أكاديمية كاوست
KAUST ACADEMY

Introduction to Bioinformatics

Stage 03

Day 01 – Session 01

The KAUST Academy & The Bioinformatics Platform

4-7 Feb 2026

أكاديمية كاوست
KAUST ACADEMY

Instructors

Husen Umer
Bioinformatics Scientist

Norah AlGhamdi
Bioinformatics PostDoc

Salim Bougouffa
Senior Bioinformatician

Support

Shouq Kaaki

Ahmed Bahaj

Suhaib AlGhamdi

Khulud Alharbi

Rahaf Kassar

Agenda – Day 01

أكاديمية كاوست
KAUST ACADEMY

Morning Session

09:00-09:30	S1: Welcome and Introduction
09:30-10:00	S2: Recap: Foundations in Bioinformatics & Overview of Stage 3
10:00-10:30	S3: Bioinformatics Computing Environments
10:30-12:00	*L1: Warming-up for Large-scale Analysis & Genomic File Formats*

Afternoon Session

02:00-02:45	S4: Transcriptomics Overview & Applications
02:45-03:30	*L2: Hands-on with Transcriptomics Data*
03:30-04:15	S5: RNA-seq Experimental Design
04:15-05:00	S6: Team Formation & Project Overview

Course Objectives:

أكاديمية كاوست
KAUST ACADEMY

1. Develop foundational skills in bioinformatics and genomics data analysis

2. Gain hands-on experience using Linux-based computational environments and standard bioinformatics tools

3. Build a solid understanding of transcriptomics and RNA sequencing (RNA-seq) principles and applications

4. Work confidently with commonly used genomics and transcriptomics data formats

5. Experience a complete, end-to-end RNA-seq analysis workflow, from raw sequencing data to biological insight, including data processing, read alignment, differential expression, and functional enrichment analysis

6. Solve real bioinformatics challenges and present bioinformatics findings

Practical Instructions:

أكاديمية كاوست
KAUST ACADEMY

- Dates: February 4-7, 2026 (Wednesday - Saturday)
- Daily Sessions: 9:00-12:00 and 14:00-17:00
- Break: 12:00-14:00
- Q/A & project help: 16:00-17:00
- Final Day: Finishes at 14:00 (2 PM)
- Venue: Building 9, Ground floor, Room 2322 (Hall 1)

Program Timeline:

- Day 1: Foundations of Bioinformatics and Transcriptomics
- Day 2: Data Processing Pipeline
- Day 3: Downstream Analysis, Visualization, and Interpretation
- Day 4: Extended Applications, Presentations, and Evaluation

Project Timeline:

أكاديمية كاوست
KAUST ACADEMY

- Day 1: Form groups, select project
- Day 2: QC, trimming, alignment, quantification
- Day 3: Differential expression, visualization, enrichment analysis
- Day 4: Flash talk presentations (5 min per group)

Day 01 Objectives

- Recap bioinformatics foundations
- Get hands-on with computing environments
- Become familiar with genomic data
- Introduce transcriptomics concepts
- Form project teams

Agenda – Day 01

أكاديمية كاوست
KAUST ACADEMY

Morning Session

9:00-9:30	S1: Welcome and Introduction
9:30-10:00	S2: Recap: Foundations in Bioinformatics & Overview of Stage 3
10:00-10:30	S3: Bioinformatics Computing Environments
10:30-12	*L1: Warming-up for Large-scale Analysis & Genomic File Formats*

Afternoon Session

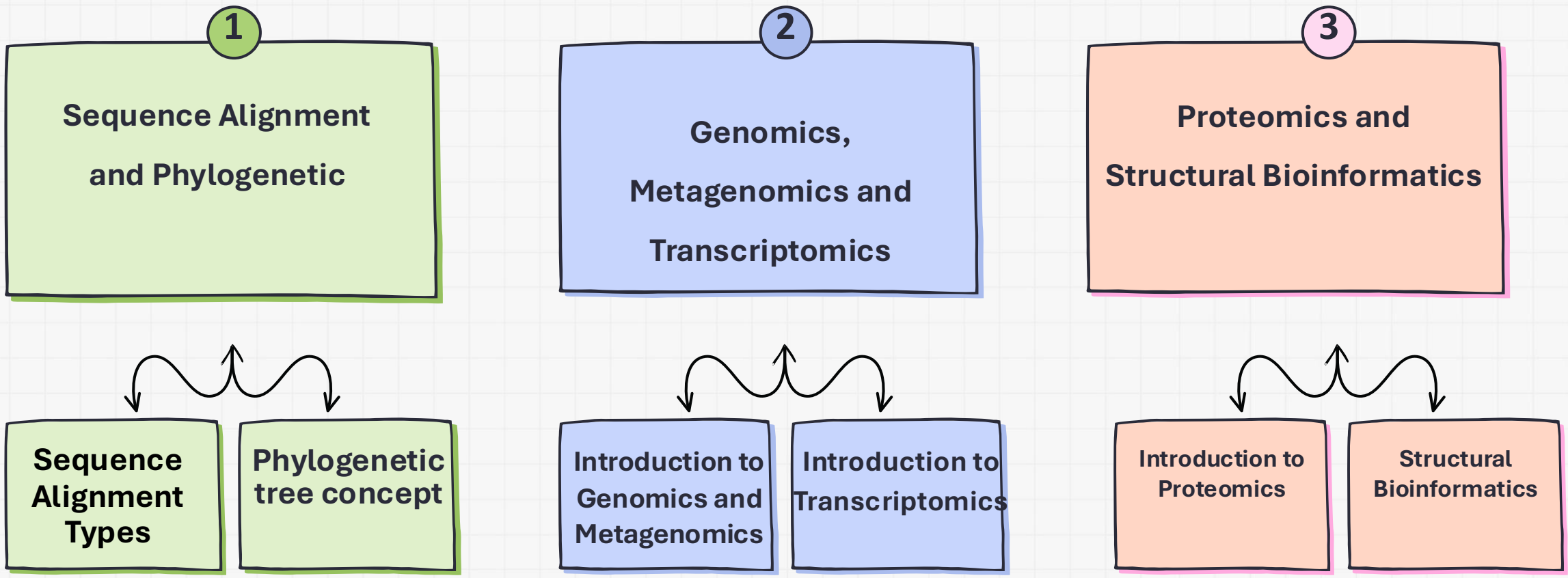
02:00-02:45	S4: Transcriptomics Overview & Applications
02:45-03:30	*L2: Hands-on with Transcriptomics Data*
03:30-04:15	S5: RNA-seq Experimental Design
04:15-05:00	S6: Team Formation & Project Overview

أكاديمية كاوست
KAUST ACADEMY

Recap: Foundations in Bioinformatics & Overview of Stage 3

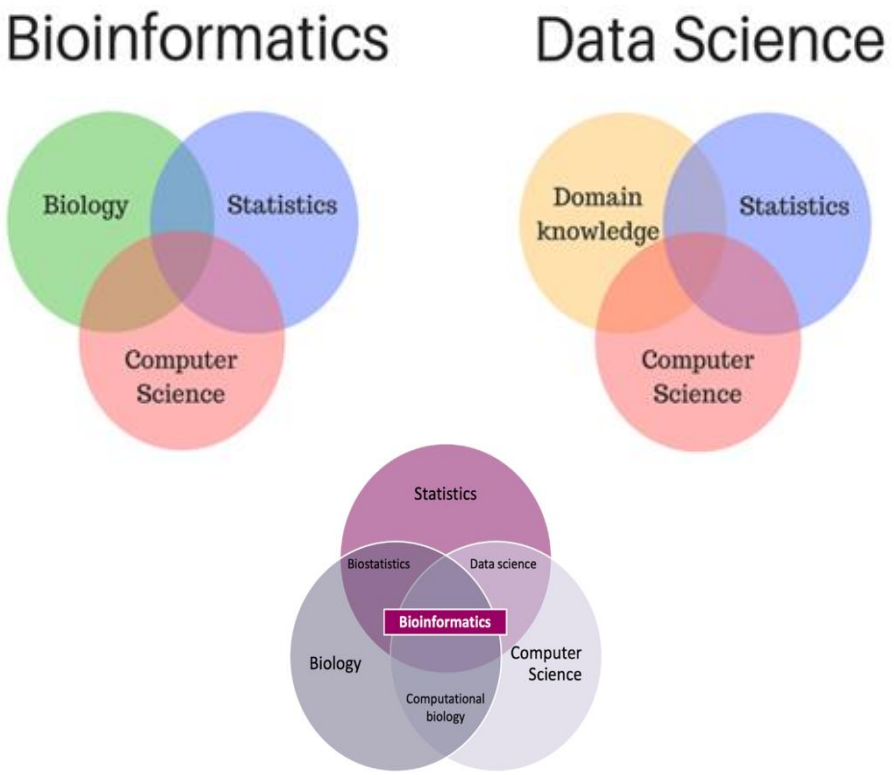
Day 01 – Session 02

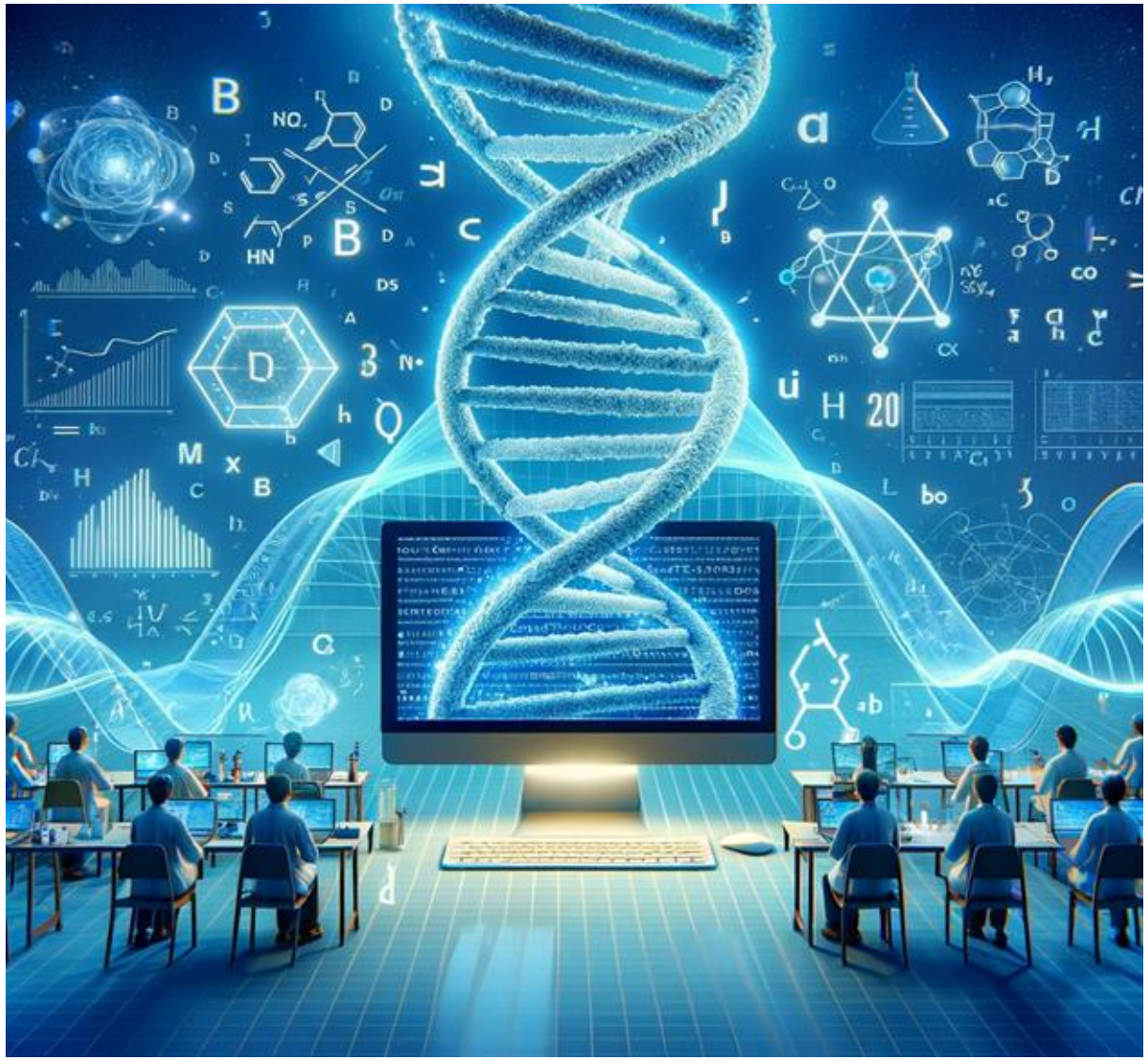
Overview of Stage 2



What is Bioinformatics?

Interdisciplinary field that combines life sciences, computer science, and mathematics to analyze and interpret biological data





Why do we need Bioinformatics ?

1. Key Complexity Factors

Volume: Massive datasets from high-throughput technologies
Variety: Different data types (Genomics, proteomics, transcriptomics, and more) and various formats (sequences, structures, networks) require specialized analysis
Velocity: Rapid data generation and updates
Variability: Inherent noise and inconsistencies in biological systems

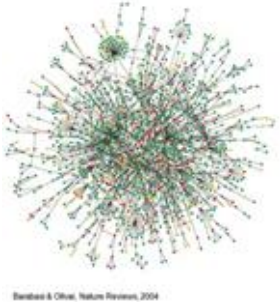
2. Challenges

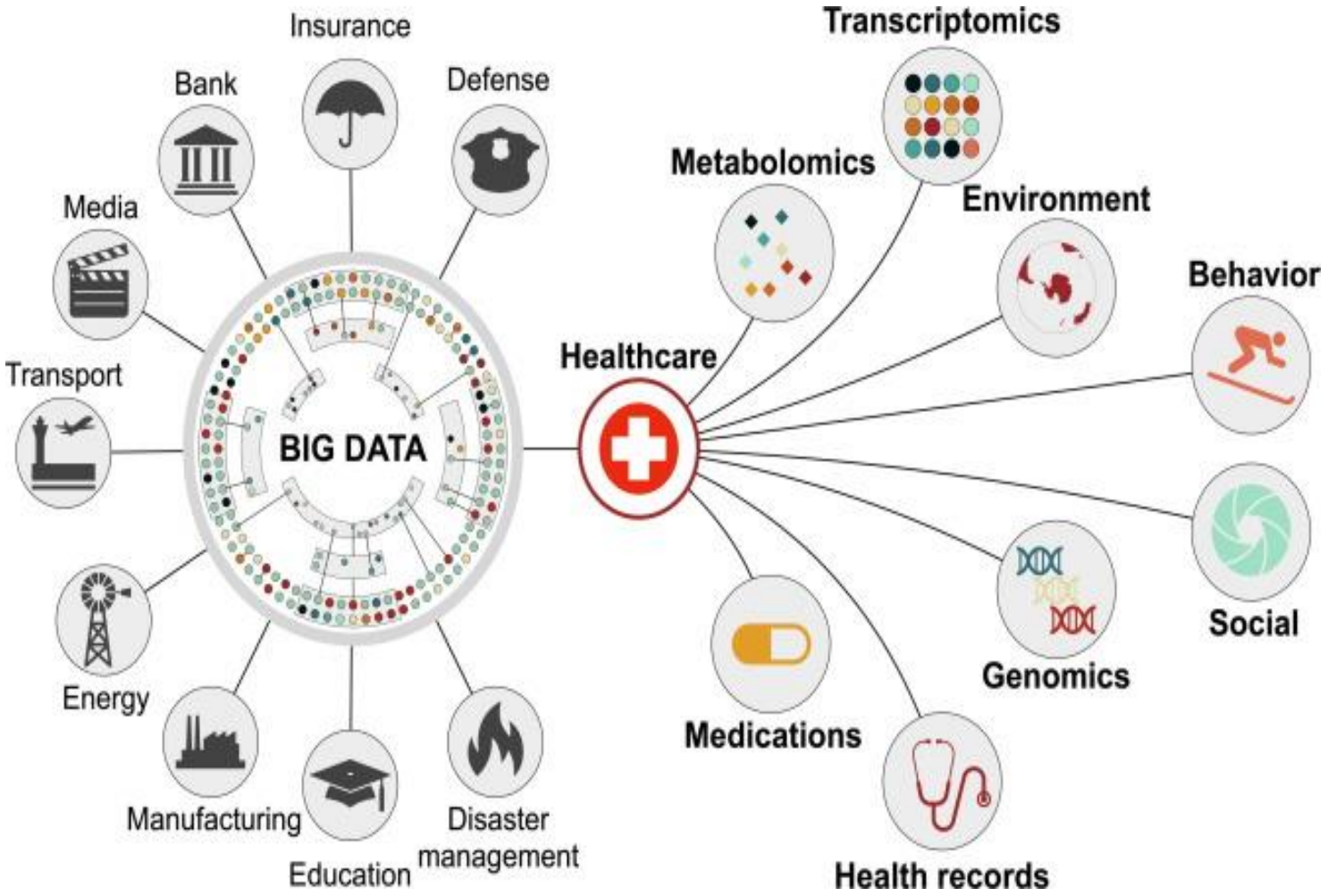
- **Integration:** Merging diverse data sources.
- **Interpretation:** Extracting meaningful insights.
- **Scalability:** Efficiently handling large datasets.

3. The Role of Bioinformatics

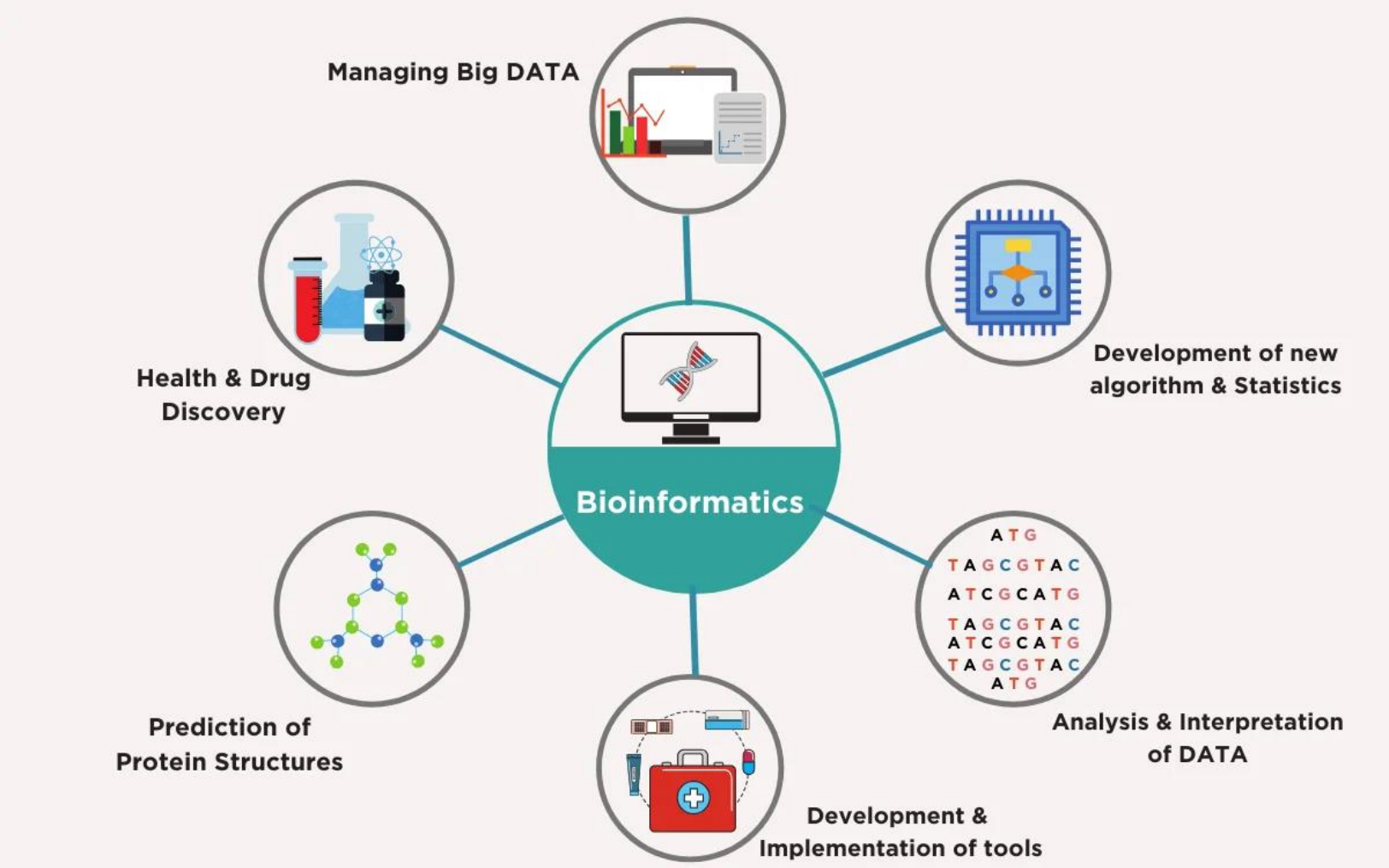
- **Solution:** Computational tools and interdisciplinary approaches to

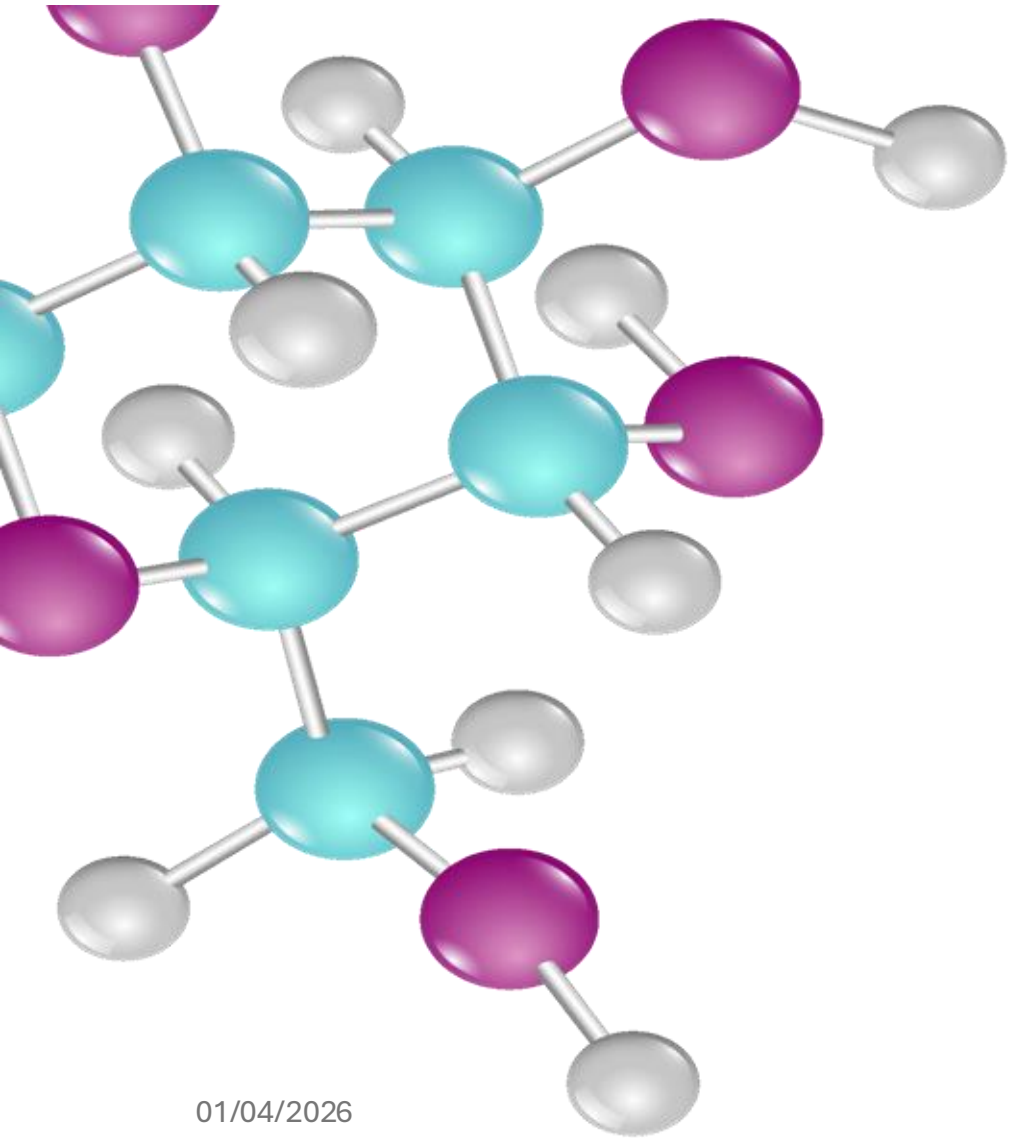
manage and interpret complex biological data.





