# HOWARD README

## 1 HOWARD

HOWARD - Highly Open Workflow for Annotation & Ranking toward genomic variant Discovery

Highly Open Workflow for Annotation & Ranking toward genomic variant Discovery

HOWARD annotates and prioritizes genetic variations, calculates and normalizes annotations, translates files in multiple formats (e.g. vcf, tsv, parquet) and generates variants statistics.

HOWARD annotation is mainly based on a build-in Parquet annotation method, and external tools such as BCFTOOLS, ANNOVAR, snpEff, Exomiser and Splice (see docs, automatically downloaded if needed). Parquet annotation uses annotation database in VCF or BED format, in mutliple file format: Parquet/duckdb, VCF, BED, TSV, CSV, TBL, JSON.

HOWARD calculation processes variants information to calculate new information, such as: harmonizes allele frequency (VAF), extracts Nomen (transcript, cNomen, pNomen...) from HGVS fields with an optional list of personalized transcripts, generates VaRank format barcode.

HOWARD prioritization algorithm uses profiles to flag variants (as passed or filtered), calculate a prioritization score, and automatically generate a comment for each variants (example: 'polymorphism identified in dbSNP. associated to Lung Cancer. Found in ClinVar database').Prioritization profiles are defined in a configuration file. A profile is defined as a list of annotation/value, using wildcards and comparison options (contains, lower than, greater than, equal...). Annotations fields may be quality values (usually from callers, such as 'GQ', 'DP') or other annotations fields provided by annotations tools, such as HOWARD itself (example: COSMIC, Clinvar, 1000genomes, PolyPhen, SIFT). Multiple profiles can be used simultaneously, which is useful to define multiple validation/prioritization levels (example: 'standard', 'stringent', 'rare variants', 'low allele frequency').

HOWARD translates VCF format into multiple formats (e.g. VCF, TSV, Parquet), by sorting variants using specific fields (example : 'prioritization score', 'allele frequency', 'gene symbol'), including/excluding annotations/fields, including/excluding variants, adding fixed columns.

HOWARD generates statistics files with a specific algorithm, snpEff and BCFTOOLS.

HOWARD is multithreaded through the number of variants and by database (data-scaling).

HOWARD is able to add plugins for further analyses.

### 1.1 Table of contents

# 2 Installation

HOWARD can be installed using Python, and a Docker installation provides a CLI (Command Line Interface) with all external tools and useful databases. Databases can be automatically downloaded, or home-made generated (created or downloaded).

## 2.1 Download

Download sources from gitHub

```
mkdir -p ~/howard/src
cd ~/howard/src
git clone https://github.com/bioinfo-chru-strasbourg/howard.git .
```

## 2.2 Python

Install HOWARD using Python Pip tool, and run HOWARD for help options:

```
conda create --name=howard python=3.10
conda activate howard
python -m pip install -e .
howard --help
```

```
usage: howard [-h] {query,stats,convert,hgvs,annotation,calculation,prioritization,process,databases,gui} ...

HOWARD:0.12.1.1 - Highly Open Workflow for Annotation & Ranking toward genomic variant Discovery

Shared arguments:
  -h, --help            show this help message and exit

Tools:
  {query,stats,convert,hgvs,annotation,calculation,prioritization,process,databases,gui}
    query               Query genetic variations file in SQL format.
    stats               Statistics on genetic variations file.
    convert             Convert genetic variations file to another format.
    hgvs                HGVS annotation (HUGO internation nomenclature) using refGene,
                        genome and transcripts list.
    annotation          Annotation of genetic variations file using databases/files and tools.
    calculation         Calculation operations on genetic variations file and genotype information.
    prioritization      Prioritization of genetic variations based on annotations criteria (profiles).
    process             Full genetic variations process: annotation, calculation, prioritization,
                        format, query, filter...
    databases           Download databases and needed files for howard and associated tools
    gui                 Graphical User Interface tools
```

Install HOWARD Graphical User Interface using Python Pip tool with supplementary packages, and run as a tool:

```
python -m pip install -r requirements-gui.txt
howard gui
```

