Using Biological Networks and Gene-Expression Profiles for the Analysis of Diseases

LIM JUNLIANG KEVIN

NATIONAL UNIVERSITY OF SINGAPORE

2015

Using Biological Networks and Gene-Expression Profiles for the Analysis of Diseases

LIM JUNLIANG KEVIN
(B.Comp. (Hons.), NUS)

A DOCTORAL THESIS SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE
NATIONAL UNIVERSITY OF SINGAPORE

2015

Declaration of Authorship

I, LIM JUNLIANG KEVIN, hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Signed:

Date:

20/11/2014

"Each problem that I solved become a rule, which served afterwards to solve other problems."

René Descartes

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Prof. Wong Limsoon, whose expertise, knowledge and patience contributed greatly to my graduate experience. His vast knowledge in analyzing gene-expression profiles as well as having an apt ability to explain and interpret data have expedited and resulted in many ideas in this thesis.

I thank Prof. Choi Kwok Pui, for his insights and knowledge in statistics. I thank Prof. Ken Sung and Prof. Thiagu, for reading and listening to my reports and presentations, as well as their advices.

I would also like to thank fellow colleagues: Goh Wilson, Yong Chern Han, Koh Chuan Hock, Li Zhenhua, Jin Jingjing, Lim Jing Quan, Fan Mengyuan, Michal Wozniak, Wang Yue and Zhou Hufeng, for discussing their ideas and making my stay in the lab a memorable one.

Finally, I thank my family members: my father, who has provided much to educate and groom me in many ways more than just academics. My late mother, who has provided me with the warmth of a home, even in times of illness. My brothers, Wilfred, Xavier and Clarence, who have encouraged me in one way or another. My wife, Christine, for her patience and support. My son, Luke, for bringing a smile in difficult times.

Abstract

The wealth of microarray data available today allows us to perform two important tasks: (1) Inferring biological explanations or causes behind diseases. (2) Using these explanations to diagnose and predict the outcome of future patients. These tasks are challenging and results are often not reproducible when different batches of data are analyzed. This problem is further aggravated by the lack of samples because many laboratories are constrained by budget, biology or other factors; making it hard to draw reasonable and consistent biological conclusions.

By using databases of biological pathways, which represent a wealth of biological information about the interdependencies between genes in performing a specific function, we are able to formulate algorithms that draw meaningful and consistent biological explanations as plausible causes of diseases. We derive and find statistically significant "subnetworks", which are smaller connected components within biological pathways, because the cause of a disease may be linked to a small subset of genes within a pathway. This, in conjunction with a unique scoring methodology, we are able to compute a test statistic that is stable even when sample sizes are small, and is consistently detected over independent batches of data, even from different microarray platforms. We are able to attain a high subnetwork-level agreement of about 58% using only 2 samples. For other contemporary methods, this number falls to 27% when analyzed using GSEA and 13% using ORA. In addition, the subnetwork-level agreement achieved by our method continues to improve when a larger sample size is used, yielding a subnetwork agreement of about 93%. Our predicted subnetworks are also supported by many existing biological literature and allow biologists further insights to the mechanisms behind the diseases studied.

This work is important because the subnetworks identified, being consistent across independent datasets, also serve as informative and relevant features. Thus, we are able to build better predictive algorithms for inferring the outcome of patients. We also present a useful subnetwork-feature scoring function that is not only able to predict the outcome of future samples measured on independent microarray platforms but is also able to handle small-size training samples. This enables researchers to find the mechanisms behind a disease and use them directly as a tool for diagnosis and prognosis.

Contents

List of Figures

List of Tables

Abbreviations

ALL	**A**cute **L**ymphoblastic **L**eukemia
AML	**A**cute **M**yeloid **L**eukemia
DEAP	**D**ifferential **E**xpression **A**nalysis of **P**athways
DEGs	**D**ifferentially **E**xpressed **G**enes
DMD	**D**uchenne **M**uscular **D**ystrophy
ESSnet	**E**xtremely **S**mall sample size **S**ubnetworks
FCS	**F**unctional **C**lass **S**coring
GGEA	**G**ene **G**raph **E**nrichment **A**nalysis
GSEA	**G**ene **S**et **E**nrichment **A**nalysis
NEA	**N**etwork **E**nrichment **A**nalysis
ODE	**O**rdinary **D**ifferential **E**quation
ORA	**O**verlap **R**epresentation **A**nalysis
PCA	**P**rinciple **C**omponent **A**nalysis
PFSnet	**P**aired **F**uzzy **S**ubnetworks
SAM	**S**ignificance **A**nalysis of **M**icroarrays
SRI	**S**ystem **R**esponse **I**nference
SVM	**S**upport **V**ector **M**achine

Dedicated to my late beloved mother...

Chapter 1

Introduction

The wealth of information contained in gene-expression databases is growing rapidly. To date, there are more than 60,000 experimental datasets stored in different gene-expression repositories; cf. fig. 1.1.

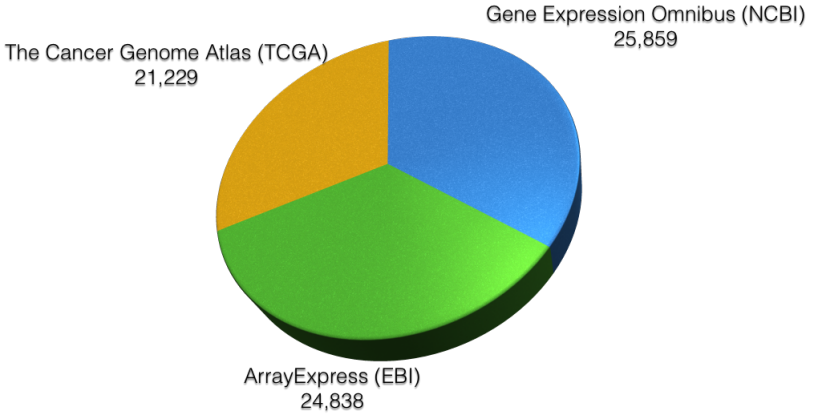


FIGURE 1.1: Number of gene-expression profile datasets in database repositories.

This quantitative measure of gene transcripts at once allows researchers to gain insight to complex diseases. The analysis can be divided into two sub-problems, which this dissertation aims to address. The first problem is concerned with identifying the difference present between patients and normal individuals. The second problem is concerned with distinguishing patients from normal given what has been identified in the first step.

1.1 Motivation

1.1.1 Identifying disease-related genes

Traditional microarray analysis is focused on determining differentially-expressed genes either between normal cells and diseased cells or between two disease subtypes. This kind of inference typically computes a measure of statistical significance for differentially expressed genes, but has been shown to have a number of problems.

1. Large numbers of false positives due to multiple hypothesis testing. If there are 30,000 genes in a microarray and assuming that the false-positive rate is about 5%, then we expect to see 1,500 genes falsely declared as differentially expressed. This large number of false positives obscures the understanding of complex diseases and makes analysis difficult.

2. Although these false positives can be alleviated by multiple hypothesis correction, genes detected as significant are sparsely scattered in biological networks, suggesting that these genes do not provide biological insights to the cause of disease. In contrast, diseases are usually triggered by a cascade of interacting genes whose expression levels are expected to change.

3. It has been widely reported that genes detected as significant in one microarray experiment are not consistently detected in another microarray experiment of the same disease phenotype (Zhang et al., 2009). And in some cases, they are no better than randomly produced gene signatures (Venet et al., 2011). For example, the Cathepsin D gene is significantly differentially expressed in one Leukemia dataset but not in another independent dataset; cf. fig. 1.2

4. In addition, the significant genes are very sensitive to sample-size changes especially when smaller sample sizes are considered. This restricts analysis to sizeable datasets, but laboratories are sometimes constrained to perform experiments with few samples.

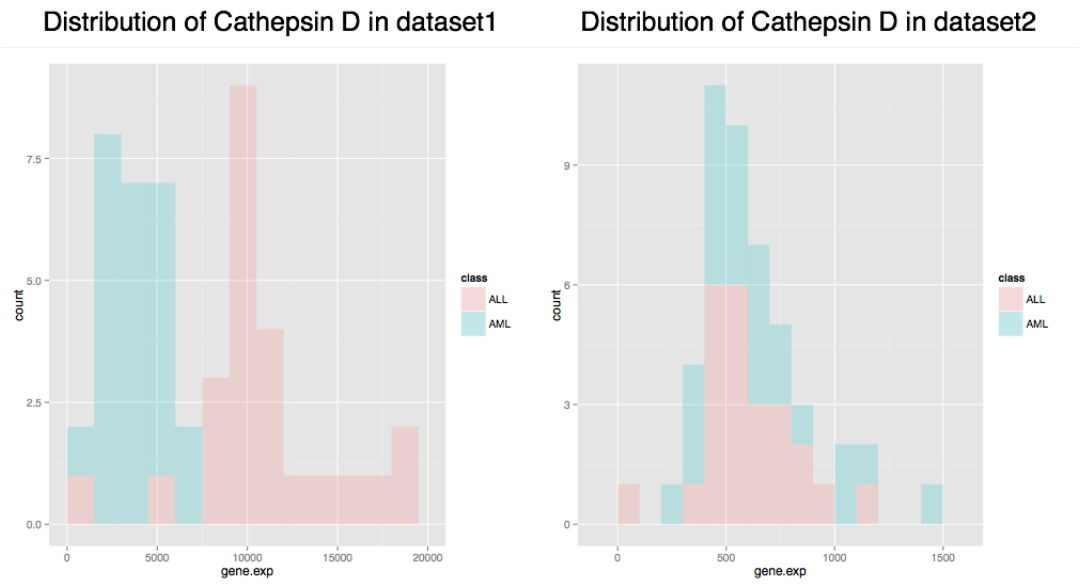


FIGURE 1.2: The distribution of the Cathepsin D gene, identified as significant by t-test in Leukemia dataset 1 but not in Leukemia dataset 2.

Modern methods try to tackle some of these problems by incorporating biological information into their framework in the form of gene sets. These gene sets represent biological processes or pathways that are known to perform specific functions. However, these methods do not solve all of the above-mentioned problems. It has been shown that these modern methods do not produce consistent results when they are applied on cross-laboratory and cross-platform data. This has a large impact on scientific studies because the significant genes often cannot be reproduced; suggesting that most genes or pathways linked by these methods to the disease may not be real. In addition, these methods try to assess an entire pathway, which may cause an actually relevant pathway to be missed because in disease state, only a part of the pathway is perturbed.

Analyzing whole pathways by themselves offers biologists little insight to a disease because it is very unlikely for a disease to affect whole large pathways. Rather, it is more plausible for a disease to target a small area within a pathway. This motivates us to work on methods that specifically consider smaller components within pathways.

1.1.2 A tool for clinical diagnosis

In another application of microarray data, differentially-expressed genes are used to predict patients from normal. Typically, a machine-learning method is employed at this step to find the labels of an unlabeled sample. This prediction task faces different kinds of problems:

1. Batch effect. In order for such methods to be practical, a classifier built using a dataset from the current time point should also give accurate results when applied to datasets obtained in the future. However, the features in a set of samples often cause the samples to be segregated into clusters based on data batches rather than based on class labels. This makes it very hard for machine-learning algorithms to make predictions. Cf. fig. 1.3

2. Although batch effect can be reduced by rank-based normalization, even in the absence of batch effect, using genes as features do not separate the classes well, as these classifiers tend to have poor predictive accuracy when they are applied to future batches of samples. Cf. fig. 1.4

There has been no method in our knowledge that can make this prediction reliably utilizing gene-expression values when data from different platforms or laboratories are used. This suggests that the traditional perspective of using individual genes for classification may be inadequate.

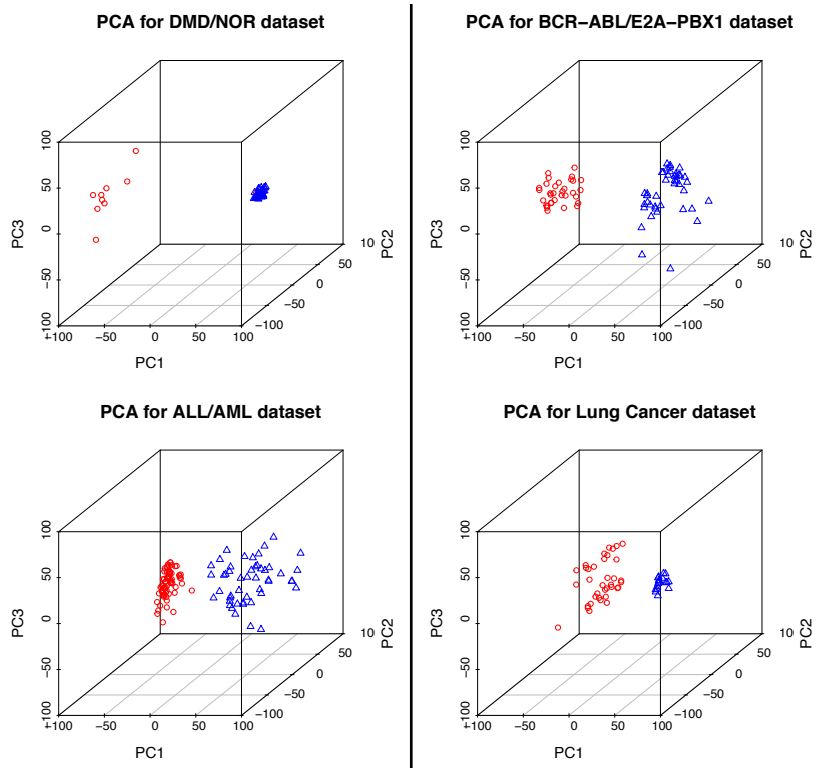


FIGURE 1.3: We perform PCA on the microarray data in 2 independent datasets. Samples are then plotted on the first three principle components. The samples are separated based on batches rather than by their labels.

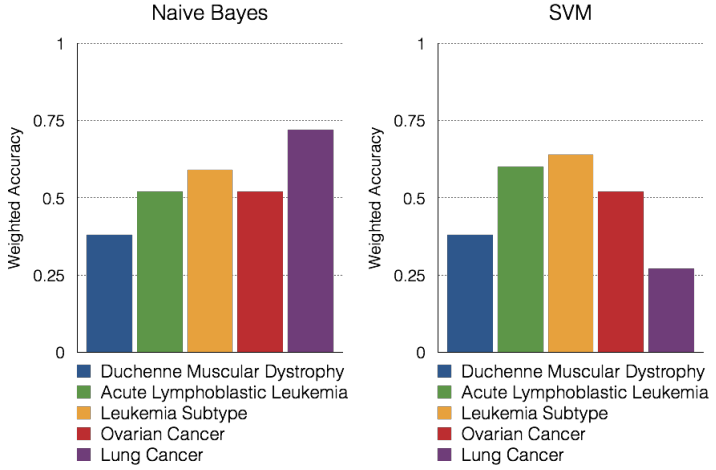


FIGURE 1.4: We use t-test to select significant genes to build classifiers from one dataset and supply an independent dataset for testing. The weighted accuracy, defined as the average of the sensitivity and specifity, indicates that classifiers built on individual gene features do not perform well.

1.2 Research challenge and contributions

The above-mentioned problems present a few difficulties that need to be addressed. We identify 3 research challenges that we aim to address in this dissertation:

1. Over all the recent methods that we surveyed, very few methods are able to reproduce subnetworks or genes in high agreement when applied independently on independent datasets. One such exception is SNet (Soh et al., 2011), but we discover that the performance of SNet varies when hard thresholds are used. On different disease types, the optimal threshold may be different. This motivates us to find a way to improve SNet to achieve high consistency without relying on tuned thresholds. We introduce, in PFSNet, two major modifications to the SNet algorithm and obtain even higher consistency than SNet. We have published the resulting work in a recent paper, viz:
   K. Lim, L. Wong. "Finding consistent disease subnetworks using PFSNet", Bioinformatics, 30(2):189-196, January 2014.

2. To date, we have not seen any published method that provides a handle on the situation where sample size is extremely small. On the other hand, we often see data from laboratories that are constrained to conduct biological experiments with extremely few samples (<5). We discover that most statistics computed under this circumstance produce a large variance, and hence low consistency, when tested across diverse datasets. One possible explanation is that the statistics are computed from very few data points. We introduce a novel method, ESSNet, that involves two major steps. The first is defining subnetworks based on the genes' average rank, which is shown to be very stable—i.e., lesser variance—across small sample subsets of the original data. The second is using biological pathways to

increase the number of data points so that the statistics can be computed more reliably despite the small sample size. We demonstrate that subnetworks are consistently identified across multiple datasets and correlate to biological processes linked to the disease. The resulting work has been submitted and is under review: K. Lim, Z. Li, K. P. Choi, L. Wong. "ESSNet: Finding consistent disease subnetworks in datasets with extremely small sample sizes".

3. The batch effect presented earlier suggests that gene-expression values do not make good feature scores for diagnosis of diseases. This motivates us to explore other forms of features derived from pathways or subnetworks. We demonstrate how subnetworks can be used as features by using a method to score samples based on FSNet and ESSNet to achieve high cross-batch prediction accuracy.

1.3 Thesis organization

Chapter 2 provides technical background on microarray analysis methods, which try to identify the cause of a disease using biological networks and pathways, as well as classification techniques. Chapter 3 describes our contribution (1) on improving SNet to achieve a higher level of consistency. Chapter 4 describes our contribution (2) on dealing with datasets with extremely-small sample sizes. In chapter 5, we discuss how subnetworks described in (1) and (2) can be used for diagnosis purpose. Finally, in chapter 6, we summarize our work and propose some future work.

Chapter 2

Related Work and Definitions

2.1 Background on gene-expression profiling

Gene-expression profiling is the simultaneous measurement of the amount of mRNAs of all the genes, which are transcribed from the genome, in the cell. At any moment, not all the genes are activated, resulting in the phenotypic difference between patients and normal. Currently, there are two major platforms for profiling gene expression: (1) traditional microarrays and (2) next-generation sequencing RNA-seq experiments. Microarrays are more pre-dominant, accounting for 87% of datasets in the ArrayExpress database (Rustici et al., 2013).

In the popular brand of microarrays made by Affymetrix (Fodor et al., 1991), complementary sequences to the targeted mRNAs are printed onto a gene chip. These complementary sequences target different parts of the mRNAs and are associated in pairs. Each pair is made of a perfect-match (PM) and a mismatch (MM) sequence; the mismatch sequence allows background noise level to be measured, and when combined

with the signal intensity from the perfect-match sequences, determines the expression value of a particular gene (see fig. 2.1).

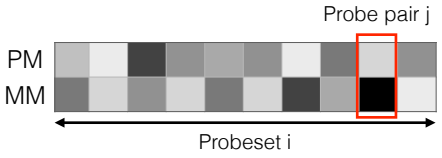


FIGURE 2.1: A figure depicting probesets and probepairs in a microarray.

RNA-seq (Chu and Corey, 2012, Wang et al., 2009) is a method which fragments the extracted mRNAs into pieces, and attach short tags to them. The tags are hybridized to beads and next-generation sequencing is employed to deduce the transcript that they belong to. A quantitative measure of a gene is derived based on the number of fragments that map to the gene's sequence. In practice, RNA-seq offers some advantages over microarrays: (1) a reference genome is not necessary prior to the experiment, (2) larger dynamic range, and (3) higher technical reproducibility.

In this thesis, we are concerned with analysis of microarray data due to its wider availability, although the methods in our dissertation can also be applied to RNA-seq data.

2.1.1 Preprocessing microarray data

Microarray data is often preprocessed before any downstream analysis is conducted. The preprocessing serves the following functions:

(1) Estimate background noise. (2) Adjusting expression values to correct for non-specific hybridization using the PM and MM sequences described earlier. (3) Normalizing values so that values can be compared across chips. (4) Summarizing multiple probe expressions into a single gene-expression value.

The two most-widely-used microarray-preprocessing tools are MAS5.0 (Affymetrix, 2002) and RMA (Irizarry et al., 2003).

2.1.1.1 MAS5.0

MAS5.0 is a proprietary software used on Affymetrix chips and is described by a white paper published by Affymetrix (Affymetrix, 2002). The j^{th} probe pair associated with the i^{th} probeset can be represented as $PM_{i,j}$ and $MM_{i,j}$ (see fig. 2.1). It uses the $MM_{i,j}$ probes to estimate the background noise and the probe signal is basically the $PM_{i,j}$ intensities subtracted by the $MM_{i,j}$ intensities. It is possible for the signal intensities from MM probes to be larger than the signal intensities for the PM probes, making it hard to estimate stray signals from the PM intensities. Hence, the MM intensities have to be adjusted. The adjusted intensity $IM_{i,j}$ for probe pair j in the i^{th} probeset is defined as:

$$IM_{i,j} = \begin{cases} MM_{i,j} & MM_{i,j} < PM_{i,j} \\ \frac{PM_{i,j}}{2^{(SB_i)}} & MM_{i,j} \geq PM_{i,j} \text{ and } SB_i > \tau_1 \\ \frac{PM_{i,j}}{2^{1+\frac{\tau_1 - SB_i}{\tau_2}}} & MM_{i,j} \geq PM_{i,j} \text{ and } SB_i \leq \tau_1 \end{cases} \tag{2.1}$$

where SB_i is a weighted average of the log ratios of probepairs j for the probeset i:

$$SB_i = weighted\_average_j \left(log_2 \left(\frac{PM_{i,j}}{MM_{i,j}} \right) \right) \tag{2.2}$$

The first case in equation 2.1 is when $MM_{i,j}$ is smaller than $PM_{i,j}$, the background signal is perfectly measured so no adjustments is made.

For the second case where $MM_{i,j}$ is greater than $PM_{i,j}$, the adjusted intensity $IM_{i,j}$ is based on the weighted average of other probe pairs within the same probeset if this weighted average is big enough, i.e. $> \tau_1$.

For the third case, if the weighted average of the other probe pairs within the same probeset is also very small, then the adjusted intensity is set to a value slightly smaller than $PM_{i,j}$, based on a scale parameter τ_2.

Tukey's biweight algorithm is used for the weighted-average computation above, which basically assigns bigger weights for values close to the median and smaller weights for values far from the median, so that the average is robust to outliers.

2.1.1.2 RMA

RMA (Irizarry et al., 2003) is another tool for microarray preprocessing. It does not rely on MM intensities to estimate background noise. Rather, it is a model-based approach that assumes background noise follows a normal distribution with mean μ and standard deviation σ and real signal follows an exponential distribution with parameter alpha α. This formulation results in a closed form solution for the expected real signal given the PM intensities once the model parameters have been estimated:

$$E[Signal|PM = x] = a + b\frac{\phi(\frac{a}{b}) - \phi(\frac{x-a}{b})}{\Phi(\frac{a}{b}) + \Phi(\frac{x-a}{b}) - 1} \tag{2.3}$$

where $a = x - \mu - \sigma^2\alpha$, $b = \sigma$, $\phi(.)$ is the density function of the normal distribution and Φ is the cumulative distribution function of the normal distribution.

2.2 Background on class comparison using genes, pathways and subnetworks

Many downstream microarray-analysis methods start after data pre-processing. In this subsection, we discuss various approaches that have been proposed for comparing the differences between patient and normal by identifying significant genes, pathways and subnetworks.

2.2.1 Identifying differential gene expression

The earliest work on class comparison (DeRisi et al., 1996, Furey et al., 2000, Golub et al., 1999a) on microarray analysis uses simple computation like fold-change, t-test and Wilcoxon rank-sum test to evaluate differential gene expression. Other methods have been later developed to help estimate false-discovery rates and to introduce statistical significance to fold-change-based methods.

2.2.1.1 Fold-change

Fold-change measures the change of expression of one gene relative to another. It is simply defined as:

$$FC(g) = \overline{x}_1/\overline{x}_2 \tag{2.4}$$

Or

$$FC(g) = \overline{x}_1 - \overline{x}_2 \tag{2.5}$$

where $\overline{x}_1$ is the mean expression value of g in one class and $\overline{x}_2$ is the mean expression value of g in the other class. Fold-change describes relative quantity without using any information about the distribution of data between the two classes.

2.2.1.2 t-test

Another way to define differential genes is the use of a statistical t-test, which tests the null hypothesis that the two distributions have equal mean.

The t-test computed on a gene g is based on the t-statistic formula:

$$T(g) = \frac{\overline{x}_1 - \overline{x}_2}{se} \tag{2.6}$$

where $\overline{x}_1$ and $\overline{x}_2$ denote the mean gene-expression value for the two classes respectively and se refers to the standard error of the difference between two means and has different forms depending on the assumptions on the data distribution. The most commonly used variants are described below.

The two classes have unequal sample sizes but equal variance:

$$se' = \sqrt{\frac{(N_1 - 1)s_1^2 + (N_2 - 1)s_2^2}{N_1 + N_2 - 2}} \cdot \sqrt{\frac{1}{N_1} + \frac{1}{N_2}} \tag{2.7}$$

In this case, the variances of the 2 classes are pooled to compute the standard error, s_1^2 and s_2^2 denote the variance of the two classes respectively, N_1 and N_2 denote the sample size of the two classes respectively. The t-test is computed under $N_1 + N_2 - 2$ degrees of freedom.

The two classes have unequal sample sizes and unequal variance:

$$se'' = \sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}} \tag{2.8}$$

where s_1^2 and s_2^2 denote the variance of the two classes respectively, N_1 and N_2 denote the sample size of the two classes respectively. Welch provides an approximation for the degrees of freedom for this test, computed as:

$$d.f. = \frac{\left(\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}\right)^2}{\frac{\left(\frac{s_1^2}{N_1}\right)^2}{N_1 - 1} + \frac{\left(\frac{s_2^2}{N_2}\right)^2}{N_2 - 1}} \tag{2.9}$$

It has been reported that most biological analyses involving the t-test use the first formula to compute the standard error (Ruxton, 2006) but have unstable performance in terms of type-I errors. For example, when the significance threshold is set at 0.05, the standard error computed assuming equal variance tend to have higher amount of type-I errors when one of the classes has smaller sample size but larger within-class variance. And, when one of the classes has smaller sample size and smaller within-class variance, the amount of type-I errors is smaller than expected. Table 2.1 shows the effect on type-I errors when se' and se" are used to compute the t-test.

TABLE 2.1: Effects of standard error on t-test (Ruxton, 2006).

N_1	N_2	s_1	s_2	Type I error	
				se'	se''
11	11	1	1	0.052	0.051
11	11	4	1	0.064	0.054
11	21	1	1	0.052	0.051
11	21	4	1	0.155	0.051
11	21	1	4	0.012	0.046
25	25	1	1	0.049	0.049
25	25	4	1	0.052	0.048

2.2.1.3 Wilcoxon rank-sum test

When the underlying distributions of the two classes are not necessarily normal, the t-test may not provide the best estimate of the p-value for the differential expression. The Wilcoxon rank-sum test provides an alternative solution in this situation. It tests the null hypothesis that the two distributions have equal median.

The Wilcoxon statistic U computed for a gene g is defined as:

$$U(g) = min(U_1, U_2) \tag{2.10}$$

where: $U_1 = R_1 - \frac{n_1(n_1+1)}{2}$, $U_2 = R_2 - \frac{n_2(n_2+1)}{2}$

and R_i is the sum of ranks in class i, and n_i is the number of samples in class i.

2.2.1.4 SAM

When the t-test and Wilcoxon test are used, a confidence interval $(1-\alpha)$ and significance level (α) are often defined. If each statistical test incurs a false-positive rate of 5%, then evaluating a microarray of 10,000 genes we would expect 500 false positives. SAM (Tusher et al., 2001) is a method for better controlling this.

SAM extends from the t-test by introducing a few modifications:

1. Modify the t-statistic for each gene g as follows:

$$T'_g = \frac{\overline{x_1} - \overline{x_2}}{se' + s_0} \tag{2.11}$$

The modified t-statistic includes a small positive constant s_0 because the t-statistic becomes artificially inflated when the standard-error term in the denominator is very small. The value for s_0 is chosen to minimize the coefficient of variation. Note that in the new version of SAM, a Wilcoxon test statistic is provided as an alternative to the t-statistic.

