Computational Methods for High-Throughput Genomics and Transcriptomics

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Dipl.-Inform. (Bioinf.) Regina Bohnert
aus Stuttgart

Tübingen
2011

Tag der mündlichen Qualifikation: 07.12.2011

Dekan:	Prof. Dr. Wolfgang Rosenstiel
1. Berichterstatter:	Dr. Gunnar Rätsch
2. Berichterstatter:	Prof. Dr. Daniel H. Huson

Für Anneliese

Abstract

The completion of genome sequences for many species, including humans and a number of model organisms, was considered a major milestone at the turn of the millennium. It has been quickly realised that focusing on a single reference genome per species is insufficient to understand the diversity within and between organisms. However, for each species, a multitude of genome sequences are required to give insight into causal sequences for variable traits. The advent of high-throughput technologies such as next-generation sequencing has undoubtedly accelerated sequencing and allowed for many exciting large-scale studies in genetics that were not previously conceivable.

Genome-wide association studies, in which the linkage of sequence and phenotype variations is investigated, have highly profited from the recent technology development. For these kinds of studies, it is indispensable to measure genotypes and relevant traits for a large set of individuals. Because such data is of immense quantity and often noisy, computational approaches are required to analyse data from next-generation genetics. In the context of my thesis, I have contributed to the analysis of biological high-throughput data in two respects. In order to accurately describe genotypes, I have designed efficient large-scale tools to identify and catalogue polymorphisms from array data. Moreover, I have developed approaches for the estimation of transcript abundances from next-generation sequencing data, enabling precise analyses of transcriptomes.

The first part of my thesis focuses on the analysis of array-based resequencing data that was obtained to describe sequence variation across 20 diverse varieties of domesticated rice. I applied sophisticated machine learning methods for efficient and accurate analysis of this enormous set of hybridisation data. Using an approach based on Support Vector Machines, I uncovered more than 300,000 non-redundant single-nucleotide polymorphisms, which were found to be highly accurate assessed on a gold standard set of polymorphisms. For the detection of complex regions of polymorphisms, I employed a second machine learning method based on Hidden Markov Support Vector Machines, revealing between 65,000 and 203,000 polymorphic regions across varieties and complementing the SNP set derived with the SVM-based approach. Altogether, detecting hundreds of thousands of polymorphisms on a genome-wide scale has enabled the assembly of the first whole genome set of polymorphisms for the world's most important crop plant.

In the second part of my dissertation, I address the question of accurate quantification of transcriptomes from RNA sequencing measurements. For this purpose, I developed a novel computational method that uses techniques from machine learning and optimisation. In particular, this tool, rQuant, infers the abundance of alternative transcripts and simultaneously estimates the effect of biases induced by experimental settings. Quantifying transcripts from artificial as well as experimental data sets demonstrated the superiority of rQuant in an evaluation for diverse settings and a comparison against other transcript quantification tools. Moreover, I adapted ideas of rQuant to develop a tool for quantitative deconvolution of RNA secondary structures. rQuant is available to the community as open-source software and as a web service.

In conclusion, my thesis contributes to key parts of research in high-throughput genomics and transcriptomics. This work will facilitate the identification of genotype and phenotype linkage and will improve our understanding of the biological processes that make individuals unique.

Zusammenfassung

Die Sequenzierung von Genomen vieler Arten, darunter die des Menschen und einiger Modellorganismen, war ein wichtiger Meilenstein der Jahrtausendwende. Es wurde schnell klar, dass es nicht ausreicht, nur ein einzelnes Referenzgenom pro Art zu betrachten, um die Vielfalt innerhalb und zwischen Organismen zu verstehen. Viele Genomsequenzen pro Art sind notwendig, um zu verstehen, welche Sequenzen ursächlich für variable Merkmale sind. Die Einführung von Hochdurchsatzverfahren, wie zum Beispiel von Sequenziermethoden der nächsten Generation, haben zweifelsohne das Sequenzieren beschleunigt und ermöglichen viele interessante Genetikstudien im großen Umfang, die zuvor undenkbar waren.

Genomweite Assoziationsstudien, in denen die Verbindung von Sequenz- und Phänotypvarianten untersucht werden, haben im großen Maße von der jüngsten Technologieentwicklung profitiert. Für diese Art von Studien ist es unabdingbar, Genotypen und relevante Eigenschaften für eine große Zahl an Individuen zu messen. Da diese Daten von immenser Größe und oft verrauscht sind, sind computergestützte Verfahren für die Datenanalyse in der Genetik notwendig. Im Rahmen meiner Doktorarbeit trug ich zur Analyse von biologischen Hochdurchsatzdaten in zweierlei Hinsicht bei. Um Genotypen genau zu beschrieben, entwarf ich effiziente Programme zur Erkennung und Katalogisierung von Polymorphismen basierend auf Arraydaten. Außerdem entwickelte ich Methoden, um Transkriptmengen aus Messungen neuartiger Sequenziertechnologien zu schätzen, die präzise Analysen von Transkriptomen ermöglichen.

Der erste Teil meiner Doktorarbeit beschäftigt sich mit der Untersuchung von arraybasierten Resequenzierdaten, die generiert wurden, um Sequenzvariation innerhalb 20 verschiedener domestizierter Reissorten zu beschreiben. Ich verwendete ausgefeilte Methoden des maschinellen Lernens, um die große Menge an Hybridisierungsdaten effizient und genau zu analysieren. Basierend auf Support-Vector-Maschinen entdeckte ich mehr als 300.000 nicht-redundante Einzelnukleotidpolymorphismen, die sich, evaluiert an Hand eines Goldstandard für Polymorphismen, als sehr genau erwiesen. Um komplexe Polymorphismenregionen zu erkennen, wandte ich eine weitere Methode des maschinellen Lernens basierend auf Hidden-Markov-Support-Vector-Maschinen an, die zwischen 65.000 und 203.000 polymorphe Regionen innerhalb der Reissorten identifizierte und den SNP-Datensatz der SVM-Methode komplemenierte. Beide Ansätze zusammengenommen detektierten genomweit Hunderttausende von Polymorphismen, wodurch der erste Polymorphismendatensatz für das vollständige Genom der weltweit wichtigsten Nutzpflanze erstellt werden konnte.

Im zweiten Teil meiner Dissertation widme ich mich der Fragestellung, Transkriptome, welche mit Hilfe von RNA-Sequenzierung gemessen werden, genau zu quantifizieren. Dazu entwickelte ich eine neuartige computerbasierte Methode, die Techniken aus dem maschinellen Lernen und der Optimierung verwendet. Dieses Programm, rQuant, inferiert Mengen von alternativen Transkripten und schätzt gleichzeitig den Einfluß von Verzerrungen, die durch experimentelle Protokollgegebenheiten herbeigeführt werden. Die Quantifizierung von Transkripten an Hand künstlicher sowie experimenteller Datensätze zeigte die Vorzüge von rQuant in einer Auswertung unterschiedlicher Programmeinstellungen und in einem Vergleich mit anderen Transkriptquantifizierungsprogrammen. Darüber hinaus verwendete ich Ideen von rQuant, um ein Programm zu entwickeln, das RNA-Sekundärstrukturen quantifiziert. rQuant ist sowohl als Open-Source-Software als auch Webservice verfügbar.

Zusammenfassend lässt sich sagen, dass meine Doktorarbeit zu wichtigen Teilen der Forschung in der Hochdurchsatzgenomik und -transkriptomik beiträgt. Diese Arbeit wird die Indentifizierung von Verbindungen zwischen Geno- und Phänotyp vereinfachen und unser Verständnis von biologischen Prozessen, die Individuen einzigartig machen, verbessern.

Acknowledgements

Above all, I would like to thank my main adviser Gunnar Rätsch for his support and guidance since the early days of his research group and for invaluable ideas and discussions that contributed to my thesis. I am deeply grateful to him for giving me the opportunity to undertake and experience scientific research not only in an excellent environment at the Max Planck Campus in Tübingen, but also at conferences.

Moreover, I am very thankful to the members of my Ph.D. advisory committee, Daniel Huson, Detlef Weigel and Karsten Borgwardt, who critically discussed my ongoing projects and gave me ideas for new directions. I would like to thank Daniel Huson in particular for his long-term interest in my work and his support since my undergraduate studies.

Furthermore, I would like to acknowledge my colleagues and collaborators for their contributions to joint projects and for providing data and code: particularly the RGASP team Jonas Behr, Georg Zeller, André Kahles and Gunnar Rätsch for sharing intensive weeks of team work; additionally Lisa Smith, Lisa Hartmann, Philipp Drewe, Oliver Stegle, Karsten Borgwardt, Richard Clark, Kevin Childs, Ali Mortazavi, Vipin Sreedharan, Christian Widmer and Nico Görnitz for valuable collaborations and discussions.

Many thanks go to former and current members of Gunnar's group for scientific and social interactions and countless inspiring discussions during lunches, bio-breakfasts, coffee talks, retreats and conferences.

Also, I would like to thank the people from whom I could benefit and learn a lot in many other ways during the last years: the small but constant group of people of our reading group on convex optimisation; my fellow Ph.D. students in the Ph.D. student council for many hours of joint efforts and fun; Elisabeth Georgii, Gabriele Schweikert, Juliane Klein and Géraldine Jean who made my stays at conferences even more enjoyable by sharing accommodation; Philipp Berens for his reliable and long-standing friendship.

In addition, I am very thankful to Gunnar Rätsch, Christian Widmer, Géraldine Jean, Lisa Smith and Sebastian Schultheiß who gave me constructive feedback to the manuscript of this thesis. I would also like to thank Stephanie Heinrich and Sebastian Schultheiß for their help in printing and submitting this dissertation.

Finally, I would like to express my deep gratitude to Marius and my family for their continuous support, understanding and patience.

This work was funded by the Max Planck Society.

Contents

1. Introduction

The completion of genome sequences for many species, including humans and a number of model organisms, was considered a major milestone at the turn of the millennium. It has been quickly realised that a single reference genome per species is insufficient to understand the diversity within and between organisms. However, for each species, a multitude of genome sequences are required to give insight into causal sequences for variable traits. The advent of high-throughput technologies such as next-generation sequencing (NGS) has undoubtedly accelerated sequencing and allowed for many exciting large-scale studies in genetics that were not previously conceivable. The goal of personal genomics to sequence individual genomes at a cost of not more than $ 1000 has moved closer to reality due to the technical advancements in sequencing. Comparatively inexpensive genome sequencing will be essential in medical applications for personalised therapies, e.g., in cancer patients, and for prediction of predisposition to diseases.

To learn why individuals exhibit different phenotypes, one important part is to measure their genotypes, but the second emancipated goal needs to be to comprehensively describe phenotypic variation at the same time. Having laid these foundations, genome-wide association studies can be undertaken to find linkage between sequence and phenotype variations [137]. As gene expression levels can have an impact on phenotype, many studies aim to correlate polymorphisms with expression quantitative trait loci (eQTL) [40, 130, 131, 165]. Similar studies are also undertaken to gain insight into the underlying varied regulation [161], or to identify SNPs associated with splicing patterns as splicing QTL [113].

Genome-wide association studies are only statistically powerful when screening a large set of individuals, requiring analysis of substantial data sets. Due to the immense quantity of data, which is often noisy and diverse, computational analyses are an essential part for next-generation genetics. Computational approaches developed in the context of my thesis contribute to the analysis of biological high-throughput data in two respects. I have developed efficient large-scale tools to identify and catalogue polymorphisms from array data, as well as for the estimation of transcript abundances from next-generation sequencing data. Measuring both genotypes and expression phenotypes in an accurate and efficient way are fundamental requirements to identify the causal gene variants for complex phenotypes and thus facilitate studies of eQTL variations.

1.1. Thesis Outline

In the following section, I introduce general concepts in machine learning and mathematical optimisation (Section 1.2), areas from which the methodologies described and developed in this thesis are derived. Further, I give a brief introduction to genome biology (Section 1.3), and present state-of-the-art sequencing technologies in Section 1.4. The subsequent part of the thesis is divided into two main parts, one part addressing the analysis of array-based resequencing data for polymorphism discovery, and the second part presenting approaches for RNA transcript and structure quantification from next-generation sequencing data.

In Chapter 2, the application of two machine learning methods for SNP discovery and detection of polymorphic regions from array-based resequencing across 20 domesticated rice varieties is described and discussed. The set of sequence variants assembled in this context was the first genome-wide collection for diverse varieties of the world's most important crop plant.

The second part of the thesis starts in Chapter 3 with the description of rQuant, an approach based on mathematical optimisation that I have developed for the accurate quantification of alternative transcripts from RNA-sequencing data. Besides abundance estimation, this approach simultaneously models biases inherent in library preparation and other experimental settings. A generalisation of the core optimisation problem of rQuant led to the development of another algorithm, sQuant, to quantify alternative RNA secondary structures, which is presented in Chapter 4. Finally in Chapter 5, I show the integration of rQuant into the web server framework Galaxy as the tool rQuant.web, and give details about the implementation of rQuant published as an open-source package.

Each chapter of this thesis contains a publication note stating who contributed to the presented work and where the used material has been published. Moreover, a list of publications and oral presentations are given at the end of the thesis, annotated with author contributions and relations to this dissertation.

1.2. Mathematical Background

1.2.1. General Concepts in Machine Learning

Machine learning is a scientific field focusing on the development and research of computer algorithms to learn from experience. These methods automatically detect hidden relations in a set of examples without explicitly modelling them. Here, generalisation of the derived relation when seeing new examples is one of the key goals to be achieved during learning. Usually, the relation is mathematically formulated as a function f that relates the input $\boldsymbol{x}$ to its output $\boldsymbol{y}$. The domain of $\boldsymbol{x}$ and $\boldsymbol{y}$ can be anything ranging from binary values, scalar vectors to structured data such as sequential data and graphs. In *classification*, the outputs $\boldsymbol{y}$ are discrete labels for classes as which the input data is categorised. Fitting a curve to pairs of input and scalar outputs is known as *regression*, probably one of the statistical data analyses that are commonly undertaken across all kind of areas of research. In *structured output learning* the relations of inputs to more complex outputs, often represented as graphs, are determined.

The diversity of the data's structure and of the approached learning tasks is reflected in the wide range of machine learning applications. To name a few, natural language processing, search engines, computer vision, and computational biology are research areas where machine learning methods can be of great benefit and is successfully used in practice. Many biological data sets represent a great challenge, as data measurements may be large in number, noisy, and of high dimensionality. This data may further be incomplete in measuring the underlying biological processes that are often highly complex and poorly understood. Methods from machine learning help in untangle these difficulties to improve and extent the existing knowledge.

All these learning tasks have been addressed by different approaches, which have been largely adopted from statistics. Probabilistic graphical models have frequently been employed and refined in machine learning. Among these, models over a directed graph include Bayesian

networks, which cover popular and well-established models such as hidden Markov models (HMM, e.g. [54, p. 128–138] or [24, p. 610–635], cf. Section 1.2.3) and neural networks (e.g. [54, Chapter 6] or [24, Chapter 5]). Also, models based on undirected graphs such as Markov random fields [24, p. 383–393] and conditional random fields (CRF) [111] have been shown to be suitable models for structured output learning tasks. By contrast, there exists non-probabilistic approaches with their prominent member being support vector machines (SVM, e.g. [43], [205, Chapter 10] and [177]) as an example for large margin methods, which is often applied to solve not-linearly separable classification tasks.

Determining the function f is in general realised by a procedure called training or learning on a set of available data. In *supervised* learning, the algorithm sees both input as well as its corresponding output or label to infer f [54, p. 16–17]. In *unsupervised* and *semi-supervised* training, the labels are unknown or only partly known and f is determined by detecting a hidden structure in the input data, resulting in clustering or dimensionality reduction of the data [54, p. 17]. *Reinforcement learning* is another learning strategy, with ,e.g., application in robot control, that describes a procedure of most on-line sequence of actions in an environment by maximising a reward [54, p. 17].

Depending on the manner how learning is conducted one distinguishes *generative* and *discriminative* learning. In generative approaches a model of the joint probability $p(x, y)$ of the input data x and labels y is derived, providing a model from which new synthetic data can be generated [149]. For classification, where the interest lies in modelling the conditional probability $p(y|x)$, this strategy might seem to be taking a way with detours [205]. In discriminative training in contrast, the posterior $p(y|x)$ is learned directly, which is sufficient for the classification task. Interestingly, a generative model can often be related to a discriminative counterpart [149]. For example for the classification task, the generative naïve Bayes classifier corresponds to logistic regression from a discriminative view [149]. Analogously, HMMs and CRF form the generative-discriminative-pairing for sequential data [111]. Recently, Franc et al. [62] placed the SVM in the view of a maximum likelihood estimate of a specific class of probabilistic models. The choice of one strategy to the other is depend on the task. Thus, choosing an appropriate learning model is mainly conditioned by the purpose, power, and favoured properties of the model that should be learned.

Apart from adopting and applying methodology from statistics and artificial intelligence, techniques from optimisation have become essential for the development of machine learning algorithms. A major part of machine learning problems can be reduced to optimisation problems [21]. The strong intertwining of the two disciplines is noticeable in the application of already existing optimisation strategies to new learning problems on the one hand, on the other hand in improving the scalability of machine learning algorithms by enhancing the specific structure of the embedded optimisation problem.

1.2.2. Optimisation

Having its origin in operations research, mathematical programming or optimisation is nowadays present in many other disciplines. Optimisation techniques have been applied in mechanics and engineering, economics, and also machine learning. In the following I describe optimisation terminology mostly based on [34], which is important to understand concepts and developed methods discussed in the Chapters 2, 3 and 4.

Terminology

The term optimisation refers to finding the optimal element in a given set, or more formally to determining the optimal value of a cost of an objective function in a defined domain. Mathematically, an optimisation problem can be described by the following definition.

Definition 1.1 (Optimisation problem)**.** The standard form of an optimisation problem is defined by

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimise}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & g_i(\boldsymbol{x}) \leq 0 \quad i = 1, \ldots, m \\ & h_j(\boldsymbol{x}) = 0 \quad j = 1, \ldots, n \end{aligned}$$

to find a vector $\boldsymbol{x} \in \mathbb{R}^d$ of dimensionality d that minimises the objective function $f(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}$ fulfilling the inequality constraints $g_i : \mathbb{R}^d \to \mathbb{R}$ and equality constraints $h_j : \mathbb{R}^d \to \mathbb{R}$. The optimal $\boldsymbol{x}$ is denoted by $\boldsymbol{x}^*$ with its optimal value $v^* = f(\boldsymbol{x}^*)$.

A subset of optimisation problems, convex problems, have been shown to be very practicable because of the special nature of these problems. They can be solved very reliable and efficient by applying relatively fast solving algorithms that exploit the properties of convexity. Before defining convex optimisation problems in Definition 1.4, I first introduce convexity in terms of sets and functions.

Definition 1.2 (Convex set)**.** A set C is a convex set if the line segment between any two points in C lies in C, or formally for any x_1, $x_2 \in C$ and any θ with $0 \leq \theta \leq 1$ it holds $\theta\ x_1 + (1 - \theta)\ x_2 \in C$.

Definition 1.3 (Convex function)**.** A function $f : \mathbb{R}^d \to \mathbb{R}$ is convex if the line segment between $(\boldsymbol{x}, f(\boldsymbol{x}))$ and $(\boldsymbol{y}, f(\boldsymbol{y}))$ lies above the graph of f for any $\boldsymbol{x}$, $\boldsymbol{y}$ in the domain of f. Formally, this can be described by the inequality

$$f(\theta\ \boldsymbol{x} + (1 - \theta)\ \boldsymbol{y}) \leq \theta\ f(\boldsymbol{x}) + (1 - \theta)\ f(\boldsymbol{y})$$

with $0 \leq \theta \leq 1$. From another perspective, this can be described by the condition that the epigraph of f, i.e., the set including the graph of f and all points above it, is a convex set. This inequality is also known as Jensen's inequality.

An important property of a convex differentiable function f is that its first-order Taylor approximation is a global underestimator of f. That means if f is convex, then $f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x})$ with ∇f the gradient of f. Also, the other direction of this statement holds. If $\nabla f(\boldsymbol{x})$ equals 0, then it holds that $f(\boldsymbol{y}) \geq f(\boldsymbol{x})$ for all $\boldsymbol{y}$ in the domain of f, and thus f achieves its global minimum at $\boldsymbol{x}$. This is the well-known *necessary* condition for minimality. The *sufficient* condition or second-order condition describes the relation that a twice-differentiable function is convex if and only if its Hessian is positive semi-definite, which is one frequently applied criterion to show convexity of a function. With Definition 1.2 and 1.3 a convex optimisation problem can be described as follows.

Definition 1.4 (Convex optimisation problem)**.** An optimisation problem as in Definition 1.1 is called convex, if the objective function $f(\boldsymbol{x})$ and inequality constraint functions $g_i(\boldsymbol{x})$ are convex functions, and the equality constraint functions $h_j(\boldsymbol{x})$ are affine functions.

Table 1.1.: Instances of convex optimisation problems. The problems are grouped by the properties of their objectives and constraints.

	Objective	Inequality constraints	Equality constraints
Linear programme (LP)	affine	affine	affine
Quadratic programme (QP)	quadratic	affine	affine
Quadratically constrained quadratic programme (QCQP)	quadratic	quadratic	affine
Semi-definite programme (SDP)	linear	non-negative matrix	linear matrix

A prominent example of a quadratic programme, i.e., an optimisation problem with quadratic objective and affine constraints, that is frequently applied in regression is the least-squared problem defined by

$$\underset{\boldsymbol{x}}{\text{minimise}} \quad f(\boldsymbol{x}) = \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2 = \boldsymbol{x}^T A^T A\boldsymbol{x} - 2b^T Ax + \boldsymbol{b}^T\boldsymbol{b}$$

where A is a matrix $\in \mathbb{R}^{m\times d}$.

This problem has an analytical solution given by $\boldsymbol{x}^* = A^\dagger\boldsymbol{b}$ solving a set of linear equations $(A^TA)\boldsymbol{x} = A^T\boldsymbol{b}$. If the rank of A is m, i.e., all rows of A are independent, the pseudo-inverse $A^\dagger$ is given by $A^\dagger = (A^TA)^{-1}A^T$. If the rank of A is d, then $A^\dagger = A^T(AA^T)^{-1}$.

One of the key concepts in machine learning is *regularisation* to avoid over-fitting of the data by the learned model. In terms of optimisation this can be achieved by a weighted combination of several criterion functions f_n in the objective, i.e., $f(\boldsymbol{x}) = \sum_{n=1}^{N} \lambda_n f_n(\boldsymbol{x})$. Each weight λ_n can be seen as the fraction how much the n-th criterion contributes to the overall cost. If, for example, f_n is a relative important function within the objective compared to others, then λ_n needs to be set to large value. Tuning a model is often realised in a cross-validation procedure by finding a set of weights for which the model performs best.

Often, norms, which preserve convexity, are used to measure the cost of the optimisation variables. By applying for instance an ℓ_2-norm or Euclidean norm to $\boldsymbol{x}$, $\|\boldsymbol{x}\|_2 = \sqrt{{x_1}^2 + \ldots + {x_d}^2}$, larger values of $\boldsymbol{x}$ are penalised more than smaller ones. In contrast, the ℓ_1-norm defined by $\|\boldsymbol{x}\|_1 = \sum_{i=1}^{d} |x_i|$ puts strong weight on small values of $\boldsymbol{x}$, but least weight on large values.

This strategy has also been incorporated in least-squared approximation. An ℓ_2-norm regularisation of least-squares, also known as Tikhonov regularisation problem or Ridge regression [82], is defined by

$$\underset{\boldsymbol{x}}{\text{minimise}} \quad f(\boldsymbol{x}) = \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \delta\|\boldsymbol{x}\|_2^2 = \boldsymbol{x}^T(A^TA + \delta I)\boldsymbol{x} - 2b^TAx + \boldsymbol{b}^T\boldsymbol{b}$$

where δ is a regularisation parameter. This problem has also an analytical solution, similar to the non-regularised version: $\boldsymbol{x}^* = (A^TA + \delta I)^{-1}A^T\boldsymbol{b}$ with $\delta > 0$.

When using an ℓ_1-norm regularisation with least-squares approximation one arrives at the formulation of lasso, the least absolute shrinkage and selection operator [198]. Here, no analytical solution exists, and the optimisation problem has to be solved by specific algorithms [198].

Optimisation Algorithms

For most optimisation problems, an analytical solution cannot be derived. There is a group of heuristics approaches, such as hill climbing, simulated annealing and genetic algorithms, which can be applied to solve non-convex optimisation problems. They find an approximate solution of a general optimisation problem, but they are often restricted to a local search or poorly perform in terms of exactness of the solution. Therefore, research has been conducted to develop efficient and accurate algorithms for solving optimisation problems, which are, however, often problem-specific and applicable to only a subset of problem classes.

Probably the most known algorithm in general and in optimisation is the simplex algorithm for solving linear programmes proposed by Dantzig [48, Chapter 5] in the middle of the 20th century. Many of the solving methods that have been developed more recently are iterative algorithms, i.e., they converge to the optimal solution by taking a number of steps along a certain search direction. This usually involves information about the gradient or the Hessian of the objective. Gradient descent, for example, is an approach that descends in direction of the negative gradient. Another method is Newton's method that uses information from the Hessian [51]. The step taken in each iteration can be seen as the amount to be added to a current point in order to minimise the second-order Taylor approximation of the objective at this point. Newton's method can be extended to solve problems with equality constraints.

In coordinate descent algorithms, only one optimisation variable is updated at every iteration. The steepest descent algorithm for the ℓ_1-norm, for example, belongs to this class of algorithms and adapts the current solution with respect of the coordinate with maximum absolute value [34, Chapter 9]. Luo and Tseng [129] analysed the convergence for a class of optimisation problems and algorithms including coordinate descent methods that update the current point at each iteration in the direction of the negative gradient of the objective with respect to linear equalities or convex inequalities. They showed that these algorithms converge at least linearly to the optimal point.

Interior-point methods are frequently used to solve convex optimisation problems with inequality constraints [34, Chapter 11]. The problem is reduced to a sequence of problems with equality constraints to which then Newton's method is applied. This is achieved by moving the inequality constraints into the objective, and, to obtain differentiability, by approximating this part by logarithmic barrier functions. The barrier method then solves the problem at each iteration, using the obtained optimal solution as a starting point. Another example for an interior-point method is the primal-dual method [34, Section 11.7], which often shows to be more efficient than the barrier method for several problem classes such as LPs or QPs [34, p. 609].

Recently, optimisation techniques for machine learning have been advanced by the so-called bundle methods [190, 197]. These methods solve regularised risk minimisation problems, which are convex, but which can have a non-smooth objective. The basic idea is to circumvent the non-smoothness by replacing the empirical risk term by first-order Taylor approximations and thus provide lower bounds for this term.

1.2.3. Selected Models for Sequential Data

This section briefly describes two models for sequential data, the hidden Markov Model (HMM) as an example for a generative model, and the related hidden Markov support vector machine (HM-SVM), an instance from the discriminative model class.

HMMs are probabilistic graphical models, which were proposed in the 1960s and have been popular since then, having been applied, among others, in speech recognition and computational biology. For example, the first gene finding system, GENSCAN [36], was based on HMMs, and they were also found to be suitable in protein modelling [109]. An HMM is a Markov process with hidden states and forms a probabilistic model $\boldsymbol{\theta}$ that is described by an alphabet Σ, a set of states S, transition probabilities $\phi_{i,j}$ from state i to state j, and emission probabilities $e_{\sigma,s}$ to emit the label $\sigma \in \Sigma$ in state s [54, p. 128–138]. The joint probability distribution of observing a sequence $\boldsymbol{x}$ and a path π, in other words the joint distribution over both the observed and latent variables of the model, is given by

$$p(\boldsymbol{x}, \boldsymbol{\pi}|\boldsymbol{\theta}) = \prod_{i=1}^{|\boldsymbol{\pi}|} e_{x_i,\pi_i} \ \phi_{\pi_i,\pi_{i+1}}.$$

Usually, the path sequence π is unknown. Decoding the underlying most probable path from the observed sequence $\boldsymbol{x}$ can be efficiently solved using the Viterbi algorithm. It recursively determines the optimal path $\pi^* = \underset{\boldsymbol{\pi}}{\operatorname{argmax}}\ p(\boldsymbol{x}, \boldsymbol{\pi}|\boldsymbol{\theta})$ with a dynamic programming approach [207]. The model parameters of an HMM can be inferred by maximum likelihood estimation using a special case of the expectation-maximisation algorithm, the Baum-Welch algorithm [16].

In contrast, the model parameters of an HM-SVMs are learned via discriminative training. This model for sequential data incorporates a state model similar as the one of HMMs, but it inherits two main properties of an SVM, namely the maximum margin principle and the kernel-centric approach to discriminate non-linear relationships [12, 202]. A simplified version of an HM-SVM taken from [12] can be formulated as an optimisation problem:

$$\begin{aligned} \underset{\boldsymbol{\theta}}{\text{minimise}} \quad & \frac{1}{2}\|\boldsymbol{\theta}\|^2 + \frac{C}{2}\sum_{i=1}^{n} \xi^{(i)} \\ \text{subject to} \quad & F_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}, \boldsymbol{\pi}^{(i)}) - \max_{\boldsymbol{\pi} \neq \boldsymbol{\pi}^{(i)}} F_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}, \boldsymbol{\pi}) \geq 1 - \xi^{(i)} \quad i = 1, \ldots, n \\ & \boldsymbol{\xi} \geq \mathbf{0} \end{aligned}$$

which can be easily transformed to an equivalent quadratic programme. Here, F is a discriminant function parametrised by $\boldsymbol{\theta}$ that scores pairs of an observed sequence $\boldsymbol{x}$ and path sequence $\boldsymbol{\pi}$. Moreover, $\boldsymbol{\xi}$ are slack variables to allow margin violations and C a regularisation parameter. The inequality constraint implements the idea of enforcing a large margin between the score evaluated at the correct path and the score from any incorrect path. Special cases of the HM-SVM optimisation problem are the formulation defining a multi-class SVM [230] and the standard SVM for binary classification (e.g. [43], [205, Chapter 10] and [177]). Compared to HMMs and other state-of-the-art models for structured output learning, HM-SVMs have found to be more accurate and tolerant to noise [225] as well as to perform better in terms of average loss per sequence [150].

1.3. An Introduction to Genome Biology

From a molecular view point, the key player in genomics is *DNA*. Nucleic acids were discovered by experiments on the chemical composition of white blood cells conducted by Friedrich Miescher in the castle of Tübingen in 1869 [45, 46, 144], which formed a basis for further investigations of DNA. Research by Oswald Avery and colleagues in 1944 demonstrated that DNA is the substance that carries genetic information [14], while the structure was characterised as a double helix by James Watson and Francis Crick in 1953 [212].

Most living organisms have entities comprised of DNA, also termed as the *genome* of an organism, which encodes information by the distinct sequence of nucleotides with either adenine, cytosine, guanine, and thymine as their bases [117, Chapter 1]. In eukaryotes, it is present as usually several chromosomes, which are located in the cell nucleus. During transcription, the information encoded in the DNA molecules is transferred to another class of nucleic acids, the ribonucleic acids (RNA). In their function as messenger RNA (mRNA), these transcripts can in turn serve as a template for the translation to proteins, which are the main molecular substances for enzymes in cellular pathways, motor and transport proteins, structure proteins and hormones. Beside their classical part in the central dogma of molecular biology, RNA transcripts such as non-coding RNA, transfer RNA (tRNA) and ribosomal RNA (rRNA) play an important role in regulation and catalysis of cell processes. The set of all transcripts in the cell, the *transcriptome*, is not static, but varies across different cell types and states, thus reflecting one part of the dynamics of a cell.

Sequence Variation

The diversity of living organisms mainly arises from their differences in genomic composition. To understand phenotypic diversity, it is therefore crucial to describe the differences in the genome sequence, that is the *sequence variation* within or between species [117, p. 57/58]. Alteration of sequence composition and order via mutations events on the level of the nucleotide, gene, chromosome or genome may affect phenotype. The most abundant class of sequence variations are single nucleotide polymorphisms (SNP), which are mutations to one nucleotide that have been fixed in a population, in contrast to single nucleotide variants (SNV) that are the class of individual point mutations found in somatic cells. Sequence can also be altered by deletions and insertions, duplications, inversion and by copy-number variations (CNV) due to rearrangements (cf. Figure 1.1). Due to their complex nature, these structural variants are often more difficult to detect. However, as linkage disequilibrium is higher in polymorphic regions (e.g., in *Arabidopsis thaliana* [107]) and common deletions and SNPs have the same level of linkage disequilibrium with surrounding SNPs in humans [81], the impact of these type of sequence variations can be assessed with association studies based on SNPs in vicinity.

RNA Processing and Splicing

After transcription, pre-mRNA is further processed to mature mRNA in the nucleus of eukaryotic cells [117, Chapter 7]. To prevent degradation of the RNA molecule by exonucleases, the RNA is modified at its ends by adding a methylation cap at the 5' terminus, and by extending the molecule at the 3' terminus with a long sequence of A-nucleotides (polyadenylation).

Another important modification step is *splicing*, which is the process of removing parts of the pre-mRNA, called introns, and connecting the remaining parts, the exons [117, Chapter 26].

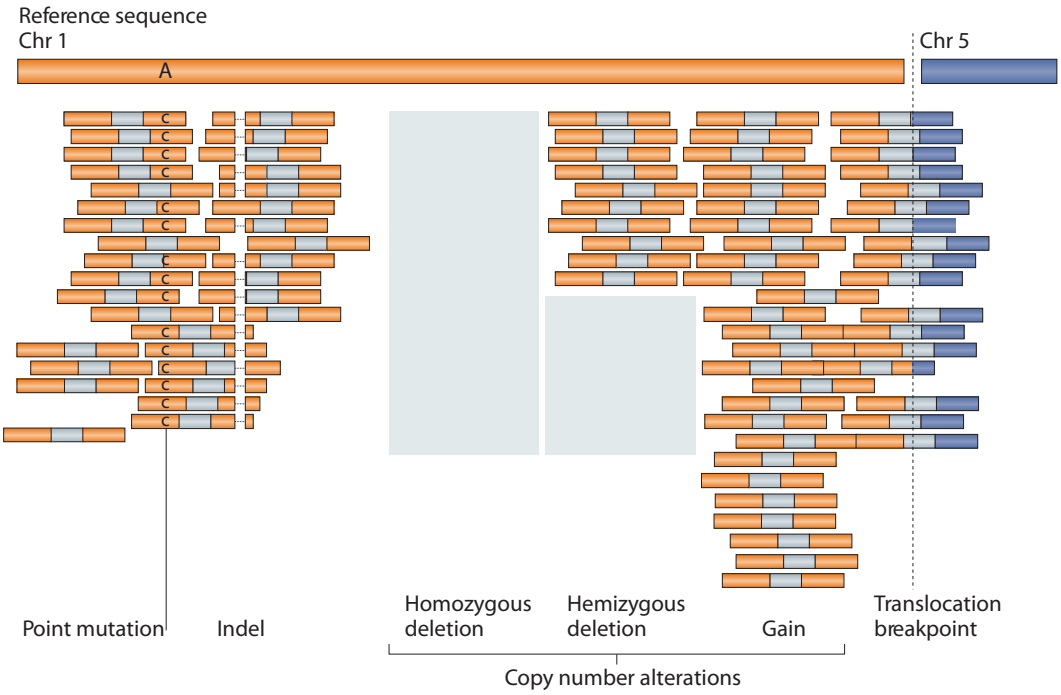


Figure 1.1.: Different types of sequence variations are illustrated detected by paired-end reads from next-generation sequencing (cf. Section 1.4.2). The reads are aligned to chromosome 1 and 5 of the human reference sequence. The grey region between the two reads of a pair shows the unsequenced portion of the fragment. A point mutation from A to C is shown as well as a short deletion. Changes in sequencing depth give hints to copy number alterations. Translocation events can be derived from fragments covering different genomic loci. Reprinted by permission from Macmillan Publishers Ltd: Nature Reviews Genetics [143], copyright (2010).

The intron is a region in the mRNA is defined by a set of *cis*-elements. The donor and acceptor splice sites determine the exon-intron junctions at the 5' and 3' end of the intron, which show highly conserved consensus sequences (GU and AG, respectively). Other elements are the branch site located 18 to 40 nucleotides upstream of the 3' end of the intron, and the polypyrimidine tract located between the branch site and the 3' end of the intron. Furthermore, exonic and intronic splicing enhancers and silencers and their corresponding binding regulators, such as SR proteins and hnRNPs, play a crucial role in splice site recognition and splicing regulation [39].

Usually, the splicing process is catalysed by the spliceosome, a large complex of proteins and five small nuclear RNAs, which are recruited and assembled for this purpose during or after transcription, and which recognise the above mentioned *cis*-elements [210]. Whether splicing occurs mostly co-transcriptionally or post-transcriptionally is still unclear [76]. Briefly, during splicing, the subunits of the spliceosome first bind to the donor site and the branch site, and catalyse the reaction of cutting the RNA at the donor site and linking the intron-exon part to an A in the branch site. This so-called lariat is removed by cleaving at the acceptor site, and the two exons are ligated.

The process of *alternative splicing*, i.e., generating more than one mature mRNA from a transcribed RNA by differential linkage of exons, is more abundant than first thought (cf. Figure 1.2 (b)). For example, in humans more than 95 % of multi-exons genes are estimated

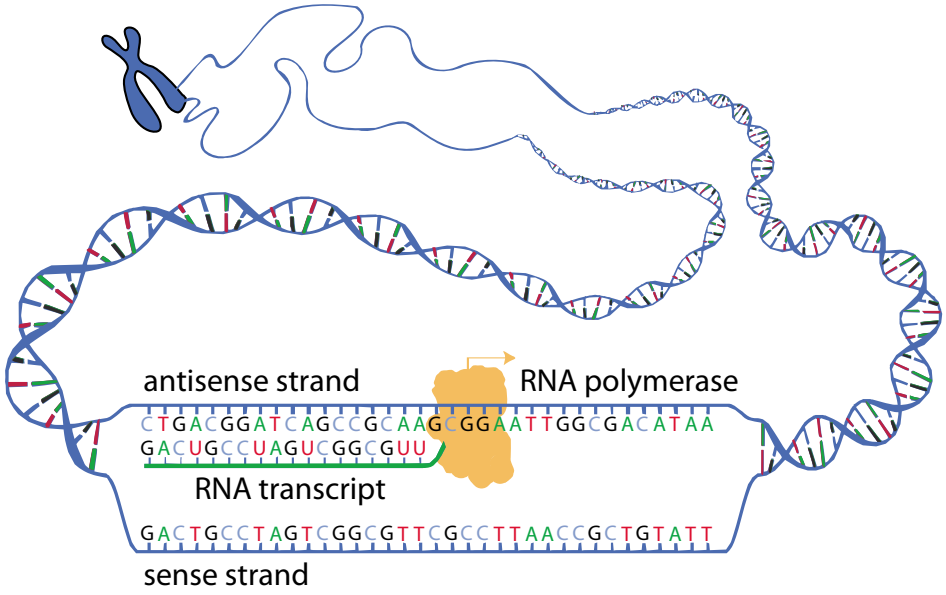
to be alternatively spliced [158]. In the model plant *Arabidopsis thaliana*, this has been recently estimated at 42 % [59]. Alternative splicing is one mechanism by which an organism can enlarge its transcriptome and proteome diversity while not simultaneously increasing the DNA sequence. It is also important for regulation of gene expression, especially in pathways specific to tissues or developmental stages. Depending on the organism, different classes of alternative splicing events occur more frequent than others. In humans, the most abundant class of events is exon skipping, in which either an exon is kept or skipped. Intron retention, an alternative splicing event in which a whole intron is retained, is more common in, e.g., *A. thaliana*. Also, alternative 5' or 3' splice sites may affect the structure of the transcript. Advances in high-throughput sequencing have helped to identify alternative splicing on a genome-wide scale, also with the aid of computational approaches [84], and to assemble a splicing code that describes splicing features and regulatory sequences, facilitating tissue-specific predictions of alternative splicing [15].

Profiling complete transcriptomes of different tissues or conditions in a qualitative and quantitative way, e.g., by next-generation sequencing is an important prerequisite to analyse changes in abundances of transcript isoforms, to identify novel isoforms and to simultaneously capture variation of *trans*-acting splicing factors [15]. Furthermore, the view on whole transcriptomes helps to reveal loci with genetic variation that correlate with splicing patterns (splicing quantitative trait loci, sQTL) [113]. These kind of studies are crucial steps towards a better understanding of the process and regulation of alternative splicing.

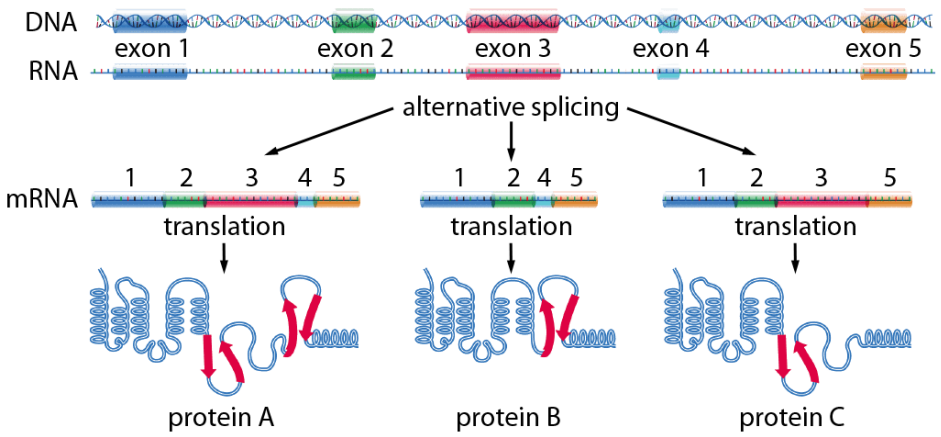
RNA Structures

As with DNA, RNA does not usually exist as a linear molecule in the cell, but forms a secondary and tertiary structure, which is often crucial for its function and regulation. Typically, RNA secondary structure is characterised by stem-loops, where Watson-Crick pairing of U-A and G-C form stems combined with loops and bulges of unpaired bases [1, Chapter 3]. A prominent example of the importance of RNA tertiary structure for functionality is the cloverleaf structure of tRNA. Furthermore, rRNAs in ribosomes build the framework to which ribosomal proteins are attached. Interestingly, RNA structure has recently been linked to a few examples of alternative splicing, such as splice site suppression and a looping-out mechanism of exon skipping, but the general mechanism in alternative splicing is still unclear and the focus of on-going research [101, 138].

RNA structure of a single molecule can be determined by methods such as X-ray crystallography, NMR, and cryo-electron microscopy [106]. One major drawback is the low-throughput nature of these techniques, and the limitation to short molecules, hence only hundreds of structures of mostly short sequences have been solved by now. Higher throughput *in vitro* approaches based on chemical and enzymatic probing exploit the difference of paired and unpaired bases to infer the secondary RNA structure (cf. Figure 4.1 in Chapter 4 and [106, 203]). However, *in vivo* approaches are necessary to determine RNA structure formations that depend on the cellular environment and conditions.



(a) Transcription.



(b) RNA splicing.

Figure 1.2.: Transcription, RNA splicing and translation. (a) During transcription, RNA polymerase generates an RNA transcript based on a DNA template. (b) After transcription, the pre-mRNA is further processed. In splicing, introns are removed from the transcript and the remaining exons are linked in the mature mRNA. In the process of alternative splicing, more than one mRNA can be produced from a premature transcript by differential linkage of exons. For protein coding transcripts, mature mRNA serves as a template for the translation of proteins. Illustrations are taken from [215] and [93].

1.4. Sequencing Technologies

1.4.1. Tiling Arrays

For the last 15 years, microarrays have been a well-established technology. Before the advent of next-generation sequencing technologies (cf. section below), they were the high-throughput method of choice for many application in genome biology. They may still be cost-effective for medical screening and genotyping of a pre-defined set of SNPs or CNVs.

The array technique measures complementary binding, or hybridisation, of sample DNA to DNA molecules that are immobilised on a solid surface such as glass in a dense grid [166]. Arrays are produced by either spotting pre-synthesised oligonucleotides on the array surface, or by *in situ* synthesis, resulting in tens of thousands probes. The target sample hybridised to the array is usually amplified by PCR, purified and processed to obtain single-stranded fluorescently labelled DNA. Scanning with a fluorescence microscope measures the fluorescence intensity for each probe-target pair, reflecting the degree of hybridisation of a certain probe to DNA from the sample (cf. Figure 1.3).

Microarrays have been widely used for gene expression studies. One drawback is that their design is normally based on a set of annotated genes, requiring prior knowledge of gene structures. For genotyping arrays, a pre-defined set of SNPs is queried across a larger set of individuals [44]. Improvements in array technology have enabled the development of arrays based on a tiling strategy. By designing array probes querying roughly equally spaced and dense positions in the desired sequence, these tiling arrays can interrogate entire genomes or transcriptomes in an unbiased way [145]. This approach has been applied to resequence genomes of subspecies in a more cost- and time-effective way than traditional Sanger sequencing for human [80], mouse [63], *A. thaliana* [41], and domesticated rice [140] (cf. Chapter 2). Such resequencing studies have enriched the set of known polymorphisms within a species, which have been used to design additional arrays to genotype more individuals, contributing, for example, to the first human haplotype map [94, 95]. Besides the application to polymorphism detection, whole-genome tiling arrays have been used to conduct transcriptome studies, to investigate non-coding RNAs, and to derive epigenetic features and transcription factor binding sites with the ChIP-chip technique [71, 222].

1.4.2. Next-generation Sequencing

The development of NGS technologies in the last ten years has revolutionised the way and the pace in which large-scale studies in genome biology are undertaken. Having determined the first genome sequence of a bacteriophage in 1977 [174], the Sanger sequencing technology that emerged from this project was further improved in subsequent years [136, 175]. This led to the sequencing efforts of predominately model organisms in the late 1990s and a human reference genome sequence at the turn of the millennium [96, 114, 206]. The bottleneck for Sanger sequencing, which is now also termed 'first-generation' sequencing, is PCR amplification of the sample DNA followed by plasmid cloning and loading of capillary-based sequencing machines. This results in costs of $ 500 per Mb, but provides highly accurate sequence reads of up to 1000 bp in length [187]. In contrast, NGS technologies can be highly multiplexed and generate more data from less starting material [91, 141, 142, 187]. However, until recently, NGS reads are shorter and with a higher error rate than Sanger sequences.

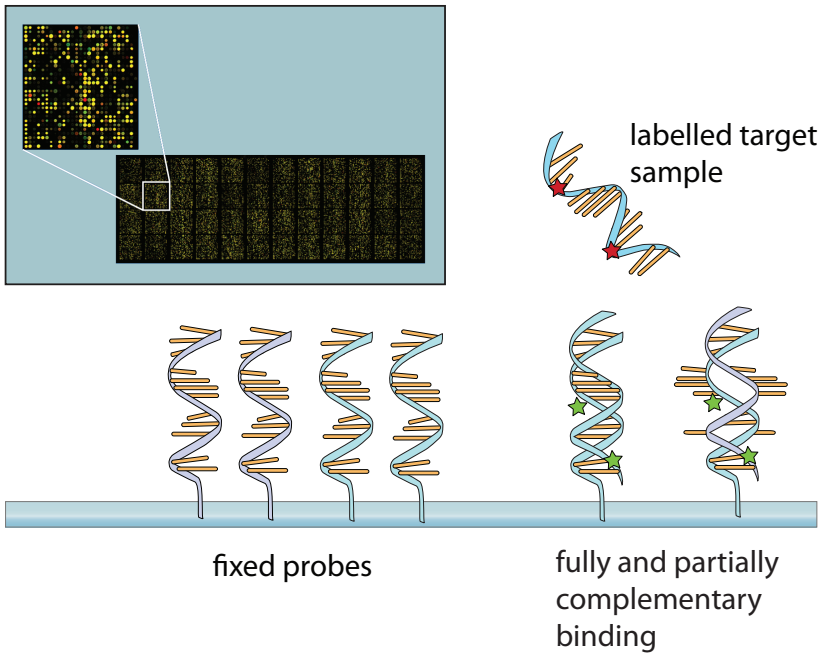


Figure 1.3.: Design of microarrays. Microarrays are composed of oligonucleotide probes that are attached on a glass surface. A target sample that is fluorescently labelled gives rise to a hybridisation signal when binding complementary to a probe. The upper left part shows the resulting image reflecting the degree of hybridisation of the fixed probe and the sample. The illustration is adapted from [216] and [217].

Technologies

All 'second-generation' sequencing technologies have entered the market since 2005 and are based on the same principle of *cyclic-array* sequencing [142]. Here, dense clusters or colonies of DNA are generated from the sample by DNA fragmentation and adaptor ligation. Sequencing is then conducted by iteratively cycling through enzymatic reactions in the clusters. These extension reactions are monitored via imaging fluorescent emission and translated into a sequence. The major differences between the technologies lie in how the DNA is amplified and the cyclic sequencing reaction is implemented. The aim of clonal amplification is to provide high-density clusters of the same molecule sequence such that fluorescent events are detectable. One strategy is *emulsion PCR*, in which a single DNA molecule is caught attached to a bead, and amplified within a droplet to create thousands of copies of the same sequence. Bridge PCR or solid-phase amplification is a second amplification strategy where immobilised templates are tethered to adjacent primers.

The most frequently used NGS machines are Illumina's Genome Analyzer (GA) and HiSeq with a market share of 62 % [4]. This technology uses bridge PCR for amplification and polymerase-based sequencing using unique fluorescent labels for each nucleotide with cyclic reversible termination [22]. Currently, Illumina machines produce reads of length up to 150 nt and have a throughput of up to 54 to 60 Gb (GA) and 540 to 600 Gb (HiSeq) per run for a cost of less than $ 1 per Mb.

At the same cost level, the company Life Technologies/Applied Biosystems provides sequencing machines based on the SOLiD principle [204]. DNA is amplified by emulsion PCR and then applied to ligase-based sequencing using support oligonucleotide ligation detec-

tion (SOLiD). The special feature here is the use of 2-base-encoded probes that read each target nucleotide twice and generate so-called colour space reads. Polonator is a cheap and open-source implementation of the SOLiD principle offered by Dover Systems [188].

