

Genome Assembly

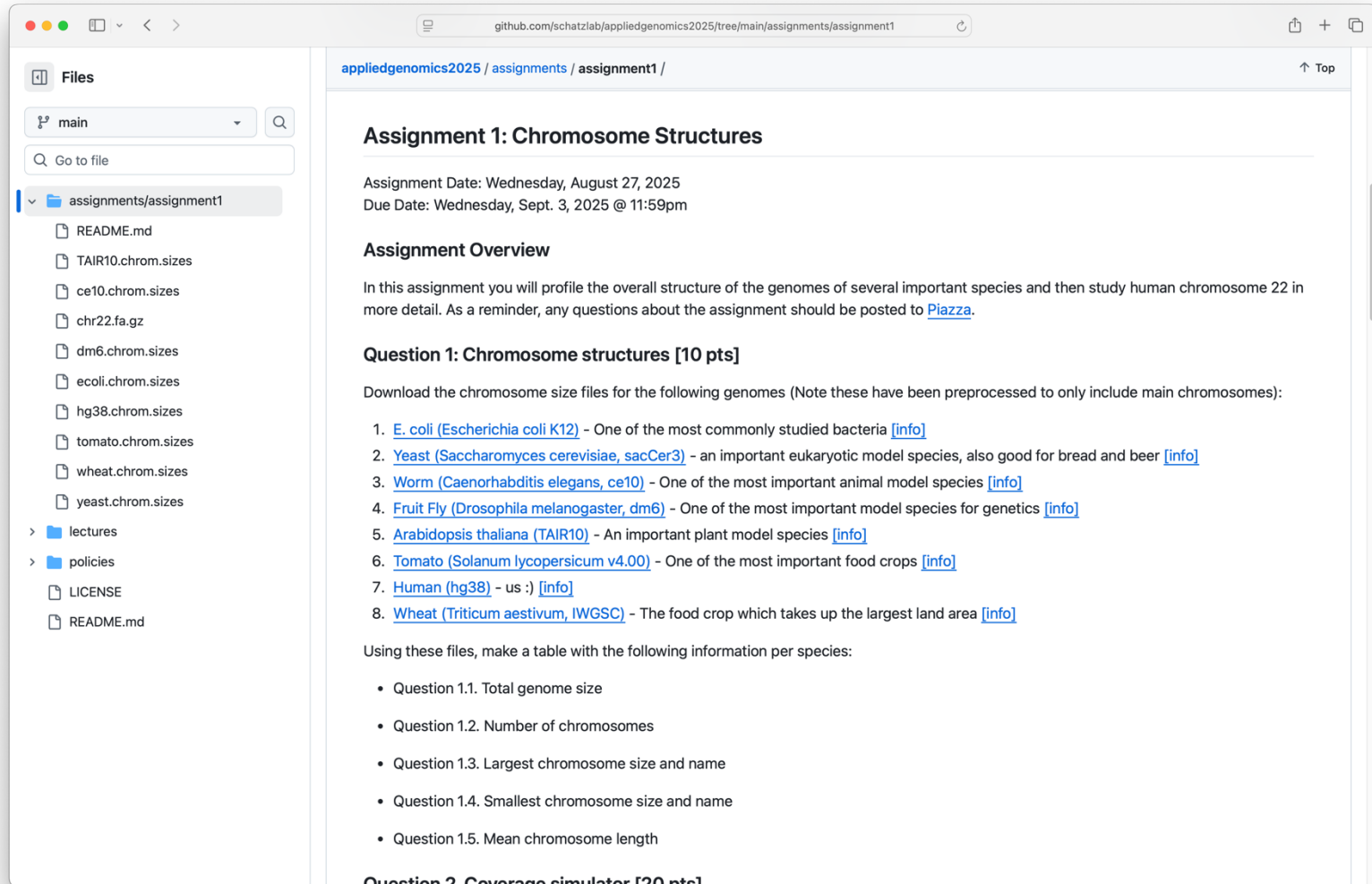
Michael Schatz

Sept 3, 2025

Lecture 3: Applied Comparative Genomics



Assignment I



The screenshot shows a web browser displaying the GitHub repository page for `appliedgenomics2025/assignments/assignment1`. The left sidebar shows the file structure with a folder named `assignments/assignment1` expanded, listing files like `README.md`, `TAIR10.chrom.sizes`, `ce10.chrom.sizes`, `chr22.fa.gz`, `dm6.chrom.sizes`, `ecoli.chrom.sizes`, `hg38.chrom.sizes`, `tomato.chrom.sizes`, `wheat.chrom.sizes`, and `yeast.chrom.sizes`. The main content area displays the assignment details.

Assignment 1: Chromosome Structures

Assignment Date: Wednesday, August 27, 2025
Due Date: Wednesday, Sept. 3, 2025 @ 11:59pm

Assignment Overview

In this assignment you will profile the overall structure of the genomes of several important species and then study human chromosome 22 in more detail. As a reminder, any questions about the assignment should be posted to [Piazza](#).

Question 1: Chromosome structures [10 pts]

Download the chromosome size files for the following genomes (Note these have been preprocessed to only include main chromosomes):

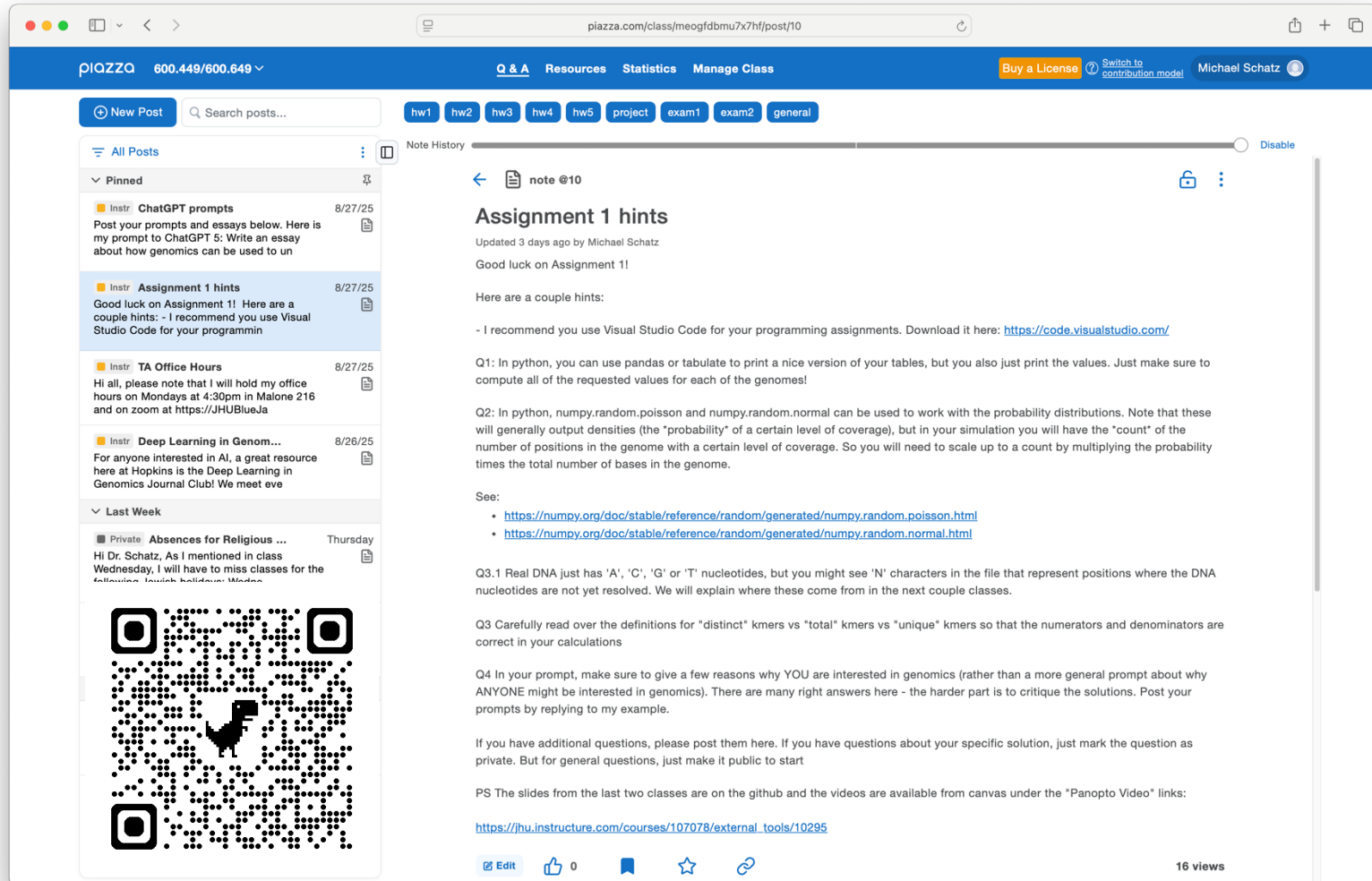
1. [E. coli \(Escherichia coli K12\)](#) - One of the most commonly studied bacteria [\[info\]](#)
2. [Yeast \(Saccharomyces cerevisiae, sacCer3\)](#) - an important eukaryotic model species, also good for bread and beer [\[info\]](#)
3. [Worm \(Caenorhabditis elegans, ce10\)](#) - One of the most important animal model species [\[info\]](#)
4. [Fruit Fly \(Drosophila melanogaster, dm6\)](#) - One of the most important model species for genetics [\[info\]](#)
5. [Arabidopsis thaliana \(TAIR10\)](#) - An important plant model species [\[info\]](#)
6. [Tomato \(Solanum lycopersicum v4.00\)](#) - One of the most important food crops [\[info\]](#)
7. [Human \(hg38\)](#) - us :) [\[info\]](#)
8. [Wheat \(Triticum aestivum, IWGSC\)](#) - The food crop which takes up the largest land area [\[info\]](#)

Using these files, make a table with the following information per species:

- Question 1.1. Total genome size
- Question 1.2. Number of chromosomes
- Question 1.3. Largest chromosome size and name
- Question 1.4. Smallest chromosome size and name
- Question 1.5. Mean chromosome length

Question 2: Coverage simulator [20 pts]

<https://github.com/schatzlab/appliedgenomics2025/tree/main/assignments/assignment1>
Due end of day on Wednesday Sept 3 (right before midnight)



The screenshot shows a web browser window displaying a Piazza forum post. The browser's address bar shows the URL `piazza.com/class/meogfdbmu7x7hf/post/10`. The Piazza interface includes a top navigation bar with links for 'Q & A', 'Resources', 'Statistics', and 'Manage Class'. A user profile for 'Michael Schatz' is visible in the top right. The main content area shows a post titled 'Assignment 1 hints' by 'note @10', updated 3 days ago. The post contains several paragraphs of text, including a welcome message, a list of hints, and links to resources like Visual Studio Code and NumPy documentation. A sidebar on the left lists other posts, including 'ChatGPT prompts', 'Assignment 1 hints', 'TA Office Hours', and 'Deep Learning in Genom...'. A large QR code is visible at the bottom of the sidebar. The post itself has 0 likes and 16 views.

Assignment 1 hints
Updated 3 days ago by Michael Schatz
Good luck on Assignment 1!

Here are a couple hints:

- I recommend you use Visual Studio Code for your programming assignments. Download it here: <https://code.visualstudio.com/>

Q1: In python, you can use pandas or tabulate to print a nice version of your tables, but you also just print the values. Just make sure to compute all of the requested values for each of the genomes!

Q2: In python, `numpy.random.poisson` and `numpy.random.normal` can be used to work with the probability distributions. Note that these will generally output densities (the "probability" of a certain level of coverage), but in your simulation you will have the "count" of the number of positions in the genome with a certain level of coverage. So you will need to scale up to a count by multiplying the probability times the total number of bases in the genome.

See:

- <https://numpy.org/doc/stable/reference/random/generated/numpy.random.poisson.html>
- <https://numpy.org/doc/stable/reference/random/generated/numpy.random.normal.html>

Q3.1 Real DNA just has 'A', 'C', 'G' or 'T' nucleotides, but you might see 'N' characters in the file that represent positions where the DNA nucleotides are not yet resolved. We will explain where these come from in the next couple classes.

Q3 Carefully read over the definitions for "distinct" kmers vs "total" kmers vs "unique" kmers so that the numerators and denominators are correct in your calculations

Q4 In your prompt, make sure to give a few reasons why YOU are interested in genomics (rather than a more general prompt about why ANYONE might be interested in genomics). There are many right answers here - the harder part is to critique the solutions. Post your prompts by replying to my example.

If you have additional questions, please post them here. If you have questions about your specific solution, just mark the question as private. But for general questions, just make it public to start

PS The slides from the last two classes are on the github and the videos are available from canvas under the "Panopto Video" links:

https://jhu.instructure.com/courses/107078/external_tools/10295

16 views

TA: Mahler Revsine



Office Hours: Mondays at 4:30pm
@ Malone 216 and Zoom



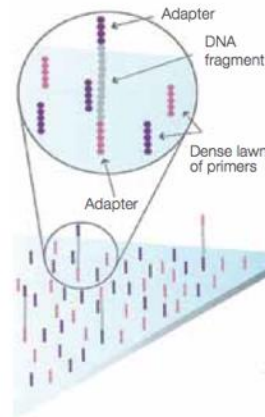
Part I: Recap and Illumina Sequencing

Second Generation Sequencing

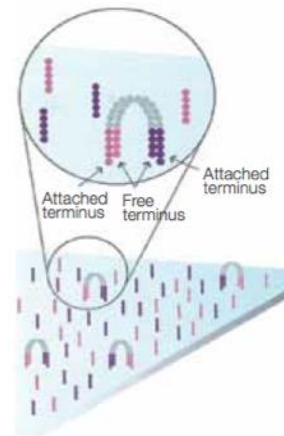


Illumina NovaSeq 6000
Sequencing by Synthesis

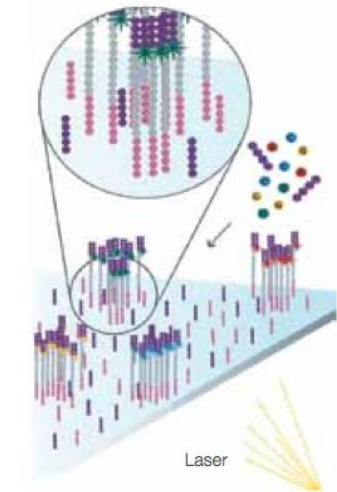
>3Tbp / day
(JHU has 4 of these!)



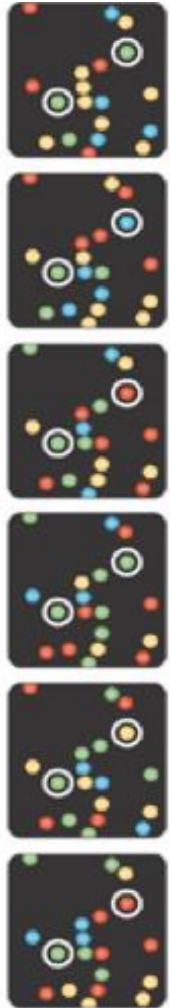
1. Attach



2. Amplify

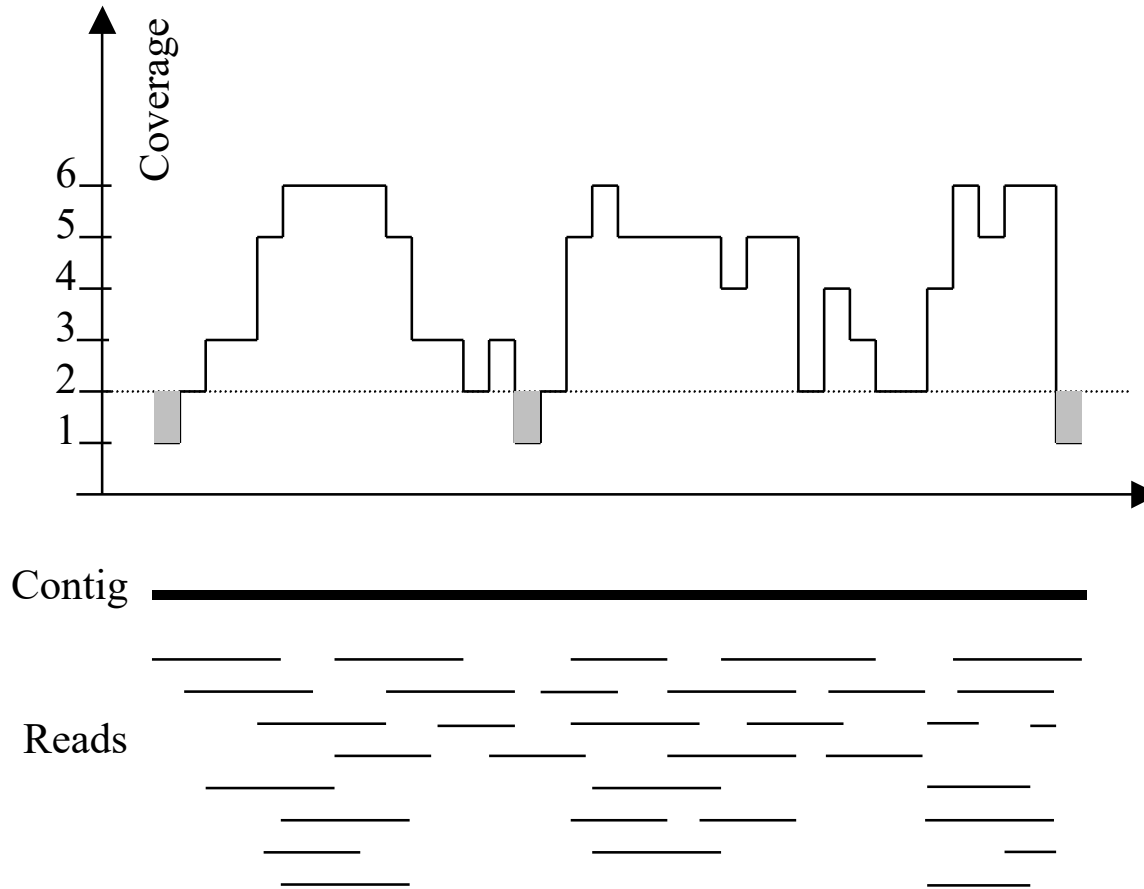


3. Image



Metzker (2010) Nature Reviews Genetics 11:31-46
<https://www.youtube.com/watch?v=fCd6B5HRaZ8>

