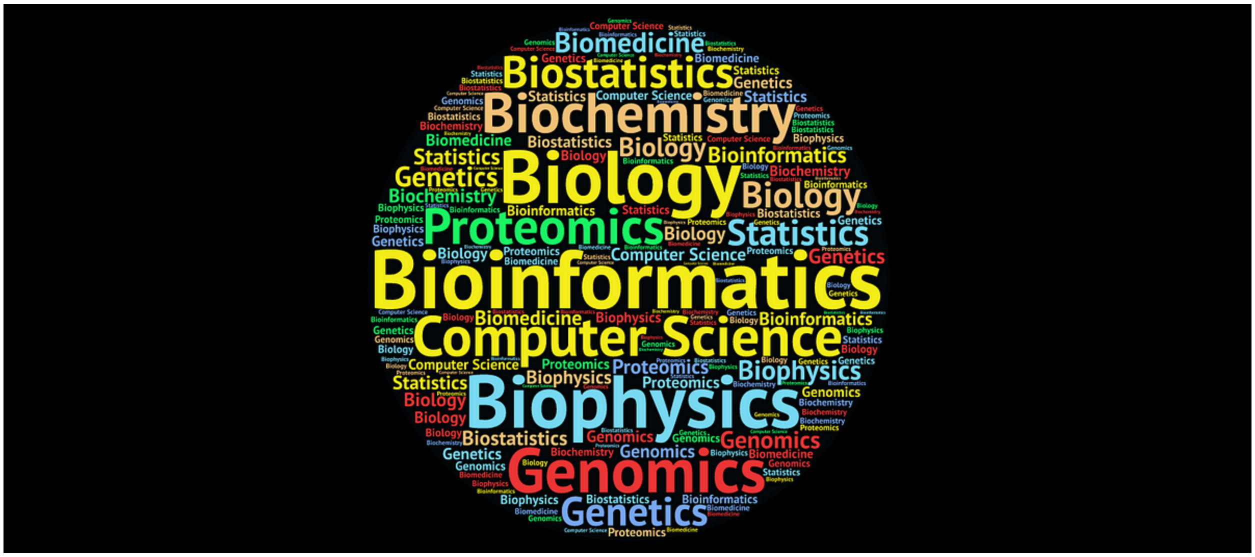
Introduction to Bioinformatics

Introduction to Bioinformatics



Bioinformatics is a rapidly growing field that combines biology, medicine, computer science and mathematics to analyse and interpret biological data.

Bioinformatics tools and techniques are used to study a wide range of biological phenomena, including the gene expression[1], protein structure and function[2] and the evolution of species.

One of the most important applications of bioinformatics is in the field of genomics. **Genomics** is the study of the genome, which is the complete set of genetic material in an organism. Bioinformatics tools are used to sequence and assemble genomes as well as to identify and analyse genes and other genomic features.

Why is it important?

Bioinformatics is important because it allows us to study biology in a quantitative and systematic way, by using computers to analyse large datasets of biological data, we can identify patterns and relationships that would be difficult to impossible to see in the naked eye inside a wet-lab.

Who should Learn Bioinformatics

Bioinformatics is a valuable skill for anyone in the health and biology fields. There are many profiles that can benefit from learning bioinformatics, including but not limited to: biologists, geneticists, biomedical engineers, medical doctors, pharmacists, veterinarians, chemists, and computer scientists. This is because bioinformatics is a multidisciplinary field that can benefit from the different knowledge and skills of people from different backgrounds.

[1] Gene Expression Omnibus (GEO) is a database of gene expression studies and datasets from several different organisms.

[2] Protein Data Bank (PDB) is a database of protein structures.

How to Learn Bioinformatics

There are many ways to learn bioinformatics; there are university courses, online courses, books, tutorials and workshops. If you are new to bioinformatics, I do recommend to star with learning the basis of biology and programming before diving into bioinformatics. You should learn how, when and why to use bioinformatics tools like BLAST, GEO and techniques like Meta-Analysis. You should also learn how to interpret the results of bioinformatics analyses.

