selection models significantly outperformed the LASSO models, the variables chosen by the forward selection models are more likely to useful to a transit model. Conversely, there are several variables that were chosen by nearly all of the LASSO models, such as the number townhouses and duplexes within 15 minutes of a transit station (`15net_hunits_attached`) and the number of small apartment buildings within 15 minutes of a transit station (`15net_hunits_medium`). Since the LASSO methods performed worse than the baseline measures, it is likely that some of these variables actually decrease the performance of a model.

Of the 94 features, 48 were not selected by any of the feature selection methods. Twenty-nine features were selected more than once, and fifteen were selected more than once by the forward selection models.

Feature Name	LASSO	Forward Selection	Total Count
near_hospitality	5	5	10
parking	4	3	7
near_university	4	2	6
15net_hunits_old	2	3	5
30net_medical	3	2	5
15net_university	1	3	4
30net_hunits_large	2	2	4
near_entertainment	2	2	4
15net_hunits_attached	4	0	4
15net_hunits_medium	4	0	4
30net_entertainment	1	2	2
near_hunits_detached	1	2	3
near_hunits_old	1	2	3
near_medical	1	2	3
near_emp_pay	2	1	3
near_employment	2	1	3
near_pop_old	2	1	3

Table 9: Features that were frequently selected by regression methodologies

5 Conclusion and Future Work

Ultimately, the error scores for the forward selection models for all five regression variants in Table 6 are very similar. There is little evidence that one regression type is significantly more effective than any other. All five regression types are able to give system-wide ridership estimates within 20% of the true value for identified sets of features. This error compares favorably with the 48% average ridership prediction error reported for 25 transportation networks in Flyvbjerg [18], where Flyvjerg uses the same system error model identified in Section 3.1.

In addition, this model produces ridership estimates based on information publicly available to citizens.

A ridership model can be trained on public data from cities with transit systems in the United States. A city with no existing intra-urban rail transit system would only need to obtain zip-code level information available from the US Census Bureau's website to apply such a model to generate ridership estimates for a new construction rail system.

The Mannheim-Florian four step model fundamentally depends on existing transit ridership information. For example, the starting point of a four step analysis for a new light rail line would be an existing bus line that runs a similar, hopefully identical, path. The advantage of this regression model is that it creates a new estimate from different sources, independent of any knowledge of the current system.

A model trained on the transit system of the six US cities included in this work, and using any of the five regression methods tested would be sufficiently accurate to justify use for validating new construction, system-wide cost estimates for urban rail transit in the United States.

5.1 Future Work

Future work in on this model could proceed in two directions. The first direction is to continue improvement of the source data. This project used only zip code shapes to estimate counting features, since job and housing data was available only for the zip code, but the population features exist in more granular detail at the Census Tract level. The author generated exclusion zones which were designed to prevent job and people from being located in parks and water could be improved by incorporating detailed city land use maps. Finally, there is a major deficiency in source data because neither federal, state nor local government workers are included in the employment data sources used by the model. Of particular concern are university employment; university jobs within walking distance was a selected feature in five of the six models. Several large universities are located on the transit networks of this study and were not accounted for, such as University of Illinois at Chicago, University of Colorado–Denver, and University of Massachusetts–Boston. There are other cities not included in this study that have very large public universities, like Minneapolis, Austin, or Columbus. Under-representation of university jobs in these cities would negatively affect the validity of any estimates, especially given the importance of university related features in Table 9.

The second direction for future work is improvement of the model itself. The feature summary for this paper only analyzed whether or not a feature was selected by any of the LASSO or forward selection regression methods. There remains to be done an analysis of the magnitude and direction of each feature's coefficient, to ensure that frequently selected features are significant. A feature selected four times, but with coefficient magnitudes both positive and negative is not as significant as a selected feature that only has positive or negative coefficients.

In this work, only ℓ_1, LASSO regularization was used. However, the R package `glmnet` is capable of performing ElasticNet regression as well. ElasticNet [29] is a linear mixture of the ℓ_1 and ℓ_2 penalties from the LASSO and ridge regression methods. For OLS and Poisson, a mixed regularization may be able to improve the performance of the feature selection.

