Kernel regularized least squares (KRLS)

Zalán Bodó

1 Least squares classification

In least squares classification (or regression) we are searching for a hyperplane separating positive and negative examples such that the labels output by the classifier and the true labels to be as close to each other as possible, in terms of squared difference. That is

$$\min_{\boldsymbol{w}} \quad \frac{1}{\ell}\|\boldsymbol{X}'\boldsymbol{w}-\boldsymbol{y}\|_2^2+\lambda\|\boldsymbol{w}\|_2^2$$

where ℓ denotes the number of training examples (i.e. number of points placed in the columns of $\boldsymbol{X}$) $\lambda\|\boldsymbol{w}\|_2^2$ is a regularization tag. The solution is given by

$$\boldsymbol{w}^*=\left(\boldsymbol{X}\boldsymbol{X}'+\lambda\ell\boldsymbol{I}\right)^{-1}\boldsymbol{X}\boldsymbol{y}$$

2 Kernelization

In this section we show a method to kernelize, different from the one shown usually in other manuals.

To kernelize RLS we need to form dot products of the examples. Our classifier is given by

$$f(\boldsymbol{x})=\boldsymbol{w}'\boldsymbol{x}=\boldsymbol{y}'\boldsymbol{X}\left(\boldsymbol{X}\boldsymbol{X}'+\lambda\ell\boldsymbol{I}\right)^{-1}\boldsymbol{x}$$

and the problem is that the input variables do not form dot products, neither on the left and right hand side of the expression, nor inside the paranthesis. Therefore, we will use a formula that rewrites $\left(\boldsymbol{X}\boldsymbol{X}'+\lambda\ell\boldsymbol{I}\right)^{-1}\boldsymbol{X}$ in another form. Consider the Searle formula [1]:

$$(\boldsymbol{A}+\boldsymbol{B}\boldsymbol{B}')^{-1}\boldsymbol{B}=\boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{I}+\boldsymbol{B}'\boldsymbol{A}^{-1}\boldsymbol{B})^{-1}$$

Applying it to our case we obtain:

$$\begin{aligned}\left(\boldsymbol{X}\boldsymbol{X}'+\lambda\ell\boldsymbol{I}\right)^{-1}\boldsymbol{X} &= \frac{1}{\lambda\ell}\boldsymbol{X}\left(\boldsymbol{I}+\frac{1}{\lambda\ell}\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\\ &= \boldsymbol{X}(\lambda\ell\boldsymbol{I}+\boldsymbol{X}'\boldsymbol{X})^{-1}\end{aligned}$$

Substituting this back into our classifier we get

$$f(\boldsymbol{x}) = \boldsymbol{w}'\boldsymbol{x} = \boldsymbol{y}'(\lambda\ell\boldsymbol{I} + \boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{x}$$

which now can be kernelized. If $\boldsymbol{K}$ denotes the kernel or Gram matrix of the training data, and $\boldsymbol{k}_{\boldsymbol{x}}$ is the vector $[k(\boldsymbol{x}_1, \boldsymbol{x}), k(\boldsymbol{x}_2, \boldsymbol{x}), \ldots, k(\boldsymbol{x}_N, \boldsymbol{x})]'$, by the *kernel trick*[1] we can write

$$f(\boldsymbol{x}) = \boldsymbol{y}'(\lambda\ell\boldsymbol{I} + \boldsymbol{K})^{-1}\boldsymbol{k}_{\boldsymbol{x}}$$

3 Kernelization using the representer theorem

The optimization problem can be written as

$$\min_f \quad \frac{1}{\ell}\sum_{i=1}^{\ell}(f(\boldsymbol{x}_i) - y_i)^2 + \lambda\|f\|_{\boldsymbol{K}}^2$$

where $\|f\|_{\boldsymbol{K}}^2$ denotes the RKHS norm of the decision function. The representer theorem [2] says that the solution can be written as

$$f(\cdot) = \sum_{i=1}^{\ell}\alpha_i k(\boldsymbol{x}_i, \cdot)$$

Thefore we can rewrite the optimization problem as

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{\ell}\|\boldsymbol{K}\boldsymbol{\alpha} - \boldsymbol{y}\|_2^2 + \lambda\|f\|_{\boldsymbol{K}}^2$$

In order to solve the optimization problem we need to express the regularization tag, that is

$$\begin{aligned}
\|f\|_{\boldsymbol{K}}^2 &= \langle f, f\rangle_{\boldsymbol{K}} \\
&= \left\langle \sum_{i=1}^{\ell}\alpha_i k(\boldsymbol{x}_i, \cdot), \sum_{j=1}^{\ell}\alpha_j k(\boldsymbol{x}_j, \cdot)\right\rangle_{\boldsymbol{K}} \\
&= \sum_{i=1}^{\ell}\sum_{j=1}^{\ell}\alpha_i\alpha_j \left\langle k(\boldsymbol{x}_i, \cdot), k(\boldsymbol{x}_j, \cdot)\right\rangle_{\boldsymbol{K}} \\
&= \sum_{i=1}^{\ell}\sum_{j=1}^{\ell}\alpha_i\alpha_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\
&= \boldsymbol{\alpha}'\boldsymbol{K}\boldsymbol{\alpha}
\end{aligned}$$

[1] The kernel trick actually refers to the application of a feature mapping ϕ to the points without specifying the mapping. The mapping is given by the kernel function, $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)'\phi(\boldsymbol{x}_j)$, where the kernel function can be any semi-positive definite, symmetric and continuous function.

Thus, if we rewrite the optimization problem,

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{\ell}\|\boldsymbol{K}\boldsymbol{\alpha} - \boldsymbol{y}\|_2^2 + \lambda \boldsymbol{\alpha}' \boldsymbol{K} \boldsymbol{\alpha}$$

by setting the derivative of the objective function to zero and expressing $\boldsymbol{\alpha}$ we obtain

$$\boldsymbol{K}\boldsymbol{\alpha} - \boldsymbol{y} + \lambda \ell \boldsymbol{\alpha} = 0$$

from which

$$\boldsymbol{\alpha} = (\boldsymbol{K} + \lambda \ell \boldsymbol{I})^{-1} \boldsymbol{y}$$

Substituting back the result into the decision function we obtain the same formula as earlier:

$$f(\boldsymbol{x}) = \boldsymbol{y}' (\boldsymbol{K} + \lambda \ell \boldsymbol{I})^{-1} \boldsymbol{k}_{\boldsymbol{x}}$$

References

[1] K. B. Petersen and M. S. Pedersen. The matrix cookbook, November 2012.

[2] B. Schölkopf and A. J. Smola. *Learning with Kernels*. The MIT Press, Cambridge, MA, 2002.