



Technical University of Denmark  
DTU Health Tech

# Master Thesis

by Elisabet Thomsen

**Linked-reads: Improving and  
benchmarking of variant calling pipeline**

## **Linked-reads: Improving and benchmarking of variant calling pipeline**

Master Thesis  
27-06-2020

By  
Elisabet Thomsen

Supervisors: Prof. Gisle Alberg Vestergaard, Technical University of Denmark  
Ólavur Mortensen, Faroe Genome Project

Copyright: Reproduction of this publication in whole or in part must include the customary bibliographic citation, including author attribution, report title,

Published by: DTU, Department of Health Technology, Ørstedes Plads, Building 345C, DK-2800 Kgs. Lyngby, Denmark

[www.healthtech.dtu.dk](http://www.healthtech.dtu.dk)

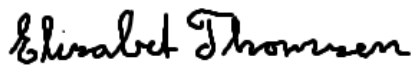
## APPROVAL

---

This thesis has been carried out for 5 months at the Faroe Genome Project (FarGen), Faroe Islands, in collaboration with the Department of Health Technology, the Technical University of Denmark (DTU), in partial fulfilment for the degree Master of Science in Engineering (Bioinformatics and Systems Biology), MSc Eng.

It is assumed that the reader has basic knowledge in the areas of genetics and bioinformatics.

Elisabet Thomsen – s182908



.....  
*Signature*

27-06-2020

.....  
*Date*

## ABSTRACT

---

The most commonly used variant calling pipeline for linked-reads is Long Ranger. This pipeline is however not capable of multi-sample calling, which is believed to be the superior method when calling variants for case-control studies. Therefore, the Faroe Genome project's (FarGen's) staff have developed a pipeline (LinkSeq) to call variants from their data in multi-sample mode. LinkSeq had not been tested and therefore its variant correctness, phase correctness and performance needed to be evaluated before variants called from this pipeline can be used for research and become a part of the Faroese health infrastructure.

There was a great difference in the variant correctness of SNPs and INDELs. The SNPs called in multi-sample mode seemed to be of good quality with a harmonic mean of 98%, while the INDELs were worse with a harmonic mean of 66%.

When comparing LinkSeq's variant correctness with Long Ranger's it was discovered that variant filtering seemed to be the major step which could make a pipeline superior to another. Long Ranger's filtration was e.g. able to increase the precision of INDELs from 62% to 95%. When comparing two filtration methods inside LinkSeq, the filtration of SNPs seemed to be better when applying Variant Quality Score Recalibration (VQSR) compared to hard filtration. In fact, plotting the hard filter parameters revealed that it was nearly impossible to separate false positive (FP) variants from true positive (TP) variants with this method.

LinkSeq's multi-sample called SNPs seem to be of good quality, however the INDELs seem to need improvement.

# CONTENTS

---

Approval .....	3
Abstract.....	4
1 Introduction.....	7
1.1 Linked-reads .....	7
1.2 Exome sequencing .....	7
1.3 How variants can cause disease .....	7
1.4 Key concepts in a variant calling pipeline .....	7
1.4.1 Data pre-processing.....	8
1.4.2 Alignment .....	8
1.4.3 Alignment post-processing .....	8
1.4.4 Variant calling.....	8
1.4.5 Variant calling post-processing.....	8
1.4.6 Phasing.....	9
1.5 INDEL challenges.....	9
1.6 Available pipelines for linked-reads .....	9
1.6.1 Long Ranger.....	9
1.6.2 EMA.....	10
1.6.3 No BQSR .....	10
1.7 FarGen.....	10
1.7.1 LinkSeq.....	10
1.7.2 Barcode contamination .....	10
1.8 Benchmarking pipelines.....	11
1.8.1 Variant correctness.....	11
1.8.2 Phasing.....	12
1.8.3 Performance .....	12
1.9 Aims.....	12
2 Method .....	13
2.1 Server and laptop software.....	14
2.2 Datasets .....	14
2.2.1 NA12878.....	14
2.2.2 FarGen data.....	14
2.2.3 References.....	14
2.3 Test-running LinkSeq .....	15
2.4 Finetuning LinkSeq.....	15

2.4.1	Inserting Read Group Lane information .....	15
2.4.2	Without BQSR .....	15
2.4.3	Study threshold for hard filter parameters .....	15
2.5	Running LinkSeq .....	16
2.5.1	Single-sample calling .....	16
2.5.2	Multi-sample calling .....	16
2.5.3	LinkSeq restrictions .....	16
2.6	Running Long Ranger .....	16
2.7	Benchmarking LinkSeq against Long Ranger .....	17
2.7.1	Variant correctness .....	17
2.7.2	Phase evaluation .....	17
2.7.3	Performance .....	17
2.8	Effect of removing barcodes .....	17
3	Results & Discussion .....	18
3.1	Test running LinkSeq .....	18
3.2	Finetune LinkSeq .....	18
3.2.1	Should BQSR be removed from LinkSeq? .....	18
3.2.2	Added lane information in the Read Groups .....	18
3.2.3	Study Hard Filter parameters .....	19
3.3	Benchmark LinkSeq against Long Ranger .....	19
3.3.1	Variant correctness .....	19
3.3.2	Phase evaluation .....	25
3.3.3	Performance .....	26
3.4	Effect of removing barcodes .....	27
4	Conclusion .....	28
5	Acknowledgement .....	29
6	References .....	30
	Supplements .....	32
	Supplement 1: Coverage .....	33

# 1 INTRODUCTION

---

Variations within the human genome (genetic variants) can be keys for understanding why individuals differ. E.g. why one individual is sick and the other healthy and why one patient responds to a treatment while the other does not. Therefore, genetic variants are widely studied. However, if it is desirable to study these variants the DNA firstly must be sequenced and secondly the sequencing data has to go through a variant calling pipeline to find the variants. There are many sequencing technologies on the market as well as pipelines and software applied in the process of finding the variants. The appliance of different technologies and software may yield different results. When using genetic variants for e.g. diagnosing diseases or discovering which variants are causative to or increase the risk of getting a disease, it is of great importance that the called variants are correct.

## 1.1 LINKED-READS

One of the relatively new technologies within the sequencing world is Linked-reads. In the library preparation long DNA fragments are isolated in GEMs (droplets) and then fragmented while adding barcodes to the reads.<sup>1</sup> The barcodes enable identification from which long DNA molecule the smaller fragments originated.<sup>1</sup> This is a great aid when assembling short reads and when doing haplotype phasing<sup>1</sup>, which are described below.

## 1.2 EXOME SEQUENCING

Exome sequencing means to sequence only a subset of the genome, often the protein coding part.<sup>2</sup> This is done by adding an “exome capture” step in the library preparation. This greatly reduces the cost of sequencing<sup>2</sup>, and e.g. is a good way to study variants responsible for Mendelian diseases, as far most of them are found in the protein coding regions.<sup>2</sup> However, variants sitting in the regulatory part of the genome will be missed.<sup>2</sup> Only sequencing a subset of the genome also means that the data requires much less storage place and when handling the data and analysing it, much more time is saved.<sup>2</sup>

## 1.3 HOW VARIANTS CAN CAUSE DISEASE

Single Nucleotide Polymorphism (SNPs) and small insertion and deletions (INDELs) are the most common variants in humans. In a study of chromosome 22, 82% and 18% of all found variants were SNPs and INDELs, respectively.<sup>3</sup> The number of found variants can differ and depends on the sequenced individual and which variant calling pipeline is used. Of all variants found by Gagliano *et al.* 60844 SNPs and 750 INDELs were found in coding regions.<sup>4</sup> Both SNPs and INDELs can alter human traits and cause human disease. E.g. the genetic disease, cystic fibrosis, is commonly caused by an INDEL in the CFTR gene, which eliminates a single amino acid.<sup>3</sup> The INDELs identified in the coding regions by Mills *et al.* were typically in-frame (multiples of 3 bp) meaning that the reading frame of the proteins remained unchanged.<sup>5</sup> SNPs and INDELs sitting inside the coding region may alter the amino acid sequence and thereby the protein's function. Especially INDELs that alter the reading frame and thereby multiple amino acids, may inhibit the protein from functioning. Variants can as well affect the gene expression by sitting in a gene's promotor or regulatory regions. E.g. Fragile X syndrome in humans is caused by a INDEL sitting in the promotor region of the FMR1 gene.<sup>3</sup>

## 1.4 KEY CONCEPTS IN A VARIANT CALLING PIPELINE

A variant calling pipeline is a process where the input is the raw data from the sequencing machine and the output is a VCF-file containing the genetic variants for each individual.<sup>6</sup> The variants can then be used in diverse studies e.g. a case-control study to identify which variants increase the risk of getting a given disease.<sup>6</sup>

Inside the pipeline there are diverse software and processes. The processes used in this project which are typical for Illumina data are explained shortly below.

#### 1.4.1 Data pre-processing

Starts with **demultiplexing** where the reads coming through the sequencing machine are separated so the reads belonging to each individual go into separate FASTQ-files. To be sequenced the DNA is loaded onto a flow cell where the DNA will be separated onto different **lanes**. During sequencing the yield from each lane can be different. Normally, there is a FASTQ-file per lane per individual, i.e. if there are four lanes on the flow cell, each individual will have four FASTQ-files.

After demultiplexing come diverse **trimming** and **filtering** processes whose purpose is to remove artefacts from the FASTQ-files.

#### 1.4.2 Alignment

In alignment the reads are aligned to a reference genome and a BAM-file is produced which has information on where each read mapped. As regions in the genome can have high similarity a read may map to several locations and the aligner has to choose where to place it. For linked-reads the barcodes are a great aid, because if a read has multiple mappings, the **barcode-aware aligner** will prefer the location that is close to other reads with identical barcodes – as the knowledge exists that they originate from the same long DNA molecule.<sup>7</sup>

#### 1.4.3 Alignment post-processing

Involves **duplicate removal** of artefacts that can appear in PCR processes and base quality score recalibration (**BQSR**) where the quality score (QS) of the bases are recalibrated. The base QS tells how confident the sequencer was to have read that particular base correctly. BQSR is mostly done because some sequencer might report QS higher than they should, and variant calling relies on these QS. During BQSR the program builds a model based on the data and a set of known variants and then the base QS are adjusted according to the model.<sup>8</sup>

#### 1.4.4 Variant calling

In variant calling locations where an individual differs from the reference are located. A VCF-file is produced listing all found variants and which genotype each individual has of a given variant.

