

# Module 1.2: Learn - Coding

## Overview

This section includes:

1. Getting connected
2. Accessing the RStudio server
3. Navigating directories
4. Copying files to your home directory
5. Analysis

## Getting Connected

### What is Agave?

Agave is a High-Performance Computing (HPC) cluster, FREE for faculty, staff, and students. This cluster architecture uses hundreds of compute servers, also called nodes, and their collective cores to help users optimize their research. This gives researchers access to high memory computation and storage while freeing the researcher's local machine. All students must be supported by ASU faculty in order to use the research computing cluster.

## Accessing the RStudio server (login.rc.asu.edu)

Step-by-step guide on accessing RStudio on the Agave Cluster

**Step 1.** Downloading a Virtual Private Network (VPN) in order to remotely access the ASU network.

Using a VPN to remotely access the ASU network is essential for using the Agave computing cluster. The CISCO ConnectAnywhere app is free to download at MyASU -> My Apps -> Keyword "VPN" -> Download the Cisco ConnectAnywhere for your operating system (PC or Mac)

**ASU**  
Arizona State University

**My ASU**


Home Finances Campus Services Profile Help

CS PeopleSoft Gmail Canvas Google Drive ASU Library HR PeopleSoft **My Apps** Calendar

Home **My Apps**

**Keyword**  **Platform**  **Categories**  **Access**  **Search**

View by title: **A B C D E F G H I J K L M N O P**

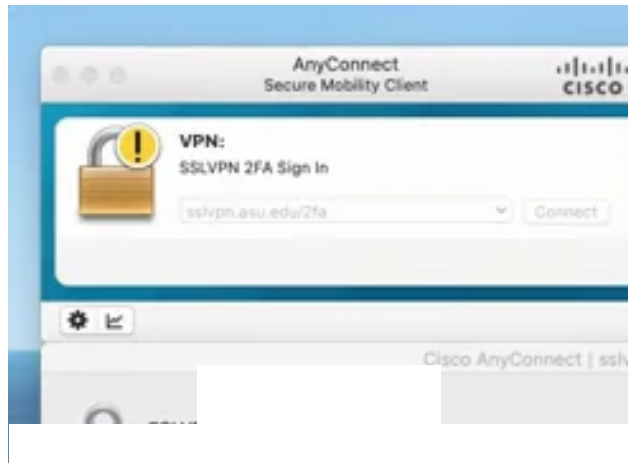
  
Cisco SSL VPN

**Click and follow the specified protocol for your machine.**

**Figure. How to download the Cisco SSL VPN through myasu.edu**

In the 'My Apps' tab of my.asu.edu, you can enter the keyword "vpn" then click on the Cisco SSL VPN icon.

Now we're ready to connect. This video will show you how to connect to the ASU cluster via the VPN. In the first ASUrite ID, followed by your password, and then the authentication factor (type "push" to receive a push notification on your mobile device or enter a passcode)




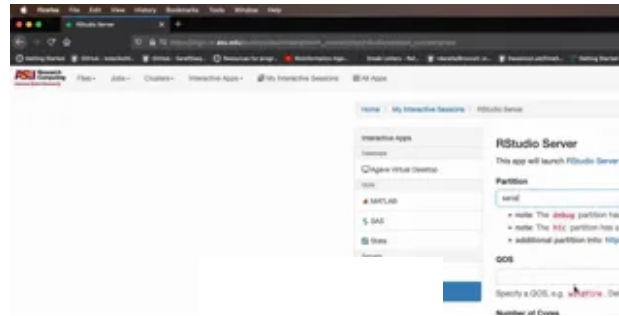
### **Video. Logging into the Agave HPC.**

This video will demonstrate how to login to the VPN to access the Agave HPC using DUO Mobile.

