Bioinformatics Workshop

Applied Bioinformatics for Genomics II (BIOL.5625.01)

Chris Miller, Ph.D.
Washington University in St. Louis

Bioinformatics Workshop 2025-2026

(aka bfx-workshop)

Applied Bioinformatics for Genomics II

(aka BIOL.5625.01)

Applied Bioinformatics for Genomics II

Course: BIOL 5624

1 Credit Hour DBBS course

- **50% grade**: Attendance
  - 75% (9 lectures) must be in person
  - 3 can be viewed via recordings

- **50% grade**: Assignments
  - choose 8 of the 10 assignments
  - due by the end of the second Friday after the lecture

Register for BFX

https://redcap.link/BFX2025

icts-precisionhealth.wustl.edu

johnegarza@wustl.edu

WHO THE HELL ARE YOU?
NETFLIX

Who we are, and why you should trust us

Chris Miller, Ph.D.

Course Director
Associate Professor
Division of Oncology

John Garza

Course Coordinator/TA
Bioinformatics/Genome Analytics
Programmer

20 years of experience in
Bioinformatics and Computational Biology

Other Lecturers/Organizers include:

Jason Walker
Obi Griffith
Jennifer Foltz

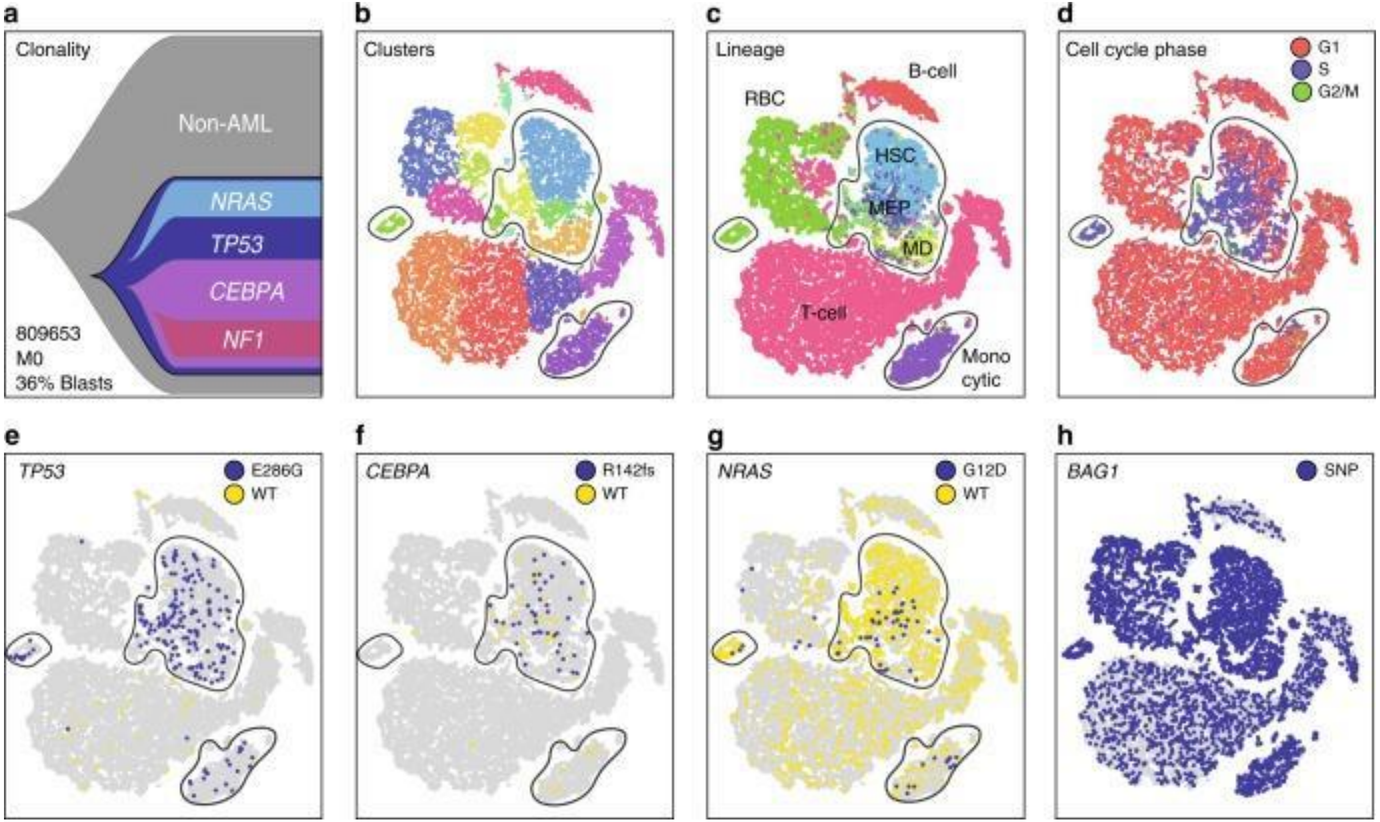
Juan Macias
Brigida Rusconi

Why learn bioinformatics?

- Biology is now a quantitative discipline - especially genomics

Why learn bioinformatics?

- Biology is now a quantitative discipline - especially genomics
- Skills in programming, statistics, and visualization help you get the most out of your data


a
Clonality
Non-AML
NRAS
TP53
CEBPA
NF1
809653
M0
36% Blasts
b
Clusters
c
Lineage
B-cell
RBC
HSC
MEP
MD
T-cell
Mono
cytic
d
Cell cycle phase
G1
S
G2/M
e
TP53
E286G
WT
f
CEBPA
R142fs
WT
g
NRAS
G12D
WT
h
BAG1
SNP

People who need complex data analysis

People who know how to do complex data analysis

https://hellogiggles.com/news/how-many-people-attend-coachella/
https://www.nytimes.com/2020/07/02/theater/germany-theater-coronavirus.html

Why learn bioinformatics?

- Biology is now a quantitative discipline - especially genomics
- Skills in programming, statistics, and visualization help you get the most out of your data
- We’re aiming to teach you the theory and practice of computational biology, with a focus on genomics but lessons that apply broadly

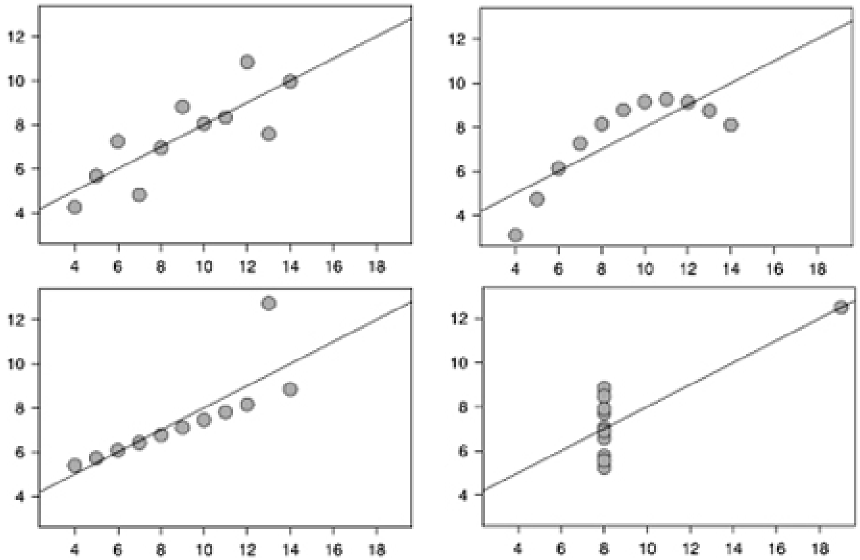
Goals:

