IMPROVED ALLELE SPECIFIC EXPRESSION (ASE) DETECTION AND THE IMPLICATIONS FOR UNDERSTANDING REGULATORY AND DISEASE GENETICS

Saukkonen, Anna

Awarding institution:
King's College London

IMPROVED ALLELE SPECIFIC EXPRESSION (ASE) DETECTION AND THE IMPLICATIONS FOR UNDERSTANDING REGULATORY AND DISEASE GENETICS

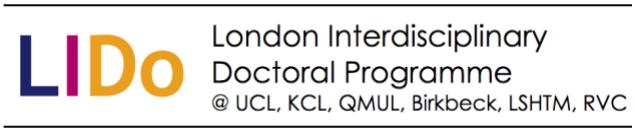
Anna Saukkonen

Student number 1788664

Department of Medical and Molecular Genetics

September 2022

Thesis submitted to King's College London in fulfilment of the degree of Doctor of Philosophy



DECLARATION

I declare that the work described in this thesis is my own unless specified otherwise.

ACKNOWLEDGEMENTS

I would like to express my sincerest gratitude to my primary supervisor Dr. Alan Hodgkinson. Words cannot describe how grateful I am for all the guidance and support while embarking on this PhD. Life and the pandemic didn't make this easy, but I have enjoyed working with you greatly. I have grown as a researcher and person more than I could have imagined. I would like to say a huge thank you to my secondary supervisor Dr. Helena Kilpinen for her supervision, challenging ideas and providing alternative views. I would like to thank my PhD committee for their insights throughout the years. I would like to thank LIDo and BBSRC for this opportunity and the funding that made this thesis possible. Completing this PhD is a dream come true.

PUBLICATION

Saukkonen A, Kilpinen H, Hodgkinson A. Highly accurate quantification of allelic gene expression for population and disease genetics. Genome Res. 2022 Jul 6;32(8):1565–72. doi: 10.1101/gr.276296.121. Epub ahead of print. PMID: 35794008; PMCID: PMC9435737.

The work in this thesis (Chapter 2: section 2.2.2 Simulated data, section 2.3.3 Developing PAC; Chapter 3: section 3.4 Validating performance of PAC; Chapter 4: section 4.3.1 Validating PAC: ASE vs eQTL aFC on GTEx whole blood data; Chapter 5: section 5.3.2 G×E analysis) has been published as part of Saukkonen *et al*., 2022.

ATTRIBUTIONS

The initial PAC pipeline (section 2.2.1) was conceived by Dr. Alan Hodkinson. GTEx samples were mapped by myself and Dr. Alan Hodkinson.

ABSTRACT

Gene expression plays a crucial role in phenotypic changes and disease. The regulation of gene expression is a complex process that involves genetics, environmental signals, epigenetics, and proteins. However, studying the interplay of these processes is key to better understanding the role of genetic variation and gene expression changes within important biological processes.

In genomic studies, gene expression changes are often studied in population-level data, however these analyses are often limited to studying the impacts of common variants on gene expression due to the limited power associated with rare variants in the populations under study. This may cause a problem, particular in small datasets focussed on rare disease, or when trying to understand the full range of genetic features that may modulate transcriptional events. Allele specific expression (ASE) offers an avenue to overcome these issues and consider the regulation of gene expression levels in smaller sample sizes, potentially capturing the impact of rare variants. ASE can also be used in combination with other population-based genetic studies to improve the overall signal.

However, ASE analysis suffers from a series of computational biases associated with short-read RNA-seq data and are particularly sensitive to sequence alignment errors driven by reads that overlap heterozygous variants. In this thesis, I have developed a Personalised ASE Caller (PAC) pipeline that improves heterozygous read alignment and reduces biases when quantifying allelic ratios. I have developed the pipeline into a streamlined tool using Nextflow and Docker technology and have made this tool available on my GitHub page for use by the scientific community.

I validated the performance of PAC against other commonly used methods showing that it significantly improves allelic quantification. I then show that PAC can identify ground truth signals in simulated data and can recapitulate population level signals better than other methods. I also demonstrate that PAC has utility in a disease context and that better allelic quantification has downstream consequences for interpreting biological data.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

ABBREVIATIONS

ASE	Allele specific expression
cASE	Conditional ASE
CNV	Copy number variant
eQTL	Expression quantitative trait loci
ENCODE	Encyclopedia of DNA Elements
GWA	Genome wide association
HipSci	Human Induced Pluripotent Stem Cell Initiative
HIS	Haploinsufficient
HS	Haplosufficient
iPSC	Induced pluripotent cell
LCL	Lymphoblast cell line
MAF	Minor allele frequency
PAC	Personalised ASE Caller
PGP	Platinum Genome Project
RAR	Reference allele ratio
reQTLs	Response eQTLs
SD	Standard deviation
SNP	Single nucleotide polymorphism
SNV	Single nucleotide variation
SV	Structural variation
TMP	Transcript per million
UTR	Untranslated region
WES	Whole exome sequencing
WGS	Whole genome sequencing

CHAPTER 1 –INTRODUCTION

1. INTRODUCTION

1.1 GENETIC VARIATION

1.1.1 EVOLUTIONARY SELECTION

The variation we see in the human genome and its architecture can be explained by evolutionary biology. According to natural selection, variants with large effects on fitness will be removed from the population and those that remain will be present at low rates. Common variants, on the other hand, are unlikely to have large effects on the fitness [1]. On average, altering protein-coding regions is more likely to have a detrimental effect on fitness as it is likely to affect protein structure and hence function. Therefore, rare variants tend to be within or close to the protein-coding regions. Conversely, altering noncoding regions tends to have no consequence or affect gene regulation. Common variants are therefore more often found in these regions [2].

Most human traits are polygenic, which means that natural selection acts on multiple variants at the same time and has more complicated effects on the fitness [3]. Here the selection pressure is distributed across multiple variants with smear effects, making cleaner signals more difficult to identify and interpret.

The relationship between effect sizes and the commonality of causal genetic variants tends to distribute along a spectrum, as illustrated in Figure 1 [4]. Here, rare variants with high penetrance tend to be within or near the coding regions and cause rare, severe diseases. On the other hand, due to lower selection, variants with low penetrance are more common in the population [5]. These are usually in the noncoding regions influencing gene regulation. However, the effective population size can influence the general notion of genetic architecture, where the strength of selection decreases with smaller populations [6]. In this case, the frequency of deleterious alleles can drift by chance to a detectable level [7].

In addition, there are exceptions to the genetic architecture, where rare variants play a role in common traits/diseases [8, 9], or where common variants contribute to rare diseases [10, 11], but these instances tend to be few in number. Although this general architecture might be biased as a consequence of different discovery methods used for rare with high

penetrance versus common with low penetrance variants [9], there are ongoing efforts to deeply sequence whole genomes of large cohorts in order to improve our understanding [12].

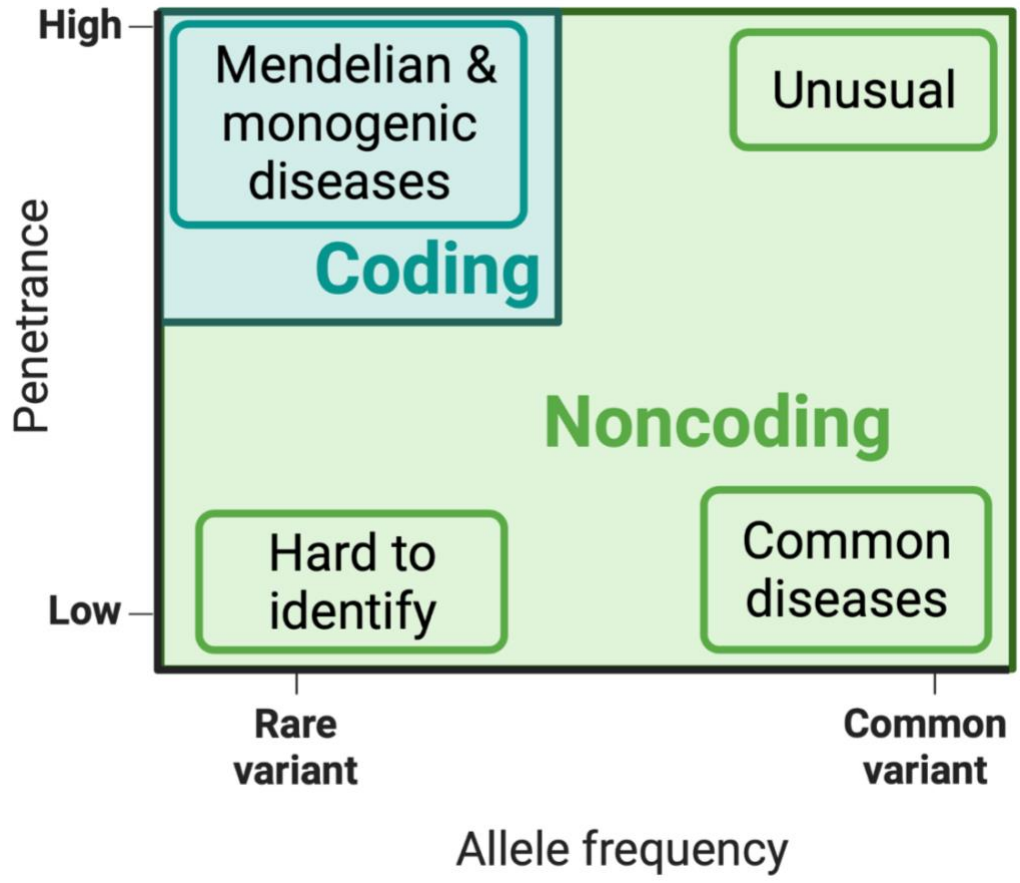


Figure 1**. Functional genetic architecture**.
The figure illustrates how allele frequencies and penetrance are distributed in human genetic architecture. Coding variants with high penetrance are most often rare. Mendelian and monogenic diseases tend to fall into this category. Common diseases usually in the noncoding regions most often have low penetrance but are common in the population. Adapted from McCarthy et al., 2008 and Lappalainen & MacArthur 2021 [4, 12]. Created with BioRender.com.

1.1.2 VARIATION IN THE HUMAN GENOME

The human genome was sequenced over 20 years ago [13, 14]. Following this, the main initial focus was to catalogue genetic variation in the human population, leading to massive collaborations that discovered the scale of variation across individuals, involving millions of variants across the genome that spanned a range of different predicted functional effects [15-20]. Some variants have harmful consequences leading to a disease such as the case in monogenic diseases. Some are benign variants that are tolerated in the population and do not cause disease [21], for example a mutation affecting cilantro taste preference [22]. And some variants might not have any detectable consequences at all. We still lack a great deal of understanding of genomic variation, and most variants identified from sequencing data have unknown consequences [23]. However, a major goal in the field of human genetics is to better understand the functional consequences of each genetic variant, particularly as the susceptibility to almost every human disease is affected by genetic variation to a certain degree [24].

The most direct way for a variant to influence disease susceptibility is by disrupting the coding sequence, and this class of variants is the cause of many genetic diseases, where the genetic code and gene annotations can aid in understanding how the variant might disrupt a protein. However, these mutations account for only a small proportion of the variation in the genome, as only 1% of DNA is protein-coding [25]. The vast majority of variants are in non-coding regions of the genome [26] and deciphering benign from pathogenic non-coding variants is more difficult since functional annotation of these regions is more complex and less complete.

On average there are 4.1-5 million sites where an individual genome differs from the reference genome, ~3.5-4.3 million being single nucleotide variants (SNVs) and ~0.5-0.6 million being short insertions/deletions (indels) depending on the population under study [17]. The vast majority of variants are benign and do not cause health consequences. However, on average every individual has over 100 protein truncating variants resulting in a premature stop codon, over 20 of which are rare in the human population and potentially deleterious [19]. There are computational algorithms to detect variants and differentiate them from sequencing errors, GATK [27] being the most commonly used, however, these

approaches also create errors and lead to confusion over the functional landscape of an individual's genome. In addition, around 8.5% of a genome is considered particularly difficult for variant calling [28]. These regions contain some clinically relevant genes [29], therefore better methods are needed to be able to study them.

Variant calling refers to the identification of variable sites from genomic sequencing data. It is important to accurately identify genetic variants from noise and sequencing errors to study their effects for example on gene expression, in disease fields or population genetics. There are numerous tools that attempt to identify genetic variation from sequencing data, but in general, almost all seek to combine multiple types of evidence to assign probability scores related to how likely it is that a variant is present at a particular location. These evidence types include, but are not limited to, the number of reads carrying reference and alternative alleles, sequencing depth, the locations of alternative alleles along each read, the direction of each read carrying alternative alleles, mapping and nucleotide quality scores, and known genetic variants [30]. Variant calling is particularly difficult in highly polymorphic or repetitive regions [31] and for genes expressed at low levels [32]. Amongst these approaches, GATK is a gold standard method [27] and is utilized across many sequencing studies. GATK is a probabilistic method where the genotypes are determined using Bayesian statistics, incorporating many of the features listed above. Here, the model calculates the probability for a given genotype given for example the base pairs qualities, error rate and read depths. Additional to variant callers, there are also other tools that seek to infer genetic information based on reference genomes. IMPUTE2 [33] is a method that utilises genotyped individuals of genetically similar populations to impute unobserved genotypes in individuals under study.

Structural variation (SV) is more severe and more difficult to deal with in genomics workflows. SVs are generally over 50 bp in size and include rearrangements such as deletions, duplications, insertions, inversions, copy number variants (CNVs), and mobile element insertions [29]. SVs can affect gene dosage, affect gene function, or rearrange genes or regulatory regions. Although SVs only account for around 0.2% of variants, they are responsible for 4-11.2% of rare high-impact coding variants [34].

Because knowledge of the consequences of genetic variants is important for the understanding of disease mechanisms, drug targets and biological pathways, many variant effect predictors have been developed. These tools often incorporate protein sequence, structural, evolutionary, epigenetic, and biophysical features to predict the effect a variant will have [23, 29]. Often, even the easiest-to-predict missense variants lack a high-quality prediction score, and even more so variants in the noncoding regions [12]. This is due to the limited understanding of regulatory regions and their environmental context. One way to verify variant consequences is experimentally, where the phenotypic consequences of variants are studied. For example, a functional assay has shown that glutamate oxidation is impaired in cultured fibroblasts derived in patients carrying a mutation in the *GC1* gene [35]. The link between mutation and phenotype in such studies can be examined by determining if the wild-type version of the gene rescues the phenotype, or by testing the consequences of variants in model organisms [36]. Another approach to understand the pathogenicity of variants is by considering mRNA expression from RNA-seq data, for example considering variants influencing splice events [37] or the loss of allelic expression [38]. Functional validation gives more information, yet it would be impossible to verify millions of variants separately. As such, experimental methods have been developed to characterise genetic variants by high-throughput protocol, where proteins containing different variants are generated and assayed for interactions or enzymatic activities [39]. However, this comes with its own restrictions, such as limited phenotype assessment. Therefore, computational methods remain the main method used to predict the effects of genetic variation at scale.

1.1.3 RARE VARIANTS

A large number of variants in the population are rare due to a human history of bottlenecks and recent expansions [40, 41]. During early human history the effective population size was small and some genetic variants that were present remain common in modern populations. However, since each generation gains around 100 new mutations, population growth has driven the accumulation of large numbers of rare variants. Understanding how rare variants contribute to human traits is lagging behind common variants. They are hard to study as most analyses rely on population level statistics and power, and therefore they are not well

represented, for example in genome-wide association (GWA) [42] studies. Rare variants are also often population specific [43, 44], are not often present on genotyping arrays [45], and they are poorly imputed with reference panels. The recognition of the importance of rare variants has led large efforts to design association studies which incorporate rare and low frequency variants [46], often through altered resequencing approaches.

1.2 GENETIC STUDIES TO UNDERSTAND DISEASE VARIANTS

1.2.1 EARLY STUDIES

Prior to modern sequencing technologies, linkage analysis and fine mapping in large multiplex families were used to identify and then sequence candidate disease genes [24]. These studies focused on rare and monogenic diseases and variant segregation was followed across multiple families and compared to healthy individuals. This process was time consuming and expensive. With the development and reducing costs of high throughput sequencing technologies, microarrays [47] and more importantly exome [48] and whole genome sequencing [49] have been crucial in genetic studies. Population-based studies have allowed the sequencing of many individuals without the need of relatedness, which can be difficult to obtain in some cases, facilitating the study of genetic differences in different populations and for different phenotypic traits. These experiments can study common, low penetrance alleles in addition to monogenic traits. With linkage studies, the genetic resolution is often poor. Therefore, population-based studies have driven the availability of vast amounts of genetic data, which can be utilised via *in silico* analysis methods to better understand disease and biological function. Linkage analysis is also difficult to perform on common diseases often involving multiple variants and genes [24], therefore GWA studies (GWAS) have also expanded knowledge in this area.

1.2.2 GWAS

GWAS detect associations between genotypes and phenotypes, where the allele frequencies of variants in individuals with the trait are compared to those without. Most

trait-associated variants identified by GWAS are in the noncoding regions of the genome [50] and therefore there is often ambiguity regarding the affected gene [51]. The GWAS hits within the noncoding genome are likely to affect gene regulation. A schematic of a GWAS study is shown in Figure 2. Often, the exact causal variant is not known due to linkage disequilibrium. Genomic variants nearby tend to correlate in GWAS with phenotypes due to haplotype blocks. Recombination mixes genomes but variants in close proximity to each other are more likely to be inherited together.

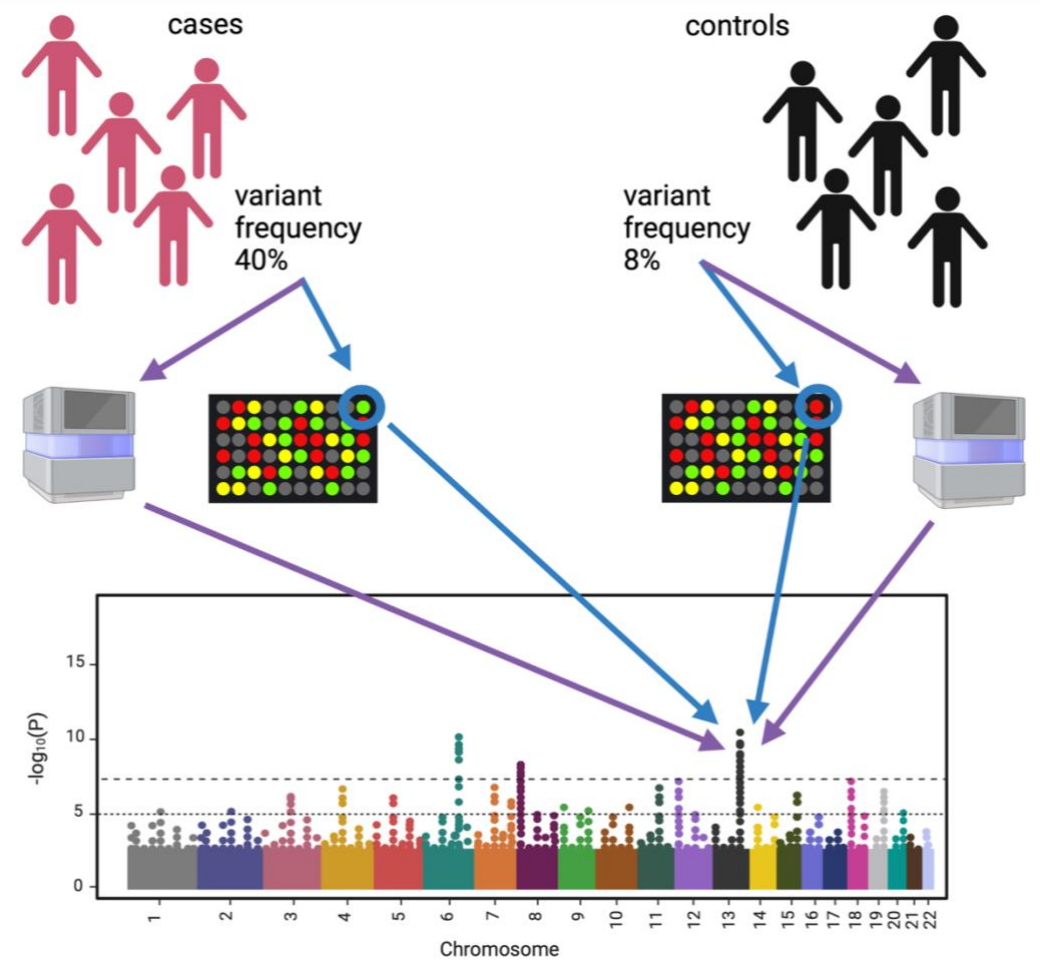


Figure 2. **An example of a case-control GWA study design.**
As an example, to illustrate GWA study, here genetic basis of a disease is under investigation. Subjects under study are divided into cases with those with a disease phenotype and into controls those without. The allelic frequency of each variant is investigated between cases and controls either by microarrays (blue pathway), which is most common, or by WGS (purple pathway). After statistical corrections, the variants contributing to the phenotype are those that show significant difference between cases and controls. Adapted from EMBL-EBI training: GWAS. Created with BioRender.com.

GWA studies have linked >200,000 variants to complex traits [12] available from the GWAS Catalog [52], yet the majority are still uncharacterised [24]. During early GWA studies it became apparent that variants identified at genome-wide significance explain only a small proportion of trait heritability, and this became known as the missing heritability problem [53]. The cumulative effect sizes of associations were smaller than those estimated for the overall heritability for the trait [24]. It was not known whether this was caused by common variants tagging causal variants imperfectly, if the causal variants were rare or if heritability has been overestimated from pedigree data. However, now we know that common diseases tend to share genetic predisposition with multiple common genetic variants, each with modest effect sizes (eg. [54, 55]), and it has been shown that rare variants contribute to the heritability, too [8]. To complicate matters further, many variants are associated with multiple traits and have different effects in different cell types [56]. It has been shown that the heritability of complex traits is largely distributed along the genome [57, 58] suggesting that many genes play a role in disease risk variation. Nevertheless, increasing sample sizes will increase the number of loci identified and build a better picture of the common genetic architecture of common diseases [59].

To date, GWA studies have mostly been performed in European populations [60]. Since allele frequencies can vary quite substantially in different ethnic backgrounds, this can cause a major problem when trying to biologically interpret the data [61-63]. Increasing sample sizes across a more diverse range of populations, and using better variant reference panels, will help to better resolve the causal variants from the haplotype block and aid in functional understanding [24].

With the lack of understanding of the biological mechanisms of causal variants [64], studies are now focusing on improving variant annotation. The Encyclopedia of DNA Elements (ENCODE) project [65] has identified various functional elements in the genome, thus allowing a better understanding of the potential mechanistic actions of unknown variants. Another large collaboration, the GTEx consortium, has identified associations between genetic variants and gene expression across hundreds of individuals across diverse tissues [66] in an effort to correlate trait-associated variants with the genes that they regulate in a tissue specific manner [67, 68]. Regardless, the consequence of a large proportion of

disease-associated variants remains unknown, especially of those in noncoding regions, which are difficult to link to the causal gene for a particular biological function [51].

Understanding which variants cause phenotypic change and characterising how they function will aid in genetic diagnosis and prognosis. This will be important for personalised medicine by revealing the causal gene(s), relevant cell types and biological pathways. One way to better understand causal variants and the mechanisms through which they act is by assessing the direct phenotype they exert, such as gene expression [69, 70], as it is known to affect many diseases and traits [71, 72].

1.3 GENE EXPRESSION STUDIES

1.3.1 RNA SEQUENCING

The human genome contains all the instructions of every cell in the body for the whole lifetime. It also has errors or mutations that will determine disease susceptibility and influence individual traits. Different cell types at different times will express different genes regulated by different regulatory regions. There are large changes in gene expression and transcript usage in several monogenic disorders [73, 74] and RNA-seq has been shown to be useful for diagnostic purposes [37, 38, 75]. Sequencing technologies have advanced at a colossal rate in accuracy, speed, and cost. This has revolutionised functional genomic studies, providing an enormous amount of data for further studies. We now have RNA and other functional sequencing data from different populations, disease cohorts and cell types.

RNA-sequencing is the most commonly used method to quantify RNA molecules in a sample. The first step of the process involves RNA extraction, after which the mRNA (or small RNAs) is enriched, or depleted of ribosomal RNA [76]. A cDNA library is then generated from the RNA molecules through reverse transcription before adaptor sequences are ligated. The cDNA library can then be PCR amplified if required. Following this, the cDNA library is sequenced at a required depth with a high-throughput sequencing platform [77]. Illumina short-read sequencing is highly accurate, however not perfect with an error rate of ~0.1-0.5% [29]. Therefore, some mistakes will occur and for this reason, higher coverage is

often preferred; most studies employ >=20× coverage to compromise on the cost and accuracy.

Short-read RNA sequencing computational workflows start by performing quality control on raw reads to detect sequencing errors, PCR artefacts or contaminations. The nucleotide quality tends to worsen towards the ends of reads, and often bases at the end are removed to improve the alignment [78]. At this step the adapters are also trimmed, and poor-quality reads removed. RNA-sequencing reads are then aligned to the reference genome to quantify the number of reads overlapping genes and transcripts, filtered and normalised for differences in library size, the lengths of the genes and technical artefacts [76]. It is important to remember that RNA sequencing provides relative, rather than actual, measurements of the expression of a gene compared to all other transcripts in the sequencing library. Different tools, parameters, or reference genome versions affect the results, and each has their limitations and biases; therefore, depending on the area of research different workflows might be employed.

There are numerous alignment tools that map sequencing reads to the reference genome such as BWA [79] and Bowtie2 [80], and across these tools, one of two broad methods are typically employed: hash table indexing or a Burrows-Wheeler transform. In general, the first step of aligners is to fragment the reference genome, where the aligner can find all exact matches for the read within a single lookup rather than scanning the whole genome for each read [81]. However, for RNA-sequencing data, there is an additional feature that needs to be considered beyond potential mismatches and indels, and that is that mRNA does not contain introns and therefore reads may be ‘split’ across adjacent genomic regions. This property of the data creates difficulty for aligners in determining the non-continuous genomic location for each sequencing read. To solve this problem, there are many different computational solutions to aligning RNA-seq reads. STAR [82], for example, allows for splice junction detection by finding the Maximal Mappable Prefix between the genome and the reads. Here, the start of the read is aligned to the reference genome, finding the maximum mappable length. During this process, if the read contains a splice junction, a part of the read will not be aligned to this initial location. Therefore, the unaligned part of the read is then aligned to the donor splice site. This way different parts of the single read can align to multiple genomic locations accounting for the splicing. Following

this, STAR joins these two parts together and quantifies all mismatches and indels, and then selects the best alignment for each read based on mismatch and indel scoring penalties. There are also other approaches including pseudoaligners, such as kallisto [83], which are based on a k-mer algorithm using a de Bruijn Graph of the reference transcriptome. Pseudoaligners speed up RNA-seq quantification by identifying a list of compatible transcripts for each read from which it could have originated, rather than considering alignment of each base pair from read to the transcripts.

There is now increasing interest in long-read sequencing as it eliminates a lot of the complications associated with aligning short reads to the reference genome. Long-read sequencing would allow *de novo* haplotype resolved genome assembly, but while the field is still new the error rate is ~10-20% [84], is more expensive and workflows are still in development. For this reason, most of the studies still use short-read sequencing [29].

1.3.2 EXPRESSION QUANTITATIVE TRAIT LOCI (EQTLS)

Much of gene expression variation is due to genetic effects [85], and we know variation in gene expression is important in disease as GWAS hits are enriched for expression quantitative trait loci (eQTLs) [67, 72, 86]. eQTL-mapping is the most common method to study gene expression variation in a population of individuals. They search for the statistical associations between a variant and the expression level of a gene [65]. In this analysis, individuals are grouped based on their genotype, then gene expression levels for each gene are compared between groups via linear models [85]. If gene expression levels are significantly higher in one group then it is assumed that a variant within the locus under study is affecting expression [87]. An illustration on how the analysis is performed is shown in Figure 3.

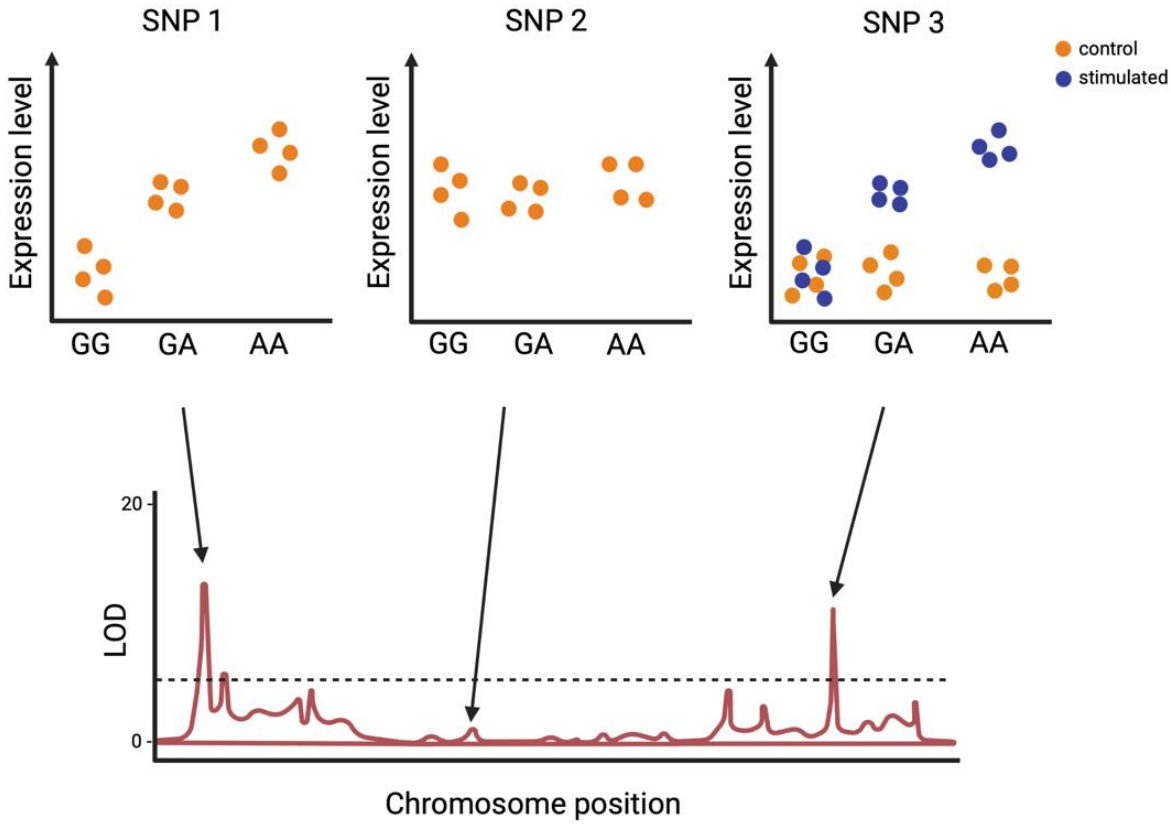


Figure 3. **The principle of eQTL detection.**
In the above example, three distinct variants are illustrated in three groups of individuals (two homozygous to the two alleles of each variant and one heterozygous). For single nucleotide polymorphism (SNP) 3, the orange shapes represent control condition when no eQTL is observed, and blue shapes represent conditional stimulation that reveals an eQTL. In the bottom panel, the logarithm of odds (LOD) is used to investigate the statistical association between genotype and gene expression. The variants above the genome wide threshold (SNP 1 and SNP 3 during stimulation) are considered an eQTL. Created with BioRender.com.

Most studies focus on genetic variants that are close to each gene and operate in *cis*. Cis-variants are located on the same allele as the gene they affect. eQTLs may act through transcription factor binding [66, 67], chromatin accessibility [68], histone modifications [69], alternative splicing, small RNAs, large intergenic non-coding RNAs, RNA editing, and mRNA degradation [65], usually from a distance of few kilobases. *Trans*-variants typically act via a diffusible factor [88], such as transcription factors, which will affect both alleles but are more complicated to study [89]. An illustration describing these processes is presented in Figure 4.

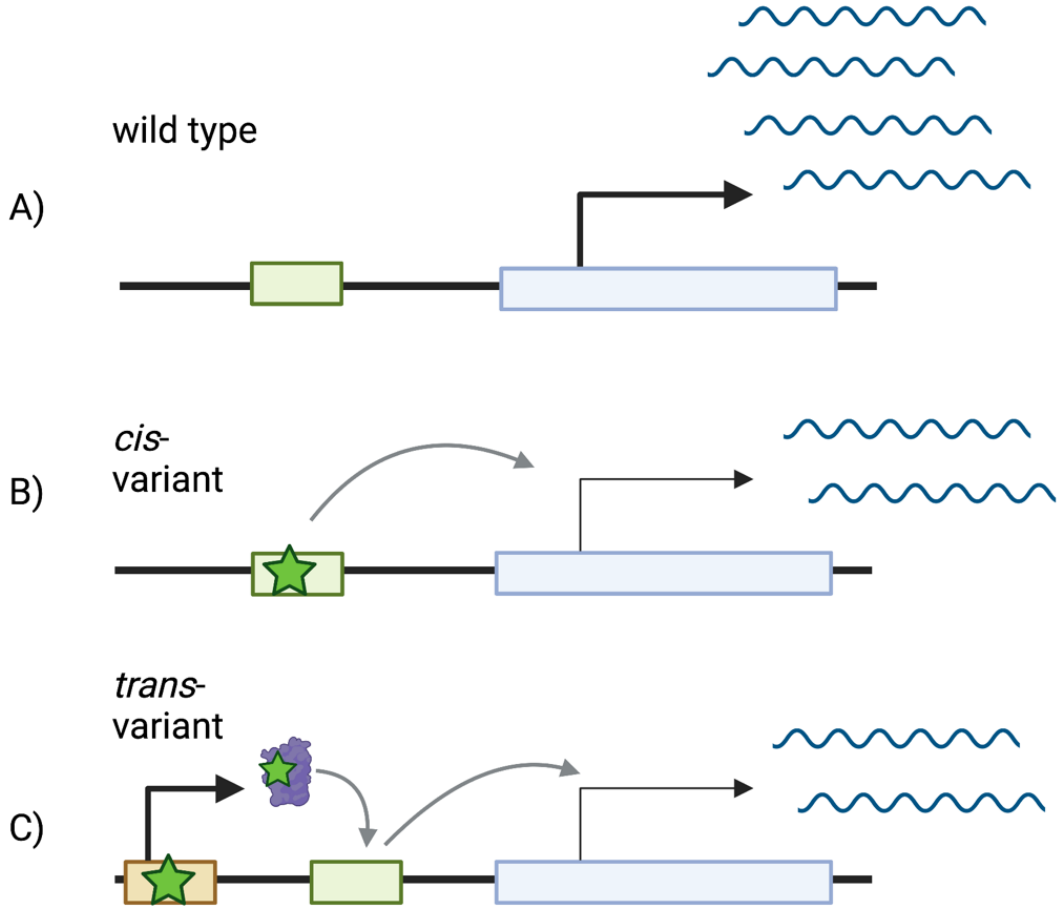


Figure 4. **Mechanism of eQTL**.
In panel A, a regulatory region (green box) regulates expression of a gene (blue box). In the panel B, there is a *cis*-variant that reduces the gene expression relative to the wild type (panel A). In the panel C, there is a *trans*-variant further away or on a different chromosome that mutates a regulatory protein that then reduces expression of a gene relative to the wild type. Created with BioRender.com

A large consortium (GTEx) has pursued the challenge of characterising the genetic architecture of gene expression across different tissue types, and now almost every gene has at least one known eQTL [66, 90]. The eQTL map changes dynamically depending on the tissue, cell type [91, 92], cellular environment, internal conditions of the source [93], and the donor genetic background [94]. To date, most eQTL studies have been performed on easily accessible tissues, such as blood, or in post mortem tissue samples [95]. However, tissue samples consist of multiple different cell types which can lead to poor resolution. Because gene expression is highly dependent on environment, the signal can be affected by

the cellular heterogeneity and cellular state. Additionally, there is differential expression across populations [64], therefore similar to GWAS these studies will benefit from more diverse samples.

eQTLs can also only capture the effects of common variants, and similar to GWA studies, due to linkage disequilibrium the causal variant is often unknown [96]. There are statistical fine-mapping approaches to try to elucidate the causal variant within the locus [97-99], but these can also often be limited. It is also often the case that there is an overlap between GWAS hits with eQTL signals [71, 72], therefore combining these analyses may help identify the connection between the causal variant and/or target gene and the mechanism of action [100]. However, this task is far from complete.

One consortium set to tackle this gap is now one of the most used genomics resources, Genotype-Tissue Expression (GTEx) project. The project has given the recourses to study how genetic variants relate to the gene expression changes, described in the section below.

1.3.3 GENOTYPE-TISSUE EXPRESSION (GTEX) PROJECT

Most genes are differentially expressed in different cell types [101] and at different developmental stages [102], with some transcripts being only transient, and as such large number of tissues and individuals are needed to capture the effects of genetic variants on gene expression. In addition, it would be beneficial to study the effects of genetic variants in the disease-relevant tissue for example, however, this is often not possible in a living donor [103]. These problems motivated the GTEx project, where gene and transcript expression has been profiled from multiple tissues, along with genotypes, from deceased individuals [86]. The current V8 release contains nearly 1000 individuals and 54 different tissue types. This enables the study of the relationship of genetic variants with gene expression. The GTEx project has also provided the scientific community with a tissue bank that allows researchers to study the relationship between genomic variants and molecular phenotypes to answer specific research questions.

The GTEx resource has been used in numerous studies and has led to many discoveries. The main aim of the consortium was to provide a resource to study eQTLs. For example, it has been crucial in elucidating that most eQTLs are shared, and while tissue-specific eQTLs are still common, eQTLs that are shared between only a few tissues are not seen as often [86]. It has also revealed that multiple *cis*-eQTLs act on the same genes [104]. This is contrary to *trans*-eQTL, where the effect is most often tissue-specific [66]. The GTEx project has also been used as a resource for other fields, including a range of studies that have used GTEx data to identify tissue-specific imprinting [105], the effect of protein-truncating variants on the gene expression in different tissues [106], and the impact of structural variation on causing eQTLs [107]. GTEx also provides a genomics data resource to study allele-specific expression (ASE) [108]. Taken together, this project has provided a standardized framework by which all other gene expression studies can be conducted, providing a standardised analysis pipeline, and reference data that is invaluable across both basic and disease biology.

Because eQTL analysis requires large sample sizes, it is not applicable to the study of rare diseases with small sample sizes. ASE offers a way to study the effects of variants on gene expression within individual samples. ASE can also be used to support eQTL analysis because eQTLs that are near genes often act through ASE [85].

1.3.4 ALLELE SPECIFIC EXPRESSION (ASE)

Allele specific expression (ASE) represents allelic imbalance between two alleles within an individual [109]. ASE analysis detects the differential expression of two alleles from the same individual at a site containing a heterozygous variant (Figure 5). ASE can be caused by a *cis*-regulatory variant affecting gene expression which can affect transcription factor binding affinity [110, 111], by epigenetic effects including methylation [112] and imprinting, or by nonsense mediated decay [113]. However, it has been shown that it is mostly due to genetic rather than epigenetic effects [64]. ASE also captures allele specific splicing events [114, 115] such as a variant causing exon skipping or intron retention; and also monoallelic expression including those observed during X chromosome inactivation [116, 117], in the olfactory receptors [118], and in immunoglobulin receptors [119, 120]. However, ASE is not

able to detect *trans* effects as both alleles share the same environment and both will be affected by *trans* effects. There are also allele specific effects downstream of transcription at the posttranscriptional stage such as ribosome protein binding affinity or mRNA regulation [121].

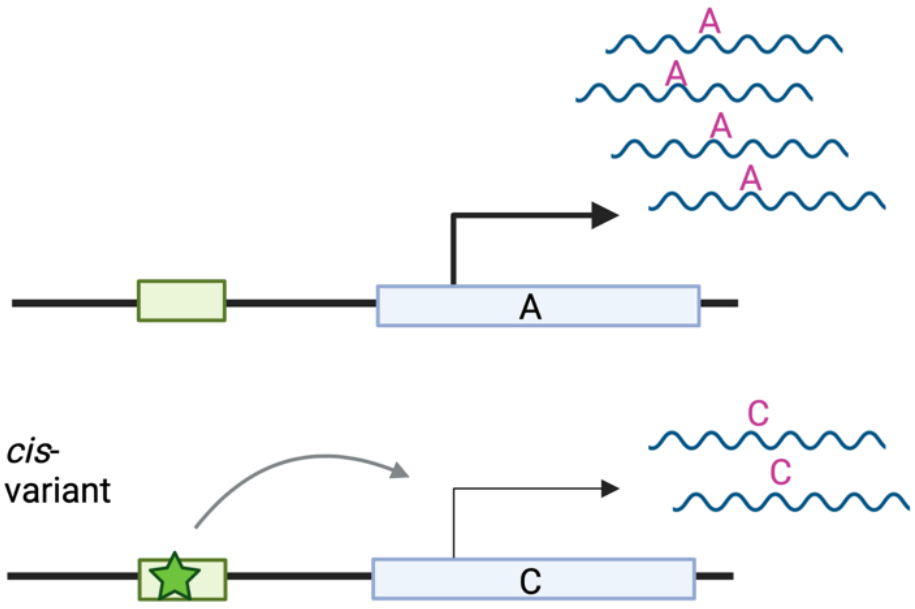


Figure 5. **Mechanism of ASE expression**.
A gene (blue box) contains a heterozygous variant, which acts as a marker to distinguish the expression from either allele. C allele has a mutation within regulatory region (green box) that reduces transcription factor binding and therefore reduces expression. As a consequence, C allele halves its expression relative to the wild type A allele that can be quantified with RNA-seq considering reads spanning the heterozygous variant within the gene. Created with BioRender.com.

RNA-seq reads provide coverage across heterozygous sites in expressed genes, and these represent transcription from maternal and paternal alleles [122, 123]. According to a null hypothesis, the ratio of reads originating from the reference allele and alternative allele should be equal. If there is a variant influencing the gene expression in one haplotype, it can cause imbalanced expression between two alleles. This is usually detected through a binomial test of read counts between two alleles [124]. If the ratio deviates significantly from 50:50, the gene is under ASE [94]. These statistical tests rely on the coverage at heterozygous sites to distinguish genuine ASE events from a mutation within a read [125] and to distinguish genuine biological events from technical variation in allele mapping. ASE detection is illustrated in Figure 6.

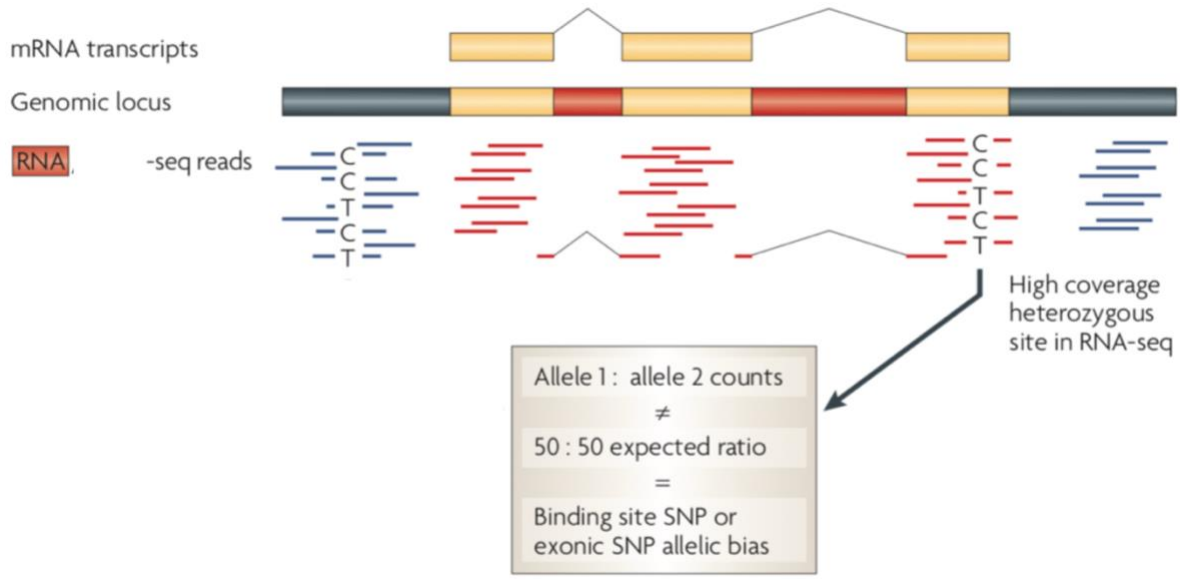


Figure 6. **The principle of ASE detection.**
The ratio of RNA-seq reads overlapping a heterozygous variant is calculated. A statistical test are performed to detect genes where the ratio significantly deviates from 1:1. Adapted from Pastinen, 2010 [126]. Reproduced with permission from Springer Nature, Copyright 2010.

ASE analysis is a powerful method that allows quantification of *cis*-variant effects within an individual rather than needing large population samples. eQTL analysis requires multiple individuals from each genotype group for each genetic variant, which can be difficult to obtain for rare variants. Comparison of eQTL and ASE is shown in Figure 7.

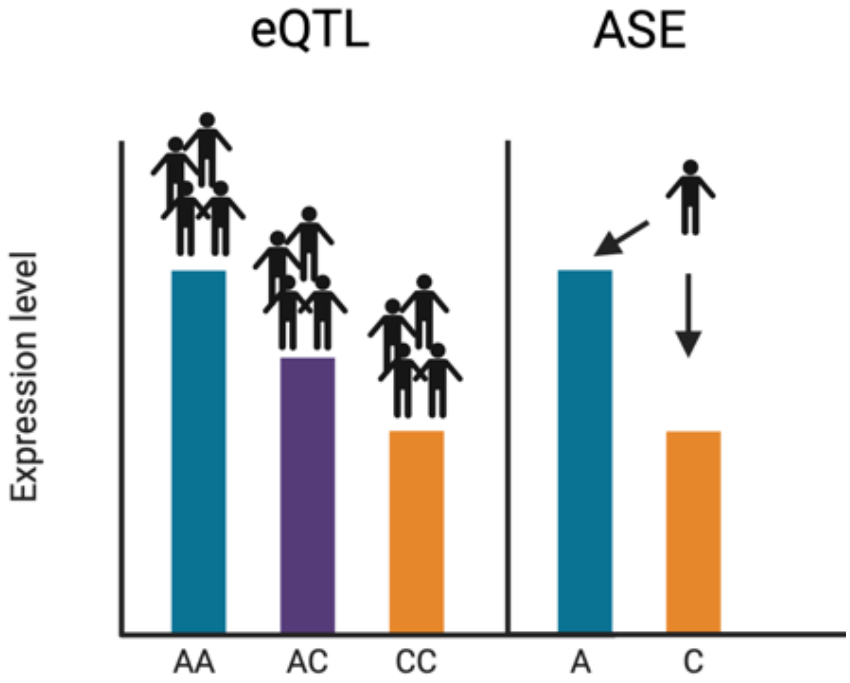


Figure 7. **Comparison of eQTL and ASE detection.**
While eQTL analysis (left side of the figure) requires large sample sizes with each genotype, variant consequences on gene expression can be measured within a single individual in ASE analysis. Created with BioRender.com.

The regulation of most genes (roughly 60%) is affected by genetic variants in *cis* [64]. ASE captures the effects of both common and rare variants [126]. However, it can also be used to detect expression modifiers of protein-coding mutations [113, 127], tumour tissue [128] and loss of function variants [129]. Because ASE is often caused by a variant or variants in a regulatory region (as opposed to imprinting, which is less common), different environmental conditions and different cellular stages may utilise different regulatory regions and therefore alter ASE effects [93]. This allows the study of the interactions between environment and genetics. For example, ASE tends to slightly increase (by 2.69% when measured at 70 and 80 years of age from the same individuals) during ageing, likely due to increased environmental variance and reduced genetic regulation [130].

Despite its promise, the use of ASE as a detection method also has its own limitations. For example, in the cancer field, ASE can be driven by technical artefacts, including copy number variants that are common in cancer and can lead to false positive ASE events, which account for 84% of allelic imbalance [131]. This is in addition to a highly mutated genome that will suffer from mapping biases. ASE can be informative in a disease context, though. There have been studies showing ASE can contribute to disease phenotypes, including cases where deleterious alleles at heterozygous sites have been shown to have a higher expression level in disease cases relative to the healthy controls [132-134]. This allows disease alleles to have a greater impact than would be expected from a recessive trait. Equally, ASE can mediate autosomal-dominant disease phenotypes by increased expression of the wild-type allele [135]. The authors hypothesised a feedback loop can be responsible for this. Indeed, a *trans* effect have been shown to mitigate *cis*-regulatory effects by a negative feedback loop in model organisms to compensate for adverse allelic imbalance [136-138].

Most ASE studies have used bulk RNA, or RNA from tissues that contains different cellular types to detect gene regulatory effects. This different cellular composition will cause variation in ASE due to different regulation in constituent cells. To address this, there are new studies focusing on single cell ASE [139]. However, this method is expensive, and the analysis is still evolving. In addition, bulk RNA-seq data availability significantly outweighs

that of single-cell data, therefore it remains a commonly used data type. RNA-seq is a highly useful resource, and the next section will describe its utility.

1.3.5 COMPUTATIONAL CHALLENGES

Analysis of next-generation sequencing data has numerous challenges due to the complexity and lack of understanding of the human genome. One of the biggest limitations in short-read sequencing is the reliance on the reference genome and the biases that follow this. The reference genome has errors and gaps, but most importantly, it lacks the diversity present in the human population. Genomic regions with high diversity are difficult to study as the sequencing read will have multiple mismatches from the reference genome. Yet some clinically relevant genes are known to be highly diverse, such as the HLA locus [140]. For the same reason, reads that are more similar to the reference sequence will be more accurately and easily aligned [141, 142]. The reads that contain alternative alleles are more likely to be discarded as they contain a mismatch from the reference genome, biasing the analysis towards the reference allele. This is known as mapping bias [141].

Another computational challenge comes from incorrect phasing. During short-read sequencing, maternal and paternal chromosomes are sequenced simultaneously. It is often left for computational tools to phase the haplotypes. Haplotype phasing means linking together genetic variants that occur on the same chromosome [143] and this information is often important when aligning sequencing data since genetic variation can influence the ability to correctly align a read. In addition, often a regulatory variant influences the allelic imbalance [144]. Potentially other *cis* variants might mitigate or exacerbate this effect. Therefore, linking the combination of variants on the chromosome is crucial for ASE analysis.

There have been large studies aiming to better understand the variation in the human genome, including the 1000 Genomes Project [44] and the HapMap 3 project [145]. Despite these efforts, variant calling is still not complete. Often the phasing information is not considered due to the cost of experiments and the computational challenges [144].

However, as eQTL studies have shown, knowledge of the background of variants is of crucial importance [146].

The most common phasing method is population-based haplotype inference. Here computational tools are used to output the most likely phase based on the modelling of haplotype frequencies [147]. This will not resolve rare and private variants, in addition to other challenges including the extent of linkage-disequilibrium. Pedigree-based analysis was widely used before large-scale sequencing data was available. However, it is often not plausible in large scale studies due to the sequencing costs of parents, availability of parental genomes and, also *de novo* mutations would not be resolved with this approach. Phasing and the difference between pedigree and haplotype method is shown in Figure 8. Other methods include haplotype assembly [144] and experimental methods such as gently fragmenting genomic DNA into large chunks and sequencing each separately; or separating metaphase chromosomes and sequencing each separately [143]. Haplotype assembly is not capable of phasing full genomes [144], and experimental methods can be very expensive. Combination methods perform the best [148], however, due to cost and data access limitations these are often not possible. Therefore, computational approaches for phasing that incorporate rare and private variants would overcome the limitations of the above.

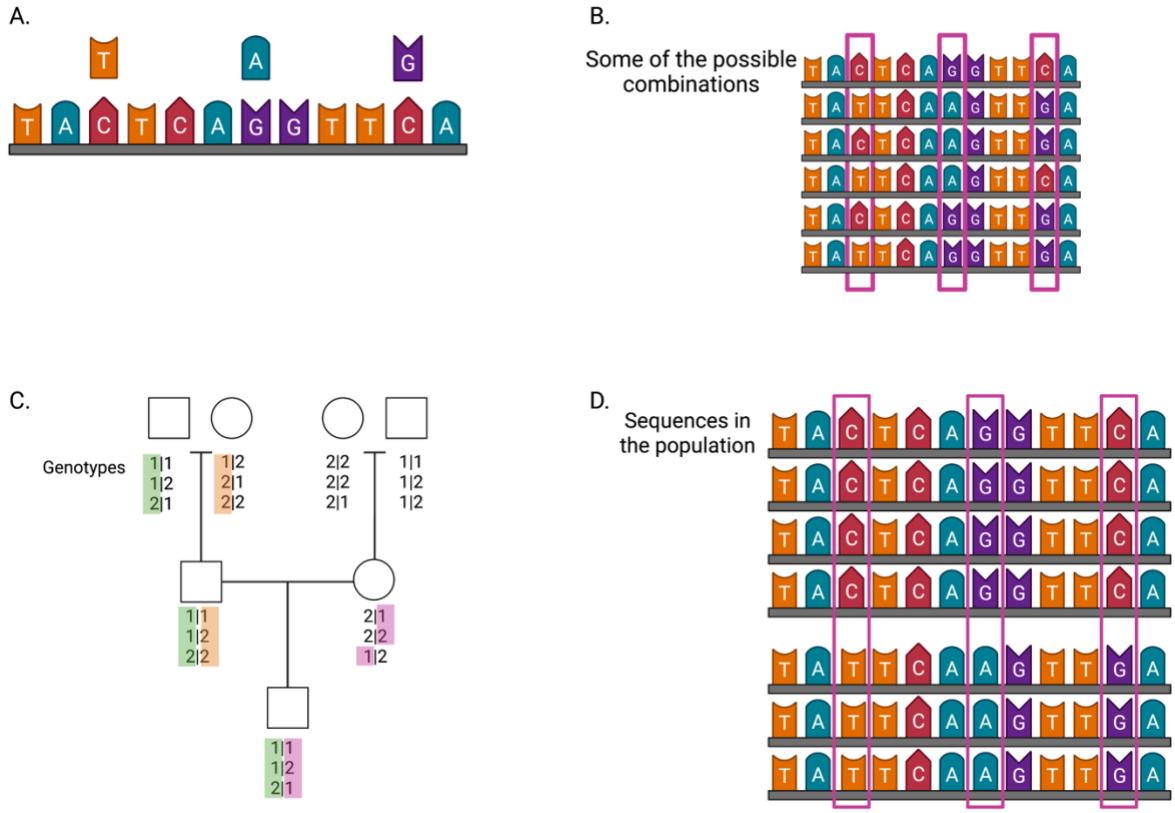


Figure 8. **The difficulty of phasing and resolution methods.**
In panel A, an allele has 3 polymorphisms. When reads are generated for this locus from a sequencing machine, the haplotype is undetermined. Panel B shows some of the possible allelic combinations by which the polymorphisms could be arranged within the locus. Panel C illustrates phasing inference with pedigrees. The haplotype inheritance is visualised by the coloured blocks. The pink haplotype goes through a recombination event. Panel D illustrates the use of population genomic information to infer the most likely phase. Created with BioRender.com.

Mapping biases and the resolution of genetic phase are sources of artificial bias that are particularly problematic for ASE. Therefore, improving these can lead to better detection of ASE at an individual level. The accurate detection of ASE can lead to a better understanding of how variants, and in particular rare variants, play a role in regulating gene expression. Because there is a vast resource of short-read sequencing, it is important to develop correction methods to be able to use the data available and expand the range of important biological findings.

1.3.6 EXISTING ASE TOOLS AND ASE USES

There are numerous tools developed to study ASE. Some of the existing methods include GATK ASEReadCounter [27, 149], WASP [123], aFC [150], AlleleSeq [151], EMASE [152], QuASAR [153], ASElux [154], ASEP [155], GeneiASE [156], MBASED [157], MMSEQ [158], and others [159-163]. There are a range of different functions across these tools that stretch from simple allele counting at heterozygous sites, to complex adjustment and normalisation of allele count data, each coming with their own advantages and limitations. In general, the existing software do not resolve all technical challenges associated with calling ASE from short read data.

Because GATK variant calling is the gold standard variant caller and genotype information is often needed for ASE analysis, the GATK tool ASReadCounter is often incorporated into ASE pipelines and is easy to use. The tool simply quantifies the allelic ratios at heterozygous sites, with the user being able to filter for qualities such as coverage, base quality, and mapping quality. However, it does not take into account any computational biases or attempt to correct for these.

