Exercises

1) What is the difference between Neural networks and Deep learning? Multiple hidden layers and hierarchy in deep learning
2) What is the advantage of having a hidden layer in deep learning? Combining or transforming the raw data without specifying how
3) Machine learning is based on features (variables); if you can find and extract the right combination of features from the data there is no advantage to deep learning over other machine learning methods such as random forest, SVM, etc. [T/F] True, the advantage of deep learning is learning features.
4) What does a convolution network enable a model to capture? enables one to capture hierarchical patterns in data, e.g. images [local to global], sequences [local to global]

Today’s objectives

- Bioinformatic Pipelines and Databases
- Microbiome and Metagenomics
- BioPython
- Database API

Bioinformatics vs. Computational Biology

Bioinformatics (emphasis on practical application)

- the collection, classification, storage, and analysis of biological data using computers
- develops methods and software tools for understanding biological data

Computational Biology (emphasis on theory)

- development and application of analysis methods, theoretical and mathematical models, and computational simulation techniques to study biological systems
- develops methods and models for understanding biological systems

Bioinformatics: Pipelines & Databases

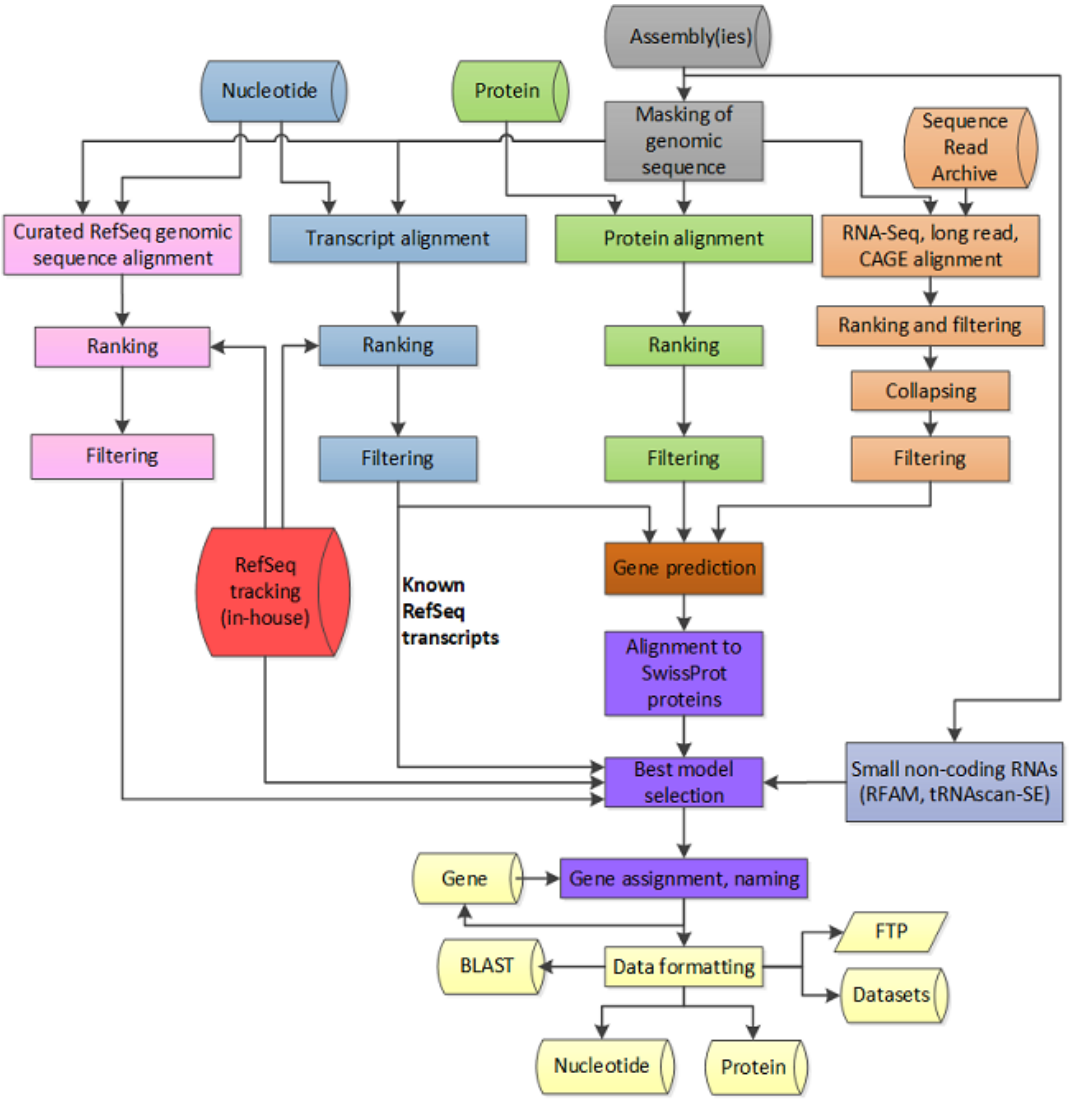
A bioinformatic pipeline is a computational workflow used to analyze data, often in parallel

A bioinformatic database is collection or repository of data in a standardized format conducive to bulk download or individual queries, often publicly available.

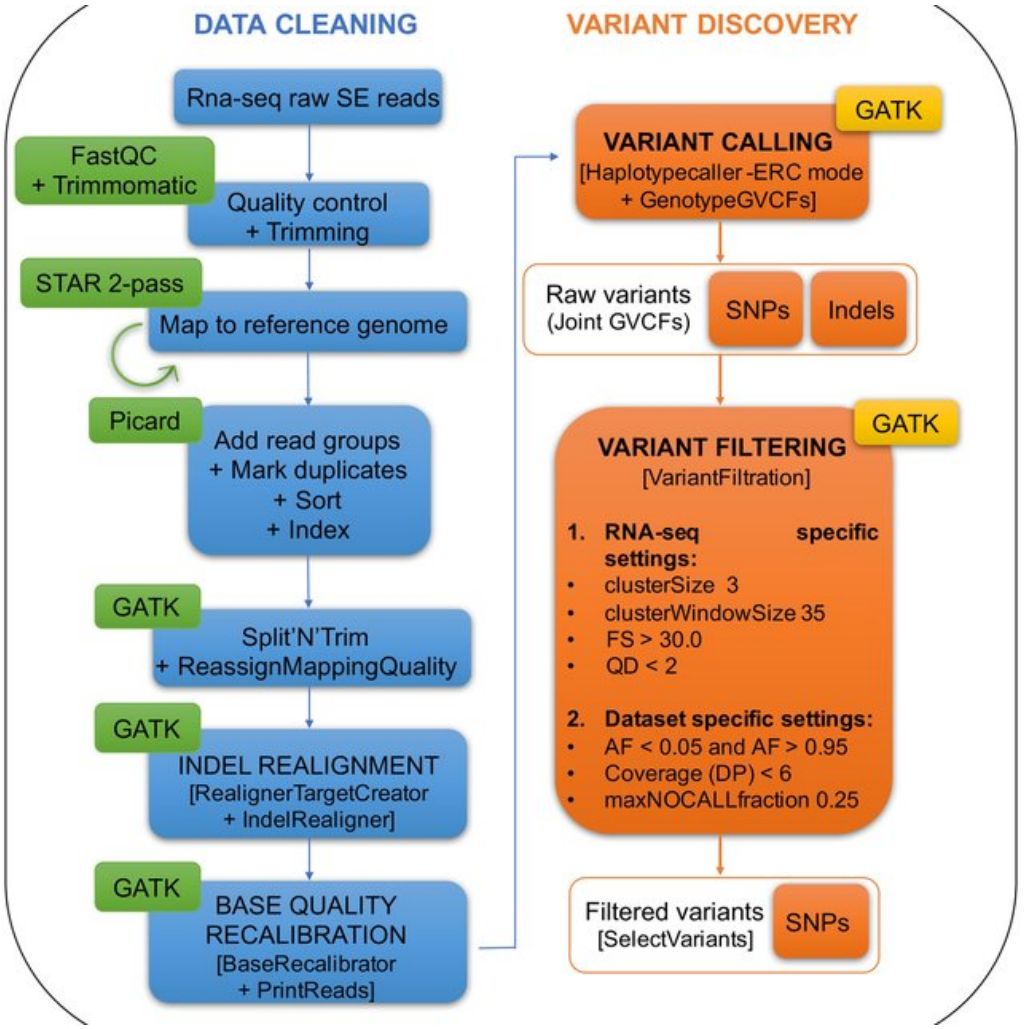
- Pipelines put together the many steps needed to process raw data to specific results
- Pipelines often stitch together different (software) tools
- Pipelines can require input from databases of public reference data

Examples

Gene annotation



Variant calling



Bioinformatic Pipelines

1) Pipeline testing and development (performance) can be more important than the performance of the tools used.

- normalization
- quality control (QC)
- software parameters

2) Automates the processing of data, enables

- parallel processing of many datasets
- replication of data analysis [methods]
- bootstrapping and permutation
  - bootstrapping is random sampling of data with replacement to estimate confidence
  - permutation is randomizing one aspect of the data to form an appropriate null distribution

A simple example

Data: whole genome sequencing (WGS) of x individuals
Goal: identify DNA sequence variants in comparison to a reference genome

For each individual:

Step 1: map reads to the reference genome
Step 2: call variants based on differences

Step 1

```
bwa mem f1.fastq >f1.sam
bwa mem f2.fastq >f2.sam
bwa mem f3.fastq >f3.sam
```

Bash and Python scripts

Execute commands from within a bash or python script

```
for i in *fastq
do
    o=${i/fastq/sam}
    bwa mem $i >$o
done
```

```
import os, re
for f in os.listdir(directory):
    if filename.endswith("fastq"):
        o = re.sub('fastq','sam',f)
        cmd = 'bwa mem ' + f + '>' + o
        os.popen(cmd)
```

Write commands to a file using a bash or python script

```
for i in *fastq
do
    o=${i/fastq/sam}
    echo "bwa mem $i >$o"
done >mapping.sh
```

```
import os, re
for f in os.listdir(directory):
    if filename.endswith("fastq"):
        o = re.sub('fastq','sam',f)
        cmd = 'bwa mem ' + f + '>' + o
        file.write(cmd)
```

```
bwa mem f1.fastq >f1.sam
bwa mem f2.fastq >f2.sam
bwa mem f3.fastq >f3.sam
```

Multi-line commands and pipes

Bash escape character backslash (\), preserves the literal value of the next character that follows, with the exception of newline. If a \newline appears, the \newline is treated as a line continuation (that is, it is removed from the input stream and effectively ignored).

```
samtools sort \
    -n \
    -O bam \
    -o f1.bam \
    f1.sam
samtools sort -n -O bam -o f1.bam f1.sam
```

Bash pipes (|) let you use output of a program as the input of another one. This eliminates intermediate files and saves space.
- useful for file conversion, file compression, etc

```
bwa mem f1.fastq | samtools view -b >f1.bam
samtools mem f1.fastq >f1.sam
samtools view -b f1.sam >f1.bam
```

Microbiome pipeline

Metagenomic samples: mixtures of different species

extraction of DNA

High-throughput sequencing

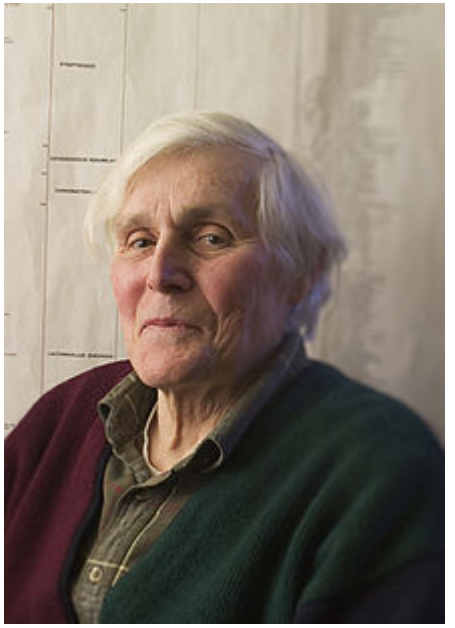
Microbiome pipeline

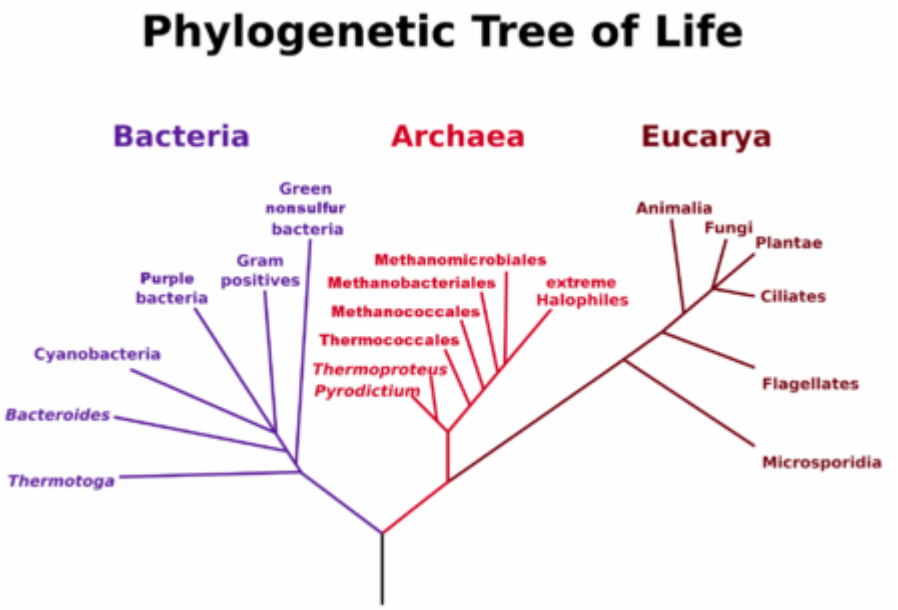
Who is present and at what abundance

- Counts (sample x species)

Archae

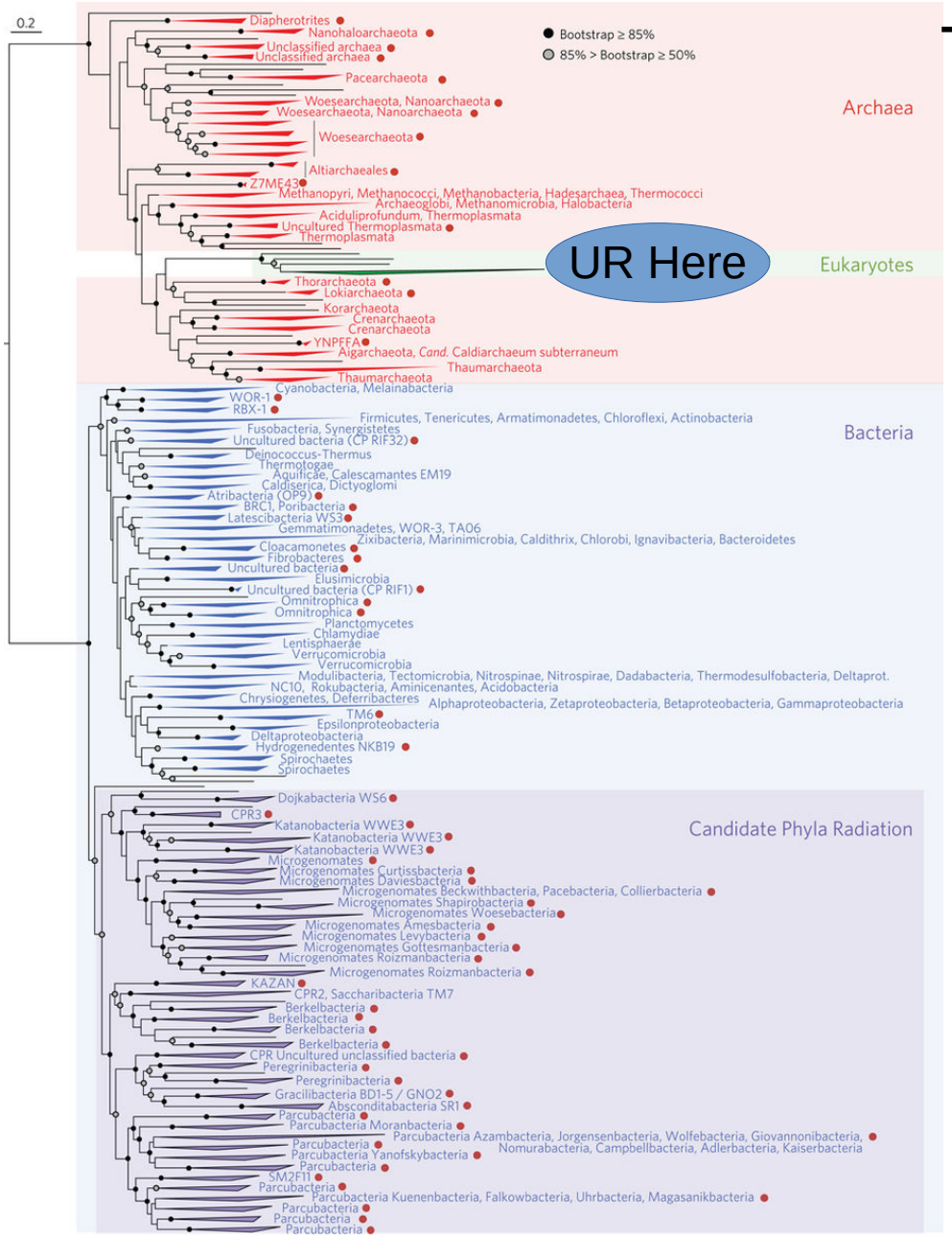
Carl Woese: Defined Archaea (a new domain of life) in 1977 by phylogenetic taxonomy of 16S ribosomal RNA.
Prokaryotes-lacks a membrane-bound nucleus, mitochondria, or any other membrane-bound organelle
Eurkaryotes-have a cell nucleus and other organelles enclosed by membranes



