

BACHELOR PROJECT ASSIGNMENT

Student: Hynek Urban
Study programme: Electrical Engineering and Information Technology
Specialisation: Cybernetics and Measurement
Title of Bachelor Project: Knowledge-Oriented Genomic Probeset Consolidation

Guidelines:

1. Get familiar with the gene expression data and microarray technology.
2. Study the available resources of background knowledge for microarray data, focus on the sequential databases.
3. Study the principal ways in which the probesets can be consolidated.
4. Pick one of the methods learnt in the step 3 and implement it.
5. Statistically evaluate fitness of the representation that you got in the step 4.
6. Use the representation to classify several of gene expression datasets.

Bibliography/Sources:

- [1] Baxeavanis, Ouellette: Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins. 3rd Edition, Wiley, 2005.
- [2] Yu, Hui; Wang, Feng; Tu, Kang; Xie, Lu; Li, Yuan-Yuan and Li, Yi-Xue: Transcript-level Annotation of Affymetrix Probesets Improves the Interpretation of Gene Expression Data. BMC Bioinformatics, Vol. 8 (11 June 2007), 194.

Bachelor Project Supervisor: Ing. Jiří Kléma, Ph.D.

Valid until: the end of the winter semester of academic year 2009/2010


prof. Ing. Vladimír Mařík, DrSc.
Head of Department




doc. Ing. Boris Šimák, CSc.
Head

Prague, February 23, 2009

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Hynek Urban
Studijní program: Elektrotechnika a informatika (bakalářský), strukturovaný
Obor: Kybernetika a měření
Název tématu: Znalostní sdružování množin sond v datech genové exprese

Pokyny pro vypracování:

1. Seznamte se s daty genové exprese, způsoby jejího měření a zpracování.
2. Seznamte se s genomickými anotačními databázemi, soustředte se na sekvenční databáze.
3. Prostudujte alternativní způsoby sdružování množin sond.
4. Vybranou metodu implementujte.
5. Statisticky vyhodnoťte kvalitu vzniklé reprezentace.
6. Využijte ji pro klasifikaci, vyhodnoťte dosažené přesnosti na několika typových úlohách.

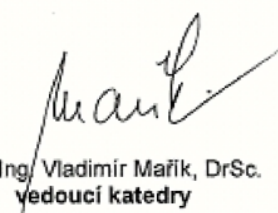
Seznam odborné literatury:

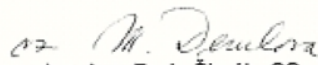
- [1] Baxevanis, Ouellette: *Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins*. 3rd Edition, Wiley, 2005.
- [2] Yu, Hui; Wang, Feng; Tu, Kang; Xie, Lu; Li, Yuan-Yuan and Li, Yi-Xue: Transcript-level Annotation of Affymetrix Probesets Improves the Interpretation of Gene Expression Data. *BMC Bioinformatics*, Vol. 8 (11 June 2007), 194.

Vedoucí bakalářské práce: Ing. Jiří Kléma, Ph.D.

Platnost zadání: do konce zimního semestru 2009/2010




prof. Ing. Vladimír Mařík, DrSc.
vedoucí katedry


doc. Ing. Boris Šimák, CSc.
děkan

V Praze dne 23. 2. 2009

Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze dne 10.6.2009



.....

Czech Technical University
Faculty of Electrical Engineering



Knowledge-Oriented Genomic Probeset Consolidation

(A Bachelor Project)

Hynek Urban

Bachelor Project Supervisor: Ing. Jiří Kléma, PhD.

Study programme: Electrical Engineering and Information Technology

Specialization: Cybernetics and Measurement

May 2009

Abstract

This work examines the possibilities of alternative genomic probeset consolidation using the Affymetrix annotation data and utilizing them in conjunction with various publicly available databases of genomic sequences. The protein- and gene-based representations were compared by means of synonymous probesets' correlation, the accuracy of an automatic classifier built upon the two representations and the number of significantly differentially expressed units. It has been found that the protein-based probeset consolidation is a good alternative to the traditional gene-based one, increasing the within-unit correlation of measured data on the one hand, but causing a decrease in the overall number of usable probesets on the other hand.

Abstrakt

Tato práce se zabývá možnostmi alternativního sdružování množin sond v rámci technologie microarray. Využity jsou přitom anotace Affymetrixu ve spojení s veřejně přístupnými databázemi genomických sekvencí. Na základě korelovanosti spřízněných množin sond, přesnosti klasifikátoru postaveného na příslušných reprezentacích a množství signifikantně diskriminujících jednotek byla porovnána sdružení množin sond pomocí genů a pomocí proteinů. Výsledkem bylo zjištění, že vedle tradičního způsobu sdružování množin sond na základě jejich příslušnosti ke genům je lze sdružovat i na základě proteinů, což má na jedné straně za následek zvýšení korelovanosti sdružených množin, ovšem zároveň snížení celkového množství využitelných naměřených dat na straně druhé.

Acknowledgements

This project would certainly never exist without Mr. Jiří Kléma, to whom I would like to express my thanks. He introduced me to the field of genomics, led me along the way and encouraged me in moments of greatest despair.

Then I would like to thank to my girlfriend Mirka for her understanding and for her careful proof-reading of this text.

Finally, I feel grateful towards all my family members and other people close to me, who supported me while I was working on this project in every thinkable way.

Contents

Abstract	ii
Abstract (Czech)	iii
Acknowledgements	iv
List of Tables	viii
List of Figures	ix
List of Abbreviations	x
1 A brief introduction to the microarray technology	1
1.1 DNA microarrays	1
1.2 Microarray data preprocessing	2
2 Sequence databases - useful background knowledge for microarray data	4
2.1 NetAffx probeset annotation	4

2.2	Sequence databases	5
2.3	Kyoto Encyclopedia of Genes and Genomes (KEGG), WikiPathways	7
2.4	Putting the information together	8
2.5	BLAST algorithm	8
3	Probeset consolidation problem	11
3.1	Introduction	11
3.2	Probe level	12
3.3	Probeset level	12
3.3.1	Probeset consolidation based on empirical results	13
3.3.2	Probeset consolidation based on an a priori knowledge	13
4	Probeset consolidation on the GPL1261 chip using the Affymetrix BLAST annotation file	14
4.1	Step 1 – mapping the probesets to proteins	15
4.2	Step 2 – assigning EC numbers to proteins	16
4.3	Step 3 – mapping the enzymes to KEGG nodes and pathways	17
4.4	Comparing the protein- and gene-based mapping upon expression correlation	20
4.5	Comparing the protein- and gene-based mapping by means of classification accuracy	22
4.6	Comparing the two mappings in terms of statistical significance	27
4.7	Results summary	32

5 Final notes	33
References	35
Appendix A - List of Software	37
Appendix B - Contents of the CD	39

List of Tables

4.1	Probeset losses	19
4.2	Properties of the final mapping	19
4.3	Summary of the SAM analysis	31

List of Figures

4.1	Mapping process summary	15
4.2	Distribution of probesets assigned to a gene or protein	17
4.3	Distribution of genes/proteins assigned to a probeset	18
4.4	Probeset–node relationship for the protein-based mapping	20
4.5	Probeset–node relationship for the gene-based mapping	21
4.6	Comparison among mappings using the ”mean PCC” measure	22
4.7	Comparison among mappings using the ”median PCC” measure	23
4.8	Comparison among mappings using the ”minimum PCC” measure	24
4.9	Learning curves for various datasets - part I	26
4.10	Learning curves for various datasets - part II	27
4.11	SAM plots for several datasets - part I	28
4.12	SAM plots for several datasets - part II	29

List of Abbreviations

BLAST	Basic Local Alignment Search Tool
DNA	Deoxyribonucleic Acid
EBI	European Bioinformatics Institute
EC	Enzyme Commission
FDR	False Discovery Rate
GEO	Gene Omnibus
GO	Gene Ontology
ID	Identifier
KEGG	Kyoto Encyclopedia of Genes and Genomes
mRNA	messenger RNA
NCBI	National Center for Biotechnology Information
PCC	Pearson's Correlation Coefficient
PICR	Protein Identifier Cross-Reference service
RNA	Ribonucleic Acid
SAM	Significance Analysis of Microarrays

Chapter 1

A brief introduction to the microarray technology

In the past two decades, new technologies have come up enabling the measurement of gene expression for large numbers of genes. Following the extraction of RNA from an organism, it has become possible to determine the activation levels of virtually all its genes. This allows for various kinds of analysis of biological processes on a scale never seen before, but at the same time constitutes a serious challenge for the analysts, as the volume of the genomic data in combination with its nature and specifics make the use of traditional analysis methods rather complicated. One of the technologies making gene expression profiling possible is the microarray technology. Although it is beyond the scope of this work to describe the whole procedure of microarray data processing, and it is certainly not my intention to picture all the known pitfalls here, in this section I would like to give a short summary of what a microarray actually is, and how the data from a microarray experiment should be dealt with.

1.1 DNA microarrays

A DNA microarray is a technology used to perform gene expression profiling (i.e. to measure gene expression levels on a genome scale) using tens of thousands of short

nucleic acid sequences arranged in an array of a predefined structure. These short (25-mer) sequences are called probes, and are designed to be complementary to the mRNA sequences of interest. Probes are consolidated into larger groups known as probesets, each probeset corresponding to a gene. During the experiment, hybridizations of probes with target sequences are detected and quantified using fluorophore-labeled targets. The fluorophore is excited with laser, and the whole microarray is then scanned in a microarray scanner. The resulting image represents the measured microarray data in the most basic form. Often, two fluorophores with different emission wavelengths are used in the so called two-channel detection – the identification of up- and down-regulated genes then proceeds on the basis of their relative intensities [19].

One of the most important microarray chip manufacturers is Affymetrix; and since all the data I used in this work came from the *Affymetrix Mouse Genome 430 2.0* chip, Chapter 2 offers a few more details about Affymetrix GeneChip design and annotation.

1.2 Microarray data preprocessing

After numerical expression values are extracted from the image, there are still two basic steps to be done before the main analysis takes place. Data cleaning is the first of them - the aim of this step is to detect and remove evident non-biological artifacts in the measured data. The affected genes (or probesets) are either entirely removed from the dataset, or, when the experimenter wants to avoid unnecessary loss of information, the outlier values are in some way substituted, for instance by averaging the expression values of the given gene from other samples.

