Chip-Seq Analysis

István Albert

Bioinformatics Consulting Center

Penn State

Presentations

Lecture 1: Principles of Chip-Seq technology
Lecture 2: Interval datatypes

Two out of these

Tutorial 1: Peak Calling, analyzing enriched regions
Tutorial 2: Using the BEDTools package
Tutorial 3: Advanced scripting: fragment size detector
Tutorial 4: Tools I wish I knew about sooner

Problems with Statistics

- Descriptive statistics has was founded in the 19th century
- Designed to model a few and expensive and observations (agriculture, crop growth)
- Now well suited to any situation with lots of measurements and systematic errors
- Today we have enormous number of cheap, and biased observations

Topic: **Protein - DNA** interactions **ChIP-Chip** and **ChIP-Seq** studies

- **ChIP** → Chromatin Immuno-Precipitation (used during sample preparation)
- **Chip** → microarray technology to detect bound genomic locations
- **Seq** → high throughput sequencing to detect bound genomic locations

Slide credits

Some of the content on the following slides follow the presentation:

Statistical Methods and Software for mRNA-Seq and ChIP-Seq: Introduction to ChIP-Seq Data Analysis

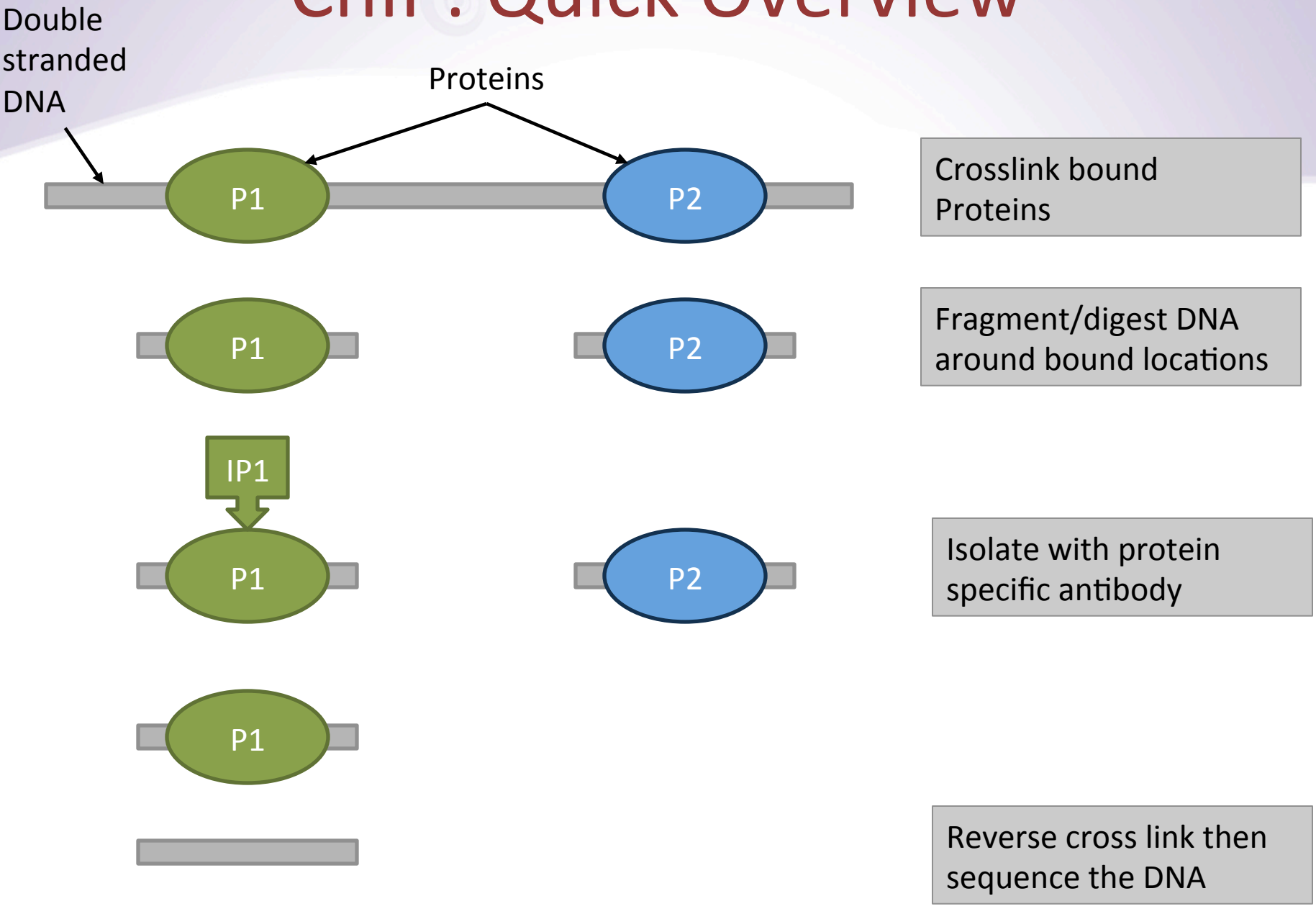
Oleg Mayba, Laurent Jacob, Sandrine Dudoit
Division of Biostatistics and Department of Statistics
University of California, Berkeley

Chromatin Immuno-Precipitation

It is a well know methodology to detect:

- transcription factor binding
- polymerase binding
- chromatin structure and modifications
- etc…

ChIP: Quick Overview



The ChIP output

- a DNA sample **enriched*** for fragments associated with the events under study
- Coverage depends on the number of sites, efficiency of the IP step.
- Accuracy depends on **fragmentation** strategy: sonication, MNASE digestion, lambda-nuclease digestion

* Note: Lots of other DNA fragments can make it through!

Library preparation

Intermediate step between ChIP and Sequencing. Lots of technical details. The goal is to create sequencing library that will result in best quality of data.

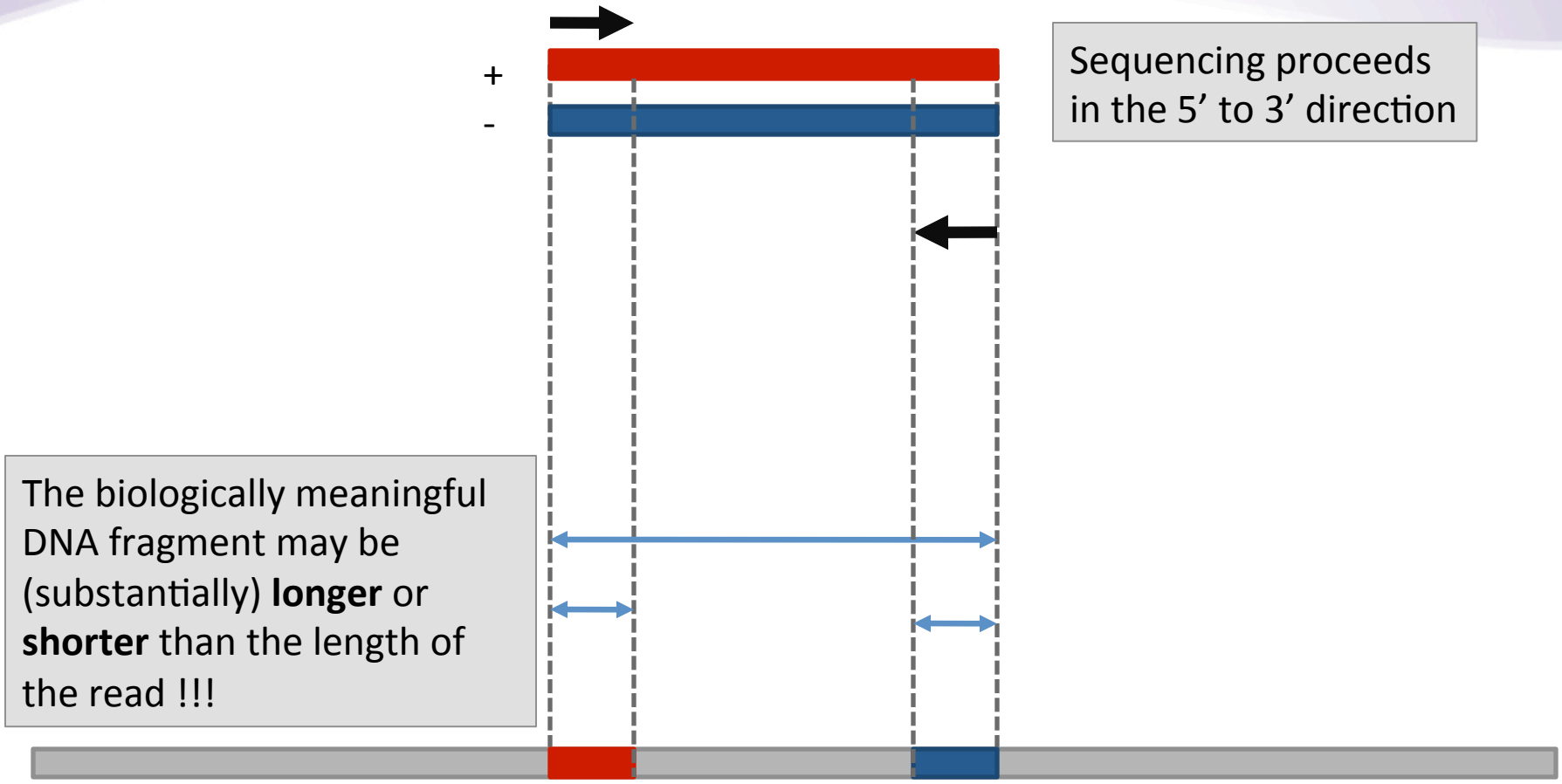
- PCR amplification – to increase amount of starting DNA.
- Fragment size selection – narrowing down the size distribution of DNA fragments.

Note that each step may also introduce errors and artifacts.
Errors almost always accumulate (often quadratically).

Chip-Seq →High throughput sequencing

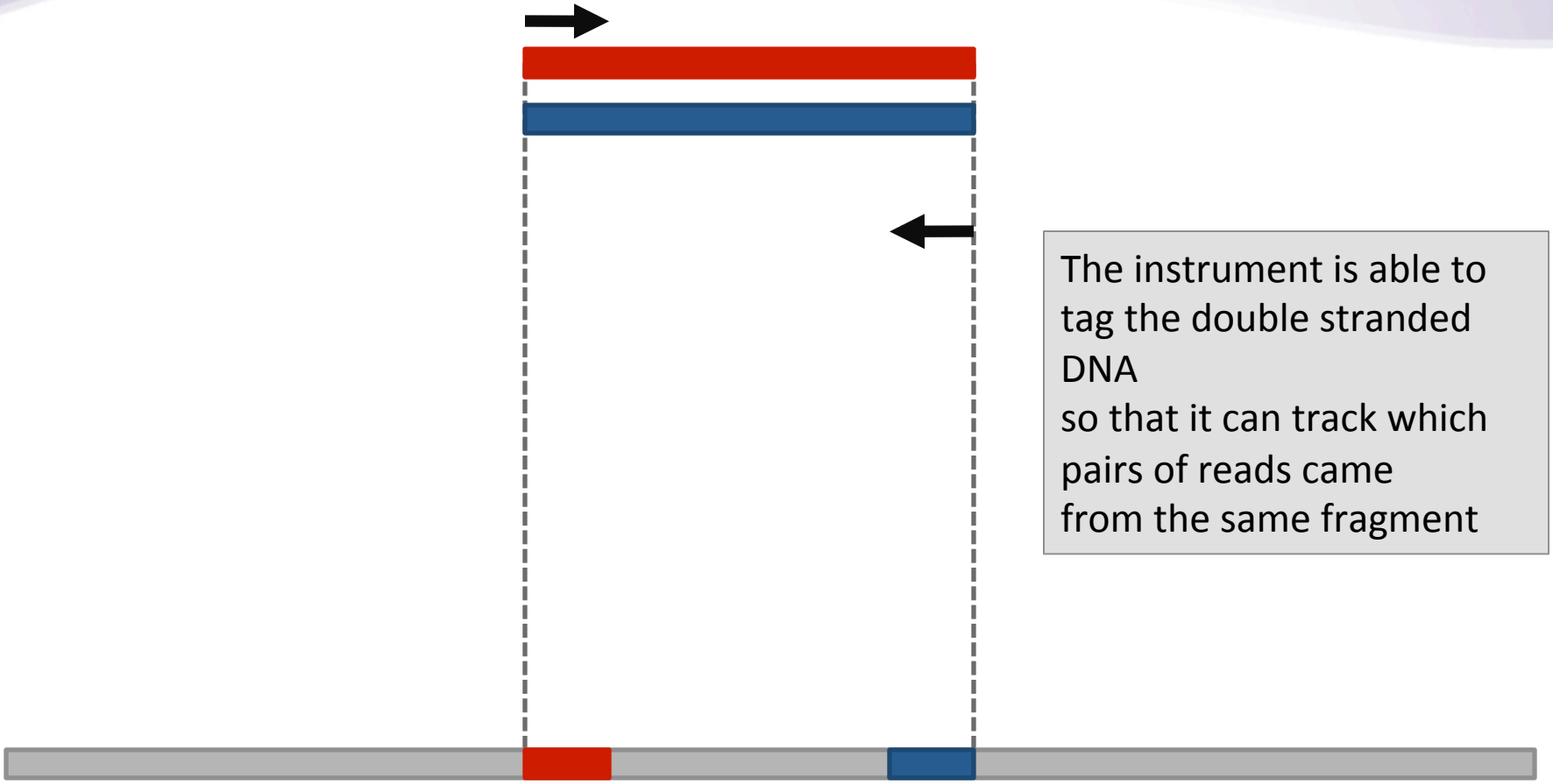
- Fragments are sequenced
- Aligned against genome (Bowtie is a good choice for Chip-Seq)
- The output is in the form of the intervals (start/end/strand) indicating locations in the genome

Single End sequencing



Plus we always have more than one fragment in one location and we don’t know which left border is paired up with which right border.

