Bayesian Analysis (2011) **6**, Number 1, pp. 1–24

Data Augmentation for Support Vector Machines

Nicholas G. Polson*and Steven L. Scott†

Abstract. This paper presents a latent variable representation of regularized support vector machines (SVM's) that enables EM, ECME or MCMC algorithms to provide parameter estimates. We verify our representation by demonstrating that minimizing the SVM optimality criterion together with the parameter regularization penalty is equivalent to finding the mode of a mean-variance mixture of normals pseudo-posterior distribution. The latent variables in the mixture representation lead to EM and ECME point estimates of SVM parameters, as well as MCMC algorithms based on Gibbs sampling that can bring Bayesian tools for Gaussian linear models to bear on SVM's. We show how to implement SVM's with spike-and-slab priors and run them against data from a standard spam filtering data set.

Keywords: MCMC, Bayesian inference, Regularization, Lasso, L^α-norm, EM, MCMC, ECME.

1 Introduction

Support vector machines (SVM's) are binary classifiers that are often used with extremely high dimensional covariates. SVM's typically include a regularization penalty on the vector of coefficients in order to manage the bias-variance trade-off inherent with high dimensional data. In this paper, we develop a latent variable representation for regularized SVM's in which the coefficients have a complete data likelihood function equivalent to weighted least squares regression. We then use the latent variables to implement EM, ECME, and Gibbs sampling algorithms for obtaining estimates of SVM coefficients. These algorithms replace the conventional convex optimization algorithm for SVM's, which is fast but unfamiliar to many statisticians, with what is essentially a version of iteratively re-weighted least squares. By expressing the support vector optimality criterion as a variance-mean mixture of linear models with normal errors, the latent variable representation brings all of conditionally linear model theory to SVM's. For example, it allows for the straightforward incorporation of random effects (Mallick et al. 2005), lasso and bridge L^α-norm priors, or "spike and slab" priors (George and McCulloch 1993, 1997; Ishwaran and Rao 2005).

The proposed methods inherit all the advantages and disadvantages of canonical data augmentation algorithms including convenience, interpretability and computational stability. The EM algorithms are stable because successive iterations never decrease the

*Booth School of Business, Chicago, IL, mailto:ngp@chicagobooth.edu
†Google Corporation, mailto:stevescott@google.com

 DOI:10.1214/11-BA601

objective function. The Gibbs sampler is stable in the sense that it requires no tuning, moves every iteration, and provides Rao-Blackwellised parameter estimates. The primary disadvantage of data augmentation methods is speed. The EM algorithm exhibits linear (i.e. slow) convergence near the mode, and one can often design MCMC algorithms that mix more rapidly than Gibbs samplers.

We argue that on the massive data sets to which SVM's are often applied there are reasons to prefer data augmentation over methods traditionally regarded as faster. First, Meng and van Dyk (1999) and Polson (1996) have shown that many data augmentation algorithms can be modified to increase their mixing rate. Second, data augmentation methods can be formulated in terms of complete data sufficient statistics, which is a considerable advantage when working with large data sets, where most of the computational expense comes from repeatedly iterating over the data. Methods based on complete data sufficient statistics need only compute those statistics once per iteration, at which point the entire parameter vector can be updated. This is of particular importance in the posterior simulation problem, where scalar updates (such as those in an element-by-element Metropolis Hastings algorithm) of a $k-$dimensional parameter vector would require $O(k)$ evaluations of the posterior distribution per iteration.

An additional benefit of our methods is that they provide further insight into the role of the *support vectors* in SVM's. The support vectors are observations whose covariate vectors lie very near the boundary of the decision surface. Hastie et al. (2009) show, using geometric arguments, that these are the only vectors supporting the decision boundary. We provide a simple algebraic view of the same fact by showing that the support vectors receive infinite weight in the iteratively re-weighted least squares algorithm.

The rest of the article is structured as follows. Section 2 explains the latent variable representation and the conditional distributions and moments needed for the EM and related algorithms. Section 3 defines an EM algorithm that can be used to locate SVM point estimates. We also describe how to use the marginal pseudo-likelihood to solve the optimal amount of regularization. The Gibbs sampler for SVM's is developed in Section 4, which also introduces spike-and-slab priors for SVM's. Section 5 illustrates our methods on the spam filtering data set from Hastie et al. (2009). Finally, Section 6 concludes.

2 Support Vector Machines

Support vector machines describe a binary outcome $y_i \in \{-1, 1\}$ based on a vector of predictors $\mathbf{x}_i = (1, x_1, \ldots, x_{k-1})$. SVM's often include kernel expansions of $\mathbf{x}_i$ (e.g. a spline basis expansion) prior to fitting the model. Our methods are agnostic to any such kernel expansions, and we assume that $\mathbf{x}_i$ already includes all desired expansion terms. The L^α-norm regularized support vector classifier chooses a set of coefficients β

to minimize the objective function

$$d_\alpha(\beta,\nu) = \sum_{i=1}^{n} \max\left(1 - y_i \mathbf{x}_i^T \beta, 0\right) + \nu^{-\alpha} \sum_{j=1}^{k} |\beta_j/\sigma_j|^\alpha \tag{1}$$

where σ_j is the standard deviation of the j'th element of $\mathbf{x}$ and ν is a tuning parameter. There is a geometric interpretation to equation (1). If a hyperplane in $\mathbf{x}$ can perfectly separate the sets $\{i : y_i = 1\}$ and $\{i : y_i = -1\}$, then the solution to equation (1) gives the separating hyperplane farthest from any individual observation. Algebraically, if $\beta^T \mathbf{x}_i \geq 0$ then one classifies observation i as 1. If $\beta^T \mathbf{x}_i < 0$ then one classifies $y_i = -1$.

The scaling variable σ_j is the standard deviation the j'th predictor variable, with the exception of $\sigma_1 = 1$ for the intercept term. There is ample precedent for the choice of scaling variables. See Mitchell and Beauchamp (1988), George and McCulloch (1997), Clyde and George (2004), Fan and Li (2001), Griffin and Brown (2005), and Holmes and Held (2006). The second term in equation (1) is a regularization penalty corresponding to a prior distribution $p(\beta|\nu,\alpha)$. The lasso prior (Tibshirani 1996; Zhu et al. 2004), corresponding to $\alpha = 1$ is a popular choice because it tends to produce posterior distributions where many of the β coefficients are exactly zero at the mode.

Minimizing equation (1) is equivalent to finding the mode of the pseudo-posterior distribution $p(\beta|\nu,\alpha,y)$ defined by

$$\begin{aligned} p(\beta|\nu,\alpha,y) &\propto \exp\left(-d_\alpha(\beta,\nu)\right) \\ &\propto C_\alpha(\nu) L(y|\beta) p(\beta|\nu,\alpha). \end{aligned} \tag{2}$$

The factor of $C_\alpha(\nu)$ is a pseudo-posterior normalization constant that is absent in the classical analysis. The data dependent factor $L(y|\beta)$ is a pseudo-likelihood

$$L(y|\beta) = \prod_i L_i(y_i|\beta) = \exp\left\{-2 \sum_{i=1}^{k} \max\left(1 - y_i \mathbf{x}_i^T \beta, 0\right)\right\}. \tag{3}$$

In principle, one could work with an actual likelihood if each L_i was replaced by the normalized value $\tilde{L}_i = [L_i(y_i)]/[L_i(y_i) + L_i(-y_i)]$, but we work with L_i instead of $\tilde{L}_i$ because it leads to the traditional estimator for support vector machine coefficients. It is also possible to learn (β,ν,α) jointly from the data by defining a joint pseudo-posterior $p(\beta,\nu,\alpha|y) \propto p(\beta|\nu,\alpha,y)p(\nu,\alpha)$ for some initial prior regularization penalty $p(\nu,\alpha)$ on the amount of regularization. Sections 3.3 and 4 explore the necessary details.

The purpose of the next subsection is to show that a formula equivalent to equation (1) can be expressed as a mixture of normal distributions. Section 2.1 establishes that fundamental result. Then Section 2.2 derives the conditional distributions used in the MCMC and EM algorithms later in the paper.

2.1 Mixture Representation

Our main theoretical result expresses the pseudo-likelihood contribution $L_i(y_i|\beta)$ as a location-scale mixture of normals. The result allows us to pair observation y_i with

a latent variable λ_i in such a way that L_i is the marginal from a joint distribution $L_i(y_i, \lambda_i|\beta)$ in which β appears as part of a quadratic form. This implies that $L_i(y_i, \lambda_i|\beta)$ is conjugate to a multivariate normal prior distribution. In effect, the augmented data space allows the awkward SVM optimality criterion to be expressed as a conditionally Gaussian linear model that is familiar to most Bayesian statisticians.

