

# Genome Informatics 2023

Lesson 1 - Introduction

# Communication

**[github.com/vladimirkovacevic/gi-2023-etf](https://github.com/vladimirkovacevic/gi-2023-etf)**

(All info about the course. Create an issue!)

[vladimir.kovacevic@etf.rs](mailto:vladimir.kovacevic@etf.rs)

General info (not relevant to all)

Questions about lessons 1-6

[vojislav.varjacic@sbgenomics.com](mailto:vojislav.varjacic@sbgenomics.com)

Questions about lessons 7-10

[marko.zecevic@sbgenomics.com](mailto:marko.zecevic@sbgenomics.com)

Questions about lessons 11-12

[marko.misic@etf.bg.ac.rs](mailto:marko.misic@etf.bg.ac.rs)

Course professor

[pediao@etf.bg.ac.rs](mailto:pediao@etf.bg.ac.rs)

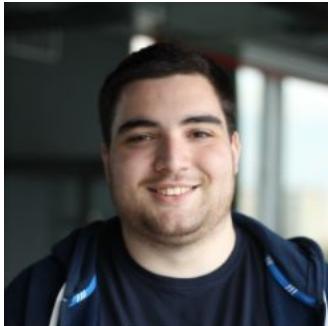
Teaching assistant

Questions about technical details and exercises

# Teachers



Vladimir Kovacevic finished computer science in 2007 at the School of Electrical Engineering where he also obtained his PhD in 2016. He worked for 5 years for Intel as algorithm developer and 5 years he's been with Seven Bridges bioinformatics company and teaching Genome informatics course at School of Electrical Engineering in Belgrade. His current interest includes machine learning applied in genomics, precision medicine, neoantigen discovery and bioinformatics in general.



Vojislav Varjacic has a masters degree from School of Electrical Engineering, department of Physical Electronics. Has been working at Seven Bridges for the last 5 years as a bioinformatics analyst; specializing in tumor heterogeneity and copy number variation analysis. Currently working on automating large scale analyses and machine learning medical imaging projects.



Marko Zecevic has obtained his master's degree from School of Electrical Engineering, department of Signals and Systems. Spent several years designing relay protection and control systems for high voltage substations before transitioning to bioinformatics in 2015. His interests include transcriptomics, populational genomics and R programming. PhD student of genomics at Faculty of Biology.

# Teachers - course support



Marko Mišić is an assistant professor at the University of Belgrade, School of Electrical Engineering, Republic of Serbia. He completed B.Sc. M.Sc. and Ph.D. in the field of Computer Engineering and Informatics, at the same university. His main areas of interest are parallel programming with an emphasis on GPU programming, algorithms, data structures, and complex network analysis.



Predrag Obradović is a senior teaching assistant at the University of Belgrade, School of Electrical Engineering, Republic of Serbia. He completed B.Sc. M.Sc. in the field of Software Engineering, at the same university, where he pursues Ph.D. in Computer Engineering and Informatics. His research interests include social and complex network analysis, bioinformatics, and machine learning.

# Course info

- 13 classes (lecture + exercise) - ~2.5h
- Exam will have both theoretical and practical part
  - 40% on the exam
  - 60% during the semester
    - 40% project assignment (with presentation)
    - 20% - exercises (homework)
- Last class - presentation of student assignments
- Exercise will follow lectures - examples in python [Jupyter notebook](#)

# Course info - syllabus

1	Course info. Bioinformatics and genomics definitions. Molecular biology basics. Genome sequencing technologies. Git.
2	Portable and reproducible bioinformatic analysis. Describing bioinformatic tools in Common Workflow Language. Exercise: Introduction to python and Jupiter environment.
3	Executing bioinformatic analysis locally and on the cloud. Variant calling. Cancer analysis. Exercise: Python structures. Pandas library. Writing tests.
4	RNA-Seq alignment. Normalization procedures; Differential expression: statistical inference, multiple testing corrections; Biological pathways.
5	Single cell RNA analysis. Introduction to spatial transcriptomics analysis.
6	Exact string matching: Boyer-Moore, Indexing structures, Hash tables, Suffix trie/tree
7	Burrows-Wheeler Transform and FM Index
8	Approximate string matching, Edit distance, Dynamic programming, Global alignment
9	Variation on global alignment (end-space-free variant, longest common substring) , local alignment, gaps. Practical: BLAST, Bowtie
10	Shortest common superstring, Overlap graph
11	De-Bruijn graph, scaffolding, error correction
12	Guest lecture
13	Presentation of the student projects.

# Literature

- Vince Buffalo: **Bioinformatics Data Skills**
- Dan Gusfield: **Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology**, Cambridge University Press
- Pavel Pevzner, Neils Jones: **An Introduction to Bioinformatics Algorithms (Computational Molecular Biology)**, MIT Press
- R. Durbin, S. Eddy, A. Krogh, G. Mitchinson: **Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids** , Cambridge University Press
- Veli Mäkinen, Djamal Belazzougui, Fabio Cunial, Alexandru I. Tomescu: **Genome-Scale Algorithm Design: Biological Sequence Analysis in the Era of High-Throughput Sequencing**, Cambridge University press

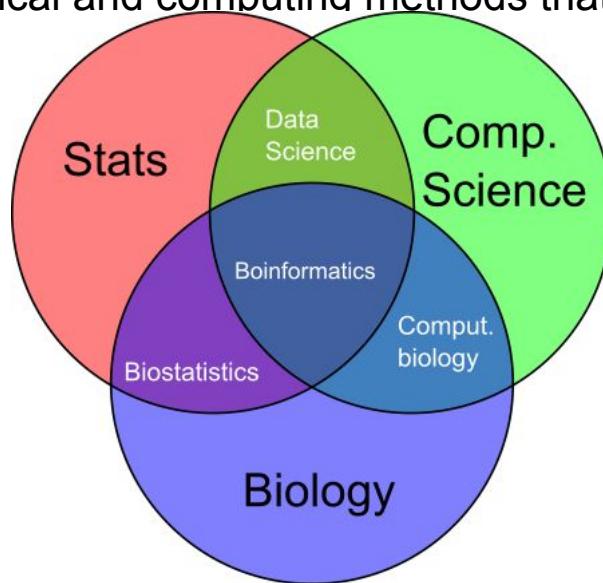
# Lesson overview

- Bioinformatics and genomics definitions
- Molecular biology basics
- Genome sequencing technologies
- We use git

# What is bioinformatics

**Bioinformatics**, n. The science of information and information flow in biological systems, esp. of the use of computational methods in genetics and genomics. (*Oxford English Dictionary*)

**Bioinformatics** - using statistical and computing methods that aim to solve biological problems.



# What is bioinformatics

"I do not think all biological computing is bioinformatics, e.g. mathematical modelling is not bioinformatics, even when connected with biology-related problems. In my opinion, bioinformatics has to do with management and the subsequent use of biological information, particular genetic information."

-- Richard Durbin

**Bioinformatics in practice:** Develops methods and software tools for storing, retrieving, organizing and analyzing biological data.

# Bioinformatics principles

- Golden rule of bioinformatics: **Never ever trust your tools (or data)**
- Adapt robust and reproducible practices
  - Document each step
  - Write down key facts
  - Conclude the work done to easy understand it again in few months (make figures, reports)
  - Automate manual tasks with script
  - Always ask yourself: How much time would it take me to do it one more time?
  - Release or publish your code and tools
- Write code for humans, Write (meta)data for computers
  - Make assertions (be loud)
  - Let the code test the code
  - Use existing libraries whenever possible
  - Frequently used scripts -> tools
  - Use code versioning (git)

# Genomics 101

**Genome:** “The complete set of genes or genetic material present in a cell or organism.” (*Oxford English Dictionary*)

- “Blueprint” or “recipe” of life
- Human genome - 6 billions of base-pairs (A, C, T, G) letters
  - Can be imagined as a string 6 billion letters long

ACTGTGTCACATCGAGAGAGATCACAAACACATAGATTACGATCGAACGTAACGTAACACACC  
AAATATACTGAGTGAGGGGTGGGGACCCCCCCCCCCCCACACACATTACAAACCTAGAT  
CACCATACTGAGATATAAGAGAGAGANACGTACGTACACAATTACAAATTACAACACAAAGTA  
CTTATACATACACATGGGACCCATAGCACACACAGATATTATAATATAGAGAGACAATGT  
CGTGCAGTAA...

# Genomics: contrast with biology & genetics\*

\* Everything on this slide is  
a gross generalization

## Biology & Genetics

Targeted studies of one or a few genes

Targeted, low-throughput experiments

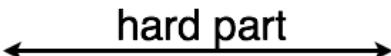
Clever experimental design, painstaking experimentation

## Genomics

Studies considering all genes in a genome

Global, high-throughput experiments

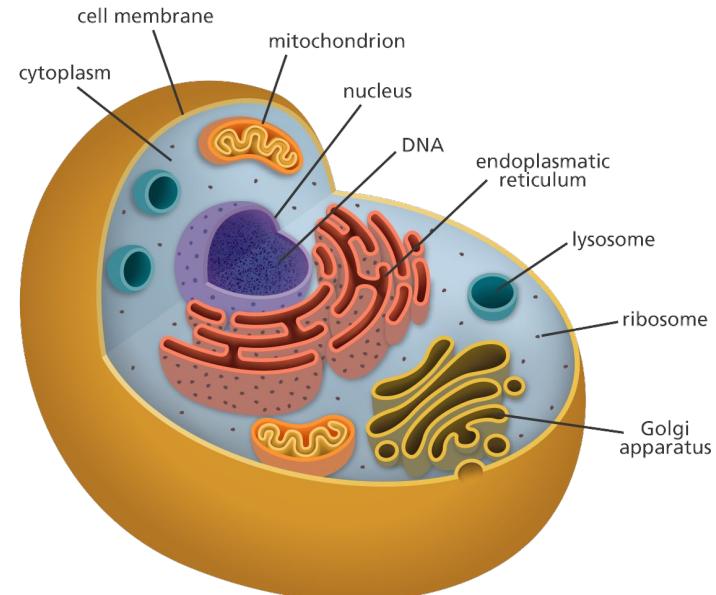
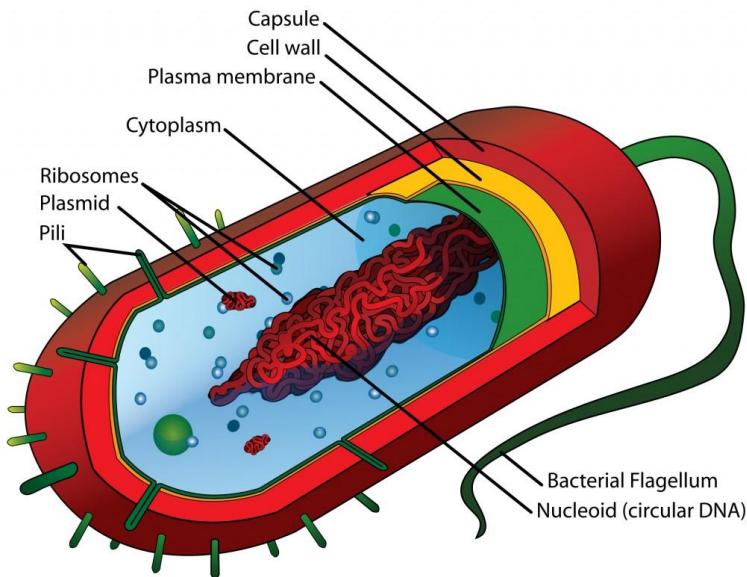
Tons of data, uncertainty, computation



# The cell

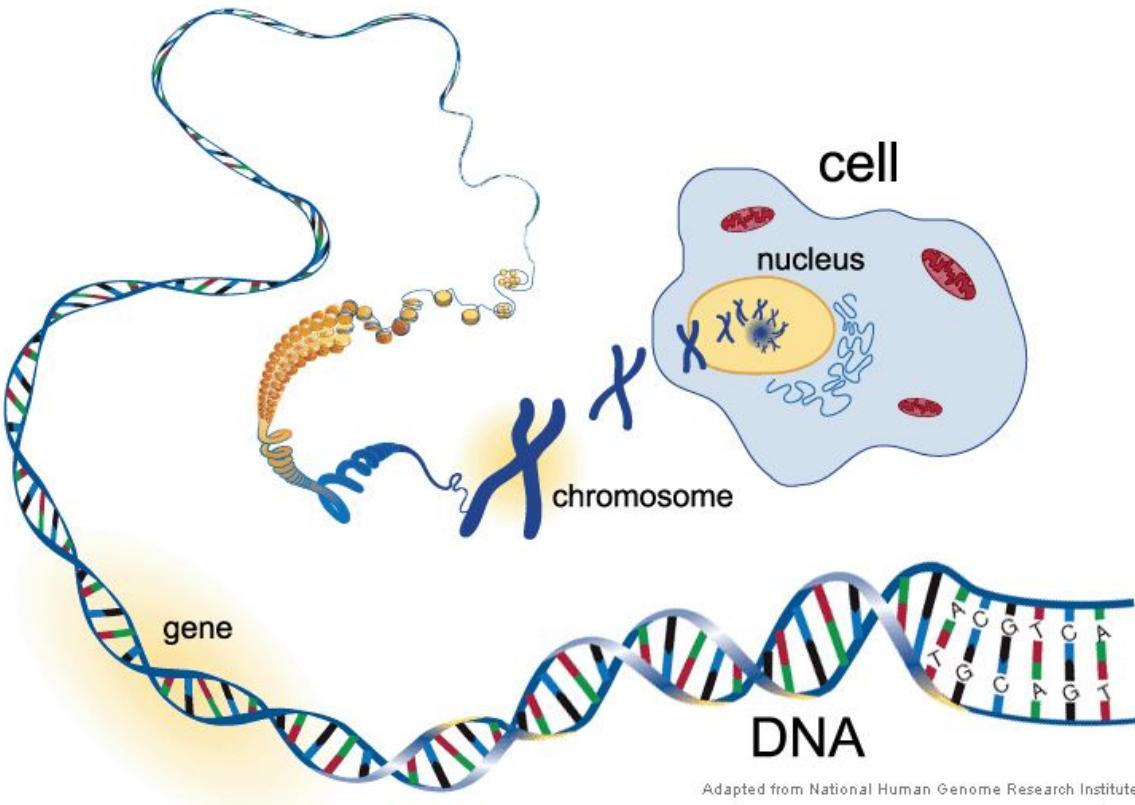
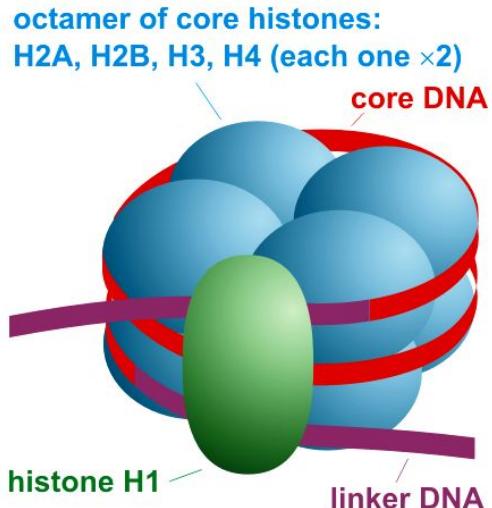
Fundamental working units of every living system.

