Supplementary Material for:

VariantMaster is a robust tool for the identification of causative variants from HTS data in familial, denovo and somatic genetic disorders including cancer

Federico A. Santoni[1,2*], Periklis Makrythanasis[1], Sergey Nikolaev[1], Michel Guipponi[2], Daniel Robyr[1], Armand Bottani[2], Stylianos E. Antonarakis[1,2,3]

[1]Department of Genetic Medicine and Development
University of Geneva Medical School,
1 rue Michel Servet, 1211 Geneva 4, Switzerland

[2]Geneva University Hospital - HUG
4 Rue Gabrielle-Perret-Gentil, 1211 Geneva 4, Switzerland

[3]iG3 Institute of Genetics and Genomics of Geneva,
University of Geneva Medical School,
1 rue Michel Servet, 1211 Geneva 4, Switzerland

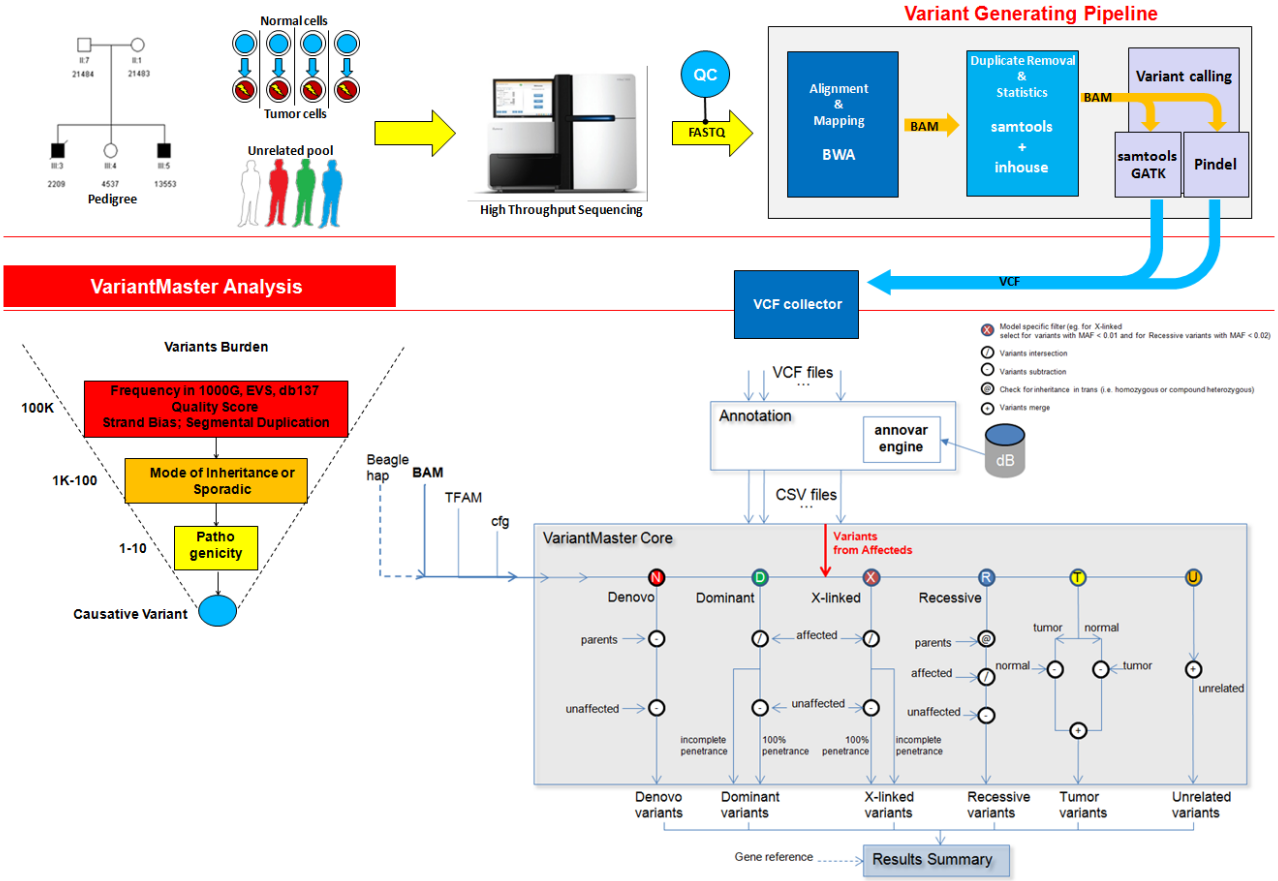
*corresponding author:
federico.santoni@unige.ch
Tel: +41 223795719
Fax: +41 223795706