Theorem 1. *The pseudo-likelihood contribution from observation y_i can be expressed as*

$$\begin{aligned} L_i(y_i|\beta) &= \exp\left\{-2\max\left(1 - y_i\mathbf{x}_i^T\beta, 0\right)\right\} \\ &= \int_0^\infty \frac{1}{\sqrt{2\pi\lambda_i}} \exp\left(-\frac{1}{2}\frac{(1+\lambda_i - y_i\mathbf{x}_i^T\beta)^2}{\lambda_i}\right)\, d\lambda_i \ . \end{aligned} \tag{4}$$

The proof relies on the integral identity $\int_0^\infty \phi\left(u|-\lambda, \lambda\right) d\lambda = e^{-2\max(u,0)}$ where $\phi(u|\cdot,\cdot)$ is the normal density function. The derivation of this identity follows from Andrews and Mallows (1974), who proved that $\int_0^\infty \frac{a}{\sqrt{2\pi\lambda}} e^{-\frac{1}{2}\left(a^2\lambda + b^2\lambda^{-1}\right)} d\lambda = e^{-|ab|}$, for any $a, b > 0$. Substituting, $a = 1, b = u$ and multiplying through by e^{-u} yields

$$\int_0^\infty \frac{1}{\sqrt{2\pi\lambda}} e^{-\frac{u^2}{2\lambda} - u - \frac{1}{2}\lambda} d\lambda = e^{-|u|-u}.$$

Finally, using the identity $\max(u, 0) = \frac{1}{2}(|u| + u)$ gives the expression

$$\int_0^\infty \frac{1}{\sqrt{2\pi\lambda}} e^{-\frac{(u+\lambda)^2}{2\lambda}} d\lambda = e^{-2\max(u,0)},$$

which is the desired result.

A corresponding result can be given for the exponential power prior distribution containing the regularization penalty,

$$p(\beta|\nu, \alpha) = \prod_{j=1}^k p(\beta_j|\nu, \alpha) = \left(\frac{\alpha}{\nu\Gamma(1+\alpha^{-1})}\right)^k \exp\left(-\sum_{i=1}^k \left|\frac{\beta_j}{\nu\sigma_j}\right|^\alpha\right). \tag{5}$$

We consider the general case of $\alpha \in (0, 2]$ though the special cases of $\alpha = 2$ and $\alpha = 1$ are by far the most important, as they correspond to "ridge regression" (Goldstein and Smith 1974; Holmes and Pintore 2006) and the "lasso" (Tibshirani 1996; Hans 2009) respectively. West (1987) develops the mixture result for $\alpha \in [1, 2]$ and the same argument extends to the case $\alpha \in (0, 1]$, see Gomez-Sanchez-Manzano et al. (2008). The latter allows us to apply our method to the "bridge" estimator framework (Huang et al. 2008). The general case is stated below as Theorem 2.

Theorem 2. *(Pollard 1946; West 1987) The prior regularization penalty can be expressed as a scale mixture of normals*

$$p(\beta_j|\nu, \alpha) = \int_0^\infty \phi\left(\beta_j|0, \nu^2\omega_j\sigma_j^2\right) p(\omega_j|\alpha)\ d\omega_j \tag{6}$$

where $p(\omega_j|\alpha) \propto \omega_j^{-\frac{3}{2}} St^+_{\frac{\alpha}{2}}(\omega_j^{-1})$ *and* $St^+_{\frac{\alpha}{2}}$ *is the density function of a positive stable random variable of index* $\alpha/2$.

A simpler mixture representation can be obtained for the special case of $\alpha = 1$.

Corollary 1. *(Andrews and Mallows 1974) The double exponential prior regularization penalty can be expressed as a scale mixture of normals*

$$p(\beta_j|\nu, \alpha = 1) = \int_0^\infty \phi\left(\beta_j|0, \nu^2\omega_j\sigma_j^2\right) \frac{1}{2} e^{-\frac{\omega_j}{2}} \, d\omega_j \tag{7}$$

and so $p(\omega_j|\alpha) \sim \mathcal{E}(2)$ *is an exponential with mean* 2.

Corollary 1 was applied to Bayesian robust regression by Carlin and Polson (1991).

2.2 Conditional Distributions

Theorems 1 and 2 allow us to express the SVM pseudo-posterior distribution as the marginal of a higher dimension distribution that includes the variables $\lambda = (\lambda_1, \ldots, \lambda_n)$, $\omega = (\omega_1, \ldots, \omega_k)$. The complete data pseudo-posterior distribution is

$$\begin{aligned} p(\beta, \lambda, \omega|y, \nu, \alpha) \propto & \prod_{i=1}^n \lambda_i^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\sum_{i=1}^n \frac{(1 + \lambda_i - y_i\mathbf{x}_i^T\beta)^2}{\lambda_i}\right) \\ & \times \prod_{j=1}^k \omega_j^{-\frac{1}{2}} \exp\left(-\frac{1}{2\nu^2}\sum_{j=1}^k \frac{\beta_j^2}{\sigma_j^2\omega_j}\right) p(\omega_j|\alpha). \end{aligned} \tag{8}$$

where, in general, $p(\omega_j|\alpha) \propto \omega_j^{-\frac{3}{2}} St^+_{\frac{\alpha}{2}}(\omega_j^{-1})$.

At first glance equation (8) appears to suggest that y_i is conditionally Gaussian. However y_i, λ_i and β each have different support, with $y_i \in \{-1, 1\}$, $\lambda_i \in [0, \infty)$, and $\beta \in \Re^k$. Equation (8) is a proper density with respect to Lebesgue measure on β, λ, ω in the sense that it integrates to a finite number, but it is not a *probability* density because it does not integrate to one. This is a consequence of our use of the un-normalized likelihood in equation (3). The previous section shows that equation (8) has the correct marginal distribution,

$$p(\beta|\nu, \alpha, y) = \int p(\beta, \lambda, \omega|\nu, \alpha, y) d\lambda \, d\omega.$$

Therefore, it can be used to develop an MCMC algorithm that repeatedly samples from $p(\beta|\lambda, \omega, \nu, y)$, $p(\lambda_i^{-1}|\beta, \nu, y)$, and $p(\omega_j^{-1}|\beta_j, \nu)$, or develop an EM algorithm based on the moments of these distributions. The next subsection derives the required full conditional distributions from equation (8), with special attention given to the cases $\alpha = 1, 2$.

There are other purely probabilistic models where our result applies. For example, Mallick et al. (2005) provide a Bayesian SVM model by adding Gaussian errors around the linear predictors in order to obtain a tractable likelihood. Effectively they consider an objective of the form $\max(1 - y_i z_i, 0)$ where $z_i = x_i'\beta + \epsilon_i$. Our data augmentation strategy can help in designing MCMC algorithms in this case as well. Pontil et al. (1998) provide an alternative probabilistic model: imagine data arising from randomly sampling an unknown function $f(x)$ according to $f(x_i) = y_i + \epsilon_i$ where ϵ_i has an error distribution proportional to Vapnik's insensitive loss function: $\exp(-V_\epsilon(x))$ defined by $V_\epsilon(x) = \max(|x| - \epsilon, 0)$. Our results show that this distribution can be expressed as a mixture of normals.

The full conditional distribution of β given λ, ω, y

Define the matrices $\Lambda = \text{diag}(\lambda)$, $\Omega = \text{diag}(\omega)$, $\Sigma = \text{diag}\left(\sigma_1^2, \ldots, \sigma_k^2\right)$, and let $\mathbf{1}$ denote a vector of 1's. Also let $\mathbf{X}$ denote a matrix with row i equal to $y_i \mathbf{x}_i$. To develop the full conditional distribution $p(\beta|\nu, \lambda, \omega, y)$ one can appeal to standard Bayesian arguments by writing the model in hierarchical form

$$\begin{aligned} \mathbf{1} + \lambda &= \mathbf{X}\beta + \Lambda^{\frac{1}{2}}\epsilon^{\lambda} \\ \beta &= \frac{1}{\nu}\Omega^{\frac{1}{2}}\Sigma^{\frac{1}{2}}\epsilon^{\beta}, \end{aligned}$$

where ϵ^β and ϵ^λ are vectors of iid standard normal deviates with dimensions matching β and λ. Hence we have a conditional normal posterior for the parameters β given by

$$p(\beta|\nu, \lambda, \omega, y) \sim \mathcal{N}(b, B) \tag{9}$$

with hyperparameters

$$B^{-1} = \nu^{-2}\Sigma^{-1}\Omega^{-1} + \mathbf{X}^T\Lambda^{-1}\mathbf{X} \text{ and } b = B\mathbf{X}^T(\mathbf{1} + \lambda^{-1}). \tag{10}$$

