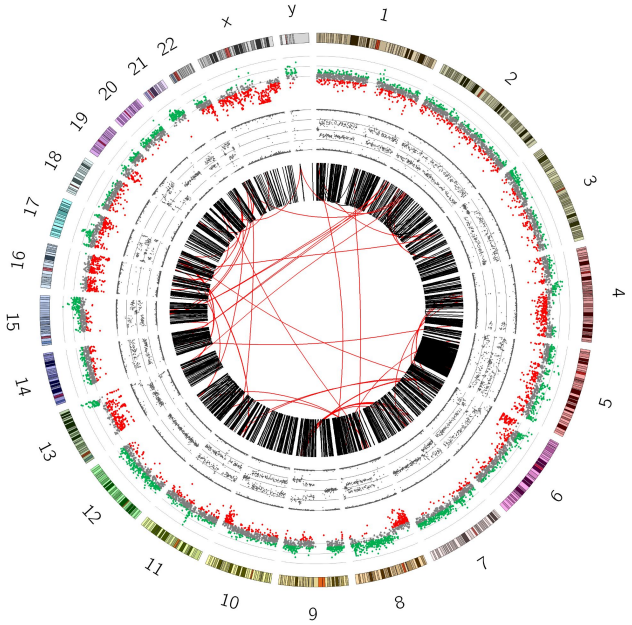
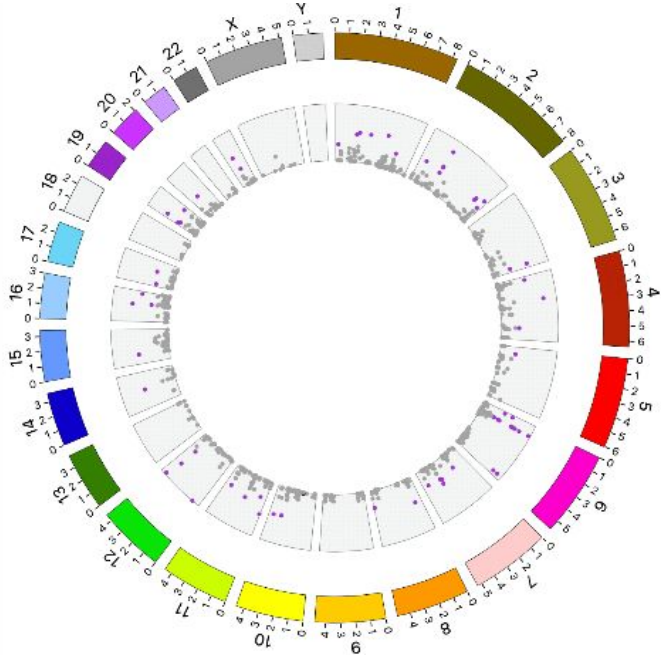
Visualizing Chromosomal Data for Cancer Detection

Peter Zhang
Mentor: Dr. Subhajyoti De

Problem Statement

- Developing an standard pipeline approach to visualize chromosomal data to better understand the biological components for non-invasive cancer detection





Objectives

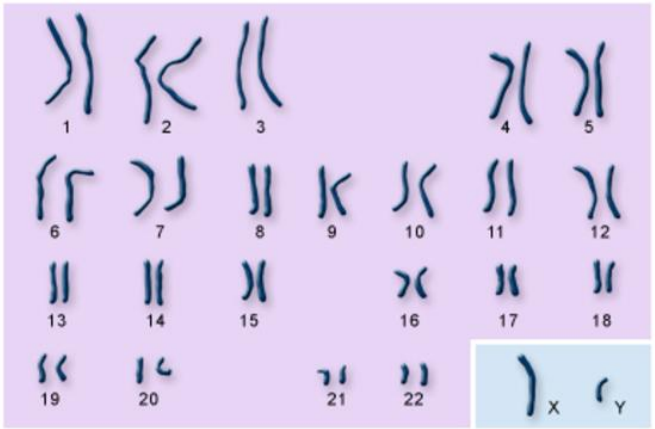
- Find a way to handle large amount of complex genomic data
- Visualize the chromosomal data through circos plot
- Process the chromosomal data in either BED or CSV file

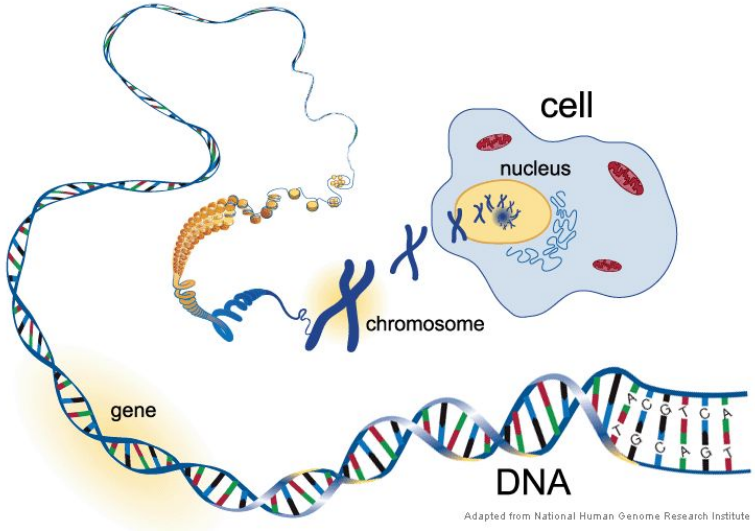
```
1  chr7    127471196  127472363
2  chr7    127472363  127473530
3  chr7    127473530  127474697
4  chr7    127474697  127475864
```