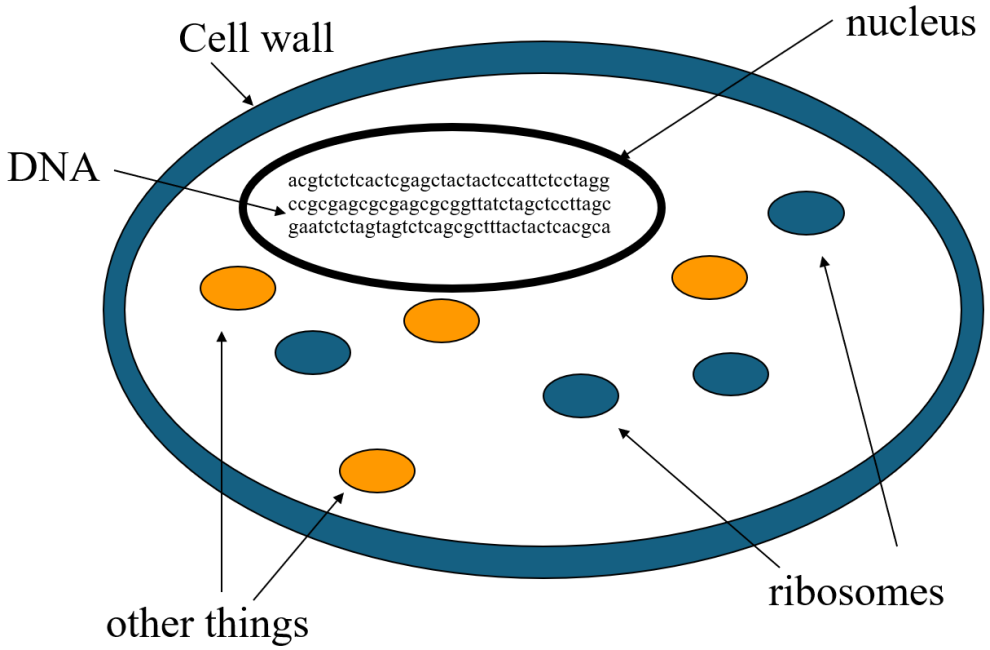
Applications



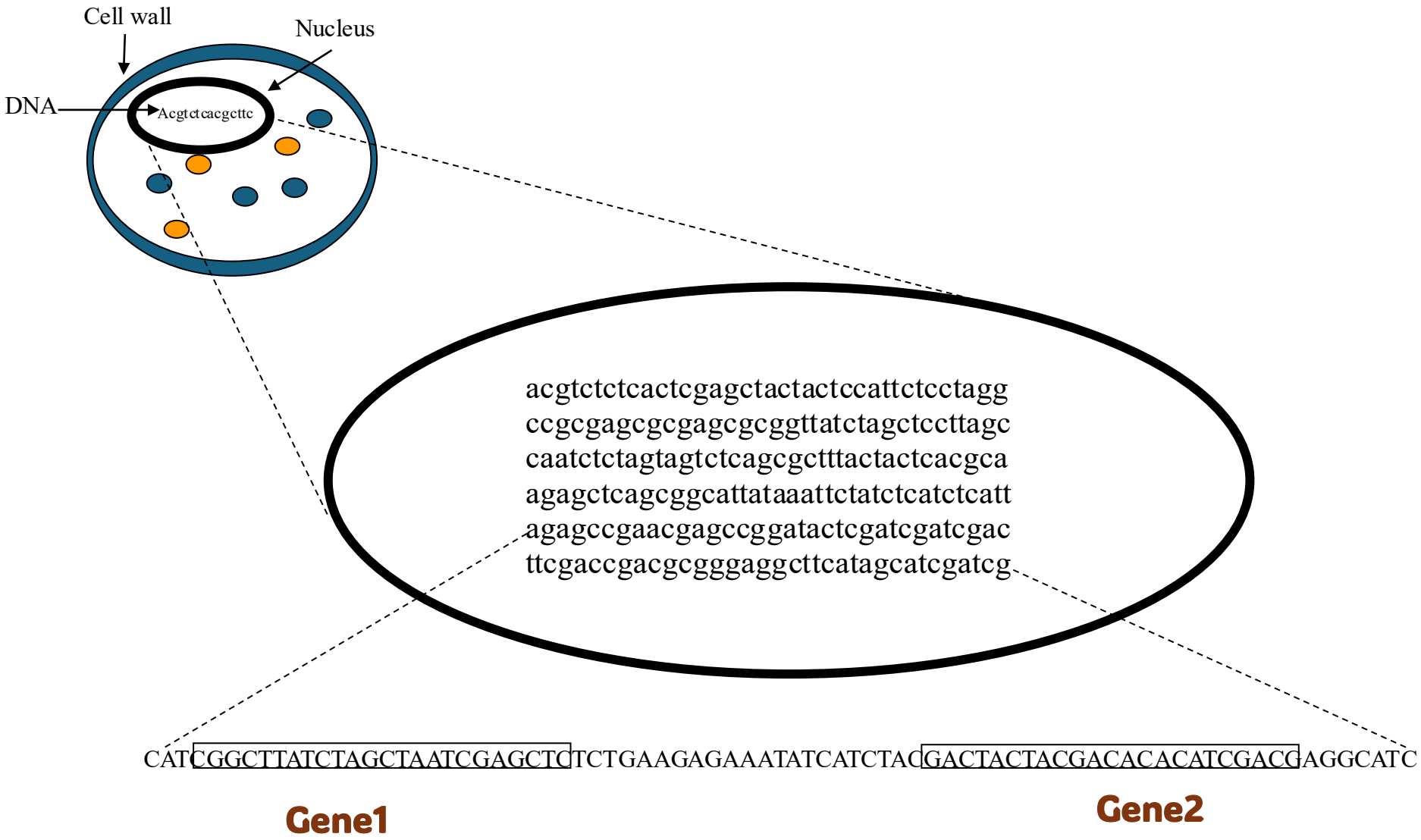


What is Molecular Biology?

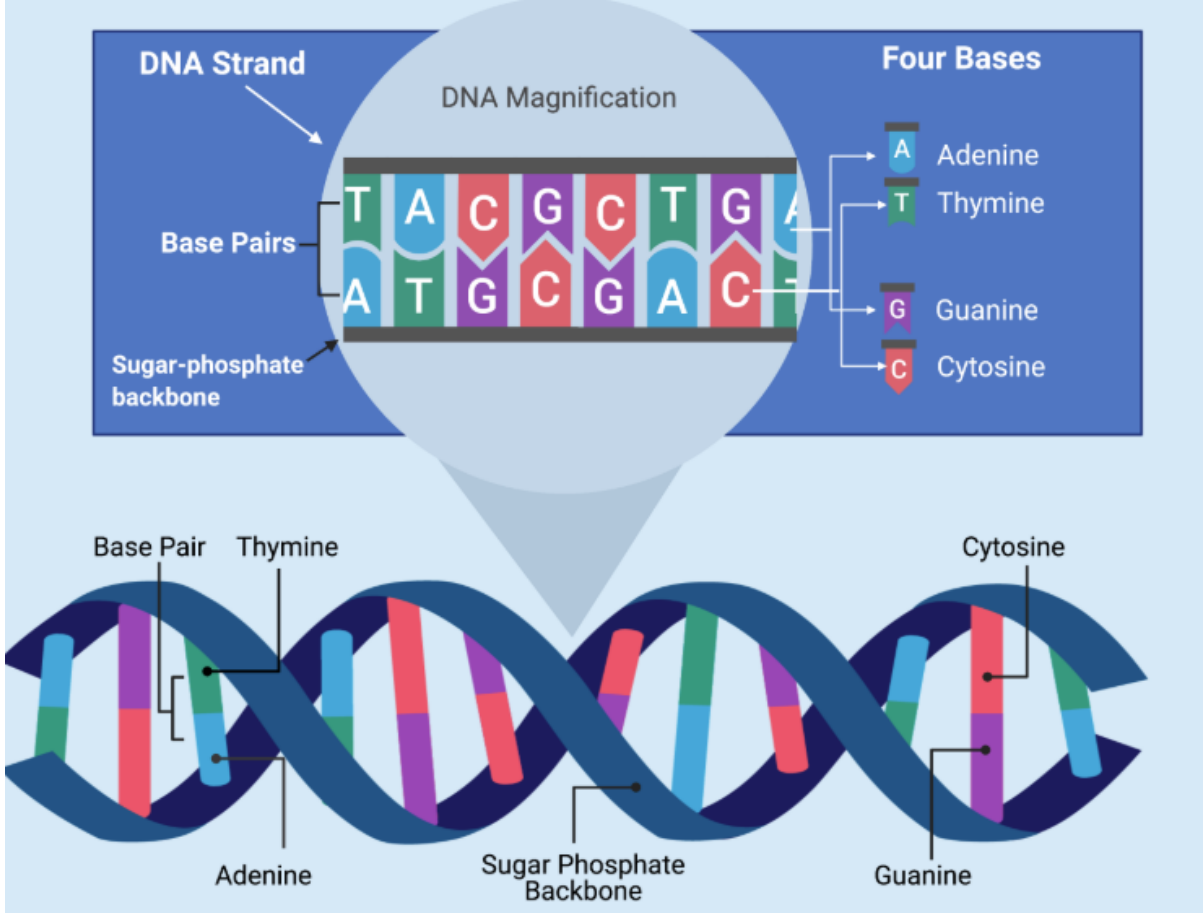
- The study of structure and function of nucleic acids and proteins of biological molecules to understand the molecular mechanisms underlying various biological processes



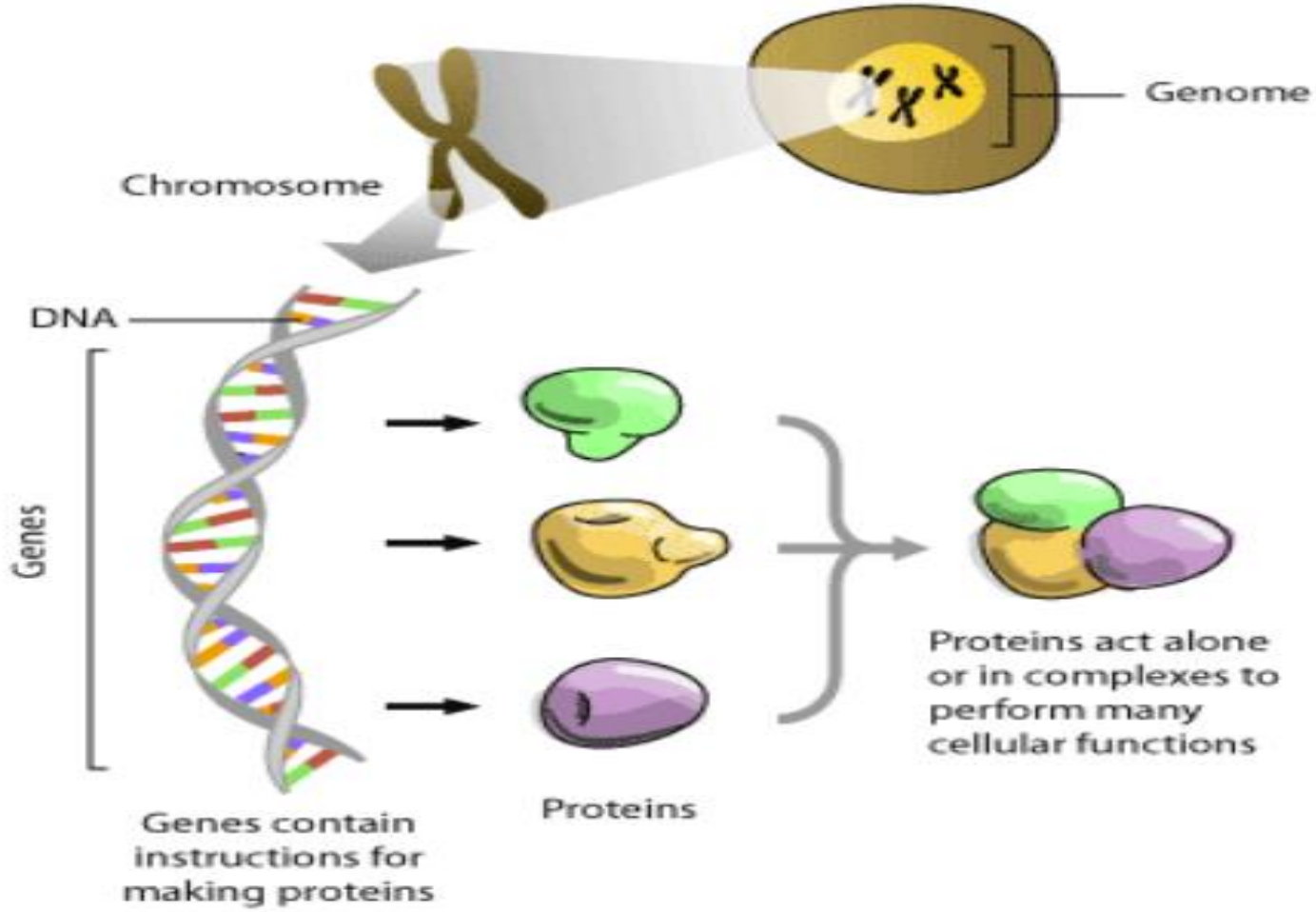
DNA


Cell wall
Nucleus
DNA
Acgtctcacgcttc
acgtctctcactcgagctactactccattctcctagg
ccgcgagcgcgagcgcggttatctagctccttagc
caatctctagtagtctcagcgctttactactcacgca
agagctcagcggcattataaattctatctcatctcatt
agagccgaacgagccggatactcgatcgatcgac
ttcgaccgacgcgggaggcttcatagcatcgatcg
CATCGGCTTATCTAGCTAATCGAGCTCTCTGAAGAGAAATATCATCTACGACTACTACGACACACATCGACGAGGCATC
Gene1
Gene2

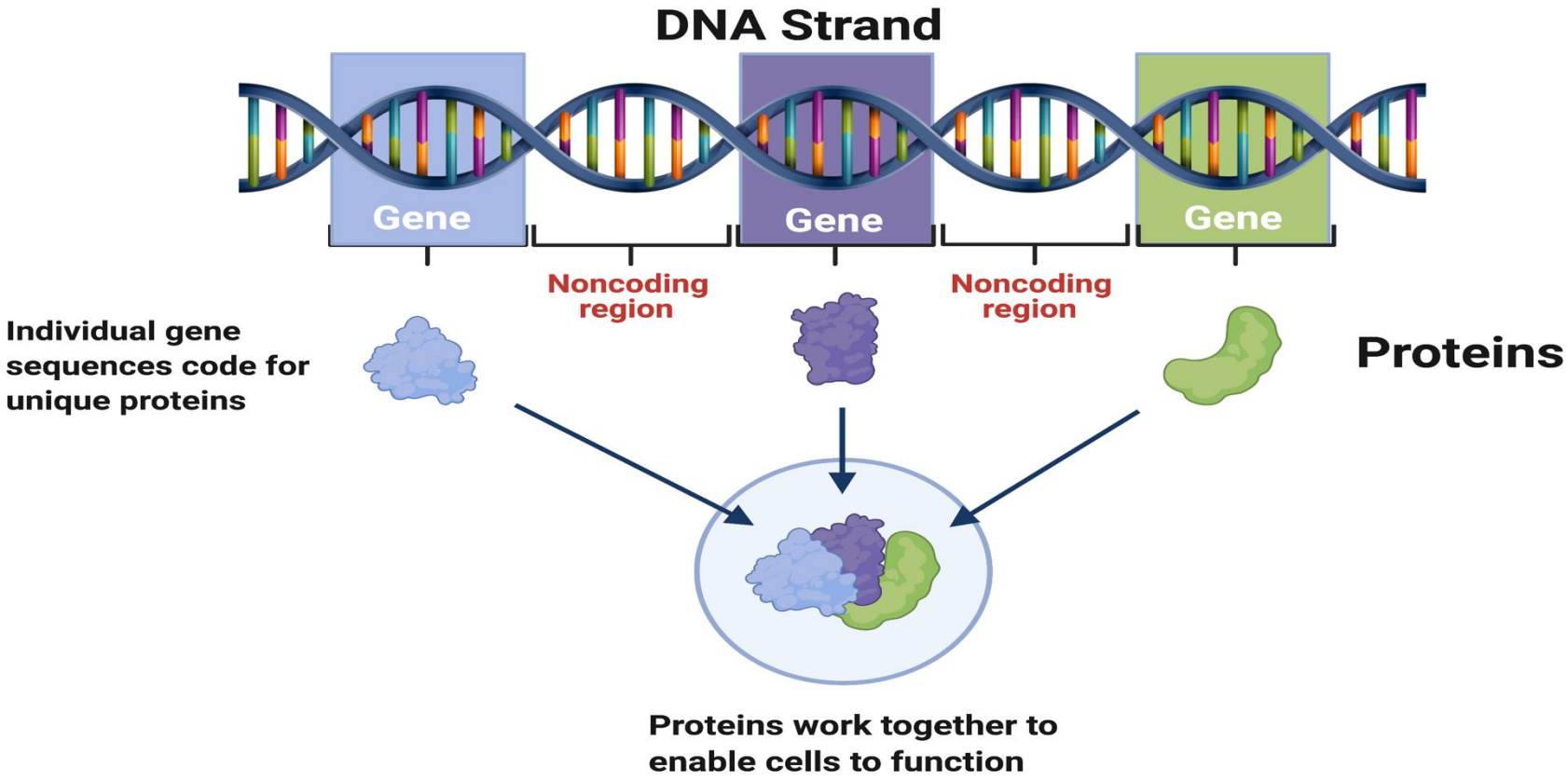
Structure of DNA



DNA contains the instructions to make proteins

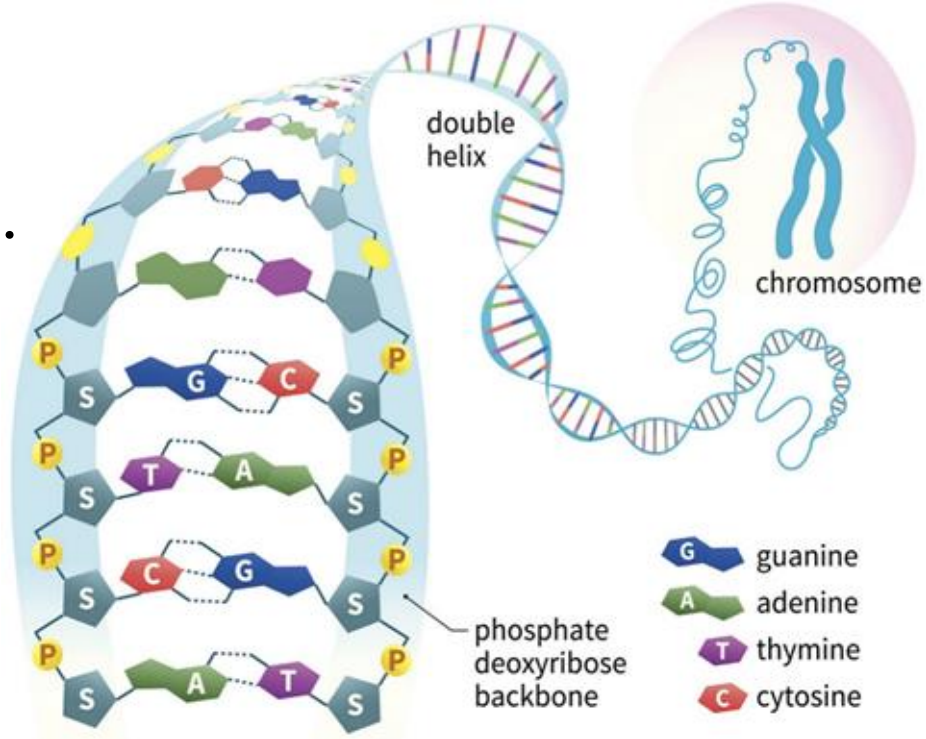


Genes and Non-Genes

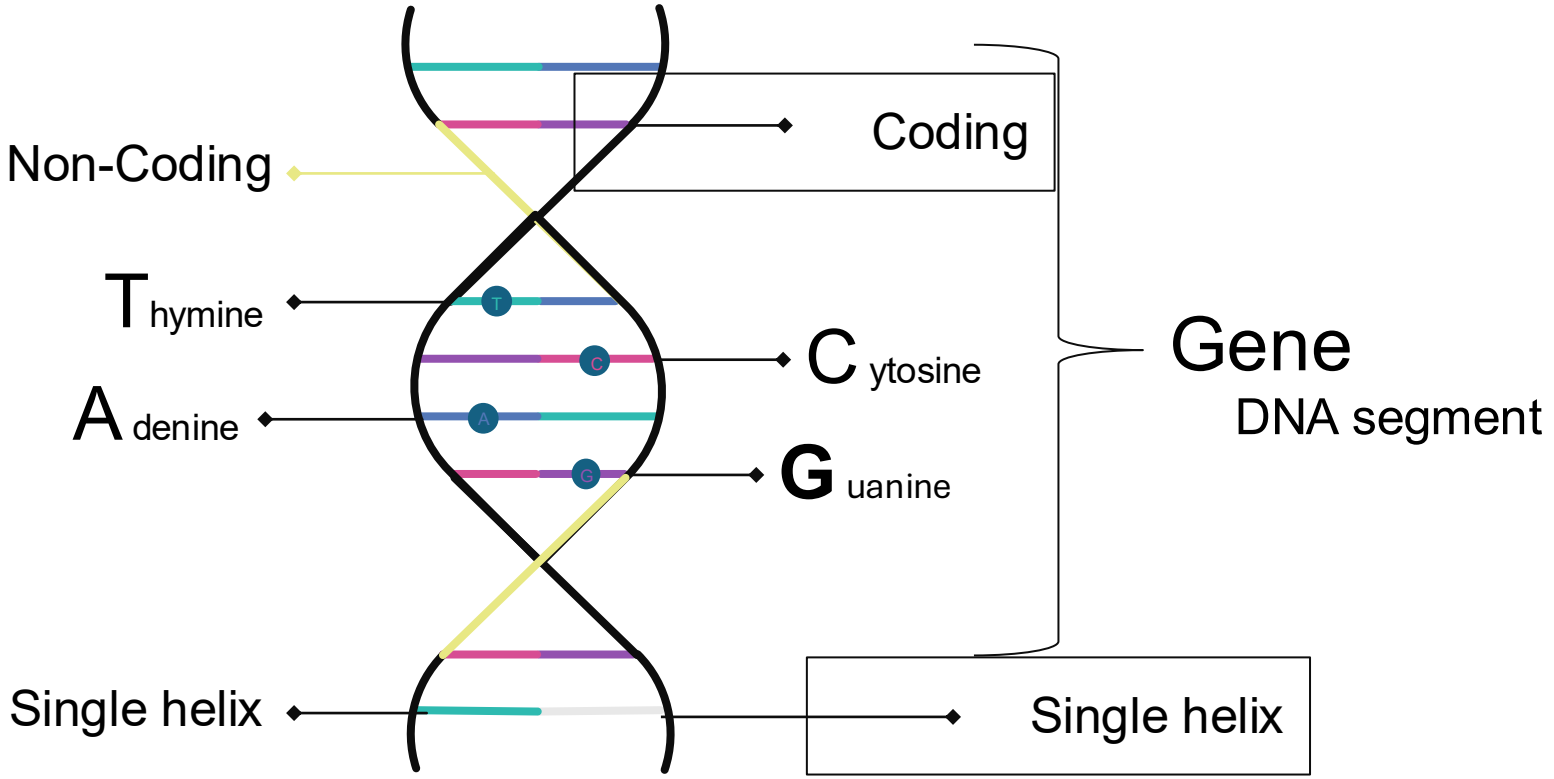


Human Genome

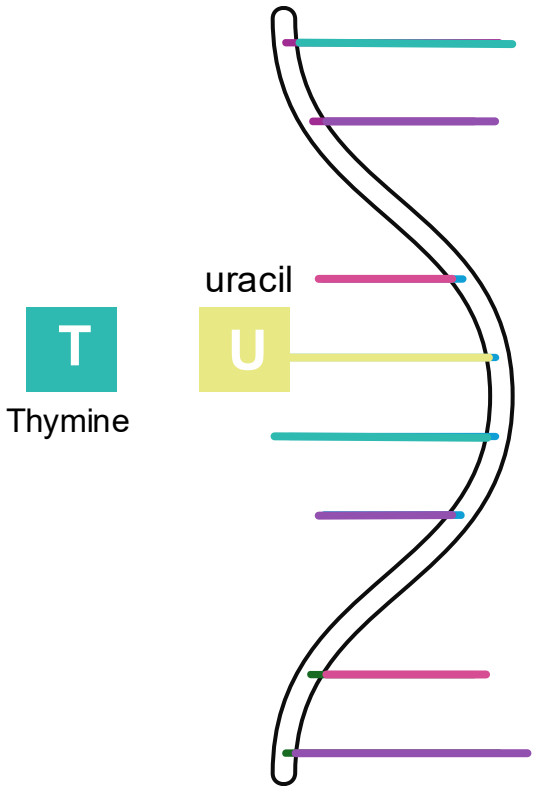
- **-ome**: totality or complete set of …
- Gen-ome: the complete set of DNA in an organism.
- Human Genome
  - Cell → Nucleus
  - Nucleus→ 23 Chromosomes
  - Chromosome→ DNA (Genome) + Histones
  - DNA→ non-genic & genes (Coding –noncoding)
  - Genes → Protein & non-protein



DNA (Double Helix)



RNA (Single Helix)



Where Does Bioinformatics Fit In?