Contents

1. Variant Generating Pipeline + VariantMaster

The pipeline is organized in three main steps: DNA extraction, exon capture and library preparation; high throughput sequencing; computational processing of row data into the variant generating pipeline. In its basic configuration, the pipeline directly processes fastq files produced by the sequencer. Specifically, the mapping of fastq reads is performed with BWA, with duplicate removal by SAMtools, against the hg19 reference. Then, SNPs and indels are called by bcftools or GATK UnifiedGenotyper and Pindel and collected in a unique VCF file using a custom python script (see Supplementary Fig.1). VCF files are then processed by VariantMaster that, after annotation, proceeds by applying the analysis mode and user defined filter settings.

In other configurations, the pipeline may start from processed BAM files.



Supplementary Figure 1. General schematic showing the integration of VariantMaster with the Variant Generating Pipeline.

2. De novo Analysis of the CEU trio

WGS CEU BAM files:

CEUTrio.0.01.WGS.b37_NA12878_clean.dedup.recal.20120117.bam
CEUTrio.0.01.WGS.b37_NA12891_clean.dedup.recal.20120117.bam
CEUTrio.0.01.WGS.b37_NA12892_clean.dedup.recal.20120117.bam

and WES CEU BAM files:

- CEUTrio.HiSeq.WEx.b37_decoy.NA12878.clean.dedup.recal.20120117.bam
- CEUTrio.HiSeq.WEx.b37_decoy.NA12891.clean.dedup.recal.20120117.bam
- CEUTrio.HiSeq.WEx.b37_decoy.NA12892.clean.dedup.recal.20120117.bam

have been downloaded from ftp://ftptrace.ncbi.nih.gov/1000genomes/ftp/technical/working/20120117_ceu_trio_b37_decoy and processed with the variant calling pipeline. Only SNPs have been retained for this specific analysis.

2.1 GATK

After duplicate removal we apply GATK Unified Genotyper to generate multisamlpe VCF files as follows:

```
java -Xmx30g -jar ~/scratch/GATK/GenomeAnalysisTK.jar
-T UnifiedGenotyper -R human_g1k_v37.fasta
-I CEUTrio.HiSeq.WGS.b37_NA12878_clean.dedup.recal.20120117.bam
-I CEUTrio.-20.WGS.b37_NA12891_clean.dedup.recal.20120117.bam
-I CEUTrio.-20.WGS.b37_NA12892_clean.dedup.recal.20120117.bam
```

2.2 Polymutt and Famseq

Polymutt v.0.14 and Famseq 1.0 were fed with GATK Unified Genotyper calls and run with default parameters.

2.3 VariantMaster

For the CEU de novo analysis, VariantMaster was run with the following configuration settings:

```
CARR_THR := 0.95
HOM_THR := 0.75
annovar := "/usr/bin/annovar/"
annovar_db := "/usr/lib/annovar/annovar_DB"
reference := 'hg19.fa'
Filter_DenovoFilter
select      Func  in    exonic;splicing
select      QS    >=    50 #(up to 1300)
select      Cov   >=    8
End_Filter
```

where we filtered out the variants with QS < [50 - 1300] and coverage (in the proband) <8.

2.4 KGGSeq

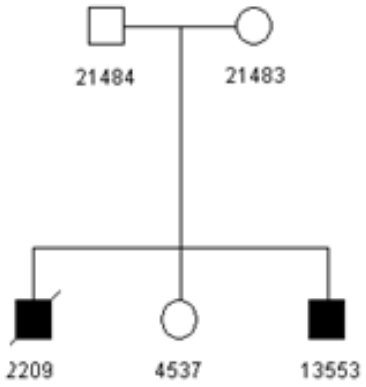
KGGSeq version 0.3 was run over GATK Unified Genotyper calls and Polymutt or Famseq modified VCF files, accordingly to the indications reported in http://statgenpro.psychiatry.hku.hk/limx/kggseq/doc/Denovo.htm with the same parameters we used for VariantMaster analysis:

```
java -jar ./kggseq.jar --buildver hg19 --vcf-file CEU_trio.vcf --ped-file CEU.tfam --pedb-filter hg19 1kg201204, hg19 dbsnp137, hg19 ESP6500AA, hg19_ESP6500EA --genotype-filter 7 --no-resource-check 0 --no-lib-check --db-gene refgene --gene-feature-in 0,1,2,3,4,5" --gty-dp 8 --seq-qual 50 --gty-af-alt 0.75 --seq-sb -10
```

3. Pedigree analysis on a real Familial Case

Exome capture of all members of the familial case (Supplementary Figure S2) was conducted using the SureSelect Human ALL Exon v3 or V4 kit (Agilent Technologies) according to manufacturer's recommendations. Three different genomic libraries were pooled and sequenced in one lane of an Illumina HiSeq2000 sequencer using a 2x95bp paired end indexing protocol.

