Bias and Variance Dilemma & Shrinkage Methods

Authors: Jing Fang, Andrew Han, Hao He. (PDF)

Course Content

During this week's lecture, we discussed about the Bias-Variance dilemma and how the expected loss could be split into reducible and irreducible error. Inside the reducible error, we are able to observe the left term as the bias and the right term as the variance.

We also revisited Lasso and Ridge specifically comparing the shrinkage methods to see how both models deal with regularization penalties.

Bias-Variance Trade Off

The Bias-Variance Decomposition

The expected squared loss is

$$E[L] = \int \{y(x) - h(x)\}^2 p(x)dx + \int\int \{h(x) - t\}^2 p(x,t)dxdt$$

The second term is independent of $y(x)$. It arises from the noise on the data and represents the minimum achievable value of the expected loss.

The first term depends on our choice for the function $y(x)$, and we will seek a solution for $y(x)$ which makes this term a minimum. Suppose we have a data set D containing a finite number N of data points, then the expected loss is

$$E_D[\{y(x;D) - h(x)\}^2] = \{E_D[y(x;D)] - h(x)\}^2 + E_D[\{y(x;D) - E_D[y(x;D)]\}^2]$$

It can be expressed as the sum of two terms: - The first term, called the squared *bias*, represents the extent to which the average prediction over all data sets differs from the desired regression function. - The second term, called the *variance*, measures the extent to which the function $y(x;D)$ is sensitive to the particular choice of data set.

We obtain the decomposition of the expected squared loss

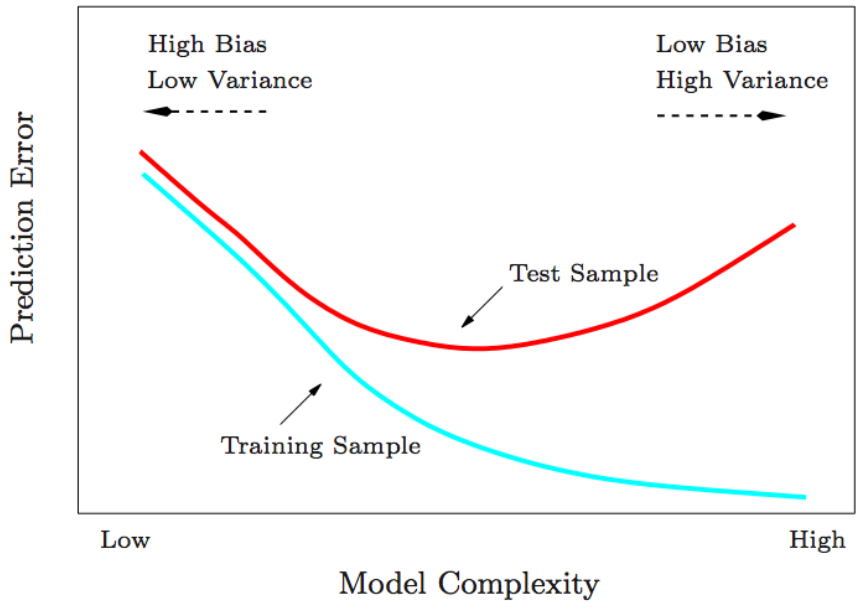
$$\text{expected loss} = (\text{bias})^2 + \text{variance} + \text{noise}$$

where

$$(\text{bias})^2 = \int \{E_D[y(x;D) - h(x)\}^2 p(x)dx, \text{variance} = \int E_D[\{y(x;D) - E_D[y(x;D)]\}^2]p(x)dx, \text{noise} = \int \int \{h(x)$$

The Bias-Variance Tradeoff

There is a trade-off between bias and variance, with very flexible models having low bias and high variance, and relatively rigid models having high bias and low variance. The model with the optimal predictive capability is the one that leads to the best balance between bias and variance.



How to Improve Model Performance: Managing Bias and Variance

1. There is no escaping the relationship between bias and variance in machine learning. Achieving low bias and low variance is the univeral goal of any supervised machine learning models. That is the measurement of the performance of a model. The parameterization of models is often a battle to balance out bias and variance. For example, changing the bundle of m and B values of a Random Forest model adjusts the number of variables used to train every tree in the forest and how many trees in total are there in the forest. Increasing m and B values on the one hand contributes to decreasing model bias

by making the model more complex, but on the other hand the more complicated model also increases model variance.

In general: - Increasing the bias will decrease the variance. - Increasing the variance will decrease the bias.

There is a trade-off at play between these two concerns and the algorithms you choose and the way you choose to configure them are finding different balances in this trade-off for your problem. However in real life, there is not a quantitative way to find the optimal model from bias-variance trade-off because we do not know the actual underlying target function. Instead we must use an accurate measure of prediction error and explore differing levels of model complexity and then choose the complexity level that minimizes the overall error.

