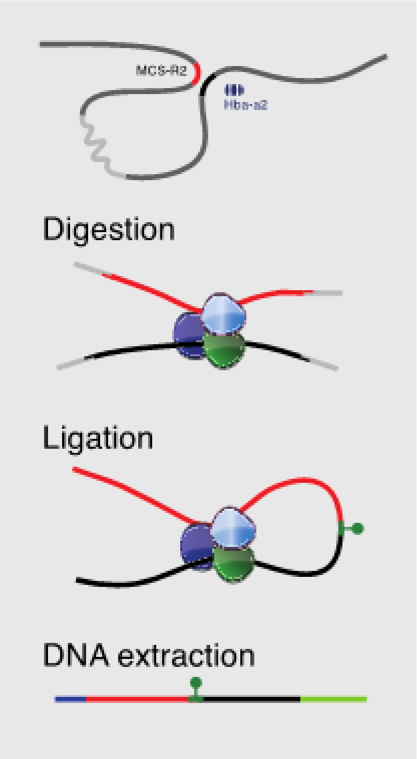
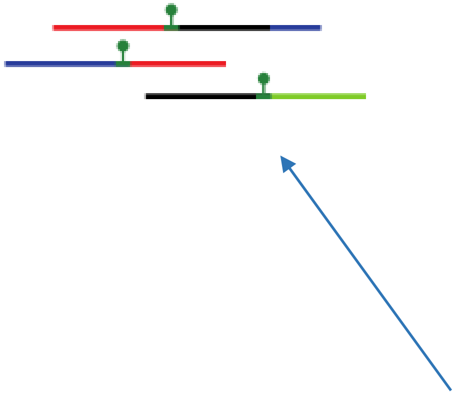
r3CSeq and FourCSeq

What can they add to our CaptureC analysis ?

Enhancer Journal Club - Jelena 14/Oct/2015

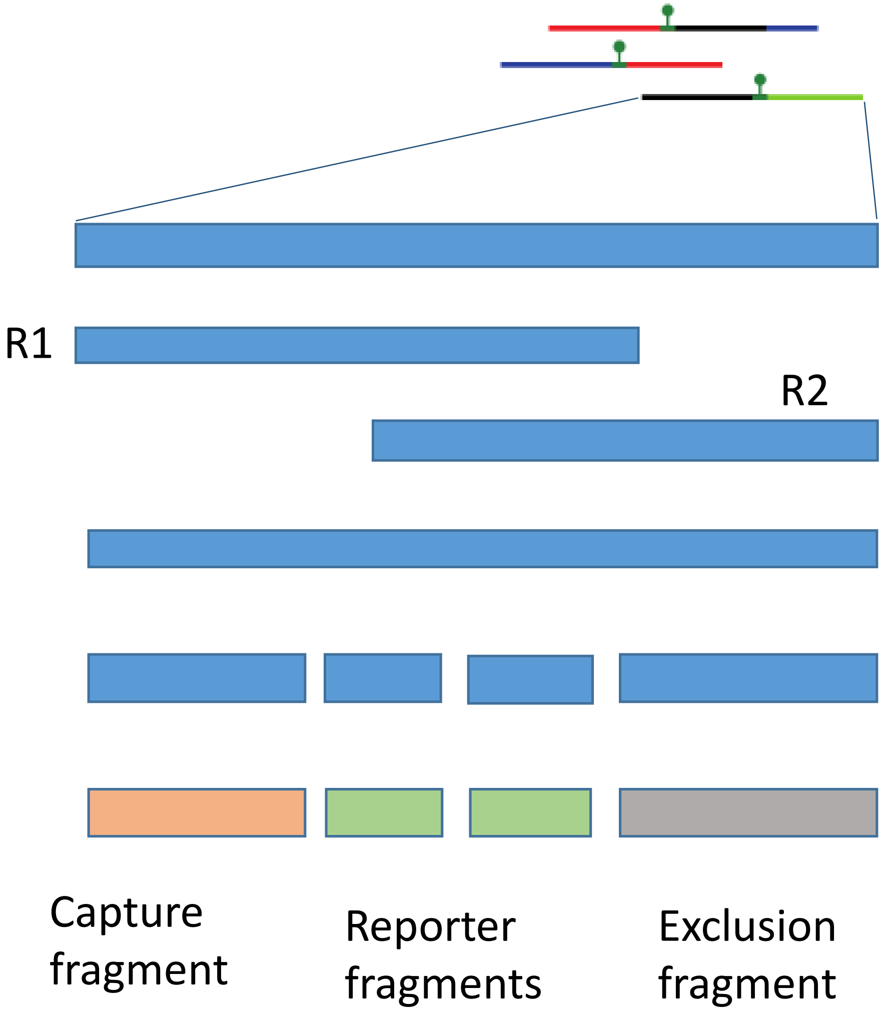
CaptureC – from the viewpoint of a bioinformatician !





This is what I see !

What is a “read” and what is a “fragment”



Sonication fragment. Captured and sent for sequencing.

Read1 and Read2 – paired end sequencing results from the fragment

Flash-merged “full read” for the sequence

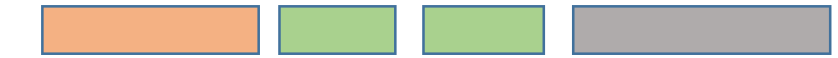
In silico RE-digested read – showing different **captured fragments** within the read

In silico RE-digested read – showing different identities of the captured fragments

What is a “capture” and what is a “reporter” fragment

Analysis read (contains all these fragments)

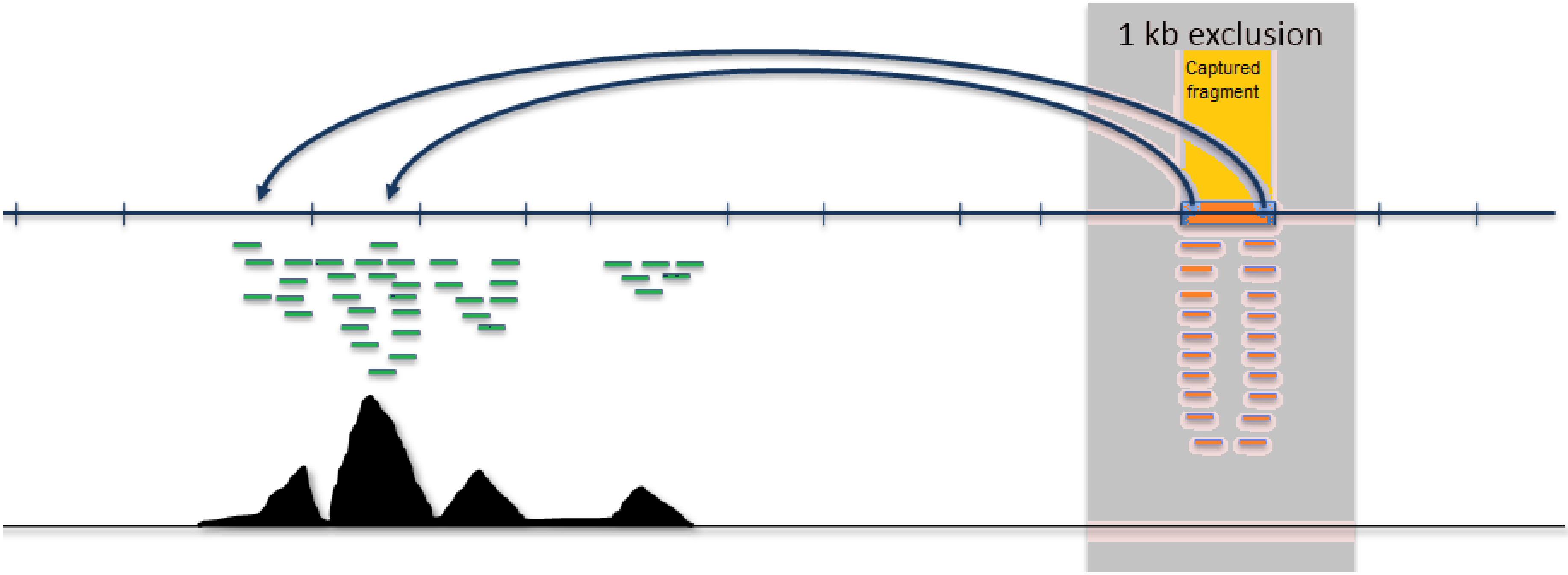
In silico RE-digested read – showing different identities of the captured fragments