Another widely used tool is WASP, which reduces biases associated with ASE analysis by removing ambiguously aligning reads. It removes reads that would not align to the same genomic location if the genotype at each heterozygous site is swapped to the opposite allele. This has been shown to reduce computational artefacts, however, it also removes a large number of reads that then impact downstream power to detect ASE events. WASP also performs other corrections, including removing duplicate reads at random rather than removing reads with the lowest mapping score (which often is the read with the alternative allele) and controlling for GC content.

Log allelic fold change (aFC), although not an ASE detection method, is commonly used in the ASE field as a downstream analysis tool that uses ASE to quantify aFC. aFC is a model that describes the magnitude of change in gene expression levels that are associated with a genetic variant(s) and can also be used to quantify allelic imbalance. aFC can be quantified from ASE and eQTL data, allowing for a direct comparison.

Another approach used to improve the quantification of ASE is to generate personalised diploid genomes for the alignment of sequencing data, based on the two inherited haplotypes of an individual, as is the case in AlleleSeq. With this method, personalised genomes are constructed that contain genetic variants from each of the two genetic backgrounds inherited from parents. RNA-seq reads are then aligned to both genomes, and because there are no mismatches associated with the alternative allele at each heterozygous site, the reference bias should not be present. Following alignment, AlleleSeq can quantify ASE events, but it is often limited by its inability to correctly deal with indels.

Other tools seek to deal with another common problem that can bias ASE data, which is reads that align to multiple locations. It is often tricky to know how to deal with these reads, as simply removing them could bias the output, and this problem is further compounded as reads containing mismatches from the reference genome are more likely to align to multiple locations. EMASE is a method that aligns multi-mapping reads by a hierarchical approach, resolving ambiguities initially at the gene level, then isoform and lastly at allele levels. Following this, the software quantifies ASE at each site. This approach retains more reads that can be informative and reduces reference bias since multi-mapping reads tend not to originate randomly from both alleles.

Although the above tools seek to solve specific technical problems with RNA-seq data, they largely focus on individual heterozygous sites for analysis or correction. ASEP is a method that aggregates single nucleotide polymorphisms (SNPs) within a gene across multiple individuals, utilising a model to quantify gene-based ASE. GeneiASE is another method that aggregates information from multiple individuals. It is based on Fisher's meta-analysis method combining P-values across individuals. It can also detect ASE events induced by a particular condition. Similarly, MBASED is a method that aggregates multiple SNV loci using meta-analysis based detection to quantify gene-level ASE within individuals.

Finally, most ASE analysis approaches require RNA-seq and genotype information. QuASAR is a statistical learning method that has been developed to detect heterozygous SNPs and quantify ASE events within an individual, removing the need to genotype the variants prior to the ASE analysis.

Outside of these tools, there are many other approaches being developed that are now being tested more thoroughly. ASElux is a tool that uses genotypes to generate an allelic reference genome. It builds all reads spanning a heterozygous site and pre-screens the RNA-seq data. The runtime is lower due to only considering genomic regions containing the heterozygous variants. MMSEQ tried to battle reference allele bias by generating a custom transcriptome onto which the reads are aligned and uses a statistical model to quantify ASE. IDP-ASE quantifies ASE by combining long and short-read sequencing [162] and Skelly *et al*. developed a Bayesian hierarchical model to quantify ASE by comparing allelic ratios in ASE to non-ASE genes [159].

ASE analysis has been applied to multiple fields and continues to gain interest in the research community seeking to understand the biological implication of altered gene expression. ASE is a powerful tool to study gene expression changes associated with a local regulatory variant, particularly with small sample sizes. Because GWAS hits are enriched for eQTLs, but the exact causal variant and linked causal gene are often unknown, ASE can be used as an extra layer of information in these cases. Indeed, ASE has been used to complement fine-mapping approaches in narrowing down a list of potential causal variants [164], and this method has been shown to reduce the potential number of causal variants by 11% [165]. In addition, numerous studies have identified genes under ASE quantitatively as a resource for the scientific community (eg. [108]).

Because ASE can be caused not only by a local regulatory variant but also by imprinting, studies have also utilised ASE for this field. One study identified novel imprinted genes utilising ASE analysis, that were verified by additional methods [166]. Another study used ASE analysis to identify a differential preference for parental expression in developing mouse brain compared to the adult brain [167]. This demonstrates the utility of ASE to investigate epigenetic effects in addition to regulatory genetic effects.

ASE can often be a preferred method to study gene expression changes associated with genetic variants in cases where only small sample sizes can be for the analysis (such as rare disease studies) or where small sample sizes are preferable in terms of cost and resources. For example, ASE can be used to study gene expression in a larger number of environmental conditions and tissues than would normally be feasible, particularly in studies investigating

GxE interactions where multiple conditions are required [93, 102]. Indeed, recent studies have shown that around 50% of genes that demonstrate ASE under particular environmental conditions are missed in large eQTL studies [102]. Furthermore, it has been demonstrated that roughly 50% of these conditional ASE genes are implicated with GWAS hits and thus complex traits, with some examples including obesity and Parkinson's disease [93].

ASE has also been widely used in the disease and diagnostic context. For example, one study identified ASE in loci associated with Parkinson's disease, implicating genes likely to be involved in the disease, and therefore potential targets for therapeutic intervention [168]. ASE has also been observed in autism spectrum disorder and schizophrenia samples, again potentially pointing to causal biology [132, 169]. Similarly, changes in ASE have also been identified in cancer samples (versus normal controls), with generalized approaches identifying targets overlapping known candidate genes [170] and targeted approaches implicating changes in the expression of known causal genes in colorectal cancer [171]. As ASE methods are still being developed and refined, their utility in clinical settings will continue to expand.

1.4 THESIS AIMS

The overall aim of my PhD is to develop a method to better quantify ASE using short-read sequencing data. In order to achieve this, I will focus on the following specific aims:

1. Develop, test, and characterise a pipeline to improve computational biases associated with ASE detection.
2. Release the pipeline into an easy-to-use, reproducible format and validate its performance against other methods.
3. Assess the performance of the pipeline in recapitulating population level signal against other methods.
4. Investigate the utility of ASE and the pipeline in detecting biologically important and disease relevant signals.

CHAPTER 2 – IMPROVING ASE DETECTION AND GENERATING A PERSONALISED ASE CALLER (PAC)

2.1 INTRODUCTION

2.1.1 ACCURATE GENE EXPRESSION QUANTIFICATION IS IMPORTANT

The impact of genetic variation can have different functional consequences depending on where in the human genome it occurs. Variants that fall within coding regions are more likely to affect the function of the protein itself, but this can also depend on the type of mutation that has occurred, be it a single base pair substitution, i.e. changing a single nucleotide within the DNA; or structural variation that changes larger DNA sequences [172]. Structural variation can be due to copy number variation, or chromosomal rearrangement events including insertions, deletions, and duplications, and are more likely to disrupt protein function. If the change is divisible by three (length of a codon) and in frame, the protein will gain or lose amino acids but the up- and downstream of the protein will not be affected. Otherwise, structural variation will lead to a frameshift affecting the downstream amino acids of the protein. Single nucleotide variation on the other hand can be synonymous, where due to redundancy in the genetic code, the variant does not change the amino acid and therefore the protein will not be affected [173], although there are exceptions where synonymous mutations increase disease risk [174]. However, a single nucleotide variant can also be a nonsense variant that changes the amino acid to a premature stop codon, truncating the protein which can lead to the gene being targeted by nonsense-mediated decay, or a missense variant that changes the amino acid within the protein. In this latter case, the consequences of the variant will depend on various factors including where the change is, how conserved the protein is, and how similar the amino acids are.

Some mutations, on the other hand, are in the non-coding genome within so-called junk DNA regions, and some are within regulatory regions influencing gene expression levels. Because we do not understand the consequences of all these mutations, especially those outside of the coding regions, it is often hard to prioritise them for further study. GWA and eQTL studies have been important in deciphering important variants in the noncoding regions, and it has become apparent that noncoding variants have a high impact on complex traits [175-177], many acting via gene expression [178]. As an example, lactose intolerance

is prevalent in certain populations, and a variant upstream to *LCT*, a lactase gene, was identified for this phenotype [179]. The variant is within a distal enhancer and is associated with lactase production in particular cell types [180], which has provided a fitness benefit in recent human evolutionary history. Therefore, understanding gene expression regulation can be informative for understanding underlying biological mechanisms and population differences.

However, it can be challenging to distinguish the causal variants with eQTL studies due to linkage disequilibrium, and because genes are regulated by multiple regulatory regions. Another drawback of eQTL studies is that they cannot be performed on smaller sample sizes due to a lack of power [181]. However, ASE is a powerful tool to study allelic imbalance that overcomes these issues and has been shown to be informative in disease contexts (eg. [168, 171]). ASE analysis is a potentially more powerful approach to detect the effects of *cis*-acting variants rather than eQTL studies [182] due to being measured within the individuals; ASE analysis in only few individuals can be enough to detect the effects of rare variants [183].

There is an enormous amount of short-read data available that offers opportunities to utilise these data for ASE analysis in many areas. The ASE field suffers from biases including ones related to computational methods that deal with short-read sequencing [142]. In order to utilise the potential of ASE analysis, it is essential to deal with these artefacts. The following sections will describe technical problems around ASE analysis.

2.1.2 BENEFITS OF ACCURATE PHASING FOR THE ASE ANALYSIS

In a typical ASE analysis, RNA-seq reads first need to be aligned to the reference genome. Reads with an alternative allele at a heterozygous site have a mismatch from the reference allele present in the reference genome, and this, in combination with other potential variants within the read, sequencing errors or low-quality nucleotides, causes reads with alternative alleles to be discarded more often. This is called reference bias [141, 184]. One way to overcome this is to construct parental genomes, where phased variants are

incorporated into the reference genome generating two parental genomes [151], onto which RNA-seq reads can be aligned.

Humans inherit one maternal and one paternal copy of a genome, each containing genetic variants. Most genetic studies do not directly differentiate between these two copies, due to the cost and complexities of sequencing both parental samples. Instead, phasing or haplotype estimation is used and is important in regions surrounding heterozygous variants, as they are key for understanding how genetic variant(s) influence gene expression in *cis*. Phasing determines whether variants come from the same or different chromosomes (Figure 9) and when constructing personalised genomes, this governs whether variants come from maternal or paternal genomes. This is a challenging task and traditionally has been performed using population-level data [148, 185, 186], however, this method struggles with rare and *de novo* variants, and in regions of high diversity [187]. Long-read sequencing can incorporate private haplotypes [188], but this method is a relatively new approach, with the academic community gaining interest in last two decades [189], but it is still a costly method where the underlying protocols and analysis methods are still being developed. Similarly, alternative methods including modified laboratory protocols [147, 148] or sequencing families [190], are also costly, especially for larger sample sizes. Conversely, there are also computational tools and approaches that seek to deal with the correct phasing of genomes, phASER [191] being one of them. This approach improves phasing by incorporating RNA-seq reads that will bring variants together over longer distances, since the same read harbouring multiple genetic variants can often be split across multiple exons as a consequence of splicing.

Accurate phasing of genetic variants impacts the ability to construct correct parental genomes and thus how well reads can be aligned to the personalised genomes. vcf2diploid is a tool within AlleleSeq that constructs personalised maternal and paternal genomes by inserting phased variants, indels and structural variants into the reference genome [151]. These genomes will contain all the individual variants and therefore, when RNA-seq reads are aligned to personalised genomes, the alternative allele can exactly match the altered reference genome.

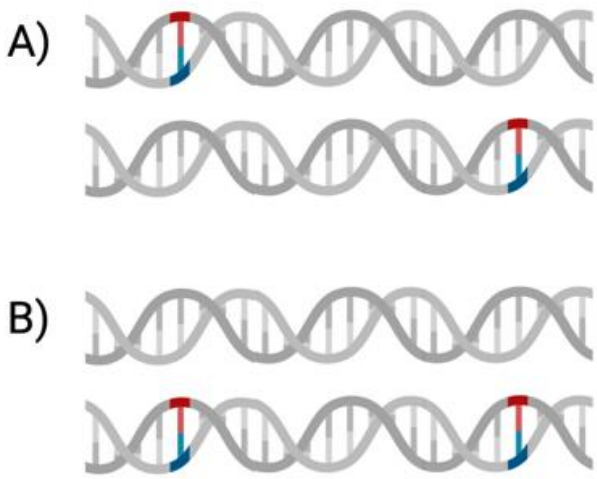


Figure 9. **Compound heterozygosity as an example for importance of phasing**
In this example, when mutations are on different chromosomes (A), the individual is wild-type for a phenotype. When the mutations are on the same chromosome (B), the individual has a disease phenotype. Created with BioRender.com

2.1.3 FILTERING OF RNA-SEQ DATA

The initial step of aligning RNA-seq reads involves multiple filtering steps performed on the raw reads, and each step has downstream consequences [78]. RNA-seq filtering steps are generally determined within research groups and there are no standardised protocols. Each filtering step has its own effects and limitations, and the aim of the research will determine how and which filtering steps to accommodate.

Most commonly reads are evaluated for sequence quality, which depends on library preparation and sequencing. A commonly used Phred quality score is used for Illumina platforms. It quantifies the probability that a given base is incorrect. Trimming removes low-quality nucleotides based on how likely those are assigned incorrectly [192] prior to aligning reads to the reference genome, usually at the end of reads, therefore reducing errors and

random sequences [193]. However, it has been shown that trimming can influence downstream analysis such as gene expression estimation [192] since shorter reads are more likely to be aligned incorrectly, as are the reads that originate from genes with low exon number or high GC content [194, 195]. The number of aligned reads is also reduced [193], which can potentially remove informative reads. This can be particularly unfortunate for studies investigating rare variants and those with small sample sizes.

Another common filtering step is soft-clipping. Soft-clipping is performed during the alignment step where the bases at the beginning and the end of the read that do not match the reference sequence are ignored [196]. The decision is often made considering the quality score and the reference sequence, whereas trimming only takes into account the quality score. In ASE analysis, this may lead to biased allelic quantification if alternative alleles at the ends of reads are preferentially trimmed, therefore exaggerating the reference bias known to occur in the read alignment.

Another common filtering is performed by removing reads that align to multiple locations (from here on referred to as 'multi-mapping' reads). Gene duplications, gene splice variants and repetitive sequences cause some short-reads to map to multiple locations in the genome [197], making it difficult to accurately quantify the expression of certain genes. Around 30% of reads can align to multiple locations [32, 198], and since multi-mapping reads are most often discarded, this removes a large proportion of potentially informative reads and underestimates sequence coverage at certain genes (with large gene families and with high homology) [197]. Methods to recover multi-mapping reads exist, but current ASE detection methods do not incorporate these approaches into the analysis.

There are tools to deal with many of the biases associated with ASE quantification, such as those that reduce mapping bias (eg. [123, 184]) and calculate ASE counts (eg. [199]), though there are no stand-alone pipelines that deal with all of these artificial biases together and produce ASE quantification. For this reason, in this chapter, I developed a pipeline testing how each of the steps described in this section influences ASE detection and to incorporate all key steps into a single stand-alone workflow. To achieve this, I generated highly realistic simulated genomic data to obtain ground truth ASE information. This allowed me to test the pipeline and refine it into a final pipeline.

2.2 METHODS

2.2.1 PRELIMINARY PAC PIPELINE

A preliminary pipeline, later termed ***P***ersonalised ***A***SE ***C***aller (PAC), to obtain improved ASE counts (Figure 10), had been in place prior to me taking over the project. It consists of the following steps.

As input, PAC takes a VCF file with phased variants (variant call format, a file that contains information on the genetic variation within the individual) and a FASTQ files with unaligned RNA-seq reads. As a first step, phASER [191] re-determines the phase of variants at the heterozygous sites where RNA-seq reads can add information (read-aware mode). The resulting VCF is used to incorporate variants into the reference genome to generate hypothetical parental genomes using AlleleSeq [151]. The reads are then aligned to both genomes separately with STAR [82]. The parameters used were as follows: adapters were trimmed (Phred score <30), keeping properly paired (-f 0x0002 using SAMtools) and uniquely mapped (NH:i:1 flag) reads. After alignment to each parental genome, a custom script was used to select the best alignment for each read from the two alignments (scoring reads by the number of matching nucleotides minus two times the number of indel positions, drawing at random when the two mappings have equal scores), and the number of each allele at each heterozygous site was counted. The files were then merged, and the output file contains the coverage and proportion of reads mapping to the reference genome for each heterozygous variant. The pipeline is illustrated in Figure 10.

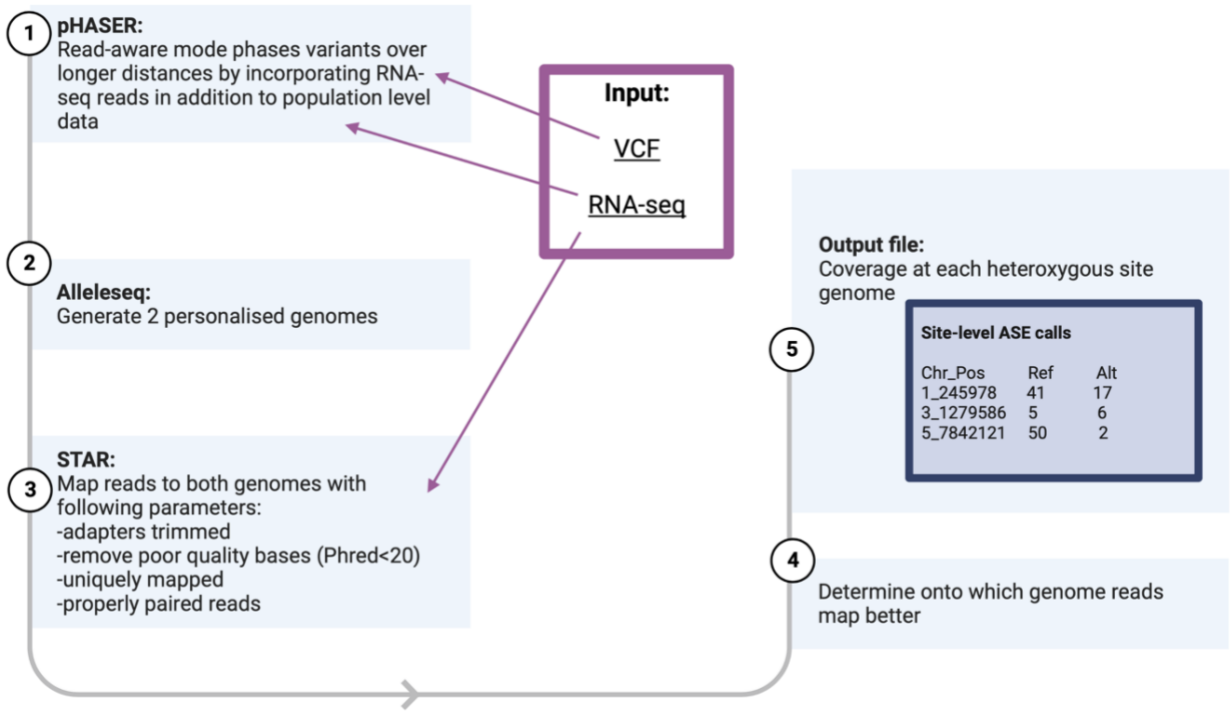


Figure 10. **Preliminary PAC pipeline.**
A schematic describing the main steps, features and outputs of preliminary PAC. Created with BioRender.com

2.2.2 PRELIMINARY ANALYSIS

2.2.2.1 HIPSCI SAMPLES

To test a preliminary version of PAC, I obtained data from Human Induced Pluripotent Stem Cell Initiative (HipSci) [200] that generated induced pluripotent stem cell (iPSC) lines from healthy and disease donors as a reference panel. I selected 10 healthy individuals for which there was genotyping and paired-end RNA-seq data available from the corresponding iPSCs, and iPSC-derived sensory neurons [201] (data acquired from the European Nucleotide Archive (Project code: PRJEB18630)). Genetic variants were obtained from whole exome sequencing (WES) data already called by HipSci. The information on donor cell lines is provided in Table 1.

Donor ID	Gender	Age	Ethnicity	Source	Cell culture conditions
HPSI0913i-eika_2	Male	45-49	White British	Fibroblasts	Feeder-free
HPSI0114i-eipl_1	Female	40-44	White British	Fibroblasts	Feeder-free
HPSI0114i-kolf_3	Male	55-59	White British	Fibroblasts	Feeder-free
HPSI0214i-kucg_2	Male	65-69	White British	Fibroblasts	Feeder-free
HPSI0114i-oevr_3	Male	70-74	White British	Fibroblasts	Feeder-free
HPSI1113i-podx_1	Female	65-69	White British	Fibroblasts	Feeder-free
HPSI0314i-qaqx_1	Female	60-64	White British	Fibroblasts	Feeder-free
HPSI0114i-rozh_5	Female	65-69	White British	Fibroblasts	Feeder-free
HPSI1013i-wuye_2	Female	30-34	White British	Fibroblasts	Feeder-free
HPSI0314i-xugn_1	Male	65-69	White British	Fibroblasts	Feeder-free

Table 1 **HipSci donor information.**

2.2.2.2 HIPSCI BASIC ANALYSIS

To obtain an overview of the performance of PAC, I used RNA-seq data and variant calls from 10 HipSci (Table 1) healthy iPSC and iPSC-derived sensory neuronal cell line samples as inputs for PAC runs, using the GRCh37 reference genome (same version used in HipSci VCF). I also aligned the same RNA-seq with STAR 2.51a with default parameters to GRCh37 reference genomes as a standard alignment approach for comparison. The heterozygous sites generated by PAC and standard alignment located on autosomes with >=10× coverage and present in both approaches and both cell lines were selected for further investigations throughout the analysis.

2.2.2.3 ASE SELECTION AND GENE ANNOTATION

To demonstrate that PAC is able to detect genes under ASE, I determined the genes under ASE from PAC output in iPSCs and neuronal cells. All analysis was performed with custom scripts written in Python (available on https://github.com/anna-saukkonen/PAC/tree/main/thesis_scripts). When comparing cell types or methods, sites present in each were selected at >=10× coverage. A two-tailed binomial test was performed on the heterozygous sites meeting these criteria. The test gives a statistical significance value of deviations from the expected observation, which in this case was 0.5, as the null hypothesis is that both alleles are expressed at the same ratio. Sites with $P<0.05$ were considered ASE sites. For more stringent testing, Bonferroni-adjustment was performed on P-values generated from the binomial test. Bonferroni-adjustment corrects for the issue of multiple comparisons where the chance of false positives increases. The P-values from the binomial test were divided by the number of tests, or the number of heterozygous sites, under study. To link ASE sites to genes, I acquired gene annotations for all sites from the GENCODE GTF file (version 19). If a site overlapped with multiple genes, all genes were included in the analysis. If at least one site was under ASE, the gene was assigned to the ASE gene list. If the gene contained an ASE and non-ASE site, the gene was considered to be under ASE and removed from the non-ASE list.

2.2.2.4 DAVID FUNCTIONAL ENRICHMENT ANALYSIS

To investigate whether genes under ASE identified by PAC were enriched for any biological processes, I annotated each variant to a gene and performed gene enrichment analysis for these genes. Gene enrichment analysis groups genes with related functions together. Gene Ontology (GO) terms are the biological annotation terms associated with a gene, and the enrichment of genes within each term is determined by statistical tests against the control, which is the defined background gene list [202, 203].

The David Bioinformatics Resource 6.8 Functional Annotation Tool [203, 204] was used to obtain GO terms and gene enrichment analysis for genes under ASE (detection Bonferroni-adjusted) in iPSCs for each donor. David uses magnitude of resources to collect gene and protein identifiers and their annotations including NCBI, Uniprot, Ensembl, and Gene Ontology. Ensembl ID was used as inputs for the David Annotation Tool. Genes with at least one site with >=10× coverage from iPSCs were selected for the background gene list. The GO ALL category was chosen, which provided the GO mappings annotated with all the levels of specificity. The GO terms were considered significant with $P<0.05$ after the Benjamini-Hochberg adjustment, which decreased the number of false positives. Benjamini-Hochberg adjustment is a default in David output. Elsewhere in the thesis Bonferroni correction is used for a more stringent correction.

2.2.3 SIMULATED DATA

2.2.3.1 PLATINUM GENOMES PROJECT

In order to determine the accuracy of PAC and to refine its parameters, I generated simulated genomic data where the underlying sequence, variant information and allelic counts are known. The first step was to obtain the most accurate variant calls available. I used phased variant calls (VCF file) that included indels and SNPs for the hg19 version of the human reference genome for an individual NA12877 from CEPH/Utah pedigree 1463 from the Platinum Genomes Project (PGP) [205] (Figure 11). This project generated deep (50× average), whole-genome sequencing data of 17 individuals in a three-generation pedigree.

The project used variant calls from six different informatics pipelines and two different sequencing technologies. Conflicts between call sets were determined by inheritance-based validation. This dataset is widely considered to represent the most accurate set of variant calls that can be achieved with current methods.

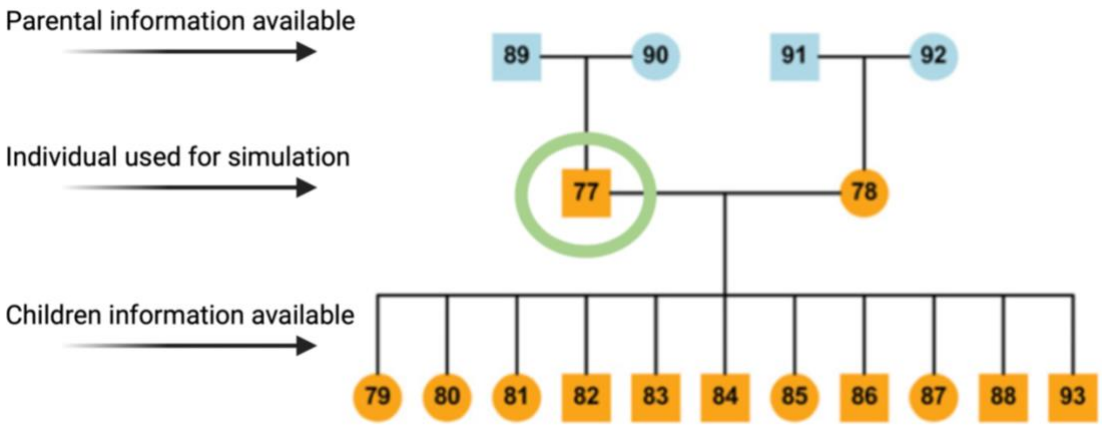


Figure 11. **Pedigree of the family of an individual NA12877 used to generate simulated genomic data.**
The high-quality genomic data from a large family that was used in Platinum Genomes Project (CEPH pedigree 1463) allowed the generation of accurate variant calls. The number for each individual represents a suffix to NA128 to generate the Coriell ID. Circled individual NA12877 was used for simulation. Modified from Eberle *et al.*, 2016 [205], Copyright 2017 (Licensed under CC BY).

2.2.3.2 GOLD STANDARD GENOMES

In order to generate accurate simulated genomic data, I generated 'gold standard' parental genomes where the exact sequence of whole genomes and each allele is known. These were later used to simulate RNA-seq and whole genome sequencing.

I used vcf2diploid within AlleleSeq [151] to incorporate high confidence phased variants from hg19 VCF from the PGP [205] for individual NA12877 into the UCSC GRCh37 reference

genome to obtain 2 phased personalised parental genomes. The output is a file for each chromosome, which I then concatenated into a complete genome per parental genome. The output from vcf2diploid also contained map files that contain coordinates of the variants and how they correspond between the parental and reference genomes, and chain files that are needed to convert reference annotation coordinates to personal ones. I used liftOver separately on each genome supplying GENCODE annotation for the reference genome and the maternal/paternal chain files to generate the maternal and paternal annotation files. I called these gold standard genomes.

The outputs from the vcf2diploid are maternal and paternal genomes. To determine the origin of either of the personalised genomes (whether the output file named maternal/paternal was NA12889/90), I downloaded 1000 genome variants calls from //ftp.1000genomes.ebi.ac.uk//vol1/ftp/technical/reference/phase2_reference_assembly_sequence/hs37d5.fa.gz. From this I compared the haplotype phasing in either of the parents and compared that to the gold standard maternal and paternal genomes to know which background they come from. I used SAMtools faidx to obtain sequence from the reference genome, 20bp up and downstream from a variant present in 1000 genomes variant calls. Then I searched this sequence in gold standard maternal and paternal genomes. Because the coordinates were shifted not all sequences were found but after 10 matching hits in both genomes, I was confident of the background and named the files appropriately.

2.2.3.3 DNA SEQUENCING SIMULATION

Simulated genomic data is important for benchmarking different bioinformatics tools, which have bloomed since the advancement of short-read sequencing [206]. Each of these tools has limitations, and therefore, validation and comparison of these are crucial. In general, the underlying ground truth behind real genomic data (such as the location of every single genetic variant, or the exact ratio of alleles at each heterozygous site) is unknown [207], and therefore, it alone is not enough to use for validation of other computational tools. With simulated genomic data however, different parameters can be controlled (eg. error rate or read length) and a large amount of desired data can be quickly generated. Most importantly, however, the underlying truth (sequence and location) is also known, allowing

researchers to test for the accuracy of their given approach [206]. For example, ART software [208] has been used to test a tool that performs quality recalibration of short mapping reads [209]. It is important for the simulators to mimic real genomic and biological features including GC-content and nucleotide substitutions, and sequencing platform-specific features including read length and fragment size distribution [206]. Therefore, numerous WGS simulation tools have been created including wgsim [210], Mason [211] and GenSIM [212]. Different simulation tools differ slightly in their performance on initial parameter options, read features, base-calling errors, accounting for PCR amplifications, quality scores and sequencing depth [207]. However, ART performs well relative to other popular simulators and is also computationally cost-effective [206]. It is easy to use and is a popular tool with over 1000 citations, and was therefore chosen for the analysis in this section.

I used the gold standard genomes to generate simulated whole genome sequencing (WGS) data in order to then use this for variant calling that then would be used as an input for PAC. To simulate WGS, I used ART software (Q Version 2.5.8) [208] on each of the parental genomes separately.

As input, ART requires parameters related to insert size, read length, coverage, and standard deviation of fragment length. To obtain a realistic simulation, I acquired these parameters from real WGS data for sample HPSI0114i-eipl_1 from the HipSci Project [200]. Initially, I used a BWA index on the UCSC GRCh37 reference genome. I mapped HPSI0114i-eipl_1 DNA-seq reads with BWA-MEM (version 0.7.17) [79]. I selected reads that were properly paired (-f 0x0002 using SAMtools) and uniquely mapped (NH:i:1 flag) and removed PCR duplicates using SAMtools [210]. I then used SAMtools-stats and obtained the following parameters: 841407464 properly paired reads, read length 151 bp (I used 150 for ART as it is the maximum read length possible), insert size average: 479.4, insert size standard deviation: 116.5, coverage = 40× ((841407464 reads×150bp)/3101788170 (size of genome) = 40.68979). Since the final coverage is 40×, I set coverage to 20× for maternal and 20× for paternal that will be combined later.

I used the following options for ART simulation: art illumina paired end reads, sequencing system: HiSeqX TruSeq, read length 150bp, fold coverage 20, mean fragment length 479, standard deviation 117.

I combined forward reads from the maternal and paternal RNA-seq simulation, and the same for reverse reads to generate wgs1.fq and wgs2.fq.

2.2.3.4 GATK VARIANT CALLING ACCURACY

To generate realistic genotypes as input for PAC, I performed GATK variant calling on simulated WGS data. Simulated WGS reads were aligned to the GRCh37 reference genome with BWA-MEM (version 0.7.17) [79]. Following this, variant calling was performed with GATK v. 4.0.12.0 according to recommended best practices [27]. This section will briefly describe the GATK variant calling best practices at the time of performing this analysis. The options and parameters for the different steps in the pipeline are listed in Table 2.

GATK variant calling starts with generating a uBAM (GATK prefers this data storage format as it allows it to store more data) from FASTQ. For this, I used picard.jar FastqToSam. This tool converts the FASTQ file to an unaligned BAM or SAM file. Then, I used picard.jar SortSam on the output. This tool sorts the SAM or BAM file by some property of the SAM file. This was followed by picard.jar MarkIlluminaAdapters, which adds adapter-trimming tags. And then again picard.jar SortSam.

I used picard.jar CreateSequenceDictionary to generate a dictionary for the reference genome. I piped together picard SamToFastq, BWA-MEM and picard MergeBamAlignment. SamToFastq takes read identifiers, read sequences, and base quality scores from SAM or BAM files to write a Sanger FASTQ format file. The options also specify removal of adapter sequences marked earlier by MarkIlluminaAdapters. BWA-MEM aligns simulated whole genome sequencing onto the reference genome. Picard MergeBamAlignment merges information from the uBAM (in the first step) and aligned BAM (previous step) conserving the read data. This was followed by picard.jar SortSam and then I performed SAMtools index to index the BAM file. I used picard.jar MarkDuplicates to identify duplicate reads followed by picard.jar SortSam.

BaseRecalibrator was then used to mask sites I downloaded (from ftp://gsapubftp-anonymous@ftp.broadinstitute.org/bundle/hg19/*) and was used on the following files:

- Mills_and_1000G_gold_standard.indels.hg19.sites.vcf.gz
- dbsnp_138.hg19.vcf.gz, 1000G_phase1.indels.hg19.sites.vcf.gz
- 1000G_phase1.snps.high_confidence.hg19.sites.vcf.gz

To index, I used gatk IndexFeatureFile. I generated the index on the reference genome with SAMtools faidx.

To remove the inbreeding coefficient, I used gatk BaseRecalibrator. I then used gatk ApplyBQSR, and BaseRecalibrator again, followed by gatk AnalyzeCovariates.

To call variants and indels by local assembly of haplotypes I used gatk HaplotypeCaller. Then to obtain variant quality scores I used gatk VariantRecalibrator. To filter variants based on their score I used gatk ApplyVQSR. gatk VariantRecalibrator and gatk ApplyVQSR were then repeated for indels.

Variants were then phased with Shapeit2 [186] using the 1000 Genomes phase 3 reference panel with standard parameters that were later supplied into PAC.

GATK STEP	OPTIONS
FastqToSam	READ_GROUP_NAME=HSXt LIBRARY_NAME=illumina_HSXt PLATFORM_UNIT=HSXt PLATFORM=illumina
SortSam	SORT_ORDER=queryname
SamToFastq	CLIPPING_ATTRIBUTE=XT CLIPPING_ACTION=2 INTERLEAVE=true NON_PF=true
MergeBamAlignment	CREATE_INDEX=true ADD_MATE_CIGAR=true CLIP_ADAPTERS=false CLIP_OVERLAPPING_READS=true INCLUDE_SECONDARY_ALIGNMENTS=true MAX_INSERTIONS_OR_DELETIONS=-1 PRIMARY_ALIGNMENT_STRATEGY=MostDistant ATTRIBUTES_TO_RETAIN=XS
MarkDuplicates	CREATE_INDEX=true
IndexFeatureFile	-F Mills_and_1000G_gold_standard.indels.hg19.sites.vcf.gz
BaseRecalibrator	--known-sites Mills_and_1000G_gold_standard.indels.hg19.sites.vcf.gz --known-sites dbsnp_138.hg19.vcf.gz --known-sites 1000G_phase1.indels.hg19.sites.vcf.gz --known-sites 1000G_phase1.snps.high_confidence.hg19.sites.vcf.gz
VariantRecalibrator (for SNPs)	--resource hapmap,known=false,training=true,truth=true,prior=15.0: hapmap_3.3.hg19.sites.vcf.gz --resource omni,known=false,training=true,truth=false,prior=12.0: 1000G_omni2.5.hg19.sites.vcf.gz --resource 1000G,known=false,training=true,truth=false,prior=10.0: 1000G_phase1.snps.high_confidence.hg19.sites.vcf.gz

	--resource dbsnp,known=true,training=false,truth=false,prior=2.0: dbsnp_138.hg19.vcf.gz -an DP -an QD -an FS -an SOR -an MQ -an MQRankSum -an ReadPosRankSum --mode SNP -tranche 100.0 -tranche 99.9 -tranche 99.0 -tranche 90.0 --max-gaussians 4
VariantRecalibrator **(for indels)**	--resource mills,known=false,training=true,truth=true,prior=12.0: Mills_and_1000G_gold_standard.indels.hg19.sites.vcf.gz --resource dbsnp,known=true,training=false,truth=false,prior=2.0:dbsnp_138.hg19.vcf.gz -an QD -an DP -an FS -an SOR -an MQRankSum -an ReadPosRankSum --mode INDEL -tranche 100.0 -tranche 99.9 -tranche 99.0 -tranche 90.0 --max-gaussians 4
ApplyVQSR **(for SNPs)**	-mode SNP --truth-sensitivity-filter-level 99.9
ApplyVQSR **(for indels)**	-mode INDEL --truth-sensitivity-filter-level 99.9

Table 2. **The different steps and options used within the GATK variant calling pipeline.**

2.2.3.5 RNA SEQUENCING SIMULATION

To obtain allelic counts from simulated data, I used gold standard genomes to generate simulated RNA sequencing data using RSEM v1.3.1 [213]. For the simulated RNA-seq data to be as realistic as possible, I obtained sequencing parameters from real RNA-seq data from parents of NA12877 that were used as inputs in RSEM.

I used bowtie-build (v 1.2.2) to index the maternal and paternal genomes. Then I used rsem-prepare-reference with each of the parental genomes and their annotations separately. This process prepares the transcript references for RSEM.

Then I used rsem-calculate-expression which calculates the gene and isoform expression from the input BAM file. For this, I used the real parents of NA12877, NA12889 and NA12890 RNA-seq from lymphoblastoid cell lines (LCLs) that were obtained from the Geuvadis Project [64] to get realistic numbers. The reads were trimmed and mapped to the hg19 reference genome using STAR v.2.5.1a [82] with default parameters. The output was a single matrix of expression levels for each transcript in the GENCODE v19 annotations.

Then I simulated RNA-seq reads from both parental genomes with the rsem-simulate-reads function. From the previous step, I obtained the following input options for this step: the fraction of reads coming from background noise (paternal: 0.27 and maternal: 0.19) and the total number of reads to be simulated (paternal: 40.9M and maternal: 28.1M). The simulated reads from the parents (FASTA files) were then merged into a single RNA-seq sample, representing the simulated transcriptome of individual NA12877.

2.2.3.6 QUANTIFICATION OF IMBALANCE AT HETEROZYGOUS SITES

To acquire ‘ground truth allele counts’, I obtained maternal and paternal allele counts at heterozygous genome positions of NA12877 from the PGP VCF file. Each RSEM simulated transcript has following information provided in the header:

- the read index starting from 0
- the read direction, 0 being forward strand ('+') and 1 being reverse strand ('-')
- ID representing the transcript this read is simulated from
- the start position of the simulated read in strand dir of transcript sid
- the insert length of the simulated read

The genomic coordinates of the simulated RNA-seq reads were obtained using custom scripts which were based on following rules for RSEM simulated reads:

1. If gene in forward direction:
    - read1 forward
        - to find start of read: count forwards from start of gene from corresponding maternal/paternal annotation file
        - to find end of read: count forward 75bp from above
    - read2 reverse
        - to find start of read: count forward length from start of read1
        - to find end of read: count 75bp backward from above
2. If read in reverse direction:
    - read1 reverse
        - to find position on id (end of read): count backwards from end of gene
        - to find start of read: count 75bp backwards from above
    - read2 forward
        - to find end of read: count backward length from end of read1
        - to find start of read: count 75bp forward from above

⇒ If gene reverse, flip above

I used LiftOver to convert read locations from the parental genomes to the reference genome using chain files generated by AlleleSeq. I then counted the number of reads from reference and alternative alleles that overlapped all heterozygous positions in NA12877 based on the PGP variant calls. To do this I used sim.isoforms.results output file from RSEM to obtain the transcript ID and GTF file from vcf2diploid for exome positions in the reference genome.

These allele counts were then combined for each site to create ground truth allele counts in the offspring. For all subsequent analyses, we used heterozygous sites with at least 20× read coverage (sum of reference and alternative allele counts) and blacklist (from ENCODE ENCFF001TDO.bed) and HLA regions (obtained from phASER) removed. Figure 12 summarises all the steps leading to the generation of ground truth ASE calls.

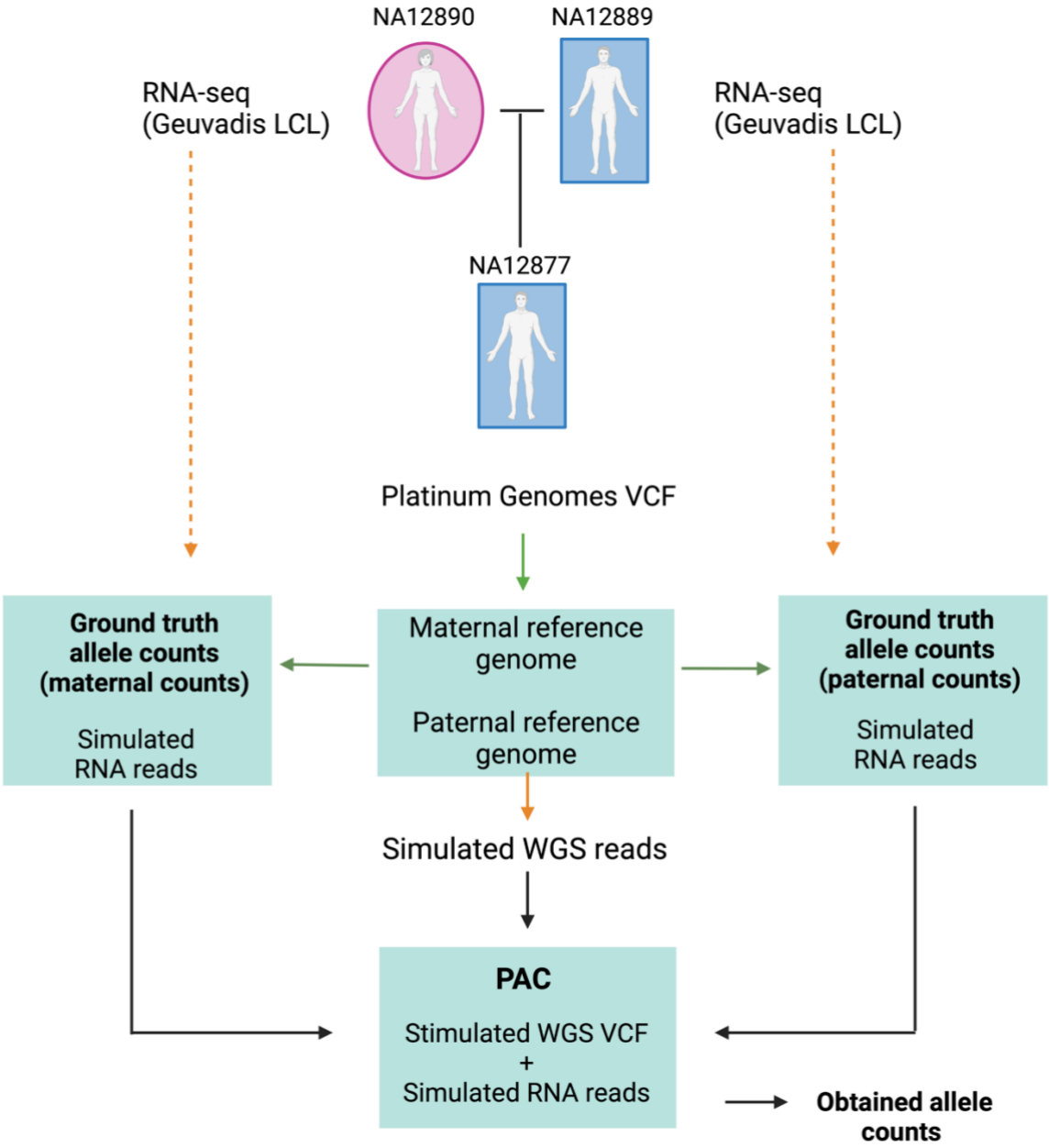


Figure 12. **Ground truth genomic data generation for individual NA12877.**
In order to obtain realistic simulated genomic data to test PAC against, PGP VCF was used, where the variants were verified using multiple sequencing platforms and analysis methods, and conflicting calls were resolved using parental genomic information. Phased variants from the PGP were used together with a reference genome to generate ground truth genomes. Ground truth genomes were then used to simulate RNA-seq reads based on sequencing parameters obtained from the Geuvadis Project that generated RNA-seq reads for the actual parents, individuals NA12890 and 12889, for LCLs. The simulated RNA-seq reads were then used to count coverage at each heterozygous site, called ground truth allele counts. Ground truth genomes were also used to simulate WGS with sequencing parameters for this obtained from HipSci sample. Simulated WGS were used to obtain variant calls using GATK best practices. This VCF, together with simulated RNA-seq reads, were used for PAC to obtain allelic count data that were compared against ground truth allele counts at heterozygous sites. Figure is from Saukkonen *et al.*, 2022 [109]. Created with BioRender.com

2.2.4 TESTING DIFFERENT PAC PARAMETERS

Once I had realistic ground truth allele counts that acted as a baseline to which to compare the performance of PAC, I set to improve PAC. I tested features I hypothesised to affect the allelic quantification. I generated different versions of PAC (Table 3), incorporating and removing the following features on the preliminary PAC pipeline and compared the allelic counts obtained to those in ground truth data:

- phASER: I tested the consequence of improved phasing with phASER with read-aware mode, which improves local phasing by considering whether nearby genetic variants fall on the same or opposite reads (or pairs). I supplied the PAC pipeline with phased VCF obtained from the GATK pipeline for phasing by phASER. PAC was also tested without the phASER step.

- Recovering multi-mapping reads: I tested if rescuing of reads that map to multiple locations would improve allelic quantification. I used RSEM [213], which takes the original alignment from STAR (containing all reads aligned to transcriptome coordinates, including reads that align to multiple locations) and re-aligns the data using the --sampling-for-bam flag to output a single location for each read based on its posterior probability generated from estimated abundances. Additional reads aligned by RSEM that were not uniquely aligned using STAR were then added to the final BAM file.

- Trimming and soft-clipping: I tested how trimming and soft-clipping, the common RNA-seq read filtering steps, affect the allelic quantification. The filters for trimming were stringency of 3bp, removing adaptors and terminal bases with Phred qualities lower than 30. Soft-clipping was performed during STAR mapping within the PAC pipeline with standard parameters.

For each parameter, heterozygous sites present in the ground truth and at >=20× coverage were selected.

Versions	Trimming	Soft-clipping	phASER	Multi-mapping
1	+	-	+	+
2	+	-	-	-
3	+	-	+	-
4	+	-	-	+
5	-	+	+	+
6	-	+	-	-
7	-	+	+	-
8	-	+	-	+

Table 3. **Different versions of PAC generated for testing.**
The rows represent combinations of included (+) or excluded (-) parameters within PAC version.

2.2.5 FINAL PAC PIPELINE

The final PAC starts as the preliminary pipeline, however at the STAR mapping stage the parameters were as follows: including soft-clipping, no trimming (opposite to the preliminary pipeline), keeping properly paired (-f 0x0002 using SAMtools) and uniquely mapped (NH:i:1 flag) reads. RSEM is used to assign a single location for multi-mapping reads based on the read depth of uniquely aligned reads and then incorporates these reads into the final aligned files. The final pipeline also produces allele counts at a haplotypic level using phASER Gene AE. The pipeline is illustrated in Figure 13.

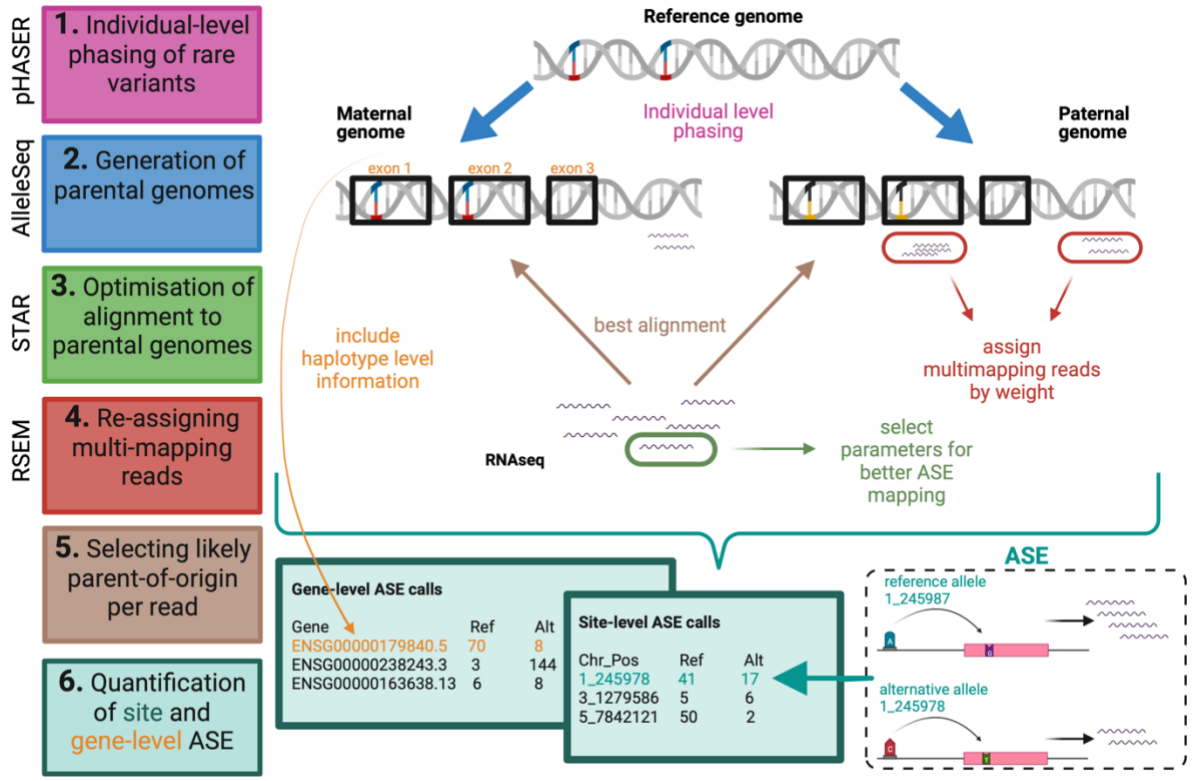


Figure 13. **Final PAC pipeline**.
A schematic describing the main steps, features and outputs of PAC after refining different parameters. Figure is from Saukkonen *et al.*, 2022 [109]. Created with BioRender.com.

2.2.6 OUTLIER ANALYSIS

To investigate if particular genomic features are enriched in the heterozygous sites from PAC that have larger differences in their allelic ratios from those in the ground truth data, I selected sites where the reference allele ratio between PAC analysis and ground truth ASE was more than 10% and had at least 20× coverage in both analyses. The variants were annotated with wANNOVAR (http://wannovar.wglab.org), which is the web version of ANNOVAR [214]. The output provided information on how the variant affected gene structure, the functional consequences, functional importance scores, and the location in the gene.

2.3 RESULTS

2.3.1 PRELIMINARY WORK

2.3.1.1 BASIC OVERVIEW

When I joined the research group, there was a preliminary pipeline in place to improve allelic quantification, later termed the ***P***ersonalised ***A***SE ***C***aller (PAC) pipeline. It implements a series of steps to detect and quantify ASE events more accurately. Briefly, PAC generates personalised diploid genomes and aligns RNA-seq reads to both parental genomes. It selects the best alignment and generates site-level allelic counts. More details are in the methods section 2.2.1.

To test this preliminary PAC pipeline, I submitted RNA-seq data obtained from 10 HipSci iPSC and iPSC-derived sensory neuron samples into PAC. I also aligned the same RNA-seq reads to the GRCh37 reference genome with STAR [82]. I then compared the reference allele ratios (RARs) at heterozygote sites between the two methods (Table 4).

At first, I looked at coverage at the heterozygous sites. If PAC improves ASE biases, fewer reads are expected not to be aligned relative to the standard alignment approach, which would remove reads due to biases described earlier in the chapter. Therefore, higher coverage at the heterozygous sites is expected in PAC. Table 4 shows that the number of sites retained across individuals when RNA-seq data was processed with PAC is greater at >=10× coverage than in the standard alignment approach in both cell types. Figure 14 shows that the number of sites in PAC is higher than in the standard alignment approach in iPSCs in all individuals.

Donor ID	Standard alignment		PAC	
	iPSC	Neuron	iPSC	Neuron
HPSI0913i-eika_2	20936	20433	21522	20950
HPSI0114i-eipl_1	20715	20945	21255	21415
HPSI0114i-kolf_3	22261	20882	22872	21366
HPSI0214i-kucg_2	22497	22376	23079	22894
HPSI0114i-oevr_3	19906	35364	20534	36254
HPSI1113i-podx_1	22723	24667	23287	25281
HPSI0314i-qaqx_1	19652	26341	20234	26929
HPSI0114i-rozh_5	17707	21730	18103	22305
HPSI1013i-wuye_2	20148	20808	20679	21299
HPSI0314i-xugn_1	20885	24595	21419	25198
Mean	20743.0	23814.1	21298.4	24389.1
Mean (between methods)	22278.55		22843.75	

Table 4. **The coverage at the heterozygous sites obtained from the preliminary PAC and standard mapping approach in HipSci samples.**
The heterozygous sites at >=10× coverage in iPSC and iPSC-derived sensory neurons from 10 HipSci donors are shown.

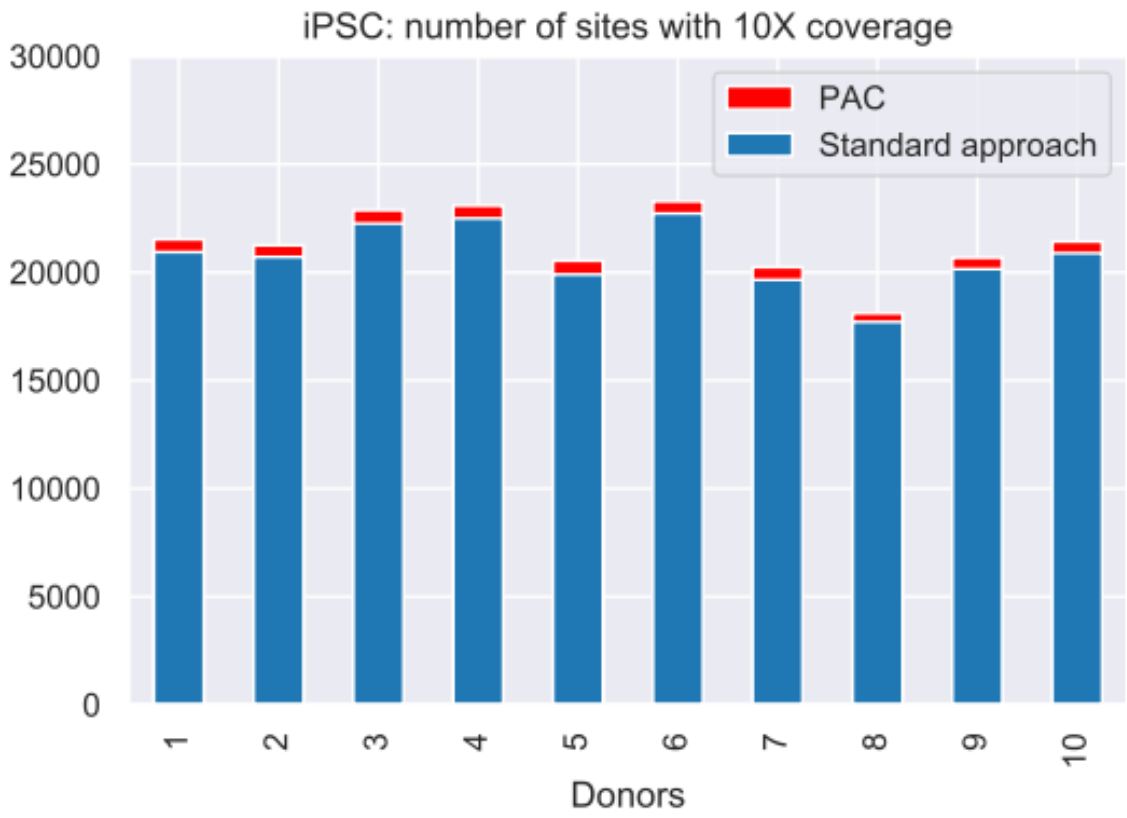


Figure 14. **Number of heterozygous sites in preliminary PAC and standard mapping across HipSci iPSC samples.** The heterozygous sites at >=10× coverage in iPSCs from 10 HipSci donors are shown.

Another assumption from improved ASE detection is that the mean RAR should be closer to 0.5 when considered across all sites. This is because on average, reads are expected to be expressed at an equal rate from each allele. Genes under ASE will deviate from this; however, they should offset across the genome, with some showing reference allele skew and others showing alternative allele skew. Hence the distribution of allele ratios should follow a bell-shaped curve with the mean close to 0.5. Figure 15 shows the RAR in a single example HipSci donor in iPSC closely follows the expected bell-shaped curve with most reads closer to 0.5 reference allele ratio in the preliminary PAC. The trend is the same across all individuals and in both cell types (data not shown). Figure 15 shows that some reference bias still remains, potentially demonstrating the scope for improvement in PAC.

