DISCOSNP++:

Colib'read

Live demo

Download and install

Download

- web page: github.com/GATB/DiscoSnp
- Chose latest release (2.2.10 today)

Latest release

v2.2.10

f12f130

v2.2.10

genscale-admin released this 14 days ago

new release

Downloads

DiscoSNP.-v2.2.10-bin-Darwin.tar.gz	11.3 MB
DiscoSNP.-v2.2.10-bin-Linux.tar.gz	14.5 MB
DiscoSNP.-v2.2.10-Source.tar.gz	13.9 MB
Source code (zip)	
Source code (tar.gz)	

Install

- Uncompress the downloaded file:

```
tar –xvzf DiscoSNP.-v2.2.10-Source.tar.gz
cd DiscoSNP++-v2.2.10-Source/
```

- Compile all tools:

```
sh INSTALL
…
*** Test: OK
```

First tests

Tests from the README – Test1

- README File indicates:
  - `./run_discoSnp++.sh -r test/fof.txt –T`
- Let's look what's in test/fof.txt:
  - `more test/fof.txt`

```
reads_sequence1.fasta.gz
reads_sequence2.fasta.gz
```

- `./run_discoSnp++.sh -r test/fof.txt –T`

```
…
fasta of predicted variant is
"discoRes_k_31_c_auto_D_100_P_1_b_0_coherent.fa"
Ghost VCF file (1-based) is
"discoRes_k_31_c_auto_D_100_P_1_b_0_coherent.vcf"
```

Tests from the README – Test1

- (head of the) generated fasta file:

```
>SNP_higher_path_3|P_1:30_C/G|high|nb_pol_1|
left_unitig_length_86|right_unitig_length_261|
left_contig_length_166|right_contig_length_761|C1_124|
C2_0|Q1_0|Q2_0|G1_0/0:10,378,2484|G2_1/1:2684,408,10|
rank_1
cggaattgctatagcccttgaacgctacatgcacgataccaagttatgtatggaccgggtcatcaataggttatagccttgtagttaacatgtagcccggccctattagtacagtag
tgccttcatcggcattctgtttattaagtttttctacagcaaaacgatCAAGTGCACTTCCACAGAGCGCGGTAGAGACTCATCCACCCGGCAGCTCTGTAATAGGGACtaaaaaa
gtgatgataatcatgagtgccgcgttatggtggtgtcggatcagagcggtcttacgaccagtcgtatgccttctcgagttccgtccggttaagcgtgacagtcccagtgaacccaca
aaccgtgatggctgtccttggagtcatacgcaagaaggatggtctccagacaccggcgcaccagttttcacgccgaaagcataaacgacgagcacatatgagagtgttagaactgga
cgtgcggtttctctgcgaagaacacctcgagctgttgcgttgttgcgctgcctagatgcagtgtcgcacatatcacttttgcttcaacgactgccgctttcgctgtatccctagaca
gtcaacagtaagcgctttttgtaggcaggggctccccctgtgactaactgcgccaaaacatcttcggatccccttgtccaatctaactcaccgaattcttacattttagaccctaat
atcacatcattagagattaattgccactgccaaaattctgtccacaagcgttttagttcgccccagtaaagttgtctataacgactaccaaatccgcatgttacgggacttcttatt
aattctttttcgtgaggagcagcggatcttaatggatggccgcaggtggtatggaagctaatagcgcgggtgagagggtaatcagccgtgtccaccaacacaacgctatcgggcga
ttctataagattccgcattgcgtctacttataagatgtctcaacggtatccg
>SNP_lower_path_3|P_1:30_C/G|high|nb_pol_1|
left_unitig_length_86|right_unitig_length_261|
left_contig_length_166|right_contig_length_761|C1_0|
C2_134|Q1_0|Q2_0|G1_0/0:10,378,2484|G2_1/1:2684,408,10|
rank_1
cggaattgctatagcccttgaacgctacatgcacgataccaagttatgtatggaccgggtcatcaataggttatagccttgtagttaacatgtagcccggccctattagtacagtag
tgccttcatcggcattctgtttattaagtttttctacagcaaaacgatCAAGTGCACTTCCACAGAGCGCGGTAGAGAGTCATCCACCCGGCAGCTCTGTAATAGGGACtaaaaaa
gtgatgataatcatgagtgccgcgttatggtggtgtcggatcagagcggtcttacgaccagtcgtatgccttctcgagttccgtccggttaagcgtgacagtcccagtgaacccaca
aaccgtgatggctgtccttggagtcatacgcaagaaggatggtctccagacaccggcgcaccagttttcacgccgaaagcataaacgacgagcacatatgagagtgttagaactgga
cgtgcggtttctctgcgaagaacacctcgagctgttgcgttgttgcgctgcctagatgcagtgtcgcacatatcacttttgcttcaacgactgccgctttcgctgtatccctagaca
gtcaacagtaagcgctttttgtaggcaggggctccccctgtgactaactgcgccaaaacatcttcggatccccttgtccaatctaactcaccgaattcttacattttagaccctaat
atcacatcattagagattaattgccactgccaaaattctgtccacaagcgttttagttcgccccagtaaagttgtctataacgactaccaaatccgcatgttacgggacttcttatt
aattctttttcgtgaggagcagcggatcttaatggatggccgcaggtggtatggaagctaatagcgcgggtgagagggtaatcagccgtgtccaccaacacaacgctatcgggcga
ttctataagattccgcattgcgtctacttataagatgtctcaacggtatccg
```

