iWhale

Dockerized Whole Exome Sequencing (WES) pipeline

A pipeline for Whole Exome Sequencing (WES) variants identification in matched tumor-normal samples. It runs into a Docker container, ready to be downloaded and used on any operating system supported by Docker.

All the steps of the pipeline and their dependencies are controlled by SCons so that in case of any stop, like killing by error or even shutting down the computer, it will automatically resume the analysis from the last run process.

Three variant calling softwares are used by the pipeline: Mutect2 , VarScan2, and Strelka2 and the user is allowed to choose which to use and change their default settings.

iWhale implementation

iWhale is divided in 2 parts, the first one is the alignment of reads to the reference genome, while the second one is variant calling and annotations (Fig. 1).

Steps done and software used in part 1:

- Reads alignment to the reference genome using **BWA MEM**
- Sorting of the obtained **SAM file** with **SortSam** command from **Picard**
- Removal of PCR duplicates with **Picard**
- Removal of low mapping quality reads using **GATK4** including:
    - Bad cigars
    - Unmapped reads
    - Not primary aligned reads
    - Failing vendor low quality check
    - Duplicated mapping
    - Quality of mapping not available
- Local realignment around indels with **GATK3**
- Quality score recalibration **(GATK4)**
- Coverage statistics with **bedtools**

The final result obtained from the first part are **.bam** files from both tumor and control samples which represent the final alignments of reads to the Human genome.

Steps done and software used in part 2:

The second phase is identification of **SNV** and **Indels** and their annotation. In total three variant callers are available for the identifications of mutations: **Mutect2**, **Varscan2** and **Strelka2**. The user can choose which to use (by default all three).

Starting from the **.bam** obtained from the first part, the steps done are the following:

- Removal of contaminations using Genome Aggregation Database gnomAD
- Launch of variant callers (Mutect2, Strelka2, Varscan2)
- Annotation of **SNV** or **Indels** identified, in particular with ClinVar, Cosmic, dbSNP, gnomAD, CGC (Cancer Gene Census). All annotations are done using SnpSift software except SnpEff which uses SnpEff database.
- Finally the variants obtained from the three variant callers are join together using GATK3 to obtain one .vcf from each sample.

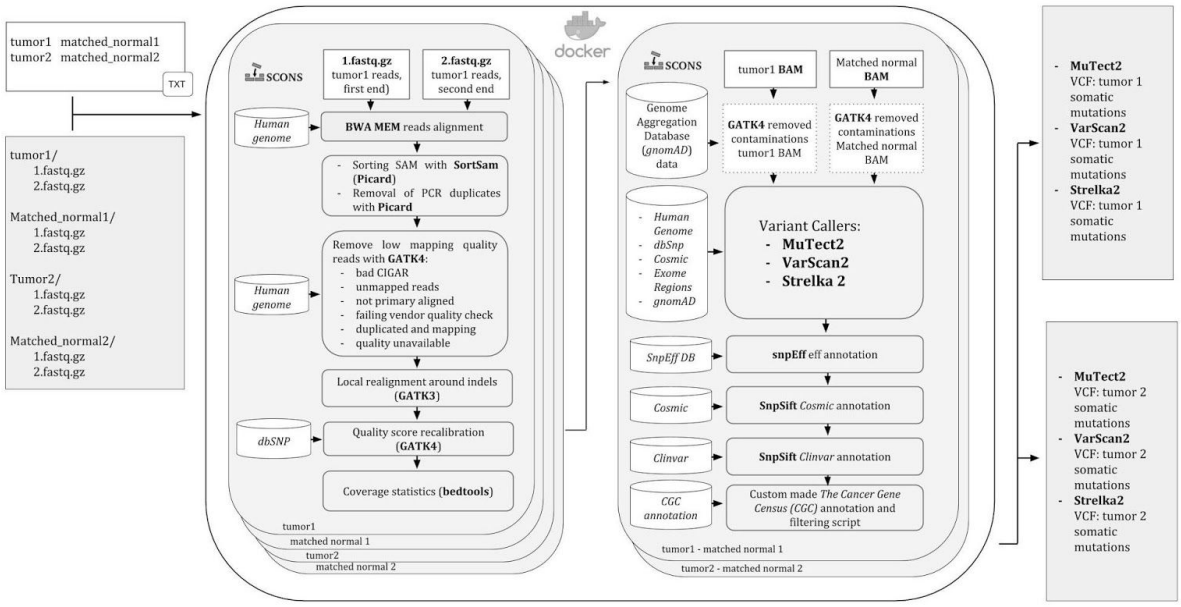

Figure1: iWhale pipeline diagram, get a bigger one from here

iWhale installation

To run iWhale you need docker, to install it follow the instructions found here: macOS, GNU/Linux or Windows

