Boosting, Bayes Nets, and Belief

A "What's the Big Idea?" Talk

Eric Bauer

February 24, 2000

What is Boosting?

Boosting is a technique for increasing classification accuracy due to Freund and Schapire.

Take a *weak learner*, a classification algorithm that (w.h.p.) has performance guaranteed to be significantly better than random guessing.

Extend it to do *weighted classification*: assign weights to training cases and calculate weighted error instead of simply counting.

For a given training set, initialize all weights to 1 and repeat the following until satisfied:

1. Use the weak learner to generate a new hypothesis.

2. Calculate the hypothesis' weighted error, and use this to generate a weight for the hypothesis.

3. Increase the weights of misclassified cases by an exponential factor; similarly, decrease weights of correctly classified cases by an exponential factor.

Afterwards, classify cases by weighted voting using all generated hypotheses.

In practice, boosting avoids overfitting: test error continues to drop well after training error hits zero. This is cool.

Why this is so has yet to be fully understood.

- Schapire et al: margins
- Friedman, Hastie, and Tibshirani: additive logistic regression
- Lafferty: generalized stepwise logistic regression using Bregman divergences

But at the end of the day, it's still just classification.

What's a Bayesian network?

Formally, a Bayesian network for a given distribution is a DAG that tells us a set of conditional independence relationships that MUST hold in that distribution.

Intuitively, we get independence for free in BNs and have to work to add dependence until all remaining independencies are legitimate.

In fact, learning BNs explicitly works this way, albeit heuristically.

Pearl asks a hard question about these kinds of models, namely, “How well can Bayesian networks represent the type of dependencies induced by probabilistic models?”

He also answers it - “The answer is not encouraging. Clearly, no graphical representation can distinguish connectivity between sets from connectivity among their elements. Separation between two sets of vertices is defined in terms of pairwise separation between their corresponding individual elements.”

These conditional independence relationships are quite useful, as they suggest a decomposition of the parameter space. Moreover, *if* the underlying world obeys such relationships, the decomposition teaches us something about the world.

But how big is the assumption that reality can be modeled using BNs?

What about trying probabilistic decompositions that do not themselves depend quite so much on independence?

IDEA: Boosting essentially produces a mixture model for classification; could we use something like it to produce a mixture model for densities?

To do this, we need to either

- Extend weak classification learners to weak density learners.

- Use classifiers to estimate densities.

The latter appears to be much easier.

Instead of getting independence for free, we would get dependence for free and (mostly) avoid trying to find independencies.

Tricky bits:

- Error becomes slippery when switching from labels to densities...
- ...which, in turn, makes reweighting more difficult.
- It's unclear how weights should relate to/affect the original data distribution.
- There may not be any weak density learners.
- Exact estimation probably requires exponential mixture components; how close can we get in a reasonable amount of time?
- Do we need to keep weights over the unobserved space?