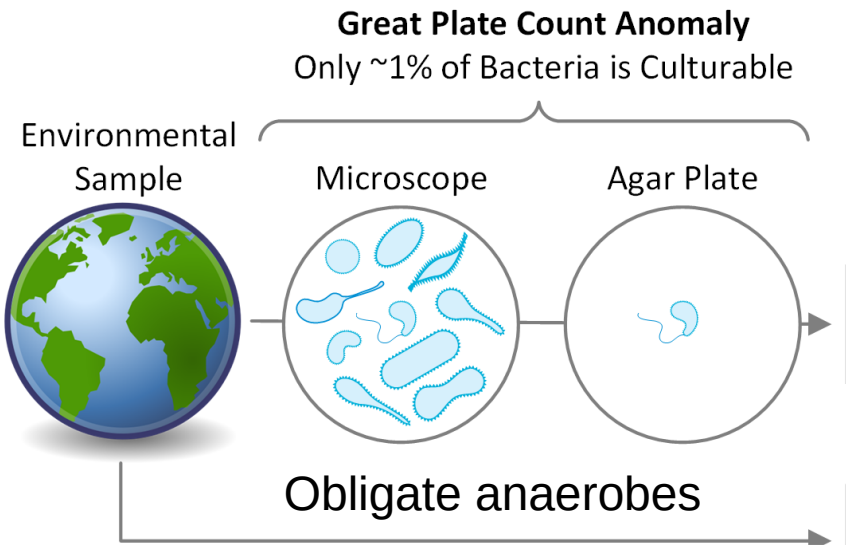


How did we miss an entire domain?

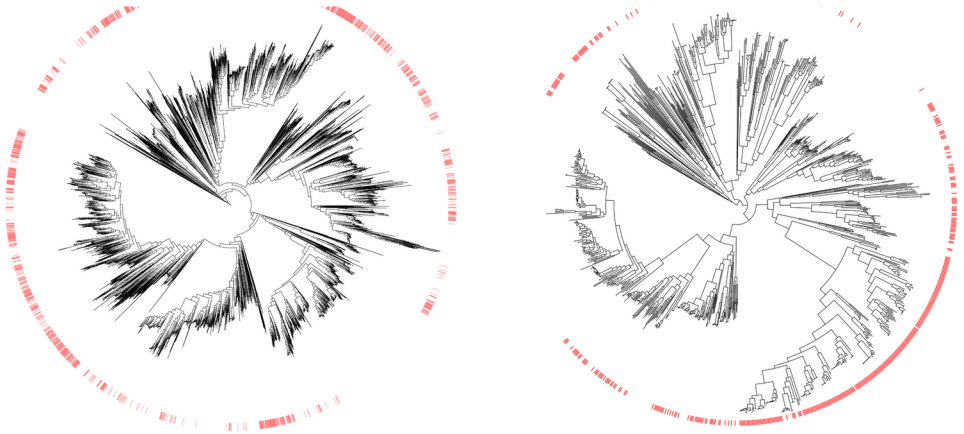
Microbial diversity vs culturable



The Great Plate Count Anomaly



8,000 Genomes from uncultivated sequencing
30% increase



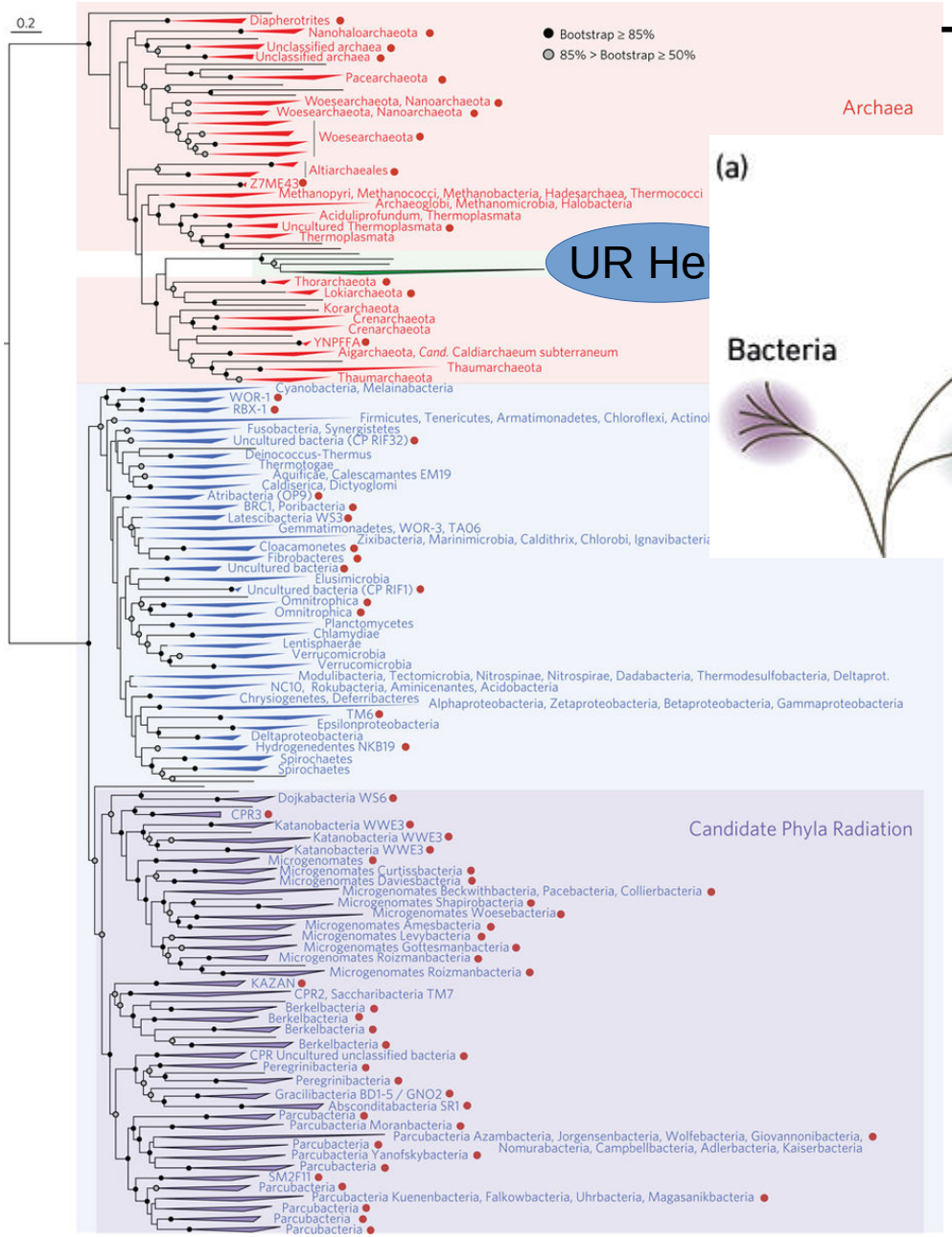
7280 bacterial

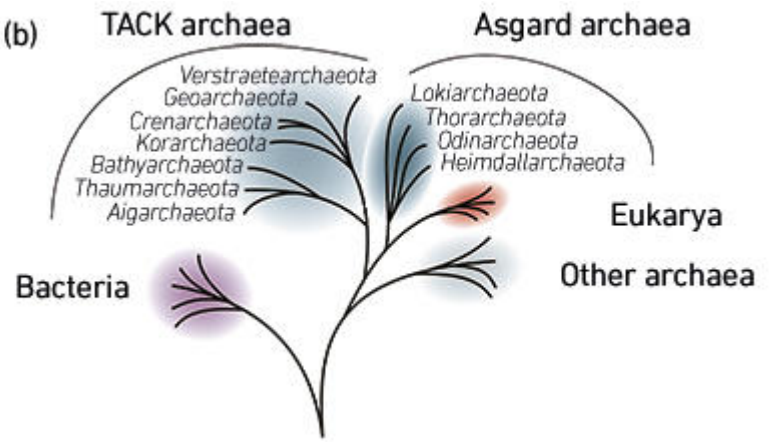
623 archeal

Parks et al. 2017

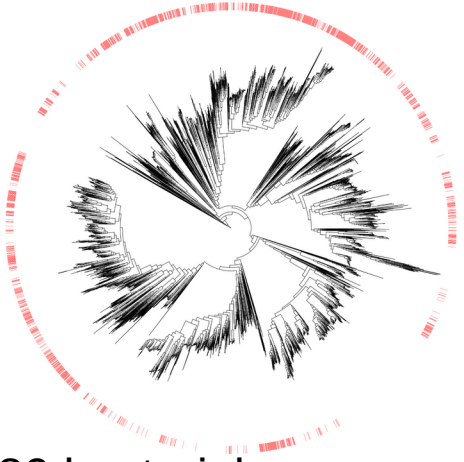
Microbial diversity vs culturable

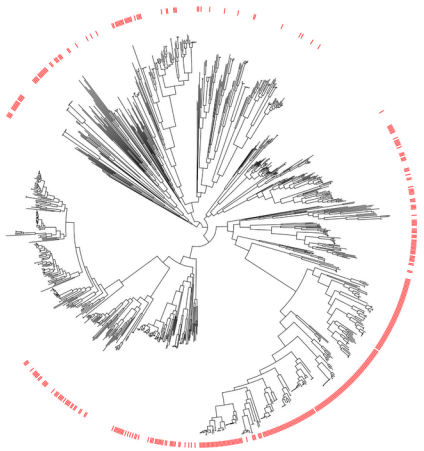
The Great Plate Count Anomaly





8,000 Genomes from uncultivated sequencing
Red = new genomes





7280 bacterial

623 archeal

Parks et al. 2017

What do we know about microbes

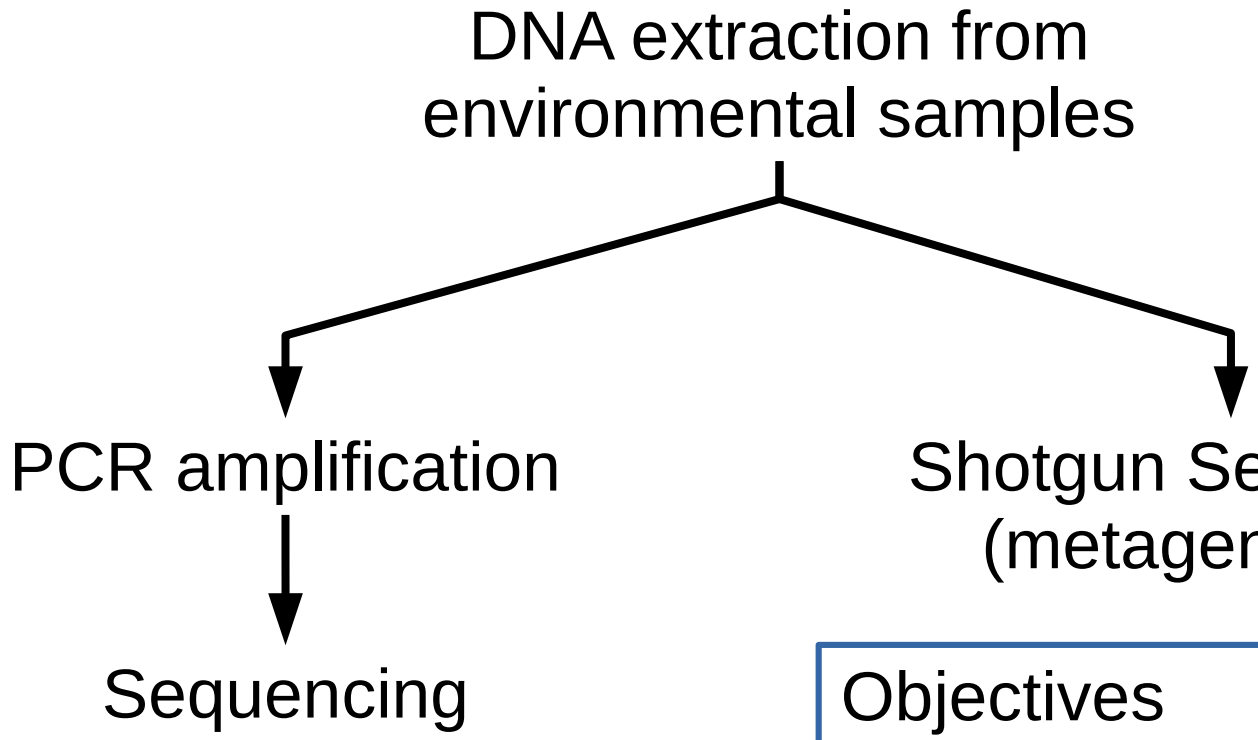
- First organisms on earth, most diversity
- Extremophiles, many are archae, found in all environments, deep underground, near boiling temperatures, arctic ice, etc. Mars?
- Play essential role in decomposition and carbon cycle
- Important roles in plant (root) and animal (gut) nutrient acquisition.
- Bacteria, Eukaryote (fungi), Archae, Viruses
- Microbiome: the community of organisms that inhabit a particular environment

Human microbiome in health and disease

Human microbiome

- 100 trillion microbes in the intestine (more than 10x human cells)
- 3 million genes
- 2 kg weight
- 300-1000 species
- depends on age, diet, geography
- associated with human health (e.g. preterm birth)

Overview of microbiome workflow



Who is there?
How many?
Do what?