Fastq files have been processed according to what previously mentioned to produce a multisample VCF file.



Supplementary Figure 2. Pedigree of a family presenting with a Mendelian disorder characterized by mental retardation, ataxia and epilepsy. All members have been sequenced: two unaffected parents (21484, 21483), two affected brothers (2209, 13553) and one unaffected sister. One of the two affected (2209) died at the age of 5.

3.1 VariantMaster

For this analysis, VariantMaster has been run with the X-linked and the Recessive model with the following parameters:

```
CARR_THR := 0.95
HOM_THR := 0.75
annovar := "/usr/bin/annovar/"
annovar_db := "/usr/lib/annovar/annovar_DB"
reference := 'hg19.fa'
Filter_RecessiveFilter
conditional_select      INFO    in      PINDEL  QS      >=      600
conditional_select      INFO    in      INDEL   QS      >=      200
select  QS      >=      50
select  Func    in      exonic;splicing
select  ExonicFunc      !=      synonymous SNV
remove  segdup  -       -
select  1000g   <=      0.02
select  snp_freq        <=      0.02
select  esp     <=      0.02
```

```
End_Filter
Filter_RecessiveFilter
conditional_select      INFO    in      PINDEL  QS      >=      600
conditional_select      INFO    in      INDEL   QS      >=      200
select  QS      >=      700
select  Func    in      exonic;splicing
select  ExonicFunc      !=      synonymous SNV
remove  segdup  -       -
select  1000g   <=      0.02
select  snp_freq        <=      0.02
select  esp     <=      0.02
End_Filter
```

The results of these analyses are reported in Supplementary Tables S1 and S2, respectively.

Supplementary Table S1. Variants matching the X-linked model

Chr	Start	Func	Gene	ExonicFunc	AA Change	Obs	Ref	Zyg	QS	Cov	Snp_db137	Snp freq	sift	Poly phen	Phylop	lrt	Mut Taster	gerp++
chrX	107782983	exonic; splicing	COL4A5	Nonsynon	p.Y30C	G	A	hom	222	17	rs150305490	2e-4	0.14	0.48	0.99	0.99	0.02	4.56
chrX	107976573	exonic	IRS4	Nonsynon	p.P1001R	C	G	hom	222	53	rs146350531	2e-4	0.3	0.99	0.85	0.68	0.01	1.05
chrX	135115628	exonic	SLC9A6	nonsynon	p.R516Q	A	G	hom	222	12	rs146263125	1e-3	0	0.99	0.99	1	0.999	4.84

Supplementary Table S2. Variant matching the recessive model

Chr	Start	Func	Gene	ExonicFunc	AA Change	Obs	Ref	Zyg	QS	Cov	Snp_db137	Snp freq	sift	PP2	Phy	llrt	mt	gerp++	21484	21483	4537
chr22	40760969	exonic	ADSL	nonsyn SNV	p.R426H	A	G	hom	222	235	rs119450941	0	0	0.7	1	1	1	5.2	het	Het	-

*variant found in homozygosity in one personal genome

3.2 GATK, Polymutt and Famseq

GATK Unified Genotyper Polymutt v.0.14 and Famseq 1.0 were fed with GATK Unified Genotyper calls and run with default parameters.

3.3 KGGSeq

VCF file was prepared as previously mentioned. In order to identify the causative mutation in a Recessive+X-linked model of inheritance, KGGSeq was run with the following parameters:

```
java -jar kggseq.jar --buildver hg19 --vcf-file Disease_Fam.vcf --ped-file
Disease_Fam.tfam      --pedb-filter      hg19_1kg201204,      hg19_dbsnp137,
hg19_ESP6500AA,hg19_ESP6500EA     --rare-allele-freq  0.02  --genotype-filter
1,2,6 --no-resource-check 0 --no-lib-check --gene-feature-in 0,1,2,3,4,5 --
db-gene refgene --seq-qual 50 --seq-mq 20 --seq-sb -10 --gty-qual 20 --gty-dp
8 --db-score dbnsfp --mendel-causing-predict all
```

Notably, adding genotype filters 4, 5 or 6 decreases the number of reported variants but it also removes the causative one from the results.

