Integrated Analysis of Next-Gen Sequencing Data using *Variant Tools*

Bo Peng, Ph.D.

Department of Bioinformatics and Computational Biology
The University of Texas MD Anderson Cancer Center

Sep 20, 2014

Outline

Outline

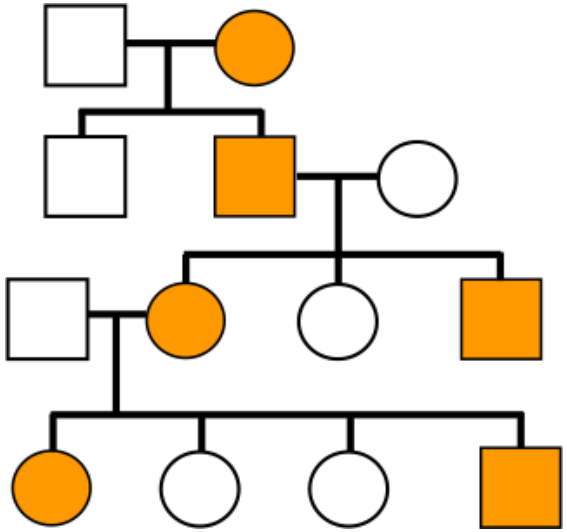
Introduction

Sequencing Analysis: sample collection

Two basic study designs for association analysis:

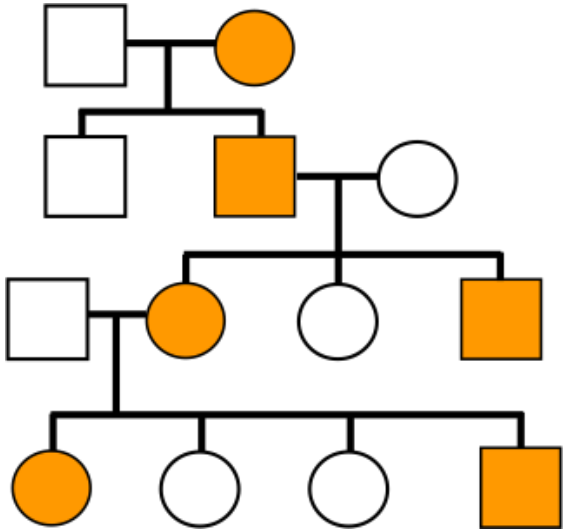
Sequencing Analysis: sample collection

Two basic study designs for association analysis:



Sequencing Analysis: sample collection

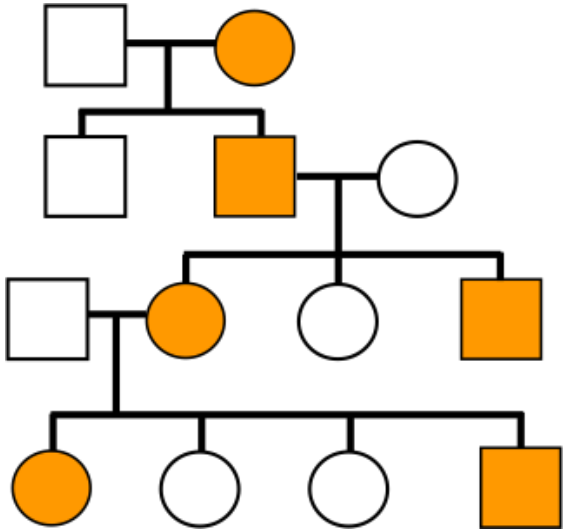
Two basic study designs for association analysis:



Family-based design

Sequencing Analysis: sample collection

Two basic study designs for association analysis:

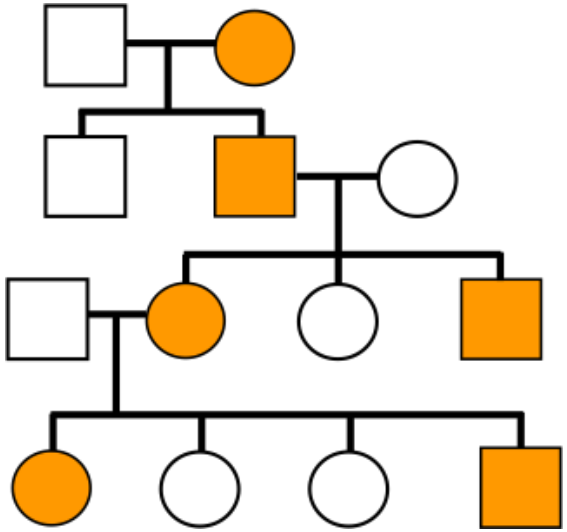


Family-based design

- Detect shared variants within families

Sequencing Analysis: sample collection

Two basic study designs for association analysis:

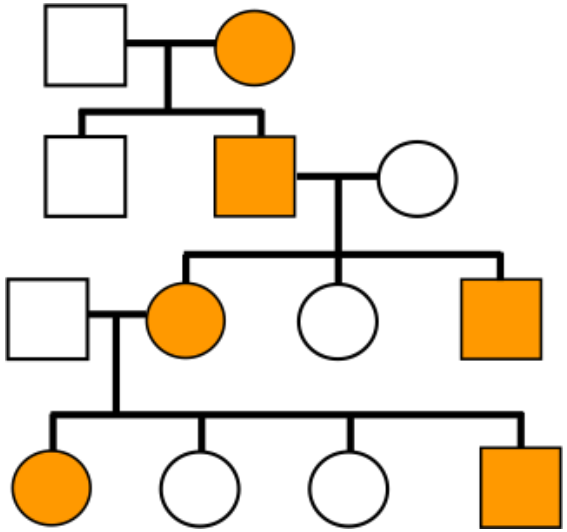


Family-based design

- Detect shared variants within families
- Parents-child trio. sibpairs, large families

Sequencing Analysis: sample collection

Two basic study designs for association analysis:

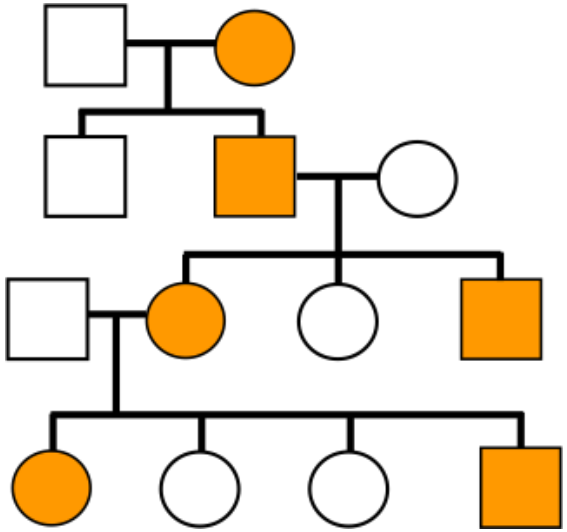


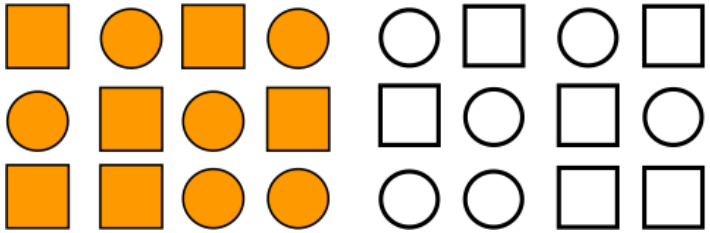
Family-based design

- Detect shared variants within families
- Parents-child trio. sibpairs, large families

Sequencing Analysis: sample collection

Two basic study designs for association analysis:



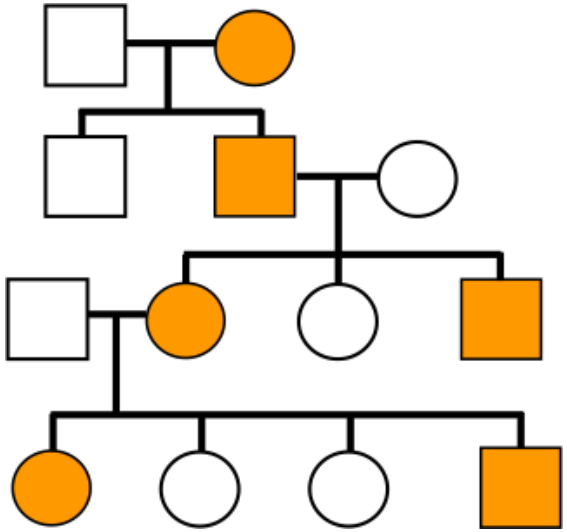


Family-based design

- Detect shared variants within families
- Parents-child trio. sibpairs, large families

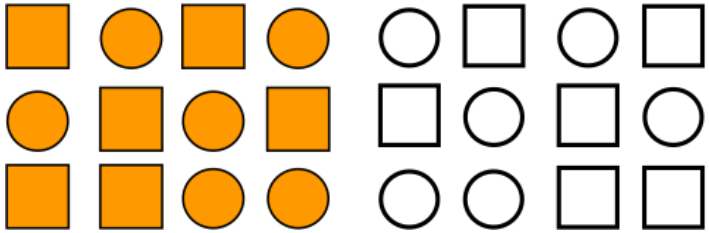
Sequencing Analysis: sample collection

Two basic study designs for association analysis:



Family-based design

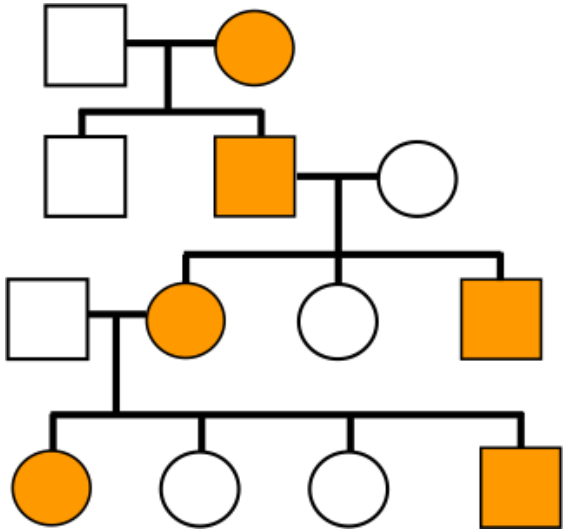
- Detect shared variants within families
- Parents-child trio. sibpairs, large families



Population based

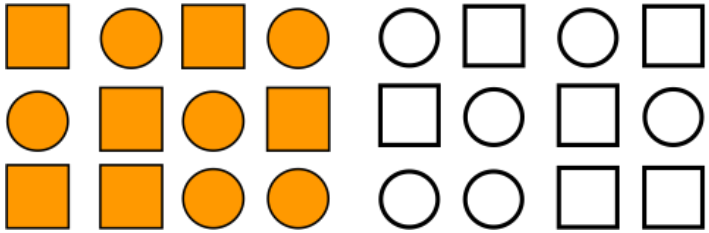
Sequencing Analysis: sample collection

Two basic study designs for association analysis:



Family-based design

- Detect shared variants within families
- Parents-child trio. sibpairs, large families

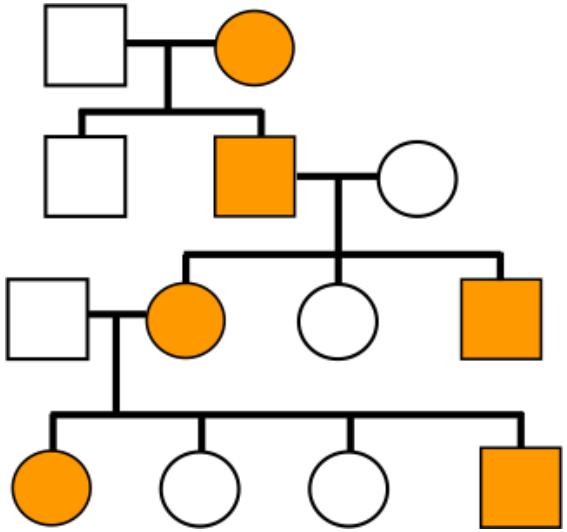


Population based

- Detect shared variants across families

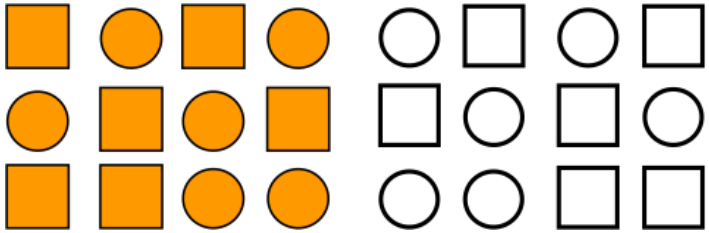
Sequencing Analysis: sample collection

Two basic study designs for association analysis:



Family-based design

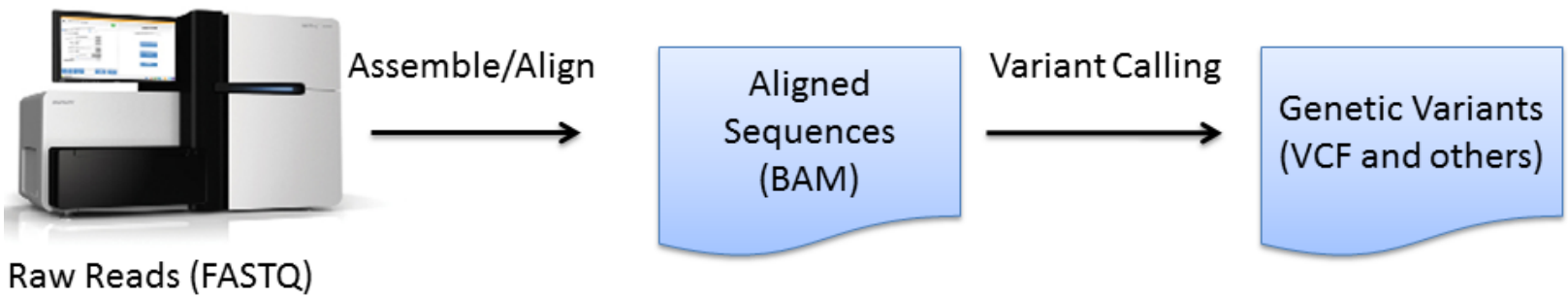
- Detect shared variants within families
- Parents-child trio. sibpairs, large families



Population based

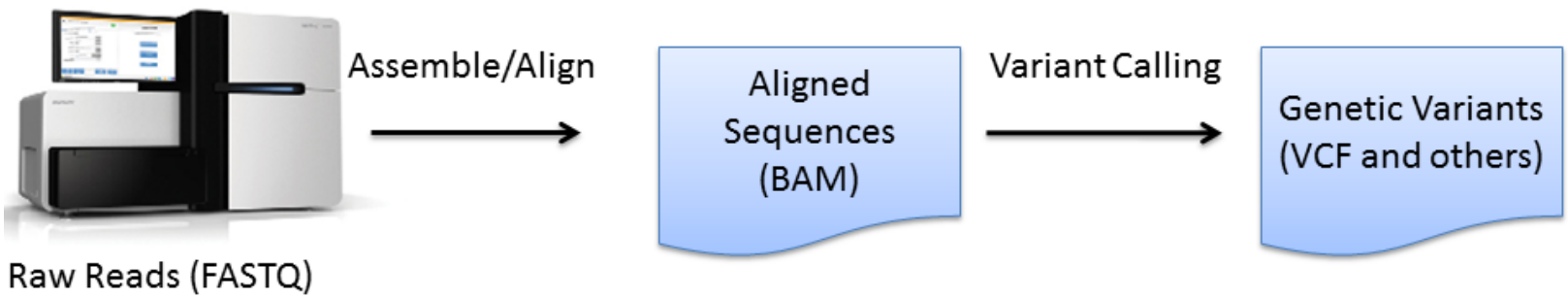
- Detect shared variants across families
- (Matched) case control samples

Sequencing Analysis: variant calling



- Align raw reads from different platforms (Sanger Capillary, Roche 454, Illumina, Applied Biosystems SOLID, Complete Genomcs, Ion Torrent, ...) to a reference genome, using different aligners such as SNAP, iSAAC, NovoAlign, Razers3, bwa, bowtie, STAR, TopHat.

Sequencing Analysis: variant calling



- Align raw reads from different platforms (Sanger Capillary, Roche 454, Illumina, Applied Biosystems SOLID, Complete Genomcs, Ion Torrent, ...) to a reference genome, using different aligners such as SNAP, iSAAC, NovoAlign, Razers3, bwa, bowtie, STAR, TopHat.
- Call small (SNVs, insertions and deletions) and structural variants (difference in the copy number, orientation or location of genomic segments > 100bp) from aligned reads, using variant calling and SV discovery tools such as GATK, CASAVA, BreakDancer, CLEVER, VNCer, PEMer, SLOPE.

Sequencing Analysis: Annotation and Prioritization

- Region: Is a variant in a gene (ref seq gene, known gene, CCDS gene), in exome regions of a gene, in a genomic duplication region?
- Database membership: Is the variant in dbSNP, 1000 genomes, dbNSFP, COSMIC (Catalogue of Somatic Mutations in Cancer), ESP (Exome Sequencing Project), gwas gatalog? Does it belong to any known cancer gene, pathway?
- Functional prediction: Is it predicted to be damaging (SIFT, Polyphen2, LRT, MutationTaster, FATHMM, GERP, PhyloP scores) or in an evolutionarily conserved region (PhastCons)?
- Population statistics: What are the population or sample frequency of the variant?

Sequencing Analysis: Association and Other Analyses

In addition to numerous applications in functional genomics, NGS data have been used to

- Identify De Novo mutations: Identify alterations that are present for the first time in one family member as a result of mutations in a germ cell (egg or sperm) of one of the parents or in the fertilized egg itself.
- Associate genotype to phenotype: Associate variants (for highly penetrant variants for Mendelian diseases) or genes (for complex traits) to qualitative or quantitative traits, using case control or family based study designs.

Design of *Variant Tools*

Variant Tools is a toolkit for the integrated annotation and analysis of genetic variants from next-gen sequencing studies.

Design of *Variant Tools*

Variant Tools is a toolkit for the integrated annotation and analysis of genetic variants from next-gen sequencing studies.

- Project-based design for integrative analysis

DESIGN OF *Variant Tools*

Variant Tools is a toolkit for the integrated annotation and analysis of genetic variants from next-gen sequencing studies.

- Project-based design for integrative analysis
- File format specification system, standardized annotation databases, and support for an alternative reference genome to free users from details about file formats and reference genomes

DESIGN OF *Variant Tools*

Variant Tools is a toolkit for the integrated annotation and analysis of genetic variants from next-gen sequencing studies.

- Project-based design for integrative analysis
- File format specification system, standardized annotation databases, and support for an alternative reference genome to free users from details about file formats and reference genomes
- Unified handling of variant info, annotation and track fields allows easy annotation, selection and reporting of variants according to multiple annotation sources

DESIGN OF *Variant Tools*

Variant Tools is a toolkit for the integrated annotation and analysis of genetic variants from next-gen sequencing studies.

- Project-based design for integrative analysis
- File format specification system, standardized annotation databases, and support for an alternative reference genome to free users from details about file formats and reference genomes
- Unified handling of variant info, annotation and track fields allows easy annotation, selection and reporting of variants according to multiple annotation sources
- vtools_report for routine analyses and pipelines for complex tasks (e.g. variant calling) and interaction with other tools.

DESIGN OF *Variant Tools*

Variant Tools is a toolkit for the integrated annotation and analysis of genetic variants from next-gen sequencing studies.

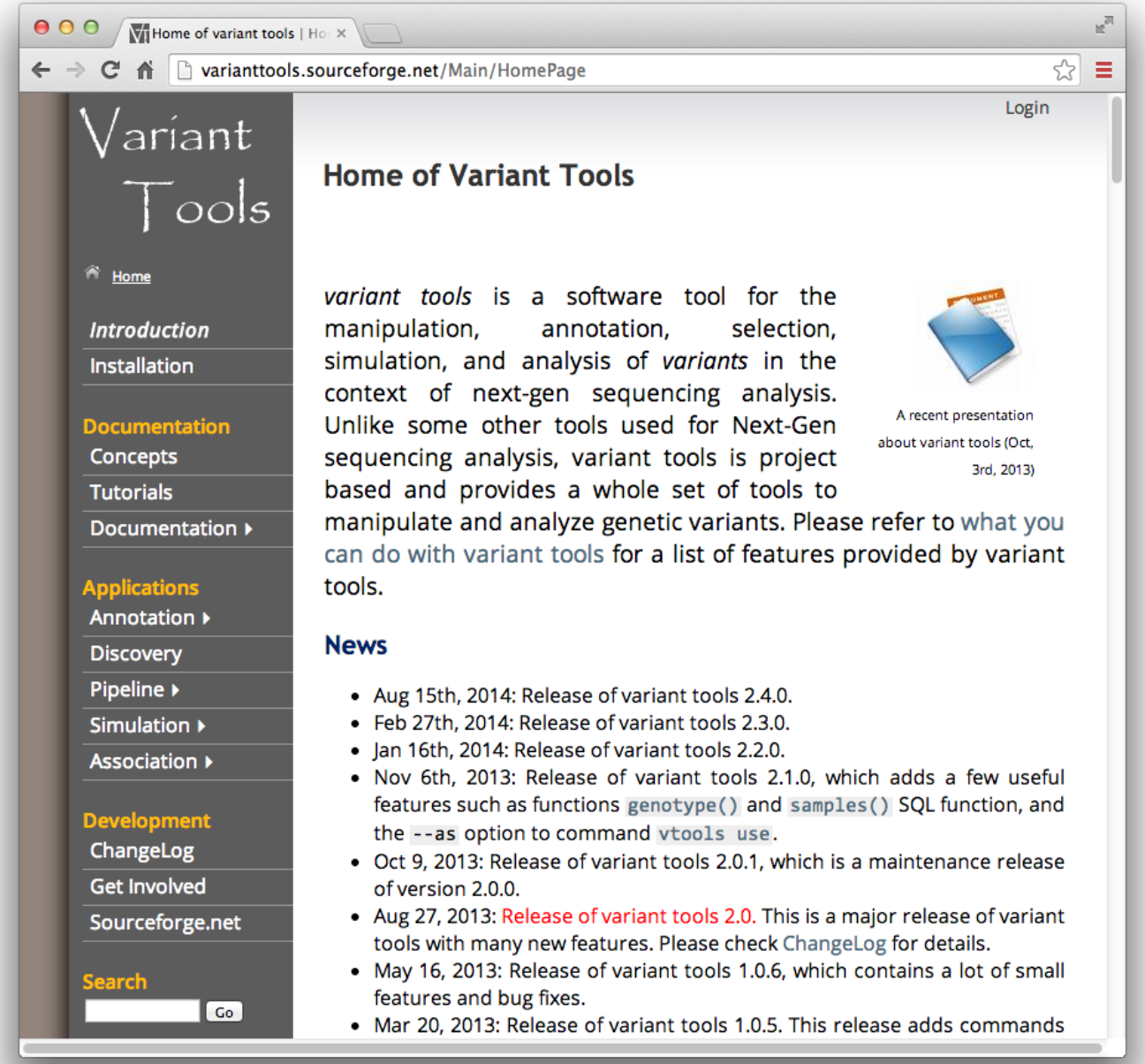
- Project-based design for integrative analysis
- File format specification system, standardized annotation databases, and support for an alternative reference genome to free users from details about file formats and reference genomes
- Unified handling of variant info, annotation and track fields allows easy annotation, selection and reporting of variants according to multiple annotation sources
- vtools_report for routine analyses and pipelines for complex tasks (e.g. variant calling) and interaction with other tools.
- An association analysis framework allows flexible and extensible association analysis

DESIGN OF *Variant Tools*

Variant Tools is a toolkit for the integrated annotation and analysis of genetic variants from next-gen sequencing studies.

