Tutorial 9 – BEDtools and plyranges

MICB405 – BIOINFORMATICS – 2021W-T1

05 NOVEMBER 2021

AXEL HAUDUC

Refresher: BED File

3 mandatory columns

- Sequence (usually a chromosome or fragment)
- Start position
- End Position

Tab-delimited

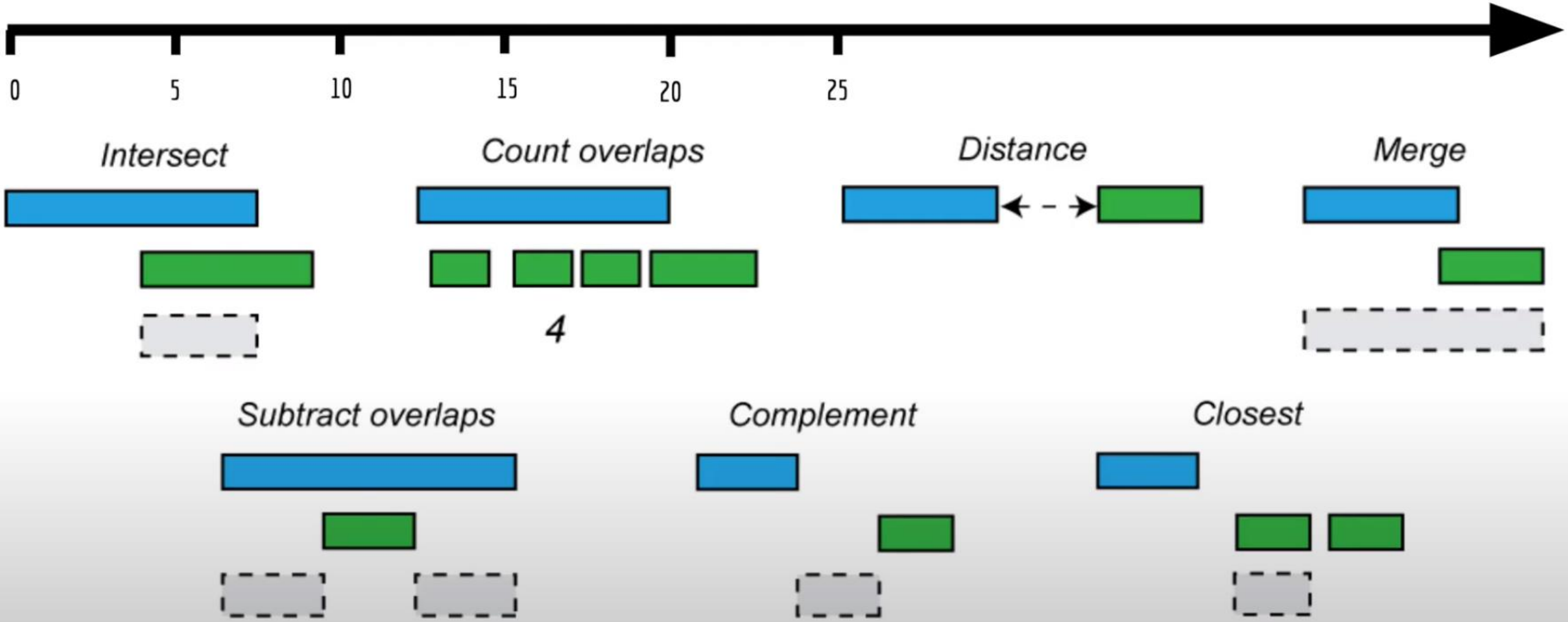
chr1	11873	12227	NR_046018_exon_0_0_chr1_11874_f	0	+
chr1	12612	12721	NR_046018_exon_1_0_chr1_12613_f	0	+
chr1	13220	14409	NR_046018_exon_2_0_chr1_13221_f	0	+
chr1	14361	14829	NR_024540_exon_0_0_chr1_14362_r	0	-
chr1	14969	15038	NR_024540_exon_1_0_chr1_14970_r	0	-
chr1	15795	15947	NR_024540_exon_2_0_chr1_15796_r	0	-
chr1	16606	16765	NR_024540_exon_3_0_chr1_16607_r	0	-
chr1	16857	17055	NR_024540_exon_4_0_chr1_16858_r	0	-
chr1	17232	17368	NR_024540_exon_5_0_chr1_17233_r	0	-
chr1	17605	17742	NR_024540_exon_6_0_chr1_17606_r	0	-