Paired End sequencing



The instrument is able to tag the double stranded DNA
so that it can track which pairs of reads came from the same fragment

Data will comes in pairs of right-left border
We always know the width of each fragment!

Alignment to the genome

After alignment we get genomic intervals for each read, minimally:

chrom, start, end, strand

The 5’ locations will then be given by

the **start** coordinate for the + strand

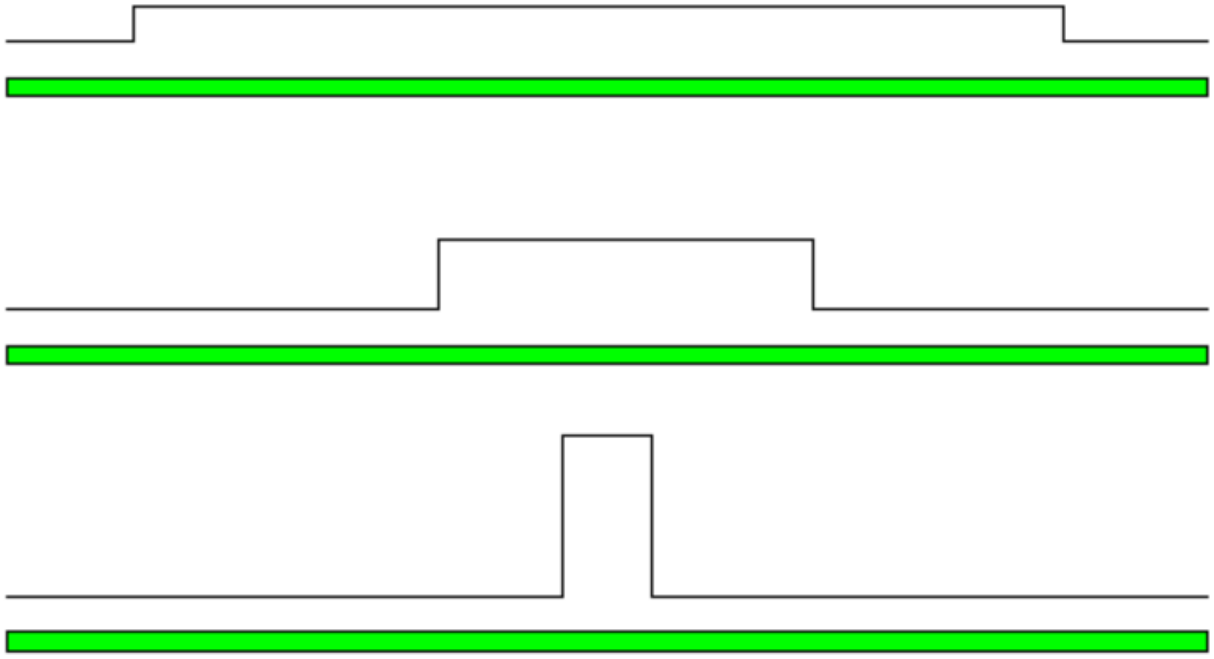
the **end** coordinate for the – strand

Other considerations

- For single end sequencing each fragment may correspond to 0, 1 or 2 reads. If it has 2 we don't know which two correspond to one another.

- For paired and sequencing each fragment corresponds to 0 or 2 reads. (occasionally 1 because of errors). We know which two correspond to one another.

Type of events of interest



Long-range

(e.g., histone modifications)

Mid-range

(e.g. polymerase binding)

Punctate

(e.g. TF binding)

Peak Calling

- Process of finding the locations enriched due to events of interest

Need to define

- **Peak Region** - contiguous set of basepairs that belong to a peak
- **Enrichment Level** - read-based measure of supporting evidence

Peak calling: base pair level measurements

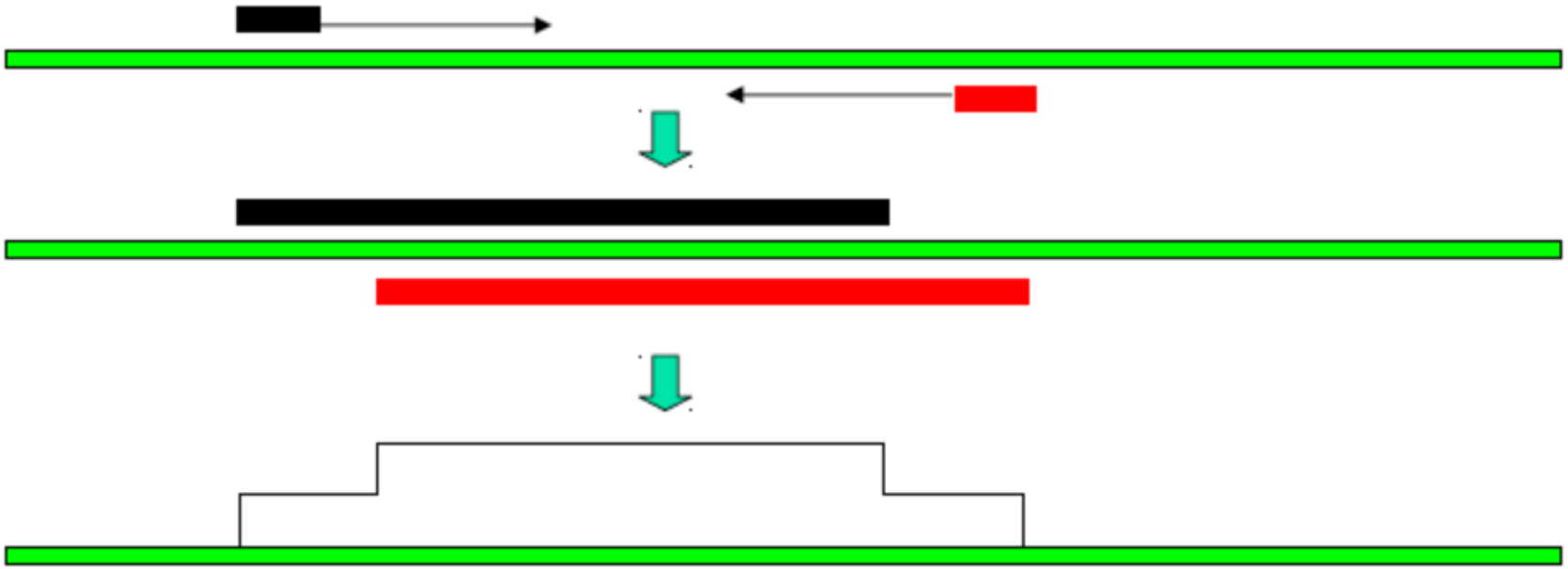
Typically two strategies:

- Find the number of fragments (usually NOT reads) overlapping that position (need to go from reads to fragments)

- Find the number of reads (fragment ends) reported at that position (usually taking strandedness into account)

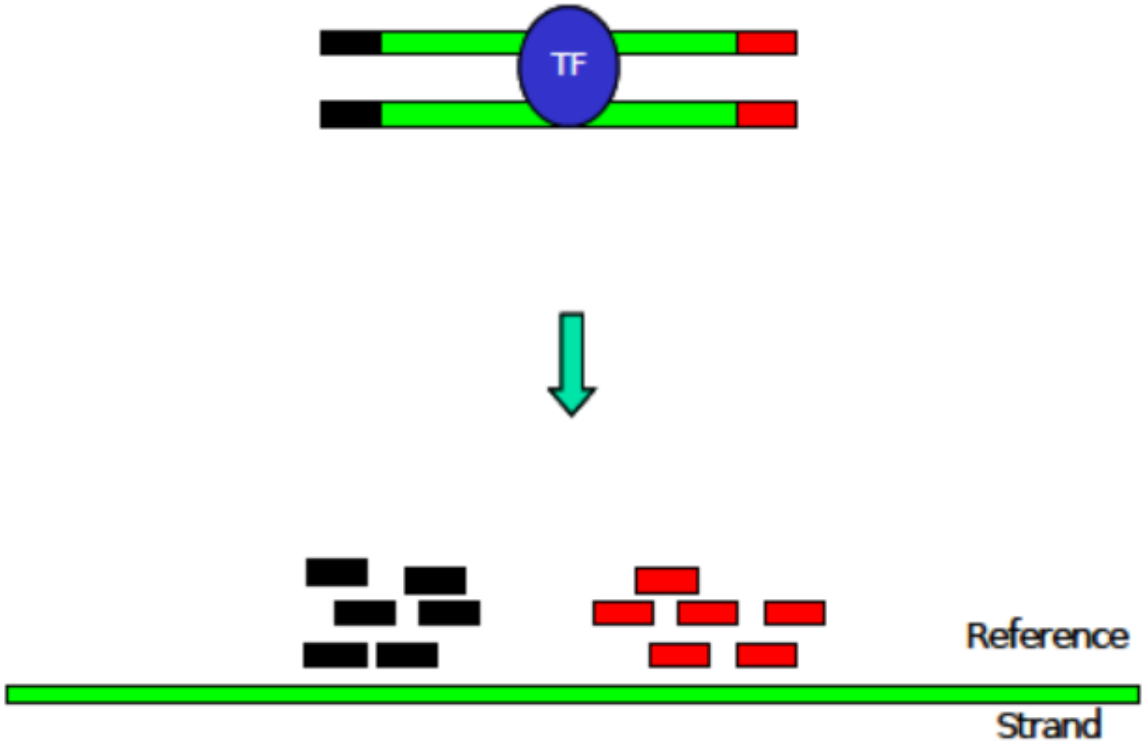
  Variation: **kernel-smoothed read density**. This is closer to overlap approach.

BP level enrichment: overlap



Number of fragments (usually NOT reads) overlapping that position

Estimation of Point-Like events



Peak calling tools

- Very large selection of tools and techniques:

  ERANGE, FindPeaks, MACS, QuEST, CisGenome, SISSRS, USeq, PeakSeq, SPP, ChIPSeqR, GLITR, ChIPDiff, T-PIC, BayesPeak, MOSAiCS, CCAT, CSAR,

Featured today as an example (historically precedes most of the tools above)

GeneTrack—a genomic data processing and visualization framework

Istvan Albert, Shinichiro Wachi, Cizhong Jiang and B. Franklin Pugh

Bioinformatics, 2008

GeneTrack

- It performs error corrections and peak predictions
- Visualizes data as tracks in a ‘browser’ interface.
- Also a integrated into our own LIMS: Laboratory Data Management System LionDB3: with projects, users, data sharing etc.

http://genetrack.googlecode.com the LIMS is at http://liondb3.atlas.bx.psu.edu/

GeneTrack Features

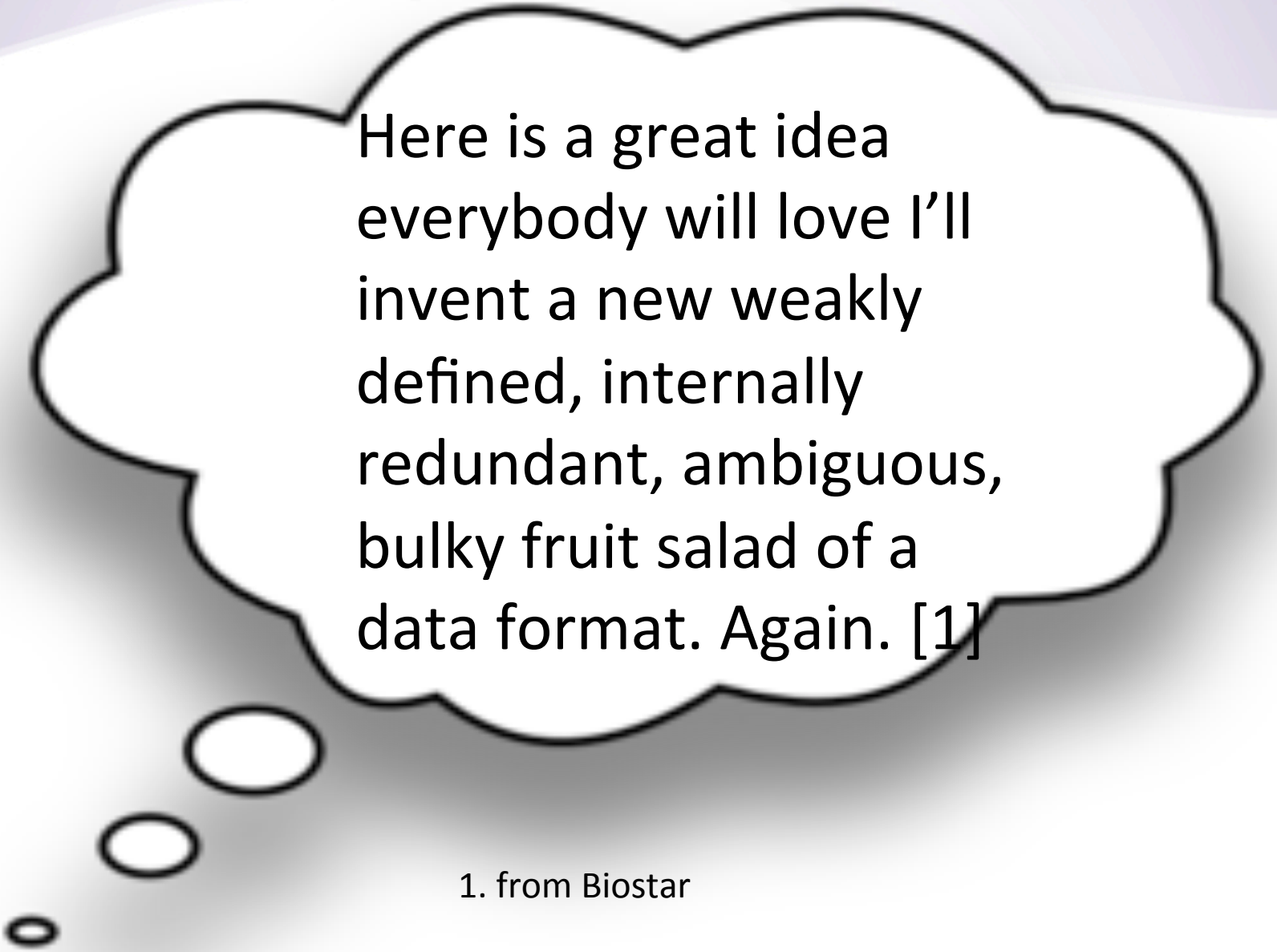
Unique features:

- Peak detection and visualization at the same time.
- User sees the effect of the peak prediction parameters right away

Much criticized missing component

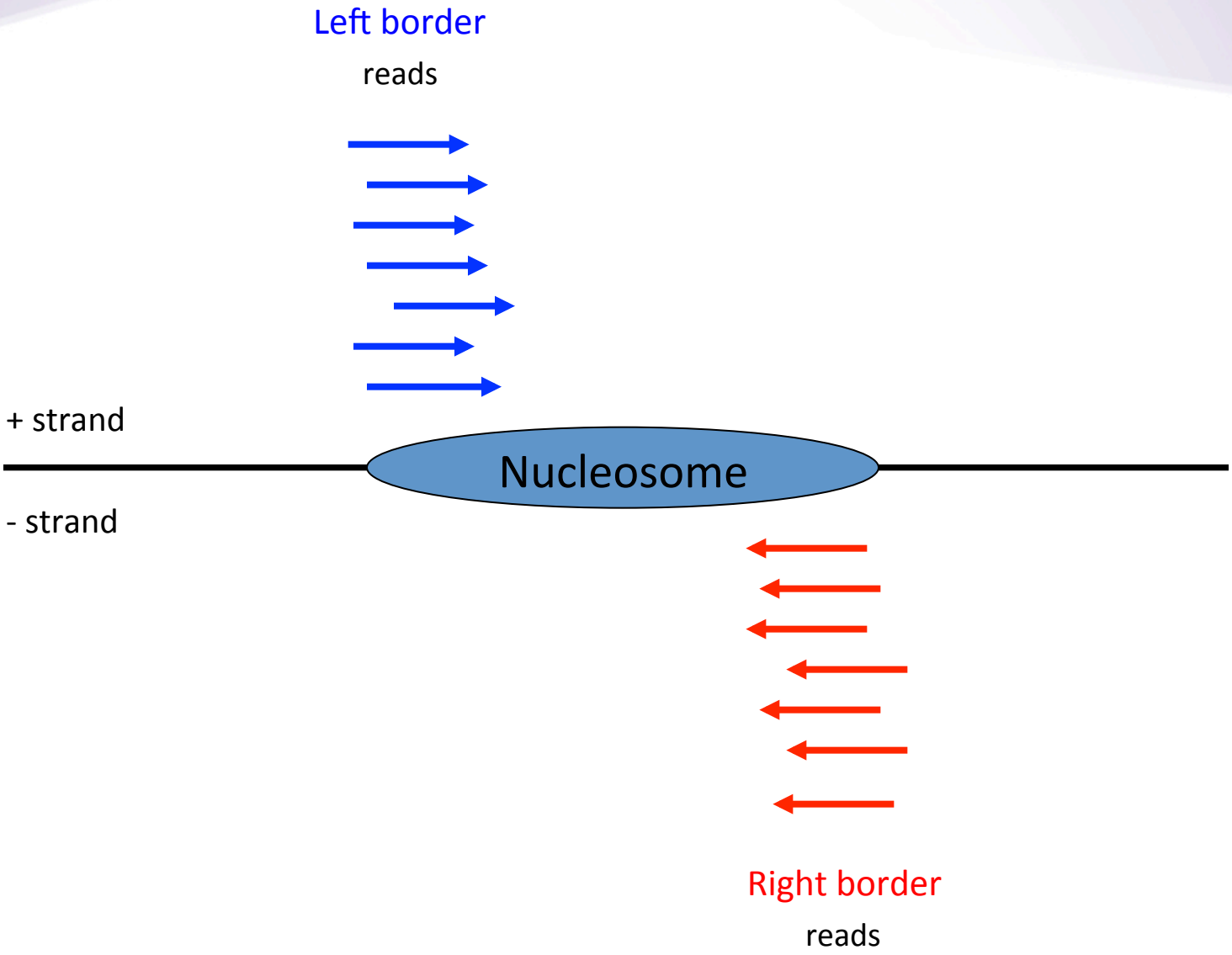
- Lacks the statistical modeling of the background (expected frequencies)

