Representation Retrieval Learning for Heterogeneous Data Integration

Qi Xu[*]   Annie Qu[†]

Abstract

In the era of big data, large-scale, multi-modal datasets are increasingly ubiquitous, offering unprecedented opportunities for predictive modeling and scientific discovery. However, these datasets often exhibit complex heterogeneity—such as covariate shift, posterior drift, and missing modalities—that can hinder the accuracy of existing prediction algorithms. To address these challenges, we propose a novel Representation Retrieval (R^2) framework, which integrates a representation learning module (the *representer*) with a sparsity-induced machine learning model (the *learner*). Moreover, we introduce the notion of "integrativeness" for representers, characterized by the effective data sources used in learning representers, and propose a *Selective Integration Penalty* (SIP) to explicitly improve the property. Theoretically, we demonstrate that the R^2 framework relaxes the conventional full-sharing assumption in multi-task learning, allowing for partially shared structures, and that SIP can improve the convergence rate of the excess risk bound. Extensive simulation studies validate the empirical performance of our framework, and applications to two real-world datasets further confirm its superiority over existing approaches.

Keywords: Block-missing Data; Multi-task Learning; Multi-modality Data; Representation learning.

---

[*]Department of Statistics and Data Science, Carnegie Mellon University, Email: qixu@andrew.cmu.edu
[†]Department of Statistics, University of California, Irvine, Email: aqu2@uci.edu

1 Introduction

Large-scale data integration has made transformative contributions across numerous fields, including computer vision, natural language processing, biomedicine, genomics and healthcare. For example, in biomedicine, integrating randomized clinical trials and observational studies is of great interest, as it leverages the benefits of both data sources [63, 39, 5]. In genomics, multi-modality and multi-batch assays enable the discovery of cellular heterogeneity and development [21, 7]. In healthcare, multiple types of time-series measurements, such as cardiovascular, physical activities, and sleep data are integrated to improve real-time health and well-being monitoring [68, 44, 38]. However, integration of large-scale data effectively remains challenging, particularly when data are collected from diverse sources or populations, and across various collections of variables and modalities.

In particular, integrating large-scale data is challenging primarily due to various types of heterogeneity. First, the marginal distribution of the same covariate is often heterogeneous across different sources or populations, a phenomenon called "distribution heterogeneity", or "covariate shift" in the literature [40]. Second, in the context of supervised learning, the conditional distribution of responses given covariates could be heterogeneous, which is named "posterior heterogeneity", or "posterior drift" [56]. Third, observed covariates or modalities are often not uniformly measured: some covariates are observed across all data sources, while others are observed in only partial data sources. We refer to this as "observation heterogeneity" or "block missing", which is considered in existing works [65, 62, 4].

In the current literature, various problem setups related to integrative supervised learning have been studied, while most of them only concern one or two types of the aforementioned heterogeneity. In particular, distribution heterogeneity, posterior heterogeneity, or both are considered in multi-task learning or transfer learning [56, 25, 47, 48]. Observation heterogeneity has been studied in multi-source data integration [65, 62, 61]. Recent work [4] has addressed all three types of heterogeneity in the transfer learning problem; however, their distributional and linear model assumption restrict their applicability for more general contexts. Sui et al. [43] propose a deep learning-based method to handle all three types of heterogeneity, where one modality is

required to be observed among all data sources, which is restrictive in practice. In this work, we target the integrative supervised learning problem and aim to improve the predictive performance for all data sources. Our framework can accommodate all three types of heterogeneity and allow for nonparametric modeling for complex association between covariates and responses. Indeed, incorporating all three types of heterogeneity within a unified framework would allow for the integration of much broader datasets, thereby enhancing prediction performance by leveraging substantially more information.

1.1 Related works

Over the last two decades, numerous methods and techniques have emerged to address these three types of heterogeneity, either individually or in combination. We review some seminal works in this subsection.

Distribution heterogeneity has attracted much attention in data integration. Primarily, it is important to discover the information sharing structure which enables one to capture common information across data sources. In particular, the joint-and-individual structure [27, 46] has been widely adopted, assuming a jointly shared component by all data sources and individual components unique to each data source. This structure is also applied to latent factor regression [53], canonical correlation analysis [26] and multi-task linear regression [9]. Recently, the partially sharing structure was developed [14, 28, 60] to accommodate more flexible information sharing structure. Specifically, each component can be shared by any number of data sources, which empower a greater heterogeneity level. Indeed, this information sharing structure is more common and natural in real applications (see motivating examples in Section 1.2). There are two techniques to discover a partially sharing structure: first, partially sharing structure can be encoded by binary matrices, which can be estimated via forward selection [28], or two-step selection [14]. Both estimations are computationally infeasible for high-dimensional or large-scale data. Second, a sparse-induced penalty [49, 41] is adopted to recover the partially sharing structure [31, 24, 60], which reduces the estimation complexity due to its convexity.

Posterior heterogeneity has been extensively studied in the literature on transfer

learning and multi-task learning. There are various structural assumptions imposed on model parameters. For instance, regression coefficients across tasks are assumed to be similar, which can be achieved by a fusion-type penalty [45], or shrinking model parameters to a prototype [9]. Other works consider more complex structures, such as latent representation [48], or angular similarity [17]. In these parametric models, parameters are stringent to be comparable across data sources, which implies that models for all sources belong to the same function class. This could be restrictive when different sources exhibit heterogeneous patterns. For example, the effects of covariates can be highly nonlinear to responses for partial data sources, while linear models can be good enough for other data sources. In this scenario, existing methods either underfit some data sources with complex patterns or overfit some data sources with simplistic patterns. Therefore, it is crucial to incorporate varying degree of complexities for different data sources in the framework to capture distinct patterns in a data-driven fashion.

Observation heterogeneity, such as the block-missing pattern has usually been treated independently. Some existing works focus on imputing missing blocks [3, 69] by leveraging low-rank structure. Other works target supervised learning with the presence of missing blocks [66, 65, 62, 64, 42, 43]. In particular, [65, 62, 42] focus on linear model prediction, inference and model selection, where [62] integrate multiple imputation for missing blocks via estimating equations, and [65, 42] directly estimate regression coefficients via covariance-regularized regression [57]. Among these, a uniform function mapping is assumed across data sources, which ignores potential distribution and posterior heterogeneity. However, this assumption could be easily violated, for example, in the motivating example in Section 1.2. In these scenarios, existing methods are incapable of capturing the heterogeneity, so their performance can be even worse than fitting models for each sources individually.

1.2 Motivating example

In this subsection, we elaborate the three types of aforementioned heterogeneity with a concrete example from the Alzheimer's Disease Neuroimaging Initiative* (ADNI) database. The data includes four sources, where each source exhibits a different obser-

vation pattern, shown in Figure 1. In total, three modalities are collected as covariates: magnetic resonance imaging (MRI), positron emission tomography (PET) and gene expression (Gene). The Mini Mental State Examination (MMSE) score is used as a response variable to measure the cognitive impairment of patients.
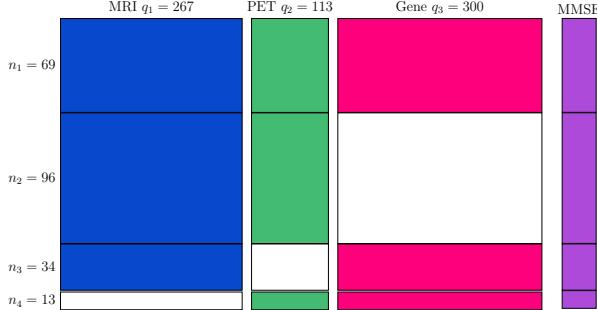

Figure 1: Block missing structure of ADNI data. White blocks are missing modalities.

In this study, we aim to build an accurate prediction algorithm to predict the MMSE score given three modalities of covariates. Given the apparent observation heterogeneity, existing methods for multi-task learning are not applicable to this dataset. [65, 62] have analyzed this dataset by assuming a uniform linear model across data sources. In the following, we examine the homogeneity of the marginal distribution of covariates and conditional distribution of responses given covariates across different data sources.

In order to examine the homogeneity of the marginal distribution of each modality, we apply sparse PCA (Witten et al. [58], sparsity is necessary here because the dimension of covariates is greater than sample sizes for all modalities and data sources) to project each modality into 2-dimensional space; and the density of each data source can be found in Appendix C. The density plots of the MRI modality have apparent discrepancy across the first three data sources, which suggests potential covariate shift.

Next, we examine the posterior homogeneity across four data sources. Since the true models are unknown to us and [65, 62] adopted a linear model for their analysis, we calculate the marginal correlation between each covariate and response across four sources, and evaluate whether these marginal correlations are homogeneous across sources. Figure 12 in Appendix C presents the plots of marginal correlations for covariates among three modalities, respectively. In the MRI and Gene modalities, marginal correlations across data sources are quite heterogeneous, where many features exhibit opposite marginal correlation to responses across data sources. Therefore, posterior

heterogeneity should be considered when analyzing this dataset.

1.3 Contributions

Our method offers several significant contributions. Methodologically, we introduce the Representation Retrieval (R^2) framework, which constructs a dictionary of representers—such as neural networks [30], kernels [20], or smoothing function bases [37]—to capture the complex distribution across multiple data sources. For each data source, a sparse learner built upon this dictionary selectively retrieves the most informative representers for prediction. Notably, the R^2 framework flexibly accommodates partially shared structures among data sources via a sparsity-inducing penalty.

Moreover, we introduce the concept of "integrativeness" of representers, defined as the effective data sources utilized for learning representers. To directly encourage the integrativeness of representers, we propose an innovative Selective Integration Penalty (SIP). Theoretically, we derive an excess risk bound for the R^2 framework, explicitly controlled by the integrativeness of representers, thereby demonstrating that SIP effectively enhances the model's generalization performance. Computationally, we develop an efficient alternating minimization algorithm to iteratively update both the representer dictionary and the sparse learners. Extensive simulation studies and real-world applications further support the superior performance of our proposed method.

1.4 Organization

The rest of paper is structured as follows: In Section 2, we introduce the notations used throughout the paper and elaborate the problem setup in this work. In Section 3.1, we first present the Representation Retrieval (R^2) learning for multi-task learning problem. The novel selective integration penalty is introduced in Section 3.2. The extension to Block-wise Representation Retrieval (BR^2) learning is elaborated in Section 3.3. Theoretical guarantees of the proposed method are provided in Section 4. Afterwards, we investigate the empirical performance of our R^2 and BR^2 learning with simulated data in Section 5, and apply our method to two real data examples to demonstrate its superior performance in Section 6. Summary and discussion are provided in Section 7.

6

2 Preliminaries

In this section, we introduce our problem setup with essential notations. The related background is also introduced as a prelude to the proposed method.

In this paper, we focus on integrating multi-source data for supervised learning problems, such as classification and regression. Suppose we have S sources of data collected: $\mathcal{D}_s = (\mathbf{X}^{(s)}, \mathbf{y}^{(s)})$, where $\mathbf{X}^{(s)} \in \mathcal{X}^{(s)} \subset \mathbb{R}^{p_s}$ denotes covariates or features, and $\mathbf{y}^{(s)} \in \mathcal{Y}^{(s)}$ denotes responses or outcomes. Note that $\mathcal{Y}^{(s)}$'s are source-specific spaces to accommodate heterogeneous responses, for example, $\mathcal{Y}^{(s)} = \{1, 2, ...,\}$ for classification problems and $\mathcal{Y}^{(s)} \in \mathbb{R}$ for regression problems. Note that $\mathcal{X}^{(s)}$'s can have different dimensions in covariates space due to observation heterogeneity. Specifically, different sets of covariates or modalities can be collected for different sources, where some covariates are observed across multiple data sources, while some covariates are only observed in certain data sources. To make it manageable, we consider the multi-modalities covariates, where M modalities of covariates are collected across S sources. A straightforward illustration of our setup is provided in the upper block of Figure 2. As noted in the Introduction, our problem setup encompasses a broad spectrum of problem setups, including multi-task learning and block-wise missing learning as special cases. Our target is to learn function mappings $f^{(s)} : \mathbf{X}^{(s)} \rightarrow \mathbf{y}^{(s)}$ for $s = 1, \cdots, S$. Although it is valid to learn individual function mapping for each data source, here we seek to integrate shared information across different data sources to strengthen predictive performance for each source.

2.1 Multi-task representation learning

In the machine learning community, representation learning has received considerable attention due to its powerful and flexible capacity to learn complex and non-linear patterns. A comprehensive review can be found in [2]. For supervised learning settings, a typical procedure following representation learning is:

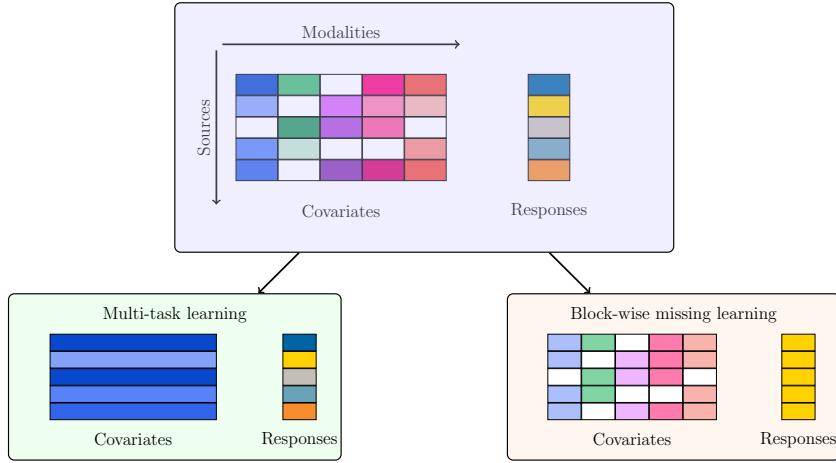$$\mathbf{X} \underbrace{\xrightarrow{\theta} \mathbf{h} \xrightarrow{g}}_{f} \mathbf{y},$$

Figure 2: Targeted problem setup and special cases. The upper block illustrates our problem setup: different colors represent different covariates modalities, and white blocks denote missing modalities. Different shade levels of the same color indicate different distributions of the same covariates. The lower left block shows the multi-task learning setup, which considers only one covariate modality and ignores observation heterogeneity. The lower right block illustrates block-wise missing learning, involving multi-modality data with missing blocks without considering distribution and posterior heterogeneity.

where θ is a latent representation algorithm (*representer*) which projects the original data into a latent space, and g is a regression or classification algorithm (*learner*) which establishes the association between latent representation and responses. In other words, the function f is a composition of g and θ, denoted by $f = g \circ \theta$. This learning paradigm is a building block in deep learning for many applications. The *representer* θ is usually a complex neural net to uncover the nonlinear, entangled features, which allows a simple *learner* g to achieve desirable empirical performance. In multi-task representation learning [8, 51, 54], each task is associated with a unique $g^{(s)}$, so the task-specific function mapping is formulated as $f^{(s)} = g^{(s)} \circ \theta$. From the information sharing perspective, this framework only assumes a jointly shared component, ignoring the distribution heterogeneity across tasks openly. In our work, we employ representation learning to capture complex and non-linear patterns, and propose a new framework to address the drawback in multi-task representation learning.
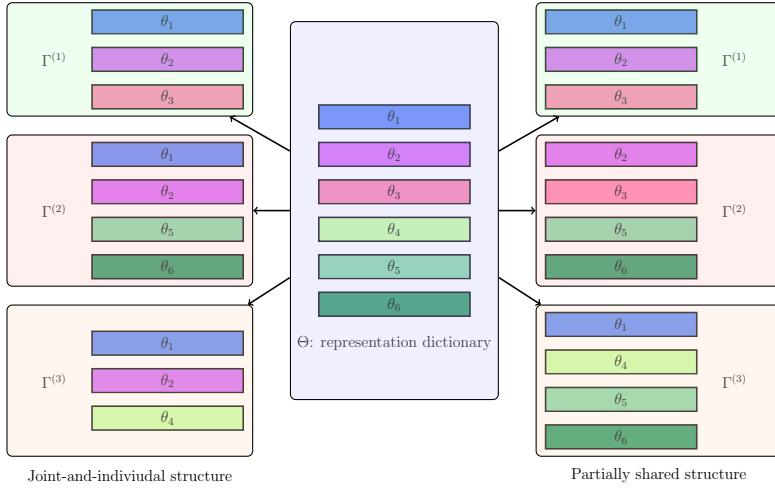
Figure 3: The left three data sources follow joint-and-individual structure, where θ_1 and θ_2 are shared by all three sources, and other representers are unique to certain sources. The right three data sources follow partially shared structure, where each representer is retrieved by some of the data sources. Both structures can be accommodated by the Representation retrieval framework.