HOWARD Graphical User Interface

## 2.3 Docker

In order to build, setup and create a persitent CLI (running container with all useful external tools such as BCFTools, snpEff, Annovar, Exomiser), docker-compose command build images and launch services as containers.

```
docker-compose up -d
```

A setup container (HOWARD-setup) will download useful databases (take a while). To avoid databases download (see Databases section to download manually), just start:

```
docker-compose up -d HOWARD-CLI
```

A Command Line Interface container (HOWARD-CLI) is started with host data and databases folders mounted (by default in ~/howard folder, i.e. `~/howard/data:/data` and `~/howard/databases:/databases`). Let's play within Docker HOWARD-CLI service!

```
docker exec -ti HOWARD-CLI bash
howard --help
```

More details

Docker HOWARD-CLI container (Command Line Interface) can be used to execute commands.

Example: Query of an existing VCF

```
docker exec HOWARD-CLI \
   howard query \
   --input='/tool/tests/data/example.vcf.gz' \
   --query='SELECT * FROM variants'
```

Example: VCF annotation using HOWARD-CLI (snpEff and ANNOVAR databases will be automatically downloaded), and query list of genes with HGVS

```
docker exec --workdir=/tool HOWARD-CLI \
   howard process \
      --config='config/config.json' \
      --param='config/param.json' \
      --input='tests/data/example.vcf.gz' \
      --output='/tmp/example.process.tsv' \
      --explode_infos \
      --query="SELECT NOMEN, PZFlag, PZScore, PZComment \
               FROM variants \
               ORDER BY PZScore DESC"
```

## 2.4 Databases

Multiple databases can be automatically downloaded with databases tool, such as:

| database | description |
|---|---|
| Genome | Genome Reference Consortium Human |
| Annovar | ANNOVAR is an efficient software tool to utilize update-to-date information to functionally annotate genetic variants detected from diverse genomes |
| snpEff | Genetic variant annotation, and functional effect prediction toolbox |
| refSeq | A comprehensive, integrated, non-redundant, well-annotated set of reference sequences including genomic, transcript, and protein |
| dbSNP | dbSNP contains human single nucleotide variations, microsatellites, and small-scale insertions and deletions along with publication, population frequency, molecular consequence, and genomic and RefSeq mapping information for both common variations and clinical mutations |
| dbNSFP | dbNSFP is a database developed for functional prediction and annotation of all potential non-synonymous single-nucleotide variants (nsSNVs) in the human genome |
| AlphaMissense | AlphaMissense model implementation |
| Exomiser | The Exomiser is a Java program that finds potential disease-causing variants from whole-exome or whole-genome sequencing data |

More details

Example: Download Multiple databases in the same time for assembly 'hg19' (can take a while)

```
howard databases \
   --assembly=hg19 \
```

```
    --download-genomes='~/howard/databases/genomes/current' \
    --download-genomes-provider='UCSC'\
    --download-genomes-contig-regex='chr[0-9XYM]+$' \
    --download-annovar='~/howard/databases/annovar/current' \
    --download-annovar-files='refGene,cosmic70,nci60' \
    --download-snpeff='~/howard/databases/snpeff/current' \
    --download-refseq='~/howard/databases/refseq/current' \
    --download-refseq-format-file='ncbiRefSeq.txt' \
    --download-dbnsfp='~/howard/databases/dbnsfp/current' \
    --download-dbnsfp-release='4.4a' \
    --download-dbnsfp-subdatabases \
    --download-alphamissense='~/howard/databases/alphamissense/current' \
    --download-exomiser='~/howard/databases/exomiser/current' \
    --download-dbsnp='~/howard/databases/dbsnp/current' \
    --download-dbsnp-vcf \
    --threads=8
```

See HOWARD Help Databases tool for more information.

Databases can be home-made generated, starting with a existing annotation file, especially using HOWARD convert tool. These files need to contain specific fields (depending on the annotation type):

- variant annotation: '#CHROM', 'POS', 'ALT', 'REF'
- region annotation: '#CHROM', 'START', 'STOP'

Each database annotation file is associated with a 'header' file ('.hdr'), in VCF header format, to describe annotations within the database.