Sequence name	Start	Stop	Feature name	Score	Strand

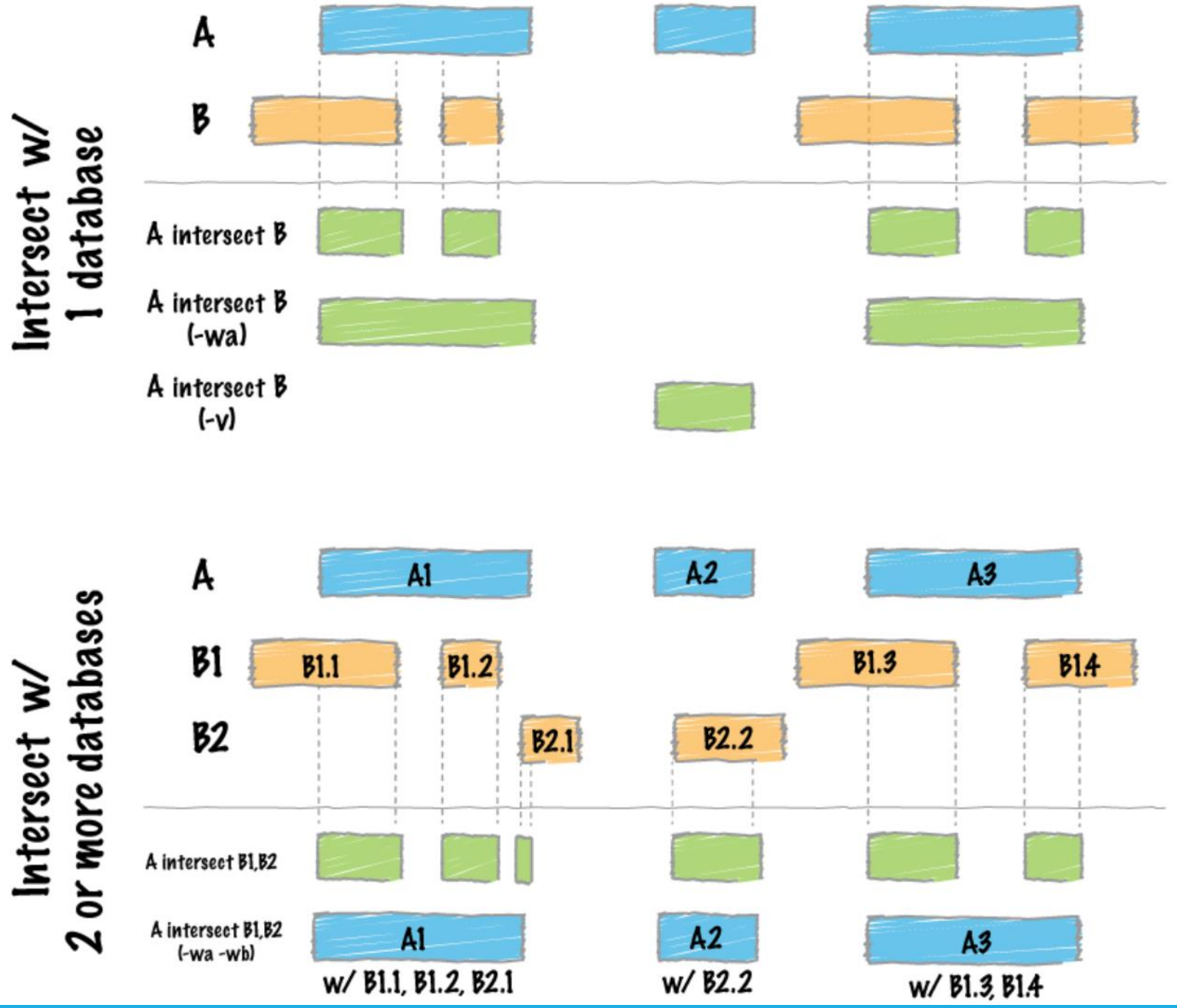
ChIP-seq Peak Calling

ChIP-seq Peak Calling

Genome arithmetic operations



bedtools intersect



bedtools closest

Don’t forget **bedtools sort** !!!

```
bedtools sort –i input.bed > input.sorted.bed
```

