MutMap+ pipeline quick start guide

Brief Version 0.9.1

2013/12/11

0. preparation

0.1. install the following programs to your system

1. Perl
2. R
3. BWA
4. SAMtools-0.1.8 or before
5. FASTX-Toolkit (http://hannonlab.cshl.edu/fastx_toolkit/)

0.2. your files

1. public genome sequence (fasta)
   (For rice, Nipponbare reference genome is used.)

2. your sample's fastq files

   - fastq files (archived in gzip) to make your reference
   - your bulked mt-type sample's fastq files (archived in gzip)
   - your bulked WT-type sample's fastq files (archived in gzip)

1.Install pipeline to your system(Linux)

```
$ cd <top_directory_where_you_extract_this_pipeline_archived>

$ tar zxvf MutMapPlus_framework1.2.4.tar.gz       (#) decompression for tar.gz

$ mv MutMapPlus_framework1.2.4 MutMapPlus_test    (#) renamed ; new name is
                                                  "MutMap plus_test" in here
```

2.Provide your sample's fastq files to each directory in this pipeline

```
$ ls /fastq/ZZZ/
ZZZ_L001_R1_001.fastq.gz         (#) e.g., Your fastq files to make your
ZZZ_L001_R2_001.fastq.gz         reference fasta are stored in "/fastq/ZZZ"
ZZZ_L002_R1_001.fastq.gz         directory
ZZZ_L002_R2_001.fastq.gz

$ ls /fastq/XXX/
XXX_L003_R1_001.fastq.gz         (#) e.g., Your bulked mt-type sample's fastq
XXX_L003_R2_001.fastq.gz         files are stored in "/fastq/XXX" directory
XXX_L004_R1_001.fastq.gz
XXX_L004_R2_001.fastq.gz

$ ls /fastq/YYY/
YYY_L005_R1_001.fastq.gz         (#) e.g., Your bulked WT-type sample's fastq
YYY_L005_R2_001.fastq.gz         files are stored in "/fastq/YYY" directory
YYY_L006_R1_001.fastq.gz
YYY_L006_R2_001.fastq.gz


$ cd ~~~/MutMapPlus_test/1.qualify_read/anyname
$ ln -s /fastq/ZZZ/ZZZ_L001_R1_001.fastq.gz p_1_1_sequence.txt.gz
$ ln -s /fastq/ZZZ/ZZZ_L001_R2_001.fastq.gz p_1_2_sequence.txt.gz
$ ln -s /fastq/ZZZ/ZZZ_L002_R1_001.fastq.gz p_2_1_sequence.txt.gz
$ ln -s /fastq/ZZZ/ZZZ_L002_R2_001.fastq.gz p_2_2_sequence.txt.gz

$ cd ~~~/MutMapPlus_test/1.qualify_read/mybulk_mt
$ ln -s /fastq/XXX/XXX_L003_R1_001.fastq.gz p_3_1_sequence.txt.gz
$ ln -s /fastq/XXX/XXX_L003_R2_001.fastq.gz p_3_2_sequence.txt.gz
$ ln -s /fastq/XXX/XXX_L004_R1_001.fastq.gz p_4_1_sequence.txt.gz
$ ln -s /fastq/XXX/XXX_L004_R2_001.fastq.gz p_4_2_sequence.txt.gz

$ cd ~~~/MutMapPlus_test/1.qualify_read/mybulk_WT
$ ln -s /fastq/YYY/YYY_L005_R1_001.fastq.gz p_5_1_sequence.txt.gz
$ ln -s /fastq/YYY/YYY_L005_R2_001.fastq.gz p_5_2_sequence.txt.gz
$ ln -s /fastq/YYY/YYY_L006_R1_001.fastq.gz p_6_1_sequence.txt.gz
$ ln -s /fastq/YYY/YYY_L006_R2_001.fastq.gz p_6_2_sequence.txt.gz

                          (#) "~~~" depends on your system environment
```

3. Provide fasta (downloaded fasta) to “downloaded_fasta” in this pipeline

```
$ cp public.fasta ~~~/MutMapPlus_test/downloaded_fasta/
```

You copy “public.fasta” which is downloaded fasta file
to “downloaded_fasta” directory in this pipeline, as shown in above.