1. We more or less know the human DNA sequence

2. We *think* there are about 25,000 genes, but we are not sure where all of them are

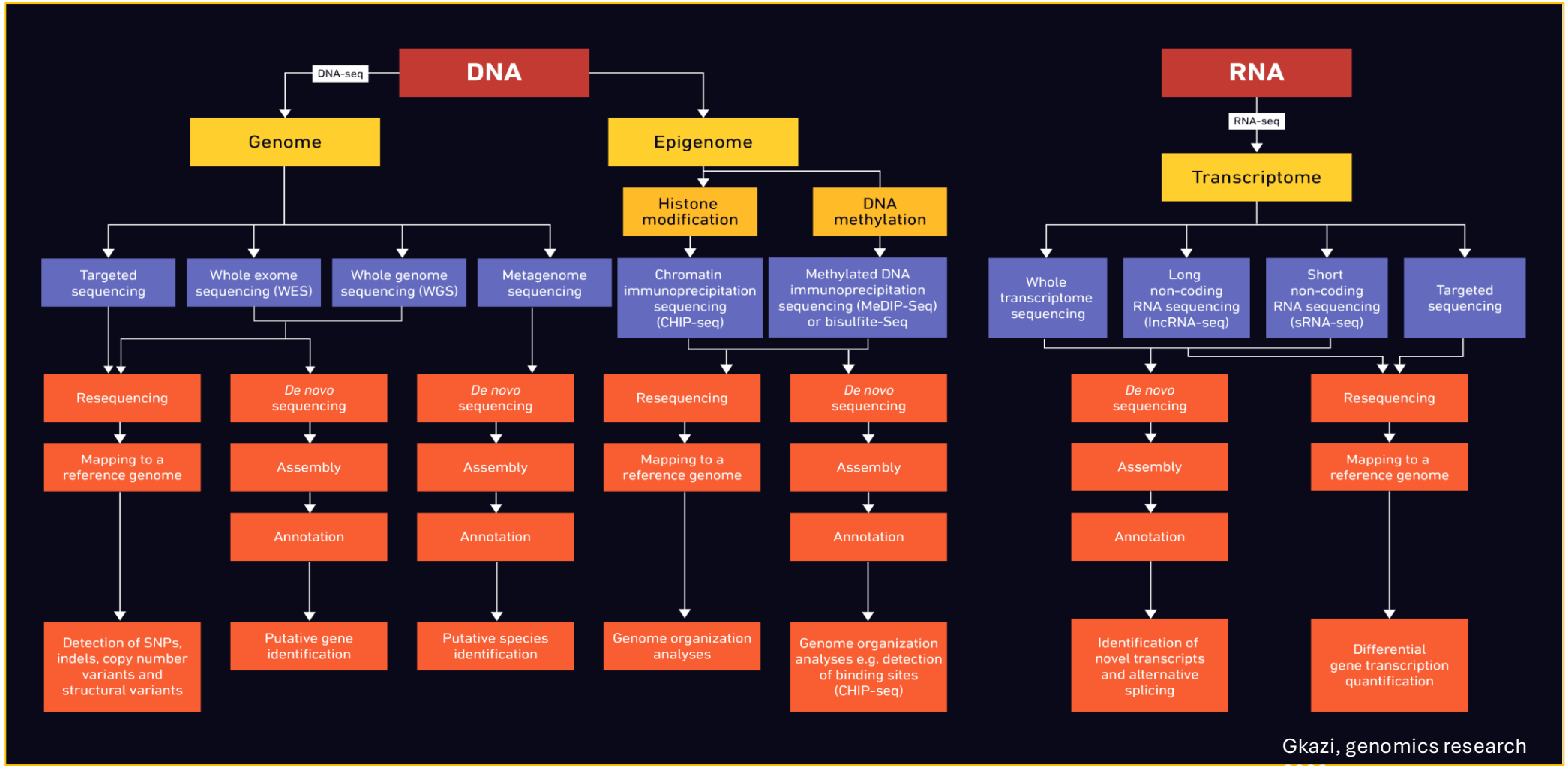
3. In a few thousand cases we know the gene, the structure of the protein it produces, and much about the function of that protein.

4. In all other cases, we usually don't know the function and mostly don't know the structure..

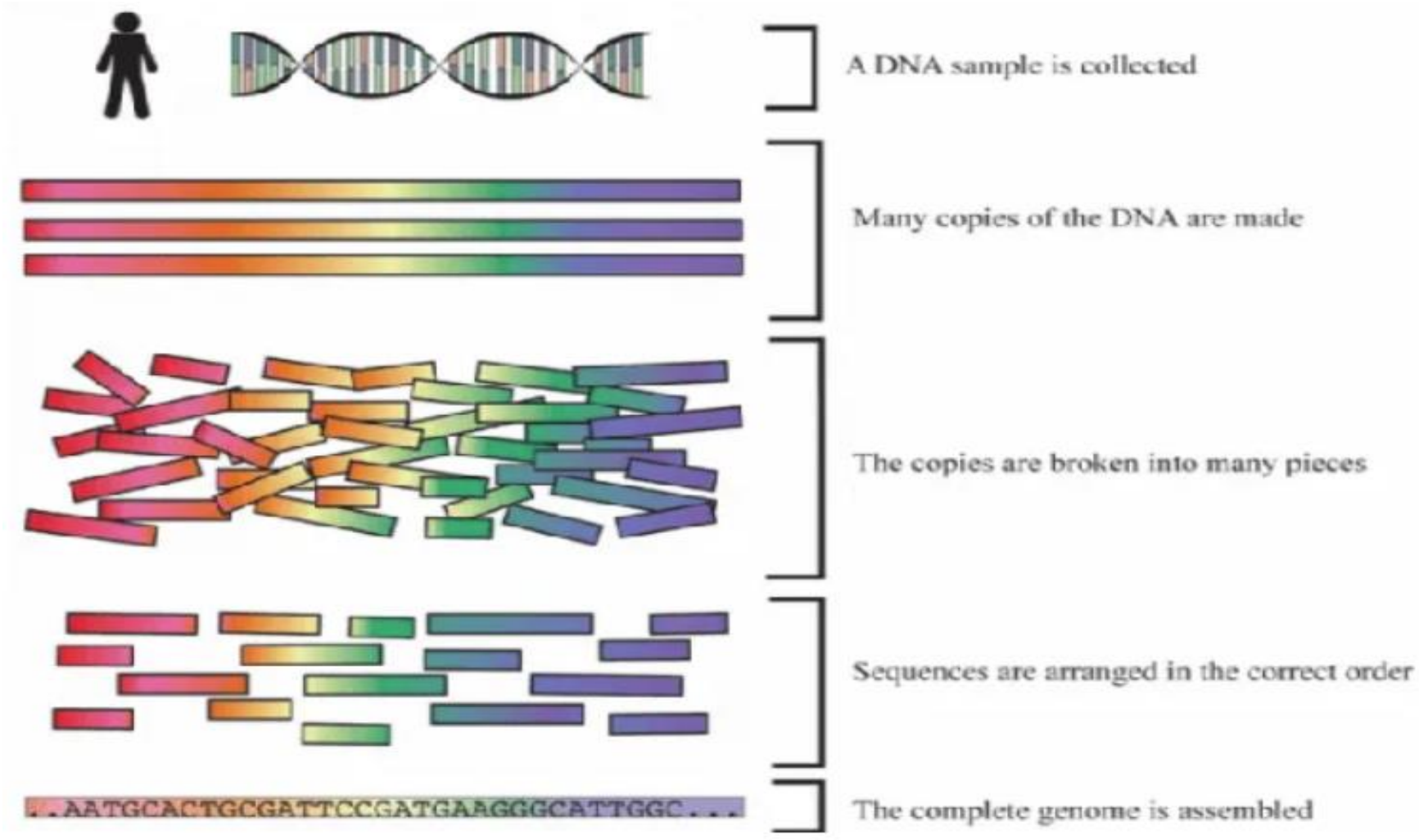
5. When a new sequence is determined, much knowledge is hidden in that sequence, waiting to be uncovered by bioinformatics techniques.

New sequences are arriving too fast for us to handle

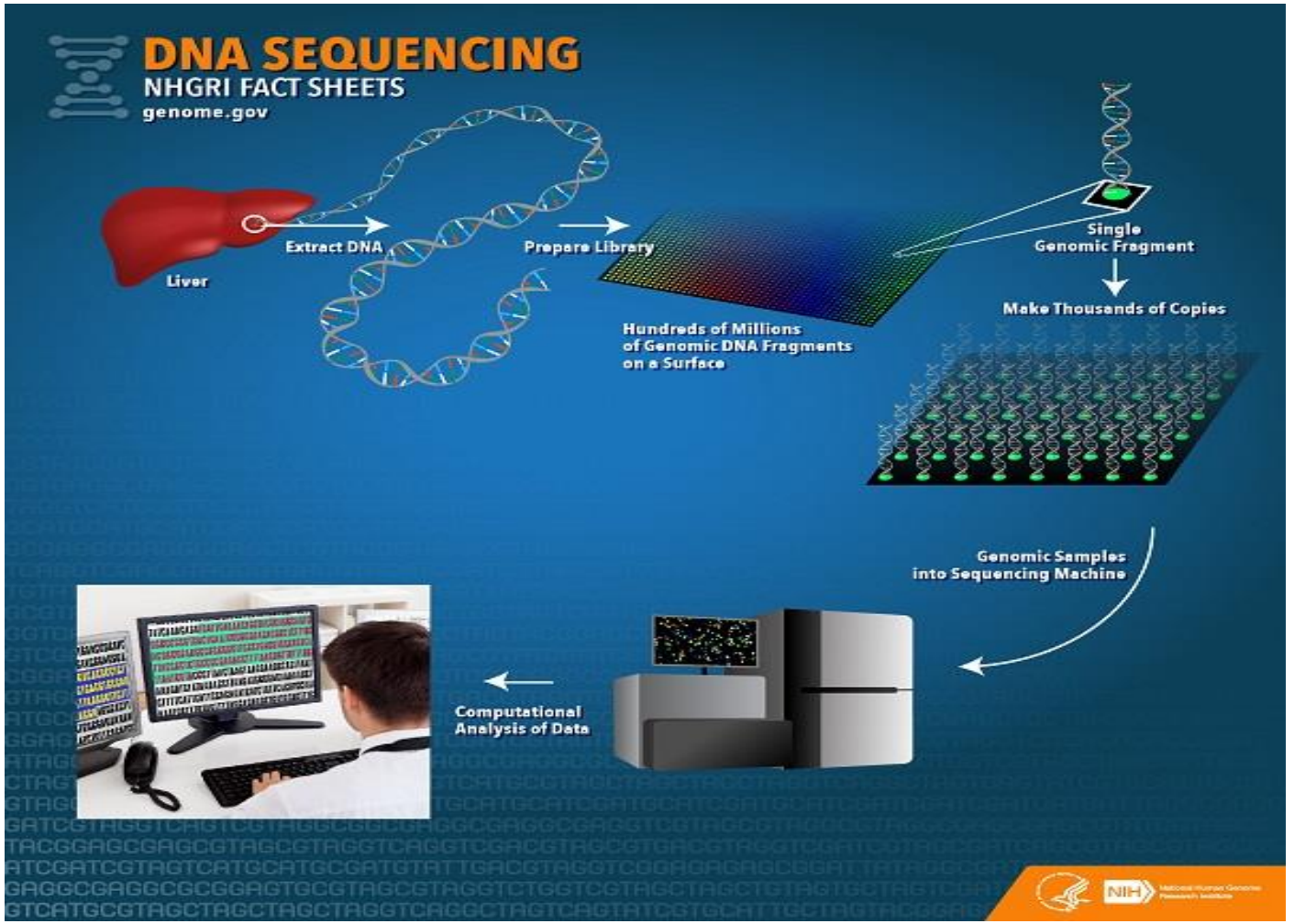
Why do we sequence?



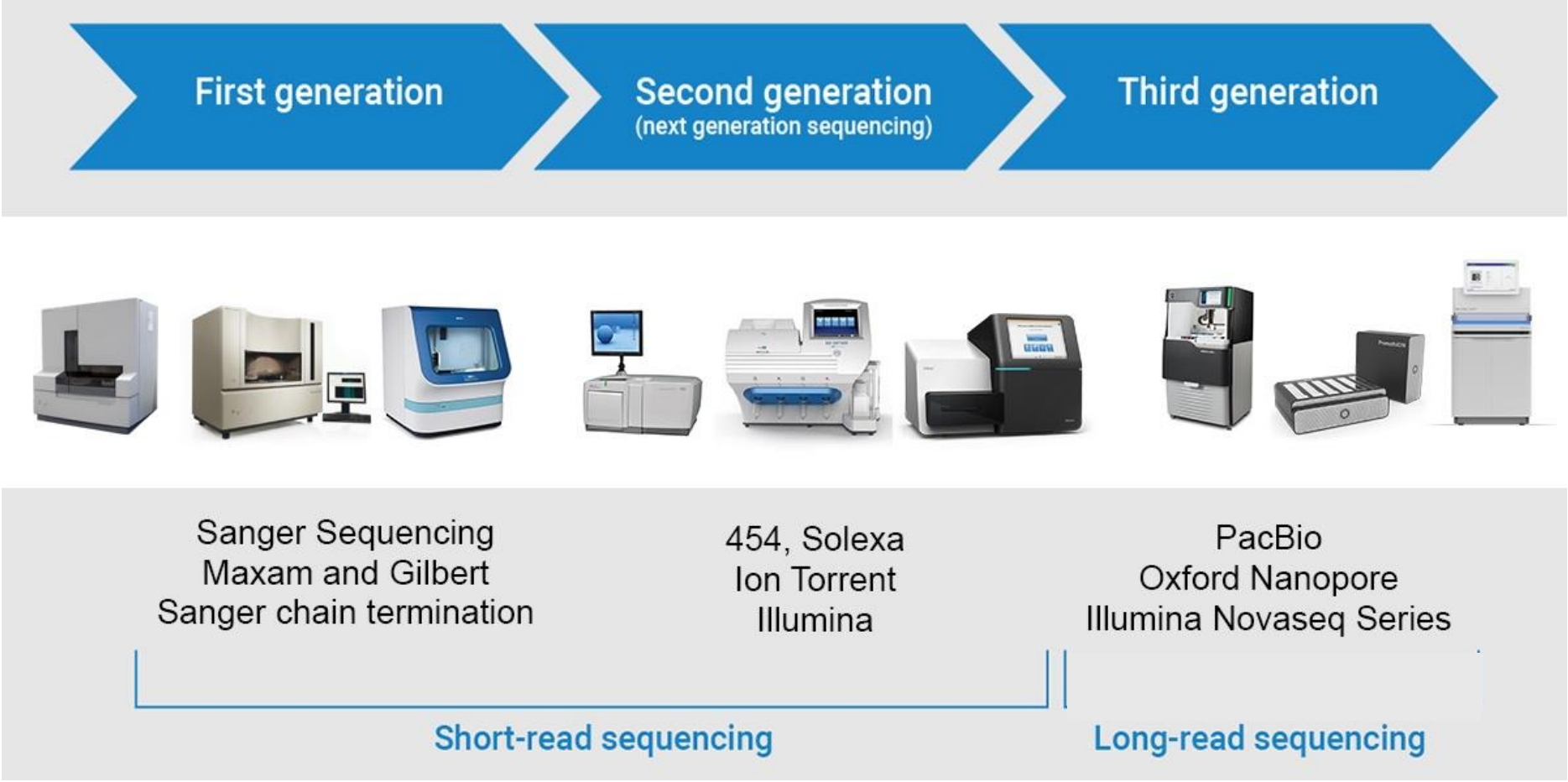
From sample to DNA



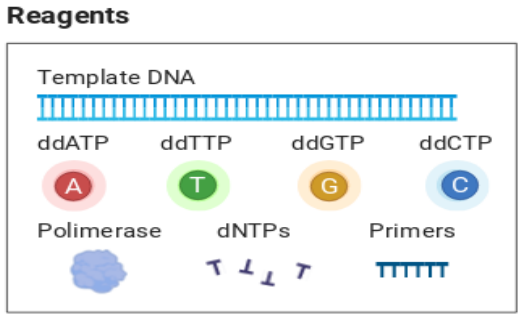
From sample to DNA

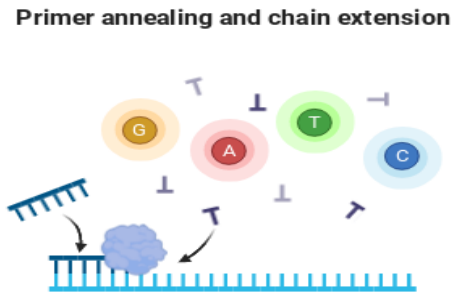


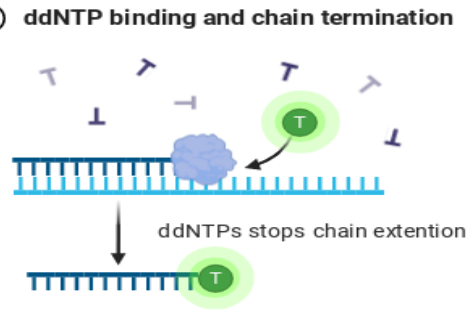
Sequencing technologies

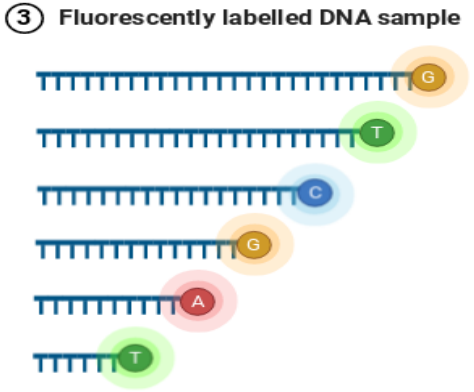


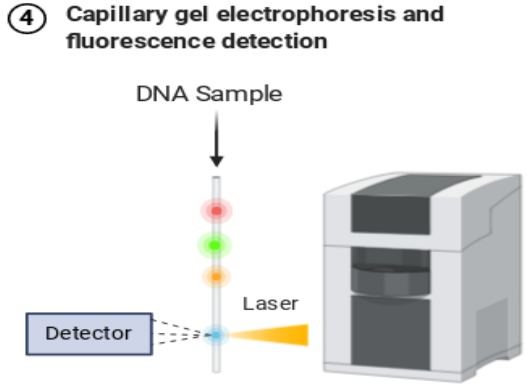
First generation sequencing

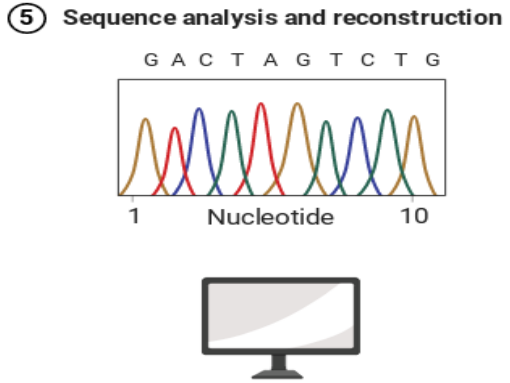




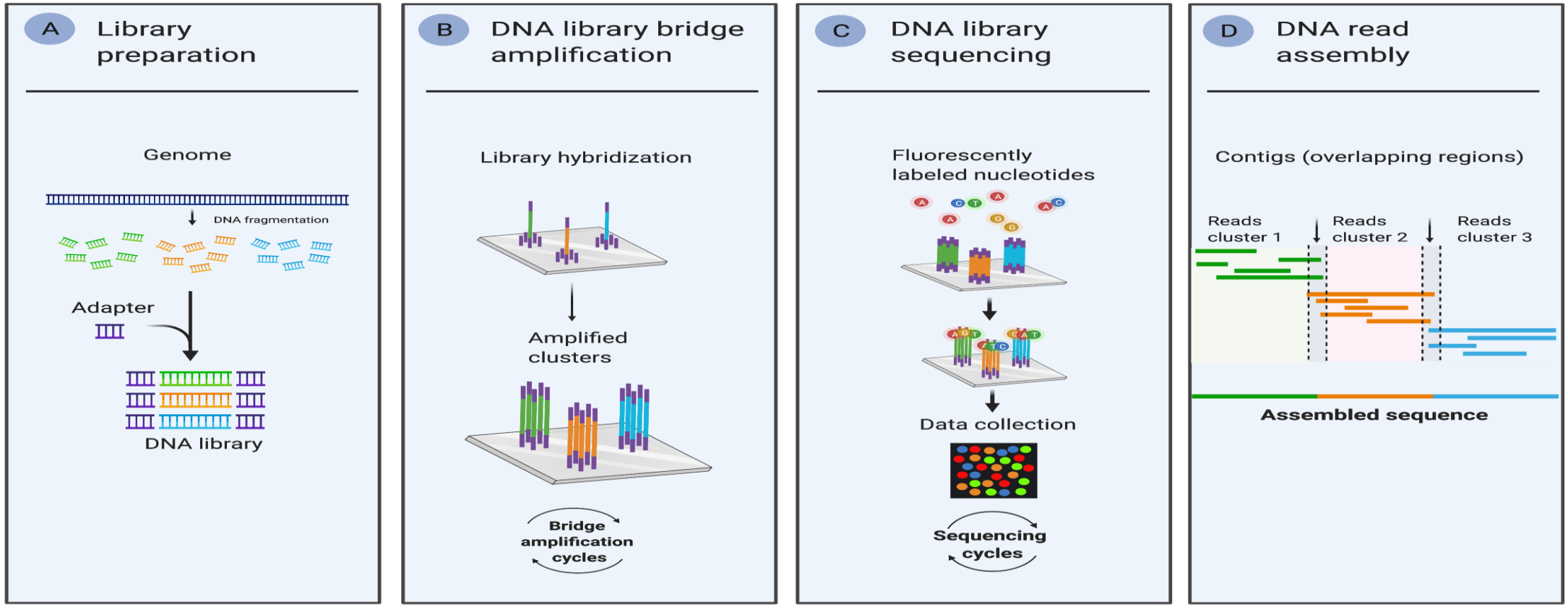








Second generation sequencing

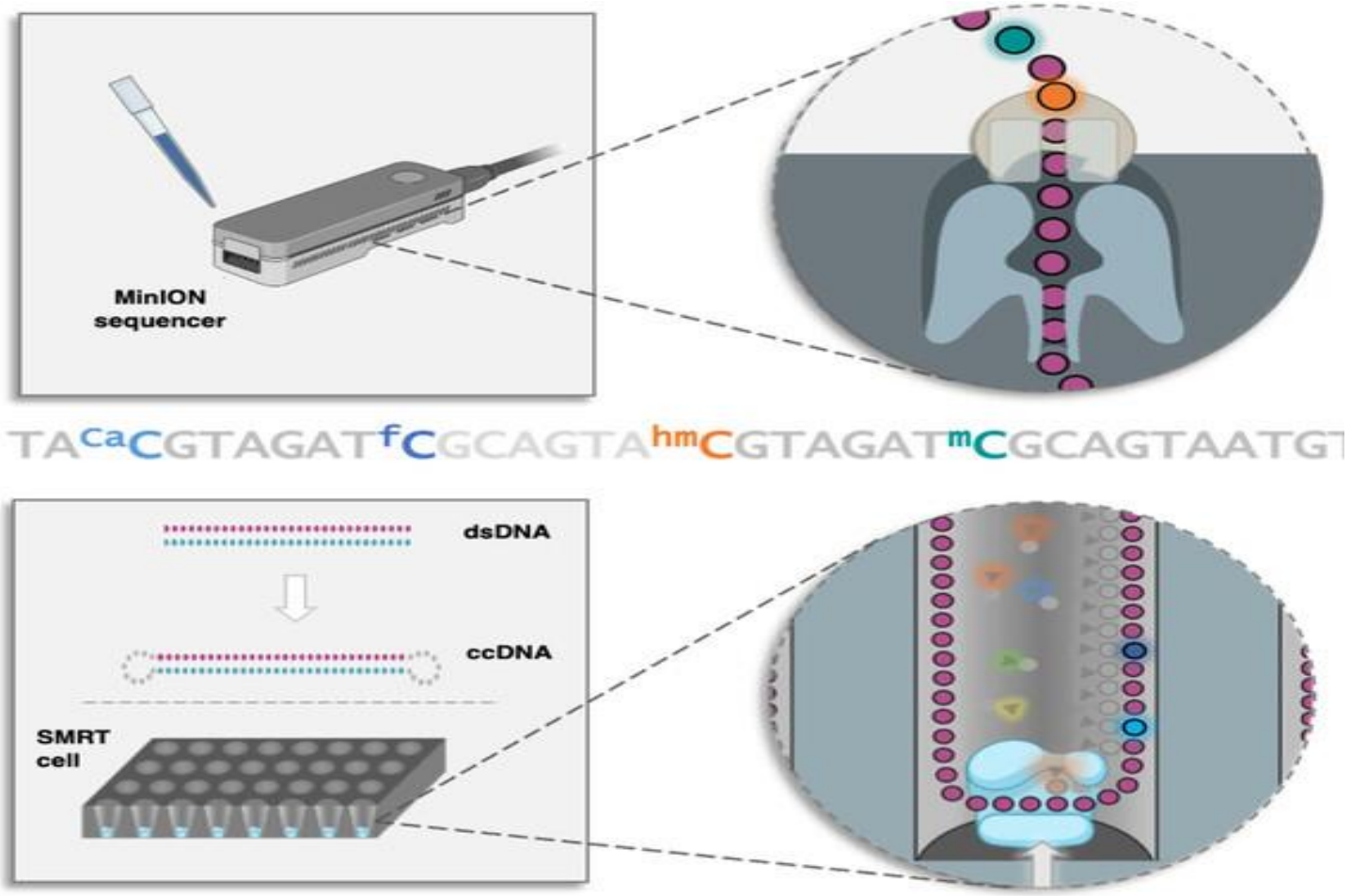


Illumina sequencing

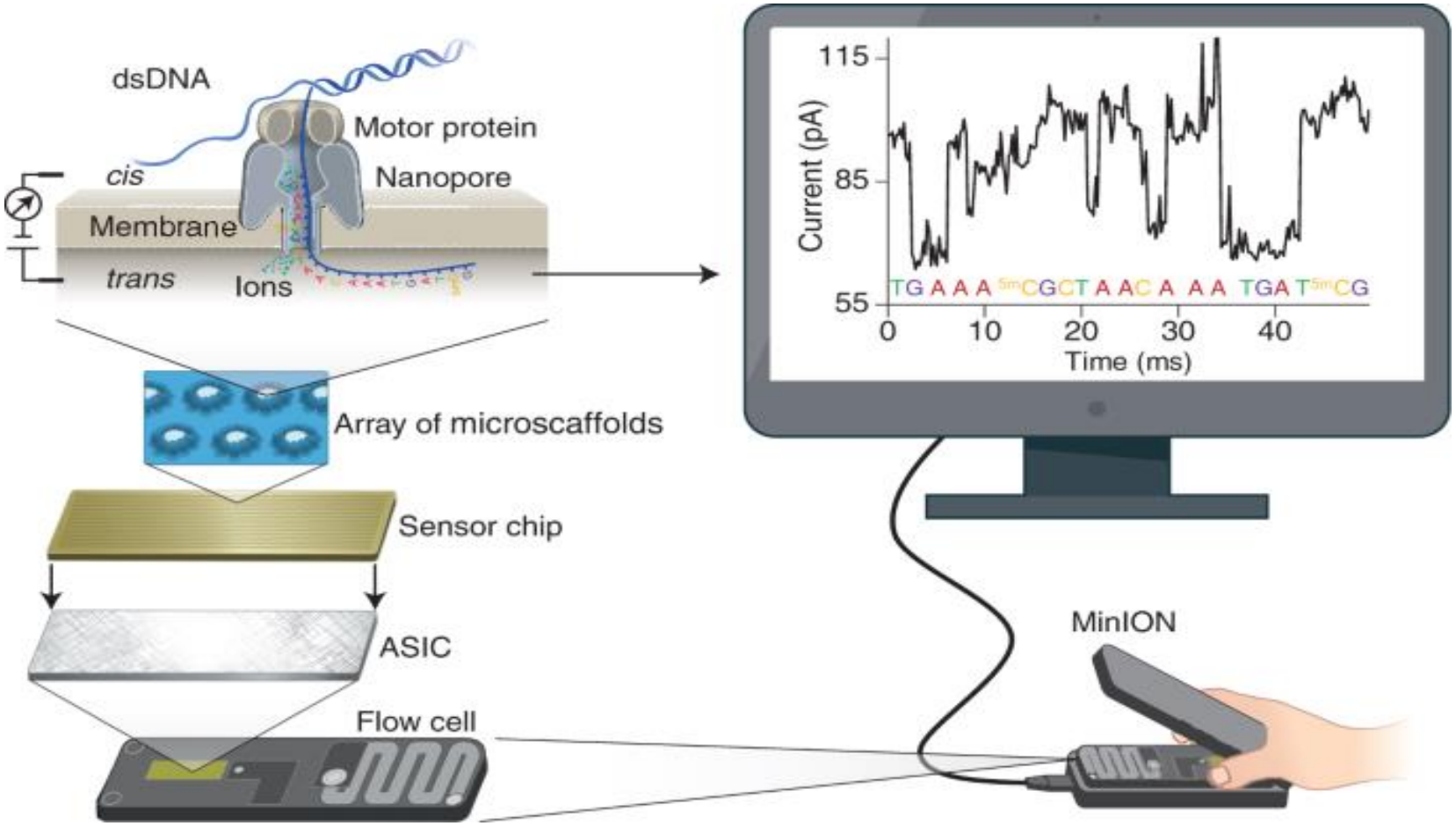
1. MiSeq
2. HiSeq
3. NovaSeq 6000
4. and NovaSeq X