- Prokaryotic (bacteria)
- Eukaryotic (higher organisms - animals, plants)



# “And inside the nucleus thou lays the mighty DNA”

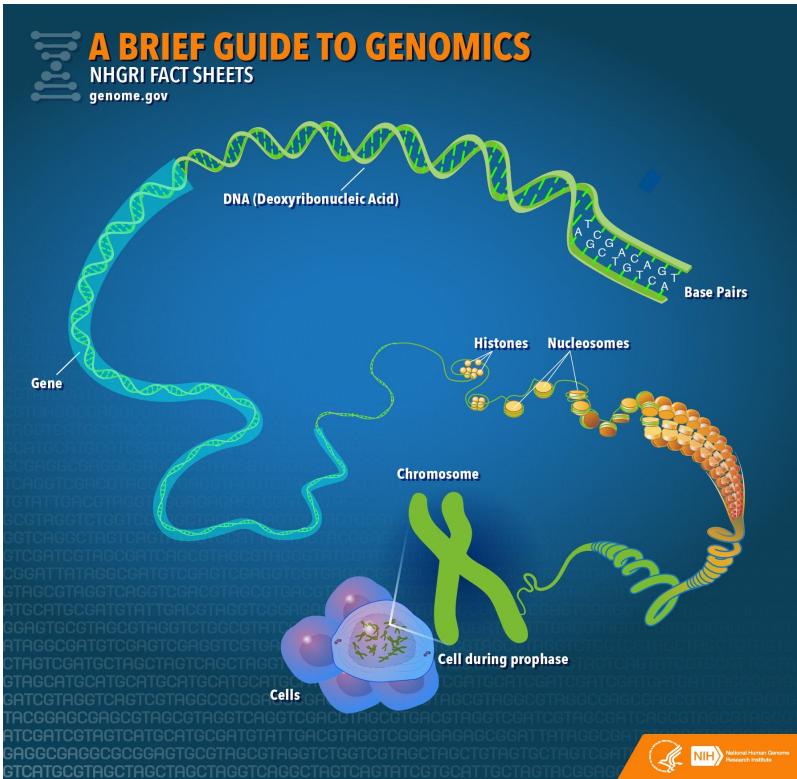
- Chromatin - tightly packed DNA
- A **nucleosome** is a basic unit of DNA packaging in eukaryotes, consisting of a segment of DNA wound in sequence around eight histone protein cores.
- Current model



Adapted from National Human Genome Research Institute

# DNA - the code of life

- DNA (deoxyribonucleic acid) - double stranded molecule
- Same in every cell - DNA replication during cell division
- More stable, redundant information - complementary double helix chain
- ~99.6% same between 2 individuals
- Base (nucleotide) pairs (complementary bases)
  - A - T (adenine and thymine)
  - C - G (cytosine and guanine)

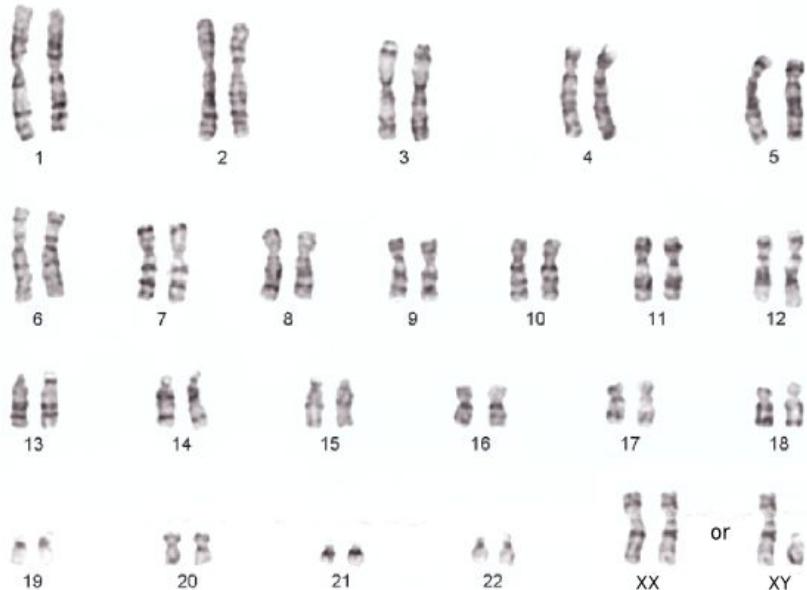


# Genome

- Set of all pairs of chromosomes

- Human genome:

- 23 pair of chromosomes (diploid)
- 22 autosomes
- 1 sex chromosome (X and/or Y)
- 3 billion base-pairs x 2
- CTGGATTATATCGAAGGGACTAT... etc
- Intron and exon (2%)

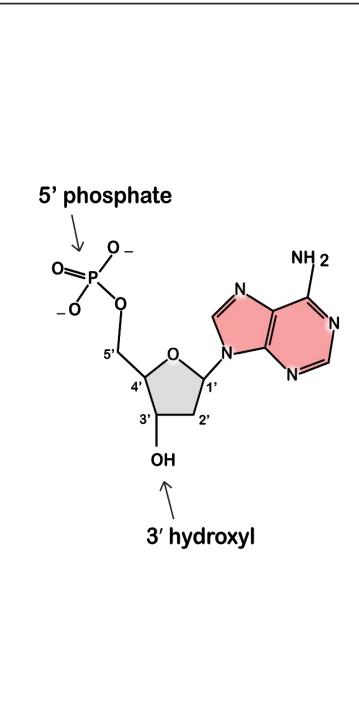
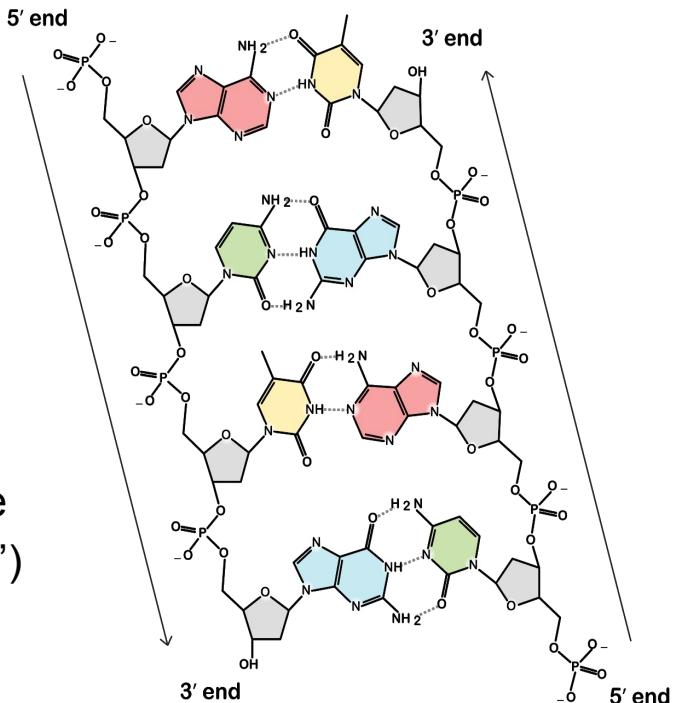


Karyogram

# DNA - structure

- Consists of:
  - Phosphate group
  - Sugar (deoxyribose)
  - Nitrogen base
- Hydrogen bonds
- Forward and reverse strand
- DNA direction:
  - 5' head and 3' tail
  - Transcribed from 5' to 3' end
- In bioinformatics we write just one strand (by convention from 5' to 3')

5' ACTG 3'  
↓  
3' TGAC 5'  
(reverse complement)

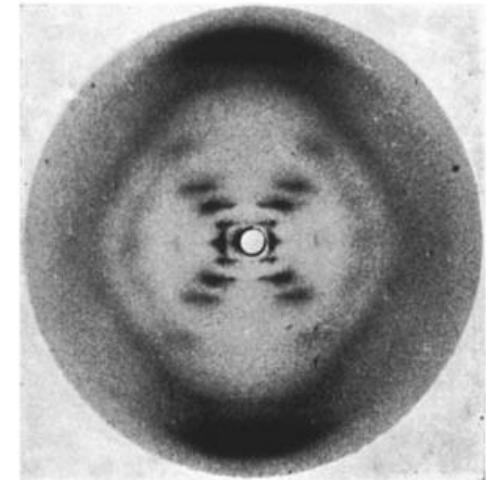


# DNA - discovery

**1952-1953** James D. Watson and Francis H. C. Crick deduced the double helical structure of DNA from X-ray diffraction images by Rosalind Franklin (provided by M. Wilkins) and data on amounts of nucleotides in DNA.

"Molecular structure of nucleic acids. A structure for deoxyribose nucleic acid"

scientist: "does everyone here know what Watson and Crick discovered?"  
me from back of room: "Rosalind Franklin's notes"



*Photo 51*

# Central dogma of molecular biology



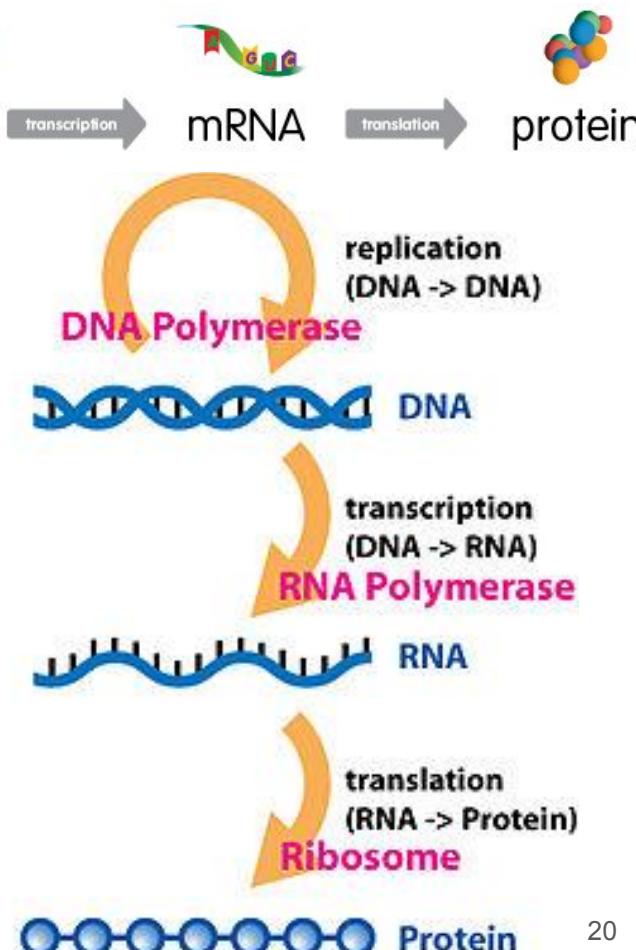
DNA ----> RNA -----> Protein

Transcription: DNA ->RNA

- particular segment of DNA is copied into RNA (especially mRNA) by the enzyme RNA polymerase.

Translation: RNA -> Protein

- process in which ribosomes synthesize proteins after the process transcription of DNA to RNA in the cell's nucleus.



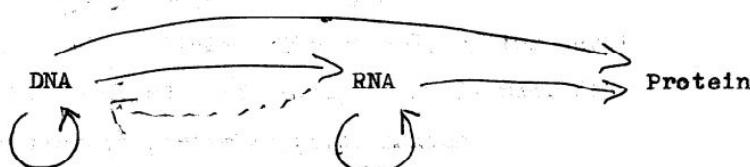
# Central dogma of molecular biology

## Ideas on Protein Synthesis (Oct. 1956)

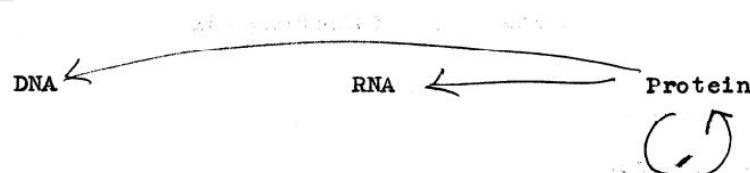
### The Doctrine of the Triad.

The Central Dogma: "Once information has got into a protein it can't get out again". Information here means the sequence of the amino acid residues, or other sequences related to it.