4.Setup “common.fnc”

4.1. edit “config.txt” in top directory of this pipeline

before line 12-15, 28

```
...
Key1_Bulked_sample_name="mybulk"
Key1_Bulked_sample_Type_mt="mt"
Key1_Bulked_sample_Type_WT="WT"
Key1_Reference_sample="anyname"
...
...
Key3_Path_public_reference_FASTA="./downloaded_fasta/IRGSP-
1.0_genome.fasta/IRGSP-1.0_genome.fasta"
```

after

```
...
Key1_Bulked_sample_name="Test"
Key1_Bulked_sample_Type_mt="XXX"
Key1_Bulked_sample_Type_WT="YYY"
Key1_Reference_sample="ZZZ"
...
...
Key3_Path_public_reference_FASTA="./downloaded_fasta/public.fasta"
```

You keep other Key paramers as default value(no change) usually

4.2. run “Bat_make_common.fnc.sh”

```
$ cd ~~~/MutMapPlus_test

$ ./Bat_make_common.fnc.sh

mv 1.qualify_read/anyname 1.qualify_read/ZZZ
mv 1.qualify_read/mybulk_mt 1.qualify_read/test_XXX
mv 1.qualify_read/mybulk_WT 1.qualify_read/test_YYY
```

5.1. [Step 1] qualify fastq files

5.1.1. run “Run_all_Bats.sh” with an argument [9|0|1]

```
$ cd ~~~/MutMapPlus_test/1.qualify_read

$ ./Run_all_Bats.sh 9
```

after that, you get qualified fastq files from your “ZZZ” fastq files to make “your reference fasta”

```
$ cd ~~~/MutMapPlus_test/1.qualify_read

$ ./Run_all_Bats.sh 0
```

after that, you get qualified fastq files from your bulked mt-type sample “XXX” fastq files

```
$ cd ~~~/MutMapPlus_test/1.qualify_read

$ ./Run_all_Bats.sh 1
```

after that, you get qualified fastq files from your bulked WT-type sample “YYY” fastq files

5.2. [Step2] make your reference fastq

5.2.1. run “Run_all_Bats.sh”

```
$ cd ~~~/MutMapPlus_test/2.make_consensus/

$ ./Run_all_Bats.sh
```

after that, you get “your reference fasta”.

5.2.2. run “Run_all_Bats.sh”

```
$ cd ~~~/MutMapPlus_test/2.make_consensus/90.align_to_this_fasta/

$ ./Run_all_Bats.sh
```

after that, you get false positive SNPs which may disturb your analysis.
in next steps, those SNPs will be rejected.

5.3. [Step3] align your bulked sample's fastq to "your reference fasta"
5.3.1. run "Run_all_Bats.sh" with an argument [0|1]

```
$ cd ~~~/MutMapPlus_test/3.alignment/

$ ./Run_all_Bats.sh 0
```

after that, you get SNPs generated from your bulked mt-type sample, via bwa alignment and coval filtering,

```
$ cd ~~~/MutMapPlus_test/3.alignment/

$ ./Run_all_Bats.sh 1
```

after that, you get SNPs generated from your bulked WT-type sample, via bwa alignment and coval filtering.

5.4. [Step4] analysis (I) pairing SNPs
5.4.1. run "Run_all_Bats.sh"

```
$ cd ~~~/MutMapPlus_test/4.search_for_partner/

$ ./Run_all_Bats.sh
```

after that, you get pairwise SNPs between your bulked mt-type sample and your bulked WT-type sample

