Learning Biologically Interpretable Latent Representations from Gene Expression Data

Ioulia Karagiannaki

Thesis submitted in partial fulfillment of the requirements for the

Masters' of Science degree in Computer Science and Engineering

University of Crete
School of Sciences and Engineering
Computer Science Department
Voutes University Campus, 700 13 Heraklion, Crete, Greece

Thesis Advisor: Prof. *Ioannis Tsamardinos*

This work has been performed at the University of Crete, School of Sciences and Engineering, Computer Science Department.

University of Crete
Computer Science Department

Learning Biologically Interpretable Latent Representations from Gene Expression Data

Thesis submitted by
Ioulia Karagiannaki
in partial fulfillment of the requirements for the
Masters' of Science degree in Computer Science

THESIS APPROVAL

Author: _____
Ioulia Karagiannaki

Committee approvals: _____
Ioannis Tsamardinos
Professor, Thesis Supervisor

_____
George Tziritas
Professor, Committee Member

_____
Yannis Pantazis
Assistant Researcher, Committee Member

Departmental approval: _____
Antonios Argyros
Professor, Director of Graduate Studies

Heraklion, August 2020

Learning Biologically Interpretable Latent Representations from Gene Expression Data

Abstract

Gene expression data are typically high dimensional with low sample size. This leads to several statistical and analytical challenges that one needs to overcome in order to analyze and infer the underlying biological mechanisms of such data. To this end, several dimensionality reduction techniques have been proposed. Dimensionality reduction techniques learn a lower dimensional space (latent space), of newly constructed features and represent the data as a sum of those (latent representations). The projection of the data to the latent feature space compresses the data, retains the significant information and reduces noise.

Typical dimensionality reduction techniques, such as Principal Component Analysis, derive latent representations that are uninterpretable biologically. In order to regain a degree of interpretability, other methods return sparse latent representations. Particularly, the new features are constructed as linear combinations of only a few of the molecular quantities. However, sparse latent representations are still hard to interpret biologically as they do not directly correspond to the known biological pathways or other known genesets.

In this thesis, we present a novel algorithm for feature construction and dimensionality reduction called Pathway Activity Score Learning (PASL). The major novelty of PASL is that the constructed features are constrained to directly correspond to known molecular pathways and can be interpreted as pathway activity scores. PASL is evaluated both on simulated and real data. We show that PASL retains the predictive information for disease classification on new, unseen datasets. We also show that differential activation analysis provides complementary information to standard geneset enrichment analysis.

Μαθαίνοντας βιολογικά ερμηνεύσιμες κρυφές αναπαραστάσεις από δεδομένα γονιδιακών εκφράσεων

Περίληψη

Τα δεδομένα γονιδιακής έκφρασης είναι κατά κύριο λόγο πολυδιάστατα με πολύ μικρό αριθμό δειγμάτων. Αυτό οδηγεί σε στατιστικές και μεθοδολογικές προκλήσεις, οι οποίες πρέπει να αντιμετωπιστούν για την περεταίρω ανάλυση και κατανόηση των υποκείμενων βιολογικών μηχανισμών που υπάρχουν σε δεδομένα αυτού του τύπου. Γι΄ αυτό το σκοπό, έχουν προταθεί μέθοδοι μείωσης διαστάσεων, οι οποίες μαθαίνουν ένα χώρο χαμηλότερης διάστασης (κρυφός διανυσματικός χώρος) που αποτελείται από νέες μεταβλητές και αναπαριστούν τα αρχικά δεδομένα ως άθροισμα αυτών (κρυφές αναπαραστάσεις). Η προβολή των αρχικών δεδομένων στον κρυφό διανυσματικό χώρο συμπιέζει τα δεδομένα, ενώ διατηρεί τη σημαντική τους πληροφορία και μειώνει το θόρυβο.

Οι κλασσικές τεχνικές μείωσης διαστάσεων, όπως η ανάλυση κυρίων συνιστωσών (PCA), βρίσκουν κρυφές αναπαραστάσεις οι οποίες δεν είναι βιολογικά ερμηνεύσιμες. Για την καλύτερη βιολογική ερμηνεία, έχουν προταθεί νέες μέθοδοι οι οποίες βρίσκουν αραιές κρυφές αναπαραστάσεις. Συγκεκριμένα, οι νέες μεταβλητες μπορούν να αναπαρασταθούν ως γραμμικός συνδιασμός μικρού πλήθους των αρχικών μοριακών ποσοτήτων. Όμως και πάλι η ερμηνεία των αραιών κρυφών αναπαραστάσεων δεν είναι πλήρως κατανοητή, διότι οι ήδη υπάρχουσες τεχνικές μαθαίνουν αραιές κρυφές αναπαραστάσεις, οι οποίες δεν αντιστοιχούν άμεσα στα ήδη γνωστά βιολογικά μονοπάτια ή σε άλλα γνωστά σύνολα γονιδίων.

Σε αυτή την εργασία, θα παρουσιάσουμε μία νέα τεχνική δημιουργία νέων μεταβλητών και μείωση διαστάσεων που ονομάζεται Pathway Activity Score Learning (PASL). Η βασική καινοτομία της μεθόδου PASL είναι ότι ο κρυφός διανυσματικός χώρος που επιστρέφει, είναι βιολογικά ερμηνεύσιμος καθώς εφαρμόζονται περιορισμοί έτσι ώστε να αντιστοιχεί σε γνωστά βιολογικά μονοπάτια. Ο έλεγχος της ορθότητας της μεθόδου γίνεται τόσο σε συνθετικά, όσο και σε πραγματικά δεδομένα. Δείχνουμε ότι η μέθοδος PASL διατηρεί την προβλεπτική ικανότητα των αρχικών δεδομένων. Επίσης η εύρεση διαφορικά εκφραζόμενων βιολογικών μονοπατιών δίνει επιπρόσθετη πληροφορία σε αναλύσεις εμπλουτισμού γονιδίων.

Acknowledgements

First of all, I would like to express my gratitude to my supervisor Professor Ioannis Tsamardinos and to my co-advisor Dr. Yannis Pantazis for their constructive ideas, continuous feedback and patience throughout these years. This thesis would not have been completed without their continuous help and guidance. I am also grateful to Professor Ekaterini Chatzaki for giving me the biological intuition behind this thesis.

I would like to thank all my colleagues at the Mens X Machina Lab for their friendship and support during these years. Especially, I would like to thank my closest friends from the Lab: Konstantina Biza, Thodoris Giakoumakis, Tasos Tsourtis and Glykeria Fragkioudaki (Group Therapy) for all the the helpful discussions, encouragement during good and bad times.

I am also grateful to Elias Papadopoulos for his endless patience and support through all these years (Ohana). I would also like to thank my sisters Katerina and Despoina for all the support and encouragement. Last but not least, I am grateful to my mother, Maria, for supporting me and believing in me.

This thesis is dedicated to the memory of my father, Manolis, who left us in the early days of this Masters, but he would be proud.

Στην οικογενειά μου

Contents

Table of Contents **i**

List of Tables **iii**

List of Figures **v**

1 Introduction **1**
1.1 Motivation 1
1.2 Literature 1
1.3 Proposed Solution 2
1.4 Evaluation of the proposed solution 3
1.5 Contributions 3
1.6 Related Publications 4
1.7 Outline 4

2 Theoretical Background **5**
2.1 Gene expression data 5
2.2 Dimensionality Reduction 6
2.2.1 Principal Component Analysis 6
2.2.2 Sparse Principal Component Analysis 7

3 Pathway Activity Score Learning Algorithm **9**
3.1 Preliminaries 9
3.2 Inference Phase 10
3.3 Discovery Phase 15

4 Selection of Hyperparameters' Value **17**
4.1 Effect of t on the explained variance and the execution time 17
4.2 Box-Cox normalization of the variance 18

5 PASL Evaluation on Simulated Data **21**
5.1 Evaluation of the Inference Phase 21
5.2 Evaluation of the Discovery Phase 22

6 PASL evaluation on Real Gene Expression Data **27**
6.1 Tools and Methods 27
6.1.1 Datasets 27
6.1.2 Provided and Discovered Genesets. 28
6.1.3 Disease Classification 28
6.1.4 Pathway Level Information ExtractoR 29
6.1.5 Gene Set Enrichment Analysis 29
6.1.6 Differential Activation Analysis 30
6.2 Constructing a Latent Feature Space with PASL and PLIER . . . 30
6.2.1 Predictive Performance in Latent Feature Space 31
6.3 From GSEA to DAA 34

7 Conclusions **39**
7.1 Conclusion 39
7.2 Future Work 40

8 Additional Information **41**

Bibliography **45**

List of Tables

3.1 Basic Definitions of PASL 10
3.2 Further definition for PASL that is used in Algorithm 1. 13

5.1 Evaluation of the inference phase. Summarized information. 23
5.2 Evaluation of the discovery phase. Summarized information 25

6.1 AUC of the test datasets for PASL, PLIER and Original space (initial test datasets). PASL and PLIER are tested for approximately equal number of non-zero entries in the dictionary matrix. For Breast cancer data PASL's latent space consists of 500 dimensions-664695 non-zeros. PLIER's latent space consists of 29 dimensions of 699976 non-zeros. For Leukemia, PASL's latent space consists of 500 dimensions of 700020 non-zeros. PLIER's latent space consists of 30 dimensions of 782114 non-zeros. 33

8.1 Phenotype information for Breast cancer test datasets. 42
8.2 Phenotype information for Leukemia test datasets. 42

List of Figures

2.1 The basic idea of PCA 7

2.2 PCA can be thought in two ways. It learns a latent space at which the data variance is maximized (left) or equivalently it learns a latent space where the the residuals are minimized (right). 7

3.1 Overview of PASL. PASL outputs a dictionary D which is constructed in two phases. During the inference phase, D_1 corresponds to the genesets of geneset matrix G that best explain the data variance. During the discovery phase, D_2 is not restricted to correspond to the prior biological knowledge, so it captures the remaining variance and reveals potentially new biological knowledge. 11

3.2 Overview of the dynamic approach. The construction of the first atom of the dictionary. The same procedure can be repeated for the construction of the rest of the atoms. 12

3.3 Overview of the predefined order of genesets. 13

3.4 Overview of the inference phase of PASL. 15

4.1 The explained variance (y-axis) as a function of the execution time (x-axis) is shown for different values of t. For $0.4 \leq t \leq 0.9$, the execution time is reduced by a percentage between 65% and 85% with minimal impact on the explained variance. 18

4.2 The simulated dictionary (ground truth; 7th bar) consists of equally distributed pathways with different number of genes ,[10, 30, 50, 70] genes for the top figure, [30, 50, 100, 300] and [30, 50, 70, 200, 300] genes for the middle and bottom figure respectively). In all three figures, the 6th bar (without normalization) shows the distribution of selected pathways without normalization, while the rest of the bars show the selected pathways for different values of the Box-Cox normalization parameter λ. Apparently, the normalization of the variance is necessary for PASL in order to avoid being biased towards selecting genesets with a larger number of genes. 20

5.1 Percentage of correctly identified atoms *PCIA* for increasing SNR. For $SNR \geq 5dB$ PASL identifies almost all genesets that generated the data. 22
5.2 Number of genesets and probesets per database. In GPL570 there are 54675 probesets in total where the 23696 of them belong to at least one geneset of the databases. 24
5.3 **(a)** *PCIA* of visible genesets in G^v (i.e., provided to PASL as prior knowledge) for increasing ratio of visible to hidden atoms $\frac{V}{H}$. PASL is able to correctly identify most known atoms even in the unknown genesets. **(b)** *PCIA* for the hidden genesets in G^h. When the known genesets are a multiple of the unknown genesets, PASL is able to correctly discover the unknown genesets. 24

6.1 Experimental Setup. For the construction of the latent feature space, the methods are trained on a collection of gene expression datasets. The evaluation is performed on new unseen test datasets, where the predictive performance and the significance of the pathways of the latent feature space are examined. 28
6.2 Mean AUC of **(a)** Breast Cancer and **(b)** Leukemia test datasets . 31
6.3 **(a)** The best visualization for PASL vs Original, **(b)** The best visualization for PASL vs PLIER (The outcome stands for the mutation status of IGHV gene and **(c)** The best visualization for PLIER vs PASL. 32
6.4 Cumulative plots of the enriched and differentially activated pathways.The x axis represents the total number of significant genesets. The y axis shows the cumulative interaction of DAA and GSEA. . 34
6.5 Box-plots of the activation scores that correspond to the first, second, third differentially activated PASL feature/pathway. It is verified that the differentially activated pathways behave differently between the phenotypes. The outcome of GSE10780 stands for Invasive Ductal Carcinoma/Unremarkable breast ducts, and the outcome of GSE15434 stands for the mutation status of Nucleophosmin 1 (NPM1). 35
6.6 Out of sample probability of test datasets reduced to their top 3 differentially activated but not enriched pathways. The outcome of GSE10780 stands for Invasive Ductal Carcinoma/Unremarkable breast ducts, and the outcome of GSE15434 stands for the mutation status of Nucleophosmin 1 (NPM1). 36

Chapter 1

Introduction

1.1 Motivation

Molecular data, such as gene expressions, are often very high dimensional, measuring tens of thousands molecular quantities. For example, the Affymetrix microarray platform GPL570 for humans measures the expressions of 54675 probe-sets, corresponding to all known human genes. As such, visually inspecting the data, understanding the multivariate gene correlations, and biologically interpreting the measurements is challenging. To address this problem, several methods have appeared that reduce the dimensionality of the data. Dimensionality reduction (a.k.a. latent representation learning) constructs new dimensions (features, quantities, variables). The purpose is to reduce the number of features making them amenable to inspection while maintain all "useful" information. For example, consider the representation of music. The raw data (original measured quantities) correspond to the sound spectrum which is visually incomprehensible to humans. However, music at each time-point can be represented as a sum of prototypical states (notes) and musical scores, which are much more intuitive [41]. Similarly, we can ask the questions: Are there prototypical cell states whose sum can represent any cell state (e.g., gene expression profile)? What are the "notes" of biology? How can we learn such representations automatically?

