Bioinformatic tools for Genome annotation

Course: Microbiomes (2021)

Sandra Godinho Silva
Instituto Superior Técnico
21/04/2021

Graphic by Jeremy Bishop @ Unsplash

About me: Sandra Godinho Silva

PhD Student @ Institute for Bioengineering and Biosciences

Instituto Superior Técnico
Lisbon, Portugal

Orcid @sandragodinhosilva

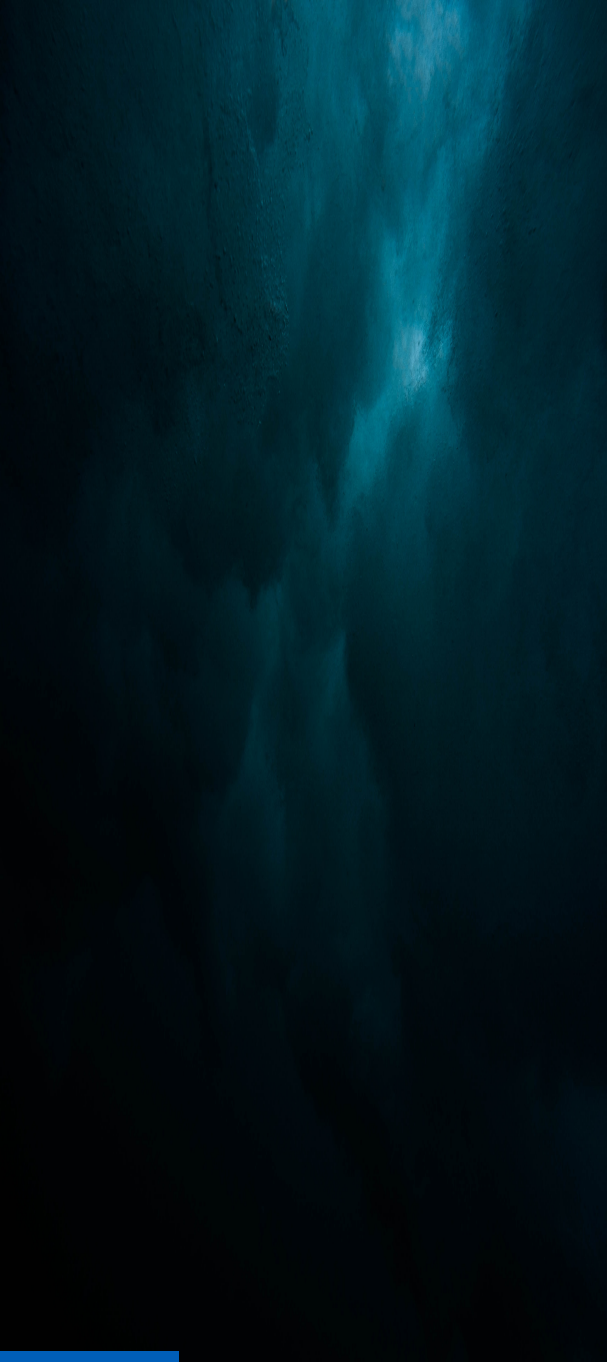
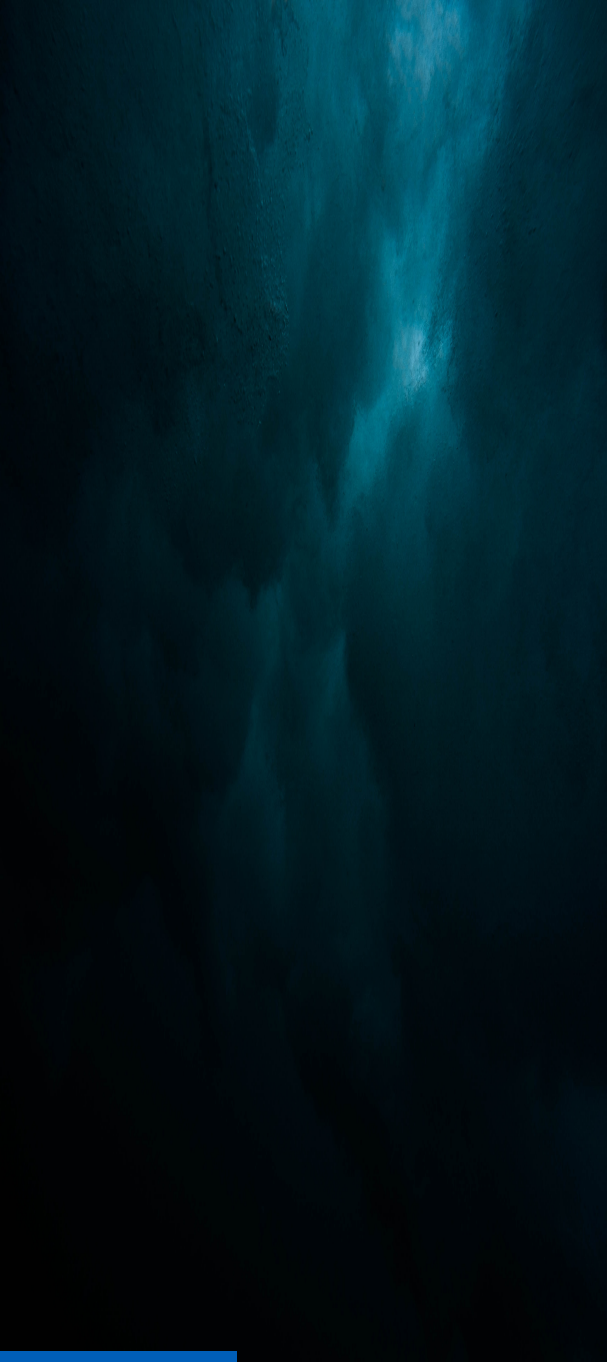


Table of contents



Part 1

Bioinformatics: what are the tools available?

Bioinformatics

Bioinformatics is the science that conjugates
biology + **statistics** + **computer sciences**
to study biological issues through the analysis of data.

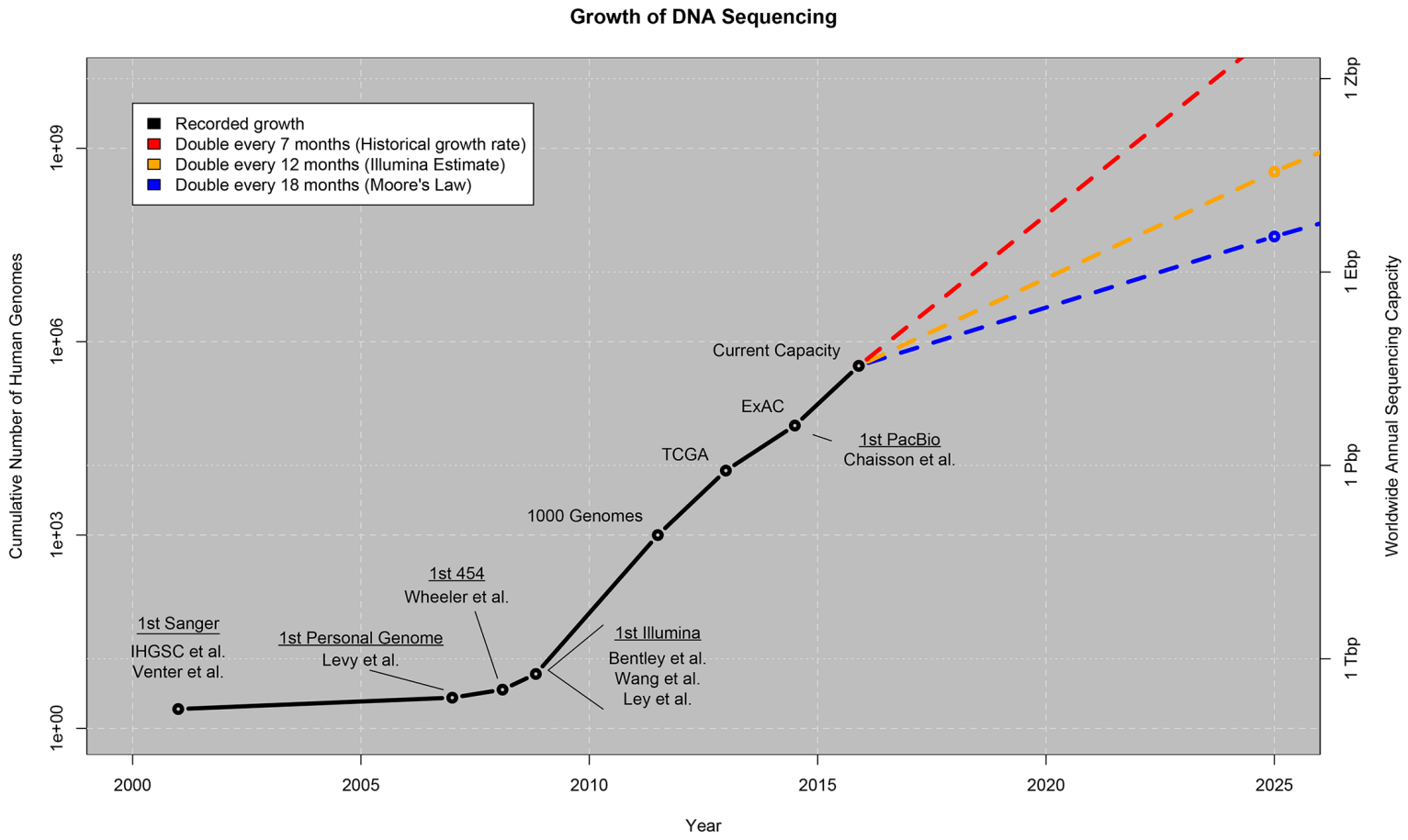
Bioinformatics is a constantly changing and updating field.

