Essentially, all models are wrong, but some are useful.

– George Box.

University of Alberta

Estimating the Overlap of Top Instances in Lists Ranked by Correlation to Label

by

Babak Damavandi

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

©Babak Damavandi
Spring 2012
Edmonton, Alberta

Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-90133-5

Our file Notre référence
ISBN: 978-0-494-90133-5

Abstract

Recent advances in high-throughput technologies, such as genome-wide SNP analysis and microarray gene expression profiling, have led to a multitude of ranked lists, where the features (SNPs, genes) are sorted based on their individual correlation with a phenotype. Multiple reviews have shown that most such rankings vary considerably across different studies, even in the case of subsampling from a single dataset. This motivates our interest in formally investigating the overlap of the top ranked features in two lists sorted by correlation with an outcome.

This dissertation presents a mathematical model for better understanding lists whose entries are ranked by Pearson correlation coefficient with an outcome. We show that our model is able to accurately predict the expected overlap between two ranked lists based on reasonable assumptions. We also discuss how to generalize this model to find the overlap between other forms of rankings, provided that they satisfy mild assumptions.

Acknowledgements

First, I would like to thank my supervisor, Dr. Russ Greiner, for his genius, generosity, and seemingly unlimited patience – which, believe me, was put to test numerous times. His guidance and advice was what made this thesis possible, and I am grateful to him for all the academic achievements I have had in the past two years. I would also like to thank Dr. Csaba Szepesvari, Dr. Sambasivarao Damaraju and Dr. Peter Hooper for their very helpful advice and suggestions on the direction of my thesis.

To Meysam Bastani and Fatemeh Miri, thank you for being the most interesting couple I have ever met. Your life is an example of what I wish I could have some day. Thanks for being there for me, thanks for listening to me whine, and, well, thank you for occasionally feeding me. You are my family here. Huge thanks to Yavar Naddaf, for being the computer scientist/philosopher/chef, a fine example of a specie I did not know existed before meeting him. I will treasure all the memories of Thursdays and Saturdays spent at your place. Credit goes to Ghazal Pasha, for her angel-like patience, and her almost supernatural ability of listening to my non-stop stream of nagging. Kudos to Farzad Sangi, for being the best roommate anyone could possibly ask for. Your life-style is a lesson in self-discipline, and your cooking truly saved me from my subpar – read non-existent – skillet skills[1]. Meisam Vosoughpour, you are truly one of the best people I have met in my life. Your unique sense of humour, your kindness, and your uncanny ability of telling me what I should be thinking, are all parts of what makes you irreplaceable. I am honoured and flattered to be your friend. Aaron Luchko, you are a case-study in perseverance. May gods save you from another flock of axis-of-evil citizens before you leave this department. Thanks to Ramtin Pedarsani, for being my go-to guy on many subjects, from math to how-you-should-live-your-life, for the better part of the past 12 years, and hopefully, many years to come. Thank you Amir-Massoud Farahmand, for all the conversations we had in sub, for your advice and moral support, and for your valuable feedback on parts of my thesis.

Thanks to all my friends in the Department of Computing Science, you were what made living here so amazingly rewarding. Thank you for tolerating all my idiosyncrasies, and my sometimes admittedly nasty sense of humour.

I think I won the parents lottery, nothing else explains why I would be part of that great family.

[1] For anyone who does not know me well enough, and has read this far: Yes, I care a lot about food.

I will not begin to explain how grateful I am to my family here, as that would probably need a thesis for itself. My father, whose thirst for knowledge was always a source of inspiration for me, saw great potential in genomics, years before the first human genome was sequenced. He urged me to consider a career in biology. My love of computers prevailed, and I chose computer science; a decision that has been arguably the single best decision of my adult life so far. The two fields are fundamentally very far apart. My work on gene signatures in this thesis, is probably the closest I will ever get to realizing his dream of being a tiny part of the next revolution in biology.

This thesis is dedicated to his memory.

Table of Contents

List of Tables

List of Figures

List of Algorithms

... ...

Chapter 1

Introduction

1.1 High-Throughput Technologies

According to WHO cancer factsheet[1], cancer is the leading cause of death, accounting for more than 13% of deaths worldwide. The same factsheet suggests that deaths caused by cancer are projected to rise to 11 million by 2030, of which more than 30% can be prevented. Several factors could contribute to lowering the death-rate of cancer, including creating more effective diagnostic and prognostic tools and drugs. The advent of high-throughput technologies, such as microarray expression profiling and Genome-Wide Association Studies (GWAS), is an important step towards realizing this goal.

As an "Array of Hope" [39], microarray technology has been considered as a potential source for delivering significant discoveries and insights, a source to inspire and influence research in biology: *"Genomics aims to provide biologists with the equivalent of chemistry's Periodic Table, an inventory of all genes used to assemble a living creature, together with an insightful system for classifying these building blocks ... Arrays offer the first great hope for such global views by providing a systematic way to survey DNA and RNA variation. They seem likely to become a standard tool of both molecular biology research and clinical diagnostics."* [39]. Microarrays are widely used in many areas, especially in oncology to discover new relevant genes, to build both predictive and prognostic classifiers, predict disease progression, assign genes to biological pathways, and many more applications [8, 49, 51, 59, 65].

One of the applications of microarray technology is using microarray data in 'Gene Signature Profiling', i.e., identifying genes whose high expression levels are significantly correlated with a phenotype. Ideally, this process should produce the same lists for the same phenotype across studies. However, researchers have found that these gene signatures vary considerably from study to study; a fact that has led some [57] to call microarrays an 'Array of Problems'.[2] This has sometimes led to confusion among researchers, with articles and responses published on discrepancy of gene signatures. As an example, take the study published by Lafferty-Whyte et al. [38], providing a

[1] `http://www.who.int/mediacentre/factsheets/fs297/en/`

[2] The idea of contrasting 'Array of Hope' and 'Array of Problems' papers was inspired by [76].

new gene signature, the responses of another research team reporting their inability to reproduce the originally published signature [17], and the original paper's authors response, claims that they were able to reproduce the original signature, and attributes the discrepancy to *"different microarray platforms, and different probe designs"* [36].

This low overlap "anomaly" has received extensive attention in the literature. As will be discussed in detail in Section 2.1, many have attributed this discrepancy to environmental factors, such as different array platforms or different patient demographics. However, there is evidence that suggests that ranked gene lists will share only a small number of genes, even in the case of very low environmental noise. This motivates our interest in formally investigating the topic of overlap between two ranked lists, focusing on two ranked gene lists as a practical illustration. In this thesis, we will build a model to estimate the expected overlap between two gene signatures (or, in general, two ranked lists). That is, we will try to answer the following question: if almost all the environmental factors were eliminated, how much will the top-ranked genes from the n-patient study A, on a given phenotype have in common with the top-ranked genes of another n-patient study B, over the same phenotype? We will show that the amount of overlap is highly dependent on the number of patients n, and is far smaller than 100% in most cases. This thesis provides a framework for approximating the overlap of two lists ranked by correlation with outcome, using gene signatures as an application. We will derive a closed form analytical solution for estimating the expected overlap, and will also provide a stochastic approximation framework for estimating the expected overlap.

1.2 Overview

In Chapter 2, we present background information on high throughput technologies and gene signatures, review studies and meta-analyses that explore the issue of overlap between different signatures, and briefly discuss the existing literature on ranking and order statistics. Chapter 3 presents our theoretical framework for computing the expected overlap between ranked lists, reviews the challenges associated with implementing the framework, and suggests methods for extending it to solve a larger class of problem. In Chapter 4, we will discuss our simulation framework for stochastically approximating the value of expected overlap for cases when computing the analytical solution is not feasible. Chapter 5 shows how our framework performs on real data, using sub-sampling from microarray datasets to empirically verify the performance of our framework. Finally, in Chapter 6, we present concluding remarks and possible future venues for further exploring this problem.

Chapter 2

Background

In this chapter we will present the background information necessary to describe our framework. Section 2.1 provides a review on microarray studies, ranked gene lists, the "anomaly" of low overlap across different gene signatures, and studies that try to explain the reasons behind the anomaly. In Section 2.2, we will review the methodology of another study on estimating the overlap of gene signatures, and describe why we believe that one of the assumptions used in their model may not be accurate. Finally, in Section 2.3, we will briefly review results from overlap in rankings and order statistics in the literature, and explain why we have picked a different approach.

2.1 Gene Expression Profiling and Gene Signatures

Recent advances in high-throughput technologies have made expression profiling less expensive and more widely available. This has led to a multitude of studies focused on finding correlations between the expression levels of genes with the presence/absence of a given phenotype. The aim of such studies is to identify a list of genes that are *differentially expressed* across the case and control groups, i.e., genes whose expression levels are significantly different for patients presenting the phenotype versus those who do not. Such lists are obtained by ranking all the p genes in the array by a criterion, and taking the top-k elements of the ranked lists, where $k \ll p$. These lists are often called *ranked gene lists* or *gene signatures* [7].

The ranking criteria for gene signatures can be divided into three main categories [7,47]:

- Simple metrics such as p-value, q-value, Wilcoxon's rank sum test, or Pearson correlation coefficient [13].
- Modifications of t-statistic for multiple testing, such as SAM [70].
- Regularization methods, such as hierarchical Bayes method, could also be used to generate a regularized t-statistic [5,26,61].

Most studies then adjust the scores obtained for multiple testing, using techniques such as Bonferroni correction or false discovery rate (FDR) [19,50,66]. Here, we will not review methods for multiple testing correction, or techniques for setting the threshold to determine the value of k to use for the top-k genes, and will instead focus on the rankings obtained after sorting the list based on the ranking criterion.

Gene signatures obtained using the aforementioned process have various applications, which can be divided into the following two broad categories:

1. Gene signatures could be used as a preliminary step in building a prognostic or predictive classifier, i.e., as a *feature selection* method. Feature selection is useful when dealing with microarray data to avoid over-fitting. Other multivariate analyses, in addition to building classifiers, could also use gene signatures as input.

2. Gene signatures are also often reported as the final result of the study, after verifying that a non-trivial portion of the ranked lists is a priori known to be correlated with the phenotype. In such cases, the rest of the genes in the signature are also assumed to be correlated with the phenotype, thus suggesting new associations between genes and phenotypes. Hence, this class of studies is called *association studies*.

Association studies can be further broken down to two categories. Some studies use a priori biological and clinical knowledge in the process of obtaining the gene signature. This knowledge is used in either designing the microarray, to produce an array that is optimized to probe the expression levels of genes that are known or suspected to be associated with the phenotype, or in further filtering the ranked list by genes present on relevant biological pathways [27,45,48,67,69]. Other studies do not use any biological knowledge to pre-filter the genes. This latter type of association studies, also known as top-down association studies, is our main focus.

Two of the most cited examples of top-down association studies are Van't Veer et al. [72] and Wang et al. [74] signatures for predicting breast cancer, sometimes referred to as the 'Amsterdam' and 'Rotterdam' signatures, respectively. The Amsterdam signature consists of 70 genes, and was generated using samples from 96 patients; the Rotterdam signature has 76 genes, taken from samples from 286 patients. Classifiers based on each signature achieved good predictive results on samples from their own study[1]. However, the two signatures have only 3 genes in common [22]. Note that the two studies used different microarray platforms – Amsterdam was based on Rosetta, whereas Rotterdam used Affymetrix, meaning that the maximum possible overlap would be 55 genes – nevertheless, one might expect the two signatures, both shown to have good predictive performances on the same phenotype, to share a higher number of genes.

This "low overlap anomaly" is not limited to the above-mentioned studies. It has been observed repeatedly for multiple studies and for various phenotypes, as pointed out by Ein-Dor et al. [21]: *"Only 17 genes appeared in both the list of 456 genes of Sorlie et al. (2001) [62] and the 231 genes*

[1] The Amsterdam signature's performance was later verified on a larger 295-patient study [71]. Mammaprint, a diagnostic tool for assessing the risk of breast cancer metastasis is also based on the Amsterdam signature.

of vant Veer et al. (2002) [72]; merely 2 genes were shared between the sets of Sorlie et al. (2001) and Ramaswamy et al. (2003) [53]. Such disparity is not limited to breast cancer but characterizes other human disease datasets (Lossos et al., 2004) [3] such as schizophrenia (Miklos and Maleszka, 2004) [44]." Many other studies [7,22,37] found similar results. Table 2.1 summarizes a number of examples of occurrences of the overlap anomaly observed in the literature.

Table 2.1: Examples of the overlap anomaly observed in the literature. k_A and k_B denote the size of the top-k lists of studies A and B, respectively.

Phenotype (Study A Study B)	k_A	k_B	**Overlap**
Breast Cancer +/- ([72] [74])	70	76	3
Breast Cancer +/- ([62] [72])	456	231	17
Breast Cancer +/- ([53] [62])	128	456	2
Schizophrenia +/- ([28] [33])	89	49	8
Schizophrenia +/- ([44][a])	138	97	8
Large B-Cell Lymphoma +/- ([2] [58])	71	13	3
Large B-Cell Lymphoma +/- ([58] [55])	13	17	0

[a] List A taken from union of two studies and B from meta-analyses. See [44] for details

There have been a number of explanations for the lack of concordance between different gene signatures. Many point to possible human and equipment error, errors in pre-processing and mRNA extraction, using different chips and array technologies, and genuine differences between patients with regards to tumour size, age and demographics. While all these environmental factors could contribute to the lack of agreement between studies, one might wonder whether there could be little to no overlap in the absence of such factors. Ein-Dor et al. [21] and Michiels et al. [43] have explored that possibility, and reported that there is little overlap even in the case of sub-sampling from a single dataset, which rules out most of the environmental factors, assuming that all samples were taken under the same conditions. They concluded that many genes could be correlated with the phenotype, and that any subset of those genes would produce a good prognostic classifier. This implies that the inclusion of genes in the ranked list heavily depends on the random selection of the patients used to obtain the ranking. They also argue that this lack of agreement is the consequence of large confidence intervals around measured scores for each gene, which is the direct result of (relatively) low number of instances used in measuring the scores; therefore, the ranking of genes could change significantly in different sub-samples. Several other studies [4, 12, 35, 52, 68] have subsequently proposed methods to check the stability of the obtained ranked lists, mostly using sub-sampling or bootstrapping techniques.

This lack of agreement between signatures is not the only criticism raised against association studies. Some researchers are concerned about the lack of biological meaning of the lists obtained without biological guidance [18]. In addition, some criticize about overoptimistic results obtained from training classifiers based on gene signatures, mostly due to information leak and overtraining on test data [18]; see [20, 63] for examples of this issue[2]. Another issue is the multiplicity of ranking criteria. Boulesteix et al. [7] list 15 different ranking criteria used in the literature to obtain gene signatures. Two main issues arise from this large number of of available ranking criteria: a) Gene signatures are not stable across different ranking criteria, i.e., using different ranking criteria may produce very different signatures [34]. b) As a corollary of point (a), there is a possibility of publication bias, i.e., some groups may choose the ranking criterion a posteriori, based on whether genes that they consider biologically relevant appear on a signature produced after ranking by that specific criterion [7].

Various explanations and remedies have been proposed for the overlap anomaly. One approach to constructing more stable ranked lists is to combine several datasets on the same phenotype into a larger dataset, and using that new dataset to extract a more stable gene signature [32, 78]. Each such study is called a *meta-analysis*; see [31] for an overview and comparison of methods to generate meta-analyses. However, meta-analyses do not address the original instability issue of ranked lists, and therefore, we will not focus on them here.

One explanation for the low overlap issue was given by Zhang et al. [77]. Instead of measuring the overlap by counting the number of genes shared between the lists obtained from studies A and B, they suggest measuring the overlap by considering whether, for each gene x in study A's signature, the same gene x appears on study B's signature, or a gene y that is biologically correlated with x is present on study B's signature. They report that by using their proposed measurement, a larger overlap would be observed between gene signatures, compared to the more widely used approach of counting the number of shared genes. However, as pointed out by [7], this approach is not completely ranked based, as they use biological knowledge in computing the overlap of the two ranked lists.

Some have also suggested that genes from different signatures belong to the same underlying biological processes and pathways, and therefore, little overlap between different signatures is not an issue. Van't veer et al suggested this in [73], and it has been alluded to and accepted by many others (see [15,37,60] for examples of this). However, Dryer et al [18] have recently shown that such claims may be exaggerated. Their enrichment analyses on Amsterdam and Rotterdam signatures showed that only one pathway was shared between the two studies - that could pass statistical significance tests - namely cell proliferation, which was already known to be correlated with cancer before the advent of high-throughput profiling technologies. Thus, they concluded that proving the 'same-biological-processes' claim needs more quantitative evidence.

2.2 Estimating the Expected Overlap

While there have been several studies exploring the biological ramifications of the overlap anomaly, very few have tried to understand it from a mathematical point of view. To our knowledge, the study

[2] This issue is more concerned with applications of association studies in building classifiers, rather than a direct criticism about association studies.

by Ein-Dor et al [22] is the only attempt at building a mathematical model for the overlap of gene signatures. In their paper, they propose a Gaussian distribution for the probability distribution of f, the percentage of overlap between two gene signatures, as follows:

$$P_{n,\alpha}(f) = \frac{1}{\sqrt{2\pi}\Sigma_n} e^{-\frac{(f-f_n^*)^2}{2\Sigma_n^2}} \tag{2.1}$$

where $\alpha = \frac{k}{p}$, k is the size of the top-k list and p is the number of genes in the array, n is the number of patients, and $\Sigma_n \simeq \frac{1}{\sqrt{p}}$ for large values of p.

We are mostly interested in f^*, the expected value of percentage of overlap between the lists. Ein-Dor et al propose an approach that involves finding the saddle point of a complex set of equations, for numerically approximating the value[3] of f^*. More importantly, in their analysis they assume that $q(Z_t)$, the distribution of correlation scores of genes with respect to outcome, is symmetric and Gaussian. As proof, they present histograms of correlation scores with Gaussian distributions fitted to the histograms, and claim that the visual fit should provide enough evidence for accepting this assumption. Figure 2.1 shows an example of histogram of correlation scores with respect to ER+/ER- phenotype, using the data from the Rotterdam signature study, with a normal distribution superimposed.