2. Use a permutation procedure to estimate significance

   Let $T'_1 < T'_2 < \ldots < T'_k$ be the ordering of k genes sorted in increasing order of the modified t-statistic. The permutation test randomly swaps the class labels of the original data, preserving the proportions of the classes, and the modified t-statistic is computed using this permuted set for each gene. The same ordering can be performed on these statistics computed for the permuted data. For example, let $T_j^{''i}$ be the statistic computed for the i^{th} permutation such that $T_{j+1}^{''i} > T_j^{''i}$, then the permutation procedure might produce the following ordering after b number of permutations:

$$\begin{array}{ccccc} T_1^{''1} & < & T_2^{''1} & < \ldots < & T_k^{''1} \\ T_1^{''2} & < & T_2^{''2} & < \ldots < & T_k^{''2} \\ \vdots & & \vdots & & \vdots \\ T_1^{''b} & < & T_2^{''b} & < \ldots < & T_k^{''b} \end{array}$$

FIGURE 2.2: Permutation procedure for SAM.

The ordered statistics over b permutations can be averaged:

$$T_i^{''B} = \frac{\sum_{j=1}^{b} T_i^{''j}}{b} \tag{2.12}$$

Since, all the T' and $T^{''B}$ are ordered, they can be plot against each other. If all the statistics are derived from the null distribution, then we expect the statistics

correlate perfectly to line up to form a 45-degree diagonal. The significant genes therefore deviate from this diagonal; the algorithm selects a parameter δ to achieve this.

3. Estimate false-discovery rate based on the permuted data

   In practice, the δ parameter is automatically selected based on the specified FDR. In order to achieve this, the algorithm finds the smallest T'_a such that genes that fall below this threshold are significantly repressed and the largest T'_b such that genes that lie above it are significantly overrepresented. The number of false positives is estimated based on the previously computed statistics on the permuted data. This is simply the average count of T'' that exceeds the T'_a and T'_b thresholds over all permutations.

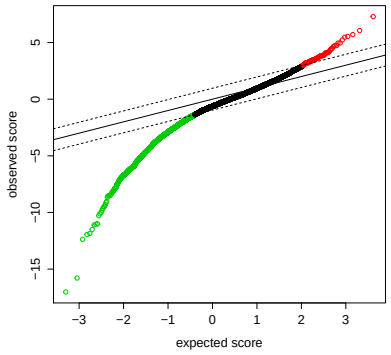


FIGURE 2.3: Plot of observed T' and expected T'' in SAM (Tusher et al., 2001).

2.2.1.5 Rank Products

The methods discussed thus far advocate the use of some statistical test over simple fold-change because (1) fold-change does not give any statistical significance, and (2) the thresholds chosen can be very arbitrary. On the other hand, statistical methods that select differentially expressed genes lack the intuitive appeal of fold-change. It is

possible for genes to have very high statistical significance but very low fold-change, whereas biologists tend to lend more confidence to genes with higher fold-change than purely based on statistical significance.

Rank products (Breitling et al., 2004) are introduced to measure the statistical significance based on fold-change. The algorithm ranks the fold-changes generated for every pairwise sample comparison. For example, if there are m samples in class 1 and n samples in class 2, then there can be $m \times n$ number of fold-changes for every gene.

$$\begin{array}{ccc} FC_{g1}^{(1,1)}, FC_{g1}^{(1,2)}, ..., FC_{g1}^{(m,n)} & & R_{g1}^{(1,1)}, R_{g1}^{(1,2)}, ..., R_{g1}^{(m,n)} \\ FC_{g2}^{(1,1)}, FC_{g2}^{(1,2)}, ..., FC_{g2}^{(m,n)} & \xrightarrow{\text{column-wise ranking}} & R_{g2}^{(1,1)}, R_{g2}^{(1,2)}, ..., R_{g2}^{(m,n)} \\ \vdots \quad \vdots \quad \vdots & & \vdots \quad \vdots \quad \vdots \\ FC_{gk}^{(1,1)}, FC_{gk}^{(1,2)}, ..., FC_{gk}^{(m,n)} & & R_{gk}^{(1,1)}, R_{gk}^{(1,2)}, ..., R_{gk}^{(m,n)} \end{array}$$

FIGURE 2.4: Example of rank product computation.

The rank product of a gene is defined as the geometric mean of its ranks across all the comparisons.

$$RankProduct(g_i) = \left(\prod_{a=1,b=1}^{m,n} R_{g_i}^{(a,b)} \right)^{\frac{1}{mn}} \tag{2.13}$$

In order to give rank products a statistical significance, a permutation test is performed. The permutation test computes the rank products obtained after randomly swapping the class labels. This generates a null distribution from which the p-values can be derived.

2.2.2 Gene-set-based methods

For several years now, there has been a paradigm shift from looking at individual genes to gene sets. Such methods avoid large multiple hypothesis testing by preselecting gene

sets using biological knowledge. These gene sets are often termed "pathways" in the literature, and are groups of genes that perform a specific function. These methods can be classified into four categories, described in separate sections below.

2.2.2.1 Over-representation analysis

Over-representation analysis (ORA) is a method that tests if the proportion of differentially-expressed genes (DEG) in a pathway are significantly different from the proportion in a random set of genes (Khatri and Drăghici, 2005). The method utilizes a hypergeometric test under the null hypothesis that there is no difference in the proportion of differentially-expressed genes (DEG) between a pathway and a random gene set.

The hypergeometric test is motivated from sampling without replacement. The probability of observing k DEGs in a population of N microarray genes, in a given pathway, can be represented by the following formula:

$$P(X = k) = \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}} \tag{2.14}$$

The numerator of the eq.2.14 represents the number of ways of sampling k DEGs in the pathway from K observed DEGs in the microarray and the number of ways of sampling $(n - k)$ non-DEGs in the pathway from $(N - K)$ non-DEGs in the microarray, and the denominator is the number of ways of sampling n genes in the pathway from N microarray genes.

This sampling without replacement can be better understood with a diagram depicting the overlap of DEGs in the microarray and the DEGs in the pathway (see figure 2.5). The hypergeometric test provides a p-value that computes the probability of observing an overlap of more than k.

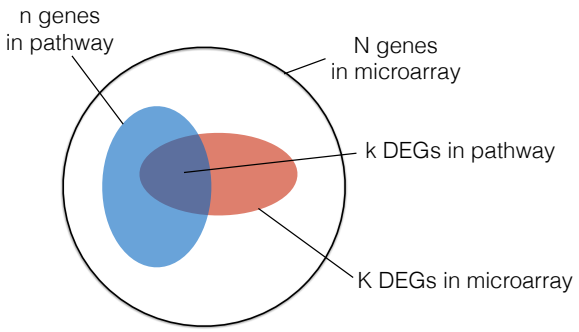


FIGURE 2.5: Figure depicting the calculations for the hypergeometric test.

$$p\_value = P(X > k) = 1 - \sum_{i}^{k} P(X = i) \tag{2.15}$$

The p-value in ORA provides a way to rank pathways according to their correlation with a disease.

2.2.2.1.1 Discussion ORA has a few shortcomings. (1) There are many ways to compute k DEGs, e.g. by a fold-change or t-test, but these methods are largely affected by the choice of thresholds used to select the DEGs. (2) Inability to detect surrounding genes that are also implicated but not differentially expressed. (3) The hypothesis of the test implies that ratio of DEGs are no different in a pathway than a random gene-set; however as genes in a pathway are generally coordinated in their behaviour to perform the specific function associated with the pathway, this null hypothesis is generally false; hence the p-value tends to be underestimated.

2.2.2.2 Direct-group methods

Direct-group methods avoid the problem of ORA by using all the genes within the pathway to compute a score instead of pre-selecting some DEGs within the pathway. The two most popular methods in this category are functional class scoring (FCS) and

gene set enrichment analysis (GSEA). They differ in the way the scores are computed. These are detailed below:

2.2.2.2.1 Functional Class Scoring FCS (Goeman et al., 2004) first considers gene-wise scores, which could be derived from t-test, analysis of variance (ANOVA) or fold-change. The scores are aggregated for each pathway by taking the arithmetic mean of the $-\log(\text{p\_value})$. A null distribution of aggregated scores is obtained by a permutation test that randomly selects a set of genes of the same size as the pathway being evaluated. The p-value assigned for a pathway is the proportion of permutations that have aggregated scores higher than the score computed for the original data.

2.2.2.2.2 Gene set enrichment analysis Gene set enrichment analysis (GSEA) is another direct group method that provides a logical hypothesis for evaluating whether the genes in a pathway are differentially expressed between two classes (Subramanian et al., 2005). The algorithm works by computing a ranked gene list, which orders the gene from the most differentially expressed to the least. A running-sum score is then computed by running through the genes one by one in the ranked list starting from the most differentially expressed, increasing (decreasing) the score every time the same gene is (is not) encountered in the pathway. The enrichment score for a pathway is a Kolmogorov-Smirnov-like statistic and is the maximum deviation of the running-sum score from zero.

A p-value is computed for each pathway by permutation test. When there a sufficiently-large number of samples, GSEA provides a p-value by computing a null distribution via class-label swapping. When there are few samples, GSEA provides the option of gene swapping to compute the null distribution.

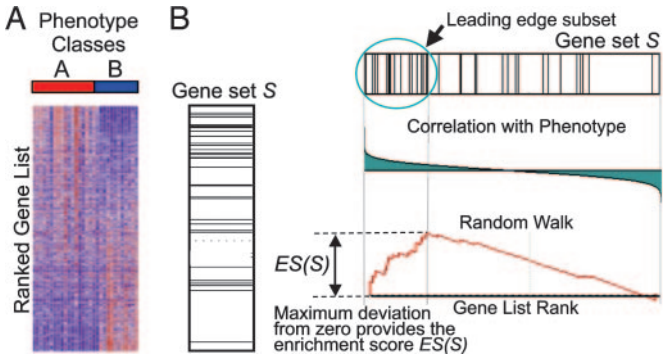


FIGURE 2.6: An example depicting how GSEA works (Subramanian et al., 2005).

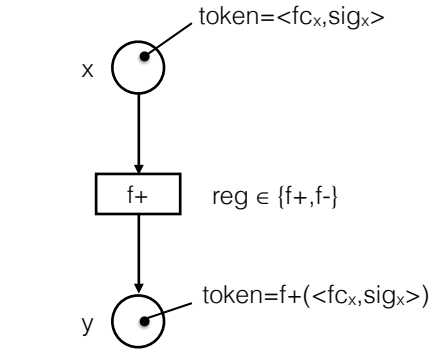
2.2.2.2.3 Discussion FCS and GSEA have the same shortcoming that parts of the large pathway that are not correlated to disease may dilute the signal if only a small branch of the pathway is relevant to disease. In addition, although pathways provide better biological interpretation than analyzing single genes, the size of some pathways are large and provide too vague an insight to the disease.

Moreover, in the permutation test in FCS and GSEA (when sample size is small) creates random sets of genes by gene swapping, ignoring the gene-gene correlation within a pathway. The p-values obtained by such a procedure can be underestimated since no coordination is expected from a random set of genes.

2.2.2.3 Model-based methods

Another way to discover disease-related pathways is to construct a dynamical model for the pathways and then reason about the constructed model. For example, a model of a disease-related pathway constructed for the disease phenotype is expected to be inconsistent when the same model is simulated on the normal phenotype. We describe two model-based methods, GGEA (based on Petri Nets) and SRI (based on ordinary differential equations):

2.2.2.3.1 Gene graph enrichment analysis GGEA is based on a Petri Net model (Geistlinger et al., 2011). The gene sets are mapped to a gene-regulatory network (GRN), thus forming induced subnetworks. These induced subnetworks are modeled as Petri nets, where the 'places' are genes and the 'transitions' are the regulatory effects. The tokens in a Petri net are 2-tuples representing fuzzy fold-change and t-test p-values. Fuzzification is a procedure that maps fold-change and t-test p-values into a value between 0 and 1 based on linear interpolation. A transition is fired based on predefined rules on the regulatory element. For example, a toy GRN depicted in figure 2.7 shows gene x activating gene y; if the place for gene x contains a token with an upregulated value, then an activating transition is fired, producing a token in the target gene y with an upregulated value (see figure 2.7).



rules	fold-change		significance	
	Down	Up	Low	High
f+	Down	Up	Low	High
f-	Up	Down	Low	High

(b)

FIGURE 2.7: An example depicting firing of a transition in a Petri net in GGEA. (a) A Petri net is modelled based on a gene regulatory network. In this example, the regulatory effect (f+) causes gene x to activate gene y. A token in node x is annotated with fold-change and p-value significance from a t-test. A transition is fired based on the rules defined by the regulatory effect. This results in a token in the output node y annotated with a fold-change and p-value based on the activation rule. (b) A set of rules governing the firing of a transition, where f+ denotes activation and f- denotes inhibition. For example an inhibitory regulatory element will map an "Up" fold-change to a "Down" fold-change

GGEA checks if the tokens in the output place in a Petri net contain values that agree with the actual data. GGEA compares these values to compute a consistency score. The consistency score of a pathway is the sum of individual genes' consistency in the

pathway normalized by the size of the pathway. This normalized score is converted to a p-value by a permutation test: class labels of the samples are randomly swapped and the consistency scores for each pathway recomputed, forming a null distribution. The actual p-value is the proportion of permutation scores larger than the observed score.

2.2.2.3.2 System response inference SRI first constructs a dynamical model for one phenotype (Zampieri et al., 2011). To do this, it first identifies gene-gene interactions that are present in each pathway by computing a simple pair-wise Pearson correlation. Two genes are inferred as being in the same pathway if their correlation exceeds a certain threshold. A system of linear differential equations (ODE) are then constructed for these putative interactions. Once the parameters for this system of ODEs are estimated, the model is simulated on the opposite phenotype. The predicted gene-expression values are compared against the actual experimental gene-expression values via a t-test, thus identifying genes that are perturbed in the two phenotypes.

2.2.2.3.3 Discussion Model-based methods work on very fine-grained pathways, while there are many large pathway databases, there are much fewer fine-grain pathways. In addition, these methods also examine gene-expression profiles over many different conditions, which is rarely available on the same microarray platform.

2.2.2.4 Network-based methods

Network-based methods address the shortcomings of direct-group methods by fragmenting the large pathways into smaller components (subnetworks) and testing them for significant correlation to phenotypes. We describe three methods in this category: network enrichment analysis (NEA), differential expression analysis for pathways (DEAP) and

significant subnetworks (SNet). They differ in the way the subnetworks are generated, as well as in the scoring method used to generate a p-value for statistical significance.

2.2.2.4.1 Network enrichment analysis NEA (Sivachenko et al., 2007) maps every gene in the microarray onto a gene regulatory network. For every such gene, its immediate neighborhood in a pathway forms a subnetwork. The subnetworks are then evaluated using statistics from direct-group methods like FCS or GSEA to see if the subnetworks are differentially expressed as a whole.

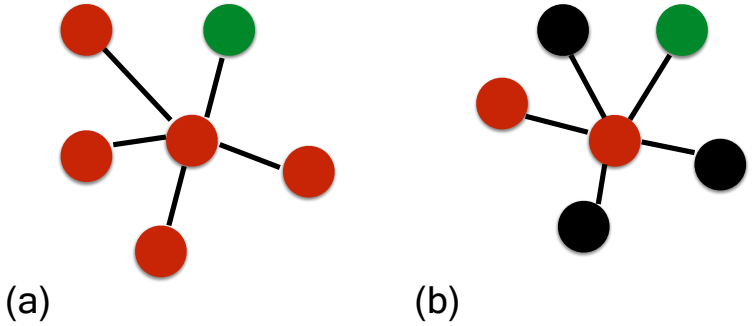


FIGURE 2.8: An example depicting the subnetworks in NEA. The green nodes denote genes upregulated in class 1, the red nodes denote nodes upregulated in class 2 and the black nodes denote genes with no differential expression. (a) A disease-relevant subnetwork predicted by NEA are"star" shaped and do not fully explain the biological cause of diease, which usually involves a cascade of genes affecting each other. (b) Some "hub" genes may contain many edges to other non-relevant genes; hence these subnetworks may be missed because the genes with no differential expression dilute the signal.

Although NEA tries to circumvent issues in direct-group methods by testing smaller parts of the pathway, it suffers from a few shortcomings. Firstly, the subnetwork generated are "star" shaped, and only provide a partial explanation to the disease, whereas diseases are usually caused by a cascade of genes whose upstream genes exert an effect on downstream genes. Secondly, the size of the neighborhood greatly affects the precision of the method because it is possible for a huge number of non-differentially expressed neighbors to dilute the signal of a disease-relevant subnetwork (see figure 2.8).

2.2.2.4.2 Differential expression analysis for pathways DEAP (Haynes et al., 2013) considers all possible maximal paths within a pathway. Each gene is given a differential-expression score defined to be the difference between the logarithm of the arithmetic mean of expression values in the two phenotypes. The algorithm chooses the path with the maximum absolute differential expression score for each pathway. The score given for each path is recursively computed based on the catalytic or inhibitory edges taken as positive and negative summands respectively. For example, consider a maximal path in a pathway with 6 genes in figure 2.9, where the green nodes represents a differential expression score of +1 and the red nodes represents a differential expression score of -1, the score of this path is computed in equation 2.16.

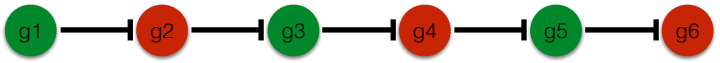


FIGURE 2.9: An example of a maximal path annotated with differential expression scores. The red nodes denote repressed genes and the green nodes denote activated genes.

$$g_1\text{–}(g_2\text{–}(g_3\text{–}(g_4\text{–}(g_5\text{–}g_6)))) = 1\text{–}(-1\text{–}(1\text{–}(-1\text{–}(1-(-1))))) = 6 \tag{2.16}$$

Although DEAP breaks up large pathways into smaller paths, maximal paths may not be by themselves differentially expressed. For example, it is possible for a subpath of a maximal path within a pathway to be correlated to phenotype, but this can be missed since the other genes in the maximal path might dilute this signal.

2.2.2.4.3 SNet Among all the methods discussed thus far, SNet (Soh et al., 2011) seems to be most motivated from a biological perspective. SNet addresses problems in the previous approaches in two ways. Firstly, the subnetworks are generated in a way that has a propensity to form connected clusters of genes. This is done by specifying

some constraints on the genes that result in a gene list segmenting the pathway into connected components: In at least $\beta\%$ of the patients of the same phenotype, the genes must be among the highly-expressed (i.e., in the top $\alpha\%$) genes in each of these patients.

We can formulate this in mathematical terms. Let $I(e_{g_i,p_j})$ be an indicator function that outputs a value 1 if g_i is in the top $\alpha\%$ of the genes in patient p_j and a value 0 otherwise. Then a gene list is formed by a voting procedure:

$$\sum_{p_j \in D} \frac{I(e_{g_i,p_j})}{|D|} > \beta\% \tag{2.17}$$

This means that we observe on average $\beta\%$ of the patients of phenotype D have gene g_i in their top $\alpha\%$ of highly expressed genes.

As some of these subnetworks may not be truly correlated to a particular phenotype, each subnetwork is given a score for each patient by summing up the votes every time a gene g_i in the subnetwork is encountered in the top $\alpha\%$ of highly expressed genes in a patient p_j. Let $\beta^*(g_i)$ denote the votes given for each gene g_i:

$$\beta^*(g_i) = \sum_{p_j \in D} \frac{I(e_{g_i,p_j})}{|D|} \tag{2.18}$$

Then, the score computed for a sample p_k, for a particular subnetwork S, is:

$$Score^{p_k}(S) = \sum_{g_i \in S} I(e_{g_i,p_k}) * \beta^*(g_i) \tag{2.19}$$

A t-test is computed over the scores for the class of patients with phenotype D and the scores for the class of patients with phenotype $\neg D$, and a p-value is computed based on the null distribution generated via class-label swapping.

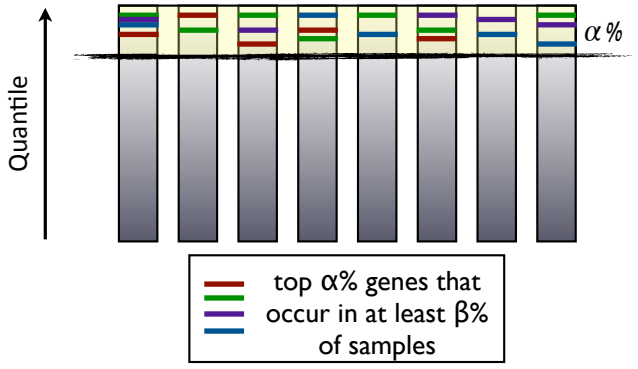


FIGURE 2.10: An example depicting how SNet works.

2.2.2.4.4 Discussion Network-based methods offer biologists more insightful glimpse at the mechanisms of disease, because they scrutinise and narrow down genes within a pathway that are plausibly linked to disease. The effective use of gene relationships represented as edges in the network provides these methods this extra benefit compared to direct-group methods. The principle behind selecting these small subnetworks is crucial to the biological interpretation of the results. In addition, the scoring of subnetworks for statistical significance also play a big role in the performance of these methods.

2.2.3 Permutation tests

A commonly-recurring sub-routine in many methods mentioned in the previous sections is a permutation test to compute p-values identifying significant genes, pathways or subnetworks. It is therefore necessary and important to discuss how various permutation-test methods can affect the performance of methods that utilize permutation tests for p-value computation.

A p-value is defined as the probability of obtaining a statistic as extreme as the observed value assuming the null hypothesis is true. The null hypothesis is a statement about the probability model generating the data, e.g. the classes are from two normal distributions

with equal means, as in a t-test. We reject the null hypothesis when the p-value is very small (usually < 0.05).

$$p\_value = P(X \geq x|H_0) \tag{2.20}$$

P-values can be obtained in a number of ways. First, well-studied test statistics like the t-statistic follow a theoretical null distribution derived from theoretical calculations under certain mathematical assumptions, from which p-values can be computed. But the null distribution may be incorrect when the mathematical assumptions are violated. Moreover, many methods compute scores that do not have theoretical null distributions. This motivates the need for a computational procedure that allows the estimation of null distribution for p-value computation purpose. We describe below 3 procedures that are commonly used in the literature.

2.2.3.1 Class-label swapping

Class-label swapping, as its name suggests, permutes the data by randomly swapping the class labels of the samples while preserving the size of the classes. Figure 2.12(a) shows an example of 5 samples in class 1 and 5 samples in class 2 depicted by red and blue respectively and the transformation of the input matrix after one round of permutation.

The number of permutations possible largely depends on the sample size. For example, if we have 10 samples in class 1 and 9 samples in class 2, then we can have $\binom{19}{10}$ unique ways to permute the data. Hence, the number of permutations that can be computed is $\binom{N+M}{M}$. The number of permutations also limits the granularity of the p-value that can be achieved by this method. For example, with 1000 permutations, the smallest possible p-value is 10^{-3}.

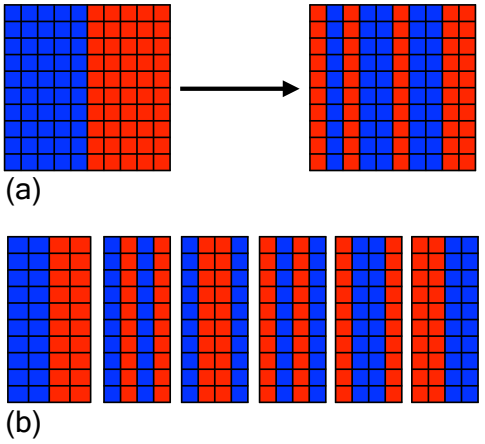


FIGURE 2.11: An example of class-label swapping. (a) In data with moderately large sample size, the permutation test provides a p-value with reasonable granularity. (b) With very small sample size, the p-value is not small enough to properly reject the null hypothesis. In this case, the smallest p-value that can be computed is 1/6.

When the sample size is small, it is impossible to get p-values of fine granularity with class-label swapping because there are not enough unique permutations that can be generated from the data. For example, figure 2.12(b) depicts the 6 unique ways the class labels can be reassigned, hence the smallest p-value is 1/6.

2.2.3.2 Gene swapping

Because class-label swapping cannot be used when sample size is small, gene-label swapping has been introduced in many methods to provide a way to estimate p-values under this situation. As microarray experiments typically involve thousands of genes, the number of ways to permute the gene labels is sizable enough to produce small p-values. For example, if an array contains m genes, then there are $m!$ permutations and the smallest p-value is $1/(m!)$.

The issue with gene swapping, however, is that gene-gene covariance within the microrarray is not preserved during the permutation process. Hence, the null distribution

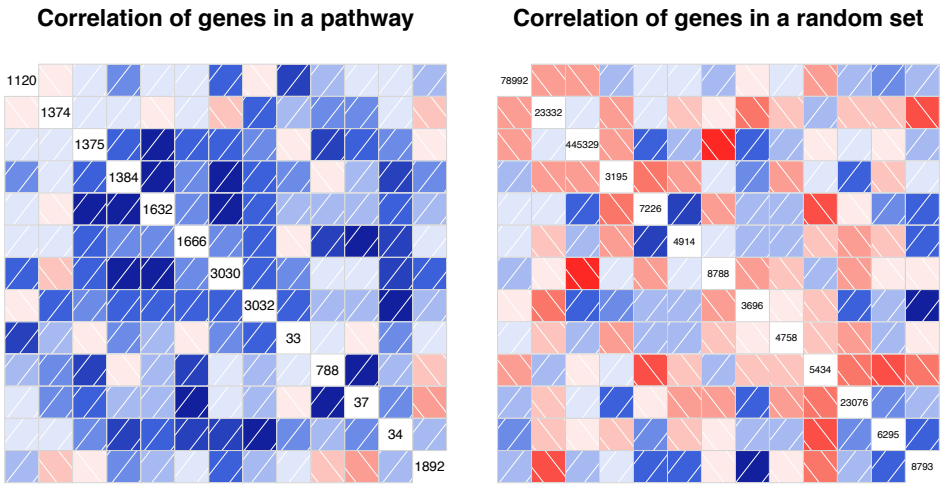


FIGURE 2.12: Gene swapping does not preserve the correlation between genes, unlike class-label swapping. An example correlation matrix is visualized here, with blue for positive correlation and red for negative correlation, the shade of color represents the strength of correlation for a pathway. The genes in the pathway (left) have generally positive correlation, but when these genes are swapped for random genes (right), the correlation between the genes changes dramatically.

generated by gene-label swapping procedure produces a p-value that is severely underestimated, resulting in a tendency of rejecting the null hypothesis.

2.2.3.3 Array rotation

In order to overcome this problem, array-rotation procedures compute a matrix that represent the inherent covariance between the genes in the samples (Dorum et al., 2009). A random rotation matrix is then used to simulate new arrays, at the same time keeping the covariance invariant across the number of rotations.

The underlying principle of array rotation is QR decomposition, which decomposes an input matrix into two matrices.

$$X = X_Q.X_R \tag{2.21}$$

where X is an $n \times m$ gene-expression matrix with n samples and m genes. X_Q is a $n \times r$ orientation matrix with r othonormal columns, where r is the rank of the X. X_R is an upper triangular configuration matrix with positive diagonal elements and is also a sufficient statistic for the covariance matrix between the genes in X.

The rotation procedure keeps the X_R configuration matrix as an invariant, while a random matrix R^* is used to rotate the X_Q orientation matrix. R^* is computed by simulation of a $n \times n$ matrix, R, of random normal deviates and then taking the Q component after QR decomposition has been performed on the R matrix.

Let R be a simulated $n \times n$ matrix of random normal deviates, then R* is computed as follows:

$$R^* = R_Q.R_R \tag{2.22}$$

The final rotated matrix is therefore computed as follows:

$$X^* = R_Q.X_Q.X_R \tag{2.23}$$

The rotation procedure offers two advantages over class-label swapping and gene-label swapping. Firstly, it has the ability to handle data with small sample sizes since there is an unbounded number of ways to rotate the input matrix. Secondly, gene-gene correlations within a pathway are kept; hence a more reasonable null distribution is computed.

However, one problem with array rotation is that artificial gene-expression values can be created and may not have any resemblance to the real gene-expression values. A second problem is that, when sample size is very small, the gene-gene covariance computed

may not capture the actual gene-gene correlations dictated by the underlying biological pathways with high fidelity.