- Standardize / cap the - log q value
  - Q value: the proportion of false positives within a region

Introduction: Chromosomal Data

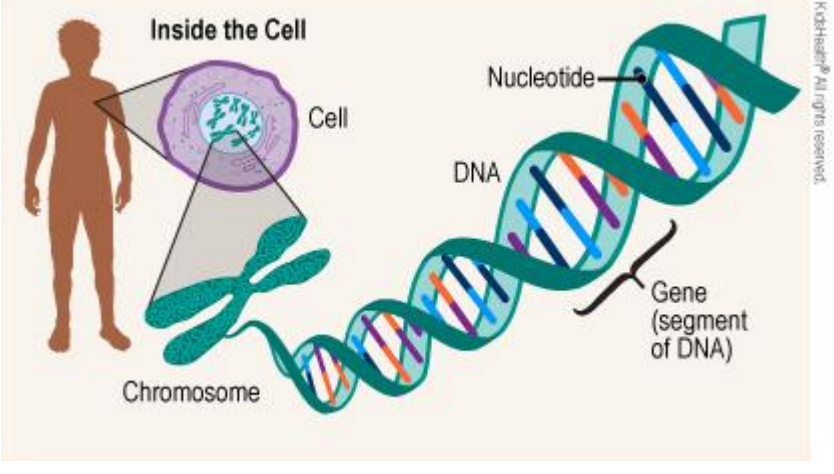
- Chromosomal data refers to the detailed information encoded within the DNA of chromosomes
- Human has 23 pairs of chromosomes (46 in total)
- XY are male, XX are female





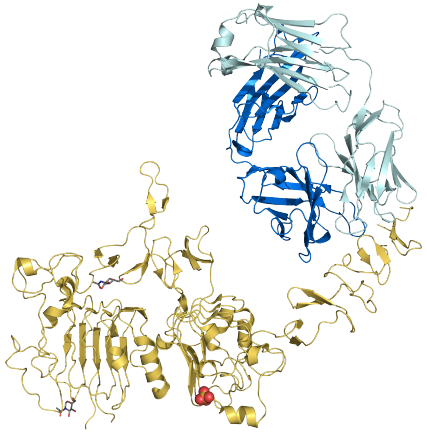
Introduction：How does that relate with cancer?

- Cancers are caused by damage to the DNA in your cells. (Remember chromosomes are made up from DNA and other proteins)
- Cells with too many mutations may stop working normally, grow out of control and become cancerous
- In our research, we will be focusing on amplification genes and deletion genes



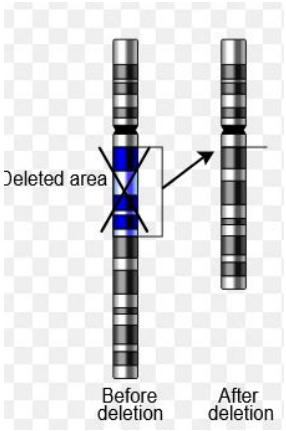
Introduction: Gene Amplification

- Amplification of chromosomes results in multiple copies of the genes on that chromosome.
- Leads to overexpression of genes, some of which may be oncogenes—genes that drive cancer development when activated or overexpressed
- Ex: HER2 is a well-known oncogene in breast cancer that is often amplified and leads to aggressive tumor growth.



Introduction: Gene Deletion

- Deletions remove portions of the chromosome that normally help regulate cell growth and division.
- Loss of these segments can lead to the absence of tumor suppressor genes, which normally help prevent cancer by repairing DNA or controlling cell division
- When these genes are lost, it can lead to uncontrolled cell growth.



