FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Information and Data Analysis System for Gene Expression

Diogo André Rocha Teixeira

DISSERTATION PLANNING

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Rui Camacho (PhD)

Second Supervisor: Nuno Fonseca (PhD)

February 3, 2014

Information and Data Analysis System for Gene Expression

Diogo André Rocha Teixeira

Mestrado Integrado em Engenharia Informática e Computação

February 3, 2014

Abstract

Resumo

Contents

CONTENTS

List of Figures

LIST OF FIGURES

List of Tables

LIST OF TABLES

Abbreviations

API	Application Programming Interface
cDNA	Complementary DNA
CSS	Cascading Style Sheets
DBMS	Database Management System
DNA	Deoxyribonucleic Acid
FEUP	Faculty of Engineering of the University of Porto *(Faculdade de Engenharia da Universidade do Porto)*
GUI	Graphical User Interface
HTML	HyperText Markup Language
IBMC	Institute for Molecular and Cell Biology *(Instituto de Biologia Molecular e Celular)*
mRNA	Messenger RNA
NGS	Next Generation Sequencing
RNA	Ribonucleic Acid
RNA-Seq	RNA Sequencing
rRNA	Ribosomal RNA
tRNA	Transfer RNA
WTSS	Whole Transcriptome Shotgun Sequencing

Chapter 1

Introduction

This chapter aims at giving a general overview about the themes address by this thesis. We will address the context in which the thesis is inserted, as well as the motivation that led to its proposal. Furthermore there will be brief description of this thesis main objectives and the methods that will be used to achieve those objectives.

1.1 Context

Molecular biology is a branch of biology that studies biological activities of living beings, at a molecular level. The early grounds for this field of study were set in the early 1930's, although it only emerged in its modern form in the 1960's, with the discovery of the structure of DNA. Among the processes studied by this branch of biology is gene expression. Gene expression (further explained in Chapter 2) is the process by which DNA molecules are transformed into useful genetic products, typically proteins, which are essential for living organisms. This knowledge is not only important in fields like evolutionary or molecular biology, but may have crucial applications in fields such as medicine. One example of such an application is the usage of gene expression analysis in the diagnosis and treatment of cancer patients [PASH03].

With the advent of NGS (Next Generation Sequencing) techniques, researchers have at their disposal huge amounts of sequencing data, that is not only cheaper and faster to produce, but also more commonly available. This data can then be used to obtain relevant information about organisms' gene expression. But, as the cost of sequencing genomes was reduced, the cost of processing such information was increased. NGS techniques tend to produce much smaller reads[1] than previously used techniques, presenting a more complicated problem, from a computational standpoint [Wol13].

[1] A *read* is a single fragment of a genome/transcriptome, obtained through sequencing techniques.

1.2 Motivation and Objectives

Despite its great advancements in the past decades, molecular biology is still a relatively new subject and, as such, there are still some unknowns and partial knowledge in this area. In respect to gene expression, some mechanisms of this intricate process are yet to be fully understood. One such mechanism is the one that regulates the transcription speed of RNA. The objective of this thesis is to understand how the final segments of the genome's exons are responsible for the speed at which the exons themselves are transcribed. This is, however, a complex task, that can be further decomposed in the two main problems that will be address in the thesis, namely:

- Assembly of the study transcriptome, using experimental sequencing reads and a reference genome. This is effectively one of the most complex problems addressed in this thesis. In order to assemble the genome a method called RNA Sequencing[2] will be used. Further insight about this method will be given in Chapter 2, with particular emphasis for RNA Sequencing tools (Section 2.2.1).
- Further analysis of the assembled transcriptomes, using machine learning algorithms applied to data mining. These techniques will be used in an effort to try to understand the already mentioned transcription mechanisms. This topic will be developed in Section 2.3.

Solving these problems requires the use of computational tools. As such, the development of a computer system to address these problems emerges as a secondary objective of the thesis. Some details of this system will be presented in Section 1.3, along with its overall structure, main components and possible technologies to be used.

1.3 Project

The project itself will revolve around the development of a prototype computer system. The first objective of this prototype is to solve the aforementioned thesis problems, namely the transcriptome assembly and analysis. Beyond this objective, the prototype should become an easy to use and useful tool for any researcher investigating this or other similar problems. To fulfill these objectives, we will need to develop a complex system, composed by several smaller systems. Therefore, the envisioned system architecture is divided into three major components, to wit:

Information system is responsible for storing and managing genetic data, coordinating interaction between the other components of the system and providing a web interface for user interaction. This component will be based mainly on typical web technologies, that is, relational databases for data storage (SQL DBMS's), web frameworks for business logic implementation (Ruby on Rails, Padrino, NodeJS[3]) and web markup and styling languages for interface implementation (HTML, CSS).

