Separating Sparse Signals from Correlated Noise in Binary Classification

Stephan Mandt[*]
Data Science Institute
Columbia University

Florian Wenzel[*]
CS Department
Humboldt U Berlin

Shinichi Nakajima
Berlin Big Data Center
Technical U Berlin

Christoph Lippert
Human Longevity, Inc
Mountain View

Marius Kloft
CS Department
Humboldt U Berlin

Abstract

Among the goals of statistical genetics is to find sparse associations of genetic data with binary phenotypes, such as heritable diseases. Often, the data are obfuscated by confounders such as age, ancestry, or population structure. A widely appreciated modeling paradigm which corrects for such confounding relies on linear mixed models. These are linear regression models with correlated noise, where the noise covariance captures similarities between the samples. We generalize this modeling paradigm to binary classification. We thereby face the technical challenge that that marginalizing over the noise leads to an intractable, high-dimensional integral. We propose a variational EM algorithm to overcome this problem, where the global model parameters are ℓ_1-norm regularized, leading to a sparse solution. The selected features are much less affected by the spurious correlations in the data, manifested by a smaller correlation between the features and the first principal component of the noise covariance. The proposed method also outperforms Gaussian process classification and uncorrelated probit regression in terms of prediction performance. In addition, we discuss ongoing work on employing stochastic gradient MCMC for this problem class.

1 Introduction

Genetic association studies have emerged as an important branch of statistical genetics [1, 2]. The goal of this field is to find causal associations between high-dimensional vectors of *genotypes*, such as single nucleotide polymorphisms (SNPs), and observable outcomes or *phenotypes*. These phenotypes may be continuous or binary, an example being the outcome of a certain disease. For various complex diseases, such as bipolar disorder or type 2 diabetes [3], the relevant causal mutations along the genome are yet largely undetected [1], and thus have been entitled *The Dark Matter of Genomic Associations* [4].

Genetic associations can be spurious, unreliable, and unreproducible when the data are subject to confounding [5, 6, 7]. Confounding can stem from varying experimental conditions and demographics such as age, ethnicity, or gender [8]. The perhaps most important type of confounding in statistical genetics arises due to population structure [9], as well as similarities between closely related samples [8, 10, 11]. Ignoring such confounders can often lead to spurious false positive findings that cannot be replicated on independent data [12]. Correcting for such confounding dependencies is considered one of the greatest challenges in statistical genetics [13].

A popular approach of correcting for spurious correlations in statistical genetics is based on linear mixed models (LMMs) [10]. These are essentially linear regression models with multivariate noise. LMMs account for a linear effect of genotypes on the phenotypes which is assumed to be sparse, motivated by the idea that most genetic mutations do not contribute to the phenotypes of interest. LMMs also include a weak noise contribution along the genes, which translates to correlated noise in the space of samples. This models the relatedness between individuals in the overall population. The resulting model thus aims to find a sparse linear weight vector while automatically accounting for spurious correlations due to relatedness between samples [5, 9].

Although successful, LMMs have been restricted to the linear regression case. We generalize this modeling paradigm to the case of binary classification. Probit regression forms the basis of our approach [14], where we add an ℓ_1-norm (Lasso) regularizer that guarantees that the resulting weight vectors are sparse [15]. However, in contrast to simple probit regression (and following the logic of LMMs), we consider a correlated multivariate noise variable that correlates the binary labels. This way, our approach generalizes

[*] Equal contributions. Contact the authors by sm3976@columbia.edu or wenzelfl@hu-berlin.de

two popular methods which result as limiting cases: ℓ_1-norm probit regression (for uncorrelated noise), and Gaussian process (GP) probit classification [16] (when the linear weight vector is zero).

Our more complex model suffers from intractable inference in high dimensions, and we therefore have to resort to approximations. We make use of variational Bayesian methods and propose two algorithms. Our first algorithm operates in sample space and makes use of approximate Gaussian quadrature [17]. Our second algorithm operates in feature space and is amenable to data subsampling and therefore scalable to very large sample sizes. Which algorithm is preferable depends on the number of data points and the feature space dimensionality.

In an experimental study on genetic data, we show the superiority of our approach over other methods. Compared to uncorrelated probit regression, our sparse features are up to 40% less correlated with the first principal component of the noise covariance that induces the spurious correlations we seek to suppress. Furthermore, compared to the LMM Lasso [18], probit regression, and GP classification [16], our approach yields up to 5% higher prediction accuracies. In a computer malware experiment we show that our approach generalizes beyond statistical genetics.

Our paper is organized as follows. Section 2 introduces the modeling framework. We first discuss the confounding problem in genetics and introduce two versions of our model: a simplified version based on a maximum-likelihood estimate of the noise variable, and the fully correlated model. Section 3 then contains the mathematical details of the inference procedure. In Section 4 we then apply our method to extract features associated with diseases and traits from confounded genetic data. We also test our method on a data set that contains a mix of different types of malicious computer software data.

2 Correlated Probit Regression

We first review the problem of spurious correlations due to population structure in statistical genetics in Section 2.1. In Section 2.2, we review LMMs and introduce a corresponding model for classification. In Section 2.3 we connect our approach with other models.

2.1 Modeling Spurious Correlations via Kernels

The problem of spurious correlations is fundamental in statistics. Spurious correlations may be due to confounding or selection bias. Confounding is induced by a common unobserved cause that underlies both the predictor variables and the traits. Selection bias emerges arises from taking non-random subsets of the population, where some members are less likely to be subsampled than others [19]. Both effects may result in the phenomena of spurious correlations which we can treat here simultaneously [5, 6, 7].

Population structure [9] implies that due to common ancestry, genes of individuals that are related co-inherit a large number of genes, making them more similar to each other, whereas the genes of people of unrelated ancestry are obtained independently, making them more dissimilar. Population structure is the root of many unwanted biases. For example, when data is collected only in selected geographical areas (such as in specific hospitals), one thereby introduces a selection bias into the sample, meaning that the collected genes do not represent the overall population. This can heavily distort the prediction quality of a classifier [13].

Another problem is that people who live geographically close often share other factors, such as similar environmental factors or culture. This, in turn, can lead to similar phenotypes (such as overweight, drinking habits, or diabetes). Thus, because genes correlate with location and location may correlate with specific phenotypes, there is a resulting correlation between genes and these phenotypes that does not have a causal interpretation—another manifestation of confounding by population structure [20]. It is an active area of research to find models that are less prone to spurious correlations [13]. In this paper, we present such a model for the setup of binary classification.

