Université de Montréal

The Detection of High-Qualified Indels in Exomes and Their Effect on Cognition

Par Nadine Younis

Unité académique de Biochimie et de Médecine Moléculaire
Faculté de Médecine

Mémoire présentée en vue de l'obtention du grade de Maîtrise
en Bio-informatique
option recherche

Décembre 2021

Université de Montréal

Faculté de Médecine

Ce mémoire intitulé

The Detection of High-Qualified Indels in Exomes and Their Effect on Cognition

Présenté par

Nadine Younis

a été évalué par un jury composé des personnes suivantes :

Martin Smith

(Président-rapporteur)

Sébastien Jacquemont

(Directeur de recherche)

Philippe Campeau

(Membre de jury)

Résumé

Plusieurs insertions/délétions (indels) génétiques ont été identifiées en lien avec des troubles du neurodéveloppement, notamment le trouble du spectre de l’autisme (TSA) et la déficience intellectuelle (DI). Bien que ce soit le deuxième type de variant le plus courant, la détection et l’identification des indels demeure difficile à ce jour, et on y retrouve un grand nombre de faux positifs. Ce projet vise à trouver une méthode pour détecter des indels de haute qualité ayant une forte probabilité d’être des vrais positifs.

Un « ensemble de vérité » a été construit à partir d’indels provenant de deux cohortes familiales basé sur un diagnostic d’autisme. Ces indels ont été filtrés selon un ensemble de paramètres prédéterminés et ils ont été appelés par plusieurs outils d’appel de variants. Cet ensemble a été utilisé pour entraîner trois modèles d’apprentissage automatique pour identifier des indels de haute qualité. Par la suite, nous avons utilisé ces modèles pour prédire des indels de haute qualité dans une cohorte de population générale, ayant été appelé par une technologie d’appel de variant.

Les modèles ont pu identifier des indels de meilleure qualité qui ont une association avec le QI, malgré que cet effet soit petit. De plus, les indels prédits par les modèles affectent un plus petit nombre de gènes par individu que ceux ayant été filtrés par un seuil de rejet fixe. Les modèles ont tendance à améliorer la qualité des indels, mais nécessiteront davantage de travail pour déterminer si ce serait possible de prédire les indels qui ont un effet non-négligeable sur le QI.

Mots clés : Variants nucléotide simple, indels, QI, apprentissage automatique, scores génétiques, analyses statistiques, trouble du spectre de l’autisme.

Abstract

Genetic insertions/deletions (indels) have been linked to many neurodevelopmental disorders (NDDs) such as autism spectrum disorder (ASD) and intellectual disability (ID). However, although they are the second most common type of genetic variant, they remain to this day difficult to identify and verify, presenting a high number of false positives. We sought to find a method that would appropriately identify high-quality indels that are likely to be true positives.

We built an indel "truth set" using indels from two diagnosis-based family cohorts that were filtered according to a set of threshold values and called by several variant calling tools in order to train three machine learning models to identify the highest quality indels. The two best performing models were then used to identify high quality indels in a general population cohort that was called using only one variant calling technology.

The machine learning models were able to identify higher quality indels that showed a association with IQ, although the effect size was small. The indels predicted by the models also affected a much smaller number of genes per individual than those predicted through using minimum thresholds alone. The models tend to show an overall improvement in the quality of the indels but would require further work to see if it could a noticeable and significant effect on IQ.

Keywords: single-nucleotide variants, indels, IQ, machine learning, genetic scores, statistical analysis, ASD.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AC: alternate allele depth

ASD : autism spectrum disorder

AUC: area under the curve

BAM: binary alignment map

Bp: base pair

CI: confidence interval

CNV : copy number variant

FPR: false positive rate

GATK: genome analysis tool kit

GIAB: genome in a bottle

gnomAD: genome aggregation database

gVCF: genome variant calling format

ID: intellectual disability

Indel: insertion/deletion

IQ: intelligence quotient

LOEUF: loss-of-function observed/expected upper bound fraction

LoF: loss-of-fuction

MAF: minor allele frequency

MRR: mutant read ratio

NDD: neurodevelopmental disorder

NGS: next generation sequencing

NPD : neurodevelopmental psychiatric disorder

ROC: receiving operating characteristic

SAM: sequence alignment map

SMRT: single molecule real time

SNV : single nucleotide variants

SPARK: Simons Foundation Power Autism Research

SSC: Simon Simplex Collection

SV: structural variants

TGS: third generation sequencing

TPR: true positive rate

UKBB: UK BioBank

VCF: variant calling format

VEP: variant effect predictor

VQSR: variant quality score recalibration

WES: whole exome sequencing

WGS: whole genome sequencing

Dedication

To my husband, Anthony, my parents, Falah & Rethab, my sister, Renade, and my oldest and best friend, Farah.

Acknowledgements

I would like to thank Dr. Sébastien Jacquemont for allowing me the opportunity to work in his lab and for taking me on as a student. His dedication to his work and his knowledge is inspiring.

I would like to give a special thanks to Mor Absa Loum and Martineau Jean-Louis, who were with me through every step of the project and provided so much insight and guidance and the occasional pep talks that helped me reach my end goal.

A special thank you to Inga Knoth for always encouraging me to stand up for myself, to Petra Tamer for fielding all my questions and for always supporting me through every challenge and a very special thank you to Charles-Olivier Martin, who I consider a mentor and, more importantly, one of my closest friends. I would also like to thank my coworkers, who have made working in the lab a true pleasure for the past four years.

Lastly, I would like to thank my husband, my family, my mother-in-law Micheline Andraos, and my friends, particularly Jaymie Guy Goulet Métivier, my sisters Nadine and Darine Tahan and Camille Georges. Their support and endless patience helped me see this project to the end.

CHAPTER 1: INTRODUCTION

Introduction

Rare single-nucleotide variants (SNVs) are major contributors to neurodevelopmental psychiatric disorders (NPDs), such as autism spectrum disorder (ASD) and intellectual disability (ID). However, our understanding of the impact of SNVs on neuropsychiatric phenotypes is limited in at least two ways: Firstly, the effects sizes of the vast majority of pathogenic SNVs on neuropsychiatric phenotypes remain undocumented, and their scarcity will prevent any individual association studies. Most studies have focused on de novo mutations [1,2], yet rare inherited mutations remain understudied. Secondly, it is unknown if deleterious SNVs in NPD genes have specific effects or if they impact the same neuropsychiatric domains through shared mechanisms. Exploring this hypothesis, previous work has shown that genetic scores and functional annotations can accurately predict the effect of any rare copy number variants (CNVs) on the intelligence quotient (IQ) with 78% accuracy[3]. These approaches have not yet been extended to SNVs and other dimensional neuropsychiatric phenotypes

This work will focus on indels that represent some of the most difficult variants to identify. Indels are genetic insertions or deletions of up to 1000 base pairs (bp). Their functional effects are as of yet not fully understood in the general population [4]. Indels that are not multiples of three can lead to a loss-of-function (LoF) mutation called a frameshift mutation. This type of variant is marked by a change in the frame of a codon, which could lead to changes in the mRNA pathway, or could create a premature stop [5]. These disruptions have been found to be linked to both ID and ASD [6,7]. However, despite their clinical importance, indels remain to this day difficult to identify and present a high number of false positives [4,8,9].

DNA Sequencing

In 1977, Fred Sanger developed a new DNA sequencing method using labelled chain-terminating inhibitors [10]. This method, now commonly known as Sanger sequencing, has since become the gold standard method for identifying DNA sequences, particularly for shorter DNA fragments [11]. Although it presents a per-base accuracy of greater than 99.99% [12], Sanger sequencing works by sequencing one DNA fragment at a time [13], making it difficult to apply this method on a large quantity of data. Throughout the years, there were attempts to improve this method to render it more efficient [14–16], but it wasn't until the advent of Next Generation Sequencing (NGS) that DNA sequencing on a larger and faster scale became feasible [17].

Developed in the late 1990s, NGS is a series of technologies that have since become the modern standard. All NGS technologies are able to sequence small DNA fragments in parallel, making them more time effective and allowing for a greater discovery of disease-causing mutations as they do not require any prior knowledge of the region under investigation [17]. While Sanger sequencing remains the ultimate method of validation [11,18], NGS technologies accuracies can be as high as 99.9% [19], with an inter-technology concordance of 99.20% [20], making them reliable tools for DNA sequencing. Many NGS technologies are currently available, but the most popular amongst them is Illumina technology, which sequencing depending on the target and has a multitude of sequencing machines that have many levels of throughput [21].

After NGS came the next wave in genomics: third generation sequencing (TGS), also known as long-read sequencing. TGS has the advantage of having longer reads than NGS, as well as providing real-time sequencing [22]. The first long-read technology to appear on the market was Single Molecule Real Time (SMRT) sequencing released by Pacific Biosciences [23] in 2009, followed by nanopore sequencing, as provided by Oxford Nanopore Technologies [24] in 2014. Long-read sequencing is notorious for having a high error-rate as compared to NGS. For example, SMRT sequencing has a 12.86% error rate across the entire genome, as opposed to 0.8% in Illumina [25].

Genome sequencing can be broken down into several groups, the two most pertinent being whole-exome sequencing (WES) and whole-genome sequencing (WGS). The core difference between the two is that WGS targets the entire genome, both coding and non-coding regions, whereas WES only targets exons, of which protein-coding exons cover ~1% of the genome [26]. WGS is an essential tool in detection of longer variants, as many of them extend into non-coding regions that are not accessible by WES. When sequencing for regions that were targeted by WES, WGS detected a greater number of SNVs [27,28]. Furthermore, WES is known to be biased by the GC content, with coverage bias found in regions of high or low GC content[29]. While WGS is an overall more powerful technology, it is also more expensive and many of the non-coding regions are difficult to interpret due to lack of information on their roles [28]. However, although WES covers only a small percentage of the genome as compared to WGS, many monogenic diseases and high-penetrance protein-altering variants are detected in the coding region, which are known to have functional consequences and to potentially be deleterious [30]. Furthermore, WES is a more cost-effective sequencing method as well as being time-effective and provides a higher coverage than WGS [31,32].

Variant Identification Pipeline

DNA sequencing is the first step in the process of analysing genomic data. Once the genome is sequenced, the next step is the sequence alignment. Alignment is the process of comparing DNA sequences against the reference sequence in order to find their similarities and differences [33]. This comparison allows the detection of variants that may have been inserted, deleted or substituted[28]. Different types of needs require different types of alignment; for example, multiple sequence alignment compares three or more sequences of the same length in part to infer the evolutionary relationship between them [34]. However, when trying to determine where on a genome a sequence is located, pairwise alignment is used [34], particularly using tools adapted for short-read sequence alignment [35]. Technologies such as Bowtie2, BWA, HISAT2 and many more can be used for short-read alignment. Out of the many different software, BWA was found to be the best performing technology with a better alignment rate and gene coverage in sequences shorter than 500 base pairs (bp) [35,36].

After alignment, the final step in variant identification is variant calling. This is the process where variants that differ from the reference genome are identified and written into a Variant Calling Format (VCF) or Genome Variant Calling Format (gVCF) file [37]. Variant calling can be computed by several different algorithms, such as the Genome Analysis Tool Kit (GATK) HaplotypeCaller [38], Freebayes [39], DeepVariant and WeCall[40]. GATK HaplotypeCaller uses *de novo* assembly within the region, therefore, when it encounters an area with a potential variant, it discards the existing mapping region and re-writes with the variants. It determines the likelihood of given haplotypes being present according to the read data. This makes the variant calling much more sensitive, which means that more variants are read. However, the downside of the increased sensitivity is that there are a lot more false positives which then need to be filtered out. The filtering is carried out by the Variant Quality Score Recalibration (VQSR). VQSR determines which variants are of higher quality and discards those that do not pass the filtering process. The filtering is done in two steps: the first step involves building a model using a high-quality variant sample set and the second step is to use a random forest model to calculate a score for each variant. It is important to note that not all datasets have a high-quality sample set, which means that it is not possible to apply VQSR filtering to every dataset. Furthermore, this type of modelling requires a large amount of data to be effective. So, while this score is more reliable than the standard QUAL score generally computed by variant calling algorithms, it is only available for a few datasets [38].

Freebayes is a bayesian model variant caller, based on the literal sequence of reads within a target and uses bayesian modeling to determine the most-likely combination of genotypes based on the reference genome. This allows for a smaller error rate, requires less computation power and allows us to read different sequences on the same position [39]. WeCall is used to call variants in NGS data. It infers the presence of variants using genomic sites where variants exist and compares these sites to the reference genome[36]. Lastly, DeepVariant is a variant caller that works using the deep learning method and is used for high throughput sequencing data. It constructs images containing multiple channels, which are the different colours of the image. Each channel represents a certain characteristic of the sample, such as read depth or mapping quality. DeepVariant analyzes these images and determines the likelihood of it being true [41]. Using Genome in a Bottle (GIAB) [42] datasets for validation and benchmarking, it aligns to the reference genome and then infers the true sequences based on that. The raw data, consisting of multiple reads of overlapping fragments, are mapped to the reference genome. The model is trained to read and identify these locations and separate them from sequencing errors [43].

