The Pennsylvania State University

The Graduate School

ACCURATE MEASUREMENT OF VARIANTS WITH CONTINUOUS RANGES OF FREQUENCIES USING NEXT-GENERATION SEQUENCING

A Dissertation in

Integrative Biosciences

by

Nicholas B Stoler

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

December 2020

The dissertation of Nicholas B Stoler was reviewed and approved by the following:

Anton Nekrutenko

Professor of Biochemistry and Molecular Biology

Dissertation Advisor

Chair of Committee

Kateryna Makova

Pentz Professor of Biology

Paul Medvedev

Associate Professor of Computer Science and Engineering

Associate Professor of Biochemistry and Molecular Biology

Francesca Chiaromonte

Professor of Statistics and Public Health Sciences

Michael DeGiorgio

Associate Professor of Biology

George Perry

Associate Professor of Anthropology and Biology

Chair, Intercollege Graduate Degree Program in Bioinformatics and Genomics

Abstract

The detection of genetic variants is central to the study of disease, evolution, and populations. Fortunately, next-generation sequencing has enabled genome-wide variant detection at affordable prices. However, detection of low-frequency variants, such as those involved in tumor evolution, mitochondrial disease, and antibiotic resistance remains a challenge because of the high signal to noise ratio in standard sequencing technologies. For applications like these, the accuracy and quality of sequencing data becomes paramount.

The genomics community has worked to address this need in many ways. First, a great deal of effort has gone into understanding the quality of the raw data produced by current sequencing methods. And second, a series of innovative methods has been developed for improving on the raw data.

Many studies have examined the error rate and sequence biases of contemporary sequencing platforms. But the data examined are small numbers of samples in controlled environments. And many manufacturers have introduced many new technologies in recent years with potential effects on sequencing quality. In this dissertation I develop a method of identifying sequencing errors which can be easily automated and applied retroactively on existing datasets. I demonstrate its utility by performing a survey of 1,943 public datasets from the Sequence Read Archive. With this survey, I am able to uncover differences in the error rates and biases of current Illumina sequencing platforms. I find that the error rates of public datasets from the more expensive, high-throughput instruments are lower and less variable than those of smaller-scale machines. But I also find great variation within each platform, especially the lower-end ones.

To improve on these error rates, a series of groups have developed methods based on consensus sequencing. This principle utilizes DNA barcodes to be able to combine multiple reads from the same molecule. The highest-fidelity design, duplex sequencing, can improve on the accuracy of standard sequencing by four orders of magnitude.

But there are limitations in the standard software for processing and combining the raw reads of duplex sequencing. Existing tools require a reference sequence to produce any

consensus sequences from the reads. This limits analysis to systems with a suitable reference, and it can introduce reference bias into the consensus sequences. Another issue is the occurrence of errors in the barcodes used to identify reads originating from the same molecule. Standard duplex processing tools simply discard reads affected by barcode errors.

Here, I present Du Novo, a tool built to process duplex sequencing reads without the need for a reference. Using real and simulated reads, I show that Du Novo is able to provide nearly the same accuracy as the existing pipeline, even as it yields more data. In simulations, Du Novo was able to detect 95% of variants at 0.01% minor allele frequency, with 0 false positives.

I also describe great improvements over the first version of Du Novo. After several performance improvements, including the replacement of the core multiple sequence aligner, Du Novo 2.0 is able to perform the alignments up to 10x faster. Another improvement to Du Novo is the addition of an error correction pipeline which can recover reads with errors in their barcodes. This feature is able to increase the yield of final consensus sequences by up to 23%.

Du Novo is the first tool able to perform reference-free processing of duplex sequencing data, and the first to correct barcode errors in the process. These features enable the analysis of more sample types, and with greater accuracy and yield than ever.

Table of Contents

List of Figures

List of Tables

Acknowledgments

Funding

All work presented in this dissertation was supported by The Huck Institutes for the Life Sciences at The Pennsylvania State University. Chapter 2 presents work supported by the National Institute of General Medical Sciences under award number T32 GM102057 and the National Human Genome Research Institute under award number U41 HG005542. The work in chapters 2 and 3 were supported by award R01 GM116044 from the National Institute of General Medical Sciences, and a grant with the Pennsylvania Department of Health using Tobacco Settlement and CURE Funds. The Department specifically disclaims any responsibility for any analyses, responsibility, or conclusions. Chapter 3 presents work supported by the Eberly College of Science's Office of Science Engagement and the Institute for CyberScience at the Pennsylvania State University. Funding for the analysis of the impact of barcode error correction was provided by the Linz Institute of Technology under award number LIT213201001, the Austrian Science Fund under award number FWFP30867000, and a J-4096 Schrödinger Fellowship. The work presented in Chapters 3 and 4 was supported by the National Human Genome Research Institute under award number U41 HG006620, the National Institute of Allergy and Infectious Diseases under award number R01 AI134384, and the National Science Foundation under award number 1661497. Any opinions, findings, and conclusions or recommendations expressed in this thesis are those of the author and do not necessarily reflect the views of the funding agencies.

Personal

I would like to extend my gratitude to everyone who supported me through my entire time at Penn State. First, I would like to thank my advisor Anton Nekrutenko. I feel very fortunate to have found a guide through this process whose approach to research and supervision I appreciate so much. He is always encouraging, and he always helped me navigate the tricky parts of both analysis and working in the system of academic science. And it is thanks

to him that I was able to pursue projects centered around my biggest passions: software development and accuracy in genomics.

I would also like to thank everyone on my committee for their time: Anton, Kateryna Makova, Francesca Chiaromonte, Michael DeGiorgio, and Paul Medvedev. I collaborated almost continuously with Kateryna, whose eye for analysis and care for getting things right I always appreciated. I am thankful for Francesca's statistical advice, and moreover the friendly atmosphere she creates. In his class, Mike led me through the dark arts of population genetics, and I always enjoyed our chats about the field. And Paul was an invaluable resource when I had questions about tools or algorithmic ideas.

One group who greatly enriched my experience at Penn State was the Galaxy Team. It is through them that I learned how a team works together to build software. And they are an incredibly open and generous team who lead an equally amazing community. I owe certain team members a particular debt of gratitude for their time and effort attending to my requests, answering my questions, and giving me advice: Martin Cech, Dave Bouvier, Nate Coraor, Dan Blankenberg, and John Chilton. I would also like to honor an irreplaceable leader of the Galaxy Team who we sadly lost recently. I enjoyed the little time I had with James Taylor, and have great respect for his passion for open science and care for doing things the right way.

There are many in the Nekrutenko and Makova labs who I would like to acknowledge. Boris Rebolledo Jaramillo taught me a lot about being an effective researcher right at the start of my graduate career. And Han Mei has been a great source of help and camaraderie toward the end. In Kateryna's lab, Marcia Shu-Wei Su has been a great friend and collaborator. Barbara Arbeithuber is a collaborator whose experimental acumen has facilitated the greatest projects of my graduate career. And I would like to thank Wilfried Guiblet, Samarth Rangavittal, and Monika Cechova for their friendship, camaraderie, and hospitality.

And there are those in the wider Penn State community whose assistance I must acknowledge. Bob Harris always has time for advice or an interesting discussion. Cathy Riemer and Rico Burhans work tirelessly to facilitate the work of everyone who depends on the systems they maintain. I owe a special thanks to Cooduvalli Shashikant, who was a steadfast ally during my time in the program. And George Perry has been carrying on that legacy. I also must thank Mike Radis and everyone in the administrative teams for their repeated help.

Outside of Penn State, I want to acknowledge Irene Tiemann-Boege and her whole lab, who are great collaborators, as well as some of the first and most loyal users of Du Novo. My tool owes most of its quality and reliability to their patient feedback.

And finally I want to extend my deepest appreciation to those in my personal life who supported me through this process. My family always had my back and provided constant encouragement. Tejaswini Mishra gave endless support, advice, and patience. And I could always rely on my friends for commiseration and a boost in morale.

Preface

Experimental data presented here were generated by Barbara Arbeithuber, unless specified otherwise. Du Novo and the overlap error detection method were implemented by me. Data analysis was performed by me, with contributions from Anton Nekrutenko and Wilfried Guiblet. Analysis of the impact of barcode error correction presented in Figure 3.2 and 3.3 was performed by Irene Tiemann-Boege and members of her lab at Johannes Kepler University, Linz, Austria (especially Monika Heinzl, Gundula Povysil, and Renato Salazar). The manuscripts in Chapters 2, 3, and 4 were prepared by Anton Nekrutenko and me.

1 Introduction

1.1 Motivation

1.1.1 The importance of genetic variation

Analysis of genetic variation has been the central problem in genetics and evolutionary biology since its early days. The rate of evolution of a population is limited by the amount of variation in it (Fisher 1930). It is often said that mutation is the engine of evolution (Hershberg 2015).

1.1.1.1 Variation in humans

Studying variation is crucial for understanding evolution and the changes in populations. But most importantly, variation is the key to understanding some of the most destructive diseases affecting humans. Alzheimer's disease, obesity, schizophrenia, bipolar disorder, alcoholism, type 1 diabetes, and thyroid cancer have all been found to be mainly caused by genetics (Gatz et al. 2006; Walley, Blakemore, and Froguel 2006; Sullivan, Kendler, and Neale 2003; Smoller and Finn 2003; Stacey, Clarke, and Schumann 2009; Hyttinen et al. 2003; Czene, Lichtenstein, and Hemminki 2002).

In the last decade, large-scale studies have been able to identify thousands of locations in the genome containing disease-associated variants (MacArthur et al. 2017). However, the vast majority of the genetic basis of disease remains undiscovered (Manolio et al. 2009). More sophisticated analysis of genetic variants is necessary to uncover this missing heritability (Bomba, Walter, and Soranzo 2017).

1.1.1.2 Variation in haploid and mixed systems

Humans, being diploid organisms, are a special genetic case. Most living organisms are not diploid. And even in humans, there are many situations where investigators are interested in variants which do not fit the standard diploid frequencies of 0%, 50%, and 100%.

The fundamental difference between a subject with a fixed ploidy and one where variants may occur at any frequency is that the latter is examined as a population of DNA. For

example, viral and prokaryotic samples are essentially populations of individual genomes. In that case, if we are interested in anything beyond fixed differences, we must consider variants in a range of frequencies. Another example is metagenomic sequencing, where the subject is a population of different species. Also, sequencing organelles like mitochondria or chloroplasts is essentially a population study similar to examining microorganism samples. In this case, the populations are locked inside eukaryotic cells. And even when we are examining eukaryotic organisms themselves, we are still examining a population, unless single-cell sequencing is used. Many studies only seek diploid genotypes (heterozygous or homozygous), but investigators are increasingly interested in somatic variation within samples. One case where examining somatic variation is explicitly necessary is the study of tumor evolution. Studies of immunological mosaicism are another case where somatic differences are the goal. And cell-free DNA is quite literally a "population of DNA". (Salk, Schmitt, and Loeb 2018)

1.1.2 The importance of low-frequency variants

In most of these examples of subjects consisting of a population of DNA, one could restrict the scope of a study to only the variants fixed in the population. These fixed variants are usually the most important, but not the only important ones. In mitochondria, for example, many disorders are linked to alleles whose frequency is proportional to the severity of the disease (Wallace and Chalkia 2013). Thus, fixed variants are the most impactful, but ones present at lower frequency are still disease-causing, and can present with different symptoms (Wallace and Chalkia 2013). Also, because of the mitochondrial bottleneck in oogenesis, a very low-frequency variant in the parental generation can be passed to the next generation at a much greater frequency (Cree et al. 2008). This means that even if it is not important to identify variants below a certain frequency for a patient's own health, it may still be important for their family planning.

1.2 Variant detection

1.2.1 Beginnings

As late as the 1960s, there was no practical way to survey variation across a population (Hubby and Lewontin 1966). In 1966, Hubby and Lewontin described a method they had devised to estimate the fraction of loci in a population which were heterozygous. The method relies on electrophoretically separating purified proteins. This method only directly measures charge on polypeptides, but this can be a proxy for genotypes. A variation in a genotype can alter the charge of its protein product. This change can be detected via gel electrophoresis. Using this method, Hubby and Lewontin were able to estimate a lower bound of 11.5% for the fraction of loci which were heterozygous in an average individual (Lewontin and Hubby 1966).

Since this early method, many new methods have been developed to more directly and efficiently assay individual variants. In the 1970s, the discovery of restriction enzymes enabled the development of the restriction fragment length polymorphism (RFLP) assay, (Botstein et al. 1980) which is able to directly detect single nucleotide polymorphisms (SNPs), the most common type of variant in humans (Frazer et al. 2009).

1.2.2 High throughput techniques

The Human Genome Project revolutionized genotyping approaches. A number of high-throughput techniques have been developed that make use of short oligonucleotide probes annealing at the site of a specific SNP. For example, **allele-specific oligonucleotide probes** is a technique where two forms of oligonucleotides are created. By varying the sequence in the middle of the oligonucleotides, each is made to hybridize only with one of two possible alleles. Fluorescent markers attached to the probes can reveal what proportion of the molecules in a sample hybridized with each oligonucleotide. (Syvänen 2001)

By incorporating allele-specific oligonucleotides into microarray chips, investigators were able to genotype half a million SNPs at once by 2006 (Pe'er et al. 2006). This was the start of the era of genome-wide association studies (GWAS). These studies aimed to determine the variants responsible for common, hereditary diseases. Arrays were designed which genotyped

enough SNPs to mark every region in the genome. Assaying thousands of individuals with (and without) a disease allowed researchers to determine which regions were associated with a disease. In the first two years of published GWAS results, over 220 studies found evidence of association between over 300 loci and 80 phenotypes (pathological and not) (Frazer et al. 2009). By 2017, there were over 70,000 published associations (MacArthur et al. 2017).

1.2.3 Next-generation sequencing

In the second half of the 2000s, several companies commercialized new technologies for sequencing DNA on massive scales. Commonly called next-generation sequencing (NGS), these methods allowed sequencing entire human genomes affordably for the first time (Metzker 2010). NGS technologies assay the sequence of segments of the genome, representing each as a "read". The first of these technologies, 454, debuted in 2005 and in a single experiment could produce over 300,000 reads with an average length of 110 nucleotides (Margulies et al. 2005). Today, there are NGS platforms which can produce billions of reads (Goodwin, McPherson, and McCombie 2016), and others which can generate reads over 1 million nucleotides in length (Jain et al. 2018; Payne et al. 2019).

Affordable NGS technology enabled the common use of resequencing as a method of measuring variants (Ng et al. 2010; Liti et al. 2009; Altshuler et al. 2010; Y. Li et al. 2010). Genotyping by NGS resequencing starts by determining the location in the genome each read comes from. This is usually done by short read aligners such as BWA, Bowtie2, or SOAP2 (H. Li 2013; Langmead and Salzberg 2012; R. Li et al. 2009). These tools use the Burrows-Wheeler transform or a derivative to create an index of all substrings in the genome. This pre-processing speeds the alignment of each of millions of reads. Once the reads are aligned to a reference sequence, one can determine which bases in the reads differ from those in the reference. The main advantage of resequencing over other genotyping methods is the ability of discovering previously unknown variants, instead of ones pre-selected for testing. Another benefit is that it makes it possible to detect variants more complex than simple SNPs. However, there are still limitations in determining length polymorphisms (**indels**) using alignment (Heng Li 2014b). And resequencing a human genome remains more expensive than a SNP survey using allele-specific oligonucleotide microarrays. Also, sequencing error in NGS platforms remains high enough to

cause frequent false positive variant detections (Margulies et al. 2005; Heng Li 2014b; Michael W. Schmitt et al. 2015).

1.2.4 Reference alignment

The standard approach to variant detection through NGS is to resequence and align to a reference genome. By the time NGS became common, reference genomes were available for most model organisms. As of April 2020, the Genome database of the National Center for Biotechnology Information (NCBI) contains complete genomes for 20,359 organisms. However, there are still many organisms with no high-quality reference sequence. A metagenomic study on cow rumen in 2011 assembled 179,092 microbial scaffolds, and found that 99.97% of them had no strong similarity to known genomes (Hess et al. 2011). Even 7 years later, a similar study found that only 0.7% of assembled rumen genomes were from identified species (Stewart et al. 2018). As for more complex organisms, there are only 1,525 complete eukaryotic genomes in the NCBI's database, out of 1.2 million catalogued species (Mora et al. 2011). Even if a reference genome was available for every species, there are some cases, like shotgun metagenomics, where the identity of the species being sequenced is not known beforehand.

Another situation where a reference can be a hindrance is when sequencing a region with large or complex structural variants. For example, the major histocompatibility complex (MHC) proteins in human are encoded in a region with a highly polymorphic haplotype (Horton et al. 2008). Read pairs spanning the junctions between structural variants may align incorrectly or not at all, depending on which reference sequence is used.

Even when there are no structural variants preventing the alignment from succeeding, indels can cause a bias in observed allele frequency. If a sample is polymorphic for an indel, reads containing the allele present in the reference may align more easily to it than reads with the alternate allele (Degner et al. 2009; Stevenson, Coolon, and Wittkopp 2013; Arbeithuber, Makova, and Tiemann-Boege 2016). When the first step in the analysis is aligning to the reference, these reads which fail to align will be lost and omitted in the final analysis. The measured allele frequency of the reference allele will then be erroneously high. An opposite effect can occur when there are two similar regions in a genome. If the reference contains only one of the regions, reads from the other can mistakenly align to it, causing artifactual variant

calls. In one test, choosing the wrong reference caused 168,800 mistaken SNP calls from this effect (Heng Li 2014b).

1.2.5 Discrete vs. continuous variant calling

The advantage of diploid sequencing is the discrete range of possible variants. If we are searching for germline variants in an organism with a known ploidy, we can leverage the expectation that variant frequencies will exist as a finite number of discrete values. Idealized germline variants in an organism with a known ploidy exist only at n+1 frequencies, where n is the ploidy. For example, there are three theoretical allele frequencies in a diploid organism: 0, 0.5, and 1. In a hexaploid organism, there are seven.

To determine whether a diploid genotype is hetero- or homozygous, the accuracy of the variant frequency estimate can be relatively low. For instance, if 8 out of 20 reads show one genotype and 12 show another, the tool FreeBayes calls the site a heterozygote with a QUAL score of 167, meaning only a 1 in 5×10^{16} chance that the polymorphism is not real, barring systemic bias (Garrison and Marth 2012). This is despite the fact that the raw variant frequency, 40%, is 10 percentage points from the actual frequency implied by the heterozygous call (50%).

But when the goal is estimating variant frequencies in a population of DNA, the frequencies can vary continuously between 0 and 1. No longer are we determining which frequency it is from a set of discrete values. Instead, there are infinite possible values. In the above example of observing a variant in 8 of 20 reads, we might estimate that the actual frequency of the variant is 40%. But with only 20 reads, we can only say with 95% confidence that it is between 18.5% and 61.5% (normal approximation). Contrast this uncertainty with the assurance of the FreeBayes call of the same raw data. When calling continuous variants, a higher precision of frequency estimation is often necessary. A mitochondrial variant may have very different health consequences at 60% frequency than at 20%. Attaining higher precision requires a greater depth of sequencing, as the confidence of frequency estimation increases with sample size.

So far, we have only discussed uncertainty from sampling error. When sequencing error is taken into account, this adds another challenge in obtaining an accurate estimate of the variant frequency. Essentially, the effect of sequencing error is that we cannot even be sure

that an observation of 8 reads of a variant means that there were actually 8 reads with this variant in our sample of reads from that locus. Some of those observations may instead be artifacts. So the actual uncertainty in variant frequency is greater.

Importantly, this also means that an observed variant may not actually be present at all. All observations could be artifactual. This becomes increasingly likely as the number of observations approaches zero. When searching for low frequency variants, this results in a great challenge. Sequencing error effectively creates a baseline of noise at the bottom of the frequency range. Below a certain frequency, the vast majority of apparent variants are artifacts. Figure 1.1 demonstrates the effect of this noise on finding true variants. Panel A shows all the apparent variants in the standard sequencing, and panel B shows the only variant found using a high-accuracy method. Almost all the other variants are likely artifacts, many with a minor allele frequency (MAF) higher than the true variant.

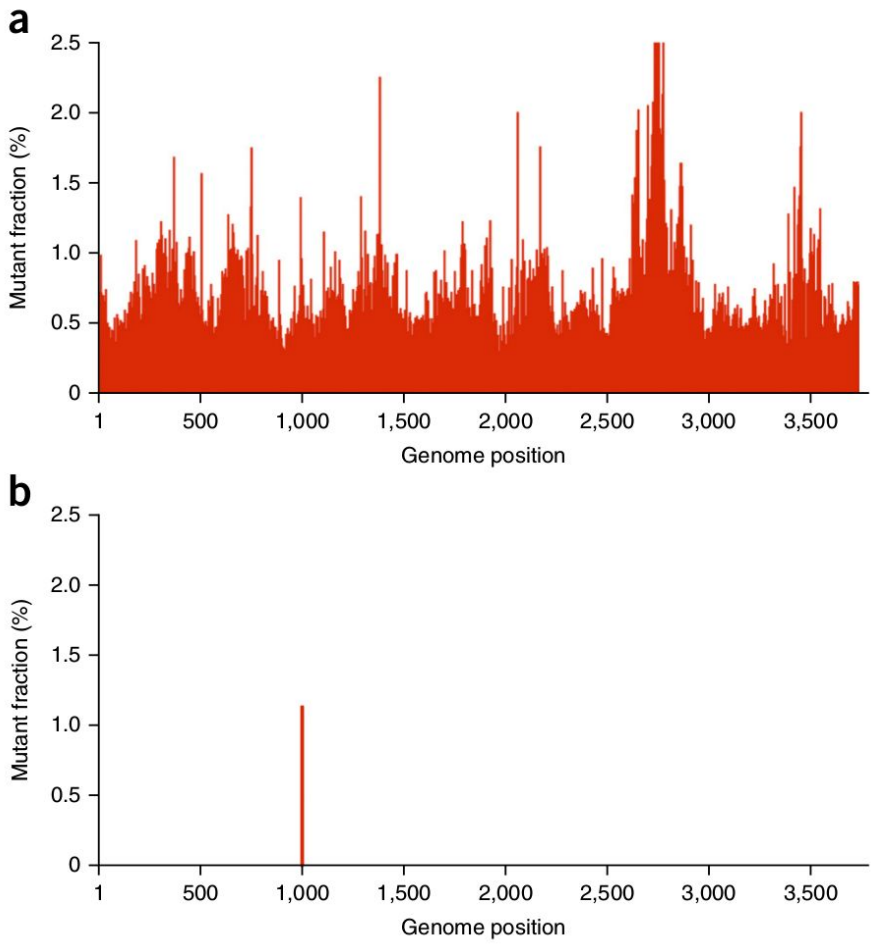


Figure 1.1 The needle in the haystack of sequencing error. In (a), target captured DNA from exons of *ABL1* was sequenced with an Illumina HiSeq 2500. Shown are all variants appearing in the data, after stringent quality filtering. In (b), a high-accuracy method, duplex sequencing, was applied to the same sample. Shown is the only variant detected in the same region. Source: (Michael W. Schmitt et al. 2015).

When calling variants with discrete frequencies, this noise is a much smaller problem. When the only hypotheses are 0%, 50%, and 100%, a detected frequency of 2% can be much more confidently judged to be a 0% perturbed by sequencing error. But with continuous variants, this 2% could actually be 2%. So when calling low-frequency, continuous variants, the rate of sequencing error is another crucial consideration, in addition to sequencing depth.

1.3 Measuring sequencing error

The accuracy of sequencing methods has been studied since the advent of sequencing. A standard approach has been to sequence a known sample, align it to a reference sequence, and call differences from the reference errors (Berno 1996; Ross et al. 2013). But with this method, if the sample itself contains actual, biological differences from the reference, these will be mistaken for sequencing errors. A way to eliminate some of these artifacts is to assume any variant occurring in the majority of the reads is the "true" biological allele. Essentially, one would perform some sort of variant calling after alignment, and only classify an allele as a sequencing error if it's different from the called variants.

This approach still requires a highly homogenous sample. Since any minor allele is assumed to be an error, diversity in the sample will result in artifactual errors. Diversity is a feature of many biological systems like microorganism cultures and organelles (Mei et al. 2019). So in these experiments, the expected frequency of any minor alleles must be much lower than the expected sequencing error rate. For most next-generation sequencing, this level of homogeneity can be achieved by using samples like clonal DNA. But this means the sample must be carefully chosen in an experiment to study sequencing error.

Another requirement of this reference alignment based approach is a high-quality reference sequence. This requirement may be self-evident, but it stands to be made explicit, since reference sequences are not available for or applicable to every system.

Another caveat with this method is that it measures more than just the error rate of the given sequencing platform. It also measures errors introduced during DNA extraction, library preparation, PCR, and every step between sample preparation and the sequencing machine. These sources of errors vary depending on the sample and library preparation method (Melanie Schirmer et al. 2016). One example workflow and the errors it introduces is shown in Figure 4.1. Oxidative damage can introduce errors during steps like fragmentation or target capture (Newman et al. 2016; Costello et al. 2013). And even high-fidelity PCR polymerases have error rates around 2×10^{-5} per base per replication, and there can be many replication phases in a PCR (Kebschull and Zador 2015). Including these types of errors makes the results less portable and useful, since other experiments may use a different series of steps.

Even with these considerations, this standard method has been used repeatedly to obtain useful figures for sequencing error rate, and to examine error biases. One finding has been that results can vary widely from experiment to experiment, even ones performed in the same way by the same group (Melanie Schirmer et al. 2016). As a consequence, these studies can at best be considered a look at sequencing accuracy in one particular set of well-controlled conditions. This may have different utility for different audiences. For experimentalists designing a study and choosing a sequencing platform, it is very useful to have a comparison where different platforms were compared under the same conditions. But for those planning a computational study making use of existing data, one is interested in the general quality of existing, real-world datasets. The level and variation of quality in published datasets may not be well reflected in controlled studies focused on accuracy.

1.4 Consensus-based sequencing

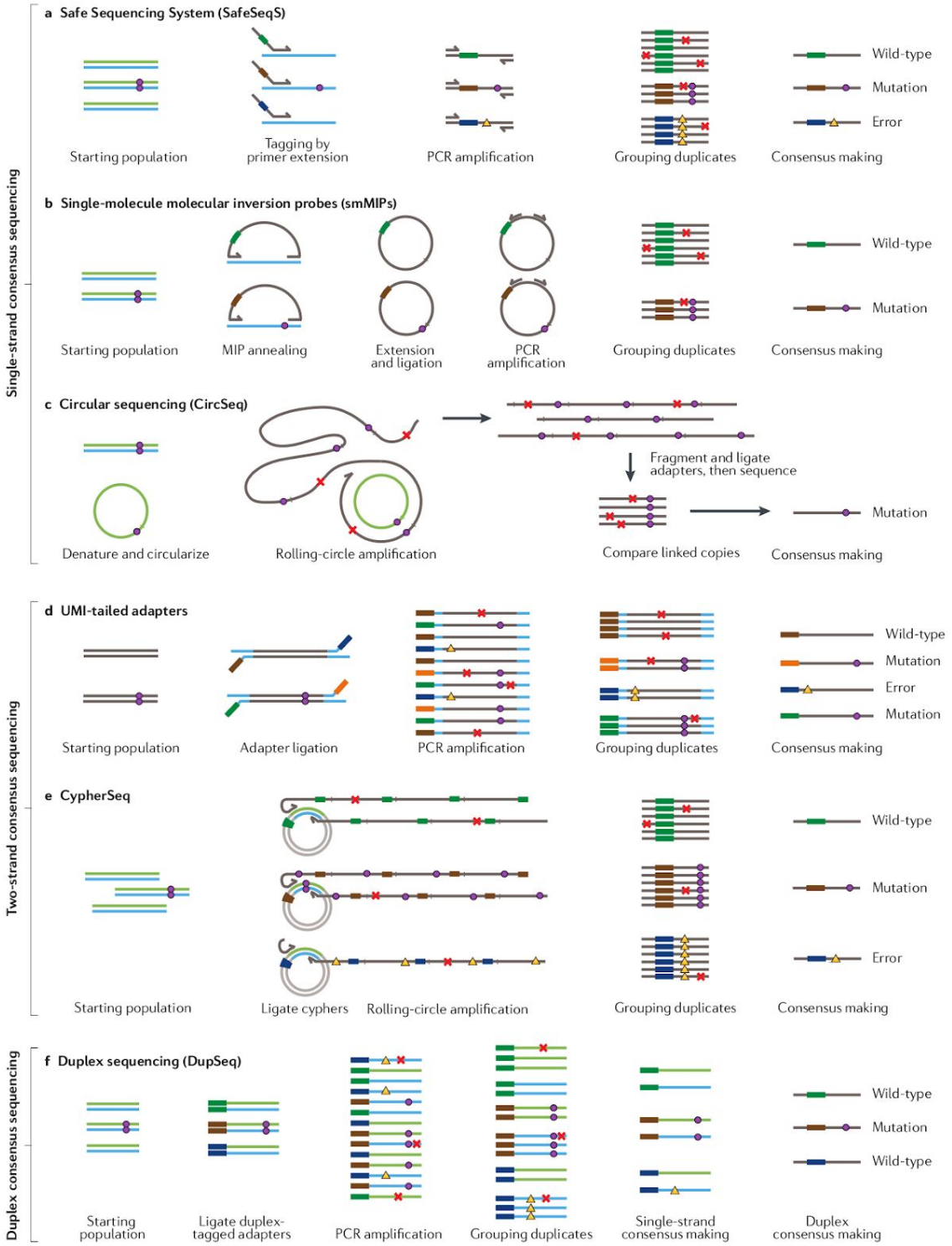


Figure 1.2 Consensus-based sequencing methods. Source: (Salk, Schmitt, and Loeb 2018)

Variant calling with standard NGS reads is now routine for discrete variants. But to reliably detect low frequency variants requires confronting the noise background in sequencing data. Instead of solving the problem with more sophisticated *in silico* analysis, one can alter the data generation method.

In 2010, Hiatt *et al.* described a method of tagging individual molecules in a sequencing library. Amplifying these fragments with PCR before sequencing them allows one to generate multiple reads from the same, identifiable molecule (Hiatt et al. 2010). Collapsing this **family** of reads into a consensus yields a sequence with much higher accuracy than any of the reads it was constructed from. The concept is that each read might contain PCR and/or sequencing errors, but if each contains different errors, they can be identified and eliminated. Even if some errors appear in multiple reads, as long as none is in the majority, the consensus will be the same sequence as in the original molecule. In the years since, the idea has been expanded into a variety of methods for consensus-based sequencing, many of which are outlined in Figure 1.2.

1.4.1 Issues in consensus sequencing

A crucial distinction between the methods is how the tagging of molecules is performed. One option is to simply use the end points of the mapped read pair (**endogenous** tags). If the shear points of the fragments are random, and the depth of coverage is low, then it is unlikely that two different fragments will share the same endpoints (Hoang et al. 2016). The method of Hiatt *et al*. essentially depends on this principle. Another, more common type of tag is a random **barcode** sequence ligated to the end(s) of the fragments (**exogenous** tags). With enough random bases, it can be vanishingly unlikely that any two tags sequenced will be identical. An important early implementation of this idea was the Safe-SeqS protocol (I. Kinde et al. 2011). Also, several methods use both sources of tagging information to increase the confidence that each fragment is uniquely identified.

In designing a barcode-based protocol, an important question is the number of random bases to use for the barcode. If two identical barcodes end up labeling different fragments, most downstream analyses will group the reads from these two fragments into a single family.

This is called a barcode **collision**. A collision can result in a low-quality consensus sequence and errors.

As mentioned, the chance of a collision can be made extremely unlikely. But the barcode length is critical. For example, there are 4^{24} (281 trillion) different combinations of 24 random bases. If 1 million fragments are sequenced, then the probability of a single collision is only 0.177%. But reducing the barcode length by just 5 bases, to 19bp, raises this to an 83.8% chance.[1] But though this means we are likely to encounter a collision in this experiment, there probably will not be more than one. If the application is resilient to collisions, and few fragments are sequenced, then a smaller barcode length may be acceptable. For instance, the Primer ID method uses barcodes with only 8 bases. But the usual application is sequencing HIV populations, and some amount of collision is acceptable. Also, the number of fragments sequenced is often in the thousands. If only 95% of the fragments need to be uniquely labeled, then an 8bp barcode is enough to tag about 2,000 fragments (Liang et al. 2014). But for other applications and larger samples, much longer barcodes are needed. The power of consensus sequencing is that it can give weight to every single observation of a variant. If one is searching for extremely rare mutations, the chance that any one observation is a collision must be very low.

Another basic issue is: how large does a family of reads have to be in order to call a consensus sequence? Starting from the bottom, a one-read family would be no better than a raw read. And in a two-read family, if the reads disagree on a base, there is only enough information to say that an error occurred, but not what the correct base is. So three reads is the minimum needed to be able to break a tie in case of a disagreement, and make a call as to what the most likely base is. From here, as the size of the family increases, the main advantage is higher confidence in the called consensus base. Eventually we will exceed the confidence needed for our application, and we need to consider the downsides of aiming for very large families. The main downside is that reads cost time and money. Consensus sequencing already requires a much greater volume of sequencing than standard sequencing does to assay the same sample. And reads used to increase our accuracy beyond our requirements could have

[1] Using the first equation in the Results of (Liang et al. 2014), we find that $1 - \exp(-(1000000^2)/(2 \cdot 4^{24})) \approx 0.0017747$ and $1 - \exp(-(1000000^2)/(2 \cdot 4^{19})) \approx 0.83781$.

been used to sequence more sample. This is why most consensus methods set a threshold at three (M. W. Schmitt et al. 2012). Even with the optimal threshold set, natural sampling variation means that most families will not be that size (Kennedy et al. 2014). So a great quantity of reads is normally wasted in families larger or smaller than the threshold.

Another important distinction between consensus sequencing methods is whether reads from the two original strands of the fragment are distinguished. For instance, with Safe-SeqS (and endogenous tags), each fragment is identified by a unique combination of start and end points. But both strands of the fragment have the same pair of end points. So there is no way to determine which strand a final read descends from. This means that errors from the first cycle of PCR can appear as true variants, even when many reads are obtained from the fragment. This can occur because even though a first-cycle PCR error will only affect one strand, if more reads are observed from that strand than the other one, the error will appear in the majority. Newman *et al.* found that 80-90% of families consist exclusively of reads descended from one strand, meaning that these first-cycle errors will usually avoid detection (Newman et al. 2016).

A way to avoid these artifacts is to use the redundancy inherent in the DNA double helix. Because DNA consists of two, complementary strands, each sequence is encoded twice. Independently assaying both strands gives the ability to check for disagreements resulting from early errors. A disagreement cannot be resolved with only two data points, but it can be identified and the site marked as ambiguous. This is much preferable to unknowingly proceeding with an incorrect base call which is given the confidence of consensus sequencing. Barcode tagging schemes exist which make this possible by marking the strand provenance of each read (M. W. Schmitt et al. 2012). Making use of this information allowed one group to increase sensitivity to variants from 86% to 96% (Newman et al. 2016).

1.4.2 Consensus sequencing methods

1.4.2.1 Single-strand consensus sequencing

One of the first consensus sequencing methods to be adopted was the Safe Sequencing System (Safe-SeqS). This method does not distinguish strands, but can use both endogenous

and exogenous tags. The method of incorporating the exogenous tags is to include them in PCR primers which are applied in the first two PCR cycles. This results in all the descendents of one strand being labeled with one barcode, and the descendents of the other strand having another, unrelated barcode. Even though each strand receives a unique barcode, there is no way to pair up barcodes which came from the same original fragment. This means that any PCR error which occurs in the first cycle will propagate to all reads bearing that barcode. But the process still ensures that all reads bearing the same barcode share a single ancestor. This allows building a consensus sequence that eliminates most PCR/sequencing errors. Safe-SeqS was reported to reduce the error rate by 70x. (I. Kinde et al. 2011) Later, it was used to identify variants at a frequency as low as 0.01% (Isaac Kinde et al. 2013). A method using a similar idea is Primer ID. This technique also incorporates barcodes into primers, but it targets RNA, so the barcode is in the reverse transcriptase primer (Jabara et al. 2011). A very different and innovative method, circle sequencing (CircSeq), emerged in 2013. In CircSeq, genomic fragments are circularized, and rolling circle amplification is used to create many copies in tandem on a single molecule. With a short enough fragment and long enough reads, several copies of the original sequence can be obtained in a single read pair. This avoids the use of barcodes or read endpoints to group families. This system is very efficient, with a very constant family size. (Lou et al. 2013) But the length of reads from current high-volume sequencers limits this method to a short the fragment size. And, as with the previous methods, it cannot distinguish strands and so is vulnerable to early PCR errors.

1.4.2.3 Duplex consensus sequencing

In 2012, Schmitt *et al*. demonstrated a technique for utilizing the redundancy present in double-stranded DNA to check for errors early in PCR. Illustrated in Figure 1.3, **duplex sequencing** tags fragments with a set of barcodes that allow discerning which strand each read descends from. The key is including the barcodes in the double-stranded portion of the Illumina Y adapters. Each fragment is ligated to two adapters, one on each end. Because the barcodes are double-stranded, each strand of the fragment will be tagged with both barcodes. Then, one strand will have the mate 1 sequencing primer on the end with one of the barcodes (referred to here as α), and the mate 2 primer with the other barcode (β). But because the primers are in

the non-complementary portion of the Y adapters, the other strand will have the mate 1 primer next to barcode β and the mate 2 primer next to barcode α. The end result, after amplification and sequencing, is that all reads from one strand will have barcode α at the start of mate 1 and barcode β in mate 2, and the reads from the other strand will have β in mate 1 and α in mate 2. So the raw reads can be grouped into two strands and an independent consensus sequence can be formed for each of the original strands. Even if there was an error in the first round of PCR, it will only have affected one of the strands. So almost all such errors should be detectable as discrepancies between the two single-strand consensus sequences (SSCS). (M. W. Schmitt et al. 2012)

With two 12bp random barcodes labeling each fragment, these exogenous barcodes alone are enough to uniquely identify every fragment with a very low probability of collision. As mentioned earlier, theoretically 24 random bases is enough to give a 99.998% probability of 0 collisions, even when labeling 1 million fragments.

