Scalable Approximate Inference for the Bayesian Nonlinear Support Vector Machine

Florian Wenzel, Matthäus Deutsch, Théo Galy-Fajou, Marius Kloft
Department of Computer Science
Humboldt University of Berlin
{wenzelfl,deutschm,galy,kloft}@hu-berlin.de

Abstract

We develop a variational inference (VI) scheme for the recently proposed Bayesian kernel support vector machine (SVM) and a stochastic version (SVI) for the linear Bayesian SVM. We compute the SVM's posterior, paving the way to apply attractive Bayesian techniques, as we exemplify in our experiments by means of automated model selection.

1 Introduction

There has recently been significant interest in utilizing max-margin based discriminative Bayesian models for various applications. For example [1] used max-margin based Bayesian classification to discover latent semantic structures for topic models, [2] to build efficient matrix factorization methods or [3] to develop new promising approaches to Hidden Markov models. All these approaches apply a Bayesian reformulation of the classic SVM [4] developed by [5]. [6] extended the model to the nonlinear case and showed that this leads to improved accuracy compared to standard methods like SVMs and Gaussian process (GP) classification. But their inference method has the drawback that it partially relies on point estimates of the latent variables and their proposed inference methods are not applicable to large datasets due to the high computational complexity.

We overcome these problems by developing an approximate Bayesian approach proposing a fast inference method based on variational inference. Since we can give a full approximate posterior our approach allows for the use of Bayesian techniques to SVMs on real world datasets as e.g. computing class probabilities, errorbars and automated hyperparameter search. Additionally, the proposed algorithms are much faster than the ones used by [6]. We exemplify this in our experiments, showing that the approach indeed leads to fast automated SVMs while directly giving uncertainty prediction without using additional heuristic methods like Platt [7]. In the end we give a short outline on how we aim to generalize and improve the model.

2 The Bayesian SVM

Let $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$ be n observations where $x_i \in \mathbb{R}^d$ is a feature vector with corresponding label $y_i \in \{-1, 1\}$. The SVM consists of finding the optimal score function f by solving the optimization problem

$$\arg\min_{f(x)} \sum_{i=1}^{n} \max\left(1 - y_i f(x_i), 0\right) + \gamma R\left(f\right), \tag{1}$$

where R is a regularizer function and γ a hyperparameter. The loss $\max\left(1 - y_i f(x_i), 0\right)$ is called hinge loss. The classifier is defined as $\text{sign}(f(x))$.

[5] developed a Bayesian formulation of the linear SVM (i.e. $f(x) = \beta^\top x$) and showed that estimating the mode of the pseudo-posterior

$$p(\beta|\mathcal{D}) \propto \prod_{i=1}^{n} L(y_i|x_i, \beta)p(\beta)$$

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

is equivalent to (1). $p(\beta)$ denotes a prior chosen such that $\log p(\beta) \propto -2\gamma R(\beta)$. In the following we use the prior $\beta \sim \mathcal{N}(0, \Sigma)$ but note that our method can be easily extended to other priors. L denotes a pseudolikelihood fulfilling $\log L \propto -2\max(1 - y_i f(x_i), 0)$. It was shown in [5] that by introducing latent variables $\lambda := (\lambda_1, \ldots, \lambda_n)^\top$ we can express L in terms of a normal variance-mean mixture, where we implicitly impose the improper prior $p(\lambda) = \mathbb{1}_{[0,\infty)}(\lambda)$ on λ. Writing $X \in \mathbb{R}^{d \times n}$ for the matrix of data points and $Y = \mathrm{diag}(y)$, the full conditionals of this model are given by

$$\begin{aligned} \beta|\lambda, \Sigma, \mathcal{D} &\sim \mathcal{N}(BZ(\lambda^{-1} + 1),\ B), \\ \lambda_i|\beta, \mathcal{D}_i &\sim \mathcal{GIG}\left(\frac{1}{2}, 1, (1 - y_i x_i^\top \beta)^2\right), \end{aligned} \tag{2}$$

with $Z = XY$ and $B^{-1} = Z\Lambda^{-1}Z^\top + \Sigma^{-1}$, $\Lambda = \mathrm{diag}(\lambda)$ and $\mathcal{GIG}$ denotes a generalized inverse Gaussian distribution.

