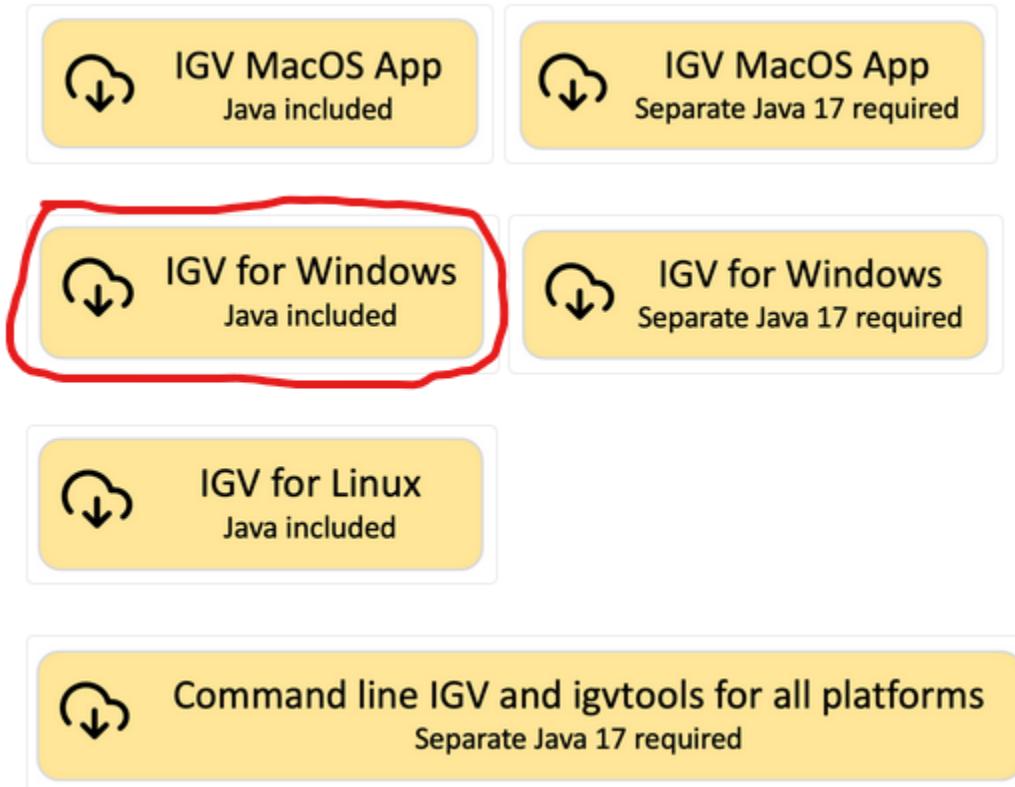


## Workshop Part 1 (IGV, Linux, R/R-studio) –

In the first part of the workshop I will be introducing the students to the Integrated Genome Viewer, known as IGV, very rudimentary Linux environment, and a very basic introduction to R/R-Studio and Bioconductor, mostly to prepare the student for using R and the Linux environment in the subsequent portion of the workshop where we will be looking at using tools such as aligners, bcftools, and samtools.

### 1. Integrated Genome Viewer

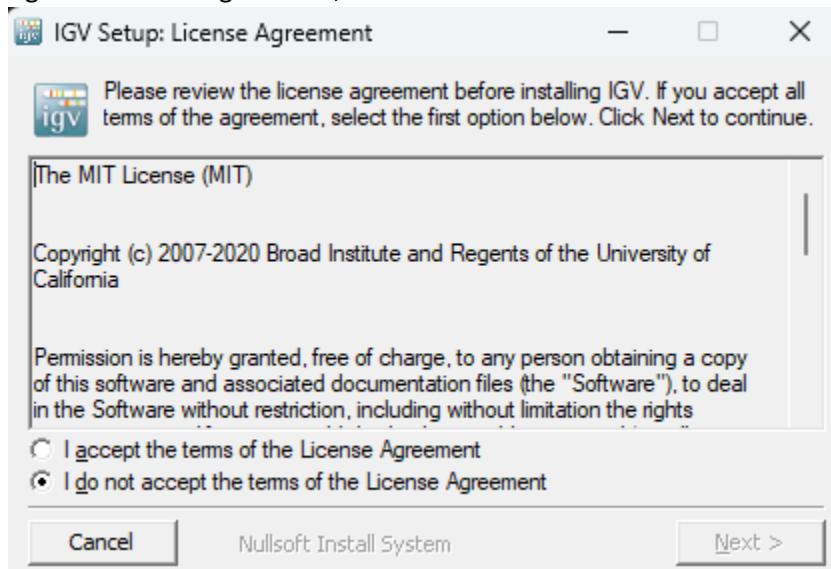
- a. Visit the following website: <https://igv.org/doc/desktop/#DownloadPage/>
- b. Scroll down and select the appropriate version:



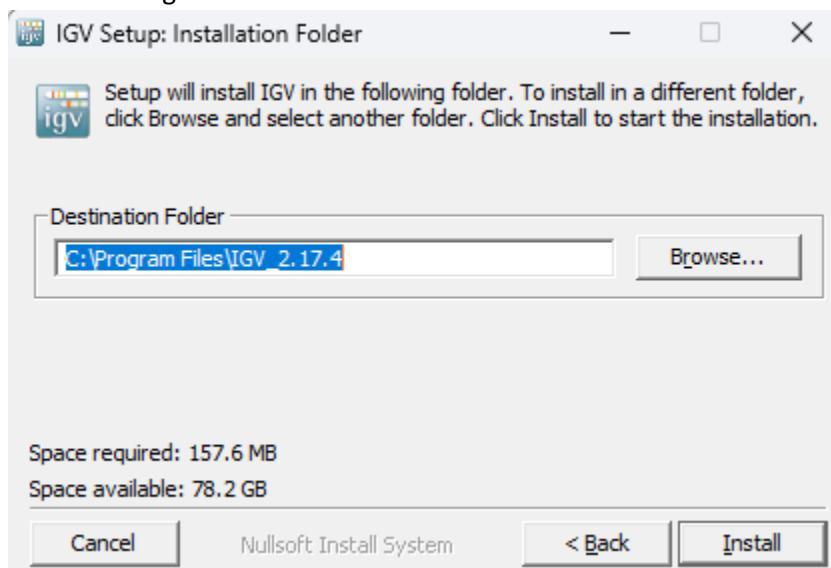
- c. Navigate to 'downloads' folder (or wherever you saved the .exe file) and right click then 'run as administrator' the .exe file.
- d. Press 'yes' to allow to make changes to the hard disk when prompted.

TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

- e. Agree to license agreement, and select 'next' :



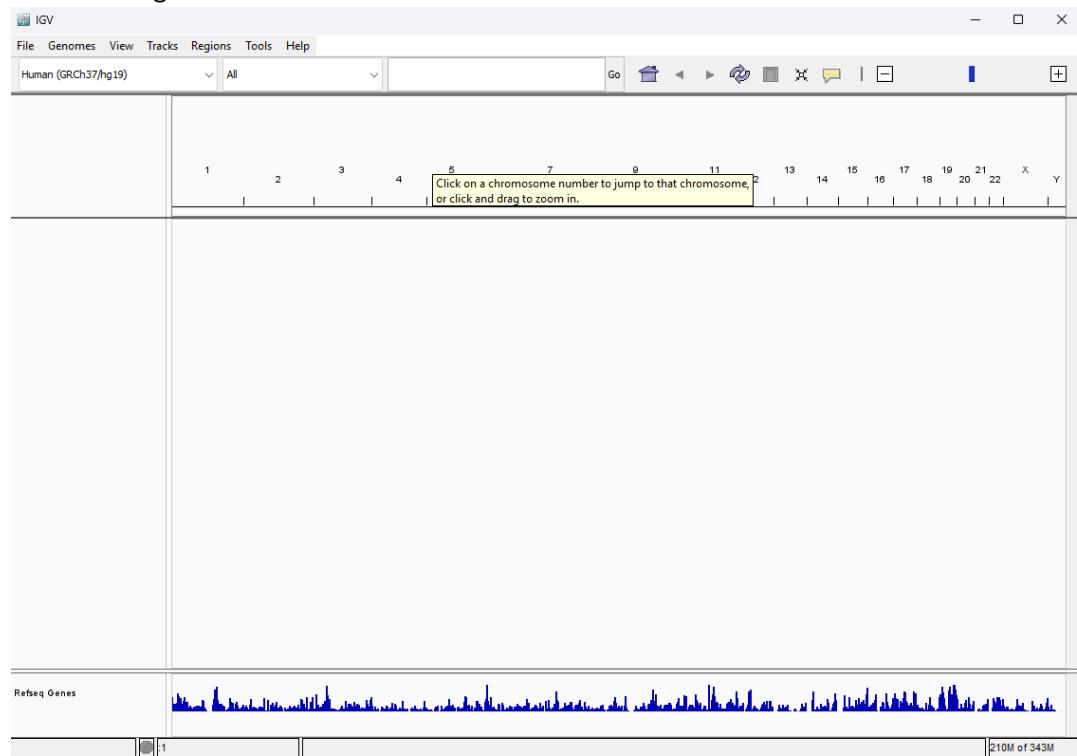
- f. Install in 'Program Files'



- g. Verify 'IGV' was successfully installed by pressing the windows key, followed by typing "IGV" into the search, then double clicking it. Upon launching you should see

TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

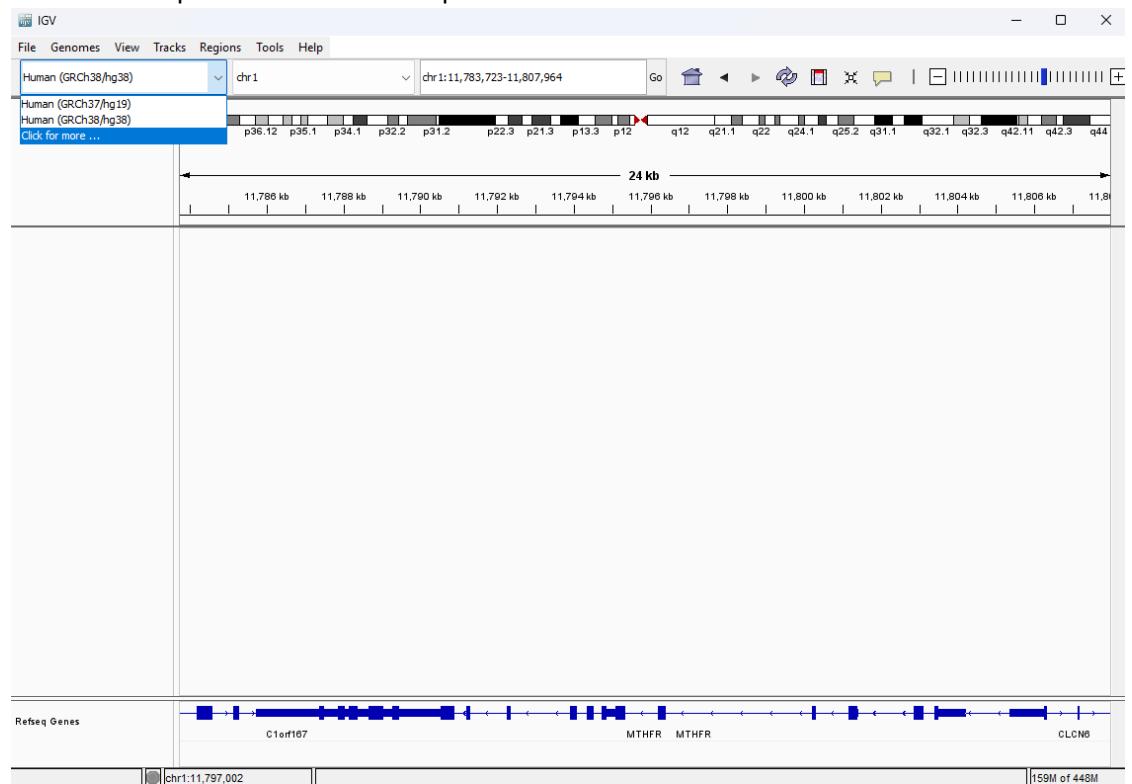
the following screen:



h. If you get a Java SDK warning just select “allow” and continue.

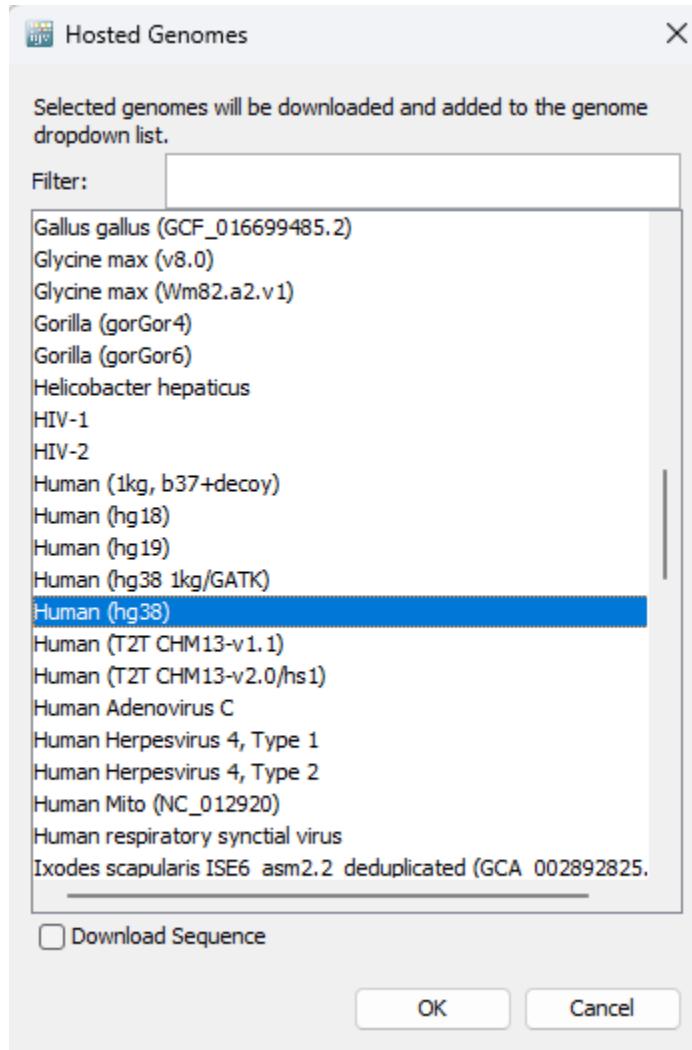
**2. Installing the Human Reference Genome GRCh38 in IGV**

a. Select the dropdown menu in the top left hand corner of the window:



TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

- b. Select “Click for More”, then scroll down in the new window and select “Human (hg38)”.



