SVMs Part 3C: Function Space Geometry

Stephen Vardeman

Analytics Iowa LLC

ISU Statistics and IMSE

Generalities

Another line of argument produces a SVM in a way that connects it to the *geometry* of support vector classification in $\Re^p$. Input feature vectors map to an abstract function space $\mathcal{A}$ via

$$T(\mathbf{x})(\cdot) = \mathcal{K}(\mathbf{x}, \cdot)$$

Subsequent to this mapping, all can be done using the abstract linear space structure. One is really defining a classifier with inputs in $\mathcal{A}$, and application of the support vector classifier argument can be made in terms of the geometry of $\mathcal{A}$. "Linear classification" in $\mathcal{A}$ is the analogue of support vector classification in $\Re^p$ if one starts from geometric motivation like that of the support vector classifier. One seeks a "unit vector" (now in $\mathcal{A}$) and a constant so that inner products of transformed data case inputs with the unit vector plus the constant, when multiplied by the y_i, maximize a margin subject to some (relaxed) constraints.

Function space formulation

All this is writable in terms of $\mathcal{A}$. That is, one wishes to

$$\underset{\substack{U \in \mathcal{A} \text{ with } \|U\|_{\mathcal{A}} = 1 \\ \text{and } \beta_0 \in \Re}}{\text{maximize}} \quad M$$

$$\text{subject to} \begin{cases} y_i \left(\langle T(\mathbf{x}_i), U \rangle_{\mathcal{A}} + \beta_0 \right) \geq M (1 - \xi_i) \quad \forall i \\ \text{for some } \xi_i \geq 0 \text{ with } \sum_{i=1}^N \xi_i \leq C \end{cases}$$

This is equivalent to the problem

$$\underset{\substack{V \in \mathcal{A} \\ \text{and } \beta_0 \in \Re}}{\text{minimize}} \quad \frac{1}{2} \|V\|_{\mathcal{A}}^2 \quad \text{subject to} \begin{cases} y_i \left(\langle T(\mathbf{x}_i), V \rangle_{\mathcal{A}} + \beta_0 \right) \geq (1 - \xi_i) \quad \forall i \\ \text{for some } \xi_i \geq 0 \text{ with } \sum_{i=1}^N \xi_i \leq C \end{cases}$$

Linear combinations of training case inputs

Then either because optimization over all of $\mathcal{A}$ looks too hard, or because some "Representer Theorem" says that it is enough to do so, one might back off from optimization over $\mathcal{A}$ to optimization over the subspace spanned by the set of N elements $T(\mathbf{x}_i)$. Then writing $V = \sum_{i=1}^{N} \beta_i T(\mathbf{x}_i)$ so that

$$\frac{1}{2}\|V\|_{\mathcal{A}}^2 = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\beta_i\beta_j\left\langle T(\mathbf{x}_i), T(\mathbf{x}_j)\right\rangle_{\mathcal{A}} = \frac{1}{2}\boldsymbol{\beta}'\mathbf{K}\boldsymbol{\beta}$$

(again, $\mathbf{K}$ is the Gram matrix) the optimization problem becomes

$$\underset{\substack{\boldsymbol{\beta}\in\Re^N \\ \text{and } \beta_0\in\Re}}{\text{minimize}} \quad \frac{1}{2}\boldsymbol{\beta}'\mathbf{K}\boldsymbol{\beta} \quad \text{subject to} \begin{cases} y_i\left(\boldsymbol{\beta}'\mathbf{K}_i + \beta_0\right) \geq (1-\xi_i) \quad \forall i \\ \text{for some } \xi_i \geq 0 \text{ with } \sum_{i=1}^{N}\xi_i \leq C \end{cases}$$

where $\mathbf{K}_i$ is the ith column of the Gram matrix.

Non-linearity in the original input space

For $\boldsymbol{\beta}^{\text{opt}}$ and β_0^{opt} solutions to the optimization problem and

$$V^{\text{opt}} = \sum_{i=1}^{N} \beta_i^{\text{opt}} T(\mathbf{x}_i)$$

the voting function for the **linear** classifier in $\mathcal{A}$ is (for argument $W \in \mathcal{A}$)

$$\langle W, V^{\text{opt}} \rangle_{\mathcal{A}} + \beta_0^{\text{opt}}$$

The corresponding voting function for the derived **non-linear** classifier on $\Re^p$ is

$$\langle T(\mathbf{x}), V^{\text{opt}} \rangle_{\mathcal{A}} + \beta_0^{\text{opt}} = \sum_{i=1}^{N} \beta_i^{\text{opt}} \mathcal{K}(\mathbf{x}, \mathbf{x}_i) + \beta_0^{\text{opt}}$$

something very similar to the heuristic application of the "kernel trick." The question is whether it is exactly equivalent to the use of "the trick."

Geometry and the kernel trick

The problem solved by $\boldsymbol{\beta}^{\text{opt}}$ and β_0^{opt} is equivalent for some $\lambda \geq 0$ to

$$\underset{\substack{\boldsymbol{\beta} \in \Re^N \\ \text{and } \beta_0 \in \Re}}{\text{minimize}} \quad \frac{1}{2}\boldsymbol{\beta}' \mathbf{K} \boldsymbol{\beta} + \lambda \sum_{i=1}^{N} \xi_i \ \text{ subject to } \begin{cases} y_i \left(\boldsymbol{\beta}' \mathbf{K}_i + \beta_0 \right) \geq (1 - \xi_i) \quad \forall i \\ \text{for some } \xi_i \geq 0 \end{cases}$$

Comparison of this to the first display on slide 7 in the 1304 deck and consideration of the argument that follows it then shows that there is a choice of C^* for which when using kernel $(1/C^*)^2 \mathcal{K}$ the heuristic/"kernel trick" method produces a solution to the present function-space-support-vector-classifier problem. This is the same circumstance as in the penalized-fitting function-space-optimization argument. (The "kernel trick" applied to kernel $\mathcal{K}$ with cost parameter C^* solves the present geometric optimization problem applied to kernel $(C^*)^2 \mathcal{K}$ with cost parameter C^*.)