Why Overfitting is More Dangerous than Just Poor Accuracy, Part I

Arguably, the most important safeguard in building predictive models is complexity regularization to avoid overfitting the data. When models are overfit, their accuracy is lower on new data that wasn't seen during training, and therefore when these models are deployed, they will disappoint, sometimes even leading decision makers to believe that predictive modeling "doesn't work".

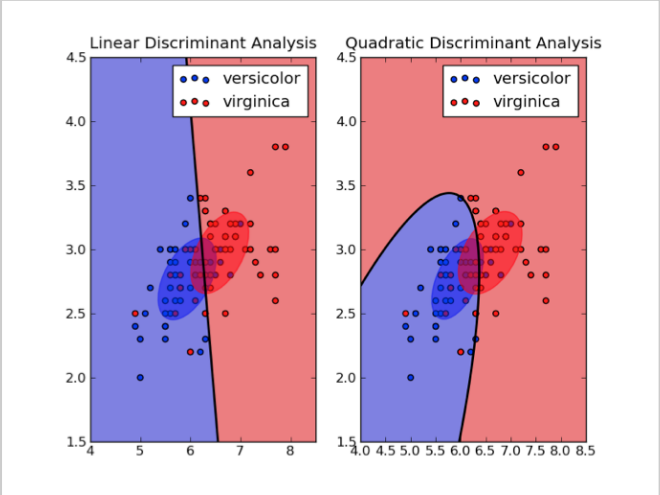
Overfit, however, is thankfully a well-known problem and every algorithm has ways to avoid it. CART® and C5 trees use pruning to remove branches that are prone to overfitting, CHAID trees require splits are statistically significant to add complexity to the trees. Neural networks use held-out data to stop training when accuracy on held-out data becomes worse. Stepwise regression uses information theoretic criteria like the Akaike Information Criterion (AIC), Minimum Description Length (MDL), or the Bayesian Information Criterion (BIC) to add terms only when the additional complexity is offset by enough reduction of error.

But overfitting has more problems than merely misclassification cases in holdout data or incurring large errors for regression problems. Without loss of generality, this discussion will only describe overfilling in classification problems, but the same principles apply in regression problems as well.

One way modelers reduce the likelihood of overfit is to apply the principle of Occam's Razor, where if two models exhibit the same accuracy, we will prefer the simpler model because it is more likely to generalize well. By simpler, we must keep in mind that we prefer models that *behave* more simply rather than models that just appear to be simpler because they have fewer terms. John Elder (a regular contributor to the PA Times) has a fantastic discussion of that topic in the book by Seni and Elder, Ensemble Methods in Data Mining.

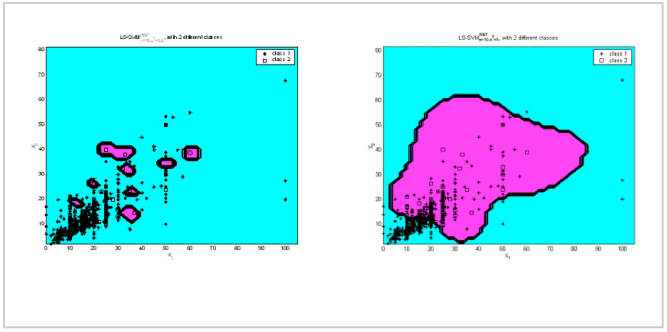
Consider this example contrasting linear and nonlinear models. The figure below shows decision boundaries for two models separates two classes of the famous Iris Data (http://archive.ics.uci.edu/ml/datasets/Iris). On the left is the decision boundary from a linear model built using linear discriminant analysis (like LDA or the Fisher Discriminant) and on the right, a decision boundary built by a model using quadratic discriminant analysis (like the Bayes Rule). The image can be found at http://scikit-learn.org/0.5/auto_examples/plot_lda_vs_qda.html.

It appears that the accuracy of both models is the same (let's assume that it is), yet the behavior of the models is very different. If there is new data to be classified that appears in the upper left of the plot, the LDA model will call the data point versicolor whereas the QDA model will call it virginica. Which is correct? We don't know which would be correct from the training data, but we do know this: there is no justification in the data to increase the complexity of the model from linear to quadratic. We probably would prefer the linear model here.



Apply models to regions in the data without data is the entire reason for avoiding overfit. The issue with the figure above was with model behavior when doing *extrapolation*, where we want to make sure that the models behave in a reasonable way for values outside (larger than or smaller than) the data used in training. But models also need to behave well when they *interpolate*, meaning we want models to behave reasonably for data in between data that exists in the training data.

Consider the second figure below showing decision boundaries for two models built from a data set derived from the famous KDD Cup data from 1998. The two dimensions in this plot are Average Donation Amount (Y) and Recent Donation Amount (X). This data tells the story that higher values of average and recent donation amounts are related to higher likelihoods of donors responding; note that for the smallest values of both average and recent donation amount, at the very bottom left of the data, the regions are colored cyan.



Both models are built using the Support Vector Machines (SVM) algorithm, but with different values of the complexity constant, C. Obviously, the model at the left is more complex than the model on the right. The magenta regions represent responders and the cyan regions represent non-responders.

In the effort to be more accurate on training data, the model on the left creates closed-decision boundaries around any and all groupings of responders. The model at the right joins these smaller blobs together into a larger blob where the model classifies data as responders. The complexity constant for the model at the right gives up accuracy to gain simplicity.

Which model is more believable? The one on the left will exhibit strange interpolation properties; data in between the magenta blobs will be called non-responders, sometimes in very thin regions between magenta regions; this behavior isn't smooth or believable. The model at the right creates a single region of data to be classified as a responder and is clearly better than the model at the left.

Beware of overfitting the data and test models not just on testing or validation data, but if possible, on values not in the data to ensure its behavior, whether interpolation or extrapolation, is believable.

In part II, the problem overfitting causes for model interpretation will be addressed.

This article first appeared at the Predictive Analytics Times, http://www.predictiveanalyticsworld.com/patimes/why-overfitting-is-more-dangerous-than-just-poor-accuracy-part-i/

POSTED BY DEAN ABBOTT AT 1:20 PM