**[View transcript. \(https://canvas.asu.edu/courses/122165/files/54792273?wrap=1\)](https://canvas.asu.edu/courses/122165/files/54792273?wrap=1)**   
**[https://canvas.asu.edu/courses/122165/files/54792273/download?download\\_frd=1](https://canvas.asu.edu/courses/122165/files/54792273/download?download_frd=1)**

**Step 2.** Accessing RStudio via the ASU Agave computing cluster.

In this course will use the **R**  (<https://www.r-project.org/>) programming language and its user-friendly interface, **RStudio** (<https://www.rstudio.com/products/rstudio/download/>); think of R as the car engine and RStudio as the dashboard design elements to make it easier to interact with the engine. Both can both be **downloaded locally to your computer** (<https://www.rstudio.com/products/rstudio/download/>); but to avoid issues with version updates and to have easy access generated for this course, we will be using an instance of RStudio from a server on the Agave computing cluster. RStudio will look identical to a local downloaded version of RStudio (which is free) so the skills you gain working on it long after this course is complete. Here are the steps for starting an instance of RStudio from the Research Com



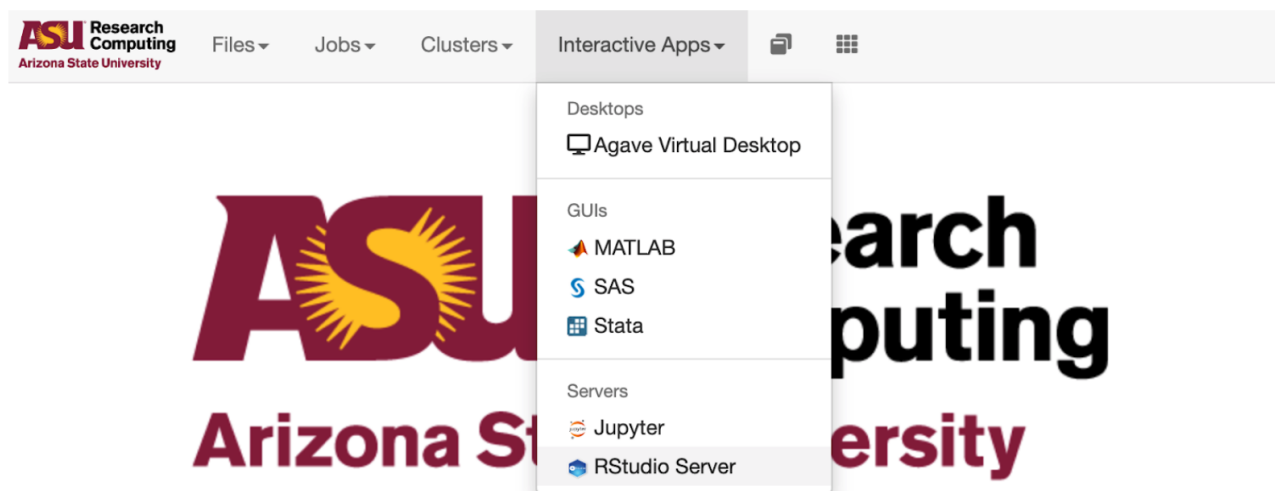
### Video. Logging into RStudio Server from a web browser.

Partition: htc, QOS: normal, Number of Cores: 1, Time: 0-4, R version: 4.1.2-BLAS

**View Transcript** (<https://canvas.asu.edu/courses/122165/files/54792266?wrap=1>)   
([https://canvas.asu.edu/courses/122165/files/54792266/download?download\\_frd=1](https://canvas.asu.edu/courses/122165/files/54792266/download?download_frd=1))

1. Login to the VPN (virtual private network) using Cisco VPN with your ASUrite credentials. You may need Duo login.

2. Log into ASU network using VPN; you MUST be logged into the VPN in order to proceed.
3. Navigate to <https://login.rc.asu.edu/> (<https://login.rc.asu.edu/>) (if you are not logged into the VPN, the webpage will prompt you to log in). The ASU Research Computing (RC) page is an excellent resource for many of your computing needs. On this page, you can find information about RC workshops, attend office hours, and access various tools. We will be using the 'Interactive Apps' tab throughout the remainder of this course, so it is recommended to bookmark or favorite this link for easy access.
4. Click the 'Interactive Apps' menu and select 'RStudio Server'



Open OnDemand provides an integrated, single access point for all of your HPC resources

Use the navigation bar at the top to get started.

