MACHINE LEARNING METHODS FOR FEATURE SELECTION AND PREDICTION APPLIED TO LARGE SCALE GENETICS DATA

By

Anirban Samaddar

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Statistics—Doctor of Philosophy

2023

ABSTRACT

The age of big data has brought exciting opportunities to elicit new insights in many scientific fields. In Genetics, big data-driven technologies can be transformative. Although big data-powered innovations are making strides in Genetics, many vital challenges remain. This dissertation focuses on developing statistical and machine learning methods for addressing the critical challenges of these complex genomic data sets.

The first chapter addresses the challenges of variable selection due to severe collinearity in the features in Genome-Wide Association (GWA) studies involving millions of DNA variants (e.g., SNPs), many of which may be in highly correlated. We devised a novel Bayesian hierarchical hypothesis testing (BHHT). We present simulation results that show that demonstrate that the proposed method can lead to high power with adequate error control and fine mapping resolution. Furthermore, we demonstrate the feasibility of using the proposed methodology with big data by using it to map risk variants for serum urate using data UK-Biobank ($n \sim 300,000$, $p \sim 15$ million SNPs).

Chapter two focuses on developing a Bayesian prior for improving the robustness of deep latent variable models. The information bottleneck framework provides a systematic approach to learning representations that compress nuisance information in the input and extract relevant information for predictions. We present a novel sparsity-inducing spike-slab categorical prior that uses sparsity as a mechanism to provide the flexibility that allows each data point to learn its own dimension distribution. Through a series of experiments using in-distribution and out-of-distribution learning scenarios on the MNIST, CIFAR-10, and ImageNet data, we show that the proposed approach improves accuracy and robustness compared to traditional fixed-dimensional priors, as well as other sparsity induction mechanisms for latent variable models proposed in the literature.

In the third chapter, we develop and benchmark machine learning for genomic prediction with ancestry-diverse data sets. Genomic prediction is commonly made by constructing polygenic scores (PGS) which are a linear combination of risk variants (SNPs). However, in modern genetic data sets complexities arise due to the presence of diverse ancestry groups. We develop a strategy to

use deep learning for genomic prediction that leverages non-linear input-output patterns among physically proximal SNPs. Using TOPMed genotype data and Monte Carlo simulations, we evaluated whether local regressions using machine learning methodology can outperform linear models in cross-ancestry prediction.

In summary, this thesis contributes novel statistical methods for mapping risk variants in the presence of collinearity, and novel machine learning methodology to infer latent representations of complex data sets and to predict disease risk using ultra-high dimensional genomic data.

To my parents Anita Samaddar and Pradip Kumar Samaddar

ACKNOWLEDGEMENTS

I want to convey my gratitude to my advisors Dr. Gustavo de los Campos, and Dr. Tapabrata Maiti for their guidance, encouragement, and insight that has helped shaped my Ph.D. experience. I would also like to extend thanks to my wonderful committee members Dr. Shrijita Bhattacharya and Dr. Chih-li Sung for their valuable feedback on my dissertation. Additionally, I would like to thank Dr. Sandeep Madireddy and Dr. Prasanna Balaprakash from the Argonne National Laboratory for providing me with the opportunity to work on cutting-edge problems during my internship and beyond. Finally, I would like to thank all my professors at Michigan State University (MSU) for enhancing my knowledge in the field of Statistics and making my academic experience truly fulfilling.

Beyond MSU, I am thankful to some of the exceptional teachers that have shaped my academic journey. I am grateful to my high-school math teacher Tushar sir for nurturing my early interest in Mathematics. I want to thank my professors at Narendrapur Ramakrishna Mission and the University of Calcutta who ignited my passion for statistics. Among them, I am particularly indebted to Dr. Gaurangadeb Chattopadhyay for his guidance and motivation during my Master's at the University of Calcutta which led me to pursue higher studies in Statistics.

This dissertation is not only a product of my efforts but of those who have always been there for me along the way. I would like to start by thanking my mother and father for the many sacrifices they made to support me in pursuing my dreams. Graduate life can be difficult sometimes but thinking about their struggles has always helped me to persevere. I like to thank my elder brother for being a constant source of positivity and inspiration throughout my life. I would like to also thank my sister-in-law who always treated me like her own brother and who I feel is always on my side. Finally, I would like to thank my wife Sampriti who makes my life brighter every day and who is always there for me when it rains.

I am fortunate to born and brought up in a close-knit family. I want to pay tribute to my late paternal and maternal grandparents who would have been proud to see this day. In addition, I would like to say thanks to - Tata, Boro Piso, Mani, Choto Piso, Jemma, Jethu, Akaki, Akaka, Tu Kaki, Tu

Kaka, Notun Dadu, Notun Mammam, Boro Mami, Choto Mama, Choto Mami who has inspired and supported me throughout my career. I would also like to thank my father- and mother-in-law, Mani, Jaya di, and Apu da for providing constant encouragement to achieve my goals. Finally, I would like to pay respect to my late *pisemosai* Dr. Shyamal Chakraborty who advised me to pursue a bachelor's degree in Statistics, and now thirteen years later when I am at the other end of my educational journey I miss him very much.

My academic experience in the past five years is incomplete without my friends around East Lansing and beyond. I extend my heartfelt gratitude to Arka, who has been my companion since the beginning of my Ph.D. voyage, consistently offering motivation and encouragement. Sikta, too, holds a special place for teaching me how to find joy amidst the demanding graduate studies routine. I am also thankful to Arka, Sikta, and Sampriti for the memorable trips and Catan nights that added a wonderful balance to my life. In addition, I would like to thank my friends in the department - Mookyong, Hema, Tathagata, Sumegha, and my friends from the applied economics department (MSU-AFRE) Myat, and Yeyoung for their constant support and friendship.

I want to extend my heartfelt gratitude to the friends who extend beyond the borders of East Lansing. To Sayak, whose engaging conversations about science and history have consistently ignited my curiosity; to Dinu, with whom I've had illuminating discussions about research and music; to Arindam, a confidant with whom I can openly share my thoughts; and to Vaibhav, who has stood by me as a true brother. Furthermore, my appreciation extends to an exceptional group of individuals - Amartya, Parthajit, Santanu, Arabinda, Prolay, and Mishra - whose friendship has enriched my life beyond measure, offering the finest friendship one could hope for. Lastly, I am immensely thankful to Sourjya, who has remained unwaveringly by my side as both a friend and a brother, a constant source of support throughout my life.

To anyone I may have unintentionally omitted, please know that your love and support are deeply cherished, and my heart remains forever indebted to you. Your presence in my journey has been invaluable, and for that, I am eternally thankful.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The age of big data has brought exciting opportunities to elicit new insights in many scientific fields. In Genetics, big data revolution-driven technologies, such as pharmacogenomics and precision medicine can be transformative. Although big data-powered innovations are making strides in Genetics, many vital challenges remain. This dissertation focuses on developing statistical methods for addressing critical challenges that emerge in the analysis of ultra-high-dimensional biobank-size data sets. Specifically, in this thesis, we have developed methods for the two main aspects of **statistical learning – (1) variable selection, and (2) prediction**.

Chapter one addresses the challenges of variable selection due to severe collinearity in the features in Genome-Wide Association Studies (GWAS) involving millions of variants, many of which may be in high linkage disequilibrium. We devised a Bayesian hierarchical hypothesis testing (BHHT)–a novel multi-resolution testing procedure that offers high power with adequate error control and fine-mapping resolution. Using Monte Carlo simulations, we show that the proposed procedure offers high power with adequate error control and fine-mapping resolution. Finally, we demonstrate the feasibility of using the proposed methodology to map risk variants for serum urate using ultra-high dimensional ($\sim$ 15 millions of SNPs) biobank size data ($n \sim 300,000$) from the UK-Biobank.

Our results show that the proposed methodology leads to many more discoveries than those obtained using traditional feature-centered inference procedures. We provide publicly available software for applying the proposed procedure at scale in the following repository: https://github.com/AnirbanSamaddar/Bayes_HHT.

Chapter two focuses on developing a Bayesian prior for improving the robustness of deep latent variable models. The information bottleneck framework [4] provides a systematic approach to learning representations that compress nuisance information in the input and extract semantically meaningful information about predictions. We present a novel sparsity-inducing spike-slab categorical prior (similar to the one used for variable selection in chapter one) that uses sparsity

as a mechanism to provide the flexibility that allows each data point to learn its own dimension distribution. Through a series of experiments using in-distribution and out-of-distribution learning scenarios on the MNIST [3], CIFAR-10 [6], and ImageNet [2] data, we show that the proposed approach improves accuracy and robustness compared to traditional fixed-dimensional priors, as well as other sparsity induction mechanisms for latent variable models proposed in the literature. The open-source software implementation is made available in the following repository: https://github.com/AnirbanSamaddar/SparC-IB.

Chapter three integrates the Bayesian linear variable selection methodology used in chapter one with deep learning methods to develop a novel machine learning (ML) framework for genomic prediction in ancestry-diverse data sets. Genomic prediction (i.e., prediction of phenotypes or disease risk using DNA information, e.g., SNPs) is commonly made by constructing polygenic scores (PGS) which is a linear combination of risk variants (SNPs) that were found in a Genome-Wide Association (GWA) study to be associated with a phenotype or disease.

Most GWA studies use linear models to construct PGS because of its simplicity and effectiveness in capturing phenotypic variability within ancestry-homogenous populations. However, in modern genetic data sets complexities arise due to the presence of diverse ancestry groups. There are biological (e.g., gene-gene or genetic-by-environment interactions) as well as technological reasons (the fact that prediction models use SNPs that are imperfect surrogates for those with causal effects) that can lead to ancestry-specific SNP effects. Such heterogeneity is not well-captured by linear PGS.

Theory and some empirical evidence suggest that SNP-SNP interactions are more prominent among SNPs that are physically proximal in their genome position. Therefore, we develop an ML methodology that aims at capturing non-linear local patterns. Our approach also considers the computational complexities that arise from the application of ML methods to ultra-high-dimensional data.

Using TOPMed [5] genotype data and Monte Carlo simulations we evaluated whether local regressions using machine learning methodology can outperform linear models in cross-ancestry

prediction. In a first simulation using short chromosome segments (100 or 1000 kilo-base-pairs, kbp) our results show that even in cases when the underlying causal model is strictly linear, due to imperfect linkage disequilibrium (a term used in genetics to describe the imperfect correlation between pairs of SNPs) between the SNPs used in the model and those with causal effects, ML methodology can potentially outperform linear models–the difference between ML and linear models was particularly sizable for short segments (100kbp) perhaps illustrating difficulties that ML method face when trying to extract fine patterns when the underlying structure is sparse and the input set becomes high-dimensional.

Then we extend the simulation and the methods to entire chromosomes. This setting presents a serious computational challenge (in TOPMed, in chromosome one there are more than 40 million SNPs). To address the statistical and computational challenges that emerge when applying ML methods to ultra-high dimensional data, and budling upon ideas that were previously used for linear models [1], we propose a two-step procedure whereas, in the first step, local PGS (ML-derived or linear ones) are derived for short chromosome segments and in a second step these scores are combined using either ML or linear methods. In total, we evaluate four procedures that emerge from the use of either ML or linear models in the first and second steps. Our simulations show that using a local Bayesian Variable Selection method in the first step and then combining the resulting scores using ML yields higher prediction accuracy in cross-ancestry prediction than using ML or linear models alone. These results are significant because they provide avenues to improve genomic prediction using ancestry-diverse data using methods that scale to whole-genome applications.

Overall, this thesis contributes to advanced methods for research problems involving both inference (e.g., fine mapping) and prediction. The methods that we develop are implemented in scalable open-source software which we make available through GitHub repositories.

BIBLIOGRAPHY

[1] de Los Campos, G., Grueneberg, A., Funkhouser, S., Pérez-Rodríguez, P., and Samaddar, A. (2023). Fine mapping and accurate prediction of complex traits using bayesian variable selection models applied to biobank-size data. *European Journal of Human Genetics*, 31(3):313–320.

[2] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

[3] Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142.

[4] Goldfeld, Z. and Polyanskiy, Y. (2020). The information bottleneck problem and its applications in machine learning. *IEEE Journal on Selected Areas in Information Theory*, 1(1):19–38.

[5] Kowalski, M. H., Qian, H., Hou, Z., Rosen, J. D., Tapia, A. L., Shan, Y., Jain, D., Argos, M., Arnett, D. K., Avery, C., et al. (2019). Use of > 100,000 nhlbi trans-omics for precision medicine (topmed) consortium whole genome sequences improves imputation quality and detection of rare variant associations in admixed african and hispanic/latino populations. *PLoS genetics*, 15(12):e1008500.

[6] Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.

CHAPTER 2

BAYESIAN HIERARCHICAL HYPOTHESIS TESTING IN LARGE SCALE GENOME-WIDE ASSOCIATION ANALYSIS

2.1 Introduction

Many modern statistical learning problems require detecting from a large number of features a small fraction of them that are jointly associated with an outcome. In a regression set-up, a common approach to detect the important variables is variable selection. In recent years, many variable selection methods have been proposed (for reviews see e.g., [7], [23]). One can also pose a variable selection task as a (multiple) hypothesis testing problem. An advantage of the hypothesis testing framework is that one can adopt a decision-theoretic approach to find an optimal decision rule that minimizes some loss function in terms of Type-I and Type-II errors. However, designing rules that can attain high power with low error rates can be challenging [6].

Despite the important advancements in theory and methods for feature selection in high-dimension regression, variable selection, and inference in the presence of highly collinear features remain challenging. The problem of selecting a subset of predictors among a large set of correlated features is ubiquitous in Genome-Wide Association (GWA) studies where the objective is to map regions of the genome (either individual variants or chromosome segments) associated with a phenotype. In the last decade, several public (e.g., UK-Biobank, Million Veteran Program, TOPMed, All of Us) and private (e.g., 23andMe®) initiatives have generated unprecedentedly large biomedical datasets that comprise genotype data linked to extensive data on phenotype/disease. The advent of big data brings unprecedented opportunities to advance genetic research and predict complex traits ([5]). However, together with larger sample sizes, these modern data sets come with a remarkable increase in marker density, with potentially millions of single nucleotide polymorphisms (SNPs) distributed throughout the genome. With such a high SNP density, many SNPs can be highly correlated due to a phenomenon called linkage disequilibrium (LD).

Bayesian variable selection methods (BVS) ([10], [16]) can be used to identify risk variants in GWA studies. Following the seminal contribution of [21], these methods have been widely used

for the prediction of complex traits in plant and animal breeding (e.g., [21], [4], [14]) and also in human genetics (e.g., [3], [19], [13], [17], [18], [32]).

Bayesian methods offer adequate quantification of uncertainty in variable selection problems. This feature, which is essential for any inferential task, has some unwanted consequences when the goal is to select individual variants associated with a phenotype. For example, when multiple variants are all in LD with one or more causal variants, the posterior probability of non-null effect (aka posterior inclusion probabilities or PIPs) of individual SNPs may not achieve a high value for any individual locus because of the uncertainty about individual SNP effects introduced by collinearity.

Therefore, variant centered BVS may not map important risk loci if those risk loci are located in regions of high LD. This phenomenon has been reported and studied before ([11]) and the general recommendation is to avoid using marginal posterior probabilities and focus instead on credible set inferences, i.e., identifying sets of predictors that are jointly associated with an outcome with high level of credibility. The joint posterior distribution (or samples from it) of BVS contains all the information needed to identify such sets. However, we lack methods to identify credible sets from posterior samples efficiently.

Our main objective is to fill this gap by developing methods for multi-resolution Bayesian hypothesis testing that can lead to powerful inferences, with adequate error control, and fine-mapping resolution, even in the presence of highly collinear predictors. Our approach integrates ideas first proposed for frequentist hierarchical hypothesis testing ([34],[21], [1], [25], [27]) into a Bayesian framework that can offer powerful inferences with adequate error control.

Our study offers the following contributions: First, we develop algorithms for Bayesian Hierarchal Hypothesis Testing (BHHT) that offers high power, with adequate error control and fine mapping resolution. These algorithms a solution to the challenge of identifying credible sets from posterior samples. The algorithms presented control the FDR at one of three possible levels: (i) individual credible sets, (ii) sub-families, and (iii) entire credible set FDR. Second, we present analytical proofs that these methods adequately control the FDR (i.e., the expected proportion of

false discovery over conceptual repeated sampling). Third, using extensive simulations, we show that the proposed BHHT methodology offers higher power than inferences based on marginal posterior probability (i.e., variant-centered) methods and is competitive, and in some settings superior, than state-of-the-art credible-set methods such as SuSiE [32]. Fourth, we demonstrate the feasibility of applying the proposed methods with ultra-high-dimensional, large sample size, data by using BHHT to map variants associated with serum urate (SU) using data from the UK-Biobank (n 300,000) involving ultra-high-density SNP genotypes (15 million variants). Fifth, we provide open-source software that implements the methods described in the study using algorithms that scale to ultra-high-dimensional biobank-sample-size data.

2.2 Bayesian Hierarchical Hypothesis Testing

Consider a multiple regression model of the form,

$$y_i = \mu + x_{i1}\beta_1 + x_{i2}\beta_2 + ... + x_{ip}\beta_p + \epsilon_i \qquad (i = 1, ..., n) \tag{2.1}$$

where y_i is a phenotype of interest, $x_{i1}, ..., x_{ip}$, are omics features, $\beta_1, ..., \beta_p$ are individual feature effects, and ϵ_i is an error term for the i-th individual in the sample of size n. Bayesian Variable Selection models often use IID priors from the Spike-and-Slab family, which include a point of mass at zero (i.e., no-effect, $\beta_j = 0$) and a slab (non-zero effect) of the form

$$p(\beta_j) = 1(\beta_j = 0)(1 - \pi) + \pi p(\beta_j|\theta) \qquad (j = 1, ..., p) \tag{2.2}$$

Above, π is the prior probability of a non-zero effect, $p(\beta_j|\theta)$ is a density function (the 'slab') describing the distribution of non-null effects, and $1(\beta_j = 0) = \{1\ if\ \beta_j = 0;\ 0\ otherwise\}$ is an indicator function. In this study, we use the BGLR R-package [24] which implements various models from the spike slab family with Gaussian and scaled-t slabs. The software uses a flat prior for the intercept and allows treating hyper-parameters controlling model sparsity and shrinkage (the error variance, and the parameters indexing the prior distribution of effect $\{\pi, \theta\}$) as unknown and produces samples from the posterior distribution of effects which are marginal with respect to hyper-parameters.

The samples from the posterior distribution of BVS models can be used to estimate the (marginal) posterior probability of non-zero effects for each of the predictors: $P(\beta_j \neq 0|data) = \widehat{PIP}_j = K^{-1}\sum_{k=1}^{K} 1(\beta_{jk} \neq 0)$. These posterior probabilities can be used to identify features that are confidently associated with an outcome, e.g., those with $\widehat{PIP}_j \geq 0.9$ corresponding to a local-FDR ≤ 0.1. However, as noted, marginal inferences can be under-powered in the presence of collinearity. We will demonstrate this behavior in detail with a GWA study example in Sec 2.4.

2.2.1 Credible Set Inference

In the presence of collinearity, more powerful inferences can be obtained by making inferences about sets of parameters i.e., using credible set methods, e.g., [11], [32]. A level-α credible set, defined by Wang et al. (2020) [32], is a set of features whose joint probability of association is greater that α. The joint probability of association of a set is the probability that at least one of the features in the set is associated with the outcome. For a set of features, the set-PIP denoted by PIP_Ω of a set of features Ω can be defined as follows.

$$p(at\ least\ one\ \beta_j \neq 0\ where\ x_j \in \Omega|data) = 1 - p(all\ \beta_j = 0\ where\ x_j \in \Omega|data)$$

These set-PIPs can be estimated using samples from the posterior distribution by simply counting the proportion of posterior samples that contain at least one of the effects in the set different than zero, for a set $\Omega = \{j, j'\}$,

$$\widehat{PIP}_\Omega = 1 - K^{-1}\sum_{k=1}^{K}\prod_{j\in\Omega} 1(\hat{\beta}_{jk} = 0) \tag{2.3}$$

Above, $k = 1, ..., K$ is an index for posterior samples. However, as noted before, given α, we lack algorithms that can identify the credible sets from posterior samples, offering high power, with low error rate, and as fine-mapping precision (i.e., small CS) as allowed by the data. To fill this gap, we propose using Bayesian Hierarchical Hypothesis Testing (BHHT).

2.2.2 Bayesian Hierarchical Hypothesis Testing

Bayesian Hierarchical Hypothesis Testing (BHHT) tests a sequence of nested hypotheses associated with hierarchical clusters of the inputs $(x_{i1}, ..., x_{ip})$. The procedure is schematically described in Fig. 2.1 which depicts an example involving four features clustered according to binary tree.

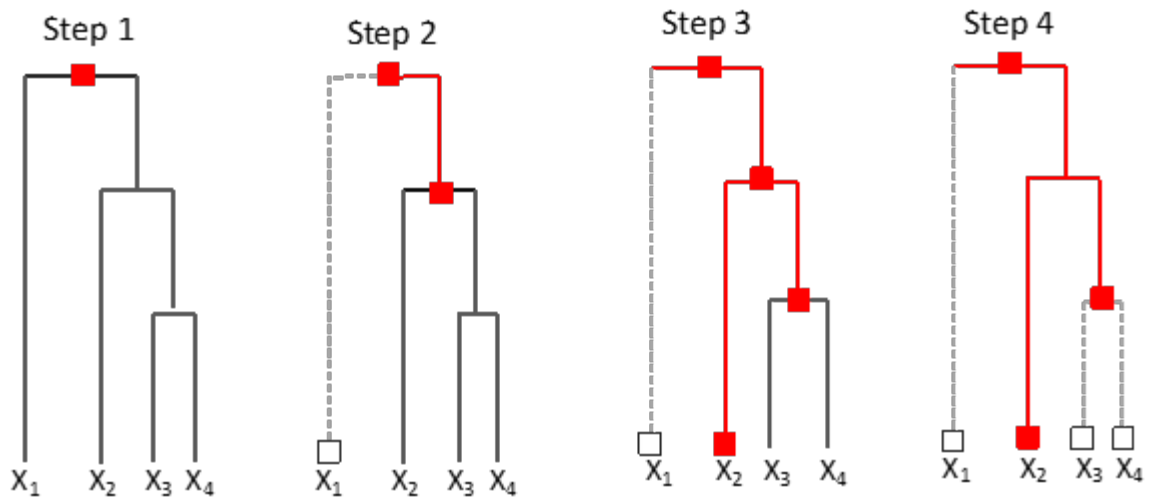


Figure 2.1 Schematic representation of Hierarchical Hypothesis Testing. The example involves 4 features ($X_1 - X_4$). Testing starts at the top node and proceeds to lower levels of the hierarchy, whenever a null hypothesis at the parent node is rejected Red (white) squares depict significant (non-significant) test results, the path involving significant results is highlighted in red.

Testing starts at the top node; every time a hypothesis is rejected, the two nested hypotheses involving the effects under the left and right branches (aka 'children') are tested. In Fig. 2.1, nodes with red (white) squares denote null hypotheses that were (were not) rejected. The path containing rejected nulls is marked in red. In the example in Fig. 2.1, testing involves four steps:

(i) $H_{(0(1))} : \beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$ Vs $H_{(a(1))}$: at least one coefficient different than 0. Decision: reject $H_{(0(1))}$.

(ii) $H_{(0(2L))} : \beta_1 = 0$ Vs $H_{(0(2L))} : \beta_1 \neq 0$, and $H_{(0(2R))} : \beta_2 = \beta_3 = \beta_4 = 0$ Vs $H_{(a(2R))}$: at least one of the three coefficients $\neq 0$. Decisions: Do not reject $H_{(0(2L))}$ and reject $H_{(0(2R))}$.

(iii) $H_{(0(3L))} : \beta_3 = 0$ Vs $H_{(0(3L))} : \beta_1 \neq 0$, and $H_{(0(3R))} : \beta_3 = \beta_4 = 0$ Vs $H_{(a(3R))}$: either β_3 or β_4 or both coefficients $\neq 0$. Decisions: reject both $H_{(0(3L))}$ and $H_{(0(3R))}$.

(iv) $H_{(0(4L))} : \beta_3 = 0$ Vs $H_{(0(4L))} : \beta_3 \neq 0$ and $H_{(0(4R))} : \beta_4 = 0$ Vs $H_{(0(4R))} : \beta_4 \neq 0$. Decision: no null rejected.

The final discovery set includes two elements X_2 (a singleton) and a duplet $\{X_3, X_4\}$.