There are two ways to call variants: **Single-sample calling** where the variants are called for each individual separately by comparing them to the reference genome. And **Multi-sample calling** where the variants first are found by comparing each individual with the reference genome and then the found variants are compared between the samples. The latter method is preferable in e.g. case-control studies because if a variant is common for multiple individuals then the caller is more certain that the variant is correct. Thereby the sensitivity for low-frequency variants is greater and more False Positive (FP) variants are filtered out.<sup>9</sup>

#### 1.4.5 Variant calling post-processing

Involves soft filtering and optionally hard filtering. In **soft filtering** variant quality score recalibration (**VQSR**) is applied. Information on known variants from e.g. HapMap and 1000 genomes is used to train a gaussian mixture model which can separate the true positive (TP) variants from the false positive



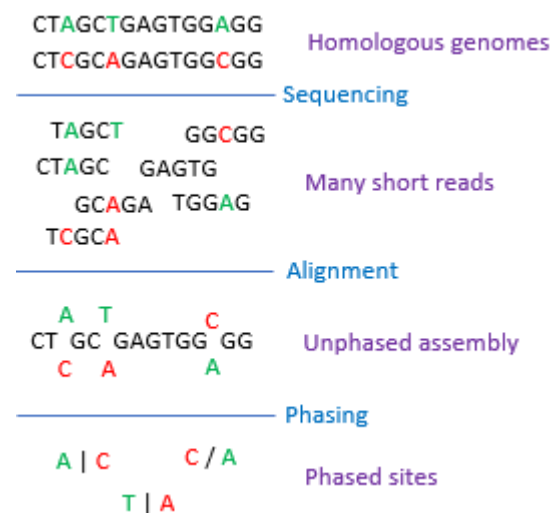
(FP) variants. In **hard filtering** cut-off thresholds are set “manually” for diverse parameters e.g. quality by depth (QD). The first method is considered to be clearly superior to the latter. However, VQSR requires e.g. that the variants are known in the organism and that there are at least 30 individuals of the same ethnicity to be able to train a proper model.<sup>10</sup>

Variant filtration is considered to be an essential step in the pipeline to be able to yield variants of proper quality.<sup>11</sup>

#### 1.4.6 Phasing

In phasing the haplotypes of each heterozygous variant is assembled into **haploblocks** (see fig. 1).<sup>12</sup> The information on the phase (haplotype) of each variant is added to the VCF-file.

The phase of variants can be essential for e.g. two deleterious mutations that occur in the same gene. If the mutations are on the same chromosome then one copy of the gene is functional, however if they in contrast are sitting on separate chromosomes, there will be a total loss of function for the gene.<sup>13</sup>



**Figure 1:** When the short reads are assembled after sequencing there will be some heterozygous sites where there are two possible alleles. To know which alleles belong together in a haplotype, the heterozygous sites are phased. The phaser will however not be able to phase all sites (the unphased sites have a “/” while the phased have a “|”).

## 1.5 INDEL CHALLENGES

INDELs are much more difficult to call correctly compared to SNPs. The concordance of INDELs has been shown to be as low as 26.8% among various callers.<sup>14</sup> Fang *et al.* analysed this matter and found that improper coverage of INDELs causes calling errors. To call INDELs with a sensitivity of 95% a coverage of 60X is needed for PCR-free whole genome sequencing (WGS) data.<sup>15</sup> Whole exome sequencing (WES) was more likely to have lower sensitivity, firstly because it failed to capture 16% of candidate exons, and secondly because the coverage distributions were skewed in the WES data, with some regions poorly covered and other regions over saturated, while the WGS data had a more uniform distribution.<sup>15</sup> Multiple signatures (more than one INDEL called) in the same genomic region gave rise to INDELs errors as well. The major source for multiple signatures was found to be poly-A/T INDELs, which are enriched in WES. PCR amplifications also induced error-prone INDELs – especially poly-A/T INDELs.<sup>15</sup> Large INDELs were also missed in the WES data. Presumably because the larger INDELs could disrupt the base pairing which is needed for the exome capture.<sup>15</sup> Fang *et al.* suggest to use two metrics: the coverage of the alternative allele and the k-mer Chi-Square score of an INDEL to distinguish problematic INDEL calls from likely true-positives.<sup>15</sup>

## 1.6 AVAILABLE PIPELINES FOR LINKED-READS

As linked-reads is a relatively new technology the available pipelines are not as diverse as for other technologies. Only two pipelines are published.

### 1.6.1 Long Ranger

Long Ranger is the pipeline made by 10X genomics who are the same who have produced the linked-reads technology. The limitations of this pipeline are, that it is only capable of single-sample calling.

The user is as well very restricted to adjust the processes inside the pipeline. Long Ranger always calls all types of variants, so the user can not choose to restrict the calling to e.g. SNPs and INDELs to save time. The user can neither choose which alignment post-processes to include and how to filter the variants. The user can however choose to use Freebayes or GATK for the variants calling process.<sup>16</sup>

### 1.6.2 EMA

The authors of EMA<sup>7</sup>, which is the alternative barcode-aware aligner, offer an alternative pipeline to Long Ranger. Besides that this aligner is open source the user can easily use EMA together with the desired processes pre and post to alignment. In the paper where they compare EMA with Long Ranger, they are using GATK to call variants and state that their pipeline both has better performance and better results.<sup>7</sup>

#### 1.6.2.1 RG issue

There is a small issue when running EMA together with GATK tools. GATK best practise pipeline expect the input to be distributed between read groups (RG)<sup>17</sup>. There should be a RG per lane per sample. Normally, there is a FASTQ-file per lane, and these can be aligned separately (with e.g. BWA) and a RG can be assigned per lane before joining the BAM-files to individual level. However, to take full advantage of the barcodes, all lanes are aligned together, so the lane information cannot be added while aligning. In the paper, EMA authors seem to use the GATK tools without the RG lane information, but it can be questioned if this is correct.

### 1.6.3 No BQSR

EMA authors seem also to not include the BQSR step.<sup>7</sup> They do not explain why BQSR is not applied, however, Tian *et al.* state that BQSR may decrease the sensitivity especially in highly diverse regions.<sup>18</sup> As BQSR is based on known variants the process will of course try to “force” the data to be similar to the known variants.

## 1.7 FARGEN

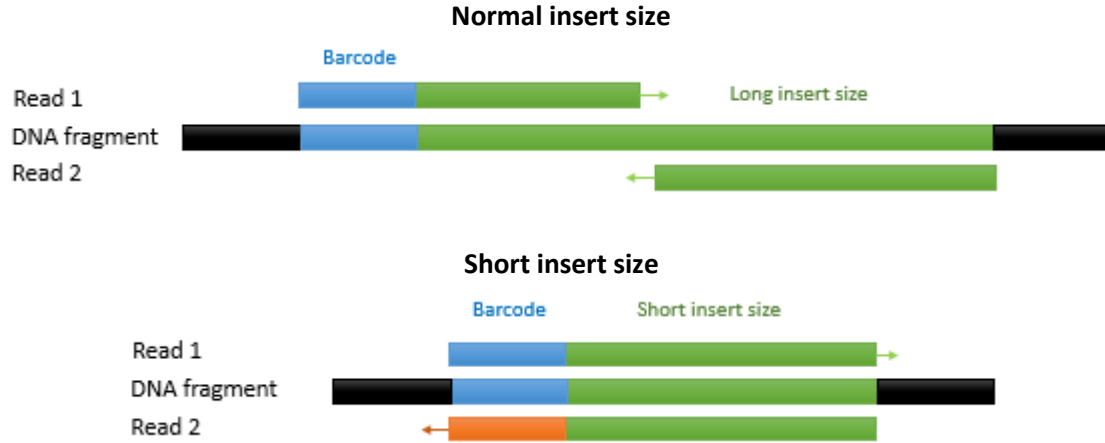
This project is done in collaboration with The Faroe Genome Project (FarGen - [www.fargen.fo](http://www.fargen.fo)). FarGen’s aim is to sequence the genomes of all Faroese people. Besides using the data for research, the data will be part of the Faroes health-infrastructure and can be an aid in diagnosing, treating and personalizing medicine for patients. As a start FarGen has started with sequencing 1500 linked-read exomes and the data from these exomes will have to go through a variant calling pipeline.

### 1.7.1 LinkSeq

FarGen’s staff are building a pipeline based on nextflow where GATK’s best practise pipeline is used together with EMA for alignment and HapCut2<sup>12</sup> for phasing (<https://github.com/olavurmortensen/linkseq>). As the pipeline is new and under development it is important that it is tested in various ways, to make sure that the processes are working as they should.

### 1.7.2 Barcode contamination

In a special course made previous to this thesis the author started with quality control of FarGen’s data and pipeline. Among other things it was discovered that the insert size was rather short, which lead to barcode contamination in R2 (see figure 2). As no tool was available for removing the barcodes, a tool was developed which now is incorporated in LinkSeq’s data pre-processing (<https://github.com/olavurmortensen/linkseq-demux>).



**Figure 2:** Are short insert size can in Linked-reads lead to barcode contamination in Read 2 (lower) compared to if the insert size is of a “normal” longer length (upper). The black areas represent the adapters that are attached to the DNA fragment.

It is not yet tested which effect the barcode contamination has on the called variants, but one could imagine that it would have the same effects as adapter contamination. Some state that adapter contamination can increase the number of false positive variants if not removed.<sup>19–21</sup> However, the aligner BWA<sup>22</sup>, which is incorporated into both Long Ranger’s aligner Lariat<sup>23</sup> and EMA<sup>7</sup>, is capable of applying clipping, which means that it can remove some bases of the ends of the reads, to get a better match.<sup>24</sup> So it might be able to align the reads properly, even if there exists contamination within them.

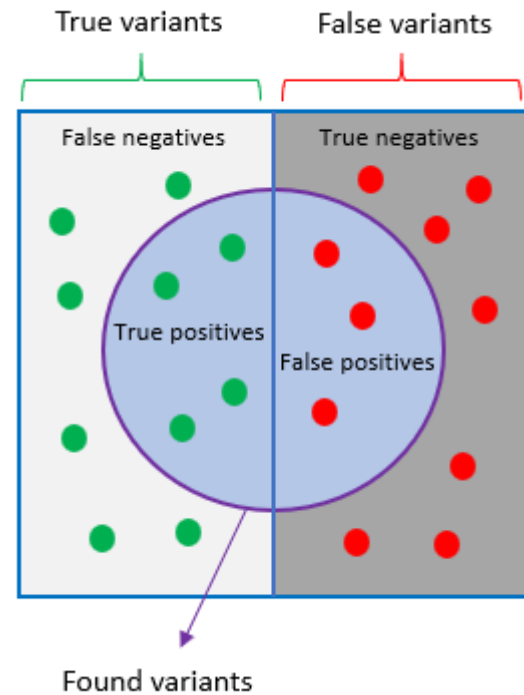
## 1.8 BENCHMARKING PIPELINES

When developing a variant calling pipeline, various things have to be in order. First the variants called have to be correct. Second the phasing of the variants must be correct. Third and last the performance of the pipeline must be good – especially because the data will be part for the health infrastructure and a fast answer can be essential to giving the patient the correct treatment.

To know if the output of the pipeline is correct a truth VCF-file containing the correct answers (variants) is needed. Luckily, NIST Genome in a bottle (GiaB) have produced a high confidence VCF-file which has been used for benchmarking in many studies.<sup>7,25</sup>

### 1.8.1 Variant correctness

When looking at the variant correctness, it is common to look at the sensitivity, precision and harmonic mean.<sup>26</sup> **Sensitivity** tells how many of the true variants were found ( $\frac{TP}{TP + FN}$ ). **Precision** tells how many of the found variants are true ( $\frac{TP}{TP + FP}$ ). And the **harmonic mean** is a “mean” between sensitivity and precision ( $2 * \frac{sensitivity * precision}{sensitivity + precision}$ ).<sup>26</sup> If the sensitivity is too low, there are too many False



**Figure 3:** When calling variants both true positive (TP) and false positive (FP) variants will be called. The aim is always to get as low FP variants without losing too many TP (and thereby getting more False negatives (FN)).