[Click here for cluster status.](#)

**Figure. Accessing the RStudio Server App from the ASU Research Computing site <https://login.rc.asu.edu/>**

Click the 'Interactive Apps' tab on the RC site, then select 'RStudio Server'.

## 5. Specify your settings and launch the session.

For this course, we will be using the `htc` partition that will give our jobs priority but has a time limit of 4 hours. of the default settings and will be using R version 4.1.2-BLAS. If you have any trouble logging in, reach out to Slack channel `#troubleshooting`.



### Interactive Apps

#### Desktops

 Agave Virtual Desktop

#### GUIs

 MATLAB

 SAS

 Stata

#### Servers

 Jupyter

 RStudio Server

## RStudio Server

This app will launch [RStudio Server](#) an IDE for [R](#) on the [Agave cluster](#).

### Partition

htc

- **note:** The `debug` partition has a max walltime of 15 minutes
- **note:** The `htc` partition has a max walltime of 4 hours
- **additional partition info:** <https://asurc.atlassian.net/l/c/DBS6VXmR>

### QOS

normal

Specify a QOS, e.g. `wildfire`. Default is `normal`.

### Number of Cores

1

Number of cores for job.

### Time

0-4

- Set an upper limit on the run time of `rstudio`.
- **time formats:** "minutes", "minutes:seconds", "hours:minutes:seconds", "days-hours", "days-hours:minutes", "days-hours:minutes:seconds"

### GRES

Request GRES here if needed, e.g. `gpu:1`

☐ I would like to receive an email when the session starts

### R version



**R version**

4.1.2-BLAS

This defines the version of R you want to load.

**Additional sbatch options**

**Advanced Feature.** Leave this blank unless you want to specify additional sbatch options.

Launch

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

**Figure. Settings to launch RStudio Server.**

The settings for this class are Partition: htc, QOS: normal,  
Number of Cores: 1, Time: 0-4, R version: 4.1.2-BLAS

6. Load the session and click 'Connect to RStudio Server' (it may take a few minutes for the button to show up).

Session was successfully created.

