CSAFE Computing Resources

Stephanie Reinders

2022-06-17

Contents

Chapter 1

Introduction

This book gives the reader step-by-step instructions for accessing and using several CSAFE computing resources. CSAFE has access to

1. Large Scale Storage (LSS)
2. CSAFE Rstudio Servers
3. The Pronto Job Scheduler

1.1 Large Scale Storage (LSS)

ResearchIT at Iowa State University provides Large Scale Storage (LSS) at low cost to research labs. CSAFE has purchased more than 30 terabytes of space on LSS. As of 2022-06-17 the following CSAFE project folders are on LSS:

- csafe-admin-vid
- csafe-firearms
- csafe-handwriting
- csafe-handwriting-irb
- csafe-mentisdb
- csafe-mobile
- csafe-shoeprints
- csafe-stego
- csafe-transcripts
- csafe-wvmediaimp

Chapter 3 explains how to connect to LSS.

1.2 CSAFE RStudio Servers

CSAFE has access to two RStudio Servers. To the user, the experience is almost exactly the same as using RStudio on a personal computer. The main benefit of running RStudio on the server is that the server has more RAM and more cores than most personal computers.

Chapter 4 explains how to use the Rstudio servers.

1.3 The Pronto Job Scheduler

ResearchIT at ISU manages a whole host of high performance servers to which CSAFE has purchased access. Instead of accessing one of these servers individually, users submit their jobs (scripts) to the Pronto Job Scheduler which assigns the jobs to the most appropriate servers.

Chapter 5 explains how to run R scripts on the Pronto Job Scheduler. Python, MATLAB, and other programming languages can be run on Pronto, but this book only covers R.

1.4 Do I use the CSAFE Rstudio Server or Pronto?

When deciding whether to use an RStudio server or Pronto, you might want to consider the following benefits and drawbacks of each.

RStudio Server

- *benefits*
  - It runs the RStudio graphical user interface so the user experience is quite similar to working in Rstudio on a personal computer.
- *drawbacks*
  - Each server has less than 20 cores
  - It is a shared user machine which means that if a lot of people are using the server, your jobs might run slower.

Pronto

- *benefits*
  - It has dozens of servers with hundreds of cores each

  - It manages the resources for users so you don't have to worry about slowing down other peoples jobs

- *drawbacks*
  - It doesn't have a graphical user interface so users interact with Pronto through the command line
  - Users submit their jobs to Pronto using something called a SLURM script, which is quite a different workflow compared running scripts in Rstudio

Chapter 2

Getting Started

If you are working from off campus you need to use ISU's virtual private network (VPN) to connect to any of the computing resources covered in this guide. You also need to set up SSH keys. Section **??**(off-campus) explains how to install and use the ISU VPN and SSH keys. If you will only be working on campus, you may skip this section.

This chapter also includes a Quick Start Guide (Section **??**(quick-start)) for people who are already comfortable with virtual private networks and command line prompts.

The last section in this chapter explains how people are granted access to the computing resources covered in this guide.

2.1 Quick Start Guide

2.1.1 Connect to Large Scale Storage

1. If you are an undergraduate student, use your CSAFE email net-id and password in the following steps.
2. If you are off campus, download the ISU virtual private from https://vpn.iastate.edu and log-in using your ISU net-id and password.
3. On **Windows** open File Explorer and in the top textbox type

   - `\\las.iastate.edu\research` if you are **on campus** or
   - `\\las-dfs-01.las.iastate.edu\lss` if you are **off campus**.

4. On **Mac** click Go > Connect to Server and type

   - `smb://iastate/lss/research` if you are **on campus** or
   - `smb://las-dfs-01.las.iastate.edu` if you are **off campus**.

5. Enter your net-id and password.
6. Open the research folder and go to your CSAFE project folder.

2.1.2 Connect to CSAFE Rstudio Server

1. If you are an undergraduate student, use your CSAFE email net-id and password in the following steps.
2. If you are off campus, download the ISU virtual private from https://vpn.iastate.edu and log-in using your ISU net-id and password.
3. Go to the website https://reiss.csafe.iastate.edu/ or https://locard.csafe.iastate.edu/.
4. Enter your net-id and password.

2.1.3 Connect to the Pronto Job Scheduler

1. If you are an undergraduate student, use your CSAFE email net-id and password in the following steps.
2. If you are off campus, download the ISU virtual private from https://vpn.iastate.edu and log-in using your ISU net-id and password.
3. On **Windows** open the PowerShell. On **Mac** open the Terminal.
4. Type `ssh your-netid@pronto.las.iastate.edu`
5. Enter your net-id and password.

2.2 Working Off Campus

If you are on campus, you may skip this section. If you are off campus, you will need to

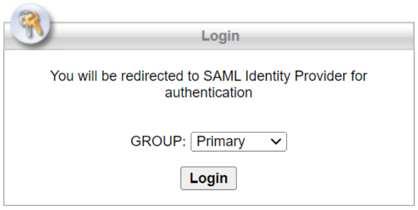
1. Download and sign in to a virtual private network (see 2.2.1)
2. Generate SSH keys (see 2.2.2)

2.2.1 Install the ISU Virtual Private Network (VPN)

Iowa State University uses Cisco AnyConnect as their virtual private network (VPN) software. You will need to install it on your computer.

