What can you do with S–MART to analyze your RNA-Seq data?

Matthias Zytnicki
URGI — INRA

ALIMENTATION
AGRICULTURE
ENVIRONNEMENT

Outline

Aim

- Be able to perform a long and undirected RNA-Seq analysis.
- Spot and extract highly transcribed regions.
- Produce a first *de novo* annotation.
- Get new transcription units.
- Highlight discrepencies between reference annotation and *de novo* annotation.
- Use WIG data.

Data

We will need:

- Some reads: the (smaller) `sample 1_2.sam` data set. Data have been mapped to the genome, the format is `SAM`. The `BAM` is also provided for visualization proposes.
- An annotation: a TAIR annotation of *A. thaliana*, `annotation.gtf`, in `GTF` format.
- Genomic sequence: the TAIR10 genome of *A. thaliana*, `genome.fasta`, in `FASTA` format.

The data is available in
`Shared data` → `Data Libraries` → `tp-mardi-smart-matthias`.

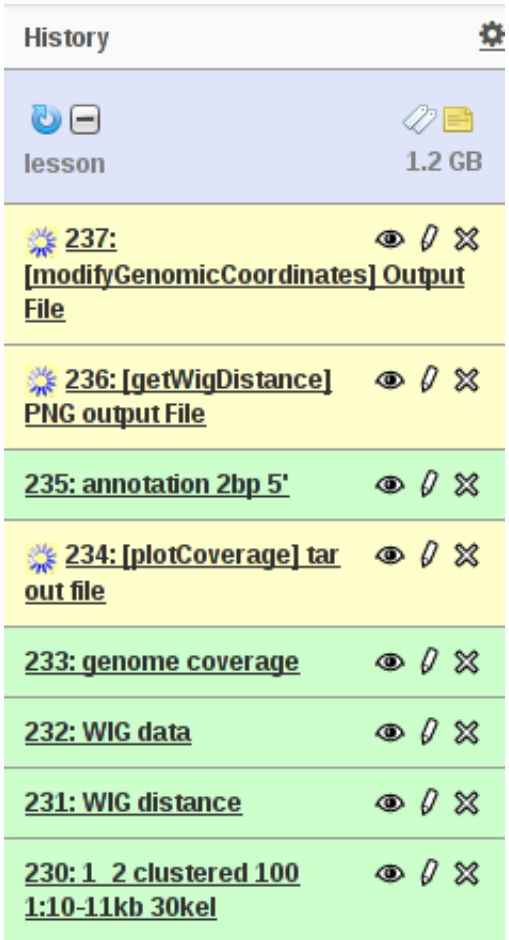
Hints — 1

- Please rename each step in you history!

- You could also create a new History for each analysis.

Hints — 2

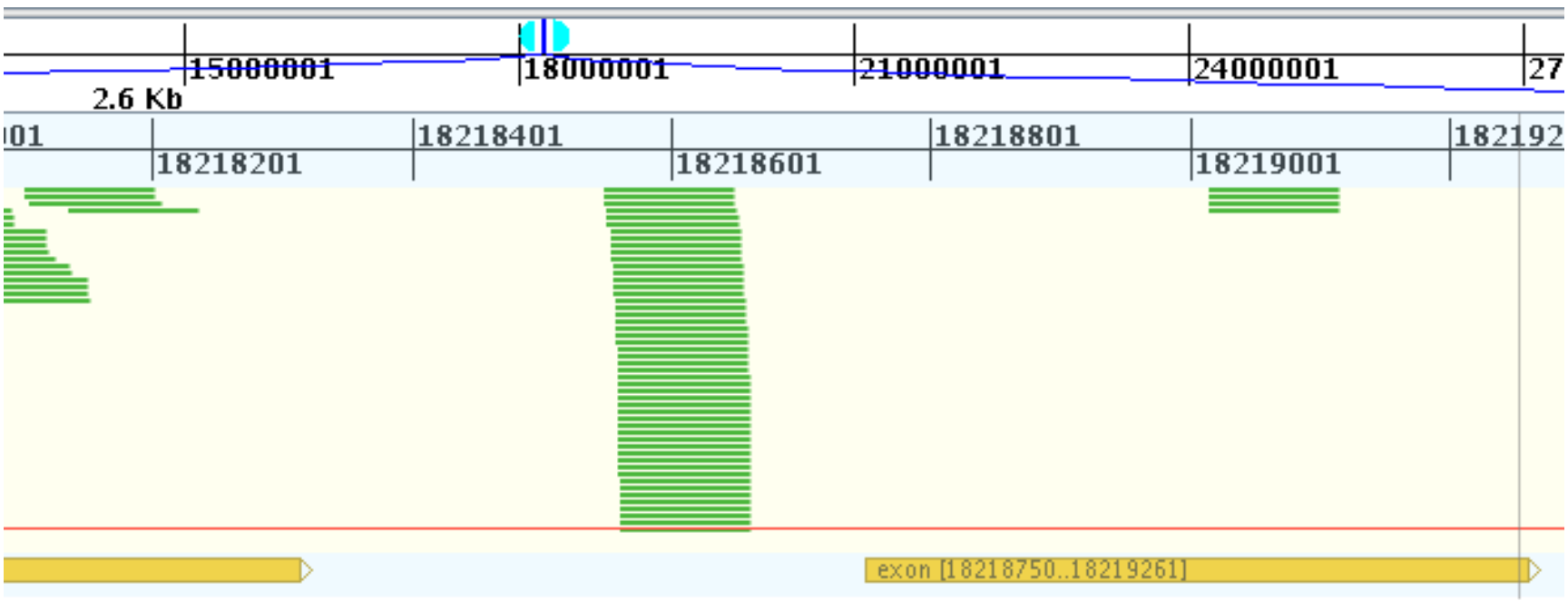
- Some tasks take time. Do not wait for the end of a job to start the next one.
  (Except if the next task needs the output of the previous task.)



Hints — 3

Visualize your results as often as possible

- Download the genome, the annotation, and the reads (in `BAM` format).
- Start IGV.
- Load the data.



Hints — 4

I prefer GenomeView

- Go to `http://genomeview.org/`.
- Start the tools by clicking on `Launch` (left side).
- Accept to run the tool.
- Dismiss the "Genome Explorer" dialog.
- Load data with `File` → `Load data` → `Local files`.
- Load the genome, the annotation, the reads (in that order).
- Create the index when asked.
- Goto position 1:18,217,800 (you may use `Ctr+G`).

