The latest version of the VCF2Dis, scripts and datasets generated in this manuscript were uploaded.

- 01.TestBySite ## benchmarking test along with the increase of the number of SNP sites
  - M*.vcf.gz ### test dataset
  - RunA.sh ### step1: unzip the test data files
  - RunB.sh ### step2: command-line to run VCF2Dis
  - RunC.sh ### step2: command-line to run fastreeR
  - aa.r ### Rscript for fastreeR used in RunC.sh
  - md5sum.txt

- 02.TestBySample # benchmarking test along with the incrase of the number of samples
  - *.vcf.gz ### test dataset
  - RunA.sh ### step1: unzip the test data files
  - RunB.sh ### step2: command-line to run VCF2dis
  - RunC.sh ### step2: command-line to run fastreeR
  - aa.r ### Rscript for fastreeR used in RunC.sh
  - md5sum.txt

- 03.Fig1Run
  - RunFig1.sh ### command-line to download the test data files and running VCF2Ds
  - pop.info ### population information for samples used in Fig1
  - sample.group ### population information of the 1000 Genome Project-Phase 3 dataset
  - subsample203.list ### list of samples used in Fig1

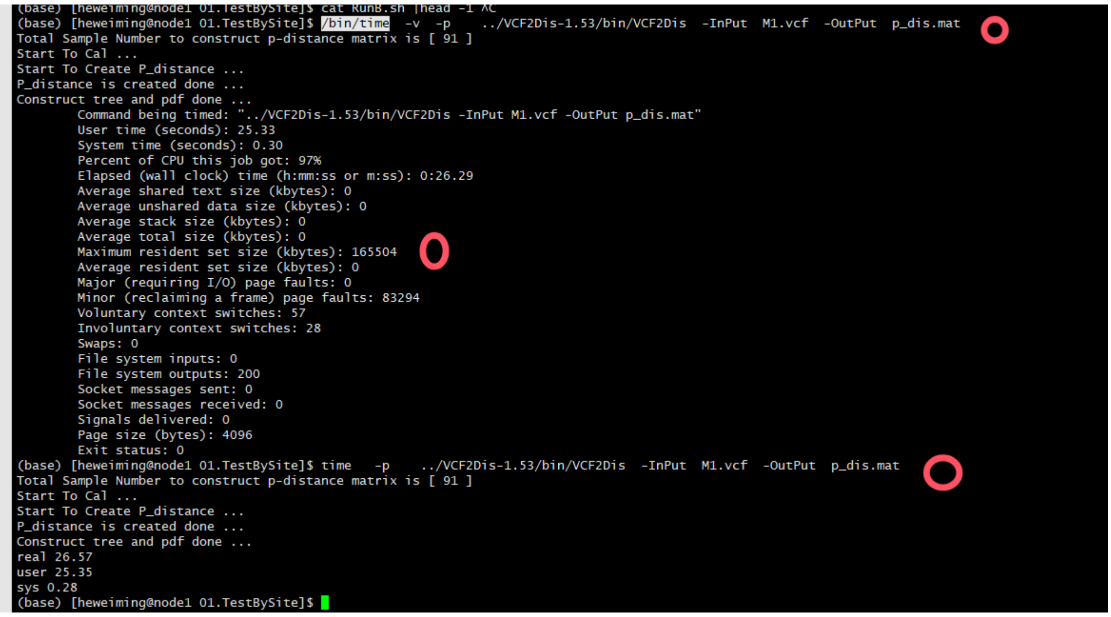
- 04.SupFigRun
  - pop.info ### population information for samples used in Fig1
  - Khuman.vcf.gz ### test dataset form VCF2Dis(VCF2PCACluster) example1
  - aa.r ## Rscript for fastreeR used in RunB.sh
  - RunA.sh ## step1: command-line to run VCF2dis
  - RunB.sh ## step1: command-line to run fastreeR ## NA19006 From JPT to CEU ##

Note:

1. NA19006 (an individual from the JPT population) was grouped with the CEU population, rather than the JPT population, by fastreeR (04.SupFigRun)
2. Our tests were conducted on the CentOS Stream system, which contains two “time” programs located in “/bin” and “/usr/bin”. These programs have different functionalities: the one in “/bin” records detailed information about the process, including memory usage and execution time, while the one in “/usr/bin” only records the running time (as shown in the

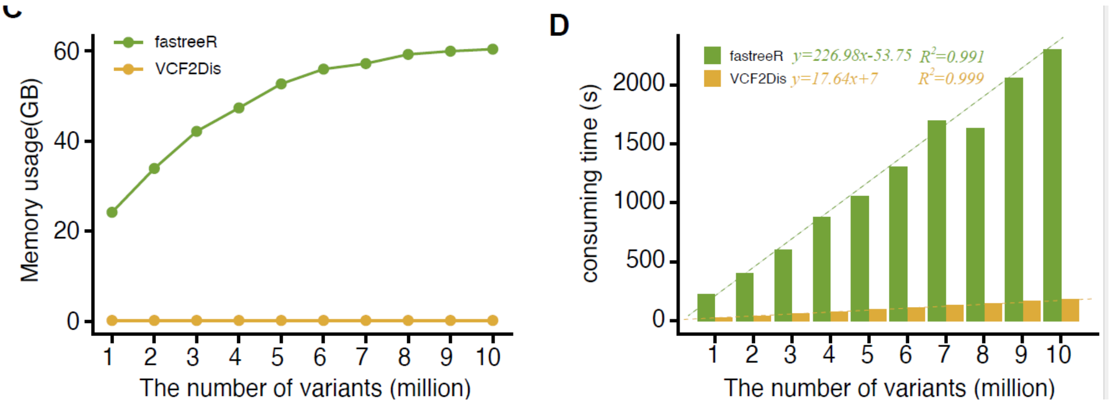
figure below). By default, “time” points to “/usr/bin”. Please check if “/bin/time” exists. If it does not, modify “/bin/time -v -p” to “time -p”.

