Chapter 4

Sparse Bayesian Learning

Sparse Baysian Learning [5] is a general Bayesian framework within supervised Machine Learning. It can be applied to both regression and classification tasks. The *Relevance Vector Machine*, or *RVM*, is a particular specialisation of the Sparse Bayesian Learning model which has identical functional form to the Support Vector Machine (SVM). However, the RVM comes with a number of key advantages over the SVM. The solution produced by a RVM is typically much sparser than the solution by a comparable SVM. Furthermore, the RVM is a probabilistic model and as such, allows us to estimate error bounds in its predictions.

In this chapter, we will derive the Sparse Bayesian Learning model for regression. We will summarize both the original inference algorithm [5] and also the faster "Sequential Sparse Bayesian Learning Algorithm" [6].

4.1 Model Specification

We are given a data set of N input vectors $\{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ and their associated *targets* $\{y^{(i)}\}_{i=1}^{N}$. The input vectors live in D-dimensional space, $\boldsymbol{x} \in \mathbb{R}^D$. The targets are real values, $y \in \mathbb{R}$. [1]

We model the data using a linearly-weighted sum of M fixed basis functions $\{\phi_j(\cdot)\}_{j=1}^{M}$ and base our predictions on the function $f(\cdot)$ defined as

$$f(\boldsymbol{x};\boldsymbol{w}) = \sum_{j=1}^{M} w_j \phi_j(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}) \tag{4.1}$$

where $\boldsymbol{w} = [w_1, \ldots, w_M]^T$ and $\boldsymbol{\phi}(\cdot) = [\phi_1(\cdot), \ldots, \phi_M(\cdot)]^T$. Using a large number M of non-linear basis functions $\phi_j : \mathbb{R}^D \to \mathbb{R}$ allows for a highly flexible model.

[1] When using the Sparse Bayesian model for regression, we assume the targets are real-valued. It is also possible to use the model for classification in which cased the targets are assumed to be discrete class labels.

The *Relevance Vector Machine*, or RVM, is a specialisation of the Sparse Bayesian Learning model in which the basis functions take the form of *kernel functions*

$$\phi_j(\cdot) \equiv K\left(\cdot, \boldsymbol{x}^{(j)}\right).$$

This defines a basis function for each training data point $\boldsymbol{x}^{(i)}$. Typically, we also include an additional *bias* term $\phi_0(\cdot) \equiv 1$, so that $M = N+1$. The RVM has identical functional form to the popular Support Vector Machine (SVM), but superior properties. It typically gives sparser solutions than the SVM and has the additional advantage of providing confidence measures for its predictions.

However, in the following derivation, we will stick to the case of general basis functions $\phi_j : \mathbb{R}^D \to \mathbb{R}$. Thus M need not equal $N+1$ and may, in fact, be a lot larger.

To train the model (4.1), i.e. find values for $\boldsymbol{w}$ that are optimal in some sense, we make the standard assumption that our training data are samples from the model with additive noise:

$$\begin{aligned} y^{(i)} &= f(\boldsymbol{x}^{(i)}; \boldsymbol{w}) + \varepsilon^{(i)} \\ &= \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}^{(i)}) + \varepsilon^{(i)} \qquad i = 1, \ldots, N. \end{aligned} \tag{4.2}$$

The errors $\{\varepsilon^{(i)}\}_{i=1}^N$ are assumed to be independent samples from a zero-mean Gaussian distribution with variance σ^2

$$p(\varepsilon^{(i)}) = \mathcal{N}(\varepsilon^{(i)} \,|\, 0, \sigma^2) \qquad i = 1, \ldots, N. \tag{4.3}$$

Combining equation (4.2) with equation (4.1), we may express the model for the complete data using matrix notation:

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{w} + \boldsymbol{\varepsilon} \tag{4.4}$$

where $\boldsymbol{\varepsilon} = \left[\varepsilon^{(1)}, \cdots, \varepsilon^{(N)}\right]^T$. The $N \times M$ matrix $\boldsymbol{\Phi}$ is known as the design matrix. The ith row of $\boldsymbol{\Phi}$ is given by $\boldsymbol{\phi}(\boldsymbol{x}^{(i)})^T$. The jth column of $\boldsymbol{\Phi}$ is given by $\boldsymbol{\phi}_j = \left[\phi_j\left(\boldsymbol{x}^{(1)}\right), \cdots, \phi_j\left(\boldsymbol{x}^{(N)}\right)\right]^T$, which is also referred to as the jth *basis vector*. Thus

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\phi}_1 & \cdots & \boldsymbol{\phi}_M \end{bmatrix} = \begin{bmatrix} \boldsymbol{\phi}(\boldsymbol{x}^{(1)})^T \\ \vdots \\ \boldsymbol{\phi}(\boldsymbol{x}^{(N)})^T \end{bmatrix}$$

Combining equation (4.4) and equation (4.3), we find that the complete data likelihood function is given by

$$\begin{aligned} p\left(\boldsymbol{y}\,|\,\boldsymbol{w},\sigma^2\right) &= \mathcal{N}\left(\boldsymbol{y}\,|\,\boldsymbol{w},\sigma^2\boldsymbol{I}_M\right) \\ &= (2\pi\sigma^2)^{-N/2}\exp\left\{-\frac{1}{2\sigma^2}||\boldsymbol{y}-\boldsymbol{\Phi}\boldsymbol{w}||^2\right\} \end{aligned} \tag{4.5}$$

where $\boldsymbol{I}_M$ is the $M \times M$ identity matrix.