A typical microarray experiment includes tens to hundreds of samples, whose gene expression levels are measured. The whole sample set might be divided into several classes corresponding to various biological phenomena such as tissue types, diseases, or distinct phenotypes. Another possibility is to measure a single biological phenomena at various time points. In any case, cross-sample comparisons are made, which

assumes that the measured values are comparable. The comparability is ensured by miscellaneous normalization methods, attempting to remove the non-biological variation in the data. The family of normalization methods includes among others scaling, cyclic-loess normalization and quantile normalization. The latter two were proposed and compared in [3], the former one is described for example in [4].

Sometimes, a third step is placed in row with the two previous ones - feature selection and/or extraction. These two concepts aim to reduce the dimension (the total number of attributes – genes) by filtering out some of them (feature selection) or by constructing new attributes using combinations of the old ones (feature extraction). However, both feature selection and extraction are very closely related to the *probeset consolidation problem*, as described in Chapter 3 and partly also Chapter 2. In fact, probeset consolidation poses a way to select and extract features using biological criteria. Standard statistical methods of feature selection and extraction, such as feature selection based on information gain, or Principal Component Analysis, are suitable for genomic data as well [9].

Further microarray data processing involves i.a. classification tasks, significance analysis, and of course interpreting the results. The following sections show a less traditional way of looking at the expression values, and also provide examples of classification and significance analysis using different genomic data representations.

Chapter 2

Sequence databases - useful background knowledge for microarray data

When processing microarray data, there is much more that can be discovered and examined than just the differences between probeset expression values. In this section, I will try to present an overview of additional information provided by Affymetrix, other information available through various public databases (focusing on the databases containing gene and protein sequences), and finally to outline how these two can be used together.

2.1 NetAffx probeset annotation

NetAffx [11] is an on-line system developed by Affymetrix with the purpose of providing additional details and annotations for probesets in Affymetrix GeneChip microarrays. There are two basic categories of information available:

Static information – Each probeset consists of eleven oligonucleotide probes. Exact nucleic acid sequences of all of them as well as the representative sequence (also

called consensus sequence) of the entire probeset, i.e. the sequence best associated with the interrogated region, are stated for each probeset. To unambiguously identify and localize the region to be interrogated, UniGene (see below) cluster ID is also present in the probeset annotation.

Sequence annotations – Those include annotations obtained from various public databases, such as Entrez Gene, GenBank or Swiss-Prot, related to the probeset representative sequence. By that, the probesets alliance with various biologically defined groups, such as genes, transcripts (proteins), enzyme families, metabolic pathways and others, is indicated, as well as secondary information in the form of GO terms, homolog/ortholog relationships to probesets on other Affymetrix chips, etc.

2.2 Sequence databases

There are hundreds of publicly available databases of genomic sequences. Some of them are limited to certain organisms, the others are universal. Although the total number of such databases is relatively high, there are three major primary sequence repositories: DDBJ (DNA databank of Japan), the EMBL (European Molecular Biology Laboratory) database and NCBI GenBank (National Center for Biotechnology Information - USA) [13]. These three databases mainly collect manually submitted nucleic acid sequences (as most of the scientific journals require such a submission before accepting an article describing the given sequence), that are mutually exchanged on a daily basis so that their contents remain practically identical. A record in any of these databases typically includes the given genomic sequence, its location and gene identification, information about coding sequences (if there are any) and their translation into amino acid sequences, cross-links to other databases, and of course submitter information.

The GenBank records are eventually reviewed and compiled into another database, NCBI RefSeq [16], comprising a non-redundant collection of RNA, DNA and protein sequences for various organisms. [16] says that "RefSeq differs from GenBank in the same way that a review article differs from a related collection of primary research

articles on the same subject.”. Data from NCBI RefSeq together with data from collaborating model organism databases are, in part as a result of curation and in part automatically, integrated into an NCBI database of gene-specific records called Entrez Gene [12]. Entrez Gene includes only gene-specific information for completely sequenced genomes. These information allow for linking genes to their products, homologs, etc.

To mention a non-NCBI genomic database, EBI Ensembl is a comprehensive genome information system, differing from Entrez Gene especially by ”providing relationships between genes and genomes in a comparative genomics framework in the form of sequence alignments, ortholog and paralog assignments and gene trees.” [6]

In the chaos of different gene IDs from different databases and single genomic sequences that can overlap or be redundant, whose gene assignment could be unclear etc., a need for a system has arisen that would be able to order all genomic data systematically. That is what UniGene, ”a largely automated analytical system for producing an organized view of the transcriptome” [15], is for. With its system of clusters, it classifies all sequences, locates them on the chromosome, links the isoforms and performs other similar tasks. NetAffx annotation uses UniGene clusters and subclusters for unambiguous identification of the interrogated regions.

Regarding the protein sequence databases – aside from dozens of less important or specialized ones, the major up-to-date protein database is the UniProt [1], specifically the UniProt Knowledge Base. Historically, UniProt developed by a fusion of three other protein databases: European Bioinformatics Institute (EBI), Swiss Institute of Bioinformatics (SIB) and Protein Information Resource (PIR). Nowadays, new proteins are added exclusively on the basis of GenBank entries, which is possible because most of the submitted genomic sequences involve the information about which parts of the sequence have been recognized as the so called *coding sequences*, i.e. sequences delimited with a start- and a stop-codon, that are later translated into proteins. These nucleic acid coding sequences can be easily converted into amino acid sequences characteristic for the individual proteins. The UniProt database also includes additional information such as different isoforms of the given protein, GO annotation and cross-links to other databases. The UniProt Knowledge Base is divided

into two branches, one being manually reviewed, annotated and checked for redundancy (UniProtKB/Swiss-Prot), the other being annotated automatically (UniProtKB/TrEMBL).

Because UniProtKB is partly redundant, and because there are other sources of protein sequences as well (Protein Data Bank, European Patent Office, GenBank...), there are efforts to include all known protein sequences into a system of unique, minimally redundant protein identifiers. Two major initiatives are the UniParc (assigning every protein sequence a stable UPI) and IPI (International Protein Index - by EBI).

2.3 Kyoto Encyclopedia of Genes and Genomes (KEGG), WikiPathways

Among other sources of biological information, I would like to point out the databases aimed at covering higher-level behavior of cells and organisms on grounds of genomic information. One of the means for doing that is to explore the networks of biochemical reactions induced and catalyzed by enzymes. Because enzymes are basically gene products, the impact of miscellaneous genes on an organism can be principally observed directly by inspecting the effects of their enzyme products. However, as the relationships within these biochemical networks are generally too complex to examine the effects of single genes and enzymes, they are modeled as a whole in the form of the so called pathway maps. There are two important databases containing detailed structures and annotations of biological pathway maps – the Kyoto Encyclopedia of Genes and Genomes (KEGG) [8] and WikiPathways [14]. As for the former, KEGG represents pathway maps as networks of nodes, each node being assigned one or more EC numbers (enzyme identifiers). By linking genes to enzymes and reflecting the measured expression values, one can inspect the levels of activation in various parts of a given pathway map. The most complete is the KEGG reference metabolic map, being also used by the gene-based cross-platform microarray analysis tool XGENE [5], and consequently by me (as described in Chapter 4).

2.4 Putting the information together

When processing microarray data, one usually wants to obtain results that can be somehow biologically interpreted. In the classical case of two-class classification, it means that we would like to find out what biologically meaningful attributes actually discriminate between the two classes. If we had probeset identifiers only, we could eventually state that a given set of probesets discriminates between the two classes – but what biological information would be there? That is why Affymetrix assigns probesets to genes, so that we don't have a set of discriminative probesets, but a set of discriminative *genes*. That is much more informative already, but we don't have to stop there, because genes can be assigned to other biologically meaningful units, such as transcripts, enzymes, pathway nodes, etc. These assignments can be done either using the ready-made Affymetrix probeset annotation, or using some of the mentioned public databases. The latter way is proper in a situation where the experimenter wants to retain full control over the assignment process.

Suppose we would like to assign probesets in the first step not to genes, but to proteins (see Chapter 3 to find out why would we do that). If we didn't want to use the assignment to SwissProt or RefSeq Protein IDs already present in the Affymetrix annotation file, we could align the probeset consensus sequences (or even the actual probe sequences) against a database of proteins to find corresponding transcripts. To achieve that, we could employ e.g. the BLAST algorithm. When further consolidating proteins into even higher-level units, we can utilize information from different databases either on grounds of cross-database links indicated in the database records or by calling on an explicit tool for finding corresponding identifiers, as for instance EBI PICR. Chapter 4 shows an example of doing all that.

2.5 BLAST algorithm

Finally, I would like to shed some light on how the alignment of genomic sequences actually proceeds. There are many ways and algorithms of sequence aligning; perhaps the most widely used one in the area of bioinformatics is BLAST, or Basic Local Align-

ment Search Tool. It has been designed for searching sequences in large databases with two specifics in mind that make the use of traditional sequence aligning methods complicated:

- the databases of genomic sequences are usually very large, which makes the speed of an aligning algorithm more important than its accuracy
- in many bioinformatical applications, it is desirable not only to find exactly matching sequences, but to return similar sequences as well, representing isoforms, homologs/paralogs of the query sequence, etc.

The search itself proceeds in three steps [7]:

1. *The seeding step* – The query sequence is divided into words of a defined length k (usually $k = 3$ for protein sequences, $k = 11$ for nucleic acid sequences), e.g. the sequence 'PEGQFG' contains words 'PEG', 'EGQ', 'GQF' and 'QFG'. For each word, a list of best-scoring matching words (built upon a scoring matrix) is generated.
2. *The extension step* – The matching words from the previous step are searched in the database. If two non-overlapping hits in a correct distance are found, the so called *extension* is invoked. Each pair of potentially matching letters is assigned a score based on the scoring matrix (the score value may be negative). In the process of extension, seeds (the matched words) are extended to both sides so that the total score is maximized. This way, a so called High Scoring Pair (HSP) is obtained.
3. *The evaluation step* – HSPs with raw scores under a predefined threshold level are discarded. For each of the remaining ones, the E-value, representing the number of alignments with the same or better score obtained on a database of the same length one would expect to find by chance, is given by the Karlin-Altschul equation:

$$E = kmne^{-\lambda S} \tag{2.1}$$

where k is a minor constant, m is length of the query sequence, n is length of the database, S is the raw score and λ is a scaling factor dependent on the particular scoring matrix. Finally, sequences containing HSPs with E-values above the user-defined threshold are reported.

Chapter 3

Probeset consolidation problem

3.1 Introduction

Affymetrix GeneChip is a microarray platform used to measure gene expression at the genome scale. Each gene is represented by one or more probesets on the chip. Ideally, different probesets representing the same gene should yield similar expression values. In reality, however, this is not always the case. To fully understand this problem, it is necessary to look more closely at how the expression values are actually obtained.

