



MAY 24, 2023

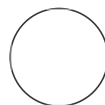
# Investigation and identification of somatic and germline variants for colorectal cancer exomes using the NGS pipeline: a computational analysis perspective

Anagha S

Chandrashekar K<sup>1</sup>, Setlur<sup>1</sup>,

Vidya Niranjana<sup>1</sup>

<sup>1</sup>Department of Biotechnology, RV College of Engineering, Bangalore-560059



Vidya Niranjana

OPEN ACCESS

DOI:

[dx.doi.org/10.17504/protocols.io.x54v9dqxqg3e/v1](https://dx.doi.org/10.17504/protocols.io.x54v9dqxqg3e/v1)

**Protocol Citation:** Chandrashekar K, Anagha S Setlur, Vidya Niranjana 2023. Investigation and identification of somatic and germline variants for colorectal cancer exomes using the NGS pipeline: a computational analysis perspective. **protocols.io** <https://dx.doi.org/10.17504/protocols.io.x54v9dqxqg3e/v1>

**License:** This is an open access protocol distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

**Protocol status:** Working  
We use this protocol and it's working

**Created:** May 23, 2023

**Last Modified:** May 24, 2023

**PROTOCOL integer ID:**  
82294

## DISCLAIMER

This protocol was run with colorectal cancer exome datasets. However, the SRR-IDs of the datasets have not been mentioned to make this protocol universal. The steps may be followed for any cancer exome, with the intent of investigating the somatic and germline mutations.

## ABSTRACT

A thorough analysis on colorectal cancer exomes reveals potential mutations such as single nucleotide polymorphisms that can be beneficial for early detection of the disease. Thus, through a comprehensive computational protocol that identifies, investigates and analyzes the identified variants, this process of early disease detection becomes much simpler. In brief, the cancer exome datasets were retrieved from publicly available databases, followed by performing quality control checks. The datasets that qualified the quality control checks were then aligned with the human reference genome. Somatic and germline mutants were then identified and called separately, with specific tools for each case. Haplotype Caller was employed for germline variant identification, and Mutect2 for somatic. The identified mutants were then normalized, annotated and post-processed using snpEFF, WANNVAR and VEP. This protocol helps in garnering insights on the various alterations that might possibly lead to colorectal cancer and suggests the possibility of utilizing the most important genes identified for wet-lab experimentation.

**Keywords:** *Colorectal cancer exomes, quality controls, somatic & germline variants, annotation & post-processing, computational analysis*

**Keywords:** Colorectal cancer exomes, quality controls, somatic & germline variants, annotation & post-processing, computational analysis

## GUIDELINES

The commands mentioned in the protocol can be run in the Linux terminal. Enough storage space is recommended while running this protocol.

## MATERIALS

Tools mentioned in the current protocol must initially be installed in the system before running the commands for the NGS pipeline. These include: SRAToolkit, Fastqc, Multiqc, Cutadapt, Bowtie2, Samtools, picard, GATK, and bcftools. wANNOVAR, VEP are available online and snpEFF is command line based.

## SAFETY WARNINGS



## ETHICS STATEMENT

The datasets used to arrive at this protocol were all from publicly available databases such as NCBI-SRA.

## BEFORE START INSTRUCTIONS

All the mentioned tools in the materials section must be installed and checked for appropriate installation.

# COLORECTAL CANCER EXOME RETRIEVAL

## 1 Retrieval of colorectal cancer exomes

This protocol was run with colorectal cancer exome datasets. However, the SRR-IDs of datasets have not been mentioned to make this protocol universal. Therefore, NCBI-SRA database (<https://www.ncbi.nlm.nih.gov/sra>) was employed to retrieve the colorectal cancer exome datasets. The SRR-IDs were noted down for further use. The exome datasets were selected based on a set of criteria. The strategy used for the datasets must be whole exome sequenced and the exomes should be of genomic source. In the present protocol, all the selected sequences were of paired layout, with Illumina used for sequencing and acquiring the reads. Other criteria as desired by the user depending on the requirement can be employed for retrieving the cancer exomes.

Additionally, the tools mentioned in the current protocol must initially be installed in the system before running the commands for the NGS pipeline. These include: SRAToolkit, Fastqc, Multiqc, Cutadapt, Bowtie2, Samtools, picard, GATK, and bcftools. wANNOVAR, VEP are available online and snpEFF is command line based.

## 2 Quality control assessments

Prior to calling variants, the quality of raw data was assessed using the below-mentioned steps. The codes used for running the quality checks are provided in the following sections.

### 2.1 Quality Check using FastQC and MultiQC (for multiple datasets)

FastQC (<https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>) and MultiQC (<https://multiqc.info/>) in the present study were used to effectively assess the sequencing data quality, to identify any possible issues with the raw sequence reads and generate informative reports. Scrutiny of the mean sequence quality per reading and per base, nucleotide content per base position, distribution of GC, etc were analyzed via FastQC. A cumulative report for all FastQC outcomes was obtained from MultiQC.

For multiple datasets, FastQC and MultiQC commands used were:

#### Command

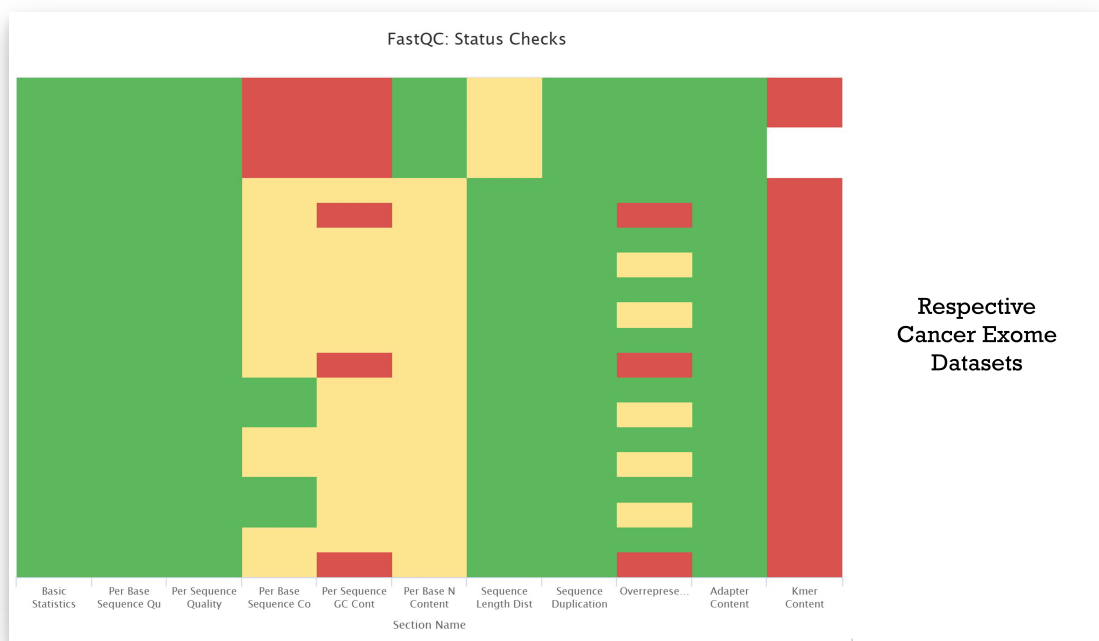
##### fastQC

```
$  
./fastqc
```

#### Command

##### multiQC

```
$ multiqc .
```



### Expected multiQC heatmap

## 2.2 Adaptor sequence removal via Cutadapt

This tool identifies and eliminates the adaptor sequences from raw reads and improves the quality and accuracy of downstream analysis. All unwanted sequences were thus removed using Cutadapt (<https://cutadapt.readthedocs.io/en/stable/>) by running:

#### Command

#### Running Cutadapt

```
$ cutadapt -a adapter-sequence* -o SRRID_cut.fastq.gz
SRRID.fastq.gz
```

## ALIGNMENT WITH HUMAN REFERENCE GENOME

### 3 Retrieval and indexing of human reference genome for analysis

Indexing is a necessary step to pre-process the reference genome so the aligner can seek potential alignment locations for the reads. This is useful during alignment with Bowtie2

(<https://bowtie-bio.sourceforge.net/bowtie2/index.shtml>), where short read aligners are processed with the reference genome. The following codes were run in the terminal for obtaining Bowtie2 outputs.

#### Command

##### Unzipping the reference genome file

```
$ gunzip  
GRCh37_latest_genomic.fna.gz
```

#### Command

##### Indexing reference genome

```
$ bowtie2-build -f GRCh37_latest_genomic.fna.gz hg19
```

### 3.1 Alignment of indexed human reference genome against exome datasets

Once indexing is complete, alignment was performed using Bowtie2.

#### Command

##### Gapped alignment command with human reference genome (hg19 in this case)

```
$ bowtie2 -x hg19/38/37 -1 SRRID_1_cut.fastq.gz -2 SRRID_2_cut.fastq.gz -S SRRID.sam  
$ head SRRID.sam
```

Running Bowtie2 with the specified parameters allows alignment of paired-end sequencing reads to a reference genome, enabling downstream analysis such as variant calling, differential gene expression, or identification of genomic features specific to colorectal cancer.

## SAM TO BAM CONVERSION

### 4 SAM (Sequence Alignment/Map) to BAM (Binary Alignment/Map) conversion

To obtain the alignment outcomes in a readable format and to act as appropriate input for the next steps, SAM to BAM (<http://www.htslib.org/>) conversion was performed. The code used was:

#### Command

##### SAM to BAM file conversion

```
$ samtools view -bS SRRID.sam > SRRID.bam
```

#### 4.1 Sorting BAM

BAM sorting and merging was performed using command:

#### Command

##### BAM sorting

```
$ samtools sort SRRID.bam -o SRRID.sorted.bam
```

BAM sorting helps in supporting quick retrieval of alignments and has a compact size for further analysis.

#### 4.2 Identifying and marking duplicate reads in BAM file

Duplicate reads may arise at times during sequencing due to several factors, inclusive of technical biases, PCR amplification artifacts and preparation of library protocols. These may skew the downstream analyses and impact the result accuracy. Therefore, marking duplicates is essential to address this issue, by using the “MarkDuplicates” function of Picard (<https://broadinstitute.github.io/picard/>) tool.

#### Command

##### Marking Duplicates

```
$ java -jar picard.jar MarkDuplicates INPUT=SRRID_sorted_reads.bam  
OUTPUT=SRRID_dedup_reads.bam METRICS_FILE=SRRID_metrics.txt
```

### 4.3 Sorting BAM files for deleted duplicate files

To improve the compression efficiency, another round of BAM sorting was performed for the deleted duplicate files.

#### Command

##### BAM sorting for deleted duplicate files

```
$ samtools sort SRRID_dedup_reads.bam -o SRRID_sorted_dedup_reads.bam
```

### 4.4 Computing and collecting alignment summary metrics from BAM file

To obtain various statistics and metrics associated with alignment of sequencing reads to a reference genome, the “CollectAlignmentSummaryMetrics” function was used. The command used to run this function was:

#### Command

##### Collecting Alignment Summary Metrics

```
$ java -jar picard.jar CollectAlignmentSummaryMetrics R=ref.fa  
I=SRRID_sorted_dedup_reads.bam O=SRRID_alignment_metrics.txt
```

ref.fa: fasta sequence file of the human reference genome

## 4.5 Estimating and collecting the insert size distribution of paired-end reads from BAM file

The primary purpose of collecting the insert sizes is to check the distribution and characteristics of the DNA fragment sizes, in a sequencing library. This enables the collection and visualization of the insert size metrics and distribution, that are useful for the quality scrutiny of the library, alignment of the reads, selection of the fragment size and quality control assessments in several genomic analyses. For this purpose, in the present work flow, the “CollectInsertSizeMetrics” function was used. The distribution of the paired end reads from BAM was analyzed using the command:

### Command

#### Collecting Insert Size Metrics

```
$ java -jar picard.jar CollectInsertSizeMetrics INPUT=SRRID_sorted_dedup_reads.bam  
OUTPUT=SRRID_insert_metrics.txt HISTOGRAM_FILE=SRRID_insert_size_histogram.pdf
```

## 4.6 Calculating coverage depth at each position in a sorted and duplicate-marked BAM file

The “samtools depth” function was used to examine the depth of coverage at every position in the genome of reference depending on the aligned reads. Computation and estimation of depth of coverage important for assessing the sequence data quality, calling of variants, structure and gene expression analysis and overall exploration of data becomes much easier with the samtools depth function. Thus, to better understand and obtain insights into the coverage depth, this function was performed via:

### Command

#### Determining coverage depth of sorted, deleted, & duplicated BAMs

```
$ samtools depth -a SRRID_sorted_dedup_reads.bam > SRRID_depth_out.txt
```

## 4.7 Add or modify read group information in BAM file



To assign or update the read group information to the sequence reads in a BAM file, the add or replace read groups function was employed. This was carried out using "picard.jar AddOrReplaceReadGroups". Sample identification, tracking of the data, sequence data differentiation, integrity checks of data and overall sequence reads compatibility was obtained after this step was performed, ensuring that the reads were properly annotated and organized, facilitating meaningful data analyses.