4. Identification of somatic variants in matched tumor-germline samples

Aligned reads (BAM) of 8 paired tumor samples + germline were downloaded from https://tcga-data.nci.nih.gov/ and the respective annotations were obtained from COSMIC (http://cancer.sanger.ac.uk/cosmic/). Note that the reference genome for this dataset is hg18.

4.1 VariantMaster

Variants have been extracted as described (Supplementary Figure S1). Functional variants from each tumor sample (nonsense, missense, frameshift and splicing) with coverage >3, QS > 15 and no constraints on frequency were filtered in by VariantMaster and considered as putative somatic mutations if not found in the related germline sample. VariantMaster was launched with the following configuration:

```
CARR_THR := .95
HOM_THR := 0.75
COV := 3
reference := 'hg18.fa'
Filter_TumorFilter
conditional_select      INFO    in      PINDEL  QS      >=      600
conditional_select      INFO    in      INDEL   QS      >=      100
select  Func    in      exonic;splicing
select  QS      >=      15
select  ExonicFunc      !=      synonymous SNV
End_Filter
```

4.2 VarScan

VarScan expects both a normal and a tumor file in SAMtools pileup format from sequence alignments in binary alignment/map (BAM) format that we obtained after the removal of duplicated reads.

As recommended in http://VarScan.sourceforge.net/somatic-calling.html, we launched VarScan applying the following format to each TCGA tumor-germline pair:

```
java -jar VarScan.v2.2.jar somatic normal.pileup tumor.pileup output
```

5. VariantMaster - User Manual

5.1 Download and Installation

VariantMaster is a python package delivered as a binary executable program for Linux based systems. Source code could be provided upon request to the author.

DISCLAIMER

Neither the University of Geneva nor the author assume any responsibility whatsoever for its use by other parties and makes no guarantees, expressed or implied, about its quality, reliability, or any other characteristics in particular for the use of this tool in clinical contexts.

By downloading the software from this page, you agree to the specified terms.

Download:

http://sourceforge.net/projects/variantmaster/

To install:

```
> gunzip VariantMaster-1.0.gz
> VariantMaster-1.0 -h
Usage: VariantMaster -c <cfg_file> -t <tfam_file> -o <output_prefix> [-s <csv_folder>
|-v <vcf_folder>] -b <bam_folder> -H <hap_file [optional]> <options>

Options:
  -h, --help            show this help message and exit
  -v VCF, --vcf=VCF     Path to vcf folder or to a multisample vcf file
  -s CSV, --csv=CSV     Path to csv folder
  -b BAM, --bam=BAM     Path to bam folder
  -c CONFIGURATION, --configuration=CONFIGURATION
                        Configuration file
  -o OUTPUT, --output=OUTPUT
                        Output filename prefix
  -t TFAM, --tfam=TFAM  pathway to pedigree structure (tfam format)
  -H HAPLOTYPE, --haplotype=HAPLOTYPE
                        Path to the Haplotype file (tped format)
  -R, --report          Only the report please!
  -A, --annotate        Annotate only!
```

VariantMaster uses Annovar to annotate VCF files. Last Annovar release and accompanying databases can be downloaded from http://www.openbioinformatics.org/annovar/.

5.2 Configure the input data

5.2.1 Creating the workspace

VariantMaster requires several input files depending on what is available for the analysis. To perform the pedigree analysis it needs:

1a. one folder (e.g. named "VCF") containing a file VCF (Variant Call Format) for at least one affected individual OR a multisample VCF;

OR

1b. one folder (e.g. "CSV") containing a file CSV for at least one affected individual (see 2.5 to adapt annotated text files to VariantMaster format);

2. (optional) one folder (e.g." HAP") containing HAP files (haplotypes in TPED format as produced by Beagle - see 2.4);

3. one TFAM file (PLINK format - see 2.3);

4. one folder (e.g. BAM) containing the BAM files (Binary Aligned Mapped reads) and the BAI (BAM Index) files for all sequenced individual - it's not mandatory but considerably improves the accuracy of the algorithm;

5. one configuration file (see 2.2).