Computational analysis
- hit (known species)
- alignment
- tree

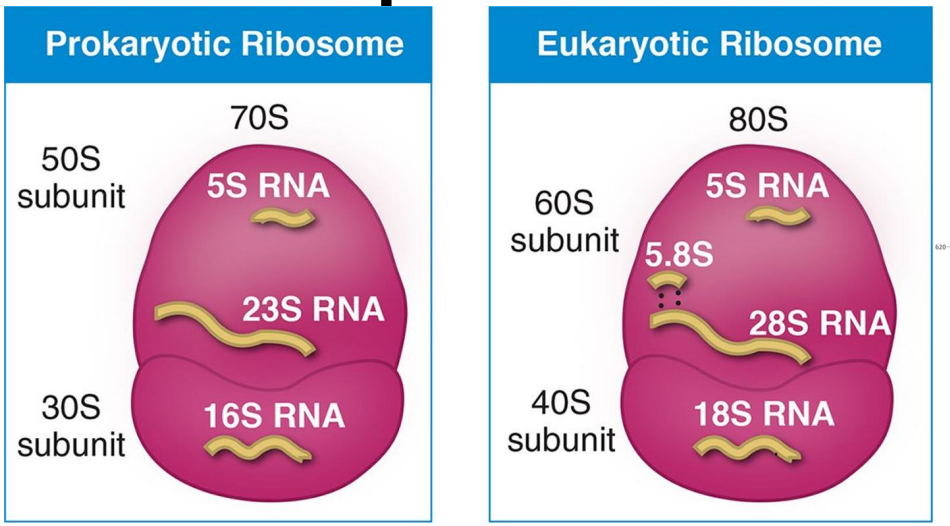
Objectives
- Placing organisms into the tree of life
- Describing composition and diversity of a community
- Relating communities to one another

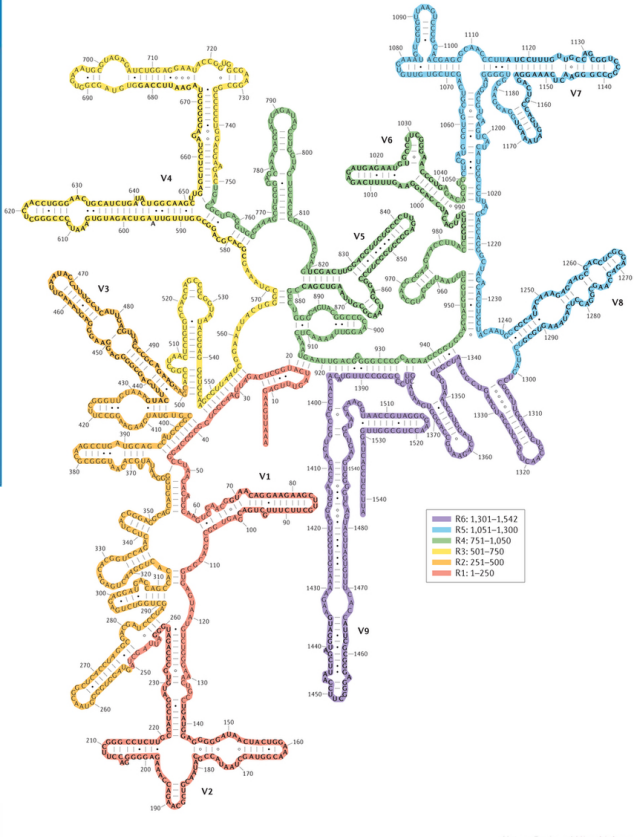
rRNA – the lens into life
Step 1: PCR

Eukaryotes
- 18s
- ITS

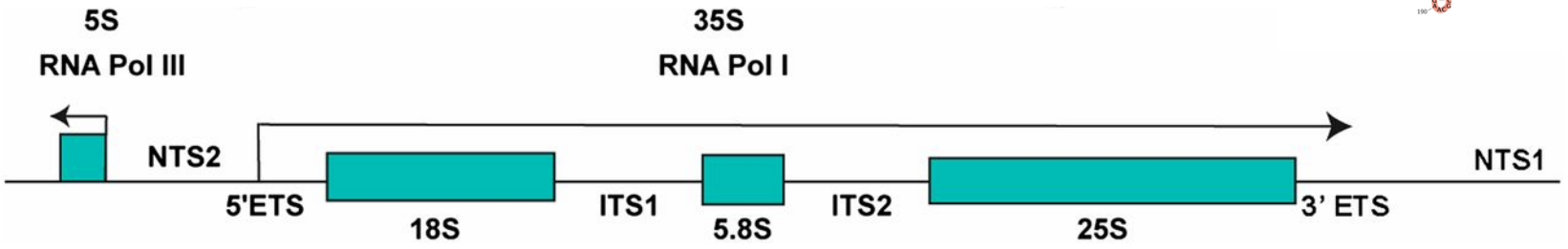
Prokaryotes
- 16s

Viruses
- Gp23
- RdRp





RNA-pairing → Exceptional conservation



OTU table: presence + abundance

An operational taxonomic unit (OTU) is an operational definition used to classify groups of closely related individuals. Typically by sequence similarity (<97%).

	Sample 1	Sample 2	Sample 3
OTU 1	4	0	2
OTU 2	1	0	0
OTU 3	2	4	2

OTU picking

An operational taxonomic unit (OTU) is an operational definition used to classify groups of closely related individuals.

- OTU is often defined by 97% identity to account for population variation + error

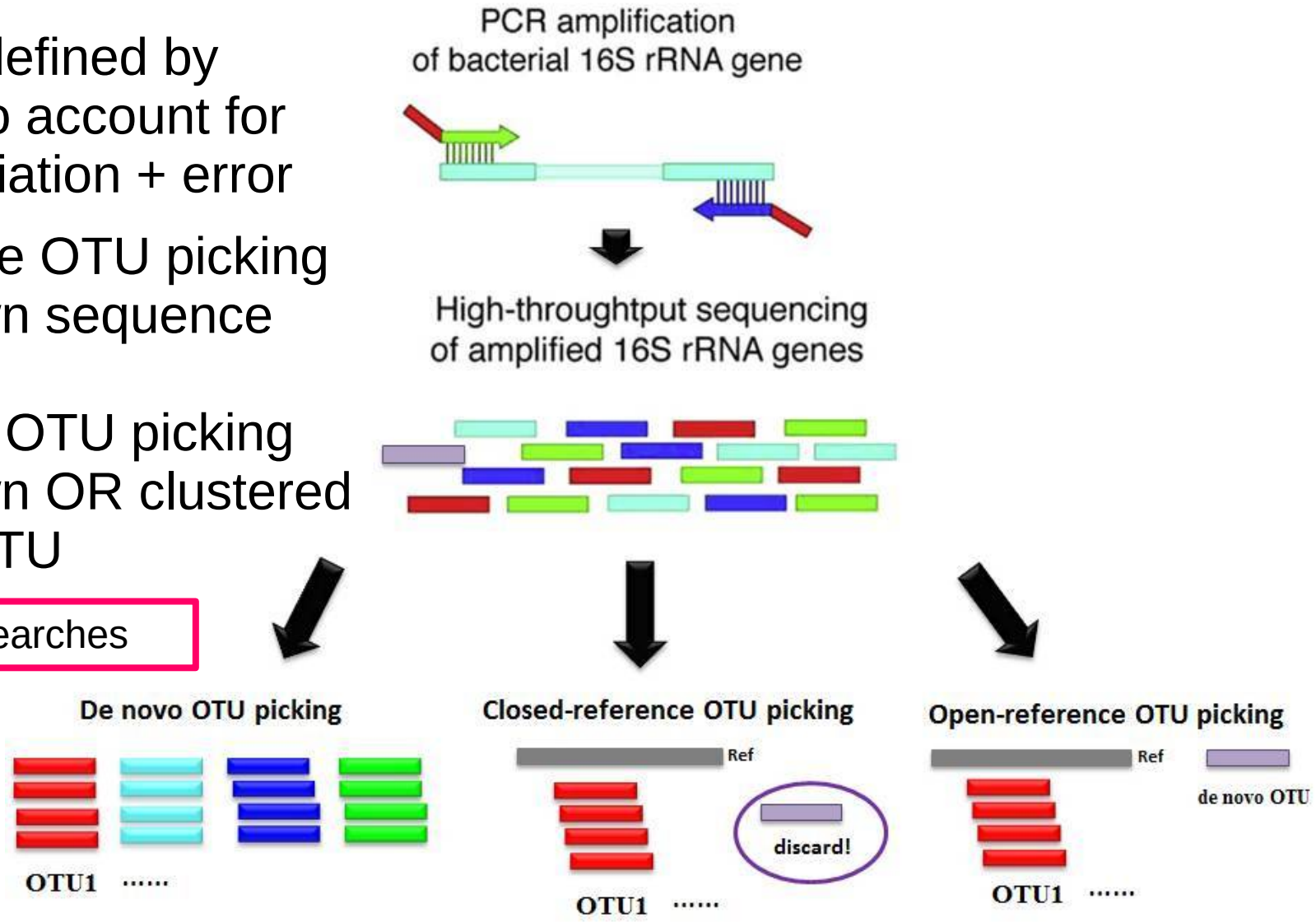
Closed reference OTU picking

- match to known sequence database

Open reference OTU picking

- match to known OR clustered to form new OTU

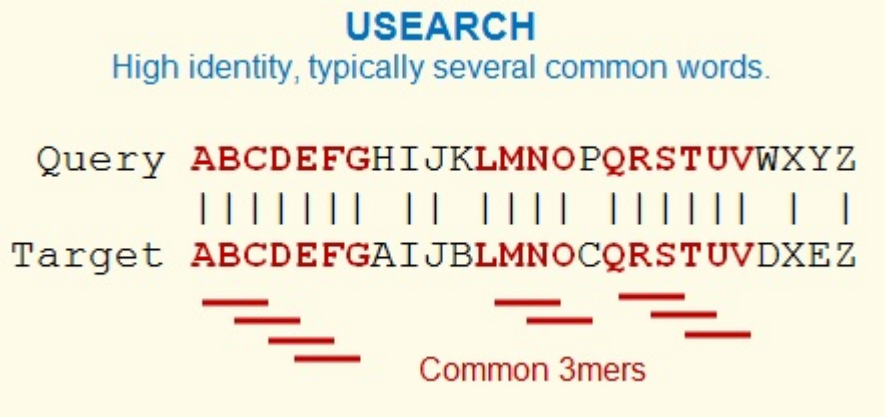
Lots of sequence searches



Computational Challenges memory and speed

- millions of reads (100-400 bp)
- thousands of reference sequences (database)

- Challenge #1 OTU picking
- Challenge #2 align all
- Challenge #3 trees



Ultrafast search (hash and extend too slow)

USEARCH and CD-hit

- kmers (3mers)

- query to database in order of decreasing Unique kmer counts

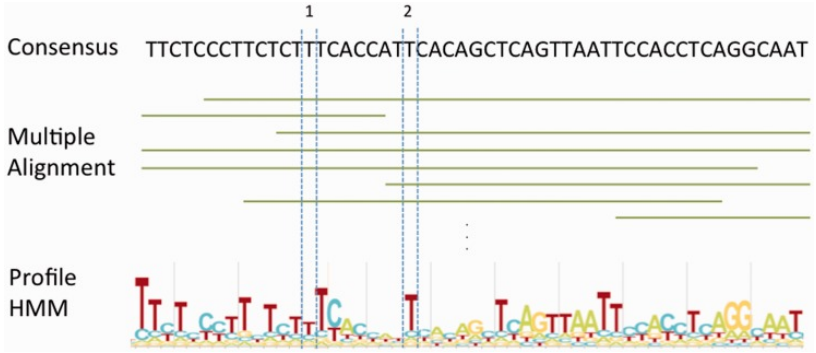
- estimate that the similarity of two sequences by simple word counting and without an actual sequence alignment

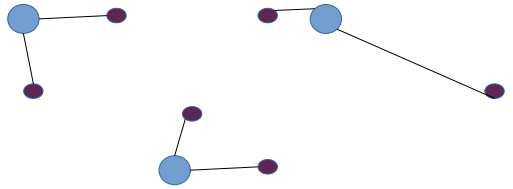
- first hit is likely best, doesn't work for low similarity

Computational Challenges memory and speed

- millions of reads
- thousands of reference sequences

- Challenge #1 OTU picking
- Challenge #2 align all
- Challenge #3 trees





Ultrafast alignment

Guide tree $O(N^2)$ limits to < 10k, and memory ~40GB for 100k

Clustal omega (progressive)

- $O(N(\log(N))^2)$ using pairwise distance to random seeds, pairwise distances are clustered, guide tree made within groups and then between groups

- profile-profile alignments (HMM) ~ O(N)

Computational Challenges memory and speed

- millions of reads
- thousands of reference sequences

- Challenge #1 OTU picking
- Challenge #2 align all
- Challenge #3 trees

FastTree2
237,882 distinct 16S ribosomal RNAs on a desktop computer in 22 hours and 5.8 gigabytes of memory.
RaxML
1000 sequences in 1 day

Ultrafast trees

Neighbor-Joining considers N^2 joins at each step so is $O(N^3)$

RAxML (maximum likelihood)

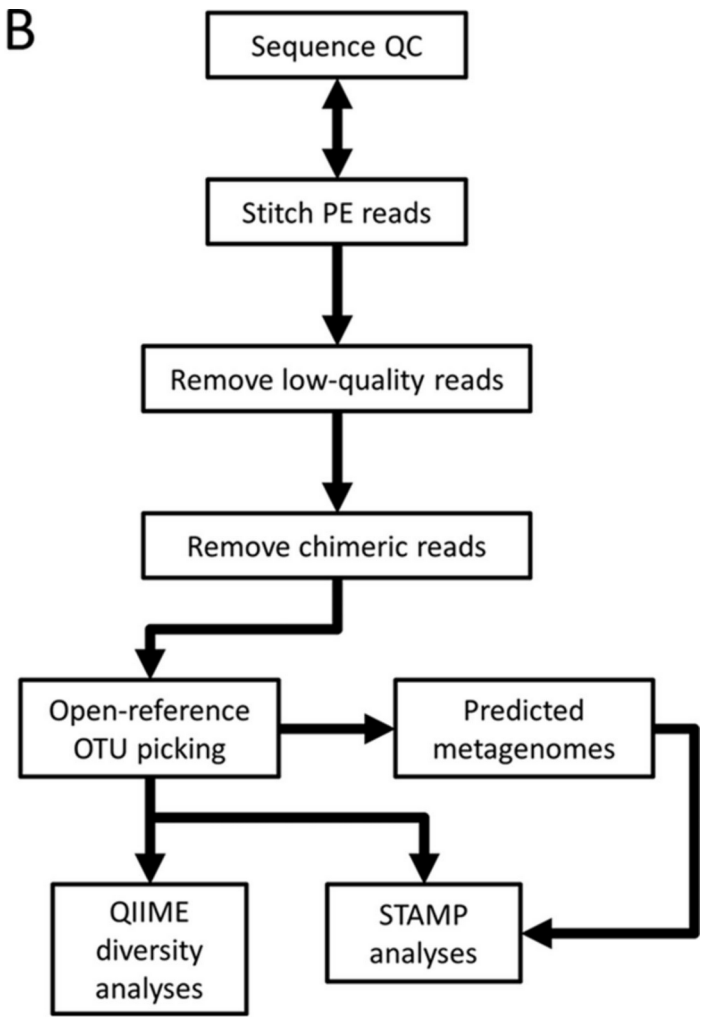
FastTree

- NJ with stored profiles

- heuristic to reduce joins

- O(NLa + N sqrt(N)) memory and O(N sqrt(N) log (N)La) time, L = length and a = characters

Microbiome analysis pipeline



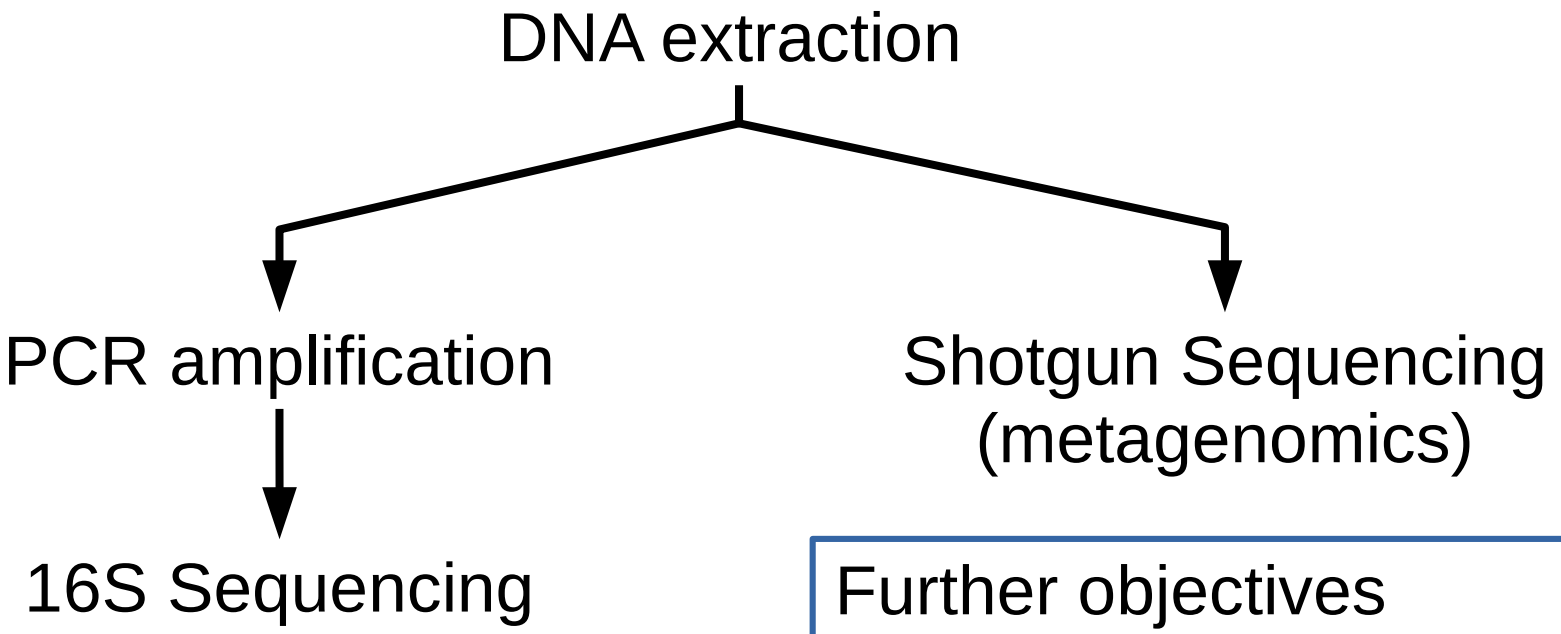
Problem

Microbiome analysis pipeline
- many individual steps
- each step based on software/tool
- sometimes compare/use multiple tools/step
- repeated for each sample

Solution

Bash scripting and/or BioPython
- automate commands to run each tool
- pipe output of one tool to another
- run each sample in parallel
- easy to modify and rerun analysis

Overview of workflow



Microbiome profiles
- hit (known species)
- alignment
- tree
- prevalence/abundance

Further objectives
- What is microbe X doing, how does it make a living?
- What is the community doing?
- Community → function

16s vs shotgun metagenomic

	16s	Metagenomic
Sequences	just bacteria	virus, fungal, bacteria, host (up to 98% of reads)
Species ID	Reference databases (e.g. RPD)	Reference genomes
PCR	primer bias (species drop out)	No PCR bias
Resolution	Species or genera (high coverage)	strain level resolution (lower coverage)
Genes/ Species	New species found	gene sequences → function (GO terms)

Metagenomic Species Reference Database

Table 5. Number of entries in commonly used reference databases

Domain	Level	Draft genomes		Complete genomes[1]	
		GenBank	RefSeq	GenBank	RefSeq
Archaea	Entries	859	351	260 (20)	225 (12)
	Species	695	204	209 (14)	178 (7)
Bacteria	Entries	89 730	78 783	7314 (1346)	6973 (1066)
	Species	19 078	11 217	2677 (542)	2586 (406)
Fungi	Entries	1897	191	28 (414)	7 (38)
	Species	997	190	17 (68)	7 (36)
Protists	Entries	430	47	2 (49)	2 (27)
	Species	226	47	2 (38)	2 (26)
Viruses	Entries	3	3	0 (0)	7214 (22)
	Species	1	3	0 (0)	7073 (22)

[1]Numbers in parentheses represent incomplete genome assemblies for which at least one chromosome was assembled. Data as of 27 May 2017.

