Module 3: Critical Thinking

Predicting Fuel Efficiency Using TensorFlow

Nolan Byrnes

CSC580 – Applying Machine Learning and Neural Networks - Capstone

Colorado State University – Global Campus

Dr. Isaac Gang

October 1st, 2023

Module 3: Critical Thinking

Training a model to predict a continuous value, such as the fuel efficiency of a vehicle is a regression problem. In a regression problem, the output is a continuous value and with the value being continuous, we must consider the loss function which “computes the distance between the current output of the algorithm and the expected output” (Pere, 2020, para. 3). This paper provides a detailed analysis of the regression model built to predict fuel efficiency using regression and observing the behaviors of the loss functions Mean Squared Error (MSE) and Mean Absolute Error (MAE).

The dataset that was used was the Auto MPG dataset, which included features such as MPG, cylinders, displacement, acceleration, model year, and origin. For this project, we separated the MPG column from the rest of the dataset, so that we could have our model predict what the MPG would be with the other features of the dataset. Figure 1 below displays the tail of the data set before we split the data into test and training data sets. The values of the MPG feature were also scaled to be between 0 and 1.

Figure 1.

Data set Tail

A screenshot of a computer

Description automatically generated

Note. This figure displays the tail of the Auto MPG dataset.

Comparing the Weight to some of the other features, it can be observed in figure 2 below that the cars that had more weight had smaller value for MPG. Also, with the car weighing more, it was also more likely to have more cylinders and displacement. Generally, we can see trending relationships between the features of the data set, but it is important to note that within the plots in figure 2, there are outliers when comparing the weight to MPG and weight to displacement.

Figure 2.

Pairplots of Weight Feature

A graph of a number of numbers

Description automatically generated with medium confidence

Note. This figure displays pairplots of the Weight Feature, with the MPG, Cylinders, and Displacement features.

Looking at the statistics of the features individually in figure 3 below, we can see that the weight feature does have an outlier, as the max is 5140, which is far off from the mean of 2990.25. Acceleration and horsepower also have outliers when comparing the mean to the maximum value.

Figure 3.

Tail of Training data Statistics

A screenshot of a computer

Description automatically generated

Note. This figure displays the tail of the training data statistics.

We used a sequential model with two hidden layers, with 64 neurons, and the final layer having one neuron, which returned the predicted MPG of the vehicle.

Since MPG is a continuous variable, it is appropriate to use a sequential model. Instead of training our model based off accuracy, a better metric to be used would be Mean Absolute Error (MAE) or Mean Squared Error (MSE). Both metrics can tell us how far off the predictions are to the actual values, so the closer to 0, the better. MAE is calculated by calculating the absolute error of all predictions, then calculating the average. Absolute Error (AE) is the absolute value of the error, which is “the difference between the calculated value and ‘true’ value” (Statistics How To, n.d., para. 1). The MSE is calculated by squaring all the calculated AE values , then taking the average (Frost, n.d., para. 3). Squaring the values makes it so that there are no negative values. The difference between the two metrics is how they avoid negative values when calculating the metric, which can make a big difference.

The MSE is sensitive to outliers because any outliers get exaggerated, due to the errors being squared, which affects the calculated mean. MAE is more resilient to outliers since the numbers are not being squared.

The input layer included the Cylinder, Displacement, Horsepower, Weight, Acceleration, Model Year, and Origin features of the data set. The data then gets sent through two hidden layers which contain 64 neurons. Having two hidden layers with 64 neurons can help the model discover the complex relationships between the features. After the two hidden layers, the data is then fed through an output layer which contains one neuron. The output layer would represent the predicted MPG based on the features of a vehicle. You can see the model’s summary in figure 4 below.

Figure 4.

Model Summary

A screenshot of a computer program

Description automatically generated

Note. This figure displays summary of the model, which contains details about the hidden layers and type of model.

Since we standardized the features of the data sets, when we make the predictions of the data set, it will return standardized values. In figure 5 below, you can observe that the predictions of the MPG of ten different samples returned values close to 0. For the values to make sense to us humans, we would need to apply the inverse to observe the true predicted MPG. Since we standardized the values by subtracting the mean from the value, then dividing by the standard deviation, we would apply the inverse to transform the predicted value back to the unstandardized value would be by multiplying the predicted value by the standard deviation and adding the mean. You can observe the standardized predictions as well as the unstandardized predictions in figures 5 and 6 below.

Figure 5.