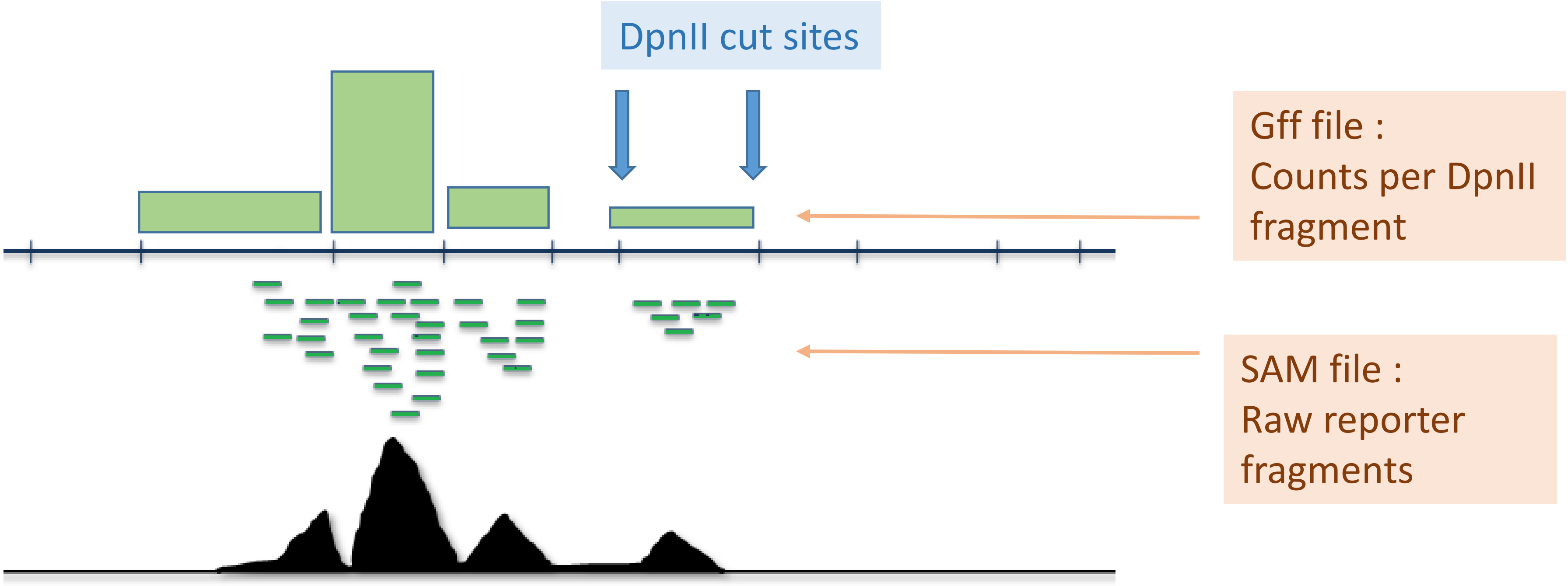
Capture fragment

Reporter fragments

Exclusion fragment



What is a SAM file and GFF file of reporter fragments ?



DpnII cut sites

Gff file :
Counts per DpnII
fragment

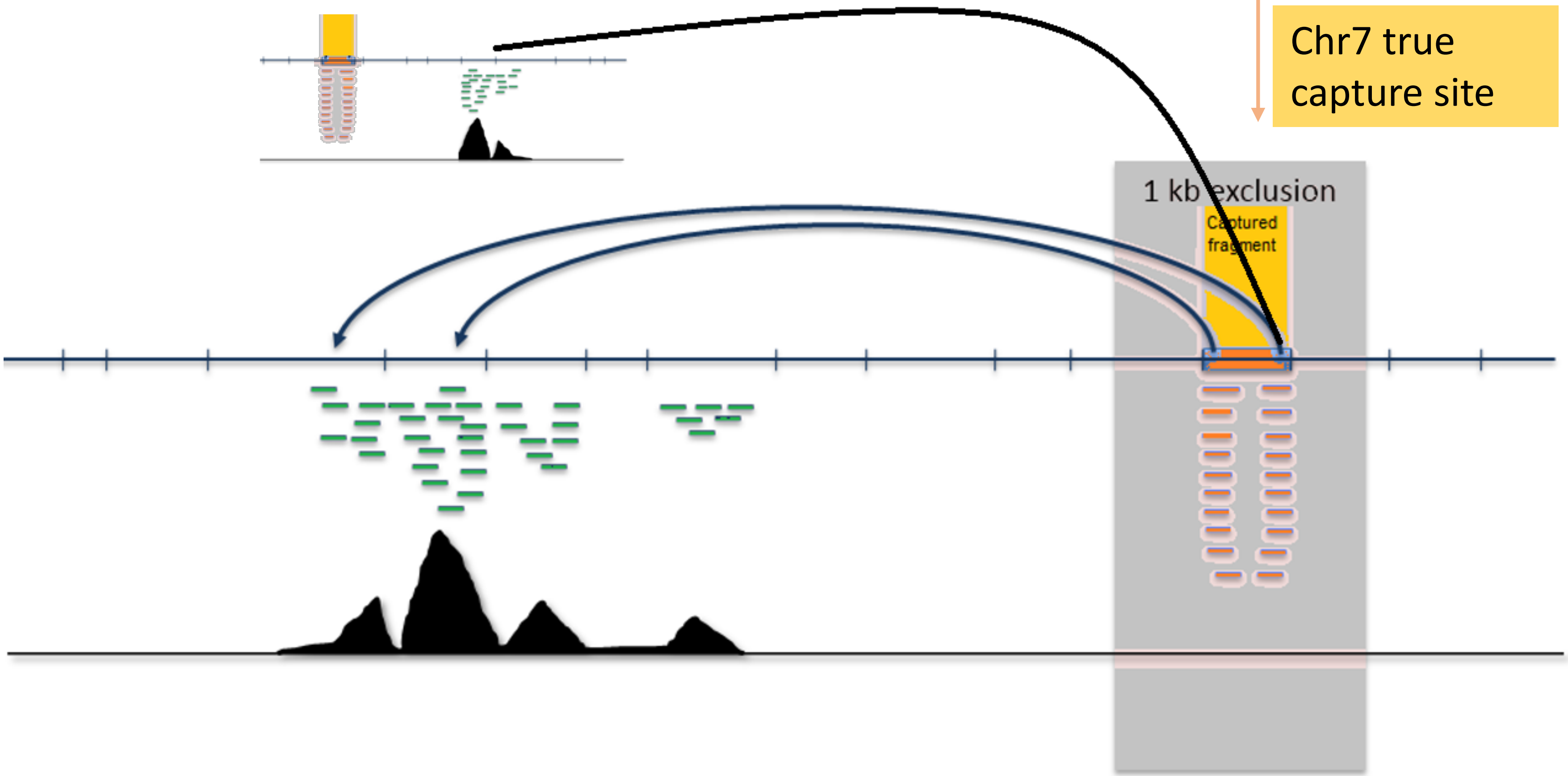
SAM file :
Raw reporter
fragments

Why do we need BLAT filter ? – homology regions ..