Home / My Interactive Sessions

Interactive Apps

Desktops

Agave Virtual Desktop

GUIs

MATLAB

SAS

Stata

Servers

Jupyter

RStudio Server

**RStudio Server (16885731)** 1 node

Host: >\_cg18-4.agave.rc.asu.edu

Created at: 2022-07-24 12:05:52 MST

Time Remaining: 3 hours and 59 minutes

Session ID: ba24b69b-69f8-4541-8c29-2ef507bd04ee

® Connect to RStudio Server

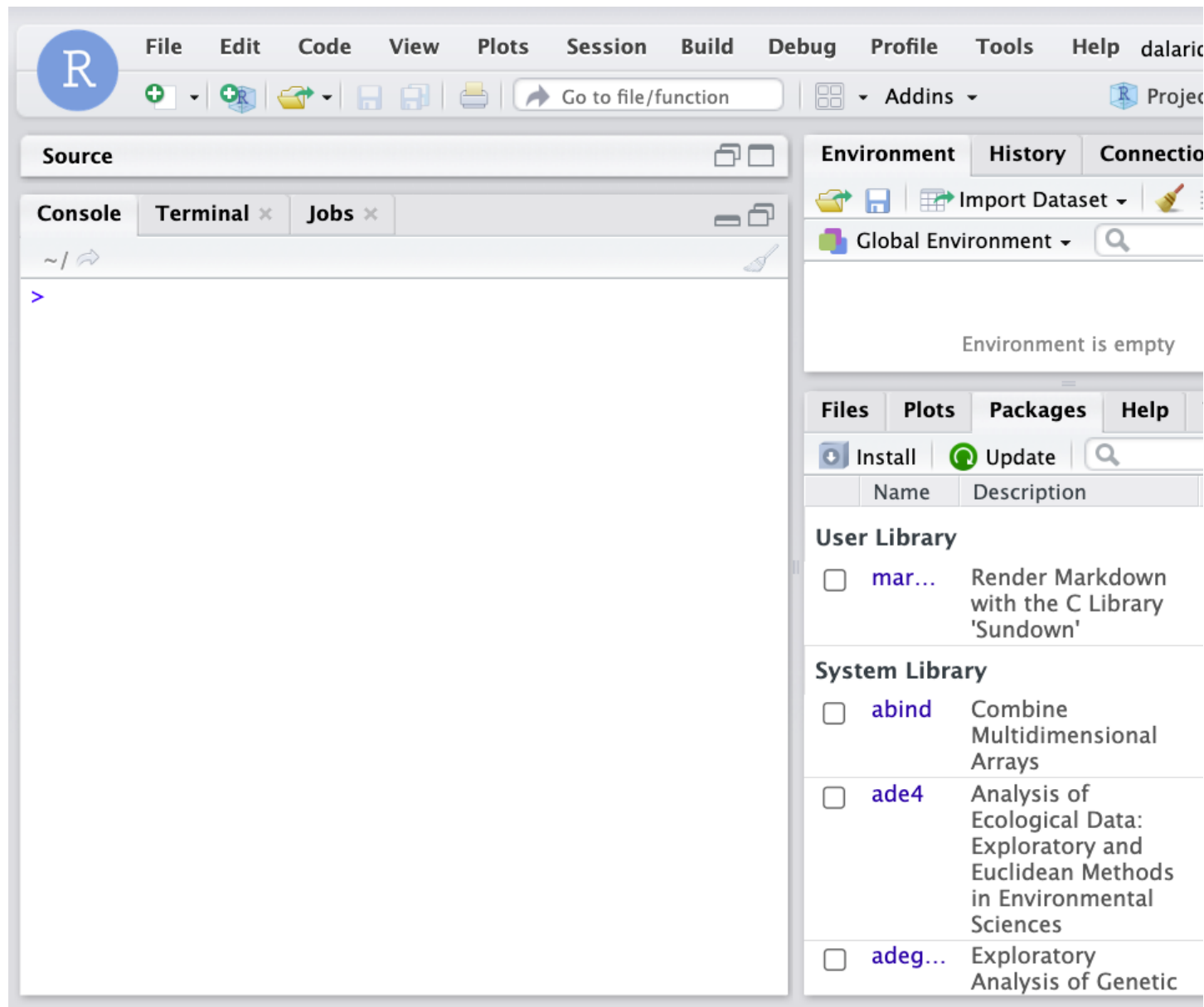
### Figure. Launching the session.

After RStudio Server has loaded, click the 'Connect to RStudio Server' button to launch.

7. You're ready to start using RStudio Server!

### Step 3. Familiarizing yourself with RStudio

Once RStudio has been accessed on the cluster you should be looking at a screen that looks like this:

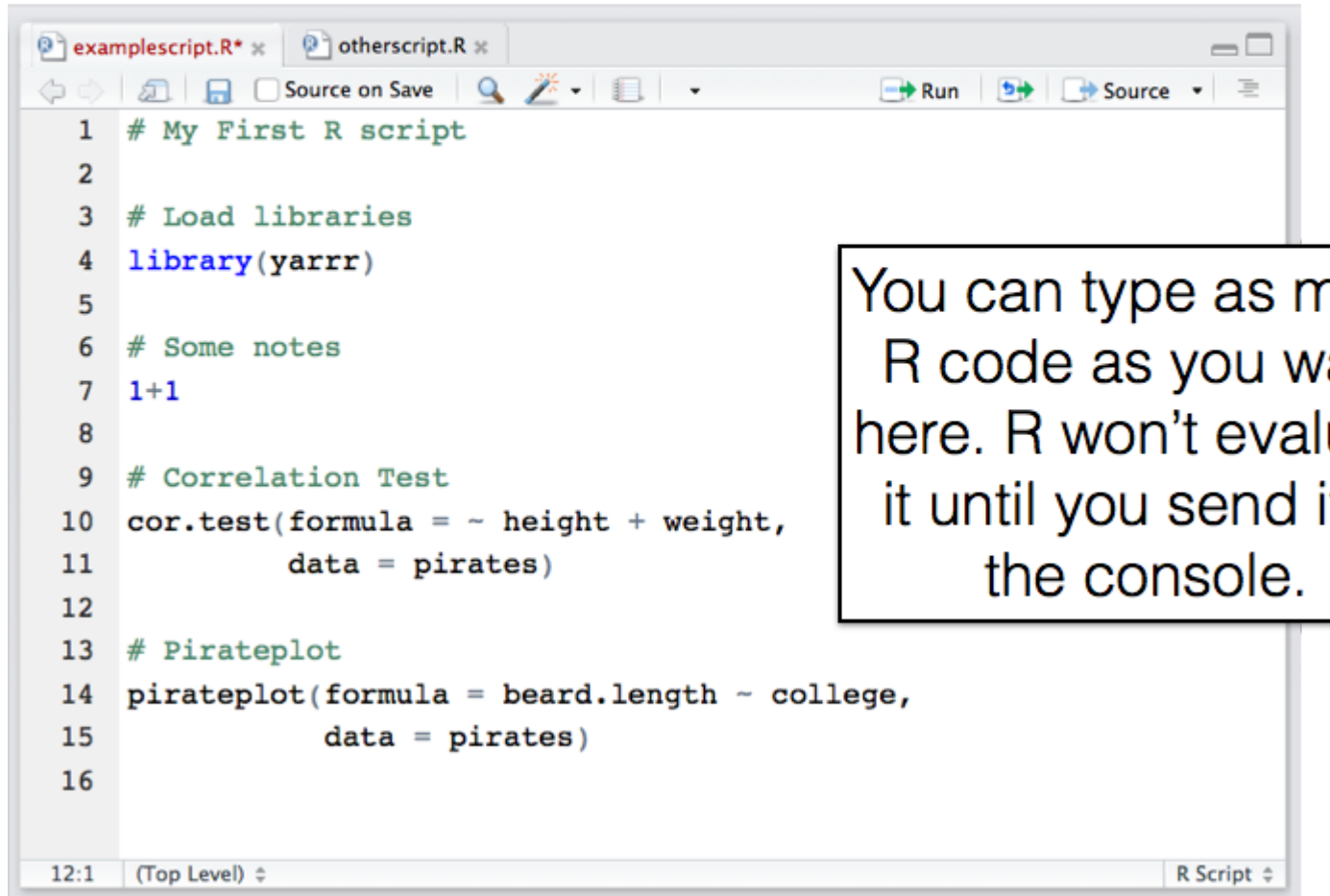


**Figure. RStudio Server on the ASU High-Performance Cluster.**

RStudio server has four panes that each perform different functions. The tabs featured in the panes above are (clockwise from the top left): Source, Environment, Packages, Console.

RStudio is organized by different windows, or panes. Each of which has different functions and interactivity. The pane you will be interacting with is the top left-hand one. Known as the source or scripting window. This is where code can be copy & pasted, and ran.

This is the **source**



**Figure. RStudio Source pane.**

The source pane is where code can be keyed before sending to the console for evaluation.

In case you want to copy and paste this, you can write your first R script using code like this:

```
# MyFirstRScript.R  
  
# This will make a vector of three values  
x <-c(0,1,2)  
  
# This will make a second vector of three values  
y <-c(5,0,4)  
  
# This will add the two vectors together and save it in a variable called "addition"  
addition <- x+y  
  
# This will let us view the new vector, addition  
addition
```

## Full data set for this research project

The full dataset generated for this course is in a directory which can only be accessed by the instructors, but a portion will be made available to you for the analysis we have planned (this is to make it easier for you to work with). In this section, we will show you what is available in the full data set so you know what we have generated before your arrival. We have shared this data with you to use the differential expression pipeline, but if our research leads us to methods that need any of the other data, we can make those files available to you too.