GeneTrack Input Files

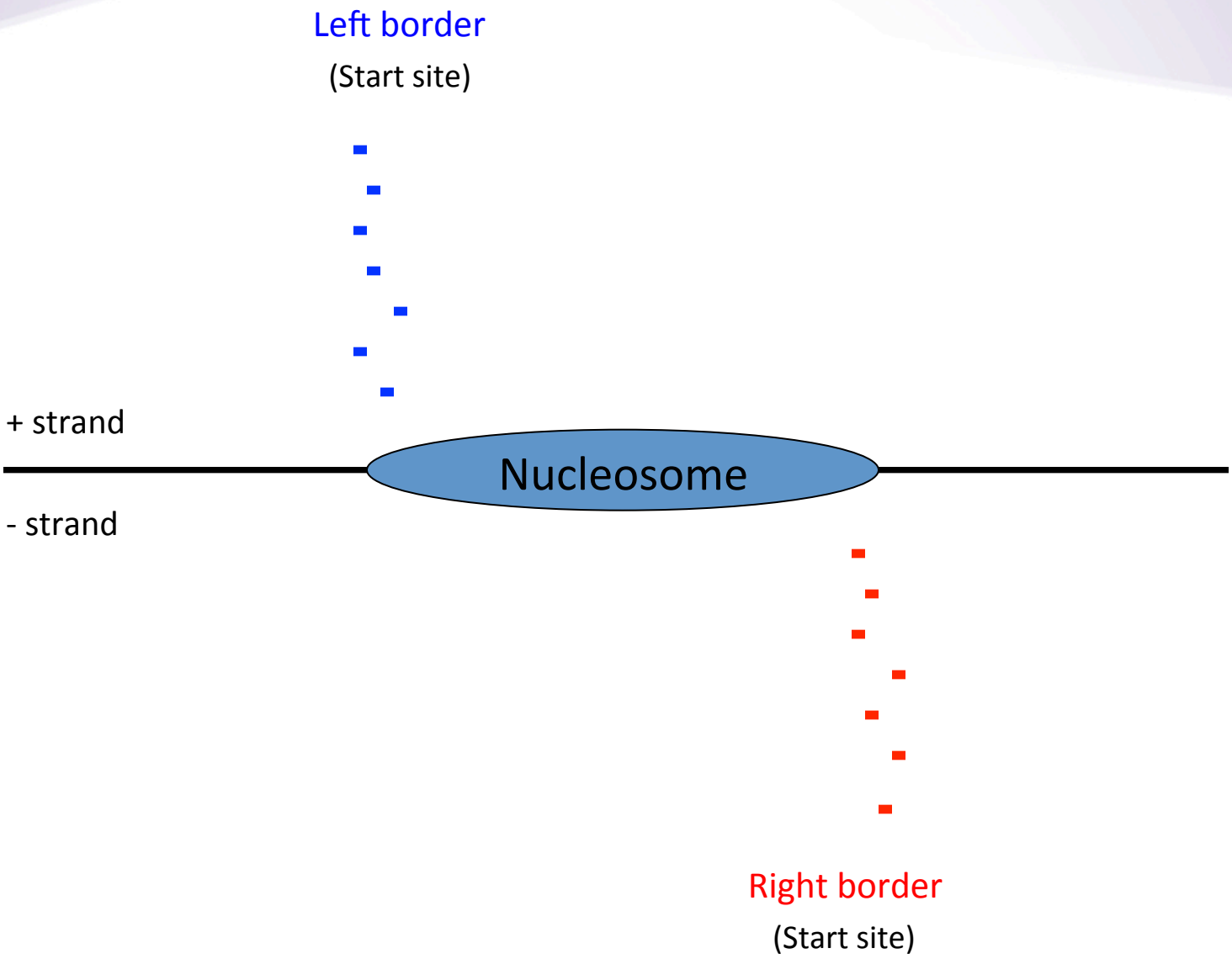


1. from Biostar

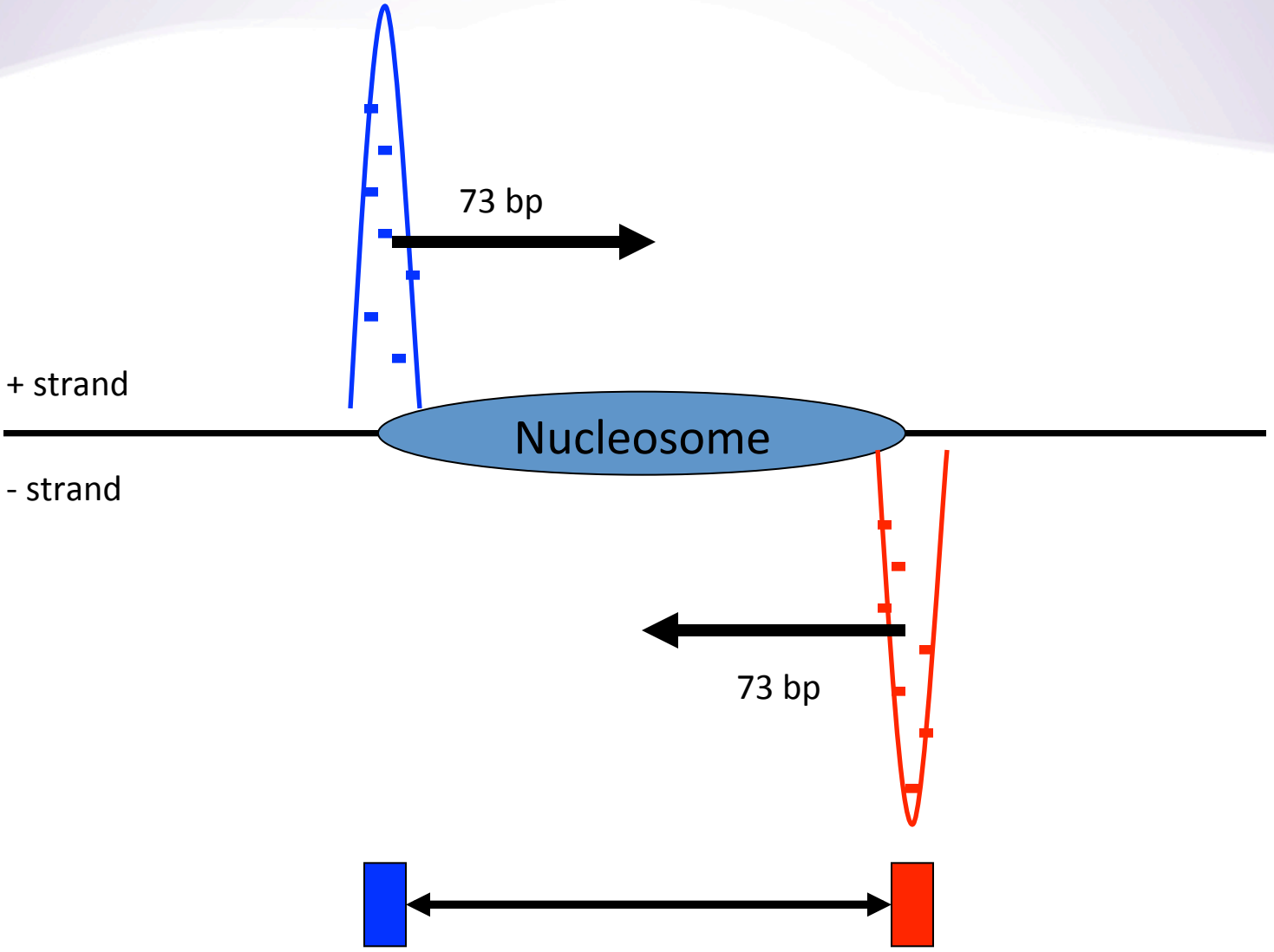
Aligning to reference genome



Find the borders

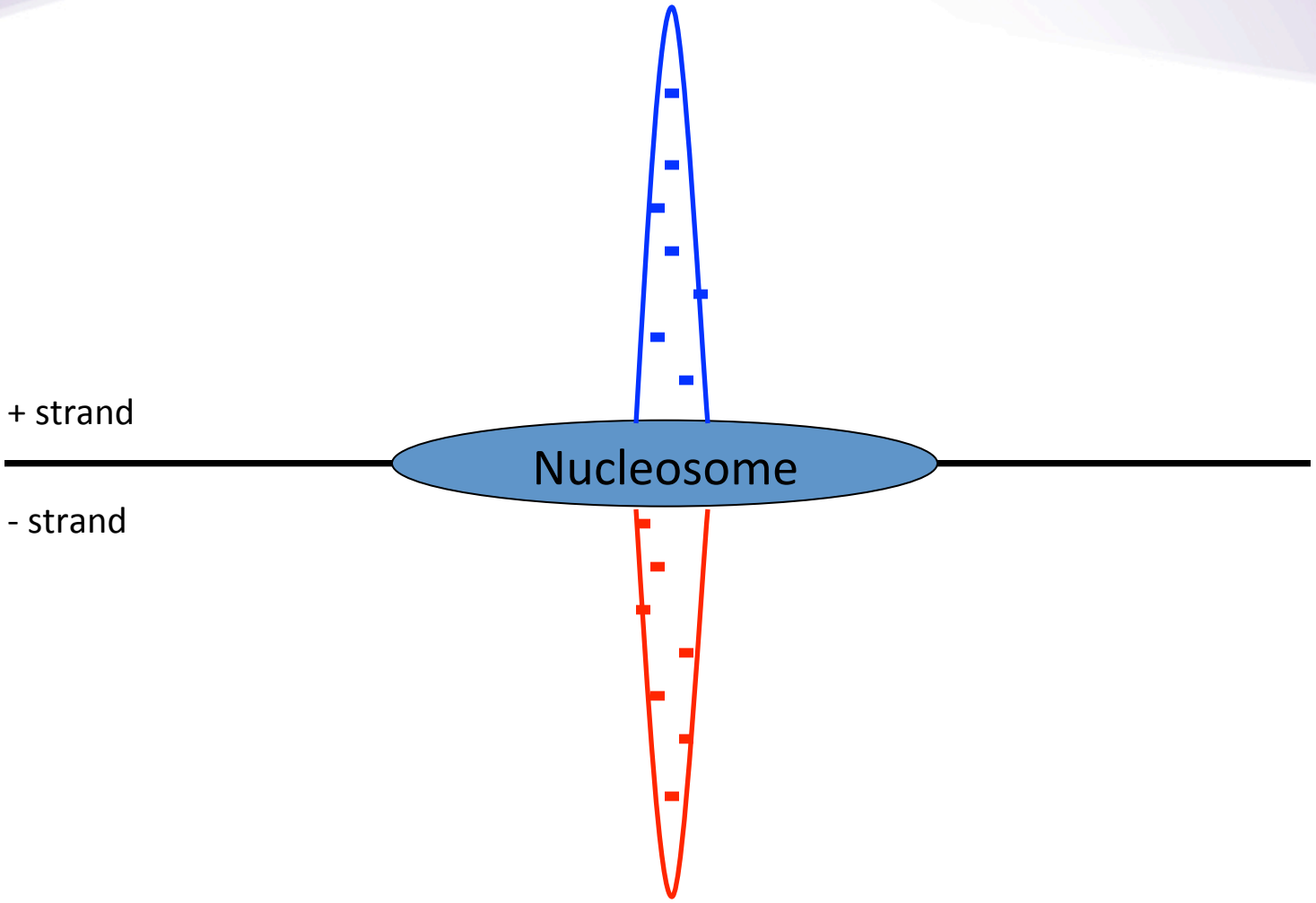


Finding the binding sizes

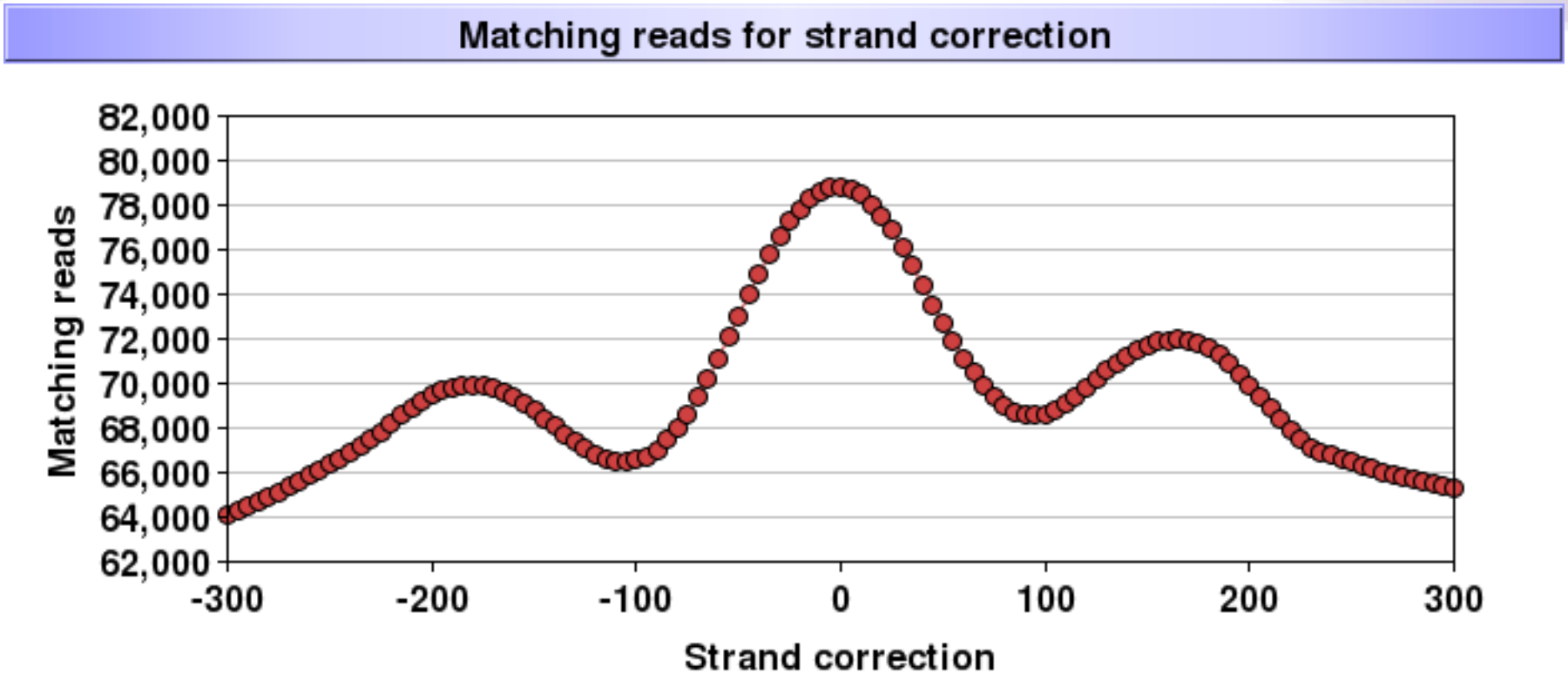


Binding Size = ~146 bp

Shift to the center

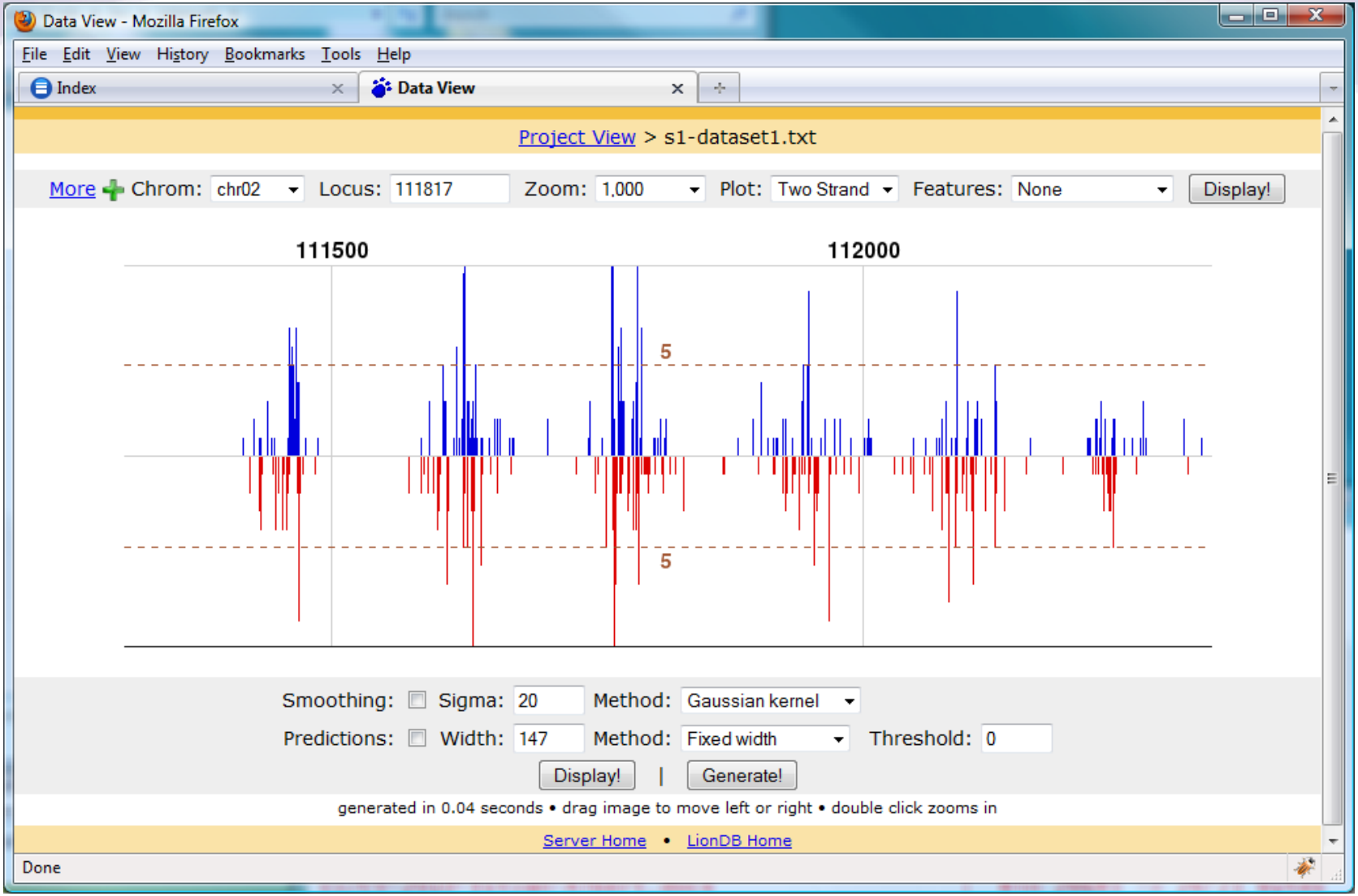


Errors and Noise



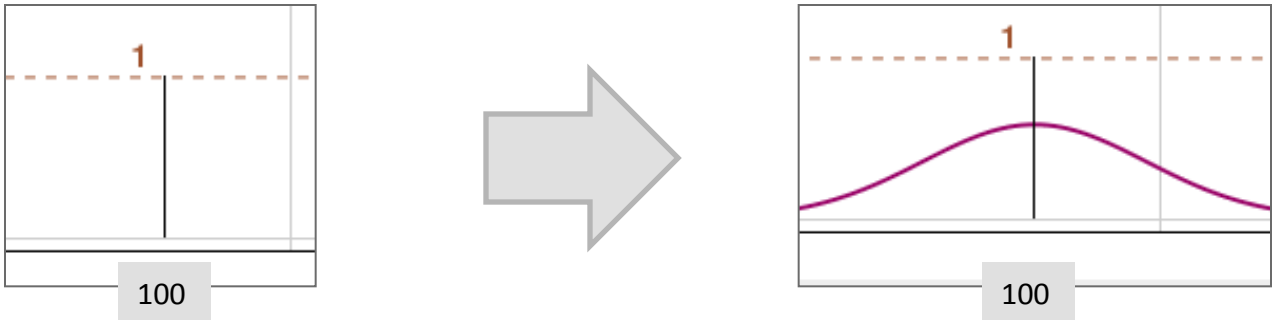
Counting positions where reads match up on both strands
Original read count ~ 1 million

