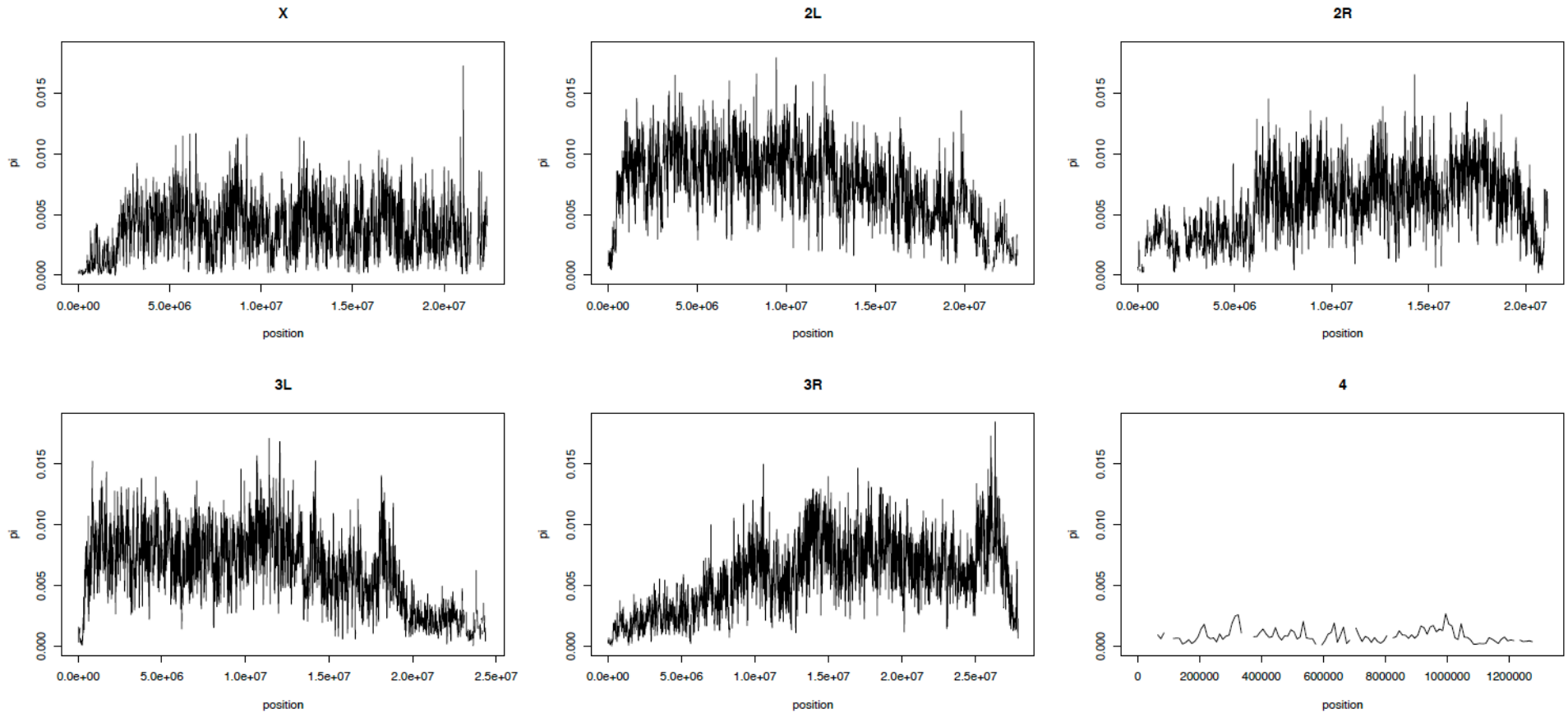
PoPoolation

Estimating natural variation in pooled populations using next generation sequencing

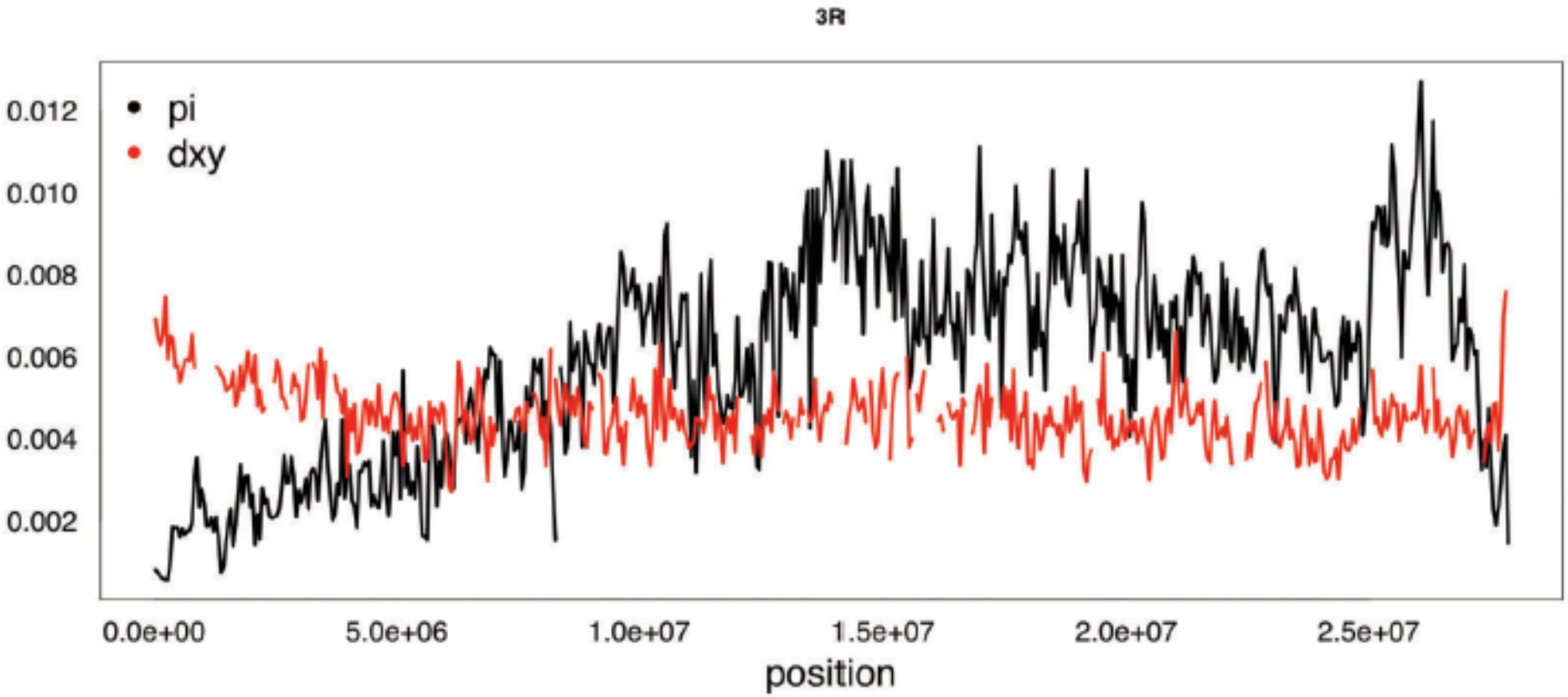
What can you do with PoPoolation?