Hints — 5

Check your annotation file

- View the annotation file `annotation.gtf`.
- It contains many types of information; here, we will work on transcripts only.
- Always the annotation file:
- tool: Clean Transcript File
- format: `GTF`
- input file: `annotation.gtf`
- other options are left as is.

→ We will always use this cleaned annotation file from now on.

WIG format

Definition

The WIG format stores 1 value for each nucleotide of the genome.

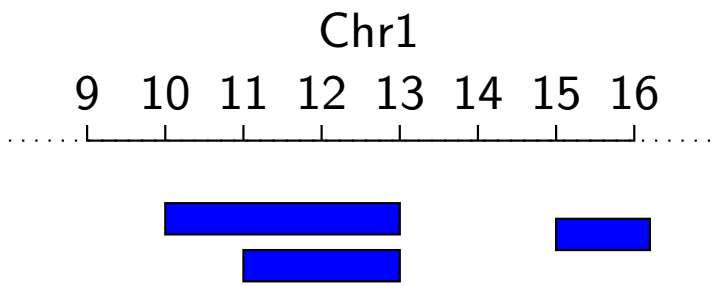
Format

There are many format. We will use this one:

```
variableStep chrom=Chr1
10 1
11 2
12 2
13 2
15 1
...
```

Example

The previous example refers to:

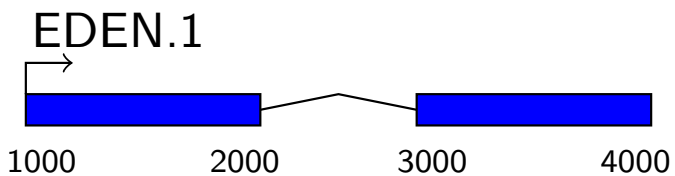


GFF format

- Except otherwise stated, S–MART usually outputs GFF3 files. This is the default output file format.
- S–MART can read the final 9th field of a GFF3 file.
- S–MART sometimes adds tags in the 9th field.

Example

```
Chr1 . mRNA 1000 4000 . + . ID=mRNA1;Name=EDEN.1
Chr1 . exon 1000 2000 . + . ID=exon1;Parent=mRNA1
Chr1 . exon 3000 4000 . + . ID=exon2;Parent=mRNA1
```



Outline

Step 1

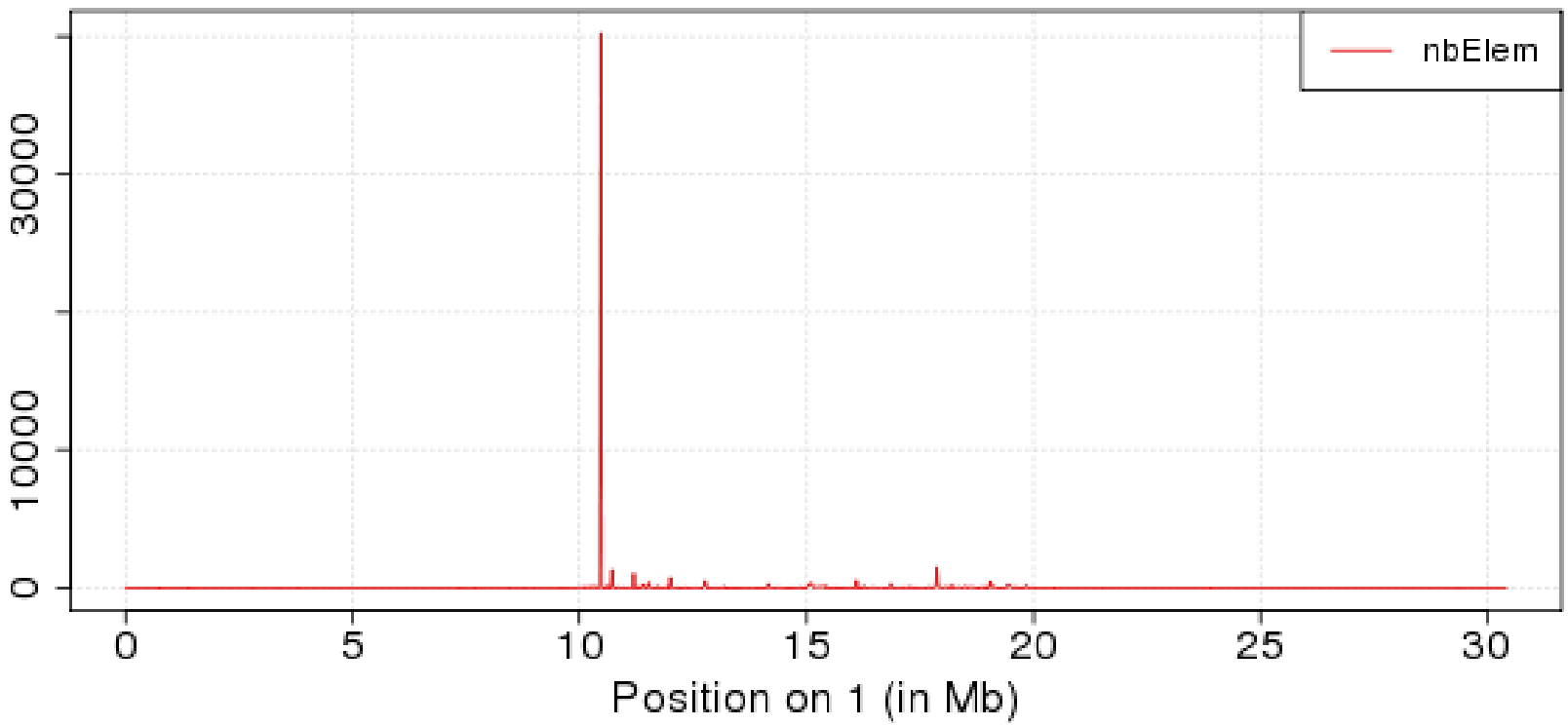
Plot the data

- tool: get distribution
- input format: `SAM`
- input file: `sample 1_2.sam`
- reference genome file: `genome.fasta`

