BIO 423 - Lab 1

Benjamin Blonder

Spring 2019

Learning outcomes

R goals:

- map data using shapefile base maps
- clean datasets by assessing column types, outlier values, and transcription errors

Content goals:

- test luxury hypothesis for plant biodiversity in urban environments
- assess species-area relationships in urban systems

Data cleaning

Before we can start working with our class dataset, we need to read in and clean the data. It is very likely that there are various errors in the data that will need to be fixed before we can use it for quantitative analyses. The best way to do this is to:

1. Download the data from Google Drive as `CSV` format (comma-separated values) and put it in the same folder as your working directory and your analysis script. You can do this from the menu item `File > Download As > Comma separated values`.
2. Load the data into R (instructions below).
3. Check for errors using R (instructions below).
4. Fix any errors in the Google Drive version.
5. Return to step 1 until no errors remain.

Load in the data

First, let's read in the data:

```
# the below line assumes that you have downloaded a copy of the data!
data = read.csv("BIO 423 Spring 2019 Field Data - Sheet1.csv")

# check out column names
names(data)
```

```
##  [1] "Group.members"
##  [2] "Property.address"
##  [3] "Property.value.Zestimate...."
##  [4] "Property.lot.size..site.area...ft2."
##  [5] "Latitude..decimal.degrees."
##  [6] "Longitude..decimal.degrees."
##  [7] "Yard.type..xeriscaped.1.lawn.5."
##  [8] "Number.of.species..trees."
##  [9] "Number.of.species..shrubs."
## [10] "Number.of.species..succulents."
```

Column class errors

The first thing we need to do is check for errors in the column classes. All of the columns besides `Group.members` and `Property.address` should have column class either `'int'` or `'num'` reflecting that they are either integers (e.g. the yard type) or numeric (e.g. property lot size). You can see the column type to the right of the column name using the `str` function.

```
str(data)
```

```
## 'data.frame':    241 obs. of  10 variables:
##  $ Group.members                     : Factor w/ 15 levels "Aidan, Joseph, Noah, Sydney",..: 1 1 1 1
##  $ Property.address                  : Factor w/ 241 levels "1015 S Ash Ave., Tempe, AZ, 85281",..:
##  $ Property.value.Zestimate....      : Factor w/ 241 levels "$1,812,372","$265,716",..: 223 239 63 9
##  $ Property.lot.size..site.area...ft2.: Factor w/ 212 levels "1,080","1,289",..: 36 42 121 160 129 28
##  $ Latitude..decimal.degrees.        : Factor w/ 215 levels "33.21312","33.213589",..: 22 26 25 24 1
##  $ Longitude..decimal.degrees.       : Factor w/ 230 levels "-111.616707",..: 111 112 93 91 114 121
##  $ Yard.type..xeriscaped.1.lawn.5.   : int  2 2 0 4 1 5 3 5 5 5 ...
##  $ Number.of.species..trees.         : int  3 2 0 2 1 4 5 6 3 2 ...
##  $ Number.of.species..shrubs.        : int  7 7 0 0 2 1 5 13 2 3 ...
##  $ Number.of.species..succulents.    : int  5 2 0 1 2 2 0 11 2 2 ...
```

If you see any data with the wrong class, your next step would be to investigate why. Common errors include: - including non-numeric values (e.g. marginal notes like `"34 (wrong)"`) - adding extra columns or dots (e.g. `"33.19.2"` instead of `33.192`) - adding extra spaces (e.g. `"33.2 "` instead of `33.2`) - adding extra commas (e.g. `"9,342"` instead of `9342`) - adding extra symbols (e.g `"$340"` instead of `340`)

You may also need to force Google Sheets to format columns as numeric, rather than as text.