Negative (FN) variants and if the precision is low there are too many False Positives (FP) variants (see fig. 3). Both of these are important in e.g. diagnoses, because if there are FN variants, some patients having risk variants many be told not to have them, and if there are FP variants then patients not having risk variants many be told to have them.

### 1.8.2 Phasing

There exists a python program (`error_rates.py`) to test if the variants are correctly phased (uses a truth VCF-file). The output is among other things switch error and mismatch error counts. **Switch error** means that at a heterozygous site the phase has switched where it should not.<sup>12</sup> This will cause the following variants to be wrongly phased until a site is met that switched the variants back to the correct phase. This is also called a long switch. **Mismatch error** is two switch errors on adjacent sites, so only a single variants phase will be switched<sup>12</sup> – also called short switch.

There also exists software which give phasing statistics which tell e.g. how long the phase blocks are, how many variants are phased etc.

### 1.8.3 Performance

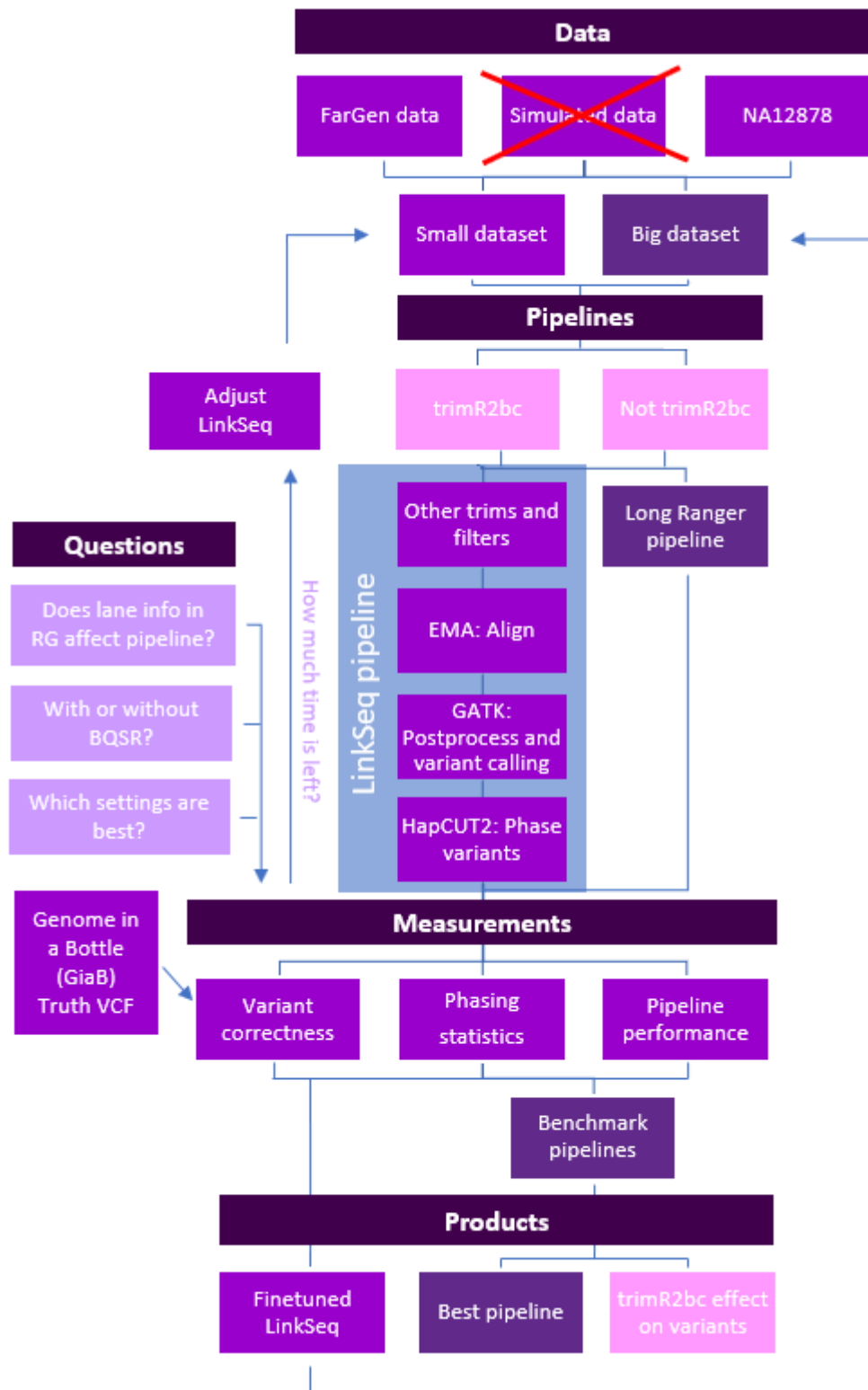
Performance is often measured in **duration**, which is the time it takes for the pipeline to run, **CPU hours**, which is how much time is used on each core<sup>25</sup> and **memory usage**, which is how much memory the process used. Together these three give a good measurement of how long it takes to run the pipeline and how many computational resources it requires.

## 1.9 AIMS

As LinkSeq has not been used yet it is important to **test run the scripts** and correct them, if errors occur. Secondly, it should be investigated if inserting the **RG lane information** affects the pipeline and if **BQSR** should be included or not. The effect the **barcode contamination** has on the variants should also be investigated. And lastly the performance and variants produced by LinkSeq should be investigated and **benchmarked against Long Ranger**.

## 2 METHOD

Fig 4. shows a graphical view of the workflow.



**Figure 4:** Graphical view of the workflow used in this project. Be aware that inside LinkSeq, variants were both called in single-sample mode and multi-sample mode. The former variants were filtered with hard filter and the latter variants were filtered with VQSR. Both methods were benchmarked.

## 2.1 SERVER AND LAPTOP SOFTWARE

This project was a collaboration between The Faroe Genome Project (FarGen - <https://www.fargen.fo/>) and Denmark's Technical University (<https://www.dtu.dk/>). The work was done on FarGen's server using OpenVPN GUI v11.13.0.0 (<https://openvpn.net/>), Windows command prompt and Rancher Command Line Interface (CLI) v2.2.0 (<https://rancher.com/>) on the author's laptop.

Occasionally R studio v1.2.5001<sup>27</sup> was used to create R scripts ([www.r-project.org](http://www.r-project.org)) to run on the server.

Conda v4.8.3 (<https://anaconda.org/>) was used to install most of the software in the server and manage the environments.

## 2.2 DATASETS

### 2.2.1 NA12878

Linked-reads FASTQ-files of the universal sample NA12878, where high confidence variant calls exist, were downloaded from 10X's webpage ([https://support.10xgenomics.com/genome-exome/datasets/2.1.4/NA12878\\_WES\\_v2](https://support.10xgenomics.com/genome-exome/datasets/2.1.4/NA12878_WES_v2)) and used to finetune LinkSeq, benchmark LinkSeq against Long Ranger and benchmark barcode trimming against no barcode trimming.

The VCF-file containing the high confidence variant calls and a BED-file containing confident regions were downloaded from Genome in a Bottle (GiaB) [ftp://ftp-trace.ncbi.nlm.nih.gov/giab/ftp/release/NA12878\\_HG001/latest/GRCh38/](ftp://ftp-trace.ncbi.nlm.nih.gov/giab/ftp/release/NA12878_HG001/latest/GRCh38/).

### 2.2.2 FarGen data

As it was not possible to find a large quantity dataset with truth VCF-files, three samples from the FarGen data were used to benchmark the multi-sample mode. The laboratory part and sequencing of this data was done in same manner as described in Mortensen *et al.* 2019.

A VCF-file which existed from a previous study containing variants from the same individuals was used as the true variants. This file had been produced from exome data (not linked-reads) by aligning with BWA<sup>24</sup> and calling with FreeBayes (<https://github.com/bolosky/freebayes>).

### 2.2.3 References

The reference genomes GRCh38 and b37 and known SNPs and INDELs were downloaded from the GATK bundle (<https://gatk.broadinstitute.org/hc/en-us/articles/360035890811-Resource-bundle>). The targets BED-files were downloaded from Agilent's webpage (<https://earray.chem.agilent.com/suredesign/>) and modified by adding a 0 in the 5<sup>th</sup> column and a '.' in the 6<sup>th</sup> column. The hg19 (for the b37) BED-file was modified to contain only the chromosomal numbers instead of e.g. "chr1". The modifications were done with awk command (see [https://github.com/ElisabetThomsen/MSc\\_thesis](https://github.com/ElisabetThomsen/MSc_thesis)).

The single sample pipeline with NA12878 was aligned and benchmarked with the GRCh38 references. The multi sample pipeline with the FarGen data was aligned and benchmarked on b37 as the existing VCF-files from the previous study was based on this reference.

## 2.3 TEST-RUNNING LINKSEQ

As LinkSeq is a pipeline under development, it needed to test-run first.

Linkseq's scripts were cloned from the git repository ( <https://github.com/olavurmortensen/linkseq> ). The main.nf script was test-run first with a subset of the NA12878 FASTQ-files (1000 reads per file) and when all processes succeeded the original size of the NA12878 FASTQ-files were run through it.

The joint\_genotyping.nf script was test-run with three of the FarGen samples.

When processes in the scripts failed and showed error messages the author fixed them, or they were reported to FarGen's staff. When fixing errors, a branch was made of the master branch and the changes made in the script were committed and pushed and a pull request was made to merge the branch with the master branch.

In the start of the project, LinkSeq only existed in multi-sample calling form, so the joint\_genotyping.nf script was modified by the author to genotype a single sample instead of multiple, so it was possible to benchmark the NA12878 sample. As VQSR needs at least 30 exomes to be accurate, VQSR (soft filter) was replaced by hard filter as recommended by the GATK team.<sup>11</sup> These processes were later added to the main.nf script so it besides preparing gVCF-files for the joint-genotyping also does single-sample calling.

## 2.4 FINETUNING LINKSEQ

When the main.nf script was running without error messages some changes were made to the pipeline, to see if it could improve the pipeline. Variant correctness (run as in Section 2.7.1) was used to assess the impact of the changes.

### 2.4.1 Inserting Read Group Lane information

A program (addRG.py), which adds the RG information to a SAM-file, was made in Python v3.7.3 ( [www.python.org](http://www.python.org) ). The program first parses through the file and finds the line in the header containing the RG information. Here it replaces the line with the new RGs containing the lane information given on the command line. Then for each read in the SAM-file, it finds the lane in the readname (first tab-separated field) and adds it to the RG tag. The program can be seen here: [https://github.com/ElisabetThomsen/MSc\\_thesis](https://github.com/ElisabetThomsen/MSc_thesis)

The BAM-file from the process sort\_bam in the already run LinkSeq was found in the work folder and modified with addRG.py. Then all previous processes were commented out and a channel which found the RG-modified BAM-file from the given path was added to LinkSeq, so the variants could be called from the modified BAM-file.

### 2.4.2 Without BQSR

To get VCF-files where the BQSR process was not included the BQSR process and all previous processes were commented out in the main.nf and a channel was made, which found the BAM-file from the previous process (mark duplicates) in the already run LinkSeq work folder.