→ notice a peak at position Chr1:10M–11M

Step 1

Figure

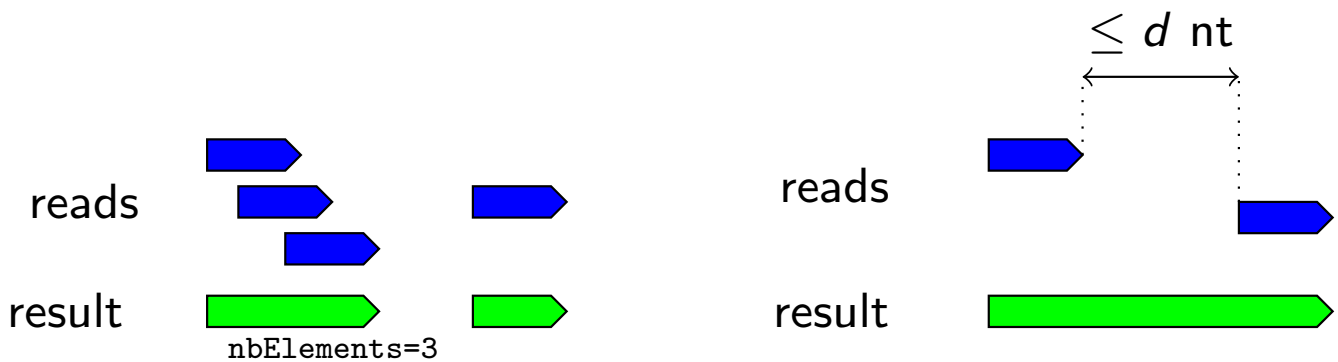


Step 2

Cluster the reads

- tool: clusterize
- input format: `SAM`
- input file: `sample 1_2.sam`
- colinear: yes
- distance: 100nt

What is clustering?



Step 3

Zoom on Chr1:10M–11M

- tool: restrict transcript list
- input format: `GFF`
- input file: the previous file
- chromosome name: yes, then 1
- restrict start: yes, then 10000000
- restrict end: yes, then 11000000

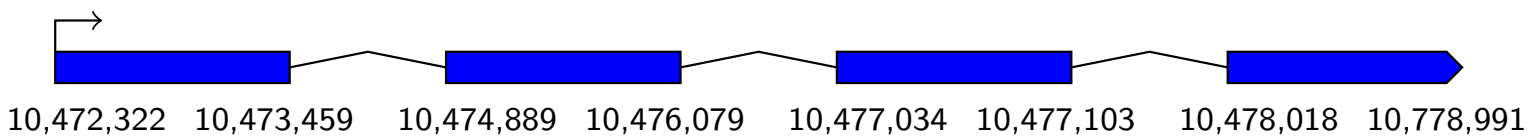
Step 4

Retrieve highly transcribed data

- tool: select by tag
- input format: `GFF`
- input file: the previous file
- tag option: `nbElements`
- minimum value: yes, then 30000
- if value is not present: yes, then 1

→ You have 5 lines: 1 line for the "transcript," and 4 for the "exons." It is only 1 transcription unit, which gathers 34,879 reads.

Transcription unit



Step 5

Get the corresponding genomic sequence

- tool: coordinates to sequence
- input format: `GFF`
- file: the previous file
- fasta file: `genome.fasta`

→ The sequence corresponds to the "mature" mRNA, *i.e.* without introns.

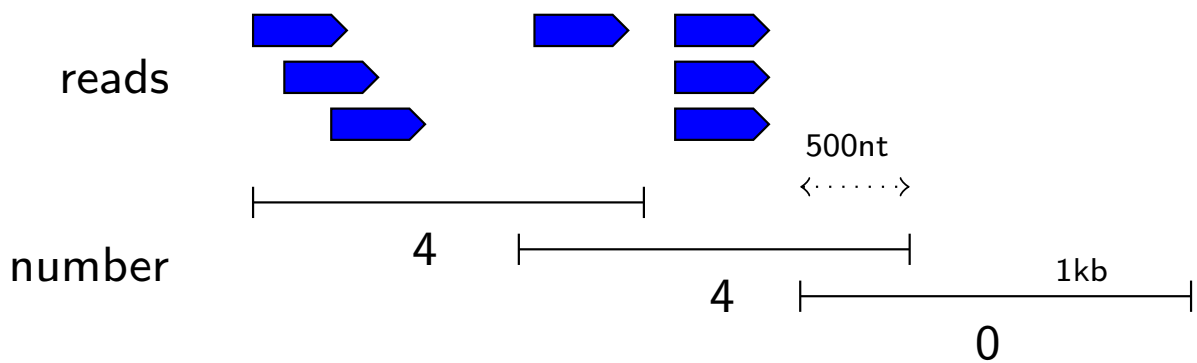
Outline

Step 1

Count the reads in each sliding window

- tool: clusterize By SlidingWindows
- input format: `SAM`
- file: `sample 1_2.sam`
- size: 1000
- overlap: 500

Sliding windows



Step 2

Select the windows with $\geq$ 10 reads

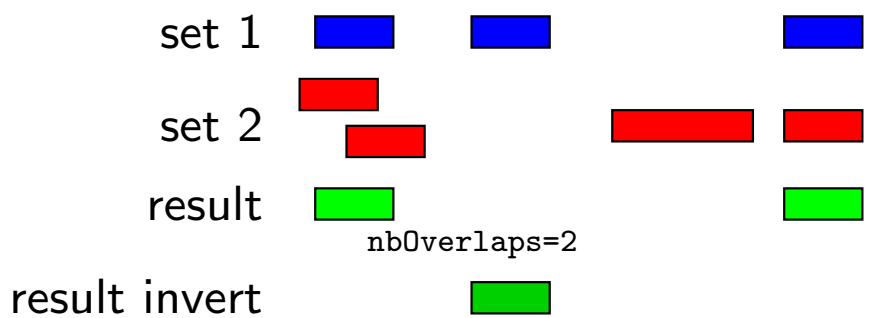
- tool: select by tag
- input format: `GFF`
- file: previous file
- minimum: yes, then `10`
- default: yes, then `1`

Step 3

Remove data which match with the annotation

- tool: Compare Overlapping Small Query
- file format 1: `GFF`
- file format 2: `GTF`
- file 1: previous file
- file 2: the clean annotation
- invert match: yes