By using the ideas of Gaussian processes [8], [6] developed a kernelized version of this model. To this end, they assume that a continuous decision function $f(x)$ is drawn from a zero-mean Gaussian process $\mathrm{GP}(0, k)$, where k is a kernel function. The random Gaussian vector $f = (f_1, ..., f_n)^\top$ corresponds to $f(x)$ evaluated at the data points. We substitute the linear function $x_i^\top \beta$ by f_i in the problem (1) and obtain the conditional posterior

$$f|\lambda, \mathcal{D} \sim \mathcal{N}(CY(\lambda^{-1} + 1),\ C),$$

with $C^{-1} = \Lambda^{-1} + K^{-1}$. For a test point x_* the conditional predictive distribution for $f_* = f(x_*)$ under this model is

$$f_*|\lambda, x_*, \mathcal{D} \sim \mathcal{N}\left(k_*^\top (K + \Lambda)^{-1} Y(1 + \lambda),\ k_{**} - k_*^\top (K + \Lambda)^{-1} k_*\right), \tag{3}$$

where $K := k(X, X)$, $k_* := k(X, x_*)$, $k_{**} := k(x_*, x_*)$ are the kernel matrices. Note that the conditional posteriors are all dependent on the local latent variable λ_i.

3 Fast Inference for the Bayesian SVM

We follow the mean field variational inference (MFVI) approach and apply stochastic variational inference (SVI) [9] to approximate the posterior of the Bayesian SVM. We extend the approach from [10] for the Bayesian linear SVM and propose a novel variational inference method for the Bayesian kernel SVM.

Variational Inference

We first consider the linear case. We aim to approximate the posterior $p(\beta, \lambda|\mathcal{D}) \approx q(\beta, \lambda) = q(\beta) \prod_{i=1}^n q(\lambda_i)$ of the global variable β and local variables $\lambda_i,\ i = 1, ..., n$. To this end we choose the variational distributions in the same family as the full conditionals,

$$\begin{aligned} q(\lambda_i) &\equiv \mathcal{GIG}(\frac{1}{2}, 1, \alpha_i) \\ q(\beta) &\equiv \mathcal{N}(\mu, \zeta), \end{aligned} \tag{4}$$

where $\alpha_i \geq 0$, $\mu \in \mathbb{R}^d$, $\zeta \in \mathbb{R}^{d \times d}$ (positive definite) are the free parameters. Since the variational distributions are in the exponential family the coordinate ascent variational inference (CAVI) updates are given by the expected natural parameters [11]. The local updates are given by

$$\alpha_i = \mathbb{E}_{q(\beta)}\left[(1 - z_i^T \beta)^2\right] = (1 - z_i^T \mu)^2 + z_i^T \zeta z_i,$$

and the natural parameter updates[1] for the variational Gaussian are given by

$$\eta_1 = \mathbb{E}_{q(\lambda)}\left[Z(\lambda^{-1} + 1)\right] = Z(\alpha^{-\frac{1}{2}} + 1),$$

and

$$\eta_2 = -\frac{1}{2}\mathbb{E}_{q(\lambda)}\left[Z\Lambda^{-1}Z^T + \Sigma^{-1}\right] = -\frac{1}{2}\left(Z(A^{-\frac{1}{2}})Z^T + \Sigma^{-1}\right),$$

[1] The standard mean and covariance parameter of our parametrization (4) can be recovered by $\zeta = -\frac{1}{2}\eta_2^{-1}$ and $\mu = \zeta\eta_1$.

where $A = \mathrm{diag}(\alpha)$ and $\alpha = (\alpha_i)_{1 \leq i \leq n}$.

For the Bayesian kernel SVM we follow again the MFVI approach and choose the variational families according to the full conditionals (2). The CAVI updates can be computed analogously to the linear case,

$$\begin{aligned}
\alpha_i &= (1 - y_i \mu_i)^2 + \zeta_{ii} \\
\eta_1 &= Y(\alpha^{-\frac{1}{2}} + 1) \\
\eta_2 &= -\frac{1}{2}\left(A^{-\frac{1}{2}} + K^{-1}\right),
\end{aligned}$$

where η_1 and η_2 are the natural parameters of the variational Gaussian. The VI scheme for the Bayesian kernel SVM is shown in Algorithm 2 in the appendix.

Stochastic Variational Inference

The batch variational inference scheme for the Bayesian linear SVM can be directly extended to a stochastic version (see Algorithm 1 in the appendix). We show in our experiments that this leads to a great speedup. We use an adaptive learning rate scheme [12] in the SVI algorithm.
Unfortunately, SVI for the Bayesian kernel SVM cannot be applied in a straight forward manner. The problem is that the probabilistic model does not have a set of global variables. Both the latent variables λ and the latent GP f correspond to the data points, i.e. they are local variables. To overcome this problem we plan in future work to use instead an inducing point GP with global sparse prior [13] that would lead to an appropriate model for SVI.

