

Detailed Instructions for the Mutation Finder and Annotator with Relatives Program

January 26, 2024

OVERVIEW: This program is designed to identify single nucleotide biallelic variants from a multi-sample VCF file, taking into account the genetic relationship of individual samples. It was built to identify induced point mutations, for example from TILLING populations, but will work with natural variants as well.

HOW IT WORKS: When launched, the user is asked for the name of a new directory that will house the results. This name should be unique. The compressed multi-sample VCF file is next chosen followed by a table of samples and their relationship. **NOTE:** The program is built for up to 6 replicates per group of related samples.

Input file (from example data)

AH1CC3SS55_S55	A	A
AH1CC3SS28_S28	A	A
AH1CC3SS105_S105	C	C

Definitions:

Replicates: A technical or biological replicate. A true induced mutations should be present in both replicates. This criteria is used when filtering the VCF. Replicates are assigned the same letter. Different letters are used for different sets of replicates or samples that are not replicated (AH1CC3SS105_S105 in this example).

Relatives: Samples that are genetically related after the induction of mutations. Along with replicates, this can relations such as siblings or parents. Thus, two individuals may share some true induced mutations, but in non-replicate relations two or more individuals need not share all induced mutations. If relatives are present, filtering is performed such that unique mutations are identified when comparing non-related samples. For each related individual, resulting induced mutations are merged into one new VCF and the total number of combined unique mutations is reported for that set of genetically related samples (e.g. a line).

Why use this rather than the original MFA program?

The original MFA program was built to analyze a single sample at a time and only took into account replicates. This means that analysis of many mutant lines requires many runs of the program. In addition if non-replicate genetically related samples are present (e.g. siblings) the program requires that the siblings are removed from the analysis. The MFAR program was developed to address these shortfalls. If you only care about one sample and only have replicates, try the MFA. If you are dealing with more samples, or have more complex genetically relationships, try this program.

Output directories produced by the program and their contents



BCftools_Stats

Contains bcftools stats files for VCFs containing unique mutations for each set of relatives



ForTrash

Contains temporary directories you can delete. These are not automatically deleted as a precaution when testing the program. You can add `rm -r ForTrash` when you convince yourself the program is behaving properly.



Merged_AnnotatedVCFs

New VCFs annotated with snpEff are created for each set of related samples. See the following pages for examples.



NonMerged_AnnotatedVCFs

Annotated VCFs of unique individuals for each sample prior to any merging. These are retained so that you can control how mutations are counted in cases of complicated sample relationships.



snpEff_genes_summary

Contains the snpEff genes.txt and html report outputs for each non-merged VCF containing unique mutations.



MFAR.log

The log file records the input sample file, the path to the VCF used, the snpEff genome used, paths to the .jar file and reports on if it found replicates.



RelativesVariantCount_01_26_2024.text

A table listing genetically related samples and the number of unique mutations identified. This can be used to calculate mutation frequency.

Test data and specific examples

Files provided in the example data directory for testing:

CoffeeMutants_NC_0399181.1_ThreesampleSNPonly.vcf.gz = the test VCF that contains a subset of data for three samples.

Scenario1 = An example input sample file where all samples are coded genetically different

Scenario2 = Example input sample file where the first two samples are replicates

Scenario3 = Example input sample file where the first to samples are related but not replicates

Scenario4 = Example input sample file where replicate and genetic relationship is wrongly coded

NOTE: To test this program you need to create a snpEff genome database for Coffee. Details on how to do this are found in the publication: “Identification of Novel Induced Mutations in Seed and Vegetatively Propagated Plants from Reduced Representation or Whole Genome Sequencing Data” that, when published, will appear in the github readme.

The next four pages describe each Scenario listed above

Scenario1 – No replicates and no genetic relationships

Input file:

```
AH1CC3SS55_S55 A      A
AH1CC3SS28_S28 B      B
AH1CC3SS105_S105 C    C
```

Expected behavior:

There are no replicates or otherwise genetically related samples. Therefore the program will treat each sample independently. Mutations unique to each of the three samples (those not found in the other 2) will be retained and counted. The merged and nonmerged directories will each contain three VCFs, BCFtools_Stats will contain a report for each sample, as will the snpEff_genes_summary directory. The RelativesVariantCount table contains a count for each sample

Output table:

```
Related_Samples Total_Unique_Biallelic_SNVs
AH1CC3SS105_S105 12
AH1CC3SS28_S28 17
AH1CC3SS55_S55 21
```

Log file:

```
Mutation Finder and Annotator with Relatives (MFAR) GUI, Version 1.7
Script Started Fri 26 Jan 2024 11:05:36 AM PST.
Checking the headers and starting positions of 1 files
Checking the headers and starting positions of 1 files
Checking the headers and starting positions of 1 files
Program finished Fri 26 Jan 2024 11:07:17 AM PST.
```

SNPeff genome used: Coffee

Path to VCF file used:

```
/home/brad/Documents/Mandana/MFA_Siblingtest/CoffeeMutants_NC_0399181.1_ThreesampleS
NPonly.vcf.gz
```

Path to SnpSIFT.jar: /home/brad/snpEff/SnpSift.jar

Path to SnpEFF.jar: /home/brad/snpEff/snpEff.jar

Samples selected with replicate and relatives relationships:

```
AH1CC3SS55_S55 A      A
AH1CC3SS28_S28 B      B
AH1CC3SS105_S105 C    C
```

Status of Replicates:

```
No replicates identified in replicate group A
No replicates identified in replicate group B
No replicates identified in replicate group C
```

Scenario2 – Replicates but no other genetic relationships

Input file:

```
AH1CC3SS55_S55 A    A
AH1CC3SS28_S28 A    A
AH1CC3SS105_S105 B   B
```

Expected behavior:

Samples S55 and S28 are replicates, so they are both designated A in the second column. Replicates are genetically related and so the samples are also given an A in the third column. Sample S105 is not related to any sample, and so is assigned a different unique letter in columns 2 and 3. The program will first compare S55 and S28 and retain only mutations that are common between the two. Next, it will compare this data against S105 and retain only variants that are not present in that sample. It will do this for all non-related samples. Sample S105 is compared to all other samples and only variants unique to that sample are retained.