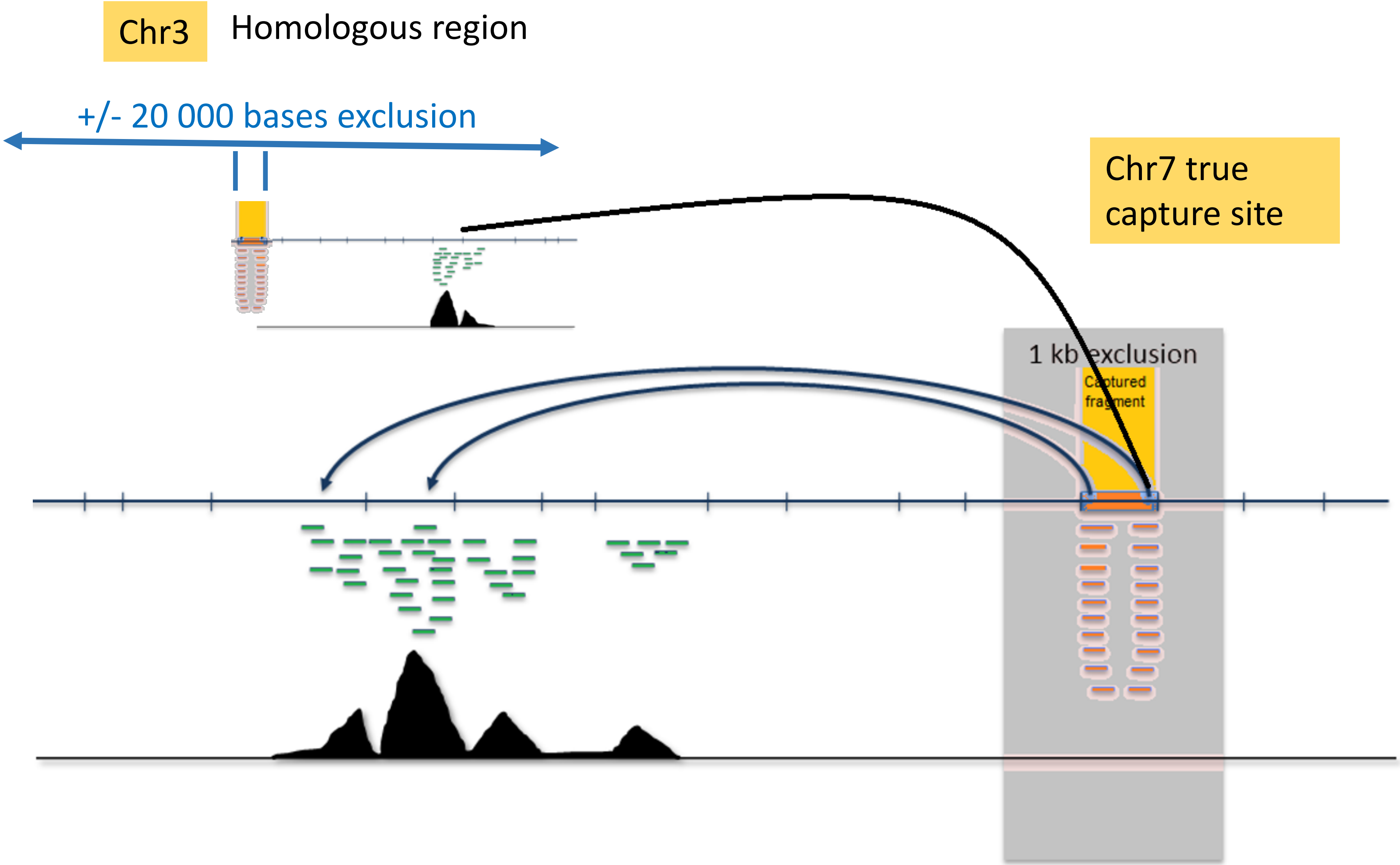
Chr3

Homologous region, which can map (bioinformatically) here :

Chr7 true capture site

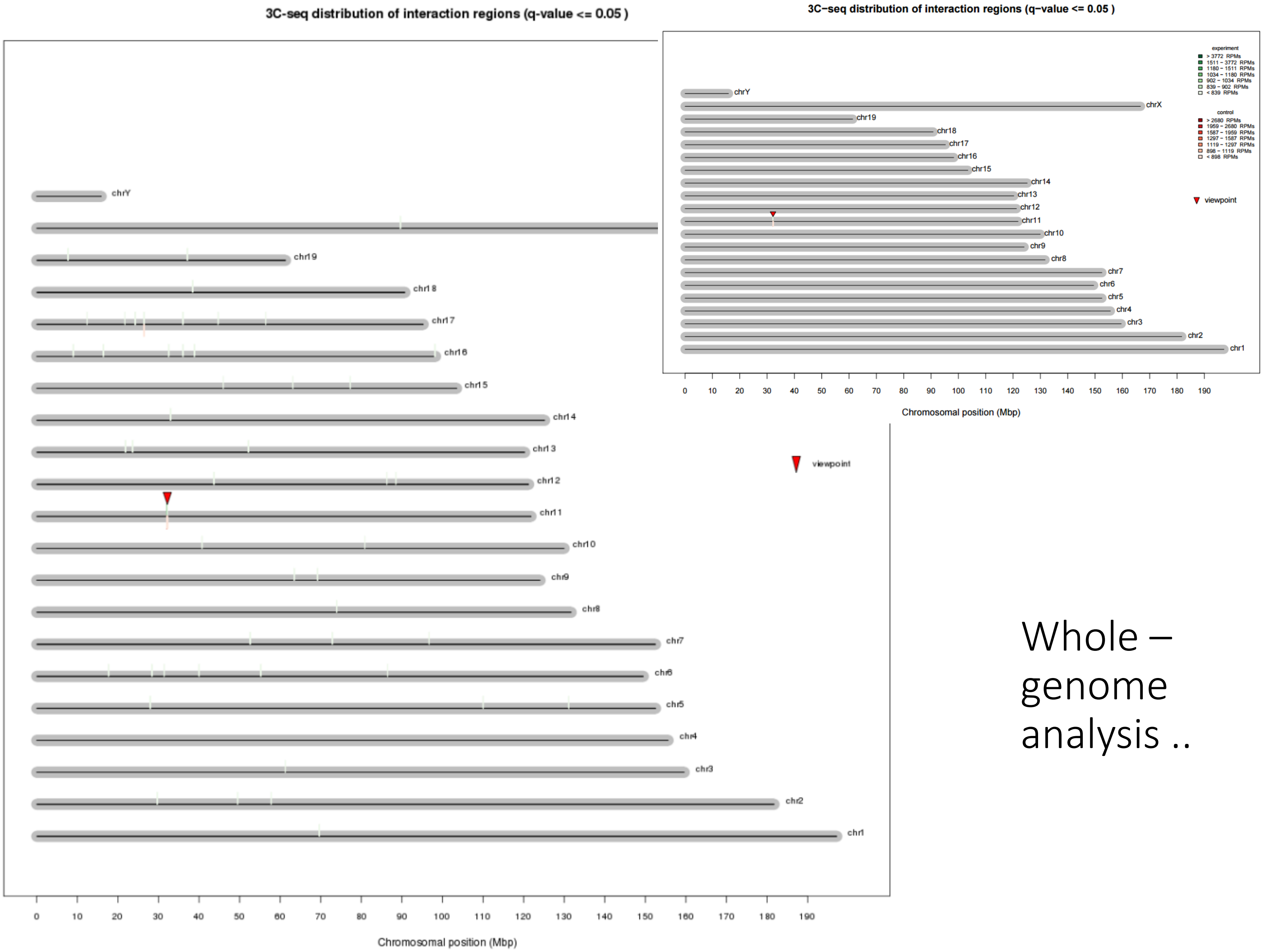


Why do we need BLAT filter ? – homology regions ..