2.2.3 Error Control

We focus on controlling the Bayesian FDR (BFDR); controlling BFDR at an α-level also provides FDR control (over conceptual repeated sampling) at the same level (see Lemma 1 and its proof in the Appendix). We consider decision rules that control the BFDR at three levels:

(i) **Local (or node) BFDR (LFDR)**: For each test, the local BFDR is the posterior probability of the null given the data: $LFDR_{\Omega} = 1 - PIP_{\Omega}$ where Ω is the set of all coefficients under that node. This quantity can be estimated for any hypothesis using expression (2.3). Thus, for LFDR control, in BHHT a null hypothesis at a node is rejected if the $LFDR_{\Omega}$ of the node is smaller than α. Note 1 of the Appendix provides a formal definition of the algorithm we use to traverse a tree, rejecting the null hypothesis provided that the $LFDR_{\Omega}$ is smaller than a tolerable threshold ($\alpha \in [0, 1]$, e.g., $\alpha = 0.05$). The algorithm renders a discovery set, $DS(\alpha) = \{\Omega_1, \Omega_2, ..., \Omega_D\}$, consisting of the smallest nodes (i.e., deepest in the tree) that have $LFDR_{\Omega} < \alpha$. This algorithm controls the posterior probability of Type-I error for each element of the discovery set.

The Bayesian FDR of the discovery set denoted by DS-BFDR is simply the average of the LFDRs of each of the elements of the discovery set. Formally for a discovery set $DS(\alpha) = \{\Omega_1, \Omega_2, ..., \Omega_D\}$, the DS-BFDR is defined as:

$$DS - BFDR = \frac{\sum_{d=1}^{D} p(all\ \beta_j = 0\ where\ x_j \in \Omega_d\ |data)}{D} \tag{2.4}$$

It follows that a decision rule that controls the LFDR for each of the elements of the DS yields DSs with a DS-BFDR $< \alpha$. However, it is known that controlling the LFDR's of each element in the DS can be conservative. Therefore, a more powerful decision rule can be obtained by controlling the discovery DS-BFDR directly.

(ii) **Discovery-Set BFDR**: To control the DS-BFDR in BHHT we propose an algorithm that conducts tests over a grid of values of thresholds for the LFDR's ($\tilde{\alpha}_1 < \tilde{\alpha}_2 < ...$). For each value in the grid, we determine the $DS(\alpha_j)$ and estimate the DS-BFDR using (3). If the estimated DS-BFDR is below the target threshold (e.g., $\alpha = 0.05$) we move to the next $\tilde{\alpha}$ in the grid, otherwise, if DS-BFDR $> \alpha$, we use $DS(\tilde{\alpha}_{(j-1)})$ as the final DS. To achieve computational efficiency, we set

our grid of values to the sorted LFDRs (see Note 2 of the Appendix for a formal description of the proposed algorithm).

(iii) **Subfamily-wise BFDR**: A decision rule that controls the DS-BFDR does not bound the LFDR in each cluster. Hence, there can be cases where clusters with low inclusion probabilities are included in a DS not based on their own merit but by benefiting from the fact that other clusters have very high inclusion probabilities. This behavior has been noted in multiple testing [28] where many true rejections can inflate the denominator of the FDR so that it allows a small number of false discoveries. Therefore, following [34], we consider a third procedure that controls the subfamily-wise FDR. In this setting, the subfamilies of a node are the set of hypotheses in a hierarchy that share the same parent. The sub-family FDR method rejects the null at a node, provided that the average local FDR of the sub-family is below the chosen FDR threshold. This approach is more conservative than a method controlling the DS-BFDR in (ii) and less conservative than those based on the node-wise BFDR in (i). Note 3 in the Appendix describes an algorithm to control sub-familywise BFDR.

2.2.4 Demonstration with a simplified example

To demonstrate the application of BHHT we present a simplified example involving mice genomic data from Wellcome Trust [31]. The data used in the example is available with the R-package BGLR [24] and consists of genotypes of 1814 mice. For our example, we select the first $p = 300$ SNPs from chromosome 1. We simulate a phenotype (y) under the linear regression model (1) with Gaussian iid error terms. Only four of the 300 SNPs had non-zero effects. We fixed the error variance to ensure that the four causal variants explained 10% variance of the phenotype (y). Subsequently, we used the BGLR R-package to fit a Bayesian linear regression model to the simulated data set with an independent point mass spike and a Gaussian slab prior (2) also known as "BayesC" for genomic prediction in [9]) on the regression coefficients. The scripts used to simulate and analyze the data are provided in the Supplementary Materials file.

Fig. 2.2 shows the posterior probability of non-zero effects (PIPs) of the SNPs, we observe that none of the SNPs reach the 0.95 PIP cut-off (dashed horizontal line, corresponding to a local

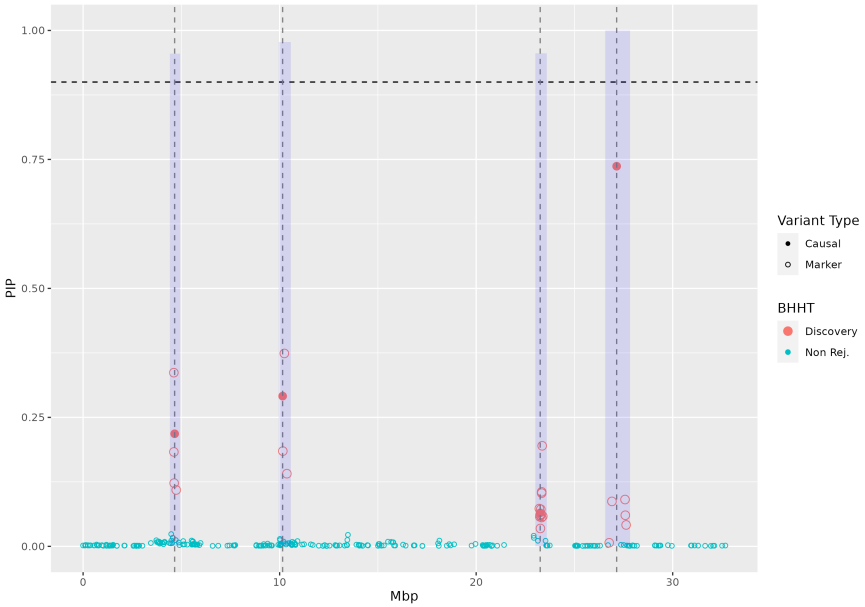


Figure 2.2 Posterior Probability of non-zero effect by SNP and Credible Set (CS) identification using Bayesian Hierarchical Hypothesis Testing (BHHT). Each point represents a SNP, the vertical axis gives the posterior probability of non-zero effect (PIP) and the horizontal axis is the map position in mega-base-pairs (Mbp). The purple vertical bars give the joint posterior probability of association of each of the CS identified through BHHT (see Table 2.1 below).

Features included	Set size	Set-PIP	Local BFDR
63, 64, 65, **66**, 68	5	0.96	0.05
125, 126, 129, 131	4	0.98	0.02
221, 223, 218, **219**, 220, 222, 216, 217, 224, 225, 226, 228, 227	13	0.96	0.04
270, 272, 271, 266, **267**, 265	6	1.00	0.00
Entire Discovery Set	**28**	**0.9721**	**0.0280**

Table 2.1 Four credible sets discovered through Bayesian Hierarchical Hypothesis Testing using the node BFDR criteria (variants with non-zero effect in the simulation are highlighted in bold).

BFDR of 0.05). The reason why no individual variant achieves high PIP is the high collinearity among variants nearby the ones with causal effects. We then apply the proposed BHHT algorithm that controls the DS-BFDR (Note 3) using $\alpha = 0.05$. BHHT with those parameters produced DS with four elements (Table 2.1), each with a high set-PIP (see vertical shaded area in Fig. 2.2 and Table 2.1). Each of the elements of the DS contained a causal variant, plus variants in high LD with it (Table 2.1).

2.3 Simulation Study

We conducted an extensive simulation study to evaluate the power-FDR performance of BHHT under scenarios that considered varying: (i) **degrees of collinearity** (moderate or high), (ii) **sample sizes** (n=10,000 and n=50,000), (iii) **model sparsity** (5 or 20 non-zero effects) and (iv) **mapping resolution** (the maximum cluster size allowed in the discovery set (we varied this from 1 to 10 features). We also used this simulation to benchmark BHHT against inferences based on marginal PIP and against SuSiE [15] – a state-of-the-art procedure for CS inference proposed for GWA studies.

For each Monte Carlo replicate we simulated 525 predictors using a Markov process such that the correlation between predictors was $Cor(x_{i(j)}, x_{i(j+k)}) = \rho^k$. Here, i is an index for the subject, j is an index for the feature in the sequence, and k is the lag-between predictors. We considered two scenarios for collinearity: $\rho = 0.45$ (moderate) and $\rho = 0.90$ (high). Then, we simulated a response y_i according to the linear model [1] with either 5 or 20 of the 525 predictors having non-zero effects ($S = 5$ or 20). The errors were *iid* Gaussian with zero mean and a variance tunned such that the proportion of variance of the outcome explained by the signal (aka heritability, h^2) was 0.0125. Further details about the simulation can be found in the Appendix.

We analyzed the simulated data using a Bayesian linear model with an independent spike-slab prior (described in Sec. 2) for the regression coefficients. Models were fitted using the BGLR function of the homonyms R-package BGLR [24]. Then, we clustered the features into a hierarchy using hierarchical clustering ([15], [33]) and applied BHHT (described in Sec. 2.2) with each of the three error control methods previously described (i.e., node-BFDR, DS-BFDR, and subfamily-wise BFDR). We considered two benchmarks for BHHT:

(i) Marginal hypothesis testing. In this method, we used marginal SNP-PIPs to produce discovery sets (see Note 4 of the Appendix for further details), and

(ii) SuSiE. a credible-set variable selection procedure Proposed in [32]. This method uses a prior distribution that leads to the formation of credible sets. We fitted SuSiE and obtained the credible sets using the R-package susieR [32].

We consider a credible set as a true discovery if it contains at least one causal variant (i.e., a variant with a non-zero effect in the simulation). However, to prevent favoring methods that yield large credible sets (i.e., poor mapping resolution) we conducted power-FDR evaluations restricting the maximum discovery set size to either 3, 5, or 10 SNPs.

2.3.1 Power-FDR performance

Fig. 2.3 shows the power-FDR performances of SNP-PIP, BHHT (using the DS-BFDR error control method), and SuSiE. The results in Fig. 2.3 were obtained by restricting the maximum discovery set size to 5 SNPs. When the number of causal effects was equal to 5 and collinearity was moderate (top-left panel in Fig. 2.3) we observed that all methods performed similarly, with the method based on marginal probabilities (SNP-PIP) performing only slightly worse in the scenario with n=10K. However, for the same trait architecture, when collinearity was high (ρ = 0.9 the bottom row of the left panel) the two credible set methods (DS-BFDR and SuSiE) outperformed marginal hypothesis testing (SNP-PIP) highlighting the power of credible set inference. In the scenario with only 5 causal variants and high collinearity, SuSiE performed slightly better than BHHT using the DS-BFDR method. However, when we considered a model with more causal variants (20 SNPs with non-zero effect, panel B in Fig. 2.3) the opposite happened, that is BHHT outperformed SuSiE. This may reflect a limitation of the variational algorithm in SuSiE which in multi-modal problems can be stuck at local optima (see the Discussion of [32]). Regardless of the scenario and method, as one would expect, for any FDR level, power was higher with the largest sample size.

The results in Fig. 2.3 were obtained by restricting the maximum credible size set to 5 variants. We also performed the same evaluation restricting the maximum credible set size to 3 or 10 variants. The results from these analyses are presented in Fig. B2.1 and Fig. B2.2 , respectively. The overall results were conceptually similar to the ones presented in Fig. 2.3 and previously discussed with a few differences: (i) As expected, imposing a more stringent restriction on the maximum cluster size reduced the power advantage of the CS inference methods relative to marginal SNP-PIP inferences (ii) When the maximum CS size was restricted to be only 3 SNPs, there were small differences

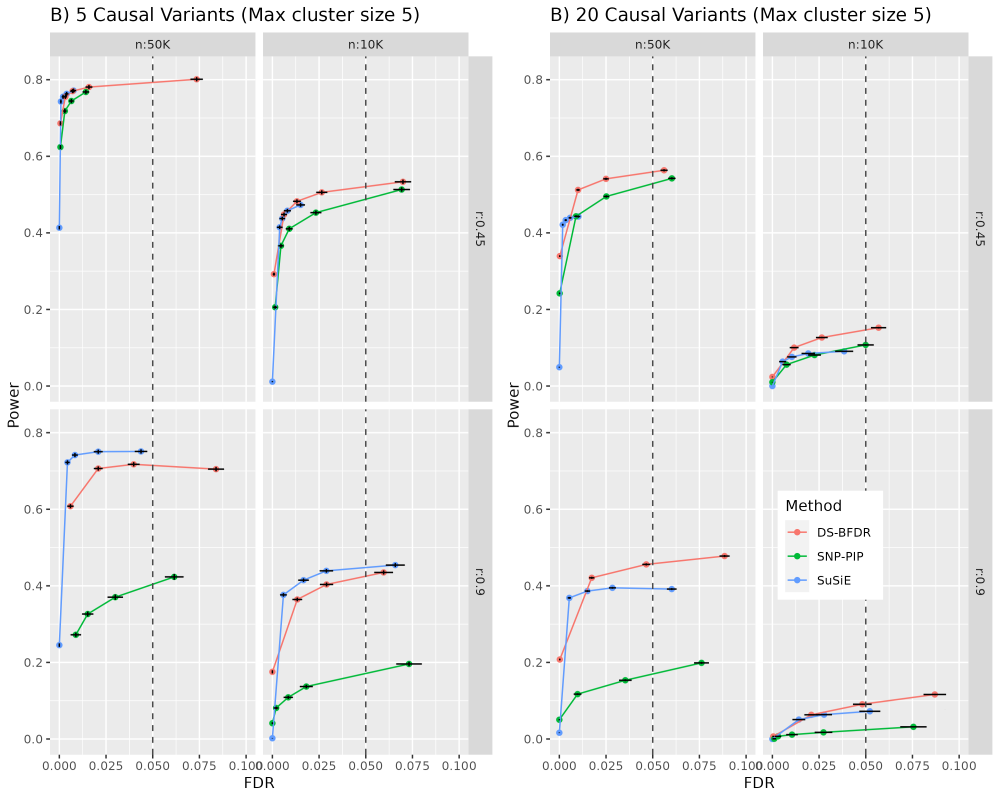


Figure 2.3 Power vs FDR curve for different simulation settings. n denotes the sample size and r denotes the correlation between adjacent SNPs. The Left and right panel represents scenarios where there are 5 and 20 causal variants respectively. We hold h^2 fixed at 1.25% for all settings. As collinearity increases the credible set inference gains significant power over individual SNP-level inference. Furthermore, the proposed method achieves higher power than SuSiE when there exists a greater number of causal variants in the true model.

between SuSiE and BHHT, and, finally, (iii) when the maximum CS size was allowed to be 10 SNPs, BHHT outperformed SuSiE by a sizable margin in the scenario with high LD and 20 causal variants. The results from BHHT previously discussed were based on the DS-BFDR error control procedure. We also evaluated BHHT using the node (local) BFDR and the subfamily-wise BFDR. Overall, there were no noticeable differences between the three error control methods (see Fig. B2.3 for results using the three methods for error control, for the same scenarios displayed in Fig. 2.3).

2.3.2 FDR control

In Sec. 3.1, we have observed that DS-BFDR can achieve high power at low FDR even in the scenario of high collinearity. Another important aspect of the method is how accurately it estimates the true FDR. In each panel of Fig. 2.4, we plot the empirical FDR vs the BFDR (estimated from

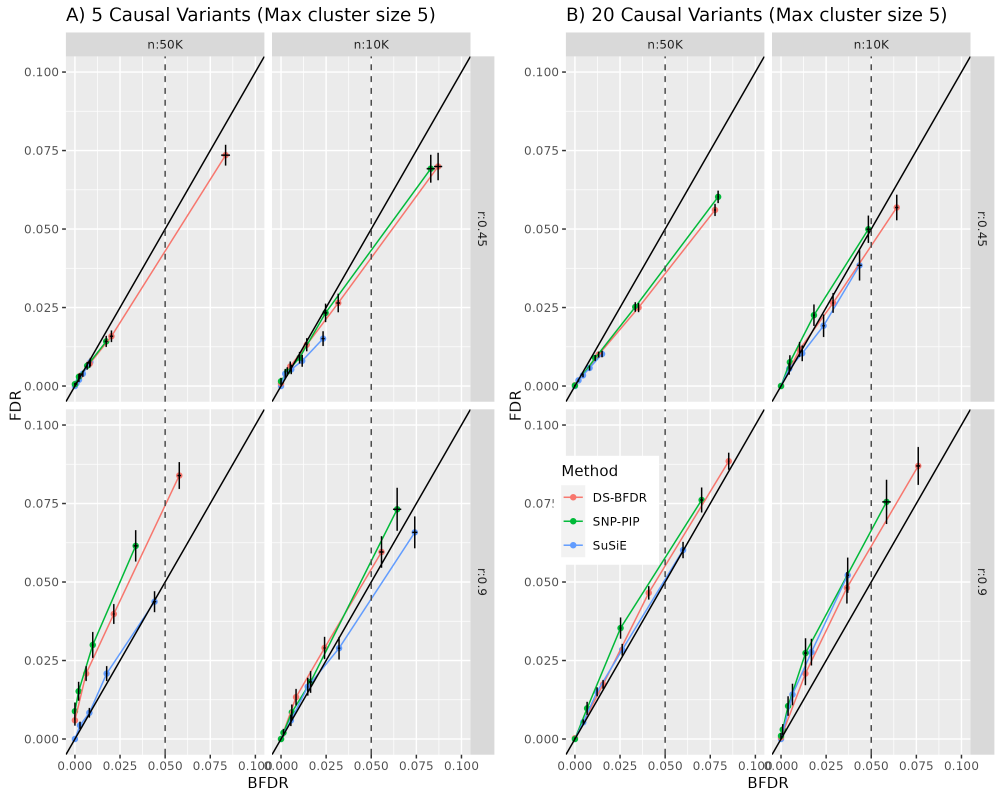


Figure 2.4 FDR vs Bayesian FDR plot for different simulation settings. n denotes the sample size and r denotes the correlation between adjacent SNPs. The Left and right panel represents scenarios where there are 5 and 20 causal variants respectively. We hold h^2 fixed at 1.25% for all settings. All the methods accurately control the FDR.

posterior samples) for the four simulation settings. For all settings, we observed that all methods accurately controlled the FDR at the desired level. Note that the different methods control different error rates. In SNP-PIP and DS-BFDR, we control the global error rate. Hence, we can see that, e.g., at $\alpha = 0.1$, the estimated BFDR for the two methods is much tighter to α (orange and green lines reach very close to 0.1) than SuSiE (blue line) for all settings.

2.4 Real Data Application

To demonstrate feasibility we applied the methods discussed in the previous section in a GWAS for serum urate using the UK Biobank dataset. The dataset contains genotype and phenotype information of $n \sim 300,000$ unrelated individuals of European ancestry. Hyperuricemia is a prevalent condition among adults in developed countries and it is a risk factor for gout [29]. Importantly, hyperuricemia has also been found to be associated with cardiovascular disease [2] and

many of the conditions that define metabolic syndrome. Furthermore, serum urate is moderately heritable (30-40% in [26]). Previous studies have reported many genes and genomic regions associated with serum urate; however, reported GWAS discoveries still explain only a fraction of the heritability of the trait.

We first log-transform the serum urate levels to make it symmetrically distributed. We then adjust it with respect to age, sex, center, and top 10-SNP derived principal components. We fitted models using (1) $n \sim 785,000$ SNPs from the SNP arrays used in the UK Biobank (hereinafter we refer to these sets as the "calls") and (2) $\sim$ 15 million SNPs from imputed genotypes provided by the UK-Biobank (the imputed SNPs were filtered to satisfy minor-allele-frequency ≥ 0.1 % and missing call rate ≤ 5 %). Due to the high density, the imputed genotypes have LD blocks with near-perfect collinearity; therefore the analyses involving the ultra-high density SNPs serve as an example of how the proposed methods can work under extreme collinearity.

It is computationally infeasible to fit a multiple regression model jointly with millions of SNPs. However, the unrelated white Europeans from UK Biobank have almost no population structure. Under these conditions, LD decays rather quickly with physical distance getting to values very close to zero at 500 kilobase pairs to 1 megabase pairs. Therefore, following [8] we conducted the association study by fitting local Bayesian regression models to overlapping segments throughout the genome. Details of the modeling strategy are discussed in Appendix.

We first selected individual SNPs with marginal hypothesis testing (Note 4) with BFDR $\leq$ 0.05. This approach yielded 123 discoveries in the analysis that used the genotyped SNPs, and 98 in the analysis that used the imputed SNPs (Table 2.2). The fact that fewer findings were obtained with the imputed SNPs, which include the genotyped SNPs as a subset, may seem counterintuitive. However, this happened because LD is much stronger in the imputed SNPs than in the calls. Under these conditions, due to high LD, many SNP that are in regions harboring causal variants do not reach high PIP. This problem can be overcome by identifying sets of SNPS with high joint-inclusion probability.

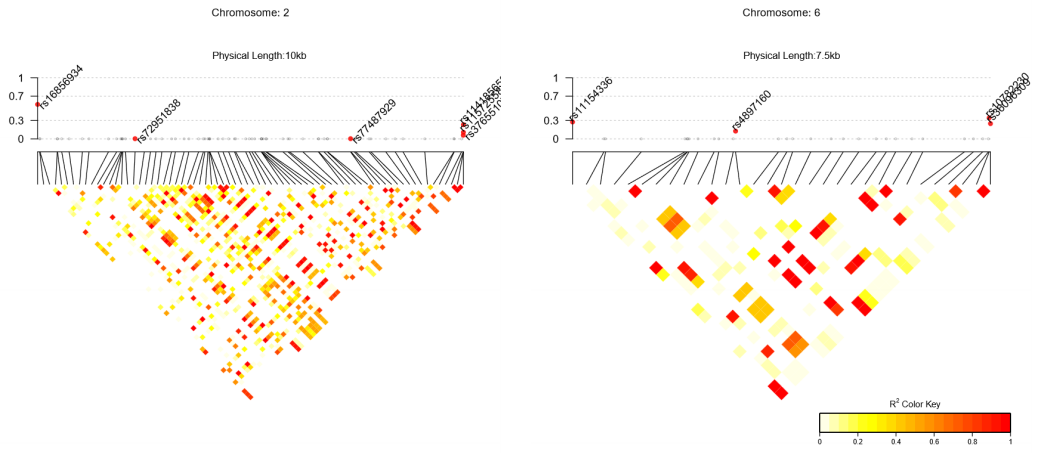
Therefore, we use the function segments() of the BGData R-package [12] to identify chromo-

Discoveries	**SNP-set**	
	Calls	**Imputed**
Singletons	128	98
Credible sets:		
Size $\in [1, 5)$	27	27
Size $\in [5, 10)$	4	11
Size $\in [10, 20)$	1	13
Size ≥ 20	11	31
Sub-Total	43	82
Total	**166**	**180**

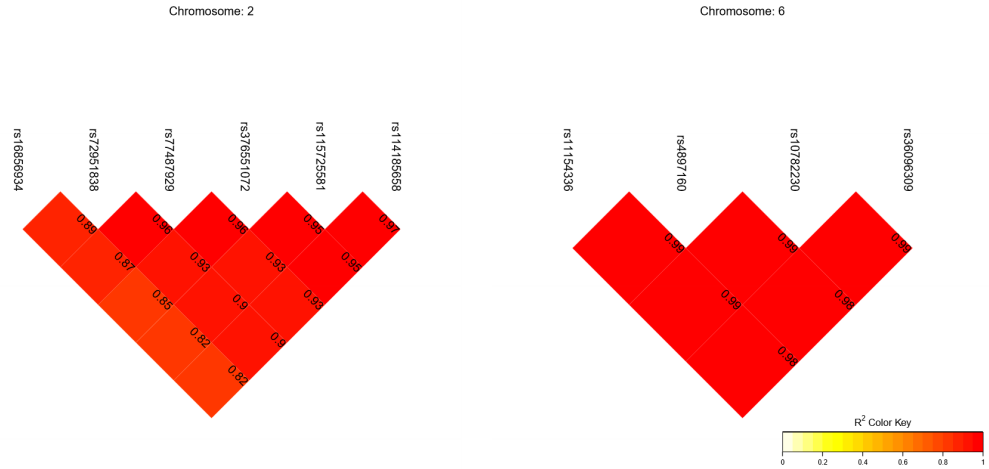
Table 2.2 Number of discoveries for serum urate by SNP-set used and type. The singletons and credible sets are from applying the marginal testing and BHHT algorithm respectively with $\alpha = 0.05$.

some segments harboring SNPs with elevated PIP. Specifically, we identify segments with SNPs that had PIP > 0.05 and were at a mutual distance smaller than 100 Kbp. Finally, we applied BHHT with DS-BFDR error control (Note 2) within each of these segments to identify credible sets. This approach led to the identification of 43 additional discoveries for the calls and 82 for the imputed genotypes. The physical positions of these segments are displayed for the imputed data set, together with the singletons, in Fig. B2.4 where singletons are represented by blue lines and segments are shown as bands on a yellow to red scale. For the calls, the use of CS inference increased the number of discoveries by 35%, and for the imputed genotypes, the increase in the number of discoveries was 83% (Table 2.2). These results highlight the importance of using credible set inferences when using high- and ultra-high-density genotypes. Most of the credible sets identified through BHHT were small (between 1 to 4 SNPs); however, some credible sets involved more than 20 SNPs (Table 2.2).