Understanding the Data - ovarian cancer

	A	B	C	D	E	F	G	H	I	J	K	L
1	Chromosome	Wide peak start	Wide peak end	amp/del	2q3	Wide Peak Limits	Peak Limits	Region Lim	q values	Residual q	Description with number	Description
2	chr19	34934008	35059410	Amp	19q12	chr19:34934008-35(	chr19:34981905-3	chr19:23143	1.89E-102	2.57E-96	CCNE1 [1]	CCNE1
3	chr3	170150840	170280412	Amp	3q26.2	chr3:170150840-17(	chr3:170249939-	chr3:149545	8.00E-63	3.28E-56	MECOM [0]	MECOM
4	chr8	128754508	129388953	Amp	8q24.21	chr8:128754508-12(	chr8:128809381-	chr8:100035	1.86E-79	3.70E-56	MYC [2]	MYC
5	chr11	77509837	77679169	Amp	11q14.1	chr11:77509837-776	chr11:77579027-7	chr11:68349	9.28E-28	9.28E-28	ALG8 [4]	ALG8
6	chr12	25059809	25394729	Amp	12p12.1	chr12:25059809-253	chr12:25311849-2	chr12:17822	6.39E-25	2.66E-23	KRAS [4]	KRAS
7	chr19	44455609	45045812	Amp	19q13.2	chr19:44455609-45(	chr19:44828320-4	chr19:23143	5.54E-29	7.57E-23	19q13.2 [24]	19q13.2
8	chr8	144180457	146072839	Amp	8q24.3	chr8:144180457-146	chr8:145628324-	chr8:100035	1.57E-41	4.34E-20	8q24.3 [75]	8q24.3
9	chr19	12983113	14103663	Amp	19p13.13	chr19:12983113-141	chr19:13749627-	chr19:87065	5.66E-26	6.50E-18	19p13.13 [29]	19p13.13
10	chr1	39109153	41523062	Amp	1p34.2	chr1:39109153-4152	chr1:40147908-4(	chr1:350309	1.28E-21	6.50E-16	1p34.2(MYCL1) [41]	1p34.2(MYCL
11	chr1	148646436	149486905	Amp	1q21.2	chr1:148646436-149	chr1:148763510-	chr1:120326	5.19E-18	1.45E-13	1q21.2(MCL1) [28]	1q21.2(MCL1)

Understanding the Data

- Chromosome number : chr1
- Start of gene's relative location : 10000
- End of gene's relative location: 15000
- Type of altercation: Amplification or Deletion : amp / del
- Q value: statistical measure for False Discovery Rate. The expression's significance.
  - Lower q value means less likely to be a false discovery
  - Derived from p-value but adjusted from multiple experiments
- Gene's description

Handling complex data: S4 Objects in R

- System for Object-Oriented Programming Version 4
- provides a formal and structured approach to object-oriented programming
- Structured data
- Able to extend the class with more properties and methods
- S4 is better at modeling complex data
- More efficient in computations and saves space

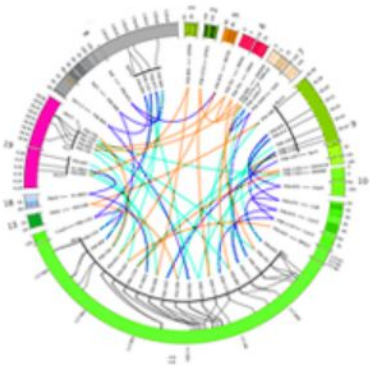
```r
# Load the methods package for S4 object systems
library(methods)

# Define the S4 class
setClass("Person",
         slots = list(
           name = "character",
           age = "numeric"
         ))
```

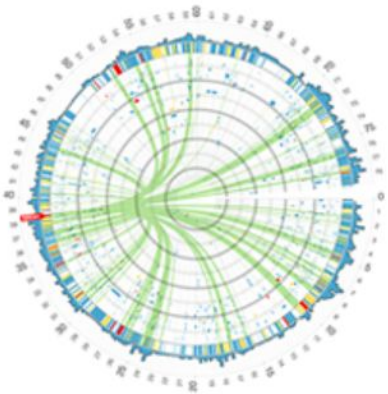
```r
setMethod("printDetails", "Person",
          function(object) {
            cat(paste("Name:", object@name,
          })
```

```r
# Create a new Person object
john <- newPerson("John Doe", 30)

# Print details of the object
printDetails(john)
```

Why use circos plot?

- Valuable in genomic research due to their unique circular layout,
- Easy to compare peaks of different chromosomes' expressions
- Able to show more layered tracks to compare expression levels, DNA copy numbers, and epigenetic data in 1 graph
- Better at showing relationships than linear plots for periodic effects





Convert Data to BED file

- Extract important column and store them in the standard genomic data

```
# Load the BED file
bed_file_path = './amp.bed'
bed_data = pd.read_csv(bed_file_path, sep='\t', header=None, names=['chrom', 'start', 'end',

# Ensure 'start' and 'end' columns are integers
bed_data['start'] = pd.to_numeric(bed_data['start'], errors='coerce')
bed_data['end'] = pd.to_numeric(bed_data['end'], errors='coerce')

# Drop any rows that couldn't be converted to numeric values
bed_data = bed_data.dropna(subset=['start', 'end'])

# Calculate the maximum of end - start for each chromosome
max_length_per_chrom = bed_data.groupby('chrom').apply(lambda x: (x['end'] - x['start']).ma

# Sort the index of max_length_per_chrom by converting chromosome names to a sortable formai
# Here we remove 'chr' prefix and convert the chromosome number to an integer for sorting
max_length_per_chrom_sorted = max_length_per_chrom.sort_index(key=lambda x: x.str.replace('
```

