User Manual for

Compressor:

A Compact and Efficient Tool for Genotype Data Compression and Reading

Last updated on January 01, 2017

Preparation

1 Run the self-installing executable file to unpack and install JDK. As part of JDK, this installation may include the Java Runtime Environment.

(http://www.oracle.com/technetwork/java/javase/downloads/index.html).

2 Download and install R on your operating system.

(https://cran.r-project.org/bin/windows/base/).

3 "**Pointer.jar in Compressor Option of website**," "**Demo data**," and "**Compressor application**" can be downloaded from our website (http://www.flybio.net/compressor.html). The .jar and Demo data must be saved in the same directory (Fig.1).

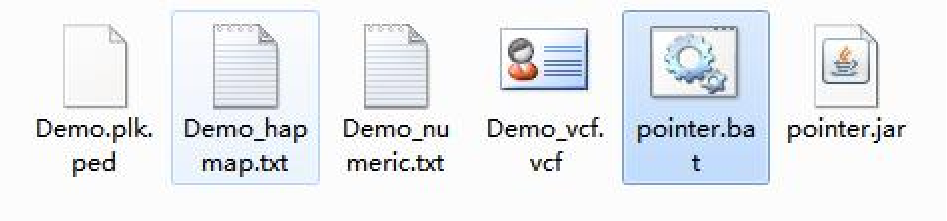


Fig.1 Working folder

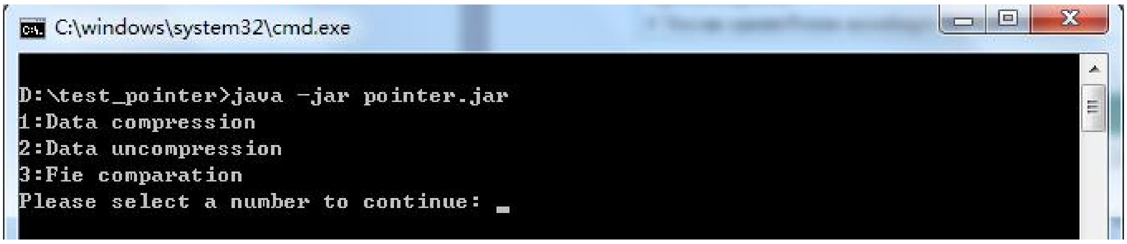
4 Operate Pointer according to the **user's manual**.

I Compression of genotype files

In this section, the genotype files could be compressed using **pointer.jar** in Windows or Linux.

1 Windows

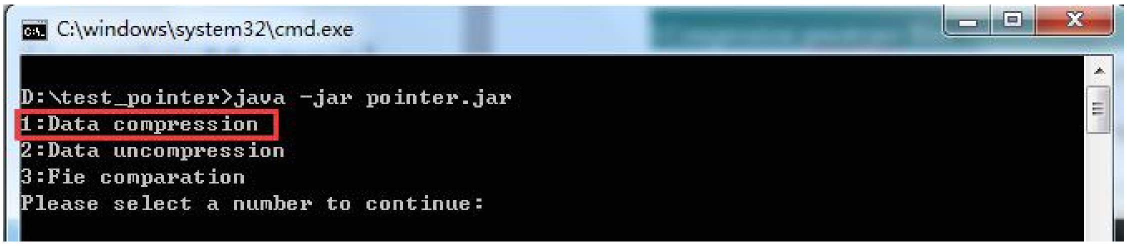
Step 1: Double-click the .dat file; you will then see three options (Fig.2).



```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue: _
```

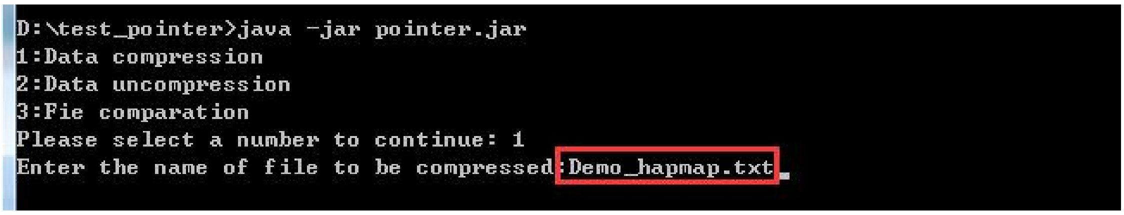
Fig.2

Step 2: Choose "option 1" to compress the data (Fig.3a). Type the name of the file to be compressed (Fig.3b), then you will see the generated file name (Fig.3c).