### 2.4.3 Study threshold for hard filter parameters

Plots for studying the hard filter parameters' thresholds were made with inspiration from <https://gatk.broadinstitute.org/hc/en-us/articles/360035890471> , but instead of splitting the variants into passed and not passed, they were split into TP and FP.

To make these plots the data had to be prepared first. The variants in the VCF-file were first annotated as True Positives (TP) or False Positives (FP) with GATK's<sup>28</sup> v4.1.0.0 VariantAnnotator with GiaB's



VCF-file as the truth file. Then GATK's VariantsToTable was used to make a new tab-separated file from the VCF, as it is easier to handle in the following steps.

A python program was made to write the TP and FP into separate files, so it was easier to import into R. It separated them by looking at the "Truth" column that VariantAnnotator and VariantsToTable added.

Last a R script was made which used the ggplot2 package to plot the TP and FP distributions of each hard filter parameter.

The scripts used can be seen here: [https://github.com/ElisabetThomsen/MSc\\_thesis](https://github.com/ElisabetThomsen/MSc_thesis)

## 2.5 RUNNING LINKSEQ

### 2.5.1 Single-sample calling

The original pipeline of LinkSeq (with BQSR and without RG lane information) was used to benchmark against Long Ranger

The FASTQ-files of NA12878 were first run through the modified version of LinkSeq-Demux (modified to take FASTQ input instead of BCL) for trimming raw reads and then run through main.nf to produce the single-sample VCF-file.

### 2.5.2 Multi-sample calling

#### 2.5.2.1 FarGen samples

As the joint-genotyping script at the time of writing is still not working for too many samples, only three FarGen samples were multi-sample called together, for the multi-sample calling.

The author received a VCF-file from a previous study (from a different exome sequencing method) from FarGen's staff. It contained variant calls of the same individuals. These individuals were filtered out of the VCF file by using GATK's SplitVcfs so they could be used for benchmarking.

#### 2.5.2.2 NA12878

The gVCF-file of NA12878 from the main.nf script run was also thrown together with the three gVCFs from the FarGen data, just to see if the variant filtration got better, when applying VQSR instead of hard filter. Have in mind that this is far from optimal, because they recommend minimum 30 exomes of the same ethnicity.

### 2.5.3 LinkSeq restrictions

Through the nextflow.config file the processes were restricted to 4 cores and 20 GB, while the executor was restricted to 40 cores and 200 GB (how much parallelized processes may use together).

## 2.6 RUNNING LONG RANGER

To be able to benchmark the performance a nextflow script (lr\_targeted.nf) was modified to run Long Ranger. The nextflow.config file had the same core and memory restrictions as LinkSeq had.

Inside the nextflow script Long Ranger v2.2.2 targeted was run with standard settings.

Initially it was the plan to run Long Ranger with GATK as caller, but it seemed to get stuck, and because no error messages were displayed, it was not easy to try to fix a potential error. So after multiple tries and letting it run for a week with access to all 96 cores and free memory access, FreeBayes was used as caller instead.



## 2.7 BENCHMARKING LINKSEQ AGAINST LONG RANGER

Both single-sample calling mode and multi-sample calling mode of LinkSeq were benchmarked against Long Ranger (which can only do single-sample calling).

### 2.7.1 Variant correctness

Hap.py v0.3.12 ( <https://github.com/Illumina/hap.py> ) was run with standard setting, targeted (-T) option and the GiaB file as truth VFC-file for the NA12878 sample. Beside using the estimates of sensitivity, precision and harmonic mean the roc files produced by hap.py were used as input to the R script producing the ROC curves. The R script can be seen here [https://github.com/ElisabetThomsen/MSc\\_thesis](https://github.com/ElisabetThomsen/MSc_thesis) . The plots were made with inspiration from hap.py's manual ( <https://github.com/Illumina/hap.py/blob/master/doc/microbench.md> ).

It was the intention to use hap.py to benchmark the FarGen VCFs as well, however, presumably by some parsing errors, in the output the truth VCFs were listed as having no variants. Therefore, it could not figure out to calculate the sensitivity and precision either. As the time was sparse, GATK's Concordance was used instead.

Intersections between the variants found by the pipelines were found by using BEDTools<sup>29</sup> and the venn diagrams were made in R with the VennDiagram package.

### 2.7.2 Phase evaluation

Originally error\_rates.py ( <https://github.com/arshajii/ema-paper-data/tree/master/data/phase> ) was intended to use for benchmarking the phase correctness. However, when it was realized that Long Ranger did not produce the block file needed, Whatsap v0.18<sup>30</sup> was used instead. The intention was to use N50, but after several tries for getting this out of Whatsapp, the average, median and longest block were used instead.

### 2.7.3 Performance

The nextflow scripts were run with the option -with-report and -with-timeline to get the duration, CPU hours and Memory peak used.

## 2.8 EFFECT OF REMOVING BARCODES

trimR2bc.py ( <https://github.com/ElisabetThomsen/trimR2bc> ) was modified in a way, so it could filter the FASTQ-files and write only the reads where barcode contamination was found. It wrote two files: A trimmed file where the barcodes in R2 were removed and an untrimmed file where the barcodes were not removed.

The LinkSeq-Demux main.nf script ( <https://github.com/olavurmortensen/linkseq-demux> ) which includes trimming of raw reads, was then modified to take FASTQ-files as input instead of BCL-files and the bctrim process was commented out. Both the trimmed and untrimmed file were then run through this modified main.nf script to get the other trimmings.

As trimming the barcode might in some cases make the read so short that it would be filtered out by minimum length filter, the FASTQ-files were synchronised afterwards with BBTool's Repair v37.62<sup>31</sup> and only the reads present in both the bctrimmed and bctrimmed FASTQ-files were kept.

## 3 RESULTS & DISCUSSION

---

### 3.1 TEST RUNNING LINKSEQ

The main.nf script, which also does the single-sample calling, ran without errors, when the correct software dependencies had been installed. However, when benchmarking the VCF-file, it was noticed that the INDELs had disappeared and only SNPs were present in the VCF-file. This was because they had to be filtered separately and a process joining them afterwards was missing, so this was added to the main.nf script as well as the joint\_genotyping.nf script.

The joint\_genotyping.nf script ran without errors when test-running with three samples. However, when trying to run the pipeline with a larger sample size (80 exomes), it by some reason is stuck in the “consolidate\_gvcf” process. This is still a problem at the time of writing.

### 3.2 FINETUNE LINKSEQ

When benchmarking the slightly modified pipelines against the original LinkSeq the sensitivity, precision and harmonic mean did not differ more than 0.002 from the original pipeline. Therefore, they seemed not to have any significant effect on the NA12878 dataset.

#### 3.2.1 Should BQSR be removed from LinkSeq?

The decrease in sensitivity that Tian *et al.* state may occur when applying BQSR was not observed in neither SNPs nor INDELs or it was so low (0.001 difference) that the impact will be very small. One reason behind this could be that it was mainly in regions with high divergence and low coverage that Tian *et al.* found the decrease in sensitivity. As a confident region BED-file was used when estimating the sensitivity, the high divergent areas might have been filtered out. Their study is as well only on simulated data from chromosome 6p21.3, which is highly divergent<sup>18</sup> and our study is on real WES data.

For the NA12878 data it took ~1,5 hours to prepare the BQSR table and apply the recalibration. Furthermore, it takes ~1 hour to generate the BQSR report, however this is a parallel process running simultaneously with the calling (see timelines [https://github.com/ElisabetThomsen/MSc\\_thesis](https://github.com/ElisabetThomsen/MSc_thesis) ).

It could be taken under consideration if the BQSR process should be taken out of the pipeline to increase the performance. However, when studying the BQSR report ( [https://github.com/ElisabetThomsen/MSc\\_thesis/blob/master/results/AnalyzeCovariates\\_NA12878.pdf](https://github.com/ElisabetThomsen/MSc_thesis/blob/master/results/AnalyzeCovariates_NA12878.pdf) ), it is noticed that the data was of good quality before recalibration, and this could explain why BQSR seems to have so little impact. It would be best to benchmarking with a dataset where base QS is greatly changed after recalibration and study the sensitivity in that dataset.

As the impact of BQSR on the variants still is uncertain for a dataset with base quality scores that need recalibration, it cannot be advised to remove it yet. One should have in mind, that if the BQSR is removed there is a risk that the base QS is incorrect (most commonly higher than what it should be) and this could give false confidence to call variants that should not be called.

#### 3.2.2 Added lane information in the Read Groups

The addition of lane information in the RG seemed to not have an impact on the variant correctness. So even if the GATK team states that the software are RG aware<sup>17</sup>, the lane information seems not to be so vital. The made program (addRG.py) took only 6 minutes to run, however, it currently needs a SAM-file as input and the conversion of BAM to SAM and SAM back to BAM took ~30 minutes. As it

seemed to have little impact on sensitivity and precision, in total slows the pipeline by 36 minutes and EMA's authors did not add the read group information, it may not seem worth to insert this in LinkSeq.

### 3.2.3 Study Hard Filter parameters

Plotting all hard-filter parameters and the recommended threshold (see [https://github.com/ElisabetThomsen/MSc\\_thesis/tree/master/results](https://github.com/ElisabetThomsen/MSc_thesis/tree/master/results)) showed that the distributions of TP and FP variants were overlapping. The only parameter where the distributions differed slightly was the Quality by Depth (QD) parameter (see fig. 5).

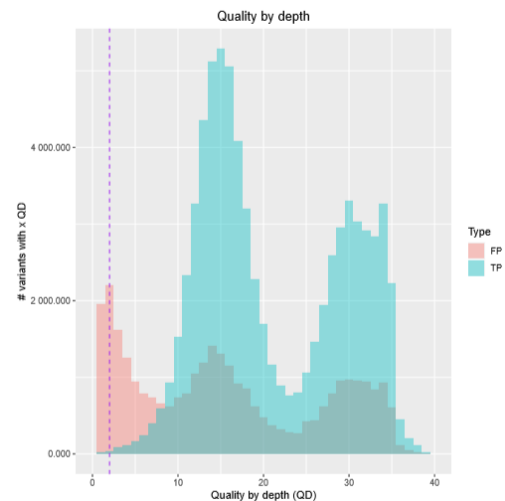
Have in mind that these plots show all called SNPs – not only the ones that are in confident regions, that hap.py uses for benchmarking. Therefore, there are more variants in these plots.

The plots support GATK team's statement<sup>10</sup>, that setting the hard filter parameters is not so strait forward and by looking at the annotation dimensions individually it is not so easy to separate TP variants from FP variants, compared with if it was possible to look at how the variants cluster along different dimensions as VQSR does.

For the NA12878 sample, hard filter does not seem to be a good solution. The most fitting approach would be to run this on multiple diverse samples, but this was not possible due to lacking 10X exome data from other samples with truth VCF-files.

There exist other solutions for single sample filtering such as a CNN model, which GATK is developing at the time of writing.<sup>11</sup> There was made an attempt to run this software on the server, but this was unsuccessful, due to the software dependencies could not be installed. Furthermore, it will be a challenge to make the environment that this software needs compatible with the environment used by LinkSeq at the moment.

There exist other no-GATK software such as GARFIELD-NGS<sup>32</sup> and VarBin<sup>33</sup>, but these were not tried out, due to time constrains.