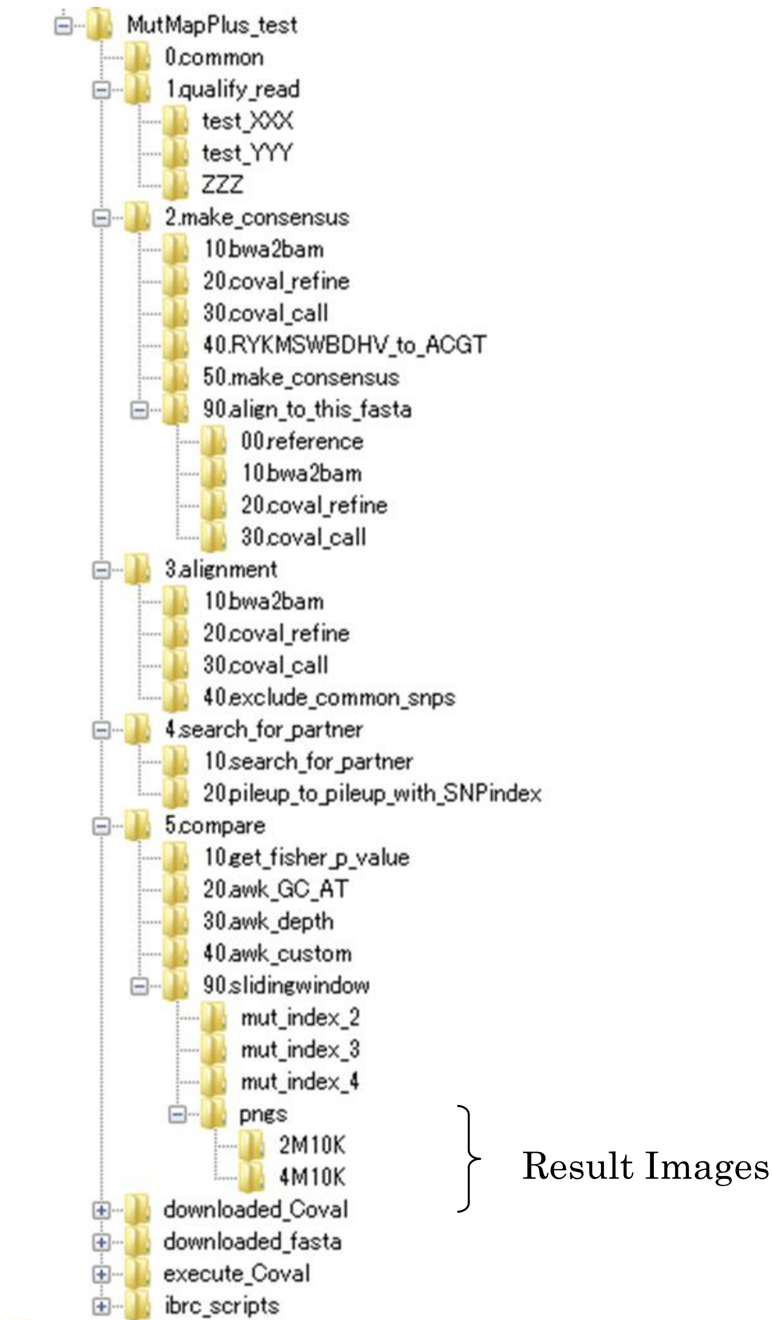
5.5. [Step5] analysis (II) compare SNPs between mt-type's and WT-type's
5.5.1. run "Run_all_Bats.sh"

```
$ cd ~~~/MutMapPlus_test/5.compare/

$ ./Run_all_Bats.sh
```