A popular approach to correcting for spurious correlations relies on similarity kernels, or kinship matrices [9]. Given n samples, we can construct an $n \times n$ matrix K that quantifies the similarity between samples based on some arbitrary measure. In the case of confounding by population structure, one typically chooses $K_{ij} = X_i^\top X_j$, where X_i is a vector of genetic features of individual i. As K_{ij} contains the scalar products between the genetic vectors of individuals i and j, it is a sensible measure of genetic similarity. As another example, when correcting for confounding by age, then we can choose K to be a matrix that contains 1 if two individuals have the same age, and zero otherwise. Details of constructing similarity kernels can be found in [9]. Next, we explain how the similarity matrix can be used to correct for confounding.

2.2 The Correlated Probit Regression (CPR) Model

Our model builds on the LMM-Lasso [18], an important method of statistical genetics to limit the impact of confounding. While the LMM-Lasso relies on linear regression, we generalize this approach to the much more involved classification setup, where the target values are binary. The correlated probit model is

$$y_i = \operatorname{sign}\left(X_i^\top w + \epsilon_i\right), \;\; \epsilon = (\epsilon_1, \ldots, \epsilon_n)^\top \sim \mathcal{N}(0, \Sigma). \qquad (1)$$

In the special case of $\Sigma = \mathbf{I}$, this is just the ordinary (uncorrelated) probit model. In the following, we refer to this model as *Correlated Probit Regression* (CPR). For now, we assume that the covariance matrix Σ is fixed and known.

In our empirical studies we use the parametrization Eq. 13 where the parameters are estimated from the data.

We now derive an objective function to find an estimate of the model parameter w. To simplify the notation, we will without loss of generality assume that *all observed binary labels y_i are 1*. The reason why this assumption is no constraint is that we can always perform a linear transformation to absorb the sign of the labels into the data matrix and noise covariance[1]. Thus, when working with this transformed data matrix and noise covariance, our assumption is satisfied.

The likelihood function, thus, is the probability that all transformed labels are 1. This is satisfied when $X_i^\top w + \epsilon_i > 0$. When integrating over all realizations of noise, the resulting (marginal) likelihood is

$$\begin{aligned} \mathbb{P}(\forall i : y_i = 1|w) &= \mathbb{P}(\forall i : X_i^\top w + \epsilon_i > 0|w) \\ &= \int_{\mathbb{R}_+^n} \mathcal{N}(\epsilon; X^\top w, \Sigma)\, d^n\epsilon. \end{aligned} \tag{2}$$

The marginal likelihood is hence an integral of the multivariate Gaussian over the positive orthant. In Section 3, we will present efficient approximations of this integral. Before we get there, we further characterize the model.

Next, we turn the correlated probit model into a model for feature selection. We are interested in a point estimate of the weight vector w that is sparse, *i.e.* contains zeros almost everywhere. This is well motivated in statistical genetics for phenotypes or diseases that are believed to be caused by a small number of genes. Sparsity is achieved using the Lasso [15], where we add an ℓ_1-norm regularizer to the negative marginal likelihood:

$$\mathcal{L}(w) = -\log \int_{\mathbb{R}_+^n} \mathcal{N}(\epsilon; X^\top w, \Sigma)\, d^n\epsilon \;+\; \lambda_0 \|w\|_1^1. \tag{3}$$

The fact that the noise variable ϵ and the weight vector w have different priors or regularizations makes the model identifiable and lets us cleanly distinguish between linear effects and effects of correlated noise. It is easy to show that the objective function Eq. 3 is convex.

2.3 Connection to Other Models

Before we come to inference, we point out how our approach connects to other methods. When removing the probit likelihood, the model becomes the LMM-Lasso [18], hence $\mathbb{P}(Y|w) = \mathcal{N}(Y; X^\top w, \Sigma)$. This model has shown to improve selection of true non-zero effects as well as prediction quality [18]. Our model is a natural extension of the LMM-Lasso to binary outcomes, such as the disease status of a patient. As we explain in this paper, inference of our model is, however, much more challenging than in [18].

[1] To this end, we apply the transformations $X \leftarrow \mathrm{diag}(y)X$ and $\Sigma \leftarrow \mathrm{diag}(y)\Sigma\mathrm{diag}(y)$.

Furthermore, by construction, our model captures two limiting cases: uncorrelated probit regression and Gaussian process (GP) classification. To obtain uncorrelated probit regression, we simply assume a covariance matrix proportional to unity. To obtain GP classification, we simply omit the fixed effect (i.e., we set $w = 0$ in Eq. 2) so that our model likelihood becomes $\mathbb{P}(Y = Y^{\mathrm{obs}}|w) = \int_{\mathbb{R}_+^n} \mathcal{N}(\epsilon; 0, \Sigma)\, d^n\epsilon$, where hence the noise variable ϵ plays the role of the latent function f in GPs [16]. We will compare our method to all three related models in the experimental part of the paper and show enhanced accuracy.

3 Inference Algorithms

We derive three different algorithms to do inference in the correlated probit model. We want to optimize the objective function of CPR Eq. 3. This goal comes along with two major problems:

1. The ℓ_1-norm regularizer for feature selection is not differentiable everywhere.
2. The likelihood contains an intractable, high-dimensional integral.

Our first algorithm, *CPR*, directly optimizes Eq. 3 by employing expectation propagation (EP) [21] and the alternating direction method of multipliers (ADMM) [22]. We note that other means of approximate inference, such as MCMC for truncated Gaussian distributions [23] are also viable options. Because this algorithm relies on approximating moments of a truncated Gaussian integral in n dimensions, it is restricted by the dimensionality of the data space n.

We also propose two methods that scale more favorably with n, but have other constraints. Our second proposed method relies on a MAP-approximation of the confounder instead of marginalizing it out. Although it is very fast, its prediction performance is substantially worse than CPR, as we show experimentally. Our third method, *Stochastic Gradient Correlated Probit Regression* (*SG-CPR*), has the same benefits of scalability. It makes use of recent breakthroughs in scalable MCMC methods [24, 25, 26]. As this algorithm samples in feature space, its performance depends on the dimensionality d.

3.1 Algorithm 1: CPR

CPR is a EM-type algorithm [27]. In the outer loop (the M-step), we follow gradients to optimize the objective. Since this objective function has an ℓ_1-norm regularizer, we have to split this outer optimization routine into two parts, one that optimizes the likelihood and one that optimizes the regularizer. This is described in 3.1.1. The inner loop (the E-step) consists of computing the gradient and the Hessian of the likelihood term by means of approximate inference, which is described in Sections 3.1.2 and 3.1.3.