INSDC, International Nucleotide Sequence Database Collaboration: DNA Databank of Japan (DDBJ), European Molecular Biology Laboratory-European Bioinformatics Institute (EMBL-EBI), National Center for Biotechnology Information (NCBI)

GenBank includes publicly available DNA sequences submitted from individual laboratories and large-scale sequencing projects. GenBank can be very redundant for some loci. GenBank sequence records are owned by the original submitter and cannot be altered by a third party.

RefSeq sequences are not part of the INSDC but are derived from INSDC sequences to provide non-redundant curated data representing our current knowledge of known genes. RefSeq records are owned by NCBI and therefore can be updated as needed to maintain current annotation or to incorporate additional information.

Number of species with genome sequences is large, but much smaller than the number of 16s sequences.

Metagenomic Species ID algorithms

Metaphlan
1 million unique, clade-specific markers, 7,500 species

classify all reads

Homology: Blast, MEGAN

Composition (kmer): Kraken, CLARK, Naive Bayes

Hybrid: PhymmBL, PhyloPythia

PhyloSift
Database of 37 universal proteins, rRNA genes, classify with phylogenies

classify subset

Phylogenetic: TreePhyler (pfam)

Markers: MetaPhlan, PhyloSift

What about a new or unknown species?

Databases and tools

BioPython

API

Options

- Database download
- Database API
- Database construction/modification

Biopython

The goal of the Biopython project is to create high-quality, reusable modules and classes for computational biology.

- parse bioinformatics files into Python utilizable data structures: clustalw, fastq, genbank files
- interact with popular on-line bioinformatics databases: Ensembl, NCBI
- interface with commonly used software: clustalw, EMBOSS, BLAST

Scripts are 90% about IO

Why Biopython

Parsing commonly used formats, e.g. GenBank

Interface with databases and software, NCBI

Both accomplished with defined classes and methods

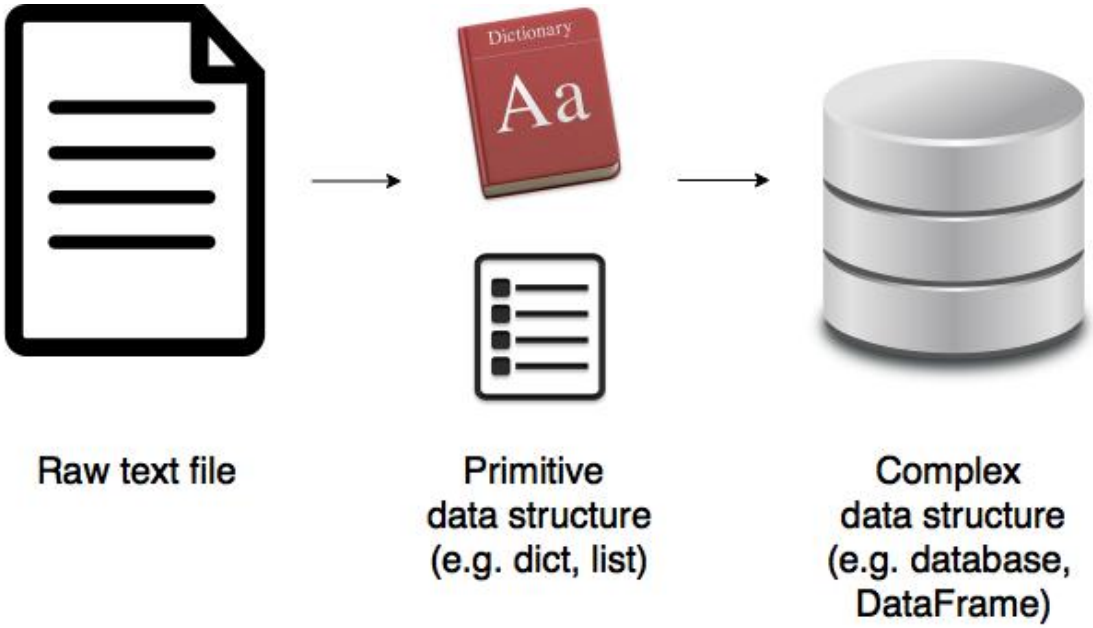
Write a python script to extract sequence from the GenBank file! Don't reinvent the wheel

```
LOCUS       XP_024307907             446 aa            linear   PRI 26-MAR-2018
DEFINITION  cystathionine beta-synthase isoform X6 [Homo sapiens].
ACCESSION   XP_024307907
VERSION     XP_024307907.1
DBLINK      BioProject: PRJNA168
DBSOURCE    REFSEQ: accession XM_024452139.1
KEYWORDS    RefSeq.
SOURCE      Homo sapiens (human)
  ORGANISM  Homo sapiens
            Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi;
            Mammalia; Eutheria; Euarchontoglires; Primates; Haplorrhini;
            Catarrhini; Hominidae; Homo.
COMMENT     MODEL REFSEQ:  This record is predicted by automated computational
            analysis. This record is derived from a genomic sequence
            (NC_000021.9) annotated using gene prediction method: Gnomon,
            supported by mRNA and EST evidence.
            Also see:
                Documentation of NCBI's Annotation Process

            ##Genome-Annotation-Data-START##
            Annotation Provider         :: NCBI
            Annotation Status           :: Full annotation
            Annotation Name             :: Homo sapiens Annotation Release 109
            Annotation Version          :: 109
            Annotation Pipeline         :: NCBI eukaryotic genome annotation
                                           pipeline
            Annotation Software Version :: 8.0
            Annotation Method           :: Best-placed RefSeq; Gnomon
            Features Annotated          :: Gene; mRNA; CDS; ncRNA
            ##Genome-Annotation-Data-END##
            COMPLETENESS: full length.
FEATURES             Location/Qualifiers
     source          1..446
                     /organism="Homo sapiens"
                     /db_xref="taxon:9606"
                     /chromosome="21"
     Protein         1..446
                     /product="cystathionine beta-synthase isoform X6"
                     /calculated_mol_wt=49088
     Region          1..439
                     /region_name="cysta_beta"
                     /note="cystathionine beta-synthase; TIGR01137"
                     /db_xref="CDD:273464"
     CDS             1..446
                     /gene="CBS"
                     /gene_synonym="HIP4"
                     /coded_by="XM_024452139.1:1288..2628"
                     /db_xref="GeneID:875"
                     /db_xref="HGNC:HGNC:1550"
                     /db_xref="MIM:613381"
ORIGIN
        1 makceffnag gsvkdrislr miedaerdgt lkpgdtiiep tsgntgigla laaavrgyrc
       61 iivmpekmss ekvdvlralg aeivrtptna rfdspeshvg vawrlkneip nshildqyrn
      121 asnplahydt tadeilqqcd gkldmlvasv gtggtitgia rklkekcpgc riigvdpegs
      181 ilaepeelnq teqttyeveg igydfiptvl drtvvdkwfk sndeeaftfa rmliaqegll
```

Biopython parses lots of formats

- Blast output – both from standalone and WWW Blast
- Clustalw
- FASTA
- GenBank
- PubMed and Medline
- ExPASy files, like Enzyme and Prosite
- SCOP, including ‘dom’ and ‘lin’ files
- UniGene
- SwissProt



Biopython classes: Seq

Seq objects (in Python an object is an instance of a class)

- acts like a string with a defined alphabet (DNA, RNA, protein)
- can be converted to string
- Seq object is immutable, ie “read only”
- Seq methods: complement, reverse_complement, transcribe, back_transcribe and translate

```
>>>from Bio.Seq import Seq
>>>my_seq = Seq("AGTACACTGGTA")
>>>my_seq = Seq("AGTACACTGGTA", IUPAC.unambiguous_dna)
>>>my_seq.count("A")
3
>>>str(my_seq)
'AGTACACTGGTA'
>>>Seq.translate(my_seq)
Seq('STLV', IUPACProtein())
```

Biopython classes: SeqRecord

SeqRecord objects

- higher level features such as identifiers and features to be associated with a sequence
- basic data type for the Bio.SeqIO sequence input/output
- attributes: .seq, .id, .name, .description, .annotations, etc

NC_005816.fna

>gi|45478711|ref|NC_005816.1| Yersinia pestis biovar Microtus ... pPCP1, complete sequence
TGTAACGAACGGTGCAATAGTGATCCACACCCAACGCCTGAAAT
CAGATCCAGGGGGTAATCTGCTCTCC

```
>>>from Bio import SeqIO
>>>record = SeqIO.read("NC_005816.fna", "fasta")
>>>record.seq
Seq('TGTAACGAACGGTGCAATAGTGATCCACACCCAACGCCTGAAATCAGATCCAG
G...CTG', SingleLetterAlphabet())
>>>record.id
'gi|45478711|ref|NC_005816.1|'
```

Biopython interact with databases

- Tools to interact with on-line bioinformatics destinations (NCBI, Ensembl REST)
- Entrez is a data retrieval system that provides users access to NCBI's databases such as PubMed, GenBank, GEO, and many others.
- The Bio.Entrez module makes use of EUtils, XML output and parse.

Pubmed Search

```
>>> from Bio import Entrez
>>> Entrez.email = "name@example.com" # Tell NCBI who you are
>>> handle = Entrez.esearch(db="pubmed", term="biopython")
>>> record = Entrez.read(handle)
>>> "19304878" in record["IdList"]
True
>>> print(record["IdList"])
['28011774', '24929426', '24497503', '24267035', '24194598', ...,
'14871861']
```

ESearch interact with databases

Genbank Search

```
>>> handle = Entrez.esearch(db="nucleotide",
term="Cypripedioideae[Orgn] AND matK[Gene]", idtype="acc")
>>> record = Entrez.read(handle)
>>> record["Count"]
'348'
>>> record["IdList"]
['JQ660909.1', 'JQ660908.1', 'JQ660907.1', 'JQ660906.1', ...,
'JQ660890.1']
```

EFetch interact with databases

Get Genbank files

```
>>> handle = Entrez.efetch(db="nucleotide", id="EU490707",
rettype="gb", retmode="text")
>>> print(handle.read())
LOCUS       EU490707               1302 bp    DNA     linear   PLN
26-JUL-2016
DEFINITION  Selenipedium aequinoctiale maturase K (matK) gene, partial
cds;
            chloroplast.
ACCESSION   EU490707
VERSION     EU490707.1
KEYWORDS    .
SOURCE      chloroplast Selenipedium aequinoctiale
  ORGANISM  Selenipedium aequinoctiale
```

To get the output in XML format, which you can parse using the Bio.Entrez.read() function, use retmode="xml":

```
>>> handle = Entrez.efetch(db="nucleotide", id="EU490707",
retmode="xml")
>>> record = Entrez.read(handle)
>>> handle.close()
>>> record[0]["GBSeq_definition"]
'Selenipedium aequinoctiale maturase K (matK) gene, partial cds;
chloroplast'
```

Biopython interface

- Tools to deal with on-line bioinformatics destinations (NCBI, ExPASy)
- Interface to common bioinformatics programs (Blast, ClustalW)
- A sequence obj dealing with seqs, seq IDs, seqfeatures

Align sequences with clustalw:

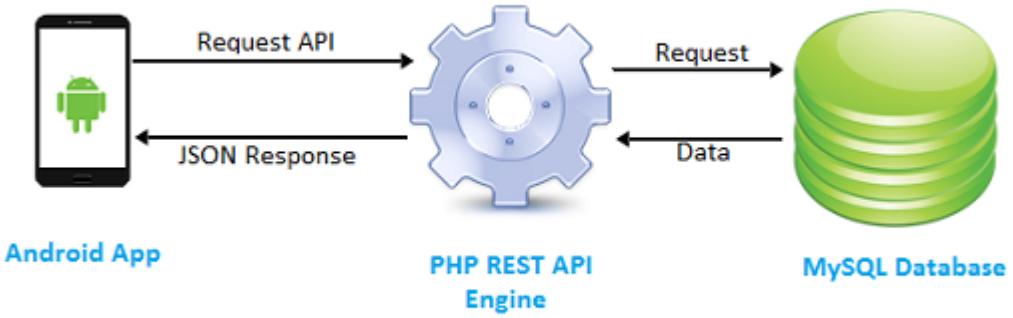
```
>>>from Bio.Align.Applications import ClustalwCommandline
>>>cline = ClustalwCommandline("clustalw2", infile="opuntia.fasta")
>>>print(cline)
SingleLetterAlphabet() alignment with 7 rows and 906 columns
TATACATTAAAGAAGGGGGATGCGGATAAATGGAAAGGCGAAAG...AGA gi|6273285|gb|
AF191659.1|AF191
TATACATTAAAGAAGGGGGATGCGGATAAATGGAAAGGCGAAAG...AGA gi|6273284|gb|
AF191658.1|AF191
TATACATTAAAGAAGGGGGATGCGGATAAATGGAAAGGCGAAAG...AGA gi|6273287|gb|
AF191661.1|AF191
```

API

Application programming interface (API) is a set of subroutine definitions, communication protocols, and tools for building software.

API may be for a web-based system, operating system, database system, computer hardware, or software library.

API makes it easier to develop a computer program by providing all the building blocks, which are then put together by the programmer.