Tests from the README – Test1

- (head of the) generated fasta file:

```
>SNP_higher_path_3|P_1:30_C/G|high|nb_pol_1|
left_unitig_length_86|right_unitig_length_261|
left_contig_length_166|right_contig_length_124|
C2_0|Q1_0|Q2_0|G1_0/0:10,378,24[illegible]10|
rank_1
cggaattgctatagcccttgaacgctacatgc[illegible]ccctattagtacagtag
tgccttcatcggcattctgtttatt[illegible]TAATAGGGACtaaaaaa
gtgatgataatcatgag[illegible]tcccagtgaacccaca
aaccgtgatg[illegible]gagtgttagaactgga
cgtgcgg[illegible]ctgtatccctagaca
gtcaaca[illegible]attttagaccctaat
atcacatc[illegible]acgggacttcttatt
aattcttt[illegible]acgctatcgggcga
ttctataa[illegible]
>SNP_[illegible]
left_[illegible]ength_261|
left_[illegible]contig_length_761|C1_0|
C2_134[illegible]0:10,378,2484|G2_1/1:2684,408,10|
rank_1
cggaattgctata[illegible]gaacgctacatgcacgataccaagttatgtatggaccgggtcatcaataggttatagccttgtagttaacatgtagcccggccctattagtacagtag
tgccttcatcggcattctgtttattaagtttttctacagcaaaacgatCAAGTGCACTTCCACAGAGCGCGGTAGAGAGTCATCCACCCGGCAGCTCTGTAATAGGGACtaaaaaa
gtgatgataatcatgagtgccgcgttatggtggtgtcggatcagagcggtcttacgaccagtcgtatgccttctcgagttccgtccggttaagcgtgacagtcccagtgaacccaca
aaccgtgatggctgtccttggagtcatacgcaagaaggatggtctccagacaccggcgcaccagttttcacgccgaaagcataaacgacgagcacatatgagagtgttagaactgga
cgtgcggtttctctgcgaagaacacctcgagctgttgcgttgttgcgctgcctagatgcagtgtcgcacatatcacttttgcttcaacgactgccgctttcgctgtatccctagaca
gtcaacagtaagcgctttttgtaggcaggggctccccctgtgactaactgcgccaaaacatcttcggatccccttgtccaatctaactcaccgaattcttacattttagaccctaat
atcacatcattagagattaattgccactgccaaaattctgtccacaagcgttttagttcgccccagtaaagttgtctataacgactaccaaatccgcatgttacgggacttcttatt
aattctttttcgtgaggagcagcggatcttaatggatggccgcaggtggtatggaagctaatagcgcgggtgagagggtaatcagccgtgtccaccaacacaacgctatcgggcga
ttctataagattccgcattgcgtctacttataagatgtctcaacggtatccg
```

C1, C2, …, see …read_files_correspondance.txt :
C_1 test/reads_sequence1.fasta.gz
C_2 test/reads_sequence2.fasta.gz

Tests from the README – Test1

- (head of the) generated VCF file:

```
#CHROM   POS      ID       REF      ALT      QUAL     FILTER
INFO     FORMAT   G1       G2

SNP_higher_path_3         196      3        C        G
.        .
Ty=SNP;Rk=1;UL=86;UR=261;CL=166;CR=761;Genome=.;Sd=.
GT:DP:PL:AD:HQ
0/0:124:10,378,2484:124,0:0,0
1/1:134:2684,408,10:0,134:0,0
```

Tests from the README – Test2

- ./run_discoSnp++.sh -r test/fof.txt –T –G test/reference_genome.fa

  …

fasta of predicted variant

"discoRes_k_

VCF

"disc

An IG

based

"discoR

_D_100_P_1_b_0_coherent_for_IGV.vcf"

Must have BWA installed

Tests from the README – Test2

- `./run_discoSnp++.sh -r test/fof.txt –T –G test/reference_genome.fa`

  …

fasta of predicted variant is "discoRes_k_31_c_auto_D_100_P_1_b_0_coherent.fa"

VCF file (1-based) is "discoRes_k_31_c_auto_D_100_P_1_b_0_coherent.vcf"

An IGV ready VCF file (sorted by position, only mapped variants, 0-based) is "discoRes_k_31_c_auto_D_100_P_1_b_0_coherent_for_IGV.vcf"

