CSI5180. Machine Learning for Bioinformatics Applications

Essential **Bioinformatics** Skills

by

Marcel Turcotte

Version **November 6, 2019**

Preamble

Preamble

Essential Bioinformatics Skills

The lecture gives an overview of the available resources that are essential for bioinformatics projects. This includes the main databases, software applications, programming languages and computing environments. We also emphasize the skills that are essential to produce robust and reproducible results.

General objective :

- **Summarize** the essential resources for conducting a bioinformatics project

Learning objectives

- **Describe** the best practices for handling large bioinformatics projects
- **Introduce** essential tools
- **Present** the major repositories and file formats, along with the command line and REST API access

Reading:

- **See** below

Plan

1. Preamble

2. Literature

3. Guidelines

4. Computing Environment

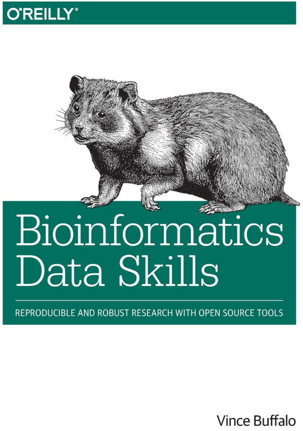
5. Data

6. REST

7. Prologue

Literature

Bioinformatics Data Skills


O'REILLY

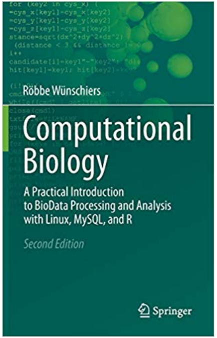
Bioinformatics Data Skills

REPRODUCIBLE AND ROBUST RESEARCH WITH OPEN SOURCE TOOLS

Vince Buffalo

- Vince Buffalo. Bioinformatics Data Skills: Reproducible and Robust Research with Open Source Tools. O'Reilly Media, 2015.

A Practical Introduction to. . .



Röbbe Wünschiers

Computational Biology

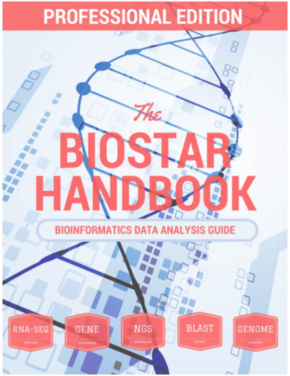
A Practical Introduction to BioData Processing and Analysis with Linux, MySQL, and R

Second Edition

Springer

- Röbbe Wünschiers. Computational Biology - A Practical Introduction to BioData Processing and Analysis with Linux, MySQL, and R. Springer, 2013. (https://link.springer.com/book/10.1007/978-3-642-34749-8)

The Biostar Handbook



- **The Biostar Handbook: Bioinformatics data analysis guide, 2019**
- https://biostar.myshopify.com

Ten (10) simple rules for. . .

- Sandve, G. K., Nekrutenko, A., Taylor, J. & Hovig, E. Ten Simple Rules for Reproducible Computational Research. PLoS Comput Biol 9, (2013).
- Boulesteix, A.-L. Ten simple rules for reducing overoptimistic reporting in methodological computational research. PLoS Comput Biol 11, e1004191 (2015).
- Prlic, A. & Procter, J. B. Ten Simple Rules for the Open Development of Scientific Software. PLoS Comput Biol 8, e1002802 (2012).
- Perez-Riverol, Y. et al. Ten Simple Rules for Taking Advantage of Git and GitHub. PLoS Comput Biol 12, e1004947 (2016).
- Sholler, D. et al. Ten simple rules for helping newcomers become contributors to open projects. PLoS Comput Biol 15, e1007296 (2019).
- Rule, A. et al. Ten simple rules for writing and sharing computational analyses in Jupyter Notebooks. PLoS Comput Biol 15, e1007007 (2019).

Ten (10) simple rules for. . .