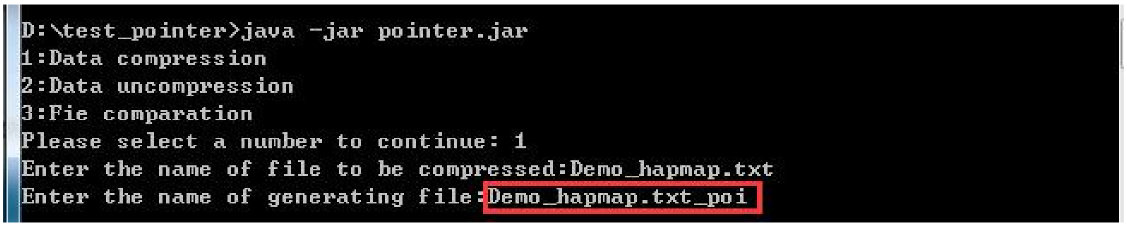
```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue:
```

Fig.3a



```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue: 1
Enter the name of file to be compressed:Demo_hapmap.txt
```

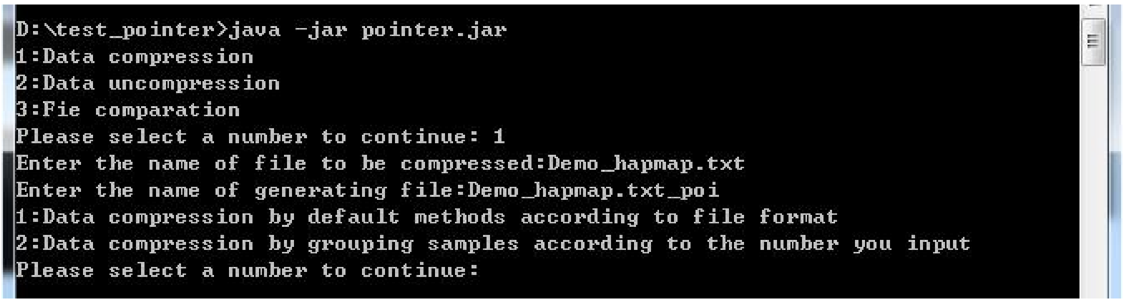
Fig.3b



```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue: 1
Enter the name of file to be compressed:Demo_hapmap.txt
Enter the name of generating file:Demo_hapmap.txt_poi
```

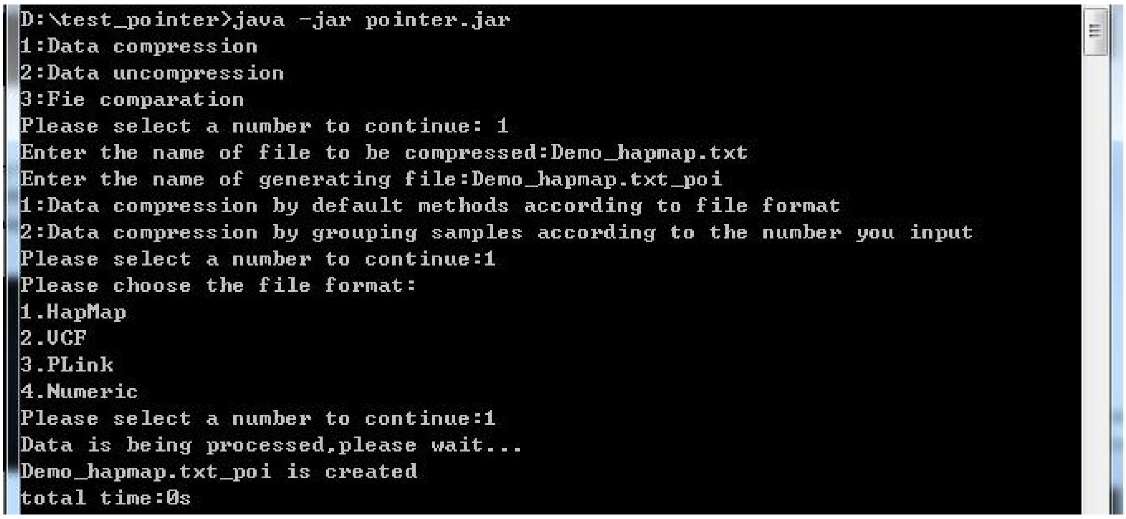
Fig.3c

Step 3: Select the compression mode (Fig.4). "Option 1" represents data compression according to the genotype file format (Fig.5a). And then type a number between 1 and 26. This software could compress data by grouping according to the number you input. (Fig.5b).



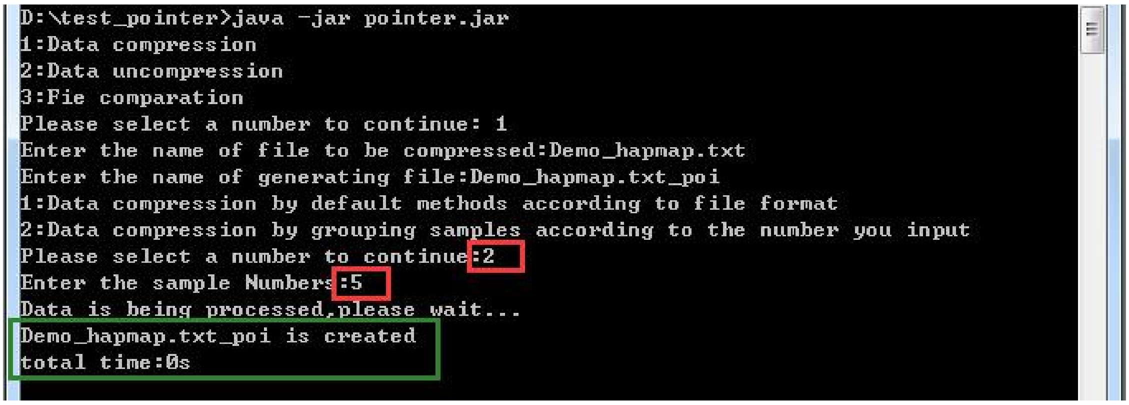
```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue: 1
Enter the name of file to be compressed:Demo_hapmap.txt
Enter the name of generating file:Demo_hapmap.txt_poi
1:Data compression by default methods according to file format
2:Data compression by grouping samples according to the number you input
Please select a number to continue:
```