1.2 Literature

In order to overcome the statistical challenges numerous dimensionality reduction techniques have been proposed. Some of the most prevalent ones are arguably the PCA, Kernel PCA [44], t-SNE [31], and Neural Network autoencoders [23] and others [24, 48]. All of these methods learn a lower dimensional space (latent space) of newly constructed features and represent the data as a linear combination of those. The projection to the latent space aims to retain the data variance and exhibit a low data reconstruction error. However, the data representation in the new feature space is biologically unintepretable. To improve interpretability other

methods introduce sparsity to the latent space in the sense that new features are constructed as linearly combinations of only a few of the original molecular quantities. Such methods are the Sparse PCA [57] and sparse variants of Non-negative Matrix Factorization [30] for molecular data [10, 21]). The new constructed features are sometimes called *meta-genes* [9]. Any clustering method could also be defined as creating meta-genes and new features. However, *the meta-genes are still hard to interpret biologically as they do not directly correspond to the known biological pathways or other known gene sets.*

Other methods aim to find a sparse representation of the input data (sparse coding) in the form of a linear combination of basic elements as well as those basic elements themselves [3, 20, 11]. The latent space is often called dictionary and the basic elements are called atoms. There are also extensions of such methods that try to incorporate prior information in the sense that they construct the dictionary in a supervised manner. Specifically, the atoms of the dictionary are constructed by the input data and by the class labels [54, 55, 26].

Pathway Level Information extractor (PLIER) [33] constructs a dictionary in an unsupervised manner and also incorporates prior knowledge. The prior knowledge corresponds to the known biological genesets. Each element of PLIER's dictionary corresponds to a relevant subset of the available genesets and also aims to maintain the significant information of the data. The interpretation of a dictionary which corresponds to the available genesets under a soft constraint is still a difficult task. PLIER is furtherly explained in Section 6.1.4.

1.3 Proposed Solution

In this work, we develop a novel method for unsupervised feature construction and dimensionality reduction based on the availability of prior knowledge, called Pathway Activity Score Learning or **PASL**. PASL aims at a trade-off between biological interpretability, and computational performance. PASL accepts as input a collection of predefined sets of genes, hereafter called **genesets**, such as molecular pathways or gene ontology groups. It has two phases, the *inference phase* and the *discovery phase*. During the inference phase, **PASL constructs new features that are constrained to directly correspond to the available genesets.** The new features could be thought as **activity scores** of the corresponding genesets. The inference phase ends when it has captured as much information as possible (maximum explained variance) given only the provided genesets. However, a large percentage of the measured quantities is not mapped to any known genesets. In the discovery phase, PASL constructs features that are not constrained to correspond to the given genesets trying to capture the remaining information (variance) in the data.

1.4 Evaluation of the proposed solution

We evaluate PASL in two sets of computational experiments. (a) We use two collections of real micro-array gene expression datasets, one for Breast Cancer and one for Leukemia. *It is shown that PASL learns latent representations that allow it to perform predictive modeling based on the novel features. The computational experiments are performed on test datasets never seen by PASL during feature construction.* For predictive modeling we use an AutoML platform for molecular data called Just Add Data Bio or **JADBIO** [49] that searches thousands of machine learning pipelines to identify the optimally predictive one and estimates the out-of-sample predictive performance of the final model in a conservative fashion. *Analysis in the new feature space is orders of magnitude faster than the one performed using the original feature space.* In addition, the resulting predictive models are on par and often outperform the ones constructed using the original molecular quantities. PASL is compared against PLIER [33], arguably the algorithm closer in spirit to PASL. PASL outperforms PLIER in terms of predictive performance.

In the second set of computational experiments, (b) we show that PASL's constructed features can complement standard gene set enrichment analysis (**GSEA**). Specifically, the geneset activity scores output by PASL can be employed to perform differential activation analysis (**DAA**) and identify the genesets that behave differently between two different classes (e.g., cases vs controls, or treatment vs controls). Conceptually, this is equivalent to gene differential expression analysis that identifies genes whose expression behaves differently in two classes. Our experiments indicate that DAA complements GSEA: it can identify genesets that are not identified by GSEA as statistically significant. Moreoever, DAA has larger statistical power than GSEA and, in general, it identifies the affected genesets with lower p values than GSEA.

1.5 Contributions

In this master thesis, we present a novel algorithm for interpretable dimensionality reduction of gene expression data, called PASL. The biological interpretation of PASL, comes from the fact that it learns a latent feature space which directly corresponds to prior biological knowledge, i.e., the biological pathways. The main contributions of this work is (a) the construction of PASL, (b) which is trained on a collection of datasets, creating a disease-specific latent space, able to identify differences between multiple outcomes (e.g. disease or mutation status, relapse/diagnosis etc.). (c) There are multiple studies that use dimensionality reduction methods as a preprocessing step prior to disease classification [4, 18, 39, 42, 32]. Specifically, in these studies the dimensionality reduction techniques are applied into a single dataset, which is then split into train and test set, the predictive model is trained on the train set and it is evaluated on the test set. Unlike the

above process, the latent space of PASL is tested on new unseen test datasets, which are never seen during training. The training on multiple datasets and testing on unseen datasets implies that the constructed latent space is robust and able to generalize [47, 12, 40]. (d) The evaluation of PASL is performed both on simulated and real data. We perform extensive experiments in order to tune PASL's hyperparameters and also to check the quality of the outcome (disease classification, differential activation analysis).

1.6 Related Publications

The Pathway Activity Score Learning Algorithm introduced in this work has been summarized in the following original publication, which has resulted from this thesis.

- Karagiannaki, I., Pantazis, Y., Chatzaki, E., Tsamardinos, I. (2020, October). Pathway Activity Score Learning for Dimensionality Reduction of Gene Expression Data. In International Conference on Discovery Science (pp. 246-261). Springer, Cham.

1.7 Outline

The remainder of this thesis is structured as follows. Chapter 2 presents the necessary preliminaries on gene expression data and dimensionality reduction. In Chapter 3 we present PASL. In Chapter 4 we point out the importance of the hyperparameters of PASL. In Chapter 5 we present some experiments of PASL applied on simulated data. In Chapter 6, PASL is evaluated on real gene expression data. Chapter 7 concludes this thesis and points out further extensions of this work. Finally, Chapter 8 contains additional information about the data phenotypes.

Chapter 2

Theoretical Background

In this chapter we provide background information regarding gene expression data. We furtherly provide the general idea of dimensionality reduction, as well as some general information about two widely used dimensionality reduction techniques, PCA and SPCA, which are also used in the method that we propose in Chapter 3.

2.1 Gene expression data

Gene expression is the process by which the genetic code (the nucleotide sequence) of a gene is used to direct protein synthesis and produce the structures of the cell [15]. Gene expression is measured with a fast and automated manner through high-throughput technologies [53]. The main characteristic of gene expression datasets is that they are high dimensional with low sample size. There are several reasons for the limited availability of samples. For example, the limited number of patients due to rare diseases and privacy or financial constraints limit the creation of new samples. The high dimensionality combined with the low sample size makes gene expression data impractical to use because of the high computational requirements. Also, there are many correlated variables in gene expression data [38], and thus there is redundant information [36] which can be eliminated by applying dimensionality reduction techniques in gene expression data [7, 36, 5].

In this study we use microarray data [43] from Affymetrix Human Genome U133 Plus 2.0 Platform (GPL570) consist of 54675 features. A typical microarray dataset may consist of tens to hundreds of samples. The fact that the number of features exceeds the number of samples is known as the "curse of dimensionality" problem and the efficient dimensionality reduction of such data plays a crucial role for further analysis.

2.2 Dimensionality Reduction

Dimensionality reduction is the transformation of data from a high dimensional space into a lower dimensional space so that the the significant information of the original data is maintained. Dimensionality Reduction techniques are used in various domains, such as signal processing, image processing, neuroinformatics, bioinformatics etc [6, 56, 45, 17, 52]. This kind of techniques are mainly used for noise reduction, data visualization, cluster analysis, or as an intermediate step for further analysis (e.g predictive analysis). Concisely, the main goals of dimensionality reduction is (a) to extract the most important information of the data, (b) to compress the data requiring smaller storage, (c) analyse and find structures of the data, which were not accessible in the original space. In the following subsections (2.2.1 and 2.2.2), we present two widely used dimensionality reduction techniques, PCA and Sparse PCA, which are also utilized by the method that we mainly present in this thesis.

2.2.1 Principal Component Analysis

The most widely used dimensionality reduction technique is the Principal Component Analysis (PCA) [1]. PCA performs an orthogonal linear transformation to the data, converting it to lower dimensional data with linearly uncorrelated variables.

The main idea of PCA is that it projects the data to the directions that capture the highest data variance (Figure 2.1). The dimensional space that consists of these directions is called the latent feature space, and its elements are called loadings or principal axes. The data that are projected into PCA's latent space are also known as PCA scores. The PCA problem can be mathematically formulated as follows

$$\begin{aligned} \max_{A} \quad & A^T X^T X A \\ \text{s.t} \quad & \mathrm{A}^T A = I \end{aligned} \tag{2.1}$$

The classic approach to maximize the objective function of 2.1 would be to compute the eigenvalues of $X^T X$ (the sample covariance matrix of dimension $p \times p$, where p is the number of dimensions/features) and set A to the eigenvectors associated with the largest eigenvalues (sorted from maximum to minimum). For gene expression data, where p is very large, it turns out to be impractical to solve. An efficient way to solve the optimization problem 2.1 is through Singular Value Decomposition (SVD). SVD returns a decomposition of the data X, such that $X = USV^T$, where the right singular vectors V are equal to the eigenvectors of the sample covariance matrix, hence equal to the PCA loadings.

So PCA estimates a linear subspace A, such that when the data are projected on A, the data variance is maximized. Every new feature could be expressed as

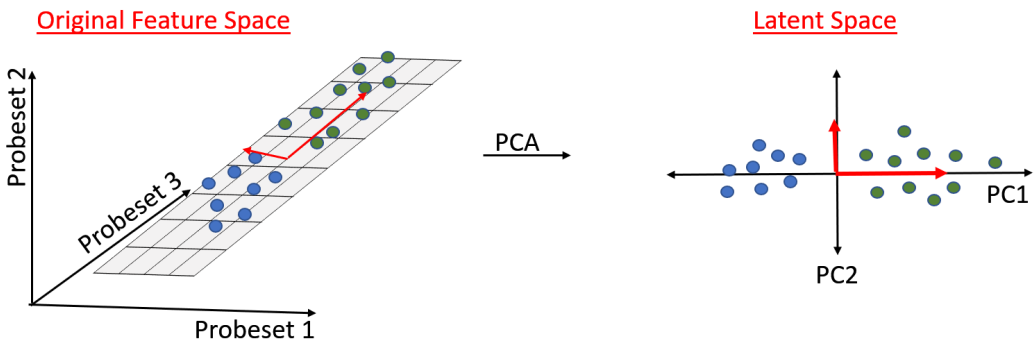

Figure 2.1: The basic idea of PCA

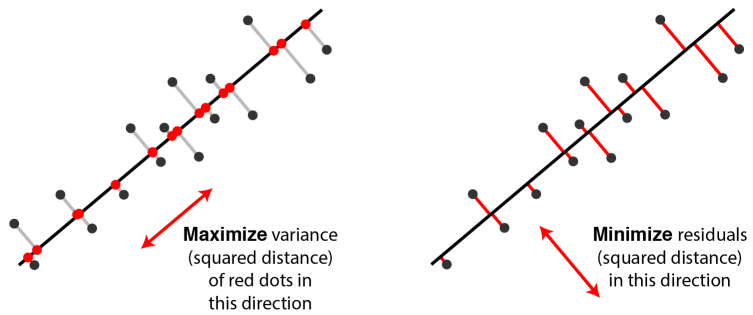

Figure 2.2: PCA can be thought in two ways. It learns a latent space at which the data variance is maximized (left) or equivalently it learns a latent space where the the residuals are minimized (right).

a linear combination of all the initial features, which in many cases is not interpretable. Specifically, in gene expression data, each new feature is a linear combination of all tens of thousands of gene probes. For this reason sparse dimensionality reduction techniques are introduced. In Section 2.2.2, we briefly explain an extension of PCA, i.e. the sparse PCA.

2.2.2 Sparse Principal Component Analysis

Sparse dimensionality reduction techniques construct a sparse latent space. In this case the results are more interpretable. Specifically, every new feature is a linear combination of only a small number of the original features. A commonly used sparse dimensionality reduction technique is Sparse Principal Component analysis (SPCA). SPCA could be thought as an extension of the PCA optimization problem with some extra sparsity and regularization terms.

In order to connect the classic PCA with SPCA, one should consider the optimization problem of SPCA from a different perspective. Instead of maximizing the data variance, it is equivalent to think PCA as an approach of finding a linear subspace that minimizes the distance of the projection in a least-squares sense (Figure 2.2). This could be expressed as

$$\begin{aligned} &\underset{A}{min}||X - XAA^T||_F^2 \\ &\text{s.t} \quad \mathrm{AA}^T = I \end{aligned} \tag{2.2}$$

The SPCA optimization problem can be formulated by applying the elastic net penalty as follows

$$\begin{aligned} &\underset{A,B}{min}||X - XBA^T||_F^2 + \sum_{j=1}^{k} ||\beta_j||_F^2 + \sum_{j=1}^{k} \lambda_{i,j}||\beta_j||_1 \\ &\text{s.t} \quad \mathrm{AA}^T = I \end{aligned} \tag{2.3}$$

where B is the learnt sparse latent space.