Biological Databases

Pubmed
- abstracts and references

Model organism databases
- strains, genes, alleles, phenotypes, etc

DNA databases
- EMBL, GenBank, Ensembl

Disease association databases
- OMIM - mendelian diseases
- dbGAP, genotypes and phenotypes
  - authorized use only

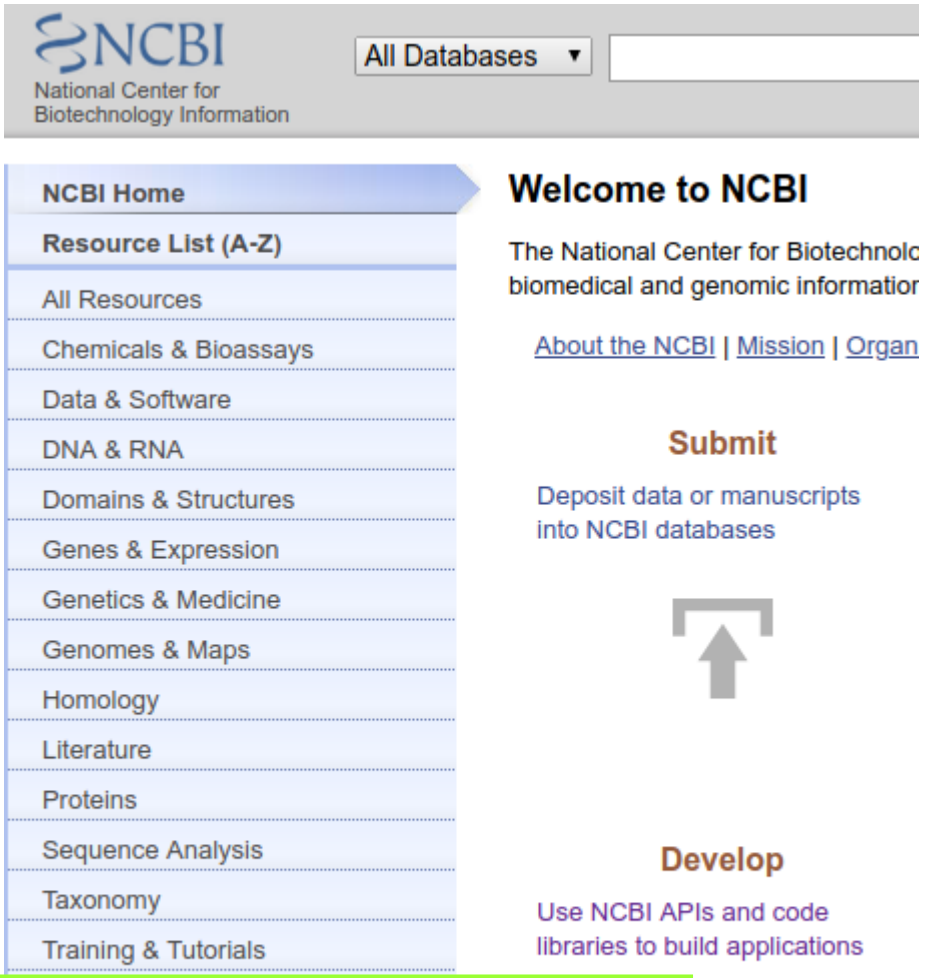
Gene expression
- ArrayExpress

Protein sequence/family
- Swiss-Prot, Pfam, InterPro

Protein structure
- Protein model portal

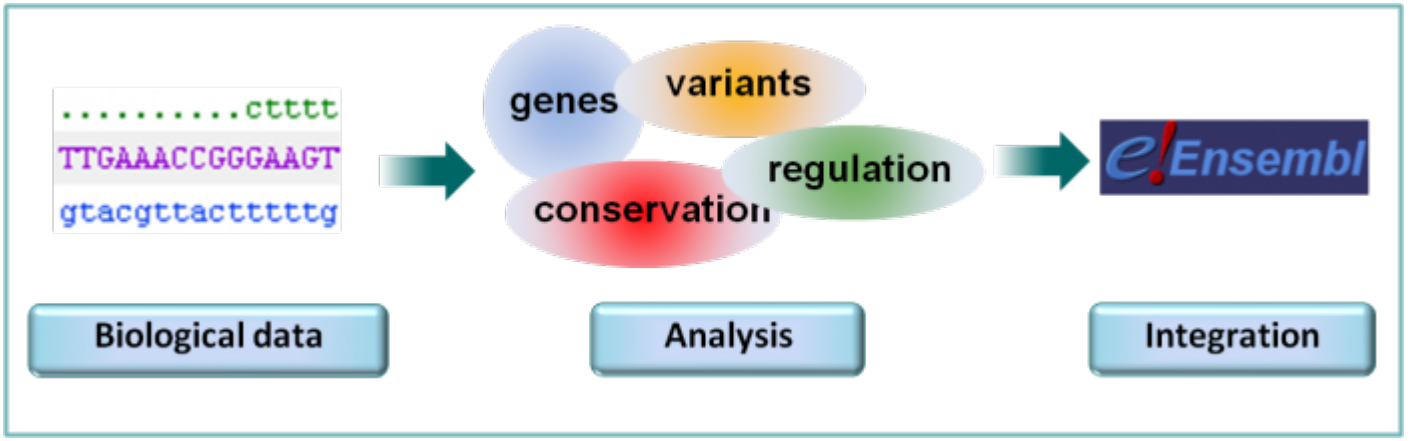
Pathways
- KEGG

DNA variation
- ExAC
- dbSNP



Manual: browser (single genes or many hours)
Automated: API (groups of genes or genomes, access specific information)
Local: download database (limited by size)

Ensembl database example

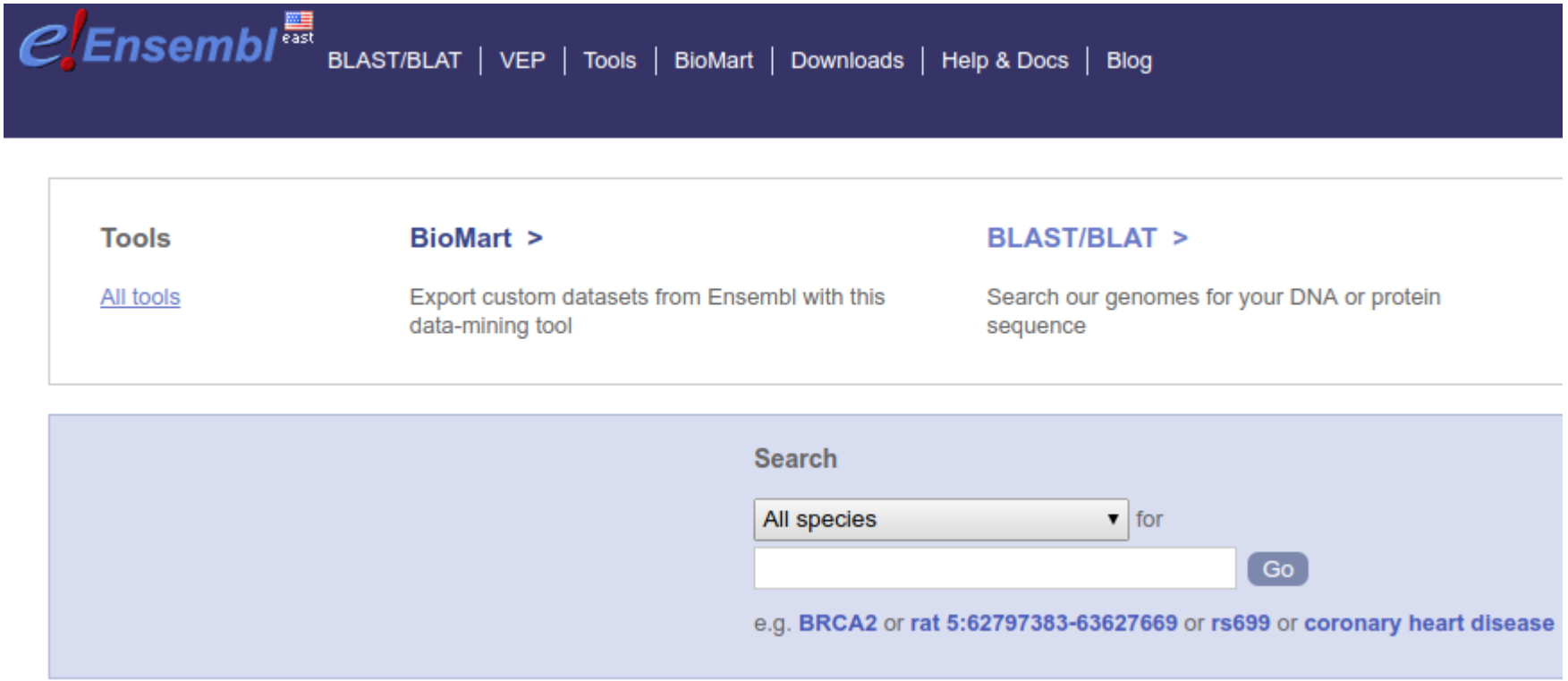


Given a gene..

- homologs in another species
- cDNA sequence, splicing and gene structure
- neighboring genes and regulation
- conserved regions (e.g. noncoding)
- sequence variation (SNPs)

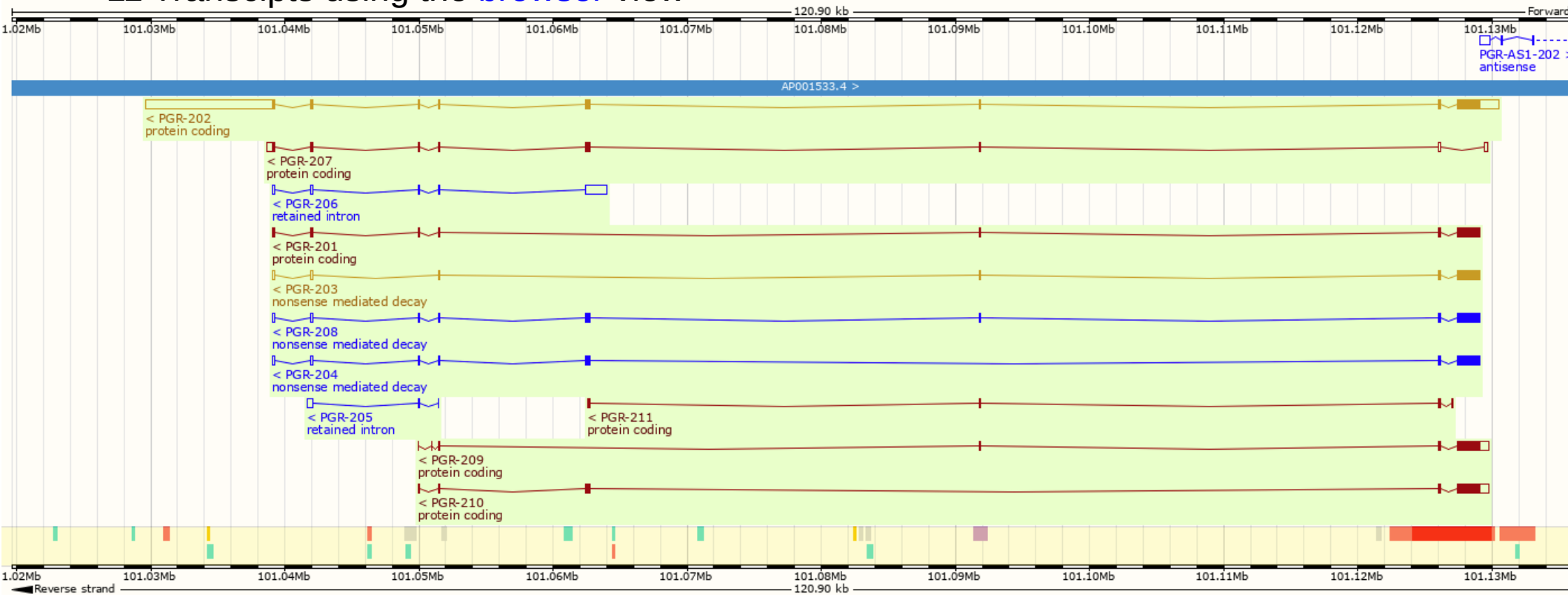
Accessing Information

1) website at www.ensembl.org
2) BioMart to quickly obtain tables of gene information, REST
3) Perl (Python-REST) APIs

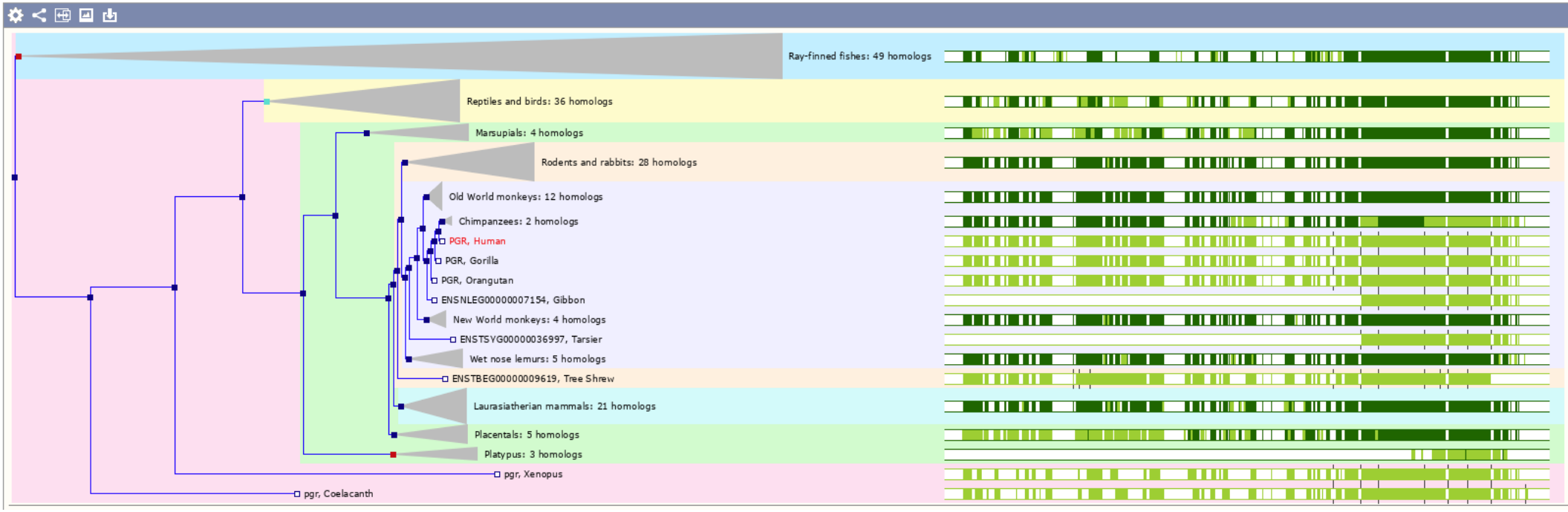


What might Chr11:101,039,119 C/G do

11 Transcipts using the browser view



Gene structure across 179 orthologs and 8 paralogs



Phenotypes

Phenotypes

Phenotype, disease and trait associated with this gene ENSG00000082175

Phenotype, disease and trait
PROGESTERONE RESISTANCE

Phenotype, disease and trait annotations associated with variants in this gene

Phenotype, disease and trait	Source(s)
ALL variants with a phenotype annotation	-
Annotated by HGMD but no phenotype description is publicly available	HGMD-PUBLIC
Diastolic blood pressure	NHGRI-EBI GWAS catalog
Glucose	dbGaP
Hematocrit	dbGaP
Menarche (age at onset)	NHGRI-EBI GWAS catalog
Severity of nausea and vomiting of pregnancy	NHGRI-EBI GWAS catalog

Phenotype, disease and trait annotations associated orthologues of this gene in other species

Show All entries

Phenotype, disease and trait	Source	Species
whole organism, female sterile	ZFIN	Zebrafish (*Danio rerio*)