Fig.4



```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue: 1
Enter the name of file to be compressed:Demo_hapmap.txt
Enter the name of generating file:Demo_hapmap.txt_poi
1:Data compression by default methods according to file format
2:Data compression by grouping samples according to the number you input
Please select a number to continue:1
Please choose the file format:
1.HapMap
2.VCF
3.PLink
4.Numeric
Please select a number to continue:1
Data is being processed,please wait...
Demo_hapmap.txt_poi is created
total time:0s
```

Fig.5a



```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue: 1
Enter the name of file to be compressed:Demo_hapmap.txt
Enter the name of generating file:Demo_hapmap.txt_poi
1:Data compression by default methods according to file format
2:Data compression by grouping samples according to the number you input
Please select a number to continue:2
Enter the sample Numbers:5
Data is being processed,please wait...
Demo_hapmap.txt_poi is created
total time:0s
```

Fig.5b

Note 1: The compression file will be saved in the same directory as the original file (Fig.6).

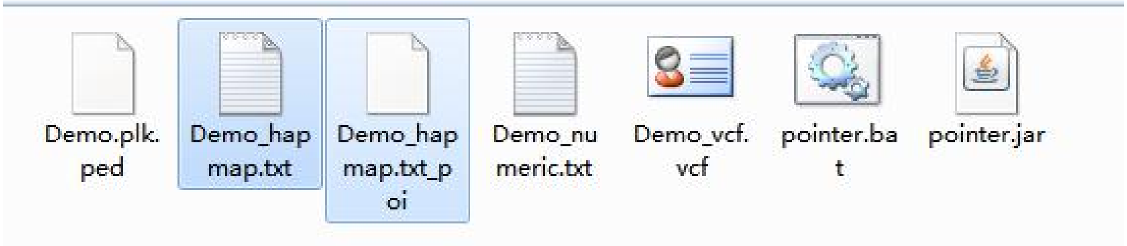
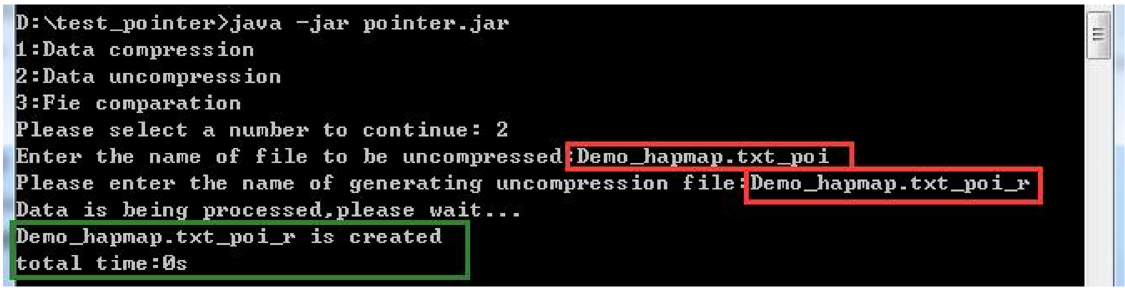


Fig.6

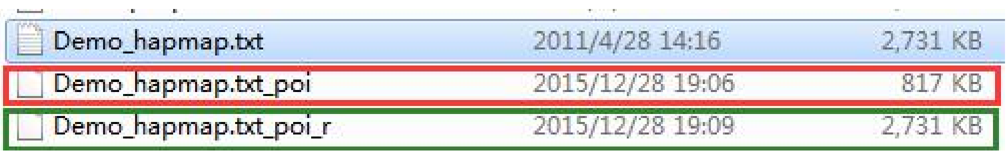
Note 2: The compression methods in options 1 and 2 are both reference-based.

Go back Fig.2. You can choose “option 2” to extract your compressed file (Fig.7). The extracted file is also located in the working directory (Fig.8); Choose “option 3” to compare the original file and decompression one (Fig.9).



