UNIVERSITEIT GENT

FACULTY OF SCIENCES

Assessing the performance of network crosstalk analysis combined with clustering

Sam De Meyer

Master's dissertation submitted to obtain the degree of
Master of Science in Biochemistry and Biotechnology
Major Bioinformatics & System Biology
Academic year 2015-2016

Promoter Ghent University: Prof. Dr. Klaas Vandepoele
VIB - Department Plant Systems Biology

PSB
A VIB-UGENT DEPARTMENT

Erasmus Promoter: Prof. Dr. Erik Sonnhammer
Scientific Supervisor: Christoph Ogris
Stockholm University, Sweden
Department of Biochemistry and Biophysics

Stockholm University

Acknowledgements

I would like to thank my thesis advisor Prof. Dr. Erik Sonnhammer and my scientific supervisor Christoph Ogris of the department of biochemistry and biophysics at Stockholm university for helpful feedback and discussions.

In addition, I would also like to thank Daniel Morgan, Stephanie Friedrich, Matteusz Kaduk and Dimitri Guala for welcoming me as a new member of the research group.

I would also like to acknowledge Prof. Dr. Klaas Vandepoele of the Bioinformatics & Systems Biology department at Ghent University for steering me in the right direction when I was in need of advice.

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Sam De Meyer

Table of contents

List of Tables

List of Figures

List of Abbreviations

AUC Area Under Curve. 44

BH Benjamini-Hochberg. vii, 9, 21, 24, 26, 30–32, 36, 50, 53, 80, 83, 85, 89, *List of Terms:* Benjamini-Hochberg

CLI Command Line Interface. iii, 38

DE Differentially Expressed. xiii, 2, 18, 21, 24, 25, 43, 48, 54

FA Functional Association. vii, 2, 5, 8–11, 16, 18, 20, 21, 25, 37, 38, 52, *List of Terms:* Functional Association

FCS Functional Class Scoring. ii, x, 3–5, 7, 8, 19–21, 25, 26, 42, 43, 46

FDR False Discovery Rate. ix

FPR False Positive Rate. 16, 24–27, 29, 33, 34, 42, 46, 52, 78

GEO Gene Expression Omnibus . vii, 19, 20, 23, 53, *List of Terms:* Gene Expression Omnibus

GO Gene Ontology. vii, 1, 2, 39, 44, *List of Terms:* Gene Ontology

GSA Gene Set Analysis. 4

GSEA Gene Set Enrichment Analysis. 4

HPA Human Proteome Atlas . iii–vii, xii, xiii, 16, 17, 39, 40, 42, 89–97, *List of Terms:* Human Proteome Atlas

HTD High Throughput Data. vii, x, 3, 10, *List of Terms:* High Throughput Data

IFA Impact Factor Analysis. 5

KEGG Kyoto Encyclopedia of Genes and Genomes. vii, x–xiii, 1–5, 14, 20, 23, 25, 27, 29–33, 36, 39, 40, 45, 51–54, 74, 77, 79, 80, 85, 89, 92, 97, *List of Terms:* Kyoto Encyclopedia of Genes and Genomes

LA Link Assignment. 8, 9

LA+S Link Assignment + Second-order conservation. 8, 9

LP Link Permutation. 8

MCL Markov Clustering. ii, vi, viii, 10–13, 17, 27–34, 36, 38, 53, 80–82, 85, 87, *List of Terms:* Markov Clustering

MG Merge Gain. 10, 11

MGclus Merge Gain Clustering. ii, vi, viii, 10, 11, 13, 17, 27–34, 36, 38–41, 45, 53, 80–82, 86, 88, 89, 92, 97, *List of Terms:* Merge Gain Clustering

MODY Maturity onset diabetes of the young. 40

MSigDB Molecular Signal DataBase . iii–vi, viii, 20, 32–36, 43, 46, 50, 52, 54, 55, 74, 77, 78, 82, 84–88, *List of Terms:* Molecular Signal DataBase

NCBI National Center for Biotechnology Information. ix, 19

NEA Network enrichment analysis. 8

NetGSA Network Based Gene Set Analysis. 5

NP Node Permutation. 8

ORA Over-Representation Analysis. ii, 3–6, 8, 10, 20, 25, 26

PADOG Pathway Analysis with Down-weighting of Overlapping Genes. viii, 4, 7, 16, 24–26, 45, 47, 53, 54, *List of Terms:* Pathway Analysis with Down-weighting of Overlapping Genes

PT Pathway Topology. ii, 3–5, 8, 14, 20, 21, 25

TF Transcription Factor. 32

TPR True Positive Rate. 24, 33

List of Terms

Benjamini-Hochberg A p-value correction to control the False Discovery Rate (FDR) for multiple testing. The original p-values are sorted and adjusted to *q-values* using the following formula:

$$q_i = \min_{j=i..m} (\min(\frac{m}{j} \times p_j,\ 1))$$

See Benjamini and Hochberg (1995) for more information on the rationale behind this procedure. vii, 9

BinoX BinoX—Binomial Crosstalk (X-talk) analysis—is a pathway analysis tool that uses the binomial distribution to model the number of links between two gene sets. Gene sets show significant crosstalk (or absence of crosstalk) if the number of links is significantly higher (or lower) than what would be expected by chance. iii, vi, xii, xiii, 5, 8, 10, 16, 17, 21, 24–43, 45–47, 53, 54, 81–83, 85–89, 92, 97

Functional Association A link/edge in a high coverage integrated biological network incorporating several kinds of evidence types. Nodes are typically either genes or proteins, and edge weight represent the level confidence that two nodes are somehow related in function. Prime examples of functional association networks are String (Snel *et al.*, 2000; Szklarczyk *et al.*, 2015) and FunCoup (Alexeyenko and Sonnhammer, 2009; Schmitt *et al.*, 2014). vii, 2

Gene Expression Omnibus A data repository for curated microarray and sequence-based data, hosted by the National Center for Biotechnology Information (NCBI) on `http://www.ncbi.nlm.nih.gov/geo/`, see also Edgar *et al.* (2002) and Barrett *et al.* (2013). vii, 19, 20

Gene Ontology The gene ontology project provides a controlled vocabulary to annotate genes and gene products. Annotations are divided into three main ontologies: biological process, molecular function and cellular component (Ashburner *et al.*, 2000; The Gene Ontology Consortium, 2014). vii, 1

High Throughput Data A large volume of data generated by a high throughput biology platform such as microarrays, next generation sequencing techniques or protein profiling. vii, x, 3

Human Proteome Atlas An online resource containing gene expression profiles for 32 human tissues. Next to RNA expression levels, additional evidence for tissue specific

gene expression, such as e.g. antibody profiling, is provided (Uhlen *et al.*, 2015). iii, iv, vii, xii, xiii, 16, 17, 39, 89–97

interesting gene list A list of genes obtained from High Throughput Data (HTD) that probably play a role in the phenotype/condition that was tested. An example is differentially expressed genes in a microarray experiment. Although the term *interesting gene list* is often used, the order of the genes in the list is, in most cases, of no importance. It would therefore be more accurate to use the term *interesting gene set* instead. Unfortunately this more accurate term is not widely used. 3

Kyoto Encyclopedia of Genes and Genomes A large database containing curated information about pathways, genes, biochemical reactions, chemical compounds and more for a number of reference species (Kanehisa and Goto, 2000; Kanehisa *et al.*, 2014). vii, x, 1

Markov Clustering A graph clustering algorithm based on a succession of two very simple matrix operations termed *expansion* and *inflation*, naturally leading to a clustering into tightly connected modules (van Dongen, 2000). ii, viii, 10, 11

Merge Gain Clustering A graph clustering algorithm that employs shared neighbours of nodes as extra extra evidence that nodes should be grouped in the same module. At the core of the algorithm is the *merge gain* score, which determines if the clustering improves if two modules are joined (Frings *et al.*, 2013). vi, viii, 10

Molecular Signal DataBase A large online collection of gene sets coming from a wide variety of sources. MSigDB contains curated gene sets such as Kyoto Encyclopedia of Genes and Genomes (KEGG) and Reactome pathways and experimental gene sets derived from e.g. gene expression data or methylation analysis. MSigDB is available at `http://software.broadinstitute.org/gsea/msigdb`, see also Subramanian *et al.* (2005). viii, 20

Pathway Analysis with Down-weighting of Overlapping Genes A Functional Class Scoring (FCS) approach to pathway analysis that down-weights genes occurring frequently in the *a priori* defined gene sets (Tarca *et al.*, 2012). viii, 4

query dataset A biological dataset arising from e.g. a microarray or RNA-seq study of a certain phenotype (e.g. a disease). A curated pathway that is affected by the phenotype or that is the cause of the phenotype is known. This pathway is called the *target pathway*. xi, 20

query gene set A gene set arising from a study of a certain phenotype (e.g. a disease). Any platform or method might have been used to obtain the gene set. A curated pathway that is affected by the phenotype or that is the cause of the phenotype is known. This pathway is called the *target pathway*. xi, 20, 52, 85

Reactome A manually curated, publicly accessible database for human pathways and reactions (Croft *et al.*, 2014; Fabregat *et al.*, 2015). x, xi, 3, 33, 44

target pathway A curated pathway (e.g. from KEGG or Reactome) that is known to cause or be affected by a certain phenotype. See also *query dataset* and *query gene set*. iv, x, 20, 21, 23–29, 31, 33, 34, 36, 43, 44, 54, 74, 77, 79, 81, 85–88

Nederlandse samenvatting

Biologisch inzicht verkrijgen uit *high throughput* gen expressie experimenten is tot op het heden een grote uitdaging. Een groot aantal methoden zijn reeds ontwikkeld in deze *pathway* analyse tak van de bio-informatica. Een groep van deze methoden evalueert de significantie van de overlap tussen een set van differentieel geëxpresseerde genen en *pathways* van een functionele annotatie databank zoals bijvoorbeeld KEGG. Enkele typische voorbeelden van deze groep zijn BiNGO (Maere *et al.*, 2005), DAVID (Huang *et al.*, 2007) en GoMiner (Zeeberg *et al.*, 2003). Een fundamenteel probleem van deze methoden is dat de huidige annotatie databanken ver van compleet zijn, wat het moeilijk maakt om een significante overlap te verkrijgen. Om dit probleem te omzeilen zijn er recent een aantal netwerk (of graaf) gebaseerde methoden ontwikkeld zoals CrossTalkZ (McCormack *et al.*, 2013), BinoX (submitted for publication), NEA (Alexeyenko *et al.*, 2012) en EnrichNet (Glaab *et al.*, 2012). Deze berekenen de significantie van het aantal links tussen twee sets van genen, en kunnen bij deze het probleem van incomplete databanken gedeeltelijk omzeilen. Een ander probleem dat nog steeds onopgelost is, is dat de sets van genen vaak "ruis" bevatten: genen die door experimentele variatie in de set terecht kwamen maar eigenlijk niet differentieel geëxpresseerd zijn. Om dit probleem aan te pakken kan clusteren eventueel helpen. Het idee is om een gen set te clusteren en *pathway* analyse uit te voeren op de individuele modules, deze methode heeft het potentieel om de sensitiviteit nog meer te verhogen.

In deze thesis heb ik het effect van deze clustering methode geëvalueerd op twee verschillende onafhankelijk gecureerde *gold standard* datasets. Ik heb ondervonden dat, wanneer grote gen sets gebruikt worden, de stijging in *false positive rate* groter is dan de stijging in sensitiviteit. Het verkrijgen van meer sensitiviteit en de daling van specificiteit wordt minder extreem naargelang kleinere gen sets gebruikt worden. Helaas, zelfs voor kleine gen sets is er geen voordeel te halen uit clusteren. Dit werd geobserveerd voor beide datasets en lijkt ook het geval te zijn bij het analyseren van gen sets van de *Human Proteome Atlas* (HPA).

English summary

Extracting biological insight from high-throughput gene expression experiments is still a major challenge and has led to the development of many new data analysis techniques. One category of these, termed pathway analysis techniques are very powerful tools for annotating functions to Differentially Expressed (DE) genes sets created by these experiments. A large number of overlap based pathway analysis techniques have been developed such as BiNGO (Maere *et al.*, 2005), DAVID (Huang *et al.*, 2007), GoMiner (Zeeberg *et al.*, 2003) and many more. These tools are based on finding a significant overlap of the given gene set with gene sets from annotation databases such as KEGG (Kanehisa *et al.*, 2014). A fundamental problem of overlap based methods is that these databases are very incomplete, and thus it is often difficult to find a statistically significant overlap. To overcome this limitation, newer techniques such as CrossTalkZ (McCormack *et al.*, 2013), BinoX (submitted for publication), NEA (Alexeyenko *et al.*, 2012) and EnrichNet (Glaab *et al.*, 2012) have been developed. These tools interpret the gene sets in the context of a functional association network such as FunCoup (Alexeyenko and Sonnhammer, 2009; Schmitt *et al.*, 2014), or String (Szklarczyk *et al.*, 2015). By searching for a significant interaction between gene sets in a network context, the sensitivity can be increased. A remaining problem is that gene sets derived from experimental data are often noisy, thus clustering these gene sets and performing pathway analysis techniques on the separate modules might increase the sensitivity even further.

In this thesis I have benchmarked the effect of clustering and found that the rise in false positive rate outweighs the gain in specificity when dealing with large gene sets. The gain in sensitivity and loss of specificity seem to decrease for smaller gene sets, but even for small gene sets there is still no advantage from using clustering. This has been observed for two independently curated gold standard datasets and also seems to be the case when analysing gene sets from the Human Proteome Atlas (HPA).

Part 1: Introduction

High throughput biological methods, capable of measuring thousands of biological molecules at the same time, have become an increasingly important source of information in the last two decades. These methods provide us with enormous amounts of data, but extracting useful biological insights from this data is still a major challenge. Many of these high throughput techniques such as DNA and RNA sequencing or protein profiling lead to large lists of interesting genes/proteins that are affected by a condition of interest. Although these lists are useful for determining genes that have a role in a certain phenomenon or condition, they are often too large to be easily interpreted by researchers. It is therefore difficult to gain any biological insight into the underlying processes that causes a certain condition or phenotype.

To tackle this problem, a suite of tools have been developed which search for biological themes in these gene lists. These tools thus reduce a long list of genes into a shorter list of themes or categories represented by the genes, reducing the complexity and providing more mechanistic insights. Examples of such themes are pathways or sets of genes that are typically affected by a certain phenotype or condition. A large number of databases have been developed to provide such themes, such as the Kyoto Encyclopedia of Genes and Genomes (KEGG) database, providing curated pathways (Kanehisa and Goto, 2000; Kanehisa *et al.*, 2014), and the Gene Ontology (GO) database, providing gene sets belonging to biological processes, molecular functions or cellular components (Ashburner *et al.*, 2000; The Gene Ontology Consortium, 2014). This approach is now a routine task when interpreting high throughput data and, when attempting to find pathways underlying a gene list, is frequently called “pathway analysis”. When one does not specifically look for pathways however, the term “functional enrichment analysis” can also be used.

Although many tools and annotation databases exist, there are still major challenges in the field of pathway analysis. For example, it is still an ongoing discussion which statistical models are most appropriate for pathway analysis. Other problems are incompleteness of annotation databases and the noisy data inherent to high throughput techniques to name a few.

In the following sections I will first introduce the concept of functional annotation and pathway databases. Then I will briefly explain the idea behind functional association networks. With this background information in mind, I will introduce the field of pathway analysis, describing different ways to approach this problem and their drawbacks. Then I will introduce four pathway analysis tools that are used throughout this thesis, followed

by a short discussion of how graph based clustering might have potential to improve some of these tools. Finally, I will briefly go over some of the biggest problems in the field—in particular the absence of a standard benchmarking procedure—and how these problems can be addressed.

1.1 Functional annotation and pathway databases

When the function of a gene or gene product is determined, this information can be deposited in one of several functional annotation databases. One of the most popular and widely used functional annotation databases is hosted by the GO consortium (Ashburner *et al.*, 2000). This database annotates a large set of genes from different organisms and contains three basic categories: biological process, molecular function and cellular component. More fine-grained annotations are possible for genes of which there is more knowledge. The GO annotations form a consistent description of genes which are interpretable for both humans and computer programs. Another example is MSigDB (Subramanian *et al.*, 2005), containing a large collection of annotated gene sets from different sources. These gene sets include curated pathways as well as gene sets derived from experiments. A certain category of these functional annotation databases are pathway databases such as the KEGG and Reactome (Milacic *et al.*, 2012; Croft *et al.*, 2014), which contain gene sets belonging to the same pathway, and are often highly curated. These pathway databases are particularly interesting for analysing Differentially Expressed (DE) gene lists with pathway analysis tools, because genes from the same pathway are often co-regulated. A major disadvantage is that only a small part of known genes belong to a certain pathway. The main focus of this thesis is on pathway analysis methods, making use of these pathway databases.

1.2 Functional association networks

By integrating the large amounts of omics data freely available online (as well as text mining and/or other data sources), high coverage biological networks can be constructed. The nodes in these networks represent biological entities such as genes, and the edges describe the degree of belief that these entities are somehow functionally related. Two examples of such high coverage Functional Association (FA) networks are FunCoup (Alexeyenko and Sonnhammer, 2009; Schmitt *et al.*, 2014) and String (Szklarczyk *et al.*, 2015). Both of these networks were constructed from a large collection of heterogeneous data types using a supervised Bayesian learning algorithm.

1.3 Three generations of pathway analysis methods

The use of high throughout techniques has been on the rise since the last two decades, and pathway analysis has become a routine task over the last ten years (Khatri *et al.*,

2012). Over this last decade, concepts of how to interpret High Throughput Data (HTD) have evolved from very simple models considering only overlap between a given gene set and pathways in an annotation database to very complex models that take into account gene to gene and pathway to pathway relationships. Khatri *et al.* (2012) have reviewed 68 of these tools in the last ten years and divided them into three generations: Over-Representation Analysis (ORA) tools, Functional Class Scoring (FCS) approaches and Pathway Topology (PT) based approaches.

1.3.1 Over-Representation Analysis

This is the first and conceptually most simple approach. Typically a list of genes affected by a condition of interest is obtained from HTD using a test statistic and a significance cutoff. A good example is a microarray experiment with x and y replicates in condition a and b respectively. One *t-test* is then performed per gene to test for a difference in mean expression between condition a and b. Finally all p-values are corrected for multiple testing and all genes with a corrected p-value below a certain significance cutoff, typically 0.05, are considered to have a role in the condition that was tested. This is what is often called an *interesting gene list*, although *gene set* would be a more accurate description since the order of genes in the list is of no importance.

The obtained gene set is then used to query other gene sets in an annotation database such as KEGG or Reactome (Croft *et al.*, 2014; Fabregat *et al.*, 2015). The first generation methods perform their statistical tests on the number of overlapping genes between the query gene set and the gene sets in the annotation database. Usually a two by two contingency table is created and a statistical test based on the hypergeometric distribution, binomial distribution or χ^2 (chi-squared) distribution is used to assess the significance of the overlap (see section 1.4.1 for an example). The output is then a list of pathways that are present in the input gene set, associated with a test statistic or a p-value. Popular first generation tools include BiNGO (Maere *et al.*, 2005), DAVID (Huang *et al.*, 2007) and GOstats (Falcon and Gentleman, 2007).

There are obvious limitations to these first generation methods. First of all, every tool tests for an overlap as extreme or more extreme as the observed overlap (taking into account gene set sizes and background genes). This means that whenever there is no overlap between two gene sets, the probability of observing an equally big or bigger overlap is equal to 1. In other words, non-overlapping gene sets will always be deemed insignificant. The problem is that no overlap can occur when taking a too strict p-value cutoff for determining the input gene list. No overlap can also be a consequence of a pathway in the annotation database being incomplete. A second problem is that many genes are co-regulated, leading to correlations between genes that are independent of the experiment performed (see also section 1.4.3). First generation methods use a set based approach to test significance, assuming that all genes are independent. This assumption is not met and gene-gene correlations severely impact the false positive rate of these first generation tools (Gatti *et al.*, 2010). A third problem is that statistics associated with the interesting gene list are discarded, treating each gene equally, no matter their log-fold change or test statistic. The fourth problem is that pathways are being treated as a mere collection of

genes, without any internal structure. Regulatory links between genes in a pathway are discarded, removing a lot information.

In conclusion, first generation methods make assumptions that do not hold, are dependent of a subjective significance cutoff for generating interesting gene lists, do not cope well with incomplete annotation databases and discard a lot of information. But despite these shortcomings, first generation methods are still very popular because they are quickly implemented and easy to use.

1.3.2 Functional Class Scoring approaches

FCS approaches generally work in three big steps. First a gene-wise statistic is computed, assessing the importance of each gene to the tested condition. This can be a obtained by a t-test or moderated t-test for example (Smyth, 2004). Second, a pathway-level statistic is computed, such as the sum, mean or median of the genes that are in the pathway. Examples of more sophisticated statistics are the maxmean statistic used by *Gene Set Analysis (GSA)* (Efron and Tibshirani 2007) or a weighted running sum used by *Gene Set Enrichment Analysis (GSEA)* (Subramanian *et al.*, 2005). Third, the significance of the pathway level statistic is assessed. Usually an empirical p-value is obtained by permuting the sample/phenotype labels of the samples x times and recomputing the pathway level statistic, generating a distribution of pathway level statistics under the null hypothesis. The significance of the pathway is then the proportion of test statistics in this null distribution that are more extreme than the true pathway level test statistic. This method of significance testing is used by most FCS approaches, including GSEA, GSA and *Pathway Analysis with Down-weighting of Overlapping Genes (PADOG)* (Tarca *et al.*, 2012).

FCS approaches address several of the inherent limitations of ORA based approaches. Most importantly, they do not depend on a subjective p-value cutoff, but use all values instead. In addition, they do not treat all genes as equal since they make use of the gene-wise test statistics. Finally, they address the gene-gene correlation problem that ORA based methods face, because swapping the sample labels creates a null distribution that preserves gene-gene correlations. Currently there seems to be an agreement in the literature in favor of this sample-permutation based null hypothesis (Ackermann and Strimmer (2009); Jiang and Gentleman (2007); Tian *et al.* (2005); Glazko and Emmert-Streib (2009); Gatti *et al.* (2010) and Goeman and Bühlmann (2007), see also Khatri *et al.* (2012) supplementary text S2.3 for a discussion on this topic).

FCS approaches still suffer from one major limitation: just like ORA methods, they treat pathways merely as gene sets. Any additional information such as regulatory links between genes in a pathway is discarded.

1.3.3 Pathway Topology based approaches

A number of annotation databases, such as KEGG (Kanehisa and Goto, 2000; Kanehisa *et al.*, 2014) and Reactome (Croft *et al.*, 2014; Fabregat *et al.*, 2015) provide detailed information about regulatory links between genes in a pathway. This is important informa-

tion that is not being used by ORA or FCS based approaches. This limitation is addressed by PT based approaches, which give more weight to genes that are central to a pathway and incorporate interaction types between upstream and downstream genes. A popular PT tool for signaling pathways is Impact Factor Analysis (IFA) (Draghici *et al.*, 2007), which takes a gene set and a pathway graph as input, and a recent improvement (Voichita *et al.*, 2012) of IFA called *pathway express*, which eliminates the need for a significance cutoff to obtain a gene set. A recent and promising tool is Network Based Gene Set Analysis (NetGSA) (Ma *et al.*, 2014), which accounts for changes in pathway topology based on the experimental condition. However, it is often not known how pathways rewire depending on the condition of the sample, which makes this method not widely applicable.

Although PT based methods address all inherent limitations of ORA and FCS based methods, they do introduce a new limitation on their own. To use these methods, detailed knowledge of the pathways is required, which is only available for model organisms and well studied pathways. In conclusion, PT based methods are the most sophisticated but at the same time the most limiting tools, while ORA based methods are the most widely applicable since they only require gene sets, and FCS based methods fall somewhere in between. So, despite their shortcomings, ORA and FCS based methods will remain important for the foreseeable future.

1.4 A selection of pathway analysis tools

In this thesis, I perform a benchmark for a few approaches to pathway analysis in combination with clustering (see chapter 2). These approaches will be introduced here. The first two are ORA based methods, while the third is a FCS approach. The fourth approach, BinoX (submitted for publication, Stockholm University), is a fairly new method that does not fit in any of the canonical categories in the literature. Conceptually, it belongs to a new group of methods that use FA networks to provide extra information for pathway analysis.

1.4.1 The Fisher test

The Fisher test (also known as Fisher's exact test) is most often used for first generation pathway analysis methods. The Fisher test uses the hypergeometric distribution for assessing the significance of overlap between an input gene set and gene sets from an annotation database. It is implemented by popular tools such as GoMiner (Zeeberg *et al.*, 2003) and BiNGO (Maere *et al.*, 2005).

Suppose we have a genome comprising N genes, an input gene set of size n and a gene set from an annotation database, for example a KEGG pathway, of size K. Also suppose that of the n genes we have, k genes that overlap with the pathway. With this information we can create a 2×2 contingency (table 1.1).

We have drawn n genes in total, of which k belong to the pathway and g belong to the collection of genes in the genome that are not a member of the pathway. The probability of getting exactly k genes out of K and g genes out G when drawing n genes out of N is

	in pathway	not in pathway	*row total*
in gene set	k	g	n
not in gene set	$\bar{k}$	$\bar{g}$	$\bar{n}$
column total	K	G	N

Table 1.1: A contingency table as used in the Fisher test.

then given by the hypergeometric distribution:

$$P(X = k) = \frac{\binom{K}{k}\binom{G}{g}}{\binom{N}{n}} \tag{1.1}$$

For ORA, a one sided test is performed to determine the probability of obtaining an overlap that is equally big or bigger than the observed overlap. The probabilities of all possible values of k equal or bigger than the observed k are summed up:

$$P(X \geq k) = \sum_{i=k}^{min(n,K)} \frac{\binom{K}{i}\binom{G}{g}}{\binom{N}{n}} \tag{1.2}$$

This will give us the p-value of the pathway, which will then be corrected for multiple testing across all gene set versus pathway combinations. Note that it does not matter which gene set is used as input and as pathway. By expanding the binomial coefficients, it can be shown that the following identity holds:

$$\frac{\binom{K}{k}\binom{G}{g}}{\binom{N}{n}} = \frac{\binom{n}{k}\binom{\bar{n}}{\bar{k}}}{\binom{N}{K}} \tag{1.3}$$

As noted before, the Fisher test assumes that, under the null hypothesis, all genes are drawn independently from each other. But, as was shown by Gatti *et al.* (2010), genes within a pathway are often correlated, even if the pathway is not affected by the experimental condition. That is, if one gene is drawn, then correlated genes are likely to be drawn as well. This way the significance of the pathway is overestimated.

1.4.2 EASE

The online DAVID tool (Huang *et al.*, 2007) uses the EASE score (Hosack *et al.*, 2003) to determine if a pathway is significantly enriched in a gene set. The EASE score is a slightly modified version of the Fisher test: it uses exactly the same procedure but uses an overlap of $k - 1$ instead of k. The reasoning behind this is that the Fisher test assigns too low p-values when there is an overlap, especially if one of the gene sets is very small. By reducing the overlap the p-value will be increased.

$$P_{EASE}(k) = \sum_{i=(k-1)}^{\min(n,K)} \frac{\binom{K}{i}\binom{G}{g}}{\binom{N}{n}} \tag{1.4}$$

The EASE score, lacking a true statistical motivation, is not an ideal solution to this limitation of the Fisher test. Also note that when $k = 0$, the EASE score is undefined. Still, I decided to also include this approach in the benchmark, since the DAVID tool is quite popular and the EASE score is quickly implemented.

1.4.3 PADOG

The PADOG tool (Tarca *et al.*, 2012) belongs to the second generation of pathway analysis approaches, and is specifically designed for microarray or RNA sequencing data. The first step of any FCS approach is to compute a gene-wise test statistic, in this case the moderated t-test is used (Smyth (2004), see also section 5.1 for details). Then, a pathway level test statistic is computed using the following formula:

$$S_{GS_i} = \frac{1}{|GS_i|} \sum_{g \in GS_i} \text{abs}(t_g) \cdot w_g \tag{1.5}$$

Where $|GS_i|$ is the set size (cardinality) of the i'th gene set, GS_i, of an annotation database. The notation "abs(t_g)" is used here for the absolute value of the moderated t-test statistic of gene g. To summarize, S_{GS_i} is the weighted mean of absolute moderated t-values of gene set i. The weights w_g are given by:

$$w_g = 1 + \sqrt{\frac{\max_{g' \in G}(f_{g'}) - f_g}{\max_{g' \in G}(f_{g'}) - \min_{g' \in G}(f_{g'})}} \tag{1.6}$$

Where f_g is the frequency with which gene g occurs in all gene sets, and G is the superset of all genes in all gene sets that are tested. This means that genes that appear in all gene sets will be given a weight of 1 while genes that are unique to one gene set will be given a weight of 2.

The pathway level statistic is then standardized by subtracting from S_{GS_i} the mean of abs$(t_g) \cdot w_g$ (for all genes in G) and dividing the result by the standard deviation of abs$(t_g) \cdot w_g$, yielding a new value $S^*_{GS_i}$. This value is then standardized again by subtracting the mean and dividing by the standard deviation of $S^*_{GS_i}$ over all gene sets, yielding the final pathway level statistic $S^{**}_{GS_i}$.

The significance of the pathway level statistic is, as for other FCS approaches, computed by making sample label permutations and recalculating the pathway level statistic for every permutation. The p-value of the pathway is the proportion of pathway level statistics under the null hypothesis that are as high or higher than the observed pathway level statistic.

As other FCS approaches, PADOG eliminates the need for an arbitrary p-value cutoff and accounts for correlations between genes by using sample label permutations as the null hypothesis. In addition it will give higher weights to genes that are unique to few pathways, increasing contrast between overlapping pathways. A limitation of PADOG and other FCS based approaches is that all the information available on pathways is not used. Pathway are treated as gene sets, instead of a complex system of interacting genes.

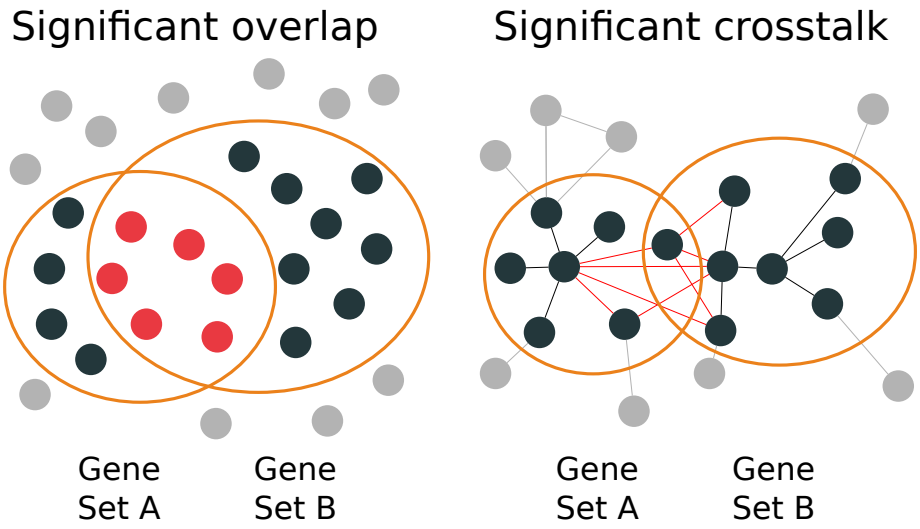


Figure 1.1: ORA versus crosstalk analysis. Difference between network based methods and traditional overlap based methods. (left) Traditional overlap-based tools will report an enrichment if the overlap between two gene sets is more than what would be expected by chance. These tools do not use any information from an underlying network. (right) Newer network-based tools make use of an underlying network. BinoX for example counts the number of links between two gene sets and will report an enrichment if this number is higher than what would be expected by chance

1.4.4 BinoX

BinoX (submitted for publication, Stockholm University) is a very recently developed tool and does not really fit into any of the categories previously mentioned (ORA, FCS, PT). It takes gene sets and a FA network such as FunCoup (Alexeyenko and Sonnhammer, 2009; Schmitt *et al.*, 2014) or String (Snel *et al.*, 2000; Szklarczyk *et al.*, 2015) as input. BinoX will then perform network crosstalk analysis: it compares the number of functional associations (network links) between two gene sets with an estimation of the number of links that would occur purely by chance. If two gene sets share more links than what would be expected by chance, then they are said to be enriched to each other. On the contrary, if the number of links is low, they are said to be depleted (fig. 1.1). In addition to BinoX, a few other network based approaches exits:

- CrossTalkZ (McCormack *et al.*, 2013), the predecessor of BinoX,
- NEA, standalone and web-server tool (Alexeyenko *et al.*, 2012) and
- EnrichNet, a web-server tool (Glaab *et al.*, 2012).

The BinoX algorithm works in three big steps. First the functional association network is randomized n_{iter} times (150 times by default) to generate the null hypothesis. Four different methods can be used for randomizing the network: Link Permutation (LP), Node Permutation (NP), Link Assignment (LA) and Link Assignment + Second-order conservation (LA+S). LP will simply swap links between nodes until all original links have been replaced, thus preserving the node degree distribution of the original network. NP will swap node labels of all nodes, but label swaps are restricted to nodes with similar node degree (more details in McCormack *et al.* (2013)). This approach will, approximately, preserve the node

degree distribution of original network. LA will remove all links from the original network and add random links to nodes as long as their node degree is lower than in the original network. LA+S is similar too LA but adds an extra constraint to the links added in the random network. For each node, the distribution of node degrees of the neighbour nodes is determined. A link from node a to node b may only be formed if the degree of b is similar to the degree of a neighbour of a and vice versa. This preservers not only the node degree distribution, but also the node degree distribution of neighbours of nodes. For LA and LA+S, it is often not possible to complete randomization due to the degree constraints. This problem is solved by link-swapping between problematic nodes (see McCormack *et al.* (2013) for details). The default option is LA+S as with this option the network properties of the randomized network are most similar to the original network.

For the second step, the number of links, k, between two gene sets is assumed to follow a binomial distribution under the null hypothesis:

$$k_0 \sim Bin(p, n) \tag{1.7}$$

There are two parameters in the binomial distribution that have to be estimated: the probability of success, p, and the number of Bernoulli trials, n:

$$n = \text{min}\left(d_{out}(A),\ d_{out}(B),\ |A| * |B|\right) \tag{1.8}$$

$$p = \frac{\frac{1}{n_{iter}} \sum_{i=1}^{n_{iter}} k_i}{n} \tag{1.9}$$

Where A and B are the gene sets, $|X|$ is the size of set X and $d(X)$ is the total number of links of all members of X in the given FA network, excluding links from genes in X to other genes in X. Here n is the number of links that can possibly occur between the gene sets, and p is the average number of links between the gene sets in the randomized networks divided by n.

Third, the probability that the observed number of links, k_{obs}, or more would occur under the null hypothesis is determined, yielding the enrichment p-value (eq. 1.10). The p-value for depletion is calculated in a similar manner (eq. 1.11). Finally the p-values are corrected for multiple testing using the Benjamini-Hochberg (BH) correction.

$$P_{enrich} = \sum_{i=k_{obs}}^{n} \binom{n}{i} p^i (1-p)^{n-i} \tag{1.10}$$

$$P_{deplete} = \sum_{i=0}^{k_{obs}} \binom{n}{i} p^i (1-p)^{n-i} \tag{1.11}$$

Two things are worth noting here. One is that the enrichment and depletion p-value are two different things and are not complementary since both include the probability of k_{obs}. I.e., P_{enrich} and $P_{deplete}$ do not add up to one. The other thing worth nothing is that if a pathway is *enriched* or *depleted* to an interesting gene set, this should not be interpreted as *upregulated* or *downregulated*. Instead, enrichment should be interpreted as

the pathway being affected by whatever condition gave rise to the gene list, while depletion should be interpreted as the pathway being unaffected.

BinoX overcomes an important limitation of the ORA based methods: it can test gene sets that have no overlap. This is important because our current knowledge of pathways is far from complete, giving rise to missing genes in pathways. By using extra information from a FA network, this limitation can be overcome. BinoX shares two important drawbacks with ORA however: getting gene lists from HTD dependents on a subjective significance cutoff, and pathway structure is not considered.

