Journal of Machine Learning Research 18 (2017) 1-30 Submitted 4/15; Revised 7/16; Published 11/17

Adaptive Randomized Dimension Reduction on Massive Data

Gregory Darnell GDARNELL@PRINCETON.EDU
Lewis-Sigler Institute
Princeton University
Princeton, NJ 08544, USA

Stoyan Georgiev SGEORG80@GMAIL.COM
Google
Palo Alto, CA 94043, USA

Sayan Mukherjee SAYAN@STAT.DUKE.EDU
Departments of Statistical Science
Mathematics, and Computer Science
Duke University
Durham, NC 27708, USA

Barbara E Engelhardt BEE@PRINCETON.EDU
Department of Computer Science
Center for Statistics and Machine Learning
Princeton University
Princeton, NJ 08540, USA

Editor: Robert McCulloch

Abstract

The scalability of statistical estimators is of increasing importance in modern applications. One approach to implementing scalable algorithms is to compress data into a low dimensional latent space using dimension reduction methods. In this paper, we develop an approach for dimension reduction that exploits the assumption of low rank structure in high dimensional data to gain both computational and statistical advantages. We adapt recent randomized low-rank approximation algorithms to provide an efficient solution to principal component analysis (PCA), and we use this efficient solver to improve estimation in large-scale linear mixed models (LMM) for association mapping in statistical genomics. A key observation in this paper is that randomization serves a dual role, improving both computational and statistical performance by implicitly regularizing the covariance matrix estimate of the random effect in an LMM. These statistical and computational advantages are highlighted in our experiments on simulated data and large-scale genomic studies.

Keywords: dimension reduction, generalized eigendecompositon, low-rank, genomics, linear mixed models, supervised, random projections, randomized algorithms, Krylov subspace methods

1. Introduction

In the current era of information, large amounts of complex high dimensional data are routinely generated across science and engineering disciplines. One perspective is that the signal in high dimensional data is often concentrated in low dimensional structure, and estimating and exploring this latent structure is of fundamental importance in a variety of applications. As the size of the data sets increases, the problem of statistical inference and computational feasibility become inextricably

linked. Dimension reduction is a natural approach to summarizing massive data and has historically played a central role in data analysis, visualization, and predictive modeling. Dimension reduction has had a significant impact on both statistical inference (Adcock, 1878; Edegworth, 1884; Fisher, 1922; Hotelling, 1933; Young, 1941), and on numerical analysis research and applications (Golub, 1969; Golub and Van Loan, 1996; Gu and Eisenstat, 1996; Golub et al., 2000); for a recent review see Mahoney (2011). Historically, statisticians have focused on the study of theoretical properties of estimators often in the context of asymptotically large number of samples. Numerical analysts and computational mathematicians, on the other hand, have been instrumental in the development of useful and tractable algorithms with provable stability and convergence guarantees. Naturally, many of these algorithms have been successfully applied to compute estimators grounded on solid statistical foundations. A classic example of this interplay is principal component analysis (PCA) (Hotelling, 1933). In PCA, an objective function is defined based on statistical considerations about the sample variance, which can then be efficiently computed using a variety of singular value decomposition (SVD) algorithms developed by the numerical analysis community.

In this paper, we consider the problem of dimension reduction, focusing on the integration of i) statistical considerations of *estimation accuracy* and out-of-sample prediction error of matrices with latent low-rank, and ii) computational considerations of run time and *numerical accuracy*. The methodology that we develop builds on a classical approach to modeling large data, which first compresses the data, minimizing the loss of relevant information, and then applies statistical estimators appropriate for small-scale problems. In particular, we focus on dimension reduction via generalized eigendecomposition as the means for data compression, and on out-of-sample residual error as the measure of information loss. The scope of this work includes applications to a large number of dimension reduction methods, which can be implemented as solutions to truncated generalized eigendecomposition problems (Hotelling, 1933; Fisher, 1936; Li, 1991; Wu et al., 2010). In this paper our first focus is on the increasing need to compute an SVD of massive data using randomized algorithms developed in the numerical analysis community (Drineas et al., 2006; Sarlos, 2006; Liberty et al., 2007; Boutsidis et al., 2009; Rokhlin et al., 2009; Halko et al., 2011) to simultaneously reduce the dimension and regularize, or control the impact of independent random noise.

The second focus in this paper is to provide efficient solvers for the linear mixed models that arise in statistical and quantitative genomics. In high-throughput genomics experiments, a vast amount of sequencing data is collected—on the order of tens of millions of genetic variants. The goal of genome-wide association studies (GWAS) is to test for a statistical association at each genetic variant (polymorphic position) to a response of interest (e.g., gene expression levels or disease status) in a sample cohort. However, as the dimension of these genomic data and sample sizes continue to increase, there is an urgent need to improve the statistical and computational performance of standard tests.

It is typical to collect several thousand individuals for one study. These individuals may come from several genetically heterogeneous populations. It has been recognized since 2001 (Pritchard and Donnelly, 2001) that the ancestry makeup of the individuals in the study has great potential to influence study results—in particular, spurious associations arise when genetic variants with differential frequencies may appear to be associated with the biased response variable via latent population structure.

The earliest methods (e.g., genomic control) accounted for population structure by using covariate estimates to correct for these confounding signals. More recently, linear mixed models have been used successfully to correct spurious results in the presence of population structure. LMMs

have been shown to improve power in association studies while reducing false positives (Yang et al., 2014). However, mixed models incur a high computational cost when performing association studies because of the computational burden of computing and inverting the covariance matrix for the random effect controlling for population structure. Significant work has gone into mitigating such costs using spectral decompositions for efficient covariance estimation (Kang et al., 2008, 2010; Yang et al., 2011; Zhou and Stephens, 2012; Listgarten et al., 2012).

In this work we show, using simulations of genomic data with latent population structure and real data from large-scale genomic studies, that our approach, adaptive randomized SVD (ARSVD), is effective in terms of both computational efficiency and numerical accuracy. Under certain settings, we find that the LMM using ARSVD outperforms current state-of-the-art approaches by implicitly performing regularization of the covariance matrix.

There are three key contributions of this paper:

(i) We develop an adaptive algorithm for randomized singular value decomposition (SVD) in which both the number of relevant singular vectors and the number of iterations of the algorithm are inferred from the data based on informative statistical criteria.

(ii) We use our adaptive randomized SVD (ARSVD) algorithm to construct truncated generalized eigendecomposition estimators for PCA and linear mixed models (LMMs) (Listgarten et al., 2012; Zhou and Stephens, 2012).

(iii) We demonstrate on simulated and real data examples that the randomized estimators provide a computationally efficient solution, and, furthermore, often improve statistical accuracy of the predictions. We show that, in an over-parametrized setting, this improvement in accuracy is due to implicit regularization imposed by the randomized approximation.

In Section 2, we describe the adaptive randomized SVD procedure we use for the various dimension reduction methods. In Section 2.5, we provide randomized estimators for linear mixed models used in statistical genetics. In Section 3, we give an explanation for why the randomized estimator for linear (mixed) models imposes regularization. In Section 4, we validate the proposed methodology on simulated and real data and compare our approach with state-of-the-art approaches. In particular, we show results from our approach for estimating low dimensional geographic structure in genomic data and for genetic association mapping applications.

2. Randomized Algorithms for Dimension Reduction

In this section, we develop algorithmic extensions for PCA. We state an algorithm that provides a numerically efficient and statistically robust estimate of the highest variance directions in the data using a randomized algorithm for singular value decomposition (Randomized SVD) (Rokhlin et al., 2009; Halko et al., 2011). In this problem, the objective is linear unsupervised dimension reduction with the low-dimensional subspace estimated via an eigendecomposition. Randomized SVD will serve as the core computational engine for the other estimators we develop in this paper.

2.1 Notation

Given positive integers p and d with $p \gg d$, $\mathbb{R}^{p\times d}$ denotes the class of all matrices of dimension $p \times d$ with real entries. We denote symmetric positive semi-definite matrices as $\mathbb{S}_+^p$. For $B \in \mathbb{R}^{p\times d}$,

span(B) denotes the subspace of $\mathbb{R}^p$ spanned by the columns of B. A *basis matrix* for a subspace $\mathcal{S}$ is any full column rank matrix $B \in \mathbb{R}^{p \times d}$ such that $\mathcal{S} = \text{span}(B)$, where $d = \dim(\mathcal{S})$. We denote the data matrix $X = (x_1, \ldots, x_n)^T \in \mathbb{R}^{n \times p}$ with observations drawn from p-dimensional marginal distribution, $x_i \sim \mathcal{P}_X$. When we consider supervised problems such as regression we denote the response vector as either a quantitative response $Y \in \mathbb{R}^m$ or a categorical response $Y \in \{1, \ldots, C\}$, here C is the number of categories. For the joint setting of response and predictor variables we assume a joint distribution, $(X, Y) \sim \mathcal{P}_{X \times Y}$. We denote the orthonormal left eigenvector basis of the data matrix X as eigen-basis(X).

2.2 Computational Considerations

The main computational tool we use is a randomized algorithm for approximate eigendecompositon, which factorizes a $n \times p$ matrix of rank r in time $\mathcal{O}(npr)$ using randomized methods that take advantage of the intrinsic low-rank of the input matrix, rather than the $\mathcal{O}(np \times \min(n, p))$ time required by deterministic approaches. This is relevant to statistical applications to high dimensional data but reflects a highly constrained process (e.g., from genomic or financial applications), which suggests that the data have low intrinsic dimensionality, i.e., $r \ll n < p$. Further improvements have been made to randomized algorithms for approximate eigendecomposition by noting that a structured random projection (such as the subsampled random Fourier transform) can achieve computational complexity of $O(np\log(r) + (m+p)r^2)$ assuming the input matrix fits in main memory (Halko et al., 2011). Since we use power iterations to decay the eigenspectrum and achieve a numerically accurate result independent of the particular spectral gaps, most of the computational gains from subsampling methods would be lost when applied in our framework. Furthermore, since matrix-matrix multiplies are highly optimized on many computational architectures, parallel implementations can reduce our asymptotic complexity to yield excellent run times in practice (Halko et al., 2011).

An appealing characteristic of our randomized algorithm is the explicit control of the trade-off between estimation accuracy relative to the exact estimates and computational efficiency. Rapid convergence to the exact estimates has been shown both empirically as well as in theory (Rokhlin et al., 2009). From the perspective of theoretical computer science and numerical analysis, the objective of randomized SVD algorithms is, given a matrix X, to efficiently compute an approximate eigendecomposition that is close to the exact eigendecomposition; we call this view the *approximation perspective*.

2.3 Statistical Considerations

A *statistical perspective* will deviate from the approximation perspective in two ways: the data matrix X is not fixed, but a noisy random sample drawn from a population, and the inferential objective is to obtain estimates of population quantities from the sample X, not estimates of the eigendecomposition of X itself. Taking a statistical perspective will drive two central concepts in this paper. The first concept is that there is utility in considering randomized algorithms as statistical models. The second concept is that many formulations of ARSVD implicitly impose regularization constraints.

The acceptable error for the approximation perspective and the statistical perspective differ; typically larger error is tolerated in the statistical perspective. For many statistical estimators, the error between the estimator and the population quantity scales as $\varepsilon = O(\frac{1}{\sqrt{n}})$ where n is the sample

size. This is much coarser than the approximation error sought in numerical analysis, where the desired error between the exact and approximate algorithms scales as ε^2, the squared error in the statistical estimate. This observation highlights that, in the statistical setting, one can use fewer computations than are typically considered in the numerical analysis setting because the accuracy of the finer approximation will be lost to the error due to sampling.