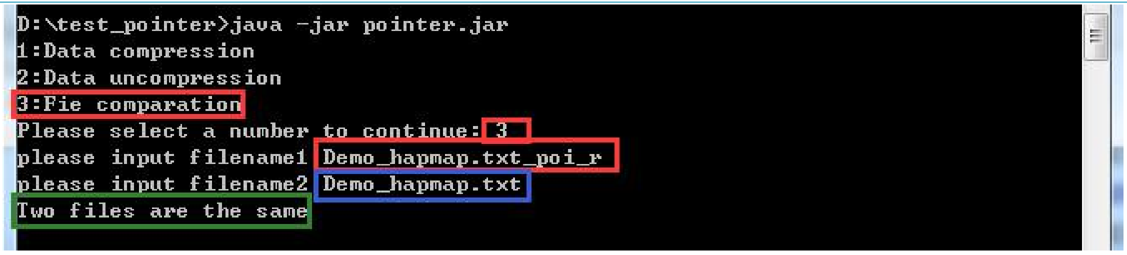
```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue: 2
Enter the name of file to be uncompressed:Demo_hapmap.txt_poi
Please enter the name of generating uncompression file:Demo_hapmap.txt_poi_r
Data is being processed,please wait...
Demo_hapmap.txt_poi_r is created
total time:0s
```

Fig.7



Demo_hapmap.txt	2011/4/28 14:16	2,731 KB
Demo_hapmap.txt_poi	2015/12/28 19:06	817 KB
Demo_hapmap.txt_poi_r	2015/12/28 19:09	2,731 KB

Fig.8

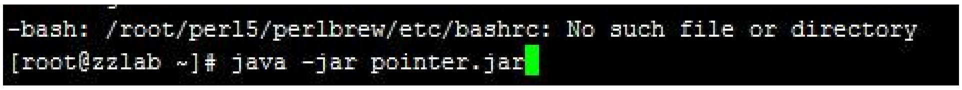


```
D:\test_pointer>java -jar pointer.jar
1:Data compression
2:Data uncompression
3:Fie comparation
Please select a number to continue: 3
please input filename1 Demo_hapmap.txt_poi_r
please input filename2 Demo_hapmap.txt
Two files are same
```

Fig.9

2 Linux

The compression operation in Linux is similar to that in Windows (Fig.10).



```
-bash: /root/perl5/perlbrew/etc/bashrc: No such file or directory
[root@zzlab ~]# java -jar pointer.jar
```

Fig.10

II Read compressed files on Java and R

After genotype file have been compressed, it can be read directly and efficiently.

1 Java

Pointer can be called in Java using special files downloaded from our website. Load Pointer.jar and UnCompressionApp.java into the corresponding folder (Fig.11).

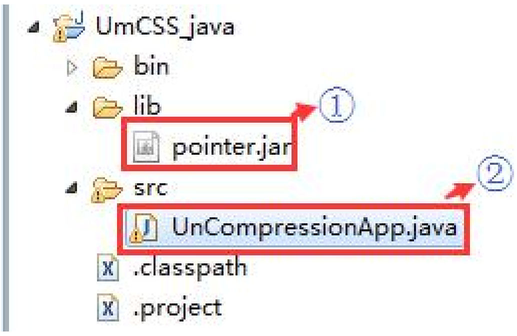
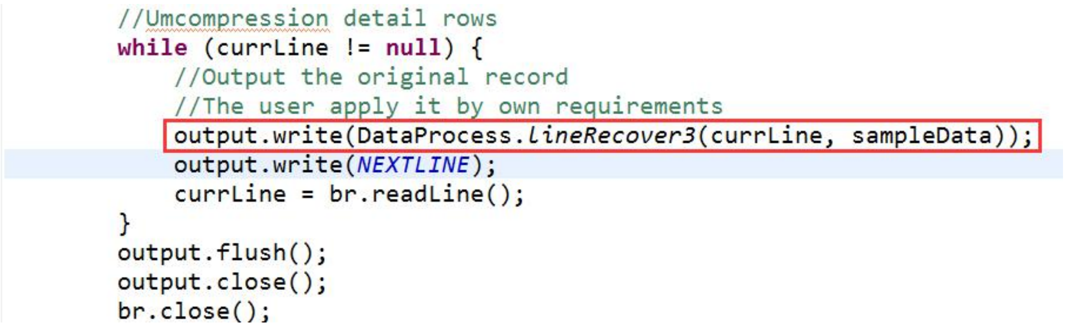


Fig.11

The above compression data can then be read in java class without decompression (Fig.12). Based on this, further program will be run .