After variant calling, the final step in the data analysis pipeline is the annotation of the variants. Annotation is the step that assigns functional information to the variants [43] and allows us to see whether the variant is present on a protein-coding gene and if it has any effect on that gene [44]. Annotation can be broken down into three steps: identification of coding regions, prediction of genes affected and identification of the processes and pathways affected [45]. Annotation can be done manually or through the use of different algorithms such as the NCBI Eukaryotic Genome Annotation Pipeline[41] or the Ensembl Variant Effect Predictor (VEP)[42].

Data Formats

Throughout the DNA analysis process, the data is stored in several types of files, one of the first being the FASTA file. A FASTA file is a text file that stores nucleotide sequences read and is represented by the sequence of bases [27]. The first line begins with a greater-than symbol ">" and is followed by the description of the sequence. The next line is the sequence itself[22]. A FASTQ file is an extension of the FASTA format, however, a FASTQ file has a score associated with each nucleotide, called a Phred score, which is determined by the sequencing accuracy. The Phred score considers the probability that a sequencer called a given base incorrectly. The equation is the following: $\mathbf{Q=-10\log_{10}P}$ where Q is the Phred score and P is the base-error probability. It has now become the standard for quality scoring [46], thus making the FASTQ format a superior tool to the FASTA format. However, it is important to note that the Phred score does not encompass every aspect of a nucleotide, and further algorithms are required to determine accuracy in later steps [47].

After the sequences are aligned, a Sequence Alignment Map (SAM) file is generated. A SAM file is a tab-delimited text file with the sequences aligned according to the reference genome.

It can consist of one header line beginning with '@' followed by the aligned sequence. The binary equivalent of a SAM file is a Binary Alignment Map (BAM) file, which is generated in order to improve performance. BAM files are compressed in BGZF format which makes them more compact and allows for faster retrieval of information [48].

After variant calling, the variants are stored in files containing formatted genotyping information, such as a VCF or gVCF file. A VCF is a text-based file that stores information on sequence variations in the genome, such as single nucleotide polymorphisms (SNP), structural variants (SV) and copy number variants (CNV). This file is a standardised output of all variant calling technologies and is separated by columns that include the basic information of each variant such as chromosome, position, reference allele, alternate allele, VQSR filter as well as any additional information such as genotype, read depth of the alternate allele and the total read depth, mapping quality, genome quality and any other relevant information [37]. A gVCF file is a VCF on which the 1000 Genome Project conventions for the representation of a genotype have been applied and which is compressed by gzip [49]. VCF and gVCF files can be manipulated using a number of tools, the most common of which are VCFtools[37] and BCFtools[50], which are software that work especially with VCF and gVCF files for any necessary manipulations such as merging files or comparing them [37].

Sequencing Errors

While sequencing methods have improved significantly over the years, errors can still occur. Errors are possible along every step of the pipeline, from sequencing to annotation. Some errors can occur due to the nature of the sequences, which make it harder for sequencing technologies to accurately detect bases. For example, homopolymers of seven bases or longer and repeated sequences within the DNA tend to have higher base call errors; one example of such is that after sequencing a homopolymer, the first base after the homopolymer will have a substitution error, and it will be substituted to the same base as the homopolymer [51]. Furthermore, segmental duplications, which are DNA sequences that are mostly repeated, can produce more errors during sequencing because of an increased chance of mis-assembly[52,53,54]. Other errors can be attributed to the region in which the sequence is found. Certain regions of the genome with a high number of repetitive regions or repeated nucleotides can have a high signal yet yield inaccurate results due to an amplification of noise[47,48]. Thus, while sequencing has improved drastically and continues to improve, there continue to be errors that can potentially bias research, particularly when sequencing for difficult to read regions and sequences.

Genetic Variants

The variant calling and identification process can help us identify all types of genetic variants. These variants can be split into three categories: SVs, single nucleotide variants (SNVs) and insertions-deletions (indels) [55]. SVs can be either inversions of the genome, translocations, insertions or deletions. If they cover 1kb or more, they are referred to as CNVs [56]. They can be benign [56] or they can be associated with several types of diseases and neurodevelopmental disorders, such as many types of cancers and autism [3,57].These variants can be inherited from the parents or can emerge as a novel mutation found in an individual, called *de novo,* caused by mutagenesis in parental gametes[58]. While both types of inheritance can be potentially deleterious, *de novo* variants are a more rare form of mutation and can be far more deleterious than inherited variants[59]. *De novo* mutations have a very low incidence in the human genome, likely due to their deleteriousness and their detection shows a high number of false positives[60].

SNVs are changes in the genome that affect one nucleotide, notably a substitution of a given base from one nucleotide to another [55]. While they are the most common type of genetic variation, the average person carries many SNVs that show no risk to their health [61]. Such types of SNVs are commonly called synonymous mutations and they are considered to be functionally silent [62]. However, the SNVs that are found in coding regions have been found to have an effect on protein-coding genes, and have been linked to several diseases [63]. In the case of SNVs, there are two types of these loss-of-function (LoF) variants that can occur: nonsense mutations and splice-site mutations. Nonsense mutations are variants that introduce a premature stop codon that can disrupt protein functioning; splice-site mutations are variants that can disrupt a splice-site, which is on the boundary between an intron and an exon [9].

The third type of variants, indels, are small insertions and deletions of bases in the genome, as suggested by their name. They are the second most common type of variant, after SNVs, but they have proven to be difficult to detect and validate due to their size [64]. While they can be benign, they can also lead to the final type of LoF called a frameshift mutation. These mutations occur when an indel is not a multiple of three and creates a disruption in a codon's reading frame [9,65]. This shift can lead to the coding of an entirely different amino acid or could result in a premature stop codon [65].

Changes in protein-coding genes have been linked to a wide range of neurodevelopmental disorders (NDDs) and SNVs have been found to play a major role in genes linked with ASD. Despite the heterogeneity of the disorder, LoF mutations are enriched in individuals with ASD as opposed to those without [66]. Frameshift mutations in particular have been found to be linked to both intellectual disability and ASD[6,7]. For example, disruptions due to frameshift variants on the

BCL11A and *SCN2A* genes have been linked to Intellectual Disabilities (ID) [6,67]. Furthermore, frameshift disruptions found on the *SHANK3* gene has been found in 18 individuals with ASD [68], while a frameshift mutation found on the *TCF20* gene has been found in a woman with both ASD and an ID [69].

Difficulties in Identifying Indels

Despite their important role in gene function, indels are notoriously hard to detect. The first challenge in detecting indels is their very small size, often as small as a single base pair, therefore making them particularly difficult to map according to the reference genome because several candidate sequences exist for the same site[8]. Furthermore, the presence of repeated bases can also be an obstacle in indel detection, particularly in areas with homopolymers, which potentially leads to incorrect nucleotide alignment [5]. Indels that are detected show low concordance between different variant callers, with only 26.8% concordance rate[70]. While all indels are difficult to detect due to their nature, indels with low read rates lead to more call-rate errors[71] and indels with low mapping quality are filtered out to improve calling rate [72]. However, despite these filters and improvements in technology the problem remains that current tools have trouble accurately detecting indels [5]. It is therefore important to develop a protocol that will not only filter out any low-quality data, but also confidently assess the likelihood that a given indel is a true positive.

Hypothesis

Using a dataset with multiple variant-calling technologies can help delineate features that identify high quality indels.

Aims

Overarching aim: To identify high quality indels regardless of the number and type of variant calling algorithms.

Specific aims:

1) Establish a set of "high quality indels" by using the initial filtering method, intersecting multiple calls and using parental information.
2) Train a logistic regression model and a random forest model to identify features that best predict high quality indels.
3) Validate and test performance of the model.
4) Apply to a general population cohort and test effects of indels on cognitive ability.

CHAPTER 2: METHODOLOGY

Methodology

Cohorts

Two diagnosis-based autism family cohorts were used, in addition to one general population cohort. The two autism family cohorts are Simons Foundation Power Autism (SPARK)[73] and Simons Simplex Collection (SSC)[74] whereas the general population cohort is the UK Biobank (UKBB)[75]. We were provided with SNV calling data from different genetic data repositories such as Autism Speaks and Simons Foundation.

Table I: Distribution of individuals in each cohort

Cohorts		Pedigree	N (Total= 236,572)	N with ASD (Total=12,144)	N males (%)	N with Intelligence measures (%) (Total = 161,917)
Autism family	SPARK	Parents	14,522	161	6,122 (42.16)	841 (5.79)
		Probands	9,607	9,607	7,615 (79.27)	1,394 (14.51)
		Siblings	3,134	0	1,599 (51.02)	325 (10.36)
	SSC	Parents	4,511	0	2,136 (46.35)	4 (0.09)
		Probands	2,376	2,376	2,072 (87.20)	2,376 (100)
		Siblings	1,791	0	908 (50.70)	0 (0)
General Population	UKBB	NA	200,631	NA	90,020 (44.87)	156,784 (78.14)

Table II: Data collection information

Cohort	Sample size	Sequencing Type	Sequencing Technology	SNV mapping genome	Alignment Technology	Variant Calling Technology
SSC	8,678	Exome	Illumina HiSeq 2000 [76]	Hg38	BWA[36]	GATK HaplotypeCaller/ Freebayes
SPARK	27,263	Exome	Illumina NovaSeq 6000[77]	Hg38	BWA	GATK HaplotypeCaller/ DeepVariant / WeCall
UKBB	200,631	Exome	Illumina NovaSeq 6000	Hg38	BWA	DeepVariant

SPARK was sequenced using Illumina NovaSeq6000 and aligned with BWA version 0.6.2-r126. Variant calling was performed using three algorithms: GATK Haploytypecaller, DeepVariant and WeCall. All variants were written into gVCF files. These gVCF files were used to convert the variants from the GRCh38 genome to the GRCh37 genome by a process called "lifting over" using CrossMap version 0.4.2[78].

SSC was sequenced using Illumina HiSeq 2000 and aligned with BWA. Variant calling was done by two algorithms: GATK HaplotypeCaller and Freebayes. All variants were written into gVCF files. These gVCF files were used to lift over the variants from the GRCh38 genome to the GRCh37 genome.

UKBB was sequenced using Illumina NovaSeq 6000 and aligned using BWA mem version 0.7.15. After alignment, the variants were called using DeepVariant version 0.8.0 using the GRCh38 genome as the reference and output into gVCF files. Finally, the gVCF files were used to lift over the variants from the GRCh38 to the GRCh37 genome, as it is the standard used in our lab.

Establishing a truth set

Before beginning any manipulations, it was important to create a set of criteria to apply to the indels in order to create a "truth set" on which to train our machine learning models. We set the following criteria: high quality indels are called by a maximum number of variant calling technologies, they have passed the quality thresholds we have set, and they are not *de novo*. The first criterion is crucial in establishing our set because of the high number of false positives found in indels; we therefore go by the assumption that an indel that is called by multiple variant calling technologies has a higher likelihood of being true, as established by many previous works[2,42]. Establishing a truth set requires the highest quality indels, therefore we are being stringent and selecting indels that have been called by all variant calling technologies available for a given cohort (3 for SPARK and 2 for SSC). The second criterion allows us to filter out any low quality indels with a small number of reads as we consider them less likely to be true. Finally, the reason for the exclusion of *de novo* variants is two-fold: we believe if a variant is seen more than once in a cohort, it has a higher likelihood to be true; furthermore, *de novo* variants are extremely rare and would therefore present a high number of false positives[60]. Because we are excluding *de novo* variants, it is important to relabel the transmission of all the probands and siblings. The reason for this is that, through filtering, a parent is excluded because they do not meet the threshold, the child must be relabelled as *de novo* because we consider the indel carried by the parent as a false positive. While this level of stringency may cause us to eliminate many potentially true indels, this is a loss we are willing to accommodate because our final set will have the highest likelihood of being true.

Normalisation

The first step before beginning any filtering is the normalisation of the variants we are using. Indels are normalised when they are both parsimonious and left-aligned. Parsimony, in this case, is the shortest possible representation of the variant that is not an allele of size zero, and left alignment is the process in which the start of the variant is at its leftmost position. These two methods allow a given variant to begin at the same position and to be identified using the same criteria within all sets[79].

Because the indels in each cohort were called using many different technologies, it is important to normalise them to create uniformity as there is no set standard for variant calling[70]. SPARK and SSC indels were extracted from the VCF files using VCFtools (version 0.1.16) and were then normalised according to the Hg19 reference genome using BCFtools norm (version 1.13). After normalisation, certain variants that were considered indels were revealed to be point mutations. The indels were therefore re-extracted from the dataset and the SNVs were discarded.

Genotype Information Storage

Once the indels are normalised and re-extracted, they are written into a genotype feature file where each line represents a given indel per individual. This file includes important information on the individual, the genotype and the family. Each line contains the following: the individual ID, the variant they carry, the alternate allele depth, the total read depth, the ratio of the alternate allele depth over the total read depth (mutant read ratio or MRR), the genotype, the transmission (inherited from the mother, father or *de novo*), the parental genotype and variant calling information. The genotype thus contains all the information necessary for filtering and annotating variants, which is the following step.

Filtering, retagging and annotation

According to the criteria set by our truth set, the normalised indels were filtered, retagged and annotated in order to keep the highest quality indels. The filtering is done in three parts: the exclusion of variants not called by all calling technologies available, the implementation of a genotype exclusion filter and the filtering through annotation. We began by removing all indels that were not called by all the variant callers. Therefore, the only remaining indels in SPARK were called by GATK, WeCall and DeepVariant and the remaining indels in SSC were called by both GATK and Freebayes. This method ensures that the SNVs have a much higher likelihood of being true indels because they were detected by multiple technologies (link GIAB and Krumm).