3.1.1 M-step

The ℓ_1-norm in the objective function Eq. 3 prevents us from directly applying gradient based methods such as Newton's method. A solution is given by ADMM that involves a generalized objective:

$$\mathcal{L}(w, z, \eta) := -\log \int_{\mathbb{R}_+^N} \mathcal{N}(\epsilon; X^\top w, \Sigma) d^n\epsilon + \lambda_0 \|z\|_1^1 + \eta^\top (w - z) + \frac{1}{2} c \|w - z\|_2^2.$$

We minimize over w and z and maximize over η. In alternating between the minimization updates for w, z and a gradient step in η, we solve the original problem [22]. While the updates for z and η have analytic solutions, we compute the updates for w by numerical optimization. The part of the ADMM objective $\mathcal{L}(w, z, \eta)$ depending on w, called $\mathcal{L}(w)$ for brevity, is effectively ℓ_2-norm regularized, enabling us to compute the gradient and the Hessian. This allows us to apply Newton's Method to obtain the ADMM update in w.

3.1.2 E-step

The inner loop of the EM-algorithm amounts to computing the gradient and Hessian of $\mathcal{L}(w, z, \eta)$. These are not available in closed-form, but in terms of the first and second moment of a truncated Gaussian density.

Since computing the derivates of the linear and quadratic term is straightforward, we focus on $\mathcal{L}_0(w) := -\log \int_{\mathbb{R}_+^N} \mathcal{N}(\epsilon; X^\top w, \Sigma) d^n\epsilon$, which contains the intractable integral. In the following, we use the short hand notation

$$\mu \equiv \mu(w) = X^\top w. \tag{4}$$

It is convenient to introduce the following probability distribution:

$$p(\epsilon|\mu, \Sigma) = \frac{\mathbf{1}[\epsilon \in \mathbb{R}_+^n] \, \mathcal{N}(\epsilon; \mu, \Sigma)}{\int_{\mathbb{R}_+^n} \mathcal{N}(\epsilon; \mu, \Sigma) \, d^n\epsilon}. \tag{5}$$

Above, $\mathbf{1}[\cdot]$ is the indicator function. Eq. 5 is just the multivariate Gaussian, truncated and normalized to the positive orthant; we call it the *posterior* distribution. We furthermore introduce

$$\begin{aligned} \mu_p(w) &= \mathbb{E}_{p(\epsilon|\mu(w),\Sigma)}[\epsilon], \\ \Sigma_p(w) &= \mathbb{E}_{p(\epsilon|\mu(w),\Sigma)}\left[(\epsilon - \mu_p(w))(\epsilon - \mu_p(w))^\top\right]. \end{aligned} \tag{6}$$

μ_p and Σ_p are the mean and the covariance of the *truncated* multivariate Gaussian, as opposed to μ and Σ which are the non-truncated ones.

In the following we abbreviate $\mu_p \equiv \mu_p(w)$ and $\Sigma_p \equiv \Sigma_p(w)$, and write $\Delta\mu = \mu_p - \mu$ for the difference between the means of the posterior (the truncated Gaussian) and the untruncated Gaussian. The gradient and Hessian of $\mathcal{L}_0(w)$ are given by

$$\begin{aligned} \nabla_w \mathcal{L}_0(w) &= \Delta\mu \Sigma^{-1} X^\top, \\ H_0(w) &= -X[\Sigma^{-1}(\Sigma_p - \Delta\mu\Delta\mu^\top)\Sigma^{-1} - \Sigma^{-1}]X^\top. \end{aligned} \tag{7}$$

Note that the variable w enters through $\Sigma_p(w)$ and $\Delta\mu(w)$. Next, we describe how we approximate the intractable expectations involved in Eq. 7.

3.1.3 Optimizing the Objective Function

In Eq. 7 we have expressed the gradient and Hessian of $\mathcal{L}_0(w)$ in terms of the first and second moment of the posterior Eq. 5. The problem is that computing the moments involves intractable expectations over this distribution. We employ EP [17] to approximate these expectations. Note that also other approximate inference schemes are possible, such as variational inference or sampling methods [16].

EP approximates moments of the posterior $p(\epsilon|\mu, \Sigma)$ in terms of a variational distribution $q(\epsilon)$, approximately minimizing the Kullback-Leibler divergence,

$$q^*(\epsilon|\mu_{q^*}, \Sigma_{q^*}) = \arg\min_q \Big(\mathbb{E}_p[\log p(\epsilon|\mu, \Sigma)] - \mathbb{E}_p[\log q(\epsilon|\mu_q, \Sigma_q)]\Big).$$

The variational distribution $q^*(\epsilon)$ is an un-truncated Gaussian $q^*(\epsilon; \mu_{q^*}, \Sigma_{q^*}) = \mathcal{N}(\epsilon; \mu_{q^*}, \Sigma_{q^*})$, characterized by the variational parameters μ_{q^*} and Σ_{q^*}. We approximate the mean and covariance of the posterior p in terms of the variational distribution, $\mu_p \approx \mu_{q^*}$, and $\Sigma_p \approx \Sigma_{q^*}$. We warm-start each gradient computation with the optimal parameters of the earlier iteration.

Algorithm 1 summarizes our procedure. We denote the EP algorithm for approximating the first and second moment of the truncated Gaussian by EP(μ, Σ). Here, μ and Σ are the mean and covariance matrix of the un-truncated Gaussian. The subroutine returns the first and second moments of the truncated distributions μ_q and Σ_q. When initialized with the outcomes of earlier iterations, this subroutine converges within a single EP loop.

Our algorithm thus consists of three nested loops; the outer ADMM loop, containing the Newton optimization loop for computing the update in w and the inner EP loop, which computes the moments of the posterior. We choose stopping *criterion 1* to be the convergence criterion proposed by Boyd [22] and choose *criterion 2* to be always fulfilled, i. e. we perform only one Newton optimization step in the inner loop. Our experiments showed that doing only one Newton optimization step, instead of executing until convergence, is stable and leads to great speed improvements. ADMM is known to converge even when the minimizations in the ADMM scheme are not carried out exactly (see e.g. [28]).

