

Assignment 3

Sam Rosenblatt, Brendan Case

2/8/2019

First we load in the databases

```
require(tidyverse)

## Loading required package: tidyverse

## -- Attaching packages ----- tidyverse 1.2.1 --

## v ggplot2 3.1.0      v purrr   0.2.5
## v tibble  1.4.2      v dplyr  0.7.8
## v tidyr   0.8.2      v stringr 1.3.1
## v readr   1.3.1      v forcats 0.3.0

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
load("COMADRE_v.2.0.1.RData")
load("COMPADRE_v.4.0.1.RData")
```

Part A

Brendan and I are working on a project modelling contagion in populations of honey bee and bumble bee species where the contagion is transmitted through several flowers.

Initially we tried to use this as an opportunity to investigate the population models of the bees, however there are no models of bee populations in the database, or even any Hymenoptera populations at all

```
grep(compadre$metadata$Order, pattern = 'Hymenoptera')
```

```
## integer(0)
```

Thus, instead of working with the bee populations we decided to look at one of the species of flower that is a transmission vector between bumblebees and honeybees: The white clover, i.e. *Trifolium repens*

First we check that it exists in the database:

```
grep(compadre$metadata$SpeciesAccepted, pattern = 'Lotus corniculatus')
```

```
## [1] 3530 3531 3532 3533 3534 3535 3536 3537 3538 3539 3540 3541 3542 3543
## [15] 3544 3545 3546 3547
```

From the above chunk we see that indeed it exists in the database, now we would like to choose the model which is the most relevant to our research, which is mainly concerned with these populations in Vermont and the greater New England area. To see which dataset is most appropriate for our purposes we will look at some of the info for each dataset.

```
allDatasets = grep(compadre$metadata$SpeciesAccepted, pattern = 'Lotus corniculatus')

for (dataset in allDatasets){
  print(paste0("Dataset #: ", dataset))
  print(paste0("Continent: ", compadre$metadata$Continent[dataset]))
  print(paste0("Country: ", compadre$metadata$Country[dataset]))
}
```

```

print(paste0("Ecoregion: ", compadre$metadata$Ecoregion[dataset]))
print(paste0("Latitude: ", compadre$metadata$Lat[dataset]))
print(paste0("Longitude: ", compadre$metadata$Lon[dataset]))
print("-----")
}

```

```

## [1] "Dataset #: 3530"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3531"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3532"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3533"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3534"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3535"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3536"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"

```

```

## [1] "-----"
## [1] "Dataset #: 3537"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3538"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3539"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3540"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3541"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3542"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3543"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3544"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"

```

```
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3545"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3546"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
## [1] "Dataset #: 3547"
## [1] "Continent: N America"
## [1] "Country: USA"
## [1] "Ecoregion: NA"
## [1] "Latitude: NA"
## [1] "Longitude: NA"
## [1] "-----"
```

Whelp, that didnt narrow down our search at all. I guess we will check out the papers to see the locations of the species.

```
for (dataset in allDatasets){
  print(paste0("Dataset #: ", dataset))
  print(paste0("DOI: ", compadre$metadata$DOI.ISBN[dataset]))
  print("-----")
}
```

```
## [1] "Dataset #: 3530"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3531"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3532"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3533"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3534"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3535"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3536"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
```

```
## [1] "-----"
## [1] "Dataset #: 3537"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3538"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3539"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3540"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3541"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3542"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3543"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3544"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3545"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3546"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
## [1] "Dataset #: 3547"
## [1] "DOI: 10.1046/j.1469-8137.1999.00540.x"
## [1] "-----"
```

From this output we can see that apparently all these datasets came from one particular publication.