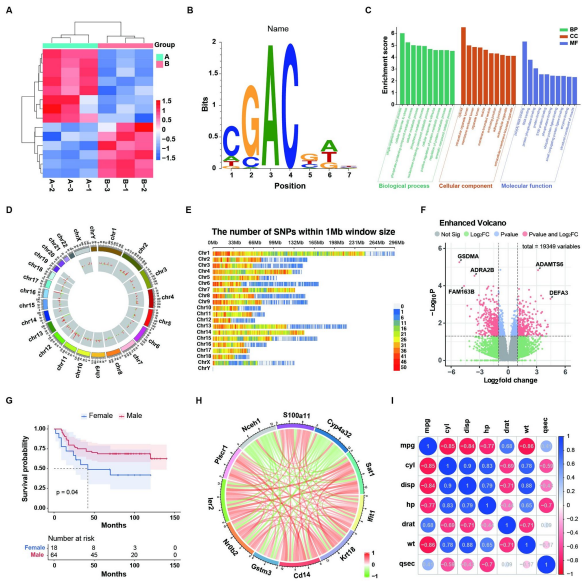
```
# Display the sorted results
max_length_per_chrom_sorted

new_bed_data = pd.DataFrame({
    '#chrom': max_length_per_chrom_sorted.index,
    'chromStart': 0,
    'chromEnd': max_length_per_chrom_sorted.values.astype(int),
    'name': max_length_per_chrom_sorted.index
})

# Save the data to a new BED file
new_bed_file_path = './dist_amp.bed'
new_bed_data.to_csv(new_bed_file_path, sep='\t', index=False, header=False)
```

#chrom	chromStart	chromEnd	name
chr1	39109153	41523062	-48.10742687494343
chr1	148646436	149486905	-39.79979797671503
chr1	232526850	233617929	-13.95822686016586
chr1	154088696	154329131	-19.36743068331123
chr1	36338557	36480225	-24.57701993425058
chr1	178780208	178854243	-6.527402871689047
chr1	241440102	247249719	-7.603886908583042
chr1	1095521	1467046	-4.071253230998703
chr2	189030591	192069072	-9.57850969534065
chr2	28497864	28879745	-2.971839981134388
chr2	113658200	113809678	-2.526266288813163
chr2	176544739	179048787	-7.23607880669681
chr2	41813425	68291342	-2.486832503053504
chr3	170150840	170280412	-142.9834193169450

Method 1: SRPlot website

- Create circos plot by using a pre-configured bioinformatic graphing website
- Easy to use
- Lack of customization for chart types, more tracks
- Cannot annotate the genes
- Not open source

chr17	13812739	13812680	2.3039126
chr17	8760421	8760480	2.3913348
chr17	44503584	44503525	2.0151665
chr12	110983302	110983243	2.7099345
chr2	128217604	128217545	5.038758
chr17	71224133	71224192	2.3658962
chr8	122789197	122789138	2.2939339
chr6	100517199	100517140	2.901734
chr6	121038012	121037953	2.9011755
chr18	54122702	54122761	2.2560575
chr17	66831861	66831920	2.3124216