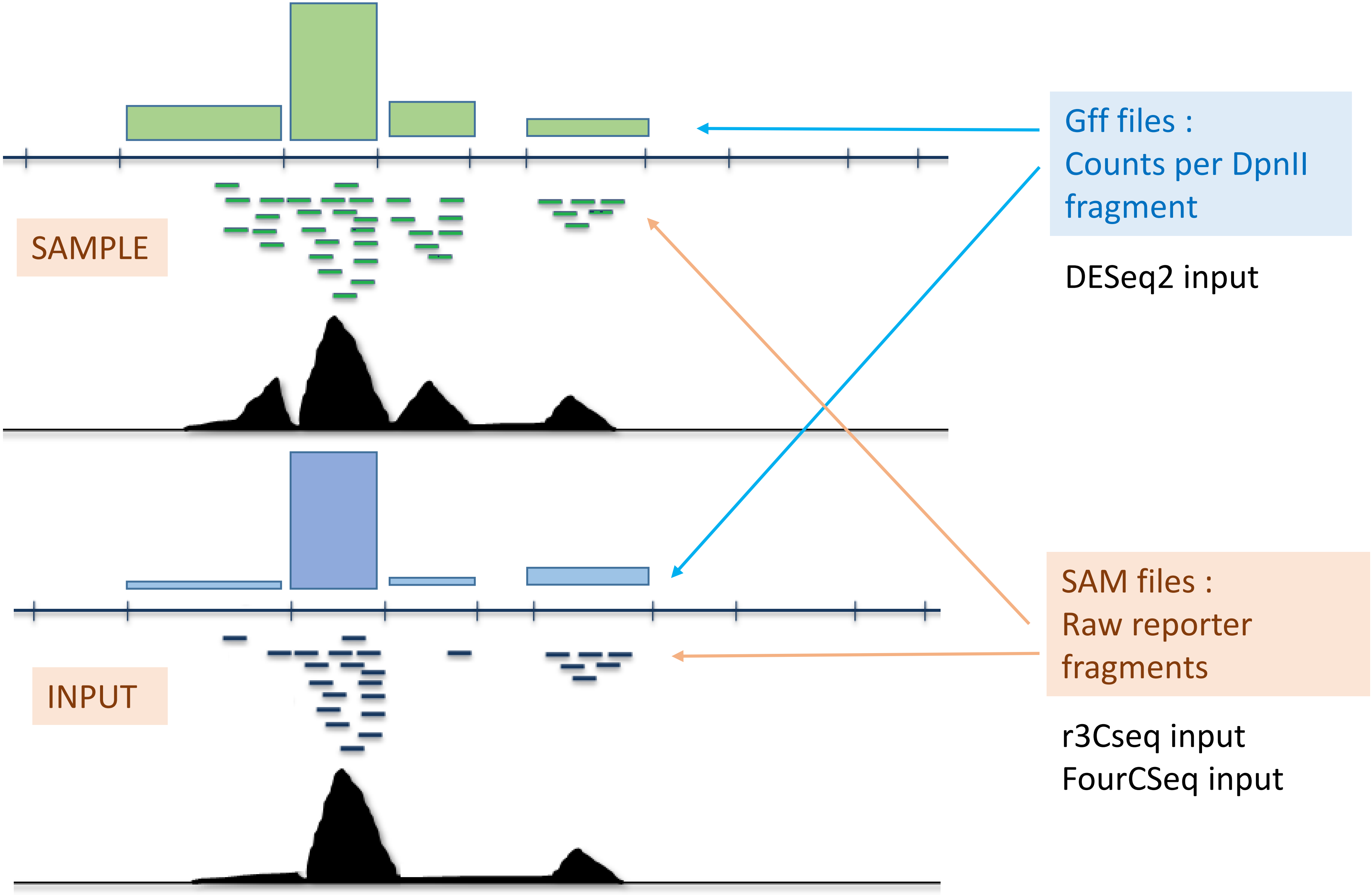
Whole-genome analysis ..

For the time being :		Soon :
Using James’ scripts to **step by step filter** the files and generate the GFF pileups for DESeq2 differential analysis.	GFF	James’ scripts for GFF filtering as integral step of CaptureC pipeline
Using Jelena’s scripts to **step by step filter** SAM files as input to r3CSeq and FourCSeq.	SAM	Jelena’s scripts for SAM filtering as integral step of CaptureC pipeline



Whole – genome analysis ..

Comparing SAMPLE and INPUT?



SAMPLE

INPUT

Gff files :
Counts per DpnII
fragment

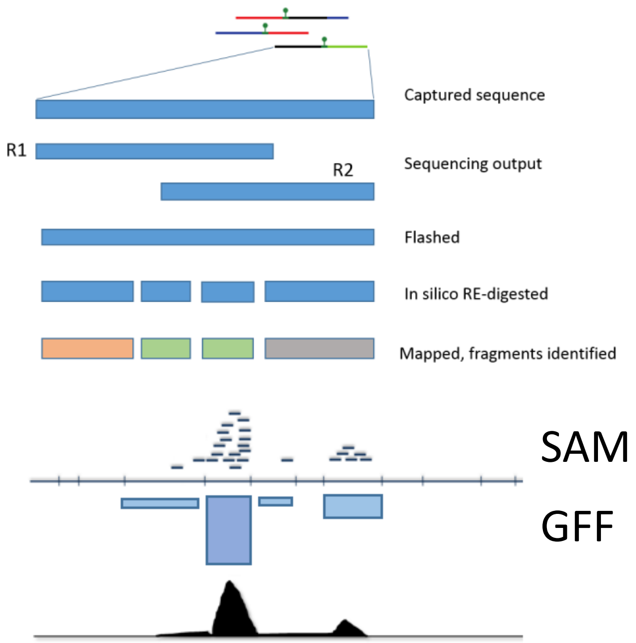
DESeq2 input

SAM files :
Raw reporter
fragments

r3Cseq input
FourCSeq input

CaptureC + DESeq2

1) Normal CaptureC analysis :



Filtering steps within CCanalyser :

- save only reads which contain a capture (something mapped between the DpnII cut sites both sides of the capture oligo)

- fragment identification (capture, exclusion, reporter)

- duplicate read removal

- duplicate reporter removal (with same read)

Output files of CCanalyser :

- sam of reporter reads for each oligo (minus exlusions, minus capture fragments)

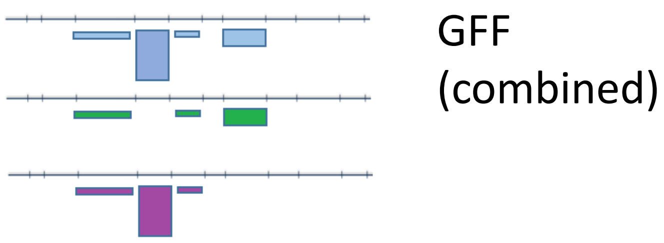
- gff of reporter reads for each oligo (counts of reporter fragments per DpnII fragment)

2) Filtering :

- filter each oligo results (reporter fragments) for presence of Blat and Ploidy regions

3) Combining :

- combine all replicates (for sample and control) to a single file with counts per RE fragment



Differential analysis ..

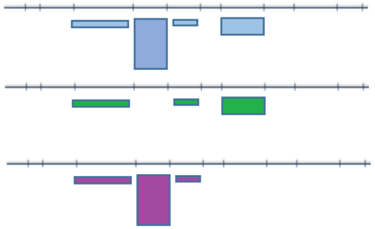
Now, how should I do this .. ?

1) Normalisation factor to count for the different sequencing depths between replicates and samples (RPM value or similar)

2) Use the within-sample mean to estimate the "true read signal" in each DpnII fragment, and the within-sample variation to estimate the error.
→ P-values here tell the regions which have RELIABLY MORE signal than the baseline

3) Use the between-samples (normalised) log-fold-change (LFC) to tell differential expressed regions.

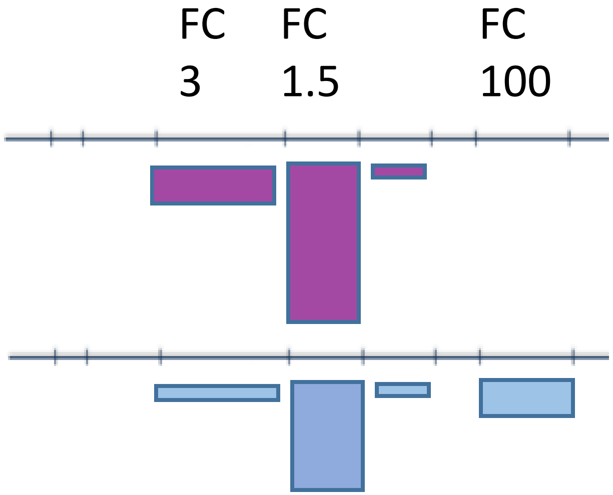
4) Generate adjusted P-values to tell which regions are RELIABLY significantly different.



GFF (combined)

Why is that not enough ?

- We tend to get large fold changes for small effect sizes (if we have relatively low signal, we get a lot of "noise" large fold changes)



Differential analysis with DESeq2 ..

