MVFtools 

2010-12-19

Contents

Version 0.6.1

Introduction

Authors

James B. Pease (http://www.peaselab.org)

Benjamin K. Rosenzweig

Contributors:

Roddra Johnson

Ellen Weinheimer

What is MVFtools?

Multisample Variant Format (MVF) is designed for compact storage and efficient analysis of aligned multi-genome and multi-transcriptome datasets. The programs provided in MVFtools support this format, including data conversion, filtering and transformation, and data analysis and visualization modules. MVF format is designed specifically for biological data analysis, and sequence data

is encoded based on the information content at a particular aligned sequence site. This contextual encoding allows for rapid computation of phylogenetic and population genetic analyses, and small file sizes that enable unified and compact data sharing and distribution.

How do I cite this?

Pease JB and BK Rosenzweig. 2018. "Encoding Data Using Biological Principles: the Multisample Variant Format for Phylogenomics and Population Genomics" *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. 15(4):1231-1238. http://www.dx.doi.org/10.1109/tcbb.2015.2509997

Please also include the URL https://www.github.com/peaselab/mvftools in your methods section where the program is referenced.

(Note this paper was originally published online in 2015, but did not receive final citation page numbering until 2018. You may see older citations as 2015, which is the same paper.)

Getting Started

Requirements

- Python 3.x: https://www.python.org/downloads/
- Biopython 1.6+: http://www.biopython.org/
- Scipy: http://www.scipy.org/
- Numpy http://www.numpy.org/
- Matplotlib http://www.matplotlib.org/

Additional Requirements for Some Modules

- RAxML: 8.x recommended; https://sco.h-its.org/exelixis/web/software/raxml/index.html)
- PAML: http://abacus.gene.ucl.ac.uk/software/paml.html

Installation

No installation is required, mvftools scripts should work as long as Python3 is installed. The repository can be cloned or downloaded as a .zip file from GitHub.

```
git clone https://www.github.com/jbpease/mvftools
```

Preparing your data

Sequence Alignment

MVF files can be created from VCF, FASTA, and MAF files using the `ConvertVCF2MVF`, `ConvertFasta2MVF`, or `ConvertMAF2MVF` commands, respectively. Once converted to MVF format, analyses and manipulations can be carried out using the rest of the commands in MVFtools.

Basic usage examples

Case #1: Generate phylogenies from 100kb windows using a VCF data

```
python3 mvftools.py ConvertVCF2MVF --vcf DATA.vcf --mvf DATA.mvf
python3 mvftools.py InferWindowTree --mvf DATA.mvf --out WINDOWTREES.txt \
--windowsize 100000
```

Case #2: Convert a large FASTA file, then generate window-based counts for DFOIL/D-statistic introgression testing from the first five samples::

```
python3 mvftools.py ConvertFasta2MVF --fasta DATA.fasta --mvf DATA.mvf
python3 mvftools.py CalcPatternCount --mvf DATA.mvf --out PATTERNS.txt \
--windowsize 100000 --samples 0,1,2,3,4
```

The file is now ready to use as an input file for with dfoil: (http://www.github.com/jbpease/dfoil).

MVF Format Specification (version 1.2)

MVF General Notes and Usage

General Features

MVF is primarily intended for site-wise analyses in phylogenomics and population genomics. MVF is formatted to contain one aligned site per line, but contains only allelic information, therefore MVF most closely mimics VCF files in formatting, but resembles MAF format in informational content, Additionally, MVF uses special formatting to lower file sizes and speed up filtering and analysis. MVF can readily be adapted from other common sequence formats including VCF, FSATA, and MAF. MVF is also designed to be able to accommodate readily store other information for phylogenomic projects, including tree topologies and sample metadata.

Native Gzip read/write

MVF is designed to work natively with GZIP compression and uses a formatting that attempts to strike a balance between fast filtering, easy visual inspection, while using character patterns that create a good Gzip compression ratio. As

long as any input or output file path ends with exactly ".gz", all MVF scripts will natively read/write to gzip-compressed files.

General Notes on Filtering

MVF was specifically designed as a "vertical" format for rapid filtering of *sites* in large-scale phylogenomic analyses. (rather than being "horizontal" to visually show alignment) Therefore, the following should be noted to take advantage of MVF formatting for rapid filtering (i.e. with grep/zgrep).

- `#` is present iff. the line is in the header
- `@` is present iff. the position is non-reference
- `X` is present in the allele string iff. the positon has ambiguity data
- `#:` can quickly filter by chromosome
- `:#` can quickly filter by coordinate numbers
- Allele strings with one or two characters have full sample coverage (no gaps)
- Allele strings with `@[any]+` have coverage=1, `[not@][any]+` have coverage=2
- One or two-character allele strings, or notation with `[any]+` CANNOT contain homoplasy or synapomorphy (by definition).

Header Specification

All header lines begin with one or more `#` and contain single-space separated fields.

MVF declaration line

First header line always starts with `##mvf`, followed by required metadata fields:

- version=1.2
- mvftype=[dna, protein, codon]

and optionally:

- an arbitrary number of metadata fields in `key=value` format (`mvftype` and `version` not allowed as key)

Sample information

Sample information (columns) header lines are specified by:

- line starts with `#s` ("s" for sample) with no leading spaces
- LABEL (must be unique, no spaces)
- an arbitrary number of metadata fields in key=value format ('label' not allowed as key)

The first entry should be the reference sequence (if aligned to reference) or can be any sequence in the case of non-reference-aligned de novo alignment).