The first NGS machine on the market was offered by Roche/454 and is based on pyrosequencing [132]. It generates longer reads at a higher cost ($ 60 per Mb). Amplification is via emulsion PCR, and the sequenced reads are generated by pyrosequencing. This special method adds a single nucleotide at each step and measures its incorporation by releasing pyrophosophate with light emission measured by digital imaging.

A newer sequencing method developed by Helicos BioSciences does not use amplification, but sequences single molecules [78]. It is therefore free from PCR biases. Sequencing itself is implemented as one-colour cyclic reversible termination.

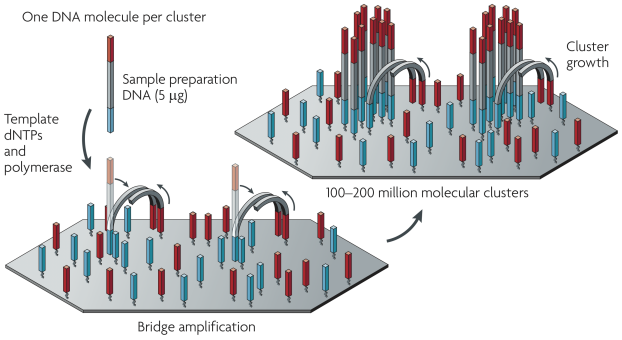
The 'third-generation' sequencing machines are on their way and one has already entered the commercial market. Pacific Biosciences has developed a technology that does not use halted progress of sequencing, but sequences in real time, imaging continuously the incorporation of dye-labelled nucleotides by polymerases that are attached to zero-mode waveguide detectors [56]. Another promising approach is Nanopore sequencing, where the sequence molecule passes through a nanopore and the differences of conductance characteristic for each nucleotide are determined [35].

Applications

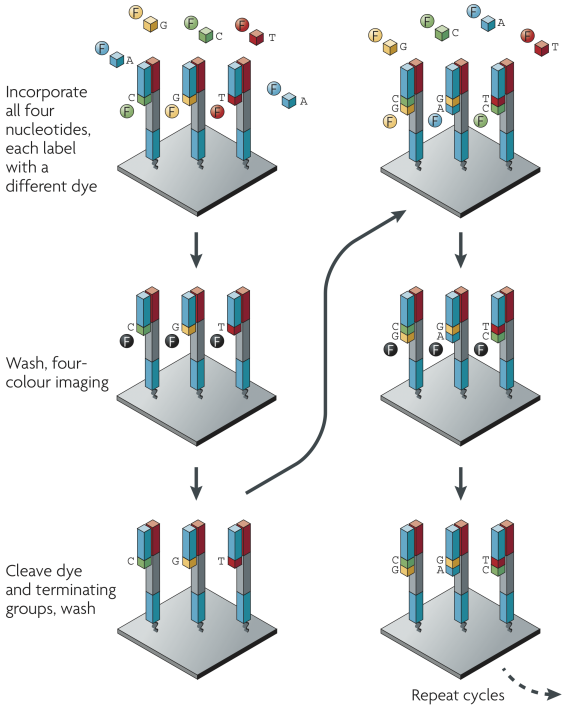
NGS technologies have obvious benefits compared to tiling arrays, as they provide a more extensive quantitative range to measure gene expression and a better resolution to characterise novel alternative splicing events, but they have limitations in measuring low abundant molecules [10, 133]. While replacing arrays in many large-scale applications, NGS has also opened new directions in genomics and transcriptomics [142, 187]. One major application has been the discovery of polymorphisms and structural variations by genome (re-)sequencing and genotyping [11, 52]. Many projects have been started in this direction, e.g. the 1000 genomes project for humans [9], 1001 genomes project for *A. thaliana* [38, 153, 213] and the 18 genomes and transcriptomes project for *A. thaliana* [64]. For computational analysis, resequencing relies on a pre-existing reference genome sequence, but often this sequence has not been generated for the organism of interest. Therefore *de novo* sequencing and assembly has also been approached by NGS technologies, but is still limited by the relative short read lengths.

To accurately discover short polymorphisms, such as SNPs and short insertions and deletions, the depth of coverage is an important consideration in the experimental design [143]. Sufficient physical coverage, i.e., the number of fragments spanning a site, is crucial to resolve structural rearrangements, and can be aided by an extension of the NGS protocol using paired-end sequencing, i.e., sequencing from both ends of the fragments. Paired-end sequencing is also highly useful for *de novo* genome and transcriptome assembly and for the study of alternative splicing. To increase coverage depth at a sufficient low cost, targeted sequencing has been used to analyse only the exome or smaller regions of interest [143, and references therein]. Another cost-effective approach for variation studies is restriction site-associated DNA sequencing, which is used to identify genetic markers on a genome-wide scale [49].

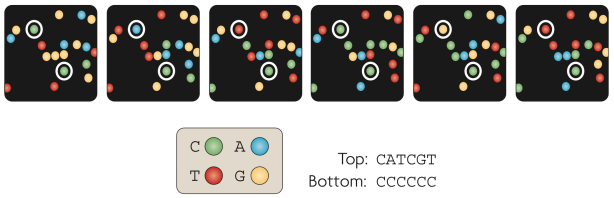
Another area of research that has profited from NGS is metagenomics, the study of microbial diversity in a sample from one habitat, e.g., in human microbiomes [142, 164] and the analysis of environmental sequences [90]. Moreover, NGS technologies have been applied in epigenetics to derive patterns of histone modifications, methylation and nucleosome occupancy by



(a) Amplification by bridge PCR.



(b) Cyclic sequencing reaction.



(c) Imaging.

Figure 1.4.: DNA sequencing with Illumina's technology. (a) After breaking genomic DNA or cDNA into smaller fragments and attaching adaptors to their ends, the fragments are amplified by bridge PCR. The forward and reverse primers for the PCR are complementary to the adaptors and attached to a slide in a highly dense grid, giving rise to an amplified cluster of the same template. (b) Sequencing is conducted by a cyclic sequencing reaction. Here, each iteration consists of the following steps: incorporation of all four nucleotides which are fluorescently modified, washing, imaging of the incorporated nucleotides and cleavage of the terminating chemical group and fluorescent dye. (c). Sequence reads are derived from four-colour images by concatenating signals from each iteration. In the example images, sequencing data for two different templates are highlighted (top and bottom). Reprinted by permission from Macmillan Publishers Ltd: Nature Reviews Genetics [142], copyright (2009).

particularly designed protocols called ChIP-seq, methyl-seq, and DNase-seq [159, 162].

Besides its applications to genomics, NGS has been also heavily used to study transcriptomes on a large-scale and at high resolution. The aims of RNA-sequencing (RNA-seq) are not only qualitative, but also quantitative. RNA-seq has been shown to be suitable for annotation of novel genes and transcripts, their quantification, determination of differential gene and transcript expression, analysis of alternative expression, detection of alternative splicing events, profiling of small RNAs and identification of transcribed SNPs [72, 187, 211, 218, and references therein].

The major part of NGS experiments is the computational analysis of the data [85]. Sboner et al. [176] recently estimated that the cost of sequencing is $\approx$ \$ 6500 on an Illumina machine, which is roughly at the same level for downstream data analyses. For RNA-seq analysis [155, 162], the pipeline usually starts with alignment of reads to a reference genome by specialised tools such as PALMapper [50, 98] and TopHat [199], or with *de novo* assembly into transcripts [70, 134, 172]. Read mapping for transcriptome data is more challenging than for genomic reads due to reads spanning exon-intron junctions and the high variation of coverage. Once transcripts have been annotated and per-base read coverage determined, quantification and analysis of differential gene and transcript expression can be undertaken.

Most current NGS technologies do not sequence RNA but DNA. Therefore, sample RNA needs to first be reverse-transcribed to obtain a cDNA library before sequencing. Also, strand-specificity needs to be preserved during library generation to study anti-sense transcription ([160, 163, 189, 208] and [116, and references therein]). Alternative approaches exist that circumvent the bias-inducing RT-PCR step. For example, Helicos BioSciences provides direct RNA sequencing [125, 155–157], determining the sequence of single RNA molecules without the need of reverse transcription. Real-time sequencing approaches for RNA-seq by Pacific Biosciences are on the way [23].

NGS has revolutionised research in genome biology and has changed the way how studies in this area are undertaken nowadays. It has enabled high-throughput measurements of whole genomes and transcriptomes at reasonable time and costs. Many diverse studies based on NGS have been realised since the advent of NGS. For example, the sequences of personal genomes have been determined to describe sequence variation in humans [9, 142]. Also, as the application areas of NGS are diverse, different type of NGS-based sequencing can be combined to simultaneously profile, e.g., genomes, transcriptomes and methylomes from different individuals or experimental conditions and to exploit this knowledge by finding relations within and across these different measurements. In a study by Gan et al. [64], for example, the genomes and transcriptomes of 18 *Arabidopsis thaliana* strains were sequenced by NGS. RNA-seq based annotation of each of these genomes separately showed that polymorphisms were often not disruptive, but compensated by an alternative gene model compared to gene models of the *A. thaliana* reference strain Col-0. The 18 strains together with Col-O are parents of more than 700 inter-crosses and sequencing their genomes and transcriptomes will allow genome-wide associations studies at an unprecedented resolution.

2. Detecting Sequence Variation from Resequencing Arrays

2.1. Introduction

For thousands of years, rice has played a major role for human food supply. From archaeological rice remains dating from 8,000 BC, it is believed that domestication and cultivation of the crop have been taken place for 10,000 years [196]. Rice feeds more than half of the human population today, being the most important food crop in Asia. It is one of the major source for caloric intake (20 % per capita) and protein supply (15 % per capita) [3]. To meet the food demand by the growing human population, rapidly increasing primarily in the developing countries, rice production must be augmented by at least one quarter [140]. One of the key prerequisites towards satisfying this demand is to study sequence variants in varieties of domesticated rice that confer advantageous traits and thus are targets for modern breeding of rice.

In scientific terms, rice belongs to the genus *Oryza*, including both wild and domesticated rices. *Oryza glaberrima* is a rice species growing in West Africa, while *Oryza sativa* is distributed all over the world with wide appearance in Asia [196]. The latter can be divided morphologically as well as genetically into five major subgroups. The traditional *indica* subspecies includes the *aus* and *indica* subgroups; *temperate japonica*, *tropical japonica* and *aromatic* define the *japonica* subspecies. The estimation of genetic divergence of the two main subspecies *indica* and *japonica* to 100,000 years suggests that two or more independent domestication events in *Oryza sativa* happened to form today's subpopulations [196].

Because of its key role in food supply and its relatively well-arranged and reasonable sized genome compared to other crops, rice has become the second most prominent model organism for plants besides *Arabidopsis thaliana* (thale cress). Efforts have been undertaken to determine the genomic sequence of two *O. sativa* subspecies by shotgun sequencing, resulting in the first sequence versions for the genomes of the *temperate japonica* subspecies Nipponbare [68, 97] and the *indica* subspecies 93-11 [223]. Based on the Nipponbare sequence published in 2005 [97], the genome size was estimated to 389 Mega bases (Mb) located on twelve chromosomes and harbouring about 32,000 genes that were conservatively identified by cDNAs and expressed sequence tags within the Rice Annotation Project [168].

Focusing on this highly important crop, this chapter presents a study on the identification of SNPs and polymorphic regions from array-based resequencing data across a selection of 20 domesticated rice varieties, arisen in a collaborative resequencing effort of the OryzaSNP project (`www.oryzasnp.org`) [139, 140]. Polymorphic regions, hereafter abbreviated as PRs, are in this context defined as regions in the genomic sequence that harbour a collection of SNPs, short insertions and deletions or a combination of these. Detecting PRs from hybridisation data is a much more difficult problem than that of calling isolated SNPs because neighbouring polymorphisms impair hybridisation signals (see also Figure A.1). After having reviewed related work for studying sequence variations in rice as well as in other organisms in the following Section 2.1.1, I describe the methods developed and applied for polymorphism

detection in Section 2.2. They comprise approaches for normalisation of array data and annotation of repetitive probes in the reference genome (Section 2.2.2), the SVM-based SNP calling approach (Section 2.2.3), the PR detection method based on a machine learning algorithm (Section 2.2.4) and the inference of non-redundant PRs (Section 2.2.5). In Section 2.3, I present results in terms of performance of the applied algorithms, show a comparison of the PR predictions to SNP calls, describe the assembly of a high quality set of PR predictions as well as PR sets suitable for primer design, and give an analysis about protein domains affected by PRs.

2.1.1. Related Work

Research on polymorphic patterns and sequence divergence between rice varieties has been undertaken since the first drafts of the rice genome were published in 2002 [68, 223]. Here, one major goal has been to assemble an inventory of polymorphisms in diverse varieties on a genome-wide scale, which is essential to carry out analyses addressing questions on rice evolution and population genetics. Studies in these fields often investigate which SNPs are causal for the variation in complex traits by using statistical methods such as quantitative trait locus (QTL) mapping [214]. To have enough statistical power in such QTL studies, a large number of genotyped individuals along with measurement of relevant traits is necessary [152]. All sequence variation studies in the first few years after publication of the two genome assemblies were limited to a small selection of varieties [146, 186] or to certain parts of the rice genome [37, 75, 126]. For example, a study of genome-wide patterns of polymorphisms described by Shen et al. [186] investigated the differences between the genomic sequence of the varieties 93-11 and Nipponbare, resulting in a polymorphism database of 1,703,176 SNPs and 479,406 indels with a SNP frequency of 3.5 SNPs per kb. Looking at a larger number of varieties but being still constrained to a selection of regions, 111 randomly chosen gene fragments were sequenced for 72 domesticated *O. sativa* and 21 wild *O. rufipogon* varieties in a study published by Caicedo et al. [37].

At the time when the OryzaSNP project [139] was initiated, the high-throughput method of choice for whole-genome SNP identification was the approach of array-based resequencing. Here, the genomes of multiple varieties of a certain species were resequenced with high-density oligonucleotide microarrays that are designed using a high-quality reference genome as a template. Comparing a given target variety to the reference sequence allowed to detect differences between these two sequences in a more efficient and cost-effective way than using the Sanger method, which was the state-of-the-art technology for *de novo* sequencing at this time. Array-based resequencing has been applied to discover SNPs in human, mouse and *A. thaliana*. Hinds et al. [80] described whole-genome patterns of common human DNA variations by resequencing genomes of 24 human individuals from Europe, Africa and Asia and genotyping roughly 1.5 million SNPs. Clark et al. [41] identified common sequence polymorphisms of 20 diverse *A. thaliana* strains, resulting in the discovery of one million non-redundant SNPs. Similar methods were used for the mouse resequencing project, in which 8.27 million SNPs densely distributed across the mouse genome were identified for 15 mouse strains [63]. In 2006, the International Rice Functional Genomics Consortium (IRFGC) set up the OryzaSNP project to discover sequence variation with 20 selected varieties of domesticated rice (cf. Table B.1 and [139]). Within this project, work on SNP and PR discovery was conducted, which is presented in this chapter.

Recent advances in NGS technologies have also promoted the resequencing efforts in domesticated rice in a more cost-effective way. In a study published by Huang et al. [88], the genomes of 150 recombinant inbreed lines derived from a cross between *japonica* and *indica* varieties were sequenced with Illumina sequencing, constructing a genetic map for these lines. In a more extensive resequencing project, 517 rice varieties were sequenced to identify about 3.6 million SNPs, resulting in a high-density haplotype map of the rice genome [89]. This rice HapMap enabled to conduct genome-wide association mapping for 14 traits in 373 *indica* varieties, identifying 80 loci associated with these traits. Besides deepening our knowledge about the content of the genomic sequence in rice varieties, digital sequencing also helped in improving the functional annotation of the transcriptome by RNA-seq. The investigation of RNA-seq data from one *japonica* and two *indica* varieties made it possible to identify ≈ 16,000 novel transcriptional active regions and thousands of SNPs in transcribed sequences [128].

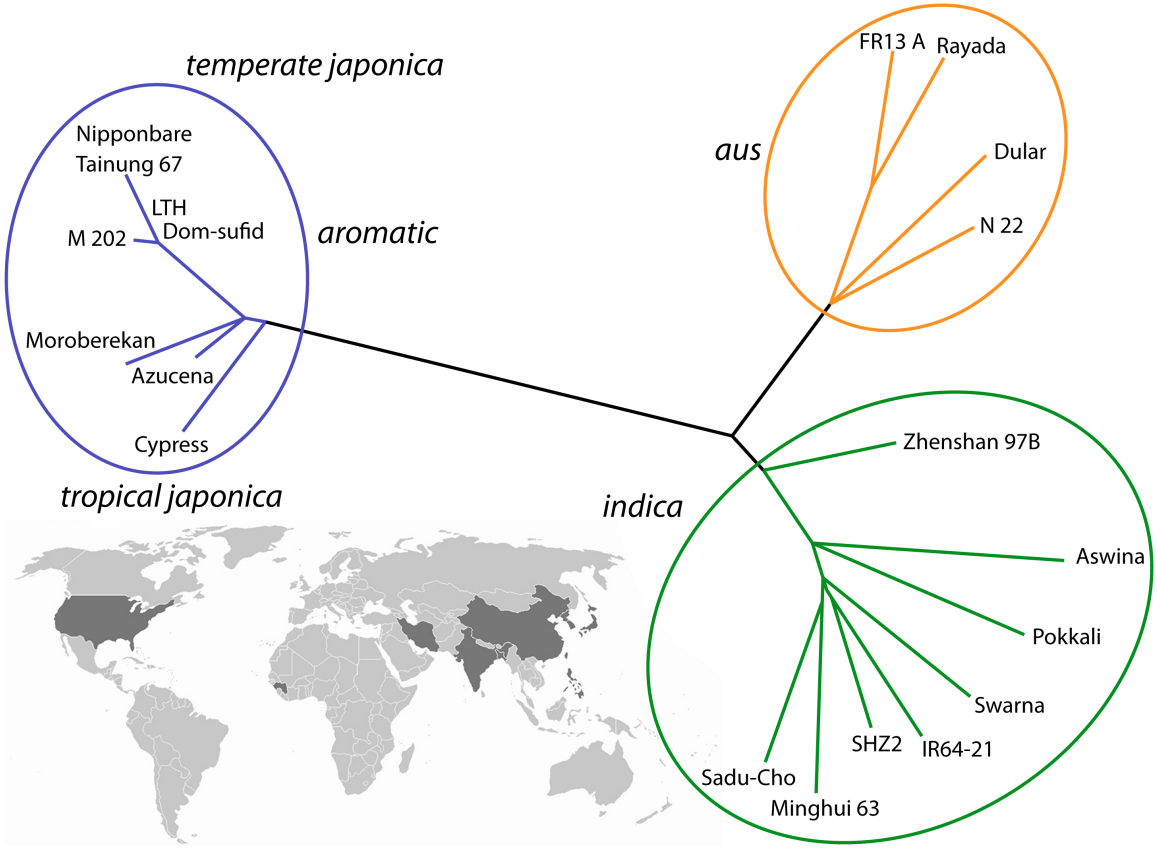


Figure 2.1.: The evolutionary distance of the 20 selected rice varieties according to the OryzaSNP set. Criteria for selection of the varieties were the value of the variety in breeding and genetic studies and the relative diversity to each other [139]. The geographic origin of the varieties is indicated in dark grey at the bottom. The phylogenetic tree is based on the MBML set and adapted from Figure 2 A in [139].

2.1.2. Publication Note

The OryzaSNP project was joint work with research groups at several international research institutes, including researchers at the Michigan State University, USA, the International Rice Research Institute (IRRI) on the Philippines and the Max Planck Institute for Developmental Biology in Tübingen. The application of the rice resequencing data set for the discovery of SNPs and polymorphic regions was joint work with Georg Zeller, Richard Clark, Gabriele Schweikert, Kevin Childs, Gunnar Rätsch and Detlef Weigel. Detlef Weigel and Gunnar Rätsch designed research; Gunnar Rätsch conceived the machine learning part of the project. Regina Bohnert, Georg Zeller and Richard Clark prepared the array data. In addition to Regina Bohnert, Gabriele Schweikert implemented code for SNP calling and Georg Zeller for data normalisation and polymorphic region detection. Regina Bohnert performed the experiments. Kevin Childs integrated polymorphic region predictions into a genome browser and performed the biological analysis. Work presented in this chapter has been in part published in the following publications. The OryzaSNP project including the sections on the analysis of repetitive probes, normalisation of the array data and SNP calling has been described in Bohnert [27] and McNally et al. [140]. Parts of the material covering the polymorphic region detection has been published in Bohnert et al. [29] and Gan et al. [64] and presented at ISCB Student Council Symposium 2008 and the ISMB Conference 2008.

2.2. Methods

2.2.1. The Resequencing Array Data

The genomic sequence of the *O. sativa* subspecies Nipponbare, IRGSP build 4 [97] was used as the reference for the design of the resequencing arrays. Because repetitive sequence parts very likely distort hybridisation signals by cross-hybridisation, a repeat-masked version of this sequence was used as a template for the design of array probes with an 1 bp tiling strategy, querying 100,104,806 bp (32 %) of the complete genomic sequence. Each queried nucleotide was represented by eight 25-mer probes (cf. Figure 2.2), which were synthesised with light-directed photolithography in conjunction with chemical coupling on the arrays by the manufacturer Affymetrix. The probes were distributed over six wafers; five of them containing 49 arrays and one containing one array used as a replicated development array to test experimental conditions. Therefore, 246 arrays were manufactured for each variety, providing 800,838,448 measurements per variety.

The target DNA that was hybridised to the array probes was generated by using 13,582 selected long range polymerase chain reaction (LR-PCR) amplicons, spanning 11,343 non-overlapping sequence fragments and covering 117,834,417 bp of unmasked genomic sequence. To facilitate the experimental setup, the amplified sequences were split up into 16 pools for each variety with around 1,000 amplicons per pool (only 76 amplicons in the pool for the developmental array). By labelling the target DNA with a fluorescent compound, the fluorescence intensities for each probe-target pair were measured reflecting the degree of hybridisation. The outcome of the array experiments were average fluorescence measurements in a range of 0 to $\approx$ 4,000 raw base calls for each strand referring to the base with the highest hybridisation intensity within a probe quartet, and discrete quality scores reflecting the error probability of calling a certain base [57]. For more details about the experimental protocols for amplification and hybridisation I refer the interested reader to the Supporting Information from [140].

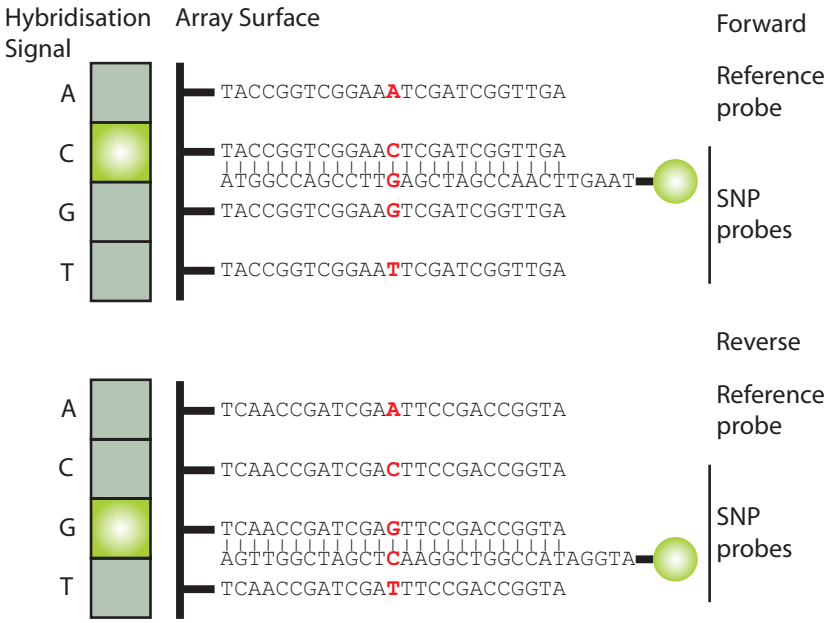


Figure 2.2.: Design of the resequencing tiling array. Each nucleotide in the repeat-masked set of the genomic sequence was queried by eight probes. The reference probe is the 25-mer *identical* to the sequence template centered at the queried base. The other three probes, the SNP probes, are sequence variants of the reference probe by replacing the centre nucleotide with one of the three other possible nucleotides. This strategy is realised for both the forward and the reverse strand, resulting in a probe octet. The example shown here illustrates the hybridisation of a target DNA fragment labelled with a fluorescence indicating a SNP at the queried position.

2.2.2. Preparation of the Input Data

Before the actual use of the hybridisation measurements and other input data for SNP calling and identification of polymorphic regions, it is very crucial to prepare the data in a proper way. Therefore, I first describe the normalisation of the array data, the annotation of repetitive probes and the assembly of the gold standard set of polymorphic annotations used in training and evaluation, which are all necessary inputs for the proposed algorithm. The hybridisation intensities, denoted by I, were mean fluorescence measurements for each of the four bases A, C, G and T on each of the forward and reverse strand. Furthermore, a raw base call B referred to the base for which the hybridization intensity was highest within a probe quartet. Additionally, quality scores QS, estimating the error probability of calling a certain base, were used.

Normalisation of the Array Data

Numerous sources, such as differences in sample preparation, variability during hybridisation or different experimentators, can cause technical variation in a multiple microarray experiment setup. For comparability of the data and thus the detection of true biological variation, normalisation of the hybridisation data is an essential preprocessing step.

For the normalisation of the rice resequencing data, a widely used technique that is based on quantiles was applied. The so-called quantile normalisation described by Bolstad et al.

[31] implements the idea that distribution of the intensity values should be identical for each array in a set of N arrays. This can be visually verified by a straight line along the unit vector $\mathbf{d} = (\frac{1}{\sqrt{N}}, \ldots, \frac{1}{\sqrt{N}})$ in an N-dimensional quantile-quantile plot. During quantile normalisation, the data vectors of each array with length P are first pooled into a $P \times N$ matrix X. A matrix X' is obtained by sorting each column, i.e., the intensity set of each array, by a permutation Π_i, $i = 1, \ldots N$. Each entry in X' is then replaced by the mean intensity value across the rows of X'. Rearranging each column by Π_i^{-1} finally transforms X' to the matrix of normalised intensities.

The hybridisation data was first normalised as described above on the level of amplicon pools across all varieties. To correct for different amplicon pool sizes, each pool was filled up with sham intensities to the identical maximal pool size. The sham intensities were sampled from the observed distributions. The data from the developmental array (pool 0 on wafer 0) was not considered for mean estimation because of too little data in this set. The intensities from pool 0 were normalised by averaging the outcomes when sorting according to the normalized intensities from five pools randomly selected from the 15 other pools (cf. Supporting Information from [140]).

As LR-PCR products served as targets to interrogate the probes on the arrays, varying levels of amplicon concentrations can also introduce technical variation of the measured hybridisation intensities. In a second normalisation step, the data was therefore corrected for between-amplicon variation by quantile-normalisation on the level of amplicons. Data corresponding to amplicons shorter than 3,000 bp were excluded from normalisation and further analysis. The final set was obtained after applying again a similar strategy as in the first normalisation step and contained 98,176,752 normalised intensity octets per variety.

Annotation of Repetitive Probe Sets in the Reference Genome

The special design of resequencing arrays having eight probes for each queried nucleotide increases the chance of cross-hybridisation and therefore unspecific and unreliable signals. Cross-hybridization of repetitive sequences can either suppress a true SNP signal and thus reduce sensitivity, or generate an incorrect SNP signal, decreasing specificity. Approximately three quarters of the *O. sativa* genome is repetitive, contributing to the appearance of cross-hybridisation. Because repetitive sequences were not completely excluded from the ≈ 100 Mb sequence interrogated with the arrays, it was important to identify oligomers with multiple occurrences. Probes were identified with a method described in the following for annotating a probe as either unique or repetitive according to defined sequence matching criteria. The annotated repetitive probe set helped in treating these probes separately in subsequent analyses.

Repetitive probes were annotated by identifying oligomers that match at least to one other 25-mer in the target DNA, allowing for some degree of degeneracy. Because all four possible nucleotides were represented at the centre position of each 25-mer, mismatches at the centre nucleotide in addition to peripheral mismatches were allowed. Based on the analysis undertaken for the *A. thaliana* resequencing project [41], three sequence match types were distinguished. Briefly, they are defined as matches for which all but the centre nucleotide of the oligonucleotide are identical to at least another 25-mer in the target genomic sequence (exact 25-mer matches), for which one additional mismatch in the inner part of the oligonucleotide is allowed (inexact 25-mer matches), and for which there exists at least one mismatch at the two positions at the boundaries (short 25-mer matches) (cf. see Supporting Online Material, p. S4/5 from [41], [27]). Additionally, the mismatch type of bulged 25-mer matches

was defined and included in the analysis. Bulges in hybridising oligomers are formed when one or more nucleotides remain unpaired [104]. Bulged 25-mer matches are then matches between an oligomer of length 25 and an oligomer of length 26 in which the longer one of the pairing strands contains a bulge of exactly one nucleotide, tolerating mismatches at the centre position. The first and last positions were not included in this analysis as they are not real bulges, but dangling ends. They are already included by the definition of short 25-mer matches. The definitions of the four match types are exclusive, i.e., the described sets are disjoint.

A list of all 25-mers contained in both the probe DNA and the amplified target reference was generated. To each 25-mer, its genomic position and affiliation either to a wafer or amplicon were assigned. The list was then sorted to obtain a lexicographically ordered 25-mer list, allowing the mismatches as defined above. Finally, the sorted list was linearly traversed to report all 25-mer occurring more than once, restricted to matches only between tiled and target DNA 25-mers. The bulged 25-mer data was processed in a way that only match pairs with distance of at least 25 bp in the genome were included in statistics and further analyses to exclude a large amount of bulged 25-mer matches rising from poly(N)-regions.

In total, 5,160,864 positions were annotated as repetitive, making up 5.16 % of all positions used for tiling (cf. Table B.2). False positives are most likely to be observed, and of consequence, at positions where the count of the nucleotide at the centre position of the repetitive 25-mers exceeds the count of matches supporting the reference nucleotide. These so-called dominating 25-mer positions were identified as a subset of the positions with repetitive 25-mers (cf. B.3).

Sets of Known Polymorphisms

Across all 19 non-reference varieties, 3,130 fragments in total of average length of 557 bp were sampled by PCR and dideoxy sequencing from the tiled regions [140]. From these sequences, a gold standard set of polymorphisms (GSP) was compiled. The data set comprises 1,743,128 sequenced nucleotides with 14,530 polymorphic sites across all 19 varieties, thereof 9,414 SNPs and 5,116 indel polymorphisms (cf. Table B.10). This set was used for training and quality assessment of the SNP calling and PR algorithm (Section 2.2.3 and 2.2.4).

To benefit from the availability of a second sequenced genome of the *O. sativa* species, the genome of the *indica* variety 93-11 [223] was used to identify polymorphisms in 93-11 with respect to the Nipponbare reference genome. This variety, which is the closest relative to IR64-21 in the set of the five varieties that were used both for array-based resequencing and in the study by Caicedo et al. [37], was not included in the set of resequenced varieties. The two genome sequences were aligned using the aligment programme NUCmer from the MUMMER software package [110]. In total, 325,420,808 nucleotides (85.2 %) of the whole genome sequence and 91,265,021 nucleotides (91.2 %) of the tiled sequence were aligned. From this alignment 717,695 polymorphic sites were obtained including 436,709 SNPs and 280,986 indel polymorphisms in the tiled sequence (cf. Table B.10). This set of polymorphisms, hereafter called 93-11P, was used as an input for SNP calling, primer PR predictions and as evaluation set for high quality (HQ) PR predictions.

2.2.3. A Machine Learning Method for SNP Identification

A two-layered approach based on Support Vector Machines (SVMs) [177, 205] was applied to predict SNPs from the hybridisation data similar to the approach used in the *A. thaliana* resequencing project [41]. In a first step, SVMs were trained using information comprising the array data, sequence characteristics and repetitiveness of the genome based on the results of the annotation of repetitive 25-mers. Since the hybridisation had been normalised beforehand, machines could be trained across all varieties instead of using a separate machine for each variety as done for *A. thaliana* (different from the approach described in [41]). After genome-wide SNP predictions had been made independently for each variety in the first step, a second layer of SVMs were trained. They were able to integrate information across varieties, as they were provided with results from layer 1 for all varieties as input. Specifically, all positions were re-examined for which a SNP had been predicted with the layer 1 SVM in at least one other variety. The trained SVMs of the second layer were applied for final genome-wide predictions.

Each layer was divided into several sub tasks. First, input vectors for positions that had passed certain filter criteria were generated. After model selection in a cross-validation procedure, SVMs were re-trained with the optimal model parameters to obtain predictions for the filtered positions across all varieties. Finally, each prediction was assigned a confidence value reflecting the posterior likelihood of a true SNP prediction.

To train the SVMs and to evaluate the performance of the classifier, the data set of known polymorphic sites described in the Section 2.2.2 was utilised.

Layer 1 SVM

Filter Criteria Applying a filter prior to the training step was aimed to increase the fraction of true polymorphic sites. This led to a more balanced set of training examples, which was less challenging for accurate discrimination. Indeed, a large fraction of the non-polymorphic sites could be discarded using the following filter criteria. This resulted in a significantly smaller data set, which also reduced computational time in both training and prediction. Specifically, positions were excluded that were identical to the reference with high probability as well as positions where the corresponding array data gave inconsistent information on the called base.

The first criterion was met for a given position p in the target variety t if the raw base call $B_t^+(p)$ and $B_t^-(p)$ of the forward and reverse strand were identical, but different from the base $RS(p)$ of the reference sequence. Secondly, the reference raw base calls of both strands $B_{ref}^+(p)$ and $B_{ref}^-(p)$ had to be consistent to each other and to $RS(p)$. To discard regions of amplicon failures, positions were also excluded if their hybridisation quality scores $\overline{QS}_t(p)$ were less than or equal to 5, averaged over a 100 bp window.

Taking these criteria together, all positions P that passed the filter can be described as the set:

$$
\begin{aligned}
P &= \cup_t \, p_t \\
\text{where } p_t &= \{p | B_t^+(p) = B_t^-(p) \neq RS(p) \wedge B_{ref}^+(p) = B_{ref}^-(p) = RS(p) \\
&\quad \wedge \overline{QS}_t(p) > 5\} \\
&\quad \text{and } t \text{ a given target variety.}
\end{aligned}
$$

Input Data Generation To be able to train SVMs on the given data set, input vectors $\boldsymbol{x}_p^1$ for each position p that had passed the filter in layer 1 were generated. Each $\boldsymbol{x}_p^1$ included measurements at this position and neighbouring positions within ± 4 bp around p: Both maximal intensities I_{max} of each of two quartets and averages of the non-maximal intensities I_{sec} of each of the quartets at each position in the 9 bp window were included. A window size of 9 bp was chosen, as the machine learning method was more accurate for SNPs separated by 7 to 30 bp according to observations for SNP detection in *Arabidopsis thaliana* [41].

Moreover, ratios of the maximal intensities at p and its neighbouring positions (Q_1) and ratios of the maximal intensities at p of the target and the reference variety (Q_2) were added. The usage of these quotients as input features was motivated by the shape of a typical SNP signal reflected in the intensity pattern (cf. Figure A.1). $\boldsymbol{x}_p^1$ also contains sequence characteristics such as mismatches M between raw base calls and the reference sequence within the window, the reference base RS, frequencies f of each base (A, C, G, T) within the 25-mer and the sequence entropy H of the 25-mer. Furthermore, the results from the 25-mer analysis (cf. Section 2.2.2) were used to include occurrence counts k of repetitive 25-mers at p.

As training was conducted variety-independently, information on the variety of which the position was taken was included, denoted as v. Table B.5 describes all inputs in more detail.

The input vectors were normalised on the training set to mean 0 and standard deviation 1 per input dimension. Additionally, a normalisation per input example was applied using $\frac{\boldsymbol{x}_p^1}{||\boldsymbol{x}_p^1||}$ for all positions p. The normalised input vectors were then used to train SVMs with an RBF kernel [177] using the SHOGUN toolbox [192], which allowed a fast and efficient training.

Model Selection in a Cross-validation Procedure Cross-validation is a procedure in which a given data set is partitioned into subsets to have disjoint sets for training and performance evaluation in permuted order and is also important for model selection. The data set of labelled positions was randomly split into five equally-sized disjoint subsets $S_1, \ldots, S_5$ with respect to an uniform distribution of positives per variety. Training and model selection was performed on five different folds in a nested cross-validation scheme. At each of the five iterations, a different subset S_i served as test set T_i. The set that was used for model selection was then defined as $X_i = \{S_j | j \neq i\}$. Thus, the set X_i at iteration step i consisted of 80 % of all labelled positions, whereas the remaining positions belonged to T_i. For parameter tuning, each set X_i, $i = 1, \ldots, 5$, was in turn partitioned into five subsets S_{ij}, $j = 1, \ldots, 5$, four of them served as training set X_{ij} at each iteration step. The prediction and evaluation was done on the omitted subset for each model k, i.e., for each combination of the model parameters. The parameters to be tuned comprised the width σ of the RBF kernel ($\sigma = [10^{-1}, 10^{-0.5}, 10^0, 10^{0.5}, 10^1]$) and the penalty for using slack variables ($\gamma = [10^{-2}, 10^{-1}, 10^0, 10^1]$). Thus, 20 models were tested on each subset S_{ij}.

To find the best model, a measurement was applied based on the idea of the area under the curve (AUC) of the receiver operating characteristic (ROC). ROC curves and their AUCs are commonly used for the performance assessment of a binary classifier. In a ROC curve, the false positive rate (sensitivity) is plotted against the true positive rate ($1 -$ specificity). The higher the AUC value is, the more accurate the classifier. Here, the aim was to maximise the area limited to a maximal ratio of true positives to false positives, i.e., the AUC for a set of classifiers with a limited FDR. Therefore, the number of true positives as a function of the number of false positives was determined. The area a_{ikj} between this curve and the line showing one false positive at five true positives was then calculated corresponding to FDRs below $\frac{1}{6}$. The optimal model for each split was determined by the model k for which the

average of the areas a_{ikj}, $j = 1, \ldots, 5$ was maximal. Thereafter, an SVM on the whole set X_i was trained with its best model parameters and predictions were calculated for the hold-out set T_i.

Prediction and Output Transformation The prediction for each position that passed the filter was computed by the SVM of layer 1 for which this position was not used in training or parameter selection. For all other positions, any of the trained SVM could be used and thus one of the five layer 1 SVMs was randomly chosen to predict their labels.

As the outputs resulted from five different SVMs, the predictions were not directly comparable. To be able to employ these predictions as an input to the layer 2 SVMs and for further analyses, each prediction was transformed into a posterior probability for being a true positive. For this purpose, the conditional likelihood of the true label being positive for a given output value was estimated.

The aim was to learn a piecewise linear function on the corresponding test set. The SVM predictions were divided into 40 quantiles of which each was represented by a supporting point $x(q)$, $q = 1, \ldots, 40$, to ensure a good estimation of the piecewise linear function. The probability $y(q)$ of being a true positive was estimated as

$$y(q) = \frac{n_{TP}(q)}{n(q)}$$

where $n_{TP}(q)$ was the number of true positives, i.e., the number of known SNP positions, with prediction values V in the range $x(q) \leq V \leq x(q+1)$ and $n(q)$ the number of all labelled positions with prediction values in that range. Analogously, estimations were made for definition of a cumulative y_c by omitting the upper bound. To obtain smooth and monotonically increasing estimates, a technique described in [191] was used.

Each prediction value V was transformed into a confidence $c(V)$ by:

$$c(V) = \begin{cases} y(1) & V \leq x(1) \\ \frac{y(q+1)\cdot(V-x(q))+y(q)\cdot(x(q+1)-V)}{x(q+1)-x(q)} & x(q) \leq V \leq x(q+1) \\ y(40) & V \geq x(40) \end{cases}$$

The transformation function for the cumulative confidence C worked in an analogous manner.

Layer 2 SVM

Filter Criteria In the second step, one additional condition was used by exploiting information from layer 1 predictions across all varieties. Only positions that had confidence values greater than some threshold θ_t in at least one other variety t were used for training the SVMs of layer 2. This threshold was determined on all test sets per variety by taking the confidence value θ_t for which n_t examples had confidence values above θ_t, where n_t was the sum of all positively labelled positions in t that passed the filter. A relaxed filter for layer 2 was further used to take positions into account that were likely to be polymorphic when observed in at least one variety, but which did not have raw base calls identical on both strands in the variety and the reference, respectively. Furthermore, the raw base calls of at least one strand had to be different to the reference sequence. For the reference raw base calls, an inconsistency to the reference sequence in one of the strands was allowed. Again, position were discarded with mean hybridisation quality score $\overline{QS}_t(p)$ less than or equal to 5.

All positions P that passed filter 2 were described as:

$$
\begin{aligned}
P &= \cup_t \; p_t \\
\text{where } p_t &= \{p| \bigvee_{s=1}^{19} [[c(V_{s,p}) > \theta_s]] \wedge (B_t^+(p) \neq RS(p) \vee B_t^-(p) \neq RS(p)) \\
&\quad \wedge (B_{ref}^+(p) = RS(p) \vee B_{ref}^-(p) = RS(p)) \wedge \overline{QS}_t(p) > 5\} \\
&\quad \text{and } t \text{ a given target variety.}
\end{aligned}
$$

where [[ ]] denoted the indicator function.

From Input Generation to Predictions The input vector $\boldsymbol{x}_p^1$ from layer 1 was extended to the input vector $\boldsymbol{x}_p^2$ for layer 2 by a binary vector b. This vector had ones at positions for which the corresponding confidence values were above the threshold θ_t. In addition, the confidence values of all varieties were included. To be able to connect a position to its target variety t during learning, the variety information was encoded by vector of length $19^2 = 361$ with b at the 19 positions corresponding to variety t and zeros elsewhere. Confidence values were encoded in the same way. The variety information v of a position was omitted. As an additional feature, knowledge about variation between the genome sequence of the *ssp. japonica* variety Nipponbare and the *ssp. indica* variety 93-11 (*ind*) was included from the 93-11P set (cf. Section 2.2.2), facilitating the detection of SNPs at those observed polymorphic positions.

After normalisation of the input vectors, the same model selection procedure was applied as for layer 1. Predictions were made for all positions which passed filter 2 by exactly one layer 2 SVM.

Predictions for All Arrayed Positions

After having trained the five layer 1 SVMs and five layer 2 SVMs based on the data set of known SNPs, these trained machines were the applied to make predictions for the tiled regions of the whole genome, including unlabelled positions that were interrogated on the hybridisation resequencing arrays. As theses positions had not been employed either for training and evaluation, any of the five SVMs could be used for prediction.

For all of the 19 non-reference varieties, a layer 1 SVM was chosen at random to make a prediction for each unlabelled position that passed filter 1. Predictions were also made for positions across all varieties that met the filter criteria in at least one other variety. Afterwards, the outputs were transformed into confidence values applying the transformation function specific to the layer 1 machine used. By this, the predictions were usable for the second prediction layer.

For all unlabelled positions that were predicted by layer 1 SVMs and passed the second filter, predictions were made by all five layer 2 SVMs. The SVM outputs were again transformed into confidences with the corresponding piecewise linear function. The resulting five values for each position were averaged to assign a final probability of being a SNP position.

Base Calling

As the output of the machine learning method only comprised the probability $C(p)$ for being a true SNP at a given position p, the corresponding observed base $B(p)$ was inferred from the hybridisation data by applying the base calling strategy described in Algorithm 2.1.

Algorithm 2.1 The algorithm for base calling

procedure BASE CALLING
  if $B^+(p) \neq B^-(p) \wedge B^+(p) \neq RS(p) \wedge B^-(p) \neq RS(p)$ **then** # case 1
    $B(p) \leftarrow$ `N`
  else if $B^+(p) = B^-(p) = RS(p)$ **then** # case 2
    $B(p) \leftarrow RS(p)$
  else if $B^+(p) = B^-(p) \neq RS(p) \wedge C(p) \geq 0.855$ **then** # case 3
    $B(p) \leftarrow B^{+/-}(p)$
  else if $B^+(p) = B^-(p) \neq RS(p) \wedge C(p) < 0.855$ **then** # case 4
    $B(p) \leftarrow$ `N`
  else if $B^+(p) = RS(p) \wedge B^-(p) \neq RS(p) \wedge C(p) \geq 0.855$ **then** # case 5
    $B(p) \leftarrow B^-(p)$
  else if $B^+(p) = RS(p) \wedge B^-(p) \neq RS(p) \wedge C(p) < 0.855$ **then** # case 6
    $B(p) \leftarrow$ `N`
  else if $B^-(p) = RS(p) \wedge B^+(p) \neq RS(p) \wedge C(p) \geq 0.855$ **then** # case 7
    $B(p) \leftarrow B^+(p)$
  else if $B^-(p) = RS(p) \wedge B^+(p) \neq RS(p) \wedge C(p) < 0.855$ **then** # case 8
    $B(p) \leftarrow$ `N`
  end if
end procedure

Evaluation

Two measures were used to investigate the accuracy and quality of the SNP detection methods. The false discovery rate (FDR) measured the fraction of spuriously predicted positives relative to all predicted positives, i.e., how often the predictor is wrong when it calls a SNP $\left(\frac{FP}{TP+FP}\right)$. The fraction of true positives and positives (P), i.e., true SNP positions that are recovered by the predictor, was denoted by recall $\left(\frac{TP}{TP+FN}\right)$. Repetitive positions and those with low dideoxy sequencing quality were used for training, but were excluded from performance evaluation.

2.2.4. A Machine Learning Method for Detecting Polymorphic Regions

The applied learning algorithm was a modification of the method proposed for the identification of PRs in resequencing data for the model plant *A. thaliana* [226]. Therefore, this section only presents a summary and major features of the mPPR (margin-based prediction of polymorphic regions) algorithm and highlights differences in the implementation to the *A. thaliana* study. Details of the algorithm can be found in [225, 226].

The mPPR Model and Algorithm

The mPPR algorithm is a label sequence learning approach and extends Hidden Markov Support Vector Machines (HM-SVMs) proposed in [12, 202] (cf. Section 1.2.3). It uses a state model similar to Hidden Markov Models (cf. e.g. [24]), but the determination of the parameter set is realised via discriminative rather than generative training by enforcing a large margin between the correct and any other wrong state sequence (path).

The state model was composed of one state P_U for polymorphic nucleotides, six states modelling decreasing intensities upstream and downstream of a PR (T_1, T_2, T_3 and T_4, T_5, T_6, respectively) and one state C_U for conserved nucleotides. The conserved and polymorphic states were duplicated for repetitive nucleotides to model them separately from unique nucleotides. Thus, the set of states was defined as $S = \{P_U, P_R, C_U, C_R, T_1, T_2, T_3, T_4, T_5, T_6\}$. Transitions $\phi(i, j)$ from state i to j were allowed as indicated in Figure 2.3 (Supplemental Figure S11 in [226]).

A function $f : X \to S^\star$ was learned by training on sequences from the GSP data. This function predicts a path $\boldsymbol{\pi} \in S^\star$ given a sequence of input features $\boldsymbol{x} \in X$ with dimensionality m, where $S^\star$ denotes the Kleene closure to the set of states S and the set of features X. This was done via a $\boldsymbol{\theta}$-parametrized discriminant function $F_{\boldsymbol{\theta}} : X \times S^\star \to \mathbb{R}$ that assigns a score to a pair of input feature $\boldsymbol{x}$ and path $\boldsymbol{\pi} = \pi_1, \ldots, \pi_t$ of length t:

$$F_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\pi}) = \sum_{p=1}^{t} \sum_{j=1}^{m} \sum_{s \in S} [[\pi_p = s]]\, g_{j,s}(x_{j,p}) + \phi(\pi_{p-1}, \pi_p).$$

A scoring function $g_{j,s} : \mathbb{R} \to \mathbb{R}$ modelled as a piecewise linear function is associated for each pair of features j and states $s \in S$. The values of the supporting points of the piecewise linear function $\theta_{j,s,l}$, $l = 1, \ldots, Q$ with Q the number supporting points, together with the transition scores ϕ composed the parameter set $\boldsymbol{\theta}$ that needed to be determined during learning. From $F_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\pi})$, f was calculated by $f(\boldsymbol{x}) = \operatorname{argmax}_{\boldsymbol{\pi} \in S^\star} F_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\pi})$ using dynamic programming.

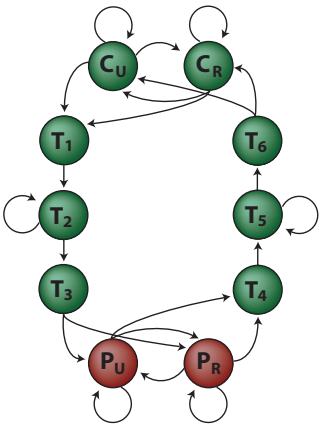


Figure 2.3.: State model of the mPPR algorithm. States are indicated as coloured circles and transitions as arrows. C_U and C_R represent conserved nucleotides (unique and repetitive, respectively) and P_U and P_R model polymorphic nucleotides. The states denoted by $T_1, \ldots, T_6$ model decreasing intensities upstream and downstream of a PR. The illustration was made by Georg Zeller and is taken from [226].