For matched Tumor-Germline analysis or for Unrelated individuals, VariantMaster requires:

1. one folder containing VCF or CSV files from samples
2. one TFAM file describing the pairs Germline - Tumor
3. one folder containing the BAM and BAI files from all samples
4. one configuration file

If all individuals have been sequenced, it is strongly advised to prepare a workspace with the following setup:

```
CSV/ (or VCF/)      BAM/   study.cfg     families.tfam
```

where CSV (or VCF) is prepared as:

```
> mkdir CSV
> ln -s  <path_to_csv_files>  CSV/
```

Analogously, BAM is created as:

```
> mkdir BAM
> ln -s  <path_to_bam_files>  BAM/
```

If only few individuals were sequenced but all haplotypes are available (see 2.4 to create haplotypes from genotyped data) then the workspace would be similar to:

```
CSV/ (or VCF/)      BAM/   HAP/   study.cfg families.tfam
```

where HAP is created with:

```
> mkdir HAP
> ln -s  <path_to_csv_files>  HAP/
```

It is worth to specify again that the usage of CSV or VCF is mutually exclusive and the HAP folder is optional.

The optional switches -A and -R solely allow to annotate VCF files (the CSV folder is produced) and to recalculate the final report, respectively.

IMPORTANT:

All file names (CSV, VCF, BAM or HAP) associated to an individual must contain the string *_ID_* where *ID* is the identifier of the individual as reported in the TFAM file (2.3).

- in a multisample VCF file, the name of each sample must be the related ID

5.2.2 Write the Configuration file

The configuration file contains some general settings.

CARR_THR is the threshold probability to consider a sample as a carrier of a given SNV (0.95 is the default),

HOM_THR allelic percentage to consider a variant as homozygous (75% is the default),

reference full path of the genomic fasta (and respective .fai) reference file to be used for INDEL processing.

gene_reference full path of the gene reference in the UCSC format. It is used for the final report.

annovar path to annovar scripts.

annovar_db path to annovar databases.

Variables are defined by the syntax

```
Var := value
```

Each analysis module has its specific filter and it is introduced by the construct:

```
Filter_NamemoduleFilter
..
END_Filter
```

Following modules are available: `DenovoFilter`, `DominantFilter`, `XLinkedFilter`, `RecessiveFilter`, `TumorFilter`, `IndependentFilter`.

IMPORTANT

Specific filtering operations are executed through `select`, `conditional_select` and `remove`. These commands act over the fields specified in the header of the annotated CSV file.

`select Field Operator Value`

select for the variants having a value in the specified Field satisfying the condition defined by Operator (one among >,>=,<,<=, == ,!=,in) and Value.

example:

```
select  Func    in      exonic;splicing
```

select for variants having exonic, splicing or exonic;splicing in the field Func

---

`conditional_select Field Operator Value Field Operator Value`

execute a select defined by the second triplet only if the first triplet is satisfied

example:

```
conditional_select      INFO    in      PINDEL  QS      >=      600
```

select for PINDEL variants having quality score >=600

`remove  Field  -  -`

will remove all variants with whatever value in Field

example:

```
remove  segdup  -       -
```

Here is an example of a configuration file for a Denovo filter:

```
#General variables
CARR_THR := 0.9
HOM_THR := 0.6
reference := '/data/hg19/hg19.unmasked.fa'
```

```
annovar := '/usr/local/annovar'
annovar_db := '/usr/local/annovar_db'
#Filter definitions
Filter_DenovoFilter
conditional_select      INFO    in      PINDEL  QS      >=      600
conditional_select      INFO    in      INDEL   QS      >=      200
select  Func    in      exonic;splicing
select  QS      >=      50
select  ExonicFunc      !=      synonymous SNV
select  Zyg     ==      het
remove  segdup  -       -
select  1000g   <=      0.01
select  snp137_freq        <=      0.01
select  esp6500     <=      0.01
End_Filter
```

IMPORTANT

Commands, operators and values are TAB separated!!!