This step was run using the following code in the terminal:

#### Command

##### Adding/modifying read group info

```
$ java -jar picard.jar AddOrReplaceReadGroups I=SRRID_sorted_dedup_reads.bam  
O=SRRID_output.bam RGID=4 RGLB=lib1 RGPL=ILLUMINA RGPU=unit1 RGSM=20
```

## 4.8 Indexing BAM file- necessary before variant calling

Prior to variant calling, another indexing of the final BAM output file was carried out to allow faster retrieval of specific genomic regions, enhanced visualization, and to facilitate random access.

#### Command

##### Indexing BAM file

```
$ samtools index SRRID_output.bam
```

## VARIANT CALLING - GERMLINE & SOMATIC

### 5 Calling of germline variants

The below sections detail preliminary steps followed for calling germline variants.

Mutations from the sequenced data were identified, processed and the variants were called using PICARD (<https://broadinstitute.github.io/picard/>) and GATK (The Genome Analysis Toolkit, <https://github.com/broadinstitute/gatk>). The Haplotype caller function was used to call the germline variants.

The purpose of using the "HaplotypeCaller" function is to identify and call genetic variants, including single nucleotide variants (SNVs), insertions, deletions, and complex variants, based on the sequencing data and the provided reference genome. The following command was employed to call the variants:

#### Command

##### Germline variant calling

```
$ gatk HaplotypeCaller -R ref.fa -I SRRID_output.bam -O  
SRRID_raw_variants.vcf
```

## 5.1 Selection of SNPs from raw variants

The "SelectVariants" function with the "-select-type SNP" option was utilized to filter and extract only the SNPs from the original VCF file. This step enabled filtering and data refining to focus on SNPs, simplified further analyses, optimized computational resources, and ensured compatibility with SNP-specific analysis tools or pipelines. For working with larger datasets, this function works best for SNP-centric analyses. Therefore, to select the SNPs, the code run was:

#### Command

##### Selection of variants from raw mutants

```
$ gatk SelectVariants -R ref.fa -V SRRID_raw_variants.vcf -select-type-to-include SNP -O  
SRRID_raw_snps.vcf
```

## 5.2 Selection of INDELs from raw variants

The purpose of using the "SelectVariants" function with the "-select-type INDEL" option is to filter and extract only the INDELs from the original VCF file. Using the GATK "SelectVariants"

with the "-select-type INDEL" option allows for the selection and extraction of INDELs from a VCF file. This step enables filtering and refining the data to focus on INDELs. This step was run prior to variant filtration.

#### Command

##### INDELs selection from raw mutants

```
$ gatk SelectVariants -R ref.fa -V SRRID_raw_variants.vcf -select-type-to-include INDEL -o SRRID_raw_indels.vcf
```

### 5.3 Variant filtration

To remove potential false positives and retain only the high quality variants, a series of filters were applied to the raw SNPs using the following codes:

#### Command

##### Calling SNPs

```
$ gatk VariantFiltration -R ref.fa -V SRRID_raw_snps.vcf -O SRRID_filtered_snps.vcf -filter-name "QD_filter" -filter "QD < 2.0" -filter-name "FS_filter" -filter "FS > 60.0" -filter-name "MQ_filter" -filter "MQ < 40.0" -filter-name "SOR_filter" -filter "SOR > 4.0" -filter-name "MQRankSum_filter" -filter "MQRankSum < -12.5" -filter-name "ReadPosRankSum_filter" -filter "ReadPosRankSum < -8.0"
```

#### Command

##### Calling INDELs

```
$ gatk VariantFiltration -R ref.fa -V SRRID_raw_indels.vcf -O SRRID_filtered_indels.vcf -  
filter-name "QD_filter" -filter "QD < 2.0" -filter-name "FS_filter" -filter "FS > 200.0" -filter-  
name "SOR_filter" -filter "SOR > 10.0"
```



## 5.4 Excluding filtered variants

To filter out the variants that do not meet the specific quality (as defined by the user), the “-exclude-filtered” option in the “SelectVariants” step was used. This ensured removal of mutants that failed specific criteria and thereby enhanced quality of the data, prioritized the high-confidence variants only, decreased the number of false positives, allowed compatibility access between datasets and also optimized the computational resources. This is a crucial step to be followed in the data refinement process.

#### Command

##### Excluding SNPs

```
$ gatk SelectVariants --exclude-filtered -V SRRID_filtered_snps.vcf -O  
SRRID_bqsr_snps.vcf
```

#### Command

##### Excluding INDELs

```
$ gatk SelectVariants --exclude-filtered -V SRRID_filtered_indels.vcf -O  
SRRID_bqsr_indels.vcf
```

## 5.5 Recalibration of base quality

To augment the accuracy and reliability of the called variants, a base recalibration was carried out by correcting the systematic errors, addressing biases, and providing more accurate base quality scores. This step is essential in ensuring that the sequencing data quality is maintained appropriately.

### Command

#### Base recalibration- Step 1

```
$ gatk BaseRecalibrator -R ref.fa -I SRRID_output.bam --known-sites  
SRRID_bqsr_snps.vcf --known-sites SRRID_bqsr_indels.vcf -O SRRID_recal_data.table
```

### Command

#### Base recalibration- step 2

```
$ gatk ApplyBQSR -R ref.fa -I SRRID_output.bam -bqsr SRRID_recal_data.table -O  
SRRID_recal_reads.bam
```

### Command

#### Base recalibration- step 3

```
$ gatk BaseRecalibrator -R ref.fa -I SRRID_recal_reads.bam --known-sites  
SRRID_bqsr_snps.vcf --known-sites SRRID_bqsr_indels.vcf -O  
SRRID_post_recal_data.table
```

#### Command

##### Base recalibration- step 4

```
$ gatk AnalyzeCovariates -before SRRID_recal_data.table -after  
SRRID_post_recal_data.table -plots SRRID_recalibration_plots.pdf
```

#### Command

##### Base recalibration- step 5

```
$ gatk HaplotypeCaller -R ref.fa -I SRRID_recal_reads.bam -O  
SRRID_raw_variants_recal.vcf
```

## 5.6 Selection of variants from recalibrated VCF files

To filter and extract only the SNPs from the recalibrated VCF files, functions “SelectVariants” along with “-selectType SNP” were used. This was useful since the data authors were working with were recalibrated variant sets.