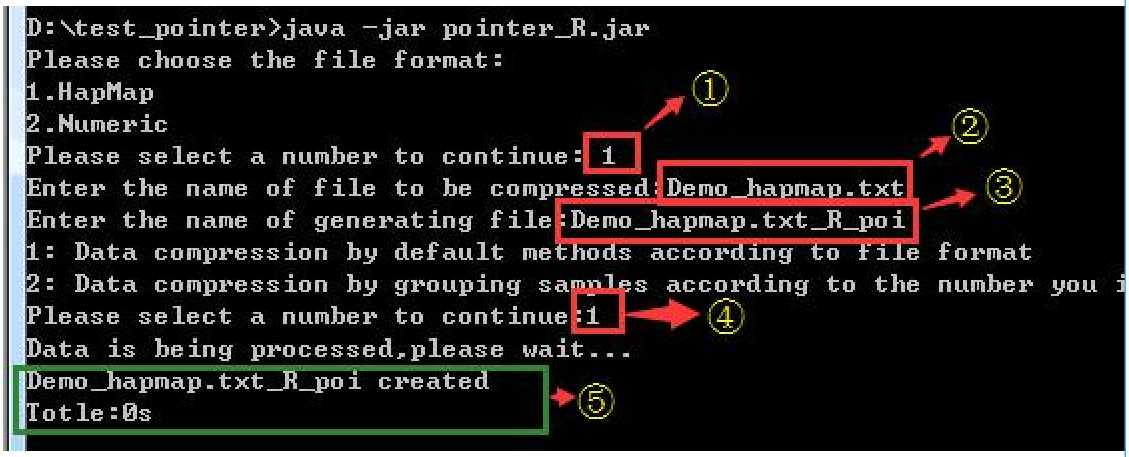
```
//Umcompression detail rows
while (currLine != null) {
    //Output the original record
    //The user apply it by own requirements
    output.write(DataProcess.lineRecover3(currLine, sampleData));
    output.write(NEXTLINE);
    currLine = br.readLine();
}
output.flush();
output.close();
br.close();
```

Fig.12

2 R

After they have been compressed, the genotype files can be read on R directly and efficiently.

Step 1: Generate a compressed file through a special .jar file on R (Fig.13).

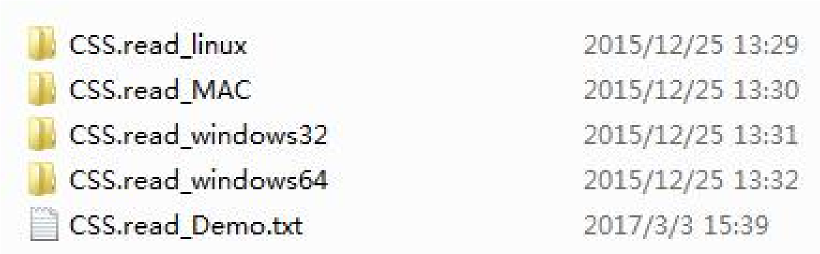


```
D:\test_pointer>java -jar pointer_R.jar
Please choose the file format:
1.HapMap
2.Numeric
Please select a number to continue: 1
Enter the name of file to be compressed:Demo_hapmap.txt
Enter the name of generating file:Demo_hapmap.txt_R_poi
1: Data compression by default methods according to file format
2: Data compression by grouping samples according to the number you i
Please select a number to continue:1
Data is being processed,please wait...
Demo_hapmap.txt_R_poi created
Totle:0s
```

Fig.13

Step 2: Download the read functions of different operation systems from our website (Fig.14) and copy the corresponding Compressor "read function" into the directory of the

compressed files (Fig.15).



CSS.read_linux	2015/12/25 13:29
CSS.read_MAC	2015/12/25 13:30
CSS.read_windows32	2015/12/25 13:31
CSS.read_windows64	2015/12/25 13:32
CSS.read_Demo.txt	2017/3/3 15:39

Fig.14

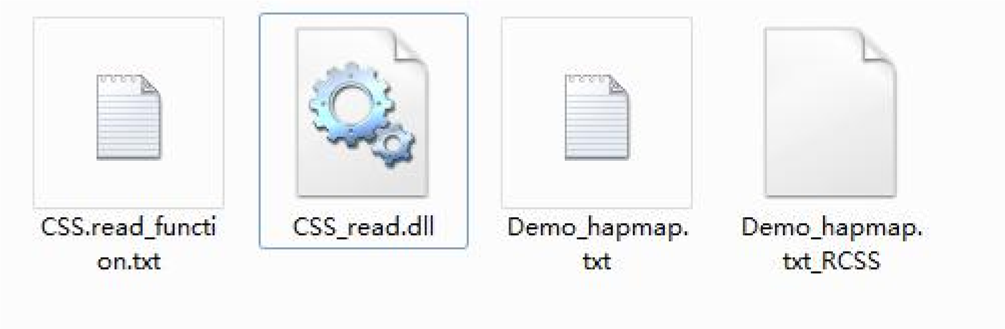
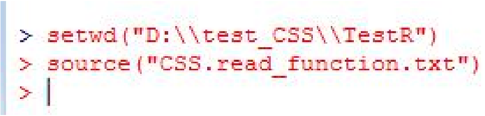


Fig.15

Step 2: Set the working directory and load Compressor read function (Fig.16).