This observation about error will impact the parameters of the randomized estimators that we propose in in this paper. An important parameter in our ARSVD algorithms is the number of power iterations t that the randomized algorithm executes (Section 2.4). Increasing the number of power iterations results in a closer approximation to the exact solution (Rokhlin et al., 2009), but also increases the runtime of the algorithm. The observation that we can afford coarser error rates between the exact and approximate solutions suggests that very few power iterations may be required. We provide empirical evidence (Section 4) that fewer power iterations of the approximate algorithm provide results that are both faster and also more accurate with respect to out-of-sample predictions. This observation suggests that the approximation induced by the randomized algorithm is a form of regularization.

2.4 Adaptive Randomized Low-Rank Approximation

In this section, we provide a brief description of a randomized estimator for the best low-rank matrix approximation, introduced by Rokhlin et al. (2009); Halko et al. (2011), which combines random projections with numerically stable matrix factorization. We consider this numerical framework as implementing a computationally efficient shrinkage estimator of the subspace capturing the largest variance directions in the data. The procedure is well suited for matrices that are low rank or matrices where the signal is low rank. Detailed discussion of the estimation accuracy of Randomized SVD in the absence of noise is provided in Rokhlin et al. (2009).

The idea of random projection was first developed as a proof technique to study the distortion induced by the low dimensional embedding of high-dimensional vectors (Johnson and Lindenstrauss, 1984), with much literature simplifying and sharpening the results (Frankl and Maehara, 1987; Indyk and Motwani, 1998; Achlioptas, 2001; Dasgupta and Gupta, 2003). More recently, the theoretical computer science and the numerical analysis communities discovered that random projections can be used for efficient approximation algorithms for a variety of applications (Drineas et al., 2006; Sarlos, 2006; Liberty et al., 2007; Boutsidis et al., 2009; Rokhlin et al., 2009; Halko et al., 2011). We focus on one such approach proposed by Rokhlin et al. (2009); Halko et al. (2011), which targets the accurate low-rank approximation of a given large data matrix $X \in \mathbb{R}^{n\times p}$. In particular, we extend the randomization methodology to the noisy setting, in which the estimation error is due to both the approximation of the low-rank structure in X and also added noise. A simple working model capturing this scenario is

$$X = X_{d^*} + E, \quad X_{d^*} \in \mathbb{R}^{n\times p}, \quad \text{rank}(X_{d^*}) = d^*,$$

where X_{d^*} captures the low dimensional signal and E is independent additive noise.

2.4.1 ALGORITHM FOR ARSVD

Given an upper bound on the target rank $d_{\max}$ and the number of necessary power iterations $t_{\max}$ ($t_{\max} \in \{5, \dots 10\}$ is sufficient in most cases), the algorithm proceeds in two stages: (1) estimate a

basis for the range of X_{d^*}, (2) project the data onto this basis and apply SVD:

Algorithm: *Adaptive Randomized SVD*(X, $t_{\max}$, $d_{\max}$, Δ)

(1) Find orthonormal basis for the range of X;

(i) Set the number working directions: $\ell = d_{\max} + \Delta$;

(ii) Generate random matrix: $\Omega \in \mathbb{R}^{n \times \ell}$ with $\Omega_{ij} \overset{iid}{\sim} \mathrm{N}(0, 1)$;

(iii) Construct blocks: $F^{(t)} = XX^T F^{(t-1)}$ with $F^{(0)} = \Omega$ for $t \in \{1, \ldots, t_{\max}\}$;

(iv) Select the optimal block $t^* \in \{1, \ldots, t_{\max}\}$ and rank estimate $d^* \in \{1, \ldots, d_{\max}\}$, using the stability criterion and Bi-Cross-Validation stated in Section 2.4.3;

(v) Compute a basis for the selected block: $F^{(t^*)} = QR \in \mathbb{R}^{n \times \ell}, Q^T Q = I$;

(2) Project data onto the range basis and compute the SVD;

(i) Project onto the basis: $B = X^T Q \in \mathbb{R}^{p \times \ell}$;

(ii) Factorize: $B \overset{\text{svd}}{=} U \Sigma W^T$, where $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_\ell)$;

(iii) Compute the rank d^* approximation: $\widehat{X}_{d^*} = U_{d^*} \Sigma_{d^*} V_{d^*}^T$
$U_{d^*} = (U_1 | \ldots | U_{d^*}) \in \mathbb{R}^{n \times d^*}$
$\Sigma_{d^*} = \text{diag}(\sigma_1, \ldots, \sigma_{d^*}) \in \mathbb{R}^{d^* \times d^*}$
$V_{d^*} = Q \times (W_1 | \ldots | W_{d^*}) \in \mathbb{R}^{p \times d^*}$;

In stage (1), we set the number of working directions $\ell = d_{\max} + \Delta$ to be the sum of the upper bound on the rank of the data $d_{\max}$, and a small oversampling parameter Δ, which ensures a more stable approximation of the top $d_{\max}$ sample variance directions; the estimator tends to be robust to changes in Δ, so we use $\Delta = 10$ as a suggested default. In step (1.iii), the random projection matrix Ω is applied to powers of XX^T to randomly sample linear combinations of eigenvectors of the data weighted by powers of the eigenvalues:

$$\underbrace{F^{(t)}}_{n \times \ell} = (XX^T)^t \Omega = US^{2t}U^T\Omega = US^{2t}\Omega^*, \quad \text{where } X \overset{\text{svd}}{=} USV^T.$$

The power iterations shrink small eigenvalues and increase large eigenvalues while leaving the eigenvectors unchanged. Observe that each column of $F^{(t)}$ can be thought of as drawn from a multivariate normal, $F_j^{(t)} \sim \mathcal{N}(0, US^{4t}U^T)$. The covariance structure of this matrix is biased towards higher directions of variation as t increases. The fact that the power iterations shrink noise directions shows that power iterations impose a form of regularization. The multivariate normal structure of this shrinkage is related to local shrinkage priors developed in Polson and Scott (2010). In step (iv), we select an optimal block $F^{(t^*)}$ for $t^* \in \{1, \ldots, t_{\max}\}$ and estimate an orthonormal basis for the column space. For numerical stability, each block in the intermediate power iterations should be orthgonalized (Halko et al., 2011; Gu, 2015). In previous work (Rokhlin et al., 2009), the authors assumed fixed target rank d^* and approximated X rather than X_{d^*}. They showed that the optimal strategy is to set $t^* = t_{\max}$, which typically achieves excellent d^*-rank approximation accuracy for X, even for relatively small values of $t_{\max}$.

In stage (2), we rotate the orthogonal basis Q computed in stage (1) to the canonical eigenvector basis and scale according to the corresponding eigenvalues. In step (2.i) the data is projected onto the low dimensional orthogonal basis Q. Step (2.ii) computes the exact SVD in the projected space.

In this work, we focus on the noisy case, where $E \neq 0$, and propose to adaptively set both d^* and t^*, aiming to optimize the generalization or out-of-sample performance of the randomized estimator. The estimation strategy for d^* and t^* is described in detail in Section 2.4.3.

2.4.2 COMPUTATIONAL COMPLEXITY

The computational complexity of the randomization step is $\mathcal{O}(np \times d_{\max} \times t_{\max})$ and the factorizations in the lower dimensional space have complexity $\mathcal{O}(np \times d_{\max} + n \times d_{\max}^2)$. With $d_{\max}$ small relative to n and p, the runtime in both steps is dominated by the multiplication by the data matrix; in the case of sparse data, fast multiplication can further reduce the run time. We use a normalized version of the above algorithm that has the same run time complexity but is numerically more stable (Martinsson et al., 2010).

2.4.3 ADAPTIVE METHOD TO ESTIMATE d^* AND t^*

We propose to use ideas of stability under random projections in combination with cross-validation to estimate the intrinsic dimensionality of the reduced subspace d^* and the optimal value of the eigenvalue shrinkage parameter t^*.

2.4.4 ESTIMATION OF t^* USING BI-CROSS-VALIDATION

We propose a procedure for selecting an optimal value for $t \in \{1, \ldots, t_{\max}\}$ by using the Bi-Cross-Validation procedure of Owen and Perry (2009), which was used to estimate the rank or cutoff for SVD. For our procedure, we consider a Bi-Cross-Validation formulation that uses the generalized Gabriel holdout pattern (Gabriel, 2002) to partition the data matrix by partitioning the rows and columns into $r = 2$ and $c = 2$ groups respectively that are non-overlapping as suggested in Owen and Perry (2009). We then compute the following Bi-Cross-Validation error by holding out each of the four blocks and estimating a block using the other three blocks

$$\begin{aligned} \text{BiCV}(t) = &\frac{1}{4}\Big[\|A - BD_t^{\dagger}C|\|_F^2 + \|B - AC_t^{\dagger}D\|_F^2 + \|B - AC_t^{\dagger}D\|_F^2 + \\ &\|C - DB_t^{\dagger}A\|_F^2 + \|D - CA_t^{\dagger}B\|_F^2\Big], \quad \text{here } X = \begin{pmatrix} A & B \\ C & D \end{pmatrix}. \end{aligned} \tag{1}$$

In the above equation, $\|\cdot\|_F^2$ is the Frobenius norm, $U_t^{\dagger}$ is the Moore-Penrose pseudoinverse of U where the SVD of U is computed using Adaptive Randomized SVD$(t, d(t), \delta = 10)$, and $d(t)$ is set using the stability criterion developed in the next section. We optimize over the range $\{1, \ldots, t_{\max}\}$ to estimate t^*

$$\hat{t} \;=\; \underset{t \in \{1,\ldots,t_{\max}\}}{\arg\min} \; \text{BiCV}(t).$$

In our simulation results, the value of t^* is estimated small enough to not incur much computational overhead, yet still yields accurate results (Results Section). The original formulation of Bi-Cross-Validation defined the hold-out error to be the Frobenius norm between the predicted and

true submatrix (Owen and Perry, 2009). While the Frobenius and spectral norms both have upper bounds with respect to approximation accuracy of truncated spectral decompositions, more recent results suggest that the spectral norm may generalize better if the goal is to produce an accurate dimension reduction of massive data such as principal components analysis (Mahoney, 2011; Szlam et al., 2014). We recognize the limitation of the Frobenius norm in this context, and acknowledge that it may be wise for Bi-Cross-Validation error to use the spectral norm.