The set of optimal parameters $\boldsymbol{\theta}^*$ was determined on n examples from the GSP set (cf. Section 2.2.2) by solving the following linear programme (LP):

$$\begin{aligned} \underset{\boldsymbol{\theta}}{\text{minimise}} \quad & \frac{1}{n}\sum_{i=1}^{n}\xi^{(i)} + \gamma\ \mathcal{R}(\boldsymbol{\theta}) \\ \text{subject to} \quad & F_\theta(\boldsymbol{x}^{(i)}, \boldsymbol{\pi}^{(i)}) - F_\theta(\boldsymbol{x}^{(i)}, \overline{\boldsymbol{\pi}}) \geq 1 - \xi^{(i)} \quad \forall\ \overline{\boldsymbol{\pi}} \neq \boldsymbol{\pi}^{(i)},\ \forall\ i = 1, \ldots, n \\ & \boldsymbol{\xi} \geq \mathbf{0} \end{aligned}$$

where $\mathcal{R}$ is a linear regularisation term of the form defined by

$$\mathcal{R}(\boldsymbol{\theta}) = |\boldsymbol{\theta}| + \sum_{j=1}^{m}\sum_{s\in\mathcal{S}}\sum_{l=1}^{Q-1}|\theta_{j,s,l} - \theta_{j,s,l+1}|\,.$$

By using slack variables $\boldsymbol{\xi}$, a soft-margin was introduced allowing some error in the predictions. To keep the absolute values small and obtain smooth piecewise linear functions, the objective of the LP was augmented by the regularisation term $\mathcal{R}$. The effect of the regularisation term in the objective was controlled by the regularisation strength γ.

Because the number of wrong paths grows exponentially with the length of the path, it was impossible to directly solve the LP, but the optimal solution was determined by using a column generation technique proposed in [12]. By this, new constraints were iteratively added to the active set of constraints that were maximally violated by the decoded path from the intermediate LP. At each iteration, the intermediate LP with the current constraints was solved with the optimisation software CPLEX [92] until convergence to the optimal solution.

Input Data Generation

Each known sequence of the GSP data (cf. Section 2.2.2) provides information about polymorphic and conserved genomic position and can be used to assign either P_U or P_R depending on their repeat annotation (either unique U or repetitive R) to all polymorphic sites and to all sites between two polymorphic labels at a distance of ≤ 18 bp (cf. Figure A.2). Three transition states flanked each boundary of a polymorphic segment, which was extended by 6 bp in each direction; all other sites were assigned to be conserved (either C_U or C_R).

In addition to the seven features used in the *A. thaliana* study (cf. Table B.6, feature 1, 2, 5, 6, 16, 17, 19, 25), 18 other features were derived from the array intensities (cf. Table B.6, feature 1-7), quality scores (feature 8-16), raw base calls and reference sequence (feature 17-24). The repeat annotation (cf. Section 2.2.2) was used to force switches between unique and repetitive states. To increase recall for primer PR prediction, a feature based on the 93-11P data was added for this application (cf. Table B.6, feature 26). All features were standardised before training; mean and standard deviation were estimated on the training set.

Training and Evaluation

Based on the input features and the label sequences, the optimal parameters of the learning function were determined in fivefold cross-validation. The GSP data was therefore randomly split into five subsets, ensuring that overlapping fragments were sampled into the same subset. At each cross-validation permutation, the predictor was trained on three of the subsets, its optimal regularisation parameter γ was selected on the fourth, and its performance was

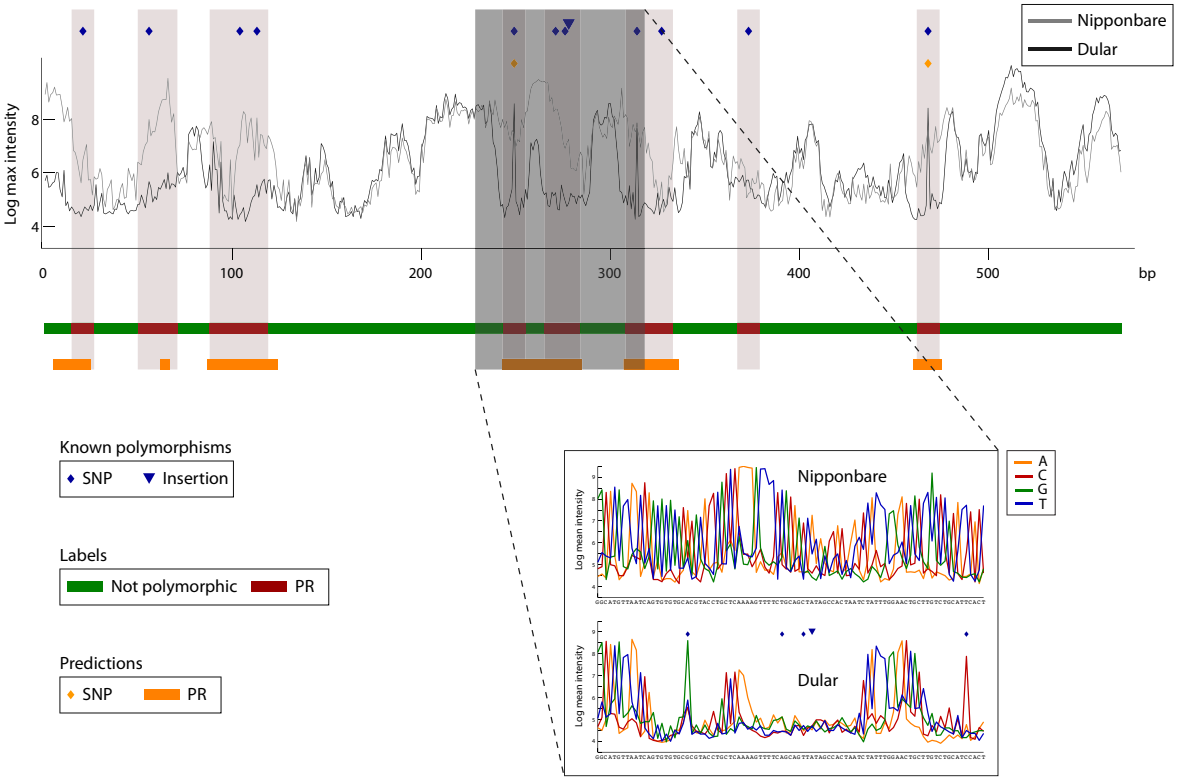


Figure 2.4.: Illustration of the hybridisation pattern, known polymorphisms and PR predictions. Log_2 intensities for the maximally hybridising oligonucleotide at each tiled position for the reference Nipponbare and the *aus* variety Dular (see inlet) are shown for a highly polymorphic region located on chromosome 11 at position 20,142,507 to 20,143,076. Known polymorphisms were retrieved from the sequenced GSP fragment (in blue); predicted SNPs (MBML data) as indicated. The PR labels used as input to the mPPR algorithm are highlighted in light grey. Only two of the twelve polymorphisms (of which eleven are SNPs) were detected in MBML. The enlarged region highlighted in dark grey shows the $\log_2$ intensities for all four bases (A: orange, C: red, G: blue, T: green) averaged over both strands with the identified sequence for both varieties.

assessed on the fifth subset. Each subset contained about 630 fragments, thus there were around 1,900 fragments available for training.

All evaluations were based on the GSP data. No predictions were made for the reference Nipponbare as information on this variety was used in feature generation. Predictions were counted as true positives (TP) if at least 50 % of the prediction overlapped with known PRs, otherwise as false positives (FP). Known PRs with all underlying polymorphisms inclusive in a prediction or those that were predicted as polymorphic in at least 50 % of their length were counted as true discoveries (TD), else as false negatives (FN). For details and visualisation, I refer the reader to Supplemental Figure S2 in [226]. These measurements were used to determine precision $\left(\frac{TP}{TP+FP}\right)$ and recall $\left(\frac{TD}{TD+FN}\right)$. Both known PRs and predicted PRs were excluded if they contained more than 75 % repetitive sites. Furthermore, to obtain HQ PRs, only those PRs, for which one quarter and one fifth of the 100 bp flanking region $\log_2$ intensities and quality scores exceeded a threshold of 6 and 5, respectively, were included to compensate for potential amplification failures. These settings were chosen in a way such that the area under the precision-recall curve was maximised. Additionally, the quality of

the genome-wide predictions was evaluated on 93-11P using the same criteria for assessing precision and recall.

By adding a set of adjustment values to the estimated transition scores, predictions could be obtained with either higher precision (adjustment value > 0) or recall (adjustment value < 0). This strategy allowed to choose the trade-off between precision and recall. The transition scores $\phi(i, i), i \in \{C_U, C_R\}$ of the state model were adjusted by 51 and 61 values uniformly chosen from the intervals $[-3, 2]$ and $[-3, 3]$ for the HQ and primer predictions, respectively, to generate precision-recall curves for all five test sets. These curves were averaged to obtain a precision-recall curve over the subset.

To obtain HQ predictions for all normalised arrayed positions, the smallest adjustment value was chosen for each predictor independently such that the predictor achieved a precision of more than 80 %. Every contiguous fragment was divided into chunks of ≈ 1 kb, ensuring that if the chunk contained a GSP fragment, the respective test prediction was used, otherwise the prediction was randomly chosen from one of the five predictions.

Primer PRs

The factors affecting the success of PCR experiment and subsequent dideoxy sequencing are various. The choice and design of the optimal primer pair flanking the region of interest plays a crucial role. A prerequisite for primer design with directed sequencing is the availability of the genomic sequence at the boundaries of the fragment to be sequenced. Often, the genome of the closest sequenced relative is used without any knowledge of potential polymorphisms, which could disable the functionality of the chosen primers. With the annotation of polymorphic regions, conserved regions with respect to the reference can be defined, and thus the available genome can serve as a template in these regions.

To annotate conserved regions for primer design, the mPPR algorithm was trained in a second experiment. The feature set was augmented by one feature based on the 93-11P data (for details, see Section 2.2 and Tables B.10 and B.6). The training and evaluation were undertaken in the same manner as for the HQ PR predictions described in Section 2.2.4. A set of predictions with varying recall cutoffs in the range of 0.1 to 0.9 was compiled.

To give evidence whether a PCR and sequencing experiment will likely be successful, the probability of observing a conserved region with a minimal length was estimated for each of the nine primer PR sets. The probability was calculated in a sliding window approach by counting the ability to find sufficiently long stretches in conserved regions within a window of a certain size. These events correspond to locations where potential sequencing primer can be designed with high success rate for a PCR and sequencing experiment. The strategy was extended for an evaluation on GSP fragments. Here, a primer of 22 nt in length was randomly selected from the conserved region and checked whether its sequence did include any polymorphism annotated in the GSP set. If the primer contained at least one polymorphism, this event was counted as unsuccessful.

2.2.5. Creating the Set of Non-redundant PRs

When comparing PRs across all varieties, one faces the problem that the boundaries of PRs shared between varieties were not strictly aligned, complicating conclusions from comparisons across or within variety subgroups. To create a set of non-redundant PRs with boundaries across all 19 varieties, an approach based on dynamic programming was developed. It gener-

ates blocks with boundaries aligned among varieties along with a score reflecting the fraction of polymorphic nucleotides per variety. The algorithm was designed in the way that the boundaries were optimally chosen with respect to the original PRs, guaranteeing that the polymorphic degree of each variety changed as little as possible within a block.

The following optimisation problem was solved recursively with the dynamic programming strategy:

$$L_n = \min_{s=1:S} \sum_{v=1}^{19} \sum_{q=n-s+1}^{n} \ell\left(p_{q,v}, \mu_{s,v}\right) + L_{n-s} + \lambda$$

where

$$\mu_{s,v} = \frac{\sum_{q=n-s+1}^{n} p_{q,v}}{n}.$$

The current cost L_n at nucleotide n was set to the cost of the sub-segment of length s, which lies within the segment of maximal length S preceding the current nucleotide n, that showed the minimal cost in terms of the loss function ℓ. S was chosen with respect of a trade-off between running time and reasonable block fragmentation. The loss function was defined as the squared deviation of being polymorphic $p_{q,v}$ at position q in the sub-segment for variety v and the average polymorphic degree in the sub-segment. Formally, the loss function can be formulated as:

$$\ell\left(p_{q,v}, \mu_{s,v}\right) = \left(p_{q,v} - \mu_{s,v}\right)^2$$

Then, the cumulative loss simplifies to:

$$\begin{aligned}\sum_{q=n-s+1}^{n} \left(p_{q,v} - \mu_{s,v}\right)^2 &= \sum_{q=n-s+1}^{n} {p_{q,v}}^2 - 2 \underbrace{\sum_{q=n-s+1}^{n} p_{q,v}}_{s\ \mu_{s,v}} \mu_{s,v} + \underbrace{\sum_{q=n-s+1}^{n} {\mu_{s,v}}^2}_{s\ {\mu_{s,v}}^2} \\ &= \sum_{q=n-s+1}^{n} p_{q,v}^2 - s\ {\mu_{s,v}}^2\end{aligned}$$

facilitating efficient computations.

The switch cost parameter λ allowed to adjust the number of generated blocks; a higher switch cost resulted in fewer blocks.

2.2.6. Identifying Protein Domains Affected by PRs

Based on the set of non-redundant PRs described in Section 2.2.5, protein domains affected by PRs were analysed. Such an analysis can give first hints about the loci that are highly variant in different subgroups, and that potentially relate to distinct traits of a subgroup and single varieties.

Gene models from the RAP and MSU annotations [154, 169] were used for the subsequent analysis. Because the design of the resequencing arrays was based on another genome coordinate system, the IRGSP version 4 pseudomolecules [97], all identified PRs were also initially localised relative to those pseudomolecules and first needed to be transferred to RAP/MSU coordinates. They were mapped relative to the MSU version 6 rice pseudomolecules [154] pursuing the following strategy. As the short lengths of many PRs would not allow accurate mapping, PRs together with 50 bp of flanking sequences on each side were aligned to the MSU rice pseudomolecules using the mapping tool GMAP [219].

PRs were annotated as to protein domains in which they were found. Protein domains from the Pfam database of protein families ([60, 61], `http://pfam.sanger.ac.uk/`) were identified within RAP/MSU rice gene models using the protein sequence search tool InterProScan [154, 224]. All protein domains that reside within regions that were resequenced were noted and protein domains that overlapped with at least one PR were identified.

2.3. Results and Discussion

2.3.1. SNP Calling

The SNP calling algorithm was trained and applied to obtain genome-wide SNP predictions for each variety as described in Section 2.2.3. Across all varieties, 1,343,270 SNPs at non-repetitive sites were discovered with this approach, corresponding to 316,373 non-redundant positions (cf. Tables 2.1, B.7 and B.8). In comparison to a model based (MB) SNP calling approach implemented by Perlegen Sciences [80], the ML method was found to be much more sensitive assessed on the GSP set by recovering 20.9 % of all known SNPs at an FDR of 8.3 %, compared to 14.4 % and 9.1 %, respectively, for the MB approach. Together, the two datasets (MBML-union) included 1,824,074 SNPs at 397,348 positions (cf. Table B.8). The intersection of MB and ML predictions contained 761,606 SNPs predictions at 159,879 non-repetitive positions constituting a set of markedly higher quality with an FDR of 2.9 %. This high-quality set was used for subsequent biological analysis undertaken by McNally et al. [140]. Approximately one-fourth of the high-quality MBML SNPs were validated at 97 % accuracy, consistent with the previously estimated FDR of 2.9 % [140]. The genome-wide average of SNPs per kb using the MBML-intersect data was 1.6.

Table 2.1.: Mean number of SNP predictions at non-repetitive sites for varieties by variety group and prediction method. The two performance measures [recall, FDR] are given in % in squared brackets. MBML-union denotes the average union of the ML and MB set, MBML-intersect are the mean number of SNPs detected by both methods, MB only and ML only are the average number of predictions made by one of the methods only. Recall and FDR are not reported (NR) (in brackets) where < 50 SNPs were available for evaluation, because of very low statistical power. Total numbers are means over all varieties.

Subgroup	MBML-union		MBML-intersect		MB only		ML only	
Temp. japonica	14,882	[NR:NR]	2,028	[NR:NR]	11,044	[NR:NR]	1,810	[NR:NR]
Trop. japonica	50,221	[18.4:14.6]	20,012	[7.3:8.5]	12,543	[2.1:34.4]	17,666	[9.0:16.0]
Aromatic	51,817	[NR:NR]	2,022	[NR:NR]	48,747	[NR:NR]	1,048	[NR:NR]
Aus	137,114	[24.9:11.6]	63,054	[12.4:2.1]	28,195	[2.7:25.4]	45,865	[9.8:17.3]
Indica	126,702	[25.3:10.4]	54,903	[11.0:2.5]	29,684	[3.8:25.5]	42,115	[10.5:11.7]
All varieties	91,203	[27.8:12.3]	38,080	[9.7:3.2]	24,040	[10.2:24.7]	29,083	[7.9:14.0]

2.3.2. Training and Performance Evaluation of the mPPR Method

The algorithm described in Section 2.2.4 was trained on 60 % of the GSP data, the hyper-parameter γ was tuned on 20 %, and 20 % were used for testing in a fivefold cross-validation strategy to ensure independent predictions for all GSP sequences. A set of precision-recall pairs was assessed by tuning the internal adjustment parameter of the algorithm (for details, see Section 2.2.4 and [226]).

In a first step, the algorithm was trained using the features 1 to 25 described in Table B.6. The resulting precision-recall curves both with and without the usage of the quality filter (cf. Section 2.2.4) are shown in Figure 2.5. Because the gain in terms of both precision and recall was largest, the setting at a precision of $\geq$ 80 % filtering fragments with low hybridisation quality was selected to generate predictions for the GSP data. For this selection, 25.7 % of the known PRs in the GSP fragments could be recovered across all varieties. Considering each variety separately, precision values were mostly distributed around 80 % when reported (cf. Table B.9). The *indica* variety Aswina with precision of 90.5 % and Zhenshan 97B with only 72.5 % precision were examples for extreme outliers. In terms of recall, the variation between varieties was even higher, ranging from 12.3 % to 39.4 %. One reason for these performance variations might be that the hybridisation quality across the varieties also showed high variation due to different success rates of the amplification and hybridisation experiments.

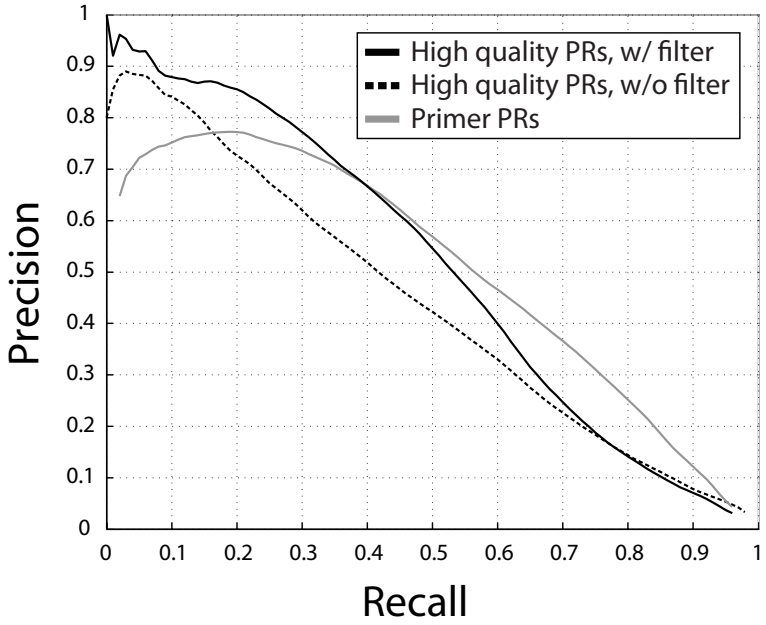


Figure 2.5.: Precision-recall curves for HQ and primer PRs. The relationship between precision and recall with an overlap criteria of 50 % for the three experiments, PR predictions without filter, PR predictions with filter (HQ PRs), and primer PR predictions, is shown in the precision-recall curves. The curves were averaged over cross-validation subsets.

The mPPR algorithm approximated the location of polymorphisms. Nevertheless, as the underlying polymorphisms were known from the GSP data, i.e., whether a site harbours a SNP or indel, the content of the PRs predicted on the GSP data could be analysed. In total, 910 regions were identified as polymorphic, of which the majority, 533 PRs (58.6 %), contained only a single SNP. PRs with more than one SNP were detected in 88 regions (9.7 %), 51 (5.6 %) contained single indel polymorphisms, and 20 (2.2 %) of the regions were more complex, i.e., consisted of a combination of SNPs and indels. Regarding recall, 26.3 % of the SNPs, 29.4 % of the deleted bases and 17.9 % of the insertions were predicted to be located in a PR.

In addition to the evaluation on the GSP data, the quality of the genome-wide predictions was assessed on the second evaluation data set 93-11P. As expected, prediction accuracy increased from distantly related *japonica* varieties to varieties in the same subpopulation (19.8 % to 53.8 % in precision, respectively). The full evaluation is presented in Table B.11.

2.3.3. Comparison to MBML Set

The PR predictions were compared against the MBML-union set (cf. Section 2.3.1) for GSP fragments that contained SNPs. The results are presented in Table 2.2. Using the mPPR algorithm, 26.3 % of the SNPs could be recovered, which contrasts to the low recall rate of 8.5 % for the MBML SNP calls. Considering clustered SNPs, i.e., SNPs with a distance of $\leq$ 18 bp to nearest polymorphism, the difference between the two methods was even more apparent: 22.7 % of the clustered SNPs were identified by the mPPR algorithm, whereas the SNP calling method failed completely. Different from *A. thaliana* study, the recall rate for isolated SNPs ($>$ 18 bp away from the nearest polymorphism) was even higher (30.1 %) than for clustered SNPs in the PR predictions and twice as high for SNPs in the MBML set. Thus, the PR predictions do not only complement the SNPs calls for SNPs found in polymorphic clusters, but also for isolated SNPs. Furthermore, 20.9 % of the SNPs in the GSP fragments were located exclusively in PRs predicted by mPPR, contractive to 3.1 % SNPs that were present in the MBML data only.

The benefit of the PR predictions could also be observed when determining the SNP recall in dependency to the distance to the closest polymorphism (cf. Figure 2.6). For all distance sets, SNP recall rates between $\approx$ 20 and $\approx$ 30 % in mPPR predicted PRs could be observed with little increase towards longer distances. For the MBML data, SNP calling failed for SNPs that have polymorphisms in vicinity (up to $\approx$ 20 bp). Isolated SNPs could be more reliably identified. Nevertheless, SNP prediction with the mPPR method was markedly superior.

2.3.4. Genome-wide PR Predictions

To obtain genome-wide predictions for all 19 non-reference varieties, the HM-SVMs trained on the GSP data with the same settings used for evaluation to obtain HQ predictions were used. Per variety, between 65,024 and 203,110 PRs were detected, comprising between 1.7 % and 5.1 % of the queried genome sequence. Based on these predictions, further analyses were conducted, which are presented and discussed in the following sections. An example of a putative long deletion on chromosome 12 discovered by the PR annotation is shown in Figure 2.7.

To provide sets of predicted PRs for primer design, the HM-SVMs trained for primer PRs (cf. Section 2.2.4) were used. With these settings, higher recall rates were obtained at lower

Table 2.2.: Recall by polymorphism and sequence type. For the PR predictions (precision ≈ 80 %), the percentage of SNPs within predicted PRs is given. The percentage of SNPs exclusively predicted by the mPPR algorithm (i.e., absent from MBML) are indicated in parentheses. For MBML (precision ≈ 90 %) the percentage of SNPs with correct position and allele is given. SNPs that are ≤ 18 bp away from the closest polymorphisms are annotated as clustered, else as isolated. The assignment to coding, UTR + intron, and intergenic was based on the RAP 2 annotation [168]. Sample sizes are given in brackets.

	Coding		UTR + intron		Intergenic		All	
SNPs	PRs	MBML	PRs	MBML	PRs	MBML	PRs	MBML
Clustered	17.7 (17.5) [513]	1.4	19.2 (18.7) [1,074]	1.1	24.7 (24.1) [3,231]	1.0	22.7 (22.2) [4,818]	1.1
Isolated	31.5 (16.0) [746]	22.9	29.2 (19.2) [1,219]	16.5	30.1 (20.8) [2,593]	14.3	30.1 (19.6) [4,558]	16.3
All	25.9 (16.6) [1,259]	14.1	24.5 (19.0) [2,293]	9.3	27.1 (22.6) [5,824]	7.0	26.3 (20.9) [9,376]	8.5

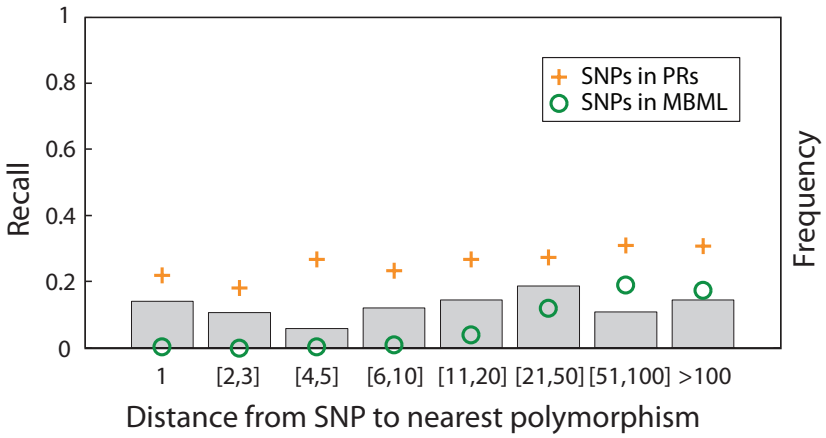


Figure 2.6.: Dependency of SNP recall on distance between polymorphisms by detection method. The frequency of SNPs in each distance category, which was derived by partitioning the SNPs according to their distance to the nearest polymorphism, is illustrated as grey bars. The recall rates per category are shown for MBML SNPs (precision ≈ 90 %) as circles, and inclusion within predicted PRs (precision ≈ 80 %) as crosses.

precision values, in return to gain precision for low recalls (cf. Figure 2.5). Genome-wide predictions were conducted for varying recall rates (0.1 to 0.9), resulting in 83,132 to 335,480 PRs on average across the varieties, which cover between 1.3 % and 68.3 % of the queried genome sequence for the different recall cutoffs (cf. Table B.12). For each of the nine primer PR sets, the probability of a successful PCR and sequencing experiment was estimated by averaging the success within a window of varying window sizes (cf. Figure A.3 (a)). As expected, the estimated success rate decreased with increased recall, since it was then harder to find long enough stretches of conserved regions. With the extended strategy of checking against the GSP fragments, the success rate decreased in general (cf. Figure A.3 (b)), which may be affected by missed PR (false negatives) in the genome-wide set. The differences between varying window size were more apparent because of the larger variation of fragment lengths with average length of 557 bp in the GSP set than in the genome-wide set.

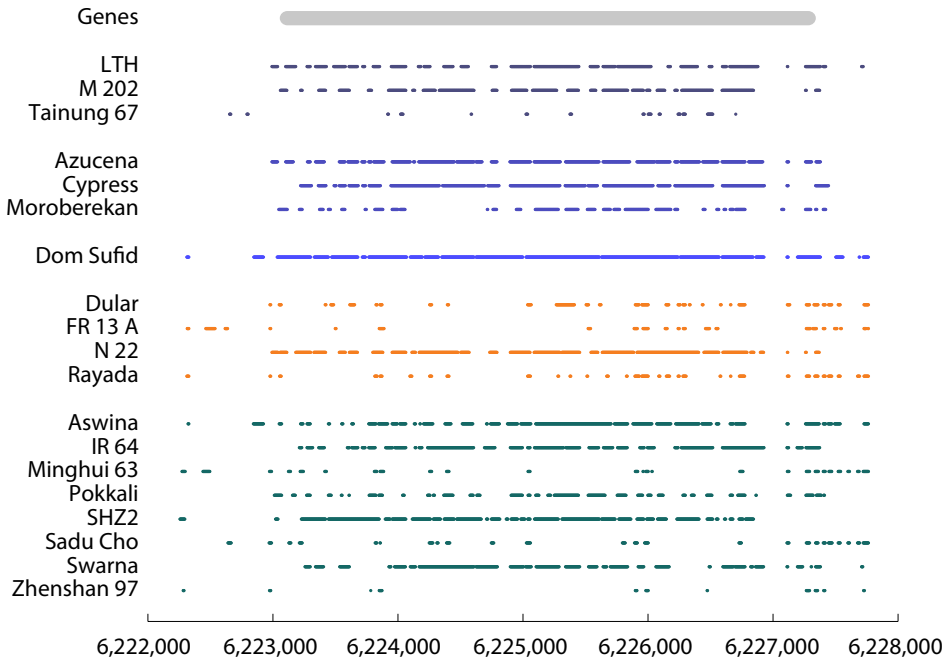


Figure 2.7.: Example for a polymorphic region spanning a sequence of ≈ 5,000 bp on chromosome 12, putatively a long deletion. PRs are indicated as coloured bars for each variety. Within this region, a gene is located between position 6,223,057 and 6,227,295 in IRGSP coordinates [97] and 6,223,134 and 6,226,115 in MSU coordinates [68], which is annotated as Os12g0217300 and LOC_Os12g11500 by the RAP [168, 169] and MSU annotation [154], respectively. As for the MSU annotation, this gene putatively functions as a disease resistance protein SlVe1 precursor.

2.3.5. Comparison to the Arabidopsis thaliana Study

Compared on the level of more than 80 % precision and using an overlap criteria of 50 %, 26 % of the known PRs in the GSP fragments could be recovered, contrasting markedly to the recall rate of 87 % that was obtained in the *A. thaliana* study [41]. By determining the averaged intensities for regions between polymorphisms at a distance less than 26 bp to each other, the characteristic pattern of reduced intensities for features between adjacent polymorphisms could be observed in the *A. thaliana* study (cf. Figure A.2 (b)). Conducting the same analysis on the GSP data for rice, such an obvious pattern could not be identified. The general shape was flatter, and intensity curves for polymorphisms with short distances were on a higher level (cf. Figure A.2 (a)).

As numerous factors differed between the *O. sativa* and *A. thaliana* array-based resequencing studies, potential reasons for the lower data quality and thus weaker performance can only be guessed. The different experimental approaches for amplification of the target DNA was certainly one factor that affected data quality. For rice, the target DNA were amplified by LR-PCR with primers complementary to the reference sequence, implicating potential amplicon failures if the primer complementarity was compromised due to variation in the target sequence. In contrast, whole genome DNA amplification was possible and was applied for generating the *A. thaliana* data, which allowed unbiased sampling. Different genome compositions (e.g., GC content, content of repetitive sequences) and limitations in optimising experimental conditions probably also contributed to the data quality.

2.3.6. Non-redundant PRs

The dynamic programming approach described in Section 2.2.5 was used to create sets of non-redundant PRs for a maximal segment length $S = 1000$ and switch cost $\lambda = \{5, 10, 20\}$. Increasing the switch cost leads to the generation of less but larger blocks, resulting in 1,473,178, 1,081,391, and 737,893 non-redundant PRs for $\lambda = \{5, 10, 20\}$, respectively. For the analysis in the following Section 2.3.7, the set of non-redundant PRs with switch cost 10 was used to obtain a reasonable number of blocks. Table B.13 lists the number of PRs from this set filtered by score ≥ 0.9 for each variety and for the five major subgroups.

As for SNP discovery, different numbers of PRs were found in rice varieties from the *indica* and *aus* subgroups compared to *japonica* or *aromatic* rice variety subgroups (cf. Table B.13). The average number of PRs identified per variety was correlated with phylogenetic distance of the subgroups from Nipponbare determined on the number of SNPs.

To allow convenient visualisation of the PRs relative to each other and to gene models, genome browsers were created to display the PRs relative to both IRGSP version 4 and MSU version 6 rice pseudomolecules [194]. Therefore, the PRs relative to the IRGSP version 4 sequence were transformed relative to MSU version 6 rice pseudomolecules with the strategy described in Section 2.2.6, successfully aligning 1,081,391 PRs (750,987 with a score ≥ 0.9). The genome browsers may be viewed at the MSU OryzaSNP website (`http://oryzasnp.plantbiology.msu.edu`). Additionally, a number of search tools can be found at this website that allow researchers to conveniently identify PRs based on chromosomal position, to find PRs that distinguish varieties, to discover PRs within specific genes and to obtain sequence flanking particular PRs.

2.3.7. Protein Domains Affected by PRs

PRs found within coding regions of rice genes were classified by coincidence with Pfam protein domains (cf. Tables B.14). Interestingly, many Pfam protein domains in which PRs are enriched are involved in protein binding [32, 33, 47, 55, 103, 108, 115].

Unlike SNP discovery, PR identification is inherently ambiguous about the nature of underlying polymorphisms. PRs may result from two or more closely positioned SNPs or from indels of any size. Although PRs within coding sequence might be assumed to more likely disrupt open reading frames, this cannot be guaranteed without additional sequence information. In fact, the data presented here suggests that PRs are often not disruptive to protein function. The protein domains that are enriched for PRs include numerous repetitive domains (e.g., MORN repeat, Kelch motif, Armadillo repeat, Tetratricopeptide repeat, leucine-rich repeats (LRR)), zinc finger domains and F-box domains. These protein domains are all believed to be involved with protein-protein, protein-nucleic acid or protein-small molecule recognition [32, 33, 47, 55, 108].

Known variations within protein domains of these types are believed to play a role in functional variation within proteins. Most LRR domains are believed to be involved in protein-protein recognition events. LRR domains contain a conserved core, but the total length of the domain can vary between 20 to 30 amino acids. Additionally, the LRR domain itself can be repeated numerous times within a gene [18]. In plants, the expanded NBS-LRR gene family constitutes an important set of pathogen resistance genes which are known to exhibit high sequence polymorphism. Natural populations of *Triticum dicoccoides* (emmer wheat) are known to contain NBS-LRR genes with highly variable LRR regions [184]. Moreover,

a *Hordeum vulgare* (barley) powdery mildew resistance locus shows high sequence polymorphism in its LRR domain [183]. Variation in the sequence of repetitive domains or variation in the number of repetitive domains within a protein can be functionally significant [25]. PRs in repetitive domains could affect domain sequence or even disrupt a domain, which would be equivalent to removing one unit of a repetitive domain.

Zinc finger domains function as substrate binding domains. The conserved portion of these domains are responsible for the general structure of the domain, but the domains also contain regions that can differ in the kinds and numbers of amino acids that are allowable [112, 185]. PRs falling within these domains could alter protein sequence and result in altered binding specificity of the domain.

F-box domains are components of substrate recognition sites and these domains were also found to contain an abundance of PRs. A study of rice F-box genes reports that a portion of the genes in this family exhibits a high rate of sequence variation and is under positive selective pressure [221]. The potentially drastic mutations associated with PRs in F-box domain-containing genes would be consistent with a quickly evolving gene family.

The nature of the proteins that contain the domains and the domains themselves that are enriched in PRs suggests that PRs represent a type of complex natural sequence variation that is biologically important.

2.4. Conclusion

This chapter presented the application of two machine learning methods for the detection of polymorphisms from microarray data across 20 domesticated rice varieties. Across all varieties, 1,343,270 SNPs at 316,373 non-redundant sites were identified with an SVM-based approach. Evaluated on a gold standard set of polymorphisms derived from dideoxy sequencing, 20.9 % of all known SNPs at an FDR of 8.3 % could be recovered. A high-quality set of 159,879 polymorphic sites (2.9 % FDR) was assembled from the intersection of SNPs predicted by the machine learning method and a model based approach. This set was used for subsequent biological analysis, revealing breeding history and relationships among the selected varieties by introgression patterns of shared SNPs [140].

In the second part of the computational analysis of the array data, a trained HM-SVM model identified between 65,000 and 203,000 PRs, which covered between 1.7 % and 5.1 % of the queried genome sequence. Here, a precision of 80 % could be achieved recovering 26 % of the polymorphisms. The predictions complement the SNP set assembled in the OryzaSNP project, as 21 % of the annotated SNPs could be exclusively identified by the PR detection algorithm. Additionally, I showed the benefit of PR predictions to define conserved regions in the genome providing a scaffold for successful primer design in PCR and sequencing experiments. Looking at protein domains disrupted by PRs suggests a high rate of sequence variations in repetitive domains that are involved in recognition of proteins, nucleic acids and other small molecules.

In both analyses, the use and power of machine learning approaches could be shown. Dealing with uncertainties that were difficult to resolve by heuristics and non-adaptive algorithms, as seen for the model-based approach in the SNP analysis, the rather noisy and challenging array data could be successfully analysed to uncover SNPs and PRs at a remarkable accuracy.

Notwithstanding the recent advances in digital sequencing, the here assembled set of sequence variants was the first set that collected polymorphisms for diverse varieties of the world's most important crop plant on a genome-wide scale. Based on the SNP inventory, genotyping markers were selected to design genotyping arrays. With these arrays, 395 diverse rice varieties were genotyped to identify patterns of polymorphisms in these varieties, to investigate population structure and to understand introgression history of domesticated rice [229].

3. rQuant: Modelling Biases for Accurate RNA-seq-based Transcript Quantification

3.1. Introduction

The recent development of NGS technologies and their application to RNA sequencing (RNA-seq) allows to obtain a much more detailed picture of transcriptomes (cf. Section 1.4.2). Studying them under varied conditions, in different tissues or in mutants will lead to a considerably improved understanding of the underlying mechanisms of gene expression, RNA processing and the regulation of these processes. An important prerequisite for such analyses is to be able to accurately determine the full complement of RNA transcripts and to infer their abundance in the cell. The problem of abundance estimation from RNA-seq data is widely known as *transcript quantification*. Compared to gene expression arrays, RNA-seq-based transcript quantification has the advantage to provide digital measurements at a more extensive quantitative range. However, it is limited to measure not absolute but relative transcript abundances within a sample.

Due to various limitations and biases in NGS technologies, transcript quantification is less straightforward than one might naïvely expect. Quantification is relatively easy when only one transcript is present at a gene locus with non-repetitive sequence. Then, the transcript abundance can be directly calculated from the read counts. Multiple isoforms at one gene locus make the inference problem more difficult, as reads can not unambiguously be assigned to their respective transcript in segments that are shared between isoforms. Ambiguity of reads can also occur across gene loci when sequences are repeated at multiple regions in the genome, for example caused by recent gene duplication.

Another challenge in transcript quantification is that the observed read coverage is usually not uniform along the transcript. This non-uniformity is caused by diverse factors. Currently available techniques rely on converting the RNA molecules in the sample into cDNA fragments prior to sequencing. A large portion of the observed distortions arise during cDNA library preparation, depending on the used protocol [28, 148]. Crucial steps are here, for example, priming, fragmentation and size selection (cf. Figure 3.1 (a)). As a result, the reads are non-uniformly distributed along the transcript, influenced by the *length* of the transcript and the *distance* to the transcript boundaries (cf. Figure 3.2 (a)). For example, using reverse transcription before the fragmentation step results in more pronounced biases because priming of complete transcripts over-represents the 5' end of the transcripts (cf. [148] and Figure 3.2 (a)).

Moreover, it has been observed that the read coverage also heavily depends on the *sequence context* of the fragments. In a study by Dohm et al. [53], it has been found that read density in genome sequencing is increased in regions of high GC content. Also, mono-nucleotides as well as di-nucleotides may not appear at the same frequency along the read, in particular at the 5' end of the read (cf. Figure 3.2 (b)). Hansen et al. [77] described a distinctive pattern of the first 13 nucleotides at 5' end of reads and proposed that random hexamer priming causes this sequence bias. Similar observations for small RNA expression profiling were made

in a study by Linsen et al. [124]. Here, they detected preferences of certain small RNAs dependent on the used library protocol, which are potentially caused by RT-PCR biases and ligase preferences.

Both transcript length and sequence biases were also observed in a work by Jiang et al. [100]. Here, they used controls of known exogenous sequences with fixed concentrations to describe these biases and to evaluate the general performance of RNA-seq. Based on their analysis, they suggested to use spike-in controls in RNA-seq experiments to be able to estimate biases and to normalise for these effects.

Biases are also induced by data processing, for instance, when aligning the sequence reads to a reference genome (cf. Figure 3.1 (b)). Depending on how well the mapping method can align reads that span splice site junctions, the read coverage in proximity of splice junctions typically drops compared to other exonic regions [102].

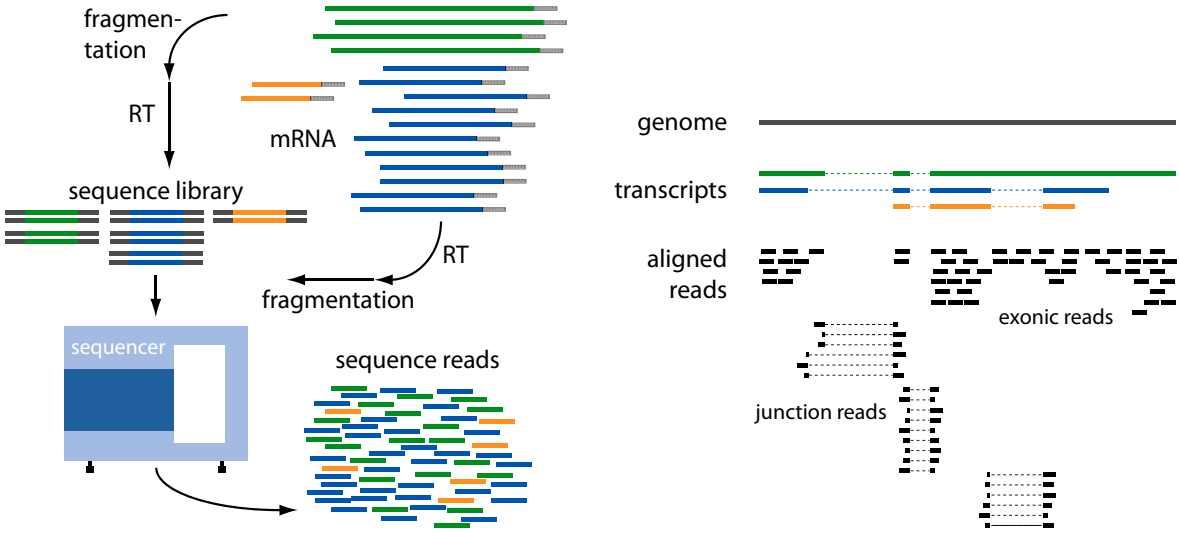
For accurate quantification of RNA transcripts, it appears essential to take the contribution of such biases and other technical limitations into account. This chapter summarises work on the quantification programme rQuant that implements this idea by simultaneously estimating the effect of biases as well as the abundance of RNA transcripts based on mathematical programming. Following a review of related work in Section 3.1.1, I first describe the methodical details of rQuant in Section 3.2. This includes the definition of the developed optimisation problem for quantification and read density estimation and the algorithm to solve this optimisation problem. Moreover, I show how rQuant can be extended to exploit information from paired-read data and from several experimental conditions. In Section 3.3, I present results from the application of rQuant to different RNA-seq data sets. Quantifying transcripts from artificial data sets demonstrates the accuracy of rQuant in an evaluation for diverse settings and a comparison against other popular quantification tools. Finally, I show the value of rQuant on experimental data sets with a comparison to biochemical quantification measurements.

3.1.1. Related Work

Since the advent of RNA-seq in 2008, a set of computational techniques has been developed to quantitatively analyse read data. These techniques are often based on similar methodical concepts. rQuant was the first approach that has addressed to model biases during quantification by estimating positional and transcript length biases. It has been later extended to account for biases induced by sequence content. One key feature and difference of rQuant compared to other quantification programmes is that rQuant uses positional read coverage rather than read counts within segments defined by the exon structure of the isoforms. Moreover, abundance estimation by rQuant is based on solving an optimisation problem using ideas from the lasso approach (cf. Section 1.2.2 and [198]).

This review does not aim to be complete, but summarises key ideas and used techniques for RNA-seq based transcript quantification. The first approach to RNA-seq-based quantification was described in a study by Mortazavi et al. [148] implemented in the software package ERANGE. Here, abundances are inferred by simply counting reads per exon and normalising this count by the total amount of reads in the sample and the length of the exon. By this, the authors defined a unit for quantification, RPKM, which is the number of reads per kilobase of exon model per million mapped reads.

The majority of more sophisticated models uses techniques from Bayesian inference. Many of them assumed uniformity of the read distribution in their first implementation and were

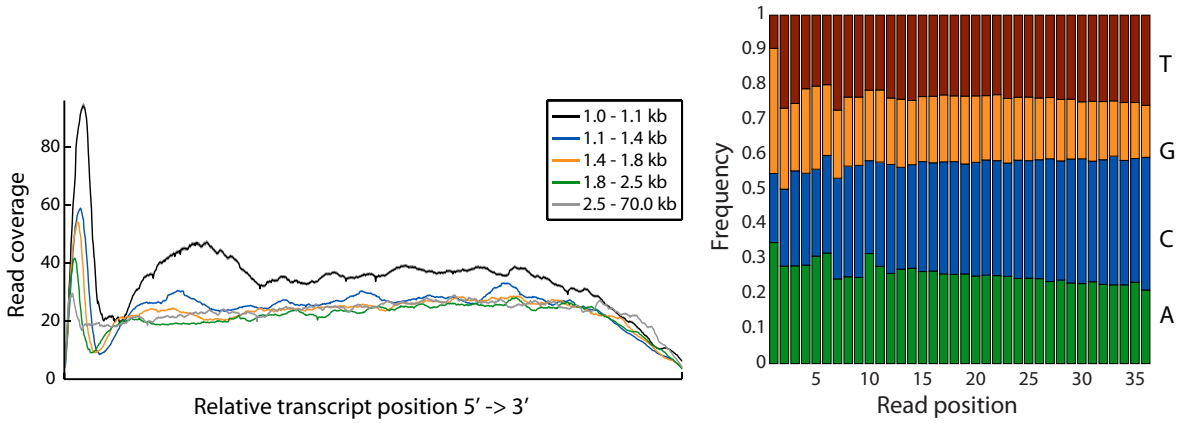


(a) RNA-seq library preparation and sequencing.

(b) RNA-seq alignment.

Figure 3.1.: RNA-seq workflow. (a) The basic steps of RNA-seq library preparation and sequencing for Illumina's sequencing machines is exemplified for transcripts of three isoforms coloured in orange, green and blue. Usually, poly-adenylated mRNA is isolated from the sample, which undergoes reverse transcription (RT) and fragmentation (in both orders). The molecules can be fragmented either chemically by an enzyme or physically sheared, for example, by sonication. Sequencing adaptors are ligated to both ends of the fragments. The range of fragments lengths is narrowed to the required length by size selection on a gel. The resulting cDNA library is then used for sequencing, generating millions of short sequence reads. (b) Typically, the sequence reads are aligned to a reference genome, given rise to read coverage measurements per genomic position. Reads that fully cover exons are denoted as exonic reads. Junction reads are reads that cover splice site junctions and are more difficult to align because of potentially long introns between two splice sites. Specialised alignment programmes such as PALMapper [98] or TopHat [199] have been developed to address the mapping of RNA-seq reads.

extended to consider biases at a later date. The programme rSeq by Jiang and Wong [99] estimates transcript abundances by determining the maximum likelihood estimate (MLE) of observing reads drawn from a Poisson distribution within segments. Here, the expected number of read counts of the Poisson model is a linear function of the transcript abundances. For the single-isoform case, the MLE is identical to computing the RPKM. The same authors suggested in a second paper [122] to calculate the expected number of read counts separately for each isoform by using a non-linear model that considers the sequence content as well gene expression level. Similar methods are Solas [170], which calculates the MLE of observing reads drawn form a multinomial distribution on exon level, and an approach by Howard and Heber [86], which assumes bionomial-distributed read counts augmented by a read density model. Another Bayesian technique for quantification is implemented in the tool RSEM and uses a Bayesian network that takes position-dependent read errors and transcript length biases into account [118, 119]. The programme Cufflinks uses a probability model based on fragments (paired-end reads) [200], where abundances are estimated by maximum *a posteriori* (MAP) and which was recently extended by models for positional and sequence bias estimation [171]. In addition, several programmes exploit information from paired-end reads, for example the distribution of insert sizes (IsoEM [151]). Katz et al. [105] developed a tool called MISO, which uses similar ideas as rSeq [99], but considers pair-end reads. The method NSMAP differs from



(a) Transcript profiles.

(b) Read content.

Figure 3.2.: Biases for the *C. elegans* SRX001872 data set [79]. The cDNA library was generated by reverse transcription with random priming prior to fragmentation. (a) Normalised read coverage with respect to the relative transcript position is shown grouped by five different transcript length bins (see inlet). Read coverage is peaked at the 5' end of the transcripts. Transcript profiles are similar in shape, but different in magnitude for different transcript lengths. (b) Nucleotide frequency in dependency of their positional occurrence in the read is shown. A distinctive pattern of nucleotide frequencies can be observed for the first 10 to 15 nucleotide at the 5' end of the read.

the other Bayesian approaches, as it uses a Laplace prior on the transcript abundances to introduce sparsity in the quantification model [220].

In contrast to the above mentioned methods, the tool FluxCapacitor [147] combines transcript quantification minimising the ℓ_1-deviation of observed and estimated number of reads within exonic segments with estimation of read distribution profiles binned by transcript length and expression levels. A recently published approach, which is similar to the idea of rQuant, is IsoInfer [58]. Here, a quadratic programme with respect to the transcript abundances is solved by minimising the squared deviation of observed and expected number of reads per segment. This was in enhanced in IsoLasso [123] by introducing sparsity through ℓ_1-norm regularisation of the abundance parameters.

3.1.2. Publication Note

rQuant was joint work with Gunnar Rätsch. Regina Bohnert and Gunnar Rätsch conceived and designed the project. Regina Bohnert implemented rQuant and performed experiments for data simulation and evaluation of the method. In addition to Regina Bohnert, Philipp Drewe and Oliver Stegle simulated and provided data for several experimental conditions. Jonas Behr provided code for paired-end analysis. Lisa Smith generated and provided RNA-seq data for *A. lyrata*. Ali Mortazavi provided NanoString data. Some material of this chapter was published in Bohnert et al. [30] and Bohnert and Rätsch [28] and presented at the HiTSeq conference on High-throughput Sequencing Analysis and Algorithms preceding the ISMB/ECCB conference 2009.

3.2. Methods

3.2.1. An Optimisation Problem Formulation for Solving the Transcript Quantification Problem

To infer the abundance of given transcripts from RNA-seq data, the programme rQuant was developed. Ideas from optimisation and machine learning were used to formulate the transcript quantification problem as an optimisation problem by adapting concepts from the lasso approach (cf. Section 1.2.2). Given an annotation of (alternative) transcripts and a set of reads that have been aligned to the reference genome, rQuant inferred the abundance of each annotated transcript by minimising the deviation of the observed from the expected read coverage at each exonic nucleotide and junction and simultaneously estimating the read density in dependency of the nucleotide position in the transcript and the sequence content.