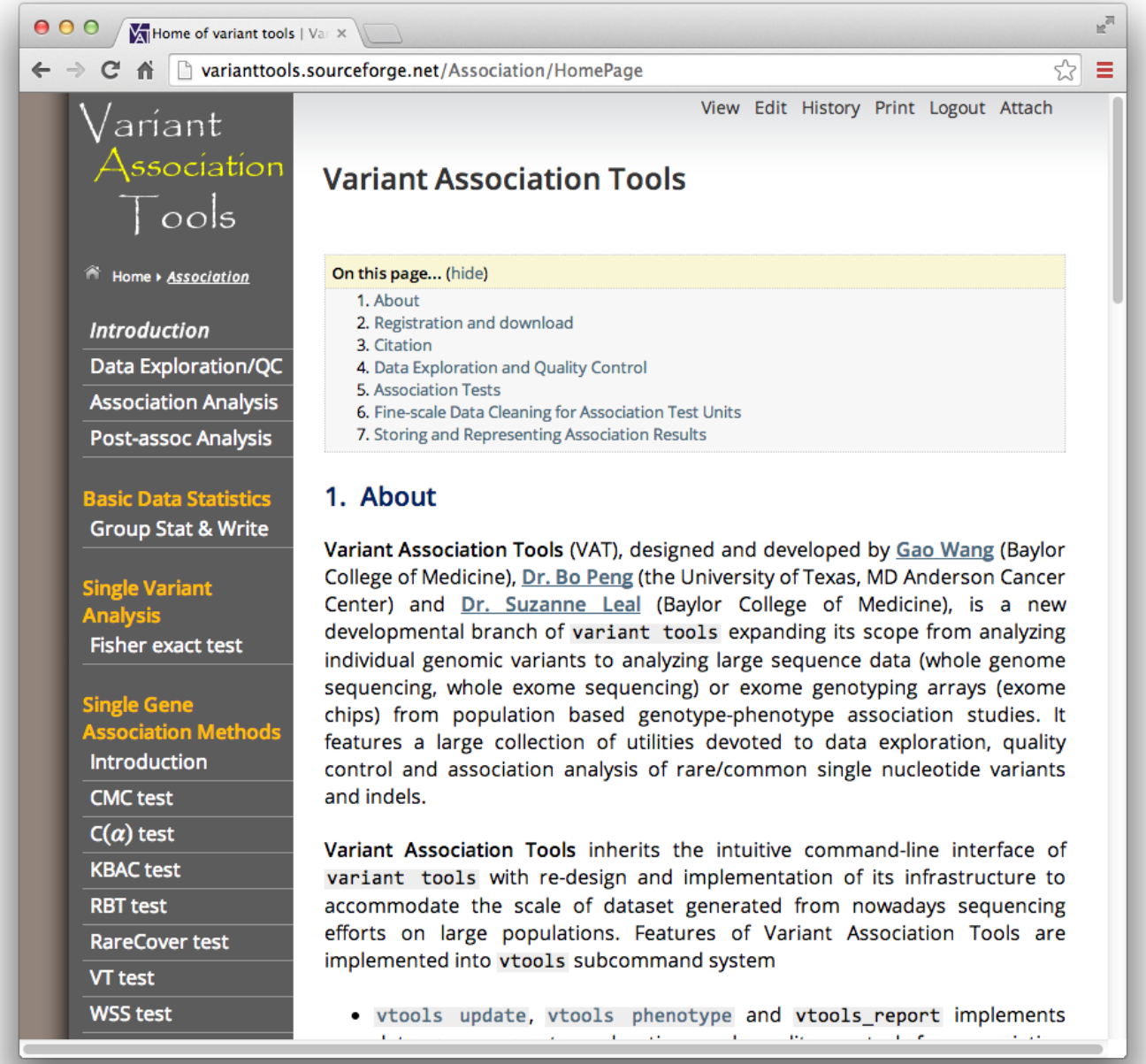
- Project-based design for integrative analysis
- File format specification system, standardized annotation databases, and support for an alternative reference genome to free users from details about file formats and reference genomes
- Unified handling of variant info, annotation and track fields allows easy annotation, selection and reporting of variants according to multiple annotation sources
- vtools_report for routine analyses and pipelines for complex tasks (e.g. variant calling) and interaction with other tools.
- An association analysis framework allows flexible and extensible association analysis
- Online resource repository of annotation databases, file formats, snapshots etc.

Variant Tools / Variant Annotation Tools



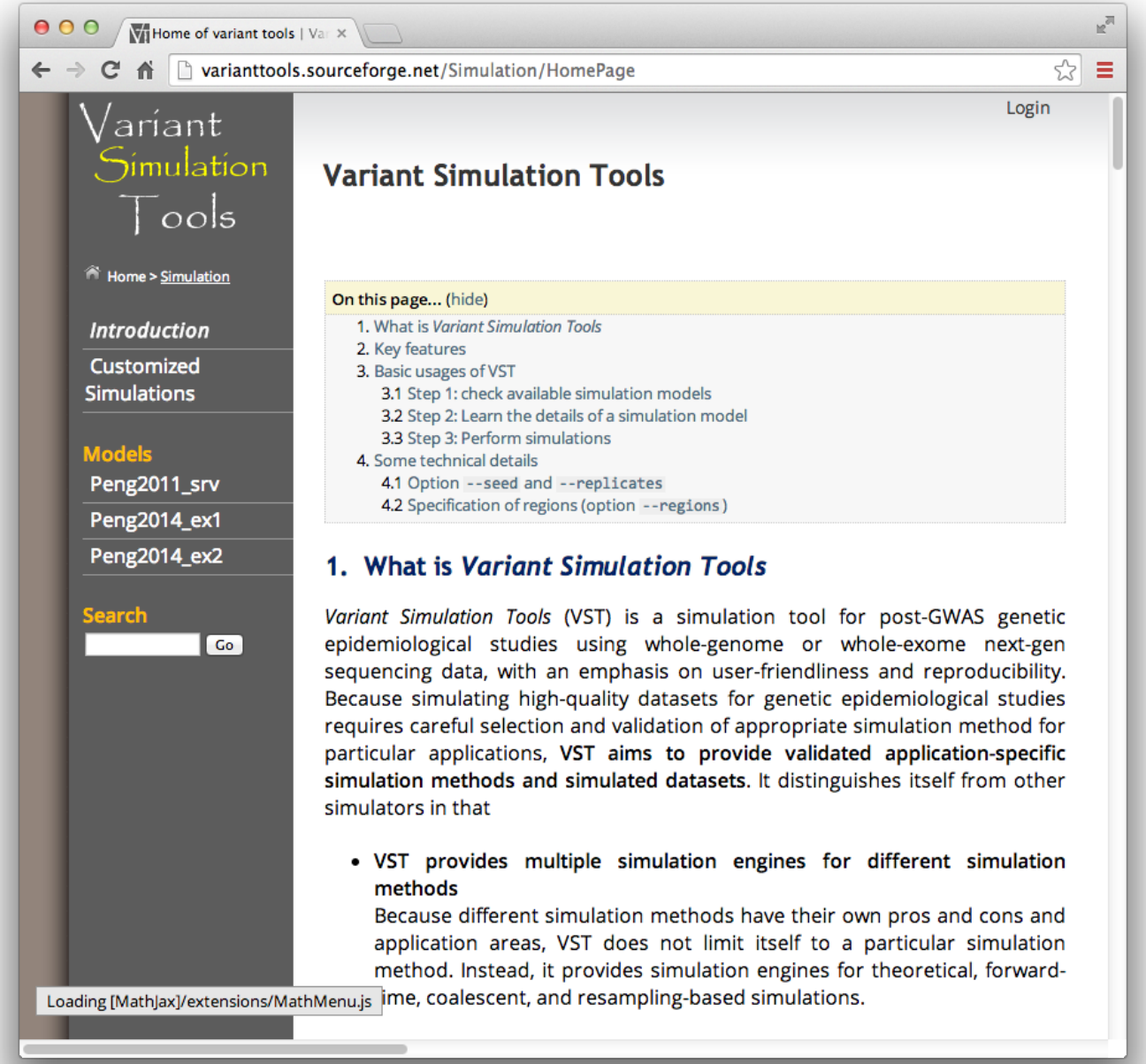
- San Lucas et al, Bioinformatics, 2012
- Import, select, and manage genetic variants
- Annotate variants using various annotation databases
- Pipelines for variant calling, annotation, and other functions

Variant Association Tools (VAT)



- Wang et al, AJHG, 2014
- Detect phenotype-genotype association
- Provide more than 20 rare variant association analysis methods

Variant Simulation Tools (VST)



- Peng, Genet. Epidemio. 2014
- Simulate realistic genotype and phenotype data for sequencing analysis
- Use forward-time, coalescent and resampling based methods

Outline

Basic concepts

VARIANT AND VARIANT TABLE

A variant refers to a mutation from `ref` to `alt` at `pos` of `chr`. A variant in *variant tools* can be SNV, small indel, or MNPs (Multiple-nucleotide polymorphism). All variants are assumed to be on the forward (+) strand.

```
$ vtools show tables
table      #variants     date message
variant        4,858    Oct02 Master variant table

$ vtools output variant chr pos ref alt --limit 5
1  1105366  T  C
1  1105411  G  A
1  1108138  C  T
1  1110240  T  A
1  1110294  G  A

$ vtools select variant 'ref="T"' --to_table refT 'variants with reference allele T'
Running: 2 846.4/s in 00:00:00
INFO: 787 variants selected.

$ vtools show tables
table      #variants     date message
refT             787    Oct02 variants with reference allele T
variant        4,858    Oct02 Master variant table

$ vtools output refT chr pos ref alt -l 5
1  1105366  T  C
1  1110240  T  A
1  3537996  T  C
1  6447088  T  C
1  6447275  T  C
```

VARIANT AND VARIANT TABLE

A variant refers to a mutation from `ref` to `alt` at `pos` of `chr`. A variant in *variant tools* can be SNV, small indel, or MNPs (Multiple-nucleotide polymorphism). All variants are assumed to be on the forward (+) strand.

```
$ vtools show tables
table       #variants     date message
variant         4,858    Oct02 Master variant table

$ vtools output varian
1  1105366  T  C
1  1105411  G  A
1  1108138  C  T
1  1110240  T  A
1  1110294  G  A

$ vtools select varian                                    e T'
Running: 2 846.4/s in
INFO: 787 variants sel

$ vtools show tables
table       #variants     date message
refT              787    Oct02 variants with reference allele T
variant         4,858    Oct02 Master variant table

$ vtools output refT chr pos ref alt -l 5
1  1105366  T  C
1  1110240  T  A
1  3537996  T  C
1  6447088  T  C
1  6447275  T  C
```

Variant Tools does not yet support large indels and structural variants such as inversions.

VARIANT INFO FIELD

Variant info fields provide annotation information for each variant. They are maintained inside the project.

```
$ vtools show fields
variant.chr
variant.pos
variant.ref
variant.alt
variant.AA
variant.DP

$ vtools output refT chr pos ref alt AA DP -l 5
1  1105366  T  C  T  3251
1  1110240  T  A  T  7275
1  3537996  T  C  C  1753
1  6447088  T  C  T  4691
1  6447275  T  C  T  6871

$ vtools update variant --from_file CEU.exon.2010_03.sites.vcf.gz --var_info id
INFO: Using primary reference genome hg18 of the project.
Getting existing variants: 100% [========================] 3,188 231.4K/s in 00:00:00
INFO: Updating variants from CEU.exon.2010_03.sites.vcf.gz (1/1)
CEU.exon.2010_03.sites.vcf.gz: 100% [======================] 3,500 8.4K/s in 00:00:00
INFO: Field id of 1,531 variants are updated

$ vtools output refT chr pos ref alt id AA DP -l 5
1  1105366  T  C  .           T  3251
1  1110240  T  A  .           T  7275
1  3537996  T  C  rs2760321   C  1753
1  6447088  T  C  rs11800462  T  4691
1  6447275  T  C  rs3170675   T  6871
```

Reference genome

A variant can have different chromosomal coordinates in different reference genomes. It is extremely important to know the reference genome used for your project.

```
$ vtools output variant chr pos ref alt 'ref_sequence(chr, pos, pos+5)' -l 5
1  1105366  T  C  TGTGGG
1  1105411  G  A  GGACCC
1  1108138  C  T  CAAGCC
1  1110240  T  A  TGCTGC
1  1110294  G  A  GTGACA

$ vtools liftover hg19
INFO: Downloading liftOver chain file from UCSC
INFO: Exporting variants in BED format
Exporting variants: 100% [==============] 4,858 129.0K/s in 00:00:00
INFO: Running UCSC liftOver tool
Updating table variant: 100% [===========] 4,858 28.4K/s in 00:00:00

$ vtools output variant chr pos ref alt 'ref_sequence(chr, pos, pos+5)' -l 5 --build hg19
1  1115503  T  C  TGTGGG
1  1115548  G  A  GGACCC
1  1118275  C  T  CAAGCC
1  1120377  T  A  TGCTGC
1  1120431  G  A  GTGACA
```

Annotation Database

Variant tools supports four types of annotation databases:

- Variant: Annotate specific variant (`chr`, `pos`, `ref`, `alt`)
  dbNSFP, dbSNP, 1000 genomes
- Position: Annotate chromosomal position (`chr`, `pos`)
  gwasCatalog
- Range: Annotate regions (`chr`, `start`, `end`)
  refGene, knownGene, ccdsGene
  refGene_exon, knownGene_exon, ccdsGene_exon
- Attribute: Annotate attribute of variants (e.g. gene)
  keggPathway, Cancer Gene Census

Annotation databases are defined by `.ann` files. Database files (`.DB.gz`) are automatically downloaded from `http://vtools.houstonbioinformatics.org`.