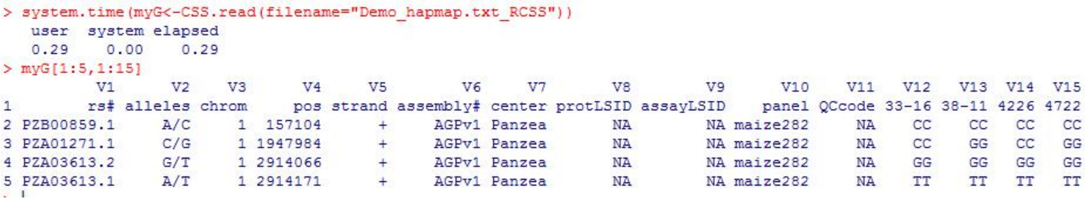


```
> setwd("D:\\test_CSS\\TestR")
> source("CSS.read_function.txt")
> |
```

Fig.16

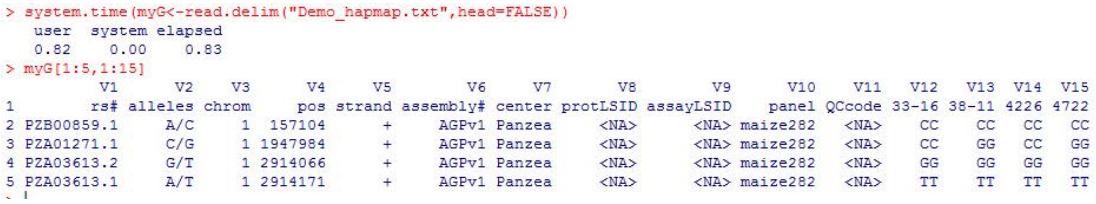
Note: The read functions in Windows 32-bit and Windows 64-bit are not the same.

Step 3: Compare the lengths of reading time with and without Compressor read function (Figs.17a and17b).



```
> system.time(myG<-CSS.read(filename="Demo_hapmap.txt_RCSS"))
   user  system elapsed
   0.29    0.00    0.29
> myG[1:5,1:15]
         V1      V2    V3      V4     V5        V6     V7       V8        V9      V10    V11   V12   V13  V14  V15
1       rs# alleles chrom     pos strand assembly# center protLSID assayLSID    panel QCcode 33-16 38-11 4226 4722
2 PZB00859.1     A/C     1  157104      +     AGPv1 Panzea       NA        NA maize282     NA    CC    CC   CC   CC
3 PZA01271.1     C/G     1 1947984      +     AGPv1 Panzea       NA        NA maize282     NA    CC    GG   CC   GG
4 PZA03613.2     G/T     1 2914066      +     AGPv1 Panzea       NA        NA maize282     NA    GG    GG   GG   GG
5 PZA03613.1     A/T     1 2914171      +     AGPv1 Panzea       NA        NA maize282     NA    TT    TT   TT   TT
```

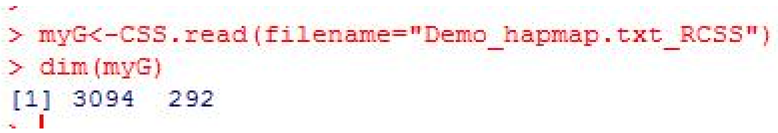
Fig.17a



```
> system.time(myG<-read.delim("Demo_hapmap.txt",head=FALSE))
   user  system elapsed
   0.82    0.00    0.83
> myG[1:5,1:15]
          V1      V2    V3      V4     V5        V6     V7       V8        V9       V10    V11   V12   V13  V14  V15
1        rs# alleles chrom     pos strand assembly# center protLSID assayLSID     panel QCcode 33-16 38-11 4226 4722
2 PZB00859.1     A/C     1  157104      +     AGPv1 Panzea     <NA>      <NA> maize282   <NA>    CC    CC   CC   CC
3 PZA01271.1     C/G     1 1947984      +     AGPv1 Panzea     <NA>      <NA> maize282   <NA>    CC    GG   CC   GG
4 PZA03613.2     G/T     1 2914066      +     AGPv1 Panzea     <NA>      <NA> maize282   <NA>    GG    GG   GG   GG
5 PZA03613.1     A/T     1 2914171      +     AGPv1 Panzea     <NA>      <NA> maize282   <NA>    TT    TT   TT   TT
```

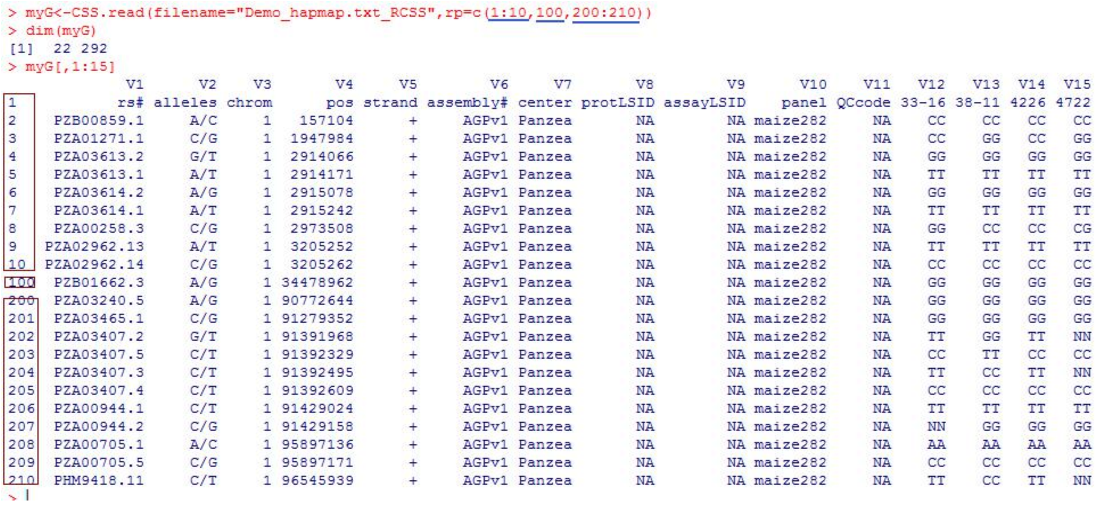
Fig.17b

Step 4: You can gain the information you want from the compressed file directly(Fig.18a-d).