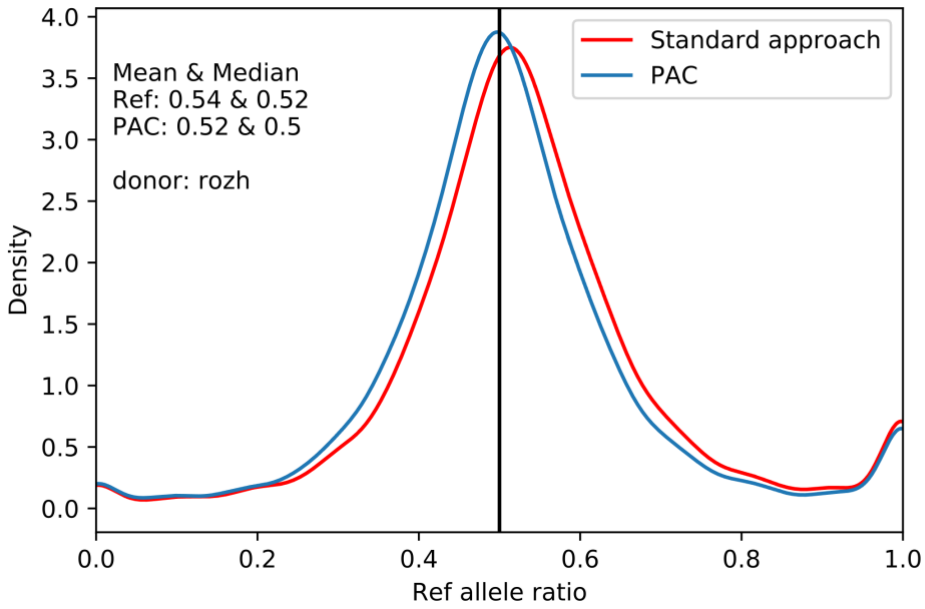


Figure 15. **The proportion of reference allele ratio in PAC and standard alignment approach.**
The data shown is for a single HipSci donor, rozh. Shown are the mean and median reference allele ratios for the individual. The bias for reference allele ratio demonstrates potential limitations in the preliminary PAC.

2.3.1.2 ASE

I then examined the ability of PAC to detect sites under ASE and the tissue specificity of ASE sites in iPSCs and iPSCs -derived sensory neuronal cells. The heterozygous ASE sites obtained from PAC were filtered for ASE with binomial test ($P<0.05$) and Bonferroni-adjustment for a more stringent selection. Table 5 shows a summary of ASE sites from all donors. Figure 16 shows a Venn diagram of these results.

The higher number of ASE sites in neurons than in iPSCs might be due to higher coverage in neuronal cells (Table 4), however, previous research [169, 215] has demonstrated a higher proportion of ASE in neuronal cell types. Table 5 and Figure 16 show that at more stringent ASE selection criteria there are fewer cell-type specific ASE events, and the majority of ASE events are retained across differentiation. This is most likely due to imprinting and other monoallelic events that retain the expression pattern. At $P<0.05$, more subtle effects are captured where cell-type specific events can be seen.

	iPSC	Neuron
Num. of ASE sites (P<0.05)	2394.3	2538.3
Num. of ASE sites (Bonferroni-adjusted)	690.8	705.0
Common ASE sites between iPSCs and neurons (P<0.05)	1174.9 (49% of sites)	1174.9 (46% of sites)
Common ASE sites between iPSCs and neurons (Bonferroni-adjusted)	484.5 (70% of sites)	484.5 (69% of sites)
Cell-type specific ASE sites (P<0.05)	1219.4 (51% of sites)	1363.4 (54% of sites)
Cell-type specific ASE sites (Bonferroni-adjusted)	206.3 (30% of sites)	220.5 (31% of sites)

Table 5. **Comparison of heterozygous sites under ASE in two cell types from PAC.**
Data is shown for iPSC and iPSC-derived sensory neurons in 10 HipSci samples.

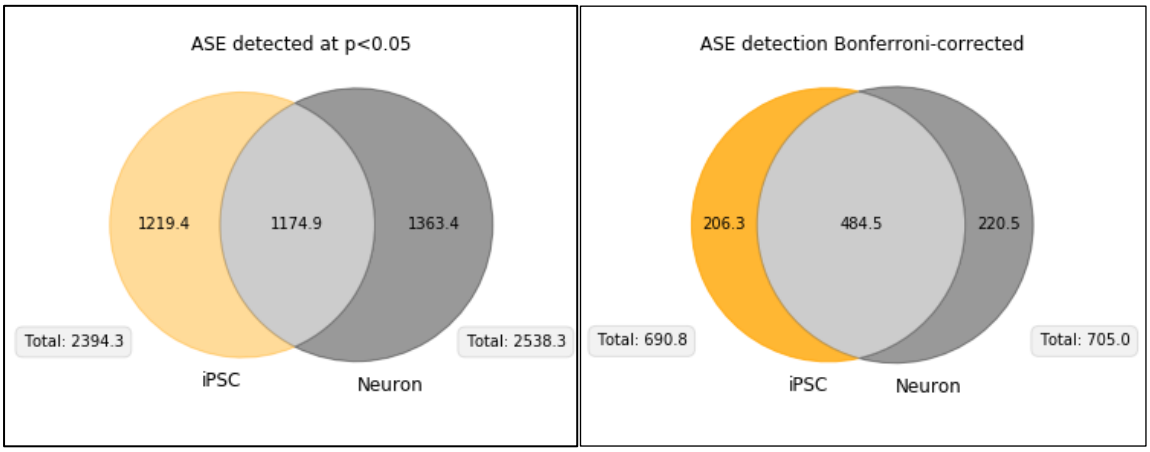


Figure 16. **Venn diagram of heterozygous cell-type specific and shared sites under ASE from PAC.**
Data is shown for iPSC and iPSC-derived sensory neurons in 10 HipSci samples. The results demonstrate that at stringent ASE detection criteria, most ASEs are shared between cell types whereas more subtle effects are seen in a cell-type specific manner.

I examined the RAR of heterozygous sites that are cell-type specific (Figure 17, A and B panels) and shared between tissues (Figure 17, C and D panels). The results from one donor are illustrated but every HipSci sample follows a similar trend (data not shown).

At a P<0.05 ASE detection criteria (Figure 17 A), cell-type specific ASE sites show a more subtle effect with a distribution closer to 0.5. The drop in ratio at 0.5 represents sites that are not under allelic imbalance and hence are not under significant ASE. The ASE sites selected with more conservative criteria with Bonferroni-adjustment (Figure 17 B) only detect sites with high effect.

Shared ASE sites (Figure 17 C and D) show stronger effects on allelic imbalances, as the distribution is closer to 1. This monoallelic expression potentially reflects imprinted genes, as they maintain their expression origin during cellular differentiation.

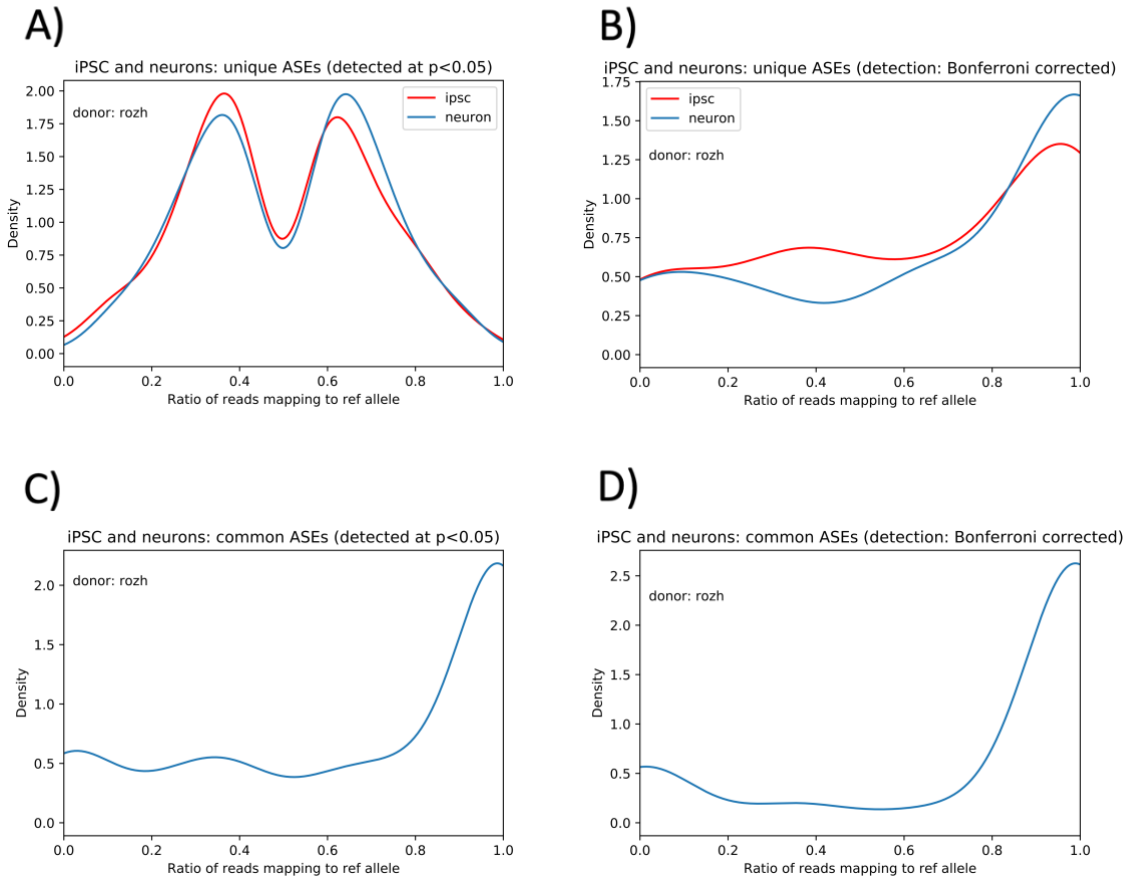


Figure 17. **The reference allele ratio (RAR) of cell type specific and shared ASE sites obtained from PAC**.
A and B panels demonstrate the density of the reference allele ratio (RAR) at heterozygous sites under ASE that are shared between iPSCs and neuronal cells (panel A detected with binomial test at P<0.05; panel B detected with binomial test with P-values Bonferroni corrected). C and D panels demonstrate the density of RAR at heterozygous sites under ASE that are specific to iPSCs and neuronal cells (panel C detected with binomial test at P<0.05; panel D detected with binomial test with P-values Bonferroni corrected). Data shown is for a single HipSci donor.

2.3.1 3 GENE ONTOLOGY ENRICHMENT OF GENES UNDER ASE

Next, I explored the possibility that genes containing an ASE site are enriched for a particular biological function. Because the donors are healthy, I hypothesised that no disease terms would be enriched. Because ASE is not a rare occurrence in healthy individuals [108], I hypothesised that if any terms would be enriched, those would likely be in genes that are highly expressed in the cell type under study as this increases statistical power to detect any imbalances.

For each of the 10 donors, genes under ASE from iPSCs were studied for GO enrichment. Genes with at least one site with >=10× coverage were selected for the background gene list, and enrichment was performed on genes under ASE (detected at $P<0.05$ and Bonferroni-adjusted).

The David Bioinformatics Resource 6.8 Functional Annotation Tool [204] was used to examine gene enrichment. Biological categories enriched for genes undergoing ASE at $P<0.05$ were observed only for 2 donors, one term per sample (Table 6). For genes under ASE at Bonferroni-adjustment, there were 3 donors with terms enriched for biological categories with all enriched terms from these individuals shown in Table 6.

The enriched terms with the highest P-values include ‘stem cell population maintenance (GO:0019827)’ and stem cell population maintenance (GO:0019827), which are expected for iPSC self-renewal function. MHC terms were also enriched. MHC regions are highly polymorphic [216] and therefore likely to exhibit ASE, however, their expression is associated with immune cells. Other terms included those relating to the endoplasmic reticulum and extracellular region, which are related to basic cellular processes.

Category	Term	P-Value	P-value corrected	Sample
ASE detected with binomial test P<0.05				
Cellular component	condensed chromosome (GO:0000793)	2.93×10^{-5}	2.79×10^{-2}	HPSI0913i-eika_2
Cellular component	extracellular matrix (GO:0031012)	3.77×10^{-5}	3.84×10^{-2}	HPSI0214i-kucg_2
ASE detected with binomial test P-value Bonferroni-adjusted				
Biological process	maintenance of cell number (GO:0098727)	1.89×10^{-8}	6.65×10^{-5}	HPSI0913i-eika_2
Biological process	stem cell population maintenance (GO:0019827)	1.25×10^{-7}	2.19×10^{-4}	HPSI0913i-eika_2
Cellular component	lumenal side of endoplasmic reticulum membrane (GO:0098553)	1.98×10^{-6}	9.15×10^{-4}	HPSI0913i-eika_2
Cellular component	integral component of lumenal side of endoplasmic reticulum membrane (GO:0071556)	1.98×10^{-6}	9.15×10^{-4}	HPSI0913i-eika_2
Cellular component	MHC protein complex (GO:0042611)	1.30×10^{-5}	3.01×10^{-3}	HPSI0913i-eika_2
Molecular function	peptide antigen binding (GO:0042605)	1.91×10^{-5}	9.66×10^{-3}	HPSI0913i-eika_2
Cellular component	MHC class II protein complex (GO:0042613)	8.66×10^{-5}	1.33×10^{-2}	HPSI0913i-eika_2
Cellular component	MHC class I protein complex (GO:0042612)	3.40×10^{-4}	3.87×10^{-2}	HPSI0913i-eika_2

Cellular component	ER to Golgi transport vesicle membrane (GO:0012507)	4.16×10^{-4}	3.78×10^{-2}	HPSI0913i-eika_2
Cellular component	extracellular region (GO:0005576)	1.86×10^{-5}	9.03×10^{-3}	HPSI0114i-kolf_3
Cellular component	extracellular region part (GO:0044421)	2.15×10^{-5}	5.23×10^{-3}	HPSI0114i-kolf_3
Cellular component	integral component of lumenal side of endoplasmic reticulum membrane (GO:0071556)	1.68×10^{-4}	2.70×10^{-2}	HPSI0114i-kolf_3
Cellular component	lumenal side of endoplasmic reticulum membrane (GO:0098553)	1.68×10^{-4}	2.70×10^{-2}	HPSI0114i-kolf_3
Cellular component	extracellular space (GO:0005615)	1.91×10^{-4}	2.30×10^{-2}	HPSI0114i-kolf_3
Cellular component	MHC class II protein complex (GO:0042613)	2.71×10^{-4}	2.61×10^{-2}	HPSI0114i-kolf_3
Cellular component	MHC protein complex (GO:0042611)	6.04×10^{-4}	4.79×10^{-2}	HPSI0114i-kolf_3
Cellular component	MHC protein complex (GO:0042611)	3.70×10^{-5}	1.63×10^{-2}	HPSI0114i-eipl_1
Cellular component	extracellular region (GO:0005576)	1.68×10^{-4}	3.66×10^{-2}	HPSI0114i-eipl_1
Cellular component	MHC class II protein complex (GO:0042613)	2.00×10^{-4}	2.91×10^{-2}	HPSI0114i-eipl_1

Table 6. **The enrichment of genes under ASE from HipSci samples.**
The GO terms and categories enriched for genes under ASE (detected by binomial test at P<0.05 in 2 individuals, and with Bonferroni-adjustment detected in 3 individuals) from 10 healthy HipSci iPSC samples. Shown are uncorrected P-values and Benjamini-Hochberg adjusted P-values for the terms. All terms from every individual are shown in the table.

2.3.2 GENERATING SIMULATED GENOMIC DATA

The work in HipSci data shows that the initial PAC pipeline leads to an improvement over standard alignment approach. However, a bias towards the reference allele remains, suggesting the pipeline can be further improved. In order to develop the pipeline, I generated highly realistic simulated genomic data where the underlying ground truth data is known. This allowed me to assess the performance of the pipeline and determine how different parameters affected the performance.

2.3.2.1 GOLD STANDARD GENOMES

In order to be able to test and develop PAC, I simulated realistic genomic DNA sequencing data where the exact origins of each sequencing read, as well as the locations of all genetic variants, were known. To achieve these, I first generated personalised genomes. I used Alleleseq [151] to incorporate highly accurate phased variants from the Platinum Genomes Project (PGP) into the UCSC GRCh37 reference genome to obtain personalised parental genomes where the exact genomic sequence is known. I termed these gold standard genomes.

2.3.2.2 VARIANT CALLING ACCURACY

To generate variant calls from gold standard genomes, I generated simulated WGS from maternal and paternal gold standard genomes separately. The simulation was achieved using ART [208]. To make the data realistic, I used parameters from real WGS data from HipSci sample. I then combined forward and reverse from each genome to generate wgs1.fq and wgs2.fq.

GATK variant calling was performed on simulated WGS data according to GATK best practises and phased variants with Shapeit2 [186]. Because variant calling is not perfect and introduces errors, this allowed me to examine how well GATK variant calling [27] compares

to the original PGP VCF file, in order to better understand where genetic variants may lead to errors in ASE analysis (Figure 18).

There are 4,042,773 genetic variants in the PGP VCF file for NA12877 and 4,011,226 in the GATK output, showing a true positive rate of 99.22%. The GATK VCF file also contains an additional 5,389 heterozygous variants that are not present in the original data, leading to a false positive rate of 0.134%. The GATK VCF misses 36,939 variants, with false negative rate being 0.914% (Figure 18). 1,409 sites out of 36,939 sites that GATK valiant calling missed were in the HLA regions, and none were in the backlisted gene list. 1 site out of the 5,389 that GATK falsely assigns were in the HLA regions, and 15 sites were in the blacklisted gene list. This demonstrates that large number of false positives and false negatives persist in the variant calling even after filtering out HLA and blacklisted regions, which is a common practise.

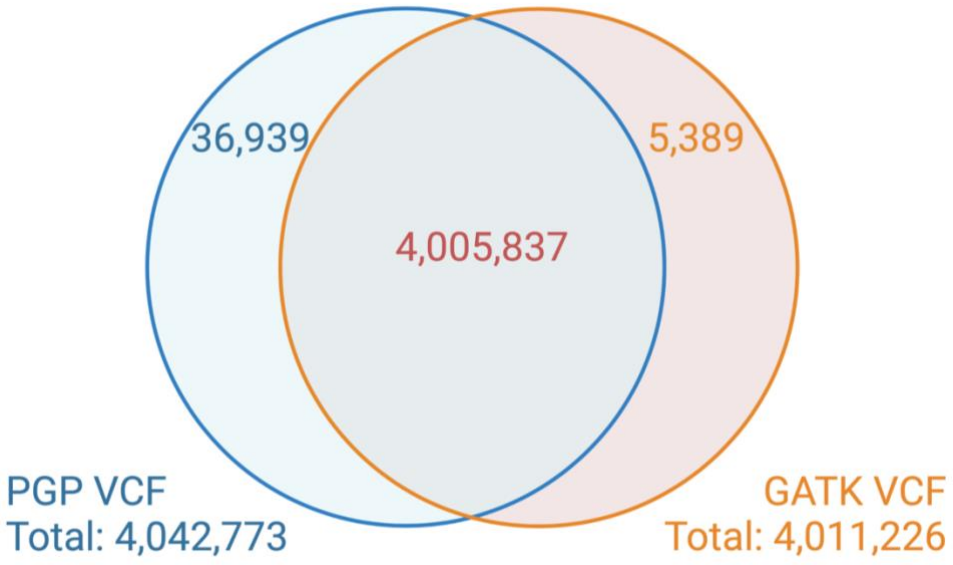


Figure 18**. Venn-diagram to demonstrate the comparison of GATK and PGP VCF files.**
The simulated data allowed me to examine the error rate in the GATK variant calling demonstrating a large number of false positive and false negative variants. Diagram not to scale.

The correct assignment of variants determines the quality of the parental genomes generated and therefore the downstream accuracy of the PAC pipeline. Therefore, I next explored the genomic locations of false positive and negative data points. Figure 19 shows that while there is a base level of variants incorrectly called or missed along most chromosomes, some areas have higher peaks of sites present in only the original PGP or GATK VCF file. For example, GATK variant calling is unable to detect the heterozygous sites in the HLA region on chromosome 6 in the original PGP VCF file, which is expected for highly polymorphic genomic regions. This region is often removed from most genetic analyses for this reason, including analysis in this thesis. GATK also falsely assigns genetic variants in chromosome Y while missing those in chromosome X. However, as I only consider autosomal genes, the peaks on sex chromosomes do not affect the analysis in this thesis.

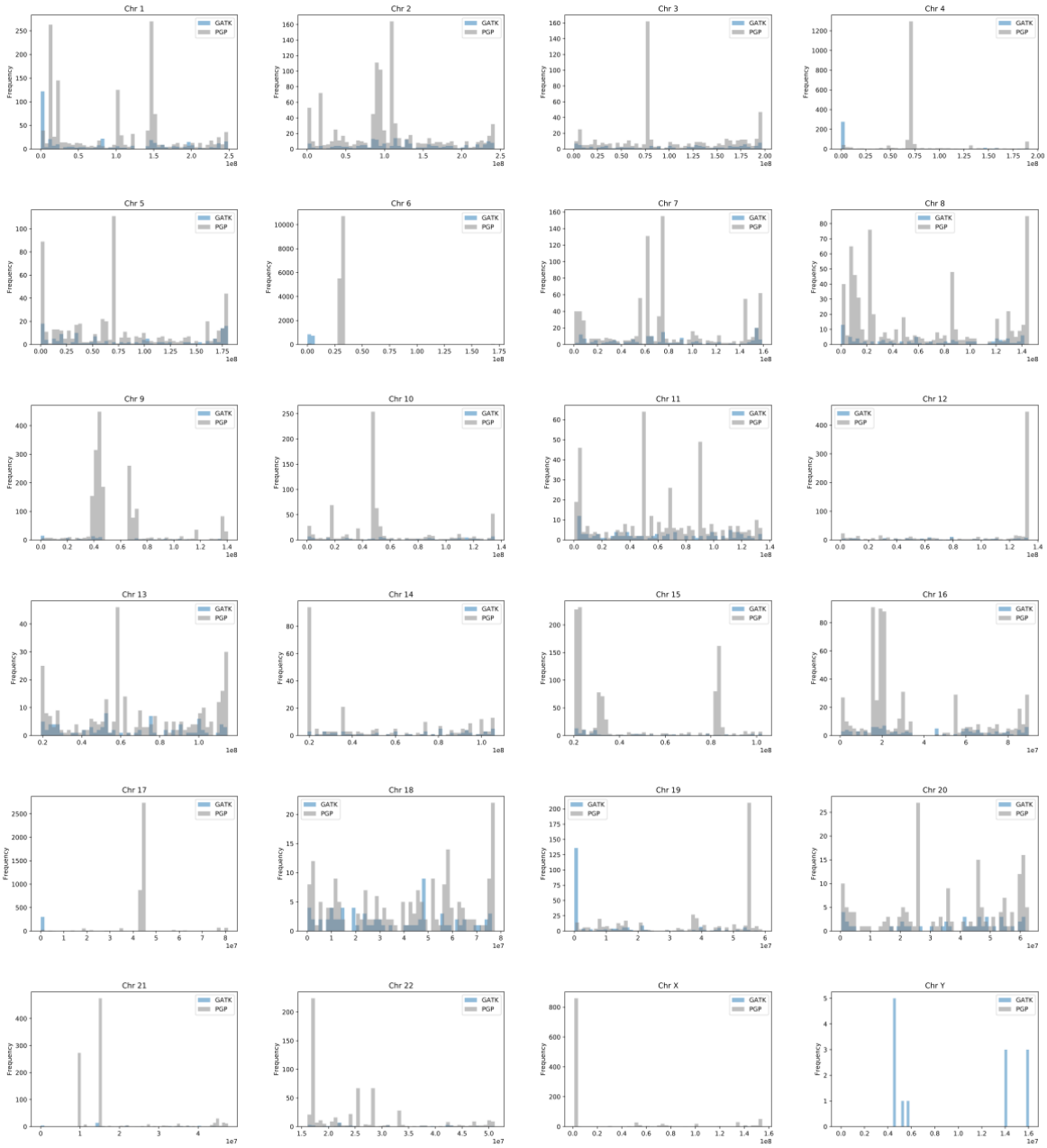


Figure 19. **The comparison of unique data points in GATK and PGP VCF files**
Grey peaks are unique variants in PGP, and thus those missed by GATK variants calling. Blue points are unique sites in GATK variant calling, those that are not present in the original PGP VCF file. Chromosome M was omitted as it did not have any genetic variants in either of the files.

2.3.2.3 QUANTIFICATION OF IMBALANCE AT HETEROZYGOUS SITES

To obtain ground truth allelic counts, I used gold standard genomes to simulate RNA-seq using RSEM [213] separately on both genomes. As an input, RSEM requires basic statistics for RNA-seq including the coverage and background noise. I obtained these from RNA-seq data generated from the real parents (NA12889 and NA12890) from Geuvadis Project [64] LCLs to get realistic parameters. I then combined RNA-seq from maternal and paternal simulations.

I quantified reference and alternative alleles from the simulated RNA-seq reads from both parental genomes that overlapped heterozygous sites from the PGP VCF file. These allele counts were then combined for each site to create ground truth allelic counts in the simulated sample. The distribution of the reference allele ratios across all >=20× sites in the ground truth data is shown in Figure 20 demonstrating it follows a bell-shaped curve as expected for unbiased data.

In the ground truth data, there are 13,211 heterozygous sites that have at least 20× coverage. This includes 499 rare variants with <1% minor allele frequency in the CEU population from the 1000 Genomes data. 1,359 variants (10.3%) were under ASE under a standard binomial test (P < 0.05, corrected for 13,211 tests). Simulated data also contained 1,237 indels (>1bp) with at least 20× coverage.

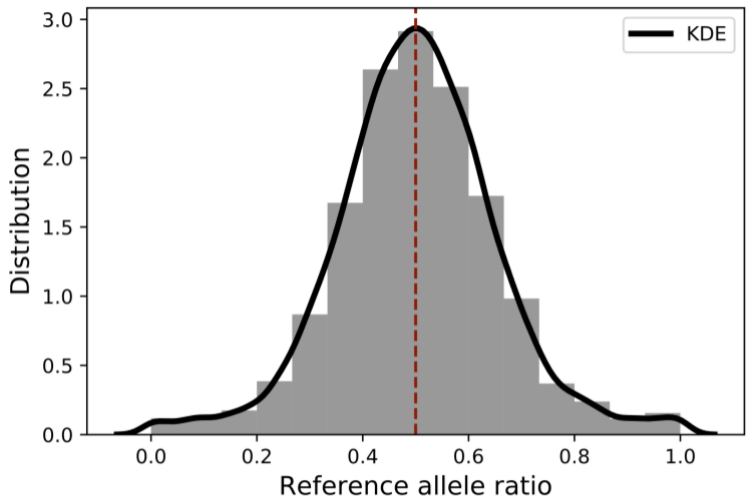


Figure 20. **The distribution of reference allele ratios in ground truth data**.
The figure shows all heterozygous sites in the ground truth data with >=20× coverage. Ratio of 0.5 (red dashed line) implies that both alleles are expressed at equal ratios. KDE = kernel density estimate.

2.3.3 DEVELOPING PAC

After showing that the preliminary PAC pipeline improves ASE detection in real HipSci data, and having created simulated ground truth data, I next tested different parameters within PAC to refine it into the final pipeline.

2.3.3.1 TESTING DIFFERENT PARAMETERS FOR PAC

In order to develop PAC, I tested how different parameters in PAC would influence allelic quantification. The parameters I tested included whether trimming or soft-clipping, together with phasing with phASER with read-aware mode, and re-allocation of multi-mapping reads with RSEM would improve the performance of PAC. I generated different versions of PAC with combinations of the following:

- With trimming, no soft-clipping, with phASER, with multi-mapping
- With trimming, no soft-clipping, without phASER, with multi-mapping
- With trimming, no soft-clipping, with phASER, without multi-mapping
- With trimming, no soft-clipping, without phASER, without multi-mapping
- No trimming, with soft-clipping, with phASER, with multi-mapping
- No trimming, with soft-clipping, without phASER, with multi-mapping
- No trimming, with soft-clipping, with phASER, without multi-mapping
- No trimming, with soft-clipping, without phASER, without multi-mapping

I ran the different versions of PAC with simulated RNA-seq reads. I measured the accuracy with the following tests:

1. I examined the number of sites that each method detected from the 13,211 sites that were present in the ground truth. The more reads that are retained, the better the alignment step is.
2. I considered the difference in the RAR from the ground truth allelic ratio. With an improved method, the difference from the ground truth is expected to decrease.

3. I measured the correlations, measured with R^2, between the method and ground truth. The stronger correlation indicated the improvement in the method.
4. I considered the number of outliers where the difference in RAR in PAC versus the ground truth was more than 10% and 20%. The improved pipeline is expected to have fewer outliers.
5. I observed the number of sites that are missed by standard alignment but picked up by PAC. These extra sites that are discarded by the standard alignment can be informative, especially in studies with small sample sizes such as in the rare disease field.

The results are shown in Table 7. The biggest improvement came from including soft-clipping instead of trimming. On average, the number of aligned reads increased by 250, whereas with phASER, the number of reads increased on average by 7, and with multi-mapping by 35.25.

The average R^2 for PAC versions with trimming was 0.9502, and without was 0.9651. The average R^2 for PAC versions with phASER and without was 0.9578 and 0.9574, respectively. The average R^2 for PAC with multi-mapping read re-allocation was 0.9606, and without it was 0.9547.

The average number of 10% outliers in PAC decreased by 95 by incorporating soft-clipping rather than trimming. The average number of these outliers decreased by 4.5 incorporating phASER. The average number of outliers decreased by 12 by re-allocating multi-mapping reads.

The parameters that performed closest to the ground truth and that became the final PAC pipeline was with phASER, multi-mapping read re-allocation, with soft-clipping and without trimming. From now on this will be called PAC.

	TRIM PHASE NO MULTIMAP	TRIM PHASE MULTIMAP	SOFT PHASE MULTIMAP	SOFT PHASE NO MULTIMAP	TRIM NO PHASE NO MULTIMAP	TRIM NO PHASE MULTIMAP	SOFT NO PHASE MULTIMAP	SOFT NO PHASE NO MULTIMAP
Sites shared with ground truth	12159	12194	12448	12415	12161	12190	12436	12405
Difference in reference allele ratio	Mean: 0.0331 Median: 0.0273	Mean: 0.0326 Median: 0.0273	Mean: 0.0248 Median: 0.0195	Mean: 0.0254 Median: 0.0196	Mean: 0.0332 Median: 0.0273	Mean: 0.0326 Median: 0.0272	Mean: 0.0249 Median: 0.0196	Mean: 0.0255 Median: 0.0196
R2 between ground truth	0.9475	0.9532	0.9681	0.96251	0.9469	0.9530	0.9679	0.9619
Outliers >20%	67	47	46	68	69	49	49	72
Outliers >10%	246	240	140	157	252	244	144	161
Sites not in standard alignment	209	242	350	318	207	233	339	311
Sites not in WASP	606	640	846	813	605	632	833	802

Table 7. **Summary of PAC parameter optimisation.**
Different parameters were tested for their impact on allelic quantification, including trimming of adaptors and low-quality nucleotides (TRIM), soft-clipping within STAR (SOFT), phasing using phASER with read-aware mode (PHASE) and rescuing multi-mapped reads (MULTIMAP). WASP is a common filtering tool that attempts to correct biases associated with ASE methodology. It does this by removing problematic reads (reads with a heterozygous site that when the genotype is flipped do not align to the same genomic location) that would otherwise contribute to a reference allele bias. WASP also incorporates other steps to improve ASE biases including correcting read depth and overdispersion statistically and choosing a duplicate read by random. WASP is introduced in more detail in section 3.1.1.2. The final version of PAC is highlighted in red. Figure is from Saukkonen *et al.*, 2022 [109].

2.3.3.2 PAC OUTLIERS

Although PAC dramatically improves the quantification of the RAR across large numbers of sites (when compared to the ground truth), there are still many sites where PAC fails to correctly account for mapping bias. There are 140 heterozygous sites where the allelic ratios differ from those in ground truth by 10%. None of these outlier sites were sites that GATK valiant calling falsely assigned (section 2.3.2.2).

To examine the outliers further, I annotated all heterozygous sites as the baseline (Figure 21 A) and the outliers with a 10% difference from ground truth (Figure 21 B), based on their location relative to different genomic elements with wANNOVAR (see methods 2.2.5). Most functional annotations in all variants and outlier variants are exonic or in the 3'UTR. The exonic noncoding RNA, which overlaps a transcript but does not have a gene definition, increased from 3% in all heterozygous sites to 10% in the outliers. These might be novel or rare variants, yet-to-be-identified gene isoforms or duplicated gene regions that lack comprehensive understanding.

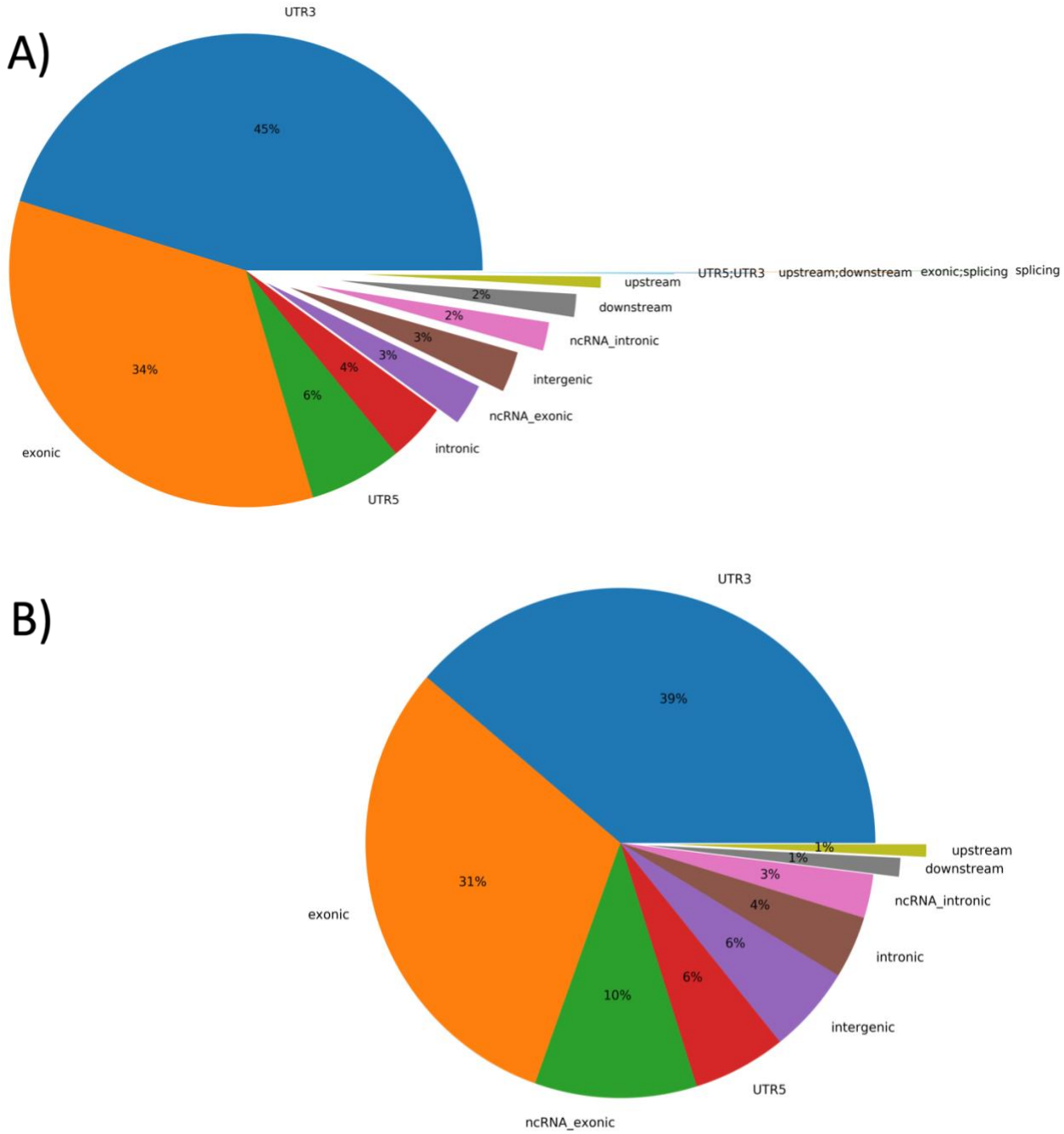


Figure 21**. The functional annotation of variants analysed by PAC**.
The heterozygous variants obtained from PAC (panel A) and those that differed by at least 10% in their reference allele ratio from the ground truth (panel B) were annotated with ANNOVAR, and their genomic locations are shown. The majority of variants are exonic and in the 3'UTR in all heterozygous sites and in outlier sites. Variants with at least 20× coverage in PAC are shown.
'exonic' = the variant overlaps a coding region (excluding 3/5'UTR regions). 'splicing' = the variant is within two basepairs of a splicing junction. 'ncRNA' = the variant overlaps a transcript that does not have a coding annotation in the gene definition (+'_intronic' = overlapping intron; +'_exonic' = overlapping coding region). 'UTR5' = the variant overlaps 5'UTR. 'UTR3' = the variant overlaps 3'UTR. 'UTR5;UTR3' = the variant overlaps both 5' and 3' UTRs (possibly on differen genes). Intronic = the variant overlaps an intron. Upstream = the variant overlaps 1 kb region upstream of transcription start site. Downstream = the variant overlaps 1 kb region downstream of transcription end site. Intergenic = the variant is in a intergenic region. exonic,splicing = the variant within exon but close to exon/intron boundary. upstream,downstream = the variant is located in both downstream and upstream regions (possibly on different genes). Label definitions taken from ANNOVAR documentation (https://annovar.openbioinformatics.org/en/latest/user-guide/gene/).

2.4 DISCUSSION

2.4.1 OTHER AVENUES TO IMPROVE ALLELIC QUANTIFICATION

ASE is a powerful tool with numerous utilities, which can improve eQTL signals [165] or help diagnose disease [168]. However, due to many computational biases and a lack of standardised analysis and detection pipelines, it has not been widely used. There are numerous computational approaches that have been developed to improve ASE detection, including filtering problematic areas [64, 142], utilising genomes that incorporate individual variants such as generating parental genomes [151, 217-219] and *de novo* assembly of the genome from the sequencing reads [220], and other computational methods [123, 124, 150]. None of these incorporates multiple strategies into a single pipeline, however. Determining the optimal combination of parameters can be a time-consuming process requiring expertise in bioinformatics knowledge. In this chapter, I developed a pipeline to tackle this gap in the research area. PAC deals with several of the well-known biases in aligning RNA-seq data and accurately quantifying the allelic reads.

2.4.2 PRELIMINARY PAC

At first, I tested the preliminary PAC on real data and showed that it improves biases associated with RNA-seq data analysis by reducing mapping bias and retaining more reads. PAC was able to detect sites under ASE and I implemented this on different cell-types within the same individual. During the preliminary PAC testing, there was still a reference bias seen when plotting the reference allele ratio. However, there has also been evidence that the reference allele is more ancestral and might be under stronger evolutionary pressure, and therefore some of the reference bias might be due to biological reasons. Nevertheless, this bias was greatly reduced when plotting the reference allele ratio in the ground truth data (Figure 20), highlighting that biases persist even in pipelines with multiple correction methods such as in the preliminary PAC approach. This demonstrates the importance of having realistic simulated data as ground truth, highlighting the space for refining PAC

further to improve the computational bias. This exhibits the benefit of realistic simulated data and that ASE studies are almost impossible to validate on real samples. Previous ASE studies use simulated genomic data [141, 157] to validate their results and performance. These simulations are usually only done on RNA-seq, however, which does not replicate realistic data as accurately.

2.4.3 SIMULATED GENOMICS DATA

Because it is impossible to remove all biases from real data to test the performance of ASE detection tools, I then developed highly accurate simulated genomic data. This allowed me to test the exact impact of different parameters of the PAC pipeline against ground truth.

I showed that PAC improved read alignment by retaining more reads and improved accuracy in measuring allelic ratios compared to the standard alignment. I showed that the remaining sites that showed a difference in the reference allele ratio from that in the ground truth have some differences in the genomic locations, with unannotated transcripts increasing in proportion for example.

With my simulation method, I show that GATK variant calling fails in certain regions. Sex chromosomes are expected to cause difficulties as they share a common origin and have repetitive sequences [221] and therefore they are usually discarded, as are HLA regions with their high polymorphism. However, even in other regions, variant calling errors are not random. Variant calling is more likely to assign heterozygotes as reference homozygotes rather than opposite [222, 223], which can affect downstream analysis [224]. I show that certain genomic regions along most chromosomes have peaks of false positive and false negative variant calling. I ran PAC on variant calls from GATK to replicate the ASE analysis in a realistic scenario. The downstream analysis of any genomic study involving genotypes will be affected by variant calling errors including generating personalised genomes.

2.4.4 PAC REFINEMENT

Using the simulated ground truth data, I tested multiple parameters and how they affect allelic quantification. Rescuing multi-mapping reads has been shown to improve ASE detection [152] previously, although no extensive studies have been done as these reads are typically discarded in most ASE analyses. I tested different versions of PAC with different parameters, and I show that deactivating trimming, including soft-clipping, incorporating phASER and re-allocating multi-mapping reads improve ASE detection. I included these parameters in the final PAC pipeline.

With the pipeline refinement, I show that each step makes a relatively small change. However, these might be important when considered together, especially in studies with small sample sizes and in rare disease fields where the population size is small and rare variants play a big role. It has been shown that rare and private variants can also have an effect on common diseases [225], however, the study of these is still ongoing due to difficulties of small sample sizes. Some studies have refined statistical analysis to try to accommodate this and have detected the widespread influence of rare variants on phenotypes [226-228] where many genes are regulated only by rare variants and are enriched for disease-linked genes [225]. Therefore, improving the detection of effects of rare variants on gene expression will play a crucial part in understanding disease mechanisms.

CHAPTER 3 – GENERATING USER-FRIENDLY PIPELINE AND APPLYING ON POPULATION DATA

3.1 INTRODUCTION

In the previous chapter, I developed PAC and showed its accuracy in reducing mapping bias by retaining more reads and identifying allelic counts at heterozygous sites more accurately. Chapter 2 overcomes many of the technical problems of ASE analysis. PAC involves multiple steps and several software. As with most genomics pipelines, implementing PAC and running it reproducibly requires a high level of computational skill. This provides a barrier to studying the regulation of gene expression in small sample size or at a rare variant level. To make PAC usable by the target audience, the implementation of the pipeline needs to be as simple and reproducible as possible. Therefore, the aim of this chapter is to convert PAC into a Nextflow package that runs with Docker/Singularity and is easily accessible from my GitHub page. Following this, I ran PAC on simulated genomic data from Chapter 2, and compared it to the most commonly used method for ASE detection, aligning RNA-seq reads by standard alignment to the reference genome. I also compared the performance to WASP, a commonly used filtering tool in the ASE field to reduce computational biases. I start the introduction by describing other methods used and then I describe the tools I used to streamline PAC.

3.1.1 ASE TOOLS PRIOR TO PAC

3.1.1.1 STANDARD ALIGNMENT

During a genomics analysis RNA-seq reads are aligned to the reference genome. This step introduces biases in ASE analysis for the reasons discussed in chapter 2. During ASE analysis, reads overlapping a heterozygous site are quantified. If the ratio significantly deviates from 1:1, it is assumed to be under ASE. However, alternative alleles carry a mismatch compared to the reference sequence, and as such reads containing these alleles are more likely to be discarded [229]. This leads to false ASE signals [141, 230]. There are methods that deal with these biases; however, their implication often requires more time-consuming and complicated analysis and therefore a simple standard alignment approach (for example, using data directly generated by STAR) is often preferred.

3.1.1.2 WASP

WASP is one of the most common filtering methods used in the ASE field. WASP aligns reads containing a heterozygous site [123] and then it filters out problematic reads by discarding those that do not align to the same location after the genotype within that read is swapped to that on the other genetic background (Figure 22). This overcomes the issue of alternative alleles more likely mapping to multiple locations by removing the reads that do so. The drawback from this is that a large number of potentially informative reads are being discarded, which can bias the expression level at genomic locations [123]. WASP incorporates other filtering steps including choosing a duplicate read by random (this avoids selecting the read with the highest score, which is most likely the reference allele) and correcting read depth and overdispersion statistically.

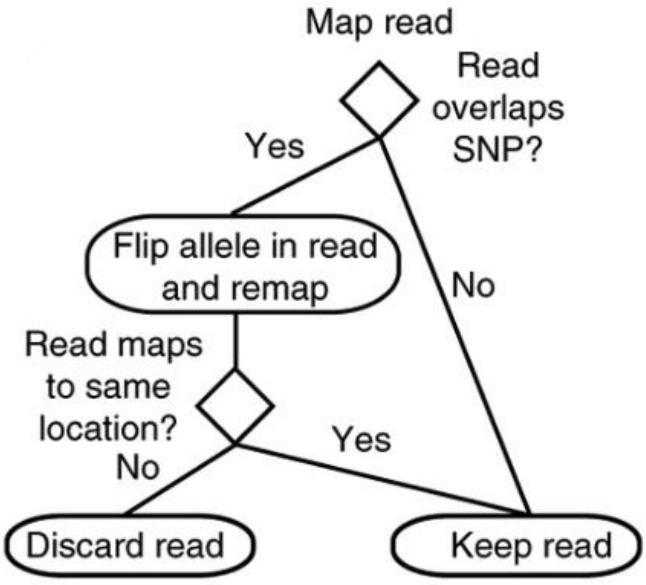


Figure 22. **Overview of WASP.**
WASP is a method that corrects biases associated with ASE. The main mechanisms by which it operates is by removing ambiguous reads. The chart describes how WASP makes a decision on whether to keep the read. From van de Geijn *et al.*, 2015 [123]. Reproduced with permission from Springer Nature, Copyright 2015.

3.1.2 GENOMICS PIPELINE INTO A STREAMLINED TOOL

Most genomic workflows rely on multiple external software and tools [231], which is also the case for PAC. Most of this software is developed in an academic setting and is not always straightforward to install and run. Each tool often functions slightly differently upon version updates, some of which might be incompatible with other tools within the workflow. And this is in addition to other dependencies and versions that software requires to run [232, 233]. For this reason, the exact result will be difficult to reproduce in different computational environments [234]. Consequently, the analysis often requires technical users or bioinformaticians [235]. A typical genomics workflow generates multiple intermediate files that will quickly take up a large amount of space [236], particularly when analysing multiple samples. For these reasons, there is a need for streamlined and easy-to-use software development that overcomes these issues [236].

3.1.2.1 DOCKER

Docker containers are computational environments that allow applications to run in the required environment with essential tools and dependencies [237]. To develop a Docker container, the developer builds an image that runs commands specified in order to first download and install all required software and dependencies. Containers are not reliant on the host computer's dependencies and therefore provide standardised platforms for ease and reproducibility. This allows the user to install a pre-built image that contains all the software and dependencies instead of downloading these separately and performing troubleshooting [232].

3.1.2.2 NEXTFLOW

Nextflow is a workflow management system that enables parallelisation and automation of computational pipelines. Nextflow has been developed for the bioinformatics field to solve the issues associated with reproducibility [236], most of which are associated with differences in computational platforms, the way intermediate files are handled, lack of good

practise, and the management of software and databases [236]. Nextflow has been extensively applied in the genomics field (eg.[238-242]). There is also a community-based effort, nf-core, that has collected best practise genomics analyses made available on Nextflow [243, 244].

Nextflow allows easy pipeline development that uses a Groovy-based domain-specific language. Nextflow can incorporate multiple scripting languages the pre-existing pipelines have been written in without needing to extensively modify the pipeline [245]. Nextflow supports Docker [236] and Singularity [246] technologies, which is beneficial in the genomics field where often multiple software and tools are incorporated. Nextflow consists of processes and pieces of workflow that can be executed independently. Each process communicates with each other through channels in the form of inputs and outputs. The output from one process feeds into the downstream process as an input. It allows parallel execution, error return and traceability. Nextflow can also be integrated into the GitHub software repository. There are also other workflow systems including Snakemake, however these are becoming less preferred due to lack of support in Docker and code sharing platforms such as GitHub [236].

3.1.2.3 GITHUB

GitHub is a popular development platform used for software building, maintaining and shipping in particular for open source projects [247]. The version control within GitHub allows groups working on the project to track and manage changes. It allows other users to access the code and use the software freely.

3.2 METHODS

3.2.1 SIMULATED GENOMIC DATA

The simulated data used in this chapter was generated and described in Chapter 2 (see section 2.2.2). Briefly, I used a GRCh37 reference genome and high confidence variant calls for an individual NA12877 from CEPH/Utah pedigree 1463 from the PGP project [205]. I incorporated the phased variants into the reference genome to generate maternal and paternal genomes. I then used these parental genomes to stimulate WGS, from which I did GATK variants calling. I also simulated RNA-seq from the parental genomes. I used these simulated RNA-seq to generate allelic counts at heterozygous sites, which acted as 'ground truth allelic counts'. The simulated genomic data was supplied into the PAC to compare the results against the ground truth data.

3.2.2 STANDARD ALIGNMENT OF SIMULATED RNA-SEQ READS

To obtain baseline allelic counts against which to compare other methods, I first aligned the simulated RNA-seq paired-end reads to a 1000 genomes version of the GRCh37 reference genome with STAR 2.51a with default parameters. This included soft-clipping, using two-pass alignment, GENCODE gene annotation (version 19), and allowing 8 mismatches per read pair, before keeping only properly paired (-f 0x0002 using SAMtools) and uniquely mapped (NH:i:1 flag) reads.

3.2.3 WASP

In order to compare the performance of WASP against standard alignment, I generated WASP-filtered alignment data. For this, I first generated WASP-filtered data by following the same approach detailed above as that for standard alignment, but with additional flag --waspOutputMode SAMtag within STAR (v2.7.3a). The VCF file generated from GATK (as described in Chapter 2) was supplied as an input.

--waspOutputMode SAMtag incorporates the WASP-filtering into the alignment by flipping the alleles, and should the flipped allele not align, or align to multiple/different locations it discards the read. I filtered the resulting BAM file for reads that were properly paired, reads without a WASP flag (and thus do not contain a genetic variant) and reads that pass WASP-filtering (with flag 'vW:i:1'), before counting reference and alternative allele coverage at the heterozygous sites.

3.2.4 EVALUATING THE ACCURACY OF ALLELE COUNTS AND THE OUTLIER ANALYSIS

To obtain allelic counts from the PAC pipeline, standard alignment, and WASP-filtered alignment and evaluate their performance, I compared results obtained with each method to the ground truth data (calculated in the previous chapter, see section 2.2.3). I excluded sites located in the blacklisted genomic regions (obtained from ENCODE ENCFF001TDO.bed) and HLA region (obtained from phASER). For the analysis, I only considered heterozygous sites with at least 20× coverage that were present in all 3 methods and the ground truth data, unless otherwise stated.

The performance of methods was compared by considering a number of heterozygous sites (obtained with SAMtools mpileup using default parameters and disabling read-pair overlap detection); the correlation, measured with R^2, between a method and ground truth reference allele ratios; the number of additional heterozygous sites aligned by a method but missed by standard alignment; the number of sites where the reference allele ratio in a method differs from the ground truth reference allele ratio by more than 10% or 20% (from hereafter referred to as 'outliers').

3.2.5 ACCURACY OF ANALYSIS NEAR INDELS AND OTHER VARIANTS

In order to investigate how PAC performs in genomic regions that are known to be difficult to align, I compared the difference in reference allele ratio in PAC, standard alignment and

WASP-filtered alignment against the ground truth reference allele ratio at heterozygous sites in regions containing other variants or indels. For indel analysis, I selected heterozygous sites that were within 500bp of an indel (with a minimum indel length of 6bp). I also investigated heterozygous sites that had another heterozygous single nucleotide variant or rare variant (MAF < 1%) within 25bp of the heterozygous site under investigation. I used CEU population data from 1000 genomes project for allele frequency information. Mann-Whitney tests were performed with Bonferroni-adjustment to correct for multiple testing.

3.3 STREAMLINING PAC

In order to streamline PAC, I wrote it in Nextflow and published the code on my Github page. In this section, I describe this process. The code underwent extensive troubleshooting and testing to ensure its correct function. I start by describing the dependencies for Nextflow, each step within Nextflow and the final tool on GitHub.

3.3.1 DEPENDENCIES

3.3.1.1 CONFIGURATION

The Nextflow configuration file defines parameters for the tool, including where to obtain the Dockerfile, number of threads, output directory, where to obtain files needed for PAC to run. These can be customised or left as default parameters.

PAC uses AWS S3 bucket supported by Nextflow, which contains Illumina iGenomes [248]. The reference genome (available GRCh37 and GRCh38) and annotation GTF files were obtained from AWS S3 bucket. The BED file was uploaded to GitHub obtained from GENCODE.

3.3.1.2 DOCKER

In order to run PAC, several tools and software are required. PAC requires specific versions of each of these and downloading them independently is not straightforward. I created a Docker container image that contains all these tools and software, which are executed at the start of a PAC run. The Dockerfile shows which commands are run (Figure 23). In order to run with Docker, option -profile docker needs to be selected.

```
# Set the base image
FROM centos

#Load dependencies

RUN dnf install -y redhat-rpm-config

RUN yum install -y \
git \
python2 \
wget \
epel-release \
python2-pip \
gcc \
python2-devel \
make \
zlib-devel \
gcc-c++ \
bzip2 \
bzip2-devel \
ncurses-devel \
xz-devel \
perl-Env \
java-devel


RUN yum install -y which

#Get STAR
RUN wget https://github.com/alexdobin/STAR/archive/2.7.4a.tar.gz
RUN tar -xzvf 2.7.4a.tar.gz
WORKDIR /STAR-2.7.4a/source
RUN make STAR
WORKDIR /
ENV PATH="/STAR-2.7.4a/bin/Linux_x86_64_static:${PATH}"
RUN STAR

#Get Alleleseq:
RUN wget http://alleleseq.gersteinlab.org/vcf2diploid_v0.2.6a.zip && unzip vcf2diploid_v0.2.6a.zip
WORKDIR vcf2diploid_v0.2.6a
RUN make
WORKDIR /

#Get Samtools:
RUN wget https://github.com/samtools/samtools/releases/download/1.10/samtools-1.10.tar.bz2
RUN bzip2 -d samtools-1.10.tar.bz2
RUN tar -xvf samtools-1.10.tar
RUN echo $(ls)
WORKDIR samtools-1.10
RUN ./configure --prefix=/usr
RUN make
RUN make install
WORKDIR /usr/bin
RUN echo $(ls)
WORKDIR /

#Liftover:
RUN wget http://hgdownload.cse.ucsc.edu/admin/exe/linux.x86_64/liftOver
RUN echo $(ls)
RUN chmod +x liftOver && mv liftOver /usr/bin

#RSEM
RUN git clone https://github.com/deweylab/RSEM.git
WORKDIR RSEM
RUN make
WORKDIR /
ENV PATH="/RSEM:${PATH}"

#Get HTSLIB:
RUN wget https://github.com/samtools/htslib/releases/download/1.10.2/htslib-1.10.2.tar.bz2
RUN bzip2 -d htslib-1.10.2.tar.bz2
RUN tar -xvf htslib-1.10.2.tar
RUN echo $(ls)
WORKDIR htslib-1.10.2
RUN ./configure --prefix=/usr
RUN make
RUN make install
WORKDIR /usr/bin
RUN echo $(ls)
WORKDIR /

#Get Bedtools:
RUN ln -snf python2.7 /usr/bin/python
RUN wget https://github.com/arq5x/bedtools2/releases/download/v2.29.2/bedtools-2.29.2.tar.gz
RUN tar -zxvf bedtools-2.29.2.tar.gz
WORKDIR bedtools2
RUN make
RUN cp bin/* /usr/local/bin/
WORKDIR /

#Get BCFTOOLS:
RUN wget https://github.com/samtools/bcftools/releases/download/1.11/bcftools-1.11.tar.bz2
RUN bzip2 -d bcftools-1.11.tar.bz2
RUN tar -xvf bcftools-1.11.tar
RUN echo $(ls)
WORKDIR bcftools-1.11
RUN ./configure --prefix=/usr
RUN make
RUN make install
WORKDIR /usr/bin
RUN echo $(ls)
WORKDIR /

#Get phaser:
RUN git clone https://github.com/secastel/phaser.git
RUN pip2 install Cython
RUN pip2 install scipy
RUN pip2 install pysam
RUN pip2 install pandas
RUN pip2 install intervaltree
WORKDIR phaser/phaser
RUN python2 setup.py build_ext --inplace
WORKDIR /
```

Figure 23. **Dockerfile for PAC.**
The file shows numerous genomics tools and softwares that are required for PAC.