1. Go to https://vpn.iastate.edu

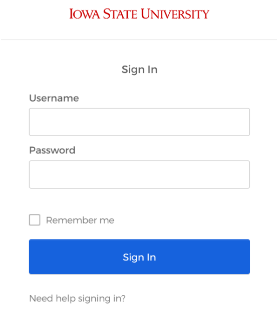
2. Select Primary as the GROUP and click Login


Login

You will be redirected to SAML Identity Provider for authentication

GROUP: Primary

3. Sign in with your ISU net-id and password


Iowa State University

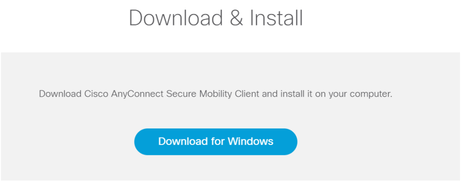
Sign In

Username

Password

□ Remember me

4. The site should detect your computer's operating system and give you the option to download the VPN for your operating system.

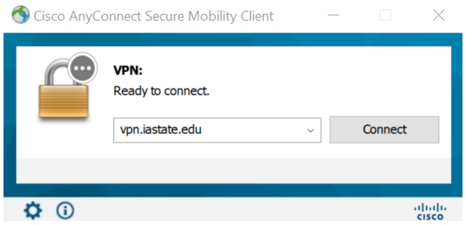

Download & Install

Download Cisco AnyConnect Secure Mobility Client and install it on your computer.

5. Click Download ### Sign In to the ISU VPN {#signin-vpn}

6. Open Cisco AnyConnect

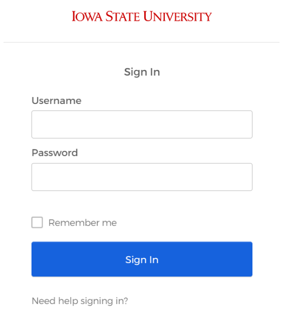
7. Type `vpn.iastate.edu` and click Connect


Cisco AnyConnect Secure Mobility Client

VPN:
Ready to connect.

vpn.iastate.edu

8. Sign in with your ISU net-id and password



2.2.2 Generate SSH keys

ResearchIT gives instructions for setting up SSH keys at https://researchit.las.iastate.edu/how-generate-ssh-keys. Setting up my SSH keys took lots of trial and error. If you run into problems or get stuck, email researchit@iastate.edu.

Chapter 3

Large Scale Storage (LSS)

The Large Scale Storage (LSS) is a research file storage service at ISU. CSAFE has purchased storage space on LSS for a handful of CSAFE projects, such as firearms and handwriting, to store project data.

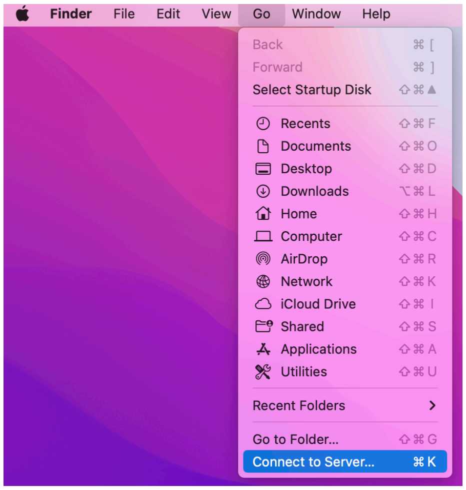
*** Add discussion of folder privileges and their impact on storing IRB data on the server

Windows

1. If you are **off campus**, log-in to the ISU VPN
2. Open file explorer
3. If you are **off campus** type `\\las-dfs-01.las.iastate.edu\lss` in the top textbox. If you are **on campus** you only need to type `\\las.iastate.edu\lss`
4. Enter your ISU net-id and password
5. Open the research folder
6. Open folder for your CSAFE project

Mac

1. If you are off campus, log-in to the ISU VPN
2. Select Go > Connect to Server



3. If you are **off campus** type `smb://las-dfs-01.las.iastate.edu` in the text box. If you are **on campus** type `smb://las.iastate.edu` Click the + sign if you would like your computer to remember the server
4. Click Connect
5. Enter your net-id and password
6. Open the LSS folder
7. Open the research folder
8. Open folder for your CSAFE project

Chapter 4

The CSAFE RStudio Servers

CSAFE has two RStudio Servers that are operated by CSSM:

1. https://reiss.csafe.iastate.edu/
2. https://locard.csafe.iastate.edu/

*** Email cssm_it@iastate.edu for help

4.1 Connect to an RStudio Server

1. If you are off campus, connect to the ISU VPN
2. Go to the website https://reiss.csafe.iastate.edu/ or https://locard.csafe.iastate.edu/
3. Log in with your ISU net-id and password. If you are an undergraduate student, use your CSAFE net-id.

4.2 Working with Directories and File Paths

*** Explain how to change your working directory to one of the CSAFE project folders on the server *** Explain how file paths need to be formatted for files in one of the CSAFE projects on the server

4.3 Trouble Shooting

*** Explain: try clearing cookies if you are unable to log-in

Chapter 5

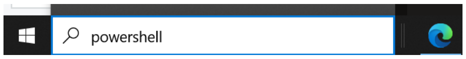
Pronto

5.1 Connecting to the Pronto job scheduler

You connect to the pronto job scheduler through the PowerShell on Windows or the Terminal on Mac.