Comparison



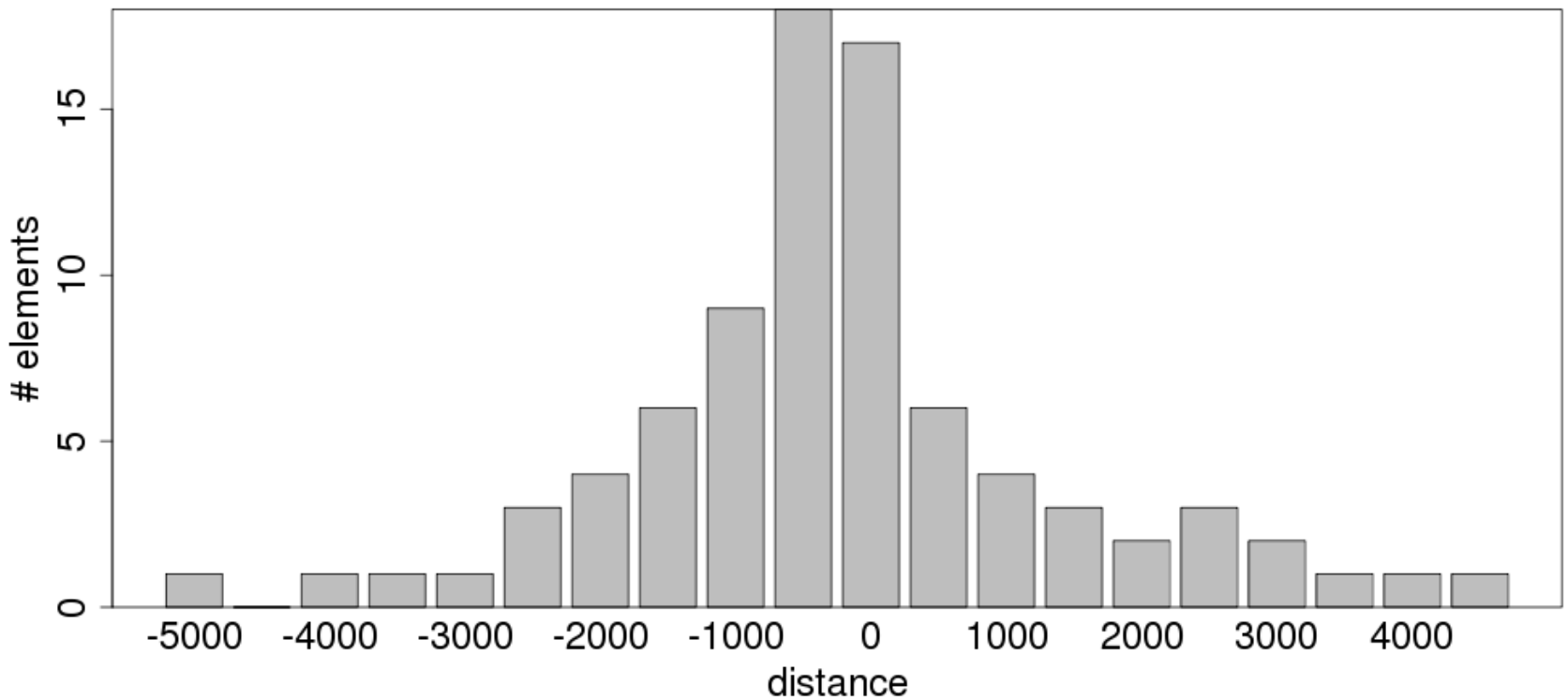
Step 4

Distance with the annotation

- tool: get distance
- file format 1: `GFF`
- file 1: previous file
- file format 2: `GTF`
- file 2: the clean annotation
- max. distance: 5000
- plot histogram: yes
- interval size: 500

Step 4

Figure



Step 5

Get new transcript unit in intergenic

- tool: Compare Overlapping Small Reference
- file format 1: `GFF`
- file 1: previous GFF file
- file format 2: `GTF`
- file 2: the clean annotation
- max. distance: 1000
- invert: yes

Step 6

Get the flanking elements of the new TU

- tool: get flanking
- file format 1: `GFF`
- file 1: previous file
- file format 2: `GTF`
- file 2: the clean annotation

Step 7

Convert into CSV (Excel-compatible)

- tool: Convert transcript file
- input file format: `GFF`
- file: previous file
- output format: `CSV`

→ You can now download the file and open it with Excel (or OpenOffice).