The use of both strands gives this method a significant advantage in specificity. A trial found 12.5x more low frequency variants in SSCS than duplex consensus sequences (DCS), likely artifactual (and 5x more variants in standard sequencing than SSCS) (Ahn and Lee 2019). But the tradeoff is that discarding families with reads from only one strand significantly reduces yield. With 80-90% of families being single-stranded, this requirement can greatly increase the amount of sequencing required (Newman et al. 2016).

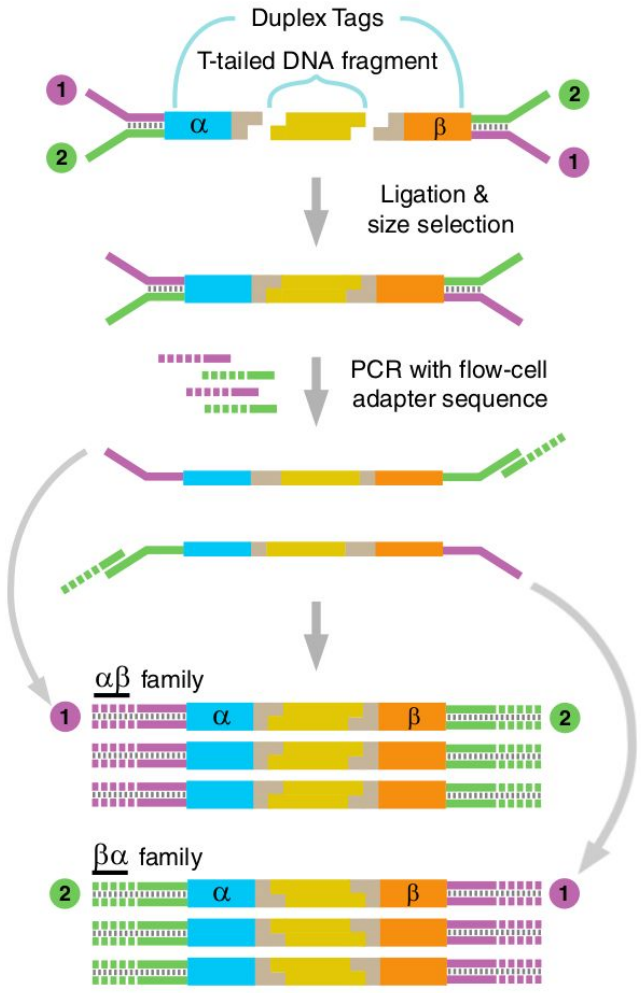


Figure 1.3 The mechanics of barcode labeling in duplex sequencing. 12bp degenerate barcodes are included in the double-stranded portions of Illumina Y adapters. These are ligated to the sample fragments, which are then amplified by PCR. This yields two populations of molecules from each original fragment. These two types of molecules can be distinguished by the order of barcodes in the resulting paired-end reads.
Adapted from: (M. W. Schmitt et al. 2012)

1.4.3 Processing duplex sequencing data

The process of resolving raw duplex reads into consensus sequences consists of three main steps: grouping reads by tag, aligning families of reads, and calling consensus sequences from the aligned families.

The naive approach to grouping reads by their barcode would be to traverse the set of raw reads and for each unique barcode, keep a list of read pairs tagged with it. This can be accomplished in most programming languages with a mapping data structure. But when an experiment generates gigabases of raw reads, keeping all the reads in memory can consume a prohibitive amount of RAM. Instead, different approaches can avoid this memory consumption.

The method used by all current tools which process duplex data is to align the reads to a reference sequence (Kennedy et al. 2014; Fennell and Homer 2018; Newman et al. 2016). This can be performed efficiently by existing short read mappers (H. Li and Durbin 2009; Langmead et al. 2009). Then, the alignment can be traversed by region, finding reads aligned to the same location. This has the effect of breaking the problem into many thousands or millions of subproblems. Each group of reads mapped to the same area will be much smaller than the set of all reads. Essentially, this technique uses endogenous tags (fragment endpoints) to aid the use of exogenous tags (barcode sequences).

Of course, it is possible to efficiently group reads by barcode without the use of a reference. A simple approach is to sort the reads by barcode, using the existing Unix `sort` command. GNU `sort` is already able to limit memory usage by using the filesystem to store temporary data (MacKenzie et al. 2018). Then, the sorted output will contain reads grouped by barcode.

The second major step, aligning families of reads, can also be assisted by reference mapping. Once the reads are aligned to a reference, they are also effectively aligned with each other. This essentially uses the reference sequence as a guide for a multiple sequence alignment of each family of reads. But alignment can also be done without a reference sequence, using one of the many published multiple sequence alignment algorithms.

The final step, calling a consensus sequence, is computationally straightforward once the reads of a family are aligned. For each position in the alignment, the bases in each read are

tabulated, and the most common one is the consensus base. Here, a threshold can be applied so the consensus base must be present in at least a certain fraction of the reads in order to use it in the final duplex read. Kennedy *et al*. report that 70% is the best compromise between accuracy and yield (Kennedy et al. 2014). The consensus sequences produced by this step can be used the same as normal sequencing reads in most downstream tools. But unlike normal reads, the accuracy of each read can be considered to be very high (Salk, Schmitt, and Loeb 2018).

At several of the steps above, a reference sequence can be used to assist processing. But this can be a limitation in many of the circumstances discussed earlier. A workflow that avoids the requirement of a reference sequence would be useful in these situations, like when a good reference is not available, or when it would introduce biases.

1.4.4 Barcode errors

1.4.4.1 Loss of data

The core part of processing duplex sequencing data is grouping reads by barcode. The simplest way to do this is to group reads with identical barcodes. But in real data, this results in a loss of reads. When sequencing or PCR errors occur in a barcode, this prevents the read from being matched with its family. The most likely scenario is that this particular error will only occur in one read, making the barcode unique. Thus, the read will appear to be the lone member of its family (a singleton). It will not be able to form a consensus sequence with other members of its family, so it will be lost.

The amount of data lost depends on the rate of errors in the barcodes. Assuming a per-base PCR error rate of e and an average of r replication events for each molecule, the chance of a PCR error at each base is $1-(1-e)^r$.[2] For *Taq* polymerase, e has been measured at 0.00013 (Potapov and Ong 2017). Assuming a PCR with an average of 20 replication events, the per-base error rate is then $1-(1-0.00013)^{20} \approx 0.0026$. If sequencing incurs a 0.0034 per-base error (Ross et al. 2013), the combined per-base error rate becomes $1-(1-0.0026)\cdot(1-0.0034) \approx$

[2] The chance of no error at a particular base is 1-e. The chance of no errors in r replications is $(1-e)^r$. So the chance of at least one error is $1-(1-e)^r$.

0.0060. Then, the chance of at least one error in a 24 base barcode is $1-(1-0.0060)^{24} \approx 0.13$. So in this scenario, 13% of read pairs will be affected by a barcode error and become their own singleton family. Imagining a simple example where all families contain 10 reads, there would be 1 true family for every 10 raw reads. With this rate of barcode errors, every 10 raw reads includes 1.3 reads with an error in the barcode. Assuming this produces a singleton family, the ratio of real to artifactual families is 1:1.3, meaning 57% of observed families would be artifacts. These singleton families would be discarded, wasting the sequencing data contained within.

1.4.4.2 Barcode error correction

The obvious response to this data loss is, then, to allow non-exact matches between barcodes. But this encounters a problem that exact matching is able to avoid. The naive approach for both exact and inexact matching is to compare pairs of barcodes to check their similarity. Each barcode would be compared against every other barcode to find matches. This requires on the order of N^2 comparisons,[3] where N is the number of barcodes. Such an algorithm is said to have $O(N^2)$ time complexity. As the size of the input dataset grows, the number of comparisons quickly becomes prohibitive.

For exact matching, though, there are straightforward shortcuts. One could use a hash table to map barcodes to their groups. Hash table lookups can be performed in constant (O(1)) time, reducing the overall complexity to O(N) (N comparisons). Or, one could simply sort the reads by their barcodes. This would yield a list of reads, with groups occurring in runs of reads. There are several sorting algorithms available with average complexity of $O(N \cdot \log(N))$ (Knuth 1973).

But for inexact matches, these shortcuts do not work. Basic hash tables require exact key matches. Sorting may put similar barcodes next to each other, but if the error occurs near the start of the sequence, they may end up very far apart. A more sophisticated approach is required.

There are existing tools which solve the inexact matching problem in different ways. The most common way is to use a reference sequence (T. Smith, Heger, and Sudbery 2017; Xu et al. 2018; Fennell and Homer 2018). Mapping to a reference is a way around the problem by

[3] $(N(N-1))/2$ comparisons, to be exact

reducing its size. Mapped reads can first be grouped by location. Then all the barcodes of reads close to each other can be matched against themselves. This solution does not improve the basic $O(N^2)$ time complexity of inexact matching. Instead, it works by breaking the problem into chunks, each of which has a much smaller N.

Another method of reducing the complexity of inexact matching is through pre-computation. Building an index of barcodes costs computational time up front, but can vastly reduce the time needed to group similar barcodes. This is the same concept used by modern short-read mappers to align reads to a reference genome. Tools like BWA and Bowtie create an FM-index of the reference genome, which speeds the process of finding a genomic region similar to the sequence of each read (H. Li and Durbin 2009; Langmead et al. 2009).

1.4.5 Multiple sequence alignment

The second major step in processing barcoded raw reads is to align the reads of each family to themselves. As mentioned above, one method is to align all raw reads to a reference sequence. This will yield reads already aligned with each other, using the reference as a guide. In a reference-free method, however, the reads must be aligned to themselves with no guide. This is a classical multiple sequence alignment (MSA) problem.

An efficient approach to MSA is progressive alignment. In progressive alignment, each sequence is aligned to a growing multiple sequence alignment, one at a time. The choice of which sequences to align first can greatly influence the final alignment, so it is important to start with the most similar sequences (Feng and Doolittle 1987). This means calculating the similarity between every pair of sequences, which has an $O(N^2)$ time complexity. This means that the computation time grows roughly with the square of the number of sequences. Since the rest of the process has a lower time complexity, the efficiency of this step dominates the performance of the overall algorithm.

There have been several approaches to efficiently calculating the matrix of pairwise similarities. A common strategy is to perform some kind of pairwise alignment, then count the number of k-mers of a small, fixed length that are shared between the two sequences. ClustalW's "k-tuple" method uses Wilbur and Lipman's 1984 algorithm to find 4 nucleotide blocks of exact, gapless matches between the sequences (Higgins and Sharp 1988; Wilbur and

Lipman 1984). In MAFFT's protein aligner, the metric of Jones *et al*. is used, but *k* is set to 6 amino acids, after they are collapsed into six physio-chemical groups (Kazutaka Katoh et al. 2002; Jones, Taylor, and Thornton 1992). This strategy of counting identical blocks is fast, but not as accurate as scoring a full alignment. Its accuracy also suffers when errors interrupt blocks of identity. In light of this limitation, Kalign uses a string matching algorithm that allows mismatches. The Wu-Manber algorithm was used in Kalign, and Muth-Manber was used in Kalign2. These algorithms allow for extremely fast string matching while still allowing mismatches, improving accuracy. (Wu and Manber 1992; Muth and Manber 1996; Lassmann and Sonnhammer 2005; Lassmann, Frings, and Sonnhammer 2009)

1.5 Performance optimization

1.5.1 Computational bottlenecks

Discussing multiple sequence alignment may give the impression that algorithmic performance is all that matters in the speed of a tool. But in practice, the most important factors are often certain bottlenecks in modern CPU architecture. Reducing the time complexity of an algorithm only directly affects how many instructions a CPU must perform. But many hardware operations take orders of magnitude more time than a simple CPU instruction. For example, fetching data from main memory (RAM) can take tens of thousands of times longer than a CPU instruction operating on register data. And fetching data from a hard disk can take millions of times longer. (Freitas et al. 2011) Avoiding these costly operations can be worth more than algorithmic improvement.

Some software architectures can necessitate more of these lengthy hardware operations. Often, a tool will launch separate child processes to perform some of its work. This can require more of these costly operations.

1.5.1.1 Temporary file manipulation

One issue with the use of child processes is that they may necessitate the use of temporary files. The child process may be a tool written by others, and so cannot be easily modified. And some tools read input from a file or write output to one, with no streaming

option. So the parent tool may have to provide its input in a file and/or receive its output from a file. This is an acceptable solution if the tool is not run too many times, or if its runtime is in minutes. But the process of creating, writing to, reading from, and deleting a file takes time. Accessing a hard disk is one of the slowest local operations that can be performed, with a seek operation taking multiple milliseconds (Freitas et al. 2011). Even solid state drives have access times much longer than main memory does. So any drive access can incur a large performance penalty. Additionally, if temporary files are being rapidly created and deleted, this "churn" can impact all other processes on the system.

1.5.1.2 Process creation

Another type of overhead incurred by the use of child processes is the creation and destruction of the process itself. In a Unix-like operating system, child processes are created with the `fork()` system call followed by the `exec()` system call. `fork()` creates a copy of the parent process, which will become the child process. The data of the parent process does not need to be physically copied, but other associated objects like file descriptors and virtual memory map entries do need to be copied. These operations will involve accessing main memory many times. `fork()` can take multiple milliseconds, depending on how much needs to be copied. (Ruan and Pai 2004) The `exec()` system call launches the executable for the new process. This means setting up the data structures for the new process, allocating memory, and performing all the other initialization functions for it. The child process also must usually read and parse its input from a file format into its internal data structures. Only then can it begin performing its main algorithmic purpose.

1.5.1.3 Impact

Normally, these types of overhead have a minimal impact on a tool's running time. But when the normal running time of a child process is very small, this overhead can become a large proportion of its total execution time. And if the child process is executed thousands or millions of times, the overhead can add up.

1.5.2 Parallelism

Since the late 2000s, difficulties in increasing processor speed have led to increasing numbers of cores in CPUs (Jagtap 2009). Consequently, the parallelizability of an algorithm can be more important than its single-threaded runtime. Fortunately, several steps in the processing of duplex sequencing data deal with only the reads of a single family. This means these steps can be divided into millions of independent tasks. And the most computationally expensive step in a reference-free pipeline, multiple sequence alignment, is one of these parallelizable steps.

So on the surface, the alignment step is embarrassingly parallel. However, there are other desirable features of the alignment tool that make implementing this parallelism non-trivial. The first of these features is the ability to stream output. Unix makes it simple to operate several steps of a pipeline in parallel, streaming data from earlier steps to later ones. In this case, the alignment step occurs before the consensus calling step, so it is possible to stream alignments from the aligning tool to the consensus calling tool, letting them both work simultaneously. If alignment takes more time than consensus calling and it is run in parallel, and there are available processors for consensus calling, that completely removes the consensus calling time from the overall wall clock time the pipeline takes.

Another desirable feature for the alignment tool is to preserve the input order in the output. That is, aligned families should appear in the output in the same order as the families appeared in the input. This property makes it easier to reproduce problems and perform functional testing. It also makes it easier to inspect intermediate files and trace a family backward from the pipeline output through the individual steps.

Building a tool that satisfies all three properties – efficient parallelization, streaming, and order preservation – is not simple. The simplest approach to parallelization is to begin a family alignment job on each available core of the machine. Then, every time an alignment finishes, output the results and start a new alignment on that core. This scheme would make the most optimal use of the CPU resources, but it would produce alignments in the order they finished, not their input order. If we abandoned the streaming constraint, we could wait until all the alignments finished, then sort them back into the input order. Instead, maintaining all three

features requires some sort of scheme where alignments are “chunked” into groups of jobs which are held in memory and output once they all finish. Then, the order can be ensured before producing any output, but the tool does not have to wait until all jobs are finished. It does have to wait until all jobs in the chunk are finished, though, leading to time where all jobs are finished except the slowest one, and the rest of the cores are idle. Thus the size of the chunk is a tradeoff between this inefficient core use and the memory use of caching a large chunk in main memory.

1.6 Summary

So far, we have seen the importance of variation and the development of methods to detect it. I have discussed the distinction between variant detection in fixed-ploidy systems versus systems where variants can be present at any frequency. And for the latter case, I have explained how it is much more sensitive to sequencing errors. Because errors are so critical, it is important to be able to measure the quality of the sequencing data we rely on. Many studies have quantified the error rate of standard next-generation sequencers. But the standard approach for measuring error rates has limitations: experiments must be tailor-made for this purpose, and the “sequencing errors” detected are not all from the sequencing instrument. So I have developed an alternative error detection method, discussed in Chapter 4. It can be automatically applied to any paired-end sequencing run, and I used this advantage to perform a large-scale survey of existing sequencing datasets. This provides valuable information on the quality of public sequencing data.

Even better than detecting sequencing errors is correcting them. Above, I have given an overview of consensus sequencing, an experimental and computational method which vastly improves on the accuracy of raw sequencing data. Duplex sequencing is perhaps the most powerful of these techniques. Yet the existing tools for resolving raw duplex reads into finished duplex consensus sequences are lacking. They rely on alignment to a reference sequence, which is not always available or desirable. In Chapter 2, I describe Du Novo, a tool I wrote to form consensuses from raw duplex data without the use of a reference.

Another issue with many existing tools is that a single error in the duplex barcode will result in that read pair being wasted. Recovering these reads is a challenge, especially if one attempts to do so without the aid of a reference sequence. And forming consensus sequences without using a reference requires some sort of multiple sequence alignment algorithm. Above, I discussed the algorithmic challenges of performing fast alignments. I also covered the practical issues inherent in modern computer architecture that can hinder performance. In Chapter 3 I show how I addressed these issues to vastly improve the performance of Du Novo. I also demonstrate a functionality to recover reads with errors in their barcodes, still without the use of a reference, resulting in much improved yield.

2 Streamlined analysis of duplex sequencing data with Du Novo

This chapter is reproduced from a research article published in *Genome Biology* in 2016 by Nicholas Stoler, Barbara Arbeithuber, Wilfried Guiblet, Kateryna D. Makova, and Anton Nekrutenko (DOI: 10.1186/s13059-016-1039-4).

2.1 Abstract

Duplex sequencing was originally developed to detect rare nucleotide polymorphisms normally obscured by the noise of high-throughput sequencing. Here we describe a new, streamlined, reference-free approach for the analysis of duplex sequencing data. We show the approach performs well on simulated data and precisely reproduces previously published results and apply it to a newly produced dataset, enabling us to type low-frequency variants in human mitochondrial DNA. Finally, we provide all necessary tools as stand-alone components as well as integrate them into the Galaxy platform. All analyses performed in this manuscript can be repeated exactly as described at http://usegalaxy.org/duplex.

2.2 Background

The term “genetic variation” is often used to imply allelic combinatorics within a diploid organism such as, for instance, humans or Drosophila. Yet the majority of organisms in the biosphere are not diploid (prokaryotes and viruses), and even those that are, include non-diploid genomes such as mitochondria and chloroplasts. Identification of genetic variants — e.g., single nucleotide polymorphisms (SNPs) and small indels — is especially challenging in non-diploid systems due to the lack of a simple “homozygote-or-heterozygote” expectation: a heterozygous site may have not just two, but multiple allelic variants, with frequencies ranging anywhere from 0 to 1 (Hodgkinson et al. 2014; Zhang et al. 2016). Because high-throughput sequencing technologies exhibit considerable amount of noise (M. Schirmer et al. 2015), it becomes increasingly difficult to reliably call variants with frequencies below 1% (M. Li and

Stoneking 2012; M. Li et al. 2015; Goto et al. 2011; Rebolledo Jaramillo et al. 2014; Quail et al. 2012; Ross et al. 2013). In these situations increased sequencing depth does not improve the predictive power but instead introduces additional noise. This complicates the identification of low-frequency variants that is becoming critically important in a variety of applications. For example, in humans there are numerous disease-causing mitochondrial variants where the disorder penetrance is proportional to the allele frequency (Wallace and Chalkia 2013). Because dramatic shifts in allele frequency can occur during mitochondrial bottleneck during oogenesis, a disease-causing variant present at a very low frequency in mother may increase in frequency in the child to exhibit a disease phenotype. The lack of cure for diseases caused by mitochondrial DNA mutations and the recent regulatory approval of tri-parental in vitro fertilization (IVF) by the UK House of Commons makes it critical to identify low-frequency variants in the human mitochondrial genome (Dimond 2015). Other examples illustrating the importance of discovering low-frequency genome alterations include tracking mutational dynamics in viral genomes, malignant lesions, and somatic tissues (Jabara et al. 2011; Michael W. Schmitt et al. 2015).

Today the vast majority of strategies for the identification of low-frequency sequence variants rely on next generation sequencing technologies (NGS). Noise reduction in these approaches ranges from a simple base-quality filtering to complex statistical strategies incorporating instrument and mapping errors (M. Li and Stoneking 2012; Rebolledo Jaramillo et al. 2014; Kim et al. 2011). However there is still considerable uncertainty about alternative alleles with frequencies below 1%. For example, Figure 2.1 illustrates the number of potential polymorphisms observed within the human mitochondrial genome as a function of the allele frequency cut-off. At 1% there is an average of 3 sites (Rebolledo Jaramillo et al. 2014), while at 0.75% the number surpasses 10, and, finally, around 0.1% almost all sites appear polymorphic. Clearly, the majority of these sites are false-positives, but how does one know for certain? Potentially, highly sensitive techniques with a high dynamic range such as droplet digital PCR (Hindson et al. 2013; Miotke et al. 2014) can be used to validate each site, but this would quickly become prohibitively expensive and laborious to be performed on hundreds or thousands of sites.

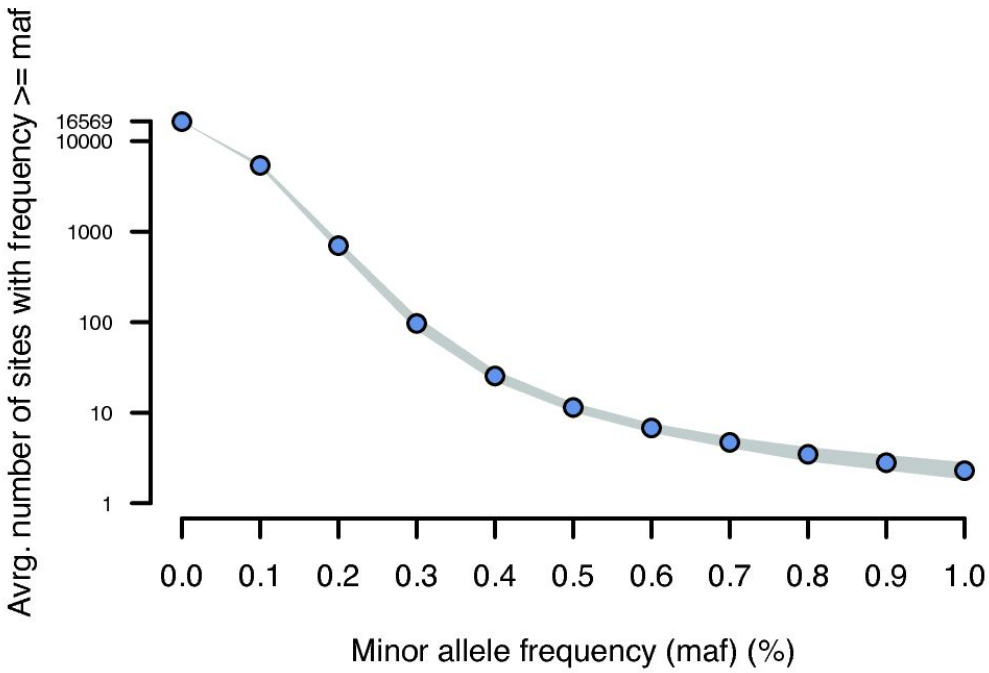


Figure 2.1 The relationship between the minor allele frequency (MAF) threshold (x-axis) and the total number of variable sites (y-axis) detected by (Rebolledo Jaramillo et al. 2014). Lowering the MAF threshold leads to an exponential increase in the number of variable positions. The image was generated by applying variable MAF thresholds to data from 156 human samples and plotting the average number of variable sites at a given MAF threshold. The line thickness corresponds to the 95 % confidence interval around the mean value.

An approach that offers a potential solution is duplex sequencing (M. W. Schmitt et al. 2012). This recently developed method was designed to increase sequencing accuracy by over four orders of magnitude. Duplex sequencing uses randomly generated barcodes to uniquely tag each molecule in a sample. The tagged fragments are then PCR amplified prior to the preparation of a sequencing library, creating fragment families characterized by unique combination of barcodes at both 5' and 3' ends (a conceptually similar Primer ID approach (Jabara et al. 2011) allows tagging of cDNA fragments at 5' end only). A family contains multiple reads, each originating from a single input DNA fragment. A legitimate sequence variant will thus be present in all reads within a family. In contrast, sequencing and amplification errors will

manifest themselves as “polymorphisms” within a family and so can be identified and removed. A consensus can be called from these read families. The consensus of all the reads originating from the same strand reduces errors originating from sequencing and PCR amplification. Then, comparing consensus sequences from complementary strands can identify early PCR errors.

Despite the fact that duplex sequencing promises great advances, the methods for both experimental and computational aspects of this technique are still evolving. In fact, the latter is lagging as it is based on alignment to a reference genome, which is disadvantageous for several reasons. The use of a reference genome biases results toward that reference, affecting studies using de novo assembly, or studies examining indels or other alleles that diverge far enough from the reference to cause alignment difficulties. The current analysis method also removes a large (and potentially useful) fraction of the original data due to stringent filters and uses suboptimal tools for variant identification. Here we describe an alternate analysis strategy, which removes reliance on a reference sequence, preserves a higher proportion of the input reads, and can be deployed as a stand alone application or as a part of the Galaxy system. We demonstrate the application of this approach by validating of rare variants in the human mitochondrial genome.

2.3 Methods

2.3.1 Duplex sequencing protocol used for human mitochondrial amplicons

Two overlapping mtDNA regions (each ~9 kb, representing the entire mitochondrial genome) were amplified from sample SC8C1-k1169-A*B (DNA extracted from buccal swabs of the child of family SC8 collected under IRB 30432EP; all experimental procedures described herein comply with the principles of the Helsinki Declaration), using the primer pairs L*2817+H*11570 and L10796+H3370 and mixed at equimolar quantities, as described previously (Rebolledo Jaramillo et al. 2014; Newman et al. 2016). Two µg of amplicons were sheared to ~550 bp and purified using 1.6 volumes of Agencourt AMPure XP beads (Beckman Coulter). Duplex sequencing libraries were prepared as described in (Kennedy et al. 2014) with several minor modifications. Briefly, T-tailed adapters were prepared by hybridization of MWS51 and MWS55, followed by extension, and a restriction digest with TaaI (HypCH4III) at 60

°C for 16 hours. Adapters were purified by precipitation with two volumes of absolute ethanol and 0.5 volumes of 5M NH4OAc. The hybridized PCR amplicon was end-repaired with the End-Repair Enzyme Mix provided in the Illumina TruSeq Kit according to manufacturer's protocol, A-tailed, and the adapter was ligated with 1,800 units of T4 ligase (NEB) with 20× molar excess at 16 °C for 30 min. Amplified tag families were generated from 15 attomoles of adapter-ligated amplicon by 23 cycles of PCR (the optimal cycle number was evaluated by real-time PCR). The library was quantified with the KAPA Library Quantification Kit (Kapa Biosystems) according to manufacturer's instructions. Sequencing was performed on an Illumina MiSeq platform using 301 bp paired-end reads.

2.3.2 Construction of read families

Read pairs are grouped into families according to the random tags which constitute the first 12bp of each read using Du Novo pipeline either in Galaxy or on command line. For each pair, we first construct a barcode which is the concatenation of the two tags from the two reads. Then the reads are sorted according to this compound barcode. Single stranded families from the same fragment will have the same 12bp tags, but in the opposite order: the α tag from one family will be the β tag in the other. In order to group single-strand families from the same fragment together, we normalize the order of the concatenation to produce a "canonical barcode" which will be identical for both strands. The order of the canonical barcode is determined by a simple string comparison. Then the original order of the tags is recorded in a separate field. Sorting the output groups the reads so that the two families constituting each duplex will be adjacent, with the read pairs separated by strand.

2.3.3 Aligning families and consensus calling

The reads in each single-strand family are aligned to themselves using a script calling the MAFFT multiple sequence aligner (K. Katoh and Standley 2013). These alignments were used to call the single-strand consensus sequences. First, a threshold is applied, requiring a specified number (default = 3) of reads to produce a consensus. Then, the consensus calling is performed by determining the majority base at each position. If no base is in the majority, "N" is used. Positions with gaps are considered in the same way as bases. Quality filtering is done at this

stage: bases with a PHRED quality score lower than a user-given threshold are not counted (default = 20). For positions with gaps, a quality score is calculated by considering the quality scores of the eight nearest bases. The calculated score is a weighted average, with the weight decreasing linearly with distance from the gap. Finally, duplex consensus sequences are called using the two single-strand consensus sequences. The two sequences are aligned using Smith-Waterman algorithm (using an existing C implementation from https://code.google.com/archive/p/swalign/), and then each pair of bases is compared. If the bases agree, that base is used in that position. If they disagree, the IUPAC ambiguity code for the two bases is used. Gap and non-gap characters produce an "N". If a single-strand consensus sequence has no matching opposite strand consensus, the user may choose to include the single-strand consensus in the output, direct it to a separate file, or discard it.

2.3.4 *In silico* mixture experiment

Twenty one heteroplasmies were randomly generated and inserted into the human mitochondrial genome (Revised Cambridge Reference Sequence (rCRS), NC_012920.1) at a spacing of at least 600 bp from each other and from the chromosome ends (the genome is circular but its textual representation is not). This is silico mutated sequence is referred to as mt-mut to distinguish it from the unmodified reference, which we would call mt-ref. Next, 600 bp fragments were randomly generated using wgsim (version 0.3.1-r13) with the error and mutation rate set to 0. For mt-mut and mt-red we generated 2,500 and 25,000,000 of such fragments, respectively. Each of the fragments was tagged on each end with a random, 12 bp barcode and a 5 bp linker sequence. Each was then subjected to *in silico* PCR and sequencing to create a family of reads descended from the same fragment. To determine the size of the family, a random number was chosen from an empirically-determined distribution, with a peak at 9 reads. A phylogenetic tree was simulated for the reads by starting at the last PCR cycle and coalescing backward, randomly joining branches based on the probability of two reads sharing an ancestor at that cycle (2-cycle). 30 cycles of PCR were simulated. Then, PCR polymerase errors were simulated by introducing random errors at each cycle, accumulating errors from each parent molecule. The error rate was 0.001 probability of an error per base, per cycle. Indels were given a 0.15 fraction of the errors, and a 0.3 probability of extension per base.

Finally, a pair of 250 bp reads was generated from each final fragment sequence. Sequencing polymerase errors were introduced at the same rates as PCR polymerase errors. Quality scores were not simulated and set to PHDER value of 40. The strandedness of each read pair was determined by which of the initial two potential daughters of the original fragment it was descended from.

Duplex consensus reads were created from these simulated reads using Du Novo with 3 reads required per single-strand consensus and base quality filtering turned off (PHRED threshold of 0). The reads were aligned to the mitochondrial genome (rCRS) with bwa-mem and filtered for alignments with a minimum mapping quality (MAPQ) of 20.

2.3.5 Error rate-singleton correlation simulation

In silico duplex sequencing of the human mitochondrial reference sequence (rCRS) was performed as described above, but with 10,000 400 bp fragments and 100 bp final reads to save computational time. Then, the reads were processed with the first part of the Du Novo pipeline, creating a strand-independent barcode from each read pair. Then, the total number of unique barcodes was counted, and the fraction of those that were present only once. This was performed once for each error rate setting.

2.4 Results and Discussion

2.4.1 A reference-free approach

Our approach is outlined in Figure 2.2. First, paired reads generated from a duplex sequencing experiment are merged into families. This is performed by sorting according to the barcode. Each fragment is expected to be represented by two single-stranded families corresponding to each strand. These two single-stranded families are expected to have the same unique tags, but in the opposite order: the α tag from one single-stranded family will be the β tag in the other (also see Figure 1 in (M. W. Schmitt et al. 2012)). In order to group single-stranded families from the same fragment together, we normalize the order of the concatenation to produce a “canonical barcode” (a concatenated string consisting of α and β tags), which will be identical for both strands. The order of the canonical barcode is determined

by a simple string comparison. Sorting the output groups the reads so that the two families constituting each duplex will be adjacent, with the read pairs separated by strand.

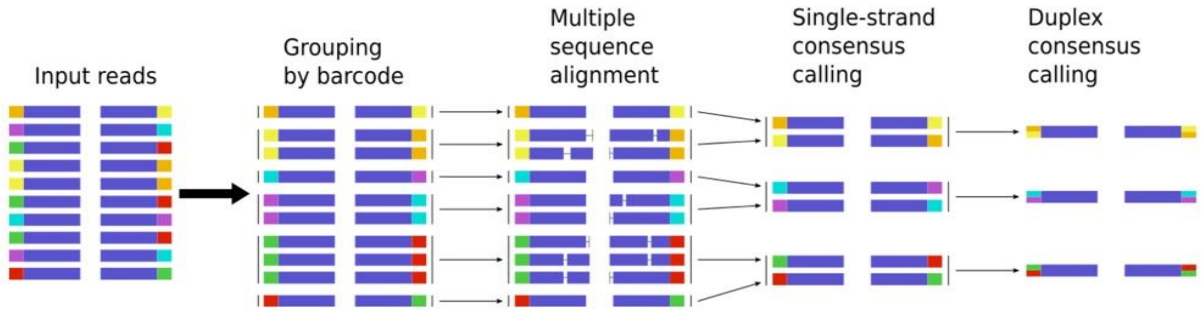


Figure 2.2 The Du Novo approach. First, reads tagged with identical barcodes are grouped into strand-specific families. Reads within each family are aligned and single-stranded consensus sequences (SSCSs) are generated. Finally, the SSCSs are reduced into duplex consensus sequences (DCSs).

Next, the reads in each single-strand family are aligned to themselves and these alignments are used to call the single-strand consensus sequence (SSCS). First, a threshold is applied, requiring a user-specified number of reads to produce a consensus (three by default). The consensus calling is conducted by determining the majority base at each position. If no base is in the majority, "N" is used. Positions with gaps are considered in the same way as bases. Quality filtering is performed at this stage: bases with a Phred quality score (Ewing and Green 1998) lower than a user-specified threshold are not counted (20 is used by default). For positions with gaps, a quality score is calculated by considering the qualities of eight neighboring bases. The calculated score is a weighted average, with the weight decreasing linearly with distance from the gap. Finally, a duplex consensus is called using the two SSCSs. The SSCSs are aligned using the Smith-Waterman algorithm (T. F. Smith and Waterman 1981), and then each pair of bases is compared. If the bases agree, that base is used in that position to generate duplex consensus. If they disagree, the IUPAC ambiguity code for the two bases is used. Gap and non-gap characters produce an "N". In the end the above approach reduces the initial set of sequencing reads to a collection of duplex consensus sequences (DCS; as the duplex sequencing experiments are performed with paired-end reads, the output of the procedure also consists of pairs corresponding to forward and reverse double-stranded

consensuses). DCS are then filtered (i.e., sequences with ambiguous nucleotides can be removed or trimmed), mapped against the reference genome, realigned to normalize gap-containing regions, and the resulting alignments are used to call variants. In this scenario variants are expected to have the full spectrum of allele frequencies between 0 and 1 and do not follow a diploid expectation. For that reason we use variant callers capable of dealing with this limitation such as the Naive Variant Caller (NVC; (Blankenberg et al. 2014)) or FreeBayes (Garrison and Marth 2012). Finally, variant calls are post-processed to compute the strand bias (using formulae from (Guo et al. 2012)). This approach is implemented in a pipeline relying exclusively on open-source software (https://github.com/galaxyproject/dunovo and accessible through the Galaxy system. We termed this approach Du Novo — for Duplex Sequencing de novo assembly-based calling.

2.4.2 Du Novo reliably identifies very low frequency variants

First, we evaluated the performance of Du Novo by applying it to a dataset generated from a simulated mixing experiment. The advantage of performing the simulation is that the "truth" is known explicitly. We randomly generated 21 "heteroplasmies" by modifying human mitochondrial sequence. This altered version of the mitochondrial genome was then "mixed" with unmodified reference sequence at a ratio of 1:10,000 (thus each "heteroplasmy" in this mix has the frequency of 0.0001), and a duplex experiment was simulated on the mixture. This was done by randomly generating 2,500 fragments from the altered sequence and 25,000,000 fragments from unmodified reference, adding barcodes, and performing in silico PCR and sequencing (see Methods). The polymerase error rate in PCR and sequencing was set at 0.1% per base. After applying Du Novo to the simulated reads and aligning the DCS to the mitochondrial reference, the median read depth was 166,574×. Next, we identified all variable sites and filtered them using a series of minor allele frequency (MAF) thresholds and requiring a minimum DCS coverage of 10,000. The relationship between MAF thresholds and the numbers of false positives and false negatives is shown in Figure 2.S1. Du Novo correctly identifies 20 of the 21 variants, with no false positives. The remaining variant was present at a frequency of 0.00004 (likely a result of random fluctuation), along with 46 false positives with an equal or higher MAF.