Chapter 3

Pathway Activity Score Learning Algorithm

3.1 Preliminaries

The PASL algorithm accepts as input two 2D matrices X and G. Matrix $X \in \mathbb{R}^{n \times p}$ contains the molecular measurements, where n is the number of samples and p the number of features. Typically $n \ll p$. For microarray gene expression data, the rows of X correspond to molecular profiles while the columns to the gene expressions of the probe-sets. Hereafter, *we will refer to probe-sets as genes for simplicity, unless otherwise noted; however, the reader is warned that there is not a one-to-one correspondence between probe-sets and genes.* PASL also accepts a gene membership matrix $G \in \{0,1\}^{g \times p}$ with g being the number of predefined groups of genes. Each row of G, denoted by $\mathbf{g}_i$ for the i-th row, corresponds to a molecular pathway, gene ontology set, or any other predefined gene collection of interest called **geneset** hereafter. We set $G_{ij} = 1$ if gene j belongs to the i-th geneset, and 0 otherwise.

PASL assumes the data X can be decomposed as

$$X = L \cdot D + \sigma I, \tag{3.1}$$

where $D \in \mathbb{R}^{a \times p}$ is a sparse matrix. In other words, each molecular profile at row j of X is a linear combination of rows of D with coefficients in the jth row of L with an isotropic noise added to it. D is called the **dictionary** and its rows the dictionary **atoms**, denoted with $\mathbf{d}_i$. In PCA terminology, D corresponds to the *loadings*. The geneset matrix (G) quantifies the prior knowledge that helps PASL to define the dictionary atoms in a way that the results are interpretable. Given training data X and the geneset matrix G, PASL outputs the two matrices D and L. D is the concatenation of two sub-dictionaries D_1 and D_2 ($D = [D_1; D_2]$) with dimensions $a_1 \times p$ and $a_2 \times p$, respectively (hence, $a = a_1 + a_2$). *D_1 is a dictionary where each atom $\mathbf{d}_i$ is constrained to correspond to only one geneset*, in the sense that the non-zero elements of $\mathbf{d}_i$ correspond to the genes in the particular geneset.

	Metric	Symbol	Dimension
Dimensions	#samples	n	$\mathbb{R}$
	# features	p	$\mathbb{R}$
	# atoms	a	$\mathbb{R}$
	# genesets	g	$\mathbb{R}$
	# genes per geneset	K	$\mathbb{R}$
PASL Input	Data	X	$\mathbb{R}^{n\times p}$
	Geneset matrix	G	$\mathbb{R}^{g\times p}$
	Normalization parameter	λ	$\mathbb{R}$
	Threshold for reordering genesets	t	$\mathbb{R}$
PASL Output	Dictionary	D	$\mathbb{R}^{a\times p}$
	Representation of data in D	L	$\mathbb{R}^{n\times a}$

Table 3.1: Basic Definitions of PASL

Thus, D_1 is the part of the dictionary that is biological interpretable. D_2 is just a sparse dictionary meant to explain the remaining variance of the data and suggest the existence of yet-to-be-discovered genesets. D_1 is the outcome of the first phase of PASL, called **inference phase**, while D_2 is the outcome of the second phase, called the **discovery phase**. $L \in \mathbb{R}^{n\times a}$ is the representation of the data in the latent feature space, corresponding to PCA *scores*. PASL provides the optimal projection of X on the row space of D and it is computed by minimizing the Frobenius norm between X and $L \cdot D$. The basic definitions and the corresponding dimensions are summarized in Table 3.1. Also an overview of PASL is presented in Figure 3.1.

3.2 Inference Phase

During the inference phase, PASL constructs a dictionary (D_1), whose atoms correspond to the genesets of the geneset matrix G. Particularly, the atoms of D_1 correspond to the genesets that best explain the data variance.

One approach to extract the genesets with the highest variance in the dataset is through a dynamic heuristic, where for each new atom, one needs to answer the following question: Which is the geneset that leads to the "next best" atom? One way to answer to this question is to reduce the data matrix to the features (genes) that correspond to a geneset, estimate the first principal component, repeat the same for all genesets and then keep the principal component ($\mathbf{d}$) with the highest variance. We mathematically formulate this problem as

$$i^* = \underset{i=1,\ldots,g}{\arg\max} \max_{\mathbf{d}\in\mathbb{R}^{||\mathbf{g}_i||_0}} ||X(:,\mathbf{g}_i)\mathbf{d}||_2^2 \tag{3.2}$$

where $X(:,\mathbf{g}_i)$ denotes the data matrix reduced to the genes of the i-th geneset.

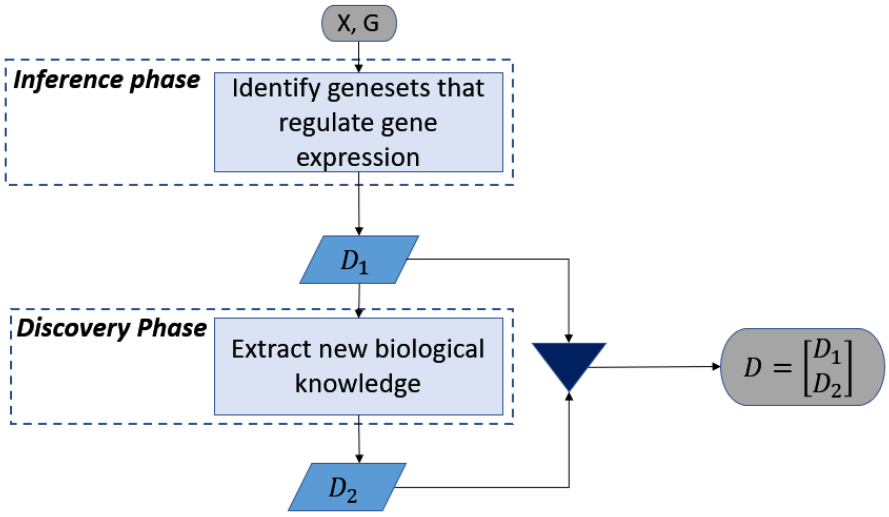

Figure 3.1: Overview of PASL. PASL outputs a dictionary D which is constructed in two phases. During the inference phase, D_1 corresponds to the genesets of geneset matrix G that best explain the data variance. During the discovery phase, D_2 is not restricted to correspond to the prior biological knowledge, so it captures the remaining variance and reveals potentially new biological knowledge.

Then, we add the i^* principal component to the dictionary, remove its contribution from the dataset and repeat the same procedure until a pre-specified criterion is met. We note that each atom is of dimension $1 \times p$, where p is the number of features of the data. The indexes of the non-zero entries of an atom correspond to the genes of the i^*-th geneset. The non-zero entries correspond to the first principal component of the i^*-th geneset. So $D_{ij} = 0$ indicates that gene j does not belong to the i-th geneset. An example of this dynamic approach is shown in Figure 3.2.

Algorithm 1 Dynamic Approach

1: Reduce the data matrix to the genes of each geneset of G ($X(:, g_i)$)
2: i^*: index of geneset whose 1st principal component has the highest variance
3: Add the i^* geneset to the dictionary and remove its contribution from the data
4: Repeat until convergence or until the dictionary has the maximum number of atoms

The described algorithm (Algorithm 1) is guaranteed to return an ordered dictionary whose atoms have the highest variance. Nevertheless, it can be prohibitively expensive in terms of computational cost since at each iteration it performs one PCA per geneset. In total it has to compute $a \cdot g$ PCAs. In order to remedy the computational burden, the dictionary could be constructed though a

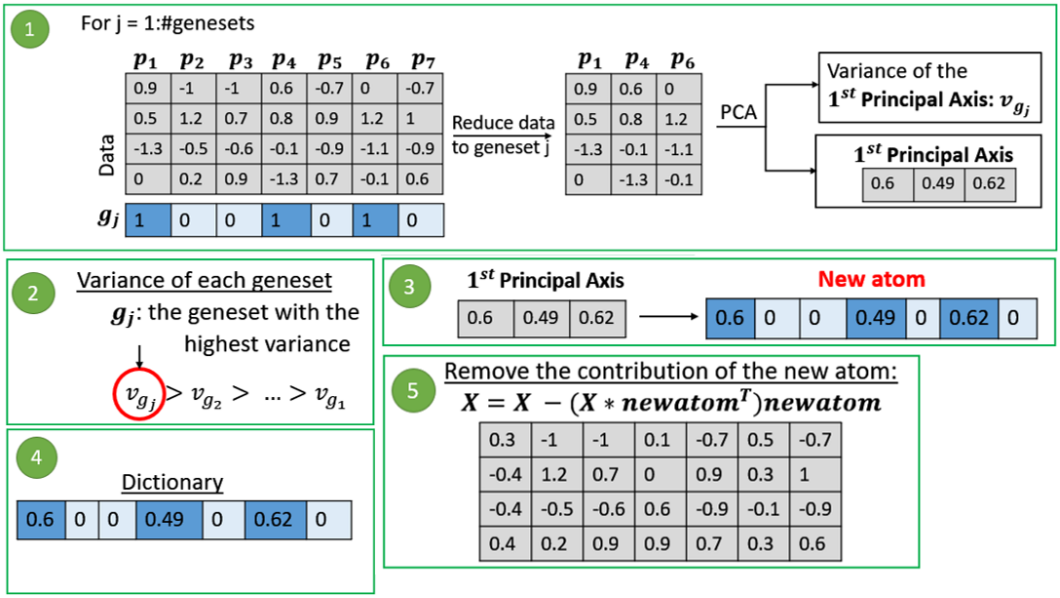

Figure 3.2: Overview of the dynamic approach. The construction of the first atom of the dictionary. The same procedure can be repeated for the construction of the rest of the atoms.

static heuristic. Specifically, one could precompute the principal components for all reduced-to-genesets data matrices, then, order them and keep the genesets with the highest variance (Algorithm 2). Figure 3.3 shows an example of the described process. The atoms of the dictionary are constructed based on this predefined order of genesets. This static approach is way more computational efficient because the number of PCAs that one has to perform is $g+a$. Specifically, g PCAs are performed in order to precompute the ordering of the genesets, and one PCA for each new atom that is added to the dictionary. Despite being computational efficient, this approach may affect the quality of the solution. Specifically, the ordering of the genesets is fixed, but at each iteration the data matrix changes because the contribution of each new atom is removed from it. In other words, the order of genesets is predefined based on the original data, without taking into account the variance that the previous atoms explain. This might affect the actual ordering of the variance, hence the quality.

The dynamic and the static approach return the same solution when the genesets of the geneset matrix do not have common genes. In this case, the contribution of each new atom does not affect the rest of the atoms, so the static approach produces the same solution with the dynamic approach.

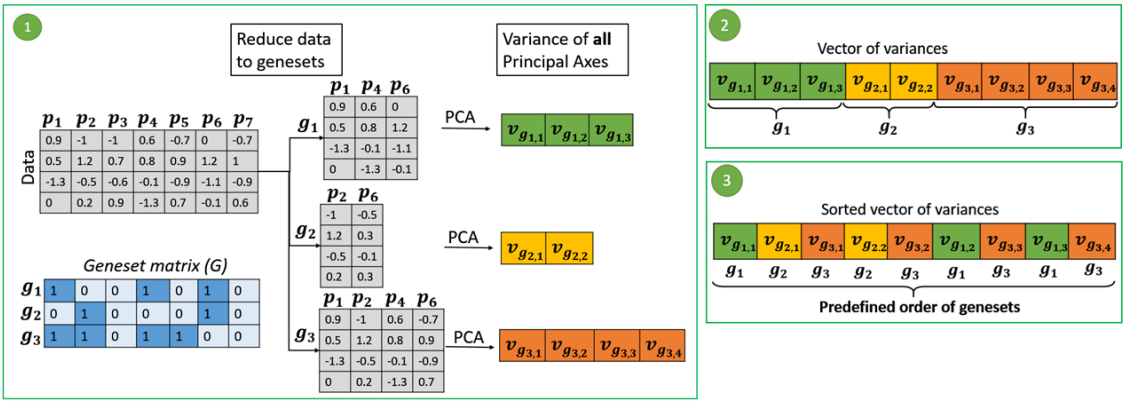

Figure 3.3: Overview of the predefined order of genesets.

Algorithm 2 Static Approach

1: Reduce the data matrix to the genes of each geneset of G ($X(:, g_i)$)
2: Apply PCA on **each** $X(:, g_i)$ and keep the variance of **all principal components of all genesets** ($v_G \in \mathbb{R}^{g \cdot min(n,K)}$)
3: $[\sim, idx] = sort(v_G, descend)$
4: Based on idx ordering add the genesets to the dictionary while removing their contribution from the data
5: Keep adding genesets until convergence or until the dictionary has the maximum number of atoms

Metric	Symbol	Dimension
z-scored Data	X_z	$\mathbb{R}^{n \times p}$
Ordered geneset matrix	$\overline{G}$	$\mathbb{R}^{g \times p}$
Data reduced to pathway	X_r	$\mathbb{R}^{n \times K}$
1st principal axis of reduced data	d_r	$\mathbb{R}^{K}$
Variance of reduced data to the selected genesets (prior)	v_{G_s}	$\mathbb{R}^{a}$
Variance of all genesets	v_G	$\mathbb{R}^{a \cdot g}$
Variance of PCA on reduced data	v_r	$\mathbb{R}^{\min(n,K)}$
Relative Reconstruction Error	v_z	$\mathbb{R}^{a}$
Indexes of all genesets	id_G	$\mathbb{R}^{a \cdot g}$
Indexes of selected genesets	id_{G_s}	$\mathbb{R}^{a}$
Threshold of convergence	tol	$\mathbb{R}$

Table 3.2: Further definition for PASL that is used in Algorithm 1.