To provide further insight on how BHHT works in Fig. 2.5(a) we showcase the plot of the SNP-PIPs of two chromosome segments identified in the GWAS with the correlation heatmap added below the horizontal axis. The solid red points show the discovery set by applying the BHHT to the imputed SNP data. None of the five SNPs identified in the segment in chromosome 2 that is displayed in the left panel of Fig. 2.5(a) and none of the four SNPs forming the CS identified in



(a)



(b)

Figure 2.5 (a) Plot of the PIPs of the SNPs of two chromosome segments with the LD structure given below the horizontal axis. The solid red points indicate the discovery set of SNPs from BHHT with DS-BFDR control at $\alpha = 0.05$. (b) Correlation matrix between the discovery set SNPs of the segments depicted in panel (a).

chromosome 6 had high PIP; however, in both cases, jointly, they achieve high set-PIP. Fig. 2.5(b) shows the LD between the SNPs forming each of the sets. We observe that the procedure has managed to discover a small set of SNPs that are in very high LD with each other and jointly have high set-PIP.

2.5 Discussion

In this study, we propose a Bayesian hierarchical hypothesis testing (BHHT) approach for identifying credible sets in the context of GWAS with large datasets. In the not-too-distant future, the UK-Biobank (and many other large-scale projects) will release full genome sequences for almost half a million individuals. The availability of whole-genome sequence genotypes will bring unprecedented opportunities for fine mapping of disease risk alleles; at least in principle, fully sequenced genomes will enable the identification of causal variants without relying on LD between causal variants and those included in a genotyping array. However, whole-genome sequence-derived SNPs will be ultra-high-dimensional (tens of millions of SNPs), and variants within a short genome distance will be highly correlated, making fine mapping increasingly challenging. Analysis based on marginal association testing (e.g., single-SNP-phenotype association testing) will lead to many discoveries, but the vast majority of such discoveries will not be risk loci; instead, these will be SNPs in LD with risk variants.

To refine the results of marginal association tests, a common approach is to apply Bayesian variable selection methods to the SNP segments discovered using marginal association tests. When this approach is used, risk loci are often identified using PIPs. However, as we show in this study, when LD is extremely high, the use of individual SNP PIPs can lead to a loss of power. Furthermore, the issue gets aggravated when the SNPs are highly dense. This problem can be addressed using BHHT approach described in this study. BHHT can be used to identify individual SNPs with high PIP and credible sets, consisting of SNPs that are jointly confidently associated with the trait being studied. We have shown through simulations and real data analysis that these procedures can provide (1) high power with low FDR, (2) accurate error control, and (3) fine-mapping resolution. We further show that these approaches gain significant power over the tests for individual effects, which is the commonplace practice in GWAS.

In linear models, collinearity poses various challenges. Most popular variable selection procedures, e.g., lasso [30], elastic net [35], tend to produce a discovery set with many false discoveries when the features with non-zero effects are highly correlated with other features. Therefore, studies

in this area have changed the objective from selecting singletons to credible sets, which captures the uncertainty in selecting from a group of highly correlated features. In Bayesian variable selection, one approach (used in the SuSiE [32] method) is to specify a prior distribution that will lead to the identification of credible sets. Previous studies have shown that SuSiE outperforms the other procedures in the literature in terms of power, FDR, and size of credible sets (e.g., [32], [27]). The problem of variable selection under collinearity has been discussed in the frequentist literature. Recent methods such as knockoffzoom [27] applied hierarchical testing to find the credible sets. Although our approach is similar, the knockoffzoom procedure controls FDR at each resolution separately. Therefore, theoretically, it does not guarantee control over the discovery set FDR. In addition, it depends on the distribution of the covariates which for many applications is difficult to estimate. According to [27], knockoffzoom performs like SuSiE (Figure 4 in [27]) in terms of power, however, SuSiE achieved higher resolution discoveries.

A potential advantage of the BHHT presented in this study is that, unlike the case of SuSiE, BHHT can be used with any variable selection prior; thus, as shown in our simulations, BHHT can outperform SuSiE if the assumptions made by the SuSiE prior do not hold. On the other hand, BHHT can sometimes produce very large credible sets–this is particularly a problem when one uses very low FDR threshold (e.g., 0.01) which forces the method to include many SNPs in the set to meet such threshold. SuSiE does not have this limitation and hence in some cases it can discover smaller credible sets. For this reason, in Fig. 2.3 when S=5, and r=0.90 we see that SuSiE shows slight power gain over the multi-resolution tests.

We evaluated the BHHT with three FDR-control methods (local-BFDR, DS-BFDR and subfamily-wise BFDR). Overall, the simulation and real-data analysis showed that all criteria performed similarly. This may seem counterintuitive because a global FDR controlling procedure (Note 2) is likely to have more power than a procedure that controls local FDR (Note 1 and Note 3). One reason for such a behavior is the distribution of the set-PIP values. We have observed that across features these probabilities exhibit a U-shaped distribution, with a high frequency of set-PIPs very close to zero for clusters not harboring causal variants and set-PIPs very close to one for those

clusters harboring one or more causal variants. With such a distribution of the set-PIPs, the global and local control of the FDR leads to similar decision rules.

There are many interesting avenues that can lead to further research in BHHT and credible set inference. To address the issue of discovering large credible sets, these approaches can be extended by considering a penalty based on cluster size in the FDR calculation. One key challenge in multi-resolution inference is how to balance power-FDR and mapping resolution (i.e., credible set size). Here we followed [20] who used a simple approach consisting of comparing power-FDR performance constraining the credible set size to be fixed according to the hierarchy. However, there seems to be an interesting avenue of research focused on considering a constrained search problem with the multi-objective function based on maximizing power, minimizing type-I error, and maximizing mapping resolution simultaneously.

On the application side, the methods discussed in this study can be applicable to a wide range of fields beyond GWAS. An interesting application area is in the field of neuroscience, which focuses on identifying regions of the human brain related to neural disorders. Modern imaging techniques such as tractography provide data on the individual neuronal tracts in the human brain, which has a highly complex correlation structure. The BHHT can be applied to these complex datasets to better understand the relationship between tracts and the disorder. In general, BHHT are suited to the area of image analysis. Since the images have structures where nearby pixels tend to be strongly correlated, one can apply the multi-resolution tests to identify a group of pixels exhibiting some properties of interest. Likewise, we believe there may be interesting applications of BHHT for analysis of high-dimensional phenotype/risk factors, where in this method could help identifying sets of correlated risk factors that are jointly associated with an outcome. Finally, in a GWAS setting, there are some potentially interesting extensions of the method proposed in this study for mapping sets of loci that are simultaneously associated with more than one trait or disease (i.e., for the study of pleiotropy).

In summary, the BHHT approach proposed in this paper are shown to provide accurate and powerful inference in a wide range of scenarios. With the use of modern software and with access

to computing clusters such as those available at many universities and research institutions, this method can be applied to ultra-high dimensional data sets.

2.6 Software availability

The code for producing the results can be found in the Supplementary Materials file and at Github https://github.com/AnirbanSamaddar/Bayes_HHT.

BIBLIOGRAPHY

[1] Barber, R. F. and Ramdas, A. (2016). The p-filter: multi-layer fdr control for grouped hypotheses.

[2] Choi, H. K., De Vera, M. A., and Krishnan, E. (2008). Gout and the risk of type 2 diabetes among men with a high cardiovascular risk profile. *Rheumatology (Oxford)*, 47(10):1567–1570.

[3] de los Campos, G., Gianola, D., and Allison, D. B. (2010). Predicting genetic predisposition in humans: the promise of whole-genome markers. *Nat Rev Genet*, 11(12):880–886.

[4] de los Campos, G., Naya, H., Gianola, D., Crossa, J., Legarra, A., Manfredi, E., Weigel, K., and Cotes, J. M. (2009). Predicting quantitative traits with regression models for dense molecular markers and pedigree. *Genetics*, 182(1):375–385.

[5] de los Campos, G., Vazquez, A. I., Hsu, S., and Lello, L. (2018). Complex-trait prediction in the era of big data. *Trends in Genetics*, 34(10):746–754.

[6] Efron, B. (2010). *Large-Scale Inference: Empirical Bayes Methods for Estimation, Testing, and Prediction*. Institute of Mathematical Statistics Monographs. Cambridge University Press.

[7] Fan, J. and Lv, J. (2010). A selective overview of variable selection in high dimensional feature space. *Statistica Sinica*, 20(1):101–148.

[8] Funkhouser, S. A., Vazquez, A. I., Steibel, J. P., Ernst, C. W., and Los Campos, G. d. (2020). Deciphering sex-specific genetic architectures using local bayesian regressions. *Genetics*, 215(1):231–241. 32198180[pmid].

[9] Genovese, C. and Wasserman, L. (2004). Bayesian Frequentist Multiple Testing.

[10] George, E. I. and McCulloch, R. E. (1997). Approaches for bayesian variable selection. *Statistica Sinica*, 7(2):339–373.

[11] Ghosh, J. and Ghattas, A. E. (2015). Bayesian variable selection under collinearity. *The American Statistician*, 69(3):165–173.

[12] Grueneberg, A. and de los Campos, G. (2019). Bgdata - a suite of r packages for genomic analysis with big data. *G3: Genes, Genomes, Genetics*, 9(5):1377–1383.

[13] Guan, Y. and Stephens, M. (2011). Bayesian variable selection regression for genome-wide association studies and other large-scale problems. *The Annals of Applied Statistics*, 5(3):1780–1815.

[14] Habier, D., Fernando, R. L., Kizilkaya, K., and Garrick, D. J. (2011). Extension of the bayesian alphabet for genomic selection. *BMC Bioinformatics*, 12(1):186.

[15] Hartigan, J. A. (1975). *Clustering Algorithms*. John Wiley & Sons, Inc., USA, 99th edition.

[16] Ishwaran, H. and Rao, J. S. (2005). Spike and slab variable selection: Frequentist and bayesian strategies. *Ann. Statist.*, 33(2):730–773.

[17] Kim, H., Grueneberg, A., Vazquez, A. I., Hsu, S., and de los Campos, G. (2017). Will big data close the missing heritability gap? *Genetics*, 207(3):1135–1145.

[18] Lee, Y., Luca, F., Pique-Regi, R., and Wen, X. (2018). Bayesian multi-snp genetic association analysis: Control of fdr and use of summary statistics. *bioRxiv*.

[19] Makowsky, R., Pajewski, N. M., Klimentidis, Y. C., Vazquez, A. I., Duarte, C. W., Allison, D. B., and de los Campos, G. (2011). Beyond missing heritability: prediction of complex traits. *PLoS Genet*, 7(4):e1002051.

[20] Meinshausen, N. (2008). Hierarchical testing of variable importance. *Biometrika*, 95(2):265–278.

[21] Meuwissen, T. H. E., Hayes, B. J., and Goddard, M. E. (2001). Prediction of Total Genetic Value Using Genome-Wide Dense Marker Maps. *Genetics*, 157(4):1819–1829.

[22] Müller, P., Parmigiani, G., and Rice, K. (2006). Fdr and bayesian multiple comparisons rules. *Johns Hopkins University, Dept. of Biostatistics Working Papers. Working Paper 115*.

[23] O'Hara, R. B. and Sillanpää, M. J. (2009). A review of Bayesian variable selection methods: what, how and which. *Bayesian Analysis*, 4(1):85 – 117.

[24] Pérez, P. and de los Campos, G. (2014). Genome-Wide Regression and Prediction with the BGLR Statistical Package. *Genetics*, 198(2):483–495.

[25] Renaux, C., Buzdugan, L., Kalisch, M., and Bühlmann, P. (2020). Hierarchical inference for genome-wide association studies: a view on methodology with software. *Computational Statistics*, 35(1):1–40.

[26] Reynolds, R. J., Irvin, M. R., Bridges, S. L., Kim, H., Merriman, T. R., Arnett, D. K., Singh, J. A., Sumpter, N. A., Lupi, A. S., and Vazquez, A. I. (2021). Genetic correlations between traits associated with hyperuricemia, gout, and comorbidities. *European Journal of Human Genetics*, 29(9):1438–1445.

[27] Sesia, M., Katsevich, E., Bates, S., Candès, E., and Sabatti, C. (2020). Multi-resolution localization of causal variants across the genome. *Nature Communications*, 11(1):1093.

[28] Siegmund, D. O., Zhang, N. R., and Yakir, B. (2011). False discovery rate for scanning statistics. *Biometrika*, 98(4):979–985.

[29] Sun, M., Vazquez, A. I., Reynolds, R. J., Singh, J. A., Reeves, M., Merriman, T. R., Gaffo, A. L., and Los Campos, G. d. (2018). Untangling the complex relationships between incident gout risk, serum urate, and its comorbidities. *Arthritis Res Ther*, 20(1):90.

[30] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

[31] Valdar, W., Solberg, L. C., Gauguier, D., Cookson, W. O., Rawlins, J. N. P., Mott, R., and Flint, J. (2006). Genetic and Environmental Effects on Complex Traits in Mice. *Genetics*, 174(2):959–984.

[32] Wang, G., Sarkar, A., Carbonetto, P., and Stephens, M. (2020). A simple new approach to variable selection in regression, with application to genetic fine mapping. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(5):1273–1300.

[33] Ward Jr., J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244.

[34] Yekutieli, D. (2008). Hierarchical false discovery rate–controlling methodology. *Journal of the American Statistical Association*, 103(481):309–316.

[35] Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320.

APPENDIX A2

SUPPLEMENTARY METHODS

A2.1 Set up for Lemma 1:

Consider the multiple regression problem in (1). We are interested in testing a set of k null hypotheses about the regression coefficients denoted by $H_{01}, ..., H_{0K}$. These hypotheses can be simple for example $\beta_{0j} : \beta_j = 0$ vs $H_{1j} : \beta_j \neq 0, \forall j = 1, ..., p$ or can be composite for example $H_{0j} : \beta_j = \beta_{j'} = 0$ vs H_{1j}: at least one of β_j or β'_j is not zero $\forall j \neq j' = 1, ..., p$.

Let, d_j denote the decision rule for testing H_{0j} where $d_j = 1$ implies a rejection of H_{0j} and $d_j = 0$ implies otherwise. The FDR is defined by,

$$FDR = \mathbb{E}_{p(y)} \left(\frac{\text{\# of true null hypotheses rejected by } d_j \forall j = 1, ..., K}{\text{\# of null hypotheses rejected by } d_j \forall j = 1, ..., K} \right)$$

Note that the expectation is with respect to the marginal density p(y) which is not tractable. Therefore, following [9], [22], we use the Bayesian FDR or BFDR to estimate the FDR. BFDR is defined as,

$$BFDR = \frac{\sum_{j=1}^{K} p(H_{0j}|data) d_j}{\sum_{j=1}^{K} d_j}$$

In the above definition, data represents the training data available that is $\{y_i, x_{ij} : i = 1, ..., n; j = 1, ..., p\}$. By the below lemma, we show that the BFDR is an unbiased estimate of the FDR (over conceptual repeated sampling from $p(y)$). Therefore, by controlling the BFDR we control the FDR at a desired level. The proof is a straightforward application of the law of the iterated expectation.

Lemma A2.1.1. *Under the above set up,* $FDR = \mathbb{E}_{p(y)}(BFDR)$

Proof. From the definition,

$$
\begin{aligned}
FDR &= \mathbb{E}_{p(y)}\left(\frac{\#\text{ of true null hypotheses rejected by } d_j \forall j = 1, ..., K}{\#\text{ of null hypotheses rejected by } d_j \forall j = 1, ..., K}\right) \\
&= \mathbb{E}_{p(y)}\left(\frac{\sum_{j=1}^{K} \mathbb{1}(H_{0j} \text{ is true}) d_j}{\sum_{j=1}^{K} d_j}\right) \\
&= \mathbb{E}_{p(y)}\left(E_{p(\beta_1,...,\beta_p|data)}\left(\frac{\sum_{j=1}^{K} \mathbb{1}(H_{0j} \text{ is true}) d_j}{\sum_{j=1}^{K} d_j}\right)\right) \\
&= \mathbb{E}_{p(y)}\left(\frac{\sum_{j=1}^{K} p(H_{0j}|data)) d_j}{\sum_{j=1}^{K} d_j}\right) \\
&= \mathbb{E}_{p(y)}(BFDR)
\end{aligned}
$$

The third equality is due to the law of iterated expectation. The fourth equality is valid since $d_j \forall j = 1, ..., K$ are functions of the training data and therefore we can apply the conditional expectation only to the indicators $\mathbb{1}(H_{0j} \text{ is true}) \forall j = 1, ..., K$ in the numerator. Hence the proof. □

A2.2 Set up for the algorithms:

Let, m hierarchical hypotheses in a tree be denoted by $(H_{01}, H_{11}), ..., (H_{0m}, H_{1m})$, where (H_{0j}, H_{1j}) denotes the null (i.e., none of the predictors in the cluster has an effect) and the alternative (at least one of the predictors belonging to the node has an effect) in the j-th cluster. Following Sec 2.1, the posterior samples can be used to estimate the set-PIPs at each cluster denoted by, $v_j = p(H_{1j}|data) \forall j = 1, ..., m$.

Note 1 describes the local BFDR control algorithm.

Algorithm A2.1 Local BFDR control

1: Arrange v_j's in descending order. Let the ordered set-PIPs be, $v_{(m)} \geq v_{(m-1)} \geq \cdots \geq v_{(1)}$;
2: Given threshold $\alpha \in (0, 1)$, start from $j = m$;
3: **while** $(1 - v_{(j)}) \leq \alpha$ **do**
4: Reject null hypothesis corresponding to v_j and set $j \leftarrow j - 1$;
5: **end while**
6: **return** Discovery set including rejected hypotheses where no further refinement is possible.

Note 2 describes the DS-BFDR control algorithm.

Algorithm A2.2 DS-BFDR control

1: Arrange v_j's in descending order. Let the ordered set-PIPs be, $v_{(m)} \geq v_{(m-1)} \geq \cdots \geq v_{(1)}$;
2: Given threshold $\alpha \in (0, 1)$, start from $j = m$;
3: Using Algorithm A2.4 with threshold $v_{(j)}$ obtain discovery set $DS(v_{(j)})$ and $DS{-}BFDR(v_{(j)})$;
4: **while** $DS - BFDR(v_{(j)}) \leq \alpha$ **do**
5: Using Algorithm A2.4 with threshold $v_{(j)}$ obtain discovery set $DS(v_{(j)})$ and $DS - BFDR(v_{(j)})$;
6: Set $j \leftarrow j - 1$;
7: **end while**
8: **return** Discovery set $DS(v_{(j-1)})$.

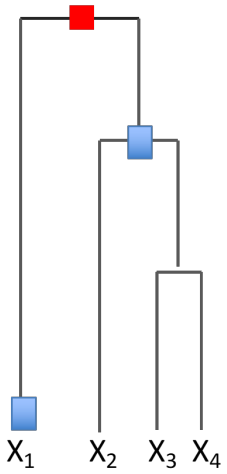


Figure A2.1 Demonstrating subfamily

Note 3 describes the subfamily-wise FDR control algorithm. In a hierarchy, a subfamily is defined by the clusters which share the same parent. For example, in Fig. A2.1, the parent node hypothesis is $H_0 : \beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$ Vs H_0: at least one coeff. $\neq 0$ which is marked by solid red squares. The subfamily of the parent node is marked by solid blue rectangles. The subfamily-wise FDR is defined as the FDR in the discovery set within a subfamily.

Algorithm A2.3 Subfamily-wise BFDR control

1: Given threshold $\alpha \in (0, 1)$, start from the top node of the hierarchy;
2: Apply Algorithm 4 with threshold α to each subfamily to control subfamily-wise BFDR;
3: **return** Discovery set including rejected hypotheses where there is no rejection in their respective subfamilies.

Note 4 Consider the multiple regression problem in (1). We are interested in testing a set of p marginal null hypotheses denoted by $H_{01}, ..., H_{0p}$ where $H_{0j} : \beta_j = 0$ vs $H_{1j} : \beta_j \neq 0, \forall j = 1, ..., p$.

Given the PIPs denoted by $v_j \forall j = 1, ..., p$ and a threshold $\alpha \in (0, 1)$, we outline the below steps for marginal hypothesis testing with BFDR controlled at a desired level:

Algorithm A2.4 Marginal hypothesis testing

1: Arrange v_j's in a descending order. Let the ordered set-PIPs be, $v_{(p)} \geq v_{(p-1)} \geq \cdots \geq v_{(1)}$;
2: Given threshold $\alpha \in (0, 1)$, start from $j = p$;
3: Obtain the discovery set $DS(v_{(j)})$ consisting hypotheses corresponding to j largest v_j values and calculate $BFDR(v_{(j)})$;
4: **while** $BFDR(v_{(j)}) \leq \alpha$ **do**
5: Obtain the discovery set $DS(v_{(j)})$ consisting hypotheses corresponding to j largest v_j values and calculate $BFDR(v_{(j)})$;
6: Set $j \leftarrow j - 1$;
7: **end while**
8: **return** Discovery set $DS(v_{(j-1)})$.

APPENDIX B2

SUPPLEMENTARY DATA

B2.1 Generation of the simulation study covariates

Here, we provide details on the generation of the covariates $\{x_{ij} : i = 1, ..., n; j = 1, ..., p\}$ for the simulation study. As mentioned in the Simulation study section, we have generated covariates where $Cor(x_{i(j)}, x_{i(j+k)}) = \rho^k$ decays with distance k. Here, i is an index for the subject, j is an index for the feature in the sequence, and k is the lag-between predictors. To generate the columns of the covariate matrix in this way we followed the below two steps,

1. Generate the first column of the covariate matrix with the n elements from the i.i.d. $Binomial(2, a)$. We set $a = 0.2$ in all simulations.
2. Generate the j-th column by randomly permuting a random fraction (f) of elements of (j-1)-th column. The random fraction $f \sim beta(a, b)$.

By changing the values of (a, b), we simulate different degrees of correlation between adjacent columns. For example, setting $(a, b) = (2, 3)$ implies correlation ~ 0.45 and $(a, b) = (0.5, 3)$ implies correlation ~ 0.9. Note that, as the distance between two columns grows, the correlation decreases as more elements change positions randomly due to permutation.

B2.2 Model hyperparameters

In this section, we provide details of the hyperparameters used in both the simulation study and the real data application. In this study, we have fitted two Bayesian methods to the data, which are independent spike-and-slab regression and SuSiE. We have used the respective R-packages BGLR [24] and susieR [32] to fit the models. Below are the models and key hyperparameter values used.

Independent spike-and-slab model: The Bayesian spike-and-slab model (aka BayesC) puts a

Hyperparameters	Simulation study		Real data application
	S = 5	S = 20	
$\frac{a}{a+b}$	$\frac{10}{p}$	$\frac{30}{p}$	$\frac{1.01}{1000}$
$a + b$	110	110	1000
# chains	4	4	4
iterations per chain	15000	15000	55000
burn-in	2500	2500	5000

Table B2.1 Table showing hyperparameter values for simulation study and real data application for the spike-and-slab model.

point mass and a gaussian slab on the regression coefficients. The model hierarchy is as follows,

$$
\begin{aligned}
y_i &= \mu + x_{i1}\beta_1 + x_{i2}\beta_2 + ... + x_{ip}\beta_p + \epsilon_i \qquad (i = 1, ..., n) \\
\epsilon_i &\sim N(0, \sigma^2_\epsilon) \\
p(\beta_j) &= 1(\beta_j = 0)(1 - \pi) + \pi p(\beta_j|\theta) \qquad (j = 1, ..., p) \\
\pi &\sim beta(a, b) \\
\sigma^2_b &\sim \chi^{-2}(df_b, S_b) \\
\sigma^2_\epsilon &\sim \chi^{-2}(df_\epsilon, S_\epsilon)
\end{aligned}
$$

Here, χ^{-2} denotes the scaled-inverse chi-square distribution. Also, since we used the Gibbs sampler to draw samples from the posterior, a few other important hyperparameters are the number of chains, number of iterations, and number of burn-in steps. We set the hyperparameters in Table B4.1.

Note that the only hyperparameter changed between the two simulation settings, $S = 5$and $S = 10$, is the prior probability of inclusion $a/(a+b)$. In our simulations, if we hold this to be fixed at $10/p$ for $S = 20$ the power of the BHHT method drops marginally. However, the key reason to change this hyperparameter with S is to be consistent with SuSiE (described below). For the other parameters, we set them to the default values of the software implementation.