Quantifying Alternative Transcripts

Formally, the objective Ω in the optimisation problem of rQuant was defined as a weighted sum of loss terms for exon and intron coverage deviation ($\mathcal{L}_{exon}$ and $\mathcal{L}_{intron}$) and a term $\mathcal{R}$ that regularised the optimisation variables to reduce or avoid model over-fitting:

$$\begin{aligned}\Omega(\boldsymbol{w}) &= \mathcal{L}_{exon} + \mathcal{L}_{intron} + \mathcal{R}(\boldsymbol{w}) \\ &= \gamma^E \sum_{p=1}^{P} \ell\left(c_p, \sum_{t=1}^{T} w_t \ \delta_{p,t}\right) + \gamma^I \sum_{i=1}^{I} \ell\left(s_i, \sum_{t=1}^{T} w_t \ \eta_{i,t}\right) + \mathcal{R}(\boldsymbol{w})\end{aligned} \tag{3.1}$$

where $\boldsymbol{w} = [w_1, \ldots, w_T] \geq \mathbf{0}$ are the optimisation variables corresponding to the transcript abundance estimates (or weights), T is the number of transcripts, I is the number of introns and P is the number of exonic positions considered (corresponding to one genic locus).

The exon loss measured the deviation of the observed read coverage c_p at position p from the expected read coverage, which is the sum of abundance estimates of transcripts that included p. This information is stored in the exon mask $\delta_{t,p}$ that equals 1 if transcript t is exonic at position p, and 0 otherwise. The exon loss is weighted by the parameter γ^E with respect to the other objective components, which was usually set to 1 in practice.

Moreover, information about alternative spliced site junctions can help in deconvolution of isoforms. This was utilised in the intron loss, penalising the difference of observed intron coverage (or strength) s_i of intron i and the expected intron coverage. The intron strength s_i was measured by the number of reads spanning this intron. Abundance weights were added for transcripts including intron i, implemented by the intron mask $\eta_{i,t}$ that equals 1 if intron i is annotated in transcript t, and 0 otherwise. γ^I is a parameter to weight the intron loss term with respect to the other components of the objective.

If a position p was annotated as repetitive, this position was excluded when estimating the transcript weights because of the ambiguity of reads generated from repetitive regions.

A quadratic loss function was used to penalise larger deviations of the estimated from the observed coverage more than smaller ones:

$$\ell(x, y) = (x - y)^2. \tag{3.2}$$

An additional term $\mathcal{R}(\boldsymbol{w})$ was added to the objective as a cost measure of the transcript weight variables. This cost was realised by an ℓ_1-norm to introduce sparsity [20]:

$$\mathcal{R}(\boldsymbol{w}) = \sum_{t=1}^{T} \gamma_t^w \ |w_t|.$$

Here, γ_t^w is a transcript-specific regularisation weight, which was chosen to be the length of transcript t. As all transcript weights must be less or equal to zero, the absolute sign could be omitted.

Estimating Read Density to Model Biases

Due to the experimental biases mentioned above, the observed read coverage is typically non-uniform over the transcript (cf. Figure 3.2 (a)). To incorporate a bias model in the quantification programme, $\delta_{t,p}$ in the objective (cf. Equation 3.1) was therefore replaced by a more sophisticated read density function. This function $D_{t,p}$ may depend on the relative position in the transcript, the transcript length and the sequence context. $D_{t,p}$ was composed of the product of two sub-functions

$$D_{p,t}(\boldsymbol{\theta}, \boldsymbol{\beta}) = \Theta(\boldsymbol{\theta}, p, t) \ B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta})$$

to model positional and transcript length biases by the function $\Theta(\boldsymbol{\theta}, p, t)$ and sequence biases by the function $B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta})$ in a multiplicative way. The set of optimisation variables was extended by $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$, which were jointly optimised with the abundance variables $\boldsymbol{w}$.

Positional and Transcript Length Bias Model The following model was motivated by the observation that read density on the one hand depends on the distance to the transcript boundaries and is on the other hand a function of the length of the transcript. These findings can be captured in transcript *profiles*, i.e., by relating read coverage to the relative position in the transcript, for different bins of transcript lengths. The profiles were modeled as piecewise linear functions (PLiFs) Θ parametrised by the profile weights $\boldsymbol{\theta}$, the position p and the transcript t:

$$\Theta(\boldsymbol{\theta}, p, t) = \begin{cases} \theta_{1,n_t} & d(p,t) < x_1 \\ \theta_{f,n_t} + \delta_{p,t} \ (\theta_{f+1,n_t} - \theta_{f,n_t}) & x_f \leq d(p,t) < x_{f+1} \\ \theta_{F,n_t} & d(p,t) \geq x_F \end{cases} \tag{3.3}$$

Here, θ_{f,n_t} denoted the profile weight at the supporting point x_f, i.e., the value of the profile function evaluated at x_f for bin n_t to which transcript t was assigned, $f \in \{1, \ldots, F\}$ with F the number of supporting points of the PLiF and $n_t \in \{1, \ldots, N\}$ with N the number of transcript length bins. Moreover, $d(p,t)$ was the closest distance of p to the boundaries of transcript t and $\delta_{p,t} := \frac{d(p,t) - x_f}{x_{f+1} - x_f}$.

The supporting points x_f were chosen in a way such that the first half $x_1, \ldots, x_{\lceil \frac{F}{2} \rceil}$ was linearly spaced between the 5' boundary of the transcript and a chosen maximal distance to the boundary. If this maximal distance was larger than the half of the transcript length, then only those supporting points were used that lay within the transcript. The second half of supporting points $x_{\lceil \frac{F}{2} \rceil + 1}, \ldots, x_F$ was analogously determined for the region at the 3' boundary. Thus, there were in total at most $F \cdot N$ optimisation variables related to the profile model.

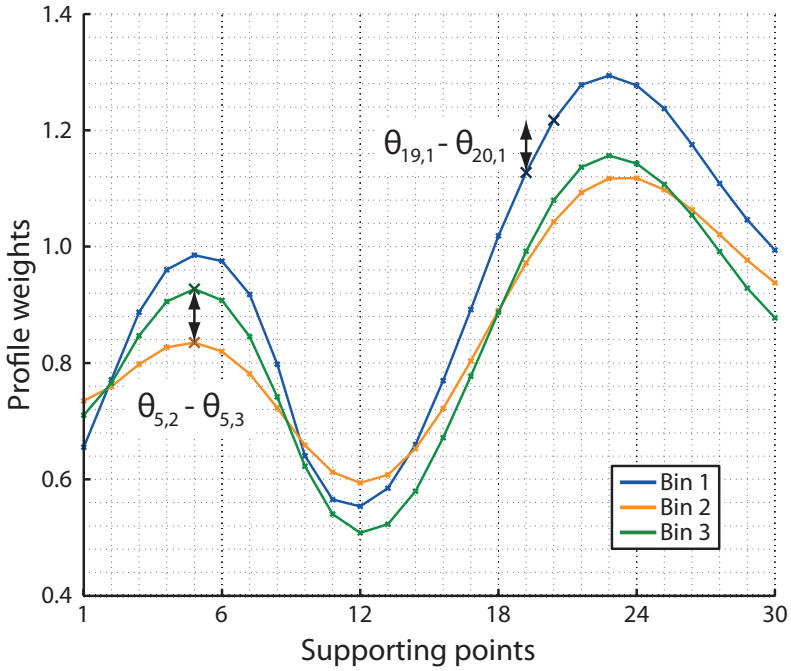


Figure 3.3.: Coupling of profile weights. Profile functions for three different transcript length bins are illustrated (see inlet). Coupling of profile weights of adjacent length bins to obtain similar profile *shapes* is indicated by an arrow for the profile weights $\theta_{5,2}$ and $\theta_{5,3}$ at supporting point 5 and length bins 2 and 3. Similarly, profile weights of adjacent supporting bins were coupled to obtain *smooth* profiles, exemplified for $\theta_{19,1}$ and $\theta_{20,1}$ at supporting points 19 and 20 and length bin 1.

To obtain smooth profile PLiFs, profile weights of adjacent supporting bins were coupled by penalising their squared deviation. Furthermore, a similar shape of PLiFs for different length bins was enforced by measuring the squared deviation of profile weights of adjacent length bins. Both measures are illustrated in Figure 3.3. In summary, the smoothness and similarity measures $\mathcal{R}^F(\boldsymbol{\theta})$ and $\mathcal{R}^N(\boldsymbol{\theta})$ were combined in the following term:

$$\mathcal{R}(\boldsymbol{\theta}) = \gamma^F \, \mathcal{R}^F(\boldsymbol{\theta}) + \gamma^N \, \mathcal{R}^N(\boldsymbol{\theta}) = \gamma^F \sum_{n=1}^{N} \sum_{f=1}^{F-1} \ell\left(\theta_{f,n},\ \theta_{f+1,n}\right) + \gamma^N \sum_{f=1}^{F} \sum_{n=1}^{N-1} \ell\left(\theta_{f,n},\ \theta_{f,n+1}\right)$$

The regularisation parameters γ^F and γ^N allowed to control the degree of smoothness and similarity, respectively. The higher the value was set, the stronger the respective effect was.

Sequence Bias Model As discussed above, the frequency of observing a read starting at certain nucleotide of the transcript is affected by the sequence content around this nucleotide. To normalise for this effect, sequence-dependent read density was estimated by a model based on Ridge regression (cf. [82] and Section 1.2.2). By this model, features $\boldsymbol{x}$ extracted from the transcript sequence by counting occurrences of k-mers closed to the considered nucleotide were related to logarithmised read start frequencies y, similarly as in [227]. Sequence content around a nucleotide p, the features $\boldsymbol{x}$ at p, was described by the occurrence of oligo-mers of length k, $k = 1, \ldots, K$, within a window around p of a certain size; K was the maximal considered oligo-mer length (e.g., $K = 3$).

The relation of features $\boldsymbol{x}$ and target values y was determined on a set of size M with $(\boldsymbol{x}^m, y^m)$-pairs, $m = 1, \ldots, M$, by solving the following optimisation problem of Ridge regression:

$$\underset{\boldsymbol{\beta}}{\text{minimise}} \sum_{m=1}^{M} \left(\boldsymbol{\beta}^T \boldsymbol{x}_m - y_m\right)^2 + \lambda \|\boldsymbol{\beta}\|_2^2$$

The predicted read start frequency at a nucleotide p of transcript t was then calculated based on the optimal $\boldsymbol{\beta}^*$:

$$\hat{y}_{p,t} = \boldsymbol{x}_{p,t}^T \, \boldsymbol{\beta}^*$$

The sequence-dependent read density $B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta})$ was estimated from the predicted read start frequencies $\hat{y}$:

$$B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) = \frac{\sum_{r \in R_p} \hat{y}_{r,t}}{\frac{1}{L} \sum_{q=1}^{L} \sum_{r \in R_q} \hat{y}_{r,t}} \tag{3.4}$$

where R_p was the set of positions where reads covering position p started and L was the length of transcript t.

Using Paired-end Read Data

Similarly to reads spanning splice site junctions, paired-end reads provide hints about which exons are connected and can thus facilitate the unambiguous assignment of reads to single transcripts. To take information from paired-end reads into account, the objective Ω was augmented by an additional term, the paired-end read loss:

$$\mathcal{L}_{paired} = \gamma^{PE} \sum_{e_1=1}^{E} \sum_{e_2=1}^{E} \frac{1}{l_{e_1} + l_{e_2}} \, \ell \left(c_{e_1,e_2}^{PE}, \sum_{t=1}^{T} w_t \, \psi_{e_1,e_2,t} \right)$$

Here, E is the size of the minimal set of segments. A segment is the maximal sequence of adjacent exons or parts of exons that is shared by a set of transcripts at a gene locus. Moreover, c_{e_1,e_2}^{PE} denoted the observed connectivity of two segments e_1 and e_2, i.e., how many paired-end reads connected the segments e_1 and e_2. Similarly, the expected connectivity of two segments e_1 and e_2 was determined by counting connected segments based on all possible paired-end reads that could be generated from the annotated transcript. The distance of the two reads of a simulated read pair, the insert size, was modelled by the median of the insert size distribution estimated from the observed paired-end reads.

Again, a quadratic loss function was used to penalise deviations of estimated and observed paired-end compatibility. These deviations were normalised by the sum of the lengths l_{e_1} and l_{e_2} of the segments e_1 and e_2 to account for differences in the sizes of the segments and thus different number of expected reads. The loss for paired-end information $\mathcal{L}_{paired}$ was weighted by an additional regularisation weight γ^{PE} in the objective.

The General rQuant Optimisation Problem

Using the definitions 3.3 and 3.4, the core rQuant optimisation problem can be formulated as:

$$\begin{aligned}\underset{\boldsymbol{w},\boldsymbol{\theta},\boldsymbol{\beta}}{\text{minimise}}\ \Omega(\boldsymbol{w},\boldsymbol{\theta},\boldsymbol{\beta}) = \gamma^E \sum_{p=1}^{P} \ell\left(c_p, \sum_{t=1}^{T} w_t\ \Theta(\boldsymbol{\theta},p,t)\ B(\boldsymbol{x}_{p,t},\boldsymbol{\beta})\right) \\ + \gamma^I \sum_{i=1}^{I} \ell\left(s_i, \sum_{t=1}^{T} w_t\ \eta_{i,t}\right) \\ + \gamma^{PE} \sum_{e_1=1}^{E}\sum_{e_2=1}^{E} \frac{1}{l_{e_1}+l_{e_2}}\ \ell\left(c^{PE}_{e_1,e_2}, \sum_{t=1}^{T} w_t\ \psi_{e_1,e_2,t}\right) \\ + \sum_{t=1}^{T} \gamma^w_t\ w_t + \gamma^F \sum_{n=1}^{N}\sum_{f=1}^{F-1} \ell\left(\theta_{f,n},\theta_{f+1,n}\right) + \gamma^N \sum_{f=1}^{F}\sum_{n=1}^{N-1} \ell\left(\theta_{f,n},\theta_{f,n+1}\right)\end{aligned}$$

$$\begin{aligned}\text{subject to}\quad & w_t \geq 0,\ \forall\ t = 1,\ldots,T \\ & \theta_{f,n} \geq 0,\ \forall\ f = 1,\ldots,F \text{ and } n = 1,\ldots,N.\end{aligned} \tag{3.5}$$

where P is the number of exonic positions of all considered genic loci and T is the number of transcripts of theses loci. For $\ell(\cdot,\cdot)$, a quadratic loss function was used (cf. Equation 3.2).

3.2.2. The rQuant Algorithm

The optimisation problem 3.5 is not convex with respect to all optimisation variables $\boldsymbol{w},\boldsymbol{\theta}$ and $\boldsymbol{\beta}$, since their appear in a product in the exon loss term. However, the problem is convex with respect to a part of variables, i.e., with respect to the transcript abundance variables $\boldsymbol{w}$, with respect to the profile variables $\boldsymbol{\theta}$ and with respect to the sequence bias variables $\boldsymbol{\beta}$.

Due to the non-convexity of the optimisation problem, it cannot be solved with standard algorithms that exploit convexity. Therefore, a solving strategy was chosen based on coordinate descent (cf. Section 1.2.2 and [34, Chapter 9]) by minimising the objective Ω with respect to a single variable or blocks of variables at one step, exploiting convexity with respect to these variables.

The rQuant algorithm is formally described in Algorithm 3.1. For read density estimation, i.e., for the estimation of the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$, the optimisation problem with respect to all variables was solved on a subset of highly abundant single-transcript genes to guarantee sufficient read coverage for bias estimation. The minimum of the objective with respect to each transcript abundance and profile variable and the set of sequence bias variables was determined. This procedure was repeated in an iterative manner until the solution of $\boldsymbol{w},\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ did not vary anymore at a given precision between two iterations. It has been shown that such an approach converges to the optimal solution because the optimisation problem has a unique minimum with respect to this block of coordinates (cf. [201, and references therein] and the following sections).

Using the estimated read density, the transcript quantification problem was subsequently solved for each gene locus from the provided annotation applying the same coordinate descent approach as above with respect to the transcript abundance variables only.

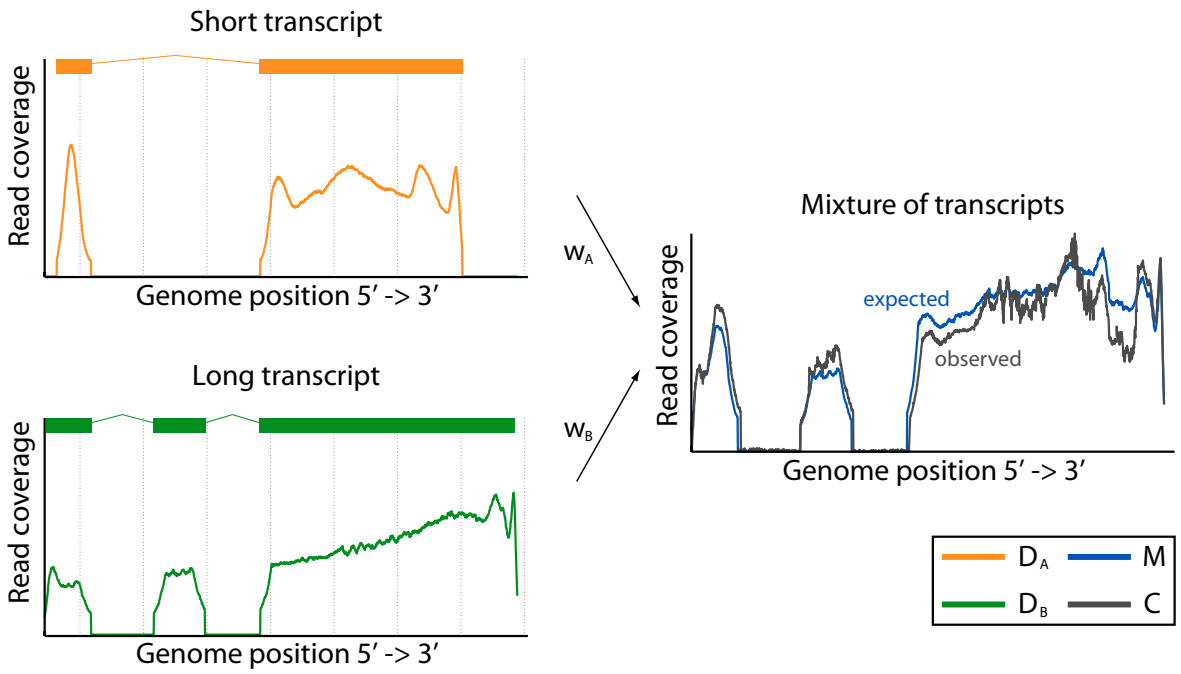


Figure 3.4.: Basic mode of action of rQuant. The key component of rQuant is to infer the underlying read coverage of all transcripts at one gene locus (two transcripts in the illustration on the left: transcript A is shown in orange and transcript B in green), such that the difference between the observed (grey) and expected (blue) read coverage is minimised. The expected read coverage $M = w_A D_A + w_B D_B$ is calculated from the transcript abundances w_A and w_B and the read densities D (shown in the graphs on the left), which are estimated simultaneously for several loci.

Algorithm 3.1 The algorithm of rQuant

procedure RQUANT(genes, aligned reads)
  $\boldsymbol{\theta}, \boldsymbol{\beta} \leftarrow$ OPT_DENSITY(genes^{train}, aligned reads)
  for all $g \in$ genes **do**
    # Finding the optimal transcript weights for gene g
    $\boldsymbol{w}^g \leftarrow \underset{\boldsymbol{w}^g \geq \mathbf{0}}{\operatorname{argmin}}\ \Omega(\boldsymbol{w}^g)$ *# Cf. Equation 3.6*
  end for
  return $\boldsymbol{w}^{\text{genes}}, \boldsymbol{\theta}, \boldsymbol{\beta}$
end procedure

procedure OPT_DENSITY(genes, aligned reads)
  $w_t \leftarrow$ mean coverage at locus of t, $\forall\ t, \ldots, T$
  $\boldsymbol{\theta} \leftarrow \mathbf{1}$
  $\boldsymbol{\beta} \leftarrow \mathbf{1}$
  repeat
    $\boldsymbol{w}^{old} \leftarrow \boldsymbol{w}$
    $\boldsymbol{\theta}^{old} \leftarrow \boldsymbol{\theta}$
    $\boldsymbol{\beta}^{old} \leftarrow \boldsymbol{\beta}$
    # Finding the optimal sequence weights
    $\boldsymbol{\beta} \leftarrow$ RIDGE_REGRESSION(genes, aligned reads, $\boldsymbol{w}$, $\boldsymbol{\theta}$) *# Cf. Equation 3.8*
    update Ω
    # Finding the optimal transcript weights
    for $t = 1, \ldots, T$ **do**
      $w_t \leftarrow \underset{w_t \geq 0}{\operatorname{argmin}}\ \Omega(w_t)$ *# Cf. Equation 3.6*
    end for
    update Ω
    # Finding the optimal profile weights
    for $f = 1, \ldots, F$ **do**
      for $n = 1, \ldots, N$ **do**
        $\theta_{f,n} \leftarrow \underset{\theta_{f,n} \geq 0}{\operatorname{argmin}}\ \Omega(\theta_{f,n})$ *# Cf. Equation 3.7*
      end for
    end for
    update Ω
  until $\left\| [\boldsymbol{w}, \boldsymbol{\theta}, \boldsymbol{\beta}] - \left[\boldsymbol{w}^{old}, \boldsymbol{\theta}^{old}, \boldsymbol{\beta}^{old}\right] \right\| < \epsilon$
  return $\boldsymbol{\theta}, \boldsymbol{\beta}$
end procedure

Finding the Optimal Transcript Weights

To determine the optimal abundances or weights of transcripts at the same genic locus by coordinate descent, the objective Ω was reformulated with respect to a single transcript weight $w_{t'}$:

$$\begin{aligned}
\Omega(w_{t'}) &= \gamma^E \sum_{p=1}^{P}\left(\sum_{t=1}^{T} w_t\ D_{p,t}(\boldsymbol{\beta},\boldsymbol{\theta}) - c_p\right)^2 + \gamma^I \sum_{i=1}^{I}\left(\sum_{t=1}^{T} w_t\ \eta_{i,t} - s_i\right)^2 \\
&\quad + \gamma^{PE} \sum_{e_1=1}^{E}\sum_{e_2=1}^{E}\left(\sum_{t=1}^{T} w_t\ \psi_{e_1,e_2,t} - c^{PE}_{e_1,e_2}\right)^2 + \sum_{t=1}^{T}\gamma_t\ w_t + \mathcal{R}(\boldsymbol{\theta}) \\
&= \gamma^E \sum_{p=1}^{P}\left(w_{t'}\ D_{p,t'}(\boldsymbol{\beta},\boldsymbol{\theta}) + \underbrace{\sum_{t:t\neq t'} w_t\ D_{p,t}(\boldsymbol{\beta},\boldsymbol{\theta}) - c_p}_{R^w_1}\right)^2 + \gamma^I \sum_{i=1}^{I}\left(w_{t'}\ \eta_{i,t'} + \underbrace{\sum_{t:t\neq t'} w_t\ \eta_{i,t} - s_i}_{R^w_2}\right)^2 \\
&\quad + \gamma^{PE} \sum_{e_1=1}^{E}\sum_{e_2=1}^{E}\left(w_{t'}\ \psi_{e_1,e_2,t} + \underbrace{\sum_{t:t\neq t'}^{T} w_t\ \psi_{e_1,e_2,t} - c^{PE}_{e_1,e_2}}_{R^w_3}\right)^2 + \gamma_{t'}\ w_{t'} + \sum_{t:t\neq t'}\gamma_t\ w_t + \mathcal{R}(\boldsymbol{\theta}) \\
&= w_{t'}{}^2 \underbrace{\left(\gamma^E \sum_{p=1}^{P} D_{p,t'}(\boldsymbol{\beta},\boldsymbol{\theta})^2 + \gamma^I \sum_{i=1}^{I}\eta_{i,t'}{}^2 + \gamma^{PE}\sum_{e_1=1}^{E}\sum_{e_2=1}^{E}\psi_{e_1,e_2,t}{}^2\right)}_{S_1} \\
&\quad + w_{t'} \underbrace{\left(2\ \gamma^E \sum_{p=1}^{P} D_{p,t'}(\boldsymbol{\beta},\boldsymbol{\theta})\ R^w_1 + 2\ \gamma^I \sum_{i=1}^{I}\eta_{i,t}\ R^w_2 + \gamma_{t'} + 2\ \gamma^{PE}\sum_{e_1=1}^{E}\sum_{e_2=1}^{E}\psi_{e_1,e_2,t}\ R^w_3\right)}_{S_2} \\
&\quad + \underbrace{\gamma^E \sum_{p=1}^{P} R^{w2}_1 + \gamma^I \sum_{i=1}^{I} R^{w2}_2 + \gamma^{PE}\sum_{e_1=1}^{E}\sum_{e_2=1}^{E} R^{w2}_3 + \sum_{t:t\neq t'}\gamma_t\ w_t + \mathcal{R}(\boldsymbol{\theta})}_{S_3}
\end{aligned}$$

With this formulation, Ω was quadratic in $w_{t'}$, thus convex, and the globally minimal $w^*_{t'}$ could be determined by fulfilling the necessary and sufficient condition for the global minimum of a function. The stationary point of Ω was calculated by setting the first derivative to 0 and solving for $w_{t'}$:

$$\begin{aligned}
\frac{d\Omega}{dw_{t'}} = 2\ S_1\ w_{t'} + S_2 &= 0 \\
\Rightarrow w^*_{t'} &= \frac{-S_2}{2\ S_1}
\end{aligned}$$

The sufficient condition $\frac{d\Omega}{dw_{t'}{}^2} = 2\ S_1 \geq 0$ is fulfilled, as S_1 is always non-negative. To fulfil the constraint that the transcript weights must be non-negative, $w^*_{t'}$ was clipped to 0 if negative:

$$w^{clipped}_{t'} = \begin{cases} w^*_{t'} & w^*_{t'} \geq 0 \\ 0 & w^*_{t'} < 0 \end{cases} . \tag{3.6}$$

All values of the objective evaluated at $w \neq w_t^*$ are larger than $\Omega(w_{t'}^*)$, as Ω with respect to one variable is a quadratic function and thus has a global minimum (at $w_{t'}^*$). In particular, $\Omega(w) > \Omega(w_{t'}^*)\ \forall\ w > w_t^*$ and $\Omega(w_2) > \Omega(w_1)$ for $w_2 > w_1$ with $w_1, w_2 > w_t^*$, i.e., Ω is strictly increasing in this interval. Therefore, if $w_{t'}^* < 0$, then $\Omega(w_{t'}^{clipped}) = \Omega(0) < \Omega(w),\ \forall\ w > 0$. Clipping to 0 thus leads to the optimal solution over $[0, \infty)$.

Finding the Optimal Profile Weights

Solving the optimisation problem 3.5 with respect to each profile weight $\theta_{f,n}$ was similarly implemented by a reformulation of the objective Ω. The derivation of the following equation can be found in Section C.1.1 in the Appendix.

$$
\begin{aligned}
\Omega(\theta_{f',n'}) &= \gamma^E \sum_{p=1}^{P} \left(\sum_{t=1}^{T} w_t\ \Theta(\boldsymbol{\theta}, p, t)\ B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) - c_p \right)^2 \\
&+ \gamma^N \sum_{f=1}^{F} \sum_{n=1}^{N-1} (\theta_{f,n} - \theta_{f,n+1})^2 + \gamma^F \sum_{n=1}^{N} \sum_{f=1}^{F-1} (\theta_{f,n} - \theta_{f+1,n})^2 \\
&+ \gamma^I \sum_{i=1}^{I} \left(\sum_{t=1}^{T} w_t\ \eta_{i,t} - s_i \right)^2 + \mathcal{R}(\boldsymbol{w}) \\
&\overset{\text{C.1.1}}{=} \theta_{f',n'}{}^2 \underbrace{\left(\gamma^E \sum_{\substack{p:\ d(p,\cdot) \geq x_{f'-1} \\ \wedge\ d(p,\cdot) < x_{f'+1}}} R_1^{\theta 2} + 2\ \gamma^N + 2\ \gamma^F \right)}_{S_1} \\
&+ \theta_{f',n'} \underbrace{\left(\gamma^E \sum_{\substack{p:\ d(p,\cdot) \geq x_{f'-1} \\ \wedge\ d(p,\cdot) < x_{f'+1}}} 2\ R_1^{\theta}\ R_2^{\theta} + \gamma^F R_1^F + \gamma^N R_1^N \right)}_{S_2} \\
&+ \underbrace{\gamma^E \sum_{\substack{p:\ d(p,\cdot) \geq x_{f'-1} \\ \wedge\ d(p,\cdot) < x_{f'+1}}} R_2^{\theta 2} + R_3^{\theta} + \gamma^F R_2^F + \gamma^N R_2^N + \gamma^I \sum_{i=1}^{I} \left(\sum_{t=1}^{T} w_t\ \eta_{i,t} - s_i \right)^2 + \mathcal{R}(\boldsymbol{w})}_{S_3}
\end{aligned}
$$

The globally minimal $\theta_{f',n'}^*$ was found at

$$\theta_{f',n'}^* = \frac{-S_2}{2\ S_1}$$

and $\theta_{f',n'}^*$ was clipped to

$$\theta_{f',n'}^{clipped} = \begin{cases} \theta_{f',n'}^* & \theta_{f',n'}^* \geq 0 \\ 0 & \theta_{f',n'}^* < 0 \end{cases} . \tag{3.7}$$

Clipping to 0 leads to the optimal solution over $[0, \infty)$ as discussed in the previous section.

Finding the Optimal Sequence Weights

The optimal sequence weights in an iteration were determined by using the analytical solution of the Ridge regression problem (cf. Equation 3.4). The solution of the problem could be calculated with the equation (cf. [82] and Section 1.2.2):

$$\boldsymbol{\beta}^* = \left(\sum_{m=1}^{M} \boldsymbol{x}_m {\boldsymbol{x}_m}^T + \lambda I \right)^{-1} \sum_{m=1}^{M} y_m \boldsymbol{x}_m \text{ with } \lambda > 0 \tag{3.8}$$

The optimal $\boldsymbol{\beta}^*$ was then used to derive $B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta})$ as described in Equation 3.4.

In contrast to the optimisation of $\boldsymbol{w}$ and $\boldsymbol{\theta}$, the components of $\boldsymbol{\beta}$ were not optimised by each coordinate, but in a whole block. Nevertheless, this approach converges to the optimal solution as the optimisation problem has a unique minimum with respect to this block of coordinates [201, and references therein].

3.2.3. Quantifying Multiple Conditions

Often, RNA-seq experiments are conducted in order to compare a set of transcriptomes derived from different stress conditions, diverse tissues or several individuals. One important part of such a comparison is to describe differences in terms of changes in transcript abundances between these samples. This might be approached by performing a statistical test based on the read count data directly, e.g., by the tools rDiff [193] and DESeq [13], or based on estimated abundances, e.g., by the tool Cuffdiff [171, 200]. When taking the way of comparing abundance estimates, RNA-seq measurements from different conditions can be exploited to obtain more accurate and stable quantification results, as many transcripts are expected to be present at the same level.

rQuant was therefore extended to quantify transcripts based several conditions at one step. The basic idea was to make transcript weights for the set of conditions similar, which was realised in a regularisation term in the objective. This was formally implemented in the optimisation problem:

$$\begin{aligned} \underset{\boldsymbol{W}}{\text{minimise}}\ \Omega(\boldsymbol{W}) = \sum_{p=1}^{P} \sum_{a=1}^{A} \ell \left(c_{p,a},\ \sum_{t=1}^{T} w_{a,t}\ \delta_{p,a,t} \right) + \sum_{t=1}^{T} \gamma_t^w \cdot \max_{\substack{a_1=1,\ldots,A \\ a_2=1,\ldots,A}} |w_{a_1,t} - w_{a_2,t}| \\ \text{subject to} \qquad w_{a,t} \geq 0,\ \forall\ t = 1, \ldots, T \text{ and } a = 1, \ldots, A. \end{aligned} \tag{3.9}$$

Terminology was similar as for the rQuant optimisation problem with a single condition. Here, A was the number of conditions and $\boldsymbol{W}$ described the matrix of transcript weights with dimensions A and T. The exon loss was summed up across all conditions $a = 1, \ldots, A$. The transcript weights were coupled by penalising the maximal deviation of weights in a pair-wise comparison across conditions within the transcript t. This regularisation term of $\boldsymbol{W}$ was weighted by the parameter γ_t^w for each transcript t, which was set to the length of the respective transcript.

For implementation simplicity, the optimisation problem for multiple experiments was solved with the MATLAB function FMINCON from the optimisation toolbox that solves constrained optimisation problems [135].

3.2.4. Data Simulation

The performance of rQuant was evaluated on a set of artificially generated reads. This was beneficial to biochemical validation experiments such as qPCR, since those techniques are usually biased and imprecise to measure the ground truth. In contrast, the ground truth was known when reads were artificially simulated from transcripts with given abundance.

Artificial Reads for C. elegans with Strong Library Biases

To generate an artificial read data set, the Flux Simulator (build 20101223) was used, a software for transcriptome and read generation that simulates the biochemical processes underlying the library preparation [173]. Reads were simulated for 5,983 randomly selected gene loci from the *Caenorhabditis elegans* WormBase 200 annotation [78]. When two transcripts of one gene locus shared more than 90 % of exonic positions, one of them was randomly filtered. The filtering was undertaken to avoid sets of transcripts that only differ at their ends, which is common in the *C. elegans* annotation. After filtering, 10,195 transcripts were left covering 10.6 million exonic nucleotides (10 % of the genome sequence). Half of the loci were annotated with multiple transcripts, of which 74 % had two, 17 % had three, 6 % had four and 3 % had at least five transcripts.

For read simulation with the Flux Simulator, the number of initial RNA molecules was set to 20 million. The parameters of the simulation were chosen in a way to obtain a data set with strong library biases: The molecules were chemically fragmented before reverse transcription with poly-dT priming, the length of cDNA molecules were set to the range of 500 to 5,500 nt, and gel size selection was restricted to lengths of 200 to 250 nt. In total, 9,545,602 reads were generated with this strategy. Positional and transcript length biases induced by this simulation strategy are visualised in Figure 3.5 (a).

Artificial Reads for C. elegans with Strong Library and Sequence Biases

Secondly, the reads generated by the strategy described above were used to simulate biases dependent on the sequence content. Reads starting with AA, AT, TA, TT and CC, CG, GC, GG were filtered in a way such that they were underrepresented by a factor of ≈ 0.7 and overrepresented by a factor of ≈ 3.6, respectively. By this, the resulting read set contained 8,698,606 reads.

Artificial Paired-end Reads for C. elegans with Weak Library Biases

A third artificial read set was obtained based on the same transcripts and sampled abundances as described above. Here, reads were generated from both ends of the fragments to obtain paired-end reads. Moreover, for a data set with weak library biases, chemical fragmentation was followed by reverse transcription with random priming of 10 to 10,000 nt long molecules. The fragments were selected by size in a range of 175 to 225 nt, resulting in pairs of reads with median insert size of 45 nt. The final data set for paired-end analysis contained 18,341,371 reads. Figure A.4 shows the profiles for this data set.

Artificial Reads for A. thaliana for Three Conditions

For the analysis of quantification across multiple conditions, a fourth data set was assembled. Here, 10,807 transcripts from the *Arabidopsis thaliana* annotation TAIR10 [195] at 4,462 randomly selected gene loci with multiple isoforms were taken for simulation of read data. Gene abundances were sampled from a distribution estimated on a true *A. thaliana* RNA-seq data set with two conditions. Relative transcript abundances were sampled from a uniform distribution within the range of 0 to 1 and normalised such that their norm was equal to 1. A change between two conditions was implemented by adapting the relative transcript abundances. Half of the genes were set to be differentially expressed (2,231 examples). Then, the relative transcript abundances were multiplied by the gene abundance to obtain absolute transcript abundances. By this strategy, transcript abundances for three conditions were simulated. These sets were used to generate reads with the Flux Simulator, simulating chemical fragmentation before reverse transcription with random primers and a selection of fragments of size between 175 and 225 nt. In total, 7,502,243 reads were obtained for condition 1, 7,528,072 reads for condition 2 and 7,402,396 reads for condition 3.

3.2.5. Preparation of RNA-seq Data from Arabidopis lyrata

Arabidopis lyrata is a plant that belongs to the genus *Arabidopsis*, which is well-known because of its prominent member, the model organism *Arabidopsis thaliana*. Its genome sequence has been recently described by Hu et al. [87]. RNA-seq reads were generated based on libraries for two biological replicates of *A. lyrata*. The reads were aligned with PALMapper [50, 98] to the *A. lyrata* genome sequence arranged in 3,648 scaffolds (JGI release v1.0, [87]), resulting in 150 million and 173 million alignments for replicate 1 and 2, respectively. The annotation (JGI release v1.0) contained 32,670 protein-coding transcripts at 32,377 loci. The majority of loci harboured one single transcript (99 %), primarily due to incompleteness of this version, since it was the first release for annotated *A. lyrata* transcripts.

3.3. Results and Discussion

3.3.1. Artificial Data Sets

rQuant was applied to quantify transcripts from simulated reads for the artificial data sets described in Section 3.2.4. The first data set was used to evaluate rQuant in is simplest settings and with profile model and to compare it to other approaches; for the assessment of the sequence bias model, the second set of reads was taken; the third data set was applied to test rQuant with its model for paired-end reads; the extension of rQuant to quantify multiple conditions simultaneously was tested on the fourth data set.

The quality of transcript abundance estimates generated by rQuant was evaluated by determining the correlation between true and inferred abundance. To measure the accuracy of rQuant independent of the gene complexity, Pearson's correlation coefficient (PCC, cf. Section C.1.2 in the Appendix) was calculated *across genes*, i.e., transcripts from all genes were considered together, regardless of which gene loci they belonged to. In addition, the correlation of true and inferred abundances normalised to the total gene abundance was determined for transcripts *per gene* with alternative transcripts and then averaged to evaluate how well transcripts could be quantified at multiple-transcript loci.

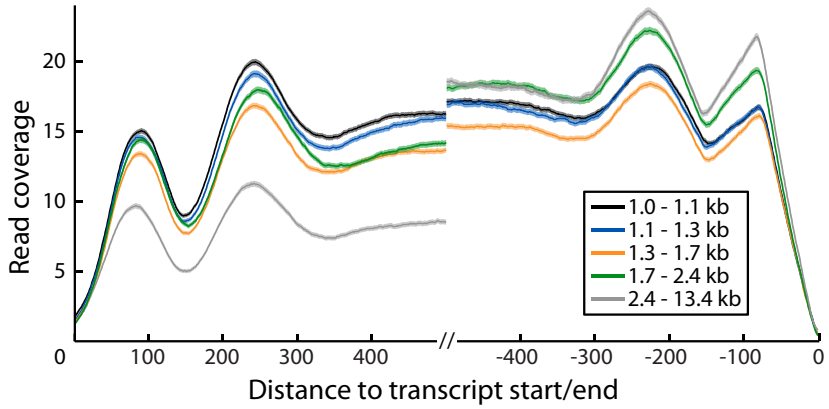
rQuant Baseline

In a first experiment, rQuant was applied in its simplest setting without modelling read density, which served as a baseline to compared against for other settings (cf. Table 3.1, section 'Baseline'). Measuring the correlation across genes, transcript abundances estimated by rQuant very well correlated with the true abundances (0.940). The average correlation determined within multiple-transcript genes was reduced, but still at a high level (0.805). Assessing the correlation across genes by low, medium and high expression levels showed that the transcript quantification problem was harder to solve for lowly than for highly expressed transcripts (from 0.109 to 0.938). This dependency was not so apparent for the correlation per gene, possibly evened out by the assignment of expression bins according to the median of transcript abundances in a gene. When calculating Spearman's correlation coefficient (SCC), which measures monotonic rather than linear dependence (cf. Section C.1.2 and Table B.15 in the Appendix), the effect of the expression level was also not as strong. This might suggest that true and inferred abundance, in particular for low expression, were non-linearly related. Grouping transcripts by their length, correlation dropped for long transcripts. This effect may not necessarily be caused by the transcript length directly, but by the degree of gene complexity, i.e., the number of alternative events, which increases with length of the processed transcribed region. Moreover, the dependency of the number of transcripts per gene locus was assessed. The correlation measured across genes declined with increased degree of gene complexity. Determined per gene however, the correlation stayed almost at the level.

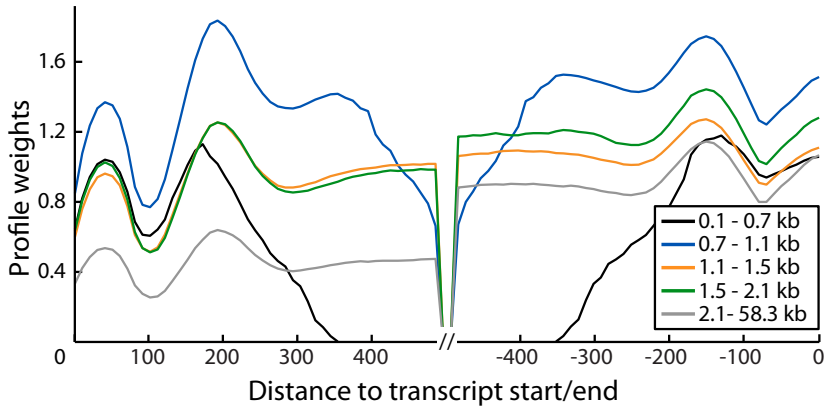
Furthermore, abundance estimates for single-transcript and multiple-transcript genes were evaluated separately (cf. Table B.16). Two main observations could be made from this comparison. The problem of quantifying genes with alternative transcripts was in general more difficult than for single-transcript genes (0.913 versus 0.996). Quantification accuracy depended on expression level, transcript length and gene complexity for multiple-transcript genes, whereas there was only slight dependency on these factors for genes with one single transcript.

rQuant with Profiles

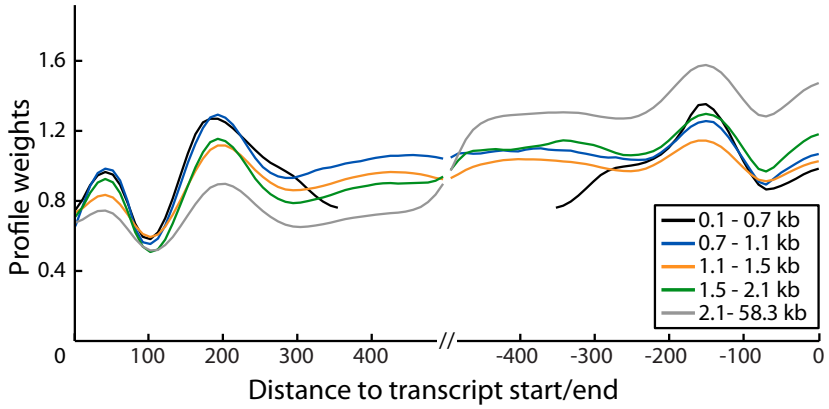
In a second experiment, the performance of rQuant with estimated profile model to normalise for positional and transcript length bias was assessed. In one approach, profiles were estimated by empirically calculating read density along the transcript for different transcript length bins on all single-transcript genes in the data set. In a second approach, these profiles were determined by using the optimisation strategy described in Section 3.2.1 on a set of randomly selected positions from single-transcript genes ($\approx$ 800,000 positions). The results shown here were determined on the best combination out of a grid of the parameters ($\gamma^E = 1$, $\gamma^F = 10^5$, $\gamma^N = 1$, $F = 100$, $N = 5$). The coordinate descent algorithm converged to the optimal solution at a precision of 10^{-2} after 64 iterations (taking $\approx$ 10 minutes each). In Figure 3.5 (b) and (c), the estimated profile piece-wise linear functions are visualised. Table 3.1, section 'Profiles' shows PCCs for the two chosen profile approaches. For both settings, the correlations considerably improved compared to the baseline method, showing that profile learning helped in improving quantification accuracy. The approach of estimating profiles by optimisation was slightly better for PCC per gene than for empirical profile estimation, and performed better for both measures assessed with SCC (cf. Table B.17). As this difference was, however, not very large and empirical profile estimation was considerably faster, this approach might be more efficient for practical usage of rQuant than profile optimisation.



(a) Simulated transcript profiles.



(b) Estimated empirical transcript profiles.



(c) Estimated optimal transcript profiles.

Figure 3.5.: Estimating transcript profiles. (a) Normalised read coverage is shown in dependency of the absolute distance to the transcript boundaries for bins of different transcript length (see inlet). The read coverage was determined from simulated reads for transcripts longer than 1,000 nt in the artificial data set with strong library biases described in Section 3.2.4. (b) The profile piece-wise linear functions that were empirically determined as described in Section 3.3.1 are illustrated. (c) The profile piece-wise linear functions that were estimated by optimisation are shown (cf. Section 3.3.1). Comparing (c) to (b), the effect of the regulariser of the profile weights is apparent, as the curves in (c) are smoother and more similar than the empirically estimated functions shown in (b).

Table 3.1.: Evaluation of rQuant on artificial data. Pearson's correlation coefficient between true and inferred transcript abundances was calculated across genes and averaged per gene. Sample sizes are given in brackets. Correlation for the baseline method (rQuant without read density estimation) is grouped by true expression (low: < 500 molecules, medium: 500 to 1,500 molecules, high: $\geq$ 1,500 molecules), transcript length (short: < 1000 nt, medium: 1,000 to 2,000 nt, high: $\geq$ 2,000 nt) and number of transcripts at one gene locus. Results for rQuant with estimated profile model that was determined empirically and by optimisation are listed. For comparison, transcript abundances were estimated by a segment-based version of rQuant, Cufflinks [171, 200] and MISO [105]. The results for rQuant baseline and rQuant with profiles are highlighted.

Approach		Pearson's correlation across genes		Pearson's correlation per gene	
rQuant baseline		**0.940**	[10,180]	**0.805**	[3,022]
	By expression				
	low	0.109	[3,770]	0.702	[720]
	medium	0.479	[3,279]	0.822	[1,310]
	high	0.938	[3,131]	0.856	[992]
	By transcript length				
	short	0.996	[3,272]	0.782	[818]
	medium	0.990	[4,680]	0.866	[1,489]
	long	0.706	[2,228]	0.702	[715]
	By number of transcripts				
	1	0.996	[2,945]	n/a	
	2	0.984	[4,500]	0.787	[2,249]
	3	0.723	[1,515]	0.852	[505]
	4 and more	0.614	[1,220]	0.868	[268]
rQuant profiles					
	empirical	**0.957**	[10,180]	0.832	[3,022]
	optimal	0.946	[10,180]	**0.836**	[3,022]
Other methods					
	rQuant segment-based	0.953	[10,195]	0.561	[3,010]
	Cufflinks	0.932	[10,180]	0.758	[3,022]
	Cufflinks bias corrected	0.936	[10,180]	0.793	[3,022]
	MISO	n/a		0.693	[2,903]

Comparison to Other Quantification Tools

Besides the internal rQuant evaluation, other quantification methods were considered for comparison (cf. Table 3.1, section 'Other methods' and Table B.16 for a comparison by gene complexity). As the majority of available quantification tools quantifies transcripts based on segments or exons rather than on positions, a similar approach as for rQuant was implemented for segments by calculating the median of positional read coverage within a segment. Correlation across genes was at the same magnitude as for the rQuant baseline, but measuring per-gene correlation dropped to 0.535. The per-gene correlation was at this low level compared to the correlation across genes because of extreme outliers contributing to the

mean. In addition, two available quantification tools, Cufflinks [171, 200] and MISO [105], were applied using their default parameter settings to quantify the transcripts in the simulated data set. Cufflinks without and with bias correction showed slightly worse performance for both settings in terms of the two correlation measures compared to the rQuant baseline. MISO only uses gene loci with alternative transcripts for quantification as it estimates abundances relative to the gene abundance. Thus, an evaluation could only be conducted on those transcripts and a comparison was only meaningful for the correlation measurements based on multiple-transcript loci. The accuracy of MISO was considerably worse compared to the rQuant baseline as well as Cufflinks.

The compared methods rQuant, Cufflinks and MISO did not only differ in terms of accuracy, but also in running time. Running rQuant without read density estimation took 14 minutes for the artificial data set, i.e., it required ≈ 0.15 seconds per gene locus on average. Cufflinks ran very fast, taking 80 seconds for the data set (≈ 0.01 seconds per gene locus). The running time of MISO was estimated to 2,400 minutes for the set of multiple-transcript loci only (≈ 49 seconds per gene locus).

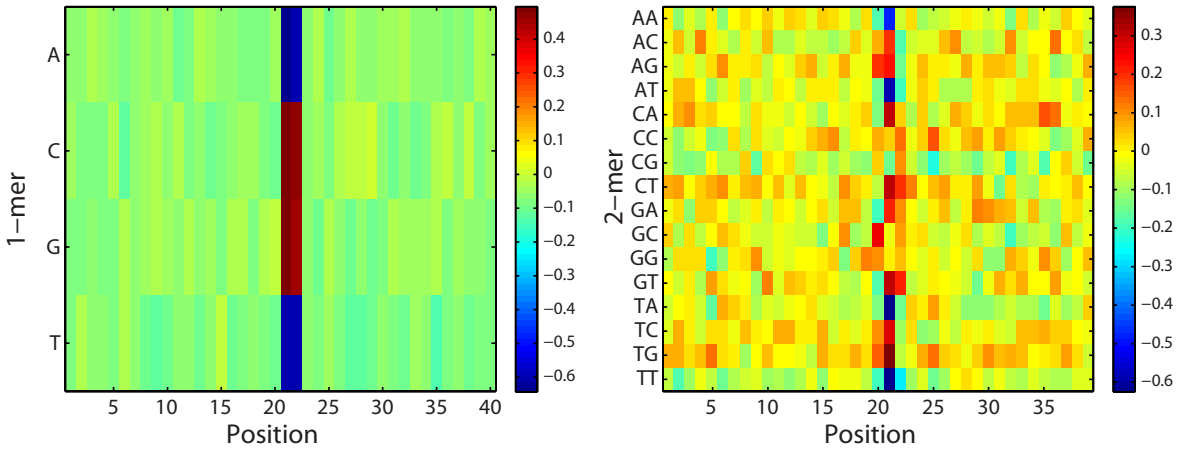
rQuant with Sequence Bias Model

On the artificial read data set with simulated sequence bias, the parameters of the sequence model described in Section 3.2.1 were determined and used for quantification with rQuant. From single-transcript gene loci with mean abundance above 150, 75,660 positions were randomly selected as a training set. For theses examples, features based on the occurrence of mono- and di-nucleotides in a window of 20 nt upstream and 20 nt downstream of considered nucleotide were generated. Thus, 784 features (160 for mono-nucleotides, 624 for di-nucleotides) per example were present. As target values for the regression, read start frequencies were calculated from the observed reads. The optimal sequence weights $\boldsymbol{\beta}^*$ were determined on this training set by using the analytical solution of the Ridge regression model (cf. Equation 3.4).