That is, we may be able to have



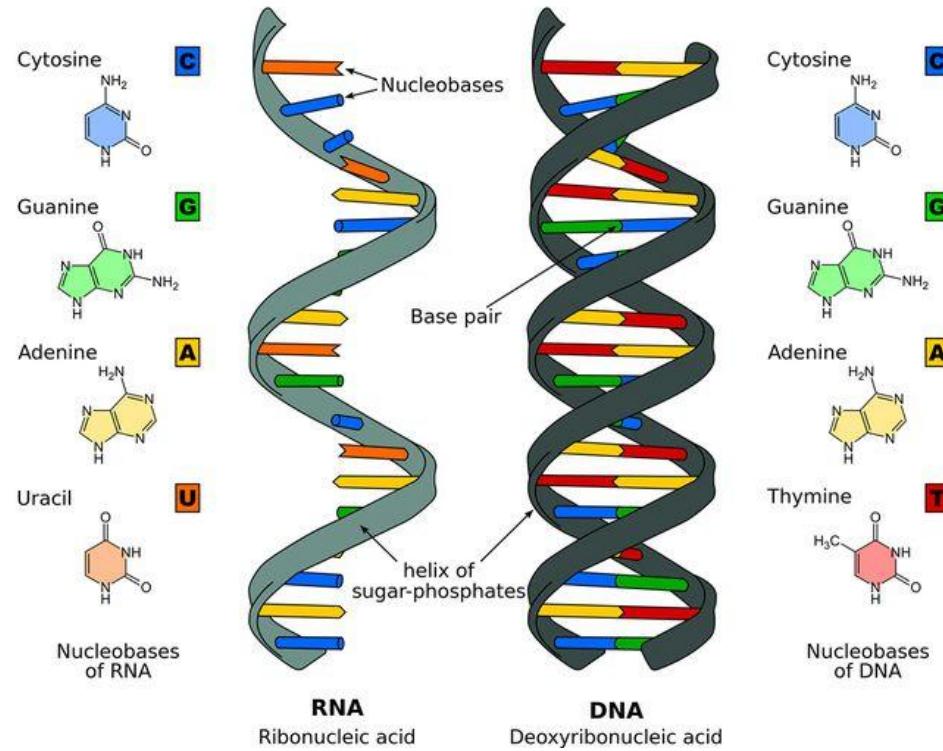
but never



where the arrows show the transfer of information.

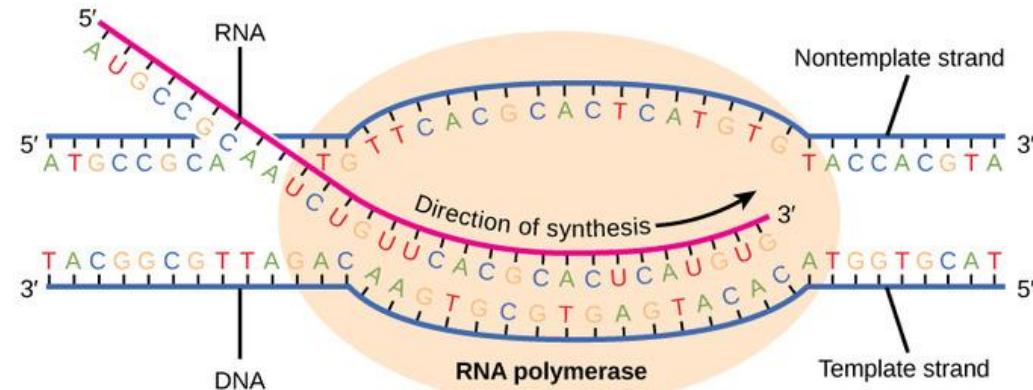
# RNA

- Single stranded
- Sugar:
  - ribose (instead of deoxyribose)
- Uracil instead of Thymine

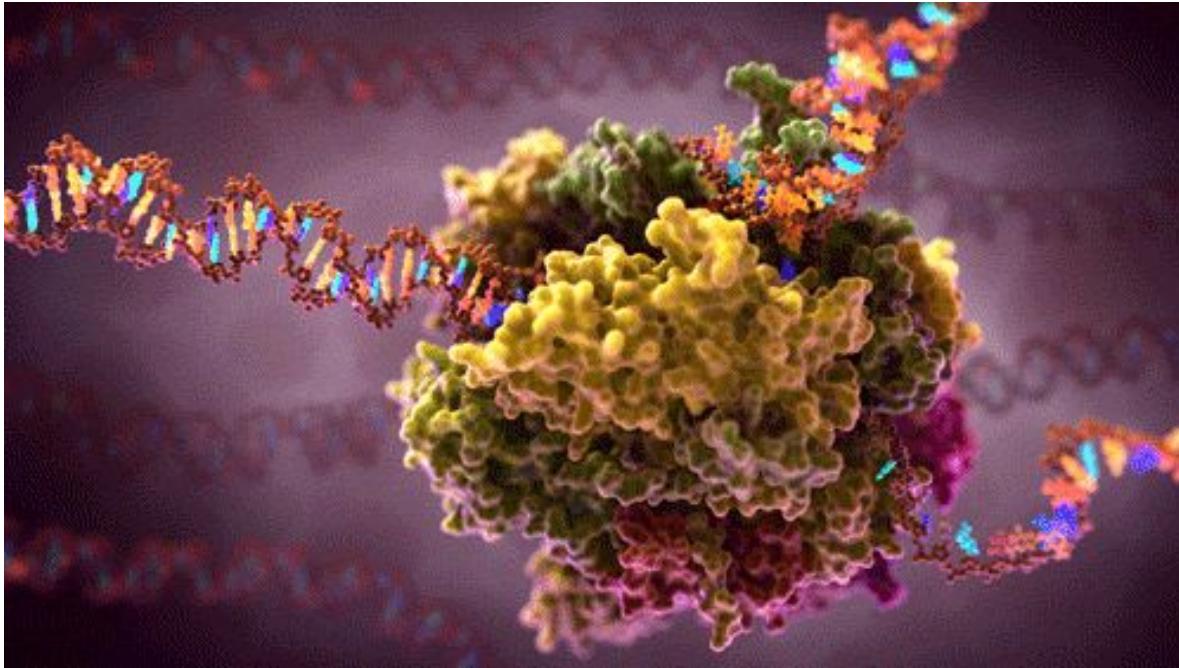


# Transcription

- Template (noncoding) strand
  - One which is transcribed by RNAP (RNA polymerase)
- Pre-mRNA is synthesized from a DNA template in the cell nucleus

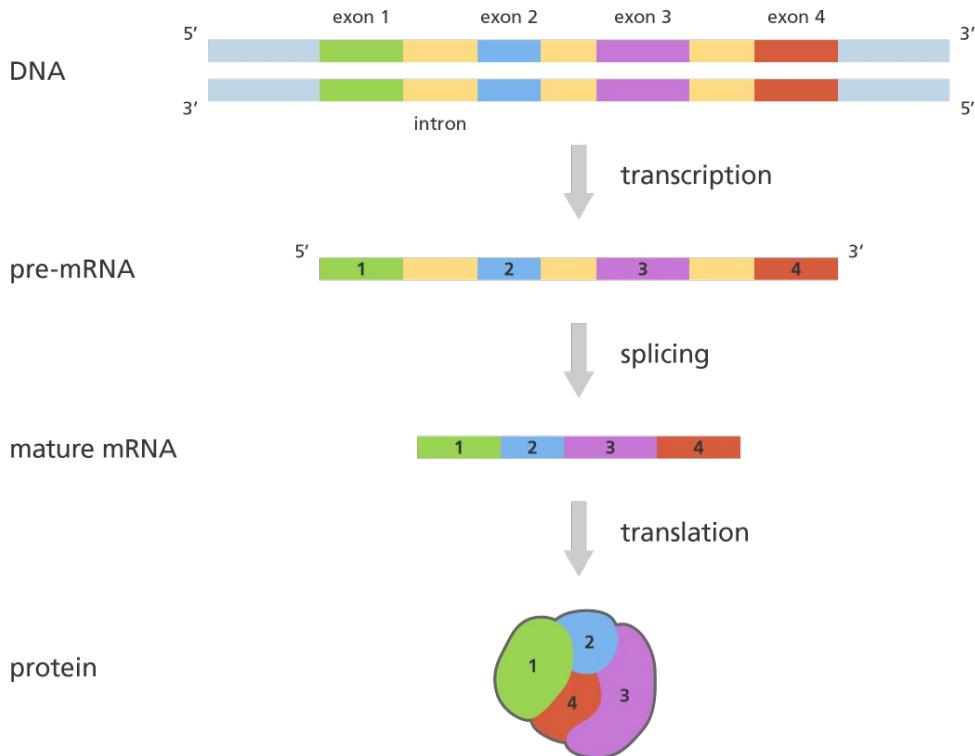


# Transcription



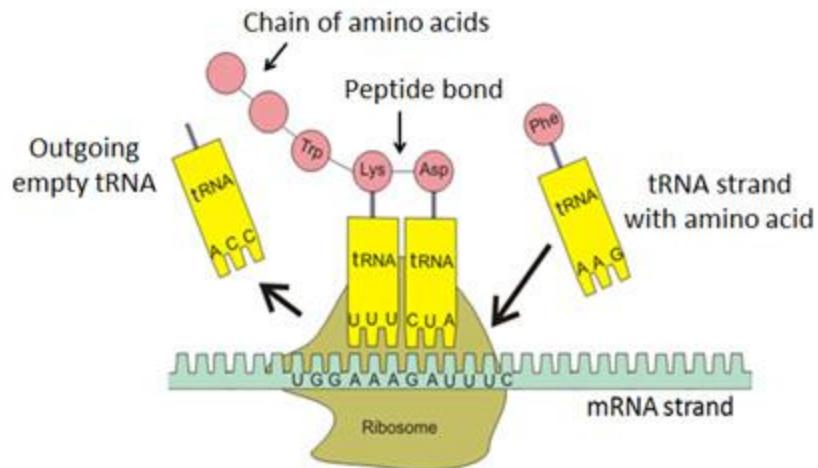
# Transcription and splicing

- Splicing: removing all the introns (non-coding regions of RNA) and splicing back together exons (coding regions)
- Alternative splicing
- Transcript (form of the gene, coding sequences for protein synthesis)



# Translation

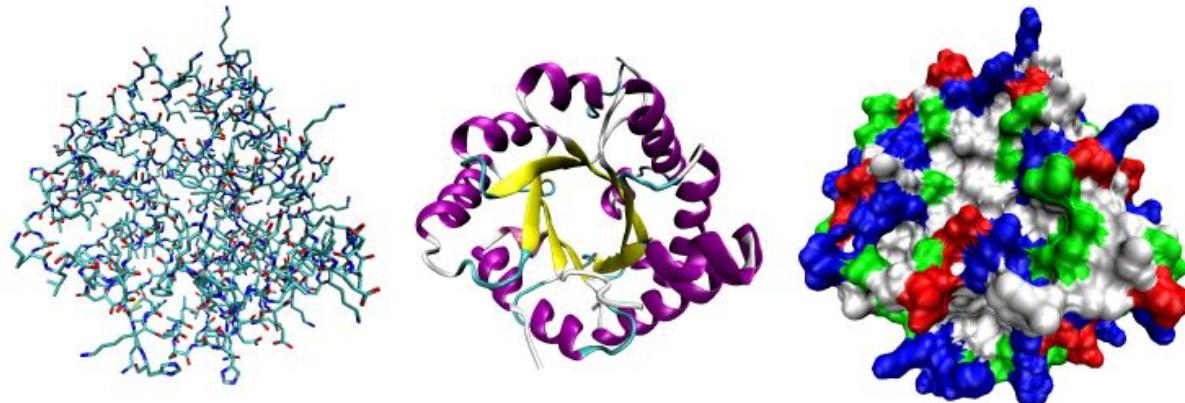
- Occurs in ribosome
- Each triplet of nucleotides (codon) codes for specific amino-acid
  - “Letters of protein code”
  - 20 amino-acid (some redundancy)



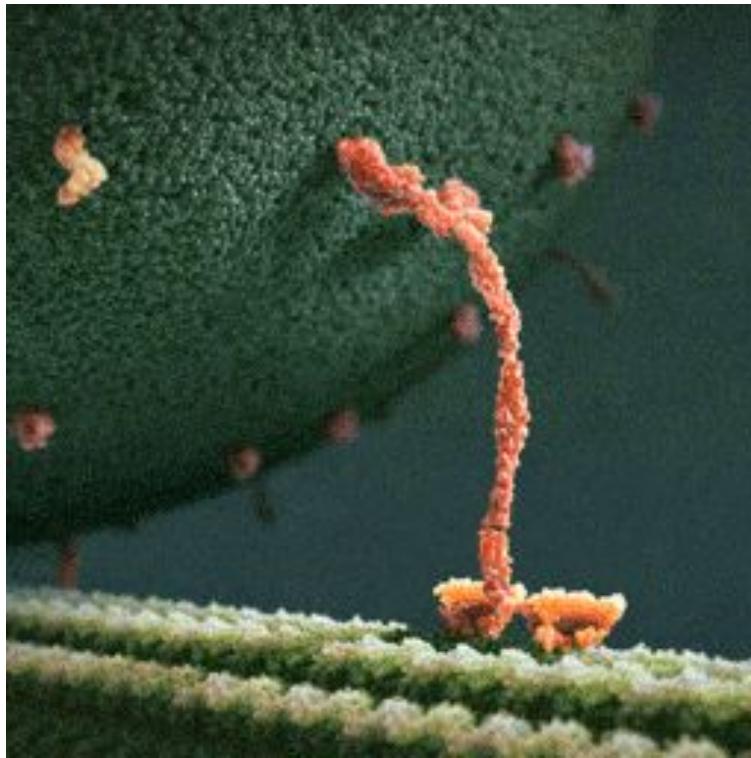
		Second letter					
		U	C	A	G		
First letter	U	UUU } Phe UUC } UUA } Leu UUG }	UCU } UCC } UCA } UCG }	UAU } Tyr UAC } UAA Stop UAG Stop	UGU } Cys UGC } UGA Stop UGG Trp	UCA } UAG }	U C A G
	C	CUU } CUC } CUA } Leu CUG }	CCU } CCC } CCA } CCG }	CAU } His CAC } CAA } Gln CAG }	CGU } CGC } CGA } Arg CGG }	UCA } UAG }	U C A G
A	A	AUU } AUC } Ile AUA }	ACU } ACC } ACA } ACG }	AAU } Asn AAC } AAA } Lys AAG }	AGU } Ser AGC } AGA } Arg AGG }	UCA } UAG }	U C A G
	G	GUU } GUC } GUA } Val GUG }	GCU } GCC } GCA } GCG }	GAU } Asp GAC } GAA } Glu GAG }	GGU } GGC } GGA } Gly GGG }	UCA } UAG }	U C A G