Fast developments in:

> - software
> - computing hardware
> - high throughput technologies

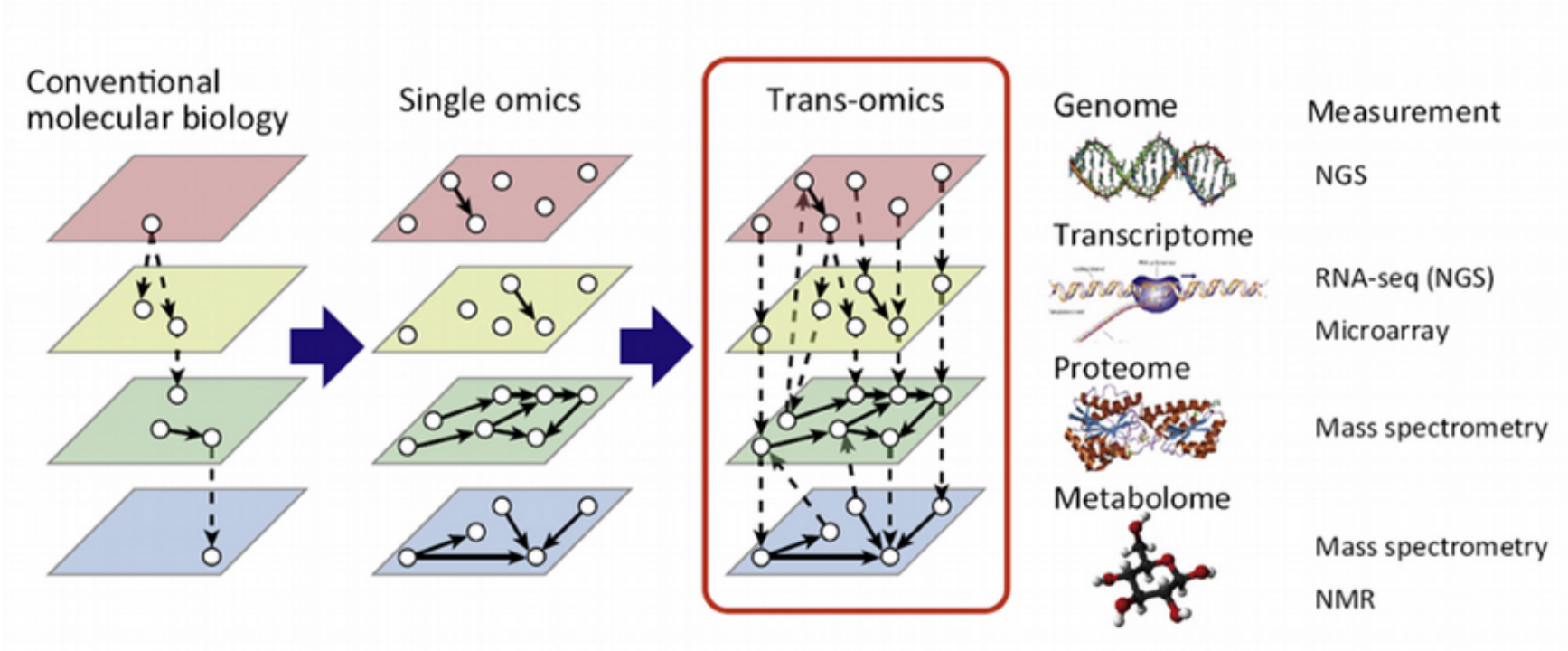
are creating a significant bottleneck:
How to thoroughly analyze this massive amount of data?

How to thoroughly analyze this massive amount of data?



Stephens et a. PLoS Biology, 2015

Integrative OMICS



Yugi et al., Trends Biotechnol. 2016 Apr; 34(4):276–290

Choose the most appropriate route for you

Nowadays, bioinformaticians and researchers working in life scientists can choose from an overwhelming collection of exciting technologies and programming languages.

Several options:

- Web-based platforms
- Scripting/programming

Web-based platforms

Allow **accessible**, **reproducible** and **transparent computational research.**

KBase | Galaxy

The U.S. Department of Energy Systems Biology Knowledgebase KBase is an open-source software and data platform designed to meet the grand challenge of systems biology — predicting and designing biological function from the biomolecular (small scale) to the ecological (large scale).
Kbase: https://www.kbase.us/

Web-based platforms

Allow **accessible**, **reproducible** and **transparent computational research.**

KBase | Galaxy

Galaxy PROJECT

Galaxy provides a system that enables researchers without informatics expertise to perform computational analyses through the web. A user interacts with Galaxy through the web by uploading and analyzing the data. Galaxy interacts with underlying computational infrastructure (servers that run the analyses and disks that store the data) without exposing it to the user. Galaxy project - Europe: https://usegalaxy.eu/

Scripting & programming

- Default login shell for most Linux distributions;
- Simple bash scripts are really useful for data manipulation.

```
ubuntu@ubuntu-VirtualBox:~/code$ bash if_with_OR.sh
Enter any number
40
You lost the game
ubuntu@ubuntu-VirtualBox:~/code$ bash if_with_OR.sh
Enter any number
15
You won the game
ubuntu@ubuntu-VirtualBox:~/code$
```

Scripting & programming

Python

- Clear and powerful object-oriented programming language;
- Portable – runs just about anywhere;
- Clear syntax – relatively easy to learn. Source: https://www.python.org/

Scripting & programming

Biopython

- Collection of Python modules that provide functions to deal with DNA, RNA & protein sequence operations;
- It has sibling projects like BioPerl, BioJava and BioRuby.
  Source: https://biopython.org/

Scripting & programming

R

- R is a powerful, popular open-source scripting language
- More than 20 years old - fairly mature - and growing in popularity.