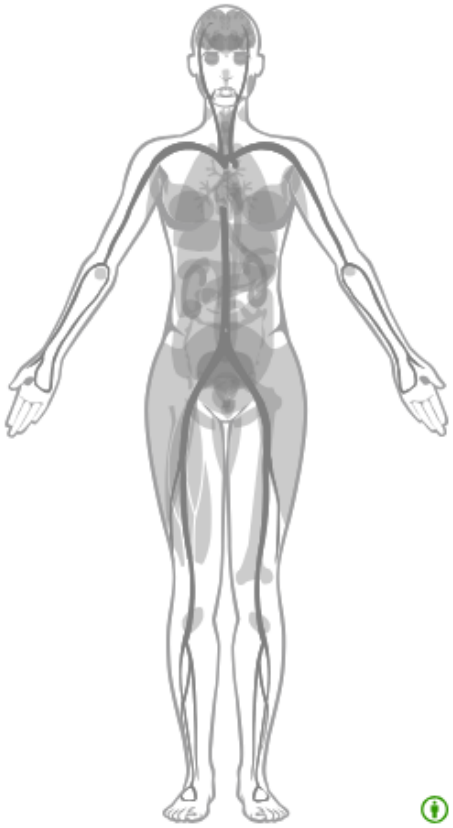
SNP features

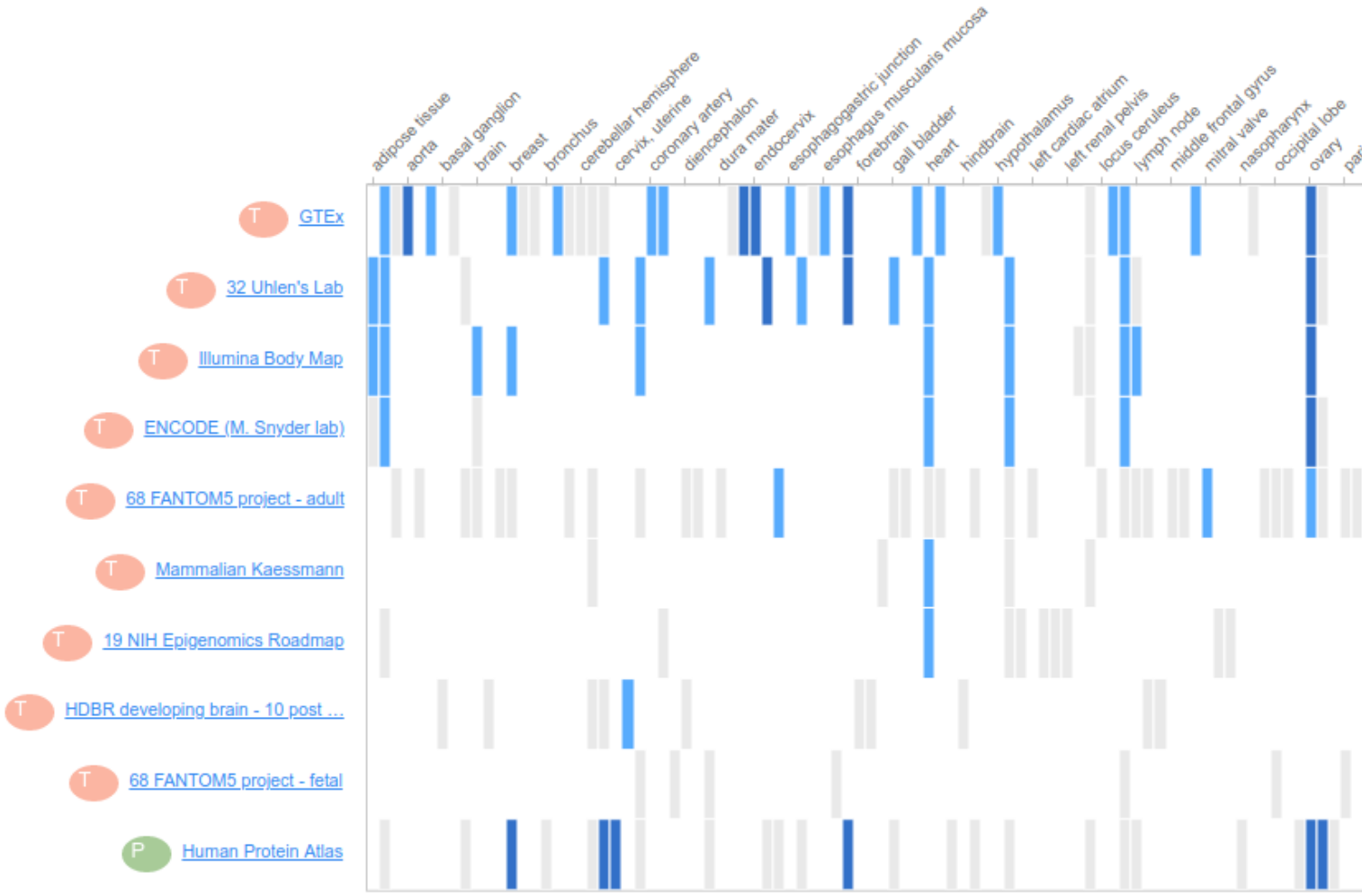
rs1158730512 SNP

Most severe consequence	**missense variant** | See all predicted consequences
Alleles	**C/G** | Ancestral: **C** | Highest population MAF: **< 0.01**
Change tolerance	CADD: G:24.1 | GERP: 1.14
Location	Chromosome **11:101039119** (forward strand) | VCF: `11 101039119 rs1158730512 C G`
Evidence status	
HGVS names	This variant has **19** HGVS names - Show
Synonyms	**ClinGen Allele Registry** CA382433217 (G)
Original source	Variants (including SNPs and indels) imported from dbSNP (release 151) | View in dbSNP
About this variant	This variant overlaps 7 transcripts.

Genes and regulation

PR expression

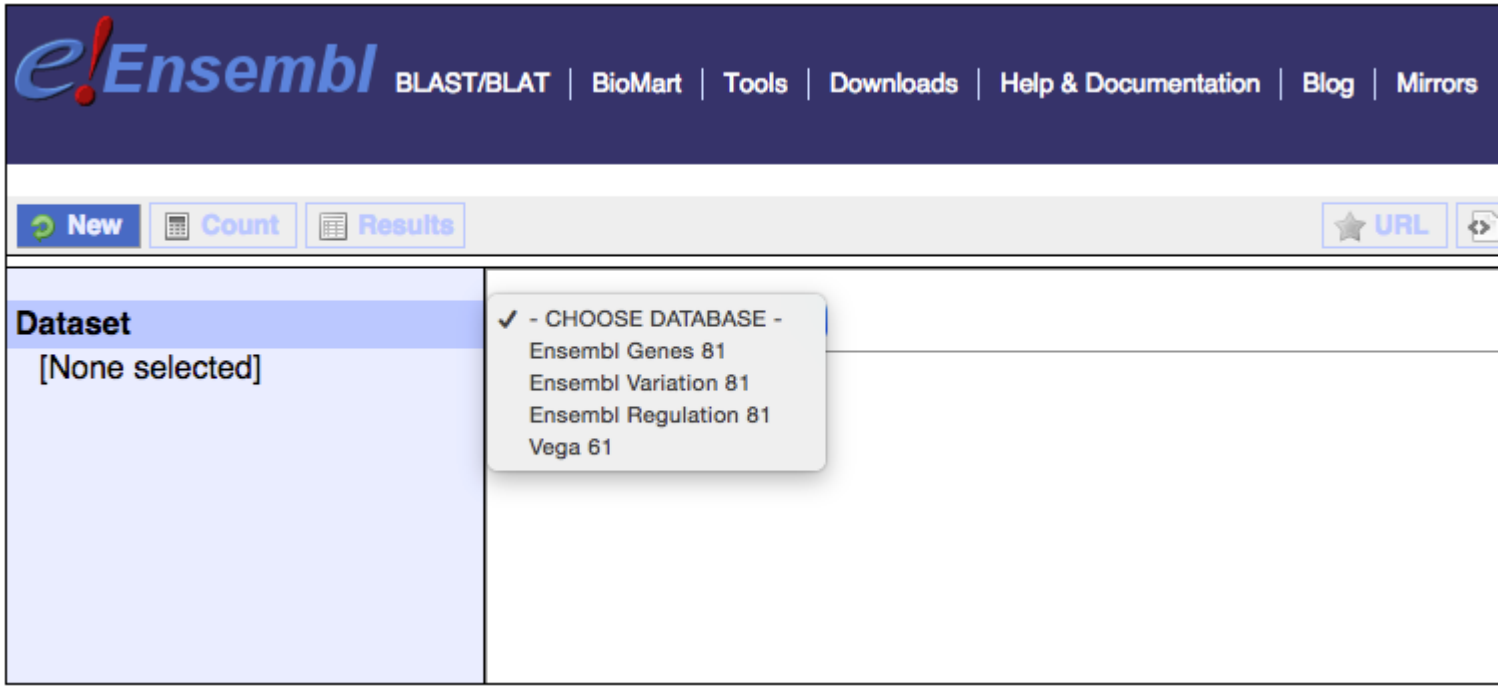




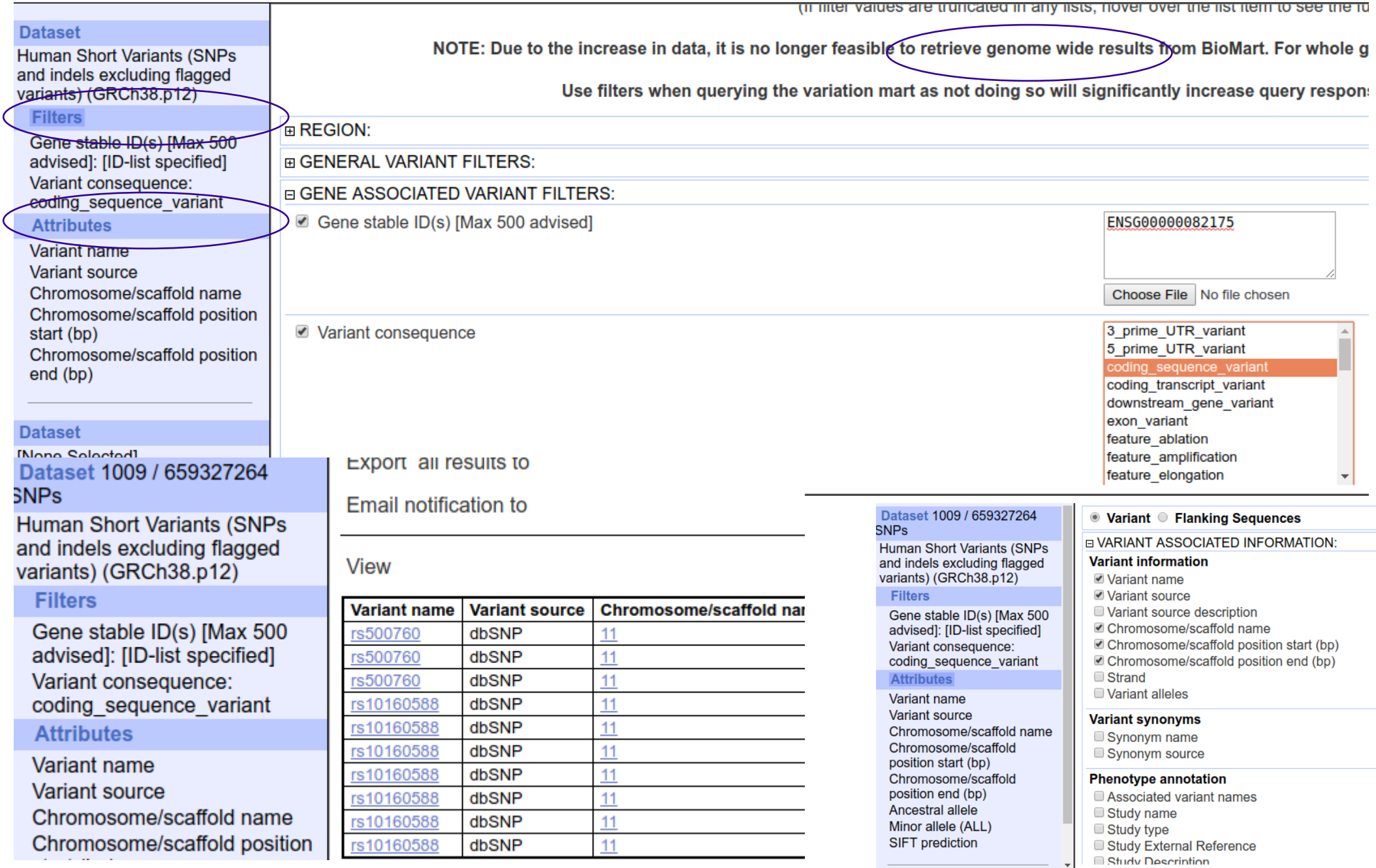
Great, now I need this information for 1000 SNPs

1) website at www.ensembl.org
2) BioMart to quickly obtain tables of gene information, REST
3) Perl (Python) APIs

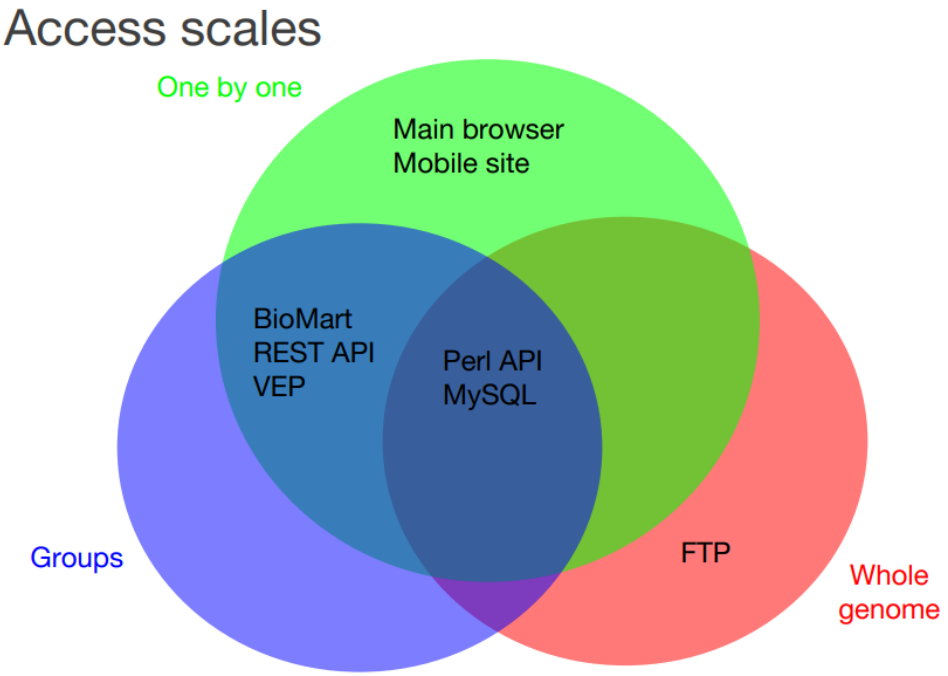
BioMart is an easy-to-use web-based tool that allows extraction of data without any programming knowledge or understanding of the underlying database structure. (web interface)



BioMart


Dataset
Human Short Variants (SNPs and indels excluding flagged variants) (GRCh38.p12)
Filters
Gene stable ID(s) [Max 500 advised]: [ID-list specified]
Variant consequence: coding_sequence_variant
Attributes
Variant name
Variant source
Chromosome/scaffold name
Chromosome/scaffold position start (bp)
Chromosome/scaffold position end (bp)
NOTE: Due to the increase in data, it is no longer feasible to retrieve genome wide results from BioMart. For whole g
Use filters when querying the variation mart as not doing so will significantly increase query respons
REGION:
GENERAL VARIANT FILTERS:
GENE ASSOCIATED VARIANT FILTERS:
Gene stable ID(s) [Max 500 advised]
ENSG00000082175
Choose File
No file chosen
Variant consequence
3_prime_UTR_variant
5_prime_UTR_variant
coding_sequence_variant
coding_transcript_variant
downstream_gene_variant
exon_variant
feature_ablation
feature_amplification
feature_elongation
Dataset 1009 / 659327264 SNPs
Human Short Variants (SNPs and indels excluding flagged variants) (GRCh38.p12)
Filters
Gene stable ID(s) [Max 500 advised]: [ID-list specified]
Variant consequence: coding_sequence_variant
Attributes
Variant name
Variant source
Chromosome/scaffold name
Chromosome/scaffold position
Export all results to
Email notification to
View
Variant name
Variant source
Chromosome/scaffold nar
rs500760 dbSNP 11
rs500760 dbSNP 11
rs500760 dbSNP 11
rs10160588 dbSNP 11
rs10160588 dbSNP 11
rs10160588 dbSNP 11
rs10160588 dbSNP 11
rs10160588 dbSNP 11
rs10160588 dbSNP 11
rs10160588 dbSNP 11
Dataset 1009 / 659327264 SNPs
Human Short Variants (SNPs and indels excluding flagged variants) (GRCh38.p12)
Filters
Gene stable ID(s) [Max 500 advised]: [ID-list specified]
Variant consequence: coding_sequence_variant
Attributes
Variant name
Variant source
Chromosome/scaffold name
Chromosome/scaffold position start (bp)
Chromosome/scaffold position end (bp)
Ancestral allele
Minor allele (ALL)
SIFT prediction
Variant
Flanking Sequences
VARIANT ASSOCIATED INFORMATION:
Variant information
Variant name
Variant source
Variant source description
Chromosome/scaffold name
Chromosome/scaffold position start (bp)
Chromosome/scaffold position end (bp)
Strand
Variant alleles
Variant synonyms
Synonym name
Synonym source
Phenotype annotation
Associated variant names
Study name
Study type
Study External Reference
Study Description

Ensembl REST API



1) website at www.ensembl.org
2) BioMart to quickly obtain tables of gene information, REST API
3) Perl (Python) APIs

REST request:
- The endpoint (URL route)
- The method (GET, POST ...)
- The headers (AUTH, cache)
- The data (or body) with POST

Representational State Transfer (REST) is a set of rules that developers follow when they create their API. One of these rules states that you should be able to get a piece of data (called a resource) when you link to a specific URL.

- Each URL is called a request while the data sent back to you is called a response.
- A RESTful API is an application program interface (API) that uses HTTP requests to GET, PUT, POST and DELETE data. REST APIs uses multiple standards like HTTP, JSON, URL, and XML, REST!=HTTP

VEP determines the effect of your variants on genes, transcripts, and protein sequence, as well as regulatory regions. Available as command line tool, REST API.