Track

Track files provide additional annotation information to variants (e.g. info fields in vcf files) or positions (e.g. alignment information at positions).

```
$ vtools output refT chr pos ref alt "track('CEU.exon.2010_03.sites.vcf.gz', 'info.AA')" -l5
1  1105366  T  C  T
1  1110240  T  A  T
1  3537996  T  C  C
1  6447088  T  C  T
1  6447275  T  C  T

$ vtools select variant "track('CEU.exon.2010_03.sites.vcf.gz', 'info.DP') > 1000" --output chr
     pos ref alt DP -l5
1  1105366  T  C  3251
1  1105411  G  A  2676
1  1108138  C  T  2253
1  1110240  T  A  7275
1  1110294  G  A  7639

$ vtools liftover hg19
$ vtools output variant chr pos "track('http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release
     /20110521/ALL.chr1.phase1_release_v3.20101123.snps_indels_svs.genotypes.vcf.gz', 'info')" --
     build hg19 -l 5
[get_local_version] downloading the index file...
1  1115503  LDAF=0.0133;AC=28;SNPSOURCE=LOWCOV,EXOME;AA=T;AN=2184;VT=SNP;THETA=0.0012;ERATE
     =0.0003;RSQ=0.9950;AVGPOST=0.9999;AF=0.01;AMR_AF=0.01;AFR_AF=0.0041;EUR_AF=0.03
1  1115548  AVGPOST=0.9983;THETA=0.0004;SNPSOURCE=LOWCOV,EXOME;AA=G;AN=2184;RSQ=0.9326;LDAF
     =0.0106;VT=SNP;AC=22;ERATE=0.0006;AF=0.01;AMR_AF=0.02;EUR_AF=0.02
1  1118275  AC=300;AA=C;THETA=0.0004;SNPSOURCE=LOWCOV,EXOME;AN=2184;AVGPOST=0.9981;LDAF=0.1372;VT=
     SNP;ERATE=0.0008;RSQ=0.9950;AF=0.14;ASN_AF=0.05;AMR_AF=0.14;AFR_AF=0.38;EUR_AF=0.04
1  1120377  THETA=0.0009;SNPSOURCE=LOWCOV,EXOME;AA=T;AN=2184;RSQ=0.9796;AC=16;AVGPOST=0.9996;VT=
     SNP;LDAF=0.0072;ERATE=0.0003;AF=0.01;AMR_AF=0.01;EUR_AF=0.02
1  1120431  AC=347;THETA=0.0096;ERATE=0.0063;AVGPOST=0.9977;RSQ=0.9945;SNPSOURCE=LOWCOV,EXOME;AN
```

SNAPSHOT

A *snapshot* contains a copy of all databases of a project. Local snapshots are used to save, restore, and transfer projects. Online snapshots are used extensively in documentation.

```
$ vtools admin --save_snapshot con1 'first snapshot for project concept'
INFO: Snapshot con1 has been saved

$ vtools show snapshots
con1                    first snapshot for project concept  (358.0KB, created:
                        Oct03 01:01:07)
vt_qc                   snapshot for QC tutorial, exome data of 1000 genomes
                        project with simulated GD and GQ scores (2.0GB, online
                        snapshot)
vt_ExomeAssociation     Data with ~26k variants from chr1 and 2, ~3k samples,
                        3 phenotypes, ready for association testing. (446.0MB,
                        online snapshot)
vt_quickStartGuide      A simple project with variants from the CEU and JPT
                        pilot data of the 1000 genome project (148.0KB, online
                        snapshot)
vt_illuminaTestData     Test data with 1M paired reads (49.0MB, online
                        snapshot)
vt_simple               A simple project with variants imported from three vcf
                        files (41.0KB, online snapshot)
vt_testData             An empty project with some test datasets (68.0KB,
                        online snapshot)

$ vtools admin --load_snapshot vt_testData
Downloading snapshot vt_testData.tar.gz from online
INFO: Snapshot vt_testData has been loaded
```

Sample, genotype and genotype info fields

A *sample* contains a list of variants, their number (0 for homozygote reference, 1 for heterozygote and 2 for homozygote alternative), and additional info (e.g. depth of coverage) detected from a physical sample.

```
$ vtools import CEU.vcf.gz --build hg18 --var_info DP --geno_info DP_geno
INFO: Importing variants from CEU.vcf.gz (1/1)
CEU.vcf.gz: 100% [============================] 300 12.5K/s in 00:00:00
INFO: 0 new variants 288 SNVs from 300 lines are imported.
Importing genotypes: 100% [=================] 18,000 9.0K/s in 00:00:02
Copying samples: 100% [=========================] 65 64.9/s in 00:00:01

$ vtools show genotypes -l 5
sample_name  filename    num_genotypes  sample_genotype_fields
NA06985      CEU.vcf.gz  287            GT,DP_geno
NA06986      CEU.vcf.gz  287            GT,DP_geno
NA06994      CEU.vcf.gz  287            GT,DP_geno
NA07000      CEU.vcf.gz  287            GT,DP_geno
NA07037      CEU.vcf.gz  287            GT,DP_geno
(55 records omitted)
```

Phenotype

Phenotypes are arbitrary properties of samples.

```
$ head -8 phenotype.txt
sample_name     aff     sex     BMI
NA06985 2       F       19.64
NA06986 1       M       None
NA06994 1       F       19.49
NA07000 2       F       21.52
NA07037 2       F       23.05
NA07051 1       F       21.01
NA07346 1       F       18.93

$ vtools phenotype --from_file phenotype.txt
INFO: Adding phenotype aff
INFO: Adding phenotype sex
INFO: Adding phenotype BMI
INFO: 3 field (3 new, 0 existing) phenotypes of 60 samples are updated.

$ vtools show phenotypes -l 8
sample_name  aff  sex  BMI
NA06985      2    F    19.64
NA06986      1    M    None
NA06994      1    F    19.49
NA07000      2    F    21.52
NA07037      2    F    23.05
NA07051      1    F    21.01
NA07346      1    F    18.93
NA07347      2    M    19.2
(50 records omitted)
```

Outline

Format of indel data

```
$ head -30 MG3037-121.pileup.indel
chr10   51372   D1      A       *       hete    25      9       33
chr10   57161   D2      AG      *       hete    33      3       21
chr10   57414   I1      G       *       hete    21      2       20
chr10   62170   I1      T       *       hete    36      10      30
chr10   62899   I3      AAA     *       hete    38      9       30
chr10   66586   D1      A       *       hete    22      5       31
chr10   85429   I1      A       *       hete    53      10      26
chr10   86294   I4      CAGC    *       hete    46      4       35
chr10   87126   I24     TGCATTTACGTGATCTTGGCTCAC        *       hete    55      8       53
chr10   88705   I1      A       *       hete    53      10      55
chr10   89448   I3      AGG     *       hete    29      5       39
chr10   93591   D1      G       *       hete    40      6       33
chr10   93753   D1      T       *       hete    29      19      79
chr10   94117   I3      CAA     *       hete    27      38      106
chr10   97572   D1      T       *       hete    40      8       51
chr10   97938   D1      T       *       hete    32      29      65
chr10   98719   I1      T       *       hete    47      10      38
chr10   100799  I1      G       *       hete    47      10      36
chr10   101382  D1      G       *       hete    53      13      36
chr10   102510  D1      C       *       hete    52      8       38
chr10   103093  D1      T       *       hete    53      23      41
chr10   106216  D4      TTTT    *       hete    53      15      35
chr10   106509  I13     TGGCCAGGCACAG   *       hete    49      3       29
chr10   107368  D1      T       *       hete    51      5       27
chr10   108915  I1      G       *       hete    54      12      31
chr10   110337  D2      GG      *       hete    55      2       18
chr10   110565  D1      A       *       hete    45      4       15
```

Format of indel data

```
$ head -30 MG3037-121.pileup.indel
chr10   51372   D1      A       *       hete
chr10   57161   D2      AG      *       hete
chr10   57414   I1      G       *       hete
chr10   62170   I1      T       *       hete
chr10   62899   I3      AAA     *       hete
chr10   66586   D1      A       *       hete
chr10   85429   I1      A       *       hete    53      10      26
chr10   86294   I4      CAGC    *       hete    46      4       35
chr10   87126   I24     TGCATTTACGTGATCTTGGCTCAC        *       hete    55      8       53
chr10   88705   I1      A       *       hete    53      10      55
chr10   89448   I3      AGG     *       hete    29      5       39
chr10   93591   D1      G       *       hete    40      6       33
chr10   93753   D1      T       *       hete    29      19      79
chr10   94117   I3      CAA     *       hete    27      38      106
chr10   97572   D1      T       *       hete    40      8       51
chr10   97938   D1      T       *       hete    32      29      65
chr10   98719   I1      T       *       hete    47      10      38
chr10   100799  I1      G       *       hete    47      10      36
chr10   101382  D1      G       *       hete    53      13      36
chr10   102510  D1      C       *       hete    52      8       38
chr10   103093  D1      T       *       hete    53      23      41
chr10   106216  D4      TTTT    *       hete    53      15      35
chr10   106509  I13     TGGCCAGGCACAG   *       hete    49      3       29
chr10   107368  D1      T       *       hete    51      5       27
chr10   108915  I1      G       *       hete    54      12      31
chr10   110337  D2      GG      *       hete    55      2       18
chr10   110565  D1      A       *       hete    45      4       15
```

Variant tools provides an input format specification system that allows processing data in arbitrary delimiter separated formats.

Input format specification

```
$ vtools show formats -v0
CASAVA18_snps
CASAVA18_indels
plink
rsname
ANNOVAR
pileup_indel
ANNOVAR_exonic_variant_function
ANNOVAR_variant_function
twoalleles
map
polyphen2
basic
vcf
CGA
csv
tped

$ vtools show format pileup_indel
Input format for samtools pileup indel caller. This format imports chr, pos,
ref, alt and genotype.

Columns:
  None defined, cannot export to this format

variant:
  chr                   Chromosome name
  pos                   Start position of the indel event.
  ref                   reference allele, '-' for insertion
  alt                   alternative allele, '-' for deletion

Genotype:
  GT                    type of indel (homozygote or heterozygote)
```

Import indel data

```
$ vtools import --format pileup_indel MG*.indel
INFO: Opening project RA.proj
INFO: Using primary reference genome hg18 of the project.
Getting existing variants: 100.0% [==============================>] 6,901,157 162.2K/s in 00:00:42
INFO: Additional genotype fields: genotype
INFO: Importing genotype from ../data/indel/MG1000-240.pileup.indel (1/5)
MG1000-240.pileup.indel: 100.0% [===================================>] 712,688 9.2K/s in 00:01:17
INFO: 847,949 new variants from 847,949 records are imported, with 0 SNVs, 348,266 insertions,
      499,683 deletions, and 0 complex variants.
INFO: Importing genotype from ../data/indel/MG1004-200.pileup.indel (2/5)
MG1004-200.pileup.indel: 100.0% [===================================>] 706,906 10.8K/s in 00:01:05
INFO: 416,517 new variants from 836,944 records are imported, with 0 SNVs, 161,927 insertions,
      254,590 deletions, and 0 complex variants.
INFO: Importing genotype from ../data/indel/MG1022-121.pileup.indel (3/5)
MG1022-121.pileup.indel: 100.0% [===================================>] 758,880 11.8K/s in 00:01:04
INFO: 314,641 new variants from 857,899 records are imported, with 0 SNVs, 117,506 insertions,
      197,135 deletions, and 0 complex variants.
INFO: Importing genotype from ../data/indel/MG1057-203.pileup.indel (4/5)
MG1057-203.pileup.indel: 100.0% [===================================>] 676,350 11.2K/s in 00:01:00
INFO: 207,950 new variants from 798,406 records are imported, with 0 SNVs, 79,766 insertions,
      128,184 deletions, and 0 complex variants.
INFO: Importing genotype from ../data/indel/MG1078-200.pileup.indel (5/5)
MG1078-200.pileup.indel: 100.0% [===================================>] 709,018 11.7K/s in 00:01:00
INFO: 191,135 new variants from 842,633 records are imported, with 0 SNVs, 72,772 insertions,
      118,363 deletions, and 0 complex variants.
INFO: 1,978,192 new variants from 4,183,831 records in 5 files are imported, with 0 SNVs, 780,237
      insertions, 1,197,955 deletions, and 0 complex variants.
INFO: Creating index on master variant table. This might take quite a while.
```

Rename samples

```
$ vtools admin --rename_samples "filename like 'MG3037%'" MG3037
INFO: 2 samples with names , SAMP1 are renamed to MG3037
$ vtools admin --rename_samples "filename like 'MG3046%'" MG3046
INFO: 2 samples with names , SAMP1 are renamed to MG3046
$ vtools admin --rename_samples "filename like 'MG3087%'" MG3087
INFO: 2 samples with names , SAMP1 are renamed to MG3087
$ vtools admin --rename_samples "filename like 'MG3140%'" MG3140
INFO: 2 samples with names , SAMP1 are renamed to MG3140
$ vtools admin --rename_samples "filename like 'MG3184%'" MG3184
INFO: 2 samples with names , SAMP1 are renamed to MG3184

$ vtools show samples
sample_name                filename
MG3037                     MG3037-121.snp.txt.vcf
MG3037                     MG3037-121.pileup.indel
MG3046                     MG3046-303.snp.txt.vcf
MG3046                     MG3046-303.pileup.indel
MG3087                     MG3087-200.snp.txt.vcf
MG3087                     MG3087-200.pileup.indel
MG3140                     MG3140-300.snp.txt.vcf
MG3140                     MG3140-300.pileup.indel
MG3184                     MG3184-301.snp.txt.vcf
MG3184                     MG3184-301.pileup.indel
SRR028961.aln.sorted.bam   varSRR028961.filtered.vcf
SRR028962.aln.sorted.bam   varSRR028962.filtered.vcf
SRR028963.aln.sorted.bam   varSRR028963.filtered.vcf
SRR028964.aln.sorted.bam   varSRR028964.filtered.vcf
SRR028965.aln.sorted.bam   varSRR028965.filtered.vcf
```

