

Development of Fast Meta-Analysis Method for RNA-Seq, and its Application to the Analysis of Conserved Coexpression Gene Network

著者	Okamura Yasunobu
学位授与機関	Tohoku University
学位授与番号	11301甲第17088号
URL	http://hdl.handle.net/10097/64048

Doctoral thesis

Development of Fast Meta-Analysis Method
for RNA-Seq, and its Application to the
Analysis of Conserved Coexpression Gene
Network

RNA-Seq を用いた高速メタ解析手法の開発および
遺伝子保存共発現ネットワーク解析への応用

Department of Applied Information Sciences
Graduate School of Information Sciences
Tohoku University

Yasunobu OKAMURA

Submitted in January 2016
Copyright ©2016 Yasunobu Okamura

Contents

Abstract	xi
General Introduction	1
Information science in the biology	1
Importance of database in the biology	2
I The development of ultra fast RNA-Seq analysis method	5
1 Introduction	7
2 Materials and Methods	11
2.1 Preparation	11
2.2 Quantification	12
2.3 Implementation	14
2.4 Comparison with other softwares	16
2.5 Test Dataset	17
3 Result & Discussion	19
3.1 Statistics of indexed N-grams	19
3.1.1 The number of genes with indexed N-grams	19
3.1.2 The distribution of indexed N-grams in transcript sequences in human	20
3.2 Quantification quality	24

3.2.1	Comparison of FPKM	24
3.2.2	Comparison of mapping rate	27
3.2.3	Compare quantification quality with synthesis data	30
3.3	Comparison of CPU time and memory usage	30
4	Conclusion	37
II	The development of massive analysis method for large RNA-Seq dataset	39
5	Introduction	41
6	Materials and Methods	43
6.1	Downloading and managing SRA files	43
6.2	Estimate gene expression level	44
6.3	Calculation of gene coexpression	44
6.3.1	Normalize expression data	44
6.3.2	Calculation of gene coexpression	45
6.3.3	Implementation of gene coexpression calculation	46
6.3.4	Evaluation of gene coexpression by using Gene Ontology	46
7	Results & Discussion	51
7.1	Statistics of SRA files	51
7.2	Gene coexpression	53
7.2.1	Comparison of normalizaiton factor	53
7.2.2	Comparison with microarray-based coexpression	53
7.2.3	Effect of sample size	54
7.2.4	Availability	54

8 Conclusion	59
III Comparison of Gene Coexpression Profiles and Construction of Conserved Gene Networks to Find Functional Modules	61
9 Introduction	65
10 Results	69
10.1 Patterns of coexpression conservation	69
10.2 Identification of conserved coexpressed genes	71
10.3 Conserved gene network in human	72
10.4 Effect of the introduction of conservation	76
10.4.1 Comparison of coexpression conservation between species	78
10.4.2 Implementation of web-based database	81
11 Material and Methods	83
11.1 Dataset	83
11.2 Detection of turning point and conserved coexpression genes	83
11.3 Definition of COXSIM	85
11.4 Analysis of the gene network and module detection	86
12 Conclusion	87
IV The development of large dataset analysis helper tools	89
13 Hyokai : A fast table viewer for big data analysis	91
13.1 Introduction	91
13.2 Implementation	91
13.3 Features	92

13.4 Result	92
13.5 Availability	93
13.6 Conclusion	93
14 DEG.js : A web-based RNA-Seq Analysis Tool	97
14.1 Introduction	97
14.2 Implementation	97
14.3 Result	98
14.4 Conclusion	98
Conclusion	101
References	103
Publications & Presentations	115
Acknowledgment	119
V Appendix	121
A Supplementary Figures for Matataki	123
A.1 Mapping result detail of SRR1639212	123
A.2 Mapping result detail of ERR266335	123
A.3 Mapping result detail of SRR1013361	123

List of Figures

2.1	Example of N-grams that are unique to genes	13
2.2	Example of common N-grams	13
2.3	Mapping example	15
3.1	Coverage of genes with indexed N-grams	21
3.2	Coverage of base pairs with indexed N-grams	21
3.3	Coverage of base pairs with indexed N-grams in human and theoretical random coverage	22
3.4	Coverage of base pairs with indexed N-grams in mouse and theoretical random coverage	22
3.5	Coverage of base pairs with indexed N-grams in <i>Arabidopsis thaliana</i> and theoretical random coverage	22
3.6	Nucleotide coverage by genes	25
3.7	The number of cover islands	25
3.8	Length of cover islands	26
3.9	The length of the longest and second longest cover islands for each gene	26
3.10	Comparison of FPKM when N was varied	28
3.11	Comparison of FPKM when step-size was varied	28
3.12	Comparison of FPKM when accept-count was varied	29
3.13	Mapping rate	31

3.14	Comparison of TPM among expected result and estimated result in a synthesis data result	32
3.15	Summary of synthesis data result	33
3.16	CPU time of methods	36
6.1	Overview of gene coexpression calculation	47
6.2	Evaluate gene coexpression with Gene Ontology	49
6.3	Description of ROC curve and AUC, partial AUC	50
7.1	Effect of sample sizes to coexpression quality	55
10.1	Overview of the conservation calculation method	70
10.2	Detected gene networks	74
10.3	Comparison between the conserved coexpression-based modules and those based on coexpression without conservation	79
10.4	Example of the correspondence between the conservation-based method modules and the COXPRESdb-based modules	79
10.5	Dendrogram of coexpression similarity	80
10.6	How to use COXSIMdb	82
13.1	Hyokai Screen Shot	93
13.2	JOIN SQL Wizard	94
13.3	JOIN SQL Result	94
13.4	Filtering Rows	95
14.1	FASTQ Selection	98
14.2	A result page of DEG.js	99
A.1	SRR1639212: A distribution of FPKM and mapping rate	124
A.2	SRR1639212: Comparison with eXpress when $M = 1$	125

A.3	SRR1639212: Comparison with eXpress when $M = 2$	126
A.4	SRR1639212: Comparison with eXpress when $M = 3$	127
A.5	SRR1639212: Comparison with eXpress when $M = 4$	128
A.6	ERR266335: A distribution of FPKM and mapping rate	129
A.7	ERR266335: Comparison with eXpress when $M = 1$	130
A.8	ERR266335: Comparison with eXpress when $M = 2$	131
A.9	ERR266335: Comparison with eXpress when $M = 3$	132
A.10	ERR266335: Comparison with eXpress when $M = 4$	133
A.11	SRR1013361: A distribution of FPKM and mapping rate	134
A.12	SRR1013361: Comparison with eXpress when $M = 1$	135
A.13	SRR1013361: Comparison with eXpress when $M = 2$	136
A.14	SRR1013361: Comparison with eXpress when $M = 3$	137
A.15	SRR1013361: Comparison with eXpress when $M = 4$	138

List of Tables

2.1	The data format of the hash table	16
3.1	The number of indexed N-grams and genes with indexed N-grams in human genes	23
3.2	A detail of uncovered genes	24
3.3	Run and mapping statistics	35
3.4	CPU Time comparison	35
3.5	Acceleration rate compared with existing methods	35
3.6	Memory usage comparison	35
6.1	Parameters to estimate gene expression	44
7.1	The number of RNA-Seq runs in Short Read Archive	52
7.2	GO prediction pAUC of <i>Homo sapiens</i>	56
7.3	GO prediction pAUC of <i>Mus musculus</i>	56
7.4	GO prediction pAUC of <i>Rattus norvegicus</i>	56
7.5	GO prediction pAUC of <i>Danio rerio</i>	56
7.6	GO prediction pAUC of <i>Drosophila melanogaster</i>	57
10.1	Detected gene modules Summary of detected gene modules and representative GO terms when SCS = 4	72

Abstract

Recently, the number of RNA-Seq data registered in public repository is rapidly increasing due to spreading of high throughput sequencing technology. Reanalyzing of these data is promising approach to reveal gene modules or pathways. Although meta-analysis of gene expression data was widely accepted in microarray data, meta-analysis in RNA-Seq is not performed widely. Since reanalyzing RNA-Seq requires a lot of computational resource, it is nearly impossible to calculate gene expression of all samples in public repository.

In this study, I proposed a novel method to estimate gene expression level from RNA-Seq data rapidly. My proposing method uses N-grams that are unique to each gene to map fragments to genes. Since aligning fragments to reference sequences requires high computational cost, my method reduced calculation cost by using two methods: using only N-grams that are unique to each gene and skipping uninformative region. As a result, my proposing method outperformed previous methods in speed and accuracy.

I applied this method to RNA-Seq data of *Homo sapiens*, *Mus musculus*, *Rattus norvegicus*, *Danio rerio* and *Drosophila melanogaster* for RNA-Seq based meta-analysis. I calculated gene coexpression from estimated gene expression level using a proposed method. As a result, RNA-Seq based gene coexpression outperformed microarray based gene coexpression in predicting gene functions in *Homo sapiens* and *Mus musculus*. Since numbers of samples is highly correlated with performance of gene function prediction, RNA-Seq based coexpression of other species will outperform microarray based coexpression in the future.

Finally, I compared gene coexpression among species. I developed a novel method

to compare gene coexpression. In human and mouse comparison, my method predicted functional gene modules more accurately than human coexpression. I also compared 11 species coexpression and a result showed that a similarity dendrogram of coexpression was consistent with NCBI taxonomy in mammalian.

I also developed software to handle large dataset. *Hyokai* is a large table viewer to summarize and narrow down. It can handle large table, such as a data with more than 1000,000 rows, quickly. *DEG.js* is a web-based RNA-Seq calculation tool for biologists. It is not required installing and using command prompt.

In conclusion, I developed fast RNA-Seq analysis method for meta-analysis, and applied it for gene coexpression and predicting gene functional module. As the result, my methods succeeded to estimate gene expression fast and accurately, and predicted gene function correctly.

General Introduction

Information science in the biology

Importance of information science in the biology is increasing due to growing of biological data. In sequence analysis field, computational methods are essential. After the development of Sanger sequencing [1], a number of nucleotide sequences were published. In 1977, the first complete DNA genome of DNA virus was reported [2]. In 1995, the first complete sequence of free-living organism was reported by Fleischmann *et al.* [3]. In the beginning of 2000s, a lot of complete genomes of model organisms, such as *Drosophila melanogaster* [4], *Arabidopsis thaliana* [5], *Homo sapiens* [6, 7], *Schizosaccharomyces pombe* [8] and *Mus musculus* [9] were published. Currently, the numbers of sequenced data in GenBank [10], European Nucleotide Archive [11], DNA Data Bank of Japan [12] continue to be increasing rapidly.

While increasing the size of biological data, computational methods to deal with these biological data were also developed. In the sequence analysis, FASTA [13] and BLAST [14] are part of the largest impact software in sequence analysis. FASTA, BLAST and related software enabled fast searching of similar sequences in the database.

Development of computational analysis methods also changed experimental methods. R.Staden [15] proposed shotgun sequencing to sequence whole genome without restricting maps. It was impossible to assemble many random sequenced fragments for human, but the development of computers enabled assembling these fragments. Shotgun sequencing accelerated decoding whole genome, and resulted first free-living organism genome [3].

Appearance of high throughput sequencing technology was an evolution in sequencing technology. High throughput sequencing technology enabled sequencing a lot of DNA and RNA with low cost. It changed approaches to biological questions. In 2004, 454 Life Science appeared in the market. From 2005, the number of high throughput sequencing platforms, such as Solexa, Illumina, Ion Torrent, were released [16]. Illumina platforms are the most succeeded platform now. The cost of sequencing was also dropping down steadily.

Because of development of high throughput sequencing technology, the sequenced data is also increasing rapidly. In 2007, NCBI started Sequence Read Archive [17]. Currently, DDBJ Sequence Read Archive [18], NCBI Sequence Read Archive [17] and EBI Sequence Read Archive [11] are collecting raw data of high throughput sequencing under International Nucleotide Sequence Database Collaboration [19].

Since high throughput sequencing technology creates a huge amount of data, a new computational analysis method is required. BWA [20, 21] and Bowtie [22, 23] are widely accepted mapping tools for high throughput sequencing. They use Burrows-Wheeler transform [24] and FM-Index [25, 26] to accelerate searching positions of reads in the genome. Burrows-Wheeler transform was introduced in computer science first. BWA and Bowtie bring this algorithm into biology.

High throughput experimental methods were also proposed in other biological field, such as gene expression using microarray or protein-protein interaction. To deal with these large amounts of data, information science becomes more important in biology.

Importance of database in the biology

Growth of biological database enabled us to take a different approach to biology. A classical approach in biology was proposing a hypothesis, designing an experiment and discussing a result. With biological database, we can use data-driven approach. Data-driven approach gives us comprehensive view of biological knowledge.

When we publish a new paper, authors are obliged to register raw data, such as sequences from high throughput sequencer in most journals' policy. For example, PLOS biology requires registering all data to public repository and putting accession numbers in a paper [27]. Therefore, almost all raw data in published papers, such as sequence data, protein structures or gene expression profiles were registered in public databases.

Meta-analysis of these data enhances detection power by merging data from many studies. For example, Sitras *et al.* [28] analyzed microarray data from 12 studies and found common pathway between two diseases. Kim *et al.* [29] reported 18 genes related with toxicity based on meta-analysis of massive gene expression profiles.

Re-analyzing published data also spawned the new aspect of biology. Gene coexpression is one of most successful analysis of re-analyzing database. Gene coexpression data show relationships between genes without any prior knowledge. Since thousands of gene expression data are required to calculate gene coexpression, database of gene expression is indispensable.

Because of development of high throughput sequencing and growth of public database, it is challenging performing meta-analysis of these data. Computational analysis with efficient algorithm is fundamental to deal with large amount of data.

In this study, I performed a meta-analysis of a large amount of RNA-Seq data. Since RNA-Seq data were generated by high throughput sequencer, the total data size of RNA-Seq data is larger than 50 TB. Existing methods are too slow to analyze these data. In Part I, I developed a novel method to quantify gene expression in RNA-Seq. The proposing method is 300 times faster than widely used methods, and 2.5 times faster than published fastest method. In Part II, I calculated gene coexpression based on RNA-Seq. The performance of RNA-Seq data based gene coexpression was better than microarray data based gene coexpression in Gene Ontology prediction. In Part III, I compared gene coexpression between species and showed conservation of gene relationship between human and mouse. I also created tools to deal with such a large amount of data, as described in Part IV.

Part I

The development of ultra fast RNA-Seq analysis method

Chapter 1

Introduction

Measuring gene expression is important to identify genes that work on a specific event or interpret a cell state. Historically, gene expression is measured by northern blotting. Northern blotting can detect the size of RNA and expression level [30, 31]. This method is widely accepted and remains useful today. Since this method cannot detect many types of mRNA at once, comprehensive study of gene expression requires other methods. In 1995, an appearance of microarray [32] technology enables high throughput, comprehensive measurement of RNA. However, accuracy of expression measurement for lowly expressed genes is limited. Dynamic range of microarray is also limited [33].

RNA-Seq [34] technology appeared nearly a decade ago. The development of high throughput sequencing technology enabled RNA-Seq. RNA-Seq has a lot of advantage in measuring gene expression, especially in accuracy and dynamic range [35]. Now, RNA-Seq is a *de facto* standard of RNA analysis, and the number of RNA-Seq that are registered in Short Read Archive is rapidly increasing. Although this technology also enabled to find *de novo* transcripts or SNP analysis, I focused on gene expression quantification in this part.

Since the number of published RNA-Seq data is rapidly increasing, meta-analysis of these data is a promising approach to investigate novel biological system. However, merging quantified expression data provided by authors is difficult because they use different reference sequences, different ID system and different quantification methods. Using different reference

sequences or difference ID system makes difficult matching genes with other data. Comparing gene expression profiles quantified by different methods cannot separate biological differences from method bias. Therefore, quantifying from raw sequence for all data is required to meta-analysis of RNA-seq.

A lot of quantification methods for RNA-Seq were proposed. One of the most used methods is the pipeline using TopHat2 [36, 37] and cufflinks [38]. This method aligns sequenced reads to a reference genome, then, it counts the number of fragments that are mapped in a gene region, and estimates expression level by transcript level. This method can be applied to species that have no reference transcript, and predict transcript candidates.

Some other methods, such as RSEM [39] and eXpress [40], map sequences to the transcript reference. Since they require only reference transcript sequences, they can be applied to species without reference genome. A de novo transcript assembler or an EST database can be used for reference transcript sequences instead of curated reference transcript databases. Both RSEM and eXpress use bowtie [22] to map a read sequence to a transcript. Some of read sequences are mapped to multiple transcripts due to splicing variants. RSEM and eXpress use Expectation-Maximization algorithm to resolve which of transcripts were multi-mapped reads come from.

These alignment-based methods, such as TopHat2/cufflinks, RSEM or eXpress require a lot of computational resource. To quantify expression of RNA-Seq sample, alignment is not required because a position of a read is not important in quantification step. Some methods do not map to transcript, but use N-gram of transcripts.

Sailfish [41] uses all N-grams found in the reference transcript. This method creates an N-gram to containing transcript table, and counts the number of occurrence in RNA-Seq data for each N-gram. Finally, it estimates expression level by transcripts using Expectation-Maximization algorithm from counts of N-grams.

Another method, RNA-Skim [42], uses more efficient method. This method introduced

sig-mers that appear only once in a subset of reference transcript. RNA-Skim counts the number of *sig-mers* occurrence while processing RNA-Seq data, and calculates expression level by Expectation-Maximization algorithm.

Kallisto [43] also uses N-grams. This method reduced calculation cost by skipping fragment searching in an index. When an N-gram appeared, the next N-gram is limited to one or few patterns. If the next N-gram is limited to one pattern, hashing the N-gram is not required to determine the source isoform. Kallisto skips these non-informative N-grams for fast estimation.

The speed of quantification is important to process thousands of RNA-Seq data. Although these alignment-free methods, such as Sailfish and RNA-Skim, are much faster than alignment based method, a faster method is needed to perform large-scale meta-analysis.

In addition, gene level quantification has enough information for usual analysis because many studies [44, 45, 46, 47] disregard isoform specific expression. Some studies [48, 49] regard differential usage of isoforms, but they analyzed few numbers of splicing changes. For example, Wu *et al.* [48] perform gene level quantification for all genes first, and isoform level quantification next.

Here, I proposed *Matataki*, the novel fast method to quantify expression in gene level. Similar to RNA-Skim, this method uses N-grams that are unique to gene to quantify expression. However, this method reduces computational cost with two different approaches. First, this method can calculate expression directly without Expectation-Maximization algorithm because this method quantifies expression in gene level, and uses only gene specific N-grams. Second, this method does not hash a fragment step by step. Since N-grams that are unique to gene are usually found sequentially, hashing all fragments of a read does not improve performance. This method hashes a fragment of reads every fixed count. To skip hashing, exactly all N-grams that are unique to a gene should be listed up in the index. Therefore, fast heuristic methods, such as bloom filter [50] cannot be applied to build index. However, in

the large-scale meta-analysis, the speed of quantification is more important than the speed of building index.

In this part, I describe the method of Matataki and the result of comparisons between my method and other methods.

Chapter 2

Materials and Methods

I developed an ultra fast RNA-Seq quantification method based on N-grams that are unique to each gene. Usually, this method requires two steps: building an index and quantifying expressions. In this section, all running time and memory usage were measured in cluster machines. Each cluster node has two Intel® Xeon® CPU E5-2680 v2 10-core 2.80GHz, and 130 GB RAM.

2.1 Preparation

In order to fast mapping, Matataki has to search all N-grams that are unique to each gene. Selected N-grams should be included in all isoform transcripts of a gene to avoid effects of differential expression of isoforms.

First, Matataki searches N-grams that are unique to each gene (Figure 2.1). Matataki considers all N-grams in transcript sequences. To judge uniqueness of N-gram, Matataki stores N-grams to a hash table. For example, first five 20-grams are unique to CARNS1, but next three 20-grams are also found in DNER (shown in Figure 2.1). Therefore, 20-grams unique to CARNS1 can be used for identifier of CARNS1, but 20-grams that are not unique to CARNS1, such as GCCACTGCCACCGCCGCGC, cannot be identifiers. Since a read strand is not fixed unless strand-specific read, all reverse complements of N-grams should be considered.

Second, Matataki checks whether all isoforms of a gene have an N-gram. Since Matataki

quantifies expression levels by genes, alternation of isoform specific expression should be ignored. When isoform A has an N-gram and other isoforms do not have the N-gram, and expression level of isoform A was replaced to the other isoform, the found reads appear to be decreased, although a total gene expression level was not changed. To avoid this isoform replace problem, N-grams that do not appear in all isoforms should be neglected. For example, a sequence `NM_001193533.1` is one isoform of `NEK4` (shown in Figure 2.2). In this sequence, a first 20-gram is unique to `NEK4`, but this sequence is not found in the isoform `XM_006713310.1`. Matataki index N-grams that are unique to each gene and found in all isoforms.

Finally, Matataki counts the numbers of indexed N-grams. This number will be used in the FPKM quantification step.

Around 20 GB memory and 3 hours are required to build human index, when I used in-memory hash table. A file based hash table mode is also available for small memory machines. This building step is required only one time for one species before using Matataki.

2.2 Quantification

Quantification step has two sub-steps. The first step is counting N-grams, and the second step is calculating FPKM and TPM from read counts.

First, Matataki searches indexed N-grams in a read. When a read has only one corresponding gene, the read is estimated as a fragment of the gene. Matataki counts the number of reads that corresponds to each gene. An example is shown in Figure 2.3 (A). In this example, first six 20-grams were unique to `SMYD1`. On the other hand, next twenty 20-grams were not found in the index because the read has a mutation (shown as red A in 2.3 (A)). When a read corresponds to two or more genes, or no genes, the read will be neglected.

In the first step, searching all fragments of reads step by step is not necessary, because the found N-grams are usually found sequentially. For example, Figure 2.3 (B) shows which

CARNS1 (carnosine synthase 1) NM_001166222.1

Sequence

GCTGTGCCACTGCCACCGCCGCCGCCG...

20-gram

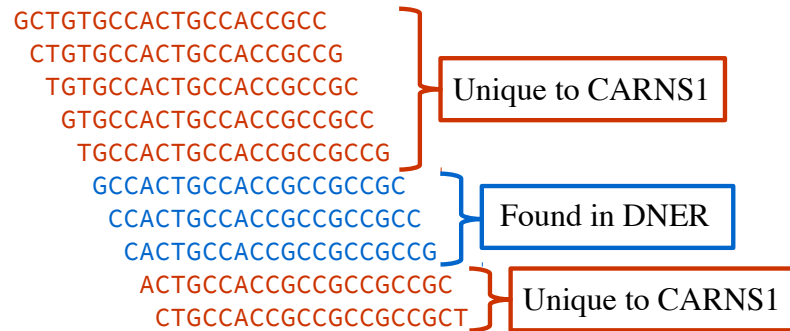


Figure 2.1: Example of N-grams that are unique to genes

NEK4 (NIMA-related kinase 4) NM_001193533.1

Sequence

AGCATGCGCAGAACTGCTCCCGGCC

20-gram

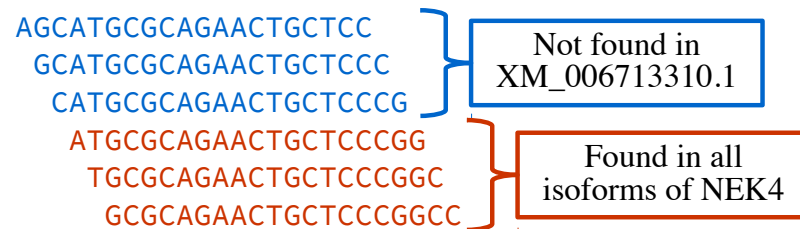


Figure 2.2: Example of common N-grams

N-grams were found in the index. The second row shows a search result of the read. The character “0” means a 20-gram from the position was a 20-gram that is unique to SMYD1, and “.” means a 20-gram from the position was not found in the index.

I introduced “step-size S ” to reduce the number of N-gram searching. As shown in Figure 2.3(C), Matataki searches N-grams for every S N-grams. Since gene specific reads have sequences of index N-grams usually, this omission does not have major effect to the estimation quality.

I also introduced “accept-count M ” to avoid a fragment of read sequence matches to some indexed N-grams by chance. Since some reads may have sequencing error, mutation or insertion/deletion, a fragment of a read can be matched to wrong indexed N-grams. Usually, these wrong matches are not found consecutively in a read. Therefore, a read that indexed N-grams appeared in less than M times deal with unmapped read to avoid wrong matching.

Second, Matataki calculated FPKM (Fragment Per Kilobase of Million) and TPM (Transcript Per Million) from gene specific read counts. FPKM can be calculated from formula 2.1 and TPM can be calculated from formula 2.2.

$$F_i = \frac{C_i/K_i}{\sum_j C_j} 10^9 \quad (2.1)$$

$$T_i = \frac{C_i/K_i}{\sum_j (C_j/K_j)} 10^6 \quad (2.2)$$

, where F_i : FPKM, T_i : TPM, C_i : a count of gene-specific reads, K_i : the number of indexed N-grams in a gene. Since Matataki uses only gene specific N-grams, no EM algorithm or other algorithm to solve expression level are not required.

2.3 Implementation

I implemented Matataki with C++03, autotools and KyotoCabinet [51]. Since simplicity of installing is important to distribute, a distribution file contains all libraries to compile, and

uses autotools, a standard tools to build complex software. Automated unit-testing is also important to maintain the quality of codes. Matataki uses Google Test for unit testing and integrated it to autotools, therefore, we have to run nothing but `make check` to run all unit tests.

In order to reduce memory usage and increase speed, a hash table format is optimized for the RNA/DNA N-grams. First 4k bytes contain a header of an index. The number of entries, the size of hash table and N are written in the header. After the header, N-grams and corresponding gene indexes are written. Each entry have two sections, a gene index and N-grams (shown in Table 2.1). N-gram is compressed as two bit representation of acids to reduce memory usage and hash value calculation time. Since each N-gram has fixed length in one index, entries do not have data of length. A hash function is also important for fast looking up of the table. I used MurMurHash3 as a hash function of the hash table because it is fast and widely accepted hash function.

Since all libraries except installed in almost all systems are included in the distribution file, no extra libraries are required to build and run this method. To make install easy, the built binaries are path independent. No special file layout or PATH environment is required to run.

Table 2.1: The data format of the hash table

Entry	Length
Gene Index	4 bytes
N-gram	Variable (compressed in two bit representation, and aligned with 4 bytes)

2.4 Comparison with other softwares

I compared performances with bowtie 1.1.2 [22]/eXpress 1.5.1 [40], Sailfish 0.7.6 [41], RNA-Skim [42] and Kallisto 0.42.4 [43]. I tested these softwares with default parameters. I used binary distributed files for bowtie, eXpress, Sailfish. RNA-Skim and our method are

compiled with GCC 5.2.0.

2.5 Test Dataset

I used RefSeq and gene2refseq [17] to create a reference database. RefSeq and gene2refseq were downloaded at June 26, 2015 from Human Genome Center, a mirror site of NCBI. I extracted sequences of human from RefSeq. In the human RefSeq, 25,894 genes and 55,100 transcripts were available.

In comparison with RNA-Skim, I used the scripts to download and build index, which are included in RNA-Skim. Therefore, the reference sequence of RNA-Skim is different from other methods.

For quantification quality examination, I used SRR1639212. This run is part of SRP048993, “Stem cell differentiation timecourse, six time points through induction from induced pluripotency (day0) towards beating cardiomyocytes, mature at day14. ” SRR1639212 is the first day 0 sample. The length of reads in SRR1639212 is 100, and the number of reads is 172,340,634.

I also compared quantification quality with synthesis data. To create synthesis data, I used `rsem-simulate-reads` that is included in RSEM. Models to synthesize were created by quantifying ERR188074 and ERR188125 with RSEM.

Chapter 3

Result & Discussion

3.1 Statistics of indexed N-grams

3.1.1 The number of genes with indexed N-grams

First, I calculated the number of genes with indexed N-grams (shown in Figure 3.2 and Table 3.1) and the nucleotide coverage of indexed N-grams (shown in Figure 3.1 and Table 3.1) in human, mouse and Arabidopsis genes. I varied N-gram length from 10 to 100.

When $N = 10$, few human genes had indexed N-grams in all species. When $N = 14$, 96.8% of human genes in RefSeq had indexed N-grams. Coverage of genes were highest in $N = 34$. On other hand, too large N made gene coverage lower, because some genes had only small transcripts.

When looking at the coverage of nucleotide (shown in Figure 3.1), $N = 14$ was not enough large to cover sequences with indexed N-grams regions. The nucleotide coverage almost hit the ceiling in $N = 18$. This observation suggests that N should be larger than 18 to cover gene-specific regions of genes.

The same results were also observed in mouse and Arabidopsis. Since the average length of genes in Arabidopsis is smaller than lengths in human and mouse, gene coverage and nucleotide coverage in $N = 10$ and 12 were better than other species. I compared nucleotide coverage with theoretical random coverage. The theoretical random coverage was defined as

following formula.

$$C = \left(1 - \frac{1}{(4^N + 4^{(N/2)})/2}\right)^l \frac{l}{L} \quad (3.1)$$

, where C is a nucleotide coverage, L is a total length of genes and l is a total length of gene specific region. The results were consistent with actual coverage (shown in Figure 3.3, 3.4 and 3.5).

The effects to performance of N size will be discussed in the following section.

3.1.2 The distribution of indexed N-grams in transcript sequences in human

I calculated the nucleotide coverage for each human gene when $N = 32$ (shown in Figure 3.6). As a result, the coverage was higher than 50% in 86.4% human genes. Moreover, in 61% genes, the coverage was higher than 90%.

The number of cover islands is shown in Figure 3.7. A cover island is a continuous region of nucleotide that is a start point of indexed N-gram. As a result, 60% of genes have only one or two cover islands. The length of cover islands and the longest and second longest length of cover islands for each gene are shown in Figure 3.8 and 3.9. Although a median of second longest cover island length for each gene is 327, a median of the longest cover island length for each gene is 1,262. These observations suggest that most genes have a few cover islands, namely a main cover island and some small satellite cover islands. Since the longest cover islands for each gene were enough longer than N and step size S , introducing step size S did not interfere quantification accuracy.