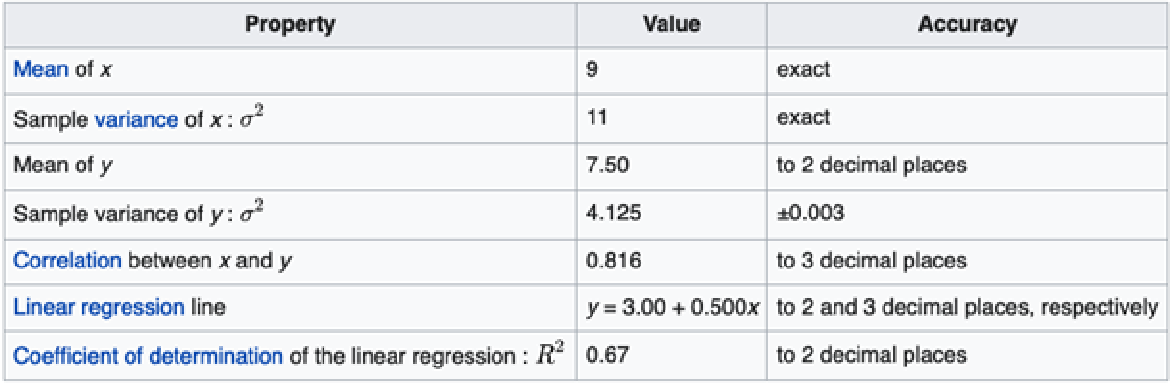
- To empower you to improve and expedite your research

- To expose you to new ideas and techniques that may advance your research program

Don’t trust your data

Trusting your data



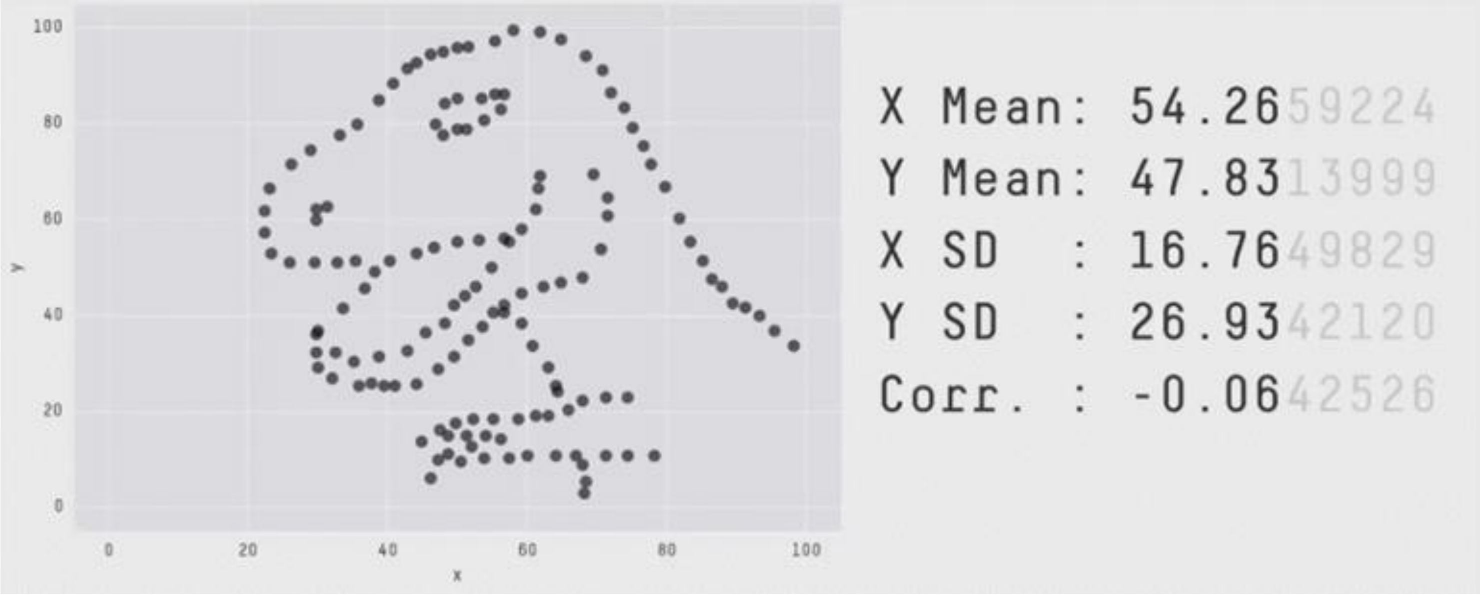


Property	Value	Accuracy
Mean of *x*	9	exact
Sample variance of *x* : σ^2	11	exact
Mean of *y*	7.50	to 2 decimal places
Sample variance of *y* : σ^2	4.125	±0.003
Correlation between *x* and *y*	0.816	to 3 decimal places
Linear regression line	$y = 3.00 + 0.500x$	to 2 and 3 decimal places, respectively
Coefficient of determination of the linear regression : R^2	0.67	to 2 decimal places