# Proteins

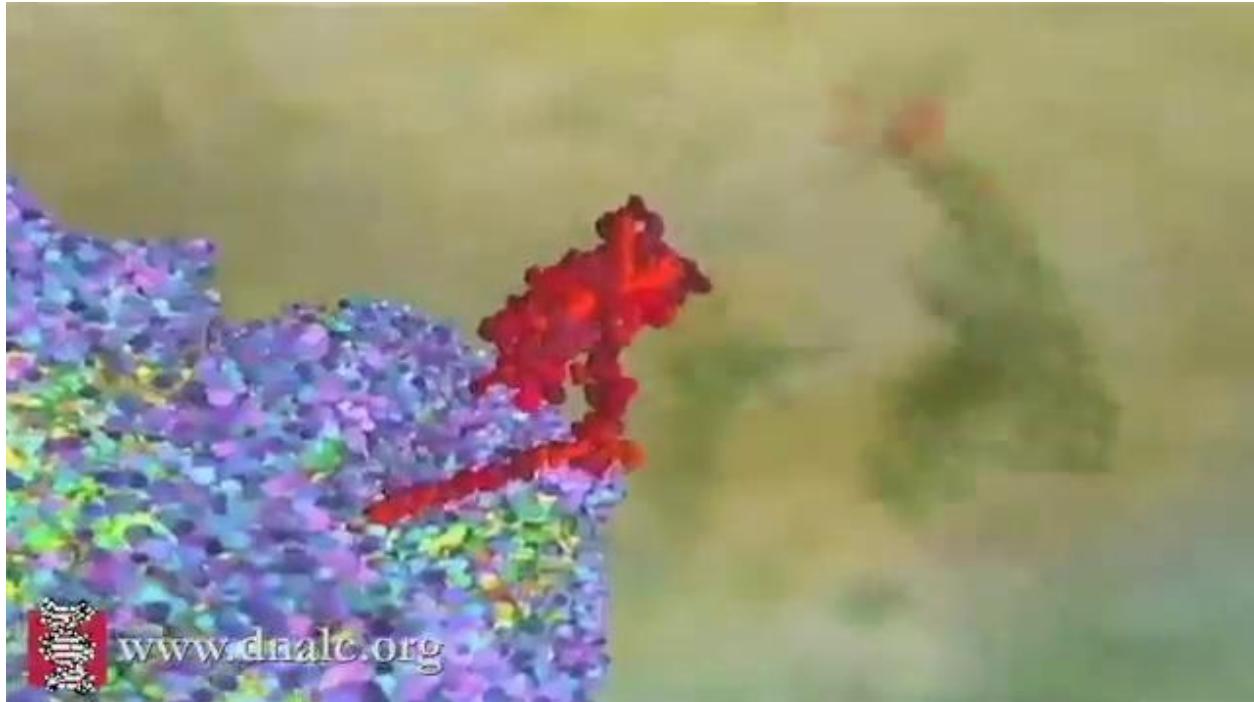
- Building blocks of life
  - Various functions in the organism (transportation, regulation, metabolism, DNA replication)
- Long chains of amino-acids, that also fold into complicated 3D structures
  - We often distinguish protein primary, secondary, tertiary and quaternary structure



# Proteins



# Proteins



[www.dnalc.org](http://www.dnalc.org)

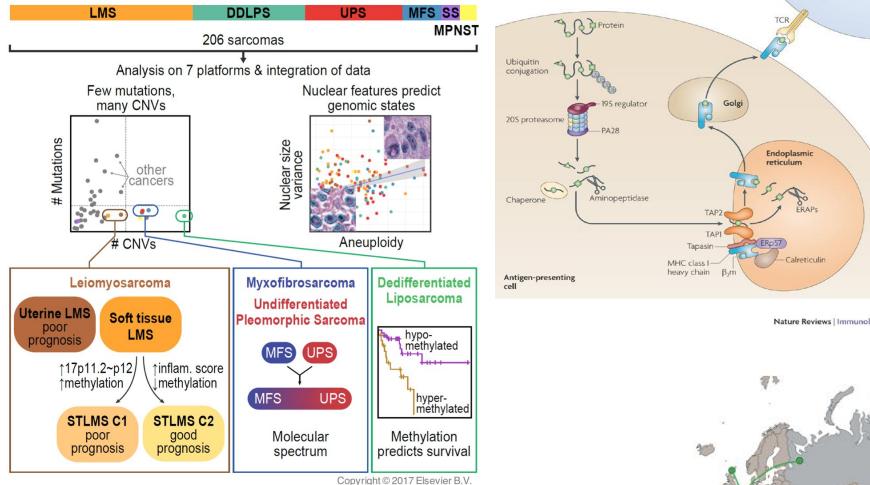
# Biology 101 - genotype vs phenotype

The **genotype** is the part of the genetic makeup of a cell, and therefore of an organism or individual, which determines one of its characteristics (phenotype).

A **phenotype** (from Greek *phainein* , meaning 'to show ', and *typos* , meaning 'type') is the composite of an organism's **observable characteristics** or traits, such as its morphology, development, biochemical or physiological properties, behavior, and products of behavior (such as a bird's nest).

# Why perform DNA sequencing?

- Rare genetic diseases
  - Origins of humans
  - Precision medicine- Cancer treatment (immunotherapy)
  - Microbes that live inside us (microbiome)
  - Study ways that genomes work
  - Gene editing
  - Forensics

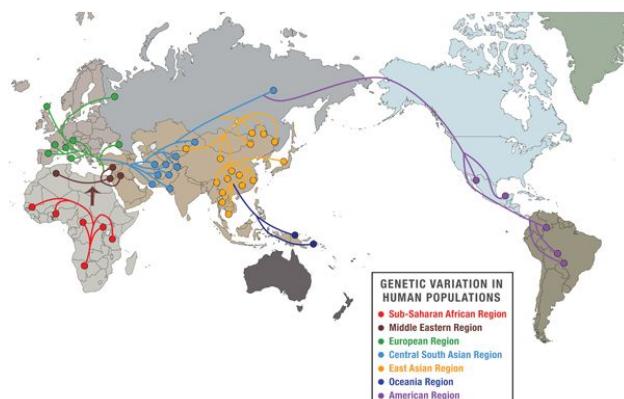
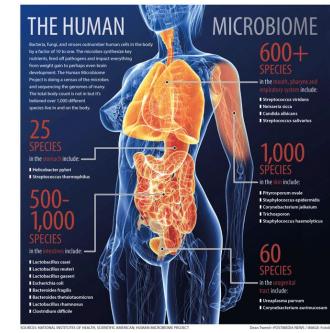


MEDICAL DISPATCH JULY 21, 2014 ISSUE

# ONE OF A KIND

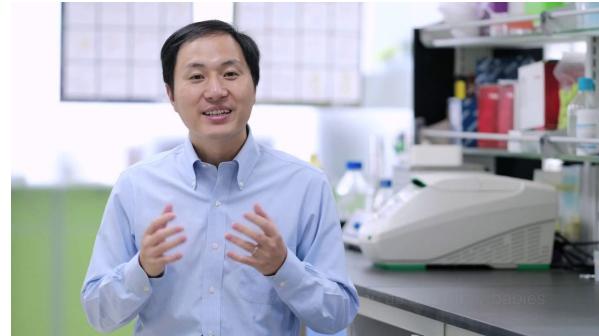
*What do you do if your child has a condition that is new to science?*

By Seth Mnookin



# The world's first germline genetically edited babies

- Clinical project: standard in vitro fertilization
- + CRISPR-Cas9 (technology that can modify DNA)
- Mother was HIV positive
- Modify the CCR5 gene on single egg cell before fertilization to confer genetic resistance to the HIV virus
- CCR5 codes for a protein that HIV uses to enter cells
- Clinical project was conducted secretly until November 2018
- Lulu and Nana are born healthy crying babies



He Jiankui

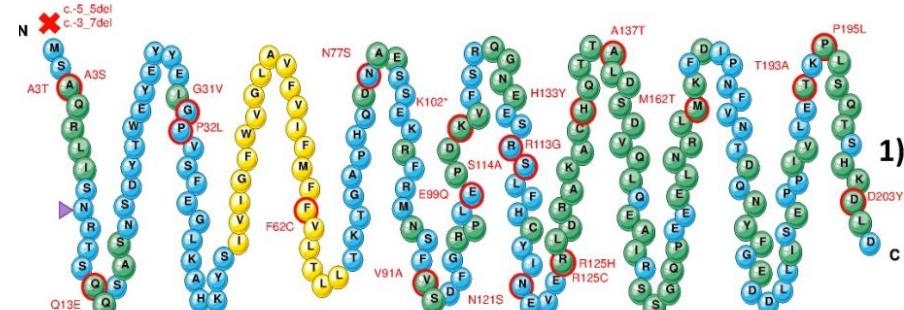
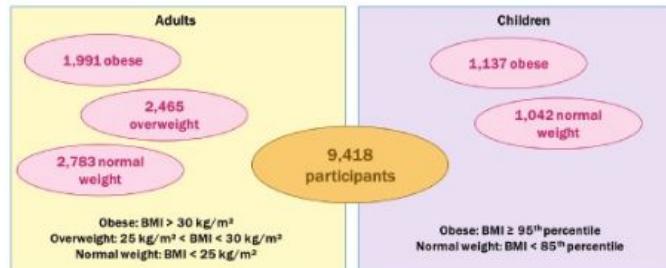
# Exome sequencing of 25000 schizophrenia cases and 100 000 controls

Tarjinder Singh

- The Schizophrenia Exome Sequencing Meta-analysis (SCHEMA) consortium (2017) - aggregating and analyzing high-throughput sequencing data
- Understand the [genetic causes of schizophrenia](#)
- Motivate the development of new therapeutics
- Life expectancy - 12-15 years shorter life expectancy
- **10 genes** that when disrupted, dramatically increase risk for schizophrenia
- Odds ratios 4 - 50,  $P < 2e-6$
- 2 genes code glutamate receptors - crucial in brain cells communication
- 10 genes have no protein-truncating variant signal

# Pathogenic loss-of-function in MRAP2 cause metabolic syndrome - A. Bonnefond

- Melanocortin receptor accessory protein 2 (MRAP2) is a transmembrane accessory protein predominantly expressed in the brain
- Deletion of Mrap2 results obesity in both mice and human
- 23 rare mutations in MRAP2
  - 2 frameshift
  - 1 non-sense
  - 20 missense



# Insight into genetic architecture of autism

Adam Rocke

- Autism sequencing consortium (SPARK)
- 18,381 autism spectrum disorder (ASD) cases and 27,969 controls
- ASD affects 1–1.5% of individuals and is highly heritable
- identifies 5 risk loci

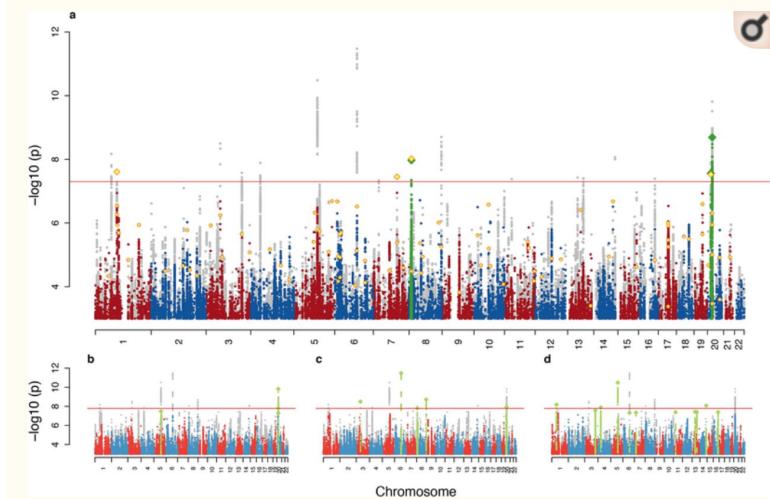


Figure 1.  
Manhattans plots:

with the x axis showing genomic position (chromosomes 1–22) and the y axis showing statistical significance as  $-\log_{10}(P)$  of z statistics. a: The main ASD scan (18,381 cases and 27,969 controls) with the results of the combined analysis with the follow-up sample (2,119 cases and 142,379 controls) in yellow in the foreground. Genome-wide significant clumps are painted green with index SNPs as diamonds. b-d: Manhattan plots for three MTAG scans of ASD together with, respectively, schizophrenia<sup>14</sup> (34,129 cases and 45,512 controls), educational attainment<sup>14</sup> (N = 328,917) and major depression<sup>14</sup> (111,902 case and 312,113 controls). See Supplementary Figures 45–48 for full size plots. In all panels the results of the composite of the five analyses (consisting for each marker of the minimal p-value of the five) is shown in grey in the background.

# Golden state killer

**SEEKING  
INFORMATION**

East Area Rapist/Golden State Killer  
California  
1976 to 1986

**UNKNOWN SUSPECT**



# Golden state killer

- <https://www.gedmatch.com>, tutorial
- applications for comparing your DNA test results with other people
- Genealogical Data Communication
- Software developed by the Church of the Latter Day Saints 
- [Site found 10-20 distant relatives](#) of the killer, roughly, equivalent of third cousins
- “When you go that far back in time, you have trees that grow huge,” Holes said.
- Census data, old newspaper and a gravesite locator relatives, websites such as LexisNexis.

