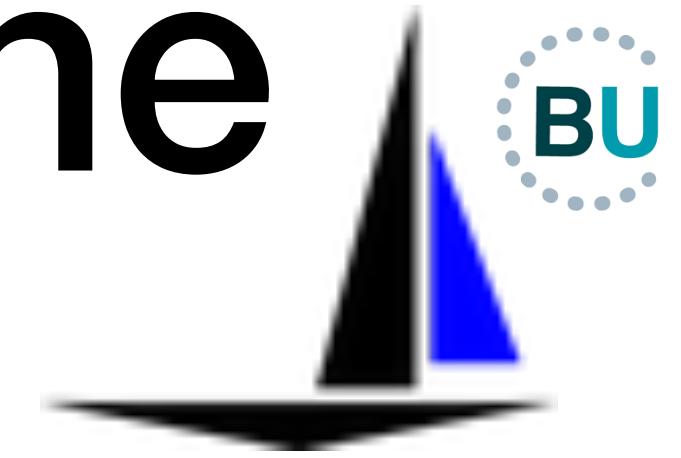


# PO: Setup varca pipeline

[https://gitlab.com/bu\\_cnio/varca](https://gitlab.com/bu_cnio/varca)





**varca**

Project ID: 12486221

141 Commits 2 Branches 0 Tags 809 KB Files 10.5 MB Storage

A Snakemake pipeline based on the GATK best-practices workflow

pipeline failed snakemake ≥5.1.5

Star 0



GitLab

master

varca

History

Find file

Clone



**Update README.md**

Elena Piñeiro authored 51 minutes ago



254f37fd

README

MIT License

CI/CD configuration

Name	Last commit	Last update
.test	Add VEP for variant annotation	1 year ago
envs	minimum Snakemake version requirement	1 week ago
img	Add logo to README	1 year ago
report	Add VCF to report.	3 years ago
rules	minimum Snakemake version requirement	1 week ago



Mölder, Felix et al. "Sustainable data analysis with Snakemake." *F1000Research* vol. 10 33. 19 Apr. 2021

**Snakemake is a workflow management system that allows:**

**Readability:** each **rule** describes a step in an analysis defining how to obtain **output files** from **input files**. Dependencies between rules are determined automatically.

**Portability:** all software dependencies of each workflow step are **automatically deployed upon execution**.

**Modularization:** rapidly implement analysis steps via direct script and jupyter notebook integration. Easily create and employ reusable tool wrappers and **split your data analysis into well-separated modules**.

**Transparency:** **automatic, interactive, self-contained reports** ensure full transparency from results down to used steps, parameters, code, and software.

**Scalability:** **workflows scale** from single to multicore, clusters or the cloud.

# HOW TO INSTALL varca



```
$ git clone https://gitlab.com/bu_cnio/varca.git
Cloning into 'varca'...
remote: Enumerating objects: 760, done.
remote: Counting objects: 100% (275/275), done.
remote: Compressing objects: 100% (133/133), done.
remote: Total 760 (delta 214), reused 167 (delta 140), pack-reused 485
Receiving objects: 100% (760/760), 1.40 MiB | 12.12 MiB/s, done.
Resolving deltas: 100% (478/478), done.
$
```

# HOW TO INSTALL Snakemake

[https://snakemake.readthedocs.io/en/stable/getting\\_started/installation.html](https://snakemake.readthedocs.io/en/stable/getting_started/installation.html)

The recommended way is via **Conda/Mamba**



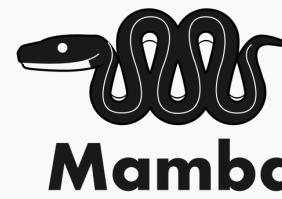
**Package, dependency and environment management** for any language.

Open source.

Runs on Windows, macOS and Linux.

Installs, runs and updates packages and their dependencies.

Conda easily creates, saves, loads and switches between environments on your local computer.