Merge samples

```
$ vtools admin --merge_samples
INFO: 10 samples that share identical names will be merged to 5 samples
Merging samples: 100% [==================================================] 10 0.5/s in 00:00:21
Removing obsolete tables: 100% [=========================================] 10 8.6/s in 00:00:01

$ vtools show samples
sample_name                filename
MG3037                     MG3037-1...21.snp.txt.vcf
MG3046                     MG3046-3...03.snp.txt.vcf
MG3087                     MG3087-2...00.snp.txt.vcf
MG3140                     MG3140-3...00.snp.txt.vcf
MG3184                     MG3184-3...01.snp.txt.vcf
SRR028961.aln.sorted.bam   varSRR028961.filtered.vcf
SRR028962.aln.sorted.bam   varSRR028962.filtered.vcf
SRR028963.aln.sorted.bam   varSRR028963.filtered.vcf
SRR028964.aln.sorted.bam   varSRR028964.filtered.vcf
SRR028965.aln.sorted.bam   varSRR028965.filtered.vcf

$ vtools admin --save_snapshot imported_data 'Imported data, SNVs and INDELs from samples are
     merged'
INFO: Snapshot imported_data has been saved
```

Merge samples

```
$ vtools admin --merge_samples
INFO: 10 samples that share identical names wi
Merging samples: 100% [=======================
Removing obsolete tables: 100% [==============

$ vtools show samples
sample_name                filename
MG3037                     MG3037-1...21.snp.txt.vcf
MG3046                     MG3046-3...03.snp.txt.vcf
MG3087                     MG3087-2...00.snp.txt.vcf
MG3140                     MG3140-3...00.snp.txt.vcf
MG3184                     MG3184-3...01.snp.txt.vcf
SRR028961.aln.sorted.bam   varSRR028961.filtered.vcf
SRR028962.aln.sorted.bam   varSRR028962.filtered.vcf
SRR028963.aln.sorted.bam   varSRR028963.filtered.vcf
SRR028964.aln.sorted.bam   varSRR028964.filtered.vcf
SRR028965.aln.sorted.bam   varSRR028965.filtered.vcf

$ vtools admin --save_snapshot imported_data 'Imported data, SNVs and INDELs from samples are
     merged'
INFO: Snapshot imported_data has been saved
```

It is a good practice to save snapshots of your project after the completion of major tasks, or before experimental processing steps.

Counting number of variants in samples

Command `vtools update` adds or updates variant info fields. This example uses special functions `#(alt)`, `#(hom)` and `#(het)` to count the number of variants, homozygotes and heterozygotes for each variant in the sample.

```
$ vtools update variant --from_stat 'num=#(alt)' 'hom=#(hom)' 'het=#(het)'
Counting variants: 100% [=====================================] 15 0.2/s in 00:01:100
INFO: Adding variant info field num
INFO: Adding variant info field hom
INFO: Adding variant info field het
Updating variant: 100% [================================] 8,904,873 45.5K/s in 00:03:15
INFO: 8904873 records are updated

$ vtools output variant chr pos ref alt num hom het -l 10
1  583   G  A  5  0  5
1  4770  A  G  5  0  5
1  5931  T  C  4  1  2
1  5966  T  G  6  1  4
1  6120  G  C  2  0  2
1  6241  T  C  4  1  2
1  6360  A  G  2  0  2
1  7401  C  A  1  0  1
1  9131  C  T  2  0  2
1  9992  C  T  3  0  3
```

Count genotypes in cases

```
$ vtools show samples
sample_name                filename
MG3037                     MG3037-1...21.snp.txt.vcf
MG3046                     MG3046-3...03.snp.txt.vcf
MG3087                     MG3087-2...00.snp.txt.vcf
MG3140                     MG3140-3...00.snp.txt.vcf
MG3184                     MG3184-3...01.snp.txt.vcf
SRR028961.aln.sorted.bam  varSRR028961.filtered.vcf
SRR028962.aln.sorted.bam  varSRR028962.filtered.vcf
SRR028963.aln.sorted.bam  varSRR028963.filtered.vcf
SRR028964.aln.sorted.bam  varSRR028964.filtered.vcf
SRR028965.aln.sorted.bam  varSRR028965.filtered.vcf

$ vtools update variant --from_stat 'case_num=#(alt)' --samples 'sample_name like "%MG%"'
INFO: 5 samples are selected
Counting variants: 100% [=======================] 10 0.1/s in 00:01:24
INFO: Adding variant info field case_num
Updating variant: 100% [================] 8,851,542 48.7K/s in 00:03:01
INFO: 8851542 records are updated

$ vtools output variant chr pos ref alt num case_num  -l 5
1  583   G  A  5  5
1  4770  A  G  5  5
1  5931  T  C  4  4
1  5966  T  G  6  6
1  6120  G  C  2  2
```

Count genotypes in cases

```
$ vtools show samples
sample_name                filename
MG3037                     MG3037-1...21.snp.tx
MG3046                     MG3046-3...03.snp.tx
MG3087                     MG3087-2...00.snp.tx
MG3140                     MG3140-3...00.snp.txt.vcf
MG3184                     MG3184-3...01.snp.txt.vcf
SRR028961.aln.sorted.bam   varSRR028961.filtered.vcf
SRR028962.aln.sorted.bam   varSRR028962.filtered.vcf
SRR028963.aln.sorted.bam   varSRR028963.filtered.vcf
SRR028964.aln.sorted.bam   varSRR028964.filtered.vcf
SRR028965.aln.sorted.bam   varSRR028965.filtered.vcf

$ vtools update variant --from_stat 'case_num=#(alt)' --samples 'sample_name like "%MG%"'
INFO: 5 samples are selected
Counting variants: 100% [=======================] 10 0.1/s in 00:01:24
INFO: Adding variant info field case_num
Updating variant: 100% [================] 8,851,542 48.7K/s in 00:03:01
INFO: 8851542 records are updated

$ vtools output variant chr pos ref alt num case_num  -l 5
1  583   G  A  5  5
1  4770  A  G  5  5
1  5931  T  C  4  4
1  5966  T  G  6  6
1  6120  G  C  2  2
```

Samples can be selected by sample names, file names, and arbitrary phenotypes.

Add phenotype

```
$ vtools show samples
sample_name                filename
MG3037                     MG3037-1...21.snp.txt.vcf
MG3046                     MG3046-3...03.snp.txt.vcf
MG3087                     MG3087-2...00.snp.txt.vcf
MG3140                     MG3140-3...00.snp.txt.vcf
MG3184                     MG3184-3...01.snp.txt.vcf
SRR028961.aln.sorted.bam  varSRR028961.filtered.vcf
SRR028962.aln.sorted.bam  varSRR028962.filtered.vcf
SRR028963.aln.sorted.bam  varSRR028963.filtered.vcf
SRR028964.aln.sorted.bam  varSRR028964.filtered.vcf
SRR028965.aln.sorted.bam  varSRR028965.filtered.vcf

$ vtools phenotype --set aff=2 --samples "sample_name like '%MG%'"
INFO: Adding phenotype aff
INFO: 10 values of 1 phenotypes (1 new, 0 existing) of 10 samples are updated.

$ vtools phenotype --set aff=1 --samples 'aff is NULL'
INFO: 5 values of 1 phenotypes (0 new, 1 existing) of 5 samples are updated.
```

Allele count by affection status

```
$ vtools show samples
sample_name                filename                    aff
MG3037                     MG3037-1...21.snp.txt.vcf  2
MG3046                     MG3046-3...03.snp.txt.vcf  2
MG3087                     MG3087-2...00.snp.txt.vcf  2
MG3140                     MG3140-3...00.snp.txt.vcf  2
MG3184                     MG3184-3...01.snp.txt.vcf  2
SRR028961.aln.sorted.bam  varSRR028961.filtered.vcf  1
SRR028962.aln.sorted.bam  varSRR028962.filtered.vcf  1
SRR028963.aln.sorted.bam  varSRR028963.filtered.vcf  1
SRR028964.aln.sorted.bam  varSRR028964.filtered.vcf  1
SRR028965.aln.sorted.bam  varSRR028965.filtered.vcf  1

$ vtools update variant --from_stat 'ctrl_num=#(alt)' --samples 'aff=1'
INFO: 5 samples are selected
Counting variants: 100% [=========================] 5 4.6/s in 00:00:01
INFO: Adding variant info field ctrl_num
Updating variant: 100% [==================] 171,861 22.5K/s in 00:00:07
INFO: 171861 records are updated

$ vtools output variant chr pos ref alt num case_num ctrl_num  -l 5
1  583   G  A  5  5  0
1  4770  A  G  5  5  0
1  5931  T  C  4  4  0
1  5966  T  G  6  6  0
1  6120  G  C  2  2  0
```

Annotation Database

```
$ vtools use dbNSFP
INFO: Downloading annotation database from annoDB/dbNSFP.ann
INFO: Downloading annotation database from http://vtools.houstonbioinformatics.org/annoDB/dbNSFP-
      hg18_hg19_2_0.DB.gz
INFO: Using annotation DB dbNSFP in project concept.
INFO: dbNSFP version 2.0, maintained by Xiaoming Liu from UTSPH. Please cite
"Liu X, Jian X, and Boerwinkle E. 2011. dbNSFP: a lightweight database of human
non-synonymous SNPs and their functional predictions. Human Mutation. 32:894-899" and
"Liu X, Jian X, and Boerwinkle E. 2013. dbNSFP v2.0: A Database of Human Nonsynonymous
SNVs and Their Functional Predictions and Annotations. Human Mutation. 34:E2393-E2402."
if you find this database useful.
```

Under the hook, vtools will

- Check for a local database `dbNSFP.DB` and use it if possible
- If unavailable, download `dbNSFP.ann` from web
- If available, download the latest version of `dbNSFP-$version.DB.gz` from web and use it
- If failed, download source of dbNSFP from a URL specified in `dbNSFP.ann`
- If succeed, create a database from source

Annotation Database

```
$ vtools show annotation dbNSFP
Annotation database dbNSFP (version hg18_hg19_2_0)
Description:            dbNSFP version 2.0, maintained by Xiaoming Liu from
  UTSPH. Please cite "Liu X, Jian X, and Boerwinkle E. 2011. dbNSFP: a
  lightweight database of human non-synonymous SNPs and their functional
  predictions. Human Mutation. 32:894-899" and "Liu X, Jian X, and Boerwinkle
  E. 2013. dbNSFP v2.0: A Database of Human Nonsynonymous SNVs and Their
  Functional Predictions and Annotations. Human Mutation. 34:E2393-E2402." if
  you find this database useful.
Database type:          variant
Reference genome hg18:  chr, hg18_pos, ref, alt
Reference genome hg19:  chr, pos, ref, alt
  chr                   Chromosome number
  pos                   physical position on the chromosome as to hg19
                        (1-based coordinate)
  ref                   Reference nucleotide allele (as on the + strand)
  alt                   Alternative nucleotide allele (as on the + strand)
  aaref                 reference amino acid
  aaalt                 alternative amino acid
  hg18_pos              physical position on the chromosome as to hg19
                        (1-based coordinate)
  genename              common gene name
  Uniprot_acc           Uniprot accession number. Multiple entries separated
                        by ";".
  Uniprot_id            Uniprot ID number. Multiple entries separated by ";".
  Uniprot_aapos         amino acid position as to Uniprot. Multiple entries
                        separated by ";".
  Interpro_domain       Interpro_domain: domain or conserved site on which the
                        variant locates. Domain annotations come from Interpro
                        database. The number in the brackets following a
                        specific domain is the count of times Interpro assigns
                        the variant position to that domain, typically coming
                        from different predicting databases. Multiple entries
```

Annotation Database

```
$ vtools show fields
variant.chr
variant.pos
variant.ref
variant.alt
variant.AA
variant.AC
variant.AN
variant.DP
variant.id
dbNSFP.chr                    Chromosome number
dbNSFP.pos                    physical position on the chromosome as to hg19
                              (1-based coordinate)
dbNSFP.ref                    Reference nucleotide allele (as on the + strand)
dbNSFP.alt                    Alternative nucleotide allele (as on the + strand)
dbNSFP.aaref                  reference amino acid
dbNSFP.aaalt                  alternative amino acid
dbNSFP.hg18_pos               physical position on the chromosome as to hg19 (1-based
                              coordinate)
dbNSFP.genename               common gene name
dbNSFP.Uniprot_acc            Uniprot accession number. Multiple entries separated by ";".
dbNSFP.Uniprot_id             Uniprot ID number. Multiple entries separated by ";".
dbNSFP.Uniprot_aapos          amino acid position as to Uniprot. Multiple entries separated
                              by ";".
dbNSFP.Interpro_domain        Interpro_domain: domain or conserved site on which the variant
                              locates. Domain annotations come from
                              Interpro database. The number in the
                              brackets following a specific domain is
                              the count of times Interpro assigns the
                              variant position to that domain,
                              typically coming from different
                              predicting databases. Multiple entries
                              separated by ";".
```