## DNA Applications:

- One-To-Many Beta - give it a try
- One-To-Many DNA Comparison Result
- One-to-One Autosomal DNA Comparison
- One-to-One X-DNA Comparison
- Admixture (heritage)
- Admixture / Oracle Population Search
- People who match me or 1 of 2 kits
- DNA File Diagnosis Analyze DNA file uploaded
- Are you related?
- 3-D Chromosome
- Archaic DNA



# Golden state killer



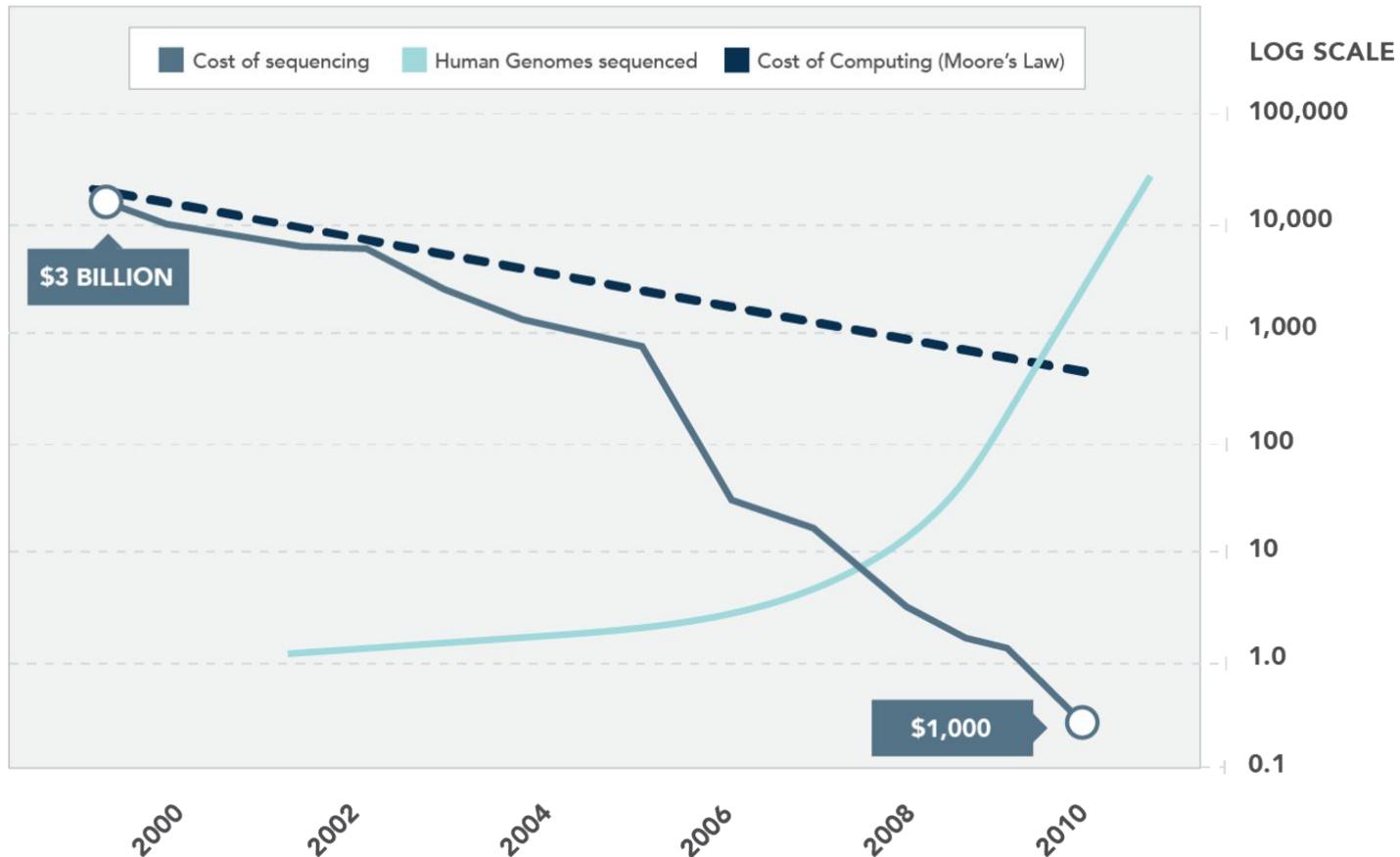
Joseph James DeAngelo

# Genome sequencing

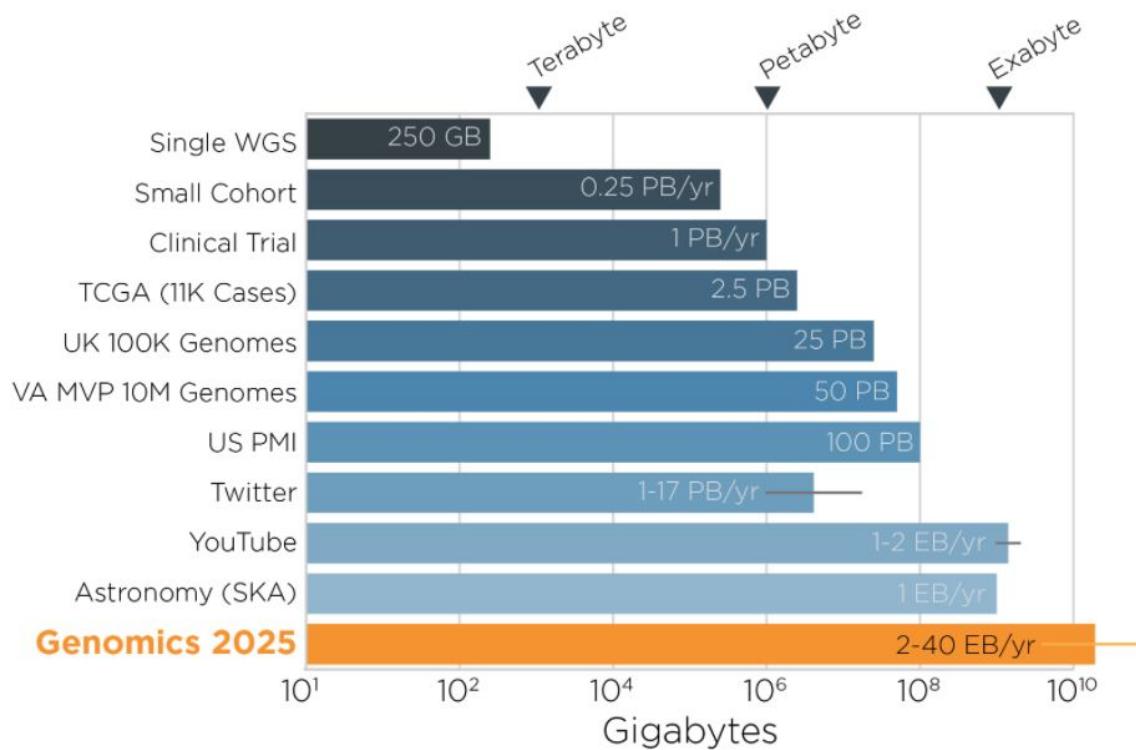
- Digitalization of genome
- **Human Genome Project** (1990-2003), 3B \$
- Birth of bioinformatics
- Sanger sequencing (First generation sequencing)
  - Long (took 13 years)
  - Costly (3B\$ for one human genome)
- Currently NGS (next generation sequencing)
  - Illumina
  - Around 200\$ and 1 day needed to sequence the genome
- Also third generation sequencing in use
  - Longer read-length (up to 50k base)
  - Oxford nanopore, PacBio
  - Higher error rate
  - Smaller in size
  - Sequencing in space



## GROWTH OF DNA SEQUENCING



# Genomics is Big Data



Source: "Big Data: Astronomical or Genomical?" *PLoS Biology* (2015).

## Sequences:

1 zetta-bases/yr

## Storage needs:

2-40 exabytes

## Compute for Alignment:

**10,000 trillion** CPU hrs  
= 83x time since Big Bang

## Variant Calling:

**~2 trillion** CPU hrs

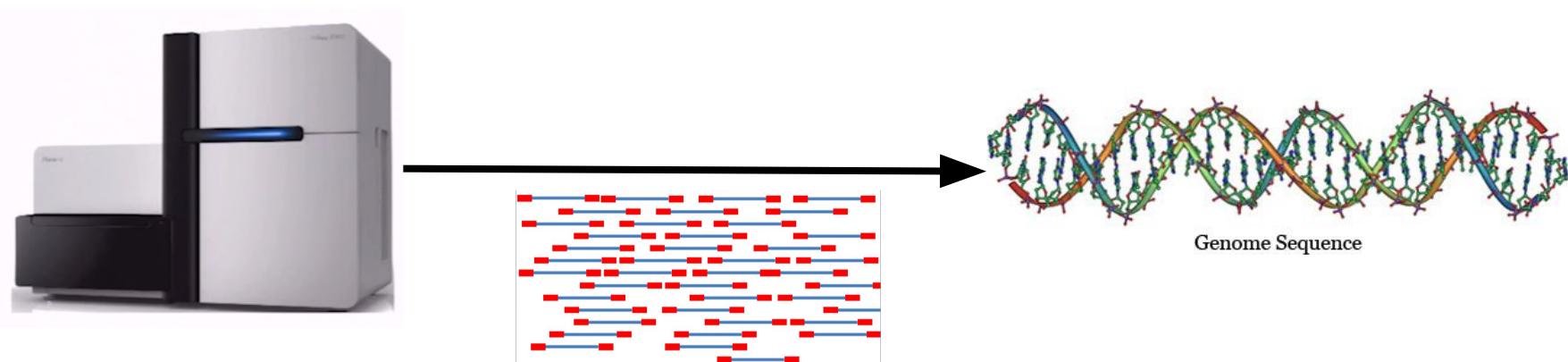
## Tertiary Analysis:

**~4 trillion** CPU hrs

= time since land-breathing mammals evolved

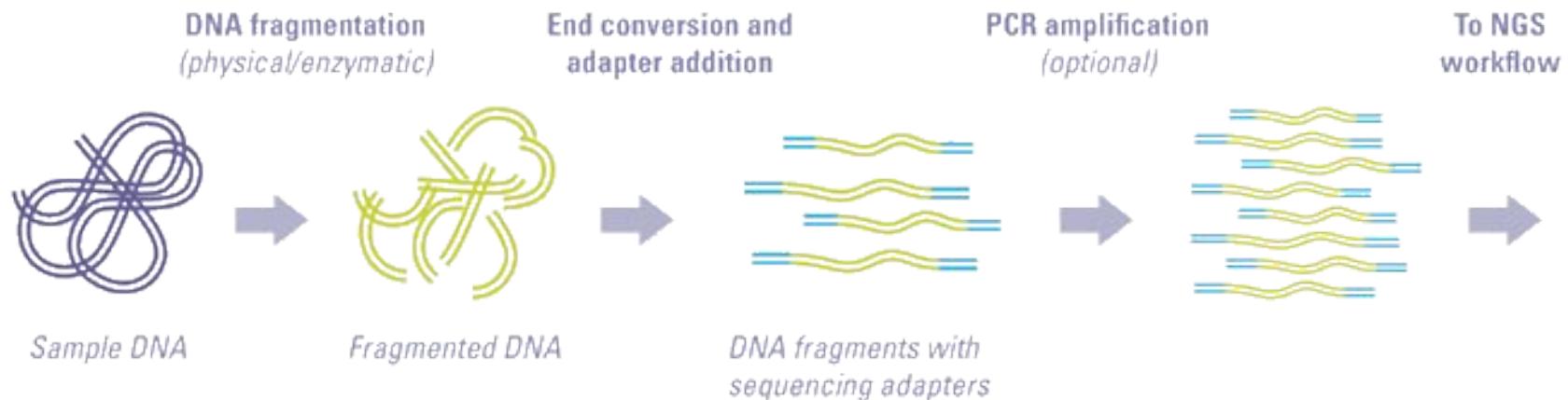
# Bioinformatics to the rescue!

- Genomes of all species are arrays of nucleotides (A, T, C, G) - strings
- The process of DNA sequencing returns only fragments of it
- Our mission: RECONSTRUCT IT!