Algorithm 1 CPR

$X = y \circ \tilde{X}$ *pre-process the data*
$\Sigma = \mathrm{diag}(y)\,\tilde{\Sigma}\,\mathrm{diag}(y)$
repeat
  get w^{k+1} by EP and Newton's Method
  initialize w = w^k
  repeat
    $(\mu_q, \Sigma_q) \leftarrow \mathrm{EP}(X^\top w, \Sigma)$
    $\Delta\mu = \mu_q - X^\top w$
    $g = \Delta\mu^\top \Sigma^{-1} X^\top + c(w - z^k + \eta^k)^\top$
    $H = X[\Sigma^{-1} - \Sigma^{-1}(\Sigma_q - \Delta\mu\Delta\mu^\top)\Sigma^{-1}]X^\top + c\mathbf{I}$
    $w = w - \alpha_t H^{-1} g$
  until criterion 2
  ADMM updates
  $w^{k+1} = w$
  $z^{k+1} = S_{\lambda/c}(w^{k+1} + \eta^k)$ *soft thresholding*
  $\eta^{k+1} = \eta^k + w^{k+1} - z^{k+1}$
until criterion 1

3.2 Algorithm 2: CPR-MAP

For simplicity, we use covariance matrices Σ of a special structure, which allows us to derive an alternative formulation of the correlated probit model[2]. In particular, we assume that Σ is a combination of diagonal noise and a linear kernel of the data matrix,

$$\Sigma \quad = \quad \lambda_1 \mathbf{I} + \lambda_2 X^\top X. \tag{8}$$

The linear kernel $X^\top X$ measures similarities between genes and therefore models the effect of genetic similarity between samples due to population structure. We use the following Gaussian integral identity:

$$\begin{aligned}
\mathcal{L}(w) &= -\log \int_{\mathbb{R}^n_+} \mathcal{N}(\epsilon; X^\top w, \lambda_1 \mathbf{I} + \lambda_2 X^\top X)\, d\epsilon \;+\; \lambda_0 \|w\|_1^1 \\
&= -\log \int_{\mathbb{R}^d} \mathcal{N}(w'; 0, \lambda_2 \mathbf{I}) \int_{\mathbb{R}^n_+} \mathcal{N}(\epsilon; X^\top (w + w'), \lambda_1 \mathbf{I}) d\epsilon\, dw' \\
&\quad + \;\lambda_0 \|w\|_1^1 \\
&\stackrel{c}{=} \log \int_{\mathbb{R}^d} p(y, w'|w)\, dw' \;+\; \lambda_0 \|w\|_1^1,
\end{aligned} \tag{9}$$

where

$$p(y, w'|w) \propto \mathcal{N}(w'; 0, \lambda_2 \mathbf{I}) \prod_{i=1}^{n} \Phi\left(\frac{X_i^\top (w + w')}{\sqrt{\lambda_1}}\right).$$

Above, $\Phi(\cdot)$ is the cumulative standard normal distribution function. We have introduced the new Gaussian noise variable w'. Conditioned on w', the remaining integrals factorize over n. However, since w' is unobserved (hence marginalized out), it correlates the samples. We interpret w' as a confounder.

The simplest approximation to Eq. 9 is to substitute the integral over w' by its maximum a posteriori (MAP) value, leading to the new objective function:

$$\begin{aligned}
\mathcal{L}(w, w') = &- \sum_{i=1}^{n} \log \Phi\left(\frac{X_i^\top (w + w')}{\sqrt{\lambda_1}}\right) \\
&+ \frac{1}{2\lambda_2} \|w'\|_2^2 + \lambda_0 \|w\|_1^1.
\end{aligned} \tag{10}$$

Under the MAP approximation, the likelihood contribution to the objective function becomes completely symmetric in w and w': only the sum $w + w'$ enters. The difference between the two weight vectors w and w' in this approximation is only due to the different regularizers: while w' has an ℓ_2-nrom regularizer and is therefore dense, w is ℓ_1-norm regularized and therefore sparse.

For optimizing the MAP approximated objective function Eq. 10 jointly in w and w', we introduce a block coordinate descent scheme alternating between updates in w and w'. For updating w' we use gradient descent, while for updating w we employ ADMM (c.f. Section 3.1.1). Note that the procedure could be made faster by using a second-order optimization method for obtaining the updates in w'.

Under the MAP approximation, every feature gets a small non-zero weight from w', and only selected features get a stronger weight from w. The idea is that w' models the population structure, which affects all genes. In contrast, we are interested in learning the sparse weight vector w, which has a causal interpretation because it involves only a small number of features.[3]

The MAP approximation is computationally more convenient, but it has its limits. In the original correlated probit model in Eq. 1, we marginalize over the confounder, which is more expensive. In contrast, under the MAP approximation we optimize over w' and the the objective function factorizes over n, which means that we have broken the correlations between the samples. This comes at the cost of reduced prediction performance. Since the MAP estimate of the confounder does not capture the full information of its distribution, the MAP probit model tends to generalize not as well as the (full) correlated probit model. We compare both approaches experimentally in Section 4.

3.3 Algorithm 3: SG-CPR

Stochastic gradient Monte Carlo methods are an active area of research in scalable Bayesian inference. These methods approximately sample from a posterior by using only a

[2]Note that the approach can be easily generalized to arbitrary covariance matrix by considering the Cholesky decomposition $\Sigma = BB^\top$.

[3]Note that the interplay of two weight vectors is different from an elastic net regularizer.

subset of data for generating a sample and, therefore, being scalable to big datasets. This is done by using stochastic optimization to provide efficient proposals for Metropolis-Hastings algorithms with high acceptance rates. We propose two versions of *SG-CPR*, one builds on Stochastic Gradient Langevin Dynamics (SGLD) [24] and the other on Constant Stochastic Gradient Descent (c-SGD) [25]. These methods assume that the likelihood factorizes, conditioned on the global variables. Up to a constant, the log posterior is

$$\log p(\theta|y) \stackrel{c}{=} \sum_{i=1}^{n} \log p(y_i|\theta) + \log p(\theta).$$

Let $\mathcal{S}$ be a set of S random indices drawn uniformly at random from the index set $\{1, \ldots, n\}$. The stochastic gradient w.r.t. the minibatch $\mathcal{S}$ of the log likelihood term is

$$\hat{g}_{\mathcal{S}}(\theta) = \frac{1}{S} \sum_{i \in \mathcal{S}} \nabla_\theta \log p(y_i|\theta).$$

SGLD and c-SGD work as follows. SGLD performs decreasing stochastic gradient step on the negative log joint distribution, but adds artificial noise to prevent convergence to the optimum. Instead, the algorithm converges to a stationary distribution, which can be shown to be the posterior [24]. Constant SGD, on the other hand, only approximates the posterior. It converges faster because it operates with constant step sizes.