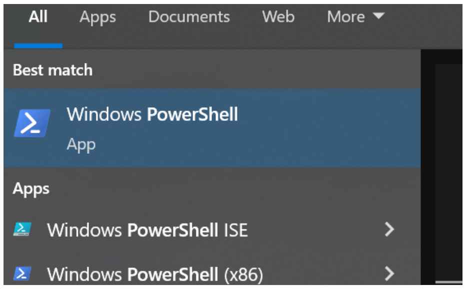
Windows

1. Type `powershell` in the Windows Search Bar next to the Start Menu

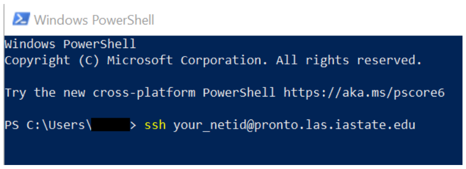


powershell

2. Click Windows PowerShell in the list to open the PowerShell



3. Type `ssh your_netid@pronto.las.iastate.edu`



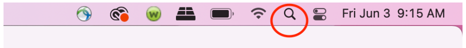
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\[redacted]> ssh your_netid@pronto.las.iastate.edu
```

4. Type your net-id password
5. You should now see

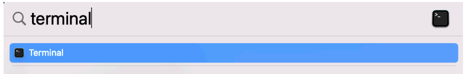
```
[your_netid@pronto ~]$
```

Mac

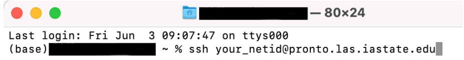
1. Do a spotlight search by clicking on the magnifying glass in the upper right corner of the screen



2. Type `terminal` and click Terminal in the list



3. In the Terminal widow, type `ssh your_netid@pronto.las.iastate.edu`



```
[redacted] — 80×24
Last login: Fri Jun  3 09:07:47 on ttys000
(base)[redacted] ~ % ssh your_netid@pronto.las.iastate.edu
```

4. Type your net-id password

5.2 Connecting to the Pronto data transfer node

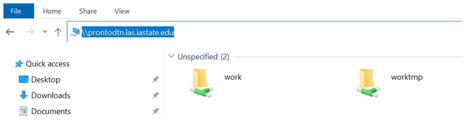
You need to transfer your data to Pronto using the Pronto data transfer node (ProntoDTN). CSAFE has a folder for CSAFE users on ProntoDTN. The file path is `prontodtn.las.iastate.edu > work > LAS > csafe-lab`. Create a folder for yourself in the csafe-lab folder. Most users use their net-id as the folder name, but I think you can name it anything you want.

The work directory is not backed-up so don't use this folder for long term storage.

Each user also has a home folder. These folders have limited storage so researchIT recommends only placing small files in the home folder. I find it easier to store everything in my work folder and not use my home folder at all.

Windows

1. Open file explorer
2. Type `\\prontodtn.las.iastate.edu` in the top textbox



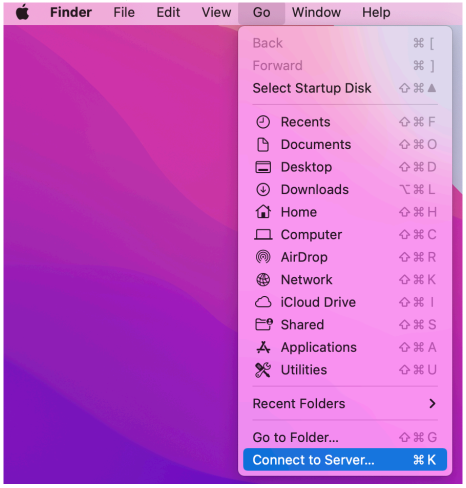
```
\\prontodtn.las.iastate.edu
```

3. Open the work folder

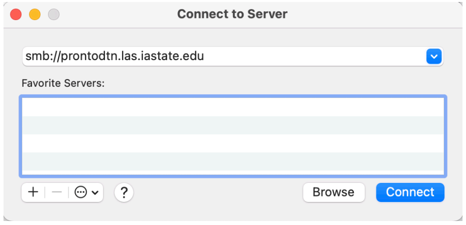
4. Open the LAS folder
5. Open the csafe-lab folder
6. Create a new folder with your net-id as the name

Mac

1. Select Go > Connect to Server



2. Type `smb://prontodtn.las.iastate.edu` in the text box. Click the + sign if you would like your computer to remember the server.



smb://prontodtn.las.iastate.edu

3. Click Connect
4. Enter your net-id and password
5. Select Work and click OK.
6. Open the LAS folder
7. Open the csafe-lab folder
8. Create a new folder with your net-id as the name

5.2.1 Transfering data from ProntoDTN to LSS

If you want to transfer large amounts of data from the Pronto Data Transfer Node (ProntoDTN) to Large Scale Storage (LSS) do the following:

1. Open the PowerShell on Windows or the Terminal on Mac.

2. Connect to the Pronto Data Transfer Node by typing `ssh your_netid@prontodtn.las.iastate.e`
3. Type `cp -r /work/LAS/csafe-lab/your_netid/copy_from_this_folder/ /lss/research/csafe-handwriting/copy_to_this_folder/`

5.2.2 Transfering large amounts of data using Globus

If you have large amounts of data to transfer, researchIT recommends that you use Globus Connect Personal. Go to https://researchit.las.iastate.edu/how-transfer-files-pronto-globus to download Globus.

Instructions for transferring data with Globus are here: https://docs.globus.org/how-to/get-started/.

5.3 Running an R script on Pronto

There are two ways to run a script on Pronto:

1. in an interactive session
2. as a batch job

If I understand correctly, an interactive session is intended for troubleshooting your code. ResearchIT says, “Once you’ve got your program running in an interactive session please switch to an sbatch script if possible.” [https://researchit.las.iastate.edu/pronto?msclkid=3263c889c17811ec81e13068586d8872]

Pronto uses the Simple Linux Utility for Resource Management (Slurm) to allocate resources (computer nodes and clusters) to users [https://en.wikipedia.org/wiki/Slurm_Workload_Manager]. In order to run your script as a batch job, you need to write a Slurm file that includes information about your job and the resources your would like to request.

5.4 Interactive Sessions

5.4.1 Example: Start an interactive session

1. If you are off campus, connect to the ISU VPN (see **??**)

2. Sign-in to Pronto (see 5).

3. In the Pronto terminal, type