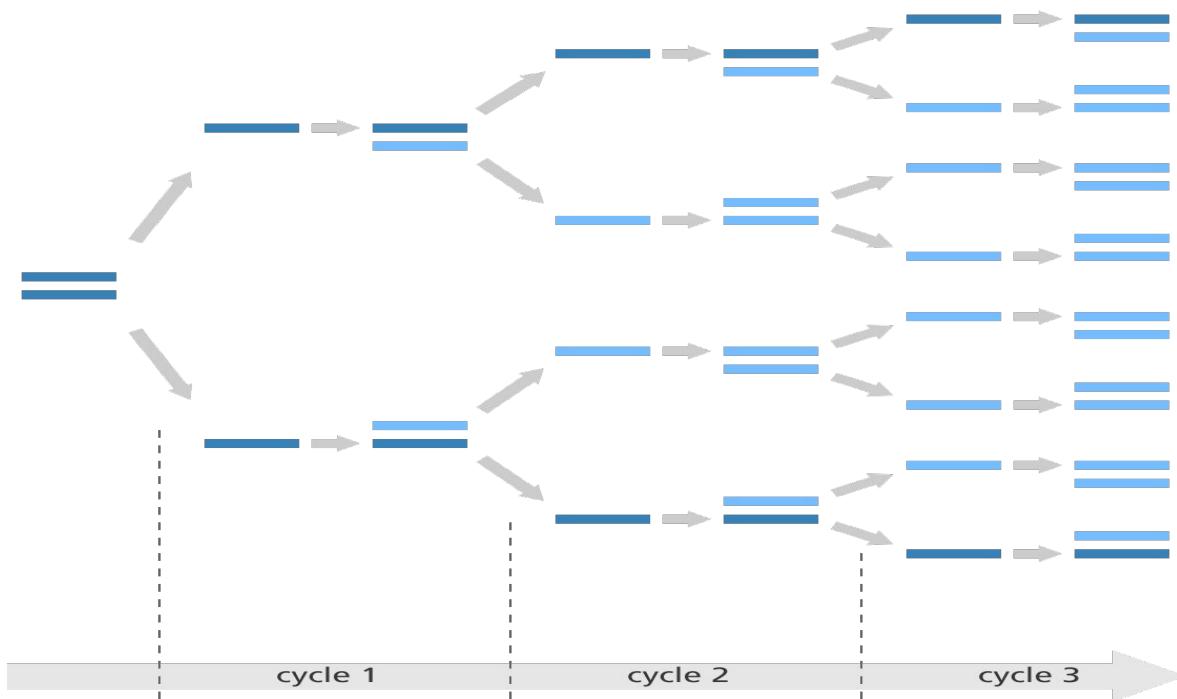


# Illumina sequencing

- Read - DNA fragment after reading it in sequencer
- Typical whole genome sequencing experiment:
  - 200-500 million reads
  - 150-250 bases (letters long)

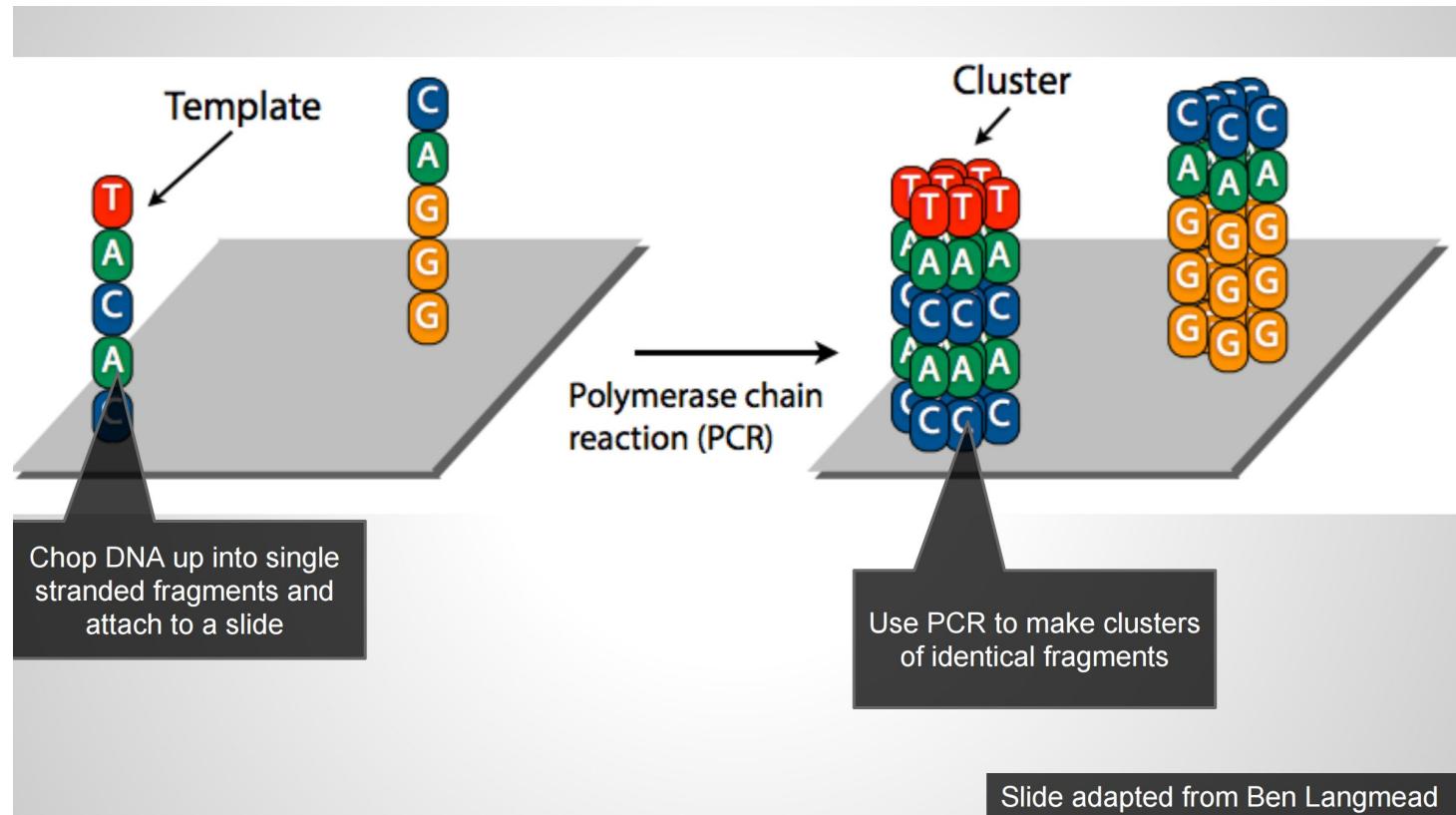


# Sequencing - PCR (polymerase chain reaction)

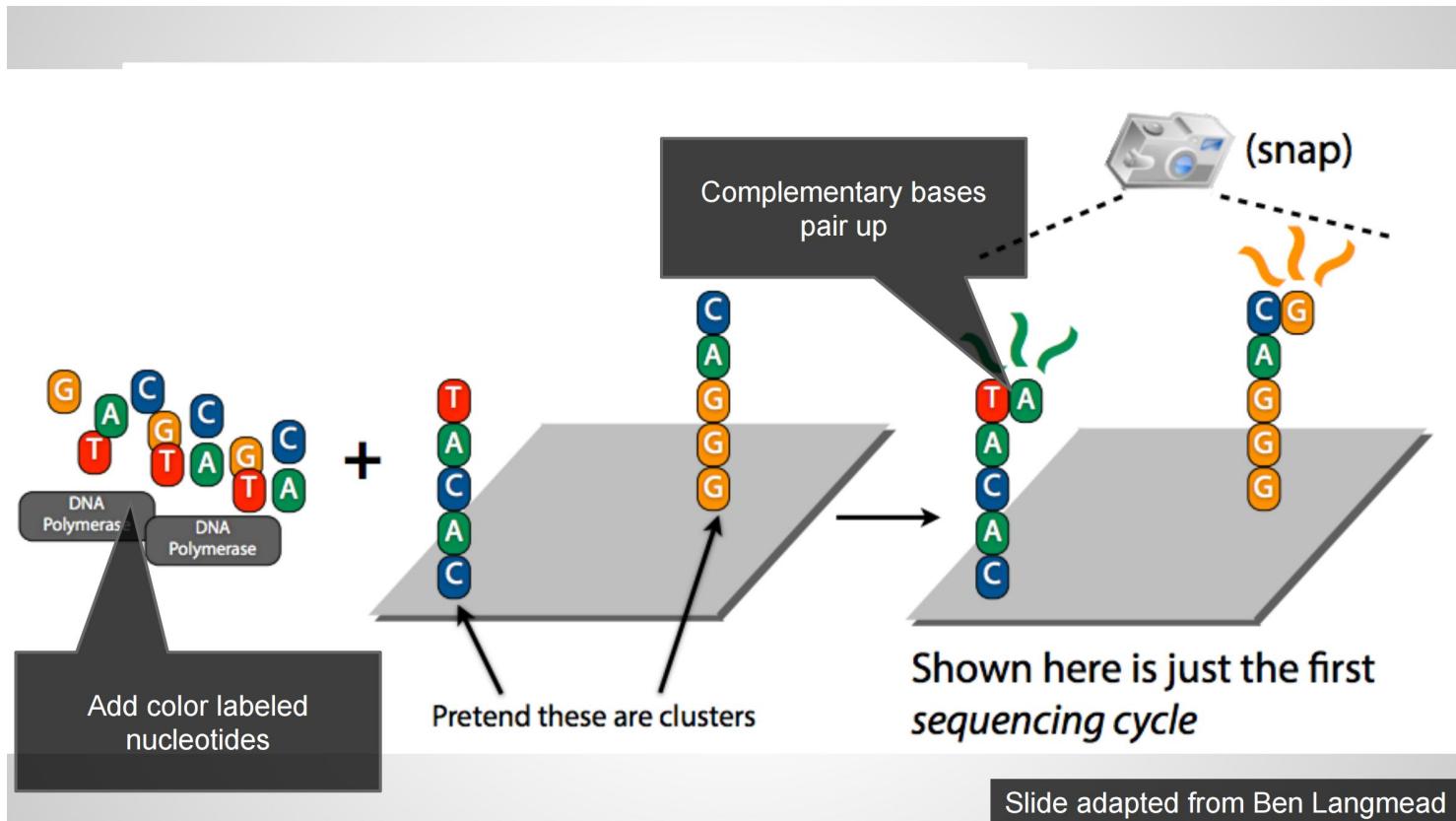


Bridge amplification

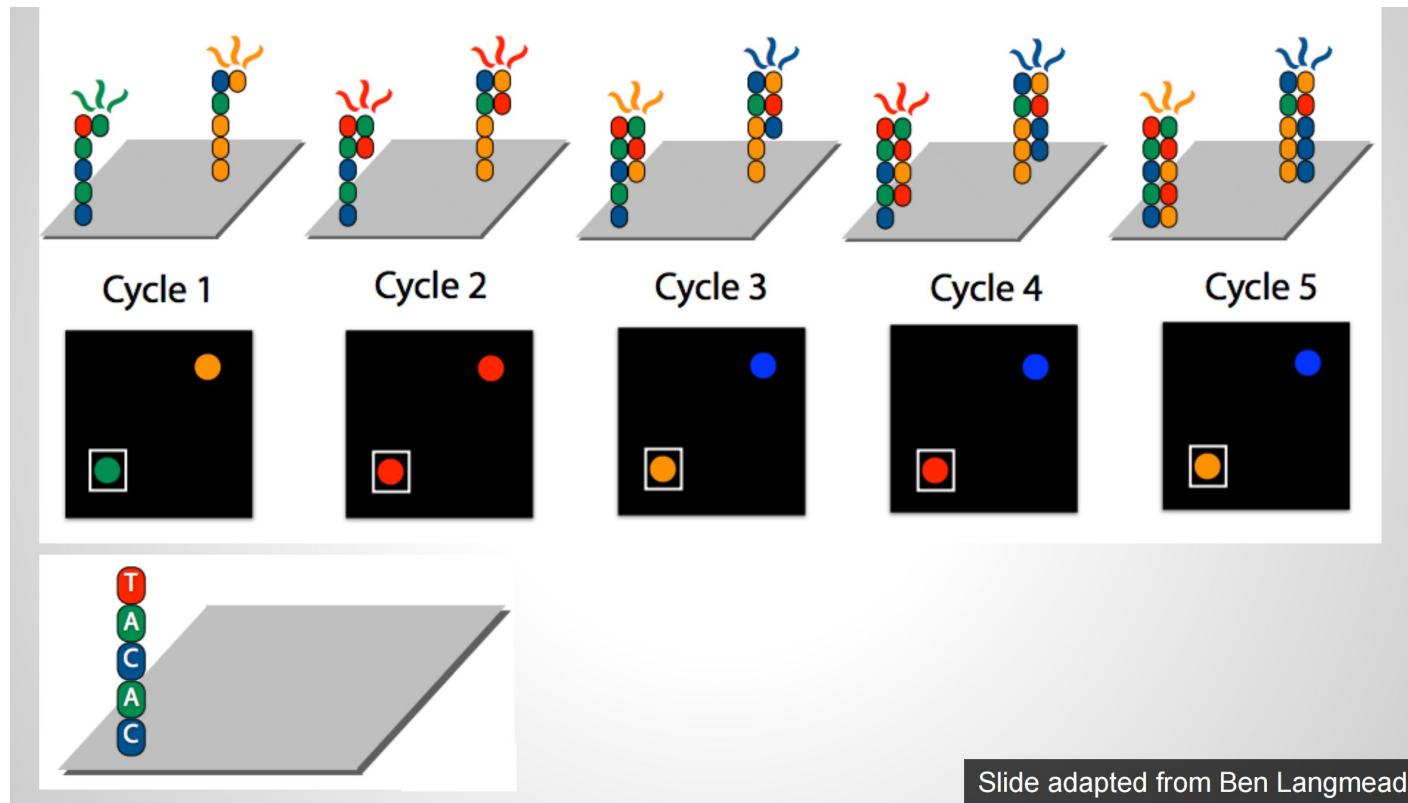
# Sequencing (Illumina)



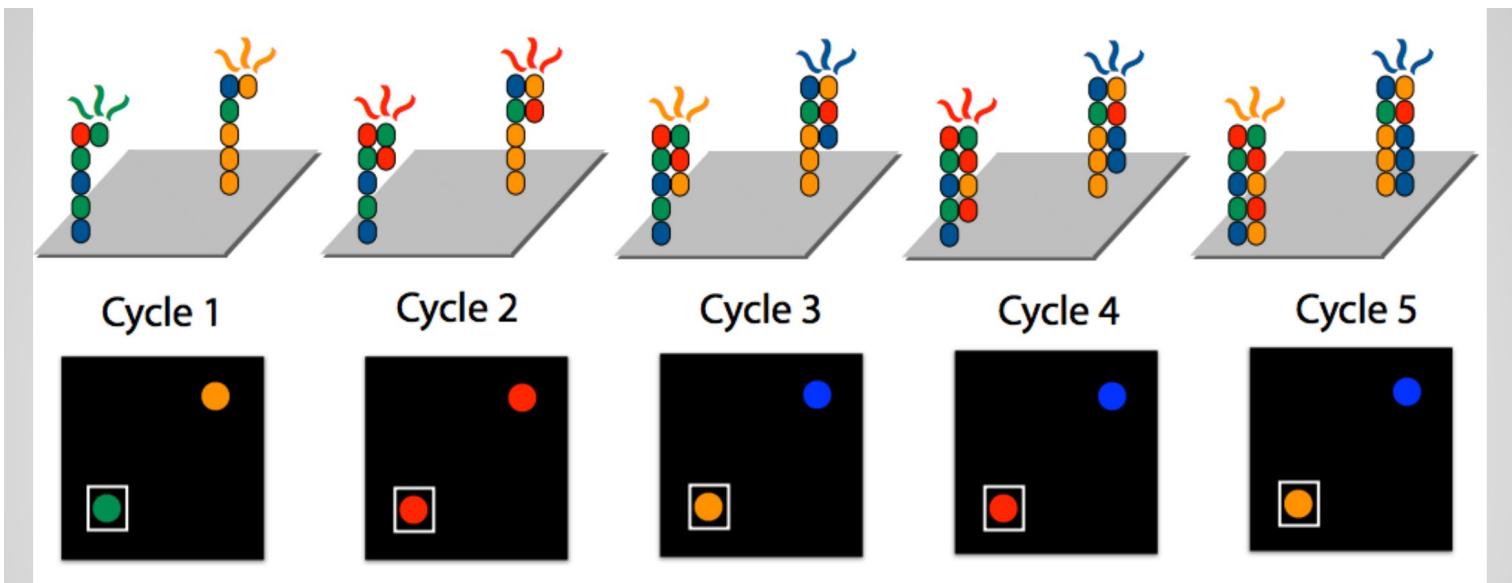
# Sequencing (Illumina)