When $N = 32$, the number of genes without indexed N-grams was 717. The detail of these uncovered genes is shown in Table 3.2. Half uncovered genes were non-coding genes. Since non-coding genes cannot be amplified in the translation step, the number of copies in genome is required to work properly. Another half of uncovered genes were protein-coding

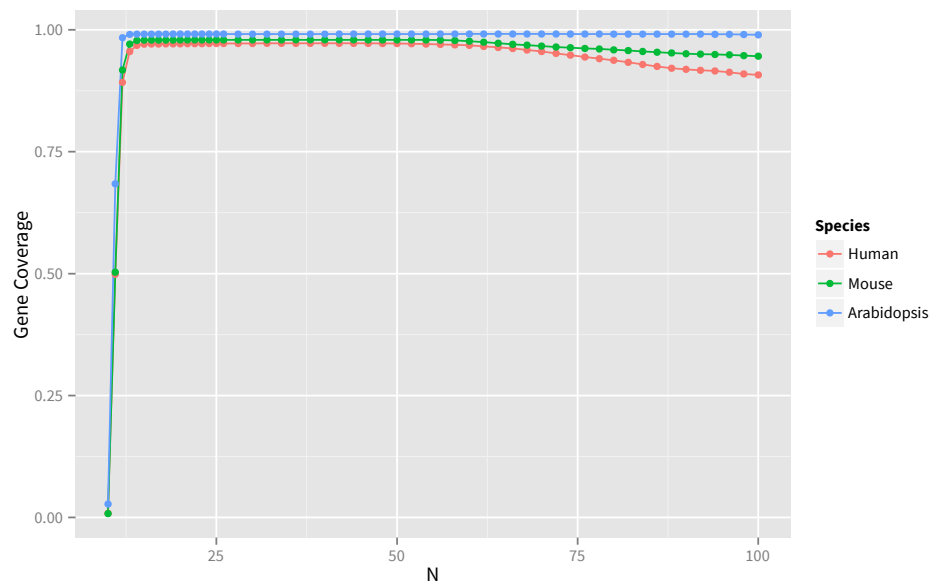


Figure 3.1: Coverage of genes with indexed N-grams

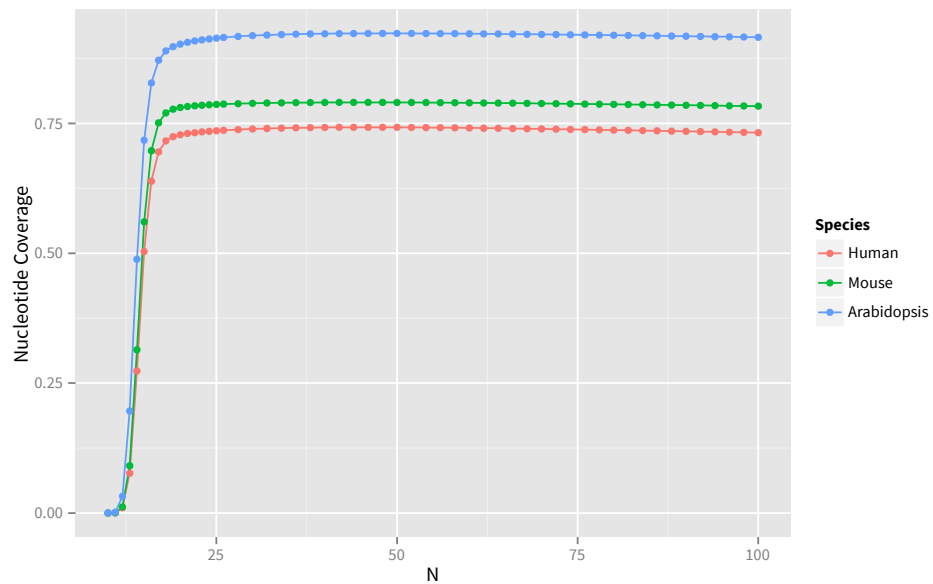


Figure 3.2: Coverage of base pairs with indexed N-grams

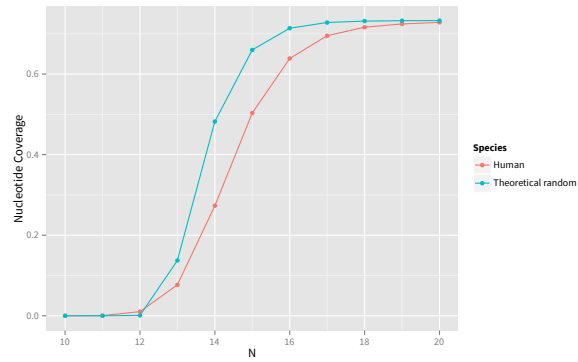


Figure 3.3: Coverage of base pairs with indexed N-grams in human and theoretical random coverage

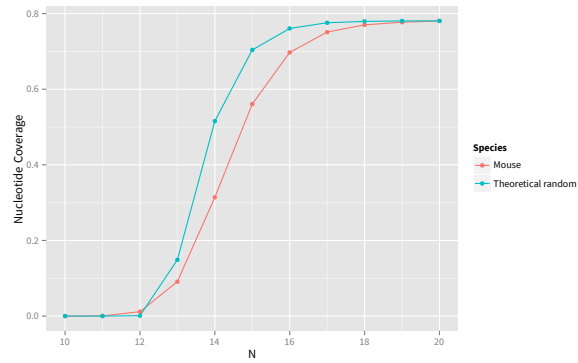


Figure 3.4: Coverage of base pairs with indexed N-grams in mouse and theoretical random coverage

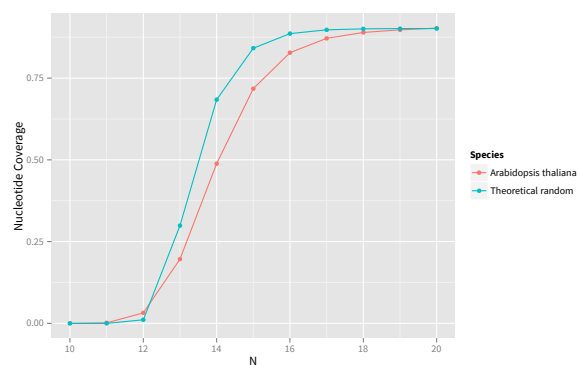


Figure 3.5: Coverage of base pairs with indexed N-grams in *Arabidopsis thaliana* and theoretical random coverage

Table 3.1: The number of indexed N-grams and genes with indexed N-grams in human genes

N	# of indexed N-grams	# of genes	Gene Coverage (%)
10	440	220	0.84
12	1535090	23101	89.21
14	41082576	25071	96.82
16	96190226	25143	97.09
18	108060135	25147	97.11
20	109992844	25153	97.13
22	110684112	25161	97.16
24	111141270	25166	97.18
26	111492136	25167	97.19
28	111769766	25171	97.20
30	111991000	25171	97.20
32	112168261	25177	97.23
34	112310635	25178	97.23
36	112421849	25176	97.22
38	112506091	25174	97.21
40	112567660	25177	97.23
42	112609502	25176	97.22
44	112633994	25176	97.22
46	112642942	25175	97.22
48	112638863	25171	97.20
50	112623384	25164	97.18
52	112597536	25155	97.14
54	112562252	25140	97.08
56	112518444	25119	97.00
58	112467189	25097	96.92
60	112409689	25064	96.79
62	112346699	25010	96.58
64	112278981	24958	96.38
66	112207391	24904	96.17
68	112132326	24820	95.85
70	112054386	24747	95.57
72	111974249	24642	95.16
74	111892097	24545	94.79
76	111808039	24452	94.43
78	111722329	24357	94.06
80	111635162	24273	93.73
82	111546656	24167	93.33
84	111457144	24054	92.89
86	111366760	23949	92.48
88	111275524	23855	92.12
90	111183488	23793	91.88
92	111090670	23745	91.70
94	110997032	23711	91.56
96	110902740	23641	91.29
98	110807946	23548	90.93
100	110712668	23497	90.74

Table 3.2: A detail of uncovered genes

Type of Gene	count
microRNA	233
ribosomal RNA	19
small nuclear RNA	35
small nucleolar RNA	45
Other non-coding RNA	61
Pseudo gene	21
protein-coding	303

genes. Most of these genes do not share functions and genome region. They may share domains with other genes.

3.2 Quantification quality

In this section, I compared the expression levels of three methods with my method using real data and synthesis data. I compared the result of bowtie/eXpress and my method. The accession ID of the test real data is SRR1639212.

3.2.1 Comparison of FPKM

I compared Fragment Per Kilobase Million (FPKM) values between bowtie/eXpress and my method. At first, I varied N from 16 to 56 and the result is shown in Figure 3.10. In Figure 3.10, the x-axis shows FPKM values of eXpress, the y-axis shows FPKM values of my method, and the color means indexed N -gram coverage of each gene. The Pearson Correlation Coefficient values between my method and eXpress were higher than 0.92 when N was larger than 24. Since larger N gives larger PCC value, large N is better to estimate accurately.

However, large N is not always the best choice to analysis. In Short Read Archive, 9.2% of human RNA-Seq data have reads shorter than 50 in lengths. To cover 99% of human RNA-Seq data, N should be smaller than 34. In the following analysis, I used $N = 32$ because Matataki prefers a multiple of 4 as N due to implementation.

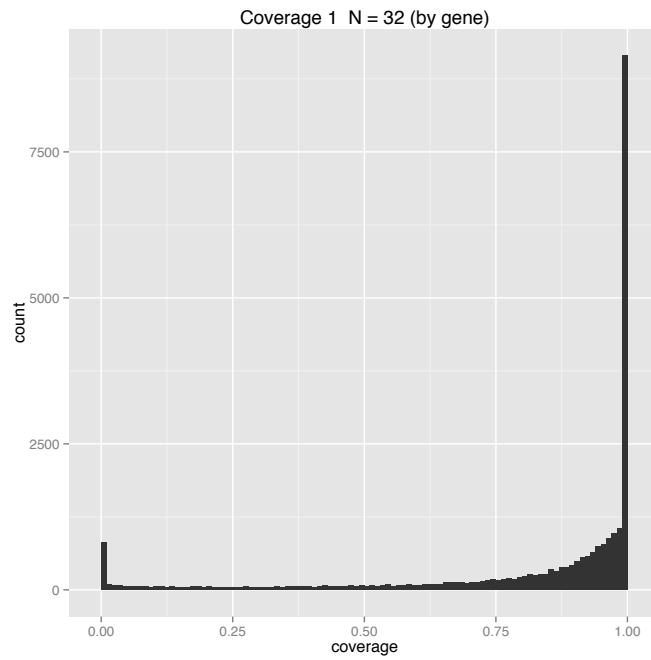


Figure 3.6: Nucleotide coverage by genes

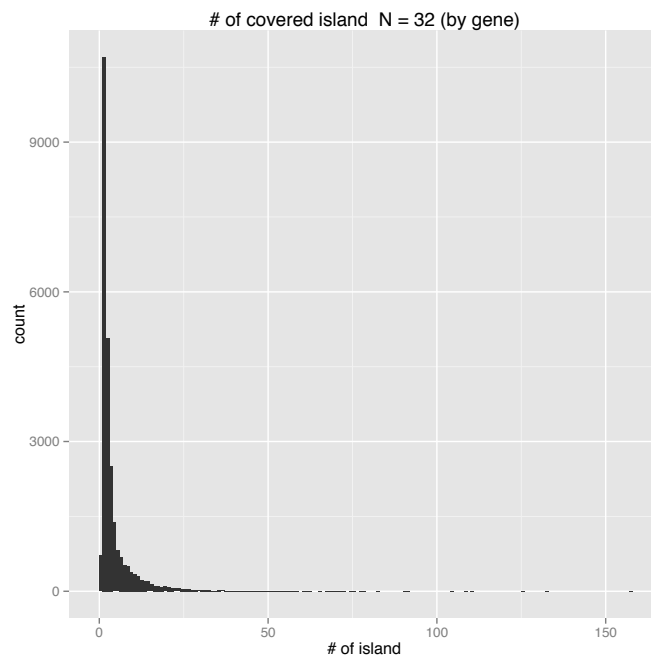


Figure 3.7: The number of cover islands

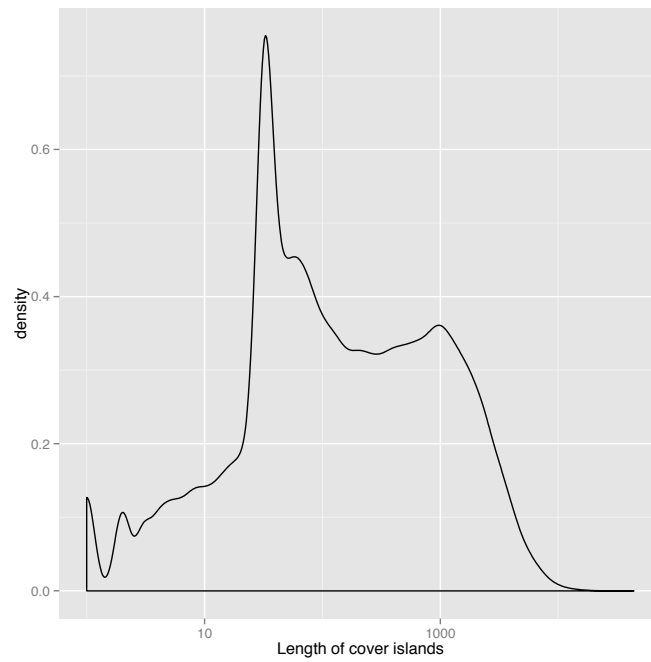


Figure 3.8: Length of cover islands

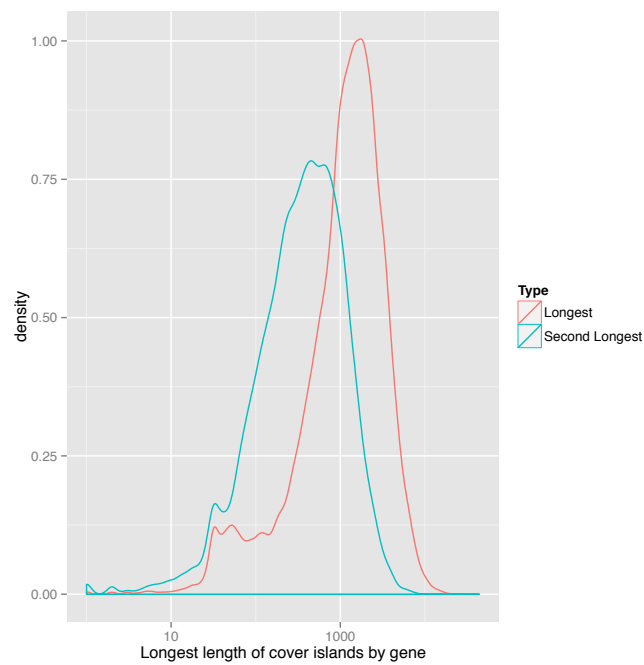


Figure 3.9: The length of the longest and second longest cover islands for each gene

Second, I varied the step size S from 1 to 16 (the result is shown in Figure 3.11). This parameter was introduced because looking up every N -grams in a read provides less information. As a result, larger S shows a better result in PCC. This result suggests some junk reads may have an N -gram that is equal to indexed N -grams by chance. Since junk reads have one or a few numbers of matched N -grams, these larger step size S can reduce invalid matches. In the following analysis, I used $S = 12$ because 12 is accurate enough.

Finally, I varied the accept-count M from 1 to 4 (the result is shown in Figure 3.12). I introduced this parameter to avoid that some fragments in a junk read matches to wrong indexed N -grams by chance. According to Figure 3.12, PCC values of $M > 1$ was better than a PCC value of $M = 1$. This result supports that some reads were counted as wrong genes. On other hand, the PCC value of $M = 4$ was worse than the PCC value of $M = 3$. Since the read length of the test data was 100, a few errors were allowed to accept a read. This observation suggests too strict condition makes the result worse.

Selecting the best combination of N , step size S and accept-count M is one of the problems in this method. The best combination depends on the read length and quality. According to Figure 3.12, some errors should be allowed for accurate quantification.

3.2.2 Comparison of mapping rate

Mapping rate is also an important measure to evaluate this method. I compared mapping rates by varying N , step size S and accept-count M . The result is shown in Figure 3.13. As expected, the mapping rate became smaller when N became large because matching condition was stricter in large N (shown in Figure 3.13 (A)). When $N = 16$, the mapping rate is larger than the rate of bowtie. This observation suggested that Matataki count junk reads as some gene's read by chance.

On other hand, steps size S affected mapping rate slightly (shown in Figure 3.13 (B)). This result indicated N -grams in a read that match to some indexed N -gram are continued.

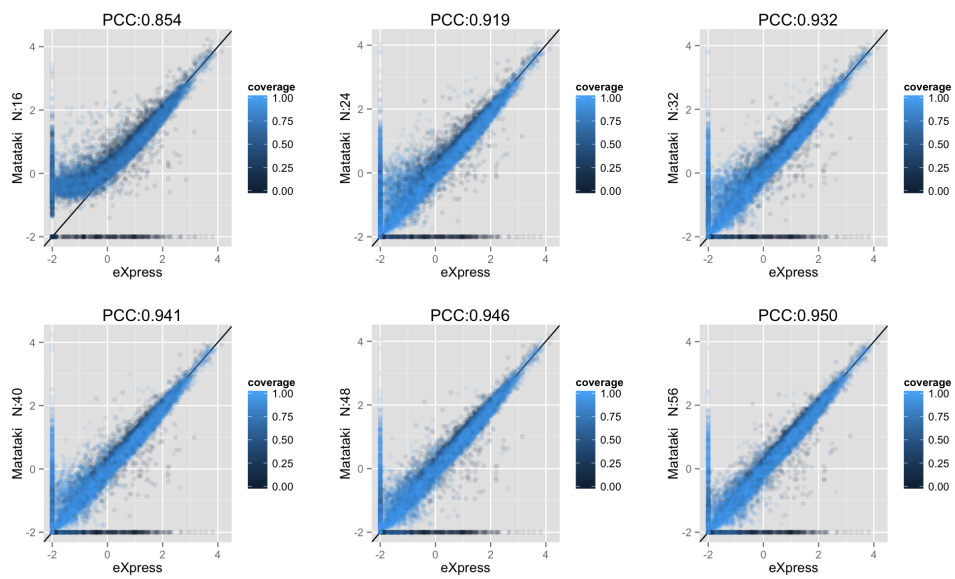


Figure 3.10: Comparison of FPKM when N was varied

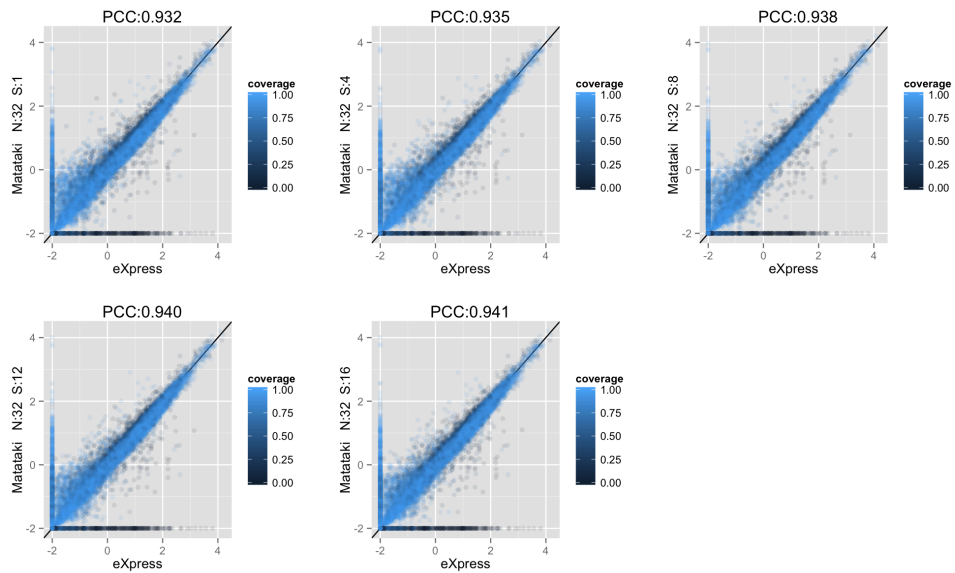


Figure 3.11: Comparison of FPKM when step-size was varied

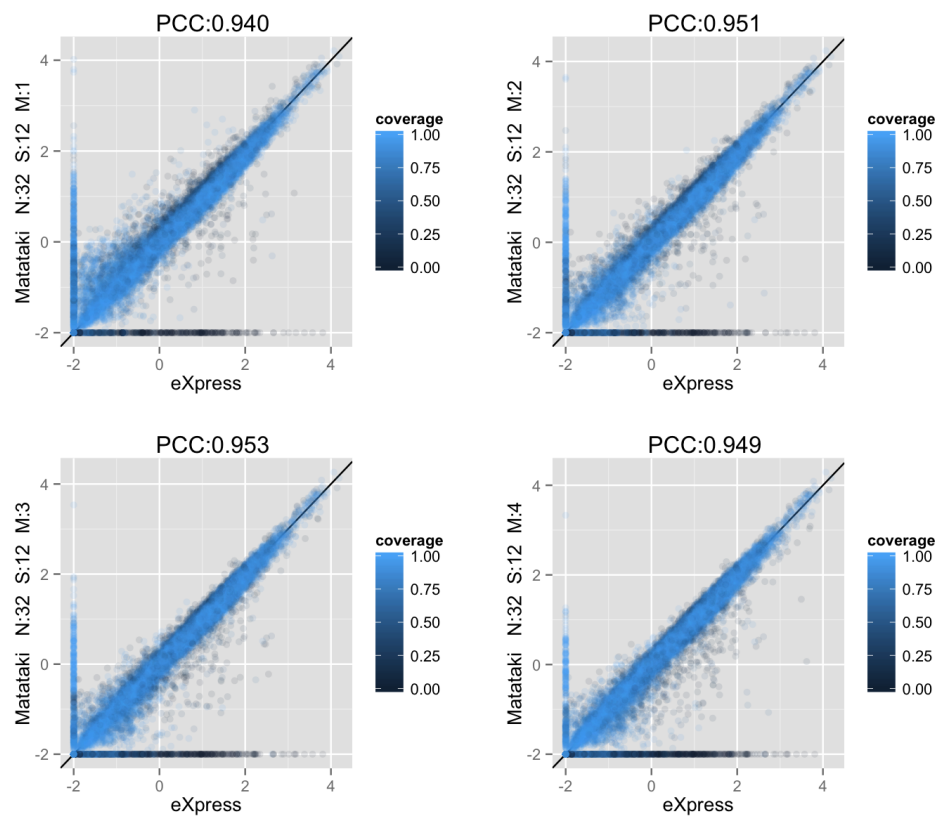


Figure 3.12: Comparison of FPKM when accept-count was varied

Finally, large accept-count M makes mapping rate lower (shown in Figure 3.13 (C)). Especially, the mapping rate dropped largely at $M = 4$ when compared with other M . Therefore, $M = 4$ is too strict in this data.

Other more detail comparison results are shown in Appendix A.

3.2.3 Compare quantification quality with synthesis data

I also compared Transcript Per Million (TPM) between my method, eXpress, Sailfish and Kallisto in synthesis data. In this comparison, I used $N = 32$, $S = 16$ and $M = 2$ as parameters of my method. As a result (shown in Figure 3.14 and Figure 3.15), my method showed second best performance in linearity (i.e. PCC, shown in Figure 3.14 A, C, E, G and Figure 3.15A) and best in error (i.e. absolute mean difference, show in Figure 3.14 B, D, F, H and Figure 3.15B) in alignment free methods. Since an alignment based method, eXpress showed the best performance in both of linearity and error, using eXpress result is the best choice to evaluate prediction performance in real data. In this analysis, I included genes that do not have indexed N-grams. My method cannot estimate expression level of these genes. When I excluded these genes to calculate accuracy, my method showed the best performance in both of linearity and error (shown in Figure 3.14 I, J and Figure 3.15 A, B as MatatakiSubset).

Although my method is fastest, my method is most accurate to estimate gene expression level for each gene in alignment free methods. Using indexed N-grams enables faster and accurate RNA-Seq quantification.

3.3 Comparison of CPU time and memory usage

In this section, I compared the CPU time and memory usage of six existing methods with Matataki in real data. I used four runs, ERR188074, ERR188125, ERR188171 and ERR188362 to compare CPU times and memory usage. The statistics of runs and mapping is shown in

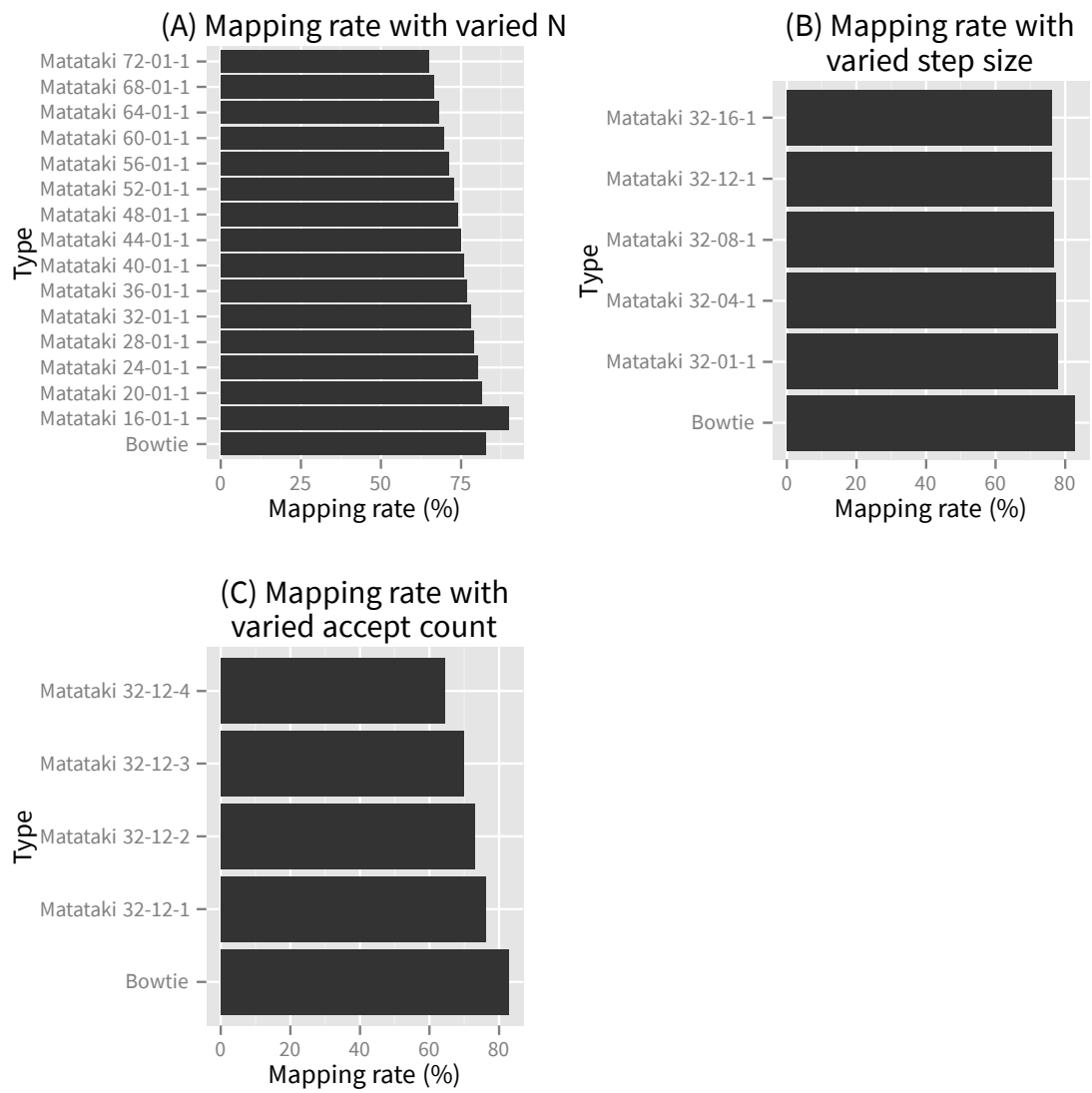


Figure 3.13: Mapping rate

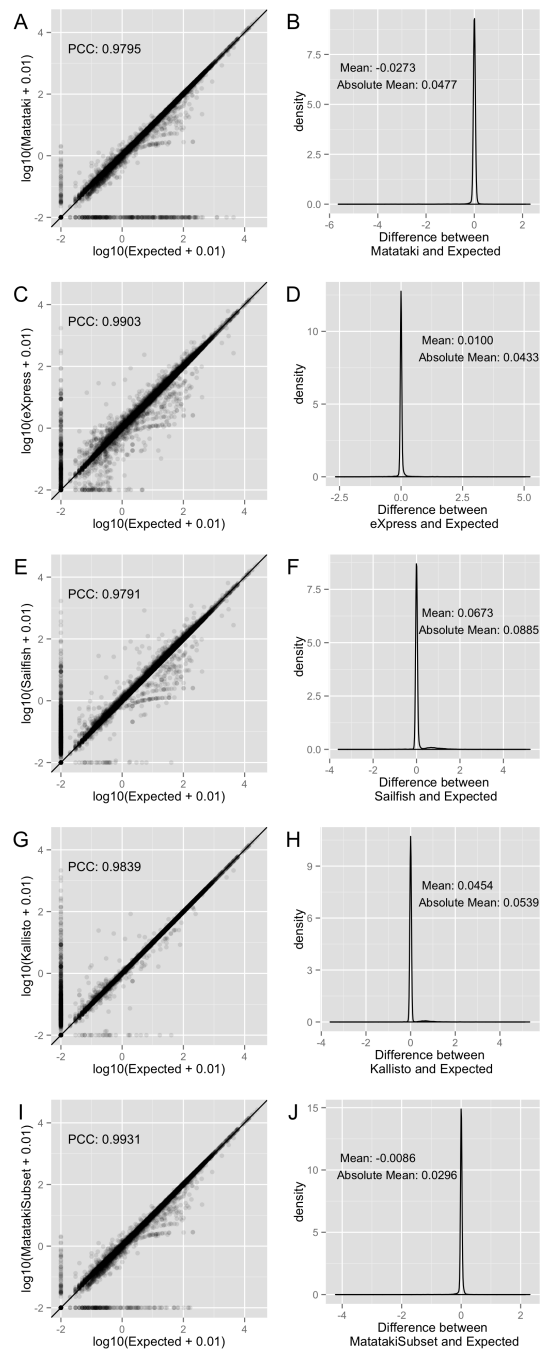


Figure 3.14: Comparison of TPM among expected result and estimated result in a synthesis data result

(A) A scatter plot of expected gene expression and estimated gene expression using proposing method. (B) A density plot of difference between expected gene expression and estimated gene expression using proposing method. (C, D) A scatter plot and a density plot using eXpress. (E, F) A scatter plot and a density plot using Sailfish. (G, H) A scatter plot and a density plot using Kallisto. (I, J) A scatter plot and a density plot using proposing method. In these figures, genes that do not have indexed N-grams are neglected.

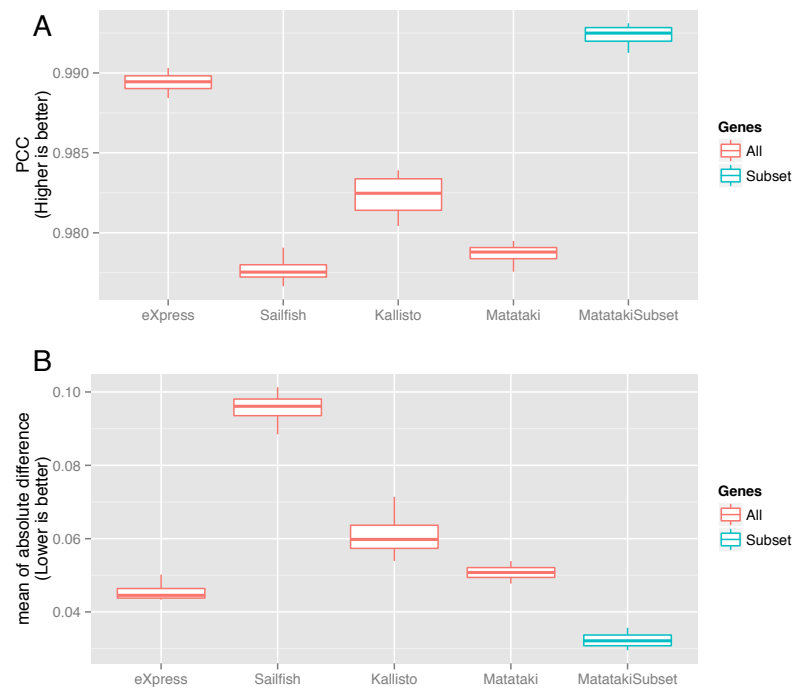


Figure 3.15: Summary of synthesis data result

(A) Pearson Correlation Coefficient with expected expression and estimated expression using each method. “Matataki” is a result of a proposing method and “MatatakiSubset” is a result of the proposing method without uncovered genes. **(B)** Mean of absolute difference from expected expression.

Table 3.3. The results of CPU time are shown in Figure 3.16 and Table 3.4, and the results of acceleration rate compared with existing methods are shown in Table 3.5. The CPU times were measured 10 times and medians of these results are shown in Table 3.4 and Table 3.5. Memory usages are shown in Table 3.6. In this comparison, I used $N = 32$, $S = 12$ and $M = 3$ as the parameters.

As results, my method was extremely faster than other alignment-based methods, Bowtie without quantification, RSEM and eXpress. Matataki was twice faster than other alignment-free methods, Sailfish, RNA-Skim and Kallisto. Since Matataki was even faster than gzip uncompression (about 55 seconds) or bzip2 uncompression (about 285 seconds), a quantification of gene expression in RNA-Seq is not bottleneck of RNA-Seq analysis. In memory usage comparison, Matataki required the smallest memory size in alignment-free methods.