Step 7

Figure

	A	B	C	D	E	F	G	H	I	J	K	U	V	W	X	Y	Z
1	chromosome	start	end	strand	exons	ID	_region_flanking	_sense_flanking	distance__flanking	feature	flanking	flanking_ID	flanking_gene_id	flanking_p_id	flanking_segedit	flanking_tss_id	nbElements
2	1	14384501	14385500	+	None	region29034	upstream	colinear	11746586	transcript	AT1G08360.1	AT1G08360.1	AT1G08360	P13660	false	TSS13184	12
3	1	14451501	14452500	+	None	region29168	downstream	colinear	34843	transcript	AT1G38440.1	AT1G38440.1	AT1G38440	P9642	false	TSS5652	12
4	1	14659001	14660000	+	None	region29583	upstream	antisense	60811	transcript	AT1G39190.1	AT1G39190.1	AT1G39190	P13066	false	TSS37736	12
5	1	14719501	14720500	+	None	region29704	upstream	colinear	2204	transcript	AT1G39513.1	AT1G39513.1	AT1G39513	None	false	TSS12259	18
6	1	14730001	14731000	+	None	region29725	downstream	colinear	76800	transcript	AT1G39590.1	AT1G39590.1	AT1G39590	None	false	TSS35472	12
7	1	14960501	14961500	+	None	region30186	upstream	colinear	40913	transcript	AT1G40081.1	AT1G40081.1	AT1G40081	P21761	false	TSS8941	10
8	1	14967501	14968500	+	None	region30200	downstream	antisense	31115	transcript	AT1G40087.1	AT1G40087.1	AT1G40087	P5779	false	TSS8020	10
9	1	15042501	15043500	+	None	region30350	upstream	antisense	33839	transcript	AT1G40091.1	AT1G40091.1	AT1G40091	None	false	TSS37994	11
10	1	15094001	15095000	+	None	region30453	upstream	colinear	8108	transcript	AT1G40104.1	AT1G40104.1	AT1G40104	P10979	false	TSS15494	13
11	1	15136001	15137000	+	None	region30537	downstream	colinear	2933	transcript	AT1G40105.1	AT1G40105.1	AT1G40105	None	false	TSS12640	10
12	1	15168001	15169000	+	None	region30601	downstream	colinear	45280	transcript	AT1G40125.1	AT1G40125.1	AT1G40125	None	false	TSS19654	12
13	1	15261001	15262000	+	None	region30787	upstream	colinear	16965	transcript	AT1G40129.1	AT1G40129.1	AT1G40129	P4574	false	TSS5369	15
14	1	15261501	15262500	+	None	region30788	downstream	colinear	38085	transcript	AT1G40137.1	AT1G40137.1	AT1G40137	None	false	TSS19843	15
15	1	15347001	15348000	+	None	region30959	upstream	antisense	30087	transcript	AT1G40143.1	AT1G40143.1	AT1G40143	None	false	TSS15716	13
16	1	15361001	15362000	+	None	region30987	downstream	colinear	60187	transcript	AT1G40076.1	AT1G40076.1	AT1G40076	P5690	false	TSS34345	13
17	1	16542001	16543000	+	None	region33349	upstream	colinear	357478	transcript	AT1G43040.1	AT1G43040.1	AT1G43040	P19395	false	TSS13039	13
18	1	16542501	16543500	+	None	region33350	downstream	antisense	8644774	transcript	AT1G67280.1	AT1G67280.1	AT1G67280	P4929	false	TSS22652	15
19	3	13587001	13588000	+	None	region86125	upstream	colinear	5726670	transcript	AT3G22237.1	AT3G22237.1	AT3G22237	P27351	false	TSS10291	13
20	3	13592001	13593000	+	None	region86135	downstream	antisense	8537483	transcript	AT3G59910.1	AT3G59910.1	AT3G59910	P19161	false	TSS14193	11
21	4	3838501	3839500	+	None	region194045	upstream	antisense	2664180	transcript	AT4G02660.1	AT4G02660.1	AT4G02660	P3628	false	TSS8619	12
22	4	3839001	3840000	+	None	region194046	downstream	colinear	41990	transcript	AT4G06670.1	AT4G06670.1	AT4G06670	None	false	TSS17002	12
23	4	3952001	3953000	+	None	region194272	upstream	antisense	11204	transcript	AT4G06701.3	AT4G06701.3	AT4G06701	P28631	false	TSS11072	17
24	4	3953001	3954000	+	None	region194274	downstream	antisense	3356416	transcript	AT4G12310.1	AT4G12310.1	AT4G12310	P27764	false	TSS37402	16
25	5	11321501	11322500	+	None	region156044	upstream	colinear	10702904	transcript	AT5G02740.2	AT5G02740.2	AT5G02740	P31272	false	TSS16836	11
26	5	11331001	11332000	+	None	region156063	downstream	antisense	8043	transcript	AT5G30102.1	AT5G30102.1	AT5G30102	P13179	false	TSS25628	10
27	5	11686501	11687500	+	None	region156774	upstream	antisense	52045	transcript	AT5G29708.1	AT5G29708.1	AT5G29708	P25890	false	TSS22810	14
28	5	11687001	11688000	+	None	region156775	downstream	colinear	13226	transcript	AT5G31770.1	AT5G31770.1	AT5G31770	None	false	TSS16897	13
29	5	11733001	11734000	+	None	region156867	upstream	antisense	11437	transcript	AT5G31804.1	AT5G31804.1	AT5G31804	None	false	TSS6714	51
30	5	11736001	11737000	+	None	region156873	downstream	antisense	30478	transcript	AT5G31927.1	AT5G31927.1	AT5G31927	None	false	TSS31680	49
31	5	11811001	11812000	+	None	region157023	downstream	antisense	1521	transcript	AT5G32107.1	AT5G32107.1	AT5G32107	None	false	TSS349	10
32	5	11840001	11841000	+	None	region157081	upstream	colinear	1268	transcript	AT5G32197.1	AT5G32197.1	AT5G32197	None	false	TSS4534	12
33	5	11849501	11850500	+	None	region157100	upstream	colinear	1151	transcript	AT5G32228.1	AT5G32228.1	AT5G32228	None	false	TSS13570	10
34	5	11850001	11851000	+	None	region157101	downstream	antisense	58710	transcript	AT5G32254.1	AT5G32254.1	AT5G32254	None	false	TSS18369	15
35	5	11989001	11990000	+	None	region157379	upstream	colinear	49093	transcript	AT5G32293.1	AT5G32293.1	AT5G32293	None	false	TSS32265	12
36	5	11989501	11990500	+	None	region157380	downstream	colinear	2017326	transcript	AT5G35840.1	AT5G35840.1	AT5G35840	P15521	false	TSS14542	16

