North Carolina Road Safety Analysis and Prediction

Adrian Chan (adrian27513@gmail.com) and Ye Qin (yqin7@ncsu.edu)

***Abstract*—North Carolina Road Safety factors can be attributed to the type of route, type of section, length of the section, number of lanes, average annual daily traffic, and congestion level. As a result, a safety prediction system will be built for different roads. This system makes it possible to make safety and accident predictions for otherwise unknown roads. This system can provide excellent convenience for future traffic travel. For example, using this system to the navigation software, the navigation software can tell the driver the difficulty of driving different routes and the probability of accidents. This technology holds great promise and is, at the same time, relevant to life. These reasons were the motivation for the research team to develop this system. In this system, a neural network will be trained with known road data and the number of accidents to predict the safety factor and the number of accidents that occur on unknown roads. The professor provided road data and accident counts.**

I. INTRODUCTION

In the first step of this project, we need to remove the data in the database where the safety column is equal to 49.99 or 49.999 or similar. This is because data like 49.99 are invalid. Also, the number of invalid data and the percentage of invalid data need to be recorded. After clearing the invalid data, we need to determine the authenticity of the data with a safety equal to 0. Similarly, the validity of 0.4 and 0.6 needs to be determined.

In the second step of the project, the distribution of safety values, that is, how many times different safety values have occurred, needs to be known. After this, we need to know the distribution of safety values under different conditions using a scatter plot. Looking at the scatter plot distribution, one can roughly determine whether a variable is linearly or nonlinearly related to the safety value.

II. DATA CLEANING AND VARIABLE CODING

First, we needed to remove invalid data like 49.99 and 49.999. We dropped 208 invalid values using pandas, which was 3.09% of the total dataset.

The second step was to determine whether the safety values of 0 were valid. Based on our judgment of the data distribution, we consider the safety value equal to 0 as a valid value. If the zero safety values were truly missing values, the distribution of the predictors said to be known transportation indicators (Congestion and Turbulence) would be a similar distribution to the whole dataset for each type. However, they were not and were significantly different from the overall dataset. This is where we suspected that 0 is not missing data but valid data.

A similar argument can be made for the values 0.4 and 0.6. If these were filler values, we expect the distribution of transportation indicators to be like the full dataset. However, they are more like the distribution for zero safety values which shows they are not just filler values.

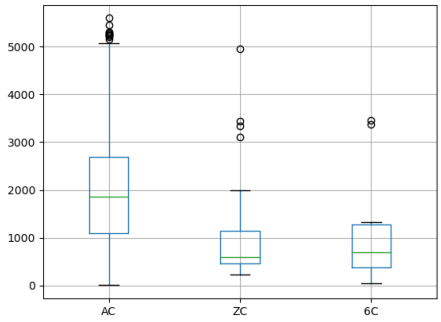

Fig. 1. Type0 figure, AC is the congestion distribution with full dataset; ZC is the congestion distribution with zero safety. 6C is the congestion distribution with 0.6 safety.

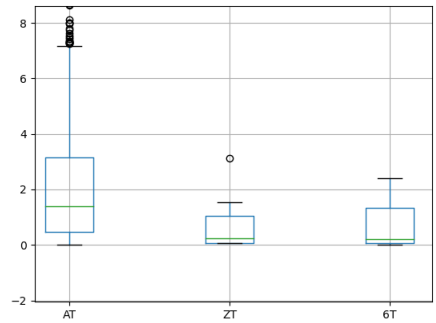

Fig. 2. Type1 figure, AT is the turbulence distribution of the full dataset; ZT is the turbulence distribution with zero safety.

To clean the dataset, we removed the ID and AADT_Profile since they were not filled with any values. Then categorical values were coded with L-1 binary values where L is the number of categories in a column.

III. Statistical Analysis

While observing the distribution of safety with a histogram, it was clear that safety is extremely right-skewed.

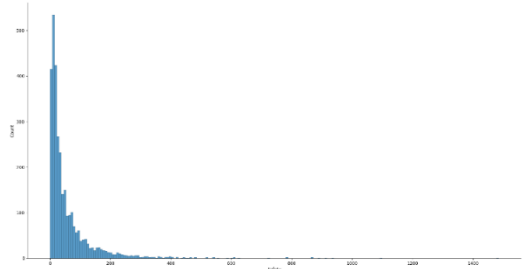

Fig 3. Safety Distribution Histogram Plot

Due to this, we decided to transform the safety response variable using log(x + 1) to change the distribution of safety to become normal.

