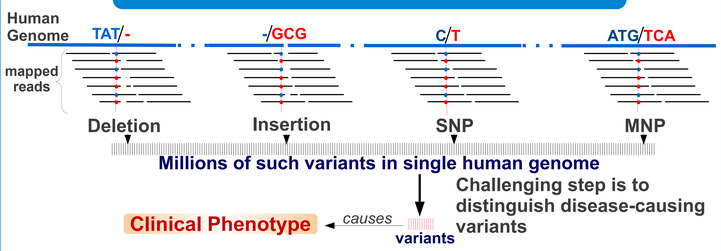
Varant: An Open Source Variant Annotation Tool

Kunal Kundu[1], Uma Sunderam[1], Steven E. Brenner[2], Rajgopal Srinivasan[1#]
[1]Tata Consultancy Services Ltd, Innovation Labs, Hyderabad, India.
[2]Department of Plant and Microbial Biology, University of California, Berkeley
[#]Corresponding author: email address raj@atc.tcs.com

Motivation



In a clinical genomics sequencing study, generally the aim is to distinguish variants which cause the disease from the millions of variants present in a single human genome. A challenge lies in interpreting the functional relevance of each variant in order to facilitate the distillation of these to a narrower set of more relevant variants for further investigation. Comprehensive annotation of variants is a necessary first step in arriving at a small subset of variants that are most likely to explain the phenotype(s) under investigation.

We have developed a **comprehensive** and **extensible** open source tool for human genetic variation annotation called Varant, written in the Python programming language.

Features & Annotations

Varant's features and annotation types were compared with other two well known tools to ensure that it has all their features in addition to the **10 annotation types that is provided only by Varant** to facilitate the variant impact interpretation.

Features	Annovar	snpEff	Varant
License	Commercial but freely available to personal, academic, and non-profit use only.	Open source:LGPLv3	Open source:LGPLv3
Variant types that the tool can annotate	SNPs, Indels	SNPs, Indels, MNPs	SNPs, Indels, MNPs
Input Format	vcf, tsv	vcf, tsv(deprecated)	vcf
Output Format	tsv	vcf, tsv	vcf, tsv

	Annotations	Annovar	snpEff	Varant
1	Region – Intergenic, Intronic, Exonic, UTR	●	●	●
2	Downstream and upstream gene for intergenic variants	●	●	●
3	Splice Sites (Donor/Acceptor)	●	●	●
4	Mutation Types – NonSyn, Syn, StartGain, StartLoss, StopGain, StopLoss, SynStop	●	●	●
5	Position Conservation(4)	●	●	●
6	TFBS	●	●	●
7	GWAS Phenotype	●	●	●
8	dbSNP, 1000Genome(MAF) and ESP(MAF)	●	●	●
9	Polyphen2 and SIFT predictions(1)	●	●	●
10	miRNA Binding Site(3)	●	●	●
11	Clinically significant variants - ClinVar	●		●
12	Gene-Disease association – OMIM, NCBI-GAD			●
13	Exonic splice enhancer / silencer site – Burge et al (2)			●
14	UTR Functional Motifs – UTRdb (5)			●
15	Flag variants at or spanning boundary region like Intron-Exon or UTR-CDS			●
16	Distance of intronic variants from splice sites			●
17	Low Complexity Region			●
18	Pseudo Autosomal Regions			●
19	Codon Usage			●
20	Capture region annotations			●
21	eQTL			●

Conclusion

- Varant provides a broad range of annotations for interpreting the functional relevance of genetic variants.
- Varant is easy to be deployed on any computer as most of the installation process is automated.
- In comparison with other well known tool, Varant provides annotations with better precision.
- Although Varant provides parser for its annotations, the annotations can be easily parsed by any VCF parser as all the annotations are written to VCF file in compliance to standard VCF format.

References

1. Boerwinkle E et al(2011). Hum Mutat. 32, 894-9. doi:10.1002/humu.21517.
2. Burge CB et al. (2002) Science. 297, 1007-13. Epub 2002 Jul 11
3. Sander C et al.(2008) Nucleic Acids Res. 36, D149-53. Epub 2007 Dec 23
4. Batzoqlou S et al. (2010). PLoS Comput Biol. 6, e1001025. doi:10.1371/journal.pcbi.1001025
5. Pesole G et al. (2010) Nucleic Acids Res. 38, D75-80. doi: 10.1093/nar/gkp902
6. Douglas M. Ruden et al. (2012) Fly (Austin). 6, 80–92. doi:10.4161/fly.19695
7. Hakon Hakonarson et al. (2010) Nucl. Acids Res.38, e164. doi:10.1093/nar/gkq603

Annotation Accuracy

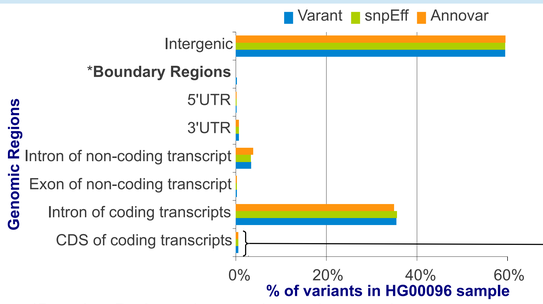
To estimate the accuracy of Varant, annotations for 3,836,489 variants(SNPs and Indels) present in HG00096 sample (from 1000 Genomes) were extensively compared among Varant, Annovar(7) and snpEff(6).