- Osborne, J. M. et al. Ten simple rules for effective computational research. PLoS Comput Biol 10, e1003506 (2014).
- Elofsson, A. et al. Ten simple rules on how to create open access and reproducible molecular simulations of biological systems. PLoS Comput Biol 15, e1006649 (2019).
- Lee, B. D. Ten simple rules for documenting scientific software. PLoS Comput Biol 14, e1006561 (2018).
- Carey, M. A. & Papin, J. A. Ten simple rules for biologists learning to program. PLoS Comput Biol 14, e1005871 (2018).
- Zook, M. et al. Ten simple rules for responsible big data research. PLoS Comput Biol 13, e1005399 (2017).

(One more) Definition

“**Bioinformatics** is conceptualizing biology in terms of **macromolecules** (in the sense of physical-chemistry) and then applying “**informatics**” techniques (derived from disciplines such as applied maths, computer science, and statistics) to **understand** and **organize** the information associated with these molecules, on a large-scale.”

Luscombe, N. M., Greenbaum, D. & Gerstein, M.
What is bioinformatics? A proposed definition and overview of the field.
Methods of information in medicine 40, 346358 (2001).

Guidelines

Robust research (Vince Buffalo)

- Pay attention to your experimental design
- Write code for humans, write code for computers
- Let the computer do the work
- Write down your assumptions and test them (unit testing)
- Use existing libraries
- Treat data as read-only

Reproducible research (Vince Buffalo)

- Share your **source code** and your **data**
- Meta-data:
  - **Versions** of the **software** and **databases** you are using
  - Write down the parameters or better yet, **make it a script**
  - One README file directory
- Make **figures**, **statistics**, and **tables** from scripts
- Not only is this more scientific, **it is almost certain that you will need to redo your analyses!**

Computing Environment

UNIX

- Both, **Bioinformatics** and **Machine Learning**, favour **UNIX**
- Quoting François Cholette (Deep Learning with Python): "Youll need access to a UNIX machine; it's possible to use Windows, too, but I don't recommend it"
- **Compute Canada** (`https://docs.computecanada.ca`)
  - **Cedar** - 58,416 CPU cores and 584 GPU devices
  - **Graham** - 36,160 cores and 320 GPU devices
  - **Béluga** - 34,880 cores and 688 GPU devices
  - **Niagara** - 61,920 cores

Access to UNIX

- Your **laptop** or **workstation**
  - As primary or secondary OS (dual boot, USB key, etc.)
  - In a **virtual machine**
    (VMWare is free for EECS students, VirtualBox is also free)
  - Windows Subsystem for Linux Installation Guide for Windows 10
    (`https://docs.microsoft.com/en-us/windows/wsl/install-win10`)
- **Cloud**
  - I have vouchers for Google Cloud Platform and Amazon (just ask me)
- **Ubuntu** is a popular distribution, but there are many others

UNIX key concepts

- **Modularity**
  - "This is the **Unix philosophy**: Write programs that **do one thing** and do it well. Write programs to **work together**. Write programs to handle text **streams**, because that is a universal interface." — Doug McIlory
- The **file system** plays a central role
  - `/dev/null`, `/dev/random`, **/dev/zero**
    - `$ head -c 10 /dev/zero > test10bytes.dat`
- The **command line**
  - `$ grep -c '>' input.fasta`
  - Shell (anatomy of a script, the magic line, and more)
  - Redirection
  - Pipe
  - https://www.ks.uiuc.edu/Training/Tutorials/Reference/unixprimer.html

Conda/Anaconda/Bioconda

- https://conda.io
  - **Conda** is a package, dependency and environment management for any programming language (Python, R, Ruby, Lua, Scala, Java, and more)
- https://anaconda.org
  - **Anaconda** is a package management service, primarily for **Python** and **R**, hundreds of packages such as numpy, scipy, scikit-learn, keras, tensorflow
- https://bioconda.github.io
  - **Bioconda** is a channel for the conda package manager specializing in bioinformatics software.