Contig information

Contig information header lines are specified by:

- line starts with `#c` ("c" for contig)
- CONTIG_ID (must be unique, alpha-numeric strong recommended, must not contain `*:;,@!+` or spaces)
- `label=[NAME]` (recommended by not required to be unique, no spaces allowed)
- `len=[LENGTH]` (integer > 0, or zero for unknown)
- `ref=[0/1]`, indicates if contig is reference-based (=1) or not (=0)
- an arbitrary number of metadata fields in key=value format ("label", "len", and "ref" not allowed as key)

Tree information

Tree information may (optionally) be specified in header lines by:

- line starts with `#t` ("t" for tree/topology)
- `TREE_ID=[###]` (must be unique, alpha-numeric)
- `TOPOLOGY=[tree_String]` in Newick/Phylip/parenthetical format (must end with ';')
- an arbitrary number of metadata fields in key=value format

To take full advantage of MVF tree storage, use the same sample labels as in the `#s` header lines

Notes

General project notes may (optionally) be specified in the header lines by:

- line starts with `#n` ("n" for notes)
- Text is unstructured and is not necessarily formatted as metadata

Example Header

```
##mvf version=1.2 mvftype=[MVFTYPE]
#s SAMPLE0 meta0=somevalue meta1=0 ...
#s SAMPLE1 meta0=somethingele meta1=1 ...
#s SAMPLE2 meta0=somesome meta1=0 ...
...
#c 0 label=CONTIG0 length=100 ref=1 meta0=somevalue ...
#c 1 label=CONTIG1 length=200 ref=0 meta0=someother ...
...
#t 0 ((SAMPLE0,SAMPLE1),SAMPLE2); model=GTRGAMMA software=RAxML
#t 1 ((SAMPLE2,SAMPLE0),SAMPLE1); model=GTRGAMMA software=RAxML partition=chrom1
...
#n Notes on this project.
```

Entry Specification

Note: all examples show an MVF entry with REF and four samples.

Entries are structured as two space-separated columns:

`ID:POSITION  ALLELES [ALLELES ALLELES ...]`

- `ID:POSITION` = chromosomal id matching the first element of a contig in the `#c` header element
- `POSITION` = 1-based position on the contig with matching `CONTIG_ID`
- `ALLELES` = one or more records of alleles at reference-based location specified by `ID:POSITION` and matching the formatting below

For mvftype=codon

- Allele columns are `PROTEIN DNA1 DNA2 DNA3` where the three DNA columns represent three codon positions in collated form
- Position is the position of the lowest numbered codon position (regardless of transcript strand) and `DNA1/2/3` codon columns are given in order to match the protein (again regardless of transcript orientation)

Allele formatting

Note: all examples show an MVF entry with five samples.

For reference-anchored contigs, the first allele is assumed to be the "reference" allele by default. Each entry must either (1) contain the same number of characters as sample labels specified in the header or (2) use one of the special cases in the section below.

`ATCTG` = (REF is 'A' samples 1&3 are 'T', sample 2 is 'C', sample 4 is 'G')

Special cases

Invariant sites

When all alleles are both present (non-gap) and all the same, this is represented by a single base.

`A = AAAAA`

Monoallelic non-reference samples

When all alleles in the samples (non-REF) are the same but differ from REF, this is represented by two bases.

`AT = ATTTT`

`Aa = Aaaaa`

Single-variant sites

When only one of the samples varies from the others, this is specified as:

`[reference_base, majority_base, "+", unique_base, unique_position]`

This is useful shorthand for both sites with one a single base that differs and samples with only one sample represented. When the site only has coverage via one sample (i.e. all other bases are empty, the '-' is omitted from the second position.

`AC+T2 = ACTCC`

`AA+C2 = AACAA`

`-+A2  = --A--`

`A+A2  = A-A--`

`A+a2  = A-a--`

`A+C2  = A-C--`

Non-reference aligned sites

Added in MVF v.1.2, this facilitates using MVF for non-reference aligned sequences (e.g. aligned sets of orthologs from de novo assembled transcripts). These non-reference-anchored alignments can comprise the entire MVF file or be included in addition to reference-aligned contigs. Non-reference-contigs in their header entry should include the keyword "nonref" (see Section 1.3). Contigs labels and coordinates are labelled the same as reference-based entries. To denote that the sequence is non-reference and not simply a deletion in the reference, the character "@" should be the first character of the alignment. In the case an entirely non-reference MVF, all contigs can be labelled as "nonref," but one sequence should be chosen as the reference for the purposes of the allele string. When this sequence is not present, `@` is still used.

`@AATT   = -AATT`

`@A+T3   = -A-T-`

`@-+A3   = ---A-`

Character encoding

Nucleotide Notation

- Standard IUPAC nucleotide codes are used: `ACGT`, and `U` for uracil in RNA
- Standard IUPAC bialleic ambiguity codes `KMRSWY` are used also.
- Standard IUPAC triallelic ambiguity codes (`BDHV`) are allowed in convsersion from FASTA and MAF, or when a polyploid VCF is converted.

For diploid VCFs, triallelic sites are converted to ambiguous (`X`) instead. (*Changed in 1.2.1*)
- Current MVF formatting does NOT recognize rare symbols (`ISOX`, or `Phi`)
- Ambiguous nucleotide is denoted by `X` instead of standard `N`

Amino Acid Notation

- Standard IUPAC amino acid codes are used: `ACDEFGHIKLMNPQRSTVWY`
- Standard stop codon symbol `*` is used
- Currently the ambiguous/rare symbols are not recognized (`BZ`)

Use of `X` for ambiguous nucleotides and amino acids

In standard notation, "`N`" is used for an ambiguous nucleotide, which could be any of A/C/G/T.
However, in amino acid notation `N` stands for "Asparagine" and is a valid character, while `X` is used for an ambiguous amino acid.

The MVF Standard since v1.2 adopts `X` as unified ambiguity character for both nucleotides and proteins for MVF files for two purposes:

1. To creates a unified ambiguity character for MVF codon files for faster processing
2. To allow fast filtering of ambiguous lines

Also note that while "`X`" in expanded IUPAC notation refers to "xanthosine," MVF currently does not support rare nucleotides.

NOTE: In all conversion utilities that export from MVF format to another file format conversion to the standard "N"/"X" for ambiguous nucleotides/amino acids should be implemented.