Typical sequencing coverage

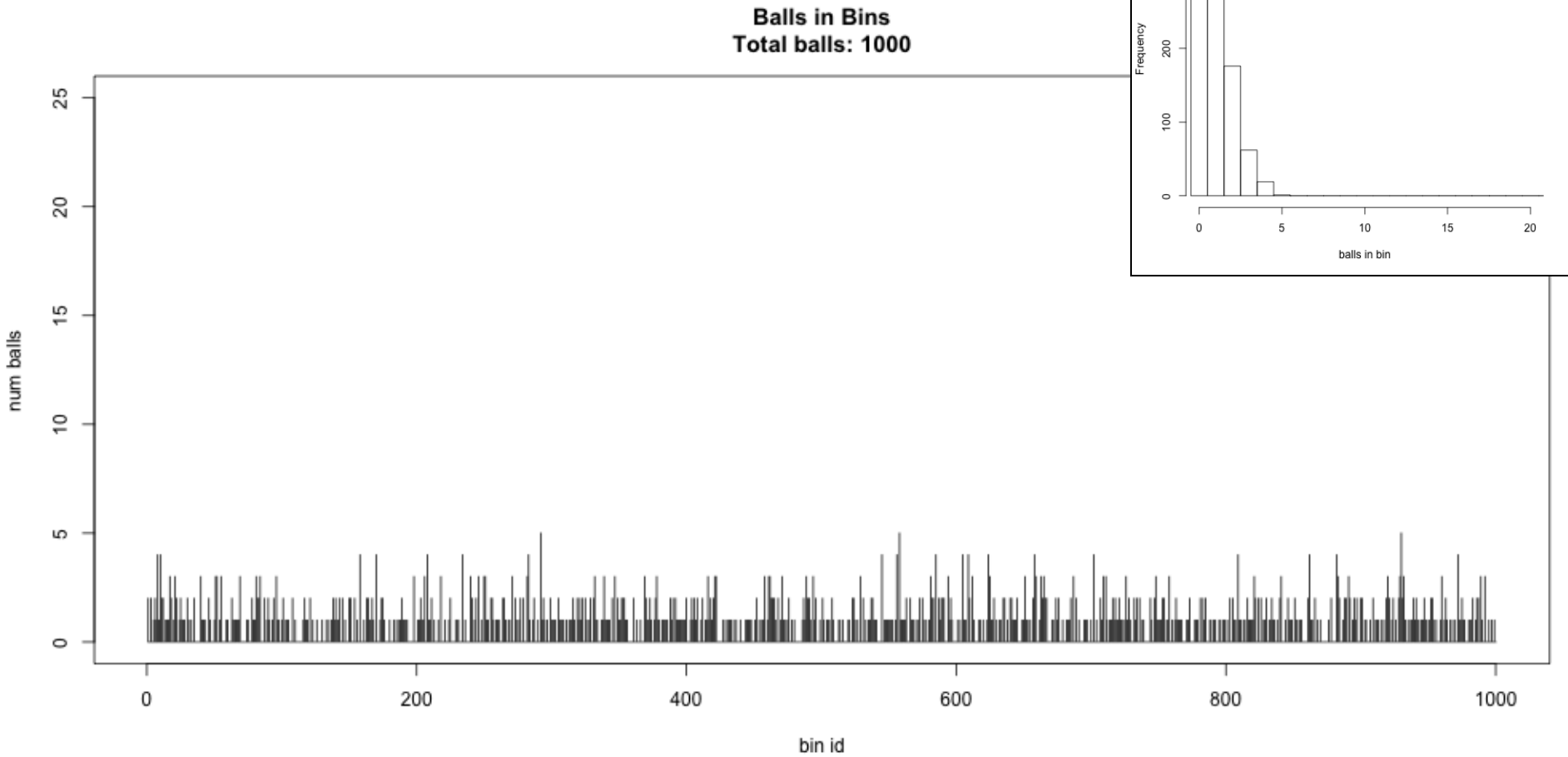


Imagine raindrops on a sidewalk

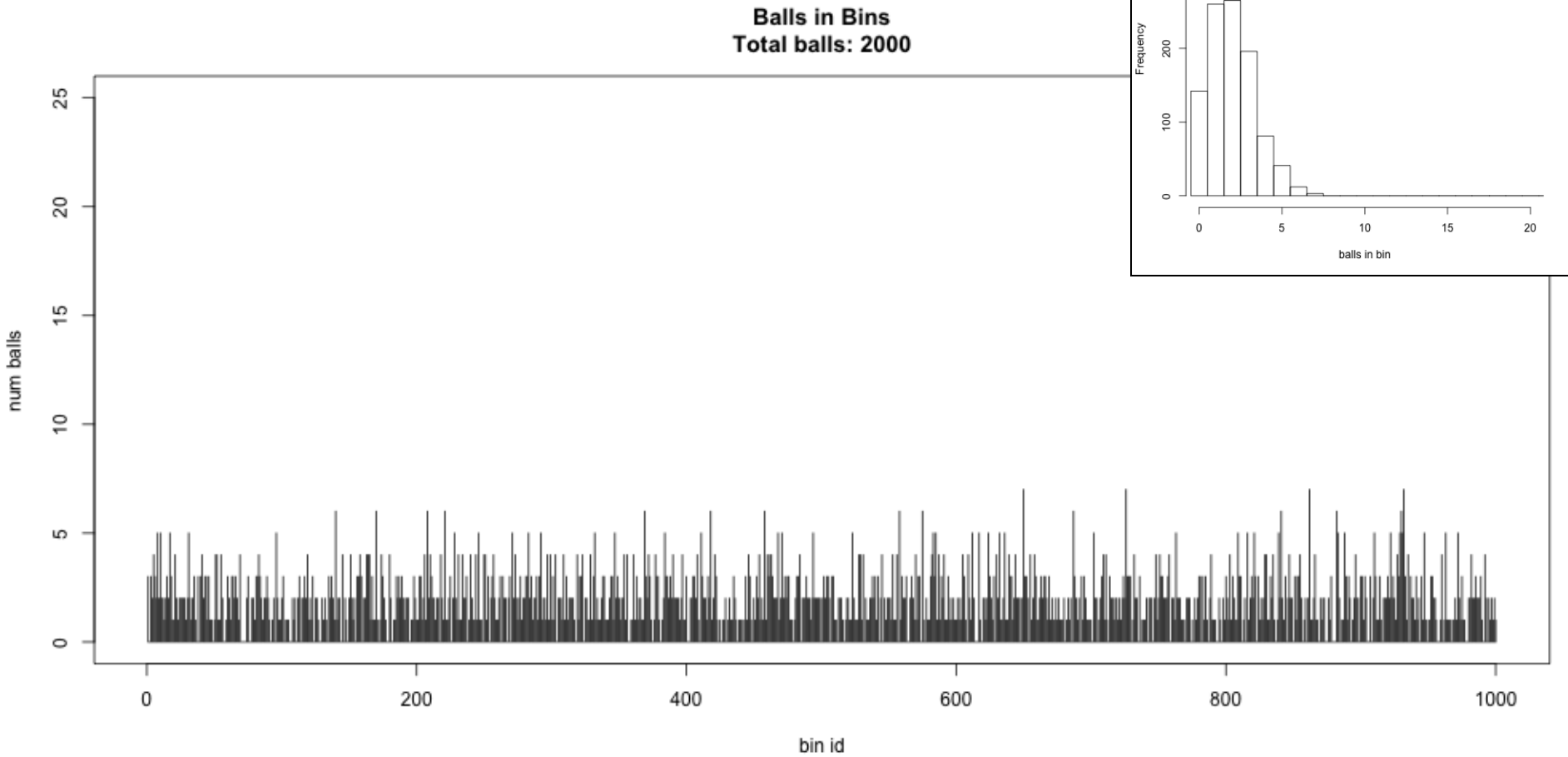
We want to cover the entire sidewalk but each drop costs \$1

If the genome is 10 Mbp, should we sequence 100k 100bp reads?

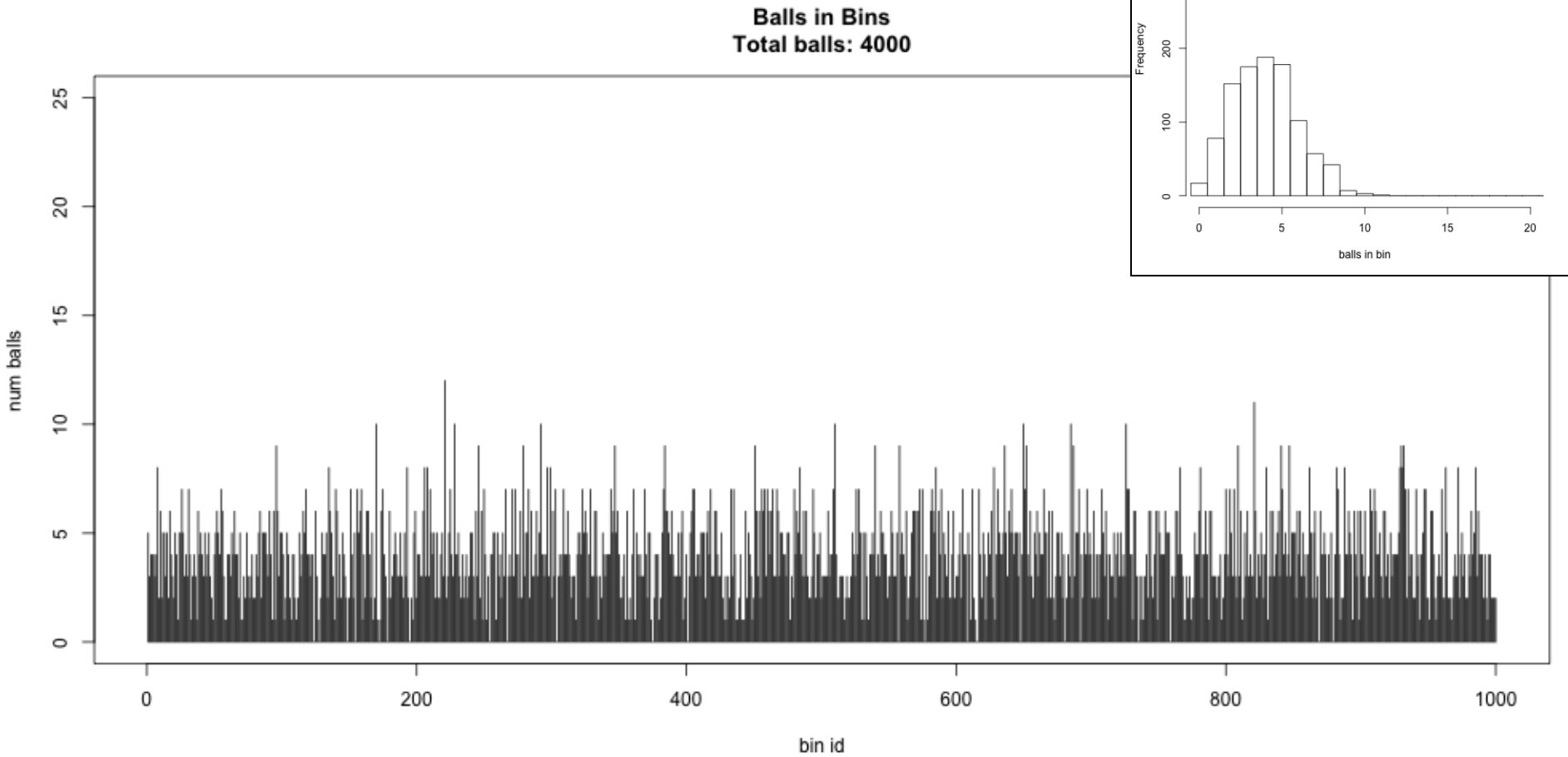
Ix sequencing



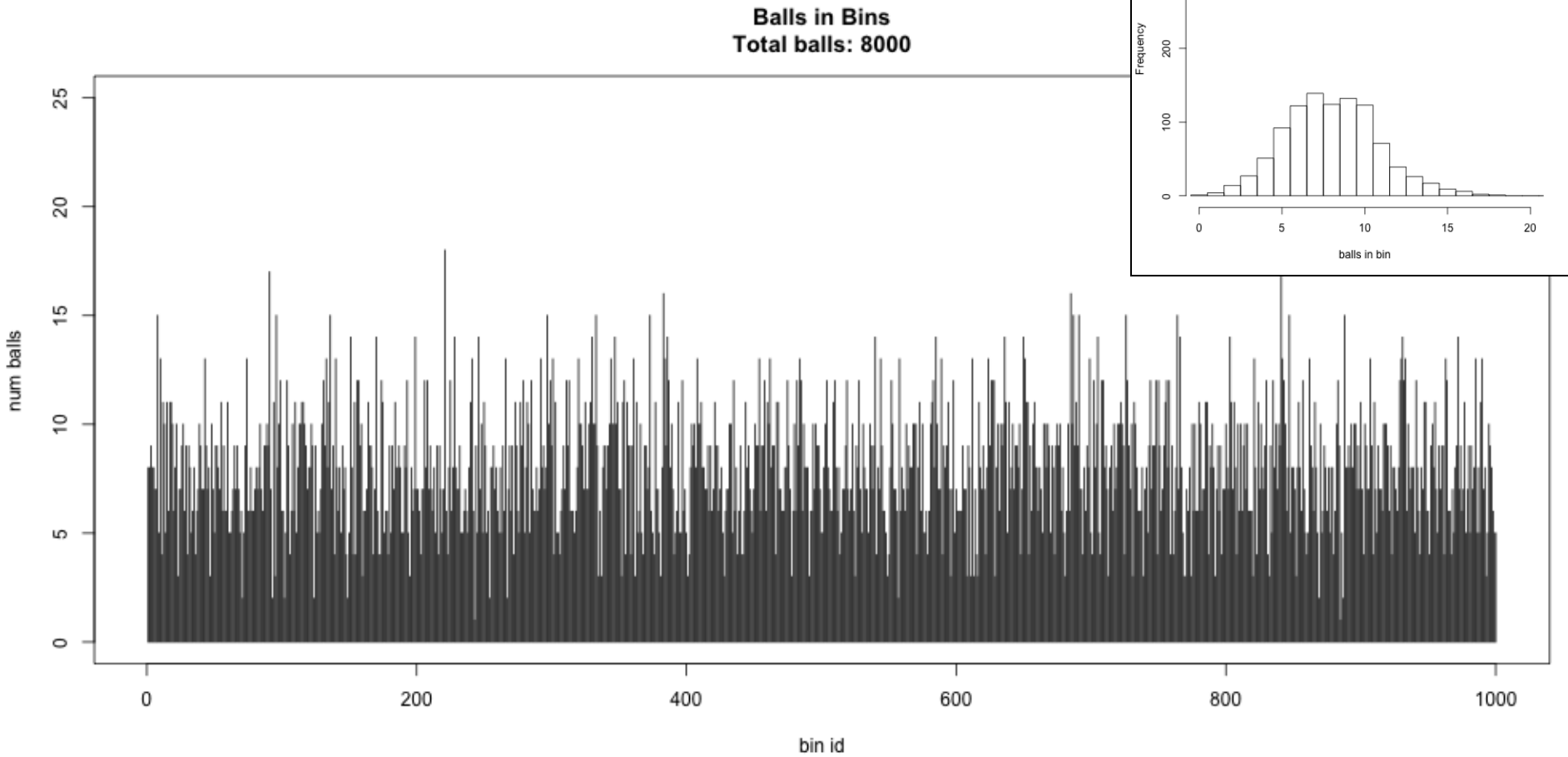
2x sequencing



4x sequencing



8x sequencing



Poisson Distribution

The probability of a given number of events occurring in a fixed interval of time and/or space if these events occur with a known average rate and independently of the time since the last event.