**Fig 5:** Histogram showing how the Quality by Depth (QD) values are distributed for true positive (TP, blue) and false positive (FP, red) SNPs. QD is one of the parameters used in hard filter.

## 3.3 BENCHMARK LINKSEQ AGAINST LONG RANGER

LinkSeq and Long Ranger were compared with regards to variants correctness, phase evaluation and performance.

### 3.3.1 Variant correctness

The variant correctness was studied both for single-sample calling and multi-sample calling mode for LinkSeq and compared to Long Rangers correctness, which always is in single-sample calling mode.

#### 3.3.1.1 Single-sample calling

As the result for SNPs and INDELs differed, they are compared separately.

### 3.3.1.1.1 SNPs – NA12878

LinkSeq and Long Ranger found 104830 and 118532 SNPs, respectively, in the targeted regions. These numbers are somewhat higher than the 60844 SNPs that Gagliano *et al.* found in coding regions.<sup>4</sup> But have in mind that this number is for unfiltered SNPs. The truth VCF had 73881 SNPs in the targeted regions (not filtered by confident regions).

When looking at the unfiltered SNPs LinkSeq is having a precision and harmonic mean that are 6% and 3% greater than Long Ranger's, respectively (fig. 5). Long Ranger has a greater sensitivity, but it is only 0.5% higher than LinkSeq's. Have in mind that the sensitivity, precision and harmonic mean (fig. 5) are calculated only from variants in the confident regions.

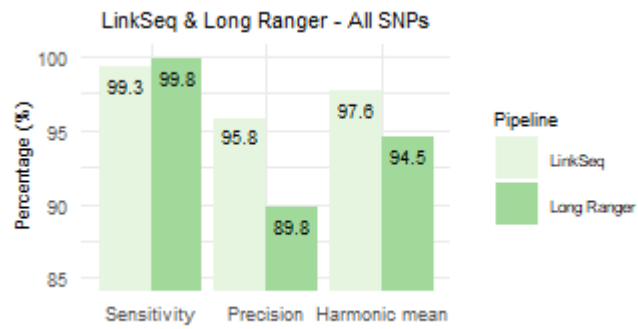
After filtration Long Ranger is however outcompeting LinkSeq and now Long Ranger's sensitivity, precision and harmonic mean are 4%, 1% and 3% greater than LinkSeq's, respectively (fig. 6).

When applying the hard filter LinkSeq's precision is increasing with 2%, but the sensitivity and harmonic mean lowered with 4% and 1%, respectively.

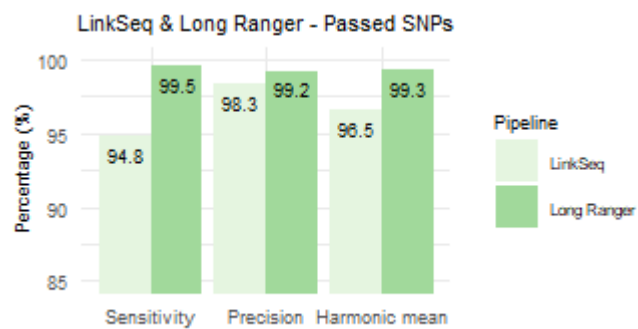
All this indicates that Long Ranger filtering process is superior, while LinkSeq's filtration could be improved for the single-sample called SNPs. If taking the hard filter threshold plots ( [https://github.com/ElisabetThomsen/MSc\\_thesis/tree/master/results](https://github.com/ElisabetThomsen/MSc_thesis/tree/master/results) ) under consideration as well as GATK's statement, that hard filter is not an optimal procedure for filtering variants<sup>10</sup>, these results are not of surprising nature.

However, it should be stated that even if there is difference between LinkSeq's and Long Ranger's correctness, it is only 1-4% as stated above, which not is a great difference.

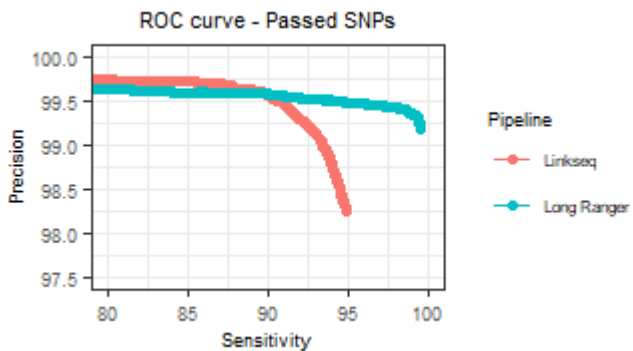
The ROC-curve (fig. 7), which shows the relationship between the sensitivity and precision, is in agreement with the results above and showed for LinkSeq that when the sensitivity is increased to more than 0.85 the precision starts lowering. Long Ranger's curve lowers more slowly and the rapid decrease in precision does not start before reaching a sensitivity of ~ 0.99.



**Fig 5:** Sensitivity, precision, harmonic mean for NA12878, unfiltered SNPs. Be aware of axis adjustment.

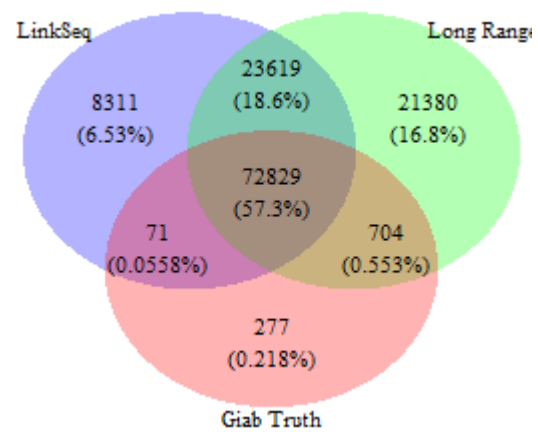


**Fig 6:** Sensitivity, precision, harmonic mean for NA12878, filtered SNPs. Be aware of axis adjustment.



**Fig 7:** ROC curve for LinkSeq and Long Ranger, NA12878, filtered SNPs.

To study now the variants found by the pipelines were overlapping a venn diagram was made (fig. 8). Of the union of all SNPs in the truth VCF, LinkSeq and Long Ranger, only 0.2% are truths SNPs that neither pipelines found (fig. 8). Their efficiency in finding the truth SNPs is in agreement with the estimated sensitivity (fig. 5). Despite they are good at finding truth SNPs, they are also finding FP SNPs. Long Ranger is finding more FP SNPs (35%), which not are in the truth VCF, compared with LinkSeq (25%). This fits with the precision being lower for Long Ranger before filtration (fig. 5). 19% of all SNPs are found both by Long Ranger and LinkSeq but are not in the truth VCF. Finally, 57% of all SNPs are common for both pipelines and the truth VCF. So 76% of all SNPs are common for both pipelines.



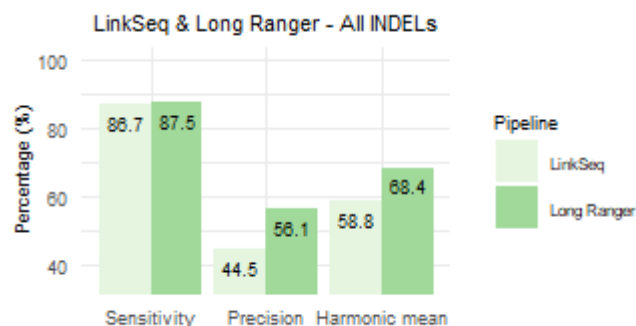
**Fig 8:** Venn diagram showing how the SNPs called by LinkSeq and Long Ranger overlap with Giab's truth VCF. Only SNPs in target regions. Unfiltered. NA12878.

### 3.3.1.1.2 INDELs – NA12878

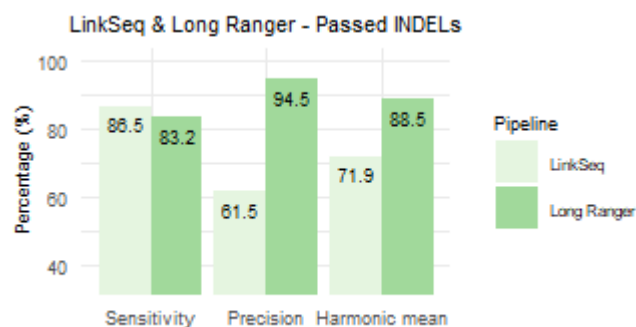
LinkSeq and Long Ranger found 22902 and 16481 INDELs in the targeted regions. This is clearly much higher than Gagliano *et al.* found in coding regions. This awakens a suspicious thought that there probably is a great number of FP INDEL calls. However, the GiaB VCF-file has 10040 INDELs in the targeted regions. So it seems like the GiaB Consortium have been able to identify a greater number of INDELs, when applying many sequencing tools and calling pipelines. However, be aware that these are the numbers before restricting to the confident regions.

For the INDELs, both pipelines start out with relative low sensitivity (87-88%) and precision (45-56%) (see fig. 9) when comparing them to the SNP's correctness (fig. 5). The reason behind the INDEL sensitivity begin low, might not be the pipelines' fault, but rather because some of the exome targeted captures have been missed. In fact, the missing 12.5-13.3% of the INDELs lies closely to the 16% INDELs that Fang *et al.* reported missing.<sup>15</sup> When both LinkSeq and Long Ranger share this "low" sensitivity it is a greater reason to believe that the INDELs are truly missing. When looking at the Venn diagram it can also be seen that only 3% are true INDELs missed by both pipelines (fig. 12). Looking at the coverage across the targeted regions (Supplement 1) also indicates that some regions might have lower coverage.

After filtration LinkSeq and Long Ranger's precision increases with 17% and 39%, respectively, and their harmonic mean increases with 13% and 21%, respectively (fig. 10).



**Fig 9:** Sensitivity, precision, harmonic mean for NA12878, unfiltered INDELs. Be aware of axis adjustment.



**Fig 10:** Sensitivity, precision, harmonic mean for NA12878, filtered INDELs. Be aware of axis adjustment.

Similar to the SNPs result Long Ranger is having a good filtering process and is able to increase the precision to 95% while LinkSeq's is still down on 62%. There was made an effort to study how Long Ranger is filtering the variants, but except that it is applying a structural variant's (SV) blacklist filter (avoiding segmental duplications) <https://support.10xgenomics.com/genome-exome/software/pipelines/latest/algorithms/overview> the author was unable to find any other documentation. There was made a try to filter LinkSeq's variants which overlapped with the SV filter BED-file which Long Ranger uses (this removed only 124 variants), but the precision remained the same (61.5%) for the INDELs. When Long Ranger is able to filter the INDELs, there should be a way – we just have to find it.

It is of importance that LinkSeq's single sample calling filtering is improved for INDELs as these results indicate that 38% of them can be expected to be false positive calls.

