Forward Stagewise Additive Modeling

Many methods make use of additive models having the general form

$$f(\vec{x}) = \sum_{m=1}^{M} \beta_m \, b(\vec{x}; \gamma_m),$$

where the β_m are coefficients and the $b(\vec{x}; \gamma_m)$ are basis functions of the multivariate argument, $\vec{x}$, characterized by a set of parameters, with the γ_m being the parameter sets for the various basis functions. For example, an additive MARS model is of this type, with the $b(\vec{x}; \gamma_m)$ being hockey stick basis functions, with the γ_m identifying the featured predictor, the location of the knot, and whether it's a primary or mirror-image hockey stick.

One might desire to fit such a model by minimizing a loss function averaged over the training data. For example, one could seek values for the β_m and the γ_m to minimize

$$\frac{1}{n} \sum_{i=1}^{n} L\left(y_i, \ \sum_{m=1}^{M} \beta_m \, b(\vec{x}_i; \gamma_m) \right),$$

perhaps using

$$L(y, f(\vec{x})) = (f(\vec{x}) - y)^2$$

(the squared-error loss fn). (Overfitting would be controlled by selecting M appropriately, given the basis functions used and the sample size of the training data.)

However, if things are sufficiently complex, the optimization may be difficult and computationally costly. In such cases, a simple alternative strategy, known as *forward stagewise additive modeling*, can be attractive when it is feasible to rapidly solve the subproblem of fitting just a single basis fn — i.e., if it isn't terribly difficult to determine the values of β and γ which minimize

$$\sum_{i=1}^{n} L(y_i, \beta b(\vec{x}_i; \gamma))$$

(where the factor of $\frac{1}{n}$ has been omitted since it does not affect the solution).

Forward stagewise additive modeling approximates the solution to the original minimization problem by sequentially adding new basis function terms to the expansion without adjusting the parameters and coefficients of those which have already been added. At the $m^{\underline{th}}$ step, one solves for the optimal basis fn, $b(\vec{x};\gamma_m)$, and corresponding coefficient, β_m, to go with the current expansion, $f_{m-1}(\vec{x})$, so that

$$f_m(\vec{x}) = f_{m-1}(\vec{x}) + \beta_m\, b(\vec{x};\gamma_m)$$

is the best choice for $f_m(\vec{x})$ having such an additive form. (Best is with regard to minimizing the overall loss for the training data.)

For example, with the squared-error loss fn, to obtain

$$f_1(\vec{x}) = \beta_1 \, b(\vec{x}; \gamma_1),$$

one determines the values of β and γ which minimize

$$\sum_{i=1}^{n} (y_i - \beta \, b(\vec{x}_i ; \gamma))^2.$$

For subsequent terms, given

$$f_{m-1}(\vec{x}) = \sum_{k=1}^{m-1} \beta_k \, b(\vec{x}; \gamma_k),$$

to obtain β_m and γ_m, one determines the values of β and γ which minimize

$$\sum_{i=1}^{n} (y_i - [f_{m-1}(\vec{x}_i) + \beta \, b(\vec{x}_i; \gamma)])^2$$

$$= \sum_{i=1}^{n} ([y_i - f_{m-1}(\vec{x}_i)] - \beta \, b(\vec{x}_i; \gamma))^2$$

$$= \sum_{i=1}^{n} (r_{m,i} - \beta \, b(\vec{x}_i ; \gamma))^2,$$

where $r_{m,i} = y_i - f_{m-1}(\vec{x}_i)$ is simply the $i^{\underline{th}}$

residual resulting from the current model — the error of the current model on the i^{th} observation. So once the values of the $r_{m,i}$ are determined at the start of the m^{th} stage, the problem of obtaining β_m and γ_m is of the same form as the problem of obtaining β_1 and γ_1.

Since the squared-error loss fn is generally not a good choice for classification, and may not be the best choice for regression, it is of interest to determine how other appropriate loss functions can be used with forward stagewise additive modeling.