3.3.1.3 SINGULARITY

Singularity is an alternative to Docker, which is often preinstalled on High Performance Compute clusters, and is therefore preferred for ease of access. Nextflow automatically tries to pull an image with the specified name in the configuration file from the Docker Hub. PAC needs to be run with -profile singularity option for this.

3.3.2 PAC INTO NEXTFLOW

Nextflow consists of processes that contain a task within the workflow that can be executed independently. The outputs from one process are distributed to other processes where they act as inputs. This allows parallelisation. As soon as the input files become available for a particular process, it will be executed. The processes can be written in Linux executable languages including Bash, Python or Perl. This section will describe each process within PAC.

Figure 24 shows the processes within PAC and where the output from each process feeds into. It demonstrates the interconnectivity and benefit of parallelisation, which speed ups the run time.

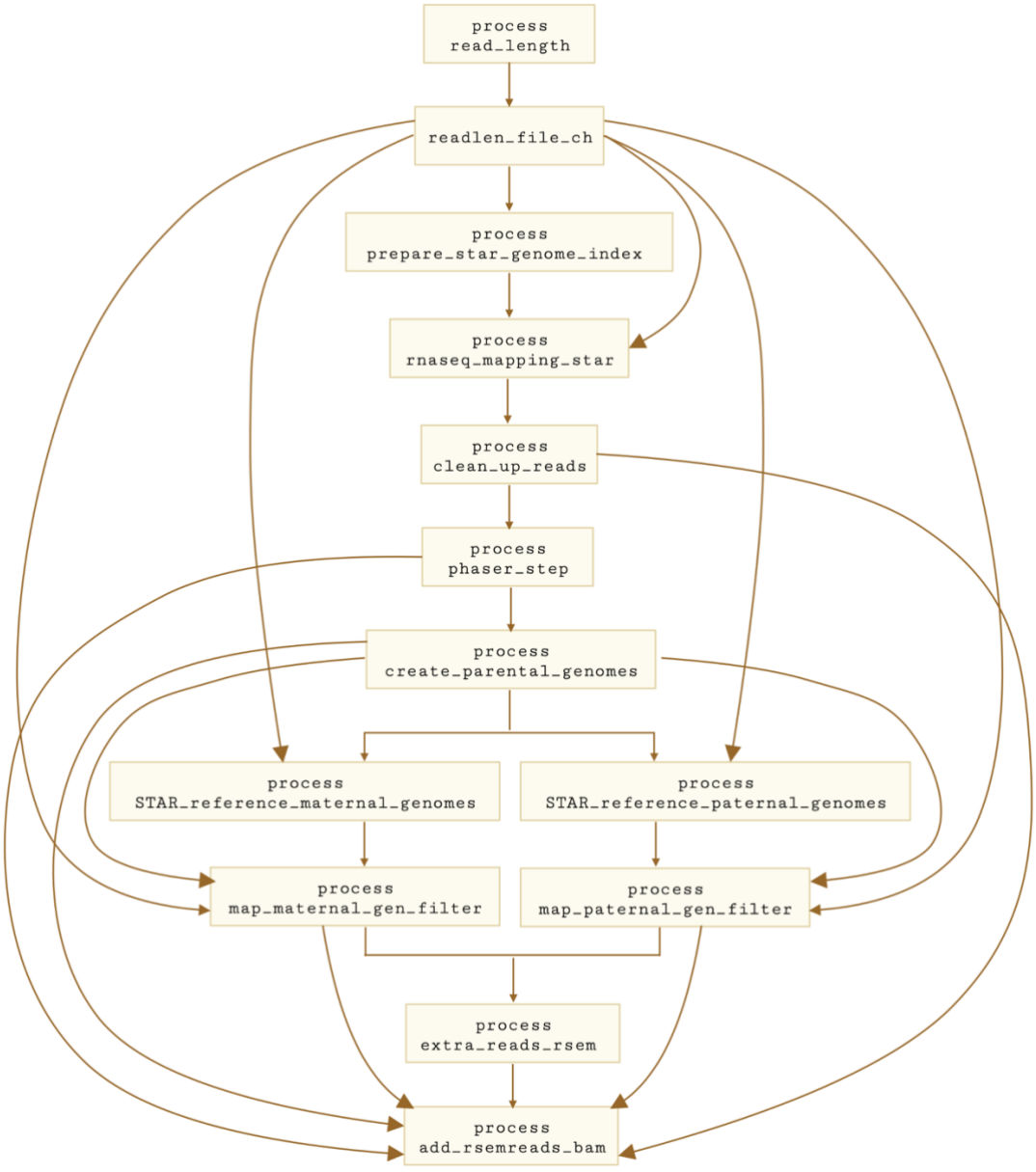


Figure 24. **The overview of PAC processes within Nextflow**.
Shown are the names of processes and where the inputs and outputs are derived from. Only the process names are shown to simplify the figure. The parameter settings section is omitted. Created with BioRender.com.

The following sections describe each process within PAC at a high level. See appendix 1 for a reference manual for a more detailed information.

3.3.2.1 setting parameters

The first step, although not a process, checks that all essential parameters are specified when executing PAC. The essential parameters are the genome version, path to RNA-seq reads, path to variants VCF file, and sample ID. If any of these are missing, PAC stops the run and gives an error message stating which parameter is missing. This section also places RNA-seq reads into multiple channels as multiple processes take them as inputs.

3.3.2.2 process read_length

Input: This process takes in RNA-seq read files as input file.

Process: Custom bash script calculates the read length.

Output: The output is a file with read length value that is used in the downstream processes throughout PAC.

Outside of the process the value from the output file is placed into different channels as multiple processes need this value.

3.3.2.3 process prepare_star_genome_index

Input: This process takes in the reference genome specified in options, annotation file, read length information from the previous process, and number of cpus as an optional input.

Process: It then generates a genome index with STAR --runMode genomeGenerate.

Output: The genome indices in STARhaploid directory. This step is necessary for standard alignment in the next process.

3.3.2.4 process rnaseq_mapping_star

Input: This process takes in the reference genome, genome index generated from process prepare_star_genome_index, read length information from process read_length, the RNA-seq reads, sample ID and number of cpus as an optional input.

Process: The step aligns reads to the reference genome and indexes the BAM file with SAMtools index. This process provides the standard alignment that the user can use as a comparison for the PAC results. The output also feeds into the phaser_step.

Output: BAM and BAM.bai files of mapped RNA-seq reads.

3.3.2.5 process clean_up_reads

Input: The process takes in the BAM files generated from process rnaseq_mapping_star, variants VCF file, sample ID and number of cpus as an optional input.

Process: In this step the mapped RNA-seq reads are filtered. SAMtools is used to keep only properly paired (where the read orientation of read pairs is as expected and the gap between them is likely based on sequencing technology) and uniquely mapped (reads mapping to single location) reads. The BAM is then created that is compatible for downstream process phaser_step.

Output: Properly paired and uniquely mapped BAM file, phaser_step process compatible BAM and BAI files in separate channels.

3.3.2.6 process phaser_step

Input: Variants VCF file, BAM and BAI files from process clean_up_reads, sample ID and number of cpus.

Process: This step uses phASER to phase variants incorporating aligned RNA-seq reads. phASER uses a read-aware mode for phasing. It selects RNA-seq reads where there are two variants, that can be split across larger genomic distances due to splicing, hence it can incorporate variants over longer distances and thereby improve phasing. This allows better phasing of rare variants and longer haplotypes.

Output: Phased variants VCF file.

3.3.2.7 process create_parental_genomes

Input: The reference genome, annotation file, phased variants VCF file from process phaser_step, sample ID and BED annotation file.

Process: This step creates personalised parental genomes. The phased variants are incorporated into the reference genome using vcf2diploid, generating maternal and paternal genomes. liftOver is then used to generate GTF and BED files with adjusted genomic coordinates for maternal and paternal genomes. This is because the coordinates will be shifted due to indels present in the VCF file. The custom scripts generate maternal and paternal VCF files where the heterozygous site coordinates are shifted to the maternal and paternal genomes.

Output: Maternal and paternal genomes, chain files for both genomes that are needed for liftOver (not needed in the downsteam process but output ensures files can be found on users' system should they need them for their own analysis), maternal and paternal GTF and BED files, files containing regions not lifted for maternal and paternal genomes, maternal and paternal VCF files.

3.3.2.8 process STAR_reference_maternal_genomes

Input: Maternal genome and maternal GTF file from process create_parental_genomes, read length information from process read_length, sample ID and number of cpus.

Process: This step generates maternal genome index with STAR --runMode genomeGenerate. This step feeds into map_maternal_gen_filter, where the RNA-seq reads are mapped to the maternal genomes.

Output: Maternal genome indices in Maternal_STAR directory.

3.3.2.9 process STAR_reference_paternal_genomes

This process is identical to process STAR_reference_maternal_genomes above but it is performed on the paternal genome.

3.3.2.10 process map_paternal_gen_filter

Input: Paternal genome indices from process STAR_reference_paternal_genomes, RNA-seq reads, paternal genome and GTF file from process create_parental_genomes, read length information from process read_length, sample ID and number of cpus.

Process: In this step the RNA-seq reads are aligned to the paternal genome with STAR. The BAM file generated from this is indexed and filtered with SAMtools to keep only properly paired and uniquely mapped reads.

RSEM is used to index the paternal genome. Following this, RSEM is used with the STAR transcriptome.bam to map the same RNA-seq reads with RSEM instead. In this case, reads that would map to multiple locations are not discarded but are allocated one location. All uniquely mapped reads are used to calculate the expression of each of these loci, and then

the multi-mapping reads are allocated a location based on these weights. The allocation is based on probabilities based on ratios of uniquely mapped reads from genomic loci where the multi-mapping read aligns to. The file is then filtered with SAMtools to keep only properly paired reads.

Output: BAM file of mapped reads to paternal genome and BAM file generated with RSEM.

3.3.2.11 process map_maternal_gen_filter

This process is identical to process map_paternal_gen_filter but performed on the maternal genome.

3.3.2.12 process extra_reads_rsem

Input: Filtered BAM file from process map_maternal_gen_filter and map_paternal_gen_filter, RSEM sampled BAM files from map_maternal_gen_filter and map_paternal_gen_filter, and sample ID.

Process: Custom script gets the extra multi-mapping reads (which now only have one location allocated by weight in the previous step) that are aligned in RSEM, but not in STAR and creates a file extra.rsem.maternal/paternal.txt. Then a new RSEM BAM file is created containing only these extra reads.

Output: BAM file for maternal and paternal extra reads that originally aligned to multiple locations, now with a single location.

3.3.2.13 process add_rsemreads_bam

Input: Maternal and paternal extra reads from RSEM generated in process extra_reads_rsem; BAM file of reads mapped to maternal and paternal genomes from map_maternal_gen_filter and map_paternal_gen_filter; map_over, and and maternal and paternal bed files with adjusted coordinates, and maternal and paternal phased VCF file from process create_parental_genomes, phased VCF file from process phaser_step, sample ID, number of cpus, properly paired and uniquely mapped reads to the reference genome from process clean_up_reads; and GENCODE BED file.

Process: For each parental genome, the STAR and RSEM BAM files are merged. Then PAC finds reads only aligned in one parent and not the other. When the reads are aligned in both maternal and paternal genomes, a custom script (filter_2genomes.pl) selects the best alignment for each read from the two alignments (scoring reads by the number of matching nucleotides minus two times the number of indel positions, drawing at random when the two alignments have equal scores).

Then two custom scripts (compare_basic_map.pl and compare_2genomes.pl) are used to count the number of alleles at each heterozygous site. Initially, this is done with standard alignment. Then the same is performed for two genomes parental alignment using the liftOver variant files.

Then phASER is used to generate the gene-level calculations using the VCF files and GTF files from each parent (generated in process create_parental_genomes). PAC then produces allele counts at haplotypic level using phASER Gene AE.

Finally, the last custom script (merge_gene_level.pl) merges the gene level counts across the two parents.

Output: The results files: site and haplotype level allelic counts and single genome alignment for comparison.

3.3.3 PAC METRICS

I run PAC with 5 frontal cortex GTEx RNA-seq samples with an average depth of ~44 million paired reads (see Table 11, alignment described in section 4.2.2) to obtain average metrics for PAC. The average metrics for each process are shown in Table 8. PAC is written in Nextflow, which allows parallelisation. This means that as soon as the input files for each process become available, the process will start. STAR_reference_paternal_genomes and STAR_reference_maternal_genomes processes start when create_parental_genomes process has finished. Equally, when STAR_reference_paternal_genomes has finished, map_paternal_gen_filter will start. And same applies to the maternal genome. When map_paternal_gen_filter and map_maternal_gen_filter have finished, extra_reads_rsem can proceed. This speeds up the run time as multiple processes can run simultaneously.

PAC takes an average of 12 h 6 minutes to generate site- and gene-level ASE data per sample, whereas generating these data from standard alignment takes an average of 3 h and 28 min on the same computational setup (requested 10 CPUs, 512000MB memory). Although PAC requires a longer run time (almost 3.5 times longer), it provides extra information and accuracy and is scalable for smaller-scale studies where ASE information is particularly useful. The improved value of using PAC in any given case is impossible to quantify as it will depend on the sample and whether the additional allelic information falls within genes that are of biological interest. However, PAC provides an option for users to obtain additional and more accurate information from their samples without the need for additional experiments/samples.

process	% time
read_length	0.01
prepare_star_genome_index	3.18
rnaseq_mapping_star	2.22
clean_up_reads	4.33
phaser_step	20.39
create_parental_genomes	14.40
STAR_reference_maternal_genomes	3.13
STAR_reference_paternal_genomes	3.11
map_maternal_gen_filter	13.08
map_paternal_gen_filter	13.32
extra_reads_rsem	2.29
add_rsemreads_bam	20.55

Table 8. **Average metrics for each process within PAC.**
The average metrics for each process within PAC when it was run with 5 test GTEx frontal cortex samples with an average depth of ~44 million paired reads. % time represents the percentage of the duration of each process relative to the overall duration.

3.3.4 PAC ON GITHUB

In order to make PAC publicly available and easy for distribution, I published it on GitHub (Figure 25). The front page with README.md contains information about PAC, how to run it and what options users can select.

main.nf is the PAC Nextflow script, each of the main processes described in the previous section.

Dockerfile is available for users' knowledge. The file is not essential as it is pulled from my Docker Hub when PAC is executed.

nextflow.config includes configurations essential for PAC run. It includes default options that can be customised by specifying each separately.

conf directory includes information where PAC can obtain reference genome and the annotation files.

bed directory contains BED annotation files from GENCODE.

bin directory contains scripts used within PAC.

article_data contains information where to obtain data used to generate the simulated genomic data from Chapter 2.

test directory that contains downsampled RNA-seq reads and VCF file from simulated genomic data (Chapter 2) that can be used to test PAC. It also contains a directory of results files generated when running PAC with these test files.

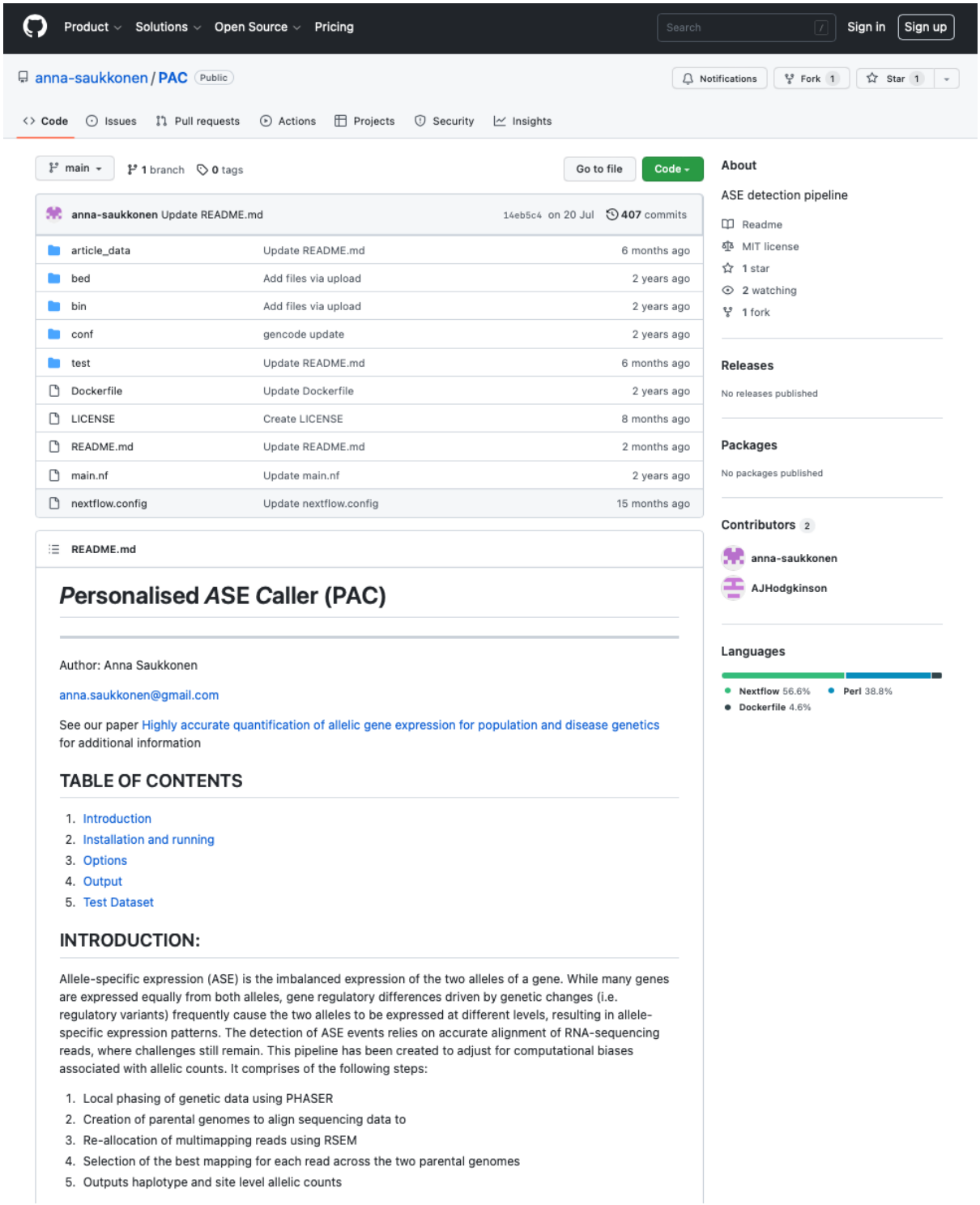


Figure 25. **PAC GitHub website.**

3.3.5 PAC USER MANUAL

NAME

PAC – accurate allelic quantification at site and haplotype level

SYNOPSIS

```
nextflow run PAC/main.nf [options] --genome_version <genome
version> –reads <path to reads> –variants <path to variants> –
id <id> -profile <profile>
```

DESCRIPTION

Allele-specific expression (ASE) is the imbalanced expression of the two alleles of a gene. While many genes are expressed equally from both alleles, gene regulatory differences driven by genetic changes (i.e. regulatory variants) frequently cause the two alleles to be expressed at different levels, resulting in allele-specific expression patterns. The detection of ASE events relies on accurate alignment of RNA-sequencing reads, where challenges still remain. This pipeline has been created to adjust for computational biases associated with allelic counts. It comprises of the following steps:

1. Local phasing of genetic data using PHASER
2. Creation of parental genomes to align sequencing data to
3. Re-allocation of multimapping reads using RSEM
4. Selection of the best mapping for each read across the two parental genomes
5. Outputs haplotype and site level allelic counts

To run a test sample, run following commands:

```
load java
load singularity
git clone https://github.com/anna-saukkonen/PAC.git
```

```
path_to_nextflow/nextflow run PAC/main.nf --
genome_version GRCh37 --reads
"PAC/test/NA12890_merged_sample_0.005_{1,2}.fq.gz" --
variants
"PAC/test/NA12877_output.phased.downsampled.vcf.gz" --id
NA12877 -profile singularity
```

OPTIONS

Required:

--genome_version <genome version>

The available genomes are: GRCh37 or GRCh38.

--reads <path to reads>

The path to reads in within quotation marks. The reads need to be in the same directory with the following format: path_to_read_1.fq.gz and path_to_read_2.fq.gz. The options is called with: "path_to_reads_{1,2}.fq.gz".

--variants <path to variants>

The path to phased VCF file within quotation marks.

--id <id>

The sample ID.

-profile <profile>

The available options are: docker or singularity.

Optional:

-cpus

The number of cpus (as an integer). The default is 10.

-outdir

The name of the output file directory. The default is "/pac_results".

-N

An email address should the user want an email notification when the run is finished.

OUTPUT

PAC generates 5 output files:

haplotype level ASE calls:

1. 'id'_gene_level_ae.txt

single nucleotide level ASE calls from PAC:

2. results_2genomes_'id'.RSEM.STAR.SOFT.NOTRIM_baq.txt
3. results_2genomes_'id'.RSEM.STAR.SOFT.NOTRIM.txt

single nucleotide level ASE calls based on standard single genome mapping for comparison:

4. results_1genome_'id'.SOFT.NOTRIM_baq.txt
5. results_1genome_'id'.SOFT.NOTRIM.txt

PREREQUISITE

Nextflow

The Nextflow can be downloaded with following command:

```
curl -fsSL get.nextflow.io | bash
```

Java

Java version 8 and above. You can check your java version with following command:

```
java -version
```

Docker or Singularity

The user needs a docker or singularity installed depending on which profile they use.

3.4 RESULTS

PAC was optimised (Chapter 2) to deal with many of the technical problems associated with ASE analysis. In this chapter, I have written PAC into an easy-to-use format with Nextflow utilising Docker to remove the burden of having to install tools and dependencies.

I next compared the performance of PAC against other commonly used methods, including standard alignment and WASP-filtering.

3.4.1 VALIDATING PERFORMANCE OF PAC

3.4.1.1 COMPARING PAC TO OTHER METHODS

In chapter 2 I simulated RNA-seq data from each of the parental gold standard genomes inherited by the offspring. I merged these simulated RNA-seq reads and calculated the ground truth allelic counts at each of the heterozygous variant positions that were identified in the GATK variant calling that was generated from the simulated WGS from the parental gold standard genomes. The accuracy of allelic counts obtained from different PAC parameters when refining PAC was compared to this ground truth allelic data. In this chapter, I validate PAC as a method and compare its performance against other methods, using ground truth allelic counts as the baseline again.

I aligned the simulated RNA-seq reads to the reference genome using STAR as a standard alignment approach as this is one of the most common alignment methods. I then aligned the same simulated RNA-seq reads with STAR using WASP-filtering. WASP is described in section 3.1.1.2. Briefly, it attempts to flip a genotype within a read to that of the other allele, and if it does not align to the same genomic location, the read is removed. This has been shown to improve mapping bias but also remove a relatively large number of reads [123]. Finally, I supplied PAC with the same simulated RNA-seq reads.

For an output from each of the methods, I counted the RAR at the heterozygous sites, and compared how closely the results replicated ground truth data (Figure 26). I only considered

heterozygous sites that had at least 20× coverage in the ground truth data and all three methods to make results directly comparable. After this filtering step, there were 11,602 heterozygous sites (Table 9), compared to 13,211 heterozygous sites at >=20× coverage in the simulated RNA-seq data.

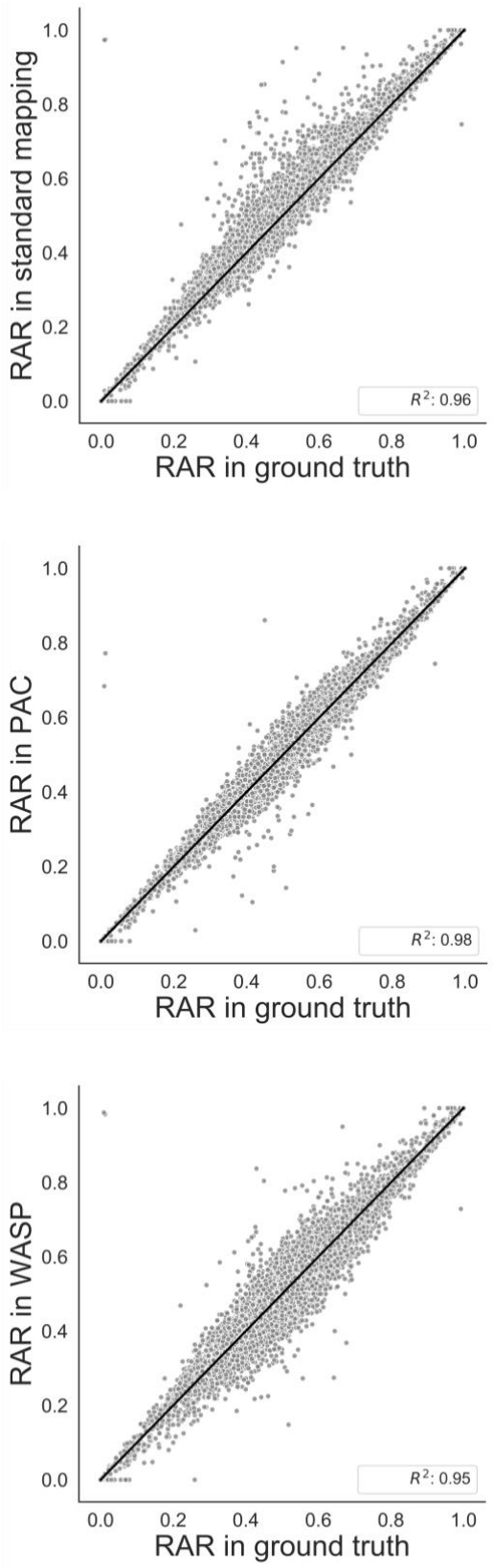


Figure 26. **Correlation of reference allele ratios (RAR) between standard alignment, WASP-filtered alignment, and PAC with the ground truth data.**
PAC shows the strongest correlation with the ground truth data, followed by WASP-filtered alignment and the poorest correlation with standard alignment. Genome-wide Pearson correlation coefficients (R^2) are shown ($P<0.05$ for all comparisons).

	Difference in reference allele ratio	R^2 between ground truth	Outliers >20%	Outliers >10%	Gained sites against standard alignment
Standard alignment	Mean: 0.0321 Median: 0.0258	0.9599	55	305	N/A
WASP	Mean: 0.0361 Median: 0.0304	0.9455	32	387	0
PAC	Mean: 0.0233 Median: 0.0192	0.9757	13	62	350

Table 9. S**ummary statistics for the different analysis methods for heterozygous sites in standard alignment, WASP-filtered alignment and PAC.**
Statistics are shown for sites with at least 20× coverage in all three methods.

When considering only heterozygous sites present in standard alignment and ground truth data at >=20× coverage, standard alignment was able to detect 12,109 sites out of 13,211 sites present in the ground truth (~91.7% of heterozygous sites). The average coverage at heterozygous sites dropped by 31× from ~175× in the ground truth data to ~144× in the standard alignment. This demonstrates the large number of reads are lost with standard alignment, many of which tend to be those with alternative alleles due to mapping bias. RAR were highly correlated between standard alignment and the ground truth data at heterozygous sites (R^2=0.960) (Figure 26). However, there were 305 heterozygous sites whose absolute difference in RAR was greater than 10% and 55 sites with a difference greater than 20%. The absolute mean difference shows a 3.21% bias across all heterozygous sites (Table 9).

Then, when I supplied the simulated RNA-seq data to PAC, the number of reads and the accuracy of allelic ratio quantification was significantly improved compared to standard alignment. When I only considered heterozygous sites present in PAC and ground truth data at >=20× coverage, the number of heterozygous sites increased by 339 compared to the standard alignment to 12,448. The average coverage was ~150× at these sites, an increase of 6× compared to the standard alignment. When considering the correlation of RAR in PAC

compared to the ground truth data, it increased to R^2=0.976 (Figure 26). The number of heterozygous sites with an absolute difference in RAR of 10% also dramatically decreased to 62, and of 20% difference it decreased to 13 sites. The mean difference from ground truth RAR is 2.33% (Table 9), which is significantly lower than that found for standard alignment at the same sites (one-sided t-test, $P=2.6\times10^{-125}$). These results show that PAC retains more reads and assigns them significantly better than alignment to the reference genome.

When I considered heterozygous sites present in the WASP-filtered alignment and the ground truth at >=20× coverage, I found that the number of sites data dropped by 836 when compared to PAC and by 497 when compared to standard alignment, down to 11,612. The average coverage at these sites was 135×, which was 9× fewer than in standard alignment and 15× fewer than in PAC. This reduction in the number of reads and coverage is likely due to WASP removing difficult-to-align reads from the analysis, and as a consequence removing potentially informative reads. The number of outliers with an absolute difference of 20% was reduced to 32 using WASP-filtered data, and the number of heterozygous sites with an absolute difference in RAR of 10% increased by 82 to 387 relative to the standard alignment. Surprisingly, the R^2 value decreased relative to PAC and standard alignment to 0.946 (Figure 26). Likewise, the mean absolute difference between WASP-filtered data and the ground truth of 3.61% is significantly higher than in standard alignment ($P=4.5\times10^{-21}$, one-sided t-test) and in PAC ($P=8.9\times10^{-272}$, one-sided t-test) (Table 9).

I then examined the RAR ratio distribution of heterozygous sites obtained from PAC, standard alignment and WASP-filtered alignment against ground truth RAR in sites that had at least 20× coverage (Figure 27). These results demonstrate that as expected, standard alignment exhibits reference allele bias with a higher proportion of reads with RAR >0.5 than in ground truth distribution. WASP-filtered alignment overcorrects the RAR relative to the ground truth distribution, with a higher proportion of reads with RAR 0.4-0.55 than in ground truth. There is also a smaller proportion of reads with RAR 0.35-0.4 and 0.6-0.75 than in ground truth. PAC also exhibits slight overcorrections similar to WASP-filtered alignment, however, this overcorrection is smaller. Overall, the RAR distribution in PAC is closest to the RAR in ground truth.

On a more granular level, there were 68,985,120 read pairs in the original raw data. PAC aligns 2,402,407 read pairs that are not aligned by either standard alignment or after WASP-filtering, of which 84,045 reads align across a heterozygous genetic variant. When only considering these reads spanning a heterozygous site, PAC places the read at the exact location correctly on the reference genome 86.3% of the time. This demonstrates that the vast majority of additional reads aligned by PAC are accurate. Additional reads that are aligned by PAC have similar GC content to reads aligned by standard alignment (47.8% in PAC compared to 49.1% in standard alignment) and do not bias towards any particular chromosome. 68 genes show a two-fold difference in either direction in the number of reads aligned by PAC relative to the standard alignment; these genes are not enriched for any particular GO functional terms.

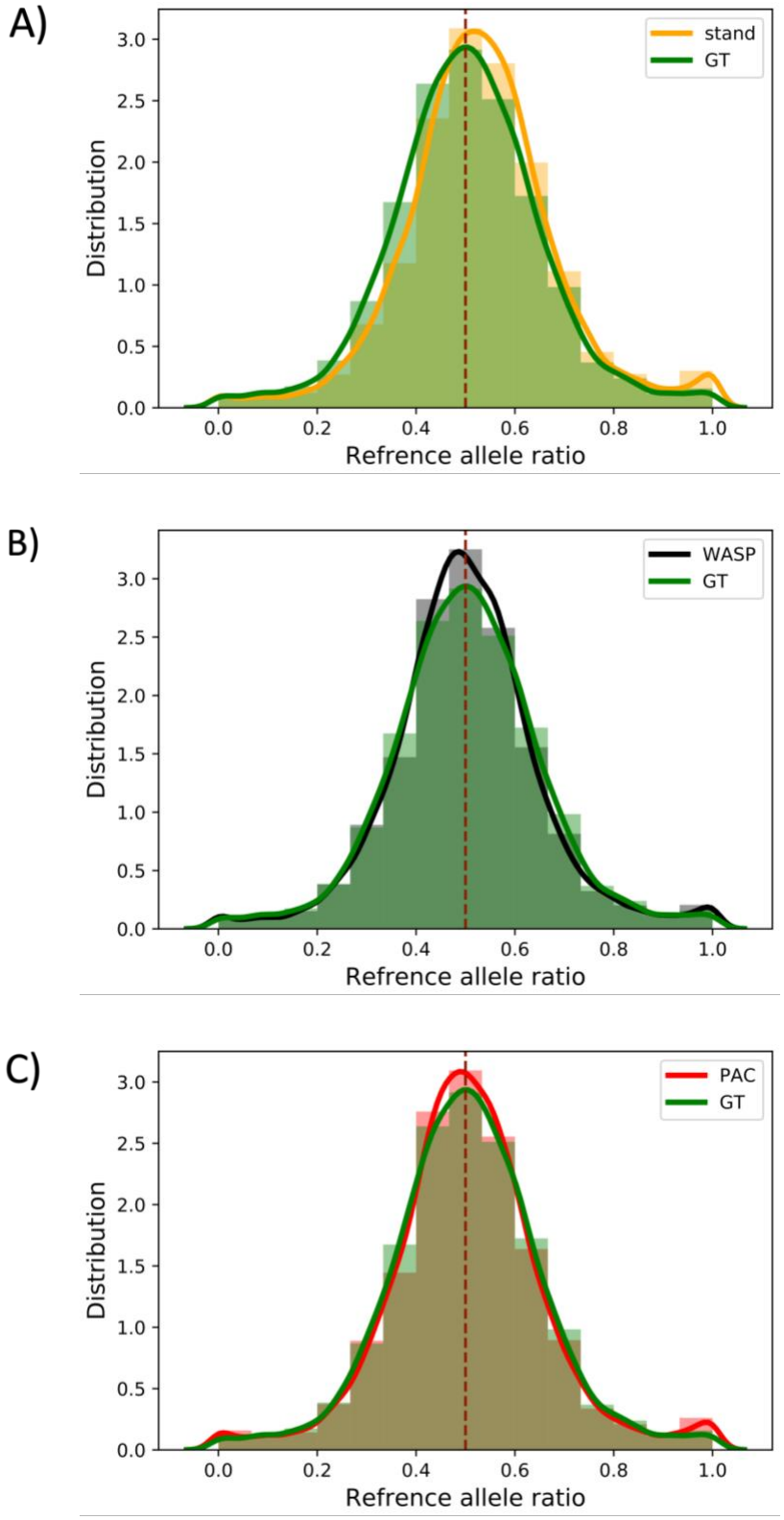


Figure 27. **The RAR in standard alignment, WASP-filtered alignment and PAC versus the ground truth.**
The distribution of RAR in heterozygous sites with at least 20× coverage in standard alignment (stand) versus ground truth (GT) (panel A), WASP-filtered alignment versus ground truth (panel B), and PAC versus standard alignment (panel C). Sites are not shared between methods. The lines represent kernel density estimates for each method.

3.4.1.2 ADDITIONAL READS PICKED UP BY PAC

Next, I set to determine if additional reads aligned by PAC were accurate or noise that WASP-filtering removes. When I only considered heterozygous sites that had >=20× coverage in PAC but were missed by standard alignment and WASP-filtering, either by not meeting this threshold or due to being filtered out during the alignment step, applying PAC resulted in an additional 350 heterozygous sites. The RAR for these sites were highly correlated between PAC and the ground truth data (R^2=0.844). Similarly, when I considered heterozygous sites that were detected in standard alignment and PAC (but not WASP-filtered data), there were an additional 496 sites with highly significant RAR when compared to ground truth data (R^2=0.956, P=2.6×10^{-266}, Figure 28). This demonstrates that PAC performs well at sites with lower coverage that may be missed by other approaches. Therefore, with the aim of improving computational biases, WASP-filtered alignment also removes reads that could be potentially informative.

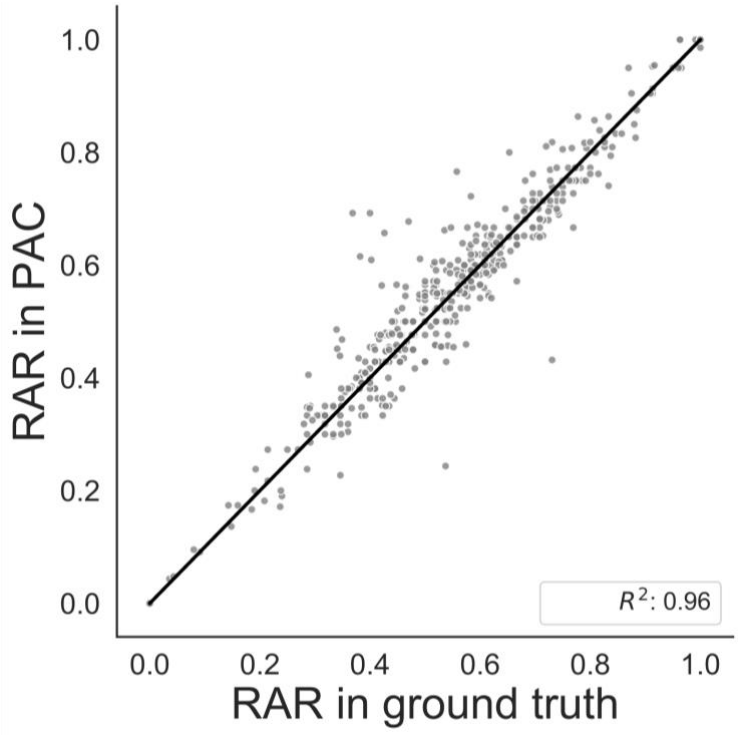


Figure 28. **The reference allele ratios (RAR) at heterozygous sites that PAC and standard alignment detect but that are discarded by WASP-filtering.**
Sites with at least 20× coverage were considered. Pearson correlation R^2=0.956, P=2.6×10^{-266}.

3.4.1.3 DOWNSAMPLING

The original simulated data was generated based on parameters from sequencing data from the Geuvadis Consortium. These data were sequenced at a very high coverage, which is rarely the case for smaller-scale studies. At a low coverage, fewer reads meet the threshold in the analysis step. Also, having lower coverage imposes difficulties in differentiating genuine variants from sequencing errors [249]. I performed a comparison of PAC and other methods against ground truth data at lower coverages of the simulated data.

I tested the performance of PAC at lower coverages by randomly resampling simulated raw data to 70% of the initial coverage (~48 million read pairs), which is roughly in line with average coverage from GTEx V8 sequencing (a commonly used resource for RNA-seq data). Because GTEx is also a large-scale study that generated high coverage, I downsampled the simulated data further to 50% (~34.5 million read pairs) and 30% (~21 million read pairs) of the original coverage. I then compared the results from each method to the ground truth allelic counts. As before, PAC outperforms both standard alignment and WASP-filtered data with all parameters tested (number of sites, difference in RAR, R^2, outliers) at all coverage levels (Table 10).

Proportion of sample	1			0.7			0.5			0.3		
Method	Standard mapping	PAC	WASP	Standard mapping	PAC	WASP	Standard mapping	PAC	WASP	Standard mapping	PAC	WASP
Sites shared with ground truth	12109	12448	11612	9984	10356	9271	8124	8452	7411	5454	5791	4888
Difference in reference allele ratio	Mean: 0.0321 Median: 0.0258	0.0233 0.0192	0.0361 0.0304	0.0359 0.0296	0.0246 0.0207	0.0412 0.0343	0.0367 0.0306	0.0252 0.021	0.0416 0.0352	0.0367 0.0299	0.0255 0.0212	0.0425 0.0357
R2 between ground truth	0.96	0.9757	0.9455	0.9541	0.9732	0.93	0.9499	0.9707	0.9244	0.9448	0.968	0.915
Outliers >20%	55	13	32	43	11	24	35	10	18	18	5	12
Outliers >10%	305	62	387	307	61	470	243	56	401	148	28	277
Sites not in standard alignment	-	350	0	-	377	0	-	333	0	-	340	0
Sites not in WASP	497	846	0	713	1089	0	713	1044	0	566	904	0

Table 10. **The impact of downsampling simulated RNA-seq data on the accuracy of standard alignment, WASP-filtered alignment and PAC.**
Correlation of reference allele ratios (RAR) between the standard alignment, WASP-filtered alignment and PAC with the ground truth data after downsampling RNA sequencing reads to 70%, 50% and 30% of the initial coverage. Pearson correlation coefficients (R^2) are shown ($P<0.05$ for all comparisons). Table is from Saukkonen *et al.*, 2022 [109].

3.4.2 PAC IN DIFFICULT-TO-MAP REGIONS

Finally, I explored the performance of PAC and other methods around regions that are known to be difficult to align and thus suffer from inaccurate allelic quantification. I measured the difference in RAR of heterozygous variants compared to the ground truth when there was an indel (>6bp) within 500bp. I also considered the heterozygous sites when there was another heterozygous site or a rare variant (MAF <1%) within close proximity (25bp). The results show that the difference in RAR from the ground truth is significantly higher in standard alignment and WASP-filtered data than PAC in all these cases (Figure 29). Therefore, at least part of the improvement in accuracy when using PAC appears to occur in the difficult-to-map genomic regions.

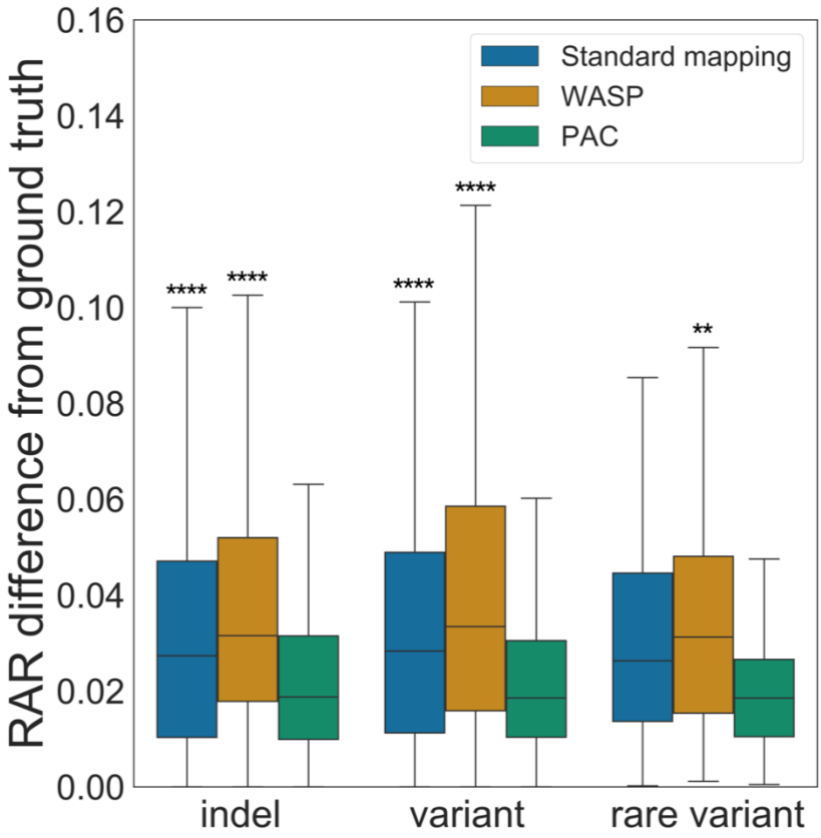


Figure 29. **The difference in reference allele ratio (RAR) of sites that are close to indel, another variant or rare variant against the ground truth.**
The RAR is significantly smaller in PAC when there is another variant, rare variant or indel close by. Heterozygous sites that are within 500bp of at least 6bp indel, within 25bp of another variant or a rare (MAF <1%) variant in different analyses shared between all methods and with at least 20× coverage are considered. A Mann–Whitney U test was performed with Bonferroni correction to adjust for multiple testing. (****) $P \leq 1 \times 10^{-4}$, (**) $1.00 \times 10^{-3} < P \leq 1.00 \times 10^{-2}$. The stars above each box plot refer to the comparison against PAC. Figure is from Saukkonen *et al.*, 2022 [109].

3.5 DISCUSSION

3.5.1 OTHER ASE TOOLS

There are other ASE detection tools, including ASEReadCounter by GATK [199], two-genome approaches such as AlleleSeq [151], and masking [141], that have been extensively used. ASEReadCounter performs basic RNA-seq read filtering before providing ASE count data. Tools with a two-genomes approach generally only generate two personalised genomes that allow the user to align RNA-seq reads to both of these. Masking low-quality heterozygous positions in the genome has been extensively performed [250-252], however, it has been shown that it does not produce more reliable results and the bias persists with this method [141].

None of the previous ASE methods address all the issues involved with the field within one workflow, nor are they easy to implement. Most likely these are the reasons that standard alignment to the reference genome is still the most commonly used method. There is no standardised ASE detection method, therefore there is a gap in the field in this area. For this reason, in this chapter, I wrote PAC into Nextflow and made it available on GitHub. The user needs minimal effort to run PAC following an initial simple install from GitHub; the run is automated including the download of all the tools and dependencies.

3.5.2 PAC INTO STREAMLINED GENOMICS WORKFLOW

Genomics pipelines are often complicated workflows with multiple steps and requiring different tools [231]. Their use often requires expertise in bioinformatics. Because academia favours rapid publications and analyses, software development is often not a priority [253]. Consequently, they are usually generated predominantly for on-premises use [244]. However, with the advancement in genomics data and workflows, streamlined tools are of paramount importance. Studies have shown that the results from genomic studies cannot be replicated nor the workflows easily adapted [254, 255]. Despite a large number of genomics tools available, while the need for these is essential for clinical settings, not many

have been translated into a clinical field. One of the reasons is believed to be the poor quality of bioinformatics tools [256]. To reproduce results from other research groups, generally a substantial amount of time and effort is necessary [255, 257].

For these reasons, I have made PAC available in a format that is reproducible on any computational system and easy to use. In this chapter, I described the steps within PAC, how it was written in Nextflow and Docker that automatically install all the necessary tools and software for PAC to run. I have provided a user and reference manual to make PAC easier to use. Upon benchmarking the run time, using five test GTEx samples of average depth of ∼44 million paired reads, PAC takes almost 3.5 times longer to run than a standard alignment approach to obtain the same data. However, PAC provides additional information and accuracy that can be crucial for studies with small sample sizes or studies that look into variants/genes with small effect sizes.

3.5.3 VALIDATING PERFORMANCE OF PAC

Following this, I validated the performance of PAC by comparing the accuracy against ground truth data obtained in Chapter 2. I compared the performance of PAC against standard alignment to the reference genome and WASP-filtered alignment methods. To date, there has not been a study with realistic simulated data with an absolute ground truth that allows to evaluate the true performance and error rate of allelic counts. The validation of ASE events in real data is almost impossible as there are multiple steps where errors might arise, including laboratory methods, data acquisition and processing and data filtering [181]. Therefore, realistic genomic data allowed me to evaluate and compare the performance of different methods.

I showed that PAC keeps more reads, the allelic ratio is closer to that of ground truth, it has fewer outliers, and the additional reads that it picks up that were discarded by other methods are assigned accurately. I show that this is unaffected by coverage of the simulated data by downsampling the simulated data. PAC shows slight overcorrection when considering the RAR distribution of heterozygous sites, however, it is less than in WASP-

filtered alignment, and the RAR is also closer to the ground truth RAR than in standard alignment.

I show that PAC makes significantly fewer mistakes in regions that are known to be problematic in the ASE field, in particular near other variants, rare variants and indels [258]. These regions are often difficult to align with traditional methods as indels can cause a shift in the codon reading frame and the read then contains mismatches that would not allow it to pass the filtering steps. Having another variant is problematic since the alternative allele will already carry a mismatch from the reference sequence, thus reducing the alignment score and being more likely to be discarded. Rare variants will be even less likely to be present in population-level data and thus more problematic, and hence these will be more likely to be removed.

I show that WASP removed a large number of reads as expected. Unexpectedly, however, WASP-filtering performed worse than standard alignment in terms of the correlation between the ground truth and WASP-filtered alignment reference allele ratio, mean difference in the reference allele ratio, and WASP-filtering increased the number of 10% outliers. WASP reduced the number of 20% outliers, where the RAR deviated strongly from the ground truth. I also showed that WASP-filtering overcorrects the data when considering the RAR distribution compared to the ground truth. Depending on the analysis, this might be an acceptable trade-off, however, the reduction in the number of reads and an overall reduction in performance can be problematic for rare variants and smaller sample sizes such as rare disease studies where ASE is gaining interest.

3.5.4 FUTURE OF PAC

In this chapter, I have developed a streamlined and easy-to-use pipeline to accurately quantify allelic counts. Because of this, users do not need an expertise knowledge in bioinformatics and PAC is therefore accessible to a wider audience. I have shown that it outperforms other commonly used methods and is able to quantify ASE reasonably well in difficult-to-map regions. I expect PAC to be particularly useful for studies in rare diseases where the sample sizes are small and the additional information that PAC provides can be

crucial. PAC would also be useful in studies that are performed at lower coverage, where common methods would not meet the threshold and would discard a higher proportion of reads that these studies cannot afford.

CHAPTER 4 – APPLYING PAC TO POPULATION LEVEL DATA

4.1 INTRODUCTION

4.1.1 RESEARCH ON GENE EXPRESSION LEVELS AT POPULATION LEVEL

GWA and eQTL studies have shown that variants in the non-coding genome contribute to variation in gene expression levels; this is associated with phenotypic variance in the population [72, 88, 92]. eQTL analysis is the most commonly used method of studying genetic regulation of expression, which is less prone to technical artefacts due to the large population samples that it requires. However, for the same reason, this method prevents the study of rare variants and small sample sizes, as can be the case in rare disease cohorts. Therefore, ASE analysis that can be measured within a single individual is becoming more widely used as a way to detect genetic regulation of gene expression in *cis*. ASE offers a method to study regulatory variation [259], but can also be applied to a wide array of different fields. ASE analysis can complement eQTL studies by improving the power to map genotypes [260], and by helping identify candidate genes [261, 262], ASE analysis has been used alone to identify genes associated with disease [168]. ASE has been used in model organisms to better understand environmental adaptations [263, 264] and can be utilised to study epigenetic gene regulation in *cis* including imprinting [265], X-chromosome inactivation [266] and monoallelic expression on autosomal chromosomes [267-269]. However, to utilise ASE analysis, statistical methods are needed that differentiate normal biological noise from a genuine variation, which is crucial when analysing smaller sample sizes where the power will be small, as is often the case with ASE analysis.

4.1.2 THE DIFFICULTIES IN INTERPRETING ASE DATA

ASE captures the cumulative effects of variants in regulatory regions on gene expression within a single individual. When the gene expression from one allele differs from that of the other, it is called allelic imbalance. The most common way of quantifying this statistically is via a binomial test, which determines if the results significantly deviate from the expected

outcome. Within the ASE field, this quantifies if the observed allelic ratio significantly deviates from the expected 1:1 ratio (where the gene/heterozygous variant is not under ASE and is expressed at the same level on both genetic backgrounds) [64, 94, 124]. This is a very simple way of testing for ASE, and therefore, is a widely used method [94, 270]. However, this ASE quantification method comes with its technical biases [271]. The binomial test assumes the data is binomially distributed, meaning each trial is independent. This may not be the case for genetics where each gene can have multiple heterozygous variants displaying imbalanced allelic ratios and display different expression patterns due to alternative splicing; or vary between cells for example due to methylation events. In addition, read count data is known to be overdispersed relative to what is expected from a binomial distribution [272]. Overdispersion means that the variance of the read count in RNA-seq data is a lot larger than the mean [273, 274]. The overdispersion is most likely due to technical and biological effects [123] such as mapping bias and measurement error. Indeed, overdispersion has been shown to be reduced with increased sequencing depth [275]. The binomial test does not account for this overdispersion. Even after quality control filters, a binomial test inflates *p* values [122], and as such is likely to be leading to false positives. There are other statistical approaches that have been applied to ASE data, including variations of the beta-binomial test to account for the degree of overdispersion [123, 159, 276]; the Beta-binomial test introduces an additional variable to account for this overdispersion that it learns from the data [277]. The results between binomial and beta-binomial tests are similar in data where the degree of overdispersion is similar [271]; therefore, the overdispersion most likely comes from high-coverage sites. There are also Bayesian inference methods to assess ASE [163, 278], however, it remains difficult to distinguish genuine biological causes from artefacts.

In addition, because genes have different isoforms, it often complicates how ASE is quantified. Traditionally, the variant with the highest coverage per gene was used as a representative for the gene. However, now with the improved methodology, it is preferred to aggregate the variants, to provide ASE estimates on a haplotype level [191]. This is illustrated in Figure 30.

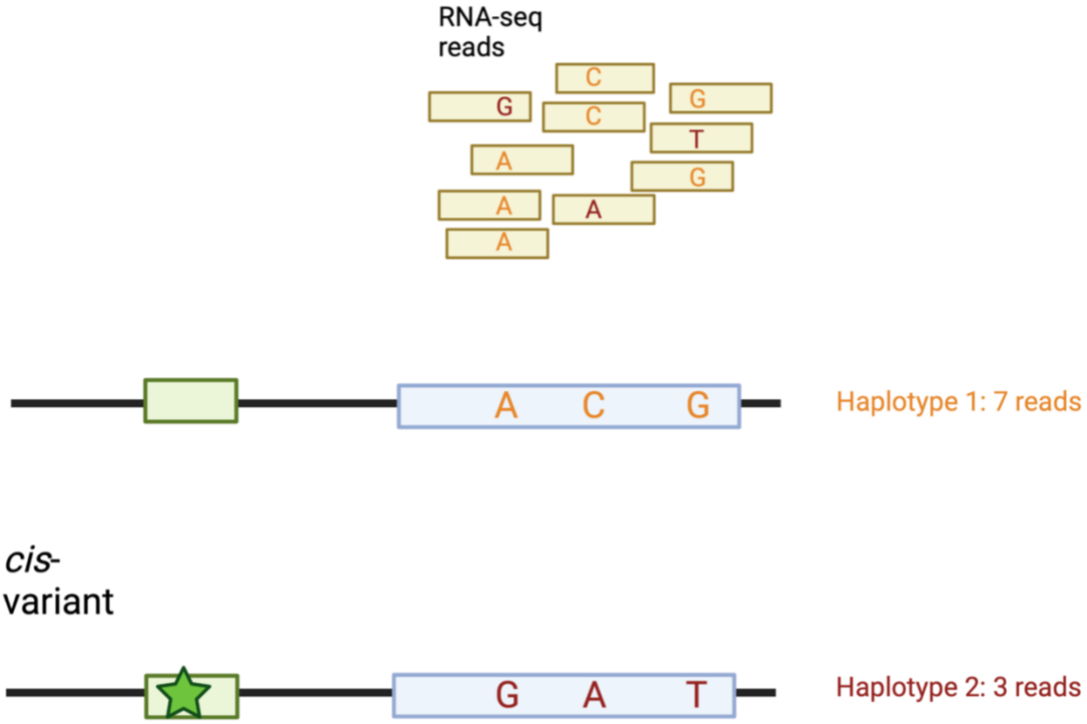


Figure 30. **Haplotype level ASE quantification.**
The variants within a read that distinguish alleles are phased and counted at a haplotype level to give ASE estimates across the haplotype. Created with BioRender.com.

4.1.3 VALIDATING ASE DATA

Since ASE suffers from many technical biases, it is difficult to validate the accuracy of allelic counts using real data, as the underlying ground truth is not usually known, and alternative methods to quantify allelic expression capture the effects of many different processes (amplification biases, alignment biases etc), or come with their own limitations. Therefore, comparison to population-based statistics is often employed to assess the accuracy of allelic counts at the single sample level, as it is more robust to artefacts due to statistical power. eQTLs are related to ASEs and therefore often can aid the discovery of regulatory events. ASE is highly heritable [94] and therefore unsurprisingly almost every gene has an eQTL [64, 86, 279]; therefore, most ASEs should be captured when comparing to eQTL signals. ASE will identify the cumulative effects of additional rare variants, however, although these may be

crucial for biology, they will be in the minority across samples and genes, and as such, should not affect population level statistics driven by associations with common variants.

eQTL effect sizes are often estimated with a linear regression slope [280, 281]. The allelic imbalance can be quantified by allelic fold change (aFC) [150] which describes expression change associated with a given variant(s). aFC can also be calculated from eQTL data, which allows a direct comparison of the two and therefore also a way to validate ASE results. aFC is explained in Figure 31. In order to ease the comparison between the two methods, a new tool has been developed, phASER-POP [108]. This tool phases variants at a gene level and at a population level, and provides an estimate of aFC from both eQTL and ASE data. phASER-POP phases variants in every individual with allelic expression by combining the integrated genotypes with haplotype level ASE in individuals under study. The correlation between eQTL and ASE aFC is often high, however it is improved with haplotype level ASE and removing mapping bias with WASP, which removes ambiguous reads [123].