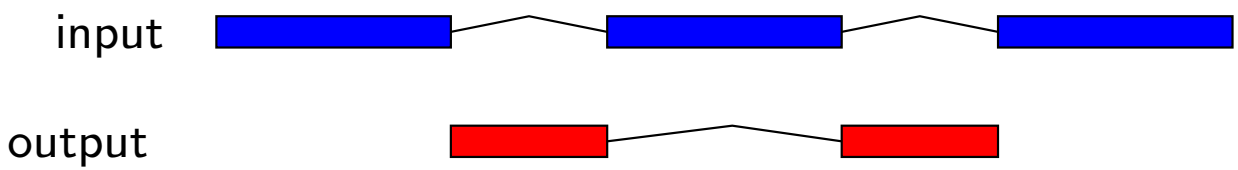
Outline

Step 1

Get introns coordinates

- tool: get introns
- input format: `GTF`
- file: the clean annotation

Get introns

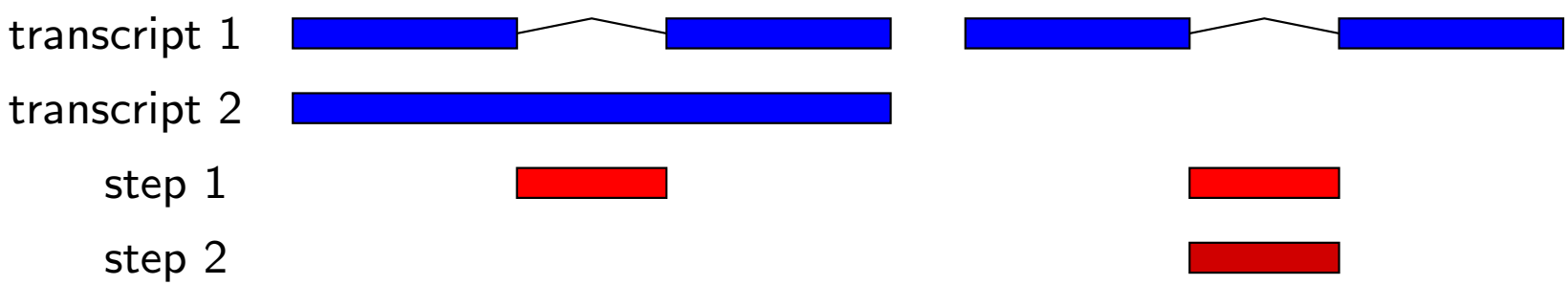


Step 2

Remove introns which overlap with exons

- tool: Compare Overlapping Small Query
- file format 1: `GFF`
- file 1: previous file
- file format 2: `GTF`
- file 2: the clean annotation
- invert: true

Problems with introns



Step 3

Get the reads inside the introns

- tool: Compare Overlapping Small Reference
- file format 1: `SAM`
- file 1: `sample 1_2.sam`
- file format 2: `GFF`
- file 2: previous file

Step 4

Clusterize the reads inside introns

- tool: Clusterize
- file format 1: `GFF`
- file 1: the previous file

Step 5

Select regions with $\geq$ 10 reads

- tool: select by tag
- file format 1: `GFF`
- file 1: the previous file
- minimum: yes, then `10`
- default: yes, then `1`

Outline

Outline of the analysis

Aim

- Display the average number of reads along each expressed gene.
- Display the average number of reads near the TSSs and the TESs.
- Compute the average number of reads for each expressed gene.

Attention

These analyses usually take time to perform.

Step 1

Select expressed genes

- tool: Compare Overlapping Small Query
- file format 1: `GTF`
- file 1: the clean annotation
- file format 2: `SAM`
- file 2: `sample 1_2.sam`

Step 2

Select genes with at least 10 reads

- tool: select by tag
- file format: `GFF`
- file: previous file
- tag: `nbOverlaps`
- min: `10`

Step 3

Convert SAM into WIG

- tool: Convert transcript file
- input file format: `SAM`
- input file: `sample 1_2.sam`
- output file format: `WIG`

Step 4

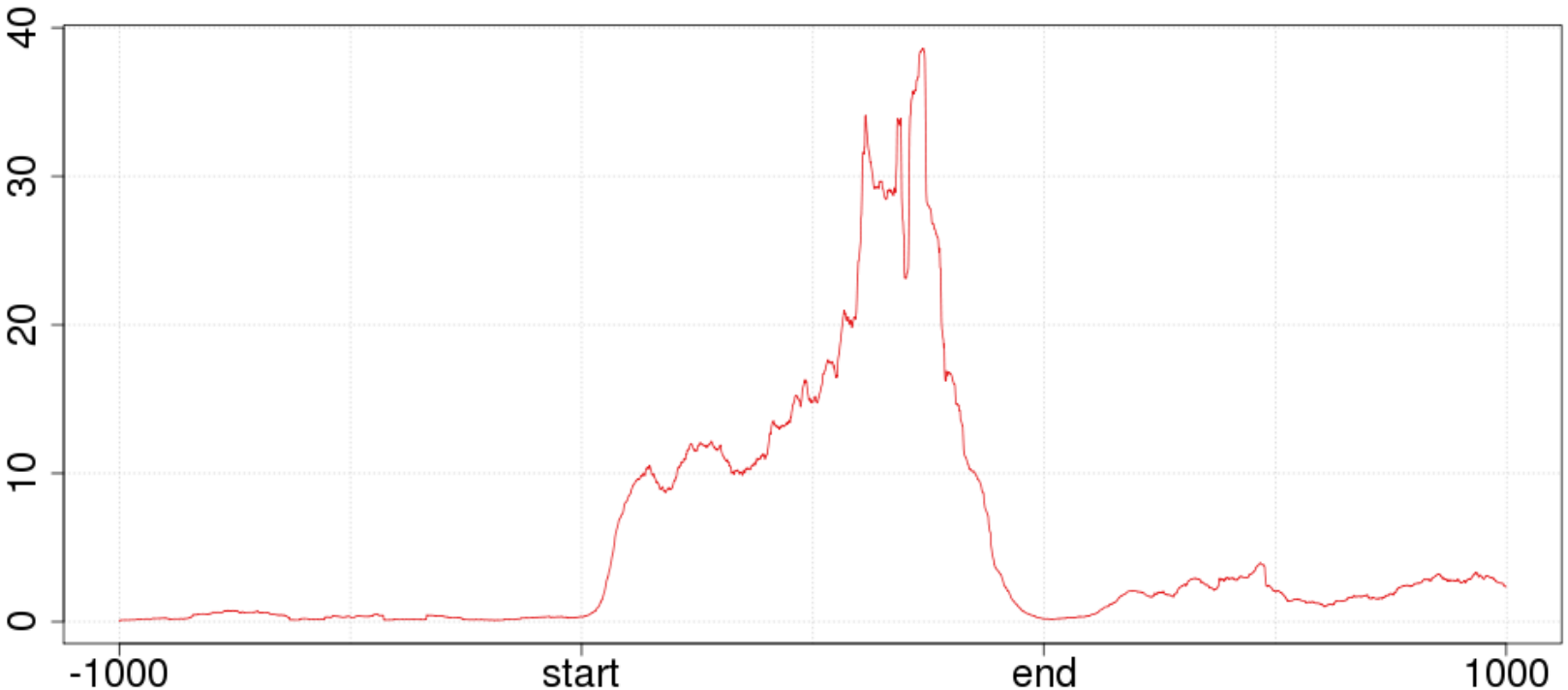
Get WIG profile

- tool: get wig profile
- WIG file: the only WIG file...
- file format: `GFF`
- file: file obtained in step 2
- number of points: 1000
- distance: 1000

→ You have the average number of reads along each expressed gene.