The trimming data set was generated with a workflow manager called Snakemake implemented with Snakemake. While Snakemake is required to run this yourself, but if you are curious about it, the workflow files are stored here in [the Wilson lab GitHub repository](https://github.com/SexChrLab/Genomics_CURE/tree/main/data_generation) ([https://github.com/SexChrLab/Genomics\\_CURE/tree/main/data\\_generation](https://github.com/SexChrLab/Genomics_CURE/tree/main/data_generation)).

The files ending in .snakefile are loaded into the Snakemake program to enter commands on the cluster and execute the workflow. Each rule in these workflow files generates an output in a specific subdirectory. Our snakemake files use a configuration file (config.json) which contain sample names and sex that we can use to specify the input to analysis programs. The Python script (merge.py) was made to put merge results from individual samples into one big table so it is easier to work with. The table below shows each snakefile and their outputs:

**Table. Description of each snakefile and their outputs**

File	Description	Output
generate_trimming_data.snakefile	sequence files from trimming software varying the trimq and minlen parameters	fastq/
fastqc.snakefile	FastQC results	qc/fastqc
align.snakefile	alignment to sex complement reference genomes and percent mapped	processed stats/
feature_counts.snakefile	counts of the number of reads aligning to each gene for each sample	featureCo
merge_counts_v2.py	all counts merged into one table; this is the main data file we will be using for our analysis	geneCount

These executable files will produce 10 directories that contain the sequences, QC reports, alignments, and feature quantification for gene expression: 9 trimmed data sets using different combinations of trimq and minlen pass the raw, untrimmed data.

Table. Structure and contents of full trimming data set

Subdirectory	Contents
fastq/	Sequences of reads from each sample (.fastq.gz)

qc/fastqc/	FastQC reports for visualizing sequencing data quality (_fastqc.html)
processed_bams/	Output from the alignment algorithm HISAT2 (.bam)
stats/	Metrics that show the success of alignment (percent reads mapped, etc) (.txt)
featureCounts/	Total reads aligned to each gene in each sample (geneCounts_XX/XY.txt, transcriptCo

We have copied the gene counts matrix for each trimmed data set and the untrimmed data set from the featureC directory that you have read access to:

```
/data/mwilsons/GenomicsResearchExperience/Placenta_SexDiff/geneCounts/
```

You will use this path to specify where the specific gene counts matrix that you will be analyzing will be stored when running your differential gene expression pipeline in R/RStudio.

## Template code for this research project

In addition to generating data, we have also developed some R code to help you get started with differential gene expression analysis. The template code is written in R Markdown language which you will be learning about later in the course. For this project, we will focus on getting the template code copied into your own home directory on Agave. You will use and edit your own RStudio and make printed reports of your own code when needed.

Let's get the files you will be using copied over to your home directory! We are including below instructions for using the tool from the same Research Computing web portal that we used to fire up an instance of RStudio on the cluster, if you have previous experience and would prefer to use other shell terminal applications such as Putty, Cygwin, or Mac OS X.

see more information about that on [Research Computing's documentation for Agave](https://asurc.atlassian.net/wiki/spaces/RC/pages/45318147/Connecting+with+SSH)   
(<https://asurc.atlassian.net/wiki/spaces/RC/pages/45318147/Connecting+with+SSH>)\_.

The shell access tool from Research Computing below does exactly the same thing; it's more beginner friendly so you don't have to download anything or enter as much information.

## Agave Interactive App for Shell Access

The package of template code used for this class has been zipped and stored at this location:

```
/data/mwilsons/GenomicsResearchExperience/Placenta_SexDiff/
```

1. Login to the VPN (virtual private network) using CISCOVPN with your ASUrite credentials. You may need Duo login.
2. Log into ASU network using VPN; you MUST be logged into the VPN in order to proceed.
3. Navigate to <https://login.rc.asu.edu/> (<https://login.rc.asu.edu/>) (if you are not logged into the VPN, the webpage will redirect you to the VPN login page).
4. Click the 'Interactive Apps' menu and select Clusters -> >\_Agave Shell Access option





Open OnDemand provides an integrated, single access point for all of your HPC resources.

Use the navigation bar at the top to get started.

