Extended Linear Models with Gaussian Priors

Joaquin Quiñonero-Candela jqc@imm.dtu.dk
Institute of Mathematical Modeling
Technical University of Denmark
DK-2800, Kongens Lyngby, Denmark

Editor: To be submitted as a Technical Report to IMM - DTU

Abstract

In extended linear models the input space is projected onto a feature space by means of an arbitrary non-linear transformation. A linear model is then applied to the feature space to construct the model output. The dimension of the feature space can be very large, or even infinite, giving the model a very big flexibility. *Support Vector Machines* (SVM's) and Gaussian processes are two examples of such models. In this technical report I present a model in which the dimension of the feature space remains finite, and where a Bayesian approach is used to train the model with Gaussian priors on the parameters. The *Relevance Vector Machine*, introduced by Tipping Tipping (2001), is a particular case of such a model. I give the detailed derivations of the *expectation-maximisation* (EM) algorithm used in the training. These derivations are not found in the literature, and might be helpful for newcomers.

Keywords: sparseness, kernel regression, Gaussian processes, expectation-maximization (EM) algorithm, linear models

1. Introduction

As a particular case of extended linear models with Gaussian priors, Tipping's relevance vector machine (RVM) both achieves a sparse solution like the support vector machine (SVM) Vapnik (1998), Schölkopf and Smola (2002) and the probabilistic predictions of Bayesian kernel machines based upon a Gaussian process (GP) priors over functions Rasmussen (1996), Williams and Rasmussen (1996), MacKay (1997), Neal (1996), Opper and Winther (2000). Sparsity is interesting both with respect to fast training and predictions and ease of interpretation of the solution. Probabilistic predictions are desirable because inference is most naturally formulated in terms of probability theory, i.e. probabilities can be manipulated through Bayes theorem, uncertain predictions can be rejected, etc.

The training of the RVM consists in the maximization of the *marginal likelihood*, where under a Bayesian scheme the parameters have been integrated out by means of their prior. This maximization cannot be achieved in a direct way, and the number of parameters involved is potentially very high, making gradient approaches difficult to use. The *expectation-maximization* (EM) (Dempster et al., 1977) algorithm comes very naturally into play in this Bayesian approach. The purpose of this technical report is to introduce extended linear models with Gaussian priors, and to give the detailed derivation of the Bayesian training with the EM algorithm.

The technical report is organized as follows: In section 2 extended linear models in a Bayesian perspective are presented for the regression case. In section 3 the training process with the EM algorithm is described in detail. In section 4 a brief indication of relevant bibliography is given.

2. Extended linear models

An extended linear model is build by transforming the input space by an arbitrary set of *basis functions* $\phi_j : R^D \rightarrow R$ that performs a non-linear transformation of the D-dimensional input space. A linear model is applied to the transformed space whose dimension is equal to the number of basis functions:

$$y(\mathbf{x}_i) = \sum_{j=1}^{M} \omega_j \, \phi_j(\mathbf{x}_i) = \mathbf{\Phi}(\mathbf{x}_i) \cdot \boldsymbol{\omega} \tag{1}$$

where $\mathbf{\Phi}(\mathbf{x}_i) \equiv [\phi_1(\mathbf{x}_i), \ldots, \phi_M(\mathbf{x}_i)]$ denotes the ith row of the *design matrix* $\mathbf{\Phi}$ and $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_N)^T$ is the *weights* vector.

It is assumed that the observations are corrupted by additive Gaussian noise. Given a set of input-target training pairs $\{x_i, t_i\}_{i=1}^N$ and assuming that the observations are independent and contaminated with Gaussian noise of variance σ^2, the likelihood of the model parameters is written as:

$$p(\mathbf{t}|\boldsymbol{\omega}, \sigma^2) = \left(2\pi\sigma^2\right)^{-N/2} \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{t} - \mathbf{\Phi}\,\boldsymbol{\omega}\|^2\right) \tag{2}$$

where $\mathbf{t} = (t_1, \ldots, t_N)^T$ is the target vector.

