Investigating Relevance Vector Machine for Sparse Learning

Artem Los (arteml@kth.se), Clement Morin (clemor@kth.se),
Kevin Dallatorre (kevindt@kth.se), Remi Lacombe (rlacombe@kth.se)

Saturday 20th January, 2018

I. Introduction

For supervised learning, support vector machine (SVM) is a state-of-the art way of performing classification and regression. The idea behind SVM is to *minimize the training set error* of a target function (define) while simultaneously maximizing the margin between the two classes [1].

However, SVM's have four problems that can be solved using the Relevance Vector Machines (RVM). First, SVM's do not give us a probabilistic answer, which is useful when we want to apply the notion of uncertainty in the classifier (i.e. similar to error bars). An SVM classification will produce a hard binary decision whereas the SVM regression will produce a point estimate. Secondly, the required number of kernel functions will grow steeply with the size of the training set. Thirdly, a constant C (aka error/margin trade off) has to be estimated. Finally, the kernel functions require Mercer's condition to be satisfied. [1]

The RVM solves all the problems mentioned above. The idea is to associate a prior to each weight (governed by a hyper-parameter for each weight), which are estimated from the data. As a result, fewer kernel functions are required than for SVMs, the error is in general smaller for RVMs than SVMs and the number of relevance vectors needed (similar to support vectors in SVMs) is smaller. On the contrary, it takes longer time to train an RVM than an SVM. [1]

II. Method

RVMs can be thought of as SVMs that use a Bayesian treatment. Instead of providing direct answers, RVMs return probabilistic predictions. We start of by computing the hyperparameters α, σ^2 of the priors for each weight (all of the priors being independent) based on the data. The roots of the derivatives or the Expectation-maximization (EM) method can be used to find optimal values of α, σ^2 (i.e. those values that maximize the *evidence*). This is what constitutes the learning process.

A. Terminology

- $\boldsymbol{x}$ – the input vector, consisting of $(x_1, \dots x_N)$.
- $\boldsymbol{X}$ – the collection of input vectors such that the nth row is x_n^T.
- $\boldsymbol{t}$ – the target values vector.
- $\boldsymbol{w}$ – the set of weights, consisting of $(w_1, \dots w_N)$.

B. Regression

Regression with RVMs consists of three main steps: initialization, learning and prediction.

1) Assigning a prior to each weight (initialization): Priors control the importance of a given basis function. The first step is to assign a prior to each weight. Note, the hyperparameters of each weight are independent of each other [1]. The parameter α^{-1} is one of the parameters that we aim to optimize later on. The other parameter is the mean μ, although it is set to zero in this step.

The prior of weight w_i is of the form

$$p(w_i|\alpha_i) = \mathcal{N}(w_i|0, \alpha_i^{-1}) \tag{1}$$

When we have multiple data points, the prior $p(\boldsymbol{w}|\boldsymbol{\alpha})$ is simply the product of all the priors of each weight.

2) Optimizing the hyperparameters (learning): In order to keep updating our belief about the weights given new data points, we need to compute the posterior. Since both the likelihood and the prior are Gaussian, the resulting posterior

$$p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2) = \mathcal{N}(m, \Sigma) \tag{2}$$

is also Gaussian, whose parameters (i.e. m and Σ) can be found given existing formulas [2].

The posterior can be simplified by integrating out the weights, which leads to the marginal likelihood $p(\boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2)$.

Now, the aim is to estimate α and σ^2 that maximize the marginal likelihood above [2], which can be accomplished in two ways. Either, we can set the derivative of the log marginal likelihood to zero, in order to obtain α and σ^2, or use expectation maximization method [1,2]. The new estimation of hyperparameters can then be used to estimate the mean and the covariance of the posterior [2]. The re-estimation process repeats until convergence [3] [1].

3) Prediction: Once the optimal values of α and σ^2 have been found, we can use them to find the predictive distribution over t [2]. Given a new input x, the predictive distribution $p(t|\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2)$ is also a Gaussian, so there exists a closed-form formula to compute the mean and the covariance [2].

C. Classification

Classification is very similar to regression, with the differences outlined below. Method wise it is the same as described in regression unless stated otherwise.

1) Two-class problem: In order to classify input values to discrete classes, the logistic sigmoid function can be used [2]. It has the property that its range is $[0, 1]$.

[1] In contrast to SVMs, where a single shared hyperparameter is used [2].

[2] This is known as type-2 maximum likelihood [2].

[3] When an appropriate convergence criterion is satisfied [1].

2) Approximating marginal likelihood: In contrast to regression, it is not possible to integrate out the weight parameter. Instead, Laplacian approximation can be used.

TODO: we can estimate the weights.

III. Result

IV. Discussion

V. References

FIX REFERENCES

1 Original article (fix ref).

2 http://users.isr.ist.utl.pt/~wurmd/Livros/school/Bishop%20-%20Pattern%20Recognition%20And%20Machine%20Learning%20-%20Springer%20%202006.pdf