Table 3.3: Run and mapping statistics

Run accession	Number of reads	Length of reads	bowtie mapping rate
ERR188074	31,540,813	75	84.7%
ERR188125	28,810,860	75	80.2%
ERR188171	30,386,179	75	84.6%
ERR188362	26,255,381	75	80.4%

Table 3.4: CPU Time comparison

Run Accession	eXpress	RSEM	Bowtie	Sailfish	RNA-Skim	Kallisto	Matataki
ERR188074	15080.6	22264.9	1477.2	303.3	521.2	119.3	47.9
ERR188125	25404.0	20428.3	1492.4	314.3	489.9	118.7	41.0
ERR188171	14109.1	21815.4	1429.5	333.5	494.2	123.2	38.6
ERR188362	24607.6	18831.7	1355.8	302.8	483.7	102.0	40.6

Table 3.5: Acceleration rate compared with existing methods

Run Accession	eXpress	RSEM	Bowtie	Sailfish	RNA-Skim	Kallisto
ERR188074	314.52	464.35	30.81	6.33	10.87	2.49
ERR188125	618.94	497.72	36.36	7.66	11.94	2.89
ERR188171	365.92	565.78	37.07	8.65	12.82	3.20
ERR188362	606.76	464.34	33.43	7.47	11.93	2.52

Table 3.6: Memory usage comparison

Method	Memory usage (GB)
eXpress	4.0
RSEM	4.0
Bowtie	1.2
Sailfish	6.2
RNA-Skim	12.1
Kallisto	3.8
Matataki	3.5

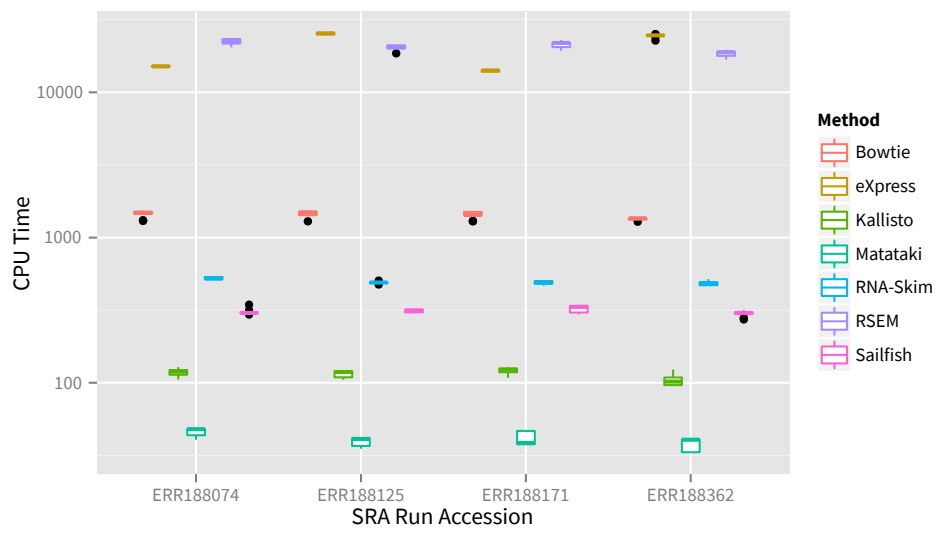


Figure 3.16: CPU time of methods

Chapter 4

Conclusion

Matataki is much faster and user-friendly quantification method for RNA-Seq. My method archived more than 300 times faster than alignment based method, bowtie/eXpress. My method is also more than two times faster than other alignment free method. In addition, the memory usage of my method is smaller than other methods. Since Matataki is even faster than uncompressing bzip2 format or SRA format, the bottleneck of RNA-Seq meta-analysis is now uncompressing sequences, not mapping reads.

Part II

The development of massive analysis method for large RNA-Seq dataset

Chapter 5

Introduction

In past two decades, the number of published microarray data was rapidly increasing. Using these microarray data, thousands of meta-analysis were performed [52]. For example, Tang *et al.* [53] analyzed microarray data in TCGA [54] and revealed let-7b as a biomarker of cancer. For another example, Mabbott *et al.* [55] found signatures to distinct B-cell subsets. Microarray meta-analysis was widely accepted method.

RNA-Seq technology [34] is now de facto standard to measure gene expression level. The number of RNA-Seq is also rapidly increasing today. The same approach with microarray can be applied to RNA-Seq data now. However, the calculation time to quantify RNA-Seq data was a big problem to perform meta-analysis of RNA-Seq. As I described in Part I, I developed very fast quantification method to resolve this problem.

Gene coexpression is one of the most powerful applications of gene expression meta-analysis to unravel novel gene-to-gene relationship or gene functions. For example, Bottcher *et al.* [56] found a biosynthetic pathway, and confirm the genes in the pathway are co-expressed using a coexpression database.

Currently, some papers describing gene coexpression databases were published. COR-NET 2.0 [57] is a plant specific database of coexpression, protein-protein interaction, regulatory interactions, gene associations and functional annotations. They created *Arabidopsis thaliana* and *Zea mays* coexpression data from microarray data. They also integrated other

annotation, and developed user-friendly web-interface.

STARNET2 [58] is another coexpression database. They targeted *Homo sapiens*, *Rattus norvegicus*, *Mus musculus*, *Gallus gallus*, *Danio rerio*, *Drosophila melanogaster*, *Caenorhabditis elegans*, *Saccharomyces cerevisiae*, *Arabidopsis thaliana* and *Oryza sativa*, and created coexpression data from microarray data. They stopped updating coexpression and the web service was inaccessible in November 2015.

GeneFriends [59] provided RNA-Seq and microarray based gene coexpression data. They targeted human and mouse. To create RNA-Seq based gene coexpression, they mapped reads to genome using STAR [60], and counted the number of reads in the gene position with their original Java based software. Owing to STAR, they processed over 4,000 RNA-Seq data and succeeded to predict some gene functions with their gene coexpression data.

COXPRESdb [61] and ATTED-II [62] are only coexpression databases that provided coexpression of plants, animal and other species now, and the qualities of almost all coexpression data were evaluated properly. Originally, these databases provide microarray based coexpression data. These databases are updated every two years, therefore, the quality of coexpression is improving steadily.

In this part, I calculated RNA-Seq based coexpression for 5 species, human, mouse, rat, zebrafish, fruit fly. All coexpression data were evaluated with Gene Ontology. Since I introduced Matataki to calculate animal gene expression, my processing pipeline can deal with larger number of RNA-Seq in near future. I describe the methods to calculate RNA-Seq based coexpression, and the results of comparison with microarray based coexpression. The calculated coexpression data were released in COXPRESdb [61].

Chapter 6

Materials and Methods

6.1 Downloading and managing SRA files

All sequenced data including genome, RNA-Seq or other NGS-based sequenced data are archived in DDBJ Sequence Read Archive [18], NCBI Sequence Read Archive [17] and EBI Sequence Read Archive [11] under International Nucleotide Sequence Database Collaboration [19]. Since the data size of archived sequences is increasing at an exponential manner [63], downloading all data in these databases is impossible in realistic time. To determine which SRA files should be downloaded, I downloaded all metadata about these SRA files first. All metadata are written in XML files separately for each submission. Each submission has the information about a submission, studies, samples, experiments and runs. Submission data have the submission date and the name of submitter. Studies data have metadata about studies, such as a study title, an abstract. Samples data have metadata about experimented samples. They contain taxonomy ID, name of the sample and other attributes of samples. Each experiment data has the information about an instrument to sequence, library strategy, sample and study ID, and other attributes about an experiment. Each experiment has a run. One run corresponds to one sequencing. I transformed this information into SQLite3 database. I selected runs that library strategy is “RNA-Seq”, instruments were manufactured by Illumina and the number of run in a study is smaller than 50 and larger than three. When I calculated gene coexpression, I normalized bias of experimenters, sequence centers by a

gene centering procedure. In the procedure, I subtracted a mean expression level from each gene expression level for each study. If one study has too many runs, it is hard to normalize condition bias. Therefore, the number of run in a study was limited to 50 for *Homo sapiens* and *Mus musculus*. Since the number of RNA-Seq sample is small in other species, the number of run in a study was limited to 200 for *Rattus norvegicus*, *Danio rerio* and *Drosophila melanogaster*.

6.2 Estimate gene expression level

Gene expression levels were estimated by using Matataki I described in Part I. The detailed parameters used in this analysis are shown in Table 6.1. To calculate expression level in parallel, calculation pipeline was integrated with GridEngine [64].

Table 6.1: Parameters to estimate gene expression

Parameters for fastq-dump	
Quality Filter	Filter used in current 1000 Genomes data
Minimum Length	50
Apply left and right clips	yes
Command Options	-W -M 50 --skip-technical -Z --split-files --qual-filter-1
Parameters for Matataki	
N	32
Step size: S	16
Accept count: M	2

6.3 Calculation of gene coexpression

6.3.1 Normalize expression data

Raw read counts of a sample is not comparable with other samples because the total number of read sequences and the total abundance of RNA are different for each sample. Some normalization methods were proposed in previous studies. I compared four methods, TMM [65], quantile normalization [66], normalize summation of FPKM and raw value of FPKM. After normalizing value, logarithm function was applied to expression value. I compared these

normalization methods by evaluating with Gene Ontology term, described in the following section.

6.3.2 Calculation of gene coexpression

Gene coexpression represents that how similar two gene expression patterns are. Similarity of two gene expression patterns was measured by Person Correlation Coefficient (PCC). I also introduced pre-process and post-process to reduce sequence bias and improve relationship prediction performance. The overview of calculation method is shown in Figure 6.1.

First, I did gene-centering to reduce bias of experimenters and machines. Since raw RNA-Seq expression profiles are biased by of experimenters or other non-biological factors [67], these biases should be normalized. In this analysis, I subtracted the mean of gene expression level from each gene expression profile for every study and every gene (shown in Figure 6.1 (A) and (B)). A definition of one study was one study entry in the Short Read Archive.

Second, Person Correlation Coefficient values were calculated for each gene pair (shown in Figure 6.1 (C)). PCC is defined in following formula.

$$\text{PCC}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (6.1)$$

, where x is a progression of a gene expression level, and y is a progression of another gene expression level. Since a value range of PCC was different for each gene, a rank of PCC was better measure to represent strength gene relationship. I used Mutual Rank (MR) [68] to measure gene relationship in this study (shown in Figure 6.1 (D), (E)). Mutual Rank is defined as following formula.

$$\text{MR}(a, b) = \sqrt{(\text{Rank of PCC } a \rightarrow b) \times (\text{Rank of PCC } b \rightarrow a)} \quad (6.2)$$

I compared the performance of gene function prediction by using gene coexpression between Mutual Rank and PCC using Gene Ontology based evaluation method describing in Section

6.3.4

6.3.3 Implementation of gene coexpression calculation

Since a gene-expression profile table is too large to calculate PCC with existing software, such as R, I implemented C++ based fast PCC table calculation software. It is optimized for workstation with large memory and multi-core CPUs. All expression profile and correlation data are stored in a memory-mapped file to access data efficiently. It is also multi-threaded using OpenMP. This software can calculate PCC, PCC rank and mutual rank for all gene pairs in 30 minutes with Intel[®] Xeon[®] CPU E5-2690 and 32 GB memory when the number of gene is 20,000 and the number of runs is 5,000.

6.3.4 Evaluation of gene coexpression by using Gene Ontology

Gene coexpression data were evaluated with Gene Ontology (GO) [69]. Gene Ontology is a vocabulary of gene function annotation. Part of genes is annotated with Gene Ontology to describe gene functions.

When gene A shares Gene Ontology Term with another gene B, gene A may be related to gene B. Since gene coexpression represents the strength of gene-to-gene relationships, these relationships can be evaluated using whether a gene shares a Gene Ontology Term with strong related genes.

The quality of gene coexpression was measured by partial area under curve (pAUC) of receiver operating characteristic curve (ROC curve). When I assume that sharing gene function between two genes corresponds to correlating gene expression pattern, the quality of gene coexpression can be evaluated with checking whether highly correlated gene pair shares Gene Ontology Terms or not. I defined truly related gene pairs with sharing at least one GO Term, and predicted related gene pairs with mutual rank (shown in Figure 6.2). When gene A has GO X and GO Y, and a threshold of a mutual rank is 1.9, gene B in Figure 6.2 is a “true positive” because gene B shares GO X with gene A and a mutual rank between gene A

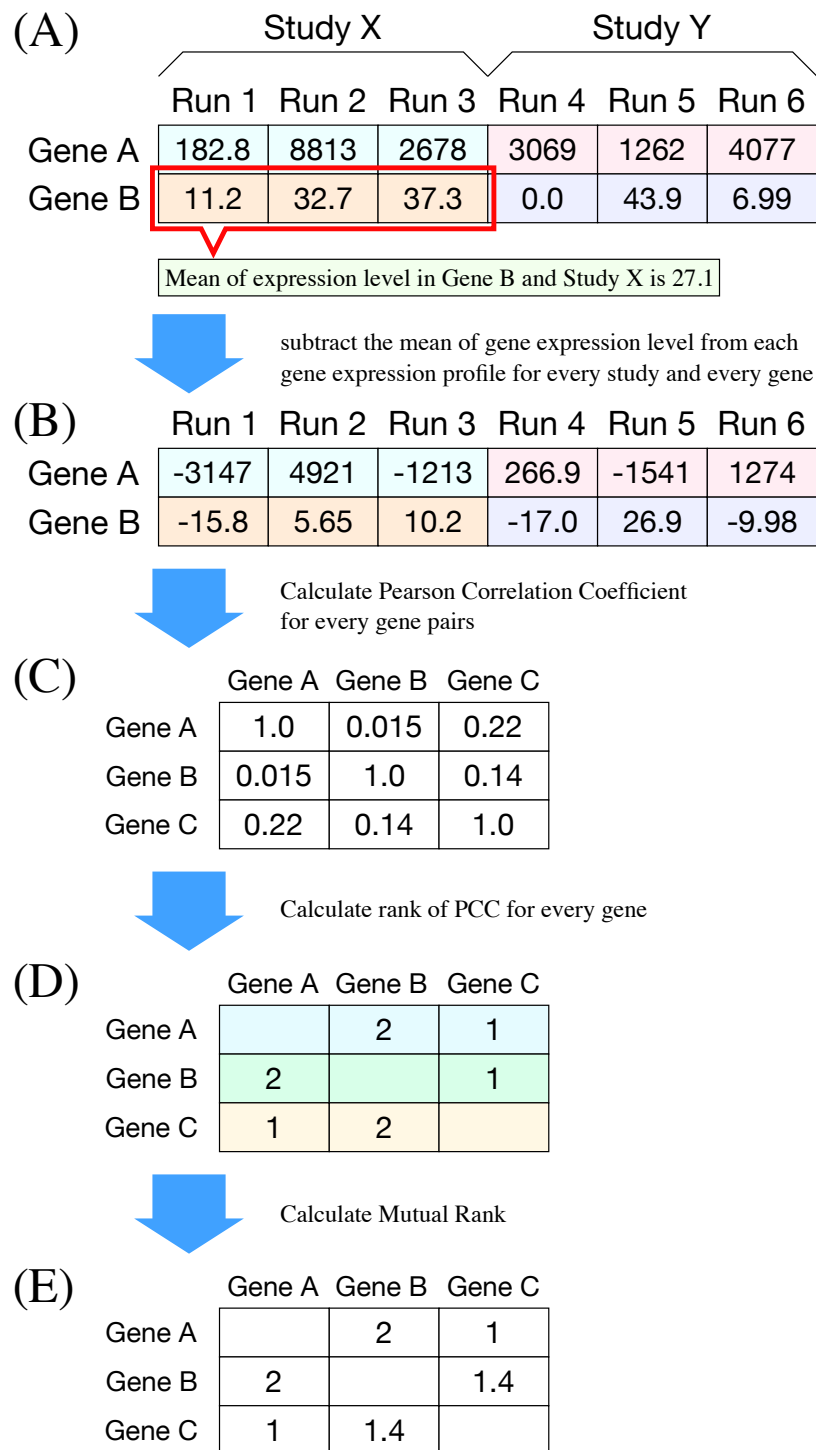


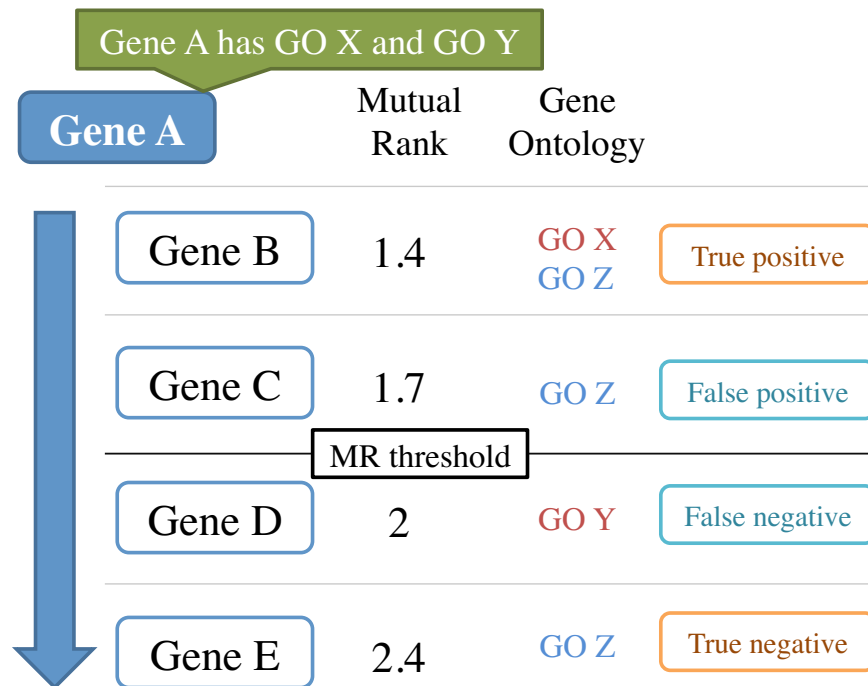
Figure 6.1: Overview of gene coexpression calculation

and gene B is smaller than the threshold. Similarly, gene C is a “false positive”, gene D is a “false negative” and gene E is a “true negative”. I repeated this procedure for all gene pairs to calculate a true positive rate and a false positive rate.

ROC curve is a plot of true positive rates and false positive rates when a threshold is varied. When a predictor can predict perfectly, ROC curve goes through top left. If a predictor predicts randomly, ROC curve looks like a diagonal. Since gene coexpression data are often used to predict protein complex, metabolic pathway or other gene functional relationships, prediction performance in low false positive area is important than total prediction performance. I used partial AUC (pAUC) instead of total AUC to evaluate prediction performance in the low false positive rate area (shown in Figure 6.3). In this study, I focused on the range that false positive rate is less than 0.01. pAUC will be 5×10^{-5} when random predictor is evaluated.

In this study, I used only limited part of Gene Ontology Terms to evaluate performance. Gene Ontology Terms that are annotated to many genes are not informative because predicting general GO is easy problem. I limited Gene Ontology that is annotated to 20 genes or fewer genes and 5 genes or more genes. I evaluated prediction performance for each namespace separately because each namespace covers different domains.

All Gene Ontology and NCBI gene2go were downloaded at August 11th 2015 from geneontology.org/ and Human Genome Center NCBI mirror site.



$$\text{True Positive Rate} = \frac{\square \text{ True positive}}{\square (\text{True positive} + \text{False negative})}$$

$$\text{False positive Rate} = \frac{\square \text{ True negative}}{\square (\text{True negative} + \text{False positive})}$$

Figure 6.2: Evaluate gene coexpression with Gene Ontology

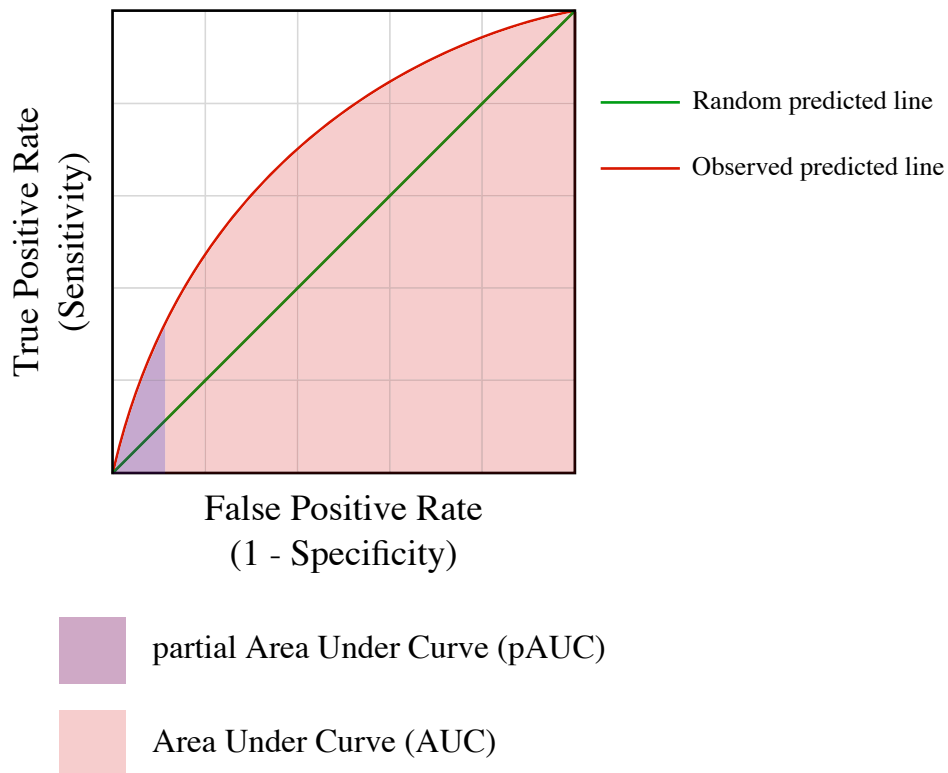


Figure 6.3: Description of ROC curve and AUC, partial AUC

Chapter 7

Results & Discussion

7.1 Statistics of SRA files

On September 9th 2015, 1,403,951 runs were registered in Short Read Archive. The number of RNA-Seq runs was 226,937. The numbers of RNA-Seq sequenced for each species are shown in Table 7.1. Since this number includes controlled-access data, such as dbGaP [70], some data is available only for permitted users. In this study, I narrowed down to runs that were sequence by Illumina sequencers to avoid the bias of sequencers, and part of “Transcriptome Analysis” to ensure that RNA-Seq was performed for transcriptome analysis. The numbers of Illumina sequenced and part of “Transcriptome Analysis” is shown in Table 7.1, column “type A runs”.

Some large scale studies have more than 100 runs. For example, a study SRP027537 has 4,894 runs. In the study, they performed a lot of single cell RNA-Seq and unraveled dynamic paracrine control of cellular variation [71]. Normalizing experimenter or sequence bias of these large scale studies is challenging because too large study may be performed by more than one experimenter or sequencers. In this study, large-scale studies, that have more than 50 runs in a study were neglected to deal with bias and reduce download and calculation time. The numbers of runs included in studies that have 50 runs or fewer are shown in Table 7.1, column “type B runs”. Since the numbers of RNA-Seq runs of other species were small, I used 200 as the upper limit of runs in a study.

Table 7.1: The number of RNA-Seq runs in Short Read Archive

	Scientific name	Taxonomy ID	# of RNA-Seq runs	# of Type A runs	# of studies	# of Type B runs
1	<i>Mus musculus</i>	10090	57311	49197	1192	12190
2	<i>Homo sapiens</i>	9606	59538	27692	1076	10738
3	<i>Arabidopsis thaliana</i>	3702	4294	3277	210	2278
4	<i>Drosophila melanogaster</i>	7227	8568	7620	228	1745
5	<i>Saccharomyces cerevisiae</i>	4932	3857	3022	90	927
6	<i>Caenorhabditis elegans</i>	6239	2125	1781	105	895
7	<i>Danio rerio</i>	7955	5676	5098	88	859
8	<i>Zea mays</i>	4577	2907	2253	82	669
9	<i>Bos taurus</i>	9913	1267	1005	46	587
10	<i>Glycine max</i>	3847	984	766	44	578
11	<i>Rattus norvegicus</i>	10116	4368	3987	59	469
12	<i>Gallus gallus</i>	9031	844	517	43	457
13	<i>Chlamydomonas reinhardtii</i>	3055	718	487	24	432
14	<i>Sus scrofa</i>	9823	495	411	39	319
15	<i>Solanum lycopersicum</i>	4081	1090	629	33	275
16	<i>Escherichia coli</i>	562	646	399	28	261
17	<i>Schizosaccharomyces pombe</i>	4896	539	222	30	222
18	<i>Oryza sativa</i>	4530	635	208	27	208
19	<i>Ovis aries</i>	9940	595	336	15	204
20	<i>Oryza sativa Japonica Group</i>	39947	427	199	15	199
21	<i>Brassica napus</i>	3708	313	268	13	193
22	<i>Aedes aegypti</i>	7159	953	839	13	156
23	<i>Triticum aestivum</i>	4565	626	304	18	147
24	<i>Medicago truncatula</i>	3880	365	143	11	143
25	<i>Macaca mulatta</i>	9544	910	577	21	142
26	<i>Gossypium hirsutum</i>	3635	335	139	21	139
27	<i>Tribolium castaneum</i>	7070	150	138	7	138
28	<i>Escherichia coli str. K-12 substr. MG1655</i>	511145	242	190	15	136
29	<i>Vitis vinifera</i>	29760	569	283	13	127
30	<i>Xenopus (Silurana) tropicalis</i>	8364	236	174	23	122
31	<i>Solanum tuberosum</i>	4113	157	118	11	118
32	<i>Malus domestica</i>	3750	278	216	13	117
33	<i>Cryptococcus neoformans</i>	5207	476	106	5	106
34	<i>Neurospora crassa</i>	5141	253	217	14	105
35	<i>Anopheles gambiae</i>	7165	505	148	13	96
36	<i>Zea mays subsp. mays</i>	381124	1017	863	10	95
37	<i>Equus caballus</i>	9796	728	656	8	95
38	<i>Dictyostelium discoideum</i>	44689	205	161	7	94
39	<i>Pseudomonas aeruginosa</i>	287	470	297	7	88
40	<i>Staphylococcus aureus</i>	1280	260	232	8	88
41	<i>Callithrix jacchus</i>	9483	166	88	7	88
42	<i>Picea abies</i>	3329	112	88	4	88
43	<i>Salmo trutta</i>	8032	88	88	4	88
44	<i>Schmidtea mediterranea</i>	79327	115	83	10	83
45	<i>Candida albicans</i>	5476	234	177	15	81
46	<i>Mus musculus domesticus</i>	10092	101	81	6	81
47	<i>Macaca fascicularis</i>	9541	312	80	5	80
48	<i>Trypanosoma brucei</i>	5691	86	78	7	78
49	<i>Gasterosteus aculeatus</i>	69293	105	75	4	75
50	<i>Sorghum bicolor</i>	4558	139	74	4	74

Type A run	Study type is "Transcriptome Analysis" and platform is "ILLUMINA"
Type B run	In type A runs, total numbers of runs in studies that have 50 runs or less

7.2 Gene coexpression

I calculated gene coexpression for *Homo sapiens* (Human), *Mus musculus* (Mouse), *Rattus norvegicus* (Rat), *Danio rerio* (Zebrafish) and *Drosophila melanogaster* (Fruit Fly). Microarray based gene coexpression data for performance comparisons were downloaded from COXPRESdb [61]. The values of pAUC in GO prediction are shown in Table 7.2, 7.3, 7.4, 7.5 and 7.6. For quality control, I limited to RNA-Seq runs that have 50 or longer sequence length and more than 5,000,000 mapped reads to calculate gene coexpression. The numbers of RNA-Seq runs that were used to calculate gene coexpression are also shown in each table.

7.2.1 Comparison of normalization factor

I compared four normalization methods: quantile normalization, TMM, summation and raw value of FPKM. The results of human gene coexpression are shown in Table 7.2. As a result, the prediction performance of quantile normalization showed the best pAUC score in mutual rank and PCC. On other hand, summation normalization and TMM normalization showed worse pAUC than other two normalization, raw FPKM and quantile normalization in both mutual rank and PCC. The pAUC values of raw FPKM were the second best score in both mutual rank and PCC, but the value of pAUC of PCC was much smaller than that of quantile normalization. This tendency was also observed in *Drosophila melanogaster* and *Mus musculus*.

7.2.2 Comparison with microarray-based coexpression

I compared the prediction performance with microarray based coexpression data. The results are shown in Table 7.2, 7.3, 7.4, 7.5 and 7.6. As a result, the prediction performances of *Homo sapiens* and *Mus musculus* were better than the performance of microarray based gene coexpression data, even if the number of samples is much smaller than microarrays. This observation suggests that gene expression quality of RNA-Seq is better than that of

microarray, and RNA-Seq based gene coexpression is promising solution to predict gene-to-gene relationship in the future.

The prediction performances based on RNA-Seq in other species, *Rattus norvegicus*, *Danio rerio* and *Drosophila melanogaster* were worse than based on microarray. Since the numbers of RNA-Seq samples of these species were much smaller than that of microarray samples, more RNA-Seq samples are required to improve prediction performance.

7.2.3 Effect of sample size

I evaluated effect of the sample size for coexpression quality. I down-sampled RNA-Seq studies and calculated pAUC of each Gene Ontology namespaces using Mutual Rank. As shown in Figure 7.1, pAUC values were highly correlated with sample sizes in human and mouse. This result suggests that Gene Ontology prediction performance of gene coexpression will be improved when the number of RNA-Seq is increased in the future.

7.2.4 Availability

The result of gene coexpression is available as web-based databases at COXPRESdb (<http://coxpresdb.jp/>) and ATTED-II (<http://atted.jp/>).