[Click here for cluster status.](#)

## Announcements

### Upcoming Workshops

We regularly host workshops, [sign up today!](#)

### Figure. Launching Agave Shell Access from the ASU Research computing Site.

Select the 'Clusters' menu and then '>\_Agave Shell Access'

This takes you to a login node on ASU Agave supercomputer (one of agave1, agave2, or agave3). If you were going to do data processing, you would connect to one of the compute nodes on the cluster by entering the command 'interact'. If you are just doing a simple copy operation of small files we can stay in the login node.

5. To see where you are when you log in, you can enter `pwd` to print the working directory.

```
pwd
```

If you aren't there already, use the `cd` command to change directory into your home directory:

```
cd /home/username
```

Enter the `ls` command to list the contents.

```
ls
```

To make sure you can access the path containing the template code, list the contents of this path by entering:

```
ls /data/mwilsons/GenomicsResearchExperience/Placenta_SexDiff
```

You should see a zipped file called `CURE_2022_Scripts.zip` and the directory containing the gene count matrix generation pipeline. Copy the `CodePackage.zip` into the directory you just made using this command:

```
cp  
/data/mwilsons/GenomicsResearchExperience/Placenta_SexDiff/CURE_2022_Scripts.zip /
```

Enter `ls` to list the contents again, and now you should see the `CURE_2022_Scripts.zip` file in your home directory.

```
ls
```

Unzip the zipped code package dump the contents into a directory in the current directory (your home directory):

```
unzip CURE_2022_Scripts.zip
```

List directory contents to see if it worked, this time you should see a new directory called `CURE_2022_Scripts`.  
using the `cd` command and list the contents to see all the template code for the next several weeks:

```
ls  
cd CURE_2022_Scripts  
ls
```

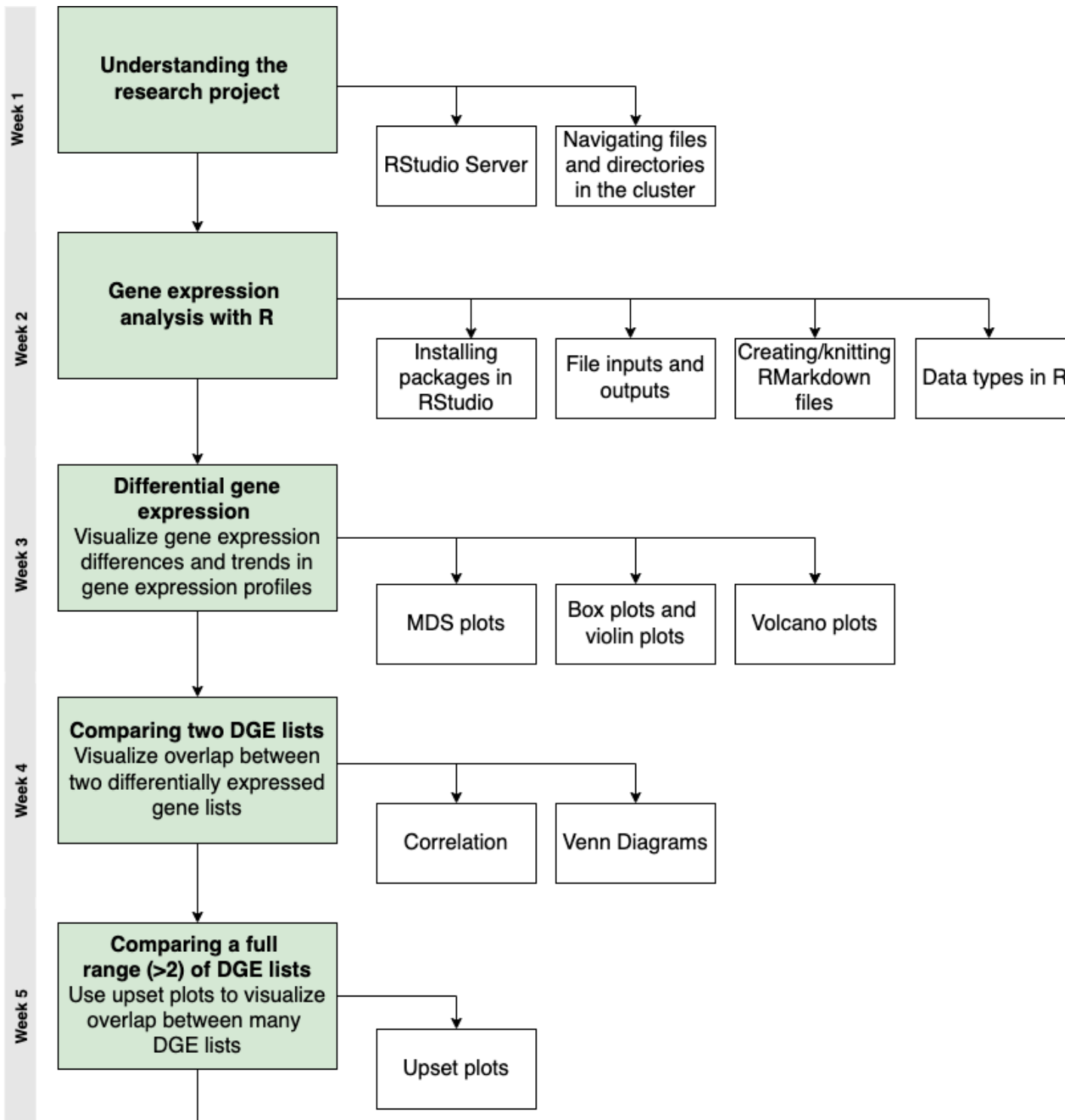
Now you have your own copies of the template code to use and change as you like in