Scripting & programming

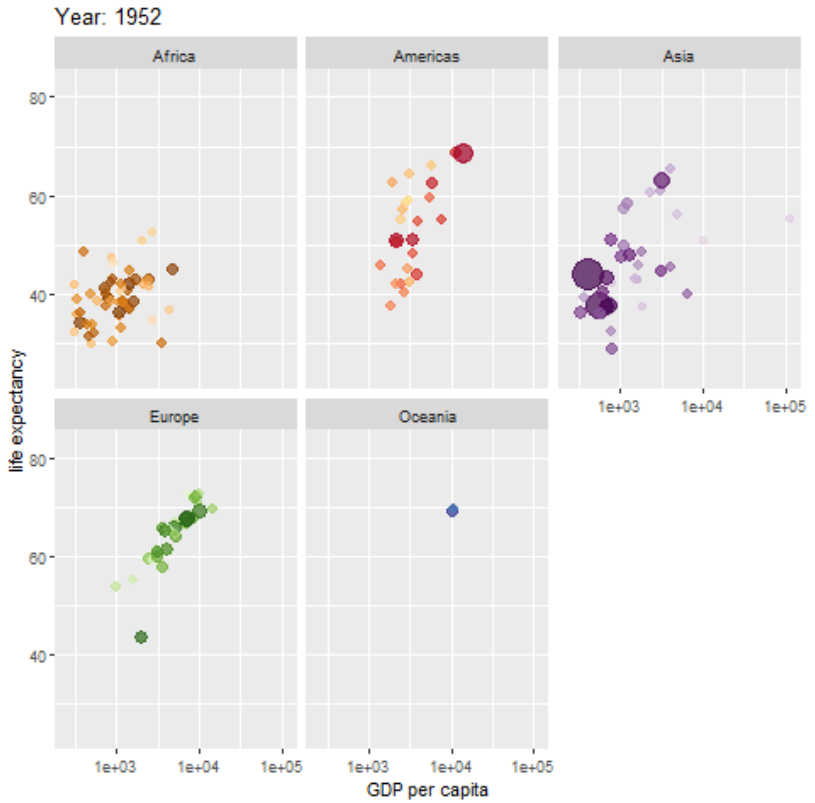
Why is R so popular?

- **Statistical Language**: R is widely used in biology, genetics as well as in statistics.
- **Vast array of packages**: With over 10,000 packages in the CRAN repository, the number is constantly growing. These packages appeal to all the areas of industry.
- **Quality Plotting and Graphing**: The popular libraries like ggplot2 and plotly advocate for aesthetic and visually appealing graphs that set R apart from other programming languages.

Scripting & programming

Dynamic graph:
ggplot2 +
gganimate

Source: link



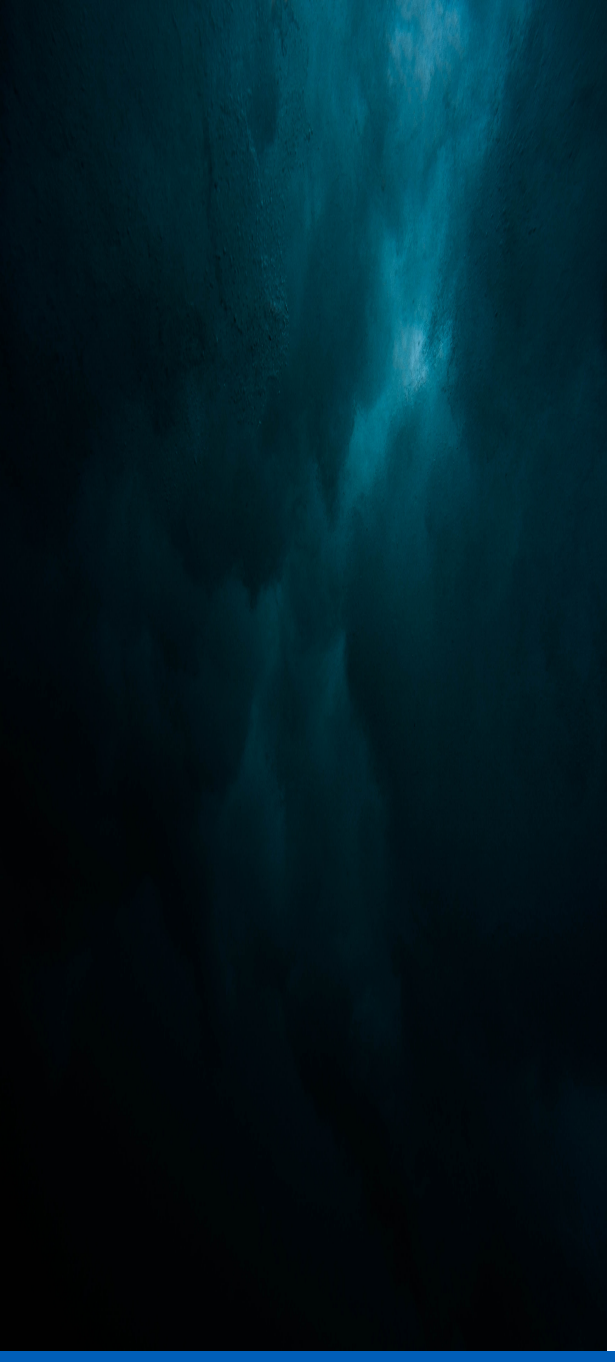
Scripting & programming

R Studio

- Integrated Development Environment (IDE).
- Provides a graphical interface to R, making it more user-friendly, and providing dozens of useful features.

Source: https://www.rstudio.com/

And much more!



Part 2

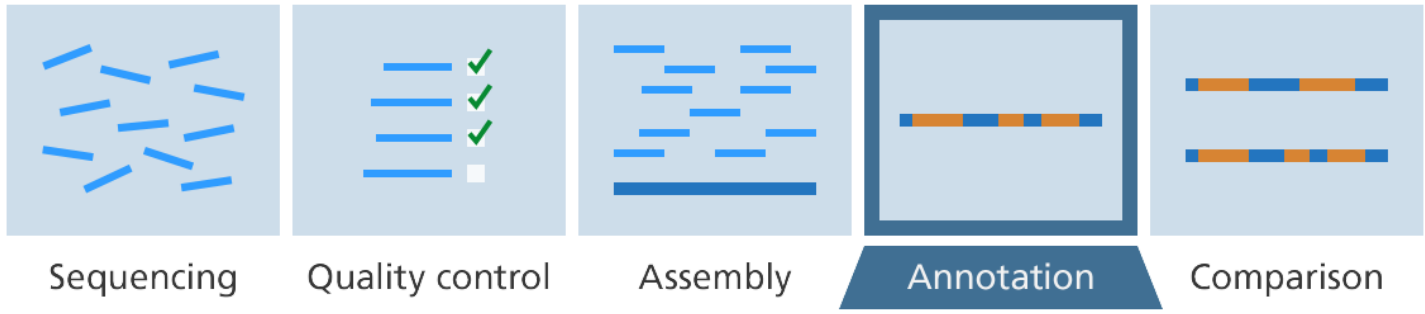
Bioinformatic tools for Genome Annotation

Genome Annotation

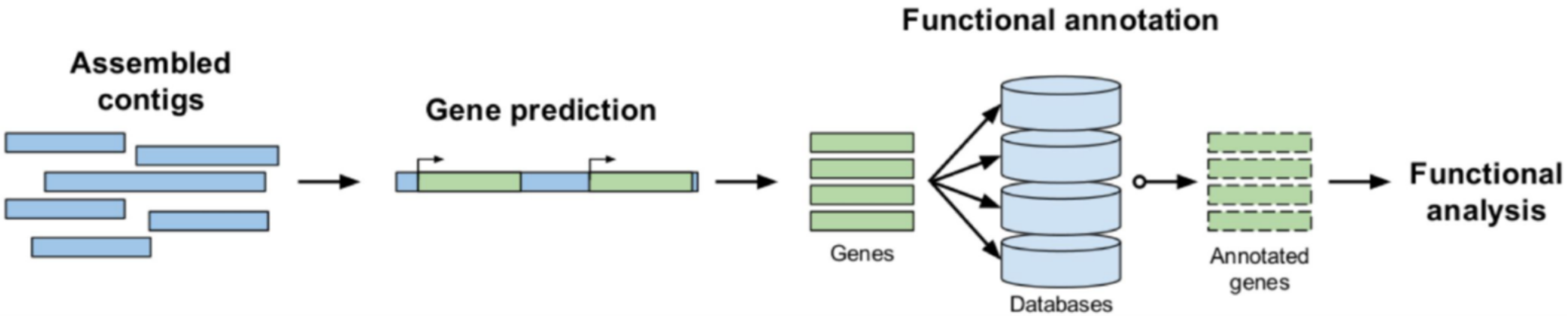
The process of attaching biological information to sequences.