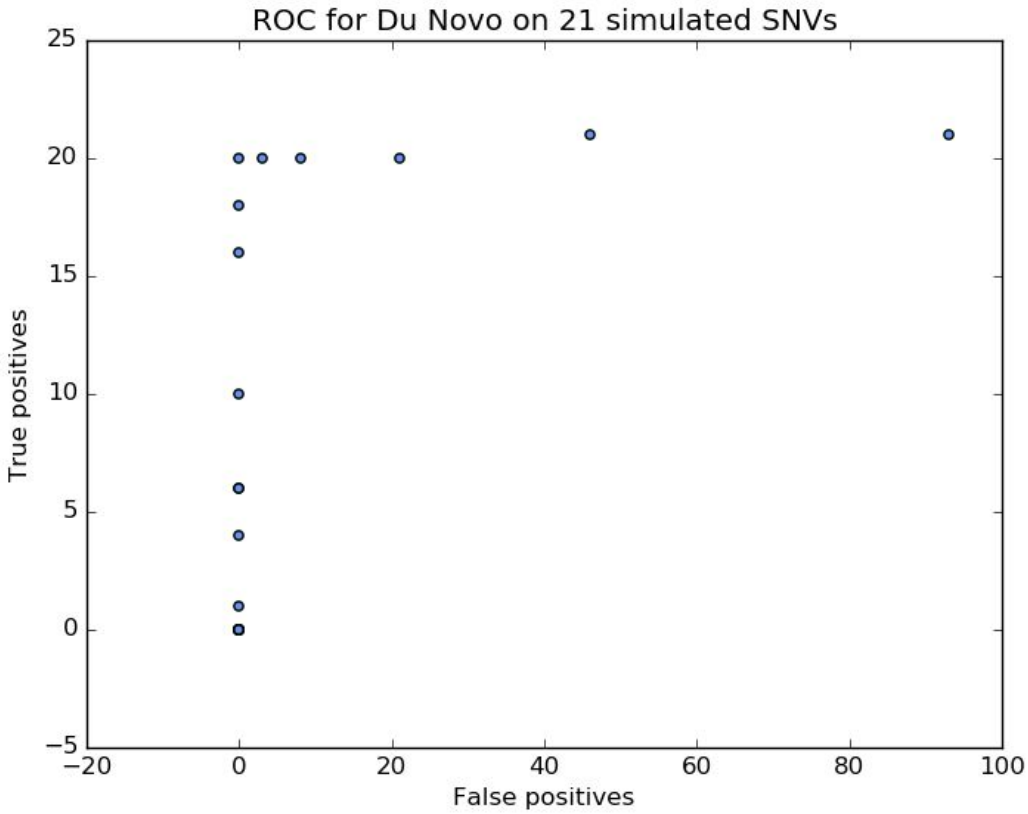


Figure 2.S1 Receiver operating characteristic (**ROC**) for Du Novo detecting 21 artificial heteroplasmies in a simulated duplex sequencing experiment. Shown are true positives versus false positives detected using different minor allele frequency thresholds, in steps of 0.00001 (the depth of coverage threshold was held constant at 10,000×). At the *bottom left*, no heteroplasmies at all are detected at a threshold MAF of 0.00016. The first variant is detected at a MAF of 0.00015, with no false positives. Continuing upward, no false positives are detected while increasing true positives are found until the *upper left corner* at a MAF of 0.00008, with 20 true positives and no false positives. Then, increasing false positives are found with no gain in true positives until the last true single-nucleotide variant (**SNV**) is found at a MAF of 0.00004, with 46 false positives also observed at that threshold.

2.4.3 Comparison with original approach: Du Novo replicates published estimates

To assess the performance of our method on real-world data and to compare it head-to-head with the original approach of (Kennedy et al. 2014) we re-analyzed a recently published dataset by Schmitt and colleagues (Michael W. Schmitt et al. 2015) using both

methods. In (Michael W. Schmitt et al. 2015) the authors have employed duplex sequencing to identify a rare mutation at the *ABL1* locus responsible for resistance to a chronic myeloid leukemia therapeutic compound imatinib. The resistance is conferred by the presence of G-to-A substitutions within the *ABL1* coding region resulting in an E279K amino acid replacement. This substitution is present in a small sub-clonal subset of cells at an ~1% frequency. The dataset (SRA accession SRR1799908) contains 6,921,891 read pairs representing 1,468,089 unique tag combinations (potential families; see Table 2.1).

Number of	*ABL1*	SC8
read pairs	6,921,891	17,385,100
unique tags	1,467,768	2,100,705
unique $\alpha\ \beta$ configurations	748,411	1,148,444
unique $\alpha\ \beta$ configurations with 1 read-pair	677,069	884,295
unique $\alpha\ \beta$ configurations with ≥ 3 read-pairs	60,333	222,823
unique $\beta\ \alpha$ configurations	743,669	1,092,748
unique $\beta\ \alpha$ configurations with 1 read-pair	672,946	832,875
unique $\beta\ \alpha$ configurations with ≥ 3 read-pairs	60,032	140,486
unique $\alpha\ \beta\ \beta\ \alpha$	24,313	140,485
unique $\alpha\ \beta\ \beta\ \alpha$ with ≥ 3 read-pairs on both strands	20,746	109,999
reads within $\alpha\ \beta\ \beta\ \alpha$ families with ≥ 3 read-pairs on both strands	2,156,105	8,636,692

Table 2.1 Characteristics of *ABL1* and SC8 duplex sequencing experiments.

First, we analyzed this dataset with Du Novo. Requiring each family to contain at least three reads reduced this number to 120,365 SSCSs and reconciling these into duplex consensus sequences (DCS) further reduced this number to 20,467 DCSs constructed from 2,083,140 read pairs (the remaining 6,921,891 - 2,083,140 = 4,838,751 were represented by families with less than three reads and were omitted; see Figure 2.S2). Mapping DCS to the reference human genome showed the G-to-A substitution with frequency varying from 1.28% to 1.31% depending on the variant caller (NVC (Blankenberg et al. 2014) and FreeBayes (Garrison and

Marth 2012), respectively) but irrespective of the mapper used (bwa-mem (H. Li 2013) or bwa (H. Li and Durbin 2009)).

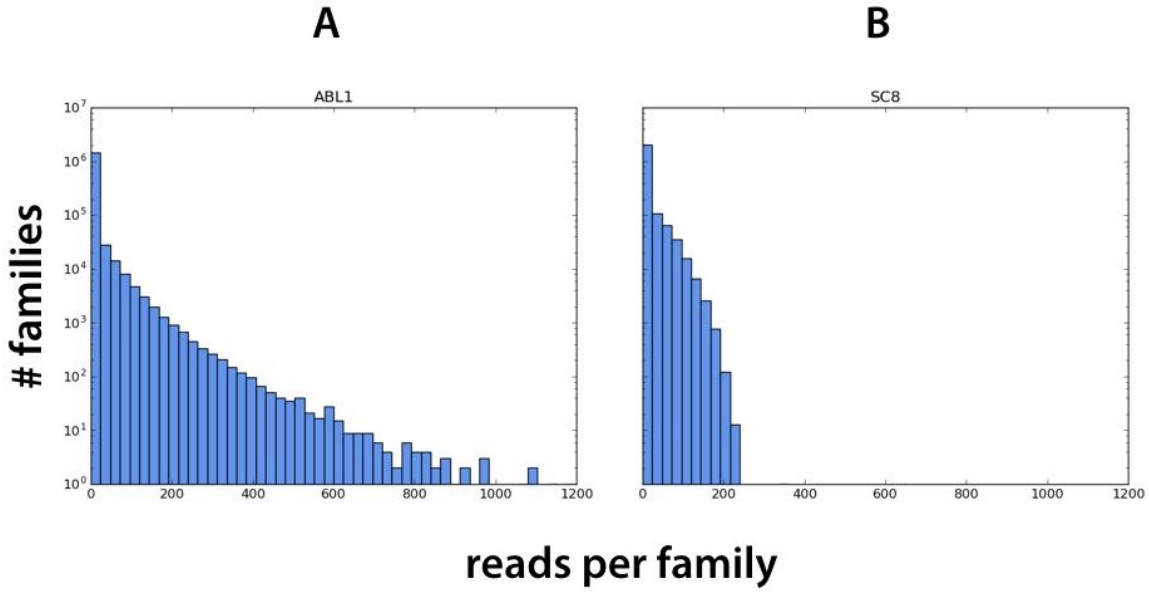


Figure 2.S2 Distribution of reads per family in *ABL1* (**a**) and SC8 (**b**) datasets.

Next, we repeated this experiment with the published duplex sequencing pipeline (Kennedy et al. 2014). This produced 1.29% and 1.31% frequencies at the G-to-A substitution site for NVC and FreeBayes, respectively. Thus the allele frequencies estimates were essentially identical between the two approaches. The Du Novo produced a higher depth at the variable site: 1,099 vs. 618 for our method vs. the published pipeline (Kennedy et al. 2014), respectively. However, at such low allele frequencies even a formidable coverage results in a relatively small proportion of reads supporting the minor allele. For example, in the case of this analysis the minor allele ('A') is supported by 14 duplex consensuses from the total of 1,099 resulting in the minor allele frequency of 1.28%. Yet each of these 14 families is in turn derived from multiple starting reads ranging from a minimum of 5 to a maximum of 102 (Figure 2.3A) providing additional support for the reliability of the minor allele calls.

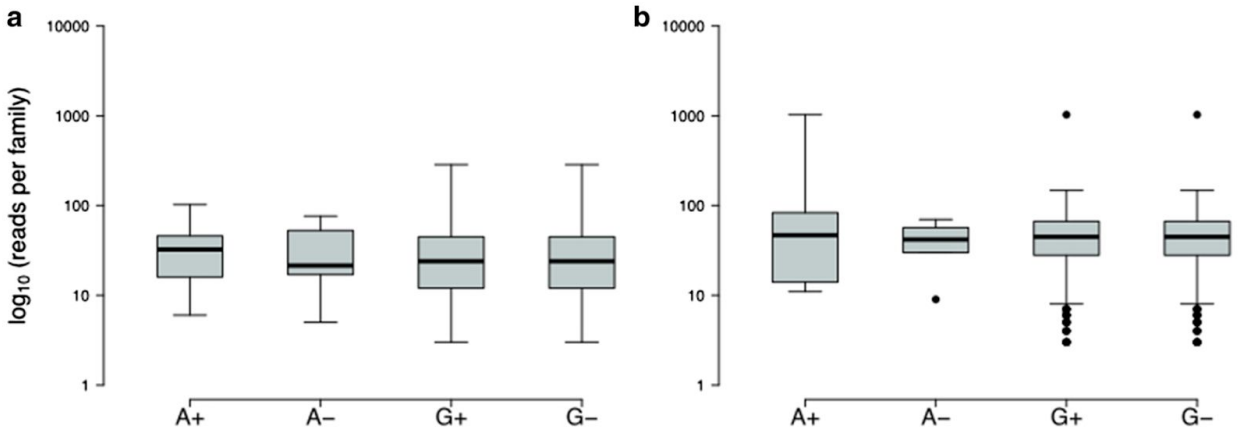


Figure 2.3 Distribution of family sizes (number of reads per family) supporting A and G alleles on both strands (plus and minus) for **a** site 130,872,141 in the *ABL1* dataset and **b** site 13,708 in the SC8 dataset.

2.4.4 Using Du Novo to call low-frequency heteroplasmies at mtDNA

After ensuring the adequate performance of Du Novo on the *ABL1* data, we applied it to the identification of low-frequency variants in human mitochondrial DNA. Previously, we have reported 174 point heteroplasmies identified from the analysis of mtDNA in 39 mother-child pairs (a total of 156 samples = 39 mothers × 2 tissues + 39 children × 2 tissues (Rebolledo Jaramillo et al. 2014)). We have chosen family SC8 as it displays significant variability across samples and individuals. There are two heteroplasmic sites in this family — at positions 7,607 and 13,708. According to our published results (Rebolledo Jaramillo et al. 2014), the minor allele frequencies at site 7,607 are 0.7%, 1.1%, 0.0%, and 0.0% in mother's buccal tissue, mother's blood, child's buccal tissue, and child's blood, respectively. The corresponding MAFs at site 13,708 are 0.0%, 0.0%, 2.2%, and 1.6%. To verify these frequencies, we performed the duplex sequencing experiment using genomic DNA extracted from SC8 child buccal tissue in which mitochondrial DNA has been enriched via the long-range PCR as previously described in (Rebolledo Jaramillo et al. 2014). We started with 17,385,100 read pairs that contained 2,100,704 unique tags and were assembled into 82,230 DCSs. The estimated allele frequency at position 13,708 was 0.53%, a figure substantially lower than 2.2% estimated previously (Rebolledo Jaramillo et al. 2014). The coverage at this site was 1,138 with 6 reads representing

the minor allele ('A'). To check the reliability of this call we estimated strand bias (SB; using formula 1 from (Guo et al. 2012)) for all sites with MAF ≥0.5%. There were 20 sites (excluding 13,708) with MAF ranging from 0.51% to 21.2% and with SB values ranging from 0.94 to 6.08 (the lower the value, the less SB there is at a site; 0 is an ideal value (Guo et al. 2012)). SB = 0.01 at site 13,708, which is outside of the SB distribution for all other variable sites in our sample, strongly suggesting that this is the only true heteroplasmy in this sample. In addition, examining individual DCSs at this site indicates that each of them is generated from a large number of original reads (Figure 2.3B) confirming this polymorphism, albeit at a significantly lower frequency.

2.4.5 The utility of SSCS

In the SC8 experiment described above, we estimated MAF at site 13,708 to be 0.53% — a much lower value compared to the original one (2.2%) obtained from re-sequencing (Rebolledo Jaramillo et al. 2014). The likely cause of this deviation lies in the design of the duplex experiment. In this study we performed duplex sequencing not directly on mitochondrial DNA, but instead on products of a long-range PCR (see Methods). In this particular case this is unavoidable as the samples are obtained by a minimally invasive "cheek swab" resulting in a very low concentration of mitochondrial DNA. The core issue is that complementary strands of the resulting PCR products (the starting material for our duplex sequencing experiment) can randomly pair after amplification, forming heteroduplexes and leading to an underestimation of minor allele frequencies when using DCSs only (Figure 2.S3). To test whether this indeed is the cause of MAF underestimation, we performed variant calling using SSCSs instead of DCSs and obtained a MAF of 1.7% (strand bias = 0.02 and depth = 4,548), a value much closer to 2.2% reported in the original publication. Thus, although the background error frequency is higher for SSCS in comparison with DCS (M. W. Schmitt et al. 2012) in certain situations, such as experiments using ampliconic DNA, the use of SSCS for polymorphism detection may be preferable to obtain more accurate allele frequencies.

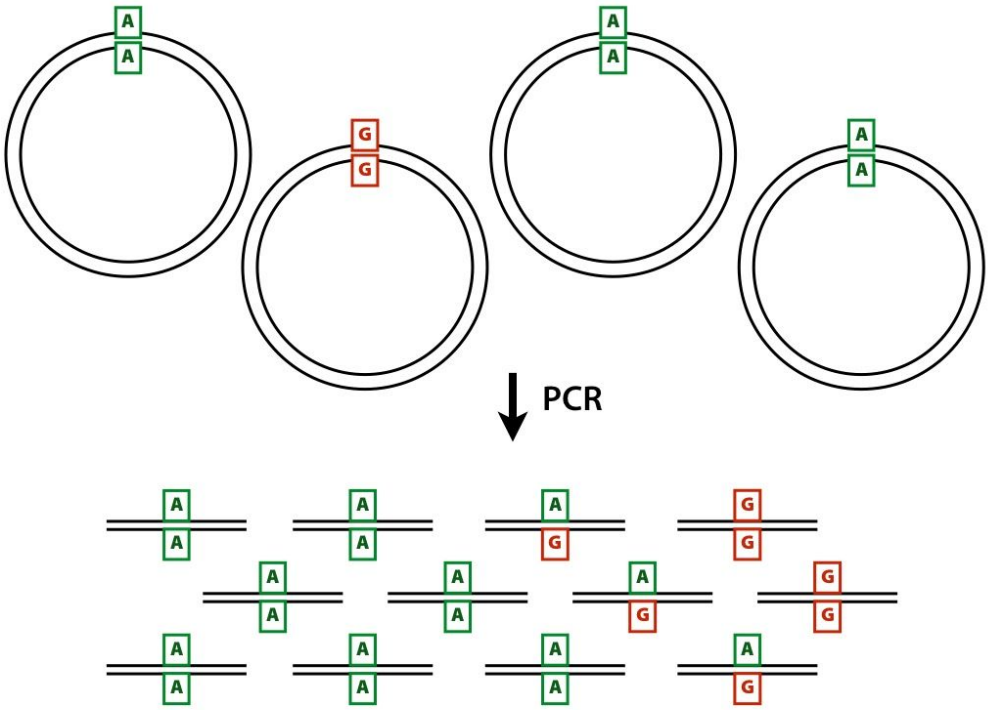


Figure 2.S3 Here there are two distinct types of mitochondrial genomes: carrying A and G. Because the population of genomes is enriched via PCR, heteroduplex formation takes place, skewing frequency estimates performed using DCSs. If this PCR-derived DNA is now used as the starting material for a duplex sequencing experiment, the heteroduplex molecules will manifest themselves as having an N base at this site (because Du Novo interprets disagreements as Ns during consensus generation). So, DCSs produced from this dataset will have A, G, and N at the polymorphic site. Yet, SSCSs will only have A and G. Thus, SSCS will give a more accurate estimate of the allele frequency at this site in this particular case.

2.4.6 Loss of data as a result of sequencing errors in duplex tags

One of the fundamental weaknesses of duplex sequencing is the fact that the majority of families in a duplex experiment contain only a single read pair (Figure 2.S4). This eliminates a substantial amount of otherwise useful data from the analysis, contributing to the inefficiency of the current protocol. To understand the potential sources of read loss, we examined individual stages of the duplex analysis process. This information is compiled in Table 2.1 and is based on the re-analysis of both previously published data (*ABL1* data; (Michael W. Schmitt et

al. 2015)) and results generated in our laboratory (the SC8 dataset described above). Both cases feature a large number of initial read pairs and unique tags. However, these numbers are rapidly reduced by requiring at least 3 reads within each single stranded family. Combining SSCSs into DCSs also greatly reduces the number of useful sequences, since both strands must be present and meet the 3 read threshold. One potential explanation for the large number of families with only one read pair is sequencing errors within duplex tags. Each barcode with an error will almost certainly be unique, creating an entirely new apparent family with only one member. The number of reads with an erroneous barcode may be a minority, but this can still result in the number of families with erroneous barcodes being very high (a majority). The fraction of erroneous barcodes (r) can be expressed in the following form:

$$(1)\ r = 1 - (1 - E)^{l}$$

where E is the per-base error rate and l is the barcode length (in this case 24 as it is a combination of α and β tags, each of which is 12 nucleotides). Here, E is a cumulative error rate taking into account the chance of a mutation at every cycle of PCR, plus the sequencing reaction. The cumulative error rate can be calculated from the error rate at each stage using the same equation (1), this time using E as the error rate per base per stage, l as the number of stages (number of PCR cycles plus 1 for the sequencing reaction), and r as the cumulative error rate. Even assuming a low per-stage error rate of 0.1%, this gives a cumulative error rate of about 3%. Using this in equation (1) again, we obtain the fraction of barcodes expected to contain an error to be 52.5%:

$$(2)\ r = 1 - (1 - 0.03)^{24} \approx 0.525$$

Now, suppose in a hypothetical duplex experiment 10 initial fragments of DNA were ligated with α and β adapters (a unique α and β for each of the 10 fragments), and the subsequent PCR amplification and Illumina sequencing process produced 100 read pairs (10 pairs per original fragment). If there are no errors, these 100 read pairs should be recognized as members of 10 duplex families during the analysis stage. If we now factor in the erroneous barcode rate of ~52% calculated above, one would observe 62 total families: 10 real families and 52 artifactual families consisting of a single read pair. This phenomenon increases the total number of families by reducing the read count within legitimate families — a trend apparent in

real data (Figure 2.S2). Furthermore, the relationship between the number of single-read families and the total number of reads can serve as a proxy for the error rate. For example, in the SC8 experiment there were 1,717,170 single read families and 17,385,100 total read pairs. Assuming that all single read families are byproducts of sequencing errors within duplex tags, this gives 1,717,170/17,385,100 = 0.098 as the fraction of erroneous barcodes (r). With l = 24 we can solve equation (1) for E obtaining an estimate of ~0.4% for the cumulative error rate.

To test this reasoning we simulated duplex experiments with different error rates. The starting distribution of family sizes was constant in each case, with 1.20% of fragments producing a family with only one read. With an error rate of zero, the proportion of output families which were composed of a single read was, as expected, precisely 1.20% (Figure 2.S4), meaning no excess beyond those with a natural family size of one. When the error rate was raised to 0.1% per base per cycle of PCR/sequencing reaction, 75.5% of output families were composed of a single read. This meant that 74.3% of families were artifacts consisting of a read which originating from a fragment that produced multiple reads. Instead of being grouped with its sibling reads, each of these instead was grouped by itself because of an error in the barcode.

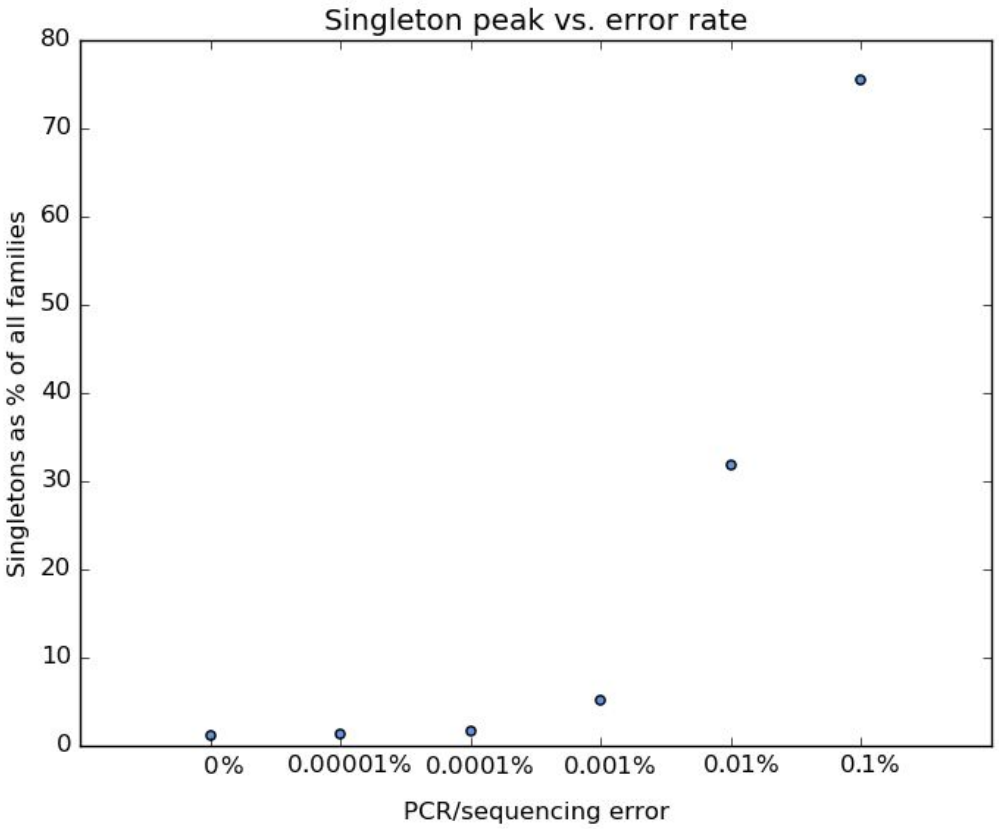


Figure 2.S4 Effect of errors on the number of single-read families. Duplex sequencing was simulated using different values for the PCR/sequencing polymerase error rates. In each case, 10,000 400-bp fragments were generated from the mitochondrial reference sequence. After simulating the duplex method, the number of reads observed for each unique barcode was counted. Shown are the fraction of families with only one read versus the polymerase error rate.

While this was only a simulation and the above calculations make a number of simplifying assumptions, they nevertheless highlight the significance of sequencing errors within tags as one of the main causes of data loss. We are currently developing a family reconstruction approach that would allow mismatches in tags and is expected to significantly reduce the number of single read families.

2.4.7 Interactive analysis of duplex data

The underlying components of the Du Novo process are distributed as an open source software and can be used from the command line (https://github.com/galaxyproject/dunovo). However, to increase the number of potential users we also make Du Novo accessible through the Galaxy system (http://usegalaxy.org). Figure 2.4 illustrates all stages of the duplex analysis workflow. This example begins with fastq datasets generated by an Illumina machine that are used as inputs in the Du Novo pipeline. Initially, reads are processed to identify and count duplex tags (Make families). Reads having identical tags (families) are aligned (Align families) and alignments are reduced to DCSs (Make consensus reads). The DCSs are trimmed to remove ambiguous nucleotides (Sequence Content Trimmer), converted to fastq format (this is because DCSs are reported as fasta datasets; Combine FASTA and QUAL), and mapped to the reference genomes (in this example with both BWA and BWA-MEM). BAM datasets produced by mappers are combined (MergeSamFiles), realigned (BamLeftAlign), and variable sites are identified with the Naive Variant Caller (NVC). A Variable Call Format (VCF) dataset generated by NVC is processed by Variant Annotator, which tabulates allele frequencies and strand bias values. Finally the data are filtered on minor allele frequency (≥0.5%) and strand bias (<1). This workflow is available at https://usegalaxy.org/u/aun1/w/duplex-analysis-from-reads. The most computationally demanding portion of the workflow is the alignment of reads within each family (Align Families). For instance, processing of 6,921,891 read pairs comprising the *ABL1* dataset [9] took an average of 0.004 seconds per pair or approximately 9 hours of wall time on a 16 CPU cluster node. One of the advantages of using Galaxy at https://usegalaxy.org/ for the analysis of duplex sequencing data is that its underlying infrastructure relies on high-performance resources provided by the Texas Advanced Computing Center (TACC) and the Extreme Science and Engineering Discovery Environment (XSEDE) making it possible to perform analyses of multiple duplex datasets by multiple users simultaneously.

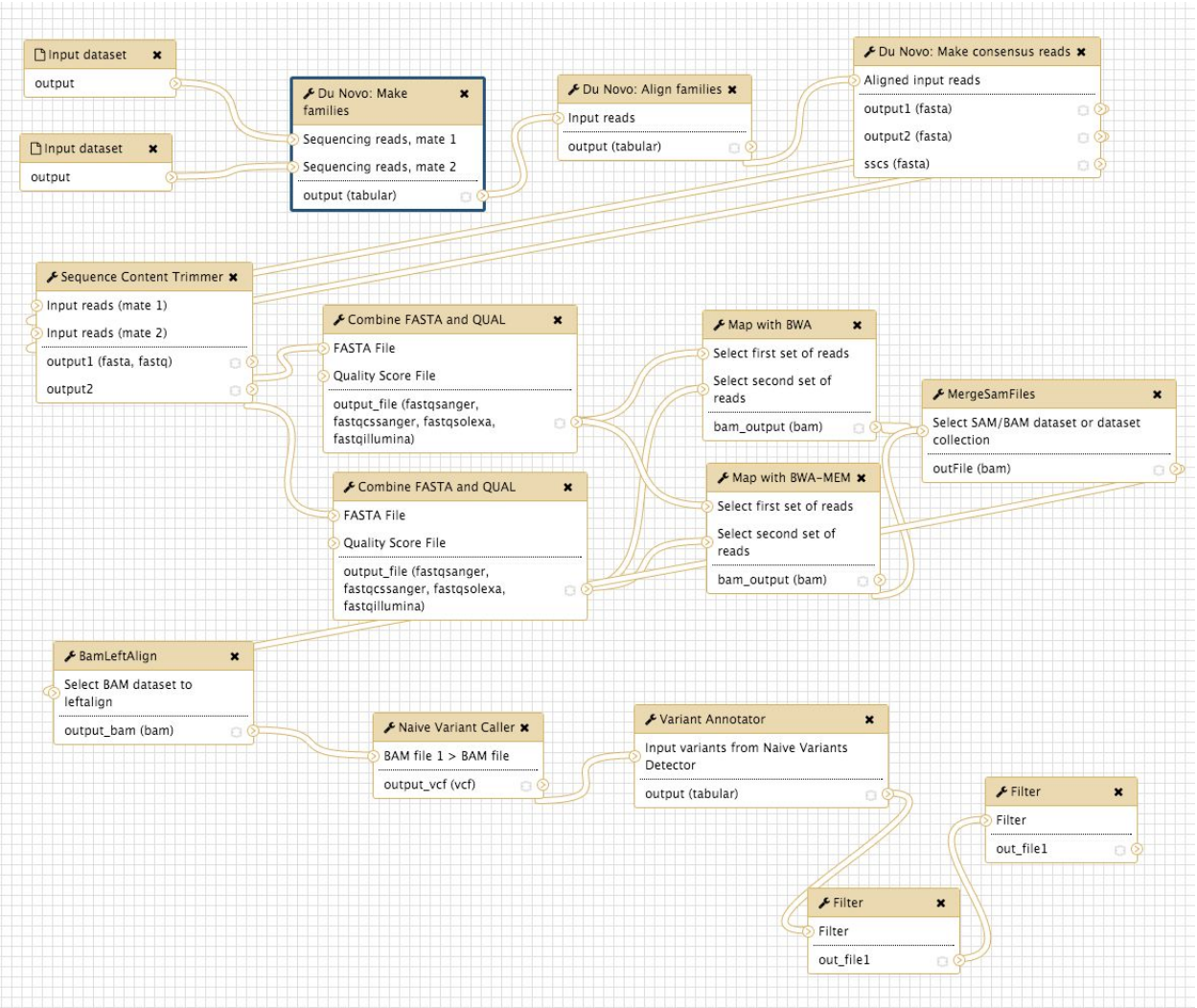


Figure 2.4 A complete workflow implementing the Du Novo approach to variant discovery from duplex sequence data. It is accessible from http://usegalaxy.org/duplex.

2.5 Declarations

2.5.1 Acknowledgements

We are grateful to Kristin Eckert, Suzanne Hile, and Howard Fescemeyer for their advice regarding establishing duplex sequencing in our laboratory. Marcia Shu-Wei Su has performed the enrichment of mitochondrial DNA via long range PCR. Nathan Coroar provided assistance

with tuning our software for the optimal utilization of high performance compute infrastructure. Michael Schmitt provided advice on the organization of the *ABL1* dataset.

2.5.2 Funding

This project was supported by NIH Grants U41 HG005542 to AN and R01 GM116044 to KDM. Additional funding is provided by Huck Institutes for the Life Sciences at Penn State and, in part, under a grant with the Pennsylvania Department of Health using Tobacco Settlement Funds. The Department specifically disclaims responsibility for any analyses, interpretations, or conclusions.

2.5.3 Availability of Data and Materials

All software described here is accessible from http://usegalaxy.org/duplex under the terms of BSD Open Source License. This publication is based on version 0.4 of Du Novo accessible from https://github.com/galaxyproject/dunovo/releases/tag/v0.4 (doi:10.5281/zenodo.57256). The *ABL1* dataset from Schmitt and colleagues is available from the Short Read Archive under the accession SRR1799908. The SC8 dataset is available from the Short Read Archive under accession SRR3749606.

2.5.4 Authors' contributions

NS developed software and performed initial analyses; BA performed the duplex sequencing procedure; WG performed data analysis; KM and AN wrote the paper and designed the Galaxy workflow. All authors read and approved the final manuscript.

2.5.5 Competing interests

The authors declare that they have no competing interests.

2.5.6 Ethics approval and consent to participate

All experimental procedures described herein comply with the principles of the Helsinki Declaration. Buccal swabs from subject SC8C1 have been collected under IRB protocol number 30432EP. SC8C1 is a part of a larger study performed within the framework of this IRB protocol. In this study patients at the pediatric outpatient clinic (located the Pennsylvania State

University Hershey Medical Center in Harrisburg, Pennsylvania) were approached by experienced coordinators and invited to participate. Samples were collected from participants providing verbal consent in the clinic. Only date of birth and date of collection were associated with each sample. No personally identifiable information was recorded.

3 Family reunion via error correction: An efficient analysis of duplex sequencing data

This chapter is reproduced from a research article published in BMC Bioinformatics in 2020 by Nicholas Stoler, Barbara Arbeithuber, Gundula Povysil, Monika Heinzl, Renato Salazar, Kateryna D Makova, Irene Tiemann-Boege, and Anton Nekrutenko (DOI: 10.1186/s12859-020-3419-8).

3.1 Abstract

3.1.1 Background

Duplex sequencing is the most accurate approach for identification of sequence variants present at very low frequencies. Its power comes from pooling together multiple descendants of both strands of original DNA molecules, which allows distinguishing true nucleotide substitutions from PCR amplification and sequencing artifacts. This strategy comes at a cost—sequencing the same molecule multiple times increases dynamic range but significantly diminishes coverage, making whole genome duplex sequencing prohibitively expensive. Furthermore, every duplex experiment produces a substantial proportion of singleton reads that cannot be used in the analysis and are thrown away.

3.1.2 Results

In this paper we demonstrate that a significant fraction of these reads contains PCR or sequencing errors within duplex tags. Correction of such errors allows "reuniting" these reads with their respective families increasing the output of the method and making it more cost effective.

3.1.3 Conclusions

We combine an error correction strategy with a number of algorithmic improvements in a new version of the duplex analysis software, Du Novo 2.0. It is written in Python, C, AWK, and

Bash. It is open source and readily available through Galaxy, Bioconda, and Github: https://github.com/galaxyproject/dunovo.

3.2 Background

Numerous, often clinically important, research scenarios require detection of sequence variants that are present in a minute fraction (10^{-5}–10^{-9}) of molecules under study. Examples include detection of cancer-related mutations in liquid biopsies, identification of fetal DNA in a mother's bloodstream, assessing dynamics of the immune system, tracing mutational landscape of bacteria through the evolution of antibiotic resistance, studying genomic changes in viral pathogens and many others (for a comprehensive review see (Salk, Schmitt, and Loeb 2018)). Conventional approaches, where a sample is sequenced and resulting reads are aligned against a reference genome to find differences, are ill suited for variants present at frequencies below 1% (Rebolledo Jaramillo et al. 2014; Michael W. Schmitt et al. 2015). A number of techniques has been developed to circumvent this issue with Duplex Sequencing (DS) being currently the most sensitive (Salk, Schmitt, and Loeb 2018; M. W. Schmitt et al. 2012). DS is based on using unique tags (also called barcodes throughout this manuscript) to label individual molecules of the input DNA. During amplification steps that are required for the preparation of Illumina sequencing libraries, each of these molecules gives rise to multiple descendants. The descendants of each original DNA fragment are identified and grouped together using tags—one simply sorts tags in sequencing reads lexicographically and all reads containing the same tag are bundled into a family. These families (with at least three members or more) form single stranded consensus sequences (SSCS) for the forward or the reverse strand, respectively. Complementary SSCSs are then grouped to produce duplex consensus sequences (DCSs; see Fig. 3.1). A legitimate sequence variant is found in the majority of the reads within a family. In contrast, sequencing and amplification errors will manifest themselves as "polymorphisms" within a family and so can be identified and removed (yellow rectangles in Fig. 3.1).

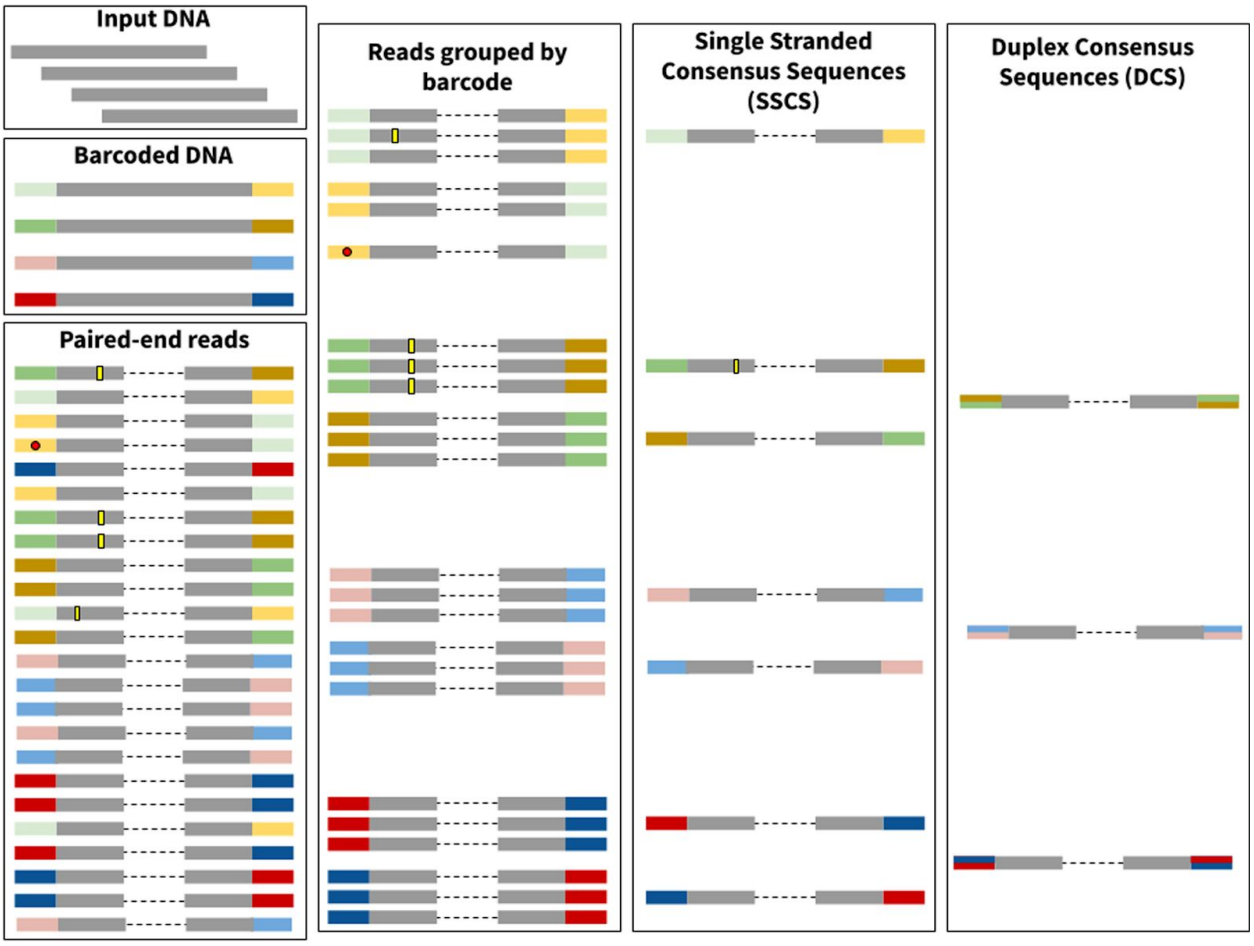


Figure 3.1 Effect of errors on the Duplex Sequencing procedure. Here input DNA is sheared and barcodes are ligated to the ends of the DNA molecules (colored rectangles in Barcoded DNA). After paired-end library preparation and sequencing each original molecules gives rise to multiple reads (Paired-end reads pane). This process also inadvertently generates sequencing errors represented by yellow rectangles and red circles. The yellow rectangles and red circles are used to depict errors arising inside read compartments corresponding to original DNA and adapters, respectively. Reads are then grouped by barcode to produce "families". In this example each family is required to contain at least three reads. As shown here one of the reads contains an error (red circle) within the barcode. The error makes this particular barcode different from others. As a result it cannot be added into the family and remains a singleton (the error correction algorithm described here was developed specifically to correct such errors and allow singletons to be joined with their respective families). Each family is subsequently reduced into a Single Strand Consensus Sequence (SSCS) and each respective SSCS is merged with its counterpart from the opposite strand to generate Duplex Consensus Sequences (DCS).