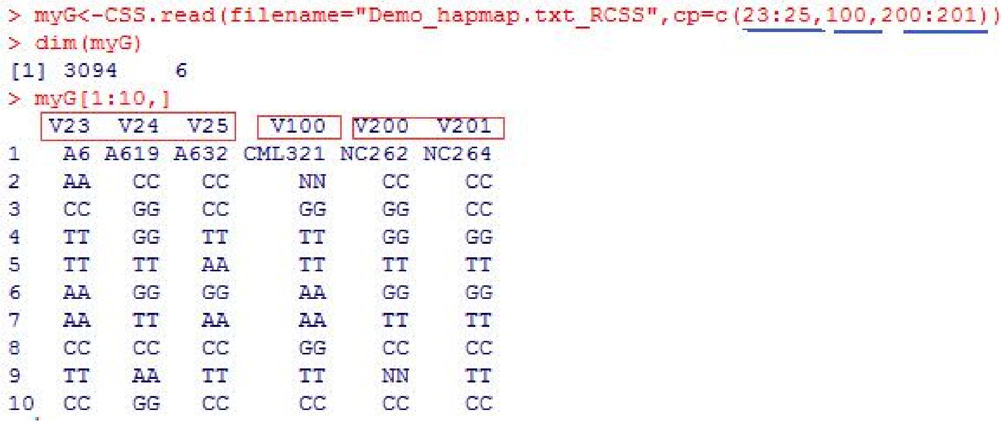
```
> myG<-CSS.read(filename="Demo_hapmap.txt_RCSS")
> dim(myG)
[1] 3094  292
```