Upon reading the publication we find that the methods are quite complex, and only some of the matrices can be used for our purposes. To identify if any of the available ones are such matrices we gather some further info about them.

```
allDatasets = grep(compadre$metadata$SpeciesAccepted,pattern = 'Lotus corniculatus')

for (dataset in allDatasets){
  print(paste0("Dataset #: ", dataset))
  print(paste0("Matrix Composite: ", compadre$metadata$MatrixComposite[dataset]))
  print(paste0("Matrix Treatment: ", compadre$metadata$MatrixTreatment[dataset]))
  print(paste0("Matrix Start Year, End Year: ", compadre$metadata$MatrixStartYear[dataset], ", ", compadre$metadata$MatrixEndYear[dataset]))
  print(paste0("Matrix Start Season, End Season: ", compadre$metadata$MatrixStartSeason[dataset], ", ", compadre$metadata$MatrixEndSeason[dataset]))
  print(paste0("Matrix Start Month, End Month: ", compadre$metadata$MatrixStartMonth[dataset], ", ", compadre$metadata$MatrixEndMonth[dataset]))
  print(paste0("Matrix Split: ", compadre$metadata$MatrixSplit[dataset]))
  print(paste0("Matrix Criteria Ontogeny: ", compadre$metadata$MatrixCriteriaOntogeny[dataset]))
  print(paste0("Matrix Criteria Age: ", compadre$metadata$MatrixCriteriaAge[dataset]))
  print(paste0("Matrix Criteria Size: ", compadre$metadata$MatrixCriteriaSize[dataset]))
  print(paste0("Matrix Observation: ", compadre$metadata$MatrixPopulation[dataset]))
}
```

```

print(paste0("Matrix Class described by author: ", compadre$matrixClass$MatrixClassAuthor[dataset]))
print(paste0("Matrix Class standardized by compadre creators: ", compadre$matrixClass$MatrixClassOrganism[dataset]))
print(paste0("Matrix Class numerical representation: ", compadre$matrixClass$MatrixClassNumber[dataset]))
print("-----")
}

## [1] "Dataset #: 3530"
## [1] "Matrix Composite: Mean"
## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3531"
## [1] "Matrix Composite: Mean"
## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3532"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3533"
## [1] "Matrix Composite: Individual"

```

```

## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] " _____"
## [1] "Dataset #: 3534"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] " _____"
## [1] "Dataset #: 3535"
## [1] "Matrix Composite: Mean"
## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] " _____"
## [1] "Dataset #: 3536"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"

```

```

## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3537"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3538"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Unmanipulated"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3539"
## [1] "Matrix Composite: Mean"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3540"
## [1] "Matrix Composite: Mean"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"

```



```

## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] " _____"
## [1] "Dataset #: 3541"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] " _____"
## [1] "Dataset #: 3542"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] " _____"
## [1] "Dataset #: 3543"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "

```

```

## [1] "-----"
## [1] "Dataset #: 3544"
## [1] "Matrix Composite: Mean"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3545"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3546"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
## [1] "Dataset #: 3547"
## [1] "Matrix Composite: Individual"
## [1] "Matrix Treatment: Clipped"
## [1] "Matrix Start Year, End Year: NA, NA"
## [1] "Matrix Start Season, End Season: NA, NA"
## [1] "Matrix Start Month, End Month: NA, NA"
## [1] "Matrix Split: Divided"
## [1] "Matrix Criteria Ontogeny: Yes"

```

```
## [1] "Matrix Criteria Age: No"
## [1] "Matrix Criteria Size: No"
## [1] "Matrix Observation: NA"
## [1] "Matrix Class described by author: "
## [1] "Matrix Class standardized by compadre creators: "
## [1] "Matrix Class numerical representation: "
## [1] "-----"
```

The only useful criteria with documentation for these matrices are: MatrixComposite, MatrixTreatment, MatrixSplit, MatrixCriteriaOntogeny, MatrixCriteriaAge, and MatrixCriteriaSize. With this limited documentation, it is impossible to distinguish the correspondance between these matrices and the ones mentioned in the article.

In the journal article the only population matrix that matches the kind we covered in class would be what the authors call the “composite matrix”, as all the other matrices only cover a small part of the lifecycle. And because of the poor documentation, it is impossible to tell which available dataset contains that matrix, if any.