Machine	Size	Data output	Status
MiSeq	Small	Low	Active
HiSeq	Large	High	Mostly retired
NovaSeq 6000	Large	Very high	Widely used
NovaSeq X	Very large	Extremely high	Newest

Third generation sequencing

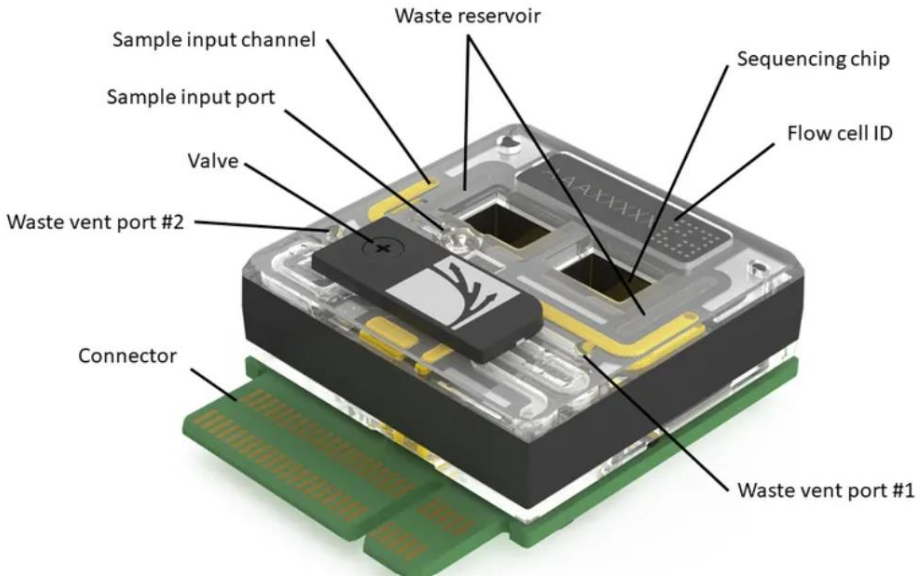


Oxford Nanopore Technologies MinION



- Small **USB stick–sized** sequencer
- Has a **hinged lid** that opens to load the flow cell
- **Needs a laptop/computer** to run (no built-in screen or CPU)

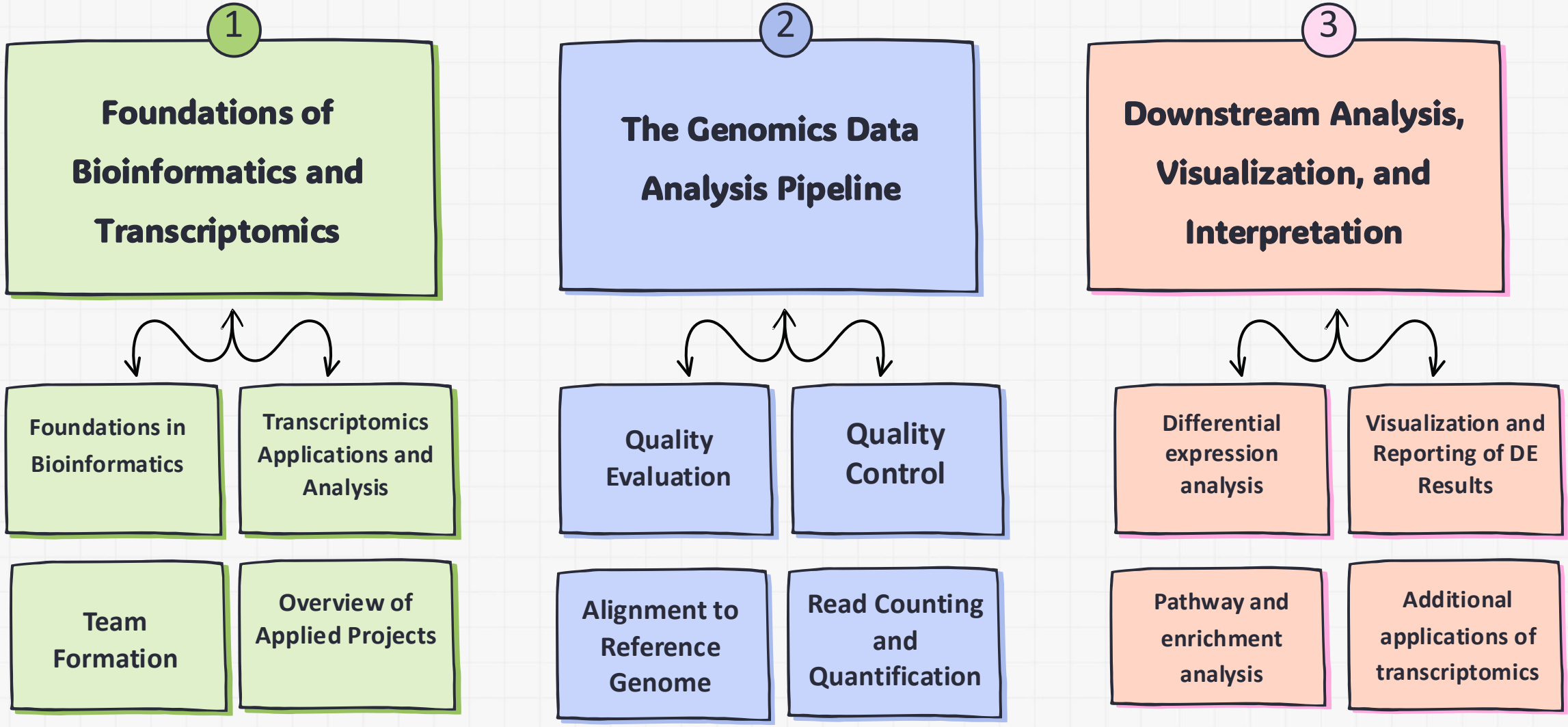
PromethION, the *big* sibling of MinION.



SMRT Cell -PacBio

- https://youtu.be/_lD8JyAbwEo

Overview of Stage 3



Agenda – Day 01

أكاديمية كاوست
KAUST ACADEMY

Morning Session

9:00-9:30	S1: Welcome and Introduction
9:30-10:00	S2: Recap: Foundations in Bioinformatics & Overview of Stage 3
10:00-10:30	S3: Bioinformatics Computing Environments
10:30-12	*L1: Warming-up for Large-scale Analysis & Genomic File Formats*

Afternoon Session

02:00-02:45	S4: Transcriptomics Overview & Applications
02:45-03:30	*L2: Hands-on with Transcriptomics Data*
03:30-04:15	S5: RNA-seq Experimental Design
04:15-05:00	S6: Team Formation & Project Overview

أكاديمية كاوست
KAUST ACADEMY

Bioinformatics Computing Environments

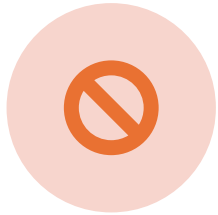
Day 01 – Session 03

Learning Objectives

GET ACQUAINTED WITH LINUX AND THE COMMAND-LINE

LEARN BASH BASIC COMMANDS

BASIC HANDLING OF TEXT



PERMISSIONS

TEXT EDITORS

SYSTEM MONITORING

NETWORKING COMMANDS

BASIC SCRIPTING

Major Operating Systems (OS)

Windows

Unix

Linux

Linux is the preferred OS!

- Stable
  - ideal for handling large-scale biological data analysis
- Secure
- Open-source
  - anyone can access the Linux source code, modify it, and customize it to suit specific tasks
- Performance and scalability
  - Efficient management of system resources, making it ideal for computationally intensive tasks like genome assembly, sequence alignment, and NGS data analysis on high-performance computing (HPC) clusters
- Command-line tools and automation
  - Many bioinformatics tools are command-line based
- Scripting for large-scale data analysis
- Most bioinformatics tools are designed for Linux
- **Multi-user, multi-tasking**
- Package managers & repositories
- Community & support
- Cloud and HPC compatibility
  - The vast majority of cloud computing platforms and HPC systems run Linux

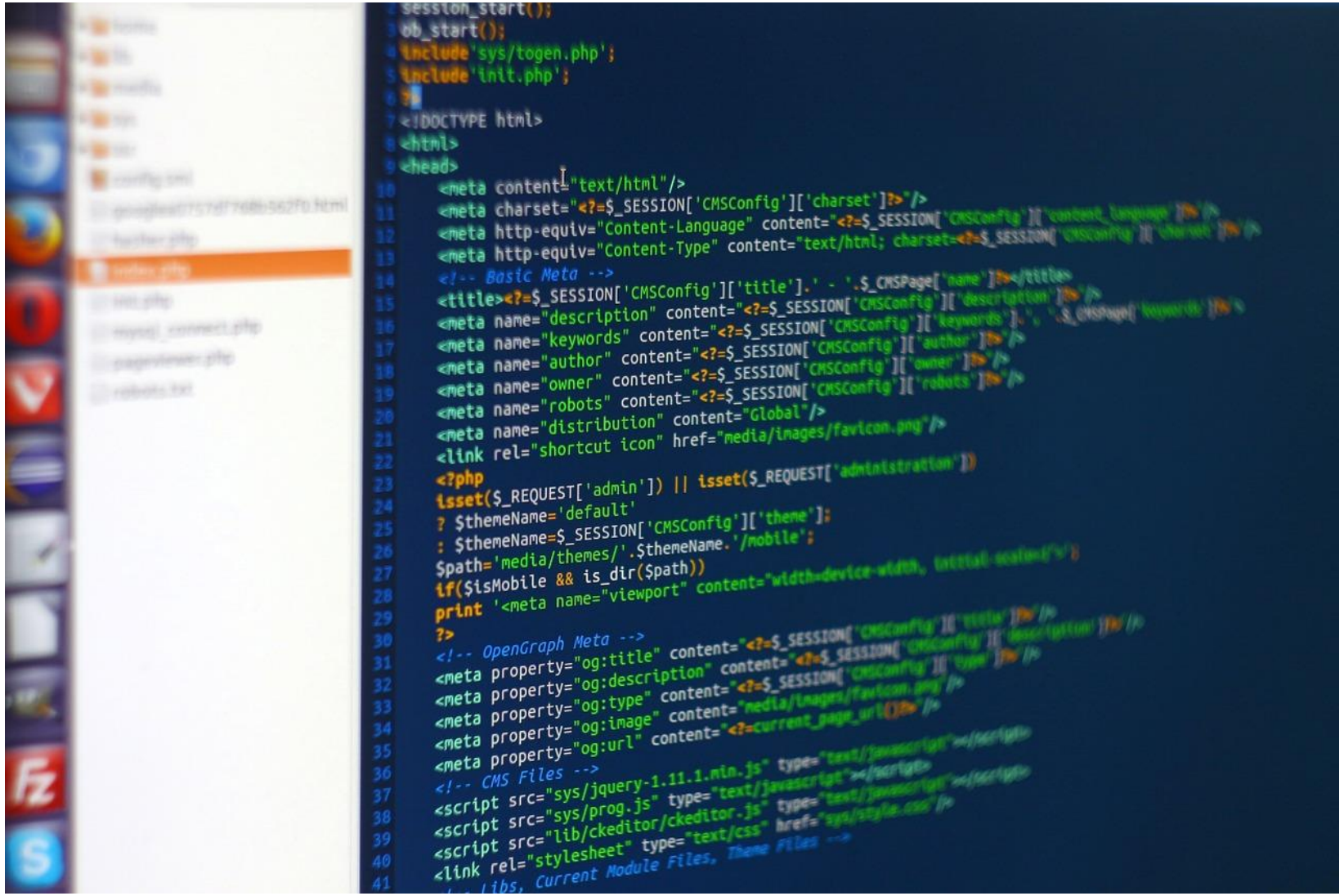
Where to use Linux?

Laptop

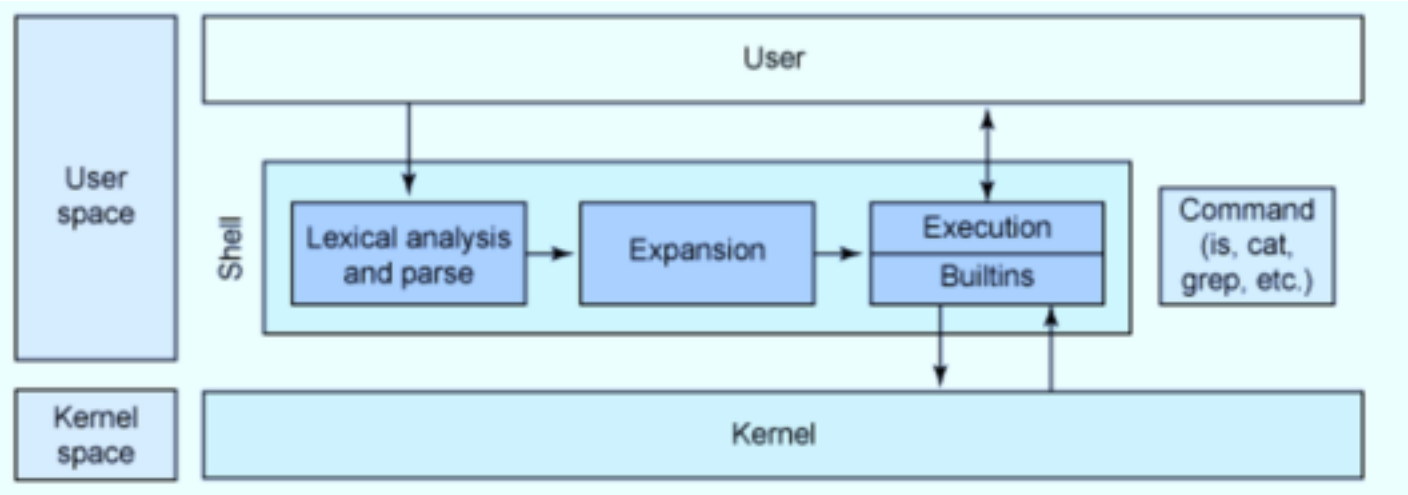
Desktop

Workstation

Cluster / Supercomputer
Onsite – Cloud


session_start();
ob_start();
include'sys/togen.php';
include'init.php';
?>
<!DOCTYPE html>
<html>
<head>
<meta content="text/html"/>
<meta charset="<?=$_SESSION['CMSConfig']['charset']?>"/>
<meta http-equiv="Content-Language" content="<?=$_SESSION['CMSConfig']['content_language']?>"/>
<meta http-equiv="Content-Type" content="text/html; charset=<?=$_SESSION['CMSConfig']['charset']?>"/>
<!-- Basic Meta -->
<title><?=$_SESSION['CMSConfig']['title'].' - '.$_CMSPage['name']?></title>
<meta name="description" content="<?=$_SESSION['CMSConfig']['description']?>"/>
<meta name="keywords" content="<?=$_SESSION['CMSConfig']['keywords'].', '.$_CMSPage['keywords']?>"/>
<meta name="author" content="<?=$_SESSION['CMSConfig']['author']?>"/>
<meta name="owner" content="<?=$_SESSION['CMSConfig']['owner']?>"/>
<meta name="robots" content="<?=$_SESSION['CMSConfig']['robots']?>"/>
<meta name="distribution" content="Global"/>
<link rel="shortcut icon" href="media/images/favicon.png"/>
<?php
isset($_REQUEST['admin']) || isset($_REQUEST['administration'])
? $themeName='default'
: $themeName=$_SESSION['CMSConfig']['theme'];
$path='media/themes/'.$themeName.'/mobile';
if($isMobile && is_dir($path))
print '<meta name="viewport" content="width=device-width, initial-scale=1">';
?>
<!-- OpenGraph Meta -->
<meta property="og:title" content="<?=$_SESSION['CMSConfig']['title']?>"/>
<meta property="og:description" content="<?=$_SESSION['CMSConfig']['description']?>"/>
<meta property="og:type" content="<?=$_SESSION['CMSConfig']['type']?>"/>
<meta property="og:image" content="media/images/favicon.png"/>
<meta property="og:url" content="<?=current_page_url()?>"/>
<!-- CMS Files -->
<script src="sys/jquery-1.11.1.min.js" type="text/javascript"></script>
<script src="sys/prog.js" type="text/javascript"></script>
<script src="lib/ckeditor/ckeditor.js" type="text/javascript"></script>
<link rel="stylesheet" type="text/css" href="sys/style.css"/>
<!-- Libs, Current Module Files, Theme Files -->

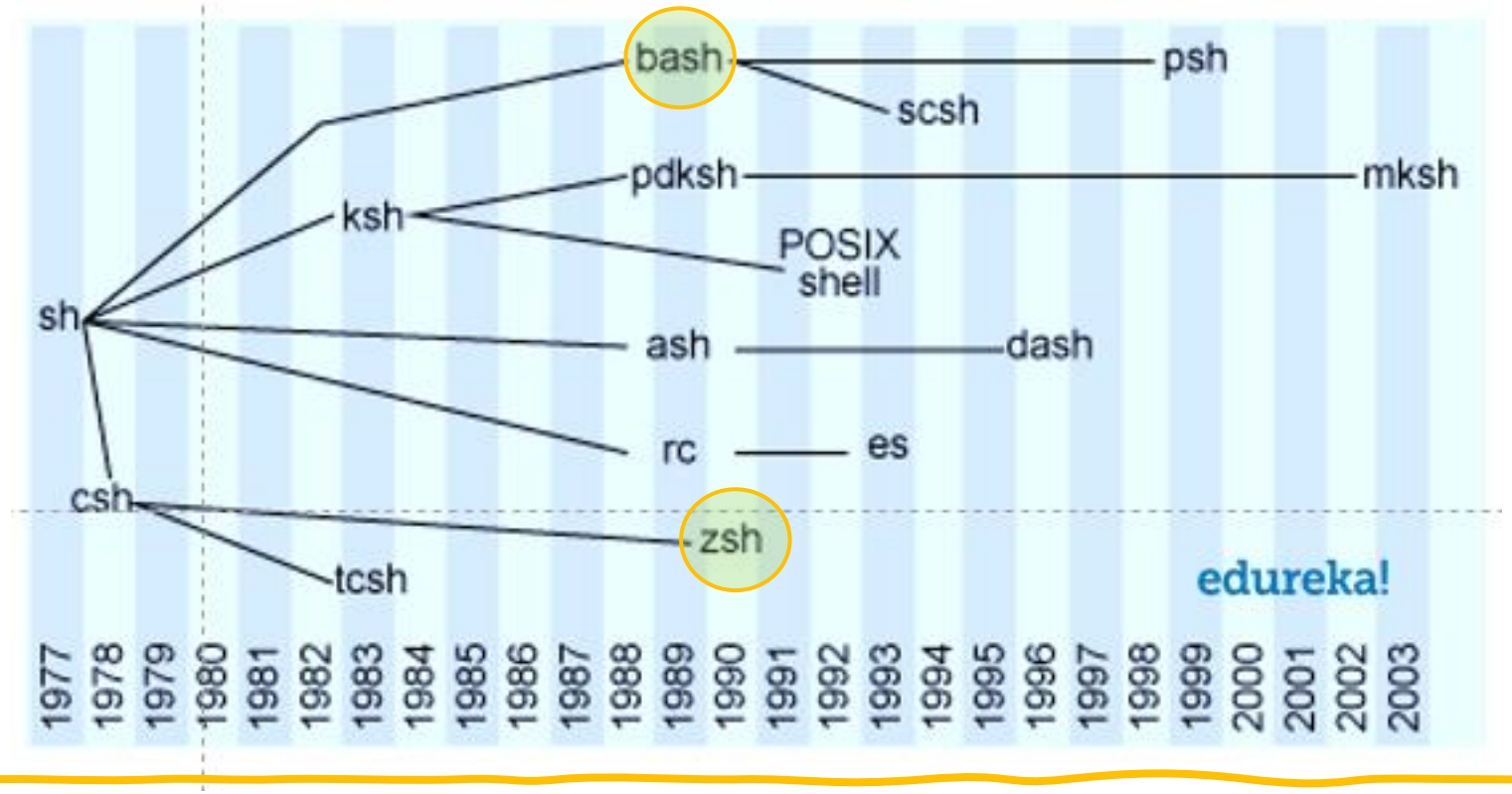
The Shell: basic architecture



Shell is an interactive environment which provides an interface to an Operating System. It gathers input from you in a sequence to implement a specific use model.

https://www.edureka.co/blog/types-of-shells-in-linux/

The Shell: types & a bit of shell phylogeny



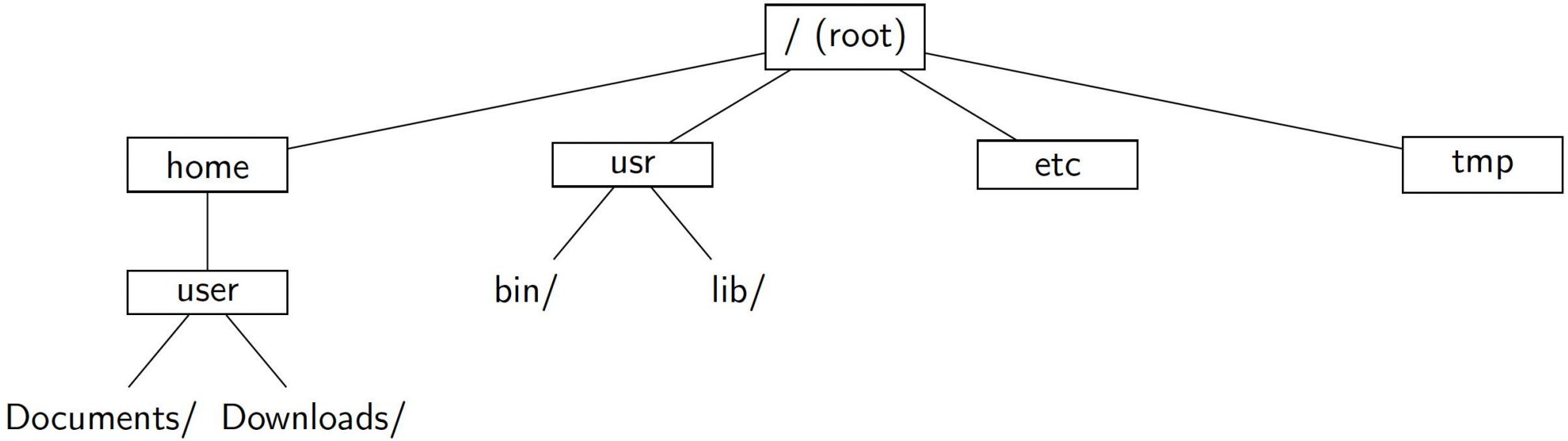
Bash: Bourne Again Shell

https://www.edureka.co/blog/types-of-shells-in-linux/

shell: the basics

- The Linux Home Directory
  - This a directory that serves as the default location for a user's personal files.
  - Every user on a Linux system will have a unique home directory
  - The home directory is typically represented by the tilde symbol (~) in the Linux command line interface.

shell: the basics



Key Concepts

- / = root directory
- ~ = home directory
- . = current directory
- .. = parent directory

Paths

- Absolute: `/home/user/file.txt`
- Relative: `./data/file.txt`

shell: the basics

- Dot (.)
  - represents the current directory in the filesystem.
- Dot-dot (..)
  - represents one level above the current directory
- Forward slash (/)
  - Represents the "root" of the filesystem.
  - Every directory/file in the Linux filesystem is nested under the root / directory.
- Tilde (~)
  - Represents the home directory of the currently logged in user.