2.4.5 ESTIMATION OF d^* USING STABILITY CRITERION

Given the number of power iterations t, we describe a procedure to estimate the rank parameter $d^*(t)$ using a stability criterion based on random projections of the data. We start with rough upper-bound estimate $d_{\max}$ for the dimension parameter d^*. We then apply a small number ($B = 5$) of independent Gaussian random projections $\Omega^{(b)} \in \mathbb{R}^{n \times d_{\max}}$, $\Omega_{ij}^{(b)} \overset{iid}{\sim} \mathrm{N}(0,1)$, for $b \in \{1, \ldots, B\}$. Given the projections, we compute an estimate of the eigenvector basis of the column space onto the projected data. We then denoise the estimate by raising all the eigenvalues to the power t:

$$U_b^{(t)} \equiv (U_{b1}^{(t)} | \ldots | U_{bd}^{(t)}) = \mathrm{SVD}[(XX^T)^t \Omega^{(b)}] \text{ for } b \in \{1, \ldots, B\}.$$

The k-th principal basis left singular vector estimate ($k \in \{1, \ldots, d\}$) is assigned a *stability score*:

$$\mathrm{stab}(t, k, B) = \frac{1}{N} \sum_{j_1=1}^{B-1} \sum_{j_2=j_1+1}^{B} \left| \mathrm{cor}\left(U_{j_1 k}^{(t)}, U_{j_2 k}^{(t)}\right) \right|, \quad \text{where } N = \frac{B(B-1)}{2}.$$

Here $U_{rk}^{(t)}$ is the estimate of the k^{th} principal eigenvector of $X^T X$ based on the r-th random projection and $\mathrm{cor}\left(U_{j_1 k}^{(t)}, U_{j_2 k}^{(t)}\right)$ denotes the Spearman rank-sum correlation between $U_{j_1 k}^{(t)}$ and $U_{j_2 k}^{(t)}$. Eigenvector directions that are not dominated by independent noise are expected to have higher stability scores. When the data has approximately low-rank, we expect a sharp transition in the eigenvector stability between the directions corresponding to signal and to noise. In order to estimate this change point, we apply a non-parametric location shift test (Wilcoxon rank-sum) to each of the $d_{\max} - 2$ stability score partitions of eigenvectors with larger versus smaller eigenvalues. The subset of principal eigenvectors that can be stably estimated from the data for the given value of t is determined by the change point with smallest p-value among all $d_{\max} - 2$ non-parametric tests.

$$\hat{d}_t = \underset{k \in \{2, \ldots, d_{\max}-1\}}{\arg\min} \text{ p-value}(k, t),$$

where p-value(k,t) is the p-value from the Wilcoxon rank-sum test applied to the $\{\mathrm{stab}(t, i, B)\}_{i=1}^{k-1}$ and $\{\mathrm{stab}(t, i, B)\}_{i=k}^{d_{\max}}$.

2.5 Fast Linear Mixed Models

Multivariate linear mixed models (LMMs) are a workhorse in statistical and quantitative genetics because they allow for the regression of explanatory variables on outcome variables while capturing potentially confounding relatedness between samples (Henderson, 1984; Price et al., 2011; Krote et al., 2012). In the context of the genetic association mapping of complex traits, LMMs are used to control for observed (known covariates) and unobserved (random effects) statistical confounding, particularly the presence of population structure amongst samples.

The linear mixed models in this paper take the form

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + e,$$

where $\mathbf{y}$ is an $n \times 1$ response vector of observed phenotypes, $\mathbf{X}$ is an $n \times q$ matrix of fixed effects that includes the genotypes (SNPs) and other confounding variables, $\boldsymbol{\beta}$ is a $q \times 1$ vector representing coefficients of the fixed effects, $\mathbf{u} \sim N(0, \sigma_g^2 K)$ is the random vector of additive genetic effects with incidence matrix $\mathbf{Z}$, and the vector $e \sim \mathcal{N}(0, \sigma_e^2 I_n)$ is the residual error. The matrix K is the kinship or genetic relatedness matrix and may be computed from genotype data. Parameter σ_g^2 is the proportion of variance in the phenotypes explained by genetic factors. The overall phenotypic variance-covariance matrix, integrating out the random effects, is $\mathbf{V} = \sigma_g^2 \mathbf{Z} K \mathbf{Z}^T + \sigma_e^2 I_n$.

The main goal in genetic association studies is to test every genetic locus for significant association to the phenotype based on the effect size of the coefficient $\boldsymbol{\beta}_j$. For each genetic locus, we apply the following hypothesis test:

$$\text{null } H_0 : \boldsymbol{\beta}_j = 0 \quad \text{alternative } H_1 : \boldsymbol{\beta}_j \neq 0, \qquad \text{for } j = 1, \ldots, p.$$

The standard procedure for inferring SNPs associated with a phenotype while correcting for population structure using an LMM proceeds in the following steps:

(1) *Construct the genetic relatedness matrix (GRM)*: The GRM captures the genetic relationship between individuals in the study to model population structure, family structure, and cryptic relatedness. There are cases where the GRM may be obtained directly if the pedigree of the individuals in the sample is known (see Thompson, 1976). In the more common setting, we are not given pedigree information, but instead we have the p dimensional genotype vector G for each of the n individuals. Given the genotype matrix, we can compute the GRM matrix as GRM $= GG^T$. In the machine learning literature, the GRM would be defined as the Gram matrix for a linear kernel. The biological interpretation of a GRM computed from genotype data differs slightly from the one specified by a pedigree.

(2) *Estimate variance components*: We first use the restricted maximum likelihood estimation (REML) method to estimate σ_g^2, the proportion of phenotypic variance attributable to additive genetic effects. Given the estimate $\hat{\sigma}_g^2$, we estimate σ_ϵ^2, the proportion of phenotypic variance attributable to environmental factors. There are several efficient algorithms for computing the REML (Johnson and Thompson, 1995; Gilmour et al., 1995; Lin et al., 2013; Matilainen et al., 2013). If one computes the random effects first and then the fixed effects, unbiased estimates of the random effects can be found; this is typically the order of computations in genomics.

(3) *Compute an association statistic at each genotype location*: There are a variety of procedures to test for the significance of a genotype j using coefficients $\widehat{\boldsymbol{\beta}}_j$. One approach is to use an F-statistic to test for whether $(G\boldsymbol{\beta})_j = 0$ for each $j = 1, \ldots, p$ (Kang et al., 2008; Kennedy et al., 1992; Henderson, 1984). Another approach is to use a likelihood ratio test considering the variance components. Denote $\ell_1(\hat{\sigma}_1$ as the likelihood under the alternate hypothesis with $\hat{\sigma}_1 = \hat{\sigma}_g^2$ as the estimate of the additive genetic variance component under the alternate model. Then denote $\ell_0(\hat{\sigma}_0)$ as the likelihood under the null ($\boldsymbol{\beta} = 0$) with $\hat{\sigma}_0 = \hat{\sigma}_g^2$ the estimate of the additive genetic variance component under the null model. The log ratio test statistic $2 \log \frac{\ell_1(\hat{\sigma}_1)}{\ell_0(\hat{\sigma}_0)}$ follows a χ^2 distribution and may be used as a test statistic (Kang et al., 2008, 2010; Zhou and Stephens, 2012).

Reducing the computational complexity of the LMM has been an active area of research driven by the increasing size of association studies. A variety of methods have been proposed to increase computational speed (see Kang et al., 2008, 2010; Lippert et al., 2011; Zhou and Stephens, 2012; Lippert et al., 2013). The software we implemented for the results in this paper as well as our methodology is based on EMMAX (Kang et al., 2010). EMMAX improved on EMMA (Kang et al., 2008), which dramatically reduced the computational cost of a standard LMM by exploiting properties of a spectral decomposition of the genotype matrix. EMMAX improves on EMMA by approximating the variance component for each SNP based on an estimate that is computed only once rather than for each SNP. Although we will speed up the model used in EMMAX, our approach can be applied to other fast LMM solvers.

Our contribution to accelerating parameter estimation is using ARSVD to reduce the computational complexity of estimating the random effect $\mathbf{u}$ associated with the design matrix $\mathbf{Z}$. An SVD of the matrix $\mathbf{Z}$ has complexity $\mathcal{O}(n^3)$. If we instead apply ARSVD to $\mathbf{Z}$, we reduce the computational complexity to $\mathcal{O}(np \times d_{\max} + n \times d_{\max}^2)$. In addition, using ARSVD to decompose the design matrix serves to denoise the GRM by retaining the low-rank structure present in the GRM. This avoids the need to manually or heuristically subset the data to achieve a low-rank representation. We will observe in both simulated and real data examples that this application of the ARSVD leads to both a substantial acceleration of the computational speed and an implicit regularization of the design matrix, which reduces type I errors substantially.

3. Regularization of ARSVD

The idea of adding randomness or noise to algorithms for the purpose of regularization has been repeatedly rediscovered (Bishop, 1995; Simard et al., 1993; Mahoney, 2011; Srivastava et al., 2014). Adding independent and identical noise to input variables was observed and rigorously shown to be identical to Tikhonov regularization (Bishop, 1995). Furthermore, Tikhonov regularization is closely related to early stopping, as both regularization methods act as low pass filters (Yao et al., 2007). On the other hand, early stopping is not subject to saturation (Vito et al., 2005; Smale and Zhou, 2007; Yao et al., 2007). In this section we explain why principal components regression (PCR) using ARSVD is a form of regularization. We are confident more refined analyses as well as sharper statements and bounds can be made; our results are more motivational than a detailed analysis.

We will use the spectral filtering framework. In particular, we will use kernel least squares ridge regression (KRR) to illustrate spectral filtering. We then show that PCR using ARSVD is also a spectral filter, and regularization is imposed by weighting and truncating eigenvalues of a positive semidefinite matrix that the algorithm constructs.

3.1 Kernel Ridge Regression

We consider the regression setting, where the number of variables is much larger than the number of observations, $p \gg n$. Given an $n \times p$ design matrix X, the ordinary least squares solution (OLS) is computed based on the normal equations:

$$\hat{\beta} = (X^T X)^{-1} X^T Y,$$

where Y is an $n \times 1$ vector of responses and $\hat{\beta}$ is the OLS estimate for the regression coefficients. When $p \gg n$, the OLS estimator does not work because $X^T X$ is not invertible. In this high-dimensional $p \gg n$ setting, ridge regression (Hoerl and Kennard, 1970) addresses many of the shortcomings of OLS. The estimation problem in ridge regression is formulated as

$$\hat{\alpha} = (XX^T + n\lambda I)^{-1} Y,$$

where λ is a regularization parameter and the induced regression function is

$$\hat{y} = \sum_{j=1}^{p} \hat{\alpha}_j x_j^T x = \hat{\beta}^T x, \quad \hat{\beta} = \sum_{j=1}^{p} \hat{\alpha}_j x_j.$$

When $\lambda = 0$, we recover the OLS estimator, and, when $\lambda = \infty$, one obtains the zero solution $\hat{\alpha} = 0$; λ trades off between fitting the observations and shrinking the solution towards zero.

A standard nonlinear extension to ridge regression is kernel ridge regression (Poggio and Girosi, 1990; Williams and Seeger, 2001) where the regression function takes the form

$$f(x) = \sum_{i=1}^{n} \alpha_i k(x, x_i),$$

and $k(u, v)$ is a positive (semi) definite function called a *kernel*. One example of a kernel function is the Gaussian kernel, $k(u, v) = \exp(-h^2 \|u - v\|^2)$. The parameters α to be estimated in kernel ridge regression (KRR) are given by the formula

$$\hat{\alpha} = (K + n\lambda I)^{-1} Y,$$

where the kernel matrix K is defined as $K_{ij} = k(x_i, x_j)$.