[2] RNA Sequencing is also referred to as "Whole Transcriptome Shotgun Sequencing", or WTSS.
[3] Ruby on Rails and Padrino are Ruby based web frameworks, while NodeJS is a Javascript based web framework.

Assembly pipeline will use genetic data stored in the information system in order to produce assembled transcriptomes. This pipeline will be composed by several tools, corresponding to each phase of the RNA Sequencing process, possibly intercalated with data format conversion programs. The tools to be used in this component will be further discussed in Section 2.2.1.

Transcriptome analysis will be responsible for the data mining analysis of the assembled transcriptomes, in the context of the problem of the thesis. It is expected that this component integrates with the rest of the system. Further information about the tools that will be used in this component is given in Section 2.3.2.

From here, this document will not dwell in the details of the implementation of such a system, focusing instead the specificities of the problem's solution, from the molecular biology and data mining perspectives. This is due to the fact that the development of the system itself is not the focus of the thesis, but rather a natural consequence of the project's work process.

1.4 Structure of the Report

Besides the introduction chapter, this document is composed three additional chapters. These chapters have the following structure:

Chapter 2 introduces some basic Biology and RNA Sequencing concepts, that are essential to understand the problems with which this document deals. Furthermore, we describe the main techniques used for genome/transcriptomesequencing and assembly, their differences and applications and the tools and data formats typically used on those areas. Lastly, we give some insight about data mining algorithms and how they will be applied to this work.

Chapter 3 outlines the main steps in the development of this thesis (and the respective software prototype) and attempts to provide a feasible schedule for the work's execution. It also presents the datasets that will be studied and used in this work, their origins and features, as well as the validation methods that will be used to ascertain the quality of our results.

Chapter 4 sums up the what has been defined in the report, emphasizing the problem that the thesis addresses and the work that will be executed towards solving that problem. It will also give a brief idea of what are the expected results at the end of the project.

Chapter 2

State-of-the-Art

In this chapter we will begin by making a more in depth presentation of the process of gene expression. This will be followed by a literature and state of the art review in the fields of genome/-transcriptome assembly and data mining. Lastly, we will present some of the tools used in each of those areas, as well as some relevant data representation formats for genetic data.

2.1 Introduction

Before dwelling in the details of the state of the art that are on the foundation of this thesis, it is important to explain some concepts of the domain of molecular biology. As explained in Section 1.1, gene expression is the mechanism by which an organism's DNA can be expressed into functional genetic products, like proteins, rRNA and tRNA. This process starts with the genetic code, or nucleotide sequence, of each gene. Different genes in an organism's DNA are responsible for the creation of different genetic products. The process of gene expression itself is composed by two main stages, transcription and translation [GEN].

Transcription is the stage at which genetic data in the form of DNA is used to synthesize RNA, being this the process that concerns the thesis main question. Several different types of RNA are produced by this process, including mRNA (which specifies the sequences of amino acids that form a protein), rRNA and tRNA, both later used in the translation stage. Simplifying a gene's structure, it can be seen as composed by two types of areas, introns and exons, as seen in Figure 2.1.

Only the exons are useful in the gene expression process, being also known as coding regions. Introns, on the other hand, are not used in the process. They are present in an early stage mRNA molecule, the precursor RNA, but are later removed (or spliced) in the final molecule before the translation stage [GEN]. Figure 2.2 illustrates the removal of introns from the mRNA molecule, during the splicing process. As stated before, the main goal of this thesis is to explain how the final nucleotide sequence of each exon affects the transcription speed of the exon itself.

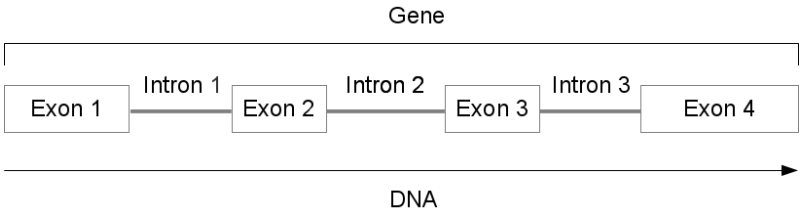

Figure 2.1: Overall structure of a gene, with its different areas (simplified)