3 Representation retrieval framework

3.1 Representation Retrieval Learning

We first consider integrating multi-source datasets with the same observed covariates for supervised learning. This is equivalent to the multi-task learning setting incorporating both distribution and posterior heterogeneity, shown in Figure 2. Therefore, we use the terms "tasks" and "sources" interchangeably in this section.

We consider the representation dictionary as a set of representers, denoted as $\Theta = \{\theta_1, \cdots, \theta_D\}$. Formally, we define a representer as a univariate function mapping $\theta_d : \mathcal{X} \to \mathbb{R}$, so we use θ_d and $\theta_d(\mathbf{x})$ interchangeably in later discussion. Multivariate representer outputs are also allowed, where the related extension can be found in Appendix A. For each task, we assume that a subset of informative representers from Θ are retrieved, denoted as $\Gamma^{(s)} = \{\theta_{d_1}, \cdots, \theta_{d_s}\}$. Naturally, we define $\Gamma^{(s)}(\mathbf{x}) = (\theta_{d_1}(\mathbf{x}), \cdots, \theta_{d_s}(\mathbf{x})) : \mathcal{X} \to \mathbb{R}^{|\Gamma^{(s)}|}$. Since we do not impose any structural assumptions on $\Gamma^{(s)}$, both joint-and-individual or partially sharing structures are incoporated as shown in Figure 3.

In order to learn the function mappings $f^{(s)}$, we build the task-specific learners upon the retrieved representers as in multi-task representation learning. Suppose $g^{(s)}(\cdot) =$

9

$\langle \cdot, \alpha^{(s)} \rangle$ is a linear function, then each function mapping $f^{(s)}$ is formulated as

$$f^{(s)}(\cdot) = \langle \Gamma^{(s)}(\cdot), \alpha^{(s)} \rangle, \quad \alpha^{(s)} \in \mathbb{R}^{|\Gamma^{(s)}|}. \tag{3.1}$$

Mathematically, the above model is equivalent to

$$f^{(s)}(\cdot) = \langle \Theta(\cdot), \beta^{(s)} \rangle = \sum_{d=1}^{D} \beta_d^{(s)} \theta_d(\cdot), \quad \beta^{(s)} \in \mathbb{R}^{|\Theta|} \text{ and } \|\beta^{(s)}\|_0 = |\Gamma^{(s)}|. \tag{3.2}$$

Formulations (3.1) and (3.2) are equivalent because $\beta^{(s)}$ is a sparse coefficient vector which amounts to select $|\Gamma^{(s)}|$ representers from Θ. In (3.1), both representers and learners are task-specific, where $\Gamma^{(s)}$ involves the representer retrieval and $\alpha^{(s)}$ is a task-specific regression coefficient. In comparison, only sparse regression coefficients $\beta^{(s)}$'s are task-specific in (3.2). By this means, we cast the representation retrieval framework as a model selection problem, which has been extensively studied [49, 70, 50, 67]. Additionally, more complex learners, such as neural nets, can be adopted to our framework, considering the sparse input features. We refer readers to [55, 12, 23, 10] for recent developments in sparse-input neural nets. To keep our introduction concise, we use linear learners in the following discussion.

To estimate the representers and learners, and pursue the sparsity in $\beta^{(s)}$'s, we minimize the following loss function for regression problems:

$$\min_{\Theta, \beta^{(1)}, \cdots, \beta^{(S)}} \sum_{s=1}^{S} \frac{1}{n_s} \sum_{i=1}^{n_s} \|y_i^{(s)} - \langle \Theta(\mathbf{x}_i^{(s)}), \beta^{(s)} \rangle\|_2^2 + \sum_{s=1}^{S} \lambda^{(s)} \|\beta^{(s)}\|_1. \tag{3.3}$$

By default, we consider representers θ_d as feed-forward neural nets [16] in this work. Note that $\lambda^{(s)}$ controls the level of sparsity, so varying values of $\lambda^{(s)}$'s are adaptive to each data source. The optimization problem in (3.3) can be solved alternatively between Θ and $\beta^{(s)}$'s: Given a fixed Θ, minimizing (3.3) is equivalent to solving Lasso problem for each data source; given fixed $\beta^{(s)}$'s, the dictionary Θ can be updated through a gradient-based algorithm. Therefore, the computation of our method is more efficient than optimizing over binary matrices as in [14, 28] to recover the partially sharing structure. To handle the classification problem, we can replace the least square loss with cross entropy loss without further modification.

In fact, the proposed model (3.2) covers many existing methods as special cases. For each single task, model (3.2) amounts to sparse multiple kernel learning [20] if θ_d corresponds to different kernels. In the simple scenario that $\theta_d(\mathbf{x}) = \mathbf{x}_d$ and $D = p$, our model is equivalent to well-studied sparse linear models for multi-task learning [29, 18, 6]. If θ_d's are linear mappings, our model is exactly a dictionary learning model for multi-task learning [31]. Our proposal is also closely related to mixture-of-expert for multi-task learning [30], where each representer can be treated as an expert.

Compared to the existing methods mentioned above, selecting neural nets in our R^2 framework unlocks greater representational capacity, as demonstrated by numerous empirical studies in different fields. Due to the non-convexity of neural nets, minimizing (3.3) can yield multiple local minimizers of Θ with similar loss. This naturally leads to the question:

Which local minimizer of Θ is more favorable than others in terms of generalization error?

In what follows, we introduce a new notion for representers that is critical for controlling the generalization error, as we will show in Section 4. Indeed, this notion of representers applies to any representers, not restricted to neural nets.

3.2 Integrativeness and selective integration penalty

In this section, we introduction the notion of *integrativeness* of the representer dictionary. Intuitively, data integration happens when the same representer is retrieved in $\beta^{(s)}$ by multiple tasks. Motivated by this, we define the integrativenss of a representer as follows:

$$\gamma_d = \sum_{s=1}^{S} \mathbb{I}(\beta_d^{(s)} \neq 0), \tag{3.4}$$

where $0 \leq \gamma_d \leq S$ counts the number of data sources that retrieves the representer θ_d for prediction. Theoretically, we show that more integrative representers lead to the sharper generalization bound in Theorem 4.1. However, the optimization problem in (3.3) does not necessarily favor more integrative representers. Therefore, we introduce additional structural assumptions in the R^2 framework for better control of integrativeness to
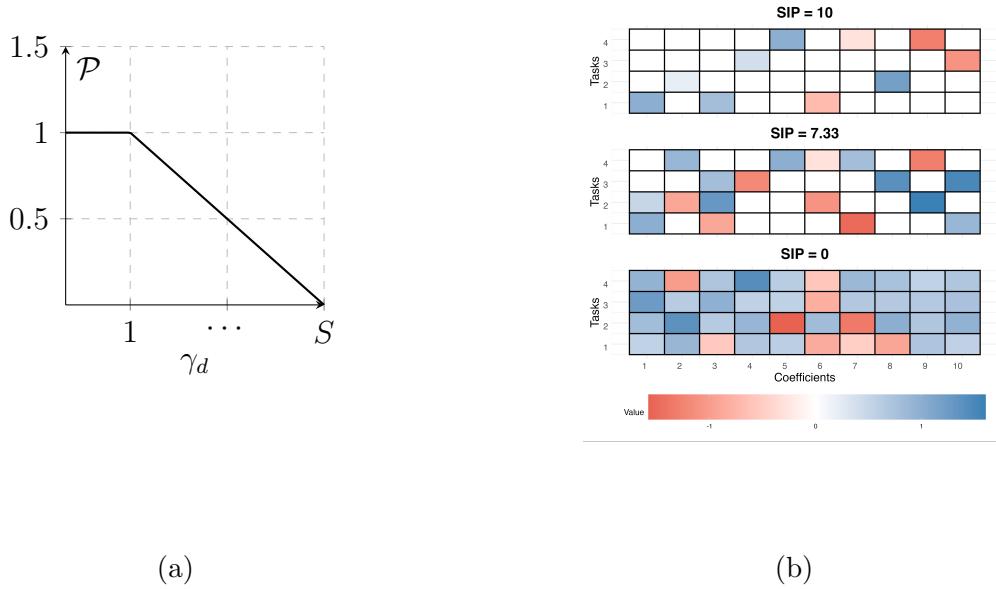
(a)　　　　　　　　　　　　　　　　　　(b)

Figure 4: (a): Visualization of SIP formula (3.5) for a single γ_d. (b): Illustrations of regression coefficients $\beta_d^{(s)}$ under $\mathcal{P} = D$, $0 < \mathcal{P} < D$ and $\mathcal{P} = 0$ scenarios. In this example, we consider $D = 10$ representers, and $S = 4$ tasks.

further improve the prediction.

Formally, we propose the following Selective Integration Penalty (SIP) to encourage the overall integrativeness of representers:

$$\mathcal{P}(\beta^{(1)}, \cdots, \beta^{(D)}) = \sum_{d=1}^{D} \min(1, \frac{S - \gamma_d}{S - 1}), \tag{3.5}$$

where $0 \leq \mathcal{P}(\cdot) \leq D$, and the SIP for a single γ_d is visualized in Figure 4(a). For each single γ_d, SIP truncates at $\gamma_d = 1$, in that the penalty is identical to 1 for $0 \leq \gamma_d \leq 1$; since any $0 \leq \gamma_d \leq 1$ do not integrate any information across tasks, therefore they should be penalized at the same level. When $\mathcal{P}(\cdot) = 0$, it indicates that $\beta_d^{(s)} \neq 0$ for all d and s, showing that all the representers are retrieved by all S data sources. On the contrary, if $\mathcal{P}(\cdot) = D$, then $\gamma_d \leq 1$ for all d, suggesting that each representer is retrieved at most once. In other words, different tasks do not share any common representers. In addition, if $0 < P(\cdot) < D$, a partially sharing structure is encoded in $\beta^{(s)}$. An illustrative example are shown in Figure 4(b), offering comparisons of these three different scenarios. In order to optimize the SIP, we adopt the truncated-ℓ_1

12

function [34, 59] as a continuous relaxation of the indicator function in (3.4):

$$\tilde{\gamma}_d(\tau) = \sum_{s=1}^{S} \min(1, \frac{|\beta_d^{(s)}|}{\tau}),$$

where $0 \le \tilde{\gamma}_d(\tau) \le S$. Based on the above derivation, the loss function of our R^2 learning is

$$\min_{\Theta, \beta^{(1)}, \cdots, \beta^{(S)}} \sum_{s=1}^{S} \frac{1}{n_s} \sum_{i=1}^{n_s} \|y_i^{(s)} - \langle \Theta(\mathbf{x}_i^{(s)}), \beta^{(s)} \rangle\|_2^2 + \sum_{s=1}^{S} \lambda_1^{(s)} \|\beta^{(s)}\|_1 + \lambda_2 \mathcal{P}_\tau(\beta^{(1)}, \cdots, \beta^{(D)}),$$

(3.6)

where $\mathcal{P}_\tau(\cdot)$ is the smoothed SIP by replacing γ_d with $\tilde{\gamma}_d(\tau)$. Here, two penalties play different roles: the ℓ_1 penalty in the second term of (3.6) targets each individual task, in order to select informative representers for each specific task. SIP, in the third term of (3.6) acts on regression coefficients across tasks, and seek to improve the overall integrativeness of representers. We employ the Adam [19] optimizer in `pytorch` to minimize the (3.6), and select the hyper-parameters via random search.

Our proposed R^2 framework with SIP offers unique advantages to multi-task learning. Specifically, we allow practitioners to choose flexible representers that are adaptive to their data structure. Since our SIP is defined directly on learners, representers can be different models with varying numbers of parameters. For example, linear mappings can capture linear structures of original data, while neural nets can capture high-order nonlinear structures, and we can combine linear and nonlinear representers in the dictionary as representer candidates for different data sources. This is advantageous for multi-source data where each data type could have varying complexity of structures. In contrast, most existing integrative penalties [45, 9, 48] require model parameters comparable across tasks to apply the penalties, which could be stringent in multi-source data integration. In particular, when multi-source data have different modalities (e.g, some of them have images, while others have text data), it is ineffective to employ the same model for all sources.

3.3 Block-wise Representation Retrieval Learning

In this section, we extend our R^2 learning to multi-modality data with block-missing structures, which addresses the challenge from observation heterogeneity.

For ease of our discussion, we first introduce the essential notations for multi-source multi-modality block-missing data. In particular, we denote covariates of the mth modality of the sth source as $\mathbf{X}_m^{(s)} = (\mathbf{x}_{1;m}^{(s)T}; \mathbf{x}_{2;m}^{(s)T}; \cdots, \mathbf{x}_{n_s;m}^{(s)T}) \in \mathbb{R}^{n_s \times q_m}$ for $m = 1, \cdots, M$ and $s = 1, \cdots, S$. The missing indicator is denoted by $\mathbb{I}_m^{(s)}$, where $\mathbb{I}_m^{(s)} = 1$ if the mth modality is observed in the sth source, and $\mathbb{I}_m^{(s)} = 0$ otherwise. In addition, let $\mathbf{X}^{(s)}$ contain all covariate modalities from the sth source: $\mathbf{X}^{(s)} = \{\mathbf{X}_m^{(s)} | m : \mathbb{I}_m^{(s)} = 1\}$, and we use $\mathbf{X}_m = \{\mathbf{X}_m^{(s)} | s : \mathbb{I}_m^{(s)} = 1\}$ to denote the observed data from the mth modality. The problem setup with notations is shown in Figure 5.
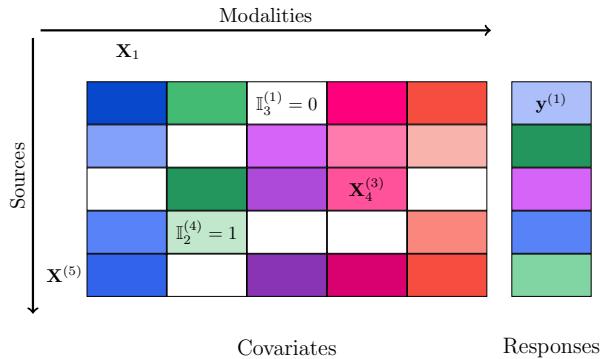

Figure 5: An example of multi-source multi-modality block-missing data with $S = 5$ sources and $M = 5$ modalities. White blocks are missing modalities. Some examples of notations are provided.

For multi-modality data with block-missing structures, the missingness patterns could be rather complicated, making imputation approaches inaccurate. First of all, it is unlikely that a block-missing structure is missing completely at random (MCAR) since the probability of missing indicator $\mathbb{P}(\mathbb{I}_m^{(s)} = 0)$ is dependent on the modality and the data source. Second, the distribution heterogeneity could lead to different conditional distributions, which invalidates multiple imputation approaches. Taking Figure 5 as an example, one may impute $\mathbf{X}_4^{(4)}$ via conditional distribution $\mathbb{P}(\mathbf{X}_4 | \mathbf{X}_5)$, which can be learned from data source $s = 1, 2, 5$. However, due to the distribution heterogeneity across sources, any of the probability $\mathbf{P}(\mathbf{X}_4^{(s)} | \mathbf{X}_5^{(s)})$ may not be equal to $\mathbf{P}(\mathbf{X}_4^{(4)} | \mathbf{X}_5^{(4)})$. Therefore, multiple imputation via conditional distribution learned from

other data sources could be inapplicable to the target source, which introduces bias in imputation and diminishes the predictive performance in downstream prediction. Similarly, an imputation approach relying on conditional distributions can fail more easily in the missing not at random (MNAR) scenario due to the distribution heterogeneity. Therefore, imputation approaches are inappropriate for multi-modality block-missing data given significant distribution heterogeneity.