A Data sources

A.1 Ridership data

Los Angeles:	`http://libraryarchives.metro.net/DPGTL/Ridership/RailActivityByStationFY2014.xls`
Chicago:	`http://www.transitchicago.com/assets/1/ridership_reports/2015_Annual.pdf`
Atlanta:	`http://documents.atlantaregional.com/transportation/TFB_2014_v17.pdf`
Boston:	`http://archives.lib.state.ma.us/bitstream/handle/2452/266319/ocm18709282-2014.pdf`
Denver:	`http://www.rtd-denver.com/documents/serviced/lrt-activity-08-2015.pdf` and `http://www.rtd-denver.com/documents/serviced/lrt-activity-Jan-April-2016.pdf`
Dallas:	`https://www.dart.org/about/dartreferencebookmar16.pdf`

A.2 US Census feature data sources

All feature data is accessed through the American Factfinder website at `factfinder.census.gov`.

Population	Table DP05, Item HC01_VC03
Population, 18 and under	Table DP05, Item HC01_VC03 - Item HC01_VC32
Population, 65 and over	Table DP05, Item HC01_VC37
Housholds	Table S1101, Item HC01_EST_VC02
Households with Children	Table S1101, Item HC01_EST_VC06
Families	Table S1101, Item HC01_EST_VC010
Population with at least Bachelors degree	Table S1701, Item HC01_EST_VC34
Population in labor force	Table S1701, Item HC01_EST_VC37
Employed population	Table S1701, Item HC01_EST_VC38
Full-time employed population	Table S1701, Item HC01_EST_VC47
Population living at greater than 500% of poverty level	Table S1701, Item HC01_EST_VC56
Population living at less than 200% of poverty level	Table S1701, Item HC01_EST_VC01 - HC01_EST_VC59
Housing units	Table DP04, Item HC01_VC03
Single-family detached housing units	Table DP04, Item HC01_VC14
Housing units in duplexes or townhouses	Table DP04, Items HC01_VC15 + HC01_VC16
Housing units in structures of 3-9	Table DP04, Item HC01_VC17 + HC01_VC18
Housing units in structures of 10+	Table DP04, Item HC01_VC19 + HC01_VC20
Housing units built before 1940	Table DP04, Item HC01_VC36
Housing units built after 2000	Table DP04, Item HC01_VC27 + HC01_VC28 + HC01_VC29
Housing units occupied by owner	Table DP04, Item HC01_VC65
Housing units occupied by renter	Table DP66
Number of Jobs	Table CB1500CZ11, Item EMP
Total pay of all jobs	Table CB1500CZ11, Item PAYANN
Number of jobs at hospitals	Table CB1500CZ21, NAICS code 622, Estimated
Number of jobs at universities	Table CB1500CZ21, NAICS code 6113, Estimated
Number of jobs in hospitalitiy field	Table CB1500CZ21, NAICS code 72, Estimated
Number of jobs in finance field	Table CB1500CZ21, NAICS code 52, Estimated
Number of jobs in professional fields	Table CB1500CZ21, NAICS code 54, Estimated
Number of jobs in entertainment fields	Table CB1500CZ21, NAICS code 71, Estimated

References

[1] Daniel Boyle. *Fixed-Route Transit Ridership Forcasting and Service Planning Methods*. Transportation Reserach Board, Washington, DC, 2006.

[2] Transportation Research Board and National Academies of Sciences, Engineering, and Medicine. *Traveler Response to Transportation System Changes Handbook, Third Edition: Chapter 1, Introduction*. The National Academies Press, Washington, DC, 2013.

[3] Michael G. McNally. The four-step model. In Kenneth J. Button, editor, *Handbook of Transport Modelling (Handbooks in Transport, Volume 1)*, chapter 3, pages 35–53. University of California, 2008.

[4] Marvin L Manheim. *Fundamentals of transportation systems analysis*. MIT Press, Cambridge, MA, 1979.