```
$ srun --nodes 1 --tasks 4 --partition interactive --time 01:00:00 --pty bash
```

This starts an interactive session.

4. Load the module for R version 4.0.4. This version of R requires that a specific version of gcc be loaded first.

```
$ module load gcc/10.2.0-zuvaafu
$ module load r/4.0.4-py3-4khjixy
```

5. Tell Pronto to install R packages in your csafe-lab work folder. (This is considered the best practice by ResearchIT https://researchit.las.iastate.edu/how-run-r-pronto?msclkid=865a487bc65e11ecbfd7e59a7ab1dc47.)

```
$ export R_LIBS_USER=/work/LAS/your-lab/yournetid/Rlibs
$ mkdir -p $R_LIBS_USER
```

6. Start the R interpreter

```
$ R
```

5.4.2 Example: Install a package from CRAN

1. Start an interactive R session (see 5.4.1)

2.
```
> install.packages("dplyr",lib="/work/LAS/csafe-lab/your_netid/Rlibs", repos="https://mirror
```

5.4.3 Example: Install a package from Github

Occasionally, you might want to install a package that isn't available on CRAN but is available on GitHub. This example shows how to do that on Pronto.

1. Start an interactive R session (see 5.4.1)

2. If you haven't already, you need to install the devtools package

```
> install.packages("devtools",lib="/work/LAS/csafe-lab/your_netid/Rlibs", repos="https://mir
```

3. Every GitHub repository url has the format https://github.com/GithubUsernam/GithubRepo. Use the GithubUsername/GithubRepo in the install_github function.

```
> devtools::install_github("GithubUsername/GithubRepo")
```

5.5 Batch Jobs

5.5.1 Example: A basic R script

1. If you are off campus, connect to the ISU VPN (see **??**)

2. Sign-in to Pronto (see 5).

3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)

4. Create a new R script with the following lines of code:

   ```
   a = 5
   print(a)
   ```

   Save the script as `basic.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

   ```
   #!/bin/bash

   #SBATCH --nodes=1 # request one node
   #SBATCH --cpus-per-task=1  # ask for 1 cpu
   #SBATCH --mem=1G #  asks for 1 GB of RAM
   #SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2

   # everything below this line is optional
   #SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console ou
   #SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messag

   module load r
   cd /work/LAS/csafe-lab/your_netid
   R --save < basic.R
   ```

   Save the file as `basic.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix basic.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.

8. In the command line prompt where you are connected to Pronto, type

   ```
   $ sbatch /work/LAS/csafe-lab/your_netid/basic.txt
   ```

   This will add your job to the queue. The script is so short, your job will probably run immediately.

9. Two new files – `job_<job#>_err.txt` and `job_<job#>_out.txt` – should appear in your csafe-lab work folder. If `basic.R` ran successfully, the `job_<job#>_out` file should contain

   ```
   > a = 5
   > print(a)
   [1] 5
   >
   ```

   and `job_<job#>_err.txt` should be blank. If the error file isn't blank, troubleshoot the error(s). If you get stuck, email researchit@iastate.edu.

5.5.2 Example: Save a dataframe to a CSV file

In this example, I will show you how to save a dataframe to a csv file.

1. If you are off campus, connect to the ISU VPN (see **??**)

2. Sign-in to Pronto (see 5).

3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)

4. Create a new R script with the following lines of code:

   ```
   # draw 10 samples from normal distribution
   data = rnorm(10)

   # put in a dataframe
   df = data.frame(x=data)

   # save dataframe
   write.csv(df, "data.csv")
   ```

   Save the script as `dataframe.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

   ```
   #!/bin/bash

   #SBATCH --nodes=1 # request one node
   #SBATCH --cpus-per-task=1  # ask for 1 cpu
   #SBATCH --mem=1G #  asks for 1 GB of RAM
   #SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2