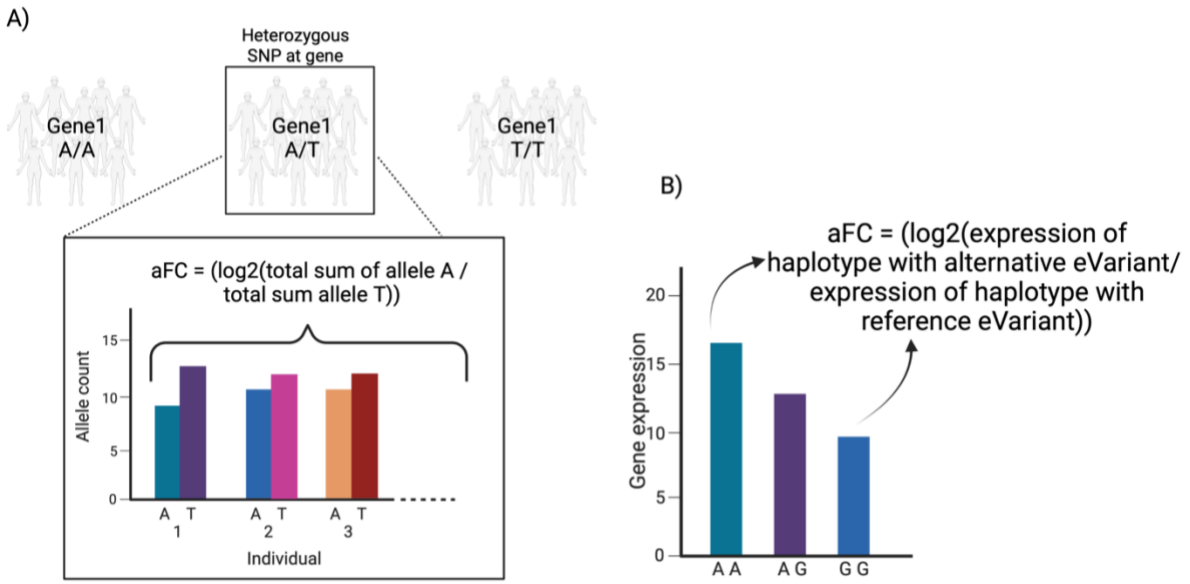


Figure 31. **aFC calculation for eQTL and ASE.**
For eQTL (panel A) aFC is calculated by obtaining log2 of total sum of allele 1 over allele 2 in individuals that are heterozygous for that SNP. For ASE (panel B), aFC is calculated by obtaining log2 of ratio between expression of haplotype carrying alternative variant and reference variants. Created with BioRender.com.

Another way to potentially validate ASE on real data is by using long-read sequencing. Long reads make it possible to more accurately align data around repetitive sequences [84], large insertions/deletions and large chromosomal rearrangements [282]. With long-read sequencing, it is also possible to study the transcript structure and quantify variants without the need to phase them. Phasing can be challenging with rare variants [283], which is important in the ASE field. Phasing is required to differentiate between compound heterozygosity (Figure 9) [144], or to correctly quantify ASE at a haplotype level. ASE analysis is often used with smaller sample sizes where rare variants may play a bigger role than in large population-based studies, which are challenging to phase due to the low presence in population-level data, which is used for computational phasing approaches. Additionally, we know that changes in transcript structure are often accompanied by transcript level changes [284], thus accurate determination is important. Long-read sequencing eliminates the burden of biases associated with short-read sequencing that can lead to inaccurate transcript structure determination and quantification. However, to date the error rate is high and the cost of sequencing is relatively high [84, 285], which limits large population-level studies. In addition, the vast amount of short-read sequencing data available makes it easier to answer a biological question of interest by re-analysing existing data.

In this chapter, I will assess the accuracy of allelic counts at heterozygous sites using PAC by comparing them to population-level data that is generally more robust. I will also attempt to validate PAC with long-read sequencing data. Finally, I will identify whether improved RNA sequencing alignment and allelic quantification with PAC allows a more robust detection of biological signals.

4.2 CHAPTER OVERVIEW

In this chapter I validated ASE data obtained from PAC. I utilised multiple methods to achieve this, and compared PAC against other commonly used methods, standard alignment and WASP-filtering. Validating ASE experimentally is difficult, with every method having its own limitations that can lead to biases. For this reason, in this chapter, I utilised multiple avenues and data types to validate ASE and in particular to quantify the improved ASE detection obtained when using PAC. Figure 32 summarises this chapter and different validation methods.

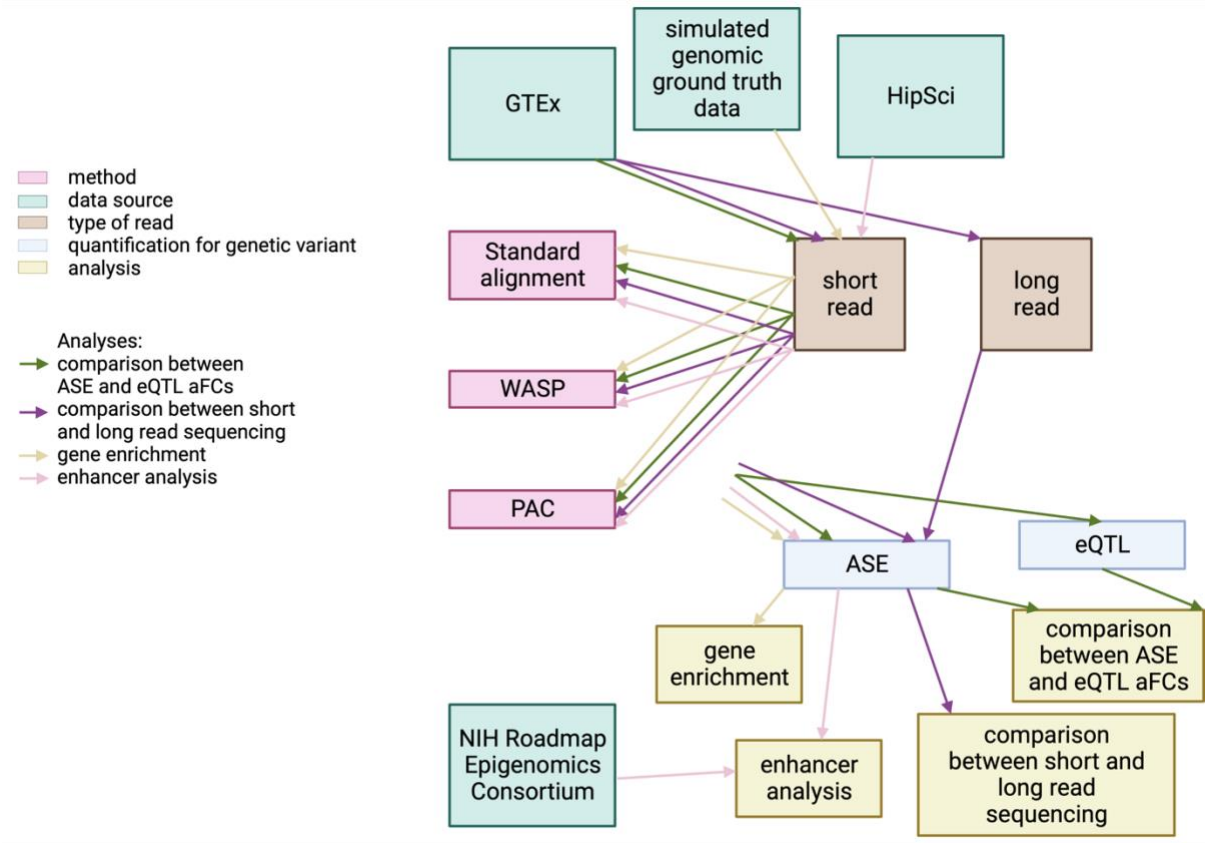


Figure 32. **Overview of Chapter 4 analyses.**
To validate PAC as a method that improves ASE detection, four different analyses were performed. In the first analysis (green arrows), GTEx short read sequencing data was aligned by standard alignment to the reference genome, with WASP-filtering and with PAC. The aFC for eQTLs and ASE were compared. In the second analysis (purple arrows), GTEx short read sequencing data was again aligned by standard alignment to the reference genome, with WASP-filtering and with PAC, and ASE count data obtained from these methods. Long read sequencing from GTEx was also used to quantify ASE events. These two ASE count datasets were then compared to different methods. In the third analysis (yellow arrows), the simulated RNA-seq data (from chapter 2) was aligned by standard alignment to the reference genome, with WASP-filtering and with PAC. The biological enrichment of genes under ASE from all these methods and the ground truth data were compared. In the fourth analysis (pink arrows), the short read sequencing data from the HipSci Project was aligned by standard alignment to the reference genome and with PAC. The abundance of enhancer regions (obtained from NIH Roadmap Epigenomics Consortium) was compared between ASE and non-ASE genes. Created with BioRender.com.

4.3 METHODS

4.3.1 DATA DESCRIPTION

4.3.1.1 GTEX

The GTEx (Genotype- Tissue Expression) Project [103] generated RNA-seq data from deceased individuals from multiple tissues, as well as WGS and genotyping data from whole blood. The Project provides a vast resource of data including genotyping calls, gene expression and eQTLs. V8 release has 948 donors and 54 tissues. The donor age is between 20-71, with 67.1% being male and the majority at 84.6% being of the white genetic ancestry. The mean number of tissues collected per donor is 19.

4.3.1.2 NANOPORE

Nanopore data for GTEx samples were obtained from Glinos *et al*., 2021 [284] where the samples were sequenced with the Oxford Nanopore Technologies platform. Expression and allelic count data are deposited on the GTEx portal. I used 5 frontal cortex, 5 atrial appendage, 5 left ventricle, 6 lung, 6 skeletal muscle and 6 liver samples. The sample IDs are shown in Table 11.

	Frontal cortex	Atrial appendage	Left ventricle	Lung	Skeletal muscle	Liver
Sample IDs	GTEX-1192X-0011	GTEX-1GN1W-0226	GTEX-1I6K7-0626	GTEX-1I6K7-1226	GTEX-1C64N-0326	GTEX-R53T-0326
	GTEX-13X6J-0011	GTEX-1IDJD-0226	GTEX-13QBU-0426	GTEX-1KXAM-0426	GTEX-1KXAM-2426	GTEX-UTHO-2426
	GTEX-14BIL-0011	GTEX-1IDJF-0826	GTEX-15RIE-1726	GTEX-14BMU-0526	GTEX-1LVA9-0326	GTEX-WY7C-0726
	GTEX-15DCD-0011	GTEX-14XAO-0926	GTEX-OHPL-0326	GTEX-1211K-0826	GTEX-13QJ3-0726	GTEX-Y5LM-0426
	GTEX-QDT8-0011	GTEX-WY7C-1126	GTEX-ZVZP-0226	GTEX-WYVS-0526	GTEX-17MFQ-1926	GTEX-ZF29-2026
				GTEX-ZT9X-0326	GTEX-ZT9X-1826	GTEX-ZPU1-0826

Table 11. **Sample IDs for GTEx samples for nanopore analysis.**

4.3.1.3 ROADMAP EPIGENOMICS PROJECT

The chromatin state data was obtained from the NIH Roadmap Epigenomics Consortium. The project consists of 111 consolidated epigenomes from the Roadmap Epigenomics Project that were analysed together with 16 epigenomes previously reported by The Encyclopedia of DNA Elements (ENCODE) project [286]. The data is publicly available and contains global maps of regulatory elements for different cell types.

The Consortium used a variety of methods including bisulfite treatment, DNA digestion by DNase I, RNA profiling, chromatin immunoprecipitation, methylated DNA immunoprecipitation and methylation-sensitive restriction enzyme digestion to identify regions of regulatory elements, including histone marks, DNA methylation, DNA accessibility and RNA expression. To identify the significant combinatorial interactions in different chromatin marks and classify genomic regions based on these data, a model based on a multivariate Hidden Markov Model was used [286]. The pluripotent cell lines that were selected for the project included eight embryonic stem cell lines (E001, E002, E003, E008, E014, E015, E016, E024), and five iPSC lines (E018, E019, E020, E021, E022) which have been shown to cluster together based on enhancer signals and are similar to each other in terms of pluripotency [286]. All active enhancer regions (EnhA1 and EnhA2 chromatin states) from 13 samples were included in the analysis. The Consortium showed that around 5% of the genomes consist of enhancer or promoter regions [286].

4.3.2 THE COMPARISON OF ASE WITH EQTL ANALYSIS

To recapitulate population level data with ASE results obtained from PAC, standard alignment and WASP-filtered data, I obtained aligned RNA-seq data containing all reads for 670 whole blood samples from the GTEx project (v8, aligned to the hg38 reference genome), which were obtained via the GTEx Portal and dbGaP (dbGaP accession number phs000424.v8.p2) [104]. I converted these files back to raw FASTQ sequence files with SAMtools. I also obtained phased genetic variant calls from WGS data obtained from the GTEx Portal via dbGaP (phASER_GTEx_v8_merged.vcf.gz). I then used the converted RNA-seq reads and phased variants as inputs for PAC (selecting the GRCh38 reference genome).

For this analysis, I updated PAC to the more recent STAR version 2.7.4a. The difference in output from more recent STAR versions is marginal. The BAM file PAC generates with STAR 2.51 (previous version) and 2.74 (current) had only 10 reads different from ~95 million when using simulated RNA-seq from Chapter 2.

With >=20× coverage, GTEx samples have an average of 9972 (SD=3809) heterozygous variants, with at least one read present for each nucleotide. PAC uses phASER to provide phased haplotypes and the haplotypic count data per individual. The formed matrix then feeds into the phASER-POP [108] to obtain aFC value per gene across individuals, for genes and samples with at least 8 reads. In all analyses, I used median aFC across individuals. I supplied lead eQTL variants identified in the GTEx project (v8) for each gene, obtained from the GTEx portal (Whole_Blood.v8.egenes.txt.gz). I also ran phASER-POP using two additional gene count matrix files, one for standard alignment (phASER_GTEx_v8_matrix.gw_phased.txt.gz), and the other for WASP-filtered alignment (phASER_WASP_GTEx_v8_matrix.gw_phased.txt.gz), both obtained through the GTEx portal, produced by Castel *et al*., 2020 [108].

I then compared the aFC between eQTL data and ASE data for PAC, standard alignment, and WASP-filtered alignment. I selected genes with at least ten individuals heterozygous for the lead eQTL variant associated with the gene, with aFC estimates generated from eQTL data after filtering genes where the eQTL association was q-value < 5%. I selected only genes that were present in all three methods after these filtering steps for direct comparison.

4.3.3 NANOPORE ANALYSIS

To compare allelic data from nanopore to allelic data obtained from PAC, standard alignment and WASP-filtered data, I obtained short-read RNA-seq data, genotyping calls and nanopore long-read sequencing (LORALS_GTEx_v9_ase_quant_results.gencode.txt.gz) data for 5 frontal cortex, 5 atrial appendage, 5 left ventricle, 6 lung, 6 skeletal muscle and 6 liver GTEx samples (Table 11) from GTEx Portal and via dbGaP. I also downloaded allelic counts per gene for the same samples for short-read RNA-sequencing aligned to the GRCh38

reference genome (phASER_GTEx_v8_matrix.gw_phased.txt.gz) and WASP-filtered alignment (phASER_WASP_GTEx_v8_matrix.gw_phased.txt.gz) samples.

The short-read RNA-seq read for the samples together with genotyping calls were used as inputs for PAC (selecting GRCh38 to make it comparable to other methods), obtaining allelic counts at haplotype level for each gene. I compared allelic ratios between standard alignment, WASP-filtered data, and PAC, to the nanopore data. For this, genes present in all methods that were on autosomal chromosomes with >=20× coverage were selected.

4.3.4 ENRICHMENT ANALYSIS

To investigate how PAC, standard alignment and WASP-filtered alignment recapitulate enrichment of genes under ASE in ground truth data, I used allelic counts from simulated data (chapter 2, section 2.2.2) for paternal and maternal alleles for individual NA12877 that were used as a ground truth baseline against which other methods were compared. The HLA (obtained from phASER) and blacklist regions (obtained from ENCODE ENCFF001TDO.bed) were removed. Only SNPs on autosomal chromosomes with at least 20× coverage were included in the analysis.

I annotated all variants with wANNOVAR (https://wannovar.wglab.org) in each method. These variants acted as a background list for enrichment analysis. I selected the heterozygous sites with the highest coverage for each gene to be representative for that gene. I counted allelic ratios and obtained significant sites under ASE using the binomial test at $P<0.05$. I performed enrichment analysis using gProfiler2 R package for the gene for which the representative variant was under ASE, with all expressed variants acting as the background. I performed this for ground truth allelic counts, and the output from PAC, standard alignment and WASP-filtered alignment, using the same simulated RNA-seq reads.

4.3.5 ENHANCER ANALYSIS

There has been evidence that shows that genetic risk variants are enriched in enhancer regions [287, 288]. For example, around 30% of non-coding SNPs associated with Alzheimer's disease are located in enhancers [289]. Another study showed that a non-coding risk variant associated with Parkinson's disease is located in an enhancer region, which upregulates the expression of a disease susceptibility gene [290]. Because the regulatory regions including enhancers influence gene expression, these variants are likely to cause ASE in many cases. This provides an excellent avenue to utilise ASE analysis to investigate if the genes under ASE differ in their local enhancer abundance. The logic behind this is that since enhancers are enriched for the genetic variants, gene expression would be influenced in an allele-specific manner. In order to examine this, I ran PAC on 10 healthy HipSci iPSC samples (donors described in Table 1, data acquisition described in section 2.2.1.1). I annotated variants using the GENCODE v19 GTF file to obtain information on which gene each variant falls into. For iPSC data for each donor, I selected genes (with at least one heterozygous site with >=10× coverage) under ASE using binomial statistics with detection at $P<0.05$ and Bonferroni-adjusted (where the P-value is divided by the number of heterozygous sites) for a more stringent criteria, and non-ASE genes where the statistics did not meet this threshold.

All active enhancer regions from 13 pluripotent samples, obtained from the NIH Roadmap Epigenomics Project, were merged together. The enhancer regions can vary slightly in their locations between cell lines. In order to remove the same regions, I removed duplicate samples by merging the enhancers that overlapped by at least 1 bp together using BEDtools.

Enhancer abundance near genes was measured by the number of base pairs, annotated as active enhancers within 1Mb from gene start and gene end site. I pooled together all donor ASE and non-ASE genes and tested for significant differences in enhancer abundance. I summed the total number of base pairs of enhancer abundance within a 1Mb range for ASE and non-ASE genes, and divided it by the total number of genes in each sample. The significance was calculated with the paired non-parametric Wilcoxon signed rank test.

I also tested the significant difference in enhancer abundance for genes under ASE and non-ASE within each donor. I summed the total number of base pairs of enhancer abundance within a 1Mb range per gene for ASE and non-ASE genes. I tested the significance with the non-paired non-parametric Wilcoxon rank sum test. All analysis was performed with custom scripts written in Python and R.

4.4 RESULTS

To validate PAC as a method, I performed analyses using real data and compared the performance of PAC against standard alignment and WASP-filtered alignment. In this chapter, I first examined how well ASE data recapitulates population-level eQTL signals, which are more robust against artificial and technical biases. I then compared ASE data from short-read sequencing obtained from PAC, standard alignment, and WASP-filtered data against that from long-read sequencing, which in theory circumvents mapping biases and phasing errors that are a problem for ASE analysis. I then validated PAC by comparing how the functional enrichment of genes under ASE best replicates the enrichment results from ground truth data, compared to other methods. And finally, I applied PAC in the context of understanding the genomic regulatory environment surrounding genes under ASE, to better understand the biology of gene expression regulation.

4.4.1 VALIDATING PAC: ASE VERSUS EQTL AFC ON GTEX WHOLE BLOOD DATA

To validate the performance of PAC, I explored how well ASE data generated by the pipeline recapitulates population-level signals obtained via eQTL analysis. Although ASE analysis is able to capture more information by not relying on large sample sizes, it also has the ability to capture population-level signals. In this analysis, I compared the allelic fold change (aFC) [150] (which describes the size of the effect of alternative alleles on gene expression levels) generated from ASE data to those obtained from eQTL mapping, with the theory being that improved allelic quantification by PAC should improve this correlation compared to other methods.

To generate allelic count data, I aligned 670 whole blood RNA-seq samples from the GTEx project (v8) [104] using PAC generating gene-level counts. Within PAC, using phASER-POP [108] I then used these counts together with information on lead eQTL SNPs to calculate the aFC for each gene (as a function of whether an individual carries the lead eQTL variant). To

compare how well PAC performs against other methods, I also obtained gene-level counts from the GTEx portal for data that had undergone standard alignment and WASP-filtered alignment, and then ran phASER-POP [108] to generate aFC per gene across individuals for each of these methods. I compared the aFC estimates obtained from allelic count data generated from PAC, standard alignment and WASP-filtered data to aFC values generated from eQTL data (data obtained from Castel *et al*., 2020 [108]). I selected the 8913 genes that had a significant eQTL (q-value < 5%) and where the aFC could be calculated from ASE data in all three methods and were present at >=20× coverage. The results are shown in Figure 33.

PAC shows the strongest correlation of Genome-wide Pearson correlation coefficients (R^2) =0.842 between gene-level aFC generated from ASE and eQTL data. The correlation in WASP-filtered alignment is lower at R^2=0.829, and for data obtained through standard alignment, there is the lowest correlation of R^2=0.820. These results show that WASP-filtering slightly improves the correlation between aFC values from ASE and eQTL data, while PAC improves it considerably more. WASP mostly works by removing ambiguous reads while PAC employs multiple correction steps including diploid genome, retention of multi-mapping reads, and optimised alignment parameters; therefore, PAC appears to have more power to recapitulate population level signals.

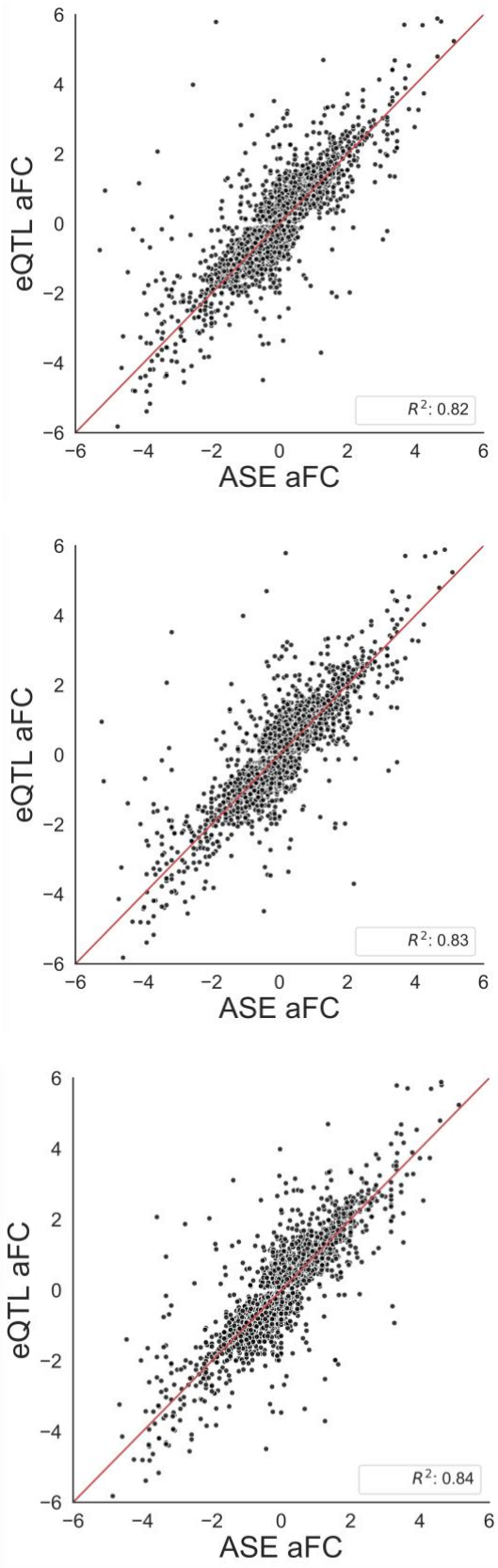


Figure 33. **Correlation of allelic fold change (aFC) values derived from ASE and eQTL analyses from GTEx whole blood samples.**
aFC from standard alignment recapitulates eQTL signal relatively well. WASP-filtering improves this correlation, and the correlation is improved even more in data obtained from PAC. Pearson's correlation coefficients are shown for eQTL versus ASE aFCs.

In addition to improving the accuracy, PAC also aligns more reads across heterozygous sites compared to standard or WASP-filtered alignment. As a result, more genes meet the minimum coverage thresholds for PAC data compared to the other methods. Compared to WASP-filtered alignment, 740 more genes were retained in PAC data at the specified coverage cut-offs (regardless of whether they were present in standard alignment data). To assess the accuracy of allele counts generated from these additional 740 genes, I next examined if these genes are informative and assigned accurately by PAC by comparing aFC generated from ASE and eQTL data. The aFC between ASE and eQTL data in these additional genes was still high at Genome-wide Pearson correlation coefficients (R^2)=0.653, P=4.0×10^{-91} (Figure 34 A). I then examined 319 genes that were not present in WASP-filtered data or standard alignment, similarly showing a significant correlation of R^2=0.643, P=1.1×10^{-38} between ASE and eQTL aFC (Figure 34 B). When looking at 421 genes that were present in PAC and standard alignment but not in WASP-filtered data, the correlation was slightly improved (R^2=0.669, P=6.9×10^{-56}) (Figure 34 C).

These results demonstrate that PAC improves the accuracy of allele counts at heterozygous sites as the correlation of aFC values between ASE and eQTL data is higher for PAC and thus better recapitulates population-level signals that are less prone to computational biases. PAC also enables the study of more genes due to increased coverage gained by accurately assigning additional reads to their correct genomic location, potentially capturing more information for biologically important genes. This is crucial for smaller sample sizes, for which ASE is often employed for.

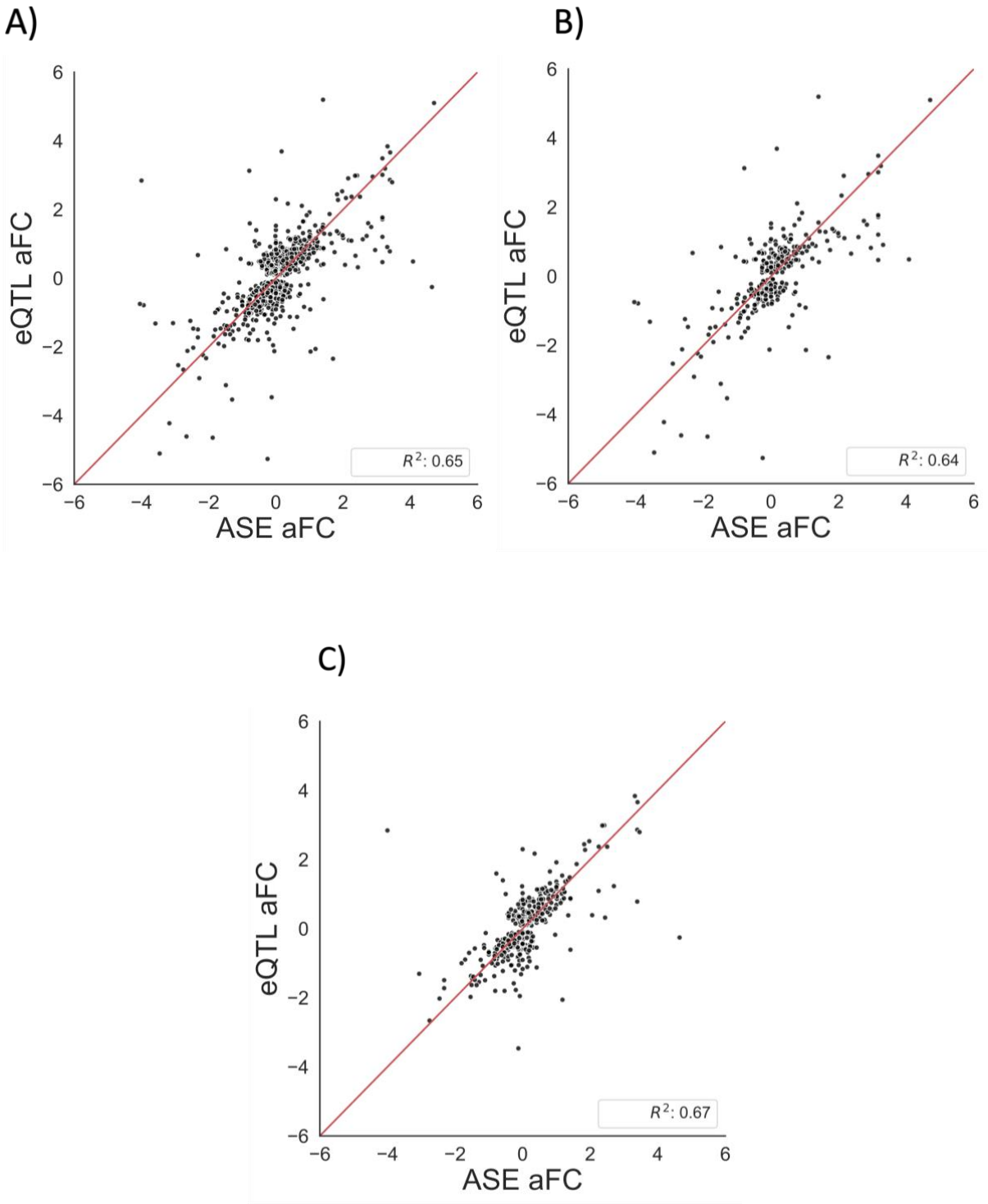


Figure 34. **Correlation of allelic fold change (aFC) values derived from ASE and eQTL analyses for extra genes that can be analysed in PAC relative to standard alignment or WASP-filtered data from GTEx whole blood samples.**
A) There are 740 extra genes present in PAC but discarded in WASP-filtered data and these show a high correlation (R^2=0.65) with the ground truth data. B) The 319 extra genes present in PAC that are discarded in standard alignment or WASP-filtered data from GTEx whole blood samples show slightly reduced but still high correlation (R^2=0.64) with the ground truth data. C) There are 421 extra genes present in PAC that are present in standard alignment but discarded in WASP-filtered data, and these show high correlation (R^2=0.67) with the ground truth data. Pearson's correlation coefficients are shown for eQTL versus ASE aFCs.

4.4.2 ASSESSING PAC WITH NANOPORE DATA

As an attempt to further validate the accuracy of allelic counts generated by PAC, I examined the utility of long-read sequencing to validate short-read ASE data. Long-read sequencing avoids phasing and the reference allele bias associated with short-read sequencing. Long-read sequencing is more likely to be uniquely mappable and therefore is thought to be a more accurate method for ASE detection. A recent study by Glinos *et al*., 2021 [284] generated long-read sequencing using GTEx data, and gene level allelic counts have been made available on the GTEx Portal for 88 samples across 14 different tissue types. For this analysis, I selected 5 frontal cortex, 5 atrial appendage, 5 left ventricle, 6 lung, 6 skeletal muscle and 6 liver samples (Table 11). I obtained allelic counts per gene for these samples, together with the same data obtained from the short-read sequencing for the standard alignment and WASP-filtered alignment. Then, I obtained raw short-read RNA sequencing data for the same samples and ran them with PAC to obtain allelic counts per gene. After collating all data across these approaches, I then compared gene-level allelic ratios generated from long-read data against those generated from short-read data after applying standard alignment, WASP filtering and PAC.

Unexpectedly, there were no significant correlations between any methods and nanopore data when comparing the allelic ratios when all tissue data was pooled together ($P>0.05$ in all cases, Figure 35). The allelic ratios in standard alignment and WASP-filtered data were more dispersed than in PAC, however, the data did not correlate in any analyses. Performing the comparison within each tissue (Figure 36), the trend remained the same ($P>0.05$ in all cases).

I considered the mean difference in allelic ratios between PAC, standard alignment and WASP-filtered alignment from that in nanopore across all tissues analysed. The mean difference in standard alignment was 0.0869, in WASP-filtered data it was 0.0863 and in PAC it was 0.0851. As such, there is no correlation between allelic ratios generated from long-read sequencing and the same data generated via any method from short-read sequencing. The original study also only observed a moderate concordance [284].

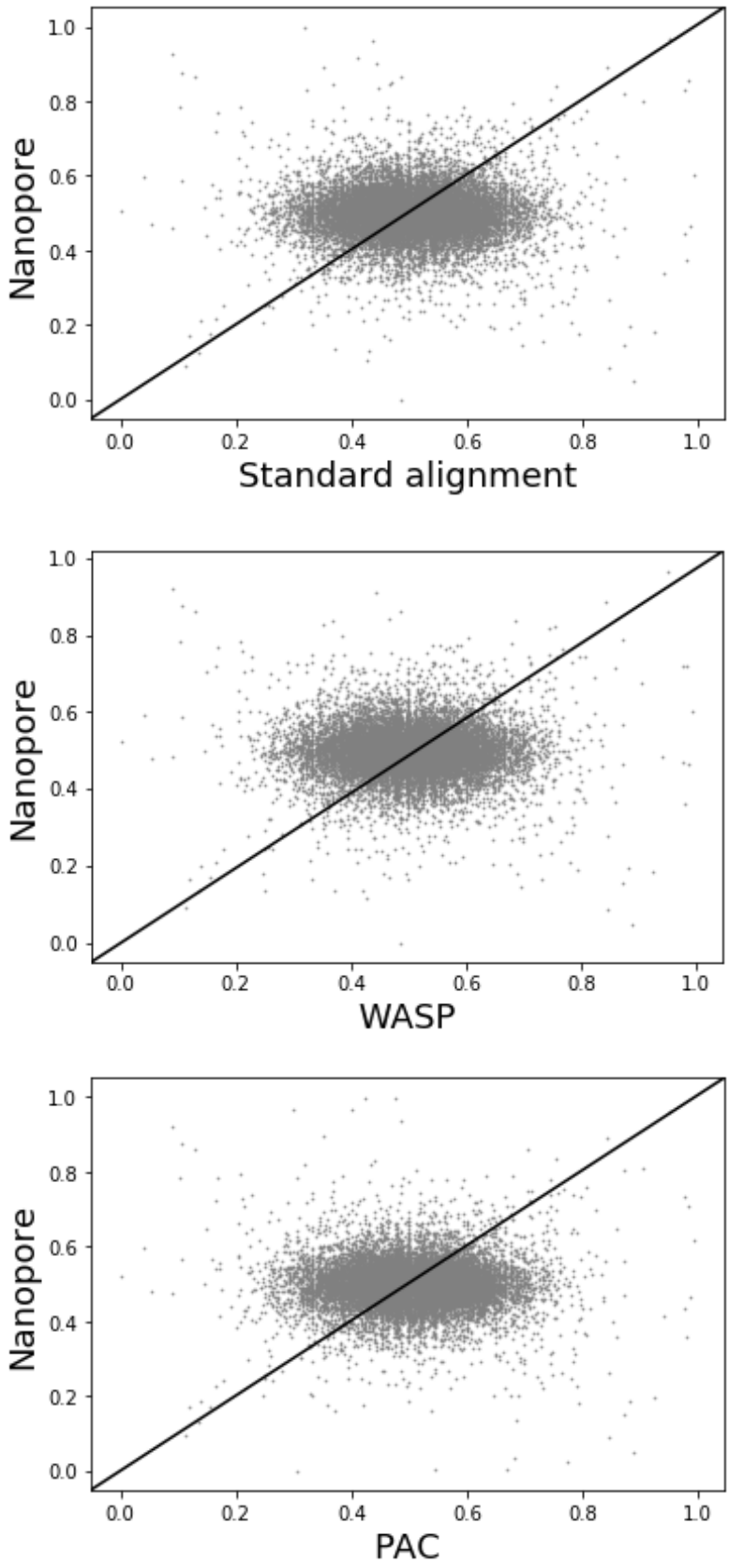


Figure 35. **Comparison of allelic ratios between Nanopore and standard alignment, WASP-filtered data and PAC.** None of the approaches had significant correlation between the short-read and the long-read sequencing. Data shown for all GTEx tissues pooled together.

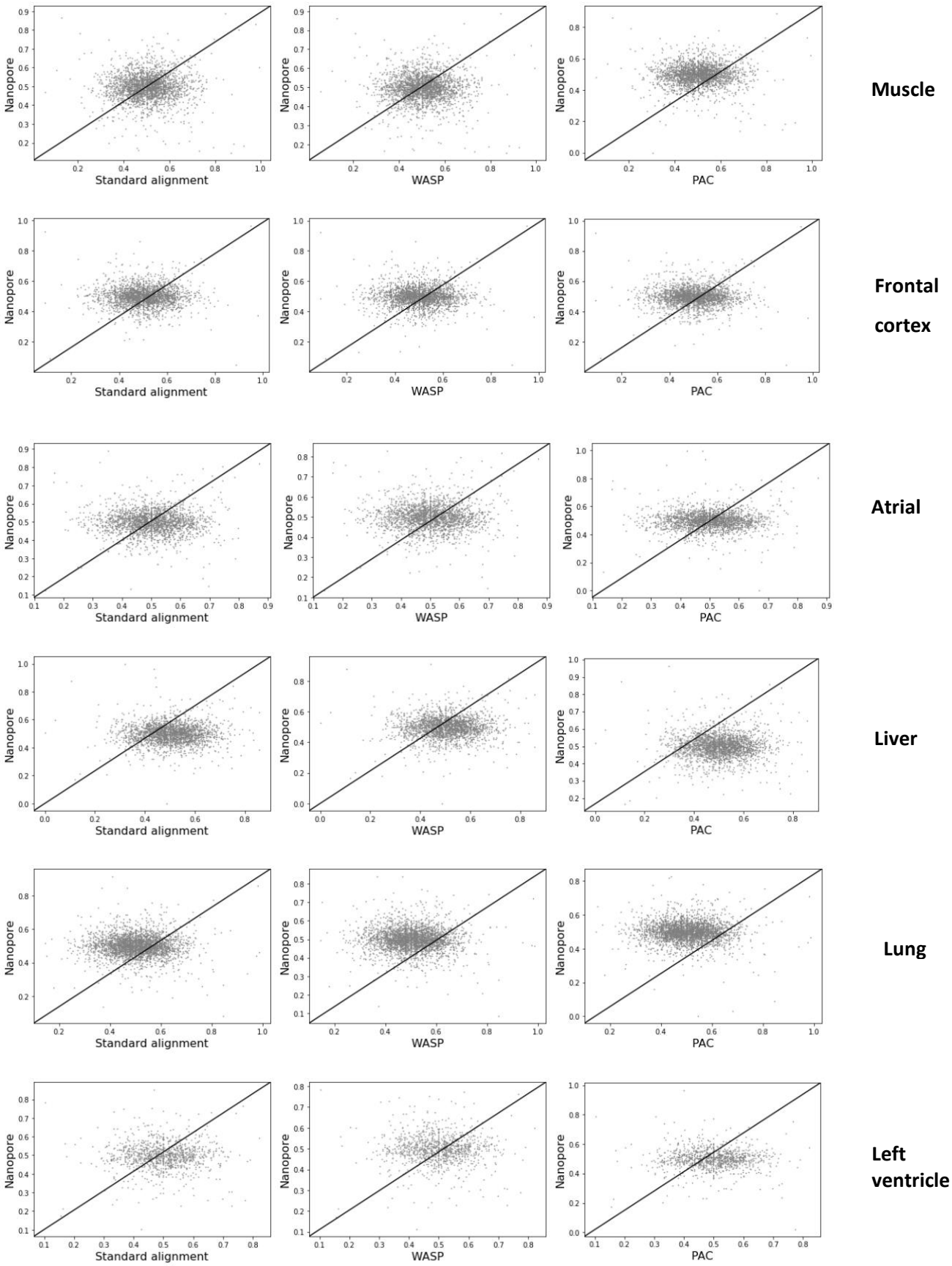


Figure 36. **Comparison of Nanopore and different analyses within each GTEx tissue.**
Data from short-read sequencing from any of the approaches or tissues did not correlate with long-read sequencing data.

4.4.3 ACCURACY OF ENRICHMENT OF ASE GENES

ASE is widespread even in healthy individuals [108], and I have shown in section 2.3.1.3 that biological enrichment terms can be obtained for genes under ASE. Because gene set enrichment analysis is a common method to analyse a gene set, in this section I set to examine how accurate detection of allelic imbalance can influence downstream analysis of biological processes. For this, I used simulated genomic data generated in Chapter 2 to test if genes under ASE are enriched for any particular biological terms. I did not expect to observe any enriched categories as the simulated data is based on a healthy individual, however, this allowed me to obtain a baseline truth against which I could compare how differences in ASE detection in other methods influenced the enriched terms.

I annotated heterozygous variants present in the simulated RNA-seq data for maternal and paternal genomes with wANNOVAR (https://wannovar.wglab.org). The variant with the highest coverage per gene acted as a representative for that gene. I calculated which genes showed significant ASE for each highest expressing variant with a binomial test $P<0.05$. I used g:Profiler2 to obtain genes and pathways that were enriched for ASE selecting KEGG pathway and GO terms (Figure 37). This analysis resulted in a list of enriched biological terms that were generated from genes showing significant ASE in the ground truth simulated data.

I then examined if ASE genes generated via PAC, standard alignment, and WASP-filtered alignment would be enriched for the same biological signals, or whether biases would lead to false biological interpretations of the data. I performed the same analysis for data obtained by PAC, standard alignment to the reference genome and WASP-filtered alignment using the same simulated RNA-seq data. I compared how these methods performed against enrichment in the ground truth (Figure 37).

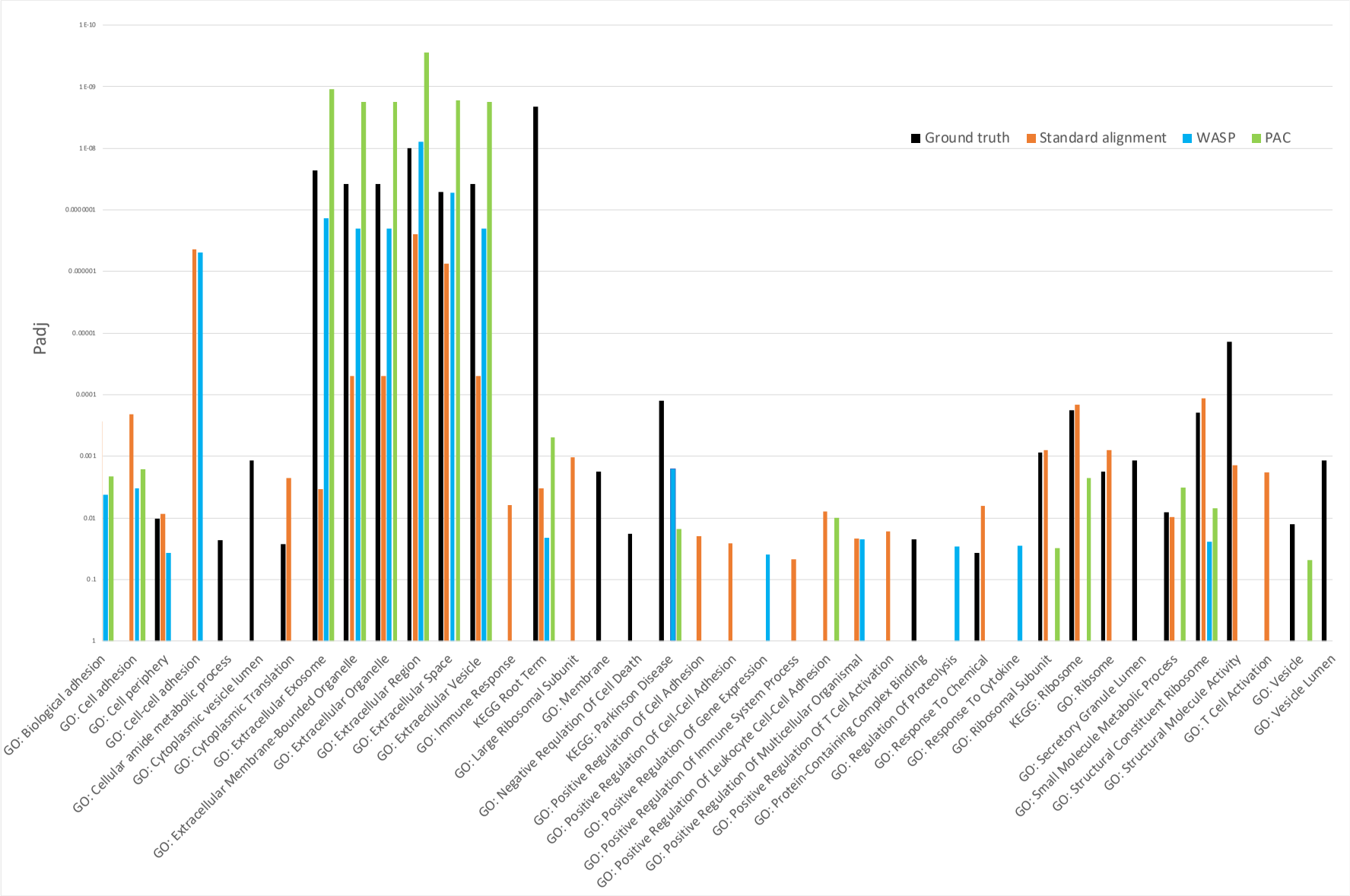


Figure 37. **Enrichment analysis of genes under ASE in the ground truth data, data obtained from standard alignment, WASP and PAC methods.**
Gene enrichment analysis was performed on genes under ASE (binomial test P<0.05) from ground truth (black) simulated data (see Chapter 2). These results were compared to the enrichment analysis of genes under ASE that were obtained from standard alignment (orange), WASP-filtered alignment (blue) and PAC (green) analysis. Padj represents G:Profiler 2 multiple testing correction by a tailor-made algorithm within the method.

Figure 37 shows that ground truth data returned 25 terms. One of these is Parkinson's disease term which could be of interest in genetic analysis.

In standard alignment, there are 28 terms in total. It wrongly picks up 12 terms and correctly assigns only 16 terms out of 25. In WASP analysis there are 17 terms in total. WASP picked 7 terms incorrectly, and correctly assigns 10 out of 25 terms. WASP picked one term PAC missed. In PAC analysis there are 16 terms in total. It has 3 incorrect terms, but it correctly assigns 13 terms out of 25. PAC picked up 4 terms that WASP missed.

Overall, PAC best recapitulates ground truth data. Both WASP-filtered alignment and standard alignment have more terms than PAC, yet have more incorrect terms. False positives are problematic for downstream biological analysis.

4.4.4 ENHANCER ANALYSIS

ASE measures allelic imbalances within a coding region, which are most often caused by *cis*-regulatory variants rather than the SNP differentiating the allele itself [288]. As genetic risk variants are enriched in enhancer regions [287, 291, 292], I hypothesised that it is possible that genes under ASE would be enriched for regulatory sequences in the surrounding regions, thus increasing the likelihood of a variant falling within an enhancer region that subsequently influences gene expression in *cis* (Figure 38).

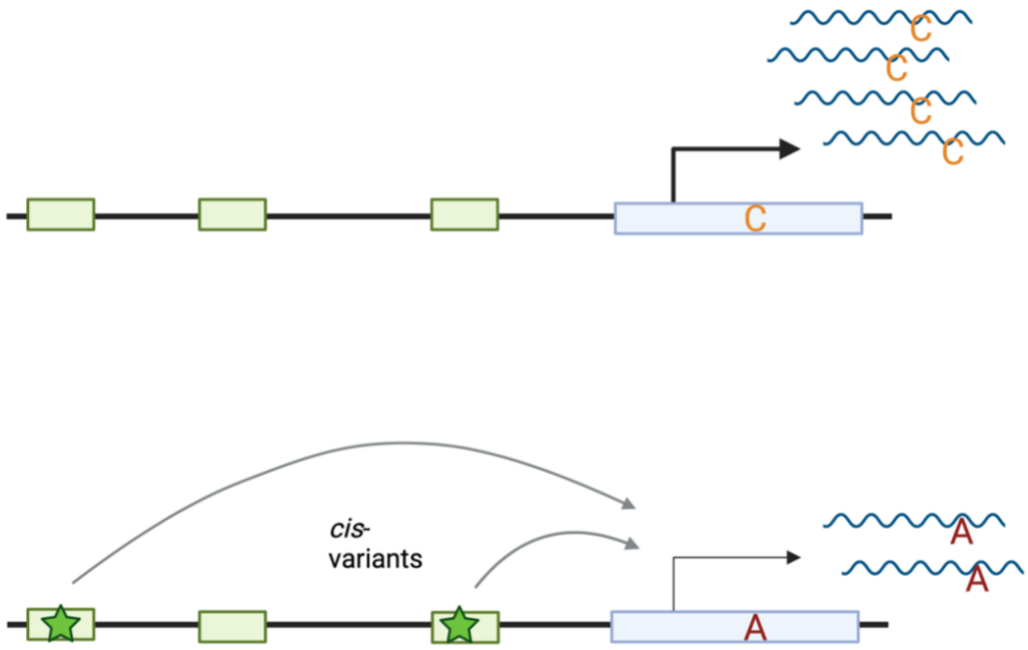


Figure 38. **Enhancer abundance analysis near genes under ASE.**
Genes under ASE are potentially enriched for enhancers and other regulatory regions that influence gene expression. The variants within regulatory regions (green boxes) accumulate and affect the gene expression of the gene (blue box) downstream or upstream. Created with BioRender.com.

I tested this hypothesis by comparing the abundance of enhancer regions near genes under ASE to non-ASE genes. I ran PAC with RNA-seq reads from iPSCs from the 10 healthy HipSci donors (see methods, section 2.2.1.1). I then obtained a list of genes that were under ASE using a binomial test (for more stringent criteria I used Bonferroni-adjustment, where the P-value is divided by the number of heterozygous sites), and genes that did not meet the ASE threshold. I then compared the average number of nucleotides within active enhancer regions (genomic locations obtained from NIH Roadmap Epigenomics Consortium) surrounding ASE and non-ASE genes when all the donor data was pooled.

When ASE detection was stringent with Bonferroni-adjustment, enhancer regions were significantly more abundant around ASE genes versus non-ASE genes (Wilcoxon signed rank test: $P=0.04883$). However, under less stringent ASE detection (binomial test at $P<0.05$) there was no significant difference between ASE and non-ASE genes (Wilcoxon signed rank test: $P= 0.375$). This might reflect that the relaxed ASE selection criteria included non-ASE genes in the list.

During comparisons within individuals, only 3 donors showed a significant difference in the total number of bases in active enhancer regions per gene when ASE detection was Bonferroni-adjusted (Wilcoxon rank sum test: $P=0.01921$, $P=0.04515$, $P=0.02059$). No statistical difference was found when ASE sites were detected at $P<0.05$ (Wilcoxon signed rank test: $P\geq0.05$).

4.5 DISCUSSION

In this chapter, I tested the accuracy of PAC in quantifying allelic expression using a series of population-based approaches and alternative datasets. First, I applied PAC to population-level gene expression data to demonstrate that it recapitulates regulation signals better than other commonly used methods. Second, I tested how PAC and other methods compare in their allelic ratios to that of nanopore sequencing. Third, I demonstrate that correct ASE detection is crucial for downstream analysis with ASE enrichment analysis. Fourth, I show that ASE can be used to answer biological questions such as demonstrating that genes under ASE are enriched for enhancer regions.

4.5.1 COMPARISON OF ASE TO POPULATION-LEVEL DATA

In the initial analysis, I compared how well aFC associated with ASE correlates with eQTL aFC. eQTL mapping is a more robust method that relies on population-level data, where errors are thought to average out across samples. Because ASE is prone to artefacts, improved ASE detection should better recapitulate the eQTL signals [108]. I show that PAC improves this correlation but also accurately quantifies aFC for extra genes that other methods lack the power to analyse. The correlation between ASE and eQTL aFC is only slightly improved in WASP-filtered alignment compared to the standard alignment, however, WASP-filtering removes a large number of genes that I have shown can be quantified accurately and therefore WASP-filtering most likely removes informative reads. The ability of PAC to accurately quantify allelic expression for a larger number of genes may be particularly important in analyses with small sample sizes such as in the rare disease field, or where other analysis approaches may cause important genes to be missed.

4.5.2 LONG-READ SEQUENCING

I next explored the validation of ASE with nanopore data and in particular if PAC can better replicate long-read ASE detection. Although PAC slightly improved the correlation of allelic

ratios at heterozygous sites with the nanopore data, it was not significant, and in general, ASE signals in short and long-read data do not seem to be consistent for these samples. Similar to my findings, there was only moderate concordance between ASE reported in short-read versus long-read sequencing in the original study [284]. The authors explained this via low read depth and some reads being filtered out at the quality control step. It has also been demonstrated that replicate RNA-seq libraries are needed in ASE analysis to reliably quantify the technical noise [293], and as such, repeat sequencing of samples here (for long and short-read sequencing) may also show high levels of discordance. Furthermore, at present long-read sequencing still suffers from a relatively high nucleotide error rate [285] with most errors being caused by indels [84], which might explain some of the unexpected results. Long-read sequencing is a rapidly evolving field, and the error rate is reducing and error correction methods improving [285]. It will be important to analyse genomic regions that are difficult to sequence with short-read sequencing including repeat sequences, around rare variants and complex chromosomal rearrangements [294], and in the future it is hoped that long-read sequencing will ease the study of ASE in cancer samples, which often have large mutation rates exacerbating biases even further.

4.5.3 THE ENRICHMENT OF ASE GENES

Next, I explored the consequences for downstream biological interpretation of improper ASE analysis. First, I show that the ASE analysis can be used to detect genes and biological terms that are enriched for genetic regulation using simulated ground truth data. Then, I show that PAC has the least number of false positive terms (18.7% (3/16)) out of all other analyses. The standard alignment had the most terms correct, however, it also had 42.9% (12/28) false positive terms. Surprisingly, WASP had the least number of correctly assigned terms but also a high number of false positives (41.2% (7/17)).

Although this analysis was performed on simulated data in order to assess the performance of each analysis to the ground truth, it demonstrates that analysis with real data may lead to a false biological interpretation of the data. In this particular case, standard alignment

would have missed the Parkinson's disease term, which may have been of particular interest to disease researchers.

4.5.4 ABUNDANCE OF ENHANCERS NEAR ASE GENES

Finally, I applied ASE to explore the potential mechanisms driving variation in gene expression across samples. Because ASE signals are often driven by *cis*- variants in regulatory regions, rather than the proxy SNP in the gene itself, and because each gene is often regulated by multiple regulatory regions, I investigated if enhancers are more abundant near genes under ASE. I show that enhancer regions are indeed enriched near genes under ASE, as expected under the most stringent criteria. This demonstrates the utility of ASE to answer biological questions and to understand underlying mechanisms.

In this chapter, I performed a series of validation exercises for the PAC derived data, and began to explore the application of PAC to population-level questions. In the next chapter, I will focus on applying ASE and PAC in disease and functional contexts to better understand underlying biological mechanisms.

CHAPTER 5 – APPLYING PAC TO DISEASE CONTEXT

5.1 INTRODUCTION

5.1.1 DETECTION OF DISEASE GENES

In clinical settings, genetic diseases are most commonly studied by targeted gene sequencing or whole exome sequencing, which are able to detect defects in protein-coding regions [295]. In total, only around 25-50% of patients with a rare disease [296-298] or Mendelian disorder [38] have a causal genetic variant identified with WES, due to the ambiguity around potential causal genes and the accuracy of genome annotation, and as such many genetic disorders remain undiagnosed [37]. In addition, an understanding of the functional molecular interpretation and clinical impact is lacking. This is now being tackled in some contexts by including the non-coding genome and functional genomic information, such as disruptions in gene expression, into the diagnostic pathway.

RNA-seq can provide information on changes in gene expression that are caused by genetic changes that might affect regulatory regions, regulatory proteins, or splice sites. Differential gene expression, where transcript levels differ between healthy and disease states [299], has been shown to be a powerful tool in understanding the biology behind many processes including psychiatric disorders [300], neurodegenerative disorders [301] and cancer [302]. Furthermore, RNA-seq has been used to validate splice-altering mutations, which enabled diagnosis in 66% of patients in one particular cohort, while only 21% of patients had received a diagnosis via WES/WGS [37]. In another study, 10% of patients where WES had failed to provide a diagnosis for a suspected mitochondrial disorder, received a diagnosis using RNA-seq and provided candidate causal genes for the rest of the patients [38]. Therefore, RNA-seq is a powerful tool to understand the biological mechanisms underlying disease states.