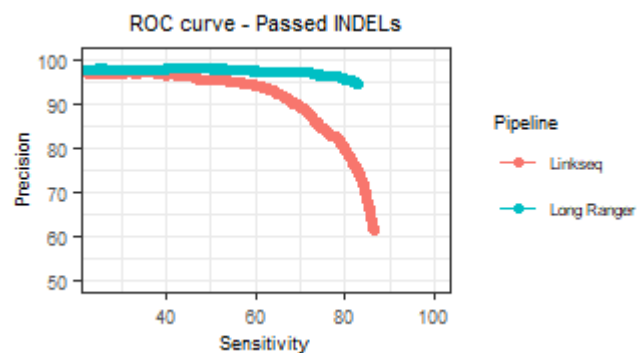
The hard filter does however seem to filter out some FP variants and when looking at [https://github.com/ElisabetThomsen/MSc\\_thesis/blob/master/results/TPFPhistoINDEL.pdf](https://github.com/ElisabetThomsen/MSc_thesis/blob/master/results/TPFPhistoINDEL.pdf), it seems that the parameter leading to this is mostly quality by depth (QD). Fang *et al.* did report that many INDEL errors were induced by low coverage (depth)<sup>15</sup>, so it seems reasonable that QD could filter some FP INDELs out.

As these results are based on only one sample it is unsure if this can be generalized to other samples. But it can be considered if the hard filter by QD should be kept in LinkSeq. It is also a possibility to try the coverage of the alternative allele and the k-mer Chi-Square score as filtering parameters, as Fang *et al.* suggest.<sup>15</sup>

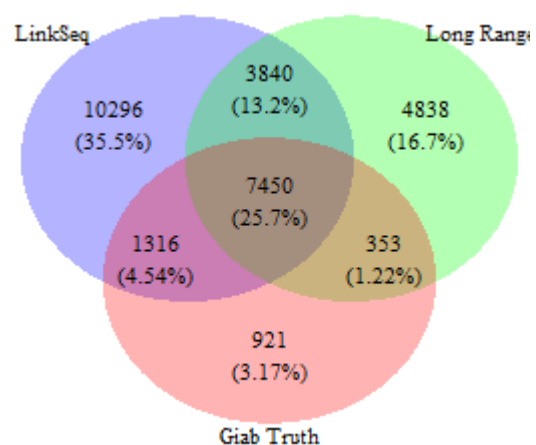
It would also be desirable to benchmark on more samples, just to exclude that Long Ranger perhaps could be overfitted for NA12878, as this is the universal sample, so the software may have been developed from it. The data is also taken from 10X's webpage, so it can be biased. But it was the only 10X exome data with a high confident VCF-file that the author could find.

The ROC-curve shows that when the sensitivity increases more than 0.4 the precision greatly drops for LinkSeq. Long Ranger's precision can be increased much more (~ 0.8) before the sensitivity drops (fig. 11).

Of the union of INDELs (fig. 12) there are 3% in the truth VCF which are missed by both callers, indicating that they are doing worse for INDELs than SNPs. This time LinkSeq is the pipeline, who is calling more FP INDELs (49%) compared to Long Ranger (30%). There are 13% FP INDEL calls that are shared between the pipelines and not in the truth VCF. The reason that FP SNP and INDEL calls are shared between the pipelines, is presumably because they are called from the same exome data – Fang *et al.* stated that INDELs call errors



**Fig 11:** ROC curve for LinkSeq and Long Ranger, NA12878, filtered SNPs.



**Fig 12:** Venn diagram showing how the INDELs called by LinkSeq and Long Ranger overlapping with Giab's truth VCF. Only INDELs in target regions. Unfiltered. NA12878.



was frequently coming from e.g. multiple signatures<sup>15</sup>, so this could be the reason that they have so many errors.

39% of all found INDELs (GiaB’s INDELs included) in the targeted regions are common for LinkSeq and Long Ranger. This is much lower than for SNPs (76%). The explanation for this could again be that the INDELs could have multiple signatures<sup>15</sup> and therefore they differ more between the pipelines.

### 3.3.1.2 Multi-sample calling

To get insights into if VQSR was better than hard filter, NA12878 was called with the multi-sample mode of LinkSeq together with three samples from the FarGen data. As the samples are supposed to be of same ethnicity the three FarGen samples were also multi-sample called separately from NA12878. The same FarGen samples were also called by Long Ranger (single-sample mode) to be able to compare them.

#### 3.3.1.2.1 NA12878 together with FarGen samples

When applying VQSR on NA12878 together with three FarGen samples, the filtration was improved for SNPs but worse for INDELs (table 1).

Have in mind that the recalibration is far from optimal – only five exomes are used, and the Faroese data will highest likely be divergent from the NA12878 sample. If proper VQSR could be applied, it would presumably improve the filtration more.

The results in table 1 partially support GATK’s statement<sup>11</sup> that VQSR is better than Hard Filter – at least for SNPs. It would be good to look more into the filtration of INDELs with proper recalibration, to make sure that they are filtered properly and don’t have a precision as low as 50-60%.

A reason behind the INDEL’s filtration being worse could be if the FarGen samples have INDELs that differ from the known INDELs and NA12878’s INDELs are “forced” to be similar to the FarGen samples’ INDELs as they are called together. However, it seems odd that the INDELs get worse and not the SNPs. Another explanation could be that the placement of the INDELs can be more flexible than the placement of SNPs, leading to more multiple signatures for INDELs especially in regions where homopolymers are present.<sup>15</sup>

If the variants are to be used in functional analysis it is important that the INDELs are called in the correct positions, because this will affect which amino acids are altered. The effect on the protein will differ greatly if e.g. the INDEL is placed in the bases coding for amino acids in the active side on an enzyme or in a subunit with low functionality.

#### 3.3.1.2.2 FarGen samples

When comparing LinkSeq’s multi-sample called variants in the FarGen samples against the existing VCF-files from other studies (table 2), they seem to have higher sensitivity, but lower precision than Long Ranger. LinkSeq has the better harmonic mean for the

	SNPs (%)	INDELs (%)
<i>Sensitivity</i>	97.2	84.1
<i>Precision</i>	99.2	53.7
<i>Harmonic mean</i>	98.2	65.5

**Table 1:** Statistics produced by hap.py (PASSED SNPs and INDELs). NA12878 multi-sample called together with three FarGen samples with LinkSeq. Long Ranger is not shown in this table, because it can only do single-sample calling, so the numbers would be the same as in fig. 6 and 10.

	LinkSeq		Long Ranger	
	SNPs	INDELs	SNPs	INDELs
<i>Sensitivity (%)</i>	45	47	30	22.5
<i>Precision (%)</i>	64	18	71	62.5
<i>Harmonic mean (%)</i>	53	27	42	34

**Table 2:** Statistics produced from GATK Concordance (harmonic mean calculated by hand). Three FarGen samples, multi-sample called with LinkSeq and single-sample called by Long Ranger.

SNPs, but the opposite applies for INDELs.

The results for the FarGen samples, is not really comparable with the others, because another tool is used. The sensitivity, precision and harmonic mean are much worse, presumably because the high confident regions, which is used in hap.py benchmarking, is not used in GATK's Concordance. Once again it is observed that the INDELs have much lower precision, especially for LinkSeq (only 18%).

Using the VCF from another study as the truth VCF is very questionable, because for the variants not shared by the pipelines it is not possible to know who of them are more correct. It might be the case that some of the variants found by e.g. LinkSeq and not are in the old VCF are in fact true and wrong for the old VCF.

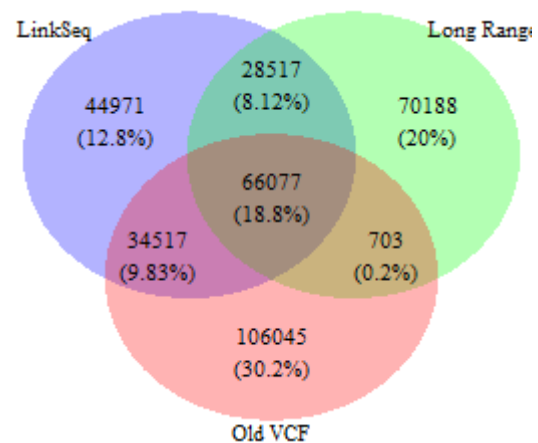
To compare them in a better way, Venn diagrams were made. For SNPs the old VCF seems to have highest quantity of SNPs and least in common with the other two pipelines – which perhaps not is unexpected when LinkSeq and Long Ranger share the same data. Differently from NA12878 (fig. 8 and 12) LinkSeq is now calling more SNPs (174082) than Long Ranger (165485) (fig 13). 19% of all SNPs are shared by all three pipelines. For INDELs (fig. 14) LinkSeq has the highest quantity and only 7% of all INDELs are shared by all pipelines.

According to the literature it is much harder to get the INDELs correct, compared to SNPs – especially for short reads.<sup>26,34</sup> According to Hasan *et al.*, who were benchmarking seven callers (including HaplotypeCaller) against each other, 76.9% of gold standard INDELs are undetected by all tools.<sup>26</sup> They therefore state that there is a great need for improving INDEL calling.

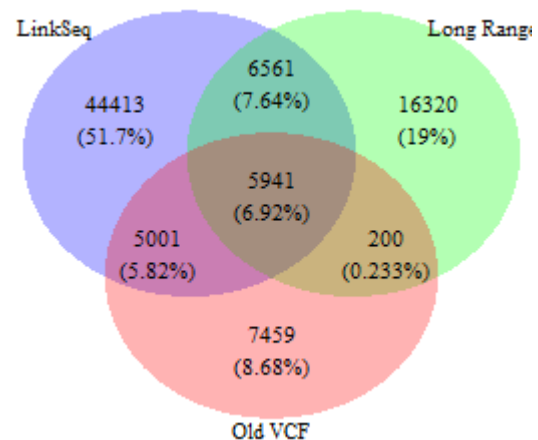
In their article they also state that there is a great need for a benchmarking dataset with a list of known indels per sample in large quantity.<sup>26</sup> As they do not have this, they use Mills *et al.* known indels<sup>35</sup> as the truth. This can lead to bias as there is no guarantee that each sample will have identical indels as in the gold standard.

In this project we have indeed also been in need of a proper large quantity benchmarking dataset, especially for the multi-sample calling. Using FarGen's samples to benchmark the multi-sample calling made it possible to try to run the multi-sample calling, but the results don't really tell if LinkSeq or Long Ranger is better, but only which of the pipelines is more similar to the pipeline used in the other study that called the other VCFs we are comparing with.

However, as it seemed like LinkSeq in this case was calling a lot more INDELs than Long Ranger (higher sensitivity), but had much lower precision, there was made a try to study the INDELs called by LinkSeq – to see if there maybe were INDELs close by in the "truth" VCF. Li *et al.* define a INDEL to be true, if there exists a INDEL within 20 bp from the left-aligned starting position of the INDEL in the truth VCF<sup>36</sup>. Because of this, all INDELs in the truth VCF that were within 20bp range of INDELs



**Fig 13:** Venn diagram showing how the SNPs called by LinkSeq, Long Ranger and the old pipeline overlap. Only SNPs in target regions. Unfiltered. One of the FarGen samples.



**Fig 14:** Venn diagram showing how the INDELs called by LinkSeq and Long Ranger overlapping with GiaB's truth VCF. Only INDELs in target regions. Unfiltered. One of the FarGen samples.



found by LinkSeq were counted and put together into a tab separated file to study them visually (Python was used for this). However, studying them this way did not give any clue on what was going on. And it did not seem like there were so many INDELs within 20bp.