Genome wide overview of natural variation; example: Pi in *D.mel.*



Detailed inspection of candidate genes; example Cyp6g1 in *D.mel.*

Calculate divergence between species



Calculate natural variation for genes (allows subsequent GO analysis)

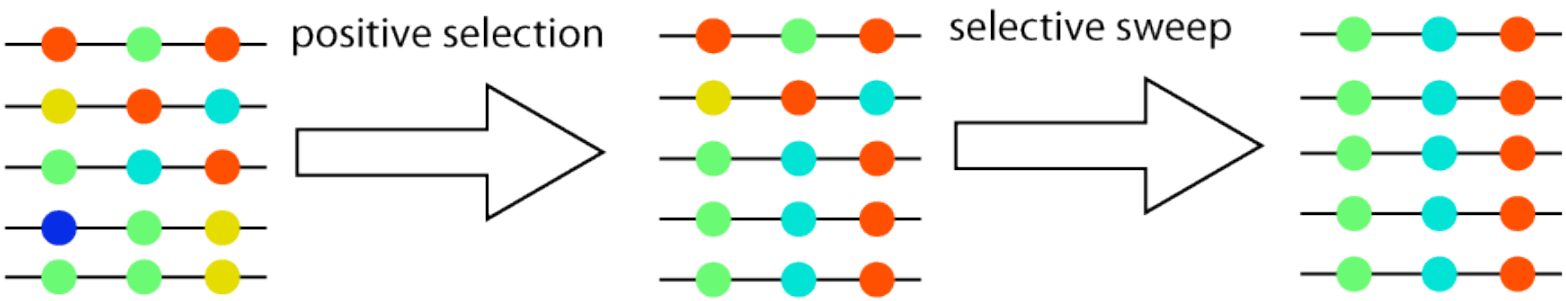
Gene ID	SNPs	cov-frac	Tajima's Pi
CG8493-RB	2	0.940	0.001171327
CG8877-RA	17	0.985	0.001175603
CG13159-RA	1	1.000	0.001213136
CG8290-RB	12	0.987	0.001254184
CG8841-RC	9	0.859	0.001480358
CG8857-RA	4	0.981	0.001549875

Introduction

Quick update on positive selection

When an allele increases in its population frequency, nearby variants also increase in its frequency -> "Hitchhiking"

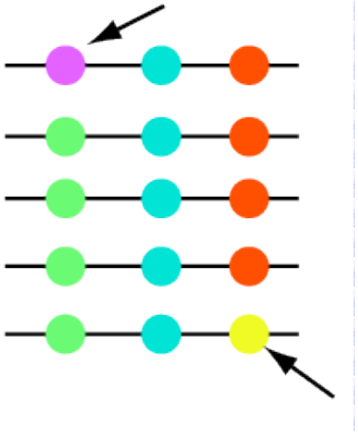
This leads to a selective sweep which erases variation around a positively selected allele



Source: Sabeti P.C. et al. (2006) – Positive Natural Selection in the Human Lineage
Robert Kofler

After the sweep

New mutations appear and restore diversity, but they appear very slowly (mutations are rare) and they are initially of low frequency.



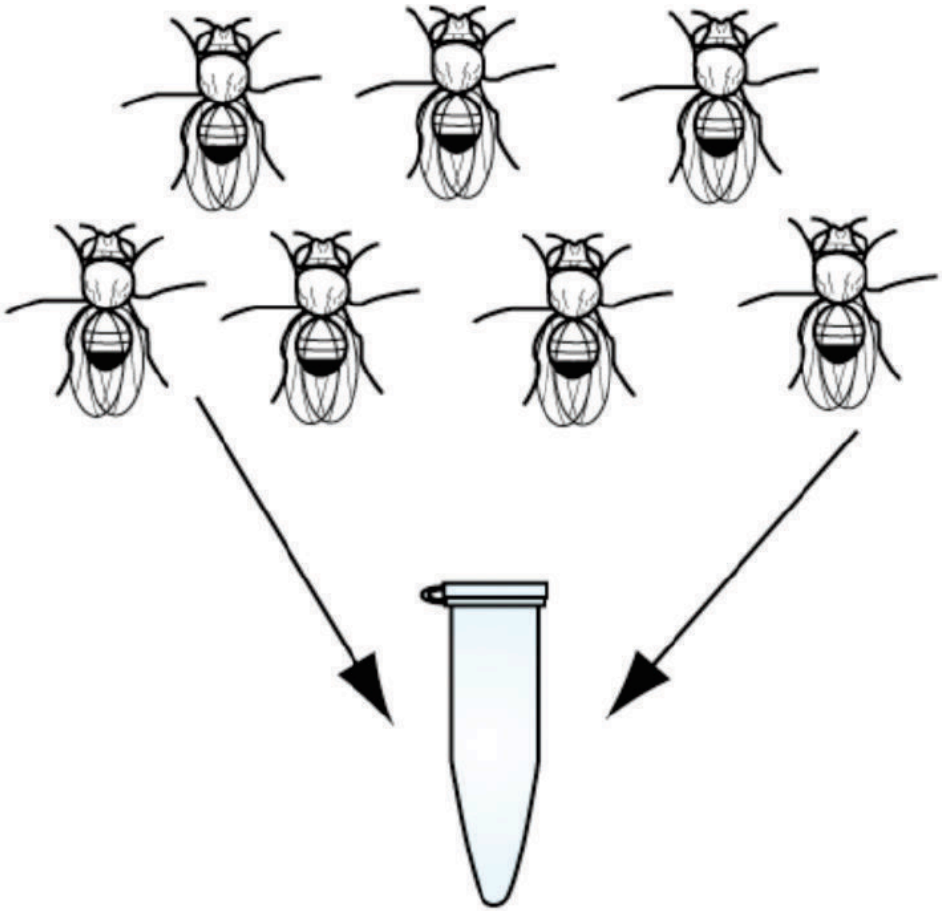
Positive selection thus creates a signature consisting of a region of low overall diversity with an excess of rare alleles => rare alleles are very important (SNP identification)
-> Low Tajima's D