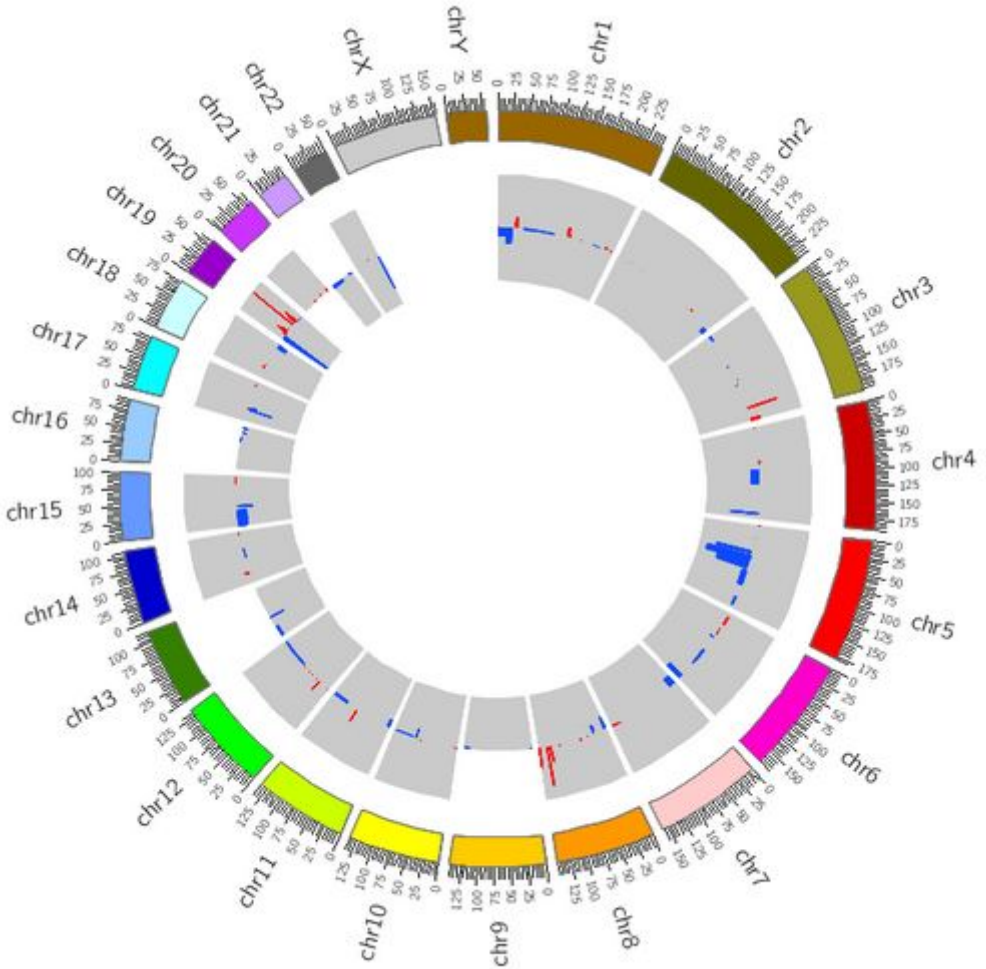


Result

- Without the gray area means there is no available data at the assigned chromosomes

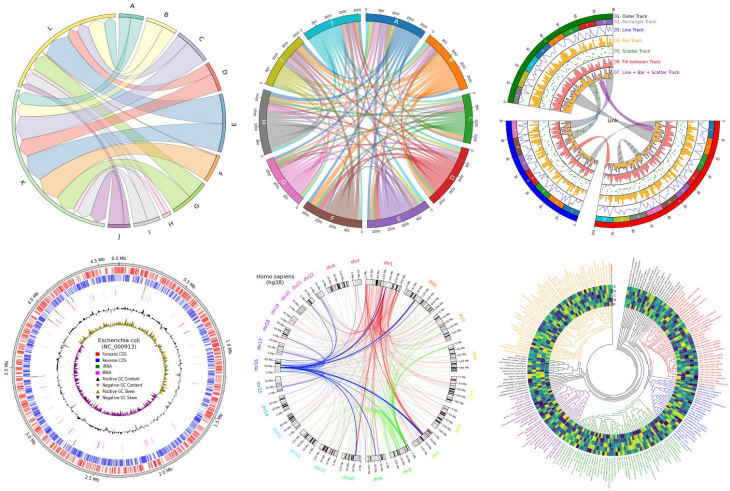
- Higher the bar means higher the significant

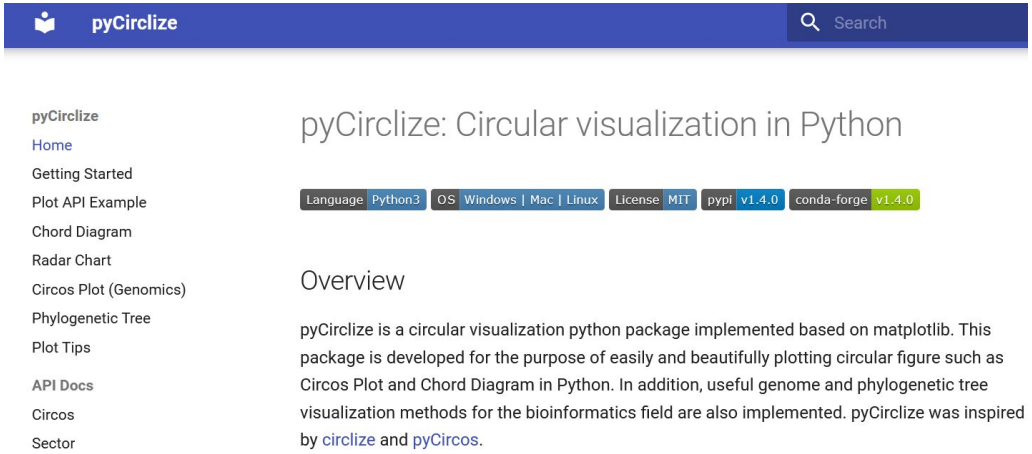
- Long clusters of same type of gene alteration could show problem



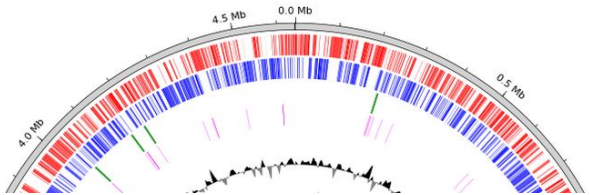
Method 2: Pycirclize

- Python library for creating circos plots
- Good at customization
- High flexibility
- Supports multiple types of circos plots
- Lots of documentation for genomic data