On the training set and on a test set of 18,916 examples, the degree of reducing variability within the set of read start frequencies was assessed. For this, the ratio of the average squared deviation of true and predicted read start frequencies $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ and the average squared deviation of true target values to their mean was calculated. This analysis showed that ≈ 5 % of the variability could be captured by the sequence model. The remaining variability was likely to be caused by the positional and transcript length bias inherent in this data set and thus could not be explained by the sequence content.

Figure 3.6 visualises the optimal sequence weights. It is obvious that under-representation of A and T as well as AA, AT, TA, TT at read starts were correctly recognised by the model by down-weighting the respective sequence features (blue blocks). Similarly, the model correctly assigned high values to overrepresented 1-mers and 2-mers (red blocks).

The trained sequence model was used to estimate the sequence-dependent read density B while quantifying transcripts. Although the regression model based on sequence features could correctly capture distortions of read starts frequencies, quantification accuracy could not be improved by incorporating the sequence model (0.927 without sequence bias correction compared to 0.918 with correction in terms of PCC across genes, 0.708 to 0.664 in terms of mean PCC per gene). A potential reason for this could be that the sequence model was determined on read starts, but the quantification approach of rQuant was based on read coverage.



(a) Mono-nucleotides (1-mers).

(b) Di-nucleotides (2-mers).

Figure 3.6.: Sequence weights from the regression model. (a) Regression weights for features describing the occurrence of mono-nucleotides (A, C, G, T) within a window of 40 nt are illustrated. Read starts correspond to position 21. High values (red) correspond to overrepresented, low values (blue) to underrepresented mono-nucleotides. (b) Regression weights for features describing the occurrence of di-nucleotides (AA, AC, ..., TG, TT) within a window of 40 nt are shown.

rQuant with Paired-end Reads

rQuant with its extension for to paired-end reads was applied to 1,151 transcripts from 255 gene loci with at least four isoforms, which were taken from the artificial data set simulated for paired-end reads. The analysis was limited to complex gene loci since the benefit of paired-end reads was expected to be stronger for several isoforms [189]. Compared to rQuant without paired-end loss, correlation across genes was at the same level for PCC (0.984), but increased from 0.910 to 0.913 in terms of SCC. Measuring correlation within genes, quantification accuracy improved from 0.953 to 0.955 in terms of PCC (0.850 to 0.855 for SCC). A theoretical analysis by Smith et al. [189] showed that $\approx$ 20 % of the isoforms in the *C. elegans* annotation at gene loci with more than three isoforms cannot be identified by paired-end reads at unlimited sequencing depth over a range of insert sizes and this number increases when sequencing depth decreases. This observation may explain why the effect of using paired-end information was not very strong for the used artificial data set.

rQuant for Quantification of Multiple Conditions

The approach described in Section 3.2.3 (rQuant.multi) was applied to the data set with artificial reads for three conditions. For comparison, the same sets of transcripts with reads simulated for the respective conditions were quantified by rQuant with very simple settings, i.e., without intron loss and bias estimation (rQuant.single).

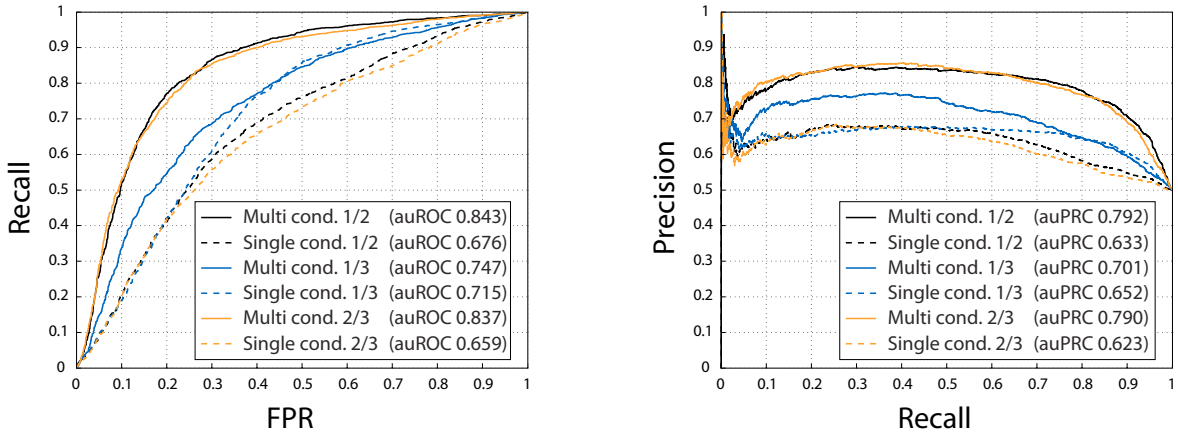
In a first step, the average Pearson's correlation coefficient within genes was determined to compare true and estimated transcript abundance for each condition separately. In terms of this measure, quantification accuracy could be improved for all three conditions when comparing rQuant.single to rQuant.multi (from 0.703 to 0.714 for condition 1, from 0.701 to 0.724 for condition 2 and from 0.719 to 0.729 for condition 3). This showed the profit of

sharing measurements from several conditions in order to make quantification results more reliable.

As a second evaluation measure, the performance of both strategies was assessed when using their estimated abundances in a statistical test to discover differentially expressed transcripts. For this purpose, transcript abundances predicted by rQuant.multi and rQuant.single for each condition were converted to read counts. Applying these counts to the tool DESeq for statistical testing [13] provided sets of differentially expressed transcripts between pairs of conditions. Gene-specific p-values were determined from transcript-specific p-values by Bonferroni correction. The prediction whether a gene was differentially expressed or not was based on these corrected p-values.

Using the predictions and the ground truth, the number of true positives, false positives, false negatives and true negatives were calculated at varying p-values cut-offs to describe the performance of the methods in terms of sensitivity in relation to false positive rate (receiver operating characteristic, ROC) and in terms of precision in dependency of recall (precision-recall curve, PRC). The resulting ROC and precision-recall curves for all pairs of conditions based on abundances estimated by rQuant.multi and rQuant.single are shown in Figure 3.7.

Both ROC and PRC analysis showed that the simultaneous usage of read coverage from several conditions (rQuant.multi) in quantification helped to improve detection accuracy of differentially expressed transcripts compared to independently quantification for the conditions. This effect was most prominent when comparing condition 1/2 and 2/3 and areas under ROC and PRC could be improved by at most 25.1 % and 27.8 %, respectively (cf. inlets of Figure 3.7).



(a) Receiver operating characteristic curves.

(b) Precision-recall curves.

Figure 3.7.: Comparison of rQuant.multi and rQuant.single. Performance of these two strategies was assessed by applying a statistical test to read counts derived from abundances estimated by rQuant.multi and rQuant.single. Curves denoted by 'multi' refer to tests based on rQuant.multi and 'single' to tests based on rQuant.single (see inlets). Tests were conducted for pairs of conditions 1/2, 1/3 and 2/3. (a) Receiver operating characteristic curves. The recall in dependency of the false positive rate (FPR) is shown for varying p-value cut-offs. The areas under the respective curves (auROC) are given in the legend. (b) Precision-recall curves. The precision in dependency of the recall at varying p-value cut-offs is illustrated. The areas under the respective curves (auPRC) are given in the legend.

3.3.2. Quantifying Transcripts from Arabidopis lyrata

Comparing replicates helps in assessing the quality of RNA-seq data because, in a ideal setting, measurements from all replicates should correlate perfectly. For complementary performance analysis, rQuant was used to estimate transcript abundances for the set of annotated transcripts at 32,377 gene loci from the *A. lyrata* RNA-seq alignments for each replicate separately (cf. Section 3.2.5). First, rQuant was applied without profile estimation and the resulting abundance estimates were compared by calculating Pearson's correlation coefficient between estimated abundances of the two replicates across genes, measuring a correlation of 0.787. The PCC per gene was not determined, since there were only a few gene loci with multiple transcripts (290 genes, 0.9 %) in the data set. Empirically estimating profiles and thus correcting for transcript length biases, the degree of similarity of the quantification results could be improved (PCC across genes of 0.942). The considerable improvement with bias correction suggested that the RNA-seq data set for *A. lyrata* harboured substantial experimental biases. This example for an experimental data set showed that bias correction by the rQuant approach in deed improves quantification estimates and is essential for RNA-seq data.

3.3.3. Application of rQuant in the RGASP Competition

An early implementation of rQuant was applied in an international competition, the RNA-seq Genome Annotation Assessment Project (RGASP) [7]. Similar to previous annotation projects for the human genome sequence (EGASP [73, 74]) and nematode genome sequences (nGASP [42]), the aim of RGASP was to identify and quantify transcripts from RNA-seq data. In total, 17 research teams participated in this competition to analyse diverse RNA-seq data sets measuring mRNA transcripts in human cell lines, cell lines from *Drosophila melanogaster* and samples from *Caenorhabditis elegans* at several developmental stages. Our research group contributed to this competition by combining read alignment (PALMapper [50, 98]), transcript identification (mTiM [69] and mGene.ngs [17, 64]) and transcript quantification (rQuant). This pipeline is now also available as the web service Oqtans (cf. Section 5.2.3).

At the time of RGASP, rQuant was implemented differently than in today's version. The main difference was that the optimisation problem with respect to the transcript abundance variables was solved separately from profile estimation. Thus, both optimisation problems were convex and could be solved with standard solvers such as the commercial solver CPLEX [92]. These two steps were altered until a stable solution was obtained. However, the problem here was that this algorithm did not converge in practice. Therefore, as described in this chapter, rQuant was re-implemented in a way that the optimisation problem with respect to all optimisation variables, i.e., transcript, profile and sequence weights, is now solved at once by coordinate descent. Moreover, the early version of rQuant did not contain a sequence bias model for read density estimation and did not use information from paired-end reads.

Within the scope of RGASP, biochemical quantification experiments based on the recently developed method NanoString [65] were conducted for a subset of transcripts for the human and worm data sets. NanoString measures the abundance of transcripts by counting single mRNA molecules using reporter probes that specifically bind to a transcript of interest and that are tagged by a fluorescent bar code.

rQuant quantified transcripts from the human HG19 annotation [2] based on RNA-seq measurements for three human data sets and transcripts from the *C. elegans* WormBase 200 annotation [78] based on RNA-seq data for one developmental stage (for detail description

Table 3.2.: Evaluation on NanoString data. Pearson's correlation between abundance estimations of rQuant and NanoString measurements are shown for all transcripts targeted by the NanoString experiment and for alternative and non-alternative transcripts separately. The size of the evaluation set is indicated in brackets. *C. elegans* L3: poly-adenylated total RNA from *C. elegans* in L3 phase; human HepG2: poly-adenylated total RNA from HepG2 (human liver carcinoma cell line); human GM12878: poly-adenylated total RNA from GM12878 (human lymphoblastoid cell line); human K562: poly-adenylated total RNA from K562 (human chronic myelogenous leukemia cell line). All RNA-seq experiments were conducted with paired-end Illumina sequencing.

	C. elegans L3		human HepG2		human GM12878		human K562	
All	0.700	[145]	0.753	[141]	0.757	[141]	0.808	[141]
Alternative	0.854	[45]	0.666	[94]	0.794	[94]	0.887	[94]
Non-alternative	0.648	[100]	0.825	[47]	0.706	[47]	0.619	[47]

see Table 3.2). The abundance estimates were compared against NanoString counts for probes measuring the abundance of 145 human and 141 *C. elegans* transcripts. Note that the NanoString measurements were one of the early results using this approach and might be error-prone. Thus, conclusions from these observations should be handled with care. rQuant abundance estimates reasonably correlated with NanoString counts (0.700 to 0.808, cf. Table 3.2). Another observation was that alternative transcripts could be quantified more accurately.

3.4. Conclusion

In this chapter, I presented the method rQuant to accurately infer abundances of alternative transcripts from RNA-seq data based on solving a task-specific optimisation problem. Besides quantification, rQuant allows to estimate read density to normalise for biases that are inherent in the read data and dependent on, for example, transcript length and sequence content.

In contrast to other quantification programmes, rQuant approaches the transcript quantification problem by solving an optimisation problem that uses ideas from the lasso approach. Another key feature and difference is that rQuant uses positional read coverage rather than read counts within exons. Moreover, rQuant was the first tool that has addressed to model biases from experimental settings during transcript quantification.

For artificial data sets as well as for experimental data, I showed that modelling biases is crucial for quantification accuracy and that rQuant is superior to other proposed quantification tools in terms of correlation between simulated and inferred abundances. Moreover, exploiting information from paired-end reads and across different experimental conditions can help to improve quantification results.

In the current implementation of rQuant, repetitive positions were ignored during quantification, reads generated from repetitive regions are ambiguous and thus not reliable. Another approach to handle ambiguous reads could be to consider all regions to which reads map multiple times at once and estimate abundances with respect to these regions. Another limitation of rQuant is that it is dependent on a set of annotated transcripts. However, annotations are often incomplete and thus novel transcripts cannot be quantified. Transcripts identified by

computational methods such as mGene.ngs [17, 64] and mTiM [69] can be used to complement annotations. Here, rQuant allows to prune not expressed transcripts, as it provides a sparse quantification solution. Approaches that simultaneously identify and quantify transcripts are a straightforward alternative to address this problem. A method that has currently been developed, called SplAdder, extends ideas from rQuant using mixed integer programming [167]. It identifies and quantifies transcripts from a splicegraph, i.e., a graph with exons as nodes and splice connections as edges.

As rQuant is general framework to quantitatively analyse sequence data, it is also applicable to other areas than transcript quantification. In Chapter 4, I show the extension of rQuant to read data profiling RNA secondary structure. Moreover, another promising application could be to use rQuant for peptide quantification based on peptide sequences from high-throughput tandem mass spectrometry.

rQuant and its extensions are important tools to understand the complete transcriptome, providing highly reliable transcript abundance estimates to compare transcripts at a quantitative level. Profiling complete transcriptomes allows us to better study the underlying biological processes involved in regulation of gene expression and RNA processing, a crucial step towards a comprehensive understanding of the central dogma of molecular biology.

4. sQuant: Generalisation of rQuant to RNA Structure Quantification

4.1. Introduction

Secondary structures of RNA transcripts play an important role in basic cellular processes, ranging from aiding in protein synthesis as tRNA to their key role in regulation of gene expression and splicing. One prominent example for the importance of RNA secondary structure is the regulation of the *trp* operon in bacteria. Here, the expression of the enzymes encoded in this operon is repressed in presence of tryptophan due to a formed stem loop in a certain part of the transcript [127, p. 116]. Recently, it has been found for a set of examples that structure functions in the regulation of alternative splicing [101, 138]. For example in *Arabidopsis thaliana*, the thiamine pyrophosphate (TPP) riboswitch mediates the formation of an unstable mRNA at the *THIC* gene locus when TPP is present [209].

For the understanding of their function and their mechanism of action, it is therefore crucial to know the secondary structure of RNA molecules. Until recently, measuring RNA structure has been limited by the low-throughput nature of existing biochemical methods and the difficulties to conduct the experiments *in vivo*. Therefore, computational prediction of RNA secondary structure from the sequence has been an orthogonal approach, stating one of the early problems of computational biology. Most existing methods that predict structure from the sequence minimise the free energy of the folded RNA molecule.

The recent development of high-throughput experimental methods has opened new potentials to improve computational prediction of RNA secondary structure [106, 203]. The outcome of such experiments can give evidence where paired and unpaired nucleotides are located in the RNA transcript. Kertesz et al. [106] proposed a method called PARS (parallel analysis of RNA structure) to profile RNA structure by a combination of enzyme digestion and high-throughput sequencing (cf. Figure 4.1). Here, folded RNA is *in vitro* digested by enzymes in two separate experiments, one enzyme (S1 nuclease) that preferentially cuts single-stranded RNA, the second enzyme (RNase V1) used to digest double-stranded RNA. The RNA fragments are further sheared, converted to cDNA and selected by size as in standard RNA-seq library protocols. However, the 5' sequencing adaptor is ligated to cutting sites only, thus ensuring that read starts measure either single-stranded or double-stranded states. The FragSeq protocol developed by Underwood et al. [203] uses similar ideas, but focuses on one single experiment to measure cleavage of single-stranded RNA by the enzyme nuclease P1.

Often, not only a single structure is folded from an RNA transcript, but different structures of the same transcript may co-exist in a cell. Read counts from a combined digestion and sequencing experiment then reflect a mixture of the underlying alternative structures, making it difficult to simply read-off the structure from the observed read counts. The problem of structure quantification from this kind of read data is addressed in this chapter. Adopting ideas from the rQuant approach for transcript quantification, a method called sQuant was developed that infers abundances of a mixture of RNA secondary structures from PARS-like experimental data. Furthermore, similarly to biases due to RNA-seq library preparation,

distortions of read density induced by the PARS library protocol are identified and addressed by a bias model to normalise read counts. Details of the proposed optimisation problem are described in Section 4.2. Results based on an artificially generated data set and based on one of the few available experimental data sets are presented in Section 4.3, showing the accuracy of the approach and that structural information can also help in transcript quantification. Moreover, test experiments suggested that structure quantification may improve computational inference of several secondary structures of one RNA sequence by combining sQuant with *de novo* structure prediction.

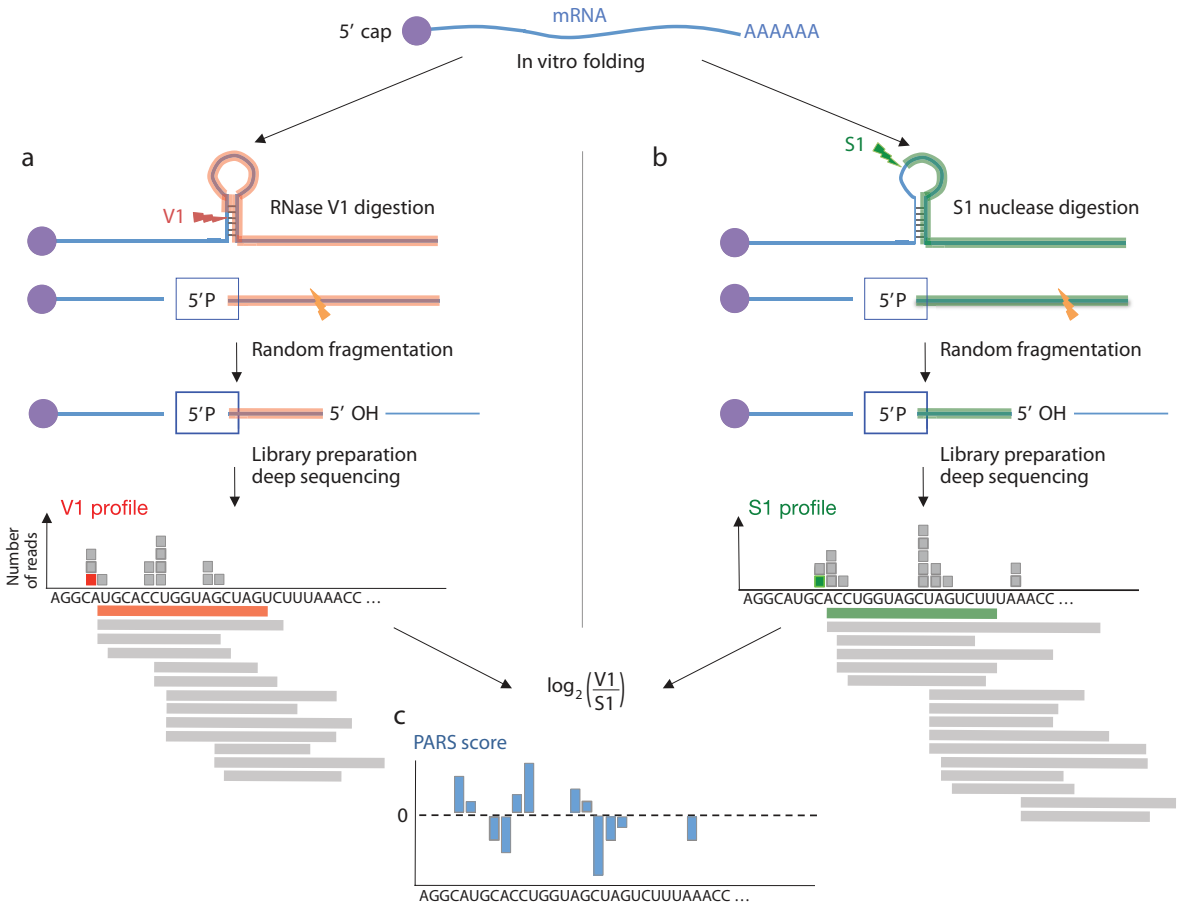


Figure 4.1.: PARS (parallel analysis of RNA structure) protocol. To profile RNA structure, folded RNA is *in vitro* digested by enzymes in two independent experiments. In the first experiment (a), the RNA is digested by RNase V1 that preferentially cuts double-stranded RNA (indicated by a red arrow). A second enzyme, S1 nuclease, shown in (b) is used to digest single-stranded RNA (indicated by a green arrow). As in standard RNA-seq library protocols, the RNA fragments are further sheared, converted to cDNA and selected by size. However, the 5' sequencing adaptor is ligated to cutting sites only, thus ensuring that read starts measure either single-stranded or double-stranded states (see V1 and S1 profiles). (c) Kertesz et al. [106] used the log ratio of V1 and S1 read counts (PARS score) to determine single-stranded or double-stranded state of a nucleotide. Reprinted by permission from Macmillan Publishers Ltd: Nature [106], copyright (2010).

4.1.1. Related Work

To the best of my knowledge, there have been no publications describing neither the inference of transcript abundances based on RNA secondary structure measurements nor a combined approach for RNA secondary structure prediction and quantification.

4.1.2. Publication Note

sQuant was joint work with Gunnar Rätsch and to a minor extend with Fabio De Bona. Regina Bohnert and Gunnar Rätsch conceived and designed the project. Regina Bohnert implemented sQuant and performed experiments for data simulation and evaluation of the method with input from Gunnar Rätsch. Gunnar Rätsch with input from Fabio De Bona performed experiments for *de novo* RNA structure prediction. sQuant was presented at the HiTSeq conference on High-throughput Sequencing Analysis and Algorithms preceding the ISMB/ECCB Conference 2011.

4.2. Methods

4.2.1. An Optimisation Problem Formulation for Solving the Structure Quantification Problem

Assuming that the set of alternative structures is known, the problem of quantifying alternative structures from read counts can be formulated as a mathematical programme. This idea was implemented in the tool sQuant, adopting concepts and experience from rQuant for solving the problem of transcript quantification (cf. Chapter 3). In the approach of sQuant, abundances of the structures are estimated by minimising the deviation of expected and observed counts of read starts from double-stranded and single-stranded measurements, respectively.

Quantifying Alternative Structures

Formally, the objective Ω of the optimisation problem consisted of the weighted sum of loss terms for double-stranded and single-stranded counts of read starts ($\mathcal{L}_{ds}$ and $\mathcal{L}_{ss}$, respectively) and a regularisation term $\mathcal{R}$:

$$\begin{aligned}\Omega(\boldsymbol{w}) &= \mathcal{L}_{ds} + \mathcal{L}_{ss} + \mathcal{R}(\boldsymbol{w}) \\ &= \gamma^{ds} \sum_{p=1}^{P} \ell\left(c_p^{ds}, \sum_{s=1}^{S} w_s \ \delta_{p,s}\right) + \gamma^{ss} \sum_{p=1}^{P} \ell\left(c_p^{ss}, \sum_{s=1}^{S} w_s \ \overline{\delta_{p,s}}\right) + \mathcal{R}(\boldsymbol{w})\end{aligned} \tag{4.1}$$

where $\boldsymbol{w} = [w_1, \ldots, w_S]$ are the optimisation variables, i.e., the abundances of the given structures. S denotes the number of structures, P is the number of exonic positions and γ^{ds} and γ^{ss} are parameters to weight the loss terms. Information about which structures included position p is stored in the structure mask $\delta_{p,s}$. This mask is 1 if position p of structure s is double-stranded, and 0 otherwise. For the single-stranded case, the complement of $\delta_{p,s}$, $\overline{\delta_{p,s}}$, is taken to define the structure mask. Moreover, c_p^{ds} and c_p^{ss} are the observed double-stranded and single-stranded counts of read starts, respectively. Note that the given structures are

not necessarily alternative structures of the same transcript, but can be multiple structures formed from a set of alternative transcripts at one gene locus.

The deviation of estimated and observed read counts was penalised by a squared loss, i.e., $\ell(x, y) = (x - y)^2$. Moreover, the abundance variables $\boldsymbol{w}$ were regularised by an ℓ_1-norm to provide a sparse solution [20]:

$$\mathcal{R}(\boldsymbol{w}) = \sum_{s=1}^{S} \gamma_s^w \ |w_s|$$

with γ_s^w a structure-specific regularisation weight. Therefore, such an approach is highly suitable to prune a set of alternative structures, resulting in a small set of structures consistent with the read data.

Given these definitions, the optimisation problem of sQuant could be formulated as:

$$\begin{aligned} \underset{\boldsymbol{w}}{\text{minimise}}\ \Omega(\boldsymbol{w}) = \gamma^{ds} \sum_{p=1}^{P} \ell\left(c_p^{ds}, \sum_{s=1}^{S} w_s \ \delta_{p,s}\right) + \gamma^{ss} \sum_{p=1}^{P} \ell\left(c_p^{ss}, \sum_{s=1}^{S} w_s \ \overline{\delta_{p,s}}\right) + \sum_{s=1}^{S} \gamma_s^w \ w_s \\ \text{subject to} \qquad w_s \geq 0, \ \forall \ s = 1, \ldots, S. \end{aligned} \tag{4.2}$$

Estimating Read Density to Model Biases

Kertesz et al. [106] stated in their evaluation of the PARS approach that they did not observe significant biases in the read data. In contrast to this, we found in our own analysis based on the read data described in their paper and a simulation study mimicking steps of the PARS protocol that several steps of the library preparation may cause a non-uniform distribution of read counts. Key observations were that cutting frequency of the enzyme, fragmentation frequency and the range of the selected fragment sizes were crucial factors influencing read density along the structure. Examples for these biases are visualised in Figure 4.2. Moreover, the order, location and distance of nucleotides in single-stranded or double-stranded state influence the size of the digestion fragments between two cutting sites. The higher the cutting frequency of the enzyme, the higher the amount of fragments and the shorter their sizes. Shorter fragments are also produced when the fragmentation process is performed for a longer time. Selecting fragments of a certain size then favours those fragments generated from cutting sites (including the transcript ends) with a particular distance.

To account for theses biases, a model based on Ridge regression (cf. [82] and Section 1.2.2) for estimating read density was formulated. In the regression model, features $\boldsymbol{x}$ derived from the structures were related to target values $\boldsymbol{y}$ based on read counts.

Motivated by the observations mentioned above, the features $\boldsymbol{x}$ used in regression were based on the structure of the transcript and the distances between cutting sites, i.e., between paired and unpaired nucleotides, respectively. They were derived from the structures by calculating histograms of distances to proximal cutting sites, i.e., distances to the closest positions in either double-stranded or single-stranded state, upstream and downstream of each position p. For the experiments presented in this chapter, the ten closest neighbours upstream and downstream to the considered nucleotide were considered and the histogram of the distances to these neighbours were modelled with ten bins for distances between 1 and 50 nt.

The target values were chosen as the ratio of observed counts of read starts and expected counts of reads, which was set to the average count for the structure. The reason for this

choice was to model the deviation of observed from expected read counts in terms of features capturing bias-inducing properties of the structure.

In the Ridge regression model, the target values $\boldsymbol{y}$ could be expressed as a function of features $\boldsymbol{x}$ and regression parameters $\boldsymbol{\beta}$. Then, the optimal $\boldsymbol{\beta}^*$ was determined on a set of size M with $(\boldsymbol{x}^m, y^m)$-pairs, $m = 1, \ldots, M$, by solving the following optimisation problem for Ridge regression:

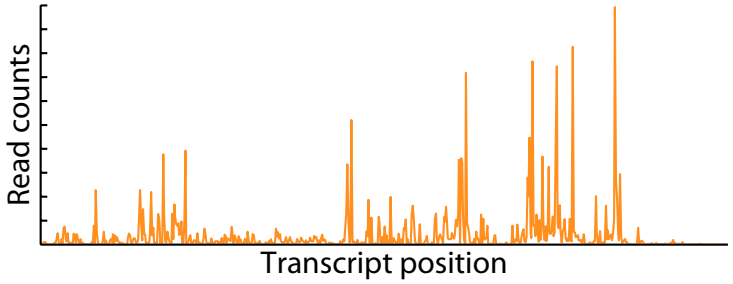
$$\underset{\boldsymbol{\beta}}{\text{minimise}} \sum_{m=1}^{M} \left(\boldsymbol{\beta}^T \boldsymbol{x}_m - y_m\right)^2 + \lambda \|\boldsymbol{\beta}\|_2^2$$

The expected number of read starts, the read density $D_{p,s}^{ds}$ for double-stranded measurements, was then estimated based on the trained regression model, i.e., the optimal $\boldsymbol{\beta}^{ds*}$, by computing

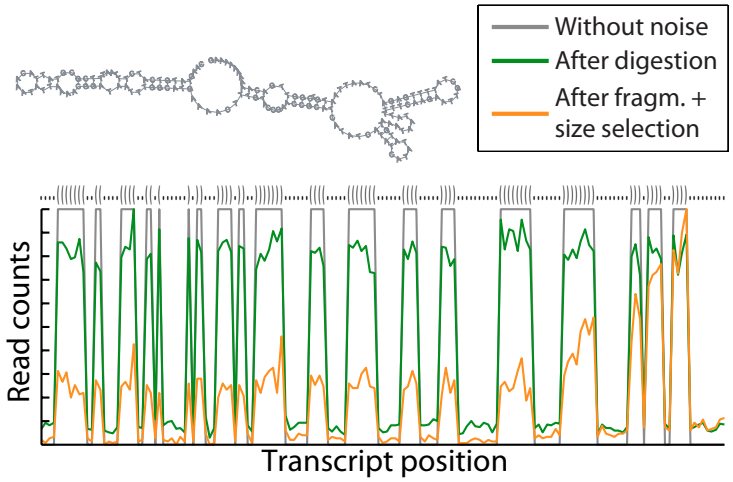
$$D_{p,s}^{ds} = \hat{y}_{p,s} \; \overline{c_p^{ds}} = \boldsymbol{x}_{p,s}^T \; \boldsymbol{\beta}^{ds*}$$

where $\overline{c_p^{ds}}$ is the average count of read starts at position p.

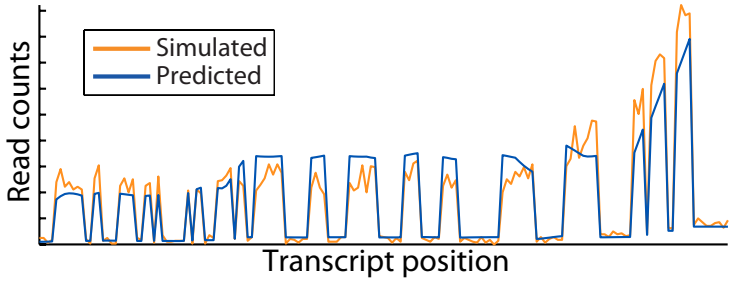
For single-stranded measurements, $D_{p,s}^{ss}$ was determined analogously to the calculation of $D_{p,s}^{ds}$. To use the bias model in quantification, the structure masks $\delta_{p,s}$ and $\overline{\delta_{p,s}}$ in the objective Ω were replaced by $D_{p,s}^{ds}$ and $D_{p,s}^{ss}$, respectively.



(a) Biases in experimental data.



(b) Biases induced by several steps of library preparation for the structure displayed above.



(c) Read density predicted with Ridge regression.

Figure 4.2.: Example for biases induced by PARS library preparation. All illustrations are for double-stranded experiments only. (a) Counts of read starts are shown in dependency of their location in the *S. cerevisiae* transcript YBL027W for the yeast read data set from [106]. Read counts increase towards the 3' boundary of the transcript. (b) Counts of read starts were simulated for the *A. thaliana* transcript AT2TE24140 according to the strategy described in Section 4.2.3. The predicted structure of the transcript is illustrated as a graph at the top as well as in bracket format aligned to the counts. Abundance of the structure can directly be read from counts generated in an ideal setting (grey). After enzyme digestion, counts are reduced and unevenly distributed (green). Here, counts at single-stranded nucleotides can be observed due to unspecific cutting by the enzyme. After the fragmentation and size selection steps, counts decreased again and relatively more reads now fall into 3' region of the transcript. (c) The number of read starts expected according to the transcript structure were predicted with a trained Ridge regression model (cf. Section 4.3.1) from structural information (blue). The comparison to true counts (orange) shows that the prediction of read counts from structure is feasible.

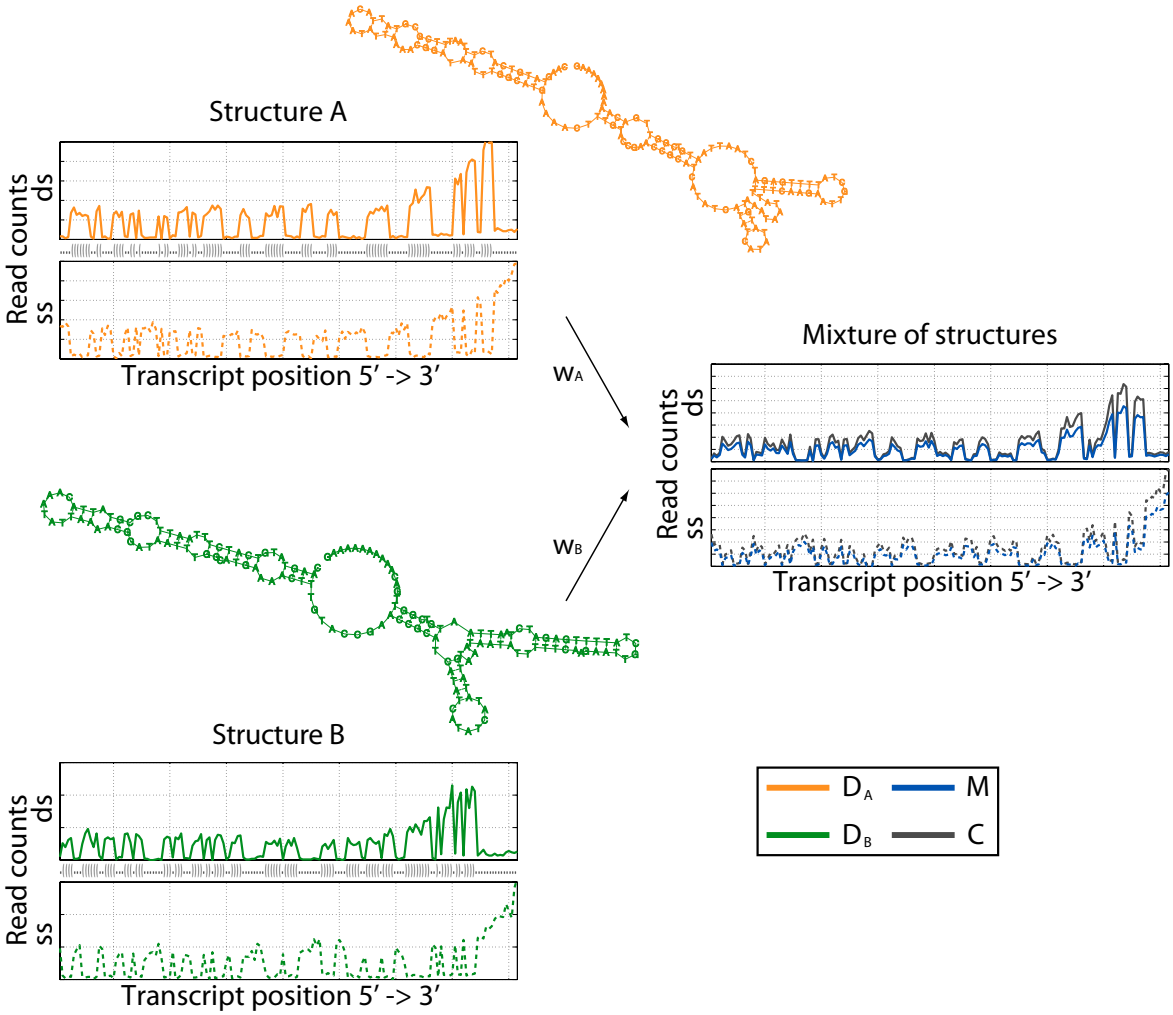


Figure 4.3.: Basic mode of action of sQuant. Double-stranded (ds) and single-stranded (ss) counts of read starts are displayed for two alternative structures A (orange) and B (green) of the same transcript together with the formed structures (illustration on the left). sQuant determines w_A and w_B by minimising the deviation of observed (grey) and estimated (blue) counts of read starts (illustration on the right). The estimated counts of read starts M^{ds} and M^{ss} are determined from the structure-dependent read densities D^{ds} and D^{ss} and the structure abundance w_A and w_B by $M^{ds} = w_A D_A^{ds} + w_B D_B^{ds}$ and $M^{ss} = w_A D_A^{ss} + w_B D_B^{ss}$.

4.2.2. The sQuant Algorithm

Similar to the rQuant algorithm described in Section 3.2.2, the optimal structure weights $\boldsymbol{w}^*$ were determined by a coordinate descent approach (cf. Section 1.2.2 and [34, Chapter 9]). The optimisation problem was solved with respect to one coordinate, i.e., one structure weight $w_{s'}$, until convergence to the optimal solution of $\boldsymbol{w}$ at a given precision (cf. [201, and references therein] and the following section). The sQuant algorithm is formally described in Algorithm 4.1. Briefly, read density parameters were first estimated on a training set of structures and then abundances were estimated for structures at each gene locus.

Finding the Optimal Transcript Weights

The objective Ω from the optimisation problem 4.2 was reformulated in terms of one variable $w_{s'}$:

$$\begin{aligned}
\Omega(w_{s'}) &= \gamma^{ds} \sum_{p=1}^{P} \left(w_{s'}\ \delta_{p,s'} + \underbrace{\sum_{s \neq s'} w_s\ \delta_{p,s}}_{R_1} - c_p^{ds} \right)^2 + \gamma^{ss} \sum_{p=1}^{P} \left(w_{s'}\ \delta_{p,s'} + \underbrace{\sum_{s \neq s'} w_s\ \overline{\delta_{p,s}}}_{R_2} - c_p^{ss} \right)^2 \\
&\quad + \gamma_{s'}^{w}\ w_{s'} + \sum_{s \neq s'} \gamma_s^w\ w_s \\
&= \gamma^{ds} \sum_{p=1}^{P} \left({w_{s'}}^2\ {\delta_{p,s'}}^2 + 2\ w_{s'}\ \delta_{p,s'}\ R_1 + {R_1}^2 \right) + \gamma^{ss} \sum_{p=1}^{P} \left({w_{s'}}^2\ \overline{\delta_{p,s'}}^2 + 2\ w_{s'}\ \overline{\delta_{p,s'}}\ R_2 + {R_2}^2 \right) \\
&\quad + \gamma_{s'}^{w}\ w_{s'} + \sum_{s \neq s'} \gamma_s^w\ w_s \\
&= {w_{s'}}^2 \underbrace{\left(\gamma^{ds} \sum_{p=1}^{P} {\delta_{p,s'}}^2 + \gamma^{ss} \sum_{p=1}^{P} \overline{\delta_{p,s'}}^2 \right)}_{S_1} \\
&\quad + w_{s'} \underbrace{\left(2\ \gamma^{ds} \sum_{p=1}^{P} w_{s'}\ \delta_{p,s'}\ R_1 + 2\ \gamma^{ss} \sum_{p=1}^{P} w_{s'}\ \overline{\delta_{p,s'}}\ R_2 + \gamma_{s'}^{w} \right)}_{S_2} \\
&\quad + \underbrace{\gamma^{ds} \sum_{p=1}^{P} {R_1}^2 + \gamma^{ss} \sum_{p=1}^{P} {R_2}^2 + \sum_{s \neq s'} \gamma_s^w\ w_s}_{S_3}
\end{aligned}$$

The globally minimal $w_{s'}^*$ could be calculated by setting the first derivative to 0 and solving for $w_{s'}$:

$$\begin{aligned}
\frac{d\Omega}{dw_{s'}} = 2\ S_1\ w_{s'} + S_2 = 0 \\
\Rightarrow w_{s'}^* = \frac{-S_2}{2\ S_1}
\end{aligned}$$

The sufficient condition for the global minimum $\frac{d\Omega}{dw_{s'}{}^2} = 2\ S_1 \geq 0$ is fulfilled as S_1 is always non-negative. The structure weights were required to be non-negative, thus $w^*_{s'}$ was clipped to 0 if negative:

$$w^{clipped}_{s'} = \begin{cases} w^*_{s'} & w^*_{s'} \geq 0 \\ 0 & w^*_{s'} < 0 \end{cases} . \tag{4.3}$$

As discussed in Section 3.2.2, clipping to 0 leads to the optimal solution over $[0, \infty)$.

Estimating Read Density

For the estimation of the read density, a training set of feature-target pairs derived from a subset of structures and read counts was assembled. For each nucleotide of a structure in this set, the feature vector based on distances to proximal cuttings sites was calculated (cf. Section 4.2.1). The respective target values were derived from the read counts as described in Section 4.2.1. The optimal parameter vectors $\boldsymbol{\beta}^{ds*}$ and $\boldsymbol{\beta}^{ss*}$ were calculated on this training set by using the analytical solution of the Ridge regression optimisation problem for double-stranded and single-stranded measurements separately (cf. [82] and Section 1.2.2):

$$\boldsymbol{\beta}^{ds*} = \left(\sum_{m=1}^{M^{ds}} \boldsymbol{x}^{ds}_m \boldsymbol{x}^{ds\,T}_m + \lambda I \right)^{-1} \sum_{m=1}^{M^{ds}} y^{ds}_m \boldsymbol{x}^{ds}_m \text{ with } \lambda > 0. \tag{4.4}$$

$\boldsymbol{\beta}^{ss*}$ was calculated analogously. The read densities $D^{ds}_{p,s}$ and $D^{ss}_{p,s}$ were determined as described in Section 4.2.1.

Algorithm 4.1 The algorithm of sQuant

```
procedure SQUANT(genes, aligned reads^ds, aligned reads^ss)
    β^ds ← OPT_DENSITY(genes^train, aligned reads^ds)
    β^ss ← OPT_DENSITY(genes^train, aligned reads^ss)
    for all g ∈ genes do
        # Finding the optimal structure weights for gene g
        w_s ← mean coverage at locus of s, ∀ s, ..., S
        repeat
            w^old ← w
            for s = 1, ..., S do
                w_s ← argmin_{w_s≥0} Ω(w_s)                    # Cf. Equation 4.3
            end for
        until ||w − w^old|| < ε
    end for
    return w^genes, β^ds, β^ss
end procedure

procedure OPT_DENSITY(genes, aligned reads)
    β ← RIDGE_REGRESSION(genes, aligned reads)                 # Cf. Equation 4.4
    return β
end procedure
```

4.2.3. Data Simulation

To evaluate the accuracy of sQuant, a set of simulated read counts was assembled by mimicking steps of the PARS library protocol [106]. Based on the *Arabidopsis thaliana* TAIR10 annotation [195], 502 single-transcript loci as well as 498 loci harbouring alternative transcripts were randomly selected. Loci with transcripts longer than 2,000 nt were not considered. In total, there were 1,659 transcripts; 78 % of the multiple-transcript loci had two, 15 % had three and 7 % had least four transcripts. For each of the transcripts, one to ten alternative RNA secondary structures were predicted using the tool ViennaRNA RNAsubopt [83]. Abundances were assigned to each structure by uniformly sampling values from the range 0 to 10,000.

To obtain non-uniform read density with authentic settings, the steps of the PARS library preparation were simulated. For enzyme digestion, the number of enzyme cleavage events e_n per transcript copy n was sampled from a Poisson distribution with a mean of five events. Then, each copy was cut at e_n sites preferred by the respective enzyme. To model that the enzyme misses or cuts at wrong sites, 5 % false negatives and 8 % false positives were allowed for the enzyme cutting double-stranded RNA and 5 % false negatives and 10 % false positives for the enzyme cutting unpaired nucleotides. These rates were set according to realistic error rates of the enzymes RNase V1 and S1 nuclease that were used in the PARS protocol. The next step in library preparation was fragmentation. This was simulated by cutting the digested fragments at a certain nucleotide with a probability of 10^{-3} in three fragmentation iterations. Only fragments of which the 5' end originated from digestion were used for the third step. Here, fragments between 50 and 200 nt in length were selected to simulate the size selection step on the gel. Counts of read starts were calculated from this final set of fragments. This strategy was applied for the two enzymes separately.

Additionally, RNA-seq reads were simulated for all structures by using the same simulation pipeline as described above with the following adaptions. The enzyme digestion step was omitted and fragments were not selected by cutting site.

4.2.4. Preparation of Yeast Data Set

Data for *Saccharomyces cerevisiae* (budding yeast) provided by Kertesz et al. [106] was downloaded from `http://genie.weizmann.ac.il/pubs/PARS10/pars10_catalogs.html`, including read counts for the double-stranded and single-stranded experiments and 3,204 annotated transcripts. Moreover, structures that were derived with PARS-assisted folding were stored for each reported transcript (3,199 in total), which are denoted as 'PARS structures' from now on. In addition, for each transcript in the set that was shorter than 3,000 nt (3,008 transcripts), seven structures were predicted with the tool ViennaRNA RNAsubopt [83], independent of the read counts.

4.3. Results and Discussion

4.3.1. Artificial Data Set

The accuracy of sQuant was assessed on the artificial data set described in Section 4.2.3. As evaluation measure, Pearson's correlation coefficient (cf. Section C.1.2 in the Appendix) of estimated and true relative abundance within a gene locus was determined and then averaged over all loci with more than one structure.

Mixture of Structures In a first experiment, sQuant was applied to quantify known structures of 446 single-transcript genes. The correlation of 0.909 showed that RNA secondary structure quantification using sQuant was feasible. Furthermore, the correlation was determined in dependency of the number of transcript structures. For two to four structures, correlation was at a very high level (0.960). The correlation slightly dropped to 0.902 for five to seven structures and was considerably lower for eight to ten structures (0.862). This illustrated that the task of structure quantification became more challenging when the number of structures increased. This result was consistent to observations for rQuant for which the accuracy was also dependent on the complexity of the gene and decreased for a higher number of transcripts (cf. Section 3.3.1).

Mixture of Structures and Transcript Isoforms sQuant was further used to quantify a mixture of structures and transcript isoforms at 498 multiple-transcript loci, which was a more realistic setting. A correlation of 0.898 was calculated for this case, which was at the same magnitude as for the scenario of a mixture of structures from one isoform. The effect of the number of structures at one loci was measured by assessing the correlation for bins of two to ten (0.938), eleven to 14 (0.889) and more than 15 structure per loci (0.865). This analysis confirmed that quantification accuracy was dependent on the number of structures.

Advantage of Bias Correction In another experiment, the effect of normalisation for library biases on structure quantification was analysed. The Ridge regression model described in Section 4.2.1 was trained on 5,908 examples from ten gene loci (15 transcripts with 86 structures) for double-stranded and single-stranded read data separately. The ratio of the average squared deviation of true and predicted read counts $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ and the average squared deviation of true target values to their mean was assessed on both on the training as well the evaluation set (1,477 examples). A value below 1 for this quotient reflected reduced variability within the set of read counts. For the chosen settings, the model explained ≈ 80 % of the variability. Other biases such as positional and transcript length were likely to be also found in the data set, but have not been modelled by the chosen features and thus were probably the reason for the remaining variability.

To measure the effect of bias correction, a simplified set of structures was assembled, randomly selecting only one structure per isoform for 498 multiple-transcript loci. Having trained the Ridge regression model, read density was normalised according to this model during structure quantification. In terms of correlation, quantification accuracy could be improved from 0.826 (without bias correction) to 0.835 (with bias correction). The effect of bias correction was only minor, suggesting that further improvements of the model need to be undertaken, for example, by extending the set of features used for regression by features modelling transcript length biases.

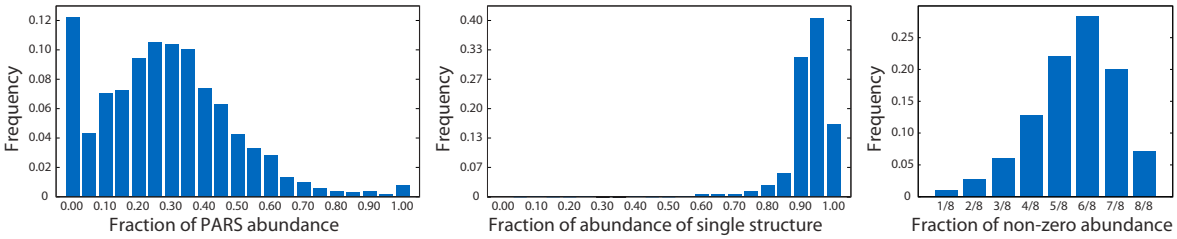
Comparison to Transcript Quantification Solving the transcript quantification problem based on RNA-seq data becomes difficult when reads are short, alternative events are far apart and exons are combined in a complex manner. Information about the RNA secondary of the transcript can help here because measurements at nucleotides that are in vicinity due to secondary structure, but distant in the primary structure may unambiguously resolve exon combinations. Therefore, sQuant applied on PARS sequencing was compared to a transcript quantification approach that uses RNA-seq read data for quantification (rQuant, cf. Chapter 3). Quantification was conducted by both methods including their respective read density model for 498 loci with alternative transcripts. Transcript abundance estimates for sQuant were calculated by adding abundance estimates for several structures of one transcript to measure accuracy on a transcript level rather than on the basis of structures. Then, the correlation was assessed for sQuant estimates, which was considerably higher than the accuracy achieved by quantifying transcripts without information in the structure (0.851 to 0.649). This showed that measurements that capturing RNA structures helped in improving transcript quantification results. Thus, using a PARS-like approach, accuracy of transcript quantification may substantially be improved in comparison to RNA-seq based quantification.