To address the above challenges, we propose the *Block-wise Representation retrieval* (BR^2) learning, which directly learns the function mappings from observed multi-modality covariates to the responses without missing imputations. Instead of imputing missing blocks and applying R^2 learning to the imputed data, we aim to borrow as much information as we can given the observed data. Specifically, we introduce multiple representer dictionaries for different modalities or combinations of modalities, to represent the observed covariates into latent space, and then adopt source-specific learners to establish connections between latent representations and responses. In particular, we introduce modality-specific representer dictionaries, denoted as $\Theta_m = \{\theta_{m;1}, \cdots, \theta_{m;D_m}\}, m = 1, \cdots, M$, and each representer is a function mapping $\theta_{m;d} : \mathcal{X}_m \to \mathbb{R}$ as in R^2 learning. Given this setup, data sources $s \in \mathcal{O}_m = \{s : \mathbb{I}_m^{(s)} = 1\}$ share the mth dictionary, which enables us to borrow information across data sources for each modality. The function mapping $f^{(s)} : \mathcal{X}^{(s)} \to \mathcal{Y}^{(s)}$ is then modeled as

$$f^{(s)}(\cdot) = \sum_{m=1}^{M} \mathbb{I}_m^{(s)} \langle \Theta_m(\cdot), \beta_m^{(s)} \rangle, \quad \beta_m^{(s)} \in \mathbb{R}^{|\Theta_m|} \text{ and } \|\beta_m^{(s)}\|_0 = |\Gamma_m^{(s)}|, \qquad (3.7)$$

where $\Gamma_m^{(s)}$ is the subset of retrieved representers from the mth modality dictionary Θ_m for the sth data source. In (3.7), the missing indicator $\mathbb{I}_m^{(s)}$ indicates that if the mth modality is unobserved in the sth data source, then the mth modality has no impact on the response. Like linear models, dictionaries Θ_m's represent the main effects of the mth modality to responses. Optionally, we can also introduce interaction dictionaries $\Theta_{m_1, m_2, \cdots, m_w}$, where $m_1, \cdots, m_w \in \{1, \cdots, M\}$ to represent the $w-$way interaction among modalities. Similarly, each interaction dictionary $\Theta_{m_1, \cdots, m_w}$ is associated with sparse learner $\beta_{m_1, \cdots, m_w}^{(s)}$ for each source. For the ease of notation, we exclude interaction dictionaries in the following estimation. Similar to the R^2 learning, the representers

and learners can be estimated by optimizing the following loss function:

$$\min_{\Theta,\beta} \sum_{s=1}^{S} \frac{1}{n_s} \sum_{i=1}^{n_s} \|y_i^{(s)} - \sum_{m=1}^{M} \mathbb{I}_m^{(s)} \langle \Theta_m(\mathbf{x}_{i;m}^{(s)}), \beta_m^{(s)} \rangle\|_2^2 + \sum_{s=1}^{S} \sum_{m=1}^{M} \lambda_{m;1}^{(s)} \|\beta_m^{(s)}\|_1$$

$$+ \sum_{m=1}^{M} \lambda_{m;2} \mathcal{P}_\tau(\beta_m^{(1)}, \cdots, \beta_m^{(S)}). \quad (3.8)$$

Note that the SIP for the mth modality is only effective for data sources in $\mathcal{O}_m$, which improves the integrativeness level of representers in Θ_m. If $|\mathcal{O}_m| = 1$, that is, only one data source observes the mth modality, then the SIP does not apply to this modality. Compared with supervised learning for individual sources ignoring the missing blocks, our BR^2 learning improves the integrativeness of each modality-specific representations that borrow information from other sources observing the same modalities, which performs at least comparably well to learning with individual sources. In comparison, supervised learning based on imputed values could diminish the predictive power if the missing mechanism is mis-specified, which may perform even worse than learning with individual sources. Empirical comparisons among these methods are provided in Section 5.2.

4 Theoretical Property

4.1 Excess risk bound for R^2 learning

In this subsection, we establish the theoretical properties of the proposed R^2 learning in the multi-task learning problem. In particular, we precisely characterize the impact of partially sharing structures on the average generalization error over all data sources, and justify the necessity of the Selective Integration Penalty (SIP) in data integration. First of all, we specify the function class considered in the R^2 learning:

$$f^{(s)}(\mathbf{X}^{(s)}) = \sum_{d=1}^{D} \theta_d(\mathbf{X}^{(s)}) \beta_d^{(s)}, s \in [S],$$

where $\Theta \in \mathcal{F} = \{(\theta_1, \cdots, \theta_D), \theta_d \in \mathcal{F}_d, d \in [D]\}$, and $\mathcal{F}_d$'s are certain function classes for representers, such as linear function or neural nets with norm constraints. In ad-

16

dition, we consider the sparse linear function class $\beta^{(s)} \in \mathcal{H}^{(s)} = \{\beta, \|\beta\|_1 \leq \alpha_s\}$ for $s \in [S]$. Throughout this section, we use $[n]$ to denote the set $\{1, \cdots, n\}$ for $n \in \mathbb{R}_+$. Across all $\beta^{(s)}$, we consider the function class $\mathcal{H} = \{(\beta^{(1)}, \cdots, \beta^{(S)}) | \beta^{(s)} \in \mathcal{H}^{(s)}, \breve{\gamma}(\tau) \geq \eta(\tau), \bar{g}(\tau) \leq \pi(\tau)\}$ where $\breve{\gamma}(\tau) = \frac{1}{D} \sum_{d=1}^{D} \tilde{\gamma}_d(\tau)$, the average continuous approximation of γ_d across data sources, and $\bar{g}(\tau) = \frac{1}{D} \sum_{d=1}^{D} \sum_{s=1}^{D} \mathbb{I}(0 < |\beta_d^{(s)}| < \tau)$, upper bound of the approximation error $\frac{1}{D} \sum_{d=1}^{D} (\gamma_d - \tilde{\gamma}_d)$. These two terms are implicitly determined by λ_2 and α_s's are implicitly controlled by $\lambda_1^{(s)}$ in (3.6). The function classes $\mathcal{F}$ and $\mathcal{H}$ characterize the partially sharing structures across tasks in that each task only retrieves some, but not necessarily all representers from the dictionary, which relaxes the fully sharing structure studied in [51, 54]. In order to estimate Θ and $\beta^{(s)}$'s, we minimize the following empirical risk over function class $\Theta \in \mathcal{F}, \beta \in \mathcal{H}$:

$$\hat{\mathcal{R}}(\Theta, \beta) = \frac{1}{S} \sum_{s=1}^{S} \frac{1}{n_s} \sum_{i=1}^{n_s} \ell\left\{ y_i^{(s)}, \sum_{d=1}^{D} \theta_d(\mathbf{x}_i^{(s)}) \beta_d^{(s)} \right\}.$$

In each task, we observe n_s random samples $(\mathbf{x}_i^{(s)}, y_i^{(s)})$ from distribution $\mu^{(s)}$. For simplicity, we assume $n = n_1 = \cdots = n_S$ in the following theoretical development. Further, the empirical risk minimizers are denoted as $\hat{\Theta}$ and $\hat{\beta}$, respectively. Analogously, the population risk is defined as

$$\mathcal{R}(\Theta, \beta) = \frac{1}{S} \sum_{s=1}^{S} \mathbb{E}_{(\mathbf{X}^{(s)}, \mathbf{y}^{(s)}) \sim \mu^{(s)}} \left[\ell\left\{ \mathbf{y}^{(s)}, \sum_{d=1}^{D} \theta_d(\mathbf{X}^{(s)}) \beta_d^{(s)} \right\} \right],$$

and Θ^*, β^* denote the minimizers of population risk. Our main focus is the excess risk between the empirical risk minimizer and the population risk minimizer:

$$\mathcal{E}(\hat{\Theta}, \hat{\beta}) = \mathcal{R}(\hat{\Theta}, \hat{\beta}) - \mathcal{R}(\Theta^*, \beta^*),$$

which quantifies the average generalization error across all tasks. Before we proceed to the theoretical results, we introduce some essential assumptions:

Assumption 4.1. *(a) (Lipschitz loss) The loss function: $\ell(\cdot, \cdot)$ is B-bounded, and $\ell(y, \cdot)$ is L-Lipschitz for all $y \in \mathcal{Y}$.*

(b) (Boundedness) The representer is bounded for $d \in [D]$: $\sup_{\mathbf{x} \in \mathcal{X}} |\theta_d(\mathbf{x})| \leq B$ for

any $\theta_d \in \mathcal{F}_d$.

Remark 4.1. *Assumptions (a) and (b) are standard assumptions in learning theory to control the complexity of function classes, which can be satisfied with bounded $\mathcal{Y}$ and $\mathcal{F}$ under common adopted loss functions, such as least square loss and cross entropy loss.*

As demonstrated in Section 3.2, integrativeness is a desirable property for representers in the dictionary. Recall that $\gamma_d = \sum_{s=1}^{S} \mathbb{I}(\beta_d^{(s)} \neq 0)$ counts the number of tasks retrieving the dth representer, and $\tilde{\gamma}_d(\tau)$ is a continuous relaxation of γ_d adopted by SIP. In the following, we show that the complexity of the representer dictionary is directly impacted by η and π.

Lemma 4.1. *For any $\beta = (\beta^{(1)}, \cdots, \beta^{(S)}) \in \mathcal{H}$ and fixed realizations $(\mathbf{x}_i^{(s)}, y_i^{(s)})_{i=1}^n, s \in [S]$, we define*

$$\mathbf{G}_\beta(\boldsymbol{\sigma}) = \sup_{\Theta \in \mathcal{F}} \frac{1}{nS} \sum_{s=1}^{S} \sum_{i=1}^{n} \sigma_{s,i} \left[\sum_{d=1}^{D} \theta_d(\mathbf{x}_i^{(s)}) \beta_d^{(s)} \right],$$

where $\sigma_{s,i}$ are i.i.d Rademacher random variables, then we have:

$$\mathbb{E}_{\boldsymbol{\sigma}} \mathbf{G}_\beta(\boldsymbol{\sigma}) \leq \frac{\alpha^2}{\tau} \sqrt{\frac{C_\Theta}{n \breve{\gamma}(\tau)}} + \alpha \sqrt{\frac{C_\Theta \bar{g}(\tau)}{n \breve{\gamma}^2(\tau)}} \tag{4.1}$$

$$\leq \frac{\alpha^2}{\tau} \sqrt{\frac{C_\Theta}{n \eta(\tau)}} + \alpha \sqrt{\frac{C_\Theta \pi(\tau)}{n \eta^2(\tau)}}, \text{ for any } \tau > 0, \tag{4.2}$$

where $\alpha = \max_s \alpha_s$. The constant C_Θ is the uniform upper bound $C_\Theta \geq C_{\theta_d}$ for all $d \in [D]$, where $\mathbb{E}\left\{ \sup_{\theta_d \in \mathcal{F}_d} \frac{1}{n} \sum_{i=1}^{n} \sigma_i \theta_d(\mathbf{x}_i) \right\} \lesssim \sqrt{\frac{C_{\theta_d}}{n}}$.

The quantity $\mathbb{E}_{\boldsymbol{\sigma}} \mathbf{G}_\beta(\boldsymbol{\sigma})$, analogous to the Rademacher complexity, quantifies the capacity of the representation dictionary to fit random noise for any fixed $\beta \in \mathcal{H}$. A similar measure has been studied for linear dictionaries in [31]. In Lemma 4.1, we refine the analysis by leveraging the support of the given $\beta^{(s)}$. See Appendix D for technical details. The upper bound (4.1) only considers the ℓ_1 penalty on $\beta^{(s)}$'s where $\breve{\gamma}(\tau)$ and $\bar{g}(\tau)$ are functions of β, which provides a precise description of the role of "integrativeness" of representers in average generalization error bound. In contrast, the upper bound (4.2) is an uniform bound for any $\beta \in \mathcal{H}$, which is useful to establish the results in the following Theorem 4.1.

18

The upper bound (4.1) in Lemma 4.1 consists of two terms. The first term bounds the complexity of Θ for tasks with $\tau \leq |\beta_d^{(s)}| \leq \alpha$, while the second term bounds the complexity for tasks with $0 < |\beta_d^{(s)}| < \tau$. In both cases, the dependence on τ is complicated, however, since the first term of (4.1) is monotonically decreasing in τ and the second term of (4.1) is monotonically increasing in τ. Therefore, an infimum of $\mathbb{E}_{\boldsymbol{\sigma}}[G_\beta(\boldsymbol{\sigma})]$ exists with respect to $0 \leq \tau \leq \alpha$. Empirically, τ should be tuned in a data-driven manner. Instead, if we ignore the approximation of $\tilde{\gamma}_d(\tau)$ to γ_d, the upper bound can be shown as $\alpha^2 \sqrt{\frac{C_\Theta}{n\bar{\gamma}}}$ where $\bar{\gamma} = \frac{1}{D}\sum_{d=1}^{D} \gamma_d$. Therefore, if $\tau = O(1)$, then the second term $\alpha\sqrt{\frac{C_\Theta \bar{g}(\tau)}{n\bar{\gamma}^2(\tau)}}$ is the cost of approximation from $\tilde{\gamma}_d(\tau)$.

For any fixed $\tau = O(1)$, the second term in (4.1) decays faster than the first term when $\breve{\gamma}(\tau) \to \infty$ as $S \to \infty$. Therefore, the complexity of the representation dictionary dominantly decays in the order of $O(n^{-1/2}\breve{\gamma}^{-1/2}(\tau))$. In particular, when $\breve{\gamma}(\tau) = S$, implying all tasks retrieve all representers in Θ and the magnitudes of $\beta_d^{(s)}$ are all greater than τ, we obtain $\mathbb{E}_{\boldsymbol{\sigma}}\mathbf{G}_\beta(\boldsymbol{\sigma}) \leq O((nS)^{-1/2})$, aligning with Lemma 11 in [31]. Conversely, when $\breve{\gamma}(\tau) = 1$, implying no representers are shared across tasks, the bound simplifies to $\mathbb{E}_{\boldsymbol{\sigma}}\mathbf{G}_\beta(\boldsymbol{\sigma}) \leq O(n^{-1/2})$, indicating the absence of any advantage from multi-task learning. In other words, it degenerates to the single task learning scenario.

Note that C_Θ is a constant related to the complexity of function classes for the representation dictionary, which depends on the choice of θ_d in the implementation. Typically, a richer function class is usually endowed with a greater C_Θ. We refer readers to [1, 33, 15] for the Rademacher complexity of norm-constrained neural nets. In our R^2 framework, we allow users to specify different function classes for different representers, resulting in varying C_{θ_d}. Given that C_{θ_d}'s can be intricate and mostly loose for complex nonlinear models, such as neural nets, we apply the same function class for θ_d in numerical experiments.

In the following, we obtain the excess risk bound for our R^2 learning:

Theorem 4.1. *Suppose Assumption 4.1 holds, and the dictionary size is fixed as D. Let $(\mathbf{x}_i^{(s)}, y_i^{(s)})$ be identically and independently distributed random samples following distribution $\mu^{(s)}$ defined over $\mathcal{X} \times \mathcal{Y}$, then for empirical risk minimizer $\hat{\Theta}$ and $\hat{\beta}$, with*

probability $1 - \delta$, *we have*

$$\mathcal{E}(\hat{\Theta}, \hat{\beta}) \leq 4L\frac{\alpha^2}{\tau}\sqrt{\frac{C_\Theta}{n\eta(\tau)}} + 4\alpha L\sqrt{\frac{C_\Theta \pi(\tau)}{n\eta^2(\tau)}} + 4\alpha BL\sqrt{\frac{2\log(2D)}{n}} + B\sqrt{\frac{2\log(2/\delta)}{nS}}.$$

(4.3)

To the best of our knowledge, Theorem 4.1 is the first excess risk bound that precisely characterizes the impact of partially shared representations on excess risk bound in the multi-task learning literature. In the upper bound, the first two terms in (4.3) are from Lemma 4.1, quantifying the cost of learning the representation dictionary. The third term of (4.3) arises from learning $\beta^{(s)}$ for each task, which scales with $O(n^{-1/2})$. And the last term of (4.3) signifies the cost of learning from random samples and converges fast as either n or S increases.