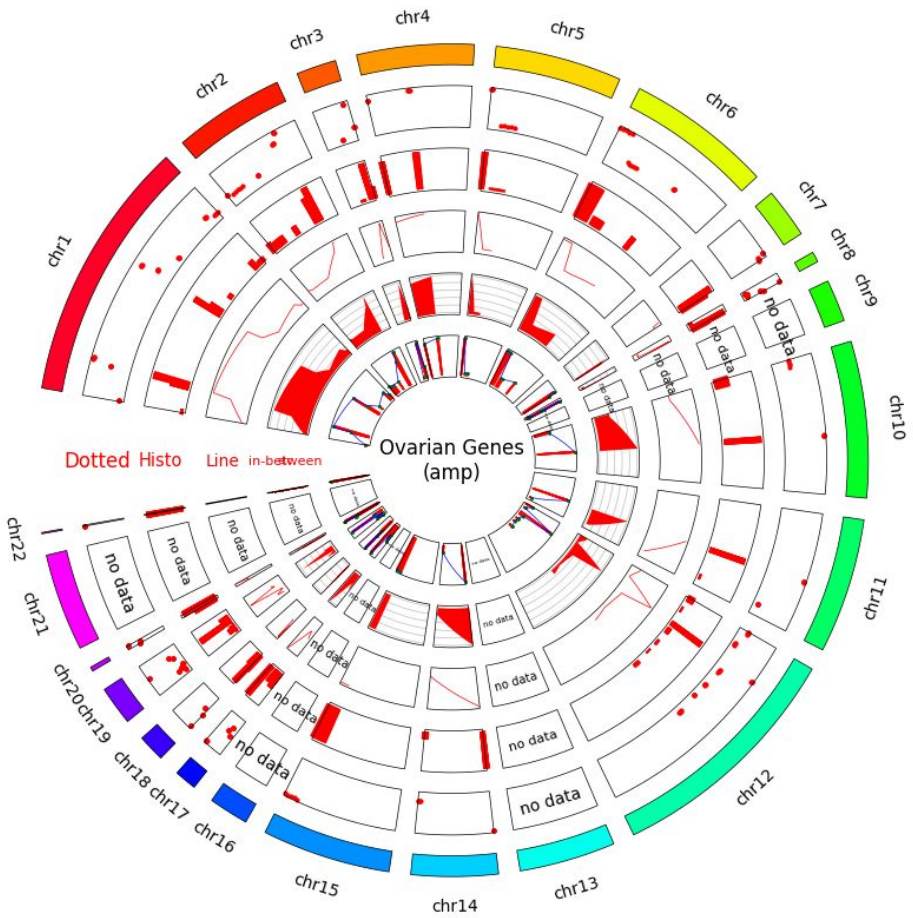

```
    Line2D([], [], color="black", label="Positive GC Content", marker="
    Line2D([], [], color="grey", label="Negative GC Content", marker="v
    Line2D([], [], color="olive", label="Positive GC Skew", marker="^",
    Line2D([], [], color="purple", label="Negative GC Skew", marker="v"
]
_ = circos.ax.legend(handles=handles, bbox_to_anchor=(0.5, 0.475), loc=
```



