

CSAFE Computing Resources

Stephanie Reinders

2022-06-07

Contents

1	Introduction	5
2	Getting Started	7
2.1	Connecting from off campus	7
3	File Storage on the CSSM Server	11
4	The CSAFE RStudio Servers	13
4.1	Connect to an RStudio Server	13
4.2	Working with Directories and File Paths	13
4.3	Trouble Shooting	14
5	Pronto	15
5.1	Connecting to the Pronto job scheduler	15
5.2	Connecting to the Pronto data transfer node	16
5.3	Running an R script on Pronto	18
5.4	Interactive Sessions	18
5.5	Batch Jobs	19
5.6	Pronto Commands	32
5.7	Install the handwriter package	33
5.8	Trouble-Shooting	34

Chapter 1

Introduction

This book gives the reader step-by-step instructions for accessing and using the following CSAFE computing resources:

1. File storage on the Center for Survey Statistics & Methodology (CSSM) server
2. High performance servers managed by CSSM and Research IT

Chapter 2

Getting Started

2.1 Connecting from off campus

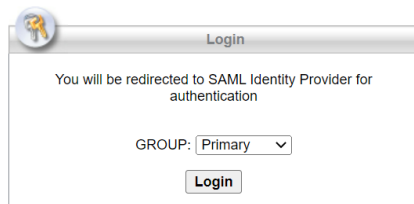
If you are on campus, you may skip to section 5.1. If you are off campus, you will need to

1. Download and sign in to a virtual private network (see 2.1.1)
2. Generate SSH keys (see 2.1.2)

2.1.1 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



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

IOWA STATE UNIVERSITY

Sign In

Username

Password

☐ Remember me

Sign In

[Need help signing in?](#)

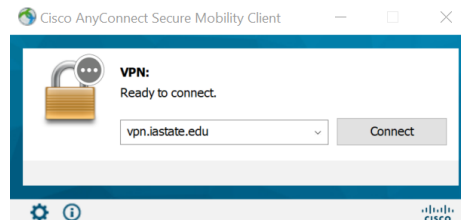
- The site should detect your computer's operating system and give you the option to download the VPN for your operating system.

Download & Install

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

Download for Windows

- Click Download ### Sign In to the ISU VPN {#signin-vpn}
- Open Cisco AnyConnect
- Type `vpn.iastate.edu` and click Connect



- Sign in with your ISU net-id and password

IOWA STATE UNIVERSITY

Sign In

Username

Password

☐ Remember me

Sign In

[Need help signing in?](#)

2.1.2 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

File Storage on the CSSM Server

A number of CSAFE projects, such as firearms and handwriting, store project data on the Center for Survey Statistics & Methodology (CSSM) server.

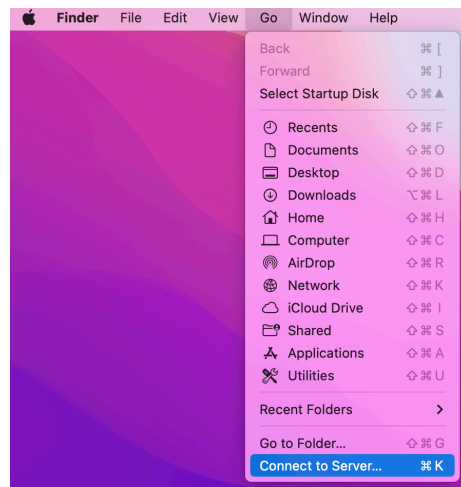
Add discussion of folder privileges and their impact on storing IRB data on the server

Windows

1. If you are off campus, log-in to the ISU VPN
2. Open file explorer
3. Type `\\las-dfs-01.las.iastate.edu\lss` in the top textbox
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. Type `smb://las-dfs-01.las.iastate.edu` in the text box. 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

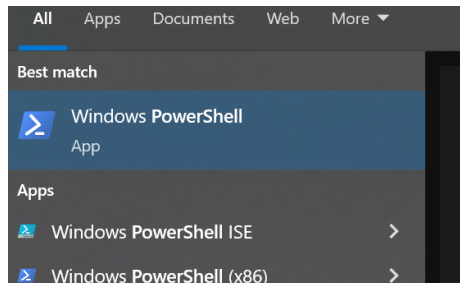
You connect to the pronto job scheduler through the PowerShell on Windows or the Terminal on Mac.