4.3.2. Yeast Data Set

sQuant was applied to quantify transcripts from the yeast data set described in Section 4.2.4 based on read counts generated in the context of the PARS protocol publication Kertesz et al. [106]. The median error for quantification was 0.20, measured in deviation of the average observed read counts and the predicted sum of structure abundances normalised to the average observed read counts.

Furthermore, the fraction of the abundance assigned to the PARS structure was determined in order to measure how many of the read counts were not explained by the PARS structure. A histogram of these fractions is shown in Figure 4.4 (a). For 12 % of the transcripts, the abundance of the PARS structure was estimated to be zero, suggesting that there were structures in the provided set that better explained the observed read counts. For half of the transcripts, the relative abundance of the PARS structure was less than 28 %. The PARS structure was the major abundant structure for only a few transcripts. To confirm these observations, the same analysis was undertaken for the artificial data set described in Section 4.2.3 with the following settings: One structure per transcript was taken as the true structure and abundances were estimated for a candidate set of one to ten structures including the true structure. The histogram of the fraction of the abundance of the true structure is shown in Figure 4.4 (b). Obviously, sQuant accurately assigned the major portion of abundance to the true structure for almost all transcripts in set. Comparing this histogram to Figure 4.4 (b), the difference between these distributions is apparent. This suggested that the observation for the distribution of the fraction of PARS abundances was no artifact and that the PARS structures insufficiently explained the read count data.

In another analysis, the fraction of structures with non-zero abundance was calculated. Their distribution is illustrated in Figure 4.4 (c). For a minority of transcripts (10 %), the read data was consistent with at most three of the predicted structures. The read counts for half of the transcripts could be explained by five or six structures, suggesting that the expected number of structures for a transcript is around five.



(a) Histogram of PARS abundance. (b) Histogram of abundance of single structure. (c) Histogram of non-zero abundance.

Figure 4.4.: Histograms of abundance fractions for the yeast data set. (a) The frequency of transcripts in dependency of the fraction of the PARS structure abundance is shown. (b) The frequency of transcripts in dependency of the abundance fraction estimated for the underlying true single structure is illustrated. The histogram was determined on the simulated data set described in Section 4.2.3. One structure per transcript was chosen as the true structure and abundances were estimated for a candidate set of one to ten structures. In contrast to (a), the major portion of abundance was assigned to the correct structure for the majority of transcripts. (c) The frequency of transcripts that have one, two, three, etc. detectable structures, i.e., structures with non-zero abundance is shown.

In summary, these observations suggested that read counts from a PARS sequencing experiment were usually not sufficiently explained by one structure, but by a mixture of a few different structures. Therefore, quantification by an approach like sQuant is important to reveal present structures described by a mixture of read counts from a candidate set of structures.

4.3.3. Application of sQuant for De Novo Structure Prediction

In general, possible RNA secondary structures are only known for few examples. Together with a structure prediction programme such as ViennaRNA RNAfold, sQuant can be used to deconvolve and identify a small set of structures underlying a mixture of read measurements. For instance, in an iterative manner, a structure predictor first suggests a structure candidate, which is then added to the active set of alternative structures. This set is subsequently quantified by sQuant. The residues of observed and predicted abundance can then be used as side-information for the structure predictor in the next iteration. Preliminary results for such an approach, sQuant.denovo, from an experiment on a test set of 50 genes showed that this task was feasible for inference of two structures, but it became more challenging, i.e., less accurate, for more complex mixtures of structures (cf. Figure A.5).

4.4. Conclusion

This chapter described sQuant, an approach for quantification of mixtures of RNA secondary structures based on read data that gives evidence about the pairing state of RNA nucleotides. sQuant has been the first method that approaches structure quantification based on NGS measurements, being ahead of current developments in this area.

On an artificially generated read data set, I demonstrated the excellent performance of sQuant, very accurately resolving abundances of several structures of alternative transcripts.

Moreover, I found that non-uniformity of the read counts distribution are likely due to settings in the sequencing library preparation based on experimental read data and a simulation study, e.g., due to frequencies of enzyme digestion and fragmentation. I showed that these non-uniformity can be modelled with a regression modell to improve quantification results.

In another experiment comparing sQuant to RNA-seq based quantification for transcript quantification, I demonstrated that a quantification method such as sQuant, which takes RNA structure measurements into account, was more accurate than an approach that uses RNA-seq data only. Thus, transcript quantification based on PARS-seq may have an advantage over RNA-seq-based quantification in the future.

So far, the accuracy of sQuant could only be evaluated on artificial data, as there is a lack of complementary biochemical quantification techniques. Moreover, existing high-throughput methods are limited, as they measure RNA structure in *in vitro*. Currently, *in vivo* methods are under way that focus on fixing the native structure before RNA extraction from cells. Thus, there will be a need of accurate quantification tools such as sQuant in the near future. In particular, the combination of computational structure prediction and quantification for RNA entities will create a powerful tool to describe the set of RNA structures and their relative abundance in a cell, enabling further analyses to study RNA function and regulation in relation to the secondary structure of RNA.

5. rQuant.web: A Web Service for RNA-seq-based Transcript Quantification

5.1. Introduction

Web services offer convenient access to any kind of software installed on a server to Internet users. Researchers in Computational Biology take this opportunity to provide their own developed software as web applications to the community of experimental and computational biologists and biochemists. Every year in July, the journal Nucleic Acids Research publishes a collection of around a hundred papers on web-based software resources in the annual NAR Web Server Issue. Besides their numerous benefits, web services harbour the problem that they are not accessible or not functional any more after some period of time. This can be a major problem when using web services to obtain scientific results. The difficulties met in scientific web services has been investigated in a study by Schultheiss et al. [180] in more detail, resulting in guidelines for providing a scientific web resource [178].

To tackle these problems, researchers at the Pennsylvania State University, USA have developed a web application framework called Galaxy, which has been implemented to perform computational analyses of genomic data [26, 66, 67]. Galaxy addresses the need of accessibility, reproducibility and transparency of scientific web services [67], and aims towards bringing together users and software developers in one framework. Computational tools provided in a Galaxy instance are *accessible* via the Internet in a web-based interface to users without extensive computational background. The usage of tools is demonstrated and explained in on-line tutorials. Users can easily import and upload external data sets and may also have access to integrated data sources. Input and output data are automatically saved in a history, supporting persistent workspaces that can be re-used for repeated analysis. By not only storing data sets, but also used tools and parameter values together with the applied order of the used tools, the functionality of histories and work flows makes an analysis with Galaxy *reproducible*. Galaxy's sharing model implements the idea of *transparent* scientific experiments by allowing to share histories and work flows via web links. From a developer's point of view, Galaxy facilitates the integration of tools with the single requirement to be callable from the command line. Whilst providing an instance at `http://main.g2.bx.psu.edu`, Galaxy has recently also be offered in cloud computing infrastructures such as the Amazon Elastic Compute Cloud (EC2). Nowadays, there exist numerous local instances around the world, a part of them dedicated to internal use only.

The research group "Machine Learning in Biology" (MLB) at the Friedrich Miescher Laboratory in Tübingen (`http://www.fml.mpg.de/raetsch`) provides a Galaxy instance at `http://galaxy.fml.mpg.de/`, which integrates a collection of tools that have been developed in the group, and which are mostly based on methods from machine learning. Besides tools integrated and developed by the main Galaxy developers, it contains

- an SVM toolbox [19],
- the gene finding system mGene.web [181],
- KIRMES, a tool for kernel-based identification of regulatory sequence modules [179],

- a GFF Toolkit, and
- the Oqtans toolbox, which combines RNA-seq analysis tools such as PALMapper [50, 98], mTiM [69], rQuant [28, 30], and rDiff [193].

In this chapter, I present the integration of rQuant in the MLB Galaxy instance as the tool rQuant.web (Section 5.2), and the rQuant release as a standalone, free and open-source software package (Section 5.3).

5.1.1. Publication Note

The web service rQuant.web was joint work with Gunnar Rätsch and Vipin Sreedharan. Regina Bohnert with contributions from Gunnar Rätsch designed the rQuant.web software and performed experiments. Regina Bohnert with help from Vipin Sreedharan integrated rQuant.web into the MLB Galaxy instance. Some material of this chapter was published in Bohnert and Rätsch [28]. As part of the Oqtans web service, it was presented at the Biology of Genomes Meeting 2010, at the Special Interest Group on High-Throughput Sequencing Analysis and Algorithms at the ISMB Conference 2010, the Galaxy Community Conference 2011, at the ISCB Student Council Symposium 2011, and at the ISMB/ECCB Conference 2011.

5.2. Usage of rQuant.web

5.2.1. Modules

rQuant.web currently consists of the three main components: data preparation, quantification, and bias estimation, which are described in more detail below.

Data preparation

As a first step when using rQuant.web, one starts with uploading a set of transcripts either in GFF3 or AGS format, and the alignments of reads from an RNA-seq experiment in the compressed BAM format [121]. These formats are described in detail in Section 5.3.6. Data can be uploaded using `Get Data` → `Upload File`. For the upload, either the `Browse` button can be used, or the URL to a file stored on, for example, an FTP server can be pasted, which is particularly suitable for larger files (necessary for files $\geq$ 2 GB).

Read alignments can also be uploaded in uncompressed alignment format (SAM) and then be converted to BAM format by applying the tool `SAM-to-BAM`, which is located in the tool section `NGS: SAM Tools` and which uses the SAMTools toolbox [121]. Taking the aligned read data in the commonly used SAM/BAM format, rQuant.web is applicable to read data from different NGS platforms, e.g., Illumina's GA or SOLiD. The bias model estimation is motivated by the observations based on Illumina read data and cDNA library preparation protocols used for this platform. However, similar observations have been made for other platforms when using similar library preparation protocols. Alternatively, raw reads can be uploaded in FASTQ format and can then be aligned to the reference genome by applying read mapping tools also provided within the Galaxy framework (cf. tool section `NGS: Mapping`, based on the Galaxy NGS Toolbox, and `NGS: QPALMA Tools` [50, 98]).

Internally, annotated transcripts provided in GFF format are prepared and converted to the

internally used AGS object. Transcripts that fall in the same genomic locus are considered as one set of transcripts for quantification, even if they are annotated to separate genes in the input annotation. The annotation in AGS format is provided as an output file, facilitating and accelerating analysis with the same set of transcripts with rQuant.web and other tools within the Galaxy service.

Before performing the actual quantification, the user has the option to perform a sanity check of the input data with the tool `ReadStats` by checking the uploaded alignments and the annotated transcripts for consistency. The tool generates statistics about the input files. It displays the number of reads identified in the given annotation, the median read coverage per gene, the number of spliced reads, and the number of spliced reads overlapping annotated introns.

Quantification

With the uploaded inputs, the core `rQuant` component determines the abundance of each transcript in the given annotation. When not using read density estimates, this tool does not have any parameters that need to be specified. The output is a GFF3 file that contains the annotation with abundances estimates given for each annotated transcript. `rQuant` computes two abundance estimates. One is based on the estimated average read coverage (ARC) for each transcript and one is the number of reads per thousand bases per million mapped reads (RPKM) [148, 162]. The ARC value is the result of the optimisation problem, i.e., corresponding to variables $w_1, \ldots, w_T$, and the RPKM value is computed based on the ARC value, the transcript length, and the total number of aligned reads.

Read Density Estimation

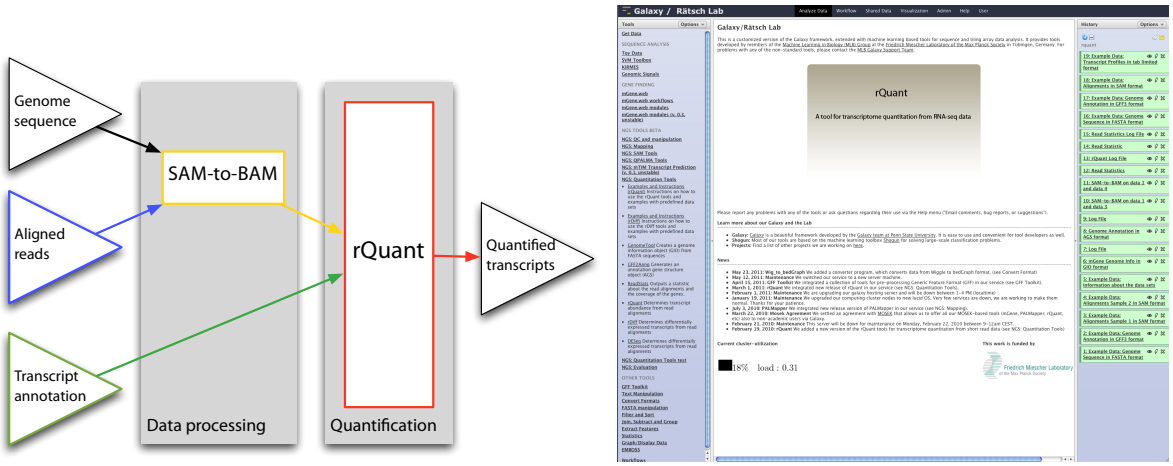
To improve the accuracy of the abundance estimation, `rQuant` can also be used to infer a read density model to predict the read density for considered transcripts. This is done by selecting `Learn Profiles`. Then, `rQuant` iteratively estimates the transcript abundances as well as the read density biases over several transcripts. The outputs are the abundance estimates as before and a file that contains the parameters for the read density model. This parameter file can be used later for quantifications without the need to re-optimise these parameters (select `Load Profiles`).

The quantifications, estimated profiles and other objects can be easily shared with other users via Galaxy's `share history` functionality. Moreover, Galaxy's `Data Libraries` contain items such as genome sequences and parsed annotations for several organisms for convenience.

5.2.2. Statistics

The median running time per gene locus is less than a second for a reasonable number of transcripts. I tested rQuant using reads from the SRX001872 RNA-seq experiment for *C. elegans* with 1,893 annotated genes. The quantification took about 5 minutes without coverage bias density. When enabling the coverage bias estimation, the whole process took roughly 1.5 hours.

Figure 5.2 visualises the number of runs who have used rQuant since its release in the MLB Galaxy instance. More details are given in the caption.



(a) Workflow of rQuant.web. (b) rQuant.web within the MLB Galaxy instance.

Figure 5.1.: The web service rQuant.web. (a) The graph visualises the workflow of rQuant.web. Given read alignments in SAM/BAM format and annotated transcripts in GFF3 format as input, rQuant.web performs quantification and outputs a GFF3 file with abundance estimates for each these transcripts. (b) rQuant.web is embedded in the tool category `NGS: Quantitation Tools` of the MLB Galaxy instance.

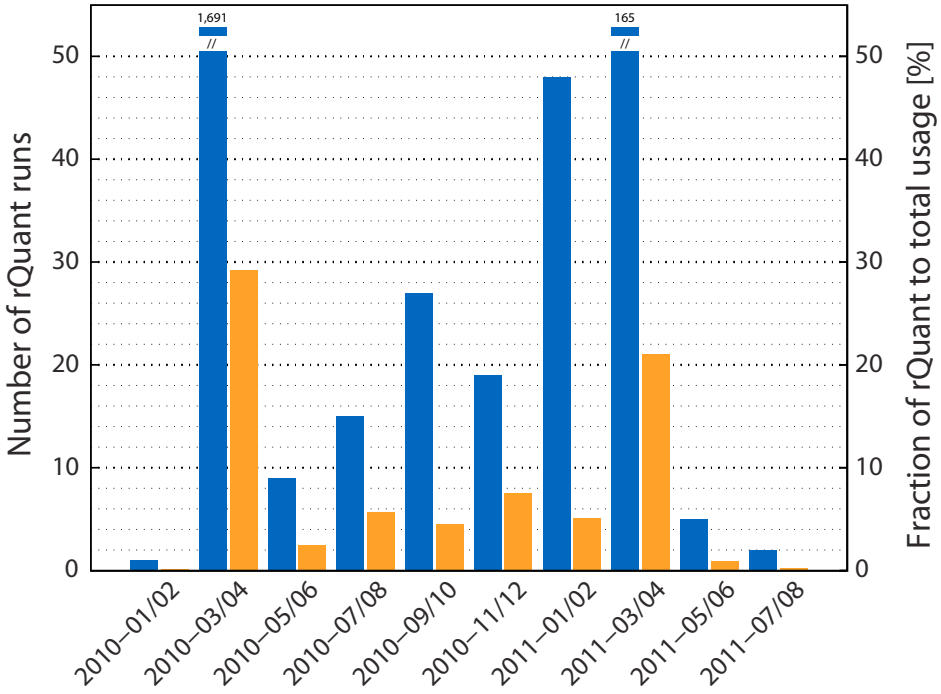


Figure 5.2.: Number of rQuant.web runs in comparison to all runs of tools in the MLB Galaxy instance. The number of rQuant.web runs are illustrated in blue bars, the fraction of rQuant.web to all runs in orange grouped by two months in a row since installation in the beginning of 2010. Only the successful and complete runs of rQuant.web were counted for this analysis. Peaks of usage can be observed during the review time of the rQuant.web paper [28] in March/April 2010 and after publication of the paper in July 2010.

5.2.3. rQuant.web as a Component of Oqtans

Usually, researchers are not only interested in undertaking one of the analysis steps of the RNA-seq analysis pipeline visualised in Figure 5.3, but they would like to analyse their RNA-seq data from the raw reads up to a comparison between different experimental conditions. Such an analysis is made possible in an easy and effective manner by Oqtans, a Galaxy-integrated work flow for **o**nline **q**uantitative **t**ranscriptome **an**alysis from RNA-seq data. Besides rQuant, it contains tools for read alignment, PALMapper [50, 98], for transcript reconstruction, mTiM [69], and for testing of differentially expressed transcripts, rDiff [193]. It can be accessed in the MLB Galaxy instance at `http://galaxy.fml.mpg.de/oqtans` installed locally, or can be used in the cloud. All the components of Oqtans are free, open source and standalone, and can be downloaded from the supplementary web pages `http://www.oqtans.org/`.

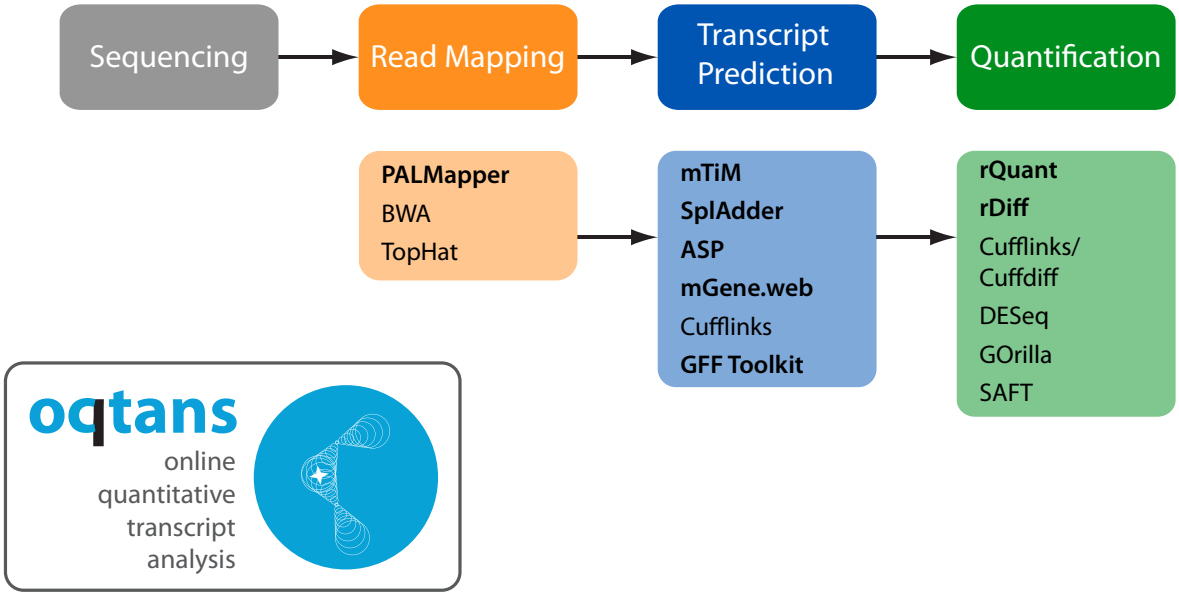


Figure 5.3.: Pipeline for the analysis of RNA-seq data implemented in Oqtans. Raw reads obtained from sequencing are aligned to a reference sequence by a suitable mapping programme (second column). These alignments may be used for transcript prediction (third column), or as an input for quantification programmes to obtain a quantified transcriptome (last column). When having RNA-seq data sets from several experiments, e.g., to study transcriptomes under various conditions or in diverse tissues, sophisticated statistical tests can be applied to derive a set of differentially expressed transcripts. Tools developed by the MLB group are marked in bold. Examples for tools that address the respective problems are PALMapper [50, 98], BWA [120], TopHat [199] (read mapping); mTiM [69], mGene.web [181, 182], Cufflinks [200] (transcript prediction); SplAdder [167] (alternative transcript inference); ASP (splice site prediction); rQuant [28, 30], Cufflinks [200] (transcript quantification); rDiff [193], Cuffdiff [200], DESeq [13] (differentially gene/transcript expression); GOrilla [228] (GO term enrichment); SAFT (alignment filtering). This figure has been adapted from an illustration provided by Géraldine Jean and Sebastian Schultheiß.

5.3. Software Release

5.3.1. Description

rQuant is a programme to determine abundances of multiple transcripts per gene locus from RNA-seq measurements. It can simultaneously estimate the effect of biases introduced by experimental protocols.

5.3.2. Availability

The current release of the rQuant software package can be downloaded from `ftp://ftp.tuebingen.mpg.de/fml/bohnert/rQuant`. rQuant is licenced under the GNU General Public License version 3 or at any later version. Information on rQuant.web and rQuant can be found on the supplementary web pages `http://fml.mpg.de/raetsch/suppl/rquant/web` and `http://fml.mpg.de/raetsch/suppl/rquant`, respectively.

5.3.3. Package Structure

`AUTHORS`	Authors who contributed code to this package.
`./bin`	Contains shell scripts to configure rQuant and to start the interpreter.
`COPYRIGHT`	Copyright of the code.
`./doc/`	Contains the documentation for rQuant.
`./examples`	Contains scripts to download and run examples.
`./galaxy`	Contains configuration XML files for Galaxy integration.
`INSTALL`	Instruction for the installation of the package.
`LICENSE`	GNU General Public License.
`./mex/`	Contains C++ functions with an interface to MATLAB/Octave. They can be compiled to mex functions using the Makefile.
`NEWS`	Describes code changes incorporated in each versions.
`README`	Readme file of the package.
`setup_rquant.sh`	Script to set up rQuant.
`./src`	Contains main code for rQuant. The script `./rquant.sh` starts rQuant. The script `./read_stats.sh` starts the programme to output the read statistics.
`./test_data`	Contains data for running a functional test in Galaxy.
`./tools`	Contains code mainly for file parsing and conversion.
`VERSION`	Version number.

5.3.4. Installation

To setup rQuant, please follow these steps:

1. Download the SAMTools (version $\geq$ 0.1.7) from `http://samtools.sourceforge.net/` and install it. You need to add the flag `-fPIC` in the SAMTools Makefile for compilation.
2. Add the SAMTools directory to ./mex/Makefile, go to ./mex and run make (`make octave` for Octave and `make matlab` for MATLAB).
3. Run `./setup_rquant.sh` and setup paths and configuration options for rQuant.
4. Download the example data with `./get_data.sh` in `./examples`.

5. Run an example by executing `./run_example.sh` with input 'small' or 'big' to work on a small (55 examples) and big (1,865 examples) *C. elegans* data set, respectively in the examples directory.

5.3.5. Interface to Galaxy (rQuant.web)

rQuant can be used as a web service embedded in a Galaxy instance (cf. `http://galaxy.fml.tuebingen.mpg.de/tool_runner?tool_id=rquantweb`). The Galaxy tool configuration file of rQuant is located in the subdirectory `./galaxy` along with an XML file for loading example data and instructions (rquant_web.xml and rquant_web_instructions.xml, respectively). Please adapt the paths to the respective tools in the command section of the XML files as indicated. The subdirectory `./test_data contains` all data for running a functional test in Galaxy (e.g. with `sh run_functional_test.sh -id rquantweb`). You may need to move these test files into the Galaxy test-data directory.

5.3.6. Running rQuant

Requirements

To use rQuant, the following programmes are required:

- Octave [5] or MATLAB [135]
- Python $\geq$ 2.6.5 [6] and Scipy $\geq$ 0.7.1
- SAMTools $\geq$ 0.1.7 [8, 121]

Command

```
./rquant.sh annotation anno_format genes.mat alignments.bam result.gff3 result_dir
load_profiles profiles_in learn_profiles profiles_out
```

Parameters and Options

Inputs

`annotation`: Annotation file either in GFF3 or AGS format, containing the necessary information about the transcripts that are to be quantified.

`alignments.bam`: The BAM alignment file that stores the read alignments in a compressed format.

Options

`anno_format`: Format of the annotation file. 0 denotes GFF3 format, and 1 AGS format.

`load_profiles`: 1 if a pre-learned profile model should be loaded, otherwise 0.

`profiles_in`: Name of text file storing the pre-learned profile model. If the option `load_profiles` is set to 1, then the profile model is loaded from this file.

`learn_profiles`: 1 enables the estimation of the profile model, 0 disables it.

`profiles_out`: Name of text file storing the estimated profile model if the option `learn_profiles` has been enabled.

Output

`genes.mat`: The gene annotation parsed from the GFF3 file and stored in AGS format.

`result.gff3`: A GFF3 file with the attributes ARC and RPKM that describe the abundance of a transcript in ARC (estimated average read coverage) and RPKM (reads per kilobase of exon model per million mapped reads), respectively, estimated by rQuant. The output files are stored in the directory defined by `result_dir`.

Formats

GFF3 Format General Feature Format is a format for describing genes and other features associated with DNA, RNA and protein sequences. GFF3 lines have nine tab-separated fields:

1. seqid - The name of a chromosome or scaffold.
2. source - The program that generated this feature.
3. type - The name of this type of feature. Some examples of standard feature types are 'gene', 'CDS', 'protein', 'mRNA', and 'exon'.
4. start - The starting position of the feature in the sequence (1-based, inclusive).
5. stop - The ending position of the feature (1-based, inclusive).
6. score - A score between 0 and 1000. If there is no score value, then '.'.
7. strand - Valid entries include '+', '-', or '.' (for not available).
8. phase - If the feature is a coding exon, frame should be a number between 0 and 2 that represents the reading frame of the first base. If the feature is not a coding exon, the value should be '.'.
9. attributes - All lines with the same group are linked together into a single item.

For quantification, two additional attributes are provided:

1. ARC: estimated average read coverage (direct output from the optimisation)
2. RPKM: the number of reads per thousand bases per million mapped reads

describing the estimated expression value for each transcript.

For more information about the GFF3 format please visit `http://www.sequenceontology.org/gff3.shtml`.

AGS Format The Annotation Gene Structure (AGS) object is an internal MATLAB/Octave structure that efficiently stores the information parsed from a GFF3 file.

SAM/BAM Format The Sequence Alignment/Map (SAM) format is a tab-delimited text format that stores large nucleotide sequence alignments [121]. BAM is the binary version of a SAM file that allows for fast and intensive data processing. The format specification and the description of SAMtools can be found on `http://samtools.sourceforge.net/`.

Example

This command is an example for running rQuant without bias estimation called from the subdirectory `./src`:

```
./rquant.sh ../test_data/rquant_nGASP-Train-I.gff3 0
../test_data/genes_parsed.mat ../test_data/rquant_nGASP-Train-I.bam
../test_data/rquant_nGASP-Train-I_rquant_case0.gff3 ../test_data
0 dummy_pr_in 0 dummy_pr_out
```

5.4. Conclusion

In this chapter, I described the integration of rQuant, the RNA-seq based quantification method presented in Chapter 3, into the MLB Galaxy instance available at `http://galaxy.fml.tuebingen.mpg.de/`. To the best of my knowledge, rQuant.web has been the first on-line tool to quantitatively analyse RNA-seq data and it has been heavily used since its installation. The web service has been an important contribution to publicly available on-line NGS tools, enabling RNA-seq quantification for any user. It facilitates the analysis of NGS data by being embedded in the MLB Galaxy installation and is very well suited to accurately infer the abundance of alternative transcripts along with a simultaneously learned bias model.

Furthermore, rQuant has been released as a standalone, free, and open-source package, which can be downloaded from an FTP site and can be installed in the user's own Galaxy instance via Galaxy-specific installation fabric scripts. It is also integrated in the RNA-seq analysis pipeline Oqtans, allowing an easy and effective analysis of RNA-seq data in a web interface for users not familiar with running programmes on a command line.

6. Conclusion

In this dissertation, I presented computational approaches based on ideas taken from machine learning and optimisation to efficiently and accurately analyse huge amounts of data from genomics and transcriptomics high-throughput experiments. Investigating a rich set of array resequencing data, I identified genome sequence variation across diverse varieties of domesticated rice (Chapter 2). The second part of my thesis addressed the question how to estimate abundances of alternative transcripts and secondary structures from next-generation sequencing data (Chapters 3, 4 and 5).

To study sequence variation in domesticated rice, I applied state-of-the-art machine learning methods based on Support Vector Machines and Hidden Markov Support Vector Machines for SNP identification and segmentation of genomes into polymorphic and non-polymorphic regions. Across all varieties, 1,343,270 SNPs at 316,373 non-redundant sites were identified with the SVM-based classification. I assessed the quality of the predictions on a gold standard set of polymorphisms, which was assembled based on dideoxy sequencing of selected genomic regions. Evaluated on a this set, 20.9 % of all SNPs at an FDR of 8.3 % could be recovered. A high-quality set of 159,879 SNPs (2.9 % FDR) was assembled from the intersection of SNPs predicted by the machine learning method and a model based approach, which was used for subsequent biological analysis.

The trained HM-SVM model detected between 65,000 and 203,000 polymorphic regions across the varieties, which covered between 1.7 % and 5.1 % of the queried reference genome sequence, respectively. While recovering 26 % of the polymorphisms in the gold standard set of polymorphisms, a precision of 80 % could be achieved. Polymorphic region predictions were also very valuable for the identification of SNPs, exclusively revealing 21 % of the annotated SNPs and thus complementing SNP predictions. Besides the assembly of the set for precise polymorphism detection, the algorithm was used to predict polymorphic regions at varying recall cut-offs. These sensitive predictions have their benefit when defining conserved regions in the genome to provide a scaffold for successful primer design for sequencing experiments.

In both analyses, the use and power of machine learning approaches could be shown. Dealing with uncertainties that were difficult to resolve by heuristics and non-adaptive algorithms, as seen for the model-based approach in the SNP analysis, the rather noisy and challenging array data could be successfully analysed to uncover SNPs and PRs at a remarkable accuracy.

At the time when the rice resequencing project was initiated, array-based resequencing was the best available method to study sequence variation on a genome-wide scale and for many individuals in parallel. Next-generation sequencing have now replaced tiling arrays in this respect. Nevertheless, methodology developed for array analysis has not been limited to this kind of data and have been applied to similar problems arising from NGS data. For example, the algorithm proposed for the inference of non-redundant polymorphic regions in the rice project has been adopted to a similar problem for regions with low and no read coverage measured in a next-generation sequencing effort for 18 *Arabidopsis thaliana* strains. Although many digitally sequenced rice genomes have been recently published, the set of polymorphisms assembled here was the first set for a wide range of diverse varieties for the world's most important crop plant on a genome-wide scale.

The second part of my thesis focused on the analysis of transcriptome data generated with RNA sequencing and presented novel computational methods developed for this purpose. To quantitatively deconvolve transcripts, I developed and implemented one of the first tools for quantification of alternative transcripts based on RNA-seq measurements (Chapter 3). This approach, called rQuant, used methodology from machine learning and mathematical programming. Adapting ideas from the lasso approach, rQuant estimated transcript abundances by minimising the deviation of observed to expected read coverage. One key feature of rQuant was that the expected read coverage was not only parametrised by the transcript abundance, but also by the expected read density. Several studies have shown that read density might be distorted due to diverse experimental steps, for example RNA-seq library preparation and sequencing, and that it was dependent on transcript properties such as its length and the sequence content. In rQuant, read density was modelled by profile functions that described the read coverage in dependency of distance to the transcript boundaries and the length. Furthermore, I incorporated a second model relating read density to occurrences of sequence oligo-mers.

On artificial read data sets simulated for transcripts from *Caenorhabditis elegans*, I showed that abundance estimates by rQuant very well correlate with the true number of molecules. Modelling read density improves quantification accuracy by $\approx 4\,\%$. Moreover, I demonstrated the superiority of rQuant in comparison to two popular quantification tools, Cufflinks and MISO. The usage of paired-end reads in the rQuant algorithm improved quantification results to a small extent. In another extension of rQuant, I adapted the method to consider RNA-seq measurements across multiple conditions. An evaluation on artificial read data for *Arabidopsis thaliana* indicated that quantifying transcripts based on read data from several conditions simultaneously made abundance estimates more stable and considerably improved results for testing of differentially expressed transcripts. Besides the remarkable performance of rQuant assessed on artificial data sets, I showed on an experimental RNA-seq data set for two replicates from *Arabidopsis lyrata* that correcting for biases was an important component of quantification, improving the replicability of abundance estimates by $\approx 20\,\%$.

A limitation of rQuant is that it is dependent on an annotation, which are often incomplete, and thus novel transcripts cannot be quantified. Computational methods such as mGene,ngs [17, 64] and mTiM [69] can be used to complement annotations. Here, rQuant allows to prune not expressed transcripts, as it provides a sparse quantification solution. Approaches that simultaneously identify and quantify transcripts are a straightforward alternative to address this problem. A method that has currently been developed, called SplAdder, extends ideas from rQuant and identifies and quantifies transcripts from a splicegraph based on mixed integer programming [167].

The approach of rQuant is a general framework to quantitatively analyse read data from any organism, even though evaluation results presented in this dissertation were obtained for a small set of (model) organisms. Also, the method of rQuant is not restricted to the analysis of transcript abundances measured with RNA-seq, but is applicable to other quantitative data in general. In Chapter 4, I exemplified this by adapting the rQuant framework to address quantification of RNA secondary structure based read count data, implemented in the tool sQuant. On key result from the analysis of sQuant was that computational analysis of structural read data does not only help in quantification of alternative RNA secondary structures, but also improves transcript quantification accuracy. Thus, transcript quantification based on PARS-seq may have an advantage over RNA-seq-based quantification in the future. Moreover, I showed that biases were observable also in this kind of data and need to be addressed in quantification.

rQuant is available to the community as open-source software and as a web service integrated in the Galaxy instance of our research group (Chapter 5). Moreover, it is an important component of the Oqtans software package, which provides access to tools for RNA-seq analysis within a local Galaxy instance or in the cloud.

Future developments and improvements in high-throughput sequencing will potentially also facilitate direct quantification. Longer sequence reads will better resolve alternative transcripts and help in unambiguously assigning them to isoforms. However, computational quantification tools such as rQuant will be needed to infer abundances of alternative transcripts. Although tuning of RNA-seq library protocols has reduced some of biases in read data, bias correction will still be one crucial step in quantitative RNA-seq analysis in the near future. In the long term, with the development of 'third-generation' sequencing machines, conversion of RNA to cDNA might be no longer necessary and library-dependent biases might thus vanish. However, these sequencing strategies will probably also harbour biases, which will affect quantification.

The techniques developed within the scope of this thesis use methodology from machine learning and optimisation. Their application to high-throughput data from molecular biology showed their power in comparison to heuristics that are often used in this domain. Both genomic and transcriptomic data analysed in my dissertation showed problems that often occur in biological data: They were noisy, of high dimensionality, large in number and harboured uncertainties. All these challenges could be successfully addressed and solved by specifically designed machine learning and optimisation algorithms.

In summary, my work presented in this thesis contributed to key parts of high-throughput genomic and transcriptomic research. Inferring genotypes and expression phenotypes in an accurate and efficient manner for many individuals is crucial for genome-wide association studies. These studies will help to identify causal gene variants for complex phenotypes such as diseases and will extend our knowledge of underlying biological processes such as regulation of gene expression and RNA processing, a crucial step towards a comprehensive understanding of the central dogma of molecular biology.

A. Supplementary Figures

A.1. Detecting Sequence Variation from Resequencing Arrays

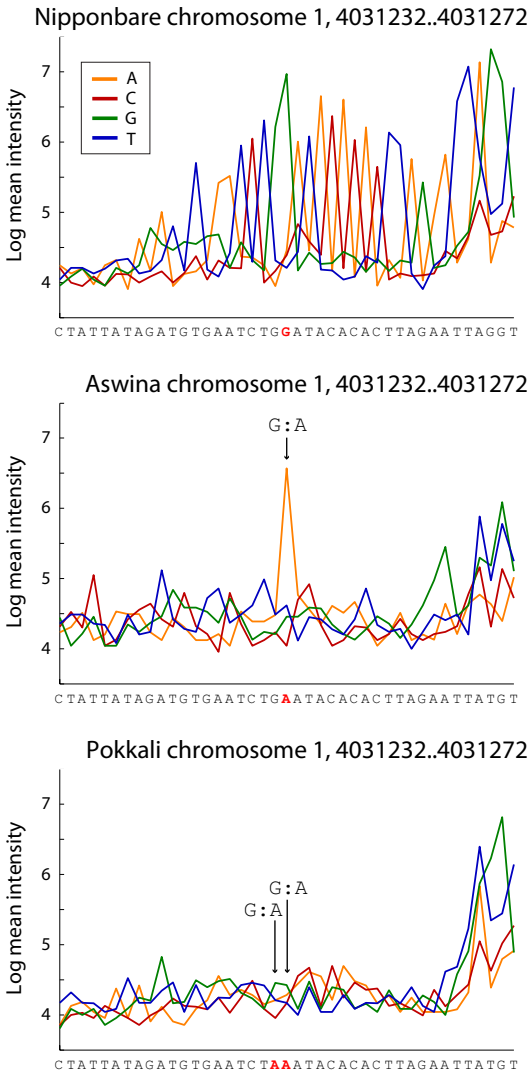
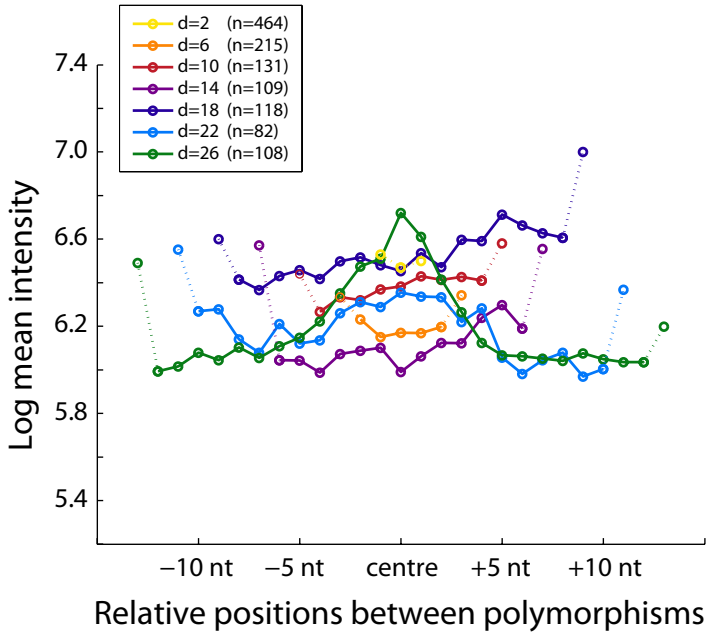
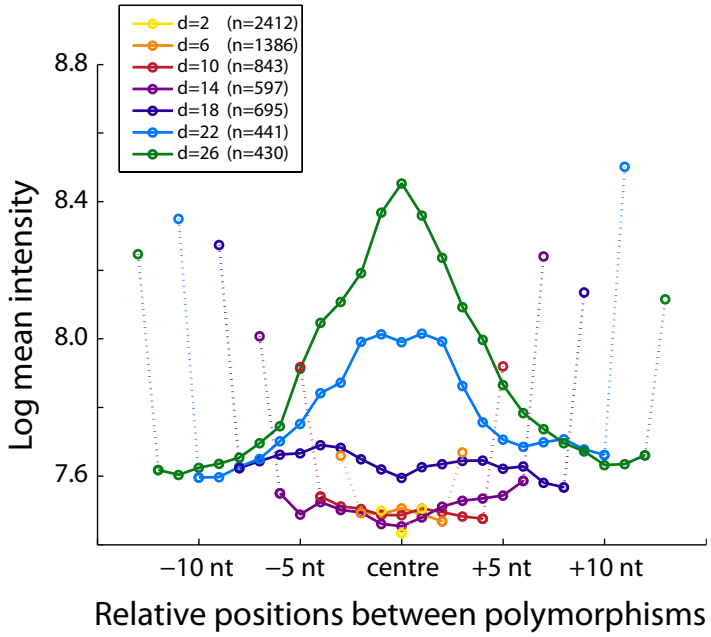


Figure A.1.: Hybridisation trace file example on chromosome 1 at position 4,031,232 to 4,031,272. Log_2 intensities for all four bases averaged over both strands are shown with the identified sequence. In conserved regions (Nipponbare trace at the top), hybridisation intensity is strongest for reference probes. At a SNP position (Aswina trace in the middle), intensity is typically stronger for the oligonucleotide representing the SNP allele. Intensity is reduced at positions next to a SNP, as these oligonucleotides all have off-centre mismatches. Therefore, nearby SNPs (Pokkali trace at the bottom) are more difficult to call.

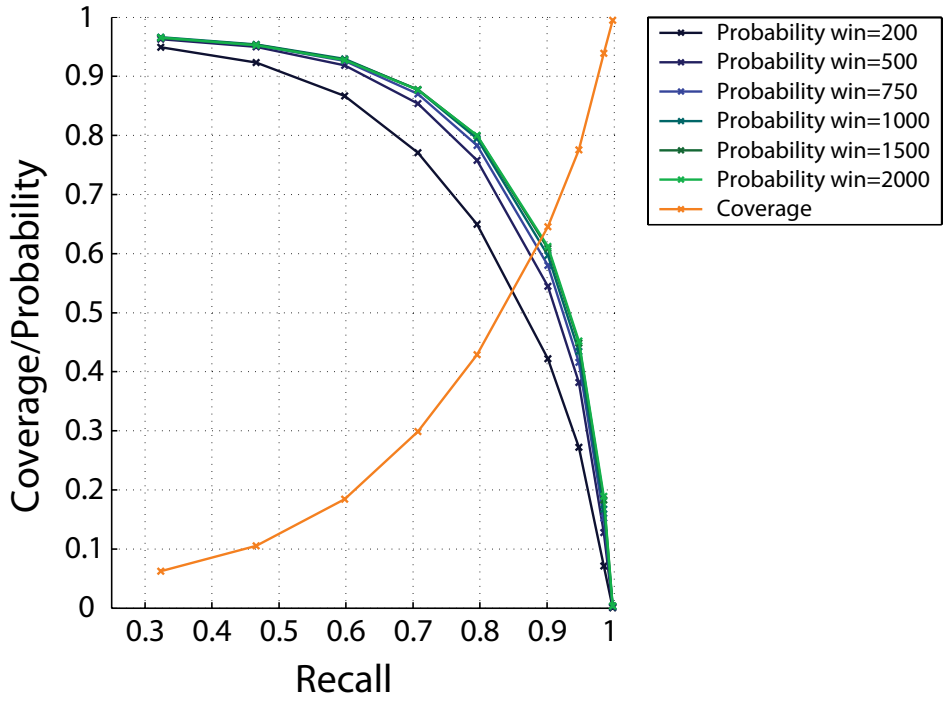


(a) Polymorphism signature for *O. sativa*.

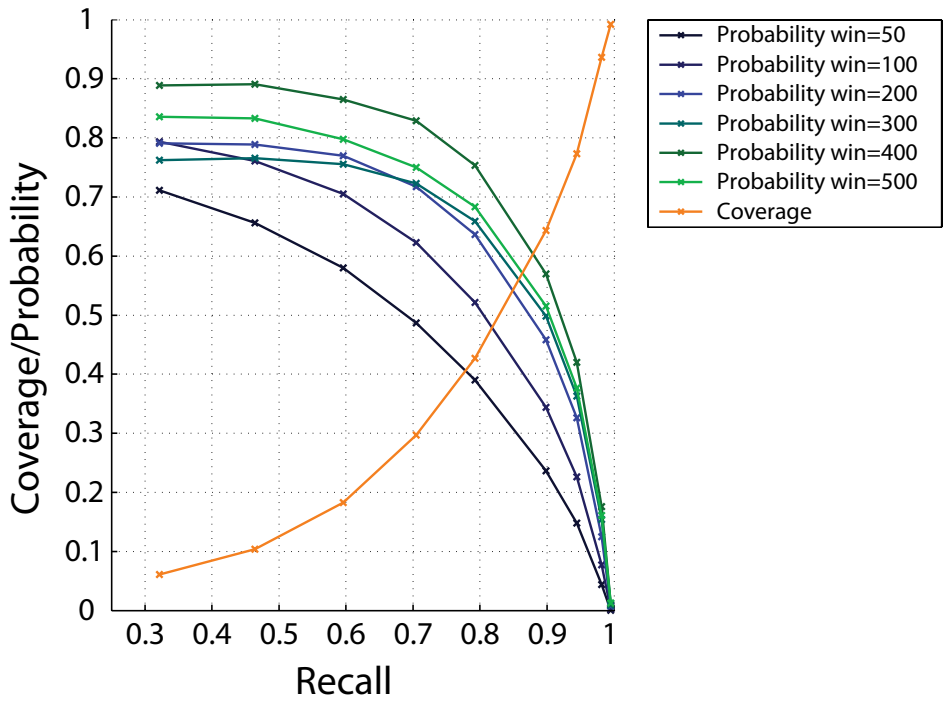


(b) Polymorphism signature for *A. thaliana*.

Figure A.2.: Polymorphism signature. Regions between two polymorphisms in the GSP data were grouped into the same category according to their distance (≤ 26 bp). For each distance category, the sample size n is given (see inlet). The maximal $\log_2$ intensities per probe quartet between polymorphisms were averaged (indicated as circles and solid lines). The intensities at the polymorphic positions are shown as the outermost circles and dotted lines. (a) For *O. sativa*, the curves are characterised by a flatter shape. For shorter distances, intensities between polymorphisms were generally suppressed, but still remained on a higher level. Nevertheless, the characteristic polymorphism signature was observable at larger distances (light blue and green curve). (b) For *A. thaliana* (cf. [41]), the pattern was more obvious, and short distance categories with suppressed intensities and long distance categories with non-suppressed intensities could be clearly distinguished. Nevertheless, the same distance (18 bp) for the definition of contiguous polymorphic regions was used for rice.



(a) Probability of successful primer design assessed on genome-wide set.



(b) Probability of successful primer design assessed on GSP set.

Figure A.3.: Probability of successful primer design. For each set of primer PRs, the recall rate was recomputed to account for differences in the composition of the GSP and the genome-wide set. The coverage curve (orange) shows the fraction of sites predicted to be polymorphic for all used sets. For different window sizes (see inlet), the success rate was estimated for each of the primer PR sets.

A.2. rQuant: Modelling Biases for Accurate RNA-seq-based Transcript Quantification

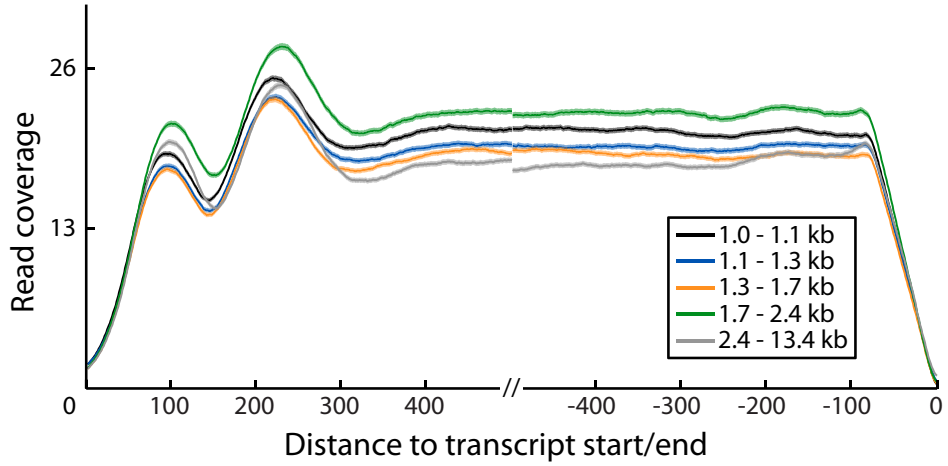


Figure A.4.: Transcript profiles for artificial data set with library biases. Normalised read coverage is shown in dependency of the absolute distance to the transcript boundaries for bins of different transcript length (see inlet). The read coverage was determined from simulated reads for transcripts longer than 1,000 nt in the artificial data set with weak library biases described in Section 3.2.4.