Despite its power DS is a complex technique. Reliable identification of sequence variants requires each initial fragment to form a family with at least three members for each strand. To achieve this, it is necessary to precisely quantify the amount of input DNA during the library preparation step. Too much DNA results in small family sizes and makes variant identification impossible, while too little creates very large families at the expense of sequencing coverage. Furthermore, because DS barcodes are a part of the sequencing read, they accumulate PCR and sequencing errors. These errors prevent matching barcodes and therefore artificially split DS read families (red dots in Fig. 3.1) decreasing the efficiency of the procedure. In this manuscript we describe a new, efficient approach to the analysis of DS data that includes barcode error correction. It significantly improves the yield and performance of the technique. We also describe new quality control approaches designed to increase output of DS experiments.

3.3 Results and Discussion

3.3.1 Datasets

To test our results we used two previously published datasets. The first dataset was produced by Schmitt et al. (Michael W. Schmitt et al. 2015), who employed DS to identify a rare mutation at the *ABL1* locus responsible for resistance to a chronic myeloid leukemia therapeutic compound imatinib. The second dataset was produced by our group as a part of an experimental evolution study where DS was used to track frequencies of adaptive mutations in plasmid pBR322 (Mei et al. 2018).

3.3.2 Barcode errors result in lost data

Typical DS tags are randomized 12-mers. Since each DNA fragment is labeled by two tags, one at each end, the theoretical upper bound for the number of unique combinations is $4^{(12+12)}$. However, the input DNA in a standard DS experiment contains ~10^6 – 10^{11} molecules, creating a large tag-to-input excess ($4^{24} \gg 10^{11}$). Because of this excess it is highly unlikely to observe distinct input DNA molecules tagged by barcodes that are highly similar to each other. In fact, we can use this assumption to identify barcodes containing sequencing errors: barcodes

that differ from each other by just a few nucleotides are likely descendants of the same original sequence with differences introduced by PCR and/or sequencing errors.

To check the validity of this reasoning, we analyzed barcodes from the two datasets mentioned above — *ABL1* and pBR322. To do so we trimmed barcodes off of all sequencing reads generating a list of 12 + 12 barcode combinations. We then selected 1,000 random combinations from this list to reduce the time required for subsequent computation (out of 1,492,080 and 671,290 barcode combinations for *ABL1* and pBR322 datasets, respectively). Because duplex reads derived from different strands of the same original fragment contain the same two 12 mers but in different order (Fig. 3.1) there is a total of 2,000 tags we chose for this analysis. This is because for each 12 + 12 bp combination ab (a is the first 12-mer and b is the second) we also selected a complementary arrangement ba. Next, we compared each of the 2,000 tag combinations (concatenated into a single 24 nucleotide fragment) against all other tag combination in the entire dataset. At each comparison we calculated the number of differences (edit distance). Results of this analysis are given in Fig. 3.2 (dark blue bar "0 mismatches"). One can see that ~100 barcode combinations (62 and 104 for *ABL1* and pBR322, respectively) out of 2,000 tested have counterparts that differ by a single nucleotide, a difference that is likely introduced by a sequencing and/or PCR error. Because of this difference reads with error-containing tags will not be included into families during the standard duplex data analysis and will be effectively lost. Fig. 3.3 (dark blue bars "0 mismatches") illustrates this point. Here we sorted all combinations of barcodes in lexicographic order and counted the number of times each combination appears in this list. There is a striking abundance of combinations that appear only once. Such singletons cannot be used in an analysis and are discarded. However, they account for a large fraction of total sequencing output of the two experiments. Our goal was to see if barcode error correction can reduce such waste by recovering reads forming singleton families and returning them into analysis.

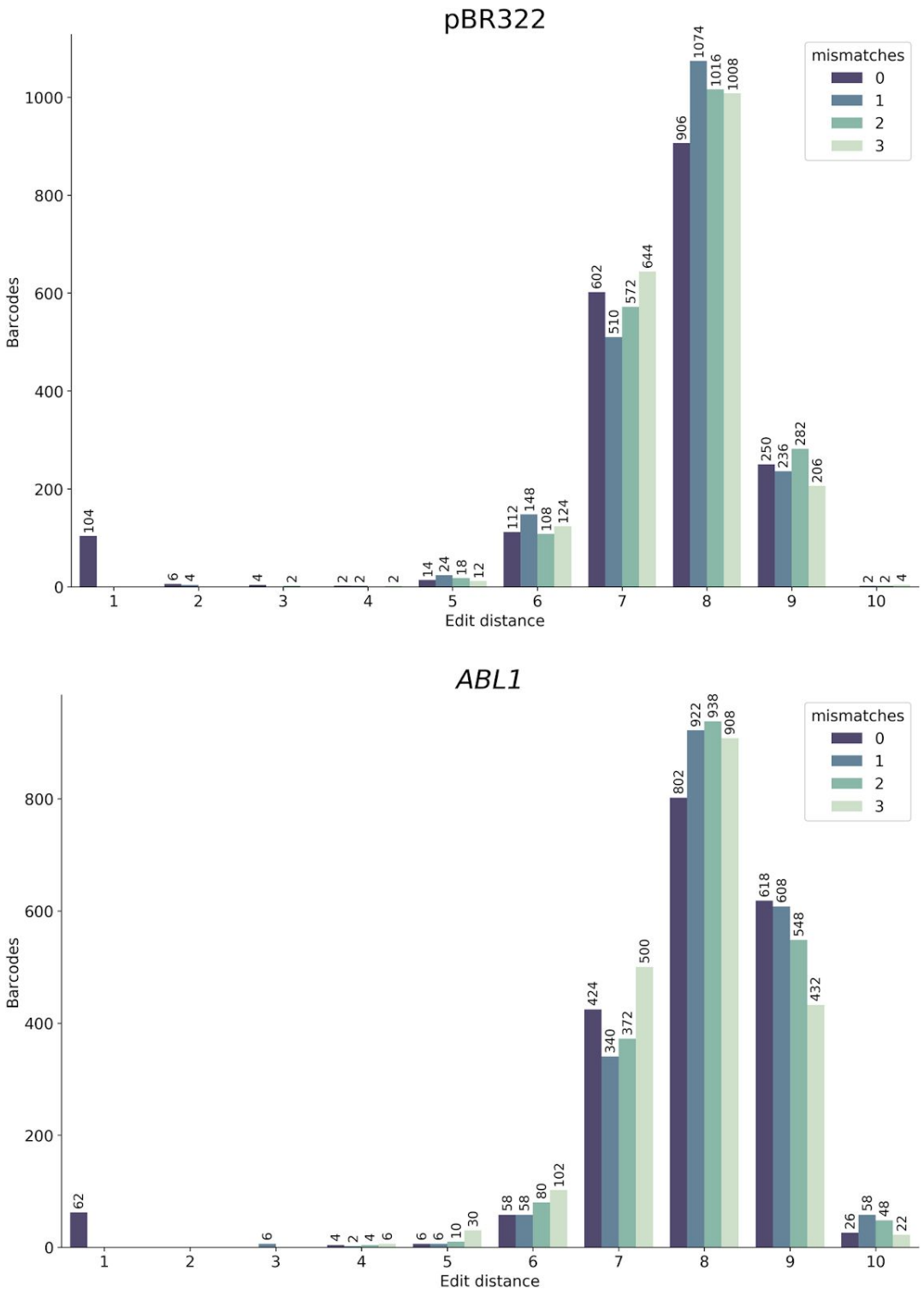
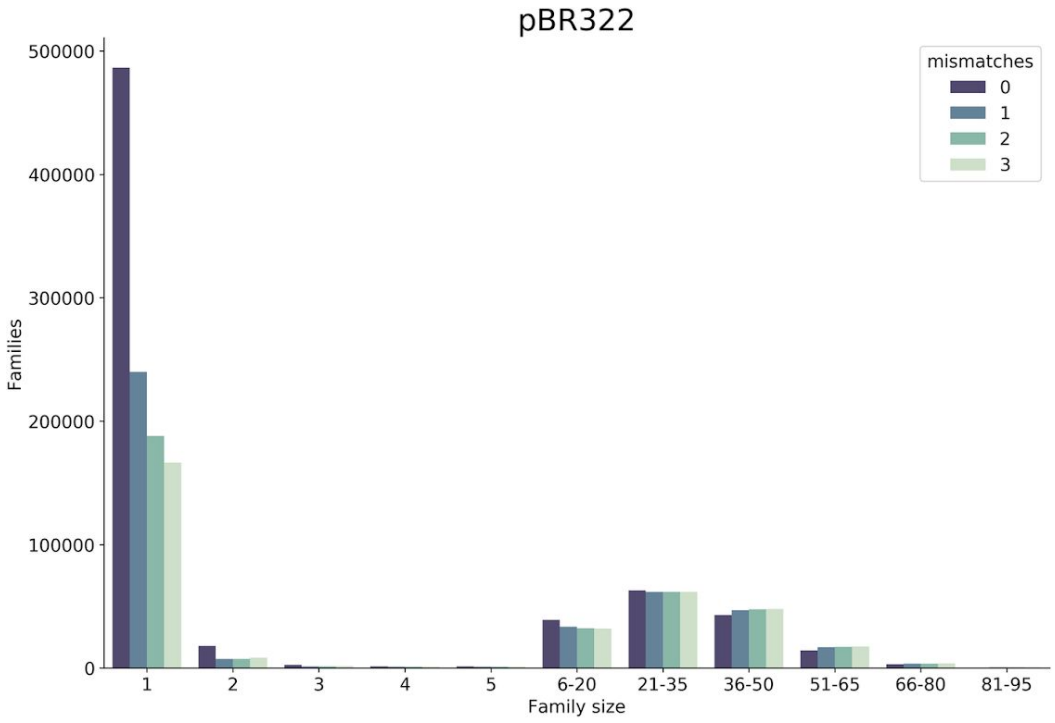


Figure 3.2 Analysis of inter-barcode edit distances with and without error correction. A randomly selected set of 2000 barcode combinations were compared against all barcodes in *ABL1* and pBR322 datasets before (0 mismatches) and after (1, 2, or 3 mismatches) error correction. The Y-axis is the

number of barcodes and the X-axis is the edit distance. For example, without error correction 104 barcodes differ by one nucleotide from barcodes in the entire pBR322 dataset. Error correction completely abolishes barcodes with 1 nucleotide difference.



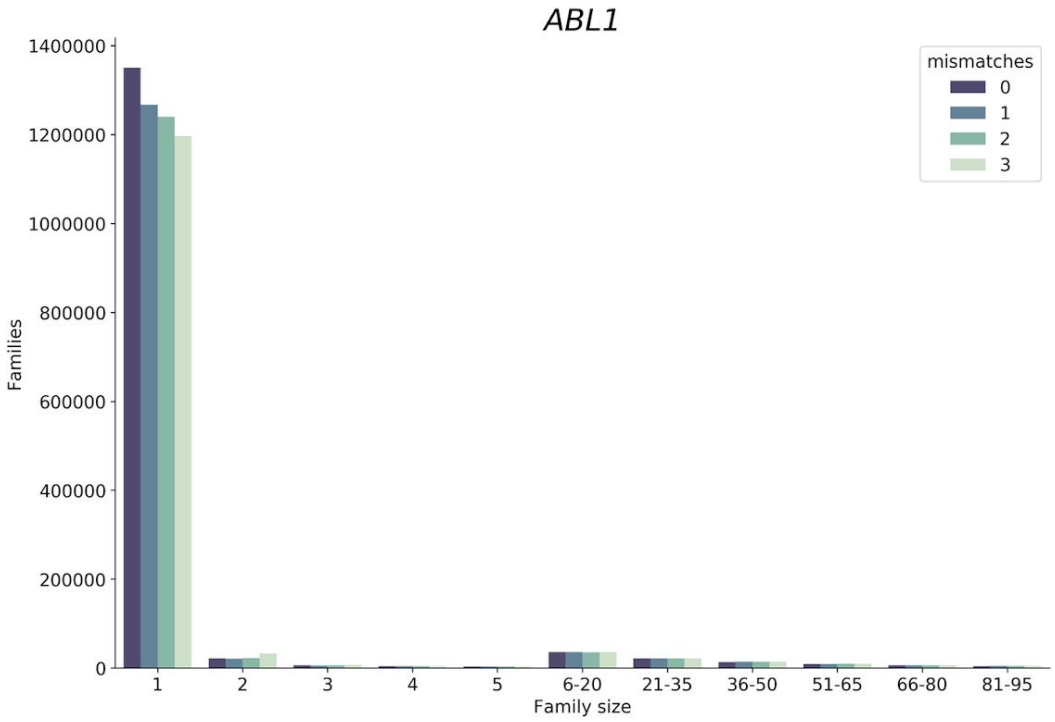


Figure 3.3 Distribution of SSCS family sizes with and without error correction. Single stranded consensus sequences (SSCS) are created when reads with identical barcodes are bundled together. A common practice requires at least three reads with identical barcodes to form a SSCS. Without error correction (0 mismatches) there is a striking abundance of singletons: single reads with a barcode that is different from all other barcodes in the sample. Applying error correction with progressively higher number of allowed mismatches (from 1 to 3) significantly decreases the number of singletons by re-uniting them with other reads.

3.3.3 Barcode error correction increases yield

Forming families of reads descended from the same original fragment requires grouping reads by barcode (Fig. 3.1 and 3.3). This is straightforward when no sequencing errors are present and can be done by simple lexicographic sorting. Yet as we have shown in the previous section, errors are widespread and this eliminates sorting as a legitimate analysis strategy. An alternative approach will involve performing all-versus-all comparison of all barcodes to identify those that differ by just a few nucleotides and further checking them to see if they are potentially derived from the same DNA fragment with differences being introduced by PCR or sequencing errors. The challenge is that the all-versus-all comparison has $O(n^2)$ time complexity and thus is prohibitive as a routine analysis strategy. There are several tools that approach this problem in different ways. The most common strategy is to reduce the search space by first aligning the raw reads to the reference genome. One can then consider barcodes of only those reads that align to one region of the reference. This does not change the time complexity of the search, but reduces the search space from the millions of barcodes in the entire sample to the dozens that may be aligned to a particular genomic location. Several tools are available which use this strategy (T. Smith, Heger, and Sudbery 2017; Xu et al. 2018; Fennell and Homer 2018). However, reference-based approaches are inevitably biased and it was our main impetus to avoid the use of a reference sequence (Stoler et al. 2016). Alternatively, a strategy implemented in MAGERI, a tool which does not require a reference sequence to form consensus sequences, is able to perform efficient barcode error correction with the use of a custom seed-and-extend alignment algorithm (Shugay et al. 2017, 2014). However, it only

forms single-strand consensus sequences, not the duplex consensus sequences required in our analysis.

To overcome these limitations we have adopted Burrows-Wheeler *k*-mer indexing implemented in Bowtie (Langmead et al. 2009) to quickly perform all-versus-all comparison of duplex tags. We are using the original version of Bowtie (not Bowtie2) that was optimized for very short reads. Specifically, we create an FM-index for all barcodes in the sample and then align individual barcodes (as if they were reads) against that index. Results of this alignment are represented as a graph where each vertex corresponds to a barcode. An edge is drawn between two vertices if an alignment exists between two barcodes. An alignment should have a user-defined minimum mapping quality and maximum edit distance, with default values set to 20 and 1, respectively (in the discussion below we vary edit distance values from 1 to 3). The resulting graph contains a large number of disconnected clusters, each of which theoretically represents a single barcode together with all its derivatives created due to PCR and sequencing errors. A correct barcode can therefore be chosen by picking the vertex whose barcode tags the highest number of reads. To assess the effectiveness of this error correction strategy we have developed a tool for producing simulated DS data (see Methods). Using this simulator we produced 400,000 duplex reads and analyzed them using our error correction approach. We then proceeded to calculate how many families (and thus, DCSs) were added to the analysis because of the correction. This increase in yield—the most important consequence of error correction— was substantial. The 400,000 simulated duplex reads produced 43,344 DCSs without correction. Running error correction by setting edit distance to one, two, or three mismatches resulted in 52,896, 53,420, and 53,454 DCSs, respectively. This constituted a 23% increase in yield (at three mismatches) compared to an uncorrected analysis. Effectively the error correction algorithm “shrinks” the pool of singletons (family size, FS, of 1) by reuniting them with families containing correct barcodes, increasing the likelihood that a group of reads surpasses the minimal member number (family size [FS] $\geq$ 3) for calling a SSCS.

Next, we proceeded to test our approach on real duplex sequencing datasets we used above. We specifically explored if tag error correction improves the number of consensus bases in SSCS and DCS, when allowing for 1, 2, or 3 mismatches in the tags. The results of error

correction are summarized in Table 3.1, Figs. 3.2 and 3.3. The error correction decreased the number of singletons (FS $\geq$ 3) while increasing the numbers of DCSs by re-incorporating singletons into duplex families. This was particularly striking in pBR322 dataset, where the number of DSC increased from 77,164 to 89,513 (Table 3.1). One can also see that increasing the edit distance during error correction to 2 or 3 did not have such a drastic effect in reducing the number of singletons and increasing the overall SSCS and DCS.

Sample	*ABL1*				pBR322			
# errors	0	1	2	3	0	1	2	3
SSCS ab	38,493	37,803	37,007	36,280	84,231	81,929	78,481	73,647
SSCS ba	38,202	37,496	36,772	36,080	84,085	81,741	78,234	73,160
DSC	20,745	21,299	22,151	23,180	77,164	80,640	84,359	89,513

Table 3.1 Effect of error correction on duplex datasets analysis as the number of single strand consensus sequences (SSCS) and duplex consensus sequences (DCS) called after no error correction (0 errors) and error correction with three thresholds of 1, 2, and 3 mismatches allowed.

3.3.4 Du Novo corrects most barcode errors

With the simulated dataset, we were able to examine the accuracy of barcode correction. Using Du Novo, we corrected the simulated data using three different thresholds for edit distance. Because the dataset was simulated, we were able to compare the corrections made by Du Novo to the ideal set of corrections (Table 3.2). We can classify these into true and false positives and negatives by considering a correction to be a “positive” and an omitted correction a “negative”. When setting a stringent edit distance threshold of 1, erroneous corrections (false positives) occurred only five times (out of 816,335 corrections). But the tradeoff was that Du Novo was only able to catch and correct 67.28% of barcodes containing errors. Setting a more aggressive edit distance threshold of 3 allowed 77.97% of erroneous

barcodes to be corrected, but this caused 2,740 barcodes to be wrongly corrected (0.29% of 936,582 corrected). So even with the most conservative threshold, over two thirds of erroneous barcodes are rejoined with their true families, and only 1 in 163,267 of these newly formed families are artifactual.

Edit distance		Positive (%)	Negative (%)
1	True	99.999	67.281
	False	0.001	32.721
2	True	99.983	74.971
	False	0.017	25.033
3	True	99.710	77.972
	False	0.290	22.033

Table 3.2 Robustness of barcode error correction, measured through simulated data. The barcodes of 400,000 simulated duplex reads were corrected with Du Novo 2.15 with three different edit distance thresholds: 1, 2, and 3. Corrected and uncorrected barcodes were compared to the original, true barcode sequences. For each corrected barcode, if the correction assigned it to its true family, this was counted as a true positive. Otherwise it was a false positive. Uncorrected barcodes which were not one of the original, true barcodes were counted as false negatives. The rest of the uncorrected barcodes were counted as true negatives. Each family was counted once, rather than each raw read.

3.3.5 New alignment engine improves consensus generation

The first version of Du Novo had a number of limitations resulting in poor performance. It was taking close to 24 hours to analyze a single duplex experiment. There were two primary reasons for this: the use of MAFFT aligner and inadequate parallelization strategy for executing multiple consensus generating jobs.

First, we sought to increase the performance of consensus generation step by employing a different multiple alignment tool that can be integrated into Du Novo codebase. We evaluated two candidate tools: SeqAn (https://github.com/seqan/seqan) and Kalign2 (Lassmann, Frings, and Sonnhammer 2009). SeqAn is a library of algorithms, including a multiple sequence aligner, specifically written to be incorporated into other genomics tools. Written in C++, it can be compiled and its functions called from Python. Kalign2 is an aligner using the Wu-Manber approximate string-matching algorithm (Wu and Manber 1992) to significantly speed up alignment while maintaining accuracy. Kalign2 is written in C and can also be compiled and called from Python. With some modification, it is possible to communicate with its functions directly from Python, without temporary files. This allows the greatest efficiency and greatest integration into a Python process. SeqAn and Kalign2 were evaluated against MAFFT, the existing algorithm in use by Du Novo. The aligners were tested by performing a multiple sequence alignment on a duplex read family extracted from a duplex experiment sequencing the whole human mitochondrial genome (Stoler et al. 2016). The family contained 74 reads, 41 single nucleotide substitutions relative to the consensus, and no indels. The number of reads in the alignment was varied from 1 to 74, and the time taken to perform the alignment was measured. Fig. 3.4 shows the results of this experiment. SeqAn was the slowest at all alignment sizes, with the worst performance at handling of large alignments. It took 58× more time than MAFFT at 10 reads, and 427× more at 40 reads. The fastest for all sizes was Kalign2. At 10 reads, it took less than 10 milliseconds. At 30 reads it was 9× faster, but at 60 reads it was only 4× faster than MAFFT. Since the median family size for an ideal duplex experiment is only around a dozen reads, Kalign2's advantage is significant and we chose it as the default alignment engine for Du Novo.

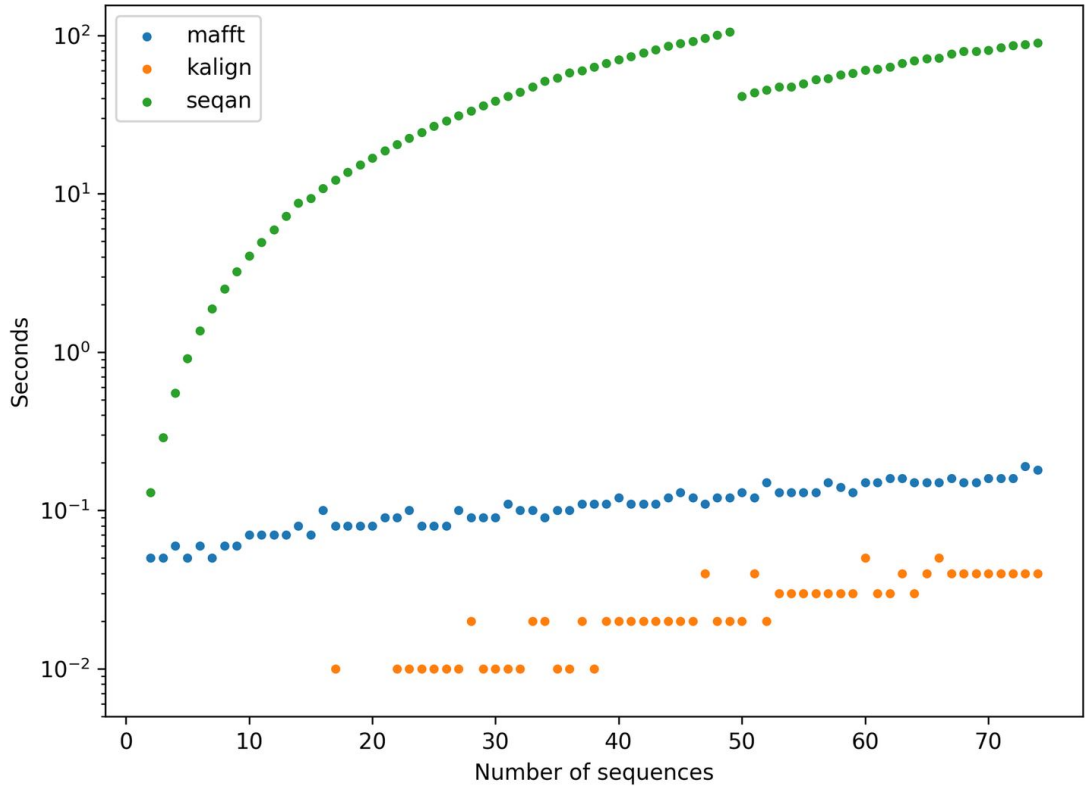


Figure 3.4 Alignment engine comparison. Comparison of the three aligners tested for use in the Du Novo pipeline.

3.3.6 Smarter parallelization improves speed

Du Novo uses the multiprocessing Python module for parallel processing. In order to maintain the ordering of the aligned families, the old algorithm would start *N* alignment jobs in parallel, then check each job in order for results. This created a bottleneck at the slowest job: the *N* jobs would take as long as the slowest one. The new algorithm maintains a queue of jobs executing or waiting to be executed. In order to maintain ordering the algorithm keeps an ordered list of submitted jobs. It fills the queue, then begins processing outputs in order of submission. This still requires waiting for all jobs in a batch to finish before continuing, but to reduce the bottleneck, the queue is made larger than the number of available workers. As soon as a job finishes and a worker is freed, it begins work on a new job. This lets new jobs run on CPUs freed by the fastest jobs while the slowest job is still running. If *W* is the number of workers and *M* is a multiplier such that *M*×*W* is the queue size, then a single batch of *M*×*W* jobs will take less time than *M* batches of *W* jobs. Diminishing returns occur as *M* grows, so *M* is set

to eight by default. To show the combined effect of the change in alignment and queueing algorithms, Du Novo 2.15, using Kalign2 and the default queue size was compared with Du Novo 0.4, using MAFFT and the old queueing algorithm. Table 3.3 shows that the combination of the two changes results in an over 9× faster performance at low levels of parallelization. The trend in memory usage is the same as when comparing Kalign2 vs. MAFFT.

version	aligner	time/ memory	CPUs					
			1	2	4	8	16	32
0.4	MAFFT	time (seconds)	28,638	15,769	8,912	5,173	3,038	1,747
2.15	MAFFT		28,754	14,282	7,079	3,463	1,686	854
	Kalign2		4,731	1,777	945	600	381	246
0.4	MAFFT	memory (MB)	23,704	12,299	6,622	3,755	2,284	1,602
2.15	MAFFT		23,927	12,599	6,850	3,985	2,541	1,810
	Kalign2		24,648	23,220	12,408	6,668	3,781	2,327

Table 3.3 Time and memory usage of different versions of **`align-families.py`**, using different multiple sequence alignment algorithms. At low levels of parallelization, Kalign2 made the process over 8 times faster, with a memory usage less than twice as much as MAFFT. The new algorithm sped up the tool between 1 and 2.05x. Naturally, at higher levels of parallelization, the reduction of the job queue bottleneck made more of a difference. Memory usage appeared to not be affected, which is expected due to the small size of the job queue compared with the rest of memory usage. To attempt to disentangle the effects of the job queueing algorithm from all the other changes between 0.4 and 2.15, the two versions were compared with all parameters set as similarly as possible. In both cases, the number of **`--processes`** was set to 32 and MAFFT was used as the aligner. Crucially, the **`--queue-size`** for the 2.15 version was set to be 32, the same as the number of **`--processes`**. This approximates the bottleneck in the pre-2.0 version of Du Novo's job queueing algorithm. Comparing the median of 3 trials of each, the wallclock time of 2.15 was 27% higher than that of 0.4. This could be

because of the higher overhead in the more complicated parallelization algorithm, or other changes between 0.4 and 2.15.

Next we used the simulated dataset to test whether the change in alignment algorithm affects the accuracy of the pipeline. The simulated experiment was the same, but with 40,000 fragments generated instead of 400,000. Because the input was one homogenous sequence with no minor variants, any differences from the input must be due to incorrect consensus base calls. Using the previous multiple sequence aligner, MAFFT, resulted in an error rate of 0.00563 differences per output base (Table 3.4). Using Kalign2 instead resulted in 0.00561 differences. Adding barcode error correction improved this figure slightly to 0.00525 while also increasing the yield. The standard pipeline published by Loeb et al. (M. W. Schmitt et al. 2012) was also compared, resulting in 0.0114 differences per output base.

Method	Aligner	Barcode error correction	Errors per base
Du Novo	MAFFT	Uncorrected	0.563%
Du Novo	Kalign2	Uncorrected	0.561%
Du Novo	Kalign2	Corrected	0.525%
Loeb	N/A	Uncorrected	1.140%

Table 3.4 Effect of aligner on “correctness”.

3.4 Conclusions

In this manuscript we have introduced an error correction approach to the analysis of duplex sequencing data. This allows correcting errors in barcodes thereby reducing data loss and increasing the yield of duplex sequencing experiments. We made a number of other improvements including a new alignment engine and advanced parallelization. Finally, we made the new software readily available to a wide audience of users. To achieve this goal we are distributing Du Novo in three complementary ways:

- Interactive pipeline at http://usegalaxy.org. Here users can upload datasets of any size and process using the complete Du Novo pipeline to produce SSCS and DCS sequences. The Galaxy system contains all tools for downstream processing including mapping and variant calling. To help users effectively use our system we have developed a detailed step-by-step tutorial that can be found here: http://bit.ly/dunovo-tutorial.
- Bioconda package. Du Novo code relies on a number of software components that need to be installed before the tool can be used. Conda package eliminated the need to install these dependencies by automatically installing all components using **`conda install dunovo`** command (see http://bit.ly/dunovo-bioconda).
- Source code for the package can be found in GitHub at https://github.com/galaxyproject/dunovo. It is distributed under Academic Free License.

3.5 Methods

3.5.1 Barcode error analysis

The edit distance quantifies similarity or dissimilarity between two DNA sequences of equal length by calculating the number of differences between them:

$$D_{i,j} = \sum_{k=1}^{n} [X_{ik} \neq X_{jk}]$$

$D_{i,j}$ is the number of sites where X_i and X_j do not match, *k* is the index of the respective site out of a total number of sites *n*. The input data was in tabular format organized into family size, the sequence of the tag, and the direction of the tag in the SSCS (*ab* = forward or *ba* = reverse). Each tag represents a family of paired-end sequences forming SSCS. Since the whole dataset contained more than one million tags, the comparison of all tags was computationally too demanding. Thus, we parallelized the algorithm and only selected 1,000 random tags from the data set and compared them to the whole dataset to estimate the minimum edit distance between tags. A sample of 1,000 tags gives a very similar estimate to the edit distance estimated for a sample of 10,000 and ~130,000 tags.

3.5.2 Error correction

The script **`baralign.sh`** performs an alignment of all barcodes against themselves (all scripts mentioned in this section can be found at https://github.com/galaxyproject/dunovo). First, it extracts all unique barcode sequences (concatenations of *a*+*b* tags) as FASTA sequences. Then it indexes them, along with their reversed (*b*+*a*) versions, with bowtie-build and aligns them to the index with **`bowtie -v 3 --best -a`**. This alignment is then read by **`correct.py`**, which uses the networkx module to construct graphs where each vertex is a barcode and each edge is a high-quality alignment between two barcodes. The definition of a high-quality alignment is configurable and based on the MAPQ mapping quality, the edit distance given by the NM tag, and the distance between the aligned starting positions of the two barcodes. The default values for these filters is 20, 1, and 2, respectively. Then, for each graph, a "correct" barcode is chosen by one of two methods. The default method is to choose the barcode which tags the largest number of reads. An alternative is to choose the barcode with the most edges to other barcodes.

3.5.3 Generating simulated duplex data

To validate the effectiveness of our approach we have first applied it to a simulated duplex sequencing dataset generated with a duplex sequencing simulator developed to test the correctness of the Du Novo algorithms against known duplex sequencing behavior and sources of errors. It simulates an entire duplex sequencing experiment but taking a reference genome sequence as input, randomly fragmenting it, adding random barcodes to the ends of these fragments, simulating PCR and sequencing errors to produce a set of simulated duplex reads. To randomly fragment the reference sequence, it uses **`wgsim`** (https://github.com/lh3/wgsim) in error-free mode (options **`-e 0 -r 0 -d 0`**), using the **`-1`** option to set the length of the fragments. Then it simulates random oligomer synthesis to produce duplex barcodes using a uniform 25% probability for each base. It concatenates these oligomers, along with a constant linker sequence, with the fragment sequence to produce starting fragments. These simulated tagged fragments then undergo *in silico* PCR in order to introduce amplification errors. First, a family size is chosen from an empirical distribution observed in a previous duplex experiment.

Then, the phylogenetic tree relating these reads is generated. For a family size of *n*, the process starts with *n* reads at the root node representing the original fragment molecule. Each read is randomly assigned to a daughter molecule with 50% probability. Then the process repeats with each daughter, using the number of reads assigned to the daughter instead of *n*. Because amplification efficiencies decline as PCR cycles continue, the probability of replication starts at 1 and is divided by 1.05 each cycle, a realistic value compared with observed reactions (Larionov, Krause, and Miller 2005). Once a tree is generated, errors are simulated at each node and propagated to their descendents. Then, sequencing is simulated, also with errors, and reads are output. A log of the errors is also saved, in order to allow checking results against the "truth". Unless noted otherwise, simulated data presented here were generated with a sequencing and PCR polymerase error rate of 0.001 errors per base. 25 cycles of PCR were simulated, the fragment lengths were set to 400bp, and the read lengths to 100bp. Using this approach we have generated a dataset containing 400,000 simulated duplex reads and applied our error correction strategy.

3.5.4 Du Novo 2.0

The basic algorithms in Du Novo 2.15 remain as described in Stoler *et al.* 2016, except the addition of barcode error correction, the Kalign2 multiple sequence aligner, and the replacement of the parallel job queueing algorithm. In all experiments described here, the threshold required to form a consensus base (`make-consensi.py`'s `--cons-thres`) was 0.7, 3 reads were required to create a consensus sequence (`--min-reads`), and a PHRED score of at least 25 was required to count a base toward the consensus (`--qual`).

When consensus reads were filtered, the script `trimmer.py` was used from the `bfx` directory of Du Novo's distribution. Unless noted otherwise, the script was set to remove the 5' end of reads when the proportion of N's in a 10 base window exceeded 0.3 (`--filt-base N --window 10 --thres 0.3`). If either of the reads in a pair was trimmed to less than 75 bases, both were removed (`--min-length 75`).

3.6 Declarations

3.6.1 Abbreviations

DCS: Duplex Consensus Sequencing; DS: Duplex Sequencing; PCR: Polymerase Chain Reaction; SSCS: Single Stranded Consensus Sequence

3.6.2 Acknowledgements

We would like to thank Michael Schmitt for advice on the *ABL1* dataset, Mikhail Shugay for clarification of MAGERI usage, Wilfried Guiblet for advice on duplex sequencing, and David Bouvier and Tejaswini Mishra for help in the conceptualization stages of Du Novo.

3.6.3 Authors' contributions

NS developed software and performed analyses, BA participated in analysis and advised on methodological aspects of duplex sequencing, GP, MH, RS developed quality control utilities and tested software components. KM, IT, and AN conceived the project and provided funding, AN wrote the manuscript.

3.6.4 Funding

This study has been funded by the funds provided by the Eberly College of Science at the Pennsylvania State University and NIH Grants U41 HG006620 and R01 AI134384-01 as well as NSF ABI Grant 1661497. Funding for R.S., M.H., G.P., and I.T-B was provided by the Linz Institute of Technology (LIT213201001) and the Austrian Science Fund (FWFP30867000).Funding bodies did not participate in the collection, analysis and interpretation of data or writing the manuscript.

3.6.5 Availability of data and material

All software is freely available under a non-restrictive AFL 2.0 license. It can be accessed via Galaxy system at https://usegalaxy.org or downloaded from GitHub at https://github.com/galaxyproject/dunovo.

3.6.6 Ethics approval and consent to participate

Not applicable.

3.6.7 Consent for publication

Not applicable.

3.6.8 Competing interests

The authors declare that they have no competing interests.

4 Sequencing error profiles of Illumina sequencing instruments

This chapter is reproduced from a manuscript by Nicholas Stoler and Anton Nekrutenko, in review with *NAR: Genomics and Bioinformatics*.