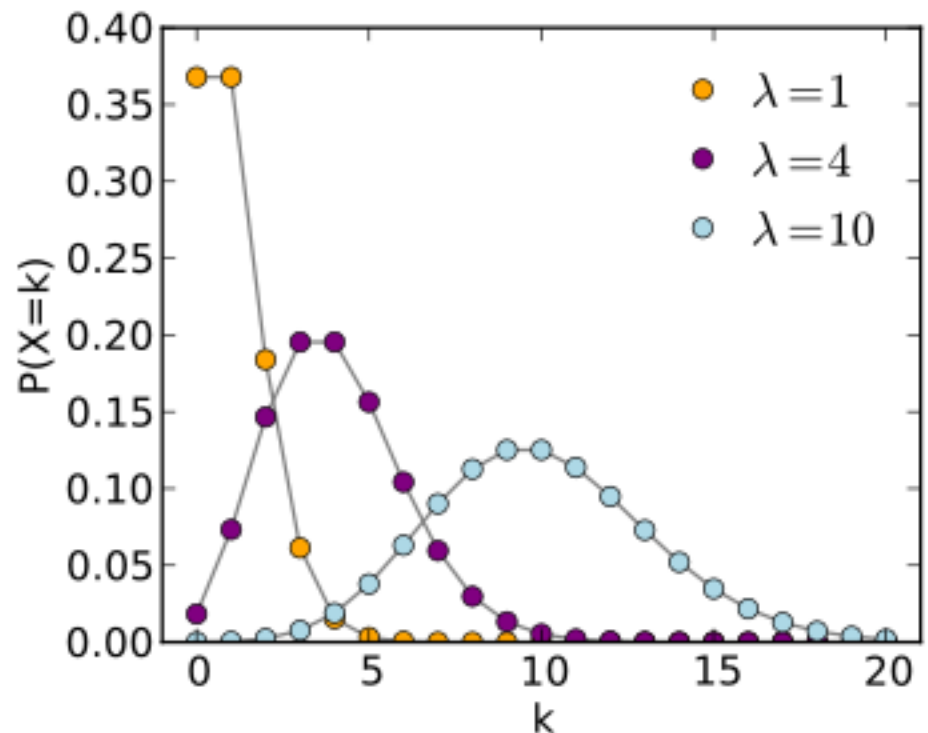
Formulation comes from the limit of the binomial equation

Resembles a normal distribution, but over the positive values, and with only a single parameter.

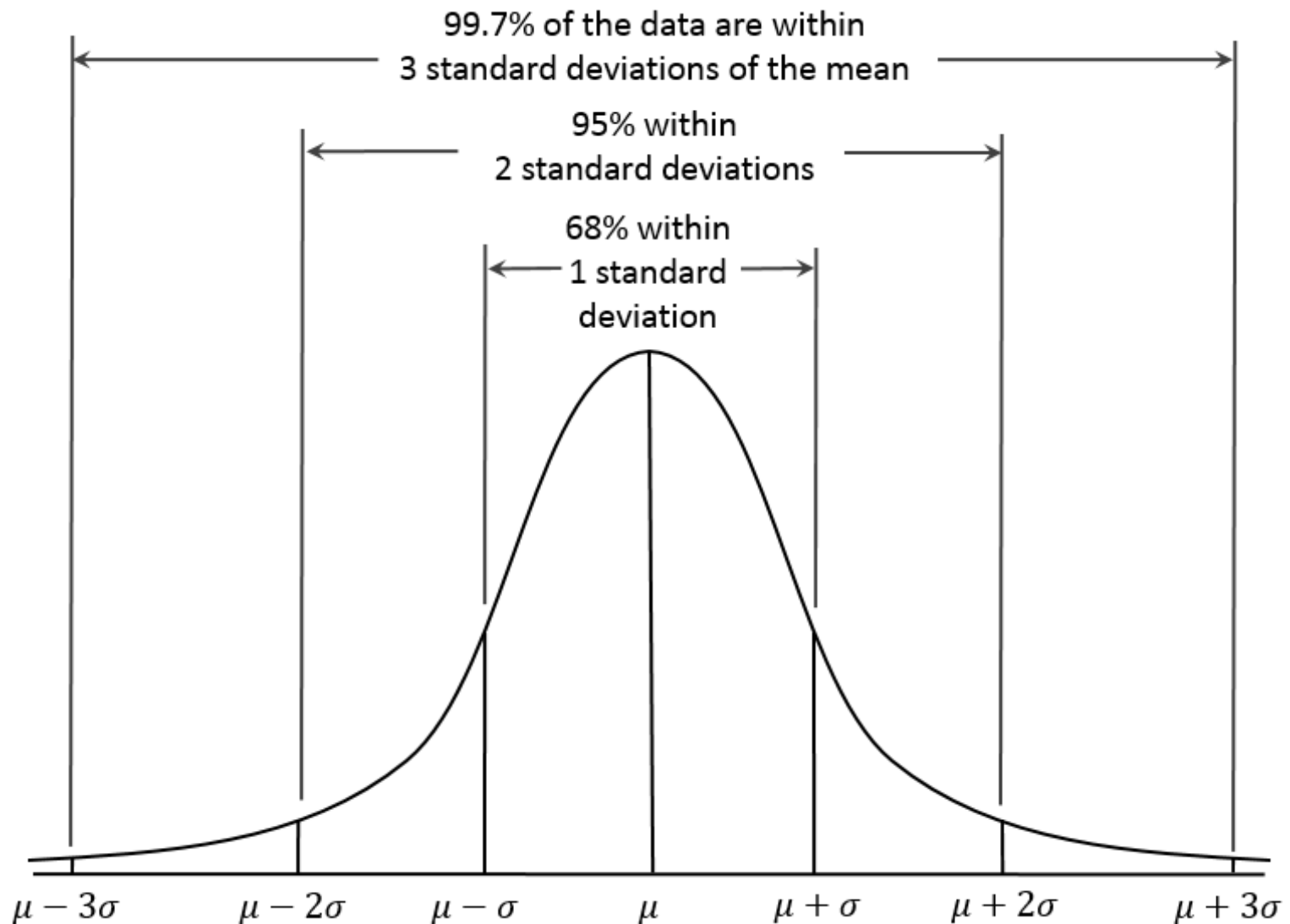
Key properties:

- ***The standard deviation is the square root of the mean.***
- ***For mean > 5, well approximated by a normal distribution***

$$P(k) = \frac{\lambda^k}{k!} e^{-\lambda}$$



Normal Approximation



Can estimate Poisson distribution as a normal distribution when $\lambda > 10$

Pop Quiz!

I want to sequence a 10Mbps genome to 24x coverage.
How many 120bp reads do I need?

I need $10\text{Mbps} \times 24x = 240\text{Mbps}$ of data
 $240\text{Mbps} / 120\text{bp} / \text{read} = 2\text{M}$ reads

I want to sequence a 10Mbps genome so that
>97.5% of the genome has at least 24x coverage.
How many 120bp reads do I need?

Find X such that $X - 2 \times \sqrt{X} = 24$

$$36 - 2 \times \sqrt{36} = 24$$

I need $10\text{Mbps} \times 36x = 360\text{Mbps}$ of data
 $360\text{Mbps} / 120\text{bp} / \text{read} = 3\text{M}$ reads (50% more \$\$\$)

K-mers and K-mer counting

GATTACATACACATTGGATG

K-mers and K-mer counting

GATTACATACACATTGGATG

GAT ACA ACA ATT GAT

ATT CAT CAC TTG ATG

TTA ATA ACA TGG

TAC TAC CAT GGA

Kmers:

- Divide a string into substrings of length k
- Notice every position is covered k times
- Notice there are $G - k + 1$ kmers from a string of length G

K-mers and K-mer counting

GATTACATACACATTGGATG

GAT ACA ACA ATT GAT

ATT CAT CAC TTG ATG

TTA ATA ACA TGG

TAC TAC CAT GGA

Kmers:

- Divide a string into substrings of length k
- Notice every position is covered k times
- Notice there are $G - k + 1$ kmers from a string of length G

Computation: Very easy to compute, exact matches, represent 32mers in 64 bits

Biological: The “atomic unit” of a sequence, creates a fingerprint of a genome/read

K-mers and K-mer counting

GATTACATACACATTGGATG

GAT ACA ACA ATT GAT

ATT CAT CAC TTG ATG

TTA ATA ACA TGG

TAC TAC CAT GGA

GAT : 2 CAT : 2 ATG : 1 TGG : 1

ACA : 3 CAC : 1 TTA : 1 TAC : 2

ATT : 2 TTG : 1 ATA : 1 GGA : 1

K-mers and K-mer counting

GATTACATACACATTGGATG

GAT : 2 CAT : 2 ATG : 1 TGG : 1

ACA : 3 CAC : 1 TTA : 1 TAC : 2

ATT : 2 TTG : 1 ATA : 1 GGA : 1

1 : 7 (ATG , TGG , ...)

2 : 4 (GAT , CAT , ATT , TAC)

3 : 1 (ACA)

See HW1

K-mers and K-mer counting

GATTACATACACATTGGATG

1: 7 (ATG, TGG, ...)

2: 4 (GAT, CAT, ATT, TAC)

3: 1 (ACA)

How long should k be?

K-mers and K-mer counting

GATTACATACACATTGGATG

- 1: 7 (ATG, TGG, ...)
- 2: 4 (GAT, CAT, ATT, TAC)
- 3: 1 (ACA)

How long should k be?

K=1 : Too short, every base is present

K=2 : Too short, every pair of bases will be present

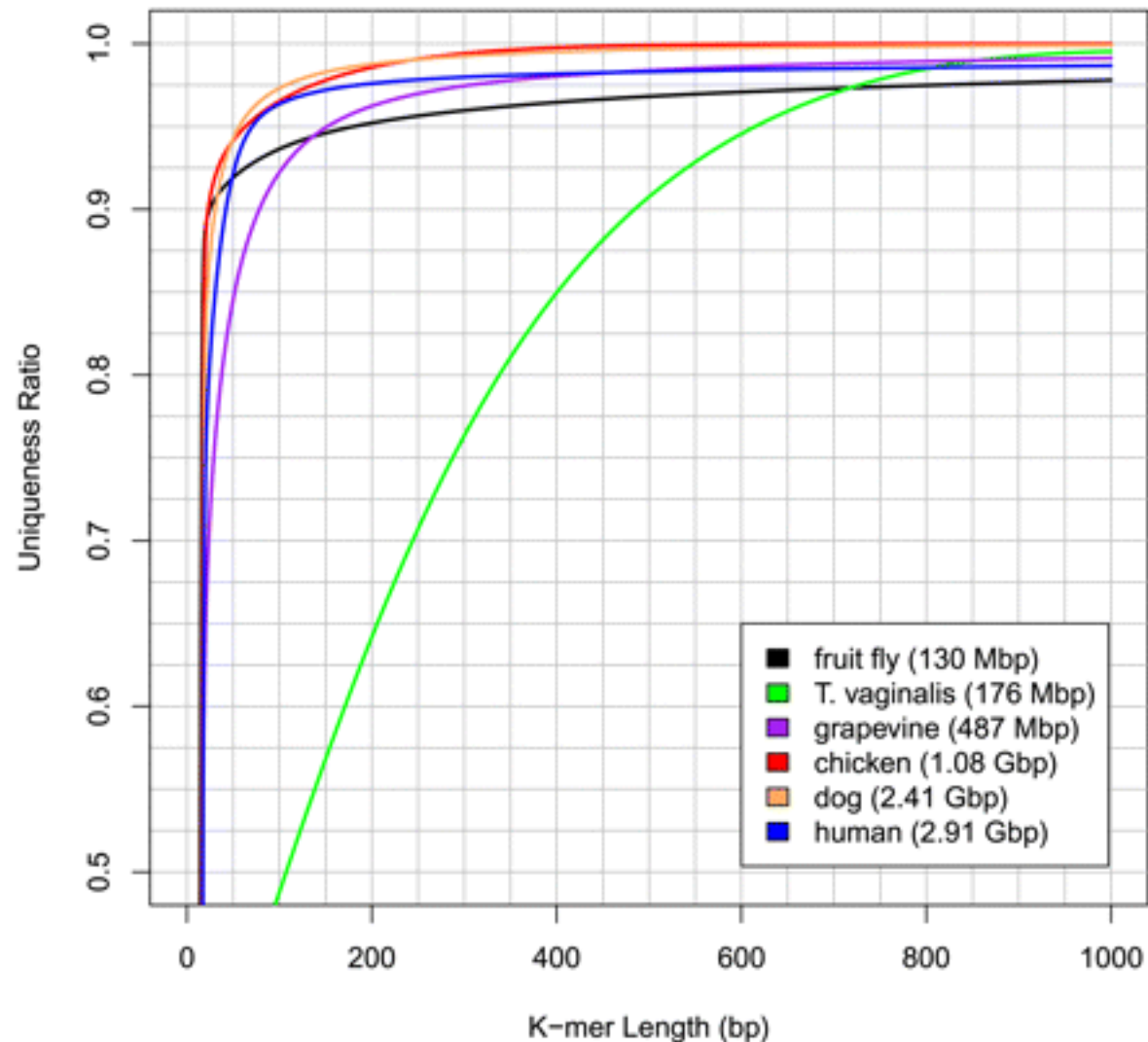
Pick k so that $G/(4^k) \ll 1$

$$k = \log_4 (G)$$

At least 15 for human, often a bit longer

But not too long or could lose resolution

K-mer Uniqueness



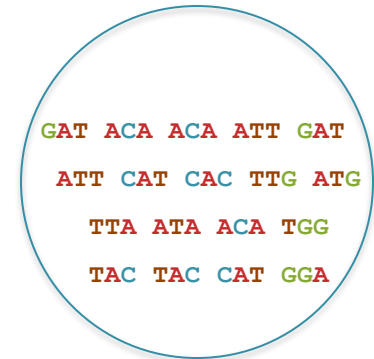
Assembly of large genomes using second-generation sequencing

Schatz et al. (2010) Genome Research. doi: 10.1101/gr.101360.109

GATTACATACACATTGGATG

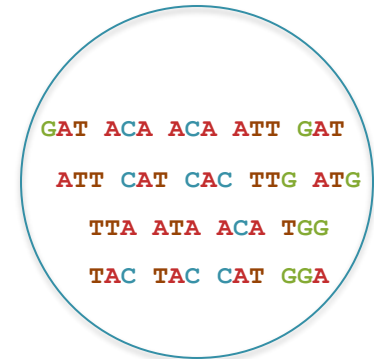
GATTACATACACATTGGATG

GATTACATACACATTGGATG
GAT ACA ACA ATT GAT
ATT CAT CAC TTG ATG
TTA ATA ACA TGG
TAC TAC CAT GGA

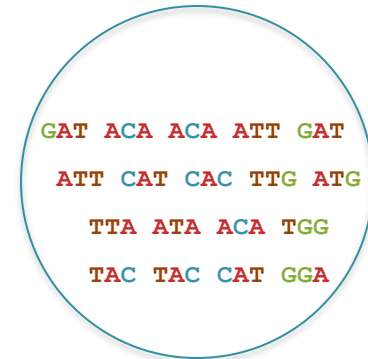


=

GATTACATACACATTGGATG
GAT ACA ACA ATT GAT
ATT CAT CAC TTG ATG
TTA ATA ACA TGG
TAC TAC CAT GGA

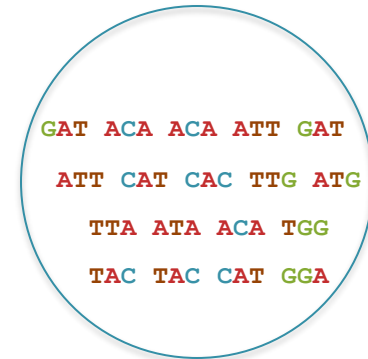


GATTACATACACATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA



=
 =

GATTACATACACATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA



Jaccard Coefficient

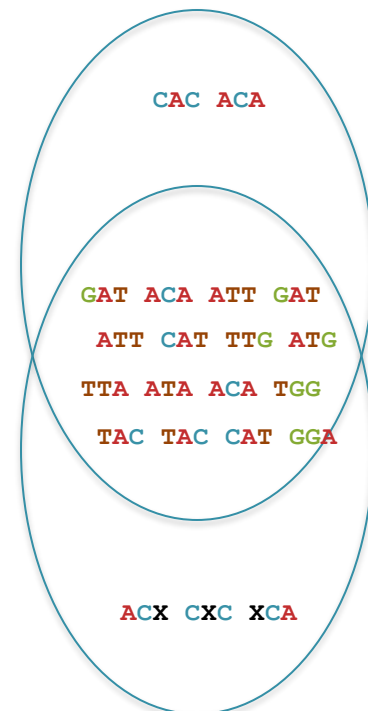
$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{18}{18} = 100\%$$