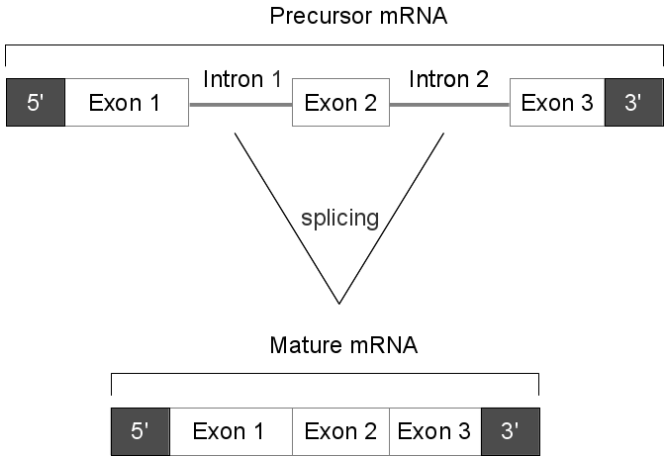

Figure 2.2: The removal (splicing) of introns from the precursor mRNA, during the transcription process

After the conclusion of the transcription process comes the translation process. In this process, the synthesized mRNA is used to specify the sequence of amino acids that constitute the particular protein being produced. The other types of RNA molecules (rRNA and tRNA) are also used in this stage of the gene expression process.

Obtaining this genetic information is done experimentally, by employing a genome sequencing technique. For quite some time this process was carried out using the Sanger sequencing method and other similar methods [RF09]. Though effective, such methods we're notably slow and costly, with large projects like the Human Genome Project (HGP) consuming roughly thirteen years and US$ 3 billion. The past few years have seen the appearance and rise in popularity of the NGS techniques. These techniques differ from the more classical ones by producing larger amounts of information, in less time. They are also typically more cost effective than previous techniques, which has greatly contributed to their popularity. As a disadvantage, NGS techniques produce shorter reads than their older counterparts, posing a complicated problem in terms of genome assembly and raising new computational challenges. Although this thesis will not deal with the problems of sequencing techniques, it is important to indicate that the read dataset that will be used is a result of NGS techniques. As such, we will use assembly techniques more suited to situations where short reads are available.

2.2 Genome Assembly and RNA Sequencing

- talk about genome assembly, talk briefly about micro arrays, more about RNA seq and de novo vs guided assembly/alignment;

2.2.1 RNA Sequencing Tools

2.2.2 Relevant Standard File Formats

2.2.3 FASTA

2.2.4 FASTQ

2.2.5 SAM and BAM

2.2.6 VCF

2.2.7 GTF and GFF

2.3 Data Mining

2.3.1 Data Mining Algorithms

2.3.2 Data Mining Tools

Except in rare cases of very specific problems, it typically makes no sense for someone to implement any data mining algorithm that they might need. In fact, today we have lots of data mining tools (many of which are free), that already implement many of those algorithms. These tools are usually customizable, making it easy to adapt them to most problems. Below we'll briefly review some of the most popular data mining tools, that apply to the specific needs of this thesis.

2.3.2.1 RapidMiner

RapidMiner[1] is a complete solution for data mining problems. It's available as a standalone GUI based application, as seen in Figure 2.3. It is a commercial application, although its core and earlier versions are distributed under an open source license and it offers a free version, beyond its multiple paid versions. Being one of the most popular data mining tools used today, its applications span several domains, including education, training, industrial and personal applications, among others. Its functionality can also be easily extended through the use of plugins[2], reflecting in an increased value for this tool. One such example in the area of bioinformatics is the integration plugin between RapidMiner and the Taverna[3] open source workflow management system [JEF11].

[1] http://www.rapidminer.com/

[2] Plugin is a software module that adds new functionality to an existing software application. Plugins are typically dependent on the platform they extend and can't be used as standalone tools.