Ensembl REST

- REST allows you to access the same information as you would from a website in an automated (API) interface.
- Language agnostic access
- REST is only a fraction of Ensembl's perl API

http://rest.ensembl.org/lookup/id/ENSG00000157764?content-type=application/json


```
{
  "source": "ensembl_havana",
  "object_type": "Gene",
  "logic_name": "ensembl_havana_gene",
  "version": 12,
  "species": "homo_sapiens",
  "description": "B-Raf proto-oncogene, serine/threonine kinase [Source:HGNC
Symbol;Acc:HGNC:1097]",
  "display_name": "BRAF",
  "assembly_name": "GRCh38",
  "biotype": "protein_coding",
  "end": 140924764,
  "seq_region_name": "7",
  "db_type": "core",
  "strand": -1,
  "id": "ENSG00000157764",
  "start": 140719327
}
```

Ensembl REST endpoints

Endpoints:
Resource
Cross-reference
EQTL Information
Ontologies and taxonomies
etc

Resource	Description
GET cafe/genetree/id/:id	Retrieves a cafe tree of the gene tree using the gene tree stable identifier
GET cafe/genetree/member/id/:id	Retrieves the cafe tree of the gene tree that contains the gene / transcript / translation stable identifier
GET cafe/genetree/member/symbol/:species/:symbol	Retrieves the cafe tree of the gene tree that contains the gene identified by a symbol
GET family/id/:id	Retrieves a family information using the family stable identifier
GET family/member/id/:id	Retrieves the information for all the families that contains the gene / transcript / translation stable identifier
GET family/member/symbol/:species/:symbol	Retrieves the information for all the families that contains the gene identified by a symbol
GET genetree/id/:id	Retrieves a gene tree for a gene tree stable identifier
GET genetree/member/id/:id	Retrieves the gene tree that contains the gene / transcript / translation stable identifier
GET genetree/member/symbol/:species/:symbol	Retrieves the gene tree that contains the gene identified by a symbol
GET alignment/region/:species/:region	Retrieves genomic alignments as separate blocks based on a region and species
GET homology/id/:id	Retrieves homology information (orthologs) by Ensembl gene id
GET homology/symbol/:species/:symbol	Retrieves homology information (orthologs) by symbol

GET gene tree documentation

GET genetree/id/:id

Retrieves a gene tree for a gene tree stable identifier

Parameters

Required

Name	Type	Description	Default	Example Values
id	*String*	An Ensembl genetree ID	-	*ENSGT00390000003602*

Optional

Name	Type	Description	Default	Example Values
aligned	*Boolean*	Return the aligned string if true. Otherwise, return the original sequence (no insertions)	*0*	-
callback	*String*	Name of the callback subroutine to be returned by the requested JSONP response. Required ONLY when using JSONP as the serialisation method. Please see the user guide.	-	*randomlygeneratedname*

Python Example

/genetree/id/ENSGT00390000003602?content-type=application/json

Example output | Perl | Python2 | **Python3** | Ruby | Java | R | Curl | Wget

```python
import requests, sys

server = "https://rest.ensembl.org"
ext = "/genetree/id/ENSGT00390000003602?"

r = requests.get(server+ext, headers={ "Content-Type" : "application/json"})

if not r.ok:
  r.raise_for_status()
  sys.exit()

decoded = r.json()
print(repr(decoded))
```

Output:


```
{
 "rooted": 1,
 "tree": {
   "lambda": 0.0000817808,
   "id": 40129314,
   "children": [
     {
       "lambda": 0.0000817808,
       "id": 40129316,
       "tax": {
         "scientific_name": "Bilateria",
         "timetree_mya": "796.6",
         "id": 33213,
         "common_name": "Bilateral animals"
       },
       "p_value_lim": 0.01,
       "name": "1",
```


REST Rate limit:

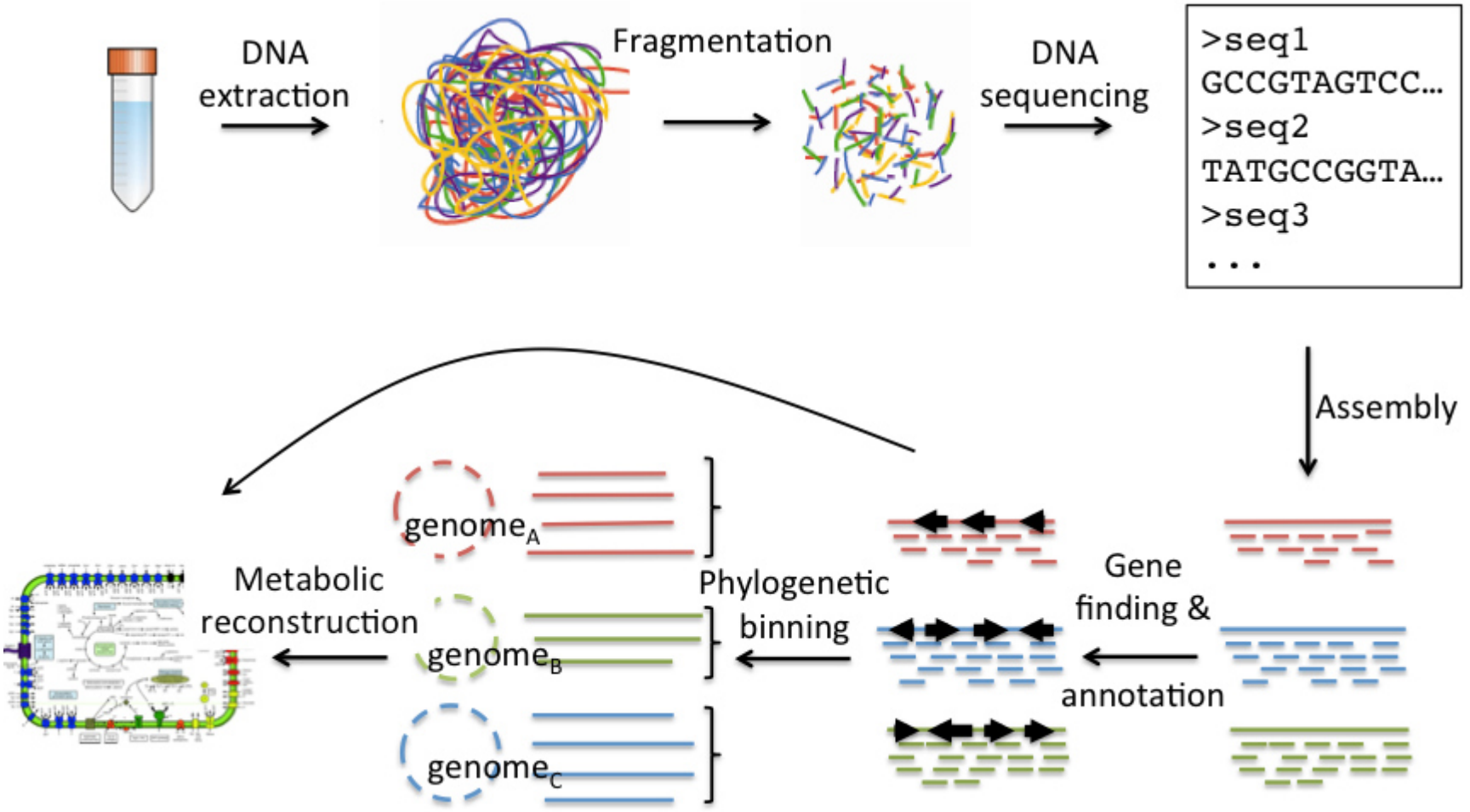
```
X-RateLimit-Limit: 55000
X-RateLimit-Reset: 892
X-RateLimit-Period: 3600
X-RateLimit-Remaining: 54999
```

55000 requests over an hour (3600 seconds) meaning an average of 15 requests per second.

16s vs shotgun metagenomic

	16s	Metagenomic
Sequences	just bacteria	virus, fungal, bacteria, host (up to 98% of reads)
Species ID	Reference databases (e.g. RPD)	Reference genomes
PCR	primer bias (species drop out)	No PCR bias
Resolution	Species or genera (high coverage)	strain level resolution (lower coverage)
Genes/ Species	New species found	gene sequences → function (GO terms)

Metagenomics: community function



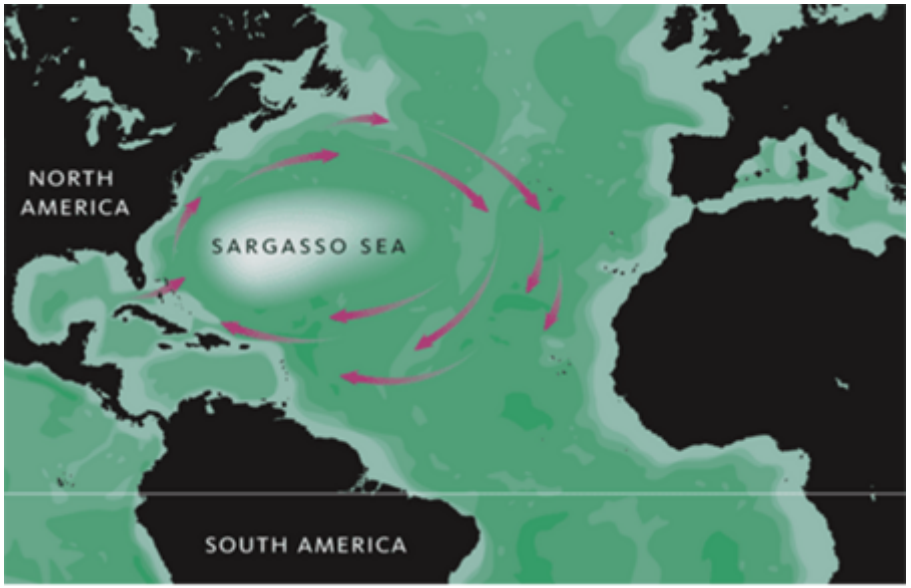
Assemblies are optional

Sargasso Sea-2004

Gene content of metagenome
– reveals community function

TIGR role category	Total genes
Amino acid biosynthesis	37,118
Biosynthesis of cofactors, prosthetic groups, and carriers	25,905
Cell envelope	27,883
Cellular processes	17,260
Central intermediary metabolism	13,639
DNA metabolism	25,346
Energy metabolism	69,718
Fatty acid and phospholipid metabolism	18,558
Mobile and extrachromosomal element functions	1,061
Protein fate	28,768
Protein synthesis	48,012
Purines, pyrimidines, nucleosides, and nucleotides	19,912
Regulatory functions	8,392
Signal transduction	4,817
Transcription	12,756
Transport and binding proteins	49,185
Unknown function	38,067
Miscellaneous	1,864
Conserved hypothetical	794,061
Total number of roles assigned	1,242,230
Total number of genes	1,214,207

- Filtered sea surface water, DNA extract and sequence
- 1.045 billion bases
- 1800 different species, including 148 types of bacteria never before seen

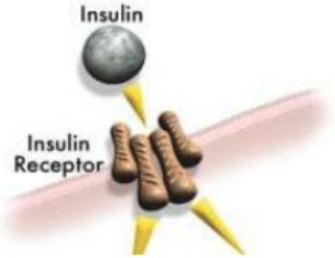


categories: how do we get these?

Gene Ontologies (GO)

1. Molecular Function

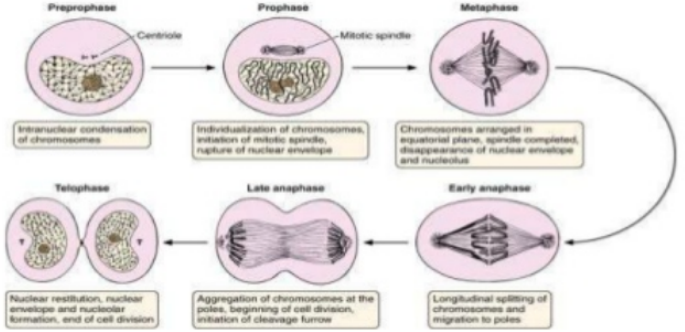
An elemental activity or task or job



- protein kinase activity
- insulin receptor activity

2. Biological Process

A commonly recognized series of events



- cell division

3. Cellular Component

Where a gene product is located

- mitochondrion
- mitochondrial matrix
- mitochondrial inner membrane

GO annotations

A GO annotation is …

…a statement that a gene product;

1. has a particular molecular function
   or is involved in a particular biological process
   or is located within a certain cellular component

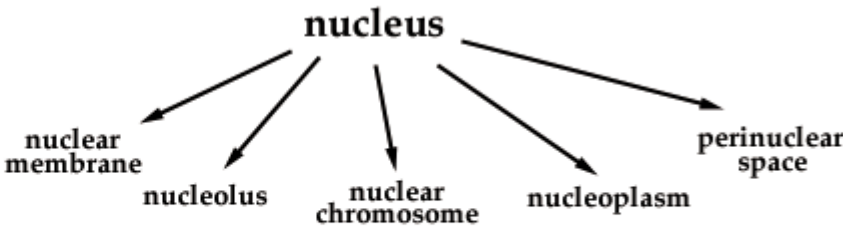
2. as described in a particular reference

3. as determined by a particular method

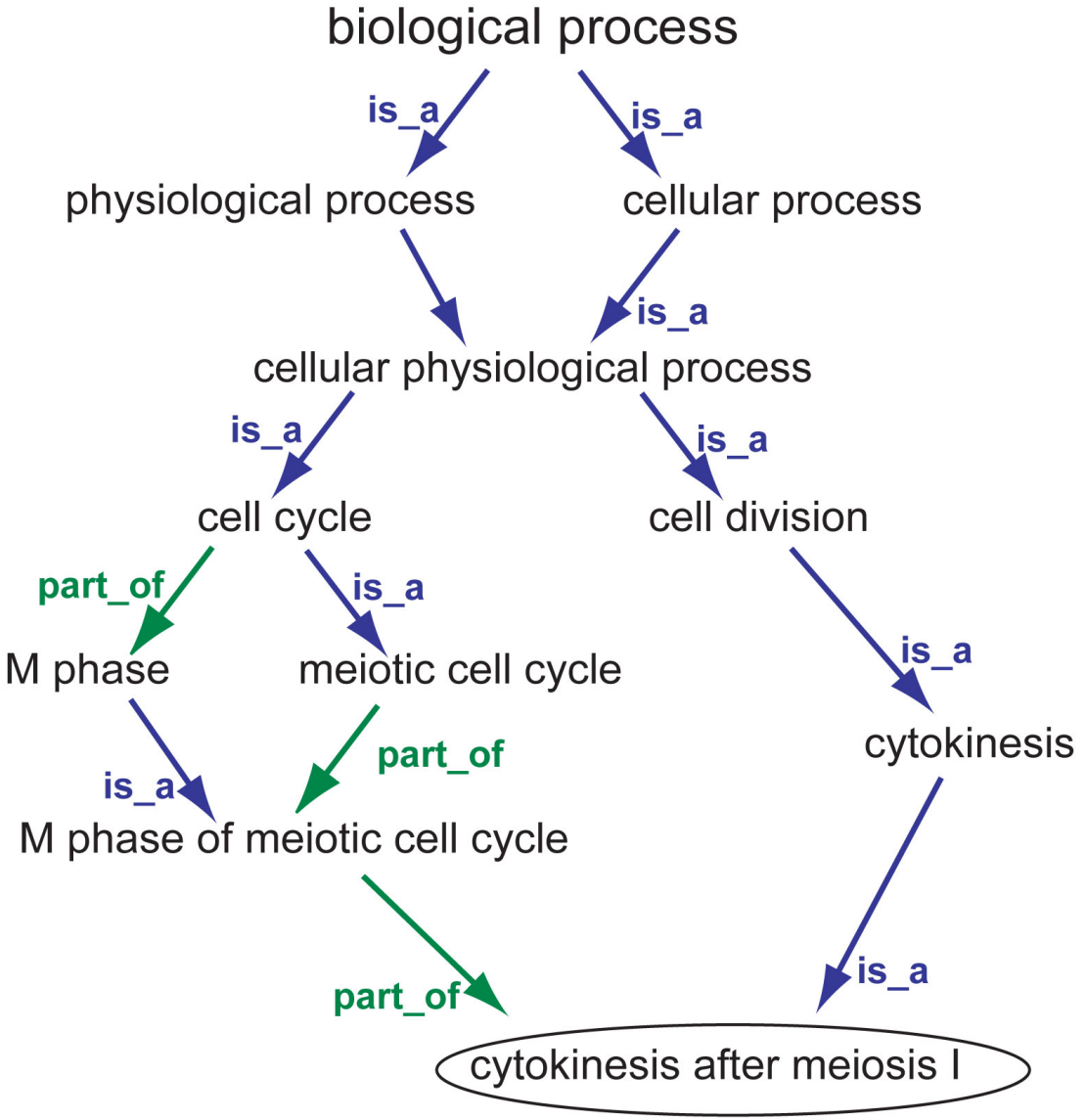
Accession	*Name*	*GO ID*	*GO term name*	*Reference*	*Evidence code*
P00505	GOT2	GO:0004069	aspartate transaminase activity	PMID:2731362	IDA