It is well known that estimating the model parameters by maximizing their likelihood (2) leads to over-fitting except for very particular choices of the basis functions. Some form of regularization is needed. Rather than making point predictions of the optimal values of the weights, in Bayesian learning, a prior distribution over the over weights is introduced. In general, the implied prior over functions is a very complicated distribution. However, choosing a Gaussian prior on the weights the prior over functions also becomes Gaussian, i.e. a Gaussian process. For the specific choice of a factorized distribution with variance α_j^{-1}:

$$p(\omega_j|\alpha_j) = \sqrt{\frac{\alpha_j}{2\pi}} \exp\left(-\frac{1}{2}\alpha_j\,\omega_j^2\right) \tag{3}$$

the prior over functions $p(\mathbf{y}|\boldsymbol{\alpha})$ is $\mathcal{N}(0, \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^T)$, i.e. a Gaussian process with covariance function given by

$$\mathrm{Cov}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{M} \frac{1}{\alpha_k} \phi_k(x_i)\phi_k(x_j) \tag{4}$$

where $\boldsymbol{\alpha} = (\alpha_0, \ldots, \alpha_N)^T$ and $\mathbf{A} = \mathrm{diag}(\alpha_0, \ldots, \alpha_N)$. Sparseness in terms of the basis vectors may arise if for some k $\alpha_k^{-1} = 0$. Then the kth basis vector $\mathbf{\Phi}_k \equiv [\phi_k(\mathbf{x}_1), \ldots, \phi_k(\mathbf{x}_N)]^T$, i.e. the kth column in the design matrix, will not contribute to the model. Associating a basis function with each input point may thus lead to a model with a sparse representations in the inputs, i.e. the solution is only spanned by a subset of all input points. This is exactly the idea behind the relevance vector machine.

The posterior distribution over the weights, obtained using Bayes rule, is a Gaussian distribution

$$p(\boldsymbol{\omega}|\mathbf{t},\boldsymbol{\alpha},\sigma^2) = \frac{p(\mathbf{t}|\boldsymbol{\omega},\sigma^2)p(\boldsymbol{\omega}|\boldsymbol{\alpha})}{p(\mathbf{t}|\boldsymbol{\alpha},\sigma^2)} = \mathcal{N}(\mathbf{t}|\boldsymbol{\mu},\boldsymbol{\Sigma}) \tag{5}$$

where $\mathcal{N}(\mathbf{t}|\boldsymbol{\mu},\boldsymbol{\Sigma})$ is a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ evaluated at $\mathbf{t}$. The mean and covariance are given by

$$\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^T\mathbf{t} \tag{6}$$

$$\boldsymbol{\Sigma} = (\sigma^{-2}\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \mathbf{A})^{-1} \tag{7}$$

The *maximum a posteriori* (MAP) estimate of the weights is given by $\boldsymbol{\mu} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \sigma^{-2}\mathbf{A}\right)^{-1}\boldsymbol{\Phi}^T\mathbf{t}$. This is a version of the normal equations that includes a regularization term determined by the hyperparameters $\{\alpha_j\}$ and by the noise σ^2.

The likelihood distribution over the training targets (2) can be "marginalized" with respect to the weights to obtain the *marginal likelihood*, which is also a Gaussian distribution

$$p(\mathbf{t}|\boldsymbol{\alpha},\sigma^2) = \int p(\mathbf{t}|\boldsymbol{\omega},\sigma^2)\, p(\boldsymbol{\omega}|\boldsymbol{\alpha})\, d\boldsymbol{\omega} = \mathcal{N}(\mathbf{t}|0,\mathbf{C}) \tag{8}$$

with covariance

$$\mathbf{C} = \sigma^2\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^T \tag{9}$$