GATTACATACATTGGATG

GATTACATACXATTGGATG

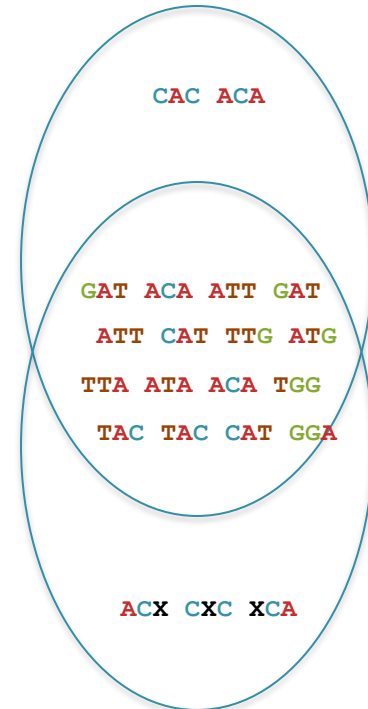
GATTACATACACAATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA

GATTACATACXCAATTGGATG
 GAT ACA ACX ATT GAT
 ATT CAT CX C TTG ATG
 TTA ATA XCA TGG
 TAC TAC CAT GGA



GATTACATACACAATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA

GATTACATACXCAATTGGATG
 GAT ACA ACX ATT GAT
 ATT CAT CX C TTG ATG
 TTA ATA XCA TGG
 TAC TAC CAT GGA

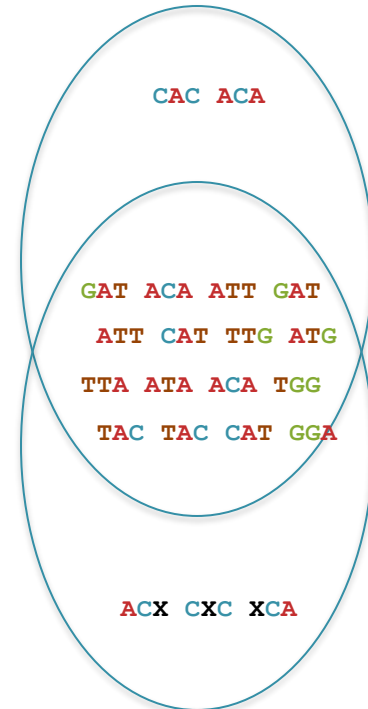


Jaccard Coefficient

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{16}{21} = 76\%$$

GATTACATACACAATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA

GATTACATACXCAATTGGATG
 GAT ACA ACX ATT GAT
 ATT CAT CX C TTG ATG
 TTA ATA XCA TGG
 TAC TAC CAT GGA



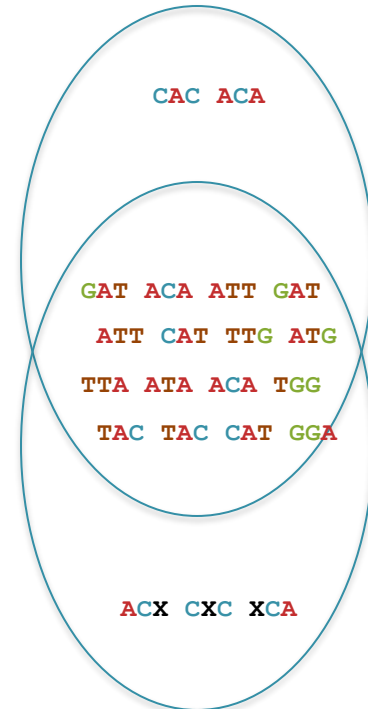
$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{16}{21} = 76\%$$

$$J \approx \frac{p}{2 - p}$$

p = prob(kmer shared)

GATTACATACACAATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA

GATTACATACXCAATTGGATG
 GAT ACA ACX ATT GAT
 ATT CAT CX C TTG ATG
 TTA ATA XCA TGG
 TAC TAC CAT GGA

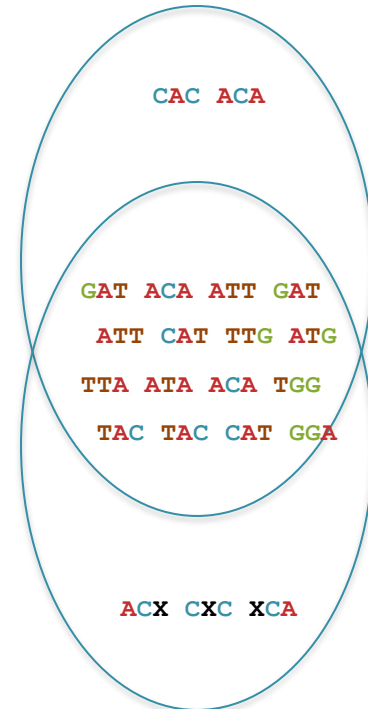


From Jaccard to Average Nucleotide Identity

$$J \approx \frac{p}{2 - p} \implies p = \frac{2J}{1 + J}$$

GATTACATACACAATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA

GATTACATACXCAATTGGATG
 GAT ACA ACX ATT GAT
 ATT CAT CX C TTG ATG
 TTA ATA XCA TGG
 TAC TAC CAT GGA



From Jaccard to Average Nucleotide Identity

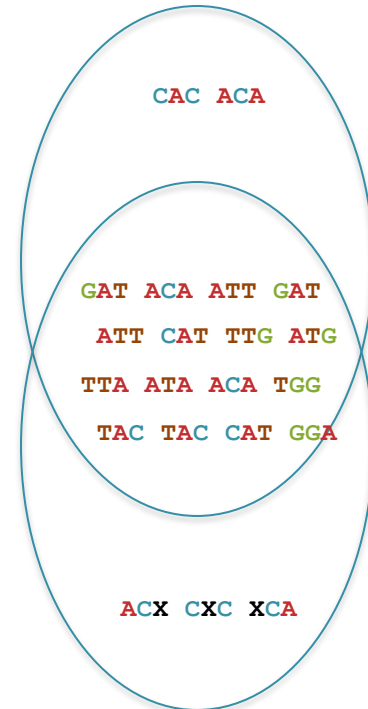
$$J \approx \frac{p}{2 - p} \implies p = \frac{2J}{1 + J}$$

$$p = I^k$$

$$I = \text{ANI} = p^{1/k} = \left(\frac{2J}{1 + J} \right)^{1/k}$$

GATTACATACACAATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA

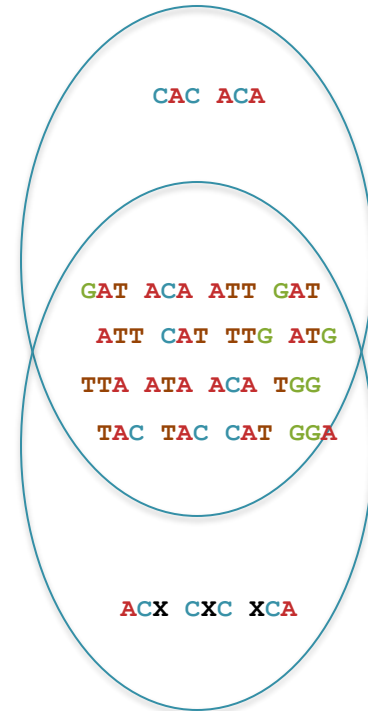
GATTACATACXCAATTGGATG
 GAT ACA ACX ATT GAT
 ATT CAT CX C TTG ATG
 TTA ATA XCA TGG
 TAC TAC CAT GGA



Identity for small differences (d): $p = I^k = (1 - d)^k \approx e^{-kd}$

GATTACATACACAATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA

GATTACATACXCAATTGGATG
 GAT ACA ACX ATT GAT
 ATT CAT CX C TTG ATG
 TTA ATA XCA TGG
 TAC TAC CAT GGA



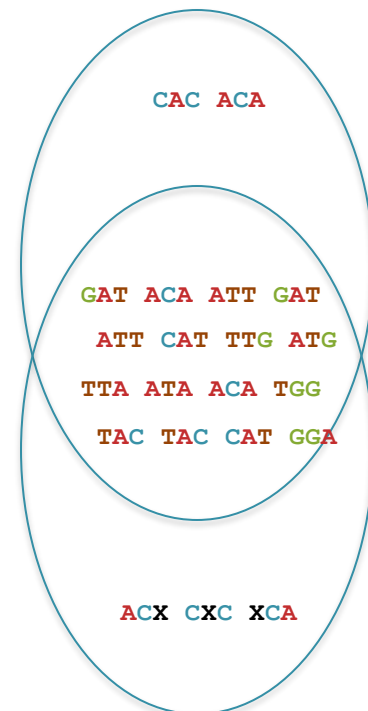
Identity for small differences (d): $p = I^k = (1 - d)^k \approx e^{-kd}$

$$p \approx e^{-kd} \Rightarrow d \approx -\frac{1}{k} \ln p \Rightarrow \text{ANI} = 1 - d \approx 1 + \frac{1}{k} \ln p.$$

$$\text{ANI} \approx 1 + \frac{1}{k} \ln \left(\frac{2J}{1 + J} \right)$$

GATTACATACACAATTGGATG
 GAT ACA ACA ATT GAT
 ATT CAT CAC TTG ATG
 TTA ATA ACA TGG
 TAC TAC CAT GGA

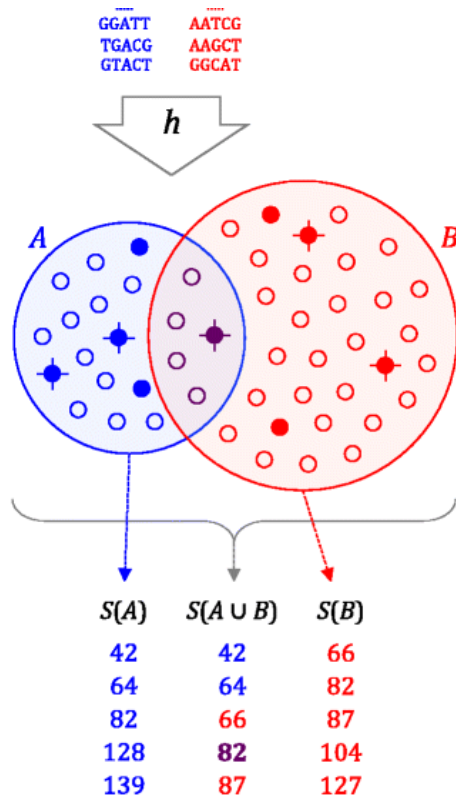
GATTACATACXCAATTGGATG
 GAT ACA ACX ATT GAT
 ATT CAT CX C TTG ATG
 TTA ATA XCA TGG
 TAC TAC CAT GGA



$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{16}{21} = 76\%$$

$$\text{ANI} \approx 1 + \frac{1}{k} \ln \left(\frac{2J}{1+J} \right)$$

ANI = 95.1%



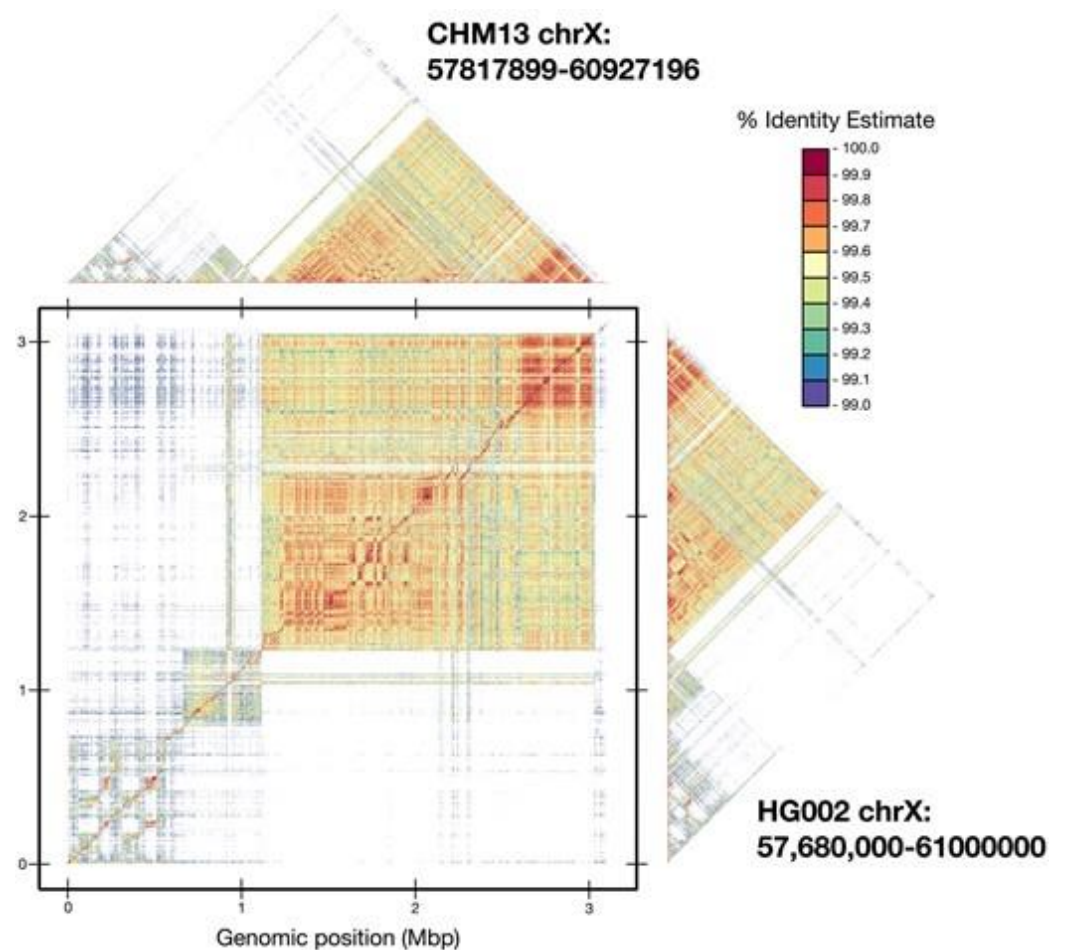
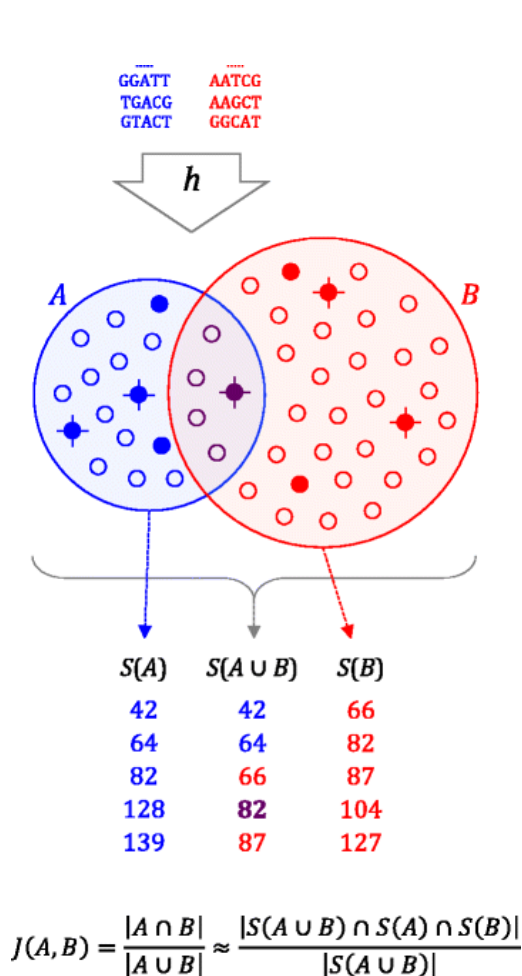
$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \approx \frac{|S(A \cup B) \cap S(A) \cap S(B)|}{|S(A \cup B)|}$$

Mash: fast genome and metagenome distance estimation using MinHash

Ondov et al (2016) Genome Biology. <https://doi.org/10.1186/s13059-016-0997-x>

ModDotPlot—rapid and interactive visualization of tandem repeats

Sweeten, Schatz, Phillippy (2024) Bioinformatics. <https://doi.org/10.1093/bioinformatics/btae493>



Mash: fast genome and metagenome distance estimation using MinHash

Ondov et al (2016) Genome Biology. <https://doi.org/10.1186/s13059-016-0997-x>

ModDotPlot—rapid and interactive visualization of tandem repeats

Sweeten, Schatz, Phillippy (2024) Bioinformatics. <https://doi.org/10.1093/bioinformatics/btae493>



Part 2: De novo genome assembly

Outline

1. Assembly theory

- Assembly by analogy

2. Practical Issues

- Coverage, read length, errors, and repeats

3. Whole Genome Alignment

- MUMmer recommended



Shredded Book Reconstruction

- Dickens accidentally shreds the first printing of A Tale of Two Cities
 - Text printed on 5 long spools

It was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness,	...
--------	-----	------	----	--------	----	-----	-----	-------	----	--------	----	-----	-----	-----	----	---------	----	-----	-----	-----	----	--------------	-----

It was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness,	...
--------	-----	------	----	--------	----	-----	-----	-------	----	--------	----	-----	-----	-----	----	---------	----	-----	-----	-----	----	--------------	-----

It was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness,	...
--------	-----	------	----	--------	----	-----	-----	-------	----	--------	----	-----	-----	-----	----	---------	----	-----	-----	-----	----	--------------	-----

It was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness,	...
--------	-----	------	----	--------	----	-----	-----	-------	----	--------	----	-----	-----	-----	----	---------	----	-----	-----	-----	----	--------------	-----

It	was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness,	...
----	-----	-----	------	----	--------	----	-----	-----	-------	----	--------	----	-----	-----	-----	----	---------	----	-----	-----	-----	----	--------------	-----

- How can he reconstruct the text?
 - 5 copies x 138,656 words / 5 words per fragment = 138k fragments
 - The short fragments from every copy are mixed together
 - Some fragments are identical

Greedy Reconstruction

It was the best of

age of wisdom, it was

best of times, it was

it was the age of

it was the age of

it was the worst of

of times, it was the

of times, it was the

of wisdom, it was the

the age of wisdom, it

the best of times, it

the worst of times, it

times, it was the age

times, it was the worst

was the age of wisdom,

was the age of foolishness,

was the best of times,

was the worst of times,

wisdom, it was the age

worst of times, it was

It was the best of

was the best of times,

the best of times, it

best of times, it was

of times, it was the

of times, it was the

times, it was the worst

times, it was the age

The repeated sequence make the correct reconstruction ambiguous

- It was the best of times, it was the [worst/age]

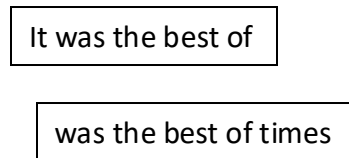
Model the assembly problem as a graph problem

How long will it take to compute the overlaps?