3.2 KRR as a Spectral Filter

For the purposes of this paper we will consider spectral filtering as a procedure to filter or smooth a signal (vector) by filtering the eigenvalues of a positive (semi) definite matrix. A signal processing perspective of KRR as a filtering operation is as follows: given the response signal Y and matrix K, the filtering procedure is a map $F(K) : \mathbb{R}^n \to \mathbb{R}^n$, where

$$\hat{Y} = FY, \quad F = K(K + n\lambda I)^{-1}. \tag{2}$$

The basic idea behind spectral filtering is that the filter F operates on the spectrum of the positive (semi) definite matrix, K. In the KRR setting, a natural basis for the matrix F is the eigenvectors of K, and we define the orthonormal matrix $V = [v_1 \cdots v_m]$ with $(v_j)_{j=1}^m$ the m eigenvectors of K with nonzero eigenvalues. The filter F can be written as

$$F = (v_1\ v_2\ \cdots\ v_m) \begin{pmatrix} f(\sigma_1) & & & \\ & f(\sigma_2) & 0 & \\ & & \ddots & \\ & 0 & & f(\sigma_m) \end{pmatrix} \begin{pmatrix} v_1^T \\ v_2^T \\ \vdots \\ v_m^T \end{pmatrix}, \tag{3}$$

where the spectrum of K is filtered by the function $f(\sigma_i) = \frac{\sigma_i}{\sigma_i+\lambda}$. The filter given in equation (3) can be thought of as a low pass filter that is smoothing the signal Y by shrinking higher frequency eigenvectors—those eigenvectors corresponding to small eigenvalues. In the case of the linear kernel, which is the focus of our paper, K is the Gram matrix, $K_{ij} = x_i^T x_j$.

3.3 Randomized Principal Component Regression as a Spectral Filter

The standard formulation of principal components regression (PCR) is specified by the model

$$y_i = \beta^T z_i + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} \mathrm{N}(0, \sigma^2), \tag{4}$$

where $z_i = x_i^T V$ with x_i an observation in $\mathbb{R}^p$ and $V = [v_1 \cdots v_m]$ are the m eigenvectors corresponding to the top eigenvalues of the empirical covariance matrix. The idea of PCR is that projection onto the top principal component reduces variance without much loss in bias. The nonzero eigenvalues of the Gram matrix and the empirical covariance matrix are identical, so one can compute V from the Gram matrix $K_{ij} = x_i^T x_j$.

Again, we consider the case where $p \gg n$ and assume that the ARSVD procedure sets the maximum rank $d_{\max} \ll n \ll p$. In the following, we will formulate principal components regression as a spectral filter analogous to a filtering formulation of KRR in (2). We will make some approximations in this analysis as the form of the filter in the case of ARSVD is not straightforward due to randomization. The randomization and power iterations of ARSVD impact the spectral filter F in two ways: The filter no longer operates on the eigenvalues of the Gram matrix, and the eigenvectors of the spectral filter are not given by the eigenvalues of the Gram matrix.

Our analysis will consist of two observations. The first is that the eigenvectors of the exact gram matrix K and the eigenvectors of the Gram matrix induced by the ARSVD procedure are close approximations. This observation will allow us to use the eigenvectors of the exact Gram matrix in our analysis of the spectral filter. The second observation is based on a series of papers (Gerfo et al., 2008; Rudi et al., 2013, 2015) that illustrated a common regularization framework for some families of truncation-based algorithms including truncated SVD and PCR, early stopping of iterative procedures, and regularization algorithms such as ridge regression. The filter function for PCR is

$$f(\sigma) = \begin{cases} 1 & \sigma \geq \tau \\ 0 & \text{otherwise,} \end{cases}$$

here τ is the eigenvalue cutoff.

In the case of ARSVD, we generate a random matrix Ω with $\Omega_{ij} \overset{iid}{\sim} \mathrm{N}(0, 1)$. The following power iterations are then taken of a random projection onto the Gram matrix

$$G^{(t)} = (XX^T)^t \Omega = U\Sigma^{2t}U^T\Omega = U\Sigma^{2t}\Omega^*, \quad \text{with } X = U\Sigma V^T,$$

where U, Σ, and V correspond to a standard SVD of X. A basis is computed from matrix $G^{(t)}$ via QR decomposition,

$$G^{(t)} = QR \in \mathbb{R}^{n\times\ell}, \quad Q^TQ = I.$$

The data are then projected onto this basis $B = X^TQ \in \mathbb{R}^{p\times\ell}$ and a standard SVD is run on the much lower rank matrix B. If the eigenvectors of XX^T and BB^T are equal, then the following

spectral filter can be specified for the PCR with ARSVD based on the eigenvalues of the Gram matrix of the data

$$\hat{Y} = FY, \quad F = V\,\Lambda_{f(\sigma)}\,V^T, \quad \Lambda_{f(\sigma)} = \text{diag}(f(\sigma_1), ..., f(\sigma_n)), \quad f(\sigma_i) = \frac{\sigma_i^{2t}}{\sigma_i^{2t} + \tau}, \quad (5)$$

where τ is a threshold parameter. The derivation of this spectral filter is based on results in Gerfo et al. (2008) and Rudi et al. (2013). In the limit of infinite power iterations, the spectral filter is simply a hard thresholding algorithm

$$\lim_{t\to\infty} f(\sigma_i) = \begin{cases} 1 & \sigma_i > 1 \\ 0 & \sigma_i > 1. \end{cases}$$

This asymptotic analysis suggests that scaling each eigenvalue $\sigma_i := \sigma_i/\tau$ can be used to threshold at the level $1/\tau$.

We now show that the eigenvectors for XX^T and BB^T are equivalent, at least for eigenvectors corresponding to larger eigenvalues. This allows us to interpret the filter in terms of the eigenvalues of the Gram matrix XX^T. If the matrix Ω is orthogonal, then the eigenvectors corresponding to the top ℓ eigenvalues of XX^T and BB^T would be equivalent, modulo a constant scale term, which we can set without loss of generality. We argue that the matrix Ω is ε-quasiorthogonal. A set of unit norm vectors $\mu_1, ..., \mu_M \in \mathcal{R}^n$ is ε-quasiorthogonal (Kainen and Kůrková, 1993; Hecht-Nielsen and Kůrková, 1992) if their inner products are small $|\mu_i \cdot \mu_j| \leq \varepsilon$. In Indyk and Motwani (1998, Appendix A), it was shown that, for a random matrix with elements drawn exactly as Ω, the columns are ε-quasiorthogonal.

Beyond principal component regression, the top eigenvalues are important to estimate for many machine learning methods, including graph Laplacian objectives. Graph Laplacians form a key component of practically important algorithms including computing the heat kernel of a graph and PageRank (Mahoney and Orecchia, 2010). It has been shown that approximation algorithms such as ARSVD solve an exact optimization problem with an explicit regularization term (Mahoney and Orecchia, 2010; Perry and Mahoney, 2011).

3.4 Randomization and Leverage Scores

The idea of subsampling observations to generate the Gram matrix is at the heart of Nyström methods (Williams and Seeger, 2001; Drineas and Mahoney, 2005). For runtime considerations, a Gram matrix is constructed as the approximation $\tilde{G} = C^T W^{-1} C$, where C is a $p \times c$ matrix where c is a uniform subsample of the n observations and W is an $n \times n$ incidence matrix of which rows and columns are included in the subsample. There is a great deal of work in the machine learning literature arguing why it is that the Nyström method results in faster algorithms, and recent work illustrating why this numerical approximation can be formulated as a regularization method (Rudi et al., 2013, 2015). Another subsampling perspective is based on leverage scores. Given a design matrix X and the corresponding left singular vectors U the leverage score of a sample is $\ell_j = ||U_j||^2$ (Gittens and Mahoney, 2013) and can be thought of as a measure of relevance of the jth sample to the linear regression function. An alternative to uniformly sampling observations is to sample them according to the leverage score, $p_j \propto \ell_j = ||U_j||^2$, so points with higher leverage scores will more likely be sampled. Indeed, it has been shown that leverage scores drive the accuracy of the the Nyström method, and the uniform sampling approach is optimal when the leverage

scores are almost equal; in the case when the leverage scores are variable, importance sampling according to the leverage score has been used (Drineas et al., 2012). It has been shown previously that random projections project into a space where leverage scores are nearly uniform (Drineas et al., 2012; Mahoney, 2011).

4. Results on Real and Simulated Data

We use real and simulated data to highlight the following three major contributions of this paper

1. In the presence of informative *low-rank* structure in the data, randomized algorithms tend to be much faster than exact methods with minimal loss in approximation accuracy.

2. The *rank* and *subspace* containing information in the data can be reliably estimated and used to provide efficient solutions for dimension reduction.

3. The randomized algorithms implicitly impose regularization, which can be adaptively controlled in a computationally efficient manner to produce improved out-of-sample performance.

4.1 Simulated Data

4.1.1 UNSUPERVISED DIMENSION REDUCTION

We begin with unsupervised dimension reduction of data with low-rank structure contaminated with Gaussian noise, and we focus on evaluating the application of *Adaptive Randomized SVD* for PCA (see Section 2.4). In particular, we demonstrate that the proposed method estimates the sample singular values with exponentially decreasing relative error in t. Then we show that achieving similar low-rank approximation accuracy to a state-of-the-art Lanczos method requires the same run time complexity, which scales linearly in both dimensions of the input matrix. This makes our proposed method applicable to large data matrices. Lastly, we demonstrate the ability to adaptively estimate the underlying rank of the data, given a coarse upper bound. In all our simulations, we set the oversampling parameter in ARSVD, $\Delta = 10$.

We note that the oversampling parameter is crucial in scientific computing applications such as ours. In particular, setting this parameter will be application-specific and depend heavily on the structure of the input data, such as sparsity and distributional properties. For worst-case matrices, our algorithm has potential to produce sub-optimal results (Mahoney, 2011). In general, high quality results—both empirical and theoretical—are achieved by setting the oversampling parameter between five and ten (Halko et al., 2011).

4.1.2 SIMULATION MODEL

We first state the simulation model used for most of the results in this subsection. The data matrix $X \in \mathbb{R}^{n \times p}$, is generated as follows: $X = USV^T + E$, where $U^TU = V^TV = I_{d^*}$. The d^* columns of U and V are drawn uniformly at random from the corresponding unit sphere and the singular values $\mathrm{S} = \mathrm{diag}(s_1, \ldots, s_{d^*})$ are randomly generated starting from a baseline value, which is a fraction of the maximum noise singular value, with exponential increments separating consecutive

entries:

$$\begin{aligned} s_j &= s_{j-1} + \nu_j, \text{ for } j \in \{2, \ldots, d^*\} \\ \nu_j &\overset{\text{iid}}{\sim} \text{Exp}(\lambda), \quad \nu_0 = s_1^{(E)}. \end{aligned}$$

The noise is iid Gaussian: $E_{ij} \overset{\text{iid}}{\sim} N\left(0, \frac{1}{n}\right)$. The gaps between singular values, ν_j, follow an exponential distribution with rate parameter λ to control the signal-to-noise ratio (Table 1). The sample variance has the SVD decomposition $E \overset{\text{svd}}{=} U_E S_E V_E^T$, where $S_E = \text{diag}\left(s_1^{(E)}, \ldots, s_{\min(n,p)}^{(E)}\right)$ are the singular values in decreasing order. While there exist other working models for the noise structure, here we chose to investigate the current model and that of latent population structure because of its relevance to the genetic data that we wish to model (Section 4.1.4). Our simulations and genetic data experiments show that the assumptions we make on this particular noise model generalize well to genetics data.