In general, genes that show differential gene expression between cases and controls are thought to reveal genes that are induced by the disease state rather than being causal of the disease itself [303]. As such, although differentially expressed genes may be useful as biomarkers in diagnostic settings (eg. [304, 305]), to understand the causal mechanisms additional information is often needed. Given that GWAS hits are commonly found in non-

coding regions of the genome, and have been shown to overlap genetic signals associated with eQTLs [71, 72, 306], variation in gene expression is likely to be the intermediate between genomic variants and phenotypes in many cases, often providing a link to the potential causal gene. However, as discussed in previous chapters, eQTL analysis is limited by several features, the most relevant of which include the inability to detect associations in smaller sample sizes and the inability to identify the correct causal variant with certainty due to linkage disequilibrium [307]. ASE offers an alternative route to disentangle these problems, and provides an additional layer of information on the role of gene expression that can be used to understand the disease and other biological processes. In this chapter, I explore the utility of ASE via two examples to better understand underlying biological processes: haploinsufficiency and gene-by-environment (G×E) interactions.

5.1.2 ASE AND HAPLOINSUFFICIENCY

Most genes have two functional copies and can tolerate a decrease in gene dosage (they are haplosufficient (HS)) [308]. Should one allele be expressed at a lower level or not expressed at all, there are often regulatory mechanisms in place to compensate for this change from the other allele to maintain expression levels required for normal gene function. Haploinsufficient (HIS) genes cannot tolerate reduced gene dosage, and deletion will cause an abnormal phenotype or disease state [309]. Similarly, there are genes where an additional copy of the gene is also not tolerated [310]. For HIS genes, two functional alleles are essential for the wild-type phenotype state. Haploinsufficient genes can be difficult to study and are often implicated in serious phenotypes including neurological disorders [311], intellectual disability [312], intellectual disability (eg. [313]), developmental [314] or metabolic disorders, or tumorigenesis. Experimental approaches to identify and verify these genes in humans are impossible due to the need to crossbreed and the severity of phenotypes. Because haploinsufficient genes are clinically relevant, there is a growing interest in utilising prediction methods to identify haploinsufficient genes to prioritise and interpret genetic variants. Current prediction methods exploit a magnitude of genomic features such as genomic conservation [315], haploinsufficiency in model organisms, mutation intolerance in population data [316], functional annotations [315], depletion of

variants [19] and epigenomic patterns [317]. These features are often incorporated into models comparing HIS to HS genes and it has been shown that many drastic differences are observed in genomic, evolutionary, functional, and network properties used for predictive models [308]. Because ASE measures allelic ratios within each sample and therefore quantifies the relative expression level of each allele at heterozygous sites, it could be an informative feature of gene dosage for use in haploinsufficiency research. In this chapter, I explore HIS genes and their allelic ratios, and hypothesise that in healthy individuals, allelic ratios of HIS genes are less likely to show deviation between the two alleles and also across individuals. So far ASE has not been utilised in HIS gene prediction models, and so this research opens new avenues to explore HIS prediction methods.

5.1.3 GENETIC VARIANTS AND G×E INTERACTIONS

In the second example, I explore ASE in the context of G×E interactions, to test how gene expression regulation changes in different environments, which can be highly relevant when trying to understand variation in transcriptional responses that may occur due to disease phenotypes. Different genes are expressed in different cell types and under different environmental stimuli, and genetic variation within individuals may affect how genes respond to these environmental changes. Large population-level studies have been informative in demonstrating the tissue specificity of gene expression [91, 318]. The GTEx consortium has shown that some eQTLs are widely shared between tissues, while some are highly tissue-specific [104]. To better understand how environmental stimuli affect gene expression, response eQTLs (reQTLs) can be utilised by treating cell cultures with different environmental treatments, which describe G×E interactions [319-321]. However, these studies miss about half of the genes associated with dynamic regulatory interactions, many of which have been implicated as disease genes; therefore, the cell type and environment play a crucial role [102]. ASE enables the study of G×E interactions by exposing different cell types to environmental stimuli [93]. Because ASE allows the study of gene expression variation on smaller sample sizes, a larger combination of environmental stimuli on more cell types can be investigated. Indeed, studies have demonstrated that 50% of genes under ASE showing G×E expression are involved in GWAS traits, significantly more than normal

genes under ASE [93]. The understanding of interactions between these features is only beginning to be investigated, and recent studies have begun to profile the transcriptional response of different cell types to different stimuli, in order to better understand the underlying causal mechanisms of the disease [102]. In this chapter, I explore how improved allelic quantification can better guide the interpretation of these results.

5.2 METHODS

5.2.1 DATA DESCRIPTION

5.2.1.1 HIS GENE LIST

To construct a HIS gene list I selected 299 known HIS genes [309] often used as a training set in many other papers that develop machine learning for haploinsufficiency prediction (eg. [308, 317]). I converted the Entrez ID to Ensembl ID with g:Profiler (https://biit.cs.ut.ee/gprofiler/convert). The converter was unable to convert 2 genes. I converted 1 gene manually and the other gene did not match any Ensembl ID. I also included 298 HIS genes from Han *et al.*, 2018 [317], used for model training. These were collected from Dang *et al.*, 2008 [309] and ClinVar. After removing duplicates, 357 HIS genes remained in total.

5.2.1.2 GREY HIS GENE LIST

There are multiple studies that have used various parameters to predict HIS genes. Some of these are genuine HIS genes that have not yet been validated, but some will be falsely classified as HIS. I called these genes 'grey' (not definite) HIS genes. To construct a grey HIS gene list (hereafter referred to as grey list), I obtained genes that were predicted to be HIS from previous publications.

1. From Han *et al.*, 2018 paper [317] I used 3406 genes with episcore >= 0.6 as predicted HIS genes. In this paper, researchers used epigenomics to predict haploinsufficiency.
2. From Shihab *et al.*, 2017 [315] I used 339 unique genes that were used as a benchmark to test their model: haploinsufficient genes in OMIM, haploinsufficient genes with *de novo* mutations in OMIM, genes where a heterozygous knockout mutation in mice causes lethality phenotypes, genes where a heterozygous knockout mutation in mice causes seizures, genes with *de novo* loss of function mutation in autism probands from Iossifov *et al*., 2012 [322], and genes with *de novo* loss of

function mutations in other sets of autism probands [323-325]. There were in total 339 genes after removing duplicates and 328 after converting to Ensembl ID.

3. From Huang *et al*., 2010 [308] I used the HI_prediction_with_imputation.bed file that contains probabilities for genes being haploinsufficient. Researchers used genomic, evolutionary, functional, and network properties to develop a model to predict HIS genes. I selected 2571 genes with >=85% probability of being haploinsufficient for the grey list. After converting to Ensembl IDs, 2596 genes remained.
4. From Lek *et al*., 2016 [19] I used 3231 genes with pLi score >0.9. Researchers used variant depletion to predict haploinsufficiency. After converting to Ensembl IDs, 3151 genes remained.
5. From Steinberg *et al.*, 2015 [326] I selected 5% of genes with the highest genome-wide haploinsufficiency (GHIS) score. The score was constructed using various features including gene co-expression and genetic variation. These were provided in Ensembl ID format.

After removing duplicates and those present in the HIS gene list, there were 7720 grey HIS genes.

5.2.1.3 HS GENE LIST

To generate the HS gene list, I obtained 574 HS genes used as a training set in Han *et al.*, 2018 [317]. They collected these genes from a paper that identified genes from healthy individuals where the copy of gene has been deleted [327].These were supplied in Ensembl ID format. I also used 386 HS genes used as a training set in Shihab *et al.*, 2017 [315]. These were collected from a paper that identified loss-of-function tolerant genes [129]. After converting to Ensembl ID with g:Progiler 350 genes remained. After removing duplicates 906 HS genes remained.

5.2.1.4 GTEX

I downloaded the haplotype expression matrix from GTEx Portal for all GTEx tissues that were obtained from WASP-filtered RNA-seq data (phASER_WASP_GTEx_v8_matrix.gw_phased.txt.gz). This contains the number of reference and alternative reads for each haplotype. I removed tissues where the number of individuals was below 100 (bladder, cervix and fallopian tube samples). I used the GTEx_Analysis_v8_Annotations_SampleAttributesDS.txt file to obtain tissue information for each sample in the haplotype expression matrix. I counted the reference allele ratio for genes where there were at least 20 individuals and the coverage for the gene was at least 20× per individual. I only used autosomal genes. I separated genes into those that were present in the HIS gene list, grey list, HS list and unknown (the rest).

5.2.2 GTEX HAPLOINSUFFICIENCY ANALYSIS

I compared the allelic ratios between HIS, grey, unknown and HS genes in all GTEx tissues that had over 100 samples. Initially, I compared genes from all tissues pooled together. There were 6,645 HIS genes; 13,030 HS genes; 142,109 grey genes and 241,837 unknown genes.

For haploinsufficiency analysis, I was interested in how large the standard deviation of the allelic ratios is across individuals, and how far the mean for allelic ratio was from 0.5. Therefore, I generated a flipped data set where when the reference allele ratio was over 0.5, I subtracted 1 from the ratio so that the value would always be 0.5 or below. For this, I measured the mean and standard deviation for each gene across individuals in all tissues. This was performed on HIS, grey, unknown and HS gene samples.

Following this, for each gene, I selected the tissue where it was expressed most highly, using the median gene transcripts per million (TPM) (GTEx_Analysis_2017-06-05_v8_RNASeQCv1.1.9_gene_tpm.gct.gz) across individuals for each gene. I plotted the reference allele ratio for each gene only from the highest expressing tissue.

The statistical significance for both of these analyses was calculated with a two-sided Mann-Whitney-Wilcoxon test with Bonferroni correction.

5.2.3 PAC AGAINST WASP AND STANDARD ALIGNMENT HAPLOINSUFFICIENCY ANALYSIS

I obtained a haplotype expression matrix with standard alignment of RNA-seq reads to the reference genome (phASER_GTEx_v8_matrix.gw_phased.txt.gz) and with WASP-filtered alignment (phASER_WASP_GTEx_v8_matrix.gw_phased.txt.gz) from the GTEx Portal. I selected samples from whole blood only. I downloaded aligned RNA-seq data for 670 whole blood samples from the GTEx Portal and converted these files back to raw FASTQ sequence files with SAMtools. These were then used as an input for PAC (selecting the GRCh38 reference genome), together with phased genetic variant calls from WGS data (obtained from the GTEx, phASER_GTEx_v8_merged.vcf.gz). I used the haplotype level data obtained from PAC for the HIS analysis.

I generated a random gene list which consisted of 5% genes from the GTEx haplotype expression matrix (2664 genes) from all tissues. 2663 genes were expressed in PAC, 2664 genes were expressed in WASP-filtered alignment and 2664 genes were expressed in standard alignment. I selected autosomal HIS, grey, HS and random gene sets that were expressed in at least 20 individuals with >=20× coverage. After filtering, PAC had 201 HIS, 4766 grey, 449 HS and 645 random genes; WASP-filtered data had 179 HIS, 4353 grey, 413 HS and 571 random genes; and standard alignment had 190 HIS, 4592 grey, 439 HS and 606 random genes.

I then compared the allelic ratios between gene lists in different methods and the statistical significance was calculated with a two-sided Mann-Whitney-Wilcoxon test with Bonferroni correction.

5.2.4 G×E ANALYSIS

To study G×E interactions, I obtained raw RNA-seq reads for cadmium-treated Lymphoblastoid Cell Lines (LCL), Induced Pluripotent Stem Cells (iPSC), and iPSC-derived cardiomyocytes from six individuals from the Sequence Read Archive (SRA) under BioProject PRJNA694697 [102]. I obtained the phased genetic variant calls from WGS within the 1000 Genomes Project [17]. Together these RNA-seq reads and phased variants were submitted into PAC (selecting the GRCh38 reference genome). I used the site-level allele counts obtained from the PAC analysis. I compared the results from PAC to the standard alignment data that were also obtained from PAC. PAC generates alignment to the reference genome with STAR filtered for properly paired and uniquely mapped reads.

For each individual and in all cell types, I selected sites with at least 20× coverage in PAC and standard alignment. I then identified heterozygous sites under significant ASE in cardiomyocytes using a binomial test ($P<0.05/18,537$ tests, which is the mean number of sites tested per sample across all methods and individuals, and ensures significance thresholds are comparable across methods), but were not under ASE in LCLs or iPSCs ($P>0.05$ in these cases). I then compared genes containing the significant ASE site to genes previously identified as potentially playing a role in coronary artery disease [328].

5.3 RESULTS

5.3.1 HAPLOINUSFFICIENCY

5.3.1.1 DETECTING HAPLOINSUFFICIENT GENES ACROSS GTEX TISSUES

HIS genes do not tolerate decrease in gene dosage, and it would be expected that their allelic ratios are less likely to tolerate changes from 1:1. If some changes from this ratio can be tolerated, it would be further expected that allelic ratios should not vary across healthy individuals. These features offer an excellent opportunity to utilise ASE analysis to potentially predict HIS genes. These genes are clinically relevant yet often difficult to identify and study. The current prediction methods incorporate multiple features; however, ASE has not been applied to this field yet. Therefore, I explored the utility of ASE to identify haploinsufficient genes.

In a healthy individual, HIS genes do not tolerate disturbances in gene dosage. Therefore, one may expect the allelic ratios at heterozygous sites within HIS genes to be restricted in one of two ways:

1) ASE may be limited overall and the reference allele ratio should be close to 50% in all individuals.
2) ASE may be limited in variance, and the reference allele ratio is expected to have low standard deviation across individuals, but not necessarily be fixed around a mean of 50%.

$\Rightarrow$ Under these assumption models, I expect a mean reference allele ratio closer to 50%, and a standard deviation in the reference allele ratio that is lower, in HIS genes when compared to HS genes.

To test these models, I first compared the mean allelic ratios and standard deviation of the allelic ratios in genes that are expressed in all GTEx tissues that had data available for over 100 individuals in each case. I calculated the reference allele ratio for each gene in WASP-filtered alignment. If the ratio was above 0.5, I subtracted 1 from the ratio to obtain the absolute distance from 1:1 ratio. I then plotted the mean (Figure 39) and standard deviation (Figure 40) of the reference allele ratio for known HIS genes, genes predicted to be HIS

(grey), known HS genes and the rest of the genes (unknown) expressed in GTEx tissues, and compared these features between groups. The unknown gene list will contain yet-to-be-discovered HIS genes and HS genes. Because most human genes are HS [308], these are expected to be the majority.

As expected, the mean (P-value=1.202×10^{-79}, U-stat=5.045×10^{7}) was significantly closer to 0.5 and standard deviation (SD) (P-value=1.008×10^{-49}, U-stat=3.766×10^{7}) was significantly smaller in HIS genes compared HS genes; also compared to unknown genes (mean: P-value=7.494×10^{-237}, U-stat=9.934×10^{8} and SD: P-value=7.763×10^{-161}, U-stat=6.472×10^{8}), perhaps again reinforcing the idea that the majority of genes with unknown classification are likely to be HS.

The mean (P-value=1.000, U-stat=4.713×10^{8}) was not significantly different between HIS and grey genes, however, SD of allelic ratio was significantly smaller in HIS genes than grey genes (P-value=4.355×10^{-02}, U-stat=4.813×10^{8}). This reflects that the prediction methods that generated the grey list likely accurately assign genes as HIS, but also that potentially some predicted HIS genes are not genuine HIS genes as the standard deviation differs between the groups.

The mean was significantly closer to 0.5 and SD was significantly smaller in the grey gene list than HS genes (mean: P-value=2.616×10^{-218}, U-stat=1.080×10^{9}; and SD: P-value=1.066×10^{-169}, U_stat=7.897×10^{8}), and unknown genes (mean: P-value=0.000, U_stat=2.126×10^{10}; and SD: P-value=0.000, U-stat=1.358×10^{10}).

The mean was also significantly closer to 0.5 (P-value=1.307×10^{-29}, U-stat=1.669×10^{9}) and SD (P-value=1.721×10^{-23}, U-stat=1.492×10^{9}) was significantly smaller in HS than in unknown genes.

These results suggest that there may be some predictive power in ASE for predicting HIS genes because these results support the hypothesis that HIS genes have a significantly smaller mean allelic ratio and standard deviation in their allelic ratios than known HS genes. These results also show that ASE analysis can differentiate unknown genes from HIS genes in terms of mean and standard deviation of allelic ratios, the majority of which are expected

to be HS as these are by large the most common genes, potentially indicating HIS predictive potential.

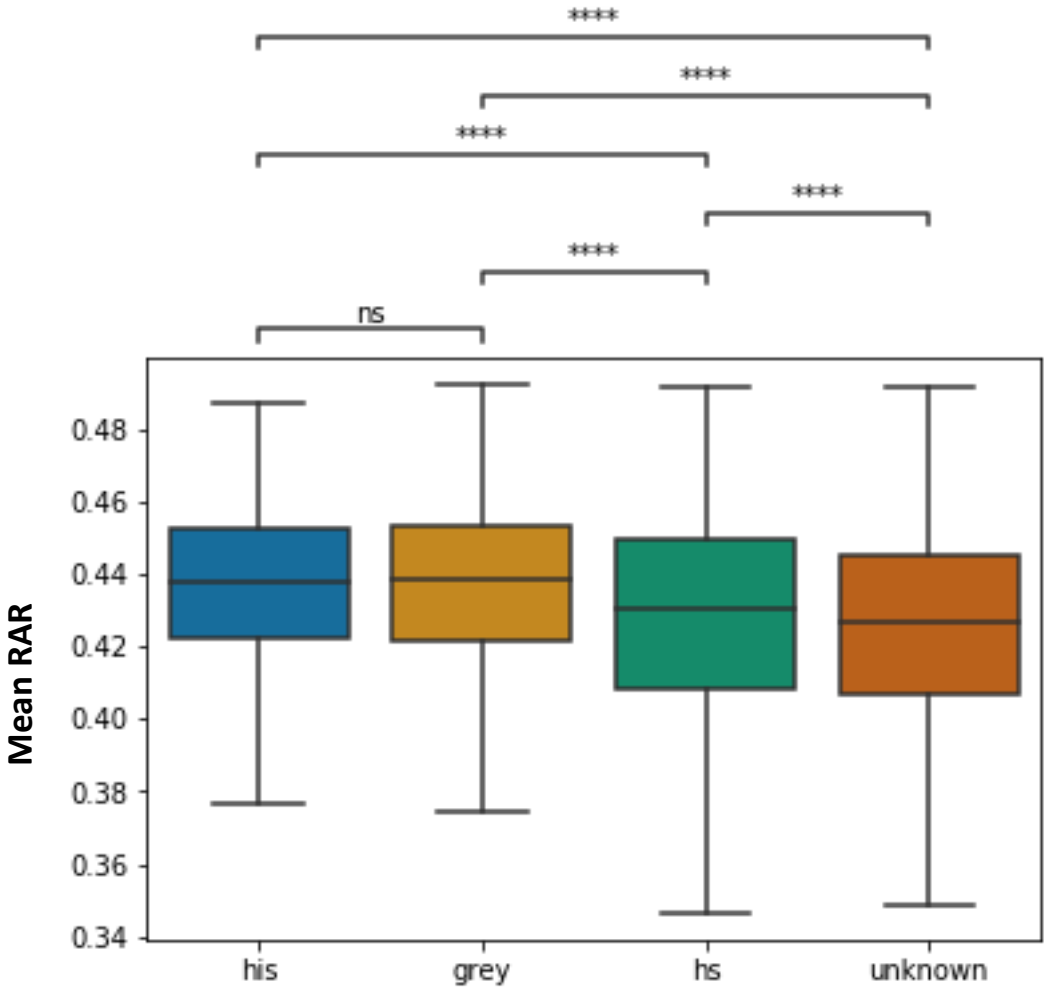


Figure 39. **The mean reference allele ratio (RAR) across individuals per gene in all GTEx tissues to detect HIS genes.** The mean reference allele ratio (RAR) was significantly closer to 0.5 in haploinsufficient (HIS) or predicted haploinsufficient (grey) and haplosufficient (HS) and genes of the unknown haploinsufficiency status (unknown), pointing to the power of ASE to detect haploinsufficiency. The mean RAR was not significantly different between HIS and grey genes, suggesting that the existing HIS prediction tools might be able to detect HIS genes. P-value annotation legend: ns: $5.00\times10^{-2} < p <= 1.00$, *: $1.00\times10^{-2} < p <= 5.00\times10^{-2}$, **: $1.00\times10^{-3} < p <= 1.00\times10^{-2}$, ***: $1.00\times10^{-4} < p <= 1.00\times10^{-3}$, ****: $p <= 1.00\times10^{-4}$.

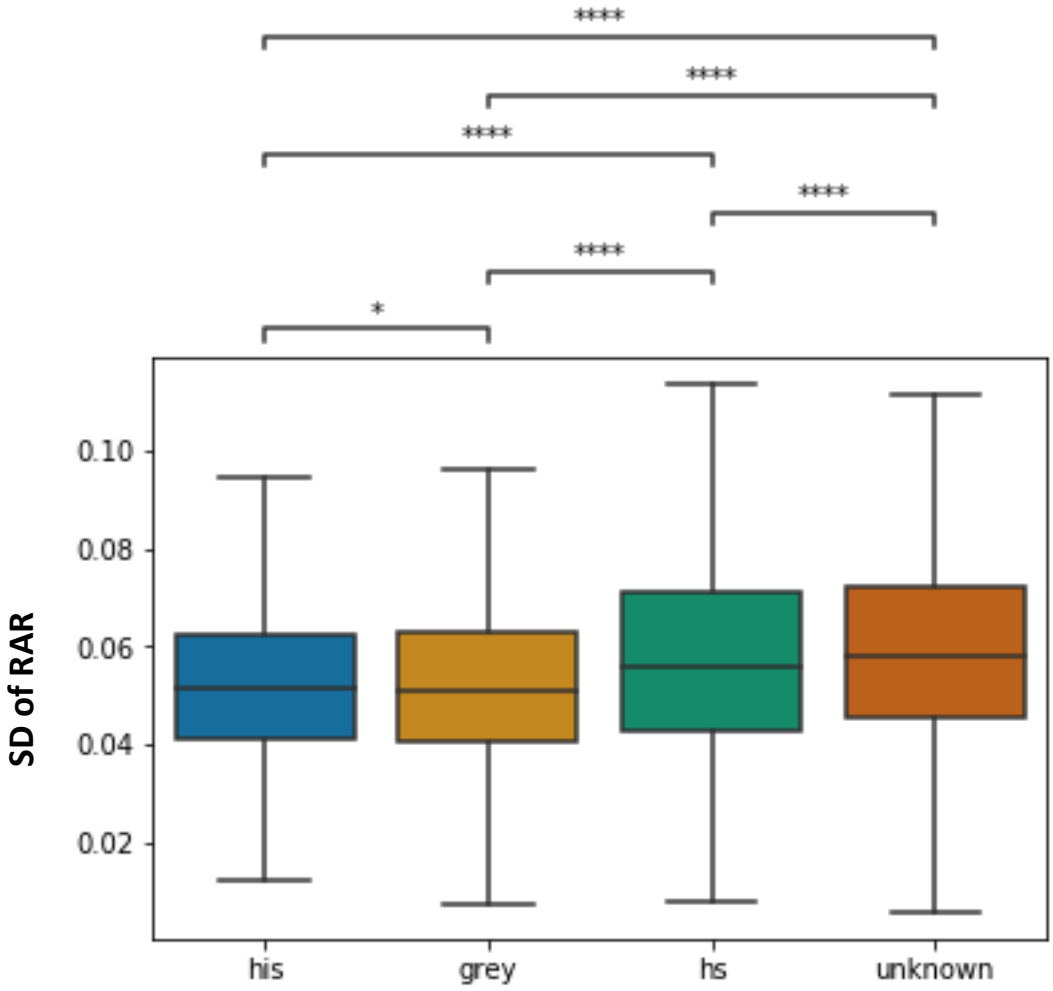


Figure 40. **The standard deviation of reference allele ratio (RAR) across individuals per gene in all GTEx tissues to detect HIS genes.**
The standard deviation (SD) of reference allele ratio (RAR)was significantly smaller in haploinsufficient (HIS) and haplosufficient (HS), predicted haploinsufficient (grey) and genes of the unknown haploinsufficiency status (unknown), suggesting ASE could potentially be used to differentiate HIS genes. P-value annotation legend: ns: $5.00\times10^{-2} < p <= 1.00$, *: $1.00\times10^{-2} < p <= 5.00\times10^{-2}$, **: $1.00\times10^{-3} < p <= 1.00\times10^{-2}$, ***: $1.00\times10^{-4} < p <= 1.00\times10^{-3}$, ****: $p <= 1.00\times10^{-4}$.

5.3.1.2 DETECTING HAPLOINSUFFICIENT GENES FROM HIGHEST EXPRESSING TISSUES

I then focused on tissues where the gene was most highly expressed, as this is probably more likely to select the tissues where the gene is most relevant and therefore may affect important downstream biological processes.

To do this, for each gene, I selected the tissue where the median TPM across all GTEx individuals for that particular gene was highest. I calculated the reference allele ratio and flipped values above 0.5 as previously described. The mean and standard deviation of genes in the highest expressing tissues is presented in Figure 41 and 42, respectively.

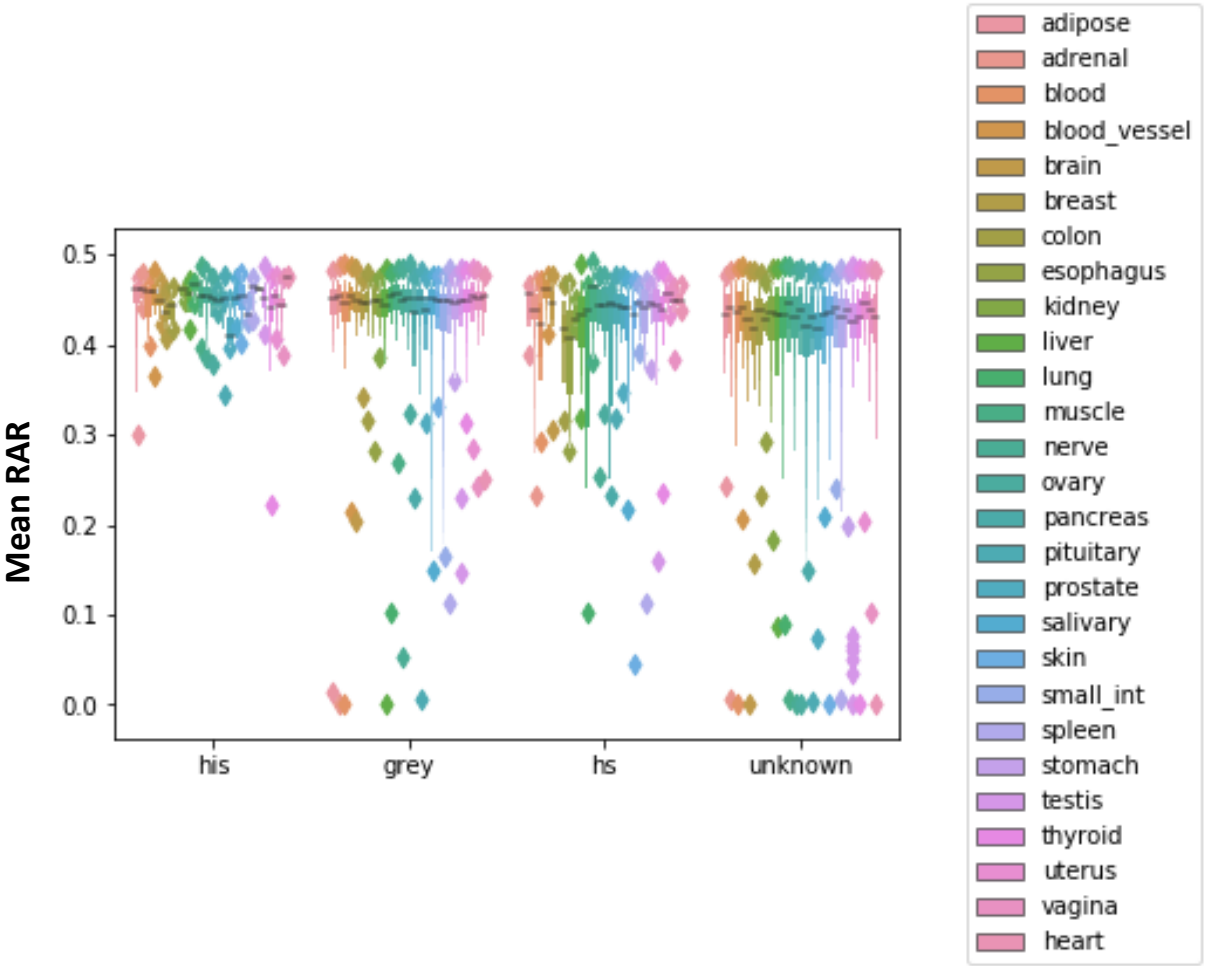


Figure 41**. The mean reference allele ratio (RAR) across individuals per gene in different GTEx tissues.**
The mean reference allele ratio (RAR) appears to be closer to 0.5 in haploinsufficient (HIS) genes. Similarly, the predicted haploinsufficient (grey) gene list appears to have the majority of mean values close to 0.5, with some variation. haplosufficient (HS) and genes of the unknown haploinsufficiency status (unknown) deviate slightly in their mean, with more variation than in other gene lists.

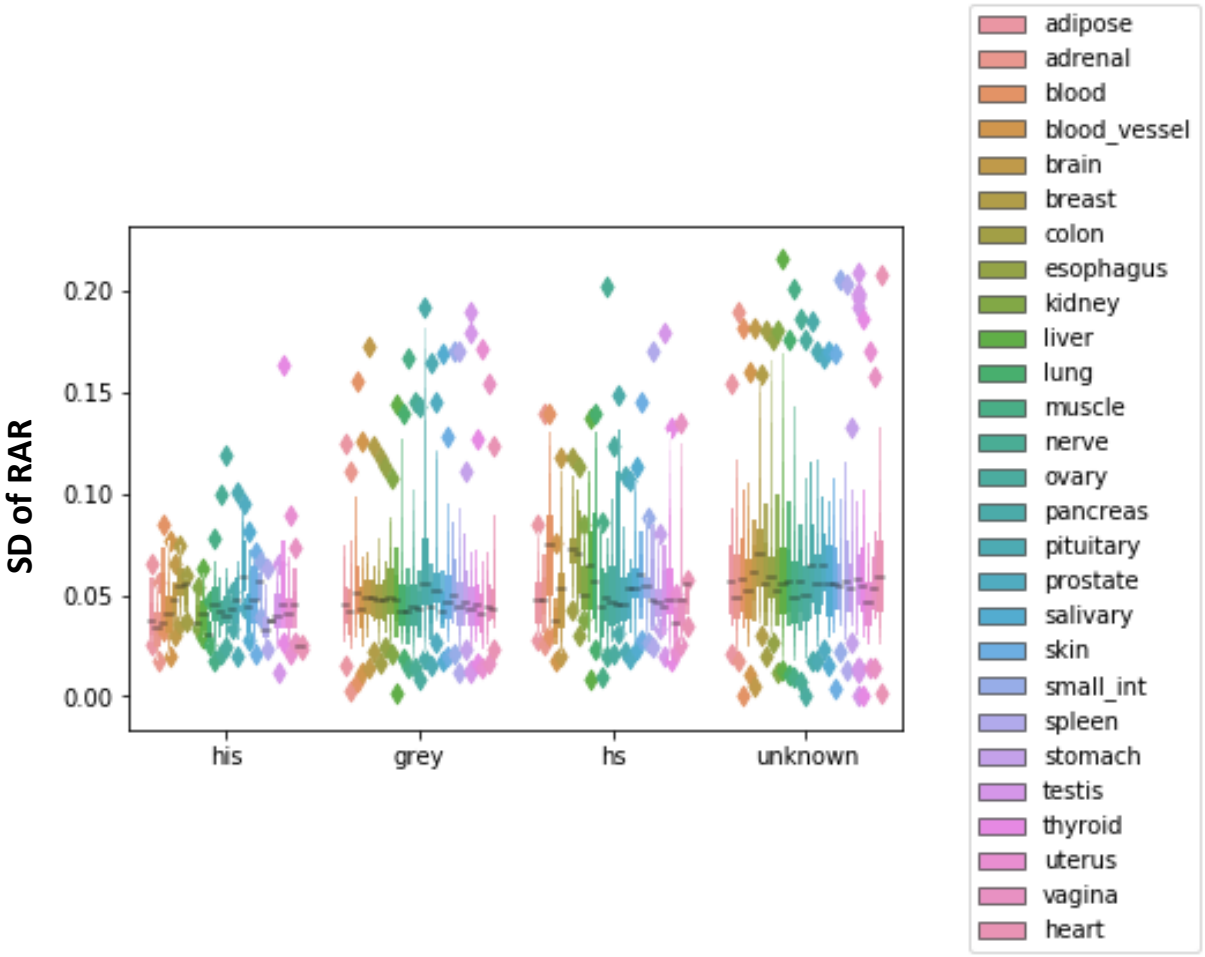


Figure 42. **The standard deviation of reference allele ratio (RAR) across individuals per gene in different GTEx tissues.** The standard deviation (SD) of reference allele ratio (RAR) appears to be closer to 0 in haploinsufficient (HIS) genes. The predicted haploinsufficient (grey) gene list appears to have the majority of SD values close to 0, with some variation. haplosufficient (HS) and genes of the unknown haploinsufficiency status (unknown) deviate slightly more in their SD, with more variation than in other gene lists.

I then combined genes within HIS, grey, HS and unknown gene lists, and plotted the mean (Figure 43) and standard deviation (Figure 44). There was no significant difference between HIS and grey genes in the mean (P-value=8.105×10^{-2}, U-stat=1.149×10^{6}) or SD (P-value=3.235×10^{-1}, U-stat=9.956×10^{5}) of allelic ratios, further suggesting that these two groups may contain a similar proportion of HIS genes.

The mean was significantly closer to 0.5 (P-value=2.667×10^{-9}, U-stat=1.383×10^{5}) and the SD (P-value=5.452×10^{-7}, U-stat=8.814×10^{4}) of allelic ratios was significantly smaller in HIS than in HS genes. HIS genes also had a significantly closer mean to 0.5 (P-value=5.309×10^{-39}, U-stat=3.753×10^{6}) and a significantly smaller SD (P-value=7.941×10^{-32}, U-stat=1.616×10^{6}) in allelic ratios than unknown genes.

The mean was significantly closer to 0.5 and SD was significantly smaller in grey than in HS genes (mean: P-value=8.341×10^{-13}, U-stat=2.640×10^{6}; SD: P-value=7.697×10^{-10}, U-stat=1.918×10^{6}), and unknown genes (mean: P-value=0.000, U-stat=7.249×10^{7}; SD: P-value=0.000, U-stat=3.579×10^{7}).The same trend was seen for the mean (P-value=2.567×10^{-15}, U-stat=6.594×10^{6}) and standard deviations (P-value=1.134×10^{-11}, U-stat=4.692×10^{6}) in HS and unknown genes.

These results demonstrate that ASE can be exploited to differentiate between HIS and HS genes. The results also show that selecting tissue of the highest expression is not necessary to detect this signal.

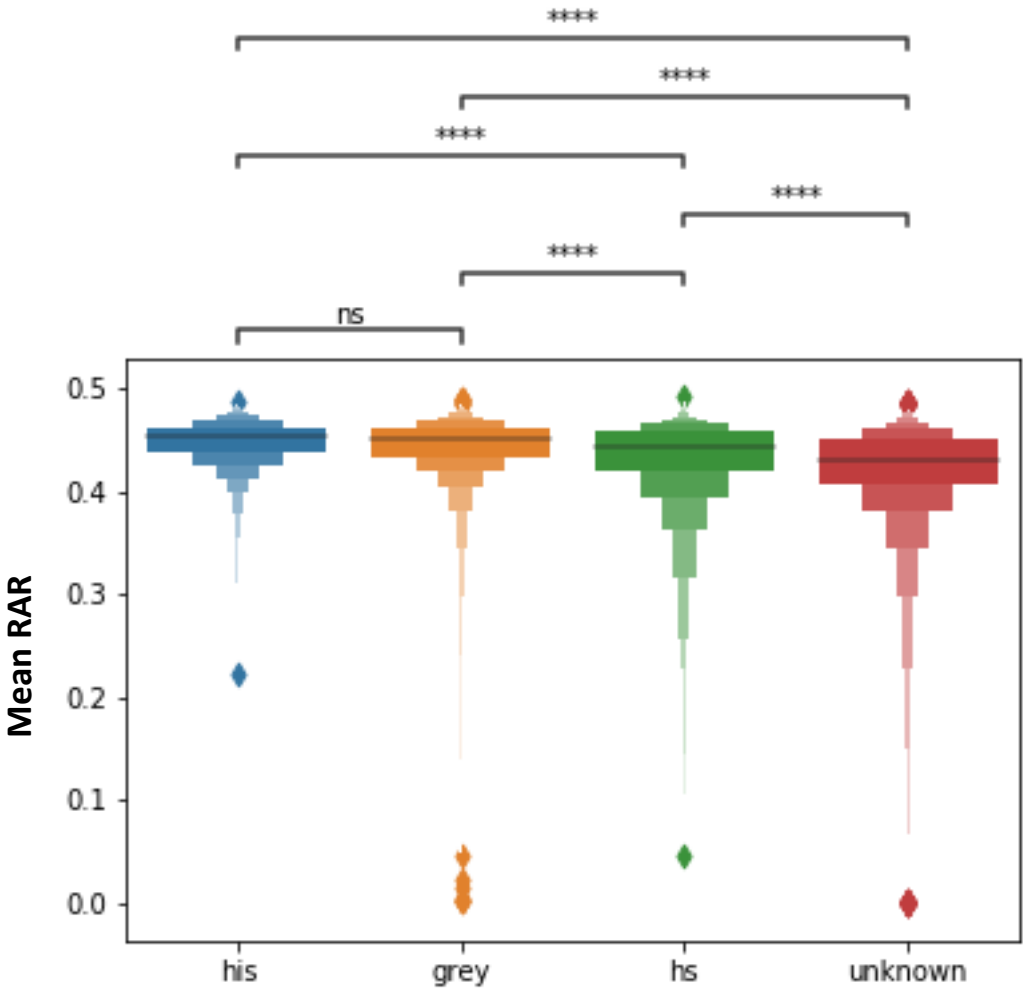


Figure 43. **The mean reference allele ratio (RAR) across individuals per gene to detect HIS genes in the highest expressing GTEx tissue.**
The mean reference allele ratio (RAR) was significantly closer to 0.5 in haploinsufficient (HIS) or predicted haploinsufficient (grey) and haplosufficient (HS) and genes of the unknown haploinsufficiency status (unknown) but not in grey genes. This again suggests that HIS and grey gene lists have similar proportions of HIS genes. The results also demonstrate ASE has the capability to differentiate HIS and HS genes in terms of mean RAR. P-value annotation legend: ns: $5.00\times10^{-2} < p <= 1.00$, *: $1.00\times10^{-2} < p <= 5.00\times10^{-2}$, **: $1.00\times10^{-3} < p <= 1.00\times10^{-2}$, ***: $1.00\times10^{-4} < p <= 1.00\times10^{-3}$, ****: $p <= 1.00\times10^{-4}$.

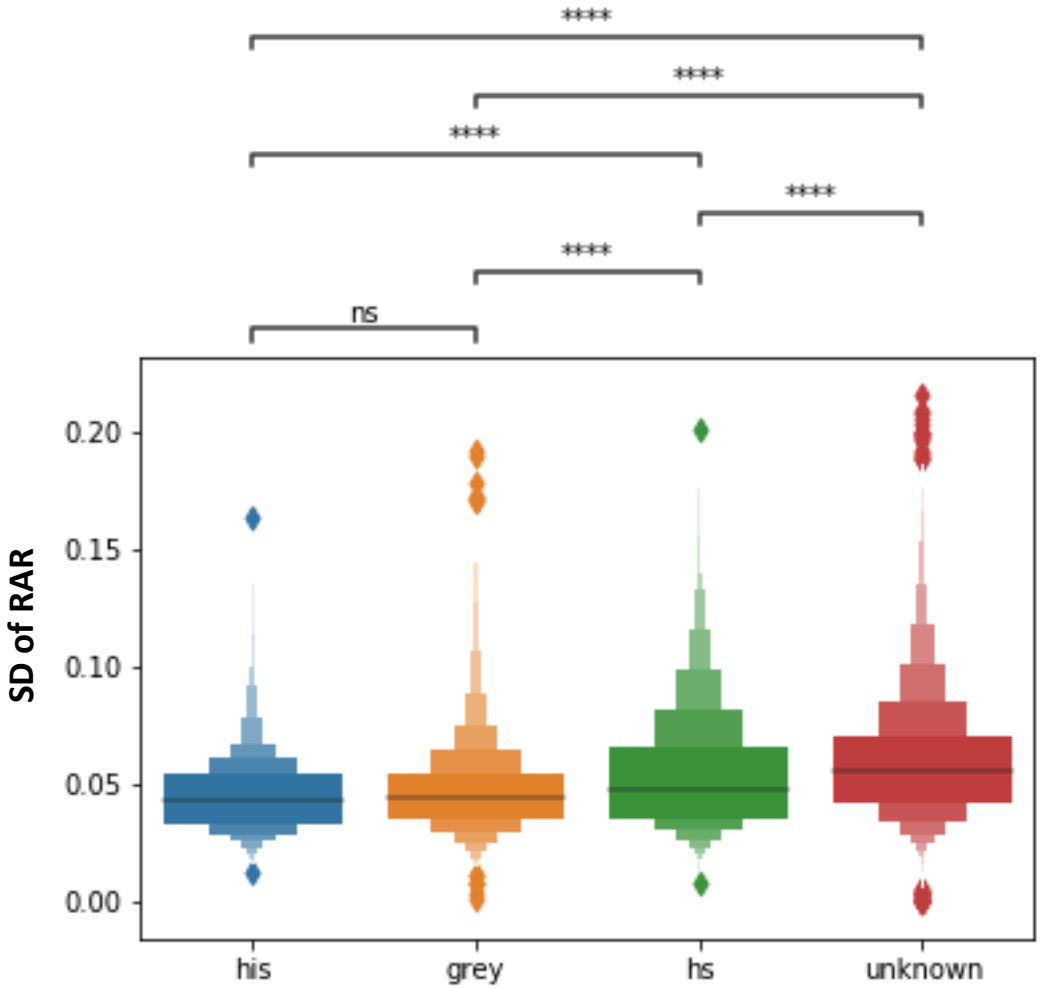


Figure 44. **The standard deviation of reference allele ratio (RAR) across individuals per gene to detect HIS genes in the highest expressing GTEx tissue.**
The standard deviation (SD) of reference allele ratio (RAR) was significantly smaller in haploinsufficient (HIS) or predicted haploinsufficient (grey) and haplosufficient (HS) and genes of the unknown haploinsufficiency status (unknown) demonstrating the power to differentiate HIS and HS genes. P-value annotation legend: ns: $5.00\times10^{-2} < p <= 1.00$, *: $1.00\times10^{-2} < p <= 5.00\times10^{-2}$, **: $1.00\times10^{-3} < p <= 1.00\times10^{-2}$, ***: $1.00\times10^{-4} < p <= 1.00\times10^{-3}$, ****: $p <= 1.00\times10^{-4}$.

5.3.1.3 COMPARING PAC TO OTHER METHODS IN HAPLOINSUFFICIENCY DETECTION

I compared how PAC performs in detecting HIS genes relative to WASP-filtered alignment and standard alignment. I obtained the gene expression matrix at the haplotype level from PAC for whole blood GTEx data for 670 individuals. I downloaded data for the same samples that were aligned to the reference genome (using a standard alignment approach) and WASP-filtered alignment from GTEx Portal. I calculated the reference allele ratios for HIS, grey, random and HS genes expressed in each method. Random gene sets consisted of 5% of the genes expressed in GTEx data for all tissues. This was used as a control gene set, although it is expected to consist mostly of HS genes, as these are the most common genes.

Overall, the improved detection of allelic ratios in PAC data translated to a stronger statistical significance for HIS detection and included more genes (see methods). When comparing the mean allelic ratio (Figure 45) and standard deviation (Figure 46) of allelic ratios across individuals, there was no significant difference between HIS and grey, random and HS, and HIS and random in any of the methods. Between HIS and HS, there was no statistical difference in standard deviation, however, the mean was significantly closer to 0.5 in HIS in all three methods (PAC: P-value=3.179×10^{-2}, U-stat=5.204×10^{4}; WASP: P-value=3.961×10^{-2}, U-stat=4.282×10^{4}; standard alignment: P-value=1.492×10^{-2}, U-stat=4.870×10^{4}). In all 3 methods grey genes were significantly closer to 0.5 in mean allelic ratios than random (PAC: P-value=1.705×10^{-6}, U-stat=1.736×10^{6}; WASP: P-value=2.094×10^{-4}, U-stat=1.383×10^{6}; standard alignment: P-value=2.084×10^{-4}, U-stat=1.544×10^{6}), and the same trend was seen when compared to HS genes (PAC: P-value=7.955×10^{-6}, U-stat=1.224×10^{6}; WASP: P-value=1.412×10^{-5}, U-stat=1.031×10^{6}; standard alignment: P-value=$1.018e^{-5}$, U-stat=1.153e+0^{6}); however PAC showed the strongest effect size.

The standard deviation was statistically smaller in grey than in random (PAC: P-value=1.265×10^{-3}, U-stat=1.389×10^{6}; WASP: P-value=3.888×10^{-2}, U-stat=1.145×10^{6}; standard alignment: P-value=2.672×10^{-2}, U-stat=1.281×10^{6}), and similarly when compared to the HS genes (PAC: P-value=5.133×10^{-3}, U-stat=9.593×10^{5}; WASP: P-value=2.949×10^{-2}, U-stat=8.147×10^{5}; standard alignment: P-value=1.475×10^{-2}, U-stat=9.107×10^{5}). As such, this work highlights how more accurate quantification of allelic reads increases the power to detect important biological signals in RNA sequencing data.

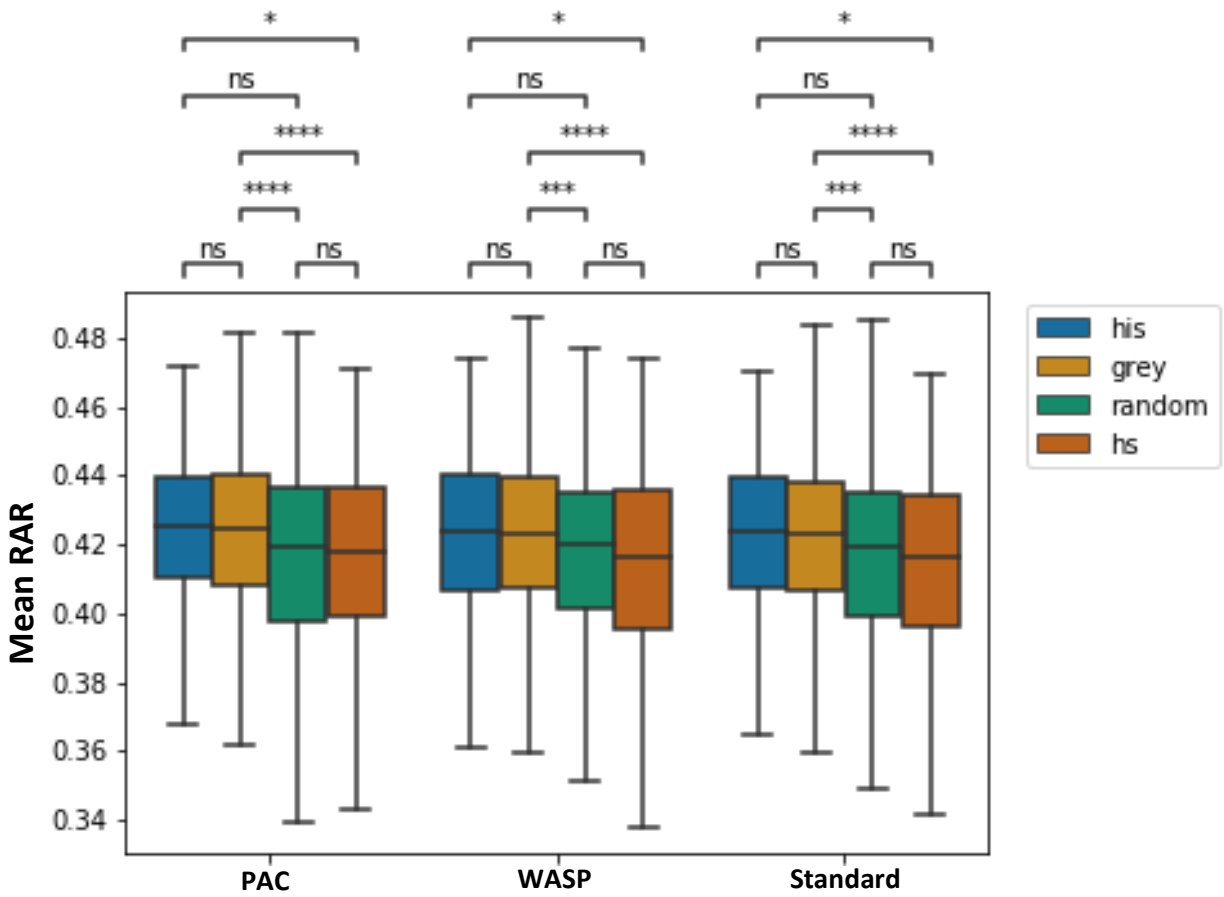


Figure 45. **The mean reference allele ratio (RAR) across individuals per gene to detect HIS gene in PAC, WASP-filtered alignment and standard alignment.**
All methods are significantly closer to 0.5 in mean reference allele ratio (RAR) when comparing haploinsufficient (HIS) and haplosufficient (HS), predicted haploinsufficient (grey) and random and grey and HS. The latter two exhibited a stronger effect size in PAC. Genes on autosomes with >=20× coverage and present in all methods are shown. P-value annotation legend: ns: $5.00\times10^{-2} < p <= 1.00$, *: $1.00\times10^{-2} < p <= 5.00\times10^{-2}$, **: $1.00\times10^{-3} < p <= 1.00\times10^{-2}$, ***: $1.00\times10^{-4} < p <= 1.00\times10^{-3}$, ****: $p <= 1.00\times10^{-4}$.

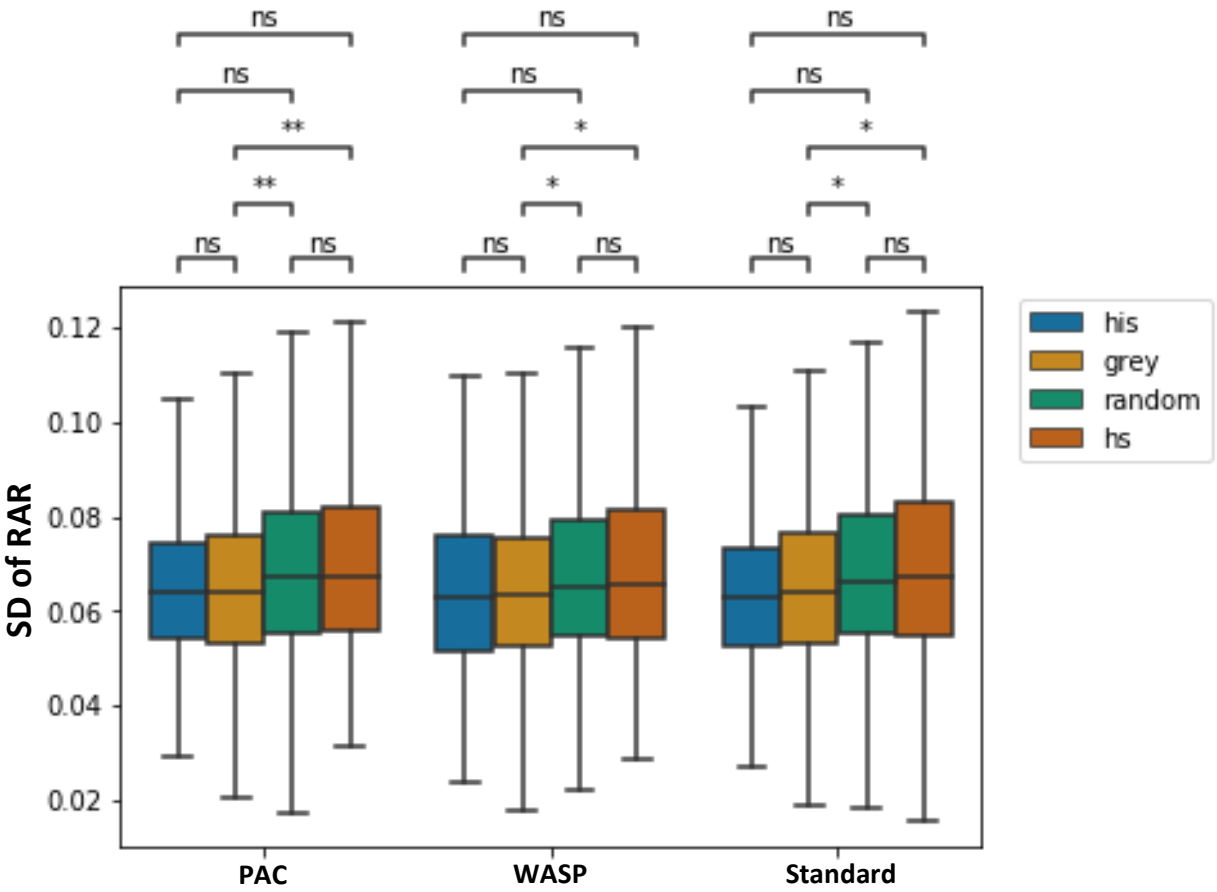


Figure 46. **The SD of reference allele ratio (RAR) across individuals per gene to detect HIS genes in PAC, WASP-filtered alignment and standard alignment.**
All methods are significantly smaller in standard deviation (SD) reference allele ratio (RAR) when comparing predicted haploinsufficient (grey) and random and grey and haplosufficient (HS). PAC exhibited a stronger effect size in both cases. Genes on autosomes with >=20× coverage and present in all methods are shown. P-value annotation legend: ns: $5.00\times10^{-2} < p <= 1.00$, *: $1.00\times10^{-2} < p <= 5.00\times10^{-2}$, **: $1.00\times10^{-3} < p <= 1.00\times10^{-2}$, ***: 1.00×10-4 < p <= 1.00×10^{-3}, ****: $p <= 1.00\times10^{-4}$.

5.3.2 G×E ANALYSIS

ASE has many utilities, and to further demonstrate the benefit of improved allelic quantification with PAC in finding biologically informative events, I explored PAC in the context of G×E interactions [109]. For this, I obtained gene expression data from Findley *et al*., 2021 [102] in an attempt to identify cell-type specific ASE. In this study, RNA-seq was performed on three different cell types (Lymphoblastoid Cell Lines (LCL), Induced

Pluripotent Stem Cells, (iPSC) and iPSC-derived cardiomyocytes) from six individuals after exposure to different treatments. It has been shown with these data that conditional ASE (cASE), induced with treatment or in a particular cell type, were enriched for genes that are linked to disease-relevant phenotypes. For example, metal treatments such as cadmium in cardiomyocytes generated the largest overlap of 7 genes between cASE and putative disease genes for coronary artery disease [102], which is consistent with previous research showing that cadmium can promote atherosclerosis [329]. Cardiomyocytes are a highly relevant cell type for this disease.

I set to explore if PAC can provide additional information relative to the standard alignment to interpret biological function. For this, I obtained RNA-seq data from LCL, cardiomyocytes and iPSC treated with cadmium from six individuals. I aligned the RNA-seq data with PAC and standard alignment as a baseline to compare the results, it being the most common alignment method. I then identified cardiomyocyte cell-type specific heterozygous sites under ASE. With data obtained with PAC, the average number of heterozygous sites in an individual is 102, which shows a significant ASE in cardiomyocytes (binomial test, $P<0.05/18,537$ tests, which is the mean number of sites tested per sample across all methods and individuals), but not in LCLs or iPSCs (binomial test, $P>0.05$ uncorrected). 13 of these 102 heterozygous sites are within genes previously linked with coronary artery disease [328]. Data from standard alignment produced the same average number of cardiomyocyte specific ASE events per individual as PAC. Conversely, many of the heterozygous sites identified in standard alignment are different. In standard alignment, 8 sites overlap putative coronary artery disease genes. When considering only heterozygous sites that show ASE in cardiomyocytes in a cell-type specific manner in PAC data only, I revealed four genes (*GPX1, RETREG3, TCTA* and *PMVK*) implicated in coronary artery disease. These would be missed using standard alignment.

5.4 DISCUSSION

In this chapter, I explored the utility of ASE to understand biological mechanisms underlying cellular processes, and showed the value of performing these analyses with more accurate read count ratios at heterozygous sites, such as those generated with PAC. For this, I used two examples: haploinsufficiency and G×E interactions.