#### Command

##### SNP selection

```
$ gatk SelectVariants -R ref.fa -V SRRID_raw_variants_recal.vcf -select-type-to-include  
SNP -O SRRID_raw_snps_recal.vcf
```

#### Command

##### INDEL selection

```
$ gatk SelectVariants -R ref.fa -V SRRID_raw_variants.vcf --select-type-to-include INDEL -  
O SRRID_raw_indels_recal.vcf
```

## 5.7 Variant filtration step

A series of filters were then applied to the raw SNPs to remove the potential false positives and retain high-quality variants. The following filters were applied:

#### Command

##### Filtration

```
$ gatk VariantFiltration -R ref.fa -V SRRID_raw_snps_recal.vcf -O  
SRRID_filtered_snps_final.vcf -filter-name "QD_filter" -filter "QD < 2.0" -filter-name  
"FS_filter" -filter "FS > 60.0" -filter-name "MQ_filter" -filter "MQ < 40.0" -filter-name  
"SOR_filter" -filter "SOR > 4.0" -filter-name "MQRankSum_filter" -filter "MQRankSum < -  
12.5" -filter-name "ReadPosRankSum_filter" -filter "ReadPosRankSum < -8.0"
```

## 6 Calling of somatic variants

The below sections detail preliminary steps followed for calling somatic variants.

### 6.1 Tumor samples with matched normal ones

Somatic variants were called using Mutect2 (<https://gatk.broadinstitute.org/hc/en-us/articles/360037593851-Mutect2>), specific for calling these variants, part of the GATK pipeline.

The purpose of this step is to identify and call somatic variants, which are genetic variations specific to the tumor sample compared to the matched normal sample. This

was first carried out via:

#### Command

##### Calling somatic variants

```
$ gatk Mutect2 -R ref.fa -I tumor.bam -I normal.bam -normal normal_sample_name --germline-resource af-only-gnomad.vcf.gz --panel-of-normals pon.vcf.gz -O somatic.vcf.gz
```

To identify and call somatic variants for more than 2 samples (which are variations specific to the tumor samples compared to the matched normal ones).

#### Command

##### Somatic variant calling for multiple samples

```
$ gatk Mutect2 -R reference.fa -I tumor1.bam -I tumor2.bam -I normal1.bam -I normal2.bam -normal normal1_sample_name -normal normal2_sample_name --germline-resource af-only-gnomad.vcf.gz --panel-of-normals pon.vcf.gz -O somatic.vcf.gz
```

## 6.2 Tumor only mode- without normal samples

To detect and call somatic variants particular to a tumor sample.

#### Command

##### Somatic variant calling for only tumor samples

```
$ gatk Mutect2 -R ref.fa -I sample.bam -O single_sample.vcf.gz
```

Additionally, to call somatic variants while applying filters depending on the population



frequencies and a panel of normals, the following codes were run:

#### Command

##### Somatic variant calling with filters

```
$ gatk Mutect2 -R ref.fa -I sample.bam --germline-resource af-only-gnomad.vcf.gz --  
panel-of-normals pon.vcf.gz -O single_sample.vcf.gz
```

## 6.3 Filtering of somatic variants

Filtering variants are an essential step before the VCF files are prepared, since it ensures that the obtained variants are more likely to be true somatic mutations. In the present work, the mutations were filtered that will further help in understanding the biology of colorectal cancer.

#### Command

##### Filtration of variants

```
$ gatk FilterMutectCalls -V raw_variants.vcf -O  
filtered_variants.vcf
```

The variant files obtained from both germline and somatic calling were then used for further analysis as described in the following steps.

## VCF FILE PREPARATION, ANNOTATION, POST-PROCESSING

### 7 VCF file preparation, annotation and post-processing of variants

Once both germline and somatic mutations were identified and called, the VCF files for both these variants were prepared using bcftools for further analysis. Annotation and processing of these variants were then carried out using snpEFF (<https://pcingola.github.io/SnpEff/>), wANNOVAR (<https://wannovar.wglab.org/>) and VEF (Variant Effect Predictor) (<https://asia.ensembl.org/Tools/VEP>).