Annotation Database

```
$ vtools output refT chr pos ref alt genename SIFT_score KGp1_AFR_AF -l5
1  1105366  T  C  TTLL10    0.07  0.00406504065041
1  1110240  T  A  TTLL10    0.92  0.0
1  3537996  T  C  .         .     .
1  6447088  T  C  TNFRSF25  0.29  0.211382113821
1  6447275  T  C  .         .     .

$ vtools select variant 'SIFT_score < 0.05' -o chr pos ref alt SIFT_score Polyphen2_HDIV_score
     Polyphen2_HDIV_pred -l 10
1  3541597   C  T  0.0   1.0          D
1  18022097  G  T  0.0   0.004        B
1  18022200  C  A  0.0   0.999        D
1  18022253  A  G  0.0   0.649        P
1  25442668  T  C  0.04  0.087        B
1  25445571  T  G  0.0   0.999        D
1  25445572  C  T  0.0   0.99         D
1  25445603  A  G  0.0   0.999        D
1  35999342  C  G  0.01  0.99;1.0     D;D
1  36002845  T  G  0.01  0.649;0.825  P;P
```

Annotation Database

```
$ vtools output refT chr pos ref alt genename SIFT_score KGp1_AFR_AF -l5
1  1105366  T  C  TTLL10    0.07  0.00406504065041
1  1110240  T  A  TTLL10    0.92  0.0
1  3537996  T  C  .         .     .
1  6447088  T  C  TNFRSF25  0.29  0.211382113821
1  6447275  T  C  .         .     .

$ vtools select variant 'SIFT_score < 0.05' -o chr pos ref alt SIFT_score Polyphen2_HDIV_score
     Polyphen2_HDIV_pred -l 10
1  3541597   C  T  0.0   1.0          D
1  18022097  G  T  0.0   0.004        B
1  18022200  C  A  0.0
1  18022253  A  G  0.0
1  25442668  T  C  0.0
1  25445571  T  G  0.0
1  25445572  C  T  0.0
1  25445603  A  G  0.0
1  35999342  C  G  0.0
1  36002845  T  G  0.0
```

Please pay close attention to the description of fields before using them. For example, a variant is predicted to be damaging with smaller SIFT score but higher Polyphen2 scores.

dbSNP

Use command `vtools use` to link to annotation databases. Databases without version name always refer to the latest version. If you need to use a particular version of database, use databases such as `dbSNP-hg18_130`.

```
$ vtools use dbSNP
INFO: Downloading annotation database from annoDB/dbSNP.ann
INFO: Downloading annotation database from http://vtools.houstonbioinformatics.org/annoDB/dbSNP-
      hg19_138.DB.gz
INFO: Using annotation DB dbSNP in project RA.
INFO: dbSNP version 138, created using vcf file downloaded from NCBI

$ vtools output variant chr pos ref alt dbSNP.name -l10
1  583   G  A  rs58108140
1  4770  A  G  rs79585140
1  5931  T  C  rs372319358
1  5966  T  G  rs200358166
1  6120  G  C  rs78588380
1  6241  T  C  rs148220436
1  6360  A  G  rs150723783
1  7401  C  A  rs200046632
1  9131  C  T  .
1  9992  C  T  rs202081272
```

refGene and refGene_exon

Several gene databases are available based on different prediction criteria.

```
$ vtools use refGene
INFO: Downloading annotation database from annoDB/refGene.ann
INFO: Downloading annotation database from http://vtools.houstonbioinformatics.org/annoDB/refGene-
      hg19_20130904.DB.gz
INFO: Using annotation DB refGene in project RA.
INFO: Known human protein-coding and non-protein-coding genes taken from the NCBI RNA reference
      sequences collection (RefSeq).

$ vtools use refGene_exon
INFO: Downloading annotation database from annoDB/refGene_exon.ann
INFO: Downloading annotation database from http://vtools.houstonbioinformatics.org/annoDB/
      refGene_exon-hg19_20130904.DB.gz
INFO: Using annotation DB refGene_exon in project RA.
INFO: RefGene specifies known human protein-coding and non-protein-coding genes taken from the
      NCBI RNA reference sequences collection (RefSeq). This database contains all exome regions
      of the refSeq genes.

$ vtools output variant chr pos ref alt refGene.name refGene.name2 refGene_exon.name2 -l 10
1  583   G  A  .          .       .
1  4770  A  G  NR_024540  WASH7P  .
1  5931  T  C  NR_024540  WASH7P  .
1  5966  T  G  NR_024540  WASH7P  .
1  6120  G  C  NR_024540  WASH7P  .
1  6241  T  C  NR_024540  WASH7P  .
1  6360  A  G  NR_024540  WASH7P  .
1  7401  C  A  NR_024540  WASH7P  .
1  9131  C  T  NR_024540  WASH7P  .
1  9992  C  T  NR_024540  WASH7P  .
```

dbNSFP

dbNSFP provides a comprehensive set of annotations, most notably function-prediction scores, for non-symnonymous SNPs in CCDS genes.

```
$ vtools use dbNSFP
INFO: dbNSFP version 2.1, maintained by Xiaoming Liu from UTSPH. Please cite
"Liu X, Jian X, and Boerwinkle E. 2011. dbNSFP: a lightweight database of human
non-synonymous SNPs and their functional predictions. Human Mutation. 32:894-899" and
"Liu X, Jian X, and Boerwinkle E. 2013. dbNSFP v2.0: A Database of Human Nonsynonymous
SNVs and Their Functional Predictions and Annotations. Human Mutation. 34:E2393-E2402."
if you find this database useful.

$ vtools output variant chr pos ref alt SIFT_score PolyPhen2_HDIV_score -l 10
1  583   G  A  .  .
1  4770  A  G  .  .
1  5931  T  C  .  .
1  5966  T  G  .  .
1  6120  G  C  .  .
1  6241  T  C  .  .
1  6360  A  G  .  .
1  7401  C  A  .  .
1  9131  C  T  .  .
1  9992  C  T  .  .
```

Identify variants in dbNSFP

Variants that are not covered by a database will conceptually have `NULL` values for all fields. Condition "`dbNSFP.chr IS NOT NULL`" can therefore be used to select all variants that are in dbNSFP.

```
$ vtools select variant 'dbNSFP.chr IS NOT NULL' -t NS 'Non-symnonymous SNPs'
Running: 20,519 234.4/s in 00:01:27
INFO: 26963 variants selected.

$ vtools output NS chr pos ref alt SIFT_score Polyphen2_HDIV_score -l 10
1  878522   T  C  1.0   0.0
1  879101   G  A  0.07  0.999;0.999;0.99
1  901458   A  G  0.0   0.518
1  904196   C  G  0.46  0.0
1  904715   G  C  1.0   0.0
1  904739   T  C  0.43  0.001
1  906412   A  G  0.37  .
1  939471   G  A  0.0   0.01
1  1148494  A  G  .     .
1  1548655  T  C  0.31  0.013;0.0;0.0

$ vtools show tables
table      #variants     date message
NS            26,963    Oct03 Non-symnonymous SNPs
variant    8,905,869    Oct03 Master variant table
```

Identify variants in exon regions and genomic duplication regions

```
vtools use refGene_exon
vtools select not_in_ctrl 'refGene_exon.chr is not NULL' -t exon

vtools use genomicSuperDups
vtools select exon 'genomicSuperDups.chr is NULL' -t exon_not_dups
```

There are many gene definition databases. Variant tools provides

- ref seq gene: from UCSC known human protein-coding gene, and non-protein-coding genes taken from the NCBI RNA reference sequences collection.
- known gene: Gene predictions from many sources
- CCDS gene: Consensus Coding Sequence
- Entrez Gene: NCBI database for gene-specific information, focus on genomes that have been completely sequenced.

The use of different databases will affect your result.

Select variants

```
$ vtools select NS 'SIFT_score < 0.05' -t NS_damaging 'Non-symnonymous SNPs with SIFT score <
      0.05'
Running: 93 177.9/s in 00:00:00
INFO: 5619 variants selected.

$ vtools select NS 'SIFT_score < 0.05 OR Polyphen2_HDIV_score_max > 0.95' -t NS_or
Running: 105 195.5/s in 00:00:00
INFO: 7800 variants selected.

$ vtools compare NS_or NS_damaging --difference NS_pp2 'Variants in table NS_or but not in
      NS_damaging'
INFO: Reading 7,800 variants in NS_or...
INFO: Reading 5,619 variants in NS_damaging...
Writing to NS_pp2: 100% [==========================] 2,181 78.2K/s in 00:00:00
2181

$ vtools output NS_pp2 chr pos ref alt SIFT_score PolyPhen2_HDIV_score LRT_pred -l 8
1  879101    G  A  0.07  0.999;0.999;0.99                                    N
1  1640705   G  A  0.08  0.097;1.0;0.243;1.0;1.0;0.998;1.0;1.0;1.0;0.999;1.0  U
1  4672577   G  A  0.32  0.999                                               N
1  6447088   T  C  0.29  1.0;1.0;1.0;1.0                                     N
1  6553693   C  T  .     .                                                   .
1  8932038   G  C  .     1.0                                                 N
1  8939791   A  G  0.13  0.984;0.971                                         N
1  11778965  G  A  0.05  0.998;0.999                                         D
```

Select variants

```
$ vtools select NS 'SIFT_score < 0.05' -t NS_d
      0.05'
Running: 93 177.9/s in 00:00:00
INFO: 5619 variants selected.

$ vtools select NS 'SIFT_score < 0.05 OR Polyphen2_HDIV_score_max > 0.95' -t NS_or
Running: 105 195.5/s in 00:00:00
INFO: 7800 variants selected.

$ vtools compare NS_or NS_damaging --difference NS_pp2 'Variants in table NS_or but not in
      NS_damaging'
INFO: Reading 7,800 variants in NS_or...
INFO: Reading 5,619 variants in NS_damaging...
Writing to NS_pp2: 100% [==========================] 2,181 78.2K/s in 00:00:00
2181

$ vtools output NS_pp2 chr pos ref alt SIFT_score PolyPhen2_HDIV_score LRT_pred -l 8
1  879101    G  A  0.07  0.999;0.999;0.99                                  N
1  1640705   G  A  0.08  0.097;1.0;0.243;1.0;1.0;0.998;1.0;1.0;1.0;0.999;1.0  U
1  4672577   G  A  0.32  0.999                                             N
1  6447088   T  C  0.29  1.0;1.0;1.0;1.0                                   N
1  6553693   C  T  .     .                                                 .
1  8932038   G  C  .     1.0                                               N
1  8939791   A  G  0.13  0.984;0.971                                       N
1  11778965  G  A  0.05  0.998;0.999                                       D
```

Descriptions to variant tables are optional, but highly recommended.

