Formatting Data for Morphology CURE

Alex Krohn

July 31 2020

Introduction

The Morphology CURE comes with a practice dataset of 800 images and associated metadata from herbarium specimens of Purple Loosestrife (*Lythrum salicaria*), on Dryad. However, if you would like to expand the CURE to another species, or download the entire Purple Loosestrife dataset (over 3,000 images), you can use this tutorial to do so with R.

This tutorial assumes you have no R experience, and that you'll be downloading R for the first time.

To download these datasets, first you must download R. Then, programatically speaking, first, you must find the records with photos on iDigBio and download the data about those records. Second, you must create a new spreadsheet with the information that is relevant to our projects: the location of the specimen, the date it was collected, and the link to the image. Third, and finally, you must download the images of those records, label each record uniquely so that it may be matched back to the correct metadata.

Installing and Using R

For this tutorial, you just need to download R. If you get excited about programming in R, I recommend you also download RStudio to really have a complete environment to work in.

To download R, visit the website (https://www.r-project.org/) and follow the instructions to download from your favorite CRAN mirror.

Once R is installed on your computer, double click to run the program. In its essence, R is a console window where you execute commands (code). If you'd like, you can save your sequence of commands in an R script (File -> New Document) to come back to them.

To use this script, simply copy the lines of code into your console, and then press enter. If you decide to save the code into a R script, highlight the code of interest and press Apple or Ctrl + Enter (depending on your computer) to run the code. That's all there is to it! Copy and paste!

Finding and Downloading Data from iDigBio

This tutorial will lead you through the steps to download the rest of the Purple Loosestrife dataset. If you would like to download the data from another species, simply replace Lythrum salicaria with your species of interest. (Don't forget that you may find additional data by searching for synonyms of your species!)

To start, navigate in your browser to https://idigbio.org. Click "Search the Portal." A heat map of millions of specimens should appear. At left, under the "Filters" tab, type Lythrum salicaria into the Scientific Name box. Near the top, check the box indicating that the records "Must have media".

As of July 16th, 2020, 3,257 ocurrence records of *Lythrum salicaria* had 3,340 images on iDigBio. Your numbers may vary. You can download my exact dataset here.

Click on the gray Download tab. Enter your email, and download the CSV. (You may have to wait a few minutes for the download to build and be ready to download.)

Once complete, you should have a zip file containing the records. Unzip the file to reveal a folder with a long name containing various TXT, XML and CSV files. you're going to assume you put these files into a folder called "CURE."

Load the Specimen Metadata

First, set the folder where you working from (i.e. where you unzipped the iDigBio file, and where you will be downloading the images). This is known as your working directory. In the `setwd()` code below, replace the file location inside the quotes and the parentheses with the location on your computer that leads to your downloaded data.

```
setwd("CURE/")
```

Next, if this is your first time using R, you'll have to install a few packages. This is easy, but if your first time, just agree to any windows that pop up. It may take a few minutes to install these two packages. Once you've installed the packages once, you don't need to install them again. You do still need to run the `library()` commands below, though.

```
install.packages("tidyverse")
install.packages("glue")
```

Then, load the data on each specimen occurence, so you can decide which specimens that your class will analyze. Here, you'll randomly subset 1000 specimens, and then only keep relevant information. Colons (":") are a special character in R. R will not treat it as special if the name is within ` `. you will change these names to make them easier to deal with. If you'd like to select more columns, add them to `select()`.

```
# Load the packages that you just installed
library(tidyverse)
library(glue)

# Load the data, keeping only the columns with locality information, a unique specimen ID, and collecti
occ.data <- read_csv("occurrence_raw.csv") %>%
  select(coreid, coll.date = `dwc:eventDate`, country = `dwc:country`, state = `dwc:stateProvince`, cou
```

If you'd could subset the data to a manageable number. For example, the practice set contains 800 random images from this larger dataset. This is optional.

```
occ.data.subset <-
  occ.data %>%
  slice_sample(n = 800)
```

You could, instead, subset by another parameter that's of interest to your class. Subsetting by geography might be best accomplished using iDigBio's filtering parameters on their youbsite. Otherwise, you could use `filter()` as below. Note: not all data in this dataset have coordinates, not all data with coordinates have states/provinces/countries given, and not all data with states/provinces/countries given have coordinates. Filter carefully!

```
# Filter to only records from New Jersey
newjersey.data <-
  occ.data %>%
  filter(`dwc:stateProvince` == "New Jersey")
```

Add the Image URL to the Occurrence Data

Continuing with the full dataset, you'll append the unique specimen identifier (`coreid`), from the multimedia.csv file, to find the correct URL. Then, you'll add this column of URLs to the occurence data spreadsheet.

```
# Import the multimedia data, but only the URL and the coreID
multimedia <-
  read_csv("multimedia.csv") %>%
  select(coreid, url = `ac:accessURI`)

# Add the URL to the occ.data.subset dataframe
occ.data.and.urls <-
  occ.data %>%
  left_join(y = multimedia, by = "coreid")
```

Note, there are some occurrences with repeated coreids. These are specimens with multiple images.

Download the Images and Label them

Next, you'll download the images to your working directory, and rename them with the coreid.

First, create a folder within your CURE folder called images. You can do this in bash using your Terminal (Mac, see below) or Run (PC), or by hand using whatever method you prefer.

```
mkdir images
```

Next, create a vector of names for each image to be saved.

```
image.names <-
  occ.data.and.urls %>%
  select(coreid) %>%
  glue_data("{coreid}.jpg")
```

Next, create a vector of URLs to download the images from.

```
urls <- occ.data.and.urls$url
```

Finally, use the download.file() function to download each file from the URL, label it according to the coreID and store it in the images folder. This may take a while, depending on the number of images.

```
setwd("images/")

# Skip over any broken links
safe_download <- safely(~ download.file(.x, .y, mode = "wb"))

# Execute
walk2(urls, image.names, safe_download)
```