Pull out overlapping gene names for mouse Naïve T-cell H3K27ac peaks

```
bedtools intersect \
-a /projects/micb405/analysis/ChIP_tutorial/Naive_H3K27ac_peaks.autosomes.narrowPeak \
-b /projects/micb405/analysis/STAR_tutorial/Mus_musculus.GRCm38.84.chr.autosomes.gtf \
-wa | awk 'BEGIN{OFS="\t"} {print $1, $2, $3, $4, $20}' | tr -d '";'
```

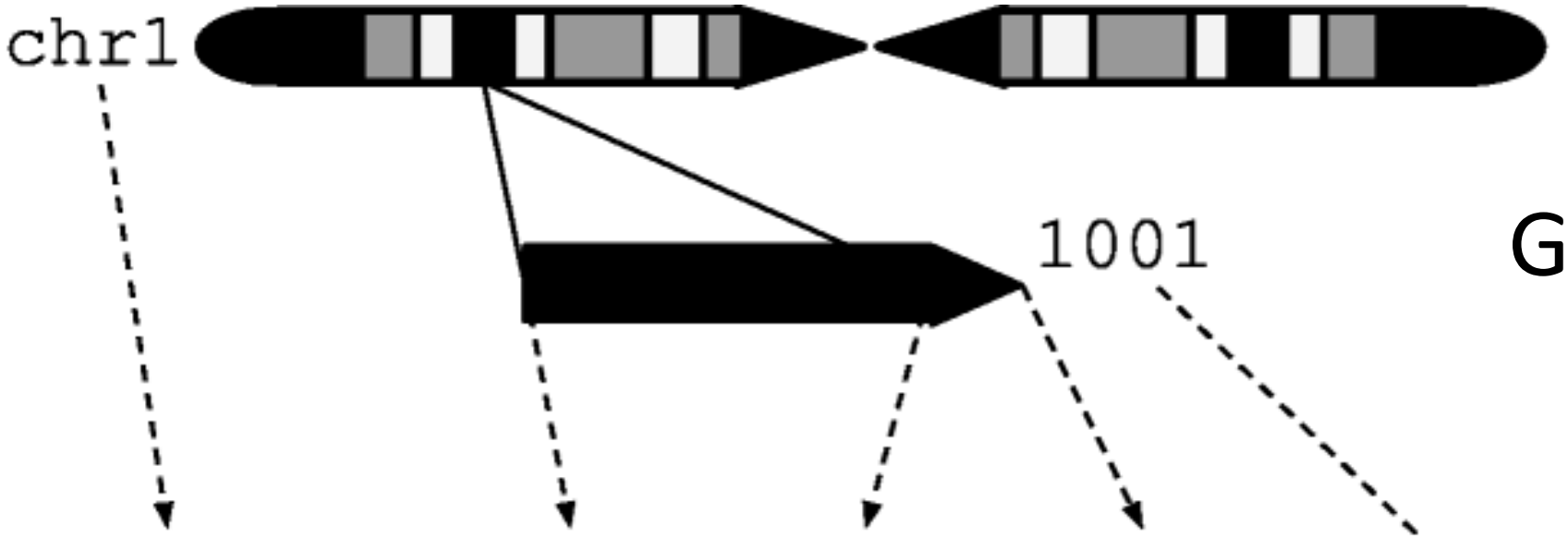
Can get very complicated using purely command line tools…

plyranges

The R/tidyverse response to BEDtools and handling bed files

Utilizes special GenomicRanges objects (“GRanges”) to efficiently perform range arithmetic operations