Fig.18 a



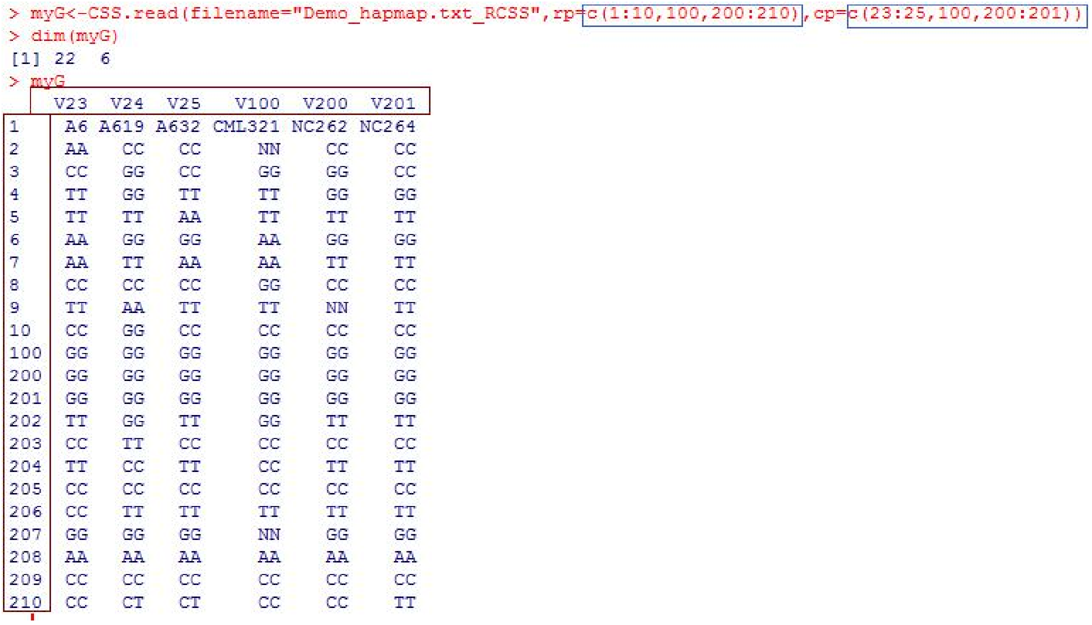
```
> myG<-CSS.read(filename="Demo_hapmap.txt_RCSS",rp=c(1:10,100,200:210))
> dim(myG)
[1]  22 292
> myG[,1:15]
             V1      V2    V3       V4     V5        V6     V7       V8        V9      V10    V11   V12   V13  V14  V15
1           rs# alleles chrom      pos strand assembly# center protLSID assayLSID    panel QCcode 33-16 38-11 4226 4722
2    PZB00859.1     A/C     1   157104      +     AGPv1 Panzea       NA        NA maize282     NA    CC    CC   CC   CC
3    PZA01271.1     C/G     1  1947984      +     AGPv1 Panzea       NA        NA maize282     NA    CC    GG   CC   GG
4    PZA03613.2     G/T     1  2914066      +     AGPv1 Panzea       NA        NA maize282     NA    GG    GG   GG   GG
5    PZA03613.1     A/T     1  2914171      +     AGPv1 Panzea       NA        NA maize282     NA    TT    TT   TT   TT
6    PZA03614.2     A/G     1  2915078      +     AGPv1 Panzea       NA        NA maize282     NA    GG    GG   GG   GG
7    PZA03614.1     A/T     1  2915242      +     AGPv1 Panzea       NA        NA maize282     NA    TT    TT   TT   TT
8    PZA00258.3     C/G     1  2973508      +     AGPv1 Panzea       NA        NA maize282     NA    GG    CC   CC   CG
9   PZA02962.13     A/T     1  3205252      +     AGPv1 Panzea       NA        NA maize282     NA    TT    TT   TT   TT
10  PZA02962.14     C/G     1  3205262      +     AGPv1 Panzea       NA        NA maize282     NA    CC    CC   CC   CC
100  PZB01662.3     A/G     1 34478962      +     AGPv1 Panzea       NA        NA maize282     NA    GG    GG   GG   GG
200  PZA03240.5     A/G     1 90772644      +     AGPv1 Panzea       NA        NA maize282     NA    GG    GG   GG   GG
201  PZA03465.1     C/G     1 91279352      +     AGPv1 Panzea       NA        NA maize282     NA    GG    GG   GG   GG
202  PZA03407.2     G/T     1 91391968      +     AGPv1 Panzea       NA        NA maize282     NA    TT    GG   TT   NN
203  PZA03407.5     C/T     1 91392329      +     AGPv1 Panzea       NA        NA maize282     NA    CC    TT   CC   CC
204  PZA03407.3     C/T     1 91392495      +     AGPv1 Panzea       NA        NA maize282     NA    TT    CC   TT   NN
205  PZA03407.4     C/T     1 91392609      +     AGPv1 Panzea       NA        NA maize282     NA    CC    CC   CC   CC
206  PZA00944.1     C/T     1 91429024      +     AGPv1 Panzea       NA        NA maize282     NA    TT    TT   TT   TT
207  PZA00944.2     C/G     1 91429158      +     AGPv1 Panzea       NA        NA maize282     NA    NN    GG   GG   GG
208  PZA00705.1     A/C     1 95897136      +     AGPv1 Panzea       NA        NA maize282     NA    AA    AA   AA   AA
209  PZA00705.5     C/G     1 95897171      +     AGPv1 Panzea       NA        NA maize282     NA    CC    CC   CC   CC
210  PHM9418.11     C/T     1 96545939      +     AGPv1 Panzea       NA        NA maize282     NA    TT    CC   TT   NN
```

Fig.18 b



```
> myG<-CSS.read(filename="Demo_hapmap.txt_RCSS",cp=c(23:25,100,200:201))
> dim(myG)
[1] 3094    6
> myG[1:10,]
   V23  V24  V25   V100  V200  V201
1   A6 A619 A632 CML321 NC262 NC264
2   AA   CC   CC     NN    CC    CC
3   CC   GG   CC     GG    GG    CC
4   TT   GG   TT     TT    GG    GG
5   TT   TT   AA     TT    TT    TT
6   AA   GG   GG     AA    GG    GG
7   AA   TT   AA     AA    TT    TT
8   CC   CC   CC     GG    CC    CC
9   TT   AA   TT     TT    NN    TT
10  CC   GG   CC     CC    CC    CC
```

Fig.18 c



```
> myG<-CSS.read(filename="Demo_hapmap.txt_RCSS",rp=c(1:10,100,200:210),cp=c(23:25,100,200:201))
> dim(myG)
[1] 22  6
> myG
     V23  V24  V25   V100  V200  V201
1     A6 A619 A632 CML321 NC262 NC264
2     AA   CC   CC     NN    CC    CC
3     CC   GG   CC     GG    GG    CC
4     TT   GG   TT     TT    GG    GG
5     TT   TT   AA     TT    TT    TT
6     AA   GG   GG     AA    GG    GG
7     AA   TT   AA     AA    TT    TT
8     CC   CC   CC     GG    CC    CC
9     TT   AA   TT     TT    NN    TT
10    CC   GG   CC     CC    CC    CC
100   GG   GG   GG     GG    GG    GG
200   GG   GG   GG     GG    GG    GG
201   GG   GG   GG     GG    GG    GG
202   TT   GG   TT     GG    TT    TT
203   CC   TT   CC     CC    CC    CC
204   TT   CC   TT     CC    TT    TT
205   CC   CC   CC     CC    CC    CC
206   CC   TT   TT     TT    TT    TT
207   GG   GG   GG     NN    GG    GG
208   AA   AA   AA     AA    AA    AA
209   CC   CC   CC     CC    CC    CC
210   CC   CT   CT     CC    CC    TT
```

Fig.18 d