During the process known as transcription, DNA is transcribed into mRNA, representing the protein-building instructions of the genes. Each probeset consists of eleven probes, each of which is a 25-mer oligonucleotide complementary to a particular mRNA sequence (called target). During the microarray experiment, probe-target hybridizations are detected and quantified, and the probeset expression is computed upon the expression values of all its probes. However, a probeset doesn't cover the whole sequence corresponding to a gene, only its parts. The problem here is that the transcription of a gene (i.e. a given sequence of DNA) into mRNA is not unambiguous. There are effects called alternative splicing and alternative polyadenylation concerning some of the genes, resulting into the fact that the same gene may produce different mRNA transcripts under different conditions (the transcript is always composed of

sequences complementary to the original DNA, however, the parts of the genomic sequence that are actually transcribed may differ). When this happens, a probeset might for example detect only one of the two possible transcripts or conversely - it can detect more than one. In case of sibling probesets (i.e. probesets representing the same gene), it is obvious that sometimes the supposedly synonymous probesets detect different transcripts of the same gene, and that is also why their expression values aren't as correlated as desirable.

Another cause for sibling probesets to yield uncorrelated expression values are annotation errors. As shown in [17], the Affymetrix annotation of probesets is sometimes inaccurate, the probeset being assigned to another gene than the one it actually represents.

There are two levels at which these problems can be dealt with when consolidating probesets: probe level and probeset level.

3.2 Probe level

The very definitions of probesets as sets of certain probes come from Affymetrix Chip Definition Files (CDF). In these files, each probeset identifier is assigned eleven probes as designed by Affymetrix. However, this assignment can be in principle changed in a custom CDF so that the newly created probe aggregations represent disjunct transcripts. Ways how to achieve this are basically the same as those described in the next section. The main drawback of this approach lies in the fact that the low-level probe expression values are often inaccessible to the experimenter. From now on, I will focus solely on the probeset level approach.

3.3 Probeset level

Another way is to re-consolidate the probesets based on another criteria than just the Affymetrix gene annotation. There are two basic ways of doing that (aside from

naive approaches like treating all probesets as distinct genes and not consolidating them at all): the one is empirical, using the measured expression values to compute correlations between sibling probesets, the other is in some way using the background knowledge hidden in the probeset sequence annotation.

3.3.1 Probeset consolidation based on empirical results

This approach follows a very simple reasoning: supposed we would like the probesets to be consolidated in such way that the consolidated probesets display correlated expression values, the easiest way to do it is to consolidate them using the measured expression values. Because it is desirable to retain biologically meaningful entities (genes assigned to probesets), this concept is usually used to separate probesets representing different transcripts of the same gene rather than to combine probesets assigned to different genes. [10] shows an example of a successful implementation of this approach using the Analysis Of Variance (ANOVA) framework.

3.3.2 Probeset consolidation based on an a priori knowledge

Chapter 2 gave the idea of the huge amount of information available through various sequence databases. Generally, it is possible to align the probe or probeset nucleotide sequences against various databases to find transcripts corresponding to the individual probesets to be further used as probeset labels instead of genes. [20] demonstrates an example of using the BLAST algorithm to align probe sequences against protein sequences extracted from GenBank, RefSeq and Ensembl. The authors show that the newly consolidated probesets are more correlated than originally. Nevertheless, it is not necessary to perform sequence aligning independently, as Affymetrix provides annotation files for each chip, containing the top results of the alignment of probesets against the GenBank protein database using the BLAST algorithm. I attempted to utilize the BLAST annotation file for the chip GPL1261 (Mouse Genome 430 2.0) and to estimate the fitness of the representation based upon the resulting probeset consolidation, and that is what the next chapter describes.

Chapter 4

Probeset consolidation on the GPL1261 chip using the Affymetrix BLAST annotation file

As outlined in the previous chapter, it is possible to group probesets by transcripts they detect; the groups of probesets obtained in this way are potentially more correlated than the original groups formed on the basis of respective genes. I chose the chip GPL1261 (Mouse Genome 430 2.0) and decided to examine the transcript-based (from now on denoted mostly as protein-level) probeset consolidation and its properties, as well as to map the consolidated probesets to higher entities (KEGG nodes and pathways) and to compare the obtained mapping with the gene-based representation on several datasets in terms of classification accuracy and the number of significant entities. In order to estimate the fitness of the protein-based mapping, I decided to use the existing gene-based mappings of XGENE for comparison. XGENE uses the standard Affymetrix probeset linking with respect to genes. The enzyme-coding genes are mapped to KEGG nodes (nodes in metabolic pathways), and in the last step, KEGG nodes are mapped to KEGG pathways. As a result, all units can be represented as groups of probesets. A node or pathway expression is then computed as the average expression across all its probesets. Note that not all probesets are included in the mapping, as the majority of genes code non-enzymatic proteins - only

2712 different probesets are mapped to enzymes.

To be able to compare the protein-based mapping directly with the existing gene-based XGENE one, I had to create a similar probeset – protein – enzyme – KEGG node – KEGG pathway mapping scheme. The whole mapping process is summarized in Figure 4.1.

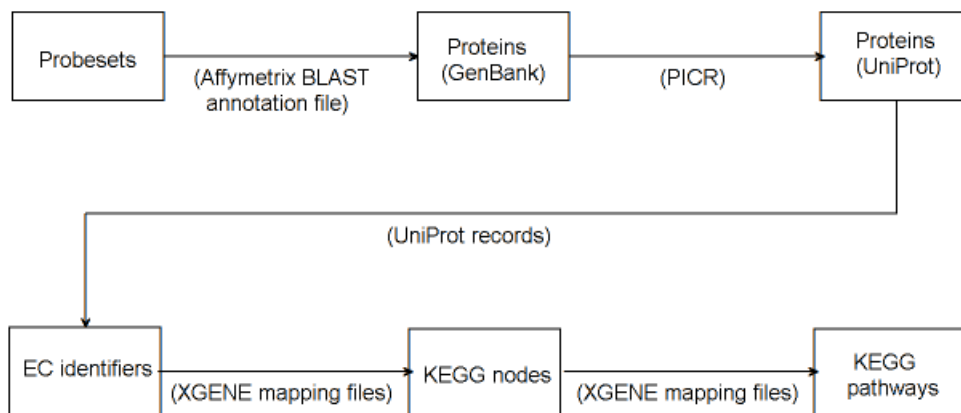


Figure 4.1: Mapping process summary

4.1 Step 1 – mapping the probesets to proteins

Two input files were used – the standard Affymetrix annotation file for the GPL1261 platform ('Mouse430_2.na27.annot.csv') and the Affymetrix BLAST annotation file ('Mouse430_2_blastx.csv'). As for the latter – all probeset consensus sequences had been BLASTed by Affymetrix against the GenBank sequence database, and proteins with the best score (with an E-value under a certain threshold) were presented as possible transcripts for each probeset. I mapped the probesets to all their candidate proteins with E-value under 10^{-50} , resulting into a set of GenBank protein identifiers, each being assigned a set of one or more corresponding probesets. The total number of proteins in the mapping was 56 897, however, because

I assigned each probeset to most of its candidate protein identifiers, many of those proteins were redundant (meaning that their sets of probesets were identical) – only 21 366 protein identifiers had unique sets of probesets; 28 590 unique probesets were mapped to them. Figures 4.2 and 4.3 show a comparison between the protein-based and gene-based mappings (only non-redundant protein identifiers were taken into account). Figure 4.3 also reveals that the total number of probesets being mapped to proteins is approximately by 9 000 lower than the number of probesets mapped to genes. Those 9 000 probesets are the first of many losses in the process of mapping probesets to higher entities.

4.2 Step 2 – assigning EC numbers to proteins

In this step, I converted the GenBank protein identifiers into UniProt identifiers using the PICR tool. This was one of the possible ways to carry out the protein – enzyme mapping, since UniProt indicates the EC identifier in each protein record (of course only when the given protein forms an enzyme). The assignment of UniProt IDs to GenBank proteins by PICR wasn't definite, because the PICR output file stated more possible variants of corresponding UniProt identifiers to each GenBank one. I took into account only those having an attribute "identical", which means that the respective proteins match completely (another attribute is "logical", which indicates only a very similar protein). Then from the set of possible corresponding UniProt identifiers (in case there were more than one), I chose preferably the one with an EC identifier present in the database record, if possible.

In the end, I had 1725 different probesets mapped to 1026 enzymes. Table 4.1 shows a well-arranged summary of probeset losses in each step.