There was unfortunately not time to study the difference in the INDELs further, however, it would be interesting to investigate if the INDELs not shared between the pipelines are due to multiple signatures or/and are in regions with homopolymers, so Fang *et al.* suggest.<sup>15</sup>

### 3.3.2 Phase evaluation

In NA12878 Long Ranger found more variants, however LinkSeq had a higher percentage of heterozygous sites and was able to phase a higher percentage of the heterozygous sites (see Table 3).

Long Ranger's haploblocks seem to be longer as Long Ranger's median block length, average block length and longest block are 47682.5 bp, 123389.74 bp and 3045509 bp, respectively, longer than LinkSeq's (see Table 3).

	LinkSeq	Long Ranger
Variants in VCF	137551	688493
Heterozygous sites (%)	66	44
Heterozygous sites phased (%)	73	53
Median block length (bp)	16282.50	63965.00
Average block length (bp)	34239.63	157629.37
Longest block (bp)	596134	3641643

**Table 3:** The phase statistics produced by WhatsHap. The total number of variants is ALL variants and not restricted to target region as in section 3.3.1.

When looking at the variants' correctness LinkSeq had only 55 and 348 switch and mismatch errors, respectively (Table 4). This does not seem so high, when having in mind that 67017 variants were phased.

	Linkseq
Switch errors	55
Mismatch errors	348
N50	113710

WhatsHap's statistics indicate which pipeline phased more variants and had longer haploblocks. However, having more phased variants and longer haploblocks does not mean that they are more correct. And from the variant correctness results it is already seen that more variants are found, greater is the chance that some of them are wrong. This cannot be said for sure for Long Ranger's phased variants, because it was not possible to estimate the phase correctness. However, GiaB has some statistics on the NA12989 sample ([ftp://ftp-trace.ncbi.nlm.nih.gov/giab/ftp/release/NA12878\\_HG001/latest/README\\_NISTv3.3.2.txt](ftp://ftp-trace.ncbi.nlm.nih.gov/giab/ftp/release/NA12878_HG001/latest/README_NISTv3.3.2.txt)). In GRCh38 they were able to phase 98% of all found variants. 60 % and 98 % of found SNPs and INDELs, respectively, were heterozygous. As LinkSeq found 68% heterozygous sites compared to Long Ranger's 44%, it seems like LinkSeq is closer to GiaB heterozygosity statistics. However, these numbers should be taken with caution as GiaB's statistics are for WGS data and we have WES data.

**Table 4:** The phase statistics produced by error\_rates.py

A reason behind Long Ranger's haploblocks are longer, could be that 553480 of the variants it finds are outside the target regions compared to LinkSeq's (9819 variants). Having variants over a greater distance is of course an advantage when trying to get longer haploblocks. Why Long Ranger has this great number of variants outside the target regions, is not easy to interpret, when it is hard to get insight into how the algorithms are working inside Long Ranger. In LinkSeq more or less all processes are restricted to the targeted BED-file. The Long Ranger pipeline is fed with the same targeted BED-file, but it is unsure in which processes it is used. The sensitivity and precision calculated in Section 3.3.1 are estimated only within the targeted and confident regions as recommended by GiaB. If all variants were included, this could change the variant correctness drastically. As many variants are called outside the targeted regions, it could indicate that either something has gone wrong in the exome capture procedure, which is hard to tell, as the data is given by 10X. Or it can mean that reads are aligning outside the targets where they should not – calling variants they should not. In the latter case, Long

Ranger's precision can be expected to become much worse. But on the other hand, Long Ranger has a great filtering process, so it might be able to remove the FP variants as it did in the target regions. As for the haploblocks it would have been really great to estimate switch errors for Long Ranger, knowing that the vast majority of variants are outside the target regions. However, this was not possible, because Long Ranger did not produce the block file needed.

When it was noticed that the `error_rates.py` script was dependent on the block file HapCut2<sup>12</sup> produced and Long Ranger did not produce this file, there was made an effort to find some other phase benchmarking tool, but this was not successful. It must therefore be stated that there seems to be lacking a phase benchmarking tool, that can take VCF-files as input. Differently from the block file, which not all phasing tools produce, all phasers produce a VCF-file so it would be more appropriate to use this filetype for benchmarking.

### 3.3.3 Performance

When Long Ranger is unrestricted and given free access to the server's 96 cores and ~ 536GB		<i>LinkSeq</i>	<i>Long Ranger restricted</i>	<i>Long Ranger unrestricted</i>
	<i>Duration</i>	13h 38m 2s	7d 20h 50m 22s	12h 52m 17s
	<i>CPU hours</i>	89	755.4	51.5
	<i>Memory Peak (GB)</i>	31.6	12.6	249.8

**Table 5:** The performance of LinkSeq with access to in total XX cores and XX GB memory (4 cores and 20 GB memory per process), Long Ranger unrestricted and Long Ranger restricted to 4 cores and 20 GB memory.

memory it outperforms LinkSeq in duration by being ~ half an hour faster and using 37.5 less CPU hours. However, its memory peak usage is 218.2 GB higher. When Long Ranger was restricted to 4 cores and 20GB it took ~ 7 days longer than LinkSeq and used 666.4 more CPU hours, but the memory peak was 19 GB less.

According to the author's knowledge there is no publication on the performance of an entire linked-read exome pipeline where all the steps are included. However, Shajii *et al.* compared EMA's performance against Lariat's (the aligner inside the Long Ranger pipeline) on a 287 GB raw dataset and found that EMA was around 7 hours faster.<sup>7</sup> Laurie *et al.* compared the variant calling of FreeBayes (Long Ranger's caller) and HaplotypeCaller (LinkSeq's caller) on a exome dataset aligned by BWA-MEM and found that FreeBayes used 5.3 and 9.4 hours and CPU hours, respectively, less than HaplotypeCaller.<sup>25</sup> When adding these two together, one can expect LinkSeq to use less time than Long Ranger in the alignment, but use more time on the calling. Looking at LinkSeq's timeline ([https://github.com/ElisabetThomsen/MSc\\_thesis](https://github.com/ElisabetThomsen/MSc_thesis)), it can be seen that the alignment takes around 1 hour and the calling takes around 5 hours, so this makes sense. Unfortunately, from the timeline produced by nextflow it is not possible to see how long time each process inside Long Ranger takes, so they cannot be compared to Laurie *et al.*'s and LinkSeq's numbers. Shajii *et al.* do not provide the CPU time for their comparison, however as they are gaining time by parallelisation, one might speculate if EMA's CPU hours might be higher.

When restricting Long Ranger, the same CPU and memory restrictions were set in the nextflow configuration file as were given to LinkSeq. However, the performance comparison of LinkSeq and Long Ranger in nextflow cannot be said to be optimal, as it is possible to monitor and adjust each individual process inside LinkSeq, but everything that is inside Long Ranger counts as one process. Therefore, when running the scripts Long Ranger only receives 4 cores and 20 GB memory in total, while LinkSeq is able to give 4 cores and 20 GB to *each process* and in total have 40 cores and 200 GB. This is probably the reason that when restricting Long Ranger it took 7 days longer than LinkSeq.

While the restriction results are not fair towards Long Ranger, the unrestricted results are unfair for LinkSeq. Now Long Ranger has 96 cores, while LinkSeq has only 40 cores. As Long Ranger unrestricted is only ~ 1 hour faster than LinkSeq, a qualified guess is that LinkSeq would probably be faster than Long Ranger if they were benchmarked properly. However, it is hard to estimate who would have less CPU hours and memory peak usage, as the restriction are very different for the current results. E.g. Long Ranger unrestricted has lower CPU hours, but when it can use more memory, maybe it will use less CPU time.

### **3.4 EFFECT OF REMOVING BARCODES**

Surprisingly, the results of removing the barcodes does not seem to have an effect at all. There was less than 1% difference in the variant correctness measurements between the barcode trimmed and barcode contaminated reads.

This can either be because the aligner is clipping the reads with barcode contamination, making them having good alignments, or it can be because the dataset was unsuitable for testing the matter of concern.

The NA12878 sample only had around 2-3% reads that were contaminated with barcodes, so maybe they are too few to test it. This can also be seen as the sensitivity and precision are low (5-58%), most likely due to low coverage.

Furthermore, the R1 file remains unchanged when the R2 file is trimmed. Perhaps, when the R1 reads are aligning equally well and BWA is aware of the insert size between the paired reads and capable of clipping<sup>22</sup> and EMA is aware of the read clouds of identical barcodes<sup>7</sup>, then they will figure out to map the R2 close to the R1 at its proper location.

One option would be to only align the R2, however as LinkSeq is intended to be applied for paired end reads, it does not make sense to test single end reads.

As the results are now, barcode trimming does not seem to have an effect on the variants.

## 4 CONCLUSION

---

By developing LinkSeq we are now close to having an open source pipeline for Linked-reads, which do not have many other alternative pipelines. LinkSeq will make it possible to achieve multi-sample calling, which is claimed to be the superior calling method for case-control studies, which the FarGen data is intended to be used for.

By comparing the variants called by LinkSeq with a truth dataset, it is now known at which level LinkSeq's variant correctness is lying and areas in LinkSeq that can be improved have been found. Especially the INDEL calls, which had a low precision both in single-sample calling (62%) and multi-sample calling (54%) seem to need improvement. It cannot be recommended to use LinkSeq's current state INDEL calls for research of other appliance as 38-66% of them can be expected to be FP calls. The SNPs do however look better.

By comparing LinkSeq's single-sample calls with Long Ranger's it was discovered that it is mainly in the variant filtration process that Long Ranger is outcompeting LinkSeq both in SNPs and INDELs.

Plots of the hard filter parameters showed that hard filter (which LinkSeq single-sample is using) is not a good solution to filter FP variants from TP variants. If the single-sample version of LinkSeq is to be used, it is possible to try other filtering tools e.g. GARFIELD-NGS<sup>32</sup> or VarBin<sup>33</sup>. LinkSeq multi-sample (which will be used for the FarGen data) uses VQSR to filter variants. Here the harmonic mean for filtered SNPs is 98%, which is only 1% behind Long Ranger's. These results are good and might have been even higher, if it was possible to call and filter the multi sample called variants from samples of proper quantity and ethnicity.

Now remains only to get LinkSeq's joint\_genotyping script to run with a large quantity of samples, then SNPs can be called from the FarGen datasets (including HBOC, ADHD, IBD etc.) and these can be used for research and be part of the Faroese health infrastructure and therefore be of great appliance.

As for the INDELs, they need to be investigated further, if they are to be used. This is not an easy task, as long as a benchmarking dataset of large quantity, same ethnicity and truth VCFs is missing. Originally the plan was to simulate reads for this purpose, but it soon turned out to be rather cumbersome as the reads needed to be simulated from minimum 30 individuals of same population. But it is a possibility, if benchmarking is desirable before a real dataset is available.

Propper benchmarking of phase correctness was neither possible, due to the lack of a software that could benchmark the phase from VCF-files. Long Ranger had longer haploblocks, but it called more variants outside the targeted regions and therefore its phase correctness can be questioned.

LinkSeq's performance looks good when comparing it with Long Ranger's. However, it is hard to compare these results properly, as it was not possible to restrict the pipelines equally through the nextflow config files.