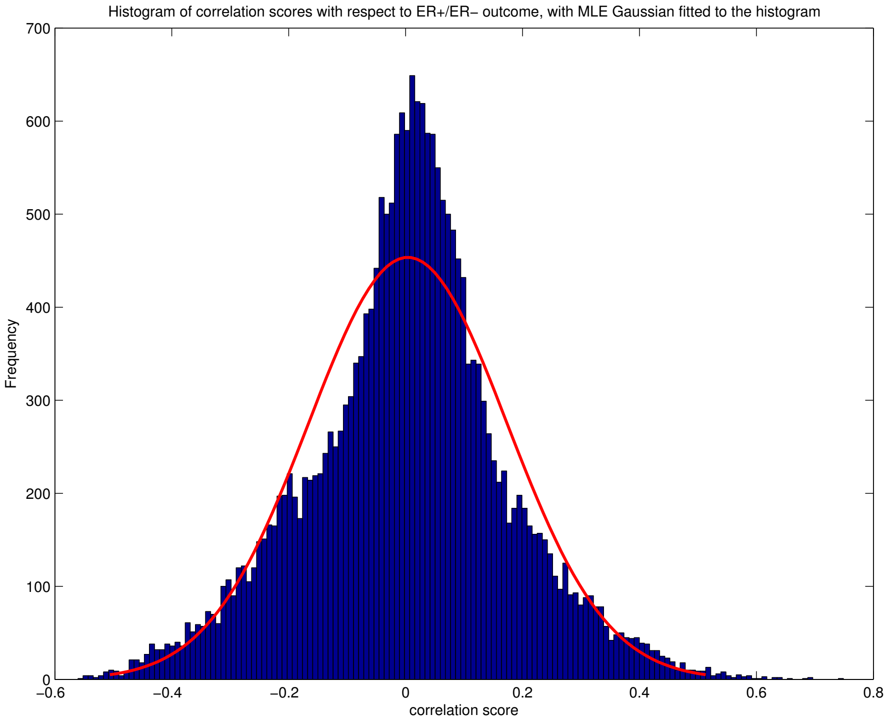
While the figure does show that the correlation scores follow a distribution that resembles a bell-curve, a visual histogram fit is not appropriate for accepting the normality assumption. To put this assumption under scrutiny, we analyzed the distribution of correlation scores from 8 microarray datasets [1, 10, 16, 30, 41, 46, 54, 74] for normality, using Lilliefors' normality test [40]. For all the datasets, we were able to reject the null hypothesis of normality with $p < 0.001$. This significantly challenges one of the key assumptions of Ein-Dor et al's analysis. In contrast, our approach, as will be detailed in Chapter 3, does not make any assumptions about the underlying distribution of correlation scores.

2.3 Ranked Lists and Order Statistics

The computer science literature has long studies the problem of extracting top-k lists – not just the ones obtained by ranking by correlation score – and measuring the overlap across rankings. Many studies focus on finding the overlap of various rankings, often to suggest methods to aggregate different rankings. In information retrieval literature, a number of studies focus exclusively on comparing the ranking results of search engines, see [6, 64] for recent examples. There is also an extensive literature on algorithms and metrics for comparing top-k lists, see [23] for an overview. However, all of the aforementioned studies focus on finding the overlap of two *observed* rankings, whereas our analysis provides a model to predict the level of noise on a score, and the effect of noise on the ranking of the elements in the list. In other words, we use this model to predict how different

[3] See the supplementary text on PNAS website [22].

Figure 2.1: Histogram of correlation scores taken from the Rotterdam signature dataset, with normal distribution superimposed. Original figure in supplementary text of [21], re-constructed using the data from [74].



an observation of a ranking would be from an unseen and unknown underlying true ranking, as opposed to modelling how two observations differ from each other. As such, most of the existing literature in this domain is not applicable to our problem.

Another possibly related area is order statistics and rank statistics [14]. Specifically, results from order statistics can be used to obtain probability density functions for the k-th smallest element of a list. Assuming that we are given a list of n observations, $X_1, \ldots, X_n$ from a probability distribution, with cumulative distribution function $F(.)$ and probability density function $f(.)$, the density function of $X_{(k)}$, the k-th smallest element in the list can be calculated as:

$$f_{X_{(k)}}(x) = \frac{n!}{(k-1)!(n-k)!} f(x)[F(x)]^{k-1}(1-F(x))^{n-k} \tag{2.2}$$

However, there are two reasons we could not use the above model: a) We cannot integrate our noise model into this model. Note that in the above formula, $F(.)$ and $f(.)$ are the cumulative and density functions of the distribution of the correlation scores, not the noise on the observed correlations, assuming that X_i models the observed correlation score of a gene. b) To use the above result, we need to know the distribution of correlation scores, and as shown in Appendix B, while we have suggestions for such a distribution, we prefer not to claim that it is, without a doubt, the underlying distribution of correlation scores. More importantly, as Chapter 3 shows, our model does not need any information about the distribution of correlation scores.

Chapter 3

Theoretical Analysis

In this chapter we will present a theoretical framework for computing the expected overlap of the top k genes[1] of two datasets. We will start by defining the notation necessary for working on the problem and formally define the problem, and list our assumptions in Section 3.1. We will derive a closed form formula for the expected overlap in Section 3.2. In Section 3.3, we will discuss the computational complexity of calculating the expected overlap, present some algorithms to efficiently carry out the calculations and show how to address the associated challenges. Finally, in Section 3.4, we will discuss how to generalize this formula to solve a larger class of problems.

3.1 Problem Definition and Notation

Assume that we are given a dataset $\mathcal{A} = [X_{\mathcal{A}}, C_{\mathcal{A}}]$, where $X_{\mathcal{A}}$ is an $n \times p$ matrix of real numbers, gene expression levels, for n patients and for p genes, and $C_{\mathcal{A}}$ is an $n \times 1$ vector, showing the status of each patient with respect to a phenotype. We refer to genes by their column index, e.g., $\mathcal{A}_1$ refers to the first gene of dataset $\mathcal{A}$, gene number 1, whose expression levels are present in the first column of matrix $X_{\mathcal{A}}$. Let $L_{\mathcal{A}} = [r_{\mathcal{A},1} \dots r_{\mathcal{A},p}]$ denote the absolute values of observed Pearson correlation coefficient for genes $\mathcal{A}_1, \dots, \mathcal{A}_p$ with respect to the given phenotype[2]. Let $\mathcal{R}(\mathcal{A}, i)$ denote the rank of $r_{\mathcal{A},i}$ after sorting $L_{\mathcal{A}}$ in descending order. As an example, consider the data matrix[3] presented in Table 3.1, where a 1 in the phenotype status column shows the presence of the phenotype, and a 0 signals the absence of the phenotype. For the dataset in Table 3.1, considering only the 6 samples and 5 genes shown in the table, we would have $L_{\mathcal{A}} = [0.0703, 0.1399, 0.5631, 0.6056, 0.1426]$. Therefore, we would have $\mathcal{R}(\mathcal{A}, 1) = 5, \mathcal{R}(\mathcal{A}, 2) = 4, \mathcal{R}(\mathcal{A}, 3) = 2, \mathcal{R}(\mathcal{A}, 4) = 1, \mathcal{R}(\mathcal{A}, 5) = 3$.

Now consider another dataset $\mathcal{B} = (X_{\mathcal{B}}, C_{\mathcal{B}})$, with $|X_{\mathcal{B}}| = n \times p$, where $C_{\mathcal{B}}$ shows the status of a different set of n patients with respect to the same phenotype as that of dataset $\mathcal{A}$. Continuing the example we had for dataset $\mathcal{A}$, we can compute the absolute values of correlation coefficients

[1] The theoretical framework presented in this chapter is not specific to genes. We only use "genes" and "patients" to help illustrate an application of our framework.

[2] From this point on, we will use "correlation score" to mean absolute value of correlation score.

[3] The expression values in this table were not taken from real microarray datasets, instead they were generated randomly from a uniform distribution. Expression values in real microarray datasets are not necessarily in the range $[0, 1]$.

Table 3.1: Example of a dataset $\mathcal{A}$ with p genes and n patients.

Patient ID	$\mathcal{A}_1$	$\mathcal{A}_2$	$\mathcal{A}_3$	$\mathcal{A}_4$	$\cdots$	$\mathcal{A}_p$	Phenotype status
1	0.8147	0.2785	0.9572	0.7922	$\cdots$	0.6787	1
2	0.9058	0.5469	0.4854	0.9595	$\cdots$	0.7577	1
3	0.1270	0.9575	0.8003	0.6557	$\cdots$	0.7431	0
4	0.9134	0.9649	0.1419	0.0357	$\cdots$	0.3922	0
5	0.6324	0.1576	0.4218	0.8491	$\cdots$	0.6555	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
n	0.0975	0.9706	0.9157	0.9340	$\cdots$	0.1712	1

Table 3.2: Example of absolute values of correlation coefficients, and ranks of genes for two datasets $\mathcal{A}$ and $\mathcal{B}$.

Feature ID (i)	$r_{\mathcal{A},i}$	$\mathcal{R}(\mathcal{A},i)$	$r_{\mathcal{B},i}$	$\mathcal{R}(\mathcal{B},i)$
1	0.0703	5	0.1712	5
2	0.1399	4	0.4555	3
3	0.5631	2	0.7577	1
4	0.6056	1	0.6431	2
5	0.1426	3	0.3922	4

for each gene with respect to outcome, and rank the genes based on correlation scores. Table 3.2 shows an example of $L_{\mathcal{B}}$ and the ranks of genes for a dataset $\mathcal{B}$.

Let $Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k)$ denote the number of shared genes, in the list of top-k genes of datasets $\mathcal{A}$ and $\mathcal{B}$, after ranking the genes by absolute value of correlation score. That is:

$$Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k) = \sum_{i=1}^{p} I(\mathcal{R}(\mathcal{A}, i) \leq k \wedge \mathcal{R}(\mathcal{B}, i) \leq k)$$

where I is the indicator function. For the example shown in Table 3.2, for $k = 3$, we have $Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, 3) = 2$, i.e., 2 genes are shared between the top 3 genes of the two datasets.. Our goal is to compute the expected value of $Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k)$, for two n-patient studies, with the same set of p genes, over the same phenotype.

To compute the expected overlap, we will assume that the correlation scores for the p genes of the two datasets are two different draws of the same unknown underlying distribution. Specifically, we assume the following:

Assumption 1 The p correlation coefficients, associated with the p features, can be modelled with independent random variables $X_1, \ldots, X_p$, where $X_i \sim N(\mu_i, \sigma^2)$, where μ_i is the true mean of observed correlation scores for gene number i, and $\sigma = \frac{1}{\sqrt{n-3}}$. This holds under mild assumptions [24, 25]. Note that the resulting X_i's are not in the range $[0, 1]$, but instead lie in $[0, +\infty)$.

Assumption 2 We are given a vector $\vec{r} = [r_1, \ldots, r_p]$, of observations of correlation scores for the

Table 3.3: A Summary of Notation. Dataset $\mathcal{A}$ is used as an example and the same notation applies to any dataset.

Notation	Meaning
$EOv(n, k, p, \vec{\mu})$	expected overlap between top-k genes of two n-patients studies with p genes, with correlation vector $\vec{\mu}$
$L_{\mathcal{A}}$	List of absolute values of correlation coefficients for genes in dataset $\mathcal{A}$
$Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k)$	overlap between top-k genes of datasets $\mathcal{A}$ and $\mathcal{B}$, with respect to correlation score
$\mathcal{A}_i$	gene number i in dataset $\mathcal{A}$
$P_k(ov = r)$	probability of having $Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k) = r$
$P_k(ov \geq r)$	probability of having $Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k) \geq r$
$P_{n,k}(X_1, \ldots, X_r)$	probability of observing genes associated with $X_1, \ldots, X_r$ in top-k genes of a dataset with n samples
$r_{\mathcal{A},i}$	the correlation score of gene number i in dataset $\mathcal{A}$
$\mathcal{R}(\mathcal{A}, i)$	the rank of gene $\mathcal{A}_i$ after sorting $L_{\mathcal{A}}$ in descending order

p genes for the phenotype of datasets $\mathcal{A}$ and $\mathcal{B}$, such that $\mu_i = \tanh^{-1}(r_i)$ could be treated as a good estimate for the true mean of the Gaussian random variable for gene i, as stated in Assumption 1.[4]

Let $P_k(ov = r)$ be the probability of observing $Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k) = r$, i.e., having an overlap of exactly size r in the top-k genes of the two studies. Similarly, let $P_k(ov \geq r)$ be the probability of observing $Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k) \geq r$. Let $P_{n,k}(X_1, \ldots, X_r)$ be the probability of observing genes associated with random variables $X_1, \ldots, X_r$ in the top-k genes of a given dataset with n patients. Since both input datasets have the same number of patients, we will suppress n from the subscripts, and just write $P_k(X_1, \ldots, X_r)$.

Our goal is to compute $EOv(n, k, p, \vec{\mu})$: A function that takes the number of patients n, size of the top-k list, number of genes[5] p, and a vector of true means of correlation scores $\vec{\mu}$ as input, and returns a number, the expected overlap between the top-k ranked genes of datasets $\mathcal{A}(X, C)$ and $\mathcal{B}(X', C')$, which corresponds to the expected value of $Ov(L_{\mathcal{A}}, L_{\mathcal{B}}, k)$.

Table 3.3 summarizes the notations used throughout this dissertation.

3.2 Computing the Expected Overlap

Our goal is to compute the expected overlap of top-k genes of two datasets, $EOv(k)$, which corresponds to:

$$EOv(k) = \sum_{r=1}^{k} r \times P_k(ov = r) = \sum_{r=1}^{k} P_k(ov \geq r) \tag{3.1}$$

where

$$P_k(ov \geq r) = \sum_{\{x_1 \ldots x_k\} \subset \{X_1 \ldots X_p\}} P_k(x_1, \ldots, x_k)[\sum_{j=r}^{k} (-1)^{j+r} \sum_{\{y_1 \ldots y_j\} \subset \{x_1 \ldots x_k\}} P_k(y_1, \ldots, y_j)] \tag{3.2}$$

[4] We will discuss how we can obtain such a vector in Chapter 5.

[5] p is also taken as input implicitly as the length of $\vec{\mu}$, and is only stated here for emphasis.

To understand (3.2), consider the example from Table 3.1, with $p = 5$ and $k = 3$. We are interested in computing $P_3(ov \geq 1)$. We also already know that the answer should be 1, as there is at least 1 element in common between two draws of size 3 from a dataset of size 5. We have:

$$\begin{aligned} P_3(ov \geq 1) = \sum_{\{x_1,x_2,x_3\}\subset\{X_1,\ldots,X_5\}} P_3(x_1,x_2,x_3)[&P_3(x_1)+P_3(x_2)+P_3(x_3) \\ &- P_3(x_1,x_2) - P_3(x_1,x_3) - P_3(x_2,x_3) \\ &+ P_3(x_1,x_2,x_3)] \end{aligned}$$

where X_i corresponds to the random variable associated with gene $\mathcal{A}_i$, $i \in \{1,\ldots,5\}$. To simplify writing the expansion of the above formula, let $sp_3(y_1,\ldots,y_j) = \sum_{i=1}^{j} P_3(y_i)$ denote the sum of probabilities of observing 3-tuples $y_1,\ldots,y_j$, where each $y_i = (abc)$ for some $\{a,b,c\} \subset \{1,\ldots,5\}$. For example, $sp_3(123,234) = P_3(X_1,X_2,X_3) + P_3(X_2,X_3,X_4)$. Therefore, focusing only on the term where $x_1 = X_1, x_2 = X_2, x_3 = X_3$, the corresponding term from the above equation would be:

$$\begin{aligned} P_3(ov \geq 1) = \ldots + sp_3(123)[&sp_3(123,124,125,134,135,145)+ \\ &sp_3(123,124,125,234,235,245) + sp_3(123,134,135,234,235,345) \\ &- sp_3(123,124,125) - sp_3(123,134,135) - sp_3(123,234,235) \\ &+ sp_3(123)] \end{aligned} \tag{3.3}$$

since the probability of observing 3-tuples as the top 3 genes is disjoint, the above equation would become:

$$P_3(ov \geq 1) = \ldots + P_3(123)[1]$$

since $\sum_{x_1,x_2,x_3\subset\{X_1,\ldots,X_5\}} P_3(x_1,x_2,x_3) = 1$, $P_3(ov \geq 1) = 1$.

As the example shows, the signs of odd and even terms oscillate as r grows in (3.2). Substituting $P_k(ov \geq r)$ in (3.1) and some simple algebra, yields:

$$EOv(k) = \sum_{\{x_1\ldots x_k\}\subset\{X_1\ldots X_p\}} P_k(x_1,\ldots,x_k)[\sum_{\substack{1\leq j\leq k \\ j\text{is odd}}} \sum_{\{y_1\ldots y_j\}\subset\{x_1\ldots x_k\}} P_k(y_1,\ldots,y_j)] \tag{3.4}$$

To check the validity of 3.4, we can check $EOv(1)$ as an example:

$$EOv(1) = \sum_{x_1\subset\{X_1\ldots X_p\}} P_1(x_1)[\sum_{\substack{j\leq 1 \\ j\text{is odd}}} \sum_{y_1\subset\{x_1\}} P_1(y_1)] = \sum_{x\subset\{X_1,\ldots,X_p\}} P_1(x)^2$$

which, as expected, is adding up the probabilities for each gene to appear in both datasets as the top gene. This exploits the independence claim: the chance of a specific gene being the top gene of dataset A is independent of it being top in dataset B.

To compute $P_k(X_1, \ldots, X_k)$, we will start with the simpler case of $P_1(X_1) = P(X_1 > X_2, X_1 > X_3, \ldots, X_1 > X_p)$ and then generalize the formula[6]. Let $A = \max\{X_2, \ldots, X_p\}$. Assuming that X_i 's are independent, and $X_i \sim N(\mu_i, \sigma)$, we can write:

$$P(A \leq a) = F_A(a) = \prod_{i=2}^{p} \Phi(\frac{a - \mu_i}{\sigma})$$

$$f_A(a) = \frac{dF_A(a)}{da} = F_A(a) \sum_{i=2}^{p} \frac{\phi(\frac{a-\mu_i}{\sigma})}{\Phi(\frac{a-\mu_i}{\sigma})}$$

where $\phi(.)$ and $\Phi(.)$ are normal distribution probability density and cumulative distribution functions respectively. We will therefore have:

$$\begin{aligned}
P_1(X_1) &= \int_{-\infty}^{+\infty} P(X_1 > a) f_A(a) da \\
&= \int_{-\infty}^{+\infty} (1 - \Phi(\frac{a - \mu_1}{\sigma})) f_A(a) da \\
&= \int_{-\infty}^{+\infty} f_A(a) da - \int_{-\infty}^{+\infty} \Phi(\frac{a - \mu_1}{\sigma}) f_A(a) da \\
&= 1 - \int_{0}^{+\infty} \Phi(\frac{a - \mu_1}{\sigma}) f_A(a) da
\end{aligned}$$

The last step is correct since the domain of a is $[0, +\infty)$, as a ranges over all possible values of $\tanh^{-1}(r_i)$, where r_i is in $[0, 1]$.