# Sequencing (Illumina)

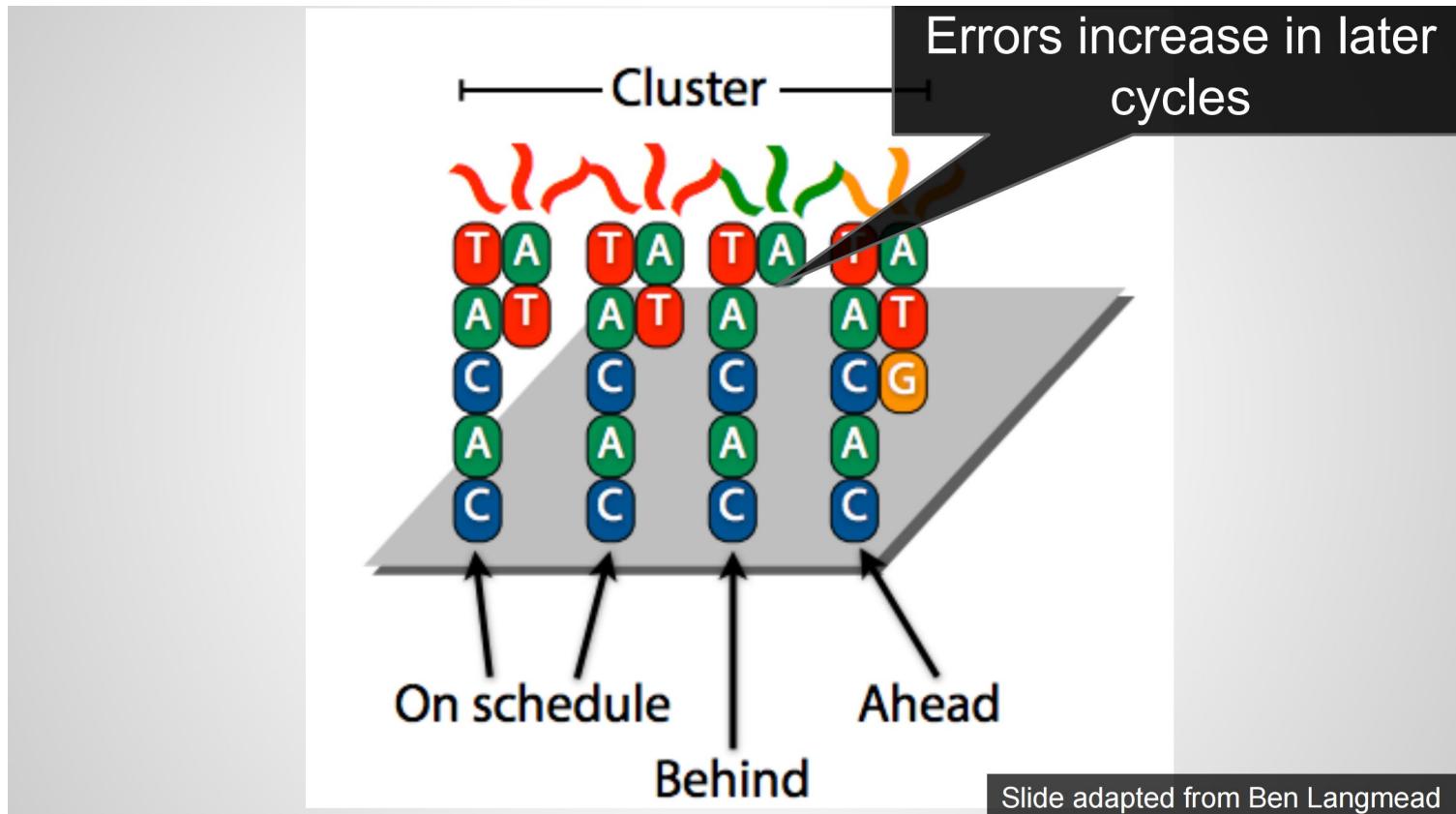


# Sequencing (Illumina)



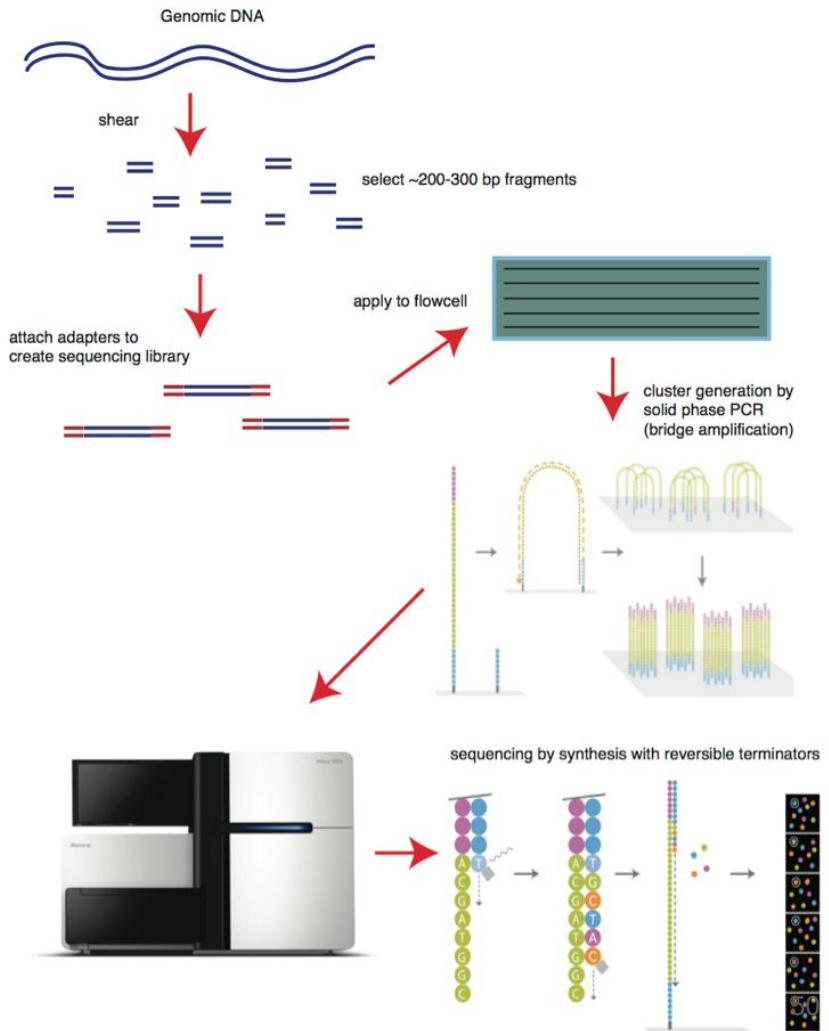
Slide adapted from Ben Langmead

# Sequencing error

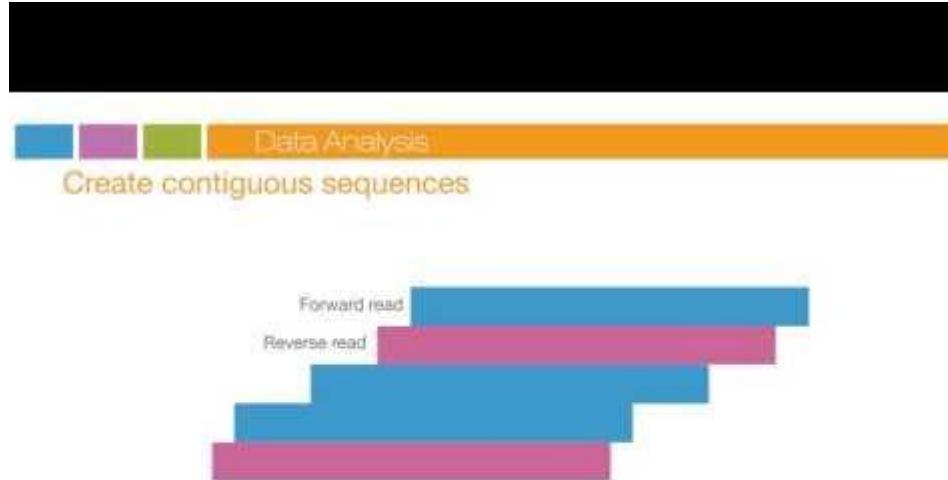


# Sequencing (sum up)

1. Shearing (fragmentation of the genome)
2. Attaching adapters
3. PCR amplification (optional)
4. Attaching template to surface/flowcel
5. PCR/bridge amplification (cluster creation)
6. Adding fluorescent bases and taking a picture after each cycle (repeat this many times)
7. Stack up images and read the sequence



# Illumina sequencing

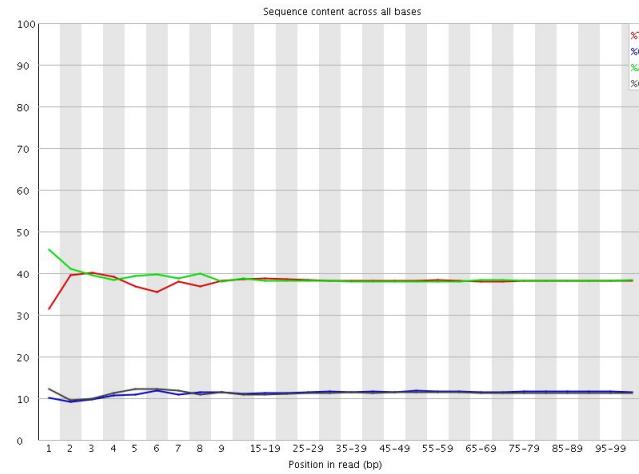
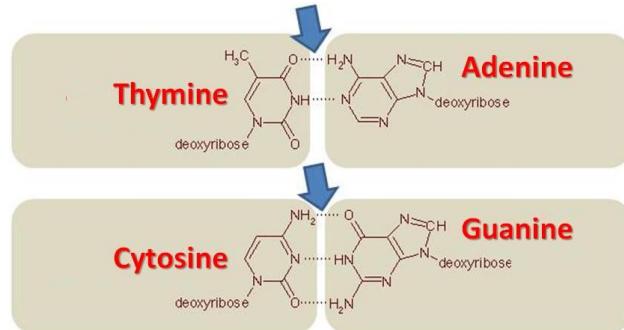