The sum of Single Effects model (SuSiE): The SuSiE model apriori assumes that out of the p

coefficients L of them are non-zero. The model hierarchy is as follows,

$$
\begin{aligned}
y_i &= \mu + x_{i1}\beta_1 + x_{i2}\beta_2 + \ldots + x_{ip}\beta_p + \epsilon_i \qquad (i = 1, \ldots, n) \\
\epsilon_i &\sim N(0, \sigma^2_\epsilon) \\
\boldsymbol{\beta} &= \sum_{\ell=1}^{L} \boldsymbol{\beta}_\ell \\
\boldsymbol{\beta}_\ell &= \boldsymbol{\gamma}_\ell \beta_\ell \\
\boldsymbol{\gamma}_\ell &\sim Mult(1, \pi) \\
\beta_\ell &\sim N(0, \sigma^2_{0\ell})
\end{aligned}
$$

In the above set of equations, $\boldsymbol{\beta}$ represents the $p \times 1$ dimensional vector of regression coefficients. The hyperparameter L here has a similar interpretation as the prior probability of inclusion in the spike-and-slab model. However, the misspecification of L has a significant impact on the power of SuSiE. Especially when$L << S$, the SuSiE method tends to lose significant power. The general advice in [32] is to choose L to be higher than S. Therefore, for the simulation settings where $S = 5$ and $S = 10$ we fix $L = 10$ and $L = 30$, respectively, to be consistent with the spike-and-slab prior. The other hyperparameters in the model are set to the default values of the software implementation.

B2.3 Real data modeling strategy

In the main paper we described the application of BHHT in a GWAS of the trait serum urate. In this section, we discuss additional details used to fit a Bayesian linear regression model using the calls ($\sim$ 785,000 SNPs) and imputed ($\sim$ 15 million SNPs) data sets.

It is computationally infeasible to fit a multiple linear regression jointly with all the SNPs. Therefore, following [8] we conducted the association study by fitting local Bayesian regression models to overlapping segments throughout the genome. For calls, we created segments of 2900 SNPs by taking disjoint cores of 1500 SNPs and adding overlapping flanking regions of 700 SNPs on both sides. Similarly for imputed data, we considered disjoint cores of 7000 SNPs and overlapping flanking regions of 1000 SNPs on both sides. From the local regression fit, we only kept samples of the effects of the SNPs from the core region. We collected samples from the

posterior distribution of the model using the BLRXy() function of the BGLR R-package [24]. This function implements the same set of models implemented in BGLR but performs the computation using sufficient statistics $(X'X, X'y, y'y)$; where X is the design matrix and y is the response vector. Following this strategy offers great computational advantages when $n >> p$.

B2.4 Supplementary figures

Below are the supplementary figures that are referred to in the main paper.

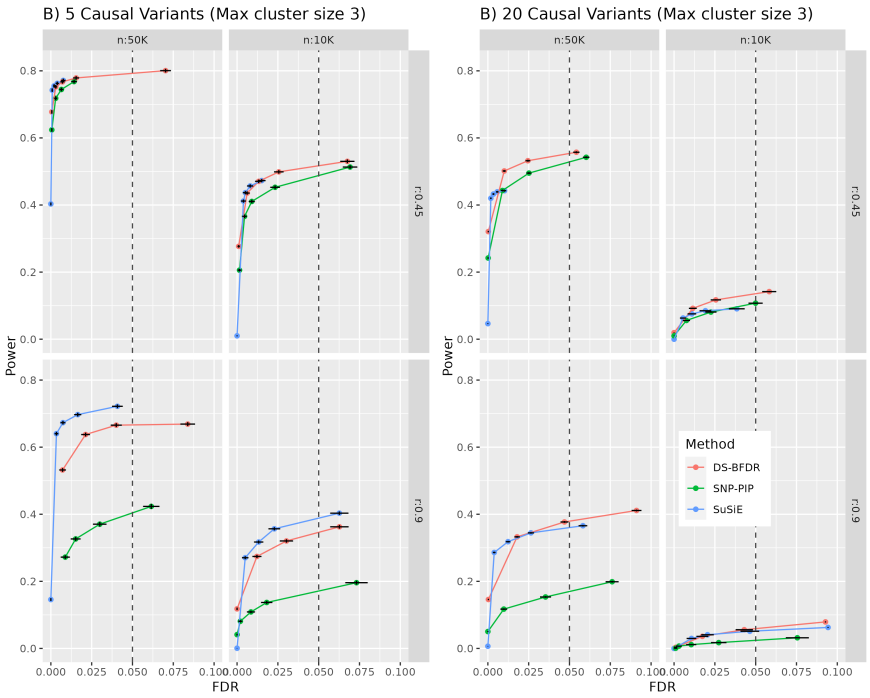


Figure B2.1 Power vs FDR curve for different simulation settings restricting maximum cluster size to 3. n denotes the sample size and r denotes the correlation between adjacent SNPs. The Left and right panel represents scenarios where there are 5 and 20 causal variants respectively. We hold h^2 fixed at 1.25% for all settings. Putting more restriction on the cluster size criteria have reduced the power advantage of CS approaches relative to marginal SNP-PIP inference.

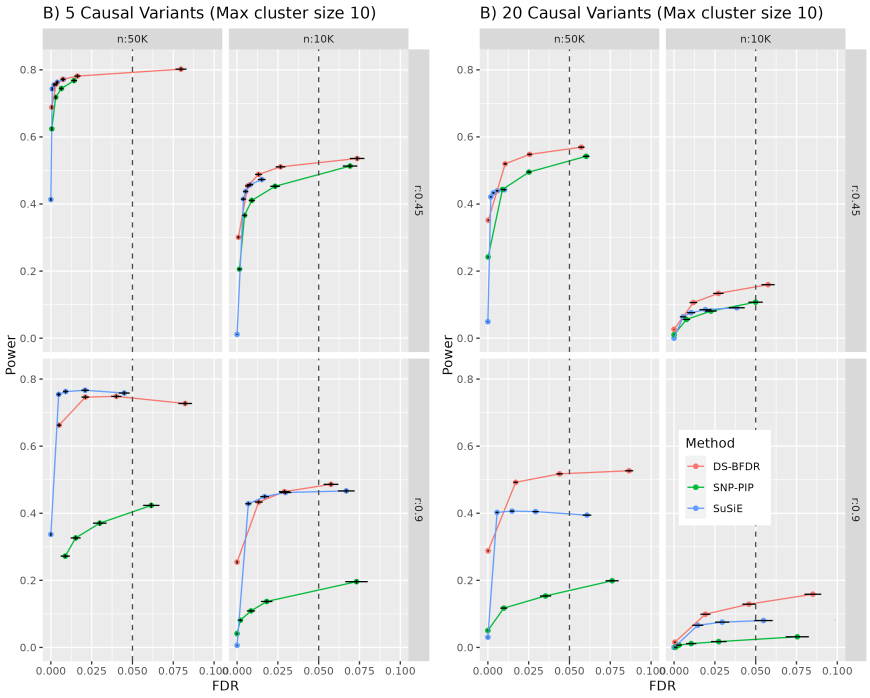


Figure B2.2 Power vs FDR curve for different simulation settings restricting maximum cluster size to 10. n denotes the sample size and r denotes the correlation between adjacent SNPs. The Left and right panel represents scenarios where there are 5 and 20 causal variants respectively. We hold h^2 fixed at 1.25% for all settings. Putting less restriction on the cluster size criteria has increased the power advantage of CS approaches relative to marginal SNP-PIP inference. Furthermore, the DS-BFDR approach gains more power due to lower-resolution discoveries relative to SuSiE.

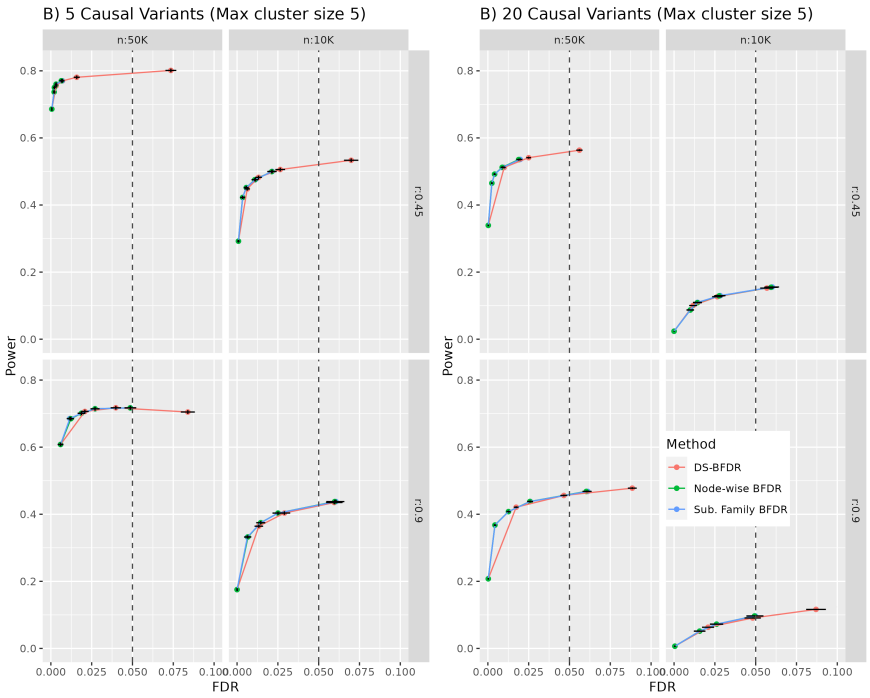


Figure B2.3 Power vs FDR curve of the three error controlling approaches for different simulation settings. n denotes the sample size and r denotes the correlation between adjacent SNPs. The Left and right panel represents scenarios where there are 5 and 20 causal variants respectively. We hold h^2 fixed at 1.25% for all settings. All the error-controlling approaches show similar performance.

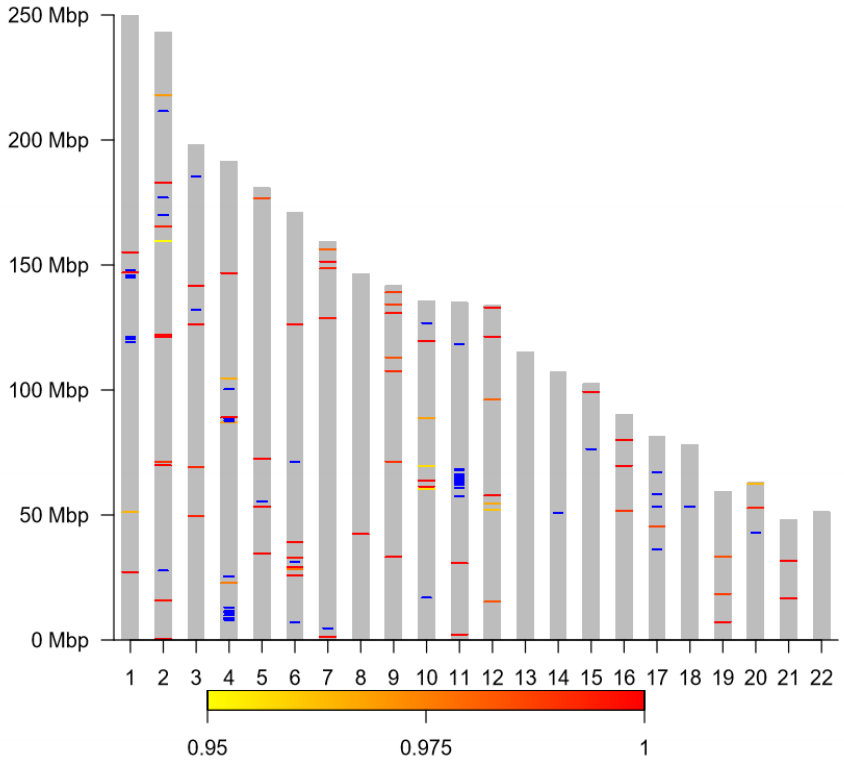


Figure B2.4 Ideogram displaying each of the 22 autosomal chromosomes for the imputed data set and singleton and segment discovered in the GWAS of the trait serum urate. The individual SNPs that cleared the 0.05 BFDR threshold are represented using blue lines along with the segments identified which are represented by bars colored as per their joint probability of inclusion in a yellow-to-red scale.

CHAPTER 3

SPARSITY-INDUCING CATEGORICAL PRIOR IMPROVES ROBUSTNESS OF THE INFORMATION BOTTLENECK

3.1 Introduction

Information bottleneck (IB) ([28]) is a deep latent variable model that poses representation compression as a constrained optimization problem to find representations Z that are maximally informative about the outputs Y while being maximally compressive about the inputs X, using a loss function expressed using a mutual information (MI) metric and a Lagrangian formulation of the constrained optimization: $\mathcal{L}_{IB} = \mathrm{MI}(X;Z) - \beta\,\mathrm{MI}(Z;Y)$. Here, $\mathrm{MI}(X;Z)$ is the MI that reflects how much the representation compresses X, and $\mathrm{MI}(Z;Y)$ reflects how much information the representation has kept from Y.

It is standard practice to use parametric priors, such as a mean-field Gaussian prior for the latent variable Z, as seen with most latent-variable models in the literature ([30]). In general, however, a major limitation of these priors is the requirement to preselect a latent space complexity for all data, which can be very restrictive and lead to models that are less robust. Sparsity, when used as a mechanism to choose the complexity of the model in a flexible and data-driven fashion, has the potential to improve the robustness of machine learning systems without loss of accuracy, especially when dealing with high-dimensional data ([1]).

Sparsity has been considered in the context of latent variable models in a handful of works. In linear latent variable modeling, [17] proposes a sparse factor model in the context of learning analytics, [4] proposes a sparse partial least squares (sPLS) method to resolve the inconsistency issue that arises in standard PLS in a high-dimensional setting, and [34] reviews advances in sparse canonical correlation analysis. Most of the work in sparse linear latent variable modeling fixes the latent dimensionality or treats it as a hyperparameter. In nonlinear latent variable modeling, sparsity was proposed primarily in the context of unsupervised learning. Notable works include sparse Dirichlet variational autoencoder (sDVAE) ([3]), epitome VAE (eVAE) ([35]), variational sparse coding (VSC) ([31]), and InteL-VAE ([21]). Most of these approaches do not take into

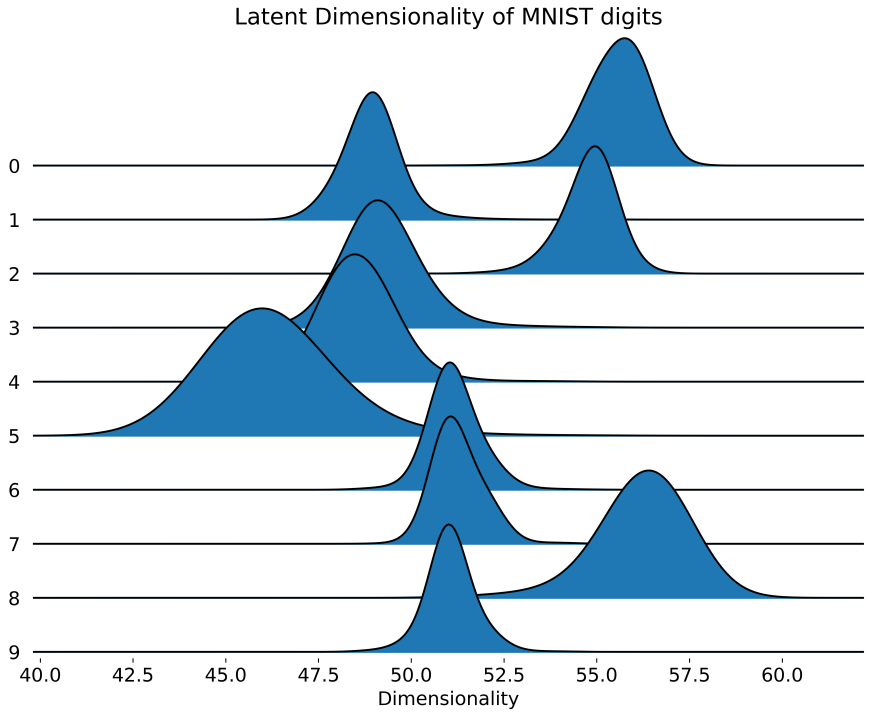


Figure 3.1 Plot of latent dimension distribution learned by *SparC-IB* aggregated for each of the MNIST data classes. In the vertical axis, we have 10 digits classes and in the horizontal axis we have their distribution of posterior modes of latent dimension aggregated across the testset data points. It shows our *SparC-IB* prior provides flexibility to learn the data-specific latent dimension distribution, in contrast to fixing to a single value.

account the uncertainty involved in introducing sparsity, and treat this as a deterministic selection problem. Approaches such as the Indian buffet process VAE ([26]) relax this and allow learning a distribution over the selection parameters that induce sparsity, but only allow global sparsity. Ignoring uncertainty in selection and the flexibility to learn a local data-driven dimensionality of the latent space for each data point can lead to a loss of robustness in inference and prediction.

We also note that the aforementioned approaches have been proposed for unsupervised learning, and we are interested in the supervised learning scenario introduced with the IB approach, which poses a different set of challenges because the sparsity has to accommodate accurate prediction of Y. Only one recent work ([13]) that we know of has considered sparsity in the latent variables of the IB model, but here the latent variable is assumed to be deterministic, and the sparsity applied indirectly by weighting each dimension using a Bernoulli distribution, where zero weight is equivalent to sparsification.

To that end, we make the following contributions:

1. We introduce a novel sparsity-inducing Bayesian spike-slab prior that is based on a beta-binomial formulation, where the sparsity in the latent variables of the IB model is modeled stochastically through a categorical distribution, and thus the joint distribution of latent variable and sparsity is learned with this categorical prior IB (*SparC-IB*) model through Bayesian inference. Unlike traditional spike-and-slab priors that are based on the beta-Bernoulli distribution ([9],[11]) and can select dimensions randomly, *SparC-IB* imposes an order in which the dimensions are selected/activated for each data point. This helps to infer the distribution of dimensionality more effectively.
2. We derive variational lower bounds for efficient inference of the proposed *SparC-IB* model.
3. Using in-distribution and out-of-distribution experiments with MNIST, CIFAR-10, and ImageNet data, we show an improvement in accuracy and robustness with *SparC-IB* compared to vanilla VIB models and other sparsity-inducing strategies.
4. Through extensive analysis of the latent space, we show that learning the joint distribution provides the flexibility to systematically learn parsimonious data-specific (local) latent dimension complexity (as shown in Fig. 3.1), enabling them to learn robust representations.

3.2 Related Works

Previous works in the literature on latent-variable models has looked at different sparsity-inducing mechanisms in the latent space in supervised and unsupervised settings. [3] propose a sparse Dirichlet variational autoencoder (sDVAE) that assumes a degenerate Dirichlet distribution on the latent variable. They introduce sparsity in the concentration parameter of the Dirichlet distribution through a deterministic neural network. Epitome VAE (eVAE) by [35] imposes sparsity through epitomes that select adjacent coordinates of the mean-field Gaussian latent vector and mask the others before feeding to the decoder. The authors propose training a deterministic epitome selector to select from all possible epitomes. Similar to eVAE, the variational sparse coding (VSC) proposed in [31] introduces sparsity through a deterministic classifier.

In this case, the classifier outputs an index from a set of pseudo-inputs that define the prior for

Approach	Latent Variable	Sparsity (global/local)
sDVAE	S	D (L)
eVAE	S	D (L)
VSC	S	D (L)
InteL-VAE	S	D (L)
Drop-B	D	S (G)
IBP	S	S (G)
SparC-IB	S	S (L)

Table 3.1 Latent-variable models with different sparsity induction strategies, where D=Deterministic and S=Stochastic. The type of induced sparsity is in parentheses, where G is global sparsity and L the local sparsity.

the latent variable. In a more recent work, InteL-VAE ([21]) introduces sparsity via a dimension selector (DS) network on top of the Gaussian encoder layer in standard VAEs. The output of DS is multiplied by the Gaussian encoder to induce sparsity and then fed to the decoder. InteL-VAE has empirically shown an improvement over VSC in unsupervised learning tasks, such as image generation. The Indian buffet process (IBP) ([26]) learns a distribution on infinite-dimensional sparse matrices where each element of the matrix is an independent Bernoulli random variable, where the probabilities are global and come from a Beta distribution. Therefore, the sparsity induced by IBP is global and can make the coordinates of the latent variable zero for all data points.

In the IB literature, we find the aspect of sparsity in the latent space rarely explored. To the best of our knowledge, Drop-Bottleneck (Drop-B) by [13] is the only work that attempts this problem. In Drop-B, a neural network extracts features from the data; then, a Bernoulli random variable stochastically drops certain features before passing them to the decoder. The probabilities of the Bernoulli distribution, similar to IBP, are assumed as global parameters and trained with other parameters of the model.

In this paper, with *SparC-IB*, we model stochasticity in both latent variables and sparsity, and we relax the global sparsity assumption by learning the distribution of (local) sparsity for each data point. In this regard, we differ from other latent-variable models. In Table 3.1 we summarize these different approaches by the types, stochastic (S) or deterministic (D), of the latent variable

and the sparsity. Furthermore, we characterize the sparsity induced by each method by whether they impose global (G) or local (L) sparsity. The table shows that very few works incorporate stochasticity in both latent variable and sparsity-inducing mechanism.

3.3 Information Bottleneck with Sparsity-Inducing Categorical Prior

3.3.1 Information Bottleneck: Preliminaries

Taking into account a joint distribution $P(X, Y)$ of the input variable X and the corresponding target variable Y, the information bottleneck principle aims to find a (low-dimensional) latent encoding Z by maximizing prediction accuracy, formulated in terms of mutual information $\mathrm{MI}(Z; Y)$, given a constraint on compressing the latent encoding, formulated in terms of mutual information $\mathrm{MI}(X; Z)$. This can be cast as a constrained optimization problem:

$$\begin{aligned} &\max_Z \mathrm{MI}(Z; Y) \\ &\text{s.t.} \quad \mathrm{MI}(X; Z) \leq C, \end{aligned} \tag{3.1}$$

where C can be interpreted as the compression rate or the minimum number of bits needed to describe the input data. Mutual information is obtained through a multidimensional integral that depends on the joint distribution and the marginal distribution of random variables given by $\int_{\mathcal{Z}} \int_{\mathcal{X}} P_\theta(x, z) \log \left(\frac{P_\theta(x,z)}{P(x)\,P(z)} \right) dx\, dz$, where $P_\theta(x, z) := P_\theta(z|x)P(x)$; a similar expression for $\mathrm{MI}(Z; Y)$ needs $P_\theta(y, z) := \int P(y|x)P_\theta(z|x)P(x)\mathrm{d}x$. The integral presented to calculate MI is generally computationally intractable for large data.

Thus, in practice, the Lagrangian relaxation of the constrained optimization problem is adopted [29]:

$$\mathcal{L}_{IB}(Z) = \mathrm{MI}(Z; Y) - \beta\, \mathrm{MI}(X; Z) \tag{3.2}$$

where β is a Lagrange multiplier that enforces the constraint $\mathrm{MI}(X; Z) \leq C$ such that a latent encoding Z is desired that is maximally expressive about Y while being maximally compressive about X. In other words, $\mathrm{MI}(X; Z)$ is the mutual information that reflects how much the representation (Z) compresses X, and $\mathrm{MI}(Z; Y)$ reflects how much information the representation has been kept from Y. Several approaches have been proposed in the literature to approximate mutual information $\mathrm{MI}(X; Z)$, ranging from parametric bounds defined by variational lower bounds ([2])

to non-parametric bounds (based on kernel density estimate) ([16]) and adversarial f-divergences [37]. In this research, we focus primarily on the variational lower bounds-based approximation. Furthermore, we take a square transformation of the term $\mathrm{MI}(X;Z)$ following [16]. Taking a convex transformation of the compression term makes the solution of the IB Lagrangian identifiable w.r.t. β ([25]). However, for the sake of clarity, we drop this transformation from the loss function derivation. From the convexity property, all derivations with standard IB loss carry over to the loss function with this transformation.

Role of Prior Distribution and Stochasticity of Latent Variable: In the IB formulation presented above, the latent variable is assumed to be stochastic, and hence the posterior distribution of it is learned using Bayesian inference. In [2], the variational lower bound of $\mathcal{L}_{IB}(Z)$ is given as

$$\mathcal{L}_{VIB}(Z) = \mathbb{E}_{X,Y}\left[\mathbb{E}_{Z|X} \log q(Y|Z) - \beta \ \mathrm{D}_{\mathrm{KL}}(q(Z|X)||q(Z))\right] \tag{3.3}$$

In equation 3.3, the prior $q(z)$ serves as a penalty to the variational encoder $q(z|x)$ and $q(y|z)$ is the decoder that is the variational approximation to $p(y|z)$. [2] choose the encoder family and the prior to be a fixed K-dimensional isotropic multivariate Gaussian distribution ($N(0_K, I_{K\times K})$). This choice is motivated by the simplicity and efficiency of inference using differentiable reparameterization of the Gaussian random variable Z in terms of its mean and sigma. *For complex datasets, however, this can be restrictive since the same dimensionality is imposed on all the data and hence can prohibit the latent space from learning robust features* ([21]).

We describe a new family of sparsity-inducing priors that allow stochastic exploration of different dimensional latent spaces. For the proposed variational family, we derive the variational lower bound that consists of discrete and continuous variables, and show that simple reparameterization steps can be taken to draw samples efficiently from the encoder for inference and prediction.