We followed this with genotype filtering, which is the process of filtering the indels according to genotype criteria such as total read depth, alternate allele depth, MRR, VQSR quality and the genome Aggregation Database (gnomAD) [80] allele frequency. While all the probands and siblings were filtered according to the same criteria, the parents were split into two filtering groups: Those carrying an indel that they transmit to their offspring and those that do not. In the case of parents who do not transmit a given indel, these criteria filter out any potentially low-quality variants, which have a higher risk of being false positives. However, this level of stringency can potentially create bias in terms of the transmission of a given indel in a child. For example, probands carrying an indel inherited from the parent would be re-labelled as carrying a *de novo* variant if the parents have been filtered out because they do not pass the cut-off points. If the level of stringency amongst parents with transmitted variants is too high, this can artificially inflate the number of *de novo* variants and create a bias. This is important because *de novo* variants are then filtered out of the dataset, as we do not consider them in our analysis.

The filtering criteria is as follows: for parents who carry a variant that is not transmitted, probands and siblings, we set a minimum threshold of 20 for the total read depth and 5 for the alternate read depth. For the parents that carry a variant that they transmit to a proband or sibling, the total read depth threshold is set at 10 reads and the alternate read depth is of 1 read. The following cut-offs apply to all individuals: a minimum of 0.05 for the MRR, a “PASS” for the VQSR quality and an allele frequency <=0.001 for both the parental minor allele frequency and the gnomAD minor allele frequency, which is the frequency of the variant found in the general population[81]. These initial filters are what we refer to as our “cold cut-off” method, which selects variants according to whether or not they meet the above threshold.

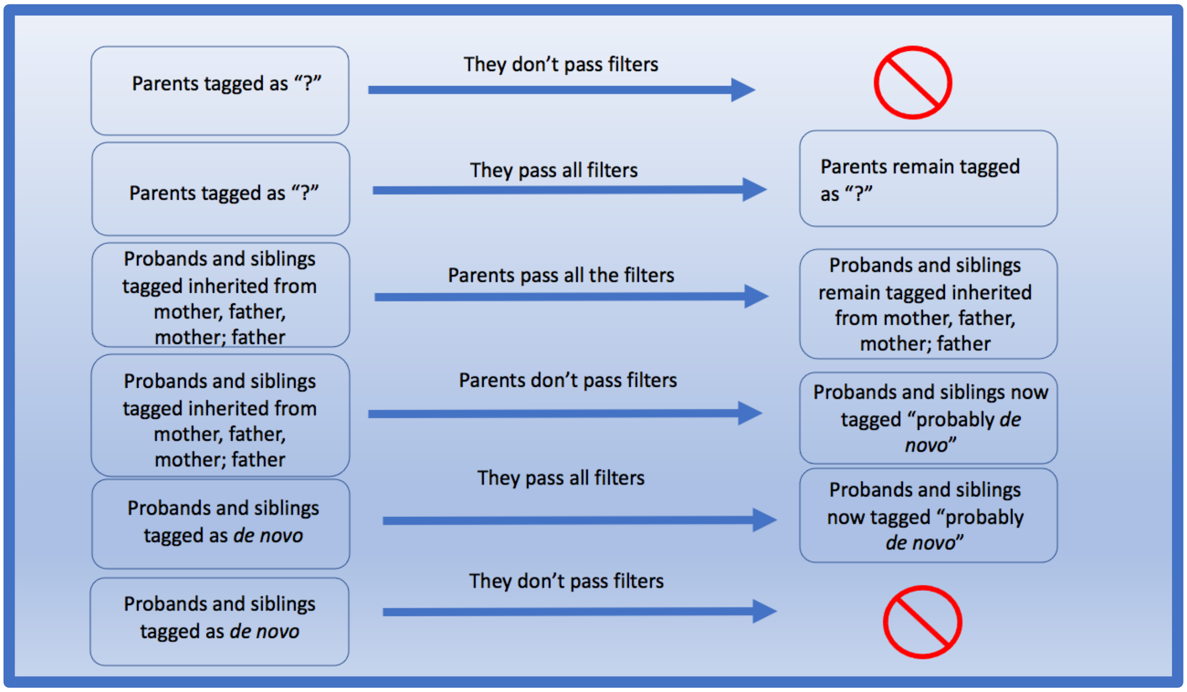

Parents tagged as "?"
They don't pass filters
Parents tagged as "?"
They pass all filters
Parents remain tagged as "?"
Probands and siblings tagged inherited from mother, father, mother; father
Parents pass all the filters
Probands and siblings remain tagged inherited from mother, father, mother; father
Probands and siblings tagged inherited from mother, father, mother; father
Parents don't pass filters
Probands and siblings now tagged "probably de novo"
Probands and siblings tagged as de novo
They pass all filters
Probands and siblings now tagged "probably de novo"
Probands and siblings tagged as de novo
They don't pass filters

Figure 1: Inheritance retagging process. The initial inheritance tag of the individual (inherited from the father, mother, both or *de novo*) is on the left and the consequence of the filtering is on the right. The arrows in the middle represent the action taken during filtering.

After filtering the SNVs by their genotype features, the next step is to annotate the variants in order to evaluate the consequences of the indels; to see whether there is any potential haploinsufficiency in the genes or whether the indel is an LoF variant. We used Ensembl's VEP annotator (release 99) with the following plugins: GeneSplicer for splicing loss of function predictions, dbNSFP (version 4) for pathogenicity predictions, and LOFTEE to estimate the confidence of LoF indels (see figure 2). We excluded any indels that were not tagged as frameshift variants, all "Low Confidence" LOFTEE calls and any GeneSplicer calls that were not "High".

Although applying quality and annotation filters allows us to discard the majority of lower quality indels, there will still be some regions where variant calling is unreliable due to the nature of the region or the variants. Therefore, we excluded all the following: regions found in the ENCODE blacklist[82], regions found in the NCBI “Sanger dead zones” [83], segmental duplications[84], repeats[85], centromeres[86] and pseudogenes[87] (see figure 2).

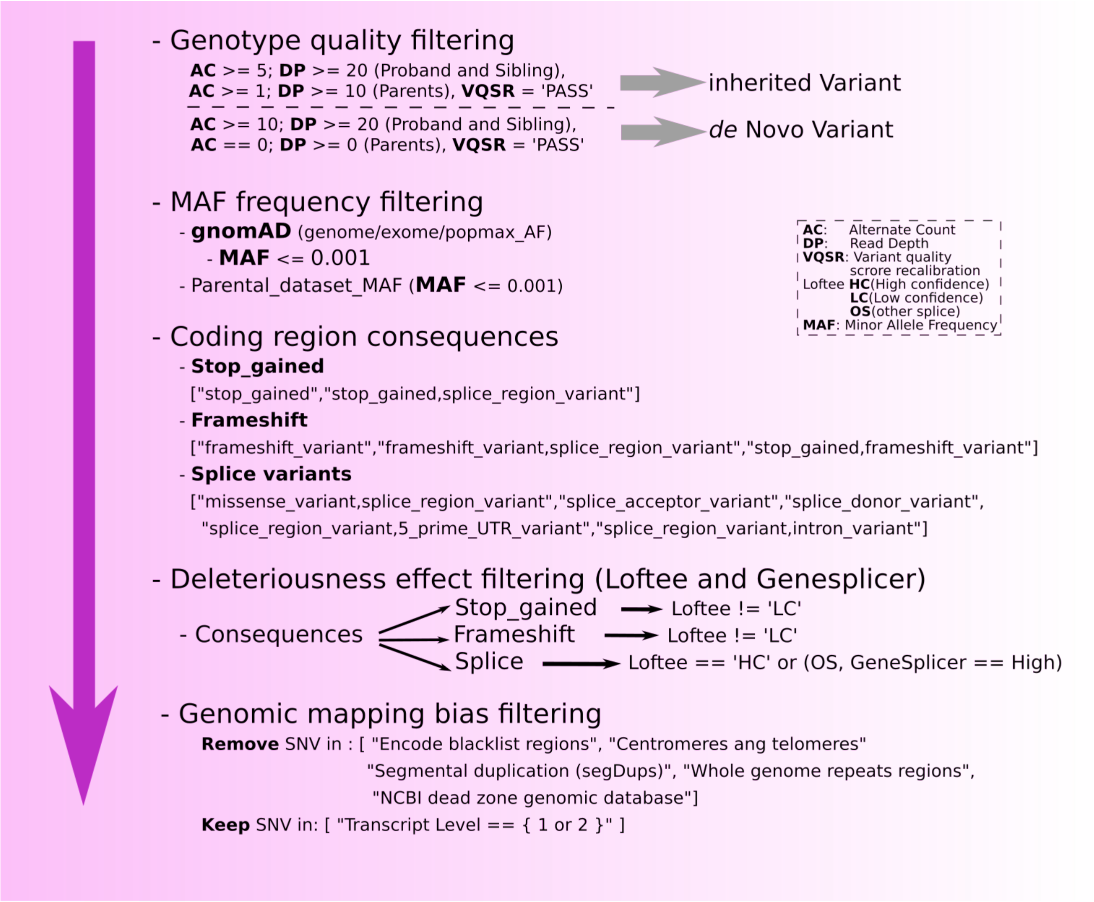

- Genotype quality filtering
AC >= 5; DP >= 20 (Proband and Sibling),
AC >= 1; DP >= 10 (Parents), VQSR = 'PASS'
inherited Variant
AC >= 10; DP >= 20 (Proband and Sibling),
AC == 0; DP >= 0 (Parents), VQSR = 'PASS'
de Novo Variant
- MAF frequency filtering
- gnomAD (genome/exome/popmax_AF)
- MAF <= 0.001
- Parental_dataset_MAF (MAF <= 0.001)
AC: Alternate Count
DP: Read Depth
VQSR: Variant quality scrore recalibration
Loftee HC(High confidence)
LC(Low confidence)
OS(other splice)
MAF: Minor Allele Frequency
- Coding region consequences
- Stop_gained
["stop_gained","stop_gained,splice_region_variant"]
- Frameshift
["frameshift_variant","frameshift_variant,splice_region_variant","stop_gained,frameshift_variant"]
- Splice variants
["missense_variant,splice_region_variant","splice_acceptor_variant","splice_donor_variant",
"splice_region_variant,5_prime_UTR_variant","splice_region_variant,intron_variant"]
- Deleteriousness effect filtering (Loftee and Genesplicer)
- Consequences
Stop_gained
Loftee != 'LC'
Frameshift
Loftee != 'LC'
Splice
Loftee == 'HC' or (OS, GeneSplicer == High)
- Genomic mapping bias filtering
Remove SNV in : ["Encode blacklist regions", "Centromeres ang telomeres"
"Segmental duplication (segDups)", "Whole genome repeats regions",
"NCBI dead zone genomic database"]
Keep SNV in: ["Transcript Level == { 1 or 2 }"]

Figure 2: Filtering and annotation pipeline. All filters and tags applied to the indels during the filtering and annotation process. This pipeline was used for SPARK, SSC and UKBB. Adapted from Jean-Louis, M., 2021, unpublished.

Following these filters, our “truth set” was established and we were able to use these variants to train our machine learning models.

The same filters were applied to the UKBB dataset with a few changes made. Considering the UKBB cohort was called using one variant calling technology, the maximum number of variant calling technologies was 1. Furthermore, as it is a general population cohort and not a family-based one, we did not split the genotype filters into two categories; the minimum cut-off value for the alternate allele depth was 5 and the minimum cut-off value for the total read depth was 20. The cohort was also not retagged for inheritance.

Table III: Number of indels after each filter

Cohort Filter	SPARK		SSC		UKBB	
	N	N per individual	N	N per individual	N	N per individual
No filter	101,869,571	3,736.55	16,420,192	1,892.16	3,103,405,081	15,468.22
Genotype filter	50,290,355	1,844.64	10,012,750	1,153.80	1,577,352,801	7,861.96
Frequency filter	511,484	18.75	75,986	8.75	549,592,333	2739.32
Annotated	84,857	3.11	18,901	2.18	13,456,233	60.87

Machine Learning Models

We used two machine learning models in order to predict which indels have the highest likelihood of being true: a logistic regression model, and a random forest model. We also took the result of the intersection of the prediction of the two models.

A logistic regression model is a supervised learning classifier that makes a binary prediction, where 1 is a success and 0 is a failure[88]. It is a simple model that can be used for many classification purposes. In our case, a 1 represents a true positive indel and a 0 is a true negative indel.

A random forest model is an ensemble supervised learning classifier that uses multiple decision trees in order to make a prediction. Each decision tree makes a prediction of its own and the model chooses the best prediction by means of a vote. One of the strengths of the random forest model is that it is able to correct for overfitting by the use of the multiple decision trees. Unlike the logistic regression model, it is capable of making multiple predictions, not just binary ones[89]. However, the predictions will be made in the same way as that of the logistic regression model (1 for true positives, 0 for true negatives).

The intersection of the two methods is manual set of predictions we are making based on the results of the logistic regression model and the random forest model. If both models predict a true positive, then the intersection will predict a true positive as well. Otherwise, the prediction is set as a true negative.

The set of features used to make the predictions follow two rules: they must be numerical in value and they must not be correlated. Numerical features are important because machine learning models cannot perform arithmetical operations on string data. Therefore, all non-numerical features have been transformed into numerical values. Moreover, the values must not correlate, as correlation can hinder the capacity of a model to make an accurate prediction, due to a tendency to overtrain[90].