GeneTrack: reads on both strands



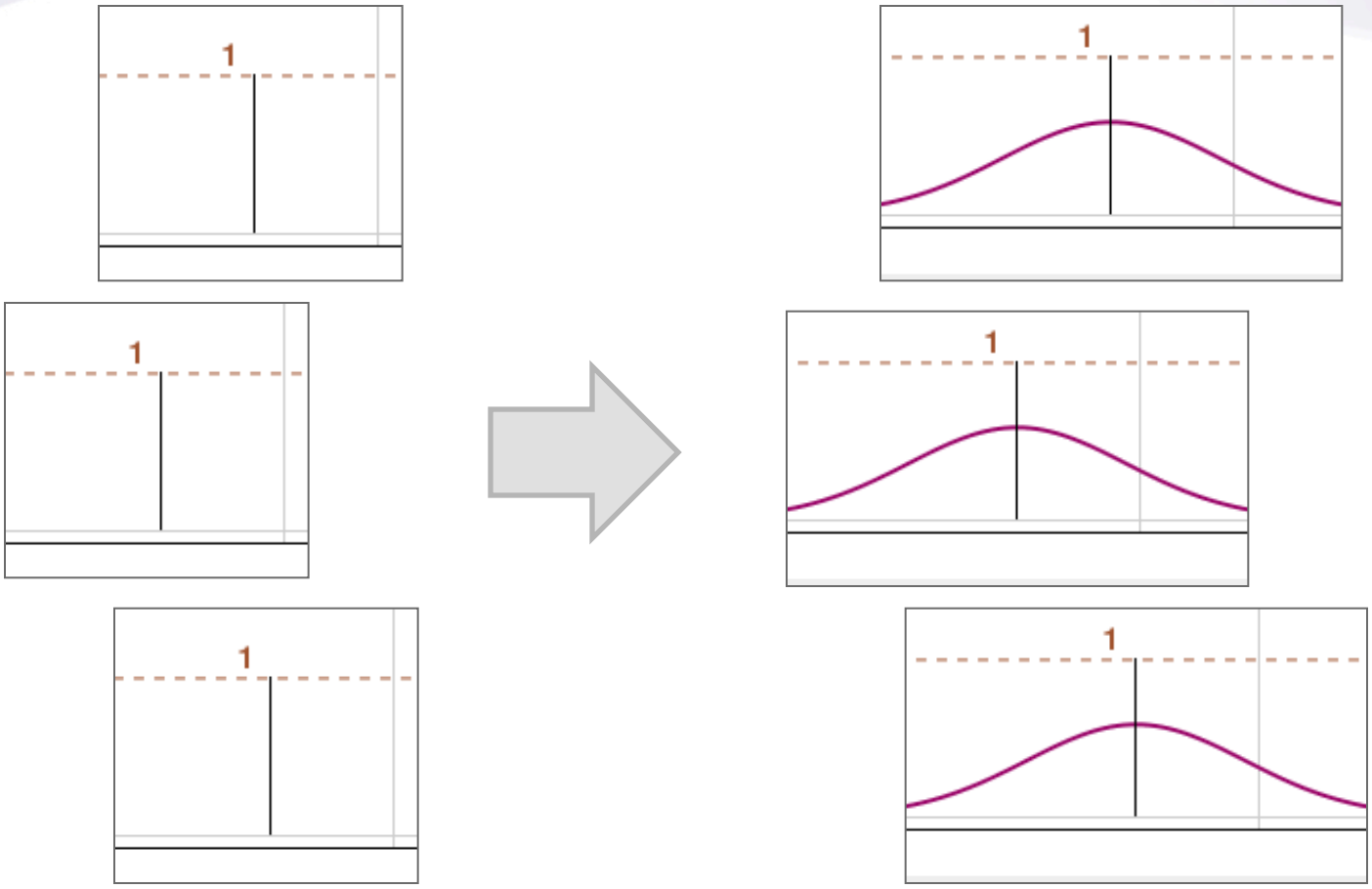
Error Model

- Replace every read with the probability of the read falling onto the observed location.



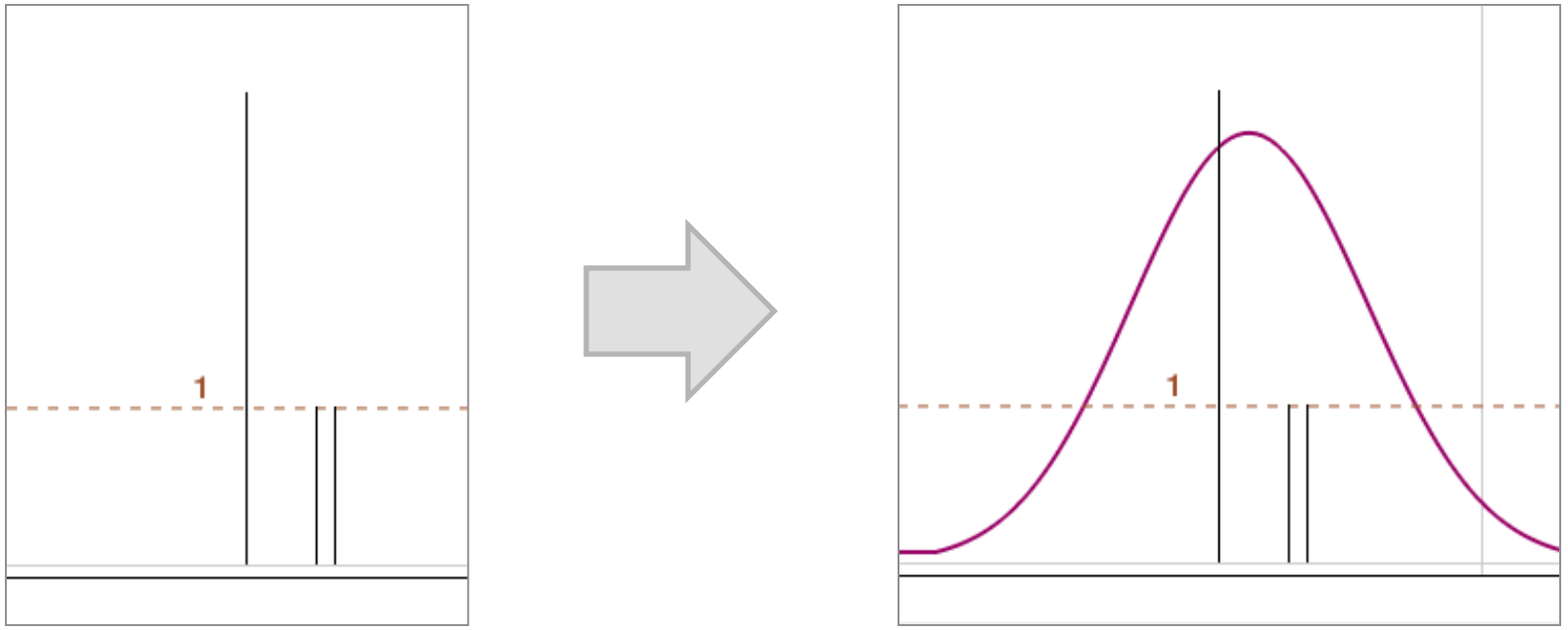
Each single coordinate will be replaced with a “smudge” of values (hundreds of them).

Sum up your “smudges”