1.5 Introducing graph clustering

Gene expression experiments, which are the most common source of interesting gene lists, are often noisy and often comprise multiple pathways. The same is true for most other high throughput methods such as DNA methylation analysis and mass spectrometry. The presence of noise and parts of different pathways in the gene list make it more difficult for certain pathway analysis tools to find enriched gene sets. It might therefore be an interesting idea to first divide the obtained gene set into modules[1] using a graph clustering algorithm and a functional annotation network. The pathway analysis can then be carried out on every individual module instead (see section 2.2).

In this thesis, I present the result of this form of pathway analysis using two different graph clustering algorithms, Merge Gain Clustering (MGclus) (Frings *et al.*, 2013) and Markov Clustering (MCL) (van Dongen, 2000). I will briefly introduce both methods.

1.5.1 MGclus

Modules in a network are typically defined as groups of nodes that are tightly connected, with a maximum number of internal connections (intra-module connectivity) and as few as possible external connections (inter-module connectivity). But this definition might be too simplistic to apply to the currently available biological networks, especially FA networks. Since they are static, condition specific edges will most likely be missing. In addition, most FA networks are derived from HTD, giving rise to many false positive edges. Too compensate for this, MGclus does not only look at the intra versus inter module connectivity but also considers shared neighbours of nodes as evidence that they belong to the same module.

MGclus works by iteratively optimizing the Merge Gain (MG) score until a certain criterion is met. The MG score is calculated as:

$$MG = 2E_{ij} - (E_i + E_j) \tag{1.12}$$

[1] By graph clustering, a graph is divided into smaller, possibly overlapping sets of nodes, called “clusters” or “modules”. The outcome of a clustering algorithm as a whole, considering all modules, is sometimes called the “clustering” of the graph. To avoid confusion of the terms “cluster” and “clustering”, I will from hereon always use the term “module” when referring to a “cluster” or “module”.

Where E_{ij} is the clustering efficiency (eq. 1.13) of the new module formed by the union of modules M_i and M_j. The clustering efficiency of any module M_c is found by:

$$E_c = \frac{N_{c,intra}}{N_{c,total}} \tag{1.13}$$

$$\text{where} \qquad N_{c,total} = |M_c|^2 \tag{1.14}$$

$$\text{and} \qquad N_{c,intra} = \sum_{x=1}^{|M_c|-1} \sum_{y=x+1}^{|M_c|} w(x,y) + \sqrt{w_{cnb}(x,y)} \tag{1.15}$$

Here $N_{c,intra}$ (eq. 1.15) is calculated by iterating over all possible node pairs in the module M_c. The value of $w(x,y)$ is the edge weight between node x and node y or 0 if there is no edge and $w_{cnb}(x,y)$ is the sum of all edges linking x and y to every common neighbour node of both x and y (which do not have to be a part of the module).

The MGclus algorithm takes as input a FA network and a parameter called the MG score cutoff C_{mg}. The MG is then optimized iteratively. This starts by considering every node as a single module and then joining modules on every iteration. Clusters are only considered for joining if they have a least one direct link. In every iteration all MG scores are calculated and a sorted list of t new module candidates with the best MG score is created. Clusters are then joined starting at the top of the list. If a new module candidate has a MG score less than the cutoff C_{mg} the join operation is not performed. If a module appears twice in the list, only the joining operation with the best MG score is done. This, in most cases, leads to less than t joining operations per iteration. When no more new modules can be made with a MG score above the cutoff, the iterations are stopped, yielding the final clustering outcome. A high C_{mg} will generally lead to smaller module sizes.

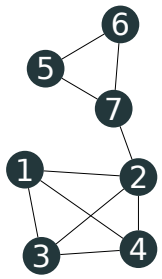
The biggest strength of MGclus is that it is relatively robust to false positive and false negative edges in the FA network. Another nice property is that the granularity can be tuned be changing the MG score cutoff. The biggest downside to MGclus is that recomputing the MG scores in every iteration results in a high and difficult to predict runtime complexity. The MGclus program, implemented in java, is also completely single threaded. These two factors make the MGclus program rather slow. Computation times of up to a few hours can be expected as soon as the edge count is higher than about $50\,000$ (on Ubuntu linux with a 2.2Ghz processor).

1.5.2 Markov Clustering

Markov clustering interprets a module as a set of nodes that, when making random walks over the edges of the graph, it is likely to end up in the same set of nodes again. Another but equivalent formulation is that a module has many high length paths that connect members within the module compared to paths that connect one module with another.

The MCL algorithm deterministically computes these probabilities of random walks using *stochastic matrices* or *Markov matrices*. A matrix is called column stochastic if all the entries per column sum up to one. Any graph, weighted or not, can be transformed into a matrix by using one row/column per node and setting every entry in the matrix to the edge

weight of the edge between the nodes indicated by the row/column index. By dividing every entry by the column total, this matrix is made stochastic (fig. 1.2).



$$\begin{bmatrix} 0 & 0.25 & 0.33 & 0.33 & 0 & 0 & 0 \\ 0.33 & 0 & 0.33 & 0.33 & 0 & 0 & 0.33 \\ 0.33 & 0.25 & 0 & 0.33 & 0 & 0 & 0 \\ 0.33 & 0.25 & 0.33 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 & 0 & 0.50 & 0.33 \\ 0 & 0 & 0 & 0 & 0.50 & 0 & 0.33 \\ 0 & 0 & 0 & 0 & 0.50 & 0.50 & 0 \end{bmatrix}$$

Figure 1.2: Transforming a graph into a stochastic matrix. The unweighted graph on the left is represented as a stochastic matrix on the right. Row and column indices correspond to node labels. Note how the modules in the graph are quite visible in structure of the matrix. The MCL algorithm aims to further enhance this property (Example adapted from course notes of Steven Maere, Ghent University).

By taking a power p of a stochastic matrix $\boldsymbol{M}$, every entry m_{ij}^p will correspond to the probability of ending in node i after making p random walks starting from node j. These probabilities are used by the MCL algorithm to determine a clustering. A problem though is that after a certain value for p, the entries in $\boldsymbol{M}^p$ will have stabilized, and cluster structure will no longer be visible in the matrix. To counteract this, the MCL algorithm will *inflate* the probabilities after every *expansion* (=taking a power of) the stochastic matrix. Inflation of the probabilities is done by raising every entry in the stochastic matrix by a power r and then rescaling the columns to make the matrix stochastic again. This operation will make the relatively higher probabilities even higher while lowering the lower probabilities.

To summarize, the MCL algorithm works in three steps. First the graph is represented as a stochastic matrix.[2] Second, the matrix is iteratively expanded with $p = 2$ and inflated with power r. The expansion step will compute random walk probabilities, which will be higher between nodes within a module, and the inflation step boosts these probabilities even further. After 3 to 10 iterations, the procedure will converge, setting many entries in the matrix to approximately 0. Third, the matrix is converted back into a graph which, after convergence of the algorithm, will be partitioned in a number of connected components. Every such component is then interpreted as a module.

There are a few nice properties about the MCL algorithm. The outcome, although based on probabilities, is deterministic. The granularity of the clustering can be tuned by changing the inflation parameter r. The algorithm itself is an elegant and simple succession of mathematical operations, instead of a set of rules, giving it a predictable runtime complexity of $\mathcal{O}(Nk^2)$ (N is the number of nodes, k is the maximum node degree). MCL has been implemented in the C programming language with support for multi-threaded com-

[2] When expanding a stochastic matrix with power p, there is strong effect from p being even or odd. To counteract this, self loops are added to every node before casting the graph into a stochastic matrix. This leads to more granular clusters and removes the parity effect. Explaining this would take us too far, see van Dongen (2000) page 43 and 44 for an example of this parity effect.

putation, making it orders of magnitude faster than MGclus. One disadvantage of MCL is that it does not work very well to find modules that are very large compared to the complete graph. In this setting, large modules are easily split into several small but highly connected modules.

To conclude the section about clustering—there is no real answer to the question of which algorithm makes better modules. Better modules are a very subjective matter, in fact the best algorithm depends on the context in which clustering is used.

1.6 Outstanding challenges

Here I will address two big and somewhat related problems in the field of pathway analysis. The first problem is that more and more tools are being created without the availability of a proper benchmark. The second problem is that there are so many pathway analysis tools that it is now becoming impossible to decide which tools to use for any given experiment, as the quality of many tools is never directly compared. Of course there are other problems as well, many of which have been discussed in sections 1.3 and 1.4.

1.6.1 How to benchmark pathway analysis?

Many pathway analysis tools have been published, but remarkably, most of these have not been benchmarked properly. There is currently no established method for benchmarking pathway analysis methods and there is little training data available. A perfect training dataset would contain experimental datasets where, for every dataset and for each pathway, it is known whether they are enriched or not. Several methods have been tried to match up to such training data.

The first method is to artificially generate positive and negative examples. The positive examples are created by taking gene sets from known ontologies and splitting them in two parts (while sometimes allowing a small overlap between both halves). This is then used to determine how well a pathway analysis tool can find the other halve, which gives an idea of the sensitivity of the tool. The negative examples are generated by taking gene sets from experiments or from known annotations/pathways and randomising them. The pathway analysis tool is then used to "enrich" the randomised gene sets, and every hit is counted as a false positive, giving an idea about the specificity of the tool. This method of generating training data (and variations thereof) is most commonly used. For example, the authors of CrossTalkZ (McCormack *et al.*, 2013) and RIDDLE (Wang *et al.*, 2012) use this method for performance evaluation. The biggest disadvantage of this approach is that one can never be sure that the performance on the generated data is similar to the performance on real data. It is all too easy to, consciously or unconsciously, generate training data that works well with the assumptions of the tool you designed.

The second approach, which was used by the authors of EnrichNet (Glaab *et al.*, 2012) for example, is to use a number of methodologically different enrichment tools to enrich the same gene sets. The intersection of their results is then used as "gold standard"

positive data. The idea originates from consensus methods in machine learning, where the consensus of many methodologically different classifiers often makes very reliable results, even if the classifiers themselves perform poorly. In some sense, pathway analysis could be seen as some sort of classification problem, where there are possibly multiple true classes or enriched pathways for every input data point. This approach unfortunately does not provide any negative data. It is not because a pathway is not found significant by any tool that it is certainly not relevant to the experiment or condition at hand. Since all tools have limitations, they might all fail to find a relevant pathway. In addition, the intersection of different tools provides you with a subset of all relevant pathways that are relatively easy to find. Relevant pathways that are difficult to find will be recovered by none or only a few tools. When a tool performs well on this data, it can safely be assumed that it is good in finding pathways that other methods find as well, but it does not tell you whether the tools extends coverage to unknown relevant pathways.

The third and so far most promising approach makes use of a small number of microarray datasets. Each dataset contains samples of healthy human tissue and samples associated with a human disease. In addition, for each of these diseases there is a KEGG pathway driving the disease phenotype. It therefore relatively safe to say that the KEGG pathway underlying the disease should be differentially expressed and therefore should be found significant by a pathway analysis tool. Although this assumption cannot be guaranteed, it is the most objective benchmark with real biological data that has been used so far. This benchmark was introduced by Tarca *et al.* (2013), and has also been used later by Dong *et al.* (2016). A modified version of this benchmark has also been used in this thesis, see section 3.1 for a more in depth review.

In fact, most published tools have not been statistically evaluated at all. In most cases, the tool is presented giving a theoretical background to show what it interprets as enrichment and why it should perform well—in theory. Sometimes a small case study is included where the output of the new tool is compared to other methods, but this is hardly sufficient to prove that the new tool brings an improvement over previous methods. By this I do not want to imply that all these tools have poor performance, only that we do not know whether they are better or worse than previously existing tools.

To make any advancements in the field of pathway analysis, it will be necessary to establish a standardized benchmark using real biological data. Without this, we cannot know whether new tools using more sophisticated models are actually an improvement or just a guess in the dark.

1.6.2 Tool overload

A major challenge for researchers who want to make use of enrichment tools is the myriad of tools available. A paper by Huang *et al.* (2009) listed 68 tools in all three categories of pathway analysis, and a more recent paper by Mitrea *et al.* (2013) lists another 22 PT based tools. These two lists are not exhaustive, and there are probably over a hundred different tools available for the same purpose of pathway analysis. In addition, many tools allow for using different databases, creating an exponential explosion of possibilities.

Examining online fora (Biostar, forum and ResearchGate forum) the decision of which tool to use is mostly based on the popularity of the tool and user friendliness. Newer and thus less known tools are less frequently used, even if they overcome major methodological limitations of older tools (Khatri *et al.*, 2012).

Part 2: Aim of Research Project

The aims of the study are to establish a robust benchmarking procedure for evaluating the performance (sensitivity and specificity) of overlap-based pathway analysis tools and the newer network-based crosstalk analysis tool BinoX. Once this benchmarking procedure is established, I will use this to assess the effect of clustering on the performance of enrichment tools. If clustered modules are more "pure" (they contain more genes from the same pathway/biological process/etc.), then sensitivity—and, possibly, False Positive Rate (FPR)—might be increased by running enrichment tools on separate modules instead of the entire list at once. The best performing tools, with or without clustering, will then be used to enrich pathways in tissue specific gene lists from the Human Proteome Atlas (Uhlen *et al.*, 2010, 2015).

2.1 Benchmarking pathway analysis methods

The problems and possible solutions for benchmarking pathway analysis methods have already been introduced in section 1.6.1. In this thesis, I will use an extension of the third approach which was introduced by Tarca *et al.* (2012). This benchmark is based on real data, which is preferable over simulated data because one cannot be certain that findings based on simulated data will also be true in real use case scenarios. A part of the workflow of this benchmark will have to be changed however to meet the needs of BinoX and to guarantee certain quality criteria; this is explored in detail in section 3.1. In addition to the overlap based tools and BinoX, I will also include the PADOG tool, made by the same authors as the benchmark. This will allow comparison of results obtained by the benchmark presented here with earlier results (Tarca *et al.*, 2012; Dong *et al.*, 2016).

2.2 Assessing the effect of clustering

We assume that gene sets derived from experiments are noisy and comprise multiple pathways and/or biological processes. This can make it more difficult for pathway analysis methods to detect relevant pathways. Clustering the genes in the gene set prior to running the tools can be used to counter this: instead of using the entire gene set, the tools will be used for every module separately. The gene set can be clustered based on a FA network. Any network based clustering tool can be used for this, examples are MCode

(Bader and Hogue, 2003), FastCommunity (Clauset *et al.*, 2004), MCL (van Dongen, 2000) and MGclus (Frings *et al.*, 2013), of which the latter two will be in used in this thesis. The main motivation for doing this is that genes in one module are more likely part of the same biological process or pathway. In addition noisy genes might be removed by clustering into single gene modules (fig. 2.1). Although this strategy might improve sensitivity, it might also increase the false positive rate. I will evaluate the trade-off between sensitivity and false positive rate to determine whether clustering results in an improvement or not.

2.3 Analysing the Human Proteome Atlas

Running the overlap based tools and BinoX—with and without clustering—on the Human Proteome Atlas (HPA) might give more insight into each tool. Is it necessary to run several tools or does one method find everything that the other methods find combined? Do several tools find the same aspects of the underlying biology or are they complementary? Are there important pathways that are missed by the tools without clustering, but can be found with clustering?

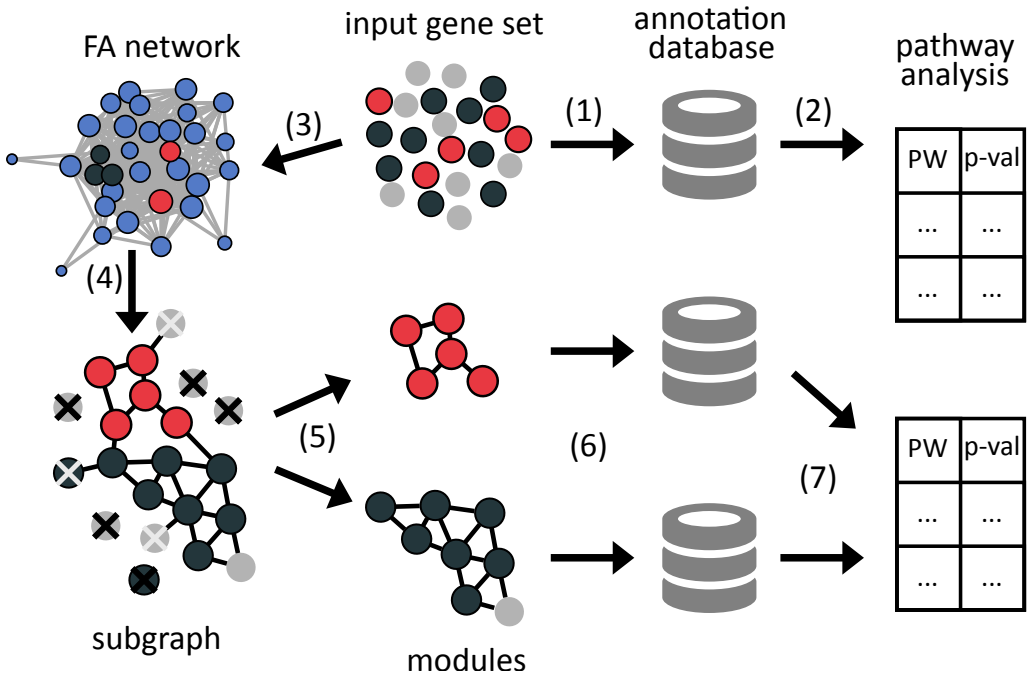


Figure 2.1: Combining pathway analysis with clustering. The traditional form of pathway analysis is to take a gene set in its entirety and enrich that to pathways from an annotation database (1). Grey genes represent "noise": genes that ended up in the gene set due to technical or biological variation but are not actually DE. Dark green and red genes are genes that belong to two different pathways that have an important role in the condition/experiment that led to the input gene set. The traditional method will, for most tools, give one p-value per pathway (2). An alternative could be to first extract the subgraph comprising the nodes of the input gene set from a FA network (3) & (4). By extracting the subgraph, some genes will become completely disconnected or might not have been in the FA network to begin with. These genes are then removed, thus extracting a subgraph can lead to a loss of information (black crosses). Then, the subgraph is clustered (5) and in the ideal scenario, different pathways would cluster together in the same module. Poorly connected genes, primarily "noise" genes in the ideal case, might end up in single-node modules and are removed before doing further analysis (white crosses). Finally, all modules are enriched separately to pathways in an annotation database (6) and the results are combined (7). Both the clustered and unclustered analysis might be combined into a meta analysis to get a complete picture of the underlying biological processes.

Part 3: Results

3.1 Benchmarking pathway analysis methods

A benchmark for pathway analysis methods is necessary in order to determine if clustering can be combined with pathway analysis in a beneficial way. As mentioned in section 1.6.1, a standardized benchmark for pathway analysis is currently absent. Here I will use an adaptation of a benchmark proposed by Tarca *et al.* (2012), which I will now discuss in more detail.

Tarca *et al.* (2012) have assembled a collection of 42 microarray datasets, all available at the Gene Expression Omnibus (GEO, Edgar *et al.*, 2002 and Barrett *et al.*, 2013), hosted by the National Center for Biotechnology Information (NCBI). Each of these datasets contains a number of healthy human tissue samples and a number of human tissue samples associated with a disease. In addition, for each of these diseases, a pathway driving the disease is known. In other words, each of the 42 datasets is coupled with one pathway that is know to be important for the studied phenotype. Pathway analysis methods aim to find pathways underlying expression data or a gene set, which means they should be able to find the known disease pathway for every dataset. This forms the basis for the sensitivity test: the sensitivity of a pathway analysis method is the number of times the known disease pathway is found to be significantly enriched, divided by the total amount of datasets.

Another important aspect of a pathway analysis method is the prioritisation, i.e., when sorting all pathways based on their significance, is the known disease pathway then found somewhere close to the top of the list? A tool might for example label the known disease pathway as significant for every dataset, but this might be only due to a very high false positive rate, and the ranking might still be poor. On the contrary, a tool might have a very poor sensitivity but might still rank the known pathway close the top of the list. Meaning this tool systematically underestimates significance, but is still capable of setting biologically important pathways apart from irrelevant pathways.

A third important measure to assess the quality of a tool is its specificity, and this is quite difficult to measure due to the lack of negative data. Two methods are often used in the literature to generate negative data. One is based on microarray or RNA-seq based workflows: all sample labels are permuted and the pathway analysis is carried out on the same data with the permuted sample labels. But, as mentioned in section 1.3.2, the FCS generation of tools uses exactly this as a null hypothesis for determining significance. Thus,

the specificity of FCS methods should always be spot on using this approach for specificity testing. For network based methods another method, which is compatible with gene set based workflows, has been used for benchmarking the specificity. Here, a collection of actual gene sets is taken (e.g. from Molecular Signal DataBase (MSigDB), Subramanian *et al.*, 2005) and all genes are replaced with other genes having similar node degree in a FA network. Regardless of how the negative data is generated, the specificity is defined as the proportion of gene sets that is found to be significant amongst all random gene sets.

Known problems Unfortunately, this benchmark is far from perfect. The sensitivity test only tests how well known disease pathways can be found. A tool that performs well on this test will not necessarily perform well in another context (e.g. in different organisms or when analysing responses to environmental stimuli). There is also a problem with the prioritisation test, even when confining ourselves to the analysis of diseased tissues. The known disease pathway does not necessarily have to rank first on the list, because it will not be the only pathway that is affected by the disease. E.g. when analysing pancreatic cancer tissue, a known affected pathway is the *pancreatic cancer* KEGG pathway (hsa05212), but another pathway that will probably be affected is the *p53 signaling* pathway (hsa04115). We cannot state with certainty which of these two pathways should rank before the other. Finally, the approaches for generating negative data are not perfect either, there will always be a difference between simulated data and real data. Unfortunately, real negative data is hard to find: it is difficult to prove that a certain pathway is absolutely not important for a given disease.

3.1.1 General overview of the benchmark

An adaptation of the benchmark used by Tarca *et al.* (2012) has been implemented in a new `R` package called `BinoX`. The general workflow of the benchmark is shown in fig. 3.1. Here, I will also introduce some terminology that I will use later when describing the benchmark and the results. In section 3.1, I introduced the concept of disease associated datasets and their known affected pathways. From hereon, I will refer to these datasets as the *query dataset* and the known pathway for a given *query dataset* will be referred to as the *target pathway*. Any other pathway is a *non target pathway*. Many pathway analysis tools cannot work with expression data directly but require a gene set instead, any gene set for which a *target pathway* is known is called a *query gene set*.

Workflow First the datasets are downloaded from the Gene Expression Omnibus (GEO). For testing sensitivity and prioritisation (fig. 3.1, solid arrows), the following steps are carried out: first, probe-wise test statistics are computed (details in section 5.1), then probesets are translated to gene identifiers supported by the pathway analysis tools to be tested (see section 5.2). Then, after transforming the obtained data in the correct input format, all the preprocessing required for FCS and PT tools is done. For ORA and network based methods, one additional step is required, which is extracting a gene set using the test statistic and optionally the fold change. These gene sets are either directly used as input for ORA and network based methods, or they can undergo one additional

step: clustering the gene set using a FA network. Finally the output of all tools is collected to compare their performance in terms of sensitivity and prioritisation.

For specificity testing, two approaches are used: label swap and gene set permutation. For the label swap, sample labels are permuted and the exact same workflow follows as for the sensitivity test (fig. 3.1, black dotted arrows). For the gene set permutation, the same workflow is followed as the sensitivity test, except that genes in the genes sets are replaced with new genes with similar (within 5%) node degree before running any tool (fig. 3.1, grey dotted arrows). The latter approach can of course not be used for FCS and PT methods.

The complete pipeline, from downloading the gene expression data to running the pathway analysis methods on positive or negative data is implemented in the `BinoX` package (section 3.3). A detailed description of the whole procedure is given in section 5.4.1.

3.1.2 Gold standard data

Tarca *et al.* (2012) have assembled 42 datasets for which a target pathway is known. Unfortunately, 8 of these datasets have a Metacore pathway as target, and Metacore is a proprietary database. These 8 datasets are therefore dropped from the benchmark. In addition, another 8 datasets contain no differentially expressed genes are therefore also dropped from the benchmark (see section 5.1 for details on determining differential expression). The final gold standard contains 26 microarray datasets, all of which have at least 10 differentially expressed genes using a BH adjusted p-value cutoff of 0.01. An overview of the gold standard data is given in table 3.1, additional information is given in appendix A.1.

3.1.3 Comparing BinoX with earlier methods

Before testing BinoX in combination with clustering, it would be interesting to know how good it performs compared to other tools. Therefore, BinoX was compared against the previously established methods using the benchmark implemented in the `BinoX` package. It would be interesting to re-do the exact same benchmark as in Tarca *et al.* (2012) or Dong *et al.* (2016), because then the outcome for BinoX could be directly compared to 16 other tools. Unfortunately, BinoX relies on the FunCoup network which uses Ensembl identifiers (Cunningham *et al.*, 2014), while the two previously mentioned publications use Entrez IDs (Maglott *et al.*, 2004). In addition, the same data could not be used either (see section 3.1.2). Yet another difference is the method used to extract gene sets after determining differential expression (section 5.1). Tarca *et al.* (2012) and Dong *et al.* (2016) use a three-step method to select which genes to include in a DE gene set: *1)* take all genes with a q-value cutoff below 0.1, *2)* if this set is smaller than 200, then take all genes with p-value below 0.05 and a fold change above 1.5 and *3)* if this set is again shorter than 200, then take the top 1% DE genes. This method is rather arbitrary and in some cases selects over $10\,000$ genes. Since both up and downregulated genes are selected, the gene set sizes can get quite big. In addition, if the sample sizes are very large then—even if

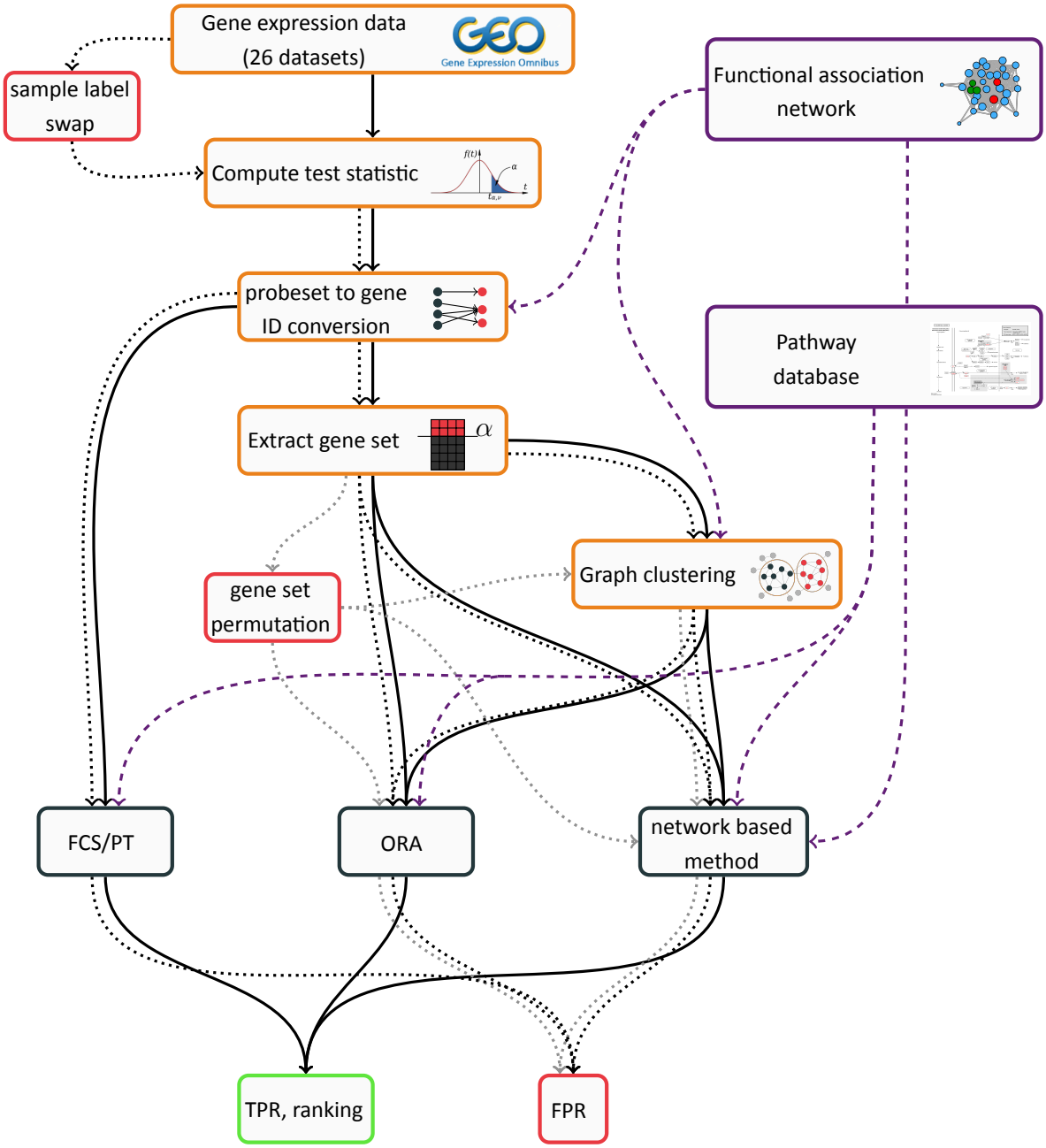


Figure 3.1: Benchmarking pipeline. General workflow of the benchmark for testing sensitivity, prioritisation and specificity of pathway analysis methods. Black arrows represent workflows that can be taken for sensitivity and prioritisation tests. Dotted arrows represent workflows for specificity testing, with either label swap (black dotted arrows) or gene set permutation retaining node degrees (grey dotted arrows). Preprocessing steps are shown in orange boxes, pathway analysis tools are shown in dark green boxes. Some preprocessing steps and the pathway analysis tools require external information, which is depicted with purple boxes and purple dashed arrows. All workflows are supported by the `BinoX` package. In addition, parallelisation is supported for nearly all steps.

GEO	target pathway	KEGG	n	a	b	paired	reference
GSE1145	Dilated cardiomyopathy	hsa05414	26	15	11	no	no reference
GSE3467	Thyroid cancer	hsa05216	18	9	9	yes	He *et al.* (2005)
GSE3585	Dilated cardiomyopathy	hsa05414	12	7	5	no	Barth *et al.* (2006)
GSE3678	Thyroid cancer	hsa05216	14	7	7	yes	no reference
GSE4107	Colorectal cancer	hsa05210	22	12	10	no	Hong *et al.* (2007)
GSE4183	Colorectal cancer	hsa05210	23	15	8	no	Galamb *et al.* (2008); Gyorffy *et al.* (2009)
GSE5281	Alzheimer's disease	hsa05010	21	9	12	no	Liang *et al.* (2007, 2008)
GSE5281	Alzheimer's disease	hsa05010	23	10	13	no	Liang *et al.* (2007, 2008)
GSE5281	Alzheimer's disease	hsa05010	31	19	12	no	Liang *et al.* (2007, 2008)
GSE7305	Endometrial cancer	hsa05213	20	10	10	yes	Hever *et al.* (2007)
GSE8671	Colorectal cancer	hsa05210	64	32	32	yes	Sabates-Bellver *et al.* (2007)
GSE9348	Colorectal cancer	hsa05210	82	70	12	no	Hong *et al.* (2010)
GSE9476	Acute myeloid leukemia	hsa05221	63	26	37	no	Stirewalt *et al.* (2008)
GSE14762	Renal cell carcinoma	hsa05211	21	9	12	no	Wang *et al.* (2009)
GSE14924	Acute myeloid leukemia	hsa05221	20	10	10	no	Le Dieu *et al.* (2009)
GSE14924	Acute myeloid leukemia	hsa05221	21	10	11	no	Le Dieu *et al.* (2009)
GSE15471	Pancreatic cancer	hsa05212	70	35	35	yes	Badea *et al.* (2008)
GSE16515	Pancreatic cancer	hsa05212	30	15	15	yes	Pei *et al.* (2009)
GSE18842	Non-small cell lung cancer	hsa05223	88	44	44	yes	Sanchez-Palencia *et al.* (2011)
GSE19188	Non-small cell lung cancer	hsa05223	153	91	62	no	Hou *et al.* (2010)
GSE19728	Glioma	hsa05214	21	17	4	no	Liu *et al.* (2011)
GSE21354	Glioma	hsa05214	17	13	4	no	Liu *et al.* (2011)
GSE23878	Colorectal cancer	hsa05210	38	19	19	yes	Uddin *et al.* (2011)
GSE24739	Chronic myeloid leukemia	hsa05220	12	8	4	no	Affer *et al.* (2011)
GSE24739	Chronic myeloid leukemia	hsa05220	12	8	4	no	Affer *et al.* (2011)
GSE32676	Pancreatic cancer	hsa05212	32	25	7	no	Donahue *et al.* (2012)

Table 3.1: Gold standard data. The number of disease and healthy tissue samples are given in columns a and b respectively, the total is in column n. When *paired* is *yes*, a paired microarray design was used (see section 5.1). In the digital copy of the thesis, GEO and KEGG IDs are hyperlinks.

the biological variation is high—the variation of the sample mean can become extremely small. Moreover, normal regulation can be disturbed so much in cancer tissues that for nearly all genes there is at least a little difference in true mean expression. In conclusion, this method leads to very large gene sets where many genes show a small—less than 10%—fold change. Instead of this three step method, gene sets were selected on two criteria: *1)* a gene must have a q-value lower than 0.01 and *2)* a fold change of at least 1.5 (either 50% up or down regulated).

Finally, the FPR test is also done differently. In the previous publications, the phenotype labels of the microarray data where permuted and gene sets were extracted using the same test statistic and three-step method as described above. These gene sets were then enriched and any significant enrichment was counted as a false positive. After the sample label permutations, there are almost never DE genes when using a q-value cutoff of 0.01, therefore step *1)* would almost never be used, and step *2)* would only sometimes be used. In the majority of cases, gene sets would be selected by step *3)*, leading to much smaller gene sets for the FPR test than those used for the True Positive Rate (TPR) test. Instead, here I also permute phenotype labels and take the top n genes where n is drawn from the gene set sizes used in the TPR test. However, a problem with the phenotype label swap is that some of the original signal might still be retained. To completely eliminate this possibility, a second FPR test is used as well where random gene sets are generated by replacing every gene in the original gene set with a new one having similar node degree in FunCoup. For both FPR tests, all 26 datasets were permuted 10 times, leading to a total of 260 random gene sets (or microarray datasets) in each scenario (see sections 5.3.1 and 5.3.2 for the exact procedure).

In the end, four aspects differ from previous publications: *1)* Ensembl IDs are used instead of Entrez IDs (see section 5.2.1), *2)* datasets with a Metacore target pathway or less than 10 DE genes are not used, *3)* a different gene set selection method is used and *4)* a different approach is used for the FPR test (sections 5.3.1 and 5.3.2). The exact protocol for the whole benchmarking procedure is given in section 5.4.1.

Results

Figure 3.2 shows the results of the benchmark. BinoX certainly is the most sensitive tool, although it also has the highest FPR. PADOG is in this perspective almost opposite to BinoX, having the lowest sensitivity but also the lowest FPR for any p-value cutoff below 0.20. The Fisher and EASE methods fall somewhat in the middle and, as expected, the EASE method has a lower FPR than the Fisher method, but is worse in terms of sensitivity. Figure 3.2b visualizes the performance of each tool in terms of prioritisation. The *rank percentage* is defined as $i/n \times 100$ where i is the rank of unadjusted p-value[1] of the target pathway in all n pathways that were tested. In terms of ranking, PADOG seems to be the clear winner, although the other methods don't fall far behind. It should be noted that PADOG has

[1] Note that adjustment with the BH method is a monotonic function of the p-values. Therefore, sorting on p-value or q-value should not make a difference. That being said, p-values close to 1 can become equal to 1 after adjustment, losing their order in the process. This is why, in this thesis, ranking is always done on the unadjusted p-values.

two clear advantages in this benchmark. Firstly it uses all the microarray data available, whereas the other tools can only use the gene sets derived from the microarray samples. Secondly, it is a FCS based tool, which uses sample label swaps as the null hypothesis to determine an empirical p-value—this is the same null hypothesis as the first FPR test, so it is obvious why it scores perfectly on this test. Any deviations from a 1:1 ratio (fig. 3.2c, dashed line) are a result from the discrete number of iterations of label swaps (50 in this case, the default setting), the more iterations used, the closer the convergence to a 1:1 ratio.