So far, we have specified the general linear regression model. To get to the sparse Bayesian formulation, we define a zero-mean Gaussian prior distribution over the parameters $\boldsymbol{w}$

$$p(\boldsymbol{w}\,|\,\boldsymbol{\alpha}) = \prod_{j=1}^{M}\mathcal{N}\left(w_j\,|\,\alpha_j^{-1}\right) \tag{4.6}$$

where $\boldsymbol{\alpha} = [\alpha_1,\cdots,\alpha_M]^T$ is a vector of M hyperparameters. It is important to note that each hyperparameter α_j is solely responsible for controlling the strength of the prior of its associated weight w_j. If α_j is large, the prior over w_j is very strongly peaked at zero. This form of the prior distribution is, more than anything, responsible for the dramatic sparsity in the final model.

To complete the specification, we must define a prior over the noise parameter σ^2 and the a hyperprior over the hyperparameters $\boldsymbol{\alpha}$. Following the derivation in [5], we use the following Gamma [2] priors

$$p(\boldsymbol{\alpha}\,|\,a,b) = \prod_{j=1}^{M}\text{Gamma}(\alpha_j\,|\,a,b)$$

$$p(\beta\,|\,c,d) = \text{Gamma}(\beta\,|\,c,d)$$

where $\beta \equiv \sigma^{-2}$. We make the simplifying assumption $a = b = c = d = 0$, giving as a uniform, but improper, hyperprior. For the general case, see [5].

As a side note, consider the prior of $\boldsymbol{w}$ after marrginalizing out the dependence on the hyperpriors $\boldsymbol{\alpha}$. Since each w_j is normally distributed with an unknown precision parameter

[2] The Gamma distribution is defined by

$$\text{Gamma}(z\,|\,a,b) = \Gamma(a)^{-1}b^a z^{a-1}\exp(-bz) \qquad z,a,b > 0$$

where $\Gamma(.)$ is the Gamma function defined by

$$\Gamma(z) = \int_0^\infty t^{z-1}\exp(-t)dt.$$

α_j and since the (hyper)prior over α_j is the Gamma distribution and therefore conjugate to $p(w_j \mid \alpha_j)$, it follows that the resulting integral can be evaluated analytically

$$\begin{aligned} p(\boldsymbol{w} \mid a,b) &= \int p(\boldsymbol{w} \mid \boldsymbol{\alpha}) p(\boldsymbol{\alpha} \mid a,b) d\boldsymbol{\alpha} \\ &= \prod_{j=1}^{M} \int \mathcal{N}(w_j \mid 0, \alpha_j^{-1}) \text{Gamma}(\alpha_j \mid a,b) d\alpha_j \\ &= \prod_{j=1}^{M} \frac{b^a \Gamma(a+\frac{1}{2})}{(2\pi)^{\frac{1}{2}} \Gamma(a)} \left(b + \frac{w_j^2}{2} \right)^{-(a+\frac{1}{2})} . \end{aligned}$$

This corresponds to a product of independent Student-t density functions over the weights w_j. The choice $a = b = 0$ implies that $p(\boldsymbol{w} \mid a,b) \propto \prod_{j=1}^{M} 1/|w_j|$. As discussed in [5], it is this hierarchical formulation of the weight prior that is ultimately responsible for encouriging sparse solutions.

4.2 Model Inference

...

4.2.1 Original Training Algorithm

Following [5]

4.2.2 Sequential Sparse Bayesian Learning Algorithm

Follows [6]

4.3 Making Predictions

For details on the RVM and its implementation see [3, 5, 6].

References

[1] Candès, E. J. and Wakin, M. B. (2008). An introduction to compressive sampling. *Signal Processing Magazine, IEEE*, 25(2):21–30.

[2] DeVore, R. A., Jawerth, B., and Lucier, B. J. (1992). Image compression through wavelet transform coding. *Information Theory, IEEE Transactions on*, 38(2):719–746.

[3] Pilikos, G. (2014). Signal reconstruction using compressive sensing. MPhil thesis, University of Cambridge.

[4] Stollnitz, E. J., DeRose, T. D., and Salesin, D. H. (1995). Wavelets for computer graphics: a primer. 1. *Computer Graphics and Applications, IEEE*, 15(3):76–84.

[5] Tipping, M. E. (2001). Sparse bayesian learning and the relevance vector machine. *The journal of machine learning research*, 1:211–244.

[6] Tipping, M. E., Faul, A. C., et al. (2003). Fast marginal likelihood maximisation for sparse bayesian models. In *AISTATS*.