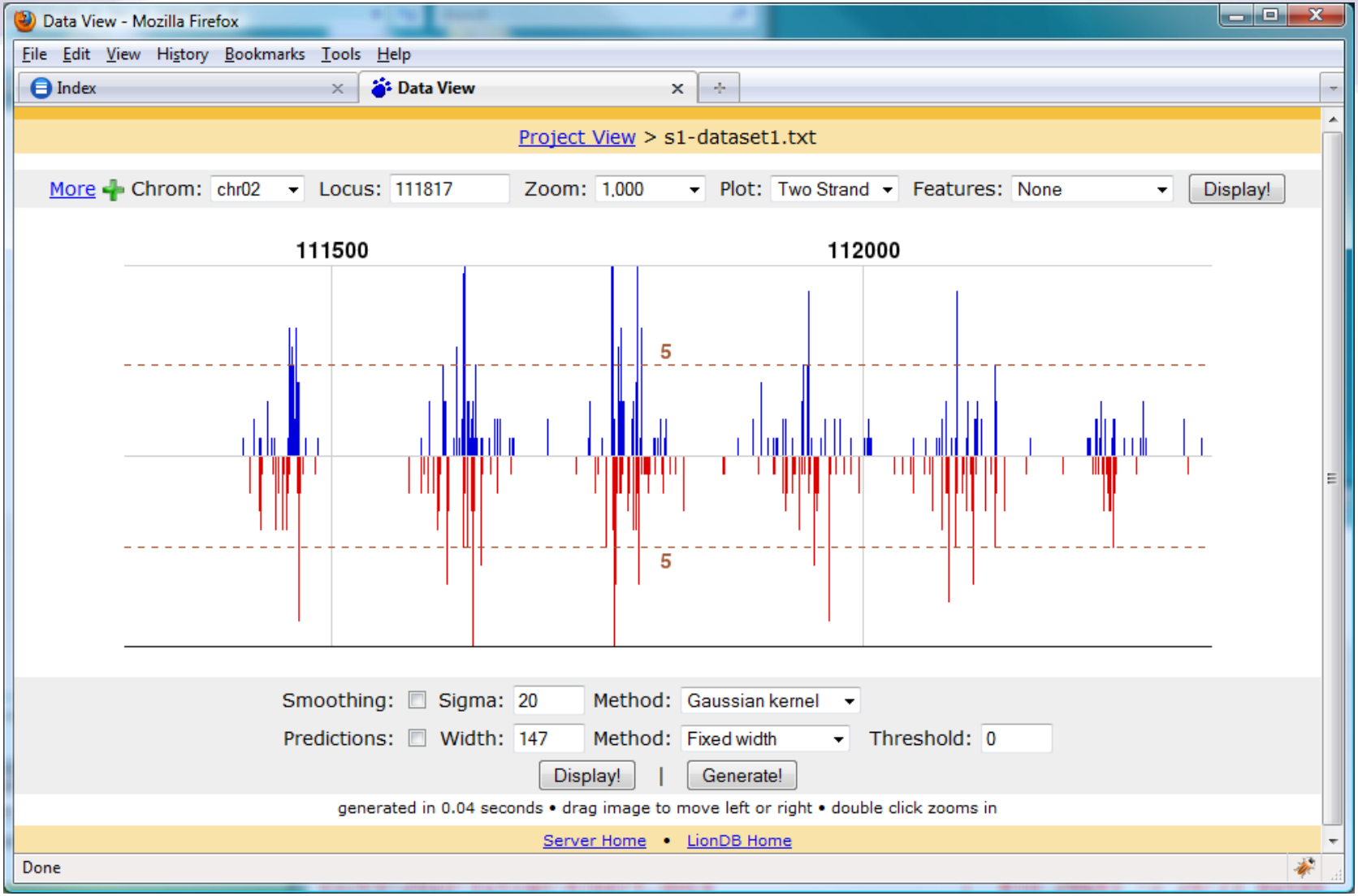


Central Limit Theorem guarantees another normal function

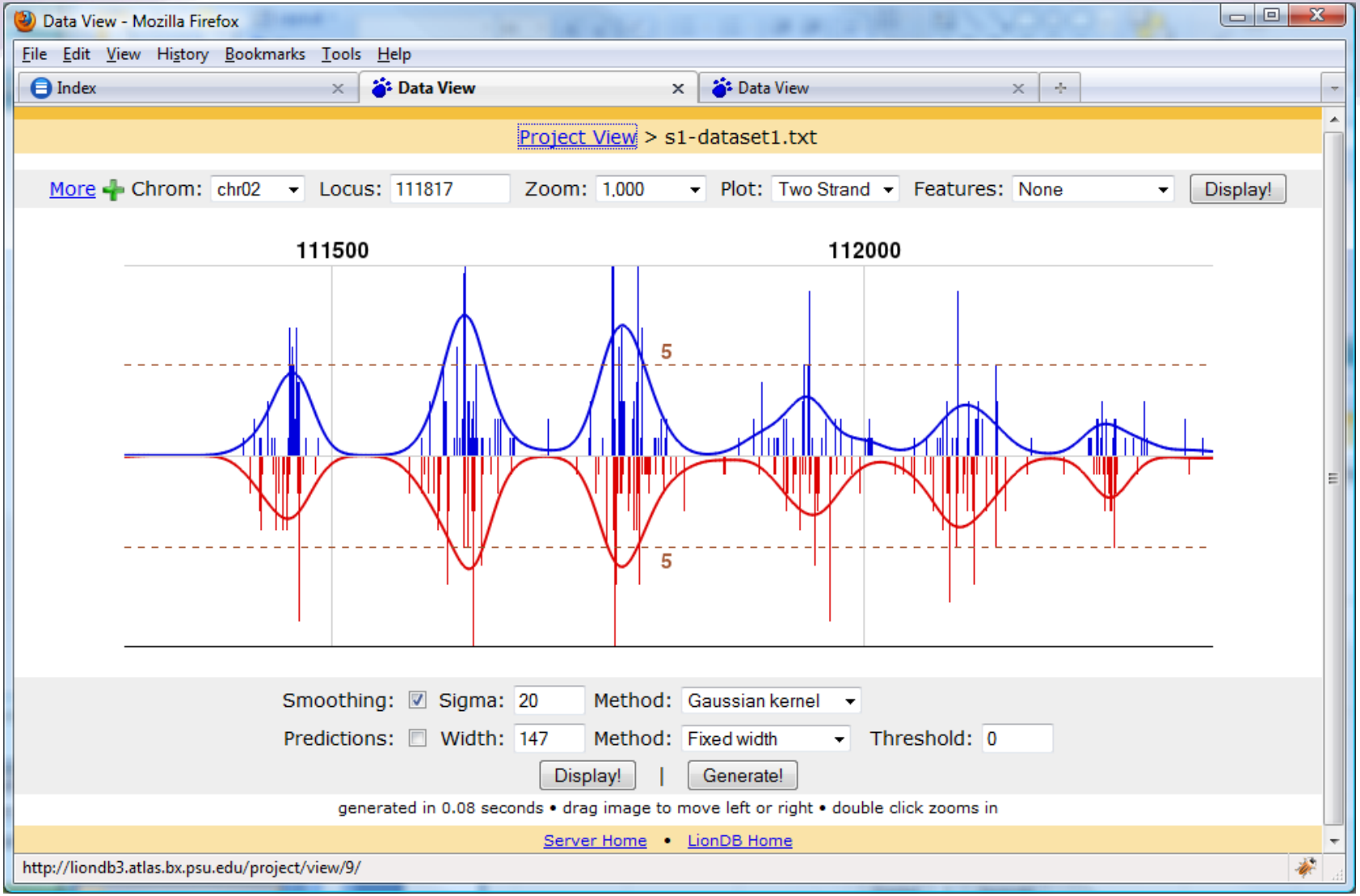
Summing up multiple errors



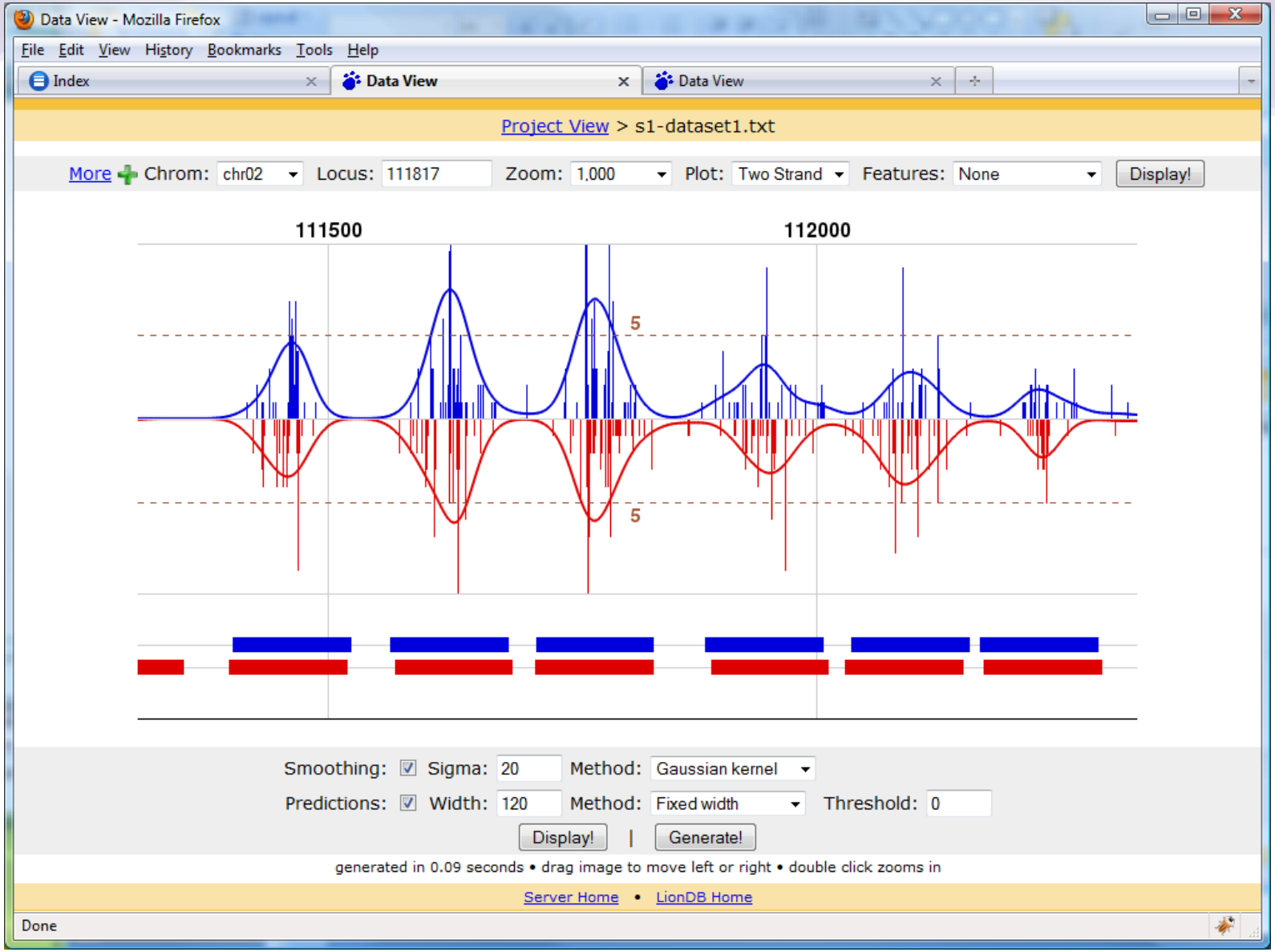
GeneTrack: revisit the reads



GeneTrack: with smoothing



GeneTrack also does peak prediction



Displayed relative to genomic features

I have my peaks what now?

- Two major directions (often both are pursued):

1. Peak locations relative to known genomic features → GO annotations → Functional inference

2. Extract the sequence that correspond to the peaks → Motif analyis

What is Chip-Seq data analysis really about?

Manipulating intervals

- query
- intersect
- summarize

Topic: Working with genomic Features

- Feature: a genomic region (interval) associated with a certain annotation (description).

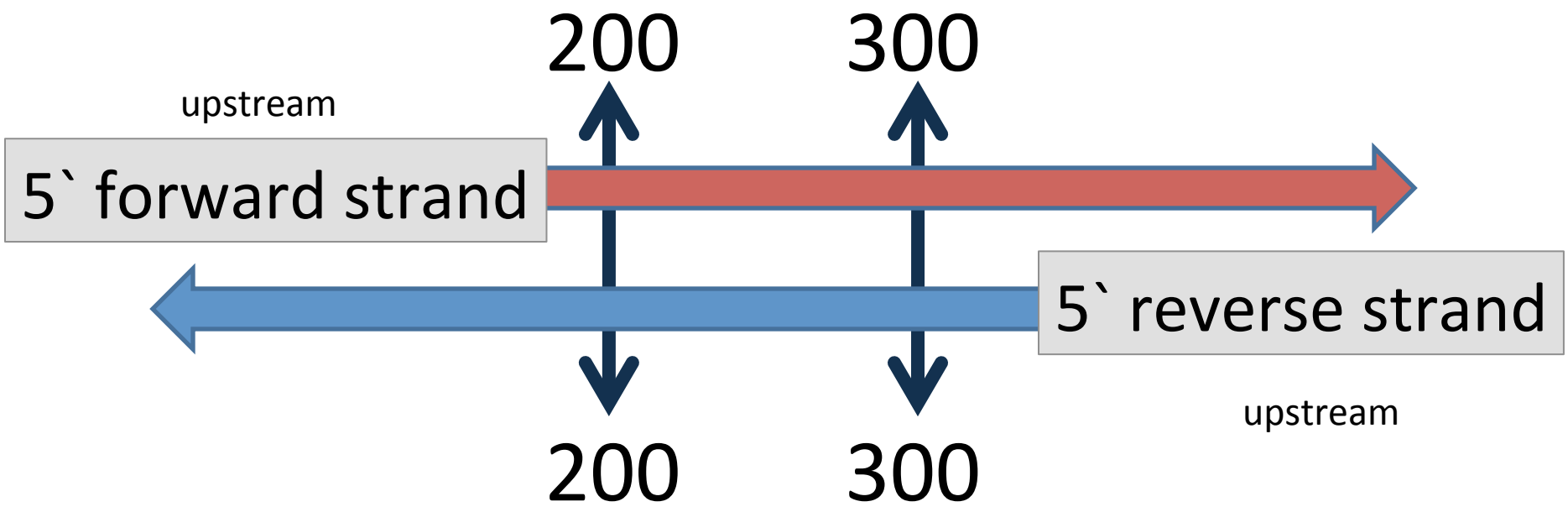
Typical attributes to describe a feature

1. chromosome
2. start
3. end
4. strand
5. Name

Related tutorial on how to operate on intervals with BEDTools

Genomic coordinates – brief overview

DNA two stranded and directional → But there is only one coordinate system



Most formats use **start < end** even on the reverse strand

The **upstream region** – before the 5' end relative to the direction of transcription

Coordinate systems

- 0 based → first 10 → 0, 1, 2, … 9
- 1 based → first 10 → 1, 2, 3, 4, … 10

Typically

- 0 based are non-inclusive 10:20 → [10, 20)
- 1 based include both ends 10:20 → [10, 20]

Why do different indexing systems even exist?

1 based indexing

Third element data[3] 

First ten : data[1:10]
Second ten: data[11:20]
Third ten : data[21:30]

Size of the slice = 10 → end-start + 1

Empyt slice: data[?] – not sure

Get a five element long segment starting at 1000

data[1000: 10000 + 4]

0 based indexing

Third element data[2]

First ten : data[0 :10]
Second ten: data[10:20]
Third ten : data[20:30]

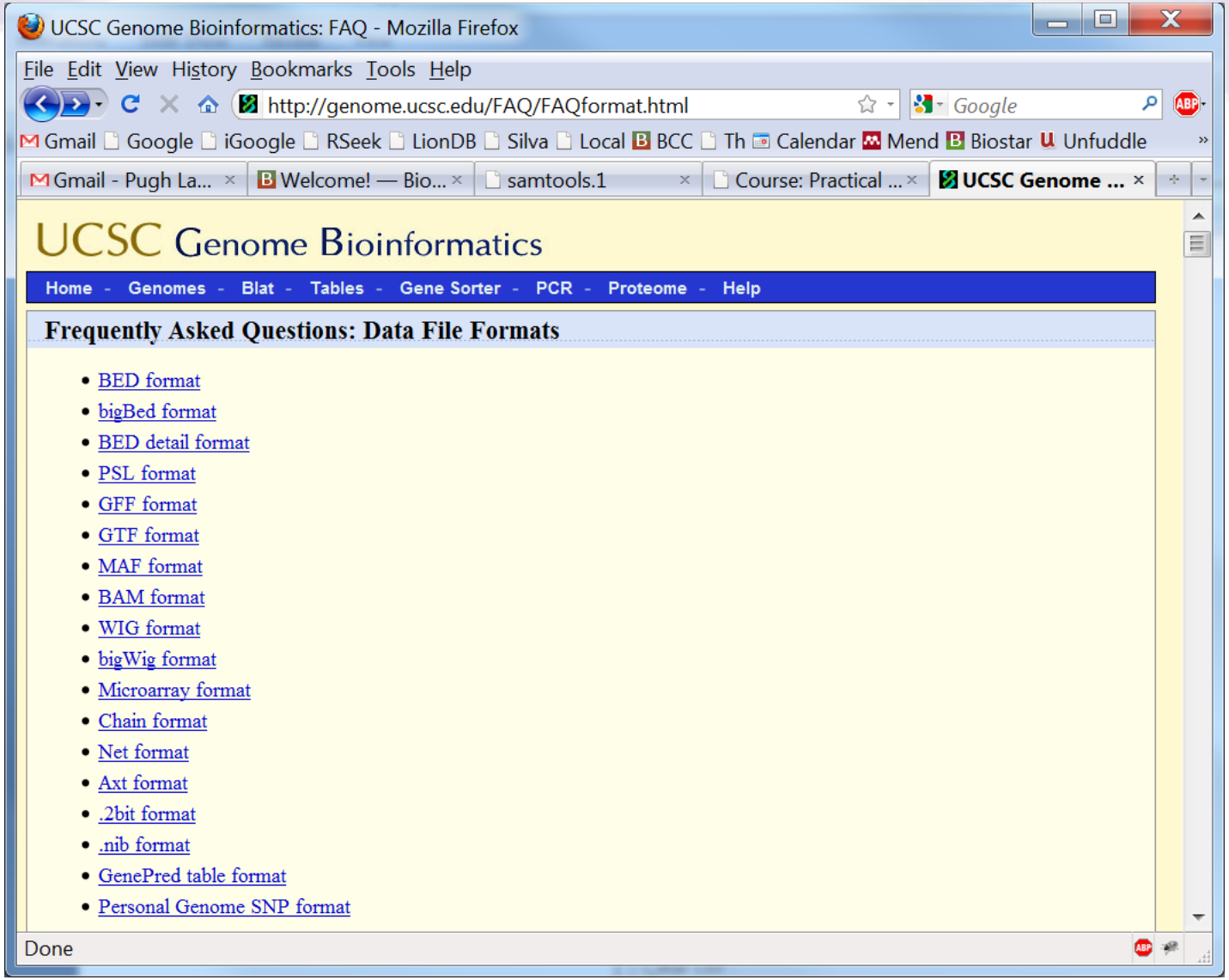
Size of the slice = 10 → end – start

Empty slice data[10:10] → size=0

Get a five element long segment starting at 1000

data[1000: 10000 + 5]

http://genome.ucsc.edu/FAQ/FAQformat.html



Two commonly used formats

- **BED** – UCSC genome browser → 0 based non inclusive → also used to display tracks in the genome browser (US “standard”)

- **GFF** – Sanger institute in Great Britain → 1 based inclusive indexing system (“European standard”)

- Neither of them includes column names

BED Format

The first three required BED fields are:

1. **chrom** - The name of the chromosome (e.g. chr3, chrY, chr2_random) or scaffold (e.g. scaffold10671).
2. **chromStart** - The starting position of the feature in the chromosome or scaffold. The first base in a chromosome is numbered 0.
3. **chromEnd** - The ending position of the feature in the chromosome or scaffold. The *chromEnd* base is not included in the display of the feature. For example, the first 100 bases of a chromosome are defined as *chromStart=0, chromEnd=100*, and span the bases numbered 0-99.

The 9 additional optional BED fields are:

4. **name** - Defines the name of the BED line. This label is displayed to the left of the BED line in the Genome Browser window when the track is open to full display mode or directly to the left of the item in pack mode.
5. **score** - A score between 0 and 1000. If the track line *useScore* attribute is set to 1 for this annotation data set, the *score* value will determine the level of gray in which this feature is displayed (higher numbers = darker gray). This table shows the Genome Browser's translation of BED score values into shades of gray:

shade									
score in range	≤ 166	167-277	278-388	389-499	500-611	612-722	723-833	834-944	≥ 945

6. **strand** - Defines the strand - either '+' or '-'.
7. **thickStart** - The starting position at which the feature is drawn thickly (for example, the start codon in gene displays).
8. **thickEnd** - The ending position at which the feature is drawn thickly (for example, the stop codon in gene displays).
9. **itemRgb** - An RGB value of the form R,G,B (e.g. 255,0,0). If the track line *itemRgb* attribute is set to "On", this RBG value will determine the display color of the data contained in this BED line. NOTE: It is recommended that a simple color scheme (eight colors or less) be used with this attribute to avoid overwhelming the color resources of the Genome Browser and your Internet browser.
10. **blockCount** - The number of blocks (exons) in the BED line.
11. **blockSizes** - A comma-separated list of the block sizes. The number of items in this list should correspond to *blockCount*.
12. **blockStarts** - A comma-separated list of block starts. All of the *blockStart* positions should be calculated relative to *chromStart*. The number of items in this list should correspond to *blockCount*.