4.1.3 RESULTS

The first objective is to show that we can accurately estimate singular values with very few power iterations. Our focus is on understanding the effect of the regularization parameter t controlling the singular value shrinkage. Larger values correspond to a stronger weighting on directions with large eigenvalues. In our first simulation we assume the rank d^* is fixed to 50 and the input matrix is $2,000 \times 5,000$. Studying the estimates of the percent relative error of the *singular values* averaged over ten simulated data sets. The relative error given a singular value estimate $\hat{\sigma}$ and singular value σ is $\left(\frac{\sigma - \hat{\sigma}}{\sigma}\right)$. We observed exponential convergence to the sample estimates with increasing t (Table 1). This suggests that we can capture the variation in the data with a few data matrix multiplications. We measure the error in our estimates using a signal-to-noise (S/N) metric

$$\frac{\|S\|_F^2}{\|E\|_F^2}, \quad S \text{ is the signal matrix and } E \text{ is the error or residual matrix.}$$

The signal matrix has a maximum of d^* non-zero singular values, and thus the calculation of the Frobenius norm only includes the top d^* singular values in computing both $\|S\|_F^2$ and $\|E\|_F^2$.

λ	S/N	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$t = 5$
2	2.39	2.34 ± 1.23	1.18 ± 0.63	0.72 ± 0.43	0.48 ± 0.32	0.35 ± 0.25
4	0.61	3.32 ± 1.32	1.67 ± 0.47	1.00 ± 0.28	0.68 ± 0.20	0.50 ± 0.16
6	0.15	5.04 ± 1.53	2.97 ± 0.66	1.86 ± 0.41	1.30 ± 0.30	0.97 ± 0.24
8	0.13	6.26 ± 1.87	3.48 ± 0.42	2.14 ± 0.28	1.47 ± 0.21	1.08 ± 0.18

Table 1: **Singular values from ARSVD.** We report the relative error for singular value estimates with ± 1 standard deviation. A linear increase in the regularization parameter t results in a exponential decrease in the error. S/N is the signal to noise ratio and a function of λ.

It is also of interest to characterize the decay in accuracy in estimating singular values using the randomized method as the magnitude of the true singular values decreases. When we consider

the rank-ordered singular values for a fixed matrix as well as the distribution of the singular values computed via various runs of the ARSVD, an interesting observation is that the estimates are biased for small singular values (Figure 1). Data were generated from $n = 1,000$ and $p = 1,000$ with true rank $d^* = 50$. For the smallest singular values, ARSVD tends to underestimate the singular values, and the standard error is larger. This bias can be considered a form of regularization that shrinks directions corresponding to small singular values. We will discuss this property further in Section 4.1.4.

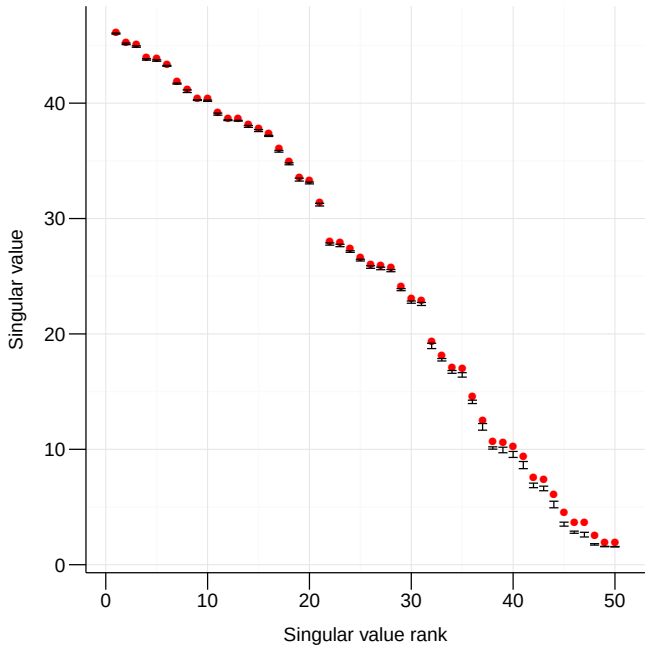

Figure 1: **Singular value accuracy of ARSVD.** Simulation results comparing estimation accuracy of singular values for ARSVD versus SVD on pseudo-random matrices of dimension $n = 1,000$ and $p = 1,000$. The exact singular values are in red and confidence intervals for the singular values computed using ARSVD are in blue.

We can compare the runtime of randomized SVD (RSVD) to two standard spectral decompositions methods. We denote the singular value decomposition of a data matrix X as SVD. We denote as eig the procedure first computing $\widehat{\Sigma} = XX^T$ and then computing the spectral decomposition of $\widehat{\Sigma}$. For this simulation, we generate pseudo-random matrices that are $n \times p$ such that $n = \frac{p}{10}$. For the RSVD procedure we will set the rank parameter to 100. We ran the SVD procedure on matrices with $p = [2000, 40000]$, we did not exceed 40,000 due to computational constraints. We ran the eigenvalue procedure on matrices with $p = [2000, 80000]$. We ran our RSVD procedure on matrices with $p = [2000, 100000]$. Examining the runtime of the three methods in terms of CPU-seconds compared to the size of the matrix, which we index as p, we see on a log scale that RSVD dramatically outperforms the other two methods (Figure 2).

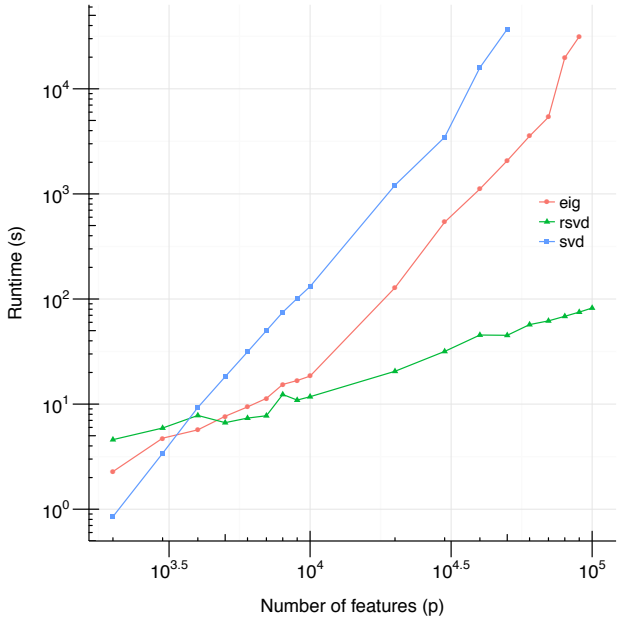

Figure 2: **Comparison of three spectral methods.** We compare RSVD, SVD, and eig. The x-axis indexes matrix size as p and the y-axis is the runtime in seconds. Both axes are on a log scale.

A natural question is whether the randomization offers any advantage over well developed efficient methods, such as Lanczos-Krylov Subspace estimation, which also operate on the data matrix only through matrix multiplies (Saad, 1992; Lehoucq et al., 1998; Stewart, 2001; Baglama and Reichel, 2006). These subspace methods are also iterative in nature, with the runtime complexity typically scaling as $O(qnp)$, where q is small. We compare the runtime ratio of our RSVD with a state-of-the-art low-rank approximation algorithm, the blocked Lanczos method implemented in the CRAN package `irlba` (Baglama and Reichel, 2006). Data were generated from $n = 1000$ and $p = 1000$ with true rank $d^* = 50$, and we varied t in ARSVD. We ran both ARSVD and blocked Lanczos until the Frobenius norm reconstruction error to the original matrix was equal to one degree of precision. We report the ratio of the runtime of ARSVD over blocked Lanczos computed on ten simulated data sets (Table 2). The relative runtime remains approximately constant with simultaneous increase in both data dimensions, which suggests similar order of complexity for both methods when the latent rank of the data (d^*) is supplied as a static parameter to each method. The relative runtime decreases exponentially when using ARSVD to dynamically estimate the latent rank versus using Bi-Cross-Validation to dynamically estimate the rank supplied to block Lanczos.

We now examine our ability to accurately estimate the rank of the matrix using the adaptive stability-based approach outlined in Section 2.4.3. We generated 50 random data sets with $n = 1000, p = 1000, d^* \overset{\text{iid}}{\sim} \text{Uniform}[10, 50]$. We set the initial rank upper bound estimate to be $2 \times d^*$ and used the stability based method (Section 2.4.3) to estimate both optimal t^* and the corresponding d^*. We compared the *true rank* and the corresponding estimates of the regularization parameter t^* for

n + p	6,000	7,500	9,000	10,500	12,000
relative time (static rank)	2.5 ± 0.05	1.84 ± 0.03	1.82 ± 0.03	1.83 ± 0.02	1.84 ± 0.04
n + p	2,000	3,500	5,000	6,500	8,000
relative time (dynamic rank)	3.49 ± 1.10	0.58 ± 0.26	0.46 ± 0.21	0.04 ± 0.01	0.25 ± 0.11

Table 2: **Runtime ratio of ARSVD versus block Lanczos.** We report the sample mean and standard error of ARSVD over block Lanczos based on ten random replicate data sets across dimensionality $n+p$. n is incremented by 500 and p is incremented by $1,000$. In the static rank experiments, the latent rank d^* is supplied as a static parameter to both methods. In the dynamic rank experiments, ARSVD estimates the latent rank using the algorithm previously described, and we use Bi-Cross-Validation to dynamically estimate the latent rank supplied to block Lanczos. ARSVD is run for as many iterations as needed until the sample error is equal to within one degree of precision, thus we do not dynamically estimate the number of power iterations, t^*, in these experiments.

two different signal-to-noise scenarios (Figure 3). In both scenarios, the rank estimates agreed with the true rank values. If the signal-to-noise ratio is low, then our procedure slightly underestimates the rank. We suspect this underestimate is due to the fact that the few smallest variance signal directions tend to be difficult to distinguish from the random noise and hence are less stable under random projections. Our approach tends to select small values for t^*, especially when there is a clear separation between the signal and the noise.

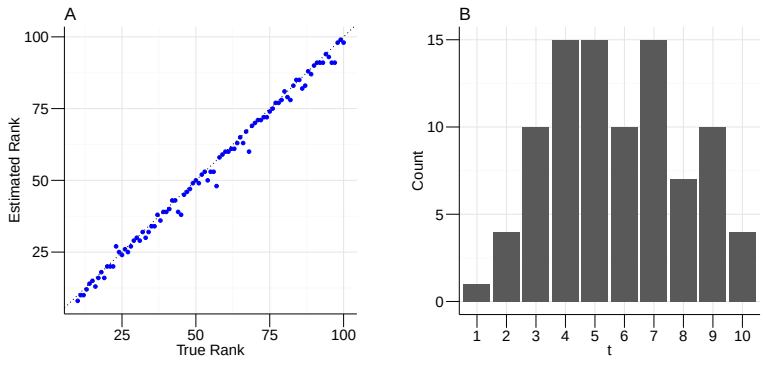

Figure 3: **Rank estimation of ARSVD.** Rank estimation results after performing ARSVD on 50 pseudo-random matrices of dimension $n = 10000$ and $p = 10000$, with true rank $d^* \overset{\text{iid}}{\sim} \text{Uniform}[10, 50]$. Matrices are generated as described in Section 4.1.1. Panel A: True rank (d^*) on the x-axis, and estimated rank ($\hat{d}^*$) on the y-axis. Panel B: estimations of t^* with max t set to ten.

4.1.4 LATENT POPULATION STRUCTURE

We examine how accurately ARSVD can be used to correct latent or cryptic population structure. Specifically, we compared the performance of an LMM using ARSVD versus a standard LMM. We simulated genotype and phenotype data where the genotypes have latent population structure that, once corrected for, there remains no association between genotype and phenotype. In other words, the phenotype is conditionally independent of the genotype given the latent (population) structure. This relation is sometimes called the confounding effect of cryptic structure in genomic data, and motivates the need for LMMs in genome-wide studies. Given that simulations are entirely under the null hypothesis of no association between genotype and phenotype, p-values should follow a uniform distribution if the random effect controls population structure appropriately. We consider any result exceeding an α-threshold a false positive, occurring at rate α.