The inference phase of PASL, which is shown in Algorithm 3 (lines 1–22 and

Algorithm 3 Pathway Activity Score Learning

Input: Data $X_{n\times p}$, Geneset Matrix $G_{g\times p}$
Output: Dictionary $D_{a\times p}$, Representation of data in D: $L_{n\times a}$

//Inference Phase
$X_z \leftarrow zscore(X)$
$X \leftarrow X_z$
$i \leftarrow 1$, $i' \leftarrow 1$ //i: running geneset index, i': atom counter
$[i_{\bar{G}}, v_{\bar{G}}] \leftarrow$ OrderOfGenesets(X, G) //$v_{\bar{G}}$: pre-computed variance
$\bar{G} \leftarrow G(i_{\bar{G}}, :)$ //$\bar{G}$: ordered geneset matrix
while $i' \leq a_1$ **do**
  $X_r \leftarrow X(:, \bar{\mathbf{g}}_i)$
  $[\mathbf{d}_r, v_r] \leftarrow pca(X_r,\ \#pc = 1)$ //v_r : current variance
  if $\frac{v_r}{v_{\bar{G}}(i)} \leq t$ **then** //how close is v_r to $v_{\bar{G}}(i)$
    $[i_{\bar{G}}, v_{\bar{G}}] \leftarrow$ OrderOfGenesets(X, G)
    $\bar{G} \leftarrow G(i_{\bar{G}}, :)$
    $i \leftarrow 1$ //Reset counter
    $X_r \leftarrow X(:, \bar{\mathbf{g}}_i)$
    $[\mathbf{d}_r, v_r] \leftarrow pca(X_r, \#pc = 1)$
  end if
  $D_1 \leftarrow [D_1; expand(\mathbf{d}_r; \mathbf{g}_i)]$ //Insert the new atom in D_1
  $X \leftarrow X\big(I - D_1(i, \mathbf{g}_i)^T D_1(i, \mathbf{g}_i)\big)$ //Remove the contribution
  $v_z \leftarrow ||X_z(I - D_1^+ D_1)||_F^2 \ / \ ||X_z||_F^2$
  if $|v_z - v_{z-1}| < tol$ **then** break **end if**
  $i \leftarrow i + 1$, $i' \leftarrow i' + 1$
end while
//Discovery Phase
$X_z \leftarrow zscore(X)$
$D_2 \leftarrow spca(X_z,\ \#pc = a_2,\ \#nz = m)$ //$a_2 = a - i'$
$D \leftarrow [D_1; D_2]$
$L \leftarrow X_z D^+$
return D, L

---

function OrderOfGenesets(X, G)
  $v_G \leftarrow \emptyset$, $i_G \leftarrow \emptyset$
  for $i \leftarrow 1$ **to** g **do**
    $X_r \leftarrow X(:, \mathbf{g}_i)$
    $[\sim, v_r] \leftarrow pca\big(X_r,\ \#pc = \min(n, ||\mathbf{g}_i||_0)\big)$
    $v_G \leftarrow \left[v_G; \frac{\lambda \cdot v_r}{(||\mathbf{g}_i||_0^{\lambda} - 1)}\right]$ //Box-Cox normalization
    $i_G \leftarrow [i_G | i | ... | i]$ //Insert $\min(n, ||\mathbf{g}_i||_0)$ elements
  end for
  $[v_{\bar{G}}, j] \leftarrow \mathrm{sort}(v_G)$
  $i_{\bar{G}} \leftarrow i_G(j)$
  return $i_{\bar{G}}, v_{\bar{G}}$ //ordered genesets ids and their corresponding variance
end function

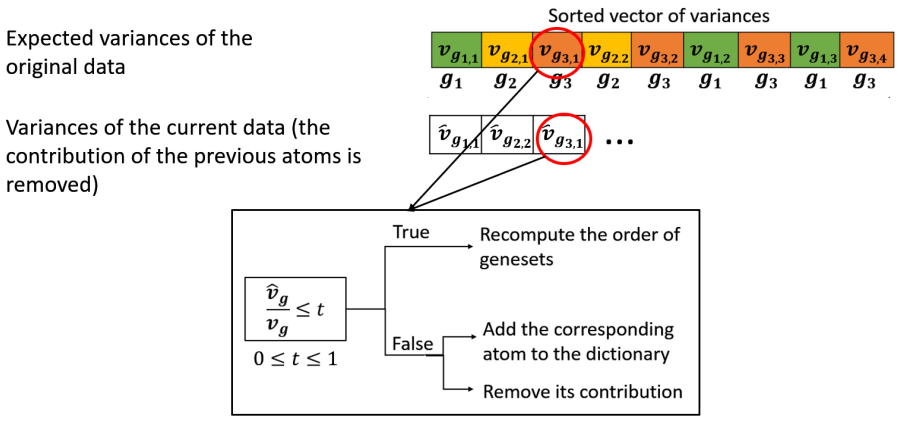

Figure 3.4: Overview of the inference phase of PASL.

29–40), balances between the dynamic and the static approach. As in the static approach, it computes the ordering of the principal components' variance (lines 5 and 29–40) and iteratively selects the atoms based on this ordering (while loop; lines 7–22). The difference between PASL and the static approach is that PASL checks how close is the current variance from the expected pre-computed variance (line 10). If the relative change is below a threshold, PASL recomputes the ordering of the principal components' variance (lines 11–15). The hyper-parameter t, which takes values between $[0, 1]$, controls how often the variance reordering is performed henceforth the proximity to optimality. The higher the value of t, the more often the evaluation of the ordering is happening, thus the dictionary is more accurate in terms of explained variance, on the cost of being computationally more expensive. The stopping criterion asserts that the inference phase of PASL stops when there is no further decrease in the relative reconstruction error (e.i., the variance of the normalized residual error) (line 20). Finally, we remark that the variance values are normalized before they are ordered (line 34). This is absolutely necessary due to the wide variation of the number of genes in each geneset which varies from few dozens to few thousands of genes. We choose as normalization method the Box-Cox transformation on the number of genes and optimize over its hyper-parameter λ. An overview of the inference phase of PASL is shown in Figure 3.4 and a brief definition the parameters and their corresponding dimension is summarized in Table 3.2.

3.3 Discovery Phase

After the inference phase where we extracted as much as possible variance from prior knowledge, we will distill the remaining variance of the data without restrictions on the location of the non-zero elements of the dictionary atoms using a sparse –hence, interpretable– dimensionality reduction technique aiming to reveal

new potential pathways which were previously unknown. Based on its generality and efficiency, we employ in our experiments Sparse Principal Component Analysis (SPCA) [57] (lines 24–25 in Algorithm 3). We note though that any sparse dimensionality reduction technique can be utilized. However, we do not tune the respective hyper-parameters, instead, we require the SPCA algorithm to return a fixed number of non-zero elements per atom. We denote this number with m and we set it to 2000 in our experiments. SPCA is furtherly explained in Section 2.2.2.

Chapter 4

Selection of Hyperparameters' Value

PASL takes as input several hyperparameters that need to be specified. The hyperparameter t, which controls how often the function OrderOfGenesets will be called, has a direct impact on the execution time of PASL. λ is a normalization parameter, which is critical for the output of PASL.

There are also hyperparameters which we do not tune, i.e. the hyperparameter m (the number of non-zero elements at each atom of the discovery phase) as well as the number of atoms at the discovery phase. The latter two hyperparameters originate from SPCA.

In Sections 4.1, 4.2 we define and show the importance of the hyperparameters t and λ respectively.

4.1 Effect of t on the explained variance and the execution time

Due to the large number of PCA calculations (one for each geneset), the most time-consuming part of PASL is the execution of the function OrderOfGenesets in Algorithm 3. The hyper-parameter t controls how often the function OrderOfGenesets will be called. When $t = 1$ (dynamic approach) then it is called at every iteration, while it is called once at the beginning and never again when $t = 0$ (static approach). In order to determine the optimal value for t, we perform an experiment with a merged collection of microarray datasets where the total number of samples is $n = 4235$, the number of genes $p = 54675$ and a fixed number of atoms $a_1 = 200$. Fig. 4.1 demonstrates the explained variance as a function of the execution time for different values of t. Based on this plot, we observe that the value of t is crucial both for the execution time of PASL and for the explained variance, so we set t to be equal to 0.9 (cyan star symbol in Fig. 4.1) in our experiments.

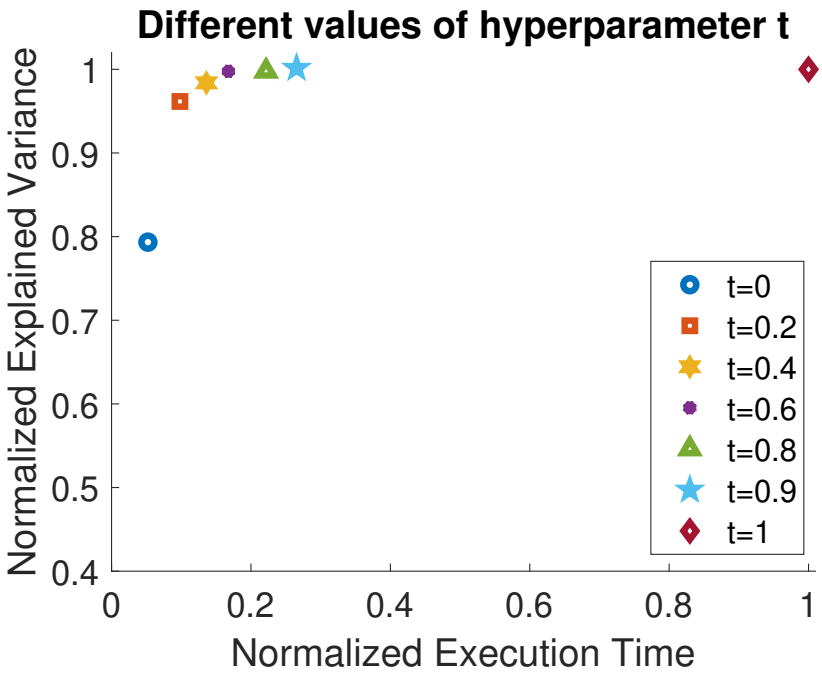

Figure 4.1: The explained variance (y-axis) as a function of the execution time (x-axis) is shown for different values of t. For $0.4 \leq t \leq 0.9$, the execution time is reduced by a percentage between 65% and 85% with minimal impact on the explained variance.

4.2 Box-Cox normalization of the variance

The number of genes, i.e., the number of non-zero elements in each row of the geneset matrix G, varies from few dozens to several thousands making the geneset ordering based on variance susceptible to such variations. Indeed, we experimentally observe that genesets with more genes tend to be selected frequently while genesets with a low number of genes were rarely selected (see also the "without normalization" bar of Figure 4.2). Therefore, it is essential to normalize the variance of each geneset relative to the number of genes it contains. We propose to normalize the variance using the Box-Cox transformation [8] on the number of genes (i.e., on $||\mathbf{g}_i||_0$) which is given by

$$y' = \begin{cases} (y^\lambda - 1)/\lambda & \text{if } \lambda \neq 0 \\ \log(y) & \text{if } \lambda = 0 \end{cases} \tag{4.1}$$

where λ is a tunable hyper-parameter which controls the power scaling on y.

The value of λ is determined by a targeted experiment using simulated data which are generated using genesets with both small and large number of genes. The simulated data are generated by first creating the geneset matrix G consisting

of equally distributed genesets with specific number of genes. Then, we construct a dictionary using randomly selected genesets which are also equally distributed. Specifically, we create $n = 400$ samples with $p = 500$ features and perform multiple experiments for different number of genes per geneset. The number of genes per geneset take values [10, 30, 50, 70] (Figure 4.2 (top)), [30, 50, 100, 300] (Figure 4.2 (middle)) and [30, 50, 70, 200, 300] (Figure 4.2 (bottom)). After an extensive search (for different number of genes per geneset, different number of atoms, and different dimensions of the simulated data), where different values of Box-Cox transformation hyper-parameter was tested, we set λ to be equal to 1/3.Some of the geneset selection results obtained with PASL are presented in Fig. 4.2. Evidently, the use of Box-Cox transformation with $\lambda = 1/3$ (4th bar) produced results similar to the ground truth (7^{th}bar) while PASL without normalization fail to correctly infer the true dictionary (6th bar).

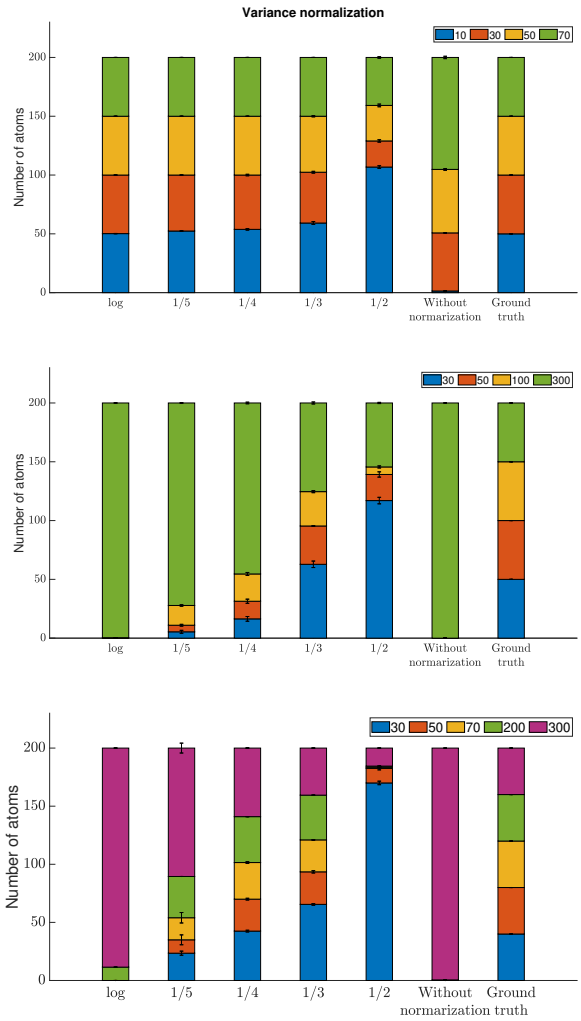

Figure 4.2: The simulated dictionary (ground truth; 7th bar) consists of equally distributed pathways with different number of genes ,[10, 30, 50, 70] genes for the top figure, [30, 50, 100, 300] and [30, 50, 70, 200, 300] genes for the middle and bottom figure respectively). In all three figures, the 6th bar (without normalization) shows the distribution of selected pathways without normalization, while the rest of the bars show the selected pathways for different values of the Box-Cox normalization parameter λ. Apparently, the normalization of the variance is necessary for PASL in order to avoid being biased towards selecting genesets with a larger number of genes.