The following formula summarizes the two methods:

$$\begin{aligned} \text{SGLD}: \quad \theta_{i+1} &= \theta_i + \frac{\gamma_t}{2} \hat{g}_{\mathcal{S}}(\theta_t) + \eta_t, \quad \eta_t \sim \mathcal{N}(0, \gamma_t), \\ \text{c-SGD}: \quad \theta_{i+1} &= \theta_i + \gamma^* \hat{g}_{\mathcal{S}}(\theta_t), \end{aligned} \tag{11}$$

where γ_t is a suitable decreasing learning rate for SGLD. For c-SGD, it was shown in [25] that the optimal constant learning rate γ^* that best approximates the posterior equals $\frac{2dS}{n\text{Tr}(\mathbb{E}[\hat{g}\hat{g}^T])}$. This definition involves the minibatch size S, feature space dimension d, and the stochastic gradient noise covariance near the optimum, $\mathbb{E}\left[\hat{g}\hat{g}^T\right]$.

We now explain how SGLD and c-SGD can be used for inference in the correlated probit model. Recall that the aim is to find the MAP estimate of the model by optimizing the objective function $\mathcal{L}(w)$ Eq. 3. In Section 3.2, we introduced an auxiliary variable w' and obtained the identity

$$\mathcal{L}(w) \stackrel{c}{=} \log \int_{\mathbb{R}^d} p(y, w'|w)\, dw' \; + \; \lambda_0 \|w\|_1^1. \tag{12}$$

$\mathcal{L}(w)$ can be optimized using the EM-algorithm [27],

E-step: Compute $p(w'|y, w^{t-1})$

M-step: $w^t = \text{argmax}_w \, \mathbb{E}_{p(w'|y,w^{t-1})}[\log p(y, w'|w)] \; + \lambda_0 \|w\|_1^1$,

which involves the posterior of the confounder $p(w'|y, w)$. This posterior can be approximated using SGLD or c-SGD. Since this algorithm is based on stochastic gradient descent *SG-CPR* scales to hundreds of thousands data points.

4 Empirical Analysis and Applications

We studied the performance of our proposed methods in experiments on both artificial and real-world data. We considered the versions *CPR* (the full correlated probit model as specified in Eq. 9) and *CPR-MAP* (its MAP approximation as contained in Eq. 10). An experimental analysis of *SG-CPR* is left to future work.

Our data was taken from the domains of statistical genetics and computer malware prediction. Our achievements can be summarized as follows:

- We compare against 3 competing methods, including uncorrelated probit regression, GP classification and the LMM Lasso. In all considered cases, CPR achieves higher classification performance.
- The features that our algorithm finds are up to 40% less confounded by population structure.
- CPR outperforms its MAP approximation across all considered datasets. Yet, in many cases CPR-MAP is a cheap alternative to the full model.

4.1 General Experimental Setup

For the real-world and synthetic experiments, we first need to make a choice for the class of kernels that we use for the covariance matrix. We choose a combination of three contributions,

$$\Sigma = \lambda_1 \mathbf{I} + \lambda_2 X^\top X + \lambda_3 \Sigma_{\text{side}}. \tag{13}$$

The third term is optional and depends on the context; it is a kernel that we extract from side information in the form of an additional feature matrix X', where we choose Σ_{side} as an RBF kernel [29] on top of the side information X'. Note that this way, the data matrix enters the model both through the linear effect but also through the linear kernel. We evaluate the methods by using n instances of the dataset for training and splitting the remaining examples equally into validation and test sets. This process is repeated 50 times, over which we report on average accuracies or areas under the Receiver Operating Characteristic (ROC) curve (AUCs) as well as standard errors [30].

The hyper parameters λ_k in Eq. 13, together with the regularization parameter λ_0, were determined on the validation set, using grid search over a sufficiently large parameter space (optimal values are attained inside the grid; in most cases $\lambda_k \in [0.1, 1000]$ for $k = 0, 1, 2, 3$). For all datasets, the features were centered and scaled to unit standard deviation, except in Section 4.4, where they are binary.

In Sections 4.3 and 4.4, we show that including a linear kernel into the covariance matrix leads to top features which are less correlated with the population structure in comparison to the features of uncorrelated probit regression. The

correlation plots[4] in Fig. 4 show the mean correlation of the top features with population structure and the corresponding standard errors.

4.2 Simulated Data

We generated $n = 200$ synthetic data points in $d = 50$ dimensions as follows. We generate a weight vector $w \in \mathbb{R}^d$ with k entries being 1, and the other $d - k$ entries being 0, where $1 \leq k \leq d$. We then create a random covariance matrix $\Sigma_{\text{side}} \in \mathbb{R}^{n \times n}$, which serves as side information matrix[5]. We draw n points $X = \{x_1, \dots x_n\}$ independently from a uniform distribution over the unit cube $[-1, 1]^d$ and create the labels according to the probit model Eq. 1, using Σ_{side} as covariance matrix. We reserve 100 samples for training and 50 for validation and testing, respectively. As a benchmark we introduce the *oracle classifier*, where we use the correlated probit model (with covariance matrix Σ_{side}) but skip the training and instead use the true underlying w for prediction.

For several $1 \leq k \leq d$, we generate a dataset according to the above described procedure. In Fig. 1, we report on the so-achieved accuracies with respect to the percentage of non-zero features ($\frac{k}{d}$). We observe that in the sparse scenarios ($\leq 20\%$ non-zero features), GP classification and LMM-Lasso are clearly outperformed by CPR, achieving an accuracy up to 10 percentage points and 23 percentage points higher, respectively. Due to being ℓ_1-norm regularized and therefore, having the capability of exploiting sparsity, uncorrelated probit regression performs best in this regime among the competitors, but still substantially worse than CPR. LMM-Lasso is also ℓ_1-norm regularized but is not designed for a classification setting. Therefore, it cannot beat uncorrelated probit regression. In the dense scenarios, CPR outperforms LMM-Lasso (by 1 to 4 percentage points) and performs similarly well to GP classification, which also takes the correlation structure into account. In this scenario, uncorrelated probit regression is clearly worse than the other methods, because it does not take the correlation structure into account. We observe that in all scenarios the prediction performance of CPR-MAP is between uncorrelated probit regression and CPR.

In Fig. 2, we inspect the computed feature weights (green dots) of ℓ_1-norm regularized and ℓ_2-norm regularized CPR, respectively. The blue solid line represents the ground truth (the true underlying weight vector w with $k = 10$ entries non-zero). We observe that the ℓ_1-norm regularized probit model finds the true weights without suffering from large noise as the ℓ_2-norm regularized counterpart does.