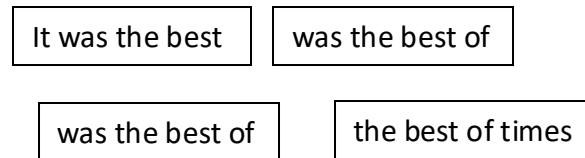
de Bruijn Graph Construction

- $G_k = (V, E)$
 - V = Length- k sub-fragments
 - E = Directed edges between consecutive sub-fragments
 - Sub-fragments overlap by $k-1$ words

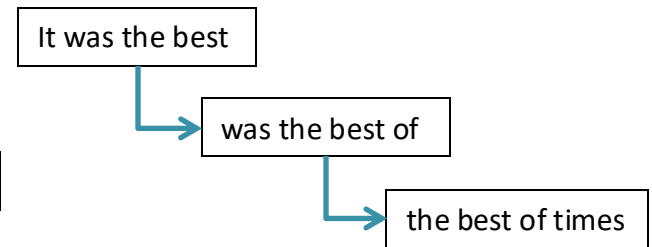
Fragments $|f|=5$



Sub-fragment $k=4$



Directed edges (overlap by $k-1$)



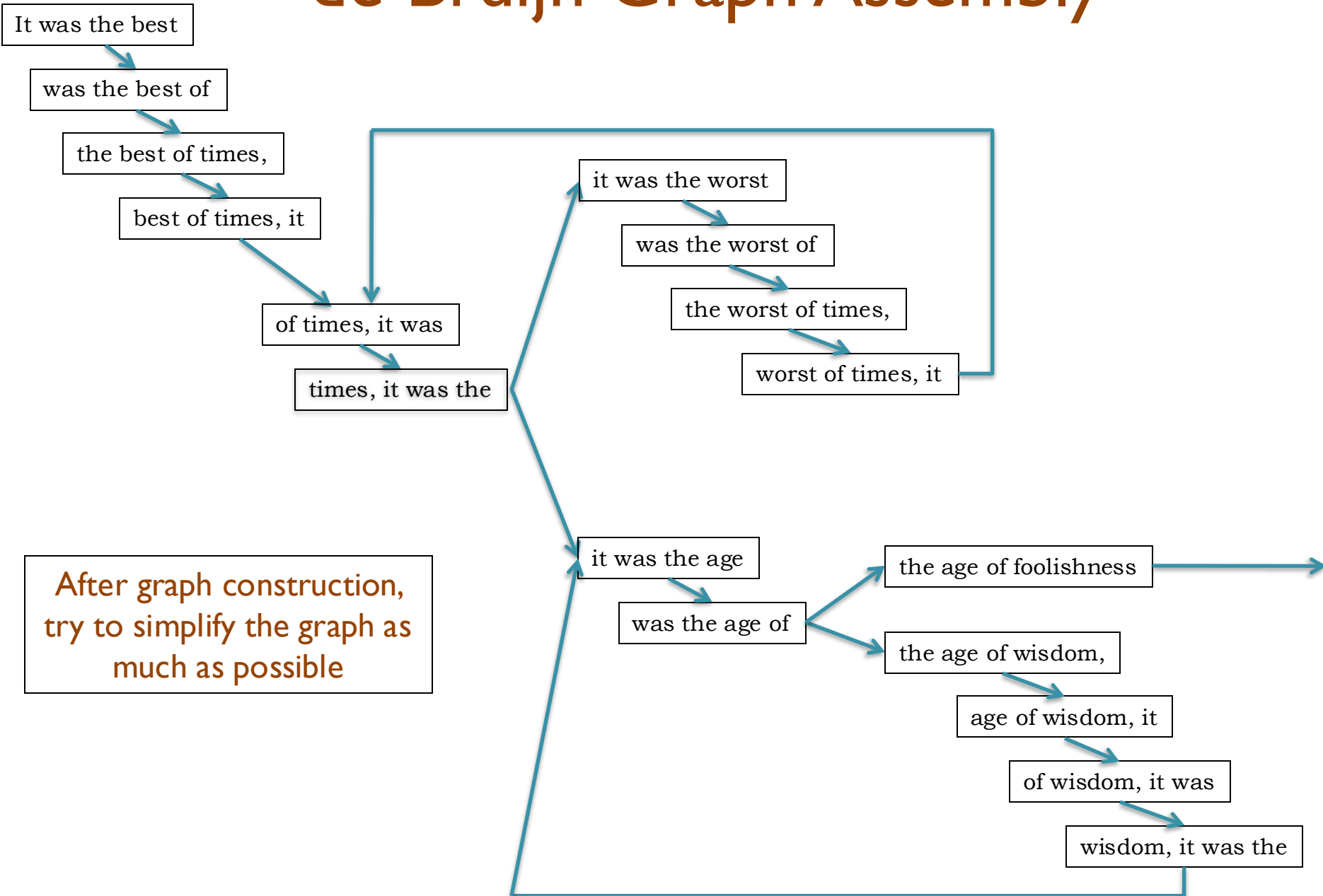
– Overlaps between fragments are implicitly computed

How to pronounce:

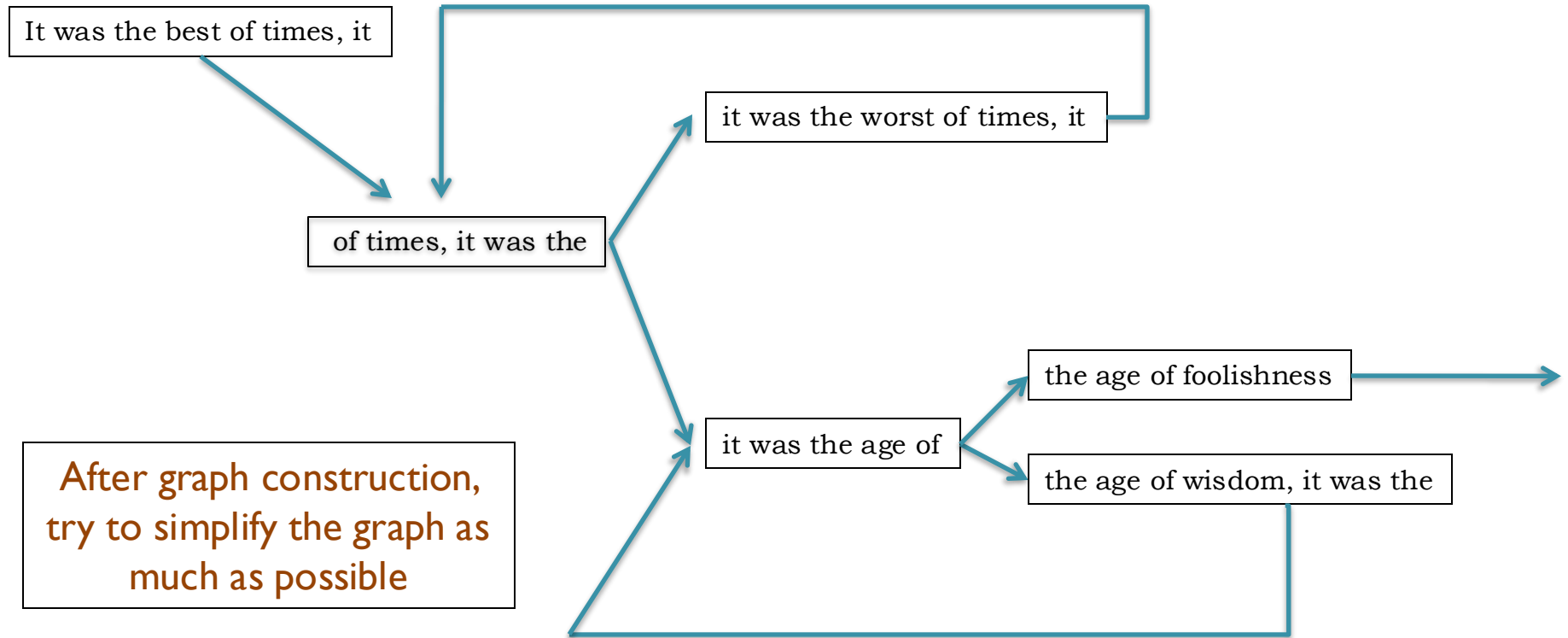
https://forvo.com/word/de_bruijn/

de Bruijn, 1946
Idury et al., 1995
Pevzner et al., 2001

de Bruijn Graph Assembly



de Bruijn Graph Assembly



The full tale

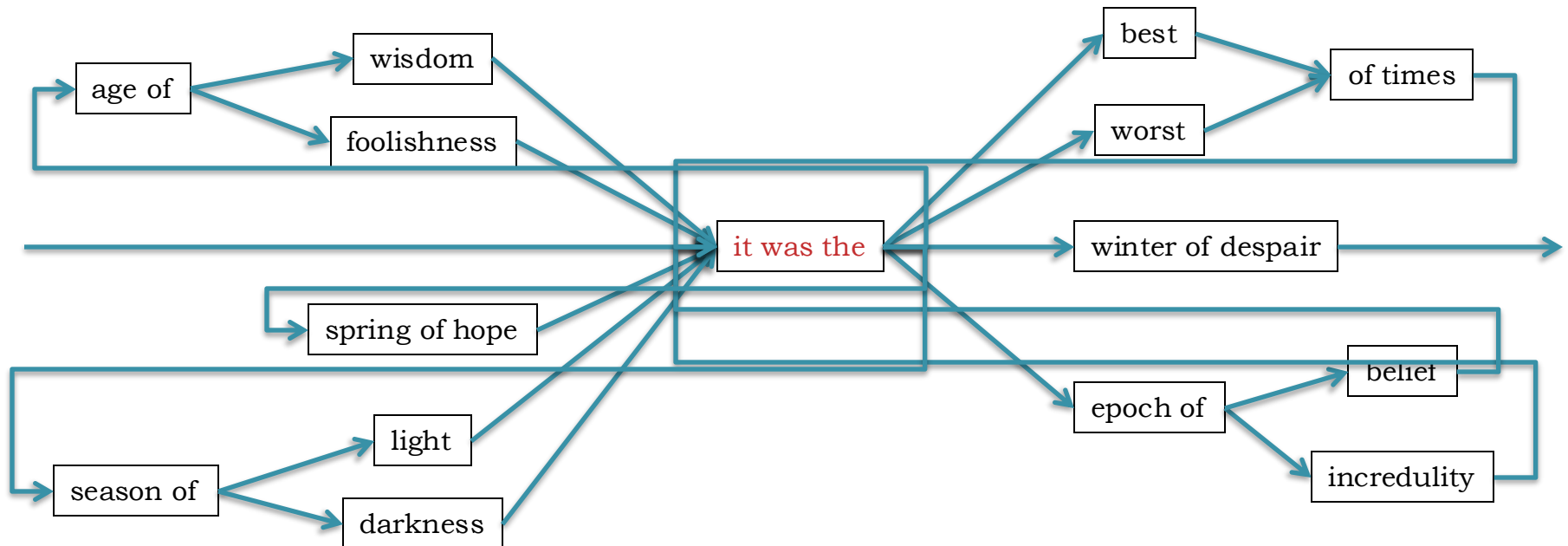
... it was the best of times it was the worst of times ...

... it was the age of wisdom it was the age of foolishness ...

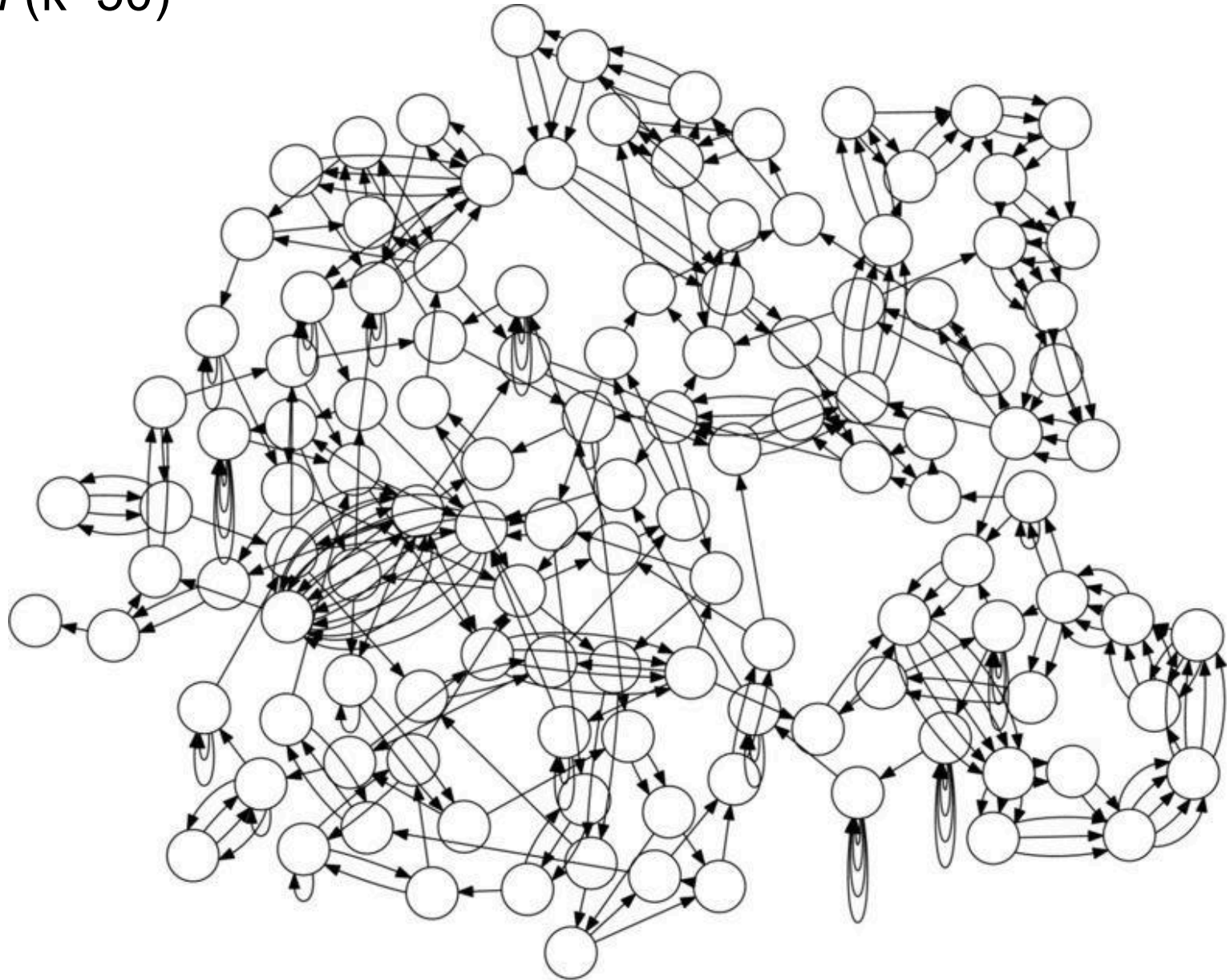
... it was the epoch of belief it was the epoch of incredulity ...

... it was the season of light it was the season of darkness ...

... it was the spring of hope it was the winder of despair ...