Of course there are many more signatures of positive selection, like the proportion of functional change (McDonald Kreitman Test), length of the haplotypes (iHS and linkage disequilibrium) or differences in allele frequencies between populations (Fst)

PoPoolation currently calculates:

- Tajimas Pi
- Wattersons Theta
- Tajima's D

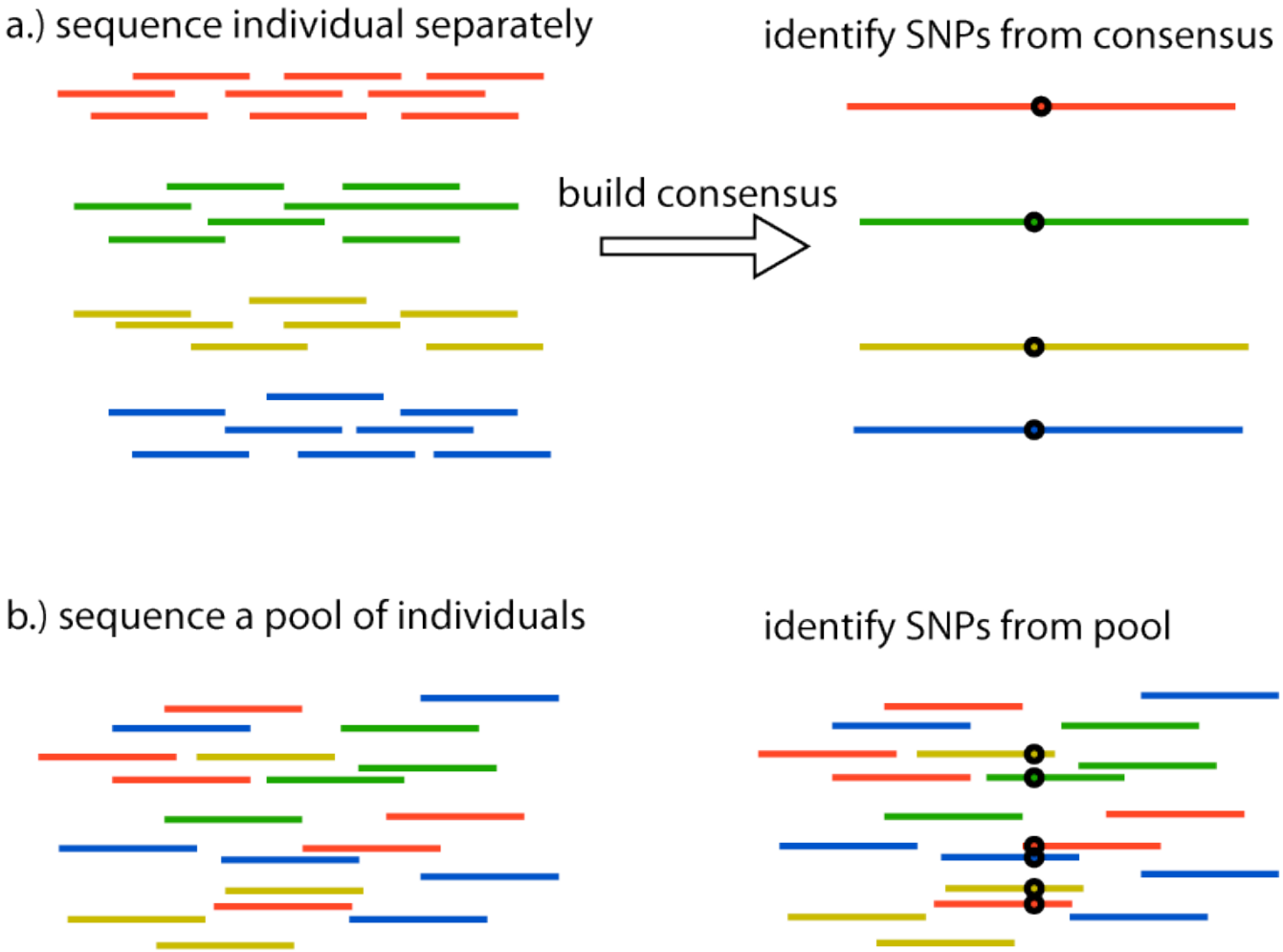
Major requirement for PoPoolation:



A pooled population!