Bioinformatics Tools and Techniques

To begin with, you should learn how to use the basic bioinformatics tools and techniques. These include:

Programming languages

There are several languages useful to bioinformatics, including R, [Python] (https://www.python.org/about/), Perl (currently in decline), C++, Julia (currently skyrocketing) and Java. R and Python are the most popular languages in the field this because they are easy to learn and have a large community of users. R is a statistical programming language used for data analysis and visualization. Python is a general-purpose programming language used for a wide range of applications, including web development, data science and machine learning. R and Python are both open source and free to use.

For this course we will be using R, because it is the most popular language in the field If you are interested in learning Python or you already have some proficiency with the language, I recommend the [Python for Biologists] (https://pythonforbiologists.com/) book.

For the sake of this course, we will be using R in its version 4.2.3(This due to compatibility issues with later releases). You can download it from the R Project website. We also encourage you to use RStudio as your IDE (Integrated Development Environment). You can download it from the RStudio website.

Bioconductor

Bioconductor is a collection of R packages designed and developed by the bioinformatics community. Bioconductor provides a wide range of tools for analysing and visualising biological data. The Bioconductor repository contains more than 2000 packages for analysing genomic data, proteomic data, metabolomic data, and other types of biological data. Bioconductor is free to use and open source.

To use all of Bioconductor's packages, you need to install the Bioconductor package. You can install the Bioconductor package by running the following code in R:

```r
if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager",version = "3.16")
```

We use the `BiocManager::install()` function to install the Bioconductor package. The `BiocManager::install()` function is part of the BiocManager package, which is used to install and update the necessary packages from Bioconductor, also note that we use the version argument to specify the version of the package we want to install, this is because the latest version of the package is not compatible with the version of R we are using.

Databases

Databases are used to store and organize biological data. There are many different types of databases, including sequence databases, structure databases, expression databases, and pathway databases. The biggest and most popular database is the National Center for Biotechnology Information (NCBI), which is a public database that contains a wide range of biological data, maintained by the National Institute of Health (NIH). The NCBI contains many different databases, including the GenBank, which is a Genomic database that contains DNA sequences, the Protein database, which contains protein sequences, and the Gene database, which contains gene sequences. The NCBI also contains databases for other types of data, including the Gene Expression Omnibus (GEO), which is a database of gene expression studies and datasets from several different organisms and the PubMed, which is a database of biomedical literature.

Data Retrieval Tools

The NCBI provides several tools for retrieving data from its databases, including the Entrez Programming Utilities (E-utilities), which is a set of 9 tools designed to provide access to the NCBI databases. The E-utilities can be accessed through the Entrez Direct (EDirect) command line interface a Python and R package or the web interface. The E-utilities can be used to search for data in the NCBI databases, download data from the NCBI databases, and upload data to the NCBI.

Ensembl is another popular database that contains a wide range of biological data, including genomic data, gene expression data, and protein data. Ensembl provides several tools for retrieving data called Ensembl REST APIs. The Ensembl REST APIs can be accessed through the Ensembl Perl API, those API's can be accessed and used without trouble using the bioconductor package biomaRt.

Example of data retrieval using biomaRt First, we need to verify if the package is installed and install it if it's not.

```
if (!requireNamespace("biomaRt", quietly = TRUE)) {
  BiocManager::install("biomaRt")
}
```

Then we need to load the package.

Once the package has been loaded, we can use all of its capabilities

```
# List available databases
listMarts()
```

```
##                biomart                version
## 1 ENSEMBL_MART_ENSEMBL      Ensembl Genes 110
## 2   ENSEMBL_MART_MOUSE      Mouse strains 110
## 3     ENSEMBL_MART_SNP  Ensembl Variation 110
## 4 ENSEMBL_MART_FUNCGEN Ensembl Regulation 110
```

```
# Select the database
mart <- useMart("ENSEMBL_MART_ENSEMBL")
```

