



# Tutorial

## Using Computational Tools and Programs for Bioinformatics

Presented by: Eric Arezza



# Learning Goals

## Computer Literacy and Setup

### 1) Fundamentals

- Navigate in the terminal with command-line operations
  - Simple exploration of data
- Connect to remote computers and transfer files

### 2) Programming essentials

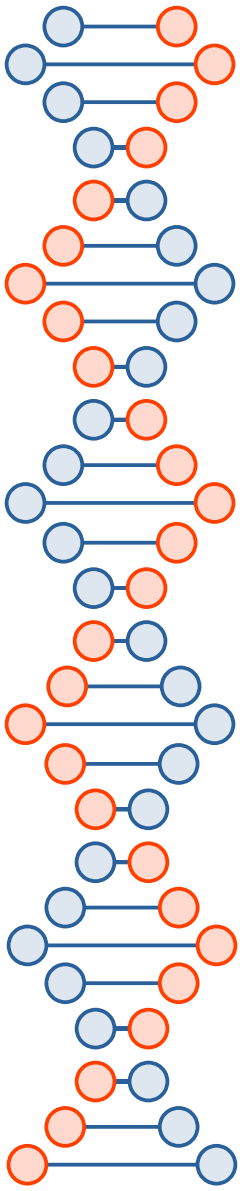
- Basics of Shell, Python, and R
- IDEs and managing environments
- Jupyter notebooks

### 3) GitHub



# Computer Literacy and Setup

- Everyone has different levels of computer literacy depending on their jobs, hobbies, experiences, etc...
- This tutorial is meant to provide an introduction to:
  - Be familiar with computers from a programmer's perspective
  - Become comfortable with seeing/using different tools
  - Understand how you can use lab scripts for bioinformatics
- This tutorial is NOT meant to:
  - Make you a computer expert or programmer
  - Be daunting, confusing, or boring



# Fundamentals

- Navigate in the terminal with command-line operations
  - Open terminal with Ctrl+Alt+T (or find with search toolbar)

A screenshot of a terminal window. The title bar at the top reads "eric@thinkpad: ~". The terminal content shows the prompt "eric@thinkpad: ~\$". Four orange arrows point from labels below to parts of the prompt: "user" points to "eric", "computer" points to "thinkpad", "current directory ~" points to "~", and "prompt \$" points to "\$".

```
eric@thinkpad: ~$
```

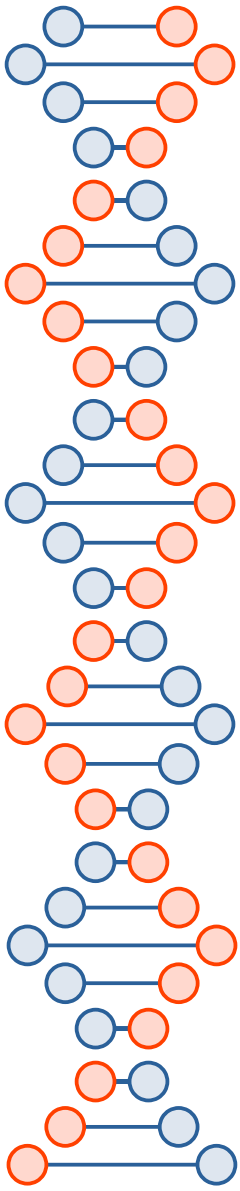
user            computer            current directory ~            prompt \$



