Graph-based Penalized Fitting/Smoothing and Semi-supervised Learning

Stephen Vardeman

Analytics Iowa LLC

ISU Statistics and IMSE

Semi-supervised learning

Consider N complete data cases $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$ and $M \geq 0$ additional data cases where only inputs $\mathbf{x}_{N+1}, \ldots, \mathbf{x}_{N+M}$ are available. There is no necessity here that $M > 0$, but it can be so in the event that predictions are desired at $\mathbf{x}_{N+1}, \ldots, \mathbf{x}_{N+M}$ whose values might not be in the training set. Where there are $M > 0$ genuine "unlabeled cases" whose inputs are assumed to come from the same mechanism as the inputs $\mathbf{x}_1, \ldots, \mathbf{x}_N$ and might be used to more or less "fill in" the relevant part of the input space not covered by the complete/labeled data cases, the terminology **semi-supervised learning** is sometimes used to describe the building of a predictor for y at all $N + M$ input vectors. The case $M = 1$ might be used to simply make a single prediction at a single input not exactly seen in a "usual" training set of N complete data pairs.

Set-up

Suppose that following the development of Section 2.4.3, one can make an adjacency matrix based on the $N+M$ input vectors,

$$\mathbf{S} = (s_{ij})_{\substack{i=1,\ldots,N+M \\ j=1,\ldots,N+M}} = \begin{pmatrix} \underset{N\times N}{\mathbf{S}_{\mathrm{L}}} & \underset{N\times M}{\mathbf{S}_{\mathrm{LU}}} \\ \underset{M\times N}{\mathbf{S}_{\mathrm{UL}}} & \underset{M\times M}{\mathbf{S}_{\mathrm{U}}} \end{pmatrix}$$

and corresponding Laplacian and symmetric normalized Laplacian matrices, respectively

$$\mathbf{L} = \begin{pmatrix} \underset{N\times N}{\mathbf{L}_{\mathrm{L}}} & \underset{N\times M}{\mathbf{L}_{\mathrm{LU}}} \\ \underset{M\times N}{\mathbf{L}_{\mathrm{UL}}} & \underset{M\times M}{\mathbf{L}_{\mathrm{U}}} \end{pmatrix} \text{ and } \mathbf{L}^* = \begin{pmatrix} \underset{N\times N}{\mathbf{L}^*_{\mathrm{L}}} & \underset{N\times M}{\mathbf{L}^*_{\mathrm{LU}}} \\ \underset{M\times N}{\mathbf{L}^*_{\mathrm{UL}}} & \underset{M\times M}{\mathbf{L}^*_{\mathrm{U}}} \end{pmatrix}$$

Graph-penalized fitting/smoothing

With

$$\underset{(N+M)\times 1}{\mathbf{Y}} = \begin{pmatrix} \underset{N\times 1}{\mathbf{Y}_{\mathrm{L}}} \\ \underset{M\times 1}{\mathbf{Y}_{\mathrm{U}}} \end{pmatrix}$$

one might wish to produce a vector of smoothed/fitted values $\underset{(N+M)\times 1}{\hat{\mathbf{Y}}}$ such that entries corresponding to input vectors with large adjacencies tend to be alike. This is possible in way highly reminiscent of the material in Sections 5.1 and 5.3. For $\mathbf{v} \in \Re^{N+M}$ written as

$$\underset{(N+M)\times 1}{\mathbf{v}} = \begin{pmatrix} \underset{N\times 1}{\mathbf{v}_{\mathrm{L}}} \\ \underset{M\times 1}{\mathbf{v}_{\mathrm{U}}} \end{pmatrix}$$

consider the optimization problem in $\Re^{N+M}$

$$\underset{\mathbf{v}\in\Re^{N+M}}{\text{minimize}} \left(\left(\mathbf{Y}_{\mathrm{L}} - \mathbf{v}_{\mathrm{L}}\right)' \left(\mathbf{Y}_{\mathrm{L}} - \mathbf{v}_{\mathrm{L}}\right) + \lambda \mathbf{v}'\mathbf{L}\mathbf{v} \right) \qquad (1)$$

Effects of graph-penalized fitting

The developments of Section 2.4.3 show that upon expanding $\mathbf{v}$ in terms of the $N + M$ (orthonormal) eigenvectors of $\mathbf{L}$ (or $\mathbf{L}^*$) it follows that components of $\mathbf{v}$ that are multiples of late eigenvectors (ones with small eigenvalues)

1. have similar entries for cases with large adjacencies, and
2. are relatively lightly penalized in the minimization.

This strongly suggests that solutions to the optimization problem (1) will provide smoothed prediction vectors $\hat{\mathbf{Y}}$ where entries with corresponding inputs with large adjacencies are similar.

Ryan and Culp results

Recent work of Culp and Ryan provides theory, methods, and software for solving the problem (1) and many nice generalizations of it (including consideration of losses other than SEL that produce methods for classification problems). Here we will provide (only) the explicit solution that is available for the SEL problem.

It turns out that the problem (1) and generalizations of it separate nicely into two parts. That is

$$\hat{\mathbf{Y}}_{\mathrm{U}}^{\mathrm{opt}} = -\mathbf{L}_{\mathrm{U}}^{-1}\mathbf{L}_{\mathrm{UL}}\hat{\mathbf{Y}}_{\mathrm{L}}^{\mathrm{opt}} \tag{2}$$

(or the same with $\mathbf{L}^*$s replacing $\mathbf{L}$s) where $\hat{\mathbf{Y}}_{\mathrm{L}}^{\mathrm{opt}} = \mathbf{v}_{\mathrm{L}}$ solving

$$\underset{\mathbf{v}_{\mathrm{L}} \in \Re^N}{\text{minimize}} \left(\left(\mathbf{Y}_{\mathrm{L}} - \mathbf{v}_{\mathrm{L}}\right)' \left(\mathbf{Y}_{\mathrm{L}} - \mathbf{v}_{\mathrm{L}}\right) + \lambda \mathbf{v}_{\mathrm{L}}' \tilde{\mathbf{L}}_{\mathrm{L}} \mathbf{v}_{\mathrm{L}} \right) \tag{3}$$

for $\tilde{\mathbf{L}}_{\mathrm{L}} = \mathbf{L}_{\mathrm{L}} - \mathbf{L}_{\mathrm{LU}}\mathbf{L}_{\mathrm{U}}^{-1}\mathbf{L}_{\mathrm{UL}}$ (or, again, the same with $\mathbf{L}^*$s replacing $\mathbf{L}$s).

SEL solution

The matrix optimization problem (3) is familiar and its solution a simple consequence of vector calculus

$$\hat{\mathbf{Y}}_{\mathrm{L}}^{\mathrm{opt}} = \left(\mathbf{I} + \lambda \tilde{\mathbf{L}}_{\mathrm{L}}\right)^{-1} \mathbf{Y}_{\mathrm{L}}$$

This is exactly parallel to the results in Sections 5.1 and 5.3. $\left(\mathbf{I} + \lambda \tilde{\mathbf{L}}_{\mathrm{L}}\right)^{-1}$ (and its starred version) is a smoother/shrinker matrix. (Further, the matrix $-\mathbf{L}_{\mathrm{U}}^{-1}\mathbf{L}_{\mathrm{UL}}$ in display (2) and its starred version are stochastic matrices and entries of $\hat{\mathbf{Y}}_{\mathrm{U}}^{\mathrm{opt}}$ are averages of the elements of $\hat{\mathbf{Y}}_{\mathrm{L}}^{\mathrm{opt}}$.) Presumably, in an application λ can be chosen by cross-validation using the labeled cases to make a CVMSPE.