5.2.3 Structure of the Pedigree (TFAM) file

The TFAM format, as defined in PLINK (http://pngu.mgh.harvard.edu/~purcell/plink/), is

```
Family_ID Individual_ID Father_ID Mother_ID Sex Phenotype
```

where sex is (0 - male, 1 - female) and Phenotype can be (1 - unaffected, 2 - affected)

For example, two nuclear families with unaffected parents and, respectively, one affected boy and two children, one affected girl and one unaffected boy may be represented as:

```
Fam001 Fath_001 0 0 0 1
Fam001 Moth_001 0 0 1 1
Fam001 Boy_001 Fath_001 Moth_001 0 2
Fam002 Fath_002 0 0 0 1
Fam002 Moth_002 0 0 1 1
Fam002 Boy_002 Fath_002 Moth_002 0 1
Fam002 Girl_002 Fath_002 Moth_002 1 2
```

IMPORTANT

All files (CSV, VCF, HAP or BAM) related to a specific individual ID must be named as *whatever_ID_somethingelse*. In other words:

- in a multisample VCF file, the name of each sample must be ID
- the ID reported in the TFAM file has to be repeated in filenames between underscores (_ID_)

- a CSV file related to *Fath001* should be named *XYZ_Fath001_KJH.csv* (the extension *.csv* is mandatory).

5.3 Creating haplotypes from genotyped data

VariantMaster accepts haplotypes in the form of TPED files. TPED format is:

The easiest way to generate haplotypes compatible with VariantMaster from genotyped data is to download Beagle (http://faculty.washington.edu/browning/beagle/beagle.jar) and the script

http://seaseq.unige.ch/~fsantoni/VariantMaster/tools/ped2beagle2ped.py.

First, we start from genotyping data in the transposed PLINK file format TPED *(chr, snpname, centimorgan, position, [genotypes]* see *http://pngu.mgh.harvard.edu/~purcell/plink/data.shtml*).

After having installed Beagle,

```
> ped2beagle2ped.py -h

Usage: ped2beagle2ped.py -c "beagle_cmd"
Options:
  -h, --help            show this help message and exit
  -c CMD, --cmd=CMD     java -Xmx2048m -jar path_to_beagle.jar
                        <trio/phased/unphased>=%s out=<output_prefix>
                        missing=?
  -t TPED, --tped=TPED  tped file
```

by applying the script to the file *Fam01.tped* (unphased genotypes)

```
> ped2beagle2ped.py -c "java -Xmx2048m -jar /usr/local/beagle.jar unphased=%s
out=example1 missing=? -t Fam01.tped"
```

we get as output *example1.Fam01.tped.beagle.phased.* Note that here beagle.jar is in */usr/local*.

```
1 rs10492937 3339339 C C C C C C C C C C C C
1 rs4648500 3340855 C C C C C C C C C C C C
1 rs2244013 3342530 G G G G G G G G G G G G
1 rs870171 3342804 C C C C C C C C C C C C
1 rs2493272 3349513 A A A A A A A A A A A A
1 rs1537406 3352227 G G G G G G G G G G G G
1 rs2236518 3352872 A A A A A A A A A A A A
```

The format (*chr, snpname,position, [genotypes]*) describes a VariantMaster compliant haplotype.

5.4 Convert any annotated variant file to be VariantMaster compliant

It is relatively easy to convert structured text files to be VariantMaster CSV compliant.

The only requirements are:

a. all files need to have the same header so they can be addressed by the filtering system;
b. first three columns must be **Chr, Start, End;**
c. additional **mandatory fields** are**: Zyg ={hom,het}; INFO** beginning with **INDEL** if the variant is an insertion/deletion and containing **DP4** as in the VCF format.
d. fields are comma separated (CSV).

For example, it is possible to use

```
awk -F, -v OFS="," '{print $4,$5,$6,$0}' file
```

to rearrange the following file

```
Gene;Function;ExonicFunction;Chr;Start;End;Zyg;dbSNP;esp6500;INFO
FOX3;exonic;non synonymous;chr17;77090533;77090533;hom;rs17344067;0;DP4=1,1,34,56
…
```

IMPORTANT

Avoid spaces in field names

5.5 Running the program

5.5.1 Example: Nuclear family

The family **fam01** is composed by unaffected father F, mother M, daughter D and one affected son (S). all members have been been sequenced. The multisample VCF file with samples *_F[M,D,S]_* is available and, accordingly, BAM files are named *_ F[M,D,S]_fam01.bam*.
The annotated CSV files are generated by VariantMaster in the folder *./masterCSV* with names *seq_F[M,D,S]_fam01.csv.*
The header of all CSV files is:

```
Chr,Start,End,Func,ExonicFunc,QS,Zyg,segdup,1000g,dbSNP,snp_freq,esp,...
```

Therefore, the family structure is (*fam01.tfam*):

```
fam01 F 0 0 0 1
fam01 M 0 0 1 1
```

```
fam01 S F M 0 2
fam01 D F M 1 1
```