https://commons.wikimedia.org/wiki/File:Anscombe%27s_quartet_3_cropped.jpg

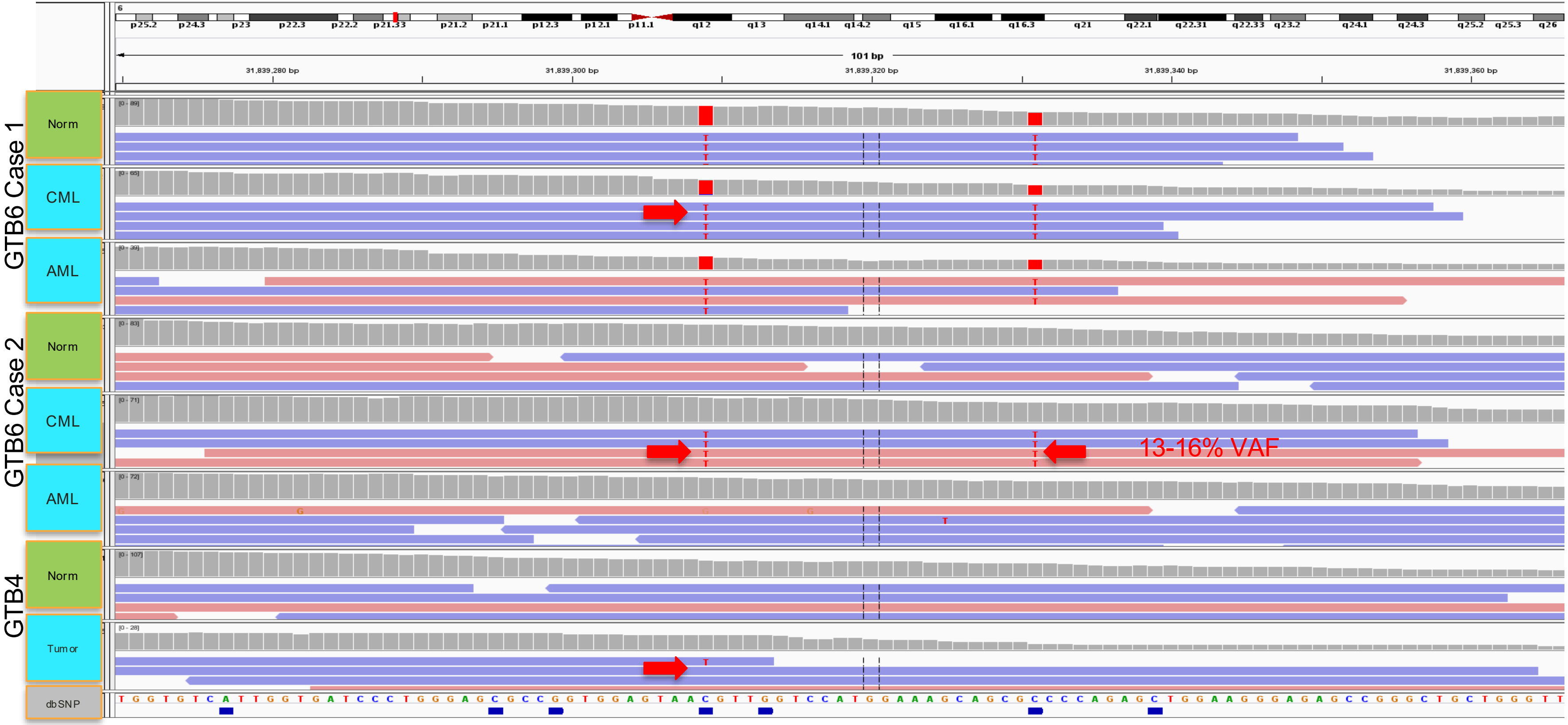
Summary statistics are dangerous

- Visualize your data!
- A picture is worth a thousand p-values



```
X Mean: 54.2659224
Y Mean: 47.8313999
X SD  : 16.7649829
Y SD  : 26.9342120
Corr. : -0.0642526
```

Contamination of CML samples


6
p25.2 p24.3 p23 p22.3 p22.2 p21.33 p21.2 p21.1 p12.3 p12.1 p11.1 q12 q13 q14.1 q14.2 q15 q16.1 q16.3 q21 q22.1 q22.31 q22.33 q23.2 q24.1 q24.3 q25.2 q25.3 q26
101 bp
31,839,280 bp
31,839,300 bp
31,839,320 bp
31,839,340 bp
31,839,360 bp
GTB6 Case 1
Norm
[0 - 89]
CML
[0 - 65]
AML
[0 - 39]
GTB6 Case 2
Norm
[0 - 83]
CML
[0 - 71]
AML
[0 - 72]
GTB4
Norm
[0 - 107]
Tumor
[0 - 28]
dbSNP
T G G T G T C A T T G G T G A T C C C T G G G A G C G C C G G T G G A G T A A C G T T G G T C C A T G G A A A G C A G C G C C C C A G A G C T G G A A G G G A G A G C C G G G C T G C T G G G T T

13-16% VAF

Watch out!

- Computational analyses require controls too!
- Look at the data and understand it's limitations!
- Don't assume that the data is clean – prove to yourself that it is!

Expectations:

- Check the prerequisites from fall weeks 1-3. Install the software, be familiar with the unix command line, know how to use docker to launch analyses
  - https://github.com/genome/bfx-workshop/tree/master/lectures/week_01

- Many of you are new to computational analysis – *ask questions*!

- Work hard, follow along, and get your money’s worth from this course

- The folks teaching and the TAs all know their stuff, *ask questions*!

Course Structure:

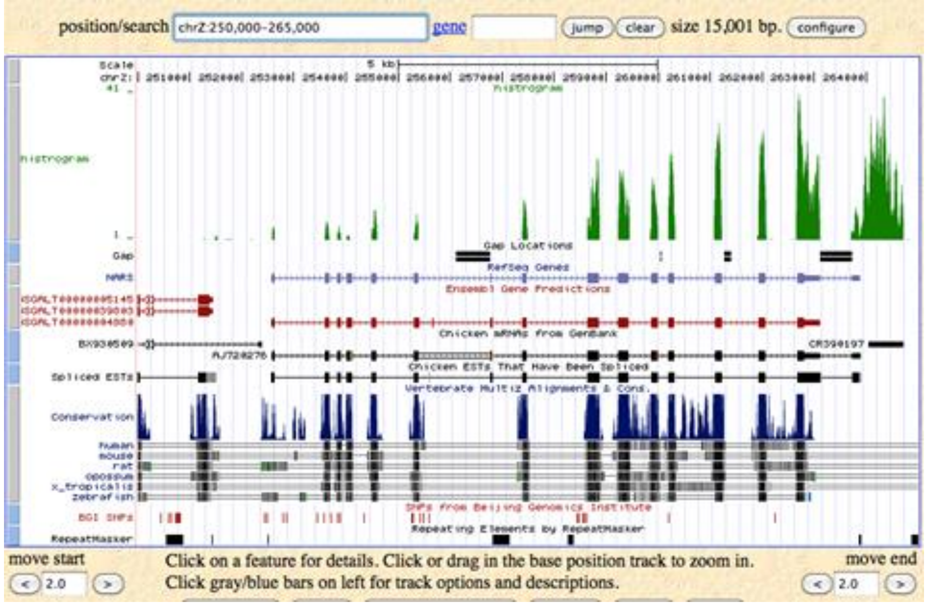
- Weekly lecture introducing topic
- Practical exercise allowing you to apply that knowledge
- https://github.com/genome/bfx-workshop
- ICTS Slack instance: #bfx-workshop channel

Genome arithmetic and bedtools

some slides adapted from :
Aaron Quinlan's Applied Computational Genomics course https://github.com/quinlan-lab/applied-computational-genomics
Griffith Lab's RNAbio course: https://rnabio.org/

What is a genome interval?

