

APPENDIX A. A COMPLETE REPRODUCIBLE WORKFLOW, FROM A SPECIES LIST TO A PHYLOGENY, AND DISTRIBUTION MAP.

If you aren't familiar with a complete workflow in R, it may be difficult to visualize the process. In R, everything is programmatic, so the whole workflow can be in one place, and be repeated whenever necessary. The following is a workflow for *taxize*, going from a species list to a phylogeny.

First, install *taxize*

```
install.packages("taxize")
```

Then load it into R

```
library(taxize)
```

Most of us will start out with a species list, something like the one below. Note that each of the names is spelled incorrectly.

```
splist <- c("Helanthus annuus", "Pinos contorta", "Collomia grandiflorra", "Rosa californica",
  "Mimulus bicolour", "Nicotiana glauca", "Maddia sativa")
```

There are many ways to resolve taxonomic names in *taxize*. Of course, the ideal name resolver will do the work behind the scenes for you so that you don't have to do things like fuzzy matching. There are a few services in *taxize* like this we can choose from: the Global Names Resolver service from EOL (see function *gnr_resolve*) and the Taxonomic Name Resolution Service from iPlant (see function *tnrs*). In this case let's use the function *tnrs*.

```
# The tnrs function accepts a vector of 1 or more
splist_tnrs <- tnrs(query = splist, getpost = "POST", source_ = "iPlant_TNRS")
# Remove some fields
(splist_tnrs <- splist_tnrs[, !names(splist_tnrs) %in% c("matchedName", "annotations",
  "uri")])

#           submittedName      acceptedName    sourceId score
# 5      Helanthus annuus      Helianthus annuus iPlant_TNRS 0.98
# 1         Pinos contorta       Pinus contorta iPlant_TNRS 0.96
# 7 Collomia grandiflorra Collomia grandiflora iPlant_TNRS 0.99
# 6        Rosa californica      Rosa californica iPlant_TNRS 0.99
# 4      Mimulus bicolour       Mimulus bicolor iPlant_TNRS 0.98
# 3      Nicotiana glauca       Nicotiana glauca iPlant_TNRS 1
# 2        Maddia sativa        Madia sativa iPlant_TNRS 0.97

# Note the scores. They suggest that there were no perfect matches, but they
# were all very close, ranging from 0.77 to 0.99 (1 is the highest). Let's
# assume the names in the 'acceptedName' column are correct (and they should
# be).
# So here's our updated species list
(splist <- as.character(splist_tnrs$acceptedName))

# [1] "Helianthus annuus"      "Pinus contorta"        "Collomia grandiflora"
# [4] "Rosa californica"      "Mimulus bicolor"       "Nicotiana glauca"
# [7] "Madia sativa"
```

Another thing we may want to do is collect common names for our taxa.

```
tsns <- get_tsn(searchterm = splist, searchtype = "sciname", verbose = FALSE)
comnames <- lapply(tsns, getcommonnamesfromtsn)
# Unfortunately, common names are not standardized like species names, so
# there are multiple common names for each taxon
sapply(comnames, length)
```

```
# [1] 3 3 3 3 3 3 3

# So let's just take the first common name for each species
comnames_vec <- do.call(c, lapply(comnames, function(x) as.character(x[1, "comname"])))
# And we can make a data.frame of our scientific and common names
(allnames <- data.frame(spname = splist, comname = comnames_vec))

#           spname           comname
# 1 Helianthus annuus common sunflower
# 2 Pinus contorta lodgepole pine
# 3 Collomia grandiflora largeflowered collomia
# 4 Rosa californica California wildrose
# 5 Mimulus bicolor yellow and white monkeyflower
# 6 Nicotiana glauca tree tobacco
# 7 Madia sativa coast tarweed
```

Another common task is getting the taxonomic tree upstream from your study taxa. We often know what family or order our taxa are in, but it we often don't know the tribes, subclasses, and superfamilies. *taxize* provides many avenues to getting classifications. Two of them are accessible via a single function (*classification*): the Integrated Taxonomic Information System (ITIS) and National Center for Biotechnology Information (NCBI); and via the Catalogue of Life (see function *col_classification*):

```
# As we already have Taxonomic Serial Numbers from ITIS, let's just get
# classifications from ITIS. Note that we could use uBio instead.
class_list <- classification(tsns)
sapply(class_list, nrow)

# [1] 12 11 12 12 12 12 12

# And we can attach these names to our allnames data.frame
library(plyr)
gethiernames <- function(x) {
  temp <- x[, c("rankName", "taxonName")]
  values <- data.frame(t(temp[, 2]))
  names(values) <- temp[, 1]
  return(values)
}
class_df <- ldply(class_list, gethiernames)
allnames_df <- merge(allnames, class_df, by.x = "spname", by.y = "Species")
# Now that we have allnames_df, we can start to see some relationships among
# species simply by their shared taxonomic names
allnames_df[1:2, ]

#           spname           comname Kingdom Subkingdom
# 1 Collomia grandiflora largeflowered collomia Plantae Viridaeplantae
# 2 Helianthus annuus common sunflower Plantae Viridaeplantae
# Infrakingdom Division Subdivision Infradivision Class
# 1 Streptophyta Tracheophyta Spermatophytina Angiospermae Magnoliopsida
# 2 Streptophyta Tracheophyta Spermatophytina Angiospermae Magnoliopsida
# Superorder Order Family Genus
# 1 Asteranae Ericales Polemoniaceae Collomia
# 2 Asteranae Asterales Asteraceae Helianthus

# Ah, so Abies and Bartlettia are in different infradivisions, but share
# taxonomic names above that point.
```