Using conda/anaconda/bioconda

- `$ conda create -n csi5180`
- `$ conda install -n csi5180 keras`
- `$ conda activate csi5180`
- `$ conda install bwa`
- `$ conda deactivate`
- `$ conda update --all`

Other considerations

- Consider using a (distributed) version control system
  - Git/GitHub has become the *de facto* standard
  - Features
    - Manage changes in your documents
    - In a distributed version control system, each developer has its own version of the source code
    - Multiple contributors
    - Creating/merging multiple branches
  - `https://git-scm.com/doc`

Data

Major repositories

Annotated/assembled nucleotide sequence

- National Center for Biotechnology Information (**NCBI**)
  - https://www.ncbi.nlm.nih.gov
- European Bioinformatics Institute (**EBI**)
  - https://www.ebi.ac.uk
- DNA Data Bank of Japan (**DDBJ**)
  - https://www.ddbj.nig.ac.jp/

See also: International Nucleotide Sequence Database Collaboration (http://www.insdc.org)

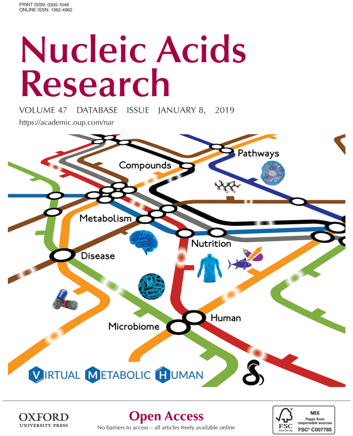
Major repositories (continued)

- **GenBank:** annotated and identified DNA sequence information
- **SRA (Short Read Archive):** measurements from high throughput sequencing experiments
- **UniProt (Universal Protein Resource):** protein sequence data
- **PDB (Protein Data Bank):** 3D structural information of macromolecules

Other data sources?

- UCSC Genome Browser
- FlyBase (Drosophila [fruit fly], WormBase (nematode), SGD: Saccharomyces Genome Database, TAIR (Arabidopsis), EcoCyc (Encyclopedia of E. coli Genes and Metabolic Pathways), etc.
- RNA-Central: meta-database

Nucleic Acids Research (NAR)



PRINT ISSN: 0305-1048
ONLINE ISSN: 1362-4962

Nucleic Acids Research

VOLUME 47 DATABASE ISSUE JANUARY 8, 2019
https://academic.oup.com/nar

OXFORD
UNIVERSITY PRESS

Open Access
No barriers to access – all articles freely available online

MIX
Paper from responsible sources
FSC
www.fsc.org
FSC® C007785

Each year, **NAR**, a high-impact journal, publishes its “database issue”:

- https://academic.oup.com/nar/issue/47/D1

Major file formats (biostar)

- Data that captures **prior knowledge** (aka reference: **FASTA**, **GFF**, **BED**)
- **Experimentally obtained** data (aka sequencing reads: **FASTQ**)
- Data **generated** by the analysis (aka results: **BAM**, **VCF**, formats from point 1 above, and many nonstandard formats)

Entrez Direct

```
$ conda install -c bioconda entrez-direct
```

GENBANK

```
$ efetch -db nuccore -id NM_000020 -format gb | less
```

```
LOCUS       NM_000020               4177 bp    mRNA    linear   PRI 16-SEP-2019
DEFINITION  Homo sapiens activin A receptor like type 1 (ACVRL1), transcript
            variant 1, mRNA.
ACCESSION   NM_000020
VERSION     NM_000020.3
KEYWORDS    RefSeq; RefSeq Select.
SOURCE      Homo sapiens (human)
  ORGANISM  Homo sapiens
            Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi;
            Mammalia; Eutheria; Euarchontoglires; Primates; Haplorrhini;
            Catarrhini; Hominidae; Homo.
REFERENCE   1  (bases 1 to 4177)
  AUTHORS   Leng H, Zhang Q and Shi L.
  TITLE     [Gene diagnosis and treatment of hereditary hemorrhagic
(...)
```