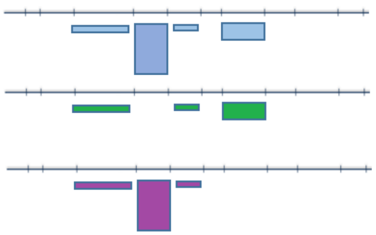
Now, how does DESeq2 fix this .. ?

1) Normalisation factor to count for the different sequencing depths between samples (RPM value or similar)

2) Use the within-sample mean to estimate the "true value" and the within-sample variation to estimate the error.
→ P-values here tell the regions which have RELIABLY MORE signal than the baseline

3) Use the between-samples (normalised) log-fold-change (LFC) to tell differential expressed regions.

4) Generate adjusted P-values to tell which regions are RELIABLY significantly different.



GFF (combined)

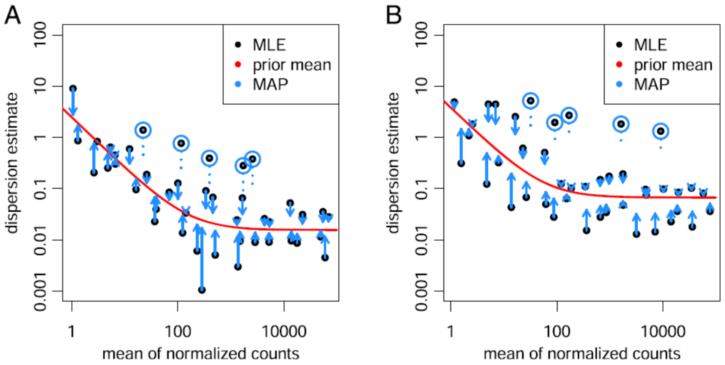


Figure 1 Shrinkage estimation of dispersion.
Plot of dispersion estimates over the average expression strengtl

Love *et al. Genome Biology* (2014) 15:550

Within-sample variation higher than 2std

- These are flagged as outliers, and the whole DpnII fragment is exluded from the analysis

- If using more than 6 replicates, the outliers are shifted towards the DpnII fragment calculated mean, not towards the general fit (the red line).

DESeq2 modifications :

Shift each DpnII fragment "errors" towards the average (normalised) count of that DpnII fragment.

Handles the large dynamic range better (does not get overwhelmed by signals of 10 reads vs 1000 reads).

Differential analysis with DESeq2 ..

Now, how does DESeq2 fix this .. ?

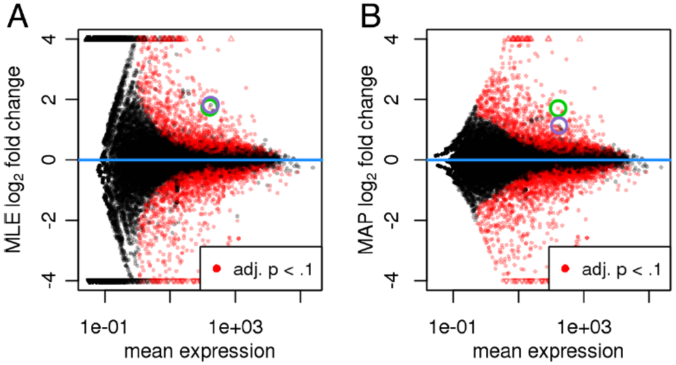
1) Normalisation factor to count for the different sequencing depths between samples (RPM value or similar)

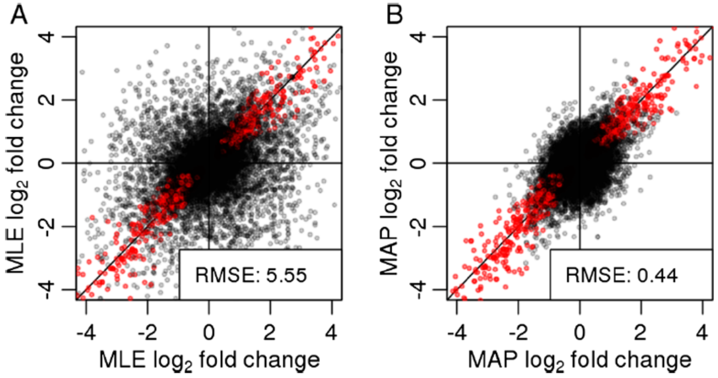
2) Use the within-sample mean to estimate the “true value” and the within-sample variation to estimate the error.
→ P-values here tell the regions which have MORE signal than the baseline

3) Use the between-samples (normalised) log-fold-change (LFC) to tell differential expressed regions.

4) Generate adjusted P-values to tell which regions are significantly different.

Love *et al. Genome Biology* (2014) 15:550





DESeq2 modifications :

Using the shifted values to count the log fold changes

The log fold plots look “easier to interpret”
(does not spread the points of low counts “artificially wide” in the plot).

Less regions get interpreted as “significantly different” ONLY because of low read counts.

Higher reproducibility of log fold changes.

Differential analysis with DESeq2 ..

Now, how does DESeq2 fix this .. ?

1) Normalisation factor to count for the different sequencing depths between samples (RPM value or similar)

2) Use the within-sample mean to estimate the "true value" and the within-sample variation to estimate the error.

→ P-values here tell the regions which have MORE signal than the baseline

3) Use the between-samples (normalised) log-fold-change (LFC) to tell differential expressed regions.

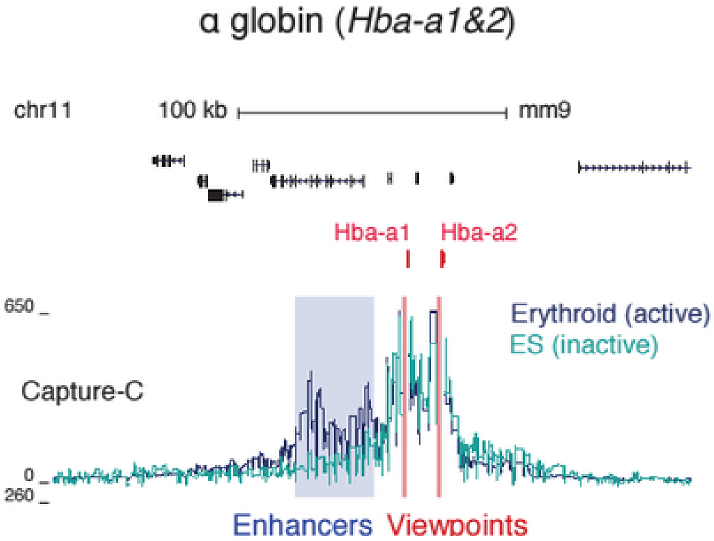
4) Generate adjusted P-values to tell which regions are significantly different.

DESeq2 modifications :

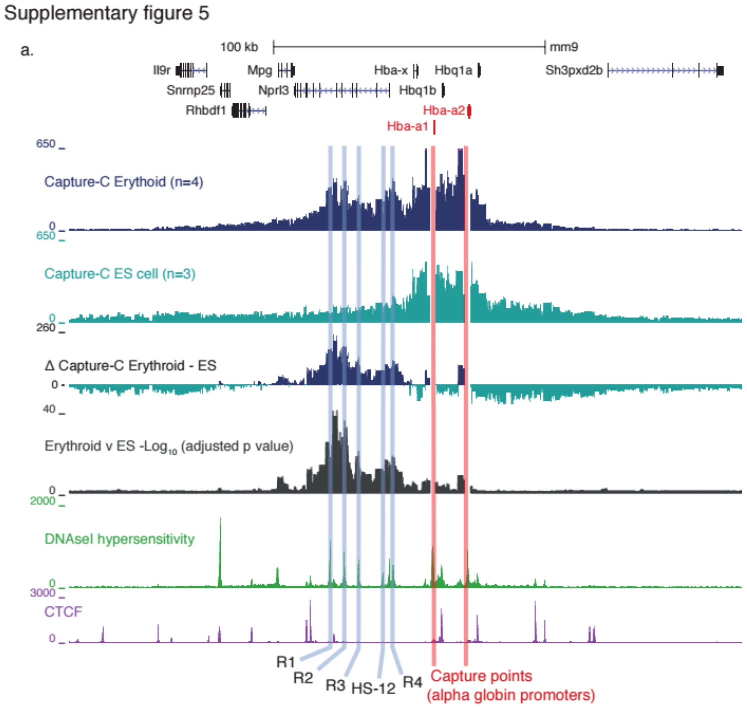
Count "more intuitive" P-values :
Take as null hypothesis

