Model Evaluation Reference:

Regression: An Explanation of Regression Metrics and What Can Go Wrong

Machine learning is continuously growing and is said to affect all domains and bring a radical change in the way the human race functions. Few advancements have already started having an impact on society like fraud detection systems, online loan approval systems, self-driving cars, tumour detection etc. Machine learning algorithms have already become part of our daily routines, from news recommendations in the morning to optimized movie recommendations from Netflix in the evening, everything we use is directly or indirectly affected or will be affected soon by machine learning.

Machine learning is basically of two types i.e Supervised Learning and Unsupervised Learning. Supervised Learning can be simply taken as learning with the help of a teacher. This means we have the data points along with labels for each data point. Unsupervised Learning, on the other hand, can be considered as learning without a teacher. In this, we are just given raw data without any labels and the algorithm is supposed to find patterns in the data and group it accordingly. Most of the progress in machine learning is achieved in the supervised learning world while the unsupervised world remains mysterious and not completely explored.

Supervised Machine learning can perform two tasks i.e Classification and Regression. Classification in very high-level terms is the task of assigning labels to data samples belonging to different classes, for e.g training, a neural network to distinguish between cats and dogs is a classification problem with cats and dogs being the two classes.

Regression, on the other hand, is the task of predicting continuous values by learning from various independent features. for e.g Predicting the price of a house based on features like the number of bedrooms, locality etc.

The basic classification or regression pipeline works as follows:

1. We start by some initial configuration of the model and predict the output based on some input.
2. The predicted value is then compared with the target and the measure of our model performance is taken.
3. Then the various parameters of the model are adjusted iteratively in order to reach the optimal value of the performance metric.

The constant performance standard is different for different tasks and efforts are taken to reach the optimal value of the standard. In the case of the classification task, the performance standard maybe accurate which means how many cases are correctly classified by our model concerning the total cases seen by our model. Other performance metrics include sensitivity(recall), specificity, precision, f1-score, AUC, mean-squared-error, mean-absolute error,

R^2, Adjusted R^2 etc and are used according to the task and data used for the task.

In this article, we would discuss metrics used in the Regression task and why R^2 becomes negative.

The regression task is the prediction of the state of an outcome variable at a particular timepoint with the help of other correlated independent variables. The regression task, unlike the classification task, outputs continuous values within a given range.

The various metrics used to evaluate the results of the prediction are :

1. Mean Squared Error(MSE)
2. Root-Mean-Squared-Error(RMSE).
3. Mean-Absolute-Error(MAE).
4. R^2 or Coefficient of Determination.
5. Adjusted R^2

Mean Squared Error: MSE or Mean Squared Error is one of the most preferred metrics for regression tasks. It is simply the average of the squared difference between the target value and the value predicted by the regression model. As it squares the differences, it penalizes even a small error which leads to over-estimation of how bad the model is. It is preferred more than other metrics because it is differentiable and hence can be optimized better.

$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y - \hat{y}\right)^2}_{\text{The square of the difference between actual and predicted}}$$

Figure 1. Mean Squared Error Formula

Root Mean Squared Error: RMSE is the most widely used metric for regression tasks and is the square root of the averaged squared difference between the target value and the value predicted by the model. It is preferred more in some cases because the errors are first squared before averaging which poses a high penalty on large errors. This implies that RMSE is useful when large errors are undesired.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Predicted_i - Actual_i)^2}{N}}$$

Figure 2. The formula of Root Mean Squared Error

Mean Absolute Error: MAE is the absolute difference between the target value and the value predicted by the model. The MAE is more robust to outliers and does not penalize the errors as extremely

as mse. MAE is a linear score which means all the individual differences are weighted equally. It is not suitable for applications where you want to pay more attention to the outliers.