# Fundamentals

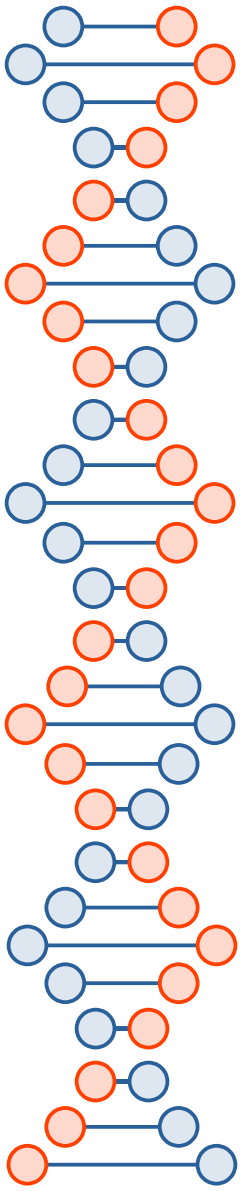
Command	Useful Options	Description
<i>pwd</i>		Print current directory
<i>ls</i>	<i>-l -a -h</i>	List files
<i>cd</i>		Change directory
<i>mkdir</i>		Make directory
<i>rm</i>	<i>-r</i>	Remove
<i>mv</i>		Move
<i>cp</i>	<i>-r</i>	Copy
<i>cat</i>		Concatenate contents of file
<i>head/tail</i>	<i>-n</i>	Show head or tail contents of file
<i>wc</i>		Count file contents
<i>nano</i>		File editor

\*\*\*Different commands for Windows\*\*\*



# Fundamentals

- Simple exploration of data
  - Get file sizes
  - Count lines in file
  - Search for pattern in file (more advanced, “|” pipe and grep <pattern>)
  - Output to new file with “>”
- Tips:
  - Use tab-completion
  - Use up/down arrow for previously executed commands
  - Use “../” to move back up file tree or “~” for home directory
  - Use “\*” wildcard character for multiple matches



# Fundamentals

- Connect to remote computers using *ssh*:  
`ssh <username>@<remoteserver.ca>`
- Transfer files using *scp* (alternative is *rsync*):
  - To remote computer  
`scp myFile.txt user@remote.ca:~/files/`
  - From remote computer  
`scp -r user@remote.ca:~/files/ ./`

# Programming Essentials

- Shell



**UNIX**

- Python



python™

- R



- Open-source, free to use
- All used to perform functions instructed by the programmer
  - Many functions are pre-written and packaged for download + installation
  - Same functions may be written in either or all languages
- Differences in syntax
- Executes code sequentially
- Great for automation



# Programming (Coding) Languages

## Shell – on Unix (Linux, macOS)



# UNIX

- What we have been using thus far *interactively* in the terminal, can be much more elaborate and saved to file
- Script file extension **.sh**
- Script starts with **#!/bin/bash**
- Execute by
  - sh myScript.sh*
  - ./myScript.sh (if in same directory)*

## PowerShell – on Windows



Microsoft



- Has similar utility, but different commands
- Windows file paths delimited by “\” instead of “/”

# Programming (Coding) Languages

## Python



- Script file extension **.py**
- Script starts with **`#!/bin/python`**
- Execute by  
*python myScript.py*

## R



- Script file extension **.R**
- Script starts with **`#!/bin/R`**
- Execute by  
*Rscript myScript.R*