3.3.2 Sparsity-Inducing Categorical Prior

A key aspect of the IB model and latent variable models in general is the dimension of Z. Fixing a very low dimension of Z can impose a limitation on the capacity of the model, while a very large Z can lead to learning a lot more nuisance information that can be detrimental to model

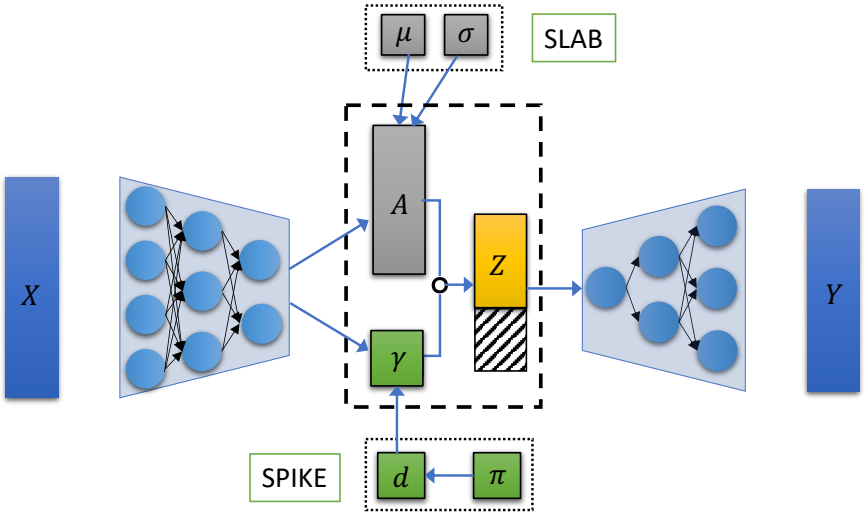


Figure 3.2 Schematic of the categorical prior information bottleneck (*SparC-IB*) model. The inputs are passed through the encoder layer to estimate the feature allocation vector A, while simultaneously using a categorical prior to sample the number of active dimensions d of A for given data that selects the complexity of latent variable Z. The obtained Z is then fed to the decoder to predict the supervised learning responses.

robustness and generalizability. We formulate a data-driven approach to learn the distribution of dimensionality Z through the design of the sparsity-inducing prior.

Let $\{X_n, Y_n\}_{n=1}^N$ be N data points with $X_n \in \mathbb{R}^{1\times p}$ and let the latent variable $Z_n = (Z_{n,1}, Z_{n,2}, ..., Z_{n,K})$, where K is the latent dimensionality. The idea is that we will fix K to be large a priori and make the prior assumption that the $k < K$ columns of Z_n are zero and therefore do not contribute to the prediction of Y_n. Therefore, the prior distribution of Z_n can be specified as follows.

$$Z_{n,k}|\gamma_{n,k} \sim (1-\gamma_{n,k})\,\mathbb{1}(Z_{n,k}=0) + \gamma_{n,k}\mathrm{N}(\mu_{n,k}, \sigma^2_{n,k})$$

OR

$$Z_n = A_n \circ \gamma_n; A_n \sim \mathcal{N}(\mu_n, \Sigma_n) \tag{3.4}$$

$$\gamma_{n,k}|d_n = \mathbb{1}(k \leqslant d_n); d_n \sim \text{Categorical}(\pi_n) \tag{3.5}$$

This prior is a variation of the spike-slab family ([9]) where the sparsity-inducing parameters follow a categorical distribution. The latent variable Z_n is an element-wise product between the feature allocation vector A_n and a sparsity-inducing vector γ_n. The K dimensional latent A_n is assumed to follow a K dimensional Gaussian distribution with mean vector μ_n and diagonal covariance matrix

Σ_n with the variance vector σ_n^2. Note that γ_n is a vector whose first d_n coordinates are 1s and the rest are 0s, and hence $A_n \circ \gamma_n$, with the result that the first d_n coordinates of Z_n are nonzero and the rest are 0. Therefore, as we sample d_n, we consider the different dimensionality of the latent space; d_n follows a categorical distribution over the K categories with π_n as the vector of categorical probabilities whose elements $\pi_{n,k}$ denote the prior probability of the kth dimension for the data point n. In Fig. 3.2 we present the flow of the model from the input X to the output Y through the latent encoding layer.

3.3.3 Variational Bound Derivation

The variable d_n augments Z_n for latent space characterization. With a Markov chain assumption, $Y \leftrightarrow X \leftrightarrow Z$, the latent variable distribution factorization as $p(Z, d) = \prod_{n=1}^{N} p(Z_n|d_n)p(d_n)$, and the same family with the same factorization of the variational posterior $q(Z_n, d_n|X)$ as the prior, we derive the variational lower bound of the IB Lagrangian as follows. In the following derivation, for simplicity, we use X interchangeably for both the random variable $\in \mathbb{R}^{1\times p}$ and the sample covariate matrix $\in \mathbb{R}^{N\times p}$.

$$
\begin{aligned}
&\mathcal{L}_{IB}(Z) = \mathrm{MI}(Z;Y) - \beta\, \mathrm{MI}(X;Z) \\
&\geqslant \mathcal{L}_{VIB}(Z) \\
&= \mathbb{E}_{X,Y}\Big[\mathbb{E}_{Z|X} \log q(Y|Z) - \beta\ \mathrm{D}_{\mathrm{KL}}(q(Z|X)||q(Z))\Big] \quad \text{(from 3.3)} \\
&= \mathbb{E}_{X,Y}\Big[\mathbb{E}_{Z|X} \log q(Y|Z) - \beta\ \mathrm{D}_{\mathrm{KL}}(q(Z,d|X)||q(Z,d))\Big] \\
&= \mathcal{L}_{SparC\text{-}IB}(Z,d)
\end{aligned}
$$

Note that we have introduced the random variable d along with Z in the density function of the second KL term. This is because the KL divergence between $q(Z|X)$ and $q(Z)$ is intractable. The equality of the loss function is valid since $d_n = \sum_{k=1}^{K} \mathbb{1}(Z_{n,k} \neq 0)$ is a deterministic function of Z_n, and we can write the density $q(z, d) = q(z)q(d|z) = q(z)\delta_{d_z}(d)$, where $\delta_c(d)$ is the Dirac delta

function at c. Therefore, we have the following.

$$\begin{aligned}
&\mathrm{D_{KL}}(q(Z,d|X)||q(Z,d)) = \mathbb{E}_{Z,d|X} \log \frac{q(Z|X)\delta_{d_Z(d)}}{q(Z)\delta_{d_Z(d)}} \\
&= \mathbb{E}_{Z|X}\mathbb{E}_{d|Z} \log \frac{q(Z|X)\delta_{d_Z}(d)}{q(Z)\delta_{d_Z}(d)} = \mathbb{E}_{Z|X} \log \frac{q(Z|X)}{q(Z)} \\
&= \mathrm{D_{KL}}(q(Z|X)||q(Z))
\end{aligned}$$

Note that $\delta_{d_Z}(d) = 1$, given Z, almost everywhere since $\delta_{d_Z}(d) = 0 \iff d \neq d_z$ has measure 0 under $q(d|z) = \delta_{d_z}(d)$. We now replace $\mathbb{E}_{X,Y}$ with the empirical version.

$$\begin{aligned}
\mathcal{L}_{SparC\text{-}IB}(Z,d) \hat{=} \frac{1}{N} \sum_{n=1}^{N} \Big[& \mathbb{E}_{Z_n|X}[\log q(Y_n|Z_n)] \\
& - \beta \ \mathrm{D_{KL}}(q(Z_n,d_n|X)||q(Z_n,d_n)) \Big] \\
= \frac{1}{N} \sum_{n=1}^{N} \Big[& \mathbb{E}_{d_n|X}\mathbb{E}_{Z_n|X,d_n}[\log q(Y_n|Z_n)] \\
& - \beta \left[\mathbb{E}_{d_n|X}[\mathrm{D_{KL}}(q(Z_n|X,d_n)||q(Z_n|d_n))] \right. \\
& \left. + \mathrm{D_{KL}}(q(d_n|X)||q(d_n)) \right] \Big]
\end{aligned}$$

We analyze the three terms in the above decomposition as follows.

$$\begin{aligned}
&(i)\ \mathbb{E}_{d_n|X}\mathbb{E}_{Z_n|X,d_n}[\log q(Y_n|Z_n)] \\
&= \sum_{k=1}^{K} \mathbb{E}_{Z_n|X,d_n=k}[\log q(Y_n|Z_n, d_n = k)]\pi_{n,k}(X)
\end{aligned}$$

This term is a weighted average of the negative cross-entropy losses from models with increasing dimension of latent space, where the weights are the posterior probabilities of the dimension encoder. Therefore, *maximizing this term implies putting large weights on the dimensions of the latent space where log-likelihood is high*. During training, this term can be computed using the Monte Carlo approximation, that is, $(i) \hat{=} \frac{1}{J} \sum_{j=1}^{J} \log q(Y_n|Z_n = Z_n^{(j)}, d_n = d_n^{(j)})$, where we draw J randomly drawn samples from $q(Z_n, d_n|X)$. In our experiments, we fixed $J = 10$ everywhere

during training.

$$(ii)\ \mathbb{E}_{d_n|X}[\mathrm{D_{KL}}(q(Z_n|X,d_n)||q(Z_n|d_n))]$$
$$= \sum_{k=1}^{K} \mathrm{D_{KL}}(q(Z_n|X,d_n=k)||q(Z_n|d_n=k))\pi_{n,k}(X)$$

Note that $q(z_n|d_n) = q(z_n|\gamma_n) = \mathcal{N}(z_n;\tilde{\mu}_n,\tilde{\Sigma}_n)$, where $\tilde{\mu}_n = \mu_n \circ \gamma_n$ and $\tilde{\Sigma}_n$ are diagonal with the entries $\tilde{\sigma}_n^2 = \sigma_n^2 \circ \gamma_n$. When $k < K$, this density does not exist w.r.t. the Lebesgue measure in $\mathbb{R}^K$. However, we can still define a density w.r.t. the Lebesgue measure restricted to $\mathbb{R}^k$ (see Chapter 8 in [24]), and it is the k-dimensional multivariate normal density with mean $\mu_{n,-\overline{K-k}} = (\mu_{n,1},...,\mu_{n,k})'$ and diagonal covariance matrix $\Sigma_{n,-\overline{K-k}}$ with diagonal entries $\sigma^2_{n,-\overline{K-k}} = (\sigma^2_{1,n},...,\sigma^2_{n,k})'$. Denoting $\mu_{n,-0} = \mu_n$, we have the following.

$$\begin{aligned}(ii)term &= \sum_{k=1}^{K} \mathrm{D_{KL}}\Big(\mathcal{N}(\mu_{n,-\overline{K-k}}(X),\Sigma_{n,-\overline{K-k}}(X)) \\ &\qquad ||\mathcal{N}(\mu_{n,-\overline{K-k}},\Sigma_{n,-\overline{K-k}})\Big)\pi_{n,k}(X) \\ &= \sum_{k=1}^{K}\sum_{\ell=1}^{k} \mathrm{D_{KL}}\Big(\mathcal{N}(\mu_{n,\ell}(X),\sigma^2_{n,\ell}(X)) \\ &\qquad ||\mathcal{N}(\mu_{n,\ell},\sigma^2_{n,\ell})\Big)\pi_{n,k}(X) \\ &= \frac{1}{2}\sum_{k=1}^{K}\sum_{\ell=1}^{k}[\sigma^2_{n,\ell}(X) - 1 - \log(\sigma^2_{n,\ell}(X)) \\ &\qquad + \mu^2_{n,\ell}(X)]\pi_{n,k}(X)\end{aligned}$$

The second-last equality is due to the fact that the KL divergence of multivariate Gaussian densities whose covariances are diagonal can be written as a sum of coordinate-wise KL divergences. Since KL-divergence is always non-negative, *minimizing the above expression implies putting more probability to the smaller-dimensional latent space models since the second summation term is expected to grow with dimension k.*

$$(iii)\ \mathrm{D_{KL}}(q(d_n|X)||q(d_n)) = \sum_{k=1}^{K} \log\frac{\pi_{n,k}(X)}{\pi_{n,k}}\pi_{n,k}(X)$$

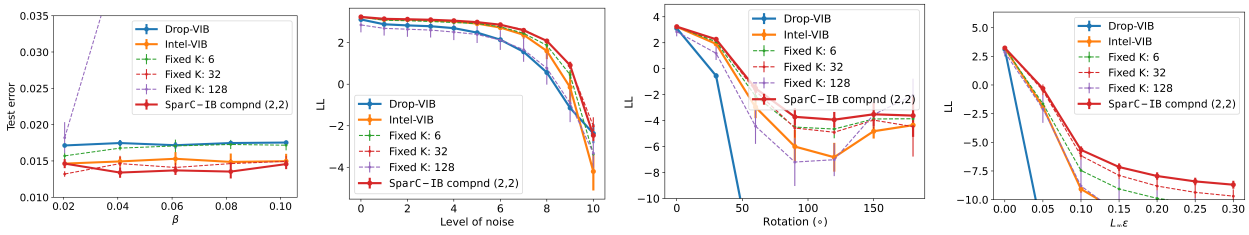
Minimizing this term forces the learned probabilities to be close to the prior. Note that we are learning these probabilities for each data point (since $\pi_{n,k}(X)$ is indexed by n). In this respect, we differ from most of the stochastic sparsity-inducing approaches, such as Drop-B ([13]) and IBP ([26]). In these approaches, the sparsity is induced from a probability distribution with global parameterization and is not learned for each data point.

Modeling Choices for the *SparC-IB* Components: Since $Z_n = A_n \circ \gamma_n$, we are required to fix the priors for (A_n, γ_n) or (A_n, d_n). We chose K-dimensional spherical Gaussian $\mathcal{N}(0, I_K)$ as the prior for the latent variable A_n. For d_n, we assume that the kth categorical probability comes from the compound distribution of a beta-binomial model, also known as the *Polya urn* model [20]. Therefore,

$$\pi_n = \mathbb{P}(d_n = k) = \binom{K-1}{k-1} \frac{\mathrm{B}(a_n + k - 1, b_n + K - k)}{\mathrm{B}(a_n, b_n)}. \tag{3.6}$$

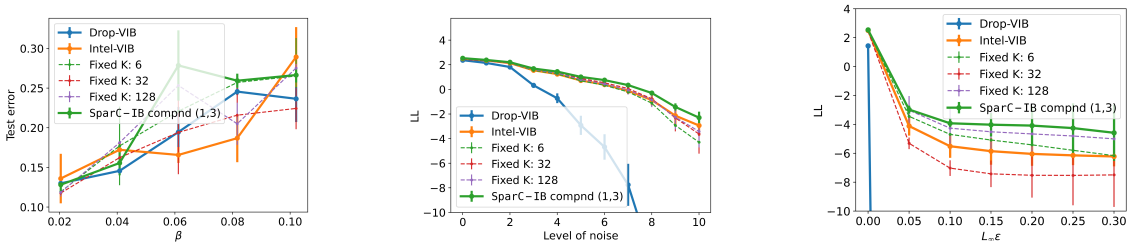
For simplicity, we set the prior value to be constant across the data points, that is, $(a_n, b_n) = (a, b)$. The key advantage of this choice is that we can write the probability as a differentiable function of the two shape parameters (a_n, b_n). Therefore, we can assume the same categorical distribution for the encoder; and instead of learning K probabilities $\pi_{n,k}(X)$ we can learn $(a_n(X), b_n(X))$, which significantly reduces the dimensionality of the parameter space. However, learning $\pi_{n,k}(X)$ provides more flexibility because the shape of the distribution is not constrained, while learning $(a_n(X), b_n(X))$ constrains $\pi_{n,k}(X)$ to follow according to the shape of the compound distribution, which depends on $a_n(X)$ and $b_n(X)$. In our experiments, we tested both approaches and found that learning $(a_n(X), b_n(X))$ produces better results.

3.4 Experimental Results



(a) Test error vs β (b) Log likelihood vs noise (c) Log likelihood vs rotation (d) Log likelihood vs attack

Figure 3.3 In- and out-of-distribution performance on MNIST.



(a) Test error vs β (b) Log likelihood vs noise (c) Log likelihood vs attack

Figure 3.4 In- and out-of- distribution performance on CIFAR-10.

In this section, we present experimental results that compare the performance of *SparC-IB* with the most recent sparsity-inducing strategies proposed in the literature: the Drop-B model ([13]) and InteL-VAE ([21]). We could not find an open source implementation for either of the two approaches and therefore we have coded our own implementation where we adapt them in the context of information bottleneck (link to the code base is provided in the Appendix A3.5). In this section, these two approaches are called Drop-VIB and Intel-VIB. In addition, we compare our model with the baseline mean-field Gaussian VIB approach, where the latent dimension is fixed to a single value across all data. Note that we apply the square transformation ([16]) to the estimator of $\mathrm{MI}(X;Z)$ for all the methods.

We evaluated the *SparC-IB* model for *in-distribution* prediction accuracy in a supervised classification scenario using the MNIST and CIFAR-10 data, which have a small number of classes, and also the ImageNet data, where the number of classes is large. Furthermore, we evaluated the robustness of *SparC-IB* trained on these three datasets in *out-of-distribution* scenarios, specifically with rotation, white-box attacks with noise corruptions, and black-box adversarial attacks.

The selection of the Lagrange multiplier β controls the amount of information learned from the input (that is, $\mathrm{MI}(X;Z)$) by the latent space. We have chosen a common β where $(\mathrm{MI}(X;Z), \mathrm{MI}(Z;Y))$ is close to the *minimum necessary information* or MNI (suggested in [7]) for all models. MNI is a point in the information plane where $\mathrm{MI}(X;Z) = \mathrm{MI}(Z;Y) = \mathrm{H}(Y)$, where the entropy is indicated by $\mathrm{H}(Y)$. We evaluated the robustness of each model using a single value of β. The value of β we chose to compare the models for MNIST is ~ 0.08, for CIFAR-10 it is ~ 0.04, and for ImageNet it is ~ 0.02. The choice of β is discussed in more detail in the Appendix A3.1.

We use the encoder-decoder architecture from [25] for MNIST, from [36] for CIFAR-10, and

from [2] for ImageNet. Note that we learn the parameters of the dimension distribution in both compound and categorical strategies by splitting the encoder network head into two parts, as depicted in Fig. 3.2. Full details of the architectures have been discussed in the Appendix A3.1. Furthermore, following [8], we pass the mean of the encoder Z to the decoder at the test time to make a prediction.

Prior probabilities act as regularizers in learning the dimension probabilities in both categorical and compound strategies (the third term of $\mathcal{L}_{SparC\text{-}IB}(Z, d)$). They also model the prior knowledge or inductive bias that one may have. The prior probability distribution, in this case, 3.6 can be set by the choice of hyperparameters (a, b). we evaluated two different cases, $(a, b) = (1, 3)$ and (2,2) for both the categorical and compound distribution strategies. The choice $(a, b) = (1, 3)$ puts more probability mass on the lower dimensions, and gradually decays with dimension, whereas $(a, b) = (2, 2)$ penalizes models of too high or too low dimensions. The appendix A3.7 shows the prior probabilities of the dimensions for both choices.

3.4.1 In-distribution Data

In this section, we compare the performance of *SparC-IB* with Intel-VIB, Drop-VIB, and the vanilla fixed-dimensional VIB approach on the MNIST, CIFAR-10 test set, and ImageNet validation set. Beyond these choices, we have also compared *SparC-IB* with a discrete latent space IB model following VQ-VAE ([33]) on MNIST data. We train each model for 5 values of the Lagrange multiplier β in the set $(0.02, 0.04, 0.06, 0.08, 0.1)$. We calculate the validation set error for each model for these β values. Since increasing β penalizes the amount of information retained by the latent space about the inputs, we expect the error to increase as β increases.

For MNIST, we find that, across β, the compound distribution prior with $(a, b) = (2, 2)$ performs best in terms of in-distribution prediction accuracy across β as compared to other *SparC-IB* choices, fixed-dimensional VIB approaches and Drop-VIB and Intel-VIB, as shown in Fig. 3.3(a). For CIFAR-10 data, we observe that compound strategy with prior $(a, b) = (1, 3)$ has the best accuracy compared to other *SparC-IB* choices at MNI, fixed-dimensional VIB models and Intel-VIB but is slightly lower than Drop-VIB (numbers are shown in Table 3.2). However, we find from Table 3.2

Methods	MNIST		CIFAR-10	
	Acc %	**LL**	**Acc** %	**LL**
SparC-IB	**98.65** (0.001)	**3.24** (0.004)	84.44 (0.015)	**2.53** (0.010)
Drop-VIB	98.25 (0.000)	3.12 (0.003)	**85.43** (0.004)	2.37 (0.015)
Intel-VIB	98.51 (0.001)	3.23 (0.007)	82.77 (0.010)	2.49 (0.063)
Fixed K: 6	98.27 (0.001)	3.22 (0.004)	82.29 (0.050)	2.49 (0.107)
Fixed K: 32	98.54 (0.001)	3.23 (0.003)	83.76 (0.012)	2.44 (0.013)
Fixed K: 128	85.28 (0.165)	2.85 (0.358)	81.88 (0.001)	2.48 (0.036)

Table 3.2 In-distribution performance of all methods in terms of accuracy and log-likelihood (SD in the parenthesis) at MNI for MNIST and CIFAR-10 (maximum for each column highlighted). *SparC-IB* performs as good as the best performing model in terms of both metrics.

Methods	ImageNet	
	Acc %	**LL**
SparC-IB	79.71 (0.001)	8.48 (0.007)
Drop-VIB	79.86 (0.000)	8.19 (0.002)
Intel-VIB	**79.88 (0.000)**	**8.53 (0.003)**
Fixed K: 1024	79.88 (0.000)	8.52 (0.005)

Table 3.3 In-distribution performance of all methods in terms of accuracy and log-likelihood (SD in the parenthesis) at MNI for ImageNet (maximum for each column highlighted). All models are close in terms of both metrics (especially log-likelihood).

that *SparC-IB* compound (1,3) has the highest log-likelihood at MNI among the other approaches. Furthermore, we have found that the best test accuracy for the discrete latent space IB is 97.79% (avg. over 3 seeds) in MNIST, which is lower than *SparC-IB*.

For Imagenet data, we observe that the compound strategy with prior $(a, b) = (2, 2)$ has the best validation accuracy compared to other *SparC-IB* choices. Furthermore, the in-distribution performance is at least as good as the fixed-dimensional VIB model (Table 3.3), but it has a slightly lower accuracy compared to the Drop-VIB and Intel-VIB. The test error for Intel-VIB is very high when $\beta > 0.04$. This behavior is due to the fact that the dimension selector in Intel-VIB (Section 6.3 in [21]) is pruning almost all values of the latent allocation vector A for values of $\beta > 0.04$.

3.4.2 Out-of-distribution Data

We consider three out-of-distribution scenarios to measure the robustness of our approach and compare it with vanilla VIB with fixed latent dimension capacity, as well as with other sparsity-inducing strategies. The first is a white-box attack, in which we systematically introduce shot noise corruptions into the test data [22, 10]. The second is a rotation transform (only for MNIST data).

The third is the black-box adversarial attack simulated using the projected gradient descent (PGD) strategy [19]. We use the log-likelihood metric to compare the out-of-distribution performance of the methods ([23]). Comparison in terms of other metrics is included in the Appendix A3.8.

3.4.2.1 Noise Corruption

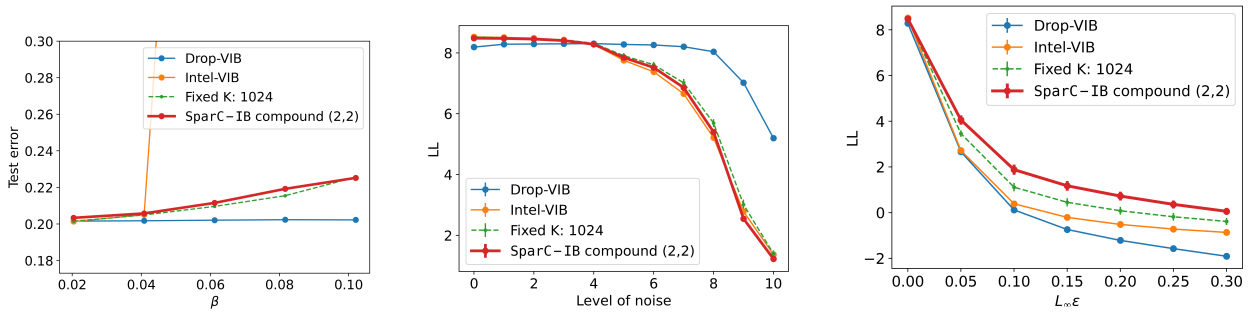
White-box attack or noise corruption is generated by adding shot noise. Following [22], Poisson noise is generated pixel-by-pixel and added to the test images for both MNIST and CIFAR-10. The levels of noise along the horizontal axis of panel (b) of Fig. 3.3 and Fig. 3.4 represent an increasing degree of noise added to the images of the validation set. For our approach and each of the five models that are being compared, we plot the log-likelihood as a function of the level of noise to assess the robustness of these approaches. We find that with both MNIST (panels (b) in Fig. 3.3) and CIFAR-10 (panels (b) in Fig. 3.4), for each of these three metrics, *SparC-IB* outperforms all other approaches compared. For ImageNet (panel (b) in Fig. A3.5), we find that Drop-VIB has a higher likelihood than our approach, possibly due to a large amount of input information (estimated $\mathrm{MI}(X;Z) = 57.56$ for Drop-VIB and = 10.35 for *SparC-IB*) learned in the latent space.