5.4.1 HAPLOINSUFFICIENCY

I used the GTEx data set to show that ASE can distinguish HIS genes from HS genes. This demonstrates that ASE can be used as a metric to predict HIS genes and may be informative when added to other genomic features within classification models. To date ASE has not been used in HIS prediction models, so expanding this work to include this measure may be more powerful. Haploinsufficiency is a feature of many devastating diseases, however, the study of these processes can be difficult due to a lack of power in population-level data to detect potential HIS genes. This may be further restricted in humans as HIS genes can be unviable as they are essential and expressed at early developmental stages [330-332]. Haploinsufficiency has been studied in animal models by gene deletions and cross-breeding, which is not possible in humans. There are studies that explore HIS genes in human cellular models [332], however, these do not always replicate results *in vivo*. Therefore, machine learning and other prediction methods offer an attractive approach to detect and study HIS genes. I propose ASE as a feature to be included in HIS models in the future.

Selecting the tissue of highest expression did not increase the signal between HIS and HS genes. However, HIS genes have been reported to be expressed during development and to be more tissue-specific [308]. In this analysis I did not select tissue-specific gene expression but rather expression data from genes in tissues where it was expressed at the highest rate, so these results are not directly comparable.

I next demonstrated that better read alignment with PAC not only allows the inclusion of many more genes for analysis but also shows higher statistical significance when detecting

HIS genes. In this analysis, PAC generated 22 more HIS genes with sufficient coverage compared to WASP-filtered alignment and 11 more genes than standard alignment to the reference genome. PAC also resulted in 36 more HS genes with sufficient coverage compared to WASP-filtered alignment and 10 more genes in standard alignment. PAC greatly increased the number of grey genes by 413 compared to WASP-filtered alignment and by 174 compared to standard alignment. Because there are a lot fewer HIS genes in the human genome, it is crucial not only to accurately quantify allelic ratios but also to not remove genes due to computational biases and data filtering.

5.4.2 G×E INTERACTIONS

Finally, I demonstrated that ASE can be a valuable tool in the study of G×E interactions. Improved quantification of allelic ratios was crucial in detecting genes under conditional ASE that were missed by standard alignment approaches. Some of these genes have been implicated in tissue relevant disease genes and are therefore likely to be informative of genuine events that would be interesting to study further. Although the number of heterozygous sites was similar between these methods, the standard alignment approach missed biologically relevant information. Given ASE analysis is able to capture a larger number of genes with G×E effects (that are missed by large eQTL studies such as GTEx and GEUVADIS [102]), ASE will become an increasingly important method in this field.

5.4.3 RNA-SEQ FOR BIOLOGICAL RESEARCH

The vast amount of RNA-seq data available offers multiple opportunities to answer important biological questions, if the data is handled and processed correctly and without bias. Another big limitation for utilising transcriptomics in a disease context arises from the availability of the disease-relevant tissue under certain environmental conditions. The environment has been shown to influence gene expression [333]. However, cell type contributes to gene expression changes more than environmental stimuli [102], which is

unsurprising given different cell types have larger differences in their function than in response to external stimulus.

Some disease-relevant tissues might be difficult to access non-invasively, such as the brain or heart. Some genetic disorders are caused by mutations in enhancer regions that are specific to a particular developmental stage or cell type [334]. However, studies are emerging that demonstrate the utility of more accessible tissues to study diseases where the disease-relevant tissue is difficult to obtain. One study showed that LCLs from blood can be used to survey multiple neurodevelopmental rare diseases as these share isoform diversities and are able to test for a large number of neurodevelopmental genes [335]. In addition, it has also been demonstrated that combining iPSCs with blood RNA-seq from the same individual supports the discovery of outliers and thus candidate genes [336]. Similarly, combining transcriptome data from blood and fibroblasts also improved the diagnostic rate [337]. Therefore, ASE is only emerging with its possibilities and has many uses for disease research.

CHAPTER 6 – CONCLUSION

6. CONCLUSION

6.1. THESIS SUMMARY

Each tissue and cell type within an individual will have largely the same genome yet produce such different functions with distinct transcriptomes [103, 338]. The precise control of gene regulation is of paramount importance for cells to be able to adapt to different environments [94, 339, 340]. It has been shown that transcriptomic changes are manifested in many disease states [337, 341, 342]. Currently, most studies investigate transcriptomes with RNA-seq [343]. The most common way to study how genetic variants influence gene expression is eQTL analysis [69, 307]. Although eQTL analysis has been successful in identifying common variants associated with gene expression levels [69, 86, 91, 101, 279, 344, 345], the research would benefit from focusing also on low-frequency and rare variants [346] using ASE analysis. ASE allows the study of allelic imbalance within a single individual, which can be highly advantageous for rare variants or in studies where the starting tissue might be scarce. Allelic imbalance is most often caused by regulatory variants. Additionally, since gene expression quantification can be influenced by experimental effects such as sequencing batch [347], ASE avoids many of these confounding issues, as the mRNA comparison is performed within an individual where the alleles share the same environment [122, 258].

However, ASE has its own limitations, computational biases being the largest caveats. In Chapter 2, I developed PAC, a pipeline to reduce artefacts associated with ASE analysis. I tested the performance of PAC against highly accurate simulated genomic data. I tested PAC with parameters including RNA-seq read trimming and soft-clipping, rescuing multi-mapping reads and using improved phasing with phASER with read-aware mode. I refined PAC to a final version that introduced the lowest level of bias. Analysis using simulated data also revealed a large number of false positive and false negative variants from variant calling, which is known to confound the ASE analysis [348]. Further, I showed that PAC increases the number of retained reads and reduces biases, such as allelic ratios being closer to 1:1 in allelic counts. I performed a preliminary analysis on real data from wild-type samples,

showing that PAC increased the number of reads available for the analysis, reduced mapping bias, and was able to detect tissue-specific ASE events.

In Chapter 3, I tackled the difficulty of running and reproducing results from genomics pipelines generated by different research groups [349]. I wrote PAC into Nextflow to streamline it and released PAC on my GitHub page to make it publicly available for the research community. Further, PAC utilises Docker to automate the download of multiple dependencies that are often difficult to install without computational competency. After, I compared how PAC performed against other commonly used methods, namely aligning RNA-seq reads to the reference genome and performing WASP-filtering prior to ASE calling. I showed that PAC retains more reads, which other methods discard, and assigns them correctly. PAC also reduces biases in allelic ratios when compared to the ground truth data, and I showed that this is maintained when the RNA-seq coverage was reduced. I showed that this improvement comes partially from regions near indels and other variants, where allelic ratios from PAC have significantly smaller differences from ground truth than other methods.

In Chapter 4, I applied PAC to population-level data in order to validate PAC as a method and showed it improves allelic quantification relative to other commonly used methods. I showed that PAC better recapitulates eQTL signals from GTEx whole blood samples than standard alignment or WASP-filtered data. I also showed that allelic ratios from 6 different GTEx tissues obtained either with PAC or the other methods do not correlate with allelic ratios derived from nanopore data. This is most likely due to the read depth, filtering steps that remove reads, and high error rates associated with long-read sequencing. I then demonstrated that PAC best replicated the enrichment of genes that were under ASE in the ground truth simulated data when compared to standard alignment or WASP-filtered data. This demonstrates that accurate alignment can significantly impact the interpretation of the downstream analysis. In this analysis, I used simulated data in order to have a ground truth baseline for methods comparison. However, it uncovers the errors that would be made in real data. I then explored if genes under ASE would be enriched for regulatory regions relative to non-ASE genes. I showed that genes under ASE are indeed significantly enriched for enhancer sequences in healthy individuals

In Chapter 5, I studied the utility of ASE in a disease context. First, I explored haploinsufficiency, where deviation from normal gene dosage causes a disease state [310]. Because these diseases tend to be rare, they are often difficult to study. Currently, there are many efforts to develop prediction methods to discover HIS genes [308, 315], however, ASE has not been used in this context. I showed that allelic ratios significantly differ between known HIS and HS genes in GTEx tissues. I observed the same result when considering only tissues where each gene was most highly expressed, indicating that this filter is not required to uncover potential HIS genes. When I compared PAC to standard alignment and WASP-filtered data, PAC showed higher statistical significance between allele ratios of HIS and HS genes in whole blood GTEx samples. Thus, I demonstrated that improved allelic quantification can be utilised as a parameter to predict HIS genes, which is important for better understanding human disease. Finally, I showed that improved detection of genes under ASE with PAC helped to identify disease relevant genes that were under G×E interactions. These genes were missed with a standard alignment approach. This again demonstrates the downstream consequences of computational biases in interpreting biologically relevant findings. Together the results from this chapter show the value of ASE analysis in identifying disease relevant genes.

6.2. THESIS IMPROVEMENTS AND FURTHER DIRECTIONS FOR RESEARCH

As with any research, the work carried out for this thesis has limitations, which I will review in this section. The main limitation of PAC is the computational time and power required to run the pipeline. With five test GTEx samples of average depth of ~44 million paired-end reads, PAC takes an average of 12 h and 6 min to generate the site and gene-level ASE data per sample, whereas generating these data from standard alignment takes an average of 3 h and 28 min with the same computational set up. This is a considerably longer time, which is mostly explained by the steps generating diploid genomes and subsequently aligning the RNA-seq reads multiple times. Within this time, PAC does however provide with standard alignment data as well. However, the PAC tool was primarily generated for the purpose of

rare variant and rare disease analysis where the sample sizes are usually smaller than for population-level studies such as eQTL or GWA studies. With more time I would have liked to develop PAC v2 that would have had more options for the user. For example, many large sequencing and genomics projects release RNA-seq reads aligned to the reference genome. With the current version of PAC, these need to be converted to raw reads. A useful option would be to skip the initial steps (process prepare_star_genome_index and process rnaseq_mapping_star) of aligning reads to the reference genome if the user has a BAM file available. With the current methods, the BAM file needs to be converted to unaligned RNA-seq reads first.

Another limitation of PAC is that the accuracy of diploid genomes it generates depends on the quality of phased variants that are supplied for PAC. The human genome contains tens of thousands of rare variants [40] which pose difficulty in variant calling and phasing, with commonly used methods that are based on population-level information. It has been demonstrated that genes under ASE are more likely to have a rare variant near them [350], which might be important for the regulation of the genes. Variant calling is also known to be especially inaccurate in regions with structural variation and this biases ASE. For example, a duplicated region will cause one allele to be expressed twice as high, while the other allele is unchanged [151]. Indeed, I have shown that the most commonly used variant calling approach with GATK introduces a large number of false positive and false negative variants. Better phasing methods will improve genomics analysis pipelines outside of the ASE field as well, however, it is beyond the scope of this thesis.

There are other tools available that generate and align to diploid personalised genomes [218, 219, 271]. However, these do not provide easy-to-use pipelines for users, nor do they adopt any of the other filtering steps that PAC performs. In addition to diploid genomes, there are also other methods to include individual variation into the reference genome, including incorporating multiple population reference genomes instead of one linear reference to reduce reference bias [184] or graph genomes [351-353]. Similarly, these are also computationally expensive as the number of variants increases. However, it would be interesting to explore how the performance of PAC would be affected by incorporating these types of references instead of a diploid genome.

With more time and resources, it would have been interesting to explore nanopore data in more detail and in particular why the correlation between allelic ratios from short and long-read data is so poor. Ideally, I would have liked to use the same samples to obtain my own nanopore data and short-read RNA-seq data at the same time, as transcriptomics data is sensitive to environmental perturbations. Long-read sequencing can sequence whole transcripts and therefore offers an incredible resource to overcome many limitations associated with short-read sequencing, in particular for ASE detection. Longer reads overcome issues with splice variants and phasing that can bias ASE analysis. Splice isoforms can be difficult to deal with and ignoring them can lead to biases when inferring the ASE at the gene level [354].

Another limitation of transcriptomics in the disease context is that it needs relevant tissue [37] as gene expression varies across tissues and environmental conditions [86, 355, 356], and the genomic regions that contribute to disease have been shown to be concentrated in transcribed regions [2]. It is often not possible to obtain disease-relevant tissues from a living donor. Reprogramming somatic cells to pluripotent cells offers an alternative to obtaining relevant tissues. Indeed, in Chapter 2, I used iPSC-derived neuronal cells to test PAC, which offered a way to explore the changes in ASE during development. Another way to measure gene expression of the target tissue is to use biofluids such as blood. It has been shown that extracellular RNA is released from cells which is bound to extracellular vesicles or RNA-binding proteins or lipoproteins to protect from degradation, and therefore can be informative for certain diseases [357]. As ASE has been used to detect disease genes [132, 168, 358, 359], it could offer avenues to study disease without invasive tissue biopsies, some of which might not be possible.

Taken together, in this thesis, I have developed a novel computational pipeline that improves ASE detection and quantification and shows that this improved allelic quantification has biological implications. With more time I would have explored ASE in rare disease samples where I believe PAC would have had the biggest impact. PAC maximises the number of reads while maintaining accuracy; therefore, disease cohorts with small sample sizes would benefit from such a tool. Another avenue I would have liked to explore is ASE in diseases where the disease mechanism involves haploinsufficiency such as Kabuki syndrome. It would be interesting to see how allelic ratios in HIS genes implicated in such

diseases would differ from those in healthy individuals. Lastly, I believe improved ASE detection with PAC would be a valuable addition to HIS gene prediction tools.

REFERENCES

1. Zeng, J., et al., *Widespread signatures of natural selection across human complex traits and functional genomic categories.* Nat Commun, 2021. **12**(1): p. 1164.
2. Boyle, E.A., Y.I. Li, and J.K. Pritchard, *An Expanded View of Complex Traits: From Polygenic to Omnigenic.* Cell, 2017. **169**(7): p. 1177-1186.
3. Pritchard, J.K., J.K. Pickrell, and G. Coop, *The genetics of human adaptation: hard sweeps, soft sweeps, and polygenic adaptation.* Curr Biol, 2010. **20**(4): p. R208-15.
4. McCarthy, M.I., et al., *Genome-wide association studies for complex traits: consensus, uncertainty and challenges.* Nature Reviews Genetics, 2008. **9**(5): p. 356-369.
5. Glassberg, E.C., et al., *Evidence for Weak Selective Constraint on Human Gene Expression.* Genetics, 2019. **211**(2): p. 757-772.
6. Fu, Y.X., *Statistical tests of neutrality of mutations against population growth, hitchhiking and background selection.* Genetics, 1997. **147**(2): p. 915-25.
7. Ramachandran, S., et al., *Support from the relationship of genetic and geographic distance in human populations for a serial founder effect originating in Africa.* Proc Natl Acad Sci U S A, 2005. **102**(44): p. 15942-7.
8. Wainschtein, P., et al., *Assessing the contribution of rare variants to complex trait heritability from whole-genome sequence data.* Nature Genetics, 2022. **54**(3): p. 263-273.
9. Gazal, S., et al., *Functional architecture of low-frequency variants highlights strength of negative selection across coding and non-coding annotations.* Nat Genet, 2018. **50**(11): p. 1600-1607.
10. Harper, A.R., et al., *Common genetic variants and modifiable risk factors underpin hypertrophic cardiomyopathy susceptibility and expressivity.* Nat Genet, 2021. **53**(2): p. 135-142.
11. Shringarpure, S.S., et al., *Large-scale trans-ethnic replication and discovery of genetic associations for rare diseases with self-reported medical data.* medRxiv, 2021: p. 2021.06.09.21258643.
12. Lappalainen, T. and G. MacArthur Daniel, *From variant to function in human disease genetics.* Science, 2021. **373**(6562): p. 1464-1468.
13. Lander, E.S., et al., *Initial sequencing and analysis of the human genome.* Nature, 2001. **409**(6822): p. 860-921.
14. Venter, J.C., et al., *The sequence of the human genome.* Science, 2001. **291**(5507): p. 1304-51.
15. McVean, G.A., et al., *An integrated map of genetic variation from 1,092 human genomes.* Nature, 2012. **491**(7422): p. 56-65.
16. *The International HapMap Project.* Nature, 2003. **426**(6968): p. 789-96.

17. Auton, A., et al., *A global reference for human genetic variation.* Nature, 2015. **526**(7571): p. 68-74.

18. McCarthy, S., et al., *A reference panel of 64,976 haplotypes for genotype imputation.* Nat Genet, 2016. **48**(10): p. 1279-83.

19. Lek, M., et al., *Analysis of protein-coding genetic variation in 60,706 humans.* Nature, 2016. **536**(7616): p. 285-91.

20. Karczewski, K.J., et al., *The mutational constraint spectrum quantified from variation in 141,456 humans.* Nature, 2020. **581**(7809): p. 434-443.

21. Niroula, A. and M. Vihinen, *How good are pathogenicity predictors in detecting benign variants?* PLoS Comput Biol, 2019. **15**(2): p. e1006481.

22. Eriksson, N., et al., *A genetic variant near olfactory receptor genes influences cilantro preference.* Flavour, 2012. **1**(1): p. 22.

23. Livesey, B.J. and J.A. Marsh, *Using deep mutational scanning to benchmark variant effect predictors and identify disease mutations.* Mol Syst Biol, 2020. **16**(7): p. e9380.

24. Claussnitzer, M., et al., *A brief history of human disease genetics.* Nature, 2020. **577**(7789): p. 179-189.

25. Clark, M.J., et al., *Performance comparison of exome DNA sequencing technologies.* Nat Biotechnol, 2011. **29**(10): p. 908-14.

26. Vitsios, D., et al., *Prioritizing non-coding regions based on human genomic constraint and sequence context with deep learning.* Nature Communications, 2021. **12**(1): p. 1504.

27. Van der Auwera, G.A., et al., *From FastQ data to high confidence variant calls: the Genome Analysis Toolkit best practices pipeline.* Curr Protoc Bioinformatics, 2013. **43**(1110): p. 11.10.1-11.10.33.

28. Regier, A.A., et al., *Functional equivalence of genome sequencing analysis pipelines enables harmonized variant calling across human genetics projects.* Nat Commun, 2018. **9**(1): p. 4038.

29. Lappalainen, T., et al., *Genomic Analysis in the Age of Human Genome Sequencing.* Cell, 2019. **177**(1): p. 70-84.

30. Koboldt, D.C., *Best practices for variant calling in clinical sequencing.* Genome Medicine, 2020. **12**(1): p. 91.

31. Zverinova, S. and V. Guryev, *Variant calling: Considerations, practices, and developments.* Human Mutation, 2022. **43**(8): p. 976-985.

32. Dharshini, S.A.P., Y.H. Taguchi, and M.M. Gromiha, *Identifying suitable tools for variant detection and differential gene expression using RNA-seq data.* Genomics, 2020. **112**(3): p. 2166-2172.

33. Howie, B.N., P. Donnelly, and J. Marchini, *A Flexible and Accurate Genotype Imputation Method for the Next Generation of Genome-Wide Association Studies.* PLOS Genetics, 2009. **5**(6): p. e1000529.

34. Abel, H.J., et al., *Mapping and characterization of structural variation in 17,795 human genomes.* Nature, 2020. **583**(7814): p. 83-89.

35. Molinari, F., et al., *Impaired mitochondrial glutamate transport in autosomal recessive neonatal myoclonic epilepsy.* Am J Hum Genet, 2005. **76**(2): p. 334-9.

36. Rodenburg, R.J., *The functional genomics laboratory: functional validation of genetic variants.* J Inherit Metab Dis, 2018. **41**(3): p. 297-307.

37. Cummings Beryl, B., et al., *Improving genetic diagnosis in Mendelian disease with transcriptome sequencing.* Science Translational Medicine, 2017. **9**(386): p. eaal5209.

38. Kremer, L.S., et al., *Genetic diagnosis of Mendelian disorders via RNA sequencing.* Nature Communications, 2017. **8**(1): p. 15824.

39. Fowler, D.M. and S. Fields, *Deep mutational scanning: a new style of protein science.* Nat Methods, 2014. **11**(8): p. 801-7.

40. Keinan, A. and A.G. Clark, *Recent explosive human population growth has resulted in an excess of rare genetic variants.* Science, 2012. **336**(6082): p. 740-3.

41. Maher, M.C., et al., *Population genetics of rare variants and complex diseases.* Hum Hered, 2012. **74**(3-4): p. 118-28.

42. Manolio, T.A., *Bringing genome-wide association findings into clinical use.* Nat Rev Genet, 2013. **14**(8): p. 549-58.

43. Simons, Y.B., et al., *The deleterious mutation load is insensitive to recent population history.* Nat Genet, 2014. **46**(3): p. 220-4.

44. Abecasis, G.R., et al., *A map of human genome variation from population-scale sequencing.* Nature, 2010. **467**(7319): p. 1061-73.

45. Cortes, A. and M.A. Brown, *Promise and pitfalls of the Immunochip.* Arthritis Res Ther, 2011. **13**(1): p. 101.

46. Walter, K., et al., *The UK10K project identifies rare variants in health and disease.* Nature, 2015. **526**(7571): p. 82-90.

47. Vissers, L.E., et al., *Array-based comparative genomic hybridization for the genomewide detection of submicroscopic chromosomal abnormalities.* Am J Hum Genet, 2003. **73**(6): p. 1261-70.

48. Ng, S.B., et al., *Targeted capture and massively parallel sequencing of 12 human exomes.* Nature, 2009. **461**(7261): p. 272-6.

49. Wheeler, D.A., et al., *The complete genome of an individual by massively parallel DNA sequencing.* Nature, 2008. **452**(7189): p. 872-6.

50. Zhang, F. and J.R. Lupski, *Non-coding genetic variants in human disease.* Hum Mol Genet, 2015. **24**(R1): p. R102-10.

51. Maurano, M.T., et al., *Systematic localization of common disease-associated variation in regulatory DNA.* Science, 2012. **337**(6099): p. 1190-5.

52. Buniello, A., et al., *The NHGRI-EBI GWAS Catalog of published genome-wide association studies, targeted arrays and summary statistics 2019.* Nucleic Acids Res, 2019. **47**(D1): p. D1005-d1012.

53. Manolio, T.A., et al., *Finding the missing heritability of complex diseases.* Nature, 2009. **461**(7265): p. 747-53.

54. Mahajan, A., et al., *Fine-mapping type 2 diabetes loci to single-variant resolution using high-density imputation and islet-specific epigenome maps.* Nat Genet, 2018. **50**(11): p. 1505-1513.

55. Lango Allen, H., et al., *Hundreds of variants clustered in genomic loci and biological pathways affect human height.* Nature, 2010. **467**(7317): p. 832-8.

56. Bulik-Sullivan, B., et al., *An atlas of genetic correlations across human diseases and traits.* Nat Genet, 2015. **47**(11): p. 1236-41.

57. Loh, P.R., et al., *Contrasting genetic architectures of schizophrenia and other complex diseases using fast variance-components analysis.* Nat Genet, 2015. **47**(12): p. 1385-92.

58. Shi, H., G. Kichaev, and B. Pasaniuc, *Contrasting the Genetic Architecture of 30 Complex Traits from Summary Association Data.* Am J Hum Genet, 2016. **99**(1): p. 139-53.

59. Tam, V., et al., *Benefits and limitations of genome-wide association studies.* Nature Reviews Genetics, 2019. **20**(8): p. 467-484.

60. Martin, A.R., et al., *Clinical use of current polygenic risk scores may exacerbate health disparities.* Nat Genet, 2019. **51**(4): p. 584-591.

61. Cook, J.P. and A.P. Morris, *Multi-ethnic genome-wide association study identifies novel locus for type 2 diabetes susceptibility.* Eur J Hum Genet, 2016. **24**(8): p. 1175-80.

62. Hoffmann, T.J., et al., *A large multiethnic genome-wide association study of prostate cancer identifies novel risk variants and substantial ethnic differences.* Cancer Discov, 2015. **5**(8): p. 878-91.

63. Shigemizu, D., et al., *Ethnic and trans-ethnic genome-wide association studies identify new loci influencing Japanese Alzheimer's disease risk.* Translational Psychiatry, 2021. **11**(1): p. 151.

64. Lappalainen, T., et al., *Transcriptome and genome sequencing uncovers functional variation in humans.* Nature, 2013. **501**(7468): p. 506-511.

65. Dunham, I., et al., *An integrated encyclopedia of DNA elements in the human genome.* Nature, 2012. **489**(7414): p. 57-74.

66. Battle, A., et al., *Genetic effects on gene expression across human tissues.* Nature, 2017. **550**(7675): p. 204-213.

67. Gamazon, E.R., et al., *Using an atlas of gene regulation across 44 human tissues to inform complex disease- and trait-associated variation.* Nat Genet, 2018. **50**(7): p. 956-967.

68. Ongen, H., et al., *Estimating the causal tissues for complex traits and diseases.* Nat Genet, 2017. **49**(12): p. 1676-1683.

69. Emilsson, V., et al., *Genetics of gene expression and its effect on disease.* Nature, 2008. **452**(7186): p. 423-428.

70. Stranger, B.E., et al., *Population genomics of human gene expression.* Nature Genetics, 2007. **39**(10): p. 1217-1224.

71. Nica, A.C., et al., *Candidate Causal Regulatory Effects by Integration of Expression QTLs with Complex Trait Genetic Associations.* PLOS Genetics, 2010. **6**(4): p. e1000895.

72. Nicolae, D.L., et al., *Trait-associated SNPs are more likely to be eQTLs: annotation to enhance discovery from GWAS.* PLoS Genet, 2010. **6**(4): p. e1000888.

73. Frésard, L., et al., *Identification of rare-disease genes using blood transcriptome sequencing and large control cohorts.* Nature Medicine, 2019. **25**(6): p. 911-919.

74. Karczewski, K.J., et al., *Systematic functional regulatory assessment of disease-associated variants.* Proc Natl Acad Sci U S A, 2013. **110**(23): p. 9607-12.

75. Kernohan, K.D., et al., *Whole-transcriptome sequencing in blood provides a diagnosis of spinal muscular atrophy with progressive myoclonic epilepsy.* Hum Mutat, 2017. **38**(6): p. 611-614.

76. Kukurba, K.R. and S.B. Montgomery, *RNA Sequencing and Analysis.* Cold Spring Harb Protoc, 2015. **2015**(11): p. 951-69.

77. Stark, R., M. Grzelak, and J. Hadfield, *RNA sequencing: the teenage years.* Nature Reviews Genetics, 2019. **20**(11): p. 631-656.

78. Conesa, A., et al., *A survey of best practices for RNA-seq data analysis.* Genome Biology, 2016. **17**(1): p. 13.

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

80. Langmead, B. and S.L. Salzberg, *Fast gapped-read alignment with Bowtie 2.* Nature Methods, 2012. **9**(4): p. 357-359.

81. Musich, R., L. Cadle-Davidson, and M.V. Osier, *Comparison of Short-Read Sequence Aligners Indicates Strengths and Weaknesses for Biologists to Consider.* Frontiers in Plant Science, 2021. **12**.

82. Dobin, A., et al., *STAR: ultrafast universal RNA-seq aligner.* Bioinformatics (Oxford, England), 2013. **29**(1): p. 15-21.

83. Bray, N.L., et al., *Near-optimal probabilistic RNA-seq quantification.* Nature Biotechnology, 2016. **34**(5): p. 525-527.

84. Zhang, H., C. Jain, and S. Aluru, *A comprehensive evaluation of long read error correction methods.* BMC Genomics, 2020. **21**(6): p. 889.

85. Pickrell, J.K., et al., *Understanding mechanisms underlying human gene expression variation with RNA sequencing.* Nature, 2010. **464**(7289): p. 768-772.

86. *Human genomics. The Genotype-Tissue Expression (GTEx) pilot analysis: multitissue gene regulation in humans.* Science, 2015. **348**(6235): p. 648-60.

87. Rantalainen, M., C.M. Lindgren, and C.C. Holmes, *Robust Linear Models for Cis-eQTL Analysis.* PLOS ONE, 2015. **10**(5): p. e0127882.

88. Westra, H.J. and L. Franke, *From genome to function by studying eQTLs.* Biochim Biophys Acta, 2014. **1842**(10): p. 1896-1902.

89. Shan, N., Z. Wang, and L. Hou, *Identification of trans-eQTLs using mediation analysis with multiple mediators.* BMC Bioinformatics, 2019. **20**(3): p. 126.

90. Võsa, U., et al., *Large-scale cis- and trans-eQTL analyses identify thousands of genetic loci and polygenic scores that regulate blood gene expression.* Nature Genetics, 2021. **53**(9): p. 1300-1310.

91. Dimas, A.S., et al., *Common regulatory variation impacts gene expression in a cell type-dependent manner.* Science, 2009. **325**(5945): p. 1246-50.

92. Brown, C.D., L.M. Mangravite, and B.E. Engelhardt, *Integrative Modeling of eQTLs and Cis-Regulatory Elements Suggests Mechanisms Underlying Cell Type Specificity of eQTLs.* PLOS Genetics, 2013. **9**(8): p. e1003649.

93. Moyerbrailean, G.A., et al., *High-throughput allele-specific expression across 250 environmental conditions.* Genome Res, 2016. **26**(12): p. 1627-1638.

94. Buil, A., et al., *Gene-gene and gene-environment interactions detected by transcriptome sequence analysis in twins.* Nat Genet, 2015. **47**(1): p. 88-91.

95. Kerimov, N., et al., *A compendium of uniformly processed human gene expression and splicing quantitative trait loci.* Nature Genetics, 2021. **53**(9): p. 1290-1299.

96. Wang, Q.S., et al., *Leveraging supervised learning for functionally informed fine-mapping of cis-eQTLs identifies an additional 20,913 putative causal eQTLs.* Nat Commun, 2021. **12**(1): p. 3394.

97. Schaid, D.J., W. Chen, and N.B. Larson, *From genome-wide associations to candidate causal variants by statistical fine-mapping.* Nat Rev Genet, 2018. **19**(8): p. 491-504.

98. Chen, W., et al., *Fine Mapping Causal Variants with an Approximate Bayesian Method Using Marginal Test Statistics.* Genetics, 2015. **200**(3): p. 719-36.

99. Maller, J.B., et al., *Bayesian refinement of association signals for 14 loci in 3 common diseases.* Nat Genet, 2012. **44**(12): p. 1294-301.

100. Croteau-Chonka, D.C., et al., *Expression Quantitative Trait Loci Information Improves Predictive Modeling of Disease Relevance of Non-Coding Genetic Variation.* PLoS One, 2015. **10**(10): p. e0140758.

101. Grundberg, E., et al., *Mapping cis- and trans-regulatory effects across multiple tissues in twins.* Nature Genetics, 2012. **44**(10): p. 1084-1089.

102. Findley, A.S., et al., *Functional dynamic genetic effects on gene regulation are specific to particular cell types and environmental conditions.* Elife, 2021. **10**.

103. Lonsdale, J., et al., *The Genotype-Tissue Expression (GTEx) project.* Nature Genetics, 2013. **45**(6): p. 580-585.

104. null, n., et al., *The GTEx Consortium atlas of genetic regulatory effects across human tissues.* Science, 2020. **369**(6509): p. 1318-1330.

105. Baran, Y., et al., *The landscape of genomic imprinting across diverse adult human tissues.* Genome Research, 2015.

106. Rivas, M.A., et al., *Effect of predicted protein-truncating genetic variants on the human transcriptome.* Science, 2015. **348**(6235): p. 666-669.

107. Chiang, C., et al., *The impact of structural variation on human gene expression.* Nature Genetics, 2017. **49**(5): p. 692-699.

108. Castel, S.E., et al., *A vast resource of allelic expression data spanning human tissues.* Genome Biology, 2020. **21**(1): p. 234.

109. Saukkonen, A., H. Kilpinen, and A. Hodgkinson, *Highly accurate quantification of allelic gene expression for population and disease genetics.* Genome Research, 2022. **32**(8): p. 1565-1572.

110. Kasowski, M., et al., *Variation in transcription factor binding among humans.* Science (New York, N.Y.), 2010. **328**(5975): p. 232-235.

111. Reddy, T.E., et al., *Effects of sequence variation on differential allelic transcription factor occupancy and gene expression.* Genome Res, 2012. **22**(5): p. 860-9.

112. Zhang, D., et al., *Genetic control of individual differences in gene-specific methylation in human brain.* American journal of human genetics, 2010. **86**(3): p. 411-419.

113. Montgomery, S.B., et al., *Rare and common regulatory variation in population-scale sequenced human genomes.* PLoS Genet, 2011. **7**(7): p. e1002144.

114. Demirdjian, L., et al., *Detecting Allele-Specific Alternative Splicing from Population-Scale RNA-Seq Data.* Am J Hum Genet, 2020. **107**(3): p. 461-472.

115. Amoah, K., et al., *Allele-specific alternative splicing and its functional genetic variants in human tissues.* Genome Research, 2021.

116. Garieri, M., et al., *Extensive cellular heterogeneity of X inactivation revealed by single-cell allele-specific expression in human fibroblasts.* Proceedings of the National Academy of Sciences, 2018. **115**(51): p. 13015-13020.

117. Shvetsova, E., et al., *Skewed X-inactivation is common in the general female population.* European Journal of Human Genetics, 2019. **27**(3): p. 455-465.

118. Chess, A., et al., *Allelic inactivation regulates olfactory receptor gene expression.* Cell, 1994. **78**(5): p. 823-34.

119. Pernis, B., et al., *Cellular localization of immunoglobulins with different allotypic specificities in rabbit lymphoid tissues.* J Exp Med, 1965. **122**(5): p. 853-76.

120. Hozumi, N. and S. Tonegawa, *Evidence for somatic rearrangement of immunoglobulin genes coding for variable and constant regions.* Proc Natl Acad Sci U S A, 1976. **73**(10): p. 3628-32.

121. Cleary, S. and C. Seoighe, *Perspectives on Allele-Specific Expression.* Annual Review of Biomedical Data Science, 2021. **4**(1): p. 101-122.

122. Castel, S.E., et al., *Tools and best practices for data processing in allelic expression analysis.* Genome Biology, 2015. **16**(1): p. 195.

123. van de Geijn, B., et al., *WASP: allele-specific software for robust molecular quantitative trait locus discovery.* Nature Methods, 2015. **12**(11): p. 1061-1063.

124. Hodgkinson, A., et al., *A haplotype-based normalization technique for the analysis and detection of allele specific expression.* BMC Bioinformatics, 2016. **17**(1): p. 364.

125. Muzzey, D., E.A. Evans, and C. Lieber, *Understanding the Basics of NGS: From Mechanism to Variant Calling.* Curr Genet Med Rep, 2015. **3**(4): p. 158-165.

126. Pastinen, T., *Genome-wide allele-specific analysis: insights into regulatory variation.* Nature Reviews Genetics, 2010. **11**(8): p. 533-538.

127. Lappalainen, T., et al., *Epistatic selection between coding and regulatory variation in human evolution and disease.* Am J Hum Genet, 2011. **89**(3): p. 459-63.

128. Tuch, B.B., et al., *Tumor transcriptome sequencing reveals allelic expression imbalances associated with copy number alterations.* PLoS One, 2010. **5**(2): p. e9317.

129. MacArthur, D.G., et al., *A systematic survey of loss-of-function variants in human protein-coding genes.* Science, 2012. **335**(6070): p. 823-8.

130. Balliu, B., et al., *Genetic regulation of gene expression and splicing during a 10-year period of human aging.* Genome Biology, 2019. **20**(1): p. 230.

131. de Santiago, I., et al., *BaalChIP: Bayesian analysis of allele-specific transcription factor binding in cancer genomes.* Genome Biology, 2017. **18**(1): p. 39.

132. Lee, C., et al., *Profiling allele-specific gene expression in brains from individuals with autism spectrum disorder reveals preferential minor allele usage.* Nature Neuroscience, 2019. **22**(9): p. 1521-1532.

133. McKean, D.M., et al., *Loss of RNA expression and allele-specific expression associated with congenital heart disease.* Nature Communications, 2016. **7**(1): p. 12824.

134. Izzi, B., et al., *Allele-specific DNA methylation reinforces PEAR1 enhancer activity.* Blood, 2016. **128**(7): p. 1003-1012.

135. de Klein, N., et al., *Imbalanced expression for predicted high-impact, autosomal-dominant variants in a cohort of 3,818 healthy samples.* bioRxiv, 2020: p. 2020.09.19.300095.

136. Bader, D.M., et al., *Negative feedback buffers effects of regulatory variants.* Molecular Systems Biology, 2015. **11**(1): p. 785.

137. Denby, C.M., et al., *Negative feedback confers mutational robustness in yeast transcription factor regulation.* Proc Natl Acad Sci U S A, 2012. **109**(10): p. 3874-8.

138. Marciano, D.C., et al., *Negative feedback in genetic circuits confers evolutionary resilience and capacitance.* Cell Rep, 2014. **7**(6): p. 1789-95.

139. Choi, K., N. Raghupathy, and G.A. Churchill, *A Bayesian mixture model for the analysis of allelic expression in single cells.* Nature Communications, 2019. **10**(1): p. 5188.

140. Hudson, L.E. and R.L. Allen, *Leukocyte Ig-Like Receptors - A Model for MHC Class I Disease Associations.* Front Immunol, 2016. **7**: p. 281.

141. Degner, J.F., et al., *Effect of read-mapping biases on detecting allele-specific expression from RNA-sequencing data.* Bioinformatics (Oxford, England), 2009. **25**(24): p. 3207-3212.

142. Stevenson, K.R., J.D. Coolon, and P.J. Wittkopp, *Sources of bias in measures of allele-specific expression derived from RNA-seq data aligned to a single reference genome.* BMC Genomics, 2013. **14**(1): p. 536.

143. Snyder, M.W., et al., *Haplotype-resolved genome sequencing: experimental methods and applications.* Nature Reviews Genetics, 2015. **16**(6): p. 344-358.

144. Tewhey, R., et al., *The importance of phase information for human genomics.* Nature Reviews Genetics, 2011. **12**(3): p. 215-223.

145. Altshuler, D.M., et al., *Integrating common and rare genetic variation in diverse human populations.* Nature, 2010. **467**(7311): p. 52-58.

146. Al Bkhetan, Z., et al., *eQTLHap: a tool for comprehensive eQTL analysis considering haplotypic and genotypic effects.* Briefings in Bioinformatics, 2021. **22**(5): p. bbab093.

147. Browning, S.R. and B.L. Browning, *Haplotype phasing: existing methods and new developments.* Nature Reviews Genetics, 2011. **12**(10): p. 703-714.

148. Choi, Y., et al., *Comparison of phasing strategies for whole human genomes.* PLOS Genetics, 2018. **14**(4): p. e1007308.

149. McKenna, A., et al., *The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data.* Genome Res, 2010. **20**(9): p. 1297-303.

150. Mohammadi, P., et al., *Quantifying the regulatory effect size of cis-acting genetic variation using allelic fold change.* Genome Res, 2017. **27**(11): p. 1872-1884.

151. Rozowsky, J., et al., *AlleleSeq: analysis of allele-specific expression and binding in a network framework.* Molecular systems biology, 2011. **7**: p. 522-522.

152. Raghupathy, N., et al., *Hierarchical analysis of RNA-seq reads improves the accuracy of allele-specific expression.* Bioinformatics, 2018. **34**(13): p. 2177-2184.

153. Harvey, C.T., et al., *QuASAR: quantitative allele-specific analysis of reads.* Bioinformatics, 2015. **31**(8): p. 1235-42.

154. Miao, Z., et al., *ASElux: an ultra-fast and accurate allelic reads counter.* Bioinformatics, 2018. **34**(8): p. 1313-1320.

155. Fan, J., et al., *ASEP: Gene-based detection of allele-specific expression across individuals in a population by RNA sequencing.* PLoS Genet, 2020. **16**(5): p. e1008786.

156. Edsgärd, D., et al., *GeneiASE: Detection of condition-dependent and static allele-specific expression from RNA-seq data without haplotype information.* Sci Rep, 2016. **6**: p. 21134.

157. Mayba, O., et al., *MBASED: allele-specific expression detection in cancer tissues and cell lines.* Genome Biology, 2014. **15**(8): p. 405.

158. Turro, E., et al., *Haplotype and isoform specific expression estimation using multi-mapping RNA-seq reads.* Genome Biol, 2011. **12**(2): p. R13.

159. Skelly, D.A., et al., *A powerful and flexible statistical framework for testing hypotheses of allele-specific gene expression from RNA-seq data.* Genome Res, 2011. **21**(10): p. 1728-37.

160. Romanel, A., et al., *ASEQ: fast allele-specific studies from next-generation sequencing data.* BMC Medical Genomics, 2015. **8**(1): p. 9.

161. Zambelli, F., et al., *aScan: A Novel Method for the Study of Allele Specific Expression in Single Individuals.* J Mol Biol, 2021. **433**(11): p. 166829.

162. Deonovic, B., et al., *IDP-ASE: haplotyping and quantifying allele-specific expression at the gene and gene isoform level by hybrid sequencing.* Nucleic Acids Res, 2017. **45**(5): p. e32.

163. Xie, J., et al., *Modeling allele-specific expression at the gene and SNP levels simultaneously by a Bayesian logistic mixed regression model.* BMC Bioinformatics, 2019. **20**(1): p. 530.

164. Zhabotynsky, V., et al., *eQTL mapping using allele-specific count data is computationally feasible, powerful, and provides individual-specific estimates of genetic effects.* PLOS Genetics, 2022. **18**(3): p. e1010076.

165. Zou, J., et al., *Leveraging allele-specific expression to refine fine-mapping for eQTL studies.* bioRxiv, 2018: p. 257279.

166. Wang, X., et al., *Transcriptome-wide identification of novel imprinted genes in neonatal mouse brain.* PLoS One, 2008. **3**(12): p. e3839.

167. Gregg, C., et al., *High-resolution analysis of parent-of-origin allelic expression in the mouse brain.* Science, 2010. **329**(5992): p. 643-8.

168. Langmyhr, M., et al., *Allele-specific expression of Parkinson's disease susceptibility genes in human brain.* Scientific Reports, 2021. **11**(1): p. 504.

169. Lin, M., et al., *Allele-biased expression in differentiating human neurons: implications for neuropsychiatric disorders.* PLoS One, 2012. **7**(8): p. e44017.

170. Tan, A.C., et al., *Allele-specific expression in the germline of patients with familial pancreatic cancer: An unbiased approach to cancer gene discovery.* Cancer Biology & Therapy, 2008. **7**(1): p. 135-144.

171. Wang, Y., et al., *Allele-specific expression of mutated in colorectal cancer (MCC) gene and alternative susceptibility to colorectal cancer in schizophrenia.* Scientific Reports, 2016. **6**(1): p. 26688.

172. McLaren, W., et al., *The Ensembl Variant Effect Predictor.* Genome Biology, 2016. **17**(1): p. 122.

173. Shameer, K., et al., *Interpreting functional effects of coding variants: challenges in proteome-scale prediction, annotation and assessment.* Brief Bioinform, 2016. **17**(5): p. 841-62.

174. Sauna, Z.E. and C. Kimchi-Sarfaty, *Understanding the contribution of synonymous mutations to human disease.* Nature Reviews Genetics, 2011. **12**(10): p. 683-691.

175. Pickrell, J.K., *Joint analysis of functional genomic data and genome-wide association studies of 18 human traits.* Am J Hum Genet, 2014. **94**(4): p. 559-73.

176. Li, Y.I., et al., *RNA splicing is a primary link between genetic variation and disease.* Science, 2016. **352**(6285): p. 600-4.

177. Welter, D., et al., *The NHGRI GWAS Catalog, a curated resource of SNP-trait associations.* Nucleic Acids Res, 2014. **42**(Database issue): p. D1001-6.

178. Lee, C. *Towards the Genetic Architecture of Complex Gene Expression Traits: Challenges and Prospects for eQTL Mapping in Humans*. Genes, 2022. **13**, DOI: 10.3390/genes13020235.

179. Enattah, N.S., et al., *Identification of a variant associated with adult-type hypolactasia.* Nature Genetics, 2002. **30**(2): p. 233-237.

180. Lewinsky, R.H., et al., *T −13910 DNA variant associated with lactase persistence interacts with Oct-1 and stimulates lactase promoter activity in vitro.* Human Molecular Genetics, 2005. **14**(24): p. 3945-3953.

181. Majewski, J. and T. Pastinen, *The study of eQTL variations by RNA-seq: from SNPs to phenotypes.* Trends in Genetics, 2011. **27**(2): p. 72-79.

182. Ge, B., et al., *Global patterns of cis variation in human cells revealed by high-density allelic expression analysis.* Nature Genetics, 2009. **41**(11): p. 1216-1222.

183. Montgomery, S.B., et al., *Transcriptome genetics using second generation sequencing in a Caucasian population.* Nature, 2010. **464**(7289): p. 773-777.

184. Chen, N.-C., et al., *Reference flow: reducing reference bias using multiple population genomes.* Genome Biology, 2021. **22**(1): p. 8.

185. Delaneau, O., et al., *Integrating sequence and array data to create an improved 1000 Genomes Project haplotype reference panel.* Nature Communications, 2014. **5**(1): p. 3934.

186. Delaneau, O., J.-F. Zagury, and J. Marchini, *Improved whole-chromosome phasing for disease and population genetic studies.* Nature Methods, 2013. **10**(1): p. 5-6.

187. Bansal, V., *Integrating read-based and population-based phasing for dense and accurate haplotyping of individual genomes.* Bioinformatics, 2019. **35**(14): p. i242-i248.

188. Pendleton, M., et al., *Assembly and diploid architecture of an individual human genome via single-molecule technologies.* Nature Methods, 2015. **12**(8): p. 780-786.

189. Schadt, E.E., S. Turner, and A. Kasarskis, *A window into third-generation sequencing.* Hum Mol Genet, 2010. **19**(R2): p. R227-40.

190. Roach, Jared C., et al., *Chromosomal Haplotypes by Genetic Phasing of Human Families.* The American Journal of Human Genetics, 2011. **89**(3): p. 382-397.

191. Castel, S.E., et al., *Rare variant phasing and haplotypic expression from RNA sequencing with phASER.* Nature communications, 2016. **7**(1): p. 1-6.

192. Williams, C.R., et al., *Trimming of sequence reads alters RNA-Seq gene expression estimates.* BMC Bioinformatics, 2016. **17**(1): p. 103.

193. Del Fabbro, C., et al., *An extensive evaluation of read trimming effects on Illumina NGS data analysis.* PloS one, 2013. **8**(12): p. e85024-e85024.

194. Zheng, W., L.M. Chung, and H. Zhao, *Bias detection and correction in RNA-Sequencing data.* BMC Bioinformatics, 2011. **12**: p. 290.

195. Risso, D., et al., *GC-Content Normalization for RNA-Seq Data.* BMC Bioinformatics, 2011. **12**(1): p. 480.

196. Liao, Y. and W. Shi, *Read trimming is not required for mapping and quantification of RNA-seq reads at the gene level.* NAR genomics and bioinformatics, 2020. **2**(3): p. lqaa068-lqaa068.

197. Deschamps-Francoeur, G., J. Simoneau, and M.S. Scott, *Handling multi-mapped reads in RNA-seq.* Computational and Structural Biotechnology Journal, 2020. **18**: p. 1569-1576.

198. McDermaid, A., et al., *A New Machine Learning-Based Framework for Mapping Uncertainty Analysis in RNA-Seq Read Alignment and Gene Expression Estimation.* Frontiers in Genetics, 2018. **9**.

199. Mendelevich, A., et al., *Unexpected variability of allelic imbalance estimates from RNA sequencing.* bioRxiv, 2020: p. 2020.02.18.948323.

200. Kilpinen, H., et al., *Common genetic variation drives molecular heterogeneity in human iPSCs.* Nature, 2017. **546**(7658): p. 370-375.

201. Schwartzentruber, J., et al., *Molecular and functional variation in iPSC-derived sensory neurons.* Nature Genetics, 2018. **50**(1): p. 54-61.

202. Ashburner, M., et al., *Gene ontology: tool for the unification of biology. The Gene Ontology Consortium.* Nat Genet, 2000. **25**(1): p. 25-9.

203. Huang da, W., B.T. Sherman, and R.A. Lempicki, *Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists.* Nucleic Acids Res, 2009. **37**(1): p. 1-13.

204. Huang da, W., B.T. Sherman, and R.A. Lempicki, *Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources.* Nat Protoc, 2009. **4**(1): p. 44-57.

205. Eberle, M.A., et al., *A reference data set of 5.4 million phased human variants validated by genetic inheritance from sequencing a three-generation 17-member pedigree.* Genome Res, 2017. **27**(1): p. 157-164.

206. Milhaven, M. and S.P. Pfeifer, *Performance evaluation of six popular short-read simulators.* Heredity, 2023. **130**(2): p. 55-63.

207. Escalona, M., S. Rocha, and D. Posada, *A comparison of tools for the simulation of genomic next-generation sequencing data.* Nature Reviews Genetics, 2016. **17**(8): p. 459-469.

208. Huang, W., et al., *ART: a next-generation sequencing read simulator.* Bioinformatics, 2012. **28**(4): p. 593-4.

209. Ruffalo, M., et al., *Accurate estimation of short read mapping quality for next-generation genome sequencing.* Bioinformatics, 2012. **28**(18): p. i349-i355.

210. Li, H., et al., *The Sequence Alignment/Map format and SAMtools.* Bioinformatics, 2009. **25**(16): p. 2078-9.

211. Holtgrewe, M., *Mason – A Read Simulator for Second Generation Sequencing Data.* Technical Report FU Berlin, 2010.

212. McElroy, K.E., F. Luciani, and T. Thomas, *GemSIM: general, error-model based simulator of next-generation sequencing data.* BMC Genomics, 2012. **13**(1): p. 74.

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

214. Wang, K., M. Li, and H. Hakonarson, *ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data.* Nucleic Acids Research, 2010. **38**(16): p. e164-e164.

215. Eckersley-Maslin, M.A., et al., *Random monoallelic gene expression increases upon embryonic stem cell differentiation.* Dev Cell, 2014. **28**(4): p. 351-65.

216. Houtman, M., et al., *Haplotype-Specific Expression Analysis of MHC Class II Genes in Healthy Individuals and Rheumatoid Arthritis Patients.* Front Immunol, 2021. **12**: p. 707217.

217. Munger, S.C., et al., *RNA-Seq alignment to individualized genomes improves transcript abundance estimates in multiparent populations.* Genetics, 2014. **198**(1): p. 59-73.

218. Yuan, S. and Z. Qin, *Read-mapping using personalized diploid reference genome for RNA sequencing data reduced bias for detecting allele-specific expression.* IEEE International Conference on Bioinformatics and Biomedicine workshops. IEEE International Conference on Bioinformatics and Biomedicine, 2012. **2012**: p. 718-724.

219. Rivas-Astroza, M., et al., *Mapping personal functional data to personal genomes.* Bioinformatics, 2011. **27**(24): p. 3427-9.

220. Sohn, J.-i. and J.-W. Nam, *The present and future of de novo whole-genome assembly.* Briefings in Bioinformatics, 2018. **19**(1): p. 23-40.

221. Webster, T.H., et al., *Identifying, understanding, and correcting technical artifacts on the sex chromosomes in next-generation sequencing data.* GigaScience, 2019. **8**(7): p. giz074.

222. Powers, S., S. Gopalakrishnan, and N. Tintle, *Assessing the Impact of Non-Differential Genotyping Errors on Rare Variant Tests of Association.* Human Heredity, 2011. **72**(3): p. 153-160.

223. Mayer-Jochimsen, M., S. Fast, and N.L. Tintle, *Assessing the Impact of Differential Genotyping Errors on Rare Variant Tests of Association.* PLOS ONE, 2013. **8**(3): p. e56626.

224. Yan, Q., et al., *The impact of genotype calling errors on family-based studies.* Scientific Reports, 2016. **6**(1): p. 28323.

225. Li, J., et al., *Rare variants regulate expression of nearby individual genes in multiple tissues.* PLOS Genetics, 2021. **17**(6): p. e1009596.

226. Cirulli, E.T., et al., *Genome-wide rare variant analysis for thousands of phenotypes in over 70,000 exomes from two cohorts.* Nature Communications, 2020. **11**(1): p. 542.

227. Hernandez, R.D., et al., *Ultrarare variants drive substantial cis heritability of human gene expression.* Nat Genet, 2019. **51**(9): p. 1349-1355.

228. Zhao, J., et al., *A Burden of Rare Variants Associated with Extremes of Gene Expression in Human Peripheral Blood.* Am J Hum Genet, 2016. **98**(2): p. 299-309.

229. Vijaya Satya, R., N. Zavaljevski, and J. Reifman, *A new strategy to reduce allelic bias in RNA-Seq readmapping.* Nucleic Acids Research, 2012. **40**(16): p. e127-e127.

230. Panousis, N.I., et al., *Allelic mapping bias in RNA-sequencing is not a major confounder in eQTL studies.* Genome Biology, 2014. **15**(9): p. 467.

231. Wratten, L., A. Wilm, and J. Göke, *Reproducible, scalable, and shareable analysis pipelines with bioinformatics workflow managers.* Nature Methods, 2021. **18**(10): p. 1161-1168.

232. Di Tommaso, P., et al., *The impact of Docker containers on the performance of genomic pipelines.* PeerJ, 2015. **3**: p. e1273-e1273.

233. Tiwari, K., et al., *Reproducibility in systems biology modelling.* Mol Syst Biol, 2021. **17**(2): p. e9982.

234. Garijo, D., et al., *Quantifying reproducibility in computational biology: the case of the tuberculosis drugome.* PLoS One, 2013. **8**(11): p. e80278.

235. Grüning, B., et al., *Practical Computational Reproducibility in the Life Sciences.* Cell Syst, 2018. **6**(6): p. 631-635.

236. Di Tommaso, P., et al., *Nextflow enables reproducible computational workflows.* Nature Biotechnology, 2017. **35**(4): p. 316-319.

237. *Docker overview*. 02.06.2022]; Available from: https://docs.docker.com/get-started/overview/.

238. Song, Z., et al., *nf-gwas-pipeline: A Nextflow Genome-Wide Association Study Pipeline.* J Open Source Softw, 2021. **6**(59).

239. Twesigomwe, D., et al., *StellarPGx: A Nextflow Pipeline for Calling Star Alleles in Cytochrome P450 Genes.* Clin Pharmacol Ther, 2021. **110**(3): p. 741-749.

240. Mpangase, P.T., et al., *nf-rnaSeqCount: A Nextflow pipeline for obtaining raw read counts from RNA-seq data.* S Afr Comput J, 2021. **33**(2).

241. Hölzer, M. and M. Marz, *PoSeiDon: a Nextflow pipeline for the detection of evolutionary recombination events and positive selection.* Bioinformatics, 2021. **37**(7): p. 1018-1020.

242. Liu, X., J.R. Bienkowska, and W. Zhong, *GeneTEFlow: A Nextflow-based pipeline for analysing gene and transposable elements expression from RNA-Seq data.* PLoS One, 2020. **15**(8): p. e0232994.

243. *nf-core.*

244. Ewels, P.A., et al., *The nf-core framework for community-curated bioinformatics pipelines.* Nature Biotechnology, 2020. **38**(3): p. 276-278.

245. *Nextflow Basic Concepts*. Available from: https://www.nextflow.io/docs/latest/basic.html.

246. *Singularity.*

247. *GitHub*. Available from: https://github.com.

248. *AWS S3 Illumina iGenomes*. Available from: https://github.com/ewels/AWS-iGenomes.

249. Sims, D., et al., *Sequencing depth and coverage: key considerations in genomic analyses.* Nature Reviews Genetics, 2014. **15**(2): p. 121-132.

250. Kousathanas, A., et al., *Whole-genome sequencing reveals host factors underlying critical COVID-19.* Nature, 2022. **607**(7917): p. 97-103.

251. Yandell, M., et al., *A probabilistic disease-gene finder for personal genomes.* Genome Research, 2011. **21**(9): p. 1529-1542.

252. Yun, S. and S. Yun, *Masking as an effective quality control method for next-generation sequencing data analysis.* BMC Bioinformatics, 2014. **15**(1): p. 382.

253. Lawlor, B. and P. Walsh, *Engineering bioinformatics: building reliability, performance and productivity into bioinformatics software.* Bioengineered, 2015. **6**(4): p. 193-203.

254. Kanwal, S., et al., *Investigating reproducibility and tracking provenance – A genomic workflow case study.* BMC Bioinformatics, 2017. **18**(1): p. 337.

255. Nekrutenko, A. and J. Taylor, *Next-generation sequencing data interpretation: enhancing reproducibility and accessibility.* Nature Reviews Genetics, 2012. **13**(9): p. 667-672.

256. Committee on the Review of Omics-Based Tests for Predicting Patient Outcomes in Clinical, T., et al., *Evolution of Translational Omics: Lessons Learned and the Path Forward*, in *Evolution of Translational Omics: Lessons Learned and the Path Forward*, C.M. Micheel, S.J. Nass, and G.S. Omenn, Editors. 2012, National Academies Press (US)

257. Hothorn, T. and F. Leisch, *Case studies in reproducibility.* Briefings in Bioinformatics, 2011. **12**(3): p. 288-300.