Theorem 4.1 shows the advantages of R^2 learning over single-task learning. In supervised representation learning, the generalization error is typically determined by the cost of learning from both represents and learners. Our results demonstrate that the cost of learning represents can be reduced when they are shared across even a subset of tasks, while the cost of learning learners remains at the same rate as in single-task learning due to its task-specific nature.

Theorem 4.1 also elucidates the roles of the ℓ_1 penalty and the SIP term in (3.6). Specifically, a larger λ_2 for $\mathcal{P}_\tau(\beta^{(1)}, \ldots, \beta^{(S)})$ in (3.6) corresponds to a larger $\eta(\tau)$ and smaller $\pi(\tau)$, resulting in an improved generalization error bound. For a rich function class $\mathcal{F}$ for represented dictionary, one can approximate $f^{(1)}, \ldots, f^{(S)}$ using multiple $\Theta \in \mathcal{F}$ and $\beta \in \mathcal{H}$ with similar approximation error. Our SIP pursues a more integrative $\Theta \in \mathcal{F}$, leading to a tighter generalization error bound. Furthermore, a larger $\lambda_1^{(s)}$ in the ℓ_1 penalty in (3.6) corresponds to a smaller α_s, which also improves the generalization error bound. Consequently, both penalties are crucial for controlling the complexity of the representation dictionary and average excess risk across tasks. Moreover, these penalties also influence the representation power in multi-task learning. A larger α_s permits $f^{(s)}$ to be approximated using a richer set of represents, while smaller values of λ_2 facilitate more flexible retrieval patterns. In summary, the ℓ_1 penalty and the SIP term jointly control the trade-off between representation capacity and generalization

error. Empirically, we suggest tuning the associated hyperparameters using independent validation datasets or cross-validation.

4.2   Excess risk bound for BR^2 learning

In this subsection, we analyze the generalization error of the BR^2 learning in the block-missing data. Similar to the R^2 learning, we consider the following function class:

$$f^{(s)}(\mathbf{X}^{(s)}) = \sum_{m=1}^{M} \mathbb{I}_m^{(s)} \sum_{d=1}^{D} \theta_{m;d}(\mathbf{X}_m^{(s)}) \beta_{m;d}^{(s)}, s \in [S],$$

where $\theta_{m;d} \in \mathcal{F}_{m;d}$ of certain function class, and $\Theta \in \mathcal{F} = \{(\theta_{1;1}, \cdots, \theta_{m;d}, \cdots, \theta_{M;D}),$ $\theta_{m;d} \in \mathcal{F}_{m;d}\}$ and $\beta_m^{(s)} = (\beta_{1;m}^{(s)}, \cdots, \beta_{D;m}^{(s)}) \in \mathcal{H}_m^{(s)} = \{\beta \in \mathbb{R}^D, \|\beta\|_1 \leq \alpha_m^{(s)}\}$ for $s \in [S]$ and $m \in [M]$. Considering the SIP, the function class for $\beta_m^{(s)}$'s is $\mathcal{H}_m = \{(\beta_m^{(1)}, \cdots, \beta_m^{(S)}) \mid \beta_m^{(s)} \in \mathcal{H}_m^{(s)}, \breve{\gamma}_m(\tau_m) \geq \eta_m(\tau_m), \bar{g}_m(\tau_m) \leq \pi_m(\tau_m)\}$ where the definition of $\breve{\gamma}_m(\tau_m)$ and $\bar{g}_m(\tau_m)$ are similar to $\breve{\gamma}(\tau_m)$ and $\bar{g}(\tau_m)$ in Section 4.2, but for the mth modality only. For notational brevity, let each modality-associated dictionary Θ_m contain D representers uniformly. We define the empirical risk as follows:

$$\hat{\mathcal{R}}(\Theta, \beta) = \frac{1}{S} \sum_{s=1}^{S} \frac{1}{n} \sum_{i=1}^{n} \ell\left\{y_i^{(s)}, \sum_{m=1}^{M} \mathbb{I}_m^{(s)} \sum_{d=1}^{D} \theta_{m;d}(\mathbf{x}_{i;m}^{(s)}) \beta_{m;d}^{(s)}\right\},$$

with empirical risk minimizer defined as $\hat{\Theta}$ and $\hat{\beta}$. Analogously, population risk is defined with expectation over $(\mathbf{X}^{(s)}, \mathbf{y}^{(s)}) \sim \mu^{(s)}$:

$$\mathcal{R}(\Theta, \beta) = \frac{1}{S} \sum_{s=1}^{S} \mathbb{E}_{(\mathbf{X}^{(s)}, \mathbf{y}^{(s)}) \sim \mu^{(s)}}\left[\ell\left\{\mathbf{y}^{(s)}, \sum_{m=1}^{M} \mathbb{I}_m^{(s)} \sum_{d=1}^{D} \theta_{m;d}(\mathbf{X}_m^{(s)}) \beta_{m;d}^{(s)}\right\}\right].$$

The missing indicators, $\mathbb{I}_m^{(s)}$, are embedded in both $\hat{\mathcal{R}}(\Theta, \beta)$ and $\mathcal{R}(\Theta, \beta)$. Note that learning $f^{(s)}$ using only observed modalities introduces non-negligible approximation errors, so the minimizers of the population risk, $\mathcal{R}(\Theta, \beta)$, are not necessarily the true $\theta_{m;d}$'s and $\beta_{m,d}^{(s)}$'s in the data generating process. Since our primary interest lies in whether integrating multi-source data outperforms individual learning despite missing blocks, we focus on analyzing the generalization error with inclusion of $\mathbb{I}_m^{(s)}$. In the following exposition, we denote the population risk minimizer as Θ^* and β^*. And we

are interested in the excess risk:

$$\mathcal{E}(\hat{\Theta}, \hat{\beta}) = \mathcal{R}(\hat{\Theta}, \hat{\beta}) - \mathcal{R}(\Theta^*, \beta^*).$$

Following the same steps as in Lemma 4.1, we obtain an upper bound on the complexity of learning the representation dictionary.

Lemma 4.2. *For any $\beta_m \in \mathcal{H}_m$ and fixed realizations $(\mathbf{x}_i^{(s)}, y_i^{(s)})_{i=1}^n, s \in [S]$, we define*

$$\mathbf{G}_\beta(\boldsymbol{\sigma}) = \sup_{\Theta \in \mathcal{F}} \frac{1}{nS} \sum_{s=1}^S \sum_{i=1}^n \sigma_{s,i} \left[\sum_{m=1}^M \mathbb{I}_m^{(s)} \sum_{d=1}^D \theta_{m;d}(\mathbf{x}_{i;m}^{(s)}) \beta_{m;d}^{(s)} \right],$$

where $\sigma_{s,i}$ are i.i.d Rademacher random variables, then we have:

$$\mathbb{E}_{\boldsymbol{\sigma}} \mathbf{G}_\beta(\boldsymbol{\sigma}) \leq \sum_{m=1}^M \frac{\alpha_m^2}{\tau_m} \sqrt{\frac{C_{\Theta_m}}{n \breve{\gamma}_m(\tau_m)}} + \sum_{m=1}^M \alpha_m \sqrt{\frac{C_{\Theta_m} \bar{g}_m(\tau_m)}{n \breve{\gamma}_m^2(\tau_m)}}, \tag{4.4}$$

$$\leq \sum_{m=1}^M \frac{\alpha_m^2}{\tau_m} \sqrt{\frac{C_{\Theta_m}}{n \eta_m(\tau_m)}} + \sum_{m=1}^M \alpha_m \sqrt{\frac{C_{\Theta_m} \pi_m(\tau_m)}{n \eta_m^2(\tau_m)}}, \text{ for any } \tau > 0, \tag{4.5}$$

where $\alpha_m = \max_s \alpha_m^{(s)}$. The constant C_{Θ_m} is the uniform upper bound $C_{\Theta_m} \geq C_{\theta_{m;d}}$ for all $d \in [D]$, where $\mathbb{E}\left\{ \sup_{\theta_{m;d} \in \mathcal{F}_{m;d}} \frac{1}{n} \sum_{i=1}^n \sigma_i \theta_{m;d}(\mathbf{x}_i) \right\} \lesssim \sqrt{\frac{C_{\theta_{m;d}}}{n}}.$

Lemma 4.2 follows all properties derived in Lemma 4.1. Since SIP is applied to each modality, varying values of γ_m and $\lambda_{m;2}$ could be selected for different modalities in (3.8). Therefore, the upper bound in (4.4) aggregates the complexities of each modality dictionary with modality-specific values of $\eta_m(\tau_m)$ and $\pi_m(\tau_m)$. In the same manner, we can derive the excess risk bound for the BR^2 learning as shown in the following corollary.

Corollary 4.1. *Suppose Assumption 4.1 holds, and the dictionary size is fixed as D for all modalities. Let $(\mathbf{x}_i^{(s)}, y_i^{(s)})$ be identically and independently distributed random samples following distribution $\mu^{(s)}$ defined over $\mathcal{X} \times \mathcal{Y}$, then for empirical risk minimizer*

$\hat{\Theta}$ and $\hat{\beta}$, with probability $1 - \delta$, we have

$$\mathcal{E}(\hat{\Theta}, \hat{\beta}) \leq 4L \sum_{m=1}^{M} \frac{\alpha_m^2}{\tau_m} \sqrt{\frac{C_{\Theta_m}}{n \eta_m(\tau_m)}} + 4L \sum_{m=1}^{M} \alpha_m \sqrt{\frac{C_{\Theta_m} \pi_m(\tau_m)}{n \eta_m^2(\tau_m)}}$$
$$+ 4\alpha BLM \sqrt{\frac{2 \log(2D)}{n}} + B \sqrt{\frac{2 \log(2/\delta)}{n}}.$$

5 Simulation Studies

We evaluate the performance of the proposed R^2 and BR^2 learning approaches under various simulation settings and compare them with several competitive methods. Section 5.1 investigates a multi-task learning (MTL) scenario with fully aligned covariates across data sources, comparing R^2 learning against standard MTL methods. Section 5.2 examines a multi-modality block-missing data setting, where BR^2 learning is compared with methods designed for block-missing data. In both cases, our methods demonstrate superior performance, highlighting their effectiveness in integrating heterogeneous data.

5.1 Simulations for Multi-task Data

We compare our method with several baseline and state-of-the-art MTL approaches. The baseline models include Individual Learning (IL), which fits a separate model for each data source using either a linear model (IL-linear) or a feed-forward neural network (IL-nn). Another baseline is Data Pooling (Pooling), which fits a single model across all sources, thereby ignoring any types of heterogeneity. These baselines represent two extremes: Pooling is optimal when all tasks are sampled from the same distribution, whereas IL is preferable when no sharing information is available.

We also compare our method against prominent multi-task methods from the literature. First, we consider Multi-task Representation Learning (MTRL; Du et al. 8, Tripuraneni et al. 51), which employs a uniform representer along with source-specific learners to accommodate both linear and nonlinear patterns. Second, we evaluate a penalized ERM algorithm (pERM; Tian et al. 48), which allows for similar, though not identical, linear representations across sources and accommodates a subset of outlier sources. Third, Adaptive and Robust Multi-task Learning (ARMUL; Duan and

Wang [9]) is included, as it encourages regression coefficients to conform to a low-rank representation, except for a small set of outliers. pERM is implemented using the open-source code provided by [48] (`https://github.com/ytstat/RL-MTL-TL`), and ARMUL is implemented based on the code from [9] (`https://github.com/kw2934/ARMUL?tab=readme-ov-file`). IL (both linear and neural network versions) and MTRL are implemented in `pytorch`. We adopt default algorithm setups, such as learning rate and epochs, in all the above implementation, and tune the penalty coefficients and neural nets structures in all methods with `optuna`.

We generate simulation data from $S = 10$ data sources as follows:

$$y^{(s)} = \Theta(\mathbf{X}^{(s)})\beta^{(s)} + \epsilon^{(s)}, \tag{5.1}$$

where covariates $\mathbf{X}^{(s)}$ follow a multivariate standard normal distribution with dimension $p = 30$. The representer dictionary $\Theta(\cdot) = (\theta_1, \cdots, \theta_{30})$ contains 30 representers, including 6 linear representers and 24 nonlinear representers, and detailed specification of these representers can be found in Appendix B.1. For each source, only three representers are informative, i.e., $\|\beta^{(s)}\|_0 = 3$ for all $s = 1, \cdots, S$. The noise term $\epsilon^{(s)}$ are i.i.d samples of standard normal distribution.

We consider various settings to reflect different levels of distribution and posterior heterogeneity. Specifically, we vary the number of shared representers among data sources by setting $I = 0, 1, 2, 3$. For instance, in the $I = 0$ setting, $\text{supp}(\beta^{(1)}) = \{1, 2, 3\}, \cdots, \text{supp}(\beta^{(10)}) = \{28, 29, 30\}$ implies that each data source has its own unique set of representers, representing the most heterogeneous scenario. In contrast, when $I = 3, \text{supp}(\beta^{(s)}) = \{1, 10, 25\}$ for all sources, indicating that all data sources share the same set of representers, corresponds to the homogeneous scenario. To introduce posterior heterogeneity, we specify the coefficients as $\beta^{(s)} = \beta + \delta^{(s)}$, where $\beta_{\text{supp}} = (1, 1, 1)$ and the entries in δ_{supp} follow $\mathcal{N}(0, \sigma^2)$, with $\sigma \in \{0.1, 1, 3\}$. For each simulation setting, we repeat 100 independent Monte Carlo experiments by default.

In the simulation experiment, we set sample size $n_s = 50$ for each source in the training, validation, and testing datasets. Model performance is evaluated using the average mean squared error (MSE) across all data sources. The model performance under varying settings $I \in \{0, 1, 2, 3\}$ and $\sigma \in \{0.1, 1, 3\}$ are shown in Figure 6. The proposed R^2
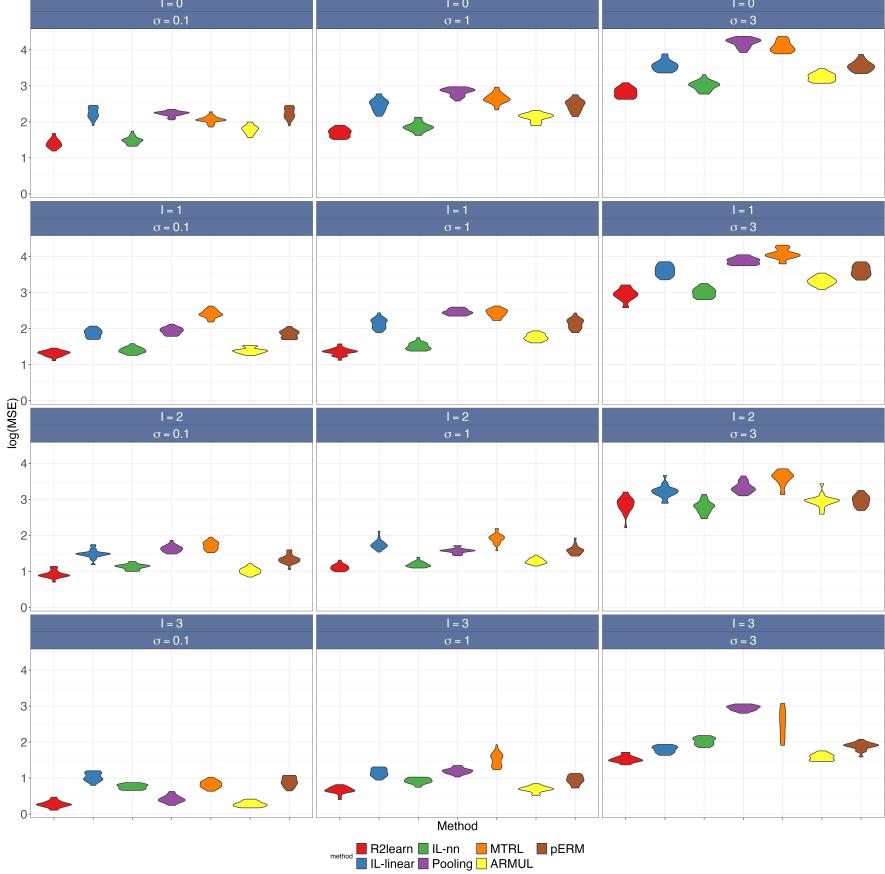
Figure 6: Visualization of log-scaled mean squared error (log(MSE)) in simulation under varying settings: $I \in \{0, 1, 2, 3\}$ and $\sigma \in \{0.1, 1, 3\}$ for the proposed method and competing methods.

learning method outperforms all other competing methods in all scenarios, indicating that it can incorporate both distribution and posterior heterogeneity properly. When both distribution shift and posterior drift are severe ($I = 0, \sigma = 3$), individual learning with neural nets is the best algorithm among competing methods, while our method can still achieve considerable improvement in prediction accuracy. Although there is no sharing of representers across data sources in the data-generating process, the SIP still promotes the integrativeness of representers. This allows us to obtain a representer dictionary which approximates the true representer dictionary, with some representers being more integrative as they may be shared across certain data sources. When both distribution shift and posterior drift are scarce ($I = 3, \sigma = 0.1$), Pooling and ARMUL stand out among competing methods, while our method can still achieve comparable prediction performance.