### 2.5 Configuration

HOWARD Configuration JSON file defined default configuration regarding resources (e.g. threads, memory), settings (e.g. verbosity, temporary files), default folders (e.g. for databases) and paths to external tools.

See HOWARD Configuration JSON for more information.

## 3 Tools

HOWARD annotates and prioritizes genetic variations, calculates and normalizes annotations, convert on multiple formats, query variations and generates statistics. These tools require options or a Parameters JSON file.

### 3.1 Parameters

HOWARD Parameters JSON file defined parameters to process annotations, prioritization, calculations, convertions and queries. Use this parameters file to configure tools, instead of options or as a main configuration (options will replace parameters in JSON file).

See HOWARD Parameters JSON for more information.

### 3.2 Stats

Statistics on genetic variations, such as: number of variants, number of samples, statistics by chromosome, genotypes by samples, annotations. Theses statsitics can be applied to VCF files and all database annotation files.

More details

Example: Show example VCF statistics and brief overview

```
howard stats \
   --input='tests/data/example.vcf.gz'
```

See HOWARD Help Stats tool for more information.

## 3.3 Convert

Convert genetic variations file to another format. Multiple format are available, such as usual and official VCF format, but also other formats such as TSV, CSV, TBL, JSON and Parquet/duckDB. These formats need a header '.hdr' file to take advantage of the power of howard (especially through INFO/tag definition), and using howard convert tool automatically generate header file fo futher use (otherwise, an default '.hdr' file is generated).

More details

Example: Translate VCF into TSV, export INFO/tags into columns, and show output file

```
howard convert \
   --input='tests/data/example.vcf.gz' \
   --explode_infos \
   --output='/tmp/example.tsv'
cat '/tmp/example.tsv'
```

See HOWARD Help Convert tool for more options.

## 3.4 Query

Query genetic variations in SQL format. Data can be loaded into 'variants' table from various formats (e.g. VCF, TSV, Parquet...). Using 'explode' option allows querying on INFO/tag annotations. SQL query can also use external data within the request, such as a Parquet file(s).

More details

Example: Select variants in VCF with INFO Tags criterions

```
howard query \
   --input='tests/data/example.vcf.gz' \
   --explode_infos \
   --query='SELECT "#CHROM", POS, REF, ALT, DP, CLNSIG, sample2, sample3
            FROM variants
            WHERE DP >= 50 OR CLNSIG NOT NULL
            ORDER BY CLNSIG DESC, DP DESC'
```

See HOWARD Help Query tool for more options.

## 3.5 Annotation

Annotation is mainly based on a build-in Parquet annotation method, using database format such as Parquet, duckdb, VCF, BED, TSV, JSON. External annotation tools are also available, such as BCFTOOLS, Annovar, snpEff, Exomiser and Splice. It uses available databases and homemade databases. Annovar and snpEff databases are automatically downloaded (see HOWARD Help Databases tool). All annotation parameters are defined in HOWARD Parameters JSON file.

Quick annotation allows to annotates by simply listing annotation databases, or defining external tools keywords. These annotations can be combined.

More details

Example: VCF annotation with Parquet and VCF databases, output as VCF format

```
howard annotation \
   --input='tests/data/example.vcf.gz' \
   --annotations='tests/databases/annotations/current/hg19/dbnsfp42a.parquet,
      tests/databases/annotations/current/hg19/cosmic70.vcf.gz' \
   --output='/tmp/example.howard.vcf.gz'
```

Example: VCF annotation with external tools (Annovar refGene and snpEff databases), output as TSV format

```
howard annotation \
   --input='tests/data/example.vcf.gz' \
   --annotations='annovar:refGene,snpeff' \
   --output='/tmp/example.howard.tsv'
```

See HOWARD Help Annotation tool for more options.

## 3.6 Calculation

Calculation processes variants information to generate new information, such as: identify variation type (VarType), harmonizes allele frequency (VAF) and calculate sttistics (VAF_stats), extracts Nomen (transcript, cNomen, pNomen...) from an HGVS field (e.g. snpEff, Annovar) with an optional list of personalized transcripts, generates VaRank format barcode, identify trio inheritance.