In general, for any given set T of size k of indices of elements in $\{X_1 \ldots X_p\}$ to consider as top genes, we can define $T' = \{1 \ldots p\} - T$ and derive the general form for $P_k(X_T)$ as follows:

$$A = \max(X_{T'}), \quad F_A(a) = \prod_{i \in T'} \Phi(\frac{a - \mu_i}{\sigma}), \quad f_A(a) = \frac{dF_A(a)}{da}$$

$$P_k(X_T) = 1 - \int_{0}^{+\infty} \prod_{i \in T} \Phi(\frac{a - \mu_i}{\sigma}) f_A(a) da \tag{3.5}$$

where $X_T = \bigcup_{k \in T} \{X_k\}$, and similarly $X_{T'} = \bigcup_{k \in T'} \{X_k\}$.

The final step in deriving (3.4) is computing $P_k(X_1, \ldots, X_r)$ when $r < k$. This probability can be computed by marginalization and summing up the probabilities of $P_k(\{X_1, \ldots, X_r\} \cup \{X_{r+1} \ldots X_k\})$ for all possible disjoint sets of $\{X_{r+1} \ldots X_k\}$.

3.3 Implementation

3.3.1 Implementation challenges

The main obstacle in the way of implementing a program to compute (3.4) is the combinatorial nature of the problem. To get an estimate of the computational complexity of the problem, we revisit equation (3.4):

[6] I am indebted to Prof. P. Hooper for this proof.

$$EOv(k) = \sum_{\{x_1 \ldots x_k\} \subset \{X_1 \ldots X_p\}} \underbrace{P_k(x_1, \ldots, x_k)}_{A} \underbrace{[\sum_{\substack{1 \le j \le k \\ j \text{is odd}}} \sum_{\{y_1 \ldots y_j\} \subset \{x_1 \ldots x_k\}} P_k(y_1, \ldots, y_j)]}_{B}$$

Part A of the above formula computes the probability of tuple $[x_1, \ldots, x_k]$ being in the top-k genes of the first dataset, and part B computes the probability of $[x_1, \ldots, x_k]$ being in the top-k of the second dataset. Note that all the marginal probabilities in part B can be expressed as a sum of probabilities computed in part A. More specifically, for any $P_k(y_1 \ldots y_j)$, we should find all $\binom{k}{k-j}$ sets of $y_{r+1} \ldots y_k$, and compute $\sum_{y_{j+1} \ldots y_k} P_k(y_1 \ldots y_j, y_{j+1} \ldots y_k)$. Therefore, the computational complexity of computing $EOv(k)$, can be estimated as:

$$\begin{aligned}
C(k) &= \binom{p}{k} \times [O(p) + \binom{k}{1}^2 + \binom{k}{3}^2 + \cdots + \binom{k}{k}^2] \\
&\le \binom{p}{k} \times [O(p) + \binom{2k}{k}] \\
&\simeq O(p^k) \times [O(p) + O(4^k)] \\
&= O(p^{k+1})
\end{aligned}$$

where p is typically in the range of $20,000$ to $50,000$ for microarrays [7]. As k grows, computing $EOv(k)$ using our analytical solution quickly becomes computationally infeasible.

3.3.2 Efficient Implementation

Even though computing $EOv(k)$ is infeasible for large k's, we can compute it for smaller k's. This will be particularly useful for validating the results of simulations. To compute $EOv(k)$, we need to numerically approximate the integral (3.5). We used MATLAB's implementation of adaptive Gauss-Kronrod quadrature method [56], which is particularly useful for approximating integrals of form $\int_a^{+\infty} g(x)dx$ when $g(x)$ decays rapidly, which is the case in our problem. We will use $quadrature(g(x), [a, b])$ to denote the Gauss-Kronrod method to approximate the value of integrating $g(x)$ over the interval $[a, b]$.

As shown in the previous section, the naive implementation Algorithm 3.1 shows the pseudocode of the algorithm that computes $EOv(2)$. In all the algorithms that follow, we assume that $\vec{r} = [r_1, \ldots, r_p]$ is sorted in descending order. This will help to significantly cut the number of calculations needed to estimate the value of $EOv(k)$, since if $(x_i, x_{i+1}, \ldots, x_j)$ are sorted in decreasing order of their mean values, $\mu_a = \tanh^{-1}(r_a)$, the value of $P_k(x_i, x_{i+1}, \ldots, x_j)$ decreases as we move towards greater values of i. We will use two values to prune the calculations:

1. **Min Value** ϵ: As detailed in the algorithms, we are summing over a large number of probabilities, $P_k(.)$, to determine the value of $EOv(k)$. When the result of computing the current

[7] In SNP studies, this number grows to $\sim 500,000$.

$P_k(.)$ falls below a specified minimum value, ϵ, we will prune the rest of the current computation. If this happens in the outer-most loop of the algorithm, the program will terminate and return the current result. Choosing ϵ depends on the precision one wishes for the estimations to have. For example, when computing $EOv(1)$, having $\epsilon = 10^{-4}$ ignores[8] adding the results smaller than 10^{-8}. We will specify the value of ϵ in each experiment that uses our analytical solution in the chapter that follow.

2. **Max Index** κ: We also set a max index value, κ, to only consider the top κ genes, corresponding to $[r_1, \ldots, r_\kappa]$ in $\vec{r} = [r_1 \ldots, r_p]$, after sorting $\vec{r}$. While a gene whose correlation is relatively small does have a non-zero chance of having an observed value that is large enough for it to appear in the top-k list, those who are ranked abobe κ are very unlikely to be observed in the top-k, and therefore we can ignore them in our computations.

The following code sections review the algorithms used in computing $EOv(2)$. We chose $EOv(2)$ as the code for $EOv(1)$ may not be general enough to explain our implementation method. Computing $EOv(k)$ for higher values of k can be easily generalized from the algorithms we present for $EOv(2)$.

Although κ and ϵ do not ultimately affect the exponential nature of the problem, they do significantly cut the number of calculations needed. As an example, consider computing $EOv(2)$ for a dataset with $p = 50,000$. Without using κ and ϵ, we would have to carry out $O(p \times (p + p^2)) = O(p^3)$ computations, which is $O(50,000^3)$ calculations. When using $\kappa = 1000$ and $\epsilon = 10^{-5}$, in practice, we computed $EOv(2)$ with $O(20 \times (\kappa + \kappa^2)) = O(\kappa^2) = O(1000^2)$ terms [9]. Therefore, we were able to speed up the computations by 8 orders of magnitude without sacrificing accuracy.

As shown previously, the runtime complexity of $EOv(k)$ is exponential in k. Although we have presented optimized algorithms that could compute $EOv(k)$ for small values of k, computing the expected overlap for large values of k is not feasible with this approach. There are a number of possible solutions to this issue:

1. **Exact solution** Formula 3.4 may be solvable by finding a relation between consecutive terms of $EOv(k)$. That is, if there was an inductive relation between $P_k(ov \geq r)$ and $P_k(ov \geq r+1)$ we could exploit that relation to devise a dynamic programming solution to $EOv(k)$. Unfortunately, we could not find an explicit inductive relation.

2. **Deterministic approximations** As shown in Appendix B, the distribution of $\vec{r}$ matches a power-law. By leveraging this fact, we could cut the number of terms we need to compute and calculate deterministic approximations for $EOv(k)$. There are, however, two issues with this approach: a) Although for most datasets, $\vec{r}$ follows a power-law $r_i \sim ai^{-\alpha}$ with a fixed α

[8] Since $EOv(1) = \sum_{X_i} P_1(X_i)^2$

[9] 20 is an estimate of the average number of iterations before the outer loop reached an $\epsilon < 10^{-5}$ for the sample datasets we used.

Algorithm 3.1 $EOv_2(\vec{r}, \sigma, \epsilon, \kappa)$

```
Require: $\vec{r}$ be sorted in descending order
  // Let $lookup[i][j]$ store computed values of $P_2(X_i, X_j)$.
  // Let $marginal[i]$ store computed values of $P_2(X_i)$.
  $result \leftarrow 0$
  for $i = 1 \dots \kappa$ do
    if $i < \kappa$ then
      $canary \leftarrow lookup[i][i+1]$ {To see if we need to proceed further}
      if $canary$ has not been assigned then
        $lookup[i][i+1] \leftarrow canary \leftarrow 1 - quadrature($Integrand$(x, i, i+1, \vec{r}, \sigma, \kappa), [0, +\infty))$
      end if
      if $canary \leq \epsilon$ then
        return $result$
      end if
    end if
    for $j = i+1 \dots \kappa$ do
      if $lookup[i][j]$ does not exist then
        $lookup[i][j] \leftarrow 1 - quadrature($Integrand$(x, i, j, \vec{r}, \sigma, \kappa), [0, +\infty))$
      end if
      for all $c \in \{i, j\}$ do
        if $marginal[c]$ does not exist then
          $marginal[c] \leftarrow$ getMarginal$(c, lookup, \vec{r}, \sigma, \epsilon, \kappa)$ {getMarginal() also populates
          $lookup$ in the process}
        end if
      end for
      $current \leftarrow lookup[i][j] \times (marginal[i] + marginal[j])$
      if $current \leq \epsilon$ then
        break out of inner-loop, proceed to next $i$
      end if
      $result \leftarrow result + current$
    end for
  end for
  return $result$
```

Algorithm 3.2 getMarginal$(i, lookup, \vec{r}, \sigma, \epsilon, \kappa)$

```
  $sum \leftarrow 0$
  for all $c \in [1 \dots \kappa] - \{i\}$ do
    if $lookup[i][c]$ does not exist then
      $lookup[i][c] \leftarrow 1 - quadrature($Integrand$(x, i, c, \vec{r}, \sigma, \kappa), [0, +\infty))$
    end if
    $sum \leftarrow sum + lookup[i][c]$
    if $lookup[i][c] \leq \epsilon$ then
      break
    end if
  end for
  return $sum$
```

Algorithm 3.3 Integrand($x, i, j, \vec{r}, \sigma, \kappa$)

```
Require: $\vec{r}$ be sorted in descending order
  for all $j \in \{1 \dots p\}$ do
    $\mu_j = \tanh^{-1}(r_j)$
  end for
  $F_A \leftarrow 1$
  $sum \leftarrow 0$
  for all $c \in [1 : \kappa] - \{i, j\}$ do
    $cdf \leftarrow \Phi(\frac{(x-\mu_c)}{\sigma})$
    $F_A \leftarrow F_A \times cdf$
    $sum \leftarrow sum + \frac{\phi(\frac{x-\mu_c}{\sigma})}{cdf}$
  end for
  $f_a \leftarrow F_A \times sum$
  $p_{xi} \leftarrow \Phi(\frac{x-\mu_i}{\sigma})$
  $p_{xj} \leftarrow \Phi(\frac{x-\mu_j}{\sigma})$
  return $p_{xi} \times p_{xj} \times f_a$
```

parameter, a tends to have a wide range across datasets. b) Using this class of approximations significantly cuts the number of calculations, which affects both the unseen constant and p in $O(p^{k+1})$, however, the runtime complexity of the solution ultimately remains exponential in k.

3. **Stochastic approximations** In particular, Monte Carlo simulations can be used to stochastically approximate the value of $EOv(k)$. This approach is the focus of Chapter 4.

3.4 Extensions

Although we made a number of assumptions in computing the expected overlap, some can be relaxed and the derived formula can be generalized to accommodate the relaxed assumptions.

We assumed that datasets $\mathcal{A}$ and $\mathcal{B}$ have the same number of patients, n. This assumption is better suited for evaluating and validating the results, and as Chapter 4 shows, makes the simulations easier. However, there is no restriction in the formula against having two different number of patients, $n_{\mathcal{A}}$ and $n_{\mathcal{B}}$. Equation (3.4) can be re-written to allow for different number of patients as follows:

$$EOv(k, n_{\mathcal{A}}, n_{\mathcal{B}}) = \sum_{\{x_1 \dots x_k\} \subset \{X_1 \dots X_p\}} \underbrace{P_{n_{\mathcal{A}},k}(x_1, \dots, x_k)}_{\mathcal{A}} \underbrace{[\sum_{\substack{1 \leq j \leq k \\ j \text{is odd}}} \sum_{\{y_1 \dots y_j\} \subset \{x_1 \dots x_k\}} P_{n_{\mathcal{B}},k}(y_1, \dots, y_j)]}_{\mathcal{B}} \tag{3.6}$$

where computing $P_{n_{\mathcal{A}},k}(X_T)$, $T \subset \{1 \dots k\}$ would be easily possible by changing σ in the original equation (3.5) to $\sigma_{n_{\mathcal{A}}} = \frac{1}{\sqrt{n_{\mathcal{A}}-3}}$, and similarly for $n_{\mathcal{B}}$.

While relaxing this assumption would lead to a small change in the formula, it will translate to a greater change in the implementation. We can no longer assume that the marginal probabilities in part $\mathcal{B}$ of equation (3.6) are the sum of some of the terms in part $\mathcal{A}$, as the σ's are now different. This will adversely affect the runtime complexity of the implementation, making computing the formula even less feasible.

We can also relax assumption 1. In fact, we can relax all the assumptions about underlying distributions and the ranking criterion, as the formula does not depend on any of those assumptions and is able to predict the expected overlap of any two lists ranked by the same criterion, as long as the ranked features can be modelled with random variables X_i such that all X_i are drawn from the same probability distribution with density function $f(.)$ and cumulative distribution function $F(.)$. To generalize the formula for this relaxed assumption, equation 3.5 should be re-written as follows:

$$A = \max(X_{T'}), \quad F_A(a) = \prod_{i \in T'} F(a; r_i), \quad f_A(a) = F_A(a) \sum_{i \in T'} \frac{f(a; r_i)}{F(a; r_i)}$$

$$P_k(X_T) = 1 - \int_{-\infty}^{+\infty} \prod_{i \in T} F(a; r_i) f_A(a) da \tag{3.7}$$

where $F(a; r_i) = P(X_i \leq a)$ denotes the cumulative distribution function of the probability distribution used to model X_i with parameter r_i.

Chapter 4

Simulation

In this chapter we will present a framework for stochastically approximating the expected overlap between two lists ranked by correlation to outcome. Section 4.1 provides a brief overview of Monte Carlo simulations. Section 4.3 describes our simulation algorithm, its parameters and workarounds to boost its performance. In Section 4.4, we will provide a comparison of the results of our simulation with the results of the analytical framework we presented in Chapter 3. Finally, in Section 4.5, we will discuss some of the results of the simulation, compare them with what we expect from the theoretical framework and the real overlap problem, and discuss possible curve fitting methods to extend our analytical results.

4.1 Monte Carlo Simulations

Monte Carlo methods are a widely used class of computational algorithms used in simulating the behaviour of systems with many degrees of freedom. Monte Carlo simulations have applications in physics, economics, computer science and many other fields. A Monte Carlo simulation involves playing 'games of chance', i.e., random draws of a complex system one wishes to model. If done correctly and over many random draws, a Monte Carlo simulation can provide a good approximation of models that might be impossible or infeasible to compute using analytical methods. Monte Carlo methods are especially useful in cases for which we know the underlying probability distributions, but cannot compute the exact answer because of the computational complexity of the calculations. This is the case for our problem: as the underlying noise model is Gaussian, computing the exact expected overlap proved to be infeasible. In the following sections we will describe the steps we used to use a class of Monte Carlo simulations to stochastically approximate the value of expected overlap.

For more background and information on Monte Carlo methods and their applications, see [29, 42].

4.2 Generative Model

Recall from Chapter 3 that our framework models the p genes from an n-patient study with random variables $X_1, \dots, X_p$, where $X_i \sim N(\mu_i, \sigma)$, where $\sigma = \frac{1}{\sqrt{n-3}}$. To model each X_i, we need μ_i, the true mean of X_i. In this chapter, we assume that we are given a vector of observed correlation scores, $\vec{r} = [r_1, \dots, r_p]$, such that $\mu_i = \tanh^{-1}(r_i)$ could be regarded as a good estimate for the mean value of X_i. The main input to our simulation algorithm is therefore $\vec{\mu} = [\mu_1, \dots, \mu_p]$, where $\mu_i = \tanh^{-1}(r_i)$. We will then take p random draws, $\vec{Z} = [Z_1, \dots, Z_p]$, where $Z_i \sim N(\mu_i, \sigma)$, to get a noisy draw of Fisher transforms of correlation scores. We will then compute $\vec{D} = \tanh(\vec{Z})$, called a *replicate*. Each replicate is a possible vector of correlation scores, that would be the result of a study with n patients and p genes. We will repeat this procedure many times, and then calculate the overlap of top-k genes across the replicates, to estimate the value of $EOv(k)$. Section 4.3 describes this procedure in detail.

4.3 Simulation Algorithm

In this section we will describe an algorithm to simulate our model for the expected overlap in the top-k genes of two datasets $\mathcal{A}(X_1, C_1)$ and $\mathcal{B}(X_2, C_2)$, where $|X_1| = |X_2| = n \times p$, and C_1 and C_2 provide class labels for the same phenotype. As described in Section 4.2, we assume that we are given $\vec{\mu} = [\mu_1, \dots, \mu_p]$, such that μ_i can be treated as a good estimate for the mean value of the Gaussian noise around the measured correlation score for gene number i. Algorithm 4.1 summarizes our simulation process.

The straightforward implementation of finding the overlap between two lists of equal size l take $O(l)$ to run. The runtime complexity of Algorithm 4.1 would therefore be $O(N_t + N_t \times N_s \times k_{max}^2) = O(N_t \times N_s \times k_{max}^2)$. However, we exploit an inductive relation between overlap in k and $k+1$ top genes in two datasets to reduce the runtime to $O(N_t \times N_s \times k_{max})$. For more details please see Appendix A.