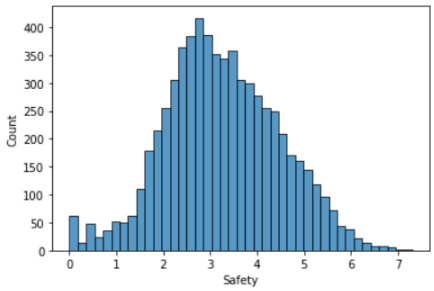

Fig 4. Transformed Safety Distribution Histogram Plot using a log(x + 1) transformation.

We also made pair scatter plots of the distribution between predictors and recorded the following highly correlated pairs: ACC/DCC Lane Length vs. Segment length, AADT main vs. AADT ON, AADT main vs. congestion level, AADT on vs. congestion level, AADT ON vs. turbulence level.

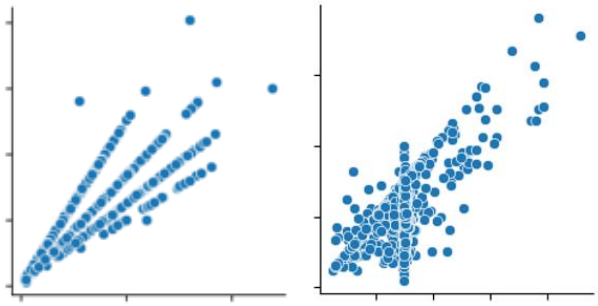

Fig. 5. Left: x axis: AADT main, y axis: congestion level. Right: x axis: Segment length, y axis: ACC/DCC Lane length.

We found that most predictors have little or no correlation with each other. They are distributed parallel to the x-axis, parallel to the y-axis, or simply a point.

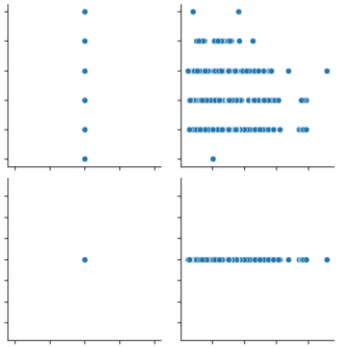

Fig. 6. Some predictors are not correlated to each other.

In addition, the correlation of some predictors is difficult to know from the scatter plot, and we may need to use other tools, such as linear regression, to know the correlation between them.

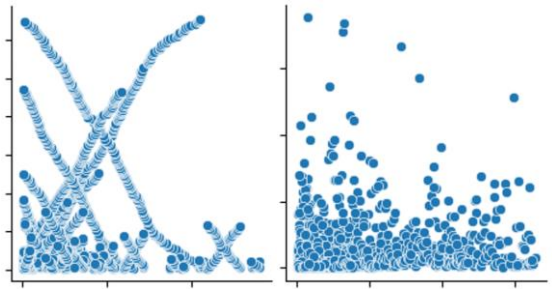

Fig. 7. Left: x axis: MM, y axis: Segment. Right: x axis Segment, y: axis Safety.

Once we examined the pair plots, we started to do a regression analysis to choose the best predictors of safety. We broke the training dataset into five equal sections and performed k-fold cross-validation. We then used scikit-learn's select K-best function using an f-regression score function which computes the F-statistic and p-values of a predictor. To determine the number of predictors we will use, we ran K-best with a range of 1 to 500, calculating the OLS MSE at each step. The number of predictors with the lowest MSE was chosen as the set of predictors used in the prediction system.

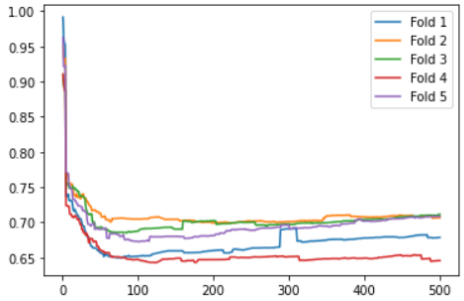

Fig 8. MSE vs Features plot of each fold for transformed response.