VARIANT SELECTING USING OTHER FIELDS

In addition to annotation fields, variant info fields, built-in function, and extended functions such as `track` can also be used for variant selection.

```
$ vtools select NS 'case_num=5' 'ctrl_num=0' -t case_only 'NS SNPs exist only in cases'
Running: 29 1.0/s in 00:00:28
INFO: 1060 variants selected.

$ vtools select NS "ref_sequence(chr, pos-1) = 'C'" "ref_sequence(chr, pos+1) = 'G'" -t CpG 'SNPs
     in CpG sites'
Running: 52 291.1/s in 00:00:00
INFO: 3144 variants selected.

$ vtools output CpG chr pos ref alt 'ref_sequence(chr, pos-2, pos+2)' -l 5
1  904739   T  C  GCTGG
1  1877105  G  A  GCGGC
1  1878053  C  A  GCCGA
1  2134648  A  G  ACAGC
1  2423760  C  T  CCCGC

$ vtools update variant --set "hwe=HWE_exact(num, het, hom)"
INFO: Adding variant info field hwe

$ vtools select NS 'hwe < 0.05' --output chr pos ref alt num het hom hwe -l 5
1  878522   T  C  17  1  8  0.000243679501334
1  904739   T  C  10  0  5  0.00136396111628
1  906412   A  G  6   0  3  0.021645021645
1  1148494  A  G  8   0  4  0.00543900543901
1  1876879  A  G  9   1  4  0.0364459070341
```

VARIANT SELECTING USING OTHER FIELDS

In addition to annotation fields, variant info fields, built-in function, and extended functions such as `track` can also be use

Genotype counts in subgroups are frequently used to detect variants that, for example, exist only in offspring (De Novo), exist only in probands (case only), or exist only as homozygotes in probands (recessive).

```
$ vtools select NS 'case_num=5' 'ctrl_num=0' -
Running: 29 1.0/s in 00:00:28
INFO: 1060 variants selected.

$ vtools select NS "ref_sequence(chr, pos-1) =                                    Ps
     in CpG sites'
Running: 52 291.1/s in 00:00:00
INFO: 3144 variants selected.

$ vtools output CpG chr pos ref alt 'ref_sequence(chr, pos-2, pos+2)' -l 5
1  904739   T  C  GCTGG
1  1877105  G  A  GCGGC
1  1878053  C  A  GCCGA
1  2134648  A  G  ACAGC
1  2423760  C  T  CCCGC

$ vtools update variant --set "hwe=HWE_exact(num, het, hom)"
INFO: Adding variant info field hwe

$ vtools select NS 'hwe < 0.05' --output chr pos ref alt num het hom hwe -l 5
1  878522   T  C  17  1  8  0.000243679501334
1  904739   T  C  10  0  5  0.00136396111628
1  906412   A  G  6   0  3  0.021645021645
1  1148494  A  G  8   0  4  0.00543900543901
1  1876879  A  G  9   1  4  0.0364459070341
```

What pathways these variants belong?

```
$ vtools use ccdsGene
INFO: Downloading annotation database from annoDB/ccdsGene.ann
INFO: Downloading annotation database from http://vtools.houstonbioinformatics.org/annoDB/ccdsGene
      -hg19_20130904.DB.gz
INFO: Using annotation DB ccdsGene in project RA.
INFO: High-confidence human gene annotations from the Consensus Coding Sequence (CCDS) project.

$ vtools use keggPathway --linked_by ccdsGene.name
INFO: Downloading annotation database from annoDB/keggPathway.ann
INFO: Downloading annotation database from http://vtools.houstonbioinformatics.org/annoDB/
      keggPathway-20110823.DB.gz
INFO: Using annotation DB keggPathway in project RA.
INFO: kegg pathway for CCDS genes
INFO: 6821 out of 27731 ccdsgene.name are annotated through annotation database keggPathway
WARNING: 128 out of 6949 values in annotation database keggPathway are not linked to the project.

$ vtools output NS chr pos ccdsGene.name KgID KgDesc -l 10
1  878522   CCDS3.1      .         .
1  879101   CCDS3.1      .         .
1  901458   .            .         .
1  904196   .            .         .
1  904715   .            .         .
1  904739   .            .         .
1  906412   .            .         .
1  939471   CCDS6.1      hsa04622  RIG-I-like receptor signaling pathway
1  1148494  CCDS12.1     .         .
1  1548655  CCDS41224.2  .         .
```

What pathways these variants belong?

```
$ vtools use ccdsGene
INFO: Downloading annotation database from ann
INFO: Downloading annotation database from htt                                    ene
      -hg19_20130904.DB.gz
INFO: Using annotation DB ccdsGene in project
INFO: High-confidence human gene annotations f

$ vtools use keggPathway --linked_by ccdsGene.name
INFO: Downloading annotation database from annoDB/keggPathway.ann
INFO: Downloading annotation database from http://vtools.houstonbioinformatics.org/annoDB/
      keggPathway-20110823.DB.gz
INFO: Using annotation DB keggPathway in project RA.
INFO: kegg pathway for CCDS genes
INFO: 6821 out of 27731 ccdsgene.name are annotated through annotation database keggPathway
WARNING: 128 out of 6949 values in annotation database keggPathway are not linked to the project.

$ vtools output NS chr pos ccdsGene.name KgID KgDesc -l 10
1  878522   CCDS3.1      .         .
1  879101   CCDS3.1      .         .
1  901458   .            .         .
1  904196   .            .         .
1  904715   .            .         .
1  904739   .            .         .
1  906412   .            .         .
1  939471   CCDS6.1      hsa04622  RIG-I-like receptor signaling pathway
1  1148494  CCDS12.1     .         .
1  1548655  CCDS41224.2  .         .
```

The `keggPathway` database annotates genes through their CCDS gene ID, which are available in `ccdsGene` and `dbNSFP`. `ccdsGene` is preferred though.

Find variants that belong to a pathway

```
$ vtools select NS 'kgID="hsa00760"' --output chr pos ref alt ccdsGene.name kgID kgDesc -l 20
1   1675900    G  T  CCDS30565.1  hsa01100  Metabolic pathways
11  70847195   G  C  CCDS8201.1   hsa01100  Metabolic pathways
11  70862326   A  C  CCDS8201.1   hsa01100  Metabolic pathways
14  20010446   G  A  CCDS9552.1   hsa01100  Metabolic pathways
16  29615851   A  G  CCDS10651.1  hsa01100  Metabolic pathways
4   15318290   G  A  CCDS3416.1   hsa04020  Calcium signaling pathway
5   43691831   C  T  CCDS3949.1   hsa01100  Metabolic pathways
5   102922572  T  C  CCDS4096.1   hsa04146  Peroxisome
6   86255952   A  G  CCDS5002.1   hsa01100  Metabolic pathways
6   132214061  A  C  CCDS5150.2   hsa01100  Metabolic pathways
1   1675941    G  A  CCDS30565.1  hsa01100  Metabolic pathways
6   132072584  T  G  CCDS47475.1  .         .
6   132071774  G  A  CCDS47475.1  .         .
6   132072589  T  C  CCDS47475.1  .         .
10  104924699  T  C  CCDS7544.1   hsa01100  Metabolic pathways
6   132071745  G  T  CCDS47475.1  .         .
2   201234575  A  G  CCDS33360.1  hsa01100  Metabolic pathways
6   132103113  G  A  CCDS5148.1   hsa01100  Metabolic pathways
11  70869465   C  T  CCDS8201.1   hsa01100  Metabolic pathways
16  29613945   C  T  CCDS10651.1  hsa01100  Metabolic pathways
```

Find variants that belong to a pathway

```
$ vtools select NS 'kgID="hsa00760"' --output chr pos ref alt ccdsGene.name kgID kgDesc -l 20
1   1675900    G  T  CCDS30565.1  hsa01100  Metabolic pathways
11  70847195   G  C  CCDS8201.1   hsa01100  Metabolic pathways
11  70862326   A  C  CCDS8201.1   hsa01100  Metabolic pathways
14  20010446   G  A  CCDS9552.1   hsa01100  Metabolic pathways
16  29615851   A  G  CCDS10651.1  hsa01100  Metabolic pathways
4   15318290   G  A  CCDS3416.1   hsa04020  Calcium signaling pathway
5   43691831   C  T  CCDS3949.1   hsa01100  Metabolic pathways
5   102922572  T  C  CCDS4096.1   hsa04146  Peroxisome
6   86255952   A  G  CCDS5002.1   hsa01100  Metabolic pathways
6   132214061  A  C  C
1   1675941    G  A  C
6   132072584  T  G  C
6   132071774  G  A  C
6   132072589  T  C  C
10  104924699  T  C  C
6   132071745  G  T  CCDS47475.1  .         .
2   201234575  A  G  CCDS33360.1  hsa01100  Metabolic pathways
6   132103113  G  A  CCDS5148.1   hsa01100  Metabolic pathways
11  70869465   C  T  CCDS8201.1   hsa01100  Metabolic pathways
16  29613945   C  T  CCDS10651.1  hsa01100  Metabolic pathways
```

Notice any problem with the output?

THE –ALL OPTION

When there are multiple records for a variant in an annotation database, variant tools by default output one of them randomly. The `--all` options tells *variant tools* to output all matching records.

```
$ vtools select NS 'kgID="hsa00760"' --output chr pos ref alt ccdsGene.name kgID kgDesc --all -l
     20
1   1675900    G  T  CCDS55562.1  .         .
1   1675900    G  T  CCDS55561.1  .         .
1   1675900    G  T  CCDS30565.1  hsa00760  Nicotinate and nicotinamide metabolism
1   1675900    G  T  CCDS30565.1  hsa01100  Metabolic pathways
11  70847195   G  C  CCDS8201.1   hsa00760  Nicotinate and nicotinamide metabolism
11  70847195   G  C  CCDS8201.1   hsa01100  Metabolic pathways
11  70862326   A  C  CCDS8201.1   hsa00760  Nicotinate and nicotinamide metabolism
11  70862326   A  C  CCDS8201.1   hsa01100  Metabolic pathways
14  20010446   G  A  CCDS9552.1   hsa00230  Purine metabolism
14  20010446   G  A  CCDS9552.1   hsa00240  Pyrimidine metabolism
14  20010446   G  A  CCDS9552.1   hsa00760  Nicotinate and nicotinamide metabolism
14  20010446   G  A  CCDS9552.1   hsa01100  Metabolic pathways
16  29615851   A  G  CCDS10651.1  hsa00760  Nicotinate and nicotinamide metabolism
16  29615851   A  G  CCDS10651.1  hsa01100  Metabolic pathways
4   15318290   G  A  CCDS3416.1   hsa00760  Nicotinate and nicotinamide metabolism
4   15318290   G  A  CCDS3416.1   hsa01100  Metabolic pathways
4   15318290   G  A  CCDS3416.1   hsa04020  Calcium signaling pathway
5   43691831   C  T  CCDS3949.1   hsa00760  Nicotinate and nicotinamide metabolism
5   43691831   C  T  CCDS3949.1   hsa01100  Metabolic pathways
5   102922572  T  C  CCDS4096.1   hsa00760  Nicotinate and nicotinamide metabolism
```

Using ANNOVAR to annotate variants

Formats such as `ANNOVAR` and `ANNOVAR_exonic_variant_function` are provided to export variants to be analyzed by other programs, and import results from output of these programs.

```
$ vtools export NS --format ANNOVAR > annovar.input
INFO: Using primary reference genome hg18 of the project.
Writing: 100% [=====================================] 26,963 45.1K/s in 00:00:000
INFO: 26963 lines are exported from variant table NS

$ ~/bin/annovar/annotate_variation.pl annovar.input ~/bin/annovar/humandb/
NOTICE: The --geneanno operation is set to ON by default
NOTICE: The --buildver is set as 'hg18' by default
NOTICE: Reading gene annotation from /Users/bpeng/bin/annovar/humandb/hg18_refGene.txt ... Done
      with 42259 transcripts (including 7526 without coding sequence annotation) for 23769 unique
      genes
NOTICE: Reading FASTA sequences from /Users/bpeng/bin/annovar/humandb/hg18_refGeneMrna.fa ... Done
       with 16660 sequences
WARNING: A total of 329 sequences will be ignored due to lack of correct ORF annotation
NOTICE: Finished gene-based annotation on 26963 genetic variants in annovar.input
NOTICE: Output files were written to annovar.input.variant_function, annovar.input.
      exonic_variant_function

$ vtools update NS --format ANNOVAR_exonic_variant_function --from_file annovar.input.
      exonic_variant_function --var_info mut_type function
INFO: Using primary reference genome hg18 of the project.
Getting existing variants: 100% [=================] 26,963 121.9K/s in 00:00:000
INFO: Updating variants from annovar.input.exonic_variant_function (1/1)
annovar.input.exonic_variant_function: 100% [=======] 23,683 8.1K/s in 00:00:020
INFO: Fields mut_type, function of 23,683 variants are updated
```

Identifying stopgain mutations

```
$ vtools output NS mut_type | sort | uniq
.
nonsynonymous SNV
stopgain SNV
stoploss SNV
synonymous SNV
unknown

$ vtools select NS 'mut_type = "stopgain SNV"' --output chr pos ref alt mut_type -l 20
1   12776677   T  A  stopgain SNV
1   20374169   G  A  stopgain SNV
1   48480815   G  T  stopgain SNV
1   143787040  C  T  stopgain SNV
1   143984723  C  T  stopgain SNV
1   159742828  C  T  stopgain SNV
1   159779491  G  A  stopgain SNV
1   221351823  G  A  stopgain SNV
1   236115192  G  A  stopgain SNV
1   246179649  T  A  stopgain SNV
10  4879403    C  T  stopgain SNV
11  5400712    C  T  stopgain SNV
11  48242807   T  A  stopgain SNV
11  48303590   G  A  stopgain SNV
11  55127957   G  A  stopgain SNV
11  56066932   A  T  stopgain SNV
11  56187792   C  T  stopgain SNV
11  60021578   C  T  stopgain SNV
11  62605063   A  C  stopgain SNV
11  62814501   G  A  stopgain SNV
```

Output summary statistics

```
$ vtools select variant 'ref="-"' --count
Counting variants: 3,059 734.6/s in 00:00:04
775833

$ vtools output variant refGene.name2 'count(*)' --group_by refGene.name2 -l 5
.          5358110
A1BG       17
A1BG-AS1   10
A1CF       144
A2M        145

$ vtools select variant "(ref='A' AND alt='G') OR (ref='G' AND alt='A') OR (ref='C' AND alt='T')
      OR (ref='T' AND alt='C')" --output 'sum(num)'
17120173

$ vtools select variant 'genename is not NULL' --output genename 'sum(case_num)' 'sum(ctrl_num)'
      --group_by genename -l 10
A1BG    10      6
A2ML1   37      0
A4GALT  2       2
A4GNT   9       0
AAAS    1       1
AADAC   9       4
AADACL2 5       8
AADACL3 32      0
AAGAB   7       0
AARS    0       1
```

Remove low quality genotypes

```
# start from a snapshot with both max_gt and GATK called variants
vtools admin --load_snapshot vcf_max_gt

# remove genotypes with low quality scores
vtools remove genotypes 'GQ_geno < 20'
vtools remove genotypes 'Q_indel < 20 or Q_max_gt < 20'

# GT=0 will not remove any genotype because wildtypes are not imported.
vtools update variant --from_stat 'total_num=#(GT)'
vtools select variant 'total_num = 0' -t to_be_removed

# 575346 variants are removed
vtools remove variants to_be_removed
```

Wild type genotypes are sometimes imported, especially from multi-sample calling pipelines. They are usually removed from analysis.