E. coli (k=50)



Reducing assembly complexity of microbial genomes with single-molecule sequencing

Koren et al (2013) Genome Biology. 14:R101 <https://doi.org/10.1186/gb-2013-14-9-r101>

Contig N50

Def: 50% of the genome is in contigs as large as the N50 value

Example: 1 Mbp genome

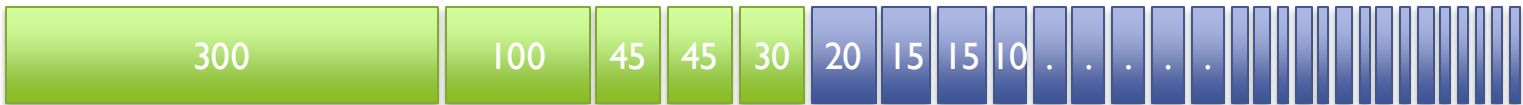
50%



1000



A



N50 size = 30 kbp

B



N50 size = 3 kbp

Contig N50

Def: 50% of the genome is in contigs as large as the N50 value

50%

Better N50s improves the analysis in every dimension

- Better resolution of genes and flanking regulatory regions
- Better resolution of transposons and other complex sequences
- Better resolution of chromosome organization
- Better sequence for all downstream analysis

Just be careful of N50 inflation!

- A very very very bad assembler in 1 line of bash:
- `cat *.reads.fa > genome.fa`

N50 size = 3 kbp

Pop Quiz I

Assemble these reads using a de Bruijn graph approach ($k=3$):

ACACG

ATTAC

GATTA

TTACA

Pop Quiz I

Assemble these reads using a de Bruijn graph approach (k=3):

ACACG : ACA → CAC → ACG
ATTAC : ATT → TTA → TAC
GATTA : GAT → ATT → TTA
TTACA : TTA → TAC → ACA

Pop Quiz I

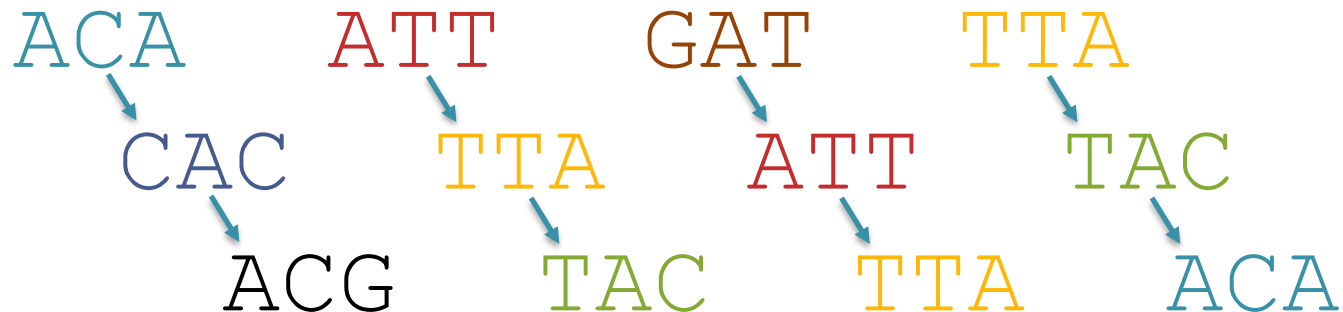
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG: ACA → CAC → ACG

ATTAC: ATT → TTA → TAC

GATTA: GAT → ATT → TTA

TTACA: TTA → TAC → ACA



Pop Quiz I

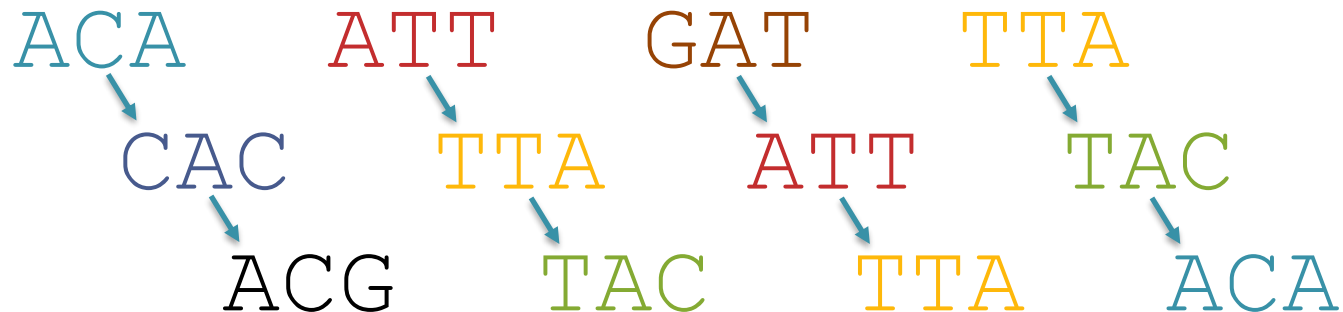
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA



Pop Quiz I

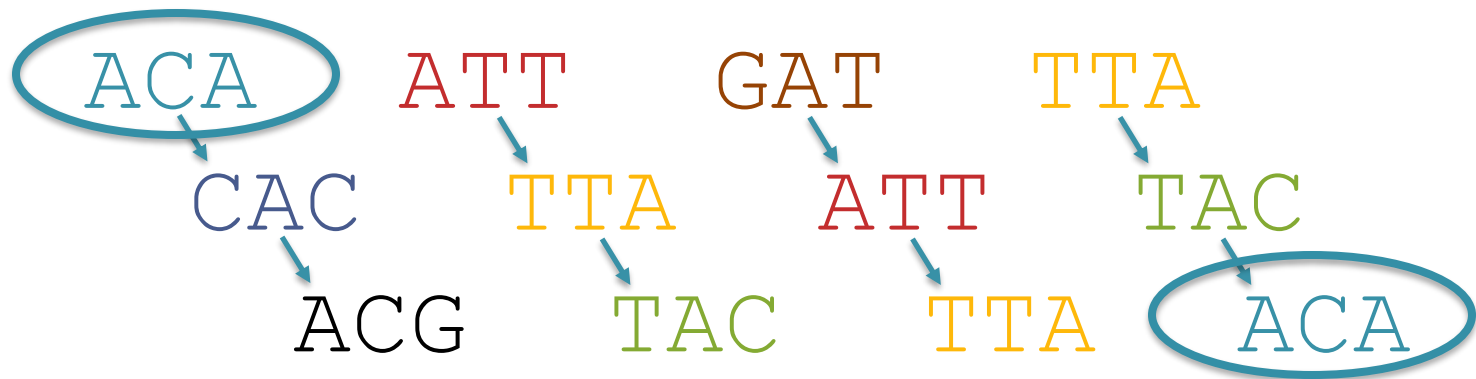
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA



Pop Quiz I

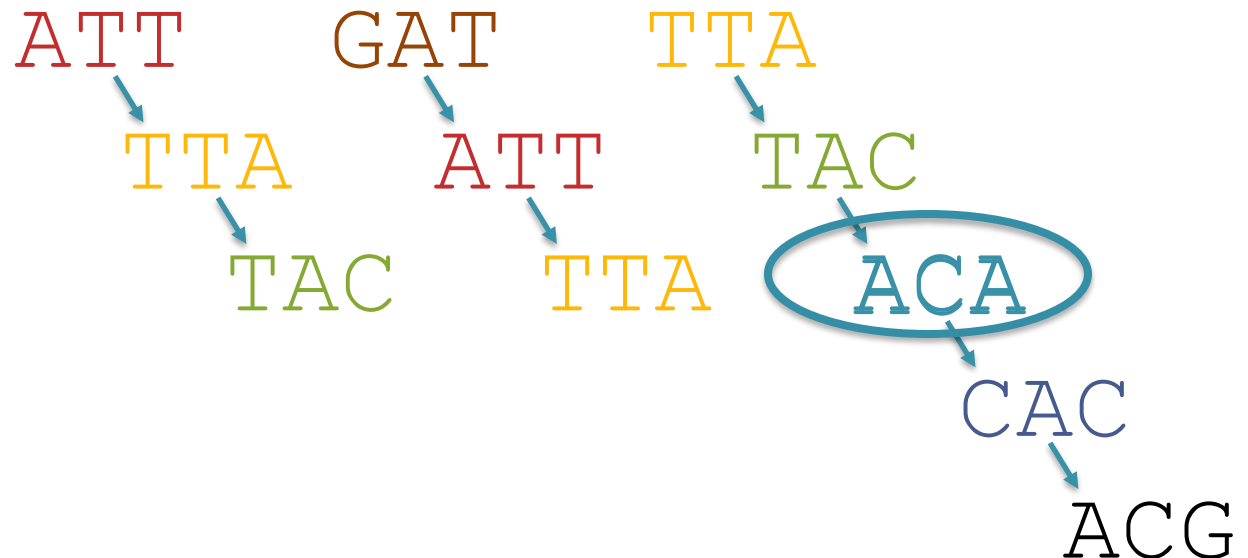
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA



Pop Quiz I

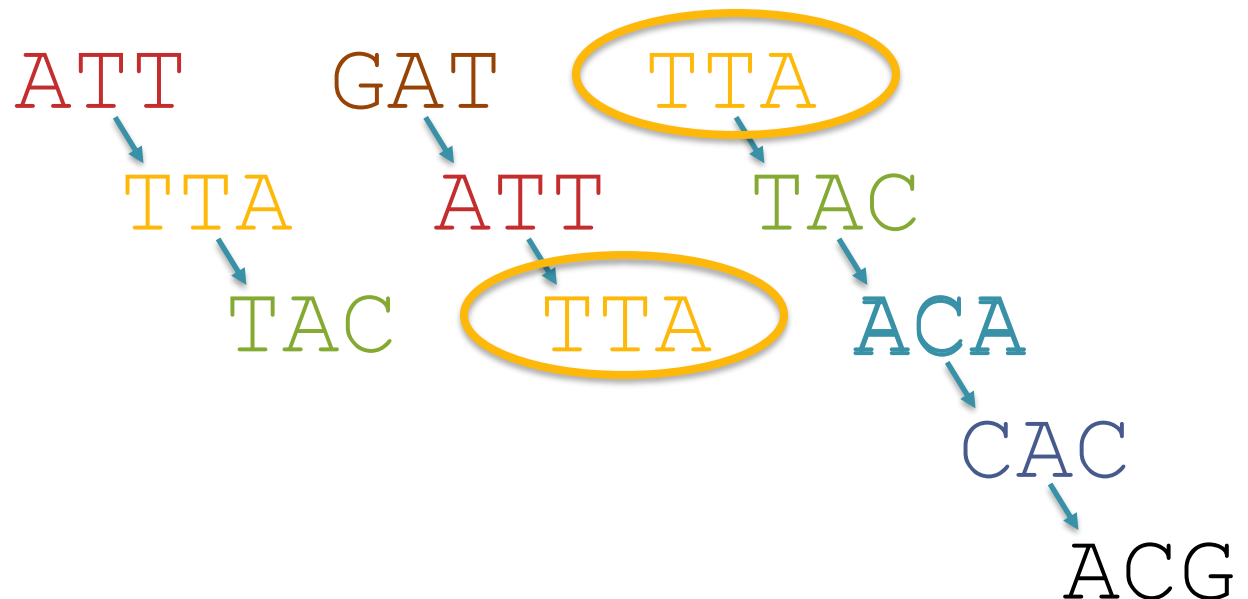
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA



Pop Quiz I

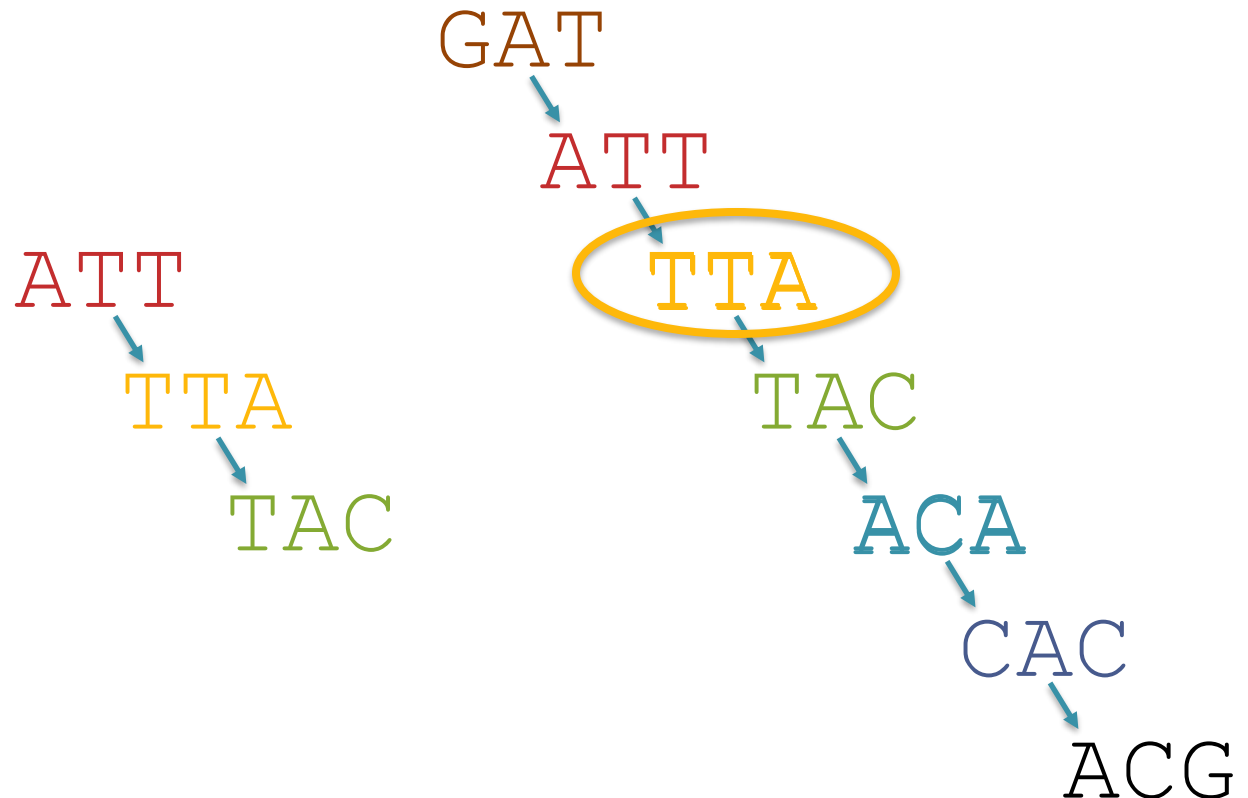
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA



Pop Quiz I

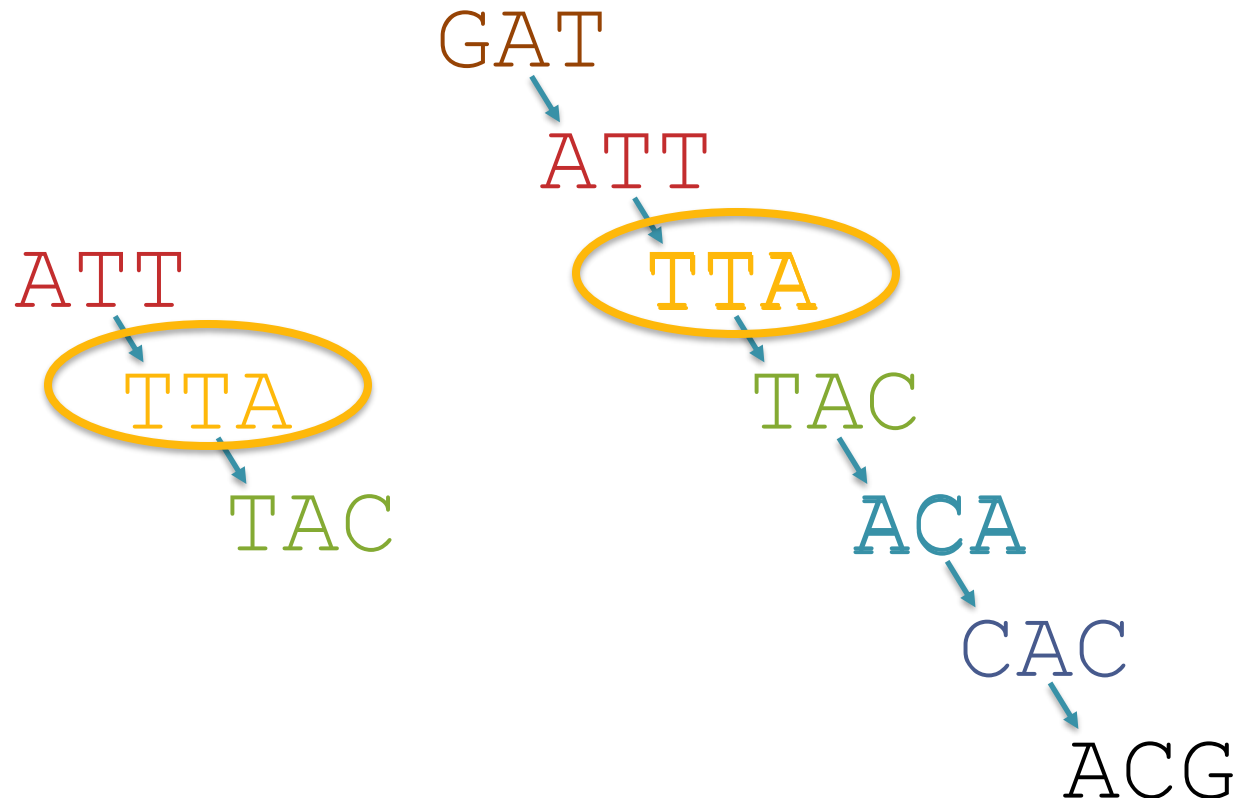
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA



Pop Quiz I

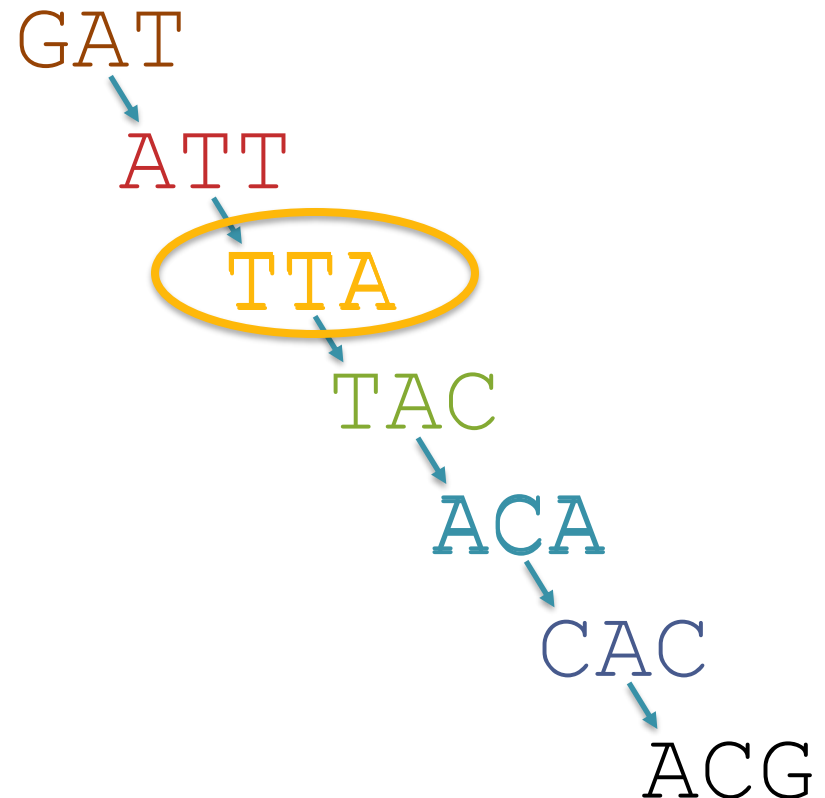
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA



Pop Quiz I

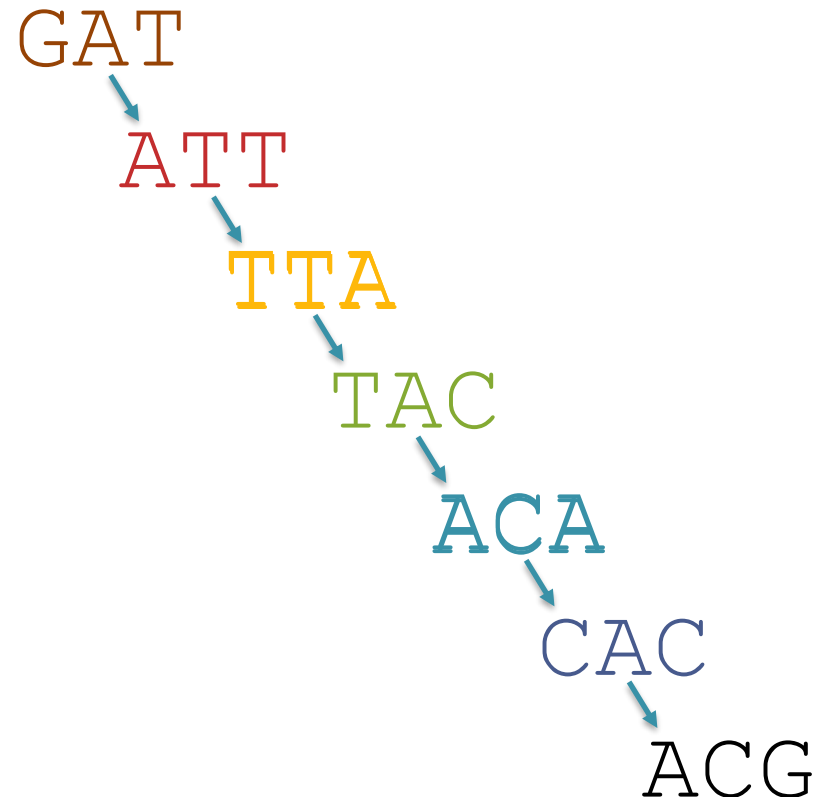
Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA



Pop Quiz I

Assemble these reads using a de Bruijn graph approach (k=3):

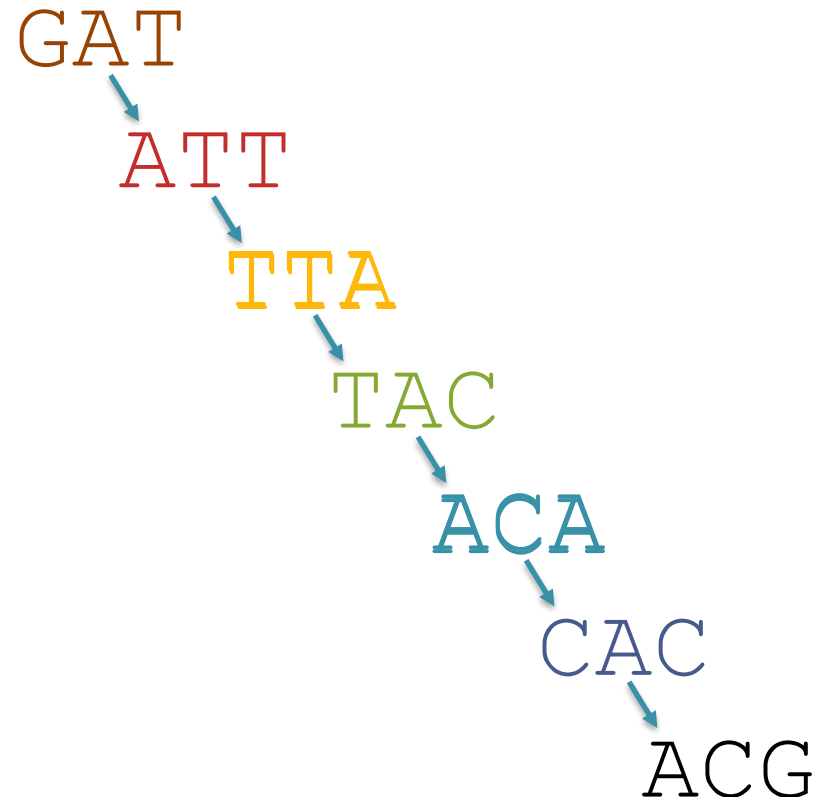
ACACG

ATTAC

GATTA

TTACA

GATTACACG



Pop Quiz I

Assemble these reads using a de Bruijn graph approach (k=3):

ACACG

ATTAC

GATTA

TTACA

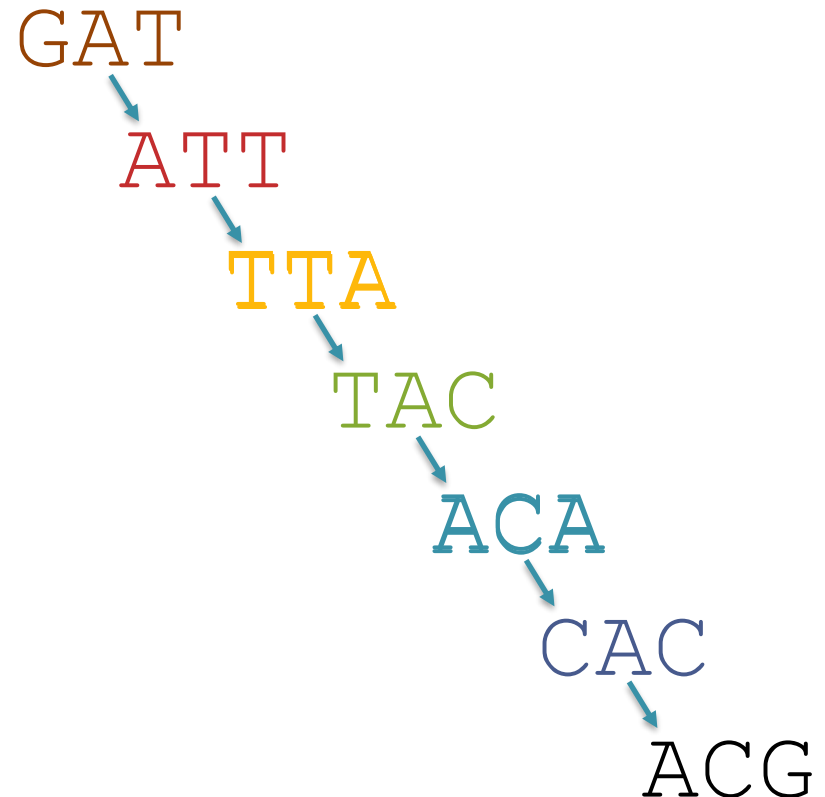
GATTACACG

GATTA

ATTAC

TTACA

ACACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach ($k=3$):