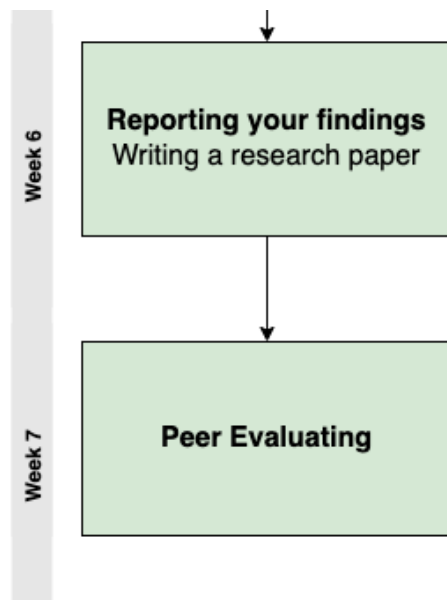
```
/home/<ASUrite_id>/CURE_2022_Scripts/
```

We will teach how to load this code into RStudio graphical interface so it is easier to work with.

## Analysis

In this module, you learned how to set up the tools and data you will be using throughout the remainder of the course. We will walk you through how we think we can answer the research question week by week:





**Figure. Coding objectives for the differential expression workflow by week.**

This green boxes show the topic of each module and the white boxes are the Learn Coding objectives you will cover.

Research does not always go as planned so we can deviate from this timeline if the results call for it. We encourage questions along the way as you learn important skills for data analysis and interpretation.

---

## Module 1.2 Additional Resources

- [The R Project for Statistics](https://www.r-project.org/) ➞ [\(https://www.r-project.org/\)](https://www.r-project.org/)
- [Download RStudio IDE \(FREE\)](https://www.rstudio.com/products/rstudio/download/) ➞ [\(https://www.rstudio.com/products/rstudio/download/\)](https://www.rstudio.com/products/rstudio/download/)
- [Github | Sex Chr Lab | Genomics Cure](https://github.com/SexChrLab/Genomics_CURE/tree/main/data_gen) ➞ [\\_CURE/tree/main/data\\_gen](https://github.com/SexChrLab/Genomics_CURE/tree/main/data_gen)
- [Basic Linux commands tutorial](https://www.pluralsight.com/guides/beginner-linux-navigation-manual) ➞ [\\_CURE/tree/main/data\\_gen](https://www.pluralsight.com/guides/beginner-linux-navigation-manual)