The analysis is executed on the denovo, recessive and X linked model. Accordingly, the configuration file is (fam01.cfg - REMEMBER - it is a **TAB delimited file**):

```
#General variables
CARR_THR := 0.95
HOM_THR := 0.75
reference := '/data/hg19/hg19.unmasked.fa'
#Filter definitions
Filter_DenovoFilter
conditional_select      INFO    in      PINDEL  QS      >=      600
conditional_select      INFO    in      INDEL   QS      >=      200
select  Func    in      exonic;splicing
select  QS      >=      50
select  ExonicFunc      !=      synonymous SNV
select  Zyg     ==      het
remove  segdup  -       -
select  1000g   <=      0.01
select  snp_freq        <=      0.01
select  esp     <=      0.01
End_Filter
Filter_RecessiveFilter
conditional_select      INFO    in      PINDEL  QS      >=      600
conditional_select      INFO    in      INDEL   QS      >=      200
select  Func    in      exonic;splicing
select  QS      >=      50
select  ExonicFunc      !=      synonymous SNV
remove  segdup  -       -
select  1000g   <=      0.01
select  snp_freq        <=      0.01
select  esp     <=      0.01
End_Filter
Filter_XlinkedFilter
conditional_select      INFO    in      PINDEL  QS      >=      600
conditional_select      INFO    in      INDEL   QS      >=      200
select  Func    in      exonic;splicing
select  QS      >=      50
select  ExonicFunc      !=      synonymous SNV
remove  segdup  -       -
select  1000g   <=      0.01
select  snp_freq        <=      0.01
select  esp     <=      0.01
End_Filter
```

Eventually the program runs as:

```
> VariantMaster -c  fam01.cfg  -t fam01.tfam  -o My_output -v VCFfile -b BAM
```

and the logfile should read as:

```
Mon Feb 18 15:20:25 2013 - N. of Families: 1
Mon Feb 18 15:20:25 2013 - Family: fam01
Mon Feb 18 15:20:25 2013 - Loaded Individual F - father 0 - mother 0 - sex M - status unaffected
Mon Feb 18 15:20:25 2013 - Loaded Individual M - father 0 - mother 0 - sex F - status unaffected
Mon Feb 18 15:20:25 2013 - Loaded Individual S - father F - mother M - sex M - status affected
Mon Feb 18 15:20:25 2013 - Loaded Individual D - father F - mother M - sex F - status unaffected
Mon Feb 18 15:20:25 2013 - Loading hg19 reference from /data/hg19/hg19.unmasked.fa
```

VariantMaster generates the folder *My_output* with a report for each requested models. Specifically, these are extensions of the original CSV files where only variants in affected individuals satisfying the respective inheritance model are reported. Columns describing the presence of the mutations in the other family member have been added. For further statistical analysis, VariantMaster produces a summary file in which it reports the mutations found in all the affected individuals, ordered by model and by gene (see §4).

IMPORTANT

VCF files need to be annotated only once. When the folder *masterCSV* is created, use option *-s masterCSV* for further analyses.

5.5.2 Example: Tumor-to-Germline comparison from VCF files

In this example, Tumor-Germline pairs have been taken and sequenced from 10 unrelated individuals presenting the same type of tumor. Unnnotated VCF files are available in the folder *./somepath/VCF* with names *seq_T[G]01_ex[1-10].csv* and, accordingly, BAM files in the format *bamfile_ [T/G][1-10]_ex[1-10].bam*.
Therefore, the structure is (*e.g.tfam*)

```
ex1     T1      0       0       0       2
ex1     G1      T1      0       0       1
ex2     T2      0       0       0       2
ex2     G2      T2      0       0       1
…
ex10    T10     0       0       0       2
ex10    G10     T10     0       0       1
```

where tumor samples have been tag with 2 (-affected) and germ cells sample with 1 (-unaffected).