4 Predictive Distributions

We use the approximation of the posterior to compute the predictive distribution and class membership probabilities. Details are given in the appendix. The class membership probability distributions are

$$\begin{aligned}
&\text{Linear BSVM:} && p(y_* = 1 | x_*, \mathcal{D}) \approx \Phi\left(\frac{x_*^\top \mu^*}{x_*^\top \zeta^* x_* + 1}\right) \\
&\text{Kernel BSVM:} && p(y_* = 1 | x_*, \mathcal{D}) \approx \Phi\left(\frac{k_* K^{-1} \mu^*}{k_{**} + k_*^\top \left(K^{-1}\zeta^* K^{-1} - K^{-1}\right) k_* + 1}\right),
\end{aligned}$$

where $\Phi(.)$ denotes the probit link function (the normal cumulative density function).

5 Hyperparameter Optimization

We estimate the hyperparameters from the data by maximizing the marginal likelihood $p(y|X, h)$ (empirical Bayes [14]). We follow an approximate approach [15, 9] and optimize the fitted variational lower bound $\mathcal{L}(h)$ over h. We update the hyperparameters simultaneously with the variational parameters. To this end we add a hyperparameter optimization step after the variational updates in the SVI scheme,

$$h^{(t)} = h^{(t-1)} + \tilde{\rho}_t \nabla_h \mathcal{L}(\alpha^{(t-1)}, \mu^{(t-1)}, \zeta^{(t-1)}, h). \tag{5}$$

6 Experiments

In the following we apply our method to synthetic and real world data and show that they are much faster than the competing methods while having similar prediction performance. We show that our method quickly finds the optimal hyperparameters. We experiment with the batch variational inference methods and the SVI method for the Bayesian kernel SVM and compare against standard SVM (LibSVM [16]), MCMC-BSVM (Gibb's sampling based on [6]) and EP-based Gaussian Process Classification [8].

Synthetic Experiment for the Bayesian Linear SVM

We experiment on synthetically generated datasets of different sizes with known underlying parameter β. In Fig. 1 we plot the estimation error of β and the time.

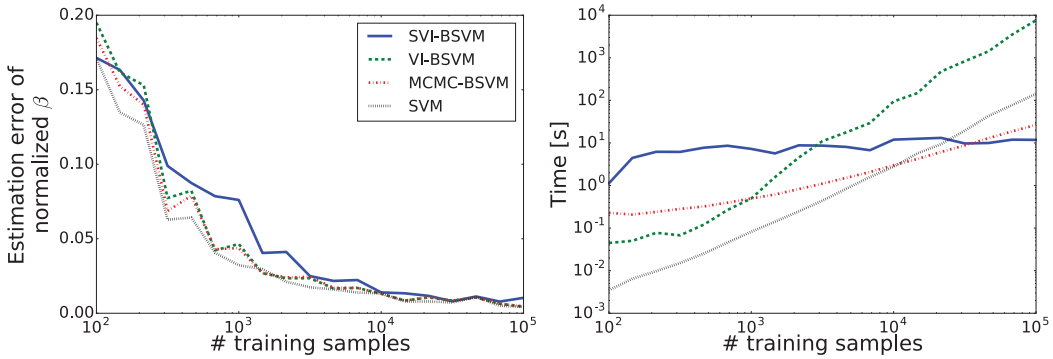

Figure 1: Performance and convergence time for the Linear SVM methods as function of dataset size.

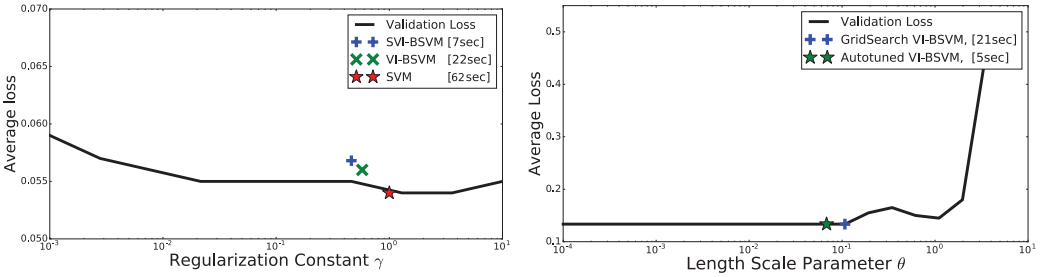

Figure 2: Mean % error as function of the hyperparamter. Left: Bayesian Linear SVM. Right: Bayesian Kernel SVM.