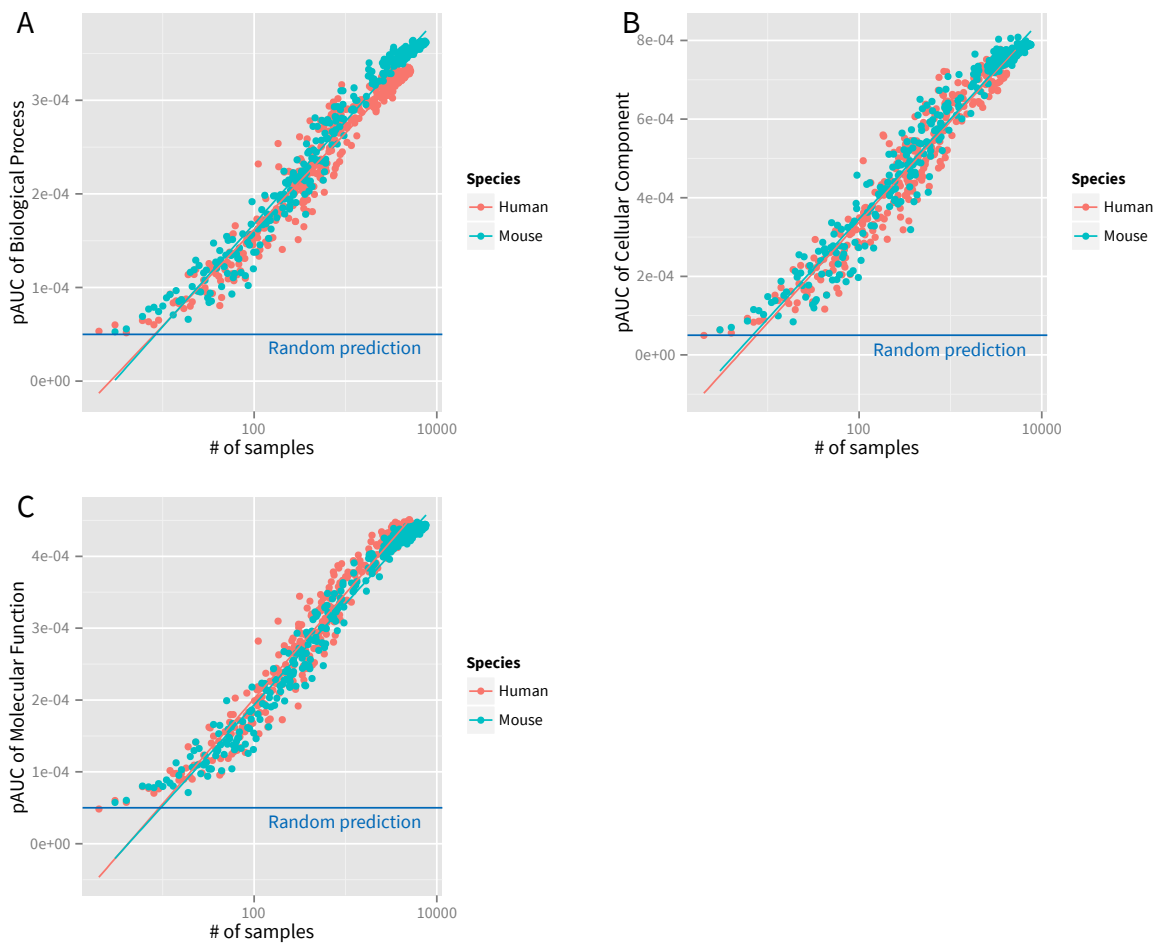


Figure 7.1: Effect of sample sizes to coexpression quality

Table 7.2: GO prediction pAUC of *Homo sapiens*

Technology	Normalization	# of genes	# of runs	pAUC of BP/MR	pAUC of CC/MR	pAUC of MF/MR	pAUC of BP/PCC	pAUC of CC/PCC	pAUC of MF/PCC
Microarray	RMA	20280	73083	2.71	6.37	3.64	1.78	4.88	2.41
Microarray	RMA	19788	12640	3.22	7.18	4.27	2.07	4.94	2.86
RNA-Seq	Quantile	19734	5163	3.31	7.61	4.47	2.76	6.15	3.93
RNA-Seq	Summation	19734	5163	3.19	7.42	4.40	1.59	3.93	2.52
RNA-Seq	FPKM	19734	5163	3.30	7.60	4.42	2.27	5.53	3.54
RNA-Seq	TMM	19734	5163	3.23	7.41	4.41	1.80	4.35	2.87

the unit of each pACU value is 10^{-4}

Table 7.3: GO prediction pAUC of *Mus musculus*

Technology	Normalization	# of genes	# of runs	pAUC of MR-BP	pAUC of MR-CC	pAUC of MR-MF	pAUC of PCC-BP	pAUC of PCC-CC	pAUC of PCC-MF
Microarray	RMA	20959	31479	3.25	7.13	3.94	2.49	6.08	3.07
RNA-Seq	Quantile	20244	8732	3.63	7.84	4.44	3.15	6.66	3.68
RNA-Seq	TMM	20244	8732	3.58	7.66	4.41	2.19	5.10	3.12
RNA-Seq	Summation	20244	8732	3.57	7.70	4.39	2.61	5.84	3.55
RNA-Seq	FPKM	20244	8732	3.58	7.82	4.40	3.11	6.85	3.76

Table 7.4: GO prediction pAUC of *Rattus norvegicus*

Technology	# of genes	# of runs	pAUC of MR-BP	pAUC of MR-CC	pAUC of MR-MF	pAUC of PCC-BP	pAUC of PCC-CC	pAUC of PCC-MF
Microarray	13751	27481	2.51	6.00	3.49	1.91	4.95	2.65
RNA-Seq	28609	657	1.91	3.64	2.40	2.06	4.02	2.55

Table 7.5: GO prediction pAUC of *Danio rerio*

Technology	# of genes	# of runs	pAUC of MR-BP	pAUC of MR-CC	pAUC of MR-MF	pAUC of PCC-BP	pAUC of PCC-CC	pAUC of PCC-MF
Microarray	10112	1727	5.97	11.55	6.60	5.37	9.64	5.14
RNA-Seq	20950	482	4.49	7.98	4.44	4.03	6.64	3.78

Table 7.6: GO prediction pAUC of *Drosophila melanogaster*

Technology	# of genes	# of runs	Correction	pAUC of MR-BP	pAUC of MR-CC	pAUC of MR-MF	pAUC of PCC-BP	pAUC of PCC-CC	pAUC of PCC-MF
Microarray	12626	4741	RMA	4.81	10.2	6.57	3.24	5.90	4.24
RNA-Seq	13114	1012	Quantile	4.17	9.03	5.96	2.55	4.02	3.56
RNA-Seq	13114	1012	Summation	4.01	8.80	5.87	2.28	3.62	3.12
RNA-Seq	13114	1012	TMM	3.96	8.74	5.87	2.19	3.74	3.05

Chapter 8

Conclusion

In this part, I calculated gene coexpression based on RNA-Seq and evaluated its performance using Gene Ontology. As the result, gene coexpression based on RNA-Seq outperformed in *Homo sapiens* and *Mus musculus*. In other species, performances of coexpression based on RNA-Seq are worse than microarray because the numbers of runs are much smaller than that of microarray. These results were published in web-based databases, COXPRESdb (<http://coxpresdb.jp/>) and ATTED-II (<http://atted.jp/>). These databases are only databases that the performances of coexpression were evaluated properly. I also evaluated normalization methods of expression profiles and concluded quantile based normalization is the best method.

Part III

Comparison of Gene Coexpression Profiles and Construction of Conserved Gene Networks to Find Functional Modules

This part is based on Okamura, Y., Obayashi, T. & Kinoshita, K. “Comparison of Gene Coexpression Profiles and Construction of Conserved Gene Networks to Find Functional Modules.” *PLoS ONE* (2015). doi:10.1371/journal.pone.0132039

Chapter 9

Introduction

With the sequencing of the human genome completed [6, 7, 72], the next step is to annotate all of the functional elements in the genome, to reveal the genomic content. In spite of intensive analyses using EST [73], CAGE [74] and/or comparative genomics [9, 75, 76], about half of the genes remain uncharacterized. Thus, the focus has shifted to the functional annotation of the genes [77, 78].

Although each gene has its specific function, complicated cellular functions are usually achieved by combinations of individual functions, as in the ribosome, which synthesizes proteins by the coordinated functions of many ribosomal proteins and RNAs. Metabolic pathways are also good examples of genes that work together to achieve various biological functions. Therefore, to understand the functional role of each gene, it is essential to find groups of genes working with the same timing, by identifying genes with functional relationships [79].

Various kinds of relationships can be considered to identify the functional modules. Protein-protein interactions (PPI), obtained by high throughput experiments such as yeast two-hybrid methods [80], provide some of the most comprehensive interaction data [81, 82], but they only cover the proteins with direct interactions. In other words, genetic interactions (e.g. transcription factor and target gene) and metabolic pathways are not included. Another way to infer gene networks is based on the manual curation of the literature [83]. This

approach provides high quality interaction data, but is quite time consuming and requires large amounts of human resources.

DNA microarrays generate profiles of comprehensive gene expression patterns and their clustering [84, 85] to detect functionally related genes. Since one gene expression profile only provides a snapshot of a cell state, many expression profiles are required to detect related genes with reliable accuracy. Currently, over ten thousand gene expression data points are available for some microarray platforms, and they have been used to identify genes [86], genetic interactions [87] and gene modules [88, 89].

To detect the regulatory relationships among genes, coexpression is a popular and promising approach [88, 90]. Coexpression is calculated from large amounts of expression data obtained by microarray [32] or RNA-seq [34] experiments, to detect the genes with similar expression profiles. In this part, I have focused on the microarray data, because the number of available microarray samples is about 10 times larger than that of RNA-seq experiments. RNA-seq has some advantages, in terms of the gene expression profile quality. However, the number of samples is also an important factor to identify good functional relationships between genes, because larger coverage of various conditions is necessary to detect subtle functional connections. According to the progress of several international projects, such as ENCODE [91], the amount of available expression data is rapidly increasing, but is still currently limited as compared with that of DNA microarrays. Our approach will be applicable to RNA-seq data in the future, when larger amounts are available.

For the identification of gene functions, sequence conservation is also very useful. Since comparative analyses of genome sequences have worked very well to identify new potentially functional elements, as in the recent comparisons of 29 mammalian genomes [75], such analyses are becoming a standard practice when new genome sequences are solved [76, 9, 92].

Since both gene expression and sequence conservation are useful to understand gene functions, the introduction of conservation into analyses of gene expression profiles should be

promising. Su *et al.* [93] compared the human and mouse transcriptomes, and found similar gene expression profiles in the corresponding organs. More recently, Brawand *et al.* [94] reported that the main differences in gene expression are due to the lineage, the chromosomes, and the tissues. These approaches were very useful to characterize the functional relationships among genes over species, but a serious problem still exists in the consideration of the conservation of gene expression patterns. It is easy to obtain samples from similar organs, but the similarity may not always indicate the correspondence of the organs. It is almost impossible to obtain samples corresponding to the same type of cells in the same state.

To overcome this difficulty, some studies have proposed methods to match samples over species. Le *et al.* [94] developed a method to match experiments over species, by introducing a new distance function between the samples, and Wise *et al.* [95] tried to match experiments based on their descriptions along with the expression data. These methods may work well to find similar gene expression states, but they naively assume that homologous genes have similar expression profiles. As I describe in this part, this assumption is not always true.

I now propose a new method to compare gene expression patterns without sample matching, to focus on the relationships among the genes in each species and to compare the relationships among species. In this approach, I assume that the interactions between genes are conserved over species, if the interactions are fundamentally important for the biological roles of the genes. More precisely, I introduced a new method to measure the coexpression similarities. I created gene networks based on the conserved gene coexpression to find the functional modules by using a graph community detection algorithm, and found some well-enriched functional gene modules without any prior knowledge.

Chapter 10

Results

10.1 Patterns of coexpression conservation

I compared the gene lists of the corresponding (or homologous) gene pairs to evaluate the conservation of coexpression patterns and expression data from two species, human and mouse. For each human gene (referred to as the guide gene), a list of coexpressed genes was created by ordering the genes by the coexpression strength, and a corresponding list of mouse genes was constructed for each homologous gene to the guide gene. The coexpression conservation of a homologous gene pair was measured as the similarity in the lists for the top N genes (Figure 10.1A). When the human guide gene had multiple homologous mouse genes, I compared the coexpressed gene lists for each pair of homologous genes. Next, I drew a “conservation chart” based on the number of corresponding gene pairs in the most coexpressed N genes, as shown in Figure 10.1B. If the human and mouse coexpression lists are exactly equal, then the conservation chart should look like the blue dashed line in Figure 10.1B. If the coexpression lists are equal to Figure 10.1A, then the conservation chart looks like the red dashed line in Figure 10.1B. A conservation chart represents the degree of similarity in the coexpression lists and indicates where the similarity exists.

One of the highly conserved genes was RPS14 (ribosomal protein S14), which had 71 corresponding genes in the top 100 most coexpressed genes (Figure 10.1C). Among the 60 genes, 55 are ribosomal genes, which correspond to 92% (=55/60) of the human ribosomal

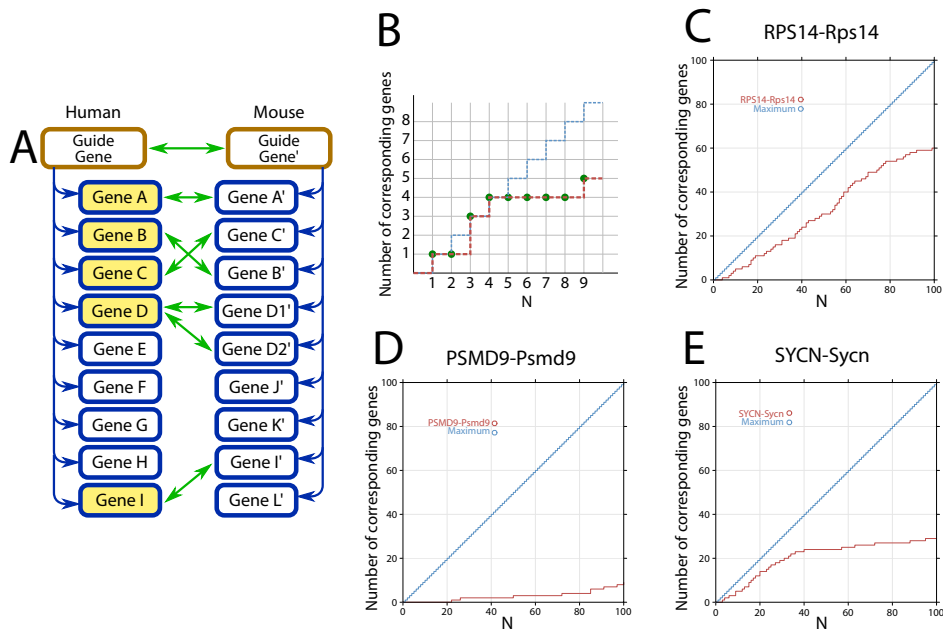


Figure 10.1: Overview of the conservation calculation method

(A) Schematic explanation of the comparison method for the conserved gene lists. Prepare a gene list pair for an orthologous gene pair from human and mouse. Count the number of human genes (yellow highlighted genes) with corresponding genes in the top N genes, where green arrows mean corresponding gene pairs. When a human gene corresponds to multiple mouse genes, I counted one human gene. However, when a mouse gene corresponds to multiple human genes, I counted all of the human genes. (B) Conservation chart of (A). This chart illustrates the change in the number of corresponding genes against the parameter value, N . (C) An example of a conservation chart for the most conserved guide gene. (D) An example of a conservation chart with a typical shape. (E) An example of genes with a turning point.

genes tested. This result partially demonstrates the potential of our approach to detect related genes. However, many genes have low coexpression conservation, as in the example of PSMD9 (Figure 10.1D). On average, 13.1 genes were found to have corresponding genes in the top 100 most coexpressed genes.

Although the “shapes of the conserved lines” in the conservation charts were quite divergent and thus prevented a systematic classification, I found an interesting pattern, as shown in Figure 10.1E for SYCN (syncollin). This gene has a well-conserved region for the top 39 genes, while there were only slight increases after that, and 24 of the 39 genes have the homologous genes in mouse. SYCN is involved in the pancreatic secretion pathway (KEGG:hsa04972), and 12 of the 24 genes are also involved in the same pathway. This observation suggested that SYCN and the 24 genes may form a functional cluster for the pathway. When we assume that functional gene clusters are conserved over species, then the two coexpression lists for the orthologous gene should be similar over species. Therefore, it may be possible to detect the functional clusters by focusing on the well-conserved regions. Hereafter, I refer to the genes in conserved regions that have corresponding mouse genes (namely, the 24 genes in the above example) as “conserved coexpressed genes” or in short “CC genes”.

10.2 Identification of conserved coexpressed genes

To detect the CC genes from the conservation chart, I tried to identify a turning point, where a well-conserved region goes into a less conserved one. For this purpose, I searched for a point by detecting a flat region in each conservation chart, because a conservation chart should be flat for the genes in a list if the orders of the two coexpression lists are random. Thus, the initial point of the flat region was defined as the turning point, and I defined the conserved region as the part on the left of the flat area. The CC genes were identified as the corresponding genes between human and mouse of a guide gene in the conserved region. See

the Materials and Methods section for the details of the turning point detection and the CC gene identification. As a result, 4,672 guide genes had a turning point. Each guide gene had 6.6 genes on average, and 3,776 non-redundant CC genes were identified.

10.3 Conserved gene network in human

To visualize the relationships among all of the guide genes and their CC genes, I represented them in a network style, where each node corresponds to a gene and an edge is drawn from a guide gene to a CC gene, and removed all of the unidirectional edges. The resulting networks are shown in Figure 10.2A. The networks consisted of one large and twenty small networks.

Since the large networks were too big to interpret, I separated them into more tightly related gene modules for convenience. For this purpose, I used the community detection algorithm developed by Palla *et al.* [97] for all of the networks shown in Figure 10.2A. This algorithm searches for densely connected sub-networks by integrating small cliques, and thus requires one parameter, the smallest clique size (SCS). I first used a default value (SCS = 4) and found 70 modules, as shown in Table 10.1. To characterize the functional roles of the modules, I performed GO enrichment analyses by the Fisher exact test, and selected the GO term with the smallest p-value from the statistically significant terms as the representative GO term.

Table 10.1: Detected gene modules Summary of detected gene modules and representative GO terms when SCS = 4

C#ID	Size	GOID	Representative GO name	# of GO annotated genes	# of intersect	p-value
1	404	GO:0002376	immune system process	1897	232	1.14E-99
2	97	GO:0043588	skin development	295	27	2.32E-19
3	83	GO:0030198	extracellular matrix organization	353	32	2.81E-26
4	67	GO:0006936	muscle contraction	255	35	4.16E-40
5	48	GO:0060271	cilium morphogenesis	153	7	1.68E-02
6	43	GO:0072376	protein activation cascade	52	11	5.09E-14
7	42	GO:0006414	translational elongation	88	35	3.85E-70
8	32	GO:0045333	cellular respiration	145	25	6.40E-41
9	31	GO:0006986	response to unfolded protein	128	10	1.84E-09
10	28	GO:0016126	sterol biosynthetic process	48	18	1.55E-35
11	23	GO:0008544	epidermis development	256	8	8.33E-05
12	22	GO:0007586	digestion	107	8	4.98E-08
13	21	GO:0007601	visual perception	175	16	4.15E-23
14	19	GO:0006520	cellular amino acid metabolic process	430	15	7.19E-16
15	19					
16	18					
17	17	GO:0048285	organelle fission	496	12	2.77E-10
18	17					
19	16					
20	15	GO:0019915	lipid storage	57	6	3.59E-07
21	14	GO:0048706	embryonic skeletal system development	116	11	4.46E-17
22	13	GO:0006458	'de novo' protein folding	52	9	8.23E-16
23	12	GO:0034728	nucleosome organization	87	5	1.42E-04
24	10	GO:0030317	sperm motility	35	3	4.26E-02
25	10	GO:0045333	cellular respiration	145	8	9.16E-11
26	10	GO:0006936	muscle contraction	255	6	1.45E-04

ID	Size	GOID	Representative GO name	# of GO annotated genes	# of intersect	p-value
27	9	GO:0007156	homophilic cell adhesion via plasma membrane adhesion molecules	91	9	2.48E-16
28	9	GO:0042438	melanin biosynthetic process	14	6	5.12E-13
29	8					
30	8					
31	7	GO:0006397	mRNA processing	393	7	2.62E-07
32	7	GO:0006096	glycolytic process	61	6	7.87E-10
33	6					
34	6	GO:0006956	complement activation	32	5	6.04E-09
35	6	GO:0031427	response to methotrexate	4	2	2.43E-02
36	6					
37	6					
38	6	GO:0043407	negative regulation of MAP kinase activity	65	4	1.50E-04
39	5	GO:0015988	energy coupled proton transmembrane transport, against electrochemical gradient	27	3	1.60E-03
40	5					
41	5	GO:0007588	excretion	63	3	2.16E-02
42	5	GO:0006364	rRNA processing	107	5	5.32E-07
43	5					
44	4	GO:0009954	proximal/distal pattern formation	29	4	3.52E-07
45	4					
46	4	GO:0002331	pre-B cell allelic exclusion	3	2	4.87E-03
47	4	GO:0006631	fatty acid metabolic process	296	4	4.65E-03
48	4					
49	4	GO:0008211	glucocorticoid metabolic process	24	4	1.57E-07
50	4					
51	4	GO:0006687	glycosphingolipid metabolic process	49	4	3.14E-06
52	4	GO:0007339	binding of sperm to zona pellucida	32	3	1.09E-03
53	4					
54	4					
55	4	GO:0006521	regulation of cellular amino acid metabolic process	60	4	7.23E-06
56	4					
57	4	GO:0022904	respiratory electron transport chain	93	3	2.83E-02
58	4					
59	4	GO:0006986	response to unfolded protein	128	4	1.58E-04
60	4					
61	4					
62	4					
63	4					
64	4	GO:0019322	pentose biosynthetic process	4	4	1.48E-11
65	4	GO:0060481	lobar bronchus epithelium development	5	2	1.62E-02
66	4	GO:0070059	intrinsic apoptotic signaling pathway in response to endoplasmic reticulum stress	29	3	8.00E-04
67	4					
68	4					
69	4					
70	4	GO:0002399	MHC class II protein complex assembly	4	2	9.74E-03

As a result, 45 of the 70 modules had significantly enriched GO terms. For example, the representative term of the largest modules shown as ID: A-1 in Figure 10.2A was GO:0002376 (immune system process), where 232 out of 404 genes had the GO term.

Some detected modules are not labeled with a Gene Ontology Term, as in the cases of the 15th, 16th, 18th and 19th modules. These modules had no significant terms with P-values < 0.05 , and thus might be novel functional modules, such as the other modules with significant terms, because they have comparatively strong conserved coexpression.

Some gene modules had similar annotations and overlaps, indicating the existence of larger modules, if I searched modules for lower density. To elucidate the relationships among the modules, I observed the overlaps by changing three different SCS parameters of the module detection algorithm. I used three, four and five as the SCS to reveal both the low-density modules and high-density modules, as recommended by Palla *et al.* [97]. The numbers of

¹C: Community

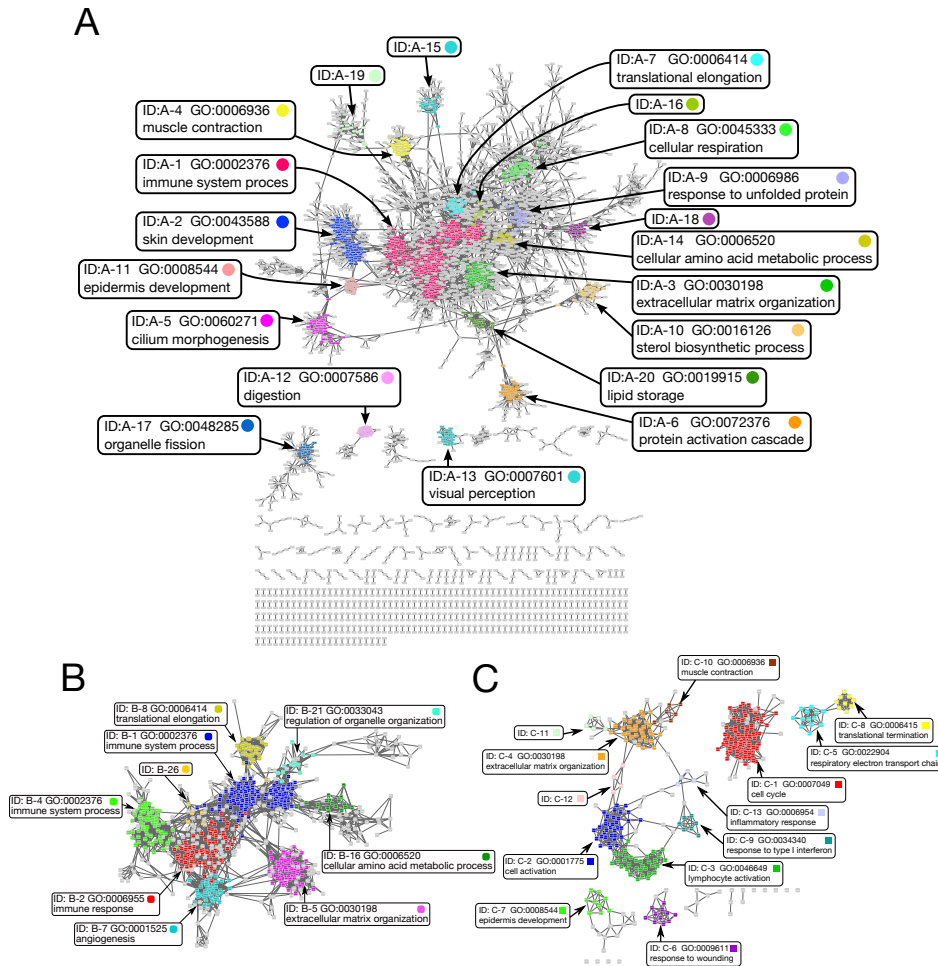


Figure 10.2: Detected gene networks

(A) Gene networks based on coexpression conservation. I generated networks with 3,776 genes. The largest gene network contained 2,717 genes. Genes (nodes) were colored when they were a member of the top 20 largest modules with $SCS=4$. Gray nodes were parts of some smaller modules, and black nodes were not parts of any modules. I prepared this picture of the network with Cytoscape [96]. (B) The largest gene modules with $SCS=3$ and the large modules with $SCS=5$ are colored. This module has the representative term “immune system process”, but not all of the sub-modules with $SCS=5$ have immune-related GO terms, as discussed in the text. (C) The gene network without a turning point. Since some gene networks had high coexpression conservation, no flat region was found. I used 100 instead of a turning point, because turning points cannot be defined for these genes. This network was generated from these highly conserved genes.

detected gene modules were 107, 70 and 42, and the mean numbers of genes were 17.4, 19.3 and 24.6, respectively. The number of detected module with SCS=4 (70 modules) may be larger than expected as expected, but it should be noted that our method will not detect the gene modules that were changed from mouse to mouse, because our method is based on the conservation between human and mouse, which may result in that the number of modules was limited.

The largest gene module in SCS = 3 is shown in Figure 10.2B. In this module, 308 out of 767 genes had the GO term GO:0002376 (immune system process). This module can be further separated into 9 sub-modules with 10 or more genes by using SCS=5, as indicated in Figure 10.2B, where different colors represent the different modules with SCS = 5. Some of the colored gene modules were related to the immune system GO term, but others were not. For example, the ID: B-1, B-2 and B-4 gene modules in Figure 10.2B are related to GO:0002376 (immune system process), while the ID: B-5 gene module at the bottom right in Figure 10.2B with the representative GO: 0030198 (extracellular matrix organization), and some other enriched GO Terms as shown in the web database at <http://v1.coxsimdb.info/coxsim/hsa-v13-01/mmu-v13-01/SCS:5/5>. Most of the enriched GO terms are directly related with immune system process, but I can also see some interesting terms such as GO: 0032963 (collagen metabolic process) and GO: 0001568 (blood vessel development). This result may indicate that the immune system tightly cooperates with collagen metabolic process, blood vessel development and other systems.

Some genes lacked turning points and had large numbers of corresponding genes, indicating that the genes are quite strongly conserved. To characterize them, I generated another gene network for them by regarding 100 as the tentative turning point, instead of determining a turning point. As a result, 336 genes, 1,953 edges and 8 individual networks were detected (shown in Figure 10.2C). Only 9 genes among the 336 genes had no connection with other genes without any turning points. I applied the community detection algorithm again for

this network, and found 13 modules. The largest module was ID: C-1 (Figure 10.2C), where 95 genes were involved and 85 of them were annotated as GO:0007049 (cell cycle). This result suggests that the genes for fundamental functions, such as cell cycle, translation or cytoskeleton, have highly conserved coexpression and are tightly connected in each function.

10.4 Effect of the introduction of conservation

I performed the same module detection analysis for a human coexpression network without conservation, to evaluate the effect of the conservation. Coexpression data for human were obtained from COXPRESdb [61], where the strengths of coexpression are described by a rank-based measure called Mutual Rank (MR) [68]. Smaller MR values indicate stronger coexpression.

When I used MR = 3, 5, 10, 15, 20, and 30 as cutoffs, 22, 165, 458, 600, 667, and 622 modules were detected, respectively. I calculated the GO enrichment of the modules for each MR threshold, and found that 5/22, 41/165, 76/458, 56/600, 56/667, and 33/622 modules were enriched with at least one GO term. However, the conservation filtering method proposed in this part detected 45 enriched modules out of 70 modules (Figure 10.3A), and the ratio of enriched modules based on coexpression conservation is clearly better than the ratios of enriched modules based on the non-filtering method with COXPRESdb at any MR threshold ($< 41/165$ with MR=5, see Figure 10.3B). This observation suggests that the conservation-based method may reduce false positives to identify functional modules.

To check the reduction of false positives in each module, I further compared the modules with MR = 10 (458 modules) and the modules identified by conserved coexpression (70 modules). I found that 47 modules were similar, where a pair of modules was judged to be similar if the number of common genes was significantly large (Fisher's exact test, p-value < 0.05 with Bonferroni correction). If a module had multiple similar modules, then only the mutually best pair was used. I also counted the number of genes with the representative

GO term of the module ($N_{\text{rep GO}}^{\text{gene}}$), and used the ratio to the number of genes in the module ($N_{\text{rep GO}}^{\text{gene}}/N^{\text{gene}}$) as an indicator to evaluate the goodness of the modules. If I assume that the representative GO term truly explains the function of a module, then a higher ratio indicates a better module explanation, or a module with fewer falsely related genes (or genes with different annotations). As a result, 13 out of 47 modules were found to share the same representative GO term, and the average ratio ($N_{\text{rep GO}}^{\text{gene}}/N^{\text{gene}}$) was 1.18 times higher in the conservation-based method than the COXPRESdb method. Notably, the raw number $N_{\text{rep GO}}^{\text{gene}}$ was also 1.18 times higher and the sizes of the conservation coexpression-based modules were larger than those of the COXPRESdb-based modules (Figure 10.3C), indicating that fewer falsely related genes were included in the modules.

Some examples of similar module pairs are shown in Figure 10.4. The first module pair in Figure 10.4 has different representative GO terms with 25 common genes, one for “skin development” and the other has no significant term, where the size of the conservation-based module ($97 = 72+25$) is much larger than that of COXPRESdb ($42 = 25 + 17$). The larger size and the existence of the representative GO term indicate the enhanced enrichment of the related genes. The second module pair also has a larger number of genes with the representative term in the conservation-based module (35) than that of COXPRESdb (24). Since it shares the same representative GO terms, the larger number of genes with the representative GO term may indicate the presence of a smaller number of related genes outside of the module. However, the ratio of the genes with a representative GO term for the conservation-based module (0.52) is smaller than that of COXPRESdb (0.62), which indicates the inclusion of a larger number of unrelated genes in the conservation-based modules. Since the conservation charts of the large module member genes have few flat regions in a small N range, the turning points of these genes were found in a large N range. Therefore, genes that are not directly related to a representative term may be included in the detected gene module. As described above, the conservation-based modules have better $N_{\text{rep GO}}^{\text{gene}}/N^{\text{gene}}$ ratios on

average, as in the case of the third example. However, in some cases the COXPRESdb-based modules produce better modules from the viewpoint of the inclusion of falsely related genes, as in the second example. In short, coexpression conservation may reduce the number of false negatives and false positives, to detect the functionally related genes on average.

10.4.1 Comparison of coexpression conservation between species

I compared coexpression similarity among 16 coexpression platforms of eleven species, including RNA-Seq based and microarray based, using mean of COeXpression SIMilarity (COXSIM) as described in section 11.3. I defined distances among coexpression platforms as mean values of COXSIMs.

As shown in Figure 10.5, coexpression platforms of one species belonged to same branch (shown in Figure 10.5 with blue dot lines) expects *Rattus norvegicus* (shown in Figure 10.5 with red dot lines). Since the number of samples in RNA-Seq based coexpression of *Rattus norvegicus* is small, the quality of the coexpression was not enough to compare coexpression similarities.

In mammalian, a dendrogram of coexpression similarity is almost consistent with NCBI taxonomy except *Canis lupus familiaris*. This result indicates that mammals have similar gene expression pattern, and similarities among these species is consistent with similarities of genomes. Since the coexpression of *Canis lupus familiaris* is calculated with small number of samples (636), the position in the dendrogram was inaccurate.

The dendrogram of other species, such as *Drosophila melanogaster* or *Caenorhabditis elegans*, was not consistent with NCBI taxonomy. For example, *Danio rerio* should be located beside *Gallus gallus*. This observation suggests that gene expression pattern of these species too divergent to discuss similarities of gene coexpression.