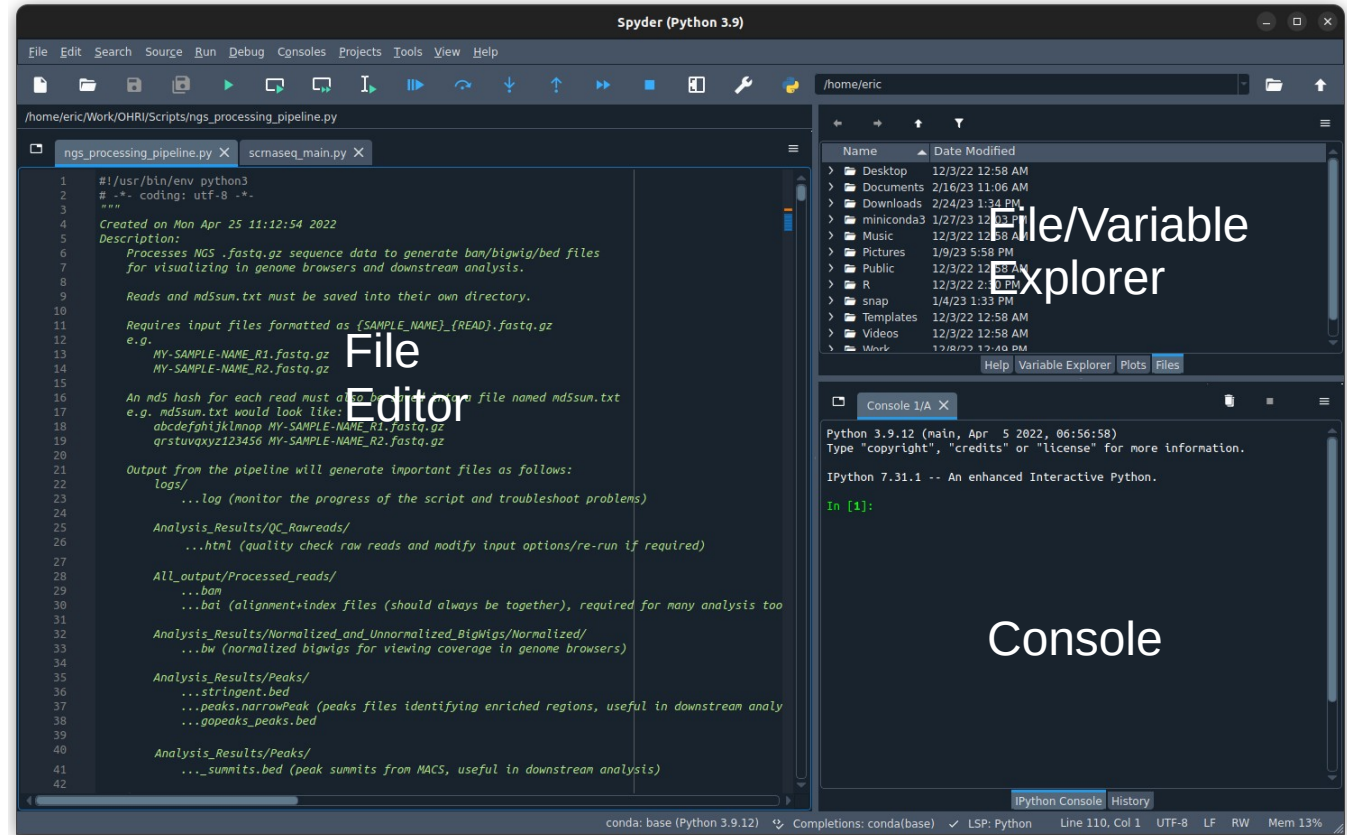
# Programming IDEs

- “Integrated Development Environment”
- Complete graphical user interfaces with development tools
  - Debugging/troubleshooting
  - Profilers for performance/efficiency
  - Viewing plots
  - Etc...
- IDEs shown here:
  - Spyder (for Python)
  - Rstudio (for R)