4.4 Comparison with Analytical Results

To verify that the result from the stochastic approximation and exact formula agree, we computed $EOv(n, 1)$ and $EOv(n, 2)$ for several datasets with various number of patients and compared the results of the two methods. We chose the first two terms, as we could analytically compute them despite the exponential runtime complexity of computing the analytical solution. Table 4.1 and Table 4.2 show the comparisons for $EOv(1)$ and $EOv(2)$, respectively. As the results suggest, there is strong agreement between our simulations and theoretical analysis. For all the experiments. except the synthesized dataset, we obtained $\vec{r}$ by calculating the correlation coefficients using all the samples in the given datasets. For more information on how we synthesized a dataset, see Appendix B.

Algorithm 4.1 Stochastic approximation of $EOv(i)$ for $i \in \{1, \dots, k_{max}\}$

```
**Input:** $\vec{\mu}$: Vector of true mean values of random variables $X_1, \dots, X_p$, number of patients $n$, number of replicates $N_t$, maximum number of top-$k$ genes to consider $k_{max}$, and number of replicates to consider for computing the overlap $N_s$, where $N_s < N_t$.
for trial $\in \{1, \dots, N_t\}$ do
  for $i \in \{1, \dots, p\}$ do
    Draw $Z_i \sim N(\mu_i, \frac{1}{\sqrt{n-3}})$
    $\rho_i \leftarrow |\tanh(Z_i)|$
  end for
  $D_{\text{trial}} \leftarrow \vec{\rho}$
end for
for $i \in \{1, \dots, N_t\}$ do
  $\vec{S} \leftarrow rand(N_s, [1 : N_t] - \{i\})$ {$\vec{S}$ stores $N_s$ random indices from the range $[1, N_t]$, excluding $i$}
  for $k \in \{1, \dots, k_{max}\}$ do
    Let $O[i][k]$ store the average overlap of $D_i$ with the $N_s$ replicates $D_{S_1}, \dots, D_{S_{N_s}}$ in the top-$k$ genes.
  end for
end for
for $k \in \{1, \dots, k_{max}\}$ do
  $mean[k] \leftarrow$ average($O[1 : N_t][k]$)
  $std[k] \leftarrow$ std($O[1 : N_t][k]$)
end for
return $mean$ and $std$
```

Table 4.1: Comparison of simulation and analytical results for $EOv(1)$.

Dataset	n	**Analytical Result**	**Simulation**
Synthesized	250	0.5374	0.5382
Synthesized	100	0.4219	0.4209
Synthesized	50	0.3264	0.3183
GSE3744 [1]	47	0.1294	0.1262
GSE3744	150	0.2012	0.2027
GSE3744	250	0.2318	0.2317
GSE10780 [10]	185	0.0074	0.0050
GSE10780	300	0.0191	0.0441
GSE22544 [30]	20	0.0430	0.0441
GSE22544	150	0.1766	0.1723
GSE7904 [54]	62	0.1020	0.1027
GSE7904	150	0.1835	0.1819

For the experiments on $EOv(1)$ we used the following parameters for theoretical analysis and simulation: $\kappa = 1000$, $\epsilon = 10^{-4}$, $N_t = 5,000$, $N_s = 400$. For experiments on $EOv(2)$ we used $\kappa = 1000$, $\epsilon = 10^{-5}$, $N_t = 10,000$, $N_s = 300$.

Table 4.2: Comparison of simulation and analytical results for $EOv(2)$.

Dataset	n	**Analytical Result**	**Simulation**
Synthesized	250	1.5466	1.5538
Synthesized	100	1.3464	1.3513
Synthesized	50	0.9584	0.9690
GSE3744	47	0.4469	0.4448
GSE3744	150	0.7129	0.7107
GSE3744	250	0.8151	0.8154
GSE10780	185	0.006	0.014
GSE10780	300	0.034	0.046
GSE22544	20	0.1419	0.1491
GSE22544	150	0.5579	0.5548
GSE7904	62	0.5145	0.5070
GSE7904	150	0.7129	0.7146

4.5 Analysing the Simulation Results

In this section we will present results from running the simulation on various datasets. For all the experiments in this section, we used microarray datasets to get $\vec{r}$, and then set n and k_{max} as needed according to the experimental setup. The datasets used are publicly available NCBI datasets GSE2034 [74][1], GSE3494 [46], GSE6532 [41] and GSE7390 [16]. These datasets have a number of phenotypes; for the sake of consistency, we used the same phenotype of ER+/ER- [2] for all the datasets.

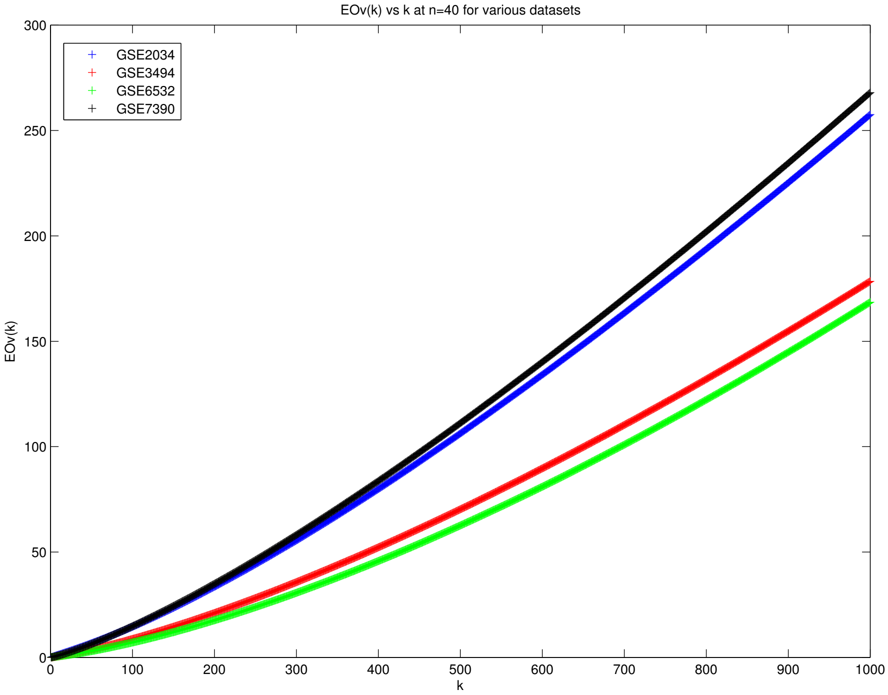
Figure 4.1 shows the results of computing $EOv(k)$ at $n = 40$, with $k_{max} = 1000$. As expected, $EOv(k)$ grows as k grows. The first derivative of $EOv(k)$ also grows with k over this range, which may describe how we can ultimately have $EOv(p) = p$. In addition, we observed that the variance of $EOv(k)$ increases as k grows over this range. This can be attributed to the higher effect of noise on the order of lower-ranked genes, which makes the list of top-k genes less consistent – i.e., with a lower number of shared genes among different replicates – as we move towards higher values of k. Figure 4.4 shows the $EOv(k)$ vs k plot for dataset GSE2034 with $k_{max} = 100$ from the same experiment, with error bars representing one standard deviation around the mean.

Figure 4.2 shows $EOv(.)$ as a function of n. To get this plot, we used a fixed value of $k = 20$. As the figure suggests, a higher number of patients decreases the noise, which makes the top-k list more consistent and leads to a higher overlap. However, some previous studies [37] have reported that the increase in the size of overlap between ranked lists is linear in n. While this statement may be true for small values of n, it does not hold for larger values, and the growth dampens as we move

[1] GSE2034 is the dataset from which the Rotterdam signature was generated.

[2] Estrogen-Receptor positive/negative.

Figure 4.1: $EOv(k)$ versus k with $k_{max} = 1000$ and $n = 40$

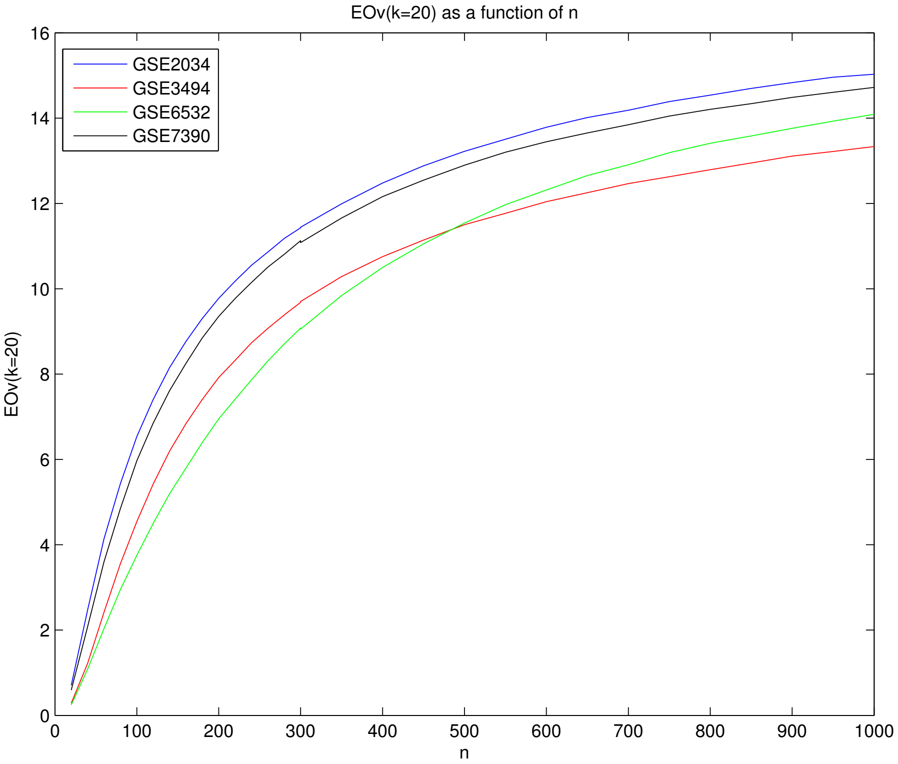


towards higher values of n. This is expected, as the amount of overlap for $n = \infty$ can not exceed k when comparing two lists of top-k genes.

Another statistic worth measuring is $P_k(X_{(i)})$, the probability of observing the top i-th gene from the sorted $\vec{r}$ in the top-k genes of a noisy replicate. In order to measure $P_k(X_{(i)})$, we stopped the simulation after generating the replicates, and calculated the frequency of occurrences of gene (i) in the top-k genes of replicates $D_1, \ldots, D_{N_t}$. We repeated this experiment for various values of n to show the effect of noise on $P_k(X_{(i)})$. Figure 4.3 shows the results of this experiment for $n = 10, 50, 100, 300$ using a $\vec{r}$ from GSE6532. N_t was set to $10,000$ for these set of experiments. As evident from the figure, an increase in the number of samples leads to an overall increase in the value of $P_k(X_{(i)})$. However, this increase is not uniform as the lower i's will approach a probability of 1 faster than others. This has an implicit effect on the runtime of our implementation of the analytical solution presented in Chapter 3: as n grows, it takes longer for values of $P_k(x_1, \ldots, x_r)$ to drop below a predefined ϵ, which causes the algorithm to a carry out a higher number of calculations before terminating. Needles to say, in the asymptotic case of $n = \infty$, all $P_k(X_{(i)})$ will be equal to 1, as all noise would be removed from the model.

By observing Figure 4.1 one might wonder if the plot could be fitted to a closed-form function of k. If such a function – $f(k)$ – exists , it could be used as an extension of the theoretical analysis to analytically compute and predict the value of $EOv(k)$, after computing the first feasibly-computable

Figure 4.2: $EOv(k)$ versus n for $n \in [20, 1000]$ at $k = 20$



terms and using the results to find the parameters of $f(k)$. In fact, as Figure 4.5 suggests, a quadratic function would provide a good fit to the data-points. In the example used for the figure, the first 100 points of $EOv(k)$ were used to fit a quadratic. However, as evident from the next figure, such a fit is not robust and will deviate from the simulation results as k grows, suggesting that $EOv(k)$ is not quadratic in k. One might wonder whether other degrees of polynomials, or exponential functions would provide more robust fits for $f(k)$. While Figure 4.6 suggests there could be a polynomial relation between $EOv(k)$ and k, we found the same problem of lack of robustness to be present for other degrees of polynomials, and even exponential fits.

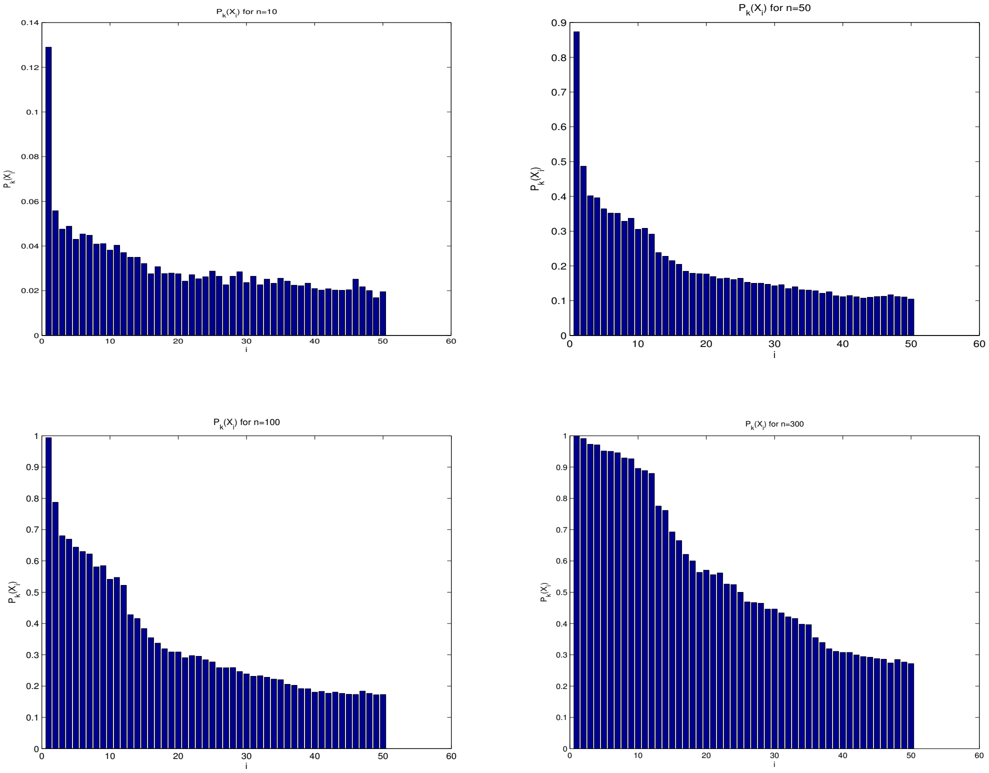
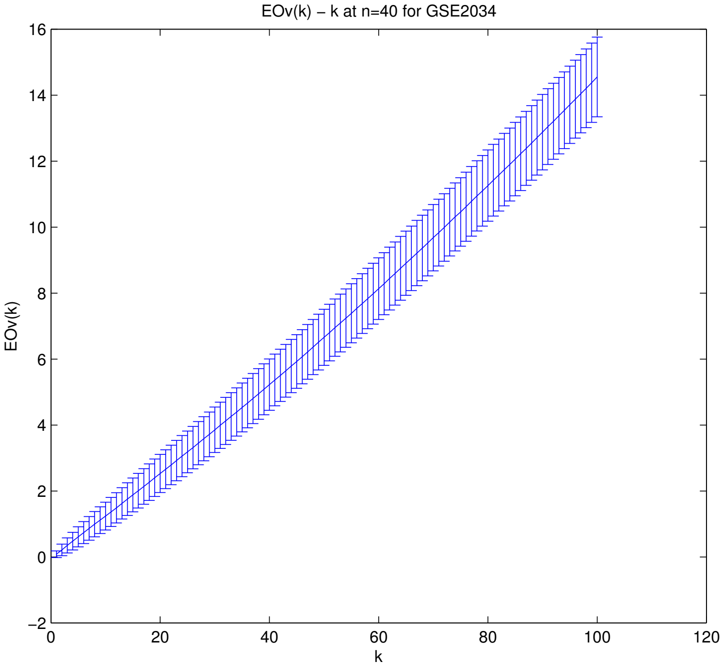