Full conditional distributions for λ_i and ω_j given β, ν, y

The full conditional distributions for λ_i and ω_j are expressed in terms of the inverse Gaussian and generalized inverse Gaussian distributions. A random variable has the inverse Gaussian distribution $\mathcal{IG}(\mu, \lambda)$ with mean and variance $E(x) = \mu$ and $Var(x) = \mu^3/\lambda$ if its density function is

$$p(x|\mu, \lambda) = \sqrt{\frac{\lambda}{2\pi x^3}} \exp\left(-\frac{\lambda(x-\mu)^2}{2\mu^2 x}\right).$$

A random variable has the generalized inverse Gaussian distribution (Devroye 1986, p. 479) $\mathcal{GIG}(\gamma, \psi, \chi)$ if its density function is

$$p(x|\gamma, \psi, \chi) = C(\gamma, \psi, \chi)x^{\gamma-1}\exp\left(-\frac{1}{2}\left(\frac{\chi}{x} + \psi x\right)\right),$$

where $C(\gamma, \psi, \chi)$ is a suitable normalization constant. The generalized inverse Gaussian distribution contains the inverse Gaussian distribution as a special case: if $X \sim \mathcal{GIG}(1/2, \lambda, \chi)$ then $X^{-1} \sim \mathcal{IG}(\mu, \lambda)$ where $\chi = \lambda/\mu^2$. This fact is used to prove the following corollary of Theorem 1.

Corollary 2. *The full conditional distributions for λ_i is*

$$p(\lambda_i^{-1}|\beta, y_i) \sim \mathcal{IG}\left(|1 - y_i \mathbf{x}_i^T \beta|^{-1}, 1\right). \tag{11}$$

Proof: The full conditional distribution is

$$\begin{aligned} p(\lambda_i|\beta, y_i) &\propto \frac{1}{\sqrt{2\pi\lambda_i}} \exp\left\{-\frac{\left(1 - y_i \mathbf{x}_i^T \beta - \lambda_i\right)^2}{2\lambda_i}\right\} \\ &\propto \frac{1}{\sqrt{2\pi\lambda_i}} \exp\left\{-\frac{1}{2}\left(\frac{(1 - y_i \mathbf{x}_i^T \beta)^2}{\lambda_i} + \lambda_i\right)\right\} \\ &\sim \mathcal{GIG}\left\{\frac{1}{2}, 1, (1 - y_i \mathbf{x}_i \beta)^2\right\}. \end{aligned}$$

Equivalently, $p(\lambda_i^{-1}|\beta, y_i) \sim \mathcal{IG}\left(|1 - y_i \mathbf{x}_i \beta|^{-1}, 1\right)$ as required.

The full conditional distribution of ω_j is proportional to the integrand of equation (6). In general this is a complicated distribution because the density of the stable mixing distribution is generally only available in terms of its characteristic function. However, closed form solutions are available in the two most common special cases. When $\alpha = 2$ then $p(\omega_j|\beta)$ is a point mass at 1. The following result handles $\alpha = 1$.

Corollary 3. *For $\alpha = 1$, the full conditional distribution of ω is*

$$p(\omega_j^{-1}|\beta_j, \nu) \sim \mathcal{IG}\left(\nu\sigma_j/|\beta_j|, 1\right). \tag{12}$$

Proof: From the integrand in equation (7) we have

$$\begin{aligned} p(\omega_j|\beta_j, \nu) &\propto \frac{1}{\sqrt{2\pi\omega_j}} \exp\left\{-\frac{1}{2}\left(\frac{\beta_j^2/\nu^2\sigma_j^2}{\omega_j} + \omega_i\right)\right\} \\ &\sim \mathcal{GIG}\left(\frac{1}{2}, 1, \frac{\beta_j^2}{\nu^2\sigma_j^2}\right). \end{aligned}$$

Hence $(\omega_j^{-1}|\beta_j, \nu) \sim \mathcal{IG}\left(\nu\sigma_j/|\beta_j|, 1\right)$. We now develop the learning algorithms.

3 Point estimation by EM and related algorithms

This Section shows how the distributions obtained in Section 2 can be used to construct EM-style algorithms to solve for the coefficients. Section 3.1 describes an EM algorithm for learning β with a fixed value of ν. Then, Section 3.2 develops an ECME algorithm when ν is unknown.

3.1 Learning β with fixed ν

The EM algorithm (Dempster et al. 1977) alternates between an *E*-step (expectation) and an *M*-step (maximization) defined by

$$\begin{aligned} &\text{E-step} && Q(\beta|\beta^{(g)}) = \int \log p(\beta|y,\lambda,\omega,\nu) p(\lambda,\omega|\beta^{(g)},\nu,y)\ d\lambda\ d\omega \\ &\text{M-step} && \beta^{(g+1)} = \arg\max_{\beta} Q(\beta|\beta^{(g)}). \end{aligned}$$

The sequence of parameter values $\beta^{(1)}, \beta^{(2)}, \ldots$ monotonically increases the observed-data pseudo-posterior distribution: $p(\beta^{(g)}|\nu,\alpha,y) \leq p(\beta^{(g+1)}|\nu,\alpha,y)$.

The Q function in the $E-$step is the expected value of the complete data log posterior, where the expectation is taken with respect to the posterior distribution evaluated at current parameter estimates. The complete data log-posterior is

$$\begin{aligned} \log p(\beta|\nu,\lambda,\omega,y) = c_0(\lambda,\omega,y,\nu) &- \frac{1}{2}\sum_{i=1}^{n} \frac{\left(1+\lambda_i - y_i \mathbf{x}_i^T\beta\right)^2}{\lambda_i} \\ &- \frac{1}{2\nu^2}\sum_{j=1}^{k} \frac{\beta_j^2}{\sigma_j^2\omega_j} \end{aligned} \tag{13}$$

for a suitable constant c_0.

The terms in the first sum are linear functions of both λ_i and λ_i^{-1}. However, the λ_i term is free of β, so it can be absorbed into the constant. Thus, the relevant portion of equation (13) is a linear function of λ_i^{-1} and ω_j^{-1}, which means that the criterion function $Q(\beta|\beta^{(g)})$ simply replaces λ_i^{-1} and ω_j^{-1} with their conditional expectations $\hat{\lambda}_i^{-1(g)}$ and $\hat{\omega}^{-1(g)}$ given observed data and the current $\beta^{(g)}$. From Corollary 2 and properties of the inverse Gaussian distribution we have

$$\hat{\lambda}_i^{-1(g)} = E(\lambda_i^{-1}|y_i,\beta^{(g)}) = |1 - y_i\mathbf{x}_i^T\beta^{(g)}|^{-1}. \tag{14}$$

The corresponding result for ω_j^{-1} depends on α. When $\alpha = 2$ then $\omega_j = 1$. The general case $0 < \alpha < 2$ is given in the following Corollary of Theorem 2.

Corollary 4. *For $\alpha < 2$, if $\beta_j^{(g)} = 0$ then $\hat{\omega}_j^{-1(g)} = E(\omega^{-1}|\beta^{(g)},\alpha,y) = \infty$. Otherwise*

$$\hat{\omega}_j^{-1(g)} = \alpha|\beta_j^{(g)}|^{\alpha-2}(\nu\sigma_j)^{2-\alpha}.$$

Proof: From Theorem 2, we have

$$p(\beta_j|\alpha) = \int \phi(\beta_j|0,\nu^2\sigma_j^2\omega_j)p(\omega_j|\alpha)d\omega_j$$

where $p(\beta_j|\alpha) \propto \exp\left(-|\beta_j/\nu\sigma_j|^{\alpha}\right)$. Now notice that

$$\frac{\partial\phi(\beta_j|0,\nu^2\sigma_j^2\omega_j)}{\partial\beta_j} = \frac{-\beta_j}{\nu^2\sigma^2\omega_j}\phi(\beta_j|0,\nu^2\sigma_j^2\omega_j).$$

Hence, for $\beta_j \neq 0$ we can differentiate under the integral sign with respect to β_j to obtain

$$\alpha(\nu\sigma_j)^{-\alpha}|\beta_j|^{\alpha-1}p(\beta_j|\alpha) = \int_0^\infty \phi(\beta_j|0, \nu^2\sigma_j^2\omega_j)p(\omega_j|\alpha)\frac{\beta_j}{\nu^2\sigma_j^2\omega_j}d\omega_j.$$

Dividing by $p(\beta_j|\alpha)$ yields

$$\alpha(\nu\sigma_j)^{-\alpha}|\beta_j|^{\alpha-1} = \frac{\beta_j}{\nu^2\sigma_j^2}\int_0^\infty \frac{1}{\omega}\frac{p(\beta_j, \omega|\alpha)}{p(\beta_j|\alpha)}d\omega = \frac{\beta_j}{\nu^2\sigma_j^2}E(\omega^{-1}|\beta_j, \alpha).$$

Solving for $E(\omega^{-1}|\beta_j, \alpha)$ completes the proof. In the case when $\alpha = 1$ we can apply Corollary 3 to obtain

$$\hat{\omega}_j^{-1(g)} = \nu\sigma_j|\beta_j^{(g)}|^{-1},$$

which matches the general case. These results lead us to the following algorithm.