## Command

### VCF file prep- step 1

```
$ Vcf-validator SRRID_vcf_file.vcf
```

## Command

### VCF file prep- step 2

```
$ bcftools view -i 'DP>10' SRRID_vcf_file.vcf >
SRRID_filtered_vcf_file.vcf
```

## Command

### VCF file prep- step 3

```
$ bcftools norm -f ref.fasta SRRID_vcf_file.vcf -o SRRID_normalized_vcf_file.vcf
```

```
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT 20
chr1 876499 . A G 308.06 PASS AC=2;AF=1;AN=2;DP=13;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=42;QD=28.01;SOR=3.442 GT:AD:DP:GQ:PL 1/1:0,11:11:32:322,32,0
chr1 877831 . T C 375.06 PASS AC=2;AF=1;AN=2;DP=15;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.21;QD=25;SOR=1.112 GT:AD:DP:GQ:PL 1/1:0,15:15:44:389,44,0
chr1 881627 . G A 570.06 PASS AC=2;AF=1;AN=2;DP=21;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=40.97;QD=27.15;SOR=1.508 GT:AD:DP:GQ:PL 1/1:0,21:21:63:584,63,0
chr1 883625 . A G 428.06 PASS AC=2;AF=1;AN=2;DP=17;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=42;QD=28.54;SOR=3.126 GT:AD:DP:GQ:PL 1/1:0,15:15:44:442,44,0
chr1 888639 . T C 650.06 PASS AC=2;AF=1;AN=2;DP=23;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=40.4;QD=30.96;SOR=0.99 GT:AD:DP:GQ:PL 1/1:0,21:21:62:664,62,0
chr1 888659 . T C 365.06 MQ_filter AC=2;AF=1;AN=2;DP=13;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=39.45;QD=33.19;SOR=2.494 GT:AD:DP:GQ:PL 1/1:0,11:11:33:379,33,0
chr1 889158 . G C 751.06 PASS AC=2;AF=1;AN=2;DP=19;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=40.16;QD=28.2;SOR=1.371 GT:AD:DP:GQ:PL 1/1:0,17:17:51:765,51,0
chr1 889159 . A C 751.06 PASS AC=2;AF=1;AN=2;DP=19;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=40.16;QD=25;SOR=1.371 GT:AD:DP:GQ:PL 1/1:0,17:17:51:765,51,0
chr1 894573 . G A 930.06 PASS AC=2;AF=1;AN=2;DP=36;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.5;QD=29.06;SOR=2.765 GT:AD:DP:GQ:PL 1/1:0,32:32:95:944,95,0
chr1 897325 . G C 2278.06 PASS AC=2;AF=1;AN=2;DP=73;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.74;QD=33.5;SOR=2.105 GT:AD:DP:GQ:PL 1/1:0,68:68:99:2292,204,0
chr1 897564 . T C 681.06 PASS AC=2;AF=1;AN=2;DP=27;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.86;QD=28.38;SOR=1.27 GT:AD:DP:GQ:PL 1/1:0,24:24:72:695,72,0
chr1 909238 . G C 220.64 PASS AC=1;AF=0.5;AN=2;BaseQRankSum=-1.039;DP=12;ExcessHet=0;FS=0;MLEAC=1;MLEAF=0.5;MQ=41.84;MQRankSum=-0.431;QD=
18.39;ReadPosRankSum=-0.165;SOR=1.609 GT:AD:DP:GQ:PL 0/1:4,8:12:89:228,0,89
chr1 909768 . A G 1079.06 PASS AC=2;AF=1;AN=2;DP=36;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=42;QD=32.7;SOR=2.833 GT:AD:DP:GQ:PL 1/1:0,33:33:99:1093,99,0
chr1 914876 . T C 411.06 PASS AC=2;AF=1;AN=2;DP=13;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.7;QD=29.56;SOR=2.494 GT:AD:DP:GQ:PL 1/1:0,11:11:33:425,33,0
chr1 914940 . T C 261.64 PASS AC=1;AF=0.5;AN=2;BaseQRankSum=-0.808;DP=30;ExcessHet=0;FS=3.453;MLEAC=1;MLEAF=0.5;MQ=41.87;MQRankSum=-1.548;QD=
9.02;ReadPosRankSum=-0.988;SOR=1.609 GT:AD:DP:GQ:PL 0/1:16,13:29:99:269,0,333
chr1 915227 . A G 626.06 MQ_filter AC=2;AF=1;AN=2;DP=22;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=39.79;QD=28.46;SOR=1.609 GT:AD:DP:GQ:PL 1/1:0,22:22:66:640,66,0
chr1 916549 . A G 248.64 PASS AC=1;AF=0.5;AN=2;BaseQRankSum=1.053;DP=18;ExcessHet=0;FS=0;MLEAC=1;MLEAF=0.5;MQ=41.89;MQRankSum=-0.717;QD=14.63;ReadPosRankSum=
0.44;SOR=0.33 GT:AD:DP:GQ:PL 0/1:7,10:17:99:256,0,178
```

Snapshot of results obtained from normalizing VCF files using bcftools

## 7.1 Annotation of variants

## Command

### Variant annotation

```
$ java -jar snpEff.jar GRChXX.86 SRRID_vcf_file.vcf > SRRID_annotated_vcf_file.vcf $
java -jar snpEff.jar -v <snpEff_db> SRRID_filtered_snps_final.vcf >
SRRID_filtered_snps_final.ann.vcf
```

```
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT 20
chr1 876490 . A G 308.06 PASS AC=2;AF=1;AN=2;DP=13;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=42;QD=28.01;SOR=3.442 GT:AD:DP:GQ:PL 1/1:0,11:11:32:322,32,0
chr1 877831 . T C 375.06 PASS AC=2;AF=1;AN=2;DP=15;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.21;QD=25;SOR=1.112 GT:AD:DP:GQ:PL 1/1:0,15:15:44:389,44,0
chr1 881627 . G A 570.06 PASS AC=2;AF=1;AN=2;DP=21;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=40.97;QD=27.15;SOR=1.508 GT:AD:DP:GQ:PL 1/1:0,21:21:63:584,63,0
chr1 883625 . A G 428.06 PASS AC=2;AF=1;AN=2;DP=17;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=42;QD=28.54;SOR=3.126 GT:AD:DP:GQ:PL 1/1:0,15:15:44:442,44,0
chr1 888639 . T C 650.06 PASS AC=2;AF=1;AN=2;DP=23;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=40.4;QD=30.96;SOR=0.99 GT:AD:DP:GQ:PL 1/1:0,23:23:62:664,62,0
chr1 888659 . T C 365.06 MQ_Filter AC=2;AF=1;AN=2;DP=13;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=39.45;QD=33.19;SOR=2.494 GT:AD:DP:GQ:PL 1/1:0,11:11:33:379,33,0
chr1 889158 . G C 751.06 PASS AC=2;AF=1;AN=2;DP=19;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=40.16;QD=25;SOR=1.371 GT:AD:DP:GQ:PL 1/1:0,17:17:51:765,51,0
chr1 889159 . A C 751.06 PASS AC=2;AF=1;AN=2;DP=19;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=40.16;QD=25;SOR=1.371 GT:AD:DP:GQ:PL 1/1:0,17:17:51:765,51,0
chr1 894573 . G A 930.06 PASS AC=2;AF=1;AN=2;DP=36;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.5;QD=29.06;SOR=2.765 GT:AD:DP:GQ:PL 1/1:0,32:32:95:944,95,0
chr1 897325 . G C 2278.06 PASS AC=2;AF=1;AN=2;DP=73;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.74;QD=33.5;SOR=2.105 GT:AD:DP:GQ:PL 1/1:0,68:68:99:2292,204,0
chr1 897564 . T C 681.06 PASS AC=2;AF=1;AN=2;DP=27;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.06;QD=28.38;SOR=1.27 GT:AD:DP:GQ:PL 1/1:0,24:24:72:695,72,0
chr1 909238 . G C 220.64 PASS AC=1;AF=0.5;AN=2;BaseQRankSum=1.039;DP=12;ExcessHet=0;FS=0;MLEAC=1;MLEAF=0.5;MQ=41.84;MQRankSum=-0.431;QD=
18.39;ReadPosRankSum=-0.165;SOR=1.609 GT:AD:DP:GQ:PL 0/1:4,8:12:89:228,0,89
chr1 909768 . A G 1079.06 PASS AC=2;AF=1;AN=2;DP=36;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=42;QD=32.7;SOR=2.833 GT:AD:DP:GQ:PL 1/1:0,33:33:99:1093,99,0
chr1 914876 . T C 411.06 PASS AC=2;AF=1;AN=2;DP=13;ExcessHet=0;FS=0;MLEAC=2;MLEAF=1;MQ=41.7;QD=29.56;SOR=2.494 GT:AD:DP:GQ:PL 1/1:0,11:11:33:425,33,0
chr1 914940 . T C 261.64 PASS AC=1;AF=0.5;AN=2;BaseQRankSum=-0.808;DP=30;ExcessHet=0;FS=3.453;MLEAC=1;MLEAF=0.5;MQ=41.87;MQRankSum=-1.548;QD=
9.02;ReadPosRankSum=-0.988;SOR=1.609 GT:AD:DP:GQ:PL 0/1:1,16:13:29:99:269,0,333
```