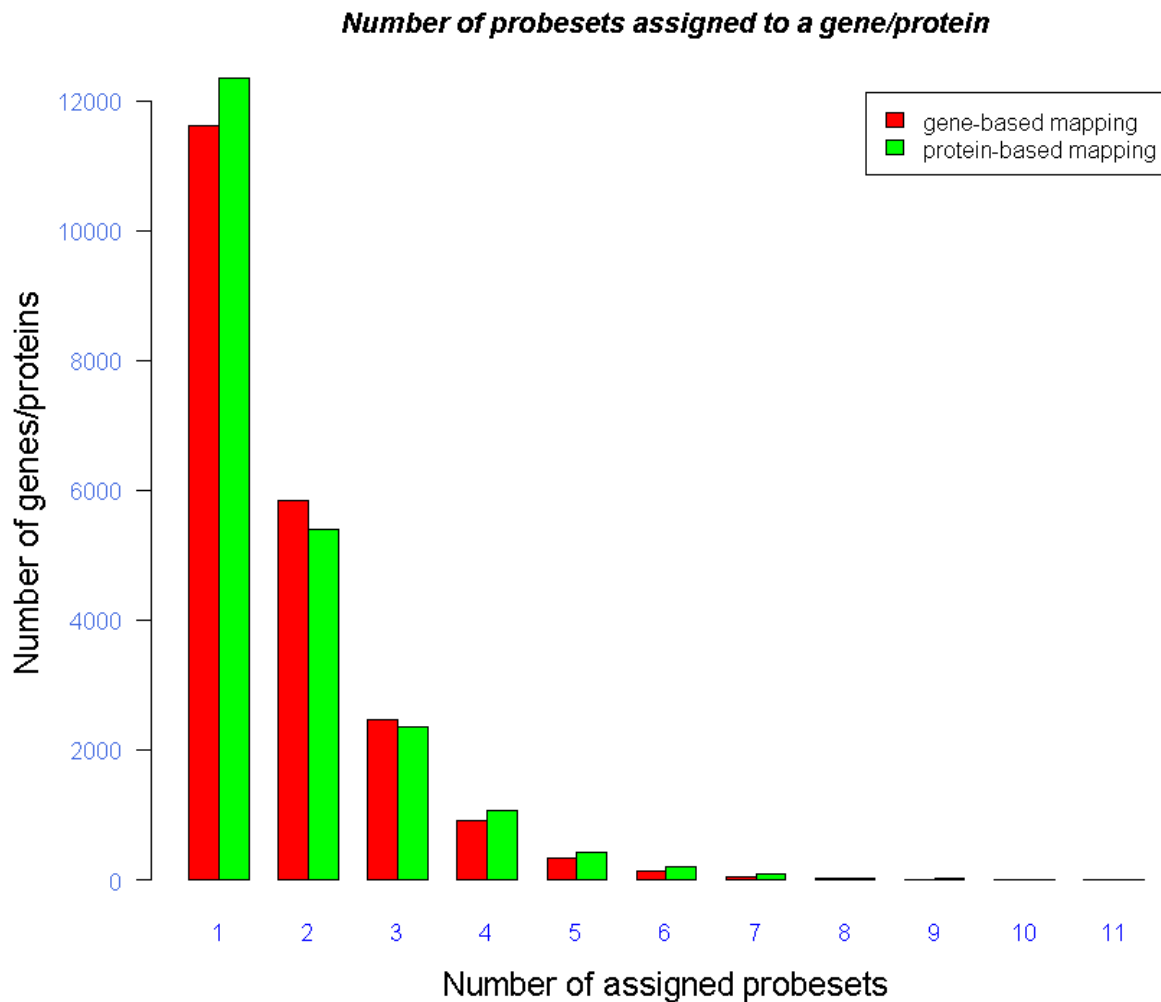


Figure 4.2: Distribution of probesets assigned to a gene or protein

4.3 Step 3 – mapping the enzymes to KEGG nodes and pathways

I simply used the XGENE files that map enzymes to KEGG nodes, and KEGG nodes to KEGG pathways to set up the same mapping. The total number of nodes in the reference metabolic map (the only map used by XGENE) is 2573, forming 261 pathways. Not all the enzymes obtained in the previous step were involved in this mapping, and not all the nodes and pathways were covered by at least one probeset.

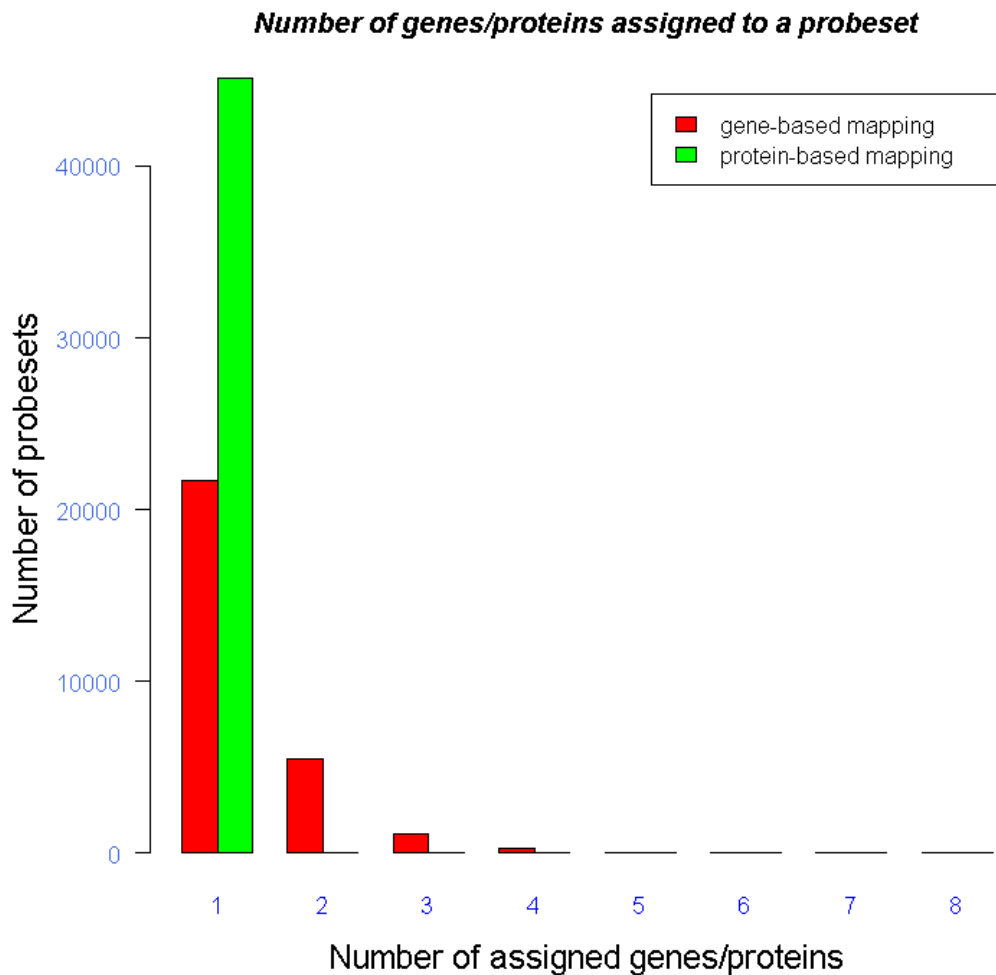


Figure 4.3: Distribution of genes/proteins assigned to a probeset

Properties of the final mapping are summarized in Table 4.2. Figures 4.4 and 4.5 show cumulative histograms of the number of probesets mapped to a node both for the protein- (Fig. 4.4) and gene-based (Fig 4.5) mappings. There is an obvious difference between the two distributions, as the nodes in the gene-based mapping tend to have more probesets mapped to them; the same holds true for pathways, which agrees with the fact that there is a higher number of unique probesets figuring in the gene-based mapping than in the protein-based one.

Step	Probesets total	Probesets successfully mapped	Number of resulting units
Probesets → GenBank protein IDs	45100	28590	21366
GenBank → UniProt IDs	28590	27097	17867
UniProt → EC IDs	27097	1725	1026
EC → KEGG nodes	1725	1337	833
KEGG nodes → KEGG pathways	1337	1337	252

Table 4.1: Probeset losses

Mapping	Unique probesets	Units total	Units covered	Max. probesets per unit
KEGG nodes, protein-based	1337	2573	833	7
KEGG nodes, gene-based	2712	2573	801	83
KEGG pathways, protein-based	1337	261	252	109
KEGG pathways, gene-based	2712	261	238	234

Table 4.2: Properties of the final mapping

Cumulative histogram of probeset number being mapped to a node

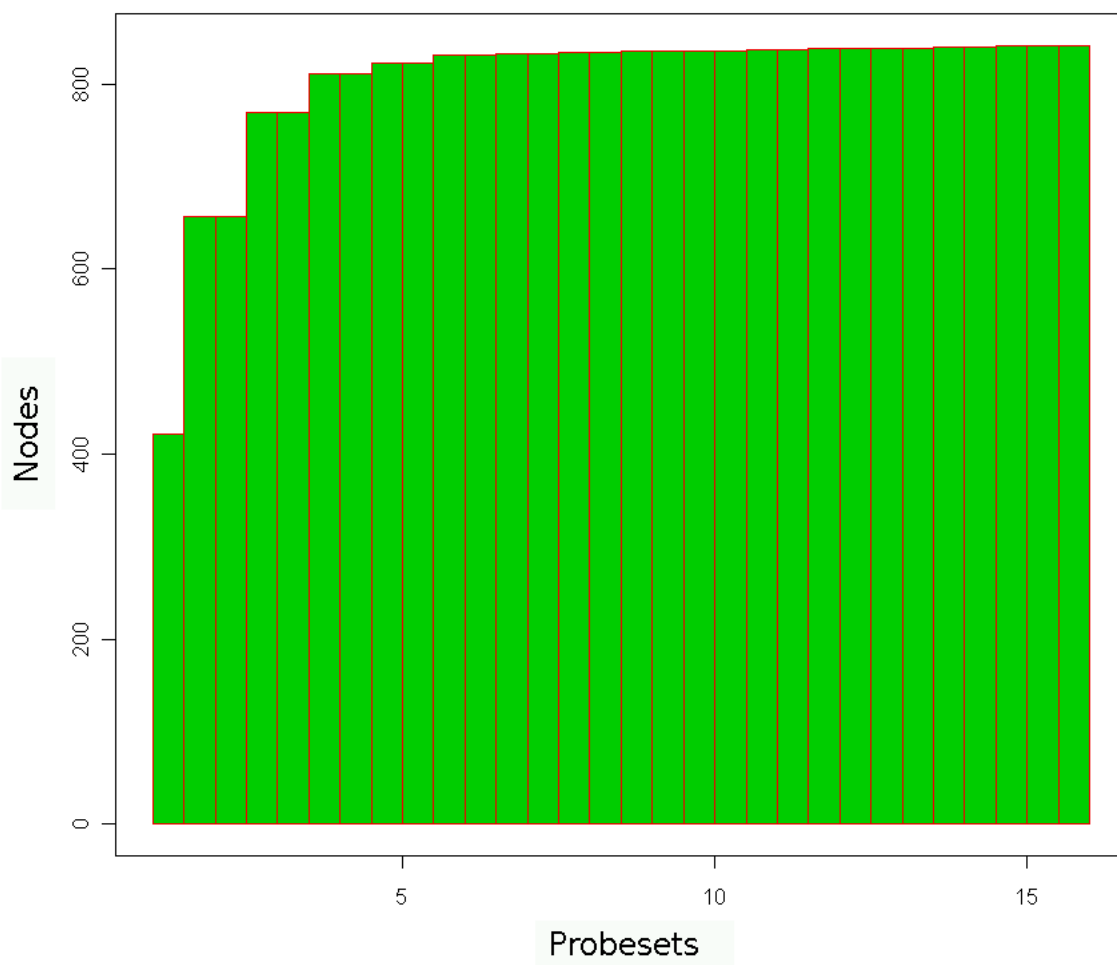


Figure 4.4: Probeset–node relationship for the protein-based mapping

4.4 Comparing the protein- and gene-based mapping upon expression correlation

As the first measure of fitness of the representation obtained in the previous steps, I decided to employ the method used in [20]. The authors computed all pairwise Pearson’s Correlation Coefficients (PCC) within each group of synonymous probesets (with a number of probesets larger than one) and selected the lowest one to represent the expression correlation of the respective group. Values obtained from all groups