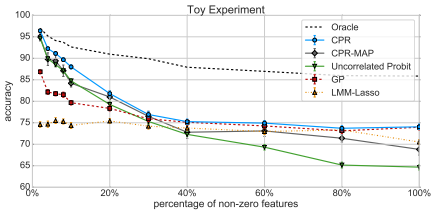

Figure 1: Toy: Average accuracies as a function of the number of true non-zero features in the generating model. (Proposed methods: CPR and CPR-MAP)

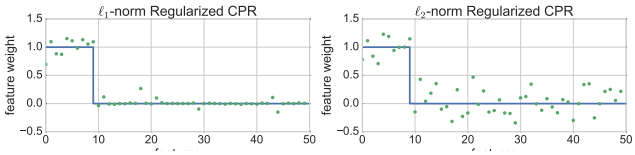

Figure 2: Toy: Ground truth (blue solid line) and feature weights (green dots) of ℓ_1-norm (Left) and ℓ_2-norm (Right) regularized correlated probit regression.

4.3 Tuberculosis Disease Outcome Prediction From Gene Expression Levels

We obtained the dataset by [32] from the National Center for Biotechnology Information website [6], which includes 40 blood samples from patients with active tuberculosis as well as 103 healthy controls, together with the transcriptional signature of blood samples measured in a microarray experiment with 48,803 gene expression levels, which serve as features for our purposes. Also available is the age of the subjects when the blood sample was taken, from which we compute Σ_{side}[7]. All competing methods are trained by using various training set sizes $n \in [40, 80]$. To be consistent with previous studies (e. g. [8]), we report on the area under the ROC curve (AUC), rather than accuracy, where we vary the hyperparameters λ_k. The results are shown in Fig. 3, left.

We observe that CPR achieves a consistent improvement over its uncorrelated counterpart (by up to 12 percentage

[4]The correlation plots in Fig. 4 are created according to [8] as follows. First, we randomly choose 70% of the available data as training set and obtain a weight vector w by training. We compute the empirical Pearson correlation coefficient of each feature with the first principle component of the linear kernel on top of the data. This is a way to measure the correlation with the population structure [31]. We define the index set I by taking the absolute value of each entry of w and sorting them in descending order. We now sort the so-obtained list of correlation coefficients with respect to the index set I and obtain a resorted list of correlation coefficients $(c_1, \dots, c_n)$. In the last step, we obtain a new list $(\hat{c}_1, \dots, \hat{c}_n)$ by smoothing the values, computing $\hat{c}_i := \frac{1}{i} \sum_k^i c_k$. Finally, we plot the values $(\hat{c}_1, \dots, \hat{c}_n)$ with respect to I. This procedure was repeated 30 times for different random choices of training sets.

[5]The covariance matrix was created as follows. The random generator in MATLAB version 8.3.0.532 was initialized to seed = 20 using the `rng(20)` command. The matrix Σ_{side} was realized in two steps via `A=2*rand(50,n)-1` and Σ_{side}=`3*A'*A+0.6*eye(200)+3*ones(200,200)`.

[6]`http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE19491`

[7]We compute Σ_{side} as RBF kernel on top of the side information age using bandwidth $\sigma = 0.2$.

points), GP classification (by up to 3 percentage points), LMM-Lasso (by up to 7 percentage points) and CPR-MAP (by up to 7 percentage points). In Fig. 4, left, we show the correlation of the top features with population structure (as confounding factor) for correlated and uncorrelated probit regression. The plot was created as explained in Section 4.1. We find that the features obtained by CPR show much less correlation with population structure than the features of uncorrelated probit regression. By inspecting the correlation coefficients of the first top 10 features of both methods, we observe that the features found by CPR are 40 % less correlated with the confounder. This is because population structure was built into our model as a source of correlated noise.

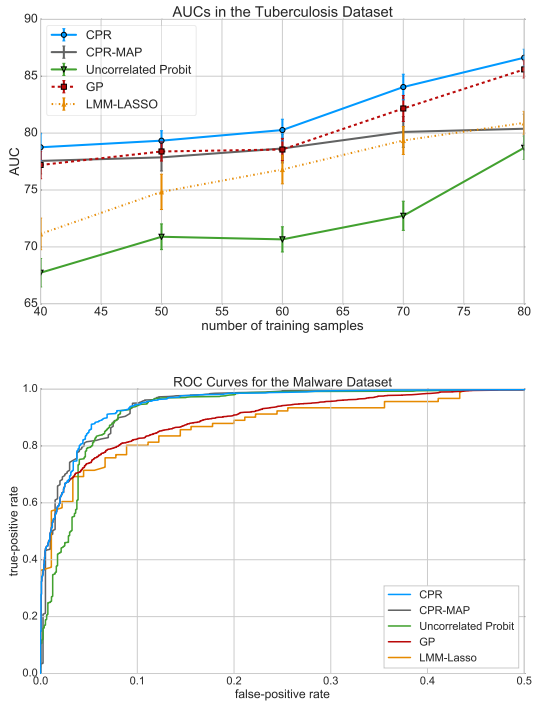

Figure 3: Top, TBC: Average AUC in the tuberculosis experiment with respect to the training set size. Bottom, Malware: Average ROC curves for the computer malware detection experiment.

4.4 Malicious Computer Software (Malware) Detection

We experiment on the Drebin dataset[8] [33], which contains 5,560 Android software applications from 179 different malware families. There are 545,333 binary features; each feature denotes the presence or absence of a certain source code string (such as a permission, an API call or a network address). It makes sense to look for sparse feature vector [33], as only a small number of strings are truly characteristic of a malware. The idea is that we consider

[8] `http://user.informatik.uni-goettingen.de/~darp/drebin/download.html`

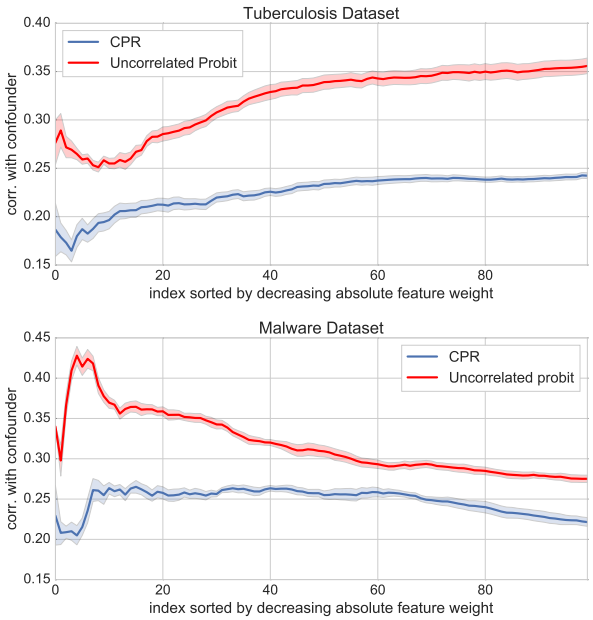

Figure 4: Correlation of the top features in the tuberculosis experiment (Top) and in the computer malware detection experiment (Bottom). The x-axis is sorted by descending absolute weights.