"log2-fold change BETWEEN 1 and -1"

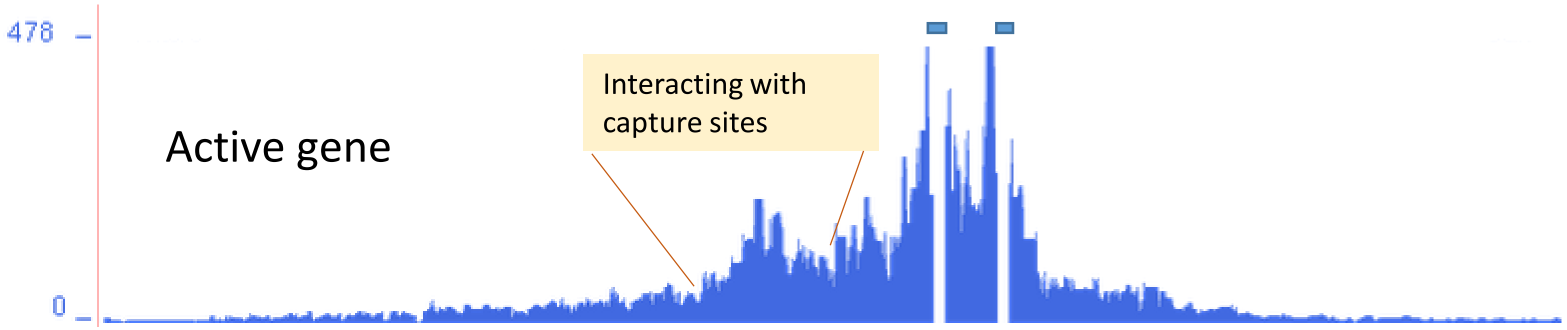
(instead of the traditional "log2fold change EXACTLY 0")



Differential analysis with DESeq2 ..

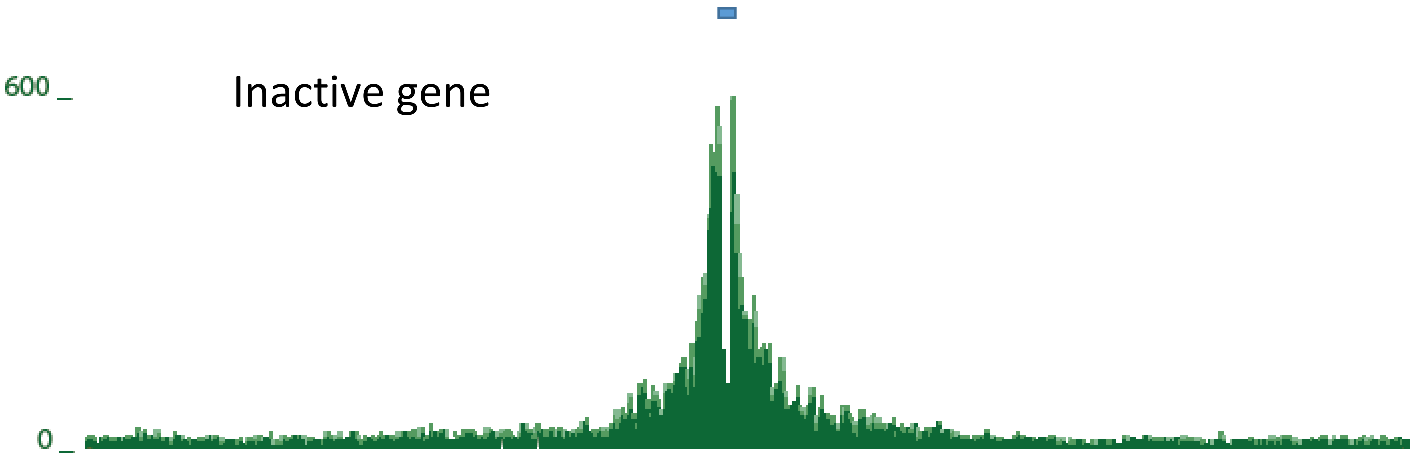


Distance from capture site ?



Active gene

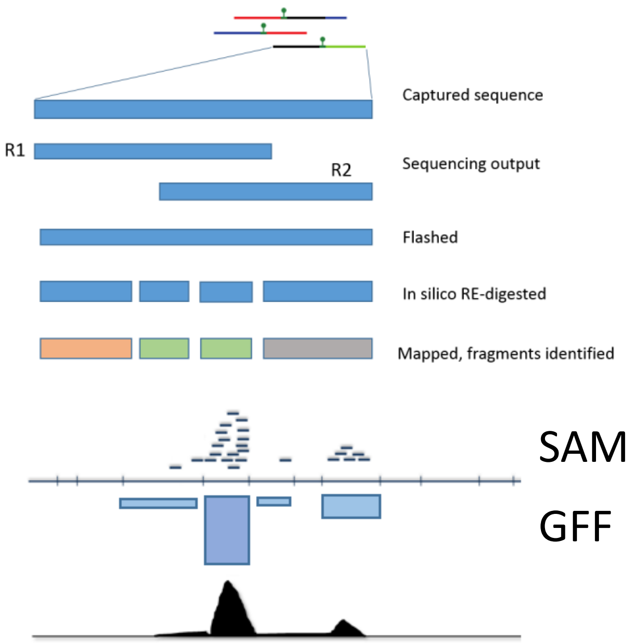
Interacting with capture sites



Inactive gene

CaptureC + FourCSeq

1) Normal CaptureC analysis :



Filtering steps within CCanalyser :

- save only reads which contain a capture (something mapped between the DpnII cut sites both sides of the capture oligo)

- fragment identification (capture, exclusion, reporter)

- duplicate read removal

- duplicate reporter removal (last fragment in R1/R2 "duplications")

Output files of Ccanalyser :

- sam of reporter reads for each oligo (minus exlusions, minus capture fragments)

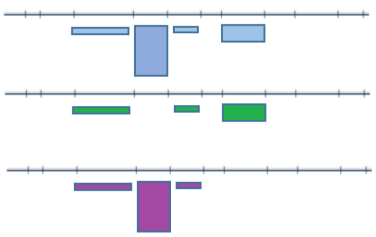
- gff of reporter reads for each oligo (counts of reporter fragments per DpnII fragment)

2) Filtering :

- filter each oligo results (reporter fragments) for presence of Blat and Ploidy regions
→ output as SAM file !

3) Read into FourCSeq :

- FourCSeq reads the SAM, and maps the reads back to DpnII fragments.



DpnII signal (reconstructed in R)

Differential analysis with FourCSeq (uses DESeq2 ..)

DESeq2-powered distance correction

1) Normalisation factor to count for the different sequencing depths between samples (RPM value or similar)

2) Use the within-sample mean to estimate the "true value" and the within-sample variation to estimate the error.
→ Z-values (how many STDs the value differs from average signal of the DpnII fragment) tell the regions which have MORE signal than the baseline

3) Use the between-samples (normalised) log-fold-change (LFC) as the starting point of the FITTING PROCEDURE

4) Fit monotone function (both sides of the capture DpnII fragment) – and like that correct for "more signal close to the capture site".

5) Repeat step 2 with the scaled values :
→ Z-values (how many STDs the value differs from average signal of the DpnII fragment) tell the regions which have MORE signal than the baseline

6) Repeat step 3 with the scaled values :
Use the between-samples (normalised) log-fold-change (LFC) to tell differential expressed regions.

7) Combine results of steps 5) and 6) : these are the final results of the tool.