**Mamba** is a **faster** reimplementation of the conda package manager

```
$ conda install -n base -c conda-forge mamba
$ conda activate base
(base)$ mamba create -c conda-forge -c bioconda -n snakemake snakemake
```

# Running varca: dry run

- Download test data: [test.zip](#)
- Uncompress the directory
- Copy `test` content into `varca` directory

```
(base) $ cp -r path_to_test/test/* path_to_varca/varca
```

- Activate the snakemake environment

```
(base) $ conda activate snakemake
```

# Running varca: dry run and load environments run

- Test dry run: this will tell you if you're good to go!

```
(snakemake) $ snakemake --use-conda -n
Building DAG of jobs...
Job stats:
job          count    min threads    max threads
-----  -----  -----  -----
all           1          1          1
bed_to_interval      1          1          1
bwa_idx_genome       1          1          1
fastqc            4          1          1
filter_mutect_2       1          1          1
filter_mutect_calls   1          1          1
genome_dict         1          1          1
```

- Run varca with --conda-create-envs-only: this will load the environments with the needed dependencies

```
(snakemake) $ snakemake --use-conda --cores 2 --conda-create-envs-only
```

# Configuration files

**2** configuration files with the **samples definition**

**1** configuration file with the **contigs** to analyse

**1** configuration file with **paths** to additional files used in the process **and parameters**

# Configuration files - samples.tsv

group	sample	control
groups of samples	list of samples	MuTect2 calling mode

- No MuTect2 execution:

group	sample	control
1	A	-

 unable MuTect2 calling on the sample

# Configuration files - samples.tsv

group	sample	control
groups of samples	list of samples	MuTect2 calling mode

- MuTect2 execution in tumor-only mode:

group	sample	control
1	A	A

- MuTect2 execution in tumor-normal mode:

group	sample	control
1	A	B
1	B	-

# Configuration files - units.tsv

	sample	unit	platform	fq1	fq2
1	A	1	ILLUMINA	data/reads/a.chr21.1.fq	data/reads/a.chr21.2.fq
3	B	1	ILLUMINA	data/reads/b.chr21.1.fq	data/reads/b.chr21.2.fq
4	B	2	ILLUMINA	data/reads/b.chr21.1.fq	

paired-end and single-end definition

sequential to identify multiple sequences for the same sample

# Configuration files - contigs.tsv

1	chr1			
2	chr2			
3	chr3			
4	chr4			
5	chr5	@SQ	SN:chr1_KI270708v1_random	LN:127682 M5:1e95e047b98ed92148dd84d6c037158c
6	chr6	@SQ	SN:chr1_KI270709v1_random	LN:66860 M5:4e2db2933ea96aee8dab54af60ecb37d
7	chr7	@SQ	SN:chr1_KI270710v1_random	LN:40176 M5:9949f776680c6214512ee738ac5da289
8	chr8	@SQ	SN:chr1_KI270711v1_random	LN:42210 M5:af383f98cf4492c1f1c4e750c26ccb40
9	chr9	@SQ	SN:chr1_KI270712v1_random	LN:176043 M5:c38a0fecae6a1838a405406f724d6838
10	chr10	@SQ	SN:chr1_KI270713v1_random	LN:40745 M5:cb78d48cc0adbc58822a1c6fe89e3569
11	chr11	@SQ	SN:chr1_KI270714v1_random	LN:41717 M5:42f7a452b8b769d051ad738ee9f00631
12	chr12	@SQ	SN:chr2_KI270715v1_random	LN:161471 M5:b65a8af1d7bbb7f3c77eea85423452bb
13	chr13	@SQ	SN:chr2_KI270716v1_random	LN:153799 M5:2828e63b8edc5e845bf48e75fbad2926
14	chr14	@SQ	SN:chr3_GL000221v1_random	LN:155397 M5:3238fb74ea87ae857f9c7508d315babb
15	chr15	@SQ	SN:chr4_GL000008v2_random	LN:209709 M5:a999388c587908f80406444cebe80ba3
16	chr16	@SQ	SN:chr5_GL000208v1_random	LN:92689 M5:aa81be49bf3fe63a79bdc6a6f279abf6
17	chr17	@SQ	SN:chr9_KI270717v1_random	LN:40062 M5:796773a1ee67c988b4de887addbed9e7
18	chr18	@SQ	SN:chr9_KI270718v1_random	LN:38054 M5:b0c463c8efa8d64442b48e936368dad5
19	chr19	@SQ	SN:chr9_KI270719v1_random	LN:176845 M5:cd5e932cf4c74d05bb64e2126873a3a
20	chr20	@SQ	SN:chr9_KI270720v1_random	LN:39050 M5:8c2683400a4aeeb40abff96652b9b127
21	chr21	@SQ	SN:chr11_KI270721v1_random	LN:100316 M5:9654b5d3f36845bb9d19a6dbd15d2f22
22	chr22	@SQ	SN:chr14_GL000009v2_random	LN:201709 M5:862f555045546733591ff7ab15bcecbe
23	chrX	@SQ	SN:chr14_GL000225v1_random	LN:211173 M5:63945c3e6962f28ffd469719a747e73c
24	chrY	@SQ	SN:chr14_KI270722v1_random	LN:194050 M5:51f46c9093929e6edc3b4dfd50d803fc
25	chrM	@SQ	SN:chr14_GL000194v1_random	LN:191469 M5:6ac8f815bf8e845bb3031b73f812c012
				LN:38115 M5:74a4b480675592095fb0c577c515b5df