GENBANK (continued)

```
(...)
FEATURES             Location/Qualifiers
     source          1..4177
                     /organism="Homo sapiens"
                     /mol_type="mRNA"
                     /db_xref="taxon:9606"
                     /chromosome="12"
                     /map="12q13.13"
     gene            1..4177
                     /gene="ACVRL1"
                     /gene_synonym="ACVRLK1; ALK-1; ALK1; HHT; HHT2; ORW2;
                     SKR3; TSR-I"
                     /note="activin A receptor like type 1"
                     /db_xref="GeneID:94"
                     /db_xref="HGNC:HGNC:175"
                     /db_xref="MIM:601284"
     exon            1..192
                     /gene="ACVRL1"
                     /gene_synonym="ACVRLK1; ALK-1; ALK1; HHT; HHT2; ORW2;
(...)
```

GENBANK (continued)

```
(...)
ORIGIN
        1 cccagtcccg ggaggctgcc gcgccagctg cgccgagcga gcccctcccc ggctccagcc
       61 cggtccgggg ccgcgcccgg accccagccc gccgtccagc gctggcggtg caactgcggc
      121 cgcgcggtgg aggggaggtg gccccggtcc gccgaaggct agcgccccgc cacccgcaga
      181 gcgggcccag agggaccatg accttgggct cccccaggaa aggccttctg atgctgctga
      241 tggccttggt gacccaggga gaccctgtga agccgtctcg gggcccgctg gtgacctgca
(...)
     4081 aaattacact tctcgtacct ggagacgctg tttgtgggag cactgggctc atgcctggca
     4141 cacaataggt ctgcaataaa ccatggttaa atcctga
//
```

FASTA

```
$ efetch -db nuccore -id NM_000020 -format fasta | less
```

```
>NM_000020.3 Homo sapiens activin A receptor like type 1 (ACVRL1), transcript variant 1, mR
CCCAGTCCCGGGAGGCTGCCGCGCCAGCTGCGCCGAGCGAGCCCCTCCCCGGCTCCAGCCCGGTCCGGGG
CCGCGCCCGGACCCCAGCCCGCCGTCCAGCGCTGGCGGTGCAACTGCGGCCGCGCGGTGGAGGGGAGGTG
GCCCCGGTCCGCCGAAGGCTAGCGCCCCGCCACCCGCAGAGCGGGCCCAGAGGGACCATGACCTTGGGCT
CCCCCAGGAAAGGCCTTCTGATGCTGCTGATGGCCTTGGTGACCCAGGGAGACCCTGTGAAGCCGTCTCG
GGGCCCGCTGGTGACCTGCACGTGTGAGAGCCCACATTGCAAGGGGCCTACCTGCCGGGGGGCCTGGTGC
ACAGTAGTGCTGGTGCGGGAGGAGGGGAGGCACCCCCAGGAACATCGGGGCTGCGGGAACTTGCACAGGG
AGCTCTGCAGGGGGCGCCCCACCGAGTTCGTCAACCACTACTGCTGCGACAGCCACCTCTGCAACCACAA
CGTGTCCCTGGTGCTGGAGGCCACCCAACCTCCTTCGGAGCAGCCGGGAACAGATGGCCAGCTGGCCCTG
ATCCTGGGCCCCGTGCTGGCCTTGCTGGCCCTGGTGGCCCTGGGTGTCCTGGGCCTGTGGCATGTCCGAC
(...)
GGCCCAATGGCCAGGGAGTGAAGGAGGTGGCGTTGCTGAGAGCAGTCTGCACATGCTTCTGTCTGAGTGC
AGGAAGGTGTTCCAGGGTCGAAATTACACTTCTCGTACCTGGAGACGCTGTTTGTGGGAGCACTGGGCTC
ATGCCTGGCACACAATAGGTCTGCAATAAACCATGGTTAAATCCTGA
```