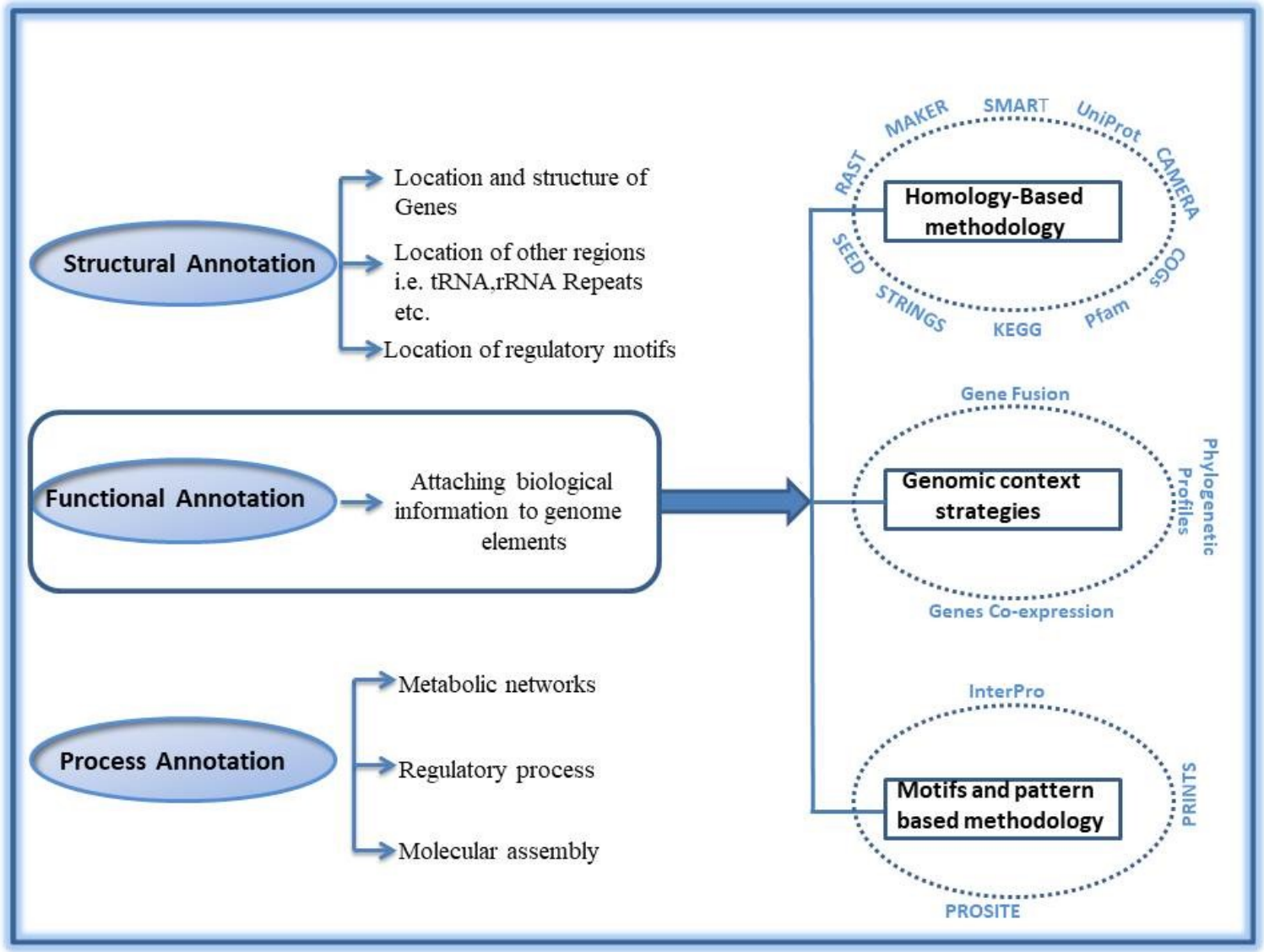
Consists of two main steps:

- structural annotation: identifying genomic elements;
- functional annotation: attaching biological information to these elements.



Source: link


Assembled contigs
Gene prediction
Functional annotation
Genes
Databases
Annotated genes
Functional analysis

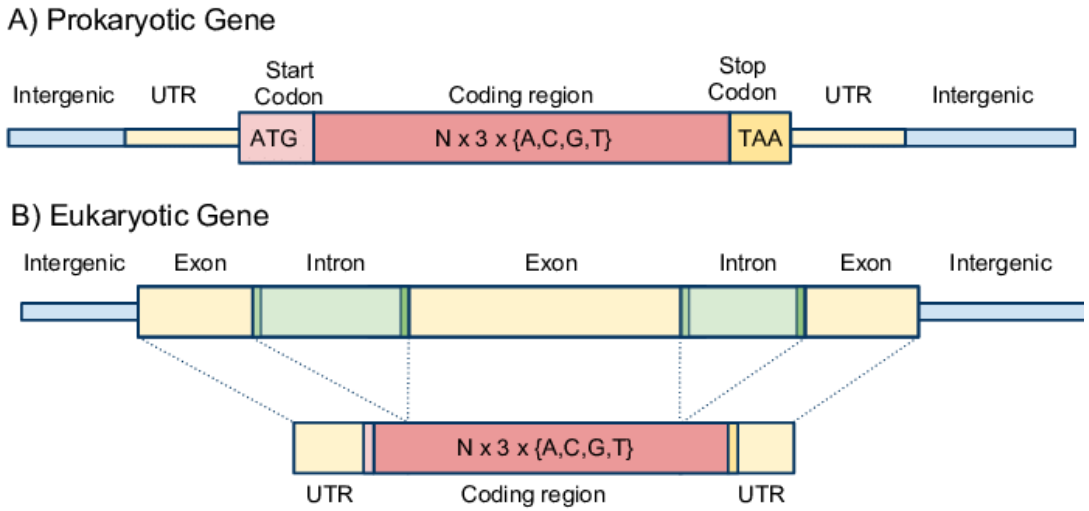


Source: https://www.ssbs.edu.in/genome-annotation.htm

Structural Annotation

The process of identifying genomic elements such as:

- open reading frames (ORFs) and their localization;
- gene structure;
- coding regions;
- regulatory motifs.



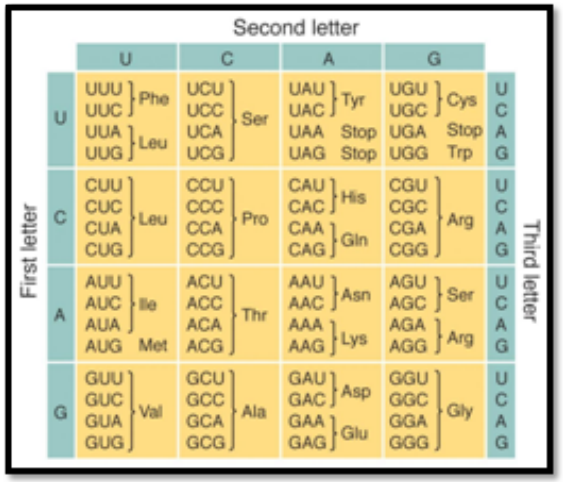
Prokaryotic and Eukaryotic gene structure. Source: link