99.15 % of variants had exactly same annotation by Varant, SnpEff and Annovar

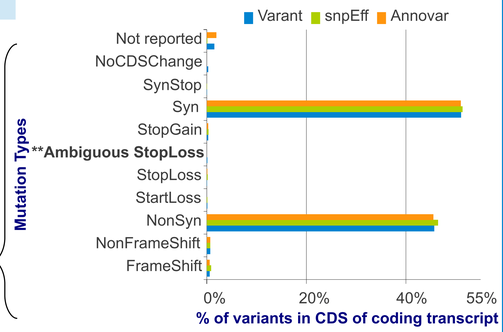
As expected there was significant overlap in the annotations – especially annotations like region type(intergenic/exon/intron), mutation type, and transcript based amino-acid changes. The discrepancy cases (0.85% of variants) we categorized in following 4 types and then were manually inspected -

Discrepancy Category	% of variants
Same annotation by Varant & snpEff but not by Annovar	0.72%
Same annotation by Varant & Annovar but not by SnpEff	0.02%
Same annotation by SnpEff & Annovar but not by Varant	0.1%
Entirely different annotation by Varant, snpEff & Annovar	0.0008% (30 variants)

After manual inspection it was observed that all of Varant's annotations were logical in comparison with other two tools.



*Boundary Region variants are the one that spans intron-exon, UTR-CDS or intergenic-UTR boudaries. **This precision of region annotations is provided only by Varant.**

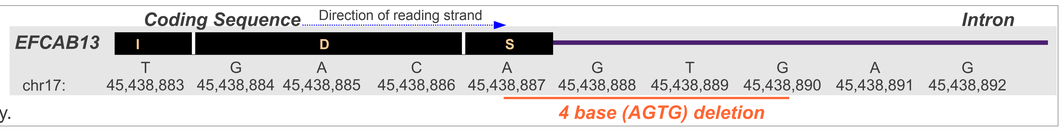


**Ambiguous StopLoss variants are the one that alters the stop codon which are present in middle of CDS rather than the end. This annotation is provided only by Varant.

Dark edge cases where discrepancies were observed

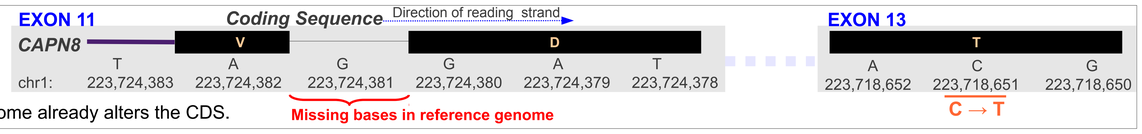
Deletion variant that spans intron exon boundary

SnpEff – splicing
Annovar – exonic;frameshift
Varant (*better precision*) – intron-exon boundary;splicing
Varant in addition labels the variant as one that spans intron exon boundary.



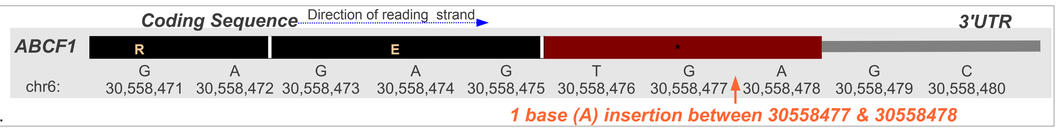
Variants on transcripts whose CDS is incomplete

SnpEff(incorrect) – Synonymous,
Annovar – exonic (mutation type not computed)
Varant – exonic (mutation type not computed)
The missing bases upstream to variant position in reference genome already alters the CDS.



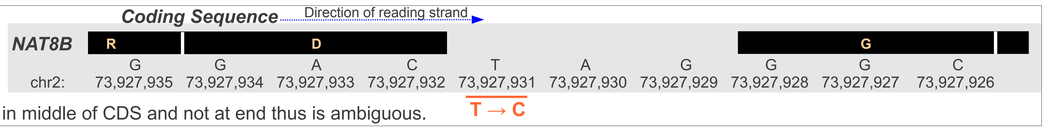
Indels that do not alter coding sequence

SnpEff(incorrect) – frameshift
Annovar(incorrect) – frameshift
Varant(*correct*) – NoCDSChange
The insertion does not alter the stop codon and thus CDS does not change.