~3300 contigs in hg38

# Configuration files - config.yaml

```
1 samples: samples.tsv
2 units: units.tsv
3 contigs: contigs.tsv
4
5 outdir: "out"
6 logdir: "log"
7
8 ref:
9 # Genome database of snpeff to be used in the annotation with this resource. Available databases can be checked with java -jar snpEff.jar databases
10 name: GRCh38.86
11 # Path to the reference genome, ideally as it is provided by the GATK bundle.
12 genome: /home/epineiro/REFERENCES/Homo_sapiens/HG38/hg38.fa
13 # Path to the directory with the reference indexes for the alignment. Indexes will be retrieved from this directory or created in it if they do not exist.
14 genome_idx: /home/epineiro/REFERENCES/Homo_sapiens/HG38/indexes/bwa_mem2/
15 # Path to any database of known variants, ideally as it is provided by the GATK bundle.
16 known-variants: /home/epineiro/REFERENCES/Homo_sapiens/HG38/dbsnp_151.hg38.vcf
```

ref

Reference Genome and known variants file

# Configuration files - config.yaml

```
18 filtering:  
19   # Set to true in order to apply machine learning based recalibration of  
20   # quality scores instead of hard filtering.  
21   vqsr: false  
22   hard:  
23     # hard filtering as outlined in GATK docs  
24     # (https://gatkforums.broadinstitute.org/gatk/discussion/2806/howto-apply-hard-filters-to-a-call-set)  
25     snvs:  
26       "QD < 2.0 || QUAL < 100.0 || DP < 50.0 || SOR > 3.0 || FS > 60.0 || MQ < 40.0 || MQRankSum < -12.5 || ReadPosRankSum < -8.0"  
27     indels:  
28       "QD < 2.0 || QUAL < 100 || DP < 50.0 || FS > 200.0 || ReadPosRankSum < -20.0"  
29   mutect:  
30     #depth of coverage threshold to apply to variants identified with MuTect2  
31     depth: "DP < 30"  
32     #apply LearnReadOrientationModel to filter out read orientation artifacts  
33     lrom: false  
34
```

filtering

Variant filtering system and parameters for hard filtering

# Configuration files - config.yaml

processing

Remove duplicates in the alignment?

```
34  
35 processing:  
36   remove-duplicates: true  
37   # Uncomment and point to a bed file with, e.g., captured regions if necessary,  
38   # see https://gatkforums.broadinstitute.org/gatk/discussion/4133/when-should-i-use-l-to-pass-in-a-list-of-intervals.  
39   restrict-regions: /home/epineiro/REFERENCES/Homo_sapiens/HG38/S04380219_Padded_v5_hg38_varca.bed  
40   # If regions are restricted, uncomment this to enlarge them by the given value in order to include  
41   # flanking areas.  
42   # region-padding: 100  
43
```

Region padding?

Restrict variant calling to specific regions?