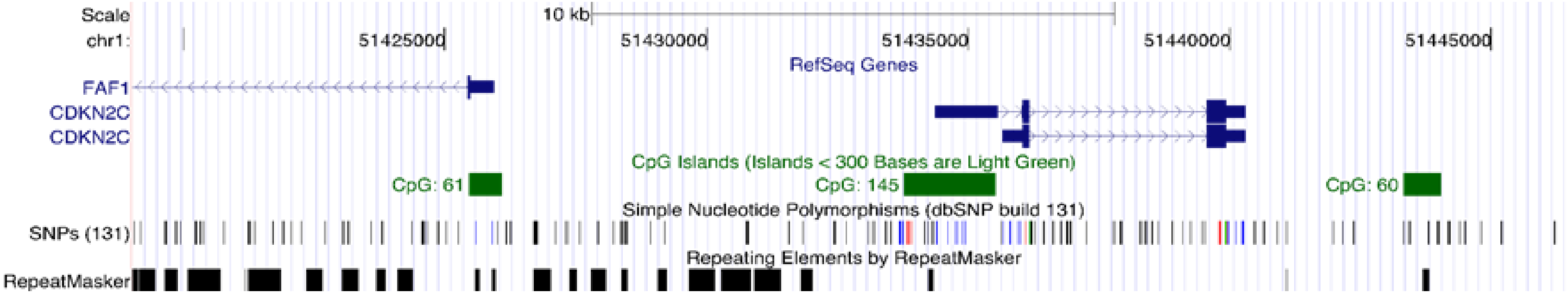
- Genes: exons, introns, UTRs, promoters (BED, GFF, GTF)
- Conservation (BEDGRAPH)
- Genetic variation (VCF)
- Sequence alignments (BAM)
- Transcription factor binding sites (BED, BEDGRAPH)
- CpG islands (BED)
- Segmental duplications (BED)
- Chromatin annotations (BED)
- Gene expression data (WIG, BIGWIG, BEDGRAPH)
- **<u>Your own observations: put them in context</u>**



position/search chrZ:250,000-265,000 size 15,001 bp.

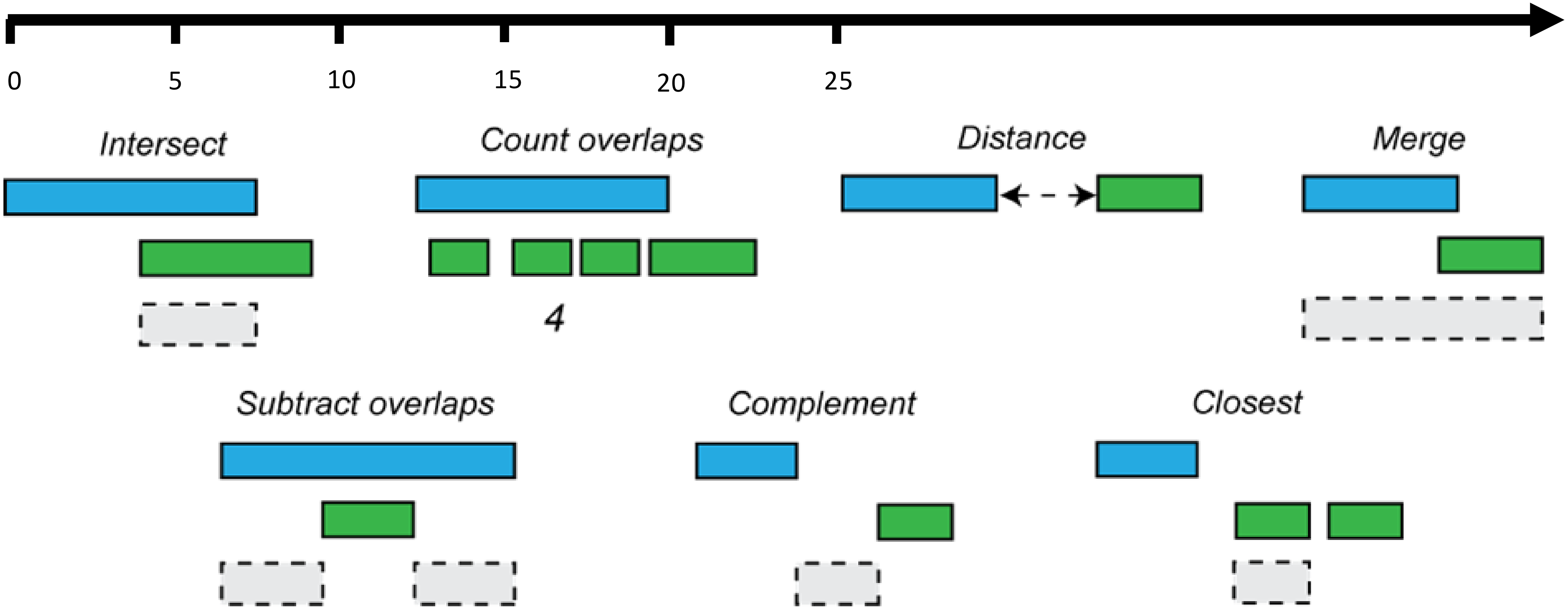
Click on a feature for details. Click or drag in the base position track to zoom in.
Click gray/blue bars on left for track options and descriptions.

Genome intervals

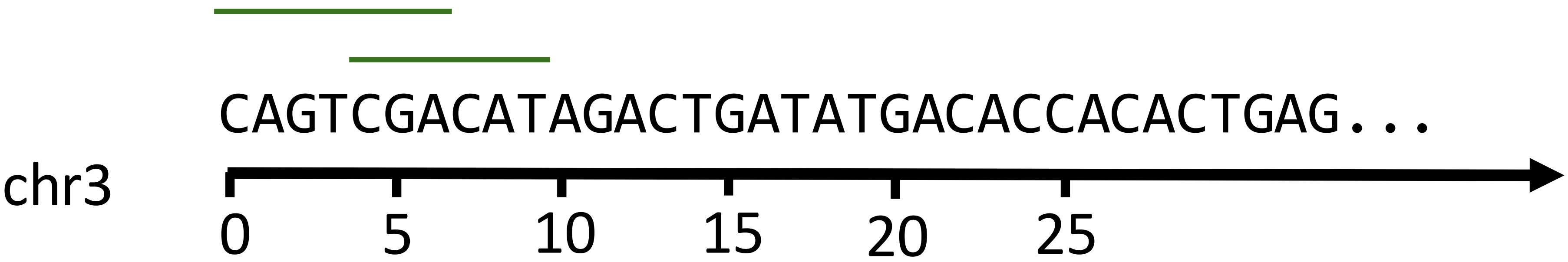

Scale
10 kb
chr1:
51425000
51430000
51435000
51440000
51445000
RefSeq Genes
FAF1
CDKN2C
CDKN2C
CpG Islands (Islands < 300 Bases are Light Green)
CpG: 61
CpG: 145
CpG: 60
Simple Nucleotide Polymorphisms (dbSNP build 131)
SNPs (131)
Repeating Elements by RepeatMasker
RepeatMasker

Genome arithmetic: the method of comparing, contrasting, and gaining insight using multiple genome interval files