4.1 Abstract

Sequencing technology has achieved great advances in the past decade. Studies have previously shown the quality of specific instruments in controlled conditions. Here we developed a method able to retroactively determine the error rate of most public sequencing datasets. To do this, we utilized the overlaps between reads that are a feature of many sequencing libraries. With this method, we surveyed 1,943 different datasets from seven different sequencing instruments produced by Illumina. We show that among public datasets, the more expensive platforms like HiSeq and NovaSeq have a lower error rate and less variation. But we also discovered that there is great variation within each platform, with the accuracy of a sequencing experiment depending greatly on the experimenter. We show the importance of sequence context, especially the phenomenon where preceding bases bias the following bases toward the same identity. We also show the difference in patterns of sequence bias between instruments. Contrary to expectations based on the underlying chemistry, HiSeq X Ten and NovaSeq 6000 share notable exceptions to the preceding-base bias. Our results demonstrate the importance of the specific circumstances of every sequencing experiment, and the importance of evaluating the quality of each one.

4.2 Introduction

Assessing the accuracy of next-generation sequencing has been the focus of much study since these techniques emerged. In 2011, studies on the Illumina Genome Analyzer (GA) and GA IIx discovered an association between errors and certain sequence motifs leading up to the error site (Nakamura et al. 2011; Meacham et al. 2011). One of the studies also produced a

profile of substitution biases, including a strong preference for T-to-G substitutions (Meacham et al. 2011). This study made use of a phenomenon where mates in paired-end sequencing experiments “overlap”. In this situation, the ends of the two reads cover the same portion of their source fragment. This allows sequencing errors in one read to be revealed by the other.

In the past few years, many new sequencing instruments have been introduced. For instance, Illumina has introduced the HiSeq X Ten, with patterned flowcells, NextSeq 500, with 2-dye chemistry, and NovaSeq 6000, combining both in an industrial-scale platform (van Pelt-Verkuil, van Leeuwen, and te Witt 2019). While the basic reversible chain-terminator principle remains unchanged, these are significant modifications which could be expected to introduce their own biases. For instance, labeling nucleotides with only two fluorophores means that guanine is detected by the absence of signal (van Pelt-Verkuil, van Leeuwen, and te Witt 2019). Some have reported that this results in overcalling of G’s when artifacts cause signal dropout (Andrews 2016). On the other hand, a controlled study compared HiSeq 2500 and NovaSeq 6000 and indicated a lower error rate in the NovaSeq (Ma et al. 2019). Evidently, these new technologies beg examination to determine their effects on sequencing errors.

Comparing the error rates of sequencing platforms has been a focus of research since sequencing began. Every new platform has its advantages and disadvantages, with its error rate being one of the most important factors. Typically the error rate is assessed by comparison of results across different platforms with multiple replicates (Ross et al. 2013). This is the gold standard for showing how the different technologies operate in the same hands. These studies are useful when one is deciding on an instrument to use. But different groups see different outcomes with the same technology. Even within the same group, there is often variation from experiment to experiment (Melanie Schirmer et al. 2016). And there may be a difference between error rates observed in an ideal scenario vs. typical use “in the wild”. So, knowing the extent of this variation is important for consumers of sequencing data produced by others. Even researchers choosing technologies for their own data may find it useful to know how much their mileage may vary.

But measuring error is a theoretically difficult task. Some have taken a simple approach, aligning reads to a reference and calling variants as errors (Ross et al. 2013). But real variants

will then be misclassified as errors as well. Instead, one could first perform variant calling, assuming the majority allele at any position is correct, and any minor alleles are errors. This will work well for samples which are known to be highly homogeneous, but otherwise there may be true minor alleles which would be mistaken for errors (Fox and Reid-Bayliss 2014). This can be the case for samples of microorganisms, viruses, cancers, or organelles. It is difficult to automatically ascertain how homogenous a sample is, making it a hurdle for an automated survey. Also, at sites with low numbers of reads, it is possible that the error base randomly occurs more often than the true sample base, causing artifacts in error detection. Another issue with both of these approaches is that they detect errors from more than just sequencing. Library preparation steps like PCR can also introduce errors. And different preparation techniques can introduce different numbers and types of errors. Both of the error detection methods above will identify both library preparation errors and sequencing errors combined.

In a paired-end experiment, when a fragment is smaller than the length of both reads combined, the ends of the reads will overlap. This means that, in this overlapping region, the same DNA fragment is assayed twice. Both reads share this same exact molecule as an ancestor, and the only source of errors in-between is from the sequencing instrument. Any PCR errors, cloning polymorphisms, DNA damage, or other library preparation errors that have occurred have already been introduced into the fragment (Fig. 4.1), and will not produce a difference between the two reads (Ross et al. 2013). This provides a powerful method of assaying the sequencing error introduced by an instrument in any paired-end dataset with sufficient overlap.

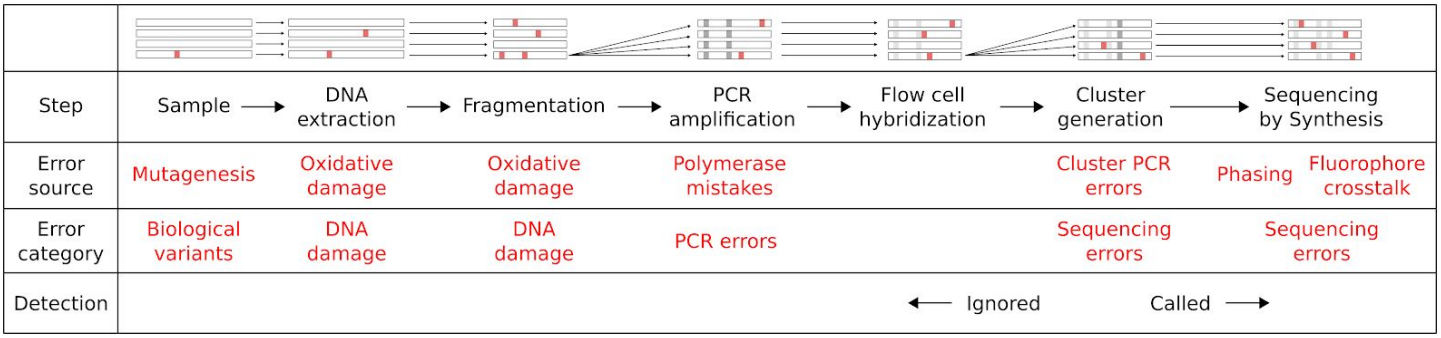


Figure 4.1 The steps involved in sequencing a biological sample, and where polymorphisms can arise. This shows a typical procedure to extract DNA from a sample, prepare a sequencing library, and sequence it. Different types of variants are labeled at the point where they can be introduced. The green dotted line separates variants which are ignored by overlap analysis from ones which are detected and considered "sequencing errors". At the top is a visualization of what happens to the DNA in the sample at each step. Arrows indicate the lineage of DNA molecules, either where they are the same actual molecule (horizontal arrows), or copies of the same ancestor (angled arrows). The visualization focuses on the ancestor molecules of one Illumina flow cell cluster (right). So at PCR amplification and cluster generation, we "zoom in" to focus on copies of the single molecule which is an ancestor. Red spots indicate variants which have been introduced, and gray spots indicate where a variant from a previous step becomes fixed in the molecules being shown. These gray variants are no longer polymorphic among the ancestors of that single, final cluster. These fixed variants are ignored by our error detection method.

Armed with a method that can be retroactively applied to a large portion of existing datasets, we can then perform a large-scale survey of real-world sequencing experiments. The Sequence Read Archive (SRA) hosts the largest public database of next-generation sequencing data (Kodama et al. 2012). The SRA provides metadata which can allow automatically filtering for qualifying datasets and categorizing them by sequencing platform. In order to enforce uniformity, we decided to focus only on one organism: *E. coli*. We chose *E. coli* because of its relatively compact genome, making sequence alignment simpler. It is one of the most studied prokaryotes, with a large number of publicly available datasets. Also, in order to focus on new Illumina technologies, we scoped our survey to only this manufacturer.

4.3 Methods

4.3.1 Obtaining SRA datasets

We selected *E. coli* datasets from the SRA using the Entrez Direct utilities from NCBI (Kans 2013). Using the query "Escherichia coli"[Organism], we fetched the metadata for all 186,022 matching runs as of 31 August 2020. 179,306 of these were by Illumina instruments, with 75,118 MiSeq, 36,034 HiSeq 2500, and 1,375 NovaSeq 6000. A full breakdown is given in Table 4.S1. We then filtered out single-ended and non-Illumina datasets, ordered the list to prefer a diversity of sequencing platforms and submitting groups, and prioritized runs that were the most likely to have the most read overlap. We then downloaded the FASTQ files using the SRA toolkit (version 2.10.0) or EBI's FTP server.

4.3.2 Determining the best reference

In order to determine the best reference sequence for read alignment, we performed a "meta-alignment" where we combined all complete *E. coli* genomes into one reference and aligned the sample reads to it. We used the NCBI Genome database to gather a list of complete genomes. Specifically, we downloaded the table available at ftp.ncbi.nlm.nih.gov/genomes/refseq/bacteria/Escherichia_coli/assembly_summary.txt and selected all assemblies with an assembly level of "Complete Genome". This was 1,292 assemblies as of 31 August 2020. We downloaded the FASTA files from the ftp_path in that table and concatenated them into a single meta-reference. For each sample, we aligned its reads to the meta-reference with BWA-MEM (Heng Li 2013) (with the `-M` flag; version 0.7.17-r1188). Then we read the alignments with samtools (H. Li et al. 2009) to count how many alignments were made to each reference. We only counted primary alignments (SAM flag 256) which were mapped (flag 4), passed instrument QC (flag 512), were not PCR/optical duplicates (flag 1024), and not supplementary (flag 2048). Then we chose the reference with the most alignments, excluding references smaller than 2Mb.

4.3.3 Detecting overlap errors

Detecting errors began by aligning the reads of each run to the chosen reference. Alignment was performed with bwa mem -M, as in the previous section. Reads were removed if they did not belong to a pair where both reads were mapped (had SAM flags 1 and 2, and not 4 or 8). They were also removed if they were secondary alignments (flag 256), failed instrument QC (flag 512), were PCR/optical duplicates (flag 1024), or were supplementary alignments (flag 2048). Then, errors were detected by pairing read bases by their reference coordinate, reporting mismatching bases as errors.

4.3.4 Calculating error rates

We calculated the error rate of each sample by dividing the number of detected errors by the amount of overlap between read pairs. The calculation excluded errors where one base was N. We also broke down the error rate for each sample by regions of the reads. For each error, we determined where in both reads it occurred. Then we divided each read into bins 1/10th of the read length long. We defined the bin of the error to be the bin of the read length it occurred in. If it was in different bins in the two reads, we took the greatest of the two (the bin furthest toward the 3'-end of the read). This is because the main purpose of binning is to reduce the effect of errors increasing toward the end of reads. If an error appeared in bin 2 of one read, and bin 10 of its mate, it's more likely to be from a sequencing error in the latter read, occurring due to how close to the end of the read it is. After the binning process, we calculated the error rate separately for each bin. For each sample, we required at least 2.5 million overlapping bases in a bin to calculate a valid error rate for it.

4.3.5 Correlating error rates with platform and lab

We selected the samples with valid error rates calculated in the previous section using the rates as our dependent variable. Our independent variables were derived from the "model", "center", "lab", and "contact" metadata fields. The latter three were combined and each combination was deemed a separate "group". In order to reduce the number of categories, we combined all groups which appeared less than four times into an "other" group. We then performed ordinary least squares regression with the model and group as the

independent variables and the error rate as the response variable. The regression was performed by the statsmodels.formula.api.ols function from the statsmodels Python package.

4.3.6 Tabulating base frequencies in error sequence contexts

For every genomic location where we detected an error, we extracted the 20bp of genomic sequence centered on the error site. For each substitution in each platform, we counted the total count of each base at each distance from the error. We determined the substitution by first examining all the read bases at the error site. We chose the most common base as the most likely major allele in the sample. Then, for each error, we assumed the base that did not match the major allele was likely the erroneous base. If neither read base matched the major allele, we did not call the substitution or include it in the analysis. Once all the substitutions were called, we chose the most common one at that site.

4.3.7 Calculating trimer frequencies

For every genomic location with a detected error, we examined the three genomic bases leading up to, and including the error site. For every platform, we counted how many trimers of each type were present in the set of unique error loci. We converted the counts to frequencies by dividing by the total number of trimers. We then normalized the frequencies by the prevalence of each trimer in the genome. We chose the most common reference sequence in our samples, NZ_CP044311.1 from strain RM13752. We counted the number of each trimer in that sequence, then converted to frequencies. We then divided each trimer's error frequency by its genomic frequency.

4.3.8 Counting post-homopolymer errors

A particular error pattern has been observed in Illumina in regions with homopolymer runs. After a homopolymer of a particular base, the base immediately following the homopolymer will often be subject to a substitution where the error base is the same as the homopolymer base (Nakamura et al. 2011). Here, we refer to this error type as a "post-homopolymer error".

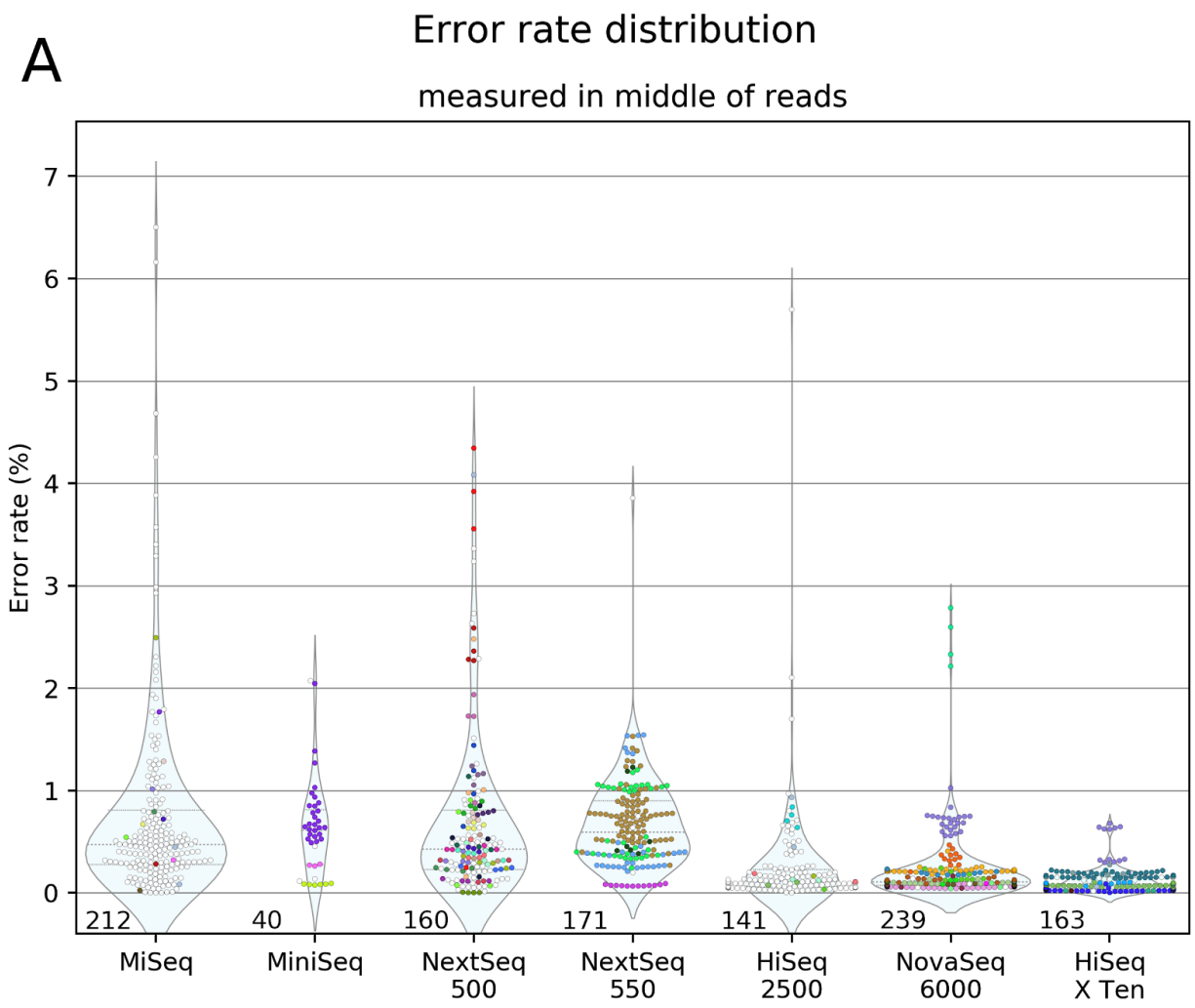
We examined the sequence context surrounding every detected error, counting errors which matched the post-homopolymer pattern. This included every error where the substituted base matched the previous genomic base. We assumed the substituted base was the base that did not match the genomic (reference) base at that site. In the raw data, we included the identity of the preceding/error base and how many times it was repeated. For Fig. 4.6, we normalized the error counts by the frequency of homopolymers in the genome. To do this, we analyzed the same genome as for the trimers, NZ_CP044311.1, counting the number of homopolymers of each base type and length. We then determined the number of post-homopolymer errors of each type we would expect at random. To do this, we first calculated the per-base frequency of each homopolymer type in the genome: the number of homopolymers of that base and length divided by the length of the genome. To get the expected number of random post-homopolymer errors, we multiplied this frequency by the total number of errors detected, and divided by four.

4.4 Results and Discussion

4.4.1 Distribution of error rates for each platform

Fig. 4.2 shows the range of error rates in samples from different platforms. For each read pair, errors were only chosen from the region between 50% and 60% of the full read length. This controls for the variation in overlap between samples. Error rate is highly correlated with the sequencing cycle, rising toward the end of each read. In samples with smaller overlaps, the detected errors will tend to be later in the reads than in samples with larger overlaps. To reduce this undesired influence, we selected errors from similar regions in the reads.

A


Error rate distribution
measured in middle of reads
Error rate (%)
0
1
2
3
4
5
6
7
212
40
160
171
141
239
163
MiSeq
MiniSeq
NextSeq 500
NextSeq 550
HiSeq 2500
NovaSeq 6000
HiSeq X Ten

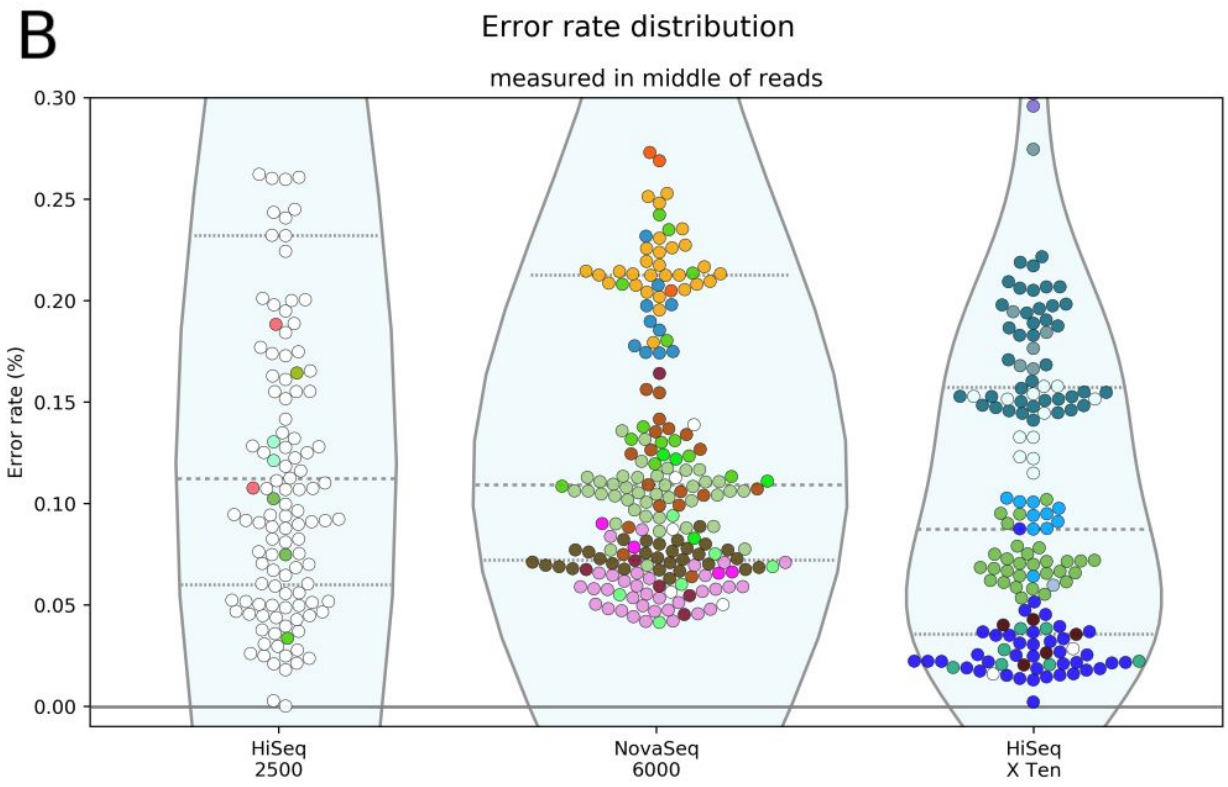


Figure 4.2 Error rates calculated from the overlap between read pairs. Errors were counted from the regions of reads 50-60% of the way through their lengths. Each SRA run is shown as one point. Only runs with sufficient overlap are shown: each must have a total of at least 2.5Mb in the 50-60% bin. The number of runs shown is displayed at the bottom of each distribution. A different color was given to each group with more than 3 samples in the survey. Groups with 3 or fewer are colored purple. Groups are defined by the combination of the center, lab, and contact metadata fields. Panel A displays all instruments in the survey with more than 10 passing runs. Panel B is a zoom on the low-error instruments, showing only runs with an error rate less than 0.3%.

The median error rate of each platform is shown in Table 4.1. These vary from 0.087% in HiSeq X Ten to 0.613% in MiniSeq. These figures are comparable to those determined in more controlled settings (May et al. 2015; Fox and Reid-Bayliss 2014). Perhaps even more striking is the variation within each platform. The error rates vary far more between samples than between platforms. Previous studies have shown small-scale indications of this phenomenon (Ma et al. 2019; Melanie Schirmer et al. 2016). Ma *et al*. shows that some portion of this

variation may come from oxidative damage introduced by differential sample handling (Ma et al. 2019).

Platform	Number of samples	Error rate (%)	
		Median	Standard deviation
MiSeq	212	0.473	0.938
MiniSeq	40	0.613	0.459
NextSeq 500	160	0.429	0.827
NextSeq 550	171	0.593	0.435
HiSeq 2500	141	0.112	0.544
NovaSeq 6000	239	0.109	0.350
HiSeq X Ten	163	0.087	0.126

Table 4.1 Summary statistics for the observed error rates of samples from each sequencing platform. The samples and error rates summarized are the same shown in Figure 4.2.

While the error rates may not be significantly different, their variation does depend greatly on the platform. The highest standard deviation is in MiSeq, at 0.938 percentage points. The lowest is in HiSeq X Ten, at 0.126. There seem to be two categories of platforms: one with higher variation and error rates, and one with lower variation and error rates. The instruments in the latter category are HiSeq 2500, HiSeq X Ten, and NovaSeq 6000 — the most expensive machines. One explanation could be that users of these machines spend more time optimizing their runs, since a low-quality run would be a much more expensive loss.

The HiSeq X Ten is easily the most consistent platform. Since this instrument is such an expensive installation, we asked whether the consistency is due to the datasets being dominated by one lab group. So we colored Fig. 4.2 by lab group, which shows that there are several groups which are represented much more than others, and the error rates do tend to be consistent within each group. A full list of lab groups is available in Table 4.S2. However,

there is substantial diversity overall. 11 different groups contributed to our HiSeq X Ten total, with nine contributing at least five samples. Another notable characteristic is the bimodal distribution of NovaSeq 6000 samples. This distribution seems to be mostly due to the fact that over two thirds of the NovaSeq samples are from one group whose samples cluster around 0.683%, rather than the 0.106% of the rest of the NovaSeq samples. While both NovaSeq figures are technically lower than that of the HiSeq 2500, their error rates are very similar, in contrast to the marked difference evident in Fig. S4 of (Ma et al. 2019). And in contrast to early reports that HiSeq X Ten had higher error rates than older HiSeq instruments (Heng Li 2014a), our survey shows the error rate of public HiSeq X Ten datasets is even lower than HiSeq 2500, and more consistent.

In order to investigate how much the error rate of each sequencing run depends on the platform vs. the group producing it, we performed linear regression on all the datasets. Fig. 4.S1 shows that the coefficients for research groups are on a similar or greater scale than that of the sequencing instruments. On the other hand, Fig. 4.S2 shows that just as many platforms as groups are significantly correlated with error rate, after Bonferroni correction. Fig. 4.2 appears to show clear group-specific patterns, and this test was able to show significant correlation in three of the groups. On the other hand, three sequencing platforms were also shown to correlate with accuracy. So this test does support the idea that despite great intra-platform variation, accuracy still does depend on the sequencing platform.

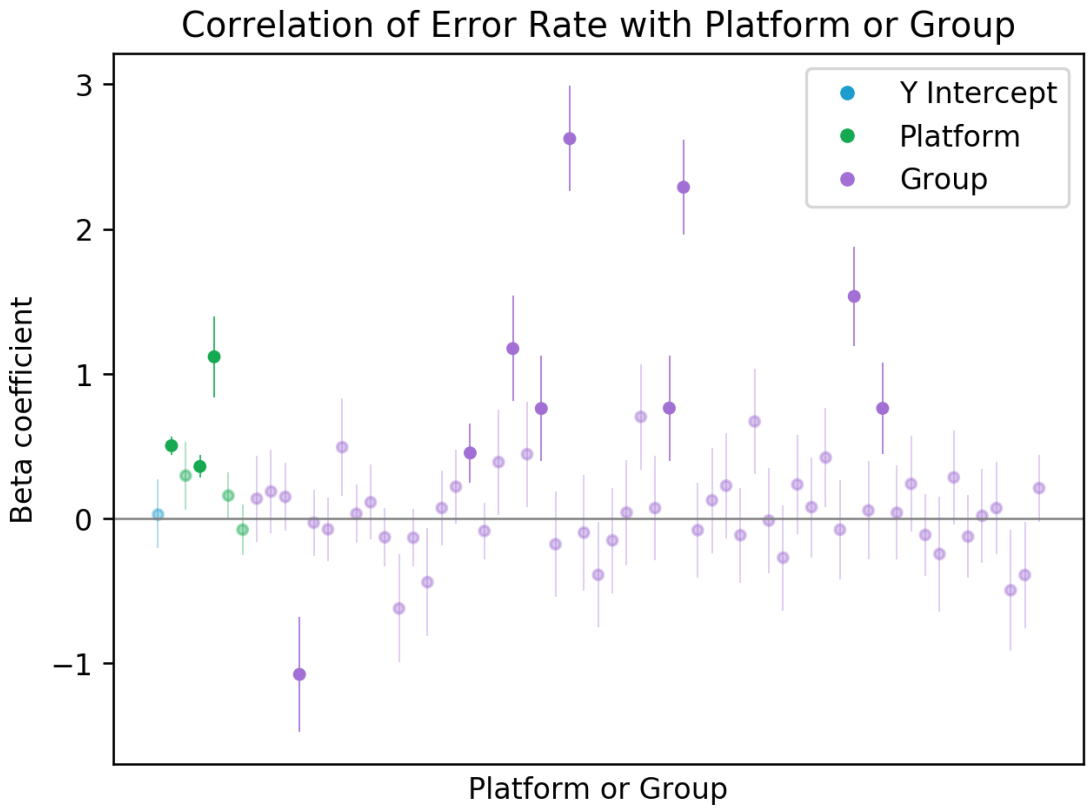


Figure 4.S1 Regression coefficients for the correlation of platform and group with error rate. Each point represents a particular sequencing platform or dataset producer. Only groups which appear at least 4 times in the survey are included. Error bars represent standard error, and coefficients with a p-value ≥ 0.05 are shown lighter than significant ones. The platforms, from left to right, are the MiSeq, MiniSeq, NextSeq 500, NextSeq 550, NovaSeq 6000, and HiSeq X Ten.

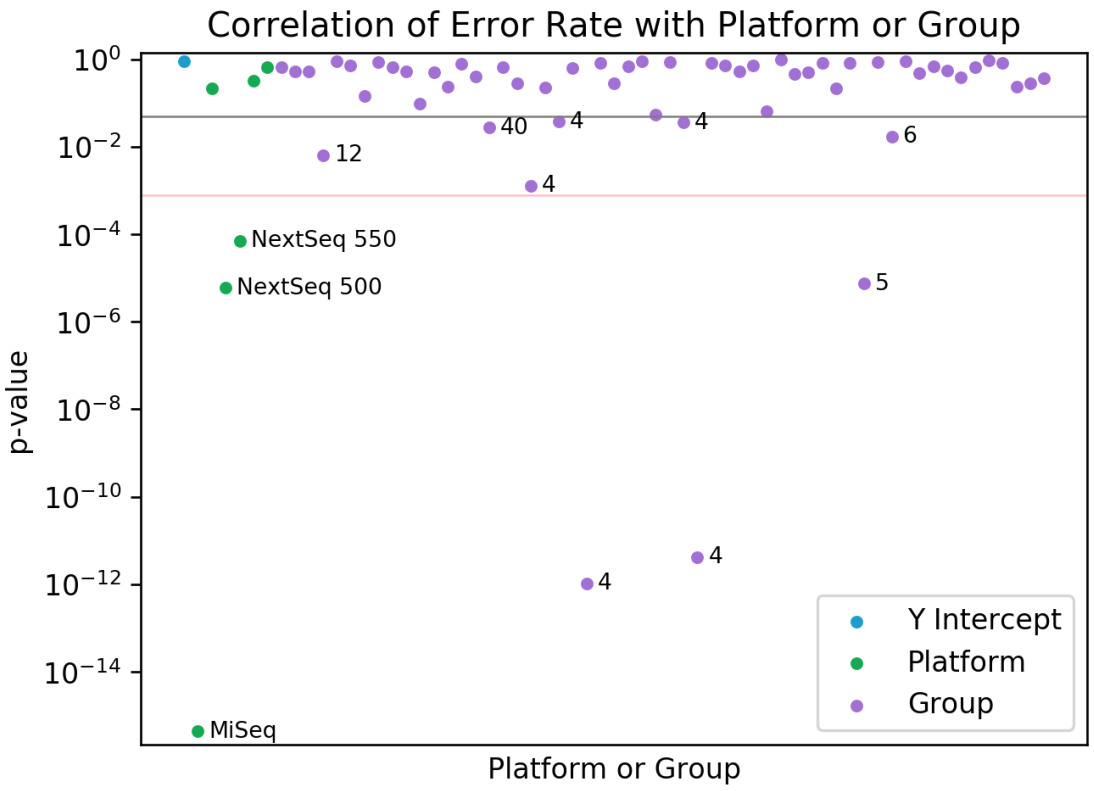


Figure 4.S2 *P*-values of regression coefficients. The upper, gray horizontal line is a *p*-value of 0.05. The lower, red horizontal line is a Bonferroni-corrected *p*-value of 0.05/64 Significant platforms and groups are labeled. Group labels are the number of runs produced by the group that appear in the survey. The platforms are in the same order as in Figure 4.S1.

4.4.2 Base frequencies near error sites

Studies of errors in Illumina sequencing have consistently shown the importance of sequence context. In order to investigate whether there are platform-specific differences, we examined the genomic context surrounding each error we detected. Fig. 4.3 shows the frequency of each base at each position relative to each error. We further divided each platform by the type of substitution, focusing in this case only on errors that replaced A with a different base.

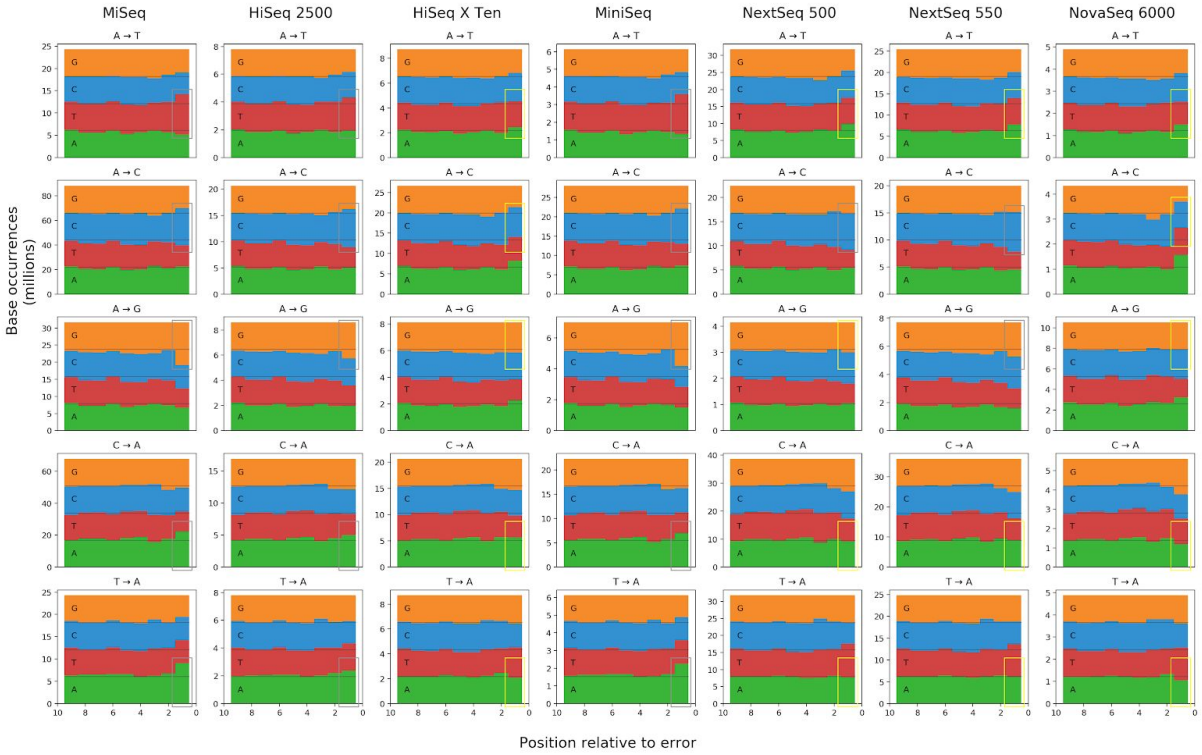


Figure 4.3 The count of each base in the genomic context surrounding each type of substitution. The X axis represents the distance from the error base. The rightmost slot is the closest (adjacent to the error base), and the leftmost is the furthest (9 bases from the error). The Y axis is the count of how many of each base was observed at each distance from a substitution of that type. Boxes highlight the count of the error base adjacent to the error. Gray boxes surround counts which are overrepresented, as expected. Yellow boxes surround counts which do not seem to follow this pattern.

Most platforms followed the common Illumina bias toward substituting a base of the same type as the one preceding the error. But this phenomenon was inconsistent. Neither HiSeq X Ten nor NovaSeq 6000 showed much evidence of this trend when looking just at the cumulative base counts. The same was true for both NextSeq platforms, but only in the case of A⟶T, C⟶A, and T⟶A substitutions. And in all of the substitutions away from A where this phenomenon was missing, A was overrepresented instead.

Notably, these patterns do not show clear signatures of errors due to the pattern of fluorophore dyes in the respective instruments. MiSeq and HiSeq use traditional 4-color

imaging, where the wavelengths for C and A overlap and T and G overlap (Whiteford et al. 2009). But Fig. 4.3 does not show the C/A and T/G associations one might expect from this overlap. And MiniSeq, NextSeq, and NovaSeq use 2-color imaging, where A shares a wavelength with both C and T, and G is unlabeled. There have been reports of this resulting in overcalling of G's, leading to stretches of polyG's (Andrews 2016). PolyG's would result in a greater overrepresentation of G's leading up to a G substitution. But this is not clearly observed in the plots. MiniSeq may be an exception, but the effect is only seen in the immediately adjacent base, not any others. Another expected error would be C→A and T→A substitutions when preceded by T or C, respectively. This could occur when phasing causes the red from a preceding C to mix with the green of a T, or vice versa. This mixed red/green signal could be misread as an A, which is normally a red/green mix. But no platform seemed to show an overrepresentation of T adjacent to C→A substitutions or C next to T→A ones.

4.4.3 Frequency of trimer motifs at error sites

Certain sequence motifs are known to be especially error-prone under Illumina sequencing. We investigated the motifs associated with our errors to discover if there were platform-dependent differences. We checked the three bases leading up to, and including the error base at every error site. Fig. 4.4 shows the frequency of each trimer at our error sites, normalized by the expected frequency if there were no correlation. The top 10 trimers in the entire survey are shown.