So instead we will try to do a similar process with another flower that both bumblebees and honeybees pollinate, the common dandelion, which according to Wikipedia, is named *Taraxacum officinale*, but also according to Wikipedia is synonymous with *Taraxacum campylodes*, which we found using the code below.

```
dandelions = grep(compadre$metadata$SpeciesAccepted, pattern = 'Taraxacum')
for (dandelion in dandelions){
  print(compadre$metadata$SpeciesAccepted[dandelion])
}
```

```
## [1] "Taraxacum erythrospermum"
## [1] "Taraxacum campylodes"
## [1] "Taraxacum campylodes"
## [1] "Taraxacum campylodes"
## [1] "Taraxacum campylodes"
## [1] "Taraxacum campylodes"
```

Now to differentiate between the datasets for *Taraxacum campylodes* we loop through them printing out the location and journal article which they come from

```
campylodeses = grep(compadre$metadata$SpeciesAccepted, pattern = 'Taraxacum campylodes')

for (campylodes in campylodeses){
  print(paste0("Dataset #: ", campylodes))
  print(paste0("DOI: ", compadre$metadata$DOI.ISBN[campylodes]))
  print(paste0("Country: ", compadre$metadata$Country[campylodes]))
  print(paste0("Latitude: ", compadre$metadata$Lat[campylodes]))
  print(paste0("Longitude: ", compadre$metadata$Lon[campylodes]))
  print("-----")
}
```

```
## [1] "Dataset #: 4978"
## [1] "DOI: 10.2307/2960501"
## [1] "Country: USA"
## [1] "Latitude: 39.6166666666667"
## [1] "Longitude: -78.05"
## [1] "-----"
## [1] "Dataset #: 4979"
## [1] "DOI: 10.2307/2960501"
## [1] "Country: USA"
## [1] "Latitude: 39.6166666666667"
```

```
## [1] "Longitude: -78.05"
## [1] "-----"
## [1] "Dataset #: 4980"
## [1] "DOI: 10.2307/2960501"
## [1] "Country: USA"
## [1] "Latitude: 39.6166666666667"
## [1] "Longitude: -78.05"
## [1] "-----"
## [1] "Dataset #: 4981"
## [1] "DOI: 10.2307/2960501"
## [1] "Country: USA"
## [1] "Latitude: 39.6166666666667"
## [1] "Longitude: -78.05"
## [1] "-----"
## [1] "Dataset #: 4982"
## [1] "DOI: 10.1890/12-1310.1"
## [1] "Country: USA"
## [1] "Latitude: 38.5"
## [1] "Longitude: -89.5"
## [1] "-----"
```

Reading the articles we find that the data in the 5th listed dataset is appropriate for our purposes for multiple reasons. With the ultimate goal in mind of using modelling approaches to inform management strategies to prevent the spread of disease between honey bees and bumblebees, these reasons include:

- Both bumble bees and honey bees pollinate the common dandelion, so it is a potential vector
- The common dandelion is one of the most recognizable flowers to the average person, thus if it were suggested by separate, associated research, that increased or decreased density of dandelions decreased the spread of diseases between bees, then public awareness campaigns to either encourage the weeding or the spreading of dandelions would be potentially useful management strategies
- The dandelion itself is an invasive species, so before any such campaigns as mentioned above were to be launched, a thorough understanding of how these populations grow would be necessary to avoid unwanted consequences

Part B

Recent research suggests that flowers pollinated by both honey bees and bumble bees is a transmission route between the two species. Additionally, out of several tested species of flowers, research has found that different flowers have different capacities for harboring certain diseases. With the goal of managing the spread of these disease in mind, we would like to be able to consider increasing or decreasing the density of different species of flower as a potential management strategy. Multiple considerations are involved in this intervention approach. While dandelions have yet to be tested for their efficacy as a disease vector, since it has been demonstrated that other species of flower have differential efficacies, we can assume that the dandelion is either less or greater than average in its spreading efficacy. The questions that we would want to answer with this matrix model depends upon the answer to that question.

If Dandelions are found to be unlikely to harbor the pathogen:

- Can we safely increase the dandelion population to decrease the pathogen without creating a runaway invasive species?
- One potential management strategy would be to launch a public awareness campaign to encourage people to spread and not kill dandelions in order to #SaveTheBees. An easy and potentially eye-catching and memorable advertising tactic might involve the image of a person blowing a dandelion [example

pictured]. However this tactic would only be effective if that stage in the lifecycle were important. So we would like to find out if that is true.

If Dandelions are found to be likely to harbor the pathogen:

- How low do we need to make the population density before it will collapse?
- Is there a particular stage of development that we should target to be most effective in eradicating the plant?