   # everything below this line is optional
   #SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console out
   #SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messag

   module load r
   cd /work/LAS/csafe-lab/your_netid
   R --save < dataframe.R
   ```

   Save the file as `dataframe.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix dataframe.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.

8. In the command line prompt where you are connected to Pronto, type

   ```
   $ sbatch /work/LAS/csafe-lab/your_netid/dataframe.txt
   ```

   This will add your job to the queue. The script is so short, your job will probably run immediately.

9. The dataframe should be saved in a file `data.csv` in your csafe-lab work folder. There should be two other new files – `data.csv`, `job_<job#>_err.txt` and `job_<job#>_out.txt` – in your csafe-lab work folder as well. If the error file isn't blank, troubleshoot the error(s). If you get stuck, email researchit@iastate.edu.

5.5.3 Example: Load data from a CSV file

In this example, we will load the csv file `data.csv` that we saved in Example 5.5.2.

1. If you are off campus, connect to the ISU VPN (see **??**)

2. Sign-in to Pronto (see 5).

3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)

4. Double-check that the file `data.csv` is still in your csafe-lab work folder. If you deleted it, follow the steps in example 5.5.2 to recreate it.

5. Create a new R script with the following lines of code:

   ```
   df = read.csv("data.csv")
   head(df)
   ```

   Save the script as `load.R` and copy it to you csafe-lab work folder.

6. Create a text file for Slurm with the following text:

   ```
   #!/bin/bash

   #SBATCH --nodes=1 # request one node
   #SBATCH --cpus-per-task=1  # ask for 1 cpu
   #SBATCH --mem=1G #  asks for 1 GB of RAM
   #SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2 seconds.

   # everything below this line is optional
   #SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console output
   #SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messages

   module load r
   cd /work/LAS/csafe-lab/your_netid
   R --save < load.R
   ```

   Save the file as `load.txt`.

7. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix load.txt` to change the line breaks to UNIX format.

8. Copy the text file to your csafe-lab work folder.

9. In the command line prompt where you are connected to Pronto, type

```
$ sbatch /work/LAS/csafe-lab/your_netid/load.txt
```

This will add your job to the queue. The script is so short, your job will probably run immediately.

10. Open the file `job_<job#>_out.txt` in your csafe-lab work folder and make sure that the first six lines of the dataframe were printed.

5.5.4 Example: Install R packages

Just like in Rstudio on your local machine, you will need to install any R packages that you want use. In this example, we will install futile.logger, doParallel, doRNG, and tidyverse so that we can use these packages in future examples.

1. If you are off campus, connect to the ISU VPN (see **??**)
2. Sign-in to Pronto (see 5).
3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)
4. Create a new R script with the following lines of code:

```
install.packages(c("futile.logger",
                   "doParallel",
                   "doRNG",
                   "tidyverse"),
                   lib="/work/LAS/csafe-lab/your_netid/Rlibs",
                   repos="https://mirror.las.iastate.edu/CRAN")
```

   The line starting with `lib=` tells Pronto to install the R packages in your csafe-lab work folder. The line starting with `repos=` tells Pronto which CRAN mirror to use. ResearchIT says that these two lines should always be used [https://researchit.las.iastate.edu/how-run-r-pronto]. Save the script as `install_packages.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

```
#!/bin/bash

#SBATCH --nodes=1 # request one node
#SBATCH --cpus-per-task=1  # ask for 1 cpu
#SBATCH --mem=1G #  asks for 1 GB of RAM
#SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2
```

```
# everything below this line is optional
#SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console output
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messages

export R_LIBS_USER=/work/LAS/csafe-lab/your_netid/Rlibs
mkdir -p $R_LIBS_USER

module load r
cd /work/LAS/csafe-lab/your_netid
R --save < install_packages.R
```

Save the file as `install_packages.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix install_packages.txt` to change the line breaks to UNIX format.
7. Copy the text file to your csafe-lab work folder.
8. In the command line prompt where you are connected to Pronto, type

```
$ sbatch /work/LAS/csafe-lab/your_netid/install_packages.txt
```

This will add your job to the queue. Just like on your local machine, it might take several minutes to install the packages.

9. There should be two other new files – `data.csv`, `job_<job#>_err.txt` and `job_<job#>_out.txt` – in your csafe-lab work folder.

5.5.5 Example: Run code in parallel

The main benefit of Pronto is that it can run code in parallel. In this example, I will show you how to run a for loop in parallel using the doParallel library. R has other packages for parallel computing, but ResearchIT uses doParallel in their how-to [https://researchit.las.iastate.edu/how-run-r-pronto] so that's what I will use here.

If you completed Example 5.5.4 the doParallel package will already be installed. If you need to install the doParallel package see Examples 5.4.2 or 5.5.4 for instructions.

1. If you are off campus, connect to the ISU VPN (see **??**)

2. Sign-in to Pronto (see 5).

3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)

4. Create a new R script with the following lines of code:

```
library(doParallel)

nCores = as.integer(Sys.getenv("SLURM_CPUS_PER_TASK"))
myCluster = parallel::makeCluster(nCores)
doParallel::registerDoParallel(myCluster)

results = foreach(i=1:10) %dopar% {
  rnorm(1000)}
saveRDS(results, "parallel_results.rds")
```

   Save the script as `parallel.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