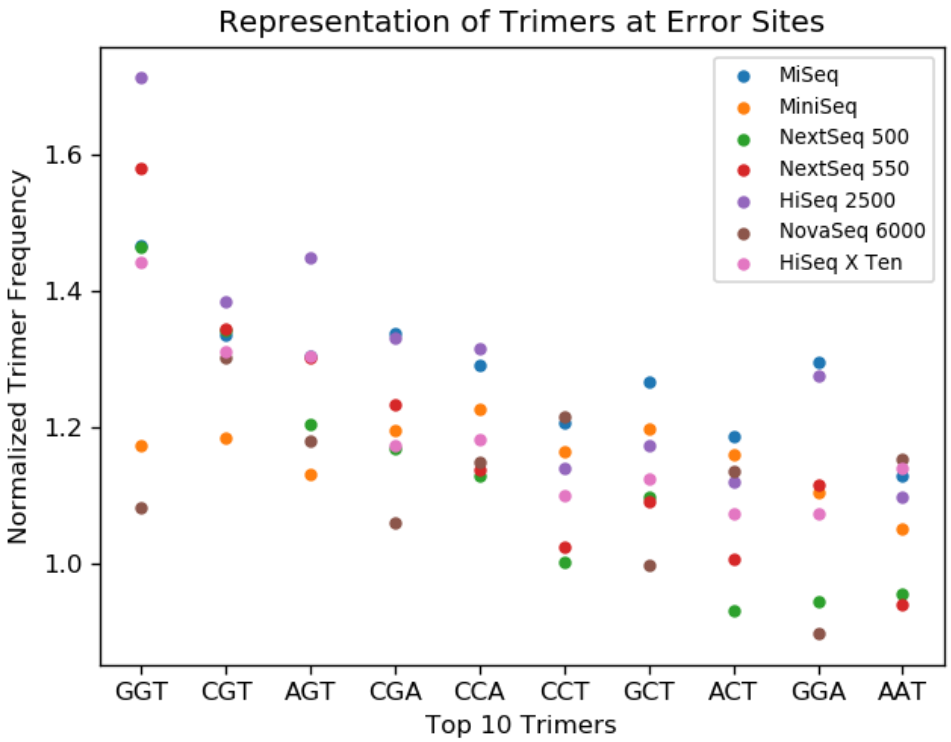


Figure 4.4 The frequency of trimers in the sequence context near errors. Each trimer is the three reference bases leading up to, and including, an error site. As described in the methods, we counted the occurrences of each trimer, then normalized by the abundance of the corresponding trimer in the genome. A normalized frequency of 1 would mean the trimer is associated with errors exactly as much as in the null hypothesis where errors are randomly distributed. Trimers are presented in order of their median frequency among all platforms.

The most error-associated trimer in our dataset is GGT, a motif which has also been seen in previous studies (Allhoff et al. 2013; Meacham et al. 2011). Most instruments show a similar trend, but there is wide variation. GGT is far more overrepresented in HiSeq 2500 errors than any other platform. NovaSeq 6000 is at the low end of several of the top trimers, indicating it is less influenced by these motifs. In contrast, HiSeq 2500 and MiSeq seem to show the most motif-dependent errors.

4.4.4 Differences in error rate types

A common error mode in Illumina platforms occurs near homopolymers. After a repeat of the same base multiple times, Illumina reads will often substitute the first base after the homopolymer with the homopolymer base. This can occur due to phasing: lagging molecules will still be incorporating homopolymer bases while the instrument is reading the post-homopolymer base (Fuller et al. 2009; Pfeiffer et al. 2018).

In our survey, we observed that this type of error is common. If we define a homopolymer as any run of three or more of the same base, this error constitutes between 0.7% and 5.3% of all errors, depending on the base and platform. Fig. 4.5 shows how common this error is, depending on how one defines a homopolymer and which base the homopolymer is composed of.

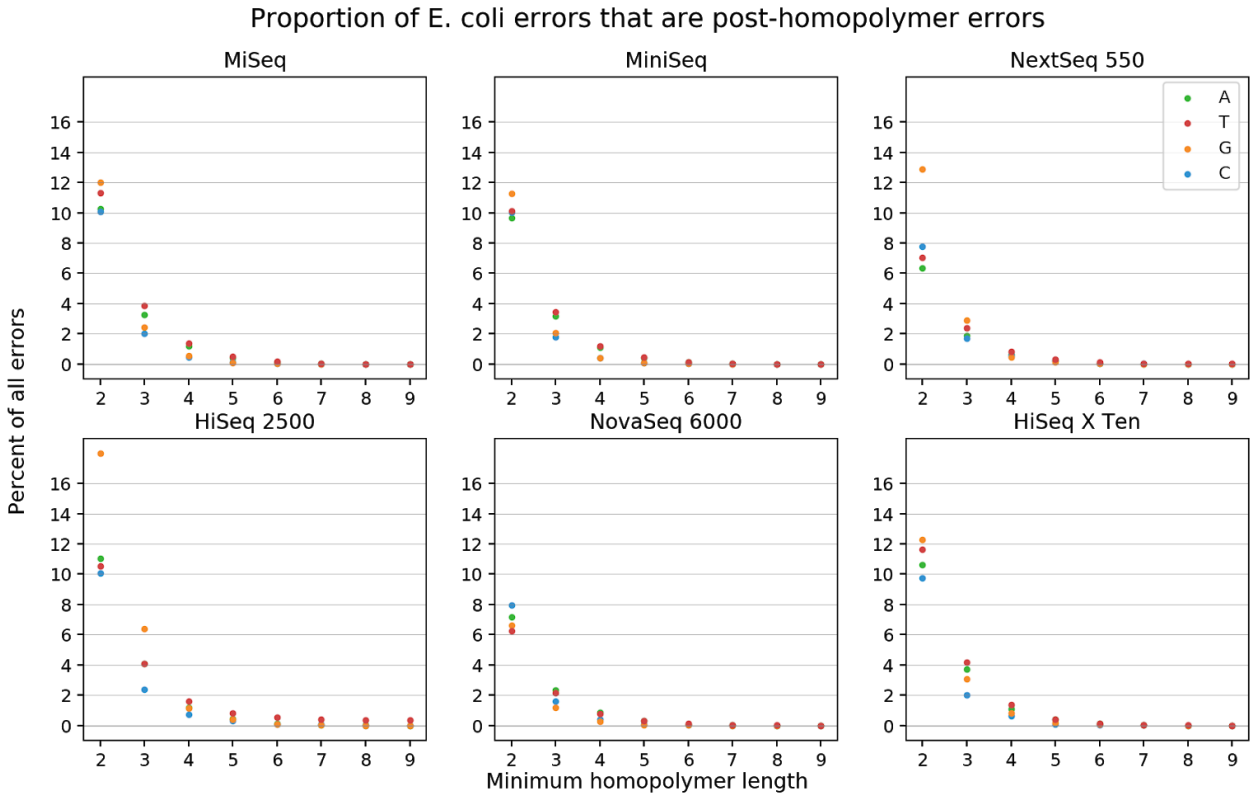


Figure 4.5 Frequency of post-homopolymer errors. For each platform, we found all errors where the error base is identical to the preceding genomic base. If the preceding genomic base is part of a single-base repeat (homopolymer), we might call the error a post-homopolymer error, depending on the length of the repeat. This figure shows what percent of all errors are post-homopolymer errors,

depending on the threshold one uses for the definition of a post-homopolymer error. Numbers are broken out by error/homopolymer base. For example, if one decides that any error preceded by a run of at least 3 bases of the same identity qualifies as a post-homopolymer error, then about 2% of all C substitutions in MiniSeq are post-homopolymer errors.

Fig. 4.6 shows the rate of these errors relative to the neutral expectation if there were no correlation between errors and homopolymers. Each point is the number of homopolymer errors divided by the number of homopolymers of that type in the *E. coli* genome. So in MiniSeq, G substitutions follow G 3mers about three times more often than if substitutions were distributed randomly. As expected, across platforms, even the normalized rates increase with homopolymer length. Another pattern that is common across platforms is that G/C homopolymers produce errors at a higher rate than A/T homopolymers. Instruments which do not fully follow this pattern are NextSeq 550, HiSeq X Ten, and NovaSeq 6000. Interestingly, in the latter case, the pattern seems to be inverted, with A/T more overrepresented. Another peculiarity of NovaSeq 6000, is that G/C post-homopolymer errors rates are generally quite low, in many cases occurring less often than expected by random chance.

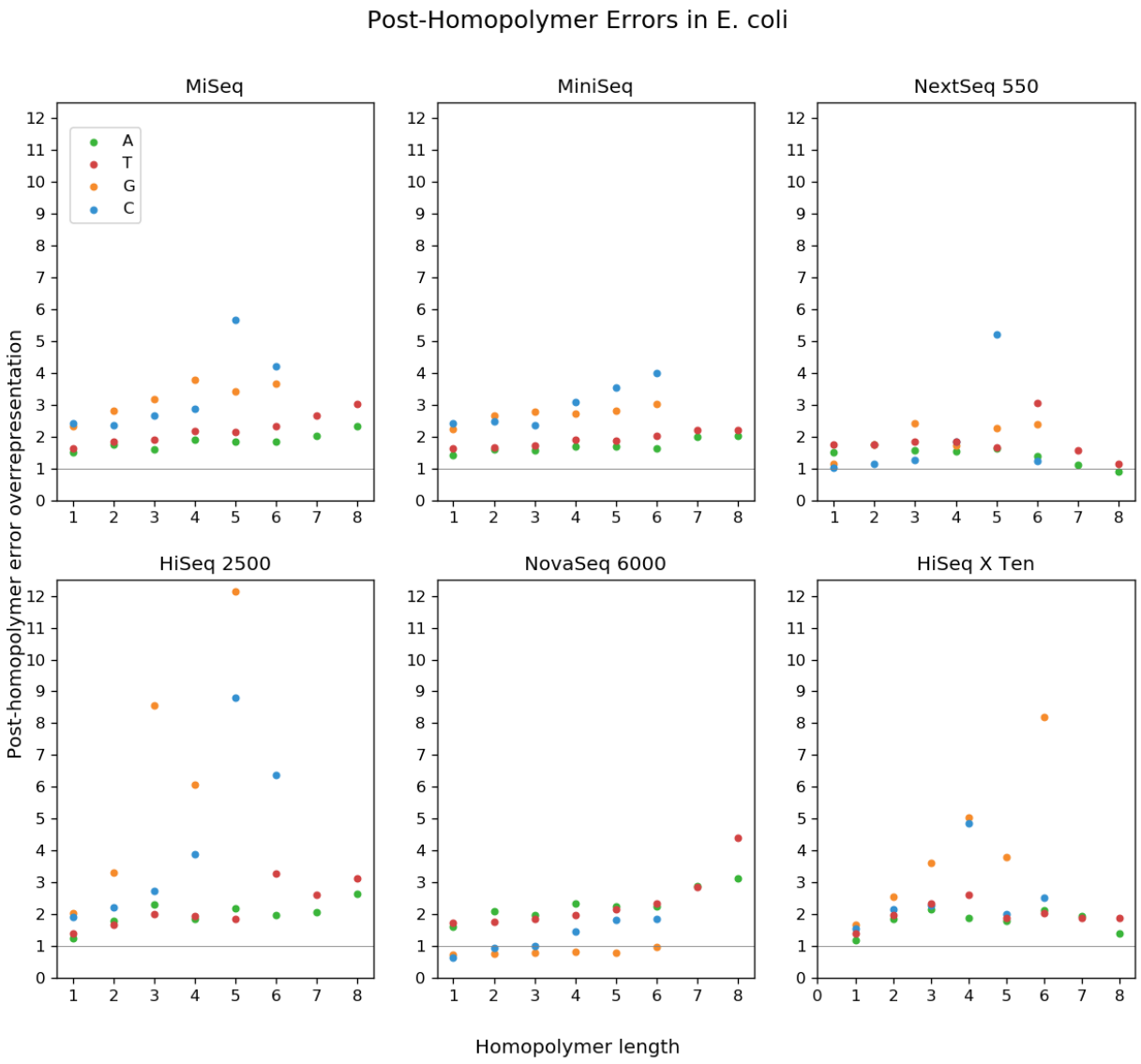


Figure 4.6 Relative frequencies of post-homopolymer errors in *E. coli*, by length. Each post-homopolymer error was categorized by its error base and the length of the preceding homopolymer. Each total was divided by the expected number of errors of that type in the null hypothesis where errors are randomly distributed. An "overrepresentation" of 1 is what would be expected in the null hypothesis, while 2 would be twice as abundant.

4.4.5 Further examination

Here we developed a method which can be automatically applied to any paired-end sequencing dataset. We demonstrated its utility by applying it to a survey beyond the scale of

manual annotation. We were able to show correlations between sequencing platforms, experimenters, and error rates. But there may be other important factors explaining differences between samples. The cluster of NovaSeq 6000 samples with higher error rates were all produced by the Gene Expression Omnibus group, which is a repository for transcriptomic data (Edgar, Domrachev, and Lash 2002). This suggests that RNA-seq datasets may tend to have higher error rates. Luckily, information like this is captured in SRA metadata. And with links to other NCBI databases like BioProject and BioSample, there is even more metadata which could be automatically obtained about each sample and tested for correlations. For instance, a previous analysis of SRA metadata distinguished between published and unpublished datasets, as this may correlate with data quality (Nakazato, Ohta, and Bono 2013). Unfortunately, the SRA metadata schema is far from complete, and there are some important features of a sequencing experiment which are not captured. For instance, the version of the Illumina reagent kit has been observed affecting the bias of sequencing data (Ross et al. 2013).

Our method of error detection has great advantages in its ease of automation, ability to be applied retroactively to a dataset, and its ability to identify errors coming from sequencing alone. But it can be made even less biased and automatable. In our survey, we relied on alignment to a reference sequence in order to find the overlaps in read pairs. In theory, the examination of overlaps does not require information from a reference at all. One could simply perform a two-way alignment of each pair of mates to each other. There are many pairwise alignment algorithms available, many of which will yield acceptable results even with default parameters. Some algorithms have trouble when there is minimal overlap between mates, but careful choice of algorithms and parameters can minimize this issue. By using pairwise alignment, we would simplify the workflow, eliminating the step that determines the best reference sequence. It would also eliminate possible biases from alignment artifacts. Additionally, it would remove the need to know anything about the subject of the sequencing experiment. This could allow surveying across organisms, so one could determine if that is a variable correlated with quality.

4.5 Acknowledgements

Special thanks to Drs. Barbara Arbeithuber and Marius van den Beek for help conceptualizing overlap error detection. These efforts have been funded in part by NHGRI Grant U41 HG006620, NSF ABI Grant 1661497, NIAID grant R01 AI134384. We would like to thank the entire Galaxy team for supporting this effort.

5 Conclusion

With the methods described here, we have pushed further toward accurate variant detection through sequencing. With the overlap error detection method, we have a way to directly measure sequencing error on a large scale. And with Du Novo, we now have an efficient and reference-free way to process the data from the most accurate consensus sequencing method, duplex sequencing.

We have demonstrated the potential of overlap error detection as a general tool for studying sequencing quality on a large scale. Its main limitation is that it requires an experiment's fragment size and read length to yield appreciable read overlap. But what is gained in exchange for this one requirement is freedom from any constraint on library preparation or sample choice, and a choice of a wide array of applicable sequencing technologies.

Results from real and simulated data show that Du Novo is at least as accurate as the standard pipeline by Kennedy *et al*. (Kennedy et al. 2014). On top of accuracy and lack of a reference requirement, Du Novo has an advantage in yield. The original release of Du Novo could already produce more duplex consensus sequences than the standard pipeline at certain sites, and the addition of barcode error correction further increased its consensus recovery rate by 12-16% in our analyses. And Du Novo 2.0 comes with massive speed improvements, up to 9x faster than the original.

5.1 Wide applicability of overlap error detection

Many next-generation sequencing methods interrogate single DNA fragments in a way that allows separate measurements of the same molecules. A common pattern is clonal amplification of a single molecule, then sequencing by synthesis in a manner that allows paired-end reads that can overlap. 454's emulsion PCR and Illumina's bridge amplification both follow this general approach (Margulies et al. 2005; Bentley et al. 2008). The Pacific Biosciences SMRT system does not produce paired-end reads, but it has the ability to resequence the same

molecule multiple times, with the “circular consensus sequencing” approach (Travers et al. 2010).

This commonality among sequencing platforms means that our approach of detecting errors from sequencing overlap should be widely applicable. Thus we could bring the benefits of retroactive, direct measurement of sequencing error to most existing datasets. Comparing different models within the Illumina platform yielded benefits, and comparing between platforms could be even more fruitful. For example, BGISEQ is a relatively new adaptation of the DNA nanoball technique which promises to be competitive with Illumina in some applications (Huang et al. 2017). The DNA nanoball process is quite different from emulsion or bridge amplification, but it remains a way to amplify a single fragment and cluster its descendents (Drmanac et al. 2010). Evaluating its accuracy in a systematic way would help researchers decide whether to try this alternative.

5.2 Uses in continuous variant calling

Du Novo has made it possible to study variants at the lowest frequencies observable with current technology. In 2019, Mei *et al*. observed the behavior of transient *de novo* mutations under selective pressure in *E. coli*. The bacteria had been transformed with a tetracycline resistance plasmid and clonal populations were incubated in a tetracycline medium. With Du Novo, the authors detected several mutations that could increase the plasmid’s copy number. These mutations increased in frequency with time, indicating selection, but peaked under 1% MAF, declining afterward toward extinction. The authors found evidence supporting a hypothesis that mutations in the bacterial chromosome arose which were even more beneficial than these plasmid mutations. These chromosomal mutations are less probable, so arise later in the experiment. And they are unlikely to arise in the 1% of the bacteria with the plasmid mutations, so these form two separate populations. Then, the population with the more beneficial chromosomal mutation outcompetes the plasmid-mutant population. So the plasmid mutations never exceed 1% MAF, so this phenomenon would be extremely difficult to witness with other methods. But duplex sequencing and Du Novo allowed

the authors to not only detect these mutations, but to observe their rise and fall, at frequencies as low as 0.07%. (Mei et al. 2019)

5.3 Further performance improvements

Du Novo 2.0 achieved a nearly order-of-magnitude improvement on the performance of Du Novo 1.0 in `align-families.py`, the major pipeline bottleneck. But there are still some remaining hurdles to faster execution.

In the implementation of MAFFT used in Du Novo, an average of 12 temporary files were created per alignment, with some of those being accessed multiple times. The use of Kalign2 and its close integration into the `align-families.py` process means that all but one of these temporary files were eliminated. But the Python process still creates one temporary file per alignment, to transmit the input sequences into Kalign2. It is completely possible to instead provide the inputs directly, as arguments to the main Kalign2 C function. Kalign2 is open source and the code is understandable, making it much easier to modify and integrate than MAFFT. But understanding the data structures used to represent input sequences is significantly more difficult than parsing those used to represent the output alignment. Recreating valid input data structures would require either collaboration with the Kalign2 authors or much more reverse-engineering.

Another clear opportunity for speed improvements is the parallelization algorithm. As described earlier, the main limitation in the parallelization of `align-families.py` is the desire to produce family alignments in the same order as they were input, and to stream the results. If one of these features were abandoned, the parallelization algorithm could make optimal use of available CPU cores. Its implementation could also be greatly simplified. For example, if order did not have to be preserved, every time a core finished an alignment, it could be output and a new one could be immediately given to it. There would not be any need for grouping jobs into "chunks", caching them in memory, and making sure to output them in their input order. The results could still be streamed, meaning order preservation is the only casualty. If a user desires ordering, this could still be allowed by inserting an index into each line of output, representing its place in the input. Then, a simple `sort` command could restore order. This would sacrifice

streaming instead of order. In this design, only two of the three features (parallelism, order, and streaming) could be offered at once, but the user could choose which two.

These two improvements are ways to push further into areas Du Novo 2.0 already addressed. But one area still unexplored is the issue of interprocess I/O. The parallelization in `align-families.py` distributes the multiple sequence alignments evenly across all the allocated cores. But the coordination, as well as the overall input and output, is performed by a central process. This means that the central process sends the entire sequence of each family to the worker processes, and receives the entire alignment back. This can cause a "traffic jam" of data flowing into and out of the central process, limiting the speedup available from parallelization. This overhead may be responsible for the limited success of parallelizing consensus creation in `make-consensi.py`. The same parallelization feature is available in `make-consensi.py` as in `align-families.py`, but below a certain number of worker processes, this step actually takes longer to complete than without worker processes. Even when utilizing 8 cores, it is slower than using a single process. This could be because of the smaller amount of time needed to create a consensus sequence than to perform a multiple sequence alignment. The amount of interprocess communication is about half as much as in alignment: the full family is sent, but only the consensus sequence is received. But the time taken to perform the work is much less, meaning the ratio of time spent in interprocess communication to that in the actual job is much higher.

5.4 Improving barcode error correction

Barcode error correction in Du Novo has been able to "rescue" reads with up to three errors in their barcodes, recovering valuable sequencing data. However, PCR/sequencing errors are not the only error modes for molecular barcoding.

5.4.1 Detecting barcode collisions

As mentioned earlier, the theoretical probability of two different fragments receiving the same barcodes can be made exceedingly small. But reducing the barcode lengths can quickly make collisions a real possibility. And care must be taken in the rest of the experimental design, since the collision risk depends as much on the number of fragments sequenced as the

barcode length (Liang et al. 2014). And even if the experimental design minimizes this risk, any imperfections in the synthesis of the degenerate oligonucleotides will increase it. So it is worth considering methods for detecting collisions and even correcting for them.

A straightforward strategy for detecting collisions is to examine a family's multiple sequence alignment for how similar the sequences are to each other. The result of a collision would be an artifactual family composed of two different real families. If there are similar numbers of reads in the two real families, it is possible that neither would reach the consensus calling threshold, resulting in an all-N consensus sequence. If one family exceeds the threshold, then its sequence will be the consensus. Then, quantifying the identity of each read with the consensus will reveal a group of reads with very low identity with it. The strategy of performing O(N) comparisons with the consensus would avoid performing $O(N^2)$ comparisons between every pair of reads.

However, the processing of a family already involves calculating the similarity between every pair of reads. As discussed earlier, Kalign2 calculates a distance matrix of all sequences. If it were possible to obtain this distance matrix once it is calculated, one could obtain a more thorough result without re-processing all the sequences in the family. The fact that Kalign2's code is open source and clean makes it feasible to make the necessary modifications.

5.4.2 Chimera detection

There is another type of PCR error which can affect barcodes, but not by changing their sequence. Through several mechanisms, it is possible for PCR to produce molecules which are chimeras of two different input fragments (Kanagawa 2003). The chimera sequence will begin the same as fragment A, and at some point in switch to that of fragment B. What this means for duplex is that in some raw read pairs, one tag will be from fragment A and the other from fragment B. If the chimera formation occurs early in PCR, this may affect the whole family. If it occurs later, only a subset of reads will have this combination of barcodes. Preliminary data using methods similar to that in section 3.3.2 indicates that some experiments can result in 20% of families being artifacts from chimera formation.

The ability to identify, and even correct these chimeras could thus recover an important amount of data. This would probably require examining the set of all barcodes, looking at the

α and β halves separately. A straightforward approach is to use a hash table to check whether a given tag appeared in two different families. This would only allow for exact matches without errors, but it could catch the majority of chimeras. These could be removed and be prevented from interfering with downstream analysis.

Correcting the chimeras, however, would require a much more sophisticated analysis. Because the splice point could occur anywhere in the fragment, distinguishing the two parts would require analyzing the similarity of portions of the sequence to some reference. This would essentially be splice junction analysis like that performed by RNA-seq tools like TopHat (Trapnell, Pachter, and Salzberg 2009). This would allow separating the two parts of the affected reads, assigning them to their true origin families, and using their sequences to help build consensuses. If the early measurements are accurate, this would result in another large fraction of data being recovered.

References

Ahn, Eun Hyun, and Seung Hyuk Lee. 2019. “Detection of Low-Frequency Mutations and Identification of Heat-Induced Artifactual Mutations Using Duplex Sequencing.” *International Journal of Molecular Sciences* 20 (1): 199.

Allhoff, Manuel, Alexander Schönhuth, Marcel Martin, Ivan G. Costa, Sven Rahmann, and Tobias Marschall. 2013. “Discovering Motifs That Induce Sequencing Errors.” *BMC Bioinformatics* 14 (SUPPL.5): 1–10.

Altshuler, David, G. R. Abecasis, A. Auton, L. D. Brooks, R. M. Durbin, Richard A. Gibbs, Matt E. Hurles, et al. 2010. “A Map of Human Genome Variation from Population-Scale Sequencing.” *Nature* 467 (7319): 1061–73.

Andrews, Simon. 2016. “Illumina 2 Colour Chemistry Can Overcall High Confidence G Bases.” *QC Fail* (blog). 2016. https://sequencing.qcfail.com/articles/illumina-2-colour-chemistry-can-overcall-high-confidence-g-bases/.

Arbeithuber, Barbara, Kateryna D. Makova, and Irene Tiemann-Boege. 2016. “Artifactual Mutations Resulting from DNA Lesions Limit Detection Levels in Ultrasensitive Sequencing Applications.” *DNA Research: An International Journal for Rapid Publication of Reports on Genes and Genomes* 23 (6): 547–59.

Bentley, David R., Shankar Balasubramanian, Harold P. Swerdlow, Geoffrey P. Smith, John Milton, Clive G. Brown, Kevin P. Hall, et al. 2008. “Accurate Whole Human Genome Sequencing Using Reversible Terminator Chemistry.” *Nature* 456 (7218): 53–59.

Berno, A. J. 1996. “A Graph Theoretic Approach to the Analysis of DNA Sequencing Data.” *Genome Research* 6 (2): 80–91.

Blankenberg, Daniel, Gregory Von Kuster, Emil Bouvier, Dannon Baker, Enis Afgan, Nicholas Stoler, James Taylor, and Anton Nekrutenko. 2014. “Dissemination of Scientific Software with Galaxy ToolShed.” *Genome Biology* 15 (2): 403.

Bomba, Lorenzo, Klaudia Walter, and Nicole Soranzo. 2017. “The Impact of Rare and Low-Frequency Genetic Variants in Common Disease.” *Genome Biology* 18 (1): 77.

Botstein, D., R. L. White, M. Skolnick, and R. W. Davis. 1980. “Construction of a Genetic Linkage Map in Man Using Restriction Fragment Length Polymorphisms.” *American Journal of Human Genetics* 32 (3): 314–31.

Costello, Maura, Trevor J. Pugh, Timothy J. Fennell, Chip Stewart, Lee Lichtenstein, James C. Meldrim, Jennifer L. Fostel, et al. 2013. “Discovery and Characterization of Artifactual Mutations in Deep Coverage Targeted Capture Sequencing Data due to Oxidative DNA Damage during Sample Preparation.” *Nucleic Acids Research* 41 (6): 1–12.

Cree, Lynsey M., David C. Samuels, Susana Chuva de Sousa Lopes, Harsha Karur Rajasimha, Passorn Wonnapinij, Jeffrey R. Mann, Hans-Henrik M. Dahl, and Patrick F. Chinnery. 2008. “A Reduction of Mitochondrial DNA Molecules during Embryogenesis Explains the Rapid Segregation of Genotypes.” *Nature Genetics* 40 (2): 249–54.

Czene, Kamila, Paul Lichtenstein, and Kari Hemminki. 2002. “Environmental and Heritable Causes of Cancer among 9.6 Million Individuals in the Swedish Family-Cancer Database.” *International Journal of Cancer* 99 (2): 260–66.

Degner, J. F., J. C. Marioni, A. A. Pai, J. K. Pickrell, E. Nkadori, Y. Gilad, and J. K. Pritchard. 2009. “Effect of Read-Mapping Biases on Detecting Allele-Specific Expression from RNA-Sequencing Data.”

Bioinformatics 25 (24): 3207–12.
Dimond, Rebecca. 2015. “Social and Ethical Issues in Mitochondrial Donation.” *British Medical Bulletin* 115 (1): 173–82.
Drmanac, R., A. B. Sparks, M. J. Callow, A. L. Halpern, N. L. Burns, B. G. Kermani, P. Carnevali, et al. 2010. “Human Genome Sequencing Using Unchained Base Reads on Self-Assembling DNA Nanoarrays.” *Science* 327 (5961): 78–81.
Edgar, Ron, Michael Domrachev, and Alex E. Lash. 2002. “Gene Expression Omnibus: NCBI Gene Expression and Hybridization Array Data Repository.” *Nucleic Acids Research* 30 (1): 207–10.
Ewing, B., and P. Green. 1998. “Base-Calling of Automated Sequencer Traces Using Phred. II. Error Probabilities.” *Genome Research* 8 (3): 186–94.
Feng, D. F., and R. F. Doolittle. 1987. “Progressive Sequence Alignment as a Prerequisite to Correct Phylogenetic Trees.” *Journal of Molecular Evolution* 25 (4): 351–60.
Fennell, Tim, and Nils Homer. 2018. *Fgbio* (version 0.6.1). fulcrumgenomics. https://github.com/fulcrumgenomics/fgbio.
Fisher, R. A. 1930. “The Genetical Theory of Natural Selection, 272 Pp.” Oxford: Clarendon Press.
Fox, Edward J., and Kate S. Reid-Bayliss. 2014. “Accuracy of Next Generation Sequencing Platforms.” *Journal of Next Generation Sequencing & Applications* 01 (01). https://doi.org/10.4172/2469-9853.1000106.
Frazer, Kelly A., Sarah S. Murray, Nicholas J. Schork, and Eric J. Topol. 2009. “Human Genetic Variation and Its Contribution to Complex Traits.” *Nature Reviews. Genetics* 10 (4): 241–51.
Freitas, Richard, Joseph Slember, Wayne Sawdon, and Lawrence Chiu. 2011. “GPFS Scans 10 Billion Files in 43 Minutes,” 1–28.
Fuller, Carl W., Lyle R. Middendorf, Steven A. Benner, George M. Church, Timothy Harris, Xiaohua Huang, Stevan B. Jovanovich, et al. 2009. “The Challenges of Sequencing by Synthesis.” *Nature Biotechnology* 27 (11): 1013–23.
Garrison, Erik, and Gabor Marth. 2012. “Haplotype-Based Variant Detection from Short-Read Sequencing.” *arXiv.org*, July. http://arxiv.org/abs/1207.3907v2.
Gatz, Margaret, Chandra A. Reynolds, Laura Fratiglioni, Boo Johansson, James A. Mortimer, Stig Berg, Amy Fiske, and Nancy L. Pedersen. 2006. “Role of Genes and Environments for Explaining Alzheimer Disease.” *Archives of General Psychiatry* 63 (2): 168–74.
Goodwin, Sara, John D. McPherson, and W. Richard McCombie. 2016. “Coming of Age: Ten Years of next-Generation Sequencing Technologies.” *Nature Reviews. Genetics* 17 (6): 333–51.
Goto, H., B. Dickins, E. Afgan, I. M. Paul, J. Taylor, K. D. Makova, and A. Nekrutenko. 2011. “Dynamics of Mitochondrial Heteroplasmy in Three Families Investigated via a Repeatable Re-Sequencing Study.” *Genome Biology* 12 (6): R59.
Guo, Yan, Jiang Li, Chung-I Li, Jirong Long, David C. Samuels, and Yu Shyr. 2012. “The Effect of Strand Bias in Illumina Short-Read Sequencing Data.” *BMC Genomics* 13 (1): 1.
Hershberg, Ruth. 2015. “Mutation—The Engine of Evolution: Studying Mutation and Its Role in the Evolution of Bacteria.” *Cold Spring Harbor Perspectives in Biology* 7 (9). https://doi.org/10.1101/cshperspect.a018077.
Hess, Matthias, Alexander Sczyrba, Rob Egan, Tae-Wan Kim, Harshal Chokhawala, Gary Schroth, Shujun Luo, et al. 2011. “Metagenomic Discovery of Biomass-Degrading Genes and Genomes from Cow Rumen.” *Science* 331 (6016): 463–67.
Hiatt, Joseph B., Rupali P. Patwardhan, Emily H. Turner, Choli Lee, and Jay Shendure. 2010. “Parallel, Tag-Directed Assembly of Locally Derived Short Sequence Reads.” *Nature Methods* 7 (2): 119–22.
Higgins, D. G., and P. M. Sharp. 1988. “CLUSTAL: A Package for Performing Multiple Sequence Alignment on a Microcomputer.” *Gene* 73 (1): 237–44.
Hindson, C. M., J. R. Chevillet, H. A. Briggs, E. N. Gallichotte, I. K. Ruf, B. J. Hindson, R. L. Vessella, and M.

Tewari. 2013. “Absolute Quantification by Droplet Digital PCR versus Analog Real-Time PCR.” *Nature Methods* 10 (10): 1003–5.

Hoang, Margaret L., Isaac Kinde, Cristian Tomasetti, K. Wyatt McMahon, Thomas A. Rosenquist, Arthur P. Grollman, Kenneth W. Kinzler, Bert Vogelstein, and Nickolas Papadopoulos. 2016. “Genome-Wide Quantification of Rare Somatic Mutations in Normal Human Tissues Using Massively Parallel Sequencing.” *Proceedings of the National Academy of Sciences* 113 (35): 9846–51.

Hodgkinson, Alan, Youssef Idaghdour, Elias Gbeha, Jean-Christophe Grenier, Elodie Hip-Ki, Vanessa Bruat, Jean-Philippe Goulet, Thibault de Malliard, and Philip Awadalla. 2014. “High-Resolution Genomic Analysis of Human Mitochondrial RNA Sequence Variation.” *Science* 344 (6182): 413–15.

Horton, Roger, Richard Gibson, Penny Coggill, Marcos Miretti, Richard J. Allcock, Jeff Almeida, Simon Forbes, et al. 2008. “Variation Analysis and Gene Annotation of Eight MHC Haplotypes: The MHC Haplotype Project.” *Immunogenetics* 60 (1): 1–18.

Huang, Jie, Xinming Liang, Yuankai Xuan, Chunyu Geng, Yuxiang Li, Haorong Lu, Shoufang Qu, et al. 2017. “A Reference Human Genome Dataset of the BGISEQ-500 Sequencer.” *GigaScience* 6 (5): 1–9.

Hubby, J. L., and R. C. Lewontin. 1966. “A Molecular Approach to the Study of Genic Heterozygosity in Natural Populations. I. The Number of Alleles at Different Loci in Drosophila Pseudoobscura.” *Genetics* 54 (2): 577–94.

Hyttinen, Valma, Jaakko Kaprio, Leena Kinnunen, Markku Koskenvuo, and Jaakko Tuomilehto. 2003. “Genetic Liability of Type 1 Diabetes and the Onset Age among 22,650 Young Finnish Twin Pairs: A Nationwide Follow-up Study.” *Diabetes* 52 (4): 1052–55.

Jabara, Cassandra B., Corbin D. Jones, Jeffrey Roach, Jeffrey A. Anderson, and Ronald Swanstrom. 2011. “Accurate Sampling and Deep Sequencing of the HIV-1 Protease Gene Using a Primer ID.” *Proceedings of the National Academy of Sciences of the United States of America* 108 (50): 20166–71.

Jagtap, Maheshkumar P. 2009. “Era of Multi-Core Processors.” *Power* 2: 2.

Jain, Miten, Sergey Koren, Karen H. Miga, Josh Quick, Arthur C. Rand, Thomas A. Sasani, John R. Tyson, et al. 2018. “Nanopore Sequencing and Assembly of a Human Genome with Ultra-Long Reads.” *Nature Biotechnology* 36 (4): 338–45.

Jones, D. T., W. R. Taylor, and J. M. Thornton. 1992. “The Rapid Generation of Mutation Data Matrices from Protein Sequences.” *Computer Applications in the Biosciences: CABIOS* 8 (3): 275–82.

Kanagawa, Takahiro. 2003. “Bias and Artifacts in Multitemplate Polymerase Chain Reactions (PCR).” *Journal of Bioscience and Bioengineering* 96 (4): 317–23.

Kans, J. 2013. *Entrez Direct: E-Utilities on the UNIX Command Line* (version 12.2). Bethesda, Maryland: National Center for Biotechnology Information. https://www.ncbi.nlm.nih.gov/books/NBK179288/.

Katoh, Kazutaka, Kazuharu Misawa, Kei-Ichi Kuma, and Takashi Miyata. 2002. “MAFFT: A Novel Method for Rapid Multiple Sequence Alignment Based on Fast Fourier Transform.” *Nucleic Acids Research* 30 (14): 3059–66.

Katoh, K., and D. M. Standley. 2013. “MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability.” *Molecular Biology and Evolution* 30 (4): 772–80.

Kebschull, Justus M., and Anthony M. Zador. 2015. “Sources of PCR-Induced Distortions in High-Throughput Sequencing Data Sets.” *Nucleic Acids Research* 43 (21). https://doi.org/10.1093/nar/gkv717.

Kennedy, Scott R., Michael W. Schmitt, Edward J. Fox, Brendan F. Kohrn, Jesse J. Salk, Eun Hyun Ahn, Marc J. Prindle, et al. 2014. “Detecting Ultralow-Frequency Mutations by Duplex Sequencing.” *Nature Protocols* 9 (11): 2586–2606.

Kim, S. Y., K. E. Lohmueller, A. Albrechtsen, Y. Li, T. Korneliussen, G. Tian, N. Grarup, et al. 2011. “Estimation of Allele Frequency and Association Mapping Using next-Generation Sequencing Data.”