The uncertainty about the optimal value of the weights captured by the posterior distribution (5) can be used to build probabilistic predictions. Given a new input $\mathbf{x}_*$, the model gives a Gaussian *predictive distribution* of the corresponding target t_*

$$p(t_*|\mathbf{x}_*,\hat{\boldsymbol{\alpha}},\hat{\sigma}^2) = \int p(t_*|\mathbf{x}_*,\boldsymbol{\omega},\hat{\sigma}^2)\, p(\boldsymbol{\omega}|\mathbf{t},\hat{\boldsymbol{\alpha}},\hat{\sigma}^2)\, d\boldsymbol{\omega} = \mathcal{N}(t_*|y_*,\sigma_*^2) \tag{10}$$

where

$$y_* = \boldsymbol{\Phi}(\mathbf{x}_*)\cdot\boldsymbol{\mu} \tag{11}$$

$$\sigma_*^2 = \hat{\sigma}^2 + \boldsymbol{\Phi}(\mathbf{x}_*)\cdot\boldsymbol{\Sigma}\cdot\boldsymbol{\Phi}(\mathbf{x}_*)^T \tag{12}$$

For regression it is natural to use y_* and $\sigma^2{}_*$ as the prediction and the error bar on the prediction respectively. The computational complexity of making predictions is thus $\mathcal{O}(M^3 + M^2P)$, where M is the number of selected basis functions (RVs) and P is the number of predictions.

It has been shown by MacKay (1997) that the same predictive distribution as in (10) is obtained by conditioning the joint distribution of training examples and target associated to a new input on the training examples, which is the standard procedure in Gaussian processes. The error bars (12) are strongly conditioned on the form of the basis functions used. For example, the use of local basis functions results in small error bars if the new input happens to have no significant output from any of the basis functions, i.e. if the new point is far away from the training data.

3. Training with the EM algorithm

The maximization of the marginal likelihood with respect to $\{\alpha_j\}$ and σ^2 cannot be done in a closed form, and requires an iterative approach. The integration with respect to the weights to obtain the likelihood of $\{\alpha_j\}$ and σ^2, eq. (8), makes the use of the EM algorithm with the weights as hidden variables very natural. The logarithm of the marginal likelihood is taken as the cost function to be maximized:

$$\mathcal{L}(\sigma^2, \boldsymbol{\alpha}) = \log p\left(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2\right) = \log \int p\left(\mathbf{t}, \boldsymbol{\omega}|\boldsymbol{\alpha}, \sigma^2\right) d\boldsymbol{\omega} \tag{13}$$

Using Jensen's inequality with an arbitrary distribution $q(\boldsymbol{\omega})$ a lower bound for (13) is obtained

$$\log \int q(\boldsymbol{\omega}) \frac{p\left(\mathbf{t}, \boldsymbol{\omega}|\boldsymbol{\alpha}, \sigma^2\right)}{q(\boldsymbol{\omega})} d\boldsymbol{\omega} \geq \int q(\boldsymbol{\omega}) \log \frac{p\left(\mathbf{t}, \boldsymbol{\omega}|\boldsymbol{\alpha}, \sigma^2\right)}{q(\boldsymbol{\omega})} d\boldsymbol{\omega} \equiv \mathcal{F}(q, \sigma^2, \boldsymbol{\alpha}) \tag{14}$$

The functional $\mathcal{F}(q, \sigma^2, \boldsymbol{\alpha})$ from equation (14) is a lower bound on the log marginal likelihood that can be re-written as:

$$\begin{aligned}
\mathcal{F}(q, \sigma^2, \boldsymbol{\alpha}) &= \int q(\boldsymbol{\omega}) \log \frac{p\left(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2\right) p\left(\boldsymbol{\omega}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2\right)}{q(\boldsymbol{\omega})} d\boldsymbol{\omega} \\
\mathcal{F}(q, \sigma^2, \boldsymbol{\alpha}) &= \int q(\boldsymbol{\omega}) \log p\left(\mathbf{t}|\boldsymbol{\omega}, \sigma^2\right) d\boldsymbol{\omega} + \int q(\boldsymbol{\omega}) \log \frac{p\left(\boldsymbol{\omega}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2\right)}{q(\boldsymbol{\omega})} d\boldsymbol{\omega} \\
\mathcal{F}(q, \sigma^2, \boldsymbol{\alpha}) &= \mathcal{L}(\sigma^2, \boldsymbol{\alpha}) + \mathcal{KL}\left(q(\boldsymbol{\omega})||p(\boldsymbol{\omega}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)\right)
\end{aligned} \tag{15}$$

where $\mathcal{KL}\left(q(\boldsymbol{\omega})||p(\boldsymbol{\omega}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)\right)$ is the Kullback-Liebler ($\mathcal{KL}$) divergence between the arbitrary distribution $q(\boldsymbol{\omega})$ and the posterior distribution of the weights, eq. (5). The $\mathcal{KL}$ divergence is always a positive number and is only equal to zero if the two functions it takes as arguments are identical.