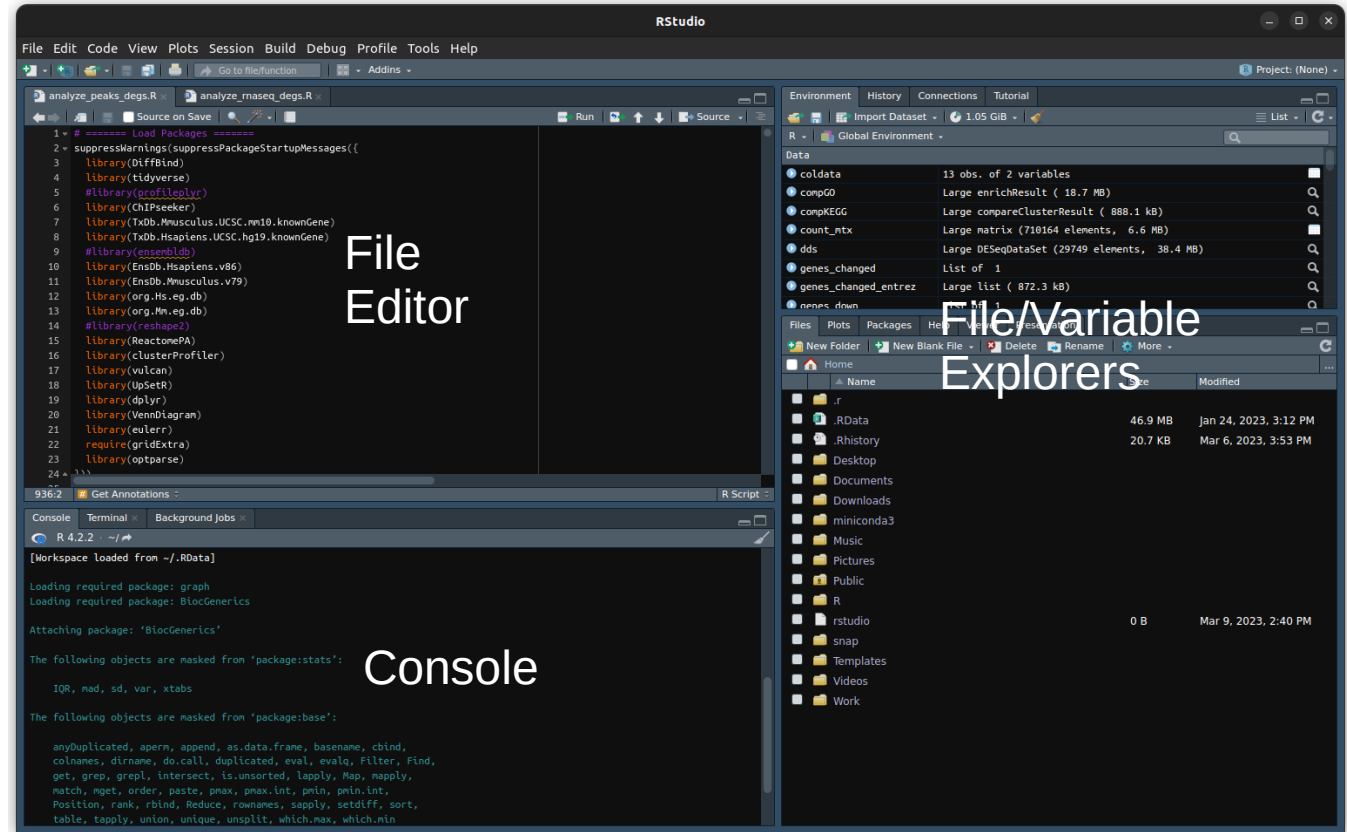
# Programming IDEs

Spyder




# Programming IDEs

RStudio





# Managing Environments

- Typically, do not want all installed packages/libraries in the same environment
  - Conflicts between packages, functions, methods
  - Conflicts between versions
  - Etc...
- More often required with **Python**, not as much with R
- Two commonly used package managers, install:
  - Virtualenv
  -  **CONDA**