Disclaimer: I expect it to work with sequenced individuals as well, but I have not tested this

Two major strategies for population genomics



Why pooling?
Pros and Cons

Pros:

- More cost effective (less sequencing is required)
- Bioinformatics analysis is simpler (in my opinion..)
- Standard population genetic estimator's (Tajima's D) may easily be assessed

Cons:

- Haplotype information is not available
- All individuals should contribute equal amount of DNA!!
- Illumina reads have a high error rate it is necessary to introduce minimum allele counts -> loosing singletons, doubletons etc
- thus: Pooling requires a correction of standard Population Genetics estimators like Tajima's π (missing singletons and multiple samplings of identical sequences)

We developed correction factors for some population genetic estimators: eg Tajima's D

$$D_{b,pool}(i) = \frac{\theta_{\pi_{b,pool}}(i) - \theta_{W_{b,pool}}(i)}{\sqrt{Var(d_{b,pool})}} \text{ with}$$

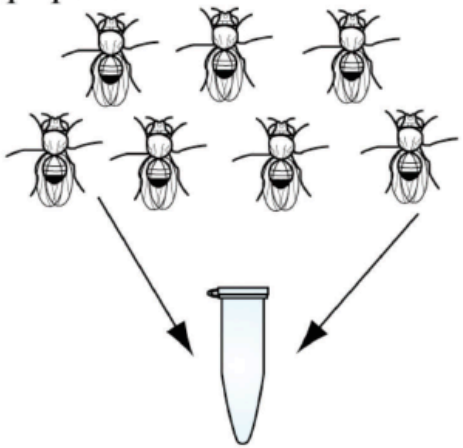
$$Var(d_{b,pool}) = \theta \sum_{m=b}^{C-b} (\theta_{\pi_{b,pool}}(m) - \theta_{W_{b,pool}}(m))^2 \sum_{k=1}^{n-1} P(m|C,n,k) \, {}^{1}\!/_{k} \text{ and}$$

$$P(m|C,n,k) = \binom{C}{m} \left(\frac{k}{n}\right)^m \left(\frac{n-k}{n}\right)^{C-m}$$

see also:
Schloetterer and Futschik (2010):Massively Parallel Sequencing of Pooled DNA Samples--The Next Generation of Molecular Markers.

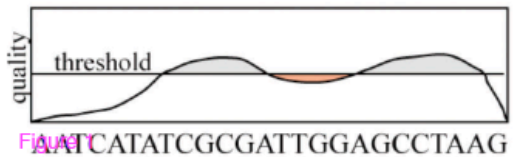
PoPoolation overview

1.) Extract DNA of a population



2.) Sequence DNA (e.g.:Illumina)

3.) Trim reads by base quality (fastq-files)



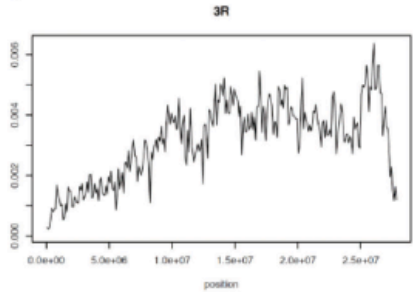
4.) Align reads to reference genome (e.g.: BWA, Bowtie)

⇨ SAM-file

5.) Filter ambiguously mapped reads (e.g.: using mapping quality and samtools)

6.) Create a pileup file (e.g.: using samtools)

7.) Run PoPoolation



PoPoolation implements these correction factors

Genome wide pattern of variability may be assessed

Recent positive selection may be identified

Validating PoPoolation

Validation of PoPoolation using simulated data

- Simulated Tajima's Pi along a chromosome of D.melanogaster using ms
- We created artificial reads and introduced sequencing errors
- We fed these artificial reads into the PoPoolation pipeline
- We compared the observed with the expected Tajima's Pi

Difference between the observed and the expected Tajima's Pi

		Cov 50	Cov 100	Cov 250
MAC 1	Error Rate 1%	3.931392	3.937907	3.935399
	Error Rate 0.2%	0.815916	0.825277	0.827133
	Error Rate 0.1%	0.412683	0.4206	0.423819
MAC 2	Error Rate 1%	0.720516	1.363739	2.815574
	Error Rate 0.2%	0.040738	0.076093	0.165799
	Error Rate 0.1%	0.020158	0.03142	0.0576
MAC 3	Error Rate 1%	0.093397	0.254378	1.118804
	Error Rate 0.2%	0.00905	0.020727	0.033771
	Error Rate 0.1%	0.011699	0.014258	0.019995

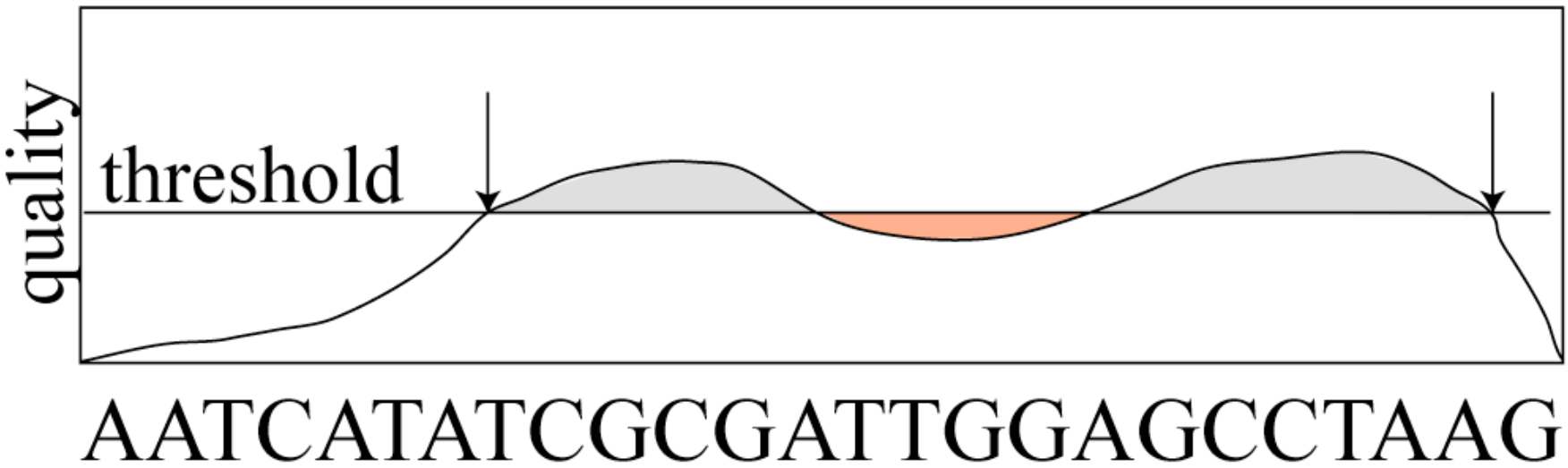
Cov: Coverage, MAC: minor allele count

⇒ low error rate of 0.1% is necessary and a mac >=2
⇒ Illumina has an error rate of ~1%

How to decrease the error rate

- Trimming of reads; remove low quality stretches
- Require a minimum quality
- Require a minimum allele count (mac)

Trimming algorithm of PoPoolation



=>The algorithm finds the highest scoring substring of the read
=>Individual bases may even be below the quality threshold, as long as a new high score can be achieved
=>The algorithm is very similar to dynamic programming (Smith-Waterman)
=> handles single end as well as paired end reads!!