populations of different families of malware when training, and hence correct for the analogue of genetic population structure in this new context, that we call "malware structure". We concentrate on the top 10 most frequently occurring malware families in the dataset[9]. We took 10 instances from each family, forming together a malicious set of 100 and a benign set of another 100 instances (i.e., in total 200 samples). We employ $n = 80$ instances for training and stratify in the sense that we make sure that each training/validation/test set contains 50% benign samples and an equal amount of malware instances from each family. Since no side information is available, we only use a linear kernel and the identity matrix as components for the correlation matrix. We report on the (normalized) area under the ROC curve over the interval $[0, 0.1]$ and denote this performance measure by $\text{AUC}_{0.1}$. In Fig. 3, right, we show the ROC curves and in Table 1 the achieved $\text{AUC}_{0.1}$.

CPR	CPR-MAP	Uncorr. Probit	GP	LMM-Lasso
74.9 ± 0.2	73.1 ± 0.4	67.2 ± 0.3	69.8 ± 0.3	66.45 ± 0.3

Table 1: $\text{AUC}_{0.1}$ attained on the malware dataset.

We observe that correlated probit regression (CPR) achieves a consistent improvement in terms of $\text{AUC}_{0.1}$ over the competitors (by up to 8.4 percentage points). Furthermore, in Fig. 4, right, we plot the correlation of the top features of correlated and uncorrelated probit regression with population structure. We observe that CPR leads to features, which are much less correlated with the malware

[9] Geinimi, FakeDoc, Kmin, Iconosys, BaseBridge, GinMaster, Opfake, Plankton, FakeInstaller, DroidKungFu.

structure.

4.5 Flowering Time Prediction From Single Nucleotide Polymorphisms

We experiment on genotype and phenotype data consisting of 199 genetically different samples from the model plant *Arabidopsis thaliana* [34]. The genotype of each sample comprises 216,130 single nucleotide polymorphism (SNP) features. The phenotype that we aim to predict is early or late flowering of a plant when grown at ten degrees centigrade. The original dataset contains the flowering time for each of the 199 genotypes. We split the dataset into the lower and upper 45%-quantiles of the flowering time and removed the middle 10%. We then binarized the labels, resulting in a set of 180 instances from which we use $n = 150$ instances for training. The results are reported in Table 2

CPR	CPR-MAP	Uncorr. Probit	GP	LMM-Lasso
84.1 ± 0.2	83.6 ± 0.3	83.5 ± 0.2	83.6 ± 0.2	79.7 ± 0.2

Table 2: FLOWERING time prediction experiment (AUCs).

and show that CPR has a slight advantage of at least 0.5 percentage points in AUC over the competitors.

An analysis restricted to the ten SNPs with largest absolute regression weights in our model showed that they lie within four well-annotated genes that all convincingly can be related to flowering, structure and growth: the gene AT2G21930 is a growth protein that is expressed during flowering, AT4G27360 is involved in microtubule motor activity, AT3G48320 is a membrane protein, involved in plant structure, and AT5G28040 is a DNA binding protein that is expressed during flowering.

5 Related Work

We have already commented on how our model relates to uncorrelated probit regression, GP classification, and linear mixed models. A common generalized linear model for classification is the logistic regression model [35]. Accounting for correlations in the data is non-straightforward [36]; one has to resort to approximate inference techniques, including the Laplace and mean field approximations that have been proposed in the context of GP classification [16], or the pseudo likelihood method, which has been proposed in the context of generalized LMMs [37]. To our knowledge feature selection has not been studied in a correlated logistic setup. On the other hand, without correlations, there is numerous work on feature selection in Lasso regression [15]. Alternative sparse priors to the Lasso have been suggested in [38] for unsupervised learning (again, without compensating for confounders). The joint problem of sparse estimation in a correlated noise setup has been restricted to the linear regression case [39, 2, 18], whereas we are interested in classification. For classification, we remark that the ccSVM [8] deals with confounding in a different way and it does not yield a sparse solution. Finally, our algorithm builds on EP for GP classification [16, 17], but note that GP classification does not yield sparse estimates and, therefore, gives no insights in the underlying structure of the problem.

6 Conclusion

We presented a novel algorithm for sparse feature selection in binary classification where the training data show spurious correlations due to confounding. Our model is inspired by the LMM of linear regression, where confounding is modeled in terms of a correlated Gaussian noise variable. While generalizing the LMM paradigm to binary classification poses technical challenges as exact inference becomes intractable, our solution relies on approximate inference. We demonstrated the use of our approach on two data sets from the field of statistical genetics; a field plagued by spurious correlations of various sorts. We showed that our algorithm finds features which show less spurious correlations and, therefore, lets us finds signals in the data that hopefully have a better causal interpretation.

Our CPR algorithm can be seen as a hybrid between an ℓ_1-norm regularized probit classifier (enforcing sparsity) and a GP classifier that takes as input an arbitrary noise kernel. It distinguishes between sparse linear effects from non-sparse effects due to confounding as modeled in terms of correlated Gaussian noise. We showed that our model selects features that are less correlated with the confounders (defined as the first principal components of the noise covariance) and therefore allows to find sparse effects in the data which has a causal interpretation.

In the future we will further explore data subsampling strategies of our approach and thereby further improve the scalability. Also, we plan to extend the correlated probit model towards a multi-class version. Another important direction is to automatically learn the noise covariance structure from the data, where methods borrowed from Gaussian process classification might help.

Acknowledgements

We thank Manfred Opper, Mehryar Mohri, David Blei, Rajesh Ranganath, Maja Rudolph, and Gunnar Rätsch for stimulating discussions. SM acknowledges the support of the U.S. National Science Foundation I2CAM International Materials Institute Award, Grant DMR-0844115, and the NSF Schloss Dagstuhl support grant for junior researchers (CNS-1257011). SM and MK gratefully acknowledge the support of the NVIDIA Corporation for the donation of the Tesla K40 GPU. MK acknowledges support from the German Research Foundation (DFG) award KL 2698/2-1 and from the Federal Ministry of Science and Education (BMBF) award 031L0023A.

References

[1] T. A. Manolio, F. S. Collins, N. J. Cox, D. B. Goldstein, L. A. Hindorff, D. J. Hunter, *et al.*, "Finding the missing heritability of complex diseases," *Nature*, vol. 461, no. 7265, pp. 747–753, 2009.