Tests from the README – Test2

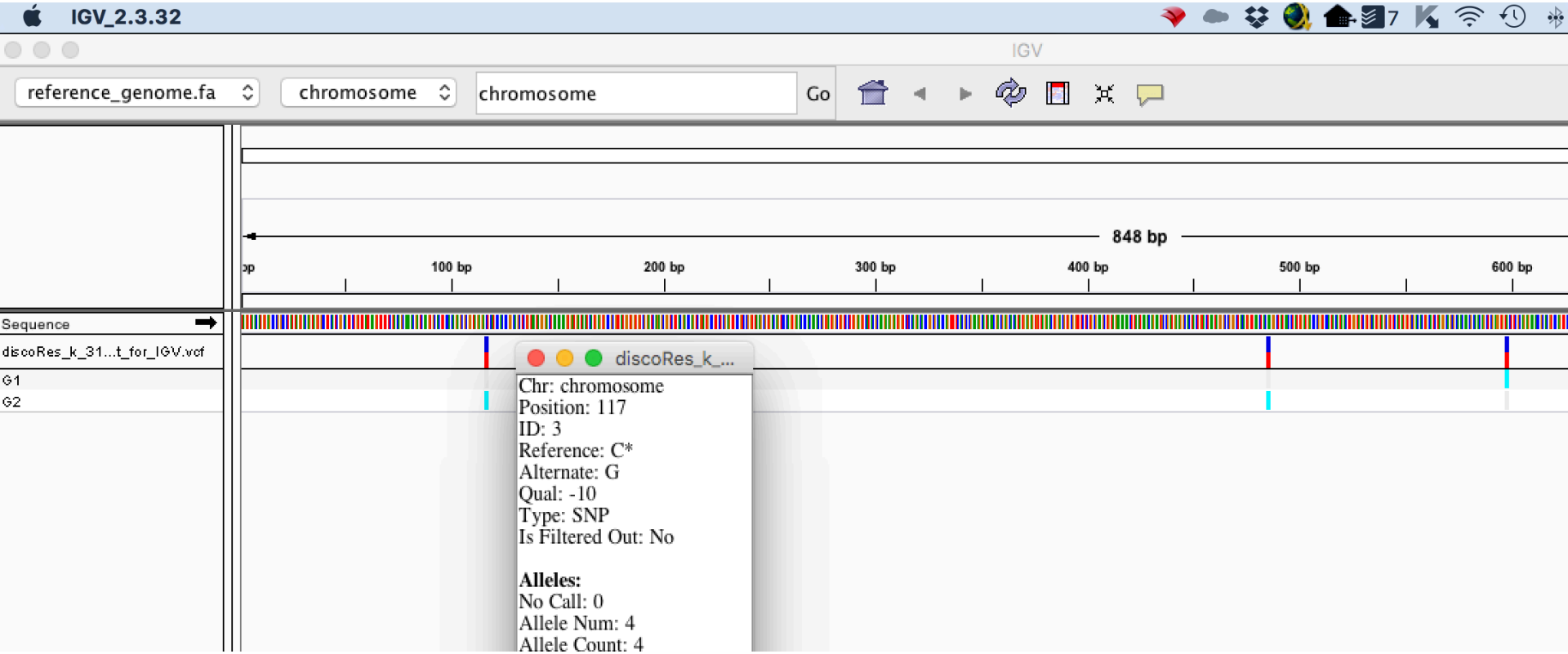
- (head of the) generated VCF file:

```
#CHROM   POS      ID       REF      ALT      QUAL     FILTER
INFO     FORMAT   G1       G2

chromosome        117      3        C        G        .
PASS
Ty=SNP;Rk=1;UL=86;UR=261;CL=166;CR=761;Genome=C;Sd=1
GT:DP:PL:AD:HQ   0/0:124:10,378,2484:124,0:0,0
1/1:134:2684,408,10:0,134:0,0
```

Tests from the README – Test2

- Two VCF files
  - …coherent.vcf = all predicted variants, sorted by rank
  - …coherent_for_IGV.vcf = mapped variants, sorted by mapping pos.



Main options

./run_discoSnp++.sh –h (overview)

bubble predictions

-r <string> read_file_of_files **(mandatory)**

-k <int> kmer size

-c <list of int> solidity threshold(s)

-b <int> branching strategy (0,1 or 2)

-D <int> max size of searched indels

-P <int> max nb close SNPs in a bubble

-l remove low complexity bubbles

-t or –T extend bubbles to unitigs or contigs

-g reuse a previously created graph

VCF creation

-G <string> file of reference genome

-R add the reference genome to the graph

Many other ‘cosmetic’ options

File of files (fof) – read set ≠ data set

Each line of the called fof is a read set

- Example of two read sets :

fof.txt:

```
dataset_reads1.fa
dataset_reads2.fa
```

- Example of one read set composed of two data sets:

fof.txt:

```
fof_two_sets.txt
```

fof_two_sets.txt:

```
dataset_reads1.fa
dataset_reads2.fa
```

- Example of two read sets each composed of two datasets

fof.txt:

```
fof_two_sets_1.txt
fof_two_sets_2.txt
```

fof_two_sets_1.txt:

```
dataset_reads1_1.fa
dataset_reads1_2.fa
```

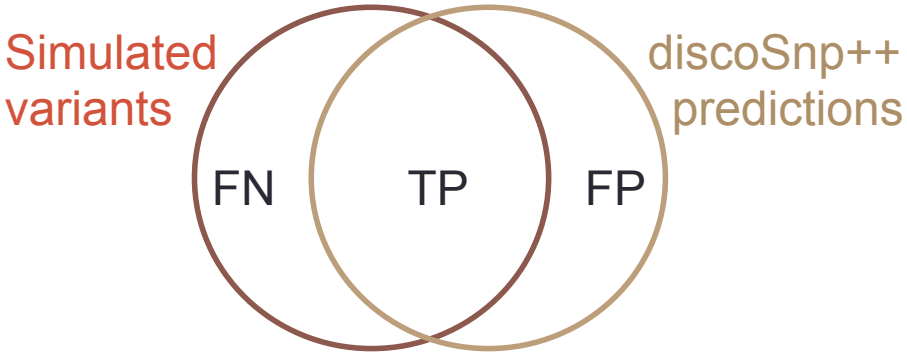
fof_two_sets_2.txt:

```
dataset_reads2_1.fa
dataset_reads2_2.fa
```

Larger test

Data & validator

- www.irisa.fr/symbiose/people/ppeterlongo/demo_disco/validator_5M.zip
  - Data:
    - First 5M nucleotides from human chromosome1
    - Simulated individuals
      - Uses 1000 genome project variant predictions
    - Simulated reads
      - 40x – 100 bp reads – 0.1% error rate
  - Validator compares predictions to **known simulated variants**:



First test

- Create the file of file:
  - `ls humch1_0*> fof`
- Run disco:
  - `../run_discoSnp++.sh -r fof -G humch1_first_5M.fasta`
- Validate results:
  - `./validator_vcf.sh discoRes_k_31_c_auto_D_100_P_1_b_0_coherent_for_IGV.vcf`

First test -- Validation

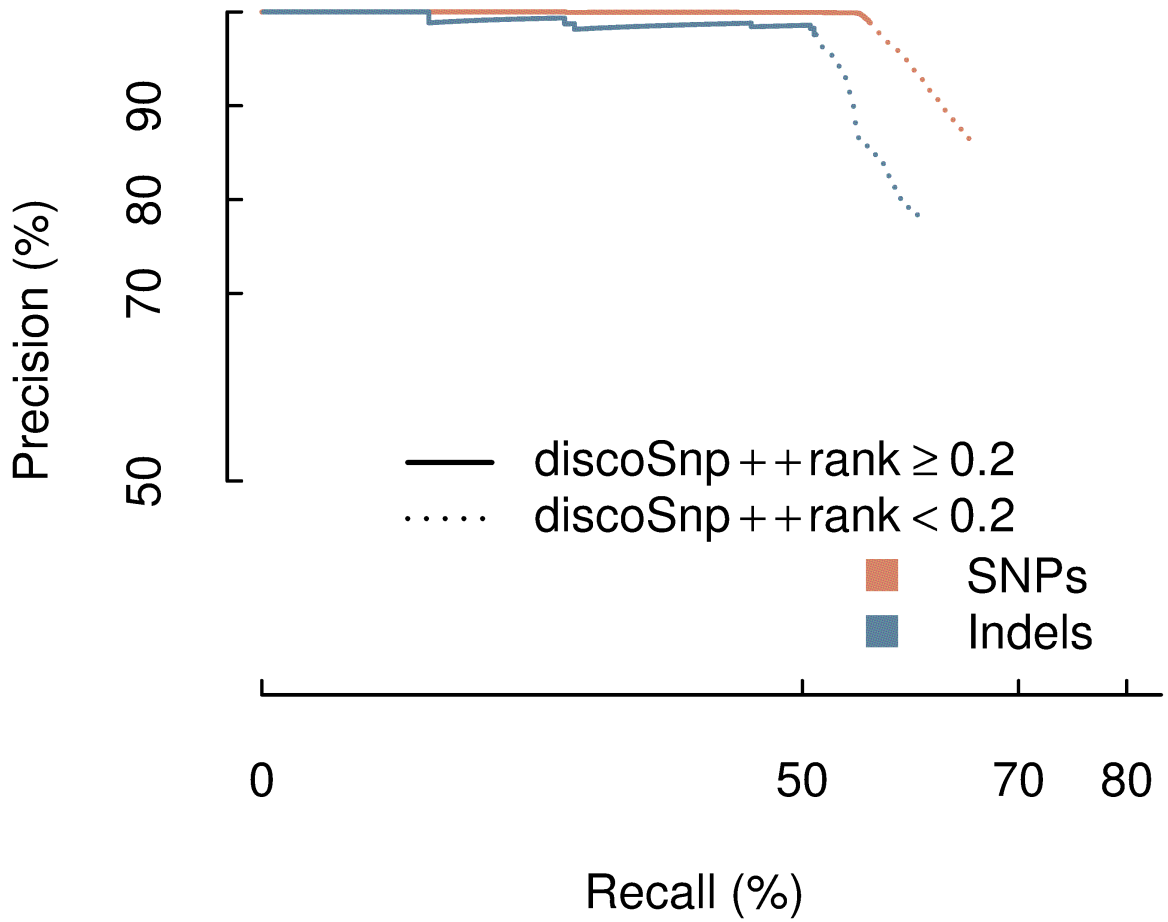
- The validation creates (among others) 3 files:
  - discoRes_k_31_c_auto_D_100_P_1_b_0_coherent.log:

```
--------------------------------
                SNP
7372 SNP in the reference.        Among them 4861 are correctly
predicted
5674 SNP were predicted.          Among them 4861 are correctly mapped
SNP precision     85.67
SNP recall        65.94
```


```
--------------------------------
                INDEL
550 INDEL in the reference.       Among them 339 are correctly
predicted
433 INDEL were predicted.         Among them 339 are correctly mapped
INDEL precision   78.29
INDEL recall      61.64
```

