

NEON Phenology Exercise

NEON + UF Environmental Science Lab Class

Felipe Sanchez, fsanchez@battelleecology.org (<mailto:fsanchez@battelleecology.org>) Renata Diaz (formatting) October 5, 2020

This exercise is modifications of <https://www.neonscience.org/osis-pheno-temp-series> (<https://www.neonscience.org/osis-pheno-temp-series>), and Modified_code_R_NEON-pheno-temp-timeseries_03-plot-discrete-continuous-data-pheno-temp.R (developed by Felipe Sanchez), set up by RMD to run on UFApps

Getting to UFApps

Here are the instructions to get to UFApps - they are the same as last week.

Navigate to: <https://info.apps.ufl.edu> (<https://info.apps.ufl.edu>)

Click on the orange “Login to UFApps” button in the upper right. This link may also take you to the login page: <https://apps.ufl.edu/> (<https://apps.ufl.edu/>)

Log in using your GatorLink.

Click on “Apps” in the top menu bar. Scroll down to RStudio and click on it to launch it. It may take a minute. Sometimes the connection will not go through, but if you launch it again it works. If you consistently can’t get through, reach out to Renata and she will try and figure out another solution.

You may see a prompt to “set up secure access to your files”. There’s no need to do this for this course, so go ahead and click Not Now.

Setting up to use NEON data in R

These instructions are *the same* as for the data downloading exercise. It will do no harm to run them again, and might help you avoid some headaches with file paths.

Go to the landing page for UFApps and launch RStudio.

As before, RStudio may have saved your workspace from last week’s lab. We want a clean area to copy-paste code. If the top left pane has anything in it (“Untitled1”, “Untitled2”), close them - you don’t need to save them. When you close everything the top left pane may minimize. That’s OK, it will come back!

Select “File”, “New File”, and “RScript” in the top left menu bar. This will give you a new place to copy and paste code.

To run code, either highlight it and click the “Run” button in the top right of the top left pane, or highlight it and press Ctrl + Enter.

Copy-paste this code to set up your files on UFApps for this week’s exercise:

```
setwd("~/")

if(!dir.exists("EVS3000L")) {
  dir.create("EVS3000L")
}

if(!dir.exists("EVS3000L/week6")) {
  dir.create("EVS3000L/week6")
}

setwd("EVS3000L/week6/")
```

We will need to install and load the `neonUtilities` package, which is NEON’s R package for working seamlessly with NEON data in R. If you have already done the Data Download exercise, you do not need to run `install.packages("neonUtilities")` again. It will be perfectly OK if you do, though.

```
install.packages("neonUtilities")
```

You will also need to install `gridExtra` to help make some aesthetically pleasing plots:

```
install.packages("gridExtra")
```

Download data for this exercise in UFApps

Before we start to work with data in R, we need to download the data files for this exercise! Exit RStudio in UFApps and follow these instructions to get set up with the data we need:

- Go to the landing page for UFApps (the page with all the app icons - in my web browser, the tab is called “Citrix Receiver”)
- Launch the Google Chrome app
- Copy-paste this URL into Chrome and push enter: <https://ndownloader.figshare.com/files/22775042> (<https://ndownloader.figshare.com/files/22775042>)
- This will download a .zip file called “NEON-pheno-temp-timeseries_v2.zip”. When the download completes, exit the Chrome app.
- Go back to the landing page and launch the M Drive app.

- Open the Downloads folder.
- Select NEON-pheno-temp-timeseries_v2.zip. You may get a warning that opening files can be harmful to your computer; dismiss it.
- In the file explorer, select “Move to” and then “Choose location”.
- In the window that pops up, navigate to `UApps > Documents > EVS3000L > week6`. Select the `week6` folder and click Move.
- In the file explorer, navigate to `Documents > EVS3000L > week6`.
- Select NEON-pheno-temp-timeseries_v2.zip.
- At the top of the file explorer window, there’s a pink button that says “Extract”. Click on it.
- Click on “Extract all”.
- Make sure the files will be extracted to the folder: `M:\Documents\EVS3000L\week6\NEON-pheno-temp-timeseries_v2`
- Click Extract.
- Exit the M drive app.
- Launch RStudio in UApps to proceed!