GFF/GTF/BED

- **Interval** formats
- Tab delimited
- Chromosomal coordinate, start, end, strand, and more
- `https://useast.ensembl.org/info/website/upload/gff3.html`

BED

3 columns:

```
chr7    127471196    127472363
chr7    127472363    127473530
chr7    127473530    127474697
```

6 columns:

```
chr1    134212701    134230065    Nuak2      8    +
chr1    134212701    134230065    Nuak2      7    +
chr1    33510655     33726603     Prim2,    14    -
chr1    25124320     25886552     Bai3,     31    -
```

Bedtools

> *"Collectively, the bedtools utilities are a* ***swiss-army knife*** *of tools for a wide-range of genomics analysis tasks. The most widely-used tools enable* ***genome arithmetic****: that is, set theory on the genome. For example, bedtools allows one to intersect, merge, count, complement, and shuffle genomic intervals from multiple files in widely-used genomic file formats such as BAM, BED, GFF/GTF, VCF."*

```
$ conda install -c bioconda bedtools
```

- https://www.biostars.org/p/17162/

.2bit

```
$ conda install -c bioconda ucsc-twobittofa
```

```
$ URL=http://hgdownload.cse.ucsc.edu/goldenpath/mm9/bigZips/mm9.2bit
$ twoBitToFa -udcDir=. $URL1 stdout > mm9.fa
```

```
$ URL=http://hgdownload.cse.ucsc.edu/goldenPath/mm9/bigZips/mm9.chrom.sizes
$ curl $URL  > mm9.chromsizes
```

Bedtools (continued)

Given **genes.bed**:

```
chr1    134212701    134230065    Nuak2    8     +
chr1    134212701    134230065    Nuak2    7     +
chr1    33510655     33726603     Prim2    14    -
chr1    25124320     25886552     Bai3     31    -
```

```
$ bedtools flank -i genes.bed -g mm9.chromsizes -l 2000 -r 0 -s
```

```
chr1    134210701    134212701     Nuak2    8     +
chr1    134210701    134212701     Nuak2    7     +
chr1    33726603     33728603      Prim2    14    -
chr1    25886552     25888552      Bai3     31    -
```

```
$ bedtools getfasta -fi mm9.fa -bed promoters.bed -fo promoters.fa
```

promoters.fa

```
>chr1:134210701-134212701
TTCTGGCACTTGGTTGTTCT...GTTTTATAGCAATTCGGAAC
>chr1:134210701-134212701
TTCTGGCACTTGGTTGTTCT...GTTTTATAGCAATTCGGAAC
>chr1:33726603-33728603
TCTCCCAGTGGCGGGAGAGT...ATTTATTTTTATGTTTATAA
>chr1:25886552-25888552
TTGCGCCTTATCCAAGTGAA...TCCCAGGAACAAATCACCAG
```

Creating a script automating our work

- Let’s now create a script capturing all this information

Magic line (shebang)

- In a Unix-like operating system, the content of an executable is passed to the interpreter designated on the **magic line**.

```
#! /bin/bash
```

- I am saving this to a file called `01_get_data.sh`
- Then, I make it executable

```
$ chmod u+x 01_get_data.sh
```

Test your assumptions

- You can test for the presence of absence of a file or a directory

```
#! /bin/bash

INPUT=genes.bed

if [ ! -f $INPUT ]; then
    echo "file not found: $INPUT"
    exit 1
fi
```