Genome arithmetic operations

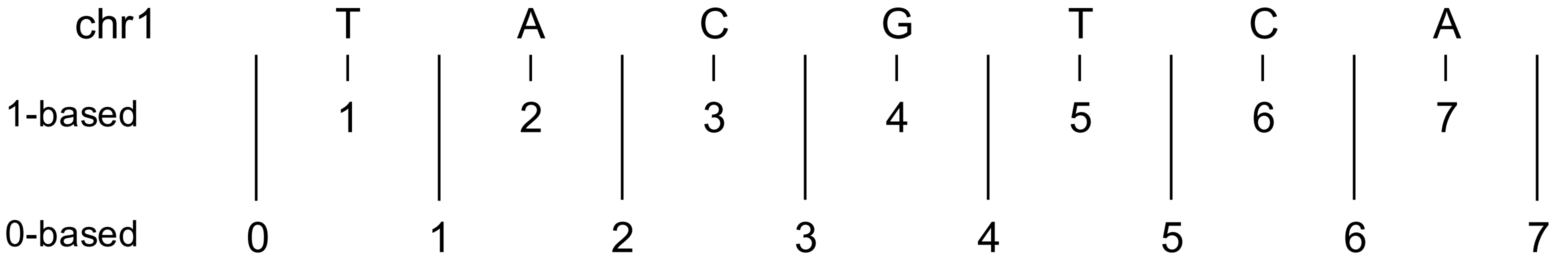


Genome arithmetic depends upon the genome coordinate system



```
chr3    0    7
chr3    4    10
```

Genomic coordinates – 1 vs 0 based



chr1 T A C G T C A

1-based 1 2 3 4 5 6 7

0-based 0 1 2 3 4 5 6 7

	1-based	0-based
Indicate a single nucleotide	chr1:4-4 G	chr1:3-4 G
Indicate a range of nucleotides	chr1:2-4 ACG	chr1:1-4 ACG

- 1-based : Single nucleotides, variant positions, or ranges are specified directly by their corresponding nucleotide numbers
  - GFF, SAM, VCF, Ensembl browser, …

- 0-based: Single nucleotides, variant positions, or ranges are specified by the coordinates that flank them
  - BED, UCSC browser, …

Genome builds

Reference Genome builds

Current human: GRCh38, hg38, b38

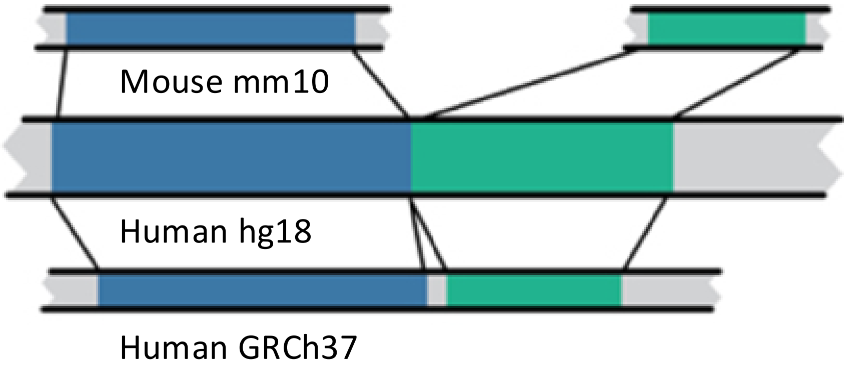
alternates: GRCh38v2_ccdg,
GRCh38_full_analysis_set_plus_decoy_hla

Previous human: GRCh37, hg19, b37

Current mouse: GRCm39
Previous mouse: GRCm38, mm10

New human assembly: T2T-CHM13, pan-genomes

Lift-over



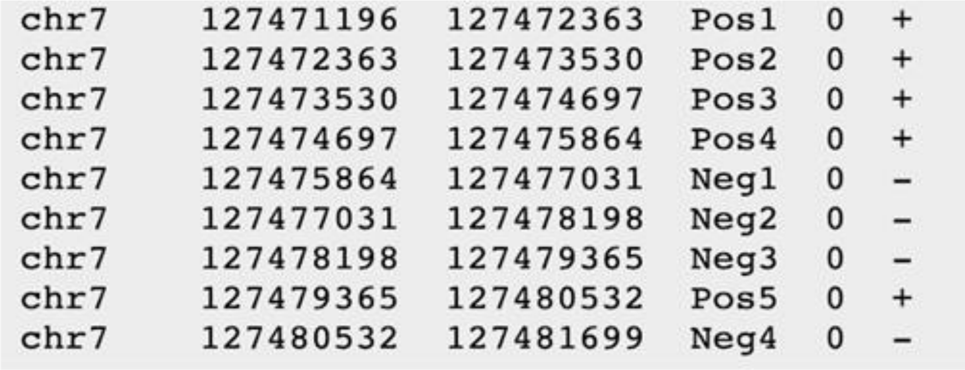
Intervals are often represented in the BED format

- There are several flavors of BED format: BED3, BED4, BED6, BED8, etc

- First 3 fields always required: **chr, start, stop**

- Followed by up to 9 additional optional fields:

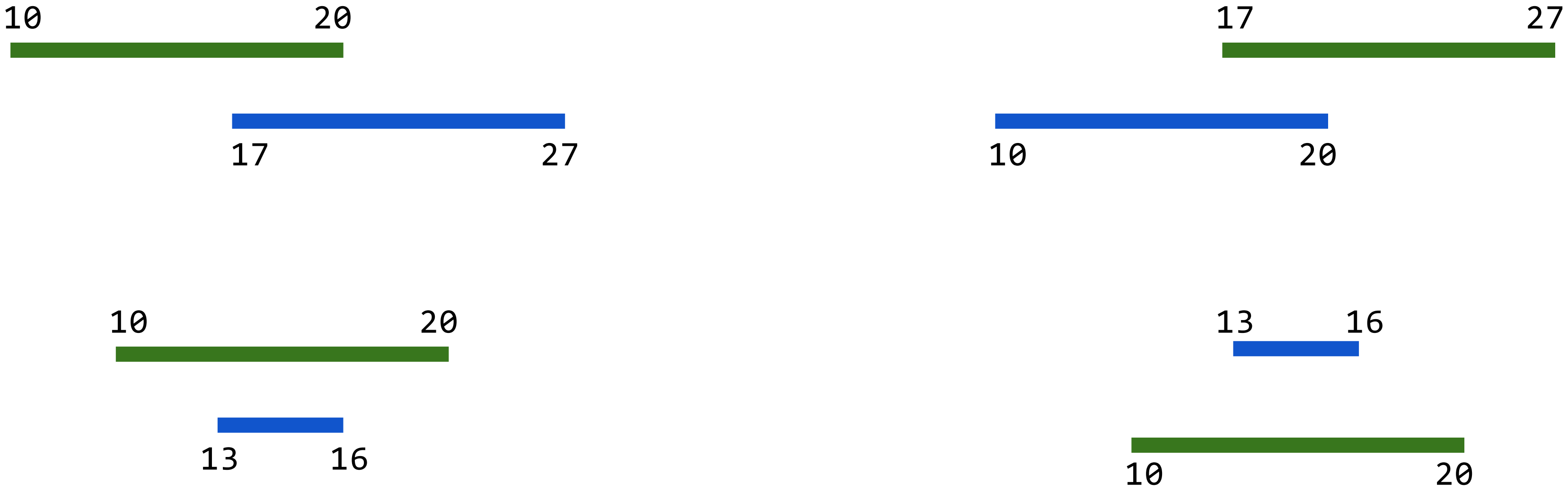
name, score, strand, thickStart, thickEnd, itemRGB, blockCount, blockSizes, blockStarts



```
chr7    127471196  127472363  Pos1  0  +
chr7    127472363  127473530  Pos2  0  +
chr7    127473530  127474697  Pos3  0  +
chr7    127474697  127475864  Pos4  0  +
chr7    127475864  127477031  Neg1  0  -
chr7    127477031  127478198  Neg2  0  -
chr7    127478198  127479365  Neg3  0  -
chr7    127479365  127480532  Pos5  0  +
chr7    127480532  127481699  Neg4  0  -
```