We use the model stated in Mimno et al. (2014) to simulate admixed genotypes with K ancestral populations. The genotype of an individual is generated by the following hierarchical model

$$\begin{aligned}
\theta_i &\sim \text{Dir}_K(\alpha), \\
\phi_k &\sim \text{Beta}(1,1),\ k = 1, ..., K, \\
(z_{1,ij}, z_{2,ij}) &\sim \Big(\text{Mult}(\theta_i), \text{Mult}(\theta_i)\Big) \text{ for } j = 1, ..., p \\
(x_{1,ij}, x_{2,ij}) &\sim \Big(\text{Bin}(\phi_{z_{1,ij}}), \text{Bin}(\phi_{z_{2,ij}})\Big) \text{ for } j = 1, ..., p.
\end{aligned}$$

The first step samples the the admixture proportions for individual i. The second step samples the allele frequency distribution for populations $k = 1, \ldots, K$. The third step samples the population of origin for both allele copies over all loci, $j = 1, \ldots, p$. The final step samples both copies of the alleles at each locus j. We generated the phenotype using the following relation: $y_i \sim \text{Be}(0.5\theta_k + 0.1(1 - \theta_k))$.

We looked at four simulation settings $n = p = 1000$, $n = p = 5000$, $n = 1000, p = 5000$, and $n = 5000, p = 1000$. Using the p-values for a LMM using ARSVD with the rank parameter d^* of the ARSVD specified, we see that the standard LMM is recovered in the limit of $d^* = p$ (Figure 4). Of these settings, the case where $n = 1000, p = 5000$ is the most similar to the standard genomics case where the number of SNPs p is much larger than the number of observations n. A summary of these simulation results is that the ARSVD method is much faster than the standard LMM and performs similarly with respect to correcting for population structure and controlling false positives. We observed that, for the simulation with $n = 1000, p = 5000$, using ARSVD leads to a substantial reduction of computational complexity with similar performance. In general, we expect the tradeoff between computational efficiency and accuracy to depend on the data. In the case of structured genomic data, we report in these simulations and in Section 4.1.4 that massive computational savings are accompanied by numerical accuracy.

4.2 Association Mapping in Large Genomic Data

We applied our LMM with ARSVD to a large genomic data set to illustrate that we can achieve considerable computational efficiency without loss in accuracy. In particular, we applied our method to the Wellcome Trust Case Control Consortium (WTCCC) data (Consortium, 2007). The data we consider consist of a case-control study of $4,684$ individuals. The cases are individuals with Crohn's disease, and the number of features are 478,765 genetic variants across the 22 autosomal chromosomes in the genome. We compared our ARSVD method with state-of-the-art LMMs designed for

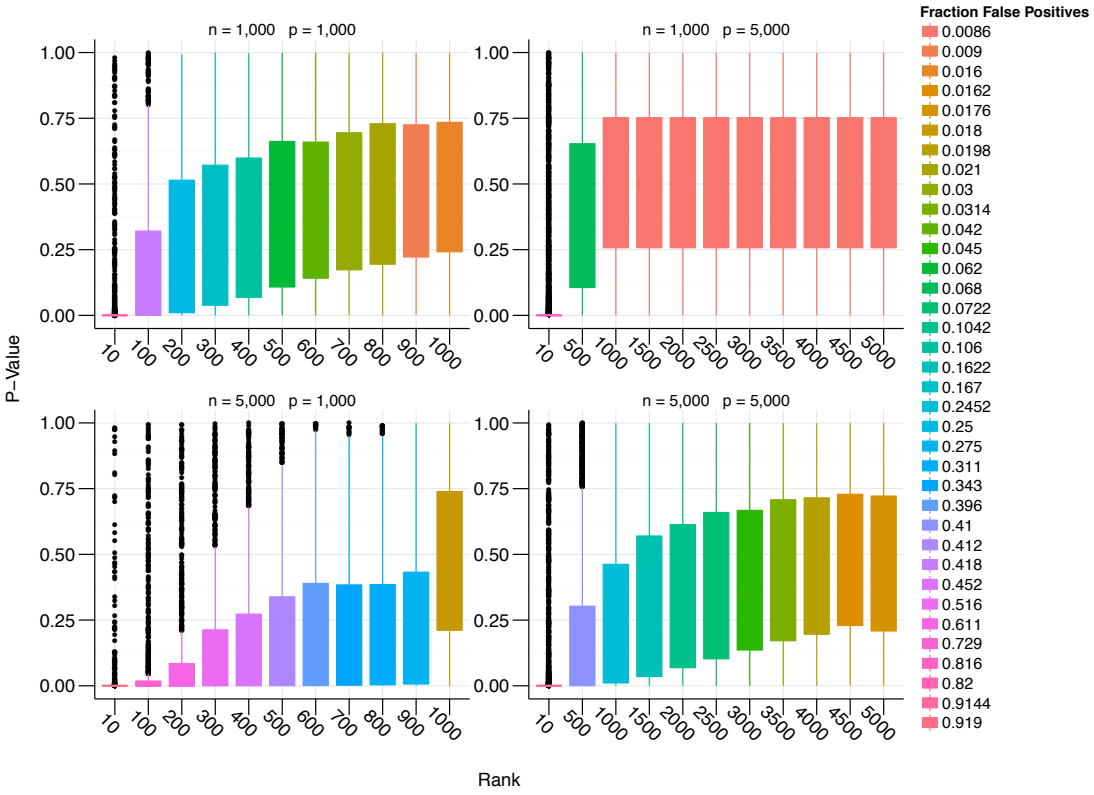

Figure 4: **Controlling for structure.** RSVD with pylmm is applied to four simulation settings with varying n and p. The x-axis is the setting of the ARSVD parameter d^*. The y-axis corresponds to the p-values from the LMM for each of the p features. The color of each box plot represents the fraction of false positive rate of the LMM using ARSVD.

association studies. Our LMM with ARSVD procedure uses EMMAX (pylmm) to solve the LMM. The two methods we compared to are EMMAX with the addition of ARSVD and GEMMA (Zhou et al., 2013) (GEMMA is executed with the `-lmm` option to most closely approximate the analysis that pylmm performs).

ARSVD on the whole genome took 82.2 seconds, while a traditional eigendecomposition of the covariance matrix in pylmm took 88 mins 23.9 seconds. In order to most accurately control for the test statistic computed in a LMM and to achieve maximal statistical power, it is suggested that a covariance matrix is constructed once per (22) chromosomes, performed by holding out the test chromosome and concatenating the remaining chromosomes (Yang et al., 2014). Our method performs the 22 decompositions in a total of 5 mins 4.8 secs, while the traditional decomposition method takes 4 hrs 24 mins.

In the remainder of this subsection, we denote the culling of EMMAX with ARSVD as pylmm. It has been observed (Zhou et al., 2013) that the effect sizes of the coefficients are similar whether one applies linear regression or logistic regression to most case-control genomic studies. Estimates

of β for pylmm and GEMMA are strongly correlated. The distributions of p-values computed by pylmm show enrichment in low p-values. This suggests that the regularization in pylmm may capture additional associations.

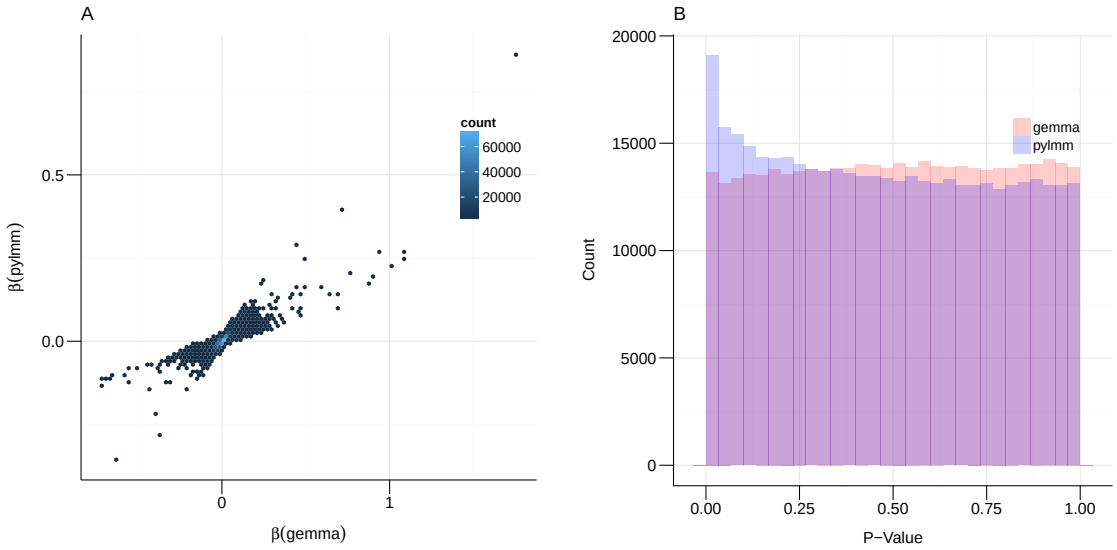

Figure 5: **Comparison of pylmm and GEMMA.** A) A scatter plot of the β-values for GEMMA on the x-axis versus the β-values for pylmm on the y-axis. B) Histogram of p-values for both methods, GEMMA in red and pylmm in blue.

We compared the most significant associations that we identified to results from other analyses of Crohn's disease. One source of associations is from a large-scale meta-analysis of Crohn's disease consisting of 6,333 affected individuals (cases) and 15,056 controls, and the top association signals were followed up on in 15,694 cases, 14,026 controls, and 414 parent-offspring trios (Franke et al., 2010). We denote this list as MA. In Listgarten et al. (2012), the list of associated genetic variants collected includes the MA list as well as the WTCCC list. The major histocompatibility complex (MHC) is a region that has been previously associated to Crohn's disease and autoimmune disease in general. We denote the list of variants in this region as MHC. We report the overlap between the results obtained from our method to the MA list, the union of the WTCCC and MA lists, and the union of the WTCCC, MA, and MHC lists. (Table 3). We select our top hits using two cutoffs: the top 0.5% with respect to negative log p-value, and the associations that pass a local false discovery rate (LFDR) of 5% (Strimmer, 2008).

We identified several potential genetic variants associated with Crohn's that were previously unidentified. In particular, there are a few genetic variants within the 3.6MB region that defines the MHC on chromosome 6 in the human genome (sequencing consortium, 1999) at an LFDR threshold of 10%. Genetic variant rs9269186 ($p \leq 7.03 \times 10^{-7}$) is below the LFDR threshold of 1% and lies within 5 kilobases (KB) of the start of the *HLA-DRB5* gene, putatively acting to regulate transcription of this protein-coding gene that produces a membrane-bound class II molecule. The

HLA-DRB5 protein is an antigen that has an important role in the human immune system, and thus may play a role in an autoimmune disorder such as Crohn's disease.

source	significant variants	top 0.5%	LFDR 5%
MA+WTCC+MHC	151	61	35
MA + WTCCC	93	48	30
MA	81	47	29

Table 3: **Overlap of genetic associations with prior studies.** Columns represent the number of associations (significant variants) in the MA, WTCCC studies and the number of variants in the MHC region; the overlap of our list with the associations in these lists with a cutoff of the top 0.5% of our associations; and overlap at an LFDR threshold of 5%.