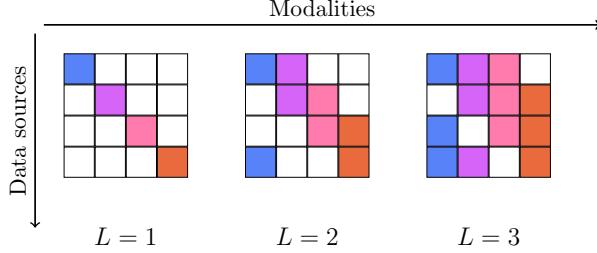Among all competing methods, individual learning (IL) performs well in the presence

Figure 7: Block missing patterns considered in simulation study. L indicates the number of observed modalities for each data source.

of distribution heterogeneity, and IL-nn generally outperforms the IL-linear due to the nonlinearity in the true data-generating process. The baseline Pooling performs well only under a setting of distribution and posterior homogeneity (e.g., $I = 3, \sigma = 0.1$). While MTRL can accommodate nonlinear patterns, its performance depends heavily on the number of shared representers across data sources. When representers are not jointly shared, the MTRL achieves prediction accuracy comparable to or even worse than Pooling. The ARMUL demonstrates robustness against posterior heterogeneity but struggles to properly handle distribution heterogeneity. Similarly, pERM, which seeks a low-rank representation, experiences performance deterioration when only a few representers are shared across data sources.

5.2 Simulations for Multi-modality Block-mising Data

This subsection examines the performance of our proposed BR^2 learning method in a multi-modality block-missing data setting. To our knowledge, no existing methods simultaneously address flexible block-missing data, complex covariates shifts, and posterior drift. We compare BR^2 learning with a benchmark Individual Learning (IL) method that uses only the observed modalities, and with DISCOM [65], which directly predicts without imputing missing modalities by leveraging all available data for estimation of covariance and cross-covariance matrices. Note that the Pooling method is inapplicable here due to missing modalities. Hyperparameters in [65] are selected via grid search in their program, and we still employ `optuna` package to tune the hyperparameters in our proposed BR^2 learning.

Figure 7 illustrates the block-missing patterns considered in the simulation where L denotes the number of modalities observed per source. In this experiment, we consider
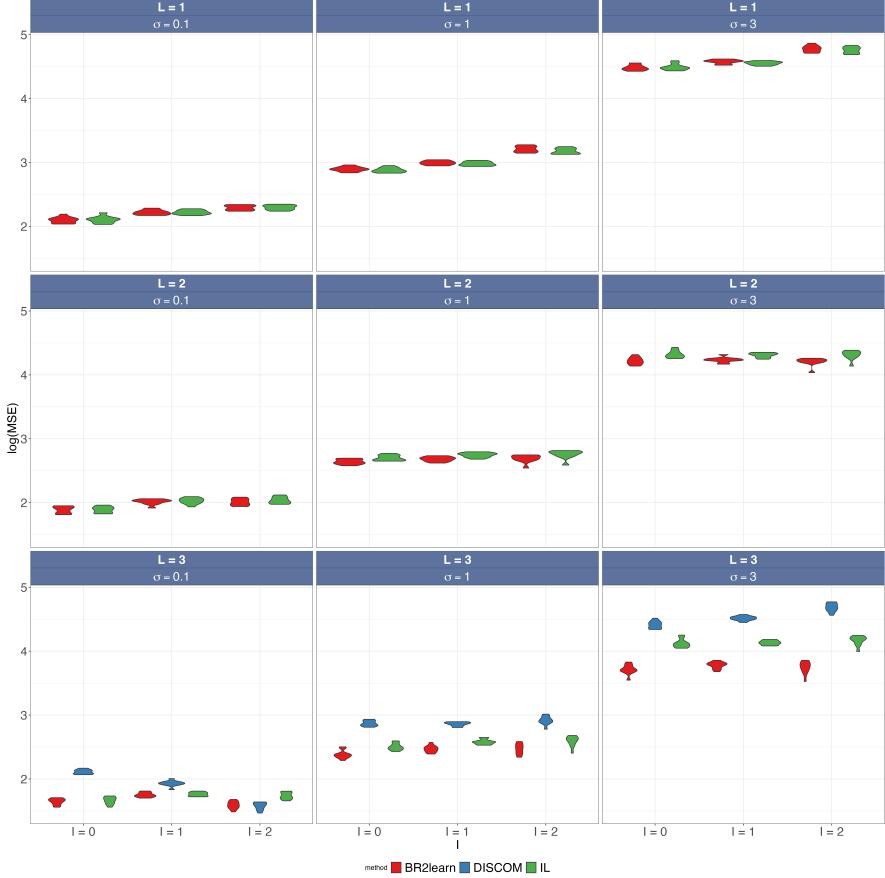
Figure 8: Visualization of log-scaled mean squared error (log(MSE)) in simulation under varying settings: $I \in \{0, 1, 2\}$, $\sigma \in \{0.1, 1, 3\}$ and $L \in \{1, 2, 3\}$ for the proposed method and competing methods.

$S = 4$ data sources, $M = 4$ modalities, and set the sample size $n_s = 200$ for training, validation, and testing. Data are generated according to:

$$y^{(s)} = \sum_{m=1}^{M} \Theta_m(\mathbf{X}_m^{(s)}) \beta_m^{(s)} + \epsilon^{(s)},$$

where modality-specific covariates $\mathbf{X}_m^{(s)}$ are sampled from a $q = 40$-dimensional multivariate normal distribution with modality-specific covariance structures (details are provided in Appendix B.2). For fair comparison, we restrict the representers to the linear case (orthogonal bases in $\mathbb{R}^q$) as DISCOM is designed for linear models. For each modality, only two representers are active, i.e., $\|\beta_m^{(s)}\|_0 = 2$, and the noise term $\epsilon^{(s)}$ are i.i.d samples of standard normal distribution..

Similar to Section 5.1, we vary the number of shared representers $I = 0, 1, 2$ and introduce posterior drift by setting $\beta_m^{(s)} = \beta_m + \delta^{(s)}$ with $\beta_{m,\text{supp}} = (1, 1)$ and

$\delta^{(s)} \sim \mathcal{N}(0, \sigma^2)$ for $\sigma \in \{0.1, 1, 3\}$. For each setting, we perform 100 Monte Carlo replications. Instead of evaluating the predictive performance on the underlying non-missing data as in [65], the validation and testing sets are generated with the same missing pattern as specified in Figure 7, and predictions are made based on observed modalities. The proposed method aims to improve the predictive performance for all data sources, regardless of missing patterns, while [65] focuses on estimating the underlying homogeneous coefficients accurately.

Figure 8 shows the log-scaled MSE under various settings of I, σ, and L. For $L = 1$, BR^2 learning achieves performance comparable to IL across all σ and I. As L and I increase, the advantages of BR^2 learning become more pronounced, demonstrating the benefits of integrating multiple data sources even in the presence of missing blocks. In contrast, DISCOM performs well only under homogeneous covariates and posterior distribution, with its performance deteriorating significantly when either distribution or posterior heterogeneity is high. In summary, BR^2 learning is robust to various block-missing patterns and maintains predictive performance better than or comparable to IL. When shared information exists across data sources, BR^2 learning effectively leverages this structure to improve predictions, even under significant distribution and posterior heterogeneity.

6 Real Data Analysis

In this section, we apply our proposed method, R^2 learning and BR^2 learning to several real-world datasets to evaluate its performance. Existing methods compared in Section 5 have also been assessed in these examples. In the pan-cancer example, all the features are consistently observed in all data sources, where R^2 learning applies; and we examine the performance of BR^2 learning in the ADNI study where some types of imaging features and genetic features are unobserved for certain groups of people.

6.1 Pan-Cancer Analysis

In this section, we analyze the pan-cancer data from [36], including five types of cancer: breast cancer (BRCA), melanoma (CM), colorectal cancer (CRC), non-small cell

lung cancer (NSCLC), and pancreatic cancer (PDAC). The data were collected from patient-derived xenografts (PDX) experiments, where each PDX line is created through transplanting a tumor into multiple mice, and applying multiple treatments to the mice. Covariates including DNA copy number, gene expression, and mutations were collected for PDX lines. To evaluate the treatment effects, a scaled measure of maximum tumor shrinkage from baseline is collected, termed as best average response (BAR). In our analysis, we are interested in predicting the tumor size shrinkage across five cancer types based on their covariates. Therefore, treatments are ignored and we focus on the untreated mice from each PDX line.

In this dataset, sample sizes for each tumor type are limited: 38 PDX lines for BRCA, 32 PDX lines for CM, 43 PDX lines for CRC, 25 PDX lines for NSCLC, and 35 PDX lines for PDAC. Therefore, the prediction performance of individual learning is inaccurate and highly variable. Furthermore, pan-cancer analysis has shown that many cancer types share some common mechanisms [35], which motivates us to integrate all these five types of tumor to achieve more accurate prediction for all tumor types. Due to the high-dimensionality of covariates, we first apply principal component analysis on each set of covariates: reducing the dimension of DNA copy number from 18,868 to 30, reducing the dimension of RNA count from 13,577 to 30, and reducing the dimension of mutations from 59 to 10. The obtained low-dimensional representations are treated as covariates and input into predictive algorithms.

To validate the predictive performance, we randomly select 60% PDX lines from each tumor type as training data, an additional 20% PDX lines from each tumor type as validation data for tuning parameters, and the remaining 20% PDX lines from each tumor type as testing data to evaluate the performance. We repeat the sample-splitting 100 times and run all the models on these independent replications. The performance of our R^2 learning and competing methods are evaluated by the mean squared error (MSE), reported in Table 1.

The proposed R^2 learning attains the minimal MSE for all five types of tumor types, which shows the advantage of our method in integrating heterogeneous datasets. ARMUL is another method improving the MSE for all five tumor types compared to Individual learning, and all other methods perform worse than Individual learning on

Tumor Type	R^2 learning	pERM	MTRL	Individual	ARMUL	Pooling
BRCA	**1.2342(0.3290)**	1.7643(0.5637)	1.6748(0.4737)	1.9837(0.4083)	1.3476(0.4873)	1.9983(0.3403)
CM	**0.8762(0.2980)**	1.3843(0.4384)	1.2893(0.4010)	1.4039(0.3987)	1.2299(0.4367)	1.8736(0.4021)
CRC	**0.8763(0.3023)**	1.8372(0.4030)	1.7903(0.3543)	1.4658(0.4203)	1.3948(0.5432)	1.8983(0.3988)
NSCLC	**1.3422(0.3433)**	1.8932(0.4012)	1.4483(0.4203)	2.4394(0.4034)	1.5328(0.3509)	1.7534(0.3874)
PDAC	**0.8232(0.2937)**	1.3324(0.3873)	1.4530(0.4234)	1.9388(0.4830)	1.3847(0.4375)	2.0933(0.4039)

Table 1: MSEs of proposed R^2 learning and competing methods on five types of tumor datasets.

at least one tumor type.

6.2 Alzheimer's Disease Neuroimaging Initiative (ADNI) Study

In this section, we analyze the Alzheimer's Disease Neuroimaging Initiative study [32] to showcase the performance of BR^2 learning in multi-source multi-modality block-missing scenarios. Our analysis aims to learn an accurate predictor for cognitive status, measured by the Mini-Mental State Examination (MMSE, [13]). The covariates are from three modalities: magnetic resonance imaging (MRI), positron emission tomography (PET) and gene expression. The block missing structure emerges in the second phase of the ADNI study at the 48th month, where the detailed block missing structure is illustrated in Figure 1.

In our analysis, we follow the pre-processing steps in [62] which extract the region of interest (ROI) level data in ADNI and obtain $q_1 = 267$ features for MRI and $q_2 = 113$ feature for PET. For the genetic features, sure independence screening (SIS) [11] is adopted and $q_3 = 300$ genetic features are retained. In total, we use $p = q_1 + q_2 + q_3 = 680$ features to build our predictor, and four data sources with varying observation patterns are considered, where the first source with n_1 observe all three modalities, while the rest three sources have one different missing modality, respectively.

In this analysis, we compare our method with the same competing methods as in Section 5.2, including individual learning (IL), DISCOM [65]. To evaluate the performance of our proposed BR^2 learning and competing methods, we split the data into training, validation and test set randomly with proportion 60%, 20% and 20%, respectively. Note that the sampling procedure follows stratified sampling, for example, the training set consists of 60% samples from each data source. We measure the performance with mean squared error (MSE) and repeat the above sampling procedure 100

times. The results are shown in Table 2.

Data Source	BR^2 learning	IL	DISCOM
1	**17.383(2.983)**	20.511(3.437)	221.389(200.228)
2	**7.452(2.093)**	8.286(2.383)	736.37(85.989)
3	**38.938(5.232)**	42.418(7.320)	212.918(180.66)
4	**13.209(2.674)**	15.663(2.991)	169.161(191.951)

Table 2: MSEs of proposed BR^2 learning and competing methods on four data sources.

The proposed BR^2 learning achieves the minimal MSE over four data sources, and clearly improves performance compared to individual learning and DISCOM consistently, since our BR^2 learning can effectively integrate available information for block-missing data. Given the exploratory analysis in Section 1.2, covariates shift and posterior drift are non-negligible across the four data sources in this dataset, which deteriorates the performance of DISCOM. In contrast, our method is robust against these sources of heterogeneity and effectively extract shared information to achieve more accurate prediction. In addition, each modality is observed in three data sources, so that our method can utilize more samples to learn the representers, leading to an improved prediction accuracy compared to IL.

7 Discussion

This paper investigates data integration in supervised learning under three types of heterogeneity: covariate shift, posterior drift, and block missing data. Accommodating these heterogeneities enables practitioners to integrate a larger number of data sources and improve predictive performance. The proposed *Block-wise Representation Retrieval* learning method is, to the best of our knowledge, the first approach in the literature that simultaneously addresses all these heterogeneities. Moreover, we introduce the notion of *integrativeness* for representers and propose a novel *Selective Integration Penalty* that promotes the selection of more integrative representers in the dictionary, thereby enhancing model generalization both empirically and theoretically. The effectiveness of our approach is demonstrated through extensive simulation studies and real data applications.

Several directions for future research emerge from our work. First, in the multi-modality block-missing problem, our method exploits all observed modalities to improve prediction accuracy, while alternative approaches, such as imputing missing blocks as in [62], remain prevalent. Although our empirical studies show that our method outperforms competitors in the presence of covariate shift and posterior drift, it is still unclear which approach is favorable under specific conditions. Future work could explore quantitative criteria to distinguish the circumstances favoring one approach over the other. Second, it is of great interest to develop statistical inference procedures for models with such complex heterogeneity. While existing methods [42] provide inferential tools for linear models, extending these techniques to more general parameters encountered in scientific problems warrants further investigation.

Acknowledgment

References

[1] Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. (2017). Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30.

[2] Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828.

[3] Cai, T., Cai, T. T., and Zhang, A. (2016). Structured matrix completion with applications to genomic data integration. *Journal of the American Statistical Association*, 111(514):621–633.

[4] Chang, J. H., Russo, M., and Paul, S. (2024). Heterogeneous transfer learning for high dimensional regression with feature mismatch. *arXiv preprint arXiv:2412.18081*.