Training

Having established our truth set through filtering, annotated relabeling, we were left with a set of 103,760 indels. These indels have all been called by the maximum number of variant calling algorithms for their cohort, they passed all the quality thresholds, they were tagged as frameshift variants (see figure 2) and none are *de novo* indels. This set was then used to train a logistic regression model that would be used to predict the likelihood of an indel being a true positive. 50% of the training dataset is SPARK and SSC variants that had passed all the filters, whereas the other 50% was a random selection of normalised SPARK and SSC variants that did not pass the filtering process (n=103,760). The filtered indels were tagged as positives (labelled as "1") whereas the variants that had been filtered out were tagged as negatives (labelled as "0"). The features used to train the model are the following: alternate allele depth (AC), MRR, the count of each nucleotide in a given INDEL, the GC% and the length of the variant. Using the scikit-learn package (version 0.20.4) of python (version 2.7), the dataset was split 80%-20% into a training set and a test set, respectively. We chose this split because of the relatively small dataset at hand; with an 80/20 split, we had enough of data points to train on, without compromising the quantity of data on which to apply our prediction, as in the case of a 90/10 split. A k-fold cross validation was done in order to ensure that we selected the appropriate number of groups the data was split into. Before commencing training, a correlation analysis was done on the features to determine whether there was any intra-feature correlation using the .corr() command of scikit-learn.

Once it was established that there was no strong correlation between the features, we trained the logistic regression model on the 80% train set. Once trained, we evaluated the model by making predictions on the test set, all through scikit-learn. The parameters for evaluation were the feature importance, a confusion matrix and by the area under the curve (AUC) of the receiver operating characteristic (ROC) curve. The feature importance of each variable, which is the evaluation method, is the coefficient value, which indicates the relationship between the predictor and the prediction. A negative value implies that the feature is more important in the prediction of a failure ("0" label) whereas a positive value implies that the feature is more important in the prediction of a success (“1” label). The second method, the confusion matrix, is a heat map of the predicted labels vs the true labels we’ve assigned them. Similar numbers will have similar colours. Finally, the ROC is a probability curve that models the relationship between the true positive rate (TPR) otherwise known as the sensitivity (y-axis) and the false positive rate (FPR) otherwise known as the specificity (x-axis). The AUC is a measure of how well the model is able to distinguish the two classes. The higher the AUC, the better it is at determining which of the two classes a particular variable belongs to.

After training and testing the logistic regression model, we trained a random forest model using the same train dataset. This model was evaluated in the same way as the logistic regression; by evaluating the feature importance, creating a confusion matrix and a ROC curve. After training this model, we created a new prediction by looking at the overlapping prediction between the two models. If both models had labelled an indel as a true indel (label “1”), then the prediction was that it was a true indel. Otherwise, it was predicted to be a false indel (label “0”). We were thus left with 3 possible predictions for each variant. A confusion matrix was created for each method of prediction, as well as a calculation of their accuracy.

Table IV: Number of indels predicted by each model

Model / Cohort	SPARK + SSC (20%)		UKBB	
	N	N per individual	N	N per individual
Logistic Regression	22,871	1.61	12,803,219	57.43
Random Forest	21,455	1.44	2,593,701	10.88
Intersection	17,030	1.57	2,497,290	10.41

Table V: Distribution of feature values in SPARK and SSC cohorts

Features / Model	Random Forest			Logistic Regression			Intersection			Cold cut-off		
	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean
Length	-83	17	-1.31	-122	24	-1.52	-32	16	-1.65	-122	144	-0.69
AC	2	295	28.23	1	447	29.2	1	295	31.29	1	447	21.82
MRR	0.07	1.00	0.46	0.05	1	0.42	0.05	0.79	0.46	0.05	1.00	0.52
A	0	7	0.36	0	8	0.32	0	6	0.30	0	34	0.51
C	0	8	0.47	0	18	0.45	0	8	0.43	0	44	0.58
G	0	9	0.39	0	21	0.35	0	9	0.33	0	67	0.51
T	0	11	0.37	0	4	0.31	0	4	0.29	0	28	0.53
GC%	0	1	0.54	0	1	0.55	0	1	0.56	0	1	0.51

AC=alternate allele depth, MRR=mutant read ratio, A=number of A nucleotides, C=number of C nucleotides, G=number of G nucleotides, T=number of T nucleotides, GC%=ratio of G and C nucleotides over total number of nucleotides

Table VI: Distribution of feature values in UKBB cohort

Features Model	Random Forest			Logistic Regression			Intersection			Cold cut-off		
	Min	**Max**	**Mean**	**Min**	**Max**	**Mean**	**Min**	**Max**	**Mean**	**Min**	**Max**	**Mean**
Length	-541	20	-2.388	-154	47	-1.62	-121	20	-1.72	-556	59	-6.63
AC	5	260	19.02	5	317	10.15	5	260	19.12	5	317	10.26
MRR	0.06	1.00	0.22	0.06	1	0.12	0.06	1.00	0.21	0.06	1.00	0.13
A	0	11	0.27	0	8	0.25	0	8	0.26	0	20	0.25
C	0	13	0.47	0	27	0.33	0	13	0.45	0	27	0.34
G	0	11	0.58	0	19	0.38	0	11	0.58	0	30	0.39
T	0	11	0.175	0	6	0.32	0	6	0.15	0	18	0.33
GC%	0	1	0.69	0	1	0.53	0	1	0.70	0	1	0.53

AC=alternate allele depth, MRR=mutant read ratio, A=number of A nucleotides, C=number of C nucleotides, G=number of G nucleotides, T=number of T nucleotides, GC%=ratio of G and C nucleotides over total number of nucleotides

Genetic Scores

We used three genetic scores in order to determine the indel effects on cognition: a modified version of the "loss-of-function observed/expected upper bound fraction" (LOEUF) score (1/LOEUF), the cortical differential stability score (DS_C) and the average number of genes affected per individual. Furthermore, we evaluated the effect on the IQ of each gene according to their tolerance, based on the LOEUF score. The scores were compiled by calculating the sum of each score for the affected genes in each individual.

The LOEUF score is the estimate of the number of observed LoF variants over the number of expected LoF variants, based on the upper-bound of a Poisson-derived confidence interval. The smaller the LOEUF score, the more deleterious the effect. The LOEUF score can be anywhere between 0.03 to 2 and is deleterious from 0.03 to 0.035[91]. We therefore use a modified version of the score in order to simplify our understanding of it; seeing as we use the sum of scores for each gene, by inversing the score (1/LOEUF), we have a more deleterious effect the higher the LOEUF score is. Therefore, it is better adapted to our usage.

The DS_C score is a correlation-based metric which assesses reproducibility of regional patterns of gene expression in the cortical brain structures[92]. This score was computed as the mean pairwise correlation between gene expression patterns of six adult human brains from the Allen human brain atlas project[93]. The highest scores represent the genes with a stable regional expression in the 6 adult brains, and the lowest scores represent the genes with a non-specific regional expression across the human cortex.

In order to evaluate the IQ according to gene tolerance, we first divided all genes into 4 categories: highly intolerant genes (LOEUF<0.2; n=1,088), moderately intolerant genes (0.2≤LOEUF<0.35; n=1,898), tolerant genes (0.35≤LOEUF<1; n=7,710) and highly tolerant genes (LOEUF≥1; n=8,501).

Statistical Analysis

We began by evaluating the effect of the SPARK and SSC indels on IQ based on the predictions made by each model. We created three different sets of the test SPARK and SSC indels: One of all the indels that were predicted as true positives using the logistic regression model, one of all the indels predicted as true positives by the random forest model, and one of the intersection of the true positive predictions of the logistic regression and random forest models.

We then applied a linear regression model to determine whether the predicted indels show any effect on IQ, based on the LOEUF and DS_C score and on the number of genes affected. This was computed using the lm() function of R (version 3.6.1). The model could be written as follows:

$$IQ \sim \alpha X + \beta Score$$

where IQ is the z-score of a standardised measure of intelligence, X represents any covariates in the model (sex and ancestry) and the βscore is any of the three scores. Individuals who did not have an IQ value were eliminated from the analysis (see table I).

Afterwards, using a linear regression, we evaluated the effect of all the indels in the UKBB cohort before any predictions were made, in order to compare these results with the effect on IQ of the predicted indels. This cohort was filtered and annotated in the same method as SPARK and SSC. After this initial evaluation, we created three different datasets of the UKBB cohort, as with the SPARK and SSC set, in order to evaluate the strength of the models. Any individual in the cohort for which we did not have an IQ value was eliminated from our analysis (see table I). Our analyses correct for both sex and ancestry for all individuals in all cohorts.

The result of the linear model shows an estimate of the effect on IQ. The higher the absolute value of the estimate, the stronger the impact on cognition. In terms of IQ, a high estimate means a bigger loss of IQ. Therefore, when mentioning the effect on IQ, we are referring the amount of loss of IQ in an individual.

CHAPTER 3: RESULTS

Results

Model Features

The correlation matrix (figure 3) shows that there is no strong correlation (> 0.7) between each feature of the model, with the exception of some correlation between the number of A nucleotides and GC% as well as the number of T nucleotides and GC%. The most important feature in the logistic regression model is the MRR, followed by the number of T nucleotides. Both of these features tend to be important in the prediction of a true negative (label “0”). The most important feature in the random forest model is the MRR, but it is closely followed by the AC, and then the length. All the features are important in the prediction of a true positive model (label “1”).

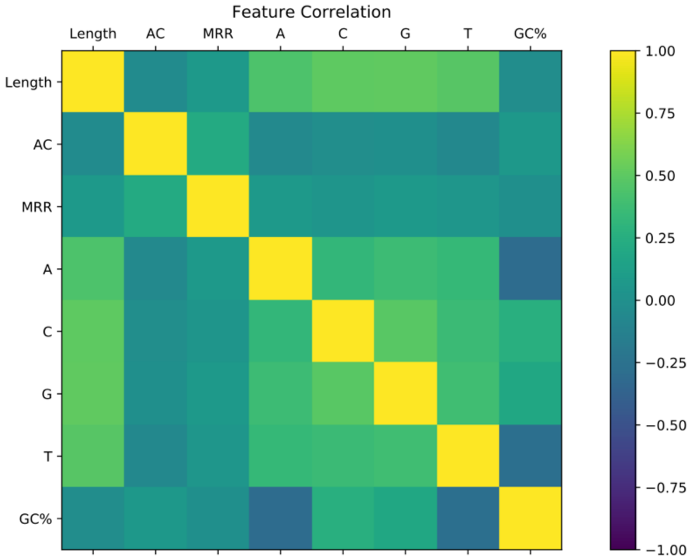
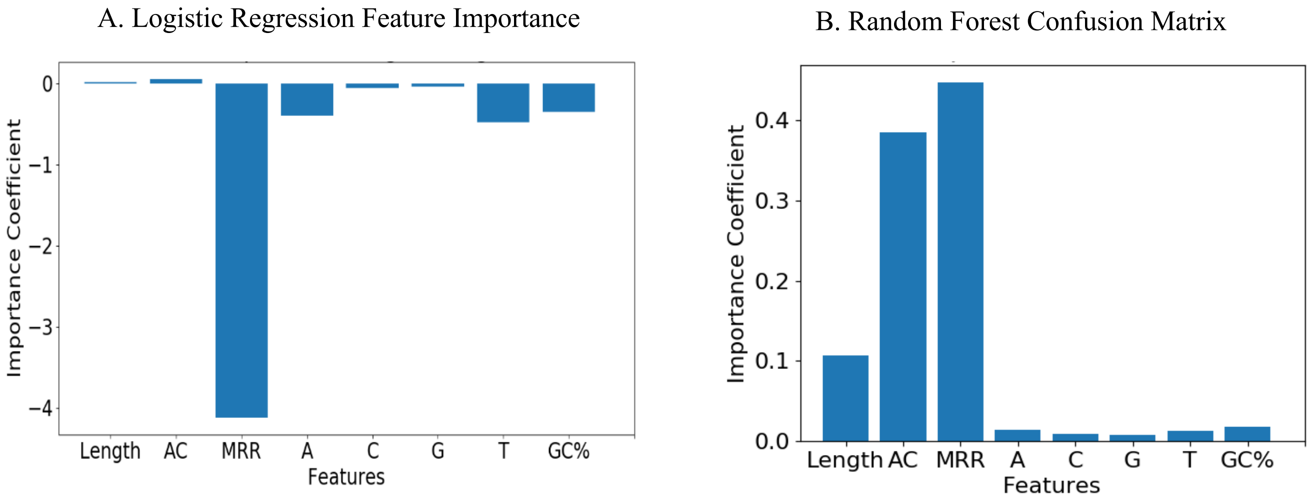


Figure 3: Correlation matrix of features. Yellow represents a positive correlation and purple is a negative correlation. The closer a value is to 1 or -1, the stronger the correlation.