Temporary space

- Sometimes you don't want to create temporary files in your user account.
    - These temporary files might be big and you don't want them to be saved by the backup system or your quota might not allow you to save them in your user space.
- Do not use `/tmp/`, this is temporary storage for the operating system, and sometimes the partition is rather small.
- Use `/var/tmp/` or a designated space, such as `/scratch`.
    - Beware! The system will automatically remove those files after a given period of time.

```bash
#! /bin/bash

# Sample Bash script to download a genome and extract information

INPUT=genes.bed

if [ ! -f $INPUT ]; then
   echo "file not found: $INPUT"
   exit 1
fi

PROJECT=csi5180-demo

# Process ID and time stamp as suffix
TMP_DIR=/var/tmp/$PROJECT-`date +"%FT%H%M%S"`-$$

if [ -d TMP_DIR ]; then
   echo "$TMP_DIR exists!"
   exit 1
fi
```

```
# Creating the temporary directory
mkdir $TMP_DIR

# The URL where the mouse genome version 9 (MM9) can be found
MM9_URL=http://hgdownload.cse.ucsc.edu/goldenpath/mm9/bigZips/mm9.2bit

# Where to save the mouse genome as a fasta file
MM9_FILE_NAME=$TMP_DIR/mm9.fa

# Download an uncompress the genome
twoBitToFa -udcDir=$TMP_DIR $MM9_URL stdout > $MM9_FILE_NAME

# URL of the file containing the size of each chromosome
MM9_SIZE_URL=http://hgdownload.cse.ucsc.edu/goldenPath/mm9/bigZips/mm9.chrom.

MM9_SIZE_FILE_NAME=$TMP_DIR/mm9.chromsizes

# Downloading the size file (to the current directory)
curl $MM9_SIZE_URL > $MM9_SIZE_FILE_NAME
```

```
# Calculating the coordinates of the promoter regions
 bedtools flank -i $INPUT -g $MM9_SIZE_FILE_NAME -l 2000 -r 0 -s > promoters.b

# Extracting the promoters
 bedtools getfasta -fi $MM9_FILE_NAME -bed promoters.bed -fo promoters.fa

# Cleaning
rm -rf $TMP_DIR

# EOF
```

REST

Representational state transfer (REST)

- Client and server interactions using HTTP (hypertext transfer protocol)
- Madeira, F. et al. The EMBL-EBI search and sequence analysis tools APIs in 2019. Nucleic Acids Res 47, W636W641 (2019).
- Tarkowska, A. et al. Eleven quick tips to build a usable REST API for life sciences. PLoS Comput Biol 14, e1006542 (2018).
- `https://www.ebi.ac.uk/training/online/course/ensembl-rest-api`
- `https://www.ncbi.nlm.nih.gov/home/develop/api/`
- `https://rest.ensembl.org`
- `https://www.encodeproject.org/help/rest-api/`

Examples:

- /sequence/id/ENST00000288602?type=cds;content-type=text/x-fasta
- /sequence/id/ENST00000288602?type=cds;content-type=text/x-fasta;start=10;end=110

ENSEMBL: GET sequence/id/:id

- https://rest.ensembl.org/documentation/info/sequence_id

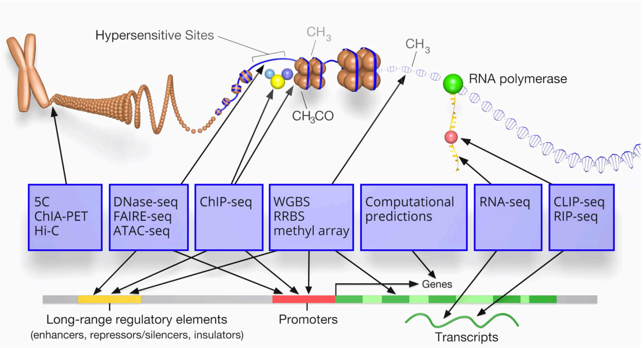
```python
import requests, sys

server = "https://rest.ensembl.org"
ext = "/sequence/id/ENST00000288602?type=cdna"

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

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


print(r.text)
```