Automated Model Selection

For the Bayesian linear SVM we experiment on a synthetic dataset of 10,000 data points and estimate the regularization constant by applying (5) and compare against grid search (grid of 1000 points) for the standard SVM. For the Bayesian kernel SVM we experiment on the Sonar dataset (see Table 1). In Fig. 2 we plot the prediction error as function of the hyperparameters.

Prediction Performance of the Bayesian Kernel SVM on Real World Datasets

For all methods we use a radial basis function kernel and estimate the kernel parameters by using our automated model selection approach (5) for VI-BSVM and cross-validation for the competitors. We experiment on different standard benchmark datasets and report the prediction performance in Table 1. Our method took between 20 and 35 seconds for the each dataset (including auto tuning of the hyperparameters).

Data set	N	d	VI-BSVM	LibSVM	GPC
Sonar	208	60	**12.5**	13.5	19.5
Crabs	200	7	**1.0**	**1.0**	3.1
Pima	768	8	**22.8**	24.7	22.8
USPS 3vs5	1540	256	2.0	**1.6**	2.3

Table 1: Mean % error from 10-fold cross-validation.

7 Conclusion

We proposed a new inference method for the Bayesian SVM that scales to large datasets and allows for approximating the full posterior. Our approach lets us automatically tune the hyperparameters of the SVM and leads to class membership probabilities. In future work we aim to make our inference method even faster by applying the concept of GPs for big data [13]. We plan to further extend the Bayesian SVM model to account for correlations between data points building on ideas from [17]. Additionally, we aim to embed the model into more general frameworks of normal variance-mean mixtures.

Acknowledgments

We thank Stephan Mandt, Manfred Opper and Patrick Jähnichen for fruitful discussions. This work was partly funded by the German Research Foundation (DFG) award KL 2698/2-1.

References

[1] J. Zhu, N. Chen, H. Perkins, and B. Zhang, "Gibbs Max-margin Topic Models with Data Augmentation," *Journal of Machine Learning Research*, vol. 15, pp. 1073–1110, 2014.

[2] M. Xu, J. Zhu, and B. Zhang, "Fast Max-Margin Matrix Factorization with Data Augmentation," in *Proceedings of the 30th International Conference on Machine Learning*, vol. 28, pp. 978–986, 2013.

[3] A. Zhang, Z. Jun, and B. Zhang, "Max-Margin Infinite Hidden Markov Models," in *Proceedings of the 31st International Conference on Machine Learning*, vol. 32, pp. 315–323, 2014.

[4] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[5] N. G. Polson and S. L. Scott, "Data augmentation for support vector machines," *Bayesian Anal.*, 2011.

[6] R. Henao, X. Yuan, and L. Carin, "Bayesian Nonlinear Support Vector Machines and Discriminative Factor Modeling," in *Proceedings of the 27th International Conference on NIPS*, 2014.

[7] P. J. C. Platt, "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods," *Advances in Large Margin Classifier*, 1999.

[8] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[9] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic Variational Inference," *Journal of Machine Learning Research*, vol. 14, pp. 1303–1347, 2013.

[10] J. Luts and J. T. Ormerod, "Mean field variational bayesian inference for support vector machine classification," *Comput. Stat. Data Anal.*, vol. 73, pp. 163–176, May 2014.

[11] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An Introduction to Variational Methods for Graphical Models," *Mach. Learn.*, vol. 37, pp. 183–233, Nov. 1999.

[12] R. Ranganath, C. Wang, D. M. Blei, and E. P. Xing, "An Adaptive Learning Rate for Stochastic Variational Inference," *Proceedings of the 30th International Conference on Machine Learning*, 2013.

[13] J. Hensman, N. Fusi, and N. D. Lawrence, "Gaussian processes for big data," in *Conference on Uncertainty in Artificial Intellegence*, 2013.

[14] J. Maritz and T. Lwin, "Empirical Bayes Methods with Applications," *Monographs on Statistics and Applied Probability.*, 1989.

[15] S. Mandt, M. Hoffman, and D. Blei, "A Variational Analysis of Stochastic Gradient Algorithms," in *Proceedings of the 33rd International Conference on Machine Learning*, vol. 48, 2016.

[16] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[17] S. Mandt, F. Wenzel, S. Nakajima, C. Lippert, and M. Kloft, "Separating Sparse Signals from Correlated Noise in Binary Classification," *Proceedings of the UAI Workshop Causation: Foundation to Application*, 2016.