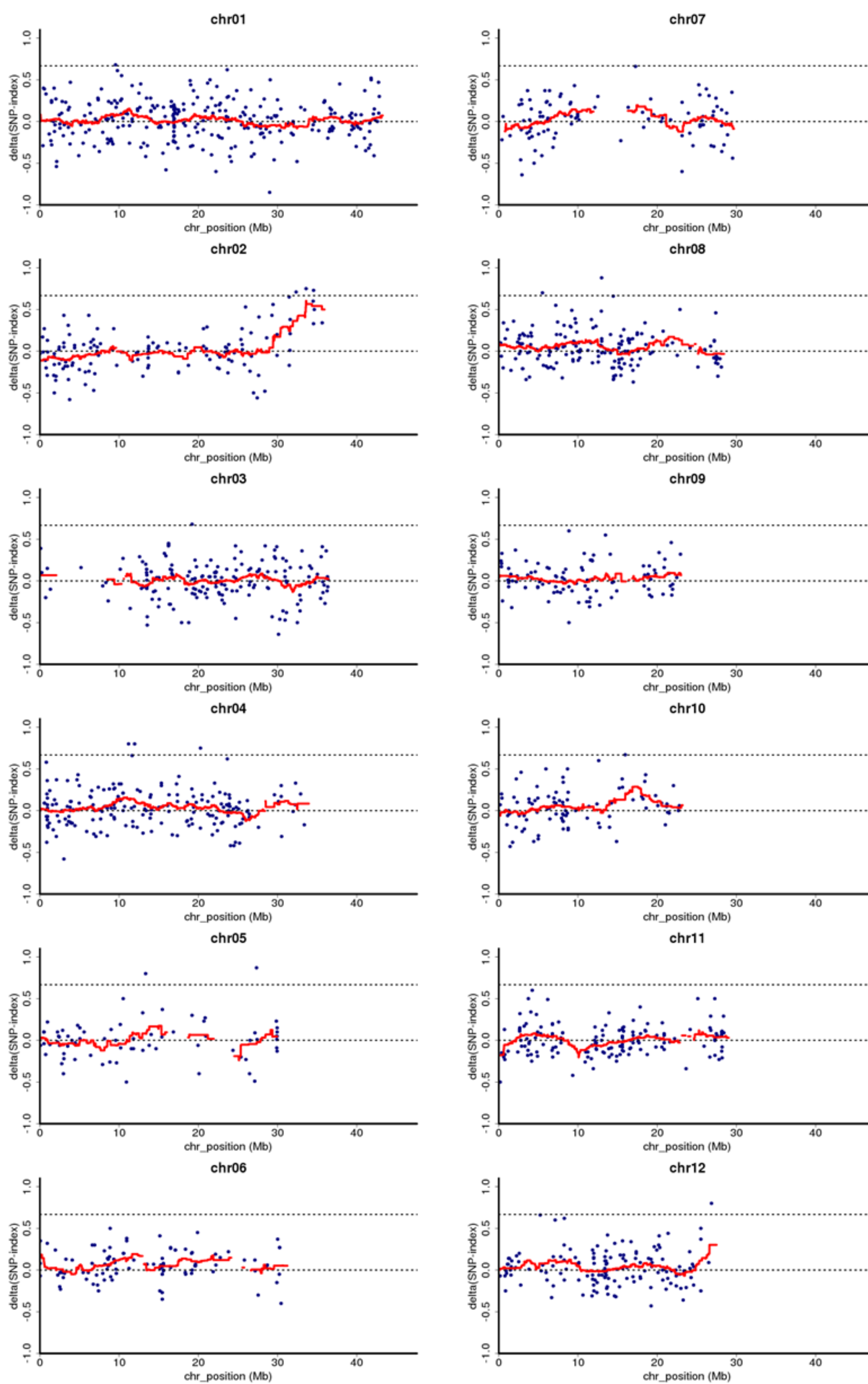
after that, you get filtered pairwise SNPs.
finally you get result as below images(png)