Algorithm: EM-SVM

Repeat the following until convergence

E-Step Given a current estimate $\beta = \beta^{(g)}$, compute

$$\begin{aligned}\hat{\lambda}^{-1(g)} &= (|1 - y_i\mathbf{x}_i^T\beta^{(g)}|^{-1}),\\ \hat{\Lambda}^{-1(g)} &= \text{diag}(\hat{\lambda}^{-1(g)}),\\ \hat{\Omega}^{-1(g)} &= \text{diag}\left(\hat{\omega}_j^{-1(g)}\right).\end{aligned}$$

M-Step Compute $\beta^{(g+1)}$ as

$$\beta^{(g+1)} = \left(\nu^{-2}\Sigma^{-1}\hat{\Omega}^{-1(g)} + \mathbf{X}^T\hat{\Lambda}^{-1(g)}\mathbf{X}\right)^{-1}\mathbf{X}^T(\mathbf{1} + \hat{\lambda}^{-1(g)}).$$

A few points about the preceding algorithm deserve emphasis. First, the $M-$step is essentially weighted least squares with weights λ_i^{-1}, though it is unusual in the sense that the weights also appear as part of the dependent variable. Second, the algorithm provides a new way of looking at support vectors. Observation i is a support vector if $y_i\mathbf{x}_i\beta = 1$, which means that it lies on the decision boundary. Equation (14) shows that support vectors receive infinite weight in the weighted least squares calculation. Thus we cannot find more than k linearly independent support vectors, and once k are found they are the only points determining the solution. Third, $\beta_j = 0$ is a fixed point in the algorithm when $\alpha < 2$. In practical terms this means that a search for a sparse model must proceed by backward selection starting from a model with all nonzero coefficients. A consequence is that the early iterations of the algorithm are the most expensive, because they involve computing the inverse of a large matrix. As

coefficients move sufficiently close to zero, the corresponding rows and columns can be removed from $\mathbf{X}^T\hat{\Lambda}^{-1(g)}\mathbf{X}$ and $\mathbf{X}^T(\mathbf{1}+\hat{\lambda}^{-1(g)})$. When k is very large, we implement the $M-$step using the conjugate gradient algorithm (Golub and van Loan 2008) initiated from the previous $\beta^{(g)}$ in place of the exact matrix inverse.

3.2 Stability

The EM algorithm in the previous section will become unstable once some elements of λ^{-1} or ω^{-1} become numerically infinite. Unlike more familiar regression problems, where infinite values signal an ill-conditioned problem, the infinities here are expected consequences of a normally functioning algorithm. When $\omega_j^{-1} = \infty$ it follows that $\beta_j = 0$, in which case one may simply omit column j from $\mathbf{X}$ and element j from β. When $\lambda_i^{-1} = \infty$ observation i is a support vector for which the constraint $y_i\beta^T\mathbf{x}_i = 1$ must be satisfied. Numerical stability can be restored by separating the support vectors from the rest of the data. Let $\mathbf{X}_s$ denote the matrix obtained by stacking the linearly independent support vectors row-wise (i.e. each row of $\mathbf{X}_s$ is a support vector). Let $\mathbf{X}_{-s}$ denote $\mathbf{X}$ with the support vector rows deleted. Let λ_{-s}^{-1} denote the finite elements of λ^{-1}, and let $\Lambda_{-s}^{-1} = \text{diag}(\lambda_{-s}^{-1})$.

A stable version of the $M-$step can be given by "restricted least squares" (Greene and Seaks 1991). The restricted least squares estimate is the solution to the equations

$$\begin{pmatrix} \mathbf{X}_{-s}^T(\mathbf{1}+\lambda_{-s}^{-1}) \\ \mathbf{1} \end{pmatrix} = \begin{pmatrix} B_{-s} & \mathbf{X}_s^T \\ \mathbf{X}_s & 0 \end{pmatrix} \begin{pmatrix} \beta \\ \psi \end{pmatrix}, \tag{15}$$

where ψ is a vector of Lagrange multipliers and

$$B_{-s} = \nu^{-2}\Sigma^{-1}\Omega^{-1} + \mathbf{X}_{-s}^T\Lambda_{-s}^{-1}\mathbf{X}_{-s}.$$

The inverse of the partitioned matrix in equation (15) can be expressed as

$$\begin{pmatrix} B_{-s}(I + \mathbf{X}_{-s}^T F\mathbf{X}_{-s}B_{-s}) & -B_{-s}\mathbf{X}_s^T F \\ -F\mathbf{X}_s B_{-s} & F \end{pmatrix}$$

where $F = -(\mathbf{X}_s B_{-s}\mathbf{X}_s^T)^{-1}$. The partitioned inverse can be used to obtain a closed form solution to equation (15).

3.3 Learning β and ν simultaneously

The expectation-conditional maximization algorithm (ECM Meng and Rubin 1993) is a generalization of EM that can be used when there are multiple sets of parameters to be located. The ECM algorithm replaces the $M-$step with a sequence of conditional maximization (CM) steps that each maximize Q with respect to one set of parameters, conditional on numerical values of the others. Liu and Rubin (1994) showed that the ECM algorithm converges faster if conditional maximizations of Q are replaced by conditional maximizations of the observed data posterior. Liu and Rubin called this

algorithm ECME, with the final "E" referring to conditional maximization of either function. The ECME algorithm retains the monotonicity property from EM.

An ECME algorithm for learning β and ν together can be obtained by assuming a prior distribution $p(\nu)$. The inverse gamma distribution

$$p(\nu^{-\alpha}) \propto \left(\frac{1}{\nu^{\alpha}}\right)^{a_\nu - 1} \exp(-b_\nu \nu^{-\alpha})$$

is a useful choice because it is conditionally conjugate to the exponential power distribution in equation (5). Under this prior one may estimate ν with a minor modification of the algorithm in Section 3.1. Note that the factor of ν^{-k} from equation (5), which could be ignored when ν was fixed, is now relevant.

Algorithm: ECME-SVM

E-Step Identical to the *E*-step of *EM-SVM* with $\nu = \nu^{(g)}$.

CM-Step Identical to the *M*-step of *EM-SVM* with $\nu = \nu^{(g)}$.

CME-Step Set

$$(\nu^{-\alpha})^{(g+1)} = \frac{b_\nu + \sum_{j=1}^{k} |\beta_j^{(g+1)}/\sigma_j|^{\alpha}}{k/\alpha + a_\nu - 1}.$$

The CME step could be replaced by a CM step that estimates ν in terms of the latent variables ω_j^{-1}, but as mentioned above doing so would delay convergence.

4 MCMC for SVM

The development of MCMC techniques for SVM's is important for two reasons. The first is that the SVM fitting algorithms in use today only provide point estimates, with no measures of uncertainty. This has motivated the Bayesian treatments of Sollich (2001), Tipping (2001), Cawley and Talbot (2005), Gold et al. (2005) and Mallick et al. (2005). Section 4.1 explains how the latent variable representation from Section 2 leads to a computationally efficient Gibbs sampler that can be seen as a competitor to these methods. The second reason is that latent variable methods for SVM's allow tools normally associated with linear models to be brought to SVM's. Section 4.2 demonstrates this fact by building an MCMC algorithm for SVM's with spike-and-slab priors.

4.1 MCMC for L^α priors

We first develop an MCMC-SVM algorithm for $\alpha = 1$. Then we describe how to deal with the general α case, including the possibility of learning α from the data.

Algorithm: MCMC-SVM ***($\alpha = 1$ case)***

Step 1: Draw $\beta^{(g+1)} \sim p(\beta|\nu, \Lambda^{(g)}, \Omega^{(g)}, y) \sim \mathcal{N}\left(b^{(g)}, B^{(g)}\right)$.