Examples of the same data in MVF Format and other formats

MVF Format

```
##mvf sourceformat=fasta version=1.2 mvftype=dna ncol=5
#s Hsapiens
#s Ptroglodytes
#s Ppaniscus
#s Ggorilla
#s Mmusculus
#c 1 label=Chromosome1 length=248956422
#n Note: This is an example file showing data formatting
1:100 A
1:101 A
```

```
1:102 A
1:103 T
1:104 TT+C4
1:105 GC
1:106 A+A4
1:107 AATTA
1:108 AC+G4
```

FASTA Format

```
>Hsapiens gi:1234 geneid:GeneOfInterest chrom:1 start:100 end:108
AAATTGAAA
>Ptroglodytes geneid:GeneOfInterest
AAATTC-AC
>Ppaniscus geneid:GeneOfInterest
AAATTC-TC
>Ggorilla geneid:GeneOfInterest
AAATTC-TC
>Mmusculus geneid:GeneOfInterest
AAATCCAAG
```

VCF Format

```
##fileformat=VCFv4.1
##samtoolsVersion=0.1.19-44428cd
##reference=hg19.fa
##contig=<ID=Chromosome1,length=248956422>
##INFO=<ID=DP,Number=1,Type=Integer,Description="Raw read depth">
##INFO=<ID=DP4,Number=4,Type=Integer,Description="# high-quality ref-forward bases, ref-re
##INFO=<ID=MQ,Number=1,Type=Integer,Description="Root-mean-square mapping quality of cover
##INFO=<ID=FQ,Number=1,Type=Float,Description="Phred probability of all samples being the
##INFO=<ID=AF1,Number=1,Type=Float,Description="Max-likelihood estimate of the first ALT a
##INFO=<ID=AC1,Number=1,Type=Float,Description="Max-likelihood estimate of the first ALT a
##INFO=<ID=AN,Number=1,Type=Integer,Description="Total number of alleles in called genotyp
##INFO=<ID=IS,Number=2,Type=Float,Description="Maximum number of reads supporting an indel
##INFO=<ID=AC,Number=A,Type=Integer,Description="Allele count in genotypes for each ALT al
##INFO=<ID=G3,Number=3,Type=Float,Description="ML estimate of genotype frequencies">
##INFO=<ID=HWE,Number=1,Type=Float,Description="Chi^2 based HWE test P-value based on G3">
##INFO=<ID=CLR,Number=1,Type=Integer,Description="Log ratio of genotype likelihoods with a
##INFO=<ID=UGT,Number=1,Type=String,Description="The most probable unconstrained genotype
##INFO=<ID=CGT,Number=1,Type=String,Description="The most probable constrained genotype co
##INFO=<ID=PV4,Number=4,Type=Float,Description="P-values for strand bias, baseQ bias, mapQ
##INFO=<ID=INDEL,Number=0,Type=Flag,Description="Indicates that the variant is an INDEL.">
##INFO=<ID=PC2,Number=2,Type=Integer,Description="Phred probability of the nonRef allele f
##INFO=<ID=PCHI2,Number=1,Type=Float,Description="Posterior weighted chi^2 P-value for tes
##INFO=<ID=QCHI2,Number=1,Type=Integer,Description="Phred scaled PCHI2.">
```

```
##INFO=<ID=PR,Number=1,Type=Integer,Description="# permutations yielding a smaller PCHI2."
##INFO=<ID=QBD,Number=1,Type=Float,Description="Quality by Depth: QUAL/#reads">
##INFO=<ID=RPB,Number=1,Type=Float,Description="Read Position Bias">
##INFO=<ID=MDV,Number=1,Type=Integer,Description="Maximum number of high-quality nonRef re
##INFO=<ID=VDB,Number=1,Type=Float,Description="Variant Distance Bias (v2) for filtering s
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GL,Number=3,Type=Float,Description="Likelihoods for RR,RA,AA genotypes (R=ref
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="# high-quality bases">
##FORMAT=<ID=DV,Number=1,Type=Integer,Description="# high-quality non-reference bases">
##FORMAT=<ID=SP,Number=1,Type=Integer,Description="Phred-scaled strand bias P-value">
##FORMAT=<ID=PL,Number=G,Type=Integer,Description="List of Phred-scaled genotype likelihoo
#CHROM    POS ID  REF ALT QUAL    FILTER  INFO    FORMAT  Ptroglodytes Ppaniscus  Ggorilla
ch01  100 .   A   .   30  .   DP=5;AF1=0;AC1=0;DP4=5,0,0,0;MQ=20;FQ=-23.4 PL:DP   0/0:0,6,
ch01  101 .   A   .   30  .   DP=5;AF1=0;AC1=0;DP4=5,0,0,0;MQ=20;FQ=-23.4 PL:DP   0/0:0,6,
ch01  102 .   A   .   30  .   DP=5;AF1=0;AC1=0;DP4=5,0,0,0;MQ=20;FQ=-23.4 PL:DP   0/0:0,6,
ch01  103 .   T   .   32  .   DP=5;AF1=0;AC1=0;DP4=5,0,0,0;MQ=20;FQ=-23.4 PL:DP   0/0:0,6,
ch01  104 .   T   C   7.61    .   DP=2;VDB=6.720000e-02;AF1=1;AC1=58;DP4=0,0,1,1;MQ=20;FQ=
ch01  105 .   G   C   32.1    .   DP=5;AF1=0;AC1=0;DP4=5,0,0,0;MQ=20;FQ=-23.4 PL:DP   0/0:
ch01  106 .   A   .   30  .   DP=5;AF1=0;AC1=0;DP4=5,0,0,0;MQ=20;FQ=-23.4 PL:DP   0:0 0:0
ch01  107 .   A   T   24.4    .   DP=5;AF1=1;AC1=58;DP4=0,0,1,0;MQ=20;FQ=-23.4    PL:DP
ch01  108 .   A   C,G 999 .   DP=52;VDB=6.361343e-02;RPB=-1.264051e+00;AF1=0.9325;AC1=54;D
```