Standardized Predictions

A computer screen with numbers and symbols

Description automatically generated

Note. This figure displays the predicted MPG values for 10 different cars, with the values being standardized.

Figure 6.

Unstandardized Predictions

A screenshot of a computer program

Description automatically generated

Note. This figure displays the predicted MPG values for 10 different cars, with the values being unstandardized, which is the values we would show end users.

During the training of the model, it went through 1000 epochs. In the beginning, when looking at the Mean Absolute Error (MAE) and Mean Square Error (MSE) for the validation data set, the validation MAE started off as 22.2164, whereas the validation MSE started off at 556.2226, and towards the end, the validation MAE was down to 2.1992, compared to the 8.3734. the validation MSE had a significant drop after the second hundred epochs, then had small decreases. The Validation MAE also had a significant drop, but then had smaller decreases around 2.2. You can view the statistics that were printed between every 100 epochs in figure 7 below. It is also important to note that the loss function was set to MSE, therefore, the values in the loss column, and MSE column are identical in figures 7 and 8 below.

Figure 7.

Model Training

A black screen with white text

Description automatically generated

Note. This figure displays the progress the model made after every 100 epochs of training.

After training the model, I printed the tail of the training history, to observe the metrics associated with the final five epochs that the model went through. The MAE and MSE metric in figure 8 below represent how close the predicted values are to the actual values in the training dataset, and the val\_MAE and val\_MSE represent the same against new data.

Figure 8.

Training History Tail

A screenshot of a computer screen

Description automatically generated

Note. This figure displays the loss function values for the final 5 epochs of training, based on the training and validation datasets.

Comparing the MAE to the validation MAE in figure 9, and the MSE to the validation MSE in figure 10, the MAE scores were closer together than the MSE. This tells us that the MAE metric is more consistent between the training and validation data. Induraj mentions that the MAE is “gives equal weight to all errors and is less sensitive to outliers” (Induraj, 2022, para. 4). After the first 100 epochs in figure 9, the MAE decreases at a slower rate than you observe the MSE in figure 10, which could potentially be because the MAE is giving equal weight to the errors.

Figure 9.

MAE

A graph with a line graph and a line graph

Description automatically generated

Note. This figure displays the resulting MAE metrics as it progressed through the 1000 epochs of training the model went through.

Looking at how MSE progressed through the epochs in figure 10 below, the MSE and the validation MSE have a wider gap between them. Induraj mentions that the MSE “gives more weight to large errors and is sensitive to outliers” (Induraj, 2022, para. 4). Because of sensitivity to outliers, you can observe a wider gap between the training and validation MSE, which shows a greater inconsistency between the data the model was trained on compared to new data. The slope of the MSE as it went through the epochs was steeper than observed with the MAE in figure 9. This indicates that the model was becoming more accurate throughout training when we paid attention to the MSE metric.

Figure 10.

MSE

A graph of a normal distribution

Description automatically generated with medium confidence

Note. This figure displays the resulting MSE metrics as it progressed through the 1000 epochs of training the model went through.

Conclusion

When training a model to predict the MPG a vehicle would have, it is best to use a sequential model, since the MPG value is continuous. Using metrics such as MSE and MAE are more appropriate to use, as it calculates how far off the predictions were to the actual values. The difference between MSE and MAE is how they handle negative values when calculating the error by subtracting the actual value from the predicted value. MSE and MAE both have their advantages and disadvantages, and we were able to observe them in this exercise. MAE was more resilient against outliers but treated all errors equally. This resulted in the validation MAE being closer to the MAE of the training data, however, the rate that the MAE improved was slower compared to the MSE. MSE gave more weight to errors, which lead to faster training, as the MSE decreased at a faster rate, compared to the MAE. If the data preprocessing included steps to remove outliers, using the MSE metric could have had more consistent results all while training faster, due to MSE giving a higher weight to errors.

**REFERENCES**

Frost, J. (n.d.). Mean Squared Error (MSE). statisticsbyjim. <https://statisticsbyjim.com/regression/mean-squared-error-mse>

Induraj. (2023, February 22). Which metrics in regression matter the most? MSE|RMSE|MAE|R2|Adj R2- Advantages/Disadvantages. Medium. <https://induraj2020.medium.com/which-metrics-in-regression-matter-the-most-mse-rmse-mae-r2-adj-r2-advantages-disadvantages-55740cb873ec>

Statistics How To. (n.d.). Absolute Error & Mean Absolute Error (MAE). <https://www.statisticshowto.com/absolute-error/>