Figure 4.3: $P_k(X_{(i)})$ for $n = 10, 50, 100, 300$

Figure 4.4: $EOv(k) \pm$ std. dev. with $k_{max} = 100$ and $n = 40$



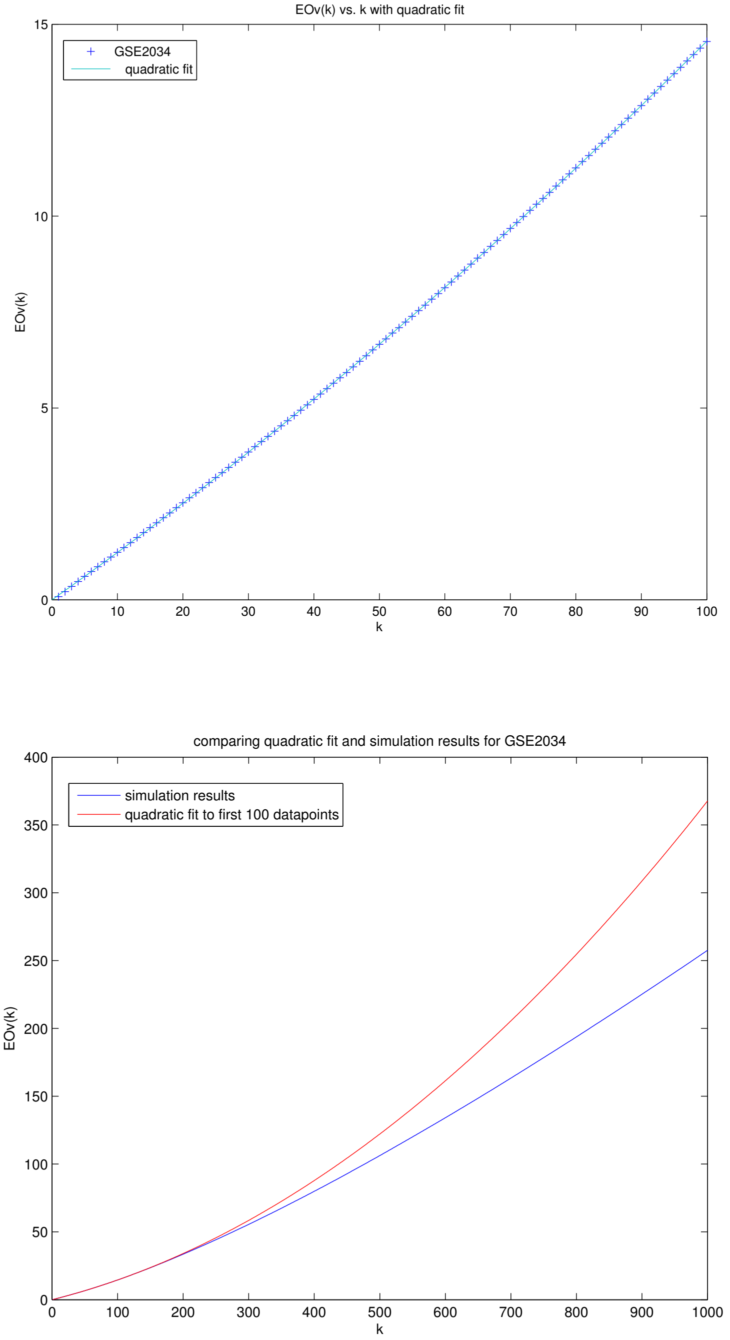
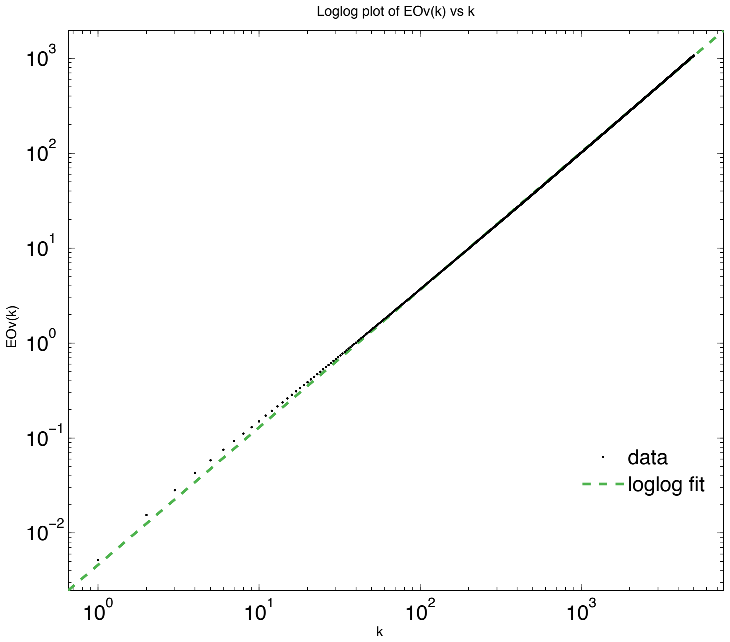


Figure 4.5: Up: a quadratic fit using $EOv(k)$ with $k_{max} = 100$. down: Comparison between the quadratic fit and the simulation results for $k_{max} = 1000$

Figure 4.6: Log-log fit of $EOv(k)$ vs k



Chapter 5

Validation

In this chapter we will discuss methods to empirically validate our model for predicting the expected overlap between two ranked gene lists. Section 5.1 describes sources to obtain $\vec{r}$, the vector of correlation scores used in our stochastic and analytical solutions as input. in Section 5.2, we describe our evaluation method. Section 5.3 presents the empirical results of validating our predictions using real datasets. We will show how these experiments provide strong evidence in support of our model, and discuss cases where we may underestimate or overestimate the size of overlap. Finally, in Section 5.4, we summarize the results of evaluating the estimations of our framework.

5.1 Sources and Robustness of $\vec{r}$

Until now, we have assumed that we are given a $\vec{r} = [r_1, \ldots, r_p]$ of true correlation scores for a study on p genes, from which we compute $\vec{\mu} = [\mu_1, \ldots, \mu_p]$ using $\mu_i = \tanh^{-1}(r_i)$, as the mean of random variable X_i for modelling the correlation score of gene number i. In this section, we will discuss the sources we can use to get $\vec{r}$, and the robustness of our approach for computing this $\vec{r}$.

We will use the following terms for the rest of this chapter:

Source: The source dataset(s) used to generate $\vec{r}$.

Target: The dataset(s) on which we want to estimate the expected overlap, using a $\vec{r}$ taken from a source.

Let $\vec{\rho} = [\rho_1, \ldots, \rho_p]$ denote the *observed* correlation scores from a source. We need to identify a source, such that we can assume that $\vec{r} \simeq \vec{\rho}$, i.e., the observed correlation scores taken from the source are good estimates of the *true* correlation scores.

There are two main factors that we need to consider when choosing such a source:

1. Homogeneity of source and target: That is, the source and target distributions, $\vec{r_s}$ and $\vec{r_t}$, should be homogeneous studies on the same p genes, for the same phenotype. Datasets built using different experimental conditions, patient demographics, etc., may produce very different $\vec{\rho}$, even when using identical array technologies and dealing with the same phenotype. Figure 5.1 shows the log-log plot of the first 1000 terms of $\vec{\rho}$, taken from 4 microarray datasets

on the same phenotype. As evident from the figure, for some studies the observed $\vec{\rho}$'s are close – for example GSE6532 and GSE7390 – while for others the difference is more significant. Therefore, using $\vec{\rho}$ as $\vec{r}$ from a source dataset whose distribution is not homogeneous with the target dataset may not produce good results. Section 5.3 shows that our model may overestimate or underestimate the expected overlap in cases where the difference between $\vec{\rho}$ of source and target datasets is significant.

2. Number of patients: The greater the number of patients, the closer $\vec{\mu}$ would be to true mean values[1]. Figure 5.2 shows how sub-sampling from a single dataset affects the resulting $\vec{\rho}$'s. As the figure suggests, using a subset with larger number of patients generates a $\vec{\rho}$ that is closer to the $\vec{\rho}$ generated using all the samples in the dataset.

In other words, condition 1 states that we prefer the means of $\vec{r}$ and $\vec{\rho}$ to be close, and condition 2 ensures that the variance around the means is small.

If the above two conditions are satisfied, we will identify the observed $\vec{\rho}$ as $\vec{r}$. Without loss of generality, let $\vec{r} = [r_1, \ldots, r_p]$ be the sorted correlation scores taken from the source. That is, renumber the genes, such that r_i is the i-th largest observed correlation score. This has no effect on the algorithms we presented in Chapter 3 and Chapter 4, as $\vec{r}$ is only used to model the underlying distribution of correlation scores, i.e., $\vec{r}$ would give us information about the distribution of true means of correlation scores.

5.2 Evaluation Method

Considering the factors mentioned in Section 5.1, we will evaluate the performance of our framework using three main approaches:

- Sub-sampling from a dataset: where source and target are different subsets of the same dataset. This approach is guaranteed to satisfy the homogeneity condition, assuming that subsets of a single dataset are homogeneous. However, the number of patients in the source dataset may be limited due to low number of patients in any given microarray dataset.
- Using different source and target datasets: use $\vec{r} = \vec{r}_A$ taken from study A with p genes on a phenotype, to estimate the overlap of sub-samples of study B, where B has p genes and provides class labels for the same phenotype. This approach allows us to use larger number of patients for the source dataset, but as will be shown, may not provide good results mainly due to not satisfying condition one.
- Comparing sub-samples of two different datasets: where two sub-samples of size n are randomly chosen as target from datasets A and B, and the remaining patients in the two datasets

[1] Assuming that observations of correlation scores would converge to true means of X_i, given infinitely large number of patients as samples.

Algorithm 5.1 EmpiricalOverlap(S, T_1, T_2): evaluate the expected overlap between top-ranked genes of target datasets T_1 and T_2 using dataset S as source.

```
Require: size of $T_1$ == size of $T_2$.
  Input: $N_t, N_s, k_{max}$: parameters of $EOv(.)$, stochastic approximation of expected overlap between top $1, \dots, k_{max}$ genes of two datasets, as described in Chapter 4.
  Output: Expected overlap of two target datasets $e$, standard deviation of expected overlap $e_{std}$, observed overlap between target datasets $O$.

  $C \leftarrow$ absolute value of correlation of $p$ genes in dataset $S$ with respect to outcome.
  $\vec{r} \leftarrow sort(C, \text{'desending order'})$
  $\vec{\mu} \leftarrow \tanh^{-1}(\vec{r})$
  $n_{target} \leftarrow |T_1|$
  $(e, e_{std}) \leftarrow EOv(\vec{\mu}, n_{target}, k_{max}, N_t, N_s)$
  for $i \in \{1, \dots, k_{max}\}$ do
    $O[i] \leftarrow$ Observed overlap of top-$i$ genes of $T_1$ and $T_2$.
  end for
  return $e, e_{std}, O$
```

are used as source to generate a $\vec{r}$ to estimate the overlap of the two sub-samples. Section 5.3.3 describes this procedure in detail.

Algorithms 5.1 describes the general procedure used to compare the estimated and observed overlaps of two datasets. All of our evaluation methods rely on sub-sampling to get the target datasets of equal size. To get more accurate results, we will repeat the procedure described in Algorithm 5.1 for different target datasets, and report the mean and standard deviation of observed and estimated results. All our evaluation methods presented in Section 5.3 use a slightly modified version of Algorithm 5.1.

5.3 Empirical Results

In this section we describe the empirical results of evaluating our framework. We will extensively focus on sub-sampling from a single dataset, as this approach satisfies both the homogeneity and number of patients conditions needed to get a good approximation for $\vec{r}$.

5.3.1 Sub-sampling From a Dataset

To evaluate the results of sub-sampling from a dataset, we used the evaluation algorithm described in Algorithm 5.2. Although we use a call to EmpiricalOverlap(S, T_1, T_2) in describing Algorithm 5.2, we modified Algorithm 5.1 slightly for efficiency, to avoid repeatedly estimating the expected overlap using the same source dataset.

In order to validate our model, we used all the datasets used in Chapter 4 that have more than 50 patients; see Table 5.1.

We used repeated random sub-sampling validation instead of cross-fold validation, since the number of folds in cross-fold validation depends on the size of the test set. As a result, for large

Figure 5.1: Log-log plot of $[r_1, \ldots, r_{1000}]$ for four datasets with identical Affymetrix platform and ER+/ER- phenotype.

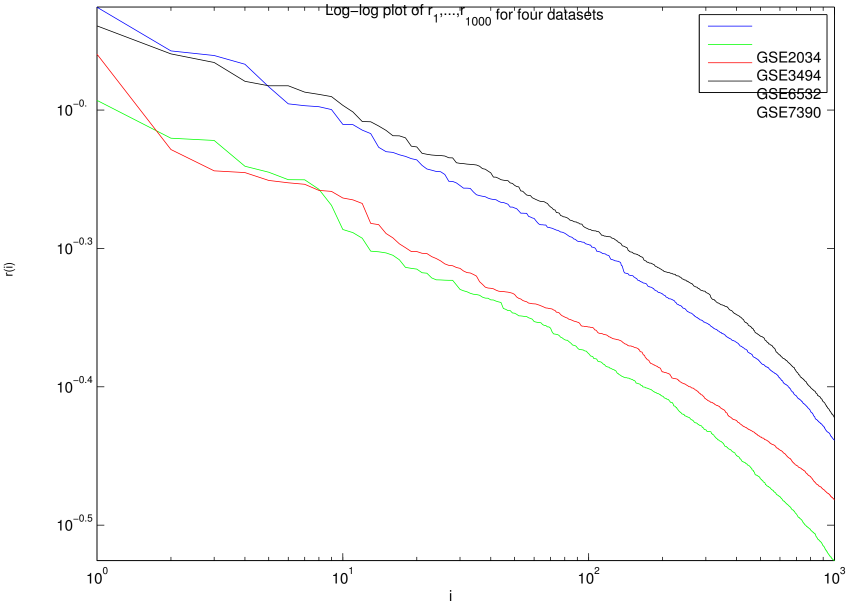


Table 5.1: Number of patients in datasets

Dataset	n
GSE2034	286
GSE3494	244
GSE6532	307
GSE7390	198
GSE7904	62

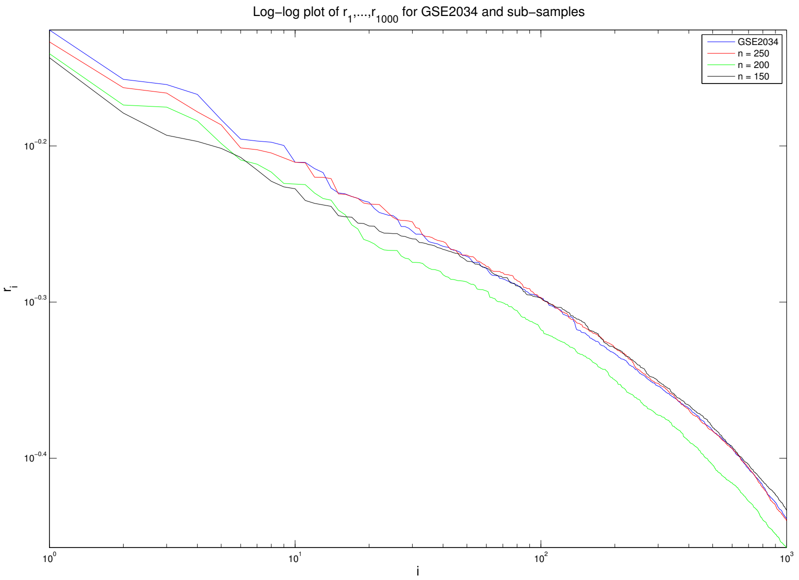
test sets one could end up with only one or two folds. The drawback of using repeated random sub-sampling is the possibility of not including some samples, or having two highly overlapping test folds. We can mitigate these drawbacks by doing many repetitions, i.e., choosing a higher value for H.

For each dataset, we experimented with at least two different values of n_{test}[2]. The values of n_{test} were chosen such that they could potentially be the sample size of microarray datasets (i.e., we did not use $n_{\text{test}} < 20$), and to maximize the possibility of having non-overlapping test sets given the number of patients in the input dataset. Since most gene list studies report the top 20, 30, 50 or 100 genes, we chose $k_{max} = 100$, and so will include the results for top-k for $k = 1, \ldots, 100$. In all the experiments we used $T = 30$, $H = 5$, $N_t = 10,000$, $N_s = 300$, unless otherwise specified.

Figure 5.3 shows a sample of the results using GSE2034 with $n_{\text{test}} = 50$. Error bars show one

[2] With the exception of GSE7904, because of its small sample size.

Figure 5.2: Effects of sub-sampling on $\vec{r}$ using sub-samples of size $n = 250, 200, 150$ of GSE2034 with $n = 286$.



standard deviation around the means of expected and observed overlaps. As the figure suggests, our analytic framework is able to predict the observed overlap with a high level of accuracy. In order to assess the quality of the predictions, we used Welch's paired t-test [75] to test the rejection of the null hypothesis of having equal means, at 5% significance level. Note that not being able to reject equality does not necessarily imply that the expected and observed overlap are equal,since statistical tests are unable to confirm that two population means are "statistically the same". However, it does provide us with some level of confidence in the model. Table 5.2 summarizes the results of experiments to validate the model using sub-sampling from datasets.

Table 5.2: Summary of sub-sampling experiments

Dataset	$n_{\textbf{test}}$	t-test fails to reject H_0
GSE2034	100	yes
GSE2034	80	yes
GSE2034	50	yes
GSE3494	80	yes
GSE3494	50	yes
GSE6532	100	yes
GSE6532	80	yes
GSE6532	50	yes
GSE7390	70	no
GSE7390	50	no
GSE7904	20	yes

Our experimental results in this section suggest strong agreement between the computed ex-

Algorithm 5.2 Subsample-Evaluate(D, n_{test}): evaluate the expected overlap between top-ranked genes of sub-samples of size n_{test} from dataset D.

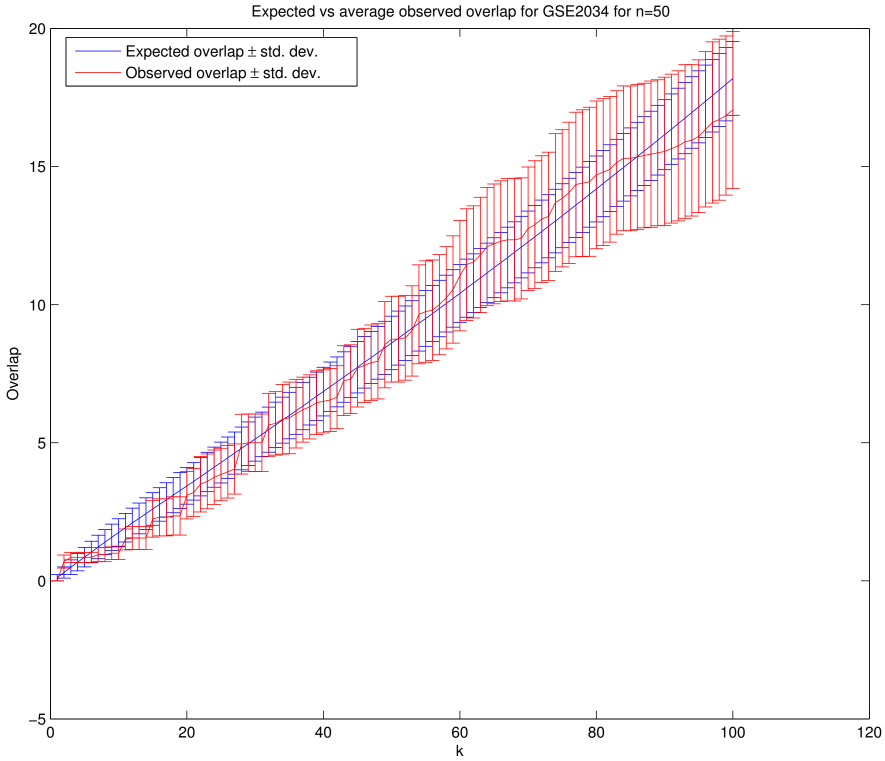
Input: N_t, N_s, k_{max}: parameters of $EOv(.)$, stochastic approximation of expected overlap between top $1, \ldots, k_{max}$ genes of two datasets, as described in Chapter 4, the number of times to repeat the evaluation H, and the number of second target datasets to select T.
Output: Expected overlap of top k_{max} genes of two target datasets $\vec{e}$, variance of expected overlap $e_{\vec{std}}$, average observed overlap of target datasets $\vec{O}$, variance of observed overlap $\vec{O_{std}}$.

```
**for** trial $\in \{1, \ldots, H\}$ **do**
  $T_1 \leftarrow$ Data for $n_{\text{test}}$ patients randomly selected from dataset $D$.
  $S \leftarrow D - T_1$
  **for** $i \in \{1, \ldots, T\}$ **do**
    $T_2 \leftarrow$ Data for $n_{\text{test}}$ patients randomly selected from $S$
    $e_i, e_{std_i}, O_i \leftarrow EmpiricalOverlap(S, T_1, T_2)$
  **end for**
  $e_{\text{trial}} \leftarrow mean(\vec{e})$
  $e_{std_{\text{trial}}} \leftarrow mean(e_{\vec{std}})$
  $O_{\text{trial}} \leftarrow mean(\vec{O})$
  $O_{std_{\text{trial}}} \leftarrow std(\vec{O})$
**end for**
**return** $\vec{e}, e_{\vec{std}}, \vec{O}, \vec{O_{std}}$
```

pected overlap and observed overlaps. However, in some cases, there was enough difference between our predictions and the observed overlap for t-test to reject the null hypothesis of equality. Figure 5.4 shows an example of overestimating the overlap for GSE7390 with $n_{\text{test}} = 50$. Even though the difference between the expected and observed overlap is large enough for t-test to reject equality, the mean expected overlap still lies within one standard deviation of the observed overlap for most values of k. In fact, in this example for $k = 100$, our model predicts an expected overlap of 20.35, whereas subsampling shows an average observed overlap of 17.8, which is probably good enough for a prediction. In contrast, Figure 5.5 shows an example of underestimating the overlap, for the same dataset with $n_{\text{test}} = 70$.

