Feature Engineering (etc.) Perspective and CV

Stephen Vardeman

Analytics Iowa LLC

ISU Statistics and IMSE

The point of "feature engineering"

- Feature engineering/data transformation replaces input $\mathbf{x}$ with $T(\mathbf{x})$
  - This *cannot* increase the amount of available information
  - In fact, unless the transform is 1-1, it potentially reduces the amount of information available
  - (The statistical theory of sufficiency is about what kinds of non-1-1 transforms do not cause loss of information)
  - So transformation is *not* about increasing available information, but is rather about putting it into forms that are natural and effective inputs to standard prediction methods (and increasing the usefulness of these methods beyond application to the "raw" data)
  - Typically, linear transforms do nothing to increase useability or range of effectiveness (linear transforms between linear spaces don't fundamentally change forms of training sets) --- so most often one considers nonlinear transforms

"Pre-processing" and transformations

- It's important to consider what is meant by the notation $T(\mathbf{x})$, namely that what is done to an input depends only on that data case
- Where a training set might be standardized, or a random subset of it might be used to make potentially useful features (e.g., like approximate partial likelihood ratios for categorical inputs to classification), or … notation like $T(\mathbf{T}, \mathbf{x})$ that recognizes the dependence of the transform on the whole training set is appropriate
- This must be handled carefully/appropriately in cross-validation of a prediction methodology
- If ultimately a predictor is to be built on *N* vectors $\left(T(\mathbf{T}, \mathbf{x}_i), y_i\right)$, then for each remainder $\mathbf{T} - \mathbf{T}_k$ (thus *K* times) one must build a predictor on the vectors $\left(T(\mathbf{T} - \mathbf{T}_k, \mathbf{x}_i), y_i\right)$ in the remainder and test on data cases in the fold $\mathbf{T}_k$

Feature selection or predictor making?

- This discussion should make clear that “feature selection” and “predictor choice and fitting” cannot really be logically separated … they are just slightly different aspects of the single (typically iterative) process of making a predictor from a training set
- They both affect predictor performance and neither can be treated as outside the process
- In recognition of that, (as always) for approximating a plausible test error via cross-validation **whatever one will ultimately do in the entire training set to make a predictor must be redone in every remainder and applied to the corresponding fold**