Example:
Here's an example of an annotation track that uses a complete BED definition:

```
track name=pairedReads description="Clone Paired Reads" useScore=1
chr22 1000 5000 cloneA 960 + 1000 5000 0 2 567,488, 0,3512
chr22 2000 6000 cloneB 900 - 2000 6000 0 2 433,399, 0,3601
```

GFF format

Search for GFF3 → http://www.sequenceontology.org/gff3.shtml

Tab separated with 9 columns. Missing attributes may be replaced with a dot → .

1. **Seqid** (usually chromosome)
2. **Source** (where is the data coming from)
3. **Type** (usually a term from the sequence ontology)
4. **Start** (interval start relative to the seqid)
5. **End** (interval end relative to the seqid)
6. **Score** (the score of the feature, a floating point number)
7. **Strand** (+/-/.)
8. **Phase** (used to indicate reading frame for coding sequences)
9. **Attributes** (semicolon separated attributes → Name=ABC;ID=1)

We may have data in different coordinate systems!

Being "**one off**" is one of the most common errors in bioinformatics.

Conversion from GFF to BED

(start, end) → (start – 1, end)

Conversion from BED to GFF

(start, end) → (start + 1, end)

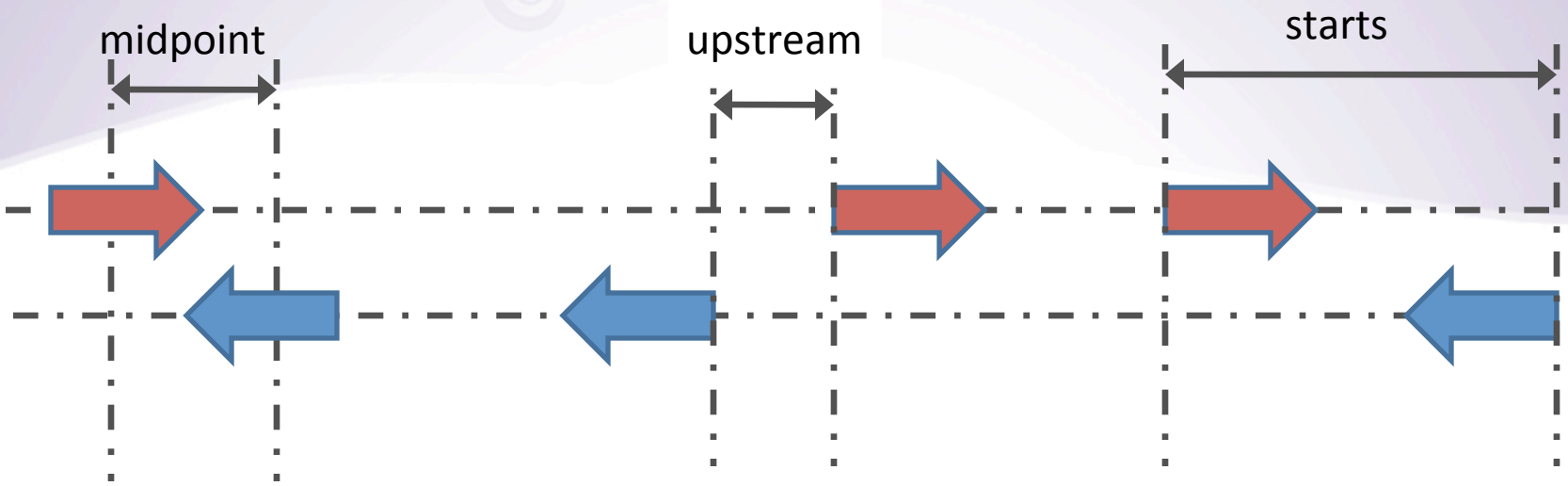
Typical interval related tasks

Finding intervals relative to one another, for example:

- for each interval on one strand find the closest on the other strand

An interval is not a point - these type of problems need to be better specified

Important details

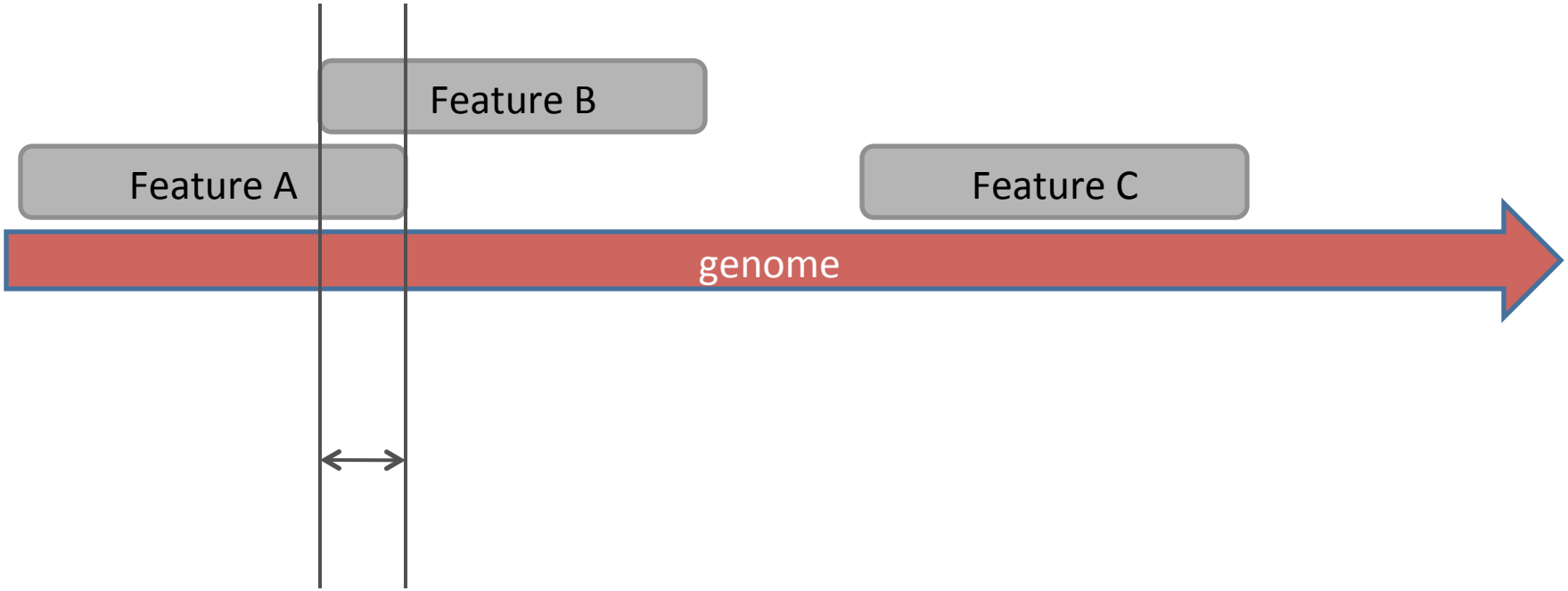


- What are the anchor points (the locations that represent the intervals)

- Which direction does the comparison proceed – upstream, downstream?

Overlap/intersect

- Two features are said to overlap or intersect if they share at least one base in common.



Further complications …

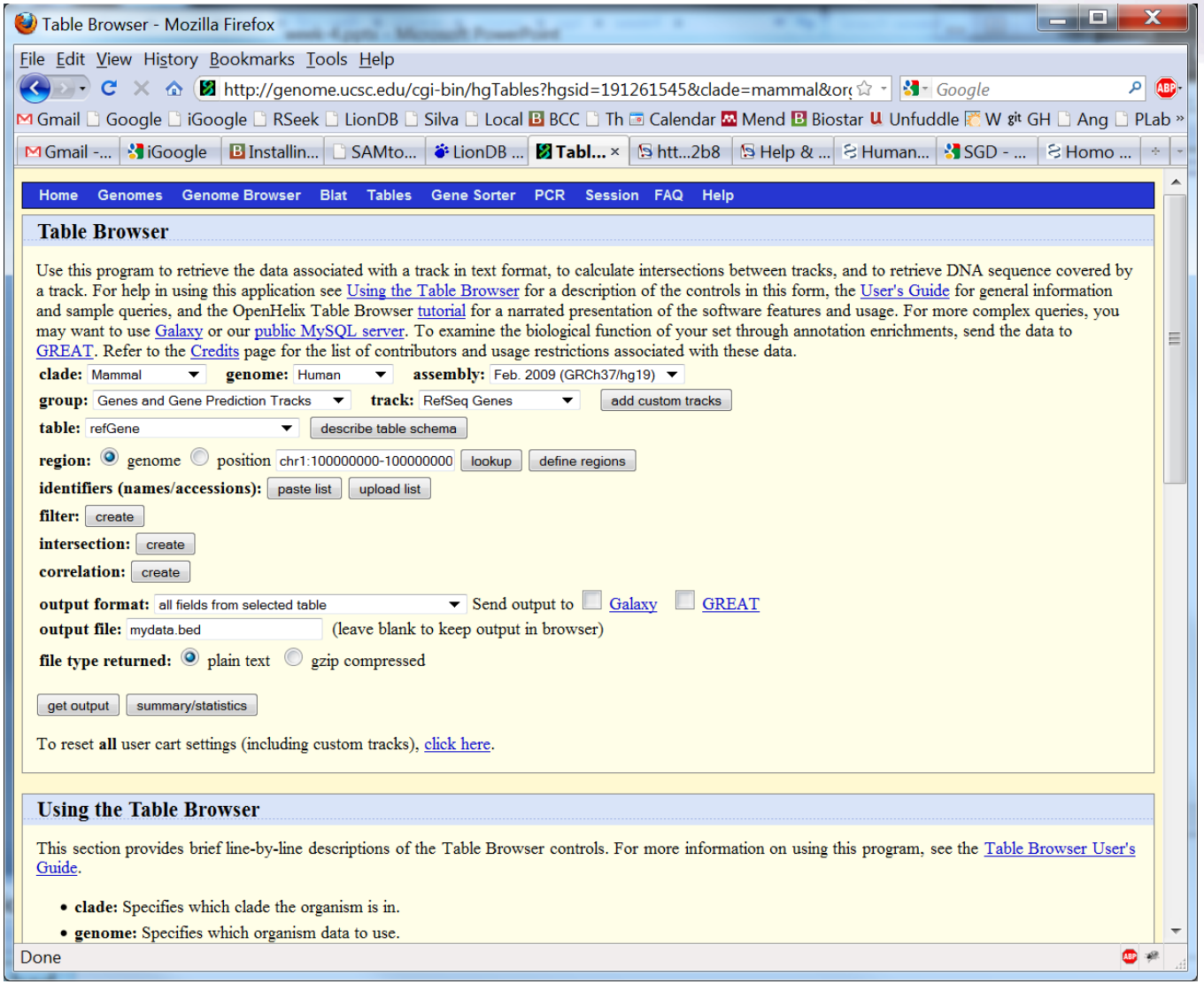
- An interval file may or may not be sorted
- Usually that means sorted by the start coordinates
- Some operations require that the files be sorted!