[3] http://www.taverna.org.uk/

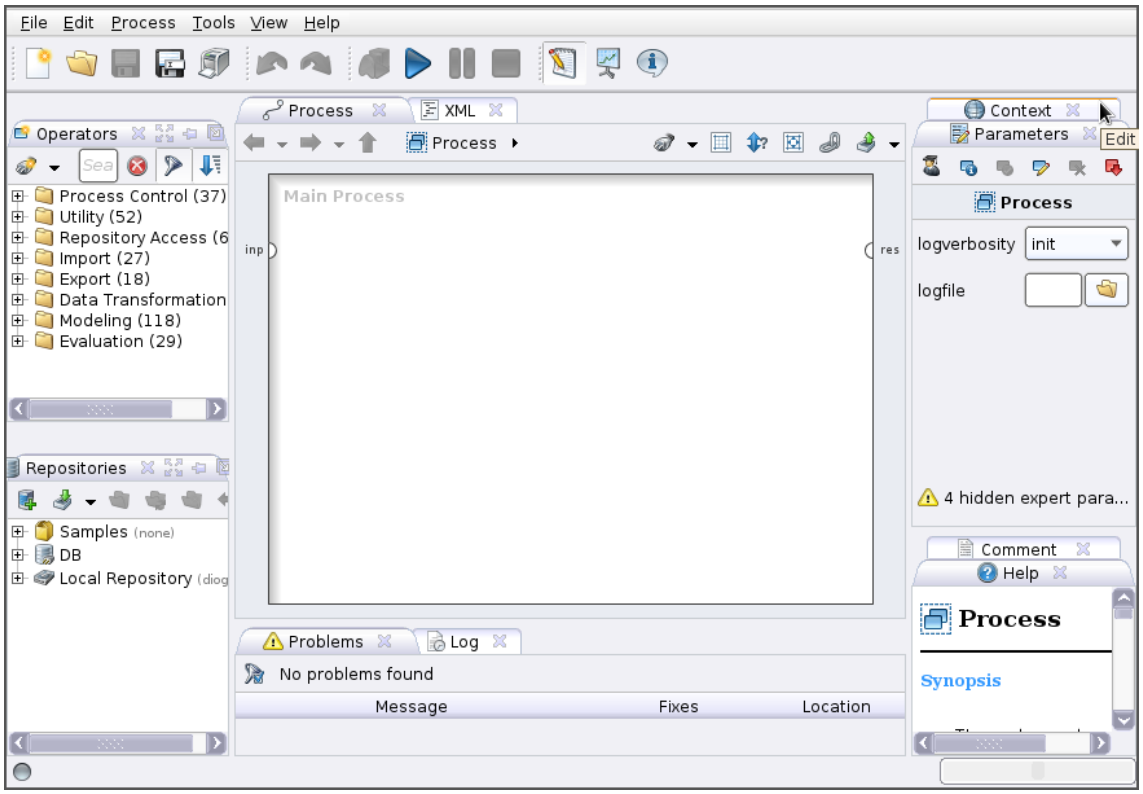

Figure 2.3: RapidMiner user interface

2.3.2.2 Weka

Weka[4] is an open source tool that collects several machine learning algorithms and allows its user to easily apply those algorithms to data mining tasks [HNF+09]. Created at the University of Waikato, New Zeland in 1997 (the current version was completely rewritten in 1997, despite the first iteration of the tool being developed as early as 1993), it's still in active development to date. Weka supports several common data mining tasks, like data preprocessing, classification, clustering, regression and data visualization. It's core libraries are written in Java and allow for an easy integration of its data mining algorithms in pre existing code and applications. Other than that, Weka can be used directly through a command line/terminal or through one of its multiple GUI's (Figure 2.4). Its simple API and well structure architecture allow it to be easily extended by users, should they need new functionalities.

2.3.2.3 R Language

R[5] is a free programming language and software environment for statistical computing and graphics generation. Originally developed by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand in 1993 [Iha98], it's still under active development. R is typically used

[4] http://www.cs.waikato.ac.nz/ml/weka/
[5] http://www.r-project.org/

Figure 2.4: Weka interface selection

by statisticians and data miners, either for direct data analysis or for developing new statistical software [FA05].

R is an implementation of the S programming language[6], borrowing some characteristics from the Scheme programming language. It's core is written in a combination of C, Fortran and R itself. It is possible directly manipulate R objects in languages like C, C++ and Java. R can be used directly through the command line or through several third party graphical user interfaces like Deducer[7]. There are also R wrappers for several scripting languages.

R provides several different statistical and graphical techniques, including linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, clustering, among others. It can also be used to produce publication-quality static graphics. Tools like Sweave [Lei02] allow users to embed R code in LaTeX documents, for complete data analysis.

Bioconductor Package

Bioconductor is a free and open source set of tools for genomic data analysis, in the context of molecular biology [Lei02]. It is primarily based on R. It is under active development, with two stable releases each year. Counting with more than seven hundred different packages, it's the most comprehensive set of genomic data analysis tools available for the R programming language. It also provides a set of tools to read and manipulate several of the most common file formats used in molecular biology oriented applications, including FASTA, FASTQ, BAM and GFF.