A.3. sQuant: Generalisation of rQuant to RNA Structure Quantification

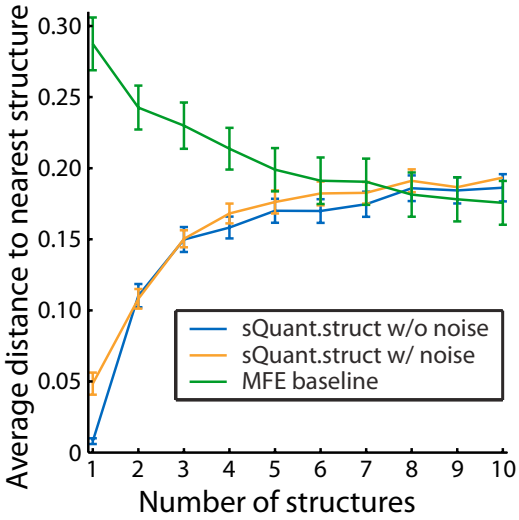


Figure A.5.: Accuracy of sQuant.denovo. The average distance to the nearest structure is shown in dependency of the number of structures in the mixture. sQuant.denovo was applied on a set of 50 genes with one to ten structures for which reads were simulated without noise (blue) and with noise (orange). MFE denotes the minimum free energy structure predicted by ViennaRNA RNAsubopt [83].

B. Supplementary Tables

B.1. Detecting Sequence Variation from Resequencing Arrays

Table B.1.: The 20 rice varieties selected for resequencing by subgroup with their agronomic attributes.

Subgroup	Variety	Origin	Agronomic Attributes
Temperate japonica			
	Nipponbare	Japan	Served as reference strain.
	LTH	China	High disease susceptibility. Cold tolerant.
	M 202	U.S.A.	Popular variety.
	Tainung 67	Taiwan	
Tropical japonica			
	Azucena	The Philippines	Grain-iron quantitative trait loci. Deep root distribution.
	Cypress	U.S.A.	Good grain quality. Cold tolerant.
	Moroberekan	Guinea	Multiple advanced breeding populations. Drought tolerant.
Aromatic			
	Dom-sufid	Iran	Basmati plant type. Aromatic rice.
Aus			
	Dular	India	Possible β-carotene donor. Red pericarp.
	FR13 A	India	Submergence tolerance. Red pericarp.
	N 22	India	Iron, red pericarp. Heat tolerant and considered drought tolerant.
	Rayada	Bangladesh	Deep-water rice.
Indica			
	Aswina	Bangladesh	Deep-water rice.
	IR64-21	The Philippines	Multiple stress tolerance (diseases and insects). Progenitor of many breeding and mapping populations.
	Minghui 63	China	
	Pokkali	India	Salt tolerant.
	SHZ2	China	Disease-resistant, high-yielding variety. In the pedigrees of many varieties in south China.
	Sadu-Cho	Korea	Long grain and indica-type endosperm.
	Swarna	India	High yield potential, wide adaptability. Widely planted variety in India.
	Zhenshan 97B	China	

Table B.2.: Repetitive 25-mer matches for *O. sativa japonica* variety Nipponbare. The percentage of repetitive positions is given in squared brackets.

25-mer Match Type	Match Pairs	Repetitive Positions
Exact	3,002,679	2,528,180 [2.53]
Inexact	6,027,990	2,530,991 [2.53]
Short	2,478,436	940,055 [0.94]
Bulged	5,983,927	1,569,404 [1.57]
Union	17,493,032	5,160,864 [5.16]

Table B.3.: Dominating repetitive 25-mer matches for *O. sativa japonica* variety Nipponbare. The percentage of repetitive positions is given in squared brackets.

25-mer Match Type	Match Pairs	Repetitive Positions
Exact	1,664,629	1,537,687 [1.54]
Inexact	212,514	208,776 [0.21]
Short	663,823	610,036 [0.61]
Bulged	1,406,553	1,048,578 [1.05]
Union	3,947,519	2,693,837 [2.69]

Table B.4.: Number of SNPs for the different sequence types in the dideoxy sequencing set across all varieties grouped by the five subgroups. The fraction to all SNPs per variety are given in squared brackets.

Subgroup/Variety	All SNPs	Coding	UTR	Intron	Intergenic
Temperate japonica					
LTH	42	3 [07.1]	7 [16.7]	5 [11.9]	27 [64.3]
M 202	55	12 [21.8]	1 [01.8]	1 [01.8]	41 [74.6]
Tainung 67	9	2 [22.2]	4 [44.4]	2 [22.2]	1 [11.1]
Tropical japonica					
Azucena	222	28 [12.6]	20 [09.0]	20 [09.0]	154 [69.4]
Cypress	412	24 [05.8]	40 [09.7]	46 [11.2]	302 [73.3]
Moroberekan	253	6 [02.4]	12 [04.7]	40 [15.8]	195 [77.1]
Aromatic					
Dom-sufid	82	12 [14.6]	10 [12.2]	20 [24.4]	40 [48.8]
Aus					
Dular	654	100 [15.3]	74 [11.3]	73 [11.2]	407 [62.2]
FR13 A	688	97 [14.1]	122 [17.7]	84 [12.2]	385 [56.0]
N 22	854	127 [14.9]	93 [10.9]	79 [09.3]	555 [65.0]
Rayada	903	94 [10.4]	123 [13.6]	90 [10.0]	596 [66.0]
Indica					
Aswina	414	54 [13.0]	43 [10.4]	67 [16.1]	250 [60.4]
IR64-21	599	51 [08.5]	116 [19.4]	126 [21.0]	306 [51.1]
Minghui 63	797	96 [12.1]	99 [12.4]	75 [09.4]	527 [66.1]
Pokkali	908	139 [15.3]	131 [14.4]	104 [11.5]	534 [58.8]
Sadu-Cho	414	113 [27.3]	38 [09.2]	46 [11.1]	217 [52.4]
SHZ2	685	125 [18.3]	83 [12.1]	80 [11.9]	397 [58.0]
Swarna	924	128 [13.9]	106 [11.5]	91 [09.9]	599 [64.8]
Zhenshan 97B	584	67 [11.5]	106 [18.2]	75 [12.8]	336 [57.5]
Total	9,499	1,278 [13.5]	1,228 [12.9]	1,124 [11.8]	5,869 [61.8]

Table B.5.: Properties of the input vectors $\boldsymbol{x}_p^1$ for layer 1 and $\boldsymbol{x}_p^2$ for layer 2 at a given position p are shown. $\Sigma = \{\text{A, C, G, T}\}$ denotes the DNA alphabet, the strands are represented by $s \in \{+,-\}$, τ is either target t or reference variety ref. Unless defined otherwise, $\Delta p \in \{-4, \dots 4\}$.

	Formula	Description	Size
I_{max}	$I^p_{max}(\Delta p, \tau, s) = \max_{\sigma\in\Sigma} \log\left(I_{\tau,s}(p+\Delta p, \sigma)\right)$	Maximal intensities for target and reference variety, strand-wise in a 9 bp window	36
I_{sec}	$I^p_{sec}(\Delta p, \tau, s) = \text{mean}_{\sigma\neq\sigma_{max}} \log\left(I_{\tau,s}(p+\Delta p, \sigma)\right)$ where $\sigma_{max} = \arg\max_{\sigma\in\Sigma} \log\left(I_{\tau,s}(p+\Delta p, \sigma)\right)$	Average of the non-maximal intensities for target and reference variety, strand-wise in a 9 bp window	36
Q_1	$Q^p_1(\Delta p, \tau, s) = \log\left(\frac{I^p_{max}(\Delta p,\tau,s)}{I^p_{max}(0,\tau,s)}\right)$ where $\Delta p \in \{-4, \dots, -1, 1, \dots 4\}$	Quotients of the maximal intensities at the neighbouring positions and p for target and reference variety, strand-wise in a 9 bp window	32
Q_2	$Q^p_2(\Delta p, s) = \log\left(\frac{I^p_{max}(\Delta p,t,s)}{I^p_{max}(\Delta p,ref,s)}\right)$	Quotients of the maximal intensities at p for target and reference variety, strand-wise in a 9 bp window	18
M	$M^p(\Delta p, \tau, s) = \delta(B_{\tau,s}(p+\Delta p), RS(p))$ where $\delta(i,j) = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$	Mismatches between the raw base calls of the target/reference variety and the reference sequence, strand-wise in a 9 bp window	36
RS	$RS^p(\sigma) = \delta(RS(p), \sigma)$	Binary vector indicating the presence of the base in the reference at p	4
f	$f^p(\sigma) = \sum_{\Delta=-13}^{\Delta=13} \delta(RS(p+\Delta), \sigma)$	Frequency of each base within the 25-mer	4
H	$H^p = -\sum_{\sigma\in\Sigma} f^p(\sigma) \log(f^p(\sigma))$	Sequence entropy of the 25-mer	1
k	$k^p(\Delta p, \sigma) = [k^p_{type}(\Delta p, \sigma), k^p_{dom\ type}(\Delta p)]$ where $type \in \{exact, inexact, short, bulged\}$	Occurrence counts of repetitive 25-mers in a 9 bp window	558
v	$v^p(\gamma) = \begin{cases} 10 & \gamma = v(p) \\ 0 & \gamma \neq v(p) \end{cases}$	Vector indicating the variety origin of p	19
	$\boldsymbol{x}^1 = [I_{max}, I_{sec}, Q_1, Q_2, M, RS, f, H, k, v]^T$		744
b	$b^p(t) = [[c^p(t) \geq th_t]]$	Binary vector indicating whether p passed th_t for variety t	361
c	$c^p(t)$	Confidences at p for all varieties t	361
ind	$ind^p = \begin{cases} +1 & RS_{jap} \neq RS_{ind} \\ -1 & RS_{jap} = RS_{ind} \end{cases}$	Variation to the *ssp. indica* reference	2
	$\boldsymbol{x}^2 = [I_{max}, I_{sec}, Q_1, Q_2, M, RS, f, H, k, b, c, ind]^T$		1,449

Table B.6.: Input features used for polymorphic region prediction. In general, p denotes the position, superscripts + and − the strand, t the target variety and ref the reference (i.e., Nipponbare), $\tau \in \{t, ref\}$, seq the reference sequence, B the raw base call and [[]] the indicator function. Other definitions are given in the descriptive text (see also [140]). Features denoted by * were used in the *A. thaliana* study [41].

	Formula	Description
*1	$IM(p) = \frac{1}{2}\,[\,\log(I^+_{max}(p)) + \log(I^-_{max}(p))\,]$	Log maximal intensity at p (averaged over both strands)
*2	$IR(p) = IM_t(p) - IM_{ref}(p)$	Ratio of target and reference maximal averaged intensities at p
3	$IR_{up}(p) = IM_t(p) - IM_t(p-1)$	Ratio of target maximal averaged intensities at p and the upstream position $p-1$
4	$IR_{down}(p) = IM_t(p) - IM_t(p+1)$	Ratio of target maximal averaged intensities at p and the downstream position $p+1$
*5	$IN(p) = \frac{1}{2} \sum_{\delta\in\{-1,+1\}} (IM_t(p) - IM_t(p+\delta))$	Average of up- and downstream intensity ratio
*6	$IW_9(p) = \frac{1}{9} \sum_{\delta=-4}^{4} IR(p+\delta)$	Target/reference intensity ratio in a 9 bp window
7	$IW_{25}(p) = \frac{1}{25} \sum_{\delta=-12}^{12} IR(p+\delta)$	Target/reference intensity ratio in a 25 bp window
8	$Q(p) = \frac{1}{2}\,[\,Q^+(p) + Q^-(p)\,]$	Quality score at p (averaged over both strands)
9	$Q_9(p) = \frac{1}{9} \sum_{\delta=-4}^{4} Q(p+\delta)$	Quality score at p (averaged over both strands) in a 9 bp window
10	$Q_{25}(p) = \frac{1}{25} \sum_{\delta=-12}^{12} Q(p+\delta)$	Quality score at p (averaged over both strands) in a 25 bp window
11	$QR(p) = \frac{1}{2} \sum_{s\in\{+,-\}} \frac{Q^s_t(p)}{1+Q^s_{ref}(p)}$	Ratio of target and reference averaged quality scores at p
12	$QR_{up}(p) = \frac{1}{2} \sum_{s\in\{+,-\}} \frac{Q^s_t(p)}{1+Q^s_t(p-1)}$	Ratio of target quality scores at p and the upstream position $p-1$
13	$QR_{down}(p) = \frac{1}{2} \sum_{s\in\{+,-\}} \frac{Q^s_t(p)}{1+Q^s_t(p+1)}$	Ratio of target quality scores at p and the downstream position $p+1$
14	$QW_9(p) = \frac{1}{9} \sum_{\delta=-4}^{4} QR(p+\delta)$	Target/reference quality score ratio in a 9 bp window
15	$QW_{25}(p) = \frac{1}{25} \sum_{\delta=-12}^{12} QR(p+\delta)$	Target/reference quality ratio in a 25 bp window
*16	$QN(p) = \frac{1}{4} \sum_{\delta\in\{-1,+1\}} \sum_{s\in\{+,-\}} \frac{Q^s_t(p)}{1+Q^s_t(p+\delta)}$	Average of up- and downstream quality score ratio

	Formula	Description
*17	$M_9(p) = \sum_{\delta=-4}^{4} (mm_t(p+\delta) - mm_{ref}(p+\delta))$	Difference of mismatch counts between raw base calls B and reference sequence seq in a 9 bp window at p, $mm_\tau(p) = [[B_\tau^+ \neq seq(p)]] + [[B_\tau^- \neq seq(p)]]$
18	$M_{25}(p) = \sum_{\delta=-12}^{12} (mm_t(p+\delta) - mm_{ref}(p+\delta))$	Difference of mismatch counts between B and seq in a 25 bp window at p
*19	$WL(p) = 1 + \log_2(wl_t(p))$	Perfect word counts (log transformed), $wl_\tau(p)$ equals the number of consecutive sites around p where $B_\tau^s(p') = seq(p')\ \forall s \in \{+,-\}$
20	$WD(p) = \frac{wl_t(p)}{wl_{ref}(p)}$	Ratio of target and reference perfect word counts
21	$WD_{log}(p) = \log_2\left(\frac{wl_t(p)}{wl_{ref}(p)}\right)$	Ratio of target and reference perfect word counts (log transformed)
22	$WQ(p) = 1 + \log_2(qs \cdot wl_t(p))$	Perfect word counts weighted with normalised quality scores qs (log transformed)
23	$CC_+(p) = 4 \cdot idx(seq(p)) + idx(B_\tau^+)$	Call concordance forward strand, $idx('A') = 0$, $idx('C') = 1$, $idx('G') = 2$, $idx('T') = 4$
24	$CC_-(p) = 4 \cdot idx(seq(p)) + idx(B_\tau^-)$	Call concordance reverse strand
*25	$R(p) = [[p \in \mathcal{R}]]$	Binary feature indicating whether p is annotated as repetitive, $\mathcal{R}$ is the set of repetitive sites
26	$IND(p) = \begin{cases} +1 & p \in \text{93-11P \& polymorphic} \\ -1 & p \in \text{93-11P \& conserved} \\ 0 & p \notin \text{93-11P} \end{cases}$	Known polymorphisms in the 93-11 genome

Table B.7.: Number of MB and ML SNP predictions by subgroup. The two performance measures [recall, FDR] are given in % in squared brackets. MBML-union denotes the union of the ML and MB set, MBML-intersect are all SNPs detected by both methods, MB only and ML only are predictions made by one of the methods only. NR denotes FDRs which could not be assessed, as no predictions were made. Note that Nipponbare was excluded both in the dideoxy SNP set and in the ML predictions.

Subgroup/Variety	MBML-union	MBML-intersect	MB only	ML only
Temperate japonica				
Nipponbare	5,425 [NR, NR]	0 [NR, NR]	5,425 [NR, NR]	0 [NR, NR]
LTH	22,233 [66.7, 00.0]	3,021 [16.7, 00.0]	17,240 [50.0, 00.0]	1,972 [00.0, NR]
M 202	27,967 [33.3, 00.0]	5,295 [08.3, 00.0]	16,972 [25.0, 00.0]	5,700 [00.0, NR]
Tainung 67	12,650 [66.7, 00.0]	0 [00.0, NR]	12,648 [66.7, 00.0]	2 [00.0, NR]
Tropical japonica				
Azucena	47,343 [23.2, 16.1]	15,269 [07.1, 20.0]	16,735 [02.7, 00.0]	15,339 [13.4, 16.7]
Cypress	60,362 [18.5, 16.7]	23,984 [09.0, 05.6]	15,044 [02.1, 33.3]	21,334 [07.4, 22.2]
Moroberekan	55,202 [12.6, 09.1]	22,044 [05.7, 00.0]	13,668 [00.6, 50.0]	19,490 [06.3, 09.1]
Aromatic				
Dom-sufid	56,090 [04.7, 33.3]	2,029 [00.0, NR]	53,013 [04.7, 33.3]	1,048 [00.0, NR]
Aus				
Dular	148,428 [22.0, 12.0]	64,039 [10.8, 00.0]	33,232 [02.0, 33.3]	51,157 [09.2, 17.8]
FR13 A	150,221 [26.9, 14.5]	66,689 [12.2, 02.0]	34,276 [02.8, 26.7]	49,256 [11.9, 21.7]
N 22	126,818 [20.8, 10.9]	58,575 [11.6, 06.3]	29,584 [02.8, 06.7]	38,659 [06.5, 19.5]
Rayada	152,094 [29.0, 05.6]	66,476 [14.8, 00.0]	34,154 [02.7, 12.5]	51,464 [11.6, 10.3]
Indica				
Aswina	160,002 [30.4, 08.8]	67,272 [12.9, 00.0]	33,926 [05.0, 20.0]	58,804 [12.5, 11.8]
IR64-21	137,327 [26.5, 07.0]	58,554 [11.9, 00.0]	32,989 [02.6, 33.3]	45,784 [11.9, 05.3]
Minghui 63	139,979 [29.8, 07.0]	60,967 [14.0, 04.2]	30,414 [03.4, 10.5]	48,598 [12.3, 09.0]
Pokkali	132,086 [18.9, 10.9]	56,099 [08.2, 06.1]	34,382 [01.8, 23.1]	41,605 [08.9, 12.3]
SHZ2	114,132 [21.9, 13.6]	48,268 [11.8, 00.0]	33,708 [03.5, 37.5]	32,156 [06.7, 17.1]
Sadu-Cho	144,528 [21.2, 08.8]	60,971 [08.9, 00.0]	36,404 [03.1, 25.0]	47,153 [09.2, 10.0]
Swarna	135,270 [24.3, 07.6]	54,369 [08.4, 00.0]	37,510 [04.4, 17.2]	43,391 [11.5, 08.7]
Zhenshan 97B	103,556 [24.0, 16.1]	38,768 [08.2, 11.1]	33,355 [05.1, 16.7]	31,433 [10.7, 19.2]
Total	1,931,713 [23.8, 10.3]	772,689 [10.7, 02.9]	554,679 [03.2, 21.1]	604,345 [09.9, 13.7]

Table B.8.: Number of MB and ML SNP predictions at non-repetitive sites by subgroup. The two performance measures [recall, FDR] are given in % in squared brackets. MBML-union denotes the union of the ML and MB set, MBML-intersect are all SNPs detected by both methods, MB only and ML only are predictions made by one of the methods only. NR denotes FDRs which could not be assessed, as no predictions were made. Note that Nipponbare was excluded both in the dideoxy SNP set and in the ML predictions.

Subgroup/Variety	MBML-union	MBML-intersect	MB only	ML only
Temperate japonica				
Nipponbare	3,968 [NR, NR]	0 [NR, NR]	3,968 [NR, NR]	0 [NR, NR]
LTH	19,547 [66.7, 00.0]	2,912 [16.7, 00.0]	14,908 [50.0, 00.0]	1,727 [00.0, NR]
M 202	25,258 [33.3, 00.0]	5,198 [08.3, 00.0]	14,549 [25.0, 00.0]	5,511 [00.0, NR]
Tainung 67	10,755 [66.7, 00.0]	0 [00.0, NR]	10,753 [66.7, 00.0]	2 [00.0, NR]
Tropical japonica				
Azucena	43,816 [23.2, 16.1]	14,981 [07.1, 20.0]	14,203 [02.7, 00.0]	14,632 [13.4, 16.7]
Cypress	55,957 [18.5, 16.7]	23,489 [09.0, 05.6]	12,348 [02.1, 33.3]	20,120 [07.4, 22.2]
Moroberekan	50,889 [12.6, 09.1]	21,565 [05.7, 00.0]	11,079 [00.6, 50.0]	18,245 [06.3, 09.1]
Aromatic				
Dom-sufid	51,817 [04.7, 33.3]	2,022 [00.0, NR]	48,747 [04.7, 33.3]	1,048 [00.0, NR]
Aus				
Dular	140,576 [22.0, 12.0]	62,977 [10.8, 00.0]	28,742 [02.0, 33.3]	48,857 [09.2, 17.8]
FR13 A	142,866 [26.9, 14.5]	65,863 [12.2, 02.0]	29,163 [02.8, 26.7]	47,840 [11.9, 21.7]
N 22	120,451 [20.8, 10.9]	57,805 [11.6, 06.3]	25,469 [02.8, 06.7]	37,177 [06.5, 19.5]
Rayada	144,562 [29.0, 05.6]	65,570 [14.8, 00.0]	29,404 [02.7, 12.5]	49,588 [11.6, 10.3]
Indica				
Aswina	151,662 [30.4, 08.8]	66,152 [12.9, 00.0]	29,261 [05.0, 20.0]	56,249 [12.5, 11.8]
IR64-21	130,244 [26.5, 07.0]	57,686 [11.9, 00.0]	28,515 [02.6, 33.3]	44,043 [11.9, 05.3]
Minghui 63	133,496 [29.8, 07.0]	60,237 [14.0, 04.2]	25,875 [03.4, 10.5]	47,384 [12.3, 09.0]
Pokkali	125,448 [18.9, 10.9]	55,402 [08.2, 06.1]	29,741 [01.8, 23.1]	40,305 [08.9, 12.3]
SHZ2	108,791 [21.9, 13.6]	47,804 [11.8, 00.0]	29,603 [03.5, 37.5]	31,384 [06.7, 17.1]
Sadu-Cho	136,996 [21.2, 08.8]	60,035 [08.9, 00.0]	31,785 [03.1, 25.0]	45,176 [09.2, 10.0]
Swarna	128,040 [24.3, 07.6]	53,511 [08.4, 00.0]	32,899 [04.4, 17.2]	41,630 [11.5, 08.7]
Zhenshan 97B	98,935 [24.0, 16.1]	38,397 [08.2, 11.1]	29,792 [05.1, 16.7]	30,746 [10.7, 19.2]
Total	1,824,074 [23.8, 10.3]	761,606 [10.7, 02.9]	480,804 [03.2, 21.1]	581,664 [09.9, 13.7]

Table B.9.: Genome-wide predicted polymorphic regions (PR) and bases (PB) counts and evaluation by variety (precision ≈ 80 % across all varieties assessed on GSP) in comparison to the number of SNPs in MBML. The fraction of polymorphic bases with respect to the total numbers of queried positions (PR: 98,176,752, MBML: 100,104,806) are given in percentage in parentheses. Precision and recall are given in percentage and are not reported (NR) where fewer than 60 known PRs were available for evaluation, because of very low statistical power.

Subgroup/Variety	No. PRs	No. PBs	Precision/Recall	SNPs in MBML
Temperate japonica				
LTH	76,671	1,930,588 (1.97)	NR/NR	19,547 (0.02)
M 202	73,725	1,819,683 (1.85)	NR/NR	25,258 (0.03)
Tainung 67	65,024	1,694,263 (1.73)	NR/NR	10,755 (0.01)
Tropical japonica				
Azucena	103,610	2,700,441 (2.75)	81.5/31.0	43,816 (0.04)
Cypress	105,313	2,572,337 (2.62)	77.1/21.9	55,957 (0.06)
Moroberekan	98,146	2,347,758 (2.39)	NR/NR	50,889 (0.05)
Aromatic				
Dom-sufid	118,858	2,938,808 (2.99)	NR/NR	51,817 (0.05)
Aus				
Dular	192,373	4,548,692 (4.63)	82.9/35.8	140,576 (0.14)
FR13 A	193,048	4,469,113 (4.55)	86.6/34.0	142,866 (0.14)
N 22	120,558	2,678,346 (2.73)	80.4/17.7	120,451 (0.12)
Rayada	165,389	3,859,283 (3.93)	83.3/23.3	144,562 (0.14)
Indica				
Aswina	188,446	4,263,204 (4.34)	90.5/39.4	151,662 (0.15)
IR64-21	166,537	4,048,716 (4.12)	87.0/27.3	130,244 (0.13)
Minghui 63	203,110	4,995,674 (5.09)	83.5/45.4	133,496 (0.13)
Pokkali	163,674	4,102,187 (4.18)	84.3/20.6	125,448 (0.13)
SHZ2	129,840	3,443,155 (3.51)	85.4/13.7	108,791 (0.11)
Sadu-Cho	202,941	4,820,233 (4.91)	79.6/42.5	136,996 (0.14)
Swarna	149,845	3,804,920 (3.88)	85.2/20.5	128,040 (0.13)
Zhenshan 97B	97,525	2,327,647 (2.37)	72.5/12.3	98,935 (0.10)

Table B.10.: Composition of the gold standard set of polymorphisms (GSP) and 93-11 polymorphism set (93-11P). Numbers are given in bp; counts per 10 kb are indicated in parentheses.

	GSP		93-11P	
Total	1,743,128		91,265,021	
Polymorphisms	14,530	(83)	717,695	(79)
SNPs	9,414	(54)	436,709	(48)
Insertions	727	(4)	60,135	(7)
Deleted Bases	4,389	(25)	220,851	(24)

Table B.11.: Performance in terms of precision and recall assessed on the 93-11P data set by variety. The mean of precision and recall is given in the last column.

Subgroup	Variety	Precision	Recall	Mean
Temperate japonica				
	LTH	24.4	3.9	14.2
	M 202	33.0	5.1	19.1
	Tainung 67	19.8	2.8	11.3
Tropical japonica				
	Azucena	31.8	7.6	19.7
	Cypress	35.2	8.4	21.8
	Moroberekan	36.5	7.9	22.2
Aromatic				
	Dom-sufid	36.0	9.9	23.0
Aus				
	Dular	48.4	23.3	35.9
	FR13 A	44.8	21.1	33.0
	N 22	39.4	10.5	25.0
	Rayada	40.6	16.1	28.4
Indica				
	Aswina	53.8	24.9	39.3
	IR64-21	48.0	19.0	33.5
	Minghui 63	53.8	27.7	40.8
	Pokkali	40.6	15.5	28.1
	SHZ2	41.2	12.3	26.8
	Sadu-Cho	54.8	28.2	41.5
	Swarna	41.5	14.8	28.2
	Zhenshan 97B	42.5	9.1	25.8

Table B.12.: Fraction of polymorphic bases with respect to the total numbers of queried positions in percentage for the primer PR sets with different recall cut-offs. The numbers are given by variety and averaged for the five major subgroups.

	Recall cut-off								
Subgroup/Variety	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Temperate japonica	0.81	1.43	2.70	4.77	7.54	14.44	21.31	37.45	66.27
LTH	0.85	1.52	2.90	5.14	8.11	15.34	22.47	39.13	68.48
M 202	0.87	1.50	2.75	4.67	7.20	13.26	19.29	33.85	62.87
Tainung 67	0.72	1.27	2.46	4.49	7.32	14.70	22.16	39.38	67.45
Tropical japonica	1.08	1.92	3.51	5.94	9.14	16.46	23.21	38.47	65.92
Azucena	1.10	2.00	3.76	6.52	10.21	18.64	26.49	43.66	71.19
Cypress	1.09	1.91	3.47	5.82	8.99	16.30	22.79	37.21	64.16
Moroberekan	1.05	1.83	3.30	5.46	8.23	14.46	20.34	34.54	62.40
Aromatic	1.21	2.12	3.86	6.48	9.81	17.11	23.81	38.97	66.16
Dom-sufid	1.21	2.12	3.86	6.48	9.81	17.11	23.81	38.97	66.16
Aus	1.51	2.78	5.05	8.28	12.15	20.32	27.38	42.19	68.21
Dular	1.89	3.37	5.84	9.23	13.21	21.35	28.36	43.15	69.31
FR13 A	1.65	3.10	5.60	9.13	13.37	22.25	29.60	44.53	69.81
N 22	1.05	1.96	3.74	6.32	9.41	16.00	21.98	35.83	63.15
Rayada	1.45	2.69	5.02	8.44	12.63	21.68	29.59	45.25	70.56
Indica	1.58	2.91	5.35	8.82	13.05	21.93	29.42	44.62	70.15
Aswina	1.85	3.33	5.68	8.70	12.12	18.95	24.81	38.15	64.27
IR64-21	1.57	2.93	5.51	9.21	13.72	22.90	30.38	44.98	69.48
Minghui 63	2.19	3.81	6.49	10.13	14.51	23.08	30.05	43.98	68.73
Pokkali	1.32	2.64	5.26	9.22	14.15	24.41	32.75	49.10	74.59
SHZ2	1.24	2.41	4.79	8.39	12.99	22.95	30.77	45.22	68.30
Sadu-Cho	2.12	3.69	6.25	9.76	13.84	22.27	29.69	45.12	71.25
Swarna	1.38	2.65	5.20	8.99	13.70	23.89	32.38	48.92	74.52
Zhenshan 97B	0.94	1.83	3.59	6.17	9.39	17.00	24.51	41.52	70.05
Average over all	1.34	2.45	4.50	7.49	11.20	19.29	26.43	41.71	68.25

Table B.13.: Number of PRs with score ≥ 0.9 in the set of non-redundant PRs (switch cost 10) by variety and average numbers for the five major subgroups.

Subgroup	Variety	Number of PRs
Temperate japonica		75,330
	LTH	79,782
	M 202	77,295
	Tainung 67	68,912
Tropical japonica		107,535
	Azucena	117,077
	Cypress	106,732
	Moroberekan	98,796
Aromatic		118,925
	Dom-sufid	118,925
Aus		159,889
	Dular	196,320
	FR13 A	182,563
	N 22	107,007
	Rayada	153,666
Indica		162,790
	Aswina	180,481
	IR64-21	166,597
	Minghui 63	211,195
	Pokkali	157,179
	SHZ2	131,859
	Sadu-Cho	204,465
	Swarna	152,286
	Zhenshan 97B	98,259
Average over all		137,337

Table B.14.: Pfam protein domains enriched in PRs. The domains are sorted by the degree of PR disruption. The length of the disrupted domains versus the total length of the domains within a domain family is reported in the last column. The fraction of these numbers are given in percentage in parentheses.

Domain	ID	Number	Disruption
MATH domain	PF00917	11	1,918/4,126 (46.5)
MORN repeat	PF02493	15	470/1,020 (46.1)
NF-X1 type zinc finger	PF01422	10	268/623 (43.0)
Pumilio-family RNA binding repeat	PF00806	28	942/2,924 (32.2)
Leucine Rich Repeat	PF00560	3,938	71,245/266,917 (26.7)
Tetratricopeptide repeat	PF00515	95	2,424/9,442 (25.7)
Leucine Rich Repeat (LRR)	PF07723	47	906/3,544 (25.6)
Kelch motif	PF01344	18	656/2697 (24.3)
Armadillo/beta-catenin-like repeat	PF00514	30	863/3,696 (23.3)
WD domain, G-beta repeat	PF00400	135	3,275/14,343 (22.8)
Zinc finger C-x8-C-x5-C-x3-H type	PF00642	33	551/2,574 (21.4)
PPR repeat	PF01535	1,372	26,791/136,199 (19.7)
SWIM zinc finger	PF04434	21	440/2,391 (18.4)
AT hook motif	PF02178	45	304/1,707 (17.8)
F-box domain	PF00646	91	2,327/13,538 (17.2)
Tetratricopeptide repeat	PF07719	18	303/1,818 (16.7)
Transposase DDE domain	PF01609	12	949/5,787 (16.4)
Zinc finger, C3HC4 type (RING finger)	PF00097	19	408/2,600 (15.7)
Zinc finger, C2H2 type	PF00096	40	423/2,765 (15.3)
Protein tyrosine kinase	PF07714	28	366/2,537 (14.4)
IQ calmodulin-binding motif	PF00612	52	434/3,224 (13.5)
EF hand	PF00036	24	258/1,956 (13.2)
HEAT repeat	PF02985	77	1,097/8,413 (13.0)
Ankyrin repeat	PF00023	206	1,715/16,483 (10.4)

B.2. rQuant: Modelling Biases for Accurate RNA-seq-based Transcript Quantification

Table B.15.: Evaluation of rQuant on artificial data. Spearman's correlation coefficient between true and inferred transcript abundances was calculated across genes and averaged per gene. Sample sizes are given in brackets. Correlation for the baseline method (rQuant without read density estimation) is grouped by true expression (low: < 500 molecules, medium: 500 to 1,500 molecules, high: $\geq$ 1,500 molecules), transcript length (short: $<$ 1,000 nt, medium: 1,000 to 2,000 nt, high: $\geq$ 2,000 nt) and number of transcripts at one gene locus. Results for rQuant with estimated profile model that was determined empirically and by optimisation are listed. For comparison, transcript abundances were estimated by a segment-based version of rQuant, Cufflinks [171, 200] and MISO [105]. The results for rQuant baseline and rQuant with profiles are highlighted.

Approach		Spearman's correlation across genes		Spearman's correlation per gene	
rQuant baseline		**0.889**	[10,180]	**0.779**	[3,023]
	By expression				
	low	0.492	[3,770]	0.652	[721]
	medium	0.651	[3,279]	0.800	[1,310]
	high	0.874	[3,131]	0.844	[992]
	By transcript length				
	short	0.930	[3,272]	0.764	[818]
	medium	0.920	[4,680]	0.843	[1,489]
	long	0.803	[2,228]	0.665	[716]
	By number of transcripts				
	1	0.979	[2,945]	n/a	
	2	0.881	[4,500]	0.786	[2,250]
	3	0.845	[1,515]	0.768	[505]
	4 and more	0.775	[1,220]	0.746	[268]
rQuant profiles					
	empirical	0.912	[10,180]	0.809	[3,022]
	optimal	**0.918**	[10,180]	**0.816**	[3,022]
Other methods					
	rQuant segment-based	0.737	[10,195]	0.535	[3,023]
	Cufflinks	0.870	[10,180]	0.740	[3,023]
	Cufflinks bias corrected	0.875	[10,180]	0.777	[3,022]
	MISO	n/a		0.674	[2,927]

Table B.16.: Evaluation of rQuant on artificial data by gene complexity. Pearson's correlation between true and inferred transcript abundances was calculated across multiple-transcript and single-transcript genes grouped by true expression (low: < 500 molecules, medium: 500 to 1,500 molecules, high: $\geq$ 1,500 molecules), transcript length (short: $<$ 1,000 nt, medium: 1,000 to 2,000 nt, high: $\geq$ 2,000 nt) and number of transcripts at one gene locus. Sample sizes are given in brackets. Results for rQuant with estimated profile model that was determined empirically and by optimisation are listed. For comparison, transcript abundances were estimated by a segment-based version of rQuant, Cufflinks [171, 200] and MISO [105]. The results for rQuant baseline and rQuant with profiles are highlighted.

Approach		Pearson's correlation			
		multiple-transcript		single-transcript	
rQuant baseline		**0.913**	[7,235]	**0.996**	[2,945]
	By expression				
	low	0.086	[2,707]	0.842	[1,063]
	medium	0.422	[2,309]	0.850	[970]
	high	0.910	[2,219]	0.996	[912]
	By transcript length				
	short	0.996	[1,842]	0.999	[1,430]
	medium	0.990	[3,595]	0.998	[1,085]
	long	0.693	[1,798]	0.979	[430]
	By number of transcripts				
	1	n/a		0.996	[2,945]
	2	0.984	[4,500]	n/a	
	3	0.723	[1,515]	n/a	
	4 and more	0.614	[1,220]	n/a	
rQuant profiles					
	empirical	**0.939**	[7,235]	0.994	[2,945]
	optimal	0.921	[7,235]	**0.997**	[2,945]
Other methods					
	rQuant segment-based	0.933	[7,235]	0.933	[2,960]
	Cufflinks	0.900	[7,235]	0.996	[2,945]
	Cufflinks bias corrected	0.906	[7,235]	0.995	[7,235]
	MISO	0.825	[6,827]	n/a	

Table B.17.: Evaluation of rQuant on artificial data by gene complexity. Spearman's correlation between true and inferred transcript abundances was calculated across multiple-transcript and single-transcript genes grouped by true expression (low: < 500 molecules, medium: 500 to 1,500 molecules, high: $\geq$ 1,500 molecules), transcript length (short: $<$ 1,000 nt, medium: 1,000 to 2,000 nt, high: $\geq$ 2,000 nt) and number of transcripts at one gene locus. Sample sizes are given in brackets. Results for rQuant with estimated profile model that was determined empirically and by optimisation are listed. For comparison, transcript abundances were estimated by a segment-based version of rQuant, Cufflinks [171, 200] and MISO [105]. The results for rQuant baseline and rQuant with profiles are highlighted.

Approach		Spearman's correlation			
		multiple-transcript		single-transcript	
rQuant baseline		**0.854**	[7,235]	**0.979**	[2,945]
	By expression				
	low	0.389	[2,707]	0.844	[1,063]
	medium	0.582	[2,309]	0.848	[970]
	high	0.855	[2,219]	0.925	[912]
	By transcript length				
	short	0.889	[1,842]	0.989	[1,430]
	medium	0.898	[2,595]	0.997	[1,085]
	long	0.771	[1,798]	0.965	[430]
	By number of transcripts				
	1	n/a		0.979	[2,945]
	2	0.881	[4,500]	n/a	
	3	0.845	[1,515]	n/a	
	4 and more	0.775	[1,220]	n/a	
rQuant profiles					
	empirical	0.883	[7,235]	**0.986**	[2,945]
	optimal	**0.893**	[7,235]	0.981	[2,945]
Other methods					
	rQuant segment-based	0.673	[7,235]	0.918	[2,960]
	Cufflinks	0.827	[7,235]	0.977	[2,945]
	Cufflinks bias corrected	0.837	[7,235]	0.972	[7,235]
	MISO	0.823	[6,827]	n/a	

C. Supplementary Formulas

C.1. rQuant: Modelling Biases for Accurate RNA-seq-based Transcript Quantification

C.1.1. Finding the Optimal Profile Weights

Loss Term $\mathcal{L}_{exon}$

$$\mathcal{L}_{exon} = \gamma^E \sum_{p=1}^{P} \left(\sum_{t=1}^{T} w_t \; \Theta(\boldsymbol{\theta}, p, t) \; B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) - c_p \right)^2$$

$$= \gamma^E \sum_{\substack{p:\; d(p,\cdot) \geq x_{f'-1} \\ \wedge\; d(p,\cdot) < x_{f'+1}}} \left(\sum_{t=1}^{T} w_t \; \Theta(\boldsymbol{\theta}, p, t) \; B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) - c_p \right)^2$$

$$+ \underbrace{\gamma^E \sum_{\substack{p:\; d(p,\cdot) < x_{f'-1} \\ \vee\; d(p,\cdot) \geq x_{f'+1}}} \left(\sum_{t=1}^{T} w_t \; \Theta(\boldsymbol{\theta}, p, t) \; B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) - c_p \right)^2}_{R_3^\theta}$$

$$= \gamma^E \sum_{\substack{p:\; d(p,\cdot) \geq x_{f'-1} \\ \wedge\; d(p,\cdot) < x_{f'+1}}} \left(\sum_{\substack{t:\; n_t = n' \;\wedge \\ d(p,t) \geq x_{f'} \;\wedge \\ d(p,t) < x_{f'+1}}} w_t \; B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) \; \left(\theta_{f',n'} \; (1 - \delta_{p,t}) + \theta_{f'+1,n'} \; \delta_{p,t} \right) \right.$$

$$+ \sum_{\substack{t:\; n_t = n' \;\wedge \\ d(p,t) < x_{f'} \;\wedge \\ d(p,t) \geq x_{f'-1}}} w_t \; B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) \; \left(\theta_{f'-1,n'} \; (1 - \delta_{p,t}) + \theta_{f',n'} \; \delta_{p,t} \right)$$

$$\left. + \sum_{t:\; n_t \neq n'} w_t \; \Theta(\boldsymbol{\theta}, p, t) \; B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) - c_p \right)^2 + R_3^\theta$$

$$= \gamma^E \sum_{\substack{p:\; d(p,\cdot) \geq x_{f'-1} \\ \wedge\; d(p,\cdot) < x_{f'+1}}} \left(\theta_{f',n'} \; \underbrace{\left(\sum_{\substack{t:\; n_t = n' \;\wedge \\ d(p,t) \geq x_{f'} \;\wedge \\ d(p,t) < x_{f'+1}}} w_t \; B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) \; (1 - \delta_{p,t}) + \sum_{\substack{t:\; n_t = n' \;\wedge \\ d(p,t) < x_{f'} \;\wedge \\ d(p,t) \geq x_{f'-1}}} w_t \; B(\boldsymbol{x}_{p,t}, \boldsymbol{\beta}) \; \delta_{p,t} \right)}_{R_1^\theta} \right.$$

$$+ \underbrace{\sum_{\substack{t:\ n_t=n' \ \wedge \\ d(p,t)\geq x_{f'} \ \wedge \\ d(p,t)<x_{f'+1}}} w_t \ B(\boldsymbol{x}_{p,t},\boldsymbol{\beta}) \ \theta_{f'+1,n'} \ \delta_{p,t} + \sum_{\substack{t:\ n_t=n' \ \wedge \\ d(p,t)<x_{f'} \ \wedge \\ d(p,t)\geq x_{f'-1}}} w_t \ B(\boldsymbol{x}_{p,t},\boldsymbol{\beta}) \ \theta_{f'-1,n'} \ (1-\delta_{p,t})}_{R_2^\theta}$$

$$+ \underbrace{\sum_{t:\ n_t\neq n'} w_t \ \Theta(\boldsymbol{\theta},p,t) \ B(\boldsymbol{x}_{p,t},\boldsymbol{\beta}) - c_p}_{R_2^\theta} \Bigg)^2 + R_3^\theta$$

$$= {\theta_{f',n'}}^2 \ \gamma^E \sum_{\substack{p:\ d(p,\cdot)\geq x_{f'-1} \\ \wedge\ d(p,\cdot)<x_{f'+1}}} {R_1^\theta}^2 + \theta_{f',n'} \ \gamma^E \sum_{\substack{p:\ d(p,\cdot)\geq x_{f'-1} \\ \wedge\ d(p,\cdot)<x_{f'+1}}} 2 \ R_1^\theta \ R_2^\theta + \gamma^E \sum_{\substack{p:\ d(p,\cdot)\geq x_{f'-1} \\ \wedge\ d(p,\cdot)<x_{f'+1}}} {R_2^\theta}^2 + R_3^\theta$$

Coupling Supporting Points $\mathcal{R}^F(\boldsymbol{\theta})$

$$\mathcal{R}^F(\boldsymbol{\theta}) = \sum_{n=1}^{N} \sum_{f=1}^{F-1} (\theta_{f,n} - \theta_{f+1,n})^2$$

$$= \left(\theta_{f',n'} - \theta_{f'+1,n'}\right)^2 + \left(\theta_{f'-1,n'} - \theta_{f',n'}\right)^2$$
$$+ \sum_{n=n'} \sum_{f\neq f'} (\theta_{f,n} - \theta_{f+1,n})^2 + \sum_{n\neq n'} \sum_{f=1}^{F-1} (\theta_{f,n} - \theta_{f+1,n})^2$$

$$= 2 \ {\theta_{f',n'}}^2 + \theta_{f',n'} \ \underbrace{\left(-2 \ \theta_{f'+1,n'} - 2 \ \theta_{f'-1,n'}\right)}_{R_1^F}$$

$$\underbrace{+ {\theta_{f'+1,n'}}^2 + {\theta_{f'-1,n'}}^2 + \sum_{n=n'} \sum_{f\neq f'} (\theta_{f,n} - \theta_{f+1,n})^2 + \sum_{n\neq n'} \sum_{f=1}^{F-1} (\theta_{f,n} - \theta_{f+1,n})^2}_{R_2^F}$$

This does not hold for the following special cases and the result term needs to be adapted accordingly:

1. $f' = F$: The first summand in the second equation vanishes.
2. $f' = 1$: The second summand in the second equation vanishes.

Coupling Transcript Length Bins $\mathcal{R}^N(\boldsymbol{\theta})$

$$\mathcal{R}^N(\boldsymbol{\theta}) = \sum_{f=1}^{F} \sum_{n=1}^{N-1} \left(\theta_{f,n} - \theta_{f,n+1}\right)^2$$

$$= \left(\theta_{f',n'} - \theta_{f',n'+1}\right)^2 + \left(\theta_{f',n'-1} - \theta_{f',n'}\right)^2$$
$$+ \sum_{f=f'} \sum_{n \neq n'} \left(\theta_{f,n} - \theta_{f,n+1}\right)^2 + \sum_{f \neq f'} \sum_{n=1}^{N-1} \left(\theta_{f,n} - \theta_{f,n+1}\right)^2$$

$$= 2\ {\theta_{f',n'}}^2 + \theta_{f',n'} \underbrace{\left(-2\ \theta_{f',n'+1} - 2\ \theta_{f',n'-1}\right)}_{R_1^N}$$

$$\underbrace{+ {\theta_{f',n'+1}}^2 + {\theta_{f',n'-1}}^2 + \sum_{f=f'} \sum_{n \neq n'} \left(\theta_{f,n} - \theta_{f,n+1}\right)^2 + \sum_{f \neq f'} \sum_{n=1}^{N-1} \left(\theta_{f,n} - \theta_{f,n+1}\right)^2}_{R_2^N}$$

This does not hold for the following special cases and the result term needs to be adapted accordingly:

1. $n' = N$: The first summand in the second equation vanishes.
2. $n' = 1$: The second summand in the second equation vanishes.

C.1.2. Evaluation Measures

Pearson's Correlation Coefficient

Pearson's correlation coefficient is given by:

$$\rho^P(\boldsymbol{x}, \boldsymbol{y}) = \frac{\sum_{n=1}^{N} \left(x_n - \overline{x}\right)\left(y_n - \overline{y}\right)}{\sqrt{\left(\sum_{n=1}^{N} x_n - \overline{x}\right)^2 \left(\sum_{n=1}^{N} y_n - \overline{y}\right)^2}}$$

to compare two samples $\boldsymbol{x}$ and $\boldsymbol{y}$ of size N assuming linear dependence of the samples.

Spearman's Correlation Coefficient

Spearman's correlation coefficient is defined as:

$$\rho^S(\boldsymbol{x}, \boldsymbol{y}) = 1 - \frac{6\ \sum_{n=1}^{N} \left(x_n - y_n\right)^2}{N\left(N^2 - 1\right)}$$

to compare two rank samples $\boldsymbol{x}$ and $\boldsymbol{y}$ of size N assuming monotonic relatedness of the samples.

Bibliography

[1] *Structural Bioinformatics*. John Wiley & Sons, Inc., 2003.

[2] The Gencode Gene Set, 2011. URL `http://www.gencodegenes.org/releases.html`.

[3] International Rice Research Institute: Rice Facts, 2011. URL `http://irri.org/about-rice/rice-facts`.

[4] NGS machines statistics, 2011. URL `http://pathogenomics.bham.ac.uk/hts/stats`.

[5] Octave, 2011. URL `http://www.gnu.org/software/octave/`.

[6] Python, 2011. URL `http://www.python.org/`.

[7] The RNA-seq Genome Annotation Assessment Project (RGASP), 2011. URL `http://www.gencodegenes.org/rgasp/`.

[8] SAMTools, 2011. URL `http://samtools.sourceforge.net/`.

[9] 1000 Genomes Project Consortium. A map of human genome variation from population-scale sequencing. *Nature*, 467(7319):1061–73, 2010.

[10] A. Agarwal, D. Koppstein, J. Rozowsky, A. Sboner, L. Habegger, L. W. Hillier, R. Sasidharan, V. Reinke, R. H. Waterston, and M. Gerstein. Comparison and calibration of transcriptome data from RNA-Seq and tiling arrays. *BMC Genomics*, 11:383, 2010.

[11] C. Alkan, B. P. Coe, and E. E. Eichler. Genome structural variation discovery and genotyping. *Nature Reviews Genetics*, 12(5):363–76, 2011.

[12] Y. Altun, I. Tsochantaridis, and T. Hofmann. Hidden Markov Support Vector Machines. In *Proceedings of the 20th International Conference on Machine Learning (ICML 2003)*, pages 3–10. AAAI Press, 2003.

[13] S. Anders and W. Huber. Differential expression analysis for sequence count data. *Genome Biology*, 11 (10):R106, 2010.

[14] O. T. Avery, C. M. Macleod, and M. McCarty. Studies on the chemical nature of the substance inducing transformation of pneumococcal types: induction of transformation by a desoxyribonucleic acid fraction isolated from Pneumococcus type III. *The Journal of Experimental Medicine*, 79(2):137–58, 1944.

[15] Y. Barash, J. A. Calarco, W. Gao, Q. Pan, X. Wang, O. Shai, B. J. Blencowe, and B. J. Frey. Deciphering the splicing code. *Nature*, 465(7294):53–9, 2010.

[16] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains. *The Annals of Mathematical Statistics*, 41(1): 164–171, 1970.

[17] J. Behr, R. Bohnert, G. Zeller, G. Schweikert, L. Hartmann, and G. Rätsch. Next generation genome annotation with mGene.ngs. *BMC Bioinformatics*, 11(Suppl 10):O8, 2010.

[18] J. Bella, K. L. Hindle, P. A. McEwan, and S. C. Lovell. The leucine-rich repeat structure. *Cellular and Molecular Life Sciences*, 65(15):2307–33, 2008.

[19] A. Ben-Hur, C. S. Ong, S. Sonnenburg, B. Schölkopf, and G. Rätsch. Support vector machines and kernels for computational biology. *PLoS Computational Biology*, 4(10):e1000173, 2008.

[20] K. P. Bennett and O. L. Mangasarian. Robust Linear Programming Discrimination of Two Linearly Inseparable Sets. *Optimization Methods and Software*, pages 23–34, 1992.