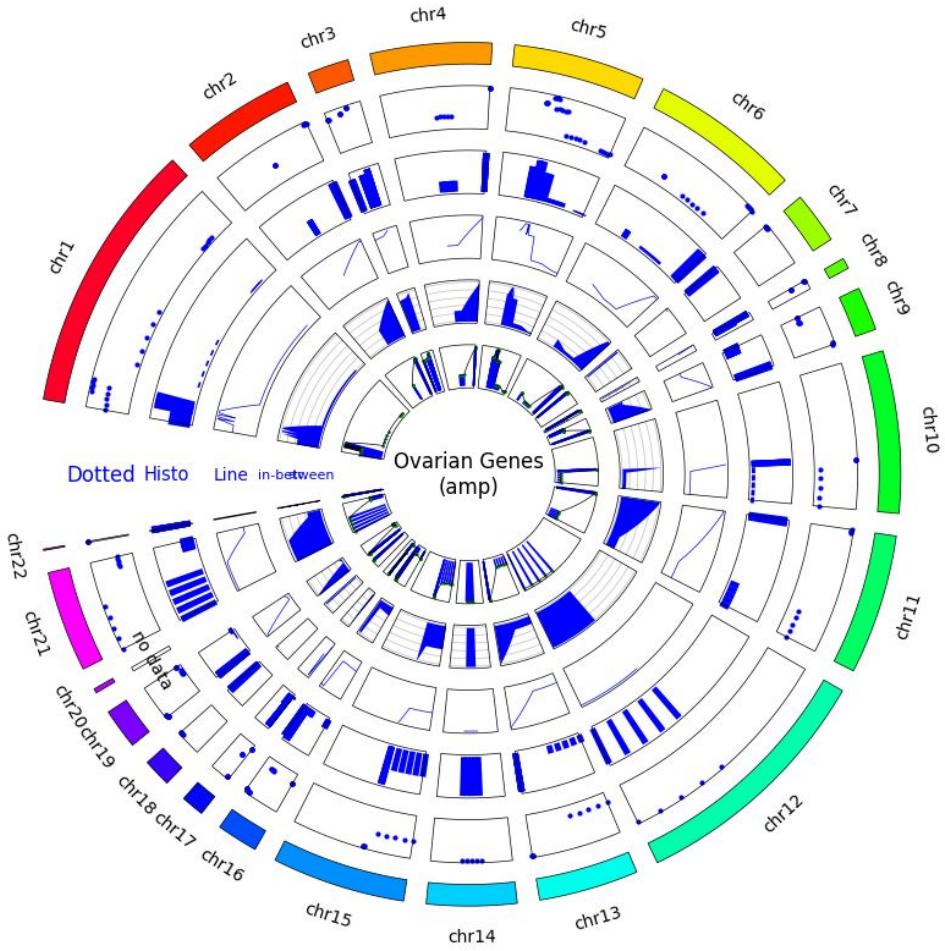
Amplification Genes

- Different tracks is different types of plots but representing the same data

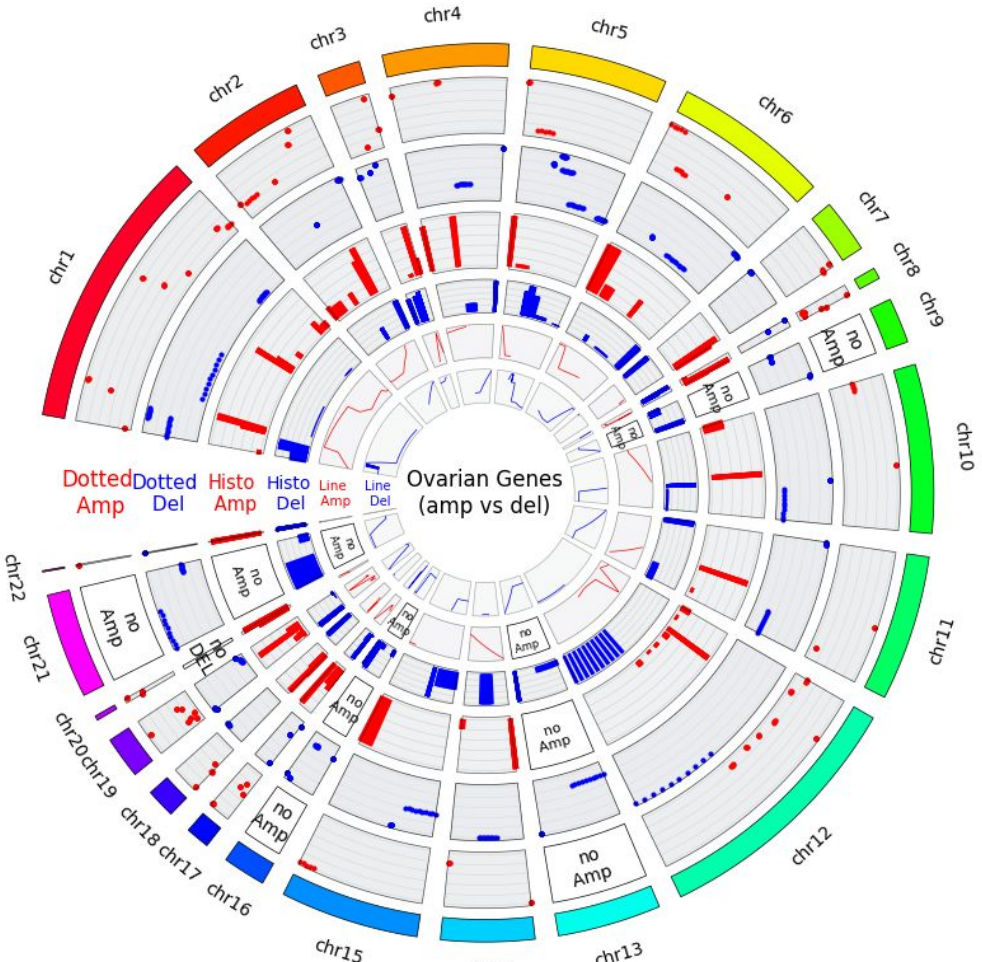
- More clear visualization then the combined comparison graph