6. Directories



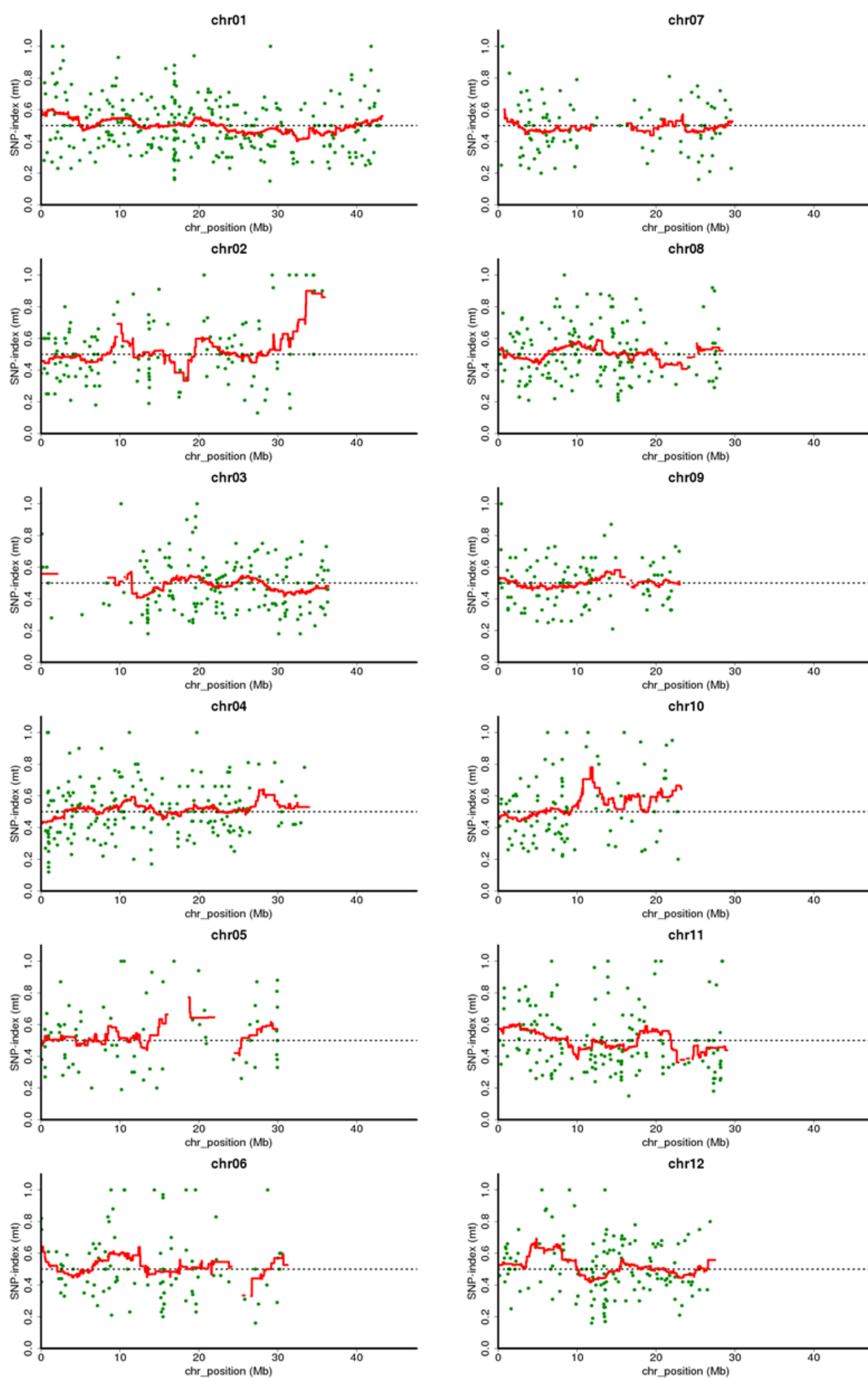
7. Result Images

7.1. delta (SNP-index), e.g., “test_XXX_YYY_sldwnd2M10K_cov2_co5.png”



