Ensembles/Combining Predictors

For SEL prediction problems, it's easy to see that linear combinations of predictors can be better than any single one of the component predictors. (See the arguments in the "stacking" section of Module 21 in this regard.) This fact suggests that a variety of prediction methods might be employed in a particular problem and then ultimately some kind of weighted version of the predictions be employed. (There is, of course, some law of diminishing returns at work here. Beyond some point adding a new methodology to the mix can't be beneficial. After all, the whole point is approximating the conditional mean function $E[y|x]$ and once one is able to get "close" to this by combining predictors $\hat{f}(x)$, nothing is to be gained by allowing the use of more complicated approximating forms.)

More or less formal ways of combining predictors include 1) "Bayesian model averaging" and 2) "stacking" discussed in Module 21. The basic story in the first of these is:

Find Bayes predictors in several models and weight these together by posterior (conditional given the training data) probabilities of those models being the best description available for the scenario under study.

The basic story of stacking is:

Use a careful cross-validation type methodology to seek good coefficients to employ when making a linear combination of a (hopefully "diverse") set of predictors.

Less formal ways of combining predictors include 1) using a judgment-based or less formal cross-validation based method to choose non-negative weights summing to 1 to average predictors together to make a single predictor and 2) employing a regression version of "boosting." The basic story of SEL boosting is:

Approach the problem by trying to making successive sequential improvements on an approximation to the conditional mean function. This proceeds by

1. fitting and making residuals,
2. fitting to residuals as a response,
3. correcting the current approximation to the conditional mean by adding a fraction (say $\nu \in [0,1]$) of the function fitted to residuals from step 2, thereby making new residuals,
4. either returning to 2. or stopping, depending upon some sensible criterion that intends to prevent over-fitting.

The value $\nu$ is called the "learning rate."