# Managing Environments

## Virtualenv Usage

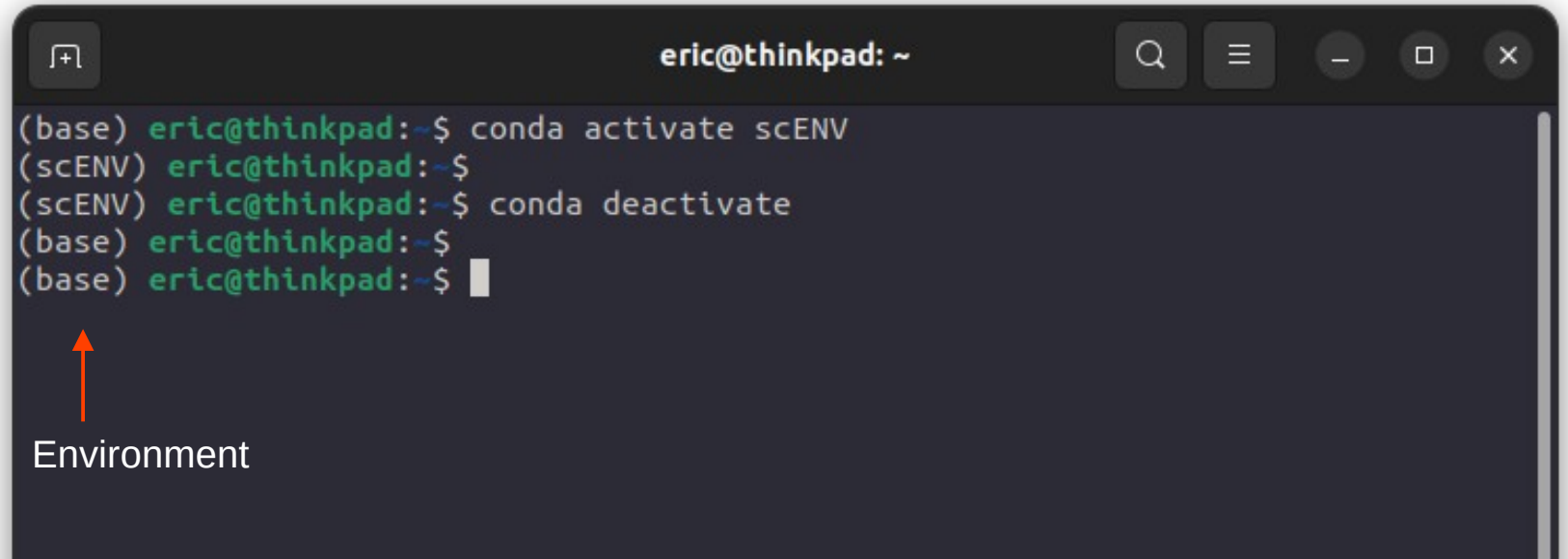
- Create  
`venv -m myENV`
- Activate  
`source myENV/bin/activate`
- Install packages  
`pip install <package-name>`
- View packages  
`pip list`
- Deactivate  
`deactivate`

## Conda Usage

- Create  
`conda create -n myENV`
- Activate  
`conda activate myENV`
- Install packages  
`conda install <package-name>`
- View packages  
`conda list`
- Deactivate  
`conda deactivate`

# Managing Environments

- Only the functions/programs installed in the current environment will be usable while in that environment



A terminal window titled "eric@thinkpad: ~" with standard window controls. The terminal shows a sequence of commands to activate and deactivate a Conda environment named "scENV". The prompt changes from "(base)" to "(scENV)" when the environment is activated. An orange arrow points from the word "Environment" below the terminal to the "(scENV)" prompt in the second line of the command history.