[21] K. P. Bennett and E. Parrado-Hernández. The Interplay of Optimization and Machine Learning Research. *Journal of Machine Learning Research*, 7:1265–1281, 2006.

[22] D. R. Bentley, S. Balasubramanian, H. P. Swerdlow, G. P. Smith, J. Milton, C. G. Brown, K. P. Hall, D. J. Evers, C. L. Barnes, H. R. Bignell, et al. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, 456(7218):53–9, 2008.

[23] P. Biosciences. SMRT Future Applications, 2011.

[24] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, Berlin, 2007.

[25] A. K. Björklund, D. Ekman, and A. Elofsson. Expansion of protein domain repeats. *PLoS Computational Biology*, 2(8):e114, 2006.

[26] D. Blankenberg, G. Von Kuster, N. Coraor, G. Ananda, R. Lazarus, M. Mangan, A. Nekrutenko, and J. Taylor. Galaxy: a web-based genome analysis tool for experimentalists. *Current Protocols in Molecular Biology*, Chapter 19:Unit 19.10.1–21, 2010.

[27] R. Bohnert. Polymorphism Detection in Rice. *Diplom Thesis*, 2007.

[28] R. Bohnert and G. Rätsch. rQuant.web: a tool for RNA-Seq-based transcript quantitation. *Nucleic Acids Research*, 38(Web Server issue):W348–51, 2010.

[29] R. Bohnert, G. Zeller, R. Clark, K. Childs, V. Ulat, R. Stokowski, D. Ballinger, K. Frazer, D. Cox, R. Bruskiewich, et al. Revealing sequence variation patterns in rice with machine learning methods. *BMC Bioinformatics*, 9(Suppl 10):O8, 2008.

[30] R. Bohnert, J. Behr, and G. Rätsch. Transcript quantification with RNA-Seq data. *BMC Bioinformatics*, 10(Suppl 13):P5, 2009.

[31] B. M. Bolstad, R. A. Irizarry, M. Astrand, and T. P. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–93, 2003.

[32] K. L. Borden and P. S. Freemont. The RING finger domain: a recent example of a sequence-structure family. *Current Opinion in Structural Biology*, 6(3):395–401, 1996.

[33] P. Bork and R. F. Doolittle. Drosophila kelch motif is derived from a common enzyme fold. *Journal of Molecular Biology*, 236(5):1277–82, 1994.

[34] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[35] D. Branton, D. W. Deamer, A. Marziali, H. Bayley, S. A. Benner, T. Butler, M. Di Ventra, S. Garaj, A. Hibbs, X. Huang, et al. The potential and challenges of nanopore sequencing. *Nature Biotechnology*, 26(10):1146–53, 2008.

[36] C. Burge and S. Karlin. Prediction of complete gene structures in human genomic DNA. *Journal of Molecular Biology*, 268(1):78–94, 1997.

[37] A. L. Caicedo, S. H. Williamson, R. D. Hernandez, A. Boyko, A. Fledel-Alon, T. L. York, N. R. Polato, K. M. Olsen, R. Nielsen, S. R. McCouch, et al. Genome-wide patterns of nucleotide polymorphism in domesticated rice. *PLoS Genetics*, 3(9):1745–56, 2007.

[38] J. Cao, K. Schneeberger, S. Ossowski, T. Günther, S. Bender, J. Fitz, D. Koenig, C. Lanz, O. Stegle, C. Lippert, et al. Whole-genome sequencing of multiple Arabidopsis thaliana populations. *Nature Genetics*, 2011.

[39] M. Chen and J. L. Manley. Mechanisms of alternative splicing regulation: insights from molecular and genomics approaches. *Nature Reviews Molecular Cell Biology*, 10(11):741–54, 2009.

[40] V. G. Cheung and R. S. Spielman. Genetics of human gene expression: mapping DNA variants that influence gene expression. *Nature Reviews Genetics*, 10(9):595–604, 2009.

[41] R. M. Clark, G. Schweikert, C. Toomajian, S. Ossowski, G. Zeller, P. Shinn, N. Warthmann, T. T. Hu, G. Fu, D. A. Hinds, et al. Common sequence polymorphisms shaping genetic diversity in Arabidopsis thaliana. *Science*, 317(5836):338–42, 2007.

[42] A. Coghlan, T. J. Fiedler, S. J. McKay, P. Flicek, T. W. Harris, D. Blasiar, nGASP Consortium, and L. D. Stein. nGASP–the nematode genome annotation assessment project. *BMC Bioinformatics*, 9:549, 2008.

[43] C. Cortes and V. N. Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.

[44] D. J. Cutler, M. E. Zwick, M. M. Carrasquillo, C. T. Yohn, K. P. Tobin, C. Kashuk, D. J. Mathews, N. A. Shah, E. E. Eichler, J. A. Warrington, et al. High-throughput variation detection and genotyping using microarrays. *Genome Research*, 11(11):1913–25, 2001.

[45] R. Dahm. Friedrich Miescher and the discovery of DNA. *Developmental Biology*, 278(2):274–88, 2005.

[46] R. Dahm. Discovering DNA: Friedrich Miescher and the early years of nucleic acid research. *Human Genetics*, 122(6):565–81, 2008.

[47] L. D. D'Andrea and L. Regan. TPR proteins: the versatile helix. *Trends in Biochemical Sciences*, 28 (12):655–62, 2003.

[48] G. B. Dantzig. *Linear programming and extensions*. Princeton University Press, 11th edition, 1998.

[49] J. W. Davey, J. L. Davey, M. L. Blaxter, and M. W. Blaxter. RADSeq: next-generation population genetics. *Briefings in Functional Genomics*, 9(5-6):416–23, 2010.

[50] F. De Bona, S. Ossowski, K. Schneeberger, and G. Rätsch. Optimal spliced alignments of short sequence reads. *Bioinformatics*, 24(16):i174–80, 2008.

[51] J. E. Dennis and R. B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. Society for Industrial and Applied Mathematics, 1996.

[52] M. A. DePristo, E. Banks, R. Poplin, K. V. Garimella, J. R. Maguire, C. Hartl, A. A. Philippakis, G. Del Angel, M. A. Rivas, M. Hanna, et al. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, 43(5):491–8, 2011.

[53] J. C. Dohm, C. Lottaz, T. Borodina, and H. Himmelbauer. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic Acids Research*, 36(16):e105, 2008.

[54] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, 2002.

[55] T. A. Edwards, S. E. Pyle, R. P. Wharton, and A. K. Aggarwal. Structure of Pumilio reveals similarity between RNA and peptide binding motifs. *Cell*, 105(2):281–9, 2001.

[56] J. Eid, A. Fehr, J. Gray, K. Luong, J. Lyle, G. Otto, P. Peluso, D. Rank, P. Baybayan, B. Bettman, et al. Real-time DNA sequencing from single polymerase molecules. *Science*, 323(5910):133–8, 2009.

[57] B. Ewing and P. Green. Base-calling of automated sequencer traces using phred. II. Error probabilities. *Genome Research*, 8(3):186–94, 1998.

[58] J. Feng, W. Li, and T. Jiang. Inference of isoforms from short sequence reads. *Journal of Computational Biology*, 18(3):305–21, 2011.

[59] S. A. Filichkin, H. D. Priest, S. A. Givan, R. Shen, D. W. Bryant, S. E. Fox, W.-K. Wong, and T. C. Mockler. Genome-wide mapping of alternative splicing in Arabidopsis thaliana. *Genome Research*, 20 (1):45–58, 2010.

[60] R. D. Finn, J. Tate, J. Mistry, P. C. Coggill, S. J. Sammut, H.-R. Hotz, G. Ceric, K. Forslund, S. R. Eddy, E. L. L. Sonnhammer, et al. The Pfam protein families database. *Nucleic Acids Research*, 36 (Database issue):D281–8, 2008.

[61] R. D. Finn, J. Mistry, J. Tate, P. Coggill, A. Heger, J. E. Pollington, O. L. Gavin, P. Gunasekaran, G. Ceric, K. Forslund, et al. The Pfam protein families database. *Nucleic Acids Research*, 38(Database issue):D211–22, 2010.

[62] V. Franc, A. Zien, and B. Schölkopf. Support Vector Machines as Probabilistic Models. In *Proceedings of the 28th International Conference on Machine Learning*, 2011.

[63] K. A. Frazer, E. Eskin, H. M. Kang, M. A. Bogue, D. A. Hinds, E. J. Beilharz, R. V. Gupta, J. Montgomery, M. M. Morenzoni, G. B. Nilsen, et al. A sequence-based variation map of 8.27 million SNPs in inbred mouse strains. *Nature*, 448(7157):1050–3, 2007.

[64] X. Gan, O. Stegle, J. Behr, J. G. Steffen, P. Drewe, K. L. Hildebrand, R. Lyngsoe, S. J. Schultheiss, E. J. Osborne, V. T. Sreedharan, et al. Multiple reference genomes and transcriptomes for Arabidopsis thaliana. *Nature*, 2011.

[65] G. K. Geiss, R. E. Bumgarner, B. Birditt, T. Dahl, N. Dowidar, D. L. Dunaway, H. P. Fell, S. Ferree, R. D. George, T. Grogan, et al. Direct multiplexed measurement of gene expression with color-coded probe pairs. *Nature Biotechnology*, 26(3):317–25, 2008.

[66] B. Giardine, C. Riemer, R. C. Hardison, R. Burhans, L. Elnitski, P. Shah, Y. Zhang, D. Blankenberg, I. Albert, J. Taylor, et al. Galaxy: a platform for interactive large-scale genome analysis. *Genome Research*, 15(10):1451–5, 2005.

[67] J. Goecks, A. Nekrutenko, J. Taylor, and Galaxy Team. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biology*, 11(8):R86, 2010.

[68] S. A. Goff, D. Ricke, T.-H. Lan, G. Presting, R. Wang, M. Dunn, J. Glazebrook, A. Sessions, P. Oeller, H. Varma, et al. A draft sequence of the rice genome (Oryza sativa L. ssp. japonica). *Science*, 296 (5565):92–100, 2002.

[69] N. Görnitz, G. Zeller, J. Behr, A. Kahles, P. Mudrakarta, S. Sonnenburg, and G. Rätsch. mTiM: margin-based transcript identification from RNA-seq. In *RECOMB-seq 2011*, 2011.

[70] M. G. Grabherr, B. J. Haas, M. Yassour, J. Z. Levin, D. A. Thompson, I. Amit, X. Adiconis, L. Fan, R. Raychowdhury, Q. Zeng, et al. Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nature Biotechnology*, 2011.

[71] B. D. Gregory, J. Yazaki, and J. R. Ecker. Utilizing tiling microarrays for whole-genome analysis in plants. *The Plant Journal*, 53(4):636–44, 2008.

[72] M. Griffith, O. L. Griffith, J. Mwenifumbo, R. Goya, A. S. Morrissy, R. D. Morin, R. Corbett, M. J. Tang, Y.-C. Hou, T. J. Pugh, et al. Alternative expression analysis by RNA sequencing. *Nature Methods*, 7(10):843–7, 2010.

[73] R. Guigó and M. G. Reese. EGASP: collaboration through competition to find human genes. *Nature Methods*, 2(8):575–7, 2005.

[74] R. Guigó, P. Flicek, J. F. Abril, A. Reymond, J. Lagarde, F. Denoeud, S. Antonarakis, M. Ashburner, V. B. Bajic, E. Birney, et al. EGASP: the human ENCODE Genome Annotation Assessment Project. *Genome Biology*, 7 Suppl 1:S2.1–31, 2006.

[75] B. Han and Y. Xue. Genome-wide intraspecific DNA-sequence variations in rice. *Current Opinion in Plant Biology*, 6(2):134–8, 2003.

[76] J. Han, J. Xiong, D. Wang, and X.-D. Fu. Pre-mRNA splicing: where and when in the nucleus. *Trends in Cell Biology*, 21(6):336–43, 2011.

[77] K. D. Hansen, S. E. Brenner, and S. Dudoit. Biases in Illumina transcriptome sequencing caused by random hexamer priming. *Nucleic Acids Research*, 2010.

[78] T. D. Harris, P. R. Buzby, H. Babcock, E. Beer, J. Bowers, I. Braslavsky, M. Causey, J. Colonell, J. Dimeo, J. W. Efcavitch, et al. Single-molecule DNA sequencing of a viral genome. *Science*, 320 (5872):106–9, 2008.

[79] L. W. Hillier, V. Reinke, P. Green, M. Hirst, M. A. Marra, and R. H. Waterston. Massively parallel sequencing of the polyadenylated transcriptome of C. elegans. *Genome Research*, 19(4):657–66, 2009.

[80] D. A. Hinds, L. L. Stuve, G. B. Nilsen, E. Halperin, E. Eskin, D. G. Ballinger, K. A. Frazer, and D. R. Cox. Whole-genome patterns of common DNA variation in three human populations. *Science*, 307 (5712):1072–9, 2005.

[81] D. A. Hinds, A. P. Kloek, M. Jen, X. Chen, and K. A. Frazer. Common deletions and SNPs are in linkage disequilibrium in the human genome. *Nature Genetics*, 38(1):82–5, 2006.

[82] A. E. Hoerl and R. W. Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1):55–67, 1970.

[83] I. L. Hofacker, W. Fontana, P. F. Stadler, L. S. Bonhoeffer, M. Tacker, and P. Schuster. Fast Folding and Comparison of RNA Secondary Structures. *Monatshefte für Chemie / Chemical Monthly*, 125:167–188, 1994.

[84] D. Holste and U. Ohler. Strategies for identifying RNA splicing regulatory motifs and predicting alternative splicing events. *PLoS Computational Biology*, 4(1):e21, 2008.

[85] D. S. Horner, G. Pavesi, T. Castrignanò, P. D. De Meo, S. Liuni, M. Sammeth, E. Picardi, and G. Pesole. Bioinformatics approaches for genomics and post genomics applications of next-generation sequencing. *Briefings in Bioinformatics*, 11(2):181–97, 2010.

[86] B. E. Howard and S. Heber. Towards reliable isoform quantification using RNA-SEQ data. *BMC Bioinformatics*, 11 Suppl 3:S6, 2010.

[87] T. T. Hu, P. Pattyn, E. G. Bakker, J. Cao, J.-F. Cheng, R. M. Clark, N. Fahlgren, J. A. Fawcett, J. Grimwood, H. Gundlach, et al. The Arabidopsis lyrata genome sequence and the basis of rapid genome size change. *Nature Genetics*, 43(5):476–81, 2011.

[88] X. Huang, Q. Feng, Q. Qian, Q. Zhao, L. Wang, A. Wang, J. Guan, D. Fan, Q. Weng, T. Huang, et al. High-throughput genotyping by whole-genome resequencing. *Genome Research*, 19(6):1068–76, 2009.

[89] X. Huang, X. Wei, T. Sang, Q. Zhao, Q. Feng, Y. Zhao, C. Li, C. Zhu, T. Lu, Z. Zhang, et al. Genome-wide association studies of 14 agronomic traits in rice landraces. *Nature Genetics*, 42(11):961–7, 2010.

[90] D. H. Huson, S. Mitra, H.-J. Ruscheweyh, N. Weber, and S. C. Schuster. Integrative analysis of environmental sequences using MEGAN4. *Genome Research*, 21(9):1552–60, 2011.

[91] C. A. Hutchison, 3rd. DNA sequencing: bench to bedside and beyond. *Nucleic Acids Research*, 35(18): 6227–37, 2007.

[92] IBM. IBM ILOG CPLEX Optimizer, 2011. URL `http://www.ibm.com/software/integration/optimization/cplex-optimizer/`.

[93] N. H. G. R. Institute. Understanding the Human Genome Project, 2011. URL `http://www.genome.gov/EdKit/bio2j.html`.

[94] International HapMap Consortium. A haplotype map of the human genome. *Nature*, 437(7063):1299–320, 2005.

[95] International HapMap Consortium, K. A. Frazer, D. G. Ballinger, D. R. Cox, D. A. Hinds, L. L. Stuve, R. A. Gibbs, J. W. Belmont, A. Boudreau, P. Hardenbol, et al. A second generation human haplotype map of over 3.1 million SNPs. *Nature*, 449(7164):851–61, 2007.

[96] International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–45, 2004.

[97] International Rice Genome Sequencing Project. The map-based sequence of the rice genome. *Nature*, 436:793–800, 2005.

[98] G. Jean, A. Kahles, V. T. Sreedharan, F. De Bona, and G. Rätsch. RNA-Seq read alignments with PALMapper. *Current Protocols in Bioinformatics*, Chapter 11:Unit 11.6, 2010.

[99] H. Jiang and W. H. Wong. Statistical inferences for isoform expression in RNA-Seq. *Bioinformatics*, 25 (8):1026–32, 2009.

[100] L. Jiang, F. Schlesinger, C. A. Davis, Y. Zhang, R. Li, M. Salit, T. R. Gingeras, and B. Oliver. Synthetic spike-in standards for RNA-seq experiments. *Genome Research*, 2011.

[101] Y. Jin, Y. Yang, and P. Zhang. New insights into RNA secondary structure in the alternative splicing of pre-mRNAs. *RNA Biology*, 8(3), 2011.

[102] A. Kahles, P. Ribeca, R. Bohnert, J. Behr, and G. Rätsch. Assessing RNA-seq alignment algorithms. *i.p.*, 2011.

[103] A. V. Kajava. Structural diversity of leucine-rich repeat proteins. *Journal of Molecular Biology*, 277(3): 519–27, 1998.

[104] M. W. Karaman, S. Groshen, C.-C. Lee, B. L. Pike, and J. G. Hacia. Comparisons of substitution, insertion and deletion probes for resequencing and mutational analysis using oligonucleotide microarrays. *Nucleic Acids Research*, 33(3):e33, 2005.

[105] Y. Katz, E. T. Wang, E. M. Airoldi, and C. B. Burge. Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nature Methods*, 7(12):1009–15, 2010.

[106] M. Kertesz, Y. Wan, E. Mazor, J. L. Rinn, R. C. Nutter, H. Y. Chang, and E. Segal. Genome-wide measurement of RNA secondary structure in yeast. *Nature*, 467(7311):103–7, 2010.

[107] S. Kim, V. Plagnol, T. T. Hu, C. Toomajian, R. M. Clark, S. Ossowski, J. R. Ecker, D. Weigel, and M. Nordborg. Recombination and linkage disequilibrium in Arabidopsis thaliana. *Nature Genetics*, 39 (9):1151–5, 2007.

[108] B. Kobe and A. V. Kajava. The leucine-rich repeat as a protein recognition motif. *Current Opinion in Structural Biology*, 11(6):725–32, 2001.

[109] A. Krogh, M. Brown, I. S. Mian, K. Sjölander, and D. Haussler. Hidden Markov models in computational biology. Applications to protein modeling. *Journal of Molecular Biology*, 235(5):1501–31, 1994.

[110] S. Kurtz, A. Phillippy, A. L. Delcher, M. Smoot, M. Shumway, C. Antonescu, and S. L. Salzberg. Versatile and open software for comparing large genomes. *Genome Biology*, 5(2):R12, 2004.

[111] J. Lafferty, A. McCallum, and F. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the 18th Annual International Conference on Machine Learning (ICML 2001)*, pages 282–289. Morgan Kaufmann, 2001.

[112] J. H. Laity, B. M. Lee, and P. E. Wright. Zinc finger proteins: new insights into structural and functional diversity. *Current Opinion in Structural Biology*, 11(1):39–46, 2001.

[113] E. Lalonde, K. C. H. Ha, Z. Wang, A. Bemmo, C. L. Kleinman, T. Kwan, T. Pastinen, and J. Majewski. RNA sequencing reveals the role of splicing polymorphisms in regulating human gene expression. *Genome Research*, 21(4):545–54, 2011.

[114] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh, et al. Initial sequencing and analysis of the human genome. *Nature*, 409(6822): 860–921, 2001.

[115] E. Lechner, P. Achard, A. Vansiri, T. Potuschak, and P. Genschik. F-box proteins everywhere. *Current Opinion in Plant Biology*, 9(6):631–8, 2006.

[116] J. Z. Levin, M. Yassour, X. Adiconis, C. Nusbaum, D. A. Thompson, N. Friedman, A. Gnirke, and A. Regev. Comprehensive comparative analysis of strand-specific RNA sequencing methods. *Nature Methods*, 7(9):709–15, 2010.

[117] B. Lewin. *Genes IX*. Jones and Barlett Publishers, Inc., 2008.

[118] B. Li and C. N. Dewey. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics*, 12(1):323, 2011.

[119] B. Li, V. Ruotti, R. M. Stewart, J. A. Thomson, and C. N. Dewey. RNA-Seq gene expression estimation with read mapping uncertainty. *Bioinformatics*, 26(4):493–500, 2010.

[120] H. Li and R. Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–60, 2009.

[121] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin, and 1000 Genome Project Data Processing Subgroup. The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16):2078–9, 2009.

[122] J. Li, H. Jiang, and W. H. Wong. Modeling non-uniformity in short-read rates in RNA-Seq data. *Genome Biology*, 11(5):R50, 2010.

[123] W. Li, J. Feng, and T. Jiang. IsoLasso: A LASSO Regression Approach to RNA-Seq Based Transcriptome Assembly. *Proceedings of the 15th Annual International Conference on Research in Computational Molecular Biology*, 6577:168–188, 2011.

[124] S. E. V. Linsen, E. de Wit, G. Janssens, S. Heater, L. Chapman, R. K. Parkin, B. Fritz, S. K. Wyman, E. de Bruijn, E. E. Voest, et al. Limitations and possibilities of small RNA digital gene expression profiling. *Nature Methods*, 6(7):474–6, 2009.

[125] D. Lipson, T. Raz, A. Kieu, D. R. Jones, E. Giladi, E. Thayer, J. F. Thompson, S. Letovsky, P. Milos, and M. Causey. Quantification of the yeast transcriptome by single-molecule sequencing. *Nature Biotechnology*, 27(7):652–8, 2009.

[126] X. Liu, T. Lu, S. Yu, Y. Li, Y. Huang, T. Huang, L. Zhang, J. Zhu, Q. Zhao, D. Fan, et al. A collection of 10,096 indica rice full-length cDNAs reveals highly expressed sequence divergence between Oryza sativa indica and japonica subspecies. *Plant Molecular Biology*, 65(4):403–15, 2007.

[127] H. Lodish, A. Berk, P. Matsudaira, C. A. Kaiser, M. Krieger, M. P. Scott, S. L. Zipursky, and J. Darnell. *Molecular Cell Biology*. W. H. Freeman and Company, 5th edition, 2004.

[128] T. Lu, G. Lu, D. Fan, C. Zhu, W. Li, Q. Zhao, Q. Feng, Y. Zhao, Y. Guo, W. Li, et al. Function annotation of the rice transcriptome at single-nucleotide resolution by RNA-seq. *Genome Research*, 20 (9):1238–49, 2010.

[129] Z.-Q. Luo and P. Tseng. Error bounds and convergence analysis of feasible descent methods: a general approach. *Annals of Operations Research*, 46(157-178), 1993.

[130] T. F. C. Mackay, E. A. Stone, and J. F. Ayroles. The genetics of quantitative traits: challenges and prospects. *Nature Reviews Genetics*, 10(8):565–77, 2009.

[131] J. Majewski and T. Pastinen. The study of eQTL variations by RNA-seq: from SNPs to phenotypes. *Trends in Genetics*, 27(2):72–9, 2011.

[132] M. Margulies, M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bemben, J. Berka, M. S. Braverman, Y.-J. Chen, Z. Chen, et al. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–80, 2005.

[133] J. C. Marioni, C. E. Mason, S. M. Mane, M. Stephens, and Y. Gilad. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Research*, 18(9):1509–17, 2008.

[134] J. Martin, V. M. Bruno, Z. Fang, X. Meng, M. Blow, T. Zhang, G. Sherlock, M. Snyder, and Z. Wang. Rnnotator: an automated de novo transcriptome assembly pipeline from stranded RNA-Seq reads. *BMC Genomics*, 11:663, 2010.

[135] MathWorks. MATLAB, 2011. URL `http://www.mathworks.com/products/matlab/`.

[136] A. M. Maxam and W. Gilbert. A new method for sequencing DNA. *Proceedings of the National Academy of Sciences of the USA*, 74(2):560–4, 1977.

[137] M. I. McCarthy, G. R. Abecasis, L. R. Cardon, D. B. Goldstein, J. Little, J. P. A. Ioannidis, and J. N. Hirschhorn. Genome-wide association studies for complex traits: consensus, uncertainty and challenges. *Nature Reviews Genetics*, 9(5):356–69, 2008.

[138] C. J. McManus and B. R. Graveley. RNA structure and the mechanisms of alternative splicing. *Current Opinion on Genetics & Development*, 2011.

[139] K. L. McNally, R. Bruskiewich, D. Mackill, C. R. Buell, J. E. Leach, and H. Leung. Sequencing multiple and diverse rice varieties. Connecting whole-genome variation with phenotypes. *Plant Physiology*, 141 (1):26–31, 2006.

[140] K. L. McNally, K. L. Childs, R. Bohnert, R. M. Davidson, K. Zhao, V. J. Ulat, G. Zeller, R. M. Clark, D. R. Hoen, T. E. Bureau, et al. Genomewide SNP variation reveals relationships among landraces and modern varieties of rice. *Proceedings of the National Academy of Sciences of the USA*, 106(30):12273–8, 2009.

[141] M. L. Metzker. Emerging technologies in DNA sequencing. *Genome Research*, 15(12):1767–76, 2005.

[142] M. L. Metzker. Sequencing technologies - the next generation. *Nature Reviews Genetics*, 11(1):31–46, 2010.

[143] M. Meyerson, S. Gabriel, and G. Getz. Advances in understanding cancer genomes through second-generation sequencing. *Nature Reviews Genetics*, 11(10):685–96, 2010.

[144] F. Miescher. Ueber die chemische Zusammensetzung der Eiterzellen. *Medicinisch-chemische Untersuchungen*, 4:441–460, 1871.

[145] T. C. Mockler, S. Chan, A. Sundaresan, H. Chen, S. E. Jacobsen, and J. R. Ecker. Applications of DNA tiling arrays for whole-genome analysis. *Genomics*, 85(1):1–15, 2005.

[146] L. Monna, R. Ohta, H. Masuda, A. Koike, and Y. Minobe. Genome-wide searching of single-nucleotide polymorphisms among eight distantly and closely related rice cultivars (Oryza sativa L.) and a wild accession (Oryza rufipogon Griff.). *DNA Research*, 13(2):43–51, 2006.

[147] S. B. Montgomery, M. Sammeth, M. Gutierrez-Arcelus, R. P. Lach, C. Ingle, J. Nisbett, R. Guigo, and E. T. Dermitzakis. Transcriptome genetics using second generation sequencing in a Caucasian population. *Nature*, 464(7289):773–7, 2010.

[148] A. Mortazavi, B. A. Williams, K. McCue, L. Schaeffer, and B. Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature Methods*, 5(7):621–8, 2008.

[149] A. Y. Ng and M. I. Jordan. On Discriminative vs. Generative Classifiers: A comparison of logistic regression and naive Bayes. In *Proceedings of the 2001 Neural Information Processing Systems (NIPS) Conference*, volume 14, 2001.

[150] N. Nguyen and Y. Guo. Comparisons of Sequence Labeling Algorithms and Extensions. In *Proceedings of the 24th Annual International Conference on Machine Learning (ICML 2007)*, pages 681–688. Omnipress, 2007.

[151] M. Nicolae, S. Mangul, I. I. Mandoiu, and A. Zelikovsky. Estimation of alternative splicing isoform frequencies from RNA-Seq data. *Algorithms for Molecular Biology*, 6(1):9, 2011.

[152] M. Nordborg and D. Weigel. Next-generation genetics in plants. *Nature*, 456(7223):720–3, 2008.

[153] S. Ossowski, K. Schneeberger, R. M. Clark, C. Lanz, N. Warthmann, and D. Weigel. Sequencing of natural strains of Arabidopsis thaliana with short reads. *Genome Research*, 18(12):2024–33, 2008.

[154] S. Ouyang, W. Zhu, J. Hamilton, H. Lin, M. Campbell, K. Childs, F. Thibaud-Nissen, R. L. Malek, Y. Lee, L. Zheng, et al. The TIGR Rice Genome Annotation Resource: improvements and new features. *Nucleic Acids Research*, 35(Database issue):D883–7, 2007.

[155] F. Ozsolak and P. M. Milos. RNA sequencing: advances, challenges and opportunities. *Nature Reviews Genetics*, 12(2):87–98, 2011.

[156] F. Ozsolak, A. R. Platt, D. R. Jones, J. G. Reifenberger, L. E. Sass, P. McInerney, J. F. Thompson, J. Bowers, M. Jarosz, and P. M. Milos. Direct RNA sequencing. *Nature*, 461(7265):814–8, 2009.

[157] F. Ozsolak, A. Goren, M. Gymrek, M. Guttman, A. Regev, B. E. Bernstein, and P. M. Milos. Digital transcriptome profiling from attomole-level RNA samples. *Genome Research*, 20(4):519–25, 2010.

[158] Q. Pan, O. Shai, L. J. Lee, B. J. Frey, and B. J. Blencowe. Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nature Genetics*, 40(12):1413–5, 2008.

[159] P. J. Park. ChIP-seq: advantages and challenges of a maturing technology. *Nature Reviews Genetics*, 10(10):669–80, 2009.

[160] D. Parkhomchuk, T. Borodina, V. Amstislavskiy, M. Banaru, L. Hallen, S. Krobitsch, H. Lehrach, and A. Soldatov. Transcriptome analysis by strand-specific sequencing of complementary DNA. *Nucleic Acids Research*, 37(18):e123, 2009.

[161] T. Pastinen. Genome-wide allele-specific analysis: insights into regulatory variation. *Nature Reviews Genetics*, 11(8):533–8, 2010.

[162] S. Pepke, B. Wold, and A. Mortazavi. Computation for ChIP-seq and RNA-seq studies. *Nature Methods*, 6(11 Suppl):S22–32, 2009.

[163] T. T. Perkins, R. A. Kingsley, M. C. Fookes, P. P. Gardner, K. D. James, L. Yu, S. A. Assefa, M. He, N. J. Croucher, D. J. Pickard, et al. A strand-specific RNA-Seq analysis of the transcriptome of the typhoid bacillus Salmonella typhi. *PLoS Genetics*, 5(7):e1000569, 2009.

[164] J. F. Petrosino, S. Highlander, R. A. Luna, R. A. Gibbs, and J. Versalovic. Metagenomic pyrosequencing and microbial identification. *Clinical Chemistry*, 55(5):856–66, 2009.

[165] J. K. Pickrell, J. C. Marioni, A. A. Pai, J. F. Degner, B. E. Engelhardt, E. Nkadori, J.-B. Veyrieras, M. Stephens, Y. Gilad, and J. K. Pritchard. Understanding mechanisms underlying human gene expression variation with RNA sequencing. *Nature*, 464(7289):768–72, 2010.

[166] G. Ramsay. DNA chips: state-of-the art. *Nature Biotechnology*, 16(1):40–4, 1998.

[167] G. Rätsch, G. Jean, J. Behr, A. Kahles, and R. Bohnert. Accurate read mapping and simultaneous transcript identification and quantification. In *Genome Informatics 2011*, 2011.

[168] Rice Annotation Project, T. Itoh, T. Tanaka, R. A. Barrero, C. Yamasaki, Y. Fujii, P. B. Hilton, B. A. Antonio, H. Aono, R. Apweiler, et al. Curated genome annotation of Oryza sativa ssp. japonica and comparative genome analysis with Arabidopsis thaliana. *Genome Research*, 17(2):175–83, 2007.

[169] Rice Annotation Project, T. Tanaka, B. A. Antonio, S. Kikuchi, T. Matsumoto, Y. Nagamura, H. Numa, H. Sakai, J. Wu, T. Itoh, et al. The Rice Annotation Project Database (RAP-DB): 2008 update. *Nucleic Acids Research*, 36(Database issue):D1028–33, 2008.

[170] H. Richard, M. H. Schulz, M. Sultan, A. Nürnberger, S. Schrinner, D. Balzereit, E. Dagand, A. Rasche, H. Lehrach, M. Vingron, et al. Prediction of alternative isoforms from exon expression levels in RNA-Seq experiments. *Nucleic Acids Research*, 38(10):e112, 2010.

[171] A. Roberts, C. Trapnell, J. Donaghey, J. L. Rinn, and L. Pachter. Improving RNA-Seq expression estimates by correcting for fragment bias. *Genome Biology*, 12(3):R22, 2011.

[172] G. Robertson, J. Schein, R. Chiu, R. Corbett, M. Field, S. D. Jackman, K. Mungall, S. Lee, H. M. Okada, J. Q. Qian, et al. De novo assembly and analysis of RNA-seq data. *Nature Methods*, 7(11): 909–12, 2010.

[173] M. Sammeth. The Flux Simulator, 2011. URL `http://flux.sammeth.net/simulator.html`.

[174] F. Sanger, G. M. Air, B. G. Barrell, N. L. Brown, A. R. Coulson, C. A. Fiddes, C. A. Hutchison, P. M. Slocombe, and M. Smith. Nucleotide sequence of bacteriophage phi X174 DNA. *Nature*, 265(5596): 687–95, 1977.

[175] F. Sanger, S. Nicklen, and A. R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the USA*, 74(12):5463–7, 1977.

[176] A. Sboner, X. J. Mu, D. Greenbaum, R. K. Auerbach, and M. B. Gerstein. The real cost of sequencing: higher than you think! *Genome Biology*, 12(8):125, 2011.

[177] B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

[178] S. J. Schultheiss. Ten Simple Rules for Providing a Scientific Web Resource. *PLoS Computational Biology*, 7(5):e1001126, 2011.

[179] S. J. Schultheiss, W. Busch, J. U. Lohmann, O. Kohlbacher, and G. Rätsch. KIRMES: kernel-based identification of regulatory modules in euchromatic sequences. *Bioinformatics*, 25(16):2126–33, 2009.

[180] S. J. Schultheiss, M.-C. Münch, G. D. Andreeva, and G. Rätsch. Persistence and availability of web services in computational biology. *PLoS One*, 6(9):e24914, 2011.

[181] G. Schweikert, J. Behr, A. Zien, G. Zeller, C. S. Ong, S. Sonnenburg, and G. Rätsch. mGene.web: a web service for accurate computational gene finding. *Nucleic Acids Research*, 37(Web Server issue):W312–6, 2009.

[182] G. Schweikert, A. Zien, G. Zeller, J. Behr, C. Dieterich, C. S. Ong, P. Philips, F. De Bona, L. Hartmann, A. Bohlen, et al. mGene: accurate SVM-based gene finding with an application to nematode genomes. *Genome Research*, 19(11):2133–43, 2009.

[183] S. Seeholzer, T. Tsuchimatsu, T. Jordan, S. Bieri, S. Pajonk, W. Yang, A. Jahoor, K. K. Shimizu, B. Keller, and P. Schulze-Lefert. Diversity at the Mla powdery mildew resistance locus from cultivated barley reveals sites of positive selection. *Molecular Plant-Microbe Interactions*, 23(4):497–509, 2010.

[184] H. Sela, J. Cheng, Y. Jun, E. Nevo, and T. Fahima. Divergent diversity patterns of NBS and LRR domains of resistance gene analogs in wild emmer wheat populations. *Genome*, 52(6):557–65, 2009.

[185] T. Sera. Zinc-finger-based artificial transcription factors and their applications. *Advanced Drug Delivery Reviews*, 61(7-8):513–26, 2009.

[186] Y.-J. Shen, H. Jiang, J.-P. Jin, Z.-B. Zhang, B. Xi, Y.-Y. He, G. Wang, C. Wang, L. Qian, X. Li, et al. Development of genome-wide DNA polymorphism database for map-based cloning of rice genes. *Plant Physiology*, 135(3):1198–205, 2004.

[187] J. Shendure and H. Ji. Next-generation DNA sequencing. *Nature Biotechnology*, 26(10):1135–45, 2008.

[188] J. Shendure, G. J. Porreca, N. B. Reppas, X. Lin, J. P. McCutcheon, A. M. Rosenbaum, M. D. Wang, K. Zhang, R. D. Mitra, and G. M. Church. Accurate multiplex polony sequencing of an evolved bacterial genome. *Science*, 309(5741):1728–32, 2005.

[189] L. M. Smith, L. Hartmann, P. Drewe, R. Bohnert, C. Lanz, and G. Rätsch. Illumina strand-specific paired-end adaptor ligation mRNA sequencing. *Submitted*, 2011.

[190] A. J. Smola, S. V. N. Vishwanathan, and Q. V. Le. Bundle Methods for Machine Learning. In *Proceedings of the 2007 Neural Information Processing Systems (NIPS) Conference*, volume 20. Curran Associates, Inc., 2007.

[191] S. Sonnenburg, G. Schweikert, P. Philips, J. Behr, and G. Rätsch. Accurate splice site prediction using support vector machines. *BMC Bioinformatics*, 8 Suppl 10:S7, 2007.

[192] S. Sonnenburg, G. Rätsch, S. Henschel, C. Widmer, J. Behr, A. Zien, F. De Bona, A. Binder, C. Gehl, and V. Franc. The SHOGUN Machine Learning Toolbox. *Journal of Machine Learning Research*, 11: 1799–1802, 2010.

[193] O. Stegle, P. Drewe, R. Bohnert, K. Borgwardt, and G. Rätsch. Statistical Tests for Detecting Differential RNA-Transcript Expression from Read Counts. *Available from Nature Precedings*, 2010.

[194] L. D. Stein, C. Mungall, S. Shu, M. Caudy, M. Mangone, A. Day, E. Nickerson, J. E. Stajich, T. W. Harris, A. Arva, et al. The generic genome browser: a building block for a model organism system database. *Genome Research*, 12(10):1599–610, 2002.

[195] D. Swarbreck, C. Wilks, P. Lamesch, T. Z. Berardini, M. Garcia-Hernandez, H. Foerster, D. Li, T. Meyer, R. Muller, L. Ploetz, et al. The Arabidopsis Information Resource (TAIR): gene structure and function annotation. *Nucleic Acids Research*, 36(Database issue):D1009–14, 2008.

[196] M. Sweeney and S. McCouch. The complex history of the domestication of rice. *Annals of Botany*, 100 (5):951–7, 2007.

[197] C. H. Teo, S. V. N. Vishwanathan, A. J. Smola, and Q. V. Le. Bundle Methods for Regularized Risk Minimization. *Journal of Machine Learning Research*, 11:311–365, 2010.

[198] R. Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 58(1):267–288, 1996.

[199] C. Trapnell, L. Pachter, and S. L. Salzberg. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, 25(9):1105–11, 2009.

[200] C. Trapnell, B. A. Williams, G. Pertea, A. Mortazavi, G. Kwan, M. J. van Baren, S. L. Salzberg, B. J. Wold, and L. Pachter. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature Biotechnology*, 28(5):511–5, 2010.

[201] P. Tseng. Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization. *Journal of Optimization Theory and Applications*, 109(3):475–494, 2001.

[202] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large Margin Methods for Structured and Interdependent Output Variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005.

[203] J. G. Underwood, A. V. Uzilov, S. Katzman, C. S. Onodera, J. E. Mainzer, D. H. Mathews, T. M. Lowe, S. R. Salama, and D. Haussler. FragSeq: transcriptome-wide RNA structure probing using high-throughput sequencing. *Nature Methods*, 7(12):995–1001, 2010.

[204] A. Valouev, J. Ichikawa, T. Tonthat, J. Stuart, S. Ranade, H. Peckham, K. Zeng, J. A. Malek, G. Costa, K. McKernan, et al. A high-resolution, nucleosome position map of C. elegans reveals a lack of universal sequence-dictated positioning. *Genome Research*, 18(7):1051–63, 2008.

[205] V. N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, Inc., 1998.

[206] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, et al. The sequence of the human genome. *Science*, 291(5507):1304–51, 2001.

[207] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, 1967.

[208] A. P. Vivancos, M. Güell, J. C. Dohm, L. Serrano, and H. Himmelbauer. Strand-specific deep sequencing of the transcriptome. *Genome Research*, 20(7):989–99, 2010.

[209] A. Wachter. Riboswitch-mediated control of gene expression in eukaryotes. *RNA Biology*, 7(1):67–76, 2010.

[210] M. C. Wahl, C. L. Will, and R. Lührmann. The spliceosome: design principles of a dynamic RNP machine. *Cell*, 136(4):701–18, 2009.

[211] Z. Wang, M. Gerstein, and M. Snyder. RNA-Seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1):57–63, 2009.

[212] J. D. Watson and F. H. C. Crick. Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid. *Nature*, 171(4356):737–8, 1953.

[213] D. Weigel and R. Mott. The 1001 genomes project for Arabidopsis thaliana. *Genome Biology*, 10(5): 107, 2009.

[214] D. Weigel and M. Nordborg. Natural variation in Arabidopsis. How do we find the causal genes? *Plant Physiology*, 138(2):567–8, 2005.

[215] Wikipedia. DNA Transcription, 2011. URL `http://upload.wikimedia.org/wikipedia/commons/3/36/DNA_transcription.svg`.

[216] Wikipedia. Microarray Hybridisation, 2011. URL `http://en.wikipedia.org/wiki/File:NA_hybrid.svg`.

[217] Wikipedia. Microarray Imaging, 2011. URL `http://upload.wikimedia.org/wikipedia/commons/0/0e/Microarray2.gif`.

[218] B. T. Wilhelm and J.-R. Landry. RNA-Seq-quantitative measurement of expression through massively parallel RNA-sequencing. *Methods*, 48(3):249–57, 2009.

[219] T. D. Wu and C. K. Watanabe. GMAP: a genomic mapping and alignment program for mRNA and EST sequences. *Bioinformatics*, 21(9):1859–75, 2005.

[220] Z. Xia, J. Wen, C.-C. Chang, and X. Zhou. NSMAP: A Method for Spliced Isoforms Identification and Quantification from RNA-Seq. *BMC Bioinformatics*, 12(1):162, 2011.

[221] G. Xu, H. Ma, M. Nei, and H. Kong. Evolution of F-box genes in plants: different modes of sequence divergence and their relationships with functional diversification. *Proceedings of the National Academy of Sciences of the USA*, 106(3):835–40, 2009.

[222] J. Yazaki, B. D. Gregory, and J. R. Ecker. Mapping the genome landscape using tiling array technology. *Current Opinion in Plant Biology*, 10(5):534–42, 2007.

[223] J. Yu, S. Hu, J. Wang, G. K.-S. Wong, S. Li, B. Liu, Y. Deng, L. Dai, Y. Zhou, X. Zhang, et al. A draft sequence of the rice genome (Oryza sativa L. ssp. indica). *Science*, 296(5565):79–92, 2002.

[224] E. M. Zdobnov and R. Apweiler. InterProScan–an integration platform for the signature-recognition methods in InterPro. *Bioinformatics*, 17(9):847–8, 2001.

[225] G. Zeller. *Machine Learning Algorithms for the Analysis of Data from Whole-Genome Tiling Microarrays*. PhD thesis, 2010.

[226] G. Zeller, R. M. Clark, K. Schneeberger, A. Bohlen, D. Weigel, and G. Rätsch. Detecting polymorphic regions in Arabidopsis thaliana with resequencing microarrays. *Genome Research*, 18(6):918–29, 2008.

[227] G. Zeller, S. R. Henz, S. Laubinger, D. Weigel, and G. Rätsch. Transcript normalization and segmentation of tiling array data. *Pacific Symposium on Biocomputing*, pages 527–38, 2008.

[228] B. Zhang, S. Kirov, and J. Snoddy. WebGestalt: an integrated system for exploring gene sets in various biological contexts. *Nucleic Acids Research*, 33(Web Server issue):W741–8, 2005.

[229] K. Zhao, M. Wright, J. Kimball, G. Eizenga, A. McClung, M. Kovach, W. Tyagi, M. L. Ali, C.-W. Tung, A. Reynolds, et al. Genomic diversity and introgression in O. sativa reveal the impact of domestication and breeding on the rice genome. *PLoS One*, 5(5):e10780, 2010.

[230] A. Zien and C. S. Ong. Multiclass Multiple Kernel Learning. In *Proceedings of the 24th International Conference on Machine Learning (ICML 2007)*, pages 1191–1198. Omnipress, 2007.

Publications

Core parts of the thesis are based on publications marked with • and author contributions are indicated. Publications marked with ◦ contain minor parts of the thesis; here, the relation to this dissertation is described.

Journal Articles

◦ Xiangchao Gan†, Oliver Stegle†, Jonas Behr†, Joshua G. Steffen†, Philipp Drewe†, Katie L. Hildebrand, Rune Lyngsoe, Sebastian J. Schultheiss, Edward J. Osborne, Vipin T. Sreedharan, André Kahles, **Regina Bohnert**, Géraldine Jean, Paul Derwent, Paul Kersey, Eric Belfield, Nicholas Harberd, Eric Kemen, Paula Kover*, Christopher Toomajian*, Richard M. Clark*, Gunnar Rätsch*, Richard Mott* Multiple reference genomes and transcriptomes for Arabidopsis thaliana. *Nature*, August 2011.

Relation to this dissertation: The algorithm designed for the generation of non-redundant polymorphic regions in *O. sativa* was re-used in this work to derive non-redundant polymorphic regions across 18 *A. thaliana* accessions.

• **Regina Bohnert** and Gunnar Rätsch. rQuant.web: a tool for RNA-Seq-based transcript quantitation. *Nucleic Acids Research*, 38(suppl2):W348–351, July 2010.

Author contributions: RB and GR conceived and designed the project. RB with contributions from GR designed the software; RB integrated rQuant.web into the MLB Galaxy instance; RB and GR wrote the paper.

• Kenneth L. McNally, Kevin L. Childs, **Regina Bohnert**, Rebecca M. Davidson, Keyan Zhao, Victor J. Ulat, Georg Zeller, Richard M. Clark, Douglas R. Hoen, Thomas E. Bureau, Renee Stokowski, Dennis G. Ballinger, Kelly A. Frazer, David R. Cox, Badri Padhukasahasram, Carlos D. Bustamante, Detlef Weigel, David J. Mackill, Richard M. Bruskiewich, Gunnar Rätsch, C. Robin Buell, Hei Leung, and Jan E. Leach. Genomewide SNP variation reveals relationships among landraces and modern varieties of rice. *Proceedings of the National Academy of Sciences of the United States of America*, July 2009.

Author contributions: KLM, DJM, CRB, HL, and JEL designed research; KLM, KLC, RMD, RS, DGB, KAF, DRC, and CRB performed research; KZ contributed new reagents/analytic tools; KLM, KLC, RB, RMD, KZ, VJU, GZ, RMC, DRH, TEB, BP, CDB, DW, RMB, GR, and JEL analyzed data; and KLM, KLC, RB, RMD, KZ, RMC, DW, GR, CRB, HL, and JEL wrote the paper.

†contributed equally
*corresponding authors

Other Publications

Jonas Behr, **Regina Bohnert**, Georg Zeller, Gabriele Schweikert, Lisa Hartmann, and Gunnar Rätsch. Next generation genome annotation with mGene.ngs. *BMC Bioinformatics*, 11(Suppl 10):O8, December 2010.

Oliver Stegle, Philipp Drewe, **Regina Bohnert**, Karsten Borgwardt, and Gunnar Rätsch. Statistical tests for detecting differential RNA-transcript expression from read counts. Available from Nature Precedings, May 2010.

• **Regina Bohnert**, Jonas Behr, and Gunnar Rätsch. Transcript quantification with RNA-Seq data. *BMC Bioinformatics*, 10(Suppl 13):P5, October 2009.

Author contributions: RB and GR conceived and designed the project. RB and GR developed the algorithm to quantify alternative transcripts with RNA-seq data and to simultaneously model experimental biases; RB performed experiments for data simulation and evaluation of the method; JB provided code for *de novo* transcript prediction; RB and GR wrote the abstract.

• **Regina Bohnert**, Georg Zeller, Richard M. Clark, Kevin L. Childs, Victor J. Ulat, Renee Stokowski, Dennis G. Ballinger, Kelly A. Frazer, David R. Cox, Richard M. Bruskiewich, C. Robin Buell, Jan E. Leach, Hei Leung, Kenneth L. McNally, Detlef Weigel, and Gunnar Rätsch. Revealing sequence variation patterns in rice with machine learning methods. *BMC Bioinformatics*, 9(Suppl 10):O8, October 2008.

Author contributions: RB with input from GZ and RMC analysed the resequencing data by using machine learning approaches for SNP calling and polymorphic region predictions; CEB, JEL, HL, KLM, DW and GR designed research; RB, GZ, RMC, KLC and CRB designed and generated the dideoxy sequencing set; DGB, KAF, DRC and RMB generated the resequencing data; RB with input from GZ and GR wrote the abstract.

Unpublished

Lisa M. Smith[†], Lisa Hartmann[†], Philipp Drewe, **Regina Bohnert**, Christa Lanz, and Gunnar Rätsch. Illumina strand-specific paired-end adaptor ligation mRNA sequencing. *Submitted*, 2011.

• **Regina Bohnert** and Gunnar Rätsch. rQuant: Modelling biases for accurate RNA-seq-based transcript quantification. *In preparation*, 2011.

Author contributions: RB and GR conceived and designed the project. RB and GR developed the algorithm to quantify alternative transcripts with RNA-seq data and to simultaneously model experimental biases; RB performed experiments for data simulation and evaluation of the method; RB and GR wrote the manuscript.

[†]contributed equally

Oral Presentations at International Conferences

Regina Bohnert. Quantitatively deconvolving alternative RNA secondary structures. *HiTSeq-SIG: Conference on High-throughput Sequencing Analysis and Algorithms preceding the 19th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB) and 10th European Conference on Computational Biology (ECCB)*, Vienna, Austria, July 2011.

Regina Bohnert. rQuant: Quantitative Detection of Alternative Transcripts with RNA-Seq Data. *HiTSeq-SIG: Conference on High-throughput Sequencing Analysis and Algorithms preceding the 17th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB) and 8th European Conference on Computational Biology (ECCB)*, Stockholm, Sweden, June 2009.

Regina Bohnert. Revealing sequence variation patterns in rice with machine learning methods. *4th ISCB Student Council Symposium preceding the 16th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB)*, Toronto, Canada, July 2008.