Program Parameters

CalcAllCharacterCountPerSample

Calculates the count of different character typesin an MVF file

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--contig-ids/--contigids` = Specify comma-separated list of contig short ids. Must match exactly. Do not use with –contig-labels. (type=None, default=None)

`--contig-labels/--contiglabels` = Specify comma-separated list of contig full labels. Must match exactly. Do not use with –contig-ids (type=None, default=None)

`--mincoverage` = Mininum sample coverage for sites. (type=integer, default=None)

`--quiet` = Suppress screen output. (flag, default=False)

`--sample-indices/--sampleindices` = Specify comma-separated list of sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--windowsize` = Set integer window size. Use 0 for whole file. Use -1 for whole contigs. (flag, default=100000)

CalcCharacterCount

Calculates the count of different character typesin an MVF file

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--base-match/--basematch` = String of bases to match (i.e. numerator). (type=None, default=None)

`--base-total/--basetotal` = String of bases for total (i.e. denominator). (type=None, default=None)

`--contig-ids/--contigids` = Specify comma-separated list of contig short ids. Must match exactly. Do not use with –contig-labels. (type=None, default=None)

`--contig-labels/--contiglabels` = Specify comma-separated list of contig full labels. Must match exactly. Do not use with –contig-ids (type=None, default=None)

`--mincoverage` = Mininum sample coverage for sites. (type=integer, default=None)

`--quiet` = Suppress screen output. (flag, default=False)

`--sample-indices/--sampleindices` = Specify comma-separated list of sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--windowsize` = Set integer window size. Use 0 for whole file. Use -1 for whole contigs. (flag, default=100000)

CalcDstatCombinations

Calculates all D-statistics for all combinations ofspecified taxa in an MVF file.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--contig-ids/--contigids` = Specify comma-separated list of contig short ids. Must match exactly. Do not use with –contig-labels. (type=None, default=None)

`--contig-labels/--contiglabels` = Specify comma-separated list of contig full labels. Must match exactly. Do not use with –contig-ids (type=None, default=None)

`--outgroup-indices/--outgroupindices` = Specify comma-separated list of outgroup sample numerical indices (first column is 0). Leave blank for all samples. Do not use with –outgroup_labels. (type=None, default=None)

`--outgroup-labels/--outgrouplabels` = Specify comma-separated list of outgroup sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –outgroup_indicies. (type=None, default=None)

`--quiet` = Suppress screen output. (flag, default=False)

`--sample-indices/--sampleindices` = Specify comma-separated list of 3 or more sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of 3 or more sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

CalcPairwiseDistances

Calculates pairwise sequence distances for combinations ofspecified taxa in an MVF file.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--ambig` = By default, ambiguous nucleotides are excluded. This option will include sets of ambiguous characters by randomly choosing one of the options

for: RYMKWS ('random2') or RYMKWS+BDHV ('random3') (type=None, default=None) Choices: ('random2', 'random3')

`--data-type/--datatype` = Data type to compare.(This option is only needed for codon MVF files, others will default.) (type=None, default=None) Choices: ('dna', 'prot')

`--emit-counts` = output additional file that presents the raw counts of pairwise patterns for each sample pair tested for each window (flag, default=False)

`--mincoverage` = Mininum sample coverage for sites. (type=integer, default=None)

`--quiet` = Suppress screen output. (flag, default=False)

`--sample-indices/--sampleindices` = Specify comma-separated list of 2 or more sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of 2 or more sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--windowsize` = Set integer window size. Use 0 for whole file. Use -1 for whole contigs. (flag, default=100000)

CalcPatternCount

Counts biallelic site patternsn (AB-patterns) forspecified combinations of taxa in an MVF file.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--mincoverage` = Mininum sample coverage for sites. (type=integer, default=None)

`--output-lists` = None (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

`--sample-indices/--sampleindices` = Specify comma-separated list of sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--windowsize` = Set integer window size. Use 0 for whole file. Use -1 for whole contigs. (flag, default=100000)

CalcSampleCoverage

Counts per-contig coverage forspecified sample columns in an MVF file.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--contig-ids/--contigids` = Specify comma-separated list of contig short ids. Must match exactly. Do not use with –contig-labels. (type=None, default=None)

`--contig-labels/--contiglabels` = Specify comma-separated list of contig full labels. Must match exactly. Do not use with –contig-ids (type=None, default=None)

`--quiet` = Suppress screen output. (flag, default=False)

`--sample-indices/--sampleindices` = Specify comma-separated list of sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

ConcatenateMVF

Combine non-overlapping contigs from one or more MVF files into asingle MVF file. This does NOT merge columns.Use MergeMVF to merge sample columns from multiple files.

Parameters

`--mvf` (required) = One or more mvf files. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--line-buffer/--linebuffer` = Number of entries to store in memory at a time. (type=integer, default=100000)

`--main_header_file/--mainheaderfile` = Output file will use same headers as this input file (default=first in list). (type=None, default=None)

`--new-contigs/--newcontigs` = By default, contigs are matched between files using their text labels in the header. Use this option to turn matching off and treat each file's contigs as distinct. (flag, default=False)

`--newsamples` = By default, samples are matched between files using their text labels in the header. Use this option to turn matching off and treat each file's sample columns as distinct. (flag, default=False)

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

ConvertFasta2MVF

Converts a FASTA file to MVF format

Parameters

`--fasta` (required) = input FASTA file(s) (type=None, default=None)

`--out` (required) = output MVF file (type=None, default=None)

`--contig-by-file/--contigbyfile` = Contigs are designated by separate files. (flag, default=False)

`--contig-field/--contigfield` = When headers are split by –field-sep, the 0-based index of the contig id. (type=integer, default=None)

`--contig-ids/--contigids` = manually specify one or more contig ids as ID:LABEL (type=None, default=None)