3. For MacOS (M1 Max), please compile VCF2Dis before running the tests.

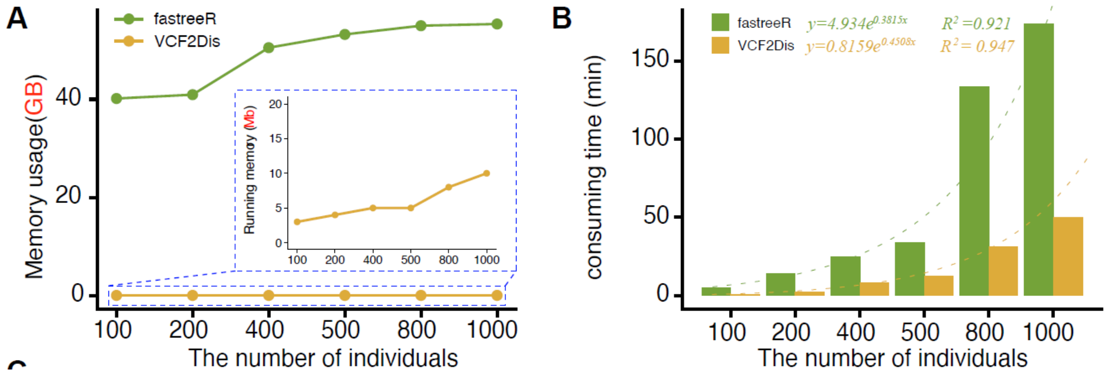


```
(base) [heweiming@node1 01.TestBySite]$ cat RunB.sh |head -1 ^C
(base) [heweiming@node1 01.TestBySite]$ /bin/time  -v  -p    ../VCF2Dis-1.53/bin/VCF2Dis  -InPut  M1.vcf  -OutPut  p_dis.mat
Total Sample Number to construct p-distance matrix is [ 91 ]
Start To Cal ...
Start To Create P_distance ...
P_distance is created done ...
Construct tree and pdf done ...
        Command being timed: "../VCF2Dis-1.53/bin/VCF2Dis -InPut M1.vcf -OutPut p_dis.mat"
        User time (seconds): 25.33
        System time (seconds): 0.30
        Percent of CPU this job got: 97%
        Elapsed (wall clock) time (h:mm:ss or m:ss): 0:26.29
        Average shared text size (kbytes): 0
        Average unshared data size (kbytes): 0
        Average stack size (kbytes): 0
        Average total size (kbytes): 0
        Maximum resident set size (kbytes): 165504
        Average resident set size (kbytes): 0
        Major (requiring I/O) page faults: 0
        Minor (reclaiming a frame) page faults: 83294
        Voluntary context switches: 57
        Involuntary context switches: 28
        Swaps: 0
        File system inputs: 0
        File system outputs: 200
        Socket messages sent: 0
        Socket messages received: 0
        Signals delivered: 0
        Page size (bytes): 4096
        Exit status: 0
(base) [heweiming@node1 01.TestBySite]$ time   -p    ../VCF2Dis-1.53/bin/VCF2Dis  -InPut  M1.vcf  -OutPut  p_dis.mat
Total Sample Number to construct p-distance matrix is [ 91 ]
Start To Cal ...
Start To Create P_distance ...
P_distance is created done ...
Construct tree and pdf done ...
real 26.57
user 25.35
sys 0.28
(base) [heweiming@node1 01.TestBySite]$
```

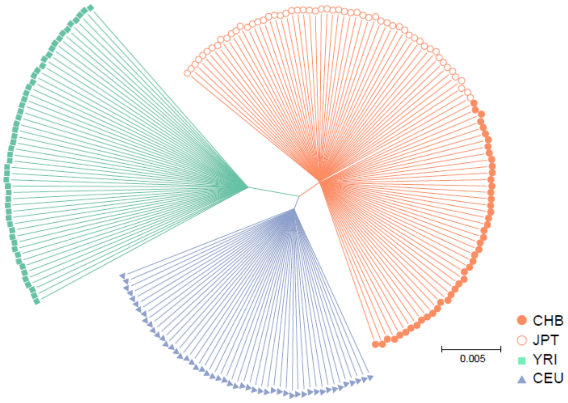
## - 01.TestBySite



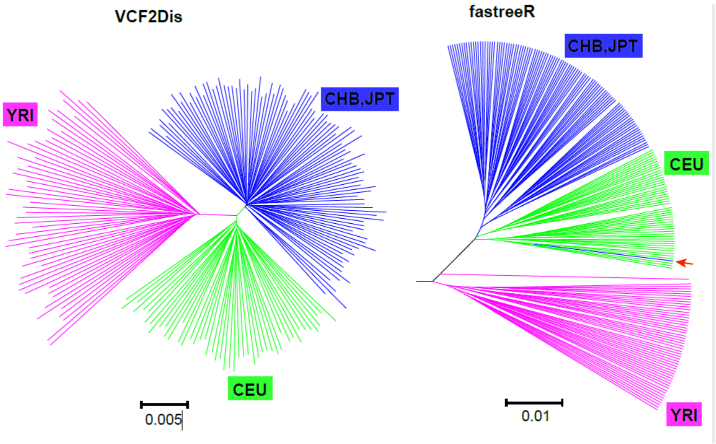
## - 02.TestBySample



## - 03.Fig1Run



## - 04.SupFigRun


VCF2Dis
YRI
CHB,JPT
CEU
fastreeR
CHB,JPT
CEU
YRI