In the end all tools performed reasonably well. Of the 26 target pathways, 11 were recovered by all tools. BinoX found all pathways that were also found by other tools, and three pathways were found by BinoX only (fig. 3.4a). For two datasets, the target pathways were not found by any tool (Chronic myeloid leukemia, hsa05220 and Alzheimer's disease, hsa05010), but these pathways were successfully recovered from other datasets having the same target pathway.

Differences between FPR tests Using the sample label swap (fig. 3.2c), specificity is much worse for every tool compared to the gene set permutations (fig. 3.2d). According to a growing consensus in the literature, this is due to the tools not taking into account gene-gene correlations (section 1.3.2). Although it might also be that some of the signal is still preserved after swapping sample labels. Anyhow, both tests show that BinoX has the highest FPR, followed by Fisher and then EASE.

Comparison with earlier publications Although many steps in the benchmark were done differently, the results for the EASE tool and for PADOG are qualitatively similar to what has been published before (Tarca *et al.*, 2012; Dong *et al.*, 2016). That is, PADOG outperforms EASE in prioritisation, but has a lower sensitivity whereas EASE has a lower specificity.

Effect of gene set selection It is likely impossible to benchmark FCS methods and set based methods such as BinoX or ORA tools against each other, as any benchmark would include a crucial step where gene sets have to be extracted from microarray data. This step influences the performance of all non FCS or PT tools only. To illustrate this, the same benchmark was repeated but the size of gene sets was limited to a maximum of 600 genes, which is slightly more than the biggest KEGG pathway (Olfactory transduction, hsa04740). Many DE genes are no longer included in the gene sets, leading to a worse performance of set based tools (fig. B.1). Thus, although results from BinoX, EASE and Fisher are directly comparable, they should be compared to PADOG with some caution.

3.2 Assessing the effect of clustering

The second objective is to assess whether clustering can be used to increase the performance of set based tools, in particular BinoX, EASE and Fisher. The original idea was that by clustering a gene set based on a FA network, you would get several big modules, and

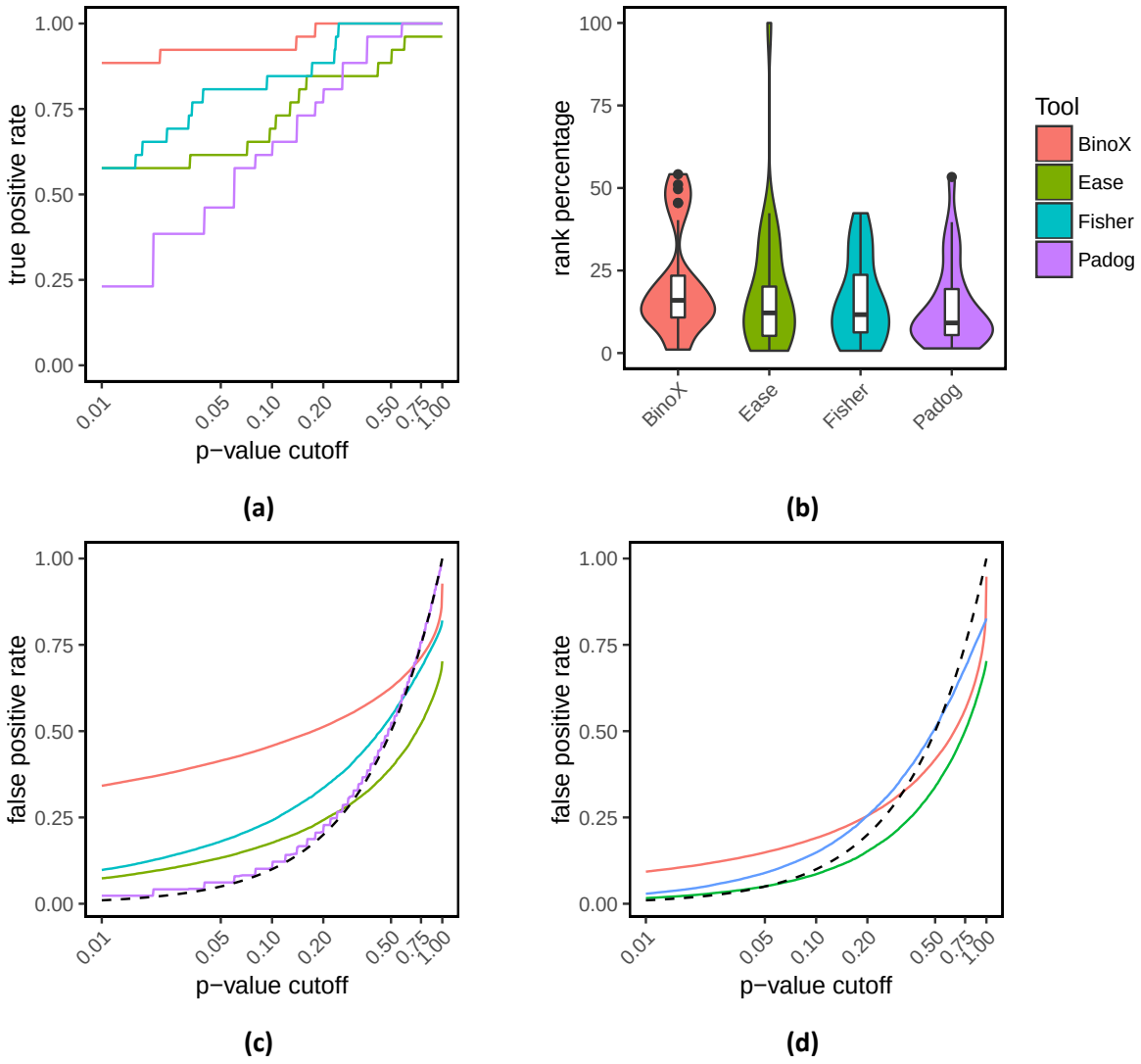


Figure 3.2: Comparing BinoX with earlier tools. The graph based tool BinoX, the two ORA methods, Ease and Fisher and the FCS tool PADOG are compared against each other using the gold standard data from table 3.1. Gene sets were extracted using a BH adjusted p-value cutoff of 0.01 and a fold change cutoff of 1.5. True positive rates for each tool are shown in (**a**) for all significance cutoff values from 0.01 to 1 (unadjusted p-value). Notice the *log10* scale on the x-axis. Rank percentage of the target pathway is shown in (**b**). The value distributions of the 26 data points are shown as violin plots, any target pathway that cannot be tested by the tool is given a rank of 100%. Boxplots represent value distributions for all target pathways the tool can test for (e.g. for EASE there is one target pathway that has no overlap with the gene set, and is hence is set 100% for the violin plot but is not included in the boxplot). False positive rates for all tools are given in plots (**c**) and (**d**), using phenotype label swap or gene permutation respectively (section 5.3). Gene permutation is not an option for PADOG Because it is an FCS tool that requires microarray data as input. The dashed black line indicates the 1:1 ratio. Ideally, any methods FPR should be on or below this line. Notice again the *log10* scale on the x-axis. Colors are given in the top right, and are the same for all four plots. Figures (**a**), (**c**) and (**d**) are also given with q-values in fig. B.6.

genes from the same pathway, would be brought together in these modules. Unrelated genes that end up in the gene set by accident (e.g. due to technical variability) would not cluster together and effectively be filtered out by the clustering process. In the end, you would obtain several big clusters that are "purified" from the gene set, and the small clusters would be of no importance. By taking for each pathway the module with the lowest p-value, you would preferentially select one of these bigger modules, which should be more pure. Hence the sensitivity would be increased by enriching the modules (instead of the whole gene set) to pathways from an annotation database.

To test whether this would work as expected, the benchmark from section 3.1.3 was repeated but the gene sets were clustered first with either MCL or MGclus. All genes that got disconnected from the graph were removed, and only clusters of at least two genes were considered for further analysis. Then, for each pathway, the lowest p-value among all modules was kept as the final p-value for this pathway. The results of this benchmark compared with the previous one are shown in fig. 3.3.

It is clear that, for all tools that were tested, this method does not have a beneficial effect. In fact, in all cases the sensitivity rose, but the FPR rose even more, a finding that is supported by both of the FPR tests. Both MCL and MGclus seem to have the same—unfortunately negative—effect on the performance of the pathway analysis tools, although the results with MCL are slightly less worse than those with MGclus. The increase in sensitivity and FPR is the most extreme for BinoX, starting of with a 70% false positive rate at a p-value cutoff of 0.01 (and also a FPR of 70% at a q-value cutoff of 0.01, see fig. B.7b). BinoX combined with clustering was able to recover all target pathways at a p-value cutoff of 0.05 (fig. 3.4b), but in the light of the extreme FPR this is not really an achievement.

Although the initial idea does not work out as was hoped for, it might still be possible to find some way to reduce the enormous FPR and thus make it work. In the following sections I will explore the results in more detail in an attempt to find out if there are ways to improve the current situation.

3.2.1 BinoX combined with clustering

Clustering increased the FPR for all tools, but for BinoX the increase was certainly the worst. A closer look to the modules created by either MCL or MGclus explains why this happens. The input gene sets are quite large, ranging from less than a 100 genes to slightly over $6\,000$, and it seems like both clustering algorithms create a large number of very small modules (2–3 nodes). Especially MCL is likely to make over a hundred of these small modules if the input graph is bigger than a thousand nodes. Figure 3.5 shows the adjusted p-values of all these modules for the random gene sets from both FPR tests and both clustering algorithms (260 random gene sets $\times$ ± 10–200 modules $\times$ 288 KEGG pathways is $\pm$ several millions of random module v.s. pathway combination tested).

Figure 3.5 shows some interesting differences in properties of both clustering algorithms. MCL always seems to make one very big module that is about half of the input size, and then a large number of small modules. Although all gene set sizes are mapped together

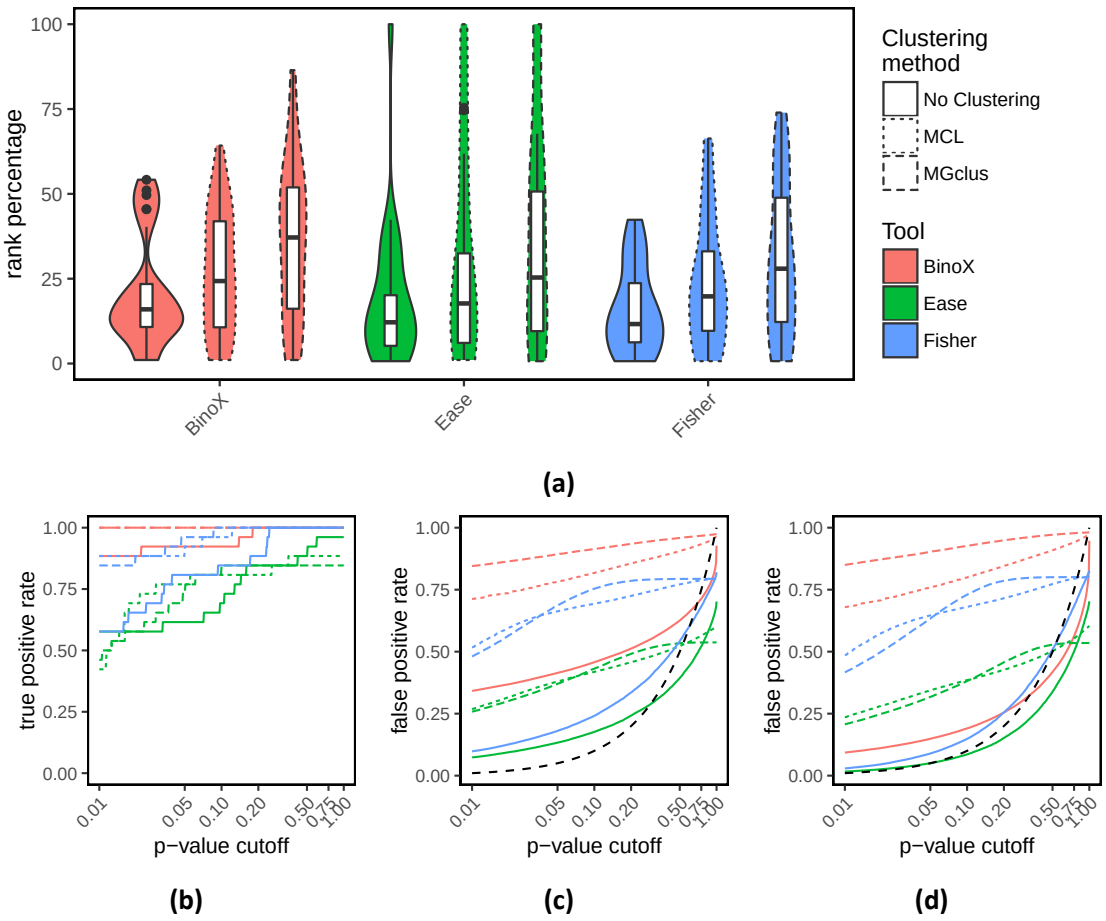
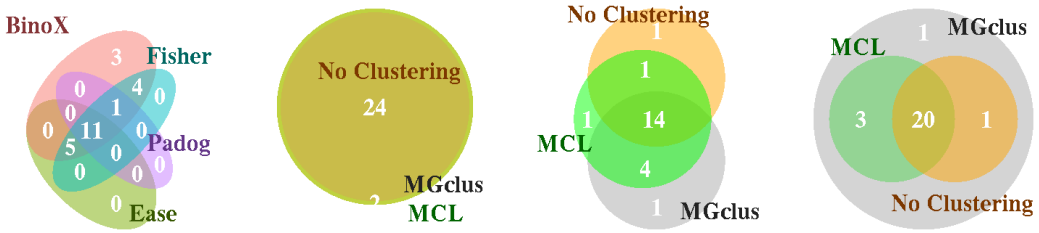


Figure 3.3: Taking the module with the lowest p-value does not improve performance of pathway analysis. The BinoX, EASE and Fisher methods were run on the same data as in fig. 3.2 but now in combination with clustering. Shown here are: (**a**) rank percentage, (**b**) true positive rates, (**c**) false positive rates using label swap and (**d**) false positive rates using gene permutations. Colors indicate the tool used and linetypes indicate the clustering algorithm. Boxplots and violin plots are as in fig. 3.2. Plots (**b**), (**c**) and (**d**) are also given with q-values in fig. B.7.



(a) No clustering **(b)** BinoX **(c)** Ease **(d)** Fisher

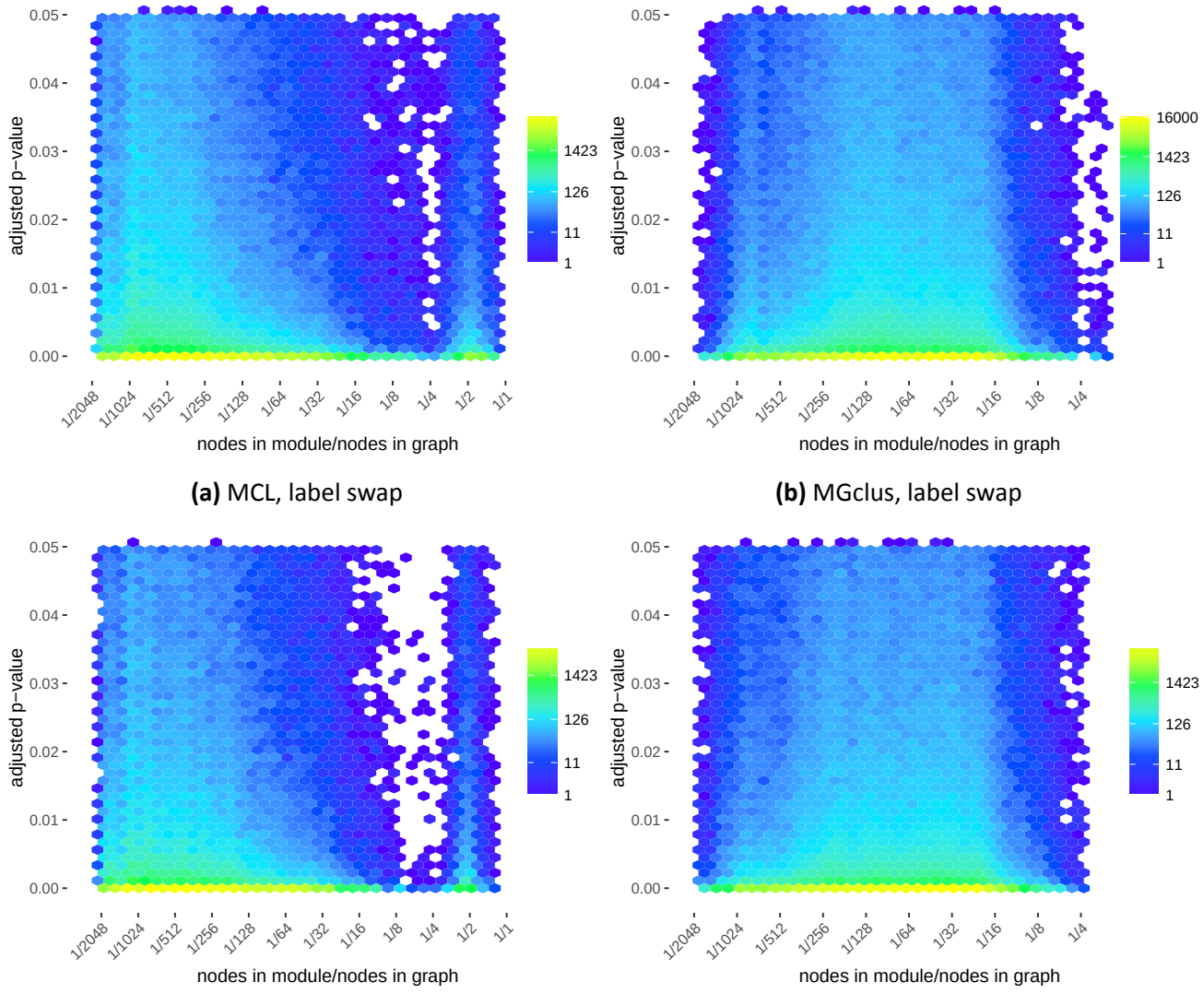
Figure 3.4: What is found by which tools?. Different tools and clustering methods lead too different target pathways being recovered. (**a**) Number of target pathways found by running the benchmark without clustering as explained in section 3.1.3 and with clustering: (**b**), (**c**) and (**d**). In (**b**), all target pathway are found for BinoX with both MCL and MGclus, whereas without clustering, 24 pathways are found. A p-value cutoff of 0.05 was used in every diagram.

on one graph here, the pattern seen here holds for any gene set size above a few hundred. MGclus, on the contrary, makes less modules in total that are more balanced in size.

Any p-value shown in fig. 3.5 is a false positive, and it looks like, at least for MCL, the huge amount of small modules are the main source of false positives. This might lead to a solution for the high FPR, i.e. by removing all modules that contain less than e.g. 1% of the nodes of the gene set, we might eliminate the most important source of false positives without harming the true positive rate too much. And maybe a similar trick might work for MGclus as well.

Unfortunately, fig. 3.6 shows that this is not the case. The size distribution of modules that show significant enrichment to their specific target pathway (= true interaction) seems to be identical to that of random modules that show significant enrichment to any KEGG pathway (= false interaction). That is, small modules whether real or random, give rise to low p-values. The original hypothesis that the true underlying signal in a gene set would cluster together in the same module(s) is not true. Instead, the target pathways are, in most cases, spread across a number of small modules. Other properties of the modules or properties of module-to-pathway combinations might show a different distribution for true interactions than for false interactions. For example, the node degree of a module or the number of links between a module and a pathway might be different for true interactions than for false interactions. Unfortunately, as shown in fig. 3.6b, this is not the case when looking at for example the average node degree of the module. In fact, there seems to be no feature that can be used to separate the true interactions from false ones.

Clustering does not add value to BinoX In conclusion, by clustering a gene set, especially a large one, noise gets clustered together almost as well as the true signal. The statistical test—the probability of observing a number of links in a binomial distribution in the case of BinoX—then evaluates this module in isolation, as if it were the entire gene set. In other words, the test statistic is calculated on the module without the context of the whole gene set, but the null hypothesis used to determine significance of the test statistic treats the module as if were the entire gene set, therefore the significance of this isolated module is overestimated. This effect is happening both for false and for true interactions. Significance for false module-to-pathway combinations gets overestimated, but it seems like significance for true interactions gets overestimated even more. Therefore, taking the lowest p-value per pathway might just as well be the least worst option to at least prioritise true interactions before false interactions. But the actual meaning of the p-value gets lost when testing a module in isolation, i.e. the p-value can not be interpreted as the probability that the observed test statistic—the number of links in the case of BinoX—is observed by chance. By using clustering, the probability of observing an extreme test statistic is increased enormously, and this is not accounted for. As the meaning of the p-value is lost by clustering, it can only be used as a ranking statistic, and even then it ranks worse than just using the normal p-value without the intermediary clustering step.



(a) MCL, label swap

(b) MGclus, label swap

(c) MCL, gene set permutation

(d) MGclus, gene set permutation

Figure 3.5: Distribution of false positives for the BinoX tool. Differentially expressed gene sets were extracted from the microarray data given in table 3.1 using an adjusted (BH) p-value cutoff of 0.01 and a fold change cutoff of 1.5 (50% up or down regulated). Using the same strategy, random gene sets were generated using phenotype label permutations of the same data (label swap, (**a**) and (**b**)). As an alternative, random gene sets were generated by taking the original gene sets and replacing each gene with a new gene having a similar node degree in the FunCoup network (gene set permutation, (**c**) and (**d**)). With both methods, 10 randomizations were done for each dataset. Finally, all randomized gene sets were clustered with MCL and MGclus and then all modules were analysed with BinoX versus all KEGG pathways. Shown here are the adjusted p-values of all module to pathway combinations that are below 0.05. The relative size of the module compared to the complete gene set graph is shown in *log2* scale on the x-axis. Colors correspond to counts after 2 dimensional binning of the p-values.

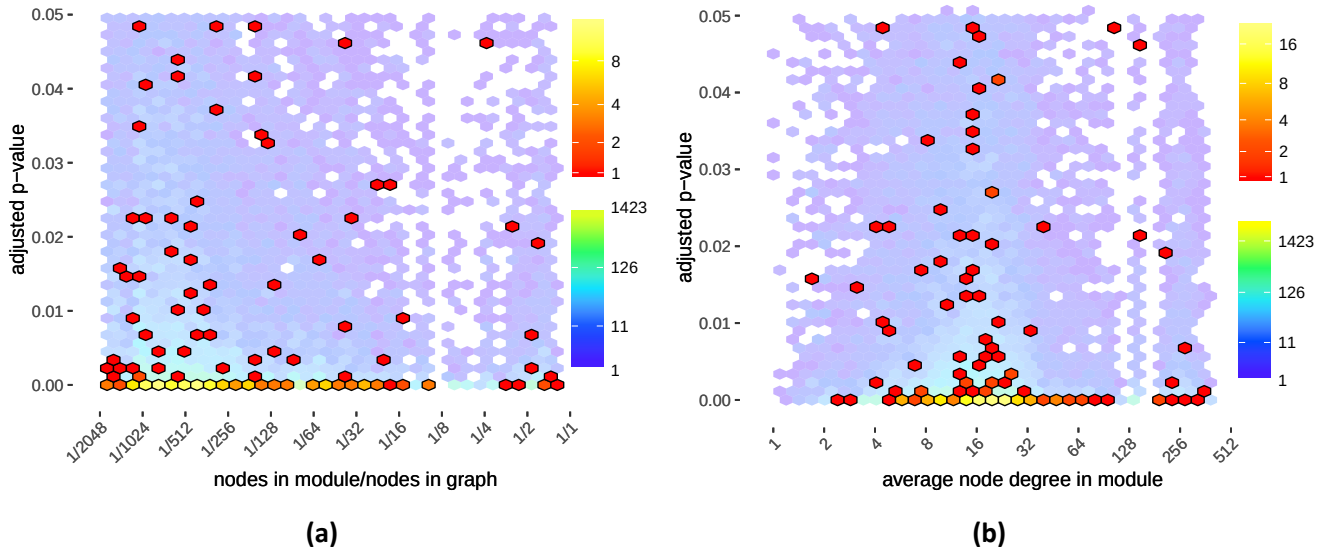


Figure 3.6: Distribution of true positives for the BinoX tool. Differentially expressed gene sets were extracted from the microarray data given in table 3.1 using an adjusted (BH) p-value cutoff of 0.01 and a fold change cutoff of 1.5. Gene sets were then clustered with MCL and the modules were enriched to KEGG using BinoX. Shown here is (**a**), the distribution of the BH adjusted p-values with respect to the relative module sizes (relative to gene set graph) and (**b**), to average node degree of nodes in the module. All p-values corresponding to a module and its target pathway (true positive) are colored according to the top color scale and the bins are surrounded by a black border. The other p-values (to non-target pathways, here interpreted as false positives) are shown in the background and are colored according to the bottom color scale. The same results using MGclus are shown in fig. B.3.

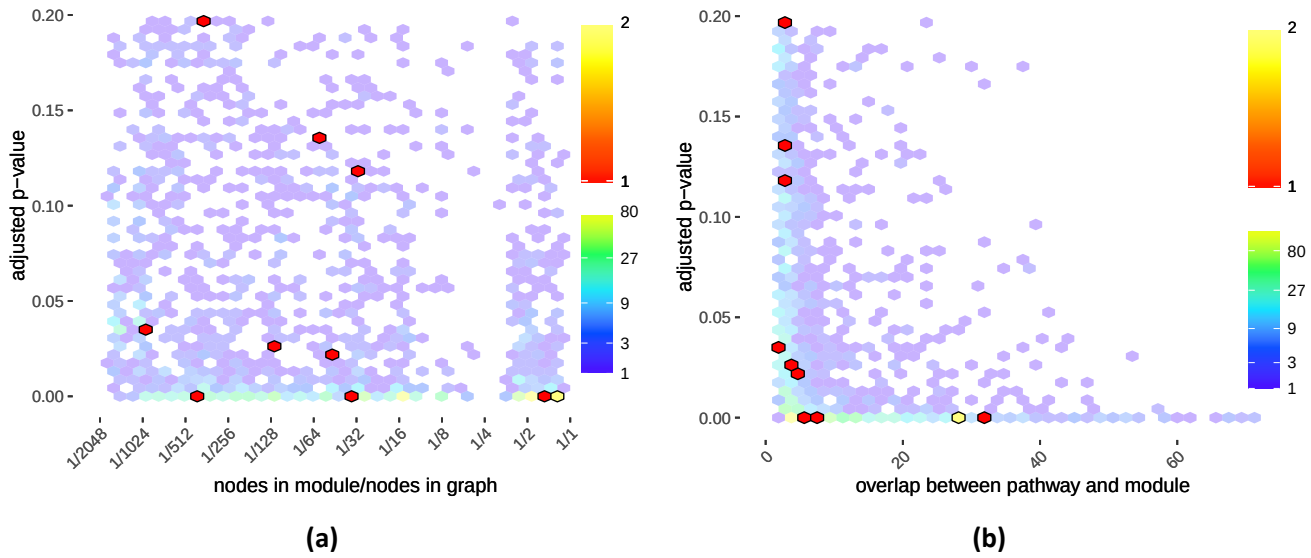


Figure 3.7: Distribution of true positives for the EASE tool. Gene sets were obtained and clustered with MCL as explained in fig. 3.6 and the modules were enriched to KEGG using the EASE score. Shown here is (**a**), the distribution of the BH adjusted p-values with respect to the relative module sizes and (**b**), to the number of overlapping genes between the module and the KEGG pathway. Color scales are the same as in fig. 3.6. Results obtained with MGclus are shown in fig. B.4.

3.2.2 EASE combined with clustering

As with BinoX in the previous section, the biggest source of false positives is again a large number of very small modules (fig. B.2). Unfortunately, the reasons why clustering does not work out for BinoX seem to be equally valid for the EASE method. There are no distinguishing features—such as module size or node size (fig. 3.7)—that set true interactions apart from false ones. Even if there was some way to filter out only the "good" modules, this would still not be very helpful, since for EASE the sensitivity increases little to nothing by introducing clustering (fig. 3.3b).

In conclusion it is save to say that, at least for the data tested here, there is not much to gain with clustering, neither with BinoX or EASE. Although not tested, this is probably true for the Fisher test as well, since the EASE method differs very little from the Fisher test.

3.2.3 Testing the effect of clustering on MSigDB data

So far, clustering has not been successful, but this is based on a small sample of only 26 datasets. To confirm what has been found so far, a second sample of 61 MSigDB datasets is used. MSigDB is a large repository where scientists can deposit gene sets that were found to be altered by a certain condition/experiment or that share a common property (Subramanian *et al.*, 2005). These gene sets are divided into 7 (C1–C7) broad categories and cover for example genes associated with chromosomal regions, genes sharing Tran-

scription Factors (TFs), curated gene sets such as KEGG or Reactome pathways and gene sets from transcription profiling experiments.

Here I use the MSigDB C2::CGP collection which contains over three thousand gene sets associated with chemical and genetic perturbations. These gene sets were download from the MSigDB website (`http://software.broadinstitute.org/gsea/msigdb/index.jsp`) and their names were matched against KEGG pathways, giving a list of gene sets that contained a KEGG pathway in their name. The descriptions of these gene sets were checked quickly to see if each gene set actually was relevant to the matched KEGG pathway. In addition, there are some gene sets that contain the suffix `UP` or `DN` in their name, indicating that the genes were up or downregulated in the experiment. Since pathway genes can be perturbed in both directions, these gene sets were joined together. In the end, a collection of 61 gene sets was left, each having an associated pathway that is likely to be affected. Finally, all KEGG and MSigDB gene sets are translated to Ensembl ID's to be able to make use of FunCoup (section 5.2.2).

The MSigDB data is used here as a new gold standard to validate previous findings. The microarray gold standard data used before is manually curated and matched to KEGG pathways, whereas the MSigDB gold standard introduced here is curated less carefully. Therefore, even though it has more data, it might be of lesser quality. An overview of the MSigDB gold standard data is given in table A.1.

Results

Clustering seems to have almost no effect on the prioritisation test, but the previously made observation that both TPR and FPR increase for all tools is also true here (fig. 3.8). Although it matters little which clustering algorithm is used for the ranking, a higher TPR and FPR are observed when using MGclus compared to MCL, which is in agreement with previous results. For BinoX, none of the modules of one of the 61 gene sets had any links with the target pathway and hence the rank was set to 100% in the violin plot in fig. 3.8a, but was omitted in the boxplot. For EASE, the modules of several gene sets had no overlap with the target pathway, so a few more ranks were set 100% (and omitted from the boxplot) compared to the non-clustered run. So, keep in mind that the boxplot for EASE after clustering is made on less data, and there is no real improvement in the overall ranking. For Fisher, all target pathways that could be assigned a rank without clustering could also be assigned a rank with clustering. In conclusion, except for the ranks being not much affected by the clustering, all other results seem to agree with previous findings.

When looking at BinoX, the lowest p-values for module to target pathway combinations do not originate from the biggest modules—instead, many small modules give rise to low p-values (fig. 3.9a). When looking at the average node degree within a module, true interactions seem to be similarly distributed to false interactions (fig. 3.9b).

When looking at EASE, there seems to be a slight preference for true interactions to originate from larger modules (fig. 3.9c). Note that the black dots in fig. 3.9c are module to non target pathway combinations. Although they are considered as true negatives here, they are not necessarily wrong, i.e. they might be biologically meaningful. One could ar-

gue that by removing small modules, the FPR problem would be solved without harming the sensitivity. But this is only supported for the few pathways that are found by EASE, and only on this data. When looking at the overlap, there seems to be a very slight preference for true interactions to have a slightly larger overlap than other interactions. But once again this is only supported by this data.

3.2.4 Can clustering improve pathway analysis?

The bigger the input gene set is, the easier it is for some random genes to group together in a module that will suddenly become "significant". This is observed in all the tests I conducted in this thesis: the larger the gene set, the worse the significance overestimation will be and conversely, the smaller the gene set, the lower the significance overestimation will be—as there is less noise that can cluster together in the first place. For example, when limiting gene set sizes too 600 (fig. B.1), the impact of the intermediary clustering step (compared to not clustering) is not as bad—although setting this arbitrary size limit of 600 is a bad idea in itself. When looking at the MSigDB gene sets (fig. 3.8), which are much smaller than 600 on average, the effect of significance overestimation is almost gone for the EASE and Fisher methods. There even seems to be a very small improvement for the Fisher method when using MCL, although the difference is so small that it might just as well be due to the low sample size.

Can we combine modules? The target pathway is most often spread across multiple modules of the gene set, but one could assume that the true signal in the gene set (= links or overlap to a target pathway) is usually spread across more modules than a randomly picked pathway. Therefore, taking into account the number of significant modules for a pathway might help to recover the target pathway. This assumption might be true for BinoX but does not hold for Ease (figs. B.10 and B.11). So for BinoX, tricks like taking into account the number of significant modules or somehow combining the p-values for different modules might work to get a good prioritisation. For EASE, when looking at the number of modules that show at least some significance to a pathway (p-value $<$ 1), it seems like target pathways have higher module counts (figs. B.12 and B.13). Hence, ranking pathways on this statistic might improve the prioritisation.

When trying things like multiplying the p-values or ranking pathways on significant or total module counts, I was not able to improve the prioritisation enough to beat the unclustered method. Although a slight improvement in prioritisation can be gained in some cases compared to the method of taking the lowest p-value. Also keep in might that such tricks lack any statistical motivation and do not give you confidence values, making sensitivity and specificity testing impossible.

3.3 The BinoX package for R

To facilitate further research, an R package was made to run the BinoX tool, either standalone or in combination with MCL or MGclus. In addition, the package provides an inter-

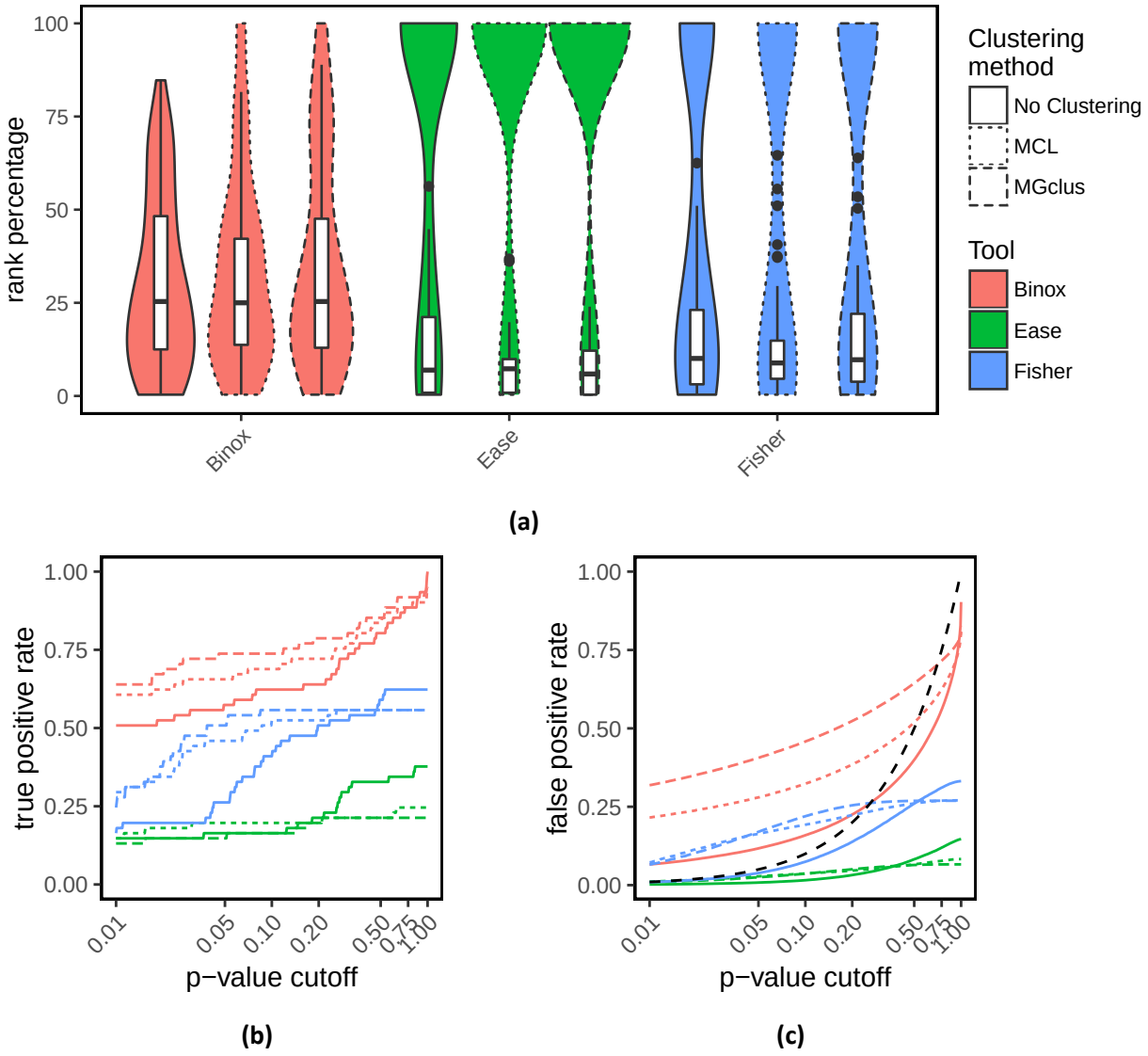


Figure 3.8: Clustering has little effect on pathway analysis applied to MSigDB gene sets. The BinoX, EASE and Fisher method were applied with and without clustering to the gold standard data based on MSigDB gene sets (table A.1). Boxplots and violin plots in (**a**) are as in fig. 3.2. True positive rates as well as false positive rates using gene set permutation are given in (**b**) and (**c**) respectively. Figures (**b**) and (**c**) are also given using q-values in fig. B.9.