Step 2: Draw $\lambda^{(g+1)} \sim p\left(\lambda|\beta^{(g+1)}, y\right)$ for $1 \leq i \leq n$, where

$$\lambda_i^{-1}|\beta, \nu, y_i \sim \mathcal{IG}\left(|1 - y_i \mathbf{x}_i^T \beta|^{-1}, 1\right).$$

Step 3: Draw $\omega^{(g+1)} \sim p\left(\omega|\beta^{(g+1)}, y\right)$ for $1 \leq i \leq p$, where

$$\omega_j^{-1}|\beta_j, \nu \sim \mathcal{IG}\left(\nu\sigma_j|\beta_j|^{-1}, 1\right).$$

There are two easy modifications of the preceding algorithm that may prove useful. First, one can add a step that samples ν from its full conditional.

Step 4: Draw $\nu^{(g+1)}$ from the conditional

$$p(\nu^{-1}|\beta, y) \sim \Gamma\left(a_\nu + k, b_\nu + \sum_{i=1}^{k} |\beta_i|\right).$$

A second, and somewhat more radical departure would be to simulate α from

$$p(\alpha|\beta, \nu) \propto \left(\frac{\alpha}{\Gamma(1 + \frac{1}{\alpha})}\right)^k \exp\left(-\sum_{i=1}^{k} \left|\frac{\beta_j}{\nu\sigma_j}\right|^\alpha\right).$$

The draw from $p(\alpha|\beta, \nu)$ is a scalar random variable on a bounded interval, which can easily be handled using the slice sampler (Neal 2003).

There is reason to believe that averaging over ν (and potentially α) will lead to increased mean squared error accuracy. Further improvements can be had by using a Rao-Blackwellised estimate of β,

$$E(\beta|y) = \frac{1}{G}\sum_{g=1}^{G} b^{(g)}.$$

Mallick et al. (2005, MGG) investigate the use of posterior means in an MCMC analysis of SVM's. MGG report that model averaging leads to dramatically increased performance relative to "optimal" SVM chosen using standard methods. Our setting differs from the MGG setup in two important respects. First, MGG modified the basic SVM model by adding Gaussian errors around the linear predictors in order to obtain a tractable likelihood, where we work with the standard SVM criterion. However, we note that because we are working with the un-normalized SVM criterion our sampler draws from a pseudo-posterior distribution. The degree to which this affects the usual wisdom of Bayesian averaging is unclear. However, if mean squared prediction error is

the goal of the analysis rather than (oracle) variable selection then the posterior predictive mean should be competitive. Second, the MCMC algorithm from MGG updates each element of β component-wise while our sampler provides a block update resulting in much less time per iteration. Our data augmentation method could also be used to draw the latent Gaussian errors introduced by MGG in a block update, resulting in a further increase in speed.

The MCMC-SVM algorithm described above will *not* produce a sparse model, even though the modal value of the conditional pseudo-posterior distribution might be zero for some elements of β for a given ν (Hans 2009). There are two ways to recover sparsity using MCMC. The first is to use MCMC as the basis for a simulated annealing alternative to the EM algorithms from Section 3. A version of the pseudo-posterior suitable for simulated annealing can be incorporated by introducing a scaling parameter τ into the distribution as follows $\tau^{-1}\exp\left(-(2/\tau)d(\beta,\nu)\right)$. Then the MCMC algorithm can be run while gradually reducing τ from 1 to 0 (see, for example Tropp 2006).

4.2 Spike and slab priors

A second, more compelling, way of introducing sparsity is to replace the L_1 regularization penalty with a "spike-and-slab" prior (George and McCulloch 1993, 1997) that contains actual mass at zero. Johnstone and Silverman (2004, 2005) have pointed out the good frequentist risk properties of spike-and-slab (and other heavy tailed regularization penalties) for function estimation.

A spike and slab prior can be defined as follows. Let $\gamma = (\gamma_j)$ where $\gamma_j = 1$ if $\beta_j \neq 0$ and $\gamma_j = 0$ otherwise. A convenient prior for γ is $p(\gamma) = \prod_j \pi_j^{\gamma}(1-\pi_j)^{1-\gamma_j}$. A typical choice is to set all π_j equal to the same value π. Choosing $\pi = 1/2$ results in the uniform prior over model space. A more informative prior can be obtained by setting $\pi = k_0/k$, where k_0 is a prior guess at the number of included coefficients. Let β_γ denote the subset of β with nonzero elements, and let Σ_γ^{-1} denote the rows and columns of Σ^{-1} corresponding to $\gamma_j = 1$. Then we can write a spike-and-slab prior as

$$\gamma \sim p(\gamma), \qquad \beta_\gamma|\gamma \sim \mathcal{N}(0, \nu^2[\Sigma_\gamma^{-1}]^{-1}). \tag{16}$$

With this prior distribution, the posterior distribution of γ may be written

$$\begin{aligned} p(\gamma|y,\lambda,\nu) \propto p(\gamma)&\frac{|\Sigma_\gamma^{-1}/\nu^2|^{1/2}}{|B_\gamma^{-1}|^{1/2}} \\ &\exp\left(-\frac{1}{2}\sum_{i=1}^{n}\frac{(1+\lambda_i - y_i\mathbf{x}_{i,\gamma}^T b_\gamma)^2}{\lambda_i} - \frac{1}{2\nu^2}b_\gamma^T\Sigma_\gamma^{-1}b_\gamma\right) \end{aligned} \tag{17}$$

where b and B are defined in equation (10).

Algorithm: MCMC-SVM (spike-and-slab)

Step 1: Draw $\lambda_i^{(g+1)} \sim p\left(\lambda|\beta^{(g)}, y\right)$ for $1 \le i \le n$, where

$$p(\lambda_i^{-1}|\beta, \nu, y_i) \sim \mathcal{IG}\left(|1 - y_i \mathbf{x}_i^T \beta|^{-1}, 1\right).$$

Step 2: For $j = 1, \ldots, k$ draw γ_i from $p(\gamma_i|\gamma_{-i})$, which is proportional to equation (17).

Step 3: Draw $\beta_\gamma^{(g+1)} \sim \mathcal{N}\left(b_\gamma^{(g+1)}, B_\gamma^{(g+1)}\right)$.

Here we can exploit the identity

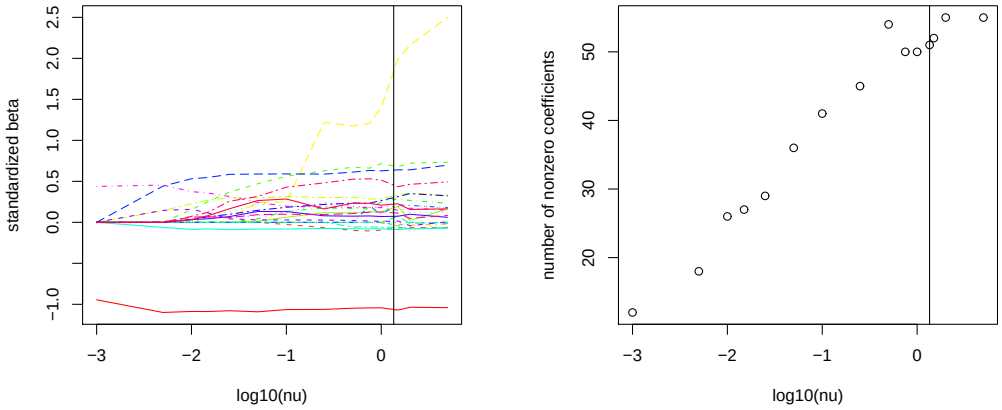
$$\sum_{i=1}^{n} \frac{(1 + \lambda_i - y_i \mathbf{x}_{i,\gamma}^T b_\gamma)^2}{\lambda_i} = c(\lambda) + b_\gamma^T \mathbf{X}_\gamma^T \Lambda^{-1} \mathbf{X}_\gamma b_\gamma - 2 b_\gamma^T \mathbf{X}_\gamma^T (\mathbf{1} + \lambda^{-1}),$$

which allows equation (17) to be evaluated in terms of the complete data sufficient statistics $\mathbf{X}_\gamma^T \Lambda^{-1} \mathbf{X}_\gamma$ and $\mathbf{X}_\gamma^T (\mathbf{1} + \lambda^{-1})$. The identities $\mathbf{X}_\gamma^T \Lambda^{-1} \mathbf{X}_\gamma = (\mathbf{X}^T \Lambda^{-1} \mathbf{X})_\gamma$ and $\mathbf{X}_\gamma^T (1 + \lambda^{-1}) = [\mathbf{X}^T (1 + \lambda^{-1})]_\gamma$, imply that (for a given imputation of λ) the complete data sufficient statistics do not need to be recomputed each time a new model is explored. Model exploration is thus very fast. This algorithm has more steps than MCMC-SVM($\alpha = 1$), but each step can be much faster because it is never necessary to invert the full $k \times k$ precision matrix.