Results obtained from snpEFF for variant annotation, using filtered SNPs as the input

## 7.2 Post-processing of variants using wANNOVAR and VEP

Variant post processing was performed using two tools: wANNOVAR and VEP

### Step 1- wANNOVAR:

This tool is accessible from: <https://wannovar.wglab.org/>.

### Basic Information

Email

Sample Identifier

Input File

+ Input File

or Paste Variant Calls

paste your variant call here

Submit







Reset

Monitor Progress

☒ I agree to the [Terms of Use](#) . Please note that commercial users would need to obtain a license.

Input information to be filled in wANNOVAR to obtain the post-processing results using

## Parameter Settings

Result duration	1 day	
Reference Genome	hg19	
Input Format	VCF	
Gene Definition	RefSeq Gene	
Individual analysis	Individual analysis	
Disease Model	none	

## The default settings for running wANNOVAR

Some expected results for wANNOVAR are provided below:

Chr	Start	End	Ref	Alt	Func.refG	Gene.refG	GeneDetail	ExonicFunc	AAChange	1000G_AF	1000G_AL	1000G_AF	1000G_AL	1000G_EA	1000G_EU	1000G_SA	ExAC_Freq	ExAC_AFR	ExAC_AMF
chr1	877831	877831	T	C	exonic	SAMD11		nonsynony	SAMD11:N	1	1	1	1	1	1	1	0.9999	1	1
chr1	881627	881627	G	A	exonic	NOC2L		synonymo	NOC2L:NA	0.44	0.064	0.44	0.62	0.63	0.57	0.5653	0.1397	0.484	
chr1	888639	888639	T	C	exonic	NOC2L		synonymo	NOC2L:NA	0.92	0.91	0.92	0.92	0.95	0.92	0.9356	0.9096	0.9558	
chr1	888659	888659	T	C	exonic	NOC2L		nonsynony	NOC2L:NA	0.92	0.91	0.92	0.92	0.95	0.92	0.9355	0.9096	0.9557	
chr1	897325	897325	G	C	exonic	KLHL17		synonymo	KLHL17:N	0.86	0.7	0.88	0.92	0.94	0.92	0.9077	0.721	0.9401	
chr1	909238	909238	G	C	exonic	PLEKHN1		nonsynony	PLEKHN1:f	0.78	0.82	0.69	0.89	0.62	0.83	0.6685	0.789	0.6912	
chr1	914876	914876	T	C	exonic	PERM1		nonsynony	PERM1:NN	0.97	0.94	0.95	1	0.96	0.98	0.9664	0.957	0.9554	
chr1	914940	914940	T	C	exonic	PERM1		synonymo	PERM1:NN	0.51	0.18	0.61	0.71	0.59	0.6	0.5874	0.2698	0.584	
chr1	915227	915227	A	G	exonic	PERM1		synonymo	PERM1:NN	0.92	0.76	0.94	1	0.96	0.98	0.9562	0.7839	0.9456	
chr1	916549	916549	A	G	exonic	PERM1		nonsynony	PERM1:NN	0.76	0.6	0.78	0.88	0.76	0.83	0.7836	0.6075	0.8048	
chr1	935222	935222	C	A	exonic	HES4		nonsynony	HES4:NM	0.49	0.036	0.63	0.77	0.59	0.63	0.6611	0.2017	0.7607	
chr1	949608	949608	G	A	exonic	ISG15		nonsynony	ISG15:NM	0.34	0.44	0.27	0.18	0.39	0.36	0.3702	0.4111	0.2454	
chr1	949654	949654	A	G	exonic	ISG15		synonymo	ISG15:NM	0.83	0.51	0.92	0.93	0.95	0.96	0.9122	0.5636	0.9528	
chr1	981931	981931	A	G	exonic	AGRN		synonymo	AGRN:NM	0.8	0.46	0.89	0.96	0.91	0.91	0.8767	0.513	0.9312	
chr1	982994	982994	T	C	exonic	AGRN		synonymo	AGRN:NM	0.84	0.52	0.9	1	0.92	0.97	0.8989	0.5354	0.942	
chr1	984971	984971	G	A	exonic	AGRN		nonsynony	AGRN:NM	0.0044	0.0023	0.0058	0.001	0.014		0.0127	0.0034	0.016	


COSMIC_ID	COSMIC_ClinVar_Si	ClinVar_Di	ClinVar_ID	ClinVar_Di	GWAS_DIS	GWAS_OR	GWAS_BE	GWAS_PU	GWAS_SN	GWAS_P	SIFT_score	SIFT_conv	SIFT_pred	Polyphen2	Polyphen2	Polyphen2
COSM414:2(thyroid);	.	.	.	.	.	.	.	.	.	.	1	0.01	T	0	0.026	B
COSM134:1(thyroid);	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
COSM414:1(thyroid)	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
COSM414:1(thyroid)	.	.	.	.	.	.	.	.	.	.	1	0.01	T	0	0.026	B
COSM399:1(haematc.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
COSM375:1(large_int.	.	.	.	.	.	.	.	.	.	.	0.195	0.244	T	0	0.026	B
COSM459:22(upper_.	.	.	.	.	.	.	.	.	.	.	0.236	0.179	T	0.001	0.067	B
COSM459:1(upper_a.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
COSM459:1(upper_a.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
COSM374:2(thyroid);	.	.	.	.	.	.	.	.	.	.	0.959	0.021	T	0.011	0.147	B
COSM375:1(thyroid); Benign	not_specif	RCV00045	MedGen	CN169374	.	.	.	.	.	.	0.311	0.139	T	0.01	0.144	B
COSM377:1(kidney)	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
COSM112:1(prostate Benign	not_specif	RCV00011	MedGen	CN169374	.	.	.	.	.	.	.	.	.	.	.	.
COSM414:1(thyroid) Benign	not_specif	RCV00011	MedGen	CN169374	.	.	.	.	.	.	.	.	.	.	.	.
.	other	not_specif	RCV00011	MedGen	CN169374	.	.	.	.	.	0.594	0.057	T	0	0.026	B
COSM561:1(breast) Benign	not_specif	RCV00011	MedGen	CN169374	.	.	.	.	.	.	.	.	.	.	.	.
COSM414:1(thyroid)	.	.	.	.	.	.	.	.	.	.	0.3	0.145	T	.	.	.
COSM441:1(haematc.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
COSM375:1(thyroid);	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