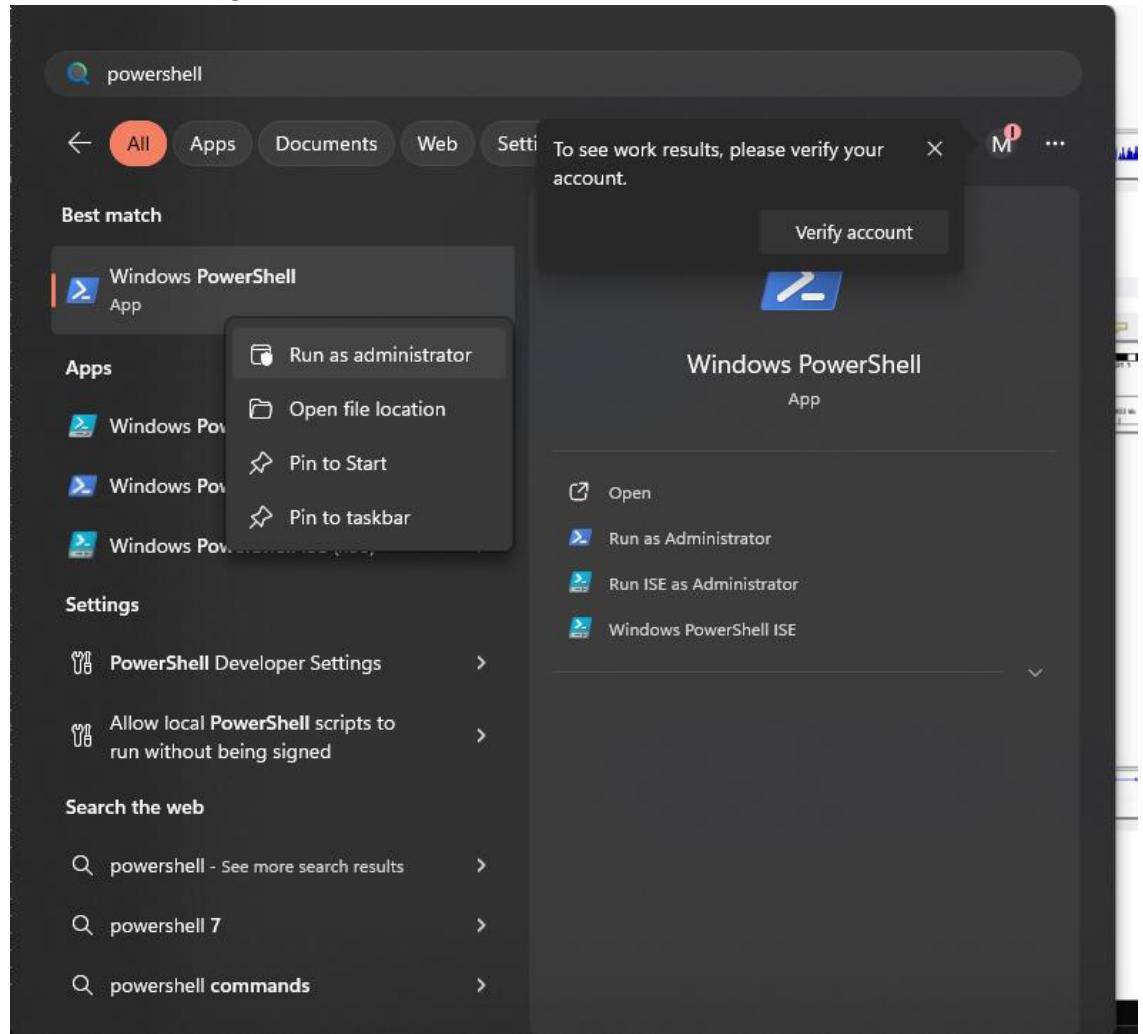
- c. Tick the “Download Sequence” check box, and press okay to begin the download.

**3. Installing the Windows Subsystem for Linux**

- a. For complete installation instructions and troubleshooting see:

<https://learn.microsoft.com/en-us/windows/wsl/install>

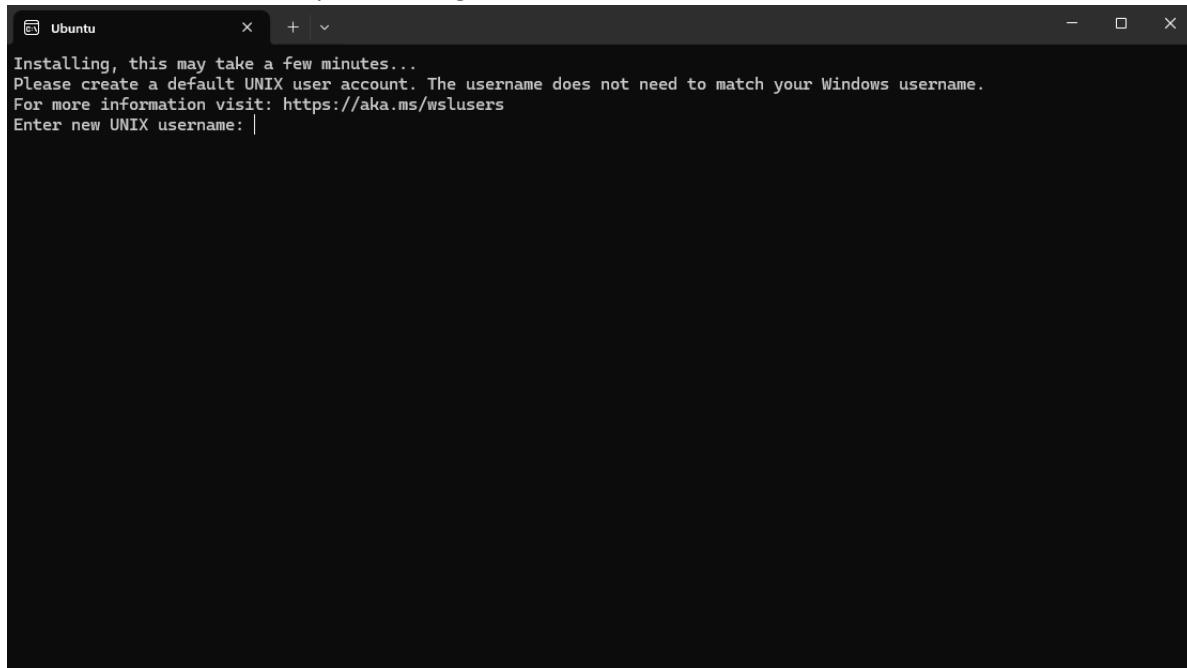
- b. Open PowerShell as an administrator by right-clicking PowerShell in the startup menu and selecting run as administrator.