These differences might be caused by having too few case and control patients in datasets. In order to have balanced subsamples with the same case to control ratio as the original dataset, our subsampling algorithm may be forced to take additional samples in some cases, therefore using an inaccurate sub-sample size. Another reason could be heterogeneity of the input, which would invalidate our assumptions of homogeneity and having near-zero environmental noise in the data. In fact, in this particular case, GSE7390 is taken from a study titled "Strong Time Dependence of the 76-Gene Prognostic Signature for Node-Negative Breast Cancer Patients in the TRANSBIG Multicenter Independent Validation Series", whose data was taken from 6 different medical centers across Europe, using frozen samples taken from patients over the span of 18 years. Therefore, it is reasonable to assume that sub-samples of this dataset are not homogeneous, and that our model may not be able to accurately predict their overlap.

Figure 5.3: Expected vs. observed overlap for $k = 100$ and $n_{\text{test}} = 50$



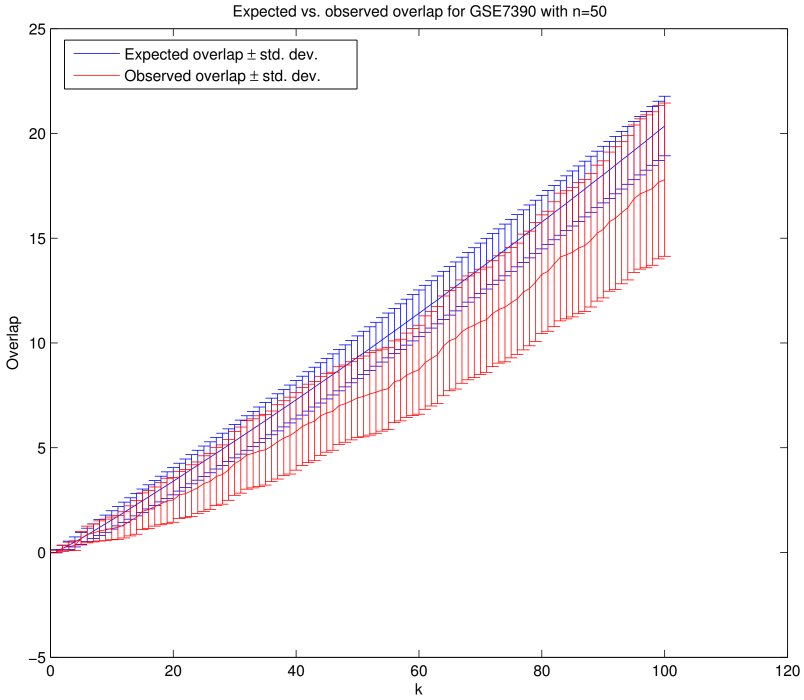
5.3.2 Different Source and Target Datasets

For this section, we used two datasets: one dataset to generate target sub-samples from, and a different dataset as source. Algorithm 5.3 summarizes the steps we took for evaluating the performance of our framework with this experimental setup.

Figure 5.1 shows the $\vec{r}$'s obtained from different datasets can be very different, even though they were based on identical array technologies and provide labels for the same phenotype. Therefore, our framework may not be able to estimate the expected overlap correctly, given a $\vec{r}$ from a different dataset. To empirically test this, we did two sets of experiments: one for two datasets whose $\vec{r}$ were similar, and one for two datasets whose $\vec{r}$ were significantly different. Note that the results of experiments in this section are intended to be used as empirical results for measuring the robustness of the estimations to $\vec{r}$, not to provide empirical evidence to validate our framework, since we chose the datasets a posteriori, after observing their respective $\vec{r}$.

In both sets of experiments, we used GSE3494 to get the target datasets. For the first set of experiments, we chose GSE6532 – whose $\vec{r}$ is somewhat similar to that of GSE3494 – as source. For the second set, we chose GSE2034 as source, whose $\vec{r}$ is different from that of GSE3494. We repeated each experiment for n_{test} values of 50 and 80. Figure 5.6 through Figure 5.9 show the

Figure 5.4: Expected vs. observed overlap for $k = 100$ and $n_{\text{test}} = 50$ for GSE7390 – An example of overestimating the overlap



results of these experiments.

As the figures suggest, the estimations of our framework are sensitive to $\vec{r}$ as input, and having $\vec{r}$ come from a dataset – that appears to not be homogeneous with the target datasets – adversely affects the estimations. As evident from the figures, our estimations were acceptable only for a subset of cases where $\vec{r}$ of source and target datasets were similar, that is, for $n_{\text{test}} = 50$, when using GSE6532 as source. Welch's t-test rejected the null hypothesis of equality at 5% significance levels for all cases except using GSE6532 with $n_{\text{test}} = 50$.

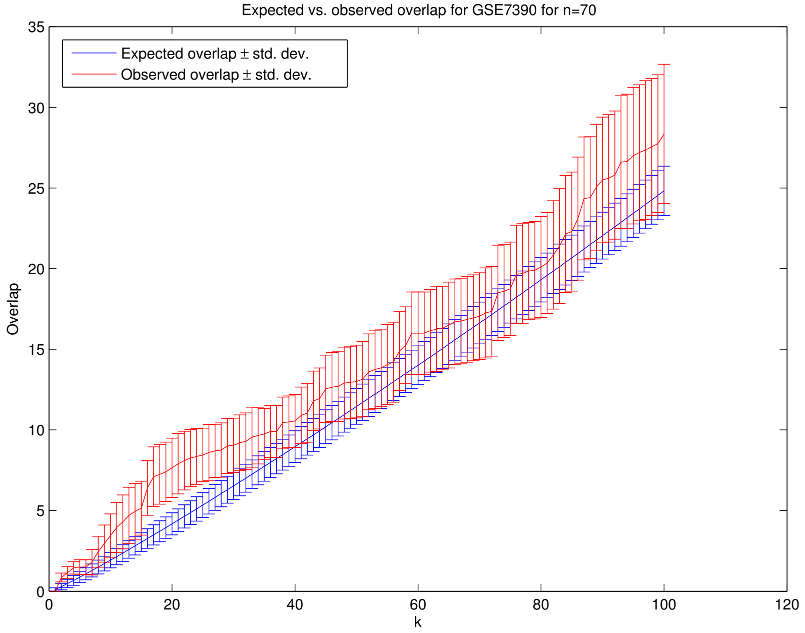
5.3.3 Sub-samples of Two Different Datasets

In this section, we will present the results for comparing two sub-samples of equal size, taken from two different datasets A and B. We also used a combination of $\vec{r}$ taken from datasets A and B. We considered two ways to combine the two $\vec{r}$s:

1. Take the union of datasets A and B, and use the new dataset to generate $\vec{r}$, i.e., source = $A \cup B$.

2. Let ρ_A and ρ_B denote the correlation scores of datasets A and B, respectively. Take two sub-samples of size n of datasets A and B, use the remaining patients in both datasets to compute ρ_A and ρ_B. Let $\rho_i = 0.5 \times (\rho_{A_i} + \rho_{B_i})$ for $i \in \{1, \ldots, p\}$. Sort $\vec{\rho}$ to get $\vec{r}$.

Figure 5.5: Expected vs. observed overlap for $k = 100$ and $n_{\text{test}} = 70$ for GSE7390 – An example of underestimating the overlap

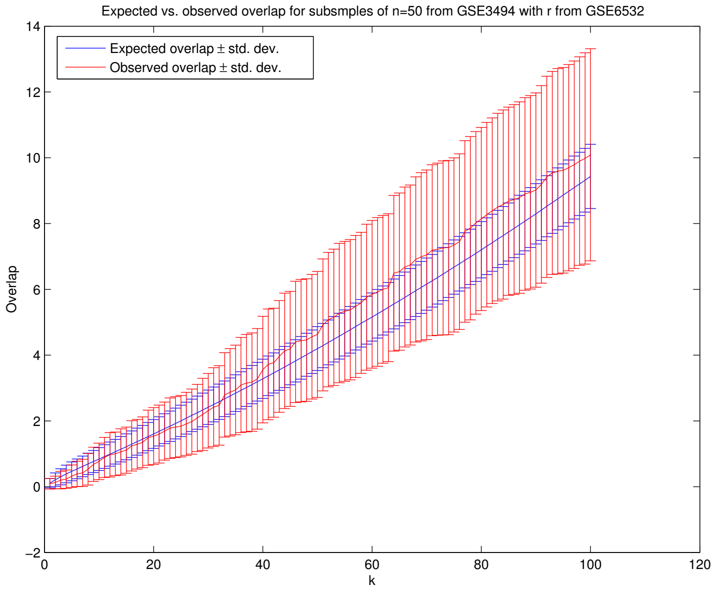


We picked the second approach, as the first approach is not straightforward due to batch effects introduced when sampling, and the bias between different datasets, and difficulties of adjusting and removing the batch effect; see [9] for an overview and evaluation of current methods. Using this method, we found that using this approach generally does not yield acceptable results, and t-test is able to reject equality at 5% significance level. Figure 5.10 shows one such attempt, using sub-samples of size $n = 50$ of two datasets GSE6532 and GSE7390.

5.4 Summary

In this chapter, we presented the results of empirically evaluating the performance of our framework for estimating the overlap of top-k genes of two n-patient studies. This required using a vector of true correlations $\vec{r}$; we needed to estimate this from some source. We then used this vector as the true means of correlation scores of genes in the target datasets. This often gave problematic answers, mostly due to heterogeneity of microarray datasets: an $\vec{r}$ from the source dataset was often inappropriate for target dataset, which meant that the source and target datasets were not homogeneous, although they were based on similar array technologies over the same phenotype. This hinders our ability to get a $\vec{r}$ that could be used to predict the expected overlaps of other datasets. However, sub-sampling from a single dataset showed strong agreement between our estimations and observed overlaps, showing that our model can accurately predict the expected overlap when provided with good estimations of $\vec{r}$.

Figure 5.6: Expected vs. observed overlap for $n_{\text{test}} = 50$ on GSE3494, with GSE6532 as source.



Algorithm 5.3 TwoDatasets-Evaluate($D_1, D_2, n_{\text{test}}$): evaluate the expected overlap between top-ranked genes of subsamples of size n_{test} from dataset D_1, using D_2 as source.

Input: N_t, N_s, k_{max}: parameters of $EOv(.)$, stochastic approximation of expected overlap between top $1, \ldots, k_{max}$ genes of two datasets (implicitly used in calling EmpiricalOverlap()), the number of times to repeat the evaluation H.

Output: expected overlap of two target datasets $\vec{e}$, standard deviation of estimated overlaps $\vec{e_{std}}$, observed overlap $\vec{O}$.

```
for $i \in \{1, \ldots, H\}$ do
    $T_1 \leftarrow$ Data for $n_{\text{test}}$ patients randomly selected from $D_1$
    $T_2 \leftarrow$ Data for $n_{\text{test}}$ patients randomly selected from $D_1$
    $e, std_e, O_i \leftarrow EmpiricalOverlap(D_2, T_1, T_2)$
end for
return $\vec{e}, \vec{e_{std}}, \vec{O}$
```

Figure 5.7: Expected vs. observed overlap for $n_{\text{test}} = 80$ on GSE3494, with GSE6532 as source.

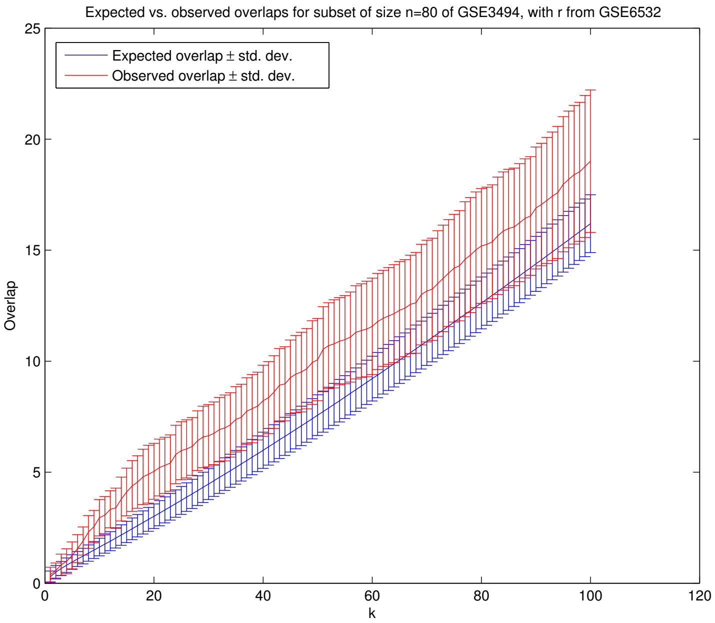


Figure 5.8: Expected vs. observed overlap for $n_{\text{test}} = 50$ on GSE3494, with GSE2034 as source.

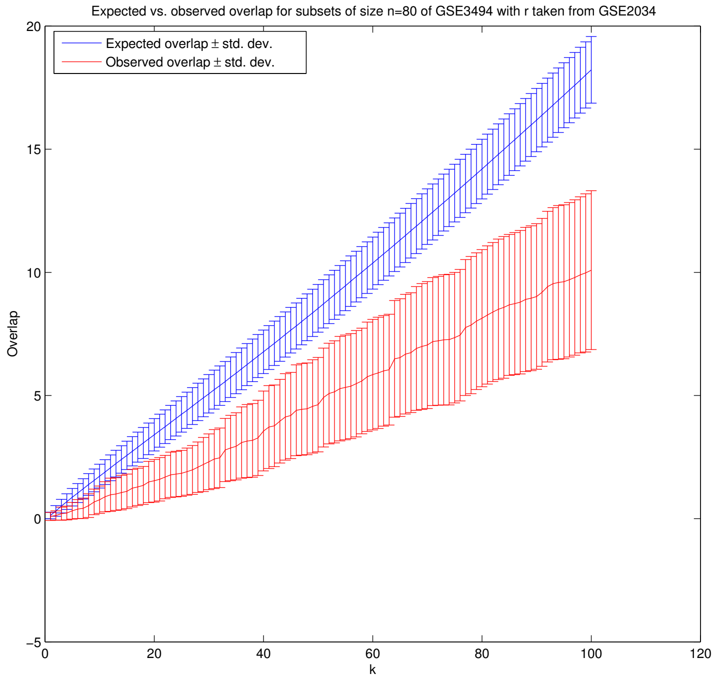


Figure 5.9: Expected vs. observed overlap for $n_{\text{test}} = 80$ on GSE3494, with GSE2034 as source.

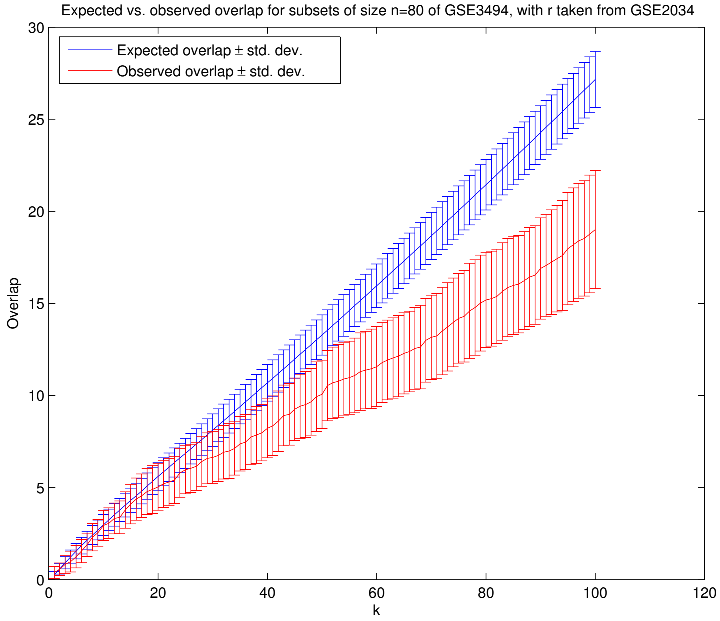
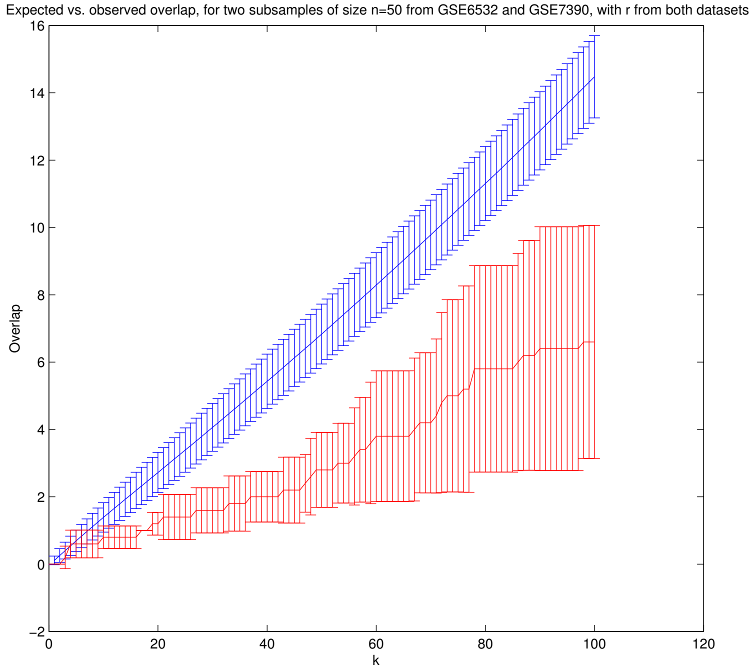


Figure 5.10: Comparing estimates and observed overlaps of sub-samples of size $n = 50$ from datasets GSE6532 and GSE7390 – using average of two $\vec{r}$ from both datasets as source.