[5] Colnet, B., Mayer, I., Chen, G., Dieng, A., Li, R., Varoquaux, G., Vert, J.-P., Josse, J., and Yang, S. (2024). Causal inference methods for combining randomized trials and observational studies: a review. *Statistical science*, 39(1):165–191.

[6] Craig, E., Pilanci, M., Menestrel, T. L., Narasimhan, B., Rivas, M., Dehghannasiri, R., Salzman, J., Taylor, J., and Tibshirani, R. (2024). Pretraining and the lasso. *arXiv preprint arXiv:2401.12911*.

[7] Du, J.-H., Cai, Z., and Roeder, K. (2022). Robust probabilistic modeling for single-cell multimodal mosaic integration and imputation via scvaeit. *Proceedings of the National Academy of Sciences*, 119(49):e2214414119.

[8] Du, S. S., Hu, W., Kakade, S. M., Lee, J. D., and Lei, Q. (2020). Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*.

[9] Duan, Y. and Wang, K. (2023). Adaptive and robust multi-task learning. *The Annals of Statistics*, 51(5):2015–2039.

[10] Fan, J. and Gu, Y. (2024). Factor augmented sparse throughput deep relu neural networks for high dimensional regression. *Journal of the American Statistical Association*, 119(548):2680–2694.

[11] 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.

[12] Feng, J. and Simon, N. (2017). Sparse-input neural networks for high-dimensional nonparametric regression and classification. *arXiv preprint arXiv:1711.07592*.

[13] Folstein, M. F., Folstein, S. E., and McHugh, P. R. (1975). "mini-mental state": a practical method for grading the cognitive state of patients for the clinician. *Journal of psychiatric research*, 12(3):189–198.

[14] Gaynanova, I. and Li, G. (2019). Structural learning and integrative decomposition of multi-view data. *Biometrics*, 75(4):1121–1132.

[15] Golowich, N., Rakhlin, A., and Shamir, O. (2018). Size-independent sample complexity of neural networks. In *Conference On Learning Theory*, pages 297–299. PMLR.

[16] Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT press Cambridge.

[17] Gu, T., Han, Y., and Duan, R. (2024). Robust angle-based transfer learning in high dimensions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, page qkae111.

[18] Jalali, A., Sanghavi, S., Ruan, C., and Ravikumar, P. (2010). A dirty model for multi-task learning. *Advances in neural information processing systems*, 23.

[19] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[20] Koltchinskii, V. and Yuan, M. (2010). Sparsity in multiple kernel learning. *The Annals of Statistics*, 38(6):3660–3695.

[21] Kriebel, A. R. and Welch, J. D. (2022). Uinmf performs mosaic integration of single-cell multi-omic datasets using nonnegative matrix factorization. *Nature communications*, 13(1):780.

[22] Ledoux, M. and Talagrand, M. (2013). *Probability in Banach Spaces: isoperimetry and processes*. Springer Science & Business Media.

[23] Lemhadri, I., Ruan, F., Abraham, L., and Tibshirani, R. (2021). Lassonet: A neural network with feature sparsity. *Journal of Machine Learning Research*, 22(127):1–29.

[24] Li, L., Chang, D., Han, L., Zhang, X., Zaia, J., and Wan, X.-F. (2020). Multi-task learning sparse group lasso: a method for quantifying antigenicity of influenza a (h1n1) virus using mutations and variations in glycosylation of hemagglutinin. *BMC bioinformatics*, 21:1–22.

[25] Li, S., Cai, T. T., and Li, H. (2022). Transfer learning for high-dimensional linear regression: Prediction, estimation and minimax optimality. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(1):149–173.

[26] Lin, K. Z. and Zhang, N. R. (2023). Quantifying common and distinct information in single-cell multimodal data with tilted canonical correlation analysis. *Proceedings of the National Academy of Sciences*, 120(32):e2303647120.

[27] Lock, E. F., Hoadley, K. A., Marron, J. S., and Nobel, A. B. (2013). Joint and individual variation explained (jive) for integrated analysis of multiple data types. *The annals of applied statistics*, 7(1):523.

[28] Lock, E. F., Park, J. Y., and Hoadley, K. A. (2022). Bidimensional linked matrix factorization for pan-omics pan-cancer analysis. *The annals of applied statistics*, 16(1):193.

[29] Lounici, K., Pontil, M., Tsybakov, A. B., and Van De Geer, S. (2009). Taking advantage of sparsity in multi-task learning. *arXiv preprint arXiv:0903.1468*.

[30] Ma, J., Zhao, Z., Yi, X., Chen, J., Hong, L., and Chi, E. H. (2018). Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1930–1939.

[31] Maurer, A., Pontil, M., and Romera-Paredes, B. (2013). Sparse coding for multitask and transfer learning. In *International conference on machine learning*, pages 343–351. PMLR.

[32] Mueller, S. G., Weiner, M. W., Thal, L. J., Petersen, R. C., Jack, C., Jagust, W., Trojanowski, J. Q., Toga, A. W., and Beckett, L. (2005). The alzheimer's disease neuroimaging initiative. *Neuroimaging Clinics of North America*, 15(4):869.

[33] Neyshabur, B., Bhojanapalli, S., and Srebro, N. (2017). A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*.

[34] Pan, W., Shen, X., and Liu, B. (2013). Cluster analysis: unsupervised learning via supervised learning with a non-convex penalty. *The Journal of Machine Learning Research*, 14(1):1865–1889.

[35] Petralia, F., Ma, W., Yaron, T. M., Caruso, F. P., Tignor, N., Wang, J. M., Charytonowicz, D., Johnson, J. L., Huntsman, E. M., Marino, G. B., et al. (2024). Pan-cancer proteogenomics characterization of tumor immunity. *Cell*, 187(5):1255–1277.

[36] Rashid, N. U., Luckett, D. J., Chen, J., Lawson, M. T., Wang, L., Zhang, Y., Laber, E. B., Liu, Y., Yeh, J. J., Zeng, D., et al. (2020). High-dimensional precision medicine from patient-derived xenografts. *Journal of the American Statistical Association*, 116(535):1140–1154.

[37] Ravikumar, P., Lafferty, J., Liu, H., and Wasserman, L. (2009). Sparse additive models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 71(5):1009–1030.

[38] Rim, J., Xu, Q., Tang, X., Guo, Y., and Qu, A. (2025). Individualized time-varying nonparametric model with an application in mobile health. *Statistics in Medicine*, 44(5):e70005.

[39] Shi, X., Pan, Z., and Miao, W. (2023). Data integration in causal inference. *Wiley Interdisciplinary Reviews: Computational Statistics*, 15(1):e1581.

[40] Shimodaira, H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244.

[41] Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2013). A sparse-group lasso. *Journal of computational and graphical statistics*, 22(2):231–245.

[42] Song, S., Lin, Y., and Zhou, Y. (2024). Semi-supervised inference for block-wise missing data without imputation. *Journal of Machine Learning Research*, 25(99):1–36.

[43] Sui, Y., Xu, Q., Bai, Y., and Qu, A. (2025). Multi-task learning for heterogeneous multi-source block-wise missing data. *openreview.net*.

[44] Sun, X., Zhao, B., and Xue, F. (2025). Generalized heterogeneous functional model with applications to large-scale mobile health data. *arXiv preprint arXiv:2501.01135*.

[45] Tang, L. and Song, P. X. (2016). Fused lasso approach in regression coefficients clustering–learning parameter heterogeneity in data integration. *Journal of Machine Learning Research*, 17(113):1–23.

[46] Tang, T. M. and Allen, G. I. (2021). Integrated principal components analysis. *Journal of Machine Learning Research*, 22(198):1–71.

[47] Tian, Y. and Feng, Y. (2023). Transfer learning under high-dimensional generalized linear models. *Journal of the American Statistical Association*, 118(544):2684–2697.

[48] Tian, Y., Gu, Y., and Feng, Y. (2023). Learning from similar linear representations: Adaptivity, minimaxity, and robustness. *arXiv preprint arXiv:2303.17765*.

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

[50] Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(1):91–108.

[51] Tripuraneni, N., Jin, C., and Jordan, M. (2021). Provable meta-learning of linear representations. In *International Conference on Machine Learning*, pages 10434–10443. PMLR.

[52] Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press.

[53] Wang, P., Wang, H., Li, Q., Shen, D., and Liu, Y. (2024). Joint and individual component regression. *Journal of Computational and Graphical Statistics*, 33(3):763–773.

[54] Watkins, A., Ullah, E., Nguyen-Tang, T., and Arora, R. (2023). Optimistic rates for multi-task representation learning. *Advances in Neural Information Processing Systems*, 36:2207–2251.

[55] Wen, W., Wu, C., Wang, Y., Chen, Y., and Li, H. (2016). Learning structured sparsity in deep neural networks. *Advances in neural information processing systems*, 29.

[56] Widmer, G. and Kubat, M. (1996). Learning in the presence of concept drift and hidden contexts. *Machine learning*, 23:69–101.

[57] Witten, D. M. and Tibshirani, R. (2009). Covariance-regularized regression and classification for high dimensional problems. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 71(3):615–636.

[58] Witten, D. M., Tibshirani, R., and Hastie, T. (2009). A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3):515–534.

[59] Wu, C., Kwon, S., Shen, X., and Pan, W. (2016). A new algorithm and theory for penalized regression-based clustering. *Journal of Machine Learning Research*, 17(188):1–25.

[60] Xiao, L. and Xiao, L. (2024). Sparse and integrative principal component analysis for multiview data. *Electronic Journal of Statistics*, 18(2):3774–3824.

[61] Xue, F., Ma, R., and Li, H. (2025). Statistical inference for high-dimensional linear regression with blockwise missing data. *Statistica Sinica*, 35:431–456.

[62] Xue, F. and Qu, A. (2021). Integrating multisource block-wise missing data in model selection. *Journal of the American Statistical Association*, 116(536):1914–1927.

[63] Yang, S. and Ding, P. (2020). Combining multiple observational data sources to estimate causal effects. *Journal of the American Statistical Association*, 115(531):1540–1554.

[64] Yu, G. and Hou, S. (2022). Integrative nearest neighbor classifier for block-missing multi-modality data. *Statistical Methods in Medical Research*, 31(7):1242–1262.

[65] Yu, G., Li, Q., Shen, D., and Liu, Y. (2020). Optimal sparse linear prediction for block-missing multi-modality data without imputation. *Journal of the American Statistical Association*, 115(531):1406–1419.

[66] Yuan, L., Wang, Y., Thompson, P. M., Narayan, V. A., and Ye, J. (2012). Multi-source learning for joint analysis of incomplete multi-modality neuroimaging data. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1149–1157.

[67] Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(1):49–67.

[68] Zhang, J., Xue, F., Xu, Q., Lee, J., and Qu, A. (2024). Individualized dynamic latent factor model for multi-resolutional data with application to mobile health. *Biometrika*, 111(4):1257–1275.

[69] Zhou, D., Cai, T., and Lu, J. (2023). Multi-source learning via completion of block-wise overlapping noisy matrices. *Journal of Machine Learning Research*, 24(221):1–43.

[70] Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429.

A   Multivariate representer output

In this part, we provide a detailed treatment when representers output a multivaraite latent representation. Formally, we consider $\theta_d : \mathcal{X} \to \mathbb{R}^{r_d}$, and the output of the dictionary Θ is of dimension $\sum_{d=1}^{D} r_d$. The corresponding learner can be formulated as $\beta^{(s)} = (\beta_{(1)}^{(s)}, \cdots, \beta_{(D)}^{(S)})$ for $s = 1, \cdots, S$ where $\beta_{(d)}^{(s)} \in \mathbb{R}^{(r_d)}$. The ℓ_1 penalty on learners in Section 3.1 are adapted as ℓ_2 penalty, i.e., group lasso penalty on each $\beta_{(d)}^{(s)}$. That is, we are interested in selecting the representers output as a whole. Therefore, the first penalty imposed on β's are $\sum_{s=1}^{S} \lambda^{(s)} \sum_{d=1}^{D} \sqrt{r_d} \|\beta_{(d)}^{(s)}\|_2$. Similarly, we can define

$$\tilde{\gamma}_d(\tau) = \sum_{s=1}^{S} \min(1, \frac{\|\beta_{(d)}^{(s)}\|_2}{\tau})$$

and plug into the SIP to improve the integrativeness of representers. For the block-missing problem, we follow the same manner that introduce ℓ_2 penalty and ℓ_2-based SIP to accommodate multivariate representer output.

B   Detailed specification of simulation settings

B.1   Simulation settings for R^2 learning

In Section 5.1, covariates $\mathbf{X}^{(s)}$ are generated from $p = 30$ dimensional standard multivariate normal distribution. The representers are generated based on orthogonal bases in $\mathbb{R}^{30}$. Specifically, we first generate a random matrix A where all entries follow standard normal distribution, then QR decomposition is applied to obtain the orthogonal bases, say, $v_1, v_2, \cdots, v_{30}$. The representers are generated as follows:

- $\theta_1(x) = v_1^T x$
- $\theta_2(x) = v_2^T x$
- $\theta_3(x) = v_3^T x$
- $\theta_4(x) = v_4^T x$
- $\theta_5(x) = v_5^T x$
- $\theta_6(x) = v_6^T x$
- $\theta_7(x) = v_7^T (x \circ x)$
- $\theta_8(x) = v_8^T (x \circ x)$
- $\theta_9(x) = v_9^T (x \circ x)$
- $\theta_{10}(x) = v_{10}^T \sin(x)$

- $\theta_{11}(x) = v_{11}^T \log(1 + x \circ x)$
- $\theta_{12}(x) = v_{12}^T \exp(x)$
- $\theta_{13}(x) = \sum_{i=1}^{15} x_i$
- $\theta_{14}(x) = \sum_{i=16}^{30} x_i$
- $\theta_{15}(x) = v_{15}^T \cos(x)$
- $\theta_{16}(x) = v_{16}^T \tanh(x)$
- $\theta_{17}(x) = v_{17}^T |x|$
- $\theta_{18}(x) = v_{18}^T \sqrt{|x|}$
- $\theta_{19}(x) = v_{19}^T \sqrt[3]{x}$
- $\theta_{20}(x) = v_{20}^T \exp(-x)$

- $\theta_{21}(x) = v_{21}^T \log(1 + |x|)$
- $\theta_{22}(x) = v_{22}^T \sinh(x)$
- $\theta_{23}(x) = v_{23}^T \cosh(x)$
- $\theta_{24}(x) = v_{24}^T \arctan(x)$
- $\theta_{25}(x) = v_{25}^T \arcsin(\text{clip}(x, -1, 1))$
- $\theta_{26}(x) = v_{26}^T \arccos(\text{clip}(x, -1, 1))$
- $\theta_{27}(x) = v_{27}^T \text{clip}(x, -0.999, 0.999)$
- $\theta_{28}(x) = v_{28}^T \text{arcsinh}(x)$
- $\theta_{29}(x) = v_{29}^T \text{arccosh}(\text{clip}(x, 1, \infty))$
- $\theta_{30}(x) = v_{30}^T \max(x, 0.5)$

B.2   Simulation settings for BR^2 learning

In Section 5.2, covariates $\mathbf{X}_m$ are generated from $q = 40$ dimensional multivariate normal distribution with mean zero, and varying covariance matrix. The covariance matrices for 4 modaliteis are as follows: Σ_1 is $q-$dimensional identity matrix, Σ_2 is AR1 matrix with correlation 0.8, Σ_3 is AR1 matrix with correlation 0.5, and Σ_4 is exchangeable matrix with correlation 0.5.
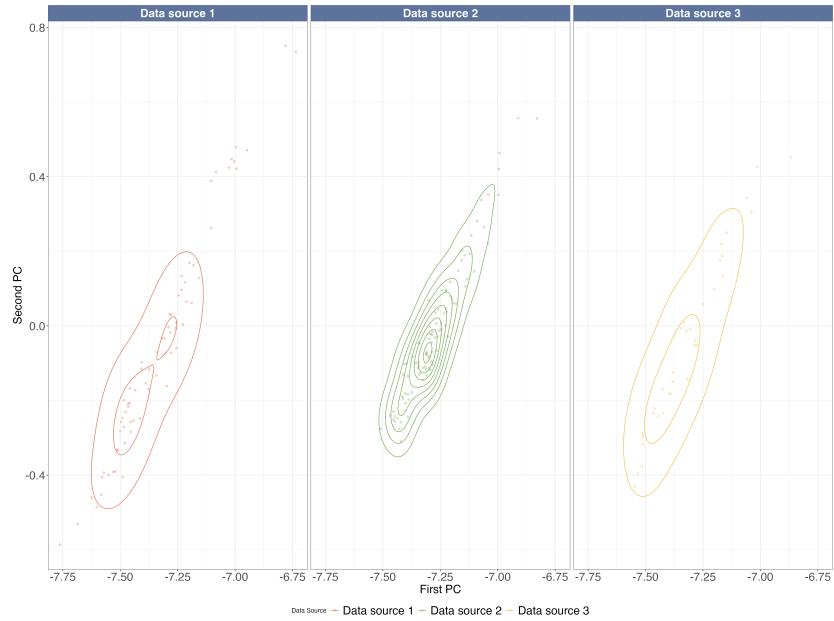
C   Motivating example figures