`--field-sep/--fieldsep` = FASTA field separator; assumes '>database accession locus' format (type=None, default=None) Choices: ['TAB', 'SPACE', 'DBLSPACE', 'COMMA', 'MIXED', 'PIPE', 'AT', 'UNDER', 'DBLUNDER']

`--flavor` = type of file [dna] or protein (type=None, default=dna) Choices: ['dna', 'protein']

`--manual-coord/--manualcoord` = manually specify reference coordinates for each file in the format CONTIGID:START..STOP, ... (type=None, default=None)

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

`--read-buffer/--readbuffer` = number of lines to hold in READ buffer (type=integer, default=100000)

`--ref-label/--reflabel` = label for reference sample (type=None, default=REF)

`--sample-field/--samplefield` = when headers are split by –field-sep, the 0-based index of the sample id (type=integer, default=None)

`--sample-replace/--samplereplace` = one or more TAG:NEWLABEL or TAG, items, if TAG found in sample label, replace with NEW (or TAG if NEW not specified) NEW and TAG must each be unique (type=None, default=None)

`--write-buffer/--writebuffer` = number of lines to hold in WRITE buffer (type=integer, default=100000)

ConvertMAF2MVF

Converts a MAF file to a MVF file

Parameters

`--maf` (required) = input MAF file (type=file path, default=None)

`--out` (required) = output MVF file (type=file path, default=None)

`--ref-tag/--reftag` (required) = Specify which TAG in –sample-tags is the reference genome. (type=None, default=None)

`--sample-tags/--sampletags` (required) = One or more TAG:NEW or TAG, items separated by commas.Each TAG is partial text-matched to the sample labels in the MAF. For example, hsap18.chr1 and hsap18.chr2 would be matched tag 'hsap18'. If :NEW is added, then the MVF sample will be labeled NEW. Otherwise, the sample will be labeled simply TAG. (type=None, default=None)

`--line-buffer/--linebuffer` = Number of entries to store in memory at a time. (type=integer, default=100000)

`--mvf-ref-label/--mvfreflabel` = new label for reference sample (default='REF') (type=None, default=REF)

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

ConvertMVF2Fasta

Converts an MVF file to a FASTA file

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output path of FASTA file. (type=file path, default=None)

`--buffer` = size (Mbp) of write buffer for each sample (type=integer, default=10)

`--gene-mode` = None (flag, default=False)

`--label-type/--labeltype` = Long labels with all metadata or short ids (type=None, default=long) Choices: ('long', 'short')

`--output-data/--outputdata` = Output dna, rna or prot data. (type=None, default=None) Choices: ('dna', 'rna', 'prot')

`--quiet` = Suppress screen output. (flag, default=False)

`--regions` = Path of a plain text file containing one more lines with entries 'contigid,stop,start' (one per line, inclusive coordinates) all data will be returned if left blank. (type=file path, default=None)

`--sample-indices/--sampleindices` = Specify comma-separated list of sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--temp_dir/--tempdir` = directory to write temporary fasta files (type=None, default=.)

ConvertMVF2FastaGene

Converts an MVF file to a set ofFASTA files per gene

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--output-dir` (required) = Output directory of FASTA files. (type=file path, default=None)

`--buffer` = size (Mbp) of write buffer for each sample (type=integer, default=10)

`--choose-allele/--chooseallele` = Chooses how heterozygous alleles are handled. (none=no splitting (default); random1=pick one allele randomly (type=None, default=none) Choices: ['none', 'random1']

`--ignore-strand` = Do not read strand info from contigs (flag, default=False)

`--output-data/--outputdata` = Output dna, rna or prot data. (type=None, default=None) Choices: ('dna', 'rna', 'prot')

`--quiet` = Suppress screen output. (flag, default=False)

`--sample-indices/--sampleindices` = Specify comma-separated list of sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--temp_dir/--tempdir` = directory to write temporary fasta files (type=None, default=.)

ConvertMVF2Phylip

Converts an MVF file to a Phylip file

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output Phylip file. (type=file path, default=None)

`--buffer` = size (bp) of write buffer for each sample (type=integer, default=100000)

`--label-type/--labeltype` = Long labels with all metadata or short ids (type=None, default=short) Choices: ('long', 'short')

`--output-data/--outputdata` = Output dna, rna or prot data. (type=None, default=None) Choices: ('dna', 'rna', 'prot')

`--partition` = Output a CSV partitions file with RAxMLformatting for use in partitioned phylogenetic methods. (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

`--regions` = Path of a plain text file containing one more lines with entries 'contigid,stop,start' (one per line, inclusive coordinates) all data will be returned if left blank. (type=file path, default=None)

`--sample-indices/--sampleindices` = Specify comma-separated list of sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--temp_dir/--tempdir` = directory to write temporary fasta files (type=None, default=.)

ConvertVCF2MVF

Converts a VCF file to an MVF file

Parameters

`--out` (required) = output MVF file (type=None, default=None)

`--alleles-from/--allelesfrom` = get additional alignment columns from INFO fields (:-separated) (type=None, default=None)

`--contig-ids/--contigids` = manually specify one or more contig ids as ID;VCFLABE;MVFLABEL, note that VCFLABEL must match EXACTLY the contig string labels in the VCF file (type=None, default=None)

`--field-sep/--fieldsep` = VCF field separator (default='TAB') (type=None, default=TAB) Choices: ['TAB', 'SPACE', 'DBLSPACE', 'COMMA', 'MIXED']

`--filter-nonref-empty` = Do not output entries that are masked or empty for all samples (not the reference). (flag, default=False)

`--line-buffer/--linebuffer` = Number of entries to store in memory at a time. (type=integer, default=100000)

`--low-depth/--lowdepth` = below this read depth coverage, convert to lower case set to 0 to disable (type=integer, default=3)

`--low-qual/--lowqual` = below this quality convert to lower case set to 0 to disable (type=integer, default=20)

`--mask-depth/--maskdepth` = below this read depth mask with N/n (type=integer, default=1)