We list all available Databases and select the Main `MART_ENSEMBL`

```
#List available datasets
datasets <- biomaRt::listDatasets(mart)
head(datasets,25)
```

```
##                       dataset
## 1  abrachyrhynchus_gene_ensembl
## 2      acalliptera_gene_ensembl
## 3    acarolinensis_gene_ensembl
## 4     acchrysaetos_gene_ensembl
## 5     acitrinellus_gene_ensembl
## 6     amelanoleuca_gene_ensembl
## 7       amexicanus_gene_ensembl
## 8       anancymaae_gene_ensembl
## 9       aocellaris_gene_ensembl
## 10        apercula_gene_ensembl
## 11  aplatyrhynchos_gene_ensembl
## 12   apolyacanthus_gene_ensembl
## 13 applatyrhynchos_gene_ensembl
## 14    atestudineus_gene_ensembl
## 15         bbbison_gene_ensembl
## 16      bgrunniens_gene_ensembl
## 17        bihybrid_gene_ensembl
## 18       bmusculus_gene_ensembl
## 19          bmutus_gene_ensembl
## 20      bsplendens_gene_ensembl
## 21         btaurus_gene_ensembl
## 22     cabingdonii_gene_ensembl
## 23           catys_gene_ensembl
## 24        cauratus_gene_ensembl
## 25        cccarpio_gene_ensembl
##                                            description                version
## 1                Pink-footed goose genes (ASM259213v1)            ASM259213v1
## 2                     Eastern happy genes (fAstCal1.2)             fAstCal1.2
## 3                      Green anole genes (AnoCar2.0v2)            AnoCar2.0v2
## 4                      Golden eagle genes (bAquChr1.2)             bAquChr1.2
## 5                       Midas cichlid genes (Midas_v5)               Midas_v5
## 6                      Giant panda genes (ASM200744v2)            ASM200744v2
## 7         Mexican tetra genes (Astyanax_mexicanus-2.0) Astyanax_mexicanus-2.0
## 8                   Ma's night monkey genes (Anan_2.0)               Anan_2.0
## 9                  Clown anemonefish genes (AmpOce1.0)              AmpOce1.0
## 10                  Orange clownfish genes (Nemo_v1)                Nemo_v1
## 11                         Mallard genes (ASM874695v1)            ASM874695v1
## 12                   Spiny chromis genes (ASM210954v1)            ASM210954v1
## 13                            Duck genes (CAU_duck1.0)            CAU_duck1.0
## 14                   Climbing perch genes (fAnaTes1.2)             fAnaTes1.2
## 15                 American bison genes (Bison_UMD1.0)           Bison_UMD1.0
## 16                 Domestic yak genes (LU_Bosgru_v3.0)         LU_Bosgru_v3.0
## 17          Hybrid - Bos Indicus genes (UOA_Brahman_1)          UOA_Brahman_1
## 18                      Blue whale genes (mBalMus1.v2)            mBalMus1.v2
## 19                        Wild yak genes (BosGru_v2.0)            BosGru_v2.0
## 20            Siamese fighting fish genes (fBetSpl5.2)             fBetSpl5.2
## 21                              Cow genes (ARS-UCD1.2)             ARS-UCD1.2
## 22 Abingdon island giant tortoise genes (ASM359739v1)            ASM359739v1
## 23                      Sooty mangabey genes (Caty_1.0)               Caty_1.0
## 24                        Goldfish genes (ASM336829v1)            ASM336829v1
## 25                  Common carp genes (Cypcar_WagV4.0)         Cypcar_WagV4.0
```

```
mart <- useMart("ENSEMBL_MART_ENSEMBL", dataset = "hsapiens_gene_ensembl")
```