8 Appendix

A SVI Scheme for the Bayesian linear SVM

Algorithm 1 SVI for the Bayesian linear SVM

set the learning rate schedule ρ_t appropriately
initialize η_1, η_2
while not converged **do**
  get $\mathcal{S}$ = minibatch index set of size s
  update $\alpha_i = (1 - z_i^T \mu)^2 + z_i^T \zeta z_i \quad \forall i \in \mathcal{S}$
  compute $A_{\mathcal{S}} = \mathrm{diag}(\alpha_i, \ \ i \in \mathcal{S})$
  compute $\hat{\eta}_1 = \frac{n}{s} Z_{\mathcal{S}} (\alpha_{\mathcal{S}}^{-\frac{1}{2}} + 1)$
  compute $\hat{\eta}_2 = -\frac{1}{2} \left(\frac{n}{s} Z_{\mathcal{S}} (A_{\mathcal{S}}^{-\frac{1}{2}}) Z_{\mathcal{S}}^T + \Sigma^{-1} \right)$
  update $\eta_1 = (1 - \rho_t)\eta_1 + \rho_t \hat{\eta}_1$
  update $\eta_2 = (1 - \rho_t)\eta_2 + \rho_t \hat{\eta}_2$
  compute $\zeta = -\frac{1}{2} \eta_2^{-1}$
  compute $\mu = \zeta \eta_1$
return $\alpha_1, \ldots, \alpha_n, \mu, \zeta$

B Batch VI Scheme for the Bayesian kernel SVM

Algorithm 2 Batch VI Scheme for the Bayesian kernel SVM

initialize μ, ζ
while not converged **do**
  update $\alpha_i = (1 - y_i \mu_i)^2 + \zeta_{ii} \quad \forall i$
  compute $A = \mathrm{diag}(\alpha)$
  compute $\eta_1 = Y(\alpha^{-\frac{1}{2}} + 1)$
  compute $\eta_2 = -\frac{1}{2} \left(A^{-\frac{1}{2}} + K^{-1} \right)$
  compute $\zeta = -\frac{1}{2} \eta_2^{-1}$
  compute $\mu = \zeta \eta_1$
return $\alpha_1, \ldots, \alpha_n, \mu, \zeta$

C Predictive Distributions

We use the approximation of the posterior to compute the predictive distribution and class membership probabilities. Compared to (3) we do not condition on λ and use the variational distributions obtained by our inference method. Let α^*, μ^*, ζ^* be the variational parameters and $x_* \in \mathbb{R}^d$ a new test point.

Bayesian Linear SVM

Let $q^*(\beta, \lambda) \approx p(\beta, \lambda | \mathcal{D})$ be the variational distribution obtained by SVI. The predictive distribution can be approximated by

$$p(f_* | x_*, \mathcal{D}) \approx \int p(f_* | \beta) q^*(\beta, \lambda) \mathrm{d}f \mathrm{d}\lambda = \mathcal{N}(f_*; \, x_*^\top \mu^*, x_*^\top \zeta^* x_*).$$

This leads to an approximation of the class membership probability,

$$p(y_* = 1 | x_*, \mathcal{D}) \approx \int \Phi(f_*) q(f_* | x_*) df_* = \Phi\left(\frac{x_*^\top \mu^*}{x_*^\top \zeta^* x_* + 1} \right),$$

where $\Phi(.)$ denotes the probit link function (the normal cumulative density function).

Bayesian Nonlinear SVM

Let $q^*(f, \lambda) \approx p(f, \lambda|\mathcal{D})$ be the variational distribution obtained by SVI. Using standard identities for Gaussian processes we obtain an approximation to the predictive distribution,

$$p(f_*|x_*, \mathcal{D}) \approx \int p(f_*|f)q^*(f, \lambda)\mathrm{d}f\mathrm{d}\lambda = \mathcal{N}\left(f_*;\, k_* K^{-1}\mu^*, k_{**} + k_*^\top \left(K^{-1}\zeta^* K^{-1} - K^{-1}\right) k_*\right).$$

The class membership probability can be approximated by

$$p(y_* = 1|x_*, \mathcal{D}) \approx \int \Phi(f_*)q(f_*|x_*)\mathrm{d}f_* = \Phi\left(\frac{k_* K^{-1}\mu^*}{k_{**} + k_*^\top \left(K^{-1}\zeta^* K^{-1} - K^{-1}\right) k_* + 1}\right).$$