Trimming statistic with different quality thresholds

Table 1: Trimming statistics of $14x10^6$ reads

	No trimming	0*	10	20	30
% reads passing trimming	100	99.73	91.93	88.92	33.49
Sum read length [Mbp]	1081.22	1077.42	960.57	912.08	298.65
Average read length	76.00	75.94	73.45	72.10	62.68
Average quality	27.50	27.56	29.51	29.90	32.23

*0: trimming includes removal of 'N'-characters at the end of reads

Quality and minor allele count

minimum allele count: 2

reads:

```
..gca A aca..
..gca T aca..
..gca T aca..
..gca A aca..
..cca A aca..
..gca A aca..
..gca T aca..
```

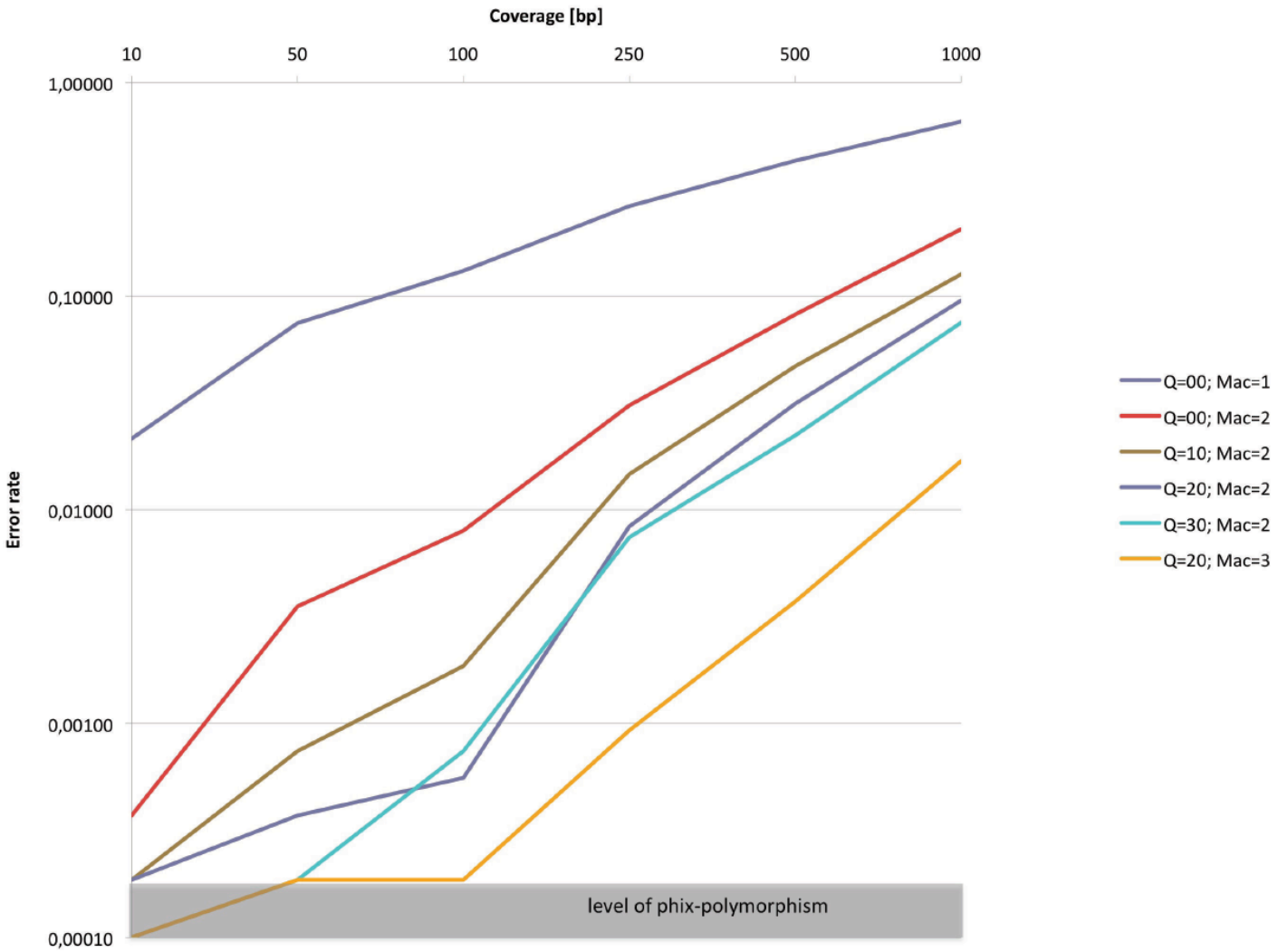
X-chr: `..GCA T ACA..`

		mq15	mq20
q32	A	A	A
q16	T	T	
q22	T	T	T
q31	A	A	A
q22	A	A	A
q10	A		
q11	T		

SNP

mq.. minimum quality

Identification of false positive SNPs using PhiX and PoPoolation



novel error rate (after trimming and quality control)