BMC Bioinformatics 12: 231.
Kinde, Isaac, Chetan Bettegowda, Yuxuan Wang, Jian Wu, Nishant Agrawal, Ie-Ming Shih, Robert Kurman, et al. 2013. "Evaluation of DNA from the Papanicolaou Test to Detect Ovarian and Endometrial Cancers." *Science Translational Medicine* 5 (167): 167ra4.
Kinde, I., J. Wu, N. Papadopoulos, K. W. Kinzler, and B. Vogelstein. 2011. "Detection and Quantification of Rare Mutations with Massively Parallel Sequencing." *Proceedings of the National Academy of Sciences of the United States of America* 108 (23): 9530–35.
Knuth, Donald E. 1973. *The Art of Computer Programming, Volume 3: Searching and Sorting*. Vol. 3. Reading, MA: Addison-Westley Publishing Company.
Kodama, Yuichi, Martin Shumway, Rasko Leinonen, and International Nucleotide Sequence Database Collaboration. 2012. "The Sequence Read Archive: Explosive Growth of Sequencing Data." *Nucleic Acids Research* 40 (Database issue): D54–56.
Langmead, Ben, and Steven L. Salzberg. 2012. "Fast Gapped-Read Alignment with Bowtie 2." *Nature Methods* 9 (4): 357–59.
Langmead, Ben, Cole Trapnell, Mihai Pop, and Steven L. Salzberg. 2009. "Ultrafast and Memory-Efficient Alignment of Short DNA Sequences to the Human Genome." *Genome Biology* 10 (3): R25.
Larionov, Alexey, Andreas Krause, and William Miller. 2005. "A Standard Curve Based Method for Relative Real Time PCR Data Processing." *BMC Bioinformatics* 6: 62.
Lassmann, Timo, Oliver Frings, and Erik L. L. Sonnhammer. 2009. "Kalign2: High-Performance Multiple Alignment of Protein and Nucleotide Sequences Allowing External Features." *Nucleic Acids Research* 37 (3): 858–65.
Lassmann, Timo, and Erik L. L. Sonnhammer. 2005. "Kalign--an Accurate and Fast Multiple Sequence Alignment Algorithm." *BMC Bioinformatics* 6: 298.
Lewontin, R. C., and J. L. Hubby. 1966. "A Molecular Approach to the Study of Genic Heterozygosity in Natural Populations. II. Amount of Variation and Degree of Heterozygosity in Natural Populations of Drosophila Pseudoobscura." *Genetics* 54 (2): 595–609.
Liang, Richard H., Theresa Mo, Winnie Dong, Guinevere Q. Lee, Luke C. Swenson, Rosemary M. McCloskey, Conan K. Woods, et al. 2014. "Theoretical and Experimental Assessment of Degenerate Primer Tagging in Ultra-Deep Applications of next-Generation Sequencing." *Nucleic Acids Research* 42 (12). https://doi.org/10.1093/nar/gku355.
Li, H. 2013. "Aligning Sequence Reads, Clone Sequences and Assembly Contigs with BWA-MEM. arXiv Prepr. arXiv 0."
Li, H., and R. Durbin. 2009. "Fast and Accurate Short Read Alignment with Burrows-Wheeler Transform." *Bioinformatics* , May, 1–7.
Li, Heng. 2013. "Aligning Sequence Reads , Clone Sequences and Assembly Contigs with BWA-MEM." *arXiv.org* 00 (00): 1–3.
Li, Heng. 2014a. "On HiSeq X10 Base Quality." *Heng Li's Blog* (blog). 2014. https://lh3.github.io/2014/11/03/on-hiseq-x10-base-quality.
Li, Heng. 2014b. "Toward Better Understanding of Artifacts in Variant Calling from High-Coverage Samples." *Bioinformatics* 30 (20): 2843–51.
Li, H., B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin. 2009. "The Sequence Alignment/Map Format and SAMtools." *Bioinformatics* 25 (16): 2078–79.
Li, Mingkun, Roland Schröder, Shengyu Ni, Burkhard Madea, and Mark Stoneking. 2015. "Extensive Tissue-Related and Allele-Related mtDNA Heteroplasmy Suggests Positive Selection for Somatic Mutations." *Proceedings of the National Academy of Sciences of the United States of America* 112 (8): 2491–96.
Li, Mingkun, and Mark Stoneking. 2012. "A New Approach for Detecting Low-Level Mutations in next-Generation Sequence Data." *Genome Biology* 13 (5): R34.

Li, R., Chang Yu, Yingrui Li, T-W Lam, S-M Yiu, Karsten Kristiansen, and Jun Wang. 2009. “SOAP2: An Improved Ultrafast Tool for Short Read Alignment.” *Bioinformatics* 25 (15): 1966–67.

Liti, Gianni, David M. Carter, Alan M. Moses, Jonas Warringer, Leopold Parts, Stephen A. James, Robert P. Davey, et al. 2009. “Population Genomics of Domestic and Wild Yeasts.” *Nature* 458 (7236): 337–41.

Li, Yingrui, Nicolas Vinckenbosch, Geng Tian, Emilia Huerta-Sanchez, Tao Jiang, Hui Jiang, Anders Albrechtsen, et al. 2010. “Resequencing of 200 Human Exomes Identifies an Excess of Low-Frequency Non-Synonymous Coding Variants.” *Nature Genetics* 42 (11): 969–72.

Lou, D. I., J. A. Hussmann, R. M. McBee, A. Acevedo, R. Andino, W. H. Press, and S. L. Sawyer. 2013. “High-Throughput DNA Sequencing Errors Are Reduced by Orders of Magnitude Using Circle Sequencing.” *Proceedings of the National Academy of Sciences* 110 (49): 19872–77.

MacArthur, Jacqueline, Emily Bowler, Maria Cerezo, Laurent Gil, Peggy Hall, Emma Hastings, Heather Junkins, et al. 2017. “The New NHGRI-EBI Catalog of Published Genome-Wide Association Studies (GWAS Catalog).” *Nucleic Acids Research* 45 (D1): D896–901.

MacKenzie, David, Jim Meyering, François Pinard, Karl Berry, Brian Youmans, and Richard Stallman. 2018. *GNU Coreutils* (version 8.30). Free Software Foundation. https://www.gnu.org/software/coreutils/manual/.

Manolio, Teri A., Francis S. Collins, Nancy J. Cox, David B. Goldstein, Lucia A. Hindorff, David J. Hunter, Mark I. McCarthy, et al. 2009. “Finding the Missing Heritability of Complex Diseases.” *Nature* 461 (7265): 747–53.

Margulies, Marcel, Michael Egholm, William E. Altman, Said Attiya, Joel S. Bader, Lisa A. Bemben, Jan Berka, et al. 2005. “Genome Sequencing in Microfabricated High-Density Picolitre Reactors.” *Nature*, July. https://doi.org/10.1038/nature03959.

Ma, Xiaotu, Ying Shao, Liqing Tian, Diane A. Flasch, Heather L. Mulder, Michael N. Edmonson, Yu Liu, et al. 2019. “Analysis of Error Profiles in Deep next-Generation Sequencing Data.” *Genome Biology* 20 (1): 1–15.

May, Ali, Sanne Abeln, Mark J. Buijs, Jaap Heringa, Wim Crielaard, and Bernd W. Brandt. 2015. “NGS-Eval: NGS Error Analysis and Novel Sequence VAriant Detection tooL.” *Nucleic Acids Research* 43 (W1): W301–5.

Meacham, Frazer, Dario Boffelli, Joseph Dhahbi, David I. K. Martin, Meromit Singer, and Lior Pachter. 2011. “Identification and Correction of Systematic Error in High-Throughput Sequence Data.” *BMC Bioinformatics* 12 (November): 451.

Mei, Han, Barbara Arbeithuber, Marzia A. Cremona, Michael DeGiorgio, and Anton Nekrutenko. 2019. “A High-Resolution View of Adaptive Event Dynamics in a Plasmid.” Edited by Wen-Hsiung Li. *Genome Biology and Evolution* 11 (10): 3022–34.

Mei, Han, Barbara Arbeithuber, Marzia Cremona, Michael DeGeorgio, and Anton Nekrutenko. 2018. “A High Resolution View of Adaptive Events.” *bioRxiv*. https://doi.org/10.1101/429175.

Metzker, Michael L. 2010. “Sequencing Technologies {\textemdash} the next Generation.” *Nature Reviews. Genetics* 11 (1): 31–46.

Miotke, Laura, Billy T. Lau, Rowza T. Rumma, and Hanlee P. Ji. 2014. “High Sensitivity Detection and Quantitation of DNA Copy Number and Single Nucleotide Variants with Single Color Droplet Digital PCR.” *Analytical Chemistry*, February, 140212095645005.

Mora, Camilo, Derek P. Tittensor, Sina Adl, Alastair G. B. Simpson, and Boris Worm. 2011. “How Many Species Are There on Earth and in the Ocean?” *PLoS Biology* 9 (8): e1001127.

Muth, Robert, and Udi Manber. 1996. “Approximate Multiple String Search.” In *Combinatorial Pattern Matching*, 75–86. Springer Berlin Heidelberg.

Nakamura, Kensuke, Taku Oshima, Takuya Morimoto, Shun Ikeda, Hirofumi Yoshikawa, Yuh Shiwa, Shu Ishikawa, et al. 2011. “Sequence-Specific Error Profile of Illumina Sequencers.” *Nucleic Acids*

Research 39 (13): e90.
Nakazato, Takeru, Tazro Ohta, and Hidemasa Bono. 2013. “Experimental Design-Based Functional Mining and Characterization of High-Throughput Sequencing Data in the Sequence Read Archive.” *PloS One* 8 (10). https://doi.org/10.1371/journal.pone.0077910.
Newman, Aaron M., Alexander F. Lovejoy, Daniel M. Klass, David M. Kurtz, Jacob J. Chabon, Florian Scherer, Henning Stehr, et al. 2016. “Integrated Digital Error Suppression for Improved Detection of Circulating Tumor DNA.” *Nature Biotechnology* 34 (5): 547–55.
Ng, Sarah B., Kati J. Buckingham, Choli Lee, Abigail W. Bigham, Holly K. Tabor, Karin M. Dent, Chad D. Huff, et al. 2010. “Exome Sequencing Identifies the Cause of a Mendelian Disorder.” *Nature Genetics* 42 (1): 30–35.
Payne, Alexander, Nadine Holmes, Vardhman Rakyan, and Matthew Loose. 2019. “Bulkvis: A Graphical Viewer for Oxford Nanopore Bulk FAST5 Files.” *Bioinformatics* 35 (13): 2193–98.
Pe’er, Itsik, Paul I. W. de Bakker, Julian Maller, Roman Yelensky, David Altshuler, and Mark J. Daly. 2006. “Evaluating and Improving Power in Whole-Genome Association Studies Using Fixed Marker Sets.” *Nature Genetics* 38 (6): 663–67.
Pelt-Verkuil, E. van, W. B. van Leeuwen, and R. te Witt. 2019. *Molecular Diagnostics: Part 1: Technical Backgrounds and Quality Aspects*. Springer.
Pfeiffer, Franziska, Carsten Gröber, Michael Blank, Kristian Händler, Marc Beyer, Joachim L. Schultze, and Günter Mayer. 2018. “Systematic Evaluation of Error Rates and Causes in Short Samples in next-Generation Sequencing.” *Scientific Reports* 8 (1): 10950.
Potapov, Vladimir, and Jennifer L. Ong. 2017. “Examining Sources of Error in PCR by Single-Molecule Sequencing.” *PloS One* 12 (7): 1–19.
Quail, Michael A., Miriam Smith, Paul Coupland, Thomas D. Otto, Simon R. Harris, Thomas R. Connor, Anna Bertoni, Harold P. Swerdlow, and Yong Gu. 2012. “A Tale of Three next Generation Sequencing Platforms: Comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq Sequencers.” *BMC Genomics* 13 (July): 341.
Rebolledo Jaramillo, Boris, Marcia Shu-Wei Su, Nicholas Stoler, Jennifer A. McElhoe, Benjamin Dickins, Daniel Blankenberg, Thorfinn S. Korneliussen, et al. 2014. “Maternal Age Effect and Severe Germ-Line Bottleneck in the Inheritance of Human Mitochondrial DNA.” *Proceedings of the National Academy of Sciences of the United States of America* 111 (43): 15474–79.
Ross, Michael G., Carsten Russ, Maura Costello, Andrew Hollinger, Niall J. Lennon, Ryan Hegarty, Chad Nusbaum, and David B. Jaffe. 2013. “Characterizing and Measuring Bias in Sequence Data.” *Genome Biology* 14 (5): R51.
Ruan, Yaoping, and Vivek Pai. 2004. “Making the ‘Box’ Transparent: System Call Performance as a First-Class Result.” *Proceedings of the USENIX ... Annual Technical Conference. USENIX Technical Conference*. https://www.usenix.org/legacy/event/usenix04/tech/general/full_papers/ruan/ruan_html/.
Salk, Jesse J., Michael W. Schmitt, and Lawrence A. Loeb. 2018. “Enhancing the Accuracy of next-Generation Sequencing for Detecting Rare and Subclonal Mutations.” *Nature Reviews. Genetics*. https://doi.org/10.1038/nrg.2017.117.
Schirmer, Melanie, Rosalinda D’Amore, Umer Z. Ijaz, Neil Hall, and Christopher Quince. 2016. “Illumina Error Profiles: Resolving Fine-Scale Variation in Metagenomic Sequencing Data.” *BMC Bioinformatics* 17 (1): 125.
Schirmer, M., U. Z. Ijaz, R. D’Amore, N. Hall, W. T. Sloan, and C. Quince. 2015. “Insight into Biases and Sequencing Errors for Amplicon Sequencing with the Illumina MiSeq Platform.” *Nucleic Acids Research* 43 (6): 1–16.
Schmitt, Michael W., Edward J. Fox, Marc J. Prindle, Kate S. Reid-Bayliss, Lawrence D. True, Jerald P. Radich, and Lawrence A. Loeb. 2015. “Sequencing Small Genomic Targets with High Efficiency and

Extreme Accuracy." *Nature Methods* 12 (5): 423–25.
Schmitt, M. W., S. R. Kennedy, J. J. Salk, E. J. Fox, J. B. Hiatt, and L. A. Loeb. 2012. "Detection of Ultra-Rare Mutations by next-Generation Sequencing." *Proceedings of the National Academy of Sciences* 109 (36): 14508–13.
Shugay, Mikhail, Olga V. Britanova, Ekaterina M. Merzlyak, Maria A. Turchaninova, Ilgar Z. Mamedov, Timur R. Tuganbaev, Dmitriy A. Bolotin, et al. 2014. "Towards Error-Free Profiling of Immune Repertoires." *Nature Methods* 11 (6): 653–55.
Shugay, Mikhail, Andrew R. Zaretsky, Dmitriy A. Shagin, Irina A. Shagina, Ivan A. Volchenkov, Andrew A. Shelenkov, Mikhail Y. Lebedin, Dmitriy V. Bagaev, Sergey Lukyanov, and Dmitriy M. Chudakov. 2017. "MAGERI: Computational Pipeline for Molecular-Barcoded Targeted Resequencing." *PLoS Computational Biology* 13 (5): 13–17.
Smith, T. F., and M. S. Waterman. 1981. "Identification of Common Molecular Subsequences." *Journal of Molecular Biology* 147 (1): 195–97.
Smith, Tom, Andreas Heger, and Ian Sudbery. 2017. "UMI-Tools: Modeling Sequencing Errors in Unique Molecular Identifiers to Improve Quantification Accuracy." *Genome Research* 27 (3): 491–99.
Smoller, Jordan W., and Christine T. Finn. 2003. "Family, Twin, and Adoption Studies of Bipolar Disorder." *American Journal of Medical Genetics. Part C, Seminars in Medical Genetics* 123C (1): 48–58.
Stacey, David, Toni-Kim Clarke, and Gunter Schumann. 2009. "The Genetics of Alcoholism." *Current Psychiatry Reports* 11 (5): 364–69.
Stevenson, Kraig R., Joseph D. Coolon, and Patricia J. Wittkopp. 2013. "Sources of Bias in Measures of Allele-Specific Expression Derived from RNA-Seq Data Aligned to a Single Reference Genome." *BMC Genomics* 14 (1): 536.
Stewart, Robert D., Marc D. Auffret, Amanda Warr, Andrew H. Wiser, Maximilian O. Press, Kyle W. Langford, Ivan Liachko, et al. 2018. "Assembly of 913 Microbial Genomes from Metagenomic Sequencing of the Cow Rumen." *Nature Communications* 9 (1): 870.
Stoler, Nicholas, Barbara Arbeithuber, Wilfried Guiblet, Kateryna D. Makova, and Anton Nekrutenko. 2016. "Streamlined Analysis of Duplex Sequencing Data with Du Novo." *Genome Biology* 17 (1): 180.
Sullivan, Patrick F., Kenneth S. Kendler, and Michael C. Neale. 2003. "Schizophrenia as a Complex Trait: Evidence from a Meta-Analysis of Twin Studies." *Archives of General Psychiatry* 60 (12): 1187–92.
Syvänen, A. C. 2001. "Accessing Genetic Variation: Genotyping Single Nucleotide Polymorphisms." *Nature Reviews. Genetics* 2 (12): 930–42.
Trapnell, C., L. Pachter, and S. L. Salzberg. 2009. "TopHat: Discovering Splice Junctions with RNA-Seq." *Bioinformatics* 25 (9): 1105–11.
Travers, Kevin J., Chen-Shan Chin, David R. Rank, John S. Eid, and Stephen W. Turner. 2010. "A Flexible and Efficient Template Format for Circular Consensus Sequencing and SNP Detection." *Nucleic Acids Research* 38 (15): e159.
Wallace, Douglas C., and Dimitra Chalkia. 2013. "Mitochondrial DNA Genetics and the Heteroplasmy Conundrum in Evolution and Disease." *Cold Spring Harbor Perspectives in Biology* 5 (11): a021220–a021220.
Walley, Andrew J., Alexandra I. F. Blakemore, and Philippe Froguel. 2006. "Genetics of Obesity and the Prediction of Risk for Health." *Human Molecular Genetics* 15 Spec No 2 (October): R124–30.
Whiteford, Nava, Tom Skelly, Christina Curtis, Matt E. Ritchie, Andrea Löhr, Alexander Wait Zaranek, Irina Abnizova, and Clive Brown. 2009. "Swift: Primary Data Analysis for the Illumina Solexa Sequencing Platform." *Bioinformatics* 25 (17): 2194–99.
Wilbur, W., and D. Lipman. 1984. "The Context Dependent Comparison of Biological Sequences." *SIAM Journal on Applied Mathematics* 44 (3): 557–67.

Wu, Sun, and Udi Manber. 1992. “Fast Text Searching: Allowing Errors.” *Communications of the ACM* 35 (10): 83–91.

Xu, Chang, Xiujing Gu, Raghavendra Padmanabhan, Zhong Wu, Quan Peng, John DiCarlo, and Yexun Wang. 2018. “smCounter2: An Accurate Low-Frequency Variant Caller for Targeted Sequencing Data with Unique Molecular Identifiers.” *bioRxiv*, 281659.

Zhang, Pan, David C. Samuels, Brian Lehmann, Thomas Stricker, Jennifer Pietenpol, Yu Shyr, and Yan Guo. 2016. “Mitochondria Sequence Mapping Strategies and Practicability of Mitochondria Variant Detection from Exome and RNA Sequencing Data.” *Briefings in Bioinformatics* 17 (2): 224–32.

Appendix A

Distribution of SRA *E. coli* runs by sequencing instrument

Table 4.S1 The distribution of SRA *E. coli* runs by sequencing instrument as of 31 August 2020.

Number of runs by sequencing machine for all *E. coli* data in the SRA as of 2020-08-31		
Platform	Model	Runs
ILLUMINA	Illumina MiSeq	75118
ILLUMINA	Illumina HiSeq 2500	36034
ILLUMINA	Illumina HiSeq 2000	18483
ILLUMINA	HiSeq X Ten	16921
ILLUMINA	NextSeq 500	16471
ILLUMINA	Illumina HiSeq 4000	7339
ILLUMINA	Illumina Genome Analyzer IIx	1582
ILLUMINA	Illumina NovaSeq 6000	1375
ILLUMINA	Illumina HiSeq X Ten	1257
ILLUMINA	NextSeq 550	1107
ILLUMINA	Illumina Genome Analyzer II	839
ILLUMINA	unspecified	619
ILLUMINA	HiSeq X Five	560
ILLUMINA	Illumina HiSeq 1500	443
ILLUMINA	Illumina HiSeq 3000	362
ILLUMINA	Illumina HiSeq 1000	291
ILLUMINA	Illumina MiniSeq	237
ILLUMINA	Illumina Genome Analyzer	210
ILLUMINA	Illumina HiScanSQ	44
ILLUMINA	Illumina iSeq 100	14
ILLUMINA	TOTAL	179306
OXFORD_NANOPORE	GridION	901
OXFORD_NANOPORE	MinION	646
OXFORD_NANOPORE	TOTAL	1547
PACBIO_SMRT	PacBio RS	1425

PACBIO_SMRT	PacBio RS II	564
PACBIO_SMRT	Sequel	158
PACBIO_SMRT	Sequel II	7
PACBIO_SMRT	TOTAL	2154
ION_TORRENT	Ion Torrent PGM	281
ION_TORRENT	Ion Torrent Proton	184
ION_TORRENT	Ion S5 XL	31
ION_TORRENT	Ion S5	24
ION_TORRENT	Ion Torrent S5	7
ION_TORRENT	Ion Torrent S5 XL	6
ION_TORRENT	TOTAL	533
BGISEQ	BGISEQ-500	68
BGISEQ	TOTAL	68
LS454	454 GS FLX Titanium	751
LS454	454 GS FLX	305
LS454	454 GS FLX+	54
LS454	454 GS 20	27
LS454	454 GS Junior	25
LS454	454 GS	2
LS454	TOTAL	1164
ABI_SOLID	AB 5500xl Genetic Analyzer	229
ABI_SOLID	AB SOLiD 4 System	85
ABI_SOLID	AB 5500 Genetic Analyzer	60
ABI_SOLID	AB SOLiD System 3.0	39
ABI_SOLID	AB SOLiD System	5
ABI_SOLID	AB SOLiD System 2.0	1
ABI_SOLID	AB SOLiD 4hq System	1
ABI_SOLID	TOTAL	420
CAPILLARY	AB 3730xL Genetic Analyzer	761
CAPILLARY	AB 310 Genetic Analyzer	66
CAPILLARY	AB 3500 Genetic Analyzer	2
CAPILLARY	AB 3730 Genetic Analyzer	1
CAPILLARY	TOTAL	830
TOTAL	TOTAL	186022

Appendix B

Submitter metadata

Table 4.S2 The submitter metadata for every unique group represented in this analysis. Groups were defined by the combination of the center, lab, and contact metadata fields. Included are groups which appear in the set of runs with a sufficient number of overlapping bases in the center bin, as defined in the methods. For brevity, some words are abbreviated, and long metadata values are truncated. The full values are available using the accession numbers provided from example runs.

Submitters represented in the analysis: SRA runs with sufficient overlap						
	Instrument	Example run	Count	Center	Lab	Contact
1	MiSeq	ERR2868175	2		Euro. Nuc. Archive	Euro. Nuc. Archive
2	MiSeq	ERR2686034	2	Inst. of Tech., Univ. of Tartu	Euro. Nuc. Archive	Euro. Nuc. Archive
3	MiSeq	DRR021342	1	RIKEN_QBC	Lab. for multiscale biosystem dynamics	Lab. for multiscale biosystem dynamics
4	MiSeq	DRR036001	1	WASEDA	Haruko Takeyama lab	Haruko Takeyama lab
5	MiSeq	DRR058068	1	NIG	Microbial Genetics Lab., Genetics Strains Re	Microbial Genetics Lab., Genetics Strains Re
6	MiSeq	DRR065947	1	NIID	Lab. of Bacterial Genomics, Pathogen Genomic	Lab. of Bacterial Genomics, Pathogen Genomic
7	MiSeq	DRR066639	1	TSUKUBA	Environmental Molecular Microbiology, Faculty of l	Environmental Molecular Microbiology, Faculty of l
8	MiSeq	DRR075635	1	NCGM	Dept. of Infectious Diseases, Nat. Center	Dept. of Infectious Diseases, Nat. Center
9	MiSeq	DRR075951	1	RIKEN_QBC	Lab. for Multiscale Biosystem Dynamics	Lab. for Multiscale Biosystem Dynamics
10	MiSeq	DRR077337	1	NIID	Dept. of Bacteriology, Nat. Inst. of	Dept. of Bacteriology, Nat. Inst. of
11	MiSeq	DRR078802	1	WASEDA	Haruko Takeyama	Haruko Takeyama
12	MiSeq	DRR086872	1	NCGM	Dept. of Infectious Diseases	Dept. of Infectious Diseases
13	MiSeq	DRR092871	1	KYOTO_GM	Clinical Lab. Medicine, Kyoto Univ. Gra	Clinical Lab. Medicine, Kyoto Univ. Gra
14	MiSeq	DRR093006	1	TUM-MID	Dept. of Microbiology and Infectious Diseases	Dept. of Microbiology and Infectious Diseases
15	MiSeq	DRR099822	1	OSAKA_IST	Metabolic Engineering Lab, Dept. of Bioinform	Metabolic Engineering Lab, Dept. of Bioinform
16	MiSeq	DRR100985	1	KYOTO_ENG	Environmental Risk Assessment Group, Dept. of	Environmental Risk Assessment Group, Dept. of
17	MiSeq	DRR120522	1	JAMSTEC	Research and Development Center for Marine Bioscie	Research and Development Center for Marine Bioscie
18	MiSeq	DRR138475	1	OSAKA_HOSP	Division of Infection Control and Prevention	Division of Infection Control and Prevention
19	MiSeq	DRR159855	1	KYUSHU	Microbiology, Graduate School of Medical Sciences,	Microbiology, Graduate School of Medical Sciences,
20	MiSeq	ERR1095123	1	Univ. Of Aberdeen	Euro. Nuc. Archive	Euro. Nuc. Archive
21	MiSeq	ERR1149378	1	The Roslin Inst.	Euro. Nuc. Archive	Euro. Nuc. Archive
22	MiSeq	ERR1474546	1	Inst. For Molecular Bioscience Uq	Euro. Nuc. Archive	Euro. Nuc. Archive
23	MiSeq	ERR1544896	1	Statens Serum Institut	Euro. Nuc. Archive	Euro. Nuc. Archive
24	MiSeq	ERR1551798	1	Utrecht Univ.	Euro. Nuc. Archive	Euro. Nuc. Archive
25	MiSeq	ERR1617848	1	Amphia Hospital	Euro. Nuc. Archive	Euro. Nuc. Archive
26	MiSeq	ERR1725935	1	The Westmead Inst. for Medical Research and Ma	Euro. Nuc. Archive	Euro. Nuc. Archive
27	MiSeq	ERR1755174	1	RJ	Euro. Nuc. Archive	Euro. Nuc. Archive
28	MiSeq	ERR1755547	1	Centre for Genomic Epidemiology, Nat. Food Ins	Euro. Nuc. Archive	Euro. Nuc. Archive
29	MiSeq	ERR1857335	1	Inst. of Microbiology and Infection School of	Euro. Nuc. Archive	Euro. Nuc. Archive
30	MiSeq	ERR1912619	1	Dept. of Paediatric Medicine, Oslo Univ.	Euro. Nuc. Archive	Euro. Nuc. Archive
31	MiSeq	ERR2019972	1	Istituto Superiore Di Sanita	Euro. Nuc. Archive	Euro. Nuc. Archive
32	MiSeq	ERR2044810	1	Scientific Insitute of Public Health	Euro. Nuc. Archive	Euro. Nuc. Archive
33	MiSeq	ERR2184786	1	Norwegian Inst. Of Public Health	Euro. Nuc. Archive	Euro. Nuc. Archive
34	MiSeq	ERR2204731	1	Univ. Medical Center Groningen	Euro. Nuc. Archive	Euro. Nuc. Archive
35	MiSeq	ERR2205887	1	Robert Koch-Inst.	Euro. Nuc. Archive	Euro. Nuc. Archive
36	MiSeq	ERR2223700	1	NVRI	Euro. Nuc. Archive	Euro. Nuc. Archive
37	MiSeq	ERR2531853	1	Univ. Of Nottingham	Euro. Nuc. Archive	Euro. Nuc. Archive
38	MiSeq	ERR2540224	1	IZSLT (Istituto Zooprofilattico Sperimentale del L	Euro. Nuc. Archive	Euro. Nuc. Archive

39	MiSeq	ERR2602229	1	Univ. Of Helsinki	Euro. Nuc. Archive	Euro. Nuc. Archive
40	MiSeq	ERR2703302	1	Univ. Of Copenhagen	Euro. Nuc. Archive	Euro. Nuc. Archive
41	MiSeq	ERR271292	1	SC		
42	MiSeq	ERR2775915	1	RIVM - Nat. Inst. for Public Health and th	Euro. Nuc. Archive	Euro. Nuc. Archive
43	MiSeq	ERR2796933	1	Biozentrum, Univ. Of Basel	Euro. Nuc. Archive	Euro. Nuc. Archive
44	MiSeq	ERR3140431	1	Dept. of Medical Microbiology, Stavanger Univ	Euro. Nuc. Archive	Euro. Nuc. Archive
45	MiSeq	ERR3140739	1	Univ. Of M?Nster	Euro. Nuc. Archive	Euro. Nuc. Archive
46	MiSeq	ERR3317749	1	Robert Koch Institut	Euro. Nuc. Archive	Euro. Nuc. Archive
47	MiSeq	ERR3449876	1	UCD-CFS	Euro. Nuc. Archive	Euro. Nuc. Archive
48	MiSeq	ERR3510698	1	Centre for Translational Microbiome Research (CTMR	Euro. Nuc. Archive	Euro. Nuc. Archive
49	MiSeq	ERR3537789	1	Wadsworth Center	Euro. Nuc. Archive	Euro. Nuc. Archive
50	MiSeq	ERR355891	1	IFH_MS	Inst. Hygiene, sequencing laboratory	Inst. Hygiene, sequencing laboratory
51	MiSeq	ERR580964	1	WWU		
52	MiSeq	ERR654983	1	Delft Univ. Of Tech.		
53	MiSeq	ERR658230	1	Statens Serum Institut		
54	MiSeq	ERR738830	1	CGS-GL		
55	MiSeq	ERR883742	1	Univ. Hospital Muenster	Euro. Nuc. Archive	Euro. Nuc. Archive
56	MiSeq	ERR968337	1	Trinity College	Euro. Nuc. Archive	Euro. Nuc. Archive
57	MiSeq	SRR10002669	1	Univ. of Nebraska Medical Center	Pathology and Microbiology	Baha Abdalhamid
58	MiSeq	SRR10013887	1	Nat. Microbiology Lab. at Guelph	Public Health Agency of Canada	John H. E. Nash
59	MiSeq	SRR10057900	1	Univ. of Sao Paulo	Intitute of Biomedical Science	Nilton Lincopan
60	MiSeq	SRR10076548	1	USDA-ARS Nat. Animal Disease Center	VPRU	Sarah Shore
61	MiSeq	SRR10099940	1	Food and Drug Administration-College Park	ORS/DM/MMSB	Narjol Gonzalez-Escalona
62	MiSeq	SRR1030395	1	Instituto Gulbenkian Ciencia	Evolutionary Biology	Ana Sousa
63	MiSeq	SRR1186325	1	Arizona State Public Health Lab		FDA/CFSAN:justin payne
64	MiSeq	SRR1198948	1	Minnesota Dept. of Health		FDA/CFSAN:justin payne
65	MiSeq	SRR1301200	1	BaseSpace		Michael Valentine
66	MiSeq	SRR1509643	1	Univ. of California San Diego		Haythem Latif
67	MiSeq	SRR1554984	1	ithree institute		Ethan Wyrsch
68	MiSeq	SRR1570182	1	EDLB-CDC	Enteric Diseases Lab. Branch	Rebecca Lindsey
69	MiSeq	SRR1613986	1	SUDB-QL	Quake	Iwijn De Vlaminck
70	MiSeq	SRR1731953	1	Harvard Medical School	George Church Lab	Stephanie Yaung
71	MiSeq	SRR1781795	1	Canadian Food Inspection Agency	Ottawa Carling Lab.	James Markell
72	MiSeq	SRR1795508	1	EDLB-CDC	Enteric Diseases Lab. Branch	Darlene Wagner
73	MiSeq	SRR1867741	1	Alberta Agriculture and Rural Developmen		Lisa Tymensen
74	MiSeq	SRR1909695	1	Alberta Agriculture and Rural Development	Irrigation and Farm Water Division	Lisa Tymensen
75	MiSeq	SRR1947852	1	Michigan State Univ.	Microbial Evolution Lab.	Pallavi Singh
76	MiSeq	SRR2001748	1	MILAB		Bolotin Dmitry
77	MiSeq	SRR2102295	1	USDA FSIS	USDA FSIS	Glenn Tillman

78	MiSeq	SRR2176280	1	FDA/CFSAN	ORS/DM/MMSB	ORS/DM/MMSB
79	MiSeq	SRR2239631	1	FDA	CVM	FDA CVM
80	MiSeq	SRR2244250	1	Univ. of KwaZulu Natal	John	John Osei Sekyere
81	MiSeq	SRR2409967	1	Instituto Gulbenkian de Ciência	Evolutionary Biology	Jorge de Sousa
82	MiSeq	SRR2420884	1	IGC	Evolutionary Biology	Ana Sousa
83	MiSeq	SRR2544681	1	FDA		
84	MiSeq	SRR2601697	1	Instituto Gulbenkian de Ciência	Evolutionary Biology	João Batista
85	MiSeq	SRR2724108	1	Norwegian Nat. Advisory Unit on Detection of A	Univ. Hospital of North Norway, Dept. of Micr	Univ. Hospital of North Norway, Dept. of Micr
86	MiSeq	SRR2761515	1	The Univ. of Texas at Austin	Molecular Biosciences	Molecular Biosciences
87	MiSeq	SRR2767734	1	Anses	AVB	AVB
88	MiSeq	SRR2921970	1	GEO		Gene Expression Omnibus (GEO), NCBI, NLM, NIH, htt
89	MiSeq	SRR2970676	1	Univ. of Oxford	Nuffield Dept. of Medicine	Nuffield Dept. of Medicine
90	MiSeq	SRR2976832	1	Brigham & Women's Hospital		
91	MiSeq	SRR3018390	1	FDA/CFSAN	Cfsan-Ors-Dm-Mmsb	Cfsan-Ors-Dm-Mmsb
92	MiSeq	SRR3040688	1	Alberta Provincial Lab. for Public Health	Bacterial Typing Unit	Bacterial Typing Unit
93	MiSeq	SRR3144118	1	Harvard Univ.	Xie	David Lee
94	MiSeq	SRR3195839	1	United State Dept. of Agriculture	Meat Safety and Quality Research Unit	Meat Safety and Quality Research Unit
95	MiSeq	SRR3202030	1	Middle Tennessee State Univ.		Rebecca Seipelt-Thiemann
96	MiSeq	SRR3241811	1	USDA-FSIS		Glenn Tillman
97	MiSeq	SRR3242018	1	CDC-CEMB		
98	MiSeq	SRR3473864	1	Instituto Gulbenkian de Ciencia	Evolutionary Biology Lab	Evolutionary Biology Lab
99	MiSeq	SRR3479287	1	Univ. of Colorado - Boulder	Chemical and Biological Engineering	Anushree Chatterjee
100	MiSeq	SRR3480971	1	Kenyon College	Dept. of Biology	Dept. of Biology
101	MiSeq	SRR3584989	1	Walter Reed Army Inst. of Research	Multidrug Resistant Organism Repository and Survei	Multidrug Resistant Organism Repository and Survei
102	MiSeq	SRR3608507	1	CeBiTec Bielefeld	Microbiology of Sustainable Energy Production	Halina Tegetmeyer
103	MiSeq	SRR3715289	1	edlb-cdc	Enteric Diseases Lab. Branch	Enteric Diseases Lab. Branch
104	MiSeq	SRR3742542	1	Health Canada	Bureau of Microbial Hazards	Jennifer Ronholm
105	MiSeq	SRR3929023	1	Emory Univ.	Gerardo Lab	Tom de Man
106	MiSeq	SRR3982215	1	CDC-DHQP-CEMB		
107	MiSeq	SRR3999088	1	ModMedMicOxford		
108	MiSeq	SRR4036810	1	TGen North		
109	MiSeq	SRR4111264	1	Arizona State Univ.	School of Life Sciences	School of Life Sciences
110	MiSeq	SRR4191396	1	Univ. of Tech. Sydney	Molecular & Genomic Microbiology, The ithree inst	Ricaurte Alejandro Marquez-Ortiz
111	MiSeq	SRR4241821	1	Australian Nat. Univ.	Evolution, Ecology and Genetics	Belinda Lallawmkimi Vangchhia
112	MiSeq	SRR4289227	1	Institue of Microbiology, Chinese Academy of Scien	CAS Key Lab. of Pathogenic Microbiology and	Yuqing Feng
113	MiSeq	SRR4299799	1	edlb-cdc	Enteric Diseases Lab. Branch	edlb-cdc service
114	MiSeq	SRR4301505	1	United State Dept. of Agriculture	Meat Safety and Quality Research Unit	James L Bono
115	MiSeq	SRR4302138	1	Tan Tock Seng Hospital	IIDE	Wei Xin Khong
116	MiSeq	SRR4997082	1	Technical Univ. of Denmark, Nat. Veterina	Section for Bacteriology, Pathology and Parasitolo	Troels Ronco