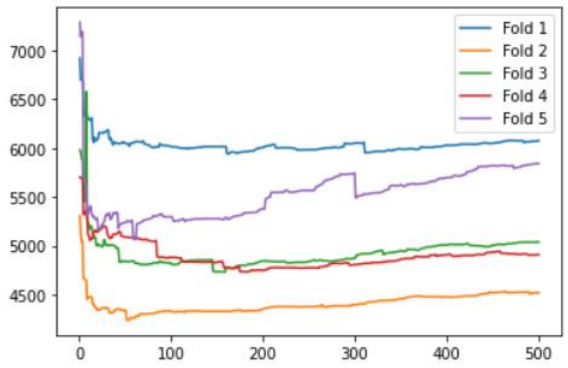

Fig. 9 MSE vs Features plot of each fold when predictions are back transformed to the original response variable.

The fold with the best original MSE score on the validation test set was fold one at 4399.205 with 51 predictors. Due to this, we selected the predictors chosen in fold 1 to be used as the final set of predictors. The predictors we found to be the most useful in explaining safety are listed below:

Segment, MM, Segment Length (ft), # of Lanes: Mainline, Acc/Dcc Lane Length (ft), Weave Segment Ls (ft), Weave Segment LCRF, Weave Segment LCFR, Weave Segment NW, Ramp to Ramp Dem. (vph), AADT_Main, AADT_On, AADT_Off, Congestion_Level, turbulance_level, Route_I227, Route_I40, Route_1440, Route_I485, Route_I540, Route_I73, Route_I74 – Greensboro, Route_I74 – Virginia Border, Route_I77, Route_I795, Route_I85, Route_NC540, Route_US1(Henderson), Route_US17(Elizabeth City), Route_US17(Windsor City), Route_US264, Route_US29, Route_US421(SilverCity), Route_US64(PhelpsLake), "Route_US64(Pittsboro), Route_US64(Rocky Mount), Route_US70(Goldsboro), Route_US74(ForestCity), Route_US(KingsMountain), Route_US74(Waynsville), Route_US74/ I74(Lumberton), Route_Us421(WinstonSalem and Boone), Route_Wade Ave., RoadType_Other, RoadType_US, General Purpose Segment Name_BERKELEY BLVD, General Purpose Segment Name_GORMAN ST, General Purpose Segment Name_I-795, Mainline Dem. (vph)_N/A_S, Mainline Single Unit Truck and Bus (%)_NA_S, and Mainline Tractor Trailer (%)_N/A_S.

Our analysis confirmed the features known to be good predictors by domain experts (Road Type, Congestion Level, and Turbulence Level) were useful in explaining safety.

IV. Prediction System Design

For the prediction system, the dataset was broken into the same five subsets from part 3 to continue running K-fold cross-validation. Using the predictors chosen from part 3, we compared three models: a GLM using a Poisson link function, an OLS model, and a neural network.

The average RMSE training performance over five folds for the Poisson GLM was 348.274, and the MAE was 9.451. The average RMSE training performance over five folds for the OLS model was 278.181, and the MAE was 193.719. The average RMSE training performance for the neural network over five folds was 342.036, and the MAE was 26.2966.

From these three models, we selected the GLM with a Poisson link function to be our final prediction system because the RMSE was comparable with the neural network's performance and the MAE of the Poisson GLM was lower than the neural network's and OLS' MAE. The final RMSE performance of the Poisson GLM was 67.424, and the MAE was 0.3018.

V. Literature Review

In textbook section 4.7.1, we learn that the closer the number is to 1, the more closely correlated the two variables are. Hence, the variable lag and variable Today are not significantly correlated. The only variables in the table that strongly connect are Year and Volume. In Figure 9, we can see that as the volume increases, the year also rises. We can learn from the textbook that digitizing the association between different predictors will be much more efficient than looking at the scatter plot distribution directly. [1]

	Year	Lag1	Lag2	Lag3	Lag4	Lag5
Year	1.0000	0.02970	0.03060	0.03319	0.03569	0.02979
Lag1	0.0297	1.00000	-0.02629	-0.01080	-0.00299	-0.00567
Lag2	0.0306	-0.02629	1.00000	-0.02590	-0.01085	-0.00356
Lag3	0.0332	-0.01080	-0.02590	1.00000	-0.02405	-0.01881
Lag4	0.0357	-0.00299	-0.01085	-0.02405	1.00000	-0.02708
Lag5	0.0298	-0.00567	-0.00356	-0.01881	-0.02708	1.00000
Volume	0.5390	0.04091	-0.04338	-0.04182	-0.04841	-0.02200
Today	0.0301	-0.02616	-0.01025	-0.00245	-0.00690	-0.03486