Download and install iWhale from Docker Hub:

```
docker pull alexcoppe/iWhale
```

iWhale data preparation

The working directory has to contain the following elements:

- two directories for each sample (one for tumor and one for control sample) including the four fastq files
- a text file called tumor_control_samples.txt
- a python file called configuration.py

The working directory must not contain other directories except the ones of the samples indicated above

Sample directories structure

Each sample must be in its own directory containing the two paired-end gz-compressed fastq files. The files must be called:

- 1.fastq.gz
- 2.fastq.gz

tumor_control_samples.txt file structure

The file tumor_control_samples.txt is a simple text file organized by two columns separated by tab: in the first column there are tumor directories names and in the second one the matched control directories names

```
tumor_sample1 control_sample1
tumor_sample2 control_sample2
...
```

configuration.py file structure

The configuration.py file is essential. It is used to set parameters of the tools used by iWhale. All the possible parameters that you can set are gathered and explained in this file: configuration.py. A very important parameter and the only one you have to specify is the exome regions in bed format by using the *exomeRegions* parameter. The default parameter is:

```
exomeRegions = "exome_regions.bed"
```

You have to change it to the exome interval used by your sequencing experiment (SureSelect_Human_All_Exon_V5 in the example below).

```
exomeRegions = "SureSelect_Human_All_Exon_V5.bed"
```

iWhale also needs the gziped version (made by bgzip) of the file and the .tbi index done using with these parameters (you need *tabix* and *bgzip* commands):

```
bgzip -c SureSelect_Human_All_Exon_V5.bed > SureSelect_Human_All_Exon_V5.bed.gz
tabix -p bed -S 3  SureSelect_Human_All_Exon_V5.bed.gz
```

Every MuTect2 and VarScan2 parameters can be set by using the configuration.py file. For example:

```
mutect2Parameters = "--normal-lod 2.5 --native-pair-hmm-threads 6"
varScanParameters = "--tumor-purity 2 --p-value 0.05"
```

Annotation data download

Annotation data, except COSMIC files, can be downloaded as a tar.gz or a zip one. Decompression takes a while. The version of used databases are listed below ("Databases currently used" section). COSMIC files, which are free only for academic researchers, can be downloaded from https://cancer.sanger.ac.uk/cosmic/download after sign up and login. The needed files are:

- *CosmicCodingMuts.vcf.gz*

- *cancer_gene_census.csv*
- *CosmicCodingMuts.vcf.gz.tbi*

The *CosmicCodingMuts.vcf.gz.tbi* should be made by the user. See the Testing the Docker image section to see how to make it.

Launching iWhale

The docker's run command is used to mention that we want to create an instance of an image. The image is then called a container. This is the command to launch an iWhale container:

```
docker run --rm -it --name iwhalexp -v $(pwd):/working -v
/home/user/databases:/annotations alexcoppe/iwhale
```

- --rm removes the container (it is optional)
- -it used for interactive processes (like a shell)
- --name used to name the container. If you do not assign a container name with the --name option, then the daemon generates a random string name for you
- -v used to share the two folders that iWhale needs: the working directory (used in the example the current directory by $(pwd)) and the folder including the databases files (in the example, /home/user/databases)
- iwhale is the name of the docker image to be run
- iwhalexp is the name of the iWhale docker container while iwhale is the name of the image

Testing the Docker image

You can download a small random sample (107M) in tgz format or as a zip file for a fast testing of iWhale. It contains 1 simulated small tumor (*tumor_sample*) and 1 control sample (*control_sample*), a *configuration.py* file and a *tumor_control_samples.txt* files ready. Then do the following steps:

- Install Docker Community Edition (CE) in your computer. Instructions and downloading links are here: macOS, Windows and Ubuntu
- Download and install the iWhale images with this command:

```
docker pull alexcoppe/iwhale
```