Because there are related samples, the BCFtools_Stats directory and Merged_AnnotatedVCFs directory will contain files for each related set (two in this case, A and B). The filenames contain the names of samples in each set. SnpEff summary directory contains reports for only one of two replicates.

Output table:

```
Related_Samples Total_Unique_Biallelic_SNVs
AH1CC3SS105_S105 12
AH1CC3SS55_S55_AH1CC3SS28_S28 2
```

Log file:

```
Mutation Finder and Annotator with Relatives (MFAR) GUI, Version 1.7
Script Started Fri 26 Jan 2024 06:42:54 PM PST.
Checking the headers and starting positions of 2 files
Checking the headers and starting positions of 1 files
Program finished Fri 26 Jan 2024 06:44:08 PM PST.
```

SNPeff genome used: Coffee

Path to VCF file used:

/home/brad/Documents/Mandana/MFA_Siblingtest/CoffeeMutants_NC_0399181.1_ThreesampleS
NPonly.vcf.gz


Path to SnpSIFT.jar: /home/brad/snpEff/SnpSift.jar

Path to SnpEFF.jar: /home/brad/snpEff/snpEff.jar

Samples selected with replicate and relatives relationships:

```
AH1CC3SS55_S55 A    A
AH1CC3SS28_S28 A    A
AH1CC3SS105_S105 B   B
```

Note: The program accepts up to 6 replicates per genetically related group. If you have more, remove them.



Status of Replicates:

Acceptable Number of Replicates Selected in Replicate Group A
No replicates identified in replicate group B

Scenario3 – No replicates but a genetic relationship

Input file:

```
AH1CC3SS55_S55 A      A
AH1CC3SS28_S28 B      A
AH1CC3SS105_S105 C     B
```

Expected behavior:

Samples S55 and S28 are related but replicates (they have different letters in column 2, but the same letter in column 3). So they may share some induced mutations but not all. The program compares S55 with S105 and retains unique variants in S55. Next, the program compares S28 with S105 and retains unique variants in S28. To determine the total number of unique in the mutant line of which S55 and S28 are members, the unique variants in S55 and S28 are merged into a new VCF and a variant count is conducted. S105 is compared to all other samples to recover mutations unique to S105. The directory Merged_AnnotatedVCFs therefore contains two VCFs one for S55_S28 (with both names in the filename) and one for S105. The same is true for BCFtools_Stats. SnpEff reports are generated for all three individuals.

Output table:

```
Related_Samples Total_Unique_Biallelic_SNVs
AH1CC3SS105_S105 12
AH1CC3SS55_S55_AH1CC3SS28_S28 40
```

Log file:

```
Mutation Finder and Annotator with Relatives (MFAR) GUI, Version 1.7
Script Started Fri 26 Jan 2024 11:10:54 AM PST.
Checking the headers and starting positions of 2 files
Checking the headers and starting positions of 1 files
Program finished Fri 26 Jan 2024 11:12:21 AM PST.
```

SNPeff genome used: Coffee

Path to VCF file used:

/home/brad/Documents/Mandana/MFA_Siblingtest/CoffeeMutants_NC_0399181.1_ThreesampleS
NPonly.vcf.gz

Path to SnpSIFT.jar: /home/brad/snpEff/SnpSift.jar

Path to SnpEFF.jar: /home/brad/snpEff/snpEff.jar

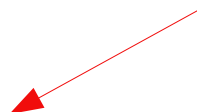
Samples selected with replicate and relatives relationships:

```
AH1CC3SS55_S55 A      A
AH1CC3SS28_S28 B      A
AH1CC3SS105_S105 C     B
```

There are no replicates in this scenario and the program agrees.

Status of Replicates:

```
No replicates identified in replicate group A
No replicates identified in replicate group B
No replicates identified in replicate group C
```



Scenario4 – An example of human error!

Input file:

```
AH1CC3SS55_S55 A    A
AH1CC3SS28_S28 A    B
AH1CC3SS105_S105 B   C
```

Expected behavior:

In this scenario the user made a mistake. Sample S55 and S28 are coded as replicates (both with the letter A in the second column), however in the relationship column (3) S55 and S28 are coded as different (A and B). This is not possible and the program gets confused. Instead of reporting unique variants for the S55_S28 set as in scenario 2, it reports each separately, but the count for the set is correct. VCF merging also fails as does BCFtools_Stats. SnpEff outputs perform as expected for scenario 2.

CONCLUSIONS:

If given an impossible combination of replicates and genetic relationships, the program will not report things properly, and could result in incorrect reporting of mutations, depending on what you do with the data. Pay attention when preparing your three column sample table.

RULE TO FOLLOW:

Biological or technical replicates are by definition genetically related. Therefore, when creating the sample input table, if two or more samples share the same letter in column 2, they should also share the same letter in column 3. NOTE: The letters in column 2 and 3 need not be the same, it just works out this way with a small example set. For example you could have two samples in column 2 having the letter Q. The letter in the third column needs to be identical (for example T), and these letters need to be unique to these samples.