- c. Select “Allow App to make changes” when prompted.  
d. In the powershell window type “wsl –install -d Ubuntu” and press enter.
- ```
PS C:\WINDOWS\system32> wsl --install -d Ubuntu
Installing: Ubuntu
Ubuntu has been installed.
Launching Ubuntu...
```
- e. You will see a status bar filling up (or it may remain stuck at zero for a while – do not panic, it is still installing), once finished the message: “Ubuntu has been installed.”

TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

Will be shown, followed by “Launching Ubuntu...” and a new window will be shown:



- f. When prompted to enter a new UNIX username, pick something that is easy to remember, such as a unique identifier for the computer, or simply the name “linuxuser” etc... after selecting a username, you will be prompted for a new password (this it for logging onto the windows subsystem for linux), I’d recommend making this something simple like “linux”, you will not see any characters being typed as you enter the password here, this is a security feature of linux, once you have typed the password once you will be prompted to type it again when you press enter:

```
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: micah
New password:
Retype new password: |
```

TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

- g. Enter the same password again and press enter, you will then see the following information displayed:

```
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: micah
New password:
Retype new password:
passwd: password updated successfully
Installation successful!
Windows Subsystem for Linux is now available in the Microsoft Store!
You can upgrade by running 'wsl.exe --update' or by visiting https://aka.ms/wslstorepage
Installing WSL from the Microsoft Store will give you the latest WSL updates, faster.
For more information please visit https://aka.ms/wslstoreinfo

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.10.16.3-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This message is shown once a day. To disable it please create the
/home/micah/.hushlogin file.
micah@MicahsPC:~$ |
```

- h. This indicates the linux subsystem for windows was successfully installed (Congratulations! You can now use Ubuntu 22.04 on your PC).  
i. Close the window, then reopen the powershell as a regular user, and type “wsl”, if the prompt changes similar to as shown here, then you were successful:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Micah> wsl
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

micah@MicahsPC:/mnt/c/Users/Micah$ |
```

#### 4. Installing R on the WSL.

- a. Installing R with WSL is straightforward. Just open the powershell, type wsl, then type: “sudo apt update” (you will then have to enter your password you just created) it might take a little while, this is determining which updates are available for the system, then type “sudo apt upgrade” followed by a “Y” when prompted, this will install all of the upgradeable packages.
- b. Next, add the CRAN repository to get the latest stable version of R, as follows:
  - i. Type “sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys E298A3A825C0D65DFD57CBB651716619E084DAB9” to add the proper repository key, then type “sudo add-apt-repository "deb https://cloud.r-project.org/bin/linux/ubuntu \$(lsb\_release -cs)-cran40/"” to add the repository (TIP: You can paste these directly into the terminal by using

TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

ctrl+shift+v)

```
micah@MicahsPC:/mnt/c/Users/Micah$ sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys E298A3A825C0D65DFD57CBB
651716619E084DAB9
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
Executing: /tmp/apt-key-gpghome.BKsp5NLwIj/gpg.1.sh --keyserver keyserver.ubuntu.com --recv-keys E298A3A825C0D65DFD57CBB
651716619E084DAB9
gpg: key 51716619E084DAB9: public key "Michael Rutter <marutter@mail.com>" imported
gpg: Total number processed: 1
gpg:           imported: 1
micah@MicahsPC:/mnt/c/Users/Micah$ sudo add-apt-repository "deb https://cloud.r-project.org/bin/linux/ubuntu $(lsb_release -cs)-cran40/"
Repository: 'deb https://cloud.r-project.org/bin/linux/ubuntu jammy-cran40/'
Description:
Archive for codename: jammy-cran40/ components:
More info: https://cloud.r-project.org/bin/linux/ubuntu
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
Adding deb entry to /etc/apt/sources.list.d/archive_uri-https_cloud_r-project_org_bin_linux_ubuntu-jammy.list
Adding disabled deb-src entry to /etc/apt/sources.list.d/archive_uri-https_cloud_r-project_org_bin_linux_ubuntu-jammy.li
st
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:5 https://cloud.r-project.org/bin/linux/ubuntu jammy-cran40/ InRelease [3626 B]
Get:6 https://cloud.r-project.org/bin/linux/ubuntu jammy-cran40/ Packages [46.2 kB]
Fetched 49.8 kB in 1s (55.4 kB/s)
Reading package lists... Done
W: https://cloud.r-project.org/bin/linux/ubuntu/jammy-cran40/InRelease: Key is stored in legacy trusted.gpg keyring (/et
c/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
micah@MicahsPC:/mnt/c/Users/Micah$ |
```

ii. Update the list of repositories by typing: “sudo apt update” again.

- c. Install R by typing “sudo apt install r-base”, then press “Y” when prompted. R and all of its system dependencies will be installed. This can take a little time.
- d. When finished, type “R”, and you should see the following:

```
Processing triggers for libgdk-pixbuf-2.0-0:amd64 (2.42.8+dfsg-1ubuntu0.3) ...
Processing triggers for libc-bin (2.35-0ubuntu3.8) ...
micah@MicahsPC:/mnt/c/Users/Micah$ R

R version 4.4.0 (2024-04-24) -- "Puppy Cup"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

- e. R has been successfully installed!