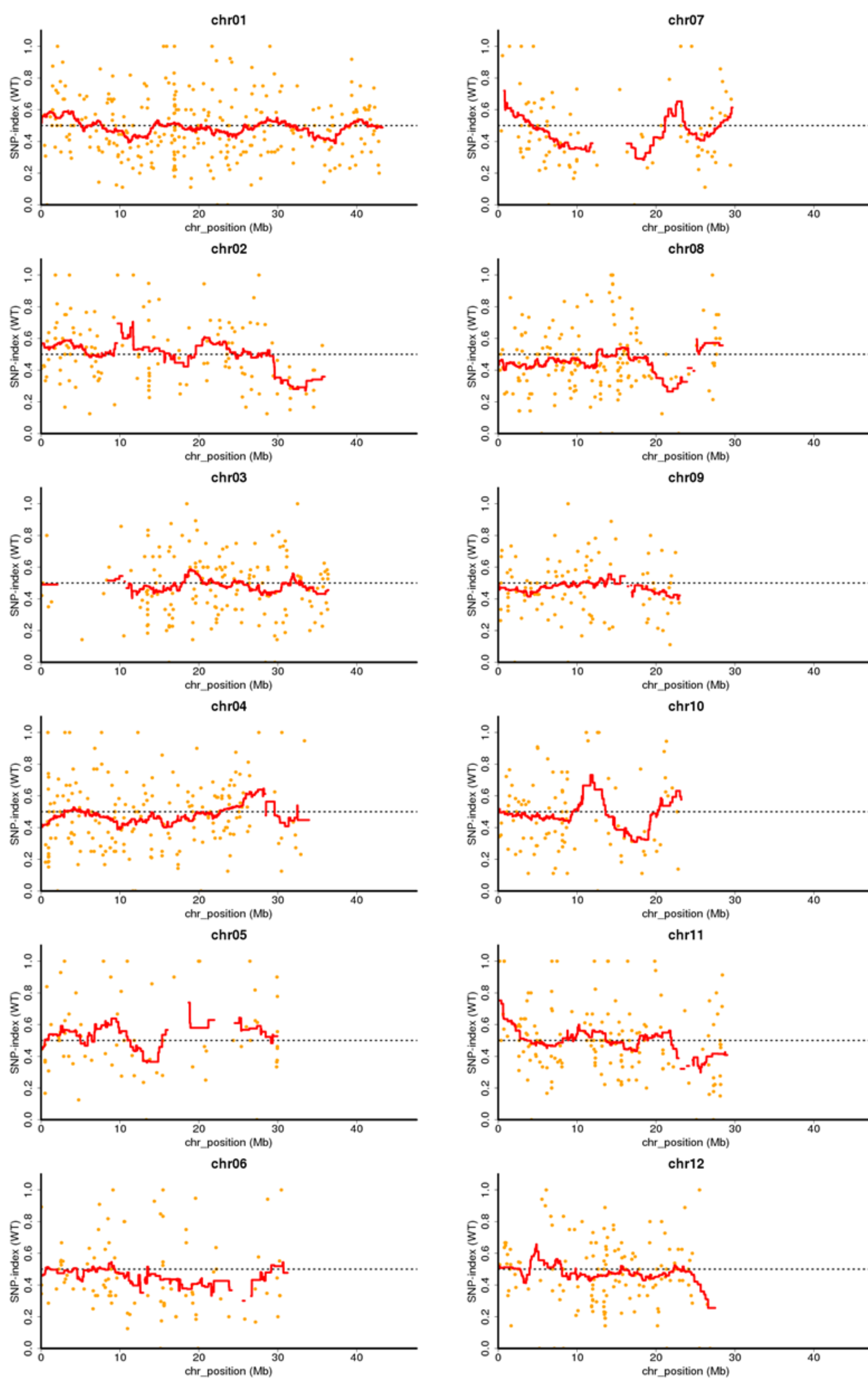
Blue dot : delta(SNP-index)
Red line : sliding window average for delta(SNP-index)

7.2. bulked mt-type's SNP-index, e.g., "test_XXX_sldwnd2M10K_cov2_co5.png"



Green dot : SNP-index of your bulked mt-type samples
Red line : sliding window average for SNP-index