# Configuration files - config.yaml

params

gatk

Configuration parameters for GATK algorithms

```
44 params:
45   gatk:
46     HaplotypeCaller: ""
47     BaseRecalibrator: ""
48     GenotypeGVCFs: ""
49     #If vqsr set to true, fill the following section with the required information (see VariantRecalibrator documentation in GATK)
50     VariantRecalibrator:
51       #paths to the necessary resources
52       hapmap: "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_hapmap_3.3.hg38.vcf.gz"
53       omni: "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_1000G_omni2.5.hg38.vcf.gz"
54       g1k: "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_1000G_phase1.snps.high_confidence.hg38.vcf.gz"
55       dbsnp: "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_Homo_sapiens_assembly38.dbsnp138.vcf"
56       #paths to the resource indexes
57       aux: ["/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_hapmap_3.3.hg38.vcf.gz.tbi",
58             "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_1000G_omni2.5.hg38.vcf.gz.tbi",
59             "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_1000G_phase1.snps.high_confidence.hg38.vcf.gz.tbi",
60             "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_Homo_sapiens_assembly38.dbsnp138.vcf.idx"]
61       #set definition for each resource
62       parameters: {"hapmap": {"known": False, "training": True, "truth": True, "prior": 15.0},
63                   "omni": {"known": False, "training": True, "truth": False, "prior": 12.0},
64                   "g1k": {"known": False, "training": True, "truth": False, "prior": 10.0},
65                   "dbsnp": {"known": True, "training": False, "truth": False, "prior": 2.0}}
66       #Names of the annotations used for calculations
67       annotation: ["QD", "MQ", "MQRankSum", "ReadPosRankSum", "FS", "SOR"]
68       extra: ""
```

# Configuration files - config.yaml

params

picard

Configuration parameter for picard

```
68
69     picard:
70         MarkDuplicates: "REMOVE_DUPLICATES=true"
```

# Configuration files - config.yaml

params

trimmomatic

Configuration parameters for trimmomatic

```
71  trimmomatic:
72    pe:
73      trimmer:
74        # See trimmomatic manual for adding additional options, e.g. for adapter trimming.
75        - "LEADING:3"
76        - "TRAILING:3"
77        - "SLIDINGWINDOW:4:15"
78        - "MINLEN:36"
79    se:
80      trimmer:
81        # See trimmomatic manual for adding additional options, e.g. for adapter trimming.
82        - "LEADING:3"
83        - "TRAILING:3"
84        - "SLIDINGWINDOW:4:15"
85        - "MINLEN:36"
86
```

# Configuration files - config.yaml

annotation

## Parameters for VEP annotation

```
86
87 annotation:
88   vep:
89     # If set to true, a cache directory must be created using the instructions of VEP. Using a cache allows for a fastest and most efficient way to use VEP
90     cache: true
91     #Indicate the version of the cache if cache is set to true. e.g. 100. This version must match the one indicated in the vep environment (vep.yaml).
92     cache_version: 100
93     #Indicate the path to the cache files if cache is set to true
94     cache_directory: /home/epineiro/Programs/VEP/VEP100/.vep/
95     #Indicate the assembly version of the reference genome
96     assembly: GRCh38
97     #Indicate the annotations to include with the VEP execution and additional parameters. See the documentation (https://www.ensembl.org/info/docs/tools/vep/script/vep\_config.html)
98     annotations: "--sift b --polyphen b --ccds --uniprot --hgvs --symbol --numbers --domains --regulatory --canonical --protein --biotype --uniprot --"
99
```

# Configuration files - config.yaml

resources

## Machine requirements

```
100 resources:
101   default:
102     threads: 1
103     mem: 32000
104     walltime: 720
105   snpeff:
106     mem: 8000
107     walltime: 480
108   call_variants:
109     mem: 10000
110     threads: 4
111     walltime: 480
112   combine_calls:
113     mem: 8000
114     threads: 1
115     walltime: 480
116   recalibrate_calls:
117     mem: 8000
118     walltime: 480
119   hard_filter_calls:
120     mem: 8000
121     walltime: 480
122   trim_reads:
123     mem: 4000
124     threads: 16
125     walltime: 480
126   map_reads:
127     mem: 32000
128     threads: 8
129     walltime: 480
```

# Extra: In case you need to install conda

- Go to <https://docs.conda.io/en/latest/miniconda.html#linux-installers>
- Download the miniconda bundle (choose the optimal version for your computer)

```
$ wget https://repo.anaconda.com/miniconda/Miniconda3-py39_4.10.3-Linux-x86_64.sh
```

- In the directory where the file Miniconda3-py39\_4.10.3-Linux-x86\_64.sh

```
$ bash Miniconda3-py39_4.10.3-Linux-x86_64.sh
```

- Once the installation is over, restart the terminal.