Chapter 6

Conclusion

In this dissertation, we presented a mathematical model for estimating the expected overlap of two lists ranked by correlation to label. As shown in Chapter 5, when we have accurate estimates of true means of correlation scores $\vec{r}$, there can be good agreement between our estimations of expected overlap and the average observed overlap in the top-k genes of two datasets of the same size. However, as the performance of our model depends on the accuracy of the given $\vec{r}$ as an estimate of true means of correlation scores, the output might not be robust when the given $\vec{r}$ is not homogeneous with target datasets for which we want to estimate the overlap. In addition to heterogeneity, another possible reason for this disparity between our estimates and the observed overlap might be lack of enough data for generating $\vec{r}$. One possible future venue is using publicly available datasets to build a database of "(phenotype, microarray) $\to$ $\vec{r}$ tuples", and store pre-computed values of $\vec{r}$ for each phenotype and array technology pair.

Appendix B presents our suggestion for the distribution of $\vec{r}$. As discussed in the appendix, we believe that the absolute values of correlation score of top ranked genes of the gene signature follow a power-law, specifically a Pareto distribution. Our experiments show that for the same array technology and phenotype, the slope, the α parameter of the Pareto distributions is generally the same, while the intercept, x_m, changes for different datasets. A possible future work is to use more datasets to get a better estimate of α and x_m for various phenotypes. Doing so will enable us to use inverse CDF sampling, with α and x_m as distribution parameters for a given phenotype, to generate $\vec{r}$. This will enable us to predict the overlap of two datasets without using a source dataset to generate $\vec{r}$.

Our framework is able to estimate the expected overlap, with no assumption about the distribution of the elements in the list, and with minimal assumptions on the distribution of noise on observations. We developed a closed form analytical solution for estimating the overlap, and showed how we can use stochastic approximation to efficiently calculate the overlap, which is relevant as it is often computationally infeasible to compute the analytical solution. We used ranked gene lists, a.k.a. gene signatures, as a practical example to illustrate an application of our framework. The results of our framework suggest that number of patients, n, is the most important determining factor in the overlap of two ranked gene lists. As such, one application of our framework would be to use the analytical solution, to better understand whether the number of patients in a study is enough for extracting gene signatures that are likely to appear in different studies.

Bibliography

[1] Andrea Alimonti, Arkaitz Carracedo, John G Clohessy, Lloyd C Trotman, Caterina Nardella, Ainara Egia, Leonardo Salmena, Katia Sampieri, William J Haveman, Edi Brogi, Andrea L Richardson, Jiangwen Zhang, and Pier Paolo Pandolfi. Subtle variations in Pten dose determine cancer susceptibility. *Nature genetics*, 42(5):454–8, May 2010.

[2] A A Alizadeh, M B Eisen, R E Davis, C Ma, I S Lossos, A Rosenwald, J C Boldrick, H Sabet, T Tran, X Yu, J I Powell, L Yang, G E Marti, T Moore, J Hudson, L Lu, D B Lewis, R Tibshirani, G Sherlock, W C Chan, T C Greiner, D D Weisenburger, J O Armitage, R Warnke, R Levy, W Wilson, M R Grever, J C Byrd, D Botstein, P O Brown, and L M Staudt. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, 403(6769):503–11, February 2000.

[3] Ash a Alizadeh, Andrew J Gentles, Alvaro J Alencar, Chih Long Liu, Holbrook E Kohrt, Roch Houot, Matthew J Goldstein, Shuchun Zhao, Yasodha Natkunam, Ranjana H Advani, Randy D Gascoyne, Javier Briones, Robert J Tibshirani, June H Myklebust, Sylvia K Plevritis, Izidore S Lossos, and Ronald Levy. Prediction of survival in diffuse large B-cell lymphoma based on the expression of two genes reflecting tumor and microenvironment. *Blood*, 118(5):1350–8, June 2004.

[4] David B Allison, Xiangqin Cui, Grier P Page, and Mahyar Sabripour. Microarray data analysis: from disarray to consolidation and consensus. *Nature reviews. Genetics*, 7(1):55–65, January 2006.

[5] Pierre Baldi and Anthony D Long. A Bayesian framework for the analysis of microarray expression data: regularized t -test and statistical inferences of gene changes. *Bioinformatics*, 17(6):509–519, June 2001.

[6] J Barilan, M Mathassan, and M Levene. Methods for comparing rankings of search engine results. *Computer Networks*, 50(10):1448–1463, July 2006.

[7] Anne-Laure Boulesteix and Martin Slawski. Stability and aggregation of ranked gene lists. *Briefings in bioinformatics*, 10(5):556–68, September 2009.

[8] Atul Butte. The use and analysis of microarray data. *Nature reviews. Drug discovery*, 1(12):951–60, December 2002.

[9] Chao Chen, Kay Grennan, Judith Badner, Dandan Zhang, Elliot Gershon, Li Jin, and Chunyu Liu. Removing batch effects in analysis of expression microarray data: an evaluation of six batch adjustment methods. *PloS one*, 6(2):e17238, January 2011.

[10] Dung-Tsa Chen, Aejaz Nasir, Aedin Culhane, Chinnambally Venkataramu, William Fulp, Renee Rubio, Tao Wang, Deepak Agrawal, Susan M McCarthy, Mike Gruidl, Gregory Bloom, Tove Anderson, Joe White, John Quackenbush, and Timothy Yeatman. Proliferative genes dominate malignancy-risk gene signature in histologically-normal breast tissue. *Breast cancer research and treatment*, 119(2):335–46, January 2010.

[11] Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman. Power-law distributions in empirical data. *SIAM Review*, 51(4):43, June 2007.

[12] Jason Comander, Sripriya Natarajan, Michael a Gimbrone, and Guillermo García-Cardeña. Improving the statistical detection of regulated genes from microarray data using intensity-based variance estimation. *BMC genomics*, 5(1):17, February 2004.

[13] Xiangqin Cui and Gary a Churchill. Statistical tests for differential expression in cDNA microarray experiments. *Genome biology*, 4(4):210, January 2003.

[14] H. A. David and H. N. Nagaraja. *Order Statistics*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., Hoboken, NJ, USA, July 2003.

[15] Femke de Snoo, Richard Bender, Annuska Glas, and Emiel Rutgers. Gene expression profiling: decoding breast cancer. *Surgical oncology*, 18(4):366–78, December 2009.

[16] Christine Desmedt, Fanny Piette, Sherene Loi, Yixin Wang, Françoise Lallemand, Benjamin Haibe-Kains, Giuseppe Viale, Mauro Delorenzi, Yi Zhang, Mahasti Saghatchian D'Assignies, Jonas Bergh, Rosette Lidereau, Paul Ellis, Adrian L Harris, Jan G M Klijn, John a Foekens, Fatima Cardoso, Martine J Piccart, Marc Buyse, and Christos Sotiriou. Strong time dependence of the 76-gene prognostic signature for node-negative breast cancer patients in the TRANSBIG multicenter independent validation series. *Clinical cancer research : an official journal of the American Association for Cancer Research*, 13(11):3207–14, June 2007.

[17] K R Doyle, M A Mitchell, C L Roberts, S James, J E Johnson, Y Zhou, M von Mehren, D Lev, D Kipling, and D Broccoli. Validating a gene expression signature proposed to differentiate liposarcomas that use different telomere maintenance mechanisms. *Oncogene*, June 2011.

[18] Yotam Drier and Eytan Domany. Do two machine-learning based prognostic signatures for breast cancer capture the same biological processes? *PloS one*, 6(3):e17795, January 2011.

[19] Sandrine Dudoit, Juliet Popper Shaffer, and Jennifer C. Block. Multiple Hypothesis Testing in Microarray Experiments. *Statistical Science*, 18(1):71–103, February 2003.

[20] Alain Dupuy and Richard M Simon. Critical review of published microarray studies for cancer outcome and guidelines on statistical analysis and reporting. *Journal of the National Cancer Institute*, 99(2):147–57, January 2007.

[21] Liat Ein-Dor, Itai Kela, Gad Getz, David Givol, and Eytan Domany. Outcome signature genes in breast cancer: is there a unique set? *Bioinformatics (Oxford, England)*, 21(2):171–8, January 2005.

[22] Liat Ein-Dor, Or Zuk, and Eytan Domany. Thousands of samples are needed to generate a robust gene list for predicting outcome in cancer. *Proceedings of the National Academy of Sciences of the United States of America*, 103(15):5923–8, April 2006.

[23] Ronald Fagin, Ravi Kumar, and D. Sivakumar. Comparing Top k Lists. *SIAM Journal on Discrete Mathematics*, 17(1):134, 2003.

[24] R. a. Fisher. Frequency Distribution of the Values of the Correlation Coefficient in Samples from an Indefinitely Large Population. *Biometrika*, 10(4):507, May 1915.

[25] R. a. Fisher. On the" Probable Error" of a Coefficient of Correlation Deduced from a Small Sample. *Metron*, 1(5):3–32, May 1921.

[26] Richard J Fox and Matthew W Dimmic. A two-sample Bayesian t-test for microarray data. *BMC bioinformatics*, 7:126, January 2006.

[27] Gennadi V Glinsky, Olga Berezovska, and Anna B Glinskii. Microarray analysis identifies a death-from-cancer signature predicting therapy failure in patients with multiple types of cancer. *The Journal of clinical investigation*, 115(6):1503–21, June 2005.

[28] Y Hakak, J R Walker, C Li, W H Wong, K L Davis, J D Buxbaum, V Haroutunian, and A A Fienberg. Genome-wide expression analysis reveals dysregulation of myelination-related genes in chronic schizophrenia. *Proceedings of the National Academy of Sciences of the United States of America*, 98(8):4746–51, April 2001.

[29] John H. Halton. A Retrospective and Prospective Survey of the Monte Carlo Method. *SIAM Review*, 12(1):1, 1970.

[30] Lesleyann Hawthorn, Jesse Luce, Leighton Stein, and Jenniffer Rothschild. Integration of transcript expression, copy number and LOH analysis of infiltrating ductal carcinoma of the breast. *BMC cancer*, 10:460, January 2010.

[31] Fangxin Hong and Rainer Breitling. A comparison of meta-analysis methods for detecting differentially expressed genes in microarray experiments. *Bioinformatics (Oxford, England)*, 24(3):374–82, February 2008.

[32] Fangxin Hong, Rainer Breitling, Connor W McEntee, Ben S Wittner, Jennifer L Nemhauser, and Joanne Chory. RankProd: a bioconductor package for detecting differentially expressed genes in meta-analysis. *Bioinformatics (Oxford, England)*, 22(22):2825–7, November 2006.

[33] Kazuya Iwamoto and Tadafumi Kato. Gene expression profiling in schizophrenia and related mental disorders. *The Neuroscientist : a review journal bringing neurobiology, neurology and psychiatry*, 12(4):349–61, August 2006.

[34] Ian B Jeffery, Desmond G Higgins, and Aedín C Culhane. Comparison and evaluation of methods for generating differentially expressed gene lists from microarray data. *BMC bioinformatics*, 7:359, January 2006.

[35] Giuseppe Jurman, Stefano Merler, Annalisa Barla, Silvano Paoli, Antonio Galea, and Cesare Furlanello. Algebraic stability indicators for ranked lists in molecular profiling. *Bioinformatics (Oxford, England)*, 24(2):258–64, January 2008.

[36] W N Keith, K Lafferty-Whyte, C J Cairney, N Zaffaroni, and A Bilsland. Response to 'Validating a gene expression signature proposed to differentiate liposarcomas that use different telomere maintenance mechanisms'. *Oncogene*, August 2011.

[37] Seon-Young Kim. Effects of sample size on robustness and prediction accuracy of a prognostic gene signature. *BMC bioinformatics*, 10:147, January 2009.

[38] K Lafferty-Whyte, C J Cairney, M B Will, N Serakinci, M-G Daidone, N Zaffaroni, A Bilsland, and W N Keith. A gene expression signature classifying telomerase and ALT immortalization reveals an hTERT regulatory network and suggests a mesenchymal stem cell origin for ALT. *Oncogene*, 28(43):3765–74, October 2009.

[39] E S Lander. Array of hope. *Nature genetics*, 21(1 Suppl):3–4, January 1999.

[40] H.W. Lilliefors. On the Kolmogorov-Smirnov test for normality with mean and variance unknown. *Journal of the American Statistical Association*, 62(318):399–402, 1967.

[41] Sherene Loi, Benjamin Haibe-Kains, Christine Desmedt, Françoise Lallemand, Andrew M Tutt, Cheryl Gillet, Paul Ellis, Adrian Harris, Jonas Bergh, John a Foekens, Jan G M Klijn, Denis Larsimont, Marc Buyse, Gianluca Bontempi, Mauro Delorenzi, Martine J Piccart, and Christos Sotiriou. Definition of clinically distinct molecular subtypes in estrogen receptor-positive breast carcinomas through genomic grade. *Journal of clinical oncology : official journal of the American Society of Clinical Oncology*, 25(10):1239–46, April 2007.

[42] Nicholas Metropolis and S. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335, September 1949.

[43] Stefan Michiels, Serge Koscielny, and Catherine Hill. Prediction of cancer outcome with microarrays: a multiple random validation strategy. *Lancet*, 365(9458):488–92, 2005.

[44] George L Gabor Miklos and Ryszard Maleszka. Microarray reality checks in the context of a complex disease. *Nature biotechnology*, 22(5):615–21, May 2004.

[45] Lance D Miller, Johanna Smeds, Joshy George, Vinsensius B Vega, Liza Vergara, Alexander Ploner, Yudi Pawitan, Per Hall, Sigrid Klaar, Edison T Liu, and Jonas Bergh. An expression signature for p53 status in human breast cancer predicts mutation status, transcriptional effects, and patient survival. *Proceedings of the National Academy of Sciences of the United States of America*, 102(38):13550–5, September 2005.

[46] Lance D Miller, Johanna Smeds, Joshy George, Vinsensius B Vega, Liza Vergara, Alexander Ploner, Yudi Pawitan, Per Hall, Sigrid Klaar, Edison T Liu, and Jonas Bergh. An expression signature for p53 status in human breast cancer predicts mutation status, transcriptional effects, and patient survival. *Proceedings of the National Academy of Sciences of the United States of America*, 102(38):13550–5, September 2005.

[47] Rainer Opgen-Rhein and Korbinian Strimmer. Accurate ranking of differentially expressed genes by a distribution-free shrinkage approach. *Statistical applications in genetics and molecular biology*, 6(1):Article9, January 2007.

[48] Soonmyung Paik, Steven Shak, Gong Tang, Chungyeul Kim, Joffre Baker, Maureen Cronin, Frederick L Baehner, Michael G Walker, Drew Watson, Taesung Park, William Hiller, Edwin R Fisher, D Lawrence Wickerham, John Bryant, and Norman Wolmark. A multigene assay to predict recurrence of tamoxifen-treated, node-negative breast cancer. *The New England journal of medicine*, 351(27):2817–26, December 2004.

[49] Edith A. Perez, Lajos Pusztai, and Marc van De Vijver. Improving patient care through molecular diagnostics. *Seminars in Oncology*, 31:14–20, October 2004.

[50] S. B. Pounds. Estimation and control of multiple testing error rates for microarray studies. *Briefings in Bioinformatics*, 7(1):25–36, February 2006.

[51] Lajos Pusztai, Fraser W Symmans, and Gabriel N Hortobagyi. Development of pharmacogenomic markers to select preoperative chemotherapy for breast cancer. *Breast cancer (Tokyo, Japan)*, 12(2):73–85, January 2005.

[52] Xing Qiu, Yuanhui Xiao, Alexander Gordon, and Andrei Yakovlev. Assessing stability of gene selection in microarray data analysis. *BMC bioinformatics*, 7:50, January 2006.

[53] Sridhar Ramaswamy, Ken N Ross, Eric S Lander, and Todd R Golub. A molecular signature of metastasis in primary solid tumors. *Nature genetics*, 33(1):49–54, January 2003.

[54] Andrea L Richardson, Zhigang C Wang, Arcangela De Nicolo, Xin Lu, Myles Brown, Alexander Miron, Xiaodong Liao, J Dirk Iglehart, David M Livingston, and Shridar Ganesan. X chromosomal abnormalities in basal-like human breast cancer. *Cancer cell*, 9(2):121–32, March 2006.

[55] Andreas Rosenwald, George Wright, Wing C Chan, Joseph M Connors, Elias Campo, Richard I Fisher, Randy D Gascoyne, H Konrad Muller-Hermelink, Erlend B Smeland, Jena M Giltnane, Elaine M Hurt, Hong Zhao, Lauren Averett, Liming Yang, Wyndham H Wilson, Elaine S Jaffe, Richard Simon, Richard D Klausner, John Powell, Patricia L Duffey, Dan L Longo, Timothy C Greiner, Dennis D Weisenburger, Warren G Sanger, Bhavana J Dave, James C Lynch, Julie Vose, James O Armitage, Emilio Montserrat, Armando López-Guillermo, Thomas M Grogan, Thomas P Miller, Michel LeBlanc, German Ott, Stein Kvaloy, Jan Delabie, Harald Holte, Peter Krajci, Trond Stokke, and Louis M Staudt. The use of molecular profiling to predict survival after chemotherapy for diffuse large-B-cell lymphoma. *The New England journal of medicine*, 346(25):1937–47, June 2002.