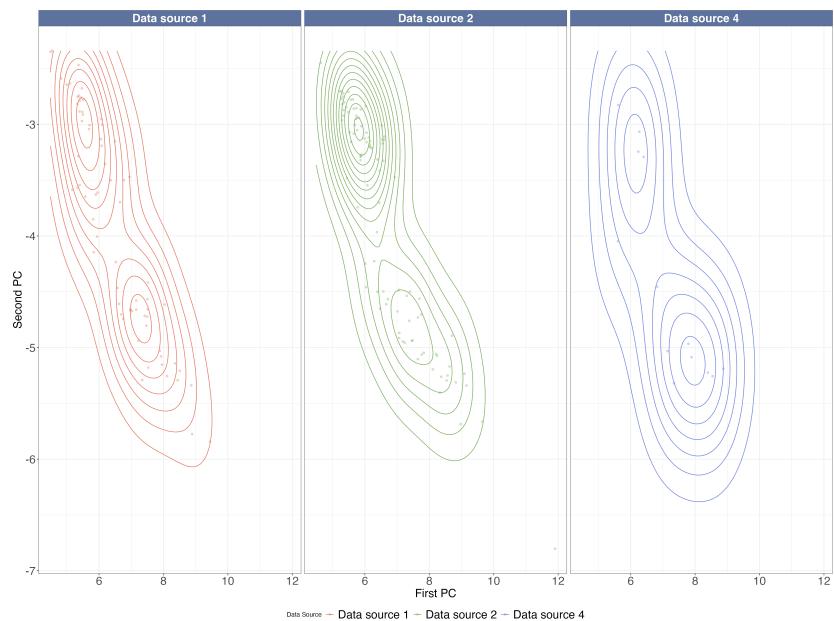Figure 9: Density plot of 2d projection of data source 1, 2, 3 of MRI modality.

Figure 10: Density plot of 2d projection of data source 1, 2, 4 of PET modality.
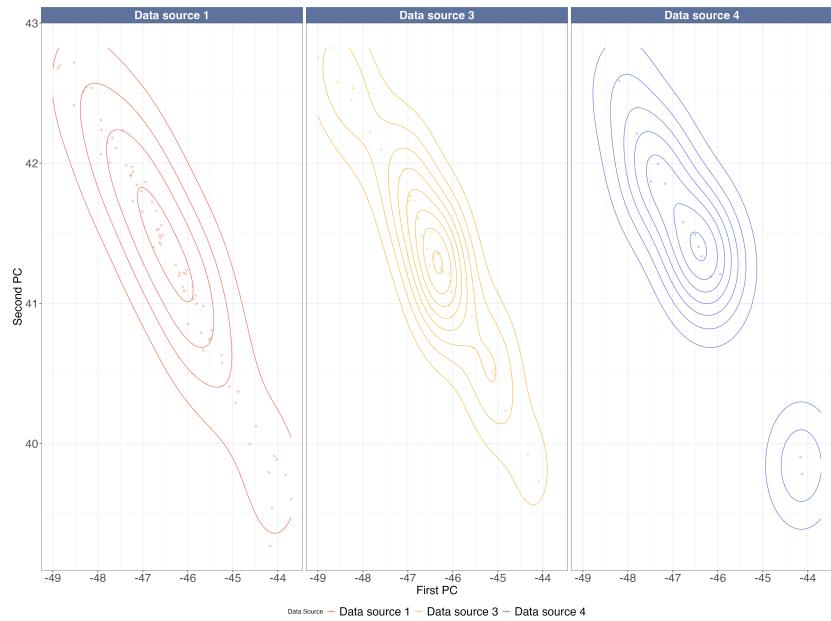
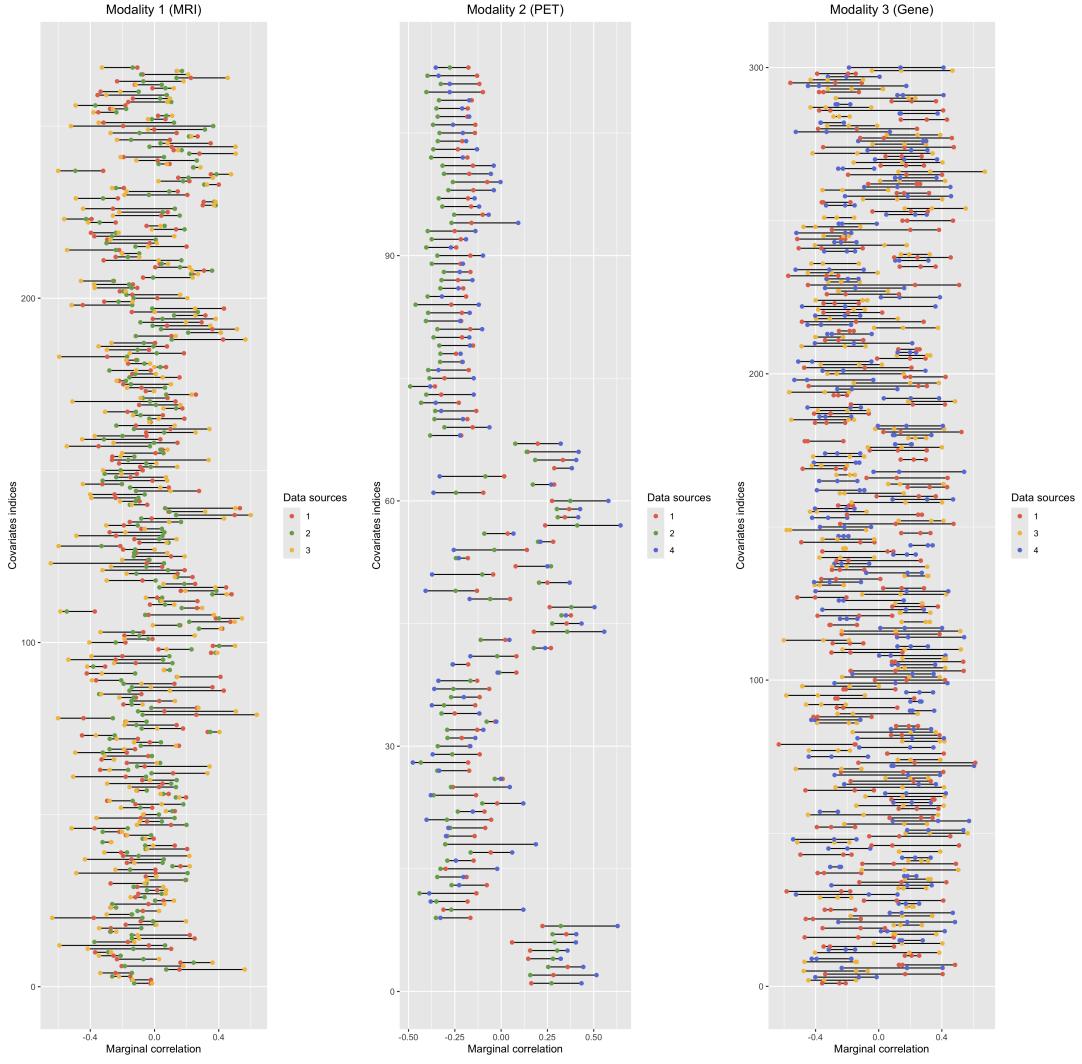Figure 11: Density plot of 2d projection of data source 1, 3, 4 of Gene modality.

Figure 12: Marginal correlation between each covariates and response across sources. The left plot shows the marginal correlation of covariates in the MRI modality. The central plot shows the marginal correlation of covariates in the PET modality, and the right plot shows the marginal correlation of covariates in the Gene modality.

D Technical proofs

In this section, we deliver the technical details of the theoretical established in Section 4. We start with the excess risk bound for R^2 learning, followed by the excess risk bound for BR^2 learning.

D.1 Proof of Lemma 4.1

Proof. In this section, we quantify the complexity of representers for any fixed β and data points $(\mathbf{x}_i^{(s)}, y_i^{(s)}), i \in [n], s \in [S]$:

$$G_\beta(\boldsymbol{\sigma}) = \sup_{\theta \in \mathcal{F}} \frac{1}{nS} \sum_{s=1}^S \sum_{i=1}^n \sigma_{s,i} \langle \Theta(\mathbf{x}_i^{(s)}), \beta^{(s)} \rangle$$

$$= \sup_{\theta \in \mathcal{F}} \frac{1}{nS} \sum_{s=1}^S \sum_{i=1}^n \sigma_{s,i} \sum_{d=1}^D \theta_d(\mathbf{x}_i^{(s)}) \beta_d^{(s)}$$

In this Lemma, we provide an upper bound on $\mathbb{E}[G_\beta(\boldsymbol{\sigma})]$. Before we move on, we define $I_d = \{s : \beta_d^{(s)} \neq 0\}$ as the set of tasks retrieving the dth representer, and define $\gamma_d = |I_d|$. Then we have

$$\mathbb{E}[G_{\boldsymbol{\sigma}}(\beta)] = \frac{1}{nS} \mathbb{E}\left[\sup_{\Theta \in \mathcal{F}} \sum_{s=1}^S \sum_{i=1}^n \sigma_{s,i} \sum_{d=1}^D \theta_d(\mathbf{x}_i^{(s)}) \beta_d^{(s)} \right]$$

$$= \frac{1}{nS} \sum_{d=1}^D \mathbb{E}\left[\sup_{\theta_d \in \mathcal{F}_d} \sum_{s \in I_d} \sum_{i=1}^n \sigma_{s,i} \theta_d(\mathbf{x}_i) \beta_d^{(s)} \right]$$

$$= \frac{1}{nS} \sum_{d=1}^D n\gamma_d \mathbb{E}\left[\sup_{\theta_d \in \mathcal{F}_d} \frac{1}{n\gamma_d} \sum_{s \in I_d} \sum_{i=1}^n \sigma_{s,i} \theta_d(\mathbf{x}_i^{(s)}) \beta_d^{(s)} \right]$$

$$\overset{(i)}{\leq} \frac{1}{nS} \sum_{d=1}^D n\gamma_d \alpha \mathbb{E}\left[\sup_{\theta_d \in \mathcal{F}_d} \frac{1}{n\gamma_d} \sum_{s \in I_d} \sum_{i=1}^n \sigma_{s,i} \theta_d(\mathbf{x}_i^{(s)}) \right]$$

$$= \frac{\alpha}{S} \sum_{d=1}^D \gamma_d \mathfrak{R}_{n\gamma_d}(\mathcal{F}_d)$$

where (i) holds since $|\beta_d^{(s)}| \leq \|\beta^{(s)}\|_1 \leq \alpha$. Additionally, due to the symmetric of $\mathcal{F}_d$ (if $\theta_d \in \mathcal{F}_d$, then $-\theta_d$ also belongs to $\mathcal{F}_d$), $|\beta_d^{(s)}|$ can be pulled out by taking the supremum over a symmetric function class because of contraction lemma [22]. $\mathfrak{R}_{n\gamma_d}(\mathcal{F}_d)$ measures the complexity of function class $\mathcal{F}_d$, and sample size $n\gamma_d$ is the number of total samples used to learn θ_d in the R^2 learning. For many commonly adopted function classes, $\mathfrak{R}_n(\mathcal{F}) \leq O(\sqrt{\frac{C_\mathcal{F}}{n}})$ with constant $C_\mathcal{F}$ associated with function class $\mathcal{F}$. For example, $\mathfrak{R}_n(\mathcal{F}) \leq O(\sqrt{\frac{B^2 X^2}{n}})$ for norm-constrained linear space $\mathcal{F} = \{f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle : \|\mathbf{w}\| \leq B\}$ for $\{\mathbf{x} : \|\mathbf{x}\| \leq X\}$. In addition, for a function class of two-layer feed-forward neural nets $\mathcal{F} = \{f_W(x) = W_2\sigma(W_1x) \mid \|W_1\|_F \leq B_1, \|W_2\|_F \leq B_2\}$, its

Rademacher complexity scales as $O(\frac{B_1 B_2}{\sqrt{n}})$. Therefore, in the following derivation, we let $\mathfrak{R}_{n\gamma_d}(\mathcal{F}) \leq \sqrt{\frac{C_{\theta_d}}{n}}$ where C_{θ_d} is a constant dependent on the choice of function class $\mathcal{F}_d$. Further, $C_\Theta \geq C_{\theta_d}$ for $d \in [D]$ is a uniform upper bound constant for all representers. Then we have

$$
\begin{aligned}
\mathbb{E}[G_\beta(\boldsymbol{\sigma})] &\leq \frac{\alpha}{S} \sum_{d=1}^{D} \gamma_d \mathfrak{R}_{n\gamma_d}(\mathcal{F}_d) \\
&\leq \frac{\alpha}{S} \sum_{d=1}^{D} \gamma_d \sqrt{\frac{C_{\theta_d}}{n\gamma_d}} \\
&= \frac{\alpha}{S} \sqrt{\frac{C_\Theta}{n}} \sum_{d=1}^{D} \sqrt{\gamma_d}
\end{aligned}
$$

However, in our R^2 learning, we maximize the continuous approximation of γ_d, $\tilde{\gamma}_d(\tau) = \sum_{s=1}^{S} \min(1, \frac{\beta_d^{(s)}}{\tau})$. In the following, we establish the upper bound with respect to $\tilde{\gamma}_d$'s. Let's define $g_d^{(s)}(\tau) = \mathbb{I}(0 < |\beta_d^{(s)}| < \tau)$ as the upper bound of the approximation error:

$$
\gamma_d - \tilde{\gamma}_d(\tau) \leq \sum_{s=1}^{S} g_d^{(s)}(\tau).
$$

Now, let's reconsider the quantity of interest as follows:

$$
\begin{aligned}
\mathbb{E}[G_{\boldsymbol{\sigma}}(\beta)] &= \frac{1}{nS} \mathbb{E}\left[\sup_{\Theta \in \mathcal{F}} \sum_{s,i} \sigma_{s,i} \sum_d \theta_d(\mathbf{x}_i^{(s)}) \beta_d^{(s)} \right] \\
&\leq \frac{1}{nS} \sum_{d=1}^{D} \mathbb{E}\left[\sup_{\theta_d \in \mathcal{F}_d} \sum_{s \in I_d} |\beta_d^{(s)}| \sum_i \sigma_{s,i} \theta_d(\mathbf{x}_i^{(s)}) \right] \\
&\leq \frac{1}{nS} \sum_{d=1}^{D} \mathbb{E}\left[\sup_{\theta_d \in \mathcal{F}_d} \sum_{s \in I_d} [\alpha(1 - g_d^{(s)}) + \tau g_d^{(s)}] \sum_i \sigma_{s,i} \theta_d(\mathbf{x}_i^{(s)}) \right] \\
&\leq \underbrace{\frac{1}{nS} \sum_{d=1}^{D} \alpha \mathbb{E}\left[\sup_{\theta_d \in \mathcal{F}_d} \sum_{s \in I_d} (1 - g_d^{(s)}(\tau)) \sum_i \sigma_{s,i} \theta_d(\mathbf{x}_i^{(s)}) \right]}_{A} \\
&\quad + \underbrace{\frac{1}{nS} \sum_{d=1}^{D} \tau \mathbb{E}\left[\sup_{\theta_d \in \mathcal{F}_d} \sum_{s \in I_d} g_d^{(s)}(\tau) \sum_i \sigma_{s,i} \theta_d(\mathbf{x}_i^{(s)}) \right]}_{B}.
\end{aligned}
$$