2.3 Background on classification in microarray analysis

It is possible to further extend the analysis pipeline after identifying relevant genes, pathways and subnetworks by predicting patient phenotypes and outcomes based on gene expression. The objective here is to build a supervised-machine-learning classifier that is able to accurately distinguish the class labels of patients given their expression profiles. This typically comprises of a few steps, described below.

2.3.1 Feature selection

In section 2.2, we have already discussed many different ways to shortlist genes, pathways and subnetworks based on some separation statistics. Most of these methods can be used to select a non-redundant set of features as input to the classifier.

2.3.2 Classification

There are many supervised-machine-learning techniques for use in this area, here we review four frequently-used classification methods:

2.3.2.1 Decision trees

Decisions can be very intuitively interpreted in the form of a tree. The internal nodes of the tree represent tests on the feature attributes and the branches represent outcomes of the tests. The leaf nodes are the decisions representing the class labels.

The algorithm works by recursively choosing an attribute that best splits the data into subsets that are enriched in one class over the other. The algorithm terminates on a few base conditions:

1. When a perfect split is achieved, i.e. 100% of the samples belong to one class only, then a leaf node is created and is denoted by the class label.
2. When all the attributes have been exhausted, the leaf node is then assigned to be the class that has the majority of samples.

There are many ways in which a node can be split, the most commonly used measures are described below.

2.3.2.1.1 Information gain Information gain chooses the attribute that minimizes the information content required to classify the samples in the resulting partitions.

It is based on the entropy of the data, representing the expected information required to classify a sample, defined as:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i) \tag{2.24}$$

where p_i is the proportion of samples in class i in the dataset D.

The amount of information still required to arrive at an exact classification after using an attribute A to partition the data is defined as:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j) \tag{2.25}$$

where D_j is the number of samples in the j^{th} partition of attribute A, and v is the total number of distinct partitions of attribute A.

The information gain is the difference between the two values above:

$$Gain(A) = Info(D) - Info_A(D) \tag{2.26}$$

Attributes with the maximum gain are chosen to split the nodes in the decision tree.

2.3.2.1.2 Gini index Gini index measures the impurity of the dataset:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2 \tag{2.27}$$

where p_i is the proportion of samples in class i in the dataset D.

The Gini index considers binary split for each attribute. The Gini index for the resulting split using an attribute A is computed as:

$$Gini_A(D) = \frac{|D_1|}{|D|}Gini(D_1) + \frac{|D_2|}{|D|}Gini(D_2) \tag{2.28}$$

where D_1 and D_2 are the resulting partitions of the binary split.

The attribute that maximizes the reduction in data impurity is then selected as the splitting attribute:

$$\Delta(A) = Gini(D) - Gini_A(D) \tag{2.29}$$

2.3.2.2 k-Nearest Neighbors (kNN)

The kNN is a lazy classifier that compares a test sample with training samples that are similar to it. If each sample is represented by n attributes, then the feature space is an n-dimensional space. kNN searches for the k training samples that are closest to the

test sample in this n-dimensional feature space. The classifier then makes a prediction by a simple majority voting on the k nearest training samples.

The measure of distance between test and training samples in the n-dimensional space can be done using various distance metrics:

1. Euclidean distance

$$Dist(x, y) = \sqrt{\sum (x - y)^2} \tag{2.30}$$

2. Minkowski distance

$$Dist(x, y) = (\sum (x - y)^p)^{\frac{1}{p}} \tag{2.31}$$

   where p is some chosen constant. Note that p = 1 gives the Manhattan distance, and p = 2 gives the Euclidean distance.

3. Manhattan distance

$$Dist(x, y) = \sum |x - y| \tag{2.32}$$

4. Mahalanobis distance

$$Dist(x, y) = \sqrt{(x - y)S^{-1}(x - y)^T} \tag{2.33}$$

   where S is the sample covariance matrix of the features.

2.3.2.3 Support Vector Machines (SVM)

SVM in the linear-separable case seeks the best line or hyperplane in an n-dimensional feature space that best separates the two classes. SVM achieves this by selecting the hyperplane with the largest margin, defined as the shortest distance between the separating hyperplane and the training samples of the classes. The separating hyperplane can be written as:

$$W.X + b = 0 \tag{2.34}$$

where W is a weight vector perpendicular to the separating hyperplane.

Additional constraints are specified so that the points that lie on one side of the hyperplane belong to class 1 and points that lie on the other side belong to class 2.

Let $y = +1$ be the class label of a training sample X if it is in class 1 and $y = -1$ if it is in class 2. Then the additional constraints are defined as:

$$W.X + b > 1 \text{ if } y_i = +1 \tag{2.35}$$

$$W.X + b < -1 \text{ if } y_i = -1 \tag{2.36}$$

The problem is now reformulated as a quadratic optimization problem to solve for the hyperplane and support vectors.

When the data is not linearly separable, a kernel is used to transform the points in this n-dimensional space into another higher-dimensional space so that the points can be linear separable in this new feature space. The three most-commonly-used kernels in SVM are:

1. Polynomial kernel

$$K(X_i, X_j) = (X_i.X_j + 1)^h \tag{2.37}$$

2. Gaussian radial basis kernel

$$K(X_i, X_j) = e^{|X_i - X_j|^2/2\sigma^2} \tag{2.38}$$

3. Sigmoid kernel

$$K(X_i, X_j) = \tanh(\kappa X_i.X_j - \delta) \tag{2.39}$$

2.3.2.4 Naïve Bayesian classifier

The naïve Bayesian classifier maximizes the posterior probability of a sample belonging to a class given its attributes:

$$\arg\max_i P(C_i|X) \tag{2.40}$$

where C_i represents the i^{th} class label and X is a sample represented by an n-dimensional attribute vector.

Based on the Bayes formula, we have:

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \tag{2.41}$$

Since the denominator is independent of the class labels, the posterior probability is proportional to the prior and likelihood, and maximizing the posterior probability is equivalent to maximizing $P(X|C_i)P(C_i)$:

$$P(C_i|X) \propto P(X|C_i)P(C_i) \tag{2.42}$$

Naïve Bayes uses an assumption that the attributes are conditionally independent of one another given the class label of a sample. Hence,

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) \tag{2.43}$$

where $P(x_k|C_i)$ represents the proportion of samples with the k^{th} attribute having the value x_k over the total number of samples in class C_i.

Thus, the naïve Bayesian classifier predicts the class in the following way:

$$\arg\max_i P(C_i|X) = \arg\max_i P(C_i). \prod_{k=1}^{n} P(x_k|C_i) \tag{2.44}$$

2.3.3 Enhancements

It is possible to enhance the predictive accuracy of supervised classifiers using the following techniques:

2.3.3.1 Bagging

The main idea behind bagging (Breiman, 1996) is the use of sampling with replacement of the training samples, known as bootstrapping. The classifier is then built on the bootstrapped training dataset and repeated n times, resulting in n different classifier models.

Given a test dataset, bagging applies all the n models on the test samples and applies a majority voting procedure to predict the class labels of the test samples.

Bagging often produces higher accuracy than the single classifier and is more robust to noise (Koh and Wong, 2012).

2.3.3.2 Boosting

In boosting (Schapire, 1990), the same bootstrapping procedure is applied. And the classifiers are iteratively added and weighted according to their predictive accuracy.

Given a test sample, boosting computes a score for each class based on the weighted votes given by each bootstrapped classifier. The class with the highest score is made as the prediction.

However, current research (Long and Servedio, 2008) suggests that boosting is not robust to noise in the training data.

2.3.4 Evaluation strategies

2.3.4.1 Training and testing on independent datasets

One popular way to evaluate classifiers is by cross validation, a process whereby many classifiers are built and tested on subsets of the original data.

However, cross validation often does not generalize to the heterogeneity in microarray data of the same disease type obtained from different laboratories and at different time-points. The same classifier that learns from one dataset usually does not work when applied to an independent dataset from a possibly different platform or time; cf. fig. 1.4.

In this thesis, we advocate testing the performance of classifiers by training the classifier on one dataset and using this model for classification on an independent microarray dataset of the same disease phenotype.

2.3.4.2 Performance indicators

In most analysis, a classifier's accuracy is used as a measure of its performance, but this ignores the size of class. For example, it is possible to easily achieve 100% accuracy for a dataset which has only samples belonging to class 1; a naïve classifier can just output class 1 all the time to achieve this 100% accuracy. In view of this, we consider other performance indicators:

1. sensitivity or recall

$$\frac{TP}{(TP+FN)} \tag{2.45}$$

2. specificity

$$\frac{TN}{(FP+TN)} \tag{2.46}$$

3. precision

$$\frac{TP}{(TP+FP)} \tag{2.47}$$

where TP, TN, FP and FN refer to the number of true positives, true negatives, false positives and false negatives respectively.

2.4 Datasets

This dissertation is concerned with the hypothesis that specific areas within biological pathways, which we term "subnetworks", are responsible for specific diseases. It has been shown that different pathway repositories have very little agreement between them (Soh et al., 2010a, Stobbe et al., 2011, Zhou et al., 2012). This makes the choice of pathways an important consideration in our study as it can greatly affect the analysis of the methods. We use pathways from PathwayAPI (Soh et al., 2010b). This is a database that unifies popular pathway databases like KEGG (Kanehisa and Goto, 2000, Kanehisa et al., 2012), Wikipathways (Kelder et al., 2012) and Ingenuity (www.ingenuity.com), so that the biological information is as comprehensive as possible. It contains 319 human pathways, 4221 genes and 61017 edges.

The microarray array datasets used in this dissertation are for specific diseases of interest. For each disease type, we use two independent microarray data sets from previously published experiments:

1. Duchenne muscular dystrophy (DMD/NOR)

   Phenotypes of interest are patients suffering from Duchenne muscular dystrophy and normal patients. The first dataset is based on the Affymetrix HG_U95v2 microarray platform and contains 12 DMD patients and 12 normal patients (Pescatori et al., 2007). The second dataset is based on the Affymetrix HG_U133A microarray platform and contains 22 DMD patients and 14 normal patients (Haslett et al., 2002).

2. Leukemia (ALL/AML)

   Phenotypes of interest are patients suffering from acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). The first dataset is based on Affymetrix

HU8600 microarrays and has 47 ALL patients and 25 AML patients (Golub et al., 1999b). The second dataset is based on Affymetrix HG_U95v2 microarrays and has 24 ALL patients and 24 AML patients (Armstrong et al., 2002).

3. Acute lymphoblastic leukemia subtypes (BCR-ABL/E2A-PBX1)

   Phenotypes of interest are patients with the BCR-ABL fusion gene and patients with the E2A-PBX1 fusion gene. The first dataset is based on Affymetrix HG_U95v2 microarrays and has 15 BCR-ABL patients and 27 E2A-PBX1 patients (Ross et al., 2004). The second dataset is based on Affymetrix HG_U133A microarrays and has 15 BCR-ABL patients and 18 E2A-PBX1 patients (Yeoh et al., 2002).

For the work on classification (Chapter 5), we use two addition datasets:

4. Ovarian cancer

   Obtained from ArrayExpress database (www.ebi.ac.uk/arrayexpress) under accession number E-GEOD-4122 and E-GEOD-26712.

5. Lung cancer

   Obtained from ArrayExpress database (www.ebi.ac.uk/arrayexpress) under accession number E-GEOD-29066 and E-GEOD-31908.

Chapter 3

Finding consistent disease subnetworks using PFSNet

3.1 Background

We begin this chapter by investigating one of the network-based methods that has reported high reproducibility of results over independent microarray datasets of the same disease phenotype, SNet (see chapter 1 for a description). This high agreement between two datasets of the same disease lends more confidence to the real cause of disease.

Although SNet claims to have high subnetwork-level overlap and gene-level overlap, it is unclear what happens to its performance over a range of thresholds (α). This is important because on a new dataset, the optimal choice of threshold may vary (the β threshold is fixed at 50% to simulate majority voting). Some important genes that are close to the α% region may fail to get voted into the gene list because they do not meet the majority-voting requirement. One might lower the α threshold to allow these genes

to be voted in, however, the number of false-positive genes voted into the subnetwork is also increased at the same time. (see figure 3.1).

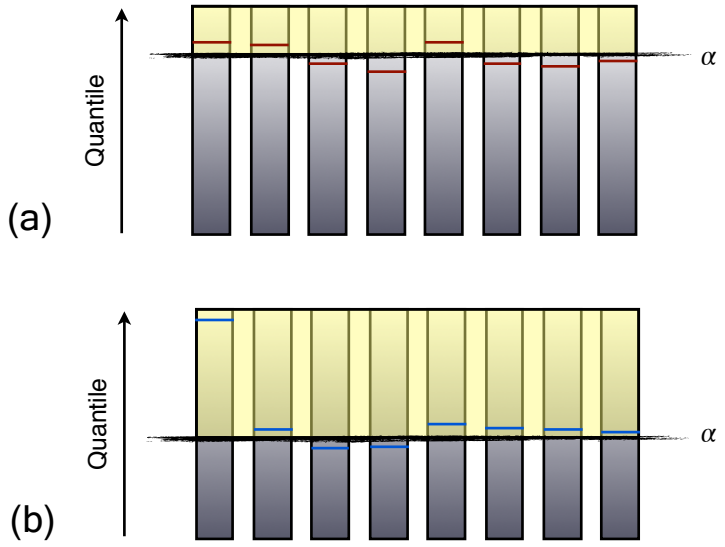


FIGURE 3.1: An example of SNet. (a) Selecting the top 10% of genes may exclude some important genes relavent to disease. (b) By loosening the threshold, more genes are selected but some spurious genes can also be included into the subnetwork, thus diluting the signal.

Indeed the performance of SNet starts to degrade when we allow more genes to be considered by increasing the α threshold. For example, the agreement of significant subnetworks between two datasets of the same disease is shown in figs. 3.2 to 3.4. Note that in their original paper, SNet uses a default α of 10% in their experiments. The results from figs. 3.2 to 3.4 also show that the subnetwork agreement is not always the highest when α is set at 10%. Moreover, when we analyze subnetworks that are upregulated in the two phenotypes of the same dataset separately, the agreement is not always consistently high. For example, when $\alpha = 10\%$, the subnetworks upregulated in AML have higher agreement (about 65%) between the two datasets than subnetworks upregulated in ALL (about 20%). Perhaps this is due to ALL being actually composed of multiple subtypes (Li et al., 2003). In addition, the subnetwork-level agreement in some dataset (e.g. leukemia subtype) is generally low regardless of the α threshold.

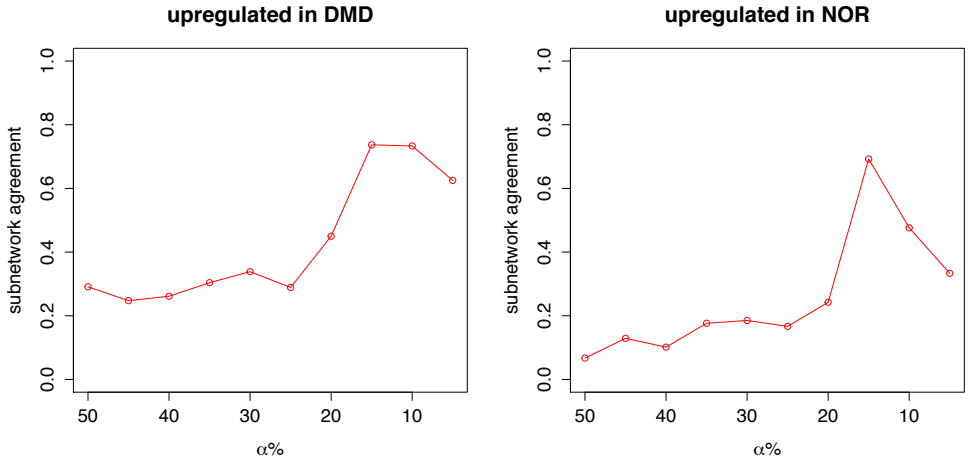


FIGURE 3.2: Subnetwork agreement for SNet in the DMD datasets over a range of α threshold values.

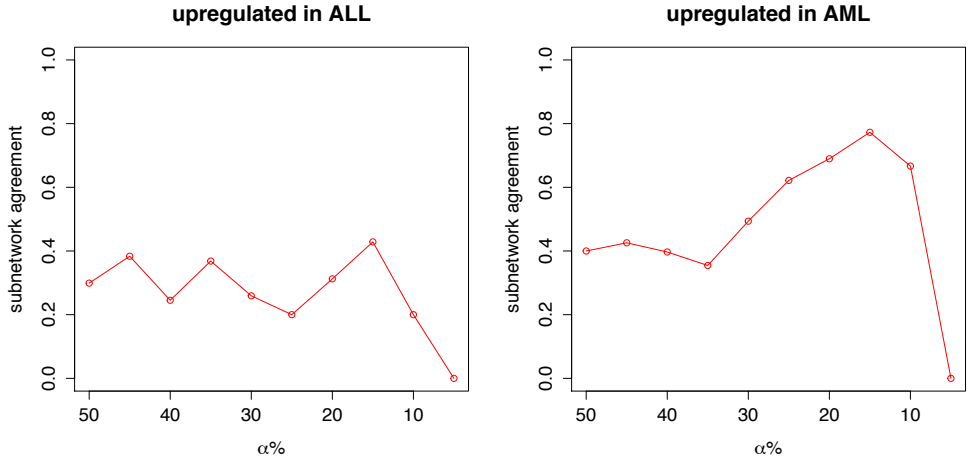


FIGURE 3.3: Subnetwork agreement for SNet in the Leukemia datasets over a range of α threshold values.

3.2 Method

In this chapter, we introduce two improvements, FSNet and PFSNet, which allow us to detect consistent disease subnetworks without the need to select a hard threshold. These generalized versions of SNet are able to predict subnetworks over a range of threshold values with even higher consistency.

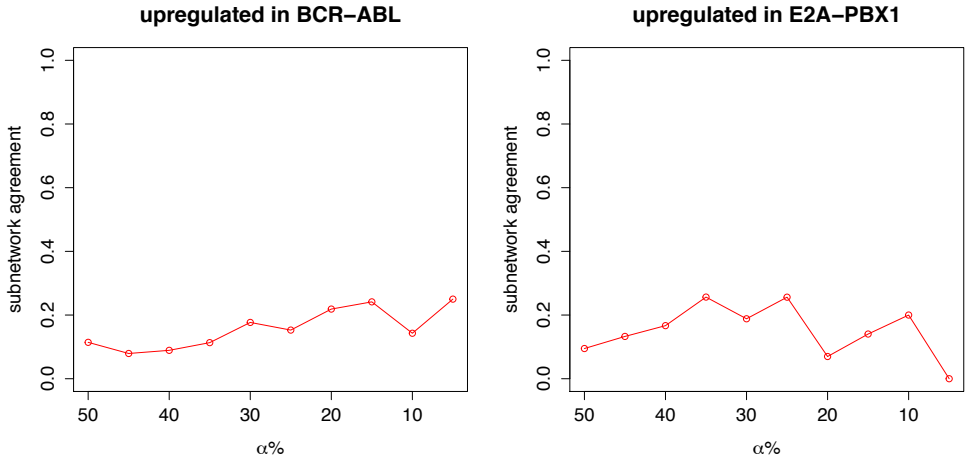


FIGURE 3.4: Subnetwork agreement for SNet in the ALL subtype datasets over a range of α threshold values.

3.2.1 Subnetwork generation

In both FSNet and PFSNet, we assign a fuzzy value, $fs(e_{gi,pj})$, to each gene g_i based on the ranking of its expression value e_{g_i,p_j} within a sample p_j. To do so, we define an upper threshold θ_1 and a lower threshold θ_2. The genes that lie above the top θ_1% are assigned a weight of 1 and the genes below θ_2% are assigned a weight of 0. The genes between θ_1 and θ_2 are given a weight between 0 and 1 by linear interpolation (see figure 3.5). We can think of the fuzzy value as a partial vote given by patient p_j for each gene g_i. In contrast, the patients in SNet give a total vote (of value 1) if g_i's expression is ranked in the top 10% in p_j or give no vote if its expression is not in the top 10% (cf. chapter 1).

Therefore, we can also simulate majority voting by summing up the partial votes given by each patient for a particular gene. The goal at this step is to compute a gene list, which segregates the genes in the pathways into smaller connected components. The voting criteria that determines whether the gene g_i is accepted into this gene list is thus modified as follows:

$$\sum_{p_j \in D} \frac{fs(e_{g_i,p_j})}{|D|} > \beta = 50\% \tag{3.1}$$

where D is the phenotype for which the subnetwork is generated, p_j ranges over the patients of phenotype D and fs is the fuzzy function which converts the gene-expression value e_{g_i,p_j} to a value between 0 and 1. Once the gene list is computed, subnetworks in each reference pathway are generated by taking connected components induced by the genes in this list. We ignore subnetworks that are less than size 5.

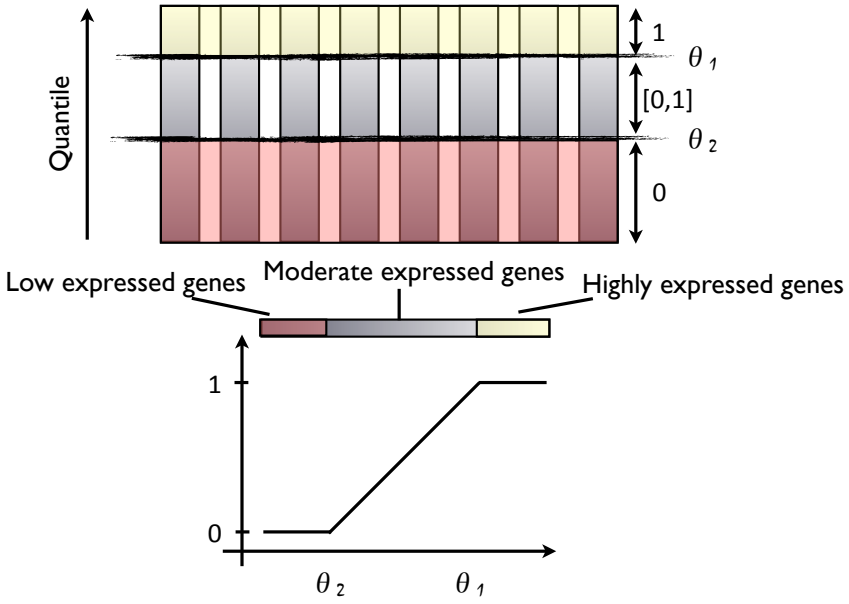


FIGURE 3.5: An example of the fuzzification process, genes in the top θ_1 percentile of the a sample is given weight 1, genes in the bottom θ_2 percentile are given weight 0 and genes in between the two thresholds are given a weight between 0 and 1 by linear interpolation.

3.2.2 Subnetwork scoring

The generated subnetworks are a representation of consistently-abundant genes that are connected in a pathway from one phenotype. Some of these subnetworks may not show

correlation to phenotypes if they are not differentially expressed in the other phenotype. Hence, we need to score the generated subnetworks for their correlation to phenotype.

The key idea used in FSNet is to obtain a distribution of subnetwork scores for each patient in phenotype D and $\neg D$ so that the two distributions can be separated by a t-test.

We use $\beta^*(g_i)$ to denote the average partial vote described by eq. 3.2 in the following:

$$\beta^*(g_i) = \sum_{p_j \in D} \frac{fs(e_{g_i,p_j})}{|D|} \tag{3.2}$$

We hypothesize that the weighted sum of average partial votes would generate two very different scores for the D and $\neg D$ populations if the subnetwork is differentially expressed in the phenotypes. This basically means that the genes within a subnetwork are consistently voted high in one phenotype and consistently voted low in the other phenotype.

The score computed for sample p_k, for a particular subnetwork S, is:

$$Score^{p_k}(S) = \sum_{g_i \in S} fs(e_{g_i,p_k}) * \beta^*(g_i) \tag{3.3}$$

In contrast, in SNet, where total votes are used, the average total vote is the percentage of phenotype-D samples having gene g_i in the top 10%.

The key idea used in PFSNet is to obtain two average partial vote scores (obtained from the D and $\neg D$ phenotype respectively) for each gene in the subnetwork. Let $\beta_1^*(g_i)$ and $\beta_2^*(g_i)$ denote the average partial votes for phenotype D and $\neg D$ respectively, as described in the following equations:

$$\beta_1^*(g_i) = \sum_{p_j \in D} \frac{fs(e_{g_i,p_j})}{|D|} \tag{3.4}$$

$$\beta_2^*(g_i) = \sum_{p_j \in \neg D} \frac{fs(e_{g_i,p_j})}{|\neg D|} \tag{3.5}$$

Accordingly, each patient is assigned a two subnetwork scores, which are the weighted sum of average partial votes computed using the phenotype D and $\neg D$ respectively.

$$Score_1^{p_k}(S) = \sum_{g_i \in S} fs(e_{g_i,p_k}) * \beta_1^*(g_i) \tag{3.6}$$

$$Score_2^{p_k}(S) = \sum_{g_i \in S} fs(e_{g_i,p_k}) * \beta_2^*(g_i) \tag{3.7}$$

We hypothesize that the difference between the two scores for each patient should be non-zero if the genes are consistently voted high in one phenotype over the other.

3.2.3 Statistical test

In FSNet, the subnetwork scores for patients in phenotype D and $\neg D$ form two separate distributions, which can be discriminated using a standard t-test. The t-statistic captures the difference between the population means.

In PFSNet, the two subnetwork scores arise from the same patient. We test the null hypothesis that the difference in these two scores give a distribution with mean equal to 0, using a paired t-test.

3.2.4 Permutation test

As the null distribution may not really be represented by the theoretical distribution (Gatti et al., 2010, Goeman and Bühlmann, 2007, Venet et al., 2011), we generate the null distribution by a permutation procedure. We randomly swap the class labels (1,000 times) for each dataset—i.e., randomly assigning a sample to belong to either phenotype D or $\neg D$ while maintaining the original proportion of D and $\neg D$ samples—and obtain a distribution of subnetwork scores. From this null distribution, we estimate at 5% significance level on one-tail of the distribution, whether a subnetwork that we compute for our original dataset is statistically significant.

3.3 Results

We use the pathways and microarray datasets detailed in chapter 2 to compare our proposed methods with other methods for microarray data analysis. To reiterate, for each disease type we obtain two independent datasets which are produced using different microarray platforms.

We run PFSNet, FSNet, SNet, GSEA, GGEA, SAM and t-test on the two datasets independently and obtain two corresponding outputs.

We compare these two corresponding outputs from the two datasets using two measures of Jaccard-like agreement, defined below.

We use the subnetwork-generation procedure mentioned in section 3.2.1 to generate the subnetworks in dataset 1. We then test these subnetworks for statistical significance using the procedure mentioned in section 3.2.2– 3.2.4 on datasets 1 and 2 independently. Let the significant subnetworks identified by dataset 1 and dataset 2 be SN_1 and SN_2

respectively. Then the subnetwork-level agreement is defined as

$$\frac{|SN_1 \cap SN_2|}{|SN_1 \cup SN_2|} \tag{3.8}$$

When testing GSEA, which identifies pathways instead of subnetworks, we measure the pathway-level agreement which is defined analogously.

We also measure the agreement between the genes in the output generated by the two independent datasets. Let the genes in SN_1 and SN_2 be G_{SN_1} and G_{SN_2} respectively, then the gene-level agreement is defined as

$$\frac{|G_{SN_1} \cap G_{SN_2}|}{|G_{SN_1} \cup G_{SN_2}|} \tag{3.9}$$

3.3.1 Comparing PFSNet, FSNet and SNet

FSNet is flexible enough to be able to emulate SNet by setting $\theta_1 = \theta_2 = 10\%$. In this way, genes above the 90th percentile are given a total vote and genes below the 90th percentile are given no vote at all. This is equivalent to setting SNet with $\alpha = 10\%$.

When comparing PFSNet, FSNet and SNet, we fix $\theta_1 = 5\%$ and vary θ_2 between 5% and 50% for PFSNet and FSNet. We also vary α between 5% and 50% for SNet. This allows more genes to be considered in the subnetworks in all the methods. β is set at 50% to emulate majority voting; cf. figs. 3.6 to 3.8.