[2] S. Vattikuti, J. J. Lee, C. C. Chang, S. D. Hsu, and C. C. Chow, "Applying compressed sensing to genome-wide association studies," *GigaScience*, vol. 3, no. 1, p. 10, 2014.

[3] N. Craddock, M. E. Hurles, N. Cardin, *et al.*, "Genome-wide association study of cnvs in 16,000 cases of eight common diseases and 3,000 shared controls," *Nature*, vol. 464, no. 7289, pp. 713–720, 2010.

[4] T. N. H. G. R. Institute, "Proceedings of the workshop on the dark matter of genomic associations with complex diseases: Explaining the unexplained heritability from genome-wide association studies," 2009.

[5] G. W. Imbens and D. B. Rubin, *Causal Inference in Statistics, Social, and Biomedical Sciences*. Cambridge University Press, 2015.

[6] J. Pearl *et al.*, "Causal inference in statistics: An overview," *Statistics Surveys*, vol. 3, pp. 96–146, 2009.

[7] S. L. Morgan and C. Winship, *Counterfactuals and causal inference*. Cambridge University Press, 2014.

[8] L. Li, B. Rakitsch, and K. M. Borgwardt, "ccsvm: correcting support vector machines for confounding factors in biological data classification," *Bioinformatics*, vol. 27, no. 13, pp. 342–348, 2011.

[9] W. Astle and D. J. Balding, "Population structure and cryptic relatedness in genetic association studies," *Statistical Science*, pp. 451–471, 2009.

[10] C. Lippert, J. Listgarten, Y. Liu, C. Kadie, R. Davidson, and D. Heckerman, "Fast linear mixed models for genome-wide association studies," *Nature Methods*, vol. 8, pp. 833–835, October 2011.

[11] N. Fusi, O. Stegle, and N. D. Lawrence, "Joint modelling of confounding factors and prominent genetic regulators provides increased accuracy in genetical studies," *PLoS comp. bio.*, vol. 8, no. 1, 2012.

[12] P. Kraft, E. Zeggini, and J. P. Ioannidis, "Replication in genome-wide association studies," *Statistical Science: A review journal of the Institute of Mathematical Statistics*, vol. 24, no. 4, p. 561, 2009.

[13] B. J. Vilhjálmsson and M. Nordborg, "The nature of confounding in genome-wide association studies," *Nature Reviews Genetics*, vol. 14, no. 1, pp. 1–2, 2013.

[14] C. I. Bliss, "The method of probits," *Science*, vol. 79, no. 2037, pp. 38–39, 1934.

[15] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[16] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[17] J. P. Cunningham, P. Hennig, and S. Lacoste-Julien, "Gaussian probabilities and expectation propagation," *arXiv preprint arXiv:1111.6832*, 2011.

[18] B. Rakitsch, C. Lipprt, O. Stegle, and K. Borgwardt, "A lasso multi-marker mixed model for association mapping with population structure correction," *Bioinformatics*, vol. 29, no. 2, pp. 206–214, 2013.

[19] O. Weissbrod, C. Lippert, D. Geiger, and D. Heckerman, "Accurate liability estimation improves power in ascertained case-control studies," *Nature methods*, vol. 12, no. 4, pp. 332–334, 2015.

[20] I. Mathieson and G. McVean, "Differential confounding of rare and common variants in spatially structured populations," *Nature genetics*, vol. 44, no. 3, pp. 243–246, 2012.

[21] T. P. Minka, "Expectation propagation for approximate bayesian inference," in *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pp. 362–369, Morgan Kaufmann Publishers Inc., 2001.

[22] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the ADMM," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[23] A. Pakman and L. Paninski, "Exact hamiltonian monte carlo for truncated multivariate gaussians," *Journal of Computational and Graphical Statistics*, vol. 23, no. 2, pp. 518–542, 2014.

[24] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient Langevin dynamics," in *Proceedings of the International Conference on Machine Learning*, 2011.

[25] S. Mandt, M. Hoffman, and D. Blei, "A variational analysis of stochastic gradient algorithms," *International Conference on Machine Learning (ICML)*, 2016.

[26] S. Ahn, A. K. Balan, and M. Welling, "Bayesian posterior sampling via stochastic gradient fisher scoring.," in *ICML*, icml.cc / Omnipress, 2012.

[27] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[28] J. Eckstein and D. P. Bertsekas, "On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators," *Math. Program.*, vol. 55, pp. 293–318, June 1992.

[29] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[30] T. Fawcett, "An introduction to ROC analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[31] A. L. Price, N. J. Patterson, R. M. Plenge, M. E. Weinblatt, N. A. Shadick, and D. Reich, "Principal components analysis corrects for stratification in genome-wide association studies," *Nat Genet*, vol. 38, pp. 904–909, Aug. 2006.

[32] M. P. Berry, C. M. Graham, F. W. McNab, Z. Xu, S. A. Bloch, T. Oni, K. A. Wilkinson, R. Banchereau, J. Skinner, R. J. Wilkinson, *et al.*, "An interferon-inducible neutrophil-driven blood transcriptional signature in human tuberculosis," *Nature*, vol. 466, no. 7309, pp. 973–977, 2010.

[33] D. Arp, M. Spreitzenbarth, M. Hübner, H. Gascon, K. Rieck, and C. Siemens, "Drebin: Effective and explainable detection of android malware in your pocket," in *Proc. of NDSS*, 2014.

[34] S. Atwell, Y. S. Huang, B. J. Vilhjálmsson, G. Willems, M. Horton, Y. Li, D. Meng, A. Platt, A. M. Tarone, T. T. Hu, *et al.*, "Genome-wide association study of 107 phenotypes in arabidopsis thaliana inbred lines," *Nature*, vol. 465, no. 7298, pp. 627–631, 2010.

[35] D. R. Cox, "The regression analysis of binary sequences," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 215–242, 1958.

[36] A. Ragab, "On multivariate logistic distribution," *Micro. Reliab.*, vol. 31, no. 2, pp. 511–519, 1991.

[37] N. E. Breslow and D. G. Clayton, "Approximate inference in generalized linear mixed models," *Journal of the American Statistical Association*, vol. 88, no. 421, pp. 9–25, 1993.

[38] S. Mohamed, K. Heller, and Z. Ghahramani, "Bayesian and l1 approaches to sparse unsupervised learning," *arXiv*

preprint arXiv:1106.1157, 2011.

[39] M. W. Seeger and H. Nickisch, "Large scale bayesian inference and experimental design for sparse linear models," *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 166–199, 2011.