## 5. Installing R-Studio for the WSL

- a. exit R by typing “q(“no”)”
- b. Install gdebi-core package: “sudo apt install gdebi-core”, type “Y” when prompted.
- c. Download the r-studio package with wget, type : “wget <https://download1.rstudio.org/electron/jammy/amd64/rstudio-2024.04.1-748-.deb>

TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

amd64.deb”

```
micah@MicahsPC:/mnt/c/Users/Micah$ wget https://download1.rstudio.org/electron/jammy/amd64/rstudio-2024.04.1-748-amd64.deb
--2024-06-10 15:10:25-- https://download1.rstudio.org/electron/jammy/amd64/rstudio-2024.04.1-748-amd64.deb
Resolving download1.rstudio.org (download1.rstudio.org)... 99.86.74.82, 99.86.74.21, 99.86.74.37, ...
Connecting to download1.rstudio.org (download1.rstudio.org)|99.86.74.82|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 197000720 (188M) [application/octet-stream]
Saving to: ‘rstudio-2024.04.1-748-amd64.deb’

rstudio-2024.04.1-748-amd64.d 100%[=====] 187.87M 18.7MB/s in 8.8s

utime(rstudio-2024.04.1-748-amd64.deb): Operation not permitted
2024-06-10 15:10:34 (21.4 MB/s) - ‘rstudio-2024.04.1-748-amd64.deb’ saved [197000720/197000720]

micah@MicahsPC:/mnt/c/Users/Micah$
```

- d. Install the package with gdebi: “sudo gdebi rstudio-2024.04.1-748-amd64.deb”:

```
micah@MicahsPC:/mnt/c/Users/Micah$ sudo gdebi rstudio-2024.04.1-748-amd64.deb
[sudo] password for micah:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Reading state information... Done
Requires the installation of the following packages: lib32gcc-s1 lib32stdc++6 libc6-i386 libclang-14-dev libclang-common-14-dev libclang-dev libclang1-14 libgcc1 liblvm14 libobjc-11-dev libobjc4 libssl-dev libxcb-xkb1 libxkbcommon-x11-0

RStudio
RStudio is an integrated development environment (IDE) designed to support multiple languages, including both R and Python. It includes a console, syntax-highlighting editor that supports direct code execution, and a variety of robust tools for plotting, viewing history, debugging and managing your workspace.
Do you want to install the software package? [y/N]:y
/usr/bin/gdebi:113: FutureWarning: Possible nested set at position 1
    c = findall("[[([(\S+)/\S+[])]]", msg)[0].lower()
Get:1 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libc6-i386 amd64 2.35-0ubuntu3.8 [2838 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 lib32gcc-s1 amd64 12.3.0-1ubuntu1~22.04 [63.9 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy/main amd64 libgcc1 amd64 1:8.0.6-1.1build1 [96.8 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 libobjc4 amd64 12.3.0-1ubuntu1~22.04 [48.6 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 libobjc-11-dev amd64 11.4.0-1ubuntu1~22.04 [196 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 liblvm14 amd64 1:14.0.0-1ubuntu1.1 [24.0 MB]
5% [6 liblvm14 487 kB/24.0 MB 2%] 532 kB/s 2min 1s
```

- e. Type linux password when prompted, followed by a y when prompted. This will

install Rstudio, and all of its dependencies.

- f. Type “rstudio”, if you get an error:

- rstudio: error while loading shared libraries: libnss3.so: cannot open shared object file: No such file or directory – Then type “sudo apt install libnss3-dev libgdk-pixbuf2.0-dev libgtk-3-dev libxss-dev”**
- rstudio: error while loading shared libraries: libasound.so.2: cannot open shared object file: No such file or directory – Then type “sudo apt install libasound2”**

- g. keep trying to type ‘rstudio’ and eventually you will see the following flash on the

screen, do not worry, even though these say “error” R-studio should still launch.