More details

Example: Identify variant types and generate a table of variant type count

```
howard calculation \
   --input='tests/data/example.full.vcf' \
   --calculations='vartype' \
   --output='/tmp/example.calculation.tsv'

howard query \
   --input='/tmp/example.calculation.tsv' \
   --explode_infos \
   --query='SELECT
               "VARTYPE" AS 'VariantType',
               count(*) AS 'Count'
            FROM variants
            GROUP BY "VARTYPE"
            ORDER BY count DESC'
```

| | VariantType | Count |
|---|---|---|
| 0 | BND | 7 |
| 1 | DUP | 6 |
| 2 | INS | 5 |
| 3 | SNV | 4 |
| 4 | CNV | 3 |
| 5 | DEL | 3 |
| 6 | INV | 3 |
| 7 | MOSAIC | 2 |
| 8 | INDEL | 2 |
| 9 | MNV | 1 |

See HOWARD Help Calculation tool for more options.

## 3.7 Prioritization

Prioritization algorithm uses profiles to flag variants (as passed or filtered), calculate a prioritization score, and automatically generate a comment for each variants (example: 'polymorphism identified in dbSNP. associated to Lung Cancer. Found in ClinVar database'). Prioritization profiles are defined in a configuration file in JSON format. A profile is defined as a list of annotation/value, using wildcards and comparison options (contains, lower than, greater than, equal...). Annotations fields may be quality values (usually from callers, such as 'DP') or other annotations fields provided by annotations tools, such as HOWARD itself (example: COSMIC, Clinvar, 1000genomes, PolyPhen, SIFT).

Multiple profiles can be used simultaneously, which is useful to define multiple validation/prioritization levels (e.g. 'standard', 'stringent', 'rare variants'). Prioritization score can be calculated following multiple mode, either 'HOWARD' (incremental) or 'VaRank' (maximum). Prioritization fields can be selected (PZScore, PZFlag, PZComment, PZTags, PZInfos).

More details

Example: Prioritize variants from criteria on INFO annotations for profiles 'default' and 'GERMLINE' (from 'prioritization_profiles.json' profiles configuration), export prioritization tags, and query variants passing filters

```
howard prioritization \
   --input='tests/data/example.vcf.gz' \
   --prioritization_config='config/prioritization_profiles.json' \
   --prioritizations='default,GERMLINE' \
   --default_profile='default' \
   --pzfields='PZFlag,PZScore,PZComment,PZTags,PZInfos' \
```

```
   --prioritization_score_mode='HOWARD' \
   --output='/tmp/example.prioritized.vcf.gz'

howard query \
   --input='/tmp/example.prioritized.vcf.gz' \
   --explode_infos \
   --query="SELECT \"#CHROM\", POS, ALT, REF, PZFlag, PZScore, PZTags, DP, CLNSIG \
            FROM variants \
            WHERE PZScore > 0 \
              AND PZFlag == 'PASS' \
            ORDER BY PZScore DESC"
```

| | #CHROM | POS | ALT | REF | PZFlag | PZScore | PZTags | DP | CLNSIG |
|---|---|---|---|---|---|---|---|---|---|
| 0 | chr1 | 28736 | C | A | PASS | 15 | PZFlag#PASS\|PZScore#15... | NaN | pathogenic |
| 1 | chr1 | 69101 | G | A | PASS | 5 | PZFlag#PASS\|PZScore#5\|... | 50.0 | None |
| 2 | chr7 | 55249063 | A | G | PASS | 5 | PZFlag#PASS\|PZScore#5\|... | 125.0 | None |

See HOWARD Help Prioritization tool for more options.

## 3.8 HGVS Annotation

HOWARD annotates variants with HGVS annotation using HUGO HGVS internation Sequence Variant Nomenclature (http://varnomen.hgvs.org/). Annotation refere to refGene and genome to generate HGVS nomenclature for all available transcripts. This annotation add 'hgvs' field into VCF INFO column of a VCF file. Several options are available, to add gene, exon and protein information, to generate a "full format" detailed annotation, to choose codon format.

See HOWARD Help HGVS tool for more options.