Now let's define the set $I_{d,1} = \{s : s \in I_d, g_d^{(s)}(\tau) = 0\}$ as the data sources whose dth coordinate $|\beta_d^{(s)}| \geq \tau$ and $I_{d,2} = \{s : s \in I_d, g_d^{(s)}(\tau) = 1\}$ as teh data sources whose dth coordinate $0 < |\beta_d^{(s)}| < \tau$. Then we have

$$
\begin{aligned}
A &\leq \frac{\alpha}{nS} \sum_{d=1}^{D} n|I_{d,1}| \mathbb{E}\left[\sup_{\theta_d \in \mathcal{F}_d} \frac{1}{n|I_{d,1}|} \sum_{s \in I_{d,1}} \sum_i \sigma_{s,i} \theta_d(\mathbf{x}_i^{(s)}) \right] \\
&\leq \frac{\alpha}{nS} \sum_{d=1}^{D} n|I_{d,1}| \sqrt{\frac{C_\Theta}{n|I_{d,1}|}} \leq \frac{\alpha}{S} \sqrt{\frac{C_\Theta}{n}} \sum_{d=1}^{D} \sqrt{|I_{d,1}|} \\
&\leq \frac{\alpha}{S} \sqrt{\frac{C_\Theta}{n}} \sum_{d=1}^{D} \sqrt{\tilde{\gamma}_d(\tau)},
\end{aligned}
$$

where the last inequality holds because $|I_{d,1}| = \gamma_d - \sum_{s=1}^{S} g_d^{(s)}(\tau) \leq \tilde{\gamma}_d(\tau)$. In the same manner, we have

$$
B \leq \frac{\tau}{S} \sqrt{\frac{C_\Theta}{n}} \sum_{d=1}^{D} \sqrt{|I_{d,2}|} = \frac{\tau}{S} \sqrt{\frac{C_\Theta}{n}} \sum_{d=1}^{D} \sqrt{g_d(\tau)},
$$

where $g_d(\tau) = \sum_{s=1}^{S} g_d^{(s)}(\tau)$. Now we analyze $\sum_{d=1}^{D} \sqrt{\tilde{\gamma}_d(\tau)}$ and $\sum_{d=1}^{D} \sqrt{g_d(\tau)}$. Based on Cauchy–Schwarz inequality, we have

$$
\begin{aligned}
\sum_{d=1}^{D} \sqrt{\tilde{\gamma}_d(\tau)} &\leq \sqrt{D \sum_{d=1}^{D} \tilde{\gamma}_d(\tau)} \leq \sqrt{D \sum_{d=1}^{D} \sum_{s=1}^{S} \min(1, \frac{|\beta_d^{(s)}|}{\tau})} \\
&\leq \sqrt{D \sum_{d=1}^{D} \sum_{s=1}^{S} \frac{|\beta_d^{(s)}|}{\tau}} \leq \sqrt{\frac{DS\alpha}{\tau}},
\end{aligned}
$$

which also implies $D \leq \frac{S\alpha}{\tau \check{\gamma}(\tau)}$, and plug into the above formula, we obtain

$$
A \leq \frac{\alpha}{S} \sqrt{\frac{C_\Theta}{n}} \sqrt{\frac{S^2 \alpha^2}{\tau^2 \check{\tau}(\tau)}} = \frac{\alpha^2}{\tau} \sqrt{\frac{C_\Theta}{n \check{\gamma}(\tau)}}.
$$

Similarly, we can derive the upper bound for B:

$$B \leq \frac{\tau}{S}\sqrt{\frac{C_\Theta}{n}}\sum_{d=1}^{D}\sqrt{g_d(\tau)} \leq \frac{\tau}{S}\sqrt{\frac{C_\Theta}{n}}\sqrt{D\sum_{d=1}^{D}g_d(\tau)}$$

$$\leq \frac{\tau}{S}\sqrt{\frac{C_\Theta}{n}}\sqrt{D^2\bar{g}(\tau)} \leq \frac{\tau}{S}\sqrt{\frac{C_\Theta}{n}}\sqrt{\frac{S^2\alpha^2\bar{g}(\tau)}{\tau^2\breve{\gamma}^2(\tau)}}$$

$$= \alpha\sqrt{\frac{C_\Theta\bar{g}(\tau)}{n\breve{\gamma}^2(\tau)}}.$$

Combining upper bounds for A and B and given the lower bound for $\breve{\gamma}(\tau)$ and upper bound for $\bar{g}(\tau)$ in $\mathcal{H}$ can conclude the results. □

D.2   Lemma D.1

Lemma D.1. *For any fixed β and data realizations, we have the following concentration inequality:*

$$\mathbb{P}\left(G_\beta(\boldsymbol{\sigma}) - \mathbb{E}[G_\beta(\boldsymbol{\sigma})] \geq \epsilon\right) \leq \exp(-\frac{nS\epsilon^2}{2B^2\alpha^2}).$$

Proof. Note that $G_\beta(\boldsymbol{\sigma})$ is viewed as a function of Rademacher random variables, we may change one value of $\sigma_{s,i}$ to $-\sigma_{s,i}$, denoted as $\sigma'_{s,i}$ and denote $(\sigma_{1,1}, \cdots, \sigma'_{s,i}, \cdots, \sigma_{S,n})$ as $\boldsymbol{\sigma}'$. Then we have the following inequality to bound the differences for any s, i:

$$|G_\beta(\boldsymbol{\sigma}) - G_\beta(\boldsymbol{\sigma}')| \leq \frac{2}{nS}\sup_{\theta\in\mathcal{F}}\sum_{d=1}^{D}|\theta_d(\mathbf{x}_i^{(s)})\beta_d^{(s)}|$$

$$\leq \frac{2}{nS}B\sum_{d=1}^{D}|\beta_d^{(s)}| = \frac{2}{nS}B\alpha$$

Followed by the McDiarmid's inequality [52], we obtain the expected concentration inequality. □

D.3   Proof of Theorem 4.1

Proof. First of all, the excess risk bound can be decomposed as follows:

$$\mathcal{E}(\hat{\Theta}, \hat{\beta}) = \underbrace{\mathcal{R}(\hat{\Theta}, \hat{\beta}) - \hat{\mathcal{R}}(\hat{\Theta}, \hat{\beta})}_{\mathcal{E}_1} + \underbrace{\hat{\mathcal{R}}(\hat{\Theta}, \hat{\beta}) - \hat{\mathcal{R}}(\Theta^*, \beta^*)}_{\mathcal{E}_2} + \underbrace{\hat{\mathcal{R}}(\Theta^*, \beta^*) - \mathcal{R}(\Theta^*, \beta^*)}_{\mathcal{E}_3},$$

where $\mathcal{E}_2 \leq 0$ by the definition of empirical minimizer. Since $|\mathcal{E}_1|, |\mathcal{E}_2| \leq \sup_{\Theta, \beta} |\hat{\mathcal{R}}(\Theta, \beta) - \mathcal{R}(\Theta, \beta)|$, we can obtain the excess risk bound by the following uniform convergence statement:

$$\mathbb{P}(\mathcal{E}(\hat{\Theta}, \hat{\beta}) \geq \epsilon) \leq \mathbb{P}(\sup_{\Theta, \beta} |\hat{\mathcal{R}}(\Theta, \beta) - \mathcal{R}(\Theta, \beta)| \geq \epsilon/2).$$

We define $G_{nS} = \sup_{\Theta, \beta} \mathcal{R}(\Theta, \beta) - \hat{\mathcal{R}}(\Theta, \beta)$, which is a random variable dependent on nS samples $(\mathbf{x}_i^{(s)}, y_i^{(s)})$ for $i \in [n]$ and $s \in [S]$, and $G'_{nS} = \sup_{\Theta, \beta} \hat{\mathcal{R}}(\Theta, \beta) - \mathcal{R}(\Theta, \beta)$, then we have

$$\mathbb{P}(\sup_{\Theta, \beta} |\hat{\mathcal{R}}(\Theta, \beta) - \mathcal{R}(\Theta, \beta)| \geq \epsilon/2) \leq \mathbb{P}(G_{nS} \geq \epsilon/2) + \mathbb{P}(G'_{nS} \geq \epsilon/2).$$

Since G_{nS} and G'_{nS} are sample mean of loss function on i.i.d samples from $\mu^{(s)}$ and B-boundedness of the loss function, we may apply McDiarmid's inequality [52] to obtain the following concentration inequality for G_{nS}:

$$\mathbb{P}(G_{nS} - \mathbb{E}[G_{nS}] \geq \epsilon) \leq \exp(-2n\epsilon^2/B^2),$$

and based on the definition of Rademacher complexity, $\mathbb{E}[G_{nS}] \leq 2\mathfrak{R}_{nS}[\mathcal{A}]$ where $\mathcal{A} = \{(\mathbf{x}^{(s)}, y^{(s)}) \mapsto \ell(y^{(s)}, \sum_{d=1}^{D} \theta_d(\mathbf{x}^{(s)})\beta_d^{(s)}), s \in [S], \theta_d \in \mathcal{F}_d, \beta^{(s)} \in \mathcal{H}^{(s)}\}$. By the $L-$Lipschitz condition for the first entry given any second entry of $\ell(\cdot, \cdot)$, we have $\mathfrak{R}_{nS}[\mathcal{A}] \leq L\mathfrak{R}_{nS}[\cup_{s=1}^{S} \mathcal{F} \circ \mathcal{H}^{(s)}]$ based on contraction principal [22] where $\mathcal{F} \circ \mathcal{H}^{(s)}$ denotes the function class considered in R^2 learning for each task. Combining all above results, we have

$$\mathcal{E}(\hat{\Theta}, \hat{\beta}) \leq 4L\mathfrak{R}_{nS}[\cup_{s=1}^{S} \mathcal{F} \circ \mathcal{H}^{(s)}] + \sqrt{\frac{2B^2 \log(2/\delta)}{n}},$$

with probability at least $1-\delta$. Therefore, all we need to finish is the upper bound of the Rademacher complexity of the function class considered in the R^2 learning. First of all, we define the empirical Rademacher complexity, whose upper bound will be established:

$$\hat{\mathfrak{R}}_{nS}[\cup_{s=1}^S \mathcal{F} \circ \mathcal{H}^{(s)}] = \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\Theta \in \mathcal{F}, \beta^{(s)} \in \mathcal{H}^{(s)}} \frac{1}{nS}\sum_{s=1}^S \sum_{i=1}^n \sigma_{s,i} \langle \Theta(\mathbf{x}_i^{(s)}), \beta^{(s)}\rangle \Big| (\mathbf{x}_i^{(s)}, y_i^{(s)}), i \in [n], s \in [S]\right],$$

where $\sigma_{s,i}$ are i.i.d Rademacher random variables. Based on the definition of $G_\beta(\boldsymbol{\sigma})$ in the proof of Lemma 4.1, the empirical Rademacher complexity is equivalent to $\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\beta \in \mathcal{H}} G_\beta(\boldsymbol{\sigma})\right]$ for fixed realization of $(\mathbf{x}_i^{(s)}, y_i^{(s)}), i \in [n], s \in [S]$. Due to the structure of $\mathcal{H}$, we have the following derivations:

$$\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\beta \in \mathcal{H}} G_\beta(\boldsymbol{\sigma})\right] \leq \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\beta \in \text{conv}(\mathcal{H})} G_\beta(\boldsymbol{\sigma})\right]$$

$$\overset{(i)}{=} \mathbb{E}_{\boldsymbol{\sigma}}\left[\max_{\beta \in \text{ext}(\text{conv}(\mathcal{H}))} G_\beta(\boldsymbol{\sigma})\right]$$

$$= \int_0^\infty \mathbb{P}(\max_{\beta \in \text{ext}(\text{conv}(\mathcal{H}))} G_\beta(\boldsymbol{\sigma}) \geq t)dt$$

$$= \int_0^{\mathbb{E}_{\boldsymbol{\sigma}} G_\beta(\boldsymbol{\sigma}) + \delta} \mathbb{P}(\max_{\beta \in \text{ext}(\text{conv}(\mathcal{H}))} G_\beta(\boldsymbol{\sigma}) \geq t)dt + \int_{\mathbb{E}_{\boldsymbol{\sigma}} G_\beta(\boldsymbol{\sigma}) + \delta}^{\infty)} \mathbb{P}(\max_{\beta \in \text{ext}(\text{conv}(\mathcal{H}))} G_\beta(\boldsymbol{\sigma}) \geq t)dt$$

$$\leq \mathbb{E}_{\boldsymbol{\sigma}} G_\beta(\boldsymbol{\sigma}) + \delta + \sum_{\beta \in \text{ext}(\text{conv}(\mathcal{H}))} \int_\delta^\infty \mathbb{P}(G_\beta(\boldsymbol{\sigma}) - \mathbb{E}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma}) \geq t)dt$$

$$\leq \mathbb{E}_{\boldsymbol{\sigma}} G_\beta(\boldsymbol{\sigma}) + \delta + (2D)^S \int_\delta^\infty \exp(-\frac{nSt^2}{2B^2\alpha^2})dt$$

$$\leq \mathbb{E}_{\boldsymbol{\sigma}} G_\beta(\boldsymbol{\sigma}) + \delta + (2D)^S \frac{\exp(-\frac{nS\delta^2}{2B^2\alpha^2})}{\frac{nS\delta}{B^2\alpha^2}}.$$

Note that $\text{conv}(\mathcal{H}) = \{(\beta^{(1)}, \cdots, \beta^{(S)}) \mid \beta^{(1)} \in \mathcal{H}^{(s)}\}$ and $\text{ext}(\text{conv}(\mathcal{H}))$ contains all sparse coordinates in S sources with cardnality $(2D)^S$. The equation (i) holds for linear functions on convex compact sets. By letting $\delta = \sqrt{\frac{2B^2\alpha^2 S \log(2D)}{nS}}$, we can conclude the results in Theorem 4.1. $\qquad\square$

D.4   Proof of Lemma 4.2 and Corollary 4.1

Following the similar steps as in the proof of Lemma 4.1, we have the following inequalities for $\mathbb{E}_{\boldsymbol{\sigma}}[G_\beta(\boldsymbol{\sigma})]$:

$$\mathbb{E}_{\boldsymbol{\sigma}}[G_\beta(\boldsymbol{\sigma})] = \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\Theta\in\mathcal{F}}\frac{1}{nS}\sum_{s=1}^{S}\sum_{i=1}^{n}\sigma_{s,i}\left[\sum_{m=1}^{M}\mathbb{I}_m^{(s)}\sum_{d=1}^{D}\theta_{m;d}(\mathbf{x}_{i;m}^{(s)})\beta_{m;d}^{(s)}\right]\right]$$

$$\leq \frac{1}{nS}\sum_{m=1}^{M}\sum_{d=1}^{D}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\theta_{m;d}}\sum_{s\in I_{m;d}}|\beta_{m;d}^{(s)}|\sum_{i}\sigma_{s,i}\theta_{m;d}(\mathbf{x}_{i;m}^{(s)})\right]$$

$$\leq \frac{1}{nS}\sum_{m=1}^{M}\sum_{d=1}^{D}\alpha\mathbb{E}\left[\sup_{\theta_{d;m}}\sum_{s\in I_{m;d}}(1-g_{m;d}^{(s)}(\tau_m))\sum_{i}\sigma_{s,i}\theta_{m;d}(\mathbf{x}_{i;m}^{(s)})\right]$$

$$+\frac{1}{nS}\sum_{m=1}^{M}\sum_{d=1}^{D}\tau_m\mathbb{E}\left[\sup_{\theta_{d;m}}\sum_{s\in I_{m;d}}g_{m;d}^{(s)}(\tau_m)\sum_{i}\sigma_{s,i}\theta_{m;d}(\mathbf{x}_{i;m}^{(s)})\right].$$

By the same reasoning as in the proof of Lemma 4.1, we can obtain the desired upper bounds. And the excess risk bound in Corollary 4.1 can be derived based on this results and techniques in the proof of Theorem 4.1.