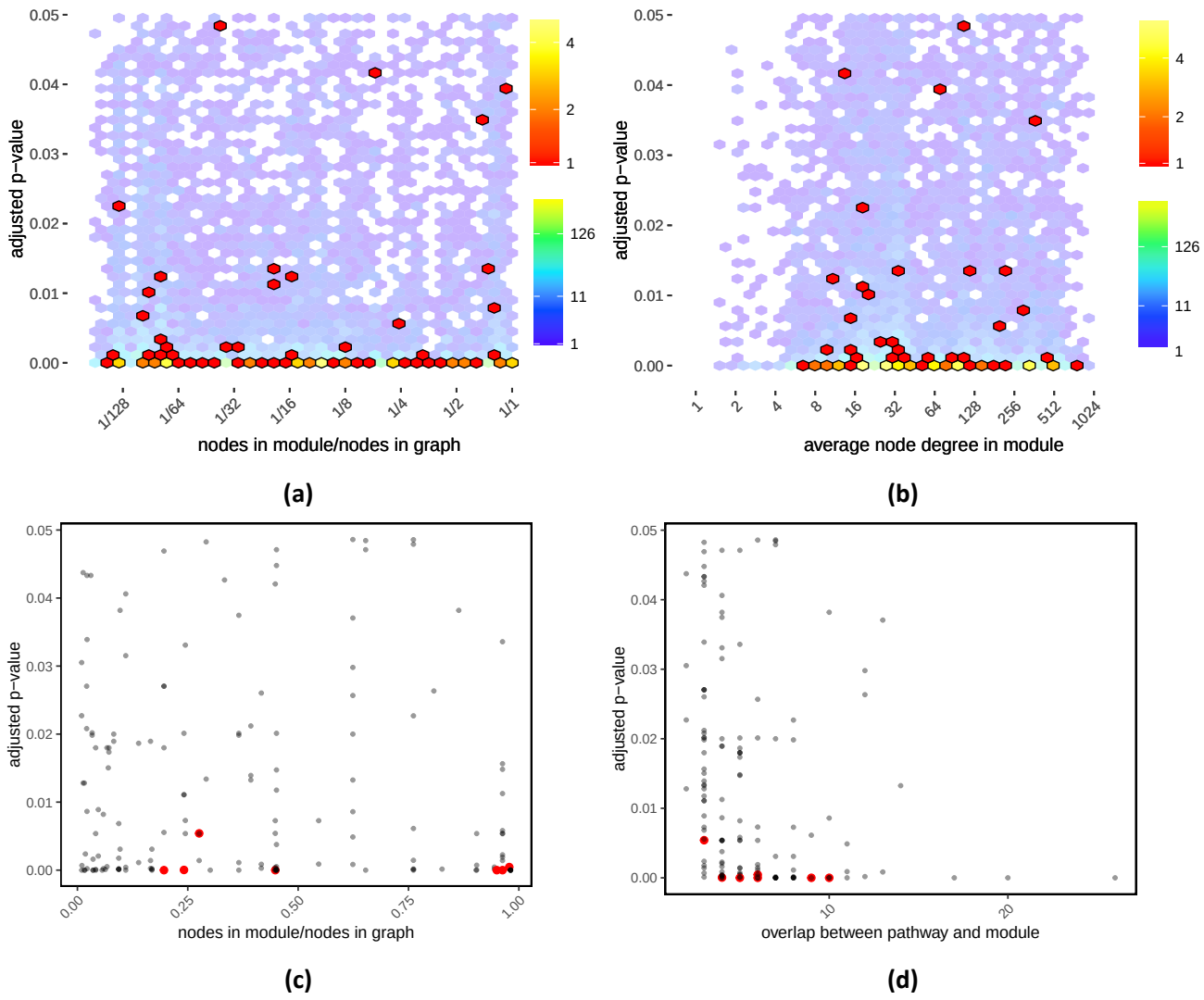


Figure 3.9: Distribution of true positives for MSigDB gene sets. A selection of gene sets from MSigDB (table A.1) were clustered using MCL and the modules were enriched to KEGG pathways. The BH adjusted p-values obtained from BinoX are shown in (**a**) and (**b**), colors and scales are as in fig. 3.6. The same modules were enriched using EASE, p-values are shown in (**c**) and (**d**). Module to target pathway combinations (true positives) are shown in red, all other p-values are shown in black. Additional results using MGclus instead are shown in fig. B.5.

face to easily plug-in any clustering method. The complete benchmark depicted in fig. 3.1 is also included in the package. The source code is hosted in a git repository and is available at `https://DMSam@bitbucket.org/DMSam/binox-r.git`. All public functions are fully documented in the reference manual for the package.

3.3.1 Using BinoX from within R

To be able to run BinoX from within `R`, the `BinoX` package and its dependencies must be installed (see first page of reference manual). Also, the BinoX command line tool must be installed and must be available on the systems `$PATH`. The BinoX command line tool can be downloaded from `http://sonnhammer.org/BinoX`.

BinoX works in two steps: first the FA network must be randomised, then gene sets can be enriched to each other. Since randomising the network can take quite some time, the advised workflow is to do this first and store the result on the disk. The randomised network can then be reused for every analysis. This can be done very easily from within `R`:

```
library(BinoX)
library(igraph)

# for reproducibility
set.seed(1234)

# make up a scale free network
fa_network <- sample_pa(1000, directed = F)

# randomize the network and store result to disk
random_network_path <- tempfile()
Binox_randomizeNetwork(randNetPath    = random_network_path,
                       network        = fa_network,
                       seed           = 1234, # Seed for BinoX network randomizer.
                       edgeWeightName = NA)   # Prevent using edge weights, all
                                              # edge weights are set to 1.0
```

If this returns `TRUE` then the randomisation is finished. The next step is to do the actual analysis:

```
# make up some gene sets and pathways
gene_sets <- rbind(
  data.frame(members = 1:200  , gene_set = "set1"),
  data.frame(members = 300:600, gene_set = "set2")
)

pathways <- rbind(
  data.frame(members = 50:210 , pathway = "pw1"),
  data.frame(members = 500:600, pathway = "pw2")
)

# enrich with BinoX
output <- Binox(groupsA = gene_sets, groupsB = pathways,
                randNetPath = random_network_path)
print(output)
```

If everything went well, this should return a `data.frame` with four rows; one row for every gene set versus pathway combination:

```
  NameGroupA NameGroupB ID    p.value       FDR relationType PFC module
1       set1        pw1  1 0.00890549 0.0178110            +   1     NA
2       set1        pw2  2 0.00705778 0.0282311            -   1     NA
3       set2        pw1  3 0.07663070 0.1021740            -   1     NA
4       set2        pw2  4 0.14139800 0.1413980            +   1     NA
```

To include clustering in the analysis, first make a closure[2] to prepare the clustering algorithm with the FA network, then pass the closure to the `BinoX` function:

```
# make a closure for Markov clustering with default settings, we are not using
# edge weights in this example
mcl_cluster_function <- prepare.MCL.ClusterFun(fa_network,
                                               edgeWeightName = NA)
# cluster a gene set with Markov clustering
modules <- mcl_cluster_function(1:20)
print(modules)

# run BinoX with clustering
output_clus <- Binox(groupsA = gene_sets, groupsB = pathways,
                     randNetPath = random_network_path,
                     clusterFun = mcl_cluster_function)
print(output_clus)
```

This will cluster every gene set and enrich every module to every pathway. The output is a `data.frame` with p-values and q-values for all module to pathway combinations. Note that any function that takes a vector of identifiers as input and returns a list of vectors with identifiers (one vector per module) as output will be accepted. So any clustering algorithm, such as those from the `igraph` package (`http://igraph.org/r`), can easily be plugged into the `BinoX` function as well. More details are available in the reference manual.

For making MCL or MGclus work, the `mcl` and `MGclus` programs must be available on the command line. They can be downloaded from `http://micans.org/mcl/index.html?sec_thesisetc` and `http://sonnhammer.sbc.su.se/download/software/MGclus` respectively.

3.3.2 Command Line Interface

In addition to the R interface, there is also a command tool included in the `BinoX` package. The command line tool allows the user the run BinoX in combination with either MCL or MGclus, provided that the `R` package and all the dependencies listed above are installed. Although the Command Line Interface (CLI) is quicker to use, it is far less flexible and exposes only a subset of the functionality provided with the `R` interface. The `--help` of the CLI is also provided in the section *BinoX-cli* of the reference manual.

[2] Closures are functions with associated data. When defining a function in R, it can "catch" objects from its parent environment and store them immutably as a part of the function.

3.3.3 Benchmarking pipeline

The complete benchmarking pipeline used in this thesis is also available in the `R` package. The `PWAbenchmark` class keeps track of all the settings such as which identifiers to use and how to select gene sets from microarray data. Every component of the benchmark can be swapped out easily by the user, and nearly all steps can be executed in parallel by simply setting the `ncores` argument to the number of processor cores you want to use. Moreover, the package will store intermediary results such as probeset to gene identifier mappings or differential expression statistics in a structured fashion on the hard drive. Whenever a setting is changed, the package will automatically determine which results have to be computed and which results can be reused from previous runs, and it will only compute what is necessary. Even when the computations have been interrupted by the user or due to an external failure, no progress will be lost and when the benchmark is resumed later, it will pick up from where it was interrupted. An introduction to the benchmark is included in the package repository (`https://DMSam@bitbucket.org/DMSam/binox-r.git`, in the file `demo/benchmark-demo.html`), and a complete reference is available in the package manual.

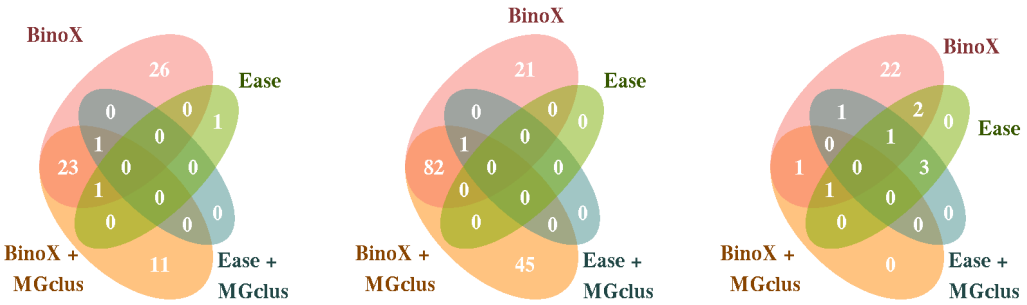
3.4 Human Proteome Atlas

One of the objectives (chapter 2) was to run BinoX and Ease on the tissue specific gene sets from the HPA with and without clustering (Uhlen *et al.*, 2015). Unfortunately, clustering turned out to have a rather negative effect on pathway analysis. Still, it might be interesting to see how BinoX and EASE behave on this data, and whether there are any important pathways that can only be found with clustering.

Tissue specific gene sets have been downloaded from the HPA website (`http://www.proteinatlas.org/humanproteome/tissue+specific`). There are three categories used by HPA to indicate how "tissue specific" a gene is to a certain tissue. These categories are "tissue enriched", "group enriched" and "tissue enhanced". Because some gene sets in either of these three categories can be extremely small (e.g. the "tissue enriched" gene set for smooth muscle comprises only one gene), the gene sets used here are the union of these categories. Pathway analysis was then carried out on all gene sets versus all KEGG pathways with BinoX and EASE with and without clustering with MGclus. Functional enrichment (with EASE) towards GO terms has been done before by the authors of the HPA project, and for three tissues there were none to a few significant GO terms, these tissues were *1)* Adipose tissue, *2)* Lung tissue and *3)* pancreas.

Results

When enriching the same tissues for KEGG pathways there are once again almost no pathways found using EASE. As was clear from the benchmark, BinoX is much more sensitive and finds more significantly enriched pathways. A brief overview of these three tissues is shown in figure fig. 3.10, and complete results are given in tables C.1 to C.3.



(a) Adipose tissue **(b)** Lung tissue **(c)** Pancreas

Figure 3.10: Pathway analysis for three tissues of the HPA. Pathway analysis was done on tissue specific gene sets provided by the HPA. Three example tissues are shown here: adipose tissue (**a**), lung tissue (**b**) and pancreas (**c**). The venn diagrams show how many KEGG pathways are found by each tool using a q-value cutoff of 0.05. The tools used are BinoX and EASE with and without clustering of the gene sets with MGclus. The full data for these three tissues is given in appendix C.

Adipose tissue BinoX, BinoX + MGclus and EASE found the PPAR signalling pathway, a key pathway for regulating fatty acid metabolism, which is an important function of adipocytes (Fujii, 2005). Another pathway found by BinoX, BinoX + MGclus and EASE + MGclus was Neuroactive ligand-receptor signalling. This is a very broad pathway containing many proteins involved in signal transduction, but it has no apparent relation to adipose tissue. There is one key pathway that is only found by EASE: the AMPK signaling pathway, which is well known to have an important role in adipose tissue energy metabolism (Steinberg and Kemp, 2009; Bijland *et al.*, 2013; Daval *et al.*, 2006).

These three pathways are the only ones found by EASE or EASE + MGclus, whereas there are 60 pathways that are only found by either BinoX or BinoX + MGclus. Among these are quite a few well known important pathways for adipose tissue such as the adipocytokine signaling pathway, fatty acid biosynthesis/degradation/metabolism and glycerolipid metabolism. Moreover, there are some key pathways that are only found by BinoX + MGclus that are completely missed by BinoX without clustering. An example is the JAK-STAT signaling pathway, which has an important role in adipose tissue development and regulation of adipose tissue metabolism (Richard and Stephens, 2014; Xu *et al.*, 2013; Moisan *et al.*, 2015).

Lung tissue One pathway that is found by BinoX, BinoX + MGclus and EASE is the focal adhesion pathway. This pathway has an important role in lung cancer (Lagares *et al.*, 2012) and might be involved in pulmonary fibrosis (Kinoshita *et al.*, 2013). This is the only pathway found by EASE. BinoX and BinoX + MGclus on the contrary together found 150 significant pathways. Given that there are 288 KEGG pathways in total, there are likely many false positives among those, some likely examples are drug metabolism, prostate cancer, pancreatic cancer and dilated cardiomyopathy, which are not of importance in normal lung tissue.

Pancreas Some known pancreatic pathways such as insulin secretion, fat digestion and absorption and Maturity onset diabetes of the young (MODY)—a genetic disorder that

causes type II diabetes—are found by both BinoX and EASE. Three important pathways are only found by EASE: carbohydrate digestion and absorption, protein digestion and absorption and pancreatic secretion. Especially the latter is a key pathway and is almost found by BinoX but missed completely by BinoX + MGclus (q-values are 0.071 and 0.146 respectively, see table C.3).

Conclusions

From the examples above, it is clear that using multiple tools is important to get a full picture. It is not because one method is more sensitive that other less sensitive methods should not be used. In two of three examples above, the less sensitive EASE tools could find important pathways that were not found by BinoX. Finally, clustering added some important pathways for the analysis of adipose tissue and the pancreas, but it also added many nonsense pathways for the lung tissue. Hence the results from clustering should be interpreted with extreme caution.

Part 4: Discussion

4.1 Assessing the performance of a new pathway analysis tool

Here I have benchmarked the new BinoX method versus earlier methods. On a qualitative level, the results are in agreement with earlier publications dealing with benchmarking pathway analysis tools (Tarca *et al.*, 2012, 2013; Dong *et al.*, 2016). Thus, despite changing a few major aspects of the benchmark compared to earlier work (section 3.1.3), it might still be possible to compare previous findings with the results presented here. If so, then BinoX, even though it does not use microarray data, performs quite well in comparison to other pathway analysis methods. When analysing small gene sets ($< \pm 500$ genes), there is often no overlap. Especially in this scenario BinoX has a clear advantage. This has been illustrated not only by the benchmark, but also when analysing gene sets from the HPA, where BinoX was able to find many relevant pathways that were not found by the overlap based EASE method. Although, when dealing with large gene sets, the FPR of BinoX can be quite high and a very strict significance cutoff combined with multiple testing is advised.

Therefore if the goal is to get as much insight as possible from a gene set, BinoX should be used, especially for small gene sets. But if the goal is to prioritise pathways for follow up experiments, then overlap based methods might still be preferable. Even if microarray data is available, BinoX is still advised to gain insight in the underlying biology. But for prioritising pathways for follow up experiments, FCS based methods should be used instead.

In conclusion: there is no silver bullet, every tool has advantages and disadvantages. The best advice is to always use as many tools as possible and then compare the output while keeping the strengths and weaknesses of the tools in mind. This method is slow and requires an understanding of how each of the tools work, but is guaranteed to bring the most insight. Ensemble[1] methods that pool the output from many tools into one analysis might be very useful as a quick way for prioritising pathways. But might not bring as much insight in the underlying mechanics as comparing the outputs manually.

[1] Not to be confused with the *Ensembl* database (`http://www.ensembl.org/`).

4.2 Remaining problems for benchmarking pathway analysis

Benchmarking pathway analysis tools has always been a difficult topic. The method used here for testing sensitivity is solid only if the gold standard data is of good quality and forms a representative sample. The second condition might still be a problem since the sample only deals with disease phenotypes and is heavily biased towards cancer pathways. In addition the microarray standard is quite small and the analysis involves many decisions such as which test statistic to use and how to compile DE gene sets. It is therefore susceptible to the phenomenon known as *researcher degrees of freedom*: the ability of a researcher to, with or without bad intentions, obtain any desired outcome by changing parameters of the analysis. To combat this, a second gold standard based on MSigDB data is used here which, in many aspects, corroborates the results obtained from the first benchmark. Unfortunately, the second benchmark is also biased towards cancer pathways, it is therefore still not completely certain if what is found here can be extrapolated to pathway analysis in general. Moreover, the prioritisation benchmark is not perfect either, since there is always more than one "target pathway" that should rank close to the top, so the target pathway used here does not necessarily have to be in the first place. Thus, the MSigDB gold standard helps to solve the benchmarking problem but is not the final solution.

Testing the false positive rate is also still problematic. There is a growing consensus forming in literature that correlations within gene sets should be accounted for in the null hypothesis, or as Gatti *et al.* (2010) puts it: "correlation within a gene set is largely a persistent property that is preserved across a wide variety of sample sources and experimental conditions". Thus, correlated genes in a gene set should not be seen as independent evidence, because this phenomenon is independent of the condition under study. Any good benchmark should thus not rely its specificity test on gene sets were this correlation structure is not present. A proposed solution is permuting the sample labels, but this poses a problem for FCS tools: benchmarking FCS tools with data generated by the same null hypothesis as used by those tools gives them an unfair advantage. Another problem is that by using sample label swaps the true signal might still be preserved to some degree. Yet another problem is that this method does not work for set based data. The alternative specificity test (using gene set permutations) solves the these problems but does not include the correlation structure of real gene sets. When comparing both tests, then the specificity of BinoX is much worse on the first, which is to be expected as BinoX does not account for correlations within gene sets in its null model. EASE and Fisher perform also worse on the first specificity test, but the results are not as extreme due to their inherent poor sensitivity.

In conclusion, although this benchmark is an improvement upon previous benchmarks, it is still uncertain whether results can be extrapolated to other conditions than diseases or even to other organisms. And the nature of specificity testing of pathway analysis is still problematic.

What is pathway analysis?

The problems presented above all tie to one underlying question: what do we mean with pathway analysis? An obvious answer would be that the aim of pathway analysis is to find pathways that have an important role in the what is being studied. But this poses a new question: what does “an important role” mean? What do we actually mean if we say that a pathway is important to a disease/phenotype/etc.? There is no strict definition of what an “important pathway”—or pathway analysis in general—means. Different tools model the answer to this question in different ways, e.g. in terms of overlap, link counts, ranking of pathway members in gene lists and more. But how do we know which of these answers is the closest to the truth if the truth is still undefined?

In the benchmark used here, it is unambiguous that if a sample is taken from disease tissue, than the disease pathway is important. But for many other pathways ambiguity arises. Especially when we ask ourselves the opposite question: which pathways are not relevant? For example, what if an energy metabolism pathway remains completely unaffected in cancer cells compared to healthy cells of the same tissue? Is this pathway then *important*? One could argue that it is not since, as it is functioning the same as in healthy cells, it could not cause the cancer phenotype. But one could also argue that it is very important since, despite the many mutations of cancer cells, this pathway is still functioning in exactly the same way. Thus, keeping this pathway going in the same state is essential for cancer cell survival. This explains why, while it is hard to test the sensitivity, it is even harder to test the specificity. There is no way to know whether a pathway is truly irrelevant. Of course, we can generate completely random pathways or gene sets and be certain that they are not relevant. But how do reliably can we extrapolate from simulation studies to real data?

In the end, creating *the* best pathway analysis method—or benchmark—is an impossible task, since every method or benchmark uses a different definition of what pathway analysis actually is. Therefore the best method can only be chosen once your have answered for yourself the question of what exactly you are looking for with pathway analysis.

Future perspectives

A few things can be done to further improve the benchmark. For the sensitivity test, gold standard datasets from different conditions and different organisms are required. It would also be nice to know how results differ when using another annotation database such as Reactome or GO. Once there is enough testing data to form a representative sample, the sensitivity testing problem is more or less solved. Although compiling this data in an objective manner would be difficult and time consuming, it is certainly not impossible. A possibility to improve the specificity test would be to keep using real data but randomize the pathways. One could enrich real data to a fake annotation database containing only the target pathway and a random pathway. Then sensitivity and specificity testing could be done in one go and it would be possible to generate ROC curves and use the Area Under Curve (AUC) as the ultimate measure to compare methods.

4.3 Should clustering be used?

Here, clustering gene sets prior to pathway analysis has been attempted with two different clustering methods on three tools. The hypothesis that clustering would purify pathways from the gene sets into the same modules as illustrated in fig. 2.1 appears to be false. Instead, pathways are spread across a number of small modules and "experimental noise" is also clustered together in modules with similar properties, making it more difficult to differentiate the signal from the noise. The only property that gives a slight indication in the right direction is the number of modules that are enriched to a given pathway. If this number is high compared to other pathways, then the given pathway is likely to be more important. But combining modules like this is a self defeating exercise because then it would be better to just use the whole gene set instead. Another problem is that clustering inflates the p-values so much that they cannot longer be trusted, making the relative ranking of pathways the only valuable output. Fisher and EASE appear to rank pathways in a more or less similar order after clustering, regardless of what clustering algorithm is used (fig. 4.1) but for BinoX the ranking is affected to a larger extend. Finally, PADOG is quite orthogonal to any other method tested here. This highlights again the need to always use different tools for analysing your data.

In conclusion, clustering seems to be a noise inducing step rather than a noise eliminating step and should not be used in combination with pathway analysis. Instead it is better to use conceptually different approaches which answer the question of which pathways are relevant to the study in different ways.

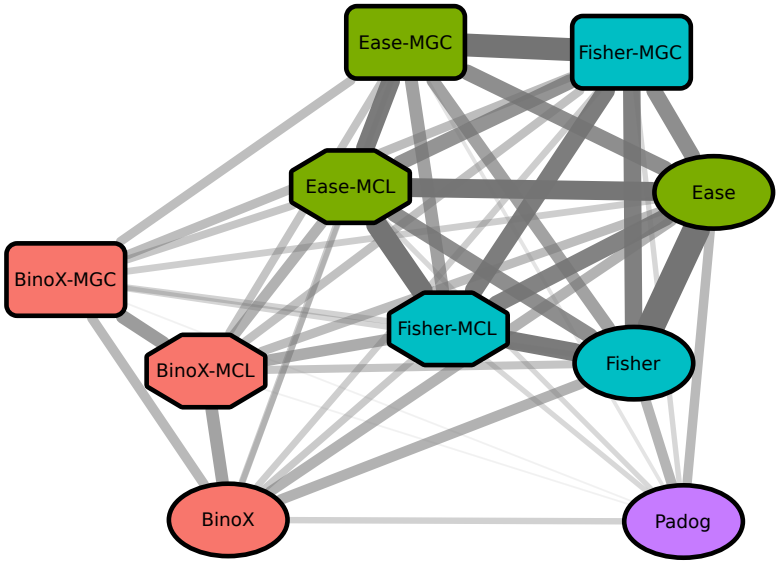


Figure 4.1: Similarity of tools in terms of ranking. For each dataset in table 3.1 the Kendall's tau coefficient was computed on the p-values of the KEGG pathways for each pair of tools. Line thickness corresponds to the average Kendall's tau across the 26 datasets. The smallest rank correlation is between BinoX + clustering and PADOG (≈ 0.20), the largest is between Fisher and EASE (≈ 0.85). MGC is short for MGclus.

4.4 Korte discussie in het Nederlands

Onderzoek naar de kwaliteit van een nieuwe *pathway* analyse methode

De nieuwe BinoX methode voor *pathway* analyse werd hier onderzocht en vergeleken met eerder gepubliceerde tools. Kwalitatief gezien zijn de resultaten vergelijkbaar met vroegere publicaties (Tarca *et al.*, 2012, 2013; Dong *et al.*, 2016). Dus, ondanks enkele grote verschillen in de analyse is het misschien toch nog mogelijk om eerdere resultaten the vergelijken met wat hier is waargenomen. Als dit het geval is, dan is BinoX een redelijk goede tool gegeven het feit dat het geen gebruik maakt van *microarray* data. BinoX lijkt voornamelijk goed te werken op kleine gen sets waar een overlap vaak niet wordt waargenomen. Voor grotere gen sets wordt de FPR van BinoX erg hoog, en gebruik van een hoge significantie *cutoff* in combinatie met *multiple testing correction* is aanbevolen.

Dus, om een dieper inzicht te krijgen in de onderliggende biologie is BinoX een aanbevolen methode, maar om *pathways* the prioriteren voor verder onderzoek kan een overlap gebaseerde methode nog steeds nuttig zijn. Zelfs als *microarray* data beschikbaar is is het nuttig om BinoX te gebruiken om meer inzicht te krijgen in de data. Voor prioritering is een FCS methode echter meer geschikt.

Ten slotte kan men stellen dat er geen *silver bullet* methode is voor *pathway* analyse. Elke methode heeft voor en nadelen en verschillende methoden gebruiken op dezelfde data is altijd aangeraden.

Onopgeloste problemen voor het evalueren van *pathway* analyse methoden

Het is altijd moeilijk geweest om *pathway* analyse methoden objectief te evalueren. De methode die hier gebruikt is is enkel geldig als de data van goede kwaliteit is en een representatief staal vormt. Aangezien de “gouden standaard” data voornamelijk kanker weefsels bevat kan de tweede voorwaarde een probleem zijn. De vele beslissingen in de data analyse en de beperkte grootte van het staal kunnen ook leiden tot het probleem van *researcher degrees of freedom*: de mogelijkheid van een onderzoeker om, met of zonder slechte intentie, elk gewenst resultaat te observeren door parameters in de analyse aan te passen. Dit probleem kan effectief aangepakt worden door een tweede “gouden standaard” te introduceren, zoals bijvoorbeeld de MSigDB data die hier is gebruikt. Zowel de MSigDB data als de *microarray* data lijken dezelfde conclusie te ondersteunen. Helaas bevat ook de MSigDB data veel kanker gerelateerde gen sets, waardoor het misschien ook een niet representatief staal is voor *pathway* analyse in het algemeen. Dus, ondanks dat de MSigDB data helpt, is het zeker niet de finale oplossing.

Ook het testen van de specificiteit is nog steeds een onopgelost probleem. Dit heeft vooral te maken met het feit dat er geen vaste definitie is voor wat *pathway* analyse werkelijk is. Het nog moeilijker om te zeggen welke *pathways* niet relevant zijn voor een bepaalde studie/ziekte of fenotype. Dit maakt het dan weer moeilijk om een goede nul hypothese te formuleren.

We kunnen concluderen dat deze *benchmark* een verbetering is ten opzichte van vorige methoden, maar het is nog niet zeker of we de resultaten gebaseerd op ziekte gerelateerde weefsels kunnen doortrekken naar een bredere context.

Kunnen we gebruik maken van clusteren?

Het effect van clusteren voor het toepassen van *pathway* analyse methoden is hier onderzocht met twee verschillende clustering methoden en drie verschillende tools. De hypothese dat clusteren een *pathway* zou uitfilteren in een aparte module lijkt niet te kloppen. In plaats daarvan wordt het signaal verspreid over een groot aantal kleine modules. Daarbovenop wordt experimentele ruis ook samen gebracht in kleine modules met gelijkaardige eigenschappen. Een tweede probleem is dat clustering de p-waarden zodanig versterkt dat ze niet langer kunnen gebruikt worden als p-waarden. Het enige wat een beetje overeind blijft is de prioritering van de *pathways*, vooral voor Fisher en EASE (fig. 2.1). Wat duidelijk is van fig. 2.1 is dat PADOG, BinoX en EASE/Fisher elk andere *pathways* prioriteren, wat nogmaals het belang van het gebruik van verschillende methodes bevestigd.

Part 5: Materials and Methods

5.1 Moderated t-test

All microarray studies in the benchmark (section 3.1) compare samples associated with a certain disease to healthy samples, they fall under two categories in terms of experimental design:

1. Either a disease tissue samples versus b healthy tissue samples, not paired, or
2. a disease tissue samples versus a paired healthy tissue samples, each pair coming from one patient.

From hereon, I will refer to the first and second design as the "unpaired" and "paired" design respectively.

The moderated t-test was used to assess which genes are DE between the disease and healthy phenotypes (Smyth, 2004). Here, the moderated t-values are derived from the coefficients of a linear model that is fitted for each gene.

$$\boldsymbol{y}_g = \boldsymbol{X}\alpha + \epsilon \tag{5.1}$$

Where the column vector $\boldsymbol{y}_g$ contains the normalized and summarized expression values of the microarray probes of one gene for all samples. $\boldsymbol{X}$ is the design matrix of the linear model and α is a column vector containing the coefficients to be estimated.

Model specification For the unpaired designs, consider a response vector $\boldsymbol{y}_g$ of length $a + b$ where the first a entries are expression values from diseased tissues, and the last b entries are expression values from healthy tissues. Then, the following design matrix was used:

$$\boldsymbol{X} = \begin{bmatrix} 1 & 0 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix} \tag{5.2}$$

Where the first a rows are $[1, 0]$ and the last b rows are $[0, 1]$. By minimizing the residual sum of squares for this model, the estimate of the first coefficient $\hat{\alpha}_{g1}$ will simply be the average of the disease expression values, and the estimate of the second coefficient $\hat{\alpha}_{g2}$ will be the average of the healthy expression values.

For the paired designs, consider a response vector $\boldsymbol{y}_g$ of length $2a$, where every entry from $y_{gi}, i \in [1, \ldots, a]$ is paired with an entry $y_{g(i+a)}$. Then, the following design matrix was used:

$$\boldsymbol{X} = \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots \\ 1 & 0 & 1 & 0 & \ldots \\ 1 & 0 & 0 & 1 & \ldots \\ \vdots & \vdots & \vdots & \vdots & \ddots \\ 0 & 1 & 0 & 0 & \ldots \\ 0 & 1 & 1 & 0 & \ldots \\ 0 & 1 & 0 & 1 & \ldots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{5.3}$$

Where the first a rows correspond to disease samples, and the last a rows correspond to healthy samples. In this model, the first coefficient α_{g1} represents the expression value for gene g in diseased tissue of patient 1 (an arbitrarily chosen patient). Similarly, the second coefficient α_{g2} represents the expression value of gene g in healthy tissue of patient 1. All other coefficients $\alpha_{g(i+1)}, i \in [2, \ldots, a]$, represent the difference in expression of gene g for patient i versus patient 1.

Contrast specification After fitting the model, the following contrast matrix, a vector in this case,[1] is specified:

$$\begin{matrix} \text{unpaired} \\ \text{designs} \end{matrix} : \quad \boldsymbol{c} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \qquad \begin{matrix} \text{paired} \\ \text{designs} \end{matrix} : \quad \boldsymbol{c} = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \tag{5.4}$$

The contrast of interest[2], β_g, which is the difference in expression of a gene between the disease and healthy phenotype, is then obtained by:

$$\beta_g = \boldsymbol{c}^\top \alpha_g \tag{5.5}$$

For unpaired designs, $\hat{\beta}_g$ is simply the difference of the mean for the disease and healthy samples. For paired designs, $\hat{\beta}_g = \hat{\alpha}_{g1} - \hat{\alpha}_{g2}$, which is the estimated difference in expression excluding patient effects.

Moderated t-value The ordinary t-value for the contrast β_g would be obtained by:

$$t_g = \frac{\hat{\beta}_g}{s_g \sqrt{v_g}} \tag{5.6}$$

[1] Using the notation of Smyth (2004), the contrast matrix would be written as $\boldsymbol{C}$, but since there is only one contrast of interest here, $\boldsymbol{C}$ reduces to a vector, which I will denote as $\boldsymbol{c}$.

[2] In Smyth (2004), the contrasts of interest are denoted as the vector β_g, and an individual contrast j as β_{gj}. Since there is only one contrast of interest here, I am dropping the subscript j.

Where s_g is the estimated standard deviation of $\boldsymbol{y}_g$ and v_g is the unscaled variance of β_g. I.e.: $v_g = \boldsymbol{c}^\top \boldsymbol{V}_g \boldsymbol{c}$ where $\boldsymbol{V}_g$ is the unscaled covariance matrix of α. This means that for an ordinary t-statistic, the variance of each gene is estimated independently. Smyth (2004) proposed a moderated t-statistic where the variance of a gene is weighted by the ordinary variance and a prior variance derived from all genes:

$$\tilde{s}_g^2 = \frac{d_0 s_0^2 + d_g s_g^2}{d_0 + d_g} \tag{5.7}$$

Where s_0^2 and d_0 are the prior variance and degrees of freedom respectively, which are estimated from the data (see Smyth (2004) for derivation of these values). Using $\tilde{s}_g$ instead of s_g in eq. 5.6 and expanding $v_g = \boldsymbol{c}^\top \boldsymbol{V}_g \boldsymbol{c} = \boldsymbol{c}^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{c}$ yields, in this case, the following equation for the moderated t-value:

$$\tilde{t}_g = \frac{\hat{\alpha}_{g1} - \hat{\alpha}_{g2}}{\tilde{s}_g \sqrt{\frac{1}{a} + \frac{1}{b}}} \tag{5.8}$$

Where, for paired designs, $b = a$ and, as mentioned earlier, for unpaired designs, $\hat{\alpha}_{g1}$ and $\hat{\alpha}_{g2}$ are simply the sample means. Thus, the variance of the contrast will decrease proportional to the sample sizes and to how well balanced the sample sizes are ($a/b \approx 1$).

The variance of $\boldsymbol{y}_g$ is easily estimated too small by accident in microarray studies with low sample sizes, inflating the corresponding p-value. The opposite, overestimation of the variance, can also easily happen for the same reason. Using the moderated t-test, sample variances are shrunken towards a common value, reducing the effect of variance over- or underestimation. Therefore the moderated t-test is advantageous over the ordinary t-value.

Final notes Using the procedure described above, a moderated t-value is obtained for each gene. Positive t-values indicate that a gene has higher expression in diseased tissues, while negative t-values indicate the opposite.