More details

Example: HGVS annotation with quick options

```
howard hgvs \
   --input='tests/data/example.vcf.gz' \
   --output='/tmp/example.process.tsv' \
   --hgvs=full_format,use_exon

howard query \
   --input='/tmp/example.process.tsv' \
   --explode_infos \
   --query="SELECT hgvs \
            FROM variants "
```

| | hgvs |
|---|---|
| 0 | WASH7P:NR_024540.1:n.50+585T>G |
| 1 | FAM138A:NR_026818.1:exon3:n.597T>G:p.Tyr199Asp |
| 2 | OR4F5:NM_001005484.2:NP_001005484.2:exon3:c.74... |
| 3 | LINC01128:NR_047526.1:n.287+3767A>G,LINC01128:... |
| 4 | LINC01128:NR_047526.1:n.287+3768A>G,LINC01128:... |
| 5 | LINC01128:NR_047526.1:n.287+3769A>G,LINC01128:... |
| 6 | EGFR:NM_001346897.2:NP_001333826.1:exon19:c.22... |

## 3.9 Process

HOWARD process tool manage genetic variations to:

- annotates genetic variants with multiple annotation databases/files and tools
- calculates and normalizes annotations
- prioritizes variants with profiles (list of citeria) to calculate scores and flags
- annotates genetic variants with HGVS nomenclature
- translates into various formats
- query genetic variants and annotations
- generates variants statistics

This process tool combines all other tools to pipe them in a uniq command, through available options or a parameters file in JSON format (see HOWARD Parameters JSON file).

See HOWARD Help Process tool tool for more information.

More details

Example: Full process command with options (HGVS, annotation, calculation and prioritization)

```
howard process \
   --input='tests/data/example.vcf.gz' \
   --output='/tmp/example.process.tsv' \
   --hgvs='full_format,use_exon' \
   --annotations='tests/databases/annotations/current/hg19/avsnp150.parquet,
      tests/databases/annotations/current/hg19/dbnsfp42a.parquet,
      tests/databases/annotations/current/hg19/gnomad211_genome.parquet,
      bcftools:tests/databases/annotations/current/hg19/cosmic70.vcf.gz,
      snpeff,
      annovar:refGene' \
   --calculations='vartype,snpeff_hgvs,VAF,NOMEN' \
   --prioritization_config='config/prioritization_profiles.json' \
   --prioritizations='default' \
   --explode_infos \
   --query="SELECT NOMEN, PZFlag, PZScore \
            FROM variants \
            ORDER BY PZScore DESC"
```

| | NOMEN | PZFlag | PZScore |
|---|---|---|---|
| 0 | WASH7P:NR_024540:n.50+585T>G | PASS | 15 |
| 1 | OR4F5:NP_001005484:exon3:c.74A>G:p.Glu25Gly | PASS | 5 |
| 2 | EGFR:NM_001346897:exon19:c.2226G>A:p.Gln742Gln | PASS | 5 |
| 3 | LINC01128:NR_047526:n.287+3767A>G | PASS | 0 |
| 4 | LINC01128:NR_047526:n.287+3768A>G | PASS | 0 |
| 5 | LINC01128:NR_047526:n.287+3769A>G | PASS | 0 |
| 6 | FAM138A:NR_026818:exon3:n.597T>G:p.Tyr199Asp | FILTERED | -100 |

# 4 Documentation

HOWARD User Guide is available to assist users for particular commands, such as software installation, databases download, annotation command, and so on.

HOWARD Tips proposes some additional advices to handle HOWARD for particular use cases.

HOWARD Help describes options of all HOWARD tools. All information are also available for each tool using `--help` option.

HOWARD Configuration JSON describes configuration JSON file structure and options.

HOWARD Parameters JSON describes parameters JSON file structure and options.

HOWARD Parameters Databases JSON describes configuration JSON file for databases download and convert.

HOWARD Plugins describes how to create HOWARD plugins.

HOWARD Package describes HOWARD Package, Classes and Functions.

# 5 Contact

Medical Bioinformatics applied to Diagnosis Lab @ Strasbourg Univerty Hospital

bioinfo@chru-strasbourg.fr

GitHub