[5] Michael Florian, Marc Gaudry, and Christian Lardinois. A two-dimensional framework for the understanding of transportation planning models. *Transportation Research Part B: Methodological*, 22(6):411–419, December 1988.

[6] Sound Transit. ST3 Regional High-Capacity Transity System Plan: Transit Ridership Forecasting Methodology Report. Technical report, Sound Transit, March 2015.

[7] Sound Transit. ST3 Regional High-Capacity Transity System Plan: Addendum to Transit Ridership Forecasting Methodology Report. Technical report, Sound Transit, April 2015.

[8] Xuehao Chu. Ridership models at the stop level. Technical Report NCTR-473-04, National Center for Transit Research, Center for Urban Transportation Research, University of South Florida, December 2004.

[9] Michael Kuby, Anthony Barranda, and Christopher Upchurch. Factors influencing light-rail station boardings in the united states. *Transportation Research, Part A: Policy and Practice*, 38(3):223–247, March 2004.

[10] Brian D. Taylor, Douglas Miller, Hiroyuki Iseki, and Camille Fink. Nature and/or nurture? analyzing the determinants of transit ridership across us urbanized areas. *Transportation Research Park A: Policy and Practice*, 43(1):60–77, January 2009.

[11] Graham Currie, A Ahern, and Alexa Delbosc. Exploring the drivers of light rail ridership: An empirical route level analysis of selected australian, north american and european systems. *Transportation*, 38(3):545–560, May 2011.

[12] Matthew Durning and Craig Townsend. Direct ridership model of rail rapid transit systems in canada. *Transportation Research Record: Journal of the Transportation Research Board*, 2537:96–102, January 2015.

[13] Javier Gutirrez, Osvaldo Daniel Cardozo, and Juan Carlos Garca-Palomares. Transit ridership forecasting at station level: an approach based on distance-decay weighted regression. *Journal of Transport Geography*, 19(6):1081 – 1092, 2011. Special section on Alternative Travel futures.

[14] Xiaobai Yao. Where are public transit needed - examining potential demand for public transit for commuting trips. *Computers, Environment and Urban Systems*, 31:535–550, 2007.

[15] Erick Guerra, Robert Cervero, and Daniel Tischler. Half-mile circle. *Transportation Research Record: Journal of the Transportation Research Board*, 2276:101–109, 2012.

[16] Ahmed El-Geneidy, Michael Grimsrud, Wasfi Rania, Paul Tetreault, and Julien Surprenant-Legault. New evidence on walking distances to transit stops: Identifying redundancies and gaps using variable service areas. *Transportation*, 41(1):193–210, January 2014.

[17] Don Pickrell. Urban rail transit projects: Forecast versus actual ridership and costs. final report. Technical report, Transportation Systems Center, October 1989.

[18] Bent Flyvbjerg, Mette Skamris, and Soren L. Buhl. Inaccuracy in traffic forecasts. *Transport Reviews*, 26(1):1–24, January 2006.

[19] Matthew H. Hardy, Soongwan Doh, Junyang Yuan, Xin Zhou, and Kenneth J. Button. The accuracy of transit system ridership forecasts and capital cost estimates. *International Journal of Transport Economics / Rivista internazionale di economia dei trasporti*, 37(2):155–168, 2010.

[20] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

[21] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

[22] J. A. Nelder and R. W. M. Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society. Series A (General)*, 135(3):370–384, 1972.

[23] Mee Young Park and Trevor Hastie. L1 regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):659–677, 2007.

[24] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[25] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[26] R. A. Olshen Leo Breiman, Jerome Freidman and Charles J. Stone. *Classification and Regression Trees*. Taylor and Francis, New York, NY, USA, 1984.

[27] Patrik Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5:1457–1469, 2004.

[28] Niall Hurley and Scott Rickard. Comparing measures of sparsity. *IEEE Transactions on Information Theory*, 55(10):4723–4741, 2009.

[29] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67:301–320, 2005.