5 Applications

The email spam data described by Hastie et al. (2009) is a benchmark example in the classification literature. It consists of 4601 rows, each corresponding to an email message that has been labeled as spam or not spam. Each message is measured on the covariates described in Table 1. There are a total of 58 predictors, including the intercept. We ran the EM and MCMC algorithms against this data set several times with several different values of the tuning parameters. Figure 1 plots the estimated coefficients for the first 25 variables in the model, showing how the dimension of the model increases with ν. Shrinkage from the lasso prior is evident in Figure 1. When ν is small there are several coefficients that are close to zero but which have not passed the numerical threshold to be counted as zero in Figure 1(b).

The ECME algorithm proved more robust than standard software at estimating SVM coefficients. Figure 2(a) compares the results from the ECME algorithm to the *R* package `penalizedSVM` using the value of $\nu = 1.353$ that ECME found optimal. The `penalizedSVM` package is parameterized in terms of $\lambda = 1/\nu$, so we used $\lambda = 1/1.353 = .739$ in the following. Figure 2(a) shows rough agreement between ECME and `penalizedSVM`, subject to two caveats. First, we expect a difference in coefficients because of different scaling conventions. Our algorithm scales the predictor variables



(a) (b)

Figure 1: *(a) The first 25 standardized coefficients ($\beta_j\sigma_j$) under the $\alpha = 1$ penalty and (b) number of nonzero coefficients as a function of ν. The vertical line dot is at the estimated optimal value of ν.*

through the factors of σ_j in the prior distribution. The `penalizedSVM` algorithm requires that the columns of $\mathbf{X}$ be scaled beforehand. The second source of disagreement is that both algorithms depend on randomization to some degree. Our algorithm initializes β to a vector of standard normal random deviates. The randomization is necessary because $\beta_j = 0$ is a fixed point of the ECME algorithm, so we cannot initialize with zeros. We ran both ECME and `penalizedSVM` multiple times, in some cases varying nothing but the random seed, in others varying the strategies for centering and scaling the predictor matrix input to `penalizedSVM`. In each case the *R* package produced between 3 and 6 large coefficients that dominated the others by 2-3 orders of magnitude. The specific sets of variables with large coefficients differed from one run to the next. Figure 2(b) shows the results from two successive runs of `penalizedSVM`, with the outliers truncated so the remaining structure can be observed. Figures 2(c) and 2(d) show three successive runs of ECME, with no truncation. The first and third runs converged, but the second had not converged after 500 iterations. The Figures show that the degree of agreement between runs of ECME (even without convergence) is greater than `penalizedSVM`, and ECME produced no large outliers.

Figure 3 plots the marginal posterior inclusion probability for each variable based on the Gibbs sampler with a spike and slab prior, where we set $\nu = 100$ so that there would be little shrinkage for variables that were included in the model. The bars in Figure 3 are shaded in proportion to the probability that the associated coefficient is positive. Thus a large black bar indicates an important variable that is associated with spam. A large white bar is an important variable that signals the absence of spam. Both the

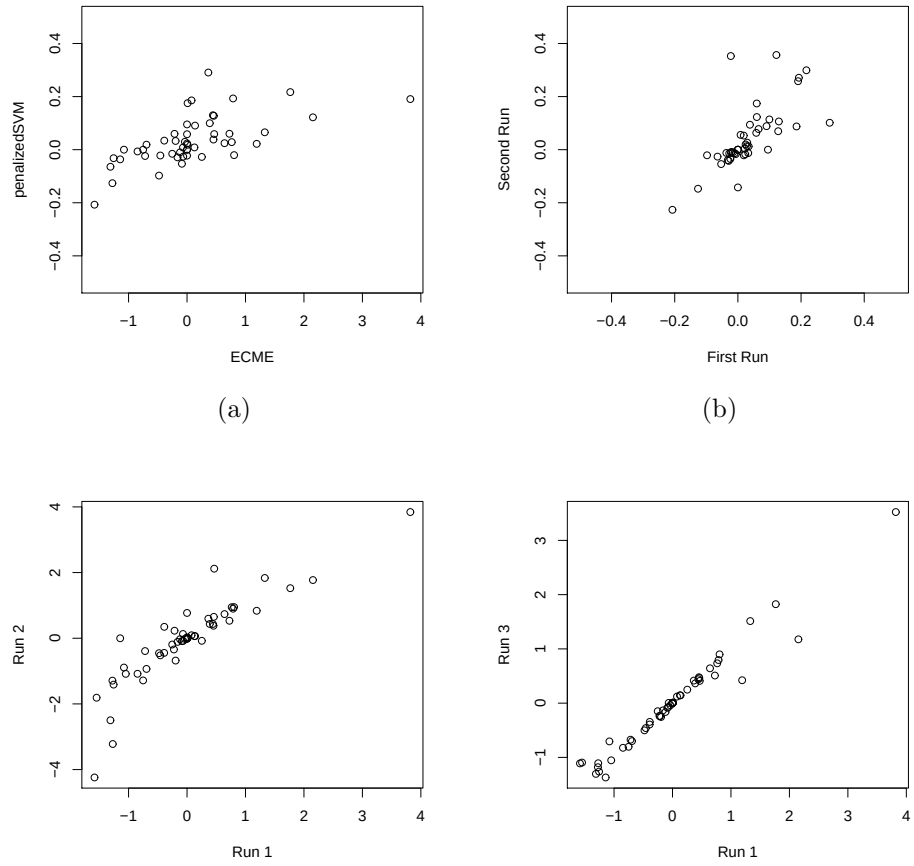


Figure 2: *(a) Coefficients from our ECME algorithm plotted against coefficients from a standard SVM fit using the R package* `penalizedSVM`*. (b) Coefficients from two runs of* `penalizedSVM` *with different random seeds. Both plots truncate a few very large outliers from* `penalizedSVM`*. (c) Coefficients from two runs of our ECME algorithm (the second run did not converge). (d) Coefficients from two converged runs of ECME.*

predictor	number	meaning
word_freq_X	48	percentage of words in the email that match the word X
char_freq_X	6	percentage of characters in the email that match the character X
CRL_average	1	average length of uninterrupted sequences of capital letters
CRL_longest	1	length of longest uninterrupted sequence of capital letters
CRL_total	1	total number of capital letters in the email

Table 1: *Variables in the spam data set.*

sparse ($\pi = .01$) and permissive ($\pi = .5$) models identify many of the same features as being associated with spam. The permissive model includes all of the variables which are certain to appear in the sparse model, as well as a few others that signal the absence of spam. Both models include many more predictors in the posterior distribution than are suggested by the prior.

The posterior draws produced by the MCMC algorithm largely agree with the point estimates from the EM and ECME algorithms. Figure 4 plots the MCMC sample paths for several coefficients, along with point estimates from the model with the optimal value of $\nu = 1.353$ estimated by ECME. Figures 4(a) and 4(b) are typical MCMC sample paths for parameters that are rarely set to zero. They mix reasonably fast, and typically agree with the ECME point estimates. Figures 4(c) and 4(d) are typical of variables with inclusion probabilities relatively far from 0 and 1. For these variables, ECME point estimates either tend to agree with the nonzero MCMC values, or else they split the difference between 0 and the conditional mean given inclusion. Figure 5 plots the coefficients and the raw data for the only two variables where the MCMC algorithm disagreed with point estimates from EM and ECME. The two variables in question are `wf_george` and `wf_cs`, both of which are strong signals indicating the absence of spam. The words "george" and "cs" are personal attributes of the original collector of this data, a computer scientist named George Forman.

A referee questioned whether the disagreement between MCMC and ECME might be due to a lack of convergence in the MCMC algorithm. To address this possibility we re-ran the sampler for 100,000 iterations, but the sampler did not leave the range of values that it visited in the shorter run of 10,000 iterations. The disagreement on these two predictors is more likely because both are such strong anti-spam signals that the model has difficulty determining just how large a weight they should receive. The prior distribution plays an important role in regularizing these types of "wild" coefficients. The spike-and-slab prior is much weaker than the double exponential prior outside a neighborhood of zero, so it offers the coefficients more leeway to drift towards positive or negative infinity. The fact that ECME and MCMC essentially agreed on the remaining 56 parameters engenders confidence that both algorithms are functioning correctly.