`--mask-qual/--maskqual` = low quality cutoff, bases replaced by N/- set to 0 to disable (type=integer, default=3)

`--no-autoindex/--noautoindex` = do not automatically index contigs from the VCF (flag, default=False)

`--out-flavor/--outflavor` = choose output MVF flavor to include quality scores and/or indels (type=None, default=dna) Choices: ['dna', 'dnaqual', 'dnaqual-indel', 'dna-indel']

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--ploidy` = Use for hexaploid and tetraploid (Experimental, use with caution (type=integer, default=2) Choices: (2, 4, 6)

`--qual` = Include Phred genotype quality (GQ) scores (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

`--ref-label/--reflabel` = label for reference sample (default='REF') (type=None, default=REF)

`--sample-replace/--samplereplace` = one or more TAG:NEWLABEL or TAG, items, if TAG found in sample label, replace with NEW (or TAG if NEW not specified) NEW and TAG must each be unique (type=None, default=None)

`--skip-contig-label-check` = When there are many contigs skip checking for repeat labels. (use with caution). (flag, default=False)

`--vcf` = VCF input file (type=file path, default=None)

`--verbose` = Output excessive data to screen for debugging (flag, default=False)

FilterMVF

Filter an MVF file using various parameters.

Parameters

`--actions` = set of actions:args to perform, note these are done in order as listed (type=None, default=None)

`--labels` = use sample labels instead of indices (flag, default=False)

`--line-buffer/--linebuffer` = Number of entries to store in memory at a time. (type=integer, default=100000)

`--more-help/--morehelp` = prints full module list and descriptions (flag, default=False)

`--mvf` = Input MVF file. (type=file path, default=None)

`--out` = Output file (type=file path, default=None)

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

`--retain-empty/--retainempty` = keep empty entries during filtering (flag, default=False)

`--test` = manually input a line for testing (type=None, default=None)

`--test-nchar/--textnchar` = total number of samples for test string (type=integer, default=None)

`--verbose` = report every line (for debugging) (flag, default=False)

InferGroupSpecificAllele

Infer Group-specific alleles using PAML.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--all-sample-trees/--allsampletrees` = Makes trees from all samples instead of only the most complete sequence from each species (flag, default=False)

`--allele-groups/--allelegroups` = GROUP1:LABEL,LABEL GROUP2:LABEL,LABEL (type=None, default=None)

`--branch-lrt/--branchlrt` = Specify the output file for and turn on the RAxML-PAML format LRT test scan for selection on the target branch in addition to the basic patterns scan (type=file path, default=None)

`--chi-test/--chitest` = Input two number values for expected Nonsynonymous and Synonymous expected values. (type=None, default=None)

`--codeml-path/--codemlpath` = Full path for PAML codeml executable. (type=file path, default=codeml)

`--end-contig/--endcontig` = Numerical id for the ending contig. (type=integer, default=100000000)

`--gff` = Input gff annotation file. (type=file path, default=None)

`--mincoverage` = Mininum sample coverage for sites. (type=integer, default=None)

`--num-target-species/--targetspec` = Specify the minimum number of taxa in the target set that are required to conduct analysis (type=integer, default=1)

`--outgroup` = Specify sample name with which to root trees. (type=None, default=None)

`--output-align/--outputalign` = Output alignment to this file path in phylip format. (type=None, default=None)

`--paml-tmp/--pamltmp` = path for temporary folder for PAML output files (type=file path, default=pamltmp)

`--quiet` = Suppress screen output. (flag, default=False)

`--raxml-path/--raxmlpath` = Full path to RAxML program executable. (type=file path, default=raxml)

`--species-groups/--speciesgroups` = None (type=None, default=None)

`--start-contig/--startcontig` = Numerical ID for the starting contig. (type=integer, default=0)

`--target` = Specify the taxa labels that define the target lineage-specific branch to be tested. (type=None, default=None)

`--use-labels/--uselabels` = Use contig labels instead of IDs in output. (flag, default=False)

`--verbose` = additional screen output (flag, default=False)

`--windowsize` = Set integer window size. Use 0 for whole file. Use -1 for whole contigs. (flag, default=100000)

InferTree

Infer phylogenies for various windows or contigs in anMVF file.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--bootstrap` = turn on rapid bootstrapping for RAxML and perform specified number of replicates (type=integer, default=None)

`--choose-allele/--chooseallele/--hapmode` = Chooses how heterozygous alleles are handled. (none=no splitting (default); randomone=pick one allele randomly (recommended); randomboth=pick two alleles randomly, but keep both; major=pick the more common allele (type=None, default=none) Choices: [‘none’, ‘randomone’, ‘randomboth’]

`--contig-ids/--contigids` = Specify comma-separated list of contig short ids. Must match exactly. Do not use with –contig-labels. (type=None, default=None)

`--contig-labels/--contiglabels` = Specify comma-separated list of contig full labels. Must match exactly. Do not use with –contig-ids (type=None, default=None)

`--duplicate-seq/--duplicateseq` = dontuse=remove duplicate sequences prior to RAxML tree inference, then add them to the tree manually as zero-branch-length sister taxa; keep=keep in for RAxML tree inference (may cause errors for RAxML); remove=remove entirely from alignment (type=None, default=dontuse) Choices: [‘dontuse’, ‘keep’, ‘remove’]

`--min-depth/--mindepth` = minimum number of alleles per site (type=integer, default=4)

`--min-seq-coverage/--minseqcoverage` = proportion of total alignment a sequencemust cover to be retianed [0.1] (type=float, default=0.1)

`--min-sites/--minsites` = minimum number of sites (type=integer, default=100)

`--output-contig-labels/--outputcontiglabels` = Output will use contig labels instead of id numbers. (flag, default=False)

`--output-empty/--outputempty` = Include entries of windows with no data in output. (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

`--raxml-model/--raxmlmodel` = choose RAxML model (type=None, default=GTRGAMMA)