3.4.2.2 Rotation

In this scenario, we evaluate the models trained on MNIST data with increasingly rotated digits following [23], which simulates the scenario of data that are moderately out of distribution to the data used for training the model. We show the results of the experiments in panel (c) in Fig. 3.3. We can see that *SparC-IB* with the compound distribution prior outperforms the rest of the five models in terms of log-likelihood, we also note that the performance of the Drop-VIB has the lowest in this scenario.

3.4.2.3 Adversarial Robustness

Multiple approaches have been proposed in the literature to assess the adversarial robustness of a model [32]. Among them, perhaps the most commonly adopted approach is to test the accuracy in adversarial examples that are generated by adversarially perturbing the test data. This perturbation is in the same spirit as the noise corruption presented in the preceding section, but is designed



(a) Test error vs β value (b) Log likelihood vs noise (c) Log likelihood vs attack radius

Figure 3.5 In- and out-of- distribution results on ImageNet.

to be more catastrophic by adopting a black-box attack approach that chooses a gradient direction of the image pixels at some loss and then takes a single step in that direction. The projected gradient descent is an example of such an adversarial attack. Following [2], we evaluated the robustness of the model to the PGD attack with 10 iterations. We use the L_∞ norm to measure the size of the perturbation, which in this case is a measure of the largest single change in any pixel. Log-likelihood as a function of the perturbed L_∞ distance for MNIST (panel (d) in Fig. 3.3), for CIFAR-10 (panel (c) in Fig. 3.4), and for ImageNet (panel (c) in Fig. A3.5) *show that the SparC-IB approach provides the highest log-likelihood across the attack radius in all three datasets.*

3.4.3 Analysis of the Latent Space

A key property of *SparC-IB* is the ability to jointly learn the latent allocation and the dimension of the latent space for each data point. In this section, our aim is to disentangle the information learned in the latent space of the *SparC-IB* approach by analyzing the posterior distribution of the dimension variable and the information learned in the latent allocation vector for MNIST data. Similar analyses for CIFAR-10 and ImageNet are in the Appendix A3.9.

3.4.3.1 Flexible Latent Space Dimension

A distinct feature of the proposed approach is the flexibility enabled by the sparsity-inducing prior for learning a data-dependent dimension distribution. To demonstrate this, for a compound model with $(a, b) = (2, 2)$ in Fig. 3.1 we show the distribution of posterior modes of dimension distribution across data points, aggregated per MNIST digit. We see that, in fact, each digit, on average, preferred to have a different latent dimension. We further note that the mode values

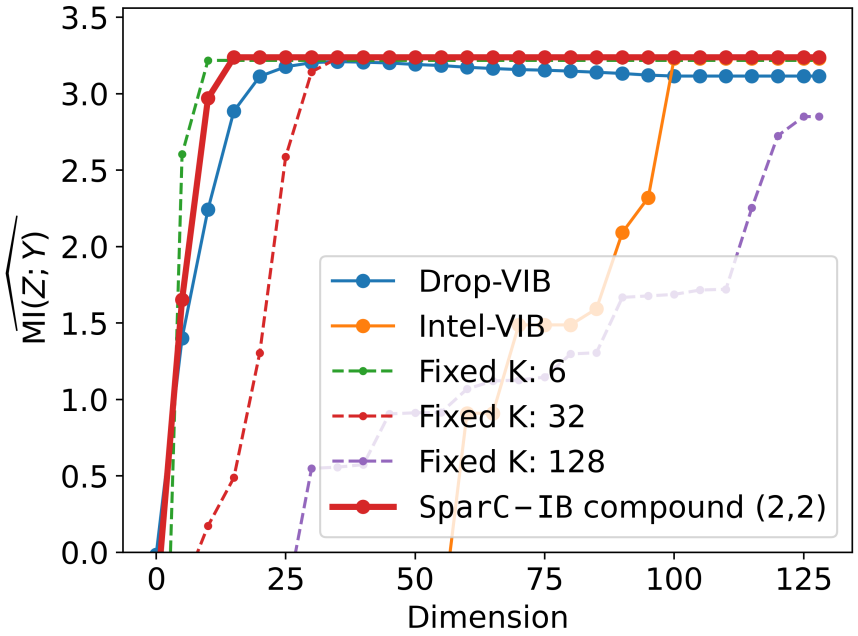


Figure 3.6 Information content plot for different latent dimensions (averaged across seeds). We observe that *SparC-IB* learns the maximum information in a small dimensional latent space.

depicted by the plot for digits 5 and 8 are farther away from the rest of the digits. Note that the pattern in Fig. 3.1 is for a single seed. Although the pattern in dimension distribution modes changes across the seeds, the separation between the classes remains (see A3.9 in the Appendix for details). We also observe a similar separation of the latent dimensionality across classes with CIFAR-10 and ImageNet data (Appendix Fig. A3.7).

3.4.3.2 Analysis of Information Content

The *SparC-IB* prior in 3.5 induces an ordered selection of the dimensions of the latent allocation vector A_n based on the dimension d_n. In Fig. 3.6, we plotted the estimated mutual information or $\widehat{\mathrm{MI}(Z;Y)}$ against the increasing dimension of the latent space (in increments of 5) for all models in MNIST. For a given dimension d, $\widehat{\mathrm{MI}(Z;Y)} = \frac{1}{N}\sum_{n=1}^{N} \log q(Y_n|Z_n = \mu_n(X) \circ \gamma(d))$, where the first d coordinates of $\gamma(d)$ are 1s and the rest are 0s. We observe that the *SparC-IB* model has been able to code the learned information in the first few ($\sim$ 15) dimensions of the latent space. In contrast, we notice that the fixed-dimensional VIB models, Drop-VIB, and Intel-VIB require close to the full latent space to encode similar information levels. This characteristic perhaps hinders these models in achieving good robustness performance consistently on all data. However, we believe

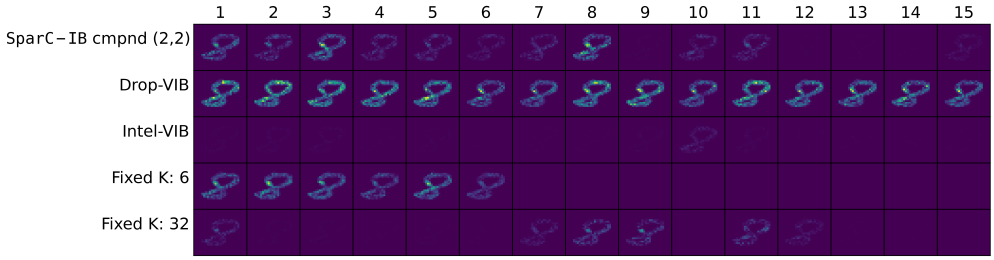


Figure 3.7 Plot of pixel importance in encoding the latent allocation mean (averaged across seeds) for a sample from the MNIST test data. We observe that *SparC-IB* learns important features of the latent space in the first few dimension of the latent space.

that further investigation is needed to establish this claim. For CIFAR-10, we observed the same characteristics of the information content plot in Fig. 3.6 whereas for ImageNet the information plateaus at a larger dimension than Fig. 3.6 (see Fig. A3.8 in the Appendix).

3.4.3.3 Visualizing the Latent Dimensions

In deep neural networks, the conductance of a hidden unit, proposed in [5], is the flow of information from the input through that unit to the predictions. In this spirit, we measure the importance scores of individual pixels in the dimensions of the latent mean allocation vector μ for the MNIST data. The computation details have been added in the Appendix A3.9.4. In Fig. 3.7, we plotted these measures for different methods (averaged across seeds) for the first 15 dimensions of μ for a sample from the MNIST test data. We observe that *SparC-IB* encodes important features in the first few dimensions of the latent space. For Intel-VIB and VIB with K = 32, we see that the pixel information is spread over a large set of dimensions of the latent space. For Drop-VIB, we notice that it learns a lot of information in all the 15 dimensions. Fig. 3.7 also helps explain the information jumps in Fig. 3.6. We notice that the jump in the information content occurs when we include the dimensions that have learned important pixel information, e.g. for *SparC-IB* dimensions 3 and 8. Note that the Intel-VIB and fixed-dimensional VIB models with high dimensions have many dimensions with very little information about the input. Although Fig. 3.7 exhibits some features of the digit learned by the latent space (e.g., the middle part of the digit has high importance for *SparC-IB*), in our experiments, we have not been able to extract meaningful features in the latent dimensions that are common across all the digits. In our view, this demands further investigation.

3.5 Conclusions

In summary, we propose the *SparC-IB* method in this paper, which models the latent variable and its dimension through a Bayesian spike-and-slab categorical prior and derived a variational lower bound for efficient Bayesian inference. This approach accounts for the full uncertainty in the latent space by learning a joint distribution of the latent variable and the sparsity. We compare our approach with commonly used fixed-dimensional priors, as well as using those sparsity-inducing strategies previously proposed in the literature through experiments on MNIST, CIFAR-10, and ImageNet in both the in-distribution and out-of-distribution scenarios (such as noise corruption, adversarial attacks, and rotation). We find that our approach obtains as good accuracy and robustness as the best-performing model in all the cases, and in some cases it outperforms the other models. This is important because we found that other VIB algorithms considered performed well on a few datasets but have significantly poor performance on the others. In addition, we show that enabling each data to learn their own dimension distribution leads to separation of dimension distribution between output classes, thus substantiating that latent dimension varies class-wise. Furthermore, we find that the *SparC-IB* approach provides a compact latent space in which it learns important data features in the first few dimensions of the latent space, which is known to lead to superior robustness properties.

There are several avenues for future research based on the proposed model *SparC-IB*. Since the latent dimensionality of the data is modeled through a Bayesian spike-and-slab prior with a categorical spike distribution over the dimension, an interesting avenue for future research could be to find a rich class of hierarchical priors or a non-parametric stick-breaking prior. Another direction might be to explore other approaches to data-driven mutual information estimation to tighten the lower bound of $\mathrm{MI}(X;Z)$. A key aspect of this work is its broader appeal in the field of adaptive selection of model complexity in machine learning. Beyond IB, the proposed sparsity-inducing prior is a promising candidate for the probabilistic node and depth selection of DNN. Our approach for data-specific complexity learning can potentially be applied to have a NN with an optimal number of layers with ordered and sparse information captured in each layer.

BIBLIOGRAPHY

[1] Ahuja, K., Caballero, E., Zhang, D., Gagnon-Audet, J.-C., Bengio, Y., Mitliagkas, I., and Rish, I. (2021). Invariance principle meets information bottleneck for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 34:3438–3450.

[2] Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. (2016). Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.

[3] Burkhardt, S. and Kramer, S. (2019). Decoupling sparsity and smoothness in the Dirichlet variational autoencoder topic model. *J. Mach. Learn. Res.*, 20(131):1–27.

[4] Chun, H. and Keleş, S. (2010). Sparse partial least squares regression for simultaneous dimension reduction and variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(1):3–25.

[5] Dhamdhere, K., Sundararajan, M., and Yan, Q. (2018). How important is a neuron?

[6] Fan, T.-H., Chi, T.-C., Rudnicky, A. I., and Ramadge, P. J. (2022). Training discrete deep generative models via gapped straight-through estimator.

[7] Fischer, I. (2020). The conditional entropy bottleneck. *Entropy*, 22(9).

[8] Fischer, I. and Alemi, A. A. (2020). Ceb improves model robustness. *Entropy*, 22(10):1081.

[9] George, E. I. and McCulloch, R. E. (1993). Variable selection via gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889.

[10] Hendrycks, D. and Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. In *7th International Conference on Learning Representations (ICLR-2019)*.

[11] Ishwaran, H. and Rao, J. S. (2005). Spike and slab variable selection: Frequentist and Bayesian strategies. *The Annals of Statistics*, 33(2):730 – 773.

[12] Jang, E., Gu, S., and Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

[13] Kim, J., Kim, M., Woo, D., and Kim, G. (2021). Drop-bottleneck: Learning discrete compressed representation for noise-robust exploration. *arXiv preprint arXiv:2103.12300*.

[14] Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

[15] Kokhlikyan, N., Miglani, V., Martin, M., Wang, E., Alsallakh, B., Reynolds, J., Melnikov,

A., Kliushkina, N., Araya, C., Yan, S., et al. (2020). Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*.

[16] Kolchinsky, A., Tracey, B. D., and Wolpert, D. H. (2019). Nonlinear information bottleneck. *Entropy*, 21(12).

[17] Lan, A. S., Waters, A. E., Studer, C., and Baraniuk, R. G. (2014). Sparse factor analysis for learning and content analytics. *Journal of Machine Learning Research (JMLR)*, 15(57):1959–2008.

[18] Maddison, C. J., Mnih, A., and Teh, Y. W. (2016). The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*.

[19] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

[20] Mahmoud, H. (2008). *Pólya urn models*. Chapman and Hall/CRC.

[21] Miao, N., Mathieu, E., Siddharth, N., Teh, Y. W., and Rainforth, T. (2021). On incorporating inductive biases into vaes. *arXiv preprint arXiv:2106.13746*.

[22] Mu, N. and Gilmer, J. (2019). Mnist-c: A robustness benchmark for computer vision. *arXiv preprint arXiv:1906.02337*.

[23] Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J., Lakshminarayanan, B., and Snoek, J. (2019). Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*, 32.

[24] Rao, C. R. (1973). *Linear statistical inference and its applications*, volume 2. Wiley New York.

[25] Rodríguez Gálvez, B., Thobaben, R., and Skoglund, M. (2020). The convex information bottleneck Lagrangian. *Entropy*, 22(1):98.

[26] Singh, R., Ling, J., and Doshi-Velez, F. (2017). Structured variational autoencoders for the Beta–Bernoulli process. In *NIPS 2017 Workshop on Advances in Approximate Bayesian Inference*.

[27] Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. (2016). Inception-v4, inception-resnet and the impact of residual connections on learning.

[28] Tishby, N., Pereira, F. C., and Bialek, W. (2000). The information bottleneck method. *arXiv preprint physics/0004057*.

[29] Tishby, N. and Zaslavsky, N. (2015). Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, pages 1–5. IEEE.

[30] Tomczak, J. M. (2022). *Deep Generative Modeling*. Springer Nature.

[31] Tonolini, F., Jensen, B. S., and Murray-Smith, R. (2020). Variational sparse coding. In *Uncertainty in Artificial Intelligence*, pages 690–700. PMLR.

[32] Tramer, F., Carlini, N., Brendel, W., and Madry, A. (2020). On adaptive attacks to adversarial example defenses. *Advances in Neural Information Processing Systems*, 33:1633–1645.

[33] Van Den Oord, A., Vinyals, O., et al. (2017). Neural discrete representation learning. *Advances in neural information processing systems*, 30.

[34] Yang, X., Liu, W., Liu, W., and Tao, D. (2019). A survey on canonical correlation analysis. *IEEE Transactions on Knowledge and Data Engineering*, 33(6):2349–2368.

[35] Yeung, S., Kannan, A., Dauphin, Y., and Fei-Fei, L. (2017). Tackling over-pruning in variational autoencoders. *arXiv preprint arXiv:1706.03643*.

[36] Yu, X., Yu, S., and Príncipe, J. C. (2021). Deep deterministic information bottleneck with matrix-based entropy functional. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3160–3164. IEEE.

[37] Zhai, P. and Zhang, S. (2021). Adversarial information bottleneck. *arXiv preprint arXiv:2103.00381*.

APPENDIX A3

SUPPLEMENTARY DATA

A3.1 Model Architectures and Hyperparameter settings

For the proposed *SparC-IB* model, we need to select the neural architecture to be used for the latent space mean and variance, as well as the dimension encoder that learns categorical probabilities or compound distribution parameters. To learn the parameters of the dimension distribution in both compound and categorical strategies, we split the head of the encoder network into two parts, as depicted in Fig. 3.2. The first part estimates the parameters of the latent allocation variable A and the second part estimates the parameters of the dimension variable d. For MNIST, we follow [25] and use an MLP encoder with three fully connected layers, the first two of dimensions 800 and ReLu activation, and the last layer with dimension $2K + 2$ for the compound strategy and $3K$ for the categorical strategy. The decoder consists of two fully connected layers, the first of which has dimension 800 and the second predicts the softmax class probabilities in the last layer. For CIFAR-10, following [36], we adopt a VGG16 encoder and a single-layered neural network as decoder. We choose the final sequential layer of VGG16 as the bottleneck layer that outputs the parameters of the latent space.

For ImageNet, we crop the images at its center to make them 299×299 pixels and normalize them to have mean = (0.5, 0.5, 0.5) and standard deviation = (0.5, 0.5, 0.5). We have followed the implementation of [2] where we transform the ImageNet data with a pre-trained Inception Resnet V2 ([27]) network without the output layer. Under this transformation, the original ImageNet images reduce to a 1534-dimensional representation, which we used for all our results. Following [2], we use an encoder with two fully connected layers, each with 1024 hidden units, and a single-layer decoder architecture.

For comparison with other sparsity-inducing approaches, we chose the two most recent works: the Drop-B model ([13]) and the InteL-VAE ([21]). The Drop-B implementation requires a feature extractor. For all three data sets, we choose the same architecture for the feature extractor as the encoder of *SparC-IB* until the final layer, which has K dimensions. We assume the same decoder

Hyperparameters	MNIST	CIFAR-10	ImageNet
Train set size	60,000	50,000	128,1167
Validation set size	10,000	10,000	50,000
# epochs	100	400	200
Training batch size	128	100	2000
Optimizer	Adam	SGD	Adam
Initial learning rate	1e-4	0.1	1e-4
Learning rate drop	0.6	0.1	0.97
Learning rate drop steps	10 epochs	100 epochs	2 epochs
Weight decay	Not used	5e-4	Not used

Table A3.1 Hyperparameter settings used to model the three data sets.

architecture for Drop-B as for *SparC-IB*. Additionally, for Drop-B, the K Bernoulli probabilities are trained with the other parameters of the model. For InteL-VAE, the encoder and decoder architectures are chosen to be the same as in *SparC-IB* for all data sets. In addition, this model requires a dimension selector (DS) network. Following the experiments in [21], we select three fully connected layers for the DS network with ReLu activations between, where the first two layers have dimension 10 and the last layer has dimension K. We fix K (that is, the prior assumption of dimensionality) to be 100 for MNIST and CIFAR-10 and 1024 for the ImageNet data.

The workflow of *SparC-IB* overlaps with the standard VIB when encoding the mean and sigma of the full dimension of the latent variable. Furthermore, *SparC-IB* encodes categorical probabilities and then draws samples from a categorical distribution. Unlike the reparameterization trick ([14]) for Gaussian variables, there does not exist a differentiable transformation from categorical probabilities to the samples. Therefore, we use the Gumbel-Softmax approximation ([18], [12]) to draw categorical samples. We apply the transformation in 3.5 to the samples and take the element-wise product with the Gaussian samples before passing it to the decoder. Note that there exists other differentiable reparameterization of the discrete samples, e.g., the Gapped Straight-Through (GST) estimator [6]. However, in our experiments, the use of the Gumbel-Softmax approximation has led to a lower loss value than that of the GST.

Fitting deep learning models involves several key hyperparameters. In Table B4.2, we provide the necessary hyperparameters for training and evaluation of all fitted models in three data sets.

A3.2 Data Augmentation

For the CIFAR-10 data, we augmented the training data using random transformations. We pad each training set image by 4 pixels on all sides and crop at a random location to return an original-sized image. We flip each training set image horizontally with probability 0.5. Furthermore, we normalize each training and validation set image with mean = $(0.4914, 0.4822, 0.4465)$ and standard deviation = $(0.2023, 0.1994, 0.2010)$.

A3.3 Convergence Checks

For CIFAR-10, we observed overfitting after 100 epochs for the Drop-VIB model, where the validation loss started to increase. Therefore, we saved the model at epoch 100 for our robustness analysis. For other models in our experiments on MNIST and CIFAR-10, we observed the convergence of train and validation loss, and we have considered models at the final epoch as the final models. For ImageNet, we saved the model with the lowest validation loss as our final model for all the methods.

A3.4 Evaluation Metrics

We calculated three evaluation metrics for each method in each scenario: test error, log-likelihood, and Brier score. For prediction in all in- and out-of-distribution scenarios, following [8], we have used the mean latent space $\mathbb{E}(Z|X)$ as input to the decoder for all methods. Note that for *SparC-IB* we calculate the marginal expectation by following, $\mathbb{E}(Z|X) = \mathbb{E}_{q(d|X)}\mathbb{E}_{q(Z|d,X)}(Z) \hat{=} \frac{1}{J}\sum_{j=1}^{J}\mathbb{E}_{q(Z|d=d_j,X)}(Z)$, where d_j is a sample from $q(d|X)$. In our experiments, we have fixed $J = 10$.

A3.5 Software AND Hardware

We have forked the code base https://github.com/burklight/convex-IB-Lagrangian-PyTorch.git that implements the Convex-IB method ([25]) using PyTorch. The code to run the models used in the experiments can be found in the following repository https://github.com/AnirbanSamaddar/SparC-IB. For modeling MNIST, we used NVIDIA V100 GPUs and for CIFAR-10 and ImageNet experiments, we used NVIDIA A100 GPUs.

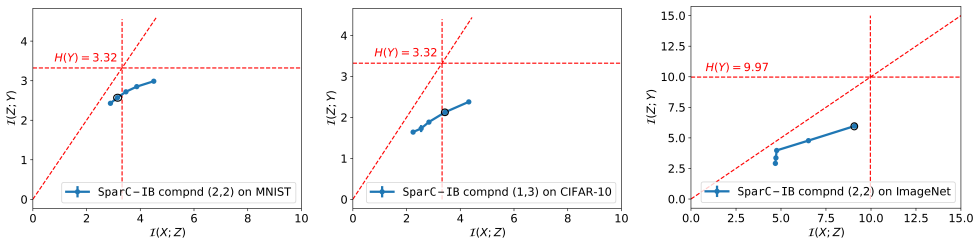


Figure A3.1 Information curve on MNIST, CIFAR-10, and ImageNet.

A3.6 Information Curve: Selection of β

In the IB Lagrangian, the Lagrange multiplier β controls the trade-off between two MI terms. By optimizing the IB objective for different values of β, we can explore the information curve, which is the plot of $(\mathrm{MI}(X;Z), \mathrm{MI}(Z;Y))$ in the 2-d plane. Fig. A3.1 shows the information curve on the validation set for the models selected for MNIST, CIFAR-10 and ImageNet for the robustness studies in the main article. For the fixed K VIB models, the information curves are similar to Fig. A3.1. *Minimum necessary information* ([7]) is a point in the information plane where $\mathrm{MI}(X;Z) = \mathrm{MI}(Z;Y) = \mathrm{H}(Y)$, where the entropy is indicated by $\mathrm{H}(Y)$. For classification tasks, where labels are deterministic given the images, the entropy $\mathrm{H}(Y) = \log_2 n_c$, where n_c is the number of classes. Therefore, for MNIST and CIFAR-10 $\mathrm{H}(Y) = \log_2 10 \sim 3.32$ and for ImageNet $\mathrm{H}(Y) = \log_2 10 \sim 9.97$. Therefore, we choose $\beta \sim 0.08$ for MNIST, $\beta \sim 0.04$ for CIFAR-10, and $\beta \sim 0.02$ for ImageNet, which gives us the closest proximity to MNI. The points are circled in Fig. A3.1.

A3.7 Selection of Prior Parameters *SparC-IB*

Fig. A3.2 shows the probabilities of the prior dimension considered for modeling the three data sets. These prior probabilities are from the compound distribution (Eq. 3.6) with $K = 100$ (left figure) and $K = 1024$ (right figure).

A3.8 Performance based on the Evaluation Metrics on *out-of-distribution* data

In the main paper, we have presented the robustness results for the three data sets in terms of the log-likelihood. Here we present the robustness results in terms of the test set error and the Brier score for all the methods across the three data sets.