5. Discussion

Massively high-dimensional data sets are ubiquitous in modern data analysis settings. In this paper, we provide a scale method for accurate computation using spectral decompositions for data analysis. The main computational tool we use is based on recent randomized algorithms developed by the numerical analysis community. To address the issue of noise, we provide an adaptive procedure to estimate both the rank d^* of the lower dimensional projection and the number of Krylov iterations t^* for the randomized approximate SVD. Using this adaptive estimator of low-rank structure, we implement efficient algorithms for PCA and linear mixed models. An interesting observation both from an empirical and theoretical perspective is that our randomized algorithm implicitly imposes regularization.

In simulated experiments we show high accuracy in recovering the true (latent) rank of matrices with low-rank substructure, without the need for many Krylov iterations. Additionally, we show in simulations that our method performs implicit regularization and improves quantitative properties of results under various types of data structure. Furthermore, our results on large genome-wide association studies show that our approach to using ARSVD in linear mixed models fills a critical need for methods with computational efficiency that do not sacrifice the desirable statistical properties of traditional LMMs.

Some important open questions still remain:

(1) There is need for a more refined theoretical framework to quantify what generalization guarantees the randomization algorithm can provide on out-of-sample data, and the dependence of this bound on the noise and the structure in the data on one hand and on the parameter settings on the other.

(2) A probabilistic interpretation of the algorithm could contribute additional insights (Mahoney, 2011) into the practical utility of the proposed approach under different assumptions. In particular, it would be interesting to relate our work to a Bayesian model with posterior modes that correspond to the subspaces estimated by the randomized approach.

(3) The implicit regularization on latent factors imposed by ARSVD should be further explored with respect to the structure of the noise, and its impact on estimates of random effects in LMMs (Runcie and Mukherjee, 2013).

Acknowledgments

SM would like to acknowledge Lek-Heng Lim, Michael Mahoney, Qiang Wu, and Ankan Saha. SM is pleased to acknowledge support from grants NIH (Systems Biology): 5P50-GM081883, AFOSR: FA9550-10-1-0436, NSF CCF-1049290, and NSF-DMS-1209155. GD and BEE would like to acknowledge Joel Tropp. BEE is pleased to acknowledge support from grants NIH R00 HG006265, NIH R01 MH101822, NIH U01 HG007900, and a Sloan Faculty Fellowship. SG would like to acknowledge Uwe Ohler, Jonathan Pritchard and Ankan Saha. The software for ARSVD is available at `https://github.com/gdarnell/arsvd`.

Appendix A. Generalized Eigendecomposition and Dimension Reduction

The appendix states a variety of dimension reduction methods supervised, unsupervised, and non-linear that can use the ARSVD engine to scale to massive data. The key requirement is a formulation of the truncated generalized eigendecomposition problem that can be implemented by the *Adaptive Randomized SVD* from Section 2.4. The dimension reduction methods we will focus on are sliced inverse regression (SIR) and localized sliced inverse regression (LSIR).

A.1 Problem Formulation

Assume we are given $\Sigma \in \mathbb{S}^p_{++}, \Gamma \in \mathbb{S}^p_+$ that characterize pairwise relationships in the data and let $r \ll \min(n, p)$ be the "intrinsic dimensionality" of the information contained in the data. In the case of supervised dimension reduction methods this corresponds to the dimensionality of the linear subspace to which the joint distribution of (X, Y) assigns non-zero probability mass. Our objective is to find a basis for that subspace. For SIR and LSIR this corresponds to the span of the generalized eigenvectors $\{g_1, \ldots, g_r\}$ with largest eigenvalues $\{\lambda_{\max} = \lambda_1 \geq \ldots \geq \lambda_r\}$:

$$\Gamma g = \lambda \Sigma g. \tag{6}$$

An important structural constraint we impose on Γ, which is applicable to a variety of high-dimensional data settings, is that it has low-rank: $r \leq d^* \equiv \text{rank}(\Gamma) \ll p$. It is this constraint that we will take advantage of in the randomized methods. In the case of $\Sigma = \text{I}$ (unsupervised case), $r = d^*$.

A.2 Sufficient Dimension Reduction

Dimension reduction is often a first step in the statistical analysis of high-dimensional data and could be followed by data visualization or predictive modeling. If the ultimate goal is the latter, then the statistical quantity of interest is a low dimensional summary $Z \equiv R(X)$ which captures all the predictive information in X relevant to Y:

$$Y = f(X) + \varepsilon = h(Z) + \varepsilon, \quad X \in \mathbb{R}^p, Z \in \mathbb{R}^r,\ r \ll p.$$

Sufficient dimension reduction (SDR) is one popular approach for estimating Z (Li, 1991; Cook and Weisberg, 1991; Li, 1992; Li et al., 2005; Nilsson et al., 2007; Sugiyama, 2007; Cook, 2007; Wu et al., 2010). In this appendix we focus on linear SDRs: $G = (g_1, \ldots, g_r) \in \mathbb{R}^{p \times r} \Rightarrow R(X) = G^T X$, which provide a prediction-optimal reduction of X.

$$(Y \mid X) \stackrel{d}{=} (Y \mid G^T X), \quad \stackrel{d}{=} \text{ is equivalence in distribution.}$$

We will consider two specific dimension reduction methods: Sliced Inverse Regression (SIR) (Li, 1991) and Localized Sliced Inverse Regression (LSIR) (Wu et al., 2010). SIR is effective when the predictive structure in the data is global, i.e., there is single predictive subspace over the support of the marginal distribution of X. In the case of local or manifold predictive structure in the data, LSIR can be used to compute a projection matrix G that contains this non-linear (manifold) structure.

A.3 Efficient Solutions and Approximate SVD

SIR and LSIR reduce to solving a truncated generalized eigendecomposition problem as formulated in (6). Since we consider estimating the dimension reduction based on sample data we focus on the sample estimators $\widehat{\Sigma} = \frac{1}{n} X^T X$ and $\widehat{\Gamma}_{XY} = X^T K_{XY} X$, where K_{XY} is symmetric and encodes the method-specific grouping of the samples based on the response Y. In the classic statistical setting, when $n > p$, both $\widehat{\Sigma}$ and $\widehat{\Gamma}_{XY}$ are positive definite almost surely. Then, a typical solution proceeds by first sphering the data: $Z = \widehat{\Sigma}^{-\frac{1}{2}} X$, e.g., using a Cholesky or SVD representation $\widehat{\Sigma} = \widehat{\Sigma}^{\frac{1}{2}} (\widehat{\Sigma}^{\frac{1}{2}})^T$. This is followed by eigendecomposition of $\widehat{\Gamma}_{ZY}$ Li (1991); Wu et al. (2010) and back-transformation of the top eigenvectors directions to the canonical basis. The computational time is $O(np^2)$. When $n < p$, $\hat{\Sigma}$ and $\hat{\Gamma}$ are rank-deficient and a unique solution to the problem (6) does not exist. One widely-used approach, which allows us to make progress in this problematic setting, is to restrict our attention to the directions in the data with positive variance. Then we can proceed as before, using an orthogonal projection onto the span of the data. The total computation time in this case is $O(n^2 p)$. In many modern data analysis applications both n and p are very large, and hence algorithmic complexity of $O[\max(n,p) \times \min(n,p)^2]$ could be prohibitive, rendering the above approaches unusable. We propose an approximate solution that explicitly recovers the low-rank structure in Γ using *Adaptive Randomized SVD* from Section 2.4. In particular, assume rank $(\Gamma) = d^* \geq r$ (where r is the dimensionality of the optimal dimension reduction subspace). Then $\Gamma \stackrel{\text{svd}}{=} U S^2 U^T$, where $U \in \mathbb{R}^{p \times d^*}$. The generalized eigendecomposition problem (6) solution becomes restricted to the subspace spanned by the columns of Γ:

$$S^{-1} U^T \Sigma U S^{-1} e \;=\; \frac{1}{\lambda} e, \qquad e \equiv S U^T g. \tag{7}$$

The dimension reduction subspace is contained in the span(G), where

$$G = (U S^{-1} e_1, \ldots, U S^{-1} e_r).$$

References

D. Achlioptas. Database-friendly random projections. In *Proceedings of the Twentieth Symposium on Principles of Database Systems*, PODS '01, pages 274–281, New York, NY, USA, 2001. ACM. ISBN 1-58113-361-8.

R.J. Adcock. A problem in least squares. *The Analyst*, 5:53–54, 1878.

J. Baglama and L. Reichel. Restarted block Lanczos bidiagonalization methods. *Numerical Algorithms*, 43(3):251–272, 2006.

C.M. Bishop. Training with noise is equivalent to Tikhonov regularization. *Neural Comput.*, 7 (1):108–116, January 1995. ISSN 0899-7667. doi: 10.1162/neco.1995.7.1.108. URL `http://dx.doi.org/10.1162/neco.1995.7.1.108`.

C. Boutsidis, M.W. Mahoney, and P. Drineas. An improved approximation algorithm for the column subset selection problem. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '09, pages 968–977. Society for Industrial and Applied Mathematics, 2009.

The Wellcome Trust Case Control Consortium. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*, 447(7145):661–678, 2007.

R.D. Cook. Fisher Lecture: Dimension Reduction in Regression. *Statistical Science*, 22(1):1–26, 2007.

R.D. Cook and S. Weisberg. Discussion of Li (1991). *J. Amer. Statist. Assoc.*, 86:328–332, 1991.

S. Dasgupta and A. Gupta. An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003.

P. Drineas and M.W. Mahoney. On the Nystrom Method for Approximating a Gram Matrix for Improved Kernel-Based Learning. *J. Mach. Learn. Res.*, 6:2153–2175, December 2005.

P. Drineas, R. Kannan, and M.W. Mahoney. Fast Monte Carlo Algorithms for Matrices II: Computing a Low-Rank Approximation to a Matrix. *SIAM J. Comput.*, 36:158–183, July 2006.

P. Drineas, M. Magdon-Ismail, M.W. Mahoney, and D.P. Woodruff. Fast approximation of matrix coherence and statistical leverage. *The Journal of Machine Learning Research*, 13(1):3475–3506, 2012.

F.Y. Edegworth. On the reduction of observations. *Philosophical Magazine*, pages 135–141, 1884.

R.A. Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Statistical Society A*, 222:309–368, 1922.

R.A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7 (2):179–188, 1936.

A. Franke, D.P.B. McGovern, J.C Barrett, K. Wang, G.L. Radford-Smith, T. Ahmad, C.W. Lees, T. Balschun, J. Lee, R. Roberts, et al. Genome-wide meta-analysis increases to 71 the number of confirmed crohn's disease susceptibility loci. *Nature genetics*, 42(12):1118–1125, 2010.

P. Frankl and H. Maehara. The Johnson-Lindenstrauss lemma and the sphericity of some graphs. *J. Comb. Theory Ser. A*, 44:355–362, June 1987.

K.R. Gabriel. Le biplot - outil d'exploration de données multidimensionnelles. *Journal de la société française de statistique*, 143(3-4):5–55, 2002.

L.L. Gerfo, L. Rosasco, F. Odone, E. De Vito, and A. Verri. Spectral algorithms for supervised learning. *Neural Computation*, 20(7):1873–1897, 2008.