Step 4

Figure



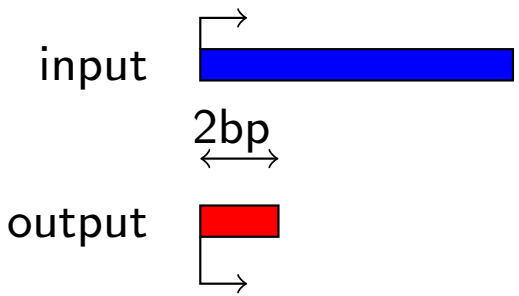
Step 5

Get the TSSs

- tool: modify genomic coordinates
- input file format: `GFF`
- input file: file generated in step 2
- restrict to start: yes, then 2

→ We chose the first 2 nucleotides to set the TSS. It should have been the first nucleotide, but it leads to a bug in S–MART.

Upstream regions

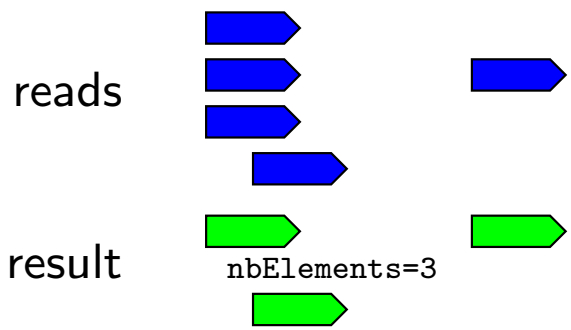


Step 6

Collapse the TSSs

- tool: collapse reads
- input file format: `GFF`
- input file: previous file

Collapsing



Step 7

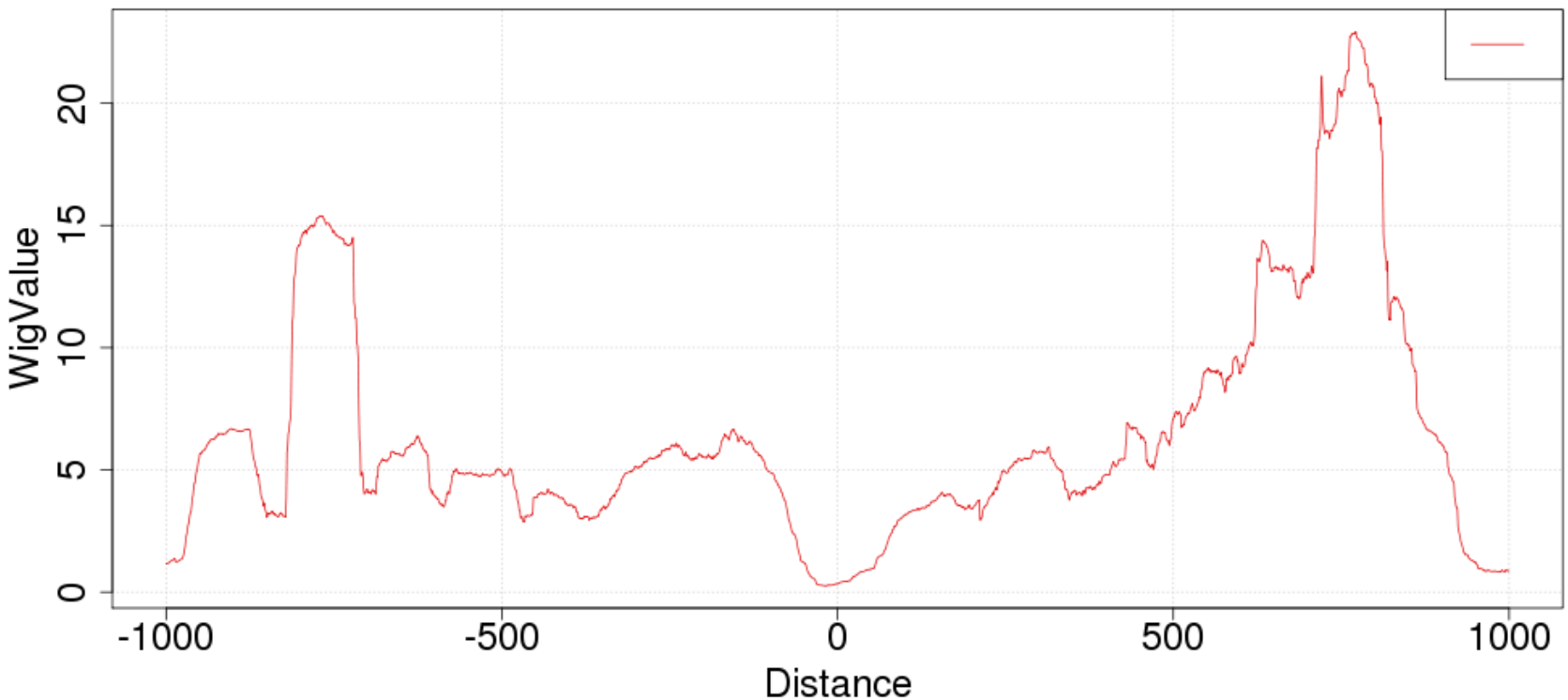
Plot the WIG distribution around TSSs

- tool: get wig distance
- WIG file: the only WIG file...
- file format: `GFF`
- file: the previous file
- distance: `1000`

→ You have the average number of reads around the TSSs of the genes. `0` is the TSS itself; positive values are inside the transcript; negative values are upstream.