ACGA

ACGT

ATAC

CGAC

CGTA

GACG

GTAT

TACG

Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

ACGT

ATAC

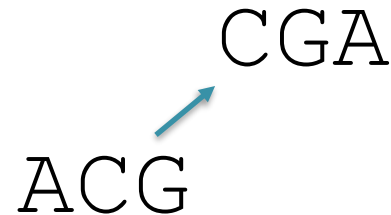
CGAC

CGTA

GACG

GTAT

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

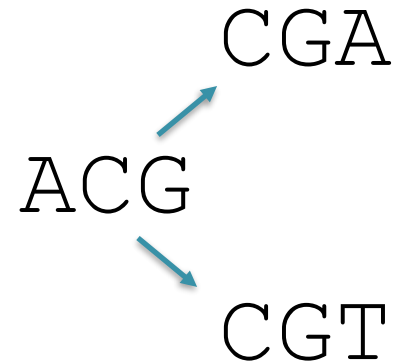
CGAC

CGTA

GACG

GTAT

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

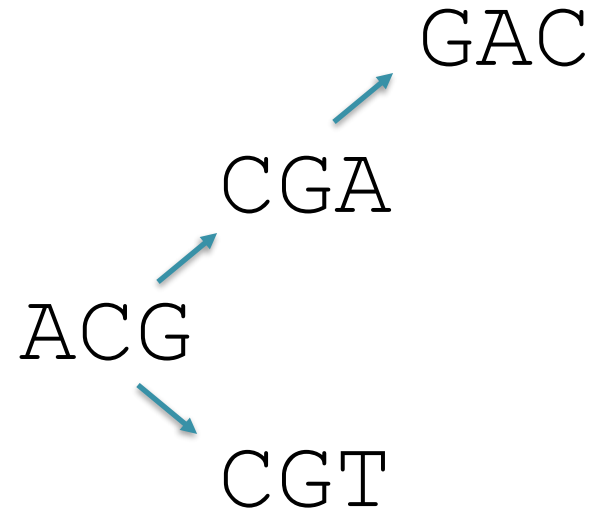
~~CGAC~~

CGTA

GACG

GTAT

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

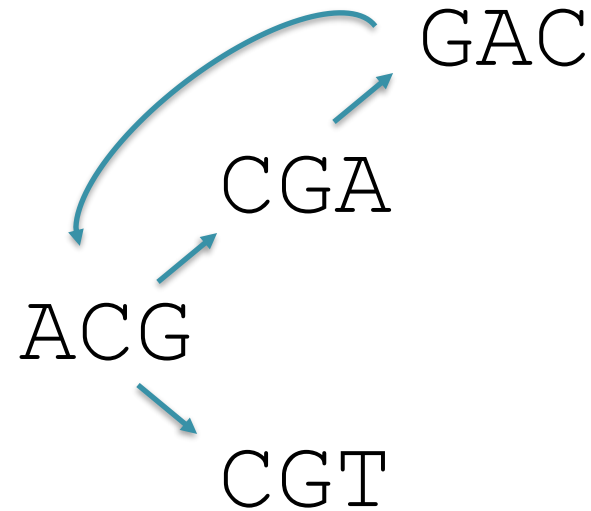
~~CGAC~~

CGTA

~~GACG~~

GTAT

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

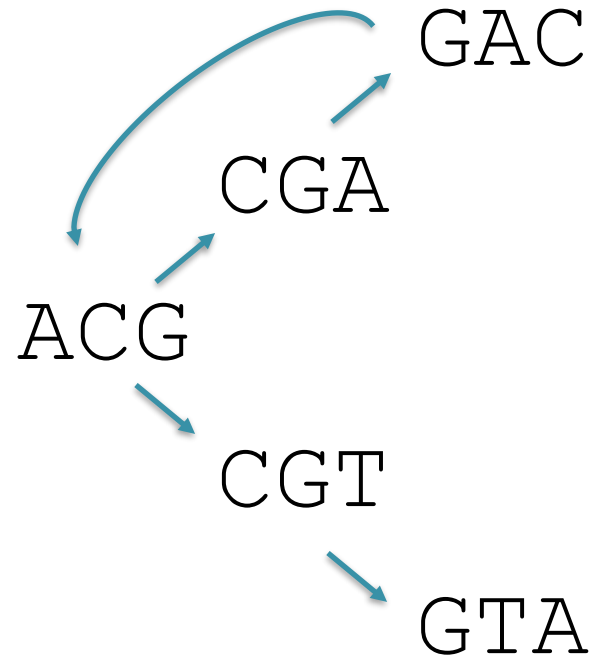
~~CGAC~~

~~CGTA~~

~~GACG~~

GTAT

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

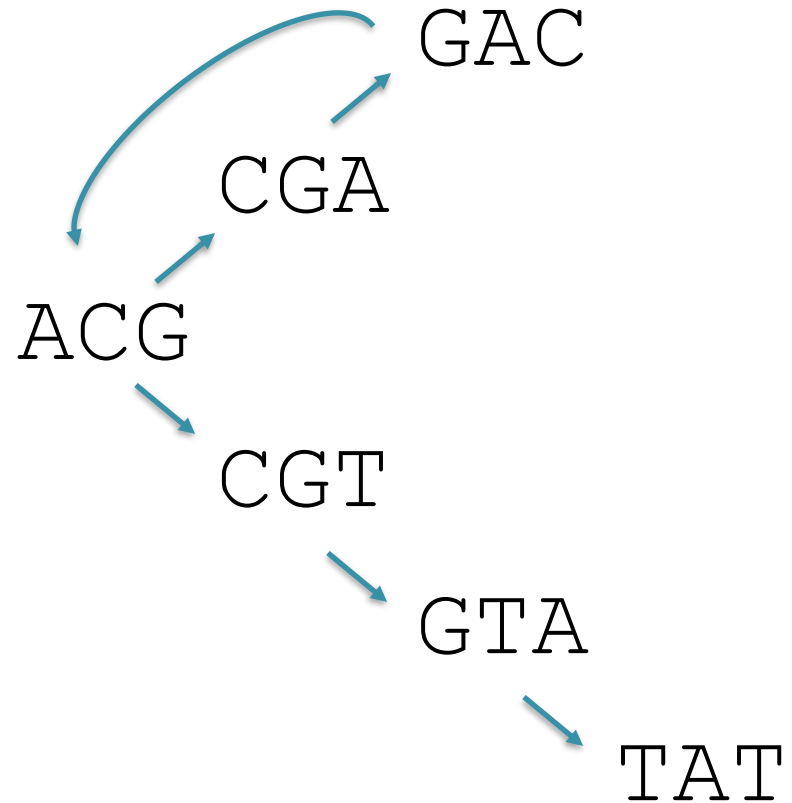
~~CGAC~~

~~CGTA~~

~~GACG~~

~~GTAT~~

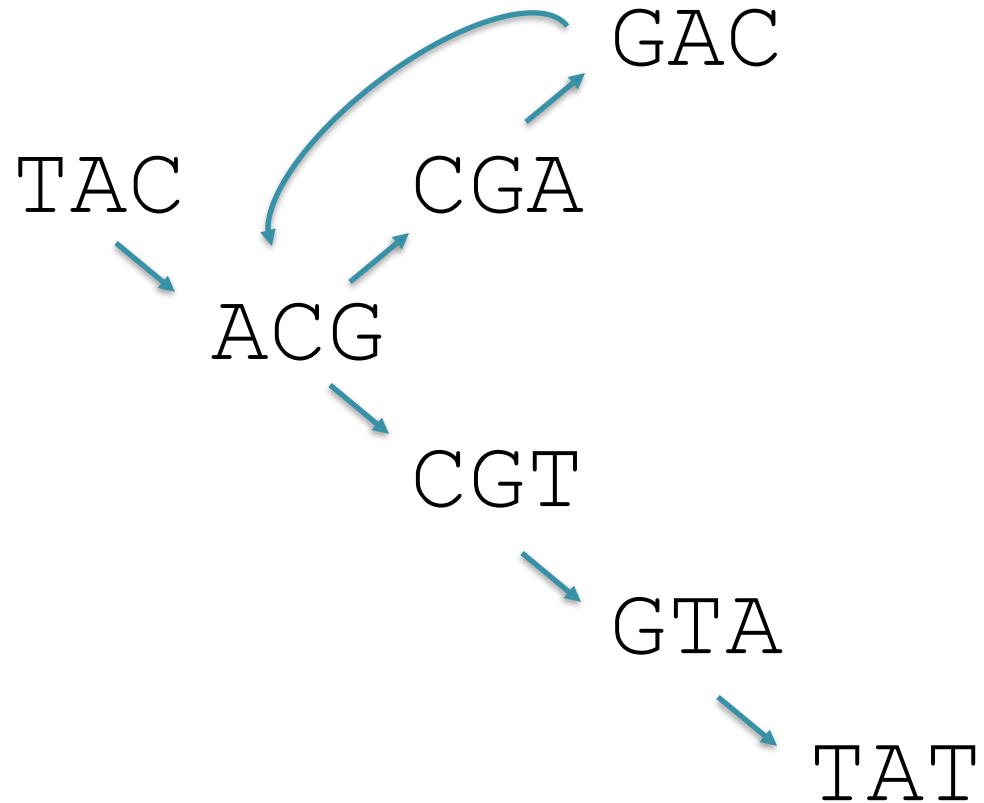
TACG



Pop Quiz 2

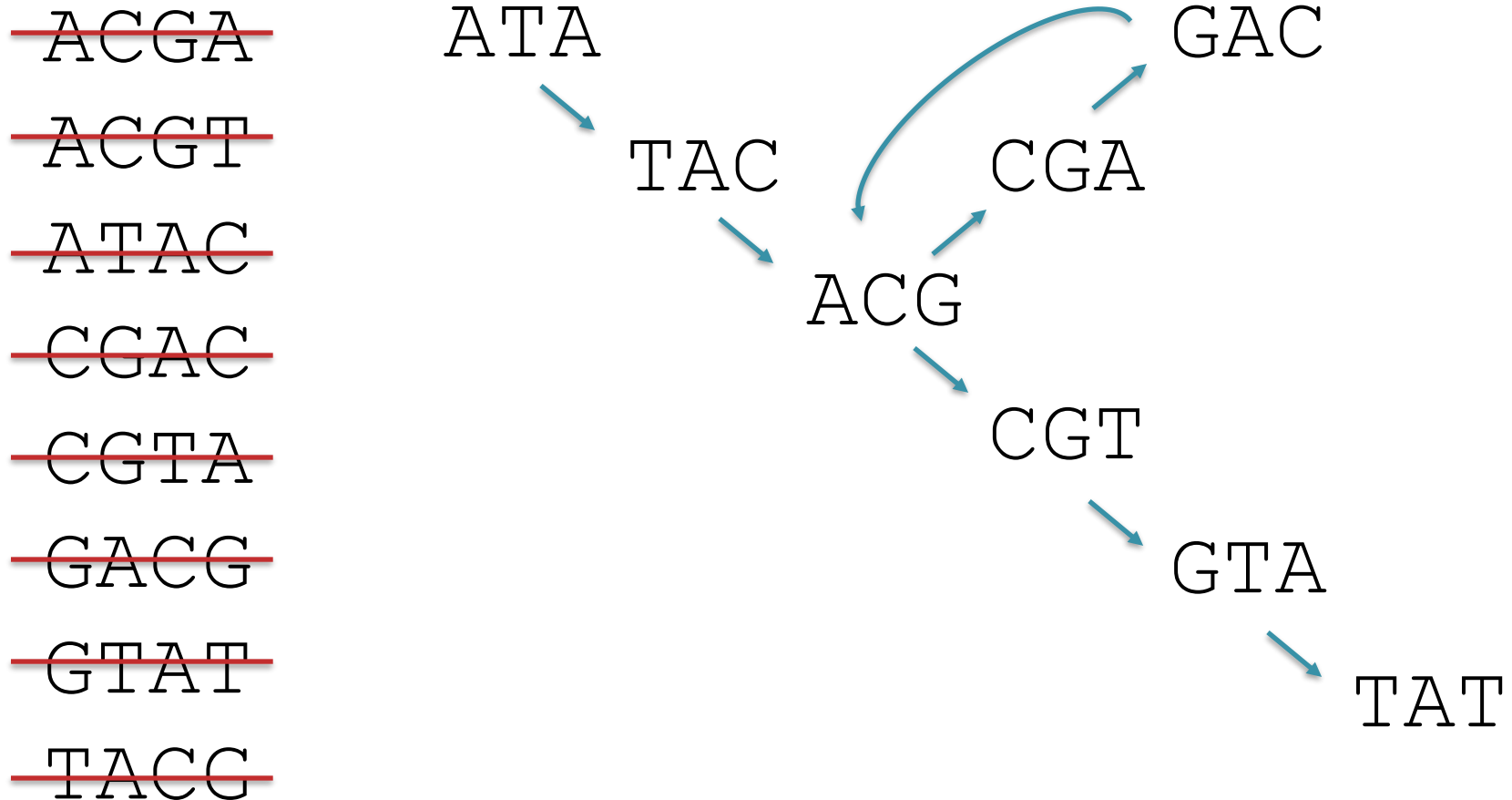
Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~
~~ACGT~~
ATAC
~~CGAC~~
~~CGTA~~
~~GACG~~
~~GTAT~~
~~TACG~~



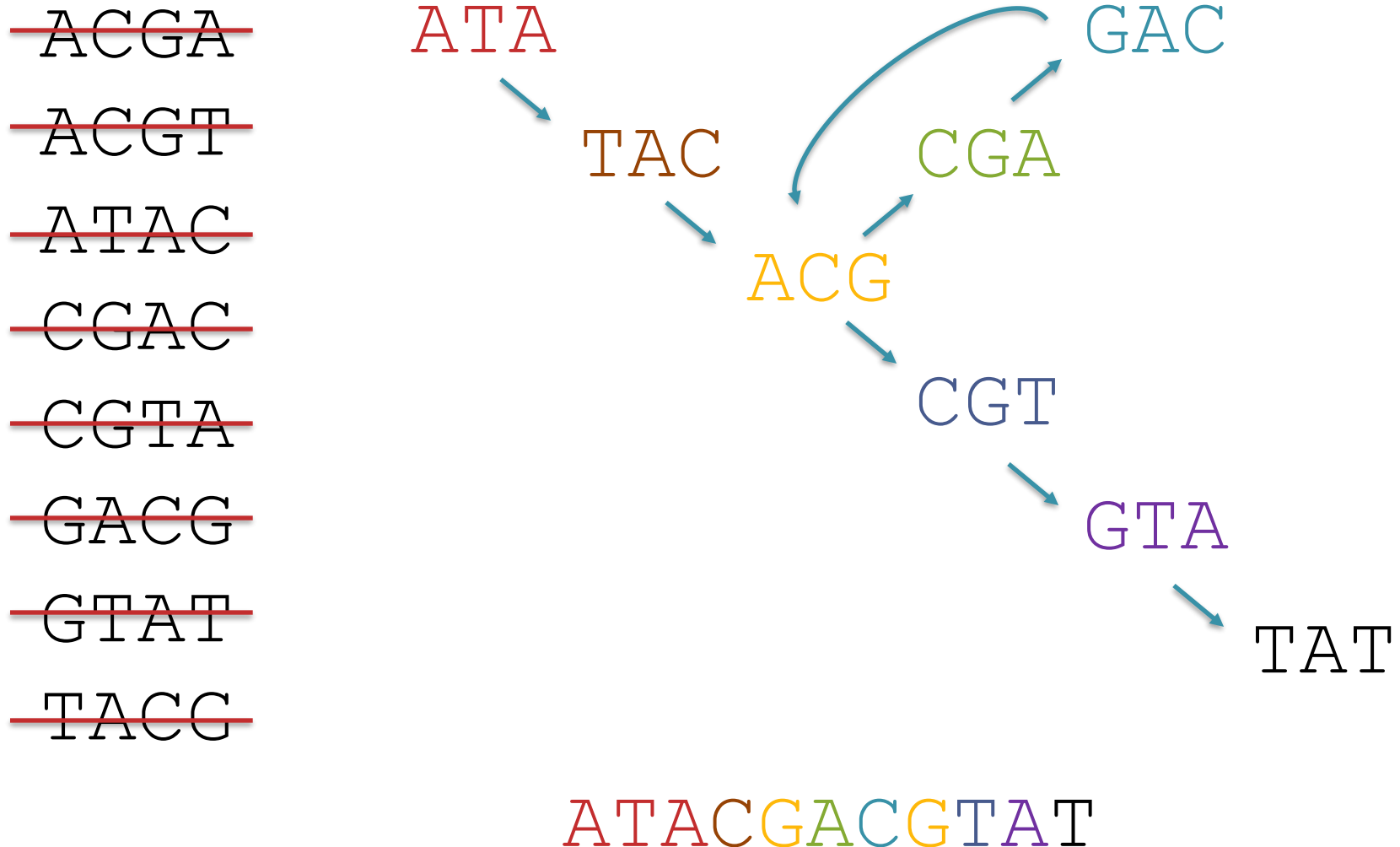
Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):



Pop Quiz 2

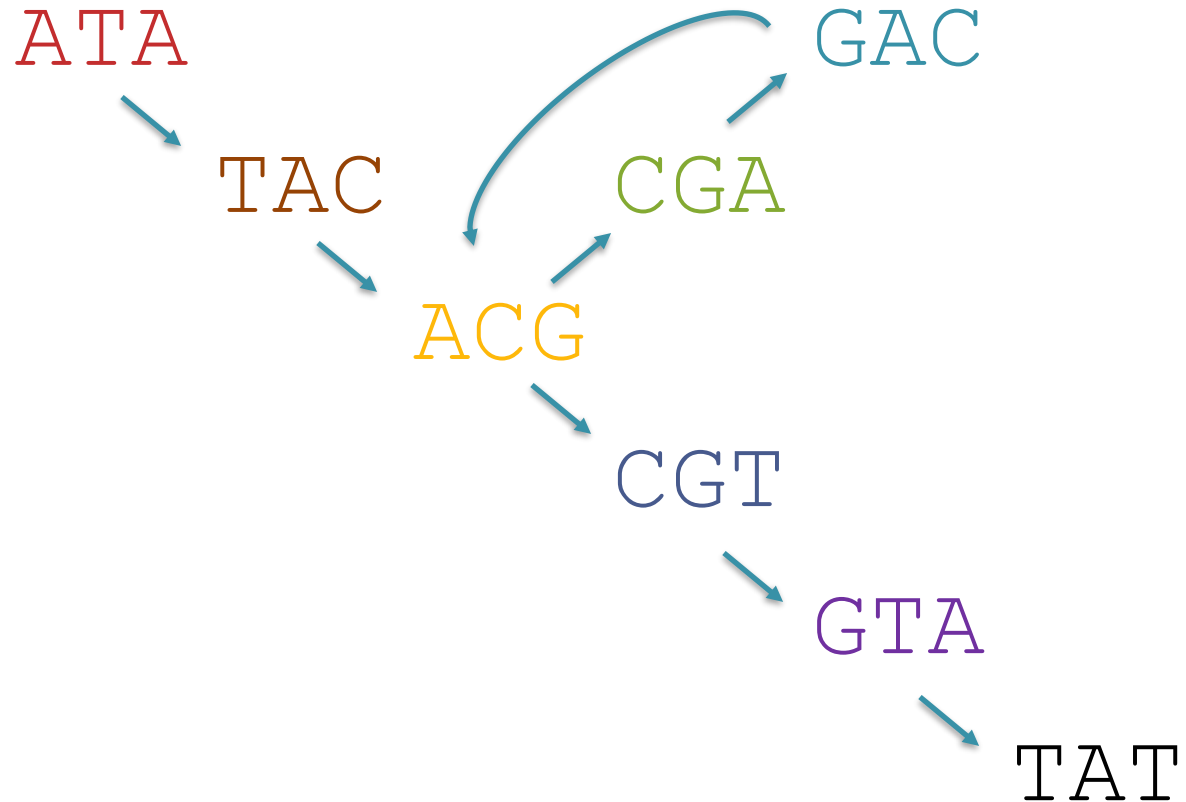
Assemble these reads using a de Bruijn graph approach (k=3):



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~
~~ACGT~~
~~ATAC~~
~~CGAC~~
~~CGTA~~
~~GACG~~
~~GTAT~~
~~TACG~~

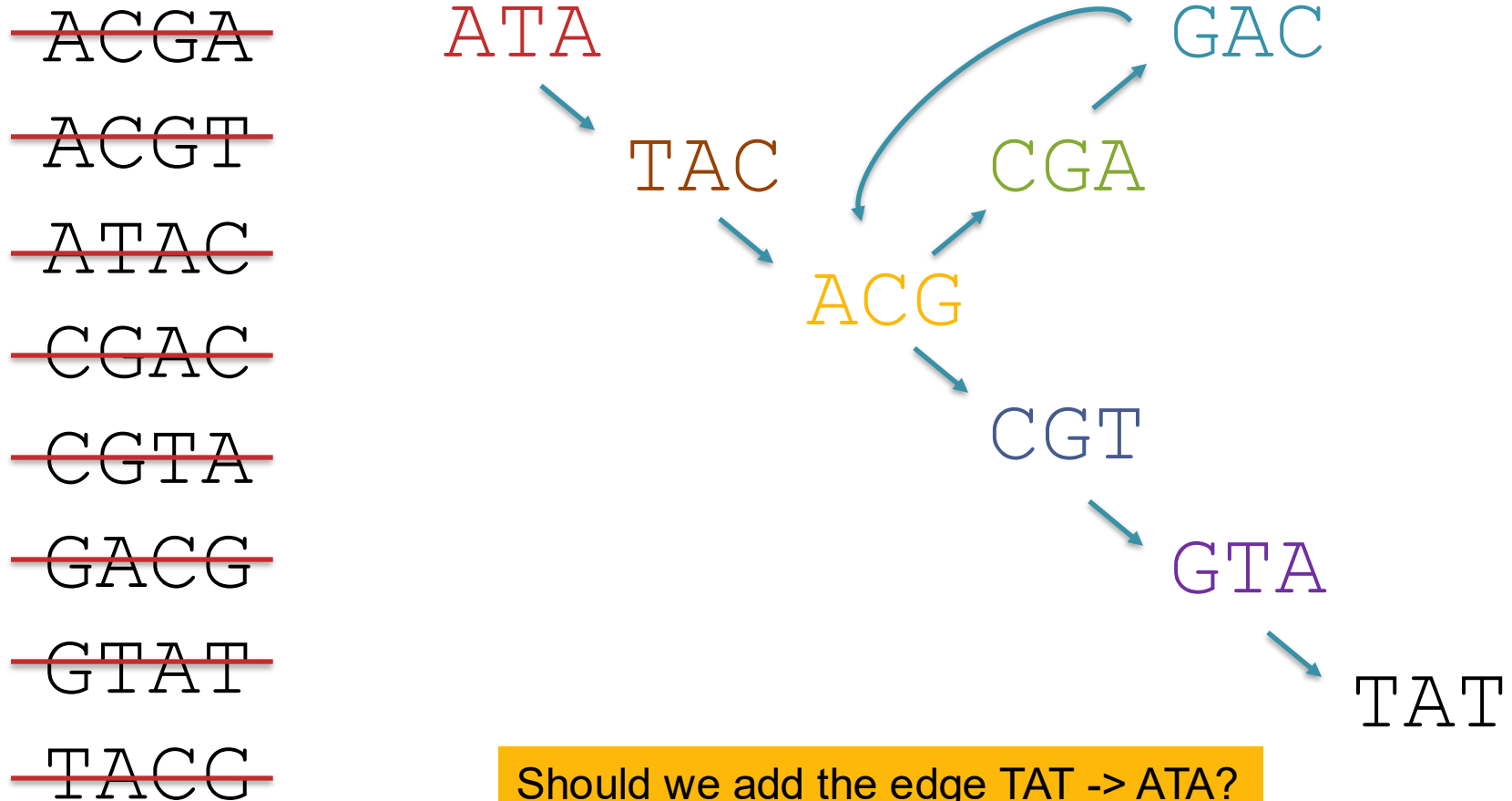


Whats another possible genome?

ATACGACGTAT

Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):



ATACGACGTAT