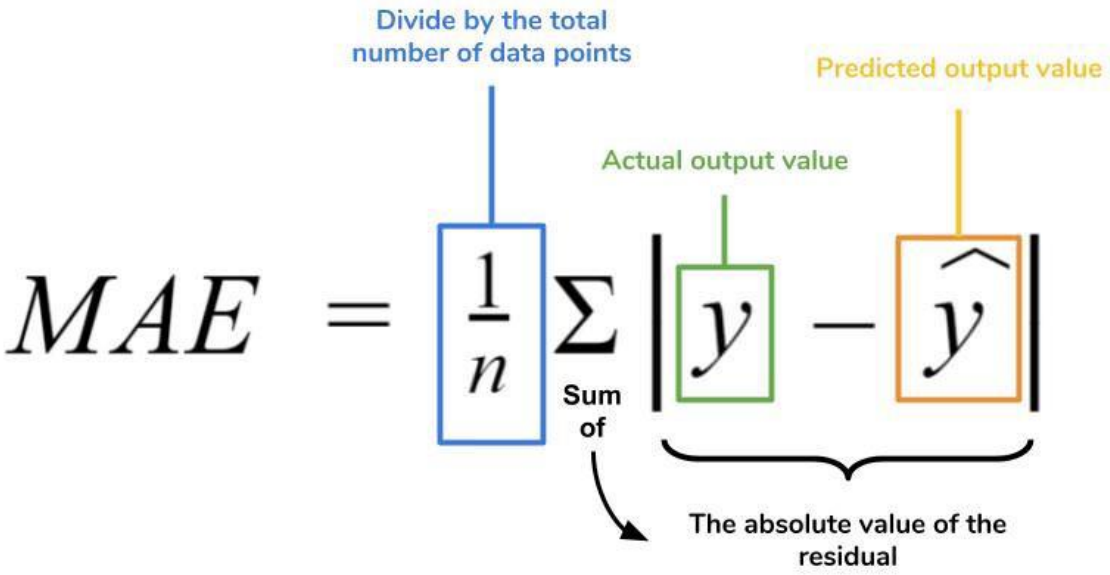


Figure 3. The Formula of Mean Absolute Error

R² Error: Coefficient of Determination or R² is another metric used for evaluating the performance of a regression model. The metric helps us to compare our current model with a constant baseline and tells us how much our model is better. The constant baseline is chosen by taking the mean of the data and drawing a line at the mean. R² is a scale-free score that implies it doesn't matter whether the values are too large or too small, the R² will always be less than or equal to 1.

$$R^2 = 1 - \frac{\mathrm{MSE(model)}}{\mathrm{MSE(baseline)}}$$

Figure 4. The Formula for R²

Adjusted R²: Adjusted R² depicts the same meaning as R² but is an improvement of it. R² suffers from the problem that the scores improve on increasing terms even though the model is not improving which may misguide the researcher. Adjusted R² is always lower than R² as it adjusts for the increasing predictors and only shows improvement if there is a real improvement.t

$$R_a^2 = 1 - \left[\left(\frac{n-1}{n-k-1}\right) \times (1 - R^2)\right]$$

where:
n = number of observations
k = number of independent variables
R_a^2 = adjusted R^2

Figure 5. The Formula of Adjusted R²

Why is R² Negative?

After giving you a brief overview of the various regression metrics, let us finally talk about why R² is negative.

There is a misconception among people that the R² score ranges from 0 to 1 but actually, it ranges from -∞ to 1. Due to this misconception, they are sometimes scared why the R² is negative which is not a possibility according to them.

The main reasons for R² to be negative are the following:

1. One of the main reasons for R² to be negative is that the chosen model does not follow the trend of the data causing the R² to be

negative. This causes the mse of the chosen model(numerator) to be more than the mse for constant baseline(denominator) resulting in negative R^2.