```
#!/bin/bash

#SBATCH --nodes=1 # request one node
#SBATCH --cpus-per-task=10  # ask for 10 cpu
#SBATCH --mem=1G #  asks for 1 GB of RAM
#SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2

# everything below this line is optional
#SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console ou
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messag

export R_LIBS_USER=/work/LAS/csafe-lab/your_netid/Rlibs

module load r
cd /work/LAS/csafe-lab/your_netid
R --save < parallel.R
```

   Save the file as `parallel.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix parallel.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.
8. In the command line prompt where you are connected to Pronto, type

```
$ sbatch /work/LAS/csafe-lab/your_netid/parallel.txt
```

9. The results of the parallel loop should be saved in `parallel.txt` in your csafe-lab work folder.

5.5.6 Example: Check whether code is running in parallel

1. If you are off campus, connect to the ISU VPN (see **??**)
2. Sign-in to Pronto (see 5).
3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)
4. Create a new R script with the following lines of code:

```
library(doParallel)
library(doRNG)


nCores = as.integer(Sys.getenv("SLURM_CPUS_PER_TASK"))
myCluster = parallel::makeCluster(nCores)
doParallel::registerDoParallel(myCluster)

r1 = foreach(i=1:10) %dorng% {
  if (i==1){Sys.sleep(30)}
  time = Sys.time()
  saveRDS(object=time, file=paste0("time_", i,".rds"))
}
```

   Save the script as `check_parallel.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

```
#!/bin/bash

#SBATCH --nodes=1 # request one node
#SBATCH --cpus-per-task=10  # ask for 10 cpu
#SBATCH --mem=1G #  asks for 1 GB of RAM
#SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2 seconds.
```

```
# everything below this line is optional
#SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console out
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messag

export R_LIBS_USER=/work/LAS/csafe-lab/your_netid/Rlibs

module load r
cd /work/LAS/csafe-lab/your_netid
R --save < check_parallel.R
```

Save the file as `check_parallel.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix check_parallel.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.

8. In the command line prompt where you are connected to Pronto, type

```
$ sbatch /work/LAS/csafe-lab/your_netid/check_parallel.txt
```

9. The script should have created 10 RDS files in your csafe-lab work folder. Check the times recorded in each file. The first file should contain a time after the other files.

5.5.7 Example: Set the seed for the random number generator

If you run Example 5.5.5 multiple times, the numbers sampled from the normal distribution will likely be different each time. If you want to make the sample reproducible (produce the same sample each time), set the seed for the random number generator. The most common way to set a seed is with the `set.seed()` function. This function does not work with `%dopar%` from the doParallel package, but it does work with `%dorng%`.

1. If you are off campus, connect to the ISU VPN (see **??**)

2. Sign-in to Pronto (see 5).

3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)

4. Create a new R script with the following lines of code:

   ```
   library(doParallel)
   library(doRNG)

   nCores = as.integer(Sys.getenv("SLURM_CPUS_PER_TASK"))
   myCluster = parallel::makeCluster(nCores)
   doParallel::registerDoParallel(myCluster)

   starting_seed = 300

   set.seed(starting_seed)
   r1 = foreach(i=1:4) %dorng%{ runif(1) }

   set.seed(starting_seed)
   r2 = foreach(i=1:4) %dorng%{ runif(1) }
   identical(r1, r2)
   ```

   Save the script as `seed.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

   ```
   #!/bin/bash

   #SBATCH --nodes=1 # request one node
   #SBATCH --cpus-per-task=10  # ask for 10 cpu
   #SBATCH --mem=1G #  asks for 1 GB of RAM
   #SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2 seconds.

   # everything below this line is optional
   #SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console output
   #SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messages

   export R_LIBS_USER=/work/LAS/csafe-lab/your_netid/Rlibs

   module load r
   cd /work/LAS/csafe-lab/your_netid
   R --save < seed.R
   ```

   Save the file as `seed.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix seed.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.
8. In the command line prompt where you are connected to Pronto, type

```
$ sbatch /work/LAS/csafe-lab/your_netid/seed.txt
```

9. Open the `job_<job#>_out.txt` file to make sure that the result of `identical(r1, r2)` is `TRUE`.

5.5.8 Example: Use functions from dplyr and magrittr in parallel

The previous parallel examples used the base R functions `rnorm` and `runif`. If we want to use functions from packages that we install, we need to pass the package names to the `%dorng%` operator. We will learn how to do this in this example.

If you haven't already installed the `tidyverse` package, follow the steps in Example 4 to install the `tidyverse` or install `dplyr` and `magrittr` separately.

1. If you are off campus, connect to the ISU VPN (see **??**)
2. Sign-in to Pronto (see 5).
3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)
4. Create a new R script with the following lines of code:

```
library(doParallel)
library(doRNG)
library(magrittr)
library(dplyr)

nCores = as.integer(Sys.getenv("SLURM_CPUS_PER_TASK"))
myCluster = parallel::makeCluster(nCores)
doParallel::registerDoParallel(myCluster)

set.seed(100)
r1 = foreach(i=1:10, .packages=c("dplyr", "magrittr")) %dorng%{
  cluster_centers = df %>% slice_sample(n=40)
}

set.seed(100)
r2 = foreach(i=1:10, .packages=c("dplyr", "magrittr")) %dorng%{
  cluster_centers = df %>% slice_sample(n=40)
}

set.seed(200)
r3 = foreach(i=1:10, .packages=c("dplyr", "magrittr")) %dorng%{
  cluster_centers = df %>% slice_sample(n=40)
}

identical(r1, r2)
identical(r1, r3)
```

Save the script as `magrittr.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

```
#!/bin/bash

#SBATCH --nodes=1 # request one node
#SBATCH --cpus-per-task=10  # ask for 10 cpu
#SBATCH --mem=1G #  asks for 1 GB of RAM
#SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2 seconds.

# everything below this line is optional
#SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console output
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messages

export R_LIBS_USER=/work/LAS/csafe-lab/your_netid/Rlibs

module load r
cd /work/LAS/csafe-lab/your_netid
R --save < magrittr.R
```

Save the file as `magrittr.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix magrittr.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.

8. In the command line prompt where you are connected to Pronto, type

```
$ sbatch /work/LAS/csafe-lab/your_netid/magrittr.txt
```

9. Open the `job_<job#>_out.txt` file to make sure that the result of `identical(r1, r2)` is `TRUE` and `identical(r1, r3)` is `FALSE`.

5.5.9 Example: Use a user-defined function to save results inside a parallel loop

1. If you are off campus, connect to the ISU VPN (see **??**)
2. Sign-in to Pronto (see 5).
3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)
4. Create a new R script with the following lines of code:

```
library(doParallel)
library(doRNG)
source("parallel_save.R")

nCores = as.integer(Sys.getenv("SLURM_CPUS_PER_TASK"))
myCluster = parallel::makeCluster(nCores)
doParallel::registerDoParallel(myCluster)

starting_seed = 300

set.seed(starting_seed)
r1 = foreach(i=1:10) %dorng% {
  save_sample(n=10000, filename=paste0("sample_",i,".rds"))
}
```

   Save the script as `run_parallel_save.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

```
#!/bin/bash

#SBATCH --nodes=1 # request one node
#SBATCH --cpus-per-task=10
#SBATCH --mem=2G
#SBATCH --time=00:15:00

# everything below this line is optional
```

```
#SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console output
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messages

export R_LIBS_USER=/work/LAS/csafe-lab/your_netid/Rlibs

module load r
cd /work/LAS/csafe-lab/your_netid
R --save < run_parallel_save.R
```

Save the file as `parallel_save.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix parallel_save.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.

8. In the command line prompt where you are connected to Pronto, type

   ```
   $ sbatch /work/LAS/csafe-lab/your_netid/parallel_save.txt
   ```

9. There should be ten rds files names sample_1.rds, sample_2.rds, etc. in your csafe-lab folder.

5.5.10 Example: Create a log file

In this example we will create a simple log file using the `futile.logger` package.

1. If you are off campus, connect to the ISU VPN (see **??**)

2. Sign-in to Pronto (see 5).

3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)

4. Create a new R script with the following lines of code:

   ```
   library(futile.logger)

   futile.logger::flog.appender(appender.file("log_file.txt"), name='log')
   futile.logger::flog.info("start script...", name='log')

   a = 5
   futile.logger::flog.info("variable a = %d", a, name='log')
   ```

Save the script as `log.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

```
#!/bin/bash

#SBATCH --nodes=1 # request one node
#SBATCH --cpus-per-task=10  # ask for 10 cpu
#SBATCH --mem=1G #  asks for 1 GB of RAM
#SBATCH --time=00:30:02 # ask that the job be allowed to run for 30 minutes and 2

# everything below this line is optional
#SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console ou
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messag

export R_LIBS_USER=/work/LAS/csafe-lab/your_netid/Rlibs

module load r
cd /work/LAS/csafe-lab/your_netid
R --save < log.R
```

Save the file as `log.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix log.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.

8. In the command line prompt where you are connected to Pronto, type

```
$ sbatch /work/LAS/csafe-lab/your_netid/log.txt
```

9. A log file called `log_file.txt` should appear in your csafe-lab work folder. The contents of the log should look like

```
INFO [2022-05-04 10:42:37] start script...
INFO [2022-05-04 10:42:37] variable a = 5
```

5.5.11 Example: Write a log message inside a parallel loop

1. If you are off campus, connect to the ISU VPN (see **??**)

2. Sign-in to Pronto (see 5).

3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafe-lab > your_netid` (see 5.2)

4. Create a new R script with the following lines of code:

```
  library(doParallel)
  library(futile.logger)

  nCores = as.integer(Sys.getenv("SLURM_CPUS_PER_TASK"))
  myCluster = parallel::makeCluster(nCores, outfile="log_parallel_loop.txt")
  doParallel::registerDoParallel(myCluster)

  # Parallel loop
  r1 = foreach(i=1:4, .packages=c("futile.logger")) %dorng%{
runif(1)
futile.logger::flog.info("Finished iteration %d", i)
  }
```

   Save the script as `parallel_log.R`. Copy the script to your csafe-lab work folder.

5. Create a text file for Slurm with the following text:

```
#!/bin/bash

#SBATCH --nodes=1 # request one node
#SBATCH --cpus-per-task=4  # ask for 4 cpu
#SBATCH --mem=1G #  asks for 1 GB of RAM
#SBATCH --time=00:30:00 # ask that the job be allowed to run for 30 minutes and 0 seconds.