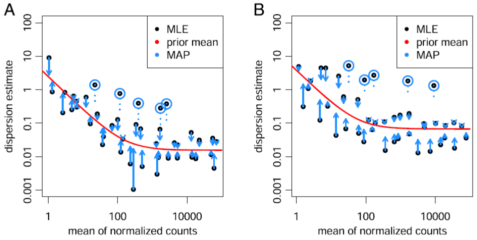
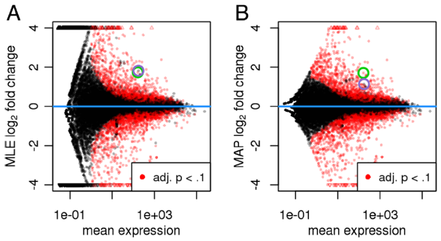


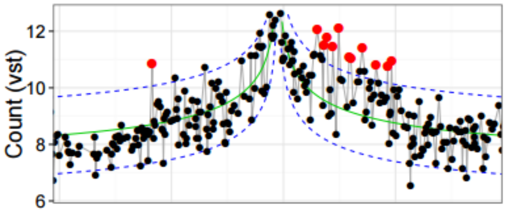
Figure 1 **Shrinkage estimation of dispersion.**
Plot of dispersion estimates over the average expression strengtl

2

Love *et al. Genome Biology* (2014) 15:550

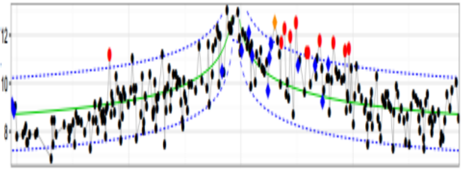


3



4 (green line)

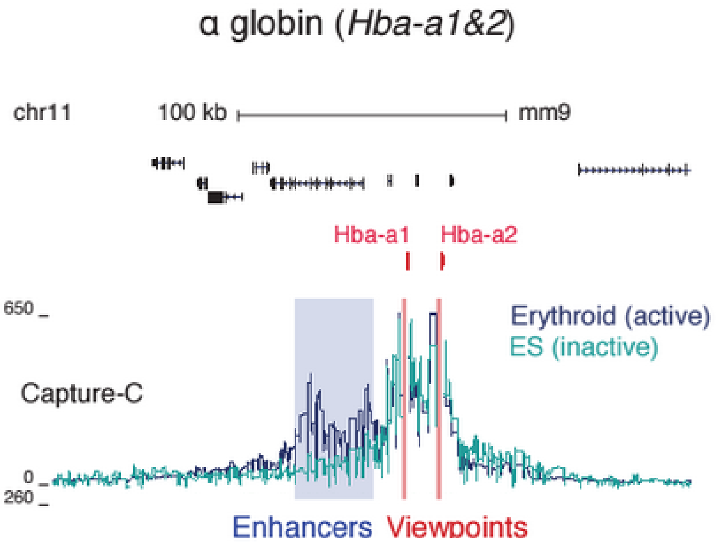
5 (red dots)



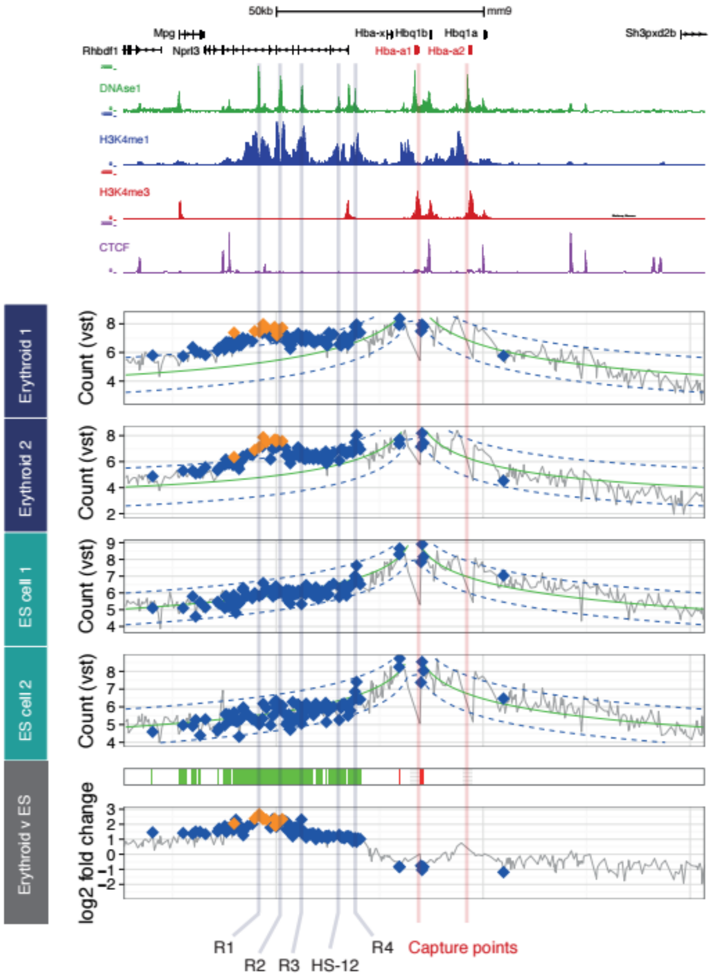
6 (blue dots)

7 (orange dots)

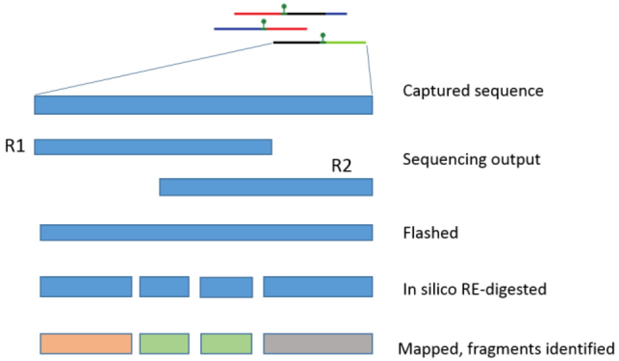
Differential analysis with FourCSeq ..



Supplementary Figure 21



Differential analysis with FourCSeq ..
Points to note..



Captured sequence

R1

R2

Sequencing output

Flashed

In silico RE-digested

Mapped, fragments identified

Signal strength (within sample) :

Highly sensitive to cutoffs in the “significantly over baseline within sample” stage – really difficult to set “universally applicable” (over all oligos) parameters to call the interaction sites reliably.

The bottleneck is not the Z-score (the difference of the red dot and green line), but its P-value (the FDR for the Z-score has to be set as high as 0.1 to see ANYTHING in our capture data).

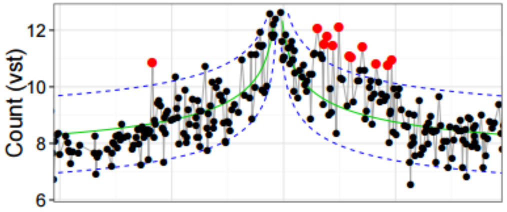
Assigning reads to fragments :

SAM Fragments are re-mapped to RE-fragments

FourCSeq demands read to start PRECISELY at DpnII cut site on BOTH sides of the read, and assumes “strandedness” (true paired end character of reads)

→ with default parameters maps only 30% of our already cleaned CCanalyser data

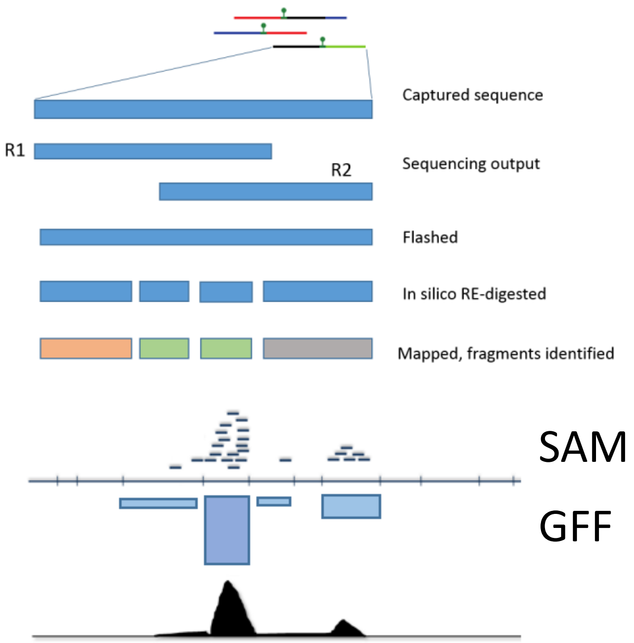
Can be tweaked, but this leads to reads mapping to neighboring fragments (in some extent), as well as “double-mapping” for short DpnII fragments.