gnomAD_g	gnomAD_g	gnomAD_g	gnomAD_g	gnomAD_g	gnomAD_g	Otherinfo	Otherinfo	Otherinfo	Otherinfo	Otherinfo	Otherinfo	Otherinfo	Otherinfo	Otherinfo	Otherinfo	Otherinfo	Otherinfo
1	1	1	1	1	1	hom	375.06	15 chr1	877831	.	T	C	375.06	PASS	AC=2;AF=1 GT:AD:DP: 1/1;0,15:15		
0.4844	0.5265	0.6751	0.6175	0.6336	0.5855	hom	570.06	21 chr1	881627	.	G	A	570.06	PASS	AC=2;AF=1 GT:AD:DP: 1/1;0,21:21		
0.9487	0.9238	0.91	0.9599	0.9456	0.9481	hom	650.06	21 chr1	888639	.	T	C	650.06	PASS	AC=2;AF=1 GT:AD:DP: 1/1;0,21:21		
0.9487	0.9238	0.91	0.9599	0.9455	0.948	hom	365.06	11 chr1	888659	.	T	C	365.06	MQ_filter	AC=2;AF=1 GT:AD:DP: 1/1;0,11:11		
0.926	0.904	0.9035	0.9382	0.9325	0.9325	hom	2278.06	68 chr1	897325	.	G	C	2278.06	PASS	AC=2;AF=1 GT:AD:DP: 1/1;0,68:68		
0.6909	0.6689	0.867	0.5611	0.5739	0.6066	het	220.64	12 chr1	909238	.	G	C	220.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;4,8:12		
0.9618	0.9338	1	0.9906	0.9723	0.9672	hom	411.06	11 chr1	914876	.	T	C	411.06	PASS	AC=2;AF=1 GT:AD:DP: 1/1;0,11:11		
0.6372	0.5331	0.729	0.5281	0.5637	0.558	het	261.64	29 chr1	914940	.	T	C	261.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;16,13:2		
0.957	0.9338	0.9994	0.9906	0.9716	0.9643	hom	626.06	22 chr1	915227	.	A	G	626.06	MQ_filter	AC=2;AF=1 GT:AD:DP: 1/1;0,22:22		
0.8079	0.7133	0.86	0.7364	0.7398	0.7316	het	248.64	17 chr1	916549	.	A	G	248.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;7,10:17		
0.6695	0.6267	0.7621	0.5362	0.5659	0.5557	het	404.64	27 chr1	935222	.	C	A	404.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;10,17:2		
0.2673	0.3245	0.2111	0.4708	0.4094	0.4071	het	634.64	50 chr1	949608	.	G	A	634.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;24,26:5		
0.9308	0.9139	0.9216	0.9519	0.9528	0.9357	hom	3334.06	103 chr1	949654	.	A	G	3334.06	PASS	AC=2;AF=1 GT:AD:DP: 1/1;0,103:1		
0.9093	0.8841	0.9654	0.9337	0.9223	0.8979	hom	332.06	12 chr1	981931	.	A	G	332.06	MQ_filter	AC=2;AF=1 GT:AD:DP: 1/1;0,12:12		
0.9248	0.9272	0.9981	0.9599	0.9313	0.9226	hom	2230.06	73 chr1	982994	.	T	C	2230.06	PASS	AC=2;AF=1 GT:AD:DP: 1/1;0,73:73		
0.006	0.0298	0	0.0043	0.0094	0.0072	het	142.64	12 chr1	984971	.	G	A	142.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;6,6:12		
0.7733	0.6258	0.7937	0.6447	0.6833	0.6766	het	472.64	32 chr1	990280	.	C	T	472.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;13,19:3		
0.7068	0.5667	0.7646	0.519	0.5505	0.535	het	161.64	13 chr1	1007432	.	G	A	161.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;6,7:13		
0.1122	0.2616	0.0914	0.2347	0.1704	0.1888	het	435.64	31 chr1	1021346	.	A	G	435.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;13,18:3		
0.9379	0.8278	1	0.8543	0.8957	0.9134	hom	1537.06	54 chr1	1158631	.	A	G	1537.06	PASS	AC=2;AF=1 GT:AD:DP: 1/1;0,54:54		
0	0	0	0.0003	0.0001	0.001	het	310.64	24 chr1	1233467	.	G	A	310.64	PASS	AC=1;AF=C GT:AD:DP: 0/1;9,15:24		
.	.	.	.	.	.	het	34.64	19 chr1	1235982	.	C	T	34.64	QD_filter	AC=1;AF=C GT:AD:DP: 0/1;15,4:15		


## Snapshots of expected wANNOVAR results

### Step 2- Variant Effect Predictor (VEP)

The VEP tool is accessible from: <https://asia.ensembl.org/Tools/VEP>

**Variant Effect Predictor** 

**New job**

Species: Homo\_sapiens   
 Assembly: GRCh38.p13  
[Change species](#)  
 If you are looking for VEP for Human GRCh37, please go to [GRCh37 website](#).