It is possible to first annotate the variants with Annovar with the switch (-A) and create a new folder (*masterCSV)* with the annotations:

```
>  VariantMaster -c  ex.cfg  -t fam01.tfam  -v VCF -A
```

The configuration file may be written according to entries in the header of newly created csv files in the masterCSV folder (- REMEMBER - it is a **TAB delimited file**):

```
#General variables
CARR_THR := 0.95
HOM_THR := 0.75
reference := '/data/hg19/hg19.unmasked.fa'
gene_reference := '/data/hg19/refGene.txt'
#Filter definitions
Filter_TumorFilter
```

```
conditional_select      INFO    in      PINDEL  QS      >=      300
conditional_select      INFO    in      INDEL   QS      >=      100
select  Func    in      exonic;splicing
select  QS      >=      10
select  ExonicFunc      !=      synonymous SNV
remove  segdup  -       -
End_Filter
```

The analysis is eventually started with:

```
>  VariantMaster -c  ex.cfg  -t fam01.tfam  -o test -s masterCSV -b BAM
```

Log will then read:

```
Tue Feb 19 15:36:30 2013 - N. of Families: 10
Tue Feb 19 15:36:30 2013 - Family: ex1
Tue Feb 19 15:36:30 2013 - Loaded Individual T1 - father 0 - mother 0 - sex M - status affected
Tue Feb 19 15:36:30 2013 - Loaded Individual G1 - father T1 - mother 0 - status unaffected
…
Tue Feb 19 15:36:30 2013 - Loading hg19 reference from /data/hg19/hg19.unmasked.fa
Tue Feb 19 15:37:21 2013 - Applying the Normal-Tumor design
Tue Feb 19 15:39:42 2013 - Inspecting test_Tumor_GAIN_LOSS
…
Tue Feb 19 15:39:42 2013 - VariantMaster - Cleaning workspace
Tue Feb 19 15:39:42 2013 - VariantMaster - Processing complete
```

5.5.3 Example: Unrelated individuals

VariantMaster analyze pools of unrelated individuals sharing a similar phenotype.

In this example, 5 unrelated individuals have been sequenced and respective CSV files are available in the folder *./somepath/CSV* with names Ind*[1-10].csv*. Accordingly, BAM files are in the format *bamfile_ ind[1-10].bam*.

Therefore, the samples are represented as (*unrel.tfam*)

```
U     ind1  0     0     0     2
U     ind2  0     0     1     2
U     ind3  0     0     0     2
U     ind4  0     0     1     2
U     ind5  0     0     0     2
```

where all individuals (3 males and 2 females) have been tagged as affected.

The configuration file (*unrel.cfg* - REMEMBER - it is a **TAB delimited file**) is

```
#General variables
CARR_THR := 0.95
HOM_THR := 0.75
reference := '/data/hg19/hg19.unmasked.fa'
#Filter definitions
Filter_UnrelatedFilter
conditional_select      INFO    in      PINDEL  QS      >=      300
conditional_select      INFO    in      INDEL   QS      >=      100
select  Func    in      exonic;splicing
select  QS      >=      10
```

```
select  ExonicFunc      !=      synonymous SNV
remove  segdup  -       -
End_Filter
```

5.6 Interpreting the output

VariantMaster yields comma separated (CSV) text files, one per each requested model, that are structured similarly to the input annotated CSV files. To note that it reports all the annotations by keeping the same header.

Moreover it creates some additional columns:

- *Filter* - [Pass/Rejected] the variants fulfill the conditions dictated by the requested model
- *Owner* - who is the variant coming from
- *is_in_<someID>* - The presence or absence of the variant in the unaffected individual *someID* identified by conditional variant calling with the calculated coverage and zygosity

Chr	Start	…	Owner	Filter	is_in_Father	is_in_Mother
chrX	12313	…	Proband	Pass	NotFound|Father|aa:0.0|cov:87	Mother|het|aa:0.62|p:0.99|cov:323

Some variants are not reported in the output depending on the mode of inheritance. Usually those variants can be "obviously" eliminated during specific analyses because they are present in the CSV files of the unaffected individuals (e.g. search for denovo variants in family trios).

VariantMaster produces a final report with all the variants that survived the filtering and some additional statistics. For each selected model, VariantMaster calculates the number of times a specific gene has been mutated considering all the affected ids from all provided families (column *NV*). If the appropriate reference is provided[1], the ratio number of variants/exon length (KB) in the column (column *NV/ExLen*) is reported. Moreover VariantMaster calculates the occurrence of each single variant in the given pool (*Occurrence)* to help the identification of eventual recurrent mutations.

Specific fields in final report are:

- *Gene* - Gene name
- *NV* - Total Number of variants found in *Gene*
- *MutInd* - Number of Individuals with mutated *Gene*

[1] gene_reference := /path to UCSC gene reference has to be added in the configuration file

- *NV/ExLen* - Total Number of variants found in *Gene* normalized per exon length
- *Occurrence* - Occurrence of a specific variant in the pool