2.4 Chapter Conclusions

[6] S is an object oriented statistical programming language, appearing in 1976 at Bell Laboratories.
[7] http://www.deducer.org/pmwiki/index.php

Chapter 3

Work Plan

This chapter describes the general work plan for the thesis, in terms of activities and their respective timings. Furthermore, we will discuss the datasets that will be used in the work, their characteristics and provenience. Lastly, we will address the subject of work evaluation and validation, explaining how it will be conducted, both during and at the end of the project.

3.1 Planning

Aside the preparation phase (already completed), the time available for the thesis will span from February to July, 2014, roughly totalling twenty weeks. It is essential to define a top level schedule beforehand, to ensure that sufficient time will be allotted for every phase of the project and that the timings of those phases are feasible. As such, Figure 3.1 represents the division of the six main phases of the project, during the available period of twenty weeks. Although each phase comprises several smaller tasks, we believe that such a small granularity planning is not needed in this phase and will be defined during the work's execution, as needed. Each main phase of the project is composed as follows:

Information system development comprises the design and development of the data management component of the project and will take roughly six weeks. Despite not being the most critical component, making it the first in the development timeline facilitates later data intensive phases like the transcriptome assembly and, at the same time, allows extensive testing and performance evaluation through usage. A substantial time allotted for this phase will be spent tackling the performance aspects of implementing a system for such large quantities of data, both in terms of database size and response times.

Assembly pipeline development consists of the construction of the tool pipeline responsible for assembling the transcriptomes. At first, several of the already mentioned tools will be studied and tested against small datasets, in an effort to ascertain which are best suited to our

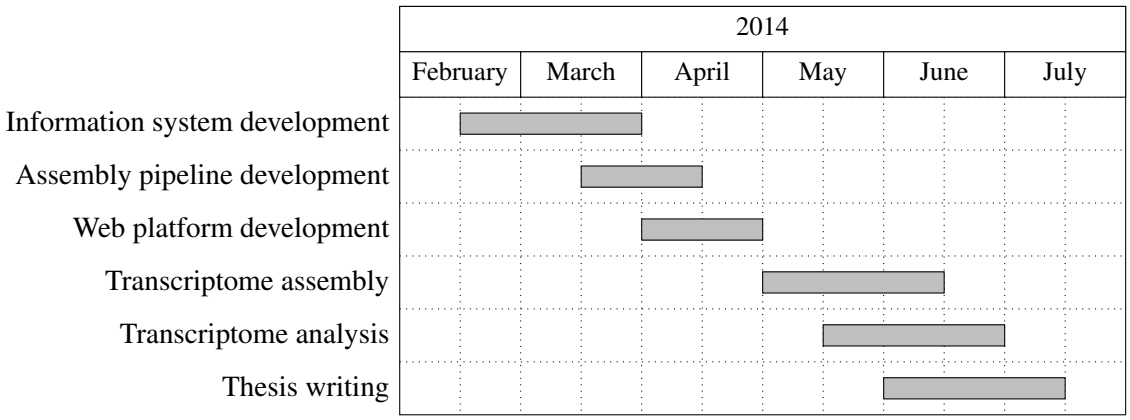

Figure 3.1: Work distribution planning

particular problem. As the tools are selected, the pipeline itself will take shape, integrating the tools in sequence. Any actual development effort in this phase will be in the form of simple data format conversion scripts, since we will use existing assembly tools. Because of this, the estimated duration of this phase is only one month or four weeks, despite its critical importance to the project.

Web platform development will take four weeks and comprises the design and implementation of the system's web front-end. The web platform will integrate the information and assembly systems, providing a user friendly interface for genetic data storage, management and assembly. From a technical standpoint it's a fairly trivial system, which explains why only four weeks were reserved to this phase.

Transcriptome assembly is the first phase after concluding the development of the main components of the system. In this phase the developed system will be used to produce the assembled transcriptome, employing the given production dataset. This phase will take about six weeks, despite no implementation work taking place (saving some small system tweaks). This is because genome and, in this case, transcriptome assembly are resource and time intensive processes that can take several days, making the extra time necessary for both new and repeat experiments.

Transcriptome analysis will consist in the usage of several data mining tools in order to try to explain the already mentioned RNA transcription mechanisms. This phase is expected to last about six weeks. Although not as resource demanding as the transcriptome assembly phase, this will require choosing and testing a new set of tools and possibly integrate them with the developed system.