Our experiments show that when θ_1 (in FSNet and PFSNet) or α (in SNet) is low, the subnetworks may not be a true representation of the disease simply because too few genes are chosen to induce the subnetworks. But when too many genes are considered, there are more false positives showing up in the subnetworks. E.g., when the value for

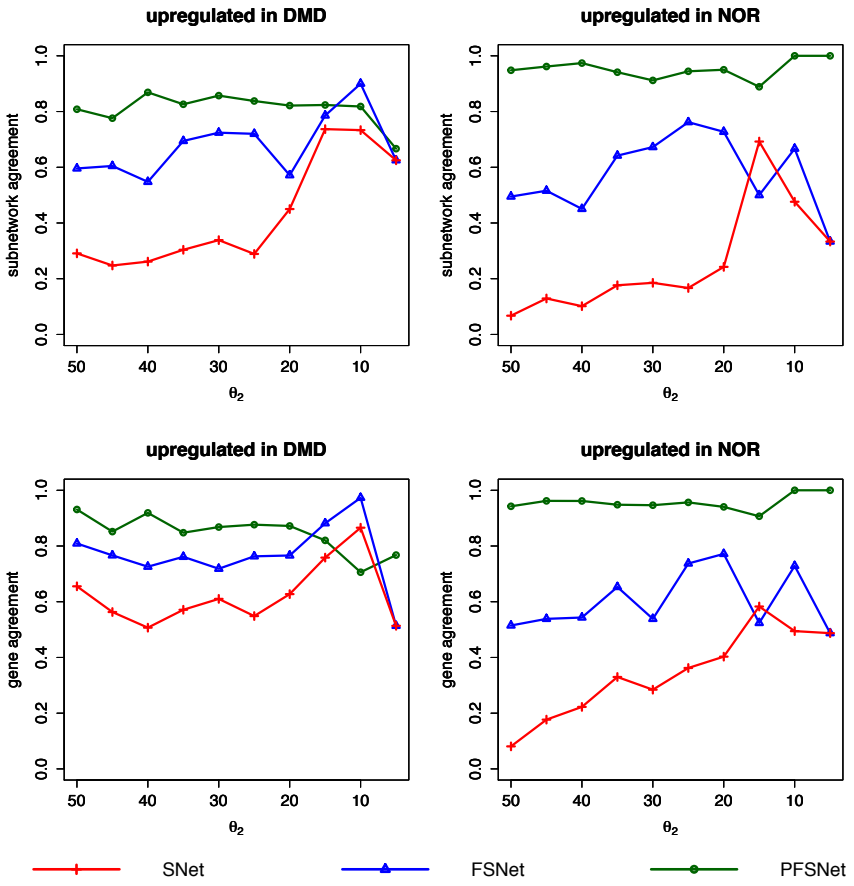


FIGURE 3.6: Consistency of predicted subnetworks in the DMD/NOR datasets (θ_1 = 5%).

θ_2 is set at the extreme ends (5% and 50%), the subnetworks have very low agreement between datasets in all 3 methods. In the Leukemia dataset, FSNet achieves the maximum subnetwork agreement of 100% (θ_2=20%) whereas SNet achieves the maximum subnetwork agreement of 77% (α=15%). In the DMD dataset, FSNet achieves maximum subnetwork agreement of 90% (θ_2=10%) whereas SNet achieves maximum subnetwork agreement of 73% (α=10%). In the ALL subtype dataset, FSNet achieves maximum subnetwork agreement of 38% whereas SNet only achieves 26%.

The results also show that giving genes that are not in the top 5% a partial vote is better than giving them a total vote. As we allow more and more genes to be considered, FSNet generally gives better subnetwork agreement than SNet. FSNet is thus more

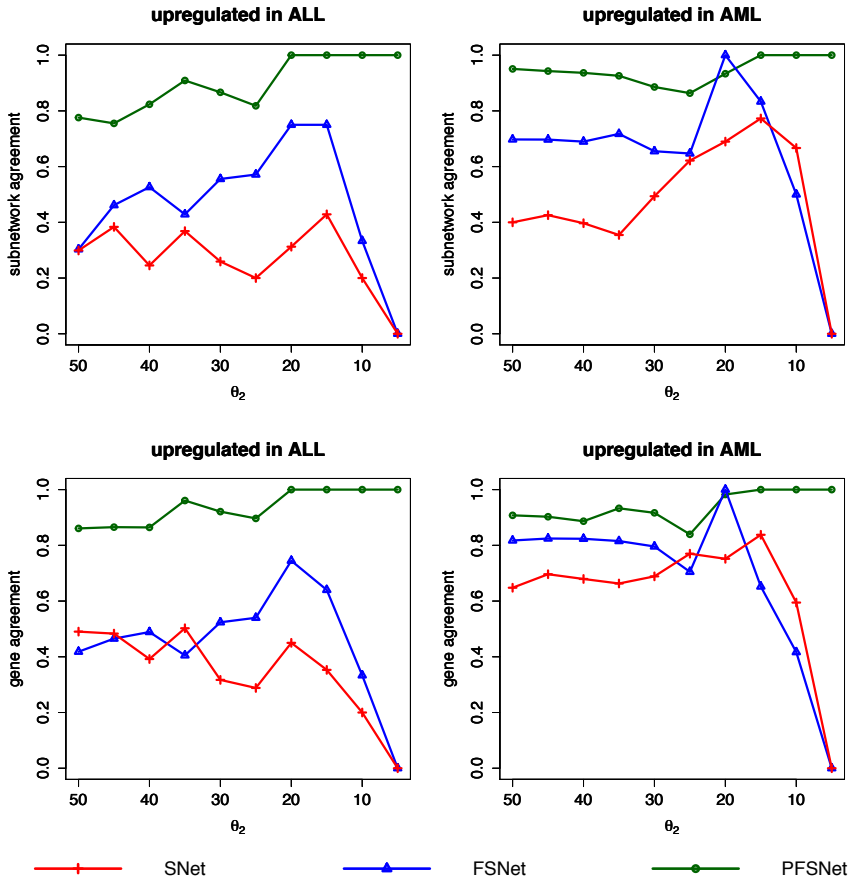


FIGURE 3.7: Consistency of predicted subnetworks in the ALL/AML datasets ($\theta_1 = 5\%$).

robust towards noise when incorporating more genes. E.g., when θ_2=α=50%, FSNet is able to get 69% subnetwork agreement but SNet only manages 40% in the Leukemia dataset. Similarly in DMD, FSNet achieves 59% whereas SNet achieves 29%.

In PFSNet, we get even higher subnetwork-level agreement than both FSNet and SNet. This shows the node scores obtained from the opposite phenotype play an important role in contributing towards consistent subnetworks. In particular, while both FSNet and SNet do not have very good subnetwork-level agreement in the ALL subtype dataset (38% and 25% respectively), PFSNet is able to achieve 74%.

We also measure the gene-level agreement to check whether the significant subnetworks contain similar genes in the two datasets. We see a similar trend that PFSNet performs

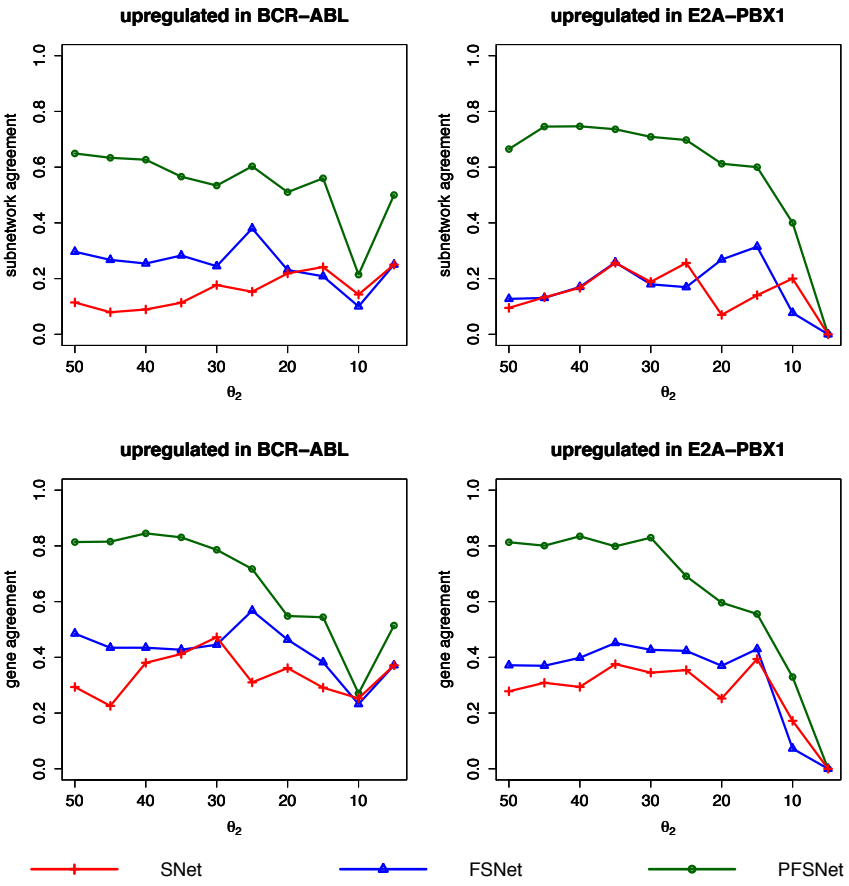


FIGURE 3.8: Consistency of predicted subnetworks in the BCR-ABL/E2A-PBX1 datasets ($\theta_1 = 5\%$).

better than FSNet which in turn performs better than SNet. In particular, for the ALL subtype dataset which has the worst pathway-level agreement reported above, the maximum gene-level agreement for PFSNet, FSNet and SNet are 84%, 57% and 47% respectively.

3.3.2 Comparing with GSEA, GGEA, SAM and t-test

We compare our methods with GSEA, GGEA, SAM and t-test. We run GSEA and GGEA on both datasets and measure the level of pathway agreement between the two datasets. In general, we achieve higher pathway-level agreement than GSEA and GGEA. PFSNet has a pathway-level agreement between 56%–100%, FSNet has a pathway-level

agreement between 38%–75%, GSEA has a pathway-level agreement between 12%–57% and GGEA has a pathway-level agreement between 18%–51%; cf. table 3.1.

We also measure the gene-level agreement from significant subnetworks between each pair of datasets. In order to compare this with GSEA, we computed the gene-level agreement from the "leading edge" gene sets in each pair of datasets. The "leading edge" genes are those genes that appear in GSEA's ranked list at the point at which the Kolmogorov-Smirnov running sum reaches its maximum deviation from zero (Subramanian et al., 2005). We also compare gene-level agreement with SAM and t-test which identifies individual differentially-expressed genes. PFSNet has a gene-level agreement between 54%–100%, FSNet has a gene-level agreement between 38%–88%, SNet has a gene-level agreement between 29%–76%. In contrast, GSEA, SAM and t-test have much worse agreement at the 5% significance level. GSEA has a gene-level agreement between 4%–44%, SAM has a gene-level agreement between 8%–50% and t-test has a gene-level agreement between 8%–41%; cf. table 3.2.

3.3.3 Comparing pathways and subnetworks

As pathways are often large, many analyses involving a whole pathway do not give consistent results. E.g., when we tested GSEA/GGEA in the previous subsection using pathways, the level of agreement was generally low.

One of the contributions in SNet, FSNet and PFSNet is the ability to break large pathways into smaller subnetworks. We select significant subnetworks from SNet, FSNet and PFSNet, and test them using GSEA. We discover that many of these subnetworks are also considered significant by GSEA/GGEA, even though GSEA/GGEA had earlier considered the original whole pathways from which these subnetworks were derived to be insignificant.

TABLE 3.1: Comparing pathway-level agreement of PFSNet, FSNet, GGEA and GSEA. (For PFSNet and FSNet, threshold values of $\theta_1 = 5\%, \theta_2 = 15\%$ are used.)

Dataset	PFSNet	FSNet	GSEA	GGEA
Leukemia	1.00	0.75	0.12	0.18
ALL (subtype)	0.56	0.38	0.34	0.37
DMD	0.82	0.79	0.57	0.51

TABLE 3.2: Comparing gene-level agreement of PFSNet, FSNet, SNet, GSEA, SAM, t-test. (For PFSNet and FSNet, threshold values of $\theta_1 = 5\%, \theta_2 = 15\%$ are used. D represents subnetworks enriched in phenotype D and $\neg D$ represents subnetworks enriched in phenotype $\neg D$. For GSEA, the "leading edge genes" were used. For SAM and t-test, we took genes at 5% significance level and also the top n genes indicated in brackets.)

Dataset	PFSNet		FSNet		SNet		GSEA		SAM(5% sig)		SAM(top 100)		t-test(5% sig)		t-test(top 100)	
	D	¬D	D	¬D	D	¬D	D	¬D	D	¬D	D	¬D	D	¬D	D	¬D
Leukemia	1.00	0.81	0.64	0.42	0.35	0.58	0.12	0.20	0.50	0.47	0.01	0.01	0.35	0.29	0.19	0.07
ALL (subtype)	0.54	0.70	0.38	0.41	0.29	0.57	0.04	0.04	0.19	0.27	0.12	0.21	0.08	0.10	0.01	0.00
DMD	0.82	0.72	0.88	0.75	0.76	0.54	0.44	0.20	0.34	0.08	0.27	0.19	0.41	0.19	0.11	0.25

TABLE 3.3: Testing subnetworks from PFSNet, FSNet and SNet using GSEA and GGEA.

	PFSNet	FSNet	SNet
Leukemia (GSEA)	0.50	0.00	0.00
Leukemia (GGEA)	0.67	0.50	0.50
ALL subtype (GSEA)	1.00	0.15	0.11
ALL subtype (GGEA)	1.00	0.47	0.35
DMD (GSEA)	0.90	0.57	0.50
DMD (GGEA)	0.54	0.71	0.45

We next test whether these subnetworks are consistently declared significant in two independent datasets by GSEA/GGEA; cf. table 3.3. Subnetworks taken from PFSNet give the highest agreement of about 100%, subnetworks taken from FSNet give the highest agreement of about 71% and the subnetworks taken from SNet give the highest agreement of about 50%. In contrast, using large pathways, GSEA and GGEA have an agreement of about 57% and 51% respectively.

3.3.4 Biologically-significant subnetworks

We also check the subnetworks consistently detected by PFSNet for biological relevance. We discover that many subnetworks and their genes are involved in relevant disease-related processes known in the literature. Some of these subnetworks predicted as significant in PFSNet are not discovered by SNet. We report these subnetworks ranked according to the p-value computed by PFSNet in table 3.4. We will describe three example subnetworks for the respective diseases to demonstrate their relevance to the diseases.

The cause of Duchenne muscular dystrophy is well known to stem from the gene Dystrophin, which codes for a protein attached to the cell membrane (sacrolemma) of striated muscle cells (Goldstein and McNally, 2010). When its expression is perturbed, the cell membrane becomes fragile and permits an amplification in calcium signals into the

muscle cell causing a cascade of signals to induce cell death. Our subnetwork is generated around the Dystrophin gene and implicates other genes belonging to the Myosin (MYBPC1,MYBPC2) and Troponin (TNNI1,TNNI2) family. The Myosin and Troponin genes are responsible for controlling muscle contractions. The down-regulation of Troponin in DMD patients might help explain muscle contracture, a condition in which the muscle shortens. This is because with lower abundance of Troponin, Myosin is able to bind to Actin. This mechanism together with the amplification of calcium causes the muscle to constantly contract, shortening over time (Goldstein and McNally, 2010, Krans, 2010).

For the Leukemia dataset (in which patients are either classified to have acute lymphoblastic leukemia or acute myeloid leukemia), one of the significant subnetworks that is biologically relevant is part of the Interleukin-4 signaling pathway. The binding of Interleukin-4 to its receptor (Cardoso et al., 2008) causes a cascade of protein activations involving JAK1 and STAT6 phosphorylation. STAT6 dimerizes upon activation and is transported to the nucleus and interacts with the RELA/NFKB1 transcription factors, known to promote the proliferation of T-cells (Rayet and Gelinas, 1999). In contrast, acute myeloid leukemia does not have genes in this subnetwork up-regulated and are known to be unrelated to lymphocytes.

For the ALL subtype dataset, the patients are categorized to either having the BCR-ABL oncogene or E2A-PBX1 oncogene. Antigen processing pathway is one of the significant subnetworks. This suggests that lymphocytes elicit different response in the two ALL subtypes. The immunophenotypic characteristics of acute leukemias have been described in the literature (Giunta and Pucillo, 2012, Hruak and Porwit-MacDonald, 2002). ALL belonging to the BCR-ABL subtype express the cluster of differentiation (CD) markers CD9 and CD10, whereas those belonging to the E2A-PBX1 subtype express the CD19 and CD45 markers.

TABLE 3.4: Top 5 subnetworks that have biological significance. (* indicates subnetworks that were not found in SNet and # indicates pathways that were missed by GSEA)

Leukemia	ALL subtype	DMD
Proteasome Degradation	Wnt Signaling*#	Striated Muscle Contraction*#
IL-4 Signaling*#	Antigen Processing	Integrin Signaling
Antigen Processing*	Jak-STAT Signaling*#	VEGF Signaling*
B-Cell Receptor Signaling#	T-Cell Receptor Signaling	Tight Junction
Wnt Signaling*#	Adherens Junction*#	Actin Cytoskeleton Signaling

3.4 Discussion

Methods for analysing microarray data that focus on identifying biological processes and pathways are superior to the traditional method of testing individual genes for two main reasons. Firstly, gene sets represented by pathways make discovery more interpretable. Secondly, they have the ability to identify gene sets whose members might have only slight changes in individual gene-expression values.

We have shown in this chapter that analysis of subnetworks provides even better biological interpretability than whole pathways, which become too generalised when they are large. Moreover, some subnetworks that were detected as significant were originally missed when the whole large pathways were tested. We have verified that SNet, a method that analyzes subnetworks, has the ability to produce more consistent results than other methods surveyed. However, SNet is not very robust when different thresholds are used, this is because a too-relaxed threshold will include some non-relevant genes and a too-stringent threshold will exclude some relevant genes.

We have introduced two improvements to SNet in this chapter: by incorporating the fuzzification technique (FSNet), and by computing paired t-statistic based on the fuzzy score of two phenotypes (PFSNet). We have found that subnetworks identified by FSNet

and PFSNet show higher consistency across independently-obtained datasets than other methods.

Chapter 4

ESSNet: Handling datasets with extremely-small sample size

4.1 Background

In microarray analysis, one is often faced with the problem of obtaining the right sample size to draw meaningful conclusions from the data. It is possible to conduct computations and simulations to estimate the sample size required upstream of a laboratory's analysis pipeline. Many methods that look into this issue examine the relationship between different statistical variables and their relationship with sample size. For instance, the t-statistic described in chapter 2 is a function of two population means divided by their standard error. The standard error is in turn a function of the sample variance and sample size. Such methods estimate sample size by predefining power and type-I error, the t-distribution can then be used to estimate the minimum sample size required.

A large sample size is usually required to maintain high statistical power and low family-wise error rate or false-discovery rate. For example, one such model (Hart et al., 2013)

requires more than 100 samples to achieve power at 0.9 and false-discovery rate at 0.05; see figure 4.1.

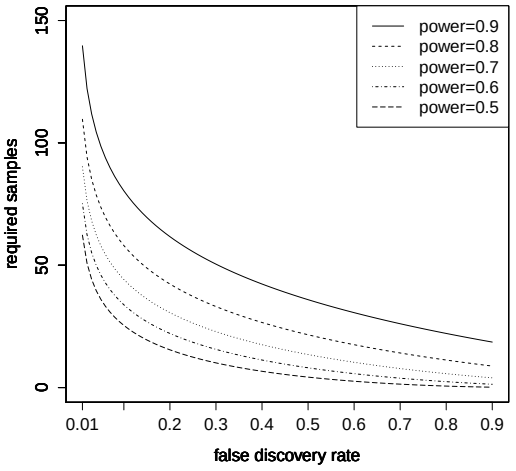


FIGURE 4.1: A model estimating require sample size for a specified power and false-discovery rate (Hart et al., 2013).

However, many laboratories do not start with an upstream pipeline of determining sample sizes. And, inevitably, some are also constrained by budget, biology, and other factors to conduct studies with small sample sizes ($N < 5$). Dealing with data of small sample size presents some complications:

1. As mentioned earlier, a gene-wise t-test between the two phenotype groups will sacrifice power and type-I error in data of small sample size.

2. Many other gene-set methods that compute a p-value based on permutation test cannot do so reliably because there are not enough samples to do class-label swapping, resulting in poor granularity of the p-value (see chapter 2).

In addition, many existing methods compute differentially-expressed genes (DEGs) as part of their framework. For example:

1. ORA examines the overlapping proportions of DEGs within a pathway.

2. In GSEA, a Kolmogorov-Smirnov-like statistic is computed from a ranked list of DEGs within a pathway.

3. DEAP has a recursive function that sums up differential expression.

Most scores of differential expression are based on fold-change and p-value from a t-test. We examine the effect of sample size on computing DEGs on many different datasets and find that genes ranked by fold-change or t-test p-values have very large variances. For example, in figs. 4.2 to 4.4, the ranks of p-values, log fold changes and gene-expression level are recorded, for every sample size (N) considered ranging from 2 to 10. This process is repeated over 100 times and the standard deviations of the respective ranks are measured. P-values and fold-change are more sensitive to sample size variation, as this is evident in the larger standard deviation of the ranks of p-values and fold-change. In contrast, gene ranking based on expression level has a much smaller standard deviation and is less sensitive to sample size variation.

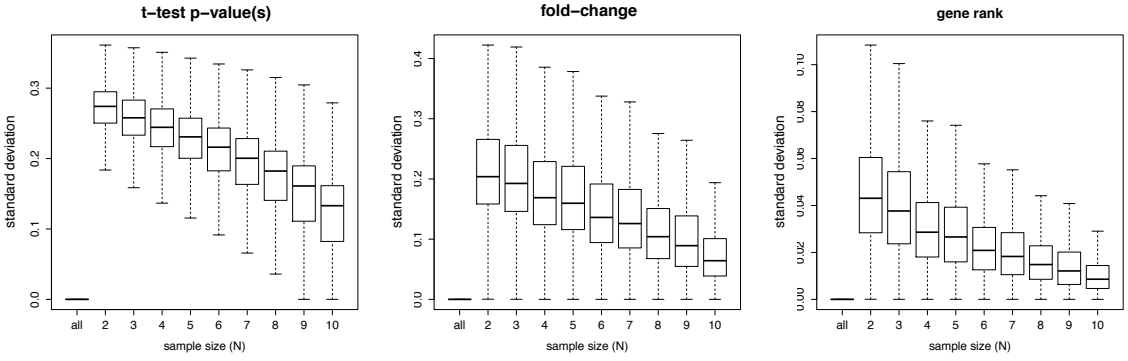


FIGURE 4.2: Effects of sample size on the ranks of differentially-expressed genes in DMD/NOR dataset.

This questions the validity of existing methods in small-sample-size situations ($N < 5$). Although in our previous chapter, PFSNet works well in moderate- and large-sample-size situations, its performance degrades as we consider sub-samples from the original sample size.

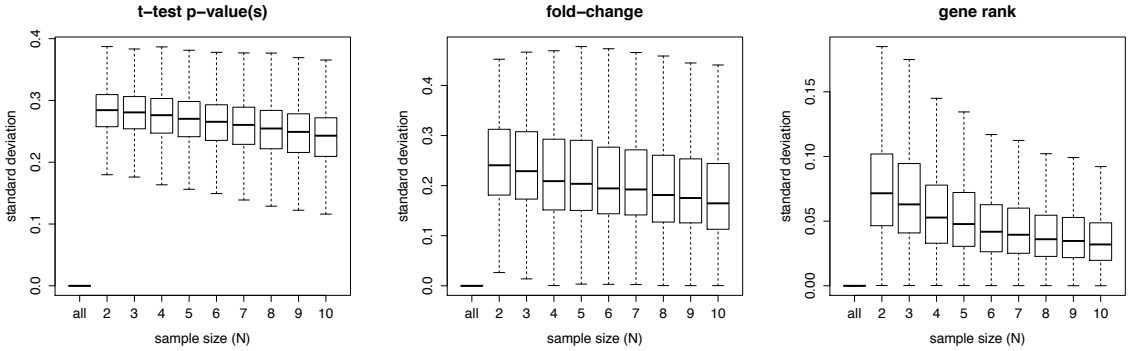


FIGURE 4.3: Effects of sample size on the ranks of differentially-expressed genes in ALL/AML dataset.

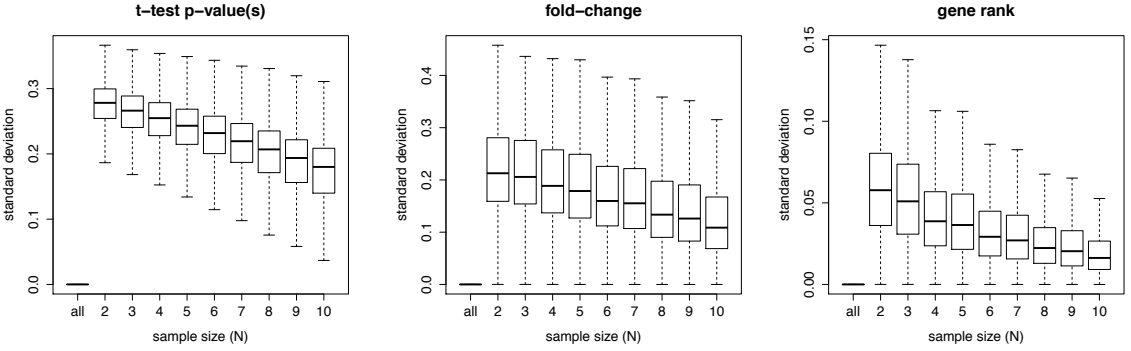


FIGURE 4.4: Effects of sample size on the ranks of differentially-expressed genes in BCR-ABL/E2A-PBX1 dataset.

In this chapter, we formulate a method, ESSNet, that is able to detect disease-relevant subnetworks even in datasets of small sample size. We then study the effects of small sample size, comparing ESSNet with other methods.

4.2 Method

4.2.1 Subnetwork generation

For each sample of phenotype D, we rank the genes by their expression values. Let $r(g_i, p_j)$ be the rank of gene i in patient j. We tested and found that gene ranks do not fluctuate as much due to sample-size variation as fold-change or p-values from t-test; see

figs. 4.2 to 4.4. Each gene is then given a rank based on the average among the samples of phenotype D:

$$rank_D(g_i) = \sum_{j \in D} \frac{r(g_i, p_j)}{|D|} \tag{4.1}$$

where $|D|$ is the number of samples belonging to the phenotype D.

We obtain a gene list extracted from the top $\alpha\%$ of the gene ranks computed in equation 4.1. We chose $\alpha = 10$ in our experiments. Genes not in this list are removed from every pathway, thus fragmenting each pathway into smaller connected components (i.e., the subnetworks). We only consider subnetworks that are of size at least 5. The subnetworks for phenotype $\neg D$ are generated analogously.

4.2.2 Subnetwork testing

4.2.2.1 Scoring

The subnetwork scores in SNet (Soh et al., 2011) and PFSNet (Lim and Wong, 2014) are assigned to patients in each phenotype D and $\neg D$. For example, a disease-related subnetwork might have a mean value of 15 in phenotype D and 51 in $\neg D$. In datasets with large sample sizes, class-label permutations are used to test whether this difference is significant. In datasets with extremely small sample size, this is not possible, and one has to resort to the theoretical t-distribution for this test. But, in this situation, a small change in sample size can produce dramatically different outcomes. For example, a t-test between two groups with two samples each—say 10, 20 and 50, 52—has a p-value of 0.077, whereas with a few more samples 9,10,20,21 and 49,50,52,53, the p-value drops to 0.0008.