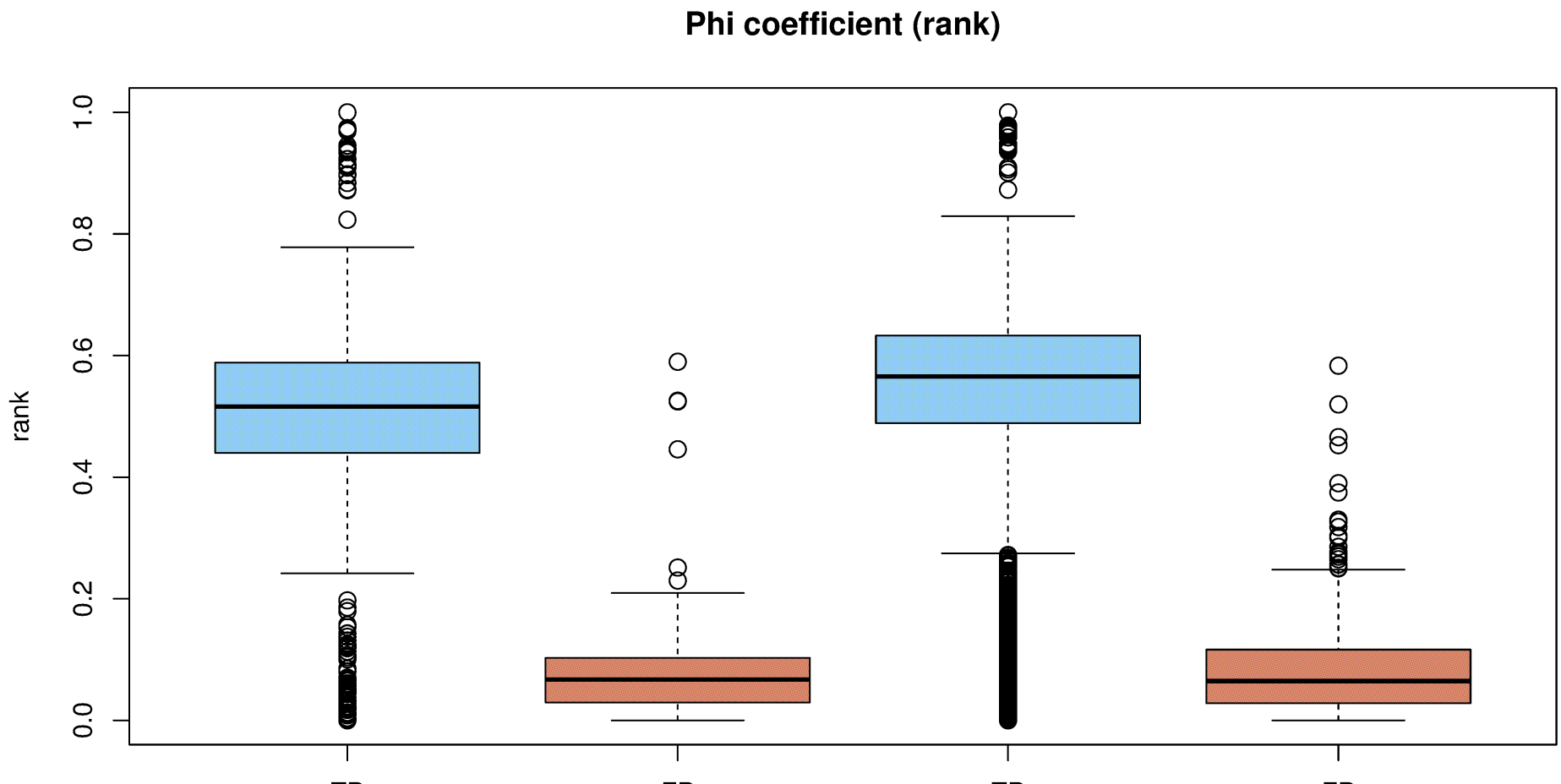
First test -- Validation

- The validation creates (among others) 3 files:
  - discoRes_k_31_c_auto_D_100_P_1_b_0_coherent.png:



First test -- Validation

- The validation creates (among others) 3 files:
  - discoRes_k_31_c_auto_D_100_P_1_b_0_coherent_stat.png:



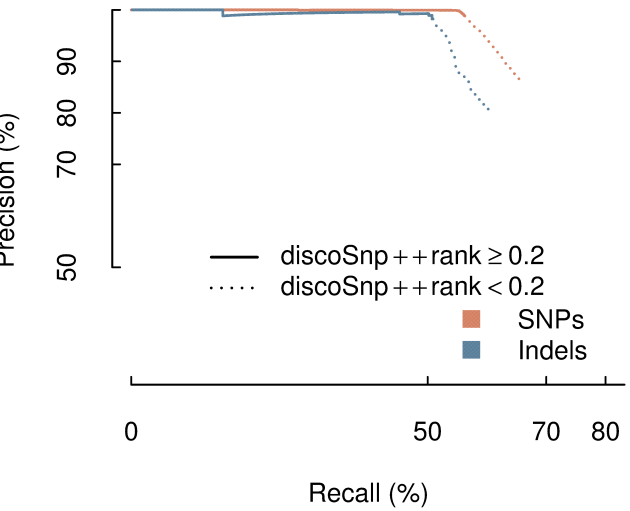
Play with parameters – b (branching strategy)

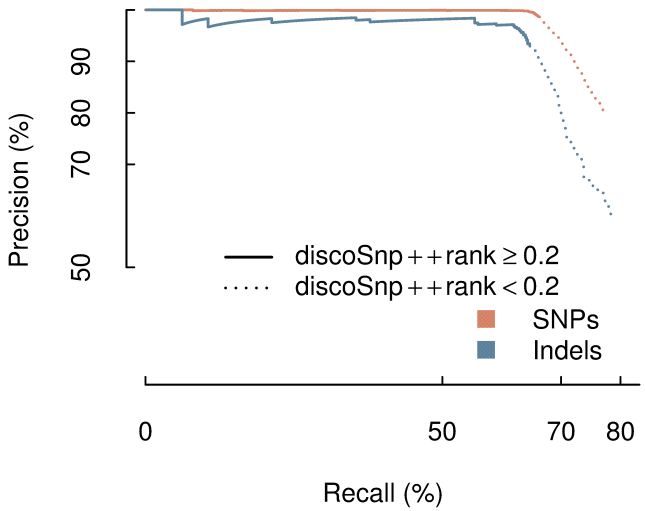
Don’t forget the –g option!

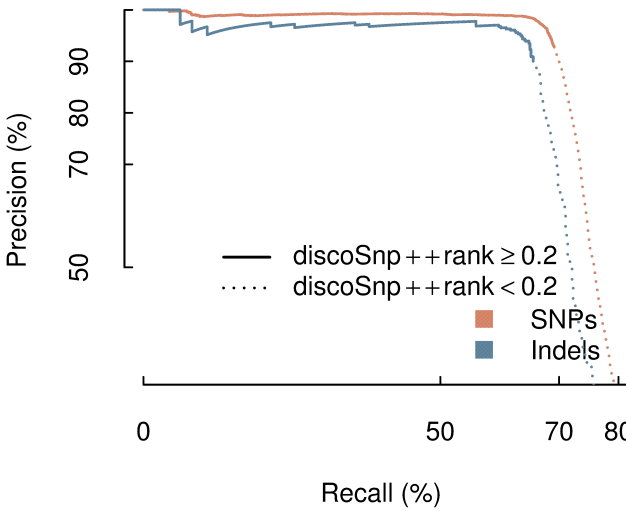
the graph does not depend on this parameter.

-g avoids to reconstruct the graph unless it’s necessary

Play with parameters – b (branching strategy)







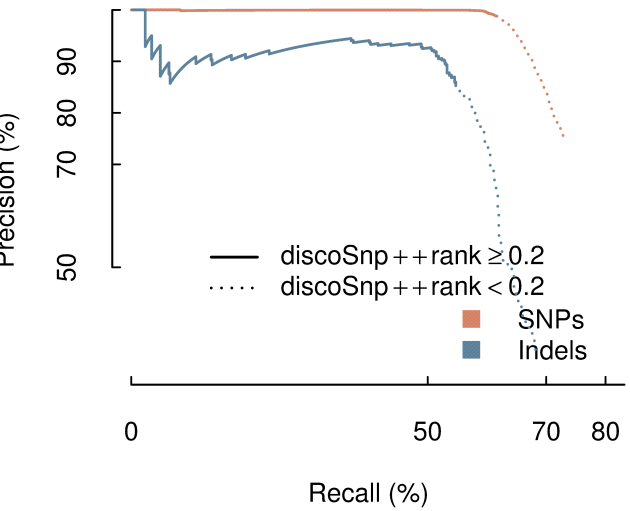
b=0
SNP precision 85.71
SNP recall 65.97
INDEL precision 80.62
INDEL recall 61.27