Chapter 5

PASL Evaluation on Simulated Data

This set of experiments demonstrates the ability of PASL to learn atoms that correspond to the correct genesets. To this end, we simulate data where the latent space is known and used as the gold standard. There are four parts in the simulation: (a) generating the geneset matrix G, (b) generating the dictionary matrix (latent feature space) D, (c) generating the score matrix L, and (d) simulating the data X. In the geneset matrix G each row corresponds to a geneset, so that $G_{ij} = 1$ implies that gene j belongs to geneset i. *We denote with g_i the ith row of G*, i.e., the ith geneset. G is randomly sampled so that each g_i has exactly K genes (ones), where K is a simulation parameter. Next we randomly sample a dictionary matrix D as follows: the kth row of D, will correspond to a randomly chosen (with replacement) geneset j. Its index is stored in vector id so that $id_k = j$. D_k contains zero coefficients for the genes that do not belong in geneset j. The remaining coefficients are randomly sampled from a uniform distribution $U[-1.5, 1.5]$. The entries of the score matrix L are again randomly sampled from a uniform distribution $U[-1.5, 1.5]$. The simulated data are then computed as:

$$X = L \cdot D + N \tag{5.1}$$

where N is a matrix of noise, specifically each element is sampled from a Gaussian distribution with mean 0, and standard deviation such that the signal to noise ratio achieves value SNR.

5.1 Evaluation of the Inference Phase

In the first group of experiments, we evaluate the inference phase of the algorithm. The geneset matrix G that was used to generate the data is given as input to PASL, i.e., all genesets are provided as prior knowledge. So, given X and G, PASL learns a dictionary $\hat{D}$ of atoms in each row $\hat{\mathbf{d}}_k$ that corresponds to a list of geneset ids $\hat{id}$. When an identified atom $\hat{\mathbf{d}}_k$ and an atom used to generate the data $\mathbf{d}_m$ correspond

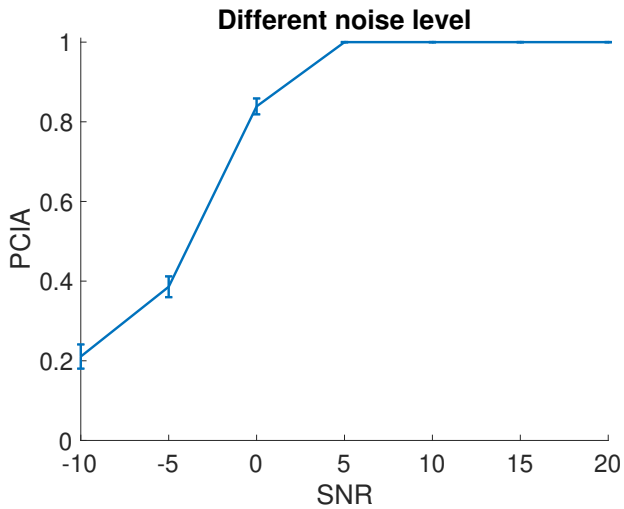

Figure 5.1: Percentage of correctly identified atoms *PCIA* for increasing SNR. For $SNR \geq 5dB$ PASL identifies almost all genesets that generated the data.

to the same geneset, we say that $\hat{\mathbf{d}}_k$ is correctly identified. The learning quality of PASL is measured by the percentage of atoms in $\hat{D}$ correctly identified:

$$PCIA = \frac{\#(id \cap \hat{id})}{\#id} \tag{5.2}$$

where *PCIA* stands for *percentage of correctly identified atoms*. Notice that id and $\hat{id}$ are treated as *multi-sets*, in other words, if geneset j is used in 5 atoms of D, we would expect PASL to also have 5 atoms in $\hat{D}$ that correspond to geneset j. For example, if the data were generated from atoms in D corresponding to genesets with ids $id = [1, 2, 3, 4, 5, 1]$ and PASL identifies $\hat{D}$ such that its rows correspond to genesets with ids $\hat{id} = [6, 7, 8, 5, 1, 1]$. Then, $PCIA = \frac{3}{6} = 0.5$. Fig. 5.1(a) shows the average (over 20 runs) *PCIA* for increasing values of *SNR*: as the signal becomes stronger, PASL is able to correctly identify all genesets that participate in the generation of the data. For this experiment, the sample size was set to 400, the feature size to 500, 150 atoms were used for D chosen among 600 genesets in G, and the number of genes per pathway K was set to 100. Also the normalization parameter λ was set to $\frac{1}{3}$ and the hyperparameter t was set to 0.9. In Table 5.1 we summarize the inputs and outputs of PASL for this set of experiments.

5.2 Evaluation of the Discovery Phase

In this set of experiments, we relax the assumption that the data have been generated from known genesets. Specifically, we simulate two geneset matrices G^v with V rows and G^h with H rows, where the former corresponds to the *visible* genesets *that are provided to PASL as input*, and the latter to the *hidden* ones,

	Metric	Explanation	Dimension
Data Generation	G	Simulated geneset matrix	$\mathbb{R}^{g \times p}$
	D	Simulated Dictionary	$\mathbb{R}^{a \times p}$
	L	simulated scores matrix	$\mathbb{R}^{n \times a}$
Input	$X = LD$	Simulated Data	$\mathbb{R}^{n \times p}$
	G		$\mathbb{R}^{g \times p}$
Output	$\hat{D}$	Learnt Dictionary	$\mathbb{R}^{a \times p}$
	$\hat{L}$	Learnt scores matrix	$\mathbb{R}^{n \times a}$
Further Notation	id	indexes of genesets in D	
	$\hat{id}$	indexes of genesets in $\hat{D}$	

Table 5.1: Evaluation of the inference phase. Summarized information.

corresponding to the yet-to-be-discovered pathways. The data are generated from $G = [G^v; G^h]$. In GPL570 only 23696 probesets out of the 54675 ones belong in any of the genesets in KEGG, REACTOME, or Biocarta (Figure 5.2). To simulate a similar situation, genesets in G^v are simulated using only the first 50% of the features, while all features are allowed to participate in genesets of G^h. The data X are simulated from a dictionary $D = [D^v; D^h]$, where D^v (D^h) contains atoms that correspond to visible (hidden) genesets in G^v (G^h). We expect the inference phase of PASL to correctly infer D^v and the discovery phase to discover D^h.

Let us call an atom that corresponds to a visible (hidden) geneset in G^v (G^h) a visible (hidden) atom. We perform an experiment for different values of visible-to-hidden atoms ratio ($\frac{V}{H}$), keeping the number of visible atoms (V) fixed and equal to 100. The ratio takes values in $\{1, 2, 5, 10, 20, 100\}$. The sample size is 500, the feature size to 1000, the number of visible genesets is 600, and $SNR = 5db$. We consider a geneset in G^h_j as "discovered" if there is a corresponding atom in $\hat{D}^h$ returned by PASL that has at least 80% of its non-zero coefficients in G^h_j. Fig. 5.3(a) shows the *PCIA* for the visible genesets in G^v with increasing ($\frac{V}{H}$). The figure supports the claim that *PASL correctly constructs atoms corresponding to known genesets even in the presence of unknown genesets contributing to the data generation.* Fig. 5.3(b) shows the same metric for the discovery of atoms in G^h. When the number of known genesets is a multiple of the unknown ones, *PASL correctly identifies atoms corresponding to the unknown genesets.* In Table 5.2 we summarize the inputs and outputs of PASL for this set of experiments.

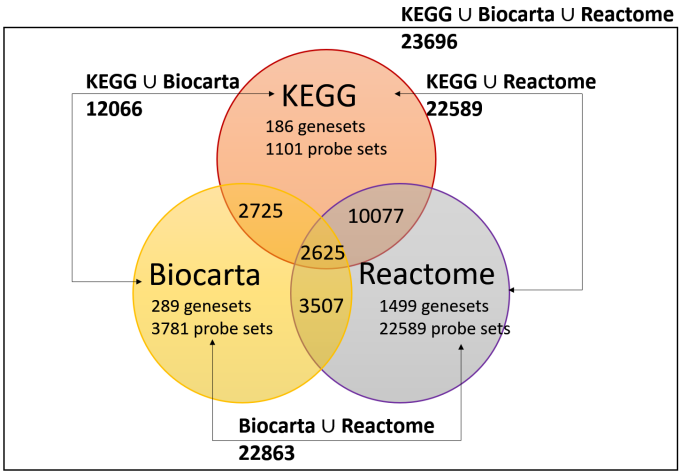

Figure 5.2: Number of genesets and probesets per database. In GPL570 there are 54675 probesets in total where the 23696 of them belong to at least one geneset of the databases.

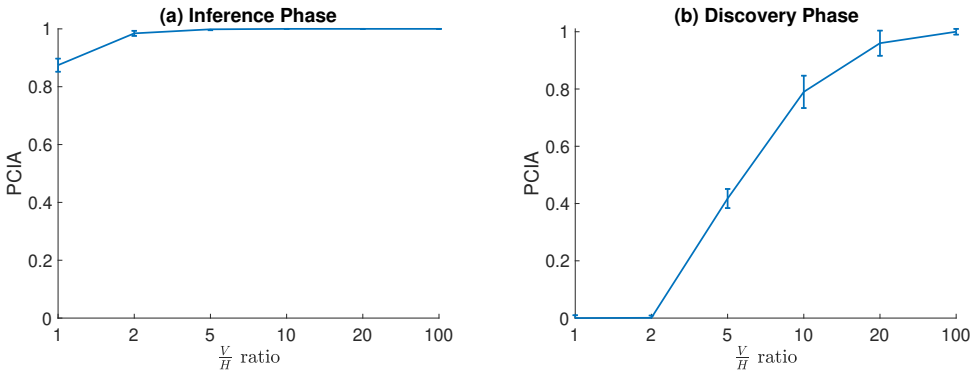

Figure 5.3: **(a)** *PCIA* of visible genesets in G^v (i.e., provided to PASL as prior knowledge) for increasing ratio of visible to hidden atoms $\frac{V}{H}$. PASL is able to correctly identify most known atoms even in the unknown genesets. **(b)** *PCIA* for the hidden genesets in G^h. When the known genesets are a multiple of the unknown genesets, PASL is able to correctly discover the unknown genesets.

	Metric	**Explanation**	**Dimension**
Data Generation	G^v	Simulated visible genesets	$\mathbb{R}^{V \times p}$
	G^h	Simulated hidden genesets	$\mathbb{R}^{H \times p}$
	D^v	Simulated visible Dictionary	$\mathbb{R}^{a_1 \times p}$
	D^v	Simulated hidden Dictionary	$\mathbb{R}^{a_2 \times p}$
	L	Simulated scores matrix	$\mathbb{R}^{n \times (a_1 + a_2)}$
Input	$X = L \cdot [D^v; D^h]$	Simulated Data	$\mathbb{R}^{n \times p}$
	G^v		$\mathbb{R}^{V \times p}$
Output	$\hat{D^v}$	Learnt Dictionary (inference phase)	$\mathbb{R}^{a_1 \times p}$
	$\hat{D^h}$	Learnt Dictionary (discovery phase)	$\mathbb{R}^{a_2 \times p}$
	$\hat{L}$	Learnt scores matrix	$\mathbb{R}^{n \times a}$
Further Notation	a_1	number of atoms (rows) of D^v	$\mathbb{R}$
	a_2	number of atoms (rows) of D^h	$\mathbb{R}$
	K	number of non-zeros per atom (inference)	$\mathbb{R}$
	m	number of non-zeros per atom (discovery)	$\mathbb{R}$

Table 5.2: Evaluation of the discovery phase. Summarized information

Chapter 6

PASL evaluation on Real Gene Expression Data

6.1 Tools and Methods

6.1.1 Datasets

For our experiments we downloaded datasets from BioDataome [29], a collection of uniformly preprocessed and automatically annotated datasets for data-driven biology (http://dataome.mensxmachina.org/). BioDataome hosts microarray gene expression datasets from the Gene Expression Omnibus database [13] and RNASeq gene expression datasets from the recount database [14].
Specifically, we downloaded all the available Breast cancer and Leukemia datasets as of May 2020 measured with the Affymetrix Human Genome U133 Plus 2.0 - GPL570 platform, each having at least 20 samples. The datasets form the *Breast Cancer collection* and *Leukemia collection*. For each collection we select 80% of the datasets to pool together and use them as training data. The dimensionality reduction algorithms are applied on this training set to learn a dictionary matrix D of atoms (Fig. 6.1(a)). For this experiment, PASL is compared against PLIER [33], which is arguably the algorithm closer in spirit to PASL. The remaining 20% of the available datasets are employed as test datasets and are *not seen by neither PASL or PLIER during training*. The selection of datasets to enter the train or the test set is random, with the restriction that test datasets have to be accompanied by a discrete outcome (phenotype) for each sample, e.g., disease or mutation status or multiple phenotypes related to the diseases (e.g. rapid/slow early responder). (Tables 8.1, 8.2). The outcome is either binary or multiclass. The training set for the Breast cancer collection contains 4200 unique gene-expression profiles, while the Leukemia collection contains 5600 unique profiles.