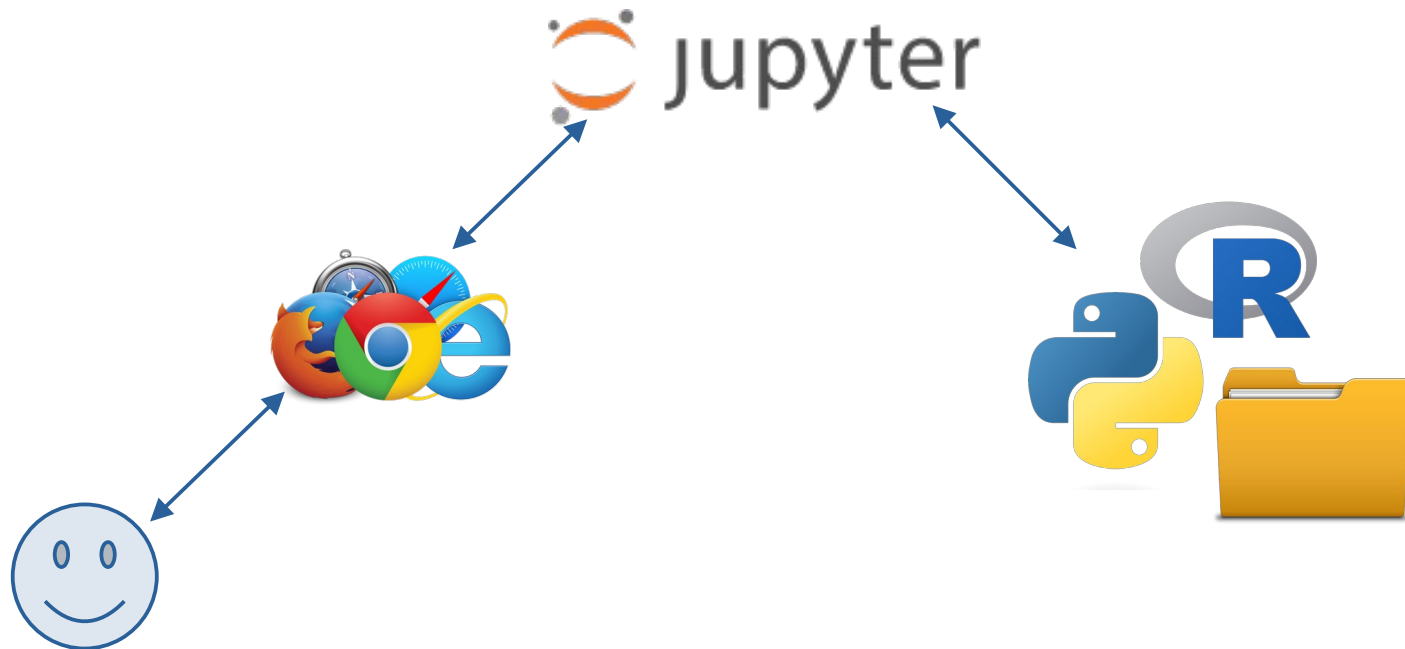
```
(base) eric@thinkpad:~$ conda activate scENV
(scENV) eric@thinkpad:~$
(scENV) eric@thinkpad:~$ conda deactivate
(base) eric@thinkpad:~$
(base) eric@thinkpad:~$
```