`--raxml-opts/--raxmlopts` = specify additional RAxML arguments as a double-quotes encased string (type=None, default=)

`--raxml-outgroups/--raxmloutgroups` = Comma-separated list of outgroup taxon labels to use in RAxML. (type=None, default=None)

`--raxml-path/--raxmlpath` = RAxML path for manual specification. (type=None, default=raxml)

`--root-with/--rootwith` = Comma-separated list of taxon labels to root trees with after RAxML (type=None, default=None)

`--sample-indices/--sampleindices` = Specify comma-separated list of sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--temp-dir/--tempdir` = Temporary directory path (type=file path, default=./raxmltemp)

`--temp-prefix/--tempprefix` = Temporary file prefix (type=None, default=mvftree)

`--windowsize` = Set integer window size. Use 0 for whole file. Use -1 for whole contigs. (flag, default=100000)

MergeMVF

Combines columns from multiple MVF files into a single output MVF(this is a newer module, use with caution!)

Parameters

`--mvf` (required) = One or more mvf files. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--line-buffer/--linebuffer` = Number of entries to store in memory at a time. (type=integer, default=100000)

`--main_header_file/--mainheaderfile` = Output file will use same headers as this input file (default=first in list). (type=None, default=None)

`--new-contigs/--newcontigs` = By default, contigs are matched between files using their text labels in the header. Use this option to turn matching off and treat each file's contigs as distinct. (flag, default=False)

`--newsamples` = By default, samples are matched between files using their text labels in the header. Use this option to turn matching off and treat each file's sample columns as distinct. (flag, default=False)

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

`--skip-index/--skipindex` = Skip index because index exists (flag, default=False)

PlotChromoplot

Plot a Chromoplot from an MVF file for all combinationsof the specified samples.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--colors` = three colors to use for chromoplot (type=None, default=None) Choices: {'lgrey': (250, 250, 250), 'dgrey': (192, 192, 192), 'black': (0, 0, 0), 'white': (255, 255, 255), 'red': (192, 0, 0), 'orange': (217, 95, 2), 'yellow': (192, 192, 0), 'green': (0, 192, 0), 'blue': (0, 0, 192), 'teal': (27, 158, 119), 'puce': (117, 112, 179), 'purple': (192, 0, 192), 'none': ()}

`--contig-ids/--contigids/--contigs` = Enter the labels of one or more contigs in the order they will appear in the chromoplot (as comma-separated list)(defaults to all ids in order present in MVF) (type=None, default=None)

`--contig-labels/--contiglabels` = Enter the ids of one or more contigs in the order they will appear in the chromoplot (as comma-separated list)(defaults to all ids in order present in MVF) (type=None, default=None)

`--empty-mask/--emptymask` = Mask empty regions with this color. (type=None, default=none) Choices: {'lgrey': (250, 250, 250), 'dgrey': (192, 192, 192), 'black': (0, 0, 0), 'white': (255, 255, 255), 'red': (192, 0, 0), 'orange': (217, 95, 2), 'yellow': (192, 192, 0), 'green': (0, 192, 0), 'blue': (0, 0, 192), 'teal': (27, 158, 119), 'puce': (117, 112, 179), 'purple': (192, 0, 192), 'none': ()}

`--info-track/--infotrack` = Include an additional coverage information track

that will show empty, uninformative, and informative loci. (Useful for ranscriptomes/RAD or other reduced sampling. (flag, default=False)

`--majority` = Plot only 100% shading in the majority track rather than shaded proportions in all tracks. (flag, default=False)

`--out-prefix/--outprefix` = Output prefix (not required). (type=None, default=None)

`--outgroup-indices/--outgroupindices` = Specify comma-separated list of 1 or more outgroup sample numerical indices (first column is 0). Leave blank for all samples. Do not use with –outgroup_labels. (type=None, default=None)

`--outgroup-labels/--outgrouplabels` = Specify comma-separated list of 1 or more outgroup sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –outgroup_indicies. (type=None, default=None)

`--plot-type/--plottype` = PNG image (default) or graph via matplotlib (experimental) (type=None, default=image) Choices: ['graph', 'image']

`--quiet` = Suppress screen output. (flag, default=False)

`--sample-indices/--sampleindices` = Specify comma-separated list of 3 or more sample numerical indices (first sample is 0). Leave blank for all samples. Do not use with –sample_labels. (type=None, default=None)

`--sample-labels` = Specify comma-separated list of 3 or more sample labels. Labels must be exact (case-sensitive). Leave blank for all samples.Do not use with –sample_indicies. (type=None, default=None)

`--windowsize` = Set integer window size. Use 0 for whole file. Use -1 for whole contigs. (flag, default=100000)

`--xscale` = Width (in number of pixels) for each window (type=integer, default=1)

`--yscale` = Height (in number of pixels) for each track (type=integer, default=20)

TranslateMVF

Annotates a chromosomal MVF file with new contigboundaries based on genes/features from a GFF file.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--filter-annotation/--filterannotation` = skip GFF entries with text matching this in their 'Notes' field (type=None, default=None)

`--gene-pattern/--genepattern` = Gene name pattern finder when interpreting GFF/GTF. Use % in place of gene name. (type=None, default=gene_id "%")

`--gff` = Input gff annotation file. (type=file path, default=None)

`--line-buffer/--linebuffer` = Number of entries to store in memory at a time. (type=integer, default=100000)

`--non-genic-margin/--nongenicmargin` = For —non-genic-mode, pad the boundaries of unannotated regions by this amount. (type=integer, default=0)

`--non-genic-mode/--nongenicmode` = Instead of returning annotated genes, return the non-genic regions without changing contigs or coordinates. (flag, default=False)

`--output-data/--outputdata` = dna=single data column of dna alleles; protein=single data column of protein alleles; codon=four columns with: protein frame1 frame2 frame3 (type=None, default=codon) Choices: ['dna', 'protein', 'codon']

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

`--require-annotation/--requireannotation` = require GFF entries with text matching this in their 'Notes' field (type=None, default=None)

`--retain-contigs` = maintain original contig numbering (flag, default=False)

`--retain-coords` = maintain original coordinates (flag, default=False)

VerifyMVF

Checks an MVF file for errors.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--quiet` = Suppress screen output. (flag, default=False)

LegacyAnnotateMVF

This is deprecated now, but maintained for legacy functions.Use TranslateMVF with –output-data dna to annotate regions.Annotates a chromosomal MVF file with new contigboundaries based on genes/features from a GFF file.