- Remember: the required error rate is 0.1-0.2%
- Remember: Illumina has an error rate of ~1%
- Trimming and the requirement for a minimum quality dramatically reduce the error rate:

trim	min.qual.	Error rate
20	0	0.15%
20	20	0.07%

=> Error rate is sufficiently reduced by trimming and minimum quality

Difference between the observed and the expected Tajima's Pi

		Cov 50	Cov 100	Cov 250
MAC 1	Error Rate 1%	3.931392	3.937907	3.935399
	Error Rate 0.2%	0.815916	0.825277	0.827133
	Error Rate 0.1%	0.412683	0.4206	0.423819
MAC 2	Error Rate 1%	0.720516	1.363739	2.815574
	Error Rate 0.2%	0.040738	0.076093	0.165799
	Error Rate 0.1%	0.020158	0.03142	0.0576
MAC 3	Error Rate 1%	0.093397	0.254378	1.118804
	Error Rate 0.2%	0.00905	0.020727	0.033771
	Error Rate 0.1%	0.011699	0.014258	0.019995

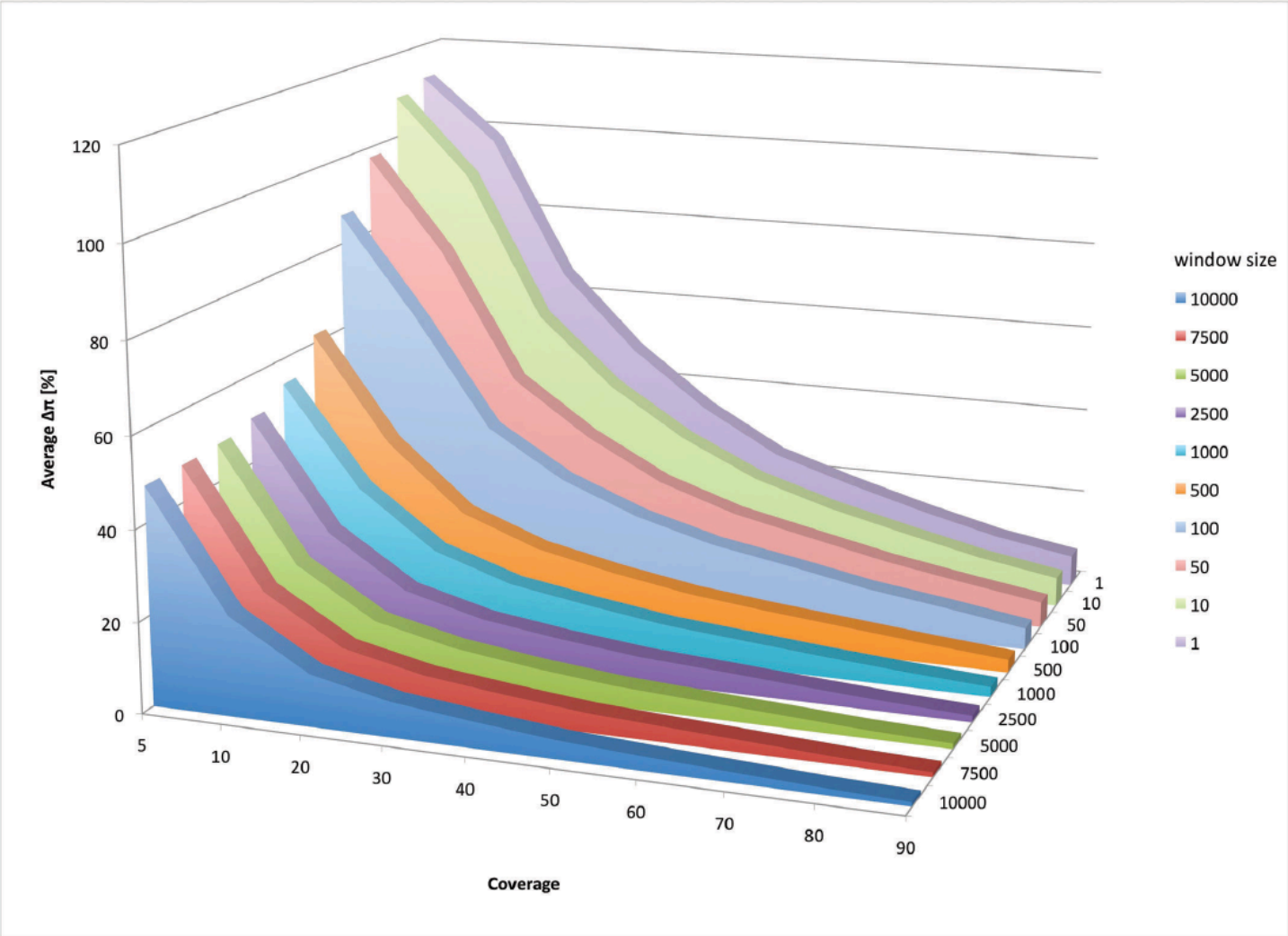
Cov: Coverage, MAC: minor allele count

⇒ low error rate of 0.1% is necessary and a mac >=2
⇒ Illumina has an error rate of ~1%

Influence of coverage and window size

- We sequence chromosome 3R of D.mel to 100x coverage
- We randomly drew a subsets of the reads to achieve varying coverages (5x -> 90x)
- We compared Pi of the subset (eg. cov.: 10x) with the Pi of the full data set (cov.: 100x)
- Furthermore we used different window sizes

Influence of coverage and window size (average of 2000 windows)



Walktrough

Preconditions:

- Mac or Linux (Unix)
- Perl and R installed
- bwa (Burrows-Wheeler Alignment Tool)
- samtools
- IGV (Integrative Genomics Viewer)
- PoPoolation 1.016

http://code.google.com/p/popoolation

=> bwa and samtools need to be in the $PATH

Get the data:

Webpage:

http://code.google.com/p/popoolation

go to ‘Downloads’ and download:

popool-teaching.sh

teaching-data.zip

Preparations:

- Unzip teaching-data.zip
- copy the file popool-teaching.sh into the unziped folder (data)
- enter the command line
- change directory into the unziped folder

```
Terminal — zsh — 94×11
robertkofler@i122mc121 %ll
total 115704
-rw-r--r--@ 1 robertkofler  staff    136886 Aug 24 14:12 cyp6g1.gtf
drwxr-xr-x@ 3 robertkofler  staff       102 Aug 26 14:57 data
-rw-r--r--@ 1 robertkofler  staff  21569799 Aug 24 14:12 dmel-2R-chromosome-r5.22.fasta
-rw-r--r--@ 1 robertkofler  staff      1781 Aug 26 14:27 popool-teaching.sh
-rw-r--r--@ 1 robertkofler  staff      1754 Aug 24 15:21 protocol
-rw-r--r--@ 1 robertkofler  staff  11279782 Aug 24 14:12 read_1.fastq
-rw-r--r--@ 1 robertkofler  staff  11279782 Aug 24 14:12 read_2.fastq
-rw-r--r--@ 1 robertkofler  staff  14956062 Aug 26 14:27 teaching-data.zip
robertkofler@i122mc121 %            /Users/robertkofler/Documents/Lectures/Sandbjergen/tmp
```

Trimming

Enter the command:

```
perl <local-popoolation-installation>/basic-
  pipeline/trim-fastq.pl
  --input1 read_1.fastq --input2 read_2.fastq
  --output trim --quality-threshold 20
  --min-length 50
```

<local-popoolation-installation> ... this is the path to your copy of popoolation

e.g.: /Users/robertkofler/dev/popoolation-1.016

Trim statistics

```
FINISHED: end statistics
Read-pairs processed: 52322
Read-pairs trimmed in pairs: 52322
Read-pairs trimmed as singles: 0


FIRST READ STATISTICS
First reads passing: 52322
5p poly-N sequences trimmed: 30
3p poly-N sequences trimmed: 124
Reads discarded during 'remaining N filtering': 0
Reads discarded during length filtering: 0
Count sequences trimed during quality filtering: 20378

Read length distribution first read
length  count
50      332
51      335
52      349
```

Prepare the reference sequence for mapping

Command line:

```
mkdir wg
mv dmel-2R-chromosome-r5.22.fasta wg
awk '{print $1}' wg/dmel-2R-chromosome-r5.22.fasta >
  wg/dmel-2R-short.fa
bwa index wg/dmel-2R-short.fa
```

akw ‘{print $1}’ .. this only prints the first column. For a fasta file this removes anything after the first space. Therefore this command shortens the header for example:
>2R name=blabla id=12345 transposons=many
will be shortend to
>2R
=> more reliable for mapping and downstream processing

mapping using 'BWA'

command line:

```
bwa aln wg/dmel-2R-short.fa trim_1 > trim_1.sai
bwa aln wg/dmel-2R-short.fa trim_2 > trim_2.sai
bwa sampe wg/dmel-2R-short.fa trim_1.sai trim_2.sai
  trim_1 trim_2 > maped.sam
```

these are subotpimal parameters as we do not want to wait the rest of the day for the mapping to finish!

for optimal results we recommend the following:

```
bwa aln -l 100 -o 2 -d 12 -e 12 -n 0.01 wg/dmel-2R-
  short.fa trim_1 > trim_1.sai
```

Check the sam-file

Command line:

```
less maped.sam
```

Terminal — less — 105×13

```
@SQ     SN:2R   LN:21146708
HWUSI-EAS300R:7:1:7:674#0       83      2R      7923527 60      74M     =       7923393 -208    TAACTTATT
TCCTAGGTATATTAATTATAAGTTACTAACTCTATAAATTATTCATGATTTGTAAACTACAAAAG      a``b_aaa`aaabb\`aaababbbaab`abaaba
aaa`baabbabbb_abaabba`ababaaba_a`V[_bbba      XT:A:U  NM:i:0  SM:i:37 AM:i:37 X0:i:1  X1:i:0  XM:i:0  X0:
i:0  XG:i:0  MD:Z:74
HWUSI-EAS300R:7:1:7:674#0       163     2R      7923393 60      70M     =       7923527 208     CCCATTTCC
TTCCAATTAGTTCAATAAGTTCACCAGTATTAGACTTTANNNGCCCACTTCGGAATTTGTA  Y[Z[W[[Y[YZU[YZW[[[U[WZ[[[Z[[Y[Y[[[[V[YY[W
[[Y[YODDDLYS[[W[YZ[YYWXYY[[Y  XT:A:U  NM:i:3  SM:i:37 AM:i:37 X0:i:1  X1:i:0  XM:i:3  X0:i:0  XG:i:0  MD:
Z:48T0A0C19
HWUSI-EAS300R:7:1:9:1897#0      99      2R      8172056 60      74M     =       8172232 248     ACTTAATTA
TTTATGCTTTTCTCTACTCTGCACGGCATGCAAATGCAATATAGATGCAAGGCGAGCCGAAACAA      aaab\`bb^bababbaababbbbbababbbbaba
aaaabbbb[_bba_`^aabaabaaaa__aaaaaa[Saa_`      XT:A:U  NM:i:0  SM:i:37 AM:i:37 X0:i:1  X1:i:0  XM:i:0  X0:
:
```

create a pileup file

First remove ambiguously mapped reads (minimum mapping quality 20) and create a sorted bam-file:

```
samtools view -q 20 -bS maped.sam| samtools sort -
  maped.sort
```

Than create a pileup file:

```
samtools pileup maped.sort.bam > cyp6g1.pileup
```

Control the pileup file:

```
less cyp6g1.pileup
```

pileup?

```
reads:  ..gca A aca..
        ..gca T aca..
        ..gca T aca..
        ..gca A aca..
        ..cca A aca..
        ..gca A aca..
        ..gca T aca..
X-chr:  ..GCA T ACA..
```

resulting pileup entry:

X-chr	2312	T	7	A..AAA.	SUUTTBB

Calculate Tajima's Pi using a sliding window approach

first check out the options:

```
perl <local-popoolation-installation>/Variance-
  sliding.pl --help
```

and run the tests (sanity control):

```
perl <local-popoolation-installation>/Variance-
  sliding.pl --test
```

than calculate Tajima's Pi

```
perl <local-popoolation-installation>/Variance-
  sliding.pl --measure pi --input cyp6g1.pileup --
  min-count 2 --min-qual 20 --min-coverage 4 --max-
  coverage 70 --pool-size 500 --window-size 1000 --
  step-size 1000 --output cyp6g1.varslid.pi --region
  2R:7800000-8300000
```