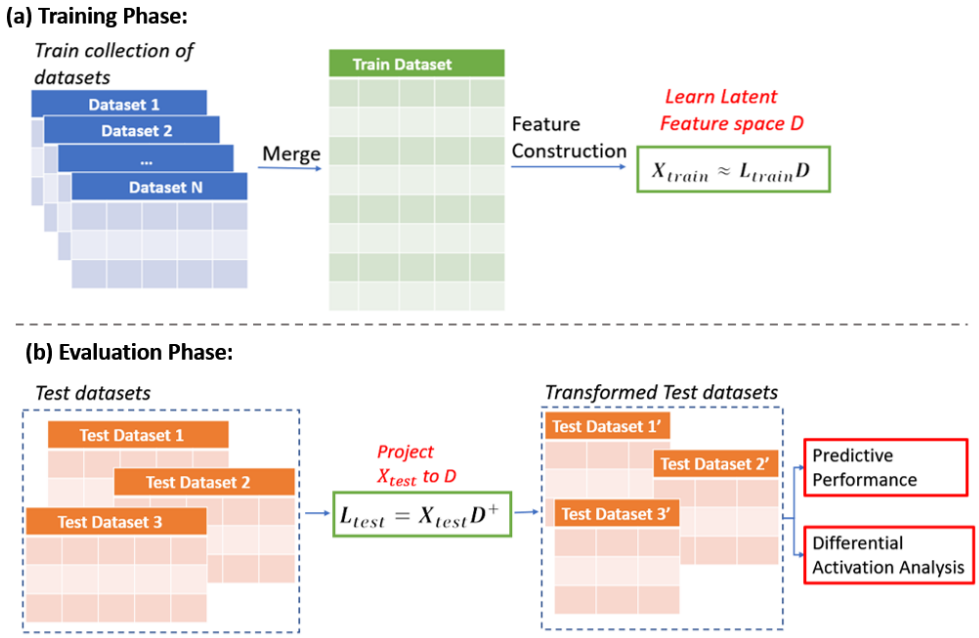

Figure 6.1: Experimental Setup. For the construction of the latent feature space, the methods are trained on a collection of gene expression datasets. The evaluation is performed on new unseen test datasets, where the predictive performance and the significance of the pathways of the latent feature space are examined.

6.1.2 Provided and Discovered Genesets.

In all experiments with real data, the geneset matrix G (gene membership matrix) includes 1974 pathways found in KEGG[27], Reactome [16] and Biocarta [35] which were downloaded from Molecular Signatures Database (MSigDB) of the Broad Institute [46]. The number of non-zeros for each atom in the discovery phase is set to 2000.

6.1.3 Disease Classification

For the disease classification experiments, we employ an automated machine learning architecture (AutoML), called **JADBIO** (Just Add Data Bio, www.jadbio.com), version 1.1.21. JADBIO has been developed specifically for small-sample, high-dimensional data, such as multi-omics data. The use of JADBIO is meant to ensure that (a) out-of-sample AUC estimates are accurate, and (b) performance does not depend on a single classifier tried with just the default hyper-parameters. Instead, for classification, JADBIO tries the SES feature selection algorithm [28], combined with ridge logistic regression, decision trees, random forests, and SVMs for modelling. It automatically tunes the hyper-parameters of the algorithms, trying thousands of combinations of algorithms and hyper-parameters. It estimates

the performance of the final winning model produced by the best configuration (pipeline of algorithms and hyper-parameter values) using the BBC-CV protocol [51]. The latter is a version of cross-validation that adjusts the estimate of performance of the winning configuration for multiple tries to provide conservative AUC estimates. A detailed description of the platform along with a massive evaluation on hundreds of omics datasets is in [49]. JADBIO has produced novel scientific results in nanomaterial prediction [50], suicide prediction [2], protein functional properties [37] and others.

6.1.4 Pathway Level Information ExtractoR

We comparatively evaluate PASL against a recently introduced algorithm called Pathway Level Information Extractor (PLIER) [33]. PLIER also accepts as input data X and a geneset matrix G. Similarly to PASL, it returns the scores L and the dictionary D, such that $X \approx L \cdot D$. PLIER imposes constraints to D to be a combination of known genesets. It is mathematically formulated as follows

$$\min_{L,D,B} ||X - LD||_F^2 + \lambda_1 ||D - BG||_F^2 - \lambda_2 ||L||_F^2 + \lambda_3 ||B||_1$$

$$\text{s.t} \quad \mathrm{D}_{ij} > 0, B_{ij} > 0 \tag{6.1}$$

B is a sparse matrix of coefficients which indicates which genesets from the geneset matrix G will correspond to each atom of D. PLIER accepts several hyper-parameters. The maxpath hyper-parameter indicates how many genesets an atom of D is supposed to correspond to. In other words, how many non-zeros per row the B matrix has. We set maxpath = 1 requesting that each atom in D corresponds to one and only geneset, so that the output is comparable to PASL. Unfortunately, *PLIER treats maxpath as indicative; atoms in D may correspond to the union of several genesets, even when maxpath = 1.* In that sense, the atoms in D are not as easy to interpret as the ones returned by PASL. PLIER also ignores genesets with features fewer than minGenes. We set minGenes = 1 so that no genesets are ignored. Finally, we note that in PLIER the scores L are computed as $X \cdot D^T \cdot (DD^T + \lambda_2 I)^{-1}$, where λ_2 is a parameter learned by the algorithm.

6.1.5 Gene Set Enrichment Analysis

A gene set is defined as a collection of genes associated with a specific biological process or disease, or even the set of genes that are present in a given pathway (e.g. a set of 128 genes is involved in the KEGG cell cycle pathway).

A common approach to interpret gene expression data based on the available genesets is the Gene Set Enrichment Analysis (GSEA) [46]. GSEA is a computational method to determine whether a pre-defined set of genes (geneset) shows a statistically significant difference between different phenotypes. GSEA first summarizes the probesets that correspond to the same gene to the maximum/minimum/mean expression value. Inherently, GSEA loses information when

this summarization is applied and it also loses information by ignoring the covariances of the gene expressions [22, 25]. Subsequently, the null hypothesis is that the p-values of the genes in the pathway have the same distribution as the genes that do not belong to the pathway.

6.1.6 Differential Activation Analysis

A widely used approach to analyse gene expression data is to identify the genes that behave differently between two or more conditions (e.g. disease vs. healthy, treatment vs. control etc.). This process is called differential expression analysis and the important genes are called differentially expressed genes. These differentially expressed genes are found by performing a multiple hypothesis testing.

In a similar fashion to differential expression analysis we perform **differential activation analysis (DAA)** on PASL's scores. Since the constructed features correspond the genesets (atoms of D), we can use their values (stored in the columns of L) to find which genesets behave differently under two conditions e.g,disease vs. healthy or treatment vs. control. Essentially, DAA estimates the genesets (pathways) that are *differentially activated*. We specifically perform DAA on the test datasets projected to the latent space of PASL (activity scores) using the Matlab's t-test function *mattest* with 10000 permutations [19].

6.2 Constructing a Latent Feature Space with PASL and PLIER

Applied to a training dataset X_{train}, PASL learns a transformation to a new feature space given data X_{train} and a geneset matrix G. Subsequently, PASL learns a dictionary D and scores L_{train} such that $X'_{train} \approx L_{train} \cdot D$. Each atom (row) in D corresponds to only one geneset in G or a newly discovered geneset (Fig. 6.1(a)). To apply the transformation to new test data X_{test}, one projects them to the row space of D by computing $L_{test} = X_{test} \cdot D^{+}$ (Fig. 6.1(b)). An important detail is that both the train and test data are first standardized using the means and standard deviations of the training data; thus, the transformation does not require any quantity to be estimated from the test data. This is important to avoid information leakage during cross-validation when evaluating predictive performance on the transformed data.

PLIER also accepts a the data X_{train} and a geneset matrix G and represents the data as $X'_{train} \approx L_{train} \cdot D$. The transformation to new test data is performed as $X \cdot D^T \cdot (DD^T + \lambda_2 I)^{-1}$, where λ_2 is a parameter learned by the algorithm. An atom in PLIER's latent space balances between the inference to the geneset matrix and the extraction of new biological knowledge. As a result, the atoms of PLIER are not as sparse as the ones that PASL outputs. For example, for the Breast Cancer collection analysis, the mean number of non-zero coefficients in each atom of PLIER is 25833 (almost half of the original feature size), while for PASL it is

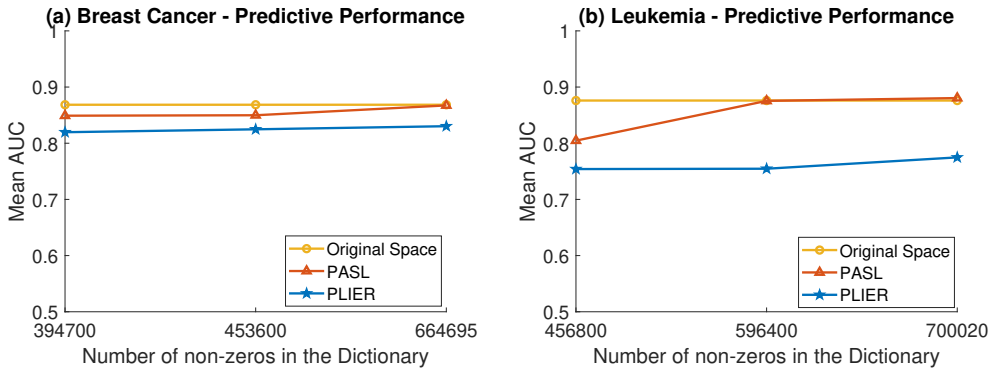

Figure 6.2: Mean AUC of **(a)** Breast Cancer and **(b)** Leukemia test datasets

1329. For the same number of atoms, PLIER has uses more degrees of freedom (non-zero coefficients) to find a suitable transformation to a latent space. For a fair comparison in the subsequent experiments, we impose the restriction that the learned dictionaries D_{PLIER} and D_{PASL} have approximately the same number of non-zero elements. To this end, we first run PLIER allowing it to construct a large number of atoms and estimate the number of atoms a required to reach approximately the same number of non-zeros as PASL. Then, we re-run PLIER constrained to produce only a atoms. Specifically, when PASL is restricted to 500 atoms, its dictionary contains 664695 and 700020 non-zeros for the Breast Cancer and the Leukemia collections, respectively. PLIER is limited to 29 and 30 atoms instead, producing dictionaries with 699976 and 782114 non-zeros, respectively.

6.2.1 Predictive Performance in Latent Feature Space

This set of experiments examines the following research question: *does the transformation to the latent feature space capture all important information*, defined as the information required to classify to typical outcomes (phenotypes) such as the disease state, the mutation status, dietary restrictions or other disease-specific phenotypes. To this end, we employ predictive modeling on the **test datasets** and estimate the predictive performance of the best identified model. Each test dataset's outcome leads to binary or multiclass classification tasks. Predictive performance is measured by the AUC metric.

We performed classification analysis using JADBIO (Section 6.1.3) on 13 and 15 test datasets for Breast Cancer and Leukemia, respectively. The analysis uses the original feature space, as well as the PLIER and PASL feature spaces, for different dimensionalities. For PASL, the number of atoms to learn take values 250, 400, and 500. The number of atoms with approximately the same non-zeros in the dictionary of PLIER is 20, 25, and 30. The dimension of the original feature space is 54675. Thus, there are 7 analyses for each dataset, and $91 + 105 = 196$ analyses in total. For the Breast Cancer datasets 860002 classification models

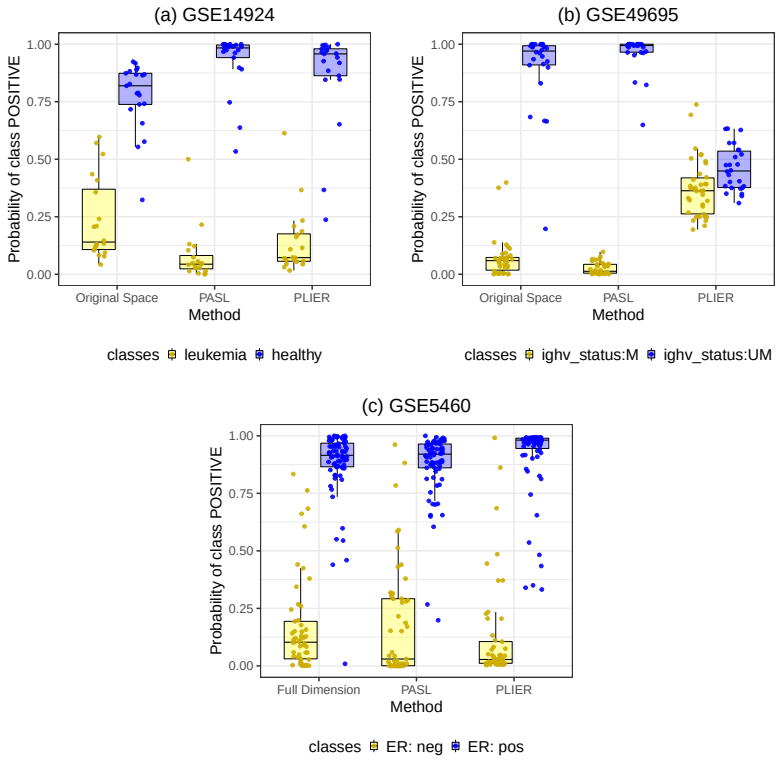

Figure 6.3: **(a)** The best visualization for PASL vs Original, **(b)** The best visualization for PASL vs PLIER (The outcome stands for the mutation status of IGHV gene and **(c)** The best visualization for PLIER vs PASL.

were trained in total by JADBIO with different combinations of algorithms and hyper-parameter values on different subsets of the input data (cross-validation). For the Leukemia, the number of trained models reaches 983425. In total 1843427 models were trained for this experiment.

Regarding the execution time, the analysis in the PASL or PLIER space takes about **1 order of magnitude less time** than in the original space. The exact execution time in JADBIO depends on several factors, such as the load of the amazon servers on which the platform runs, and thus exact timing results are meaningless. Indicatively, we mention a typical case: the analysis of GSE61804 for the original space took 1.15 hour, 9 minutes and 5 minutes for PASL and PLIER respectively. Figure 6.2(a),(b) shows the average AUC over all test datasets for each disease for increasing number of non-zeros. **PASL outperforms PLIER and it is on par with analyses on the original space**. Thus, the learned dictionary by PASL generalizes to new test data and captures the important information to perform classification with various disease-related outcomes. At the same time, *PASL achieves 2-orders of magnitude dimensionality reduction by a sparse matrix whose atoms directly correspond to known genesets (pathways).*

Table 6.1: AUC of the test datasets for PASL, PLIER and Original space (initial test datasets). PASL and PLIER are tested for approximately equal number of non-zero entries in the dictionary matrix. For Breast cancer data PASL's latent space consists of 500 dimensions-664695 non-zeros. PLIER's latent space consists of 29 dimensions of 699976 non-zeros. For Leukemia, PASL's latent space consists of 500 dimensions of 700020 non-zeros. PLIER's latent space consists of 30 dimensions of 782114 non-zeros.