bash: basic commands

- **pwd**: print working directory
- **ls**: list directory content
- **cd**: change directory
  - cd /path/to/directory ← navigate to /path/to/directory
  - cd ~/ ← Go to home directory
  - cd ../ ← Move one level up
  - cd ../.. ← Move two levels up
- **mkdir**: make directory
  - mkdir directory_name ← create a folder named directory_name
  - mkdir dir1 dir2 dir3 ← create three folders at once
  - mkdir –p dir1/dir2/dir3 ← create nested directories

bash: basic commands

- **cat**: Concatenate & display files
  - cat file1.txt ← display content of file1.txt
  - cat file1.txt file2.txt > file3.txt ← concatenate file1 and 2 into file3
- **cp**: copy files and directories
  - cp file1.txt file2.txt ← make a copy of file1.txt and name it file2.txt
  - cp file1 /path/to/other/location/file2.txt
  - cp –r dir1 dir2 ←create another recursive copy of dir1
- **mv**: move files and directories
  - Mv file1.txt /path/to/other/location/. ← move file to a different location
  - Mv file1.txt file2.txt ← rename file1.txt to file2.txt in the same location

bash: basic commands

- **rmdir**: remove an empty directory
  - rmdir ./dir ← remove directory dir which must be empty
- **rm**: remove files and directories
  - rm file1.txt file2.txt ← remove said files…IRREVESIBLE
  - rm –r dir ← recursively delete folder dir and all of its content
- **man**: display the user manual of any command
- **find**: search for files and directories
  - find /my/path –name “myFile.txt”
- **history**: View command history

bash: text processing

- **head**: display first part of a text file
  - head file1.txt ← display first 10 lines
  - head –20 file1.txt ← display first 20 lines
- **tail**: display last part of a text file
  - tail file1.txt ← display last 10 lines
  - tail –n20 file1.txt ← display last 20 lines
- **more**: view (not modify) a text file on screen
  - More file.txt ← view said file; exit by hitting the **q** button on keyboard
- **less**: view (not modify) a text file on screen
  - Less file.txt ← view said file; exit by hitting the **q** button on keyboard
- **wc**: word count
  - wc myfile.txt → lines words characters

bash: advanced text processing

- **grep**: global regular expression print
  - To search for a pattern in a file, use grep 'pattern' filename
- **awk**: Aho, Weinberger, and Kernighan
  - handling text files and used for data extraction and reporting
- **sed**: stream editor
  - It's a powerful tool for making quick edits to files or streams of data.
- **cut**: it’s just cut
  - remove sections from each line of files
- **sort**:
  - sort lines of text files
- **uniq**:
  - filter out adjacent duplicate lines from text files or standard input
- **column**:
  - format text into aligned columns
- **Pipes (|)**:
  - connect the standard output (stdout) of one command to the standard input (stdin) of another, enabling real-time data flow without temporary files.
  - command1 | command2

bash: networking

- **ssh**: secure shell; connect to a remote machine securely.
  - ssh myUser@remote.machine.kaust.edu.sa
- **wget**: world wide web get, download files from the web
  - wget http://example.com/file.txt
- **curl**: client URL; transfer data from or to a server using various protocols like HTTP, HTTPS, FTP, and more
  - curl http://example.com/file.txt
- **scp**: secure copy; securely copy files between hosts on a network.
  - scp file.txt user@example.com:/home/user/
- **rsync**: remote synchronisation; efficiently transfer and synchronize files across computer systems
  - rsync -avz /local/dir/ user@example.com:/remote/dir/

bash: System and Process Monitoring

- **df**: Show disk space usage.
- **du**: Estimate file space usage.
- **free**: Display memory usage.
- **ps**: List running processes.
- **top**: Monitor system processes in real time.
- **kill**: Terminate a process by ID (PID).

bash: File Permissions and Ownership

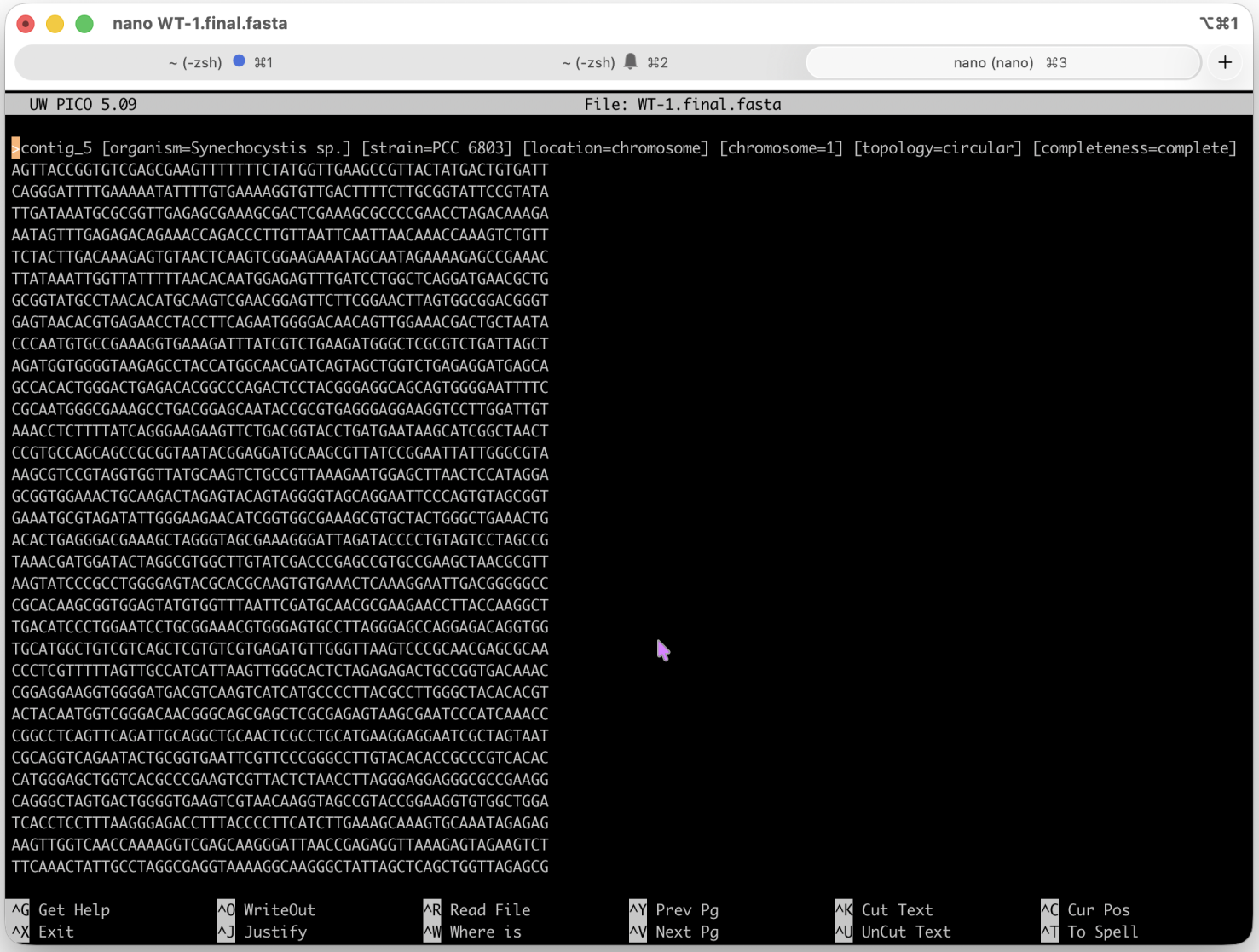
- File permissions and ownership are crucial for managing access to files and directories.
- Each file has an **owner**, a **group**, and a set of permissions that determine who can read, write, or execute the file.
- myFile.txt rwxrwxrwx
- File permissions can also be represented using numbers
  - myFile.txt 777

- `0`: No permission
- `1`: Execute permission
- `2`: Write permission
- `3`: Write and execute permissions
- `4`: Read permission
- `5`: Read and execute permissions
- `6`: Read and write permissions
- `7`: Read, write, and execute permissions

- **chmod**: Change file permissions
  - chmod g+rx myfile.txt
  - chmod 755 myfile.txt
- **chown**: Change file ownership
  - chown -R user:group /path/to/directory
- **chgrp**: Change group ownership
  - chgrp -R group /path/to/directory

bash: editing files

- nano:
  - a small editor for on the terminal
  - nano myfile.txt



bash: good to know

- &
- Control c
- nohup

bash: scripting

- Basic rules in bash scripting
  - Comments: Comments start with a # and Bash ignores them.
  - Command Order: Commands run one after the other, from top to bottom.
  - Semicolons: Use ; to run multiple commands on the same line.

Example: Simple Bash Script

```
#!/bin/bash
# This script prints a greeting message
echo "Hello, World!"
```

bash: scripting - example

```
#!/bin/bash
#This script will print some text
# redirect this input into a text file
# then print some stats about this file content
myName=“salim”
echo “hello, my name is ${myName}”;
echo  “hello 1” > myFile.txt
echo  "hello 2" >> myFile.txt
echo  "hello 3" >> myFile.txt
wc myFile.txt
```

QUIZ

أكاديمية كاوست
KAUST ACADEMY

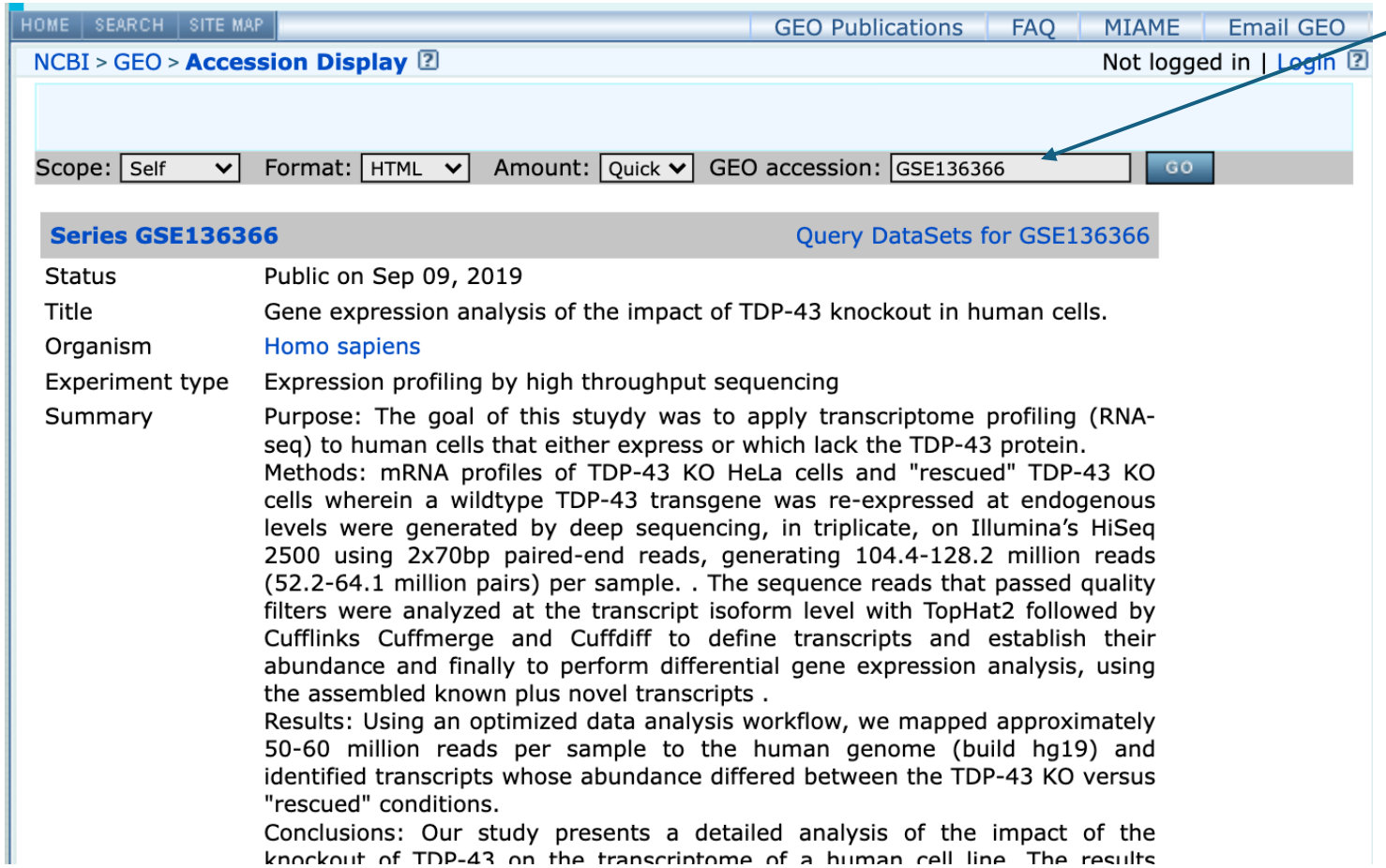
Hands-on Session: Computing Environment & Genomic Data Sets

Day 01 – Lab 01

Tasks

- Set up a structured project directory for data analysis
- Explore NCBI for raw sequencing datasets
- Explore ENSEMBL for reference genome files
  - Download and explore reference genome and annotation files
- Learn common genomic file formats (FASTA, GTF)

NCBI



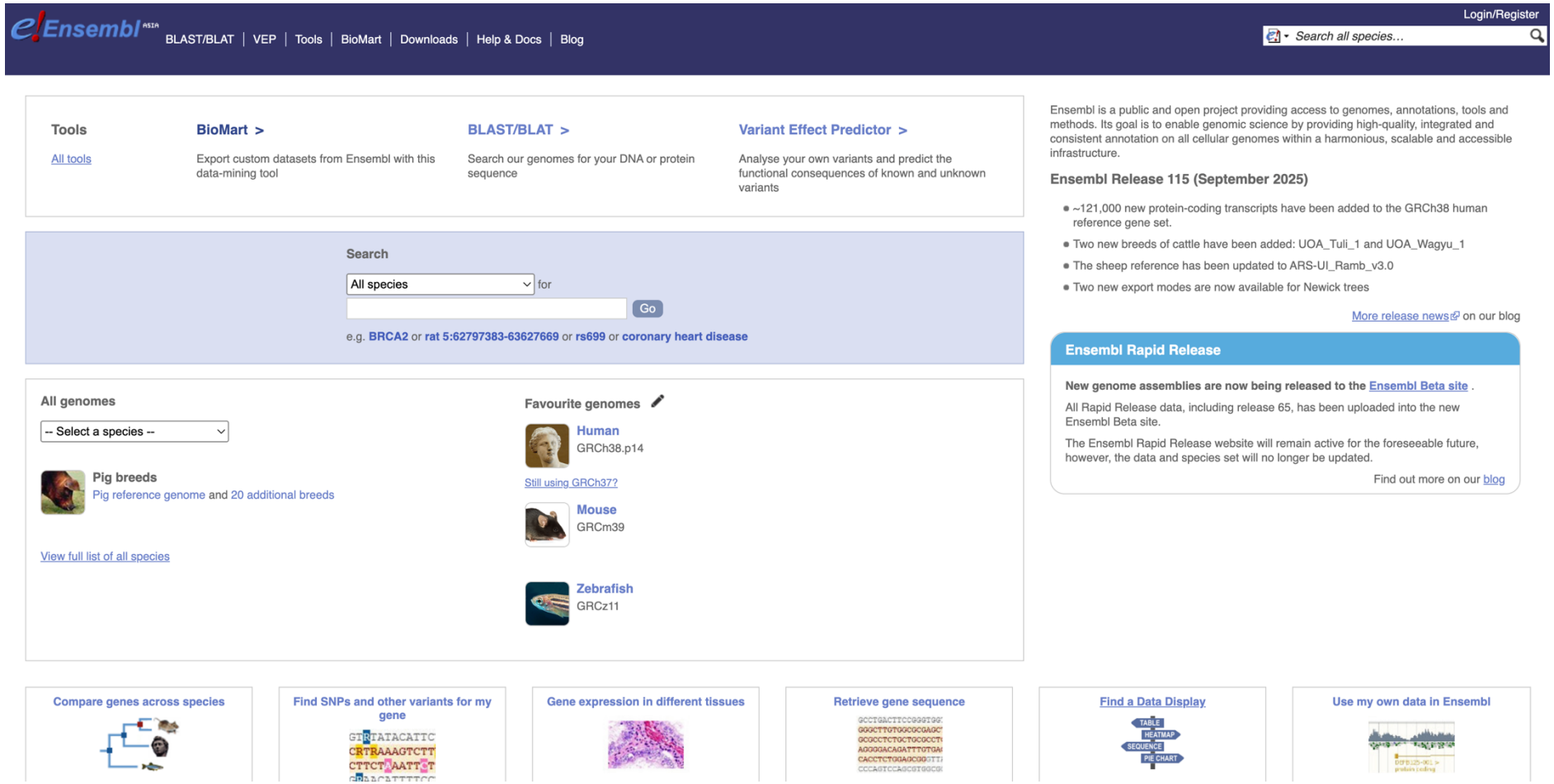
Submission date	Aug 26, 2019
Last update date	Oct 08, 2019
Contact name	Shawn Michael Ferguson
E-mail(s)	shawn.ferguson@yale.edu
Phone	2037375505
Organization name	Yale School of Medicine
Department	Cell Biology
Lab	Ferguson
Street address	295 Congress Ave
City	New Haven
State/province	CT
ZIP/Postal code	06510
Country	USA

Platforms (1)	GPL16791 Illumina HiSeq 2500 (Homo sapiens)
Samples (6)	GSM4047464 TDP-43 KO rep. 1
Less...	GSM4047465 TDP-43 KO rep. 2
	GSM4047466 TDP-43 KO rep. 3
	GSM4047467 TDP-43 Rescue rep. 1
	GSM4047468 TDP-43 Rescue rep. 2
	GSM4047469 TDP-43 Rescue rep. 3

Relations

BioProject	PRJNA562297
SRA	SRP219885

Ensembl data Access



Agenda – Day 01

Morning Session

9:00-9:30	S1: Welcome and Introduction
9:30-10:00	S2: Recap: Foundations in Bioinformatics & Overview of Stage 3
10:00-10:30	S3: Bioinformatics Computing Environments
10:30-12	*L1: Warming-up for Large-scale Analysis & Genomic File Formats*

Afternoon Session

02:00-02:45	S4: Transcriptomics Overview & Applications
02:45-03:30	*L2: Hands-on with Transcriptomics Data*
03:30-04:15	S5: RNA-seq Experimental Design
04:15-05:00	S6: Team Formation & Project Overview

أكاديمية كاوست
KAUST ACADEMY

Transcriptomics Overview & Applications

Day 01 – Session 04

Key topics

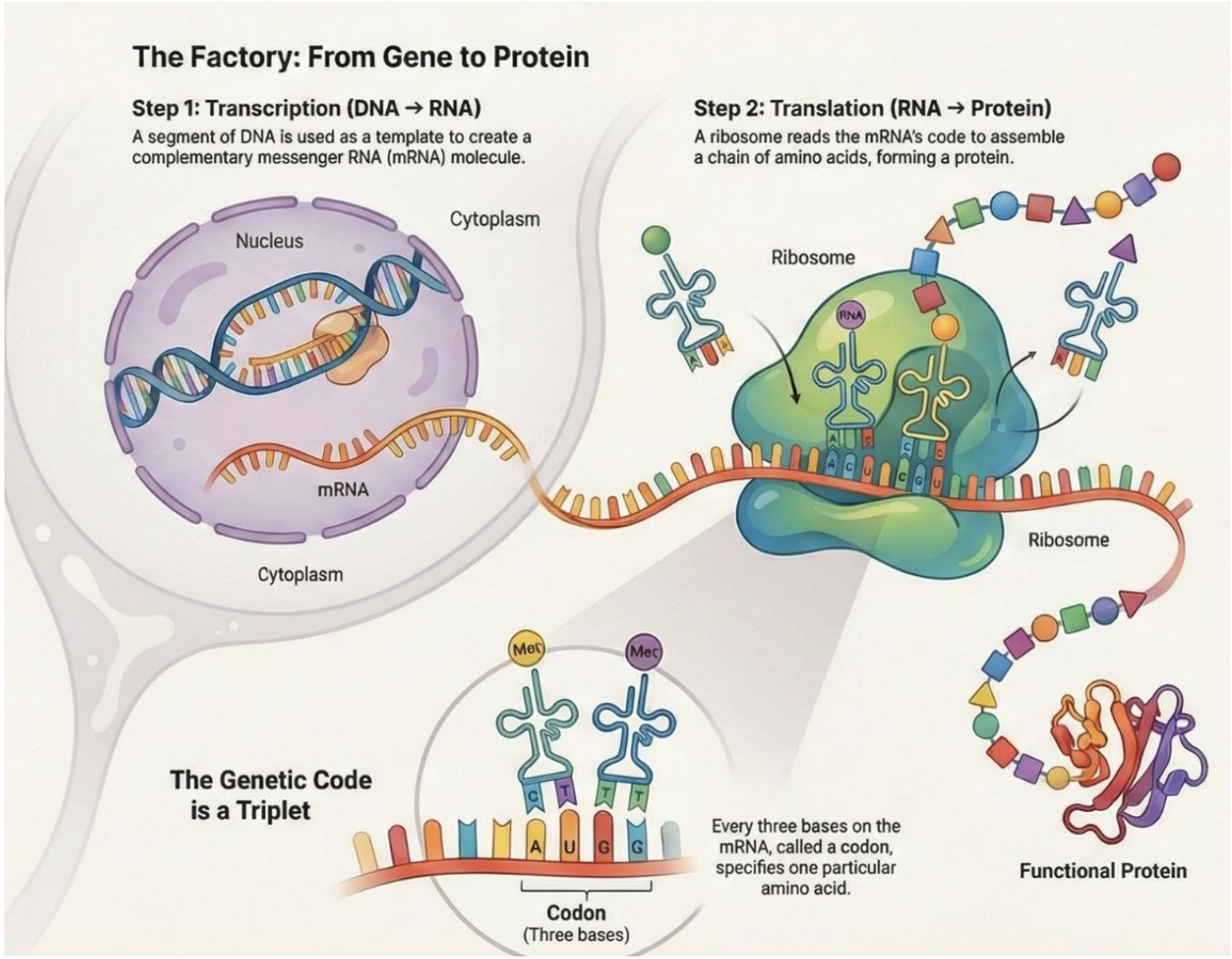
- Learn about different RNA-seq protocols and applications
- Identify the importance, potentials and limitations of RNA-Seq analysis
- Explore the sequencing platforms that perform bulk RNA-seq
- Identify the most common sources of error in RNA-Seq experiments
- Discover the most important aspects to consider when designing an RNA-Seq experiment

Content

- Central dogma
- RNA-seq for Transcriptomics
- Gene structure: exon, intron, UTR, etc
- RNA types: protein-coding, ncRNA, miRNA, etc
- Types of RNA-seq experiments: mRNA, rRNA, single cell, spatial
- Technologies for RNA-seq: short-read vs long read
- Aims for RNA-seq: gene differential expression, novel transcript, isoforms and gene fusions, cell type annotation and clustering, etc.
- RNA-seq protocol (NGS): strand-specific, read length,
- Experimental design: sample type, replicates, depth, library prep

Central dogma

Genotype to Phenotype



DNA → RNA (Transcription)

✞ DNA serves as template to synthesize messenger RNA (mRNA)

RNA → Protein (Translation)

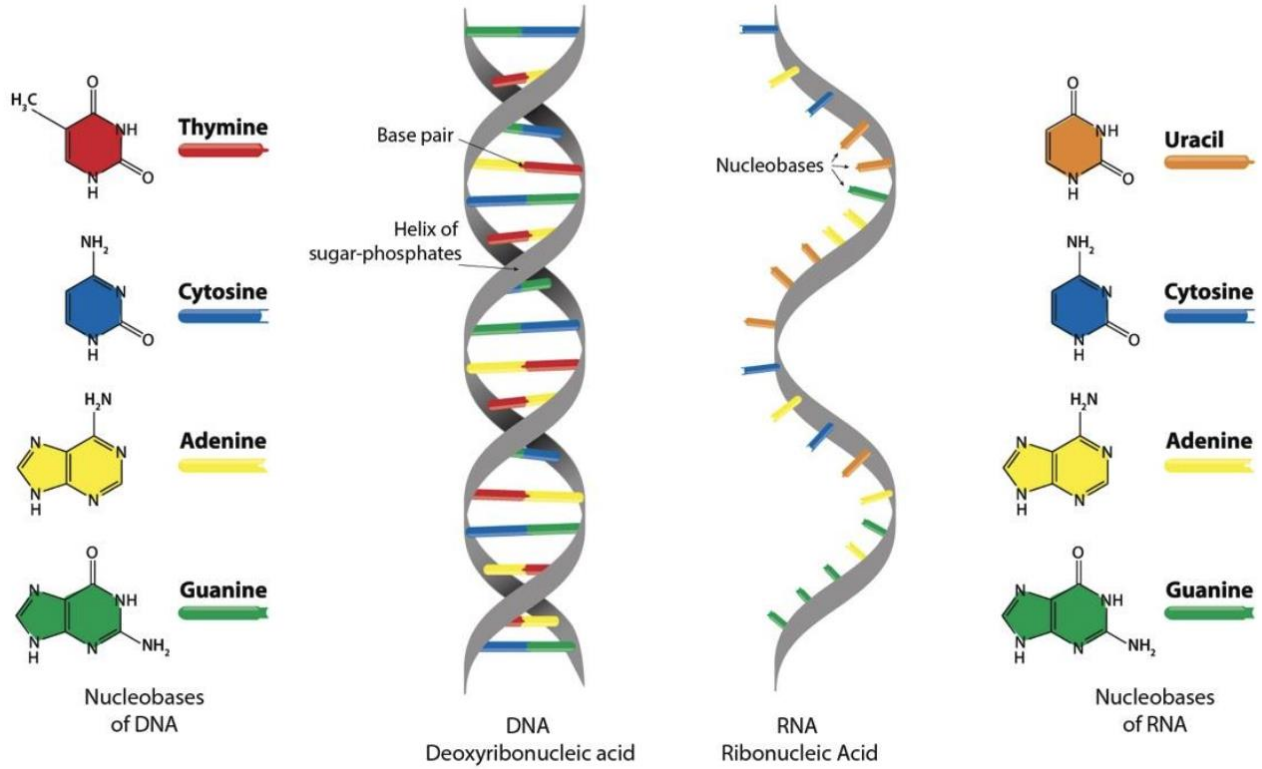
✞ Ribosomes translate mRNAs into proteins in the cytoplasm