Structural Annotation: Methods

- **Similarity**
  Similarity between sequences

```
>Synth12 on TGACv1_scaffold_221301_3B  dna:scaffold:1:198209:1
Length=198209

 Score = 2805 bits (3110),  Expect = 0.0
 Identities = 1559/1560 (99%), Gaps = 1/1560 (0%)
 Strand=Plus/Minus

Query  1       TTATTCAAGAATATTAGTGTTACTAAATAATTATTTTAGTATTTTGAATTTTGGTCAAAT  60
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142977  TTATTCAAGAATATTAGTGTTACTAAATAATTATTTTAGTATTTTGAATTTTGGTCAAAT  142918

Query  61      ATGGTCAAACTGTGGTCAAACTATGGTCAAACTGTGGCCAAACTATGGTCAAACTGTGGT  120
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142917  ATGGTCAAACTGTGGTCAAACTATGGTCAAACTGTGGCCAAACTATGGTCAAACTGTGGT  142858

Query  121     CAAACTATGGTCAAACTACTTATTCAAGAAATATTAGTGTTACTAAATAATTATTATTAT  180
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142857  CAAACTATGGTCAAACTACTTATTCAAGAAATATTAGTGTTACTAAATAATTATTATTAT  142798

Query  181     TTAGAATAATAGTTTTAAACTCAAACAGTGAAACGTGTGACTTCATGCTCAGGCTAAACT  240
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142797  TTAGAATAATAGTTTTAAACTCAAACAGTGAAACGTGTGACTTCATGCTCAGGCTAAACT  142738

Query  241     CCTGGGGGTTAATAGGATTGACATCTTAGTCTTACTATTGTCAGGAAAACAACAAGTGCA  300
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142737  CCTGGGGGTTAATAGGATTGACATCTTAGTCTTACTATTGTCAGGAAAACAACAAGTGCA  142678

Query  301     GACTTGGAAATGAGGGAGAATAGAACCCAGAAGTTAAGCGTGCTCAGGCTGGAGTAGTGA  360
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142677  GACTTGGAAATGAGGGAGAATAGAACCCAGAAGTTAAGCGTGCTCAGGCTGGAGTAGTGA  142618

Query  361     GAGGATGGGTGACCGGCCGAGAAGTTAGATGATTTGGAATGATGAGGGGTGATTAGAGAT  420
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142617  GAGGATGGGTGACCGGCCGAGAAGTTAGATGATTTGGAATGATGAGGGGTGATTAGAGAT  142558

Query  421     TAGAGGTTAAATTAAACAGTGATGAGTGGTGATTAGAGattaaattgtaaaataattcag  480
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142557  TAGAGGTTAAATTAAACAGTGATGAGTGGTGATTAGAGATTAAATTGTAAAATAATTCAG  142498

Query  481     aaatttgaaaattgggaaaaaattaaaaaaaattcaaaaaataaaatttcggaaaatTTG  540
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142497  AAATTTGAAAATTGGGAAAAAATTAAAAAAAATTCAAAAAATAAAATTTCGGAAAATTTG  142438

Query  541     TTACCAACCGGGACTAAAGGTGGAGCTCCAGACAGCGGCCGCGTGGAGGGCCTTTAGTCC  600
               ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  142437  TTACCAACCGGGACTAAAGGTGGAGCTCCAGACAGCGGCCGCGTGGAGGGCCTTTAGTCC  142378
```

- ***ab-initio* prediction**
  Genes are predicted based on gene content and signal detection (e.g. start/stop codon; Ribosome Biding Site (RBS), etc.).



Examples of *ab-initio* tools:

- Glimmer
- GenemarkHMM

- **PRODIGAL**
  PROkaryotic DYnamic programming Gene-finding ALgorithm

  - Predicts protein-coding genes
  - Handles gaps and partial genes
  - Identifies translation initiation sites
  - Handles finished genomes, draft genomes and metagenomes.
  - Runs quickly
  - Runs unsupervised: is an **unsupervised machine learning algorithm**. Automatically learns the properties of the genome from the sequence itself, including RBS motif usage, start codon usage, and coding statistics.
  - Source: https://github.com/hyattpd/Prodigal

Functional Annotation

The process of **attaching biological information to genomic elements** by describing the biochemical and biological function of proteins.

Possible analysis:

- similarity searches;
- gene cluster prediction for secondary metabolites;
- identification of transmembrane domains in protein sequences;
- finding gene ontology terms;
- pathway information.

Functional roles are assigned to coding sequences (CDSs).

Functional Assignment Databases

NCBI Nucleotide and Proteins databases

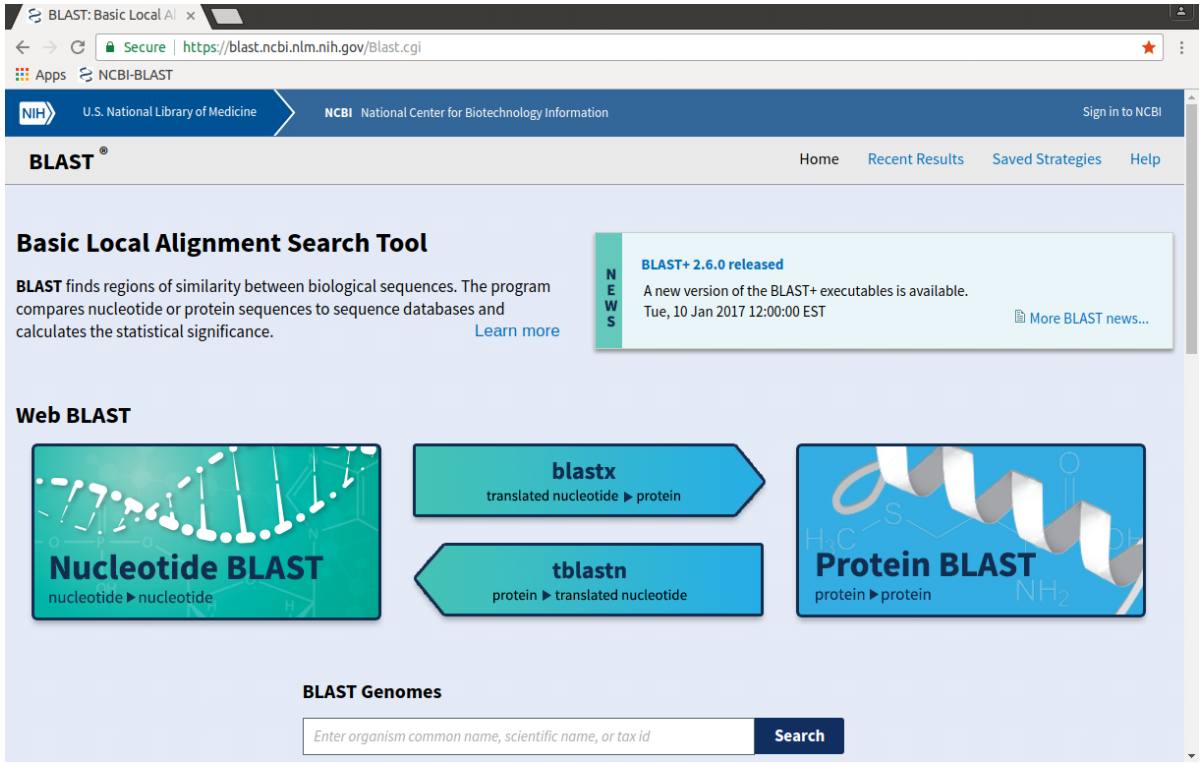
NCBI hosts constantly updated databases of proteins and DNA from several sources that include most of the newly sequenced organisms.

Queries to this database can be performed through the BLAST web server, also hosted by the NCBI.

BLAST can be considered the basic level of annotation for finding similarities.

BLAST (Basic Local Alignment Search Tool)

BLAST finds regions of similarity between biological sequences. The program compares nucleotide or protein sequences to sequence databases and calculates the statistical significance.



Databases

Specialist databases

Some databases offer extra information and search criteria for specific fields of study, while usually including the sequences already in public databases such as NCBI.