The E-step corresponds to maximizing the lower bound $\mathcal{F}$ on the target function with respect to $q(\boldsymbol{\omega})$, which from equation (15) is equivalent to minimizing $\mathcal{KL}\left(q(\boldsymbol{\omega})||p(\boldsymbol{\omega}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)\right)$. This is equivalent to making $q(\boldsymbol{\omega})$ as similar as possible to the posterior in the $\mathcal{KL}$ sense. In some cases $q(\boldsymbol{\omega})$ can be made identical to the posterior on the weights, in which case $\mathcal{F}$ is made equal to the $\mathcal{L}$ and the E-step is said to be *exact*. In the case of regression with extended linear models with Gaussian priors, the E-step can be made exact: it amounts to estimating the mean and covariance of the posterior on the weights, equations (6) and (7), for $\boldsymbol{\alpha}$ and σ^2 given.

The M-step corresponds to maximizing $\mathcal{F}(q, \sigma^2, \boldsymbol{\alpha})$ with respect to σ^2 and $\boldsymbol{\alpha}$ with a fixed $q(\boldsymbol{\omega})$. It is now useful to rewrite $\mathcal{F}$ in a different manner:

$$\begin{aligned}
\mathcal{F}(q, \sigma^2, \boldsymbol{\alpha}) &= \int q(\boldsymbol{\omega}) \log p\left(\mathbf{t}, \boldsymbol{\omega}|\boldsymbol{\alpha}, \sigma^2\right) d\boldsymbol{\omega} - \int q(\boldsymbol{\omega}) \log q\left(\boldsymbol{\omega}\right) d\boldsymbol{\omega} \\
\mathcal{F}(q, \sigma^2, \boldsymbol{\alpha}) &= \left\langle \log p\left(\mathbf{t}, \boldsymbol{\omega}|\boldsymbol{\alpha}, \sigma^2\right)\right\rangle_{q(\boldsymbol{\omega})} + \left\langle \log q(\boldsymbol{\omega})\right\rangle_{q(\boldsymbol{\omega})}
\end{aligned} \tag{16}$$

The M-step consists therefore in maximizing the average of the log marginal likelihood over $q(\boldsymbol{\omega})$ with respect to the parameters σ^2 and $\boldsymbol{\alpha}$:

$$\left\langle \log p\left(\mathbf{t}, \boldsymbol{\omega}|\boldsymbol{\alpha}, \sigma^2\right)\right\rangle_{q(\boldsymbol{\omega})} = \left\langle -\frac{N}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\|\mathbf{t} - \boldsymbol{\Phi}\,\boldsymbol{\omega}\|^2 + \frac{1}{2}\log|\mathbf{A}| - \frac{1}{2}\boldsymbol{\omega}^T\mathbf{A}\,\boldsymbol{\omega}\right\rangle_{q(\boldsymbol{\omega})} + K \tag{17}$$

where K is a constant that does not depend on $\boldsymbol{\omega}$, $\boldsymbol{\alpha}$ or σ^2.