Breast Cancer				**Leukemia**			
Data ID	**PASL**	**PLIER**	**Original**	**Data ID**	**PASL**	**PLIER**	**Original**
54002	0.999	1	0.995	15434	0.985	0.747	0.987
5460	0.952	0.958	0.96	14924	0.996	0.987	0.91
36771	0.935	0.933	0.963	23025	0.762	0.766	0.741
66161	0.664	0.486	0.579	21029	0.95	0.694	0.966
76124	0.976	0.98	0.97	28654	0.767	0.616	0.762
66159	0.759	0.506	0.776	14671	0.59	0.674	0.625
66305	0.513	0.569	0.535	7440	0.73	0.52	0.736
10780	0.976	0.995	0.962	66006	0.926	0.792	0.952
27562	0.835	0.776	0.914	28460	0.719	0.542	0.697
27830	0.725	0.671	0.759	26713	0.998	0.997	0.952
36769	0.953	0.963	0.96	31048	0.984	0.981	0.99
29431	0.997	0.982	0.991	39411	0.997	0.956	0.985
42568	0.991	0.975	0.927	49695	1	0.612	0.998
				50006	0.979	0.994	0.983
				61804	0.823	0.744	0.869
Mean	**0.8673**	**0.830**	**0.868**	**Mean**	**0.8804**	**0.7748**	**0.876**
Median	**0.952**	**0.958**	**0.96**	**Median**	**0.95**	**0.747**	**0.952**

We now focus on the experiments for the largest dimension of PASL and PLIER. The number of atoms in PASL is set to 500 (664695 non-zeros for Breast Cancer, 700020 non-zeros for Leukemia). PLIER's latent space consists of 29 (699976 non zeros) and 30 (782114 non-zeros) atoms for Breast Cancer and Leukemia respectively. Table 6.1 contains the detailed results for each dataset and method. The worst case (best case) for PASL is dataset with ID 27562 (14924) where it achieves 8 AUC points (8 AUC points) lower (higher) performance vs no dimensionality reduction. In contrast, there are several datasets (IDs 66161, 66159, 27562, 15434, 21029, 7440, 66006, 28460, 28460, 49695, 61804) where PLIER's performance is lower than 10 or more AUC points.

In the lower row of Fig. 6.3 we visually demonstrate the ability of PASL to lead to highly predictive models. Each panel correspond to a different test dataset. Specifically, we chose to present the visualizations from datasets that lead to the "best" visual differences for PASL vs the original space, PASL vs PLIER, and

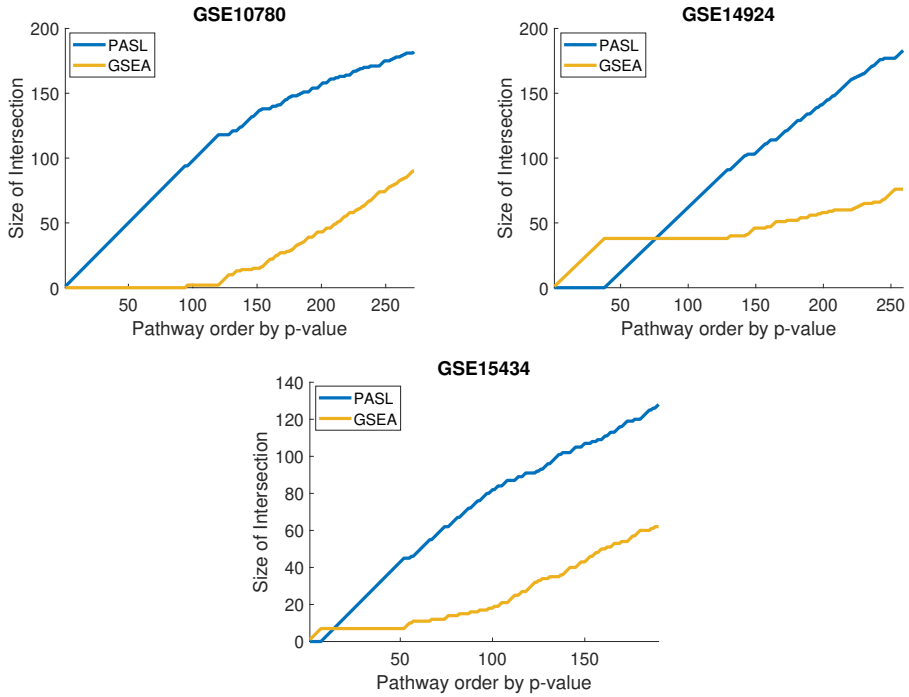

Figure 6.4: Cumulative plots of the enriched and differentially activated pathways.The x axis represents the total number of significant genesets. The y axis shows the cumulative interaction of DAA and GSEA.

PLIER vs PASL, in Fig. 6.3(a)–(c)), respectively. Each panel shows the box-plots of the *out-of-sample probability* of each molecular profile to belong to the positive class for the models produced in the original, PASL, and PLIER feature space. The out-of-sample predictions are calculated by JADBIO during the cross-validation of the winning model and thus, they do not correspond to the fitting of the samples used for training. The larger the separation of the distribution of the predicted probabilities, the larger the AUC.

6.3 From GSEA to DAA

The biological interpretability of PASL's feature space is demonstrated in the following experiments. Since the constructed features correspond to the genesets (atoms of D), we can use their values (stored in the columns of L) to find which genesets behave differently under two conditions, e.g., disease vs. healthy or treatment vs. control. To this end, we perform **Differential Activation Analysis (DAA)**,described in Section 6.1.6, to identify the differentially activated genesets. A current standard alternative method that provides insight into the underlying biology is to use Gene Set Enrichment Analysis (**GSEA**), which is described in Section 6.1.5

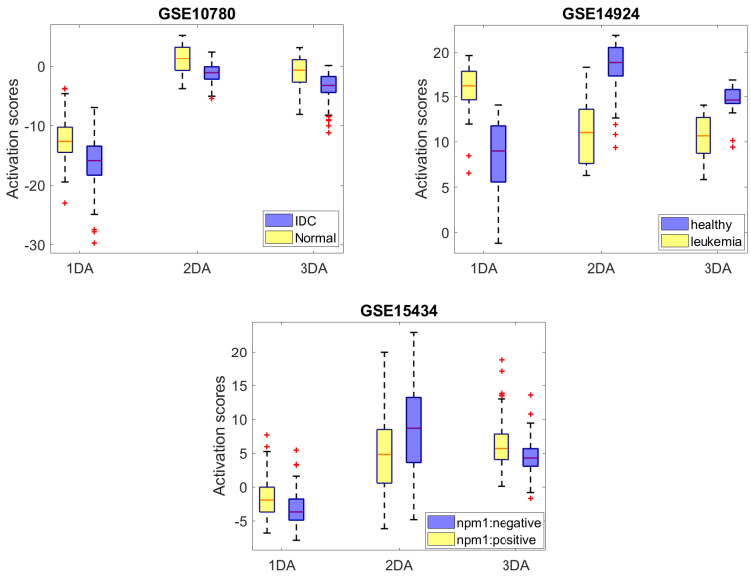

Figure 6.5: Box-plots of the activation scores that correspond to the first, second, third differentially activated PASL feature/pathway. It is verified that the differentially activated pathways behave differently between the phenotypes. The outcome of GSE10780 stands for Invasive Ductal Carcinoma/Unremarkable breast ducts, and the outcome of GSE15434 stands for the mutation status of Nucleophosmin 1 (NPM1).

Specifically, we examine the ability of PASL to identify genesets (pathways) that behave differently between two classes and compare it against GSEA. We employ the GSEA v4.0.3 tool from `https://www.gsea-msigdb.org/gsea/index.jsp` [34, 46]. We run GSEA on the test datasets in the original feature space using 10000 phenotype permutations for the permutation-based statistical test employed in the package. The input genesets are the same as the ones provided to PASL in the geneset matrix G. We also perform DAA on the test datasets projected to the latent space of PASL (activity scores) using the Matlab's t-test function *mattest* with 10000 permutations. The list of p-values from DAA and GSEA can then be used to identify the affected pathways.

Fig. 6.4 shows the number of pathways identified by each method (y-axis) in the top k (lowest p-value) pathways, for each k (x-axis). Each panel corresponds to a different test dataset. We observe that the pathways identified by PASL have lower p-values and are encountered first on the list; PASL has higher statistical power in identifying some genesets that behave differently. PASL's features correspond to pathways. The statistically significant ones are referred as *differentially activated*.

Fig. 6.5 visualizes why the PASL features are identified as *differentially activated*. Each panel shows the box-plots for the activation scores corresponding to the first, second, and third most statistically significant PASL feature/pathway

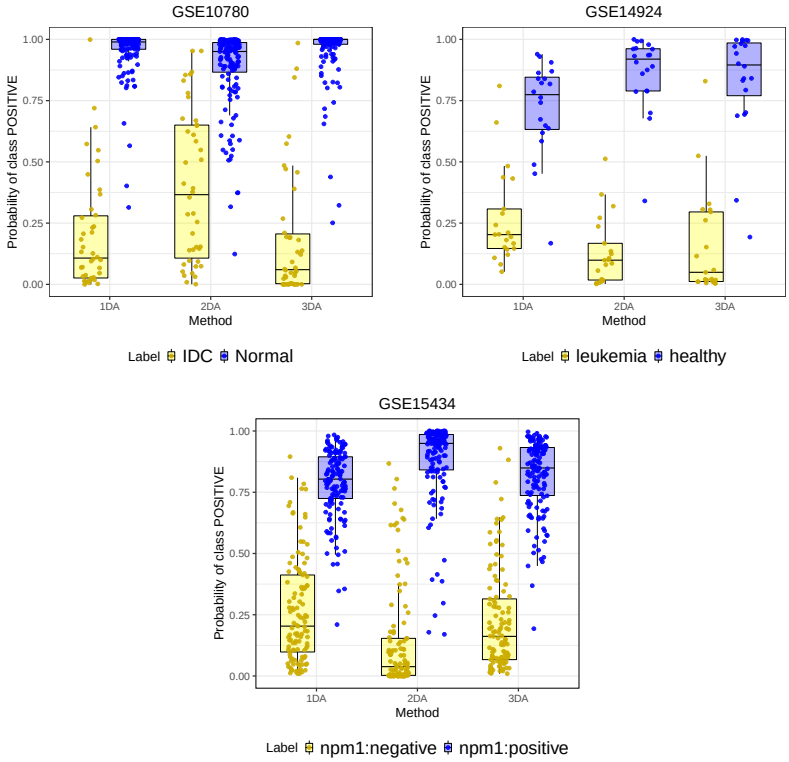

Figure 6.6: Out of sample probability of test datasets reduced to their top 3 differentially activated but not enriched pathways. The outcome of GSE10780 stands for Invasive Ductal Carcinoma/Unremarkable breast ducts, and the outcome of GSE15434 stands for the mutation status of Nucleophosmin 1 (NPM1).

(denoted with names 1DA, 2DA, and 3DA, respectively).

Specifically, the top 3 differentially activated pathways of GSE10780 are the "Reactome signaling by GPCR", "Reactome Fructoce Catabolism" and "Reactome Hemostasis". The top 3 differentially activated pathways of GSE14924 is the "Reactome metabolism of Lipids", "Reactome Chromatin Organization" and "Reactome Gene Expression Transcription". The top 3 differentially activated pathways of GSE15434 are the "Reactome Transport of Small Molecules", "Reactome Developmental Biology", "Reactome Post Translational Protein Modification". *It is visually verified that the scores are different between the phenotypes in an easy to understand and intuitive plot.*

We furtherly want to verify that differential activated pathways are indeed significant for the phenotype separation of the initial test data. Figure 6.6 shows the out-of-sample probability of the test datasets reduced to the genes of the top three differentially activated pathways identified by DAA. Indeed, these pathways appear to be significant for the data separation.

While DAA using PASL seems to offer several advantages (lower p-values,

intuitive visualization), it also has a major limitation. PASL requires a training set that is related to the application (test) set. It learns atoms that only pertain to capturing information regarding the train data. For example, DAA using PASL cannot be applied to a schizophrenia dataset, before we construct a sufficiently large training dataset for the disease. As such, we consider DAA and GSEA complementary and synergistic. Also, one other limitation is that DAA takes as input the PASL scores which correspond to the genesets that belong to the PASL's dictionary. On the contrary, GSEA takes as input all the genesets that belong to the geneset matrix G.

Chapter 7

Conclusions

7.1 Conclusion

Molecular omics and multi-omics data are notoriously high-dimensional. Statistical or machine learning analysis of such data could hit computational obstacles due to the high dimensionality; results may be hard to interpret (e.g., interpreting thousands of differentially expressed genes or pair-wise correlations and covariances). In order to overcome the statistical challenges, several dimensionality reduction methods for such data have been proposed.

Dimensionality reduction projects the high-dimensional data into a lower dimensional space, which keeps the valuable information, while it reduces space and computational requirements for further analysis. Traditional dimensionality reduction methods, such as the Principal Component Analysis (PCA) and similar methods usually end up with an unintepretable new feature space. To overcome this issue there have been proposed sparse dimensionality reduction techniques that could be easier to interpret biologically i.e., the new constructed features is a linear combination of only a small number of the original features (probe sets).

In this work we propose a novel dimensionality reduction method, called Pathway Activity Score Learning (PASL). To the extent of our knowledge, PASL is the first technique where the newly constructed features directly correspond to prior knowledge about genesets. PASL is relatively computationally efficient by relying on a greedy, yet effective heuristic to construct the next atom.