Part C

Eigenvalues and stable distribution

```
compadre$matrixClass[[4982]]

## # A tibble: 3 x 3
##   MatrixClassOrganized MatrixClassAuthor MatrixClassNumber
## * <chr>                <chr>                <dbl>
## 1 prop                  Seed                  1
## 2 active                Nonreproductive      2
## 3 active                Reproductive        3

m = compadre$mat[[4982]][[1]]
m

##           A1    A2          A3
## [1,] 0.8500 0.00 1041823.20
## [2,] 0.0007 0.38   46648.93
## [3,] 0.0000 0.01     0.01

m_eigen = eigen(m)
m_eigen

## eigen() decomposition
## $values
## [1] 21.8022116 -21.3965657  0.8343541
##
## $vectors
##           [,1]          [,2]          [,3]
## [1,] -9.990410e-01 -9.989569e-01 -1.000000e+00
## [2,] -4.378451e-02 -4.566282e-02  1.237995e-06
## [3,] -2.009181e-05  2.133122e-05  1.501776e-08
```

The (dominant) eigenvalue is 21.8022116, while the stable state distribution is given by the dominant right eigenvector, i.e., the first column of the `vectors` variable above. To make the distribution more readable, we can round and normalize them:

```
stab_dist = round(m_eigen$vectors[,1]/sum(m_eigen$vectors[,1]), 5)
stab_dist

## [1] 0.95800 0.04199 0.00002
```

Stability and elasticity

To compute the stability and elasticity, we will also need the left eigenvalue:

```
m_eigen_t = eigen(t(m))
repr_val = round(m_eigen_t$vectors[,1]/sum(m_eigen_t$vectors[,1]), 8)
repr_val
```

```
## [1] 0.00000002 0.00046659 0.99953340
```

Since it appears the easiest way to compute the elasticity for a Leslie matrix is to compute the sensitivity first, we do this first.

$$S = v \cdot w^T,$$

where v is the reproductive value and w is the stable age distribution.

```
m_sen = repr_val %*% t(stab_dist)
m_sen
```

```
##           [,1]      [,2]      [,3]
## [1,] 1.916000e-08 8.398000e-10 4.000000e-13
## [2,] 4.469932e-04 1.959211e-05 9.331800e-09
## [3,] 9.575530e-01 4.197041e-02 1.999067e-05
```

This tells us that if we were to change the values in each term of m by a fixed small amount one at a time, we would see the greatest change in λ at index (1,3), that is, the transition value for going from a seed to a reproductive adult.

Now, the elasticity follows directly.

$$E = \frac{1}{\lambda} \cdot S \circ M,$$

where \circ denotes the Hadamard product and M is the Leslie matrix.

```
m_elas = m_sen * m / m_eigen[[1]]
m_elas
```

```
##           A1           A2           A3
## [1,] 7.469884e-10 0.000000e+00 1.911408e-08
## [2,] -1.462362e-08 -3.479532e-07 -2.034525e-05
## [3,] 0.000000e+00 5.030287e-04 2.395945e-07
```

This tells us that a small change in the nonreproductive to reproductive transition value will cause the most change in λ , *proportional* to its original value.

The validity of our S and E can easily be checked by simply varying M in the ways described above.

```
# test the sensitivity
x = matrix(nrow=3, ncol=3)
for (i in 1:nrow(m)) {
  for (j in 1:ncol(m)) {
    tmp = m[i, j]
    m[i, j] = tmp + .05
    x[i, j] = eigen(m)[[1]][1]
    m[i, j] = tmp
  }
}
x # the eigenval at (1, 3) should be largest
```

```
##           [,1]      [,2]      [,3]
## [1,]  21.80223 21.80221 21.80221
## [2,]  22.35544 21.82743 21.80222
## [3,] 229.68292 53.10821 21.82700

# test the elasticity
for (i in 1:nrow(m)) {
  for (j in 1:ncol(m)) {
    tmp = m[i, j]
    m[i, j] = tmp * 1.05
    x[i, j] = eigen(m)[[1]][1]
    m[i, j] = tmp
  }
}
x # the eigenval at (2, 3) should be largest
```