Environment



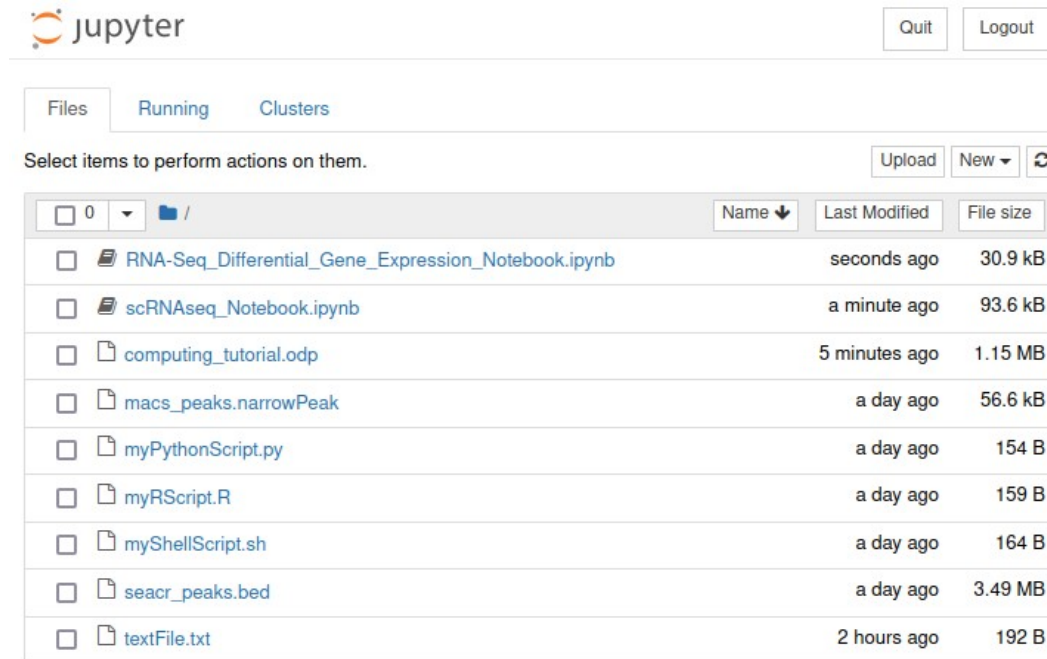
# Jupyter Notebooks

- Jupyter provides a web browser interface to the file system and programming languages from which it's run



# Jupyter Notebooks

- Notebooks enable you to interact with code, view plots, and can contain any type of (HTML) web page content
- File extension **.ipynb**



A screenshot of the Jupyter Notebook web interface. At the top, there is a 'jupyter' logo and two buttons: 'Quit' and 'Logout'. Below this is a navigation bar with three tabs: 'Files' (selected), 'Running', and 'Clusters'. Under the 'Files' tab, there is a message 'Select items to perform actions on them.' and two buttons: 'Upload' and 'New' with a dropdown arrow. Below this is a table listing files and folders in the current directory. The table has columns for 'Name', 'Last Modified', and 'File size'. The files listed include two .ipynb files, several .odp, .py, .R, .sh, .bed, and .txt files.

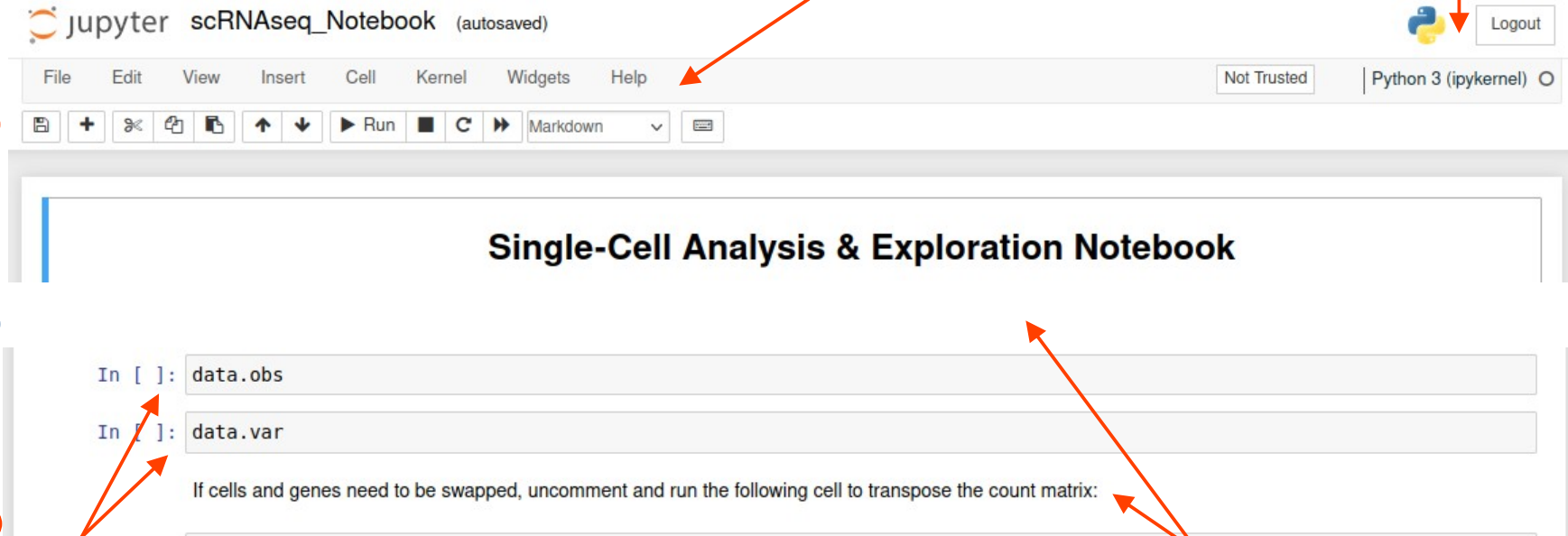
	Name	Last Modified	File size
<input type="checkbox"/>	/		
<input type="checkbox"/>	RNA-Seq_Differential_Gene_Expression_Notebook.ipynb	seconds ago	30.9 kB
<input type="checkbox"/>	scRNAseq_Notebook.ipynb	a minute ago	93.6 kB
<input type="checkbox"/>	computing_tutorial.odp	5 minutes ago	1.15 MB
<input type="checkbox"/>	macs_peaks.narrowPeak	a day ago	56.6 kB
<input type="checkbox"/>	myPythonScript.py	a day ago	154 B
<input type="checkbox"/>	myRScript.R	a day ago	159 B
<input type="checkbox"/>	myShellScript.sh	a day ago	164 B
<input type="checkbox"/>	seacr_peaks.bed	a day ago	3.49 MB
<input type="checkbox"/>	textFile.txt	2 hours ago	192 B