A.R. Gilmour, R. Thompson, and B.R Cullis. Average information reml: an efficient algorithm for variance parameter estimation in linear mixed models. *Biometrics*, pages 1440–1450, 1995.

A. Gittens and M.W. Mahoney. Revisiting the Nystrom method for improved large-scale machine learning. *arXiv preprint arXiv:1303.1849*, 2013.

G.H. Golub. Matrix decompositions and statistical calculations. In *Statistical Computation*, pages 365–397. New York: Academic Press, 1969.

G.H. Golub and C.F. Van Loan. *Matrix computations*. John Hopkins University Press, Baltimore, MD, 3rd edition, 1996. ISBN 0-8018-5413-X.

G.H. Golub, K. Sløna, and P. Van Dooren. Computing the SVD of a General Matrix Product/Quotient. *SIAM J. Matrix Anal. Appl*, 22:1–19, 2000.

M. Gu. Subspace iteration randomization and singular value problems. *SIAM Journal on Scientific Computing*, 37(3):A1139–A1173, 2015.

M. Gu and S.C. Eisenstat. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM J. Sci. Comput.*, 17(4):848–869, July 1996.

N. Halko, P-G. Martinsson, and J.A. Tropp. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *SIAM Review*, 53(2):217–288, 2011.

R. Hecht-Nielsen and Věrá Kůrková. Quasiorthogonal dimension of euclidean spaces. *Technical Report Series*, INC-9205, 1992.

C.R. Henderson. *Applications of Linear Models in Animal Breeding*. University of Guelph, 1984.

A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

H. Hotelling. Analysis of a complex of statistical variables in principal components. *Journal of Educational Psychology*, 24:417–441, 1933.

P. Indyk and R. Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, STOC '98, pages 604–613, New York, NY, USA, 1998. ACM. ISBN 0-89791-962-9.

D.L. Johnson and R. Thompson. Restricted maximum likelihood estimation of variance components for univariate animal models using sparse matrix techniques and average information. *Journal of dairy science*, 78(2):449–456, 1995.

W. Johnson and J. Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. In *Conference in modern analysis and probability (New Haven, Conn., 1982)*, volume 26 of *Contemporary Mathematics*, pages 189–206. American Mathematical Society, 1984.

P.C. Kainen and V. Kůrková. Quasiorthogonal dimension of euclidean spaces. *Applied mathematics letters*, 6(3):7–10, 1993.

H.M. Kang, N.A. Zaitlen, C.M. Wade, A. Kirby, D. Heckerman, MJ. Daly, and E. Eskin. Efficient control of population structure in model organism association mapping. *Genetics*, 178(3):1709–1723, 2008.

H.M. Kang, J.H. Sul, S.K. Service, N.A. Zaitlen, S. Kong, N.B. Freimer, C. Sabatti, and E. Eskin. Variance component model to account for sample structure in genome-wide association studies. *Nature genetics*, 42(4):348–354, 2010.

B.W. Kennedy, M. Quinton, and J.A. Van Arendonk. Estimation of effects of single genes on quantitative traits. *Journal of Animal Science*, 70(7):2000–2012, 1992.

A. Krote, B.J. Vilhjálmsson, V. Segura, A. Platt, Q. Long, and M. Nordburg. A mixed-model approach for genome-wide association studies of correlated traits in structured populations. *Nature genetics*, 44(9):106610711, 2012.

R.R.B. Lehoucq, D.D.C. Sorensen, and C-C. Yang. *Arpack User's Guide: Solution of Large-Scale Eigenvalue Problems With Implicitly Restorted Arnoldi Methods*, volume 6. SIAM, 1998.

B. Li, H. Zha, and F. Chiaromonte. Contour Regression: A General Approach to Dimension Reduction. *The Annals of Statistics*, 33(4):1580–1616, 2005.

K.C. Li. Sliced inverse regression for dimension reduction (with discussion). *J. Amer. Statist. Assoc.*, 86:316–342, 1991.

K.C. Li. On principal Hessian directions for data visulization and dimension reduction: another application of Stein's lemma. *J. Amer. Statist. Assoc.*, 87:1025–1039, 1992.

E. Liberty, F. Woolfe, P-G. Martinsson, V. Rokhlin, and M. Tygert. Randomized algorithms for the low-rank approximation of matrices. *Proceedings of the National Academy of Sciences*, 104(51): 20167–20172, 2007.

B. Lin, Z. Pang, and J. Jiang. Fixed and random effects selection by reml and pathwise coordinate optimization. *Journal of Computational and Graphical Statistics*, 22(2):341–355, 2013.

C. Lippert, J. Listgarten, Y. Liu, C.M. Kadie, R.I. Davidson, and D. Heckerman. Fast linear mixed models for genome-wide association studies. *Nature Methods*, 8(10):833–835, 2011.

C. Lippert, G. Quon, E.Y. Kang, C.M. Kadie, J. Listgarten, and D. Heckerman. The benefits of selecting phenotype-specific variants for applications of mixed models in genomics. *Scientific reports*, 3, 2013.

J. Listgarten, C. Lippert, C.M. Kadie, R.I. Davidson, E. Eskin, and D. Heckerman. Improved linear mixed models for genome-wide association studies. *Nature methods*, 9(6):525–526, 2012.

M.W. Mahoney. Randomized Algorithms for Matrices and Data. *Foundations and Trends in Machine Learning*, 3(2):123–224, 2011.

M.W. Mahoney and L. Orecchia. Implementing regularization implicitly via approximate eigenvector computation. *arXiv preprint arXiv:1010.0703*, 2010.

P.-G. Martinsson, A. Szlam, and M. Tygert. Normalized power iterations for the computation of SVD. *NIPS workshop on low-rank methods for large-scale machine learning*, 2010.

K. Matilainen, E.A. Mäntysaari, M.H. Lidauer, I. Strandén, and R. Thompson. Employing a monte carlo algorithm in newton-type methods for restricted maximum likelihood estimation of genetic parameters. *PloS one*, 8(12):e80821, 2013.

D. Mimno, D.M. Blei, and B.E. Engelhardt. Posterior predictive checks to quantify lack-of-fit in admixture models of latent population structure. *arXiv preprint arXiv:1407.0050*, 2014.

J. Nilsson, F. Sha, and M.I. Jordan. Regression on Manifolds Using Kernel Dimension Reduction. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

A.B. Owen and P.O. Perry. Bi-cross-validation of the SVD and the nonnegative matrix factorization. *The Annals of Applied Statistics*, 3:564–594, 2009. doi: 10.1214/08-AOAS227.

P.O. Perry and M.W. Mahoney. Regularized laplacian estimation and fast eigenvector approximation. In *Advances in Neural Information Processing Systems*, pages 2420–2428, 2011.

T. Poggio and F. Girosi. Regularization Algorithms for Learning that are Equivalent to Multilayer Networks. *Science*, 247:978–982, February 1990. doi: 10.1126/science.247.4945.978.

N.G. Polson and J.G. Scott. Shrink Globally, Act Locally: Sparse Bayesian Regularization and Prediction. In *Bayesian Statistics 9*. Oxford University Press, 2010.

A.L. Price, A. Helgason, G. Thorleifsson, S.A. McCarroll, A. Kong, and K. Stefansson. Single-tissue and cross-tissue heritability of gene expression via identity-by-descent in related or unrelated individuals. *PLoS Genet*, 7, 02 2011.

J.K. Pritchard and P. Donnelly. Case–control studies of association in structured or admixed populations. *Theoretical population biology*, 60(3):227–237, 2001.

V. Rokhlin, A. Szlam, and M. Tygert. A Randomized Algorithm for Principal Component Analysis. *SIAM J. Matrix Anal. Appl.*, 31(3):1100–1124, August 2009.

A. Rudi, G.D. Cañas, and L. Rosasco. On the sample complexity of subspace learning. In *Advances in Neural Information Processing Systems*, pages 2067–2075, 2013.

A. Rudi, R. Camoriano, and L. Rosasco. Less is more: Nyström computational regularization. In *Advances in Neural Information Processing Systems*, pages 1648–1656, 2015.

D.E. Runcie and S. Mukherjee. Dissecting high-dimensional phenotypes with bayesian sparse factor analysis of genetic covariance matrices. *Genetics*, 194(3):753–767, 2013.

Y. Saad. *Numerical methods for large eigenvalue problems*, volume 158. SIAM, 1992.

T. Sarlos. Improved approximation algorithms for large matrices via random projections. In *Foundations of Computer Science, 2006. FOCS '06. 47th Annual IEEE Symposium on*, pages 143 –152, October 2006.

The MHC sequencing consortium. Complete sequence and gene map of a human major histocompatibility complex. *Nature*, 401(6756):921–923, 1999.

P. Simard, Y. LeCun, and J.S. Denker. Efficient pattern recognition using a new transformation distance. In *Advances in Neural Information Processing Systems 5, [NIPS Conference]*, pages 50–58, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc. ISBN 1-55860-274-7. URL http://dl.acm.org/citation.cfm?id=645753.668226.

S. Smale and D. Zhou. Learning theory estimates via integral operators and their approximations. *Constructive approximation*, 26(2):153–172, 2007.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, January 2014. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=2627435.2670313.

G.W. Stewart. *Matrix Algorithms: Volume 2, Eigensystems*, volume 2. SIAM, 2001.

K. Strimmer. fdrtool: a versatile r package for estimating local and tail area-based false discovery rates. *Bioinformatics*, 24(12):1461–1462, 2008.

M. Sugiyama. Dimension reduction of multimodal labeled data by local Fisher discriminant analysis. *Journal of Machine Learning Research*, 8:1027–1061, 2007.

A. Szlam, Y. Kluger, and M. Tygert. An implementation of a randomized algorithm for principal component analysis. *arXiv preprint arXiv:1412.3510*, 2014.

E.A. Thompson. Population correlation and population kinship. *Theoretical population biology*, 10 (2):205–226, 1976.

E.D. Vito, L. Rosasco, A. Caponnetto, U.D. Giovannini, and F. Odone. Learning from examples as an inverse problem. *Journal of Machine Learning Research*, 6(May):883–904, 2005.

C.K.I. Williams and M. Seeger. Using the nyström method to speed up kernel machines. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 682–688. MIT Press, 2001. URL http://papers.nips.cc/paper/1866-using-the-nystrom-method-to-speed-up-kernel-machines.pdf.

Q. Wu, F. Liang, and S. Mukherjee. Localized Sliced Inverse Regression. *Journal of Computational and Graphical Statistics*, 19(4):843–860, 2010.

J. Yang, S.H. Lee, M.E. Goddard, and P.M Visscher. GCTA: a tool for genome-wide complex trait analysis. *The American Journal of Human Genetics*, 88(1):76–82, 2011.

J. Yang, N.A. Zaitlen, M.E. Goddard, P.M. Visscher, and A.L. Price. Advantages and pitfalls in the application of mixed-model association methods. *Nature genetics*, 46(2):100–106, 2014.

Y. Yao, L. Rosasco, and A. Caponnetto. On early stopping in gradient descent learning. *Constructive Approximation*, 26(2):289–315, 2007.

G. Young. Maximum likelihood estimation and factor analysis. *Psychometrika*, 6:49–53, 1941.

X. Zhou and M. Stephens. Genome-wide efficient mixed-model analysis for association studies. *Nature genetics*, 44(7):821–824, 2012.

X. Zhou, P. Carbonetto, and M. Stephens. Polygenic modeling with Bayesian sparse linear mixed models. *PLoS genetics*, 9(2):e1003264, 2013.