Range errors

Once you have resolved the column class issues, the next thing to check is that the data have a reasonable range of values. For example, we know that measurements of body mass must be non-negative, and we can imagine that if those body mass estimates were for insects, values over 1 kilogram are probably erroneous. These errors are easy to check for in R using functions you already know:

The easiest way is to simply run a summary and look for the min/mean/max values - then decide if they are realistic:

```
summary(data)
```

```
##                          Group.members
##  Jennifer, Austin, Makena, Edgar: 21
##  Aidan, Joseph, Noah, Sydney    : 20
##  Andy, Teddy, Haleigh           : 20
##  Annie, Aaron, Mickey           : 20
##  Bradley, Allison, Ryan         : 20
##  Deidra, Eli, Morgan            : 20
##  (Other)                        :120
##                                  Property.address
##  1015 S Ash Ave., Tempe, AZ, 85281         :  1
##  102 E Hayward Ave., Phoenix, Arizona 85020:  1
##  1021 E Concorda Drive, Tempe, AZ          :  1
##  1021 S Ash Ave., Tempe, AZ, 85281         :  1
##  1033 E Concorda Drive, Tempe, AZ          :  1
##  1051 E Concorda Drive, Tempe, AZ          :  1
##  (Other)                                   :235
```

```
##  Property.value.Zestimate.... Property.lot.size..site.area...ft2.
##  $1,812,372:  1               1,080  :  4
##  $265,716  :  1               36,155 :  4
##  1,040,509 :  1               43,560 :  4
##  1,088,416 :  1               8,050  :  4
##  1,088,775 :  1               36154.8:  3
##  1,150,000 :  1               38,333 :  3
##  (Other)   :235               (Other):219
##  Latitude..decimal.degrees. Longitude..decimal.degrees.
##  33.2955   :  6             -111.9422118:  3
##  33.408908 :  5             -112.3202142:  3
##  33.408665 :  4             -111.8864   :  2
##  33.41816  :  4             -111.90335  :  2
##  33.21359  :  3             -111.915703 :  2
##  33.4591847:  3             -111.92289  :  2
##  (Other)   :216             (Other)     :227
##  Yard.type..xeriscaped.1.lawn.5. Number.of.species..trees.
##  Min.   :0.000                   Min.   : 0.000
##  1st Qu.:1.000                   1st Qu.: 1.000
##  Median :3.000                   Median : 2.000
##  Mean   :2.892                   Mean   : 2.353
##  3rd Qu.:5.000                   3rd Qu.: 3.000
##  Max.   :5.000                   Max.   :16.000
##
##  Number.of.species..shrubs. Number.of.species..succulents.
##  Min.   : 0.000             Min.   : 0.000
##  1st Qu.: 2.000             1st Qu.: 0.000
##  Median : 3.000             Median : 0.000
##  Mean   : 3.768             Mean   : 1.718
##  3rd Qu.: 5.000             3rd Qu.: 2.000
##  Max.   :15.000             Max.   :62.000
##
```

Once you have identified columns where there are likely errors, you need to determine which rows are the problematic ones. We can do this using the `which` functionality for subsetting that you already know:

```
# we know that the yard types should range from 1 to 5 -
# find out of range values
data[which(data$Yard.type..xeriscaped.1.lawn.5. < 1),]
```

```
##                 Group.members     Property.address
## 3 Aidan, Joseph, Noah, Sydney 3815 W Butler Street
##   Property.value.Zestimate.... Property.lot.size..site.area...ft2.
## 3                       224524                                6098
##   Latitude..decimal.degrees. Longitude..decimal.degrees.
## 3                  33.301029                 -111.906677
##   Yard.type..xeriscaped.1.lawn.5. Number.of.species..trees.
## 3                               0                         0
##   Number.of.species..shrubs. Number.of.species..succulents.
## 3                          0                              0
```

```
data[which(data$Yard.type..xeriscaped.1.lawn.5. > 5),]
```

```
##  [1] Group.members
##  [2] Property.address
##  [3] Property.value.Zestimate....
```

```
##  [4] Property.lot.size..site.area...ft2.
##  [5] Latitude..decimal.degrees.
##  [6] Longitude..decimal.degrees.
##  [7] Yard.type..xeriscaped.1.lawn.5.
##  [8] Number.of.species..trees.
##  [9] Number.of.species..shrubs.
## [10] Number.of.species..succulents.
## <0 rows> (or 0-length row.names)
```

Any results that give non-empty results represent errors you need to fix on Google Drive. You should check for errors not just in the xeriscaped column, but also in all the others. For example, you might want to ensure that the latitude and longitude values are reasonable (check especially for sign errors - longitudes in Phoenix should be somewhere around -111°, not 111°). I recommend that we divide this work up among groups in the class, as everyone will ultimately use the same cleaned dataset together.