However, taxonomy can only get you so far. Shared ancestry can be reconstructed from molecular data, and phylogenies created. Phylomatic is a web service with an API that we can use to get a phylogeny.

```
# Fetch phylogeny from phylomatic
phylogeny <- phylomatic_tree(taxa = as.character(allnames$spname), taxnames = TRUE,
  get = "POST", informat = "newick", method = "phylomatic", storedtree = "R20120829",
  taxaformat = "slashpath", outformat = "newick", clean = "true", parallel = TRUE)
# Format teeth-labels
phylogeny$tip.label <- capwords(phylogeny$tip.label, onlyfirst = TRUE)
# plot phylogeny
plot(phylogeny)
```

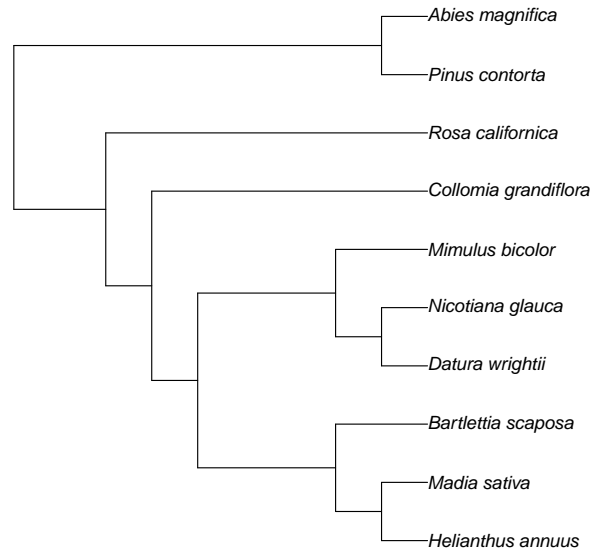


FIG. A.1. A phylogeny.

Using the species list, with the corrected names, we can now search for occurrence data. The Global Biodiversity Information Facility (GBIF) has the largest collection of records data, and has a API that we can interact with programmatically from R. First, we need to install rgbif.

```
# Install rgbif from github.com
install.packages("devtools")
library(devtools)
install_github("rgbif", "ropensci")
```

Now we can search for occurrences for our species list and make a map.

```
library(rgbif)
library(ggplot2)
# get occurrences
occurr_list <- occurrencelist_many(as.character(allnames$spname), coordinatestatus = TRUE,
  maxresults = 100, removeZeros = TRUE, fixnames = "changealltorig")
# Make a map
p <- gbifmap_list(occurr_list) + guides(col = guide_legend(title = "", nrow = 3,
  byrow = TRUE)) + theme(legend.position = "bottom", legend.key = element_blank()) +
```

```
coord_equal()
```

p

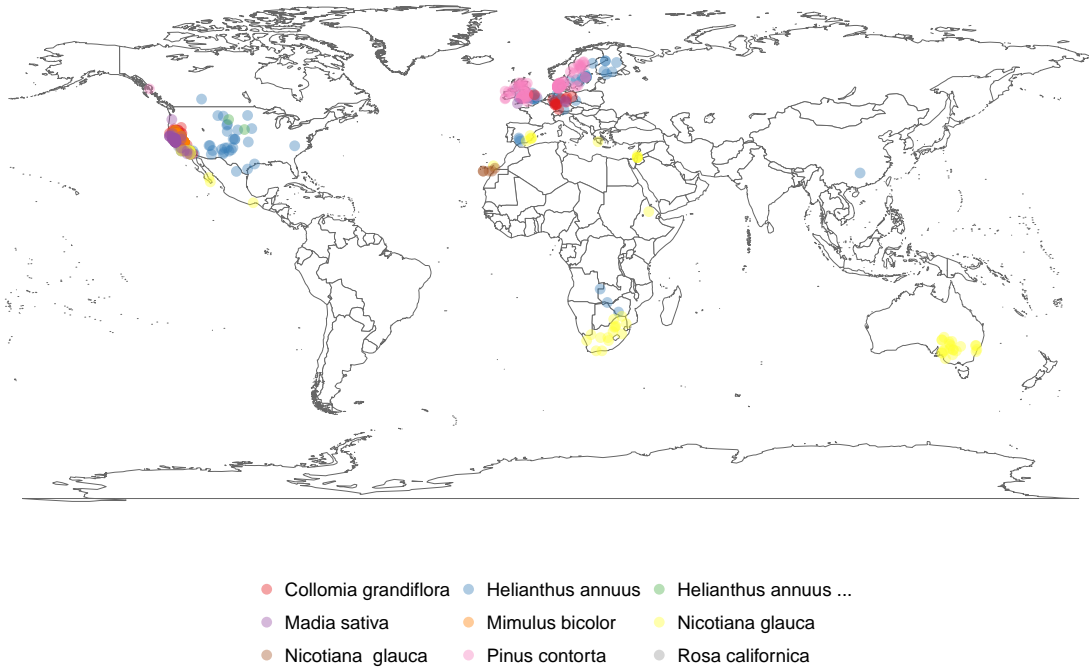


FIG. A.2. A map.