Our subnetwork score is based on a novel idea. We postulate that, when a subnetwork is irrelevant to the distinction between phenotypes D and $\neg D$, the difference of the expression values of any gene in this subnetwork in any pair of samples of D and $\neg D$ should be very small. Suppose there are k genes in a subnetwork, m patients in phenotype D and n patients in phenotype $\neg D$. Then there are $m * n$ possible pairs of differences for each of the k genes. According to the postulate, if the subnetwork is irrelevant, these $M = k * m * n$ paired differences should be distributed around 0. Thus we propose to evaluate these paired differences using a paired t-test.

Let $\delta(g_i, p_j, p'_l) = e(g_i, p_j) - e(g_i, p'_l)$ be the distribution of paired expression differences for each p_j in D, p'_l in $\neg D$ and g_i in subnetwork S, where $e(a, b)$ represents the expression value of gene a in patient b. Then the t-statistic computed for the subnetwork S is:

$$T_S = \frac{\mu_\delta}{sd_\delta / \sqrt{M}} \tag{4.2}$$

where μ_δ and sd_δ are the average and standard deviation over all paired differences of genes in the subnetwork S respectively and $M = k * m * n$.

Returning to our example of using two samples per group, although we have only 2 samples per phenotype, we can have up to $4 * k$ paired differences if k is the size of the subnetwork.

It is also possible to define a similar distribution based on the rank differences of the genes in subnetwork S; i.e. $\delta'(g_i, p_j, p'_l) = r(g_i, p_j) - r(g_i, p'_l)$ where r represents the rank function defined in equation 4.1. The t-statistic computed for the paired rank difference is analogously defined to that of the paired expression differences in equation 4.2.

The choice of statistical test depends on the assumptions that govern the dataset. The t-test is a parametric test that assumes data normality. If the number of samples and the

size of subnetworks are very small, it is hard to verify this normality assumption. Hence it might be necessary to consider an alternative test statistic that does not assume data normality. In our software, we provide the option of specifying a Wilcoxon-like test. However, in our experiments, the two tests actually produce very similar results.

The Wilcoxon-like test statistic for subnetwork S is computed by

$$W_S = \left| \sum_{i=1}^{k} \sum_{j=1}^{n} \sum_{l=1}^{m} sign(\delta(g_i, p_j, p'_l)).R_{i,j,l} \right| \tag{4.3}$$

where sign is a function that maps positive numbers to 1 and negative numbers to -1 and $R_{i,j,l}$ is the rank of the absolute value of the differences over all genes in the subnetwork S in increasing order.

The Wilcoxon-like test has a similar null hypothesis to the t-test without restricting to a normal distribution and tests if the differences have a median 0.

4.2.2.2 Estimating the null distribution

Our conjecture that $\delta(g_i, p_j, p'_l)$ is a distribution around 0 can be tested on a null distribution. There are two traditional ways that estimates this null distribution, in which randomized columns or rows of the expression matrix is used to re-compute the statistic over a number of iterations.

The first way assumes the null hypothesis that the subnetwork being tested is irrelevant to distinguishing the two phenotypes. Thus the gene-expression profiles of any pair of patients from the two phenotypes are exchangeable for computing points in the null distribution. In other words, class labels are randomly swapped to create new data inputs from which the null distribution is formed. This method is used by GSEA to

evaluate the significance of the Kolmogorov-Smirnov-like statistic of the pathway when sample size is sufficiently large. This method preserves the full gene-gene correlations, as well as gene-expression values, in each patient. However, when sample size is small there are limited ways in which the class labels can be permuted, resulting in a sparse null distribution. This greatly affects the reliability and the graularity of the p-values.

The second way postulates that any two gene-expression values within the same patient are exchangeable to compute the null distribution. This method creates new data inputs by randomly re-labeling genes. This method is used by GSEA to evaluate the significance of the Kolmogorov-Smirnov-like statistic of the pathway when the dataset has a small sample size, since a sizeable null distribution can be generated this way. However, this postulate is based on the assumption that the genes' expression are independent of each other, ignoring the correlation between genes. In other words, this method actually tests if the genes in the pathway behave no differently from a random set of genes. But the genes in any pathway are coordinated by nature, whereas a random set of genes is not. Hence this null hypothesis is not appropriate. So it has a tendency of being rejected, producing false positives.

We rely instead on a third way to produce the null distribution for our test. It postulates that randomized gene-expression profiles that preserve the gene-gene correlation structure in the original dataset are exchangeable with it. This postulate is consistent with the assumption that genes in any pathway are coordinated and gene expressions are governed by biological pathways. Due to exchangeability that follows from the postulate, it is statistically sound to use correlation-preserving randomized gene-expression profiles to obtain a null distribution of the test statistic. As mentioned in chapter 2, array rotation (Dorum et al., 2009) is one of the known techniques for producing a large number of these correlation-preserving randomized gene-expression profiles. We use this technique to produce statistically-valid p-values for our test statistic.

4.2.3 Weighted differences

The performance of our method might be affected by the selection of hard thresholds ($\alpha\%$), since allowing more genes to be considered in the subnetwork might increase the number of spurious genes selected within each subnetwork whereas using a very conservative threshold might select very small subnetworks. In order to overcome this issue, we weigh the differences based on the gene ranks computed in equation 4.1. We use two thresholds θ_1 and θ_2 ($\theta_1 < \theta_2$) on the gene ranks, genes with ranks above θ_1 are given a weight of one and genes with ranks between θ_1 and θ_2 are given weights between zero and one by linear interpolation. Genes with ranks below θ_2 have are given weights zero since they are not used to induce the subnetworks. (In our experiments, $\theta_1 = 10\%$ and $\theta_2 = 20\%$.)

For each difference δ of the subnetwork computed in phenotype D, we adjust δ to be:

$$\delta''(g_i, p_j, p_l') = w(rank_D(g_i)) * \delta(g_i, p_j, p_l'). \tag{4.4}$$

where w is a function mapping the gene ranks to a weight between zero and one as explained above. The difference for the subnetwork computed in phenotype $\neg D$ is computed analogously.

4.3 Results

We randomly partition the two independent datasets into subsets of smaller sample sizes ranging from 2 to 10 from each phenotype. In order to observe the effect of sample size on various methods, we compare the subnetwork overlap of the corresponding methods with varying sample sizes.

For every sample size (N) considered, we partition the datasets accordingly and use the subnetwork-generation procedure mentioned in section 4.2.1 to generate the subnetworks in one dataset. We then test these subnetworks for statistical significance, under a significance threshold of 5% using the procedure mentioned in sections 4.2.2– 4.2.3 on the two datasets independently. The subnetwork overlap is a Jaccard-like agreement, defined as follows: Let the two sets of significant subnetworks identified by dataset 1 and dataset 2 using N samples be SN_1^N and SN_2^N respectively. Then the subnetwork-level agreement is defined as

$$\frac{|SN_1^N \cap SN_2^N|}{|SN_1^N \cup SN_2^N|}. \tag{4.5}$$

There are many ways to partition a dataset of M samples into subsets of N samples. For our experiments, we test the procedure many times and report the average subnetwork-level agreement.

Since ORA and GSEA identify whole pathways instead of subnetworks, in testing these methods, we measure the pathway-level agreement which is defined analogously.

We also measure the overlap in genes between the predicted subnetworks, which is defined analogously below, where $Genes_i^N$ denotes the set of genes in SN_i^N:

$$\frac{|Genes_1^N \cap Genes_2^N|}{|Genes_1^N \cup Genes_2^N|}. \tag{4.6}$$

4.3.1 Comparing subnetwork- and gene-level overlap

We compare the subnetwork-level agreement of our method, ESSNet-unweighted, with other gene-set methods (ORA-hypergeo, ORA-paired, GSEA, NEA-paired, DEAP, and PFSNet); see figs. 4.5 to 4.7.

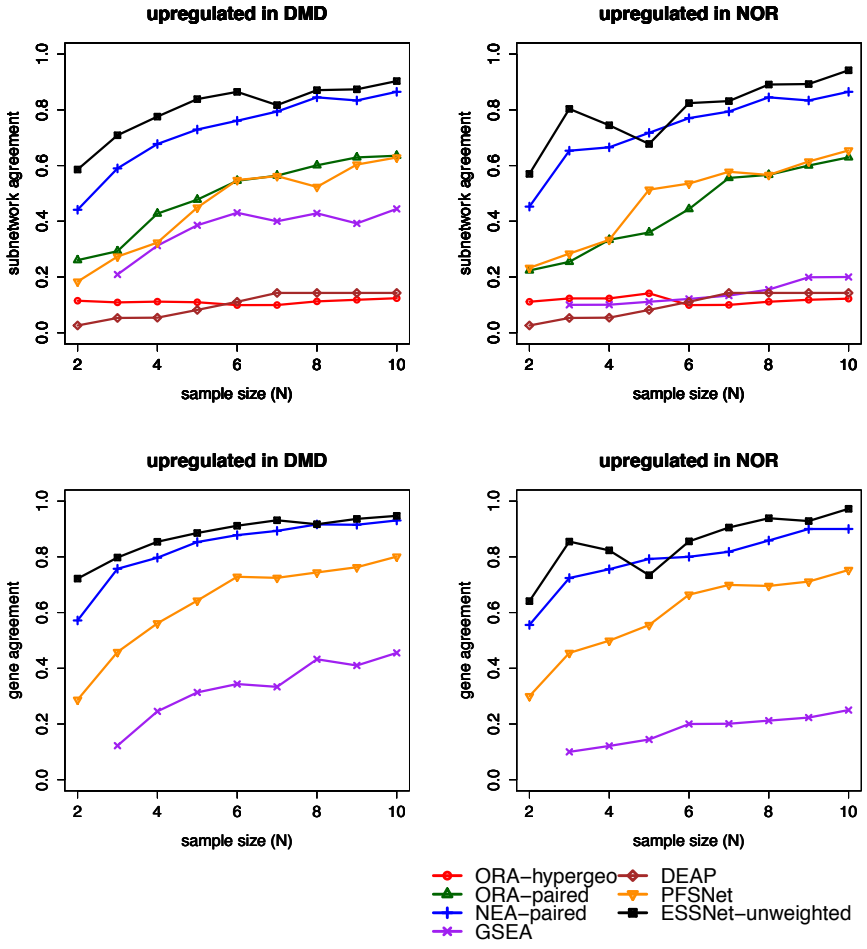


FIGURE 4.5: Consistency of subnetworks and their genes in DMD/NOR dataset using partitions of smaller sample sizes ranging from 2 to 10 from each phenotype.

ORA-hypergeo is the usual overlap-analysis method. It tests whether a pathway is significant by intersecting the genes in the pathway with a pre-determined list of differentially expressed genes (here, we use all genes whose t-statistic meets the 5% significance threshold), and checking the significance of the size of the intersection using the hypergeometric test. ORA-paired is a modification of ORA-hypergeo; it does not use a pre-determined list of differentially expressed genes and the hypergeometric test. Instead, it applies the t-test described in section 4.2.2.1 using all the genes in the pathway. GSEA is a direct-group method based on the Kolmogorov-Smirnov test as described in section 4.2.2.1. As mentioned earlier, the gene-permutation option is used to evaluate

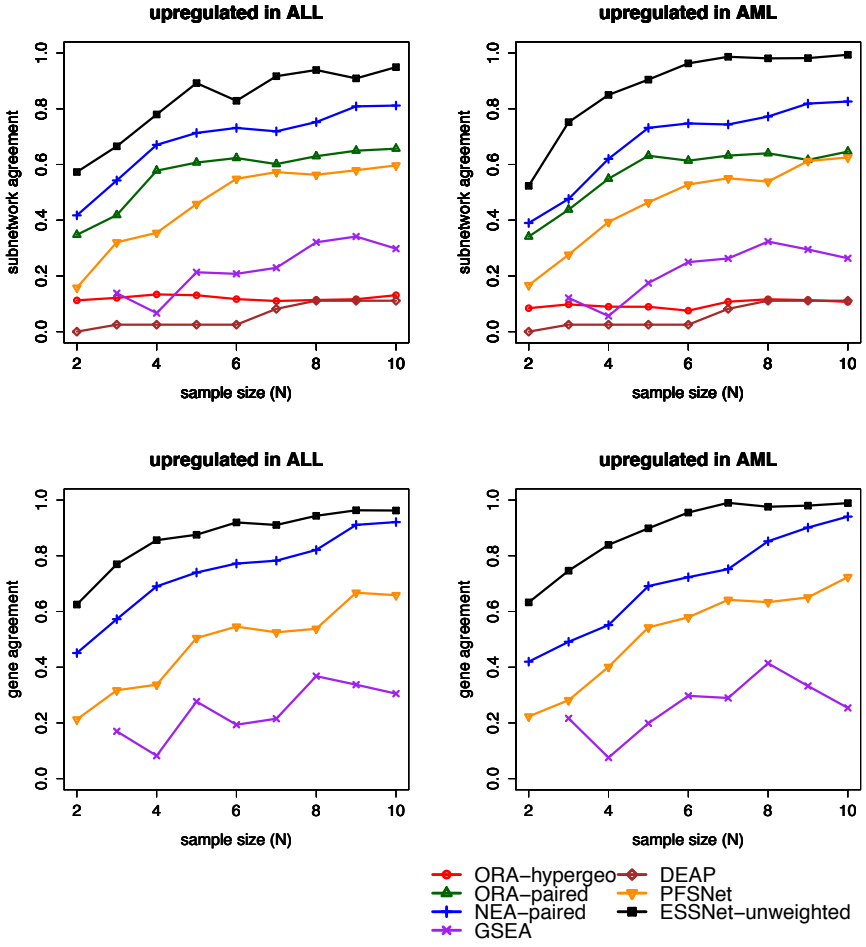


FIGURE 4.6: Consistency of subnetworks and their genes in ALL/AML dataset using partitions of smaller sample sizes ranging from 2 to 10 from each phenotype.

significance. NEA-paired is a network-based method where each gene and its immediate neighbourhood in a pathway form a subnetwork. The subnetworks are subjected to the t-test discussed in section 4.2.2. PFSNet is a network-based method as previously discussed in chapter 3. ESSNet-unweighted generates subnetworks based on the method discussed in section 4.2.1 and tests each subnetwork for statistical significance using the scores computed from section 4.2.2.

ORA-hypergeo has very low pathway-level overlap even when sample size is 10. There are three weaknesses that contribute to its poor performance. Firstly, it amounts to testing whether the entire pathway is significantly differentially expressed. If only a branch

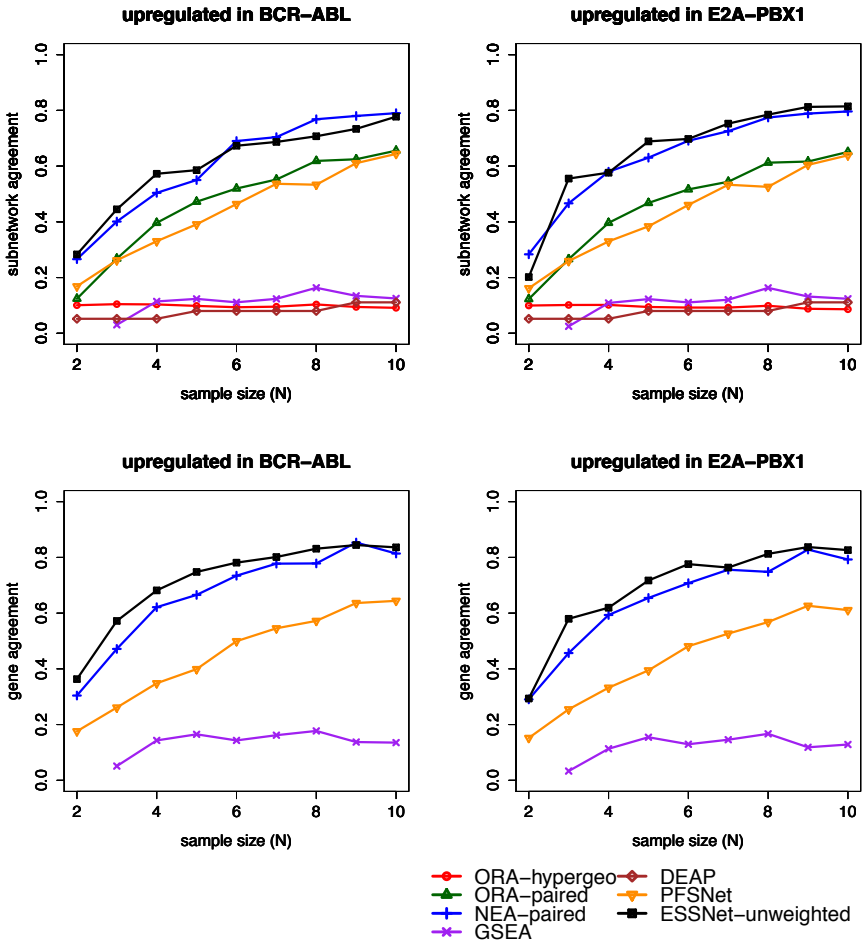


FIGURE 4.7: Consistency of subnetworks and their genes in BCR-ABL/E2A-PBX1 dataset using partitions of smaller sample sizes ranging from 2 to 10 from each phenotype.

of a large pathway is relevant to the phenotypes, the noise from the large irrelevant part of the pathway can mask the signal from that branch. Secondly, it relies on a pre-determined list of differentially-expressed genes. This list is sensitive to the choice of threshold that defines which genes to be considered as differentially expressed. And, irrespective of the threshold, as shown in figs. 4.2 to 4.4, this list lacks consistency when sample size is small. Thirdly, its use of the hypergeometric test corresponds to the null hypothesis that genes in the pathway behave no differently from random sets of genes of the same size as the pathway. As genes in a pathway are generally coordinated in their behaviour to perform the specific function associated with the pathway, this null

hypothesis is generally false. Thus this hypergeometric test tends to reject the null hypothesis.

ORA-paired circumvents the second weakness of ORA-hypergeo since it does not need any list of differentially-expressed genes. It also eliminates the third weakness of ORA-hypergeo since it uses a biologically-more-plausible null hypothesis that genes in the pathway have similar expression values between the two phenotypes if the pathway is irrelevant to the difference of the two phenotypes. Therefore, ORA-paired performs much better than ORA-hypergeo. The subnetwork-level agreement increases to as high as 65% verses 13% in ORA-hypergeo when $N = 10$ and 34% verses 11% when $N = 2$. This suggests that the paired-difference t-test is a strategy that works extremely well in a small-sample-size situation.

A disease could be the result of the dysfunction of a small part of a large pathway. In this situation, most of the genes in this large pathway may not be differentially expressed. Even though ORA-paired has improved on ORA-hypergeo, it is still unlikely to find this large pathway significant. That is, ORA-paired retains the first weakness of ORA-hypergreo. Thus, it makes sense to directly extract subnetworks from pathways and test these subnetworks individually for significance.

We apply the NEA idea (Sivachenko et al., 2007) to generate candidate subnetworks from a pathway, after which we apply ORA-paired to determine the significant ones. This NEA-paired approach circumvents all three weaknesses of ORA-hypergeo. Hence it performs even better than ORA-paired. The subnetwork-level agreement increases to as high as 85% when $N = 10$ and 43% when $N = 2$. This suggests that the subnetwork-generation procedure increases the sensitivity of the paired-difference t-test. We believe this is because paired differences around the neighborhood of selected genes enable the

test to correctly reject subnetworks that have no differentially expressed genes within them.

GSEA also eliminates the second weakness of ORA-hypergeo because it does not need any list of pre-determined differentially expressed genes. GSEA's Kolmogorov-Smirnov statistic is based on the rank of the t-statistic values of the genes in the pathway. However, GSEA retains the first weakness of ORA-hypergeo. And, when the gene-permutation option is used to determine the significance of the Kolmogorov-Smirnov statistic, as in this chapter, it also retains the third weakness of ORA-hypergeo. Therefore, while it outperforms ORA-hypergeo, it is inferior to ORA-paired and NEA-paired. GSEA achieves a maximum pathway-level overlap of 45% when N is 10 and 27% when $N = 3$. We are unable to evaluate GSEA when $N = 2$ because it requires a minimum of 3 samples.

DEAP examines all possible maximal linear paths in the pathway and chooses the path with maximum absolute differential expression score. The score given for a path is recursively computed based on the catalytic or inhibitory edges taken as positive and negative summands respectively (Haynes et al., 2013). DEAP partially eliminates the first weakness of ORA-hypergeo because it breaks the pathway into maximal linear paths, but it does not consider non-maximal subpaths. It also shares the second weakness of ORA-hypergeo because it computes a score for each path based on differential expression which is unstable when sample size is small. As a result, DEAP has poor performance. DEAP achieves a maximum pathway-level overlap of 28% when N is 10 and 6% when N is 2.

PFSNet does not need any list of pre-determined differentially-expressed genes, eliminating the second weakness of ORA-hypergeo. It generates subnetworks, and so eliminates the first weakness of ORA-hypergeo. For each subnetwork and each sample, it computes

a pair of scores for that sample based on phenotype D data and phenotype $\neg D$ data respectively. It postulates very reasonably that, if the subnetwork is irrelevant to the difference between D and $\neg D$, these pairs of scores should be distributed around 0. It then uses class-label permutations to evaluate this null hypothesis. Thus PFSNet also eliminates the third weakness of ORA-hypergeo. However, when sample size is small, the null distribution cannot be properly produced using class-label permutations. Thus PFSNet has good performance when N is reasonably high but inferior performance when N is small. PFSNet achieves a maximum overlap of 65% when $N = 10$ and 21% when $N = 2$.

Finally, we apply the same set of tests to ESSNet-unweighted, which selects subnetworks as described in section 4.2.1 and tests these subnetworks for significance using the scores computed in section 4.2.2. Clearly, ESSNet-unweighted also eliminates all three weaknesses of ORA-hypergeo in a manner analogous to NEA-paired. It has excellent performance, superior to all other methods studied here. We get much higher subnetwork overlap of up to 99% when $N = 10$ and 58% when $N = 2$. We believe ESSNet-unweighted performs better than other methods because of the following additional reasons.

Even though NEA-paired performs well, each of its subnetwork is based on a seed gene and its immediate neighbouring genes in that pathway, regardless of whether those neighbouring genes are themselves differentially or highly expressed. This can potentially cause a loss in signal, especially when the seed gene has a large number of immediate neighbours. Moreover, such a subnetwork cannot capture a long causal chain of genes. These two issues are rectified in ESSNet-unweighted which forms a subnetwork in a pathway based on a connected component comprising entirely of high-ranking genes and, as shown earlier in figs. 4.2 to 4.4, relying on gene ranking is more robust to sample-size variation.

TABLE 4.1: Precision and recall of ESSNet-unweighted.

		Precision						Recall					
		DMD		ALL		BCR		DMD		ALL		BCR	
		D	¬D	D	¬D	D	¬D	D	¬D	D	¬D	D	¬D
sample size (N)	2	0.96	0.88	0.87	0.95	0.93	0.91	0.45	0.31	0.34	0.25	0.19	0.17
	3	0.93	0.86	0.99	0.89	0.90	0.87	0.56	0.45	0.56	0.41	0.21	0.16
	4	0.88	0.88	0.97	0.92	0.91	0.87	0.67	0.50	0.51	0.53	0.35	0.48
	5	0.89	0.88	0.94	0.90	0.89	0.90	0.73	0.52	0.74	0.55	0.36	0.38
	6	0.82	0.88	0.93	0.92	0.89	0.91	0.78	0.62	0.74	0.62	0.44	0.438
	7	0.85	0.86	0.95	0.93	0.90	0.87	0.75	0.59	0.66	0.64	0.55	0.53
	8	0.84	0.89	0.97	0.94	0.90	0.92	0.81	0.69	0.74	0.66	0.61	0.66
	9	0.88	0.90	0.94	0.92	0.89	0.89	0.90	0.67	0.76	0.74	0.65	0.67
	10	0.88	0.93	0.97	0.92	0.90	0.90	0.86	0.84	0.89	0.74	0.66	0.73

4.3.2 Precision and recall

As shown later in section 4.3.5, ESSNet-unweighted attains very high subnetwork overlap when the sample size is large. So, it is possible to define a set of gold-standard subnetworks as follows, to estimate the false-positive and false-negative subnetworks induced by small samples:

$$G = SN_1^{all} \cap SN_2^{all} \tag{4.7}$$

where SN_i^{all} is the set of significant subnetworks produced by ESSNet-unweighted based on the entire dataset i.

The precision and recall are defined respectively as:

$$precision = \frac{|SN^N \cap G|}{|SN^N|}, \quad recall = \frac{|SN^N \cap G|}{|G|} \tag{4.8}$$

where SN^N is the set of significant subnetworks produced by ESSNet-unweighted using an N-sample subset of one entire dataset.

It is surprising that the precision does not drop much even when smaller sample sizes are considered. For example, we get a precision of about 88%, 87% and 91% even when $N = 2$ in the DMD, Leukemia and ALL-Subtype datasets respectively. On the other hand, the maximum recall when $N = 2$ is about 50% in the DMD dataset. Cf. 4.1.

Thus, more bona fide subnetworks are missed from the predictions when N is very small, while few false positives are produced. This is reasonable as a small sample may not have captured all the causes underlying a phenotype.

4.3.3 Comparing expression-difference, rank-difference t-test and Wilcoxon-like test

We also want to test whether there is any difference between using a parametric t-test and a non-parametric Wilcoxon test in extremely-small-sample-size situations. We evaluate the t-test using the distribution of expression-difference and rank-difference defined by $\delta(g_i, p_j, p'_l)$ and $\delta'(g_i, p_j, p'_l)$ respectively in section 4.2.2.1.

Our results show that there is little difference in the subnetwork-level overlap computed using these tests; see figs. 4.8 to 4.10

4.3.4 Comparing unweighted and weighted ESSNet

The weighted version of ESSNet modifies each pairwise difference for each gene by a function of the gene's average ranking amongst the class of samples in which the subnetwork was derived.

Our results show that subnetwork-level agreement improves when ESSNet-weighted is used; see figs. 4.11 to 4.13. For example, when the top 20% genes are considered (but with genes in the top 10-20% given lesser weight), ESSNet-unweighted and ESSNet-weighted

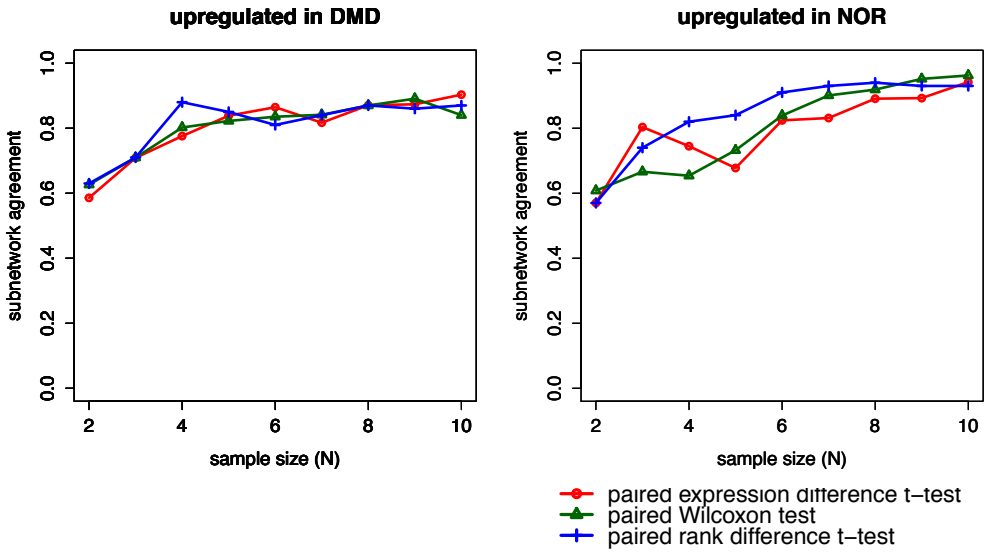


FIGURE 4.8: Consistency of subnetworks in ESSNet between t-test and wilcoxon test in DMD/NOR dataset.