Working with the clean data

Once you have clean data, you can begin to make some sceintific analyses. To help illustrate this, let's work with a 'fake' version of the data - a subset of the whole dataset that I have cleaned for you. You don't need to use these data yourself, but you can look at this code to help you with your analyses for the class.

```
data_fake = read.csv('clean data subset.csv')
str(data_fake)
```

```
## 'data.frame':    4 obs. of  10 variables:
##  $ Group.members                      : Factor w/ 4 levels "Aidan, Joseph, Noah, Sydney",..: 1 3 4 2
##  $ Property.address                   : Factor w/ 4 levels "1315 E Orange St, Tempe, AZ 85281",..: 4
##  $ Property.value.Zestimate....       : int  540757 386489 287332 261511
##  $ Property.lot.size..site.area...ft2.: int  19166 43560 8050 6347
##  $ Latitude..decimal.degrees.         : num  33.3 33.2 33.3 33.4
##  $ Longitude..decimal.degrees.        : num  -112 -112 -112 -112
##  $ Yard.type..xeriscaped.1.lawn.5.    : int  2 5 5 5
##  $ Number.of.species..trees.          : int  3 1 3 2
##  $ Number.of.species..shrubs.         : int  7 0 4 0
##  $ Number.of.species..succulents.     : int  5 0 1 0
```

First, let's load in some libraries you will find helpful. If these don't load, make sure to install them.

```
library(raster)
```

```
## Loading required package: sp
```

```
library(rgeos)
```

```
## rgeos version: 0.4-2, (SVN revision 581)
##  GEOS runtime version: 3.6.1-CAPI-1.10.1
##  Linking to sp version: 1.3-1
##  Polygon checking: TRUE
```

```
library(rgdal)
```

```
## rgdal: version: 1.3-6, (SVN revision 773)
##  Geospatial Data Abstraction Library extensions to R successfully loaded
##  Loaded GDAL runtime: GDAL 2.1.3, released 2017/20/01
##  Path to GDAL shared files: /Library/Frameworks/R.framework/Versions/3.5/Resources/library/rgdal/gdal
##  GDAL binary built with GEOS: FALSE
##  Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ_VERSION: 493]
```

```
##  Path to PROJ.4 shared files: /Library/Frameworks/R.framework/Versions/3.5/Resources/library/rgdal/pr
##  Linking to sp version: 1.3-1
```

```
library(maptools)
```

```
## Checking rgeos availability: TRUE
```

```
library(ggplot2)
library(RColorBrewer)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:rgeos':
##
##     intersect, setdiff, union
```

```
## The following objects are masked from 'package:raster':
##
##     intersect, select, union
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Second, let's load in some spatial data showing the layout of cities in Phoenix. You should download the `tl_2016_04_place.zip` file and extract it into a folder called `tl_2016_04_place` in the same directory as this R script (i.e. your working directory).