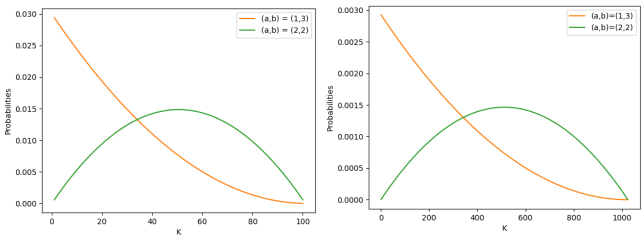
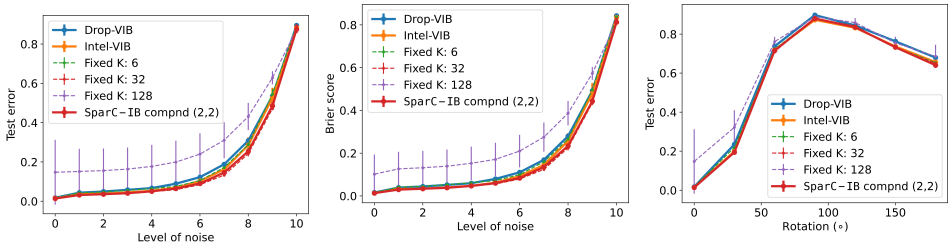
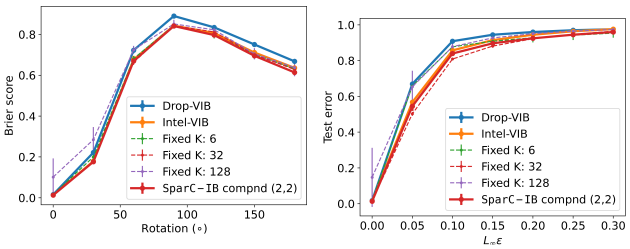


Figure A3.2 Plot of prior dimension probabilities for choices of (a, b).

Fig. A3.3 (a)-(e) shows the results for the *out-of-distribution* MNIST test data. From the figures, we observe that in all the scenarios *SparC-IB* performs as well as the best performing model. We further observe that the separation between the models in terms of the test error and the Brier score is less than the log-likelihood which is our chosen metric in the main paper. Similar results for CIFAR10 and ImageNet have been presented in Fig. A3.4 and Fig. A3.5 respectively. For CIFAR10 (Fig. A3.4), we observe similar behavior in terms of the test error and the Brier score as for MNIST. For ImageNet (Fig. A3.5), we observe that *SparC-IB* is doing better in terms of test error in the L_∞ attacks. However, Drop-VIB seems to be performing better than other approaches for the white-box attacks. This behavior has been discussed in the main paper (Sec. 3.4.2.1).

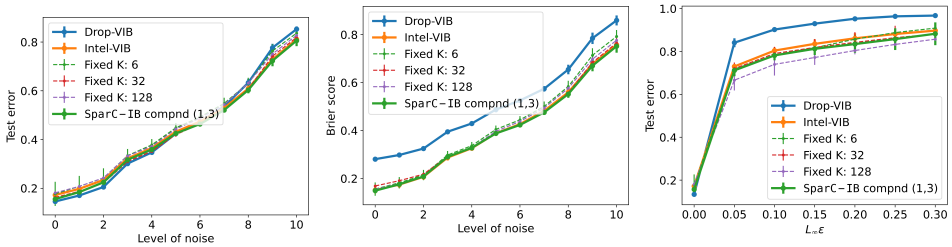


(a) Test error vs noise (b) Brier score vs noise (c) Test error vs rotation



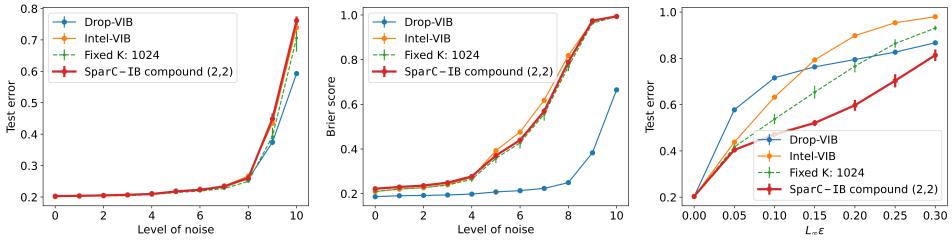
(d) Brier score vs rotation (e) Test error vs L_∞ radius

Figure A3.3 Out-of-distribution performance in terms of the test error, and the Brier score on MNIST. We observe that *SparC-IB* approach with compound strategy and $(a, b) = (2, 2)$ (red line) performs as well as the best-performing model in all the cases.



(a) Test error vs noise (b) Brier core vs noise (c) Test error vs L_∞ radius

Figure A3.4 Out-of-distribution performance in terms of the test error, and the Brier score on CIFAR-10. We observe that *SparC-IB* approach with compound strategy and $(a, b) = (1, 3)$ (green line) performs as good as the best performing model in all the cases.



(a) Test error vs noise (b) Brier score vs noise (c) Test error vs L_∞ radius

Figure A3.5 Out-of-distribution performance in terms of the test error, and the Brier score on ImageNet. In the white-noise scenario, the drop-VIB performs better than our approach possibly due to learning more information about X. However, we observe that *SparC-IB* approach *outperforms other models in black-box attacks*.

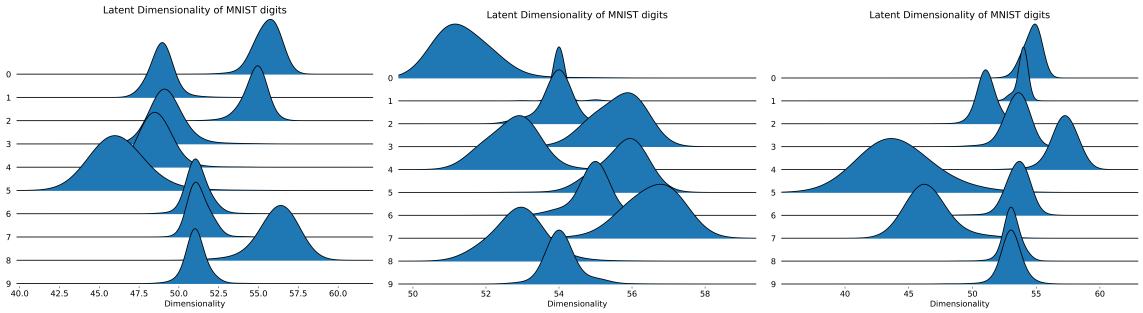
A3.9 Analysis of Latent Space

A3.9.1 Dimension Distribution Mode Plot across Seeds

Fig. A3.6 shows the mode plot for *SparC-IB* compound (2,2) across 3 seeds on MNIST data. The overall range of dimensions remains unchanged across the 3 seeds; however, we observe that each digit prefers a different dimension of the latent space.

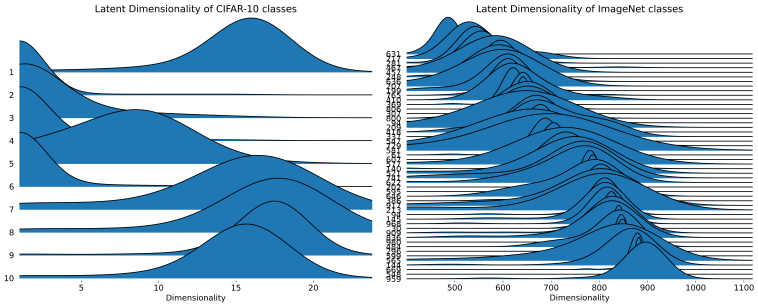
A3.9.2 The Dimension Distribution Mode Plot for CIFAR-10 AND ImageNet

Fig. A3.7 shows the dimension distribution mode plot across classes of CIFAR-10 and ImageNet. We show these plots for the *SparC-IB* compound (1,3) in CIFAR-10 and the *SparC-IB* compound (2,2) in ImageNet. In both data sets, we observe that each class prefers a different latent dimension (especially on ImageNet).



(a) (b) (c)

Figure A3.6 Mode plot for *SparC-IB* compound $(a, b) = (2, 2)$ across the 3 seeds on MNIST. We observe separation of posterior modes between the MNIST digits for all 3 seeds.



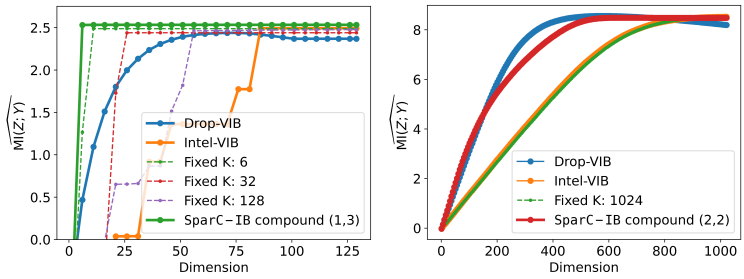
(a) Dimension mode plot for CIFAR-10

(b) Dimension mode plot for ImageNet.

Figure A3.7 Plot of the posterior modes of the dimension variable for (a) CIFAR-10 and (b) 50 randomly chosen classes of ImageNet. Both plot show separation between the latent dimension of the classes chosen by *SparC-IB*.

A3.9.3 Information Content Plot for CIFAR-10 and ImageNet

Fig. A3.8 shows the estimated MI$(Z;Y)$ (expression provided in Sec. 3.4.3.2) against the increasing dimension of the latent space for CIFAR-10 and ImageNet. In CIFAR-10, *SparC-IB* provides the most compact representation among the other models in which the information plateaus within the first dimensions (~ 5) of the latent space. For ImageNet, we observe that the information plateaus around dimension 500 which is smaller than the fixed-dimensional VIB and Intel-VIB models but higher than the Drop-VIB model. In addition, we note that the behavior of the estimated MI$(Z;Y)$ as a function of dimension is much smoother than those of the other two data sets. The reason for such behavior is perhaps the complexity in the ImageNet data, where it requires a high-dimensional latent space to encode the necessary information of X where each dimension's contribution is small.



(a) Information content vs dimension plot for CIFAR-10

(b) Information content vs dimension plot for ImageNet.

Figure A3.8 Plot of the information content vs dimension of the mean of the encoder for (a) CIFAR-10 and (b) ImageNet. Both plot show *SparC-IB* encodes the maximum information in a smaller dimensional latent space than other models.

A3.9.4 Calculating Pixel-wise Importance Scores for Dimension Visualization on MNIST

We have used the Captum package [15] to calculate the pixel importance scores to visualize the latent space (Sec. 3.4.3.3) for MNIST. Given an input image x and a baseline image x', the importance score for the i th pixel on the d th dimension of the mean of the latent space is calculated using the following expression.

$$\text{Importance Score}_i^d(x) = (x_i - x_i') \int_{\alpha=0}^{1} \frac{\partial \mu_d(\alpha x + (1-\alpha)x')}{\partial x_i} d\alpha$$

In the above expression, $\mu(.)$ is the mean vector of the latent space and $\mu_d(.)$ represents its d-th coordinate. We used a blank image where every pixel value is 0 as a baseline x'. We can interpret the score as the sensitivity of the dimensions of $\mu(x)$ to a small change in each pixel integrated on the images that fall on the line given by $\alpha x + (1-\alpha)x'$.

CHAPTER 4

PREDICTION OF THE POLYGENIC RISK SCORES USING DEEP LEARNING IN ANCESTRY-DIVERSE GENETIC DATA SETS

4.1 Introduction

Recent years have observed unprecedented increases in the sample size of biomedical data sets. Government initiatives (e.g. UK-Biobank, All of Us) and private organizations (e.g. 23andMe®) have generated genotype information linked to phenotypic and disease data on millions of individuals. These large-scale data sets have led to important advances in genomics research, including mapping of risk loci (aka Genome-Wide Association (GWA) analysis) and in developing models to predict phenotypes and disease risk using genomic information [11, 18]. In genetic studies, the prediction of a phenotype or disease risk is commonly made by polygenic scores (PGS) which are weighted sums of risk alleles at Single Nucleotide Polymorphism (SNP) that in a GWA study were found to be associated with a phenotype of interest [18]. Despite important advances in PGS prediction, many important challenges persist. One such challenge is the prediction of PGS in ancestry-diverse data sets which may involve heterogeneity of SNP effects between ancestry groups.

Most human genomic studies build PGS using linear models with SNPs as the covariates and the phenotype of interest as the response. A PGS is typically derived in two stages: (i) A GWA study is conducted to identify SNPs significantly associated with the outcome of interest using univariate regression, either a penalized or a Bayesian procedure (a form of sure independent screening [7]), then, (ii) The effects of the selected SNPs are estimated using a linear regularized regression [3]. With the increasing sample size in genetic data sets, studies have expanded the set of tools to include machine learning (ML) models for prediction to account for possible interactions between SNPs ([1],[2],[6],[13],[17], [16]). In some studies, ML models have shown gains in prediction accuracy compared to the linear model for some traits ([1],[17],[16]); however, other studies have not shown a consistent superiority of ML over traditional linear models ([2], [13]).

The majority of the Genome-Wide Association (GWA) and PGS studies used data from Eu-

ropeans [14]. Within homogeneous populations, linear models can capture a very large fraction of the genetic variance, even if the underlying genetic architecture involves complex gene-gene interactions. The reasons why linear models can provide very good approximations to genome-to-phenotype maps that are highly nonlinear have been studied extensively in quantitative genetics [9]. However, the approximation provided by a linear function has validity within the context (genetic background) where it was developed. Therefore, linear models often do not generalize well across populations because differences in allele frequencies, linkage disequilibrium patterns, and genetic-by-environment interactions between ancestry groups make the linear effects of SNPs vary between populations. Therefore, recent studies have emphasized utilizing ML models to capture such non-linearities in ancestry-diverse data sets. For instance, Elgart et al. [6] showed that an ensemble method using gradient-boosted regression trees performs better than traditional linear models in modeling PGS for a wide range of traits in a multi-ancestry data set.

In genetics, gene-by-gene interactions are referred to as *epistatic* effects. Although epistatic interactions can occur among distant loci, the literature on epistasis suggests that most gene-gene interactions are likely to happen between loci that are physically proximal in their genome position positions (e.g., between SNPs within the same gene). There are biological and statistical reasons that support the hypothesis that most epistatic interactions happen between SNPs that are physically proximal. For example, a mutation in a regulatory region may modulate the effect of mutations in a coding region. Likewise, two mutations within a gene may have larger effects on the resulting protein structure than a single mutation. Additionally, limitations of the genomic data available can lead to apparent epistasis (aka phantom epistasis [5]) even if the underlying causal model is linear. The notion that most SNP-by-SNP interactions may happen between SNPs that are physically proximal may be used to develop ML that are both statistically and computationally efficient. To the best of our knowledge, there are no ML models in the literature which aim to capture these local interaction effects.

Therefore, the objective of this study is to develop and investigate ML methods for PGS in the context of ancestry-diverse populations that can capture local epistatic interactions. To attain this

goal, we develop ML models with local neural network learners throughout the genome. Using the multi-ancestry Trans-Omics in Precision Medicine (TOPMed) [10], we empirically show that our approach significantly improves the prediction accuracy on small genomic segments throughout the genome.

Applying ML methods on a whole-genome scale is computationally challenging. Therefore, inspired by the encouraging results of the application of ML methods to short chromosome segments, and also on previous research [4] we propose a novel two-step procedure where local regressions are used to learn DNA-phenotype patterns for short chromosome segments (this step is fully parallelizable) and then combine the genomic scores of each chromosome segment into a single PGS. The proposed architecture (1) captures local interaction effects, and (2) is tuned to learn highly sparse and heterogenous signals throughout the genome. Using simulations, we benchmark the prediction performance of four specifications obtained by either using a Bayesian Linear Regression (BLR) or Machine Learning (ML) for the first and the second step of the procedure. Our results suggest that combining variable selection in the first step through a BLR followed by ML in the second step can improve prediction accuracy in cross-ancestry prediction relative to BLR or DL alone. This is a baseline study that serves as a step towards formalizing the ML application in genomic prediction.

4.2 Materials and Methods

To carry out our simulations we used genotype data from the TOPMed data set which contains whole-genome sequenced genomes that led to the calling of about 40 million SNPs across 23,078 samples from various ancestry groups. We choose this data set because it was derived from whole-genome sequencing (as opposed to genotyping based on SNP arrays), it has a relatively large sample size, and it is ancestry-diverse.

Training and Testing sets: We perform a principal component analysis on the SNPs and then apply K-means clustering to categorize the samples into 6 distinct population groups. Appendix Fig. A4.1 shows differences in the sub-populations in the plot of the first 2 principal components. Our focus is on developing methods that can leverage multi-ancestry data to improve the performance of polygenic prediction in non-Europeans who are severely underrepresented in genomics

research. Therefore, to test the cross-ancestry prediction accuracy of PGS, we select a held-out set of samples from sub-population 6 in Fig. A4.1 (which corresponds to individuals primarily of African ancestry) for testing the prediction accuracy and use the remaining samples for training.

Simulation set up: Our primary objective is to assess how ML techniques can capture non-linearity in the genotype-phenotype relationship. Most genomic data sets use SNP arrays with ~ 1 million SNPs. This represents a small fraction of all the SNPs in the human genome. In this context, most of the variants with causal effects are not genotyped. Therefore, in GWA studies and in PGS prediction, the SNPs available act as surrogates for the SNPs with causal effects. Mapping of causal loci and PGS prediction is possible because of linkage-disequilibrium (i.e., correlation) between the genotyped SNPs and those with causal effects. We leveraged the fact that TOPMed offers whole-genome sequence data to simulate phenotype data using randomly selected causal variants and then developed prediction models which excluded those causal variants. This strategy allowed us to represent a realistic scenario in which causal variants are not genotyped.

In the simulation study, there are many ways to introduce non-linearity in the relationship between the SNPs and the phenotype. The model used to simulate data (i.e., the causal model) may involve group-specific effects which may emerge due to epistasis or genetic-by-environment interactions. However, it is also known that even in cases where the model is linear at the causal level, the use of SNPs that are imperfect surrogates for the genotypes at causal loci may lead to non-linearities in the SNP-phenotype map [5].

Our objective was to benchmark ML methods in a challenging scenario that does not offer a clear advantage to ML models relative to linear PGS. Therefore, in all our simulations, following [5] we assume that the causal model is linear (with effects that are the same across ancestry groups) and then analyzed the data excluding the causal variants. In this scenario, non-linearities in the SNP-phenotype map may emerge because of heterogeneity in the correlation between the SNPs used for analysis across ancestry-diverse groups.

Let the causal model be defined by,

$$\boldsymbol{y} = \boldsymbol{Z}\boldsymbol{b} + \boldsymbol{e} \tag{4.1}$$

Here, $\boldsymbol{b} = [b_1, ..., b_S]'$ is the effect size vector , $\boldsymbol{Z} = [\boldsymbol{Z}_1, ..., \boldsymbol{Z}_S] \in \mathbb{R}^{n \times S}$ represents n samples of S causal loci (randomly selected among the available SNPs) and the errors $\boldsymbol{\epsilon} \in \mathbb{R}^n$ are chosen from iid Gaussian distribution with mean 0 and variance σ^2. In our simulations, we tuned the error variance to achieve different scenarios regarding the proportion of variance of the phenotype explained by genetic effects (aka heritability).

To develop prediction models we regressed the simulated phenotype ($\boldsymbol{y}$ from the causal model Eq. 4.1) using SNPs $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ that were not involved in the causal model ($\boldsymbol{X}$ does not include any of the variants included in the causal model Eq. 4.1). The instrumental model takes the form,

$$\boldsymbol{y} = f(\boldsymbol{X}) + \boldsymbol{\epsilon} \tag{4.2}$$

Here, $f(.)$ is any regression function, and $\boldsymbol{\epsilon} \in \mathbb{R}^n$ are model residuals. In ML, $f(.)$ is non-linear on its inputs; on the other hand, in the linear models that we used to benchmark ML $f(.) = \boldsymbol{X\beta}$.

Our simulations were based on SNPs in chromosome 1; this chromosome has a length of about 250 mega-base-pairs (mbp) and represents approximately 8.3% of the human genome. We first conducted a simulation based on short chromosome segments (up to 1 mega-base-pairs, mbp) and then extended this to a simulation that involve the entire chromosome.

Short-segment simulation: In this simulation, we applied Eq. 4.1 and Eq. 4.2 to short chromosome segments. In each Monte Carlo replicate, we sampled at random a position in chromosome 1 and then using the SNPs in a segment around that position, either a short one, 100 kilo-base-pairs (kbp) long, or a longer one (1,000 kbp=1 mbp) to simulate and analyze the data using Eq. 4.1 and Eq. 4.2 respectively. We tuned the error variance to achieve a proportion of variance explained by the causal loci (aka heritability) of either 0.2 or 0.4.

In our data, there are approximately 40 SNPs per kbp and 400 SNPs per mbp. In the simulation involving 100kbp segments, we assumed that 4 SNPs in the segment had effects and that the remaining 36 SNPs had no effects (i.e., at the causal level there was 90% of sparsity). In the simulation involving 1 mbp segments, we assumed that 16 of the 400 SNPs had effects (i.e., 96% sparsity).

	Chromosome 1									
SNPs	1-400	401-800	...	3601-4000	...	7601-8000	...	79601-80000	...	99600-100000

Figure 4.1 Map of chromosome divided into segments each containing 400 SNPs (1 mega-base-pairs). The segments highlighted (in green) each contains 1 causal SNP. There were 20 segments with one causal variant. Data were simulated using the 20 causal variants, according to Eq. 4.1 The causal SNPs were removed to implement the regression of Eq. 4.2.

Chromosome-wide simulation: In a second simulation we expanded our research to simulation and analysis involving the entire chromosome 1. Within the human genome, the correlation between SNPs typically decays with physical distance. In most populations, the correlation drops to values close to zero within roughly 1 mbp. Therefore, in the chromosome-wide simulation, we divide chromosome 1 into 250 segments each of width $\sim$ 400 SNPs (1 mbp) and randomly select 20 causal SNPs from distant segments. Similar to before, we use Eq. 4.1 and an assigned heritability of 0.2 to generate the phenotype. For illustration, Fig. 4.1 shows a graphical representation of 100,000 SNPs of chromosome 1 divided into segments, each with 400 SNPs. The highlighted segments (in green) each contain 1 causal SNP.

Analysis Methods: In genetics studies, a variety of machine learning models have been previously explored (see [13] for a review). In this study, we chose a feed-forward multi-layer perceptron neural network to model the data since it has shown promise in modeling local interaction patterns [17].

Single chromosome-segments simulation: For the segment-wise analysis (first simulation), we benchmark the performance of the ML model against (1) a Bayesian Linear Regression (BLR) with priors from the spike-slab family and (2) a BLR model with random SNP-ancestry groups interaction effect [19] [19] (i.e., a model that allows for the SNP effects to vary between ancestry-groups). Further details about the analysis methods are provided in the Appendix B4 section in the appendix.

Whole-chromsome simulation: It is computationally infeasible to fit a single model to the $\sim$ 100,000 SNPs of chromosome 1. Therefore, as suggested by [8], we apply all models in overlapping segments of 3 mbp (approximately 1200 SNPs). We then shift this window by 1 mbp,

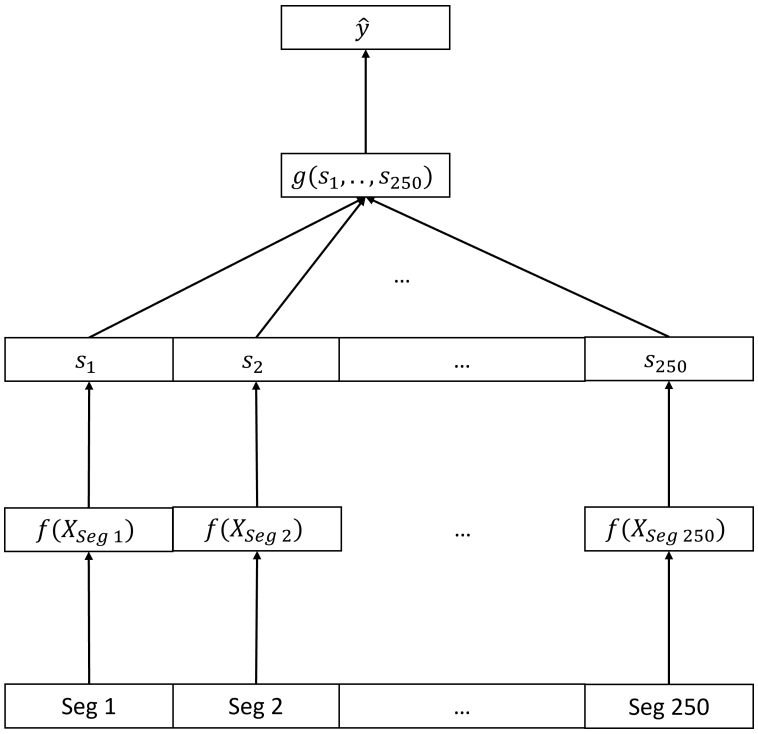


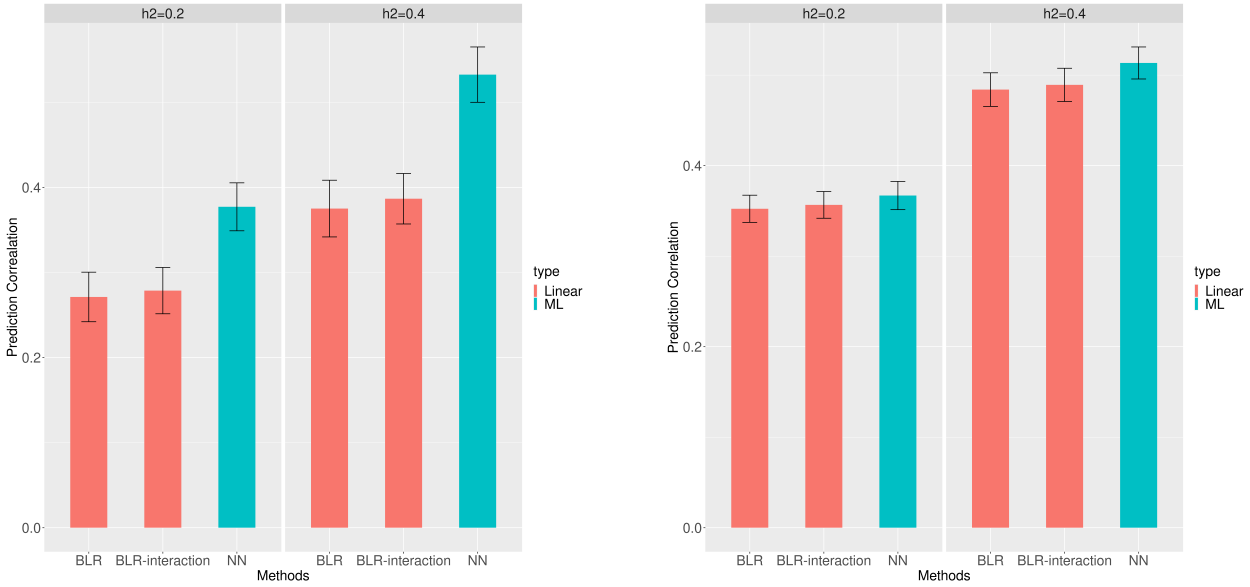
Figure 4.2 Schematic representation of the two-stage modeling of SNPs from chromosome 1 and a simulated phenotype ($\boldsymbol{y}$) in the whole-chromosome simulation.