✞ Transfer RNA (tRNA) and ribosomal RNA (rRNA) help decode the mRNA sequence into amino acids, forming a polypeptide chain

Protein → Function

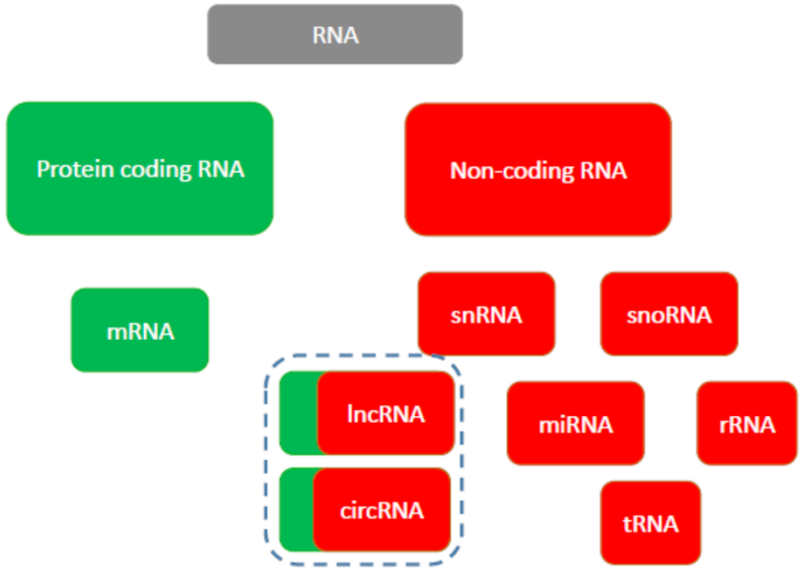
The synthesized protein folds into its functional three-dimensional shape and performs biological functions (e.g., enzymes, structural proteins, signaling molecules)

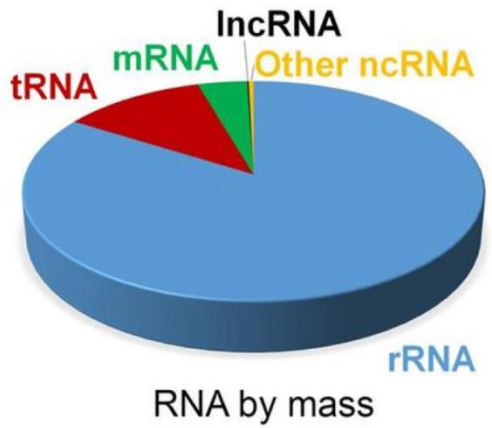
DNA (DeoxyRibonucleic acid) vs RNA (Ribonucleic Acid)



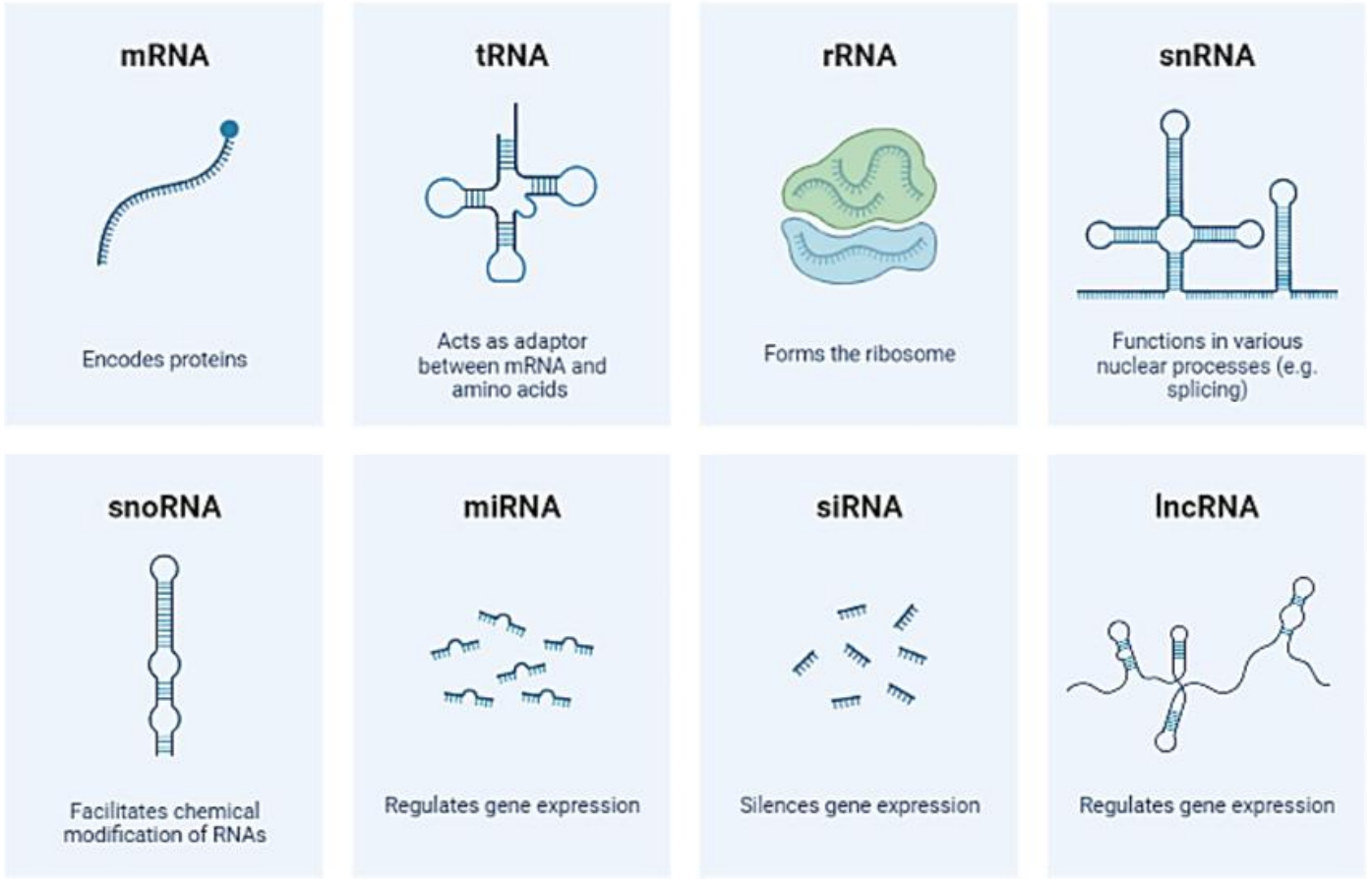
- ✓ DNA is a double-stranded, stable molecule that stores genetic information
- ✓ RNA is a single-stranded, dynamic molecule involved in gene expression, regulation, and protein synthesis

Different types of RNA

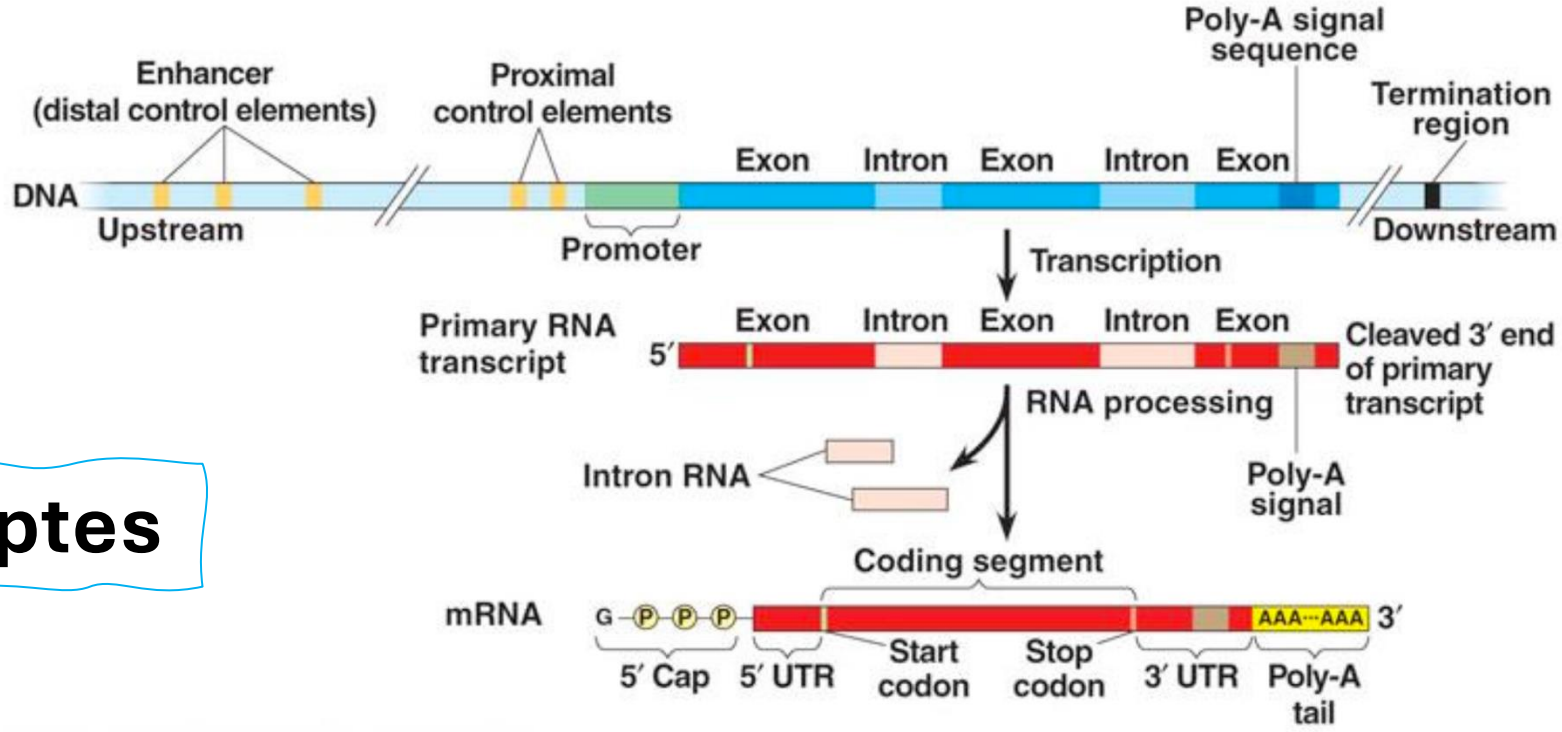




RNA by mass



RNA Transcription



Eukaryoptes

Source: rnabio.org

Transcriptome

Study of RNA molecules in a cell

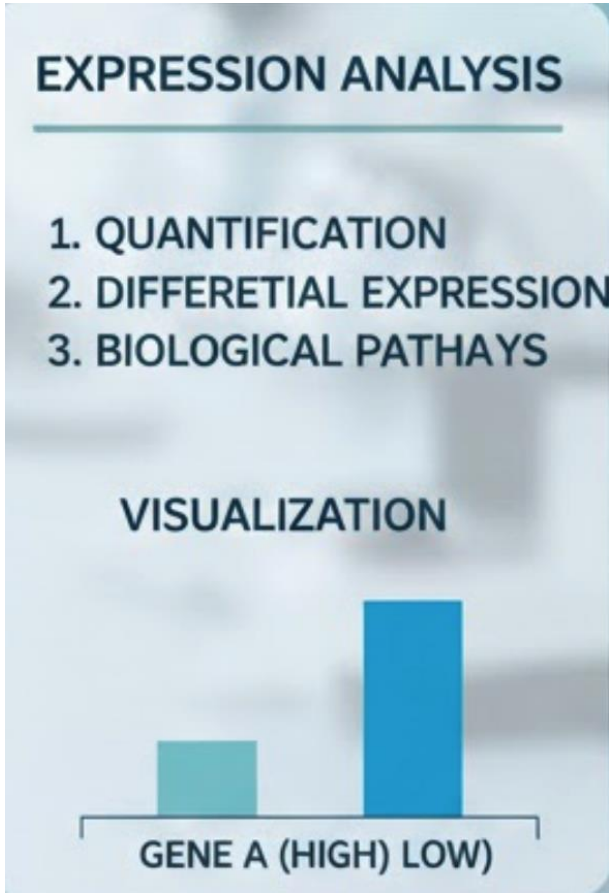
the complete set of RNA transcripts produced by the genome at any given time, providing a snapshot of gene expression in cells or tissues

Applications of Transcriptomics

- Gene expression profiling
- Differntial expression analysis
- Alternative transcripts & gene fusions
- Novel genes & transcripts
- Splice junctions
- Genome annotation
- Identify cell types

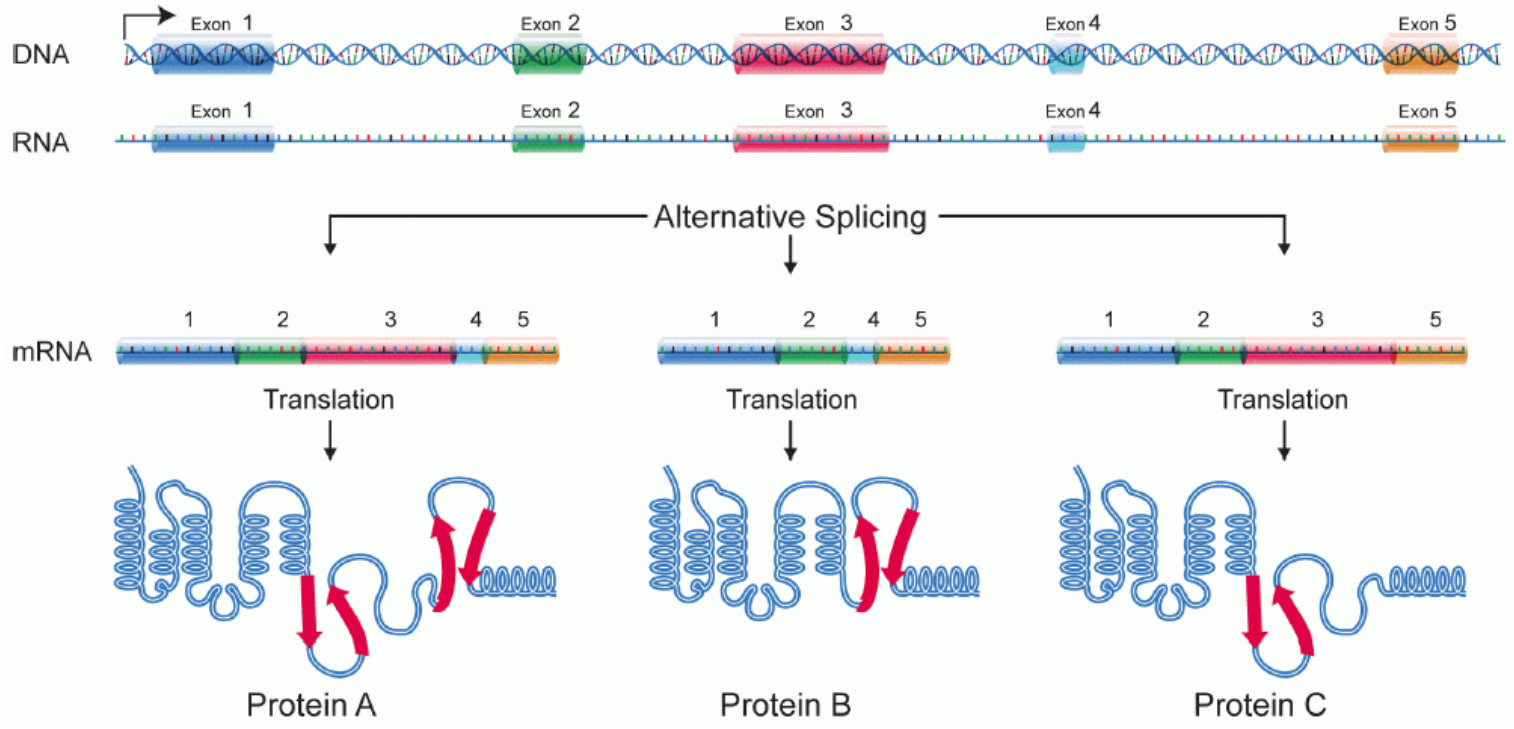
Why Transcriptomics: Gene expression profiling

- Get expression level per gene and per transcript: genes are up or down regulated
- Identify differentially expressed genes in various biological conditions (disease vs. control, treament vs. control, cancer vs. normal , etc.)
- Identify pathways and biological processes that are affected under particular conditions



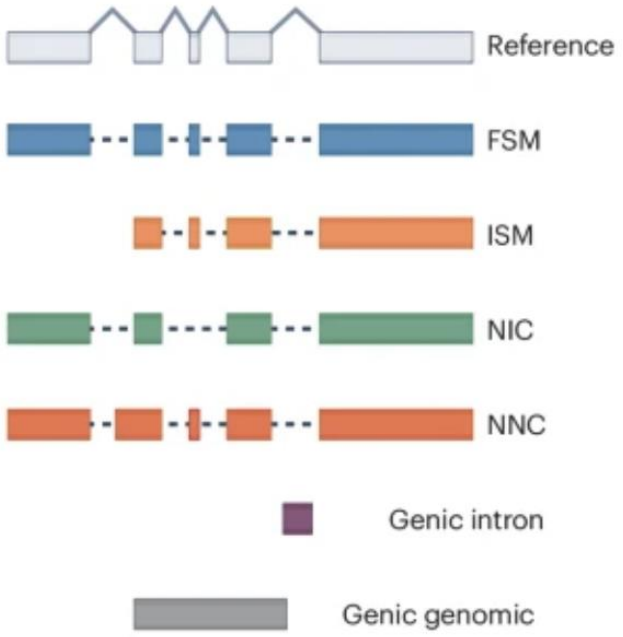
Why Transcriptomics: Alternative splicing

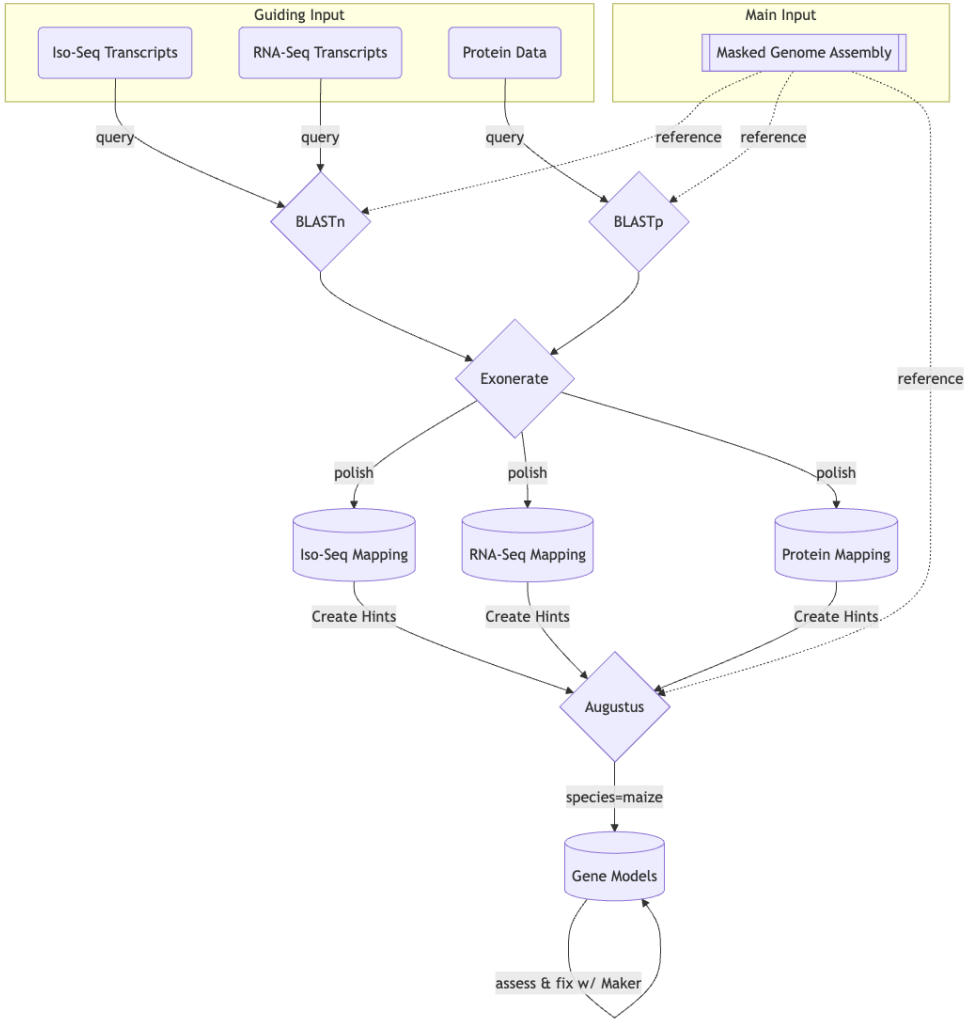
- A gene may encode more than one transcript through splicing different exon combinations



Why Transcriptomics: Genome annotation

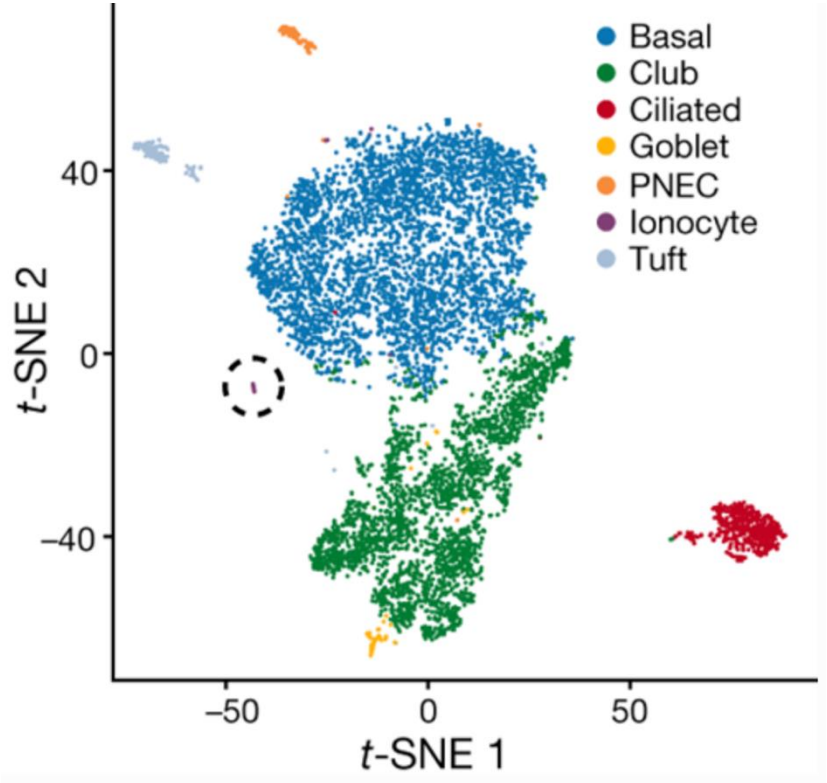
- Align reads obtained from sequencing RNA to the reference genome to identify genes & transcripts





Why Transcriptomics: Identify cell types

- Cluster cells based on expression values of genes to identify
  - Cell types and associated biomarkers
  - Disease-specific sub populations
  - Cell lineages
  - Heterogeneity within a sample



Montoro et al., 2018

RNA Challenges

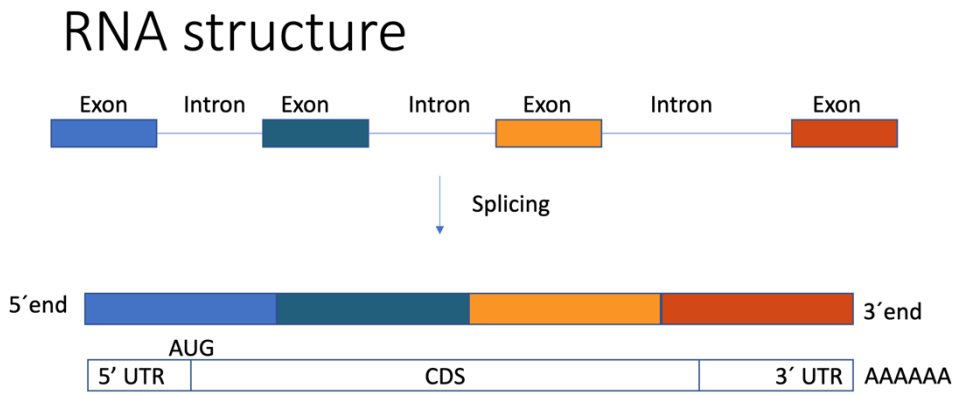
DNA is the same in all cells and tissues

RNA varies by cell type, tissue types, and time points

Many types of RNAs

Each gene encodes multiple RNAs (isoforms)