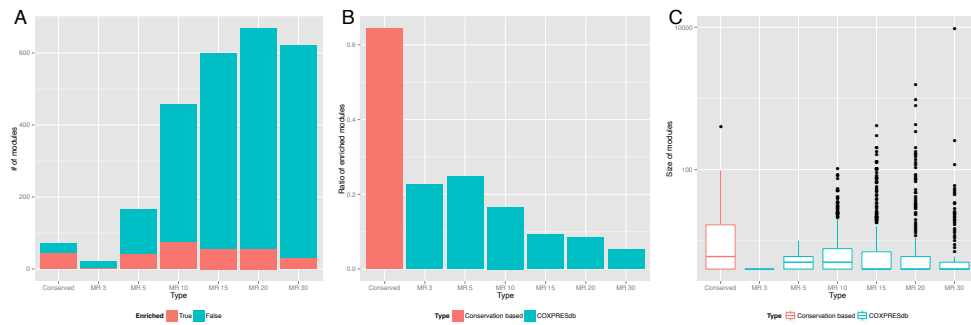


Figure 10.3: Comparison between the conserved coexpression-based modules and those based on coexpression without conservation

(A) The number of detected gene modules against MR for the coexpression-based method (left 6 bars) and the conservation-based method (right bar). The modules are colored according to whether a module had enriched GO terms. (B) The ratio of enriched gene modules. (C) A box plot of the gene module size distribution.

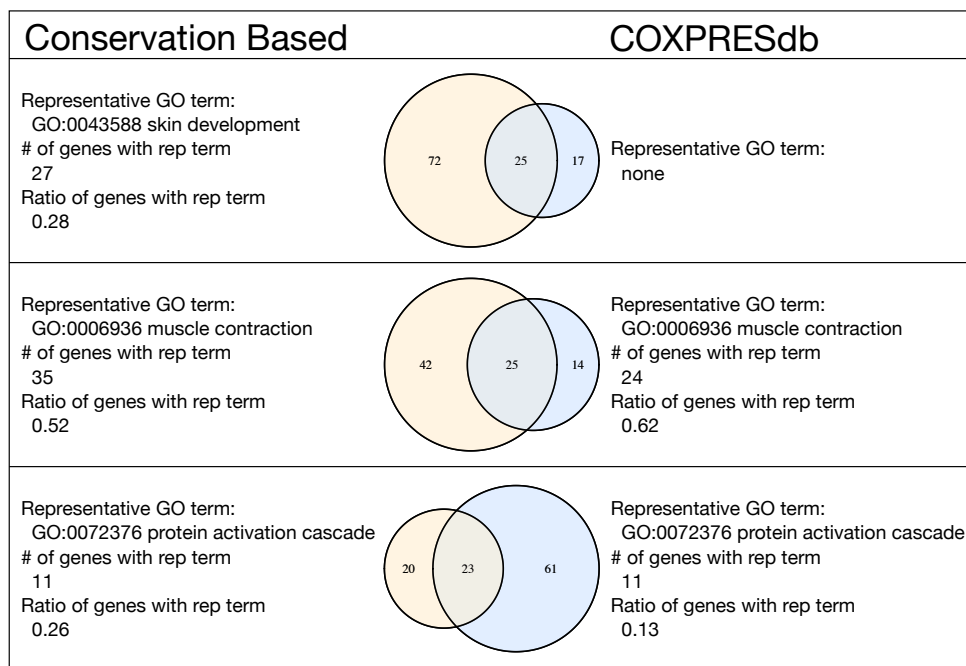


Figure 10.4: Example of the correspondence between the conservation-based method modules and the COXPRESdb-based modules

The three module pairs with the largest numbers of intersecting genes are shown.

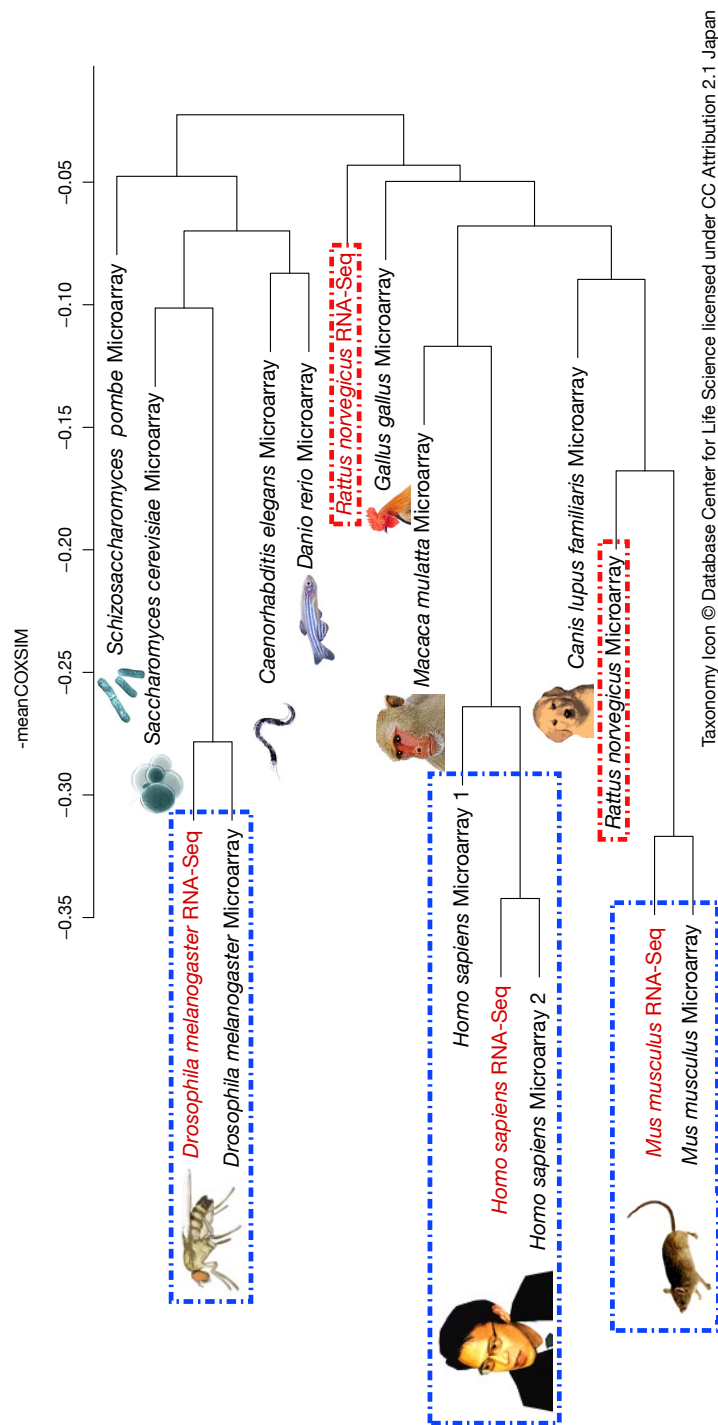


Figure 10.5: Dendrogram of coexpression similarity

10.4.2 Implementation of web-based database

All results of coexpression conservation, CC genes, and module detection are available through the web database named COeXpression SIMilarity DataBase (COXSIMdb, <http://v1.coxsimdb.info>). The overview of the database is shown in Figure 10.6. To use this web database, insert the gene symbol or entrez gene ID into the search field at the top of the COXSIMdb page (shown in Figure 10.6A). This web service provides a list of genes related to the query, with a view of the results of the coexpression conservation of a gene (Figure 10.6B). Figure 10.6C illustrates an example of a COXSIMdb main result view. The result view has up to 4 sections. The first section is a summary of the human and mouse genes and a conservation chart. The second section is a list of CC genes and any associated KEGG pathway. The third section is a list of detected gene modules that include the gene if it is involved in the modules. The gene modules detected with SCS=4 are shown in the default mode, but links to the modules detected with SCS=3 and SCS=5 are also provided. The last section is a table view of the comparison of coexpressed genes between human and mouse. Each gene is colored by the gene type and whether it is a CC gene, and homologous genes are shown in a pop-up window when the cursor moves over the genes.

A COXSIMdb
COExpression SIMilarity database
Input gene symbol or entrez gene ID
Gene / GO / Taxonomy QSearch
Coexpression Pair List

B Search DGAT2
DGAT2 Search
Genes
Click gene ID that you interested in
Gene ID Symbol Species Description
6664 DGAT2 Homo sapiens diacylglycerol O-acyltransferase 2
67800 Dgat2 Mus musculus diacylglycerol O-acyltransferase 2

C hsa-v13-01/DGAT2 - mmu-v13-01/Dgat2
Coexpression Pair hsa-v13-01 mmu-v13-01
Gene Summary
Target Gene DGAT2
Target Gene Descript... diacylglycerol O-acyltransferase 2
Reference Species Mus musculus
Reference Coexpress... mmu-v13-01 Dgat2
Reference Gene Dgat2
Target Gene Descript... diacylglycerol O-acyltransferase 2
Turning Point 37
Conservation Chart
Conserved Coexpression Genes List of CC genes
Gene ID Symbol Description AMPK ... PPAR ... Adipocytokine ... Metabolic ... Insulin ...
2819 GPD1 glycerol-3-phosphate dehydrogenase 1 (soluble)
7069 THRAP thyroid hormone responsive
56246 MRAP melanocortin 2 receptor accessory protein
1050 CEBPA CCAAT/enhancer binding protein (C/EBP) alpha
2180 ACSL1 acyl-CoA synthetase long-chain family member 1
3991 LIPE lipase, hormone-sensitive
5105 PCK1 phosphoenolpyruvate carboxykinase 1 (soluble)
50486 GSI2 GGI1 switch 2
364 AQP7 aquaporin 7
63924 CIDEA cell death-inducing DFFA-like effector c
1066 CES1 carboxylesterase 1
1149 CIDEA cell death-inducing DFFA-like effector a
KEGG Pathway List
KEGG Name Genes
AMPK signaling pathway - Homo sapiens (human) LIPE PCK1
PPAR signaling pathway - Homo sapiens (human) ACSL1 PCK1 AQP7
Adipocytokine signaling pathway - Homo sapiens (human) ACSL1 PCK1
Metabolic pathways - Homo sapiens (human) ACSL1 PCK1 CES1
Insulin signaling pathway - Homo sapiens (human) LIPE PCK1
Conservation Coexpression Gene Modules Detected gene modules
Module Module Representative GO Module Members
4 2D Detail 15 [GO:0019915] lipid storage GPD1 FABP4 PLIN4 LIPE ADIPOQ PPAR3 THRAP ACVR1C DGAT2 AQP7 CIDEA MRAP LIPE CIDEA CIDEA
Coexpression Compare Comparison of human and mouse coexpression
CC Genes N = 10 N = 25 N = 50 N = 2000
Not CC Genes N = 10 N = 25 N = 50 N = 2000
Homo sapiens Gene Homo sapiens MR Mus musculus Gene Mus musculus MR
1 PLIN4 4.9 Atgl1 1
2 MARCK1 5.29 Agpat2 2.24
3 SIRT1 6.93 LIPE 4.9
4 CIDEA 8.94 Tgat2 5.48
5 PLIN1 9 Adip 5.74
6 CIDEA 10.25 Agpat9 6.56
7 SIRT1 11.75 Pck1 7.75
8 KLB 12.65 Thrap 9.17
9 THRAP 12.65 Alas1 11.14
10 CEBPA 12.73 Tgat2 11.62
11 ECHDC3 13.19 Eghc2 12.96
12 SLC19A3 14.42 Tgat2 13.6
13 ADIPOQ 16.52 Acot1 14.07
14 LIPE 17.23 Tgat2 14.28

Figure 10.6: How to use COXSIMdb

(A) First, search for a gene by its symbol or entrez gene ID. (B) Second, select a gene of interest. (C) View of the coexpression conservation results. This view provides a summary of the genes, a list of CC genes, the detected gene modules, and a comparison of coexpression.

Chapter 11

Material and Methods

11.1 Dataset

All human and mouse coexpression data were obtained from COXPRESdb [61], versions Hsa.c4-1 (20,280 genes) and Mmu.c3-1 (20,959 genes), respectively. COXPRESdb is a database of co-regulated gene relationships. The coexpression strengths were obtained from COXPRESdb, and are represented by Mutual Rank (MR) [68]. MR is a rank-based measure, and smaller values indicate stronger coexpression. I prefer MR over the Pearson Correlation Coefficient (PCC), because MR shows better performance in GO prediction [68]. All homologous gene sets were obtained from HomoloGene [17], version build 65, and the genes that were not in HomoloGene were removed from the analyses. There were 18,981 human genes and 21,766 mouse genes in HomoloGene, and I used 14,611 homologous gene pairs between human and mouse in our analyses. I used Gene Ontology Terms (GO Terms) [69] to annotate the functions of the gene modules. The correspondence between the genes and the GO terms was obtained from the gene2go file in NCBI [17].

11.2 Detection of turning point and conserved coexpression genes

As described in the Results and Discussion section, I counted the number of human genes with mouse homologs to draw the conservation chart (Figure 10.1A and B), and then searched

for the lines with a turning point. It should be noted that I counted the number of human genes when a gene had multiple homologous genes in mouse. In other words, a human gene with two or more homologous genes in mouse was counted as one, while a mouse gene with two human homologs was counted twice.

In the example shown in Figure 10.1E, some conservation charts have two distinct regions, highly conserved and non-conserved, which can be detected as a turning point in the conservation chart. When a functional gene relationship is conserved between two species, the gene coexpression relationship will also be conserved. Therefore, to detect the functional modules, I tried to detect the turning point in each conservation chart.

The turning point is detected by focusing on the flat area in a conservation chart. If a gene module has k genes, then the two coexpression lists should have the same order in the top k genes, but the orders in the list after the k genes can be expected to be random. Therefore, if no new corresponding genes are found after the highly conserved region, it should be the turning point. I defined the turning point as the region with a 10-length flat region, which is a region with no new corresponding genes, and defined the conserved region as the region to the left of the turning point. I searched for turning points among the top 400 coexpressed genes.

When I also checked 5, 10, 15 and 20 as the length of the flat region to define the turning point, 1,890, 3,776, 3,478 and 2,783 non-redundant conserved-coexpressed genes (CC genes, as described below) were found, respectively. I selected the length of the flat region to maximize the number of CC genes. On the one hand, the use of flat regions longer than 10 to detect the turning point decreased the number of CC genes, because no flat region was found in the conservation chart. On the other hand, the shorter flat region also made the number of CC genes decrease, because turning points were found in the first position.

The genes in the conserved regions can be considered to have strong functional relationships. Therefore, I focused on the genes in the conserved regions, to emphasize their

strong relationship with the guide gene. Since some unrelated genes can be mixed in the coexpression lists due to coexpression noise, I used the genes mutually found in the conserved regions and named them CC genes. In other words, if gene A is the CC gene of guide gene B, then guide gene B should also be a CC gene of gene A. If there were multiple turning points, our turning point detection algorithm selected the first one of them, and tended to select the turning point at the smallest N.

Some genes did not have a flat area because their coexpression lists were highly conserved. I also generated a conserved coexpression gene network by using the following method. Since these genes did not have a flat area, I could not determine a turning point. I used 100 as the threshold of N instead of the turning point in these cases. Subsequently, I generated a coexpression gene network without a flat area, using the same procedure described above.

11.3 Definition of COXSIM

I defined $\text{COXSIM}_{\text{human}}(M, g_{\text{human}}, g_{\text{mouse}})$ to measure coexpression similarity with one value. The definition of this $\text{COXSIM}_{\text{human}}$ values is follows:

$$\text{COXSIM}_{\text{human}}(M, g_{\text{human}}, g_{\text{mouse}}) = \frac{\sum_{N=1}^M \text{number of corresponding genes}(N, g_{\text{human}}, g_{\text{mouse}})}{M(M+1)/2}$$

, where the value of M is a parameter to define the similarity, g_{human} and g_{mouse} are gene lists of human and mouse, and smaller values of M indicate that I focus on tightly coupled gene clusters. Very large M values have no meaning, because the minimum number of corresponding genes will increase. Compared with the similar formulation proposed by Yang *et al.* [98], our simple formulation can be applied to more complicated gene relationships including homologous genes.

COXSIM is sensitive to the parameter M by nature, but other rank-base indexes such as Spearman correlation coefficient are not suitable for this comparison. Since the importance

of genes is different between a strongly coexpressed gene and a lowly coexpressed gene, I introduced our original conservation similarity, COXSIM, to focus on strongly coexpressed genes. It emphasized the effects of tightly coexpressed genes because this score is cumulative value.

11.4 Analysis of the gene network and module detection

Since the CC genes are those with a tight functional relationship to the guide gene, I represented the relationship as a network, where a node indicated a gene and an edge represented a relationship between a CC gene and the guide gene.

Biological networks tend to be scale-free, with a small world network and a modular structure [99, 100, 101]. Since our network also had similar features, we applied a community detection algorithm implemented in networkx [102] to find the functional modules, according to Palla *et al.* [97]. To characterize the functional roles of the modules, enrichment analyses were performed, using TargetMine [103] and based on Fisher's exact test. I defined the representative GO term as the GO term with the smallest p-value in a module.

Since some gene modules had overlaps or similar annotations, we performed the module detection with three different strictness values, corresponding to the change in a parameter for the smallest clique size (SCS) used in Palla *et al.* [97]. Detection with a larger SCS yielded smaller and higher clustering coefficient modules. More precisely, I used three, four and five for the three different SCS values, and calculated the overlaps between the detected gene modules. Finally, I performed clustering of the gene modules by connecting the overlapped modules.

Chapter 12

Conclusion

In this part, I have described a new method to compare gene expression patterns by focusing on gene coexpression, to avoid the problem of sample matching. I also developed an algorithm to detect the conserved modules, and the GO term enrichment analyses revealed that the conserved gene modules have strong functional relationships. In other words, our method could detect some functional modules, without any prior knowledge. Many modules are well known, such as ribosomal protein or immune system, but some detected modules have significantly enriched GO terms, and thus they will be good candidates for further experimental analyses to identify the novel functional modules.

Part IV

The development of large dataset analysis helper tools

Chapter 13

Hyokai : A fast table viewer for big data analysis

13.1 Introduction

In bioinformatics research area, huge amount of data are generated everyday. Therefore, data handling is one of the fundamental problems to use such big data. Microsoft Excel is widely used as the first choice for users to view these large results. Excel is convenient to represent, visualize, filter and summarize data, but handling large data with Excel requires much machine power. In contrast, R language [104] and its GUI clients, such as Rstudio, are also powerful tools to summarize and visualize huge data, but users are required to learn R programming. In addition, R has no easy-to-use GUI to represent raw table data. Another solution to handle big data is SQL-based databases such as SQLite [105] and MySQL. They can store large data and look the data up fast, but users should learn SQL programming. To achieve powerful analytic environment, I developed a new GUI client for big data analysis based on SQLite and R.

13.2 Implementation

I implemented Hyokai, a fast table viewer for big data analysis. Hyokai is cross-platform, tested on Windows and Mac OS X. The interface is built on Qt, a cross-platform application and UI framework. Hyokai was implemented with C++ because C++ is one of the highest

performance programming languages. I use SQLite3 as database format, therefore, database can be accessed from C, Python, R, Ocaml or a lot of other programming languages. Since Hyokai was built on well developed file formats and GUI toolkit, Hyokai and work on various platforms and can be integrated with various programming languages.

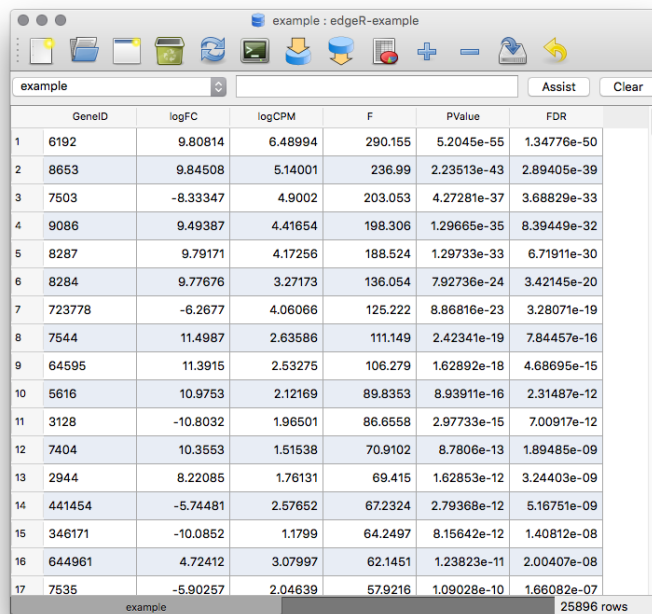
13.3 Features

This software has simple GUI, with powerful data analysis backends, SQLite and R language. Learning SQL and R language are not required for basic use, such as filtering data, joining two tables, plotting histograms or plotting scatter plots. For advanced use, such as combinations of joining and filtering or data grouping, SQL and R language are useful. Since Hyokai is based on a relational database, SQLite, it is easy to convert IDs, join two different tables or add annotations to rows. Since writing SQL codes to join tables is difficult for beginners, Hyokai has GUI wizard for joining tables. Designing a table schema to use SQL can also be problem for SQL beginners, but automatic table schema suggesting system is included in Hyokai. This system suggests a suitable table schema for tab-separated values (TSV) or comma separated values (CSV) that are exported from Excel or other software. Hyokai is also integrated with R language. This software can generate codes to export filtered or joined data to R. This software also has the visualizer to plot histograms and scatter plots.

13.4 Result

One of most usable application of Hyokai is differentially expressed gene (DEG) analysis. When I used edgeR [65] for DEG analysis, I have to deal with large table as shown in Figure 13.1. Since gene IDs in this figure were written in Entrez Gene ID, it is hard to associate gene IDs to gene functions. In this case, SQL Join Wizard (shown in Figure 13.2 and result of joining is shown in 13.3) is useful to convert to gene symbols and descriptions from gene IDs. Filtering rows by some conditions is also common task to analyze data. Hyokai can

filter rows with complex conditions by using SQL Where statement (shown in Figure 13.4).



	GeneID	logFC	logCPM	F	PValue	FDR
1	6192	9.80814	6.48994	290.155	5.2045e-55	1.34776e-50
2	8653	9.84508	5.14001	236.99	2.23513e-43	2.89405e-39
3	7503	-8.33347	4.9002	203.053	4.27281e-37	3.68829e-33
4	9086	9.49387	4.41654	198.306	1.29665e-35	8.39449e-32
5	8287	9.79171	4.17256	188.524	1.29733e-33	6.71911e-30
6	8284	9.77676	3.27173	136.054	7.92736e-24	3.42145e-20
7	723778	-6.2677	4.06066	125.222	8.86816e-23	3.28071e-19
8	7544	11.4987	2.63586	111.149	2.42341e-19	7.84457e-16
9	64595	11.3915	2.53275	106.279	1.62892e-18	4.68695e-15
10	5616	10.9753	2.12169	89.8353	8.93911e-16	2.31487e-12
11	3128	-10.8032	1.96501	86.6558	2.97733e-15	7.00917e-12
12	7404	10.3553	1.51538	70.9102	8.7806e-13	1.89485e-09
13	2944	8.22085	1.76131	69.415	1.62853e-12	3.24403e-09
14	441454	-5.74481	2.57652	67.2324	2.79368e-12	5.16751e-09
15	346171	-10.0852	1.1799	64.2497	8.15642e-12	1.40812e-08
16	644961	4.72412	3.07997	62.1451	1.23823e-11	2.00407e-08
17	7535	-5.90257	2.04639	57.9216	1.09028e-10	1.66082e-07

Figure 13.1: Hyokai Screen Shot

13.5 Availability

The source code of Hyokai is available at GitHub (<https://github.com/informationsea/Hyokai>) under GNU General Public License version 3. The binary distribution and sample databases are available without any fee at <https://hyokai.info>.

13.6 Conclusion

Hyokai is a use-friendly big data analysis tool. Hyokai enables fast viewing, filtering and summarizing big table. Since Hyokai uses common data format, SQLite, it is easy to exchange data with other programming languages.

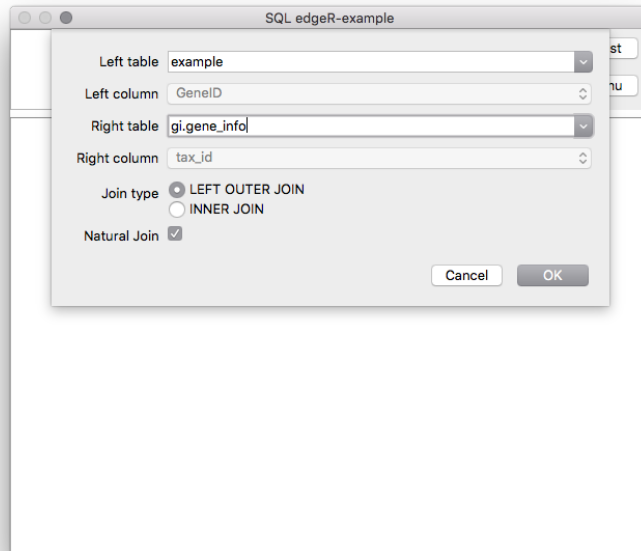


Figure 13.2: JOIN SQL Wizard

	GeneID	logFC	logCPM	F	Pvalue	FDR	Symbol	description	tax_id	LocusTag	Synonym
1	6192	9.80814	6.48994	290.155	5.2045e-55	1.34776e-50	RPS4Y1	ribosomal protein S4, Y-linked 1	9606	PRO2648	RPS4Y1
2	8653	9.84508	5.14001	236.99	2.23513e-43	2.69405e-39	DDX3Y	DEAD (Asp-Glu-Ala-Asp) box helicase 3, Y-linked	9606	-	DBY
3	7503	-8.33347	4.9002	203.053	4.27281e-37	3.68829e-33	XIST	X inactive specific transcript (non-protein coding)	9606	-	DXS108
4	9086	9.49387	4.41854	198.308	1.29685e-35	8.59449e-32	EIF1AY	eukaryotic translation initiation factor 1A, Y-linked	9606	-	wF-4C
5	8287	9.79171	4.17256	188.524	1.29733e-33	6.71911e-30	USP9Y	ubiquitin specific peptidase 9, Y-linked	9606	-	DFFRY1
6	8284	9.77676	3.27173	136.054	7.92736e-24	3.42145e-20	KDM5D	lysine (K)-specific demethylase 5D	9606	-	HYHYA
7	723778	-6.2677	4.06066	125.222	8.86816e-23	3.28071e-19	MIR650	microRNA 650	9606	-	MIRN65
8	7544	11.4387	2.63588	111.149	2.42341e-19	7.84457e-16	ZFY	zinc finger protein, Y-linked	9606	-	ZNF911
9	64595	11.5915	2.53275	106.279	1.62892e-18	4.68895e-15	TTY15	testis-specific transcript, Y-linked 15 (non-protein-coding)	9606	-	NCRNA1
10	5616	10.9753	2.12169	89.8353	8.93911e-16	2.31487e-12	PKXY	protein kinase, Y-linked, pseudogene	9606	-	PKCKP3
11	3128	-10.8032	1.96501	86.6558	2.97735e-15	7.00917e-12	HLA-DRB8	major histocompatibility complex, class II, DR beta 8	9606	-	-
12	7404	10.3553	1.51538	70.9102	8.78061e-13	1.89485e-09	UTY	ubiquitously transcribed tetratricopeptide repeat...	9606	-	KDM6A1
13	2944	8.22085	1.76131	69.415	1.62853e-12	3.24403e-09	GSTM1	glutathione S-transferase mu 1	9606	-	GST1/G1

Figure 13.3: JOIN SQL Result

example : edgeR-example

example Assist Clear

	GeneID	logFC	logCPM	F	PValue	FDR
1	6192	9.80814	6.48994	290.155	5.2045e-55	1.34776e-50
2	8653	9.84508	5.14001	236.99	2.23513e-43	2.89405e-39
3	7503	-8.33347	4.9002	203.053	4.27281e-37	3.68829e-33
4	9086	9.49387	4.41654	198.306	1.29665e-35	8.39449e-32
5	8287	9.79171	4.17256	188.524	1.29733e-33	6.71911e-30
6	8284	9.77676	3.27173	136.054	7.92736e-24	3.42145e-20
7	723778	-6.2677	4.06066	125.222	8.86816e-23	3.28071e-19
8	7544	11.4987	2.63586	111.149	2.42341e-19	7.84457e-16
9	64595	11.3915	2.53275	106.279	1.62892e-18	4.68695e-15
10	5616	10.9753	2.12169	89.8353	8.93911e-16	2.31487e-12
11	3128	-10.8032	1.96501	86.6558	2.97733e-15	7.00917e-12
12	7404	10.3553	1.51538	70.9102	8.7806e-13	1.89485e-09
13	2944	8.22085	1.76131	69.415	1.62853e-12	3.24403e-09
14	441454	-5.74481	2.57652	67.2324	2.79368e-12	5.16751e-09
15	346171	-10.0852	1.1799	64.2497	8.15642e-12	1.40812e-08
16	644961	4.72412	3.07997	62.1451	1.23823e-11	2.00407e-08
17	7535	-5.90257	2.04639	57.9216	1.09028e-10	1.66082e-07

example deg_with_info 46 rows

Figure 13.4: Filtering Rows

Chapter 14

DEG.js : A web-based RNA-Seq Analysis Tool

14.1 Introduction

Recent years, RNA-Seq is widely performed because of development of high throughput sequencing technology. To make use of RNA-Seq, bioinformatics plays a large role. Since RNA-Seq data is large and cannot interpret directly, it is hard to analyze without information science background. Existing widely used open-source methods requires command line operation. Some methods, such as RNASeqGUI [106], can analyze with graphical user interface, but they still hard to install into a local machine. To make RNA-Seq analysis easy without any complex operations, I developed *DEG.js*, a web-based RNA-Seq analysis platform. Since I built on standard web platform, users can perform simple analysis without downloading any software expect a modern browser.

14.2 Implementation

To make easy to use, client side software was implemented with JavaScript. Since recent modern browsers, such as Google Chrome, Firefox or Internet Explorer 11, support File API, a FASTQ file can be parsed with JavaScript on client browser. Server side software was implemented with JavaScript and C++. Since counting and quantifying gene expression require high calculation cost, these functions are implemented with C++. Other server side

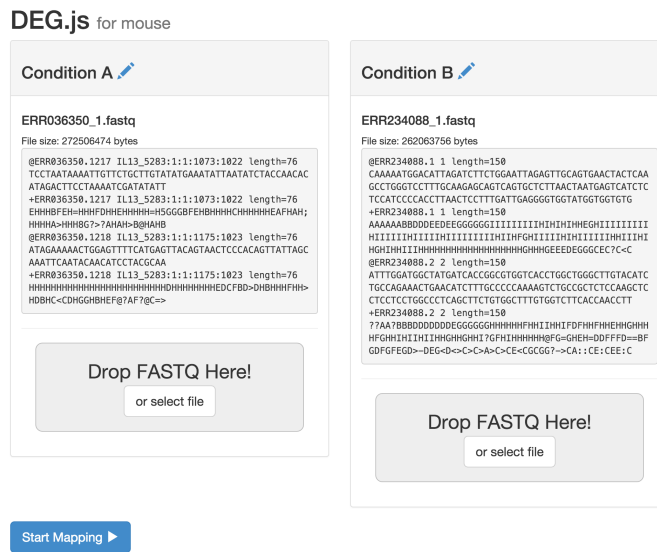


Figure 14.1: FASTQ Selection

software, such as communicating with client or handing counting and quantifying software, is implemented with JavaScript. To communicate a client and a server, I used socket.io.

14.3 Result

DEG.js is easy to use. When a user opened DEG.js with a browser, FASTQ selector is opened (shown in Figure 14.1). When the user selects FASTQ files and clicks “Start Mapping”, DEG.js starts mapping fragments and visualizing result. As shown in Figure 14.2, a result page is refreshing in real-time. The user can stop mapping and download a result at any time.

14.4 Conclusion

I developed easy to use RNA-Seq analysis tool, DEG.js. DEG.js do not require a user to install any software to use. Since any difficult handing, such as command line operation or downloading correct files, is not required to use DEG.js, any users who have basic computer skills can perform simple RNA-Seq analysis easily.

DEG.js for mouse

ERR036350_1.fastq

48.8%

ERR234088_1.fastq

50%

Export result as CSV

Show/Hide info

Export result as CSV

Show/Hide info

Cancel Mapping

Export Mapping as json

Which genes are you interested in?