Databases

Specialist databases

PFAM | COG | Kegg | CAZymes

Pfam is a large collection of protein families, represented by multiple sequence alignments and hidden Markov models (HMMs).
Source: http://pfam.xfam.org/

Databases

Specialist databases

PFAM | COG | Kegg | CAZymes

COG

Clusters of Orthologous Groups (COGs)

Database of proteins generated by comparing the protein sequences of complete genomes.
Each COG consists of individual orthologous proteins or orthologous sets of paralogs from at least 3 lineages.
Orthologs typically have the same function, allowing transfer of functional information from one member to an entire COG.
Source: https://www.ncbi.nlm.nih.gov/research/cog-project/

Databases

Specialist databases

PFAM | COG | **Kegg** | CAZymes

Kyoto Encyclopedia of Genes and Genomes

Database resource for understanding high-level functions and utilities of biological systems, such as the cell, the organism and the ecosystem, from molecular-level information.
Source: https://www.genome.jp/kegg/

Databases

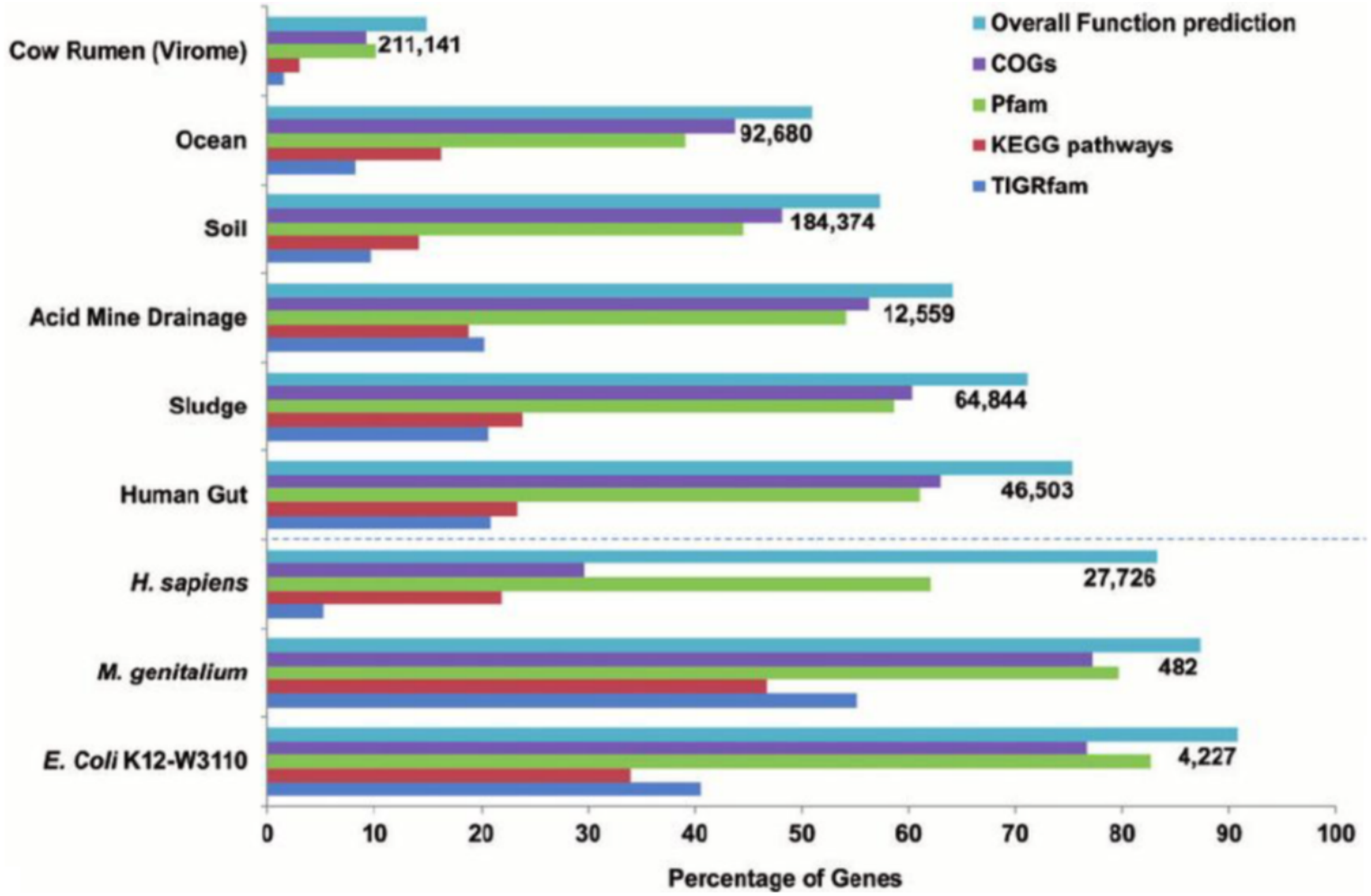
Specialist databases

PFAM | COG | Kegg | CAZymes

Carbohydrate-Active enZymes

Enzymes involved in the synthesis, metabolism, and transport of carbohydrates.
CAZymes are organized in families in the continuously updated database CAZy.

Includes: glycoside hydrolases (GHs), glycosyltransferases (GTs), polysaccharide lyases (PLs), carbohydrate esterases (CEs) and carbohydrate binding modules (CBMs).
Source: http://www.cazy.org/



Prakash, Tulika & Taylor, Todd. (2012). Functional assignment of metagenomic data: Challenges and applications. Briefings in bioinformatics. 10.1093/bib/bbs033. (link)

Functional assignment workflows

- Prokaryotic Genomes Automatic Annotation Pipeline (via NCBI)
- InterPro
- Prokka
  Coomand-line tool.Also integrated into K-Base.
- RAST
  A web server for annotating bacterial and archaeal genomes that provides annotation results in under a day

InterPro

Initial quality control

Identification of rRNA reads

Binning of 16S rRNA gene reads

QIIME-based taxonomic assignment

Non-RNA reads: ORF prediction

Functional annotation with
InterPro (IPR) Scan

Integrates several other databases
Updated every 8 weeks

PROKKA

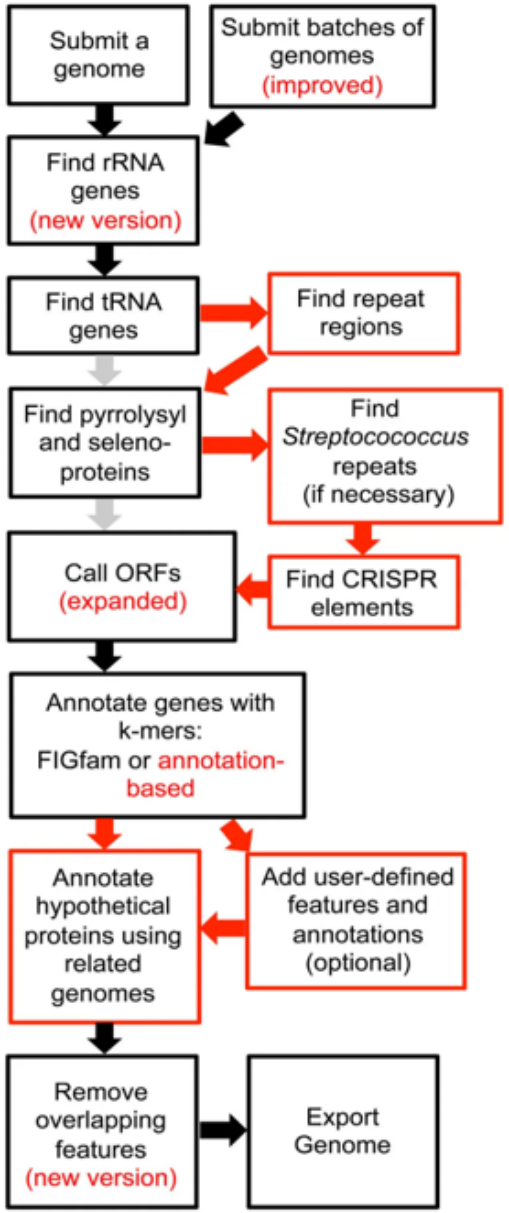
The approach:

Prokka uses a **variety of databases** to **assign function to predicted CDS features**.
It takes a hierarchical approach to make it fast.

Proteins coding genes are annotated in two stages:

> **1) Prodigal** identifies the coordinates of candidate genes, but does not describe the putative gene product.
>
> **2.1)** A small, core set of well characterized proteins are first searched using **BLAST+**. This combination of small database and fast search typically completes about 70% of the workload.
>
> **2.2)** A series of slower but more sensitive HMM databases are searched using **HMMER3**.