P-values obtained from these t-statistics were adjusted for multiple testing using the BH procedure (Benjamini and Hochberg, 1995).

All steps described in this section were carried out in `R` (R Development Core Team, 2016) using the package `limma` (Ritchie *et al.*, 2015). See the Implementation of the `makeModeratedTTestTopTable` function in the `BinoX` package for the code.

5.2 Translating gene identifiers

Two methods have been used to map probeset identifiers to Ensembl IDs. The first method, used in section 5.2.1 was used for the benchmark based on the microarray data. The second, used in section 5.2.2 was used for the MSigDB based benchmark.

5.2.1 Mapping IDs to Ensembl through the FunCoup network

All microarray datasets in the benchmark are either based on the Affymetrix `hgu133a` chip or the `hgu133plus2` chip. Both chips have an annotation package in Bioconductor named `hgu133a.db` and `hgu133plus2.db` respectively. For both packages version 3.2.2 was used. These packages provide direct probeset to Ensembl translations, but there are often many to many relationships between probeset and gene identifier. This problem was dealt with in two steps: First, from all Ensembl IDs, the one with highest node degree in FunCoup was taken. Second, if multiple probesets—each one having an associated test statistic for differential expression–map to the same Ensembl ID, the one with lowest p-value was taken. In pseudocode:

Require: probeset IDs mapped to a test statistic (and p-value) for differential expression
Ensure: Ensembl IDs mapped to a test statistic (and p-value) for differential expression

```
{Step 1}
mapping ← empty dictionary
for all probeset ID from platform do
  matching Ensembl IDs ← all Ensembl IDs from annotation package matching probeset ID
  if there are matching Ensembl IDs then
    if any matching Ensembl IDs ∈ FunCoup network then
      mapping(probeset ID) ← highest degree node from matching Ensembl IDs
    else
      mapping(probeset ID) ← first ID from matching Ensembl IDs
    end if
  end if
end for

{Step 2}
EnsemblID_to_testStatistic ← empty dictionary
for all Ensembl ID in values of mapping do
  matching probeset IDs ← all keys (probeset IDs) that map to the current Ensembl ID in mapping
  testStat ← most extreme test statistic for all probesets in matching probeset IDs
  pValue ← p-value associated with TestStat
  EnsemblID_to_testStatistic(Ensembl ID) ← (testStat; pValue)
end for
```

In the end, a non-redundant list of Ensembl IDs is obtained, each one having an associated test statistic and p-value for differential expression. The translated list will be shorter than the number of probeset IDs since some probeset IDs cannot be translated to Ensembl IDs and because many to many mappings are resolved.

For translating KEGG IDs to Ensembl, the same procedure as “Step 1” was used, except that Entrez IDs were translated instead of probeset IDs. For mapping Entrez to Ensembl, the `org.Hs.eg.db` package version 3.2.3 from Bioconductor was used. Duplicate Ensembl IDs (arising from multiple Ensembl to one Entrez ID mappings) were removed to obtain non redundant gene sets.

5.2.2 Entrez to Ensembl mapping from earlier publication

The CrossTalkZ pathway analysis tool (McCormack *et al.*, 2013), also requiring Ensembl IDs, has been tested earlier with simulated data as well as with KEGG and MSigDB gene sets. For the benchmark based on MSigDB data in this thesis (section 3.2.3), I have reused these already translated gene sets.

5.3 False positive rate estimation

5.3.1 Sample label swap

For testing the FPR, microarray data was taken but the sample labels were permuted n times. This way, every sample is randomly assigned a disease or healthy label. Because the groups are now random, differential expression should not be observed after correcting for multiple testing. This is true in all but a few cases were the permuted sample labels closely match the original sample labels. Using this sample label swap, genes that are correlated are more likely to be selected together. Therefore, this method can be useful for testing whether tools account for gene-gene correlations.

5.3.2 Gene set permutation

For the alternative FPR test, the query gene sets were permuted by replacing every gene by a new one with similar node degree (maximum 5% difference). Replacement genes were not allowed to be in the query gene set set and may not have been picked before. When there are no such genes, a completely random gene from FunCoup is picked as replacement.

Require: input gene set; FA network
Ensure: permuted gene set with approximately the same node degree distribution as input gene set

permuted gene set ← empty set
for all *gene* ∈ *input gene set* **do**
  candidates ← genes differing no more than 5% in node degree from *gene* given FA network
  candidates ← all *candidates* ∉ *input gene set*
  candidates ← all *candidates* ∉ *permuted gene set*
  if there are candidates **then**
    add random gene from *candidates* to *permuted gene set*
  else
    unpicked ← genes from FA network ∉ *input gene set*
    unpicked ← all *unpicked* ∉ *permuted gene set*
    add random gene from *unpicked* to *permuted gene set*
  end if
end for

5.4 Procedures used for benchmarking

5.4.1 Benchmark based on microarray data

Sensitivity and prioritisation test

1. The 26 datasets from table 3.1 were downloaded from NCBI GEO.
2. The moderated t-test was done as described in section 5.1 using the samples given in appendix A.1.
3. Probeset identifiers were translated to Ensembl as described in section 5.2.1.

   PADOG was run by passing the following input to the `PADOG` function from the `PADOG` Bioconductor package `http://www.bioconductor.org/packages/release/bioc/html/PADOG.html`.:

   (a) The probeset to Ensembl translations from step 3

   (b) The microarray data from step 1

   (c) The KEGG pathways translated to Ensembl as described in section 5.2.1

   (d) Number of iterations = 50
4. For each dataset, all genes with a BH adjusted p-value of at least 0.01 and a fold change of at least 50% were used to form a gene set. This means both up and downregulated genes are included in the gene set.

   BinoX was run on these gene sets with `relationType = +`, i.e. only enrichment p-values were calculated. The FunCoup network version 3.0 was used with and edge weight cutoff of 0.8 and 150 randomisations. KEGG pathways were the same as for PADOG.

   EASE & Fisher methods were run on the same gene sets and KEGG pathways as BinoX.
5. Gene sets were clustered by:

   (a) Extracting a subnetwork from the complete FunCoup graph (no cutoff) containing the genes from the gene set.

   (b) Removing genes that have no edges in the subnetwork.

   (c) Applying either MGclus or MCL to the obtained subnetwork. Both tools were run with default settings: merge gain cutoff of 0 for MGclus and inflation of 2 for MCL. Edge weights were used for both algorithms.

   (d) Removing all genes that end up in a one-node module.

   BinoX was run on each module of each gene set versus all KEGG pathways. The minimum allowed gene set size was lowered to 2. P-value correction was done across all module to pathway combinations tested. Per pathway, the lowest p-value across all modules was kept.

EASE & Fisher were also run on all modules versus all KEGG pathways, and also corrected for all combinations. Per pathway, the lowest p-value across all modules was kept.

6. Sensitivity and prioritisation was done on the output from PADOG in step 3 and the output from BinoX, EASE and Fisher from step 4 and 5.

   sensitivity This is the number of target pathways that have a p-value below the significance cutoff. The sensitivity over a range of cutoff values is given in figs. 3.2a and B.1b.

   prioritisation The pathways are sorted on p-value per gene set, then their position in the sorted list is determined. This is divided by the total number of pathways and multiplied by 100 to get the rank percentage. Sometimes a p-value cannot be determined or is equal to exactly 1. In this case, the rank percentage was set to 100% for the violin plots in figs. 3.3a and B.1a. For the boxplots these rank percentages were omitted.

Specificity test

Two tests have been used. For the first, the sample labels from the microarrays were permuted (see section 5.3.1). Then step 2 and step 3 of the sensitivity benchmark is repeated on this false data. Instead of step 4, the first n genes are picked (sorted on p-value for DE) where n is a number drawn randomly from the gene set sizes of the true gene sets obtained in step 4. Then, step 5 is again the same as in the sensitivity test. The specificity is simply the proportion of p-values that fall below a certain cutoff. The specificity over a range of cutoffs is shown in figs. 3.3c and B.1c. For the second: the true gene sets from step 4 of the sensitivity test were permuted as explained in section 5.3.2. These gene sets where then clustered as described in step 5 of the sensitivity test. Specificity is again the proportion of p-values below a certain cutoff (figs. 3.3d and B.1d).

Both of the specificity tests have been repeated 10 times on all 26 datasets/gene sets.

5.4.2 Benchmark based on MSigDB gene sets

Sensitivity and prioritisation test

KEGG and MSigDB gene sets were already made available in Ensembl IDs by McCormack *et al.* (2013). The MSigDB gene sets from table A.1 were enriched versus all KEGG pathways. They were also clustered as described in step 5 of section 5.4.1 and then every module was enriched to every KEGG pathway. Sensitivity and prioritisation were calculated as described in step 6 of section 5.4.1.

Specificity test

Since no microarray data is available, sample label swaps have not been used for specificity testing. Specificity testing based on gene set permutations was done in exactly the same

way as described in section 5.4.1: by permuting the MSigDB gene sets while retaining first order node degree. Once again, every gene set was permuted 10 times.

5.5 Hardware and software used

hardware

Two platforms have been used for both building the `BinoX` package and running the benchmarks:

OS	kernel release	cores	max speed (MHz)	memory (kb)
Arch linux	4.5.1-1-ARCH	4	3000.000	8046156
Ubuntu 14.04.4 LTS	3.13.0-79-generic	8	800.000	33013632

Table 5.1: Computer platforms used. Arch linux has a rolling release model, no version numbers are used.

Software for data analysis

The `R` version and packages used are listed below:

```
R version 3.3.0 (2016-05-03)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 14.04.4 LTS

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=sv_SE.UTF-8        LC_COLLATE=en_US.UTF-8
 [5] LC_MONETARY=sv_SE.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=sv_SE.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=sv_SE.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] BinoX_0.0.1    nvimcom_0.9-14

loaded via a namespace (and not attached):
 [1] igraph_1.0.1         Rcpp_0.12.3          AnnotationDbi_1.32.3
 [4] magrittr_1.5         roxygen2_5.0.1       BiocGenerics_0.16.1
 [7] devtools_1.10.0      IRanges_2.4.8        munsell_0.4.3
[10] doParallel_1.0.10    colorspace_1.2-6     foreach_1.4.3
[13] plyr_1.8.3           stringr_1.0.0        tools_3.3.0
[16] parallel_3.3.0       grid_3.3.0           Biobase_2.30.0
[19] gtable_0.2.0         DBI_0.3.1            withr_1.0.1
[22] iterators_1.0.8      digest_0.6.9         readr_0.2.2
```

```
[25] ggplot2_2.1.0        S4Vectors_0.8.11     codetools_0.2-14
[28] memoise_1.0.0        RSQLite_1.0.0        limma_3.26.8
[31] stringi_1.0-1        scales_0.4.0         stats4_3.3.0
```

Software used for making this document

This document was typeset by the author using LuaTEX (http://www.luatex.org/), images were created with: inkscape (https://inkscape.org/en/), TikZ (https://www.ctan.org/pkg/pgf), ggplot2 (http://ggplot2.org/), VennDiagram (https://cran.r-project.org/web/packages/VennDiagram/index.html) and Cytoscape (http://www.cytoscape.org/).

References

Ackermann M, Strimmer K (2009) A general modular framework for gene set enrichment analysis. *BMC bioinformatics* 10(1):47.

Affer M, Dao S, Liu C, Olshen AB, Mo Q, Viale A, Lambek CL, Marr TG, Clarkson BD (2011) Gene Expression Differences between Enriched Normal and Chronic Myelogenous Leukemia Quiescent Stem/Progenitor Cells and Correlations with Biological Abnormalities. *Journal of oncology* 2011:798592.

Alexeyenko A, Lee W, Pernemalm M, Guegan J, Dessen P, Lazar V, Lehtiö J, Pawitan Y (2012) Network enrichment analysis: extension of gene-set enrichment analysis to gene networks. *BMC bioinformatics* 13(1):226.

Alexeyenko A, Sonnhammer ELL (2009) Global networks of functional coupling in eukaryotes from comprehensive data integration. *Genome research* 19(6):1107–16.

Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature genetics* 25(1):25–9.

Badea L, Herlea V, Dima SO, Dumitrascu T, Popescu I (2008) Combined gene expression analysis of whole-tissue and microdissected pancreatic ductal adenocarcinoma identifies genes specifically overexpressed in tumor epithelia. *Hepato-gastroenterology* 55(88):2016–27.

Bader GD, Hogue CWV (2003) An automated method for finding molecular complexes in large protein interaction networks. *BMC bioinformatics* 4:2.

Barrett T, Wilhite SE, Ledoux P, Evangelista C, Kim IF, Tomashevsky M, Marshall KA, Phillippy KH, Sherman PM, Holko M, Yefanov A, Lee H, Zhang N, Robertson CL, Serova N, Davis S, Soboleva A (2013) NCBI GEO: archive for functional genomics data sets–update. *Nucleic acids research* 41(Database issue):D991–5.

Barth AS, Kuner R, Buness A, Ruschhaupt M, Merk S, Zwermann L, Kääb S, Kreuzer E, Steinbeck G, Mansmann U, Poustka A, Nabauer M, Sültmann H (2006) Identification of a common gene expression signature in dilated cardiomyopathy across independent microarray studies. *Journal of the American College of Cardiology* 48(8):1610–7.

Benjamini Y, Hochberg Y (1995) Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing.

Bijland S, Mancini SJ, Salt IP (2013) Role of AMP-activated protein kinase in adipose tissue metabolism and inflammation. *Clinical science (London, England : 1979)* 124(8):491–507.

Clauset A, Newman MEJ, Moore C (2004) Finding community structure in very large networks *Physical Review E* 70(6):066111.

Croft D, Mundo AF, Haw R, Milacic M, Weiser J, Wu G, Caudy M, Garapati P, Gillespie M, Kamdar MR, Jassal B, Jupe S, Matthews L, May B, Palatnik S, Rothfels K, Shamovsky V, Song H, Williams M, Birney E, Hermjakob H, Stein L, D'Eustachio P (2014) The Reactome pathway knowledgebase. *Nucleic acids research* 42(Database issue):D472–7.

Cunningham F, Amode MR, Barrell D, Beal K, Billis K, Brent S, Carvalho-Silva D, Clapham P, Coates G, Fitzgerald S, Gil L, Girón CG, Gordon L, Hourlier T, Hunt SE, Janacek SH, Johnson N, Juettemann T, Kähäri AK, Keenan S, Martin FJ, Maurel T, McLaren W, Murphy DN, Nag R, Overduin B, Parker A, Patricio M, Perry E, Pignatelli M, Riat HS, Sheppard D, Taylor K, Thormann A, Vullo A, Wilder SP, Zadissa A, Aken BL, Birney E, Harrow J, Kinsella R, Muffato M, Ruffier M, Searle SMJ, Spudich G, Trevanion SJ, Yates A, Zerbino DR, Flicek P (2014) Ensembl 2015. *Nucleic acids research* 43(D1):D662–669.

Daval M, Foufelle F, Ferré P (2006) Functions of AMP-activated protein kinase in adipose tissue. *The Journal of physiology* 574(Pt 1):55–62.

Donahue TR, Tran LM, Hill R, Li Y, Kovochich A, Calvopina JH, Patel SG, Wu N, Hindoyan A, Farrell JJ, Li X, Dawson DW, Wu H (2012) Integrative survival-based molecular profiling of human pancreatic cancer. *Clinical cancer research : an official journal of the American Association for Cancer Research* 18(5):1352–63.

Dong X, Hao Y, Wang X, Tian W (2016) LEGO: a novel method for gene set over-representation analysis by incorporating network-based gene weights *Scientific Reports* 6:18871.

Draghici S, Khatri P, Tarca AL, Amin K, Done A, Voichita C, Georgescu C, Romero R (2007) A systems biology approach for pathway level analysis. *Genome research* 17(10):1537–45.

Edgar R, Domrachev M, E Lash A (2002) Gene Expression Omnibus: NCBI gene expression and hybridization array data repository *Nucleic Acids Research* 30(1):207–210.

Efron B, Tibshirani R (2007) On testing the significance of sets of genes *The Annals of Applied Statistics* 1(1):107–129.

Fabregat A, Sidiropoulos K, Garapati P, Gillespie M, Hausmann K, Haw R, Jassal B, Jupe S, Korninger F, McKay S, Matthews L, May B, Milacic M, Rothfels K, Shamovsky V, Webber M, Weiser J, Williams M, Wu G, Stein L, Hermjakob H, D'Eustachio P (2015) The Reactome pathway Knowledgebase. *Nucleic acids research* 44(D1):D481–7.

Falcon S, Gentleman R (2007) Using GOstats to test gene lists for GO term association. *Bioinformatics (Oxford, England)* 23(2):257–8.

Frings O, Alexeyenko A, Sonnhammer ELL (2013) MGclus: network clustering employing shared neighbors *Molecular BioSystems* 9(7):1670.

Fujii H (2005) [PPARs-mediated intracellular signal transduction]. *Nihon rinsho. Japanese journal of clinical medicine* 63(4):565–71.

Galamb O, Györffy B, Sipos F, Spisák S, Németh AM, Miheller P, Tulassay Z, Dinya E, Molnár B (2008) Inflammation, adenoma and cancer: objective classification of colon biopsy specimens with gene expression signature. *Disease markers* 25(1):1–16.

Gatti DM, Barry WT, Nobel AB, Rusyn I, Wright FA (2010) Heading Down the Wrong Pathway: on the Influence of Correlation within Gene Sets *BMC Genomics* 11(1):574.

Glaab E, Baudot A, Krasnogor N, Schneider R, Valencia A (2012) EnrichNet: network-based gene set enrichment analysis. *Bioinformatics (Oxford, England)* 28(18):i451–i457.

Glazko GV, Emmert-Streib F (2009) Unite and conquer: univariate and multivariate approaches for finding differentially expressed gene sets. *Bioinformatics (Oxford, England)* 25(18):2348–54.

Goeman JJ, Bühlmann P (2007) Analyzing gene expression data in terms of gene sets: methodological issues. *Bioinformatics (Oxford, England)* 23(8):980–7.

Gyorffy B, Molnar B, Lage H, Szallasi Z, Eklund AC (2009) Evaluation of microarray preprocessing algorithms based on concordance with RT-PCR in clinical samples. *PloS one* 4(5):e5645.

He H, Jazdzewski K, Li W, Liyanarachchi S, Nagy R, Volinia S, Calin GA, Liu CG, Franssila K, Suster S, Kloos RT, Croce CM, de la Chapelle A (2005) The role of microRNA genes in papillary thyroid carcinoma. *Proceedings of the National Academy of Sciences of the United States of America* 102(52):19075–80.

Hever A, Roth RB, Hevezi P, Marin ME, Acosta JA, Acosta H, Rojas J, Herrera R, Grigoriadis D, White E, Conlon PJ, Maki RA, Zlotnik A (2007) Human endometriosis is associated with plasma cells and overexpression of B lymphocyte stimulator. *Proceedings of the National Academy of Sciences of the United States of America* 104(30):12451–6.

Hong Y, Downey T, Eu KW, Koh PK, Cheah PY (2010) A 'metastasis-prone' signature for early-stage mismatch-repair proficient sporadic colorectal cancer patients and its implications for possible therapeutics. *Clinical and experimental metastasis* 27(2):83–90.

Hong Y, Ho KS, Eu KW, Cheah PY (2007) A susceptibility gene set for early onset colorectal cancer that integrates diverse signaling pathways: implication for tumorigenesis. *Clinical cancer research : an official journal of the American Association for Cancer Research* 13(4):1107–14.

Hosack DA, Dennis G, Sherman BT, Lane HC, Lempicki RA (2003) Identifying biological themes within lists of genes with EASE. *Genome biology* 4(10):R70.

Hou J, Aerts J, den Hamer B, van Ijcken W, den Bakker M, Riegman P, van der Leest C, van der Spek P, Foekens JA, Hoogsteden HC, Grosveld F, Philipsen S (2010) Gene expression-based classification of non-small cell lung carcinomas and survival prediction. *PloS one* 5(4):e10312.

Huang DW, Sherman BT, Lempicki RA (2009) Bioinformatics enrichment tools: paths to-

ward the comprehensive functional analysis of large gene lists. *Nucleic acids research* 37(1):1–13.

Huang DW, Sherman BT, Tan Q, Collins JR, Alvord WG, Roayaei J, Stephens R, Baseler MW, Lane HC, Lempicki RA (2007) The DAVID Gene Functional Classification Tool: a novel biological module-centric algorithm to functionally analyze large gene lists. *Genome biology* 8(9):R183.

Jiang Z, Gentleman R (2007) Extensions to gene set enrichment. *Bioinformatics (Oxford, England)* 23(3):306–13.

Kanehisa M, Goto S (2000) KEGG: kyoto encyclopedia of genes and genomes. *Nucleic acids research* 28(1):27–30.

Kanehisa M, Goto S, Sato Y, Kawashima M, Furumichi M, Tanabe M (2014) Data, information, knowledge and principle: back to metabolism in KEGG. *Nucleic acids research* 42(Database issue):D199–205.

Khatri P, Sirota M, Butte AJ (2012) Ten years of pathway analysis: current approaches and outstanding challenges. *PLoS computational biology* 8(2):e1002375.

Kinoshita K, Aono Y, Azuma M, Kishi J, Takezaki A, Kishi M, Makino H, Okazaki H, Uehara H, Izumi K, Sone S, Nishioka Y (2013) Antifibrotic effects of focal adhesion kinase inhibitor in bleomycin-induced pulmonary fibrosis in mice. *American journal of respiratory cell and molecular biology* 49(4):536–43.

Lagares D, Busnadiego O, García-Fernández RA, Kapoor M, Liu S, Carter DE, Abraham D, Shi-Wen X, Carreira P, Fontaine BA, Shea BS, Tager AM, Leask A, Lamas S, Rodríguez-Pascual F (2012) Inhibition of focal adhesion kinase prevents experimental lung fibrosis and myofibroblast formation. *Arthritis and rheumatism* 64(5):1653–64.

Le Dieu R, Taussig DC, Ramsay AG, Mitter R, Miraki-Moud F, Fatah R, Lee AM, Lister TA, Gribben JG (2009) Peripheral blood T cells in acute myeloid leukemia (AML) patients at diagnosis have abnormal phenotype and genotype and form defective immune synapses with AML blasts. *Blood* 114(18):3909–16.

Liang WS, Dunckley T, Beach TG, Grover A, Mastroeni D, Walker DG, Caselli RJ, Kukull WA, McKeel D, Morris JC, Hulette C, Schmechel D, Alexander GE, Reiman EM, Rogers J, Stephan DA (2007) Gene expression profiles in anatomically and functionally distinct regions of the normal aged human brain. *Physiological genomics* 28(3):311–22.

Liang WS, Reiman EM, Valla J, Dunckley T, Beach TG, Grover A, Niedzielko TL, Schneider LE, Mastroeni D, Caselli R, Kukull W, Morris JC, Hulette CM, Schmechel D, Rogers J, Stephan DA (2008) Alzheimer's disease is associated with reduced expression of energy metabolism genes in posterior cingulate neurons. *Proceedings of the National Academy of Sciences of the United States of America* 105(11):4441–6.

Liu Z, Yao Z, Li C, Lu Y, Gao C (2011) Gene expression profiling in human high-grade astrocytomas. *Comparative and functional genomics* 2011:245137.

Ma J, Shojaie A, Michailidis G (2014) Network-Based Pathway Enrichment Analysis with Incomplete Network Information page 33.

Maere S, Heymans K, Kuiper M (2005) BiNGO: a Cytoscape plugin to assess overrepresentation of gene ontology categories in biological networks. *Bioinformatics (Oxford, England)* 21(16):3448–9.

Maglott D, Ostell J, Kim PD, Tatusova T (2004) Entrez Gene: gene-centered information at NCBI *Nucleic Acids Research* 33(Database issue):D54–D58.

McCormack T, Frings O, Alexeyenko A, Sonnhammer ELL (2013) Statistical Assessment of Crosstalk Enrichment between Gene Groups in Biological Networks *PLoS ONE* 8(1):e54945.

Milacic M, Haw R, Rothfels K, Wu G, Croft D, Hermjakob H, D'Eustachio P, Stein L (2012) Annotating cancer variants and anti-cancer therapeutics in reactome. *Cancers* 4(4):1180–211.

Mitrea C, Taghavi Z, Bokanizad B, Hanoudi S, Tagett R, Donato M, Voichiţa C, Drăghici S (2013) Methods and approaches in the topology-based analysis of biological pathways. *Frontiers in physiology* 4:278.

Moisan A, Lee YK, Zhang JD, Hudak CS, Meyer CA, Prummer M, Zoffmann S, Truong HH, Ebeling M, Kiialainen A, Gérard R, Xia F, Schinzel RT, Amrein KE, Cowan CA (2015) White-to-brown metabolic conversion of human adipocytes by JAK inhibition. *Nature cell biology* 17(1):57–67.

Pei H, Li L, Fridley BL, Jenkins GD, Kalari KR, Lingle W, Petersen G, Lou Z, Wang L (2009) FKBP51 affects cancer cell response to chemotherapy by negatively regulating Akt. *Cancer cell* 16(3):259–66.

R Development Core Team (2016) R: A Language and Environment for Statistical Computing.

Richard AJ, Stephens JM (2014) The role of JAK-STAT signaling in adipose tissue function. *Biochimica et biophysica acta* 1842(3):431–9.

Ritchie ME, Phipson B, Wu D, Hu Y, Law CW, Shi W, Smyth GK (2015) Limma powers differential expression analyses for RNA-sequencing and microarray studies *Nucleic Acids Research* 43(7):e47.

Sabates-Bellver J, Van der Flier LG, de Palo M, Cattaneo E, Maake C, Rehrauer H, Laczko E, Kurowski MA, Bujnicki JM, Menigatti M, Luz J, Ranalli TV, Gomes V, Pastorelli A, Faggiani R, Anti M, Jiricny J, Clevers H, Marra G (2007) Transcriptome profile of human colorectal adenomas. *Molecular cancer research : MCR* 5(12):1263–75.

Sanchez-Palencia A, Gomez-Morales M, Gomez-Capilla JA, Pedraza V, Boyero L, Rosell R, Fárez-Vidal ME (2011) Gene expression profiling reveals novel biomarkers in nonsmall cell lung cancer. *International journal of cancer* 129(2):355–64.

Schmitt T, Ogris C, Sonnhammer ELL (2014) FunCoup 3.0: database of genome-wide functional coupling networks. *Nucleic acids research* 42(Database issue):D380–8.

Smyth GK (2004) Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical applications in genetics and molecular biology* 3:Article3.

Snel B, Lehmann G, Bork P, Huynen MA (2000) STRING: a web-server to retrieve and display the repeatedly occurring neighbourhood of a gene. *Nucleic acids research* 28(18):3442–4.

Steinberg GR, Kemp BE (2009) AMPK in Health and Disease *Physiological Reviews* 89(3):1025–1078.

Stirewalt DL, Meshinchi S, Kopecky KJ, Fan W, Pogosova-Agadjanyan EL, Engel JH, Cronk MR, Dorcy KS, McQuary AR, Hockenbery D, Wood B, Heimfeld S, Radich JP (2008) Identification of genes with abnormal expression changes in acute myeloid leukemia. *Genes, chromosomes and cancer* 47(1):8–20.

Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES, Mesirov JP (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America* 102(43):15545–50.

Szklarczyk D, Franceschini A, Wyder S, Forslund K, Heller D, Huerta-Cepas J, Simonovic M, Roth A, Santos A, Tsafou KP, Kuhn M, Bork P, Jensen LJ, von Mering C (2015) STRING v10: protein-protein interaction networks, integrated over the tree of life. *Nucleic acids research* 43(Database issue):D447–52.

Tarca AL, Bhatti G, Romero R (2013) A comparison of gene set analysis methods in terms of sensitivity, prioritization and specificity. *PloS one* 8(11):e79217.

Tarca AL, Draghici S, Bhatti G, Romero R (2012) Down-weighting overlapping genes improves gene set analysis. *BMC bioinformatics* 13:136.

The Gene Ontology Consortium (2014) Gene Ontology Consortium: going forward. *Nucleic acids research* 43(D1):D1049–1056.

Tian L, Greenberg SA, Kong SW, Altschuler J, Kohane IS, Park PJ (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the United States of America* 102(38):13544–9.

Uddin S, Ahmed M, Hussain A, Abubaker J, Al-Sanea N, AbdulJabbar A, Ashari LH, Alhomoud S, Al-Dayel F, Jehan Z, Bavi P, Siraj AK, Al-Kuraya KS (2011) Genome-wide expression analysis of Middle Eastern colorectal cancer reveals FOXM1 as a novel target for cancer therapy. *The American journal of pathology* 178(2):537–47.

Uhlen M, Fagerberg L, Hallstrom BM, Lindskog C, Oksvold P, Mardinoglu A, Sivertsson A, Kampf C, Sjostedt E, Asplund A, Olsson I, Edlund K, Lundberg E, Navani S, Szigyarto CAK, Odeberg J, Djureinovic D, Takanen JO, Hober S, Alm T, Edqvist PH, Berling H, Tegel H, Mulder J, Rockberg J, Nilsson P, Schwenk JM, Hamsten M, von Feilitzen K, Forsberg M, Persson L, Johansson F, Zwahlen M, von Heijne G, Nielsen J, Ponten F (2015) Tissue-based map of the human proteome *Science* 347(6220):1260419–1260419.

Uhlen M, Oksvold P, Fagerberg L, Lundberg E, Jonasson K, Forsberg M, Zwahlen M, Kampf C, Wester K, Hober S, Wernerus H, Björling L, Ponten F (2010) Towards a knowledge-based Human Protein Atlas. *Nature biotechnology* 28(12):1248–50.

van Dongen S (2000) Graph Clustering by Flow Simulation .

Voichita C, Donato M, Draghici S (2012) Incorporating gene significance in the impact analysis of signaling pathways in *Proceedings - 2012 11th International Conference on Machine Learning and Applications, ICMLA 2012* volume 1 pages 126–131.

Wang PI, Hwang S, Kincaid RP, Sullivan CS, Lee I, Marcotte EM (2012) RIDDLE: reflective diffusion and local extension reveal functional associations for unannotated gene sets via proximity in a gene network. *Genome biology* 13(12):R125.

Wang Y, Roche O, Yan MS, Finak G, Evans AJ, Metcalf JL, Hast BE, Hanna SC, Wondergem B, Furge KA, Irwin MS, Kim WY, Teh BT, Grinstein S, Park M, Marsden PA, Ohh M (2009) Regulation of endocytosis via the oxygen-sensing pathway. *Nature medicine* 15(3):319–24.

Xu D, Yin C, Wang S, Xiao Y (2013) JAK-STAT in lipid metabolism of adipocytes. *JAK-STAT* 2(4):e27203.

Zeeberg BR, Feng W, Wang G, Wang MD, Fojo AT, Sunshine M, Narasimhan S, Kane DW, Reinhold WC, Lababidi S, Bussey KJ, Riss J, Barrett JC, Weinstein JN (2003) GoMiner: a resource for biological interpretation of genomic and proteomic data. *Genome Biology* 4(4):R28.

Part A: Gold standard data

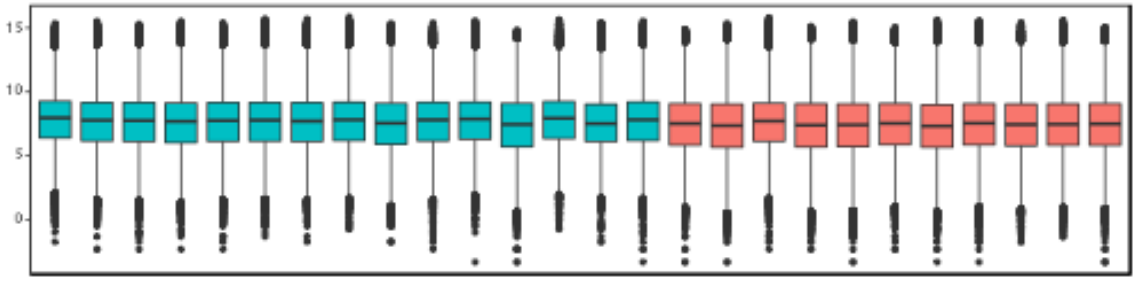
A.1 Value distributions of gold standard microarray data

Below are the samples used for every dataset in table 3.1, boxplots of the expression value distributions are also included. The first a boxplots are the disease tissue samples while the last b boxplots are the healthy tissue samples.