Gene

Entrez Gene ID or Symbol

Add a gene

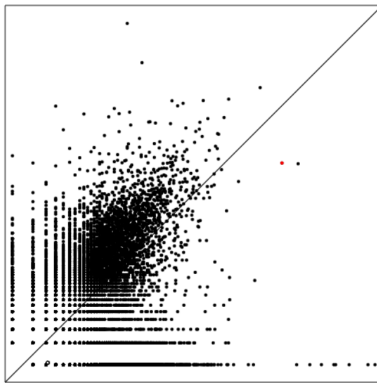
DEG result

Export DEG result as CSV

Export up-regulated DEG result as CSV

Export down-regulated DEG result as CSV

Scatter Plot



Rn28s1 28S ribosomal RNA

Gene Info	NCBI
logFC	-2.2524473860427525
Expression in condition A	6009.927904503014
Expression in condition B	1261.2902406609498

Up-regulated genes

Thbs1 (logFC: 9.31)
Ptx3 (logFC: 9.02)
Serpine1 (logFC: 8.97)
Acan (logFC: 7.82)
Pappa2 (logFC: 7.74)
Cemip (logFC: 7.72)
Hmox1 (logFC: 7.63)
Syt13 (logFC: 7.53)
Fmod (logFC: 7.49)
Fn1 (logFC: 7.18)
Fbln2 (logFC: 6.72)
Grem2 (logFC: 6.62)
Medag (logFC: 6.57)
Col5a2 (logFC: 6.48)
Ptgs2 (logFC: 6.39)

Figure 14.2: A result page of DEG.js

Conclusion

In this thesis, I focused fast meta-analysis method for RNA-Seq. The number of RNA-Seq data that are deposited in public database is quickly increasing because of spreading of high throughput sequencing technology. Although reanalyzing these a lot of RNA-Seq data is promising approach, few studies focused in meta-analysis of RNA-Seq. I resolve bottleneck problems to perform RNA-Seq meta-analysis and apply to gene function prediction using gene coexpression.

A most large problem of RNA-Seq meta-analysis is high calculation cost to estimate gene expression level from raw RNA-Seq data. To calculate gene expression level fast, I used two approaches: using only N-grams that are unique to each gene for mapping and skipping uninformative N-grams for mapping. Proposed method outperformed previous methods in both speed and accuracy. Proposed method is 300 times faster than previous alignment based methods and twice faster than a fastest previous alignment free method.

I applied proposed method to gene coexpression. Previously, gene coexpression was calculated from microarray-based gene expression. With proposed RNA-Seq quantification method, I succeeded to calculate RNA-Seq based gene coexpression in realistic time. When I applied RNA-Seq based gene coexpression to gene function prediction, RNA-Seq based coexpression shows better result than microarray based method in human and mouse. Since the number of samples plays a major role in quality of gene coexpression, prediction performances of RNA-Seq based gene coexpression in other species will be expected to exceed performances of microarray based gene coexpression in the future.

I also compared gene coexpression between species and predicted gene modules. As a result, I succeeded to predict more accurate functional gene modules by using coexpression conservation than single species coexpression.

Here, I resolved bottleneck of RNA-Seq based meta-analysis and performed meta-analysis in several species. Performance of RNA-Seq based coexpression in gene function prediction was better than that of microarray based coexpression. By introducing coexpression conservation, performance in functional gene module prediction was improved. In conclusion, my method enabled RNA-Seq based meta-analysis and allowed data-driven research using RNA-Seq data.

References

- [1] Sanger, F. and Coulson, A. A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase. *Journal of Molecular Biology*, 94(3), 1975, [http://dx.doi.org/10.1016/0022-2836\(75\)90213-2](http://dx.doi.org/10.1016/0022-2836(75)90213-2)
- [2] Sanger, F., Air, G., Barrell, B., et al. Nucleotide sequence of bacteriophage phi x174 DNA. *Nature*, 265(5596):687–695, 1977, <http://www.ncbi.nlm.nih.gov/pubmed/870828>
- [3] Fleischmann, R., Adams, M., White, O., et al. Whole-genome random sequencing and assembly of haemophilus influenzae rd. *Science (New York, N.Y.)*, 269(5223):496–512, 1995
- [4] Adams, M., Celniker, S., Holt, R., et al. The genome sequence of drosophila melanogaster. *Science (New York, N.Y.)*, 287(5461):2185–2195, 2000
- [5] Arabidopsis Genome Initiative. Analysis of the genome sequence of the flowering plant arabidopsis thaliana. *Nature*, 408(6814):796–815, 2000, <http://dx.doi.org/10.1038/35048692>
- [6] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001, <http://dx.doi.org/10.1038/35057062>
- [7] Venter, J., Adams, M., Myers, E., et al. The sequence of the human genome. *Science (New York, N.Y.)*, 291(5507):1304–1351, 2001, <http://dx.doi.org/10.1126/science.1058040>
- [8] Wood, V., Gwilliam, R., Rajandream, M.-A., et al. The genome sequence of schizosaccharomyces pombe. *Nature*, 415(6874):871–880, 2002, <http://dx.doi.org/10.1038/nature724>

- [9] Mouse Genome Sequencing Consortium et al. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420(6915):520–562, 2002, <http://dx.doi.org/10.1038/nature01262>
- [10] Benson, D. A., Cavanaugh, M., Clark, K., et al. GenBank. *Nucleic Acids Research*, 41(D1), 2013, <http://dx.doi.org/10.1093/nar/gks1195>
- [11] Leinonen, R., Akhtar, R., Birney, E., et al. The european nucleotide archive. *Nucleic acids research*, 39(Database issue):D28–D31, 2011, <http://dx.doi.org/10.1093/nar/gkq967>
- [12] Kosuge, T., Mashima, J., Kodama, Y., et al. DDBJ progress report: a new submission system for leading to a correct annotation. *Nucleic acids research*, 42(Database issue):D44–D49, 2014, <http://dx.doi.org/10.1093/nar/gkt1066>
- [13] Lipman, D. and Pearson, W. Rapid and sensitive protein similarity searches. *Science (New York, N.Y.)*, 227(4693):1435–1441, 1985, <http://dx.doi.org/10.1126/science.2983426>
- [14] Altschul, S., Gish, W., Miller, W., Myers, E., and Lipman, D. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990, [http://dx.doi.org/10.1016/S0022-2836\(05\)80360-2](http://dx.doi.org/10.1016/S0022-2836(05)80360-2)
- [15] Staden, R. A strategy of DNA sequencing employing computer programs. *Nucleic Acids Research*, 6(7), 1979, <http://dx.doi.org/10.1093/nar/6.7.2601>
- [16] Barba, M., Czosnek, H., and Hadidi, A. Historical perspective, development and applications of Next-Generation sequencing in plant virology. *Viruses*, 6(1), 2014, <http://dx.doi.org/10.3390/v6010106>
- [17] NCBI Resource Coordinators. Database resources of the national center for biotechnology information. *Nucleic acids research*, 43(Database issue):D6–17, 2015, <http://dx.doi.org/10.1093/nar/gku1130>
- [18] Kodama, Y., Shumway, M., Leinonen, R., and International Nucleotide Sequence Database Collaboration. The sequence read archive: explosive growth of sequencing data. *Nucleic acids research*, 40(Database issue):D54–D56, 2012, <http://dx.doi.org/10.1093/nar/gkr854>

-
- [19] Leinonen, R., Sugawara, H., Shumway, M., and International Nucleotide Sequence Database Collaboration. The sequence read archive. *Nucleic acids research*, 39(Database issue):D19–D21, 2011, <http://dx.doi.org/10.1093/nar/gkq1019>
- [20] Li, H. and Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics (Oxford, England)*, 25(14):1754–1760, 2009, <http://dx.doi.org/10.1093/bioinformatics/btp324>
- [21] Li, H. and Durbin, R. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics (Oxford, England)*, 26(5):589–595, 2010, <http://dx.doi.org/10.1093/bioinformatics/btp698>
- [22] Langmead, B., Trapnell, C., Pop, M., and Salzberg, S. L. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome biology*, 10(3):R25, 2009, <http://dx.doi.org/10.1186/gb-2009-10-3-r25>
- [23] Langmead, B. and Salzberg, S. L. Fast gapped-read alignment with bowtie 2. *Nature Methods*, 9(4), 2012, <http://dx.doi.org/10.1038/nmeth.1923>
- [24] Burrows, M. and Wheeler, D. A block-sorting lossless data compression algorithm. *SRC Research Report Digital Equipment Corporation*, 1994, <http://www.hpl.hp.com/techreports/Compaq-DEC/SRC-RR-124.pdf>
- [25] Ferragina, P. and Manzini, G. Opportunistic data structures with applications. *Proc. IEEE Symposium on Foundations of Computer Science (2000)*, 12, <http://dx.doi.org/10.1109/SFCS.2000.892127>
- [26] Ferragina, P. and Manzini, G. An experimental study of a compressed index. *Information Sciences*, 135(1-2), 2001, [http://dx.doi.org/10.1016/S0020-0255\(01\)00098-6](http://dx.doi.org/10.1016/S0020-0255(01)00098-6)
- [27] PLOS Biology. *Submission Guidelines*, <http://journals.plos.org/plosbiology/s/submission-guidelines>
- [28] Sitras, V., Fenton, C., and Acharya, G. Gene expression profile in cardiovascular disease and preeclampsia: a meta-analysis of the transcriptome based on raw data from human studies deposited in gene expression omnibus. *Placenta*, 36(2):170–178, 2015, <http://dx.doi.org/10.1016/j.placenta.2014.11.017>

- [29] Kim, H., Kim, J. H., Kim, S. Y., et al. Meta-Analysis of Large-Scale toxicogenomic data finds neuronal regeneration related protein and cathepsin d to be novel biomarkers of Drug-Induced toxicity. *PloS one*, 10(9):e0136698, 2015, <http://dx.doi.org/10.1371/journal.pone.0136698>
- [30] Alberts, B., Johnson, A., Lewis, J., et al. *Molecular Biology of THE CELL Fifth Edition*, pages 538–539. Garland Science, 2008
- [31] Alwine, J., Kemp, D., and Stark, G. Method for detection of specific RNAs in agarose gels by transfer to diazobenzyloxymethyl-paper and hybridization with DNA probes. *Proceedings of the National Academy of Sciences of the United States of America*, 74(12):5350–5354, 1977
- [32] Schena, M., Shalon, D., Davis, R., and Brown, P. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science (New York, N.Y.)*, 270(5235):467–470, 1995
- [33] Wickramasinghe, S., Cánovas, A., Rincón, G., and Medrano, J. F. RNA-Sequencing: a tool to explore new frontiers in animal genetics. *Livestock Science*, 166, 2014, <http://dx.doi.org/10.1016/j.livsci.2014.06.015>
- [34] Mortazavi, A., Williams, B. A., Kenneth, M., Schaeffer, L., and Wold, B. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature methods*, 5(7):621–628, 2008, <http://dx.doi.org/10.1038/nmeth.1226>
- [35] Wang, Z., Gerstein, M., and Snyder, M. RNA-Seq: a revolutionary tool for transcriptomics. *Nature reviews. Genetics*, 10(1):57–63, 2009, <http://dx.doi.org/10.1038/nrg2484>
- [36] Kim, D., Pertea, G., Trapnell, C., et al. TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biology*, 14(4):R36, 2013, <http://dx.doi.org/10.1186/gb-2013-14-4-r36>
- [37] Trapnell, C., Pachter, L., and Salzberg, S. L. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, 25(9), 2009, <http://dx.doi.org/10.1093/bioinformatics/btp120>

-
- [38] Trapnell, C., Roberts, A., Goff, L., et al. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and cufflinks. *Nature Protocols*, 7(3), 2014, <http://dx.doi.org/10.1038/nprot.2012.016>
- [39] Li, B. and Dewey, C. N. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC bioinformatics*, 12:323, 2011, <http://dx.doi.org/10.1186/1471-2105-12-323>
- [40] Roberts, A. and Pachter, L. Streaming fragment assignment for real-time analysis of sequencing experiments. *Nature Methods*, 2012, <http://dx.doi.org/10.1038/nmeth.2251>
- [41] Patro, R., Mount, S. M., and Kingsford, C. Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. *Nature Biotechnology*, 32(5), 2014, <http://dx.doi.org/10.1038/nbt.2862>
- [42] Zhang, Z. and Wang, W. RNA-Skim: a rapid method for RNA-Seq quantification at transcript level. *Bioinformatics*, 2014, <http://dx.doi.org/10.1093/bioinformatics/btu288>
- [43] Bray, N., Pimentel, H., Melsted, P., and Pachter, L. Near-optimal rna-seq quantification. *arXiv*, 12, <http://arxiv.org/abs/1505.02710>
- [44] Janzen, D., Tiourin, E., Salehi, J., et al. An apoptosis-enhancing drug overcomes platinum resistance in a tumour-initiating subpopulation of ovarian cancer. *Nature communications*, 6:7956, 2015, <http://dx.doi.org/10.1038/ncomms8956>
- [45] Madan, B., Ke, Z., Harmston, N., et al. Wnt addiction of genetically defined cancers reversed by PORCN inhibition. *Oncogene*, 2015, <http://dx.doi.org/10.1038/onc.2015.280>
- [46] Cacchiarelli, D., Trapnell, C., Ziller, M. J., et al. Integrative analyses of human reprogramming reveal dynamic nature of induced pluripotency. *Cell*, 162(2):412–424, 2015, <http://dx.doi.org/10.1016/j.cell.2015.06.016>
- [47] Lu, H., Li, Z., Zhang, W., et al. Gene target specificity of the super elongation complex (SEC) family: how HIV-1 tat employs selected SEC members to activate viral transcription. *Nucleic acids research*, 43(12):5868–5879, 2015, <http://dx.doi.org/10.1093/nar/gkv541>

- [48] Wu, Y., Wang, X., Wu, F., et al. Transcriptome profiling of the cancer, adjacent non-tumor and distant normal tissues from a colorectal cancer patient by deep sequencing. *PloS one*, 7(8):e41001, 2012, <http://dx.doi.org/10.1371/journal.pone.0041001>
- [49] Zhang, J., Lieu, Y. K., Ali, A. M., et al. Disease-associated mutation in SRSF2 misregulates splicing by altering RNA-binding affinities. *Proceedings of the National Academy of Sciences of the United States of America*, 112(34):E4726–E4734, 2015, <http://dx.doi.org/10.1073/pnas.1514105112>
- [50] Bloom, B. H. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7), 1970, <http://dx.doi.org/10.1145/362686.362692>
- [51] FAL Labs. *KyotoCabinet*, 2011, <http://fallabs.com/kyotocabinet/>
- [52] Tseng, G. C., Ghosh, D., and Feingold, E. Comprehensive literature review and statistical considerations for microarray meta-analysis. *Nucleic Acids Research*, 40(9):3785–3799, 2012, <http://dx.doi.org/10.1093/nar/gkr1265>
- [53] Tang, Z., Ow, G. S., Thiery, J. P., Ivshina, A. V., and Kuznetsov, V. A. Meta-analysis of transcriptome reveals let-7b as an unfavorable prognostic biomarker and predicts molecular and clinical subclasses in high-grade serous ovarian carcinoma. *International Journal of Cancer*, 134(2):306–318, 2014, <http://dx.doi.org/10.1002/ijc.28371>
- [54] Tomczak, K., Czerwinska, P., and Wiznerowicz, M. Review the cancer genome atlas (TCGA): an immeasurable source of knowledge. *Contemporary Oncology*, 1A:68–77, 2015, <http://dx.doi.org/10.5114/wo.2014.47136>
- [55] Mabbott, N. A. and Gray, D. Identification of co-expressed gene signatures in mouse b1, marginal zone and b2 b-cell populations. *Immunology*, 141(1):79–95, 2014, <http://dx.doi.org/10.1111/imm.12171>
- [56] Bottcher, C., Chapman, A., Fellermeier, F., et al. The biosynthetic pathway of Indole-3-Carbaldehyde and Indole-3-Carboxylic acid derivatives in arabidopsis. *PLANT PHYSIOLOGY*, 165(2):841–853, 2014, <http://dx.doi.org/10.1104/pp.114.235630>
- [57] Bodt, S. D., Hollunder, J., Nelissen, H., Meulemeester, N., and Inzé, D. CORNET 2.0: integrating plant coexpression, protein-protein interactions, regulatory interactions, gene associations and functional annotations. *New Phytologist*, 195(3):707–720, 2012, <http://dx.doi.org/10.1111/j.1469-8137.2012.04184.x>

-
- [58] Jupiter, D., Chen, H., and Vincent, V. STARNET 2: a web-based tool for accelerating discovery of gene regulatory networks using microarray co-expression data. *BMC bioinformatics*, 10:332, 2009, <http://dx.doi.org/10.1186/1471-2105-10-332>
- [59] Dam, S., Craig, T., and Magalhães, J. P. P. GeneFriends: a human RNA-seq-based gene and transcript co-expression database. *Nucleic acids research*, 43(Database issue):D1124–D1132, 2015, <http://dx.doi.org/10.1093/nar/gku1042>
- [60] Dobin, A., Davis, C., Schlesinger, F., and Drenkow, J. STAR: ultrafast universal RNA-seq aligner. 2013, <http://dx.doi.org/10.1093/bioinformatics/bts635>
- [61] Okamura, Y., Aoki, Y., Obayashi, T., et al. COXPRESdb in 2015: coexpression database for animal species by DNA-microarray and RNAseq-based expression data with multiple quality assessment systems. *Nucleic acids research*, 43(Database issue):D82–D86, 2015, <http://dx.doi.org/10.1093/nar/gku1163>
- [62] Obayashi, T., Okamura, Y., Ito, S., et al. ATTED-II in 2014: evaluation of gene coexpression in agriculturally important plants. *Plant & cell physiology*, 55(1):e6, 2014, <http://dx.doi.org/10.1093/pcp/pct178>
- [63] EBI. *Statistics About the european nucleotide archive.*, 2015, <http://www.ebi.ac.uk/ena/about/statistics>
- [64] University of Liverpool. *GridEngine*, 2014, <https://arc.liv.ac.uk/trac/SGE>
- [65] Robinson, M. D., McCarthy, D. J., and Smyth, G. K. edgeR: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140, 2010, <http://dx.doi.org/10.1093/bioinformatics/btp616>
- [66] Bolstad, B. M. *preprocessCore: A collection of pre-processing functions*. R package version 1.30.0
- [67] Danielsson, F., James, T., Gomez-Cabrero, D., and Huss, M. Assessing the consistency of public human tissue RNA-seq data sets. *Briefings in Bioinformatics*, 2015, <http://dx.doi.org/10.1093/bib/bbv017>
- [68] Obayashi, T. and Kinoshita, K. Rank of correlation coefficient as a comparable measure for biological significance of gene coexpression. *DNA research : an international*

journal for rapid publication of reports on genes and genomes, 16(5):249–260, 2009, <http://dx.doi.org/10.1093/dnares/dsp016>

- [69] Gene Ontology Consortium, Blake, J., Dolan, M., et al. Gene ontology annotations and resources. *Nucleic acids research*, 41(Database issue):D530–D535, 2013, <http://dx.doi.org/10.1093/nar/gks1050>
- [70] Mailman, M. D., Feolo, M., Jin, Y., et al. The NCBI dbGaP database of genotypes and phenotypes. *Nature Genetics*, 39(10), 2007, <http://dx.doi.org/10.1038/ng1007-1181>
- [71] Shalek, A. K., Satija, R., Shuga, J., et al. Single-cell RNA-seq reveals dynamic paracrine control of cellular variation. *Nature*, 510(7505), 2014, <http://dx.doi.org/10.1038/nature13437>
- [72] International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–945, 2004, <http://dx.doi.org/10.1038/nature03001>
- [73] Sim, G., Kafatos, F., Jones, C., et al. Use of a cDNA library for studies on evolution and developmental expression of the chorion multigene families. *Cell*, 18(4):1303–1316, 1979, [http://dx.doi.org/10.1016/0092-8674\(79\)90241-1](http://dx.doi.org/10.1016/0092-8674(79)90241-1)
- [74] Mutsumi, K., Itoh, M., Kawaji, H., et al. Unamplified cap analysis of gene expression on a single-molecule sequencer. *Genome research*, 21(7):1150–1159, 2011, <http://dx.doi.org/10.1101/gr.115469.110>
- [75] Lindblad-Toh, K., Garber, M., Zuk, O., et al. A high-resolution map of human evolutionary constraint using 29 mammals. *Nature*, 478(7370):476–482, 2011, <http://dx.doi.org/10.1038/nature10530>
- [76] Chimpanzee Sequencing and Analysis Consortium. Initial sequence of the chimpanzee genome and comparison with the human genome. *Nature*, 437(7055):69–87, 2005, <http://dx.doi.org/10.1038/nature04072>
- [77] Harrow, J., Frankish, A., Gonzalez, J., et al. GENCODE: the reference human genome annotation for the ENCODE project. *Genome research*, 22(9):1760–1774, 2012, <http://dx.doi.org/10.1101/gr.135350.111>

-
- [78] Andersson, R., Gebhard, C., Irene, M., et al. An atlas of active enhancers across human cell types and tissues. *Nature*, 507(7493):455–461, 2014, <http://dx.doi.org/10.1038/nature12787>
- [79] Hartwell, L., Hopfield, J., Leibler, S., and Murray, A. From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–C52, 1999, <http://dx.doi.org/10.1038/35011540>
- [80] Fields, S. and Song, O. A novel genetic system to detect protein-protein interactions. *Nature*, 340(6230):245–246, 1989, <http://dx.doi.org/10.1038/340245a0>
- [81] Ito, T., Chiba, T., Ozawa, R., et al. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proceedings of the National Academy of Sciences of the United States of America*, 98(8):4569–4574, 2001, <http://dx.doi.org/10.1073/pnas.061034498>
- [82] Uetz, P., Giot, L., Cagney, G., et al. A comprehensive analysis of protein-protein interactions in *saccharomyces cerevisiae*. *Nature*, 403(6770):623–627, 2000, <http://dx.doi.org/10.1038/35001009>
- [83] Chatr-Aryamontri, A., Breitkreutz, B., Heinicke, S., et al. The BioGRID interaction database: 2013 update. *Nucleic acids research*, 41(Database issue):D816–D823, 2013, <http://dx.doi.org/10.1093/nar/gks1158>
- [84] Spellman, P., Sherlock, G., Zhang, M., et al. Comprehensive identification of cell cycle-regulated genes of the yeast *saccharomyces cerevisiae* by microarray hybridization. *Molecular biology of the cell*, 9(12):3273–3297, 1998
- [85] Wu, L., Hughes, T., Davierwala, A., et al. Large-scale prediction of *saccharomyces cerevisiae* gene function using overlapping transcriptional clusters. *Nature genetics*, 31(3):255–265, 2002, <http://dx.doi.org/10.1038/ng906>
- [86] Bornholdt, J., Friis, S., Godiksen, S., et al. The level of claudin-7 is reduced as an early event in colorectal carcinogenesis. *BMC cancer*, 11:65, 2011, <http://dx.doi.org/10.1186/1471-2407-11-65>
- [87] Allocco, D., Kohane, I., and Butte, A. Quantifying the relationship between co-expression, co-regulation and gene function. *BMC bioinformatics*, 5:18, 2004, <http://dx.doi.org/10.1186/1471-2105-5-18>

- [88] Usadel, B., Obayashi, T., Mutwil, M., et al. Co-expression tools for plant biology: opportunities for hypothesis generation and caveats. *Plant, cell & environment*, 32(12):1633–1651, 2009, <http://dx.doi.org/10.1111/j.1365-3040.2009.02040.x>
- [89] Shi, Z., Derow, C., and Zhang, B. Co-expression module analysis reveals biological processes, genomic gain, and regulatory mechanisms associated with breast cancer progression. *BMC systems biology*, 4:74, 2010, <http://dx.doi.org/10.1186/1752-0509-4-74>
- [90] Stuart, J., Segal, E., Koller, D., and Kim, S. A gene-coexpression network for global discovery of conserved genetic modules. *Science (New York, N.Y.)*, 302(5643):249–255, 2003, <http://dx.doi.org/10.1126/science.1087447>
- [91] ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74, 2012, <http://dx.doi.org/10.1038/nature11247>
- [92] Pontius, J., Mullikin, J., Smith, D., et al. Initial sequence and comparative analysis of the cat genome. *Genome research*, 17(11):1675–1689, 2007, <http://dx.doi.org/10.1101/gr.6380007>
- [93] Su, A., Cooke, M., Ching, K., et al. Large-scale analysis of the human and mouse transcriptomes. *Proceedings of the National Academy of Sciences of the United States of America*, 99(7):4465–4470, 2002, <http://dx.doi.org/10.1073/pnas.012025199>
- [94] Brawand, D., Soumillon, M., Necsulea, A., et al. The evolution of gene expression levels in mammalian organs. *Nature*, 478(7369):343–348, 2011, <http://dx.doi.org/10.1038/nature10532>
- [95] Wise, A., Oltvai, Z., and Ziv, B. Matching experiments across species using expression values and textual information. *Bioinformatics (Oxford, England)*, 28(12):i258–i264, 2012, <http://dx.doi.org/10.1093/bioinformatics/bts205>
- [96] Smoot, M., Ono, K., Ruscheinski, J., Wang, P., and Ideker, T. Cytoscape 2.8: new features for data integration and network visualization. *Bioinformatics (Oxford, England)*, 27(3):431–432, 2011, <http://dx.doi.org/10.1093/bioinformatics/btq675>
- [97] Palla, G., Derényi, I., Farkas, I., and Vicsek, T. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005, <http://dx.doi.org/10.1038/nature03607>

-
- [98] Yang, X., Bentink, S., Scheid, S., and Spang, R. Similarities of ordered gene lists. *Journal of bioinformatics and computational biology*, 4(3):693–708, 2006, <http://dx.doi.org/10.1142/s0219720006002120>
- [99] Rives, A. W. and Galitski, T. Modular organization of cellular networks. *Proceedings of the National Academy of Sciences*, 100(3):1128–1133, 2003, <http://dx.doi.org/10.1073/pnas.0237338100>
- [100] Spirin, V. and Mirny, L. A. Protein complexes and functional modules in molecular networks. *Proceedings of the National Academy of Sciences*, 100(21):12123–12128, 2003, <http://dx.doi.org/10.1073/pnas.2032324100>
- [101] Chen, J. and Yuan, B. Detecting functional modules in the yeast protein-protein interaction network. *Bioinformatics (Oxford, England)*, 22(18):2283–2290, 2006, <http://dx.doi.org/10.1093/bioinformatics/btl370>
- [102] Hagberg, A. A., Schult, D. A., and Swart, P. J. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pages 11–15, Pasadena, CA USA, August 2008
- [103] Chen, Y., Tripathi, L., and Mizuguchi, K. TargetMine, an integrated data warehouse for candidate gene prioritisation and target discovery. *PloS one*, 6(3), 2011, <http://dx.doi.org/10.1371/journal.pone.0017844>
- [104] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2015, <https://www.R-project.org/>
- [105] *SQLite*, <https://www.sqlite.org/>
- [106] Russo, F. and Angelini, C. RNASeqGUI: a GUI for analysing RNA-Seq data. *Bioinformatics (Oxford, England)*, 30(17):2514–2516, 2014, <http://dx.doi.org/10.1093/bioinformatics/btu308>

Publications & Presentations

Papers

- Aoki, Y., Okamura, Y., Ohta, H, Kinoshita, K. and Obayashi “ALCOdb: Gene Coexpression Database for Microalgae” *Plant and Cell Physiology* (2016) doi:10.1093/pcp/pcv190
- Aoki, Y.†, Okamura, Y.†, Tadaka, S., Kinoshita, K. and Obayashi, T. “ATTED-II in 2016: A Plant Coexpression Database Towards Lineage-Specific Coexpression” *Plant and Cell Physiology* (2016) doi:10.1093/pcp/pcv165
- Okamura, Y., Obayashi, T. & Kinoshita, K. Comparison of Gene Coexpression Profiles and Construction of Conserved Gene Networks to Find Functional Modules. *PLoS ONE* (2015). doi:10.1371/journal.pone.0132039
- Okamura, Y.†, Aoki, Y.†, Obayashi, T.†, Tadaka, S., Ito, S., Narise, T. and Kinoshita, K. COXPRESdb in 2015: coexpression database for animal species by DNA-microarray and RNAseq-based expression data with multiple quality assessment systems. *Nucleic Acids Res*, 2014 Nov 11; doi:10.1093/nar/gku1163
- Katayama, T., Wilkinson, MD., Aoki-Kinoshita, KF., Kawashima, S., Yamamoto, Y., Yamaguchi, A., Okamoto, S., Kawano, S., Kim, JD., Wang, Y., Wu H., Kano, Y., Ono, H., Bono, H., Kocbek, S., Aerts, J., Akune, Y., Antezana, E., Arakawa, K., Aranda, B., Baran, J., Bolleman, J., Bonnal, RJ., Buttigieg PL., Campbell, MP., Chen, YA., Chiba, H., Cock, PJ., Cohen, KB., Constantin A., Duck, G., Dumontier, M., Fujisawa, T., Fujiwara, T., Goto, N., Hoehndorf R., Igarashi, Y., Itaya, H., Ito, M., Iwasaki, W., Kalas, M., Katoda, T., Kim T., Kokubu, A., Komiyama, Y., Kotera, M., Laibe, C., Lapp, H., Lütteke, T., Marshall, MS., Mori, T., Mori, H., Morita, M., Murakami, K., Nakao, M., Narimatsu, H., Nishide, H., Nishimura, Y., Nystrom-Persson, J., Ogishima S., Okamura, Y., Okuda, S., Oshita, K., Packer, NH., Prins, P., Ranzinger, R., Rocca-Serra, P., Sansone, S., Sawaki, H., Shin, SH., Splendiani, A., Strozzi, F., Tadaka,

- S., Toukach, P., Uchiyama, I., Umezaki, M., Vos, R., Whetzel, PL., Yamada, I., Yamasaki, C., Yamashita, R., York, WS., Zmasek CM., Kawamoto, S. and Takagi, T., BioHackathon series in 2011 and 2012: penetration of ontology and linked data in life science domains. *J Biomed Semantics*, 2014 Feb 5;5(1):5 doi:10.1186/2041-1480-5-5
- Obayashi, T., Okamura, Y., Ito, S., Tadaka, S., Aoki, Y., Shirota, M. and Kinoshita, K.. ATTED-II in 2014: evaluation of gene coexpression in agriculturally important plants. *Plant Cell Physiol*, 2014 Jan;55(1):e6 doi:10.1093/pcp/pct178
 - Obayashi, T., Okamura, Y., Ito, S., Tadaka, S., Motoike, I.N., and Kinoshita, K. COXPRESdb: a database of comparative gene coexpression networks of eleven species for mammals. *Nucleic Acids Research*, 2013. doi:10.1093/nar/gks1014

†: equally contributed author

Awards

- ポスター賞
機械学習を用いた遺伝子発現プロファイルの分類, 岡村容伸, 第三回生命医薬情報学連合大会 2014, 平成26年10月4日, 仙台国際センター
- **Excellent Poster Award**
Yasunobu Okamura, Takeshi Obayashi, Kengo Kinoshita, GO analysis of gene expression patterns comparison among organs and species, 生命医薬情報学連合大会, 平成24年10月16日, タワーホール船橋

Oral presentation

Domestic Society

- 岡村容伸, 大林武, 木下賢吾 「DEG.js : Streaming web-based differential expression genes analysis tool for RNA-seq」, 生命情報科学若手の会 第6回研究会、2014年10月30日、理化学研究所 発生・再生科学総合研究センター (兵庫県)
- 岡村容伸, 田高周, 大林武, 木下賢吾. 「Hyokai: Powerful data viewing and analysis tool for big tables.」, 生命情報科学若手の会 第五回研究会, 2014年2月17日~19日, 東京大学検見川セミナーハウス (千葉県)

- 岡村容伸, 大林武, 木下賢吾. 「遺伝子発現パターンの比較による遺伝子機能ごとの発現解析」, 生命情報科学若手の会第4回研究会, 2013年3月2日, 岡崎
- 岡村容伸, 大林武, 木下賢吾, 生物種間の遺伝子発現パターンの比較解析, 第28回バイオ情報学研究発表会, 平成24年3月28日, 東北大学
- Yasunobu Okamura, Takeshi Obayashi, Kengo Kinoshita, GO analysis of gene expression patterns comparison among organs and species, 生命医薬情報学連合大会, 平成24年10月16日, タワーホール船橋
- 岡村容伸, 大林武, 木下賢吾「ヒトとマウスの遺伝子発現パターンの比較解析」第25回生体生命工学研究会, 平成23年10月26日, 東北大学
- 岡村容伸, 大林武, 木下賢吾, 遺伝子発現パターンから見た生物の進化, 第27回生体生命工学研究会, 平成24年11月19日, 東北大学

Poster presentation

International Society

- Yasunobu Okamura and Kengo Kinoshita, “Fast gene-expression quantification tool for massive RNA-sequence analysis”, 23rd Annual International Conference on Intelligent Systems for Molecular Biology, July 12-14, 2015, Dublin Ireland
- Yasunobu Okamura, Takeshi Obayashi and Kengo Kinoshita, "RNA-seq profile classification by machine learning", Genome Informatics Workshop 2014, December 15-18, 2014, Tokyo Japan
- Yasunobu Okamura, Takeshi Obayashi and Kengo Kinoshita “Classify RNA-seq runs as origin organs or other features by using machine learning”, 22nd Annual International Conference on Intelligent Systems for Molecular Biology, July 13-15, 2014, Boston USA.
- Yasunobu Okamura, Takeshi Obayashi and Kengo Kinoshita. “Functional gene network prediction based on conservation of gene expression patterns.”, 21st Annual International Conference on Intelligent Systems for Molecular Biology, July 23, 2013, Berlin, Germany

- Y Okamura, T Obayashi, K Kinoshita, Comparative analysis of gene expression pattern similarity between human and mouse by gene functions, 20th Annual International Conference on Intelligent Systems for Molecular Biology, July 16, 2012, Long Beach USA

Domestic Society

- Yasunobu Okamura, Kengo Kinoshita, “An ultra fast gene expression quantification tool using unique N-gram”, 生命医薬情報連合大会2015, 2015年10月29-31日, 京都大学宇治キャンパス (京都府)
- 岡村容伸、大林武、木下賢吾、「機械学習を用いた遺伝子発現プロファイルの分類」生命医薬情報連合大会2014、2014年10月03日、仙台国際センター (宮城県)
- Yasunobu Okamura, Takeshi Obayashi, Kengo Kinoshita. “Gene network and gene module prediction based on coexpression conservation” 第36回日本分子生物学会年会, 2013年12月4日, 神戸
- Yasunobu Okamura, Shu Tadaka, Takeshi Obayashi, Kengo Kinoshita. “Hyokai: The fast table viewer for big data analysis” 第2回 生命医薬情報学連合大会, 2013年10月29 - 31日, タワーホール船橋 (東京)
- Yasunobu Okamura, Takeshi Obayashi, Kengo Kinoshita, “GO analysis of gene expression patterns comparison among organs and species”, 生命医薬情報学連合大会, 平成24年10月15-17日, タワーホール船橋
- 岡村容伸, 大林武, 木下賢吾「7生物種における遺伝子共発現の比較と保存共発現の有効性」第13回日本進化学会 平成23年7月29-31日, 京都

Acknowledgment

I would like to give heartfelt thanks to Prof. Kengo Kinoshita whose comments and suggestions were innumerably valuable throughout the course of my study. Special thanks also go to Dr. Takeshi Obayashi, Dr. Hafumi Nishi and other lab members whose comments made enormous contribution to my work. I would also like to express my gratitude to my family for their moral support and warm encouragements.