3.1 EM Update rule for $\boldsymbol{\alpha}$

The right part of equation (17) that depends on $\boldsymbol{\alpha}$ can be rewritten as:

$$\begin{aligned}\left\langle \frac{1}{2}\log|\mathbf{A}| - \frac{1}{2}\boldsymbol{\omega}^T\mathbf{A}\,\boldsymbol{\omega}\right\rangle_{q(\boldsymbol{\omega})} &= \frac{1}{2}\log|\mathbf{A}| - \frac{1}{2}\,\mathrm{Tr}\left\langle \mathbf{A}\,\boldsymbol{\omega}\,\boldsymbol{\omega}^T\right\rangle_{q(\boldsymbol{\omega})}\\ &= \frac{1}{2}\log|\mathbf{A}| - \frac{1}{2}\,\mathrm{Tr}\left(\mathbf{A}\,\boldsymbol{\Sigma} + \mathbf{A}\boldsymbol{\mu}\,\boldsymbol{\mu}^T\right)\end{aligned} \tag{18}$$

where Tr is the trace operator. From equation (18), the derivative of $\mathcal{F}$ with respect to α_j is given by

$$\frac{\partial\mathcal{F}}{\partial\alpha_j} = \frac{1}{2}\left(\frac{1}{\alpha_j} - \boldsymbol{\Sigma}_{jj} - \mu_j^2\right)$$

that set to zero gives the following update rule:

$$\alpha_j^{new} = \frac{1}{\mu_j^2 + \Sigma_{jj}} \tag{19}$$

3.2 EM Update rule for σ^2

The right part of equation (17) that depends on σ^2 can be rewritten as:

$$\left\langle -\frac{N}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\|\mathbf{t} - \boldsymbol{\Phi}\,\boldsymbol{\omega}\|^2\right\rangle = -\frac{N}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\,\mathrm{Tr}\left\langle (\mathbf{t} - \boldsymbol{\Phi}\,\boldsymbol{\omega})^T(\mathbf{t} - \boldsymbol{\Phi}\,\boldsymbol{\omega})\right\rangle \tag{20}$$

The last term of the right part of equation (20) can be rewritten as:

$$\begin{aligned}\mathrm{Tr}\left\langle (\mathbf{t} - \boldsymbol{\Phi}\,\boldsymbol{\omega})^T(\mathbf{t} - \boldsymbol{\Phi}\,\boldsymbol{\omega})\right\rangle &= \mathrm{Tr}\left\langle \mathbf{t}^T\mathbf{t} - 2\boldsymbol{\omega}^T\boldsymbol{\Phi}^T\mathbf{t} + \boldsymbol{\omega}^T\boldsymbol{\Phi}^T\boldsymbol{\Phi}\,\boldsymbol{\omega}\right\rangle\\ &= \mathbf{t}^T\mathbf{t} - 2\boldsymbol{\mu}^T\boldsymbol{\Phi}^T\mathbf{t} + \boldsymbol{\mu}^T\boldsymbol{\Phi}^T\boldsymbol{\Phi}\,\boldsymbol{\mu} + \mathrm{Tr}(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\,\boldsymbol{\Sigma})\\ &= \|\mathbf{t} - \boldsymbol{\Phi}\,\boldsymbol{\mu}\|^2 + \mathrm{Tr}(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\,\boldsymbol{\Sigma})\end{aligned} \tag{21}$$

Now the derivative of $\mathcal{F}$ with respect to σ^2 is given by

$$\frac{\partial\mathcal{F}}{\partial\sigma^2} = \frac{N}{2\sigma^2} - \frac{1}{\sigma^4}\left(\|\mathbf{t} - \boldsymbol{\Phi}\,\boldsymbol{\mu}\|^2 + \mathrm{Tr}(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\,\boldsymbol{\Sigma})\right)$$

and by setting it to zero the following update rule is obtained:

$$(\sigma^2)^{new} = \frac{||\mathbf{t} - \boldsymbol{\Phi}\,\mu||^2 + \mathrm{Tr}(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\,\boldsymbol{\Sigma})}{N} \tag{22}$$

This expression can be written in a different way with the following manipulations:

$$
\begin{aligned}
\mathrm{Tr}\left(\mathbf{\Phi}^T\mathbf{\Phi}\,\mathbf{\Sigma}\right) &= \mathrm{Tr}\left((\sigma^2)^{old}\left((\sigma^{-2})^{old}\mathbf{\Phi}^T\mathbf{\Phi} + \mathbf{A} - \mathbf{A}\right)^{-1}\mathbf{\Sigma}\right) \\
&= (\sigma^2)^{old}\,\mathrm{Tr}\left(\mathbf{I} - \mathbf{A}\,\mathbf{\Sigma}\right) \\
&= (\sigma^2)^{old}\sum_{j=1}^{M}(1 - \alpha_j\,\mathbf{\Sigma}_{jj}) \qquad (23)
\end{aligned}
$$

since in the E-step the covariance of the posterior, $\mathbf{\Sigma}$, is estimated as

$$
\mathbf{\Sigma} = \left((\sigma^{-2})^{old}\mathbf{\Phi}^T\mathbf{\Phi} + \mathbf{A}\right)^{-1}
$$