- These are equivalent to BED files

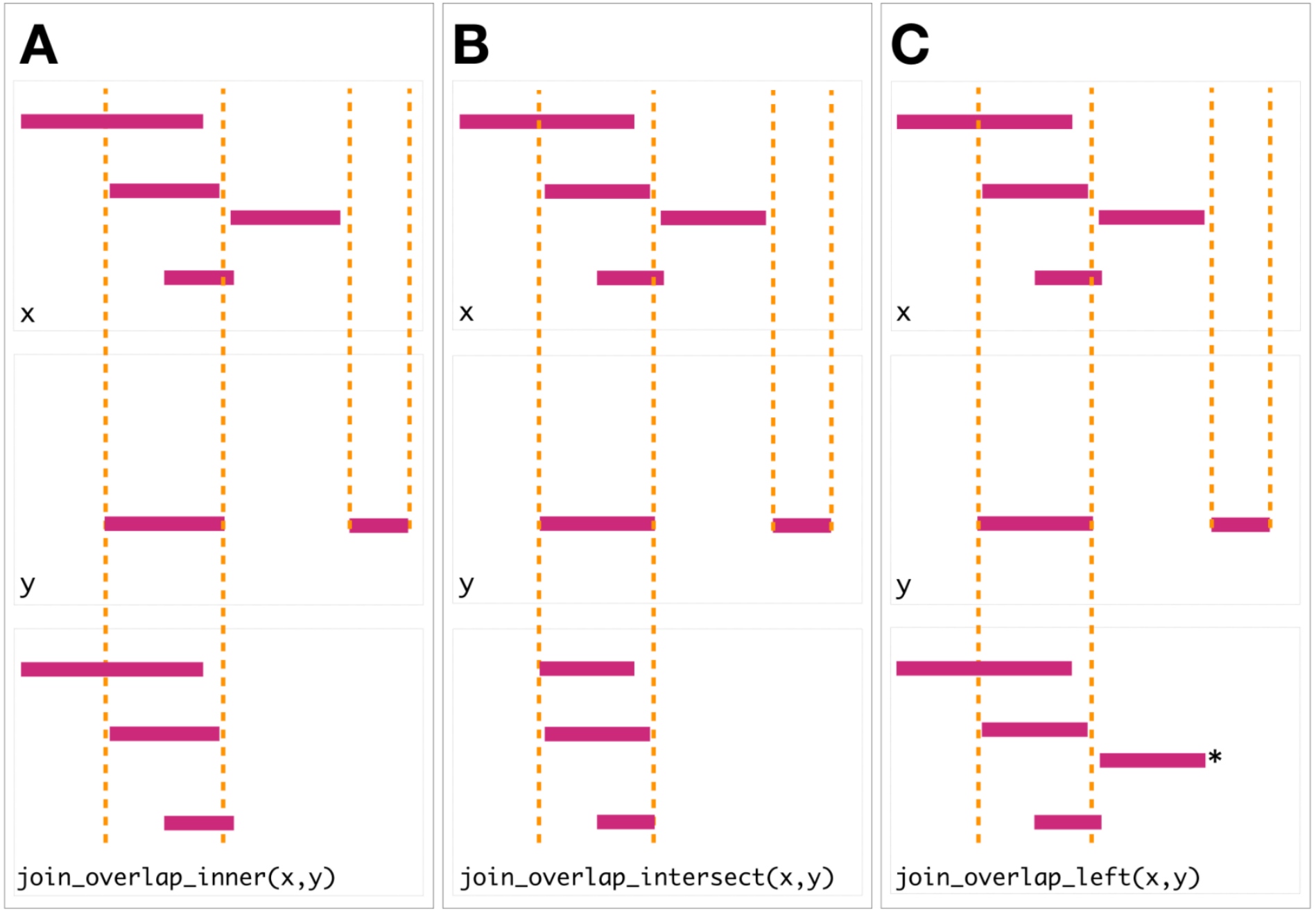


GRanges objects

seqname	start	end	strand	gene_id	score
chr1	102012	120303	+	1001	10
chr1	520211	526211	-	2151	25
...	...	...	...	...	...