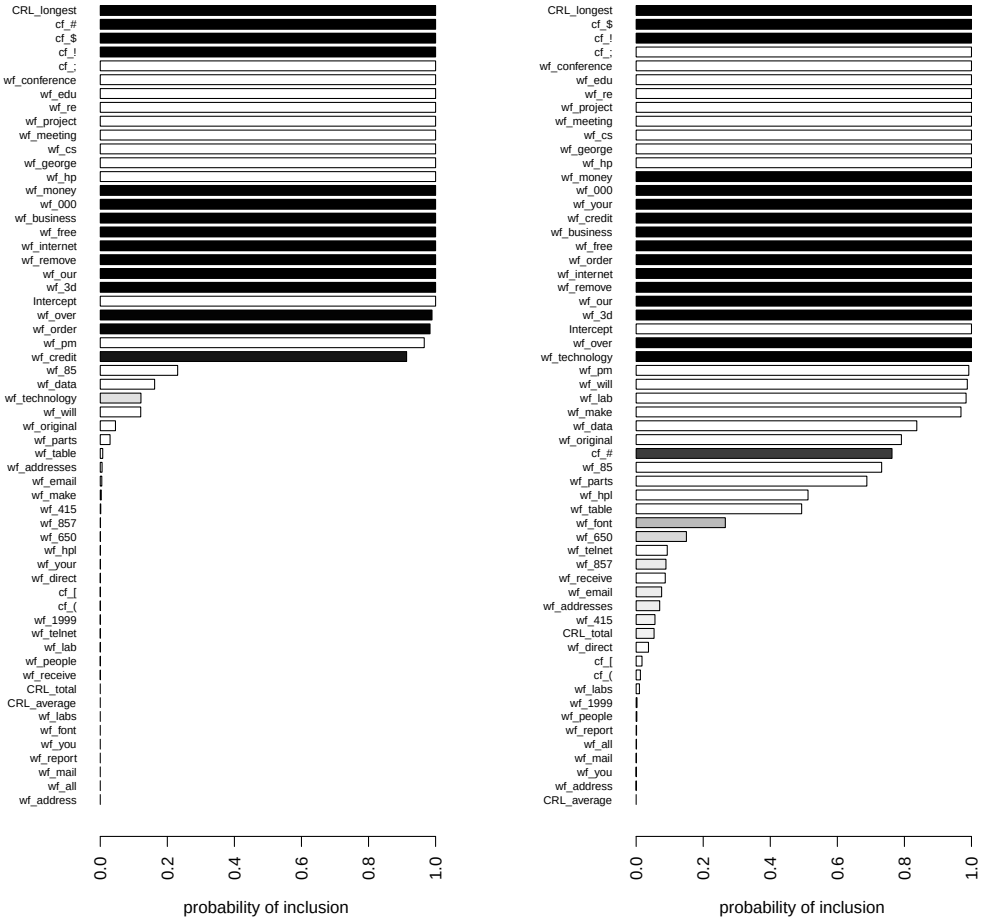
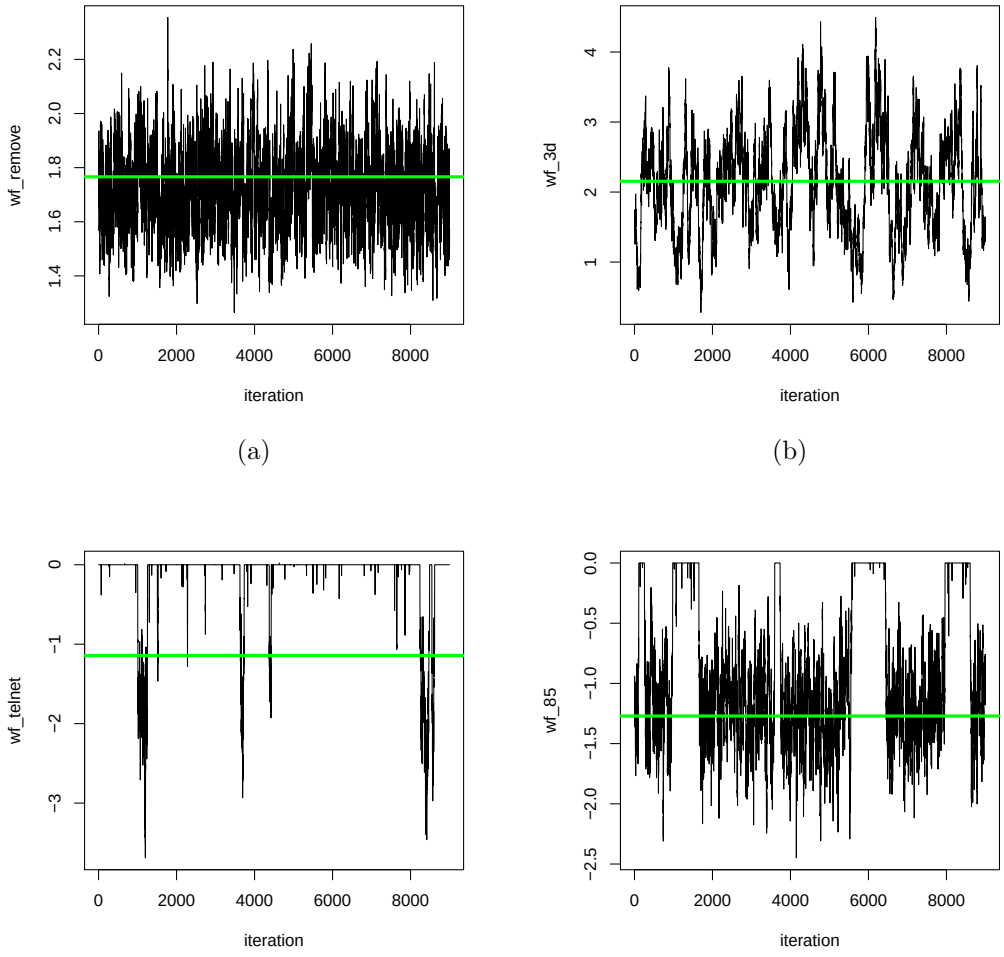


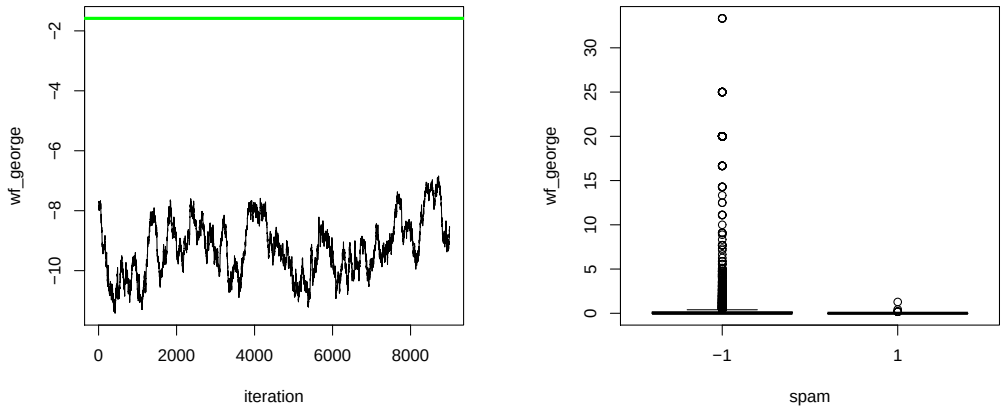
Figure 3: *Inclusion probabilities for spike and slab SVM's fit to the spam data set with (a)* $\pi = .01$ *and (b)* $\pi = .5$*. The bars are shaded in proportion to* $Pr(\beta_j > 0|y)$*, so the darker the bar the greater the probability the variable is positively associated with spam.*



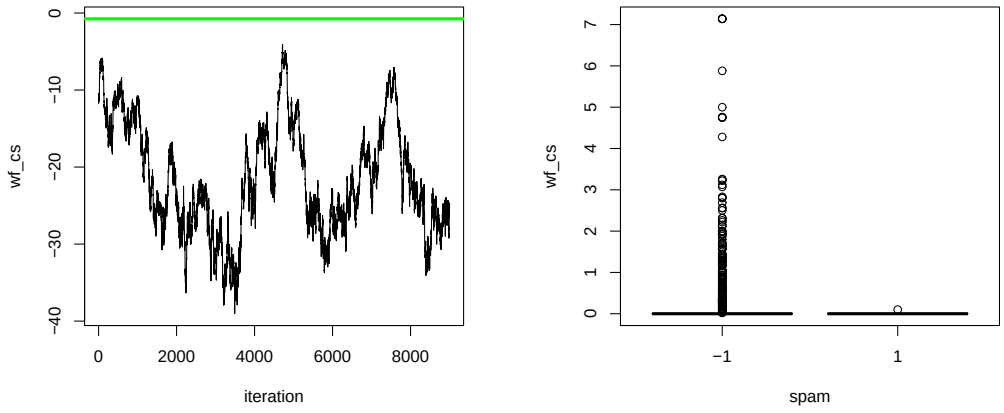
(a) (b)

(c) (d)

Figure 4: *Sample paths from the spike-and-slab sampler with* $\nu = 100$ *and* $\pi = .5$. *The horizontal line is the point estimate from the ECME algorithm that jointly estimated* β *and* ν.