# everything below this line is optional
#SBATCH --output=/work/LAS/csafe-lab/your_netid/job_%J_out.txt # store console output
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messages

export R_LIBS_USER=/work/LAS/csafe-lab/your_netid/Rlibs

module load r
cd /work/LAS/csafe-lab/your_netid
R --save < parallel_log.R
```

   Save the file as `parallel_log.txt`.

6. (Windows Only) If you are using Windows, your text file will likely have DOS line breaks instead of UNIX line breaks which will cause an error when the file is run on Pronto. Open Terminal in RStudio, use `cd`

`folder/containing/text_file` to change directories to the folder containing your text file, and type `dos2unix parallel_log.txt` to change the line breaks to UNIX format.

7. Copy the text file to your csafe-lab work folder.

8. In the command line prompt where you are connected to Pronto, type

```
$ sbatch /work/LAS/csafe-lab/your_netid/parallel_log.txt
```

9. A log file called `log_parallel_loop.txt` should appear in your csafe-lab work folder. The contents of the log should look like

```
starting worker pid=149912 on localhost:11324 at 15:05:55.201
starting worker pid=149924 on localhost:11324 at 15:05:55.452
starting worker pid=149936 on localhost:11324 at 15:05:55.706
starting worker pid=149948 on localhost:11324 at 15:05:55.963
Loading required package: foreach
Loading required package: rngtools
Loading required package: foreach
Loading required package: rngtools
Loading required package: foreach
Loading required package: foreach
Loading required package: rngtools
Loading required package: rngtools
INFO [2022-06-17 15:05:56] Finished iteration 3
INFO [2022-06-17 15:05:56] Finished iteration 1
INFO [2022-06-17 15:05:56] Finished iteration 2
INFO [2022-06-17 15:05:56] Finished iteration 4
```

5.6 Pronto Commands

5.6.1 List loaded modules

1. In an interactive session, type

```
$ module list
```

5.6.2 Purge all loaded modules

1. In an interactive session, type

```
$ module purge
```

5.6.3 View your job(s) in the Pronto queue

1. In pronto terminal (not in an interactive session), type

   ```
   $ squeue -u your-netid
   ```

5.6.4 Cancel a job

1. In pronto terminal (not in an interactive session), type

   ```
   $ scancel jobid
   ```

   where jobid is the six-digit number for the job you want to cancel.

5.7 Install the handwriter package

CSAFE created an R package for statistical handwriting analysis called handwriter. The handwriter package requires a number of other R packages, some of which caused errors when I tried to install handwriter in an interactive session on Pronto following the instructions in 5.4.2. ResearchIT helped me figure out how to install the problem packages and ultimately install handwriter. This section records the step-by-step details of how I was able to install handwriter. My hope is that you will also be able to sucessfully install handwriter if you follow these steps. If you run into problems, email researchit@iastate.edu.

1. If you are off campus, connect to the ISU VPN (see **??**)
2. Sign-in to Pronto (see 5).
3. Some of the packages used by handwriter are only available on Pronto for R 4.0.4. Start the R 4.0.4 interpreter in an interactive session on Pronto.

   ```
   $ srun --nodes 1 --tasks 4 --partition interactive --time 01:00:00 --pty bash
   $ module purge
   $ module load gcc/10.2.0-zuvaafu
   $ module load r/4.0.4-py3-4khjixy
   $ R
   ```

4. Try to install the handwriter package

```
> install.packages("handwriter",lib="/work/LAS/csafe-lab/your_netid/Rlibs", repos=
```

I received error messages stating that the packages igraph, randomForest, and magick could not be installed. I had to install these packages separately.

5. Install igraph

```
> install.packages("igraph",lib="/work/LAS/csafe-lab/your_netid/Rlibs", repos="htt
```

6. Then randomForest

```
> install.packages("https://cran.r-project.org/src/contrib/Archive/randomForest/ra
```

7. Install magick

```
> install.packages("magick",
configure.vars=c("INCLUDE_DIR='. -D_GLIBCXX_USE_CXX11_ABI=0 -fopenmp -DMAGICKCORE_
lib="/work/LAS/csafe-lab/your_netid/Rlibs",
repos="https://mirror.las.iastate.edu/CRAN")
```

5.8 Trouble-Shooting

5.8.1 Line breaks error

If you are using Windows and you try to run a text file created on Windows on Pronto, you will likely get the following error:

```
sbatch: error: Batch script contains DOS line breaks (\\r\\n)
sbatch: error: instead of expected UNIX line breaks (\\n).
```

One way to fix the problem is to - Open Rstudio - Go to Tools > Terminal > New Terminal - Changes the directory to the folder containing basic.txt by typing ‘cd /documents/my/folder’ without the quotes and hitting enter. - Type ‘dos2unix basic.txt’ without the quotes and hit enter.