258. Wood, D.L.A., et al., *Recommendations for Accurate Resolution of Gene and Isoform Allele-Specific Expression in RNA-Seq Data.* PLOS ONE, 2015. **10**(5): p. e0126911.

259. Pirinen, M., et al., *Assessing allele-specific expression across multiple tissues from RNA-seq read data.* Bioinformatics, 2015. **31**(15): p. 2497-504.

260. Sun, W., *A statistical framework for eQTL mapping using RNA-seq data.* Biometrics, 2012. **68**(1): p. 1-11.

261. Liu, Y., et al., *Genome-wide analysis of expression QTL (eQTL) and allele-specific expression (ASE) in pig muscle identifies candidate genes for meat quality traits.* Genetics Selection Evolution, 2020. **52**(1): p. 59.

262. Liu, Y., et al., *Trait correlated expression combined with eQTL and ASE analyses identified novel candidate genes affecting intramuscular fat.* BMC Genomics, 2021. **22**(1): p. 805.

263. Cooper, R.D. and H.B. Shaffer, *Allele-specific expression and gene regulation help explain transgressive thermal tolerance in non-native hybrids of the endangered California tiger salamander (Ambystoma californiense).* Mol Ecol, 2021. **30**(4): p. 987-1004.

264. Tangwancharoen, S., B.X. Semmens, and R.S. Burton, *Allele-Specific Expression and Evolution of Gene Regulation Underlying Acute Heat Stress Response and Local Adaptation in the Copepod Tigriopus californicus.* J Hered, 2020. **111**(6): p. 539-547.

265. Tucci, V., et al., *Genomic Imprinting and Physiological Processes in Mammals.* Cell, 2019. **176**(5): p. 952-965.

266. Galupa, R. and E. Heard, *X-Chromosome Inactivation: A Crossroads Between Chromosome Architecture and Gene Regulation.* Annu Rev Genet, 2018. **52**: p. 535-566.

267. Chess, A., *Monoallelic Gene Expression in Mammals.* Annu Rev Genet, 2016. **50**: p. 317-327.

268. Gimelbrant, A., et al., *Widespread monoallelic expression on human autosomes.* Science, 2007. **318**(5853): p. 1136-40.

269. Gendrel, A.-V., et al., *Random monoallelic expression of genes on autosomes: Parallels with X-chromosome inactivation.* Seminars in Cell & Developmental Biology, 2016. **56**: p. 100-110.

270. Kilpinen, H., et al., *Coordinated effects of sequence variation on DNA binding, chromatin structure, and transcription.* Science, 2013. **342**(6159): p. 744-7.

271. Chen, J., et al., *A uniform survey of allele-specific binding and expression over 1000-Genomes-Project individuals.* Nature Communications, 2016. **7**(1): p. 11101.

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

273. Zhang, H.S.B.P.L.T., *Statistical methods for overdispersion in mRNA-seq count data.* Open Bioinformatics Journal, 2013.

274. Yuan, Y., et al., *Differentially Expressed Heterogeneous Overdispersion Genes Testing for Count Data.* bioRxiv, 2023.

275. Cai, G., et al., *Local sequence and sequencing depth dependent accuracy of RNA-seq reads.* BMC Bioinformatics, 2017. **18**(1): p. 364.

276. Kumasaka, N., A.J. Knights, and D.J. Gaffney, *Fine-mapping cellular QTLs with RASQUAL and ATAC-seq.* Nature Genetics, 2016. **48**(2): p. 206-213.

277. Zhang, X. and J.J. Emerson, *Inferring the genetic architecture of expression variation from replicated high throughput allele-specific expression experiments.* bioRxiv, 2019: p. 699074.

278. Nariai, N., et al., *A Bayesian approach for estimating allele-specific expression from RNA-Seq data with diploid genomes.* BMC Genomics, 2016. **17**(1): p. 2.

279. Battle, A., et al., *Characterizing the genetic basis of transcriptome diversity through RNA-sequencing of 922 individuals.* Genome Research, 2014. **24**(1): p. 14-24.

280. Shabalin, A.A., *Matrix eQTL: ultra fast eQTL analysis via large matrix operations.* Bioinformatics, 2012. **28**(10): p. 1353-8.

281. Flutre, T., et al., *A statistical framework for joint eQTL analysis in multiple tissues.* PLoS Genet, 2013. **9**(5): p. e1003486.

282. Sakamoto, Y., S. Sereewattanawoot, and A. Suzuki, *A new era of long-read sequencing for cancer genomics.* J Hum Genet, 2020. **65**(1): p. 3-10.

283. Sharp, K., et al., *Phasing for medical sequencing using rare variants and large haplotype reference panels.* Bioinformatics, 2016. **32**(13): p. 1974-1980.

284. Glinos, D.A., et al., *Transcriptome variation in human tissues revealed by long-read sequencing.* bioRxiv, 2021: p. 2021.01.22.427687.

285. Morisse, P., T. Lecroq, and A. Lefebvre, *Long-read error correction: a survey and qualitative comparison.* bioRxiv, 2021: p. 2020.03.06.977975.

286. Kundaje, A., et al., *Integrative analysis of 111 reference human epigenomes.* Nature, 2015. **518**(7539): p. 317-30.

287. Nasser, J., et al., *Genome-wide enhancer maps link risk variants to disease genes.* Nature, 2021. **593**(7858): p. 238-243.

288. Corradin, O. and P.C. Scacheri, *Enhancer variants: evaluating functions in common disease.* Genome Medicine, 2014. **6**(10): p. 85.

289. Kikuchi, M., et al., *Enhancer variants associated with Alzheimer's disease affect gene expression via chromatin looping.* BMC Medical Genomics, 2019. **12**(1): p. 128.

290. Soldner, F., et al., *Parkinson-associated risk variant in distal enhancer of α-synuclein modulates target gene expression.* Nature, 2016. **533**(7601): p. 95-99.

291. Hannon, E., et al., *Genetic risk variants for brain disorders are enriched in cortical H3K27ac domains.* Molecular Brain, 2019. **12**(1): p. 7.

292. Pelikan, R.C., et al., *Enhancer histone-QTLs are enriched on autoimmune risk haplotypes and influence gene expression within chromatin networks.* Nat Commun, 2018. **9**(1): p. 2905.

293. Mendelevich, A., et al., *Replicate sequencing libraries are important for quantification of allelic imbalance.* Nature Communications, 2021. **12**(1): p. 3370.

294. Sun, X., et al., *Nanopore Sequencing and Its Clinical Applications.* Methods Mol Biol, 2020. **2204**: p. 13-32.

295. Smith, M. and P.L. Flodman, *Expanded Insights Into Mechanisms of Gene Expression and Disease Related Disruptions.* Front Mol Biosci, 2018. **5**: p. 101.

296. Sawyer, S.L., et al., *Utility of whole-exome sequencing for those near the end of the diagnostic odyssey: time to address gaps in care.* Clin Genet, 2016. **89**(3): p. 275-84.

297. Chong, J.X., et al., *The Genetic Basis of Mendelian Phenotypes: Discoveries, Challenges, and Opportunities.* Am J Hum Genet, 2015. **97**(2): p. 199-215.

298. Yang, Y., et al., *Molecular findings among patients referred for clinical whole-exome sequencing.* Jama, 2014. **312**(18): p. 1870-9.

299. Finotello, F. and B. Di Camillo, *Measuring differential gene expression with RNA-seq: challenges and strategies for data analysis.* Briefings in Functional Genomics, 2015. **14**(2): p. 130-142.

300. Gandal, M.J., et al., *Transcriptome-wide isoform-level dysregulation in ASD, schizophrenia, and bipolar disorder.* Science, 2018. **362**(6420): p. eaat8127.

301. Mathys, H., et al., *Single-cell transcriptomic analysis of Alzheimer's disease.* Nature, 2019. **570**(7761): p. 332-337.

302. Uhlen, M., et al., *A pathology atlas of the human cancer transcriptome.* Science, 2017. **357**(6352).

303. Porcu, E., et al., *Differentially expressed genes reflect disease-induced rather than disease-causing changes in the transcriptome.* Nature Communications, 2021. **12**(1): p. 5647.

304. Hibbs, K., et al., *Differential Gene Expression in Ovarian Carcinoma: Identification of Potential Biomarkers.* The American Journal of Pathology, 2004. **165**(2): p. 397-414.

305. Pan, Y., et al., *Analysis of differential gene expression profile identifies novel biomarkers for breast cancer.* Oncotarget, 2017. **8**(70): p. 114613-114625.

306. Hernandez, D.G., et al., *Integration of GWAS SNPs and tissue specific expression profiling reveal discrete eQTLs for human traits in blood and brain.* Neurobiol Dis, 2012. **47**(1): p. 20-8.

307. Albert, F.W. and L. Kruglyak, *The role of regulatory variation in complex traits and disease.* Nature Reviews Genetics, 2015. **16**(4): p. 197-212.

308. Huang, N., et al., *Characterising and predicting haploinsufficiency in the human genome.* PLoS Genet, 2010. **6**(10): p. e1001154.

309. Dang, V.T., et al., *Identification of human haploinsufficient genes and their genomic proximity to segmental duplications.* European Journal of Human Genetics, 2008. **16**(11): p. 1350-1357.

310. Morrill, S.A. and A. Amon, *Why haploinsufficiency persists.* Proceedings of the National Academy of Sciences, 2019. **116**(24): p. 11866-11871.

311. Sisodiya, S.M., et al., *Role of SOX2 Mutations in Human Hippocampal Malformations and Epilepsy.* Epilepsia, 2006. **47**(3): p. 534-542.

312. Hamdan, F.F., et al., *De novo mutations in moderate or severe intellectual disability.* PLoS Genet, 2014. **10**(10): p. e1004772.

313. Meechan, D.W., et al., *When half is not enough: gene expression and dosage in the 22q11 deletion syndrome.* Gene Expr, 2007. **13**(6): p. 299-310.

314. Fitzgerald, T.W., et al., *Large-scale discovery of novel genetic causes of developmental disorders.* Nature, 2015. **519**(7542): p. 223-228.

315. Shihab, H.A., et al., *HIPred: an integrative approach to predicting haploinsufficient genes.* Bioinformatics, 2017. **33**(12): p. 1751-1757.

316. Cassa, C.A., et al., *Estimating the selective effects of heterozygous protein-truncating variants from human exome data.* Nat Genet, 2017. **49**(5): p. 806-810.

317. Han, X., et al., *Distinct epigenomic patterns are associated with haploinsufficiency and predict risk genes of developmental disorders.* Nature Communications, 2018. **9**(1): p. 2138.

318. van der Wijst, M.G.P., et al., *Single-cell RNA sequencing identifies celltype-specific cis-eQTLs and co-expression QTLs.* Nat Genet, 2018. **50**(4): p. 493-497.

319. Knowles, D.A., et al., *Determining the genetic basis of anthracycline-cardiotoxicity by molecular response QTL mapping in induced cardiomyocytes.* Elife, 2018. **7**.

320. Huang, Q.Q., et al., *Neonatal genetics of gene expression reveal potential origins of autoimmune and allergic disease risk.* Nat Commun, 2020. **11**(1): p. 3761.

321. Alasoo, K., et al., *Genetic effects on promoter usage are highly context-specific and contribute to complex traits.* eLife, 2019. **8**: p. e41673.

322. Iossifov, I., et al., *De novo gene disruptions in children on the autistic spectrum.* Neuron, 2012. **74**(2): p. 285-99.

323. Neale, B.M., et al., *Patterns and rates of exonic de novo mutations in autism spectrum disorders.* Nature, 2012. **485**(7397): p. 242-5.

324. O'Roak, B.J., et al., *Sporadic autism exomes reveal a highly interconnected protein network of de novo mutations.* Nature, 2012. **485**(7397): p. 246-50.

325. Sanders, S.J., et al., *De novo mutations revealed by whole-exome sequencing are strongly associated with autism.* Nature, 2012. **485**(7397): p. 237-41.

326. Steinberg, J., et al., *Haploinsufficiency predictions without study bias.* Nucleic Acids Res, 2015. **43**(15): p. e101.

327. Shaikh, T.H., et al., *High-resolution mapping and analysis of copy number variations in the human genome: a data resource for clinical and research applications.* Genome Res, 2009. **19**(9): p. 1682-90.

328. Zhang, Y., et al., *PTWAS: investigating tissue-relevant causal molecular mechanisms of complex traits using probabilistic TWAS analysis.* Genome Biology, 2020. **21**(1): p. 232.

329. Messner, B., et al., *Cadmium Is a Novel and Independent Risk Factor for Early Atherosclerosis Mechanisms and In Vivo Relevance.* Arteriosclerosis, Thrombosis, and Vascular Biology, 2009. **29**(9): p. 1392-1398.

330. Zug, R., *Developmental disorders caused by haploinsufficiency of transcriptional regulators: a perspective based on cell fate determination.* Biol Open, 2022. **11**(1).

331. Fotaki, V., et al., *Dyrk1A haploinsufficiency affects viability and causes developmental delay and abnormal brain morphology in mice.* Mol Cell Biol, 2002. **22**(18): p. 6636-47.

332. Sarel-Gallily, R., et al., *Genome-wide analysis of haploinsufficiency in human embryonic stem cells.* Cell Reports, 2022. **38**(13): p. 110573.

333. Favé, M.J., et al., *Gene-by-environment interactions in urban populations modulate risk phenotypes.* Nat Commun, 2018. **9**(1): p. 827.

334. Claringbould, A. and J.B. Zaugg, *Enhancers in disease: molecular basis and emerging treatment strategies.* Trends Mol Med, 2021. **27**(11): p. 1060-1073.

335. Rentas, S., et al., *Diagnosing Cornelia de Lange syndrome and related neurodevelopmental disorders using RNA sequencing.* Genet Med, 2020. **22**(5): p. 927-936.

336. Bonder, M.J., et al., *Identification of rare and common regulatory variants in pluripotent cells using population-scale transcriptomics.* Nat Genet, 2021. **53**(3): p. 313-321.

337. Murdock, D.R., et al., *Transcriptome-directed analysis for Mendelian disease diagnosis overcomes limitations of conventional genomic testing.* J Clin Invest, 2021. **131**(1).

338. Sonawane, A.R., et al., *Understanding Tissue-Specific Gene Regulation.* Cell Rep, 2017. **21**(4): p. 1077-1088.

339. Pillon, N.J., et al., *Transcriptomic profiling of skeletal muscle adaptations to exercise and inactivity.* Nature Communications, 2020. **11**(1): p. 470.

340. Gaur, P., et al., *Temporal transcriptome analysis suggest modulation of multiple pathways and gene network involved in cell-cell interaction during early phase of high altitude exposure.* PLoS One, 2020. **15**(9): p. e0238117.

341. Small, K.S., et al., *Regulatory variants at KLF14 influence type 2 diabetes risk via a female-specific effect on adipocyte size and body composition.* Nature Genetics, 2018. **50**(4): p. 572-580.

342. Moradifard, S., et al., *Analysis of microRNA and Gene Expression Profiles in Alzheimer's Disease: A Meta-Analysis Approach.* Scientific Reports, 2018. **8**(1): p. 4767.

343. Wang, Z., M. Gerstein, and M. Snyder, *RNA-Seq: a revolutionary tool for transcriptomics.* Nat Rev Genet, 2009. **10**(1): p. 57-63.

344. Zhang, T., et al., *Cell-type-specific eQTL of primary melanocytes facilitates identification of melanoma susceptibility genes.* Genome Res, 2018. **28**(11): p. 1621-1635.

345. Zeller T, W.P., Szymczak S, Rotival M, Schillert A, Castagne R, et al., *Genetics and Beyond – The Transcriptome of Human Monocytes and Disease Susceptibility.* PLoS ONE, 2010.

346. Eichler, E.E., et al., *Missing heritability and strategies for finding the underlying causes of complex disease.* Nat Rev Genet, 2010. **11**(6): p. 446-50.

347. t Hoen, P.A.C., et al., *Reproducibility of high-throughput mRNA and small RNA sequencing across laboratories.* Nature Biotechnology, 2013. **31**(11): p. 1015-1022.

348. M, P.N., et al., *Estimating the Allele-Specific Expression of SNVs From 10× Genomics Single-Cell RNA-Sequencing Data.* Genes (Basel), 2020. **11**(3).

349. Papin, J.A., et al., *Improving reproducibility in computational biology research.* PLOS Computational Biology, 2020. **16**(5): p. e1007881.

350. Ferraro, N.M., et al., *Transcriptomic signatures across human tissues identify functional rare genetic variation.* Science, 2020. **369**(6509): p. eaaz5900.

351. Rakocevic, G., et al., *Fast and accurate genomic analyses using genome graphs.* Nat Genet, 2019. **51**(2): p. 354-362.

352. Li, H., X. Feng, and C. Chu, *The design and construction of reference pangenome graphs with minigraph.* Genome Biology, 2020. **21**(1): p. 265.

353. Kim, D., et al., *Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype.* Nature Biotechnology, 2019. **37**(8): p. 907-915.

354. Dong, L., J. Wang, and G. Wang, *BYASE: a Python library for estimating gene and isoform level allele-specific expression.* Bioinformatics, 2020. **36**(19): p. 4955-4956.

355. Melé, M., et al., *Human genomics. The human transcriptome across tissues and individuals.* Science, 2015. **348**(6235): p. 660-5.

356. Wang, E.T., et al., *Alternative isoform regulation in human tissue transcriptomes.* Nature, 2008. **456**(7221): p. 470-6.

357. Byron, S.A., et al., *Translating RNA sequencing into clinical diagnostics: opportunities and challenges.* Nature Reviews Genetics, 2016. **17**(5): p. 257-271.

358. Valle, L., et al., *Germline allele-specific expression of TGFBR1 confers an increased risk of colorectal cancer.* Science, 2008. **321**(5894): p. 1361-5.

359. Sen, A., et al., *Allele-specific expression reveals genes with recurrent cis-regulatory alterations in high-risk neuroblastoma.* Genome Biology, 2022. **23**(1): p. 71.

APPENDIX

APPENDIX 1. PAC REFERENCE MANUAL

REFERENCE MANUAL FOR PAC

The PAC reference manual provides with detailed information on PAC software. The manual starts with description how to obtain and run PAC. Following this, a description of each process within the PAC, starting from first to last process, is described.

TABLE OF CONTENTS

1.1 SOFTWARE SETUP

PAC requires Nextflow, Java v8+, and a docker or singularity (depending on the profile the user selects).

To download PAC, download it from the GitHub with the following command:

```
git clone https://github.com/anna-saukkonen/PAC.git
```

To download Nextlow, run following command:

```
curl -fsSL get.nextflow.io | bash
```

1.2 PAC PROCESSES

This sections describes each process within PAC, as the software is written. However, once a process has available input files available from previous processes, it will start running to speed up the run time by parallelisation. See thesis section 3.3 for more information.

1.2.1 SETTING PARAMETERS

```
/*
 * Defines some parameters in order to specify the refence genomes
 * and read pairs by using the command line options
 */


params.genome        = params.genomes[ params.genome_version
]?.genome
params.annot         = params.genomes[ params.genome_version
]?.annot
params.gencode_bed   = params.genomes[ params.genome_version
]?.gencode_bed


// Check if genome exists in the config file
if (params.genomes && params.genome_version &&
!params.genomes.containsKey(params.genome_version)) {
    exit 1, "The provided genome '${params.genome_version}' is not
available. Currently the available genomes are
${params.genomes.keySet().join(", ")}. Please check your spelling."
}


if (!params.variants) exit 1, "Path to phased variants has to be
specified!"
if (!params.reads) exit 1, "Path to reads has to be specified!"
if (!params.id) exit 1, "Sample ID not supplied, needs to be same as
in the VCF"
```

```
Channel
  .fromFilePairs(params.reads)
  .ifEmpty { exit 1, "Cannot find any reads matching: ${reads}\nNB:
Path needs to be enclosed in quotes!\n"}
  .into {reads_ch; reads_ch1; reads_ch2; reads_ch3}
```

The first step, although not a process, checks that all essential parameters are specified when executing PAC. The essential parameters are the genome version, path to RNA-seq reads, path to variants VCF file, and sample ID. If any of these are missing, PAC stops the run and gives an error message stating which parameter is missing. This section also places RNA-seq reads into multiple channels as multiple processes take them as inputs.

1.2.2 PROCESS READ_LENGTH

```
process read_length {

  input:
    set val(id), file(reads) from reads_ch

  output:
    file "readLength_file.txt" into readlen_file_ch

  shell:
  '''
  gunzip -c *_1.{fq,fastq}.gz | sed '2q;d' | wc -m | awk '{print $1-
1}' >> readLength_file.txt
  '''
}

readlen_file_ch.map { it.text.trim().toInteger() }.into {
read_len_ch1; read_len_ch2; read_len_ch3; read_len_ch4;
read_len_ch5; read_len_ch6 }
```

Input: This process takes in RNA-seq read files as input file.

Process: Custom bash script calculates the read length.

Output: The output is a file with read length value that is used in the downstream processes throughout PAC.

Outside of the process the value from the output file is placed into different channels as multiple processes need this value.

1.2.3 PROCESS PREPARE_STAR_GENOME_INDEX

```
process prepare_star_genome_index {

  input:
    path genome from params.genome
    path annot from params.annot
    val x from read_len_ch1
    val cpus from params.cpus

  output:
    path STARhaploid into genome_dir_ch

  script:
  """
  mkdir STARhaploid
  STAR --runMode genomeGenerate \
       --genomeDir STARhaploid \
       --genomeFastaFiles ${genome} \
       --sjdbGTFfile ${annot} \
       --sjdbOverhang ${x} \
       --runThreadN ${cpus}
  """}
```

Input: This process takes in the reference genome specified in options, annotation file, read length information from the previous process, and number of cpus as an optional input.

Process: It then generates a genome index with STAR --runMode genomeGenerate.

Output: The genome indices in STARhaploid directory. This step is necessary for standard alignment in the next process.

1.2.4 PROCESS RNASEQ_MAPPING_STAR

```
process rnaseq_mapping_star {

  input:
    path genome from params.genome
    path STARhaploid from genome_dir_ch
    set val(id), file(reads) from reads_ch1
    val x from read_len_ch2
    val id from params.id
    val cpus from params.cpus

  output:
    tuple \
      val(id), \

path("${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.bam"),
\

path("${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.bam.bai
") into aligned_bam_ch

  script:
  """
  # Align reads to genome
  STAR --genomeDir STARhaploid \
       --readFilesIn ${reads} \
```

```
        --readFilesCommand zcat \
        --runThreadN ${cpus} \
        --outSAMstrandField intronMotif \
        --outFilterMultimapNmax 30 \
        --alignIntronMax 1000000 \
        --alignMatesGapMax 1000000 \
        --outMultimapperOrder Random \
        --outSAMunmapped Within \
        --outSAMattrIHstart 0 \
        --outFilterIntronMotifs RemoveNoncanonicalUnannotated \
        --sjdbOverhang ${x} \
        --outFilterMismatchNmax ${(x-(x%13))/13} \
        --outSAMattributes NH nM NM MD HI \
        --outSAMattrRGline  ID:${id} PU:Illumina PL:Illumina
LB:${id}.SOFT.NOTRIM SM:${id}.SOFT.NOTRIM CN:Seq_centre \
        --outSAMtype BAM SortedByCoordinate \
        --twopassMode Basic \
        --outFileNamePrefix ${id}.SOFT.NOTRIM.STAR.pass2. \
        --outSAMprimaryFlag AllBestScore
  # Index the BAM file
  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.bam
  """
}
```

Input: This process takes in the reference genome, genome index generated from process prepare_star_genome_index, read length information from process read_length, the RNA-seq reads, sample ID and number of cpus as an optional input.

Process: The step aligns reads to the reference genome and indexes the BAM file with SAMtools index. This process provides the standard alignment that the user can use as a comparison for the PAC results. The output also feeds into the phaser_step.

Output: BAM and BAM.bai files of mapped RNA-seq reads.

1.2.5 PROCESS CLEAN_UP_READS

```
process clean_up_reads {

  input:
    tuple val(id), path(bam), path(index) from aligned_bam_ch
    path variants from params.variants
    val id from params.id
    val cpus from params.cpus

  output:
    path ("STAR_original/phaser_version.bam") into phaser_ch
    path ("STAR_original/phaser_version.bam.bai") into phaser_bai_ch
    path
("STAR_original/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.o
ut.PP.UM.bam") into pp_um_ch

  script:
  """
  mkdir STAR_original
  #KEEP ONLY PROPERLY PAIRED READS
  samtools view -@ ${cpus} -f 0x0002 -b -o
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.bam
  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam
  #KEEP UNIQUELY MAPPED READS
  samtools view -h
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam | grep
-P "NH:i:1\t|^@" | samtools view -bS - >
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
  #Create BAM compatible with PHASER:
  gunzip -c ${variants} | grep -q 'chr' || (samtools view -h
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam |
sed -e 's/chr//' >> phaser_version.sam; samtools view -bh
phaser_version.sam >> phaser_version.bam; samtools index
phaser_version.bam; rm phaser_version.sam)
```

```
  gunzip -c  ${variants} | grep -q 'chr' && (samtools view -bh
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam >>
phaser_version.bam; samtools index phaser_version.bam)
  mv phaser_version.bam STAR_original/phaser_version.bam
  mv phaser_version.bam.bai STAR_original/phaser_version.bam.bai
  mv
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
STAR_original/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out
.PP.UM.bam
  """
}
```

Input: The process takes in the BAM files generated from process rnaseq_mapping_star, variants VCF file, sample ID and number of cpus as an optional input.

Process: In this step the mapped RNA-seq reads are filtered. SAMtools is used to keep only properly paired (where the read orientation of read pairs is as expected and the gap between them is likely based on sequencing technology) and uniquely mapped (reads mapping to single location) reads. The BAM is then created that is compatible for downstream process phaser_step.

Output: Properly paired and uniquely mapped BAM file, phaser_step process compatible BAM and BAI files in separate channels.

1.2.6 PROCESS PHASER_STEP

```
process phaser_step {

  input:
  path variants from params.variants
  path ("phaser_version.bam") from phaser_ch
  path ("phaser_version.bam.bai") from phaser_bai_ch
  val id from params.id
  val cpus from params.cpus
```

```
  output:
  path ("${id}_output_phaser.vcf") into (phaser_out_ch1,
phaser_out_ch2)


  script:
  """
  tabix -f -p vcf ${variants}
  python2 /phaser/phaser/phaser.py --vcf ${variants} --bam
phaser_version.bam --paired_end 1 --mapq 0 --baseq 10 --isize 0 --
include_indels 1 --sample ${id} --id_separator + --pass_only 0 --
gw_phase_vcf 1 --threads ${cpus} --o ${id}_output_phaser
  gunzip ${id}_output_phaser.vcf.gz
  rm phaser_version.bam
  rm phaser_version.bam.bai
  """
}
```

Input: Variants VCF file, BAM and BAI files from process clean_up_reads, sample ID and number of cpus.

Process: This step uses phASER to phase variants incorporating aligned RNA-seq reads. phASER uses a read-aware mode for phasing. It selects RNA-seq reads where there are two variants, that can be split across larger genomic distances due to splicing, hence it can incorporate variants over longer distances and thereby improve phasing. This allows better phasing of rare variants and longer haplotypes.

Output: Phased variants VCF file.

1.2.7 PROCESS CREATE_PARENTAL_GENOMES

```
process create_parental_genomes {

  input:
    path genome from params.genome
    path annot from params.annot
    path ("${id}_output_phaser.vcf") from phaser_out_ch1
```

```
    val id from params.id
    path gencode_bed from params.gencode_bed

  output:
    path ("STAR_2Gen_Ref/maternal.chain") into maternal_chain_ch
    path ("STAR_2Gen_Ref/paternal.chain") into paternal_chain_ch
    path ("STAR_2Gen_Ref/${id}_maternal.fa") into (mat_fa1, mat_fa2)
    path ("STAR_2Gen_Ref/${id}_paternal.fa") into (pat_fa1, pat_fa2)
    path ("STAR_2Gen_Ref/mat_annotation.gtf") into
(mat_annotation_ch1, mat_annotation_ch2)
    path ("STAR_2Gen_Ref/not_lifted_m.txt") into not_lift_m_ch
    path ("STAR_2Gen_Ref/pat_annotation.gtf") into
(pat_annotation_ch1, pat_annotation_ch2)
    path ("STAR_2Gen_Ref/not_lifted_p.txt") into not_lift_p_ch
    path ("STAR_2Gen_Ref/map_over.txt") into adjusted_ref_ch
    path ("STAR_2Gen_Ref/${id}_output_phaser.mother.vcf.gz") into
mothervcf_ch
    path ("STAR_2Gen_Ref/${id}_output_phaser.father.vcf.gz") into
fathervcf_ch
    path ("STAR_2Gen_Ref/mat.bed") into mat_bed_ch
    path ("STAR_2Gen_Ref/pat.bed") into pat_bed_ch

  script:
  """
  mkdir STAR_2Gen_Ref
  java -Xmx10000m -jar /vcf2diploid_v0.2.6a/vcf2diploid.jar -id
${id} -chr ${genome} -vcf ${id}_output_phaser.vcf -outDir
STAR_2Gen_Ref > logfile.txt

  liftOver -gff ${annot} STAR_2Gen_Ref/maternal.chain
STAR_2Gen_Ref/mat_annotation.gtf STAR_2Gen_Ref/not_lifted_m.txt
  liftOver -gff ${annot} STAR_2Gen_Ref/paternal.chain
STAR_2Gen_Ref/pat_annotation.gtf STAR_2Gen_Ref/not_lifted_p.txt
  liftOver ${gencode_bed} STAR_2Gen_Ref/maternal.chain
STAR_2Gen_Ref/mat.bed STAR_2Gen_Ref/not_bed_lifted_m.txt
  liftOver ${gencode_bed} STAR_2Gen_Ref/paternal.chain
STAR_2Gen_Ref/pat.bed STAR_2Gen_Ref/not_bed_lifted_p.txt
```

```
  cat STAR_2Gen_Ref/chr1_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr2_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr3_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr4_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr5_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr6_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr7_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr8_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr9_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr10_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr11_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr12_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr13_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr14_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr15_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr16_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr17_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr18_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr19_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa
```

```
  cat STAR_2Gen_Ref/chr20_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr21_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chr22_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chrX_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chrY_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa

  cat STAR_2Gen_Ref/chrM_${id}_maternal.fa >>
STAR_2Gen_Ref/${id}_maternal.fa


  cat STAR_2Gen_Ref/chr1_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr2_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr3_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr4_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr5_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr6_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr7_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr8_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr9_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr10_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr11_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr12_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr13_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa
```

```
  cat STAR_2Gen_Ref/chr14_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr15_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr16_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr17_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr18_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr19_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr20_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr21_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chr22_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chrX_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chrY_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa

  cat STAR_2Gen_Ref/chrM_${id}_paternal.fa >>
STAR_2Gen_Ref/${id}_paternal.fa


  sed 's/\\*/N/g' STAR_2Gen_Ref/${id}_maternal.fa >
STAR_2Gen_Ref/${id}_maternal.hold.fa

  mv STAR_2Gen_Ref/${id}_maternal.hold.fa
STAR_2Gen_Ref/${id}_maternal.fa


  sed 's/\\*/N/g' STAR_2Gen_Ref/${id}_paternal.fa >
STAR_2Gen_Ref/${id}_paternal.hold.fa

  mv STAR_2Gen_Ref/${id}_paternal.hold.fa
STAR_2Gen_Ref/${id}_paternal.fa

  mv ${id}_output_phaser.vcf STAR_2Gen_Ref/${id}_output_phaser.vcf

  cd STAR_2Gen_Ref/


  perl ${baseDir}/bin/adjust_reference.pl ${id}_output_phaser.vcf
${id}
```

```
  perl ${baseDir}/bin/adjust_reference_vcf.pl
${id}_output_phaser.vcf ${id}

  grep "^#" ${id}_output_phaser.mother.vcf >
${id}_output_phaser.mother.s.vcf

  grep -v "^#" ${id}_output_phaser.mother.vcf | sort -k1,1V -k2,2g
>> ${id}_output_phaser.mother.s.vcf

  grep "^#" ${id}_output_phaser.father.vcf >
${id}_output_phaser.father.s.vcf

  grep -v "^#" ${id}_output_phaser.father.vcf | sort -k1,1V -k2,2g
>> ${id}_output_phaser.father.s.vcf

  mv ${id}_output_phaser.mother.s.vcf ${id}_output_phaser.mother.vcf

  mv ${id}_output_phaser.father.s.vcf ${id}_output_phaser.father.vcf

  bcftools view ${id}_output_phaser.mother.vcf -Oz -o
${id}_output_phaser.mother.vcf.gz

  bcftools view ${id}_output_phaser.father.vcf -Oz -o
${id}_output_phaser.father.vcf.gz

  tabix ${id}_output_phaser.father.vcf.gz

  tabix ${id}_output_phaser.mother.vcf.gz

  """

}
```

Input: The reference genome, annotation file, phased variants VCF file from process phaser_step, sample ID and BED annotation file.

Process: This step creates personalised parental genomes. The phased variants are incorporated into the reference genome using vcf2diploid, generating maternal and paternal genomes. liftOver is then used to generate GTF and BED files with adjusted genomic coordinates for maternal and paternal genomes. This is because the coordinates will be shifted due to indels present in the VCF file. The custom scripts generate maternal and paternal VCF files where the heterozygous site coordinates are shifted to the maternal and paternal genomes.

Output: Maternal and paternal genomes, chain files for both genomes that are needed for liftOver (not needed in the downsteam process but output ensures files can be found on users' system should they need them for their own analysis), maternal and paternal GTF and BED files, files containing regions not lifted for maternal and paternal genomes, maternal and paternal VCF files.

1.2.8 PROCESS STAR_REFERENCE_MATERNAL_GENOMES

```
process STAR_reference_maternal_genomes {


  input:
    path ("STAR_2Gen_Ref/${id}_maternal.fa") from mat_fa1
    path ("STAR_2Gen_Ref/mat_annotation.gtf") from
mat_annotation_ch1
    val x from read_len_ch3
    val id from params.id
    val cpus from params.cpus


  output:
    path Maternal_STAR into Maternal_STAR_ch


  script:
  """
  mkdir Maternal_STAR
  STAR --runMode genomeGenerate --genomeDir Maternal_STAR --
genomeFastaFiles STAR_2Gen_Ref/${id}_maternal.fa --sjdbGTFfile
STAR_2Gen_Ref/mat_annotation.gtf --sjdbOverhang ${x} --runThreadN
${cpus} --outTmpDir mat
  """


}
```

Input: Maternal genome and maternal GTF file from process create_parental_genomes, read length information from process read_length, sample ID and number of cpus.

Process: This step generates maternal genome index with STAR --runMode genomeGenerate. This step feeds into map_maternal_gen_filter, where the RNA-seq reads are mapped to the maternal genomes.

Output: Maternal genome indices in Maternal_STAR directory.

1.2.9 PROCESS STAR_REFERENCE_PATERNAL_GENOMES

```
process STAR_reference_paternal_genomes {


  input:
    path ("STAR_2Gen_Ref/${id}_paternal.fa") from pat_fa1
    path ("STAR_2Gen_Ref/pat_annotation.gtf") from
pat_annotation_ch1
    val x from read_len_ch4
    val id from params.id
    val cpus from params.cpus


  output:
    path Paternal_STAR into Paternal_STAR_ch


  script:
  """
  mkdir Paternal_STAR
  STAR --runMode genomeGenerate --genomeDir Paternal_STAR --
genomeFastaFiles STAR_2Gen_Ref/${id}_paternal.fa --sjdbGTFfile
STAR_2Gen_Ref/pat_annotation.gtf --sjdbOverhang ${x} --runThreadN
${cpus} --outTmpDir pat
  """


}
```

This process is identical to process STAR_reference_maternal_genomes above but it is performed on the paternal genome.

1.2.10 PROCESS MAP_PATERNAL_GEN_FILTER

```
process map_paternal_gen_filter {
  tag "$id"


  input:
    path Paternal_STAR from Paternal_STAR_ch
```

```
    set val(id), file(reads) from reads_ch2
    path ("STAR_2Gen_Ref/pat_annotation.gtf") from
pat_annotation_ch2
    path ("STAR_2Gen_Ref/${id}_paternal.fa") from pat_fa2
    val x from read_len_ch5
    val id from params.id
    val cpus from params.cpus


  output:
    path
("STAR_Paternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.o
ut.PP.UM.bam") into (paternal_mapgen_ch1, paternal_mapgen_ch2)
    path ("STAR_Paternal/${id}.RSEM.TEST.genome.PP.SM.bam") into
pat_rsem_ch


  script:
  """
  STAR --genomeDir Paternal_STAR \
       --runThreadN ${cpus} \
       --quantMode TranscriptomeSAM \
       --readFilesIn $reads \
       --readFilesCommand zcat \
       --outSAMstrandField intronMotif \
       --outFilterMultimapNmax 30 \
       --alignIntronMax 1000000 \
       --alignMatesGapMax 1000000 \
       --outMultimapperOrder Random \
       --outSAMunmapped Within \
       --outSAMattrIHstart 0 \
       --outFilterIntronMotifs RemoveNoncanonicalUnannotated \
       --sjdbOverhang ${x} \
       --outFilterMismatchNmax ${(x-(x%13))/13} \
       --outSAMattributes NH nM NM MD HI \
       --outSAMattrRGline  ID:${id}.SOFT.NOTRIM PU:Illumina
PL:Illumina LB:${id}.SOFT.NOTRIM SM:${id}.SOFT.NOTRIM CN:Seq_centre
\
```

```
        --outSAMtype BAM SortedByCoordinate \
        --twopassMode Basic \
        --outFileNamePrefix ${id}.SOFT.NOTRIM.STAR.pass2. \
        --outSAMprimaryFlag AllBestScore


  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.bam
  #KEEP ONLY PROPERLY PAIRED READS
  samtools view -@ ${cpus} -f 0x0002 -b -o
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.bam
  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam
  #KEEP UNIQUELY MAPPED READS
  samtools view -h
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam | grep
-P "NH:i:1\t|^@" | samtools view -bS - >
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
  mkdir STAR_Paternal
  mv
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
STAR_Paternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out
.PP.UM.bam
  ##Create RSEM Files:
  mkdir RSEM_MAT_GEN
  /RSEM/rsem-prepare-reference -p ${cpus} --gtf
STAR_2Gen_Ref/pat_annotation.gtf STAR_2Gen_Ref/${id}_paternal.fa
RSEM_MAT_GEN/RSEM_MAT_GEN
  /RSEM/rsem-calculate-expression --bam --output-genome-bam --
sampling-for-bam -p ${cpus} --paired-end
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.toTranscriptome.out.bam
RSEM_MAT_GEN/RSEM_MAT_GEN ${id}.RSEM.TEST
  samtools view -@ ${cpus} -f 0x0002 -b -o
${id}.RSEM.TEST.genome.PP.bam ${id}.RSEM.TEST.genome.bam
  samtools sort -@ ${cpus} -o ${id}.RSEM.TEST.genome.PP.s.bam
${id}.RSEM.TEST.genome.PP.bam
  mv ${id}.RSEM.TEST.genome.PP.s.bam ${id}.RSEM.TEST.genome.PP.bam
  samtools index ${id}.RSEM.TEST.genome.PP.bam
```

```
  samtools view -h ${id}.RSEM.TEST.genome.PP.bam | grep -P
"ZW:f:1|^@" | samtools view -bS - > ${id}.RSEM.TEST.genome.PP.SM.bam
  samtools index ${id}.RSEM.TEST.genome.PP.SM.bam
  mv ${id}.RSEM.TEST.genome.PP.SM.bam
STAR_Paternal/${id}.RSEM.TEST.genome.PP.SM.bam
  """

}
```

Input: Paternal genome indices from process STAR_reference_paternal_genomes, RNA-seq reads, paternal genome and GTF file from process create_parental_genomes, read length information from process read_length, sample ID and number of cpus.

Process: In this step the RNA-seq reads are aligned to the paternal genome with STAR. The BAM file generated from this is indexed and filtered with SAMtools to keep only properly paired and uniquely mapped reads.

RSEM is used to index the paternal genome. Following this, RSEM is used with the STAR transcriptome.bam to map the same RNA-seq reads with RSEM instead. In this case, reads that would map to multiple locations are not discarded but are allocated one location. All uniquely mapped reads are used to calculate the expression of each of these loci, and then the multi-mapping reads are allocated a location based on these weights. The allocation is based on probabilities based on ratios of uniquely mapped reads from genomic loci where the multi-mapping read aligns to. The file is then filtered with SAMtools to keep only properly paired reads.

Output: BAM file of mapped reads to paternal genome and BAM file generated with RSEM.

1.2.11 PROCESS MAP_MATERNAL_GEN_FILTER

```
process map_maternal_gen_filter {
  tag "$id"
  input:
    path Maternal_STAR from Maternal_STAR_ch
```

```
    set val(id), file(reads) from reads_ch3
    path ("STAR_2Gen_Ref/mat_annotation.gtf") from
mat_annotation_ch2
    path ("STAR_2Gen_Ref/${id}_maternal.fa") from mat_fa2
    val x from read_len_ch6
    val id from params.id
    val cpus from params.cpus


  output:
    path
("STAR_Maternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.o
ut.PP.UM.bam") into (maternal_mapgen_ch1, maternal_mapgen_ch2)
    path ("STAR_Maternal/${id}.RSEM.TEST.genome.PP.SM.bam") into
mat_rsem_ch


  script:
  """
  STAR --genomeDir Maternal_STAR \
       --runThreadN ${cpus} \
       --quantMode TranscriptomeSAM \
       --readFilesIn $reads \
       --readFilesCommand zcat \
       --outSAMstrandField intronMotif \
       --outFilterMultimapNmax 30 \
       --alignIntronMax 1000000 \
       --alignMatesGapMax 1000000 \
       --outMultimapperOrder Random \
       --outSAMunmapped Within \
       --outSAMattrIHstart 0 \
       --outFilterIntronMotifs RemoveNoncanonicalUnannotated \
       --sjdbOverhang ${x} \
       --outFilterMismatchNmax ${(x-(x%13))/13} \
       --outSAMattributes NH nM NM MD HI \
       --outSAMattrRGline  ID:${id}.SOFT.NOTRIM PU:Illumina
PL:Illumina LB:${id}.SOFT.NOTRIM SM:${id}.SOFT.NOTRIM CN:Seq_centre
\
```

```
        --outSAMtype BAM SortedByCoordinate \
        --twopassMode Basic \
        --outFileNamePrefix ${id}.SOFT.NOTRIM.STAR.pass2. \
        --outSAMprimaryFlag AllBestScore
  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.bam
  #KEEP ONLY PROPERLY PAIRED READS
  samtools view -@ ${cpus} -f 0x0002 -b -o
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.bam
  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam
  #KEEP UNIQUELY MAPPED READS
  samtools view -h
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.bam | grep
-P "NH:i:1\t|^@" | samtools view -bS - >
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
  samtools index
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
  mkdir STAR_Maternal
  mv
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out.PP.UM.bam
STAR_Maternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out
.PP.UM.bam
  ##Create RSEM Files:
  mkdir RSEM_MAT_GEN
  /RSEM/rsem-prepare-reference -p ${cpus} --gtf
STAR_2Gen_Ref/mat_annotation.gtf STAR_2Gen_Ref/${id}_maternal.fa
RSEM_MAT_GEN/RSEM_MAT_GEN
  /RSEM/rsem-calculate-expression --bam --output-genome-bam --
sampling-for-bam -p ${cpus} --paired-end
${id}.SOFT.NOTRIM.STAR.pass2.Aligned.toTranscriptome.out.bam
RSEM_MAT_GEN/RSEM_MAT_GEN ${id}.RSEM.TEST


  samtools view -@ ${cpus} -f 0x0002 -b -o
${id}.RSEM.TEST.genome.PP.bam ${id}.RSEM.TEST.genome.bam
  samtools sort -@ ${cpus} -o ${id}.RSEM.TEST.genome.PP.s.bam
${id}.RSEM.TEST.genome.PP.bam
  mv ${id}.RSEM.TEST.genome.PP.s.bam ${id}.RSEM.TEST.genome.PP.bam
```

```
  samtools view -h ${id}.RSEM.TEST.genome.PP.bam | grep -P
"ZW:f:1|^@" | samtools view -bS - > ${id}.RSEM.TEST.genome.PP.SM.bam

  samtools index ${id}.RSEM.TEST.genome.PP.SM.bam

  mv ${id}.RSEM.TEST.genome.PP.SM.bam
STAR_Maternal/${id}.RSEM.TEST.genome.PP.SM.bam

  """

}
```

This process is identical to process map_paternal_gen_filter but performed on the maternal genome.

1.2.12 PROCESS EXTRA_READS_RSEM

```
process extra_reads_rsem {


  input:

    path
("STAR_Maternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.o
ut.PP.UM.bam") from maternal_mapgen_ch1

    path ("STAR_Maternal/${id}.RSEM.TEST.genome.PP.SM.bam") from
mat_rsem_ch

    path
("STAR_Paternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.o
ut.PP.UM.bam") from paternal_mapgen_ch1

    path ("STAR_Paternal/${id}.RSEM.TEST.genome.PP.SM.bam") from
pat_rsem_ch

    val id from params.id


  output:

    path ("Maternal.RSEM.bam") into mat_rsembam

    path ("Paternal.RSEM.bam") into pat_rsembam


  script:

  """

  samtools view
STAR_Maternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out
.PP.UM.bam | cut -f1 | sort | uniq >> maternal_tags_UM.txt
```

```
  samtools view STAR_Maternal/${id}.RSEM.TEST.genome.PP.SM.bam | cut
-f1 | sort | uniq > maternal_tags_UM.RSEM.txt

  perl ${baseDir}/bin/filter_rsem.pl maternal

  samtools view
STAR_Paternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out
.PP.UM.bam | cut -f1 | sort | uniq >> paternal_tags_UM.txt

  samtools view STAR_Paternal/${id}.RSEM.TEST.genome.PP.SM.bam | cut
-f1 | sort | uniq > paternal_tags_UM.RSEM.txt

  perl ${baseDir}/bin/filter_rsem.pl paternal

  samtools view -H STAR_Maternal/${id}.RSEM.TEST.genome.PP.SM.bam >
Maternal.RSEM.sam

  samtools view STAR_Maternal/${id}.RSEM.TEST.genome.PP.SM.bam |
grep -Fwf extra.rsem.maternal.txt | sed -e 's/339\tchr/83\tchr/' |
sed -e 's/355\tchr/99\tchr/' | sed -e 's/403\tchr/147\tchr/' | sed -
e 's/419\tchr/163\tchr/' >> Maternal.RSEM.sam

  samtools view -bS Maternal.RSEM.sam -o Maternal.RSEM.bam

  samtools view -H STAR_Paternal/${id}.RSEM.TEST.genome.PP.SM.bam >
Paternal.RSEM.sam

  samtools view STAR_Paternal/${id}.RSEM.TEST.genome.PP.SM.bam |
grep -Fwf extra.rsem.paternal.txt | sed -e 's/339\tchr/83\tchr/' |
sed -e 's/355\tchr/99\tchr/' | sed -e 's/403\tchr/147\tchr/' | sed -
e 's/419\tchr/163\tchr/' >> Paternal.RSEM.sam

  samtools view -bS Paternal.RSEM.sam -o Paternal.RSEM.bam

  """

}
```

Input: Filtered BAM file from process map_maternal_gen_filter and map_paternal_gen_filter, RSEM sampled BAM files from map_maternal_gen_filter and map_paternal_gen_filter, and sample ID.

Process: Custom script gets the extra multi-mapping reads (which now only have one location allocated by weight in the previous step) that are aligned in RSEM, but not in STAR and creates a file extra.rsem.maternal/paternal.txt. Then a new RSEM BAM file is created containing only these extra reads.

Output: BAM file for maternal and paternal extra reads that originally aligned to multiple locations, now with a single location.

1.2.13 PROCESS ADD_RSEMREADS_BAM

```
process add_rsemreads_bam {
  publishDir "$params.outdir/", mode: 'copy'

  input:
    path ("Maternal.RSEM.bam") from mat_rsembam
    path ("Paternal.RSEM.bam") from pat_rsembam
    path
("STAR_Paternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.o
ut.PP.UM.bam") from paternal_mapgen_ch2
    path
("STAR_Maternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.o
ut.PP.UM.bam") from maternal_mapgen_ch2
    path ("STAR_2Gen_Ref/map_over.txt") from adjusted_ref_ch
    path ("${id}_output_phaser.vcf") from phaser_out_ch2
    val id from params.id
    val cpus from params.cpus
    path
("STAR_original/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.o
ut.PP.UM.bam") from pp_um_ch
    path ("STAR_2Gen_Ref/${id}_output_phaser.mother.vcf.gz") from
mothervcf_ch
    path ("STAR_2Gen_Ref/${id}_output_phaser.father.vcf.gz") from
fathervcf_ch
    path ("STAR_2Gen_Ref/mat.bed") from mat_bed_ch
    path ("STAR_2Gen_Ref/pat.bed") from pat_bed_ch
    path gencode_bed from params.gencode_bed



  output:
    path ("results*.txt")
    path ("${id}_gene_level_ae.txt")

  script:
  """
```

```
  samtools merge Maternal.RSEM.STAR.bam
STAR_Maternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out
.PP.UM.bam Maternal.RSEM.bam

  samtools merge Paternal.RSEM.STAR.bam
STAR_Paternal/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out
.PP.UM.bam Paternal.RSEM.bam

  samtools view Maternal.RSEM.STAR.bam | cut -f1 | sort | uniq >>
maternal_tags.txt

  samtools view Paternal.RSEM.STAR.bam | cut -f1 | sort | uniq >>
paternal_tags.txt

  cat maternal_tags.txt paternal_tags.txt | sort | uniq -u >>
unique_tags.txt

  cat maternal_tags.txt paternal_tags.txt | sort | uniq -d >>
duplicate_tags.txt

  samtools view Maternal.RSEM.STAR.bam | grep -Fwf
duplicate_tags.txt >> tempout_mat.sam

  samtools view Paternal.RSEM.STAR.bam | grep -Fwf
duplicate_tags.txt >> tempout_pat.sam

  sort -k 1,1 tempout_mat.sam > tempout_mat.sort.sam

  sort -k 1,1 tempout_pat.sam > tempout_pat.sort.sam

  perl ${baseDir}/bin/filter_2genomes.pl tempout_mat.sort.sam
tempout_pat.sort.sam

  cat maternal_wins.txt unique_tags.txt > maternal_wins_final.txt

  cat paternal_wins.txt unique_tags.txt > paternal_wins_final.txt

  samtools view -H Maternal.RSEM.STAR.bam > final_mat.sam

  samtools view -H Paternal.RSEM.STAR.bam > final_pat.sam

  samtools view Maternal.RSEM.STAR.bam | grep -Fwf
maternal_wins_final.txt >> final_mat.sam

  samtools view Paternal.RSEM.STAR.bam | grep -Fwf
paternal_wins_final.txt >> final_pat.sam

  samtools view -bS final_mat.sam -o final_mat.bam

  samtools sort -@ ${cpus} -o final_mat.sorted.bam final_mat.bam

  samtools index final_mat.sorted.bam

  samtools view -bS final_pat.sam -o final_pat.bam

  samtools sort -@ ${cpus} -o final_pat.sorted.bam final_pat.bam

  samtools index final_pat.sorted.bam

  perl ${baseDir}/bin/compare_basic_map.pl ${id}_output_phaser.vcf
STAR_original/${id}.SOFT.NOTRIM.STAR.pass2.Aligned.sortedByCoord.out
```

```
.PP.UM.bam ${id} results_1genome_${id}.SOFT.NOTRIM_baq.txt
results_1genome_${id}.SOFT.NOTRIM.txt

  perl ${baseDir}/bin/compare_2genomes.pl STAR_2Gen_Ref/map_over.txt
${id}_output_phaser.vcf final_mat.sorted.bam final_pat.sorted.bam
${id} results_2genomes_${id}.RSEM.STAR.SOFT.NOTRIM_baq.txt
results_2genomes_${id}.RSEM.STAR.SOFT.NOTRIM.txt

  tabix STAR_2Gen_Ref/${id}_output_phaser.mother.vcf.gz

  tabix STAR_2Gen_Ref/${id}_output_phaser.father.vcf.gz

  python2 /phaser/phaser/phaser.py --vcf
STAR_2Gen_Ref/${id}_output_phaser.mother.vcf.gz --bam
final_mat.sorted.bam --paired_end 1 --mapq 0 --baseq 10 --isize 0 --
include_indels 1 --sample ${id} --id_separator + --pass_only 0 --
gw_phase_vcf 1 --threads ${cpus} --o ${id}_mat_output_phaser

  python2 /phaser/phaser/phaser.py --vcf
STAR_2Gen_Ref/${id}_output_phaser.father.vcf.gz --bam
final_pat.sorted.bam --paired_end 1 --mapq 0 --baseq 10 --isize 0 --
include_indels 1 --sample ${id} --id_separator + --pass_only 0 --
gw_phase_vcf 1 --threads ${cpus} --o ${id}_pat_output_phaser

  python2 /phaser/phaser_gene_ae/phaser_gene_ae.py --
haplotypic_counts ${id}_mat_output_phaser.haplotypic_counts.txt --
features STAR_2Gen_Ref/mat.bed --id_separator +  --o
${id}_maternal_phaser_gene_ae.txt

  python2 /phaser/phaser_gene_ae/phaser_gene_ae.py --
haplotypic_counts ${id}_pat_output_phaser.haplotypic_counts.txt --
features STAR_2Gen_Ref/pat.bed --id_separator +  --o
${id}_paternal_phaser_gene_ae.txt

  perl ${baseDir}/bin/merge_gene_level.pl ${gencode_bed}
${id}_maternal_phaser_gene_ae.txt ${id}_paternal_phaser_gene_ae.txt
${id}

  """


}
```

Input: Maternal and paternal extra reads from RSEM generated in process extra_reads_rsem; BAM file of reads mapped to maternal and paternal genomes from map_maternal_gen_filter and map_paternal_gen_filter; map_over, and and maternal and paternal bed files with adjusted coordinates, and maternal and paternal phased VCF file from process create_parental_genomes, phased VCF file from process phaser_step, sample ID, number of cpus, properly paired and uniquely mapped reads to the reference genome from process clean_up_reads; and GENCODE BED file.

Process: For each parental genome, the STAR and RSEM BAM files are merged. Then PAC finds reads only aligned in one parent and not the other. When the reads are aligned in both maternal and paternal genomes, a custom script (filter_2genomes.pl) selects the best alignment for each read from the two alignments (scoring reads by the number of matching nucleotides minus two times the number of indel positions, drawing at random when the two alignments have equal scores).

Then two custom scripts (compare_basic_map.pl and compare_2genomes.pl) are used to count the number of alleles at each heterozygous site. Initially, this is done with standard alignment. Then the same is performed for two genomes parental alignment using the liftOver variant files.

Then phASER is used to generate the gene-level calculations using the VCF files and GTF files from each parent (generated in process create_parental_genomes). PAC then produces allele counts at haplotypic level using phASER Gene AE.

Finally, the last custom script (merge_gene_level.pl) merges the gene level counts across the two parents.

Output: The results files: site and haplotype level allelic counts and single genome alignment for comparison.

1.3 OUTPUT

PAC generates 5 output files:

- haplotype level ASE calls:
    6. 'id'_gene_level_ae.txt

Haplotype level ASE results columns	Description
contig	chromosome
start	gene start position
stop	gene end position
name	gene name
aCount	haplotype a coverage
bCount	haplotype b coverage
totalCount	total coverage

Figure 1. **Columns and their descriptions for haplotype level ASE results from PAC output.**
The 'id'_gene_level_ae.txt contains this file format.

- single nucleotide level ASE calls from PAC:
    7. results_2genomes_'id'.RSEM.STAR.SOFT.NOTRIM_baq.txt
    8. results_2genomes_'id'.RSEM.STAR.SOFT.NOTRIM.txt
- single nucleotide level ASE calls based on standard single genome mapping for comparison:
    9. results_1genome_'id'.SOFT.NOTRIM_baq.txt
    10. results_1genome_'id'.SOFT.NOTRIM.txt

Single nucleotide level ASE results columns	Description
Chr	chromosome
Pos	position along chromosome
RefAl	reference allele
AltAl	alternative allele
MapRef	reference allele coverage
MapAlt	alternative allele coverage
MapRatio	reference allele ratio
Mapcov	total coverage at the site

Figure 2. **Columns and their descriptions for single nucleotide level ASE results from PAC output.**
The results_2genomes_ID.RSEM.STAR.SOFT.NOTRIM_baq.txt, results_2genomes_ID.RSEM.STAR.SOFT.NOTRIM.txt, results_1genome_ID.SOFT.NOTRIM_baq.txt and results_1genome_ID.SOFT.NOTRIM.txt contain this file format.