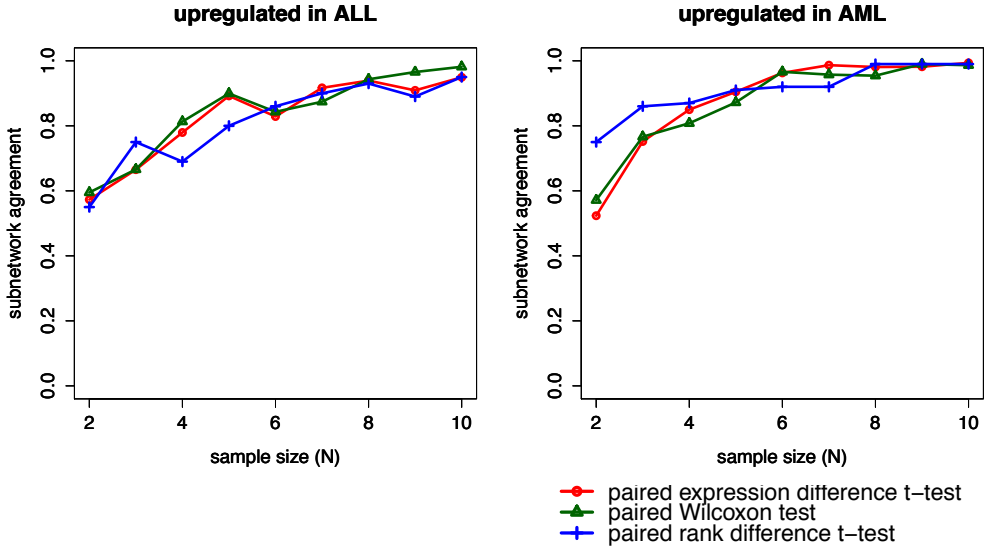


FIGURE 4.9: Consistency of subnetworks in ESSNet between t-test and wilcoxon test in ALL/AML dataset.

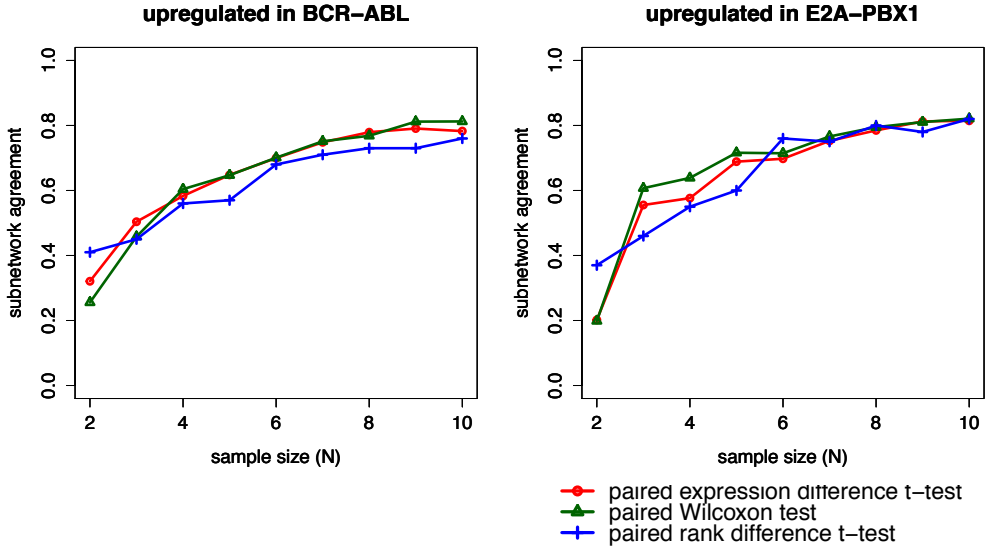


FIGURE 4.10: Consistency of subnetworks in ESSNet between t-test and wilcoxon test in BCR-ABL/E2A-PBX1 dataset.

has a maximum subnetwork overlap of 56% and 62% respectively when $N = 2$. By down-weighting spurious genes towards zero, spurious subnetworks are rejected thereby increasing the subnetwork-level overlap.

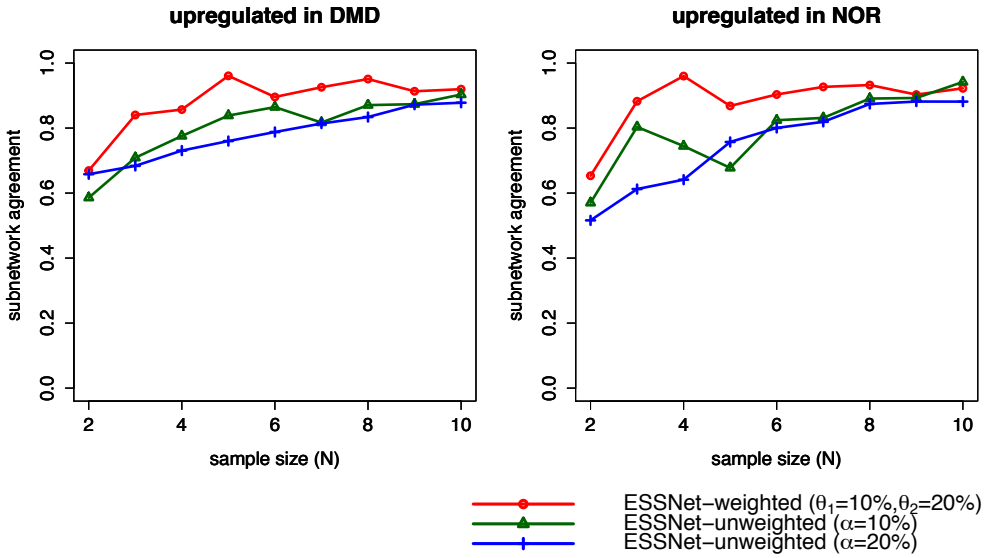


FIGURE 4.11: Consistency of subnetworks between weighted and unweighted ESSNet in DMD/NOR dataset.

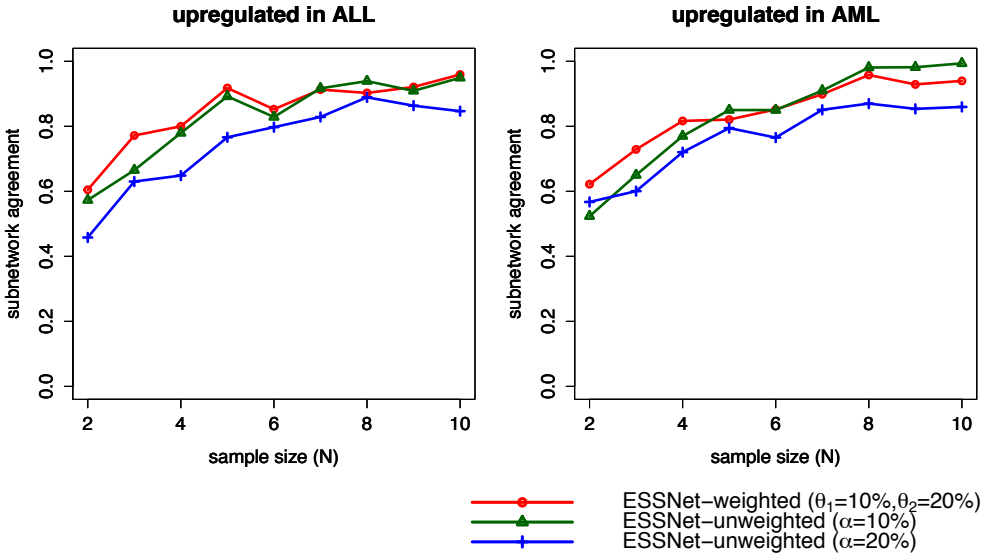


FIGURE 4.12: Consistency of subnetworks between weighted and unweighted ESSNet in ALL/AML dataset.

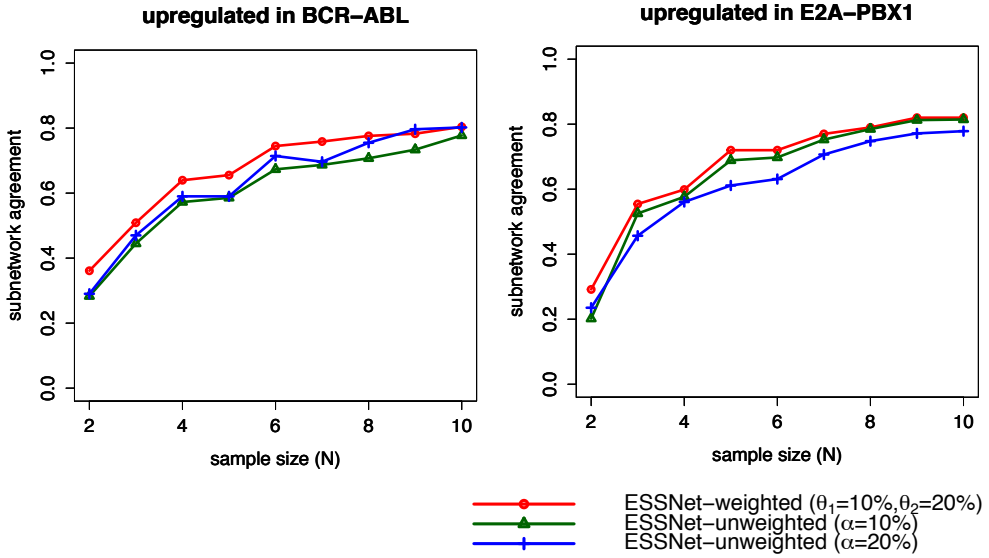


FIGURE 4.13: Consistency of subnetworks between weighted and unweighted ESSNet in BCR-ABL/E2A-PBX1 dataset.

4.3.5 Comparing different null-distribution-generation methods in large-sample-size data

The rotation test is a viable alternative for generating the null distribution in datasets that have very small sample sizes, since it preserves gene-gene correlations. When sample size is large, it is possible to generate the null distribution using class-label swapping. We compare these two different methods of generating the null distribution in large-sample-size datasets because class-label permutation imposes less assumptions on the data and is a good baseline to compare against. In our experiments, we find that rotation test and class-label swapping produce similar results. This lends us confidence on the results obtained from the rotation tests in datasets with smaller sample size. ESSNet achieves very good subnetwork-level agreement when sample size is small, and it continues to be superior when sample size is large; cf. tables 4.2 and 4.3.

TABLE 4.2: Average number of subnetworks predicted by ESSNet over the sample sizes (N); the first number denotes the number of subnetworks in the numerator of the subnetwork-level agreement and the second number denotes the number of subnetworks in the denominator of the subnetwork-level agreement; cf. equation 4.5.

		DMD	ALL	BCR
sample size (N)	2	8.2/13.4	7.0/11.9	4.8/12.6
	3	11.1/15.9	11.3/17.9	5.0/11.7
	4	13.18/16.5	11.9/15.9	6.2/10.4
	5	14.2/16.7	14.6/18.3	7.9/12.7
	6	15.14/17.6	14.9/18.0	11.0/15.7
	7	15.2/17.4	16.1/19.2	12.9/17.5
	8	15.4/17.5	16.2/19.0	15.3/20.4
	9	16.6/18.8	17.0/19.8	15.8/20.8
	10	17.6/19.7	17.3/19.7	16.2/20.8

TABLE 4.3: Number of subnetworks predicted by the various methods on a full dataset where the null distribution is computed using array rotation (rot), class-label swapping (cperm) and gene swapping (gswap); the first number denotes the number of subnetworks in the numerator of the subnetwork-level agreement and the second number denotes the number of subnetworks in the denominator of the subnetwork-level agreement; cf. equation 4.5.

	DMD		ALL		BCR	
	rot	cperm	rot	cperm	rot	cperm
ESSNet	20/23	13/15	22/24	25/27	24/29	30/32
NEA-paired	77/98	91/115	140/163	109/119	176/192	37/43
ORA-paired	30/62	30/62	34/74	34/74	53/99	53/99
ORA-hypergeo	20/46	41/141	24/60	48/73	4/14	32/166
DEAP	14/62	-	0/2	-	1/16	-
	cperm	gswap	cperm	gswap	cperm	gswap
GSEA	23/64	24/69	8/52	17/48	7/57	5/46

4.3.6 Comparing number of predicted subnetworks using negative control data

It is also possible to test whether ESSNet is robust to false positives. We conduct in-silico testing by randomly generating matrices of gene-expression data; for each gene we sample from a random normal distribution, using the same mean and standard deviation in both phenotypes. The purpose of the test is to see if ESSNet detects any subnetworks as significant when it should not. On these random input matrices, ESSNet reports very small number of false subnetworks (typically less than 3), well within that expected from the p-value threshold and much fewer than other methods; see fig. 4.14

4.3.7 Informative subnetworks

While biological pathways provide a wealth of information to explain disease phenotype, large pathways offer little biological insight. On the other hand, subnetworks may narrow down the biological cause of a disease but very small subnetworks are trivial and non-informative. In order to assess how informative our significant subnetworks are, we

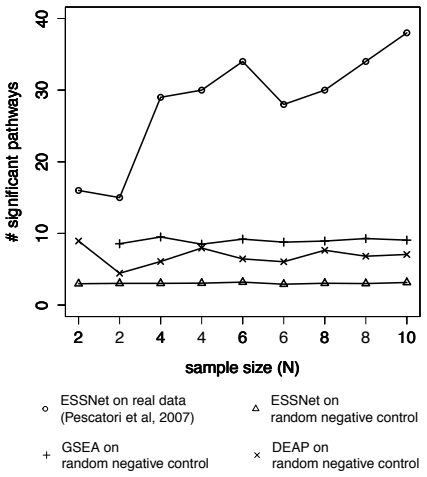


FIGURE 4.14: Number of significant subnetworks predicted by ESSNet, GSEA and DEAP on randomized negative control.

compare the size of the significant subnetworks identified by ESSNet with subnetworks induced from individual genes declared significant by t-test.

When subnetworks are induced by fragmenting pathways using significant individual genes, the genes are scattered over the pathways and have very few edges with other significant genes in the pathway. This results in very-small-sized subnetworks that contains very little useful biological information. In contrast, the subnetworks detected by ESSNet are bigger and thus more informative; cf. fig. 4.15.

Another way to determine how informative our predicted subnetworks are, is to see if they overlap with results produced by other methods. We select significant subnetworks predicted by ESS and test them using GSEA. While GSEA often does not declare a pathway to be significant when the entire pathway is supply as input, it often declares the subnetworks identified ESSNet in that pathway to be significant. Specifically, GSEA is able to recover 100%, 51% and 54% of the subnetworks in the DMD, Leukemia and ALL Subtype dataset respectively. When we included PFSNet to this analysis, the percentages increased to 100%, 90% and 91% respectively. This demonstrates subnetworks predicted by ESSNet can be recovered by other methods (provided these methods are

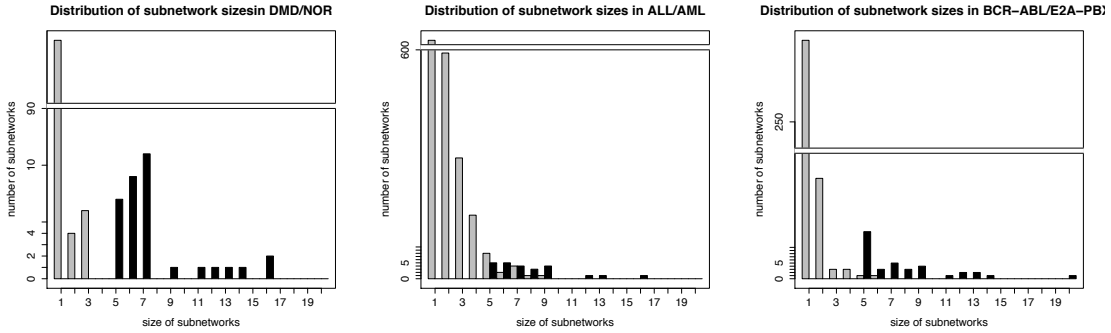


FIGURE 4.15: A figure showing the sizes of subnetwork identified by ESSNet as compared to subnetworks that are formed by significant individual genes from t-test.

supplied the subnetworks as input, and not the entire pathways they come from), and also suggests the plausibility that they are useful and pertinent.

4.3.8 Relative sensitivity

Due to the lack of gold-standard subnetworks, we evaluate the sensitivity of our method relative to other existing methods. We assume that the consistently detected pathways/subnetworks across two independent full datasets using other existing methods, are real disease-relevant subnetworks/pathways and try to measure the proportion of false negatives, i.e. disease subnetworks/pathways that are missed, from this set of gold-standard subnetworks/pathways. Similarly, we can estimate the relative sensitivity of other methods against consistently significant subnetworks identified by ESSNet.

The relative sensitivity of ESSNet when consistently significant pathways from GSEA were used as the gold-standard ranges from 60% to 80% in the three datasets evaluated. On the other hand, GSEA only recovers between 10% to 30% of the pathways that were consistently significant in ESSNet. This shows that ESSNet is more sensitive relative to GSEA. The relative sensitivity between ESSNet and PFSNet are not significantly different from each other; cf. fig. 4.16.

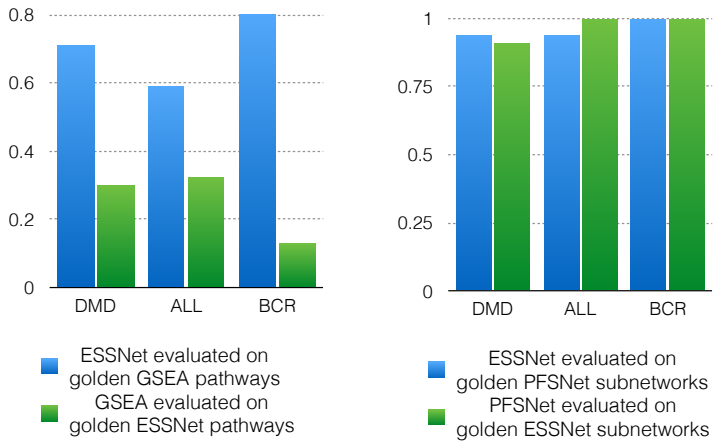


FIGURE 4.16: ESSNet has higher sensitivity relative to GSEA. On the other hand, ESSNet and PFSNet have about the same relative sensitivity.

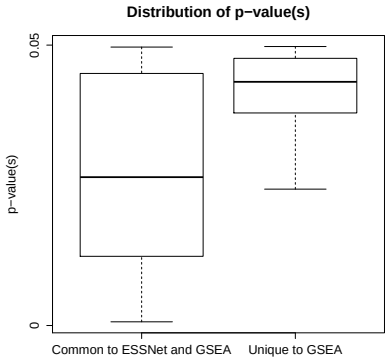


FIGURE 4.17: The pathways that are common to ESSNet and GSEA have smaller p-values than pathways unique to GSEA alone. This might suggest that pathways picked up by ESSNet is more likely to be real.

Although ESSNet does not recover all of the gold-standard pathways obtained from GSEA, the pathways that are common between ESSNet and GSEA have lower p-values than those that are unique to GSEA alone. This suggests that the pathways that are recovered by ESSNet are more reliable and pathways that are unique to GSEA could be false positives; cf. fig. 4.17.

4.3.9 Biologically-significant subnetworks

The subnetworks predicted by ESSNet have very strong biological relevance even when a small sample size is used. We used sample sizes of 2, 2 and 4 for the DMD, Leukemia and ALL Subtype datasets respectively as these sample sizes give roughly the same subnetwork agreement; cf. figs. 4.5 to 4.7. As there are many different predictions since there are many ways to partition the data into subsets of smaller sample sizes from the entire dataset, we report the subnetworks that are detected most frequently in table 4.4.

For DMD, striated muscle contraction and actin cytoskeleton signaling are the main cause of the disease (Goldstein and McNally, 2010, Krans, 2010). ESSNet was not only able to detect these two subnetworks but also other biologically-significant signaling pathways that might be the trigger for these main pathways. For example, it was reported that PTEN signaling (Dogra et al., 2006) contributes to PI3K/Akt signaling (Feron et al., 2009) which in turn affects the DMD gene found in striated muscle contraction. ECM receptor interaction was also implicated in DMD (Vidal et al., 2012).

For Leukemia, numerous works have reported the involvement of ERK/MAPK signaling (Das Gupta et al., 2013), Toll-like receptor signaling (Dimicoli et al., 2013) and JAK/-STAT signaling (Furqan et al., 2013) in interfering with apoptosis. Other subnetworks like antigen processing (Hruak and Porwit-MacDonald, 2002) and metabolism of xenobiotics by cytochrome P450 (Kanagal-Shamanna et al., 2012) have also been linked to Leukemia.

Similarly, for ALL subtype, the various subnetworks identified also have biological support, including antigen processing (Giunta and Pucillo, 2012), IFNG signaling (Kim et al., 2010), Wnt signaling (Ress and Moelling, 2005), IL-4 signaling (Cardoso et al., 2008), JAK/STAT signaling and T-Cell receptor signaling (Mumprecht et al., 2009).

TABLE 4.4: Biologically relevant subnetworks predicted by ESSNet.

DMD (N=2)	Leukemia (N=2)	ALL Subtype (N=4)
PI3K/Akt Signaling	ERK/MAPK Signaling	Antigen processing
PTEN Signaling	Toll-like receptor Signaling	IFNG Signaling
ECM Receptor	Apotosis Signaling	Wnt Signaling
Actin Cytoskeleton Signaling	JAK/STAT Signaling	IL-4 Signaling
Striated Muscle Contraction	Antigen processing	JAK/STAT Signaling
Integrin Signaling	Metab. of xenobiotics by P450	T-Cell Receptor

4.4 Discussion

Detecting disease-relevant subnetworks/genes is a difficult task when sample size is small. We have shown in this chapter that many methods that compute gene-wise differential expression perform poorly when sample size is small due to the large variance in fold-change and t-test p-values that is inevitable when a small sample population is considered.

An ideal method should be able to pick out all relevant factors underlying the phenotypes that are present in a given sample set and should not report any irrelevant factors. It follows from this ideal that we can expect a good method to satisfy these three hallmarks: (i) The selected subnetworks are reproduced when applied to new batches of data that are sufficiently representative of the phenotypes. (ii) The selected subnetworks from a large dataset should be a superset of those chosen from a subset of the dataset. (iii) The relevant subnetworks can be identified using as small a dataset as possible.

We are able to reproduce similar subnetworks in independent batches of data, this is evident in the high subnetwork-level agreement; cf. figs. 4.5 to 4.7. ESSNet also demonstrates very good precision, when compared against a set of gold-standard subnetworks derived from the full datasets; cf. table 4.1. This suggests that most of the subnetworks predicted using a small sample size are also detected in the large dataset and further implies that it does not produce a lot of false positives even when sample size is small.

On the other hand, ESSNet misses out on some gold-standard subnetworks because the small number of samples are unable to capture all the underlying phenotypic differences. However, ESSNet is also superior for large sample sizes; cf table 4.3.

Our method, ESSNet is unlike other previous methods because we do not greedily select genes to be included based on differential expression but rely on gene-expression-level ranking within a phenotype, which is shown to be stable even under extremely-small sample sizes. In addition, our conjecture that $\delta(g_i, p_j, p'_l)$ is a distribution around 0, is tested on the null distribution obtained by array rotation; this preserves the gene-gene correlation structure and is suitable for datasets with small sample size. This allows us to consistently predict relevant subnetworks even when sample size is small.

The subnetworks that we discover using ESSNet are also supported by many relevant biological literature and have the potential to allow biologist further insights to the mechanism behind the disease.

Chapter 5

Classification using subnetworks

5.1 Background

In the beginning of this dissertation, we have highlighted that classifiers for distinguishing patients from controls often have poor classification accuracy when the model is based on significant genes from one dataset and then tested on an independent dataset. This suggests that the significant genes may not necessarily be discriminatory when tested on a new batch of data.

The poor predictive accuracy can be attributed in part to batch effects causing expression values for genes in one dataset to differ greatly in another dataset. Batch effects are usually explained by some technical source of variation when experiments are done at different times and platforms. This is evident in all the datasets that we have examined in this thesis. We can visualize batch effects by performing principal component analysis (PCA), a procedure that is commonly used to find patterns in high-dimensional data. PCA transforms the data into a coordinate system whose N^{th} dimension has the N^{th} largest variance in the data. The PCA plots for the first three principle components

show that the datasets are clearly separated by their batches, making it hard to build a classifier that can predict well on both datasets (see figs. 5.1 to 5.5).

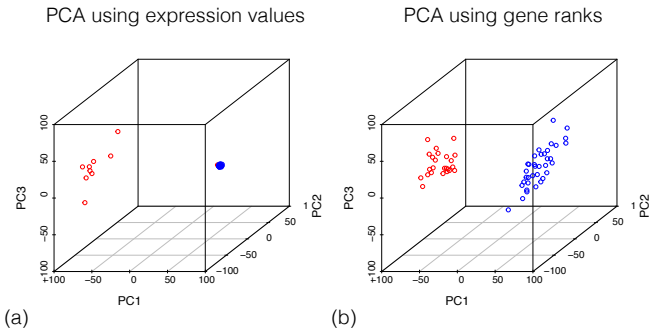


FIGURE 5.1: Batch effects in the DMD/NOR datasets, the blue and red color denote different data batches. (a) Scatterplot on the first 3 components using gene-expression values. (b) Scatterplot on the first 3 components using gene ranks.

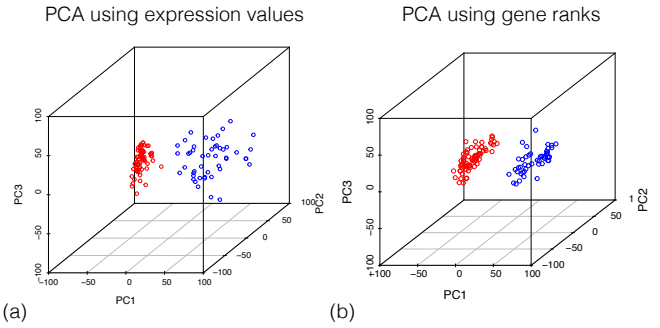


FIGURE 5.2: Batch effects in the ALL/AML datasets, the blue and red color denote different data batches. (a) Scatterplot on the first 3 components using gene-expression values. (b) Scatterplot on the first 3 components using gene ranks.

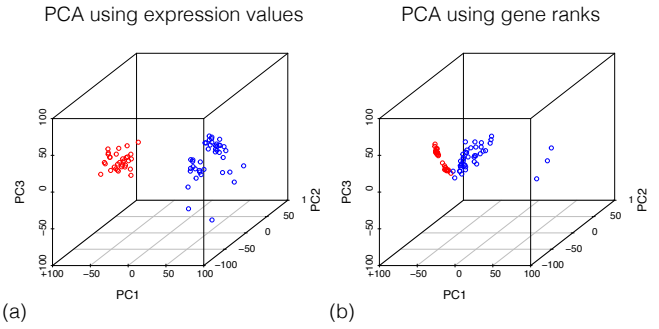


FIGURE 5.3: Batch effects in the BCR-ABL/E2A-PBX1 datasets, the blue and red color denote different data batches. (a) Scatterplot on the first 3 components using gene-expression values. (b) Scatterplot on the first 3 components using gene ranks.

Because batch effects cause gene-expression values to differ greatly in independent

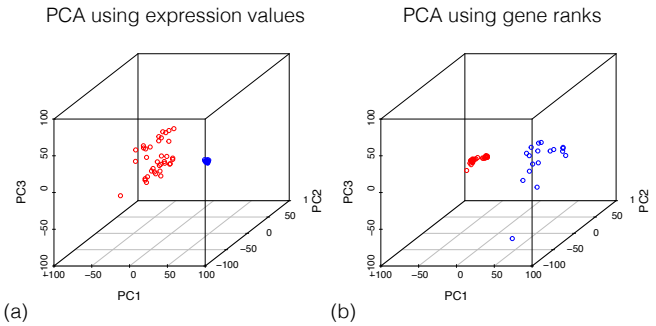


FIGURE 5.4: Batch effects in the Lung cancer datasets, the blue and red color denote different data batches. (a) Scatterplot on the first 3 components using gene-expression values. (b) Scatterplot on the first 3 components using gene ranks.