Name for this job (optional):

Input data:

Either paste data:

Examples: [Ensembl default](#), [VCF](#), [Variant identifiers](#), [HGVS notations](#), [SPDI](#)

Or upload file:  No file chosen


Or provide file URL:


## Input data settings for VEP


Transcript database to use:


- ☒ Ensembl/GENCODE transcripts
- ☐ Ensembl/GENCODE basic transcripts
- ☐ RefSeq transcripts
- ☐ Ensembl/GENCODE and RefSeq transcripts


Additional configurations:


**Identifiers**  Additional identifiers for genes, transcripts and variants

**Variants and frequency data**  Co-located variants and frequency data

**Additional annotations**  Additional transcript, protein and regulatory annotations

**Predictions**  Variant predictions, e.g. SIFT, PolyPhen

**Filtering options**  Pre-filter results by frequency or consequence type

**Advanced options**  Additional enhancements

## Additional configurations and the type of transcript database to be used for VEP analysis

#Uploads	Location	Allele	Consequence	IMPACT	SYMBOL	Gene	Feature_t	Feature	BIOTYPE	EXON	INTRON	HGVSc	HGVSp	cDNA_pos	CDS_pos	Protein_pos	Amino_aci	Codons	Existing_v	DISTANCE	STRAND	FLAGS	SY
1:876499:-G			downstream MODIFIER NOC2L		ENSG000001	Transcript	ENST000001	protein_cc	-	-	-	-	-	-	-	-	-	-	rs4372192	3085	-1	-	HC
1:876499:-G			intron_var MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	protein_cc	-	-	05-Nov	-	-	-	-	-	-	-	rs4372192	-	-	-	1 cds_start_HC
1:876499:-G			downstream MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	protein_cc	-	-	Jul-13	-	-	-	-	-	-	-	rs4372192	-	-	-	1 - HC
1:876499:-G			downstream MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	protein_cc	-	-	-	-	-	-	-	-	-	-	rs4372192	1828	-1	-	1 cds_end_HC
1:876499:-G			intron_var MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	protein_cc	-	-	01-Jun	-	-	-	-	-	-	-	rs4372192	-	-	-	1 cds_start_HC
1:876499:-G			upstream MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	retained_j	-	-	-	-	-	-	-	-	-	-	rs4372192	1047	-1	-	HC
1:876499:-G			upstream MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	retained_j	-	-	-	-	-	-	-	-	-	-	rs4372192	984	-1	-	HC
1:876499:-G			non_coding MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	retained_j	-	-	01-Apr	-	-	-	44	-	-	-	rs4372192	-	-	-	1 - HC
1:876499:-G			downstream MODIFIER NOC2L		ENSG000001	Transcript	ENST000001	retained_j	-	-	-	-	-	-	-	-	-	-	rs4372192	3086	-1	-	HC
1:876499:-G			intron_var MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	processed	-	-	01-Feb	-	-	-	-	-	-	-	rs4372192	-	-	-	1 - HC
1:876499:-G			downstream MODIFIER NOC2L		ENSG000001	Transcript	ENST000001	retained_j	-	-	-	-	-	-	-	-	-	-	rs4372192	3085	-1	-	HC
1:876499:-G			downstream MODIFIER NOC2L		ENSG000001	Transcript	ENST000001	processed	-	-	-	-	-	-	-	-	-	-	rs4372192	4200	-1	-	HC
1:876499:-G			regulatory MODIFIER		-	-	Regulatory	ENSR000001	promoter	-	-	-	-	-	-	-	-	-	rs4372192	-	-	-	-
1:877831:-C			downstream MODIFIER NOC2L		ENSG000001	Transcript	ENST000001	protein_cc	-	-	-	-	-	-	-	-	-	-	rs6672356	1753	-1	-	HC
1:877831:-C			missense MODERAT SAMD11		ENSG000001	Transcript	ENST000001	protein_cc	-	-	08-Dec	-	-	750	751	251 W/R	Tgg/Cgg	rs6672356	-	-	-	1 cds_start_HC	
1:877831:-C			missense MODERAT SAMD11		ENSG000001	Transcript	ENST000001	protein_cc	-	-	Oct-14	-	-	1110	1027	343 W/R	Tgg/Cgg	rs6672356	-	-	-	1 - HC	
1:877831:-C			downstream MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	protein_cc	-	-	-	-	-	-	-	-	-	-	rs6672356	3160	-1	-	1 cds_end_HC
1:877831:-C			missense MODERAT SAMD11		ENSG000001	Transcript	ENST000001	protein_cc	-	-	04-Jul	-	-	507	508	170 W/R	Tgg/Cgg	rs6672356	-	-	-	1 cds_start_HC	
1:877831:-C			non_coding MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	retained_j	-	-	01-Feb	-	-	286	-	-	-	-	rs6672356	-	-	-	1 - HC
1:877831:-C			non_coding MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	retained_j	-	-	02-Feb	-	-	191	-	-	-	-	rs6672356	-	-	-	1 - HC
1:877831:-C			non_coding MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	retained_j	-	-	03-Apr	-	-	389	-	-	-	-	rs6672356	-	-	-	1 - HC
1:877831:-C			downstream MODIFIER NOC2L		ENSG000001	Transcript	ENST000001	retained_j	-	-	-	-	-	-	-	-	-	-	rs6672356	1754	-1	-	HC
1:877831:-C			downstream MODIFIER SAMD11		ENSG000001	Transcript	ENST000001	processed	-	-	-	-	-	-	-	-	-	-	rs6672356	278	-1	-	HC
1:877831:-C			downstream MODIFIER NOC2L		ENSG000001	Transcript	ENST000001	retained_j	-	-	-	-	-	-	-	-	-	-	rs6672356	1753	-1	-	HC
1:877831:-C			downstream MODIFIER NOC2L		ENSG000001	Transcript	ENST000001	processed	-	-	-	-	-	-	-	-	-	-	rs6672356	2868	-1	-	HC

## Expected result for VEP

## CONCLUSIONS AND FUTURE PERSPECTIVES

8 Colorectal cancers have several diagnostic and treatment options to combat it, however, a delay in disease detection is life-threatening. Therefore, this protocol emphasizes on the identification and analysis of somatic and germline variants for colorectal cancer exome datasets, retrieved from publicly available databases such as NCBI-SRA. These steps describe the various tools required to run this pipeline, with the codes required to run each tool. Moreover, although this protocol is described for colorectal cancer exomes, the same set of codes and steps may be followed for the overall analysis of any cancer exome, when the desired outcome is to obtain and analyze somatic and germline variants separately. This computational pipeline requires further in-vitro and in-vivo studies, however, the outcomes will offer prospects for similar such studies that are crucial for designing a cure, prognosis or a treatment strategy for colorectal cancers, based on the scrutinized variants.

The outcomes from this protocol can be carried forward to building and designing decision support systems using artificial intelligence and machine learning (AI/ML), so that the identified somatic and germline mutants can be used for early colorectal cancer prediction through a prediction model. This will aid clinicians/ researchers/diagnosticians to help take better decisions for treatment strategies.