2. Model complicity does not necessarily secure model performance in real life. It is common for people to think that they should minimize bias as much as possible even at the expense of variance with the mindset that a presence of bias indicates something basically wrong with their model and algorithm. On the other hand, eventhough a high variance is also not good but a model could at least live with that and predict well on average, at least it is not fundamentally wrong.

This kind of logic is wrong. It is true that a high variance and low bias model can preform well in some sort of long-run average sense. However, in real life we are dealing with a single realization of the data set in most cases. In these cases, long run averages are irrelevant, what is important is the one-shot performance of your model on your data and in this case bias and variance are equally important and one should not be improved at an excessive expense to the other.

3. How to reduce model variance: Bagging and resampling. Resampling techniques like Bagging could be used to reduce model variance. In bagging, numerous replicates of the original data set are created using random selection with replacement. Each derivative data set is then used to construct a new model and the models are gathered together into an ensemble. To make a prediction, all of the models in the ensemble are polled and their results are averaged.

Random Forests is one of the powerful algorithms which is built on Bagging. It works by training numerous decision trees each based on a different resampling of the original training data. In Random Forests the bias of the full model is equivalent to the bias of a single decision tree (which itself has high variance). By creating many of these trees, in effect a "forest", and then averaging them the variance of the final model can be greatly reduced over that of a single tree. In practice the only limitation on the size of the forest is computing time as an infinite number of trees could be trained without ever increasing bias and with a continual (if asymptotically declining) decrease in the variance.

Shrinkage Method: Lasso vs. Ridge

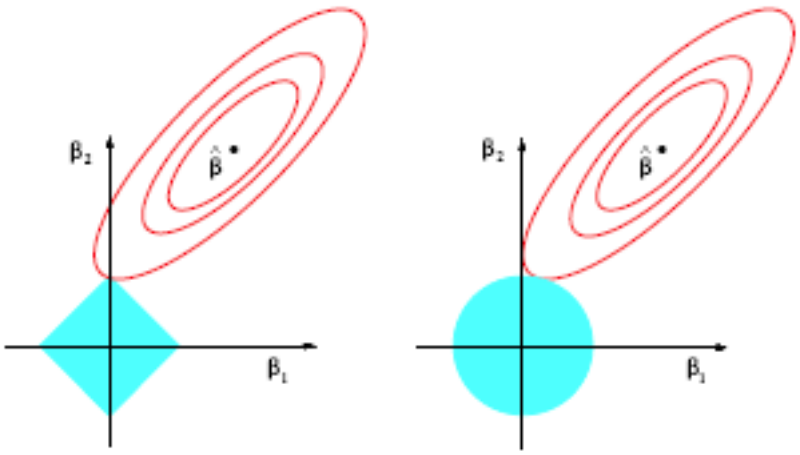


Figure 3.12: *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions* $|\beta_1| + |\beta_2| \leq t$ *and* $\beta_1^2 + \beta_2^2 \leq t^2$*, respectively, while the red ellipses are the contours of the least squares error function.*

Shrinkage happens when the coefficients decrease towards zero compared to the OLS parameter estimates which is also called regularization.

In both cases, we are assuming that linear regression with two variables with the mean removed. When we choose different β_1s and β_2s, we end up with different values for the objective function which is the residual sum of squares (RSS). The red contour lines are for the objective function or the RSS when different βs are used. $\hat{\beta}$ is the optimal β and that yields the minimum RSS.

For Ridge, we select a β that lies within the sphere as $\sum_{i=1}^{p}(\beta_j)^2 < c$. Therefore, mathematically the constraint is the L2 norm which adds up the square of every dimension and taking the sqaure root ($\beta_1^2 + \beta_2^2 \leq c$). The best β we can get from this constraint is the intersection of the sphere with the contour line.

On the other hand, the constraint that Lasso assuming that $\sum_{i=1}^{p}|\beta_j| < c$ has is a diamond. This is considered the L1 norm which is basically adding up the absolute value of every dimension ($|\beta_1| + |\beta_2| \leq c$). The ellipse will touch at an axis and this happens when the best intersection sets one coordinate precisely to 0.

As p increases, in Lasso, the multidimensional diamond will have increasing number of corners, and so it is more likely that some coefficients will be set to zero. On the other hand, the coefficients reduce by the same proportion closer to zero but not being set to zero.

Source

Understanding the Bias-Variance Tradeoff

Gentle Introduction to the Bias-Variance Trade-Off in Machine Learning

A Modern Take on the Bias-Variance Tradeoff in Neural Networks

Regularization and Shrinkage: Ridge, Lasso and Elastic Net Regression