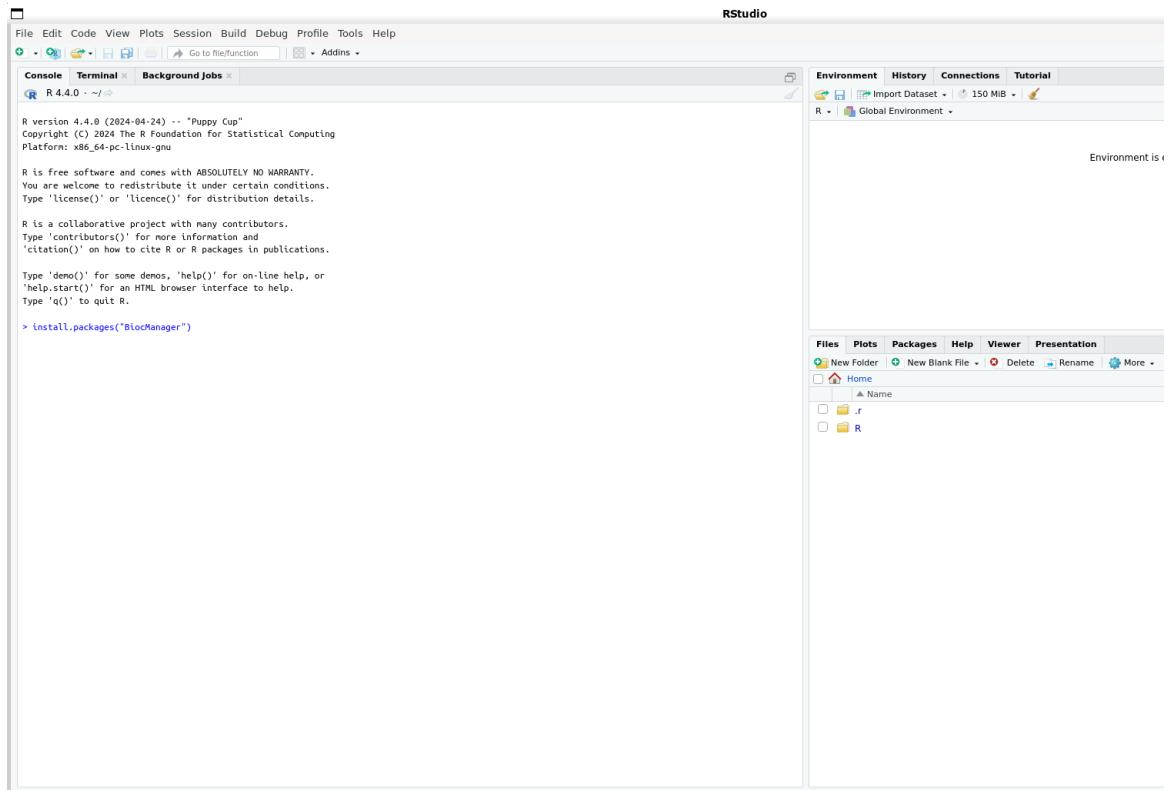
```
micah@MicahsPC:/mnt/c/Users/Micah$ rstudio
[9163:0610/152110.633916:ERROR:bus.cc(407)] Failed to connect to the bus: Failed to connect to socket /run/dbus/system_bus_socket: No such file or directory
[9163:0610/152115.815979:ERROR:bus.cc(407)] Failed to connect to the bus: Failed to connect to socket /run/dbus/system_bus_socket: No such file or directory
[9163:0610/152115.816150:ERROR:bus.cc(407)] Failed to connect to the bus: Failed to connect to socket /run/dbus/system_bus_socket: No such file or directory
[9163:0610/152115.818982:ERROR:bus.cc(407)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
[9163:0610/152115.819052:ERROR:bus.cc(407)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
[9163:0610/152115.819160:ERROR:bus.cc(407)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
[9195:0610/152115.831309:ERROR:gl_factory.cc(120)] Requested GL implementation (gl=none,angle=none) not found in allowed implementations: [(gl=egl-angle,angle=default),(gl=egl-gles2,angle=none)].
[9195:0610/152115.833581:ERROR:viz_main_impl.cc(196)] Exiting GPU process due to errors during initialization
```

TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

- h. When you see the R-Studio screen, you know you have successfully installed R-Studio.

**6. Installing Bioconductor, and other R-Packages:**

- We will use R-Studio to install some essential bioinformatics packages:
- In the main window of R-Studio, type: "install.packages("BiocManager")":



- You should see something similar to the following:

```
> install.packages("BiocManager")
Installing package into ‘/home/micah/R/x86_64-pc-linux-gnu-library/4.4’
(as ‘lib’ is unspecified)
trying URL 'https://cloud.r-project.org/src/contrib/BiocManager_1.30.23.tar.gz'
Content type 'application/x-gzip' length 589753 bytes (575 KB)
=====
downloaded 575 KB

* installing *source* package ‘BiocManager’ ...
** package ‘BiocManager’ successfully unpacked and MD5 sums checked
** using staged installation
** R
** inst
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
** testing if installed package can be loaded from temporary location
** testing if installed package can be loaded from final location
** testing if installed package keeps a record of temporary installation path
* DONE (BiocManager)

The downloaded source packages are in
‘/tmn/RtmnXKOvuM/downloaded packages’
```

## 7. Installing DESeq2 and EdgeR

- a. To install DESeq2 and its dependencies in R, run the command:  
“BiocManager::install(“DESeq2”)”.
- You will see a lot of output as there are many dependencies for DESeq2, you may encounter an error here:

- i. Several packages were not installed correctly because a dependency is missing in linux:

```
ERROR: dependency 'GenomeInfoDb' is not available for package 'GenomicRanges'  
* removing '/home/micah/R/x86_64-pc-linux-gnu-library/4.4/GenomicRanges'  
ERROR: dependencies 'GenomicRanges', 'GenomeInfoDb' are not available for package 'SummarizedExperiment'  
* removing '/home/micah/R/x86_64-pc-linux-gnu-library/4.4/SummarizedExperiment'  
ERROR: dependencies 'GenomicRanges', 'SummarizedExperiment' are not available for package 'DESeq2'  
* removing '/home/micah/R/x86_64-pc-linux-gnu-library/4.4/DESeq2'
```

```
The downloaded source packages are in  
      '/tmp/RtmpXK0vuM/downloaded_packages'  
Installation paths not writeable, unable to update packages  
  path: /usr/lib/R/library  
  packages:  
    codetools, lattice, MASS, Matrix, spatial  
Warning messages:  
1: In install.packages(...):  
  installation of package 'curl' had non-zero exit status  
2: In install.packages(...):  
  installation of package 'httr' had non-zero exit status  
3: In install.packages(...):  
  installation of package 'UCSC.utils' had non-zero exit status  
4: In install.packages(...):  
  installation of package 'GenomeInfoDb' had non-zero exit status  
5: In install.packages(...):  
  installation of package 'GenomicRanges' had non-zero exit status  
6: In install.packages(...):  
  installation of package 'SummarizedExperiment' had non-zero exit status  
7: In install.packages(...):  
  installation of package 'DESeq2' had non-zero exit status
```