GSE1145

disease samples: GSM18422 GSM18423 GSM18424 GSM18425 GSM18426 GSM18427 GSM18428 GSM18429 GSM18430 GSM18431 GSM18432 GSM18433 GSM18434 GSM18435 GSM18436

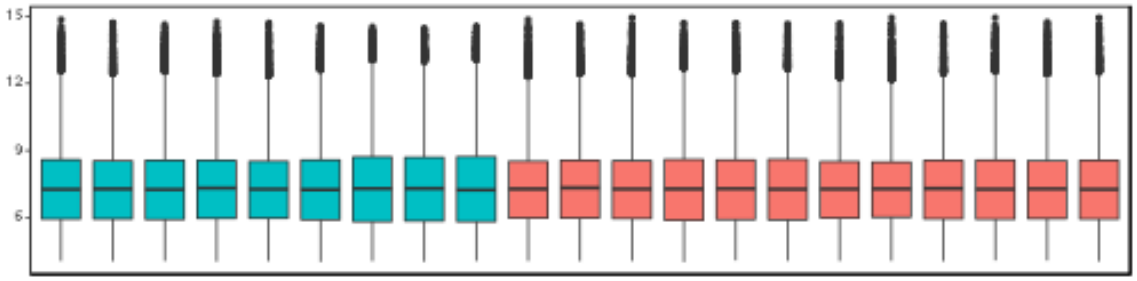
control samples: GSM18442 GSM18443 GSM18444 GSM18445 GSM18446 GSM18447 GSM18448 GSM18449 GSM18450 GSM18451 GSM18452



GSE14762

disease samples: GSM368639 GSM368640 GSM368641 GSM368642 GSM368643 GSM368644 GSM368645 GSM368646 GSM368648

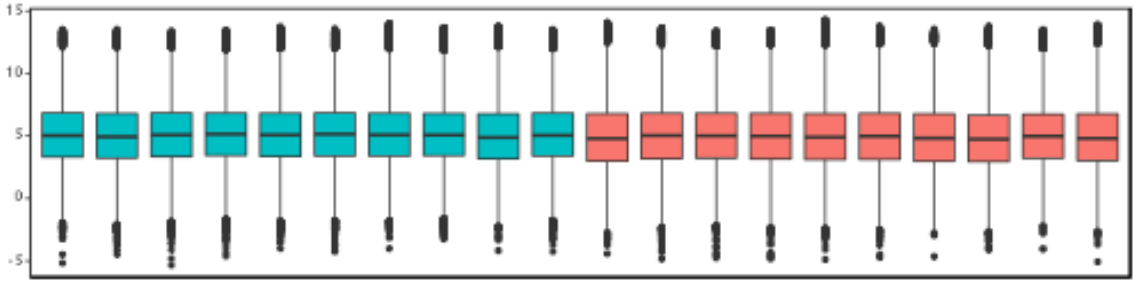
control samples: GSM368649 GSM368650 GSM368651 GSM368652 GSM368653 GSM368654 GSM368655 GSM368656 GSM368657 GSM368658 GSM368659 GSM368660



GSE14924-CD4

disease samples: GSM372701 GSM372702 GSM372703 GSM372704 GSM372705 GSM372706 GSM372707 GSM372708 GSM372709 GSM372710

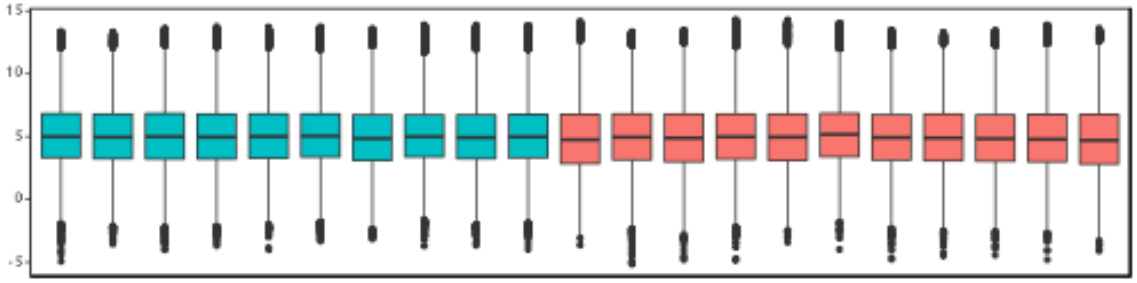
control samples: GSM372721 GSM372722 GSM372723 GSM372724 GSM372725 GSM372726 GSM372727 GSM372728 GSM372729 GSM372730



GSE14924-CD8

disease samples: GSM372711 GSM372712 GSM372713 GSM372714 GSM372715 GSM372716 GSM372717 GSM372718 GSM372719 GSM372720

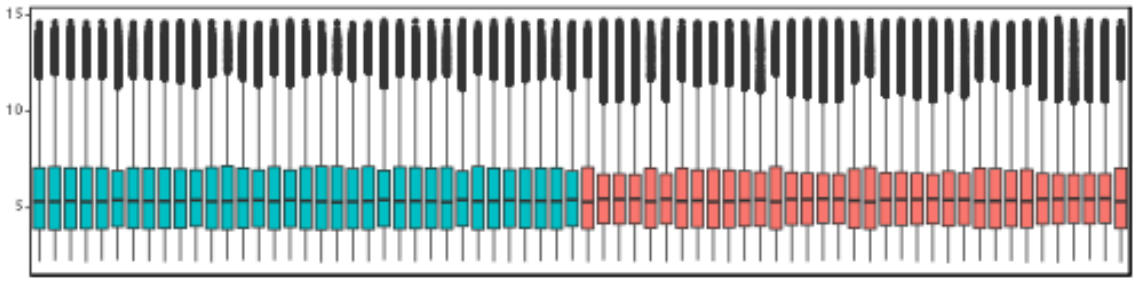
control samples: GSM372731 GSM372732 GSM372733 GSM372734 GSM372735 GSM372736 GSM372737 GSM372738 GSM372739 GSM372740 GSM372741



GSE15471

disease samples: GSM388115 GSM388117 GSM388119 GSM388121 GSM388122 GSM388123 GSM388124 GSM388125 GSM388126 GSM388127 GSM388128 GSM388129 GSM388130 GSM388131 GSM388132 GSM388133 GSM388134 GSM388135 GSM388136 GSM388137 GSM388138 GSM388139 GSM388140 GSM388141 GSM388142 GSM388143 GSM388144 GSM388145 GSM388146 GSM388147 GSM388148 GSM388149 GSM388151 GSM388152 GSM388153

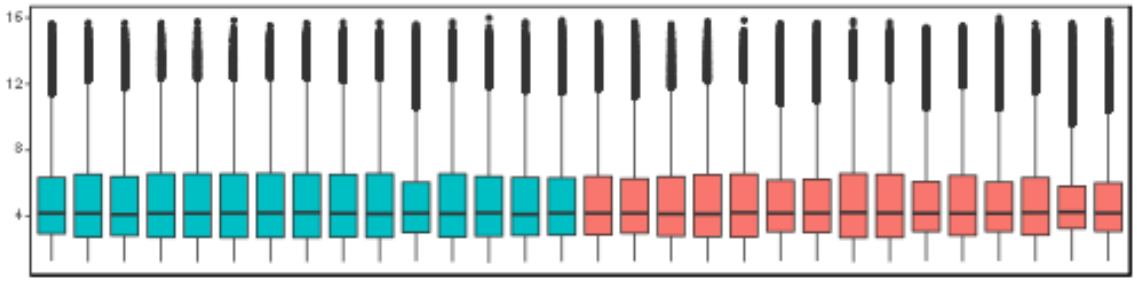
control samples: GSM388076 GSM388078 GSM388080 GSM388082 GSM388083 GSM388084 GSM388085 GSM388086 GSM388087 GSM388088 GSM388089 GSM388090 GSM388091 GSM388092 GSM388093 GSM388094 GSM388095 GSM388096 GSM388097 GSM388098 GSM388099 GSM388100 GSM388101 GSM388102 GSM388103 GSM388104 GSM388105 GSM388106 GSM388107 GSM388108 GSM388109 GSM388110 GSM388112 GSM388113 GSM388114



GSE16515

disease samples: GSM414927 GSM414929 GSM414933 GSM414937 GSM414939 GSM414941 GSM414946 GSM414952 GSM414954 GSM414956 GSM414962 GSM414965 GSM414969 GSM414971 GSM414974

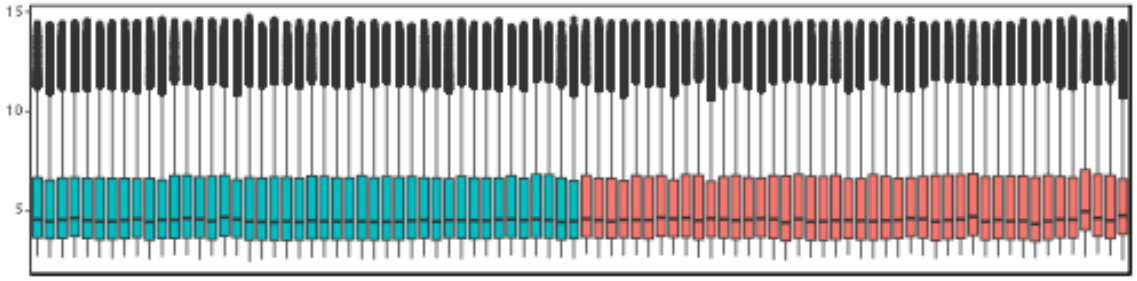
control samples: GSM414928 GSM414930 GSM414934 GSM414938 GSM414940 GSM414942 GSM414947 GSM414953 GSM414955 GSM414957 GSM414963 GSM414966 GSM414970 GSM414972 GSM414975



GSE18842

disease samples: GSM466947 GSM466949 GSM466951 GSM466952 GSM466954 GSM466956 GSM466958 GSM466960 GSM466962 GSM466963 GSM466965 GSM466967 GSM466969 GSM466971 GSM466973 GSM466975 GSM466977 GSM466980 GSM466982 GSM466983 GSM466985 GSM466987 GSM466989 GSM466991 GSM466993 GSM466994 GSM466996 GSM466998 GSM467004 GSM467006 GSM467008 GSM467010 GSM467012 GSM467014 GSM467016 GSM467018 GSM467021 GSM467023 GSM467026 GSM467028 GSM467029 GSM467032 GSM467034 GSM467036

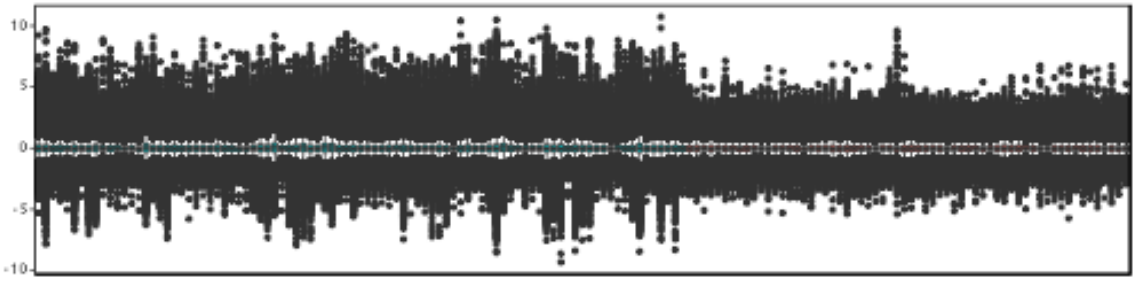
control samples: GSM466948 GSM466950 GSM466953 GSM466955 GSM466957 GSM466959 GSM466961 GSM466964 GSM466966 GSM466968 GSM466970 GSM466972 GSM466974 GSM466976 GSM466978 GSM466979 GSM466981 GSM466984 GSM466986 GSM466988 GSM466990 GSM466992 GSM466995 GSM466997 GSM466999 GSM467000 GSM467001 GSM467002 GSM467003 GSM467005 GSM467007 GSM467009 GSM467011 GSM467013 GSM467015 GSM467017 GSM467020 GSM467022 GSM467025 GSM467027 GSM467031 GSM467033 GSM467035 GSM467037



GSE19188

disease samples: GSM475656 GSM475661 GSM475662 GSM475664 GSM475668 GSM475670 GSM475672 GSM475674 GSM475676 GSM475677 GSM475679 GSM475681 GSM475683 GSM475685 GSM475687 GSM475689 GSM475691 GSM475692 GSM475694 GSM475696 GSM475698 GSM475700 GSM475701 GSM475703 GSM475706 GSM475708 GSM475709 GSM475710 GSM475712 GSM475713 GSM475715 GSM475717 GSM475719 GSM475720 GSM475722 GSM475724 GSM475727 GSM475728 GSM475730 GSM475731 GSM475733 GSM475735 GSM475737 GSM475739 GSM475741 GSM475744 GSM475747 GSM475748 GSM475751 GSM475753 GSM475756 GSM475758 GSM475759 GSM475760 GSM475761 GSM475762 GSM475763 GSM475765 GSM475768 GSM475769 GSM475770 GSM475772 GSM475773 GSM475774 GSM475776 GSM475777 GSM475778 GSM475779 GSM475780 GSM475782 GSM475784 GSM475785 GSM475787 GSM475788 GSM475789 GSM475791 GSM475792 GSM475793 GSM475794 GSM475795 GSM475796 GSM475797 GSM475799 GSM475801 GSM475802 GSM475803 GSM475804 GSM475805 GSM475806 GSM475808 GSM475810

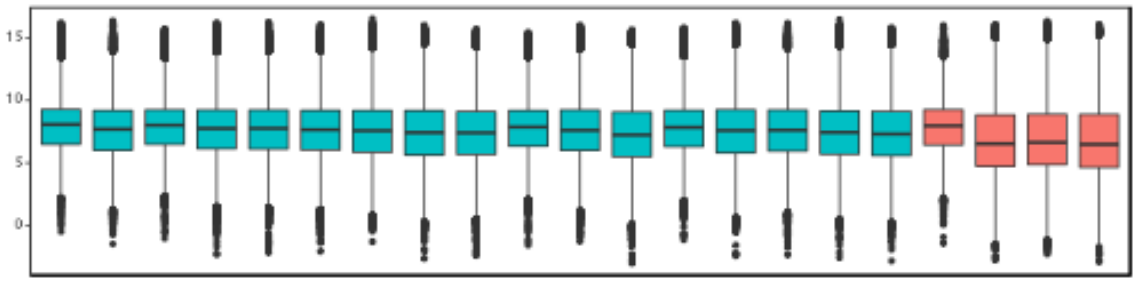
control samples: GSM475657 GSM475658 GSM475660 GSM475663 GSM475665 GSM475667 GSM475669 GSM475671 GSM475673 GSM475675 GSM475678 GSM475680 GSM475682 GSM475684 GSM475686 GSM475688 GSM475690 GSM475693 GSM475695 GSM475697 GSM475699 GSM475702 GSM475704 GSM475705 GSM475707 GSM475711 GSM475714 GSM475716 GSM475718 GSM475721 GSM475723 GSM475725 GSM475726 GSM475729 GSM475732 GSM475734 GSM475736 GSM475738 GSM475740 GSM475742 GSM475743 GSM475745 GSM475746 GSM475749 GSM475750 GSM475752 GSM475754 GSM475755 GSM475757 GSM475764 GSM475766 GSM475767 GSM475771 GSM475775 GSM475783 GSM475786 GSM475790 GSM475798 GSM475800 GSM475807 GSM475809 GSM475811



GSE19728

disease samples: GSM492650 GSM492651 GSM492652 GSM492653 GSM492654 GSM492655 GSM492656 GSM492657 GSM492658 GSM492659 GSM492660 GSM492661 GSM492662 GSM492663 GSM492664 GSM492665 GSM492666

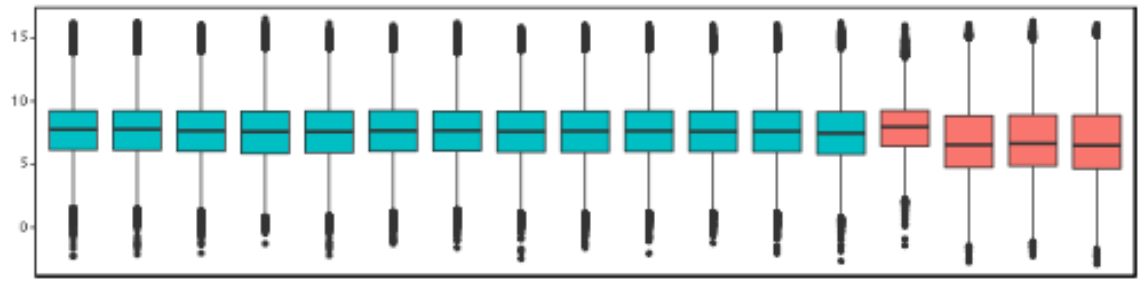
control samples: GSM492649 GSM525014 GSM525015 GSM525016



GSE21354

disease samples: GSM492653 GSM492654 GSM492655 GSM492656 GSM533622 GSM533623 GSM533624 GSM533625 GSM533626 GSM533627 GSM533628 GSM533629 GSM533630

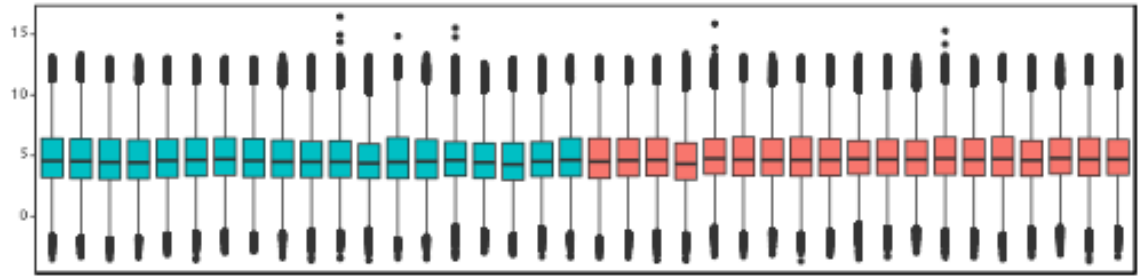
control samples: GSM492649 GSM525014 GSM525015 GSM525016



GSE23878

disease samples: GSM588828 GSM588829 GSM588831 GSM588832 GSM588833 GSM588835 GSM588838 GSM588839 GSM588840 GSM588841 GSM588842 GSM588843 GSM588844 GSM588845 GSM588846 GSM588847 GSM588849 GSM588850 GSM588852

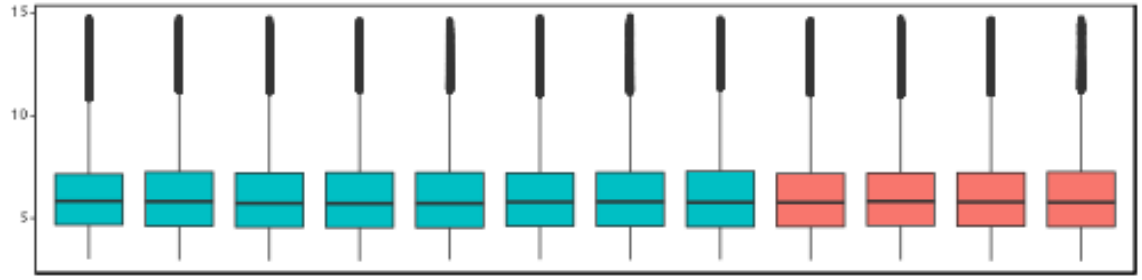
control samples: GSM588863 GSM588864 GSM588865 GSM588867 GSM588868 GSM588871 GSM588873 GSM588874 GSM588875 GSM588876 GSM588877 GSM588878 GSM588879 GSM588880 GSM588881 GSM588882 GSM588884 GSM588885 GSM588886



GSE24739-G0

disease samples: GSM609346 GSM609347 GSM609348 GSM609349 GSM609350 GSM609351 GSM609352 GSM609353

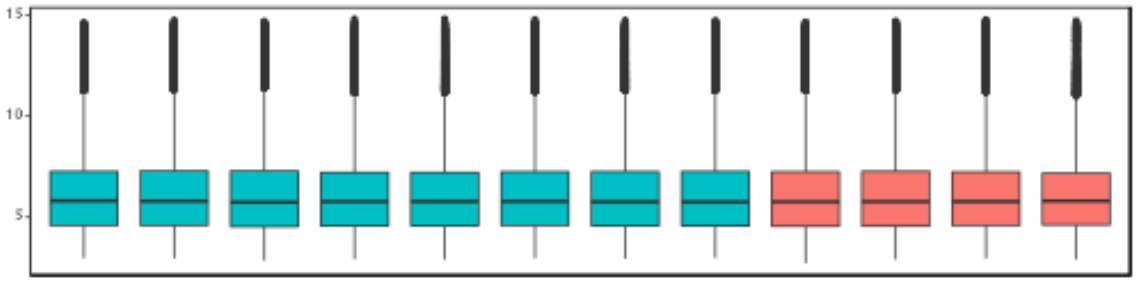
control samples: GSM609354 GSM609355 GSM609356 GSM609357



GSE24739-G1

disease samples: GSM609358 GSM609359 GSM609360 GSM609361 GSM609362 GSM609363 GSM609364 GSM609365

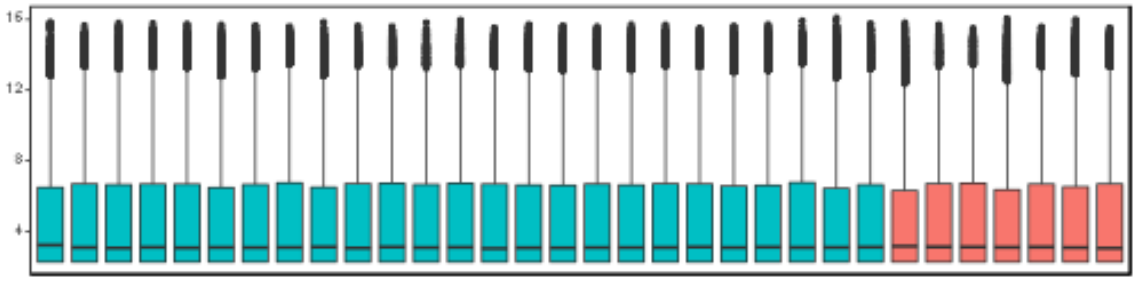
control samples: GSM609366 GSM609367 GSM609368 GSM609369



GSE32676

disease samples: GSM811004 GSM811005 GSM811006 GSM811007 GSM811008 GSM811009 GSM811010 GSM811011 GSM811012 GSM811013 GSM811014 GSM811015 GSM811016 GSM811017 GSM811018 GSM811019 GSM811020 GSM811021 GSM811022 GSM811023 GSM811024 GSM811025 GSM811026 GSM811027 GSM811028

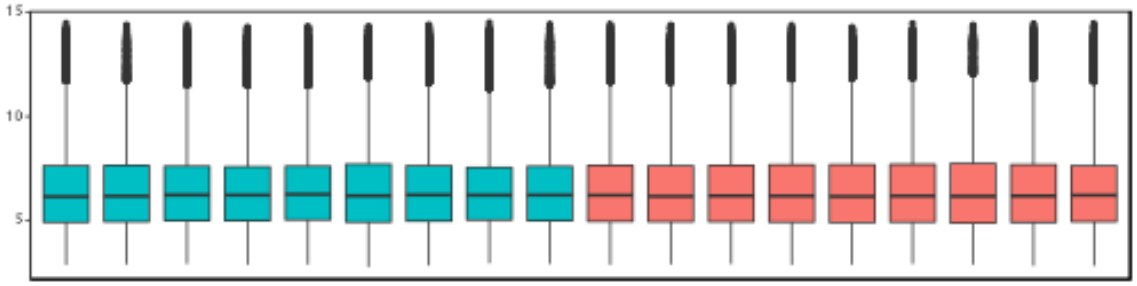
control samples: GSM811029 GSM811030 GSM811031 GSM811032 GSM811033 GSM811034 GSM811035



GSE3467

disease samples: GSM77363 GSM77365 GSM77367 GSM77369 GSM77371 GSM77373 GSM77375 GSM77377 GSM77379

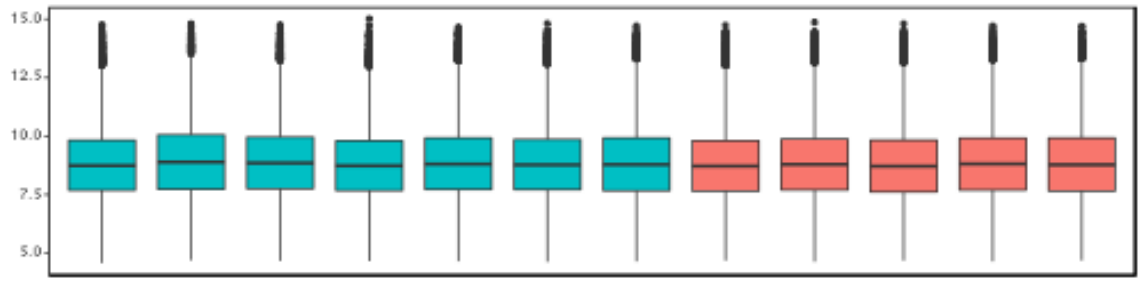
control samples: GSM77362 GSM77364 GSM77366 GSM77368 GSM77370 GSM77372 GSM77374 GSM77376 GSM77378



GSE3585

disease samples: GSM82386 GSM82387 GSM82388 GSM82389 GSM82390 GSM82391 GSM82392

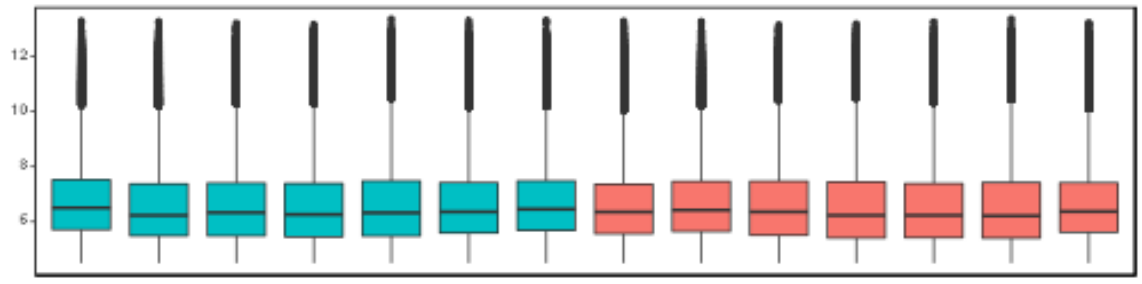
control samples: GSM82381 GSM82382 GSM82383 GSM82384 GSM82385



GSE3678

disease samples: GSM85222 GSM85223 GSM85224 GSM85225 GSM85226 GSM85227 GSM85228

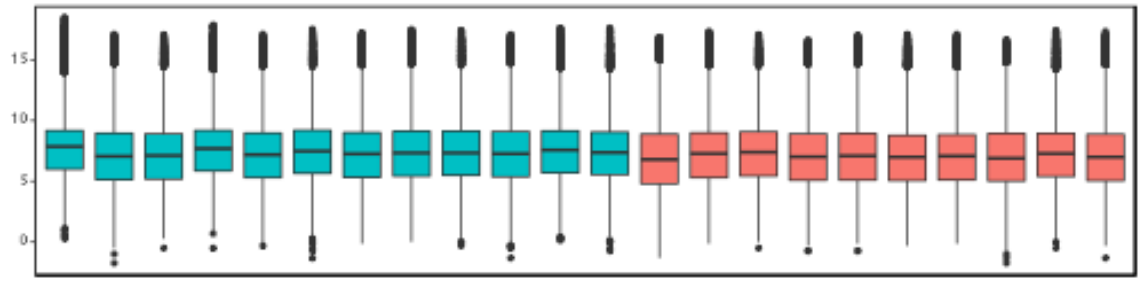
control samples: GSM85215 GSM85216 GSM85217 GSM85218 GSM85219 GSM85220 GSM85221



GSE4107

disease samples: GSM93789 GSM93920 GSM93921 GSM93922 GSM93923 GSM93924 GSM93925 GSM93926 GSM93927 GSM93928 GSM93929 GSM93932

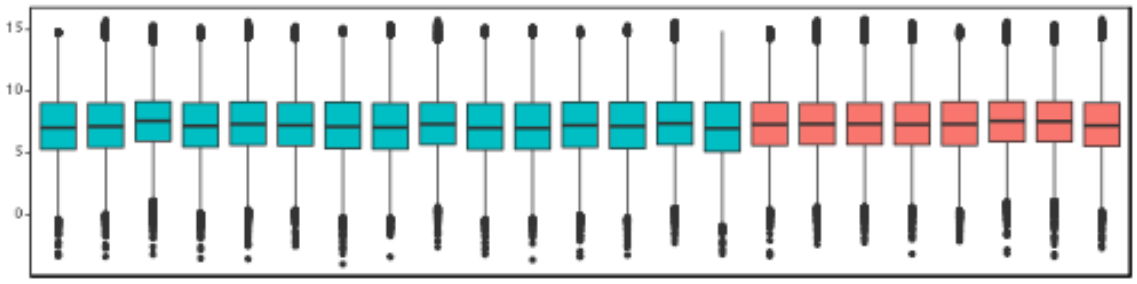
control samples: GSM93938 GSM93939 GSM93941 GSM93943 GSM93944 GSM93946 GSM93948 GSM93950 GSM93952 GSM93954



GSE4183

disease samples: GSM95496 GSM95497 GSM95498 GSM95499 GSM95500 GSM95501 GSM95502 GSM95503 GSM95504 GSM95505 GSM95506 GSM95507 GSM95508 GSM95509 GSM95510

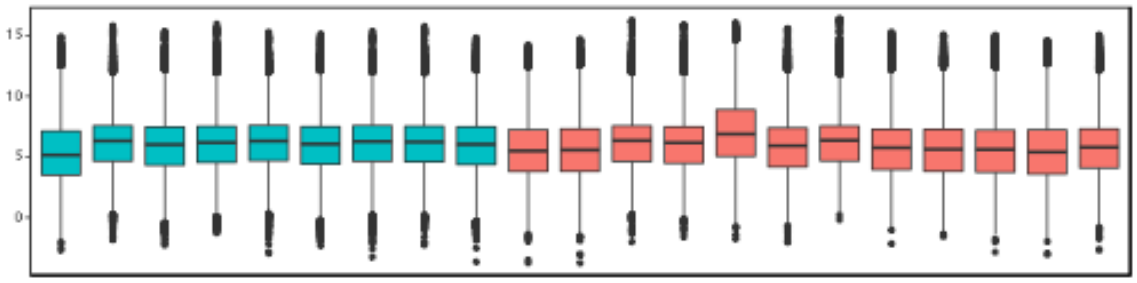
control samples: GSM95473 GSM95474 GSM95475 GSM95476 GSM95477 GSM95478 GSM95479 GSM95480



GSE5281-EC

disease samples: GSM238790 GSM238791 GSM238792 GSM238793 GSM238794 GSM238795 GSM238796 GSM238797 GSM238798

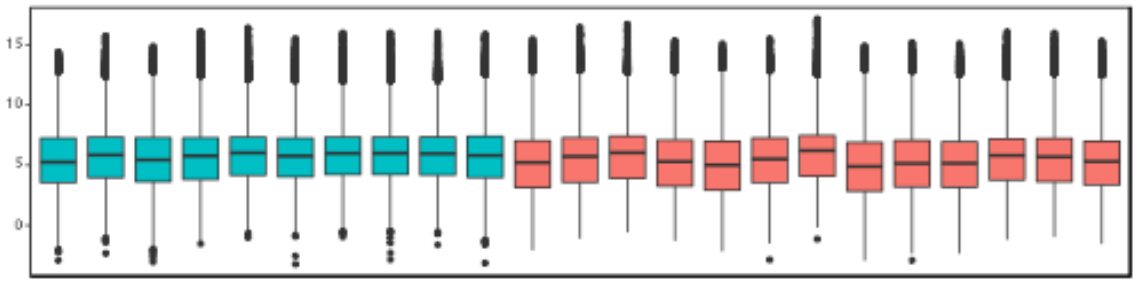
control samples: GSM119615 GSM119616 GSM119617 GSM119618 GSM119619 GSM119620 GSM119621 GSM119622 GSM119623 GSM119624 GSM119625 GSM119627



GSE5281-HIP

disease samples: GSM238799 GSM238800 GSM238801 GSM238802 GSM238803 GSM238804 GSM238805 GSM238806 GSM238807 GSM238808

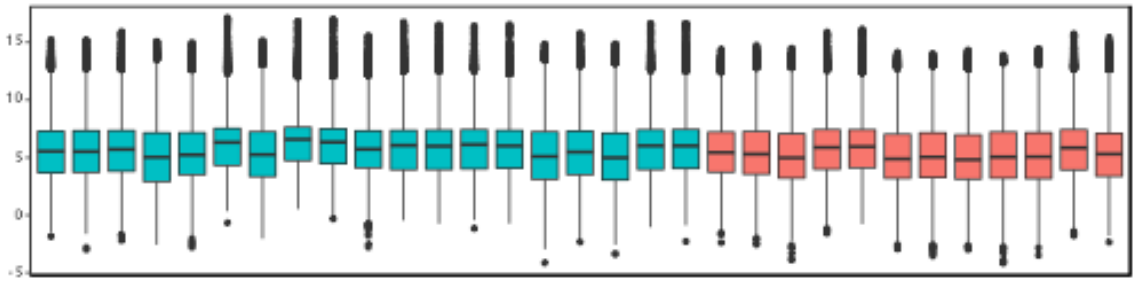
control samples: GSM119628 GSM119629 GSM119630 GSM119631 GSM119632 GSM119633 GSM119634 GSM119635 GSM119636 GSM119637 GSM119638 GSM119639 GSM119640



GSE5281-VCX

disease samples: GSM238872 GSM238873 GSM238874 GSM238875 GSM238877 GSM238941 GSM238942 GSM238943 GSM238944 GSM238945 GSM238946 GSM238947 GSM238948 GSM238949 GSM238951 GSM238952 GSM238953 GSM238955 GSM238963

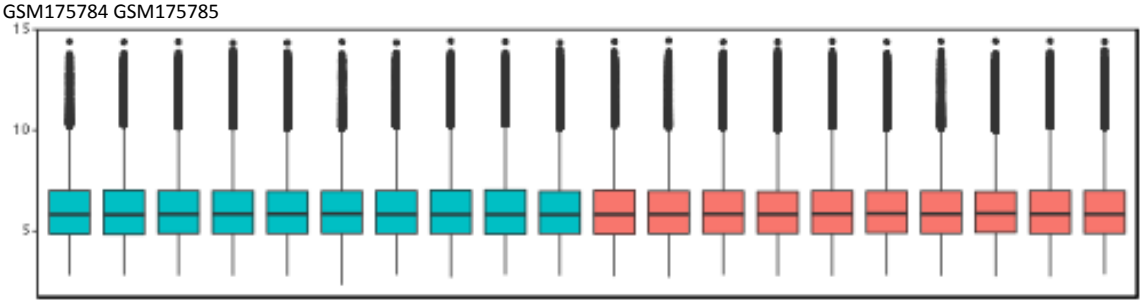
control samples: GSM119677 GSM119678 GSM119679 GSM119680 GSM119681 GSM119682 GSM119683 GSM119684 GSM119685 GSM119686 GSM119687 GSM119688



GSE7305

disease samples: GSM175766 GSM175767 GSM175768 GSM175769 GSM175770 GSM175771 GSM175772 GSM175773 GSM175774 GSM175775

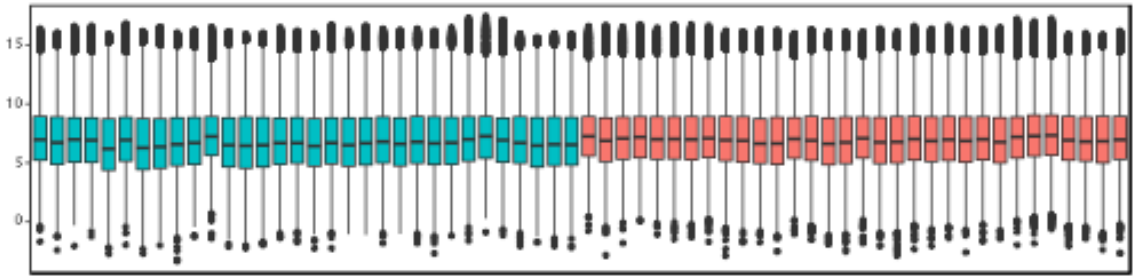
control samples: GSM175776 GSM175777 GSM175778 GSM175779 GSM175780 GSM175781 GSM175782 GSM175783 GSM175784 GSM175785



GSE8671

disease samples: GSM215083 GSM215084 GSM215085 GSM215086 GSM215087 GSM215088 GSM215089 GSM215090 GSM215091 GSM215092 GSM215093 GSM215094 GSM215095 GSM215096 GSM215097 GSM215098 GSM215099 GSM215100 GSM215101 GSM215102 GSM215103 GSM215104 GSM215105 GSM215106 GSM215107 GSM215108 GSM215109 GSM215110 GSM215111 GSM215112 GSM215113 GSM215114

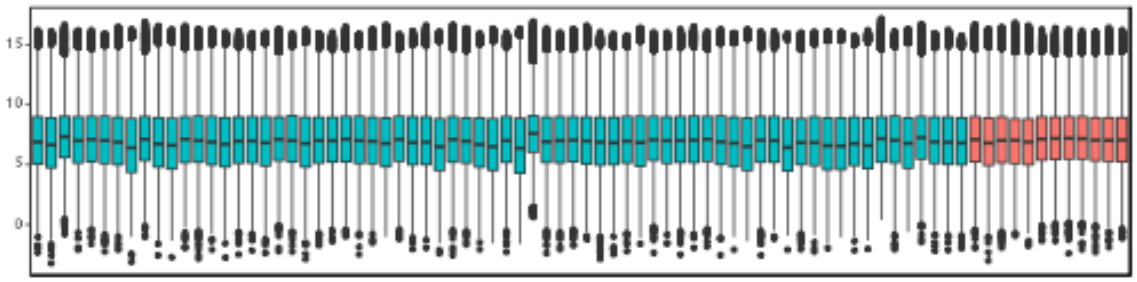
control samples: GSM215051 GSM215052 GSM215053 GSM215054 GSM215055 GSM215056 GSM215057 GSM215058 GSM215059 GSM215060 GSM215061 GSM215062 GSM215063 GSM215064 GSM215065 GSM215066 GSM215067 GSM215068 GSM215069 GSM215070 GSM215071 GSM215072 GSM215073 GSM215074 GSM215075 GSM215076 GSM215077 GSM215078 GSM215079 GSM215080 GSM215081 GSM215082



GSE9348

disease samples: GSM237914 GSM237915 GSM237916 GSM237917 GSM237918 GSM237919 GSM237920 GSM237921 GSM237922 GSM237923 GSM237924 GSM237925 GSM237926 GSM237927 GSM237928 GSM237929 GSM237930 GSM237931 GSM237932 GSM237933 GSM237934 GSM237935 GSM237936 GSM237937 GSM237938 GSM237939 GSM237940 GSM237941 GSM237942 GSM237943 GSM237944 GSM237945 GSM237946 GSM237947 GSM237948 GSM237949 GSM237950 GSM237951 GSM237952 GSM237953 GSM237954 GSM237955 GSM237956 GSM237957 GSM237958 GSM237959 GSM237960 GSM237961 GSM237962 GSM237963 GSM237964 GSM237965 GSM237966 GSM237967 GSM237968 GSM237969 GSM237970 GSM237971 GSM237972 GSM237973 GSM237974 GSM237975 GSM237976 GSM237977 GSM237978 GSM237979 GSM237980 GSM237981 GSM237982 GSM237983

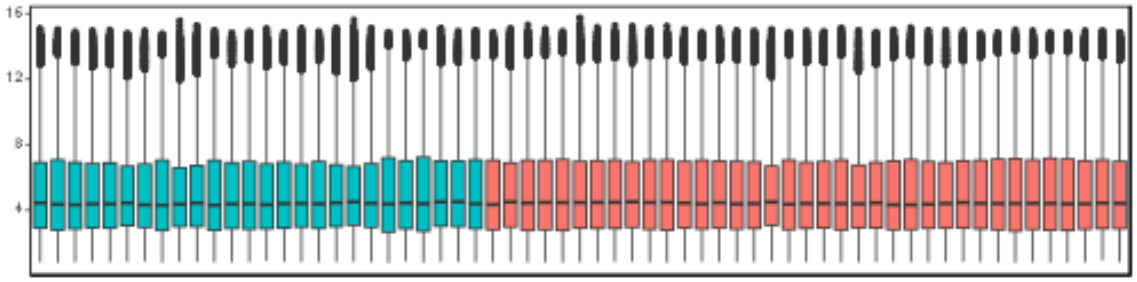
control samples: GSM237984 GSM237985 GSM237986 GSM237987 GSM237988 GSM237989 GSM237990 GSM237991 GSM237992 GSM237993 GSM237994 GSM237995



GSE9476

disease samples: GSM239345 GSM239346 GSM239348 GSM239363 GSM239371 GSM239460 GSM239485 GSM239487 GSM239488 GSM239489 GSM239490 GSM239491 GSM239492 GSM239493 GSM239494 GSM239495 GSM239496 GSM239497 GSM239498 GSM239516 GSM239520 GSM239580 GSM240405 GSM240406 GSM240427 GSM240429

control samples: GSM239170 GSM239323 GSM239324 GSM239326 GSM239328 GSM239329 GSM239331 GSM239332 GSM239333 GSM239334 GSM239335 GSM239338 GSM239339 GSM239340 GSM239341 GSM239342 GSM239343 GSM239344 GSM240430 GSM240431 GSM240432 GSM240494 GSM240495 GSM240496 GSM240497 GSM240498 GSM240499 GSM240500 GSM240501 GSM240502 GSM240503 GSM240504 GSM240505 GSM240506 GSM240507 GSM240508 GSM240509



A.2 Gene sets used for MSigDB gold standard data

The original gold standard data given in table 3.1 only contains 26 data points, therefore results based on this might be unreliable. To confirm that the effects seen with the original gold standard data set are not a result of specific peculiarities of this data, another validation set was needed. Below is the validation set comprising 61 gene sets compiled from the MSigDB C2 collection. Each of these is matched to a KEGG pathway that is expected to be affected by the experimental condition giving rise to the gene set. This is the data that was used in section 3.2.3.

