Regularization

Anish Susarla

May 2022

1 Introduction

Figure 1: Dr. Andrey N. Tikhonov

In machine learning, overfitting refers to when a model learns the detail and noise in the training data to the extent where model performance will be degraded when evaluated on testing data. The goal for an ideal machine learning algorithm is for it to learn the significant features from a training dataset so that it can effectively evaluate data on data it has not seen before. This is where overfitting techniques come into picture. While there are numerous techniques to reduce overfitting, such as k-fold cross-validation and data augmentation, one of the most common techniques to reduce overfitting is regularization. One of the earliest forms of regularization was Tikhonov regularization, developed by Soviet mathematician Dr. Andrey N. Tikhonov. Tikhonov regularization was meant to solve the issue of multicollinearity, which is when one independent variable is highly correlated with one or more independent variables. This is a problem because it undermines the statistical significance of an independent variable. Tikhonov regularization, now known as ridge regression, detailed below.

2 Bias vs. Variance

To fully understand regularization, we must also understand the difference between bias and variance (in a machine learning sense). Bias refers to the difference between the average prediction of the model and the ground truth/correct value that the model should have returned if the model were to have 100% accuracy. An ideal machine learning model would reduce either the absolute value or the squared value of the bias (to account for if the predicted value is less than the ground truth value), since we want the difference between the model's predicted value and the ground truth to be as small as possible (so long as it can be generalized to the testing data as well). Variance refers to the variability of model prediction for a given data point or value - in other words, how much the model can adjust given the dataset. A low bias usually corresponds with a high variance, and vice versa, and if we think about it, this makes intuitive sense! As mentioned previously, a low bias would correspond to the model being able to predict values in a dataset relatively well, which means that the model can adjust given the dataset (since not every value in the dataset is going to be the same), which corresponds to a high variance.

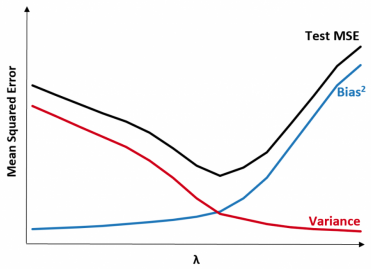
2.1 Residual Sum of Squares

One fundamental part of regularization is the residual sum of squares (RSS). If you haven't heard of this term before, you can think of it as basically the same thing as mean squared error (MSE). The only difference is that the residual sum of squares is, as the name suggests, the sum of squared errors, whereas the mean squared error is the average of the squared errors.

$$\text{RSS} = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$$

2.2 Bias v. Variance Tradeoff

While it may seem that we should always strive for the highest variance as possible, this can actually increase bias, and therefore, model loss (such as mean squared error). However, a fundamental part of machine learning is understanding the bias versus variance tradeoff, and that we need to find an appropriate value for our algorithm (where λ can represent algorithm complexity). As can be seen, while having a high variance often correlates with a lower , and lower bias, as can be seen, the relative mean squared error is high. This is because total error can usually be thought of as the bias^2 + variance + some other irreducible error. Finding the point of lowest error is one of the purposes of regularization! Specifically, we'll build off of RSS and go into how to find an appropriate λ value.



3 Ridge vs. Lasso Regression

3.1 Ridge Regression

Ridge regression builds off the residual sum of squares by including a shrinkage quantity. It would make intuitive sense for the goal to reduce/minimize this function, since we would want to reduce the RSS/MSE. Ridge regression is usually used when you have more parameters than samples. In the equation below, λ can be used to represent algorithm complexity. In addition to the RSS, as mentioned previously, a shrinkage quantity is added, which multiplies λ by the squared weight of each individual feature.

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

In short, ridge regression can be seen as almost identical to linear regression, except that if we introduce a small bias, we can get better long term predictions for the model that should perform generally well on data that it has not seen before as well. Additionally, ridge regression should be used when there are many significant predictor variables, since it will keep all the predictors in the model.

3.2 Lasso Regression

Lasso regression is almost identical to Ridge regression, except that when calculating the penalty term, the absolute values of the weights are taken, versus the squared values of the weights, like that is done in Ridge regression.

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 + \lambda\sum_{j=1}^{p}|\beta_j| = RSS + \lambda\sum_{j=1}^{p}|\beta_j| + \lambda\sum_{j=1}^{p}\beta_j^2$$

In short, lasso regression is extremely similar to ridge regression, instead the absolute values of the weights are taken. Additionally, lasso regression should be used when there are a small number of significant predictor variables, as lasso regression will attempt to shrink non-important coefficients to zero and remove them from the model.

4 Elastic-Net Regression

As can be inferred from the previous section, one fundamental difference between Lasso and Ridge regression is how it excludes variables. Lasso regression can exclude non-significant parameters/features by settings its slope to 0. However, in Ridge regression, we can only shrink the slope asymptotically close to 0. Now, why is this important? This is where Elastic-Net Regression comes into picture, which is basically just combining Ridge and Lasso regression. This form of regularization groups and shrinks the parameters, and either leaves them in the equation, or removes them all at once. This is accomplished by having two separate λ values - one for the Ridge regression penalty and another for the Lasso regression penalty.

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda_1\sum_{j=1}^{p}\beta_j^2 + \lambda_2\sum_{j=1}^{p}|\beta_j| = RSS + \lambda_1\sum_{j=1}^{p}\beta_j^2 + \lambda_2\sum_{j=1}^{p}|\beta_j|$$

5 Sources

1. https://towardsdatascience.com/regularization-in-machine-learning-76441ddcf99a
2. https://en.wikipedia.org/wiki/Tikhonov_regularization
3. https://medium.com/@mackenziemitchell6/multicollinearity-6efc5902702
4. https://www.statology.org/when-to-use-ridge-lasso-regression/
5. https://towardsdatascience.com/understanding-the-bias-variance-tradeoff-165e6942b229
6. https://link.springer.com/chapter/10.1007/978-0-585-25657-3_37
7. https://www.bmc.com/blogs/bias-variance-machine-learning/
8. https://github.com/RichmondAlake/tensorflow_2_tutorials
9. https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/