Here we select the *Homo sapiens* dataset

```
#List available attributes
attributes <- biomaRt::listAttributes(mart)
head(attributes,25)
```

```
##                             name                                  description
## 1                ensembl_gene_id                               Gene stable ID
## 2        ensembl_gene_id_version                       Gene stable ID version
## 3          ensembl_transcript_id                         Transcript stable ID
## 4  ensembl_transcript_id_version                 Transcript stable ID version
## 5             ensembl_peptide_id                            Protein stable ID
## 6     ensembl_peptide_id_version                    Protein stable ID version
## 7                ensembl_exon_id                               Exon stable ID
## 8                    description                             Gene description
## 9                chromosome_name                     Chromosome/scaffold name
## 10                start_position                              Gene start (bp)
## 11                 end_position                                Gene end (bp)
## 12                       strand                                       Strand
## 13                         band                               Karyotype band
## 14             transcript_start                        Transcript start (bp)
## 15               transcript_end                          Transcript end (bp)
## 16     transcription_start_site               Transcription start site (TSS)
## 17            transcript_length   Transcript length (including UTRs and CDS)
## 18               transcript_tsl               Transcript support level (TSL)
## 19     transcript_gencode_basic                     GENCODE basic annotation
## 20            transcript_appris                            APPRIS annotation
## 21      transcript_is_canonical                            Ensembl Canonical
## 22       transcript_mane_select        RefSeq match transcript (MANE Select)
## 23 transcript_mane_plus_clinical RefSeq match transcript (MANE Plus Clinical)
## 24           external_gene_name                                    Gene name
## 25         external_gene_source                          Source of gene name
##            page
## 1  feature_page
## 2  feature_page
## 3  feature_page
## 4  feature_page
## 5  feature_page
## 6  feature_page
## 7  feature_page
## 8  feature_page
## 9  feature_page
## 10 feature_page
## 11 feature_page
## 12 feature_page
## 13 feature_page
## 14 feature_page
## 15 feature_page
## 16 feature_page
## 17 feature_page
## 18 feature_page
## 19 feature_page
## 20 feature_page
## 21 feature_page
## 22 feature_page
## 23 feature_page
## 24 feature_page
## 25 feature_page
```

```r
#attributes
```

Then list **some** of the available attributes(The list is too long to add it whole here)[Feel free to uncomment the last line to list all available attributes]

```r
#List available filters
filters <- biomaRt::listFilters(mart)
head(filters,25)
```

```
##                           name
## 1             chromosome_name
## 2                        start
## 3                          end
## 4                   band_start
## 5                    band_end
## 6                marker_start
## 7                  marker_end
## 8                      strand
## 9          chromosomal_region
## 10               with_biogrid
## 11                  with_ccds
## 12                with_chembl
## 13                with_dbass3
## 14                with_dbass5
## 15 with_entrezgene_trans_name
## 16                  with_embl
## 17          with_arrayexpress
## 18             with_genecards
## 19                    with_go
## 20            with_goslim_goa
## 21                  with_hgnc
## 22                   with_hpa
## 23            with_protein_id
## 24          with_ens_lrg_gene
## 25    with_ens_lrg_transcript
##                                                                              description
## 1                                                                Chromosome/scaffold name
## 2                                                                                   Start
## 3                                                                                     End
## 4                                                                              Band Start
## 5                                                                                Band End
## 6                                                                            Marker Start
## 7                                                                              Marker End
## 8                                                                                  Strand
## 9                                                  e.g. 1:100:10000:-1, 1:100000:200000:1
## 10 With BioGRID Interaction data, The General Repository for Interaction Datasets ID(s)
## 11                                                                     With CCDS ID(s)
## 12                                                                   With ChEMBL ID(s)
## 13                                     With DataBase of Aberrant 3' Splice Sites ID(s)
## 14                                     With DataBase of Aberrant 5' Splice Sites ID(s)
## 15                                               With EntrezGene transcript name ID(s)
## 16                                              With European Nucleotide Archive ID(s)
## 17                                                         With Expression Atlas ID(s)
## 18                                                                With GeneCards ID(s)
## 19                                                                       With GO ID(s)
## 20                                                               With GOSlim GOA ID(s)
## 21                                                              With HGNC Symbol ID(s)
## 22                                                      With Human Protein Atlas ID(s)
## 23                                                         With INSDC protein ID ID(s)
## 24                                              With LRG display in Ensembl gene ID(s)
## 25                                        With LRG display in Ensembl transcript ID(s)
```

```
#filters
```