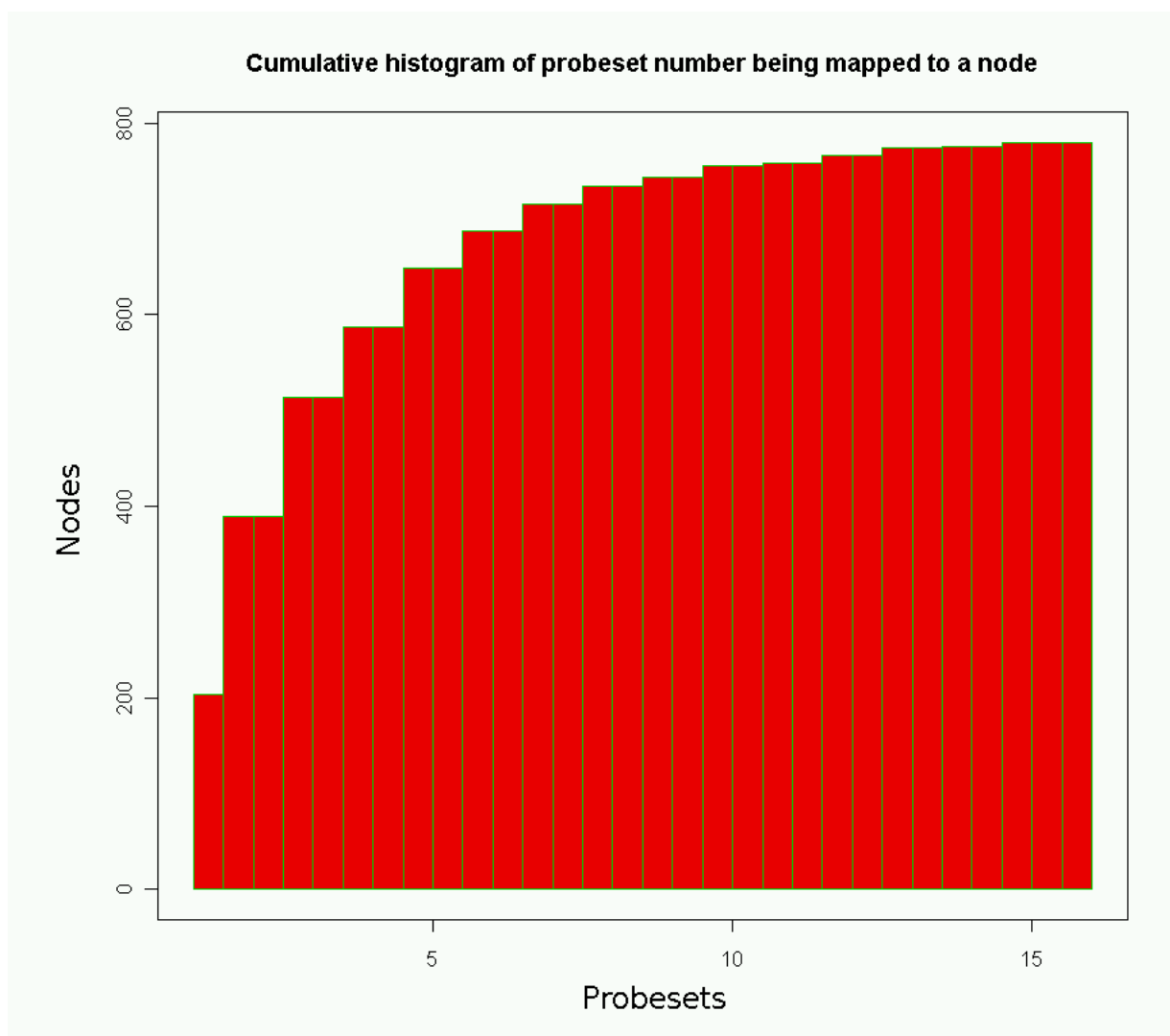


Figure 4.5: Probeset–node relationship for the gene-based mapping

were then averaged into a single value representing the overall correlation of the given dataset. However, since the distribution of probesets among nodes in the gene- and protein-based mappings differs, such measure is not an objective one, as larger groups of values naturally tend to have lower minimums. That is why I added another two values to each dataset, one computed upon the mean, the other upon the median PCC within each group. The results, obtained on several sample GPL1261 datasets downloaded from NCBI GEO [2] and one dataset coming from the original XGENE case study (denoted as "Set 1"), are shown in Figures 4.6, 4.7 and 4.8. The protein-

based mapping consistently displays higher correlation by all three measures across multiple datasets than the gene-based one.

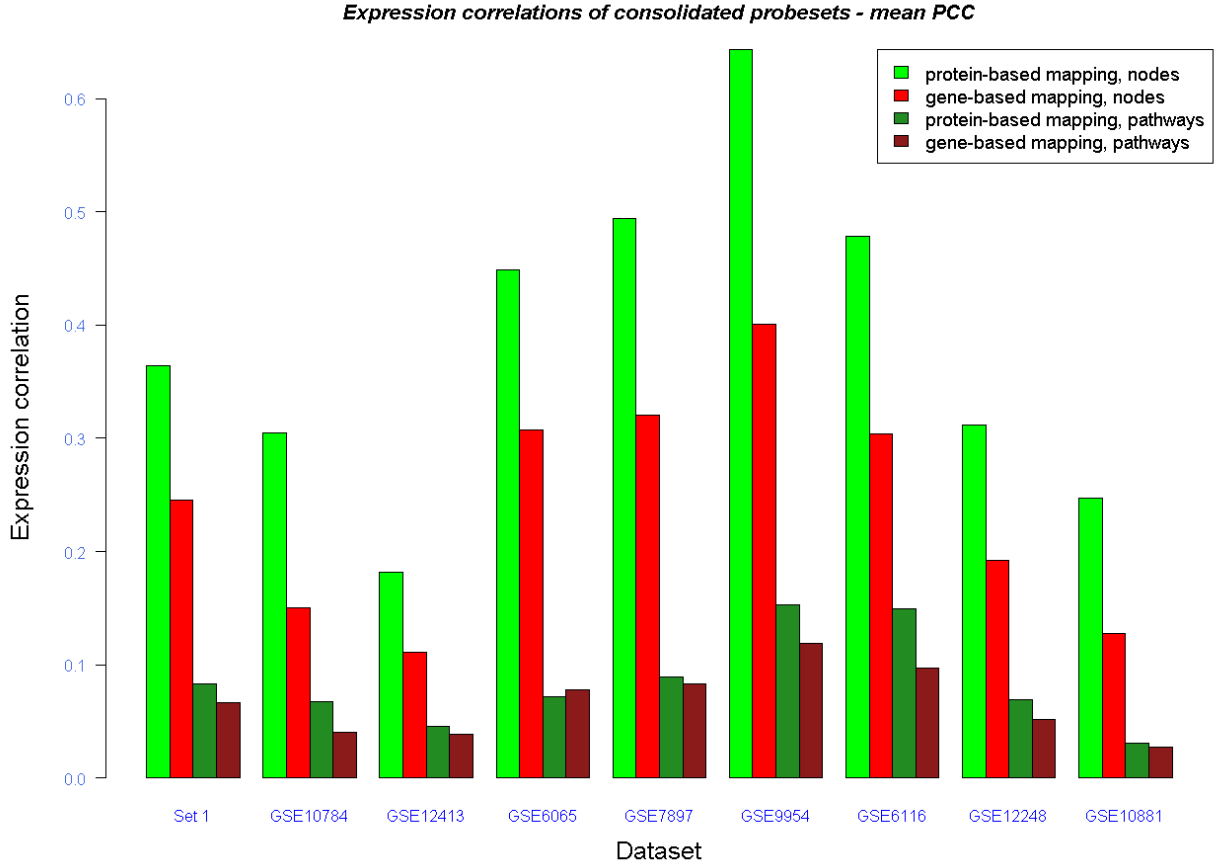


Figure 4.6: Comparison among mappings using the "mean PCC" measure

4.5 Comparing the protein- and gene-based mapping by means of classification accuracy

Next, I wanted to find out whether predictive classification models built upon the two representations somehow differ in their predictive powers. Following the reasoning in [5], where the authors point out that decision tree classifiers represent the family

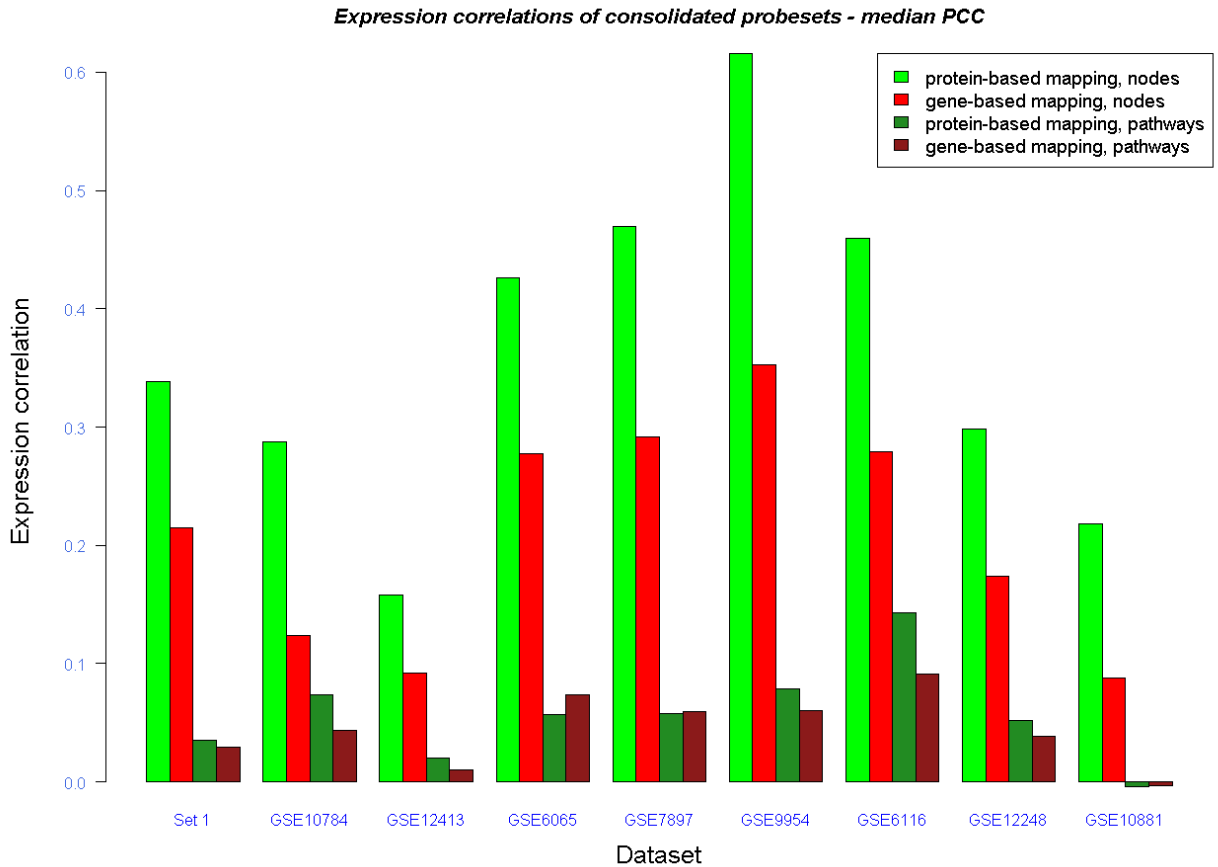


Figure 4.7: Comparison among mappings using the "median PCC" measure

of machine learning algorithms most natural to process microarray data consolidated into biologically meaningful units, as they allow a direct biological interpretation of results, I conducted a series of experiments using the J48 decision tree learner (implemented in the Weka software). The attributes of the samples used for classification were pathway and node expression values, computed as average expressions across all respective probesets.