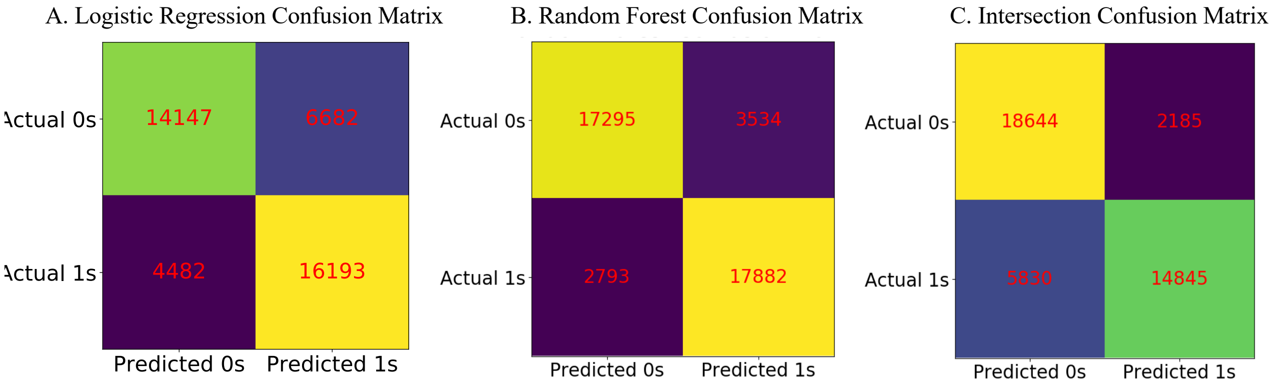
A. Logistic Regression Feature Importance

B. Random Forest Confusion Matrix

Figure 4: The importance of each feature in a machine learning model. (A) The feature importance for the logistic regression model. A negative importance coefficient means the feature is important in the prediction of a true negative (B) The feature importance for the random forest model. A positive value means the feature is important in the prediction of a true positive. AC represents the alternate allele depth; MRR is the mutant read ratio (alternate depth/total depth); A, C, G, T are the numbers of each nucleotide in an indel, respectively; GC% is the ratio of the number of G and C nucleotides over the total length of the nucleotide

Model Accuracy

The confusion matrix of the logistic regression model (figure 5.a) shows that of the total 22,875 true positive predictions, 16,193 of them were concordant with those we labelled as true positives. Out of the total 41,504 predictions, 30,340 of them were concordant with the label we gave them. This gives us an accuracy of 73.10%. The confusion matrix generated by the random forest model (figure 5.b) shows us that the model made a total of 21,416 true positive predictions, of which 17,822 are actual positives. Out of the total 41,504 predictions made, 35,177 were correct, which gives us an accuracy of 84.76%. The confusion matrix generated by the intersection model (figure 5.b) shows that the model made a total of 17,030 true positive predictions, of which 14,845 were correct. Out of the total 41,504 predictions made, 33,489 of them were correct, which indicates an accuracy of 80.69%.

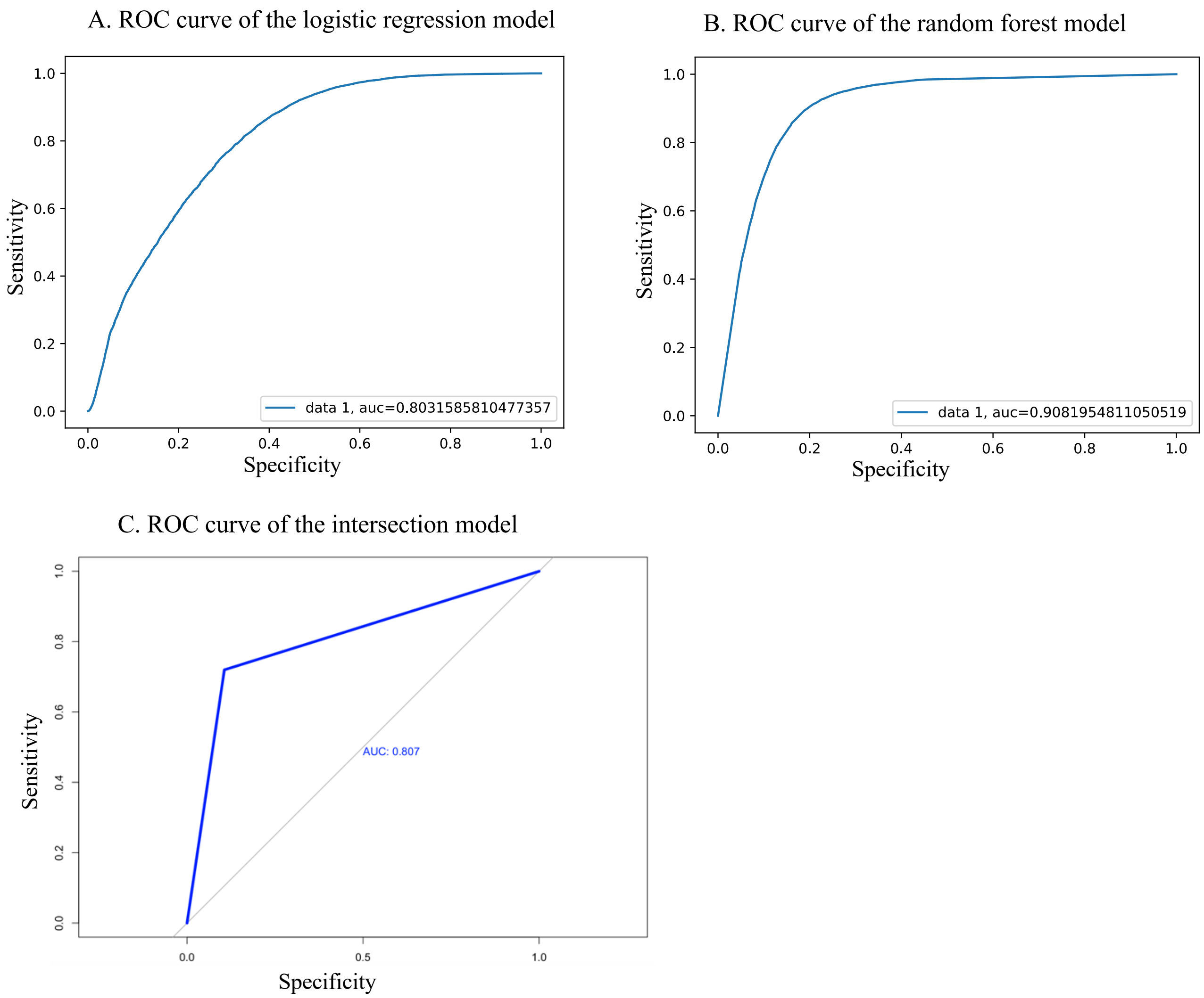

A. Logistic Regression Confusion Matrix

B. Random Forest Confusion Matrix

C. Intersection Confusion Matrix

Figure 5: The number of predictions for each model. (A) Logistic regression predictions vs actual values. (B) Random forest predictions vs actual values. (C) The intersection of random forest and logistic regression predictions vs actual values. 1 represents a true positive indel, 0 represent a false negative. Similar colours represent similar values

The ROC curve of the logistic regression model shows an AUC of 0.80, the ROC curve of the random forest model has an AUC of 0.91 and the ROC curve of the intersection of the two has an AUC of 0.81.


A. ROC curve of the logistic regression model

B. ROC curve of the random forest model

C. ROC curve of the intersection model

Figure 6: ROC curves of different models. (A) The ROC curve of the logistic regression model (B) the random forest model (C) the intersection of the two models. The specificity represents the true positive rate (TPR) of the model, and the sensitive is the false positive rate (FPR) for all three plots.

The average number of genes per individual in the SPARK and SSC dataset that have been filtered by cold cut off (see figure 2) shows an average of 2.19 genes per individual. In the case of UKBB, the average is 60.87 genes per individual. The SPARK and SSC predictions made by the logistic regression model show an average of 1.61 whereas the UKBB logistic regression predictions show 57.43.

The SPARK and SSC dataset as predicted by the intersection model shows an average of 1.44 genes per individual while dataset predicted by the random forest model shows an average of 1.57. The UKBB dataset as predicted by the intersection model shows an average of 10.41 genes per individual, whereas the dataset predicted by the random forest model has an average of 10.88

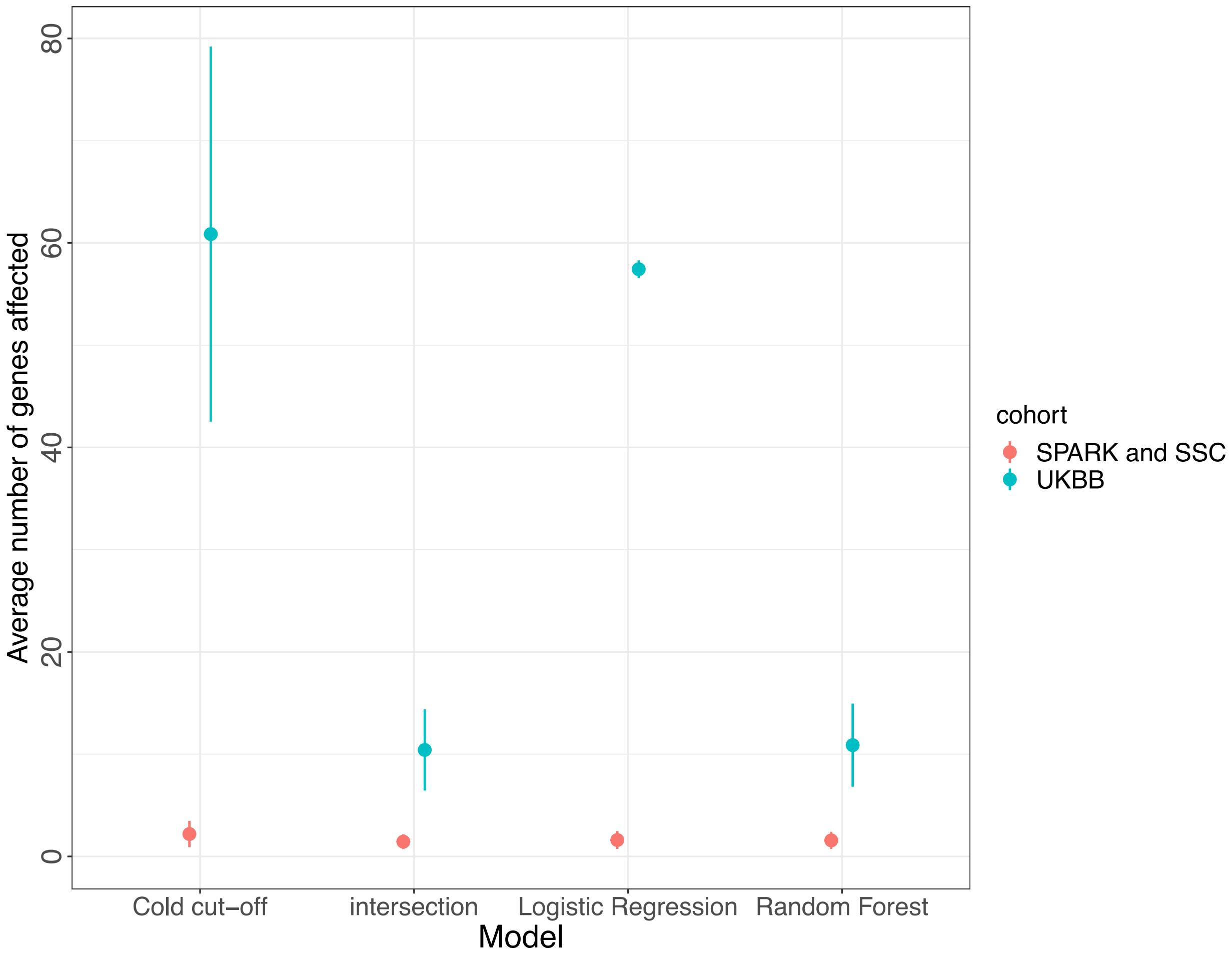


Figure 7: Average number of genes per cohort. The average number of genes per cohort as predicted by each model, according to the model used to predict indels. SPARK and SSC are in pink, and UKBB is in blue.

Effect on IQ

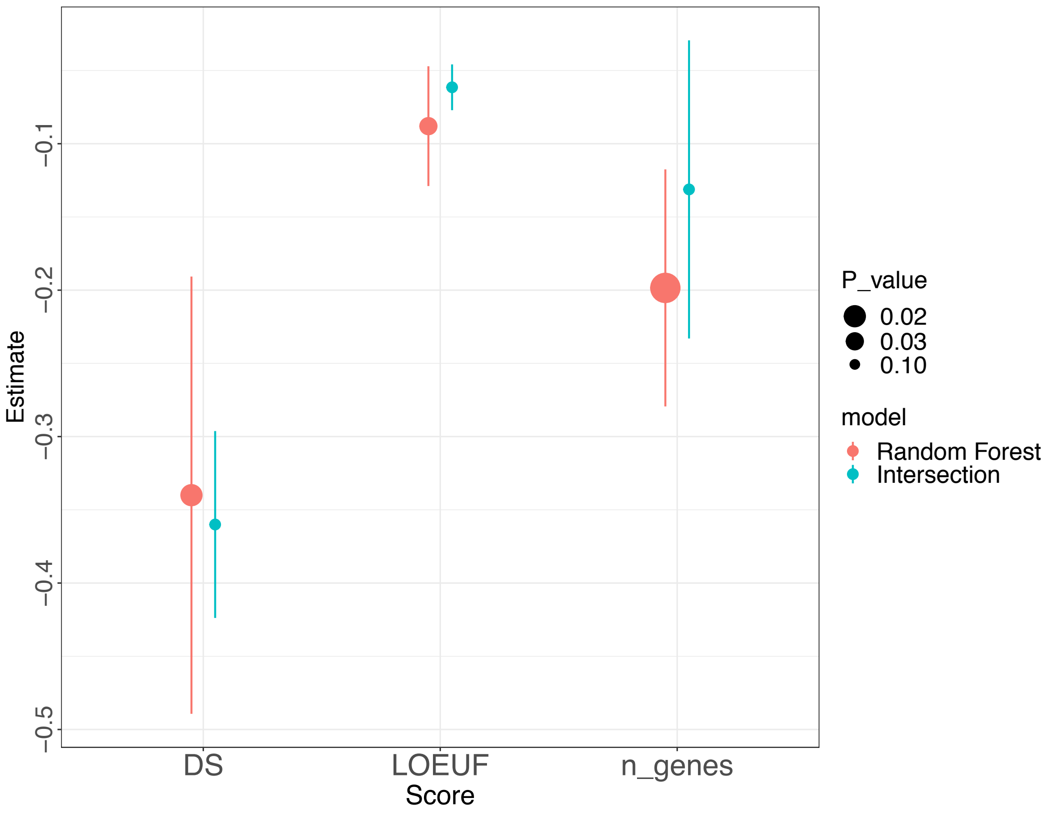


Figure 8: Effect of indels on IQ in SPARK and SSC cohorts. The estimate is calculated based on each score. The random forest model is in red and the intersection model is in blue. P-values range according to the size of each point; the bigger the point, the smaller the p-value.