PASL projects to a new feature space that maintains the predictive information for a wide range of outcomes, e.g., disease status, diet, mutation status, and others. To test predictive performance we employed held-out datasets never seen by the PASL. For predictive modeling, we employed an automated machine learning architecture (AutoML) that performs an automated algorithm and hyper-parameter value tuning and returns conservative estimates of performance, called JADBIO. The classification models created on PASL's space outperform the ones created on the PLIER's space and are on par with the ones using the original features. Classification analysis is one order of magnitude faster in PASL space than in the original

space. PASL's learned features can also be used for Differential Activation Analysis identifying the pathways that behave differently between two classes. This analysis is synergistic to gene set enrichment analysis, it is intuitively visualized, and often produces smaller p-values.

7.2 Future Work

Based on these promising results, as a future work PASL will be applied on a much larger corpus of gene expression data, spanning a wide plethora of diseases and conditions. The aim of this experiment is to create a biologically interpretable low dimensional latent space able to capture the biological information for multiple outcomes. Essentially, every dataset, regardless of disease or outcome, could be projected into this latent space, keeping the valuable information, and also gaining a high compression ratio. This speeds up further analysis (e.g. disease classification).

Also, despite the fact that PASL uses a heuristic which is computational efficient, there are ideas to make it faster in order to handle larger number of samples and genesets.

Chapter 8

Additional Information

In Tables 8.1, 8.2 we present the phenotypes of the test datasets that were used for the evaluation of PASL.

Further information for Table 8.1:

- ER +/-: estrogen receptors positive/negative
- IER: Intermittent energy restriction
- triple-negative breast cancer subtypes:
  - BLIA: Basal-Like Immune-Activated
  - BLIS: Basal-Like Immune-Suppressed
  - MES: Mesenchymal
  - LAR: Luminal-AR
- DER: Dietary energy restriction
- Applied therapies:
  - chem: chemotherapy
  - lap: lapatinib
  - tras: trastuzumab

Further information for Table 8.2:

- npm1 positive/negative: mutation of nucleophosmingene (npm1)
- t-MDS/AML: Treatment-related myelodysplastic syndrome in Acute Myeloid Leukemia
- RER: rapid early responder
- SER: slow early responder
- CCR: complete continuous remission

ID	#samples	class1	class2	class3	class4	class5
54002	433	Tumor	Non Tumor			
5460	129	ER+	ER-			
36771	107	ER+	ER-			
66161	74	Before IER	After IER			
76124	198	MES	BLIA	BLIS	LAR	
66159	76	DER	not DER			
66305	88	chem+tras+lap	chem+tras	chem+lap		
10780	185	Healthy	IDC			
27562	162	Healthy	Malignant	Benign	Post-surgery	Ectopic
27830	155	BRCA1	BRCA2	CHEK2	No mutation	
36769	60	Healthy	Breast Cancer			
29431	66	Healthy	Breast Cancer			
42568	121	Healthy	Breast Cancer			

Table 8.1: Phenotype information for Breast cancer test datasets.

ID	#samples	class1	class2	class3
15434	251	npm1: positive	npm1: negative	
14924	41	acute myeloid leukemia	healthy	
23025	124	t-MDS/AML	not t-MDS/AML	
21029	62	Peripheral_blood	Bone_marrow	Lymph_node
28654	112	igvh_mutated	igvh_unmutated	
14671	59	Responder	Non-Responder	
7440	99	RER	SER	CCR
66006	98	B-other	Ph-positive	MLL-positive
28460	98	relapse	diagnosis	
26713	124	Acute Lymphoblastic Leukemia	healthy	
31048	221	chronic lymphocytic leukemia	healthy	
39411	152	chronic lymphocytic leukemia	healthy	
49695	64	ighv_mutated	ighv_unmutated	
50006	220	chronic lymphocytic leukemia	healthy	
61804	325	FLT3-ITD positive	FLT3-ITD negative	

Table 8.2: Phenotype information for Leukemia test datasets.

Acknowledgements

This research has been co-financed by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH-CREATE–INNOVATE (project code:T1EDK-00905) and the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement n. 617393.

Bibliography

[1] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[2] Marios Adamou, Grigoris Antoniou, Elissavet Greasidou, Vincenzo Lagani, Paulos Charonyktakis, Ioannis Tsamardinos, and Michael Doyle. Toward automatic risk assessment to support suicide prevention. *Crisis: The Journal of Crisis Intervention and Suicide Prevention*, 2018.

[3] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, 54(11):4311–4322, 2006.

[4] Anestis Antoniadis, Sophie Lambert-Lacroix, and Frédérique Leblanc. Effective dimension reduction methods for tumor classification using gene expression data. *Bioinformatics*, 19(5):563–570, 2003.

[5] Rabia Aziz, CK Verma, and Namita Srivastava. Dimension reduction methods for microarray data: a review. *AIMS Bioengineering*, 4(2):179–197, 2017.

[6] Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 245–250, 2001.

[7] Adrian P Bird. Gene number, noise reduction and biological complexity. *Trends in Genetics*, 11(3):94–100, 1995.

[8] George EP Box and David R Cox. An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological)*, 26(2):211–243, 1964.

[9] Jean-Philippe Brunet, Pablo Tamayo, Todd R Golub, and Jill P Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the national academy of sciences*, 101(12):4164–4169, 2004.

[10] Pedro Carmona-Saez, Roberto D Pascual-Marqui, Francisco Tirado, Jose M Carazo, and Alberto Pascual-Montano. Biclustering of gene expression data by non-smooth non-negative matrix factorization. *BMC bioinformatics*, 7(1):78, 2006.

[11] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.

[12] Jung Kyoon Choi, Jong Young Choi, Dae Ghon Kim, Dong Wook Choi, Bu Yeo Kim, Kee Ho Lee, Young Il Yeom, Hyang Sook Yoo, Ook Joon Yoo, and Sangsoo Kim. Integrative analysis of multiple gene expression profiles applied to liver cancer study. *FEBS letters*, 565(1-3):93–100, 2004.

[13] Emily Clough and Tanya Barrett. The gene expression omnibus database. In *Statistical genomics*, pages 93–110. Springer, 2016.

[14] Leonardo Collado-Torres, Abhinav Nellore, Kai Kammers, Shannon E Ellis, Margaret A Taub, Kasper D Hansen, Andrew E Jaffe, Ben Langmead, and Jeffrey T Leek. recount: A large-scale resource of analysis-ready rna-seq expression data. *BioRxiv*, page 068478, 2016.

[15] Francis Crick. Central dogma of molecular biology. *Nature*, 227(5258):561–563, 1970.

[16] David Croft, Antonio Fabregat Mundo, Robin Haw, Marija Milacic, Joel Weiser, Guanming Wu, Michael Caudy, Phani Garapati, Marc Gillespie, Maulik R Kamdar, et al. The reactome pathway knowledgebase. *Nucleic acids research*, 42(D1):D472–D477, 2014.

[17] John P Cunningham and M Yu Byron. Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509, 2014.

[18] Jovani Taveira De Souza, Antonio Carlos De Francisco, and Dayana Carla De Macedo. Dimensionality reduction in gene expression data sets. *IEEE Access*, 7:61136–61144, 2019.

[19] Sandrine Dudoit, Juliet Popper Shaffer, and Jennifer C Boldrick. Multiple hypothesis testing in microarray experiments. *Statistical Science*, pages 71–103, 2003.

[20] Kjersti Engan, Bhaskar D Rao, and Kenneth Kreutz-Delgado. Frame design using focuss with method of optimal directions (mod). In *Proc. NORSIG*, volume 99, pages 65–69, 1999.

[21] Elana J Fertig, Jie Ding, Alexander V Favorov, Giovanni Parmigiani, and Michael F Ochs. Cogaps: an r/c++ package to identify patterns and biological process activity in transcriptomic data. *Bioinformatics*, 26(21):2792–2793, 2010.

[22] Jelle J Goeman and Peter Bühlmann. Analyzing gene expression data in terms of gene sets: methodological issues. *Bioinformatics*, 23(8):980–987, 2007.

[23] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[24] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430, 2000.

[25] Zhen Jiang and Robert Gentleman. Extensions to gene set enrichment. *Bioinformatics*, 23(3):306–313, 2007.

[26] Zhuolin Jiang, Zhe Lin, and Larry S Davis. Label consistent k-svd: Learning a discriminative dictionary for recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2651–2664, 2013.

[27] Minoru Kanehisa and Susumu Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30, 2000.

[28] Vincenzo Lagani, Giorgos Athineou, Alessio Farcomeni, Michail Tsagris, and Ioannis Tsamardinos. Feature selection with the r package mxm: Discovering statistically-equivalent feature subsets. *arXiv preprint arXiv:1611.03227*, 2016.

[29] Kleanthi Lakiotaki, Nikolaos Vorniotakis, Michail Tsagris, Georgios Georgakopoulos, and Ioannis Tsamardinos. Biodataome: a collection of uniformly preprocessed and automatically annotated datasets for data-driven biology. *Database*, 2018, 2018.

[30] Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[31] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[32] Tamilselvi Madeswaran and GM Kadhar Nawaz. A comparative analysis of classification of micro array gene expression data using dimensionality reduction techniques. *International Journal of Computer and Electronics Research*, 1(4), 2012.

[33] Weiguang Mao, Elena Zaslavsky, Boris M Hartmann, Stuart C Sealfon, and Maria Chikina. Pathway-level information extractor (plier) for gene expression data. *Nature methods*, 16(7):607–610, 2019.

[34] Vamsi K Mootha, Cecilia M Lindgren, Karl-Fredrik Eriksson, Aravind Subramanian, Smita Sihag, Joseph Lehar, Pere Puigserver, Emma Carlsson, Martin Ridderstråle, Esa Laurila, et al. Pgc-1α-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature genetics*, 34(3):267–273, 2003.

[35] Darryl Nishimura. Biocarta. *Biotech Software & Internet Report: The Computer Software Journal for Scient*, 2(3):117–120, 2001.

[36] Martin A Nowak, Maarten C Boerlijst, Jonathan Cooke, and John Maynard Smith. Evolution of genetic redundancy. *Nature*, 388(6638):167–171, 1997.

[37] Georgia Orfanoudaki, Maria Markaki, Katerina Chatzi, Ioannis Tsamardinos, and Anastassios Economou. Maturep: prediction of secreted proteins with exclusive information from their mature regions. *Scientific reports*, 7(1):1–12, 2017.

[38] Daniel Pinkel, Richard Segraves, Damir Sudar, Steven Clark, Ian Poole, David Kowbel, Colin Collins, Wen-Lin Kuo, Chira Chen, Ye Zhai, et al. High resolution analysis of dna copy number variation using comparative genomic hybridization to microarrays. *Nature genetics*, 20(2):207–211, 1998.

[39] Lubo Popelínskỳ. Combining the principal components method with different learning algorithms. In *In: Proc. of ECML/PKDD IDDM Workshop (Integrating Aspects of Data Mining, Decision Support and Meta-Learning.(2001.* Citeseer, 2000.

[40] Adaikalavan Ramasamy, Adrian Mondry, Chris C Holmes, and Douglas G Altman. Key issues in conducting a meta-analysis of gene expression microarray datasets. *PLoS Med*, 5(9):e184, 2008.

[41] Adam Roberts, Jesse H Engel, Sageev Oore, and Douglas Eck. Learning latent representations of music to generate interactive musical palettes. In *IUI Workshops*, 2018.

[42] S Sasikala and S Appavu Alias Balamurugan. Data classification using pca based on effective variance coverage (evc). In *2013 IEEE International Conference ON Emerging Trends in Computing, Communication and Nanotechnology (ICECCN)*, pages 727–732. IEEE, 2013.

[43] Mark Schena, Dari Shalon, Ronald W Davis, and Patrick O Brown. Quantitative monitoring of gene expression patterns with a complementary dna microarray. *Science*, 270(5235):467–470, 1995.

[44] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998.

[45] Natasha Singh-Miller, Michael Collins, and Timothy J Hazen. Dimensionality reduction for speech recognition using neighborhood components analysis. In *Eighth Annual Conference of the International Speech Communication Association*, 2007.

[46] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

[47] Jonatan Taminau, Cosmin Lazar, Stijn Meganck, and Ann Nowé. Comparison of merging and meta-analysis as alternative approaches for integrative gene expression analysis. *International Scholarly Research Notices*, 2014, 2014.

[48] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

[49] Ioannis Tsamardinos, Pavlos Charonyktakis, Kleanthi Lakiotaki, Giorgos Borboudakis, Jean Claude Zenklusen, Hartmut Juhl, Ekaterini Chatzaki, and Vincenzo Lagani. Just add data: Automated predictive modeling and biosignature discovery. *bioRxiv*, 2020.

[50] Ioannis Tsamardinos, George S Fanourgakis, Elissavet Greasidou, Emmanuel Klontzas, Konstantinos Gkagkas, and George E Froudakis. An automated machine learning architecture for the accelerated prediction of metal-organic frameworks performance in energy and environmental applications. *Microporous and Mesoporous Materials*, page 110160, 2020.

[51] Ioannis Tsamardinos, Elissavet Greasidou, and Giorgos Borboudakis. Bootstrapping the out-of-sample predictions for efficient and accurate cross-validation. *Machine learning*, 107(12):1895–1922, 2018.

[52] Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10(66-71):13, 2009.

[53] Wiesława Widłak. High-throughput technologies in molecular biology. In *Molecular Biology*, pages 139–153. Springer, 2013.

[54] Meng Yang, Lei Zhang, Xiangchu Feng, and David Zhang. Fisher discrimination dictionary learning for sparse representation. In *2011 International Conference on Computer Vision*, pages 543–550. IEEE, 2011.

[55] Qiang Zhang and Baoxin Li. Discriminative k-svd for dictionary learning in face recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2691–2698. IEEE, 2010.

[56] Li Zhuo, Bo Cheng, and Jing Zhang. A comparative study of dimensionality reduction methods for large-scale image retrieval. *Neurocomputing*, 141:202–210, 2014.

[57] Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006.