[56] L SHAMPINE. Vectorized adaptive quadrature in MATLAB. *Journal of Computational and Applied Mathematics*, 211(2):131–140, February 2008.

[57] Cormac Sheridan. Third Tysabri adverse case hits drug class. *Nature reviews. Drug discovery*, 4(5):357–8, May 2005.

[58] Margaret A Shipp, Ken N Ross, Pablo Tamayo, Andrew P Weng, Jeffery L Kutok, Ricardo C T Aguiar, Michelle Gaasenbeek, Michael Angelo, Michael Reich, Geraldine S Pinkus, Tane S Ray, Margaret A Koval, Kim W Last, Andrew Norton, T Andrew Lister, Jill Mesirov, Donna S Neuberg, Eric S Lander, Jon C Aster, and Todd R Golub. Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature medicine*, 8(1):68–74, January 2002.

[59] Richard Simon. Roadmap for developing and validating therapeutically relevant genomic classifiers. *Journal of clinical oncology : official journal of the American Society of Clinical Oncology*, 23(29):7332–41, October 2005.

[60] A H Sims. Bioinformatics and breast cancer: what can high-throughput genomic approaches actually tell us? *Journal of clinical pathology*, 62(10):879–85, October 2009.

[61] Gordon K Smyth. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical applications in genetics and molecular biology*, 3(1):Article3, January 2004.

[62] T Sø rlie, C M Perou, R Tibshirani, T Aas, S Geisler, H Johnsen, T Hastie, M B Eisen, M van de Rijn, S S Jeffrey, T Thorsen, H Quist, J C Matese, P O Brown, D Botstein, P Eystein Lønning, and A L Bø rresen Dale. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences of the United States of America*, 98(19):10869–74, September 2001.

[63] Christos Sotiriou and Lajos Pusztai. Gene-expression signatures in breast cancer. *The New England journal of medicine*, 360(8):790–800, February 2009.

[64] Amanda Spink, Bernard J. Jansen, Vinish Kathuria, and Sherry Koshman. Overlap among major web search engines. *Internet Research*, 16(4):419–426, 2006.

[65] Roland B Stoughton. Applications of DNA microarrays in biology. *Annual review of biochemistry*, 74:53–82, January 2005.

[66] Korbinian Strimmer. A unified approach to false discovery rate estimation. *BMC bioinformatics*, 9:303, January 2008.

[67] Shin Takahashi, Takuya Moriya, Takanori Ishida, Hiroyuki Shibata, Hironobu Sasano, Noriaki Ohuchi, and Chikashi Ishioka. Prediction of breast cancer prognosis by gene expression profile of TP53 status. *Cancer science*, 99(2):324–32, February 2008.

[68] Zhi Qun Tang, Lian Yi Han, Hong Huang Lin, Juan Cui, Jia Jia, Boon Chuan Low, Bao Wen Li, and Yu Zong Chen. Derivation of stable microarray cancer-differentiating signatures using consensus scoring of multiple random sampling and gene-ranking consistency evaluation. *Cancer research*, 67(20):9996–10003, October 2007.

[69] Melissa a Troester, Jason I Herschkowitz, Daniel S Oh, Xiaping He, Katherine a Hoadley, Claire S Barbier, and Charles M Perou. Gene expression patterns associated with p53 status in breast cancer. *BMC cancer*, 6:276, January 2006.

[70] V G Tusher, R Tibshirani, and G Chu. Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences of the United States of America*, 98(9):5116–21, April 2001.

[71] Marc J van de Vijver, Yudong D He, Laura J van't Veer, Hongyue Dai, Augustinus a M Hart, Dorien W Voskuil, George J Schreiber, Johannes L Peterse, Chris Roberts, Matthew J Marton, Mark Parrish, Douwe Atsma, Anke Witteveen, Annuska Glas, Leonie Delahaye, Tony van der Velde, Harry Bartelink, Sjoerd Rodenhuis, Emiel T Rutgers, Stephen H Friend, and René Bernards. A gene-expression signature as a predictor of survival in breast cancer. *The New England journal of medicine*, 347(25):1999–2009, December 2002.

[72] Laura J van 't Veer, Hongyue Dai, Marc J van de Vijver, Yudong D He, Augustinus A M Hart, Mao Mao, Hans L Peterse, Karin van der Kooy, Matthew J Marton, Anke T Witteveen, George J Schreiber, Ron M Kerkhoven, Chris Roberts, Peter S Linsley, René Bernards, and Stephen H Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–6, January 2002.

[73] Laura J van't Veer and René Bernards. Enabling personalized cancer medicine through analysis of gene-expression patterns. *Nature*, 452(7187):564–70, April 2008.

[74] Yixin Wang, Jan G M Klijn, Yi Zhang, Anieta M Sieuwerts, Maxime P Look, Fei Yang, Dmitri Talantov, Mieke Timmermans, Marion E Meijer-van Gelder, Jack Yu, Tim Jatkoe, Els M J J Berns, David Atkins, and John a Foekens. Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *Lancet*, 365(9460):671–9, 2005.

[75] B. L. WELCH. The generalization of Student's problem when several different population variances are involved. *Biometrika*, 34(1-2):28–35, 1947.

[76] Min Zhang, Chen Yao, Zheng Guo, Jinfeng Zou, Lin Zhang, Hui Xiao, Dong Wang, Da Yang, Xue Gong, Jing Zhu, Yanhui Li, and Xia Li. Apparently low reproducibility of true differential expression discoveries in microarray studies. *Bioinformatics (Oxford, England)*, 24(18):2057–63, September 2008.

[77] Min Zhang, Lin Zhang, Jinfeng Zou, Chen Yao, Hui Xiao, Qing Liu, Jing Wang, Dong Wang, Chenguang Wang, and Zheng Guo. Evaluating reproducibility of differential expression discoveries in microarray studies by considering correlated molecular changes. *Bioinformatics (Oxford, England)*, 25(13):1662–8, July 2009.

[78] Elias Zintzaras and John P a Ioannidis. Meta-analysis for ranked discovery datasets: theoretical framework and empirical demonstration for microarrays. *Computational biology and chemistry*, 32(1):38–46, February 2008.

Appendix A

Efficient Algorithm for Finding the Overlap Between Two Ranked Lists

Efficient algorithms for finding the size of intersection between two lists, S_1 and S_2 of lengths m and n have a runtime complexity of $O(m+n)$. Algorithm A.1 shows the simple outline for such an algorithm:

Algorithm A.1 overlap(S_1, S_2)

```
$lookup \leftarrow \emptyset$
for $i = [1 : \text{length}(S_1)]$ do
  $lookup[S_1[i]] \leftarrow True$
end for
$overlap \leftarrow 0$
for $j = [1 : \text{length}(S_2)]$ do
  if $lookup[S_2[j]]$ then
    $overlap \leftarrow overlap + 1$
  end if
end for
return $overlap$
```

Let $\mathcal{O}(S_{1,2}[k])$ denote the size of overlap between the first k elements of two lists S_1 and S_2. Algorithm A.1 is efficient for finding the size of overlap of two lists. However, this implementation is not efficient if the goal is to repeatedly call the above function to find the overlap of k_{max} lists, i.e., finding $\forall k \in \{1, \cdots, k_{max}\}\mathcal{O}(S_{1,2}[k])$[1]. The main reason is that there is an inductive relation between $\mathcal{O}(S_{1,2}[k-1])$ and $\mathcal{O}(S_{1,2}[k])$. That is, we have:

$$\mathcal{O}(S_{1,2}[k]) = \begin{cases} 0 & \text{if } k = 0; \\ \mathcal{O}(S_{1,2}[k-1]) + 1 & \text{if } S_2[k] \in \{S_1[1], \cdots, S_1[k]\}; \\ \mathcal{O}(S_{1,2}[k-1]) & \text{otherwise.} \end{cases}$$

Therefore we can have a dynamic programming solution to efficiently compute the overlap for all the first k_{max} substrings of S_1 and S_2. To do so, we would need two lookup tables, ‘lookup’ and ‘future’. ‘lookup’ has the same functionality as Algorithm A.1, but is modified to hold the index of

[1] Assuming that both lists have the same length of k_{max}.

the occurrence of each element in S_1. This will not be problematic, as each element is a gene's index and is guaranteed to occur only once in each ranked list in our domain. 'future' holds the index of elements that will eventually be seen for some $i > k_{\text{current}} \in [1 : k_{max}]$. Algorithm A.2 describes the full algorithm.

Algorithm A.2 overlap_1_k(S_1, S_2, k_{max})

```
Require: both lists $S_1$ and $S_2$ have at least $k_{max}$ elements
  $lookup \leftarrow \emptyset$
  for $i = [1 : k_{max}]$ do
    $lookup[S_1[i]] \leftarrow i$
  end for
  $overlap[1 : k_{max}] \leftarrow 0$
  $future \leftarrow \emptyset$
  $O \leftarrow 0$ {$O$ holds the overlap so far}
  for $k = [1 : k_{max}]$ do
    if $lookup[S_2[k]]$ does not exist then
      $overlap[k] \leftarrow O$
      continue to next $k$
    end if
    $index_{S_1} \leftarrow lookup[S_2[k]]$
    if $0 < index_{S_1} \leq k$ then
      $O \leftarrow O + 1$
    else if $0 < index_{S_1} \leq k_{max}$ then
      $future[index_{S_1}] \leftarrow True$
    end if
    if $future[k]$ then
      $O \leftarrow O + 1$
    end if
    $overlap[k] \leftarrow O$
  end for
  return $overlap$
```

The runtime of algorithm A.2 is $O(k_{max})$. Algorithm A.2 can be easily generalized to find the overlap in the top k_{max} elements of a source list S_1 and target lists $S_2, \cdots, S_{N_s}$ in $O(N_s \times k_{max})$. Therefore, as mentioned in Chapter 4, we can use this algorithm to reduce the runtime of the simulation from $O(N_t \times N_s \times k_{max}^2)$ to $O(N_t \times N_s \times k_{max})$. As in most cases k_{max} is in the range of 20 to 100 (and sometimes 1000), this will provide a significant speedup to the simulation.

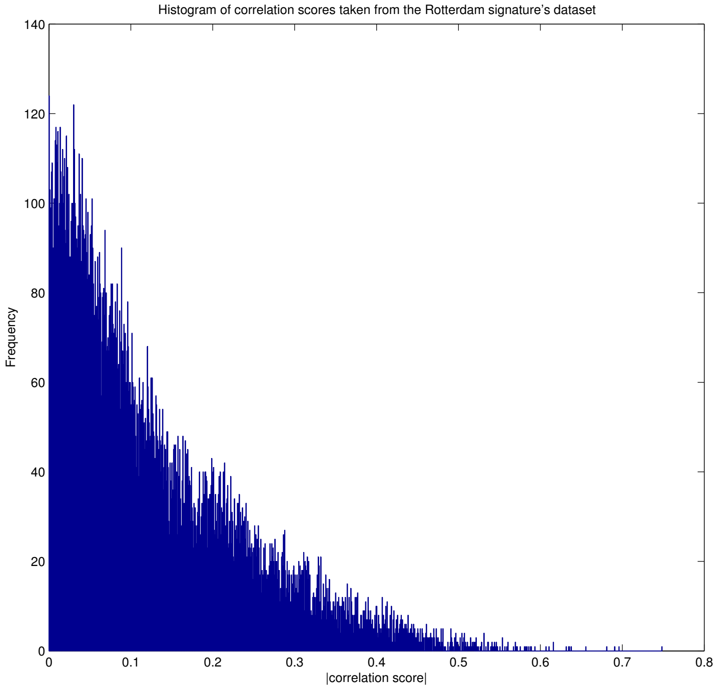
Appendix B

Distribution of Correlation Scores

The analytic framework we presented in this dissertation does not make any direct assumptions about the underlying distribution of correlation scores. However, if such a distribution exists, we could use it to synthesize datasets, and obtain $\vec{r}$ without using other microarray datasets, provided that we know the parameters of the distribution.

Figure B.1 shows the histogram of absolute values of correlation scores, taken from the Rotterdam signature's dataset. As the figure suggests, the frequency of low correlation scores is high, whereas the occurrence of high correlation scores is very infrequent. Such a trend suggests that the underlying distribution of correlation scores may be a power-law.

Figure B.1: Histogram of correlation scores taken from the Rotterdam signature's dataset



Power-laws are present as the underlying distribution of a wide variety of phenomena. A power-law states that the frequency of an item is inversely proportional to its frequency rank, i.e., high values of correlation scores are infrequent, and low values occur frequently. Power-laws are a family of probability distributions, one of which is the Pareto distribution. A Pareto distribution is defined by two parameters, α and x_m, as follows:

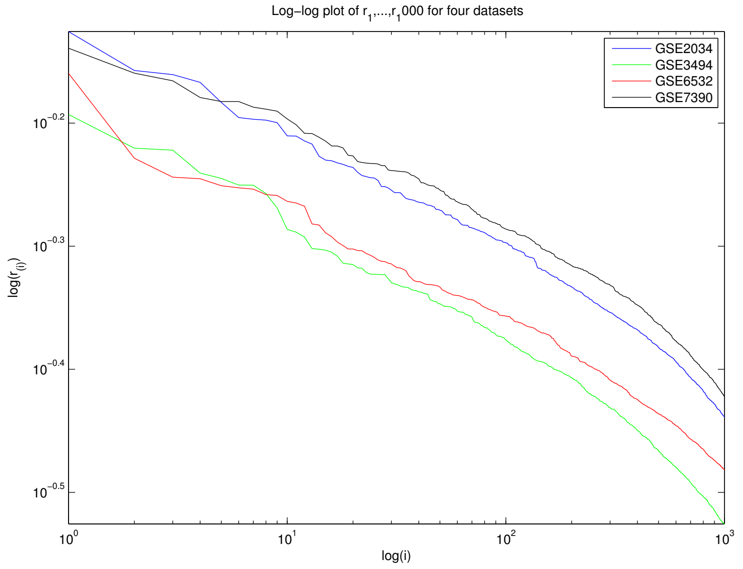
$$P(X \geq x) = \begin{cases} (\frac{x_m}{x})^{\alpha} & \text{for } x \geq x_m, \\ 1 & \text{for } x < x_m. \end{cases}$$

Therefore, the probability density function of a Pareto distribution is:

$$f_X(x) = \begin{cases} \alpha \frac{x_m^{\alpha}}{x^{\alpha+1}} & \text{for } x \geq x_m, \\ 0 & \text{for } x < x_m. \end{cases}$$

One of the characteristics of Pareto distributions is that their score versus rank plot appears as a line on the log-log scale. Figure B.2 shows the first 1000 terms of $\vec{r}$ for four microarray datasets, re-printed from Chapter 5, suggesting that such a linear trend exists for the microarray datasets we observed.

Figure B.2: Log-log plot of top 1000 correlation scores for four microarray datasets.

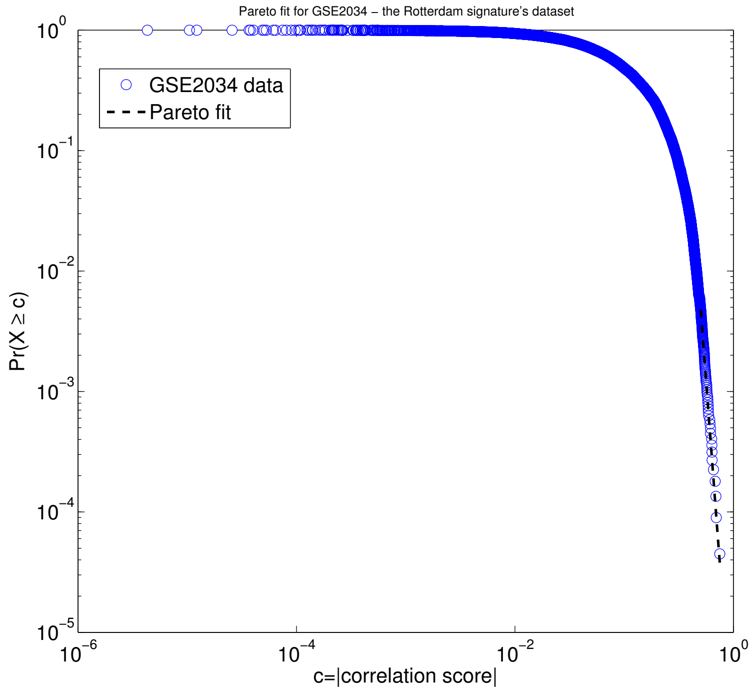


Assuming that we are given the α and x_m parameters, we can synthesize a dataset that follows a Pareto(α, x_m), using inverse-CDF sampling. To get T, a random sample from a Pareto(α, x_m) distribution, we can use the following equation:

$$T = \frac{x_m}{U^{\frac{1}{\alpha}}}$$

where U is a random number taken from the uniform distribution $U[0, 1]$. Using this technique p times, we can synthesize a dataset whose p correlation scores follow a Pareto with x_m and α

Figure B.3: Probability of observing a correlation score greater than c, for $c \in [0, 1]$, generated using the Rotterdam signature's dataset, with a Pareto fit superimposed.



parameters.

In order to get the maximum likelihood estimations[1] of α and x_m, we used the method suggested by Clauset et al. [11]. Using this method, we found that α is generally fixed, and in the range of $[12.5 - 14]$ for the microarray datasets we analyzed, which were based on Affymetrix platform, with an ER+/ER- phenotype. Figure B.3 shows the distribution of the Rotterdam dataset, with a black dotted-line showing the Pareto distribution superimposed. However, as the figure suggests, the Pareto fit only describes the behaviour of the tail of the distribution, i.e., the larger correlation scores. Moreover, repeating this experiment for other datasets showed that while the range of α is almost fixed for a (array type, phenotype) pair, x_m has a wider range, and therefore we may not be able to find an underlying distribution for the correlation scores of a known phenotype and array technology.

[1] Assuming that the underlying distribution is Pareto.