```
# load in census data for places and roads
places <- shapefile('tl_2016_04_place/tl_2016_04_place.shp')
places@data$id = rownames(places@data)
places.points = fortify(places, region="id")
places.df = inner_join(places.points, places@data, by="id")
# add or delete place-names as desired below to change map extent
places.df = places.df[places.df$NAME %in%
              c("Phoenix","Tempe","Gilbert",
                "Chandler","Mesa","Guadalupe",
                "Scottsdale","Fountain Hills",
                "Paradise Valley"),]
```

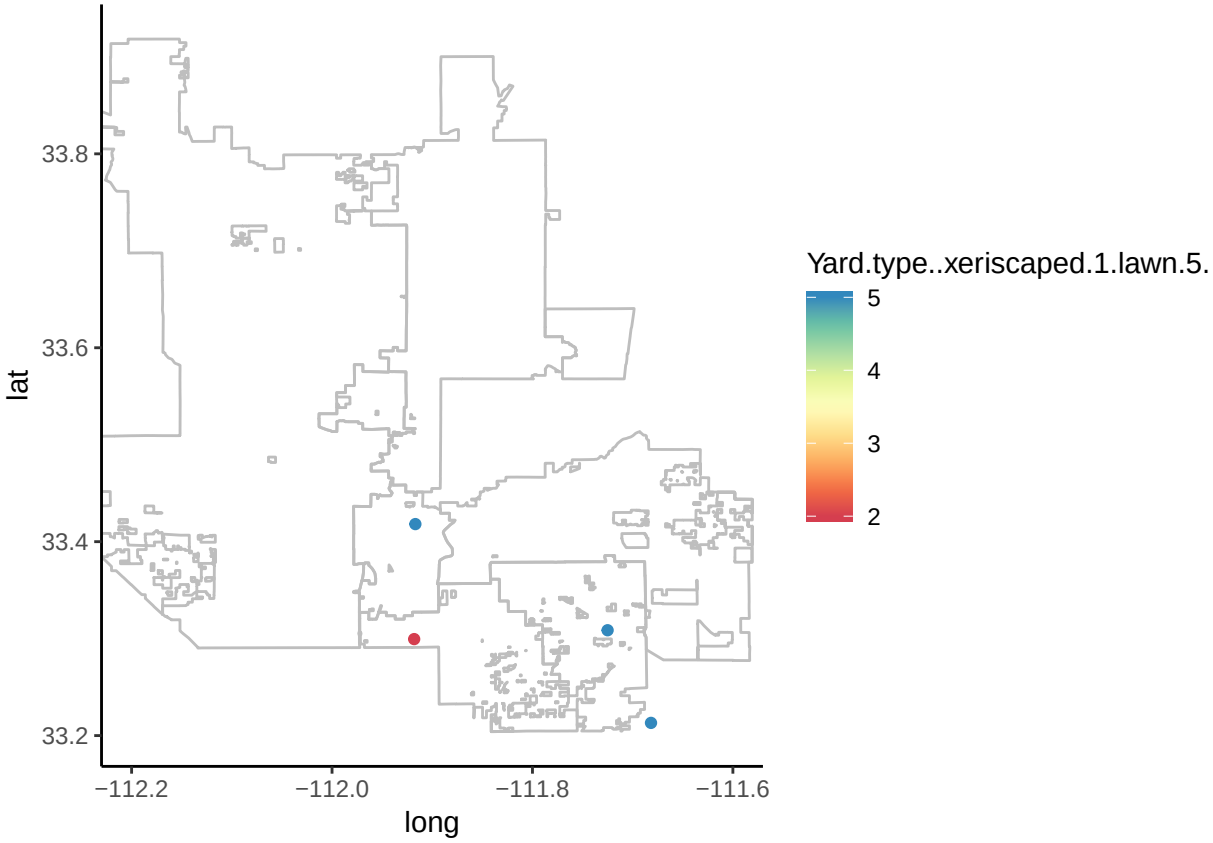
Now we can make a map illustrating certain data columns by coloring points.

```
# set up a base map
map_base = ggplot(places.df) +
  theme_classic() +
  aes(long,lat,group=group) +
  geom_polygon(fill=NA,col='gray') +
  coord_cartesian(xlim=c(-112.2,-111.6)) # you can change these values, or add ylim= to zoom in

# make a map including the base map and the layer we care about (say, yard type)
map_yardtype = map_base +
    geom_point(data= data_fake, # change this to your real data eventually!
               mapping=aes(
```

```
                    x=Longitude..decimal.degrees.,
                    y=Latitude..decimal.degrees.,
                    group=NULL,
                    col= Yard.type..xeriscaped.1.lawn.5.)) + # change to your variable eventually!
    scale_color_gradientn(colors = brewer.pal(9,"Spectral"))

# plot the map
map_yardtype
```



Here also is an example multiple regression. To make a model of z $x + y$ where z is the dependent variable, and x and y are two independent variables, you can write `lm(z~x+y,data=mydataframe)`