Output:

```
less cyp6g1.varslid.pi
```

Terminal — less — 48×11

```
2R      7800500 2       0.607   0.002824208
2R      7801500 7       0.959   0.003533902
2R      7802500 21      1.000   0.008611489
2R      7803500 4       0.999   0.002103835
2R      7804500 9       0.829   0.005926810
2R      7805500 22      1.000   0.014296730
2R      7806500 3       0.990   0.001243119
2R      7807500 17      0.971   0.010188391
2R      7808500 20      0.939   0.012196236
2R      7809500 7       0.984   0.003776529
:
```

```
        col 1: reference chromosome
        col 2: position in the reference chromosome
        col 3: number of SNPs in the sliding window; These SNPs have been
used to calculate the value in col 5
        col 4: fraction of the window covered by a sufficient number of re
ads. Suficient means higher than min-coverage and lower than max-coverage
        col 5: population genetics estimator (pi, theta, D)
```

Prepare output for IGV

Command line:

```
perl <local-popoolation-installation>/
  VarSliding2Wiggle.pl --input cyp6g1.varslid.pi --
  output cyp6g1.pi.wig --trackname "nat-pop-pi”
```

Index the bam file:

```
samtools index maped.sort.bam
```

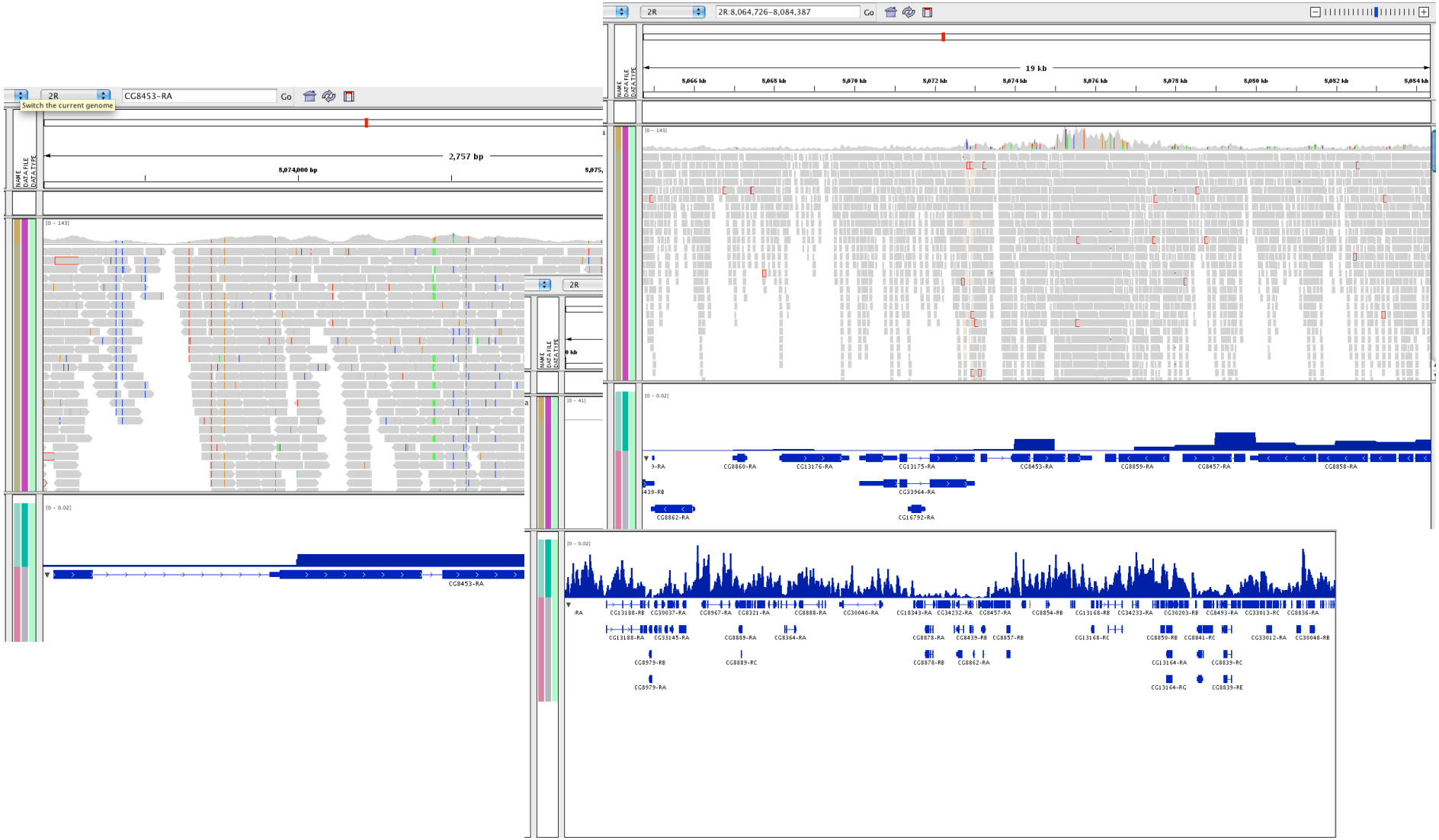
Why not directly a wiggle as output??

you loose information in wiggle, e.g.: snp count or covered fraction

Visualize in IGV

- open IGV (`./igv_mac-intel.command`)
- File -> Import Genome...
  Name: dmel-2r
  Sequence File: dmel-short.fa
- File -> Load from File...
  cyp6g1.gtf
  maped.sort.bam
  cyp6g1.pi.wig

Biology! Here I come..



You may try this at home!

The WebPage contains the full guide:

http://code.google.com/p/popoolation/wiki/TeachingPoPoolation

$PATH

vi the config file for your shell

BASH: ~/.bash_profile

ZSH: ~/.zshrc

add the following line adapted to your system:

```
export PATH=/Users/robertkofler/programs/bwa-0.5.7:/Users/robertkofler/
    programs/samtools-0.1.7_i386-darwin:$PATH
```

load the new configuration:

```
source ~/.zshrc
```

test:

```
bwa
```