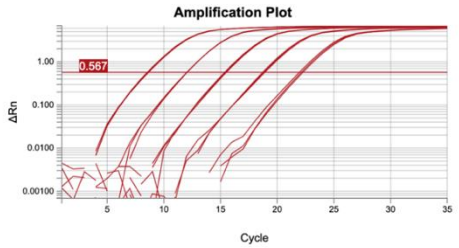
Genes are regulated by other factors



Historical methods for transcriptomics

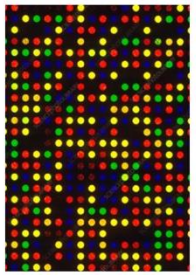
RT-PCR
1990-2000

- Low cost & fast
- Homogeneous samples
- Supervised target (Primers)
- High sensitivity
- Relative quantification
- Simple analysis



Microarray
2000-2010

- High Throughput & Cost-Effective
- Homogenous samples
- Well-established & Standardized pipelines
- Reliable for Known Sequences (probe-based)
- Good for **clinical diagnostics**
- Fast Data Processing

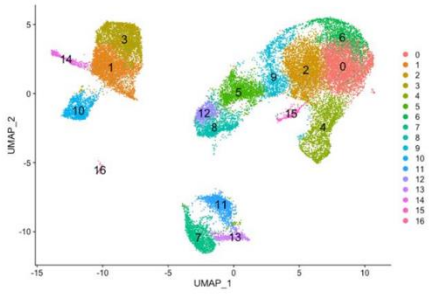


Bulk RNA-Seq
2010-2020

- Medium-high cost
- Homogeneous samples
- Unbiased detection
- True whole transcriptome
- Relative quantification
- Medium/complex analysis

Single Cell RNA-Seq
2020-

- High cost
- Heterogeneous samples
- Unbiased detection
- True whole transcriptome
- Count-based quantification
- Complex analysis



RNA-seq Strategies

	Library prep method	Core idea	Typical read layout	Best-fit sequencing technologies
RNA species / selection	Poly(A)-selected mRNA-seq	Enrich polyadenylated transcripts with oligo-dT, then fragment and make cDNA.	75–150 bp paired-end Illumina; 50 bp single-end possible for simple DE.	Illumina short-read platforms; some Ion Torrent implementations.
	rRNA-depleted total RNA-seq	Deplete rRNA and sequence the remaining RNA (coding plus many noncoding).	75–150 bp paired-end Illumina.	Illumina short-read; also used as input for some long-read WTS protocols.
	3' tag-based RNA-seq (3' RNA-seq, Tag-Seq, DGE-Seq)	Capture and sequence a short region near the 3' end of each poly(A)+ transcript.	50–100 bp single-end or short paired-end Illumina.	Illumina short-read.
	Small RNA / miRNA-seq	Size-select 18–30 nt RNAs, ligate specific adapters to RNA, then RT-PCR.	50–75 bp single-end Illumina.	Illumina (MiSeq/NextSeq/NovaSeq class); some Ion Torrent.
Strandedness	Non-stranded RNA-seq (poly(A) or total)	Classic cDNA libraries without preserving orientation.	75–150 bp paired-end or single-end Illumina.	Illumina short-read.
	Strand-specific (directional) RNA-seq	Preserve strand information via dUTP second-strand marking or directional adapters.	75–150 bp paired-end Illumina.	Illumina short-read; nearly all modern kits support this format.
Input amount / RNA quality	Standard-input bulk RNA-seq	Poly(A) or rRNA-depleted libraries from tens–hundreds of ng total RNA.	75–150 bp paired-end Illumina.	Illumina short-read.
	Low-input / ultra-low-input bulk RNA-seq	Template switching or heavy pre-amplification to work from pg–ng RNA.	Similar layouts to standard bulk (paired-end Illumina).	Illumina short-read; some long-read workflows from amplified cDNA.
	Degraded / FFPE-optimized RNA-seq	rRNA-depletion or capture-based protocols tuned for fragmented RNA.	75–150 bp paired-end Illumina.	Illumina short-read.
Single-cell / spatial	Single-cell RNA-seq (scRNA-seq)	Partition individual cells into droplets/plates, barcode cell and molecules, usually 3' or 5' tag-based with UMIs.	Paired-end Illumina (short read for cell/UMI, longer read for transcript tag).	Illumina short-read platforms.
	Single-nucleus RNA-seq (snRNA-seq)	Similar to scRNA-seq but from isolated nuclei; often 3' tag-based.	Paired-end Illumina.	Illumina short-read.
	Spatial transcriptomics (array-based)	Capture transcripts on spatially barcoded arrays or slides, then sequence barcoded cDNA.	Paired-end Illumina (one read for spatial barcode, one for transcript).	Illumina short-read.
Long-read / full-length	Long-read full-length cDNA (Iso-Seq, similar)	Convert poly(A)+ RNA to full-length cDNA, size-select, ligate long-read adapters; sequence entire transcripts.	kb-scale reads on PacBio or ONT (no fixed insert size).	PacBio (Iso-Seq), ONT PromethION/GridION/MinION.
	Direct RNA-seq (ONT)	Attach adapters to native RNA and sequence directly, without RT or PCR.	Long nanopore reads across full transcripts.	Oxford Nanopore devices.
	Long-read total/nascent RNA (WTS, GRO/PRO-like)	Adapt rRNA-depleted or nascent RNA for long-read sequencing, often with minimal fragmentation.	Long PacBio or ONT reads; protocol-dependent.	PacBio WTS, ONT long-read.
Specialized functional	Nascent transcription RNA-seq (GRO-seq, PRO-seq, NET-seq)	Label or capture nascent RNA associated with RNA polymerase and sequence it.	Typically Illumina short-read single- or paired-end, depending on protocol.	Illumina short-read.
	Ribo-seq (ribosome profiling)	Digest unprotected RNA, isolate ribosome-protected fragments, and sequence them.	50–75 bp single-end Illumina.	Illumina short-read.
	CLIP-seq-type methods (HITS-CLIP, PAR-CLIP, iCLIP, eCLIP)	Crosslink RNA–protein complexes, immunoprecipitate an RBP, and sequence bound RNA fragments.	50–100 bp single-end or paired-end Illumina.	Illumina short-read.
	Targeted RNA-seq panels	Hybrid-capture or amplicon enrichment of specific transcripts or fusions.	75–150 bp paired-end Illumina.	Illumina, Ion Torrent.

RNA-seq Technologies

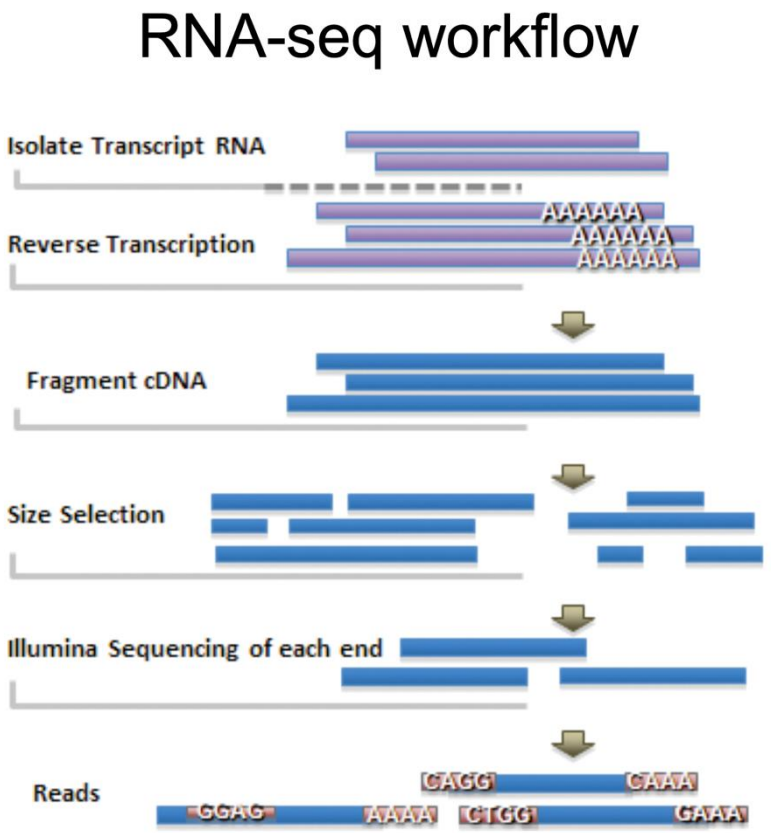
Library Prep Strategies

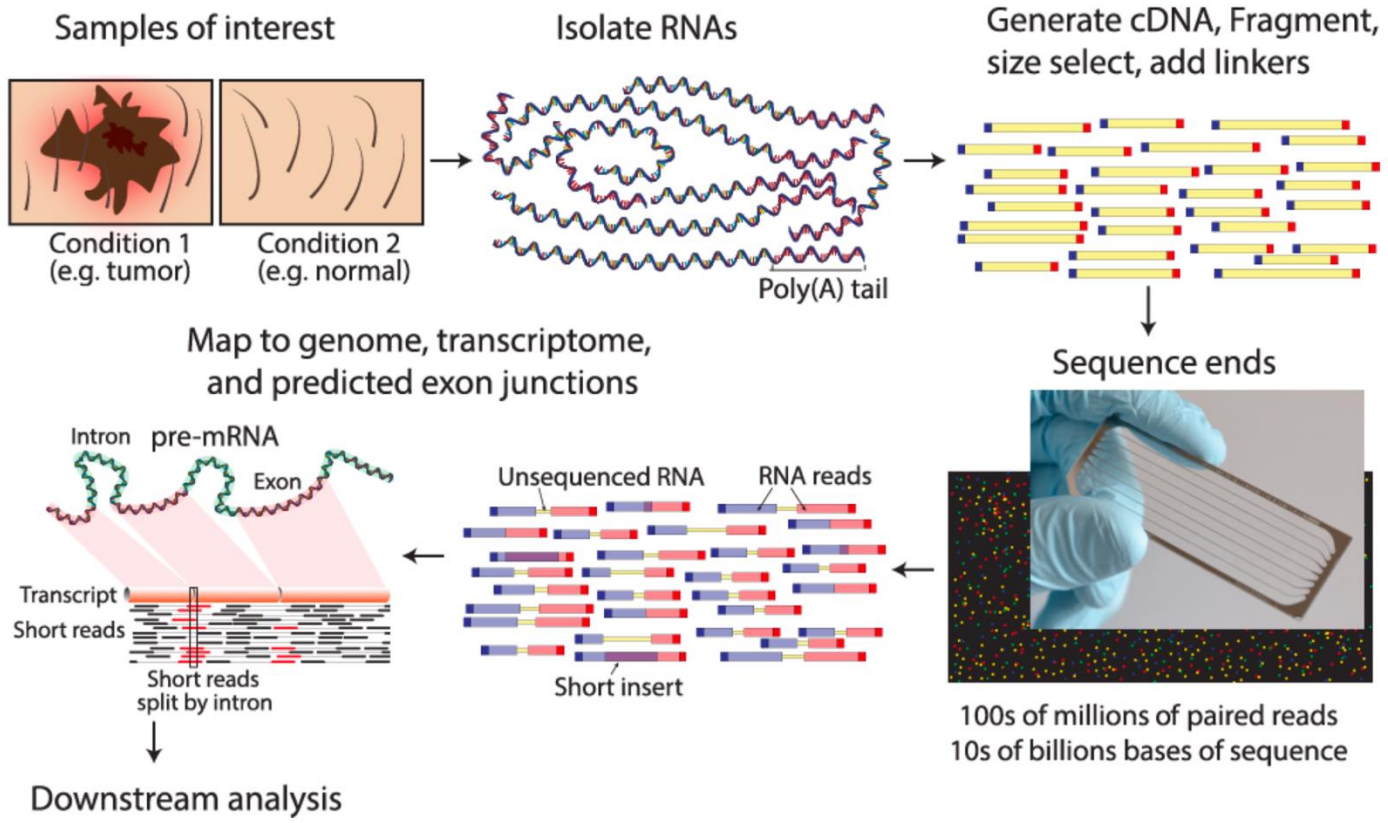
Category	Method	Platform/Kit	Application
RNA Selection	Poly(A) Selection	TruSeq, NEBNext Poly(A)	mRNA-focused studies
RNA Selection	rRNA Depletion	Ribo-Zero, QIAseq FastSelect	All RNA types, degraded
RNA Selection	3' Tag-seq	QuantSeq, Lexogen	Gene expression counting
RNA Selection	Small RNA-seq	Illumina Small RNA kit	miRNA, piRNA profiling
Strandedness	Strand-Specific	dUTP, Accel-NGS	Antisense detection
Strandedness	Non-Stranded	Standard TruSeq	Simple, cost-effective
Input Requirements	Ultra-Low (pg)	SMART-seq3, FLASH-seq	Single cells, FFPE
Input Requirements	Low Input (ng)	SMARTer, Pico v2	Limited samples
Input Requirements	Standard (µg)	Standard protocols	Abundant material
Single-cell/Spatial	Single-cell	10x Chromium, Drop-seq	Cell heterogeneity
Single-cell/Spatial	Spatial	Visium, MERFISH, Xenium	Tissue architecture
Long-read/Full-length	Long-read	ONT Direct RNA	Isoform variants
Long-read/Full-length	Full-length cDNA	PacBio Iso-Seq	Complete transcripts
Specialized Functional	Nascent RNA	SLAM-seq, TimeLapse-seq	RNA dynamics/stability
Specialized Functional	CLIP/RIP-seq	eCLIP, PAR-CLIP	RNA-protein binding

Sequencing Technologies

Technology	Platform	Read Length	Throughput	Key Applications
Short-Read SBS	Illumina (NovaSeq X, NovaSeq, NextSeq, MiSeq)	50–300 bp	Very High (up to 20 Tb)	Gene expression, differential analysis, variant calling
Long-Read (SMS)	PacBio (Sequel II/IIe, Revio)	10–25 kb (HiFi)	Moderate (90 Gb HiFi)	Full-length isoforms, alternative splicing, fusion detection
Nanopore	Oxford Nanopore (MinION, GridION, PromethION)	>100 kb possible	Moderate–High (up to 7 Tb)	Direct RNA-seq, real-time analysis, base modifications

Short-read RNA-seq for transcriptomics





Source: rnabio.org

Agenda – Day 01

أكاديمية كاوست
KAUST ACADEMY

Morning Session

09:00-09:30	S1: Welcome and Introduction
09:30-10:00	S2: Recap: Foundations in Bioinformatics & Overview of Stage 3
10:00-10:30	S3: Bioinformatics Computing Environments
10:30-12:00	*L1: Warming-up for Large-scale Analysis & Genomic File Formats*

Afternoon Session

02:00-02:45	S4: Transcriptomics Overview & Applications
02:45-03:30	*L2: Hands-on with Transcriptomics Data*
03:30-04:15	S5: RNA-seq Experimental Design
04:15-05:00	S6: Team Formation & Project Overview

أكاديمية كاوست
KAUST ACADEMY

RNA-seq Experimental Design

A Comprehensive Guide to Planning and Executing RNA Sequencing Experiments

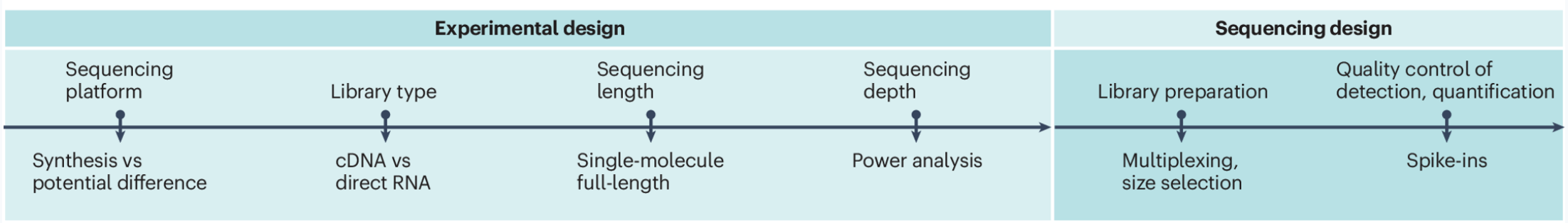
Day 01 – Session 05

RNA-seq Workflow Overview

Key Success Factors

- Clear hypothesis and well-defined research questions
- Adequate biological replicates (minimum 3 per condition) sequenced to a *sufficient* depth
- High-quality RNA with RIN ≥ 7

Experimental Design Considerations



Questions to Define

- What is the research question?
- Quantitative (DGE) or qualitative (annotation)?
- How many conditions/treatments?
- Number of replicates needed?
- Sequencing depth required?
- Single-end or paired-end reads?

Common Design Types

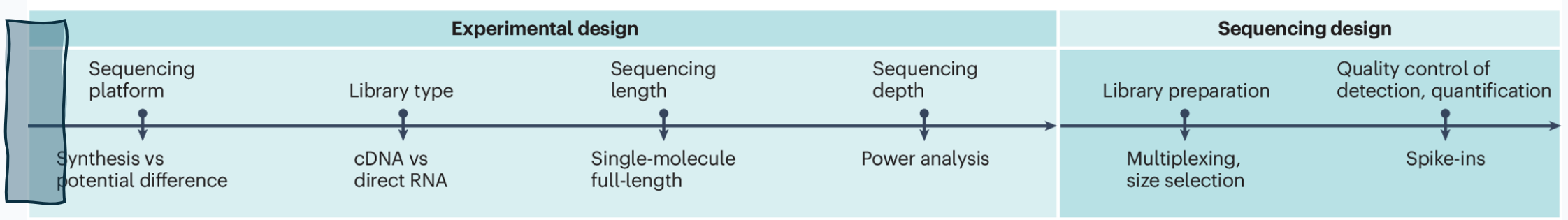
Two-group comparison
Control vs Treatment

Time course
Multiple time points

Factorial design
Multiple factors (2×2, 3×2)

Dose-response
Multiple concentrations

Experimental Design Considerations



Biological Context & Scientific Question

Before any technical decision, clearly define:

- The biological system (organism, tissue, cell type, developmental stage)
- The question being asked (differential expression, isoform usage, RNA processing, discovery vs. quantification)
- Expected magnitude of change and biological variability

RNA-seq does not create signal - it samples what is already present. Poorly defined biological questions lead to misinterpreted experiments.

Common Design Types

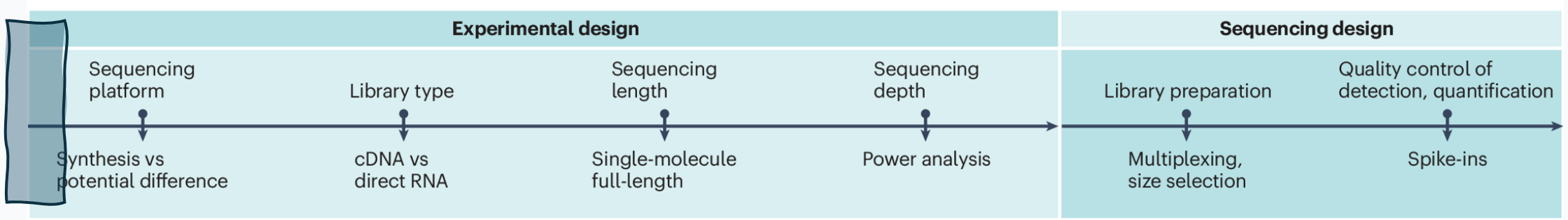
Two-group comparison
Control vs Treatment

Time course
Multiple time points

Factorial design
Multiple factors (2×2, 3×2)

Dose-response
Multiple concentrations

Experimental Design Considerations



Biological Context & Scientific Question

- Is the Signal Present in the Transcriptome?
- Is the expected signal transcriptional?
- Are changes expected at the gene, isoform, or RNA processing level?
- Could regulation be post-transcriptional or protein-level only?

- If the signal is weak or rare: More depth; More biological replicates; Targeted or enrichment-based approaches

Sample Complexity & RNA Repertoire

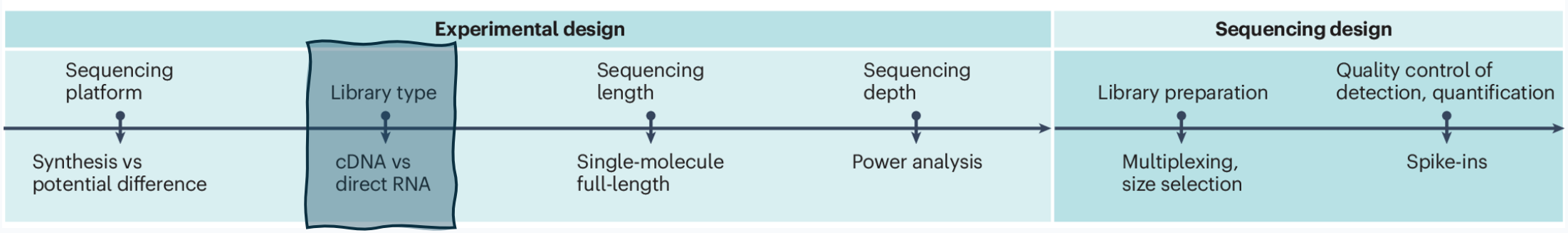
- A sample contains: Mature mRNA; Pre-mRNA; Non-coding RNAs; Degradation intermediates

Consider:

- RNA maturation and turnover dynamics
- Tissue-specific complexity
- Dominance of a few highly expressed transcripts

RNA-seq is a snapshot in time: Is a single time point sufficient? Or transient responses expected? Would multiple time points improve interpretability?

Experimental Design Considerations

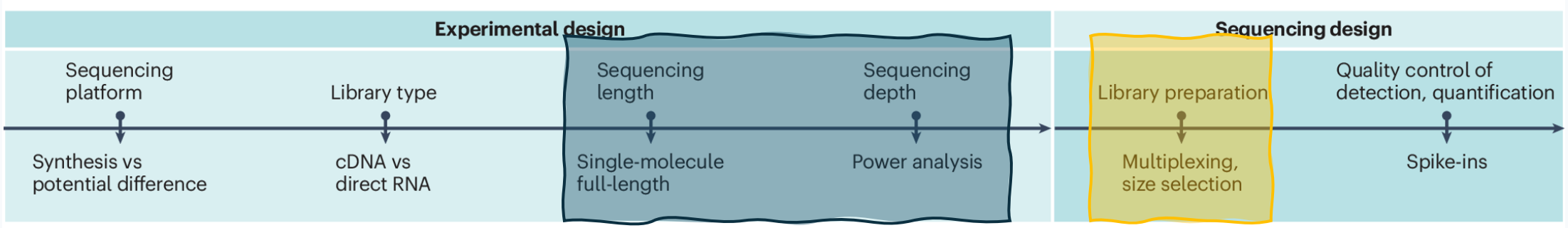


Library Types

Which RNA to sequence and how:

- RNA enrichment: poly(A), rRNA depletion, total RNA
- Compatibility with degraded samples
- Strand-specific vs. unstranded libraries

Experimental Design Considerations



Sequencing Strategy

- Single-end: cheaper, sufficient for gene-level analysis
- Paired-end: better for isoforms, splicing, novel transcripts
- Short-Read lengths: 50, 75, 100, 150, etc.
- Sequencing Depth

Influences:

Mapping accuracy; Isoform resolution; Detection of splice junctions

Library Prep:

- Transcriptome complexity & Biological signal strength
- Number of conditions
- Replicate count
- PCR Cycles: Less is less bias

Typical trade-off: More samples vs. deeper sequencing

Sample Multiplexing

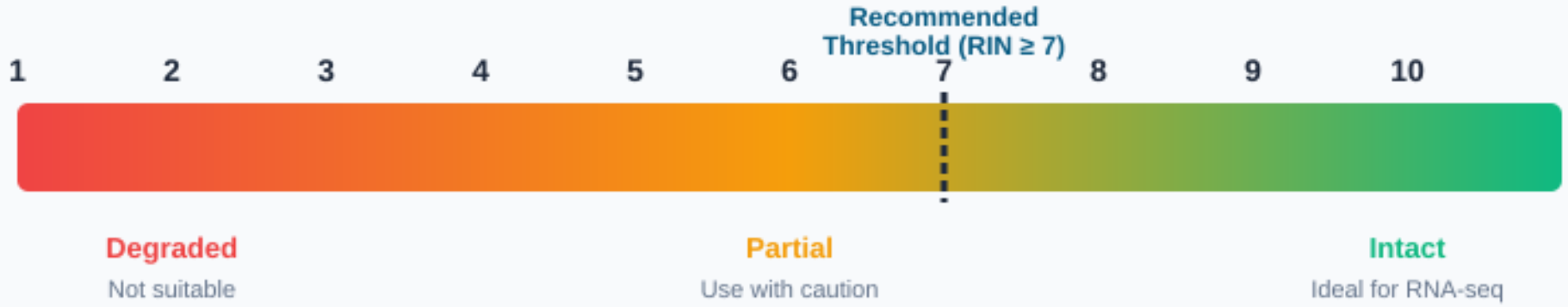
- Index hopping risk
- Balance of read depth across samples
- Avoid mixing very different RNA complexities in one pool

Biological Replicates & Statistical Power

Replicates	Use Case	Statistical Power
n = 2	Exploratory only	**Very Low**
n = 3	Minimum standard for DGE	**Moderate**
n = 5-6	Recommended for robust analysis	**Good**
n ≥ 10	High biological variability	**Excellent**

RNA Quality Control

RNA Integrity Number (RIN) Scale



Quality Assessment Methods

Method	Key Metric	Purpose
Agilent Bioanalyzer	RIN score (1-10)	Gold standard for integrity
NanoDrop Spectrophotometer	260/280 ratio (~2.0)	Purity assessment
Qubit Fluorometer	Concentration (ng/µL)	Accurate quantification

Sequencing Parameters: human genome

Parameter	Recommendation	Notes
Sequencing Depth	20–30 million reads/sample	Higher for low-abundance genes or complex organisms
Read Length	100–150 bp	Longer reads for isoform detection and novel transcripts
Read Type	Paired-end preferred	Better alignment, splice junction detection
Strand Specificity	Stranded library	Essential for antisense transcripts and overlapping genes

