VarAnnot v1.0

**A Python Variant Annotator for Human Genetic Variations in VCF v4.0 format generated by Platypus v0.8.1 Variant Caller**

**Introduction**

This is a simple VCF annotator to parse VCF v4.0 files generated by Platypus v0.8.1 and annotate human genetic variants according to a patient’s genotype using Exome Aggregation Consortium (EXAC) and ClinVar databases.

Annotation involves adding information on the consequence of the variant on protein sequence, rsID, depth of coverage at the site of variation, depth of ALT allele at the site of variation, allele balance ratio, EXAC frequency of the variant if available, associated gene and transcript names.

This software is inspired from Ensemble’s Variant Effect Predictor (**VEP**) and Varant tools.

**Requirements**

This software was coded for **python v2.7.10**. This program requires the following python libraries and files:

* Operator
* Optparse
* Collections
* Requests v2.11.1 -- HTTP library
* PyVCF v0.6.8 -- VCF parser
* Pysam v0.9.1.4 – a Samtools wrapper to query VCF files **(available only for Linux and macOS)**
* Clinvar.vcf.gz & Clinvar.vcf.gz.tbi **(Included in the bin folder)**

**Check if the packages are already installed**:

Packages operator, optparse and collections are inbuilt in python 2.7. Type the following commands in python environment to check if the packages are installed properly.

from operator import \*

import optparse

import collections

import requests

import vcf

import pysam

**Install Packages**

The required packages can be installed with the commands:

sudo pip install requests **(or)** pip install --user requests

sudo pip install pyvcf **(or)** pip install pyvcf

sudo pip install pysam **(or)** pip install pysam

**For Anaconda**

conda create -n myenv python=[python\_version]

source activate myenv

conda install requests

conda install -c bioconda pyvcf

conda install -c bioconda pysam

conda install -override-channels –c conda-forge bzip2

**Sample Usage**

Download varAnnot-master.zip from <https://github.com/GayathriRajan/varAnnot>

Unzip the file and run

cd varAnnot-master/var\_annot\_v1.0/

**python varAnnot.py –help**

**python varAnnot.py -i test/test2\_data.vcf -s sample -c bin/clinvar.vcf.gz**

Check for result file test/sample\_annotated.txt

**Input/Output**

Input – v4.0 VCF from Platypus v0.8.1

Output – samplename\_annotated.txt in the same path as the input folder

**Output Columns**:

* CHROM
* POS
* REF
* ALT
* rsID (dbSNP ID)
* ClinSig (Clinical Significance)
* ConseqType
* NR (Depth of Coverage)
* NV (Allele Depth)
* ABratio (Allele Balance)
* ExacAF (EXAC Allele Frequency)
* Gene
* Transcript

See Annotations below for more explanation about each field.

**Annotations:**

1. **Variant Clinical Significance (from ClinVar)**

|  |  |
| --- | --- |
| **Significance Code** | **Significance** |
| 0 | Uncertain |
| 1 | Not provided |
| 2 | Benign |
| 3 | Likely Benign |
| 4 | Likely Pathogenic |
| 5 | Pathogenic |
| 6 | Drug Response |
| 7 | Histocompatibility |
| 255 | Other |

1. **Consequence of the Variant on Protein Sequence (Adapted from VEP)**

For each variant, the ensemble transcript is identified and then using Sequence Ontology (SO) terms, the effects that the allele may have on the transcript is obtained. For variants that may have more than one predicted impact, the impact that might be more severe is selected. If the variant affects more than one gene, both the genes are provided in the annotation. Consequence classes in the order of their severity are shown in the table below:

**Consequence Classes of Variants on Protein Function**

|  |  |  |
| --- | --- | --- |
| Class Rank | Impact Class | Impact on Protein |
| 1 | High  (The variant is assumed to have high (disruptive) impact in the protein, probably causing protein truncation, loss of function or triggering nonsense mediated decay.) | Transcript\_ablation  Splice\_acceptor\_variant  Splice\_donor\_variant  Stop\_gained  Frameshift\_variant  Stop\_lost  Start\_lost  Transcript\_amplification |
| 2 | Moderate  (A non-disruptive variant that might change protein effectiveness.) | Inframe\_insertion  Inframe\_deletion  Missense\_variant  Protein\_altering\_variant |
| 3 | Low  (Assumed to be mostly harmless or unlikely to change protein behaviour.) | Splice\_region\_variant  Incomplete\_terminal\_codon\_variant  Stop\_retained\_variant  Synonymous\_variant |
| 4 | Modifier  (Usually non-coding variants or variants affecting non-coding genes, where predictions are difficult or there is no evidence of impact.) | Coding\_sequence\_variant  Mature\_miRNA\_variant  5\_prime\_UTR\_variant  3\_prime\_UTR\_variant  Non\_coding\_transcript\_exon\_variant  Intron\_variant  NMD\_transcript\_variant  Non\_coding\_transcript\_variant  Upstream\_gene\_variant  Downstream\_gene\_variant  TFBS\_ablation  TFBS\_amplification  TF\_binding\_site\_variant  Regulatory\_region\_ablation  Regulatory\_region\_amplification  Feature\_elongation  Regulatory\_region\_variant  Feature\_truncation  Intergenic\_variant |

1. **Depth of Coverage at the Site of Variation**

This is the total number of reads at the site of variation, obtained from the attribute:

##FORMAT=<ID=NR> attribute from the input VCF.

1. **Allele Depth**

This is the number of reads at the site of variation that support the variant, obtained from the attribute: ##FORMAT=<ID=NV> attribute from the input VCF.

1. **Allele Balance Percent**

This is defined as the proportion of reads that are supporting the variant vs. the reference:

Allele Balance (AB) = (# ALT reads) \*100/ (# REF + ALT reads)

= (NV\*100)/ (NR)

**Features:**

* Annotates only variants present in given sample by genotype
* Code can be modified to handle multiple samples
* Features obtained from EXAC can be customized
* Handles EXAC annotations from multiple genes with same consequence type for a variant
* Handles non-chromosome scaffolds

**Challenges:**

* Rank annotations by subtype of Impact class
* Platform compatibility of python libraries used (Pysam)

**Contact**

For any queries regarding the execution of the script please contact: Gayathri Rajan (gayathri.rajan89@gmail.com).