Part V
Appendix

Appendix A

Supplementary Figures for Matataki

A.1 Mapping result detail of SRR1639212

The experiment accession for SRR1639212 is SRX750225, and the study accession for SRR1639212 is SRP048993. The study abstract is “Stem cell differentiation timecourse, six time points through induction from induced pluripotency (day0) towards beating cardiomyocytes, mature at day14. Accompanying study investigates careful differentiation protocols.”. Figure A.1 shows a distribution of FPKM and mapping rates of Matataki when parameters were varied. Figure A.2, A.3, A.4 and A.5 show comparison with eXpress results.

A.2 Mapping result detail of ERR266335

The experiment accession for ERR266335 is ERX182652, and the study accession for ERR266335 is ERP002045. The study title is “Transcriptional and epigenetic profiling of the progression of hESCs to beta cells”. Figure A.6 shows a distribution of FPKM and mapping rates of Matataki when parameters were varied. Figure A.7, A.8, A.9 and A.10 show comparison with eXpress results.

A.3 Mapping result detail of SRR1013361

The experiment accession for SRR1013361 is SRX365386, and the study accession for SRR1013361 is SRP031478. The study title is “Altered Epigenetic Regulation of Homeobox Genes in Human Oral Squamous Cell Carcinoma Cells”. Figure A.11 shows a distribution of FPKM and mapping rates of Matataki when parameters were varied. Figure A.12, A.13, A.14 and A.15 show comparison with eXpress results.

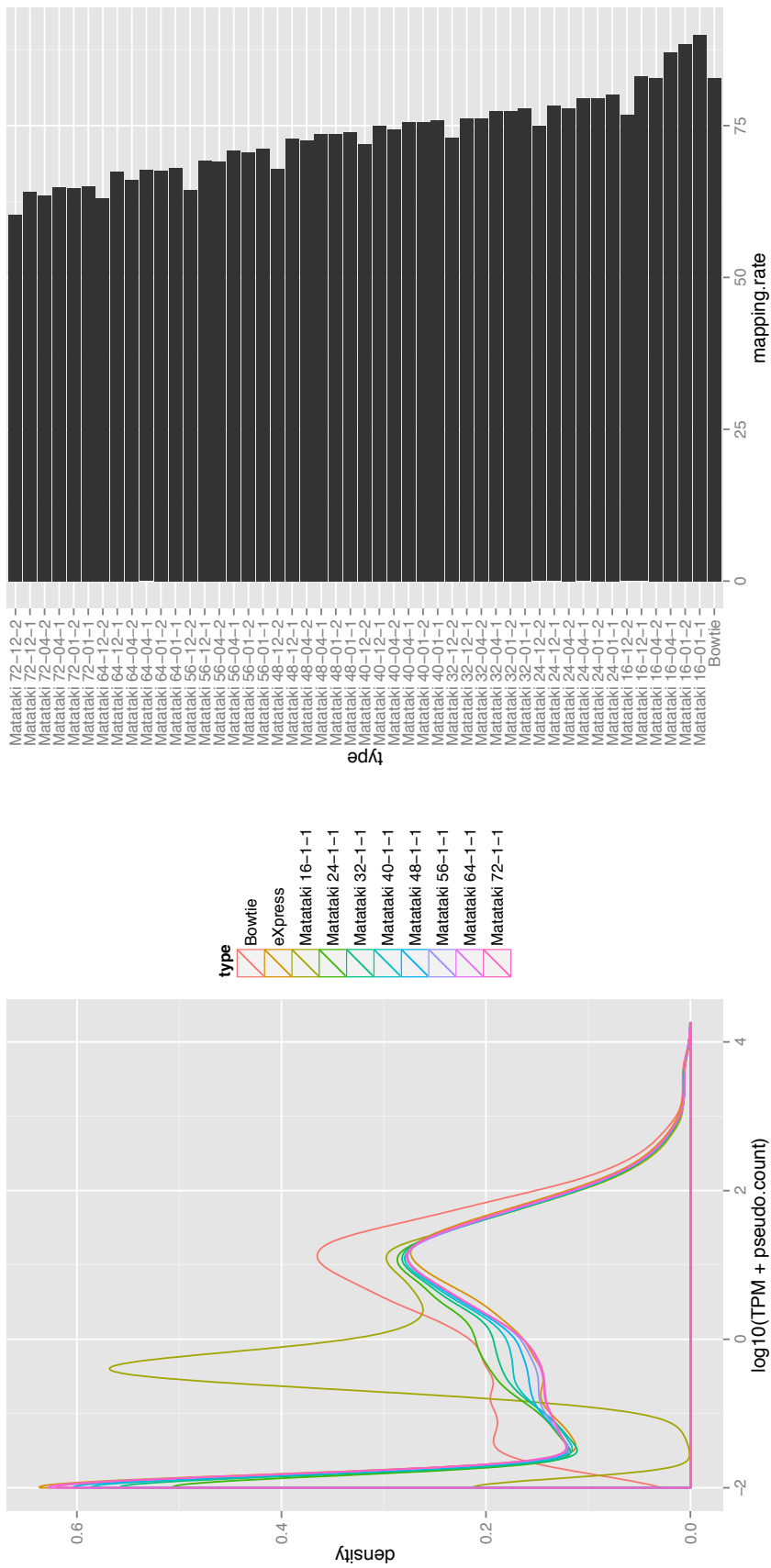


Figure A.1: SRR1639212: A distribution of FPKM and mapping rate

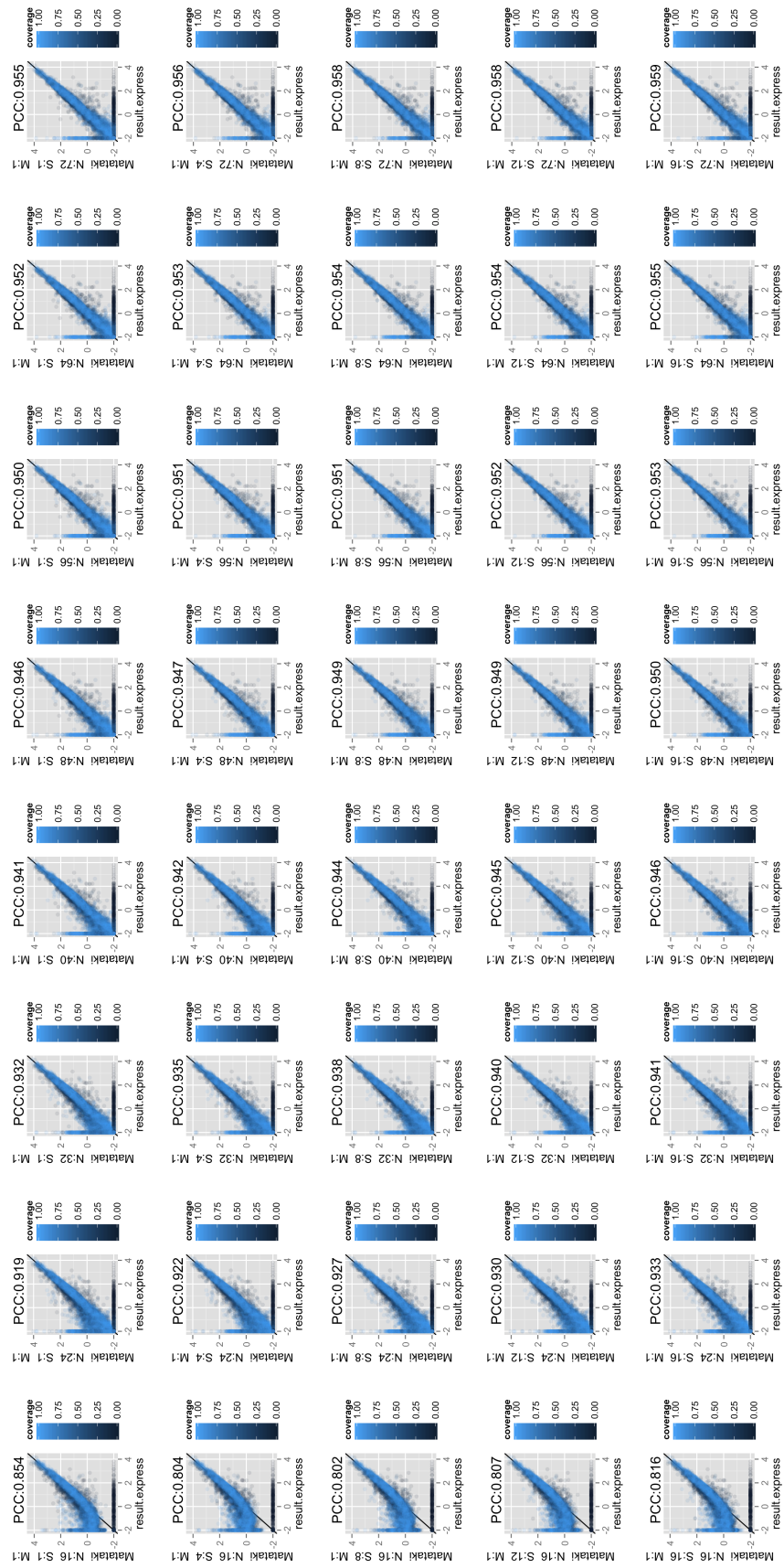
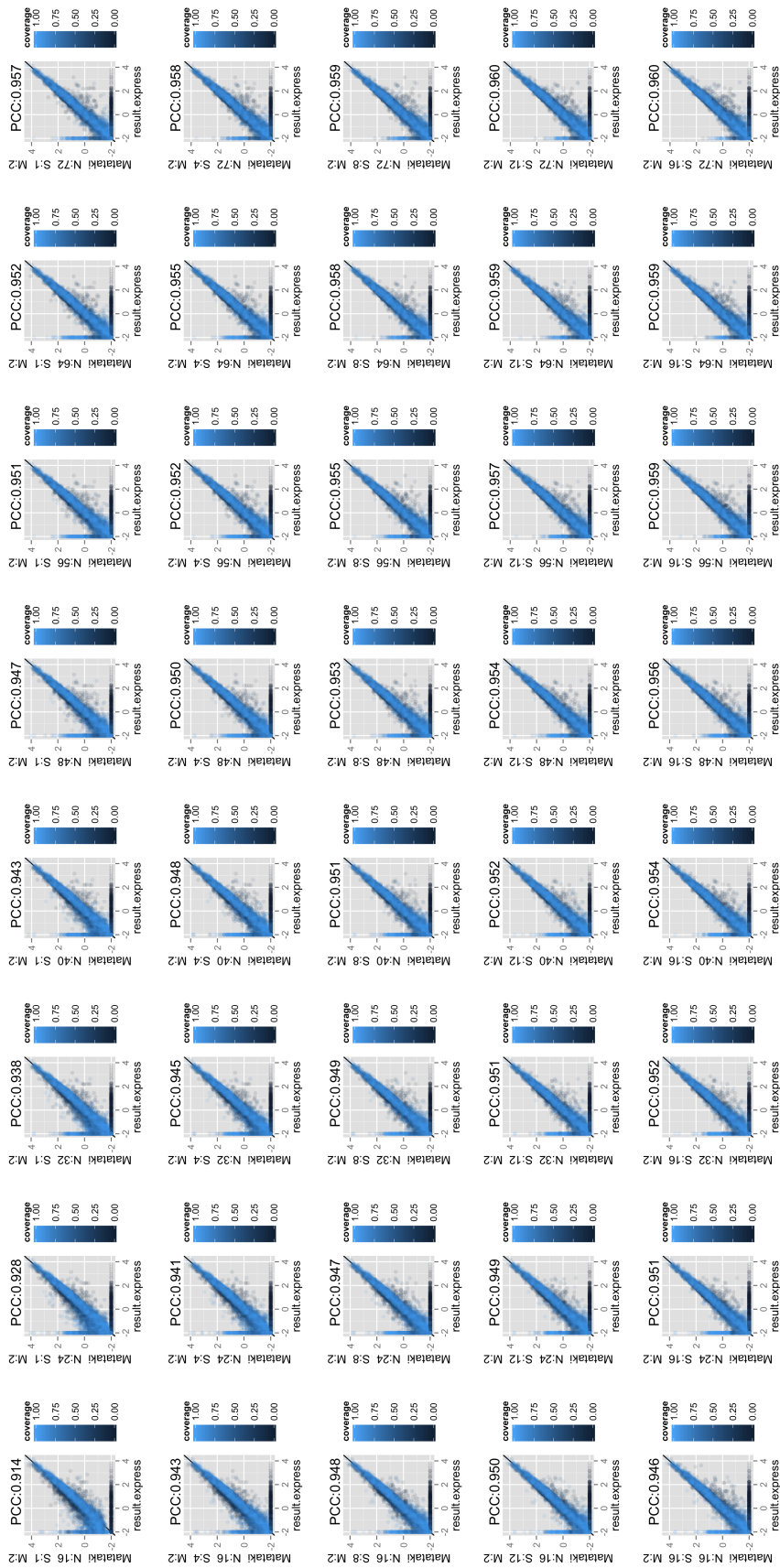
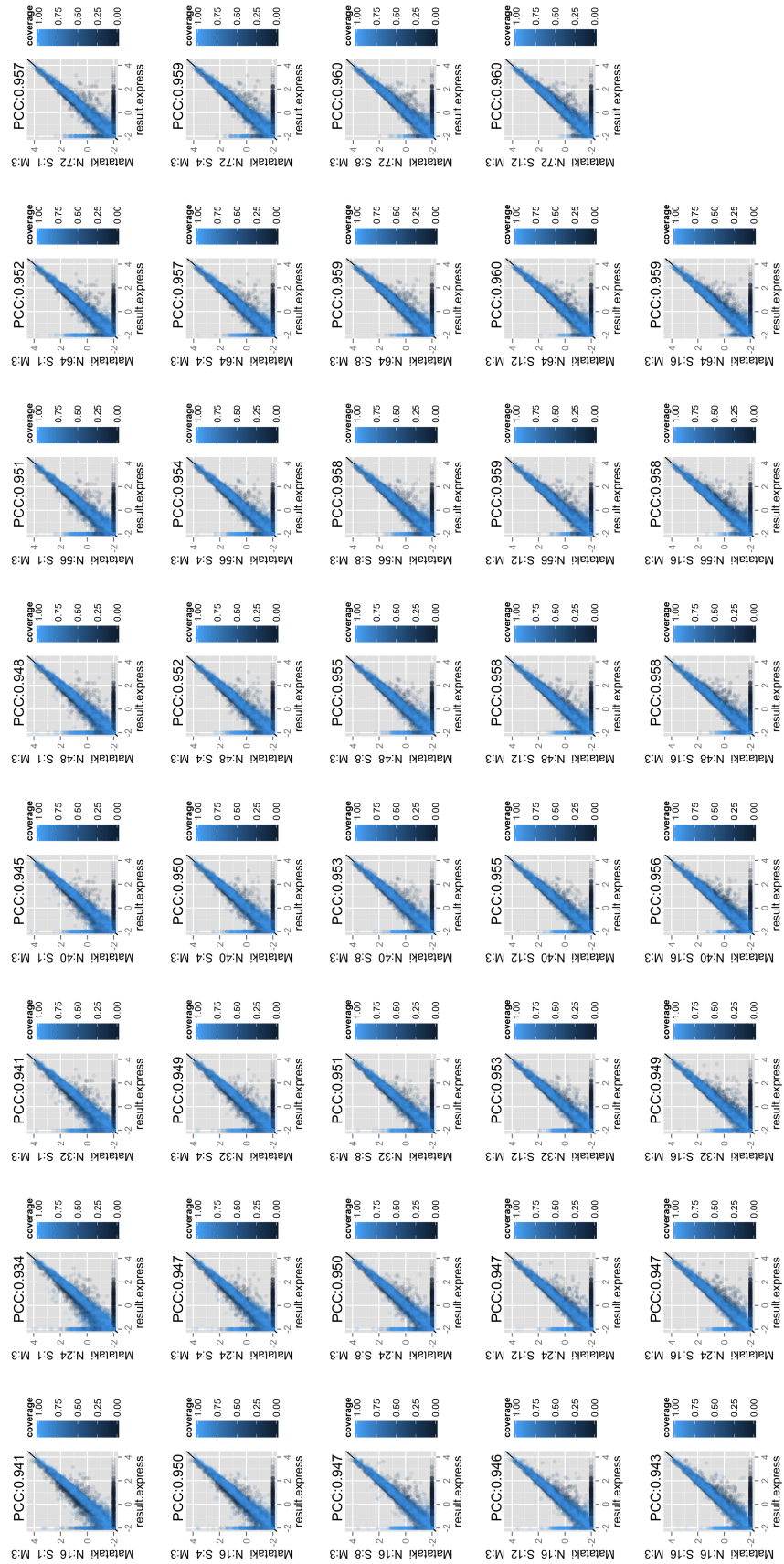


Figure A.2: SRR1639212: Comparison with eXpress when $M = 1$

Figure A.3: SRR1639212: Comparison with eXpress when $M = 2$

Figure A.4: SRR1639212: Comparison with eXpress when $M = 3$

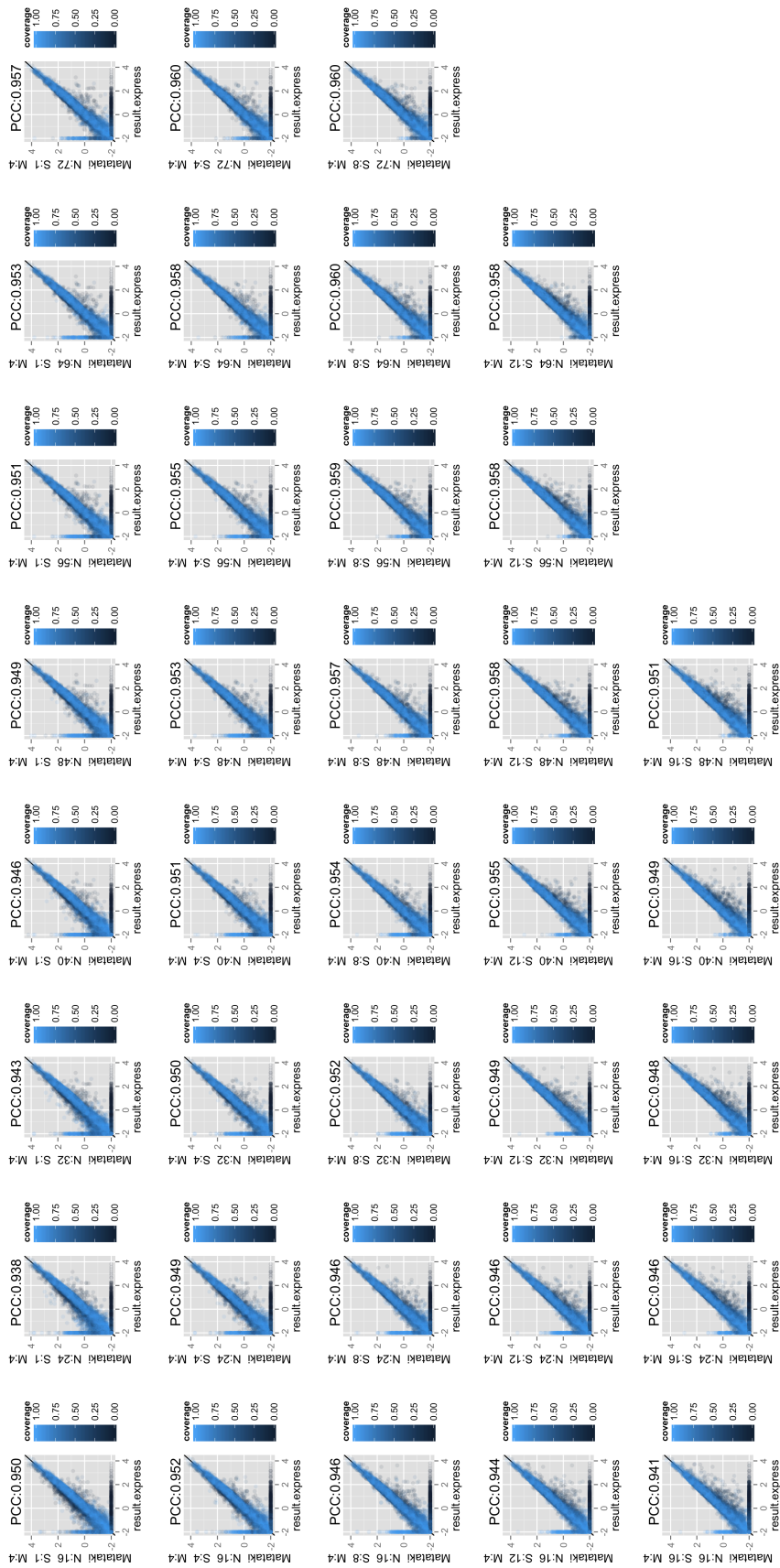


Figure A.5: SRR1639212: Comparison with eXpress when $M = 4$

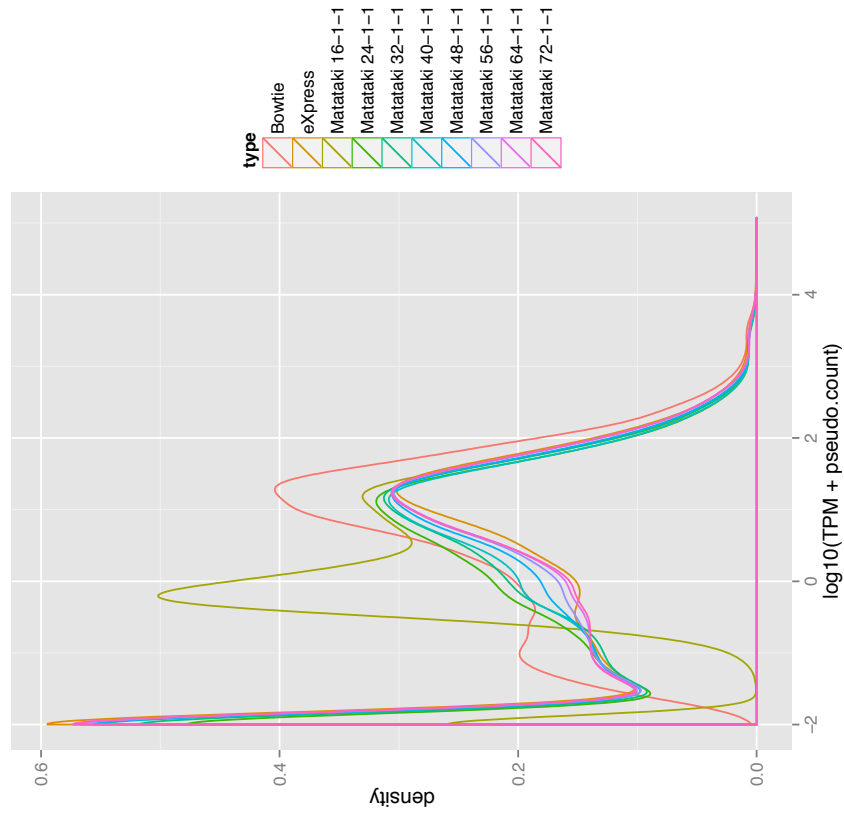
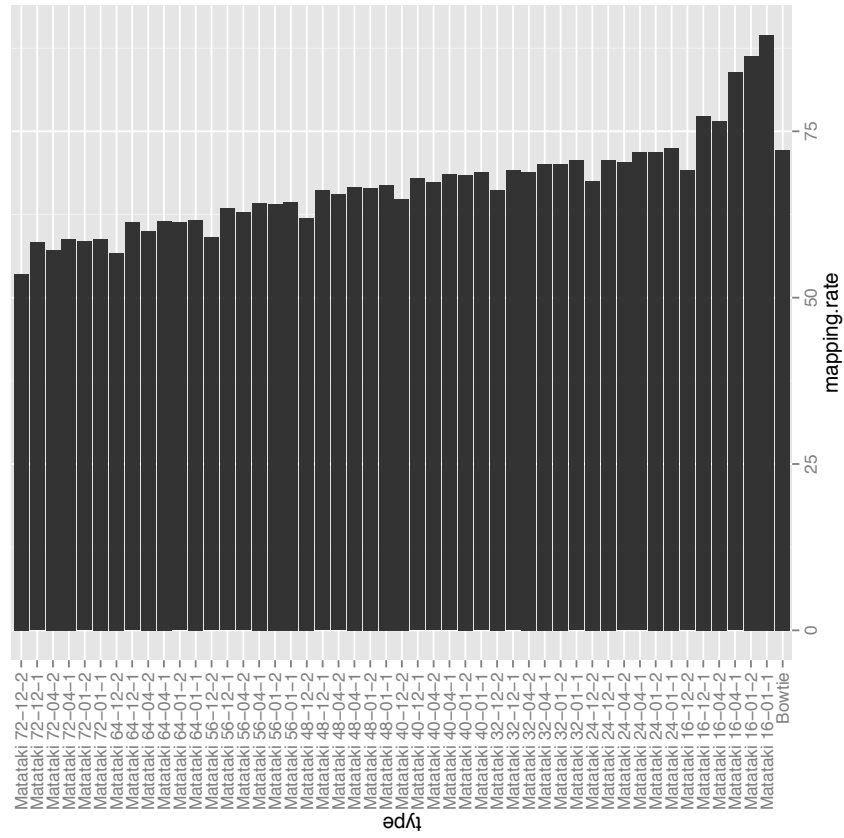