We evaluated the effect of the indels predicted by the random forest model and intersection of the two models on IQ using a linear regression. The indels predicted using the random forest model had a significant impact on IQ when evaluated based on the DS_C score and the LOEUF score (DS_C: est.=-0.34, 95% CI=[-0.63, -0.05], p=0.02; LOEUF: est.=-0.09, 95% CI=[-0.17,-0.007], p-value=0.03). Furthermore, the number of genes affected by the indels shows a significant effect on IQ (est.=-0.2, 95% CI=[-0.35,-0.04], p-value=0.014) for the random forest predictions as well. However, when considering the intersection of the random forest and linear regression, we see no impact when evaluating the two scores (DS_C: est.=-0.36, 95% CI=[-0.48,-0.23], p-value=0.2; LOEUF: est.=-0.06, 95% CI=[-0.09,-0.03], p-value=0.38) and when evaluating the number of genes affected (est.=-0.13, 95% CI=[-0.33,0.06], p-value=0.2).

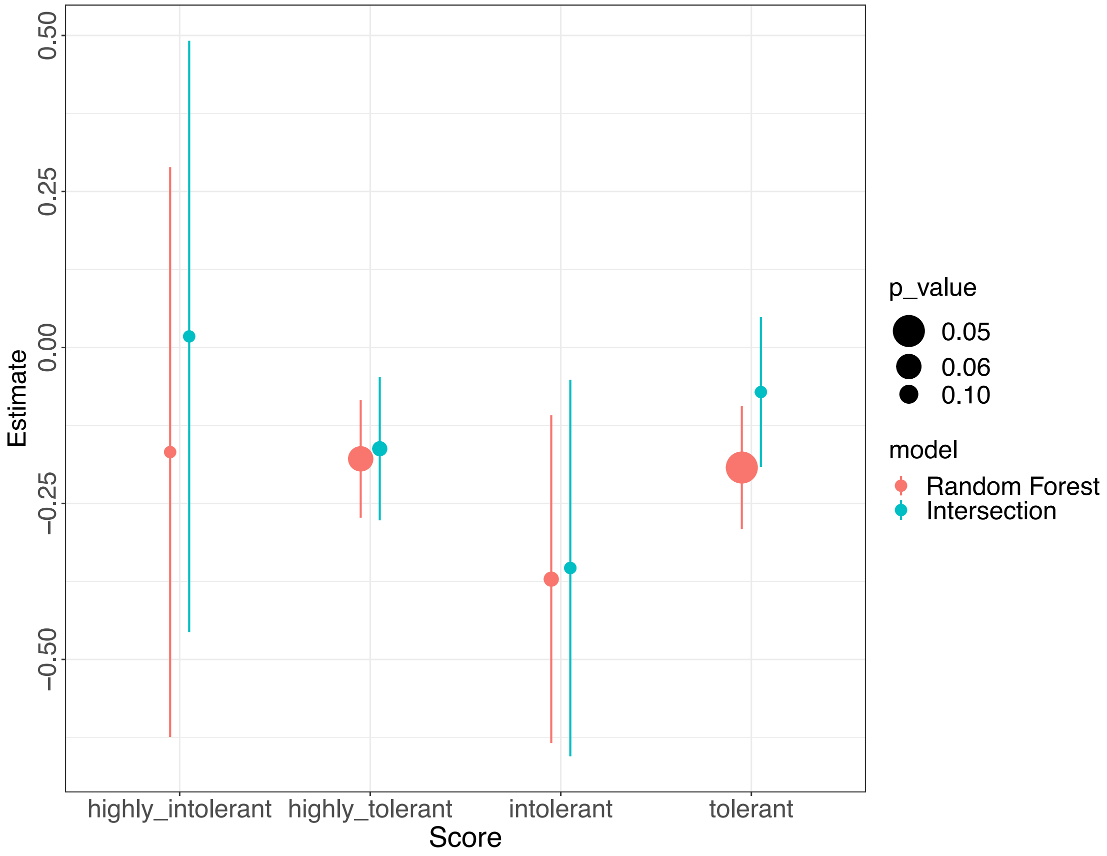


Figure 9: Effect on IQ According to Gene Tolerance in SPARK and SSC. The estimate is calculated based on gene tolerance. Indels predicted by the intersection model are in blue and indels predicted by the random forest model are in pink. P-values range according to the size of each point; the bigger the point, the smaller the p-value.

We evaluated the predictions of the intersection and random forest model by evaluating the impact of IQ according to the tolerance of each gene. The indels predicted by the intersection model have no significant effect on IQ regardless of gene tolerance (HI: est.=0.018, 95% CI=[-0.91;0.95], p=0.97; I: est.=-0.35, 95% CI=[-0.95;0.24], p=0.24; T: est.=-0.07, 95% CI=[-0.31;0.16], p=0.55; HT: est.=-0.16, 95% CI=[-0.39;0.06]). The indels predicted by the random forest show no significant effect on the IQ for the highly intolerant, intolerant and highly tolerant genes (HI: est.=-0.17, 95% CI=[-1.06;0.72], p=0.71; I: est.=-0.37, 95% CI=[-0.89;0.14], p=0.16; HT: est.=-0.18, 95% CI= [-0.36;], p=6e-02) and only slightly significant for the tolerant genes (est.=-0.19, 95% CI=[-0.39;1.5e-03], p=0.05).

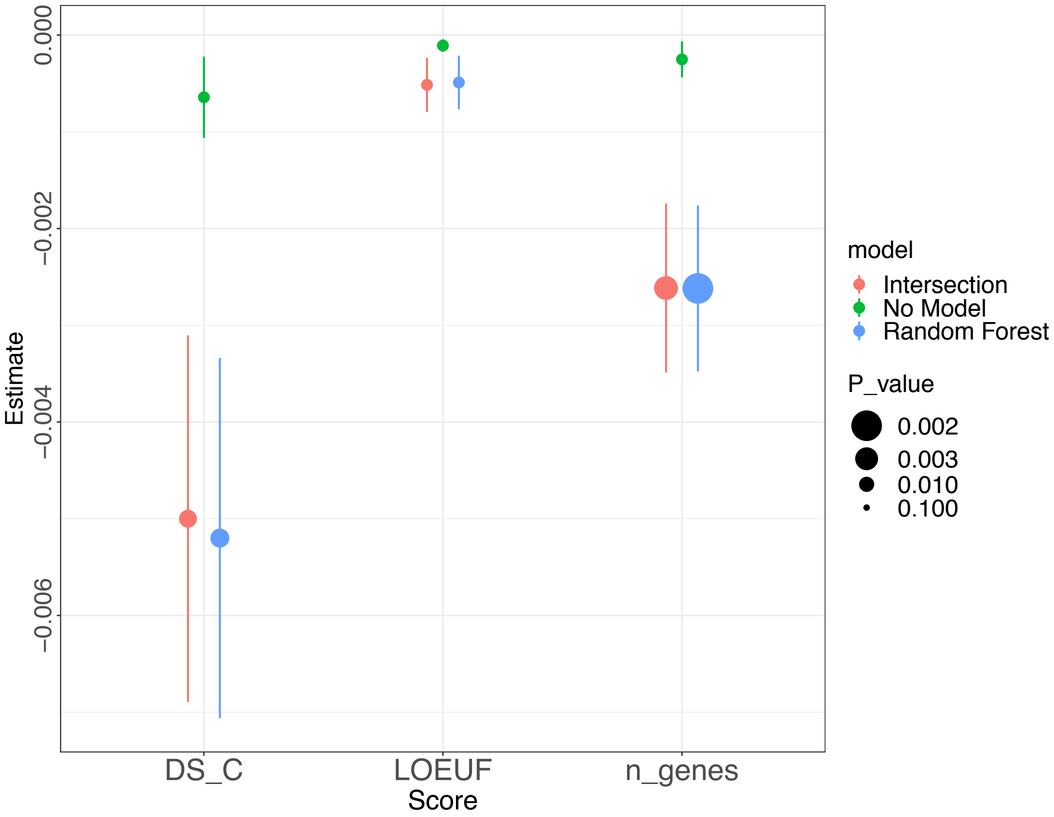


Figure 10: Effect of indels on IQ in the UKBB cohort. The estimate is calculated based on each score. The indels predicted by the intersection model are in pink, the indels predicted by the random forest model is in blue and the indels not predicted by any model (cold cut-off) are in green. P-values range according to the size of each point; the bigger the point, the smaller the p-value.

When looking at the indel effect on IQ after the application of the filters (see figure 2) but before the application of a machine learning model, we can see that there is no significant relationship between any of the scores and IQ (DS_C: est.= -6.4e-04, 95% CI=[1.4e-03, 7e-04], p=0.12; LOEUF: est.=-1.084e-04, 95% CI=[-2.3e-04, 2.0e-05], p=0.1) and no relationship between the number of genes affected and IQ (est.=-2.5e-04, 95% CI=[-6e-04, 1.15e-04], p-value=0.18).

The effect of the indels predicted by the random forest model on IQ shows a significant relationship for the DS_C score and IQ (est.= -5.2e-03, 95% CI= [-8.9e-03; -1.6e-03], p=0.005) but no significance between the LOEUF score and IQ (LOEUF: est.=-1.084e-04, 95% CI=[-9e-03, -1.5e-03], p-value=0.08). Furthermore, there is a significant effect on IQ when considering the number of genes that have been affected (est.=-0.003, CI=[-0.004, -9.34e-4], p-value=0.002).

The effect of the indels predicted by the intersection model on IQ shows a significant relationship for the DS_C score and IQ (est.= -0.005, 95% CI= [-8.9e-03; -1.5e-03], p=0.006) but no significance between the LOEUF score and IQ (LOEUF: est.= -5.146e-04, 95% CI=[-1.0e-03; 3.3e-05], p=0.06). There is a significant relationship between the number of genes affected and the IQ (est.=-2e-03, CI=[-4.0e-03, -9.03e-04], p-value=0.003).

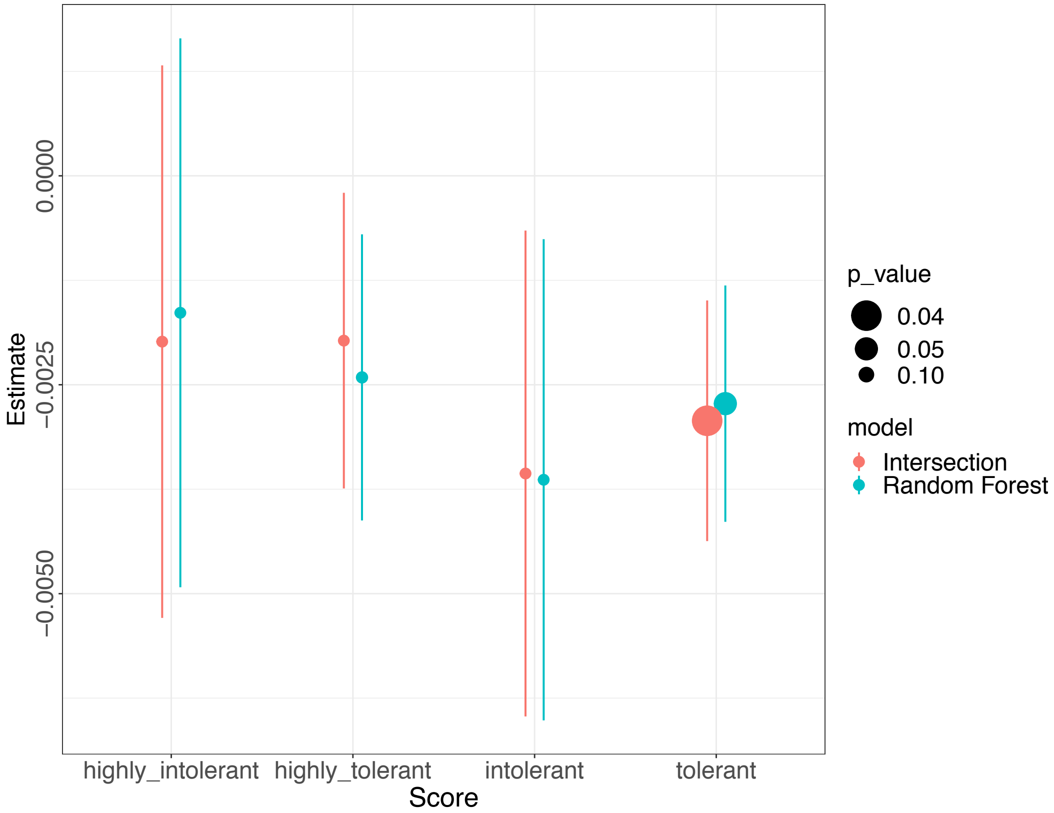


Figure 11: Effect on IQ According to Gene Tolerance in UKBB. The estimates are calculated based on each score. Indels predicted by the intersection model are in blue and indels predicted by the random forest model are in pink. P-values range according to the size of each point; the bigger the point, the smaller the p-value.