## 5 ACKNOWLEDGEMENT

---

First, I would like to thank FarGen and DTU for making this Master project possible. Thanks to Elsa, my other office mates and people in the “research house” (granskingarlonini) for keeping me company while doing the bioinformatics and writing this thesis. Thanks to my family, friends and especially my partner Sergio for their support and understanding of my absence due to the countless hours spent on this thesis. Thanks to my supervisor Ólavur Mortensen for teaching me to work on FarGen’s server and GitHub, making conda environments, coding in nextflow and a lot of other stuff and for all the hard work he put into trying to get the last processes in LinkSeq to work so we could benchmark the scripts. And thanks to my other supervisor prof. Gisle Alberg Vestergaard for offering to be my DTU supervisor and taking time for all the online meetings with good humor as well as the frustrating moments when we realized that it was much harder to benchmark the pipelines than originally thought. Thanks for all the suggestions and advices in the moments when things seemed to get stuck.

## 6 REFERENCES

---

1. Marks, P. *et al.* Resolving the full spectrum of human genome variation using Linked-Reads. *Genome Res.* (2019) doi:10.1101/gr.234443.118.
2. Warr, A. *et al.* Exome sequencing: Current and future perspectives. *G3 Genes, Genomes, Genet.* (2015) doi:10.1534/g3.115.018564.
3. Mullaney, J. M., Mills, R. E., Stephen Pittard, W. & Devine, S. E. Small insertions and deletions (INDELs) in human genomes. *Hum. Mol. Genet.* (2010) doi:10.1093/hmg/ddq400.
4. Gagliano, S. A. *et al.* Relative impact of indels versus SNPs on complex disease. *Genet. Epidemiol.* (2019) doi:10.1002/gepi.22175.
5. Mills, R. E. *et al.* An initial map of insertion and deletion (INDEL) variation in the human genome. *Genome Res.* (2006) doi:10.1101/gr.4565806.
6. Meena, N., Mathur, P., Medicherla, K. & Suravajhala, P. A Bioinformatics Pipeline for Whole Exome Sequencing: Overview of the Processing and Steps from Raw Data to Downstream Analysis. *BIO-PROTOCOL* (2018) doi:10.21769/bioprotoc.2805.
7. Shajii, A., Numanagić, I., Whelan, C. & Berger, B. Statistical Binning for Barcoded Reads Improves Downstream Analyses. *Cell Syst.* (2018) doi:10.1016/j.cels.2018.07.005.
8. GATK. Base Quality Score Recalibration (BQSR). <https://gatkforums.broadinstitute.org/gatk/discussion/44/base-quality-score-recalibration-bqsr>.
9. GATK. Why do joint calling rather than single-sample calling? <https://gatkforums.broadinstitute.org/gatk/discussion/comment/11390/>.
10. GATK. I am unable to use VQSR (recalibration) to filter variants. <https://gatk.broadinstitute.org/hc/en-us/articles/360037499012>.
11. GATK. Filter variants either with VQSR or by hard-filtering. <https://gatk.broadinstitute.org/hc/en-us/articles/360035531112>.
12. Edge, P., Bafna, V. & Bansal, V. HapCUT2: Robust and accurate haplotype assembly for diverse sequencing technologies. *Genome Res.* (2017) doi:10.1101/gr.213462.116.
13. Tewhey, R., Bansal, V., Torkamani, A., Topol, E. J. & Schork, N. J. The importance of phase information for human genomics. *Nature Reviews Genetics* (2011) doi:10.1038/nrg2950.
14. O’Rawe, J. *et al.* Low concordance of multiple variant-calling pipelines: Practical implications for exome and genome sequencing. *Genome Med.* (2013) doi:10.1186/gm432.
15. Fang, H. *et al.* Reducing INDEL calling errors in whole genome and exome sequencing data. *Genome Med.* (2014) doi:10.1186/s13073-014-0089-z.
16. 10xGenomics. Targeted Phasing and SV Calling. <https://support.10xgenomics.com/genome-exome/software/pipelines/latest/using/targeted>.
17. GATK. Data pre-processing for variant discovery. <https://gatk.broadinstitute.org/hc/en-us/articles/360035535912-Data-pre-processing-for-variant-discovery>.
18. Tian, S., Yan, H., Kalmbach, M. & Slager, S. L. Impact of post-alignment processing in variant discovery from whole exome data. *BMC Bioinformatics* (2016) doi:10.1186/s12859-016-1279-z.
19. Didion, J. P., Martin, M. & Collins, F. S. Atropos: Specific, sensitive, and speedy trimming of

- sequencing reads. *PeerJ* (2017) doi:10.7717/peerj.3720.
20. Sturm, M., Schroeder, C. & Bauer, P. SeqPurge: Highly-sensitive adapter trimming for paired-end NGS data. *BMC Bioinformatics* (2016) doi:10.1186/s12859-016-1069-7.
  21. Del Fabbro, C., Scalabrin, S., Morgante, M. & Giorgi, F. M. An extensive evaluation of read trimming effects on illumina NGS data analysis. *PLoS One* (2013) doi:10.1371/journal.pone.0085024.
  22. Li, H. & Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* (2009) doi:10.1093/bioinformatics/btp324.
  23. Bishara, A. *et al.* Read clouds uncover variation in complex regions of the human genome. *Genome Res.* (2015) doi:10.1101/gr.191189.115.
  24. Li, H. & Durbin, R. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics* **26**, 589–595 (2010).
  25. Laurie, S. *et al.* From Wet-Lab to Variations: Concordance and Speed of Bioinformatics Pipelines for Whole Genome and Whole Exome Sequencing. *Hum. Mutat.* (2016) doi:10.1002/humu.23114.
  26. Hasan, M. S. habbi., Wu, X. & Zhang, L. Performance evaluation of indel calling tools using real short-read data. *Hum. Genomics* (2015) doi:10.1186/s40246-015-0042-2.
  27. R Studio Team. R Studio. *R.S. ed.* <http://www.rstudio.com/>. (2015) doi:<http://www.rstudio.com/>.
  28. Van der Auwera, G. A. *et al.* From fastQ data to high-confidence variant calls: The genome analysis toolkit best practices pipeline. *Curr. Protoc. Bioinforma.* (2013) doi:10.1002/0471250953.bi1110s43.
  29. Quinlan, A. R. & Hall, I. M. BEDTools: A flexible suite of utilities for comparing genomic features. *Bioinformatics* (2010) doi:10.1093/bioinformatics/btq033.
  30. Martin, M. *et al.* WhatsHap: fast and accurate read-based phasing. *bioRxiv* (2016) doi:10.1101/085050.
  31. Bushnell, B. BBMap. [sourceforge.net/projects/bbmap/](http://sourceforge.net/projects/bbmap/).
  32. Ravasio, V., Ritelli, M., Legati, A. & Giacomuzzi, E. GARFIELD-NGS: Genomic vARiants FIltering by dEep Learning moDEls in NGS. *Bioinformatics* (2018) doi:10.1093/bioinformatics/bty303.
  33. Durtschi, J., Margraf, R. L., Coonrod, E. M., Mallempati, K. C. & Voelkerding, K. V. VarBin, a novel method for classifying true and false positive variants in NGS data. *BMC Bioinformatics* (2013) doi:10.1186/1471-2105-14-S13-S2.
  34. Li, S. *et al.* SOAPindel: Efficient identification of indels from short paired reads. *Genome Res.* (2013) doi:10.1101/gr.132480.111.
  35. Mills, R. E. *et al.* Natural genetic variation caused by small insertions and deletions in the human genome. *Genome Res.* (2011) doi:10.1101/gr.115907.110.
  36. Li, H. Exploring single-sample snp and indel calling with whole-genome de novo assembly. *Bioinformatics* (2012) doi:10.1093/bioinformatics/bts280.
  37. Okonechnikov, K., Conesa, A. & García-Alcalde, F. Qualimap 2: Advanced multi-sample quality control for high-throughput sequencing data. *Bioinformatics* (2016) doi:10.1093/bioinformatics/btv566.

## SUPPLEMENTS

---

Apart from Supplement 1 which shows the coverage across the targeted regions in the reference (made inside LinkSeq by Qualimap<sup>37</sup>) commands and scripts made and used in this thesis can be found in the GitHub repository:

[https://github.com/ElisabetThomsen/MSc\\_thesis](https://github.com/ElisabetThomsen/MSc_thesis)

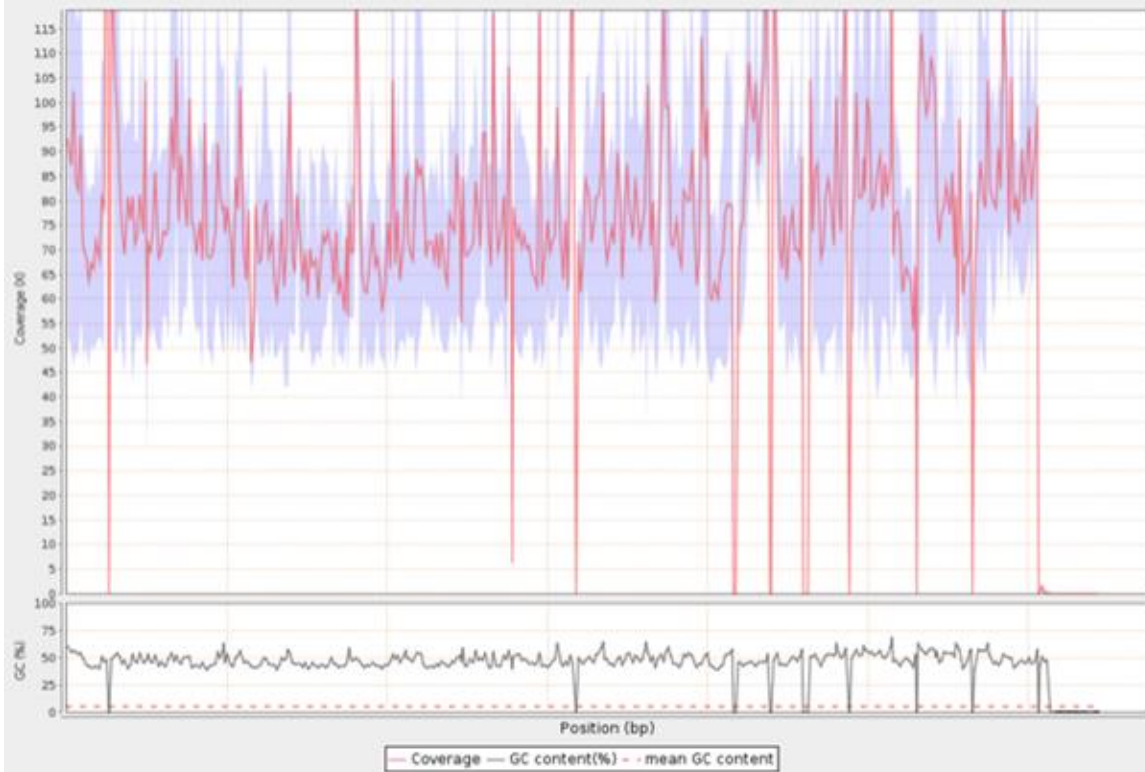


## SUPPLEMENT 1: COVERAGE

NA12878

Coverage across reference

Mean: 83X



FarGen Sample

Coverage across reference

Mean: 44X