```
model_yardtype_propertyvalue_propertysize  =
  lm(Yard.type..xeriscaped.1.lawn.5. ~
       Property.value.Zestimate.... + Property.lot.size..site.area...ft2.,
     data=data_fake)

summary(model_yardtype_propertyvalue_propertysize)
```

```
##
## Call:
## lm(formula = Yard.type..xeriscaped.1.lawn.5. ~ Property.value.Zestimate.... +
##     Property.lot.size..site.area...ft2., data = data_fake)
##
## Residuals:
##         1         2         3         4
## -0.011974 -0.002314  0.140693 -0.126405
##
```

```
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       8.271e+00  3.331e-01   24.83   0.0256
## Property.value.Zestimate....     -1.300e-05  9.513e-07  -13.67   0.0465
## Property.lot.size..site.area...ft2.  4.034e-05  7.016e-06    5.75   0.1096
##
## (Intercept)                       *
## Property.value.Zestimate....      *
## Property.lot.size..site.area...ft2.
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1895 on 1 degrees of freedom
## Multiple R-squared:  0.9947, Adjusted R-squared:  0.984
## F-statistic: 93.45 on 2 and 1 DF,  p-value: 0.07295
```

```
confint(model_yardtype_propertyvalue_propertysize)
```

```
##                                         2.5 %        97.5 %
## (Intercept)                      4.039016e+00  1.250356e+01
## Property.value.Zestimate....    -2.509174e-05 -9.181070e-07
## Property.lot.size..site.area...ft2. -4.880595e-05  1.294895e-04
```

```
# here in this demo the overall model p-value is 0.07, so the result is not significant
# and the explained variation is high (0.99) primarily due the low number of data points
# and the estimate fo the independent effect of property value is negative
# and the estimate for the independent effect of property lot size is positive.
```

What to do next

Analyze the real clean data from the class to answer the following questions (same as in syllabus):

1. What is the distribution of alpha diversity (species richness) across all sites? What about for each of trees, shrubs, and succulents? Summarize with mean, standard deviation, minimum, and maximum. (Hint: you need to make a new column for total species richness)
2. How does species richness (for all species types together) change with property size? Construct a species-area relationship (SAR). You can try log-transforming x- or y- axes depending on the model you think is relevant. Report regression coefficients (with 95% confidence intervals), model R^2, and model p-value. (Hint: use `lm`, then use `confint` and `summary`)
3. Does species richness increase with increasing property value? How does yard type influence species richness? Report regression coefficients (with 95% confidence intervals), model R^2, and model p-value. (Hint: use `lm` again)
4. Is the hypothesis that plant diversity is higher where property values are high supported by your data? Over what spatial and temporal scales is your conclusion likely to be valid?
5. Are there aspects of the data collection or analysis procedure that could have biased or influenced your conclusion? For example, the sampling methods, the species naming methods, the statistical approaches used, etc.
6. What other factors might influence species diversity in urban environments? You can speculate based on your field experience, or cite evidence from the primary literature you read (either papers I provided, or that you found on your own).
7. How do your findings compare to the results reported in the published studies you read as background for this project?
8. Based on your findings, do you personally think that urban biodiversity is equitably distributed among rich and poor people in the Phoenix area?

Write a report with short (1-paragraph) answers to each of the above questions, summarizing your findings.The final report should be single-spaced, 12 pt font, 1" margins. It will probably require approximately 3 pages but may be longer if figures/graphs are included. Graduate students may include more sophisticated analyses, e.g. spatial analyses, testing for interactions between predictor variables or pairing data to other public socioeconomic datasets. The R script you used should be included, pasted as an appendix on the last page of the report. As before, you should also include an author contribution statement.

Optional questions for graduate students

- Is there a significant interaction between property size and property value, or with yard type, in models of species richness? How strong is it, and in which direction does it operate?
- What SAR model best fits the data - power law, logistic, log, etc? (use `AIC` to compare different models in the `mmSAR` package)
- Do results vary between different cities in the Phoenix metro area? (You will need to use `%over%` to classify points into city polygons, or you can do this manually in the datasheet)
- Are the property values obtained for this dataset representative of the overall income levels in the region? You can download median household income by census tract data from

`https://census.gov/data/data-tools.html`

and shapefiles from

`https://www.census.gov/geo/maps-data/data/cbf/cbf_tracts.html`

You will probably have to classify points into census tracts using `%over%` and then use `inner_join` to merge against income data.