- Download and decompress annotations.tar.gz or annotations.zip
- Download the *CosmicCodingMuts.vcf.gz* and *cancer_gene_census.csv* files from COSMIC version 37 of the genome using your credentials. You need to create a .tbi file from the *CosmicCodingMuts.vcf.gz* with these commands (needed the tabix and bgzip softwares from Samtools):

```
gunzip CosmicCodingMuts.vcf.gz
bgzip -c CosmicCodingMuts.vcf > CosmicCodingMuts.vcf.gz
tabix -p vcf CosmicCodingMuts.vcf.gz
```

- Finally launch iWhale from the *iwhale_example* directory with a command similar to the following one. Just remember that the path indicated in the command, */path_to_user_annotations_directory*, should be changed to the real path were you decompressed the annotations.tar.gz file, for example */home/user/annotations*:

```
docker run --rm -it --name iwhalexp -v $(pwd):/working -v
/path_to_user_annotations_directory:/annotations alexcoppe/iwhale
```

Paths in Windows should be written differently. For example:

```
docker run --rm -it --name iwhalexp -v c:/Users:/working -v
c:/Users/annotations:/annotations alexcoppe/iwhale
```

Read Get started with Docker for Windows for a tutorial on using Docker on Windows.

Re-launching iWhale

In case of iWhale accidentally stopped, you have to re-launch the same Docker container that was running before stopping. To do this use the following command:

```
docker start -a iwhalexp
```

Obviously the name *iwhalexp* is the same of used by the *docker run* used in the "Launching iWhale" above section

Output

The final results are obtained by merging the VCFs produced by each chosen variant caller (MuTect2, VarScan2,Strelka2) and are located in the *Combined_VCFs_by_sample* directory included in the *VCF* directory. For each matched-samples pair, the called snps and indels are put into two different vcf files. In particular their name will be *tumor_control*_merged_*typeofvariants*.vcf. The index files of VCFs are also present in the directory. Here is an example of results:

```
[root@iwhale Combined_VCFs_by_sample] ls -l

-rw-r--r-- 1 root root 46578 Mar 15 15:24 5_6_merged_indels.vcf
-rw-r--r-- 1 root root  1477 Mar 15 15:24 5_6_merged_indels.vcf.idx
-rw-r--r-- 1 root root 54011 Mar 15 15:24 5_6_merged_snps.vcf
-rw-r--r-- 1 root root  3354 Mar 15 15:24 5_6_merged_snps.vcf.idx
-rw-r--r-- 1 root root 46578 Mar 15 15:25 7_8_merged_indels.vcf
-rw-r--r-- 1 root root  1477 Mar 15 15:25 7_8_merged_indels.vcf.idx
-rw-r--r-- 1 root root 54011 Mar 15 15:25 7_8_merged_snps.vcf
-rw-r--r-- 1 root root  3354 Mar 15 15:25 7_8_merged_snps.vcf.idx
```

The VCFs produced by each variant caller are present in the upper directory called *VCF*. The *Variants* directory contains the intermediate files produced by the pipeline divided by software and pair of matched-samples directories.

Software versions currently used

Program	Description	Version
Burrows-Wheeler Aligner (BWA)	BWA is a software package for mapping low-divergent sequences against a large reference genome.	0.7.17

Picard	A set of command line tools for manipulating high-throughput sequencing (HTS) data and formats.	2.17.11
GATK4	GATK4 is the first and only open-source software package that covers all major variant classes for both germline and cancer genome analysis.	4.0.6.0
GATK3	A variety of tools with a primary focus on variant discovery and genotyping.	3.8-1
Strelka2	Strelka2 is a fast and accurate small variant caller optimized for analysis of germline variation in small cohorts and somatic variation in tumor/normal sample pairs.	2.9.2
VarScan	VarScan is a platform-independent software tool developed at the Genome Institute at Washington University to detect variants in NGS data.	2.4.2
SnpEff	Genomic variant annotations and functional effect prediction toolbox.	4_3t
bedtools	Collectively, the bedtools utilities are a swiss-army knife of tools for a wide-range of genomics analysis tasks.	2.17.0

Databases currently used

iWhale uses databases and sequences indicated in the table below. Many of these sequence files should be indexed. We provide a bash script that do all the steps, from download to index.