GRanges objects

```
> my_peaks
GRanges object with 106195 ranges and 3 metadata columns:
           seqnames              ranges strand |                   name    qValue            gene_id
              <Rle>           <IRanges>  <Rle> |            <character> <numeric>        <character>
       [1]     chr1     4785292-4786021      * |   Naive_H3K27ac_peak_1   21.1013 ENSMUSG00000033845
       [2]     chr1     4785292-4786021      * |   Naive_H3K27ac_peak_1   21.1013 ENSMUSG00000033845
       [3]     chr1     4785292-4786021      * |   Naive_H3K27ac_peak_1   21.1013 ENSMUSG00000033845
       [4]     chr1     4785292-4786021      * |   Naive_H3K27ac_peak_1   21.1013 ENSMUSG00000033845
       [5]     chr1     4785292-4786021      * |   Naive_H3K27ac_peak_1   21.1013 ENSMUSG00000033845
       ...      ...                 ...    ... .                   ...       ...                ...
  [106191]     chr9 124422932-124426220      * | Naive_H3K27ac_peak_1..   77.1462 ENSMUSG00000093803
  [106192]     chr9 124422932-124426220      * | Naive_H3K27ac_peak_1..   77.1462 ENSMUSG00000093803
  [106193]     chr9 124422932-124426220      * | Naive_H3K27ac_peak_1..   77.1462 ENSMUSG00000093803
  [106194]     chr9 124422932-124426220      * | Naive_H3K27ac_peak_1..   77.1462 ENSMUSG00000093803
  [106195]     chr9 124422932-124426220      * | Naive_H3K27ac_peak_1..   77.1462 ENSMUSG00000093803
  -------
  seqinfo: 19 sequences from an unspecified genome; no seqlengths
```


A
x
y
join_overlap_inner(x,y)
B
x
y
join_overlap_intersect(x,y)
C
x
y
*
join_overlap_left(x,y)

read_bed(), read_narrowpeak(), read_gff(), etc... convert files to GRanges objects

```
naive_peaks <-
read_narrowpeaks("Naive_H3K27ac_peaks.autosomes.narrowPeak")

mouse_genes <-
read_gff("Mus_musculus.GRCm38.84.chr.autosomes.gtf")
```

Return original Naïve T-cell peaks with overlapping gene information attached

```
naive_peaks_w_overlapped_genes <- naive_peaks %>%
  join_overlap_left(mouse_genes) %>%
  select(name, qValue, gene_id)
```

You can apply any tidyverse verbs you want to GRanges!

```
naive_peaks_w_overlapped_genes %>%
  group_by(gene_id) %>%
  summarize(qValue_mean = mean(qValue))
```

Category	Verb (i.e. Function)	Description
Aggregate	***summarize()***	Aggregate over column(s)
	disjoin_ranges()	Aggregate column(s) over the union of end coordinates
	reduce_ranges()	Aggregate column(s) by merging overlapping and neighboring ranges
Modify (Unary)	***mutate()***	Modifies any column
	select()	Select columns
	arrange()	Sort by columns
	stretch()	Extend range by fixed amount
	shift_(direction)	Shift coordinates
	flank_(direction)	Generate flanking regions
	%intersection%	Row-wise intersection
	%union%	Row-wise union
	compute_coverage	Coverage over all ranges
Modify (Binary)	*%setdiff%*	Row-wise set difference
	between()	Row-wise gap range
	span()	Row-wise spanning range

Category	Verb (i.e. Function)	Description
Merge	*join_overlap_*()*	Merge by overlapping ranges
	join_nearest	Merge by nearest neighbor ranges
	join_follow	Merge by following ranges
	join_precedes	Merge by preceding ranges
	union_ranges	Range-wise union
	intersect_ranges	Range-wise intersect
	setdiff_ranges	Range-wise set difference
	complement_ranges	Range-wise set complement
Operate	*anchor_direction()*	Fix coordinates at direction
	group_by()	Partition by column(s)
	group_by_overlaps()	Partition by overlaps
Restrict	***filter()***	Subset rows
	filter_by_overlaps()	Subset by overlap
	filter_by_non_overlaps()	Subset by no overlap

Resources

BEDtools Documentation

Bedtools lecture from original creator

BEDtools tutorial with more commands

plyranges vignette

Getting started with plyranges

plyranges original publication