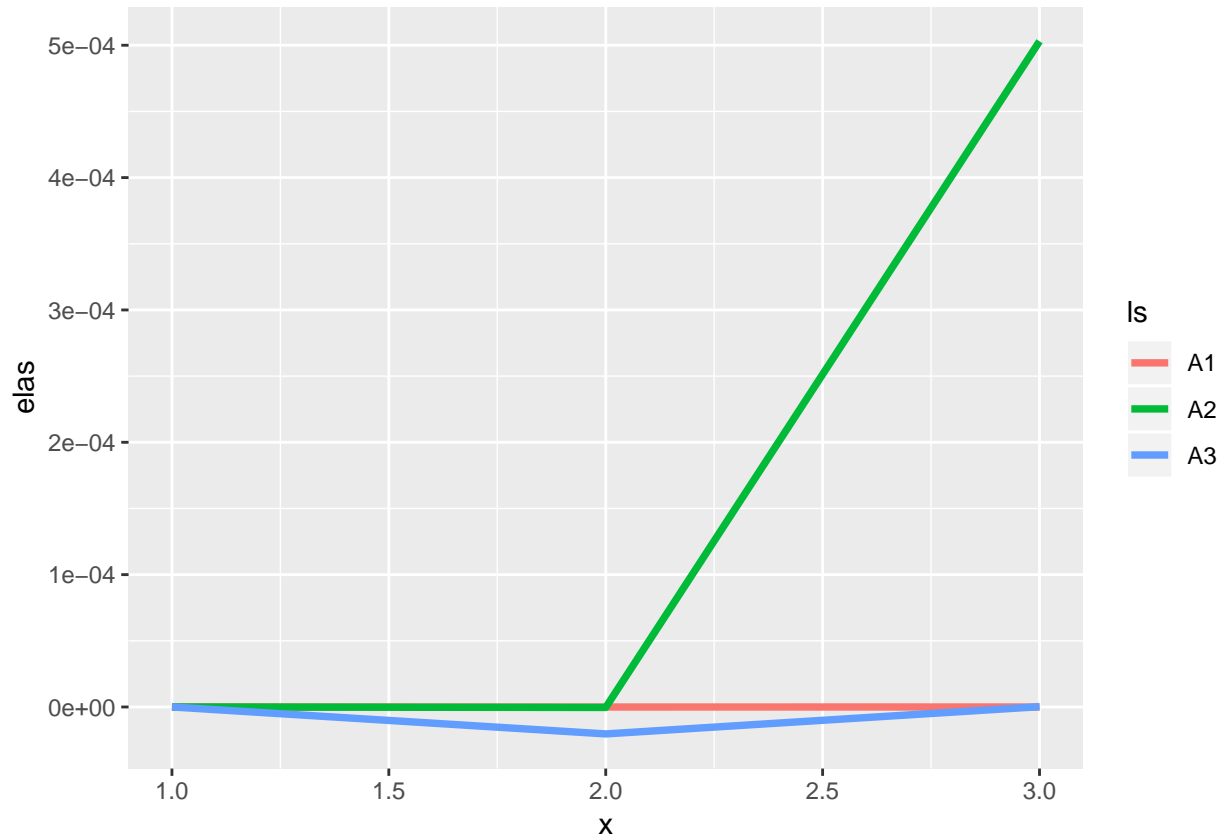
```
##           [,1]      [,2]      [,3]
## [1,] 21.80223 21.80221 21.80261
## [2,] 21.80261 21.81179 22.33518
## [3,] 21.80221 22.33556 21.80246
```

Part D

One of the nice things about the elasticity is that 0 values in the Leslie matrix will have the smallest elasticity. In terms of informing management strategy, this is useful because generally a 0 value indicates a biologically impossible event. Therefore, to assess what sort of management will make the most impact, it makes sense to consider elasticity over sensitivity.

Let's examine the elasticity matrix in more detail. The following plot separates each life stage in to a separate line. On the x-axis is the "to" part of the transition matrix, for the given "from" part, and the y-axis gives the elasticity.

```
m_elas %>%
  data.frame(x=1:nrow(m)) %>%
  gather(ls, elas, -x) %>%
  ggplot(aes(x=x, y=elas, col=ls)) + geom_line(size=1.3)
```



We see that changing the transition value for going from unreprouductive to reprouductive adults will have the greatest effect on population growth. Therefore, if we wanted to, for example, decrease the value of λ closer to 0 in order to subdue this population, we would best focus our efforts on limiting the number of nonreprouductive adults which transition to reprouductive status (through chemicals or something?), rather than running around and plucking a bunch of reprouductive adults out of the ground.