Tutorial: https://training.galaxyproject.org/training-material/topics/genome-annotation/tutorials/annotation-with-prokka/tutorial.html



Rast

The new RAST-tk annotation pipeline

Very unique “Package” which works with its own database (SEED) and anotation scheme (“subsystems technology”).

Take-home messages: Genome Annotation

What?

Attributes functions to genes predicted from genomes or metagenomes

Why?

To answer the “What do they do?” question

How?

Through the pipeline ORF/Gene calling – Database search – functional analytics

Take-home messages: Genome Annotation

Ideally: Annotation of **all available/acquirable genes** to provide functional overview and hints on possible novel genes

The annotation accuracy is only as good as the available supporting data!

As new data becomes available, gene predictions and functional assignments will change.
Ex: Characterization of hypotetical proteins

Introduction to File Formats

FASTA GFF3 GENBANK

DNA and protein sequences can be written in FASTA format. First line:">" followed by the description. In the second line the sequence starts.

```
>gi|528476558|ref|NC_018928.2| Homo sapiens chromosome 17
TCCACTCACTGAGACAATAGACCCAAGCACATCAGCTCTGAGGCCTC
AGCACATCAGCTCTGAGGCCTCCACTCACTGAGACAATAGACCCAAG
TCACTGAGACAATAGACCCAAGCACATCAGCTCTGAGGCCTCCACTC
AACAGCTCTGAGGGCTCCACTCACTGAGACAACAGACCCAAGAACAA
>gi|528476670|ref|NC_018912.2| Homo sapiens chromosome 1
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
CCTAACCCTAACCCTAACCCTAACCCTAACCCTAACCCTAACCCTAACCCTAACCCTAA
ACCCTAACCCTAACCCTAACCCTAACCCAACCCTAACCCTAACCCTAACCCTAACCCTA
TAACCCTAACCCTAAACCCTAAACCCTAACCCTAACCCTAACCCTAACCCTAACCCCAA
```

Introduction to File Formats

FASTA | GFF3 | GENBANK

General feature format (gene-finding format, generic feature format, GFF) is a file format used for describing genes and other features of DNA, RNA and protein sequences.

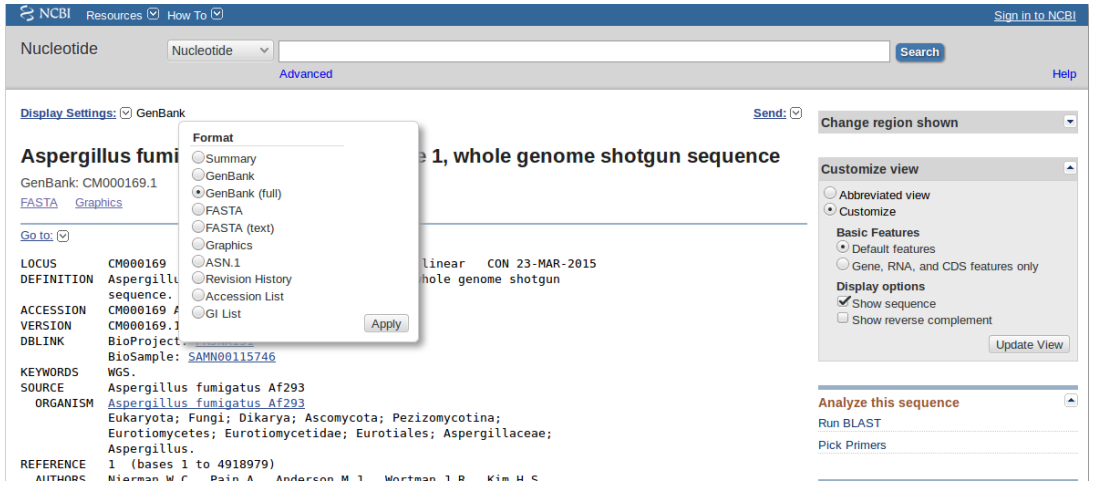
```
# start gene CM000169.1.g1
```

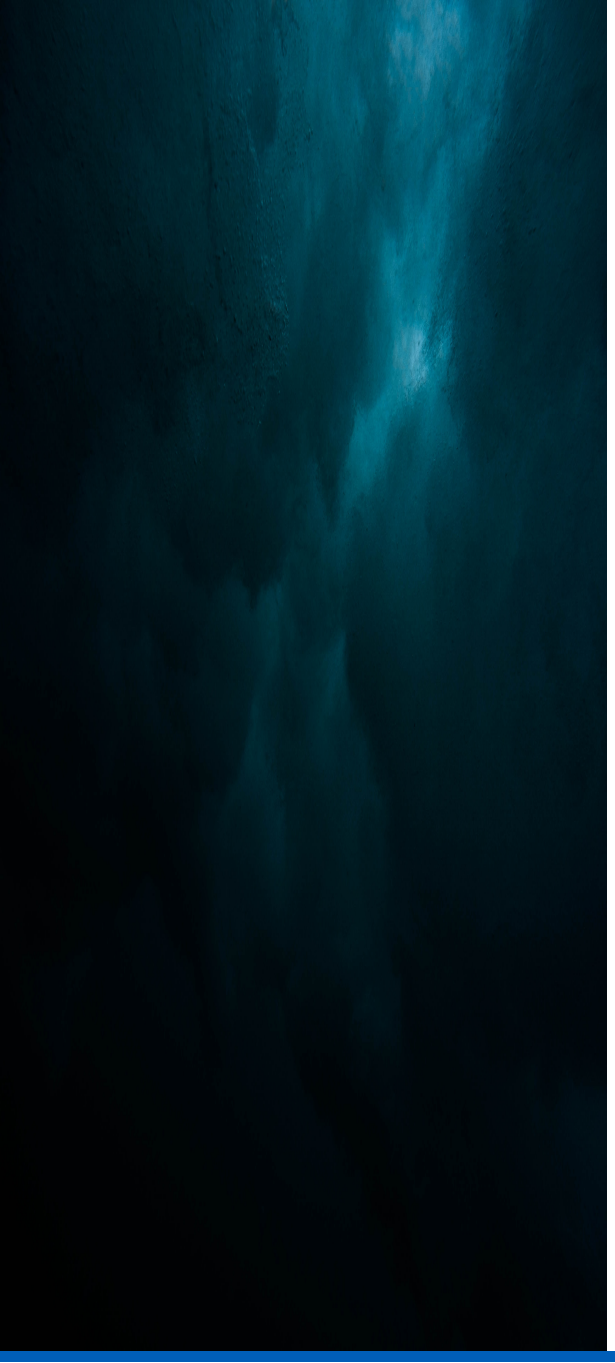
CM000169.1	AUGUSTUS	gene	3168	4292	0.99	+	.	ID=CM000169.1.g1
CM000169.1	AUGUSTUS	transcript	3168	4292	0.99	+	.	ID=CM000169.1.g1.t1;Parent=CM000169.1.g1
CM000169.1	AUGUSTUS	start_codon	3168	3170	.	+	0	Parent=CM000169.1.g1.t1
CM000169.1	AUGUSTUS	intron	3339	3422	1	+	.	Parent=CM000169.1.g1.t1
CM000169.1	AUGUSTUS	CDS	3168	3338	0.99	+	0	ID=CM000169.1.g1.t1.cds;Parent=CM000169.1.g1.t1
CM000169.1	AUGUSTUS	CDS	3423	4289	1	+	0	ID=CM000169.1.g1.t1.cds;Parent=CM000169.1.g1.t1
CM000169.1	AUGUSTUS	stop_codon	4290	4292	.	+	0	Parent=CM000169.1.g1.t1

Introduction to File Formats

FASTA GFF3 GENBANK

The genbank sequence format is a rich format for storing sequences and associated annotations.