Thesis writing is the last phase of the project, with an expected six weeks allocated time. These last six weeks refer to a period to collect and report the obtained results and to make the final reviews to the produced content. However, it is expected that the thesis report will

be worked on continuously from the start of the project, in parallel with the other project phases.

3.2 Experimental Data

During this project there will be essentially three types of datasets used: read data, genome data and test data. Each type of dataset has its own nature, origin and purpose. We will use real genetic data from a fly species called *Drosophila melanogaster*, commonly known as fruit fly, which can be seen in Figure 3.2. It is one of the most frequently used organisms to provide its genetic data for these kind of studies and work.

Figure 3.2: Specimen of *Drosophila melanogaster*, viewed from above[1]

The read data will be made available during the project through IBMC. As stated, this data consists of several short sequencing reads of the *Drosophila melanogaster* genome. It is this dataset that will be ultimately used for assembly and posterior data mining analysis. It should be noted that this will be real data, experimentally obtained in a laboratory for this project.

The genome data will consist of already assembled *Drosophila melanogaster* genome(s), that will be used as a reference in our own assembly process. This data will be obtained through FlyBase[2]. FlyBase is an online and publicly accessible database of *Drosophila* genes and genomes. This database allows its data to be downloaded in several formats, that can be either directly used in our assembly pipeline, or be automatically converted by one of the created conversion tools.

Lastly, we will use some small scale datasets for the test and calibration of the assembly pipeline. Such datasets are usually shipped with the assembly tools themselves. If needed, a combination of the two previous datasets can be used to produce small scale test data for this purpose.

[1] Image taken from http://pt.wikipedia.org/wiki/Drosophila_melanogaster.
[2] www.flybase.org

3.3 Thesis Work Evaluation

In the second part of the project, that is the transcriptome assembly and analysis phases, results evaluation and validation is essential. Even more so when such results are typically evaluated from a molecular biology standpoint and therefore are out of the scope of knowledge of the thesis itself. In such cases we will have two evaluation methods at our disposal.

The first method is based on relevant metrics for the problems at hand, from both the transcriptome assembly and data mining parts. Such metrics are usually produced by the tools themselves. As for these metrics there is usually a well defined range of expected results, which makes them a very important method of early result evaluation, in the sense that they can be interpreted without a profound knowledge about molecular biology.

The second method available is the evaluation by IMBC's technicians, that will assist us whenever expert biology knowledge is required. This will ultimately be the method that will provide a real measure of the success of the project.

Furthermore, IMBC's technicians will be essential during the entirety of the project. They will help steer the project into its intended direction, giving some insight about their expectations towards the system. Project phases like the implementation of the information system or the transcriptome analysis will be driven by their feedback, giving us a sense about what should be done. Lastly, they will also be present throughout the project to help with any biology related questions that arise.

Chapter 4

Conclusions

Conclusions

References

[FA05] John Fox and Robert Andersen. Using the R statistical computing environment to teach social statistics courses. *Department of Sociology, McMaster ...*, (January), 2005.

[GEN] GENIE. Gene expression and regulation. Available at http://www2.le.ac.uk/departments/genetics/vgec/schoolscolleges/topics/geneexpression-regulation, last access on January 2014.

[HNF+09] Mark Hall, Hazeltine National, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The WEKA Data Mining Software : An Update. *SIGKDD Explorations*, 11(1):10–18, 2009.

[Iha98] Ross Ihaka. R: Past and future history. *COMPUTING SCIENCE AND STATISTICS*, 1998.

[JEF11] Simon Jupp, James Eales, and Simon Fischer. Combining RapidMiner operators with bioinformatics services–a powerful combination. *Rapid-Miner ...*, 2011.

[Lei02] Friedrich Leisch. Sweave: Dynamic generation of statistical reports using literate data analysis. In Wolfgang Härdle and Bernd Rönz, editors, *Compstat 2002 — Proceedings in Computational Statistics*, pages 575–580. Physica Verlag, Heidelberg, 2002. ISBN 3-7908-1517-9.

[PASH03] Lajos Pusztai, Mark Ayers, James Stec, and Gabriel N Hortobágyi. Clinical Application of cDNA Microarrays in Oncology. *The Oncologist*, 8(3):252–258, January 2003.

[RF09] Jorge S Reis-Filho. Next-generation sequencing. *Breast cancer research : BCR*, 11 Suppl 3:S12, January 2009.

[Wol13] Jochen B W Wolf. Principles of transcriptome analysis and gene expression quantification: an RNA-seq tutorial. *Molecular ecology resources*, 13(4):559–72, July 2013.