Electronic annotations	269,207,317
Manual annotations*	2,752,604

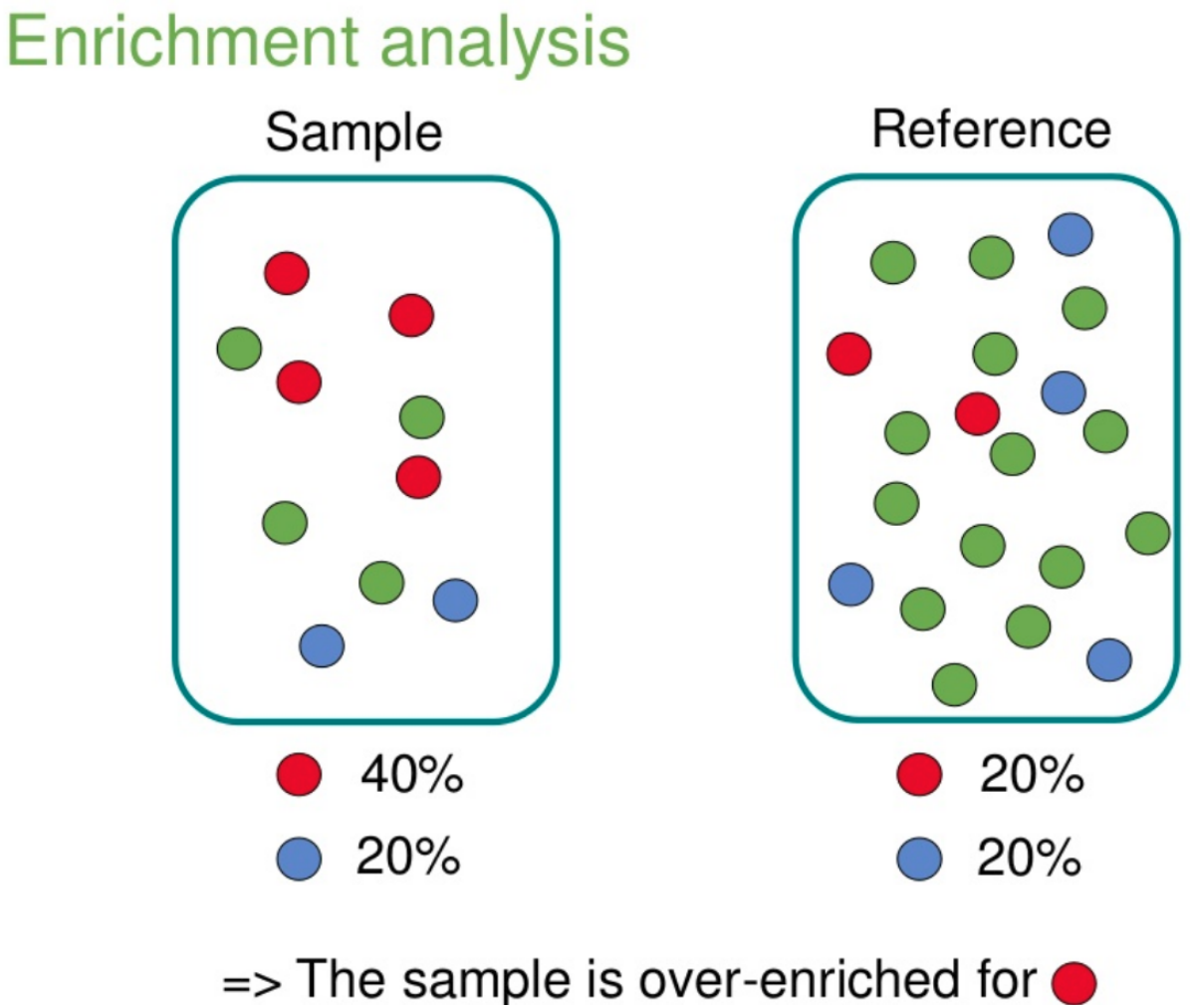
Ontology



The cellular component *nucleus*, shown with five children

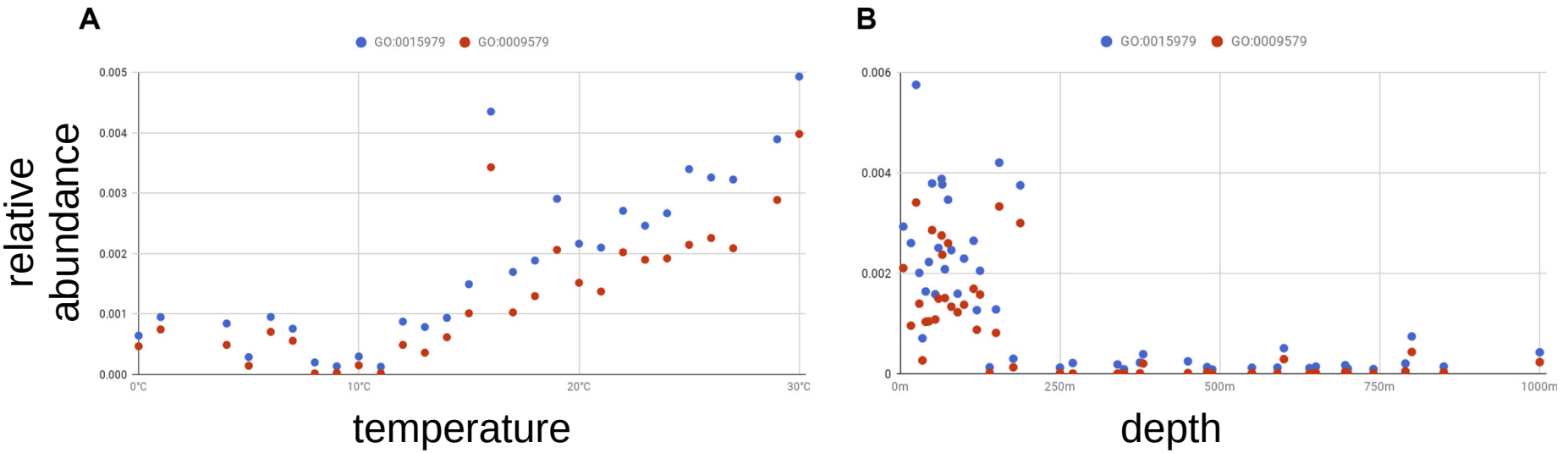


GO enrichment analysis



Metagenomics from oceanographic samples

Correlation between temperature (A) and depth (B) and photosynthesis-related GO term counts



Mitchell et al 2018

Functional Annotations with KEGG

KEGG (Kyoto Encyclopedia of Genes and Genomes) more than just GO

- A collection of databases dealing with genomes, biological pathways, diseases, drugs, and chemical substances.
- KEGG is utilized for bioinformatics in genomics, metagenomics, metabolomics.

Systems information

- PATHWAY — pathway maps for cellular and organismal functions
- MODULE — modules or functional units of genes
- BRITE — hierarchical classifications of biological entities

Genomic information

- GENOME — complete genomes
- GENES — genes and proteins in the complete genomes
- ORTHOLOGY — ortholog groups of genes in the complete genomes

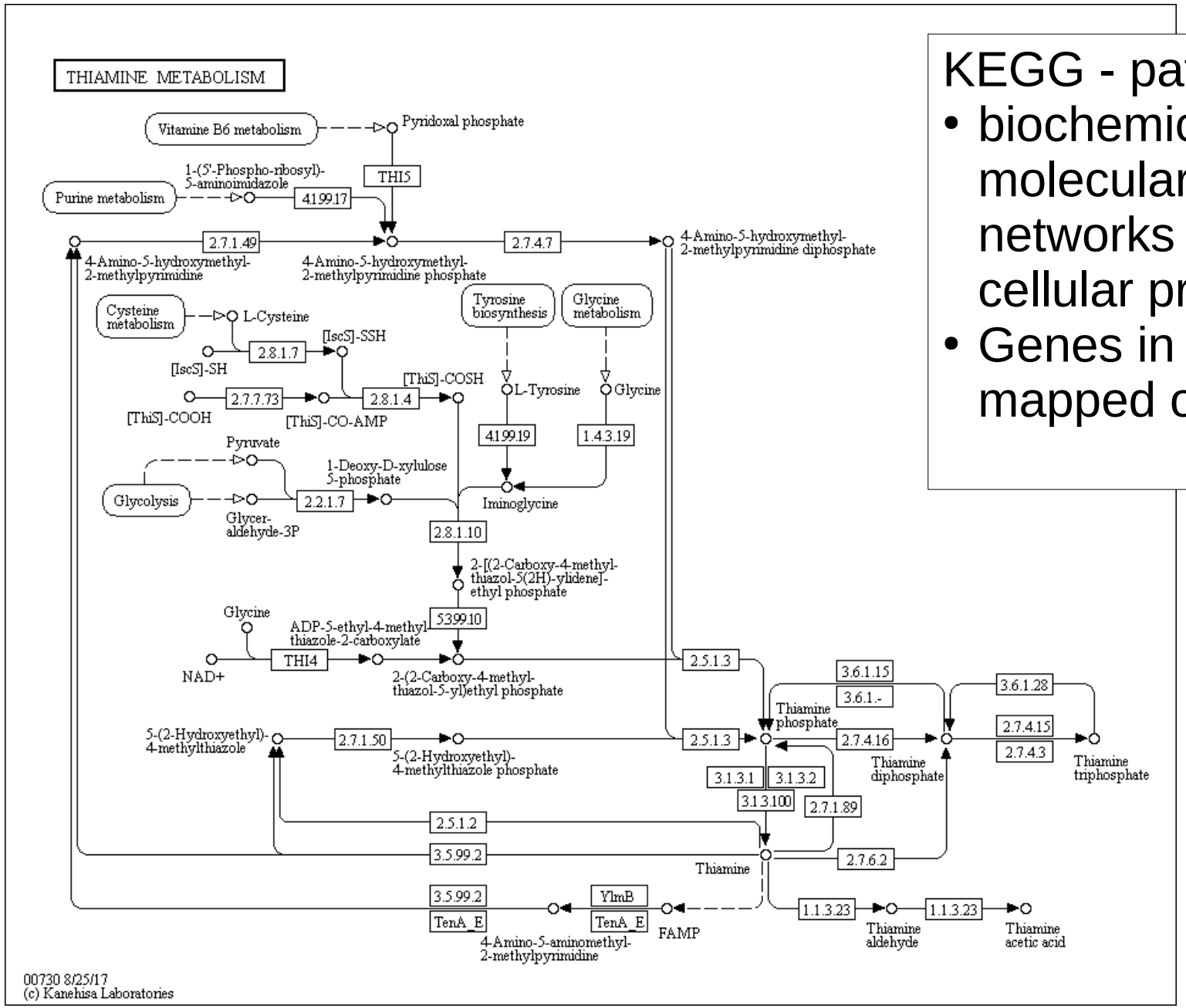
Chemical information

- COMPOUND, GLYCAN — chemical compounds and glycans
- REACTION, RPAIR, RCLASS — chemical reactions
- ENZYME — enzyme nomenclature

Health information

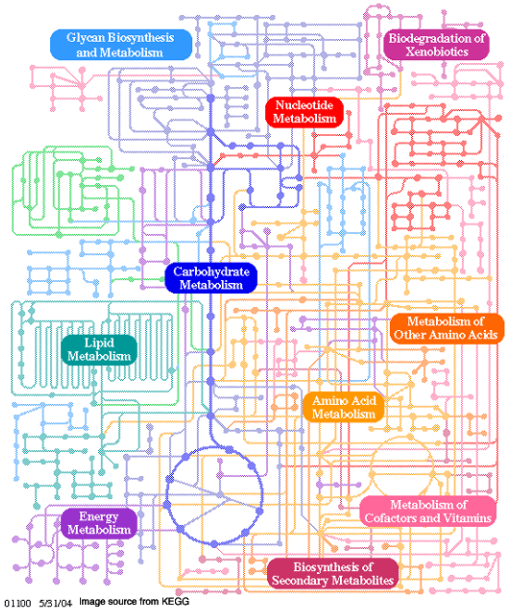
- DISEASE — human diseases
- DRUG — approved drugs
- ENVIRON — crude drugs and health-related substances

KEGG-pathways



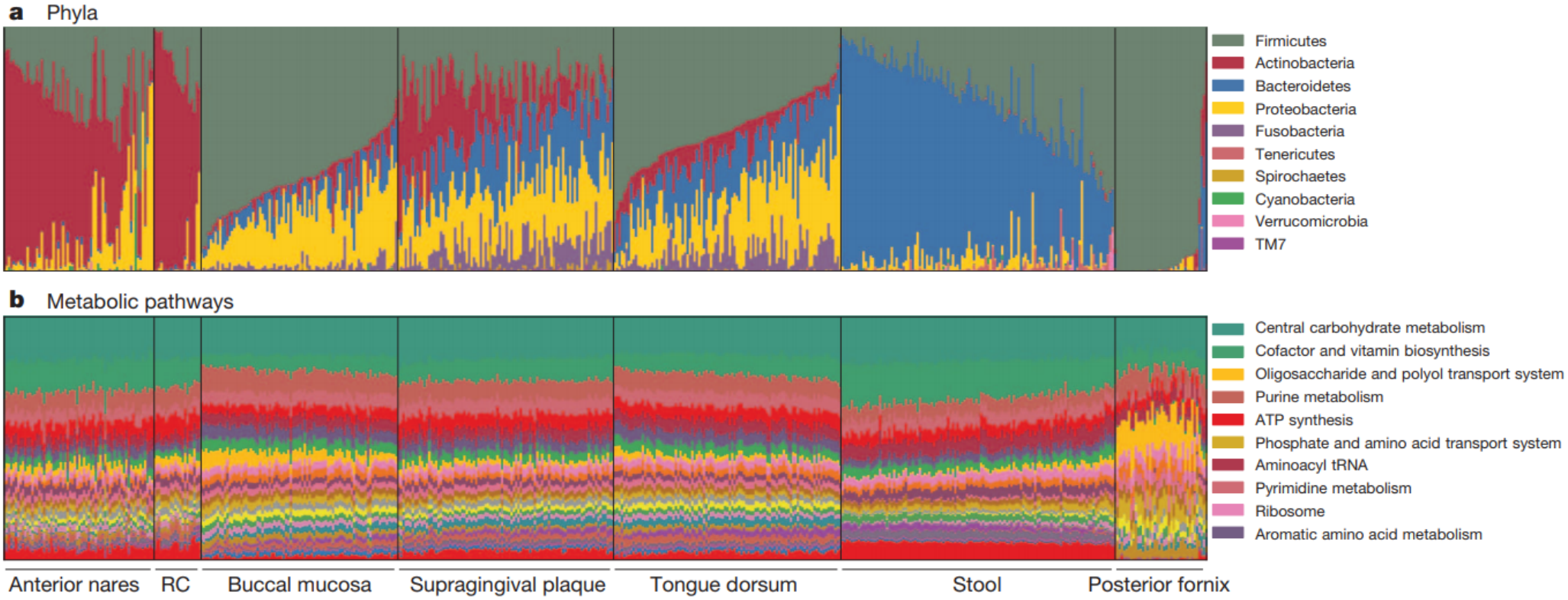
KEGG - pathways

- biochemical pathways, molecular interactions, networks for metabolism, cellular processes, etc
- Genes in each organism mapped onto pathways



Human Microbiome Project

What is the microbiome from healthy human adults?



- Phyla differ by body site
- ‘core microbiome’ of shared organisms, genes or functional capabilities (e.g. metabolic pathways)

Exercises

1) Give three advantages of using Biopython?

2) What are three ways of accessing Ensembl, which would you use for single locus query, groups of genes, whole genomes?

4) Why use pipes in the command line interface?

5) What is the advantage of using bash or Python to create a bioinformatics pipeline (combination of commands to process data)?

6) Why is API a better way to get information from a database than (i) web-browser, (ii) download database?

7) What is the advantage of shotgun metagenomic compared to 16S PCR microbiome analysis?

8) How would you compare the function of two microbiomes?