Hard to get “enough” red dots here

CaptureC + r3Cseq

1) Normal CaptureC analysis :



Filtering steps within CCanalyser :

- save only reads which contain a capture (something mapped between the DpnII cut sites both sides of the capture oligo)

- fragment identification (capture, exclusion, reporter)

- duplicate read removal

- duplicate reporter removal (last fragment in R1/R2 "duplications")

Output files of Ccanalyser :

- sam of reporter reads for each oligo (minus exlusions, minus capture fragments)

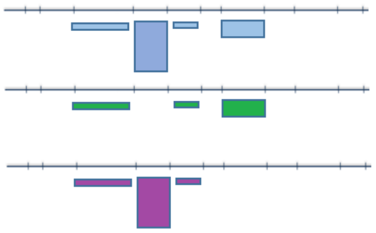
- gff of reporter reads for each oligo (counts of reporter fragments per DpnII fragment)

2) Filtering :

- filter each oligo results (reporter fragments) for presence of Blat and Ploidy regions
→ output as SAM file !

3) Read into FourCSeq :

- FourCSeq reads the SAM, and maps the reads back to DpnII fragments.



DpnII signal (reconstructed in R)

Differential analysis with r3Cseq

1) Power-law fit as the distance correction

- parameters of the curve were set by ChIP peak set, including beta globin, and tested with real data and known elements

→ They call these power-law linearized values as "RPM values"

→ The deviation of these values from the power law curve, are called the Z-values (how many STDs the values differ)

→ The q-values and p-values are counted based on this fit – the higher you are "from the baseline", the more probably you are a "real interaction" (FDR rate 0.05 set as cutoff within-sample)

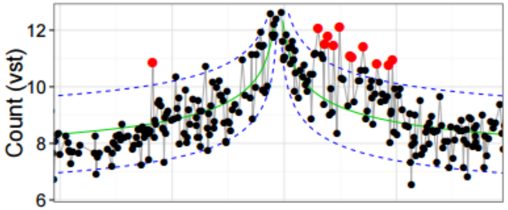
2) Differential analysis :

- Simple log2 fold change (below -1, or higher than 1)

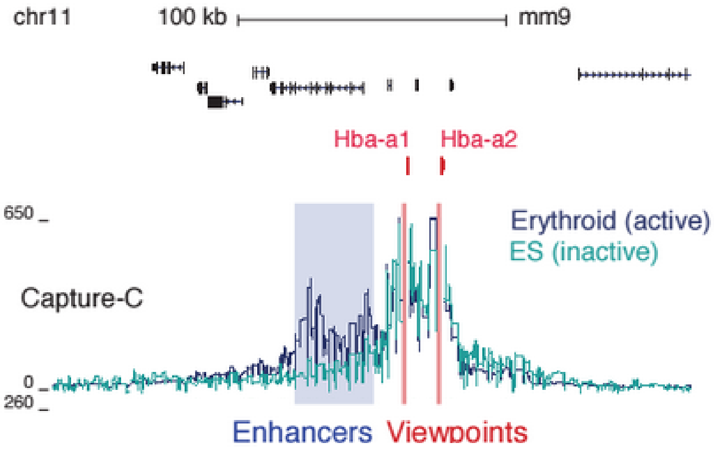
3) Replicates in differential analysis

→ "intersect" = significant log fold change in signal for at least ONE of the samples

→ "union" = significant log fold change in signal for ALL of the samples

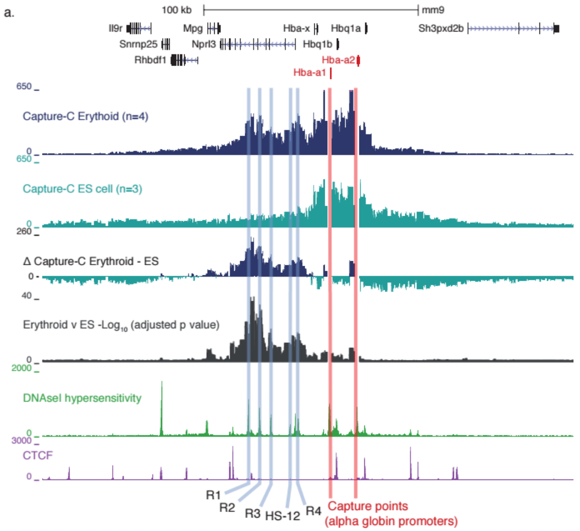


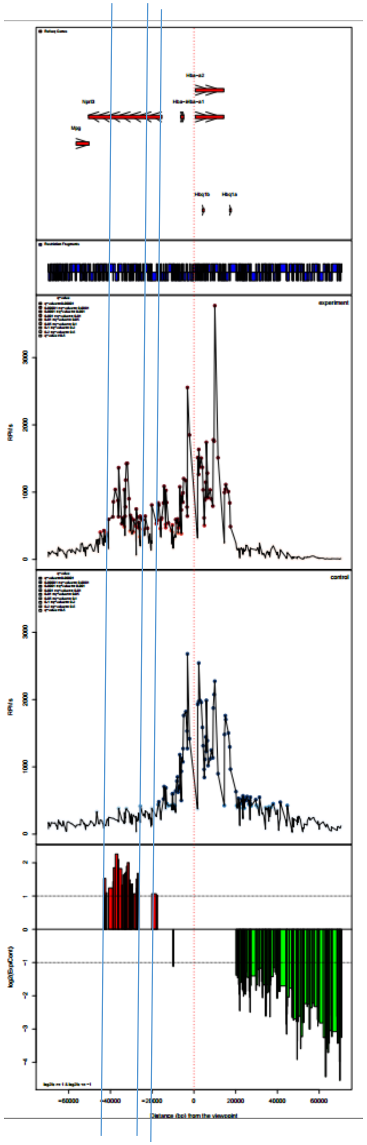
α globin (*Hba-a1&2*)

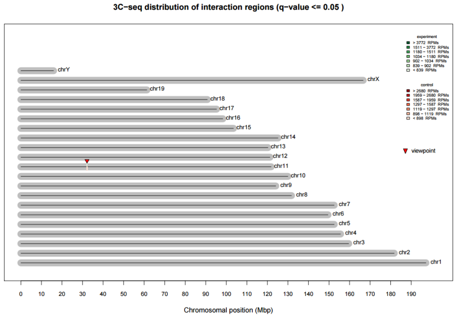


Differential analysis with r3Cseq ..

Supplementary figure 5






3C−seq distribution of interaction regions (q−value <= 0.05)

Summary – differences between the tools

	DESeq2	FourCSeq	r3Cseq
Takes in CCanalyser output ?	Seamless (CCanalyser is designed to do this)	Cumbersome (4C paired end reads as default input)	Straightforward (only a small hassle)
Assigning to RE fragments	Uses the counts generated by CCanalyser	Troublesome – produces errors (would need a custom pseudo-SAM generator code to eliminate all wrong assignments)	Straightforward (assigns to RE fragments the same way as CCanalyser, and trusts user that only "already filtered" reads are inputed)
Data transform to allow interpretable [10 .. 10 000] reads dynamic range ?	Yes (DESeq2 is designed to do this)	Yes (uses DESeq2 to do this)	No (so – calls large significant differences also in regions where very few reads coverage)
Distance correction ?	No	Monotone decay (both sides of capture site) to DESeq2 transformed counts.	Power law fit (using parameters typically suitable for ChIP-seq data distribution)
Correct predictions if significantly different interacting regions ?	Calls large regions (like "traditional peak caller") – for 9 out of 10 of the 30genes set these predictions are correct	Calls individual RE fragments (like MACS2 peak call) – there is hope to tune the call to generate "default call parameters for all CCanalyser data" to use this to routinely fine-tune DESeq2 analysis	Tends to over-call (far from capture site) and under-call (close to capture site). Parameter adjustment changes results only very little. Whole genome interaction tracks give nice illustration of possible trans and long range cis interactions.