First stage (local)	Second stage	Label
Linear	Linear	BLR + BLR
Linear	Neural network	BLR + NN
Neural network	Linear	NN + BLR
Neural network	Neural network	NN + NN

Table 4.1 Labels for the models used in the chromosome-wide analysis.

creating local regressions consisting of a 1 mbp core and two 1 mbp flanking regions on either side. Further details are provided in the Appendix B4 section.

We fit the chromosome-wide models using a two-stage procedure, (1) fit local regression models and get the predicted values separately on each segment, and (2) combine the scores to make the final prediction. Fig. 4.2 shows a schematic representation of the two-stage modeling where in the initial stage, SNPs from each segment passed through a function $f(.)$ to predict the scores $s_k, k = 1, ..., 250$. This first step was applied using the sliding window approach previously described, in parallel for each of the overlapping segments. This step produces a single score $f(.)$ for each segment. In the second stage, we combine the scores using another function $g(.)$ to make



(a) Prediction correlation on 100 kbp segments

(b) Prediction correlation on 1 mbp segments

Figure 4.3 The figure shows (a) ML clearly outperforms the linear models in terms of test set prediction correlation (error bars represent standard errors) in the scenario using 100 kilo-base-pairs (kbp) segments and simulated phenotypes. (b) ML performs similarly or slightly better compared to the baselines when applied to larger (1 mega-base-pairs) segments throughout the genome. **BLR**: Bayesian linear regression model, **BLR-interaction**: BLR model with random interaction effects, and **NN**: Neural network.

the final prediction $\hat{y}$. With the approach in Fig. 4.2, we choose from four models for building the PGS whereas at each stage, the candidate models (i.e. the choice of $f(.)$ and $g(.)$) considered are either a BLR model or a neural network. Table 4.1 summarizes the strategies we used for data analysis.

4.3 Results

Performance on *Short-segments* simulation: Fig. 4.3 shows the Pearson's correlation coefficient between the predicted and true phenotype on the test data set for the three models. For the simulation involving short (100kbp-long, Fig. 4.3 panel (a)) we observe that the ML model clearly outperforms the two linear models. This even though the model used to simulate data was linear. However, for a larger 1 mbp segment (Fig. 4.3 panel (b)), the difference in prediction correlation between linear models and the neural network was much smaller. Fig. 4.3(b) shows that the ML model is performing similarly to or marginally better than the baseline models for $h^2 = 0.2$ and

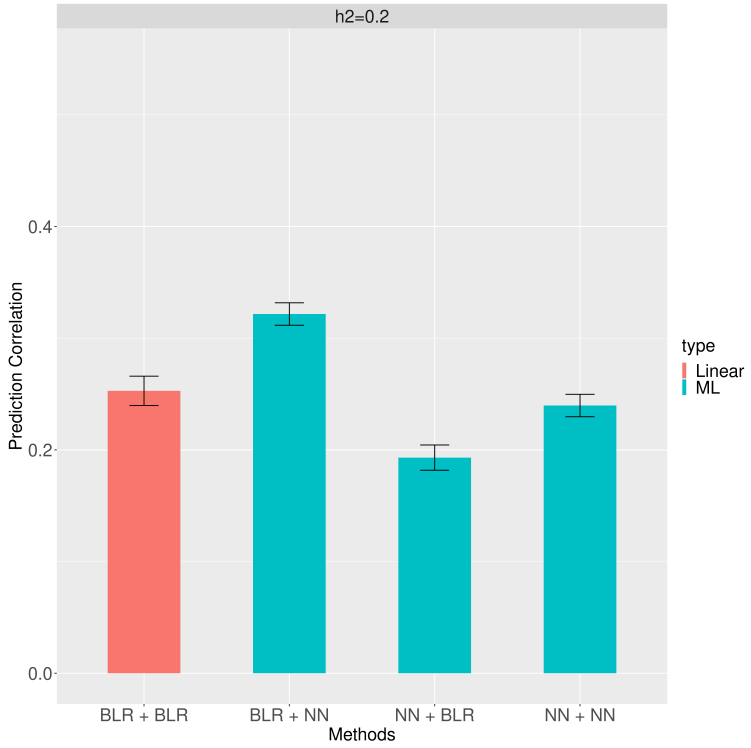


Figure 4.4 Chromosome-wide prediction accuracy (error bars represent standard errors) shows that local neural network methods do not perform as well as the linear model. The method based on linear models only (BLR+BLR) performs marginally better than the one based solely on NN (NN+NN). Using a linear (variable selection) model in the first stage and a NN in the second stage (BLR+NN) was the best-performing method. **BLR + BLR**: BLR in both the stages, **BLR + NN**: BLR in the first stage and neural network in the second stage, **NN + BLR**: Neural network in the first stage and BLR in the second stage, and **NN + NN**: Neural network in both the stages.

0.4 respectively. Furthermore, we observe that the linear models in Fig. 4.3(b) show significantly better results than panel (a) where the ML model maintains a similar prediction correlation. For large segments, the prediction accuracy of linear models improves because longer segments can leverage the long-range LD between SNPs. As one would expect, we observe that higher h^2 (i.e., higher signal-to-noise ratio) led to better prediction performance for all models.

Performance on *chromosome-wide* simulation: Fig. 4.4 shows the prediction correlation (averaged over 50 bootstrap samples in the test data set) of the models in the chromosome-wide simulation. We observe that all models are significantly less accurate compared to Fig. 4.3. This was expected because the signal is distributed over many more SNPs ($\sim$ 99% sparsity), therefore, posing a greater challenge for the models to detect them. In addition, we observe that the prediction

accuracy decreases when the neural network is applied in the first stage (NN + BLR and NN + NN) compared to the linear model (BLR + BLR and BLR + NN). This behavior is primarily because the simulated signal is sparse (many segments did not have effects, see Fig. 4.1) and neural networks do not perform as well as the BLR (which uses a sparsity-inducing prior) model at separating the segments with the signal from the rest. The best prediction accuracy is achieved by using a linear model in the first stage and a neural network in the second stage (**BLR + NN** model).

4.4 Discussion

In this study, we proposed strategies for the use of ML approaches for constructing polygenic scores on an ancestry-diverse data set. Over the past decade, machine learning research has seen tremendous growth in all fields of science. However, in Genetics, despite the age of big data ushering in important advancements, the application of ML has seen limited success. With the future increase in data collected on populations from diverse ancestry (e.g. All of Us), we highlight the importance of ML compared to linear models in constructing cross-population PGS that achieves better generalization across populations.

Modern machine-learning applications in PGS prediction have reported mixed results. However, recent studies ([6],[16]) have shown promise for ML in genomic prediction in the domain of cross-population PGS construction. Although our objective is similar, we show that ML models gain significant improvement in prediction accuracy over baseline models when applied to local genomic segments (Fig. 4.3(a)). This demonstrates the potential impact of local epistatic interactions [5], which lead to decreased accuracy of the linear model.

The chromosome-wide application shows that the prediction accuracy decreases when using the local neural network approach in the first stage of the two-stage modeling. This corroborates with the findings of earlier studies [2] which show the performance of the neural network deteriorates when there are excessive covariates that have no effect on the output. However, using a BLR (for the first stage) followed by NN to combine effects yielded better results than simply using NNs or linear models.

This study provides numerous opportunities for future research in ML applications in genomic

prediction. The proposed two-stage approach allows for the application of this method on a whole-genome scale. This is because the first step can be fully parallelized, and the second step operates in a reduced dimension using local polygenic scores inferred in the first stage. As noted earlier, for the two-stage method in the chromosome-wide simulation, the most effective approach was the one using a variable selection model in the initial stage, and a NN in the second stage. This outcome presents a chance to create techniques that can produce reliable predictions while being computationally feasible. The initial step, which is computationally expensive, can be done in parallel and using linear models that are less expensive and simpler to implement than an ML model.

Therefore, our next objective will be to expand this approach to whole-genome simulations, evaluating it with real data, and, finally developing software that could implement these methods and a user-friendly, computationally efficient framework. One possible future research is to explore machine learning methods that incorporate linear dimension reduction (e.g. random projections [12]) of the input data into their architecture, or neural networks that induce sparsity on the input-output map.

In conclusion, this study provides insights into the application of ML in capturing epistatic interactions in modern ancestry-diverse data sets. Our research indicates that ML models have the potential to capture the local non-linearity that occurs as a result of imperfect linkage disequilibrium (LD) between SNPs. As a result, we have developed a two-stage framework, based on local polygenic scores, that is effective in constructing PGS on large chromosome-wide data sets. With the help of standard software and computational resources available to researchers, this framework can be applied genome-wide to construct PGS on underrepresented population groups using ancestry-diverse ultra-high dimensional genetic data sets.

BIBLIOGRAPHY

[1] Alzoubi, H., Alzubi, R., and Ramzan, N. (2023). Deep learning framework for complex disease risk prediction using genomic variations. *Sensors*, 23(9):4439.

[2] Bellot, P., de Los Campos, G., and Pérez-Enciso, M. (2018). Can deep learning improve genomic prediction of complex human traits? *Genetics*, 210(3):809–819.

[3] Choi, S. W., Mak, T. S.-H., and O'Reilly, P. F. (2020). Tutorial: a guide to performing polygenic risk score analyses. *Nature protocols*, 15(9):2759–2772.

[4] de Los Campos, G., Grueneberg, A., Funkhouser, S., Pérez-Rodríguez, P., and Samaddar, A. (2023). Fine mapping and accurate prediction of complex traits using bayesian variable selection models applied to biobank-size data. *European Journal of Human Genetics*, 31(3):313–320.

[5] de Los Campos, G., Sorensen, D. A., and Toro, M. A. (2019). Imperfect linkage disequilibrium generates phantom epistasis (& perils of big data). *G3: Genes, Genomes, Genetics*, 9(5):1429–1436.

[6] Elgart, M., Lyons, G., Romero-Brufau, S., Kurniansyah, N., Brody, J. A., Guo, X., Lin, H. J., Raffield, L., Gao, Y., Chen, H., et al. (2022). Non-linear machine learning models incorporating snps and prs improve polygenic prediction in diverse human populations. *Communications Biology*, 5(1):856.

[7] Fan, J. and Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):849–911.

[8] Funkhouser, S. A., Vazquez, A. I., Steibel, J. P., Ernst, C. W., and Los Campos, G. d. (2020). Deciphering sex-specific genetic architectures using local bayesian regressions. *Genetics*, 215(1):231–241. 32198180[pmid].

[9] Hill, W. G., Goddard, M. E., and Visscher, P. M. (2008). Data and theory point to mainly additive genetic variance for complex traits. *PLoS genetics*, 4(2):e1000008.

[10] Kowalski, M. H., Qian, H., Hou, Z., Rosen, J. D., Tapia, A. L., Shan, Y., Jain, D., Argos, M., Arnett, D. K., Avery, C., et al. (2019). Use of > 100,000 nhlbi trans-omics for precision medicine (topmed) consortium whole genome sequences improves imputation quality and detection of rare variant associations in admixed african and hispanic/latino populations. *PLoS genetics*, 15(12):e1008500.

[11] Lewis, C. M. and Vassos, E. (2020). Polygenic risk scores: from research tools to clinical instruments. *Genome medicine*, 12(1):1–11.

[12] Liu, Z., Bhattacharya, S., and Maiti, T. (2022). Variational bayes ensemble learning neural networks with compressed feature space. *IEEE Transactions on Neural Networks and Learning*

Systems.

[13] Lourenço, V. M., Ogutu, J. O., Rodrigues, R. A., and Piepho, H.-P. (2022). Genomic prediction using machine learning: A comparison of the performance of regularized regression, ensemble, instance-based and deep learning methods on synthetic and empirical data. *bioRxiv*, pages 2022–06.

[14] Mills, M. C. and Rahal, C. (2019). A scientometric review of genome-wide association studies. *Communications biology*, 2(1):9.

[15] Pérez, P. and de Los Campos, G. (2014). Genome-wide regression and prediction with the bglr statistical package. *Genetics*, 198(2):483–495.

[16] Sigurdsson, A. I., Ravn, K., Winther, O., Lund, O., Brunak, S., Vilhjalmsson, B. J., and Rasmussen, S. (2022). Improved prediction of blood biomarkers using deep learning. *medRxiv*, pages 2022–10.

[17] Sigurdsson, A. I., Westergaard, D., Winther, O., Lund, O., Brunak, S., Vilhjálmsson, B. J., and Rasmussen, S. (2021). Deep integrative models for large-scale human genomics. *bioRxiv*, pages 2021–06.

[18] Torkamani, A., Wineinger, N. E., and Topol, E. J. (2018). The personal and clinical utility of polygenic risk scores. *Nature Reviews Genetics*, 19(9):581–590.

[19] Veturi, Y., de Los Campos, G., Yi, N., Huang, W., Vazquez, A. I., and Kühnel, B. (2019). Modeling heterogeneity in the genetic architecture of ethnically diverse groups using random effect interaction models. *Genetics*, 211(4):1395–1407.

APPENDIX A4

TOPMED DATA SET

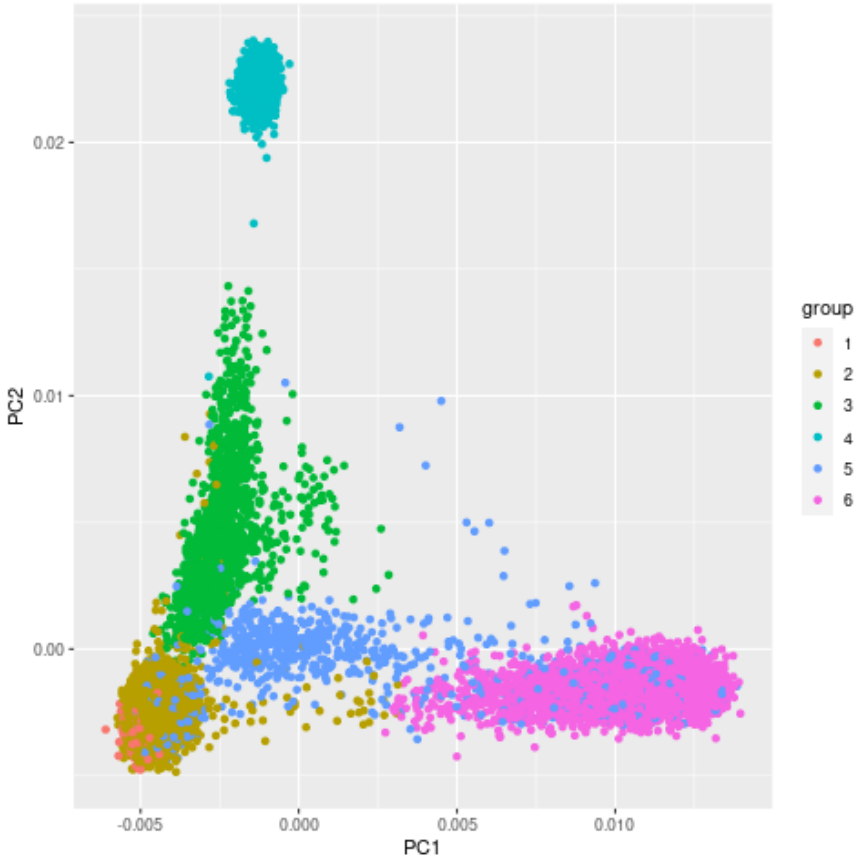


Figure A4.1 Plot of first two principle components of the SNPs shows clear population stratifications.

APPENDIX B4

SUPPLEMENTARY METHODS

B4.1 Local regression

In this section, we demonstrate the local regression approach used to model the whole chromosome data using a sequence of 1600 SNPs. We divide the whole set into 4 segments each of size 400 SNPs ($\sim$ 1 mbp). Fig. B4.1 shows the 400 core SNPs and flanking regions for the four local regression models. In the chromosome-wide application, the local regressions are used to calculate scores for each segment and then combined in the second stage. To calculate the scores using the BLR model, following [8], we only use the effects from the core region.

B4.2 Bayesian linear model

Let us assume, a linear regression model with $\boldsymbol{y}$ as the response and $\boldsymbol{X}$ as the covariate matrix is of the form,

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \qquad \epsilon_i \overset{iid}{\sim} \mathrm{N}(0, \sigma^2_\epsilon) \tag{B4.1}$$

where $\boldsymbol{y} = (y_1, y_2, ..., y_n)'$ is a vector of phenotypes, $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_p)$ is an $n \times p$ SNPs matrix, $\boldsymbol{\beta}$ is the p-dimensional vector of regression coefficients, $\boldsymbol{\epsilon}$ is a vector of iid Gaussian error terms, $\sigma^2_\epsilon > 0$ is the common error variance.

Prior We chose a Bayesian Variable Selection model with IID priors from the spike-and-slab family on the regression coefficient of Eq. B4.1 that includes a point mass at 0 and a Gaussian slab. The formulation of the prior is as follows,

$$p(\beta_j|\pi, \sigma^2_b) \overset{iid}{=} \pi \mathrm{N}(0, \sigma^2_b) + (1 - \pi)\mathbb{1}(\beta_j = 0)$$

$$\pi \sim beta(a, b)$$

$$\sigma^2_b \sim \chi^{-2}(df_b, S_b)$$

$$\sigma^2_\epsilon \sim \chi^{-2}(df_\epsilon, S_\epsilon)$$

Here, χ^{-2} denotes the scaled-inverse chi-square distribution.

SNPs	1-400	401-800	801-1200	1201-1600
	Model fit			
Fit 1	Core 1	Flank 1R		
Fit 2	Flank 2L	Core 2	Flank 2R	
Fit 3		Flank 3L	Core 3	Flank 3R
Fit 4			Flank 4L	Core 4

Figure B4.1 The local regression models fitted on a sequence of 1600 SNPs.

Hyperparameters	Values
$\frac{a}{a+b}$	$\frac{1}{10}$
$a+b$	110
iterations	10000
burn-in	2000

Table B4.1 Table showing hyperparameter values for the BLR model.

Hyperparameters We used the R package BGLR [15] to draw samples from the posterior. The Gibbs sampler in BGLR requires important hyperparameters such as the number of iterations and the number of burn-in steps. We set the hyperparameters according to Table B4.1.

B4.3 Bayesian linear model with group-specific interaction

Following [19], we define the Bayesian linear model with group-specific interaction effects. In addition to the main SNPs effects, we introduce *SNPs* × *population dummy* effects in the model. For the simplicity of notation, we describe the model assuming 2 populations,

$$\begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{1}\mu_1 \\ \mathbf{1}\mu_2 \end{bmatrix} + \begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \end{bmatrix} \boldsymbol{\beta}_0 + \begin{bmatrix} \boldsymbol{X}_1 \\ \mathbf{0} \end{bmatrix} \boldsymbol{\beta}_1 + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{X}_2 \end{bmatrix} \boldsymbol{\beta}_2 + \begin{bmatrix} \boldsymbol{\epsilon}_1 \\ \boldsymbol{\epsilon}_2 \end{bmatrix} \tag{B4.2}$$

Here, $\boldsymbol{\beta}_0 = \{\beta_{0j}\}_{j=1}^{p}$ are the main effects and $\boldsymbol{\beta}_i = \{\beta_{ij}\}_{j=1}^{p}, i = 1, 2$ are the interactions effects for populations 1 and 2. We assume the spike-and-slab prior described in the previous section on the regression coefficients of Eq. B4.2. The effects are estimated using BGLR software [15] with the values of the hyperparameters from Table B4.1.

Hyperparameters	Segments	Chromosome-wide
Train set size	22,778	22,578
Test set size	300	500
# epochs	100	100
Training batch size	128	128
Optimizer	Adam	Adam
Initial learning rate	1e-4	1e-4
Learning rate drop	0.6	0.6
Learning rate drop steps	10 epochs	10 epochs

Table B4.2 Hyperparameter settings used for modeling the local segments and the entire chromosome.

B4.4 Neural network model

For the simplicity of notation, we first define a shallow neural network model. Let, the output of a shallow neural network is $\hat{\boldsymbol{y}}$ then the model is defined as

$$\hat{\boldsymbol{y}} = b_0 + \sum_{j=1}^{K} b_j \psi\Big(\gamma_{j0} + \sum_{h=1}^{p} \gamma_{jh} \boldsymbol{x_h}\Big) \tag{B4.3}$$

Here, the input $\boldsymbol{X}$ is passed through one layer with K hidden nodes before making the prediction $\hat{\boldsymbol{y}}$. The parameter vectors $\{\gamma_{j0}, \gamma_{jh}\}_{h=1}^{p}$ denotes the biases and the weights for the j-th node of the hidden layer and $\{b_0, b_j\}_{j=1}^{K}$ denotes the biases and weights used to combine the output of the hidden layer for the final prediction. The non-linearity is introduced in the network by the activation function $\{\psi\}$. Deep neural networks expand shallow neural networks to have more than one hidden layer. We omit the full equation of deep neural networks for brevity.

Model architecture In this study, we use feed-forward multi-layer perceptron (MLP) neural networks. In Sec. 4.3, we have discussed two purposes of applying the neural networks, (1) local regression on segments, and (2) combining the scores of the local models. For local models, we considered an encoder-decoder architecture where both the encoder and decoder are shallow neural networks Eq. B4.3 with ReLU activations. In all our simulations, we set the number of hidden nodes $K = 128$. For combining the local regression scores, we used a single and much smaller shallow neural network with $K = 50$. All models were fitted using early stopping on 10-fold cross-validation data splits of the training set and the ensemble of the 10 models was taken as the final model.

Fitting deep learning models involves several key hyperparameters. In Table B4.2, we provide the necessary hyperparameters for the training and evaluation of all fitted models.

CHAPTER 5

CONCLUSION

This dissertation made significant strides in the field of machine learning, particularly in the realm of supervised learning encompassing both classification and regression tasks. We developed scalable machine learning methods and explored their practical application for feature selection and prediction in large-scale genetics data, resulting in several noteworthy discoveries:

Chapter 1 introduced the Bayesian Hierarchical Hypothesis Testing (BHHT) algorithm, which significantly enhanced fine-mapping tasks in Genome-wide association studies. By applying BHHT to identify risk variants for serum-urate traits using data from UKBiobank, the method demonstrated an impressive increase in discoveries by capturing highly correlated SNPs that conventional variant-centered procedures failed to detect.

Chapter 2 addressed the improvement of a deep latent variable model, the information bottleneck, by introducing the novel sparsity-inducing spike-slab categorical prior (SparC-IB). Experiments on MNIST, CIFAR-10, and ImageNet data showcased the superior prediction accuracy and robustness of the proposed approach compared to traditional fixed-dimensional priors and other sparsity induction mechanisms for latent variable models found in the literature. SparC-IB's ability to enable data-point-specific dimensionality learning further demonstrated the successful separation of latent dimensions between image classes.

In Chapter 3, we focused on developing machine learning methods for constructing polygenic scores (PGS) for ancestry-diverse datasets. Using the TOPMed genotype data on short chromosome segments and simulated phenotypes, we show that the local neural network learners improve cross-ancestry polygenic risk scores (PGS) prediction over state-of-the-art Bayesian variable selection models. Inspired by this, we propose a two-step procedure that enables expanding the use of deep learning for whole-genome PGS prediction. Our simulation results show that a two-step procedure that involves inference of linear local scores for individual chromosome segments, later combined into a single PGS using a neural network provides better prediction performance than linear models or models based on neural networks only.

In summary, the thesis contributes valuable insights into the realm of machine learning methodologies for feature selection and prediction in the context of large-scale genetics data. The methods and algorithms that we developed hold the potential to advance genetic research and pave the way for improved precision in understanding complex genetic traits and their associations.