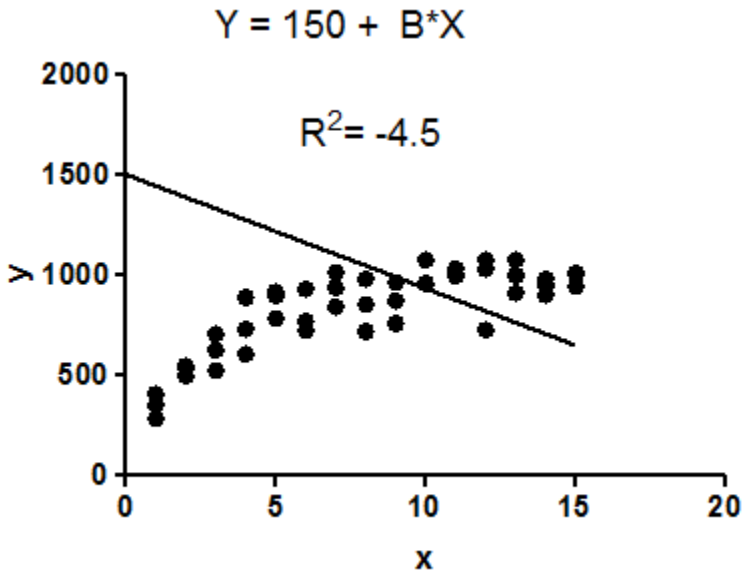


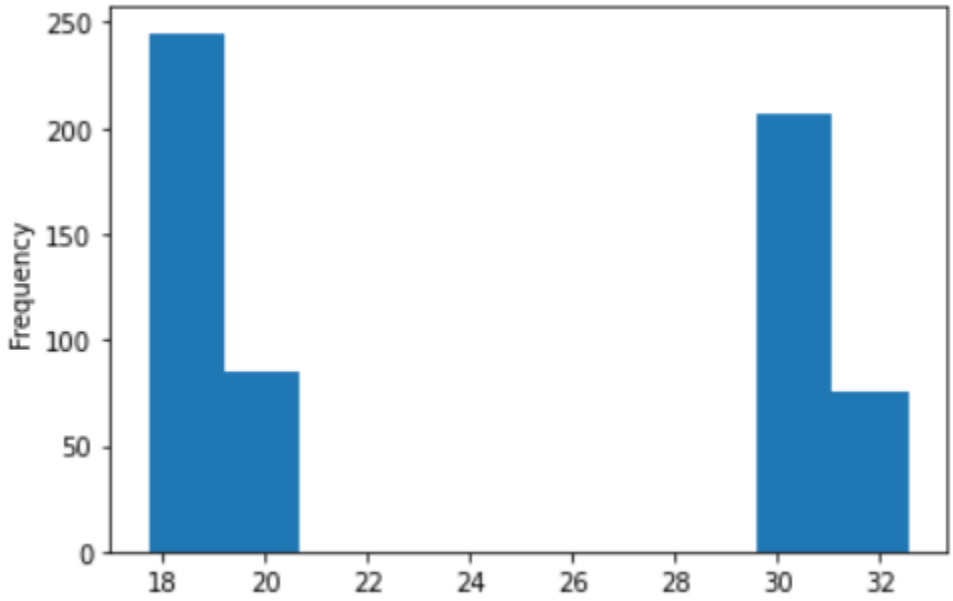
Figure 6.

2. Maybe there are a large number of outliers in the data that causes the mse of the model to be more than mse of the baseline causing the R^2 to be negative(i.e the numerator is greater than the denominator).

3. Sometimes while coding the regression algorithm, the researcher might forget to add the intercept to the regressor which will also lead to R^2 being negative. This is because, without the benefit of an intercept, the regression could do worse than the sample mean(baseline) in terms of tracking the dependent variable (i.e., the numerator could be greater than the denominator). However, most of the standard machine learning libraries like scikit-learn include the intercept by default but if you are using the stats-model library then you have to add the intercept manually.

Personal Experience:

Recently I was working on a regression problem where my model was trained on a data having a range of dependent variables between 17–35 but the range was disjoint in the middle. To clarify, the data had values from 17–22 then a break and then again from 29–33. If I plot the data, it would be something like in the figure below.



The model had an r2_score of 0.95 on the validation set. After the model was trained, I was asked to test the model's performance on subsets of data that is between 17–22 and 29–33. The model's performance on each of the subsets has a negative r2_score and I was very confused and wondered where I was going wrong. I checked the performance of the model on the combined dataset and it was similar to that of the validation set but as I divided the testing data into subsets and tested, a negative r2_score was seen. Then after thinking for a long time and implementing the r2_score

function by hand and printing output at each stage of the calculation, I realized the problem.

So let us consider, the 17–22 case and figure out why the r2_score was negative. As discussed above, r2_score tells the performance of the model as compared to the mean estimator i.e. a model that takes the mean of the dependent variable and predicts it for all the entries. In normal cases, the mean estimator is a bad model and fails. But when we are subsetting our data into smaller subsets, the mean estimator is actually a good estimator. If for example, we consider that the mean of our dependent variable is 19 for the 17–22 subset. Now if we consider our model that is trained on data from the entire range and not just 17–22, it would make an error of say 1 on average which is very good if we consider the entire range from 17–33 but if we subset the data into 17–22, an average error of 1 might be worse than the mean estimator thereby leading to negative r2_score. A similar scenario would be observed when the data will be a subset in the 19–33 subset.

Conclusion:

In this post, we discovered the various metrics used in regression analysis and also tried to answer the question of *why R^2 is negative?*

Reference Link: https://towardsdatascience.com/regression-an-explanation-of-regression-metrics-and-what-can-go-wrong-a39a9793d914