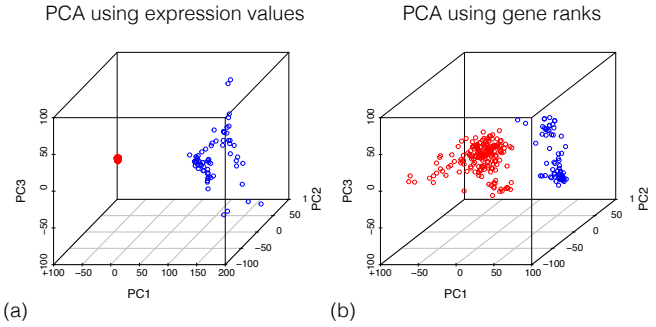


FIGURE 5.5: Batch effects in the Ovarian cancer dataset, the blue and red color denote different data batches. (a) Scatterplot on the first 3 components using gene-expression values. (b) Scatterplot on the first 3 components using gene ranks.

datasets, researchers resort to rank-based normalization to help minimize these effects. However, the PCA plots for the respective datasets show the presence of batch effects even after using rank-based normalization as a preprocessing step (see figs. 5.1 to 5.5). This suggests that rank-based normalization may not fully solve the problem and further demonstrates the drawback of using individual genes as classification features.

This leads us to find alternative features that might possibly withstand the batch effects, and achieve better prediction accuracy across the disease datasets studied in this dissertation. In this chapter, we discuss how subnetworks make good features for classifying patients.

5.2 Method

5.2.1 PFSNet feature scores

Under the PFSNet methodology, it is intuitive to use the sample scores for each subnetwork, which is computed with respect to the D and $\neg D$ phenotypes respectively, described in eqs. (3.6) and (3.7). Therefore, each subnetwork directly gives us two feature scores for a patient p_k:

$$PFSNet\_feature_1^{p_k,S} = Score_1^{p_k}(S) \tag{5.1}$$

$$PFSNet\_feature_2^{p_k,S} = Score_2^{p_k}(S) \tag{5.2}$$

where $Score_1^{p_k}(S)$ and $Score_2^{p_k}(S)$ are described in chapter 3, eqs. (3.6) and (3.7).

We add a third feature score defined as the difference between these two scores:

$$PFSNet\_feature_3^{p_k,S} = Score_1^{p_k}(S) - Score_2^{p_k}(S) \tag{5.3}$$

5.2.2 ESSNet feature scores

ESSNet can derive reliable subnetworks from small-sample-size datasets. If we have a way of scoring these subnetworks for each patient, it potentially allows us to train a reliable classifier even for small training datasets. However, it is not possible to directly use subnetworks predicted by ESSNet as features, because the scores computed by ESSNet for the subnetworks are not for individual patient. We describe below a novel idea to get around this problem.

The idea is to see whether the pair-wise differences of genes within a subnetwork between a given sample p_x and the two separate groups (D and $\neg D$) have a distribution around zero, defined as:

$$\Delta_{(D)}(S, p_x) = \{e_{g_i,p_x} - e_{g_i,p'} \mid g_i \in S \text{ and } p' \in D\} \tag{5.4}$$

$$\Delta_{(\neg D)}(S, p_x) = \{e_{g_i,p_x} - e_{g_i,p'} \mid g_i \in S \text{ and } p' \in \neg D\} \tag{5.5}$$

where $e(g_i, p_k)$ is the gene expression of gene g_i of patient p_k. It is also possible to use values after rank-based normalization and fold-change for this computation.

We can use $\Delta_{(D)}(S, p_x)$ and $\Delta_{(\neg D)}(S, p_x)$ to derive feature scores of the subnetwork S for patient p_x. If the subnetwork S is useful for classification, we expect $\Delta_{(D)}(S, p_x)$ and $\Delta_{(\neg D)}(S, p_x)$ to have a positive or negative median for patients in one of the classes. Conversely, if the subnetwork S is not useful for classification, we expect both $\Delta_{(D)}(S, p_x)$ and $\Delta_{(\neg D)}(S, p_x)$ to have a median around zero for patients in both classes, regardless of their class labels.

The feature scores are derived by using three numbers to summarize the distributions $\Delta_{(D)}(S, p_x)$ and $\Delta_{(\neg D)}(S, p_x)$, viz. taking the median and 2 standard deviations above and below the median. This gives us 6 feature scores for each subnetwork:

$$ESSNet\_feature_1^{p_x,S} = median(\Delta_{(D)}(S, p_x)) \tag{5.6}$$

$$ESSNet\_feature_2^{p_x,S} = median(\Delta_{(D)}(S, p_x)) + 2\sigma \tag{5.7}$$

$$ESSNet\_feature_3^{p_x,S} = median(\Delta_{(D)}(S, p_x)) - 2\sigma \tag{5.8}$$

and $ESSNet\_feature_4^{p_x,S}, ESSNet\_feature_5^{p_x,S}$ and $ESSNet\_feature_6^{p_x,S}$ are defined analogously based on $\Delta_{(\neg D)}(S, p_x)$.

We can also obtain pair-wise differences of genes within a subnetwork among all possible pairs of patients of phenotype D and $\neg D$, this gives us new distributions from which a t-statistic can be computed and used as feature scores.

$$\Delta_{(D-\neg D)}(S) = \{e_{g_i,p'} - e_{g_i,p''} \mid g_i \in S \text{ and } p' \in D \text{ and } p'' \in \neg D\} \tag{5.9}$$

The t-statistic between $\Delta_{(\neg D)}(S, p_x)$ and $\Delta_{(D-\neg D)}(S)$ measures how far the their means are; if p_x is of phenotype D, we would expect the two means to be close if the subnetwork is discriminatory. Conversely, if p_x is of phenotype $\neg D$, then their means should be further apart.

Similarly, we can define the other distributions $\Delta_{(\neg D-\neg D)}$, $\Delta_{(\neg D-D)}$ and $\Delta_{(D-D)}$ as:

$$\Delta_{(\neg D-\neg D)}(S) = \{e_{g_i,p'} - e_{g_i,p''} \mid g_i \in S \text{ and } p' \in \neg D \text{ and } p'' \in \neg D \textit{ and } p' \neq p''\} \tag{5.10}$$

$$\Delta_{(\neg D-D)}(S) = \{e_{g_i,p'} - e_{g_i,p''} \mid g_i \in S \text{ and } p' \in \neg D \text{ and } p'' \in D\} \tag{5.11}$$

$$\Delta_{(D-D)}(S) = \{e_{g_i,p'} - e_{g_i,p''} \mid g_i \in S \text{ and } p' \in D \text{ and } p'' \in D \text{ and } p' \neq p''\} \tag{5.12}$$

These are the 4 additional features derived from the t-statistic computed from eq. 2.6 for patient p_x:

$$ESSNet\_feature_7^{p_x,S} = T\_statistic(\Delta_{(\neg D)}(S, p_x), \Delta_{(D-\neg D)}(S)) \tag{5.13}$$

$$ESSNet\_feature_8^{p_x,S} = T\_statistic(\Delta_{(\neg D)}(S, p_x), \Delta_{(\neg D-\neg D)}(S)) \tag{5.14}$$

$$ESSNet\_feature_9^{p_x,S} = T\_statistic(\Delta_{(D)}(S, p_x), \Delta_{(D-D)}(S)) \tag{5.15}$$

$$ESSNet\_feature_{10}^{p_x,S} = T\_statistic(\Delta_{(D)}(S, p_x), \Delta_{(\neg D-D)}(S)) \tag{5.16}$$

5.3 Results

5.3.1 Batch-effect reduction

We perform PCA on the subnetwork scores from PFSNet and plotted the first three principle components on a scatterplot, the data points are colored red for one dataset and blue for another dataset from another batch. The plot in fig. 5.6 shows that the data points are no longer clustered by batch in most datasets. Rather, the data points in the PCA plots are now separated by class labels regardless of which batch it comes from; see fig. 5.7.

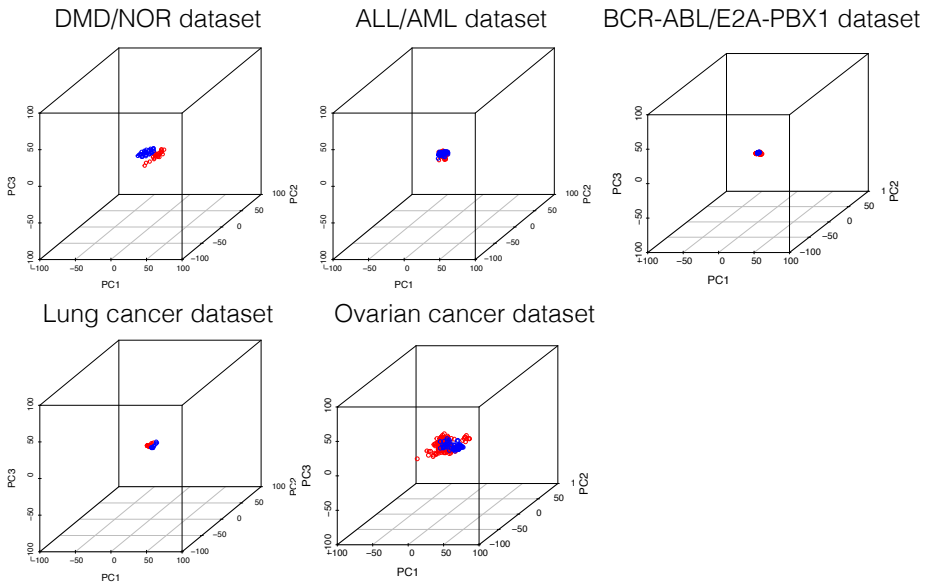


FIGURE 5.6: A figure showing that the batch effects are reduced by PFSNet subnetwork features. The colors red and blue represent different batches.

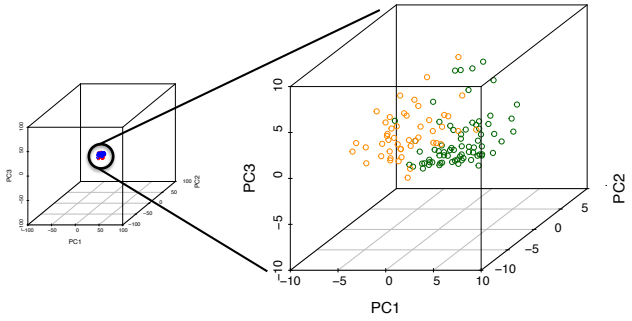


FIGURE 5.7: A figure showing that data points are separated by class labels instead of batch when PFSNet features are used. The colors green and orange represent different classes.

5.3.2 Predictive accuracy

We use Naïve Bayes (cf. Chapter 2) to build the classifier based on the significant subnetwork identified by PFSNet and ESSNet as features using their scores described in section 5.2.1 and 5.2.2. We compare this classifier against other classifiers that are built using gene features. The significant genes are chosen based on several popular gene-selection procedures detailed in chapter 2, including t-test, SAM and rank products.

Classifiers based on individual gene features do not do well, as shown in the beginning of this thesis; cf. fig. 1.4. So, we introduce a few enhancements to try and improve classifiers built on gene features that have been shown quite effective (Koh and Wong, 2012):

1. Rank-based normalization as a preprocessing step.

2. Bagging of classifiers after rank-based normalization.

When evaluating the classifiers, we use give equal weight to sensitivity and specificity. The predictive accuracy is thus defined as:

$$predictive\_accuracy = 0.5\ sensitivity + 0.5\ \ specificity \tag{5.17}$$

5.3.2.1 Gene-feature-based classifier with and without rank normalization

Rank-based normalization is able to boost predictive accuracy in some datasets. For example, in the DMD/NOR dataset predictive accuracy has increased to 90% from 50%, in the ALL/AML dataset predictive accuracy has increased to 60% from 52%, in the BCR-ABL/E2A-PBX1 dataset predictive accuracy has increased to 56% from 52%, in the Lung cancer dataset predictive accuracy has increased to 59% from 50%, and in the Ovarian cancer dataset predictive accuracy has increased to 73% from 50%, when t-test is used as the feature-selection method at the 5% significance level. Our experiments also show that the other feature-selection methods like SAM and rank products are not significantly better than the classic t-test; cf. figs. 5.8 to 5.12.

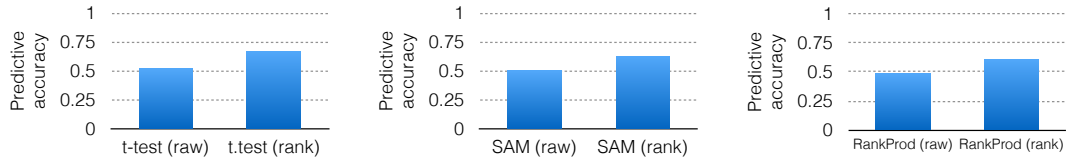


FIGURE 5.8: Predictive accuracy of gene-feature-based classifiers with and without rank normalization in the DMD/NOR dataset.

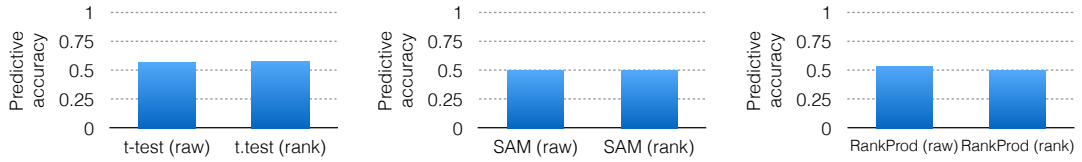


FIGURE 5.9: Predictive accuracy of gene-feature-based classifiers with and without rank normalization in the ALL/AML dataset.

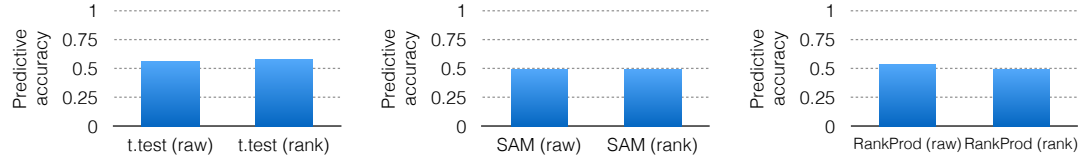


FIGURE 5.10: Predictive accuracy of gene-feature-based classifiers with and without rank normalization in the BCR-ABL/E2A-PBX1 dataset.

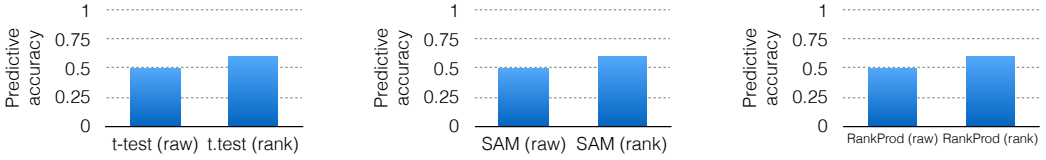


FIGURE 5.11: Predictive accuracy of gene-feature-based classifiers with and without rank normalization in the Lung cancer dataset.

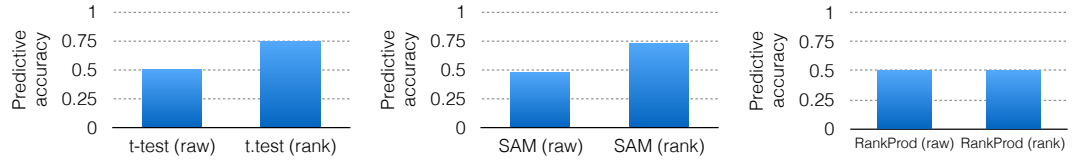


FIGURE 5.12: Predictive accuracy of gene-feature-based classifiers with and without rank normalization in the Ovarian cancer dataset.

5.3.2.2 Comparing with enhancement by bagging

We perform bagging of classifiers on the set of features selected by the t-test with rank-based normalization. Bagging increases predictive accuracy in only one dataset; in ALL/AML, predictive accuracy increases to 70% from 66.7%. In other datasets like DMD/NOR and Ovarian cancer, predictive accuracy remain about the same at 90% and 72% respectively. In the BCR-ABL/E2A-PBX1 and Lung cancer datasets, predictive accuracy is also not significantly different at 53% and 59% respectively (cf. fig. 5.13).

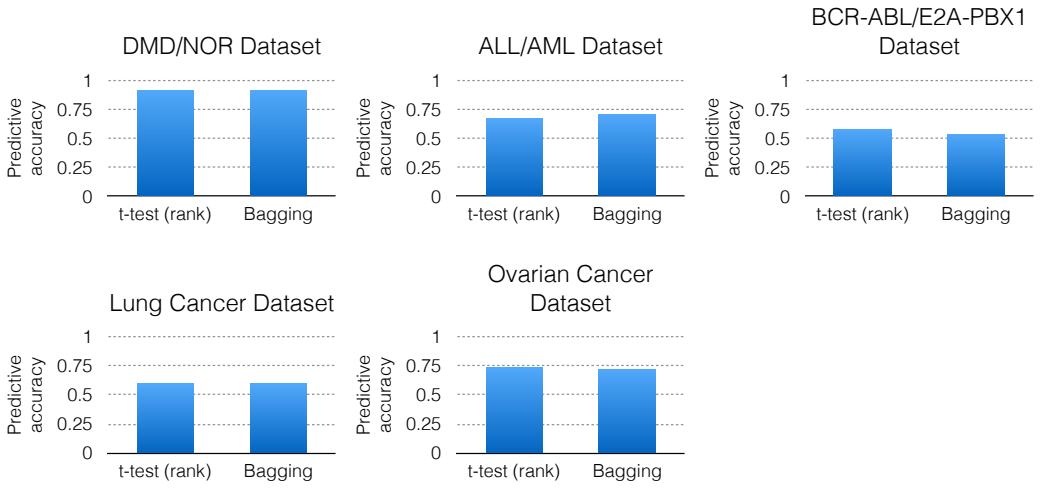


FIGURE 5.13: Predictive accuracy of gene-feature-based classifier compared to bagging.

5.3.2.3 Comparing ranked gene features, pathway features and subnetwork features from PFSNet and ESSNet

We compare, in fig. 5.14, a gene-feature-based classifier with rank normalization as described earlier with a pathway-feature-based classifier and subnetwork-feature-based classifiers. The pathway-feature-based classifier uses significant pathways identified by GSEA. However, since GSEA does not compute a feature score for its significant pathways, we generate pathway feature scores based on section 5.2.2. We also construct two subnetwork-feature-based classifiers that are based on PFSNet and ESSNet.

The GSEA classifier performs well in the ALL/AML dataset and achieves 95% predictive accuracy. However, in other datasets, it performed no signficiantly better than the gene-feature-based classifier, achieving 70%, 64%, 55% and 82%, in comparison with 90%, 58%, 59% and 74% in the gene-feature-based classifier in the DMD/NOR, BCR-ABL/E2A-PBX1, Lung cancer and Ovarian cancer datasets respectively. This suggests that signal dilution from non-disease-relevant genes in the pathway might have also affected how discriminatory the features are.

On the other hand, the PFSNet classifier outperforms the rank-normalized-gene-feature classifier in all datasets. The predictive accuracy reaches as high as 100% in the DMD/NOR and Ovarian cancer datasets. In datasets in which rank-based normalization has been shown to offer not much improvement in predictive accuracy, e.g. BCR-ABL/E2A-PBX1 and Lung cancer datasets, the PFSNet classifier achieves 83% and 92% predictive accuracy respectively compared to 58% and 59% respectively achieved by the rank-normalized-gene-feature classifier. The ESSNet classifier shows predictive accuracy very close to the PFSNet classifier. Its predictive accuracy is about 94% for the DMD/NOR dataset, 85% for the ALL/AML dataset, 83% for the BCR-ABL/E2A-PBX1 dataset, 85% for the Lung cancer dataset and 96% for the Ovarian cancer dataset. These results clearly show the superiority of the features extracted using PFSNet and ESSNet. To further demonstrate that subnetwork acts as better features, we extract and use the gene-expression profiles of those genes that belonged to the subnetworks predicted by ESSNet. We find that predictive accuracy drops to close to 50% for all the datasets we observe; cf. fig. 5.15. This suggests that the genes by themselves are not a good discriminator for classification but rather the subnetwork scores that are logically and biologically motivated serve as better features for the discriminatory task.

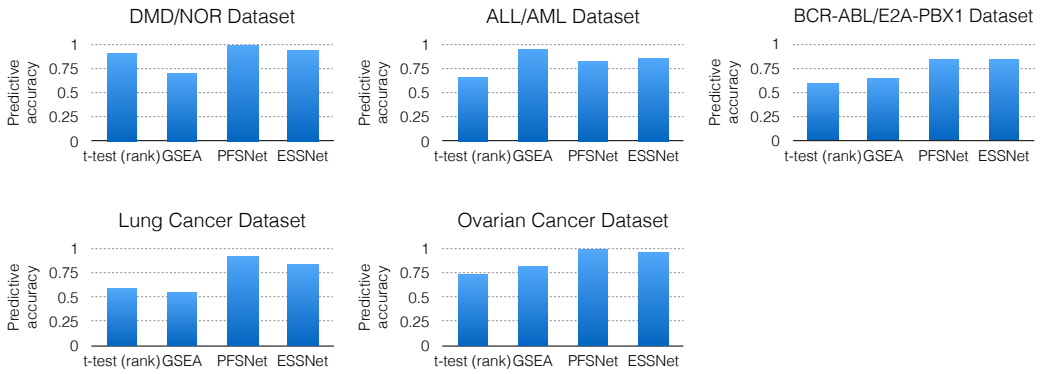


FIGURE 5.14: Predictive accuracy of gene-feature-based classifier compared to PFSNet and ESSNet classifier.

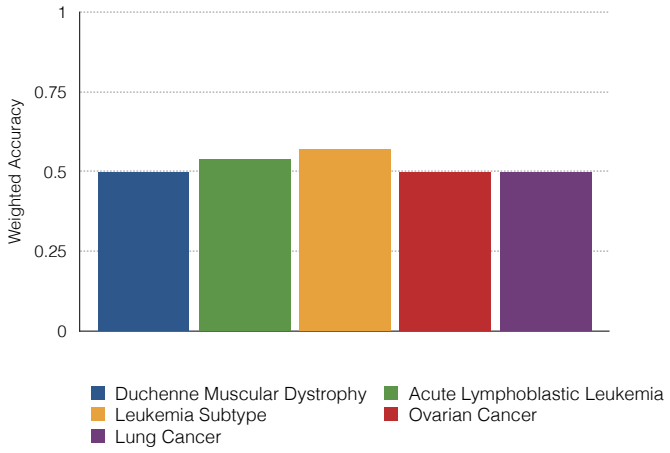


FIGURE 5.15: Predictive accuracy of gene-feature-based classifier using genes extracted from subnetworks in ESSNet; demonstrating that genes in the subnetworks by themselves are not a good discriminator for classification and the methodology behind our subnetwork feature scores detailed in sections 5.2.1 and 5.2.2 is key to the better performance.

5.3.2.4 Effects of sample size on predictive accuracy of PFSNet and ESSNet

We reduce the sample size of the training datasets to examine the effects of sample size on predictive accuracy of PFSNet classifier and ESSNet classifier respectively. For each dataset, we partition the training dataset into subsets of smaller sample sizes ranging from 2 to 10 for each phenotype. This training dataset is then used to build a Naïve Bayes classifier based on the PFSNet and ESSNet subnetwork features described in section sections 5.2.1 and 5.2.2. The predictive accuracy is then computed using a full

independent test dataset. As there are many possible ways to partition the training dataset, we report the average predictive accuracy over all the subsets for a particular sample size; cf. fig. 5.16. The PFSNet classifier's predictive accuracy drops when the training dataset sample size is reduced, achieving 58% predictive accuracy when sample size is 2 and 81% when sample size is 10. In contrast, the ESSNet classifier is more robust to smaller training datasets, achieving 68% when sample size is 2 and 90% when sample size is 10.

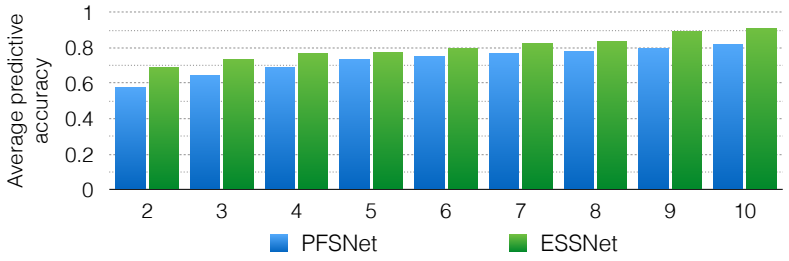


FIGURE 5.16: Predictive accuracy of PFSNet and ESSNet classifier when the training dataset is partitioned into smaller subsets and tested on a full independent dataset.

5.3.3 Unsupervised clustering

In order to further show that the subnetworks chosen as features provide good discriminatory power, we demonstrate that the subnetwork scores also do well in unsupervised learning. When the class labels are hidden from the learning model, subnetwork scores are clustered together based on their phenotype. Fig 5.17 shows hierarchical clustering performed on the various datasets using subnetwork features.

5.4 Caveats

As some of the features for the ESSNet classifier are derived from computing a t-statistic using pair-wise gene-expression differences within a subnetwork, the pair-wise differences

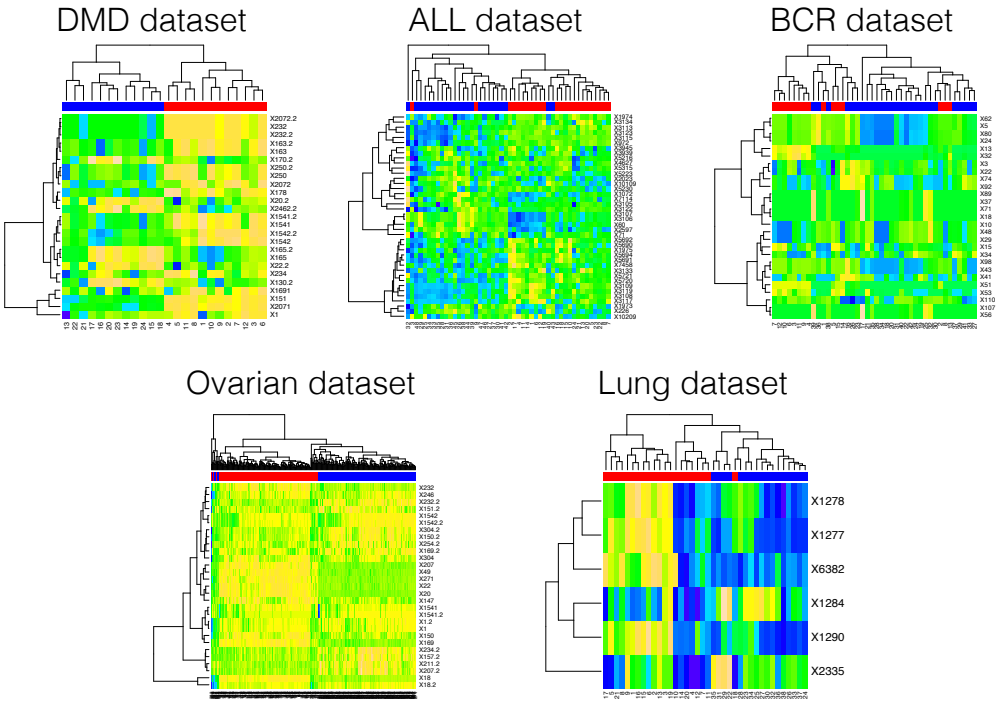


FIGURE 5.17: A figure depicting heirarchical clustering performed on the patient's subnetwork scores.

might not be independent of one another. One possible improvement is to select specific samples that are used in computing the gene-expression differences so that the differences are independent of one another.