117	MiSeq	SRR5168391	1	NISC		Morgan Park
118	MiSeq	SRR5237400	1	Brigham & Women's Hospital	Pathology	Xiaomin Zhao
119	MiSeq	SRR5252311	1	The Univ. of Texas at Austin	Molecular Biosciences	Jeffrey E Barrick
120	MiSeq	SRR5278308	1	Agriculture and Agri-Food Canada	GFRD	Muhammad Attiq Rehman
121	MiSeq	SRR5328644	1	South China Agricultural Univ.		Xing-Ping Li
122	MiSeq	SRR5386010	1	North Dakota State Univ.	Microbiological Sciences	Teresa Bergholz
123	MiSeq	SRR5430177	1	UC San Diego	Bioengineering	Jonathan Monk
124	MiSeq	SRR5470047	1	CFSAN		fda service
125	MiSeq	SRR5481546	1	California Dept. of Public Health	Microbial Disease Lab. Program	Varvara K Kozyreva
126	MiSeq	SRR5493709	1	Univ. of Canterbury	School of Biological Sciences	Nicole Elizabeth Wheeler
127	MiSeq	SRR5512150	1	The Australian Nat. Univ.	Research School of Biology	David Michael Gordon
128	MiSeq	SRR5527242	1	Instituto Gulbenkian de Ciencia	Evolutionary Biology	Jorge Moura de Sousa
129	MiSeq	SRR5527727	1	NNF-CFB DTU	NNF-CFB-DTU	Rachel Amanda Hickman
130	MiSeq	SRR5629433	1	USDA-ARS-Nat. Animal Disease Center	Infectious Bacterial Diseases	Tyler C Thacker
131	MiSeq	SRR5666068	1	Korea Research Inst. of Bioscience and Biotech	Infectious Disease Research Center	Haeyoung Jeong
132	MiSeq	SRR5666624	1	Centers for Disease Control and Prevention	Clinical and Environmental Microbiology Branch	Adrian Lawsin
133	MiSeq	SRR5805010	1	NIH Clinical Center	Dept. of Lab. Medicine	Pavel P Khil
134	MiSeq	SRR5821465	1	Univ. of Queensland	School of Chemistry and Molecular Biosciences	Brian Forde
135	MiSeq	SRR587217	1	MFPL	CeBiTec	Sebastian Jünemann
136	MiSeq	SRR5886045	1	mEpilab, Massey Univ.	The Inst. of Veterinary, Animal and Biomedical	Andrew Springer Browne
137	MiSeq	SRR5927226	1	USDA	ARS	Gian Marco Baranzoni
138	MiSeq	SRR5936527	1	Univ. of Queensland	UQ Centre for Clinical Research	Patrick Harris
139	MiSeq	SRR6118146	1	U.S. Dept. of Agriculture	Agricultural Research Service: Eastern Regional Re	Erin Reichenberger
140	MiSeq	SRR6123469	1	Institut for Medizinische Mikrobiologie	Microbilogy	Hiren Ghosh
141	MiSeq	SRR6154935	1	Canadian Food Inspection Agency	Research and Development, Science Branch	Catherine Carrillo
142	MiSeq	SRR6197658	1	Technical Univ. of Denmark	Novo Nordisk Foundation Center for Biosustainabili	Lejla Imamovic
143	MiSeq	SRR6224590	1	Univ. of Muenster	Inst. of Hygiene	Ulrich Dobrindt
144	MiSeq	SRR6281327	1	CNRS	IBPC (UMR7099)	Federica Angius
145	MiSeq	SRR6321260	1	NHS Lothian	Scottish E.coliO157/STEC Reference Lab.	Anne Holmes
146	MiSeq	SRR6376576	1	USGS	Alaska Science Center	Christina Ahlstrom
147	MiSeq	SRR6409916	1	Univ. College Dublin	Veterinary Medicine	Damien Farrell
148	MiSeq	SRR6451073	1	Instituto Gulbenkian de Ciencia	Evolutionary Biology	Joao Barroso-Batista
149	MiSeq	SRR6456913	1	UWML	Miller Lab.	Eli Weiss
150	MiSeq	SRR6507244	1	delphine_lariviere's shared submissions	Dept. of biochemistry and molecular, biology,	Delphine Marie Lariviere
151	MiSeq	SRR6703042	1	Carleton Univ.	Biology	Alex Wong
152	MiSeq	SRR6750083	1	Linkoping Univ.	Clinical and Experimental Medicine	Bjorn Berglund
153	MiSeq	SRR6764110	1	FDA/CFSAN	Cfsan-Ors-Dm-Mmsb	Narjol Gonzalez-Escalona
154	MiSeq	SRR6804878	1	North West Univ. Potchefstroom	Unit for Environmental Science and Management: Mic	Daniel Gonzalez-Ibeas
155	MiSeq	SRR6818582	1	Univ. Medical Center Groningen	Medical Microbiology	Silvia Garcia-Cobos

156	MiSeq	SRR6891487	1	Univ. of Maryland	JIFSAN	Xin Gao
157	MiSeq	SRR6892760	1	Columbia Univ.	Division of Infectious Diseases	Nenad Macesic
158	MiSeq	SRR7091281	1	Seqomics Ltd	Bioinformatics Dept.	Balazs Balint
159	MiSeq	SRR7195285	1	edlb-cdc		edlb-cdc service
160	MiSeq	SRR7211942	1	Univ. of Bath	Biology and Biochemistry	Ben Pascoe
161	MiSeq	SRR7213661	1	Iowa State Univ.	Dept. of Veterinary Diagnostic and Production	Fengwei Jiang
162	MiSeq	SRR7250902	1	Univ. of New South Wales	School of Biotech. and Biomolecular Sciences	Ruiting Lan
163	MiSeq	SRR7278569	1	Texas A&M Univ.	Veterinary Integrative Biosciences	Keri Norman
164	MiSeq	SRR733099	1	UT Austin	Barrick	Jeffrey Barrick
165	MiSeq	SRR7435861	1	Canadian Food Inspection Agency	Research and Development, Science Branch	Catherine D Carrillo
166	MiSeq	SRR7475455	1	Columbia Univ.	Systems Biology	German A Plata
167	MiSeq	SRR7631959	1	Bernhard Palsson Lab at UCSD	Bioengineering	Jonathan Pekar
168	MiSeq	SRR7644830	1	AgResearch Limited	Food & Bio-based Products	Adrian Cookson
169	MiSeq	SRR7656066	1	CFSAN	CFSAN	Maria Sanchez Leon
170	MiSeq	SRR7685995	1	Penn State Univ.	Biochemistry and Molecular Biology	Han Mei
171	MiSeq	SRR7690027	1	Kenyon College	Dept. of Biology	Joan L Slonczewski
172	MiSeq	SRR7699085	1	Univ. of Georgia	Population Health	Daniel Ward Nielsen
173	MiSeq	SRR7757609	1	Hvidovre Hospital - Univ. of Copenhagen	Clinical Microbiology	Heidi Gumpert
174	MiSeq	SRR7758899	1	Univ. of Minnesota	Veterinary and Biomedical Sciences	Timothy J Johnson
175	MiSeq	SRR7788551	1	Government of Canada	R&D	Andrew Low
176	MiSeq	SRR7790724	1	CSIRO Agriculture and Food	Food Safety and Stability	Glen Mellor
177	MiSeq	SRR7815230	1	UC San Diego	Bioengineering	Colton J Lloyd
178	MiSeq	SRR7819026	1	Univ. of Aberdeen	School of Biological Sciences	Norval Strachan
179	MiSeq	SRR7840049	1	Univ. of Michigan	Microbiology and Immunology	Ali Mohammed Pirani
180	MiSeq	SRR8082143	1	Univ. of Texas at San Antonio	Biology	Anna Allue Guardia
181	MiSeq	SRR8162810	1	McMaster Univ.	Biochemistry and biomedical sciences	Wael Elhenawy
182	MiSeq	SRR8186688	1	FDA ORA	San Francisco Lab.	Yun Wu
183	MiSeq	SRR8238186	1	Massey Univ.	Inst. of Fundamental Sciences	Vuong Van Hung Le
184	MiSeq	SRR8268071	1	Canadian Food Inspection Agency	R&D	Andrew Low
185	MiSeq	SRR8306314	1	JMI Laboratories	Molecular	Andrew Davis
186	MiSeq	SRR8486178	1	Broad Inst.	Infectious Disease and Microbiome Program	Jonathan Michael Stokes
187	MiSeq	SRR8517669	1	SRCAMB	Science Dept.	Alexander Bogun
188	MiSeq	SRR8573926	1	universite paris Diderot site bichat	Microbiology	Andre birgy
189	MiSeq	SRR8625612	1	Microbiological Diagnostic Unit		Anders Goncalves da Silva
190	MiSeq	SRR8639466	1	Univ. of KwaZulu-Natal (UKZN), South Africa	Pharmaceutical microbiology	Daniel Gyamfi Amoako
191	MiSeq	SRR8832574	1	Li Lab - MIT	Biology	Darren John Parker
192	MiSeq	SRR8836034	1	Univ. of Illinois at Urbana-Champaign	Civil and Environmental Engineering	Yue Xing
193	MiSeq	SRR8874461	1	Universite Libre de Bruxelles	Cell. & Mol. Microbiol.	Nathan Fraikin
194	MiSeq	SRR8924682	1	Penn State Univ.	Food Science	Hillary M Figler

195	MiSeq	SRR8946395	1	Kiel Univ.	Inst. of Microbiology	Tanita Wein
196	MiSeq	SRR9041545	1	Univ. of Sharjah	Infectious Diseases and Anti-Infective therapy Res	Mohamed Ezzat El Zowalaty
197	MiSeq	SRR941832	1	Research Center Borstel		Uwe Mamat
198	MiSeq	SRR9619947	1	Robert Koch Inst.	NG1: Microbial Bioinformatics	Felix Hartkopf
199	MiSeq	SRR9640531	1	South Dakota State Univ.	Veterinary and Biomedical Sciences	Joy Scaria
200	MiSeq	SRR9663517	1	Univ. of Veterinary and Pharmaceutical Scienc	Dept. of Biology and Wildlife Diseases	Adam Valcek
201	MiSeq	SRR9665353	1	UNC Chapel Hill	Chemistry	Greggory Mathew Rice
202	MiSeq	SRR9671401	1	MRC Lab. of Molecular Biology	PNAC	Daniel de la Torre
203	MiSeq	SRR9694420	1	USDA	Agricultural Research Service	George C. Paoli
204	MiSeq	SRR9696347	1	North-West Univ.	Microbiology	Kotsoana Peter Montso
205	MiSeq	SRR9732245	1	USDA-FSIS		Labeed Ben-Ghaly
206	MiSeq	SRR975378	1	CFSAN		FDA/CFSAN:Justin Payne
207	MiSeq	SRR9824964	1	Univ. of Nebraska - Lincoln	Animal Science	Christopher Anderson
208	MiSeq	SRR9864911	1	Wayne State Univ.	Chemistry	Ramin Sakhtemani
209	MiSeq	SRR9964283	1	Nat. Univ. Biomedical Research Inst.	Bioinformatic and Bio-statistics	Sofia Ali
210	MiSeq	SRR9968301	1	Pulsenet		pulsenet service
1	MiniSeq	ERR2777515	27	RIVM - Nat. Inst. for Public Health and th	Euro. Nuc. Archive	Euro. Nuc. Archive
2	MiniSeq	SRR10097238	6	USDA	ARS	Aixia Xu
3	MiniSeq	SRR9886731	3	Pulsenet		pulsenet service
4	MiniSeq	SRR7403873	1	Robert Koch Institut	Enteropathogenic Bacteria and Legionella	Christina Lang
5	MiniSeq	SRR8573904	1	universite paris Diderot site bichat	Microbiology	Andre birgy
6	MiniSeq	SRR8695865	1	Belarusian State Univ.	The Faculty of Biology	Alexander Lagonenko
7	MiniSeq	SRR9691129	1	Skolkovo Inst. of Science and Tech.	Life Sciences	Aleksandra Vasileva
1	NextSeq 500	DRR051046	4	CALGARY	Johann D.D. Pitout, Microbiology, Calgary Laborato	Johann D.D. Pitout, Microbiology, Calgary Laborato
2	NextSeq 500	DRR065639	4	KYOTO_GM	Clinical Lab. Medicine, Kyoto Univ. Gra	Clinical Lab. Medicine, Kyoto Univ. Gra
3	NextSeq 500	DRR129837	4	NIID	Lab. of Bacterial Genomics, Pathogen Genomic	Lab. of Bacterial Genomics, Pathogen Genomic
4	NextSeq 500	ERR3063452	4	Instituto de Salud Carlos III	Euro. Nuc. Archive	Euro. Nuc. Archive
5	NextSeq 500	ERR1837604	4	Saolta	Euro. Nuc. Archive	Euro. Nuc. Archive
6	NextSeq 500	SRR10065352	4	Walailak Univ.	Akkhrararatchakumari Veterinary College	Thotsapol Thomrongsuwannakij
7	NextSeq 500	SRR10094626	4	Washington Univ. in St. Louis School of Medic	Computational and Systems Biology	Alaric W D'Souza
8	NextSeq 500	SRR2970288	4	FDA	DMB	SOLOMON GEBRU
9	NextSeq 500	SRR3989534	4	CFSAN		
10	NextSeq 500	SRR6048398	4	Inst. of Medical Microbiology , Justus Liebig	Microbilogy	Hiren Ghosh
11	NextSeq 500	SRR6049579	4	Western Sydney Local Health District	Centre for Infectious Diseases and Microbiology- P	Rajat Dhakal
12	NextSeq 500	SRR6388512	4	FDA	CFSAN	Maria Sanchez Leon
13	NextSeq 500	SRR6456008	4	Univ. of Sydney	Westmead Inst. for Medical Research	Nouri Laura BEN ZAKOUR
14	NextSeq 500	SRR7799232	4	CFSAN		fda service
15	NextSeq 500	SRR7040505	4	jilhan's shared submissions	Biology	Judith Ilhan
16	NextSeq 500	SRR7297600	4	Univ. of Pittsburgh	School of Medicine	Marissa Pacey

17	NextSeq 500	SRR8449220	4	GEO		Gene Expression Omnibus (GEO), NCBI, NLM, NIH, htt
18	NextSeq 500	SRR7535007	4	Genomic Microbiology Group	Biology	Judith Ilhan
19	NextSeq 500	SRR7828822	4	Univ. of Queensland	UQ Centre for Clinical Research	Patrick Harris
20	NextSeq 500	SRR8426583	4	Queensland Health Forensic and Scientific Services	Public Health Microbiology	Christine J.D. Guglielmino
21	NextSeq 500	SRR8449826	4	Sao Paulo State Univ.	Biological Sciences	Patrick da Silva
22	NextSeq 500	SRR8984168	4	UMC Utrecht	Medical Microbiology	Anita Schurch
23	NextSeq 500	SRR9663534	4	Univ. of Veterinary and Pharmaceutical Scienc	Dept. of Biology and Wildlife Diseases	Adam Valcek
24	NextSeq 500	SRR9665352	4	UNC Chapel Hill	Chemistry	Greggory Mathew Rice
25	NextSeq 500	ERR2027869	3	Dept. of Evolutionary Biology and Environment	Euro. Nuc. Archive	Euro. Nuc. Archive
26	NextSeq 500	ERR2192287	3	Forensic and Scientific Services	Euro. Nuc. Archive	Euro. Nuc. Archive
27	NextSeq 500	ERR2259383	3	Federal Inst. For Risk Assessment (Bfr)	Euro. Nuc. Archive	Euro. Nuc. Archive
28	NextSeq 500	ERR2352505	3	Univ. Of Cologne	Euro. Nuc. Archive	Euro. Nuc. Archive
29	NextSeq 500	ERR3209523	3	Univ. Hospital Basel	Euro. Nuc. Archive	Euro. Nuc. Archive
30	NextSeq 500	SRR3886552	3	Univ. of Lodz	Dept. of Molecular Biophysics, Biobank Lab	Dept. of Molecular Biophysics, Biobank Lab
31	NextSeq 500	SRR6232123	3	New Jersey Medical School-Rutgers Univ.	PHRI center	Liang Chen
32	NextSeq 500	SRR7416822	3	Yonsei Univ.	Chemistry	Soyeong Jun
33	NextSeq 500	SRR7454728	3	Technical Univ. of Denmark	The Novo Nordisk Foundation Center for Biosustaina	Christian Bille Jendresen
34	NextSeq 500	SRR7815271	3	UC San Diego	Bioengineering	Colton J Lloyd
35	NextSeq 500	SRR8179887	3	Pennsylvania State Univ.	Chemical Engineering, Biological Engineering	Howard M Salis
36	NextSeq 500	SRR8533909	3	Public Health Ontario	Research and Development	Nathalie Tijet
37	NextSeq 500	DRR100386	2	KEIO-IAB	Institue for Advanced Biosciences, Keio Univ.	Institue for Advanced Biosciences, Keio Univ.
38	NextSeq 500	ERR2193922	2	Center for Genomic Epidemiology	Euro. Nuc. Archive	Euro. Nuc. Archive
39	NextSeq 500	ERR2534372	2	Dept. of Microbiology and Molecular Genetics,	Euro. Nuc. Archive	Euro. Nuc. Archive
40	NextSeq 500	ERR3317514	2	Utrecht Univ.	Euro. Nuc. Archive	Euro. Nuc. Archive
41	NextSeq 500	SRR5138863	2	UMass Medical School	Center for Microbiome Research	Doyle Ward
42	NextSeq 500	SRR8844760	2	Washington Univ. in St. Louis School of Medic	Pathology and Immunology	Robert Thaenert
43	NextSeq 500	SRR9924677	2	Univ. of Queensland	School of Chemistry and Molecular Biosciences	Scott A Beatson
44	NextSeq 500	ERR1025350	1	The Hebrew Univ. of Jerusalem, Israel	Euro. Nuc. Archive	Euro. Nuc. Archive
45	NextSeq 500	ERR1989106	1		Euro. Nuc. Archive	Euro. Nuc. Archive
46	NextSeq 500	ERR2704796	1	Nat. Center for Microbiology - Inst. of He	Euro. Nuc. Archive	Euro. Nuc. Archive
47	NextSeq 500	SRR5344482	1	Veterinary Research Inst.	Immunology	Darina Cejkova
48	NextSeq 500	SRR5558420	1	Nat. Center for Microbiology - Inst. of He	Reference and Research Lab. of Food and Wate	Sergio Sanchez
49	NextSeq 500	SRR5813581	1	DTU	Novo Nordisk Foundation Center for Biosustainabili	Andreas Porse
50	NextSeq 500	SRR5927195	1	Aligarh MUslim Univ.	Interdisciplinary Biotech. Unit	Asad U Khan
51	NextSeq 500	SRR6364638	1	Univ. of Queensland	School of Chemistry and Molecular Biosciences	Brian M Forde
52	NextSeq 500	SRR6940103	1	Arizona State Univ.	Biodesign Center for Mechanisms of Evolution	Megan Grace Behringer
53	NextSeq 500	SRR7349974	1	Nat. Univ. of Singapore and Genome Instit	Medicine / Infectious Diseases	Swaine L Chen
54	NextSeq 500	SRR7624353	1	Univ. of Pittsburgh School of Medicine	Division of Infectious Diseases	Yohei Doi
55	NextSeq 500	SRR7819144	1	US Food and Drug Administration	Dept. of Molecular Biology	Jayanthi Gangiredla

56	NextSeq 500	SRR8494137	1	Washington Univ. in Saint Louis	Pathology and Immunology	Nathan Crook
57	NextSeq 500	SRR9645716	1	Skolkovo Inst. of Science and Tech.	Skoltech Center of Life Sciences	Anna Shiriaeva
1	NextSeq 550	ERR1841152	80	Warwick Univ.	Euro. Nuc. Archive	Euro. Nuc. Archive
2	NextSeq 550	SRR7866058	35	Kenyon College	Biology	Jeremy Philippe Moore
3	NextSeq 550	SRR5121873	33	Kenyon College	Biology	Preston Basting
4	NextSeq 550	SRR4164003	12	Univ. of Georgia	Environmental Health Science	Adelumola Oladeinde
5	NextSeq 550	SRR8835636	7	Yonsei Univ.	Chemistry	Jeewon Lee
6	NextSeq 550	SRR6801478	2	Arbor Biotechnologies	Research	David Cheng
7	NextSeq 550	SRR8186051	1	Univ. of Sao Paulo	Microbiology	Miriam R Fernandes
8	NextSeq 550	SRR8441387	1	Cleveland Clinic	Translational Hematology and Oncology Research	Jacob G Scott
1	HiSeq 2500	SRR1217287	4	Penn State Univ.	Mwangi Lab	Juan Antonio Raygoza Garay
2	HiSeq 2500	DRR061476	2	Uni_North_Carolina	Medicine / Division of Infectious Diseases, Unive	Medicine / Division of Infectious Diseases, Unive
3	HiSeq 2500	ERR1351685	2	Weizmann Institue Of Science	Euro. Nuc. Archive	Euro. Nuc. Archive
4	HiSeq 2500	ERR1370918	2	VETAGROSUP	Euro. Nuc. Archive	Euro. Nuc. Archive
5	HiSeq 2500	ERR1949075	2	Embl Euro. Bioinformatics Inst.	Euro. Nuc. Archive	Euro. Nuc. Archive
6	HiSeq 2500	ERR1957976	2	Univ. of York	Euro. Nuc. Archive	Euro. Nuc. Archive
7	HiSeq 2500	ERR2039643	2	Univ. Of Manchester	Euro. Nuc. Archive	Euro. Nuc. Archive
8	HiSeq 2500	ERR2135235	2	The Roslin Inst.	Euro. Nuc. Archive	Euro. Nuc. Archive
9	HiSeq 2500	ERR2226598	2	Univ. Of North Carolina At Chapel Hill	Euro. Nuc. Archive	Euro. Nuc. Archive
10	HiSeq 2500	ERR2580161	2	Nanyang Tech. Univ. Food Tech. C	Euro. Nuc. Archive	Euro. Nuc. Archive
11	HiSeq 2500	ERR2639318	2		Euro. Nuc. Archive	Euro. Nuc. Archive
12	HiSeq 2500	ERR2834322	2	Norwegian Veterinary Inst.	Euro. Nuc. Archive	Euro. Nuc. Archive
13	HiSeq 2500	ERR3341400	2	Inserm, Iame, Umr 1137	Euro. Nuc. Archive	Euro. Nuc. Archive
14	HiSeq 2500	ERR588856	2	GSC		
15	HiSeq 2500	SRR1914373	2	NTU	SCELSE	Krishnakumar Sivakumar
16	HiSeq 2500	SRR2054511	2	Indiana Univ.	Lynch Lab	Jean-Francois GOUT
17	HiSeq 2500	SRR2060010	2	Beijing Inst. of Genomics, Chinese Academy of		Kaile Wang
18	HiSeq 2500	SRR2062509	2	Nanyang Tech. Univ.		Maria Yung
19	HiSeq 2500	SRR2864015	2	Nanjing Agricultural Univ.	Key Lab of Animal Bacteriology, Ministry of Agricu	Xiangkai Zhuge
20	HiSeq 2500	SRR3144708	2	Tian Jin Univ. of Science and Tech.		Du Wen
21	HiSeq 2500	SRR3305242	2	Peking Univ.	Biodynamics Optical Imaging Center	Biodynamics Optical Imaging Center
22	HiSeq 2500	SRR3587387	2	Broad Inst./MIT	Biological Enginnering	Biological Enginnering
23	HiSeq 2500	SRR3615377	2	Harvard Univ.	Systems Biology	Systems Biology
24	HiSeq 2500	SRR4292314	2	Singapore Centre for Environmental Life Sciences E	Integrative Analysis Unit	Rohan Williams
25	HiSeq 2500	SRR6825086	2	PHE		
26	HiSeq 2500	SRR5381109	2	Univ. of Pittsburgh School of Medicine	Division of Infectious Diseases	Yohei Doi
27	HiSeq 2500	SRR5447686	2	Harvard Medical School	Genetics	Gleb Kuznetsov
28	HiSeq 2500	SRR5863010	2	Cornell Univ.	Dept. of Food Science	Sophia Harrand
29	HiSeq 2500	SRR6184375	2	mEpilab, Massey Univ.	The Inst. of Veterinary, Animal and Biomedical	Andrew Springer Browne

30	HiSeq 2500	SRR6003477	2	Zoological Inst., Kiel Univ.	Evolutionary Ecology and Genetics	Wentao Yang
31	HiSeq 2500	SRR6024993	2	Univ. of Bern	CMPG, Inst. of Ecology and Evolution	Isabelle Duperret
32	HiSeq 2500	SRR6067093	2	Peking Univ. People's Hospital	Dept. of Clinical Lab.	Ruobing Wang
33	HiSeq 2500	SRR6701909	2	BI	?	bi service
34	HiSeq 2500	SRR7040494	2	jilhan's shared submissions	Biology	Judith Ilhan
35	HiSeq 2500	SRR7216845	2	Anhui Agriculture Univ.	School of Life Sciences	Lumin Yu
36	HiSeq 2500	SRR7361756	2	SCELSE, Nanyang Tech. Univ., Singapor	Cluster 4	Gurjeet S Kohli
37	HiSeq 2500	SRR7457529	2	ARMRC	Biomolecular Engineering	Jay W Kim
38	HiSeq 2500	SRR7535018	2	Genomic Microbiology Group	Biology	Judith Ilhan
39	HiSeq 2500	SRR7592218	2	AgResearch Limited	Food & Bio-based Products	Adrian Cookson
40	HiSeq 2500	SRR7695728	2	Universite de Montreal	Microbiologie, infectiologie et immunologie	Marc Drolet
41	HiSeq 2500	SRR7749992	2	College of Animal Science and Tech.	Anhui Agriculture Univ.	Mei Xue
42	HiSeq 2500	SRR7905333	2	Tianjin institute of industrial biotechnology, Chi	Industrial Enzymes Nat. Engineering Lab.	Zhenxiao Yu
43	HiSeq 2500	SRR8182907	2	Nanyang Tech. Univ.	Nanyang Tech. Univ. Food Tech. C	Moon Tay
44	HiSeq 2500	SRR8404664	2	Centre for DNA Fingerprinting and Diagnostics	Lab. of Bacterial Genetics	J Gowrishankar
45	HiSeq 2500	SRR8607537	2	Chinese Academy of Sciences	Wuhan Inst. of Virology	jintian xu
46	HiSeq 2500	SRR8634287	2	Inst. for Genome Sciences, Univ. of Maryl	Bioinformatics	Suvarna Nadendla
47	HiSeq 2500	DRR016068	1	GIFU_MED	Gifu Univ. School of Medicine Pathogenic Bact	Gifu Univ. School of Medicine Pathogenic Bact
48	HiSeq 2500	DRR089627	1	AIST	Nat. Inst. of Advanced Industrial Science	Nat. Inst. of Advanced Industrial Science
49	HiSeq 2500	DRR093119	1	WASEDA	Dept. of Life Science and Medical Bioscience,	Dept. of Life Science and Medical Bioscience,
50	HiSeq 2500	DRR100436	1	UT-ARTSCI	Wakamoto Lab.	Wakamoto Lab.
51	HiSeq 2500	ERR1276261	1	PHE	Euro. Nuc. Archive	Euro. Nuc. Archive
52	HiSeq 2500	ERR1618847	1	Amphia Hospital	Euro. Nuc. Archive	Euro. Nuc. Archive
53	HiSeq 2500	ERR2209435	1	IZSLT (Istituto Zooprofilattico Sperimentale del L	Euro. Nuc. Archive	Euro. Nuc. Archive
54	HiSeq 2500	ERR2365423	1	Istituto Superiore Di Sanita	Euro. Nuc. Archive	Euro. Nuc. Archive
55	HiSeq 2500	ERR2455336	1	Univ. Of Sheffield	Euro. Nuc. Archive	Euro. Nuc. Archive
56	HiSeq 2500	ERR3265013	1	Nat. Inst. of Public Health and Environmen	Euro. Nuc. Archive	Euro. Nuc. Archive
57	HiSeq 2500	ERR3407987	1	Institut de Biologie Integrative de la Cellule	Euro. Nuc. Archive	Euro. Nuc. Archive
58	HiSeq 2500	ERR3440515	1	Univ. Of Colorado Denver, School Of Medicine	Euro. Nuc. Archive	Euro. Nuc. Archive
59	HiSeq 2500	ERR766384	1	RIVM	Euro. Nuc. Archive	Euro. Nuc. Archive
60	HiSeq 2500	SRR1186394	1	UMIGS	Genomics Resource Core	Liu Xinyue
61	HiSeq 2500	SRR1283288	1	Health Protection Agency	Lab. of Gastrointestinal Pathogens	Timothy Dallman
62	HiSeq 2500	SRR1768060	1	Univ. of Minnesota	Johnson Lab	Kevin Lang
63	HiSeq 2500	SRR1793813	1	Univ. of Delaware	Papoutsakis Lab	Eleftherios Papoutsakis
64	HiSeq 2500	SRR1813744	1	Broad Inst. of MIT and Harvard	Livny	Jonathan Livny
65	HiSeq 2500	SRR1919614	1	Michigan State Univ.	Whitehead Lab	James Stapleton
66	HiSeq 2500	SRR1931189	1	UMIGS	Genomics Resource Core	Luke Tallon
67	HiSeq 2500	SRR2915097	1	Indiana Univ.	Michael Lynch lab	Hongan Long
68	HiSeq 2500	SRR3722117	1	Harvard Medical School	Systems Biology	Systems Biology

69	HiSeq 2500	SRR4115684	1	edlb-cdc	Enteric Diseases Lab. Branch	edlb-cdc service
70	HiSeq 2500	SRR4140240	1	JGI		JGI SRA
71	HiSeq 2500	SRR5088180	1	edlb-cdc		edlb-cdc service
72	HiSeq 2500	SRR5194987	1	The Univ. of Queensland	Australian Infectious Diseases Research Centre, Sc	Minh-Duy Phan
73	HiSeq 2500	SRR5258776	1	Nestle Inst. of Health Sciences	Functional Genomics	Kassam Mohamed
74	HiSeq 2500	SRR5279323	1	North Dakota State Univ.	Dept. of Microbiological Sciences	Oleksandr Maistrenko
75	HiSeq 2500	SRR5280252	1	Indiana Univ.	Biology	Hongan Long
76	HiSeq 2500	SRR5482170	1	Arsanis Biosciences GmbH	MIV	Michele Mutti
77	HiSeq 2500	SRR5521494	1	Peking Univ. People's hospital	Dept. of Clinical Lab.	Yawei Zhang
78	HiSeq 2500	SRR5572609	1	UW-Madison, Pfleger Lab	Chemical and Biological Engineering	Gina Gordon
79	HiSeq 2500	SRR5830100	1	UC San Diego	Bioengineering	Jonathan Monk
80	HiSeq 2500	SRR5936511	1	Univ. of Queensland	UQ Centre for Clinical Research	Patrick Harris
81	HiSeq 2500	SRR6513355	1	Wuhan Inst. of Tech.	School of Chemical Engineering and Pharmacy	Junliang Zhong
82	HiSeq 2500	SRR7230064	1	Univ. of Bath	Biology and Biochemistry	Nicola Marie Coyle
83	HiSeq 2500	SRR7469883	1	Univ. of Tech., Sydney	i3	Max Laurence Cummins
84	HiSeq 2500	SRR7819316	1	Univ. Grenoble Alpes	LiPhy	Stephan Lacour
85	HiSeq 2500	SRR8060798	1	Broad Inst.	Infectious Disease and Microbiome (IDM) Program	Xiaofang Jiang
86	HiSeq 2500	SRR8115633	1	UNIFESP	Microbiology and Immunology	Fernanda Fernandes dos Santos
87	HiSeq 2500	SRR8176980	1	UMIGS	Genomics Resource Center	Luke Tallon
88	HiSeq 2500	SRR8368414	1	Federal research and Clinical Center of Physical-C	Dept. of molecular biology and genetics	Andrei Guliaev
89	HiSeq 2500	SRR8480428	1	Amrita Viswa Vidyapeetham	School of Biotech.	Sanjay Pal
90	HiSeq 2500	SRR8792696	1	Public Health England	Gastrointestinal Bacteria Reference Unit	David R Greig
91	HiSeq 2500	SRR8867895	1	Theo Allnutt Bioinformatics	Bioinformatics	Theodore Richard Allnutt
92	HiSeq 2500	SRR9620280	1	Universite Laval	Centre de Recherche en Infectiologie	Philippe Leprohon
93	HiSeq 2500	SRR9717097	1	Univ. of Veterinary Medicine Vienna	Dept. for Farm Animal and Public Health in Ve	Catia Pacifico
1	NovaSeq 6000	SRR11810195	40	NEIKER - Basque Inst. for Agricultural Researc	Animal Health Dept.	Medelin Ocejo
2	NovaSeq 6000	ERR4221187	34	IMBIM(Dept. of Medical Biochemistry and Micro	Euro. Nuc. Archive	Euro. Nuc. Archive
3	NovaSeq 6000	SRR11075478	30	Univ. of Illinois at Urbana-Champaign	Civil and Environmental Engineering	Yue Xing
4	NovaSeq 6000	SRR12020688	30	Univ. of Tech. Sydney	ithree institute	Veronica Jarocki
5	NovaSeq 6000	SRR7664908	23	GEO		Gene Expression Omnibus (GEO), NCBI, NLM, NIH, htt
6	NovaSeq 6000	SRR8610355	18	Univ. of Bern	Inst. for Infectious Diseases	Mathieu Clement
7	NovaSeq 6000	SRR9648314	13	JGI		JGI SRA
8	NovaSeq 6000	SRR11018568	12	Colorado State Univ.	Dept. of Biology	Daniel B Sloan
9	NovaSeq 6000	SRR11286234	10	Boston Univ.	Biomedical Engineering	Carly Ching
10	NovaSeq 6000	SRR11307872	6	Yangzhou Univ.	College of Veterinary Medicine	peili wang
11	NovaSeq 6000	SRR11791061	6	Stanford Univ.	BioEngineering	Anthony Lyndon Shiver
12	NovaSeq 6000	SRR11593518	4	Univ. College Cork	Cancer Research at UCC	Sidney Walker
13	NovaSeq 6000	SRR12041927	4	Univ. of California Irvine	Ecology and Evolutionary Biology	Tiffany N Batarseh
14	NovaSeq 6000	SRR9074522	4	Univ. of Tech. Sydney	ithree institute	Kay Anantanawat

15	NovaSeq 6000	ERR3713240	1	RIVM	Euro. Nuc. Archive	Euro. Nuc. Archive
16	NovaSeq 6000	ERR4010337	1	Univ. of Greifswald, Inst. of Pharmacy	Euro. Nuc. Archive	Euro. Nuc. Archive
17	NovaSeq 6000	ERR4059196	1	Leiden Univ. Medical Center	Euro. Nuc. Archive	Euro. Nuc. Archive
18	NovaSeq 6000	SRR9047311	1	Univ. of Toronto	Molecular Genetics	William Wiley Navarre
19	NovaSeq 6000	SRR9643625	1	Nat. Inst. of Advanced Industrial Science	Bioproduction Research Inst.	Kentaro Miyazaki
1	HiSeq X Ten	SRR7879965	40	BI		bi service
2	HiSeq X Ten	SRR7716577	39	West China Hospital, Sichuan Univ.	Center of Infectious Diseases	Zhiyong Zong
3	HiSeq X Ten	SRR6069470	29	Peking Univ. People's Hospital	Dept. of Clinical Lab.	Ruobing Wang
4	HiSeq X Ten	SRR7280090	12	Beijing university of agriculture	Food science and engineering	xiaoxia Li
5	HiSeq X Ten	SRR7537408	12	GEO		Gene Expression Omnibus (GEO), NCBI, NLM, NIH, htt
6	HiSeq X Ten	SRR9855382	10	South China Agricultural Univ.	College of Veterinary Medicine	Xiufeng Zhang
7	HiSeq X Ten	SRR7641178	7	College of Animal Science South China Agricultural	No.483 Wushan, Tianhe District, Guangzhou City	yiwen yang
8	HiSeq X Ten	SRR7474260	6	Jiangnang Univ.	Nat. Engineering Lab. for Cereal Ferment	Kangjia Zhu
9	HiSeq X Ten	SRR8335003	5	College of Biotech. and Pharmaceutical Engine	State Key Lab. of Materials-Oriented Chemica	Yong Chen
10	HiSeq X Ten	SRR7205166	2	Liaocheng Univ.	Agricultural College	WANG Ren-hu
11	HiSeq X Ten	ERR2893770	1		Euro. Nuc. Archive	Euro. Nuc. Archive

VITA

Nicholas Stoler

Education

2012 Aug – 2020 Dec	Ph.D. program in Bioinformatics and Genomics	Pennsylvania State University
2010 Aug – 2012 May	M.S. program in Bioinformatics	Johns Hopkins University
2004 Aug – 2008 May	B.S. program in Biology	Georgetown University

Publications

2020 **Stoler, Nicholas**, Barbara Arbeithuber, Gundula Povysil, Monika Heinzl, Renato Salazar, Kateryna D. Makova, Irene Tiemann-Boege, and Anton Nekrutenko. "Family Reunion via Error Correction: An Efficient Analysis of Duplex Sequencing Data." BMC Bioinformatics 21 (1): 1–10.

2016 **Stoler, Nicholas***, Barbara Arbeithuber*, Wilfried Guiblet, Kateryna D. Makova, and Anton Nekrutenko. "Streamlined Analysis of Duplex Sequencing Data with Du Novo." Genome Biology 17 (1): 180.

2014 Rebolledo-Jaramillo, Boris*, Marcia Shu-wei Su*, **Nicholas Stoler,** Jennifer A McElhoe, Benjamin Dickins, Daniel Blankenberg, Thorfinn S Korneliussen, Francesca Chiaromonte, Rasmus Nielsen, Mitchell M. Holland, Ian M. Paul, Anton Nekrutenko, and Kateryna D. Makova. "Maternal Age Effect and Severe Germ-Line Bottleneck in the Inheritance of Human Mitochondrial DNA." Proceedings of the National Academy of Sciences of the United States of America 111 (43): 15474–79.

2014 Dickins, Benjamin*, Boris Rebolledo-Jaramillo*, Marcia Shu Wei Su, Ian M Paul, Daniel Blankenberg, **Nicholas Stoler**, Kateryna D Makova, and Anton Nekrutenko. "Controlling for Contamination in Re-Sequencing Studies with a Reproducible Web-Based Phylogenetic Approach." *BioTechniques* 56 (3): 134–41.

2014 Blankenberg, Daniel, Gregory Von Kuster, Emil Bouvier, Dannon Baker, Enis Afgan, **Nicholas Stoler**, James Taylor, and Anton Nekrutenko. "Dissemination of Scientific Software with Galaxy ToolShed." Genome Biology 15 (2): 403.

*contributed equally

Fellowships

2014 Jan – 2015 Dec	NRSA Institutional Predoctoral Training Grant (T32)
2004 Aug – 2008 May	Georgetown-Howard Hughes Undergraduate Research Scholarship