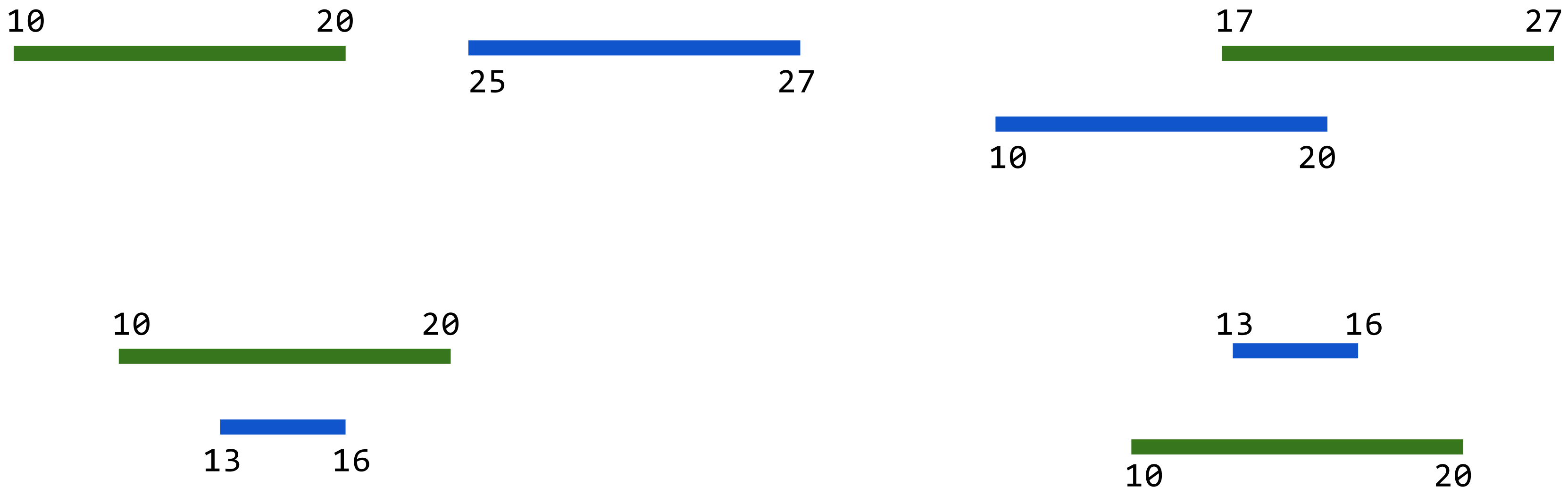
- Bed files are always 0-based! (some lookalikes may not be)

Do two intervals intersect (overlap)?



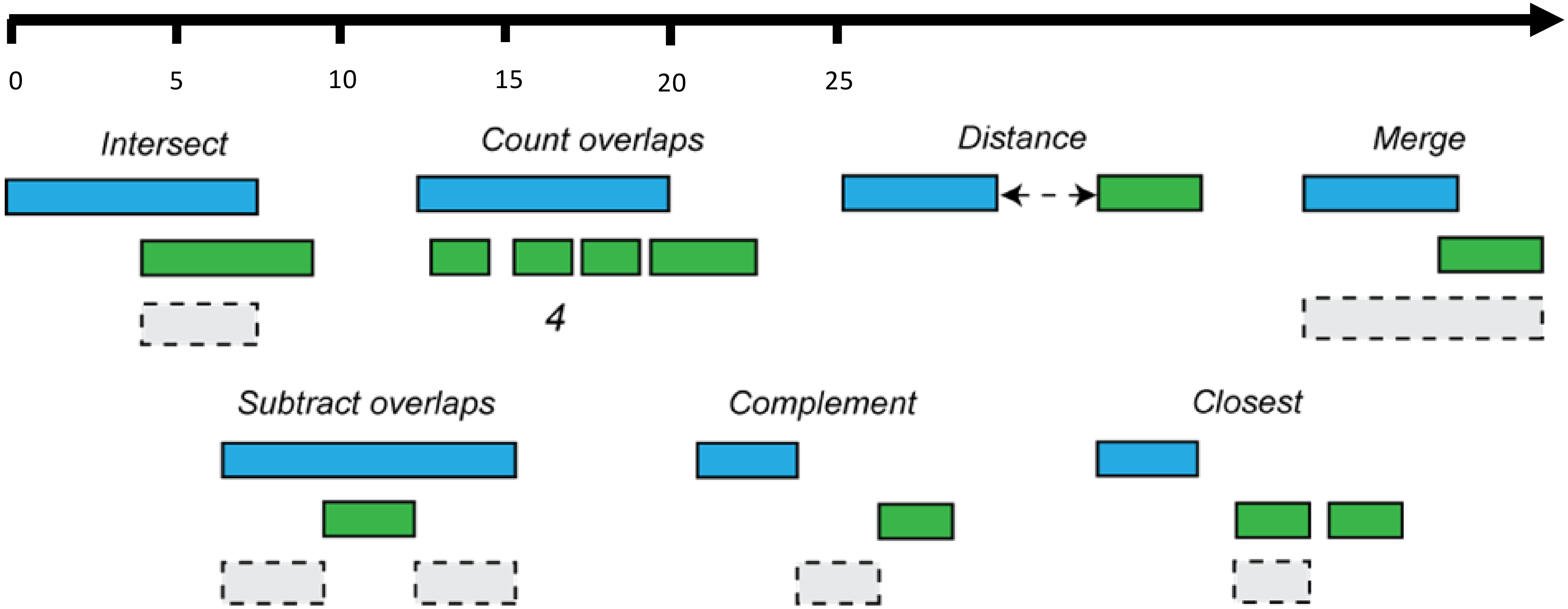
```
if ((a.start <= b.start and a.end >= b.start) or
    (b.start <= a.start and b.end >= a.start) or
    (a.start <= b.start and a.end >= b.end)   or
    (b.start <= a.start and b.end >= a.end))
{
  INTERSECTION!!!
}
else NADA!!!
```

Do two intervals intersect (overlap)? A simpler way.