Create variant tables for each sample

```
for name in CASE001 CASE072 CASE107 CASE003 CASE134 CTRL113 CTRL132 CTRL140 \
    CASAVA_CASE001 CASAVA_CASE072 CASAVA_CASE107 CASAVA_CASE003 \
    CASAVA_CASE134 CASAVA_CTRL113 CASAVA_CTRL132 CASAVA_CTRL140
do
   vtools select variant --samples "sample_name='$name'" -t $name
   vtools select $name 'length(ref)=1' 'length(alt)=1' "ref!='-'" " alt!='-'" -t ${name}_SNP
   vtools compare $name ${name}_SNP --difference ${name}_INDEL
done

vtools select variant 'length(ref)=1' 'length(alt)=1' "ref!='-'" " alt!='-'" -t SNP
vtools compare variant SNP --difference INDEL
```

Indel variants have – as reference (insertion) or alternative (deletion) allele. Variant tools does not support large indels and genomic structural variants.

Show variant tables

```
$ vtools show tables
table                    #variants     date message
CASAVA_CASE001           4,463,909    Jan14
CASAVA_CASE001_INDEL       693,171    Jan14
CASAVA_CASE001_SNP       3,770,738    Jan14
CASAVA_CASE003           4,482,247    Jan14
CASAVA_CASE003_INDEL       699,540    Jan14
CASAVA_CASE003_SNP       3,782,707    Jan14
CASAVA_CASE072           4,408,125    Jan14
CASAVA_CASE072_INDEL       670,720    Jan14
CASAVA_CASE072_SNP       3,737,405    Jan14
CASAVA_CASE107           4,434,639    Jan14
CASAVA_CASE107_INDEL       676,914    Jan14
CASAVA_CASE107_SNP       3,757,725    Jan14
CASAVA_CASE134           4,523,237    Jan14
CASAVA_CASE134_INDEL       697,605    Jan14
CASAVA_CASE134_SNP       3,825,632    Jan14
CASAVA_CTRL113           4,455,796    Jan14
CASAVA_CTRL113_INDEL       676,469    Jan14
CASAVA_CTRL113_SNP       3,779,327    Jan14
CASAVA_CTRL132           4,526,319    Jan14
CASAVA_CTRL132_INDEL       680,362    Jan14
CASAVA_CTRL132_SNP       3,845,957    Jan14
CASAVA_CTRL140           4,473,640    Jan14
CASAVA_CTRL140_INDEL       660,227    Jan14
CASAVA_CTRL140_SNP       3,813,413    Jan14
CASE001                  4,788,107    Jan14
CASE001_INDEL              664,344    Jan14
CASE001_SNP              4,123,763    Jan14
CASE003                  4,812,347    Jan14
CASE003_INDEL              670,833    Jan14
CASE003_SNP              4,141,514    Jan14
CASE072                  4,667,062    Jan14
CASE072_INDEL              642,317    Jan14
```

Confirm parent/offspring relationships

```
$ vtools compare CASE003 CASE001
INFO: Reading approximately 4,812,347 variants in CASE003...
INFO: Reading approximately 4,788,107 variants in CASE001...
INFO: Number of variants in A but not B, B but not A, A and B, and A or B
1071458 1047218 3740889 5859565

$ vtools compare CTRL132 CASE134
INFO: Reading approximately 4,815,797 variants in CTRL132...
INFO: Reading approximately 4,811,365 variants in CASE134...
INFO: Number of variants in A but not B, B but not A, A and B, and A or B
1085244 1080812 3730553 5896609

$ vtools compare CASE107 CTRL113
INFO: Reading approximately 4,749,771 variants in CASE107...
INFO: Reading approximately 4,779,875 variants in CTRL113...
INFO: Number of variants in A but not B, B but not A, A and B, and A or B
1699152 1729256 3050619 6479027

$ vtools compare CTRL113 CTRL140
INFO: Reading approximately 4,779,875 variants in CTRL113...
INFO: Reading approximately 4,760,537 variants in CTRL140...
INFO: Number of variants in A but not B, B but not A, A and B, and A or B
1749642 1730304 3030233 6510179
```

Parent/offspring share more variants than unrelated samples.

Depth of coverage of these variants in BAM file

```
$ vtools output inconsistent chr pos ref alt "track('sample1.bam')" "track('sample2.bam')"
1       174980243       AAAAAAAA        -               43      30
4       4865498         AA              -               33      24
4       88537204        C               T               22      19
11      70281359        G               T               45      34
16      11966239        -               A               65      37
16      24267639        -               CA              30      29
19      39399199        G               A               42      33
19      4523091         -               T               37      36
3       12598526        -               CGGCGTGCGC      30      22
```

Reads aligned around these variants

```
$ vtools output inconsistent chr pos ref alt "track('/Volumes/Home/Data/HFamily/Recalled/LP6005158
     -DNA_B01_new.bam', 'reads?color=1&start=-5&width=20&limit=3')"
1       174980243       AAAAAAAA        -               .......             |........
     |...........
4       4865498         AA              -               ......              |......
     |........
4       88537204        C               T               ................C.T|.....T..T.....C
     .....|.....T..T.....C.....
11      70281359        G               T               ..........          |......C....
     |............
16      11966239        -               A               .......             |......
     |..........
16      24267639        -               CA              ........            |...............
     |....C......G..C
19      39399199        G               A               .C.G..C              |........
     |.....A........A....G
19      4523091         -               T               ......              |.......
     |....G....
3       12598526        -               CGGCGTGCGC      ................  |............A
     ......|............G.......
```

Insertion, nucleotide at variant location will be displayed in color with option `color=1`.

VTOOLS_REPORT

vtools_report is built on top of vtools to perform tasks that would require the use of multiple vtools commands.

```
$ vtools_report -h
usage: vtools_report [-h] [--version]

                     {trans_ratio,avg_depth,variant_stat,discordance_rate,sequence,plot_fields,
                          plot_geno_fields,plot_association,meta_analysis}
                     ...

A collection of functions that analyze data using vtools and generate various
reports

optional arguments:
  -h, --help            show this help message and exit
  --version             show program's version number and exit

Available reports:
  {trans_ratio,avg_depth,variant_stat,discordance_rate,sequence,plot_fields,plot_geno_fields,
       plot_association,meta_analysis}
    trans_ratio         Transition count, transversion count and
                        transition/transversion ratio
    avg_depth           Average depth for each variant, can be divided by
                        sample variant count
    variant_stat        Reports number of snps, insertions, deletions and
                        substitutions for groups of samples with some size
                        metrics to characterize the indels
    discordance_rate    Calculate discordance rate between pairs of samples
    sequence            Obtain DNA sequence in specified chromosomal region.
                        This command by default outputs nucleotide sequence at
                        the reference genome.
    plot_fields         Dump values of specified variant info field(s) and/or
```

Transition/transversion ratio

Command `trans_ratio` calculates transition - transversion ratio of all mutations in the samples, using an existing field that records the number of variants in the samples.

```
$ vtools_report trans_ratio variant -n num
num_of_transition       num_of_transversion     ratio
16,534,168              8,213,424               2.01307

$ vtools_report trans_ratio variant -n num --group_by num
num     num_of_transition       num_of_transversion     ratio
0       0                       0                       0.00000
1       1,471,898               789,039                 1.86543
10      2,176,350               1,062,220               2.04887
11      51,282                  20,757                  2.47059
12      74,784                  30,504                  2.45161
13      29,458                  12,064                  2.44181
14      43,596                  18,032                  2.41770
15      20,490                  8,055                   2.54376
16      34,896                  14,288                  2.44233
17      11,067                  4,624                   2.39338
18      25,560                  10,332                  2.47387
19      4,294                   1,634                   2.62791
2       1,490,186               763,804                 1.95101
20      11,580                  4,640                   2.49569
3       1,552,902               785,208                 1.97770
4       1,686,952               853,176                 1.97726
5       1,798,620               917,430                 1.96050
6       1,574,268               764,286                 2.05979
7       1,514,898               726,768                 2.08443
8       1,718,088               824,472                 2.08386
9       1,242,999               602,091                 2.06447
```

Compare variants called from two projects?

```
# create variant tables for sample using commands such as
vtools select variant --samples "sample_name='CASE001'" -t max_gt_CASE001
#
mkdir compare
cd compare
vtools init merged --children ../max_gt ../poly
vtools compare max_gt_CASE001 poly_CASE001
vtools compare max_gt_CASE003 poly_CASE001
vtools compare max_gt_CASE072 poly_CASE001
vtools compare max_gt_CASE107 poly_CASE001
vtools compare max_gt_CASE134 poly_CASE134
vtools compare max_gt_CTRL113 poly_CTRL113
vtools compare max_gt_CTRL132 poly_CTRL132
vtools compare max_gt_CTRL140 poly_CTRL140
```

Recall variants using the GATK pipeline

```
vtools execute bwa_gatk23_hg19 align \
    --input input_illumina_bam_file \
        --output gatk_realigned_bam_file reduced_bam_file \
        --name sample_name --production true  \
        --gatk_path /path/to/GATK \
        --picard_path /path/to/Picard \
        --opt_java '-Xmx24g -XX:-UseGCOverheadLimit -Djava.io.tmpdir=/path/to/local/temp'

vtools execute bwa_gatk23_hg19 call --input gatk_realigned_bam_file \
        --output recalled_vcf_file   --name sample_name --production true \
        --gatk_path /path/to/GATK \
        --picard_path /path/to/Picard \
        --opt_java '-Xmx24g -XX:-UseGCOverheadLimit -Djava.io.tmpdir=/path/to/local/temp'
```

- It tooks more than a week for the pipeline to complete, triple the time with cluster problems.
- The latest variant calling pipeline using the GATK best practice guideline is `bwa_gatk28_hg19`.

Annotate variants using snpEff

```
$ vtools execute snpEff --var_table exon1 --snpeff_path ~/bin/snpEff/ \
    --eff_fields EFF EFF_Type EFF_Impact EFF_Functional_Class
INFO: Executing snpEff.eff_0: Load specified snapshot if a snapshot is specified. Otherwise use
     the existing project.
INFO: Executing snpEff.eff_10: Check the existence of command java
INFO: Executing snpEff.eff_11: Check if snpEff is installed and executable
INFO: Executing snpEff.eff_12: Check the data storage location in snpEff.config file.
INFO: Executing snpEff.eff_14: Download reference database for the project reference genome
INFO: Executing snpEff.eff_20: Export variants in VCF format
INFO: Running vtools export exon1 --format vcf --output cache/snpEff_input.vcf
INFO: Executing snpEff.eff_30: Execute snpEff eff to annotate variants
INFO: Running java -jar -Xmx4g -XX:-UseGCOverheadLimit /Volumes/Home/bin/snpEff//snpEff.jar -c /
     Volumes/Home/bin/snpEff//snpEff.config -v hg19 cache/snpEff_input.vcf > cache/snpEff_output.
     vcf
INFO: Executing snpEff.eff_40: Importing results from snpEff

$ vtools select exon1 "EFF_Type='SYNONYMOUS_CODING'" -t synonymous
$ vtools compare exon1 synonymous --diff exon2
```

Export variants in csv and other formats

```
for table in case_only_not_1000g  case_only_not_1000g_exon  case_only_not_1000g_exon_cancer_gene
     in_4_case_not_1000g in_4_case_not_1000g_exon in_4_case_not_1000g_exon_cancer_gene
do
    vtools export ${table} --samples 1 --format csv \
             --fields chr pos ref alt 'ref_sequence(chr, pos, pos+10)' \
                    dbSNP.name refGene.name2 refGene.name EFF \
        --order_by chr pos \
                --header \
             chr pos ref alt 'ref (pos ~ pos+10)' rsname gene 'refgene name' \
            'snpEFF prediction' > ${table}.csv
done
```