We use the subnetwork features to generate 4 clusters of datapoints representing the distributions $\Delta_{(D-\neg D)}(S)$, $\Delta_{(\neg D-D)}(S)$, $\Delta_{(D-D)}(S)$ and $\Delta_{(\neg D-\neg D)}(S)$, on the training dataset. A sample is chosen for each of these 4 clusters based on the smallest Euclidean distance to the respective centroid of these 4 clusters. Hence, the features in eqs. (5.13) to (5.16) are modified as:

$$ESSNet\_feature_7^{p_x,S'} = T - statistic(\Delta_{(\neg D)}(S, p_x), \Delta_{(\neg D)}(S, p_{y_1})) \tag{5.18}$$

where p_{y_1} is a sample from the D phenotype whose features have the smallest Euclidian distance to the $\Delta_{(D-\neg D)}(S)$ centroid.

$$ESSNet\_feature_8^{p_x,S'} = T - statistic(\Delta_{(\neg D)}(S, p_x), \Delta_{(\neg D)}(S, p_{y_2})) \tag{5.19}$$

where p_{y_2} is a sample from the $\neg D$ phenotype whose features have the smallest Euclidian distance to the $\Delta_{(\neg D - \neg D)}(S)$ centroid.

$$ESSNet\_feature_9^{p_x,S'} = T - statistic(\Delta_{(D)}(S, p_x), \Delta_{(D)}(S, p_{y_3})) \tag{5.20}$$

where p_{y_3} is a sample from the D phenotype whose features have the smallest Euclidian distance to the $\Delta_{(D-D)}(S)$ centroid.

$$ESSNet\_feature_{10}^{p_x,S'} = T - statistic(\Delta_{(D)}(S, p_x), \Delta_{(D)}(S, p_{y_4})) \tag{5.21}$$

where p_{y_4} is a sample from the $\neg D$ phenotype whose features have the smallest Euclidian distance to the $\Delta_{(\neg D - D)}(S)$ centroid.

When these alternative feature scores are used to built the classifier, the predictive accuracy are more or less consistent with the classifier built on the unmodified feature scores, achieving about 97% for the DMD/NOR dataset, 88% for the ALL/AML dataset, 85% for the BCR-ABL/E2A-PBX-1 dataset, 85% for the Lung cancer dataset and 92% for the Ovarian cancer dataset; cf. fig. 5.18

5.5 Discussion

Traditional methods of classifying patients using their gene-expression profiles are unable to accurately predict the outcome of new batches of patients partly because of the inherent batch effect. Methods that try to tackle this issue by normalizing the gene-expression profiles also do not consistently achieve good results. Instead, our method

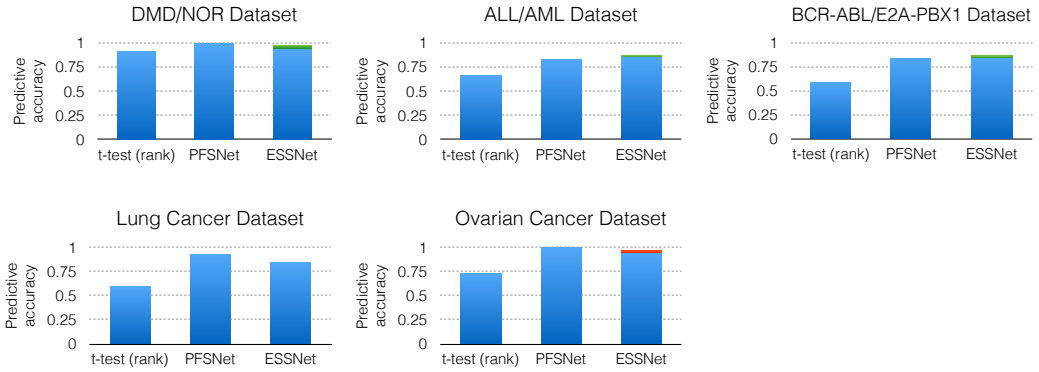


FIGURE 5.18: Predictive accuracy of the ESSNet classifier using the modified subnetwork features. The green and red bars represent an increase and decrease in predicitive accuracy when the modified subnetwork features are used (cf. eqs. (5.13) to (5.16)) compared to the original subnetwork features.

does not rely directly on gene-expression values as feature values but transforms these gene-expression values into a different feature space defined by subnetworks, previously discussed in chapters 3 and 4. By doing so, we circumvent the batch effects and are able to achieve good prediction results. In the case of ESSNet, we have also shown that these features still remain relevant when the training data is small.

Isolating disease-relevant subnetworks from pathways offer many insights to researchers. The machine-learning methods employed in this chapter lends further confidence that the subnetworks generated by both PFSNet and ESSNet are not only explanatory to the cause of disease but are also discriminatively accurate when they are used to identify unlabeled patients given their gene-expression profiles.

Chapter 6

Discussion and Future Work

6.1 Conclusions

High-throughput microarray experiments are often analyzed to obtain useful biological insights. Many modern methods incorporate biological pathways into their analysis to provide better biological interpretability of the results, but many of these methods are unable to reproduce their results when applied to an independent dataset describing the same disease. They fare even worse in datasets with small sample sizes.

In our first contribution, as discussed in Chapter 3, we investigated ways to relax the over-reliance of hard thresholds imposed by SNet (Soh et al., 2011), one of the methods that have previously demonstrated high levels of subnetwork-level agreement between two batches of data belonging to the same disease phenotype. This is crucial because the recommended thresholds to use may not always yield optimal results in different datasets. Although it may be possible to experimentally obtain optimal thresholds, an optimization task is cumbersome, and may lead to over-tuning of the parameters. In contrast, our method relies on a fuzzy function that adapts to the majority-voting

feature of SNet and is perfectly able to emulate SNet. This, coupled with a more robust significance test, by computing a distribution of the differences in the two subnetwork scores and testing if this distribution has a mean around zero, enables us to produce even higher subnetwork-level agreement than SNet as well as other previously analyzed methods. The higher subnetwork-level agreement that results from our method, PFSNet (Lim and Wong, 2014), lends more confidence to the real cause of disease.

Our second contribution, described in chapter 4, extended the microarray analysis using pathway databases to situations where the dataset contains only a small number of samples. This is particularly novel and useful because many laboratories are constrained to a small number of samples for various reasons but contemporary methods for analyzing these data perform poorly. This is because most methods focus on differentiating two phenotype scores but a small sample of scores may not be truly representative of the population. We differentiated ourselves from other methods by reasoning about a distribution computed from the difference in gene-expression values of the two phenotypes in isolated subnetworks instead. In addition, unlike other contemporary methods like ORA (Khatri and Drăghici, 2005), GSEA (Subramanian et al., 2005) and DEAP (Haynes et al., 2013) that rely on a pre-computing every genes' differential expression, we generate subnetworks around high-ranking genes and are less susceptible to the high variance of t-test p-values in small-sample-size datasets. Because of these reasons, we are able to detect consistent subnetworks, using our method ESSNet, even in small-sample-size datasets with the help of biological pathways.

Our final contribution, as detailed in chapter 5, uses the subnetworks predicted in the previous two chapters as features in a machine-learning algorithm. We demonstrate that traditional methods that perform feature extraction on individual genes are not good because of two reasons. 1) Their selected genes are not consistently detected in different batches of data of the same disease type. 2) They have poor cross-batch classification

accuracy. This is in part due to batch effects, a phenomenon where the results are correlated to technical sources of variation instead of biological outcome. Modern methods try to overcome this problem by normalization but this did not consistently deliver good results in the datasets that we tested. In contrast, we once again differentiated ourselves by using the extracted subnetwork features and their scoring methodology, derived from PFSNet and ESSNet, which allowed us to achieve much better cross-batch classification accuracy than many other methods.

6.2 Future work

Our results motivate many other interesting and broader areas of research, which are elaborated below.

6.2.1 Multi-omics analysis

One way that could further validate our methods is to integrate the analysis of other sources of data. For example, researchers may identify causal genes by looking at single-nucleotide polymorphism (SNPs), methylation sites or copy-number variations. However, these analyses detect millions of mutations or methylation sites because the experiment is done at single-nucleotide resolution. Using our subnetworks as a basis for this analysis could potentially narrow down this huge list of mutations to a handful. In addition, one could also make a conjecture that the influence of these SNPs would lie on the boundary of the subnetworks, triggering a cascade of events that explain the onset of disease. Thus, multi-omics analysis serves two purposes. On one hand, a large list of predictions from other analysis can be narrowed down. For example, in fig. 6.1, the number of differentially-methylated sites detected by t-test is greatly reduced. On the other hand, our subnetworks can be validated with additional supporting information.

For example, fig. 6.2a shows the methylation sites of a gene in an example subnetwork in liver cancer that influences a cascade of genes leading to the compromise of the hepatic tight junction in fig. 6.2b.

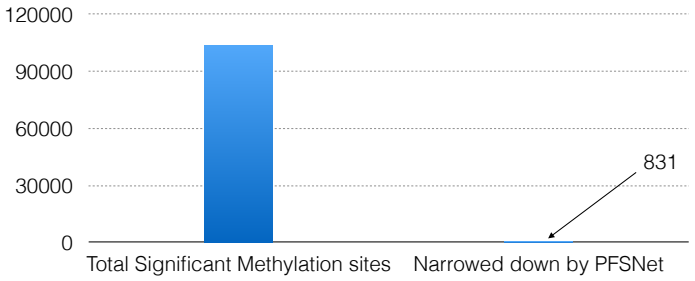


FIGURE 6.1: By looking at the genes in the induced subnetworks predicted by PFSNet, we are able to narrow down a huge list of differentially methylation sites to a handful and more insightful genes for further analysis.

6.2.2 Applications to RNA-seq data

One of the newer technologies in measuring gene-expression profiles is the use of next-generation sequencing. In RNA-seq, transcribed genes are fragmented into pieces and tagged so that sequencing can identify as well as quantify the gene's expression. Although RNA-seq data is less widely available due to experimental costs, it offers a wider dynamic range, resulting in gene-expression profiles that have high granularity. It also has increased sensitivity since it is able to better detect genes expressed at very low levels. It is possible to use RNA-seq data as inputs to PFSNet since it is robust to the inclusion of lowly-expressed genes. In addition, because RNA-seq experiments are more costly, ESSNet would be a good choice to analyze small-sample-size datasets.

6.2.3 Utilizing directional gene relationships

So far our methods regard the biological pathways as an undirected graph. However, in certain cases, biological events are not merely gene interactions but could indicate a

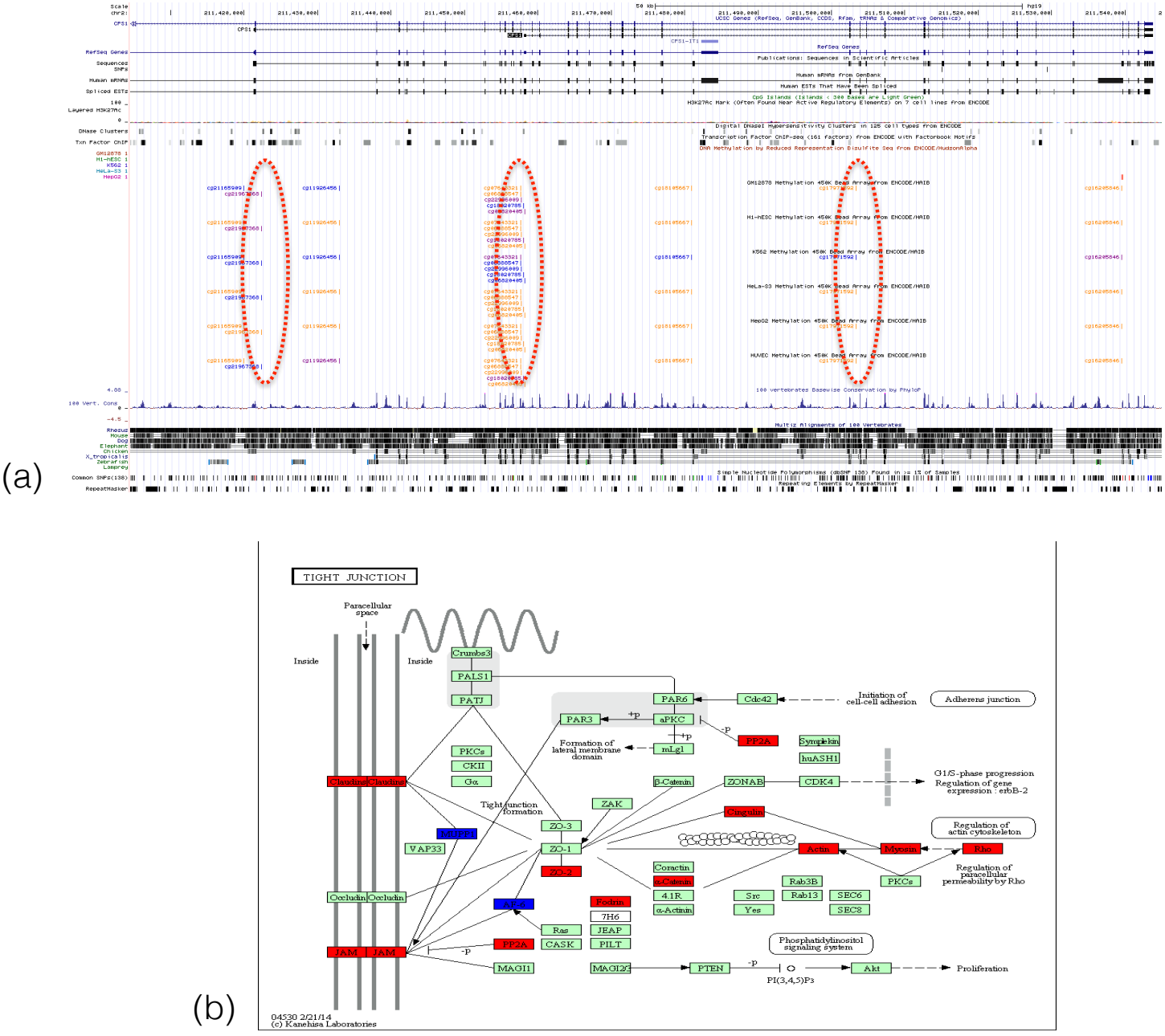


FIGURE 6.2: An example of validating PFSNet subnetworks via multi-omics data. a) the methylation sites circled in red are highly correlated in the disease phenotype and are also detected in one of the subnetworks predicted by PFSNet (Image reproduced from USCS Genome Browser). b) the hyper-methylation of subnetwork genes highlighted in blue effect a cascade of genes leading to the compromise of the hepatic tight junction, possibly explaining the metastatis of liver cancer.

transfer of molecules in metabolic pathways or a transfer of signals in signaling pathways. It might be possible to also take into account the directionality of the edges within such biological pathways so that the relationship between genes within the significant subnetworks are also consistently preserved across the datasets.

Bibliography

Affymetrix (2002). http://www.affymetrix.com/support/technical/whitepapers/sadd_whitepaper.pdf. [Online; accessed 15-June-2014].

Armstrong, S. A., Staunton, J. E., Silverman, L. B., Pieters, R., den Boer, M. L., Minden, M. D., Sallan, S. E., Lander, E. S., Golub, T. R., and Korsmeyer, S. J. (2002). MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nat Genet*, 30(1):41–47.

Breiman, L. (1996). Bagging predictors. *Mach. Learn.*, 24(2):123–140.

Breitling, R., Armengaud, P., Amtmann, A., and Herzyk, P. (2004). Rank products: a simple, yet powerful, new method to detect differentially regulated genes in replicated microarray experiments. *FEBS Lett.*, 573(1-3):83–92.

Cardoso, B. A., Martins, L. R., Santos, C. I., Nadler, L. M., Boussiotis, V. A., Cardoso, A. A., and Barta, J. T. (2008). Interleukin-4 stimulates proliferation and growth of t-cell acute lymphoblastic leukemia cells by activating mtor signaling. *Leukemia*, 23(1):206–208.

Chu, Y. and Corey, D. R. (2012). RNA sequencing: platform selection, experimental design, and data interpretation. *Nucleic Acid Ther*, 22(4):271–274.

Das Gupta, S., Halder, B., Gomes, A., and Gomes, A. (2013). Bengalin initiates autophagic cell death through ERK-MAPK pathway following suppression of apoptosis in human leukemic U937 cells. *Life Sci.*, 93(7):271–276.

DeRisi, J., Penland, L., Brown, P. O., Bittner, M. L., Meltzer, P. S., Ray, M., Chen, Y., Su, Y. A., and Trent, J. M. (1996). Use of a cDNA microarray to analyse gene expression patterns in human cancer. *Nat. Genet.*, 14(4):457–460.

Dimicoli, S., Wei, Y., Bueso-Ramos, C., Yang, H., Dinardo, C., Jia, Y., Zheng, H., Fang, Z., Nguyen, M., Pierce, S., Chen, R., Wang, H., Wu, C., and Garcia-Manero, G. (2013). Overexpression of the toll-like receptor (TLR) signaling adaptor MYD88, but lack of genetic mutation, in myelodysplastic syndromes. *PLoS ONE*, 8(8):e71120.

Dogra, C., Changotra, H., Wergedal, J. E., and Kumar, A. (2006). Regulation of phosphatidylinositol 3-kinase (PI3K)/Akt and nuclear factor-kappa B signaling pathways in dystrophin-deficient skeletal muscle in response to mechanical stretch. *J. Cell. Physiol.*, 208(3):575–585.

Dorum, G., Snipen, L., Solheim, M., and Saeb?, S. (2009). Rotation testing in gene set enrichment analysis for small direct comparison experiments. *Stat. Appl. Genet. Mol. Biol.*, 8:Article34.

Feron, M., Guevel, L., Rouger, K., Dubreil, L., Arnaud, M. C., Ledevin, M., Megeney, L. A., Cherel, Y., and Sakanyan, V. (2009). PTEN contributes to profound PI3K/Akt signaling pathway deregulation in dystrophin-deficient dog muscle. *Am. J. Pathol.*, 174(4):1459–1470.

Fodor, S. P., Read, J. L., Pirrung, M. C., Stryer, L., Lu, A. T., and Solas, D. (1991). Light-directed, spatially addressable parallel chemical synthesis. *Science*, 251(4995):767–773.

Furey, T. S., Cristianini, N., Duffy, N., Bednarski, D. W., Schummer, M., and Haussler, D. (2000). Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914.

Furqan, M., Mukhi, N., Lee, B., and Liu, D. (2013). Dysregulation of jak-stat pathway in hematological malignancies and jak inhibitors for clinical application. *Biomarker Research*, 1(1):5.

Gatti, D., Barry, W., Nobel, A., Rusyn, I., and Wright, F. (2010). Heading down the wrong pathway: On the influence of correlation within gene sets. *BMC Genomics*, 11(1):574.

Geistlinger, L., Csaba, G., Küffner, R., Mulder, N., and Zimmer, R. (2011). From sets to graphs: Towards a realistic enrichment analysis of transcriptomic systems. *Bioinformatics*, 27(13):i366–i373.

Giunta, M. and Pucillo, C. (2012). Bcr-abl rearrangement and hla antigens: a possible link to leukemia pathogenesis and immunotherapy. *Revista brasileira de hematologia e hemoterapia*, 34(5):323–324.

Goeman, J. J. and Bühlmann, P. (2007). Analyzing gene expression data in terms of gene sets: Methodological issues. *Bioinformatics*, 23(8):980–987.

Goeman, J. J., van de Geer, S. A., de Kort, F., and van Houwelingen, H. C. (2004). A global test for groups of genes: Testing association with a clinical outcome. *Bioinformatics*, 20(1):93–99.

Goldstein, J. A. and McNally, E. M. (2010). Mechanisms of muscle weakness in muscular dystrophy. *The Journal of General Physiology*, 136(1):29–34.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D., and

Lander, E. S. (1999a). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D., and Lander, E. S. (1999b). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537.

Hart, S. N., Therneau, T. M., Zhang, Y., Poland, G. A., and Kocher, J. P. (2013). Calculating sample size estimates for RNA sequencing data. *J. Comput. Biol.*, 20(12):970–978.

Haslett, J. N., Sanoudou, D., Kho, A. T., Bennett, R. R., Greenberg, S. A., Kohane, I. S., Beggs, A. H., and Kunkel, L. M. (2002). Gene expression comparison of biopsies from duchenne muscular dystrophy (dmd) and normal skeletal muscle. *Proceedings of the National Academy of Sciences USA*, 99(23):15000–15005.

Haynes, W. A., Higdon, R., Stanberry, L., Collins, D., and Kolker, E. (2013). Differential expression analysis for pathways. *PLoS Comput. Biol.*, 9(3):e1002967.

Hruak, O. and Porwit-MacDonald, A. (2002). Antigen expression patterns reflecting genotype of acute leukemias. *Leukemia*, 16(7):1233–1258.

Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U., and Speed, T. P. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, 4(2):249–264.

Kanagal-Shamanna, R., Zhao, W., Vadhan-Raj, S., Nguyen, M. H., Fernandez, M. H., Medeiros, L. J., and Bueso-Ramos, C. E. (2012). Over-expression of cyp2e1 mrna and protein: implications of xenobiotic induced damage in patients with de novo

acute myeloid leukemia with inv(16)(p13.1q22). *Int. J. Environ. Res. Public Health*, 9(8):2788–2800.

Kanehisa, M. and Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 28(1):27–30.

Kanehisa, M., Goto, S., Sato, Y., Furumichi, M., and Tanabe, M. (2012). KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Research*, 40(D1):D109–D114.

Kelder, T., van Iersel, M. P., Hanspers, K., Kutmon, M., Conklin, B. R., Evelo, C. T., and Pico, A. R. (2012). Wikipathways: Building research communities on biological pathways. *Nucleic Acids Research*, 40(D1):D1301–D1307.

Khatri, P. and Drăghici, S. (2005). Ontological analysis of gene expression data: Current tools, limitations, and open problems. *Bioinformatics*, 21(18):3587–3595.

Kim, D. H., Kong, J. H., Byeun, J. Y., Jung, C. W., Xu, W., Liu, X., Kamel-Reid, S., Kim, Y. K., Kim, H. J., and Lipton, J. H. (2010). The IFNG (IFN-gamma) genotype predicts cytogenetic and molecular response to imatinib therapy in chronic myeloid leukemia. *Clin. Cancer Res.*, 16(21):5339–5350.

Koh, C. H. and Wong, L. (2012). Embracing noise to improve cross-batch prediction accuracy. *BMC Systems Biology*, 6(Suppl 2):S3.

Krans, J. L. (2010). The sliding filament theory of muscle contraction. *Nature Education*, 3(9):66.

Li, J., Liu, H., Downing, J. R., Yeoh, A. E., and Wong, L. (2003). Simple rules underlying gene expression profiles of more than six subtypes of acute lymphoblastic leukemia (ALL) patients. *Bioinformatics*, 19(1):71–78.

Lim, K. and Wong, L. (2014). Finding consistent disease subnetworks using PFSNet. *Bioinformatics*, 30(2):189–196.

Long, P. M. and Servedio, R. A. (2008). Random classification noise defeats all convex potential boosters. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 608–615, New York, NY, USA. ACM.

Mumprecht, S., Claus, C., Schurch, C., Pavelic, V., Matter, M. S., and Ochsenbein, A. F. (2009). Defective homing and impaired induction of cytotoxic T cells by BCR/ABL-expressing dendritic cells. *Blood*, 113(19):4681–4689.

Pescatori, M., Broccolini, A., Minetti, C., Bertini, E., Bruno, C., D'amico, A., Bernardini, C., Mirabella, M., Silvestri, G., Giglio, V., Modoni, A., Pedemonte, M., Tasca, G., Galluzzi, G., Mercuri, E., Tonali, P. A., and Ricci, E. (2007). Gene expression profiling in the early phases of DMD: A constant molecular signature characterizes DMD muscle from early postnatal life throughout disease progression. *The FASEB Journal*, 21(4):1210–1226.

Rayet, B. and Gelinas, C. (1999). Aberrant rel/nfkb genes and activity in human cancer. *Oncogene*, 18(49):6938–6947.

Ress, A. and Moelling, K. (2005). Bcr is a negative regulator of the Wnt signalling pathway. *EMBO Rep.*, 6(11):1095–1100.

Ross, M. E., Mahfouz, R., Onciu, M., Liu, H.-C., Zhou, X., Song, G., Shurtleff, S. A., Pounds, S., Cheng, C., Ma, J., Ribeiro, R. C., Rubnitz, J. E., Girtman, K., Williams, W. K., Raimondi, S. C., Liang, D.-C., Shih, L.-Y., Pui, C.-H., and Downing, J. R. (2004). Gene expression profiling of pediatric acute myelogenous leukemia. *Blood*, 104(12):3679–3687.

Rustici, G., Kolesnikov, N., Brandizi, M., Burdett, T., Dylag, M., Emam, I., Farne, A., Hastings, E., Ison, J., Keays, M., Kurbatova, N., Malone, J., Mani, R., Mupo, A., Pedro Pereira, R., Pilicheva, E., Rung, J., Sharma, A., Tang, Y. A., Ternent, T., Tikhonov, A., Welter, D., Williams, E., Brazma, A., Parkinson, H., and Sarkans, U. (2013). ArrayExpress update–trends in database growth and links to data analysis tools. *Nucleic Acids Res.*, 41(Database issue):D987–990.

Ruxton, G. D. (2006). The unequal variance t-test is an underused alternative to Student's t-test and the Mann–Whitney u test. *Behavioral Ecology*, 17(4):688–690.

Schapire, R. E. (1990). The strength of weak learnability. *Mach. Learn.*, 5(2):197–227.

Sivachenko, A. Y., Yuryev, A., Daraselia, N., and Mazo, I. (2007). Molecular networks in microarray analysis. *Journal of Bioinformatics and Computational Biology*, 5(2b):429–456.

Soh, D., Dong, D., Guo, Y., and Wong, L. (2010a). Consistency, comprehensiveness, and compatibility of pathway databases. *BMC Bioinformatics*, 11(1):449.

Soh, D., Dong, D., Guo, Y., and Wong, L. (2010b). Consistency, comprehensiveness, and compatibility of pathway databases. *BMC Bioinformatics*, 11(1):449.

Soh, D., Dong, D., Guo, Y., and Wong, L. (2011). Finding consistent disease subnetworks across microarray datasets. *BMC Bioinformatics*, 12(Suppl 13):S15.

Stobbe, M., Houten, S., Jansen, G., van Kampen, A., and Moerland, P. (2011). Critical assessment of human metabolic pathway databases: a stepping stone for future integration. *BMC Systems Biology*, 5(1):165.

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S., and Mesirov, J. P. (2005). Gene set enrichment analysis: A knowledge-based approach for interpreting

genome-wide expression profiles. *Proceedings of the National Academy of Sciences USA*, 102(43):15545–15550.

Tusher, V. G., Tibshirani, R., and Chu, G. (2001). Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences USA*, 98(9):5116–5121.

Venet, D., Dumont, J. E., and Detours, V. (2011). Most random gene expression signatures are significantly associated with breast cancer outcome. *PLoS Comput. Biol.*, 7(10):e1002240.

Vidal, B., Ardite, E., Suelves, M., Ruiz-Bonilla, V., Janué, A., Flick, M. J., Degen, J. L., Serrano, A. L., and Muñoz-Cánoves, P. (2012). Amelioration of duchenne muscular dystrophy in mdx mice by elimination of matrix-associated fibrin-driven inflammation coupled to the αMβ2 leukocyte integrin receptor. *Human Molecular Genetics*, 21(9):1989–2004.

Wang, Z., Gerstein, M., and Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.*, 10(1):57–63.

Yeoh, E. J., Ross, M. E., Shurtleff, S. A., Williams, W. K., Patel, D., Mahfouz, R., Behm, F. G., Raimodi, S. C., Relling, M. V., Patel, A., Cheng, C., Campana, D., Wilkins, D., Zhou, X., Li, J., Liu, H., Pui, C.-H., Evans, W. E., Naeve, C., Wong, L., and Downing, J. R. (2002). Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. *Cancel cell*, 1:133–143.

Zampieri, M., Legname, G., Segrè, D., and Altafini, C. (2011). A system-level approach for deciphering the transcriptional response to prion infection. *Bioinformatics*, 27(24):3407–3414.

Zhang, M., Zhang, L., Zou, J., Yao, C., Xiao, H., Liu, Q., Wang, J., Wang, D., Wang, C., and Guo, Z. (2009). Evaluating reproducibility of differential expression discoveries in microarray studies by considering correlated molecular changes. *Bioinformatics*, 25(13):1662–1668.

Zhou, H., Jin, J., Zhang, H., Yi, B., Wozniak, M., and Wong, L. (2012). IntPath—an integrated pathway gene relationship database for model organisms and important pathogens. *BMC Systems Biology*, 6(Suppl 2):S2.