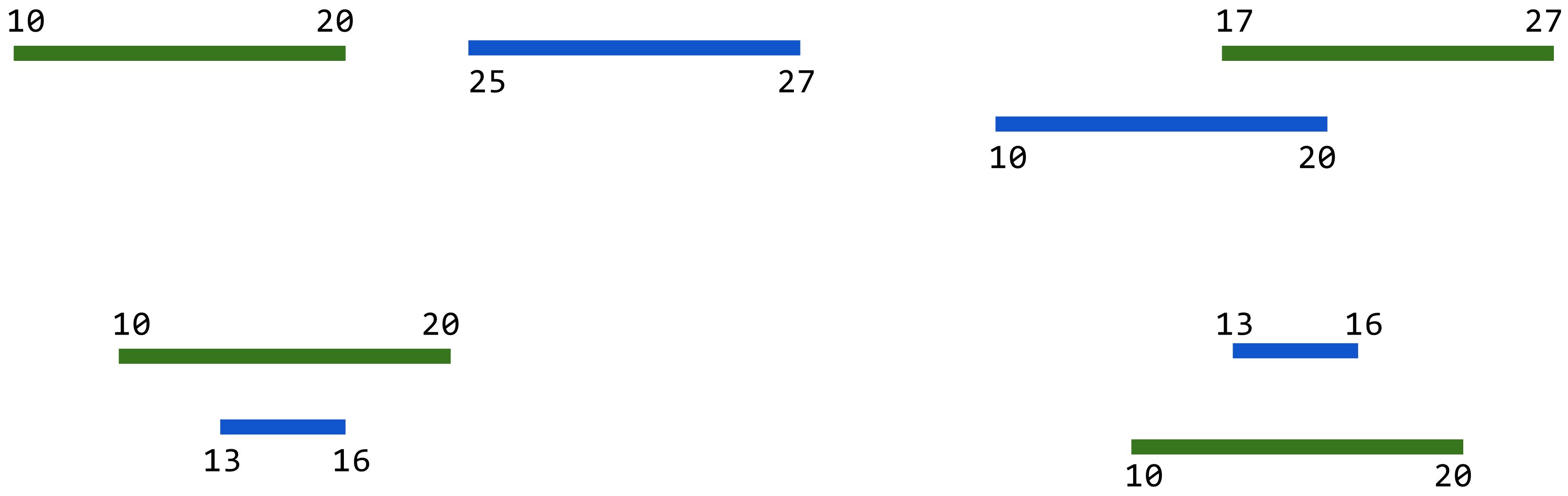


```
I = min(a.end, b.end) - max(a.start, b.start)

         if I > 0, intersection,
   if I <= 0, distance between the intervals

        = min(20, 27) - max(10, 17)
                = 20-17 = 3
```

Genome arithmetic operations

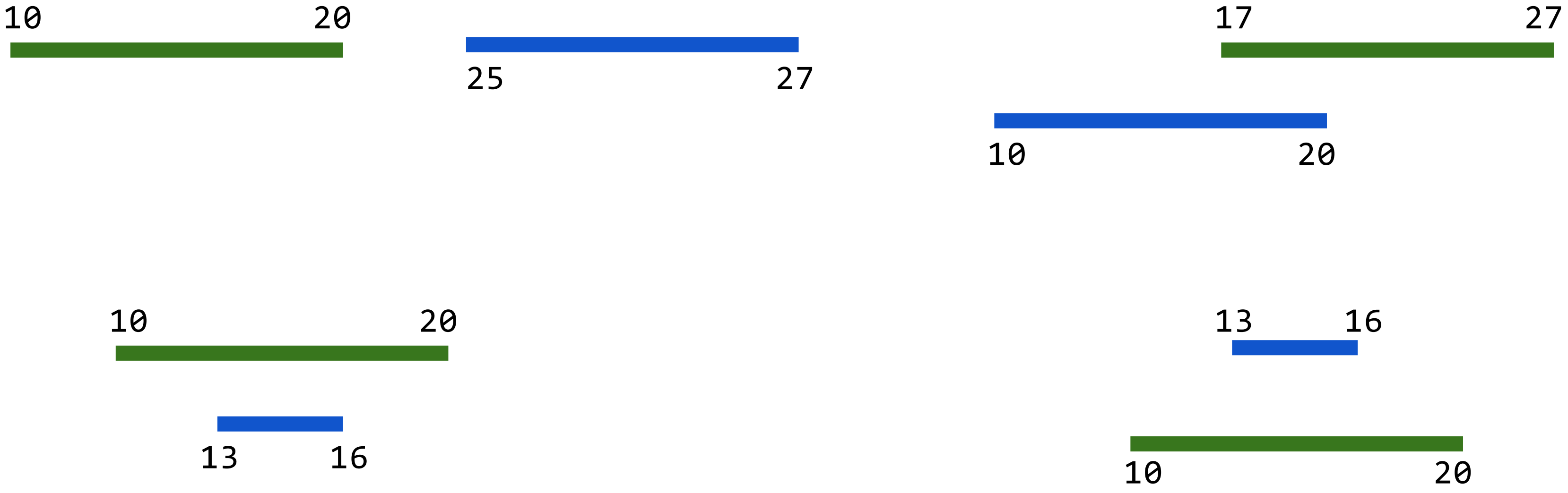


Do two intervals intersect (overlap)? A simpler way.



```
I = min(a.end, b.end) - max(a.start, b.start)

if I > 0, intersection,
if I <= 0, distance between the intervals

= min(20, 27) - max(10, 17)
= 20-17 = 3
```

Do two intervals intersect (overlap)? A simpler way.



```
bedtools intersect [options] [files]
```

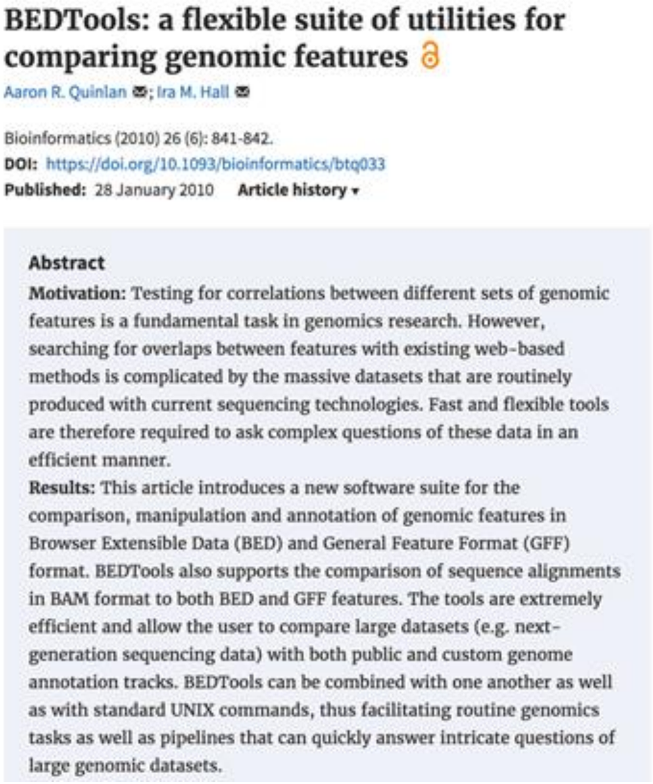
Manipulation of SAM/BAM and BED files

- Several tools are used ubiquitously in sequence analysis to manipulate these files

- SAM/BAM files
    - samtools
    - bamtools
    - Picard

- BED files
    - bedtools
    - bedops

Bedtools: a swiss army knife for genome analysis



BEDTools: a flexible suite of utilities for comparing genomic features

Aaron R. Quinlan ✉; Ira M. Hall ✉

Bioinformatics (2010) 26 (6): 841-842.
DOI: https://doi.org/10.1093/bioinformatics/btq033
Published: 28 January 2010 **Article history ▾**

Abstract

Motivation: Testing for correlations between different sets of genomic features is a fundamental task in genomics research. However, searching for overlaps between features with existing web-based methods is complicated by the massive datasets that are routinely produced with current sequencing technologies. Fast and flexible tools are therefore required to ask complex questions of these data in an efficient manner.