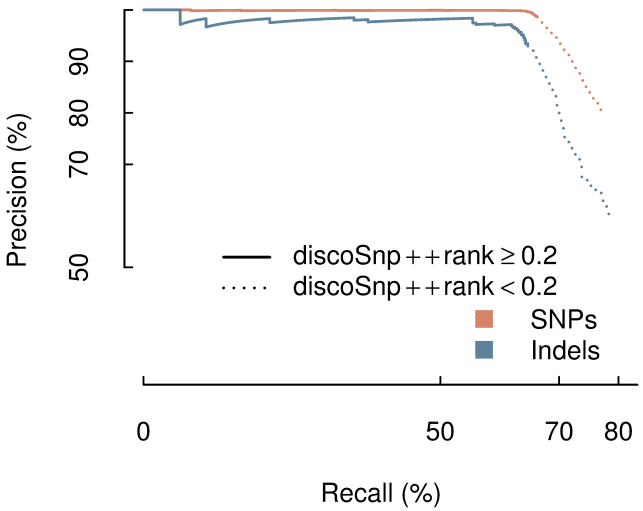
b=1
SNP precision 79.71
SNP recall 77.54
INDEL precision 59.92
INDEL recall 79.09

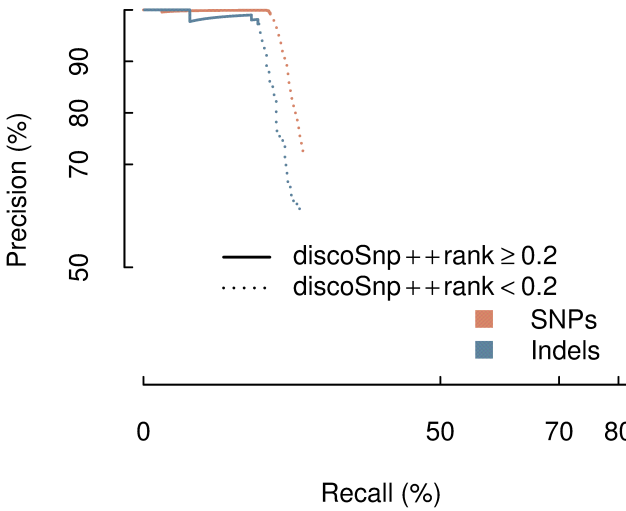
b=2
SNP precision 14.21
SNP recall 83.99
INDEL precision 18.63
INDEL recall 81.27

D=10, k=31, (c=3)

Play with parameters – k







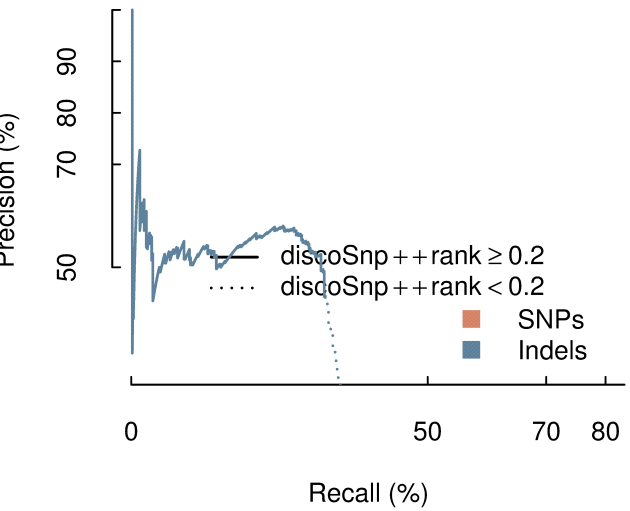
k=21 (c=7)
SNP precision 75.29
SNP recall 73.03
INDEL precision 33.51
INDEL recall 68.73

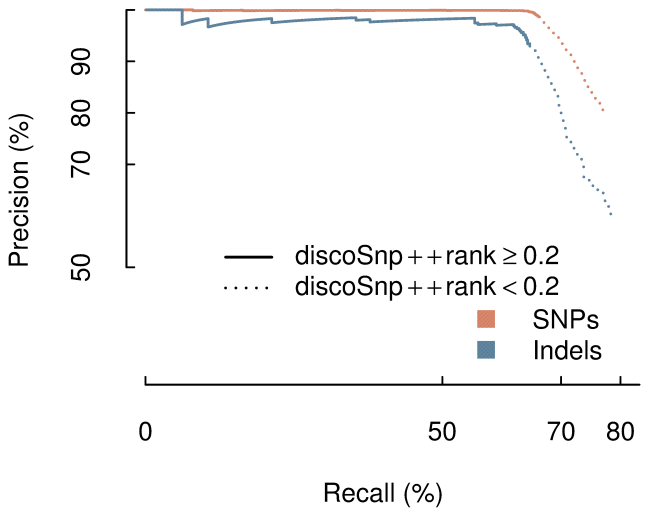
k=31 (c=3)
SNP precision 79.71
SNP recall 77.54
INDEL precision 59.92
INDEL recall 79.09

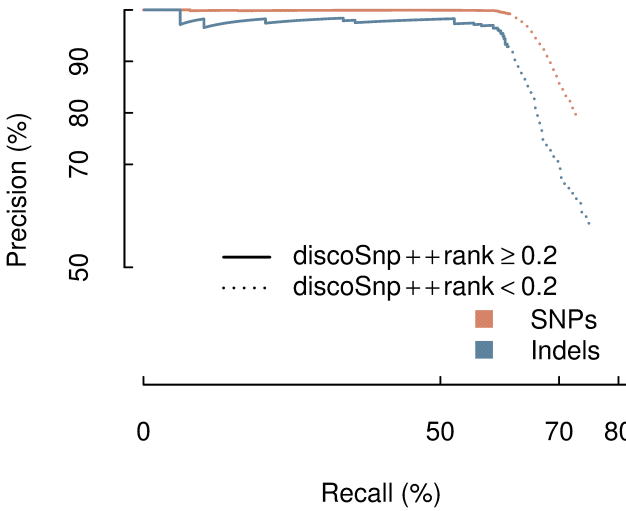
k=61 (c=3)
SNP precision 72.64
SNP recall 26.90
INDEL precision 61.18
INDEL recall 26.36