MSigDB gene set	target pathway	KEGG ID	n
AGUIRRE-PANCREATIC-CANCER-COPY-NUMBER-UP AGUIRRE-PANCREATIC-CANCER-COPY-NUMBER-DN	Pancreatic cancer	hsa05212	572
ALCALA-APOPTOSIS	Apoptosis	hsa04210	87
BARIS-THYROID-CANCER-UP BARIS-THYROID-CANCER-DN	Thyroid cancer	hsa05216	73
BEIER-GLIOMA-STEM-CELL-UP BEIER-GLIOMA-STEM-CELL-DN	Glioma	hsa05214	108
BENNETT-SYSTEMIC-LUPUS-ERYTHEMATOSUS	Systemic lupus erythematosus	hsa05322	23
BOHN-PRIMARY-IMMUNODEFICIENCY-SYNDROM-UP BOHN-PRIMARY-IMMUNODEFICIENCY-SYNDROM-DN	Primary immunodeficiency	hsa05340	62
CONCANNON-APOPTOSIS-BY-EPOXOMICIN-UP CONCANNON-APOPTOSIS-BY-EPOXOMICIN-DN	Apoptosis	hsa04210	428
DELYS-THYROID-CANCER-UP DELYS-THYROID-CANCER-DN	Thyroid cancer	hsa05216	614
DUTTA-APOPTOSIS-VIA-NFKB	Apoptosis	hsa04210	31
EGUCHI-CELL-CYCLE-RB1-TARGETS	Cell cycle	hsa04110	19
GEORGES-CELL-CYCLE-MIR192-TARGETS	Cell cycle	hsa04110	59
GRUETZMANN-PANCREATIC-CANCER-UP GRUETZMANN-PANCREATIC-CANCER-DN	Pancreatic cancer	hsa05212	542
HAMAI-APOPTOSIS-VIA-TRAIL-UP HAMAI-APOPTOSIS-VIA-TRAIL-DN	Apoptosis	hsa04210	460
HEIDENBLAD-AMPLIFIED-IN-PANCREATIC-CANCER	Pancreatic cancer	hsa05212	31

HOLLMAN-APOPTOSIS-VIA-CD40	Apoptosis	hsa04210	489
HWANG-PROSTATE-CANCER-MARKERS	Prostate cancer	hsa05215	29
KAUFFMANN-MELANOMA-RELAPSE-UP KAUFFMANN-MELANOMA-RELAPSE-DN	Melanoma	hsa05218	63
KONDO-PROSTATE-CANCER-HCP-WITH-H3K27ME3	Prostate cancer	hsa05215	97
KONDO-PROSTATE-CANCER-WITH-H3K27ME3	Prostate cancer	hsa05215	196
KUUSELO-PANCREATIC-CANCER-19Q13-AMPLIFICATION	Pancreatic cancer	hsa05212	28
LAIHO-COLORECTAL-CANCER-SERRATED-UP LAIHO-COLORECTAL-CANCER-SERRATED-DN	Colorectal cancer	hsa05210	214
LAU-APOPTOSIS-CDKN2A-UP LAU-APOPTOSIS-CDKN2A-DN	Apoptosis	hsa04210	60
LINDGREN-BLADDER-CANCER-CLUSTER-1-UP LINDGREN-BLADDER-CANCER-CLUSTER-1-DN	Bladder cancer	hsa05219	496
LINDGREN-BLADDER-CANCER-CLUSTER-2A-UP LINDGREN-BLADDER-CANCER-CLUSTER-2A-DN	Bladder cancer	hsa05219	149
LINDGREN-BLADDER-CANCER-CLUSTER-2B	Bladder cancer	hsa05219	389
LINDGREN-BLADDER-CANCER-CLUSTER-3-UP LINDGREN-BLADDER-CANCER-CLUSTER-3-DN	Bladder cancer	hsa05219	548
LINDGREN-BLADDER-CANCER-HIGH-RECURRENCE	Bladder cancer	hsa05219	43
LINDGREN-BLADDER-CANCER-WITH-LOH-IN-CHR9Q	Bladder cancer	hsa05219	116
LIN-MELANOMA-COPY-NUMBER-UP LIN-MELANOMA-COPY-NUMBER-DN	Melanoma	hsa05218	106
LIU-PROSTATE-CANCER-UP LIU-PROSTATE-CANCER-DN	Prostate cancer	hsa05215	102
LUI-THYROID-CANCER-CLUSTER-1	Thyroid cancer	hsa05216	53
LUI-THYROID-CANCER-CLUSTER-2	Thyroid cancer	hsa05216	44
LUI-THYROID-CANCER-CLUSTER-3	Thyroid cancer	hsa05216	29
LUI-THYROID-CANCER-CLUSTER-4	Thyroid cancer	hsa05216	27
LUI-THYROID-CANCER-CLUSTER-5	Thyroid cancer	hsa05216	19
LUI-THYROID-CANCER-PAX8-PPARG-UP LUI-THYROID-CANCER-PAX8-PPARG-DN	Thyroid cancer	hsa05216	96
MONTERO-THYROID-CANCER-POOR-SURVIVAL-UP MONTERO-THYROID-CANCER-POOR-SURVIVAL-DN	Thyroid cancer	hsa05216	19

NGO-MALIGNANT-GLIOMA-1P-LOH	Glioma	hsa05214	7
NUTT-GBM-VS-AO-GLIOMA-UP NUTT-GBM-VS-AO-GLIOMA-DN	Glioma	hsa05214	94
OSMAN-BLADDER-CANCER-UP OSMAN-BLADDER-CANCER-DN	Bladder cancer	hsa05219	411
OUYANG-PROSTATE-CANCER-MARKERS	Prostate cancer	hsa05215	24
OUYANG-PROSTATE-CANCER-PROGRESSION-UP OUYANG-PROSTATE-CANCER-PROGRESSION-DN	Prostate cancer	hsa05215	41
ROSS-ACUTE-MYELOID-LEUKEMIA-CBF	Acute myeloid leukemia	hsa05221	86
ROVERSI-GLIOMA-COPY-NUMBER-UP ROVERSI-GLIOMA-COPY-NUMBER-DN	Glioma	hsa05214	132
ROVERSI-GLIOMA-LOH-REGIONS	Glioma	hsa05214	39
SATO-SILENCED-BY-METHYLATION-IN-PANCREATIC-CANCER-2	Pancreatic cancer	hsa05212	48
SATO-SILENCED-EPIGENETICALLY-IN-PANCREATIC-CANCER	Pancreatic cancer	hsa05212	46
SCIAN-CELL-CYCLE-TARGETS-OF-TP53-AND-TP73-UP SCIAN-CELL-CYCLE-TARGETS-OF-TP53-AND-TP73-DN	Cell cycle	hsa04110	31
SETLUR-PROSTATE-CANCER-TMPRSS2-ERG-FUSION-UP SETLUR-PROSTATE-CANCER-TMPRSS2-ERG-FUSION-DN	Prostate cancer	hsa05215	83
STEGMEIER-PRE-MITOTIC-CELL-CYCLE-REGULATORS	Cell cycle	hsa04110	11
TOMLINS-PROSTATE-CANCER-UP TOMLINS-PROSTATE-CANCER-DN	Prostate cancer	hsa05215	73
WALLACE-PROSTATE-CANCER-UP WALLACE-PROSTATE-CANCER-DN	Prostate cancer	hsa05215	24
WALLACE-PROSTATE-CANCER-RACE-UP WALLACE-PROSTATE-CANCER-RACE-DN	Prostate cancer	hsa05215	456
WANG-HCP-PROSTATE-CANCER	Prostate cancer	hsa05215	82
WANG-PROSTATE-CANCER-ANDROGEN-INDEPENDENT	Prostate cancer	hsa05215	72
WINNEPENNINCKX-MELANOMA-METASTASIS-UP WINNEPENNINCKX-MELANOMA-METASTASIS-DN	Melanoma	hsa05218	207
WONG-ENDOMETRIAL-CANCER-LATE	Endometrial cancer	hsa05213	9
WU-APOPTOSIS-BY-CDKN1A-NOT-VIA-TP53	Apoptosis	hsa04210	11
WU-APOPTOSIS-BY-CDKN1A-VIA-TP53	Apoptosis	hsa04210	37

WU-SILENCED-BY-METHYLATION-IN-BLADDER-CANCER	Bladder cancer	hsa05219	44
YEGNASUBRAMANIAN-PROSTATE-CANCER	Prostate cancer	hsa05215	158

Table A.1: MSigDB gold standard data. The first column is the name of the MSigDB gene set used, *up* and *down* regulated gene sets of the same condition were joined together in one gene set. The target pathway and the KEGG identifier are shown in the second and third column respectively. The fourth column, n, is the number of genes in the, optionally combined, gene set after translating to Ensembl IDs. In the digital version of this thesis, all MSigDB gene set names and KEGG identifiers are clickable hyperlinks.

Part B: Additional results on clustering combined with pathway analysis

There are many parameters that can be changed in the benchmark presented in this thesis. There are two clustering methods and two ways of estimating the FPR for example. There is also the alternative gold standard data based on MSigDB gene sets. Finally there are different tools that can be used in combination with clustering. This leads to dozens of possible ways to run the benchmark. To keep the main text to the point, not all combinations I examined are included there. Here I present a few other results from the benchmark using different settings.

B.1 First benchmark: microarray data

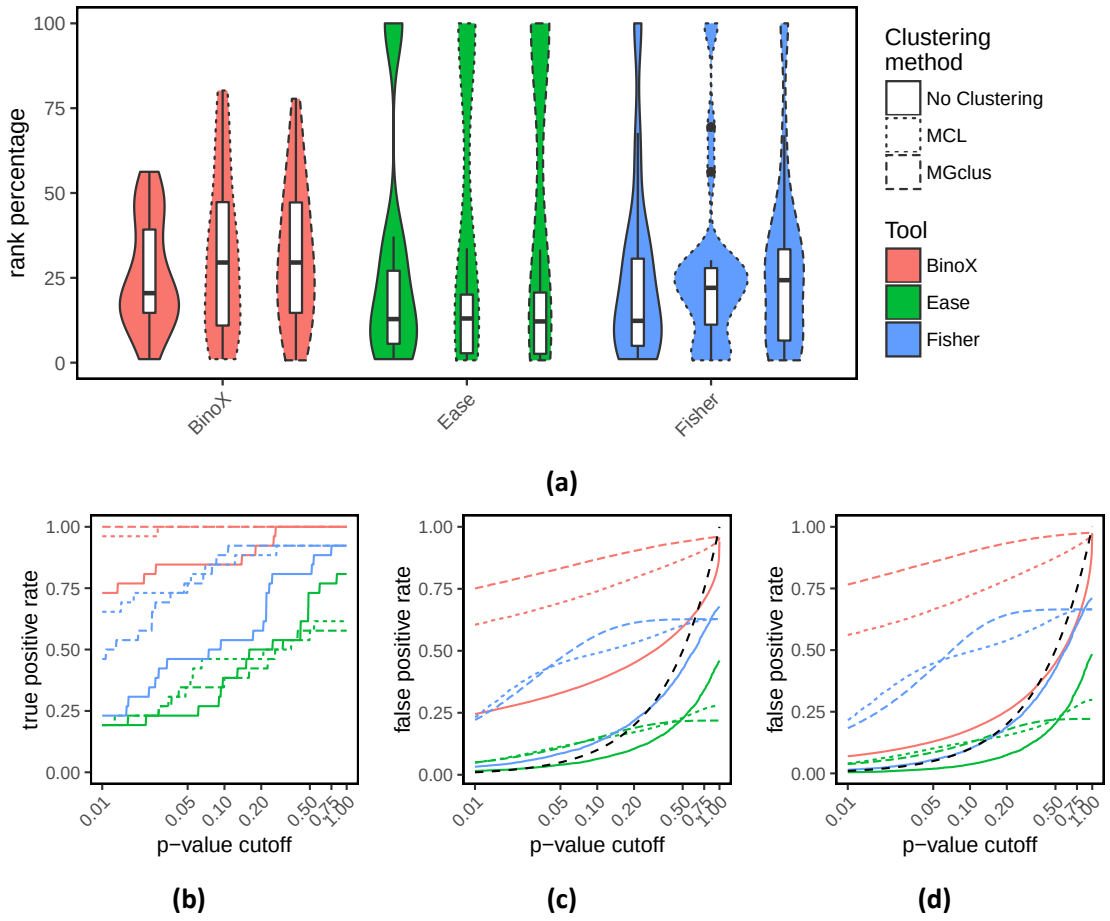
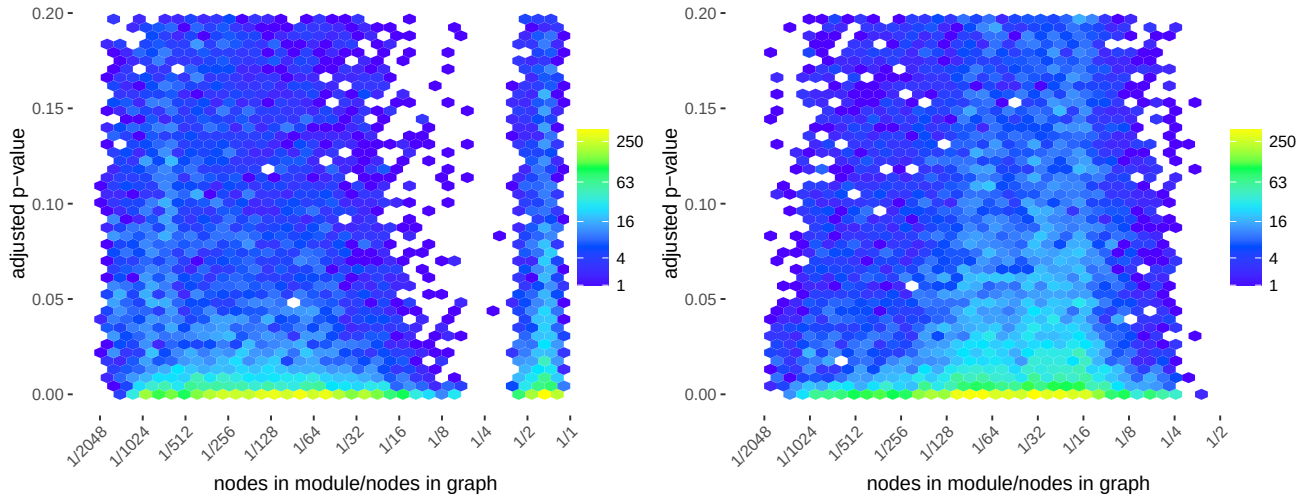
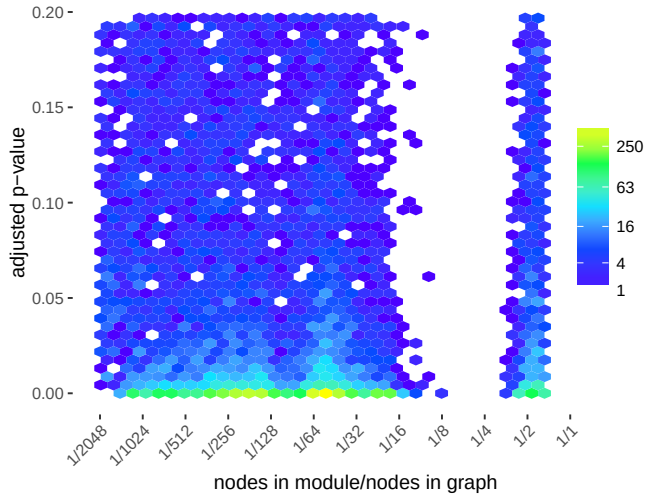


Figure B.1: Performance of pathway analysis combined with clustering on smaller gene sets. The method used for extracting a gene set from microarray data has a huge impact on pathway analysis methods that take a gene set as input. This effect is illustrated here by extracting gene sets using the same cutoffs as in fig. 3.3 but limiting the maximum gene set size to 600, which is a little over the largest KEGG pathway size (olfactory transduction, hsa04740). Shown are (**a**) the rank percentages of the target pathways, (**b**) the true positive rates and (**c**), (**d**), the false positive rates using label swap and gene permutation respectively. Additional figures using q-values are given in fig. B.8.

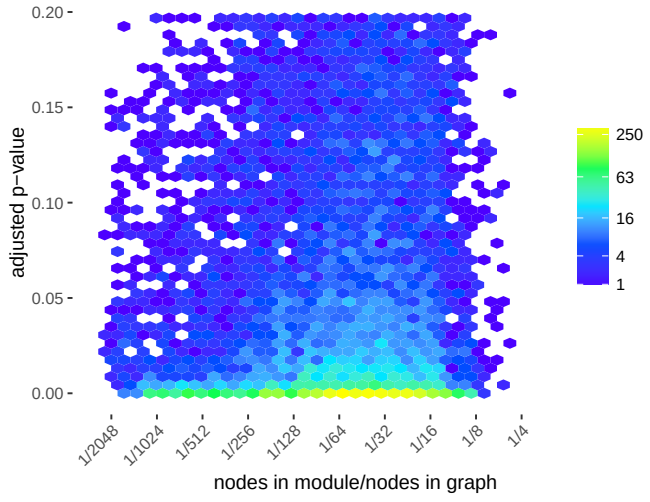


(a) MCL, label swap

(b) MGclus, label swap



(c) MCL, gene set permutation



(d) MGclus, gene set permutation

Figure B.2: Distribution of false positives for the EASE tool. Randomized modules were generated as explained in fig. 3.5 and all module to KEGG pathway combinations were analysed using the EASE score. Shown here are the BH adjusted p-values for enrichment that are below 0.20. Axes and colors are as in fig. 3.5.

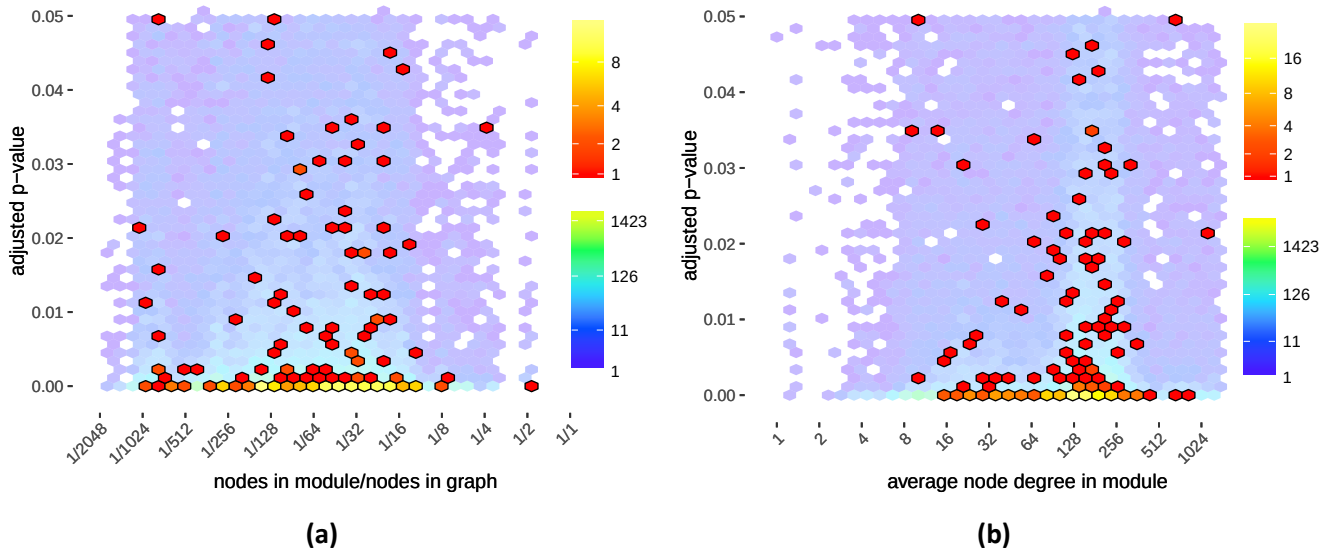


Figure B.3: Distribution of true positives for the BinoX tool using MGclus. In the main text, I only show this distribution for BinoX combined with MCL and argue that module size and average node degree are not informative features that set target pathways apart from other pathways. The same seems to be true when using MGclus instead of MCL. Axes and colors in (**a**) and (**b**) are as in fig. 3.6.

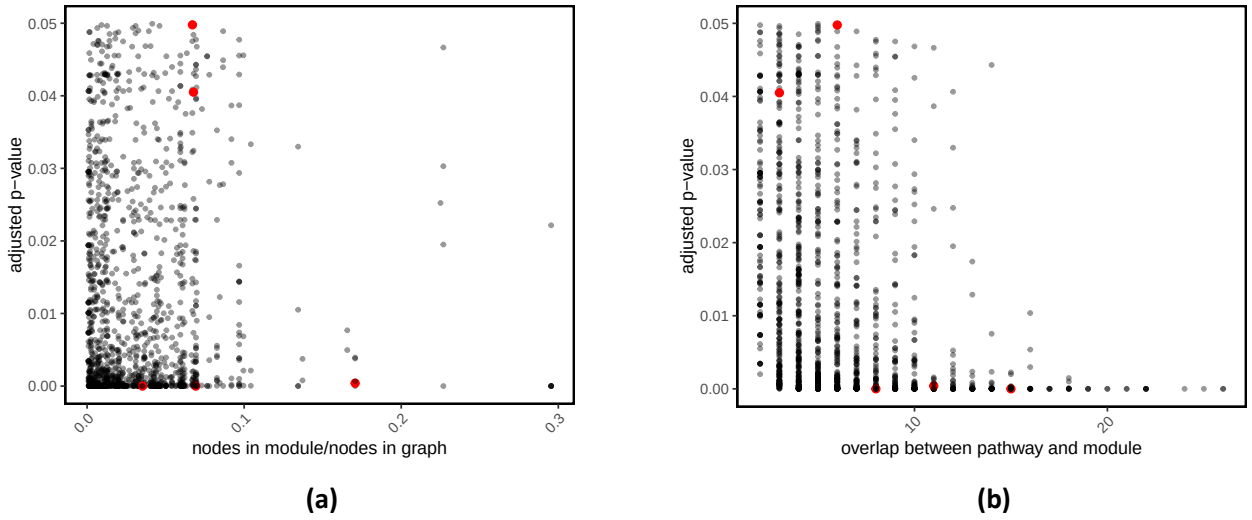


Figure B.4: Distribution of true positives for the Ease tool using MGclus. The analysis from fig. 3.7 was repeated but using MGclus instead. Due to sparsity of the data, two dimensional binning is not very well suited to visualise the data. The data is now displayed as a scatterplot instead, but the meaning of the axes have not changed from fig. 3.7. Big red dots represent a module to target pathway combination, all other dots are module to non target pathway combinations.

B.2 Second benchmark: MSigDB data

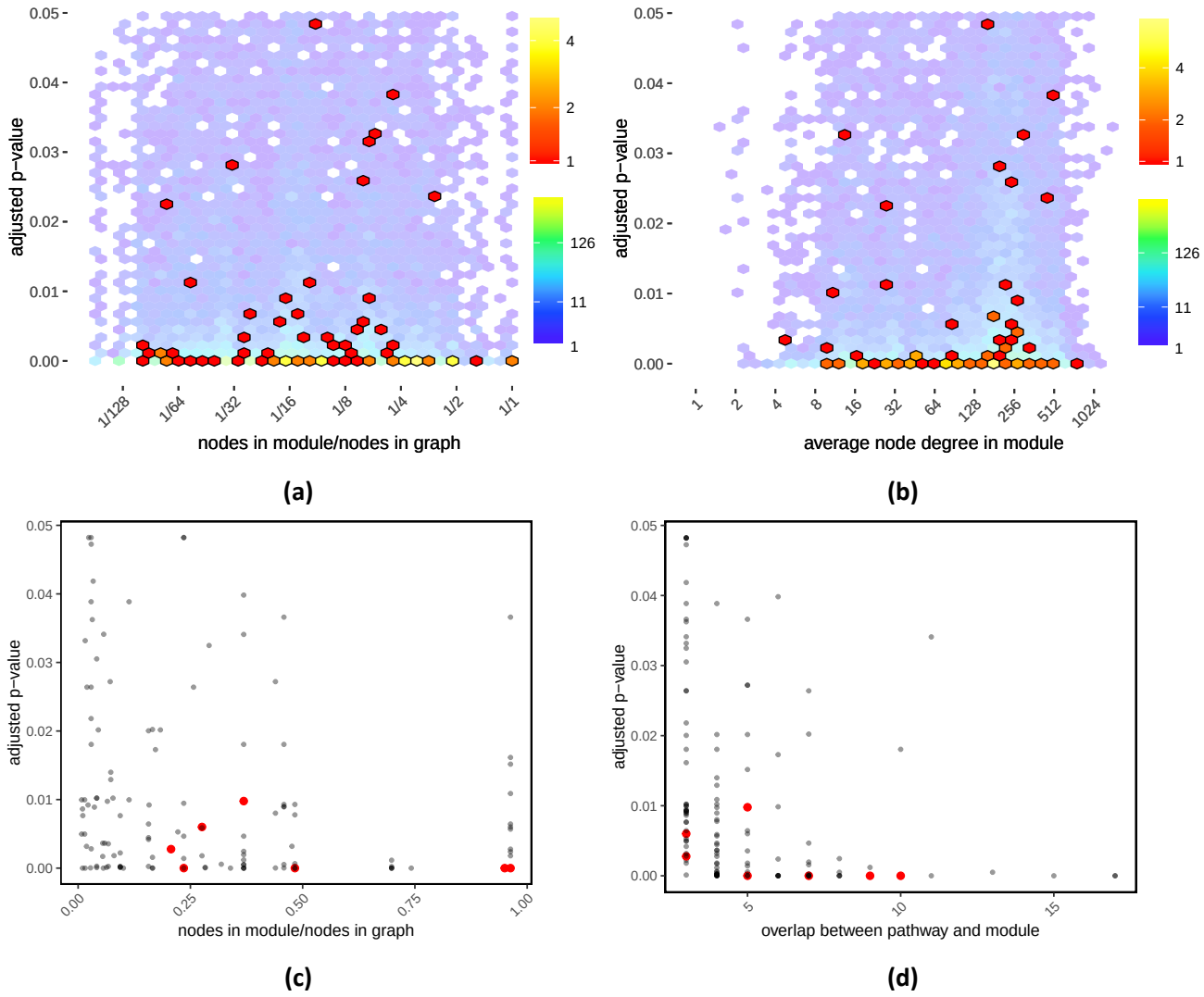


Figure B.5: Distribution of true positives for MSigDB gene sets using MGclus. The MSigDB data from table A.1 that was analysed in fig. 3.9 is here reanalysed using MGclus instead of MCL. Axes and colors are again as in fig. 3.9. BinoX output is shown in (**a**) and (**b**) and EASE output is shown in (**c**) and (**d**).

B.3 Sensitivity and specificity tests using q-values

In the main text, sensitivity and specificity plots have been given using p-values instead of q-values (= p-values controlled with BH procedure). The reason for this is that for some tools there are many ties created at either 0 or 1 by adjusting the p-values. These ties make it more difficult to distinguish the lines on the plots. But it is common practice to control the p-value for multiple testing. For completeness, here are the same plots repeated but the sensitivity/specificity is given over a range of q-values instead of p-values. The results are always corrected across all pathways and datasets at once. When clustering is used, the correction is done across all pathways and modules at once. Every figure below is linked to an equivalent figure somewhere else in the text that uses p-values. Follow these links for the figure description.

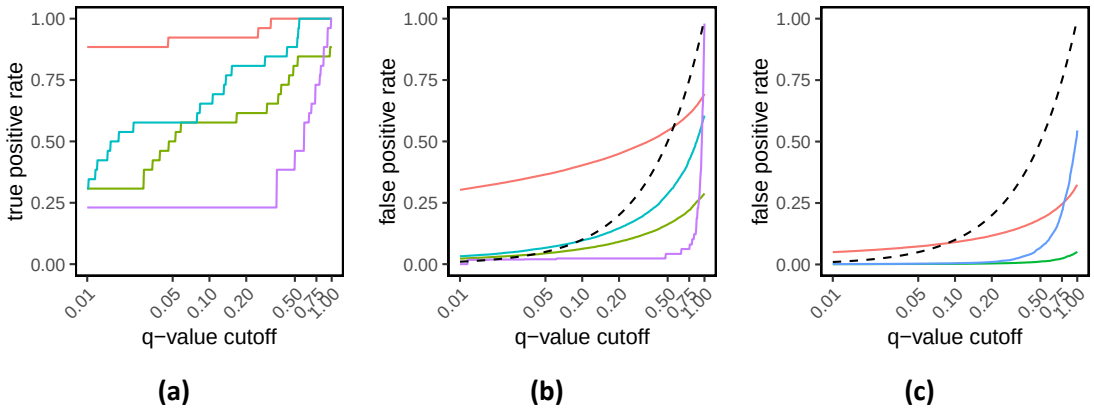


Figure B.6: Comparing BinoX with earlier tools (q-values). The same data as presented in fig. 3.2 is presented here but using q-values as a significance cutoff instead of p-values.

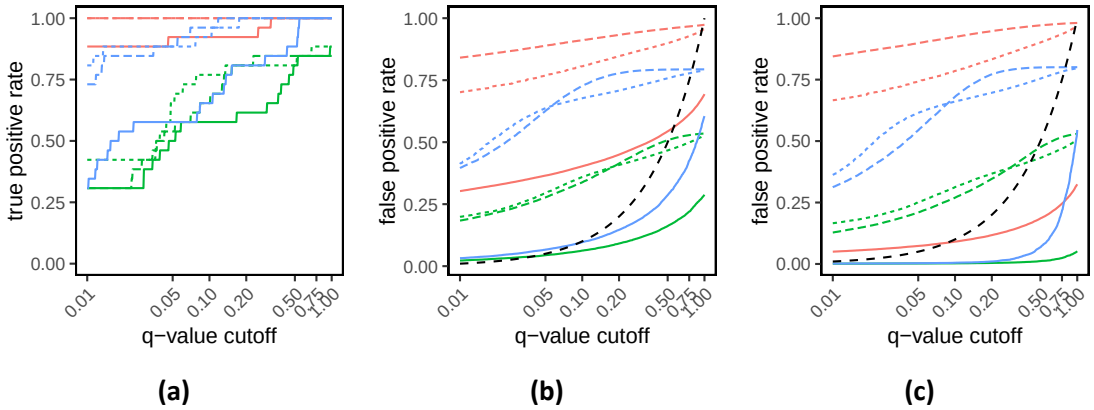


Figure B.7: Assessing the effect of clustering (q-values). The same data as presented in fig. 3.3 is presented here but using q-values as a significance cutoff instead of p-values.

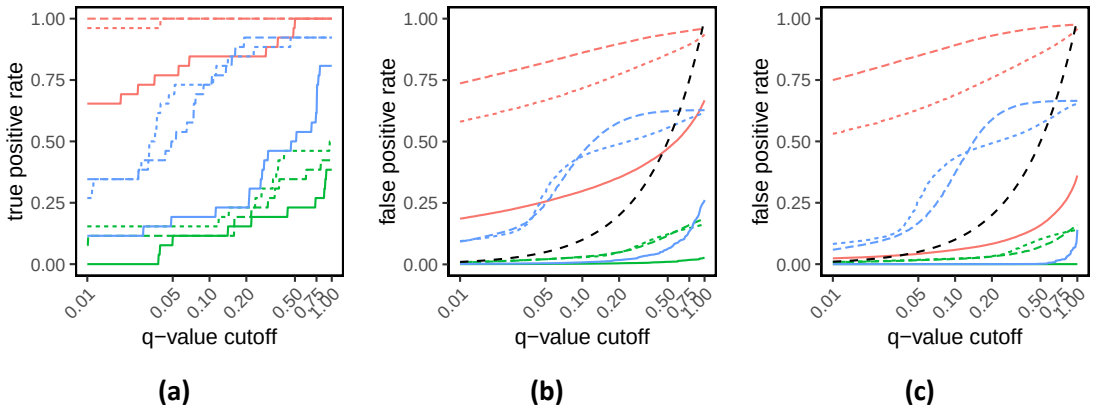


Figure B.8: Assessing the effect of clustering on smaller gene sets (q-values). The same data as presented in fig. B.1 is presented here but using q-values as a significance cutoff instead of p-values.

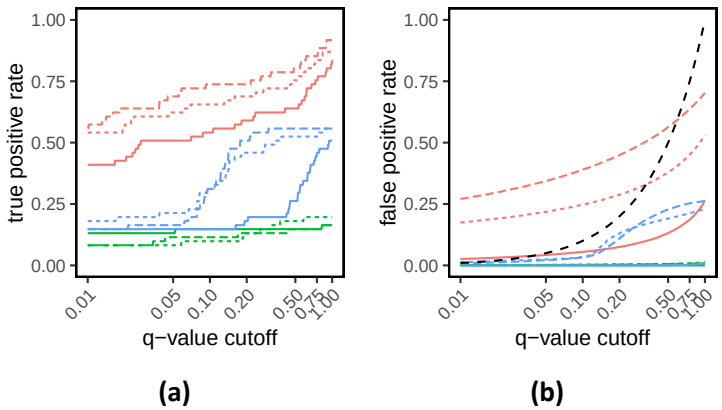


Figure B.9: Assessing the effect of clustering on MSigDB data (q-values). The same data as presented in fig. 3.8 is presented here but using q-values as a significance cutoff instead of p-values.