Sequencing Depth Guidelines by Application

Gene-level DGE

20-30M reads

Transcript-level analysis

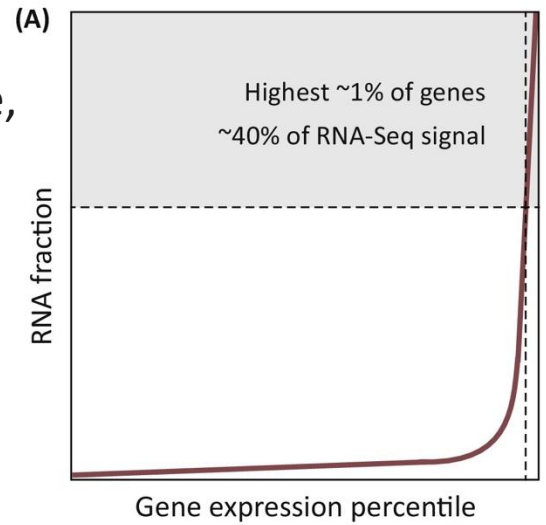
30–50M reads

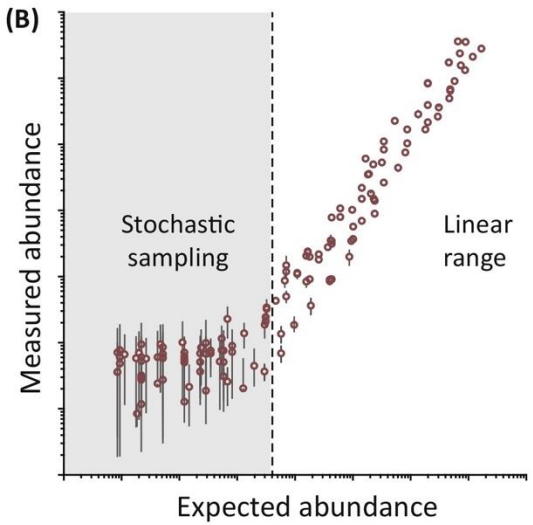
Novel transcript discovery

50–100M reads

RNA-seq: matter of length and depth

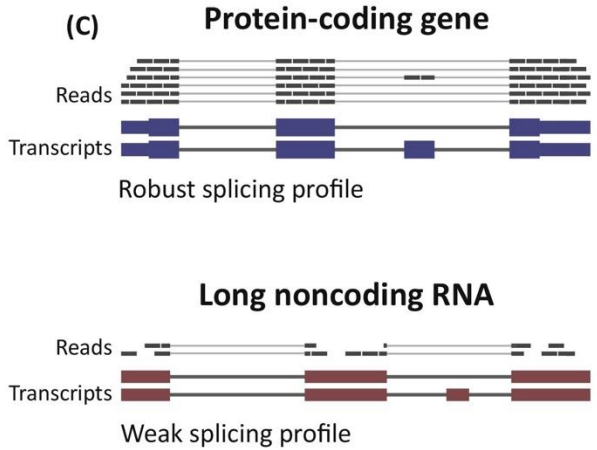
in a typical human sample, the top 1% most highly expressed protein-coding genes commonly soak up ~40% of sequencing reads

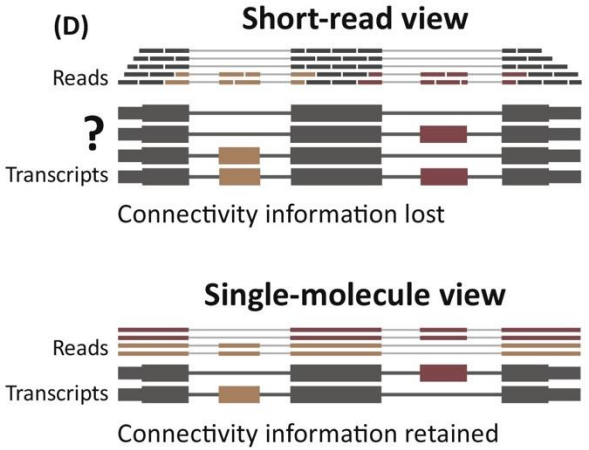




spike-in transcripts at high and moderate abundance are robustly quantified. However, among spike-ins of lower abundance, stochastic sampling leads to quantitative variability and, ultimately, loss of linearity between expected and observed abundance

rare transcripts, such as lncRNAs, are often poorly resolved





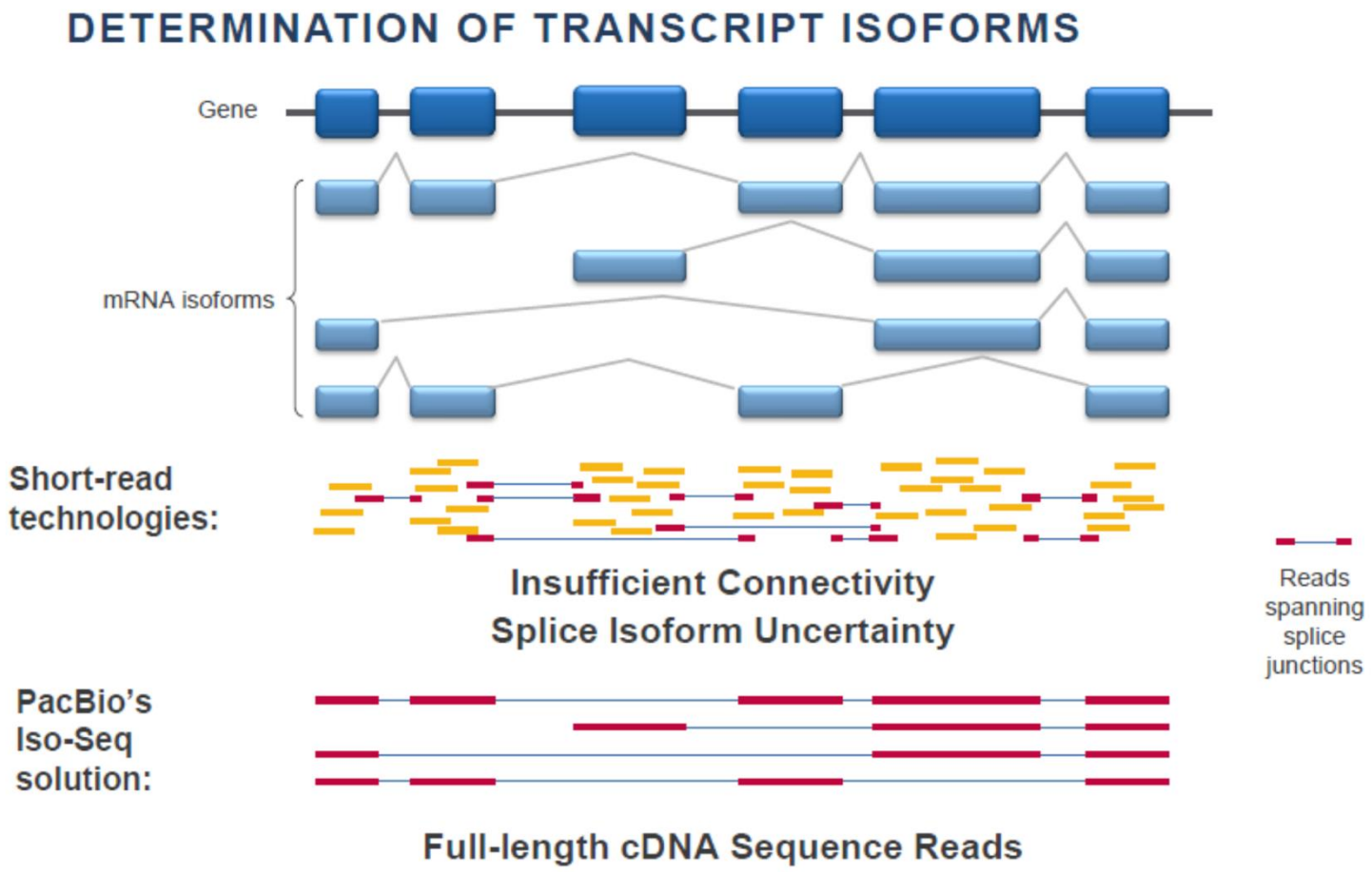
the relationship between distant exons cannot be judged because these are never represented on the same sequenced fragment

Long read technology better resolve the transcript structure

Trends in Genetics

Deveson *et al*, *TiGs* (2017), doi: 10.1016/j.tig.2017.04.004

Long read vs. short read sequencing



Long read technology resolve the transcript structure better

Common Pitfalls & Best Practices

⚠ Common Pitfalls

- ✗ Insufficient biological replicates
- ✗ Poor RNA quality (RIN < 7)
- ✗ Batch effects not controlled
- ✗ Inadequate sequencing depth
- ✗ Ignoring library complexity
- ✗ No randomization of samples

✓ Best Practices

- ✓ Use ≥3 biological replicates per condition
- ✓ Ensure RIN ≥ 7 for all samples
- ✓ Process samples in randomized batches
- ✓ Aim for 20M+ reads per sample
- ✓ Include spike-in controls (ERCC)
- ✓ Document all protocols thoroughly

Experimental Design Checklist

Planning

- ☐ Define hypothesis
- ☐ Choose design type
- ☐ Calculate sample size
- ☐ Plan for batch effects

Sample Prep

- ☐ Verify RNA quality (RIN ≥ 7)
- ☐ Measure concentration
- ☐ Randomize processing

Sequencing

- ☐ Select appropriate depth
- ☐ Choose read length
- ☐ Use stranded protocol

A well-designed experiment is the foundation for meaningful RNA-seq results

Additional reading

REVIEW Open Access

A survey of best practices for RNA-seq data analysis

Ana Conesa[1,2*], Pedro Madrigal[3,4*], Sonia Tarazona[2,5], David Gomez-Cabrero[6,7,8,9], Alejandra Cervera[10], Andrew McPherson[11], Michał Wojciech Szcześniak[12], Daniel J. Gaffney[3], Laura L. Elo[13], Xuegong Zhang[14,15] and Ali Mortazavi[16,17*]

Genome Biology (2016), doi: 10.1186/s13059-016-0881-8

"Data don't make any sense,
we will have to resort to statistics."

Agenda – Day 01

أكاديمية كاوست
KAUST ACADEMY

Morning Session

09:00-9:30	S1: Welcome and Introduction
09:30-10:00	S2: Recap: Foundations in Bioinformatics & Overview of Stage 3
10:00-10:30	S3: Bioinformatics Computing Environments
10:30-12:00	*L1: Warming-up for Large-scale Analysis & Genomic File Formats*

Afternoon Session

02:00-02:45	S4: Transcriptomics Overview & Applications
02:45-03:30	*L2: Hands-on with Transcriptomics Data*
03:30-04:15	S5: RNA-seq Experimental Design
04:15-05:00	S6: Team Formation & Project Overview

أكاديمية كاوست
KAUST ACADEMY

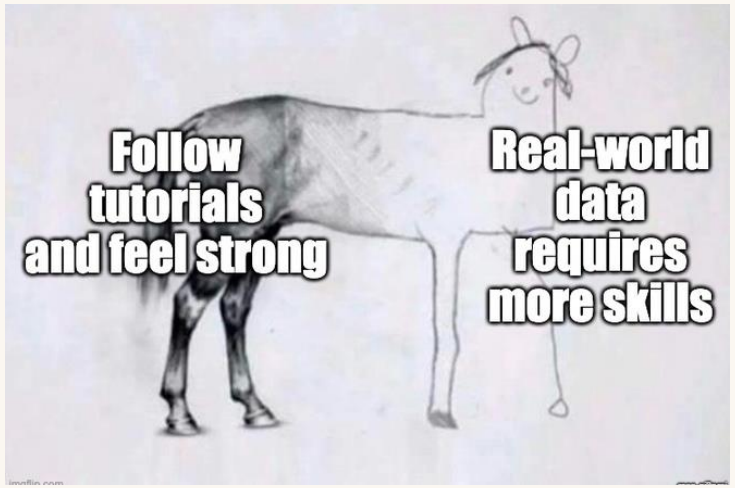
Bioinformatics challenges, Project Overview, & Team Formation

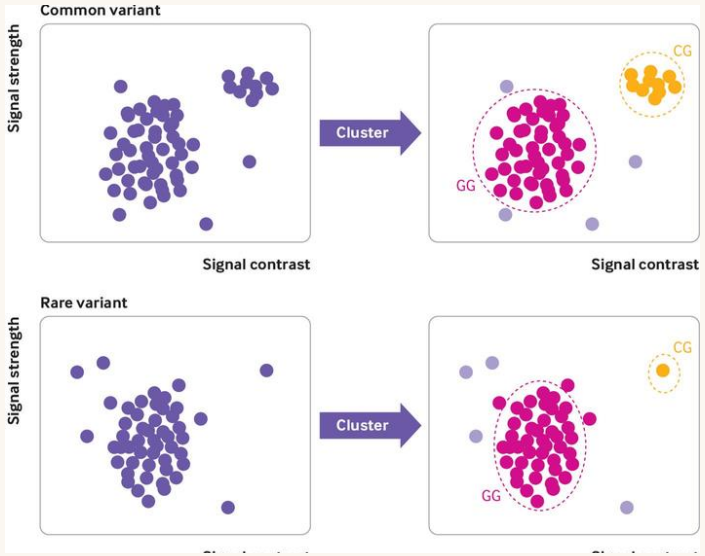
Bioinformatics challenges, applied RNA-seq project scenarios, team formation, dataset and topic selection

Day 01 – Session 06

Diaries of a Bioinformatician

ICE BREAKER





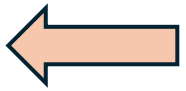
Guess the issue:

Examples:

```
x1 = pd.read_csv("/home/user/Desktop/sample1_counts.txt", sep="\t")
x2 = pd.read_csv("/home/user/Desktop/sample2_counts.txt", sep="\t")

foo = x1.merge(x2, on="gene_id")

temp2 = foo.iloc[:, 1:]
temp2 = temp2 + 1
temp2 = np.log2(temp2)

y = temp2.mean(axis=1)

plt.figure()
plt.plot(y)
plt.savefig("/home/user/Desktop/plot.png")

# repeat same thing again for no reason
x3 = pd.read_csv("/home/user/Desktop/sample3_counts.txt", sep="\t")
x4 = pd.read_csv("/home/user/Desktop/sample4_counts.txt", sep="\t")

foo2 = x3.merge(x4, on="gene_id")
temp3 = foo2.iloc[:, 1:]
temp3 = temp3 + 1
temp3 = np.log2(temp3)

y2 = temp3.mean(axis=1)

plt.figure()
plt.plot(y2)
plt.savefig("/home/user/Desktop/plot2.png")
```

No comments

No Functions



Hard-coded paths

Examples:

```python
# ----------------------------
# Core utilities
# ----------------------------
def load_counts_table(path: Path, gene_id_col: str) -> pd.DataFrame:
    """
    Load a gene-level counts table.
    Expected: a tab-delimited file with a gene_id column and one counts column.
    """
    if not path.exists():
        raise FileNotFoundError(f"Counts file not found: {path}")

    df = pd.read_csv(path, sep="\t")
    if gene_id_col not in df.columns:
        raise ValueError(f"Missing required column '{gene_id_col}' in {path.name}")

    # Keep only gene id + numeric columns
    numeric_cols = [c for c in df.columns if c != gene_id_col]
    if len(numeric_cols) == 0:
        raise ValueError(f"No count columns found in {path.name}")

    # Ensure numeric
    for c in numeric_cols:
        df[c] = pd.to_numeric(df[c], errors="raise")

    return df[[gene_id_col] + numeric_cols]


def merge_counts(files: List[Path], gene_id_col: str) -> pd.DataFrame:
    """
    Merge multiple count tables by gene_id (inner join to keep shared genes).
    """
    merged = None
    for f in files:
        df = load_counts_table(f, gene_id_col)
        merged = df if merged is None else merged.merge(df, on=gene_id_col, how="inner")

    assert merged is not None, "No files provided."
    return merged
```

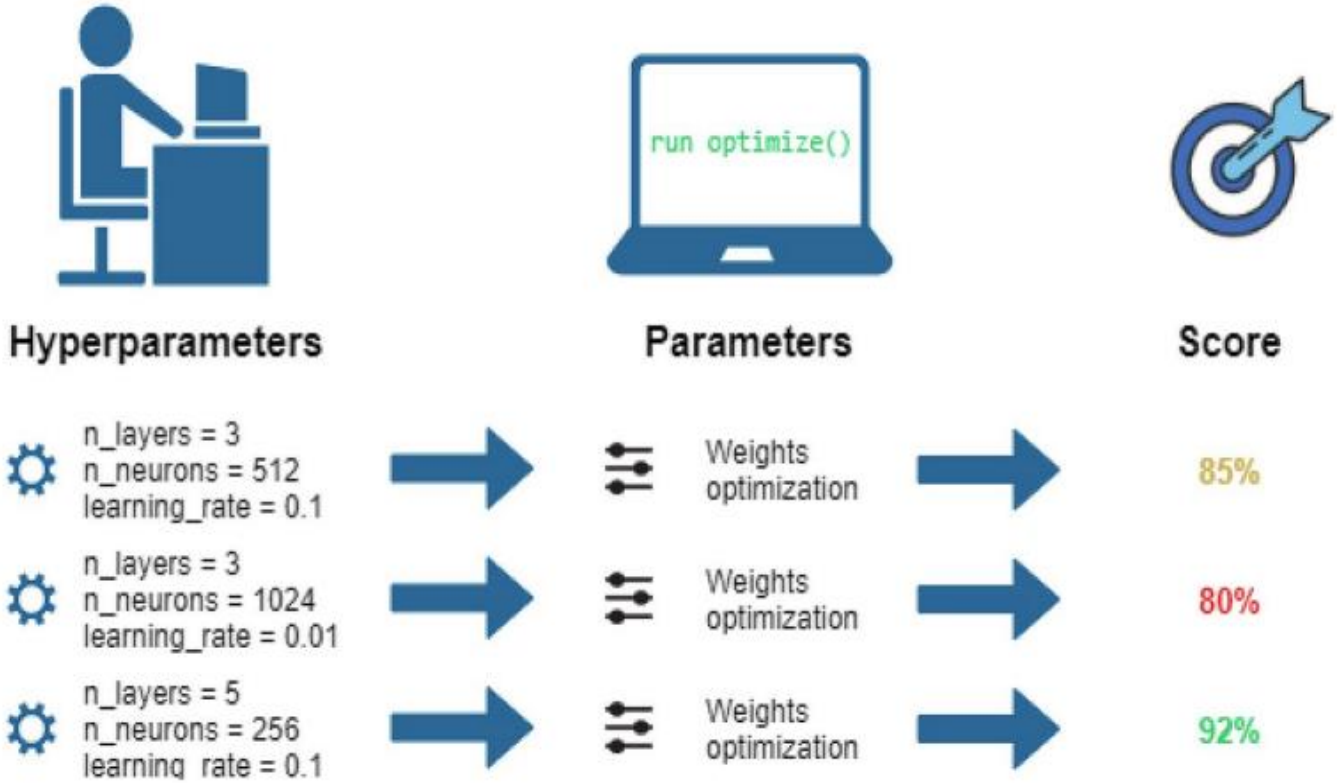
Functions

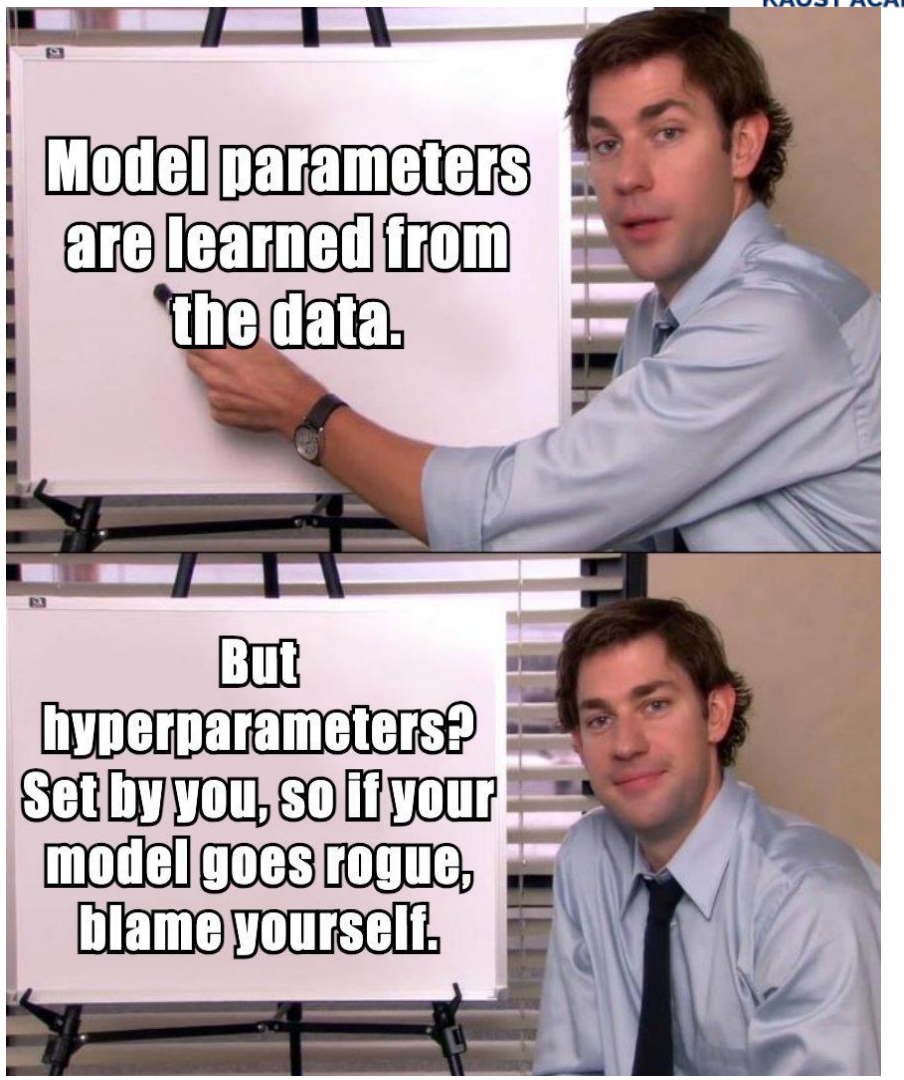
Comments

```python
# ----------------------------
# Pipeline
# ----------------------------

def run_pairwise_mean_plot(sample_a: Path, sample_b: Path, cfg: Config, label: str) -> Tuple[pd.Series, Path]:
    """
    Load two count tables, merge, log2-transform, compute mean per gene, and plot.
    """
    merged = merge_counts([sample_a, sample_b], cfg.gene_id_col)
    log_df = log2_transform_counts(merged, cfg.gene_id_col, cfg.pseudocount)
    mean_per_gene = mean_expression_per_gene(log_df, cfg.gene_id_col)

    out_png = cfg.out_dir / f"mean_expression_{label}.png"
    save_line_plot(mean_per_gene, out_png, title=f"Mean expression per gene ({label})")
    return mean_per_gene, out_png


def main() -> None:
    cfg = Config(
        data_dir=Path("data"),       # <-- change to your folder
        out_dir=Path("results"),     # <-- change to your folder
    )

    # Example inputs (no hard-coded absolute paths)
    pair1 = (cfg.data_dir / "sample1_counts.txt", cfg.data_dir / "sample2_counts.txt", "pair1")
    pair2 = (cfg.data_dir / "sample3_counts.txt", cfg.data_dir / "sample4_counts.txt", "pair2")

    for a, b, label in [pair1, pair2]:
        mean_series, fig_path = run_pairwise_mean_plot(a, b, cfg, label)
        print(f"[OK] {label}: {len(mean_series):,} genes plotted -> {fig_path}")


if __name__ == "__main__":
    main()
```

Distinguished paths

Parameters VS Hyperparameters


run optimize()
Hyperparameters
Parameters
Score
n_layers = 3
n_neurons = 512
learning_rate = 0.1
Weights optimization
85%
n_layers = 3
n_neurons = 1024
learning_rate = 0.01
Weights optimization
80%
n_layers = 5
n_neurons = 256
learning_rate = 0.1
Weights optimization
92%


Model parameters are learned from the data.
But hyperparameters? Set by you, so if your model goes rogue, blame yourself.

3-step framework to learn Bioinformatics

1. **Document**: Read & take notes.

2. **Demonstrate**: Practice examples exactly.

3. **Replicate**: Apply to real-life data.

Applied RNA-seq project scenarios.

Which genes and pathways are affected by TDP-43 knockout in your assigned chromosome ?

Project Background

- TDP-43 is a ubiquitously expressed RNA-binding protein that regulates thousands of target genes across the genome.

- During the labs we focus on data from chromosome 11
- Each group will analyze data from one of the chromosomes, check your chromosome (group) number.

- Your challenge is to identify:
  - Which genes on your chromosome are affected by TDP-43 knockout
  - What biological processes are disrupted
  - Whether your chromosome has unique or particularly strong responses to TDP-43 loss
  - Potential links to ALS/FTD pathology based on affected genes

Project Tasks I

1. Data Preparation

Each group must:

- Download the **reference genome FASTA, GTF annotation, and cDNA FASTA** for your assigned chromosome from **Ensembl (GRCh38, same release as labs)**.
- Extract chromosome-specific annotations from the GTF file.
- Report number of genes, transcripts, DNA base pairs, and number of transcript nucleotides.
- Clearly document:
  - Genome assembly version
  - Ensembl release number
  - Chromosome analyzed

2. FASTQ Exploration & Statistics

Explore your assigned raw data

Report and compare between the two conditions:

- Total reads per sample
- Average read length
- GC content
- Quality scores

Briefly comment on: Any notable differences between conditions

Team formation

- Visit the project web page and identify your group:
  https://bioinfo-kaust.github.io/academy-stage3-2026/html/projects.html

- Please let us know if your group need to be changed

SCAN ME

Thank you