"Super Learning" for Online Data

Jeremy Coyle, Sara Moore

March 19, 2013

Background

We will implement the Super Learner model stacking approach in an on-line machine learning setting. We expect that this approach will do at least as well as the best individual model.

Super Learner

- A wide variety of algorithms can be used to predict an outcome: simple regression, regularized regression (LOESS or ridge), SVM, etc. For any new prediction problem, it is not clear which approach will best capture the true relationship between the features and the outcome.
- Super Learner is an algorithm that combines other algorithms by minimizing the squared error loss (or other suitable loss function) of a linear combination of their predictions under cross-validation.
- van der Laan et al. (2003, 2006) proved that this approach will perform asymptotically as well or better to the best algorithm in the library of algorithms.

SuperLearner

Algorithm 1: Super Learner

Split the data into 10 folds
for $i \in 1 \ldots 10$ **do**
    Hold out observations in fold i as a validation set
    Use observations from the other 9 folds as the training set
    Using the training set:
    Fit library of predictive algorithms
    Using the validation set:
    Obtain predictions for each validation observation from each algorithm
end
Using the predictions on the validation set and the true outcome on the validation set, find the linear combination of the algorithms that minimizes the squared-error loss

Online Learning

- In many contexts, the amount of data generated is so large that it is not feasible to store it. Online machine learning, a paradigm in which models are fit to data coming in a "stream", eliminates the need to store all past observations.
- For each block of incoming data, the fit of the model is updated and then the data are discarded. Before the fit is updated, each new block can also be used to assess the performance of the model (a sort of cross-validation).
- Any algorithms that can be fit with stochastic gradient descent can be easily used in an online context.

Proposed approach

Algorithm 2: Online Super Learner

Using the first data block:
Fit each algorithm in the library of algorithms
Using the second data block:
Predict the outcome of each observation using each algorithm
Find the best linear combination of algorithms for predicting the true outcome
for *each subsequent data block* **do**
    Obtain predictions for each observation from each algorithm
    Calculate the risk for this data block (Mean Squared Error)
    Update the linear combination of algorithms
    Update the individual algorithm fits
end

Analysis Plan

- Algorithm Library:
  - OLS
  - ridge
  - LASSO
  - boosting
  - SVM
  - other approaches?
- Performance assessment:
  - Model fit will be assessed using MSE/AUC using the predictions from each block (before updating the models).
  - Performance of the individual algorithms and the overhead to combine them will be assessed.

Data and tools

- We will use the Twitter feed data to simulate an online learning context. Blocks will be used once and then discarded.
- Time permitting, we will apply our algorithm to data obtained from the real Twitter streaming API.
- We will use emoticons as a proxy for sentiment and then do sentiment analysis.
- Our online Super Learner will be implemented in Scala and BIDMat. Results will be summarized in R.