Sequencesor	Description	Version

Databases		
Human genome database	Feb. 2009 assembly of the human genome (hg19,GRCh37 Genome Reference Consortium Human Reference 37	hg19 or GRCh37
dbSNP	dbSNP contains human single nucleotide variations, microsatellites, and small-scale insertions and deletions	All_20180418.vcf.gz
gnomAD	The Genome Aggregation Database (gnomAD), developed by an international coalition of investigators, with the goal of aggregating and harmonizing both exome and genome sequencing data from a wide variety of large-scale sequencing projects	2018-05-22
SnpEff GRCh37	SnpEff annotation for the human genome reference genome GRCh37)	GRCh37
ClinVar	ClinVar aggregates information about genomic variation and its relationship to human health	20190311
COSMIC	COSMIC, the Catalogue Of Somatic Mutations In Cancer, is the world's largest and most comprehensive resource for exploring the impact of somatic mutations in human cancer	v89

Getting database files and indexing by yourself

All commands are launch from the directory containing the downloaded data. Many of the command take a LOT OF TIME to conclude.

Download and setting of reference genome

The reference genome can be downloaded from http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/ by using these commands:

```
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr1.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr2.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr3.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr4.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr5.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr6.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr7.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr8.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr9.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr10.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr11.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr12.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr13.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr14.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr15.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr16.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr17.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr18.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr19.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr20.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr21.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr22.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chrX.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chrY.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chrM.fa.gz
```

Join all chromosomes into one big FASTA file with all human genome in it and remove chr from FASTA headers (like >chr1 to >1)

```
zcat chr1.fa.gz chr2.fa.gz chr3.fa.gz chr4.fa.gz chr5.fa.gz chr6.fa.gz chr7.fa.gz
chr8.fa.gz chr9.fa.gz \
  chr10.fa.gz chr11.fa.gz chr12.fa.gz chr13.fa.gz chr14.fa.gz chr15.fa.gz chr16.fa.gz
chr17.fa.gz \
```

```
  chr18.fa.gz chr19.fa.gz chr20.fa.gz chr21.fa.gz chr22.fa.gz chrX.fa.gz chrY.fa.gz
chrM.fa.gz  |  sed 's/>chr/>/' > reference.fa
```

Then you can remove chromosome files with this command

```
rm chr*.fa.gz
```

BWA indexing of Human genome.

It produces many files:

- reference.fa.amb
- reference.fa.ann
- reference.fa.bwt
- reference.fa.pac
- reference.fa.sa

This step takes a lot of time:

```
bwa index reference.fa
```

Index of the FASTA file with human genome data for picard.

The produced files is:

- reference.dict

```
java -jar ~/local/picard.jar CreateSequenceDictionary R=reference.fa O=reference.dict
```

Creation of the reference.fa.fai index.

The produced files is:

- reference.fa.fai

```
samtools faidx reference.fa
```

dbSNP download:

```
wget
ftp://ftp.ncbi.nih.gov/snp/organisms/human_9606_b151_GRCh38p7/VCF/All_20180418.vcf.gz
```

Removing *chr* from dbSNP downloaded file (from chr1 to 1)

```
gunzip All_20180418.vcf.gz
cat All_20180418.vcf | sed 's/>chr/>/' | bgzip -c >  All_20180418.vcf.gz
```

Indexing dbSNP VCF with tabix.

Need to install tabix in your computer. Produced file:

- All_20180418.vcf.gz.tbi

This step takes a lot of time

```
tabix -fp vcf  All_20180418.vcf.gz
```

Download of gnomAD data

Download gnomAD (version 2.0.2) data.

```
wget http://compgen.bio.unipd.it/downloads/gnomad.exomes.r2.0.2.sites.vcf.gz
wget http://compgen.bio.unipd.it/downloads/gnomad.exomes.r2.0.2.sites.vcf.gz.tbi
```

The gnomad.exomes.r2.0.2.sites.vcf.gz is compressed by *bgzip* and indexed by *tabix*.

Download of ClinVar data

Download of ClinVar data from ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf_GRCh37/. Obtained files:

- clinvar_20190311.vcf.gz
- clinvar_20190311.vcf.gz.tbi

Their date portion could be different, if so, remember to put the right filename in the configuration.py file:

```
clinvar = "clinvar_20190311.vcf.gz"
```

Download of COSMIC data

COSMIC files, which are free only for academic researchers, can be downloaded from https://cancer.sanger.ac.uk/cosmic/download after sign up and login. The needed files are:

- *CosmicCodingMuts.vcf.gz*
- *cancer_gene_census.csv*

Making the *.tbi* file:

```
gunzip CosmicCodingMuts.vcf.gz
bgzip -c CosmicCodingMuts.vcf > CosmicCodingMuts.vcf.gz
tabix -p vcf CosmicCodingMuts.vcf.gz
```