First I chose the already introduced "Set 1" plus another three datasets from NCBI GEO (see below) suitable for simple classification tasks, i.e. with a reasonable number of samples and with 2 - 3 defined classes. Then the classification learning curves both at the node and pathway level were computed for the gene- as well as the protein-based mappings using the data from each dataset. The beginning of a

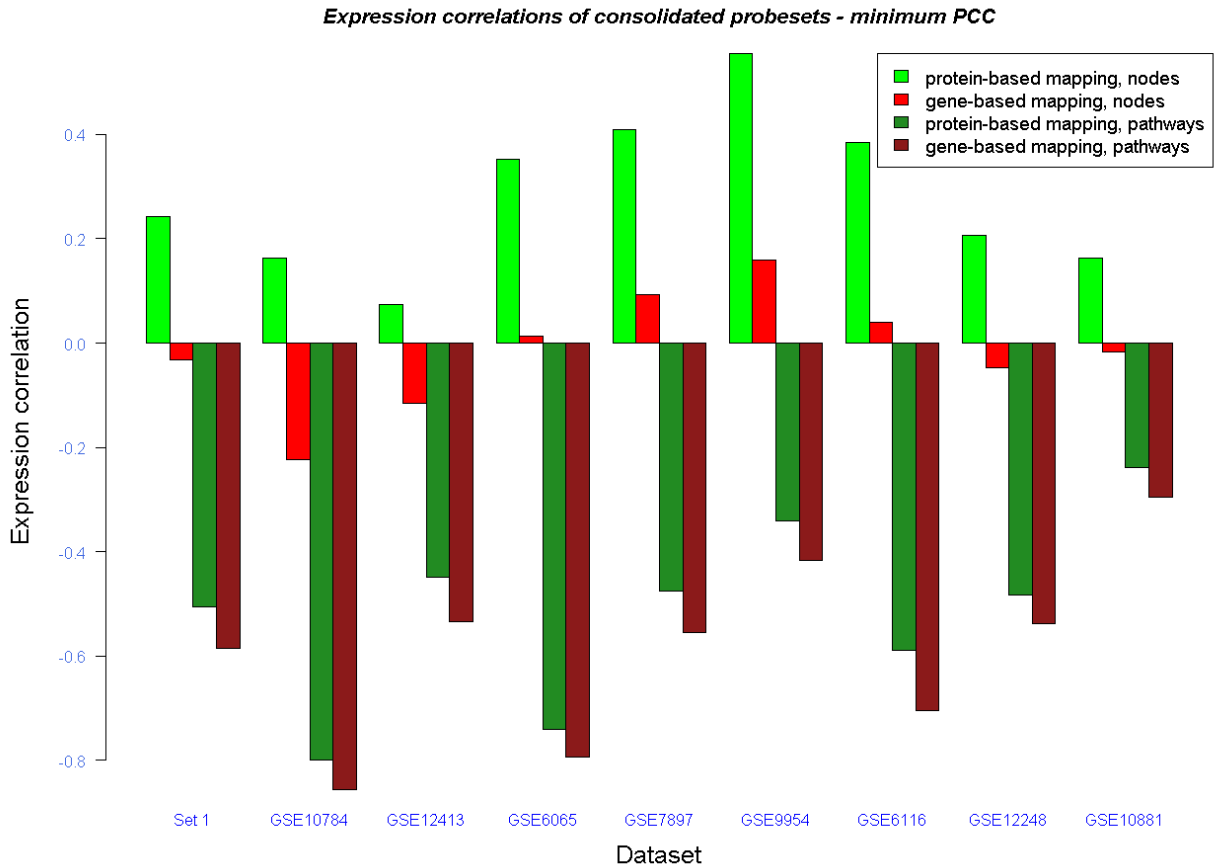


Figure 4.8: Comparison among mappings using the "minimum PCC" measure

learning curve represents a situation where only a small fraction of samples is used to train and test the classifier, while the final part of the curve corresponds to a set-up, where the vast majority of samples is used. At each point, the partitioning into a training and a test set was repeated ten times, the results averaged, and then the final value was plotted as an estimate of the classification accuracy.

The chosen datasets were:

Set 1 – The dataset also used in the XGENE study, consisting of 46 samples of hematopoietic (blood-forming) and 19 samples of stromal (supportive) cells from the bone marrow tissue.

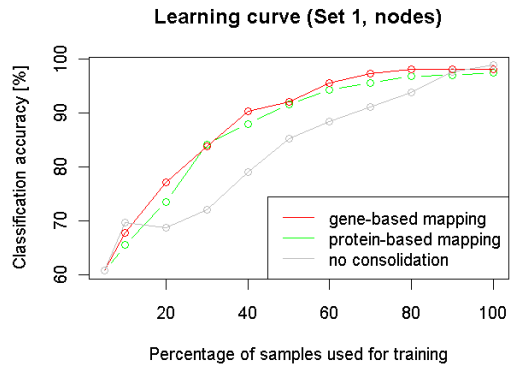
GSE10784 – citing the description from the NCBI GEO database record: "This

represents an unbiased evaluation of the transcriptional response in the prefrontal cortex and hippocampus areas in the Df(16)A/+ mice, a mouse model of human 22q11 microdeletion syndrome.” The dataset consists of 20 wildtype control samples, and 20 samples of Df(16)A/+ mutants.

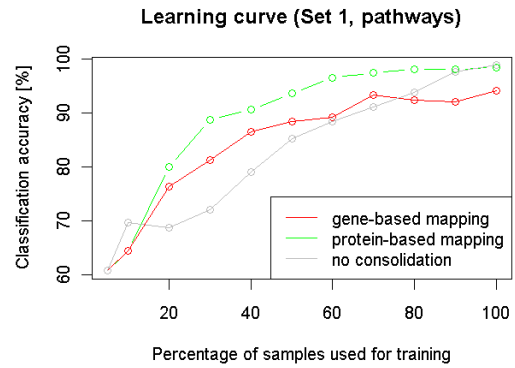
GSE12413 – This set was originally meant to discriminate among different cardiovascular phenotypes of mice subjected to catecholamine stress. However, since the dataset consists of two approximately equally sized classes when divided into a group of non-treated (41) and isoprotenerol-treated (45) samples, I decided rather to discriminate between mice being subjected to catecholamine stress and the control cases.

GSE7897 – A dataset created with the aim of exploring the additional somatic alterations contributing to the heterogeneity of B-cell lymphoma tumors. 25 E μ -myc mice with early-onset and 25 E μ -myc mice with late-onset lymphomas as well as 10 control samples were subjected to a microarray analysis; the classifier tries to discriminate among these three classes.

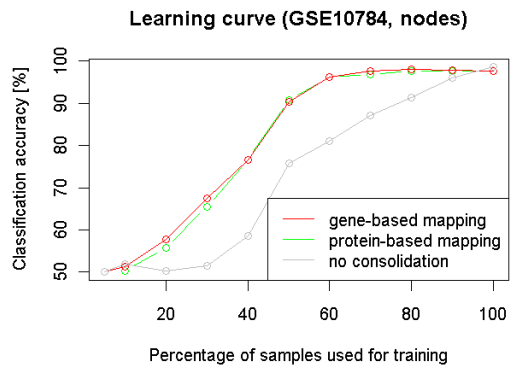
The resulting learning curves are shown in Figures 4.9 and 4.10. It seems that neither the gene-based nor the protein-based mapping is superior to the other one when used as a basis for classifier building. For comparison, learning curves were also computed for the case of non-consolidated probesets, and especially the beginnings of learning curves expose a low quality of such representation. In two cases the protein-based classifier performs slightly better than the gene-based one, two cases display an equal performance, and in four cases it is the gene-based mapping that leads to better results. Notably, in two cases (*GSE12413* and *Set 1*), the protein-based classifier outperforms the gene-based one at the pathway level, despite being comparable or inferior at the node level. However, a more detailed analysis would have to be carried out to assess whether any of the used mappings really gives better classification results; there are even doubts if consolidating the probesets into biologically meaningful units, such as pathways, is in any way more favorable in terms of classification accuracy than consolidating them in a purely random manner when using data from a single chip [5].