# Sequencing errors

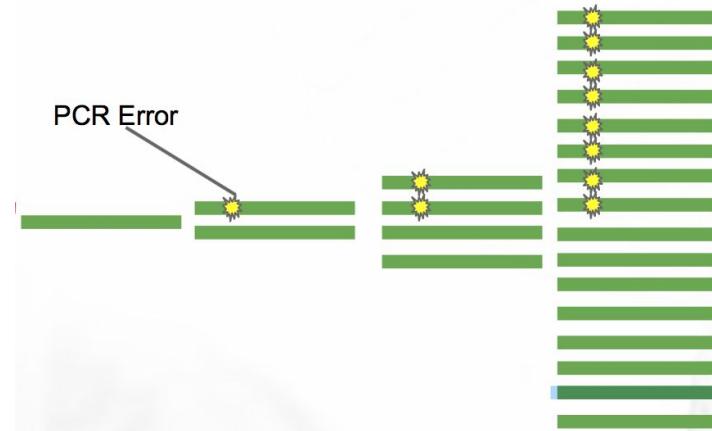
## 1. GC bias

35% to 60% - human

~20% - Plasmodium falciparum

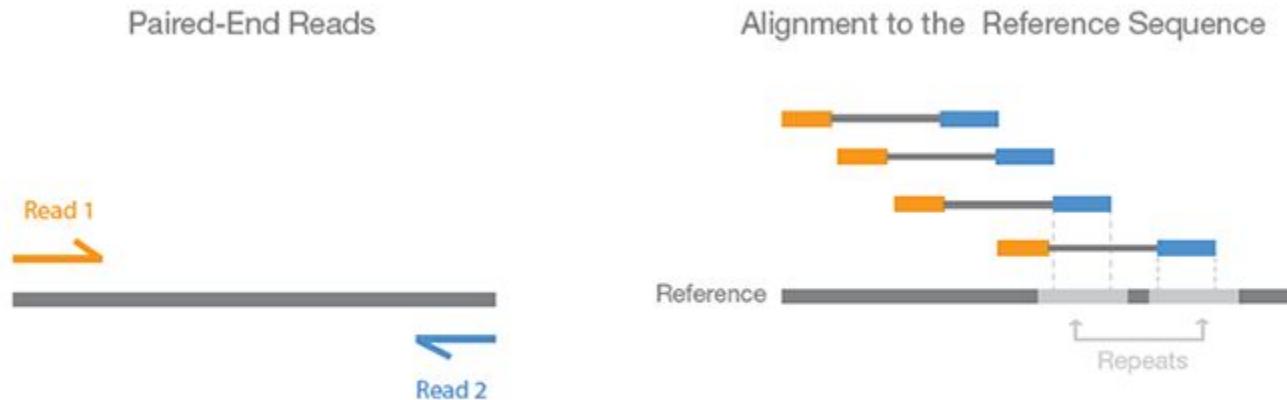


## 2. Error propagation (1 in 10.000 error rate)



# Paired-end sequencing

Figure 4. Paired-End Sequencing and Alignment

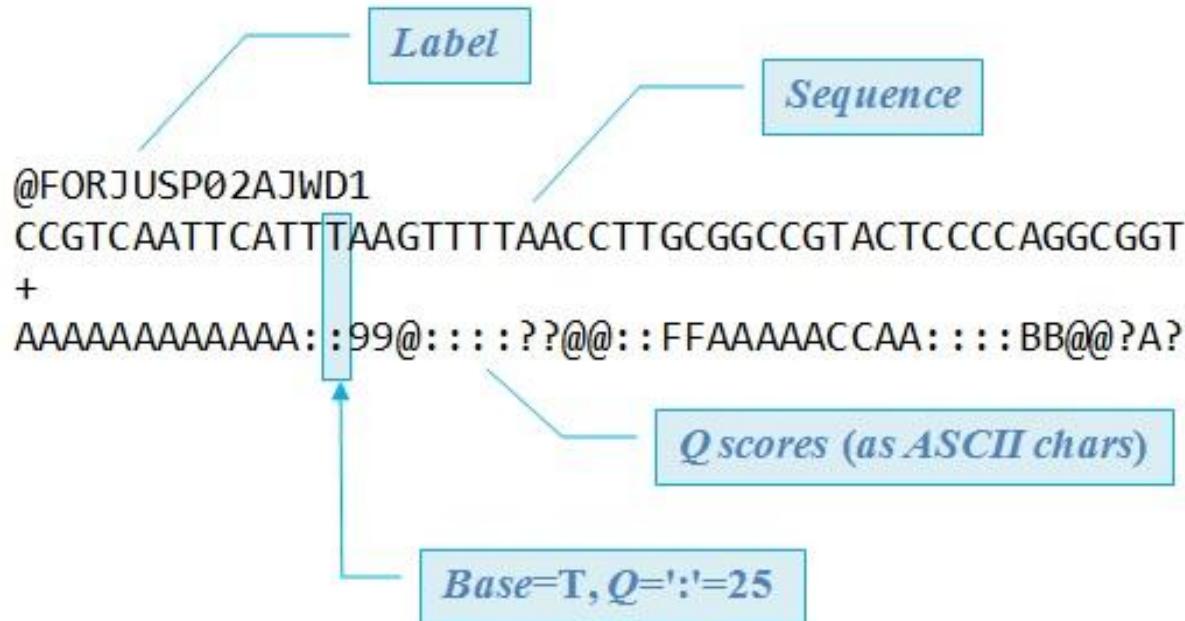


Paired-end sequencing enables both ends of the DNA fragment to be sequenced. Because the distance between each paired read is known, alignment algorithms can use this information to map the reads over repetitive regions more precisely. This results in much better alignment of the reads, especially across difficult-to-sequence, repetitive regions of the genome.

# Sequencing data - FASTQ file

4 lines for each read

- Read id
- Read sequence
- + sign
- ASCII encoded quality



## Sequencing data - FASTQ file

# Genome reconstruction

Result of sequencing experiment

- FASTQ file
- 100-500 GB
- Each read(line) containing a genome sequence 50-250 bp long



# Genome reconstruction

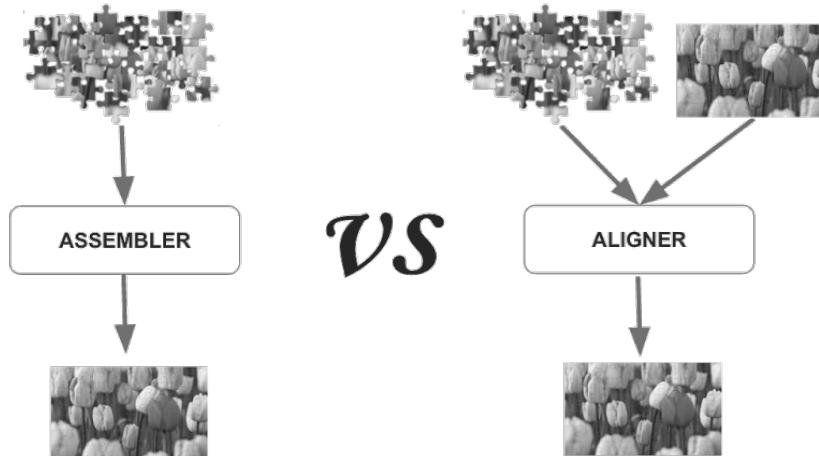
How do we reconstruct genome from reads?

## 1. Alignment

- Using reference genome to map the position of the reads

## 2. Assembly

- Reconstructing the genome by finding the links between the reads



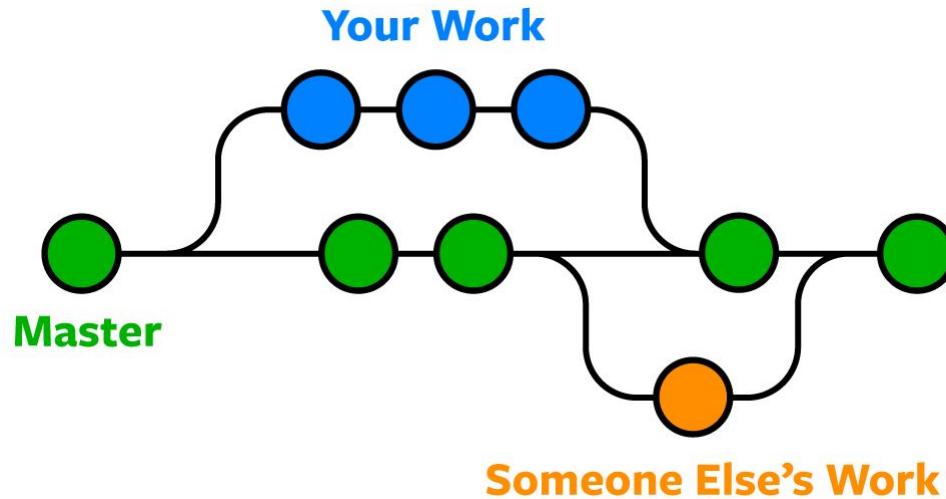
# Alignment

AAGGACAAGA	TCTTTTATG	
ATGA <del>CCAC</del>	<del>GA</del> ATGC <del>AAGG</del>	<del>CCAC</del> <del>A</del> TCTTT
ATGATTAGA		

# Assembly

AAGGACAAGA TCTTTTATG  
ATGA~~CCAC~~ GAATGC~~AAGG~~ CCAC~~A~~TCTTT  
ATGATTAGA

# Why Git?



# We use Git!

- Created by Linus Torvalds, creator of Linux, in 2005
- Came out of Linux development community
- Designed to do version control on Linux kernel
- Goals of Git:
  - Speed
  - Support for non-linear development (thousands of parallel branches)
  - Fully distributed
  - Able to handle large projects efficiently

(A "git" is a cranky old man. Linus meant himself.)

- Instructions to install Git: <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

# Installing/learning Git!

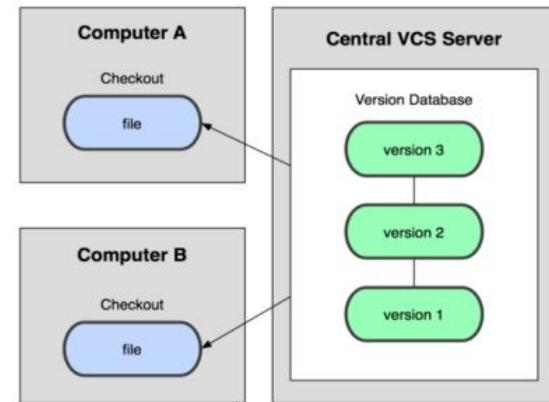
- Git website: <http://git-scm.com/>
- Free online book: <http://git-scm.com/book>
- Reference page for Git: <http://gitref.org/index.html>
- Git tutorial: <http://schacon.github.com/git/gittutorial.html>
- Git slides: <https://courses.cs.washington.edu/courses/cse403/13au/lectures/git.ppt.pdf>
- Git for Computer Scientists: <http://eagain.net/articles/git-for-computer-scientists>
- At command line: (where verb = config, add, commit, etc.)

```
git help verb
```

- Instructions to install Git: <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

# Centralized Versioning Control System

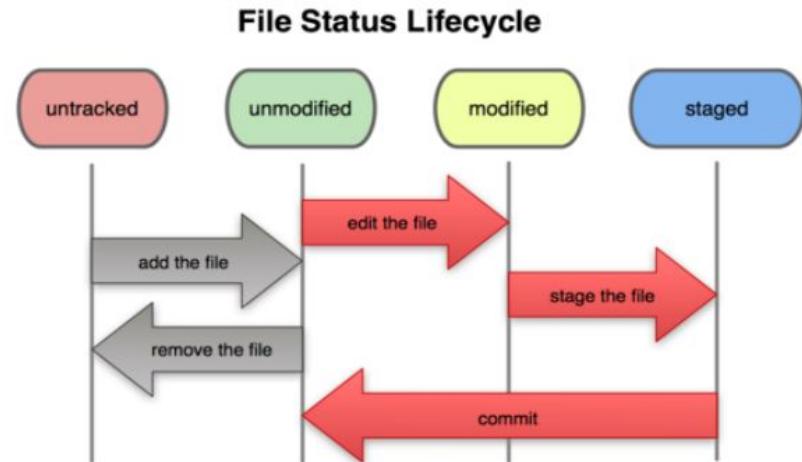
- A central server repository (repo) holds the "official copy" of the code
- The server maintains the sole version history of the repo
- You make "checkouts" of it to your local copy
- You make local modifications
- Your changes are not versioned
- When you're done, you "check in" back to the server
- your check in increments the repo's version



# Basic Git flow

1. Modify files in your working directory
2. Stage files, adding snapshots of them to your staging area
3. Commit, which takes the files in the staging area
4. Store that snapshot permanently to your Git directory

```
git add file.py  
git commit -m "Description of change."  
git push origin master
```



# Initial Git configuration

Set the name and email for Git to use when you commit:

- git config --global user.name "Bugs Bunny"
- git config --global user.email bugs@gmail.com

You can call git config –list to verify these are set.

# Git commands

command	description
<code>git clone url [dir]</code>	copy a Git repository so you can add to it
<code>git add file</code>	adds file contents to the staging area
<code>git commit</code>	records a snapshot of the staging area
<code>git status</code>	view the status of your files in the working directory and staging area
<code>git diff</code>	shows diff of what is staged and what is modified but unstaged
<code>git help [command]</code>	get help info about a particular command
<code>git pull</code>	fetch from a remote repo and try to merge into the current branch
<code>git push</code>	push your new branches and data to a remote repository
<code>git checkout filename</code>	undoes your changes
Others: init, reset, branch, checkout, merge, log, tag	

# We use Github!

- GitHub.com is a site for online storage of Git repositories.
- You can create a remote repo there and push code to it.
- Many open source projects use it, such as the Linux kernel.
- You can get free space for open source projects, or you can pay for private projects.
- Free private repos for educational use: [github.com/edu](https://github.com/edu)
- Question: Do I always have to use GitHub to use Git?
  - Answer: No! You can use Git locally for your own purposes.
  - Or you or someone else could set up a server to share files.
  - Or you could share a repo with users on the same file system, as long everyone has the needed file permissions).

# Setup Github repo

- Create account on  
[www.github.com](https://www.github.com)
- Set an image :)
- Create repository

My Data Science Center

- Initialize with README
- .gitignore Python
- MIT License

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner  / Repository name \*  

Great repository names are short. Your new repository will be created as My-Data-Science-Center. [right?](#)

Description (optional)

 Public  
Anyone on the internet can see this repository. You choose who can commit.

 Private  
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

Initialize this repository with a README  
This will let you immediately clone the repository to your computer.

Add .gitignore: Python  

**Create repository**

# Setup Github repo

- Add short biography
- Projects will come on the way

The screenshot shows a GitHub repository page for 'My-Data-Science-Center'. The top navigation bar includes links for Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security (0), Insights, and Settings. There are also buttons for Unwatch (1), Star (0), Fork (0), Find file, and Copy path. The main content area shows a commit history with one entry from 'vladimirkovacevic' updating the README.md file. The commit hash is b3e3a5a and it was made 'now'. Below the commit, the README.md content is displayed:

```
My-Data-Science-Center

Who am I

...
Projects

...
```

At the bottom, there are links for Raw, Blame, History, and a copy icon. The footer of the page includes copyright information for GitHub, Inc., and links for Terms, Privacy, Security, Status, Help, Contact GitHub, Pricing, API, Training, Blog, and About.

# Resources and additional reads

Presentation available at: [github.com/vladimirkovacevic/gi-2023-etf](https://github.com/vladimirkovacevic/gi-2023-etf)

- [A Computer Scientist's Guide to Cell Biology, A Travelogue from a Stranger in a Strange Land](#)
- [Genomics 101, Edition 2016](#)
- [Bioinformatics at COMAV - SNP Calling](#)
- [Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM](#)
- [High-Throughput Sequencing Technologies - Review paper](#)
- Vince Buffalo: Bioinformatics Data Skills
- Dan Gusfield: Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology, Cambridge
- Pavel Pevzner, Neils Jones: An Introduction to Bioinformatics Algorithms (Computational Molecular Biology), MIT
- R. Durbin, S. Eddy, A. Krogh, G. Mitchinson: Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids, Cambridge University Press
- Veli Mäkinen, Djamal Belazzougui, Fabio Cunial, Alexandru I. Tomescu: Genome-Scale Algorithm Design: Biological Sequence Analysis in the Era of High-Throughput Sequencing, Cambridge University press
- Molekularna biologija 1; Dušanka Savić-Pavićević, Gordana Matić; NNK International, 2020
- [RNA-seqlopedia](#); Cresko Lab, University of Oregon