Making Quantitative Features for Classification

Stephen Vardeman

Analytics Iowa LLC

ISU Statistics and IMSE

Dummy variables ("one hot encoding")

Often one or more inputs x_j in a classification problem are categorical. Numerical features are more directly handled. So consider construction of numerical features from categorical variables in K-class models.

Suppose that some part of the input variable $\mathbf{x}$, say $\mathbf{x}^c$ (consisting of some coordinates of the original p inputs), is categorical with M possible values $\{\mathbf{x}_1^c, \mathbf{x}_2^c, \ldots, \mathbf{x}_M^c\}$. ($M$ will be the product of the numbers of possible values for the individual inputs represented in $\mathbf{x}^c$. What follows could be applied separately to multiple different categorical parts of $\mathbf{x}$.)

One way to represent $\mathbf{x}^c$ is through a set of $M-1$ dummy variables

$$d_m = I\left[\mathbf{x}^c = \mathbf{x}_m^c\right]$$

But this is problematic, as the number of dummy variables explodes with both p and the number(s) of possible values of categorical inputs.

"Partial" optimal features

In the K-class classification model, the variable $\mathbf{x}^c$ inherits one of K possible conditional (on y) marginal distributions specified by

$$p\left(\mathbf{x}_m^c|k\right) = P\left[\mathbf{x}^c = \mathbf{x}_m^c|y = k\right]$$

For $s\left(\mathbf{x}_m^c\right) = \sum_{k=0}^{K-1} p\left(\mathbf{x}_m^c|k\right)$ the $K-1$ ratios

$$\mathcal{L}_1\left(\mathbf{x}^c\right) \equiv \frac{p\left(\mathbf{x}^c|1\right)}{s\left(\mathbf{x}^c\right)}, \mathcal{L}_2\left(\mathbf{x}^c\right) \equiv \frac{p\left(\mathbf{x}^c|2\right)}{s\left(\mathbf{x}^c\right)}, \ldots, \mathcal{L}_{K-1}\left(\mathbf{x}^c\right) \equiv \frac{p\left(\mathbf{x}^c|K-1\right)}{s\left(\mathbf{x}^c\right)}$$

encode all the information about class membership available in $\mathbf{x}^c$. If all one had available for classification was categorical input $\mathbf{x}^c$ and the distributions $p\left(\mathbf{x}|k\right)$ were known perfectly, an optimal feature vector derived from $\mathbf{x}^c$ would consist of these $K-1$ features. In practice one doesn't know the distributions $p\left(\mathbf{x}^c|k\right)$ perfectly, and only approximation to these $K-1$ features is possible.

Approximate "partial" optimal features

With N_k the count of training cases with $y = k$, one can estimate as

$$\widehat{p\left(\mathbf{x}_m^c | k\right)} = \frac{1}{N_k} \sum_{i=1}^{N} I\left[\text{case } i \text{ has } y_i = k \text{ and categorical response } \mathbf{x}_i^c = \mathbf{x}_m^c\right]$$

Raw ratios of these would suffer large variance. A way to reduce variance of ratios of estimates $\widehat{p\left(\mathbf{x}_m^c | k\right)}$ is to instead use something like

$$\widetilde{p\left(\mathbf{x}_m^c | k\right)} \equiv \frac{N_k \widehat{p\left(\mathbf{x}_m^c | k\right)} + \alpha}{N_k + M\alpha}$$

for some small $\alpha > 0$. (This produces a compromise between the class k empirical distribution of $\mathbf{x}^c$ across $\{\mathbf{x}_1^c, \mathbf{x}_2^c, \ldots, \mathbf{x}_M^c\}$ and a uniform distribution across the same set.)

Considerations for practice

A trade-off will often have to be made here regarding the size of M. The larger is M the more effective will be the "partial" optimal features and the less effectively will their empirical approximations represent them. A way to try to handle this issue in practice is to build sets of these features with a spectrum of values M and look for one that is overall most effective.

Further, some ISU experience with this idea seems to suggest that both making these features and fitting a predictor on the same training set produces hopelessly optimistic prediction of performance. (After all, the making of the $p\left(\mathbf{x}_m^c \middle| k\right)$ already involves the y_i. Subsequent fitting is more or less double use of them.) It seems that a training set may need to be split into "feature-making" and "fitting" parts in order to avoid over-fitting. (Cross-validation of *all*–including potential data splitting, choice of M, and feature making—is needed to empirically gauge likely performance on new cases.)