Hands-on 1

Genome annotation with the Clusters of Orthologous Genes (COG) Database

Worflow

Genome annotation with Clusters of Orthologous Genes (COG) Databa

K-Base

1) Download bins from K-Base in fasta format

Rast

2) Upload bins in Rast
3) Perform gene annotation
4) Download fasta aminoacid files

WebMGA

5) Upload downloaded bins in WebMGA
6) Perform COG annotation
7) Download COG annotation from WebMGA

R

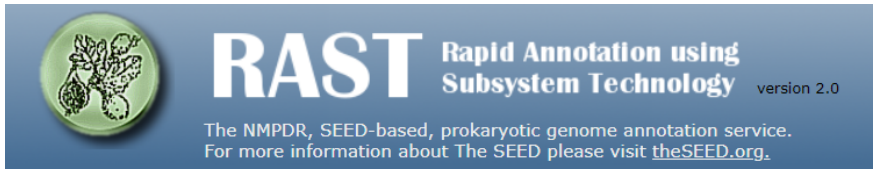
8) Join COG annotation into single table - R script

K-Base

1) Download bins from K-Base in fasta format;

Link

Rast



2) Upload bins (fasta files) in Rast;

3) Perform genome annotation;

4) Download fasta aminoacid files;

Link

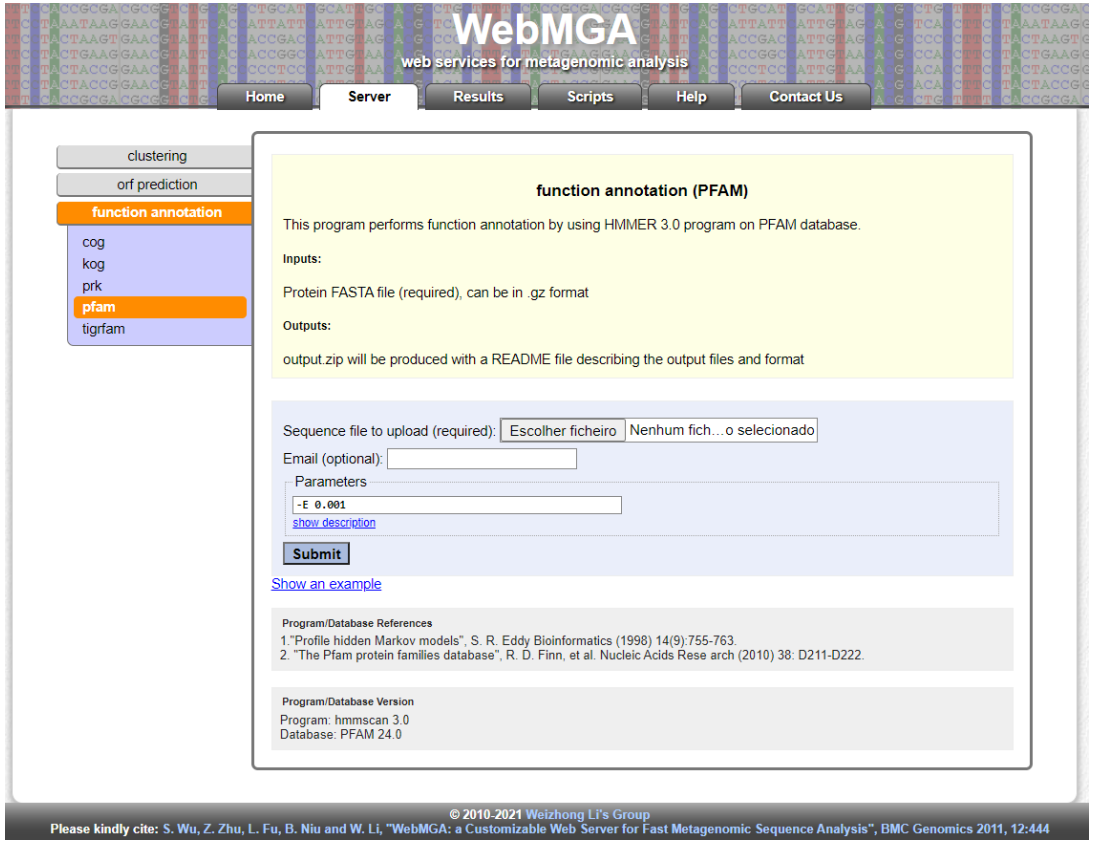
WebMGA

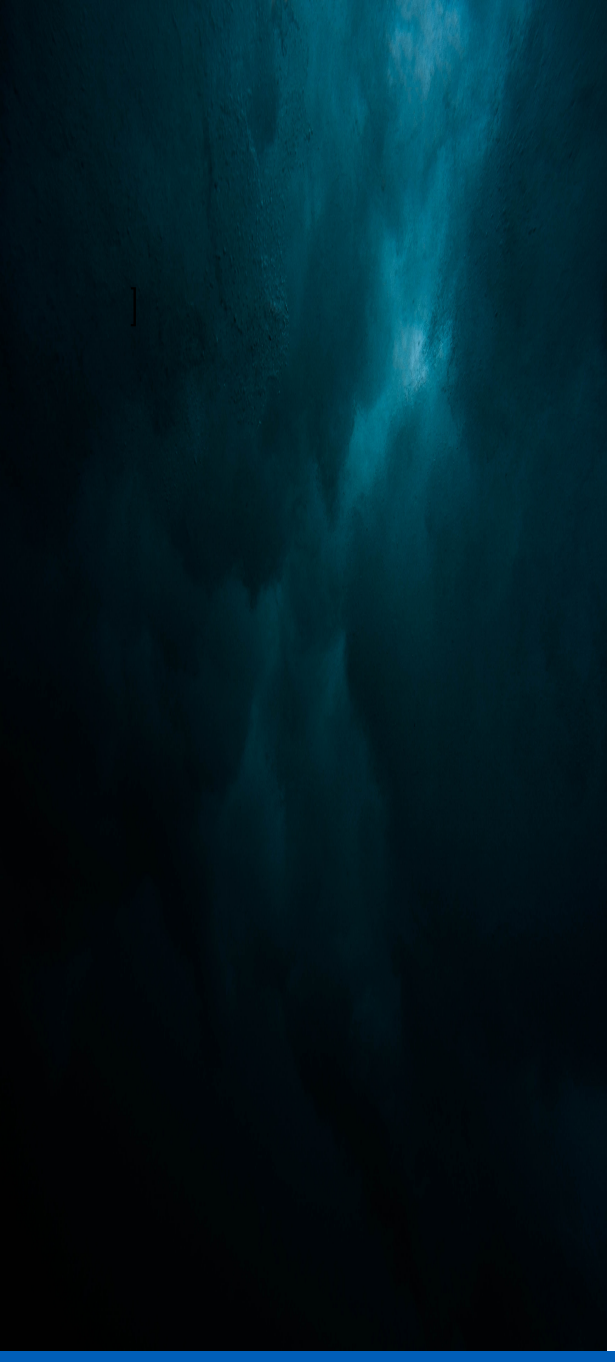
5) Upload genomes (fasta aminoacid files) in WebMGA;

6) Perform COG annotation;

7) Download COG annotation from WebMGA.

Link





Hands-on 1.2

1.2.1 Install R and RStudio

Link: 1.2.1 R setup

1.2.2 Introduction to R and RStudio

Link: Getting started with R

1.2.3 Run R script to join COG annotation files

Workflow to join COG annotation in a single file

For each bin:

> 1) Download COG annotation from WebMGA (will be a zipped file)
> 2) Unzip file;
> 3) Look for the .txt file named "cog";
> 4) Rename this file with the bin's name (Ex: cog_SAMPLEX_bin1)

For all bins:

> 5) Move all cog annotation files to the same folder;
> 6) Download R script and move it to this folder;
> 7) Open all cog annotation files and delete the character "#" from the first line;
> 8) Run R script.

Final output: COG table with annotations for all bins

Download script: Script_Merge_COGs.R
Note R extension!

That is all for today!

Any questions?

Thank you!

Here's where you can find me...

sandragodinhosilva.netlify.com
@SandraGodSilva

Slides created via the R package **xaringan**