Now we list all available filters we can apply to the information we retrieve, interpreting filters as the way we want to select the information important to us, for example in the next following code we are using HGNC gene symbols to select only the genes we want to retrieve.

```
#Query the database
genes <- getBM(attributes = c("ensembl_gene_id", "hgnc_symbol", "chromosome_name", "start_position", "er
               filters = "hgnc_symbol",
               values = c("BRCA1", "BRCA2","HTT","APOE","PRNP"),
               mart = mart)
genes
```

```
##   ensembl_gene_id hgnc_symbol chromosome_name start_position end_position
## 1 ENSG00000130203        APOE              19       44905791     44909393
## 2 ENSG00000012048       BRCA1              17       43044295     43170245
## 3 ENSG00000139618       BRCA2              13       32315086     32400268
## 4 ENSG00000197386         HTT               4        3041363      3243957
## 5 ENSG00000171867        PRNP              20        4686350      4701590
##   strand
## 1      1
## 2     -1
## 3      1
## 4      1
## 5      1
```

In this Function we are doing the following: We use the getBM function to `query`the BioMart(hence the BM in the function's name) we select the attributes we want to see in the attributes parameter(for those who don't know everything inside the parenthesis in a function is called a parameter) we filter the information by gene symbol as noted before and list the genes we want to retrieve using the `c` function to concatenate all the gene symbols and last select the mart we are accessing

With the previous code, we can see how easy it is to retrieve data from the Ensembl database using the biomaRt package. This is just a small example of what you can do with this package, you can retrieve a vast amount of data from the Ensembl database using this package, and you can also retrieve data from other databases like the Vega Genome database.

Entrez

It's also possible to retrieve data from the NCBI with another package called rentrez, this package is very similar to biomaRt.

```
# Check if the package is installed and install it if it's not
if (!requireNamespace("rentrez", quietly = TRUE)) {
  install.packages("rentrez")
}
```

To exemplify the use of this package we will retrieve the information of the gene *BRCA1* from the NCBI database.

```
# Specify the gene name
gene_name <- "(Homo sapiens[Organism]) AND PRNP[Gene Name] "

# Use the esearch function to search the gene in the nucleotide database
search_results <- entrez_search(db="gene", term=gene_name)

# Use the esummary function to retrieve the gene summary
gene_summary <- entrez_summary(db="gene", id=search_results$ids[1])

print("Chromosome")
```

```
## [1] "Chromosome"
```

```
gene_summary$chromosome
```

```
## [1] "20"
```

```
print("Gene Nomenclature name")
```

```
## [1] "Gene Nomenclature name"
```

```
gene_summary$nomenclaturename
```

```
## [1] "prion protein"
```

```
print("Description")
```

```
## [1] "Description"
```

```
gene_summary$summary
```

```
## [1] "The protein encoded by this gene is a membrane glycosylphosphatidylinositol-anchored glycoprote
```

SRA

It's also possible to utilize R to obtain information from the SRA (Sequence Read Archive) which is a repository

```
py$pydatasets <-  r_to_py(datasets)
py_run_string("
import pandas as pd
print(type(pydatasets))
print(pydatasets)
")
```

```
## <class 'pandas.core.frame.DataFrame'>
##                           dataset  ...                            version
## 0    abrachyrhynchus_gene_ensembl  ...                         ASM259213v1
## 1        acalliptera_gene_ensembl  ...                          fAstCal1.2
## 2      acarolinensis_gene_ensembl  ...                         AnoCar2.0v2
## 3       acchrysaetos_gene_ensembl  ...                          bAquChr1.2
## 4       acitrinellus_gene_ensembl  ...                            Midas_v5
## ..                            ...  ...                                 ...
## 209           vpacos_gene_ensembl  ...                            vicPac1
## 210         vursinus_gene_ensembl  ...  bare-nosed_wombat_genome_assembly
## 211          vvulpes_gene_ensembl  ...                          VulVul2.2
## 212       xmaculatus_gene_ensembl  ...               X_maculatus-5.0-male
## 213      xtropicalis_gene_ensembl  ...                       UCB_Xtro_10.0
##
## [214 rows x 3 columns]
```