When measuring the effect of gene tolerance of the indels predicted by the intersection model on IQ , we see that there is no significant relationship for the highly intolerant, intolerant and highly intolerant genes (HI: est.=-2e-03, 95% CI=[-8e-03;4.5e-03], p=0.55; I: est.=-3.5e-03, 95% CI=[-9e-03;2.1e-03], p=0.22; HT: est.=-2.0e-03, 95% CI=[-5.4e-03;1.5e-3]) but there is a small but significant effect when considering the tolerant genes (est.=-3e-03, 95% CI=[-5.7e-03;-1.1e-04], p=0.04;). The results are similar for the indels predicted by the random forest model; highly intolerant, intolerant and highly tolerant genes show no significant effect (HI: est.=-1.6e-03, 95% CI=[-8e-03;4.8e-03], p=0.62; I: est.=-3.6e-03, 95% CI=[-9.3e-03;2e-03], p=0.22; HT:

est.=-2.4e-03, 95% CI=[-5.76e-03;9.4e-04], p=0.16), with tolerant genes showing a small but significant effect on IQ (est.=-2.7e-03, 95% CI=[-5.5e-03;4.5e-05], p=0.05).

CHAPTER 4: DISCUSSION AND CONCLUSION

Discussion

Model Features

The first step of the work was the selection of a set of features which would be used to describe our indels, on which I would base my predictions. The current set of features (see figure 3) shows that there is no strong correlation between any of the features, save for a small correlation between the number of A nucleotides and GC% and the number of T nucleotides and GC%. This correlation is to be expected as the GC% is the ratio of the number of G and C nucleotides over the total number of nucleotides.

When assessing the importance of each feature, the logistic regression model shows that the most important feature is the MRR, which could lead to potential bias in the model. Indeed, when compared to the random forest model, for which there are three important features (MRR, AC and length), the logistic regression model does not perform as well, showing worse accuracy (see figure 5). This could be explained by the random forest's ensemble method of machine learning; by taking into consideration multiple decision trees, it is able to correct for any overfitting, which is not possible in the logistic regression model.

Model Accuracy

After determining the best set of features for prediction, the next step was to test the accuracy of each model. While none of the models were completely inaccurate (see figure 5), the random forest and the intersection models showed the best accuracy with 84.76% and 80.69% respectively. Seeing as the intersection of the model represents both the random forest predictions and the logistic regression predictions, it's understandable that the accuracy is lower; the logistic regression prediction had a smaller accuracy than the random forest, therefore it could have biased the intersection results.

The ROC curve of each of the three models shows that the models are indeed good at categorising indels as true positives or true negatives. When basing our evaluation on both the AUC and the accuracy of each model, we can see that the random forest and intersection models are stronger. Although the AUC of the linear regression model and the intersection are similar, the accuracy of the intersection makes it a better choice than the linear regression.

The predictions made by both the random forest model and the intersection model show a significant reduction in the average number of genes affected per individual for all the cohorts. For the SPARK and SSC datasets, the predicted indels have half the average value as those that have been selected through cold cut-off only, which are the variants that pass a quality threshold that we've set. The criteria is as follows: a minimum threshold for 20 for the total read depth and 5 for the alternate read depth for all probands, siblings and parents carrying a variant that is not transmitted; a minimum threshold of 10 for the total read depth and 1 for the alternate read depth for parents that carry a variant they transmit to a proband or a sibling; a minimum ratio of 0.05 for the MRR for all individuals; a "PASS" in VQSR quality for all individuals; and an allele frequency <=0.001 for both the parental minor allele frequency and the gnomAD minor allele frequency. For the UKBB dataset, the cold cut-off values have an average of 60.87, whereas the random forest and intersection predicted indels have an average of 10.88 and 10.41 respectively, showing a six-fold decrease. This suggests that the models are able to clean up the dataset more efficiently than simply using threshold values; a reduction in the average number of genes affected will show a reduction in noise during analysis.

Effect on IQ

The effect on IQ was first measured using the predicted indels in SPARK and SSC. The random forest model shows a significant effect when basing our model on the LOEUF score, the DS_C score and the number of genes affected by the predicted indels, thus implying that the model was able to detect and identify indels that have a higher likelihood of being true positive indels. However, when evaluating the effect of gene tolerance on IQ, although there is a slight significance for tolerant genes, the gene tolerance for most categories has no significant effect on IQ, suggesting that there is not enough power to explain any cognitive effect. The results of the random forest model tend to suggest that it is not so much the quality of the genes that affects IQ, but rather the quantity. This is further supported by the fact that the p-value of the relationship between the number of genes affected and the IQ is smaller than for any other significant score.

Although there is no significant effect on IQ in the intersection of the two models for the LOEUF and DS_C scores, the number of genes and the tolerance of the genes, the negative effect sizes suggest that the indels predicted by the model have a higher likelihood of being true positive, high-quality indels. This lack of significance could once again be explained by the lack of power in the analysis, considering the smaller number of indels predicted (see table IV).

Following SPARK and SSC, the effect of IQ of all the filtered and annotated indels of the UKBB cohort was measured in order to compare those results with those of the predicted indels. The indels without prediction show a very small negative effect size, but no significant effect on IQ. Considering that these indels were called using only one calling algorithm, these results could be biased by a high number of false positives. Furthermore, UKBB is a general population cohort (meaning there is no diagnosis requirement to be included), therefore, the vast majority of the called indels could be benign, with no significant effect on gene expression. Finally, since the average number of genes affected by the indels is quite high, this lack of significance could also be explained by the noise found in the cohort.

When evaluating the effect of the indels predicted by random forest on IQ, however, we can see that there is a significant decrease in IQ when we assess the model using the DS_C score and the number of genes. However, the effect size is quite small, and the uncertainty is high, as demonstrated by the large confidence intervals. Furthermore, although the LOEUF score shows no significant decrease in IQ, the p-value decreases, as opposed to the p-value with no machine learning predictions, suggesting an improvement in quality of indels. This could also be a consequence of the sample size, as the random forest predictions represent a much smaller number of variants when compared to the variants that were selected through cold cut-off. When estimating IQ according to gene tolerance, we see that there is only a mildly significant effect when considering tolerant genes. However, the effect size is small, and the p-value is borderline significant (p=0.05), therefore the effect has no tangible impact on the IQ. The large confidence interval can be explained by the small number of genes in each category, particularly for the highly intolerant and intolerant genes.

The indels predicted by the intersection model show a significant effect on the loss of IQ when based on the DS_C score and the number of genes, thus suggesting this model is also able to predict higher quality indels, although the effect sizes are small. And while there is no effect on IQ when considering the LOEUF score, it is important to note that the p-value of the effect is smaller for the intersection model than it is without any model or the random forest model, implying that the intersection model was able to predict higher quality indels that have an effect on IQ across all scores. As with the random forest model, the intersection model shows that tolerant genes have a small, mildly significant effect on IQ; it could almost be negligeable.

These results suggest that, while the indels predicted by the random forest model and the intersection are more likely to be high-quality, true positive indels than those simply filtered with cold cut-off value, the model is not quite able to predict indels that will show a strong effect size on the IQ. The results of the gene tolerance and the effect of the number of genes on IQ further testify to the additive nature of IQ, as suggested by the SPARK and SSC results.

It is important to note that by removing *de novo* variants from the analysis, the effect sizes will have a tendency to be smaller. *De novo* variants tend to have a more extreme effect on genes[60], thus leading to bigger effect sizes when estimating IQ. Therefore, it is not surprising to see small effect sizes, even when the results are significant. However, the effect sizes in the SPARK and SSC cohort are larger than those in UKBB; being autism-diagnosis based cohorts, it is to be expected as ASD presents a comorbidity with loss of IQ[94].

Moreover, of all the individuals in SPARK and SSC, only 4,944 individuals out of a total of 35,941 have an IQ value that we were able to use in our analysis. Therefore, the small effect sizes could be a problem due more to the lack of power in the analysis, rather than the strength of the models. This could potentially explain why the intersection model showed more significant results for UKBB than for SPARK and SSC. We therefore expect to find a stronger association between the scores and the loss of IQ if we have more individuals with an IQ value.

Measuring the effect on IQ is one of the indirect ways of assessing the strength of the model, as our general assumption is that true quality indels are more likely to show an effect on cognition. However, considering UKBB is a general population cohort, the model could be predicting truly, high-quality indels that simply do not have an effect on IQ; these indels could affect different aspects of cognition, or different spheres of a person's health, or they could simply have no effect on protein function. When comparing the number of genes affected per individual of the predicted indels to those of the cold cut-off indels, it is quite clear there is a certain improvement; the decrease in numbers shows that the model is able to "clean" the dataset more efficiently than with cold cut-offs alone, which leads to a decrease in noise.

Conclusion

This study sought to develop a method to better identify high-quality, true positive indels using machine learning algorithms. We also sought to measure the impact of these indels on IQ. While the cognitive effect of the predicted indels is small, the random forest and the intersection of the random forest and logistic regression models show a significant effect on IQ in the UKBB cohort. Furthermore, as evidenced by the average number of genes per individual found for each model, the models were able to select indels that impacted a smaller number of genes, as opposed to no model. Therefore, the use of machine learning models is able to select higher quality indels, and can be considered an extra filtering step, but would require further work in order to determine whether we can select indels that have a higher impact on cognitive ability.

Limitations

The machine learning models were trained using a dataset of 207,520 indels, of which 50% were labelled as true indels (label "1") and 50% were labelled as false indels (label "0"). While this balance is necessary for the training of a model, the reality of the datasets is not as such. In the example of SPARK and SSC, only 0.09% of the dataset was labelled as a true positive indel.

Ideally, the entire dataset of negatives and positives would be used to train the model and methods such as boosting, and bagging can compensate for the unbalance in the positive and negative labels. However, more sophisticated technologies and more computing power are required to attempt this method of training and to improve our models.

References

1. Iossifov, I. *et al.* The contribution of de novo coding mutations to autism spectrum disorder. *Nature* **515**, 216–221 (2014).
2. Krumm, N. *et al.* Excess of rare, inherited truncating mutations in autism. *Nat. Genet.* **47**, 582–588 (2015).
3. Huguet, G. *et al.* Measuring and Estimating the Effect Sizes of Copy Number Variants on General Intelligence in Community-Based Samples. *JAMA Psychiatry* **75**, 447–457 (2018).
4. Montgomery, S. B. *et al.* The origin, evolution, and functional impact of short insertion-deletion variants identified in 179 human genomes. *Genome Res.* **23**, 749–761 (2013).
5. Bennett, E. P. *et al.* INDEL detection, the "Achilles heel" of precise genome editing: a survey of methods for accurate profiling of gene editing induced indels. *Nucleic Acids Res.* **48**, 11958–11981 (2020).
6. Sanders, S. J. *et al.* Progress in Understanding and Treating SCN2A-Mediated Disorders. *Trends Neurosci.* **41**, 442–456 (2018).
7. Bernier, R. *et al.* Disruptive CHD8 mutations define a subtype of autism early in development. *Cell* **158**, 263–276 (2014).
8. Albers, C. A. *et al.* Dindel: accurate indel calls from short-read data. *Genome Res.* **21**, 961–973 (2011).
9. MacArthur, D. G. *et al.* A systematic survey of loss-of-function variants in human protein-coding genes. *Science* **335**, 823–828 (2012).
10. Sanger, F., Nicklen, S. & Coulson, A. R. DNA sequencing with chain-terminating inhibitors. *Proc. Natl. Acad. Sci. U. S. A.* **74**, 5463–5467 (1977).

11. Arteche-López, A. *et al.* Sanger sequencing is no longer always necessary based on a single-center validation of 1109 NGS variants in 825 clinical exomes. *Sci. Rep.* **11**, 5697 (2021).

12. Wang, X. V., Blades, N., Ding, J., Sultana, R. & Parmigiani, G. Estimation of sequencing error rates in short reads. *BMC Bioinformatics* **13**, 185 (2012).

13. NGS vs. Sanger Sequencing. https://www.illumina.com/science/technology/next-generation-sequencing/ngs-vs-sanger-sequencing.html.

14. Jorgenson, J. W. & Lukacs, K. D. Free-zone electrophoresis in glass capillaries. *Clin. Chem.* **27**, 1551–1553 (1981).

15. Smith, L. M. *et al.* Fluorescence detection in automated DNA sequence analysis. *Nature* **321**, 674–679 (1986).

16. Messing, J., Crea, R. & Seeburg, P. H. A system for shotgun DNA sequencing. *Nucleic Acids Res.* **9**, 309–321 (1981).