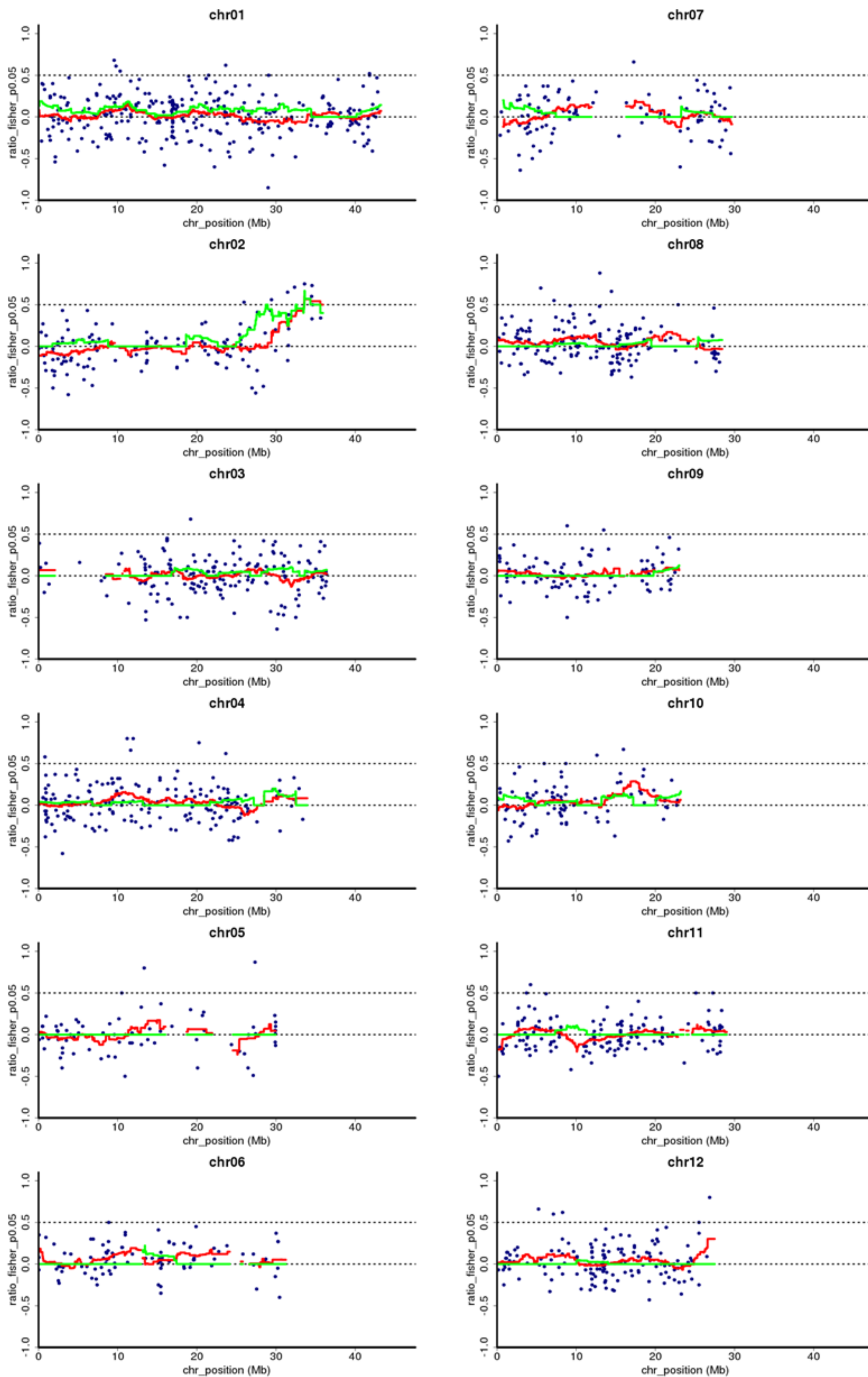
7.3. bulked WT-type's SNP-index, e.g., "test_YYY_sldwnd2M10K_cov2_co5.png"



Orange dot : SNP-index of your bulked WT-type samples
Red line : sliding window average for SNP-index

7.4. delta (SNP-index) with additional curve,

e.g., “ratio_test_XXX_YYY_sldwnd2M10K_cov2_co5.png”



Blue dot : delta(SNP-index)

Red line : sliding window average for delta(SNP-index)

Green line : sliding window average for the ratio of counts of SNP which gives fisher’s p-value < 0.05 in a window