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--filter-annotation/--filterannotation` = Skip entries in the GFF file that contain this string in their 'Notes' (type=None, default=None)

`--gene-pattern/--genepattern` = Gene name pattern finder when interpreting GFF/GTF. Use % in place of gene name. (type=None, default=gene_id "%")

`--gff` = Input gff annotation file. (type=file path, default=None)

`--line-buffer/--linebuffer` = Number of entries to store in memory at a time. (type=integer, default=100000)

`--nongenic-margin/--nongenicmargin` = for –nongenic-mode, only retain positions that are this number of bp away from an annotated region boundary (type=integer, default=0)

`--nongenic-mode/--nongenicmode` = Instead of returning annotated genes, return the non-genic regions without without changing contigs or coordinates (flag, default=False)

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--quiet` = Suppress screen output. (flag, default=False)

LegacyTranslateMVF

Note this is deprecated now, but maintained for legacy function.Use TranslateMVF with '–output-data protein'or'–output-data codon.'Translate a DNA MVF to a protein or codon MVF

Parameters

`--mvf` (required) = Input MVF file. (type=file path, default=None)

`--out` (required) = Output file (type=file path, default=None)

`--filter-annotation/--filterannotation` = skip GFF entries with text matching this in their 'Notes' field (type=None, default=None)

`--gff` = Input GFF3 file. If GFF3 not provided, alignments are assumed to be in-frame coding sequences. (type=file path, default=None)

`--line-buffer/--linebuffer` = Number of entries to store in memory at a time. (type=integer, default=100000)

`--output-data/--outputdata` = protein=single data column of protein alleles; codon=four columns with: protein frame1 frame2 frame3 (type=None, default=codon) Choices: ['protein', 'codon']

`--overwrite` = USE WITH CAUTION: force overwrite of outputs (flag, default=False)

`--parent-gene-pattern/--parentgenepattern` = Parent genes prefix when interpretingGFF files. For GFF3 files, 'gene:' is standard, but for older or custom GFF files this may vary. Use 'none' to make empty. (type=None, default=gene_id "%")

`--quiet` = Suppress screen output. (flag, default=False)

`--require-annotation/--requireannotation` = require GFF entries with text matching this in their 'Notes' field (type=None, default=None)

`--verbose` = Output excessive data to screen for debugging (flag, default=False)

Retired Modules

- JoinMVF > Use ConcatenateMVF instead.
- CheckMVF > Use VerifyMVF instead.

Version History

v.0.6.1

2021-04-20: Updates to fix problems with the ConvertMAF2MVF Module. Note that commas now separate the –sample-tags and –ref-tag is now required.

v.0.6.0

2020-12-20: Major Update - Major update to the back-end of MVF for speed and stability. Adds support for faster file access by creating an MVF index, fixes to the filtering modules, changes to the specification of sample and contig labels to improve usability. Major improvements to the MVFTranslate module. Added support for tetraploid and hexaploid VCF files. Many other fixes

v.0.5.4

2019-07-16: Adds the CalcAllCharacterCountPerSample function, continued upgrades to the screen output to provide more realtime information. Other small fixes to the CalcPairwiseDistances module in ambiguous character mode.

v.0.5.3

2019-07-14: Critial update, strongly recommend updating to this version. Major efficiency fix in the base iteration modules. Several key bug fixes implemented in FilterMVF, MergeMVF. Enhanced support for ambiguous sequences and polyploids in several modules including CalcPairwiseDistance. Restructuring of FilterMVF for cleaner syntax.

v.0.5.2

2019-06-29: Added MergeMVF to join several files together (still experimental, use with caution). JoinMVF is now called ConcatenateMVF to avoid confusion. CheckMVF changed to VerifyMVF to make it more clear. ConvertVCF2MVF now has experimental support for tetraploid and hexaploid VCF files through the –ploidy flag. Other small fixes to the software and manual to update issues with the VCF interpreter.

v.0.5.1

2018-02-01: Changes to the –sample and –outgroup arguments for some calculations into separate –sample-indices and –sample-labels arguments. This fixes an issue where if the sample labels are numerical they are misinterpreted when specified at the command line. All sample/outgroup indices or labels should be specified as a single comma-separated list.

v.0.5.0

2017-11-27 - Major Upgrade: Change to single-command structure

v.2017-06-25

Major Upgrade: Full manual documentation added, standardization and cleanup of paramaters and upgrades and bugfixes throughout.

v.2017-05-18

Fixes to VCF conversion for compatibility

v.2017-04-10

Added MVF-to-Phylip output conversion `mvf2phy`

v.2017-03-25

Multiple bug fixes, merged and removed the development instance

v.2016-02-15

Fix to vcf2mvf for VCF with truncated entries

v.2016-10-25

Efficiency upgrades for mvfbase entry iteration.

v.2016-09-10

Minor fixes to gz reading and MVF chromoplot shading

v.2016-08-02

Python3 conversion, integrate analysis_base

v.2016-01-11

fix for dna ambiguity characters

v.2016-01-01

Python3 compatiblity fix

v.2015-12-31

Header changesand cleanup

v.2015-12-15

Python3 compatibilty fix

v.2015-09-04

Small style fixes

v.2015-06-09

MVF1.2.1 upgrade

v.2015-02-26

Efficiency upgrades for iterators

v.2015-02-01

First Public Release

License