17. Behjati, S. & Tarpey, P. S. What is next generation sequencing? *Arch. Dis. Child. Educ. Pract. Ed.* **98**, 236–238 (2013).

18. Totomoch-Serra, A., Marquez, M. F. & Cervantes-Barragán, D. E. Sanger sequencing as a first-line approach for molecular diagnosis of Andersen-Tawil syndrome. *F1000Res.* **6**, 1016 (2017).

19. Salk, J. J., Schmitt, M. W. & Loeb, L. A. Enhancing the accuracy of next-generation sequencing for detecting rare and subclonal mutations. *Nat. Rev. Genet.* **19**, 269–285 (2018).

20. Kung, A., Munné, S., Bankowski, B., Coates, A. & Wells, D. Validation of next-generation sequencing for comprehensive chromosome screening of embryos. *Reprod. Biomed. Online* **31**, 760–769 (2015).

21. Slatko, B. E., Gardner, A. F. & Ausubel, F. M. Overview of Next-Generation Sequencing Technologies. *Curr. Protoc. Mol. Biol.* **122**, e59 (2018).

22. van Dijk, E. L., Jaszczyszyn, Y., Naquin, D. & Thermes, C. The Third Revolution in Sequencing Technology. *Trends Genet.* **34**, 666–681 (2018).

23. Eid, J. *et al.* Real-time DNA sequencing from single polymerase molecules. *Science* **323**, 133–138 (2009).

24. Jain, M. *et al.* MinION Analysis and Reference Consortium: Phase 2 data release and analysis of R9.0 chemistry. *F1000Res.* **6**, 760 (2017).

25. Quail, M. A. *et al.* A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers. *BMC Genomics* **13**, 341 (2012).

26. Ng, S. B. *et al.* Targeted capture and massively parallel sequencing of 12 human exomes. *Nature* **461**, 272–276 (2009).

27. Meynert, A. M., Ansari, M., FitzPatrick, D. R. & Taylor, M. S. Variant detection sensitivity and biases in whole genome and exome sequencing. *BMC Bioinformatics* **15**, 247 (2014).

28. Belkadi, A. *et al.* Whole-genome sequencing is more powerful than whole-exome sequencing for detecting exome variants. *Proc. Natl. Acad. Sci. U. S. A.* **112**, 5473–5478 (2015).

29. Barbitoff, Y. A. *et al.* Systematic dissection of biases in whole-exome and whole-genome sequencing reveals major determinants of coding sequence coverage. *Sci. Rep.* **10**, 2057 (2020).

30. Bamshad, M. J. *et al.* Exome sequencing as a tool for Mendelian disease gene discovery. *Nat. Rev. Genet.* **12**, 745–755 (2011).

31. van Dijk, E. L., Auger, H., Jaszczyszyn, Y. & Thermes, C. Ten years of next-generation sequencing technology. *Trends Genet.* **30**, 418–426 (2014).

32. Schwarze, K., Buchanan, J., Taylor, J. C. & Wordsworth, S. Are whole-exome and whole-genome sequencing approaches cost-effective? A systematic review of the literature. *Genet. Med.* **20**, 1122–1130 (2018).

33. Wiltgen, M. Algorithms for Structure Comparison and Analysis: Homology Modelling of Proteins. in *Encyclopedia of Bioinformatics and Computational Biology* (eds. Ranganathan, S., Gribskov, M., Nakai, K. & Schönbach, C.) 38–61 (Academic Press, 2019).

34. EMBL-EBI. [No title]. https://www.ebi.ac.uk/Tools/psa/.

35. Musich, R., Cadle-Davidson, L. & Osier, M. V. Comparison of Short-Read Sequence Aligners Indicates Strengths and Weaknesses for Biologists to Consider. *Front. Plant Sci.* **12**, 657240 (2021).

36. Li, H. & Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25**, 1754–1760 (2009).

37. Danecek, P. *et al.* The variant call format and VCFtools. *Bioinformatics* **27**, 2156–2158 (2011).

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

39. Garrison, E. & Marth, G. Haplotype-based variant detection from short-read sequencing. *arXiv [q-bio.GN]* (2012).

40. *wecall: Fast, accurate and simple to use command line tool for variant detection in NGS data*. (Github).

41. Blog. *Google AI Blog* https://ai.googleblog.com/2020/09/improving-accuracy-of-genomic-analysis.html.

42. Zook, J. M. *et al.* A robust benchmark for detection of germline large deletions and insertions. *Nat. Biotechnol.* **38**, 1347–1355 (2020).

43. Stein, L. Genome annotation: from sequence to biology. *Nat. Rev. Genet.* **2**, 493–503 (2001).

44. Salzberg, S. L. Next-generation genome annotation: we still struggle to get it right. *Genome Biol.* **20**, 92 (2019).

45. de Sá, P. H. C. G. *et al.* Chapter 11 - Next-Generation Sequencing and Data Analysis: Strategies, Tools, Pipelines and Protocols. in *Omics Technologies and Bio-Engineering* (eds. Barh, D. & Azevedo, V.) 191–207 (Academic Press, 2018).

46. Cock, P. J. A., Fields, C. J., Goto, N., Heuer, M. L. & Rice, P. M. The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Res.* **38**, 1767–1771 (2010).

47. Liao, P., Satten, G. A. & Hu, Y.-J. PhredEM: a phred-score-informed genotype-calling approach for next-generation sequencing studies. *Genet. Epidemiol.* **41**, 375–387 (2017).

48. Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078–2079 (2009).

49. gVCF Files. https://support.illumina.com/help/BS_App_TSA_help/Content/Vault/Informatics/Sequencing_Analysis/BS/swSEQ_mBS_gVCF.htm.

50. Li, H. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics* **27**, 2987–2993 (2011).

51. Stoler, N. & Nekrutenko, A. Sequencing error profiles of Illumina sequencing instruments. *NAR Genom Bioinform* **3**, lqab019 (2021).

52. Kelley, D. R. & Salzberg, S. L. Detection and correction of false segmental duplications caused by genome mis-assembly. *Genome Biol.* **11**, R28 (2010).

53. Margulies, M. *et al.* Genome sequencing in microfabricated high-density picolitre reactors. *Nature* **437**, 376–380 (2005).

54. Treangen, T. J. & Salzberg, S. L. Repetitive DNA and next-generation sequencing: computational challenges and solutions. *Nat. Rev. Genet.* **13**, 36–46 (2011).

55. Eichler, E. E. Genetic Variation, Comparative Genomics, and the Diagnosis of Disease. *N. Engl. J. Med.* **381**, 64–74 (2019).

56. Freeman, J. L. *et al.* Copy number variation: new insights in genome diversity. *Genome Res.* **16**, 949–961 (2006).

57. Collins, R. L. *et al.* Defining the diverse spectrum of inversions, complex structural variation, and chromothripsis in the morbid human genome. *Genome Biol.* **18**, 36 (2017).

58. Pranckėnienė, L., Jakaitienė, A., Ambrozaitytė, L., Kavaliauskienė, I. & Kučinskas, V. Insights Into de novo Mutation Variation in Lithuanian Exome. *Front. Genet.* **9**, 315 (2018).

59. Veltman, J. A. & Brunner, H. G. De novo mutations in human genetic disease. *Nat. Rev. Genet.* **13**, 565–575 (2012).

60. Mani, A. Pathogenicity of De Novo Rare Variants: Challenges and Opportunities. *Circulation. Cardiovascular genetics* vol. 10 (2017).

61. 1000 Genomes Project Consortium *et al.* A global reference for human genetic variation. *Nature* **526**, 68–74 (2015).

62. Cooper, G. M. & Shendure, J. Needles in stacks of needles: finding disease-causal variants in a wealth of genomic data. *Nat. Rev. Genet.* **12**, 628–640 (2011).

63. Ng, P. C. *et al.* Genetic variation in an individual human exome. *PLoS Genet.* **4**, e1000160 (2008).

64. Mullaney, J. M., Mills, R. E., Pittard, W. S. & Devine, S. E. Small insertions and deletions (INDELs) in human genomes. *Hum. Mol. Genet.* **19**, R131-6 (2010).

65. Rodriguez-Murillo, L. & Salem, R. M. Insertion/Deletion Polymorphism. in *Encyclopedia of Behavioral Medicine* (eds. Gellman, M. D. & Turner, J. R.) 1076–1076 (Springer New York, 2013).

66. de la Torre-Ubieta, L., Won, H., Stein, J. L. & Geschwind, D. H. Advancing the understanding of autism disease mechanisms through genetics. *Nat. Med.* **22**, 345–361 (2016).

67. Korenke, G. C., Schulte, B., Biskup, S., Neidhardt, J. & Owczarek-Lipska, M. A Novel de novo Frameshift Mutation in the ***BCL11A*** Gene in a Patient with Intellectual Disability Syndrome and Epilepsy. *Mol. Syndromol.* **11**, 135–140 (2020).

68. Loureiro, L. O. *et al.* A recurrent SHANK3 frameshift variant in Autism Spectrum Disorder. *NPJ Genom Med* **6**, 91 (2021).

69. Babbs, C. *et al.* De novo and rare inherited mutations implicate the transcriptional coregulator TCF20/SPBP in autism spectrum disorder. *J. Med. Genet.* **51**, 737–747 (2014).

70. O'Rawe, J. *et al.* Low concordance of multiple variant-calling pipelines: practical implications for exome and genome sequencing. *Genome Med.* **5**, 28 (2013).

71. Fang, H. *et al.* Reducing INDEL calling errors in whole genome and exome sequencing data. *Genome Med.* **6**, 89 (2014).

72. Yang, R., Van Etten, J. L. & Dehm, S. M. Indel detection from DNA and RNA sequencing data with transIndel. *BMC Genomics* **19**, 270 (2018).

73. SPARK Consortium. Electronic address: pfeliciano@simonsfoundation.org & SPARK Consortium. SPARK: A US Cohort of 50,000 Families to Accelerate Autism Research. *Neuron* **97**, 488–493 (2018).

74. Sanders, S. J. *et al.* Insights into Autism Spectrum Disorder Genomic Architecture and Biology from 71 Risk Loci. *Neuron* **87**, 1215–1233 (2015).

75. Sudlow, C. *et al.* UK biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Med.* **12**, e1001779 (2015).

76. HiSeq 2500 System. https://www.illumina.com/systems/sequencing-platforms/hiseq-2500.html.

77. NovaSeq Applications & Methods. https://www.illumina.com/systems/sequencing-platforms/novaseq/applications.html.

78. Zhao, H. *et al.* CrossMap: a versatile tool for coordinate conversion between genome assemblies. *Bioinformatics* **30**, 1006–1007 (2014).

79. Tan, A., Abecasis, G. R. & Kang, H. M. Unified representation of genetic variants. *Bioinformatics* **31**, 2202–2204 (2015).

80. gnomAD. https://gnomad.broadinstitute.org/about.

81. Hall, C. L. *et al.* Frequency of genetic variants associated with arrhythmogenic right ventricular cardiomyopathy in the genome aggregation database. *Eur. J. Hum. Genet.* **26**, 1312–1318 (2018).

82. Amemiya, H. M., Kundaje, A. & Boyle, A. P. The ENCODE Blacklist: Identification of Problematic Regions of the Genome. *Sci. Rep.* **9**, 1–5 (2019).

83. Mandelker, D. *et al.* Navigating highly homologous genes in a molecular diagnostic setting: a resource for clinical next-generation sequencing. *Genet. Med.* **18**, 1282–1289 (2016).

84. Sharp, A. J. *et al.* Segmental duplications and copy-number variation in the human genome. *Am. J. Hum. Genet.* **77**, 78–88 (2005).

85. Tørresen, O. K. *et al.* Tandem repeats lead to sequence assembly errors and impose multi-level challenges for genome and protein databases. *Nucleic Acids Res.* **47**, 10994–11006 (2019).

86. Barra, V. & Fachinetti, D. The dark side of centromeres: types, causes and consequences of structural abnormalities implicating centromeric DNA. *Nat. Commun.* **9**, 4340 (2018).

87. Tutar, Y. Pseudogenes. *Comp. Funct. Genomics* **2012**, 424526 (2012).

88. Sperandei, S. Understanding logistic regression analysis. *Biochem. Med.* **24**, 12–18 (2014).

89. Schonlau, M. & Zou, R. Y. The random forest algorithm for statistical learning. *Stata J.* **20**, 3–29 (2020).

90. Nicodemus, K. K. & Malley, J. D. Predictor correlation impacts machine learning algorithms: implications for genomic studies. *Bioinformatics* **25**, 1884–1890 (2009).

91. Karczewski, K. J. *et al.* The mutational constraint spectrum quantified from variation in 141,456 humans. *Nature* **581**, 434–443 (2020).

92. Hawrylycz, M. *et al.* Canonical genetic signatures of the adult human brain. *Nat. Neurosci.* **18**, 1832–1844 (2015).

93. Arnatkeviciute, A., Fulcher, B. D. & Fornito, A. A practical guide to linking brain-wide gene expression and neuroimaging data. *Neuroimage* **189**, 353–367 (2019).

94. Cervantes, P. E. & Matson, J. L. Comorbid Symptomology in Adults with Autism Spectrum Disorder and Intellectual Disability. *J. Autism Dev. Disord.* **45**, 3961–3970 (2015).