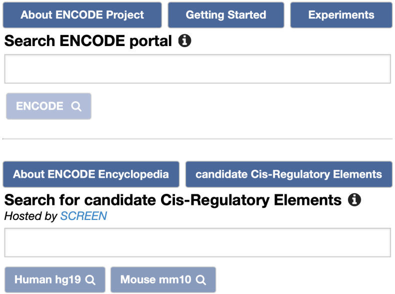
A Python script can also be made executable

```python
#!/usr/bin/env python3

import requests, sys

server = "https://rest.ensembl.org"
ext = "/sequence/id/ENST00000288602?type=cdna"

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

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

print(r.text)
```

ENCODE

ENCODE: Encyclopedia of DNA Elements





Search ENCODE portal

Search for candidate Cis-Regulatory Elements

Hosted by SCREEN

https://www.encodeproject.org

Pipelines

- https://www.encodeproject.org/pipelines/
- https://www.encodeproject.org/chip-seq/transcription_factor/
- https://github.com/ENCODE-DCC/chip-seq-pipeline

Discussion groups

- https://bioinformatics.stackexchange.com/
- https://www.biostars.org/

Tutorials

- https://www.nihlibrary.nih.gov/services/bioinformatics-support/online-bioinformatics-tutorials
- https://www.biostars.org/

Prologue

Summary

- Strive to make your research **robust** and **reproducible**

Summary

- Strive to make your research **robust** and **reproducible**
- **UNIX** is the preferred environment for **bioinformatics** and **machine learning**

Summary

- Strive to make your research **robust** and **reproducible**
- **UNIX** is the preferred environment for **bioinformatics** and **machine learning**
- Conda/Anaconda/Bioconda will simplify your life tremendously

Summary

- Strive to make your research **robust** and **reproducible**
- **UNIX** is the preferred environment for **bioinformatics** and **machine learning**
- Conda/Anaconda/Bioconda will simplify your life tremendously
- NCBI/EBI/DDBJ are the major repositories for bioinformatics data

Summary

- Strive to make your research **robust** and **reproducible**
- **UNIX** is the preferred environment for **bioinformatics** and **machine learning**
- Conda/Anaconda/Bioconda will simplify your life tremendously
- NCBI/EBI/DDBJ are the major repositories for bioinformatics data
- There are many **specialized** bioinformatics repositories

Summary

- Strive to make your research **robust** and **reproducible**
- **UNIX** is the preferred environment for **bioinformatics** and **machine learning**
- Conda/Anaconda/Bioconda will simplify your life tremendously
- NCBI/EBI/DDBJ are the major repositories for bioinformatics data
- There are many **specialized** bioinformatics repositories
- GenBank, Fasta, and BED are examples of file formats

Summary

- Strive to make your research **robust** and **reproducible**
- **UNIX** is the preferred environment for **bioinformatics** and **machine learning**
- Conda/Anaconda/Bioconda will simplify your life tremendously
- NCBI/EBI/DDBJ are the major repositories for bioinformatics data
- There are many **specialized** bioinformatics repositories
- GenBank, Fasta, and BED are examples of file formats
- Entrez Direct/REST

Summary

- Strive to make your research **robust** and **reproducible**
- **UNIX** is the preferred environment for **bioinformatics** and **machine learning**
- Conda/Anaconda/Bioconda will simplify your life tremendously
- NCBI/EBI/DDBJ are the major repositories for bioinformatics data
- There are many **specialized** bioinformatics repositories
- GenBank, Fasta, and BED are examples of file formats
- Entrez Direct/REST
- Pipelines

Next module

- **Fundamentals** of Machine Learning

References

Vince Buffalo.
Bioinformatics Data Skills: Reproducible and Robust Research with Open Source Tools.
O'Reilly Media, 2015.

Röbbe Wünschiers.
Computational Biology - A Practical Introduction to BioData Processing and Analysis with Linux, MySQL, and R.
Springer, 2013.

The Biostar Handbook: Bioinformatics data analysis guide, 2019.
Shopify, 2019.

Marcel **Turcotte**

`Marcel.Turcotte@uOttawa.ca`

School of Electrical Engineering and **Computer Science** (EECS)
University of Ottawa