# Jupyter Notebooks

Notebook file

Programming  
Language

Toolbars



The screenshot shows the Jupyter Notebook interface for a file named 'scRNAseq\_Notebook (autosaved)'. The top bar includes the Jupyter logo, the file name, and a 'Logout' button. Below this is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. A secondary toolbar contains icons for saving, creating new files, opening recent files, undo, redo, and running cells. The main content area is titled 'Single-Cell Analysis & Exploration Notebook'. It contains two code cells, each starting with 'In [ ]:' followed by 'data.obs' and 'data.var' respectively. Below these is a text cell containing the instruction: 'If cells and genes need to be swapped, uncomment and run the following cell to transpose the count matrix:'. On the left side of the notebook, there is a vertical DNA double helix graphic with blue and orange circles representing base pairs.

Coding  
Cells

Markdown (HTML)  
Cells

# Programming Final Statement



- Problems will almost always arise
- Things might not work how you expect
- Can always improve how things are done
- Google is your best friend
- Stack Overflow/Exchange will likely have a solution for every problem
- Be patient and try to use the right search terms

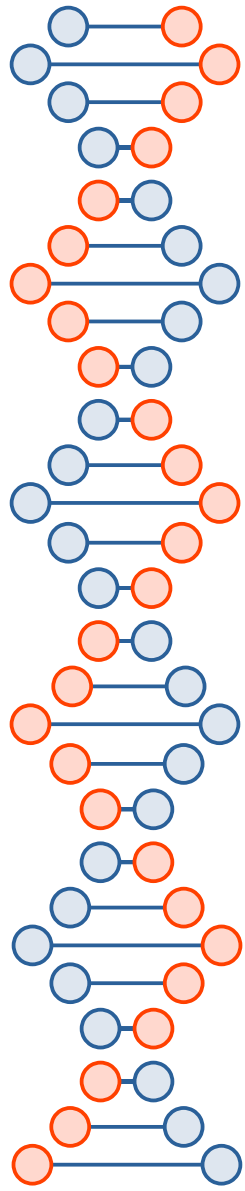




# GitHub

- Hosts code, programs, files, etc...for software developers
- Allows people to share their projects
- Uses Git (a version control system)
  - Enables tracking of software development
- Bioinformatics Examples:
  - <https://github.com/macs3-project/MACS>
  - <https://github.com/mikelove/DESeq2>
  - <https://github.com/Dilworth-Laboratory>





# Questions?

Contact:  
[earezza@ohri.ca](mailto:earezza@ohri.ca)