Results: This article introduces a new software suite for the comparison, manipulation and annotation of genomic features in Browser Extensible Data (BED) and General Feature Format (GFF) format. BEDTools also supports the comparison of sequence alignments in BAM format to both BED and GFF features. The tools are extremely efficient and allow the user to compare large datasets (e.g. next-generation sequencing data) with both public and custom genome annotation tracks. BEDTools can be combined with one another as well as with standard UNIX commands, thus facilitating routine genomics tasks as well as pipelines that can quickly answer intricate questions of large genomic datasets.

Papers:

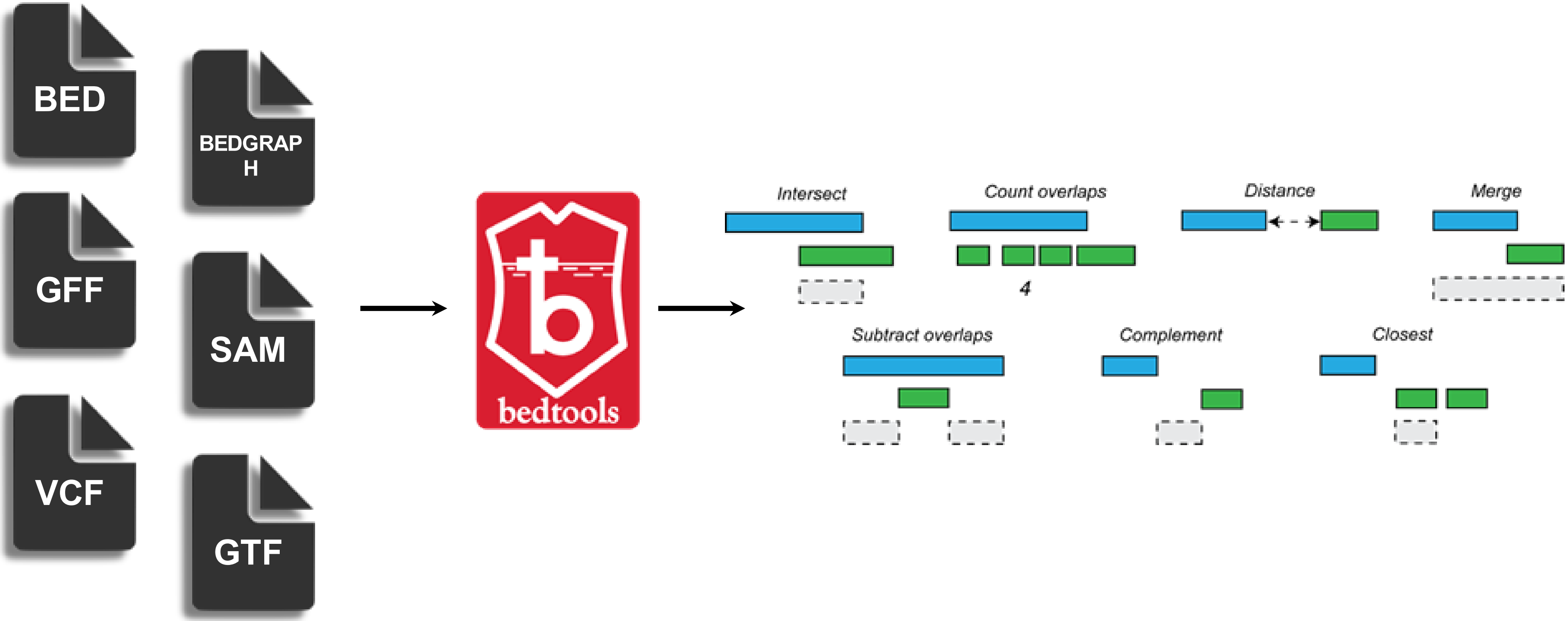
https://doi.org/10.1093/bioinformatics/btq033
DOI: 10.1002/0471250953.bi1112s47

Documentation:
http://bedtools.readthedocs.io/en/latest/

Code:
https://github.com/arq5x/bedtools2

http://bedtools.readthedocs.io/en/latest/

Supports most interval formats & handles diff. coordinate systems



http://bedtools.readthedocs.io/en/latest/

Bedtools: example analyses

- Closest gene to a ChIP-seq peak.
- Is my latest discovery novel?
- Is there strand bias in my data?
- How many genes does this deletion affect?
- Where did I fail to collect sequence coverage?
- Is my favorite feature significantly correlated with some other feature?
- What is the density of variants in "windows" along the genome?

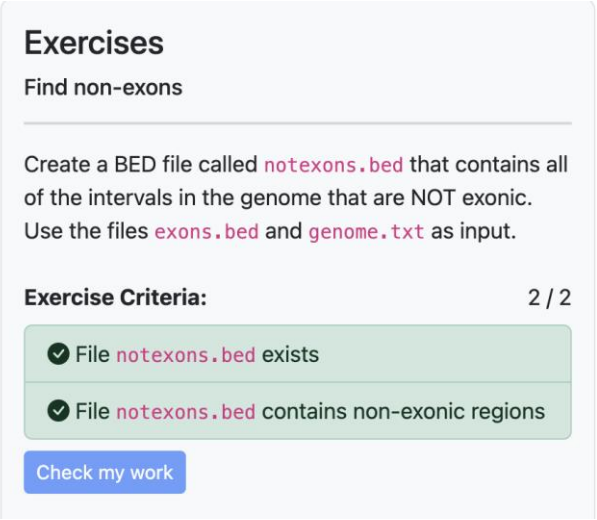
http://bedtools.readthedocs.io/en/latest/

Assignment: work through the bedtools tutorial

https://sandbox.bio/tutorials/bedtools-intro

For-credit students:

Pages 15-19 contain 5 exercises – submit a screenshot of the validated exercise on slide 19 after completing it (like this example from another step):



Exercises

Find non-exons

Create a BED file called `notexons.bed` that contains all of the intervals in the genome that are NOT exonic. Use the files `exons.bed` and `genome.txt` as input.

Exercise Criteria: 2 / 2

✓ File `notexons.bed` exists

✓ File `notexons.bed` contains non-exonic regions

Send it to johnegarza@wustl.edu with "bfx exercise week 13" as the subject