Using the result of equation (23), the update rule for σ^2 from equation (22) can be now written as:

$$
(\sigma^2)^{new} = \frac{||\mathbf{t} - \mathbf{\Phi}\,\mu||^2 + (\sigma^2)^{old}\sum_j \gamma_j}{N} \qquad (24)
$$

where the quantity $\gamma_j \equiv 1 - \alpha_j\Sigma_{jj}$ is a measure of how "well-determined" each weight ω_j is by the data (MacKay, 1992, Tipping, 2001).

3.3 MacKay's own M-step

One can obtain a different update rule for the M-step that gives faster convergence. Although it is suboptimal in the EM sense, since the M-step does not maximize the functional $\mathcal{F}$, it has not been observed in practice that this update leads to "downhill" steps. The rule, derived by Tipping (2001), is obtained by differentiation of the target function, equation (13), and by an arbitrary choice of independent terms as is done by MacKay (1992). It makes use of the terms $\{\gamma_j\}$:

$$
\begin{aligned}
\alpha_j^{new} &= \frac{\gamma_j}{\mu_j^2} \qquad (25) \\
(\sigma^2)^{new} &= \frac{||\mathbf{t} - \mathbf{\Phi}\,\mu||^2}{N - \sum_j \gamma_j} \qquad (26)
\end{aligned}
$$

4. Further reading

One limiting factor of the training of extended linear models with Gaussian priors is the need to compute the covariance matrix of the weights at each E-step of the EM algorithm. This inversion has a computational cost bounded from below by $\mathcal{O}(M^2\,N)$ where M is the number of active basis functions (relevance vectors) and N is the number of training examples. That lead Tipping to suggest in an appendix in Ref. Tipping (2001) an incremental learning strategy that starts with only a single basis function and adds basis functions along the iterations. The total number of basis functions is kept low because basis functions are also removed. In two papers that are currently under revision (Quiñonero-Candela et al., 2002, Quiñonero-Candela and Winther, 2002), we formalize this strategy using straightforward expectation-maximization (EM) arguments to prove that the scheme is the guaranteed convergence to a local maximum of the likelihood of the model parameters. In the second paper we also address the problem of classification.

References

N. M. Dempster, A.P. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Statist. Soc. B*, 39:185–197, 1977.

David J. C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, 1992.

David J. C. MacKay. Gaussian processes - a replacement for supervised neural networks? Lecture notes for a tutorial in Advances in Neural Information Processing Systems, 1997.

Radford M. Neal. *Bayesian Learning for Neural Networks*. Springer, New York, 1996.

Manfred Opper and Ole Winther. Gaussian processes for classification: Mean field algorithms. *Neural Computation*, 12:2655–2684, 2000.

Joaquin Quiñonero-Candela and Ole Winther. Incremental gaussian processes. Number 14, 2002. Submited to Advances in Neural Information Processing Systems.

Joaquin Quiñonero-Candela, Ole Winther, and Lars Kai Hansen. Large scale bayesian kernel learning with the subspace em algorithm. 2002. Submited to the Journal of Machine Learning Research.

Carl E. Rasmussen. *Evaluation of Gaussian Processes and Other Methods for Non-linear Regression*. PhD thesis, Dept. of Computer Science, University of Toronto, 1996.

Bernhard Schölkopf and Alex J. Smola. *Learning with Kernels*. MIT Press, Cambridge, 2002.

Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

Chris K. I. Williams and Carl E. Rasmussen. Gaussian proceses for regression. In *Advances in Neural Information Processing Systems*, number 8, pages 514–520, 1996.