Step 7

Figure



Step 8

Get the TESs

- tool: modify genomic coordinates
- input file format: `GFF`
- input file: file obtained in step 2
- restrict to end: yes, then 2

Collapse the TESs

Step 9

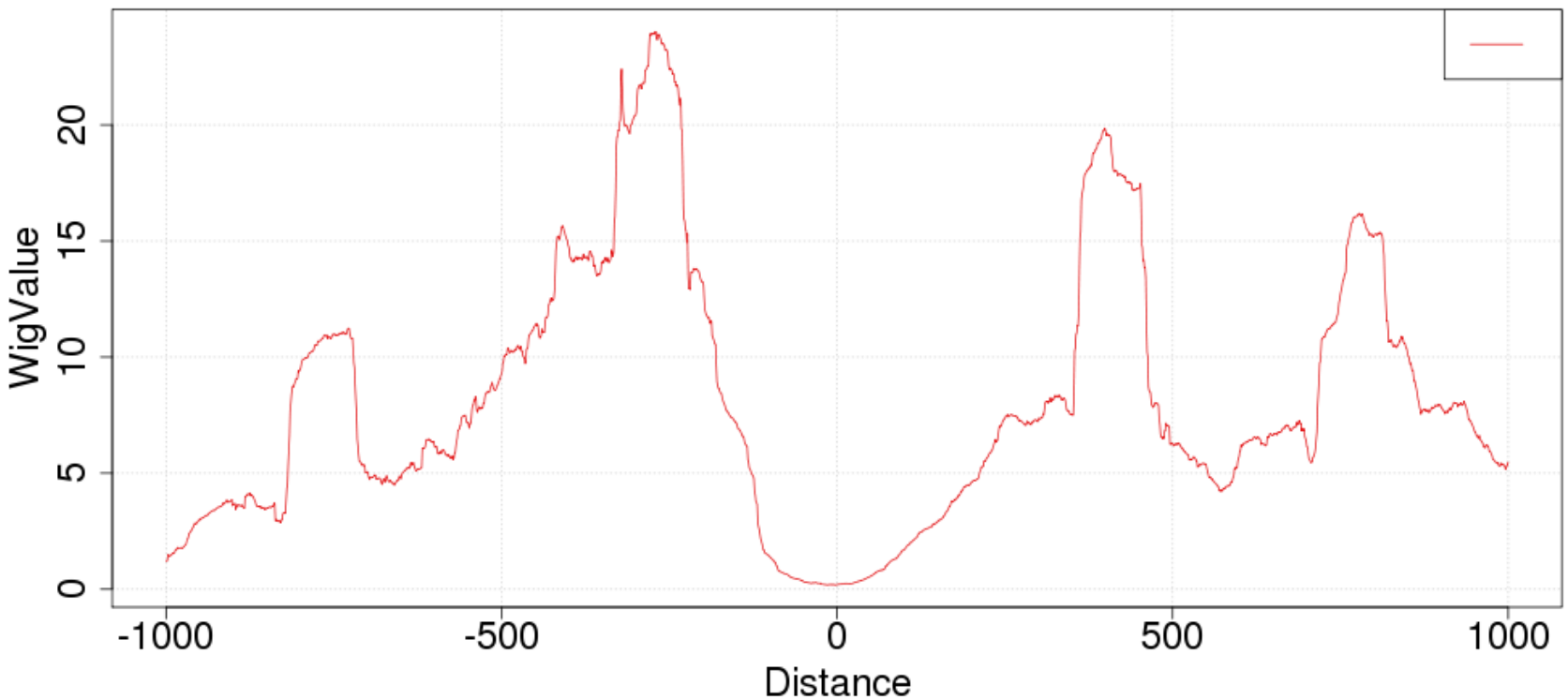
Plot the WIG distribution around TESs

- tool: get wig distance
- WIG file: the only WIG file...
- file format: `GFF`
- file: the previous file
- distance: `1000`

→ You have the average number of reads around the TESs of the genes. `0` is the TES itself; negative values are inside the transcript; positive values are upstream.

Step 9

Figure



Step 10

Compute the average coverage

- tool: get wig data
- WIG file: the only WIG file. . .
- file: the file obtained in step 2
- tag: `coverage`

→ You have the average number of reads covering each nucleotide of a gene in its `coverage` tag.

Step 11

Select genes which average coverage ≥ 10

- tool: select by tag
- file format: `GFF`
- file: previous file
- tag: `coverage`
- min: `10`

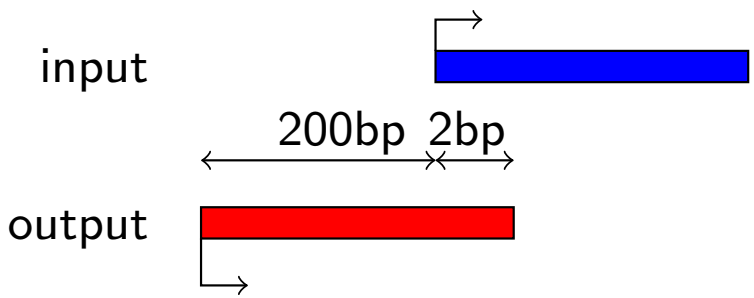
Outline

Step 1

Get upstream regions

- tool: modify genomic coordinates
- input file format: `GTF`
- input file: the clean annotation
- start: 2
- 5': 200

Upstream regions



Step 2

Get reads inside upstream regions

- tool: Compare Overlapping Small Reference
- file format 1: `SAM`
- file 1: `sample 1_2.sam`
- file format 2: `GFF`
- file 2: the previous file

Step 3

Discard reads which overlap with genes

- tool: Compare Overlapping Small Reference
- file format 1: `GFF`
- file 1: previous file
- file format 2: `GTF`
- file 2: the clean annotation
- invert: yes

Step 4

Clusterize the upstream reads

- distance: 100

Select the clusters with ≥ 10 reads

Get the name of the transcripts

- tool: Compare Overlapping Small Reference
- file format 1: `GFF`
- file 1: the upstream regions
- file format 2: `GFF`
- file 2: the previous file

$\rightarrow$ You now have the genes such that there is evidence of transcription in the upstream region.

Outline

Step 1

Select clusters with $\geq$ 10 reads

Select clusters with $\geq$ 10 reads of size $\geq$ 1kb

- tool: restrict from size
- file format: `GFF`
- file: the previous file
- minimum: yes, the 1000

Select clusters with $\geq$ 10 reads of size $\geq$ 1kb outside genes

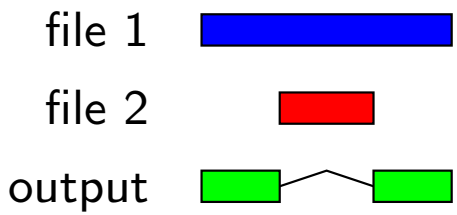
$\rightarrow$ There is no such clusters.

Step 2

Get the difference between the clusters and the annotation

- tool: get Difference
- file format 1: `GFF`
- file format 2: `GTF`
- file 1: the previous non-empty file
- file 2: the clean annotation

Difference



Step 3

Get the sizes of the differences

- tool: get sizes
- file format 1: `GFF`
- file 1: the previous file

Step 3

Figure