Loading packages

Back in RStudio, use this to load the packages we’ll need for this exercise:

```
library(ggplot2)
library(dplyr)
library(gridExtra)
library(scales)

options(stringsAsFactors=F)
```

Loading and formatting data

Now let’s load the data we downloaded and make sure everything is formatted properly:

```
temp_day <- read.csv('NEON-pheno-temp-timeseries_v2/NEON-pheno-temp-timeseries/NEONsaat_daily_SCBI_2018.csv') # RMD modified paths

phe_1sp_2018 <- read.csv('NEON-pheno-temp-timeseries_v2/NEON-pheno-temp-timeseries/NEONpheno_LITU_Leaves_SCBI_2018.csv') # RMD modified paths

# Convert dates
temp_day$Date <- as.Date(temp_day$Date)
# use dateStat - the date the phenophase status was recorded
phe_1sp_2018$dateStat <- as.Date(phe_1sp_2018$dateStat)
```

Phenology is the timing of life events for organisms - like when juveniles become adults, or when

trees being to leaf out, flower, or shed their leaves. Here we will explore how environmental conditions correlate with tree phenology using NEON data.

The data we have downloaded are the **daily maximum air temperature** and **number of individual trees in leaf-out** from 2018 from the NEON site at the Smithsonian Conservation Biology Institute, located near Fort Royal, Virginia. You can read more about this site here:

<https://www.neonscience.org/field-sites/field-sites-map/SCBI> [_ \(https://www.neonscience.org/field-sites/field-sites-map/SCBI\)](https://www.neonscience.org/field-sites/field-sites-map/SCBI)

Rough plots

Let's plot temperature and leafing:

```
## ----stacked-plots-----
# first, create one plot
phenoPlot <- ggplot(phe_1sp_2018, aes(dateStat, countYes)) +
  geom_bar(stat="identity", na.rm = TRUE) +
  ggtitle("Total Individuals in Leaf") +
  xlab("") + ylab("Number of Individuals")

# create second plot of interest
tempPlot_dayMax <- ggplot(temp_day, aes(Date, dayMax)) +
  geom_point() +
  ggtitle("Daily Max Air Temperature") +
  xlab("Date") + ylab("Temp (C)")

# Then arrange the plots - this can be done with >2 plots as well.
grid.arrange(phenoPlot, tempPlot_dayMax)
```

We can tweak the x-axis to make it more legible:

```
## ----format-x-axis-labels-----
# format x-axis: dates
phenoPlot <- phenoPlot +
  (scale_x_date(breaks = date_breaks("1 month"), labels = date_format("%b"))))

tempPlot_dayMax <- tempPlot_dayMax +
  (scale_x_date(breaks = date_breaks("1 month"), labels = date_format("%b"))))

grid.arrange(phenoPlot, tempPlot_dayMax)
```

And add labels to help us interpret what we are seeing:

```
## ----set-x-axis-----
# first, lets recreate the full plot and add in the
phenoPlot_setX <- ggplot(phe_1sp_2018, aes(dateStat, countYes)) +
  geom_bar(stat="identity", na.rm = TRUE) +
  ggtitle("Total Individuals in Leaf") +
```

```

xlab("") + ylab("Number of Individuals") +
scale_x_date(breaks = date_breaks("1 month"),
             labels = date_format("%b"),
             limits = as.Date(c('2018-01-01', '2018-12-31'))))

# create second plot of interest
tempPlot_dayMax_setX <- ggplot(temp_day, aes(Date, dayMax)) +
  geom_point() +
  ggtitle("Daily Max Air Temperature") +
  xlab("Date") + ylab("Temp (C)") +
  scale_x_date(date_breaks = "1 month",
              labels=date_format("%b"),
              limits = as.Date(c('2018-01-01', '2018-12-31'))))

# Plot
grid.arrange(phenoPlot_setX, tempPlot_dayMax_setX)

```

It's hard to see pattern, because the datasets don't overlap fully in their temporal coverage! Let's filter to the Venn diagram of where they do intersect:

```

## ----align-datasets-replot-----
# filter to only having overlapping data
temp_day_filt <- filter(temp_day, Date >= min(phe_1sp_2018$date) &
                        Date <= max(phe_1sp_2018$date))

# Check
range(phe_1sp_2018$date)
range(temp_day_filt$date)

#plot again
tempPlot_dayMaxFiltered <- ggplot(temp_day_filt, aes(Date, dayMax)) +
  geom_point() +
  scale_x_date(breaks = date_breaks("months"), labels = date_format("%b")) +
  ggtitle("Daily Max Air Temperature") +
  xlab("Date") + ylab("Temp (C)")

grid.arrange(phenoPlot, tempPlot_dayMaxFiltered)

```

Save plots for submission

Use this to save your final plots. They will be saved in your M drive on eLearning. Upload them to eLearning as part of your submission for this week. **Also submit either a Word document or a text submission with your answers to the reflection questions [here \(https://ufl.instructure.com/courses/407839/pages/neon-reflection-questions\)](https://ufl.instructure.com/courses/407839/pages/neon-reflection-questions).**

```

##### From RMD - Saving plots for submission
jpeg("pheno_and_temp.jpg")

```

```
grid.arrange(phenoPlot_setX, tempPlot_dayMax_setX)  
dev.off()
```