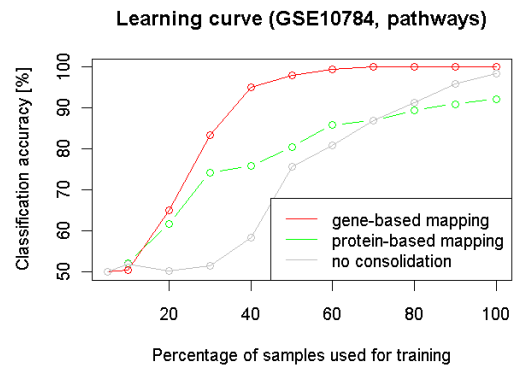
(a) Set 1, nodes



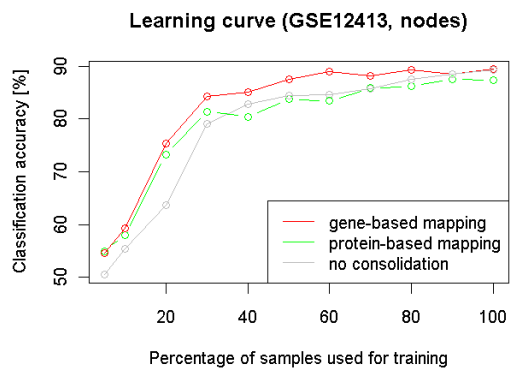
(b) Set 1, pathways



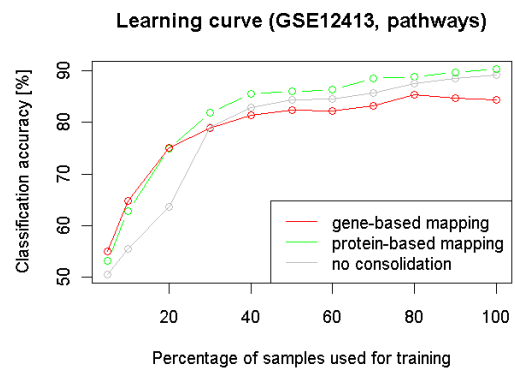
(c) GSE10784, nodes



(d) GSE10784, pathways



(e) GSE12413, nodes



(f) GSE12413, pathways

Figure 4.9: Learning curves for various datasets - part I

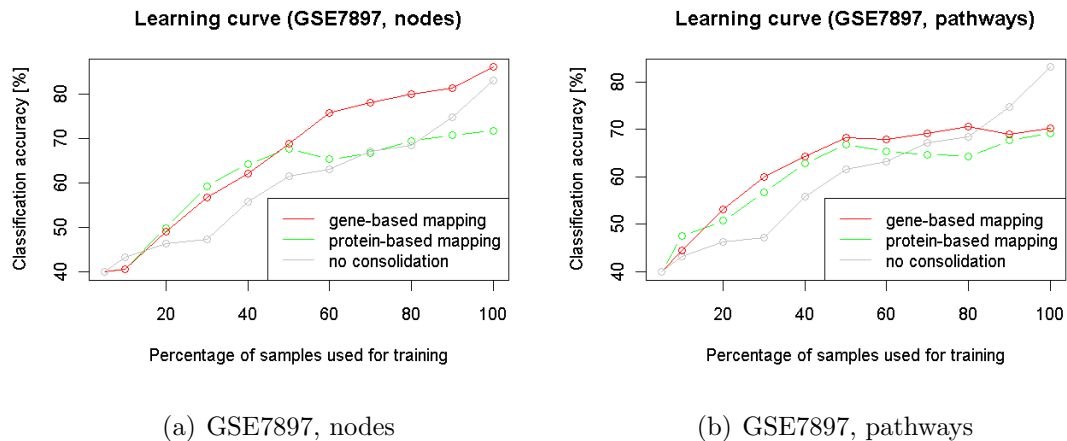


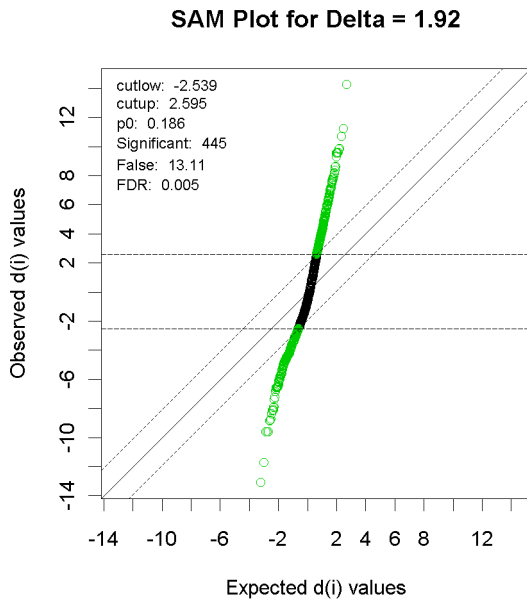
Figure 4.10: Learning curves for various datasets - part II

4.6 Comparing the two mappings in terms of statistical significance

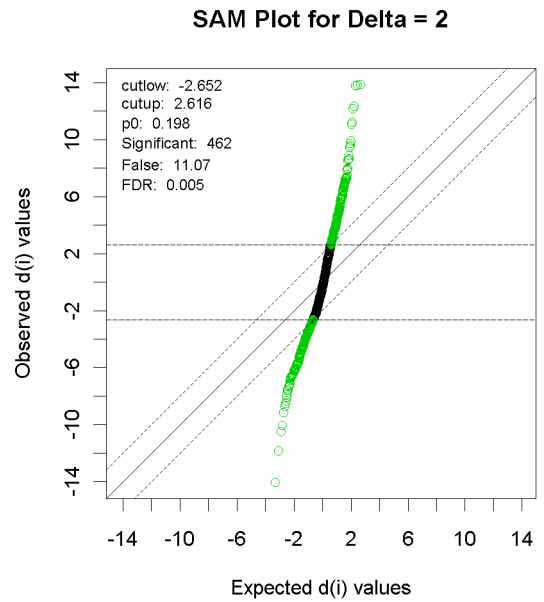
Table 4.2 reveals that the total number of probesets mapped to KEGG nodes and pathways at the protein level is distinctively lower than it is at the gene level. That means, in a sense, that more information get lost during the process of mapping the probesets through transcripts than through genes. However, this "loss of probesets" is not necessarily a negative phenomenon, as it can also be seen as noise filtering. Indeed, the correlation results suggest that the signal to noise ratio in the protein-based mapping is rather strengthened, but there still remains a question that has to be addressed: at the level of nodes and pathways, is the number of highly discriminative units built upon proteins at least comparable to the number of such units built on genes?

To answer that question, I decided to use the standard SAM (Significance Analysis of Microarrays) method [18] implemented in R. SAM is a method specifically designed to determine which genes (if any) are significantly differentially expressed in the two given classes.

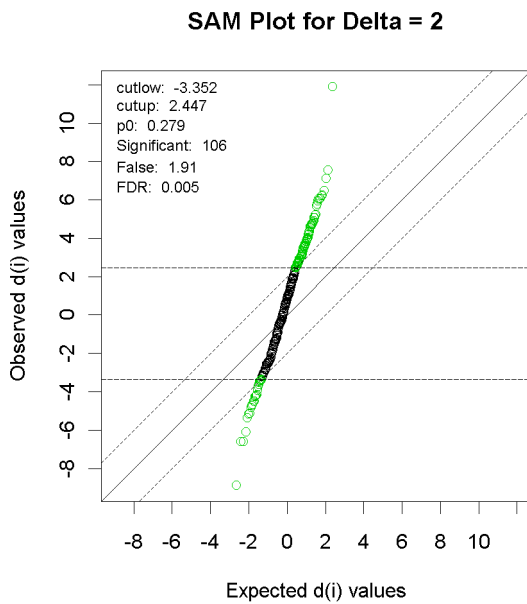
The fundamental quantity in SAM is called *relative difference*, and it is in principle



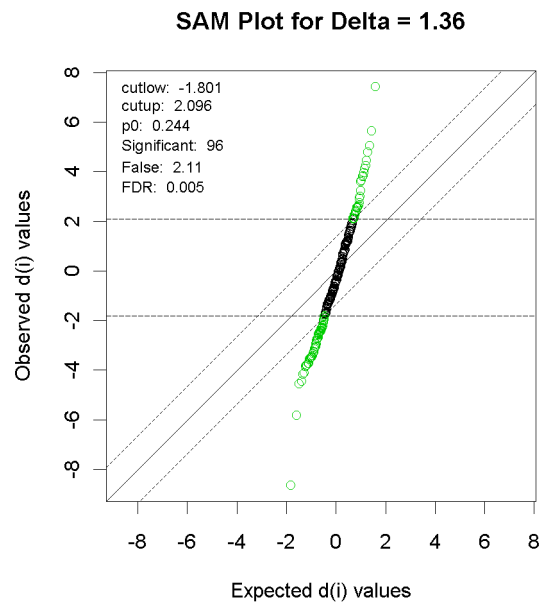
(a) Set1, node level, gene-based mapping



(b) Set1, node level, protein-based mapping

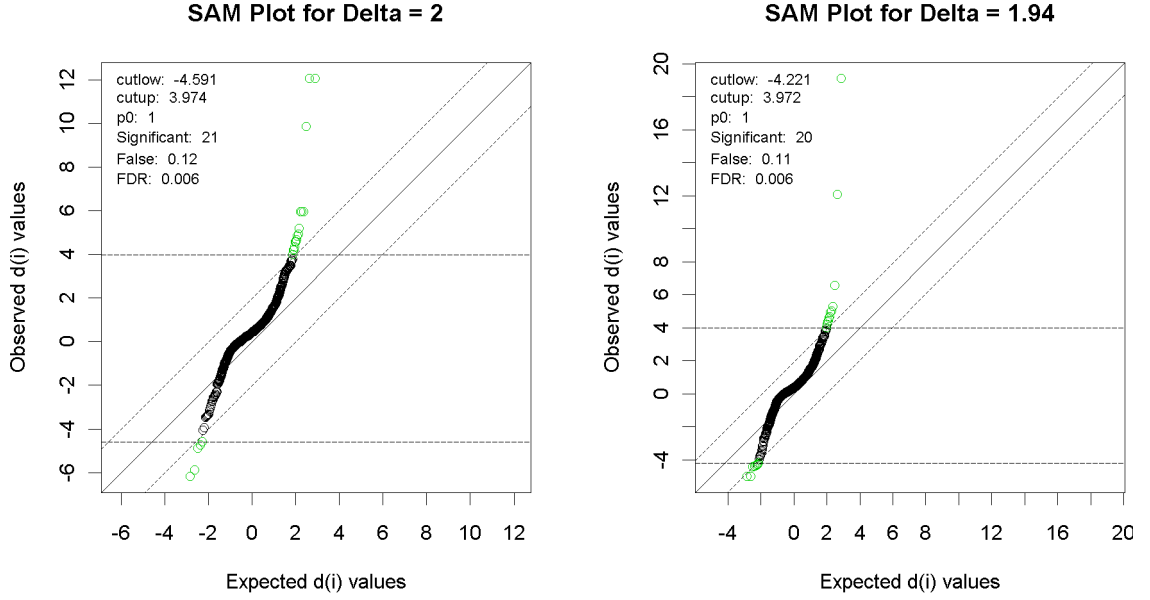


(c) Set1, pathway level, gene-based mapping



(d) Set1, pathway level, protein-based mapping

Figure 4.11: SAM plots for several datasets - part I



(a) GSE10784, node level, gene-based mapping

(b) GSE10784, node level, protein-based mapping

Figure 4.12: SAM plots for several datasets - part II

the change in the mean gene expression between two classes relative to the standard deviation of repeated measurements. The relative difference $d(i)$ for gene i is:

$$d(i) = \frac{\bar{x}_1(i) - \bar{x}_2(i)}{s(i) + s_0} \quad (4.1)$$

where $\bar{x}_1(i)$ and $\bar{x}_2(i)$ are the mean expression values of gene i in classes 1 and 2, s_0 is a small positive constant and $s(i)$ is defined as follows:

$$s(i) = \sqrt{a\left\{\sum_m [x_m(i) - \bar{x}_1(i)]^2 + \sum_n [x_n(i) - \bar{x}_2(i)]^2\right\}} \quad (4.2)$$

where $x_m(i)$ is the expression of gene i in measurement m in the class 1, $x_n(i)$ is the expression of gene i in measurement n in the class 2, and

$$a = \frac{\frac{1}{n_1} + \frac{1}{n_2}}{n_1 + n_2 - 2} \quad (4.3)$$

where n_1 and n_2 are the numbers of measurements in states 1 and 2 respectively.