Deletion Genes

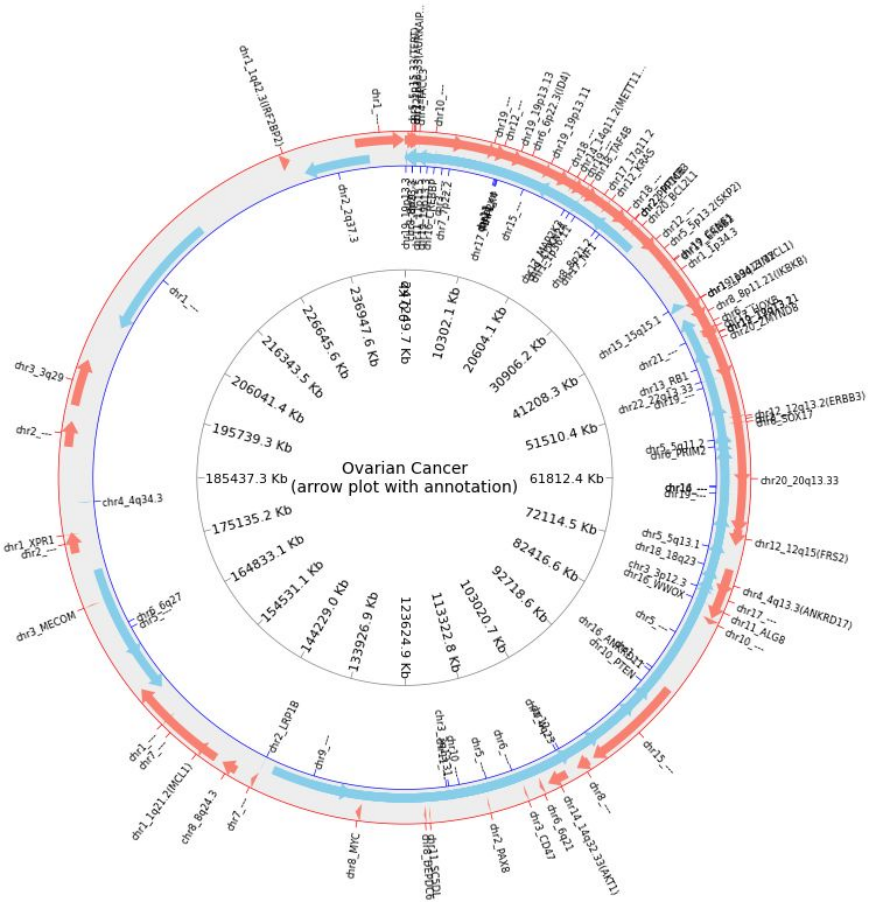


With Both Gene Amplification and Deletion



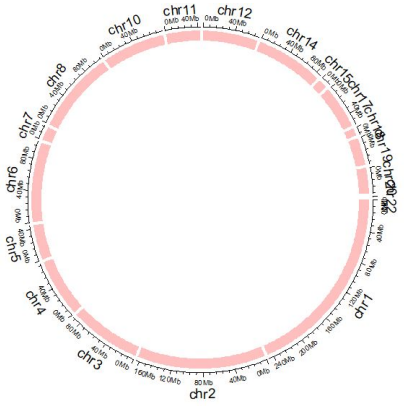
Arrowed Circos Plot with gene description

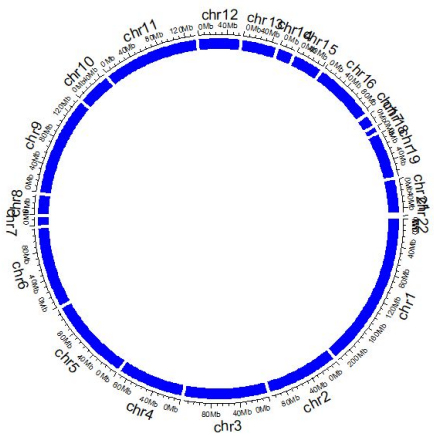
- The plot goes counter-clockwise

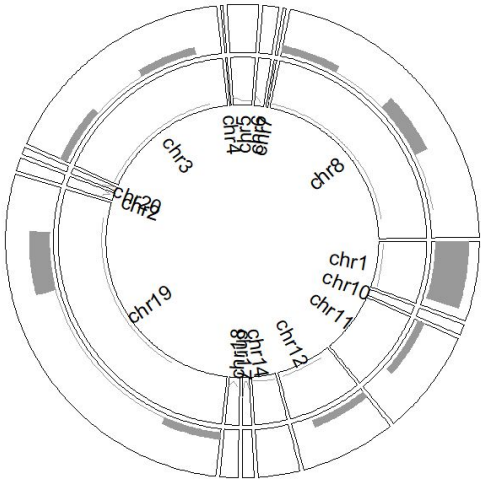


Challenges & Limitations

- Tried R's circlize library but too difficult and too long to debug
- Switched to use pycirclize library but R is more standard for processing genomic data
- R is easier to integrate with other bioinformatics applications or other analysis pipeline
- Hoping to switch to using R to plot







Conclusion

- Circos plot is a powerful tool to visualize the patterns of genomic data.
- Able to detect the peaks and lows in the zones of gene amplification and gene deletion, however, more thorough analysis still required numerical and machine learning prediction models
- Circos plot can provide the baseline of where potentially might be the problem of the chromosomes based on contrast of amp and del zones
- There are numerous circos libraries with support to different languages. However, I still find it difficult to navigate my way starting from the pure genomic data.
- More detailed user guides and documentations for new users are needed.

Future works

- Move to R
- Integrate with web application with more interactive features
- Able to upload and choose your own dataset and automatically generate circos plots with descriptions and annotations
- Potential collab with LLM for explaining the types of genes.

References

https://moshi4.github.io/pyCirclize/

https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0294236

https://circos.ca/

https://www.genome.gov/about-genomics/fact-sheets/Chromosomes-Fact-Sheet

https://www.mountsinai.org/health-library/special-topic/chromosome