Figure A.6: ERR266335: A distribution of FPKM and mapping rate

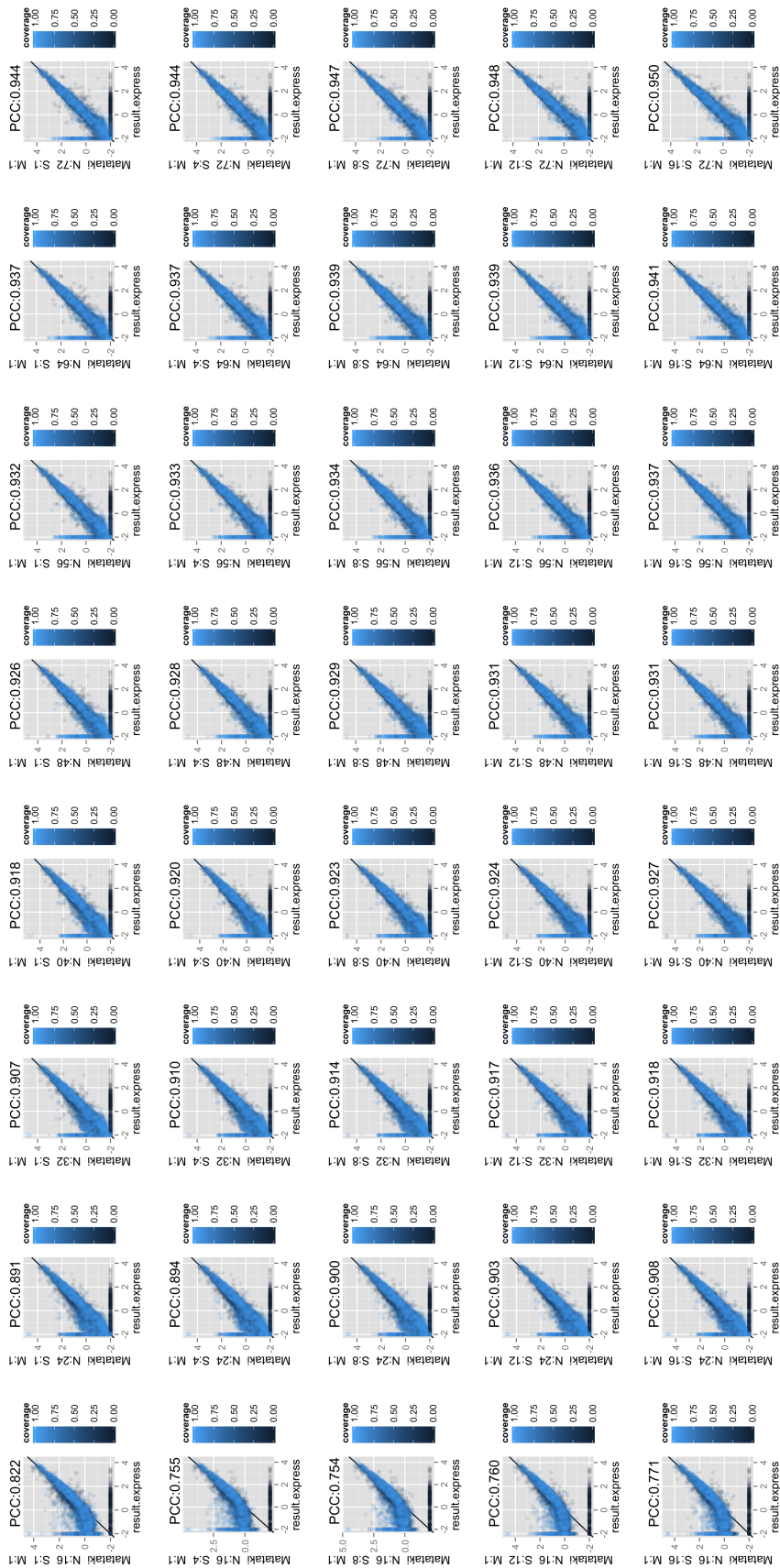
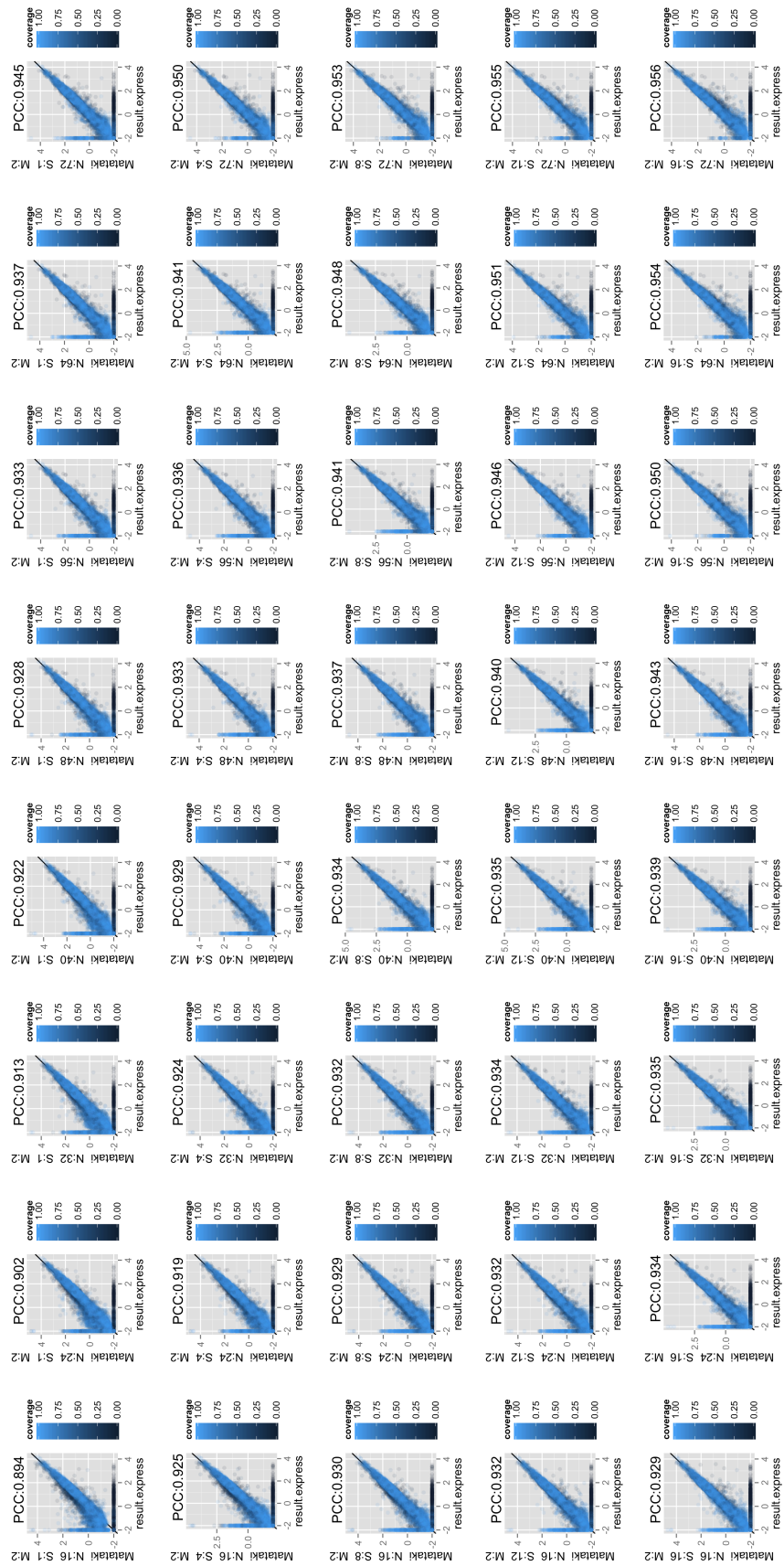


Figure A.7: ERR266335: Comparison with eXpress when $M = 1$

Figure A.8: ERR266335: Comparison with eXpress when $M = 2$

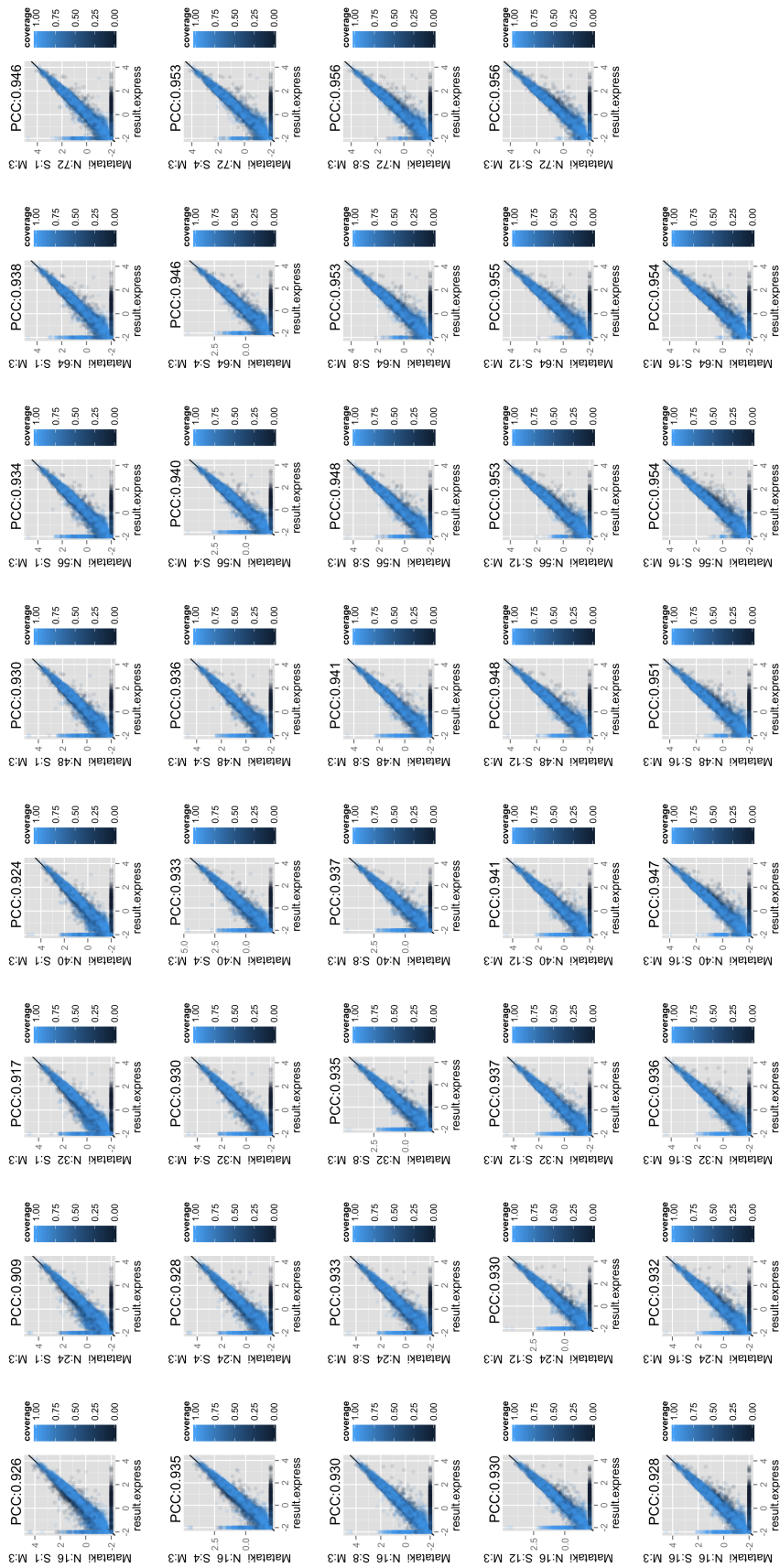


Figure A.9: ERR266335: Comparison with eXpress when $M = 3$

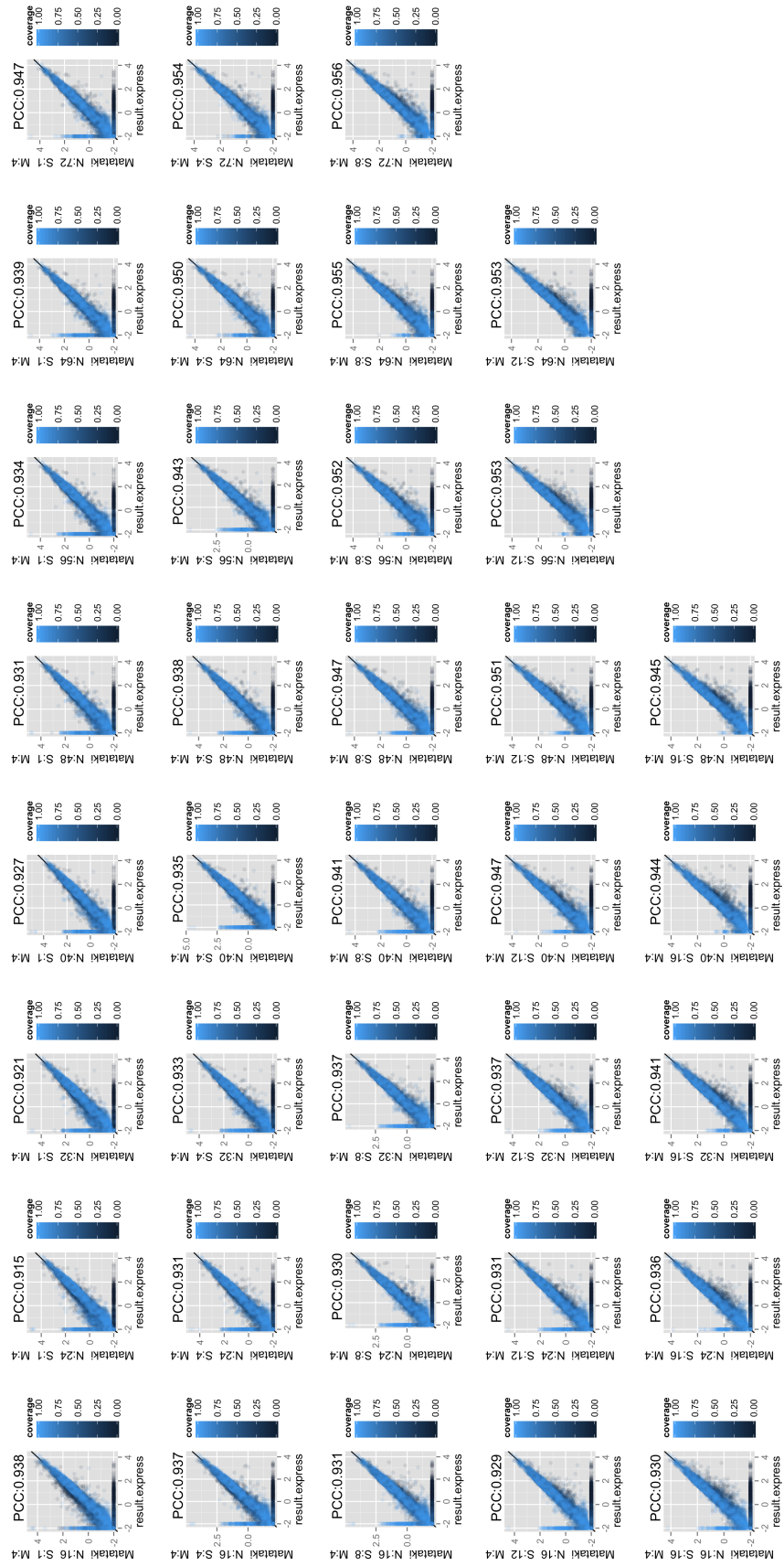


Figure A.10: ERR266335: Comparison with eXpress when $M = 4$

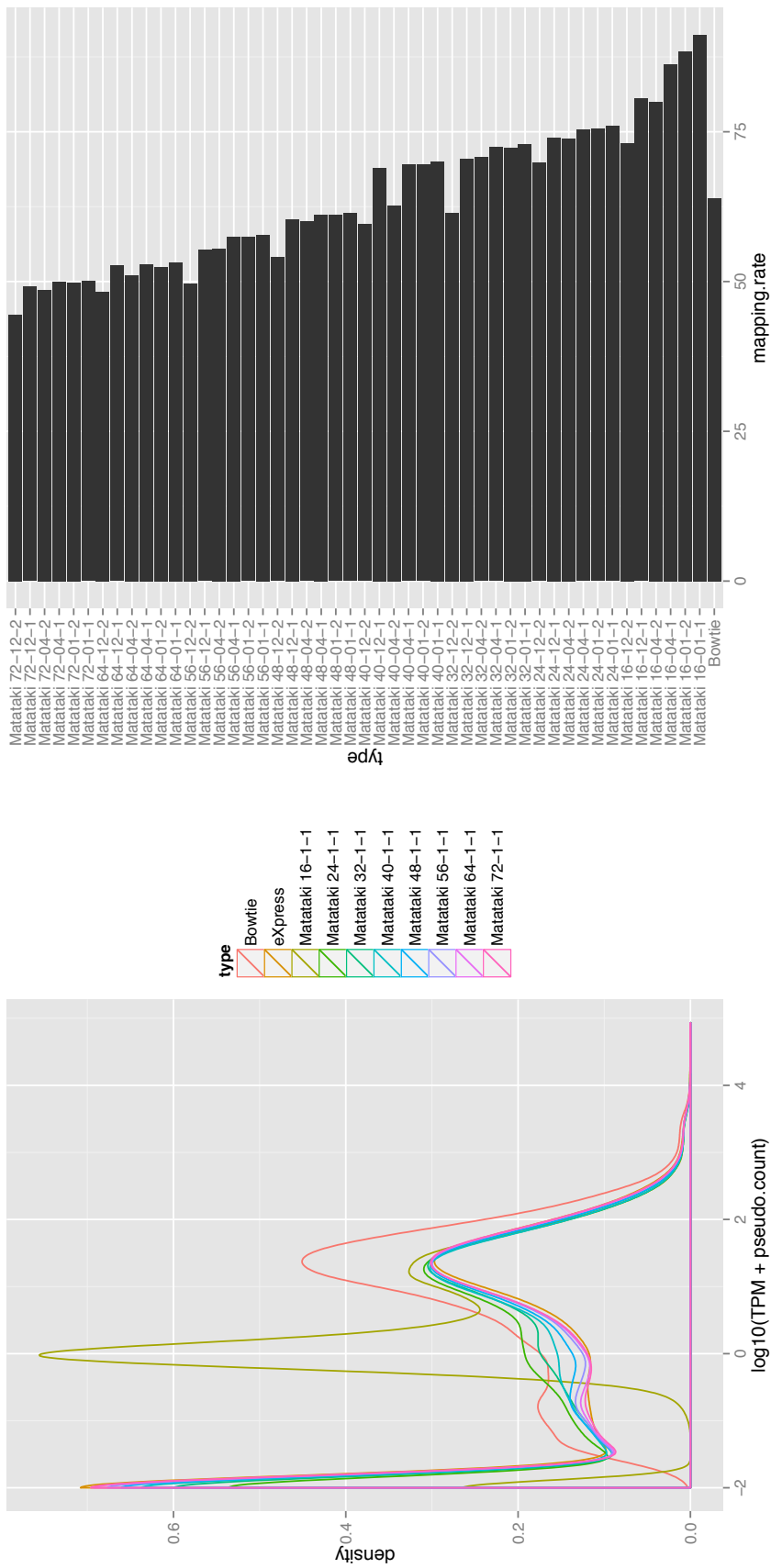


Figure A.11: SRR1013361: A distribution of FPKM and mapping rate

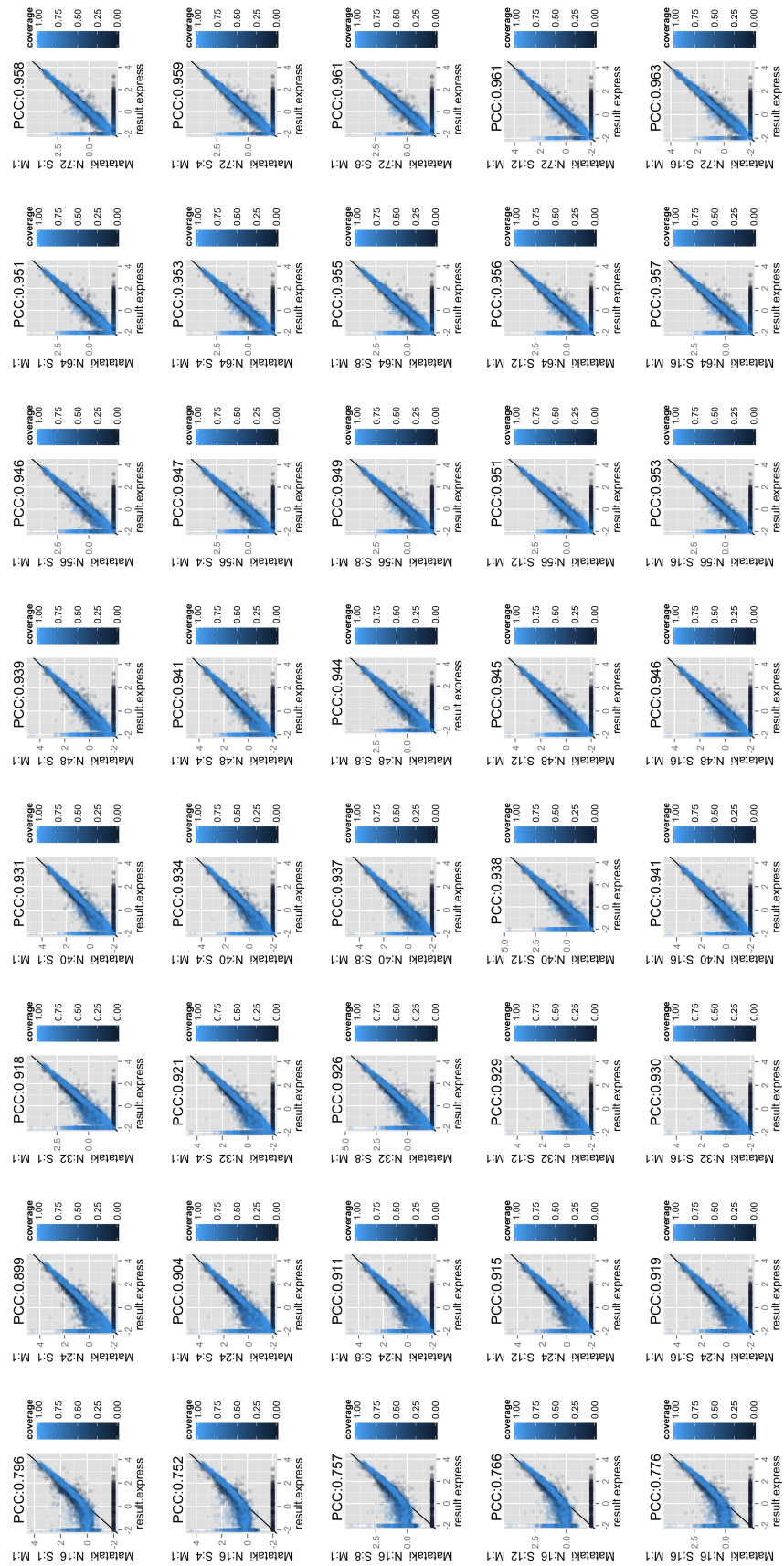


Figure A.12: SRR1013361: Comparison with eXpress when $M = 1$

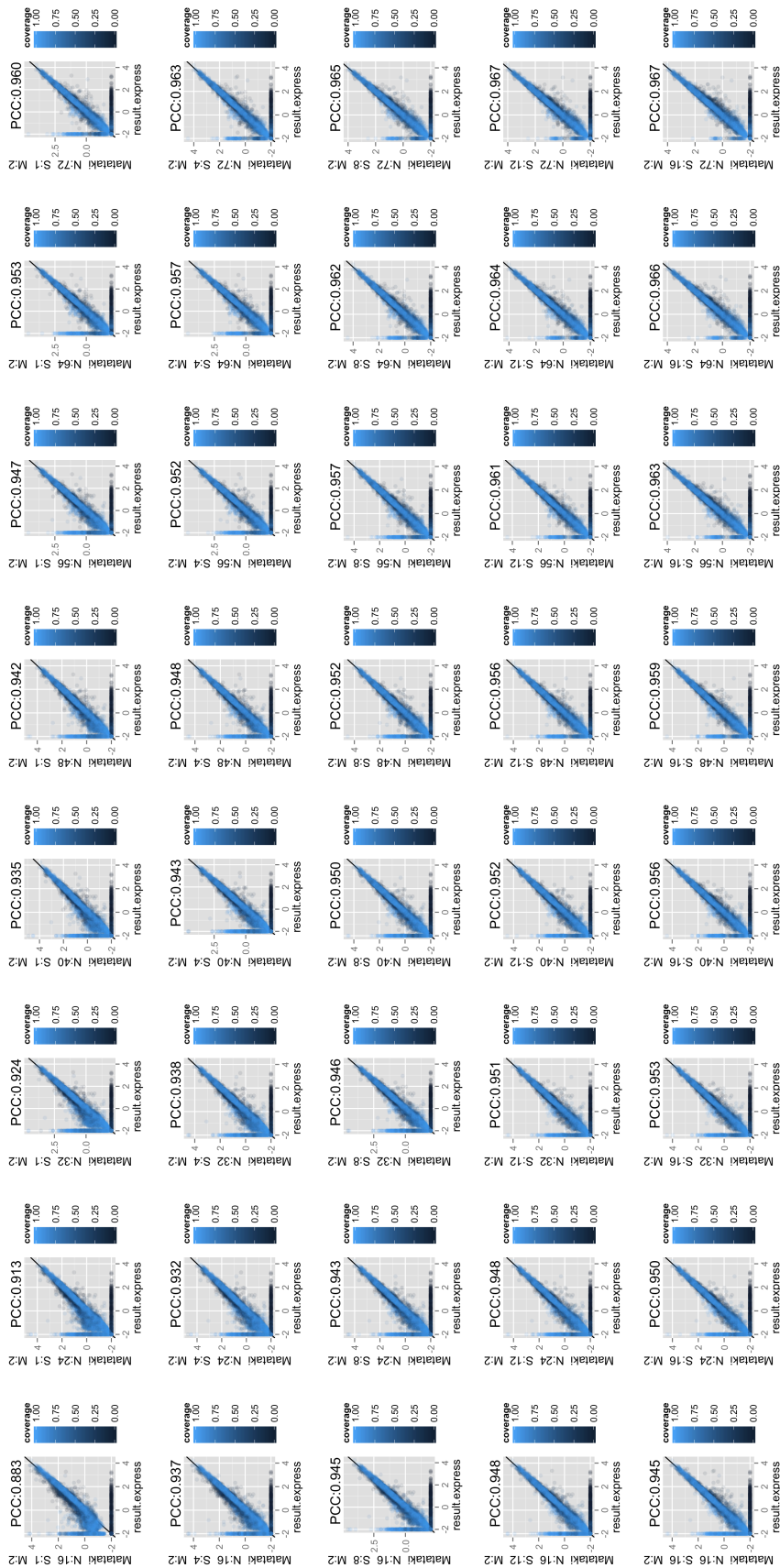


Figure A.13: SRR1013361: Comparison with eXpress when M = 2

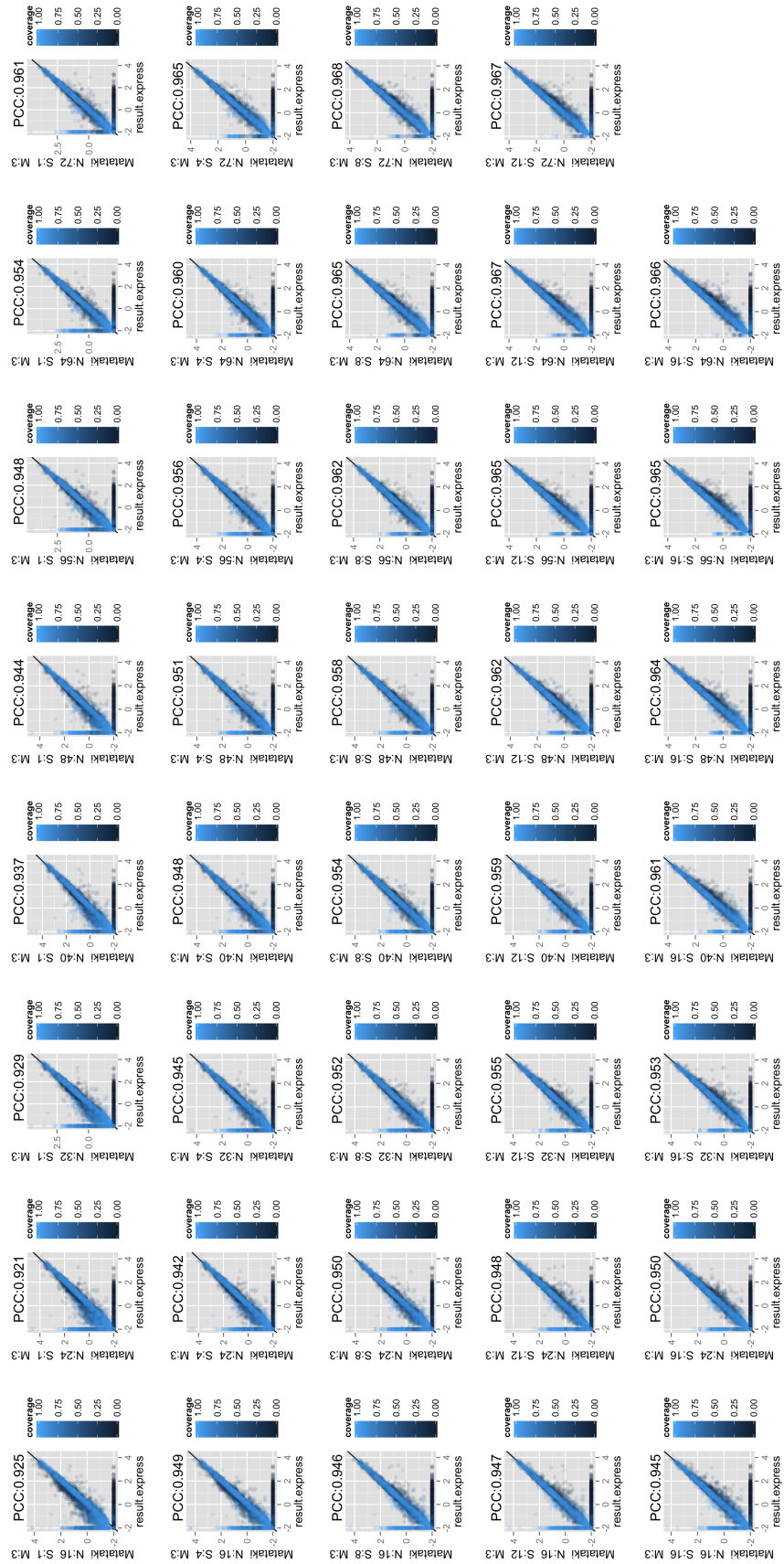


Figure A.14: SRR1013361: Comparison with eXpress when $M = 3$

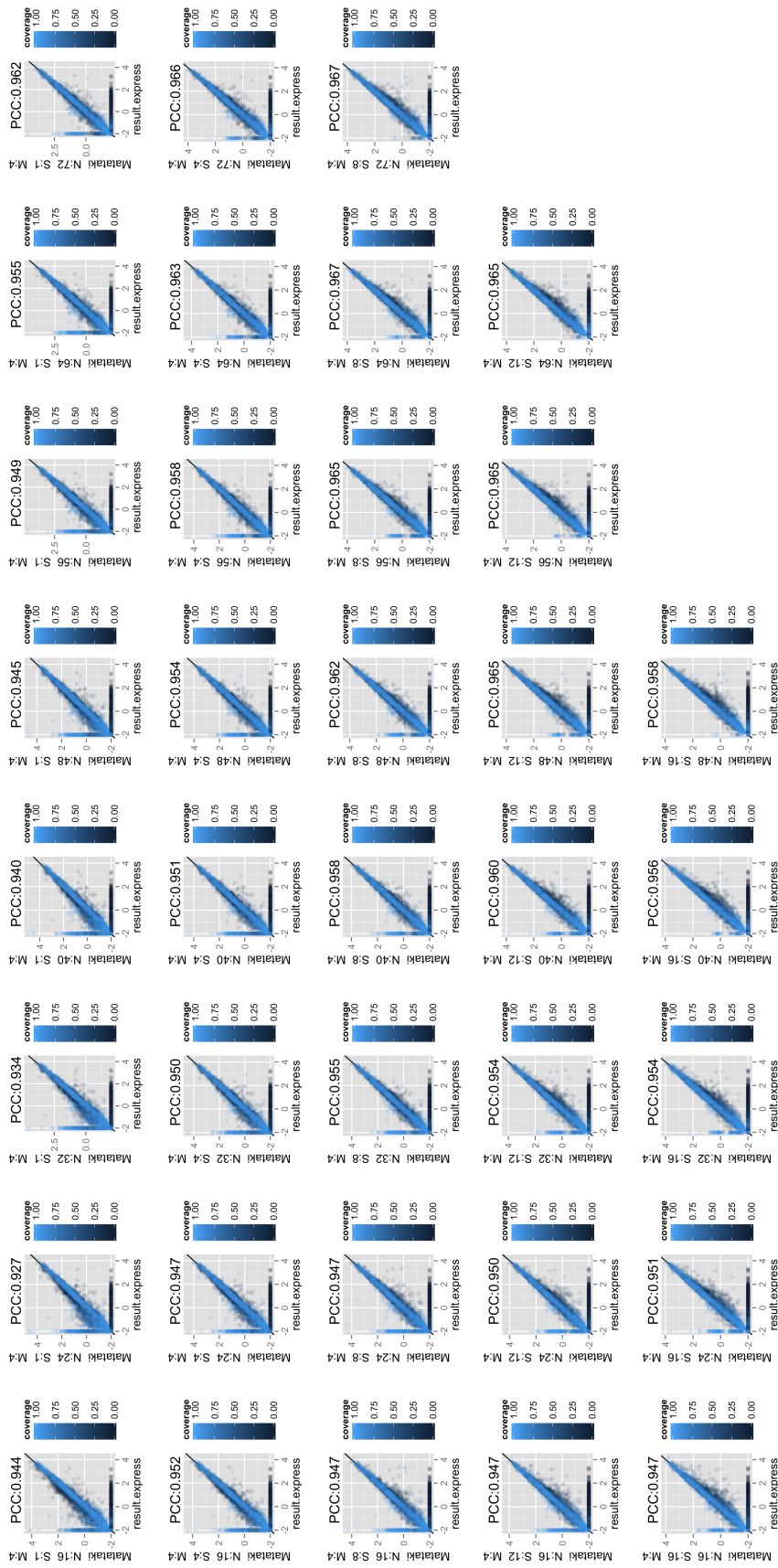


Figure A.15: SRR1013361: Comparison with eXpress when M = 4