Variants that alters stop codon

SnpEff – StopLoss
Annovar - does not compute mutation type,
Varant(*better precision*) – Ambiguous_StopLoss
Varant's annotation indicates that the stop codon which is altered is present in middle of CDS and not at end thus is ambiguous.



Easy Install

The installation requires 3 major steps (performed automatically by install script) -

1. **Download the data sources**
   Note that **Varant depends upon 17 data sources**.
2. **Create SQLite databases**
   Most of the data sources are converted to high performance SQLite databases and the data are fetched from databases using their respective API during annotation process.
3. **Set path** to the location of data sources and their SQLite databases

Working

For every variant in a VCF file Varant provides **5 categories of annotations** as illustrated in the figure. Each category is supported by their respective data sources. Finally all the annotations are written back to INFO field of VCF file in compliance to the VCF format.

Varant provides tools to update the SQLite database in an automated way. Each SQLite database captures the details of when the database was created or last updated.

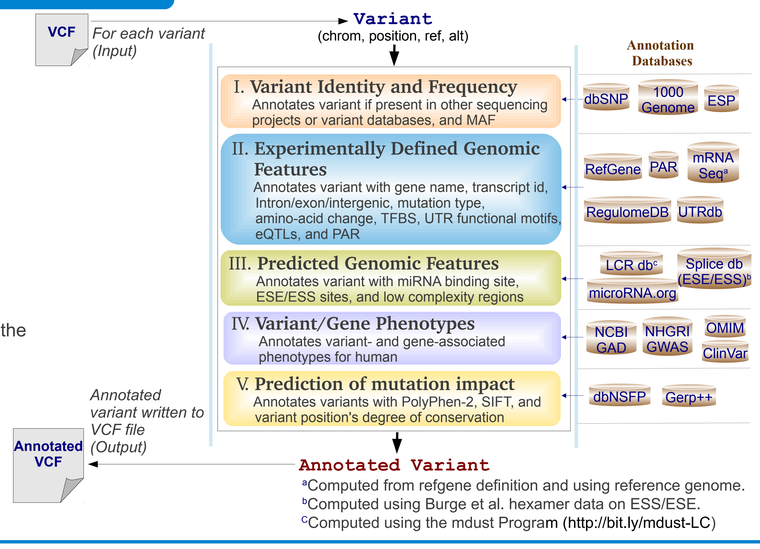
Exensible features

Varant provides following 3 modules that brings out its extensible feature -

Module to add new annotations to a VCF file from VCF file

Module to add new annotations to a VCF file from bed file

Parser for Varant's annotations



[a]Computed from refgene definition and using reference genome.
[b]Computed using Burge et al. hexamer data on ESS/ESE.
[c]Computed using the mdust Program (http://bit.ly/mdust-LC)

Comprehensive Output

All the annotations are written to the INFO field of VCF file in compliance to the VCF file format and can be easily parsed by any VCF parser.
Following are two annotations in VCF file for which Varant provides a specific grammar that integrates multiple annotations based on gene and its transcripts -

1. For the **intergenic variants**, the downstream and upstream genes that overlaps with 5000bp on either side of variant position are reported along with the distance to the genes in following format -

   VARANT_INTERGENIC = UpstreamGene (dist = XYZ), DownstreamGene (dist = XTZ)

2. For the **genic variants**, **transcript based annotations** followed by **gene associated clinical phenotypes** are reported in following format -

   VARANT_GENIC = Gene (Transcript_id | Region | Exon_number | AltId | mRNAPos | SpliceSite | UTRMotif | Mutation | Codon_Change | AminoAcid_Change | Protein_Length | Codon_Usage | SIFT(pred_score) | PolyPhen2(pred_score) | Warning : OMIM_Phenotype : OMIM_Ids : GAD_Phenotype)

   If there is more than one transcripts for the gene, the annotations for the respective transcripts are appended by ':' and finally followed by the clinical phenotype annotations.

EXAMPLES

Intergenic variant which is upstream of DDA1 gene and is associated with a phenotype.

```
#CHROM  POS   ID REF  ALT  QUAL FILTER  INFO
19  17420289 rs2303745 G  T  100.0   PASS VARANT_INTERGENIC=MRPL34(dist=2637):DDA1(dist=48);dbSNPBuildID=100
```

A genic variant which is causing a non-synonymous mutation. The gene is associated with clinical phenotype.

```
#CHROM  POS   ID REF  ALT  QUAL FILTER  INFO
1       9324213 rs17368528      C       T       100.0   PASS    VARANT_GENIC=H6PD(NM_004285|CodingExonic|5|1|1934|||NonSyn|CCG/CTG|P554L|791||D_0.02|
                                                                PP2PD_0.913|:CORTISONE_REDUCTASE_DEFICIENCY_1:604931:
                                                                polycystic_ovary_syndrome);dbSNPBuildID=123
```