B.4 Module counts for target pathways and non target pathways

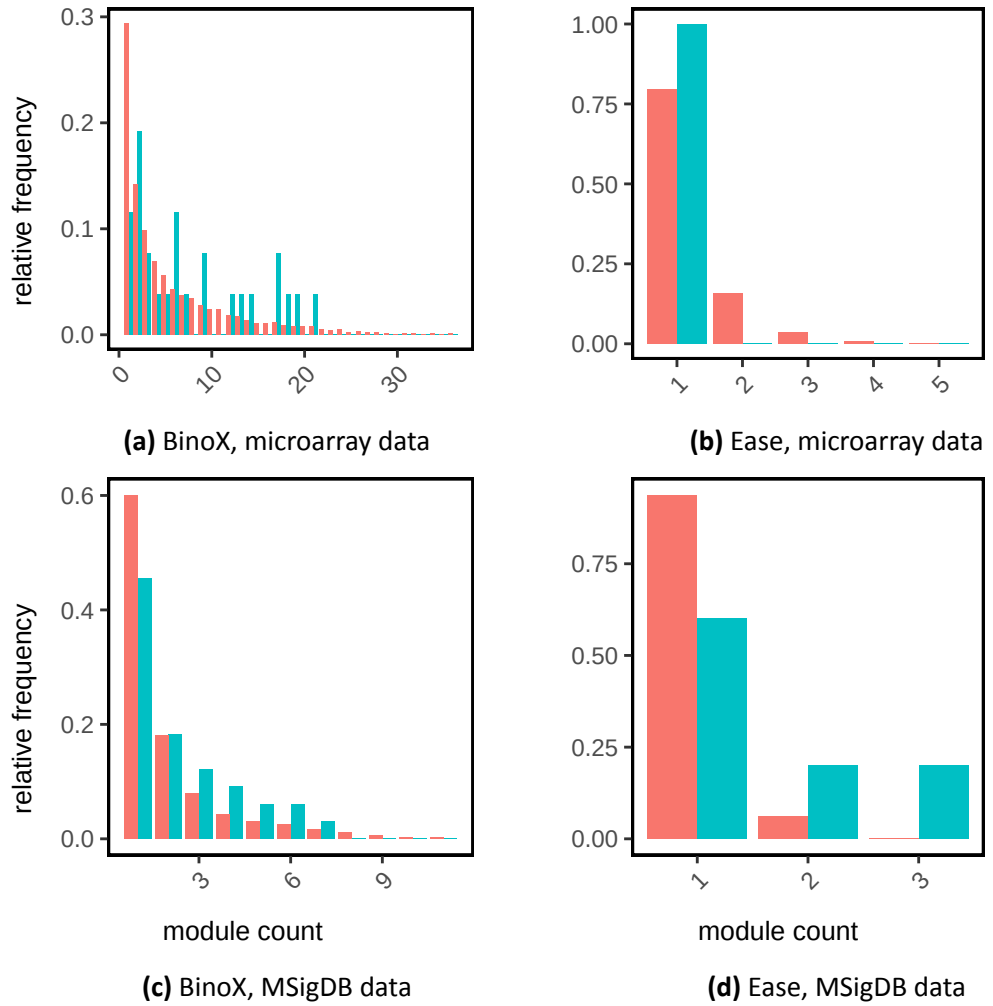
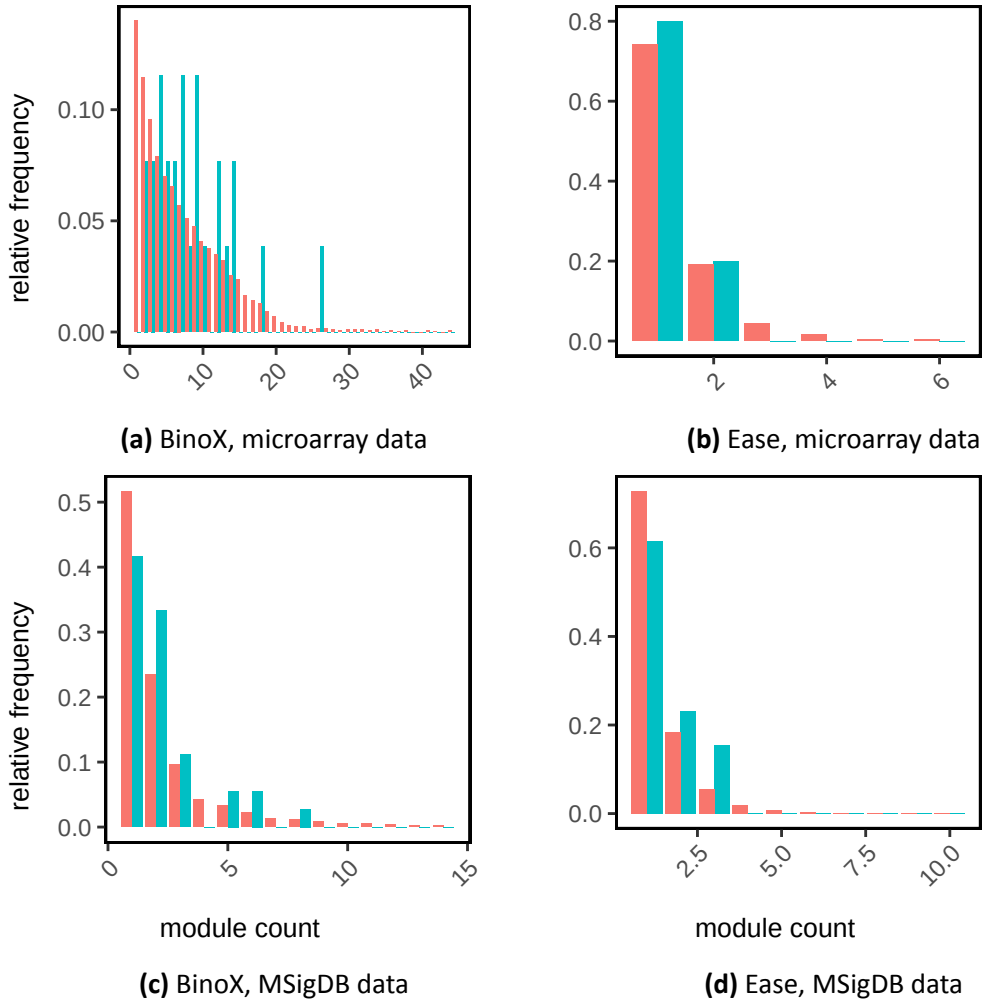


Figure B.10: Distribution of the number of significantly enriched modules using MCL. After running the benchmark on both the microarray and MSigDB data, the amount of modules made by MCL that show a significant enrichment to a pathway was counted. In plot (**a**) for example, the gene sets obtained from the microarray data (section 3.1.2) were clustered using MCL and then BinoX was run on each module versus all KEGG pathways. Shown in blue is the distribution of the number of modules that show significant enrichment for a gene set versus target pathway combination. In pink, the same distribution is shown for gene set versus non target pathway combinations. Take plot (**b**) for example: for every query gene set there was only one module that was significantly enriched towards the target pathway. Since for all the query gene sets there was only 1 significant module, the bar at 1 is set to 100% (= 1.0). And in plot (**c**) for example, about 45% of the gene sets had only one module that was enriched to the target pathway, hence the blue bar at 1 is set to $\pm 45\%$. Modules in plots (**a**) and (**c**) were counted as "significant" if the BH adjusted q-value from BinoX was at least 0.05. For plots (**b**) and (**d**), modules were "significant" if the q-value from EASE was at least 0.10.



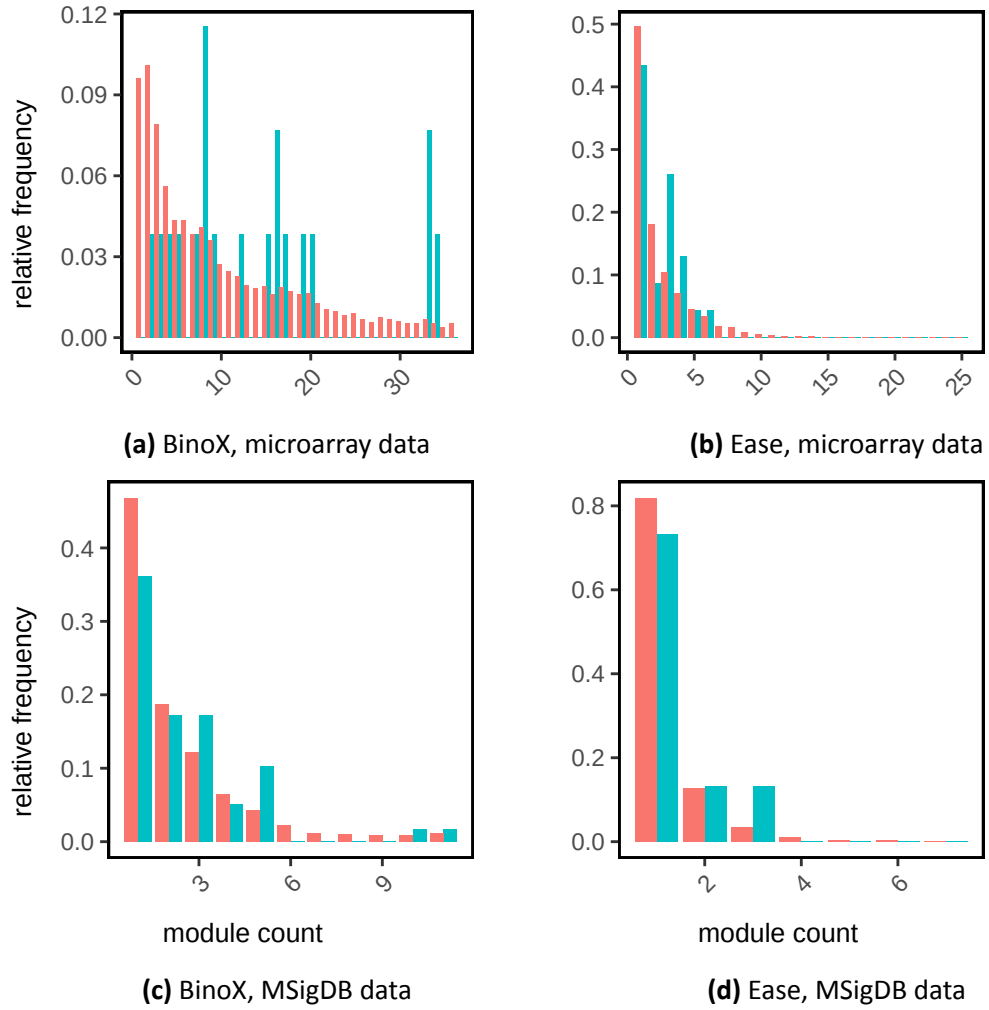
(a) BinoX, microarray data

(b) Ease, microarray data

(c) BinoX, MSigDB data

(d) Ease, MSigDB data

Figure B.11: Distribution of the number of significantly enriched modules using MGclus. The gene sets from both benchmarks (microarray data and MSigDB data) were clustered using MGclus and the number of significant modules was counted. Module to target pathway combinations are shown in blue, all other combination are shown in pink. Modules in (**a**) and (**c**) were counted as "significant" if the q-value from BinoX was at least 0.05. Modules in (**b**) and (**d**) were counted as "significant" if the q-value from Ease was at least 0.1. See fig. B.10 for a more in-depth explanation.



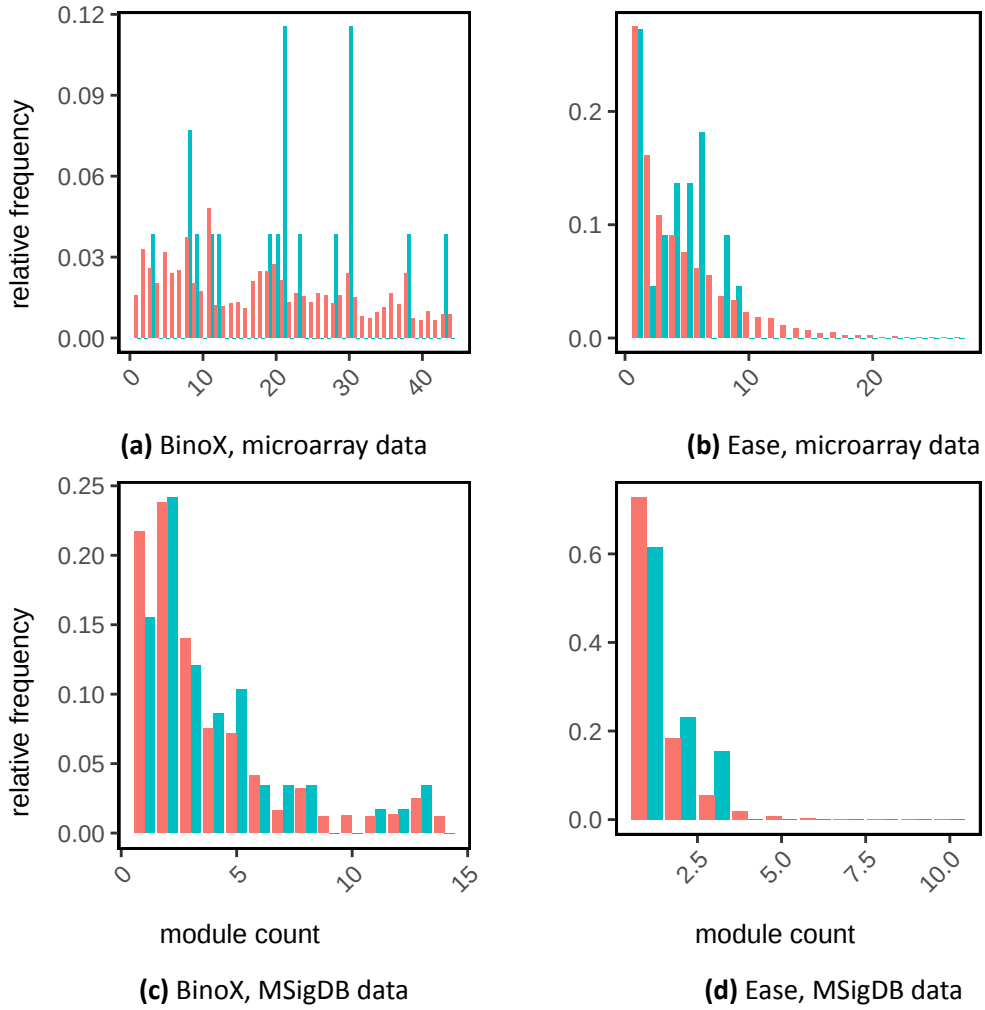
(a) BinoX, microarray data

(b) Ease, microarray data

(c) BinoX, MSigDB data

(d) Ease, MSigDB data

Figure B.12: Distribution of the number of modules with a p-value less than 1 using MCL. The gene sets from both benchmarks (microarray data and MSigDB data) were clustered using MCL and the number of modules that have an unadjusted p-value of less than one to a pathway were counted for BinoX in (**a**) and (**c**) and for EASE in (**b**) and (**d**). In the case of BinoX, the p-value for enrichment between a pathway and a module is less than 1 if there is at least one link between the module and the pathway. If there are no links, then the probability of seeing at least as many links under null hypothesis is equal to 1. In the case of EASE the p-value is less than 1 if the overlap between a module and a pathway is at least two genes. If the overlap is only one gene than $k-1$ in eq. 1.4 becomes zero, and the probability of seeing an overlap as extreme as zero is equal to 1. If there is no overlap, then $k-1$ will become -1 and the p-value is undefined. Module to target pathway combinations are shown in blue, all other combinations are shown in pink. Axes are explained in detail in fig. B.10.



(a) BinoX, microarray data

(b) Ease, microarray data

(c) BinoX, MSigDB data

(d) Ease, MSigDB data

Figure B.13: Distribution of the number of modules with a p-value less than 1 using MGclus. The gene sets from both benchmarks (microarray data and MSigDB data) were clustered using MGclus and the number of modules that have an unadjusted p-value of less than one to a pathway were counted for BinoX in (**a**) and (**c**) and for EASE in (**b**) and (**d**). Module to target pathway combinations are shown in blue, all other combinations are shown in pink. For a more thorough explanation, see fig. B.12. Axes are explained in detail in fig. B.10.

Part C: Human Proteome Atlas

In section 3.4 a few tissue specific gene sets from the HPA were briefly explored to see if pathway analysis combined with clustering could pick up any important pathways that were not found without clustering. The data for constructing fig. 3.10 is shown here.

In the tables below, a pathway is given if it is found significant by at least one of the tools. A pathway is considered significant if it has a BH adjusted p-value of at least 0.05. These q-values are given in the last four columns; q-values below 0.05 have a gray background and q-values equal to 1 have been omitted. The pathways are grouped based on which tools found them significant. Within each group the pathways are sorted in alphabetical order. In the digital version of this thesis, the KEGG identifiers are hyperlinks.

C.1 Adipose tissue

pathway name	KEGG ID	BinoX	BinoX + MGclus	EASE	EASE + MGclus
PPAR signaling pathway	hsa03320	0.038	3.59×10^{-03}	7.77×10^{-03}	
Neuroactive ligand-receptor interaction	hsa04080	1.89×10^{-03}	1.47×10^{-03}	0.958	0.012
ABC transporters	hsa02010	5.99×10^{-05}	5.06×10^{-07}		
Adipocytokine signaling pathway	hsa04920	3.79×10^{-05}	2.95×10^{-04}		
Amino sugar and nucleotide sugar metabolism	hsa00520	0.030	2.44×10^{-03}		
Arginine and proline metabolism	hsa00330	0.021	0.048		
Biosynthesis of amino acids	hsa01230	1.02×10^{-05}	4.87×10^{-05}		
Carbon metabolism	hsa01200	3.56×10^{-10}	2.11×10^{-07}		
Citrate cycle (TCA cycle)	hsa00020	1.04×10^{-04}	7.64×10^{-04}		
Cytokine-cytokine receptor interaction	hsa04060	1.04×10^{-05}	0.041	0.110	
Fatty acid biosynthesis	hsa00061	6.38×10^{-06}	8.04×10^{-07}		
Fatty acid degradation	hsa00071	2.53×10^{-08}	1.37×10^{-04}		
Fatty acid metabolism	hsa01212	5.05×10^{-07}	1.37×10^{-04}		
Glycerolipid metabolism	hsa00561	9.98×10^{-04}	1.74×10^{-03}	0.219	
Glycolysis / Gluconeogenesis	hsa00010	1.09×10^{-07}	8.82×10^{-06}		

Maturity onset diabetes of the young	hsa04950	7.06×10^{-04}	0.031
Nicotine addiction	hsa05033	0.020	0.045
Olfactory transduction	hsa04740	9.70×10^{-15}	2.53×10^{-57}
Pentose and glucuronate interconversions	hsa00040	0.016	0.021
Peroxisome	hsa04146	1.71×10^{-10}	8.59×10^{-16}
Pyruvate metabolism	hsa00620	2.82×10^{-06}	1.02×10^{-04}
Spliceosome	hsa03040	1.41×10^{-06}	0.048
Starch and sucrose metabolism	hsa00500	6.01×10^{-04}	4.73×10^{-04}
Sulfur metabolism	hsa00920	0.035	0.013
Synaptic vesicle cycle	hsa04721	8.42×10^{-03}	0.031
Ascorbate and aldarate metabolism	hsa00053	0.028	0.098
Biosynthesis of unsaturated fatty acids	hsa01040	0.033	
Cell cycle	hsa04110	3.63×10^{-03}	
Collecting duct acid secretion	hsa04966	0.049	0.399
Complement and coagulation cascades	hsa04610	6.71×10^{-04}	0.947
Cysteine and methionine metabolism	hsa00270	0.025	0.354
DNA replication	hsa03030	4.40×10^{-03}	
Focal adhesion	hsa04510	0.048	
Fructose and mannose metabolism	hsa00051	0.015	0.212
Glyoxylate and dicarboxylate metabolism	hsa00630	1.93×10^{-03}	0.153
Histidine metabolism	hsa00340	0.015	0.087
Lysine degradation	hsa00310	0.011	0.160
Mismatch repair	hsa03430	9.52×10^{-03}	
Nucleotide excision repair	hsa03420	8.17×10^{-03}	
Pentose phosphate pathway	hsa00030	0.033	0.146
Platelet activation	hsa04611	0.020	
Propanoate metabolism	hsa00640	0.030	0.538
Pyrimidine metabolism	hsa00240	7.94×10^{-03}	
Regulation of actin cytoskeleton	hsa04810	0.025	
Ribosome biogenesis in eukaryotes	hsa03008	0.030	
Ribosome	hsa03010	1.06×10^{-03}	
Tryptophan metabolism	hsa00380	0.013	0.219
Type II diabetes mellitus	hsa04930	0.038	0.225
Valine, leucine and isoleucine degradation	hsa00280	0.045	
Vitamin digestion and absorption	hsa04977	0.018	0.056
Wnt signaling pathway	hsa04310	0.020	

Alzheimer's disease	hsa05010	0.194	0.041		
Bile secretion	hsa04976	0.221	0.021		
Calcium signaling pathway	hsa04020	0.052	7.60×10^{-04}		0.296
cGMP-PKG signaling pathway	hsa04022	0.226	2.78×10^{-03}		
Endocytosis	hsa04144	0.078	0.012		
Ether lipid metabolism	hsa00565	0.199	1.09×10^{-03}		
Jak-STAT signaling pathway	hsa04630	0.175	0.033	0.592	
N-Glycan biosynthesis	hsa00510	0.498	0.011		
Oxidative phosphorylation	hsa00190	0.107	0.040		
Parkinson's disease	hsa05012	0.477	0.035		
Salivary secretion	hsa04970	0.421	0.037		
AMPK signaling pathway	hsa04152	0.158	0.121	0.014	

Table C.1: Pathway analysis of the HPA adipose tissue genes. Data used for constructing fig. 3.10a. See also the introduction of appendix C and section 3.4.

C.2 Lung tissue

pathway name	KEGG ID	BinoX	BinoX + MGclus	EASE	EASE + MGclus
Focal adhesion	hsa04510	4.31×10^{-05}	7.46×10^{-19}	0.062	0.020
ABC transporters	hsa02010	5.09×10^{-06}	2.14×10^{-05}		
Adherens junction	hsa04520	5.33×10^{-04}	1.62×10^{-09}		
Adipocytokine signaling pathway	hsa04920	0.018	3.42×10^{-05}		
Aldosterone-regulated sodium reabsorption	hsa04960	3.12×10^{-03}	4.26×10^{-04}		
Allograft rejection	hsa05330	4.68×10^{-09}	7.69×10^{-16}		
Antigen processing and presentation	hsa04612	8.80×10^{-09}	1.98×10^{-16}		
Arrhythmogenic right ventricular cardiomyopathy (ARVC)	hsa05412	0.022	5.86×10^{-04}		
Asthma	hsa05310	2.38×10^{-06}	6.19×10^{-11}		
Autoimmune thyroid disease	hsa05320	3.86×10^{-09}	7.67×10^{-16}		
Axon guidance	hsa04360	3.68×10^{-06}	4.34×10^{-12}		
Bacterial invasion of epithelial cells	hsa05100	5.56×10^{-03}	9.66×10^{-11}		
Bile secretion	hsa04976	5.37×10^{-05}	0.012	0.505	
Bladder cancer	hsa05219	0.016	5.32×10^{-04}		
cAMP signaling pathway	hsa04024	1.51×10^{-03}	9.37×10^{-05}		
Cell adhesion molecules (CAMs)	hsa04514	8.62×10^{-16}	1.66×10^{-15}		
Chemical carcinogenesis	hsa05204	0.032	2.09×10^{-05}		
Chemokine signaling pathway	hsa04062	2.95×10^{-05}	3.15×10^{-09}		
Collecting duct acid secretion	hsa04966	8.11×10^{-03}	4.17×10^{-04}		
Cytokine-cytokine receptor interaction	hsa04060	5.32×10^{-05}	3.65×10^{-10}		
Dorso-ventral axis formation	hsa04320	0.025	0.016		
Drug metabolism	hsa00982	0.029	2.82×10^{-07}		
ECM-receptor interaction	hsa04512	6.38×10^{-07}	7.28×10^{-06}	0.736	
Endocrine and other factor-regulated calcium reabsorption	hsa04961	3.88×10^{-03}	1.62×10^{-03}		
Endocytosis	hsa04144	1.04×10^{-09}	2.37×10^{-09}		
Epithelial cell signaling in Helicobacter pylori infection	hsa05120	3.82×10^{-03}	6.11×10^{-03}		
Estrogen signaling pathway	hsa04915	0.020	3.16×10^{-03}		
Ether lipid metabolism	hsa00565	7.82×10^{-04}	6.13×10^{-04}		
Fatty acid biosynthesis	hsa00061	3.82×10^{-03}	1.16×10^{-06}		
Fc gamma R-mediated phagocytosis	hsa04666	0.016	8.40×10^{-03}		

GABAergic synapse	hsa04727	2.83×10^{-03}	4.17×10^{-04}	
Gap junction	hsa04540	0.015	5.28×10^{-05}	
Glioma	hsa05214	0.032	1.45×10^{-05}	
Glutamatergic synapse	hsa04724	6.41×10^{-05}	2.27×10^{-04}	
GnRH signaling pathway	hsa04912	1.84×10^{-03}	5.08×10^{-07}	
Graft-versus-host disease	hsa05332	1.09×10^{-09}	9.91×10^{-17}	
Hematopoietic cell lineage	hsa04640	1.67×10^{-04}	1.07×10^{-04}	
Herpes simplex infection	hsa05168	8.70×10^{-03}	1.11×10^{-07}	
Hippo signaling pathway	hsa04390	0.031	1.21×10^{-04}	
Inflammatory bowel disease (IBD)	hsa05321	9.93×10^{-06}	4.58×10^{-10}	
Intestinal immune network for IgA production	hsa04672	2.21×10^{-07}	6.64×10^{-12}	
Leishmaniasis	hsa05140	1.91×10^{-07}	4.25×10^{-08}	
Leukocyte transendothelial migration	hsa04670	7.29×10^{-05}	3.01×10^{-08}	
Long-term depression	hsa04730	4.84×10^{-05}	1.51×10^{-04}	
Lysosome	hsa04142	2.56×10^{-05}	6.81×10^{-03}	
MAPK signaling pathway	hsa04010	8.30×10^{-03}	6.43×10^{-07}	
Melanoma	hsa05218	4.06×10^{-03}	1.21×10^{-05}	
Metabolism of xenobiotics by cytochrome P450	hsa00980	8.36×10^{-03}	2.09×10^{-05}	
MicroRNAs in cancer	hsa05206	3.17×10^{-04}	2.94×10^{-11}	
Mineral absorption	hsa04978	2.18×10^{-03}	0.020	
Morphine addiction	hsa05032	1.20×10^{-03}	5.10×10^{-03}	
Natural killer cell mediated cytotoxicity	hsa04650	6.58×10^{-04}	3.52×10^{-03}	
Neuroactive ligand-receptor interaction	hsa04080	0.021	1.21×10^{-03}	
Nicotine addiction	hsa05033	0.030	1.33×10^{-03}	
Osteoclast differentiation	hsa04380	0.029	0.012	
Ovarian steroidogenesis	hsa04913	2.33×10^{-04}	7.34×10^{-04}	
Pancreatic cancer	hsa05212	2.67×10^{-03}	3.05×10^{-03}	
Pancreatic secretion	hsa04972	1.27×10^{-03}	3.12×10^{-05}	
Pathways in cancer	hsa05200	1.11×10^{-03}	1.29×10^{-10}	
Phagosome	hsa04145	2.97×10^{-14}	1.42×10^{-08}	0.257
PI3K-Akt signaling pathway	hsa04151	1.49×10^{-03}	2.14×10^{-12}	
Protein digestion and absorption	hsa04974	6.34×10^{-03}	6.53×10^{-03}	
Proteoglycans in cancer	hsa05205	3.42×10^{-05}	1.65×10^{-17}	
Proximal tubule bicarbonate reclamation	hsa04964	1.37×10^{-03}	0.021	
Rap1 signaling pathway	hsa04015	2.55×10^{-06}	3.01×10^{-16}	
Ras signaling pathway	hsa04014	8.76×10^{-05}	3.96×10^{-15}	
Regulation of actin cytoskeleton	hsa04810	7.73×10^{-03}	4.43×10^{-09}	

Retrograde endocannabinoid signaling	hsa04723	5.00×10^{-03}	5.71×10^{-05}
Rheumatoid arthritis	hsa05323	5.47×10^{-07}	1.20×10^{-06}
Serotonergic synapse	hsa04726	8.96×10^{-04}	1.70×10^{-03}
Signaling pathways regulating pluripotency of stem cells	hsa04550	7.93×10^{-03}	5.18×10^{-05}
Small cell lung cancer	hsa05222	0.017	0.013
Staphylococcus aureus infection	hsa05150	7.92×10^{-11}	5.04×10^{-11}
Steroid hormone biosynthesis	hsa00140	0.031	1.13×10^{-03}
Synaptic vesicle cycle	hsa04721	0.011	1.38×10^{-03}
Systemic lupus erythematosus	hsa05322	3.25×10^{-03}	2.05×10^{-06}
TGF-beta signaling pathway	hsa04350	3.24×10^{-03}	1.25×10^{-05}
Toxoplasmosis	hsa05145	2.18×10^{-06}	1.88×10^{-05}
Tuberculosis	hsa05152	1.48×10^{-05}	4.04×10^{-05}
Type I diabetes mellitus	hsa04940	2.13×10^{-07}	4.11×10^{-14}
Type II diabetes mellitus	hsa04930	0.021	0.012
VEGF signaling pathway	hsa04370	4.64×10^{-03}	3.19×10^{-08}
Viral myocarditis	hsa05416	1.14×10^{-06}	9.80×10^{-13}
Carbohydrate digestion and absorption	hsa04973	0.044	0.059
Cell cycle	hsa04110	1.46×10^{-04}	0.098
Chagas disease (American trypanosomiasis)	hsa05142	7.52×10^{-04}	0.141
Cocaine addiction	hsa05030	0.025	0.090
Complement and coagulation cascades	hsa04610	4.97×10^{-04}	0.071
Gastric acid secretion	hsa04971	4.58×10^{-03}	0.067
Huntington’s disease	hsa05016	0.013	0.220
Insulin secretion	hsa04911	0.023	0.164
mRNA surveillance pathway	hsa03015	6.50×10^{-03}	
Nucleotide excision repair	hsa03420	1.60×10^{-03}	
Olfactory transduction	hsa04740	1.25×10^{-04}	0.652
Oocyte meiosis	hsa04114	8.19×10^{-03}	0.951
Pertussis	hsa05133	2.09×10^{-04}	0.093
Proteasome	hsa03050	7.87×10^{-04}	
Pyrimidine metabolism	hsa00240	9.83×10^{-03}	
Ribosome	hsa03010	5.46×10^{-10}	0.136
RNA degradation	hsa03018	0.036	
RNA transport	hsa03013	7.32×10^{-05}	
Salivary secretion	hsa04970	8.75×10^{-03}	0.127
Spliceosome	hsa03040	2.73×10^{-07}	0.683
Thyroid hormone synthesis	hsa04918	0.016	0.074
Ascorbate and aldarate metabolism	hsa00053	0.076	4.00×10^{-06}

B cell receptor signaling pathway	hsa04662	0.343	7.03×10^{-03}	
Calcium signaling pathway	hsa04020	0.148	6.75×10^{-05}	
cGMP-PKG signaling pathway	hsa04022	0.055	0.025	
Chronic myeloid leukemia	hsa05220	0.357	3.74×10^{-03}	
Circadian entrainment	hsa04713	0.135	0.047	
Dilated cardiomyopathy	hsa05414	0.067	0.035	0.781
Dopaminergic synapse	hsa04728	0.211	5.30×10^{-03}	
Endometrial cancer	hsa05213	0.071	8.67×10^{-06}	
ErbB signaling pathway	hsa04012	0.177	1.54×10^{-05}	
Fatty acid degradation	hsa00071	0.260	9.78×10^{-05}	
Fatty acid metabolism	hsa01212	0.296	1.14×10^{-04}	
Fc epsilon RI signaling pathway	hsa04664	0.119	1.92×10^{-03}	
FoxO signaling pathway	hsa04068	0.148	3.28×10^{-04}	
Glycerophospholipid metabolism	hsa00564	0.111	0.032	
Glycolysis / Gluconeogenesis	hsa00010	0.141	0.016	
HIF-1 signaling pathway	hsa04066	0.268	0.022	
HTLV-I infection	hsa05166	0.071	2.31×10^{-04}	
Hypertrophic cardiomyopathy (HCM)	hsa05410	0.086	0.044	0.669
Influenza A	hsa05164	0.214	9.55×10^{-04}	
Inositol phosphate metabolism	hsa00562	0.080	0.031	
Insulin signaling pathway	hsa04910	0.070	1.04×10^{-04}	
Neurotrophin signaling pathway	hsa04722	0.369	0.046	
N-Glycan biosynthesis	hsa00510	0.691	6.43×10^{-03}	
Non-small cell lung cancer	hsa05223	0.086	3.12×10^{-03}	
Notch signaling pathway	hsa04330	0.456	0.029	
Oxytocin signaling pathway	hsa04921	0.060	3.04×10^{-03}	
Pentose and glucuronate interconversions	hsa00040	0.065	8.15×10^{-05}	
Peroxisome	hsa04146	0.382	6.02×10^{-05}	
Platelet activation	hsa04611	0.066	6.84×10^{-04}	
Porphyrin and chlorophyll metabolism	hsa00860	0.205	3.25×10^{-04}	
PPAR signaling pathway	hsa03320	0.423	2.64×10^{-04}	
Primary immunodeficiency	hsa05340	0.403	0.033	
Prolactin signaling pathway	hsa04917	0.413	0.043	
Prostate cancer	hsa05215	0.198	5.04×10^{-04}	
Renin-angiotensin system	hsa04614	0.107	0.048	
Retinol metabolism	hsa00830	0.110	4.73×10^{-05}	
Starch and sucrose metabolism	hsa00500	0.172	7.94×10^{-04}	

Thyroid hormone signaling pathway	hsa04919	0.111	0.025
Tight junction	hsa04530	0.141	0.039
Toll-like receptor signaling pathway	hsa04620	0.382	0.025
Transcriptional misregulation in cancer	hsa05202	0.325	2.59×10^{-03}
Vascular smooth muscle contraction	hsa04270	0.151	3.39×10^{-05}
Vibrio cholerae infection	hsa05110	0.141	0.012
Wnt signaling pathway	hsa04310	0.381	0.027

Table C.2: Pathway analysis of the HPA lung tissue genes. Data used for constructing fig. 3.10b. See also the introduction of appendix C and section 3.4.

C.3 Pancreas tissue

pathway name	KEGG ID	BinoX	BinoX + MGclus	EASE	EASE + MGclus
Insulin secretion	hsa04911	0.022	2.58×10^{-04}	0.021	0.181
Olfactory transduction	hsa04740	1.42×10^{-23}	4.12×10^{-43}		
Fat digestion and absorption	hsa04975	6.22×10^{-03}		1.79×10^{-03}	0.025
Maturity onset diabetes of the young	hsa04950	0.022	0.350	7.75×10^{-06}	
Neuroactive ligand-receptor interaction	hsa04080	5.32×10^{-03}	0.346	0.039	
Starch and sucrose metabolism	hsa00500	2.28×10^{-04}	0.513	0.059	0.024
Adipocytokine signaling pathway	hsa04920	2.77×10^{-03}	0.349		
Amino sugar and nucleotide sugar metabolism	hsa00520	0.032	0.584		
AMPK signaling pathway	hsa04152	2.52×10^{-05}			
Biosynthesis of amino acids	hsa01230	1.22×10^{-08}			
Carbon metabolism	hsa01200	6.91×10^{-10}			
Circadian rhythm	hsa04710	1.62×10^{-05}			
FoxO signaling pathway	hsa04068	0.020			
Fructose and mannose metabolism	hsa00051	8.93×10^{-03}			
Galactose metabolism	hsa00052	3.39×10^{-05}	0.350		
Glycine, serine and threonine metabolism	hsa00260	0.027		0.313	
Glycolysis / Gluconeogenesis	hsa00010	5.29×10^{-10}	0.895		
HIF-1 signaling pathway	hsa04066	5.16×10^{-03}			
Hypertrophic cardiomyopathy (HCM)	hsa05410	1.39×10^{-03}	0.511		
Insulin signaling pathway	hsa04910	2.36×10^{-04}			
Oxytocin signaling pathway	hsa04921	6.17×10^{-03}			
Pantothenate and CoA biosynthesis	hsa00770	1.78×10^{-04}			
Pentose phosphate pathway	hsa00030	4.40×10^{-05}			
Porphyrin and chlorophyll metabolism	hsa00860	0.010			
Pyruvate metabolism	hsa00620	4.05×10^{-03}			
Rheumatoid arthritis	hsa05323	0.045			
Ribosome	hsa03010	4.30×10^{-04}			
Synaptic vesicle cycle	hsa04721	0.011	0.718		
Carbohydrate digestion and absorption	hsa04973			0.035	0.017
Pancreatic secretion	hsa04972	0.071	0.146	3.10×10^{-22}	3.07×10^{-09}
Protein digestion and absorption	hsa04974	0.534		1.06×10^{-09}	4.50×10^{-07}

Table C.3: Pathway analysis of the HPA pancreas tissue genes. Data used for constructing fig. 3.10c. See also the introduction of appendix C and section 3.4.