If this happens, close R-Studio and return to the command prompt and enter: “sudo apt install build-essential libcurl4-gnutls-dev” enter password then type “Y” when prompted, and reopen rstudio by typing “rstudio”, and retype the command “BiocManager::install(“DESeq2”)” (it will be much shorter this time). Upon successful completion you should see:

```
installing to /home/micah/R/x86_64-pc-linux-gnu-library/4.4/00LOCK-DESeq2/00new/DESeq2/libs  
** R  
** inst  
** byte-compile and prepare package for lazy loading  
** help  
*** installing help indices  
** building package indices  
** installing vignettes  
** testing if installed package can be loaded from temporary location  
** checking absolute paths in shared objects and dynamic libraries  
** testing if installed package can be loaded from final location  
** testing if installed package keeps a record of temporary installation path  
* DONE (DESeq2)
```

```
The downloaded source packages are in  
      '/tmp/RtmpUffFyn6/downloaded_packages'  
Installation paths not writeable, unable to update packages  
  path: /usr/lib/R/library  
  packages:  
    codetools, lattice, MASS, Matrix, spatial  
> |
```

TWU Project ACCESS Bioinformatics Workshop (July 2024) Software Installation Instructions [Windows]  
Micah Andrew Thornton

- b. to install “EdgeR” package, type “BiocManager::install(“edgeR”)” in the Rstudio Window \*(this will be a much quicker install). When finished you should see:

```
installing to /home/micah/R/x86_64-pc-linux-gnu-library/4.4/00LOCK-edgeR/00new/edgeR/libs
** R
** inst
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
** testing if installed package can be loaded from temporary location
** checking absolute paths in shared objects and dynamic libraries
** testing if installed package can be loaded from final location
** testing if installed package keeps a record of temporary installation path
* DONE (edgeR)

The downloaded source packages are in
  '/tmp/RtmpUfFyn6/downloaded_packages'
Installation paths not writeable, unable to update packages
  path: /usr/lib/R/library
  packages:
    codetools, lattice, MASS, Matrix, spatial
```

- c. To test that both packages were successfully installed, type “library(“DESeq2”)” and library(“edgeR”) into the R-Studio window, for DESeq2 you will see some output that starts with:

```
> library("DESeq2")
Loading required package: S4Vectors
Loading required package: stats4
Loading required package: BiocGenerics

Attaching package: 'BiocGenerics'

The following objects are masked from 'package:stats':
  IQR, mad, sd, var, xtabs

The following objects are masked from 'package:base':
  anyDuplicated, aperm, append, as.data.frame, basename, cbind, colnames, dirname, do.call, duplicated, eval, evalq, Filter,
  Find, get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget, order, paste, pmax, pmax.int, pmin,
  pmin.int, Position, rank, rbind, Reduce, rownames, sapply, setdiff, table, tapply, union, unique, unsplit, which.max,
  which.min

Attaching package: 'S4Vectors'
```

- d. For “edgeR” you will see:

```
> library("edgeR")
Loading required package: limma

Attaching package: 'limma'

The following object is masked from 'package:DESeq2':
  plotMA

The following object is masked from 'package:BiocGenerics':
  plotMA

> |
```

8. Installing HISAT2, samtools, bcftools, Salmon, and Kallisto.

- a. Close R-Studio, we are done setting it up now,
  - b. In the wsl command prompt, type: “sudo apt install hisat2”, type password if necessary, and “Y” when prompted.
  - c. When finished, type “which hisat2”, you should see:  

```
micah@MicahsPC:/mnt/c/Users/Micah$ which hisat2
/usr/bin/hisat2
```
  - d. Type “which samtools” then “which bcftools”, you should see:  

```
micah@MicahsPC:/mnt/c/Users/Micah$ which samtools
/usr/bin/samtools
micah@MicahsPC:/mnt/c/Users/Micah$ which bcftools
/usr/bin/bcftools
```
  - e. Type “sudo apt install salmon” to install salmon, then when completed type “which salmon” you should see:  

```
micah@MicahsPC:/mnt/c/Users/Micah$ which salmon
/usr/bin/salmon
```
  - f. Type “sudo apt install kallisto” to install kallisto, then when completed type “whch kallisto” you should see:  

```
micah@MicahsPC:/mnt/c/Users/Micah$ which kallisto
/usr/bin/kallisto
```
9. We now have all of the tools that we need to do some sophisticated bioinformatics! Congratulations if you made it this far, if there were some errors or other problems during the installation, feel free to reach out to me, Micah Thornton, at ([mathornton@smu.edu](mailto:mathornton@smu.edu)).

**THANKS FOR YOUR TIME AND EFFORTS!!!**