In the process of finding significantly changed genes, relative differences of all the genes are computed, and the genes are ranked by the magnitude of their $d(i)$ value so that $d(1)$ represents the largest relative difference, $d(2)$ the second largest etc. Subsequently, the expression values are permuted multiple times, the relative differences of all genes are again computed and the genes are ranked, so that $d_p(i)$ is the i th largest relative difference in the p th permutation. The expected relative difference $e(i)$ of a gene is then defined as follows:

$$e(i) = \frac{\sum_p d_p(i)}{P} \quad (4.4)$$

where P is the total number of permutations.

By plotting $d(i)$ vs. $e(i)$, it is immediately evident which genes lie close to the diagonal (and therefore aren't of any interest) and which lie more far (and therefore are more likely statistically significant). By defining a threshold distance, above which the genes are declared as "significantly differentially expressed", one can easily count if, and how many such genes there are in the given dataset. The threshold distance is set empirically so that the FDR (false discovery rate) is acceptably low. The number of falsely significant genes is computed by counting genes with $d(i)$ exceeding the interval between the lowest positive still significant $d(i)$ and largest negative still significant $d(i)$ (the so called horizontal cutoffs) in each permutation, and then averaging the number across all permutations. Finally, FDR is a ratio between this value and the total number of significant genes.

Even though the whole method was originally designed for genes, it can be utilized to count the numbers of significantly expressed KEGG nodes or pathways just as well. I did that for the three datasets I used in the previous step where two classes were defined, and the plots of $d(i)$ vs. $e(i)$ for some of them are shown in Figures 4.11 and 4.12. The obtained results are summarized in Table 4.3. In order to get

Dataset	Level	Mapping	Significant units	FDR
Set 1	Nodes	Gene-based	445	0,005
Set 1	Nodes	Protein-based	462	0,005
Set 1	Pathways	Gene-based	106	0,005
Set 1	Pathways	Protein-based	96	0,005
GSE10784	Nodes	Gene-based	21	0,006
GSE10784	Nodes	Protein-based	20	0,006
GSE10784	Pathways	Gene-based	14	0,005
GSE10784	Pathways	Protein-based	29	0,005
GSE12413	Nodes	Gene-based	163	0,001
GSE12413	Nodes	Protein-based	149	0,001
GSE12413	Pathways	Gene-based	77	0,001
GSE12413	Pathways	Protein-based	82	0,001

Table 4.3: Summary of the SAM analysis

comparable values, I tried to maintain the same FDR within corresponding rows, but because of rounding, there is a certain tolerance interval around the numbers of significant units, and therefore they cannot be precisely compared when close to each other. Still, it is obvious that the values are relatively similar, and thus the opening question can be answered as follows: numbers of significantly differentiated units for the representation built upon protein-based mapping consistently come up to those for the gene-based one. Moreover, visual inspection of SAM plots reveals that protein-based representation often gives several extremely discriminative units, even if no such units are present in the gene-based one. Those could be of use when attempting to biologically interpret the obtained results.

4.7 Results summary

I used the Affymetrix BLAST annotation file for the GPL1261 chip to assign probesets to proteins. Subsequently, a mapping to KEGG nodes and pathways synonymous to the XGENE mapping was created upon the proteins and compared with the gene-based one. The main drawback of the resulting mapping is a decreased number of probesets being mapped to the final entities. Anyway, it shows a higher correlation within the defined units (nodes and pathways), so if the aim of searching for an alternative probeset consolidation is to increase the expression correlation, the protein-based representation is clearly competent. In terms of classification accuracy and number of significantly differentially expressed units, the protein-based mapping proved to be an equally good option to the gene-based one.

Chapter 5

Final notes

The main motivation at the very beginning of my work was to find a way of using the sequence annotations to consolidate probesets so that their correlation increases. While doing the research and looking for articles dealing with the same or similar subject, I stumbled across an article appearing to solve my problem completely and in an elegant way [20]. The authors show what I attempted to prove as well, namely that consolidating probesets on the basis of proteins is in terms of correlation superior to the gene-based consolidation. However, as already mentioned in section 4.4, they used the minimum PCC measure, which is utterly misleading when the distributions of probesets differ between the compared mappings. Despite its obviousness, it took me quite some time, during which I was getting splendid results, to realize this. I guess people tend not to challenge their results if they are in line with what they would like to get.

In the protein-enzyme mapping step, I was trying to minimize the probeset losses. To achieve that, I examined several ways of mapping proteins to enzymes. Aside from the described assignments directly through UniProt, I also tried to map the proteins using the KEGG API, which provides a function `get_enzymes_by_gene()`, as the UniProt records often contain cross-links with KEGG gene identifiers. There were two main reasons why I decided not to use that mapping in the subsequent steps: first, the number of probesets assigned to enzymes was just as low as when

assigned directly through UniProt, and second, I wasn't sure if using the KEGG gene identifiers doesn't actually take me back to the gene level. I also tested the ability of NetAffx to assign EC identifiers to probesets downright without even mapping probesets to proteins, but the correlation of the resulting consolidation was markedly lower than in the other cases. Besides that, it was difficult to find any information about how the probesets were actually mapped to enzymes and I wanted to retain full control over the mapping process.

In spite of having disadvantages of its own, the final protein-based consolidation fulfills the original objective, i.e. to consolidate probesets in such way that they are more correlated than when consolidated upon genes. At the present moment, I don't know if this result will be of any practical relevance, but the protein-based approach might get integrated into the future versions of XGENE. I believe that it has a big potential as it fits to the actual biochemical nature of the processes in the organisms and in the microarrays more correctly than considering probesets to be equivalent with genes.

Bibliography

- [1] A. Bairoch et al. The Universal Protein Resource (UniProt). *Nucleic Acids Research*, 33:154–159, 2005.
- [2] T. Barrett et al. NCBI GEO: mining tens of millions of expression profiles - database and tools update. *Nucleic Acids Research*, 35, 2007.
- [3] B. M. Bolstad, R. A. Irizarry, M. Astrand, and T. P. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, 2003.
- [4] W. Dubitzky, M. Granzow, and D. P. Berrar, editors. *Fundamentals of Data Mining in Genomics and Proteomics*. Springer, 2007.
- [5] M. Holec, F. Železný, J. Kléma, and J. Tolar. Integrating multiple-platform expression data through gene set features. In *Proceedings of ISBRA 2009: The 5th International Symposium on Bioinformatics Research and Applications*, 2009. Accepted for publication.
- [6] T. J. Hubbard et al. Ensembl 2009. *Nucleic Acids Research*, 37, 2009.
- [7] Indipedia. BLAST — Indipedia, A wikipedia for India, 2008. [Online; accessed in May 2009].
- [8] M. Kanehisa and S. Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 28(1), 2000.
- [9] M. Krejník. Předzpracování a klasifikace genomických dat. Master’s thesis, Czech Technical University, 2008.

- [10] H. Li, D. Zhu, and M. Cook. A statistical framework for consolidating 'sibling' probe sets for affymetrix genechip data. *BMC Bioinformatics*, 9:188, 2008.
- [11] G. Liu et al. Netaffx: Affymetrix probesets and annotations. *Nucleic Acids Research*, 31(1), 2003.
- [12] D. Maglott, J. Ostell, K. D. Pruitt, and T. Tatusova. Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Research*, 33, 2005.
- [13] I. Mizrachi. *The NCBI handbook [Internet]*, chapter 1. Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information, 2007.
- [14] A. R. Pico, T. Kelder, M. P. van Iersel, K. Hanspers, B. R. Conklin, et al. WikiPathways: Pathway editing for the people. *PLoS Biology*, 6(7), 2008.
- [15] J. U. Pontius, L. Wagner, and G. D. Schuler. *The NCBI handbook [Internet]*, chapter 21. Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information, 2003.
- [16] K. Pruitt, T. Tatusova, and D. Maglott. *The NCBI handbook [Internet]*, chapter 18. Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information, 2007.
- [17] M. A. Salteri and A. P. Harrison. Interpretation of multiple probe sets mapping to the same gene in affymetrix genechips. *BMC Bioinformatics*, 8:13, 2007.
- [18] V. G. Tusher, R. Tibishirani, and G. Chu. Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences of the United States of America*, 98(9), 2001.
- [19] Wikipedia. DNA microarray — Wikipedia, the free encyclopedia, 2009. [Online; accessed in May 2009].
- [20] H. Yu, F. Wang, K. Tu, L. Xie, Y.-Y. Li, and Y.-X. Li. Transcript-level annotation of affymetrix probesets improves the interpretation of gene expression data. *BMC Bioinformatics*, 8:194, 2007.

Appendix A

List of Software

Bash	Bourne-Again Shell - A system-oriented scripting language. I used it in conjunction with Cygwin to co-ordinate actions of the individual scripts.
PICR	Protein Cross-Reference Service - an on-line tool provided by EBI for finding corresponding protein identifiers in various databases.
Python	An open-source programming language. I also used several freely available third-party extension modules, namely: Psyco - a Python library for enhancing the speed of code execution, NumPy - a scientific computing library (among others containing functions for fast matrix operations) and Statistics for Python. Most of the tasks (computation of correlation, the actual probeset consolidation, etc.) have been accomplished using Python.
R	A statistics-focused scripting language. All figures in this text have been exported from R, plus I used the XGENE scripts for normalization and SAM-analysis provided to me by Mr. Jiří Kléma. These scripts use Bioconductor, a third-party R package containing various bioinformatics-related functions.
Weka	A machine-learning environment. I used it to compute the learning curves.

Besides from what's mentioned here, I used various databases and on-line tools (such as NetAffx) referenced throughout the text.

Appendix B

Contents of the CD

The CD content is divided into the following directories:

data	Sample data (mostly from NCBI GEO) meant to be processed by the scripts.
doc	Additional information related to the directory structure and notes to the actual implementation of the scripts.
latex	L ^A T _E X source codes of this text.
pdf	This text in the pdf format.
scripts	Relevant scripts in Python, Bash and R. There are two subdirectories containing the mapping-related scripts and the scripts used for correlation measurement, automated classification and SAM analysis.