Windows

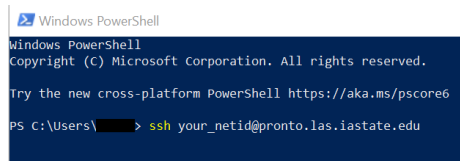
1. Type **powershell** in the Windows Search Bar next to the Start Menu



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



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



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

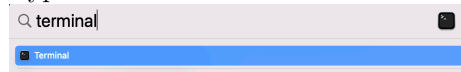
```
[your_netid@pronto ~]$
```

Mac

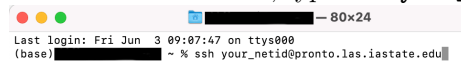
1. Do a spotlight search by clicking on the magnifying glass in the upper right corner of the screen



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



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



4. Type your net-id password

5.2 Connecting to the Pronto data transfer node

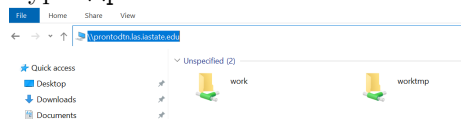
You need to transfer your data to Pronto using the Pronto data transfer node (ProntoDTN). CSAFE has a folder for CSAFE users on ProntoDTN. The file path is `prontodtn.las.iastate.edu > work > LAS > csafe-lab`. Create a folder for yourself in the `csafe-lab` folder. Most users use their net-id as the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

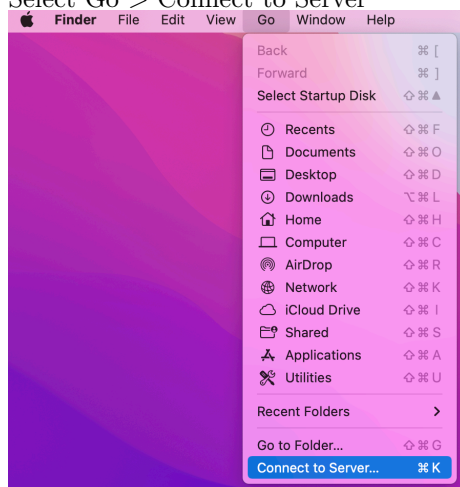


3. Open the work folder

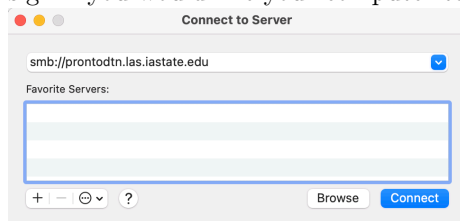
4. Open the LAS folder
5. Open the csafe-lab folder
6. Create a new folder with your net-id as the name

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.



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 Transferring 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 you 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 5.1).
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 dplyr package

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.its-upi.com/CRAN/")`

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 5.1).
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 seconds.
```

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

module load r
cd /work/LAS/csafelab/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/csafelab/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 5.1).

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 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 < 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/csafelab/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 csafelab work folder. There should be two other new files – `data.csv`, `job_<job#>_err.txt` and `job_<job#>_out.txt` – in your csafelab 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 5.1).
3. Open your csafelab work folder `prontodtn.las.iastate.edu > work > LAS > csafelab > your_netid` (see 5.2)
4. Double-check that the file `data.csv` is still in your csafelab 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 your csafelab 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

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

```
module load r
cd /work/LAS/csafelab/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/csafelab/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 5.1).
3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafelab > 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/csafelab/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 out
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error message

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 5.1).
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 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/csafelab/your_netid/Rlibs

module load r
cd /work/LAS/csafelab/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/csafelab/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: 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 5.1).
3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafelab > 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.

5.5.7 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 5.1).
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")) %dornrg%{
  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/csafelab/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.

5.5.8 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 5.1).
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/csafelab/your_netid/job_%J_out.txt # store console output
#SBATCH --error=/work/LAS/csafelab/your_netid/job_%J_err.txt # store error messages

export R_LIBS_USER=/work/LAS/csafelab/your_netid/Rlibs

module load r
cd /work/LAS/csafelab/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/csafelab/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.

5.5.9 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 5.1).
3. Open your csafe-lab work folder `prontodtn.las.iastate.edu > work > LAS > csafelab > 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 out
#SBATCH --error=/work/LAS/csafe-lab/your_netid/job_%J_err.txt # store error messa

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
```

5.6 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 successfully 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 5.1).
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/csafelab/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/csafelab/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/csafelab/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.