(a) (b)



(c) (d)

Figure 5: *Panels (a) and (c) show MCMC sample paths for the only two coefficients where MCMC disagrees with the point estimates from ECME (shown by the horizontal line). Panels (b) and (d) describe the distribution of the predictor variables for spam and non-spam cases. Both variables are strong signals against spam.*

6 Discussion

At first sight, the hinge objective function $\max(1 - y_i x_i'\beta, 0)$ for SVM's seems to make traditional Bayesian analysis hard, but we have shown that the pseudo-likelihood for SVM's can be expressed as a mixture of normal distributions that allow SVM's to be analyzed using familiar tools developed for Gaussian linear models. We have developed an EM algorithm for locating point estimates of regularized support vector machine coefficients, and an MCMC algorithm for exploring the full pseudo-posterior distribution. The MCMC algorithm allows useful prior distributions that have been developed for Gaussian linear models, such as spike-and-slab priors, to be used with SVM's in an automatic way. These priors have an established track record of good performance in Bayesian variable selection problems. Similar benefits can be expected for SVM's. Extending our methods to hierarchical Bayesian SVM models and nonlinear generalizations is a direction for future research.

References

Andrews, D. F. and Mallows, C. L. (1974). "Scale Mixtures of Normal Distributions." *Journal of the Royal Statistical Society, Series B: Methodological*, 36: 99–102. 4, 5

Carlin, B. P. and Polson, N. G. (1991). "Inference for Nonconjugate Bayesian Models Using the Gibbs Sampler." *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 19: 399–405. 5

Cawley, G. C. and Talbot, N. L. C. (2005). "Constructing Bayesian formulations of sparse kernel learning methods." *Neural Networks*, 18(5-6): 674–683. 11

Clyde, M. and George, E. I. (2004). "Model uncertainty." *Statistical Science*, 19: 81–94. 3

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). "Maximum likelihood from incomplete data via the EM algorithm (C/R: p22-37)." *Journal of the Royal Statistical Society, Series B, Methodological*, 39: 1–22. 8

Devroye, L. (1986). *Non-uniform Random Variate Generation*. Springer-Verlag. URL http://cg.scs.carleton.ca/~luc/rnbookindex.html 6

Fan, J. and Li, R. (2001). "Variable Selection Via Nonconcave Penalized Likelihood and Its Oracle Properties." *Journal of the American Statistical Association*, 96(456): 1348–1360. 3

George, E. I. and McCulloch, R. E. (1993). "Variable Selection Via Gibbs Sampling." *Journal of the American Statistical Association*, 88: 881–889. 1, 13

— (1997). "Approaches for Bayesian Variable Selection." *Statistica Sinica*, 7: 339–374. 1, 3, 13

Gold, C., Holub, A., and Sollich, P. (2005). "Bayesian approach to feature selection and parameter tuning for support vector machine classifiers." *Neural Networks*, 18(5-6): 693–701. 11

Goldstein, M. and Smith, A. F. M. (1974). "Ridge-type Estimators for Regression Analysis." *Journal of the Royal Statistical Society, Series B: Methodological*, 36: 284–291. 4

Golub, G. H. and van Loan, C. F. (2008). *Matrix Computations*. John Hopkins Press, third edition. 10

Gomez-Sanchez-Manzano, E., Gomez-Villegas, M. A., and Marin, J. M. (2008). "Multivariate exponential power distributions as mixtures of normals with Bayesian applications." *Communications in Statistics*, 37(6): 972–985. 4

Greene, W. H. and Seaks, T. G. (1991). "The restricted least squares estimator: a pedagogical note." *The Review of Economics and Statistics*, 73(3): 563–567. 10

Griffin, J. E. and Brown, P. J. (2005). "Alternative Prior Distributions for Variable Selection with very many more variables than observations." (working paper available on Google scholar). 3

Hans, C. (2009). "Bayesian lasso regression." *Biometrika*, 96(4): 835–845. 4, 13

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer, second edition. 2, 14

Holmes, C. C. and Held, L. (2006). "Bayesian Auxiliary Variable Models for Binary and Multinomial Regression." *Bayesian Analysis*, 1(1): 145–168. 3

Holmes, C. C. and Pintore, A. (2006). "Bayesian Relaxation: Boosting, the Lasso and other L^{α}-norms." In Bernardo, J. M., Bayarri, M. J., Berger, J. O., Dawid, A. P., Heckerman, D., Smith, A. F. M., and West, M. (eds.), *Bayesian Statistics 8*, 253 – 283. Oxford University Press. 4

Huang, J., Horowitz, J., and Ma, S. (2008). "Asymptotic properties of Bridge estimators in sparse high-dimensional regression models." *The Annals of Statistics*, 36: 587–613. 4

Ishwaran, H. and Rao, J. S. (2005). "Spike and Slab Gene Selection for multigroup microarray data." *Journal of the American Statistical Association*, 100: 764–780. 1

Johnstone, I. M. and Silverman, B. W. (2004). "Needles and Straws in Haystacks: Empirical Bayes Estimates of Possibly Sparse Sequences." *The Annals of Statistics*, 32(4): 1594–1649. 13

— (2005). "Empirical Bayes Selection of Wavelet Thresholds." *The Annals of Statistics*, 33(4): 1700–1752. 13

Liu, C. and Rubin, D. B. (1994). "The ECME Algorithm: A Simple Extension of EM and ECM With Faster Monotone Convergence." *Biometrika*, 81: 633–648. 10

Mallick, B. K., Ghosh, D., and Ghosh, M. (2005). "Bayesian classification of tumours by using gene expression data." *Journal of the Royal Statistical Society, Series B, Statistical Methodology*, 67(2): 219–234. 1, 6, 11, 12

Meng, X.-L. and Rubin, D. B. (1993). "Maximum Likelihood Estimation Via the ECM Algorithm: A General Framework." *Biometrika*, 80: 267–278. 10

Meng, X.-L. and van Dyk, D. A. (1999). "Seeking efficient data augmentation schemes via conditional and marginal augmentation." *Biometrika*, 86(2): 301–320. 2

Mitchell, T. J. and Beauchamp, J. J. (1988). "Bayesian Variable Selection in Linear Regression (C/R: P1033-1036)." *Journal of the American Statistical Association*, 83: 1023–1032. 3

Neal, R. M. (2003). "Slice Sampling." *The Annals of Statistics*, 31(3): 705–767. 12

Pollard, H. (1946). "The representation of $e^{-x^{\lambda}}$ as a Laplace integral." *Bull. Amer. Math. Soc.*, 52(10): 908–910. 4

Polson, N. G. (1996). "Convergence of Markov Chain Monte Carlo Algorithms." In Bernardo, J. M., Berger, J. O., Dawid, A. P., and Smith, A. F. M. (eds.), *Bayesian Statistics 5 – Proceedings of the Fifth Valencia International Meeting*, 297–321. Clarendon Press [Oxford University Press]. 2

Pontil, M., Mukherjee, S., and Girosi, F. (1998). "On the Noise Model of Support Vector Machine Regression." *A.I. Memo, MIT Artificial Intelligence Laboratory*, 1651: 1500–1999. 6

Sollich, P. (2001). "Bayesian methods for support vector machines: evidence and predictive class probabilities." *Machine Learning*, 46: 21–52. 11

Tibshirani, R. (1996). "Regression Shrinkage and Selection Via the Lasso." *Journal of the Royal Statistical Society, Series B: Methodological*, 58: 267–288. 3, 4

Tipping, M. E. (2001). "Sparse Bayesian learning and the Relevance Vector Machine." *Journal of Machine Learning Research*, 1: 211–244. 11

Tropp, J. A. (2006). "Just relax: Convex programming methods for identifying sparse signals." *IEEE Info. Theory*, 55(2): 1039–1051. 13

West, M. (1987). "On Scale Mixtures of Normal Distributions." *Biometrika*, 74: 646–648. 4

Zhu, J., Saharon, R., Hastie, T., and Tibshirani, R. (2004). "1-norm Support Vector Machines." In Thrun, S., Saul, L. K., and Schoelkopf, B. (eds.), *Advances in Neural Information Processing 16*, 49–56. 3