D=10, b=1

Play with parameters – c (solidity threshold)







c=1

SNP precision 0.57
SNP recall 25.65
INDEL precision 15.49
INDEL recall 39.27

c=3 (auto)

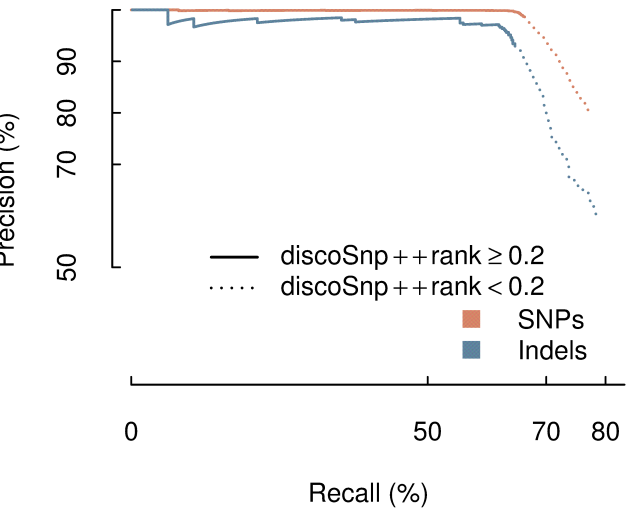
SNP precision 79.71
SNP recall 77.54
INDEL precision 59.92
INDEL recall 79.09

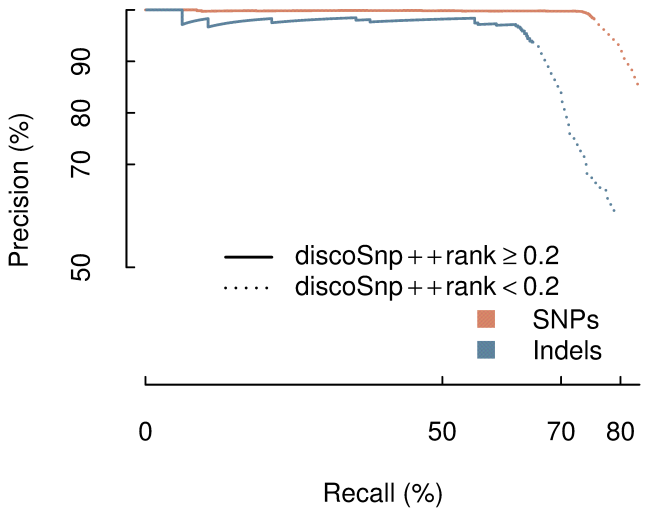
c=5

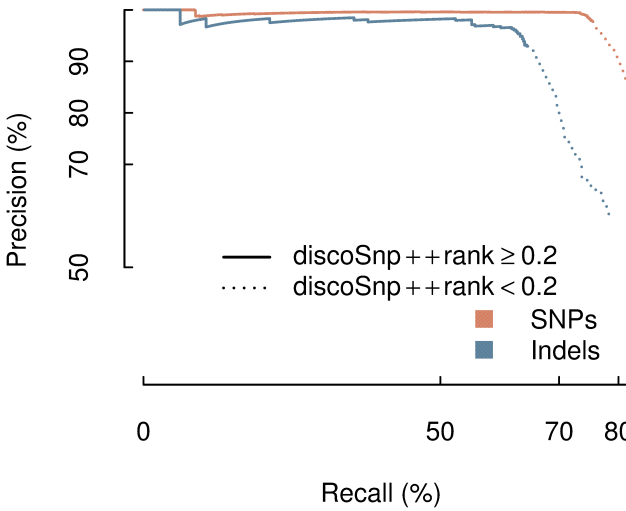
SNP precision 79.71
SNP recall 72.88
INDEL precision 58.12
INDEL recall 75.45

D=10, b=1, k=31

Play with parameters – P (nb close SNPs)







P=1 (default)
SNP precision 79.71
SNP recall 77.54
INDEL precision 59.92
INDEL recall 79.09

P=3
SNP precision 72.52
SNP recall 88.55
INDEL precision 60.33
INDEL recall 79.64

P=10
SNP precision 65.45
SNP recall 88.85
INDEL precision 59.92
INDEL recall 79.09

D=10, b=1, k=31, c auto (=3)

Limitations

Dangerous parameters

- Large P
  - Long enumeration of divergent paths
- Large D
  - Long enumeration of divergent paths
- Small c
  - Lot of spurious kmers. Long enumeration of non coherent paths
- b 2 branching strategy
  - Enumeration of numerous paths