	Volume	Today
Year	0.5390	0.03010
Lag1	0.0409	-0.02616
Lag2	-0.0434	-0.01025
Lag3	-0.0418	-0.00245
Lag4	-0.0484	-0.00690
Lag5	-0.0220	-0.03486
Volume	1.0000	0.01459
Today	0.0146	1.00000

Fig. 10. From textbook: Stock Market

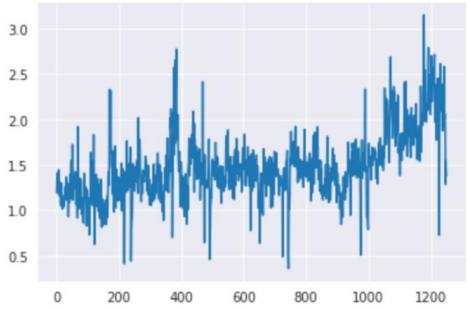

Fig. 11. x axis: volume, y axis: year

In the project, we also encountered problems like some predictors that are qualitative rather than quantified. For example, route type and segment type, these qualitative predictors also impact the safety results. For more

information on how to deal with normalized data, refer to Textbook 3.3.1. [1]

In the textbook, the authors keep the β value and use 1 and -1 to indicate whether a person has a house or not, thus determining the effect of house on the balance. In our project we can also try the same approach to quantify the qualitative predictor. [1]

$$x_i = \begin{cases} 1 & \text{if } i\text{th person owns a house} \\ -1 & \text{if } i\text{th person does not own a house} \end{cases}$$

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i = \begin{cases} \beta_0 + \beta_1 + \epsilon_i & \text{if } i\text{th person owns a house} \\ \beta_0 - \beta_1 + \epsilon_i & \text{if } i\text{th person does not own a house.} \end{cases}$$

VI. Learning Outcome

Ye Qin

In this project, my primary responsibility was to do the literature review and write the paper report. My primary responsibility in the previous semester's project in ECE301 was building the code, so I tried a different task this time. The most important thing I learned from this project is that I know how to apply the knowledge I learned in the course to the application of the project. In most of my ECE classes in undergrad, I just knew how to do the problems. Although I got most of the problems correct, I had yet to learn why the problems were done and what problems they solved. This project was an excellent way to improve my knowledge about linear regression, K-fold, and bootstrap in the course. In addition, the code was much more complex than in the previous course. Compared to MATLAB, the code introduced many different Python libraries, and I tried to learn how to use them. This project is a valuable reference for my senior design, and both projects have common aspects of training and validation.

Adrian Chan

My contribution to the project was primarily focused on implementing the code. Some tasks I did were processing the data to be analyzed through plots, training and creating models, and ultimately implementing the final prediction system of our project.

Through the data analysis part of the project, I learned how to process real-world data using pandas and other Python packages. Looking at the raw data, a numerical summary of the data and the data on a plot can give radically different insights about the data and change how it can be interpreted. The difficulty in the process and where I learned the most was understanding how best to explain if a given value is a filler value or an actual value.

The statistical analysis part of the project was the most interesting to me. I learned in detail how to choose certain features for prediction, do k-fold cross-validation, and apply the skills taught in class to create an end-to-end pipeline for machine learning. Moreover, through the statistical analysis, I gained a better understanding of making decisions about machine learning tasks to create a more accurate prediction system.

The final training of the prediction system was where I spent much time tuning the neural network model to perform better than the linear model. However, in the end, I found that linear models will perform equally or better than a generic neural network when certain assumptions are met.

VII. Code Download Link

https://drive.google.com/file/d/1CY3zDpQyCy2PDGhrKPHzCNeGjBCtyBx3/view?usp=sharing

VIII. Video Presentation Link

https://drive.google.com/file/d/15sFUpsGmzoQtvBrynihCqQZ26m-_vdpB/view?usp=sharing

IX. Reference

[1] James, Gareth author. (Gareth Michael) et al, An Introduction to Statistical Learning: With Applications in R. ([Uncorrect]. ed.) New York: Springer, 2013;2017;103.