Where to get genomic features

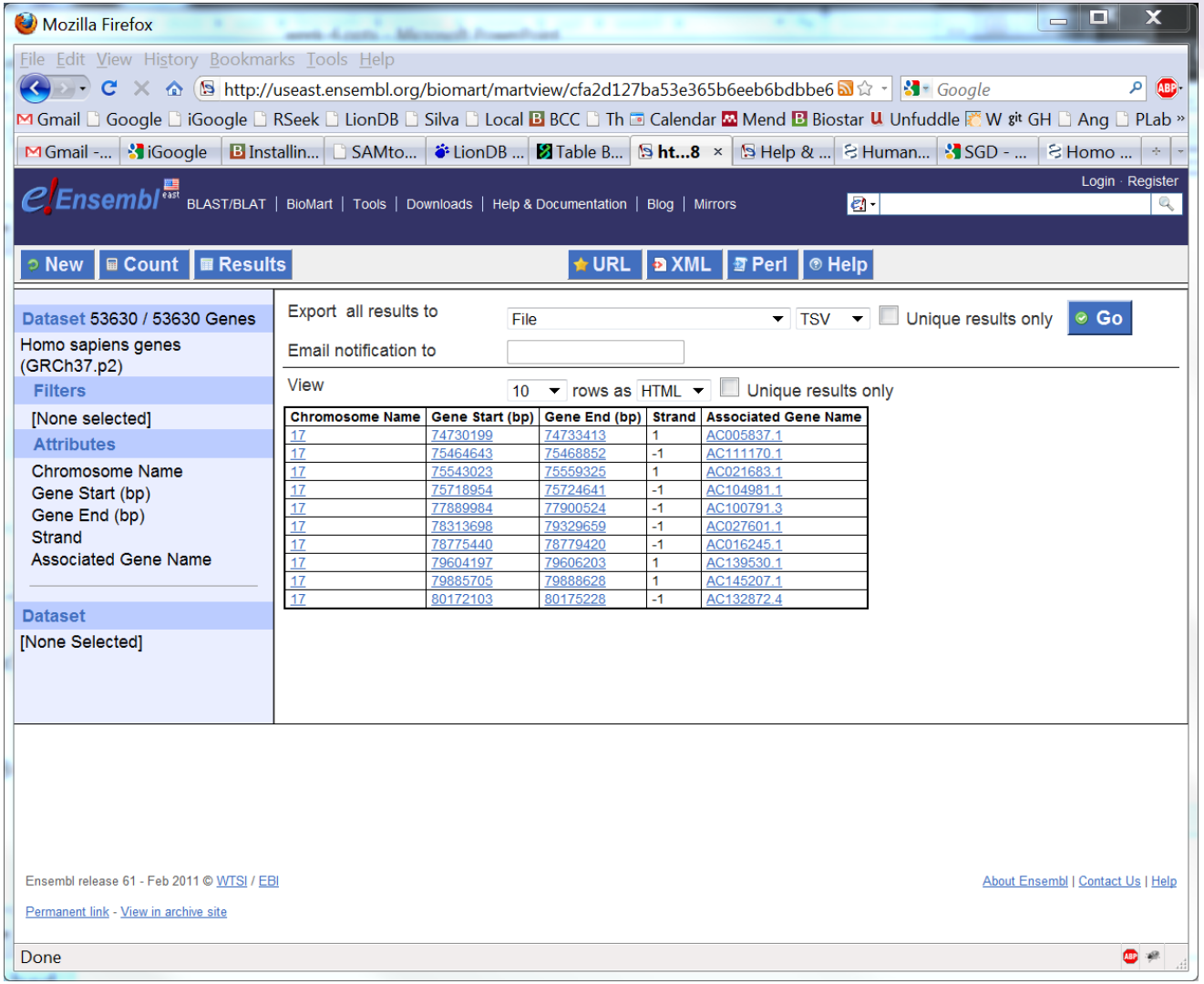
- UCSC: http://genome.ucsc.edu/
- Ensembl/Biomart: http://www.ensembl.org/

Numerous custom databases tuned for a certain organism or disease.

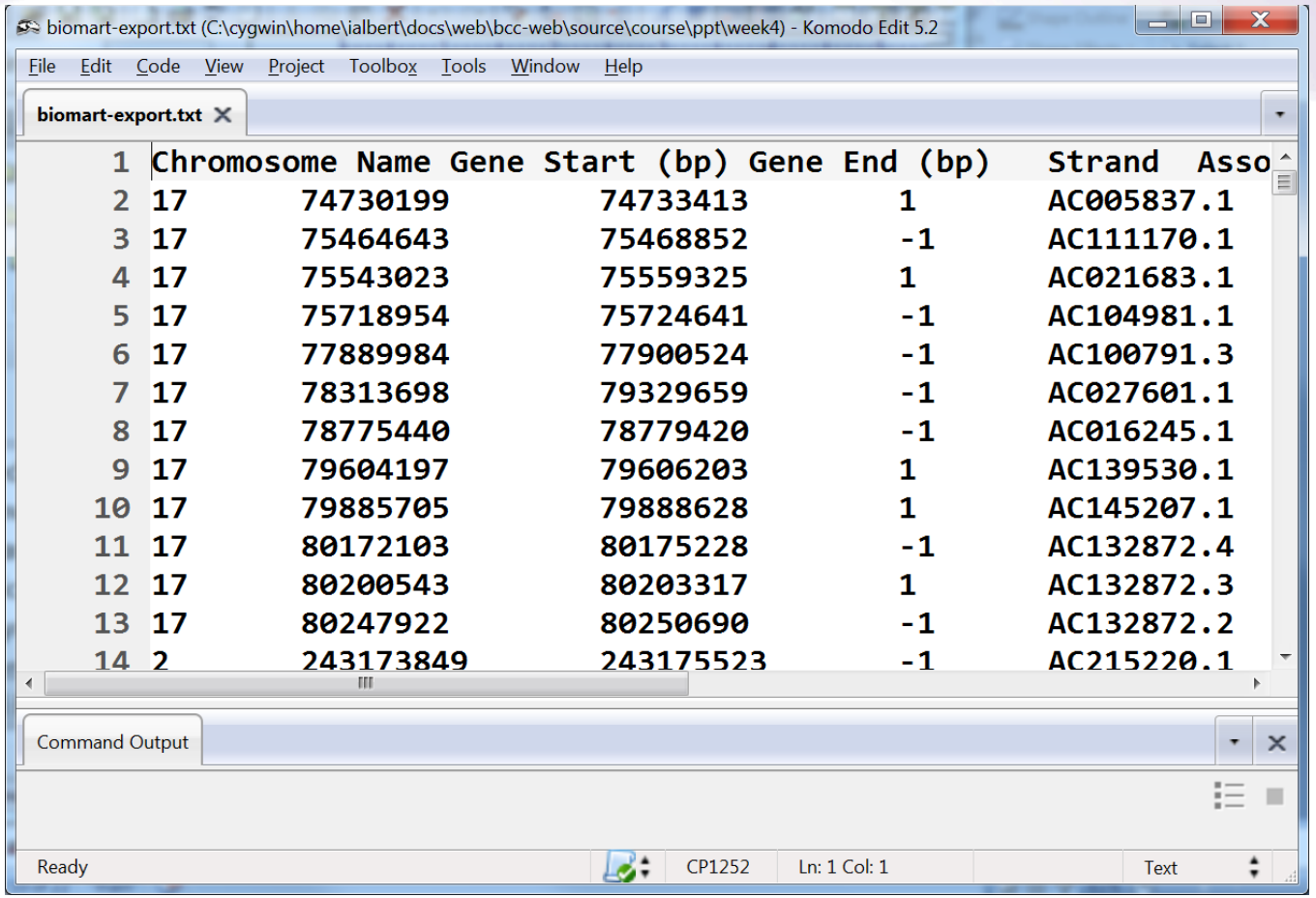
Exporting data from the UCSC genome browser



Exporting data from BioMart



Sometimes we cannot get the data in the exact form we needed

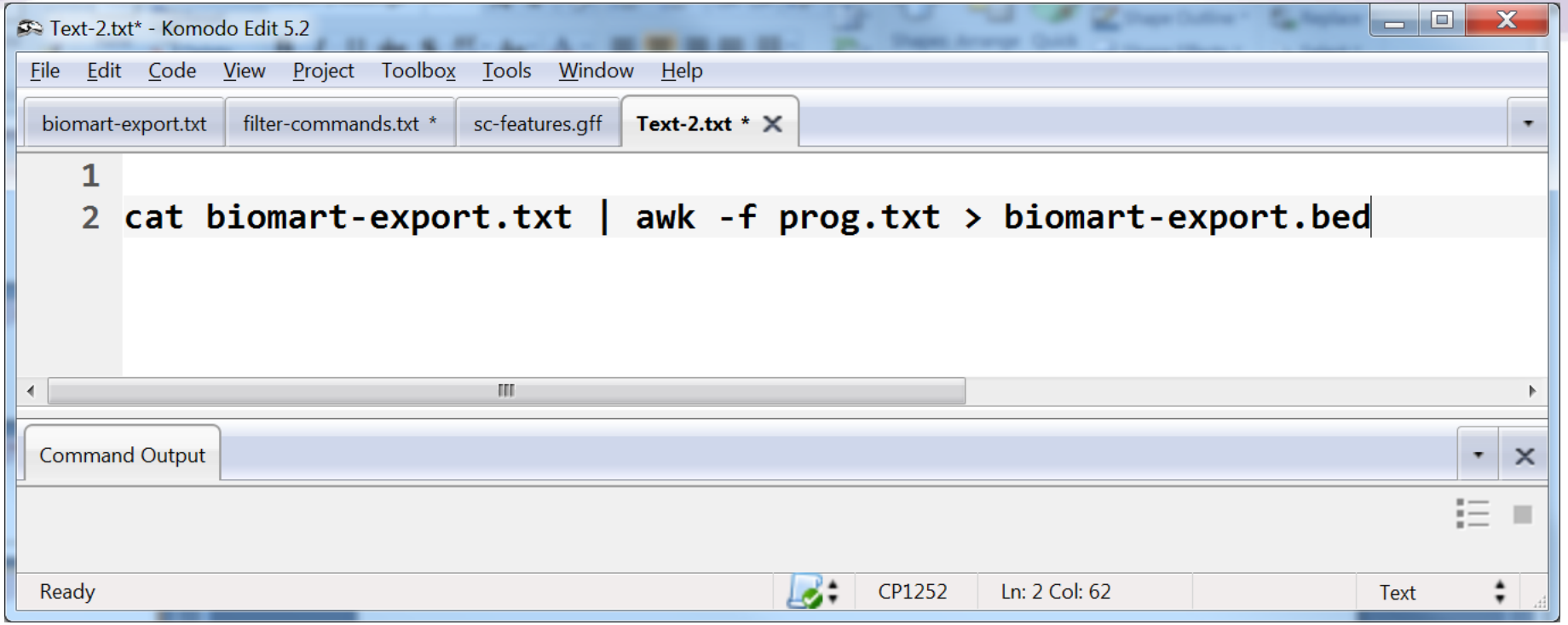


	Chromosome Name	Gene Start (bp)	Gene End (bp)	Strand	Asso
2	17	74730199	74733413	1	AC005837.1
3	17	75464643	75468852	-1	AC111170.1
4	17	75543023	75559325	1	AC021683.1
5	17	75718954	75724641	-1	AC104981.1
6	17	77889984	77900524	-1	AC100791.3
7	17	78313698	79329659	-1	AC027601.1
8	17	78775440	78779420	-1	AC016245.1
9	17	79604197	79606203	1	AC139530.1
10	17	79885705	79888628	1	AC145207.1
11	17	80172103	80175228	-1	AC132872.4
12	17	80200543	80203317	1	AC132872.3
13	17	80247922	80250690	-1	AC132872.2
14	2	243173849	243175523	-1	AC215220.1

this is not a BED file although it is similar to it – you may not be able to transform it yourself

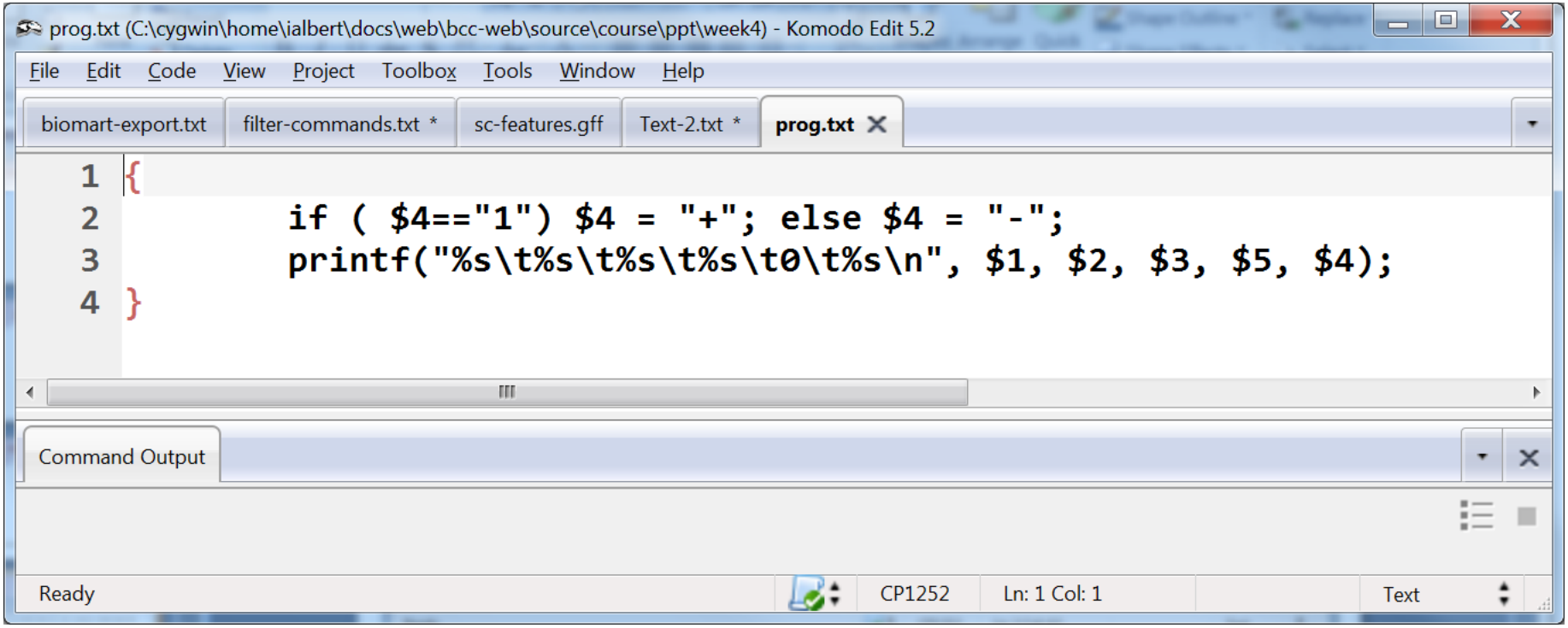
it is a lot easier to find help to transform this file into a BED format than finding it in BED format in the first place

A possible solution



where someone else gives you the content of **prog.txt**

This is what it a prog.txt could look like



You don’t need to know how to make this.
This is (should be) very easy for someone that hired to do programming.

Transition into the tutorial

- Peak calling tutorial & Cis-regulatory element annotation systems

- BEDTools tutorials