Assignment 6: Generalized Linear Models

Sarah Ko

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics (ENV872L) on generalized linear models.

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Use the lesson as a guide. It contains code that can be modified to complete the assignment.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Be sure to **answer the questions** in this assignment document. Space for your answers is provided in this document and is indicated by the ">" character. If you need a second paragraph be sure to start the first line with ">". You should notice that the answer is highlighted in green by RStudio.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file. You will need to have the correct software installed to do this (see Software Installation Guide) Press the `Knit` button in the RStudio scripting panel. This will save the PDF output in your Assignments folder.
6. After Knitting, please submit the completed exercise (PDF file) to the dropbox in Sakai. Please add your last name into the file name (e.g., "Salk_A06_GLMs.pdf") prior to submission.

The completed exercise is due on Tuesday, 26 February, 2019 before class begins.

Set up your session

1. Set up your session. Upload the EPA Ecotox dataset for Neonicotinoids and the NTL-LTER raw data file for chemistry/physics.

2. Build a ggplot theme and set it as your default theme.

```
#1
#load packages
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.5.2
```

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

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

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

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.5.2
```

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

```
## v ggplot2 3.1.0     v readr   1.3.1
## v tibble  2.0.1     v purrr   0.3.0
## v tidyr   0.8.2     v stringr 1.3.1
```

```
## v ggplot2 3.1.0     v forcats 0.3.0
## Warning: package 'ggplot2' was built under R version 3.5.2
## Warning: package 'tibble' was built under R version 3.5.2
## Warning: package 'tidyr' was built under R version 3.5.2
## Warning: package 'readr' was built under R version 3.5.2
## Warning: package 'purrr' was built under R version 3.5.2
## -- Conflicts ---------------------------------------------------------------- tidyverse_conflicts()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidyr)
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 3.5.2
## Loading required package: magrittr
##
## Attaching package: 'magrittr'
## The following object is masked from 'package:purrr':
##
##     set_names
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
# get working directory
getwd()
```

```
## [1] "C:/Users/Sarah/Documents/Duke/Year 2/Spring 2019/Data Analytics/Environmental_Data_Analytics"
```

```
# set wd to the filepath of Environmental_Data_Analytics to use relative filepath

NTL.ChemPhys <- read.csv("./Data/Raw/NTL-LTER_Lake_ChemistryPhysics_Raw.csv")

Neonicotinoids.Raw <- read.csv("./Data/Raw/ECOTOX_Neonicotinoids_Mortality_raw.csv")

# check the class of the date columns

class(NTL.ChemPhys$sampledate)
```

```
## [1] "factor"
```

```
class(Neonicotinoids.Raw$Pub..Year)
```

```
## [1] "integer"
```

```
# change class date. must define original format of data
NTL.ChemPhys$sampledate <- as.Date(NTL.ChemPhys$sampledate, format = "%m/%d/%y")

# the Neonicotinoid column Pub..Year is already an integer, and does not have a month/day associated wi

# confirm class of the date column

class(NTL.ChemPhys$sampledate)
```

```
## [1] "Date"
```

```
#2

SKotheme <- theme_gray(base_size = 15) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "right",
        plot.title = element_text(hjust = 0.5))

theme_set(SKotheme)
```

Neonicotinoids test

Research question: Were studies on various neonicotinoid chemicals conducted in different years?

3. Generate a line of code to determine how many different chemicals are listed in the Chemical.Name column.
4. Are the publication years associated with each chemical well-approximated by a normal distribution? Run the appropriate test and also generate a frequency polygon to illustrate the distribution of counts for each year, divided by chemical name. Bonus points if you can generate the results of your test from a pipe function. No need to make this graph pretty.
5. Is there equal variance among the publication years for each chemical? Hint: var.test is not the correct function.

```
#3

# check class of Chemical.Name
class(Neonicotinoids.Raw$Chemical.Name)
```

```
## [1] "factor"
```

```
# count # of different factor levels
length(levels(Neonicotinoids.Raw$Chemical.Name))
```

```
## [1] 9
```

```
#4

year.analysis.neonico <- Neonicotinoids.Raw %>%
  group_by(Chemical.Name) %>%
  summarise(statistic = shapiro.test(Pub..Year)$statistic,
            p.value = shapiro.test(Pub..Year)$p.value)
print(year.analysis.neonico)
```

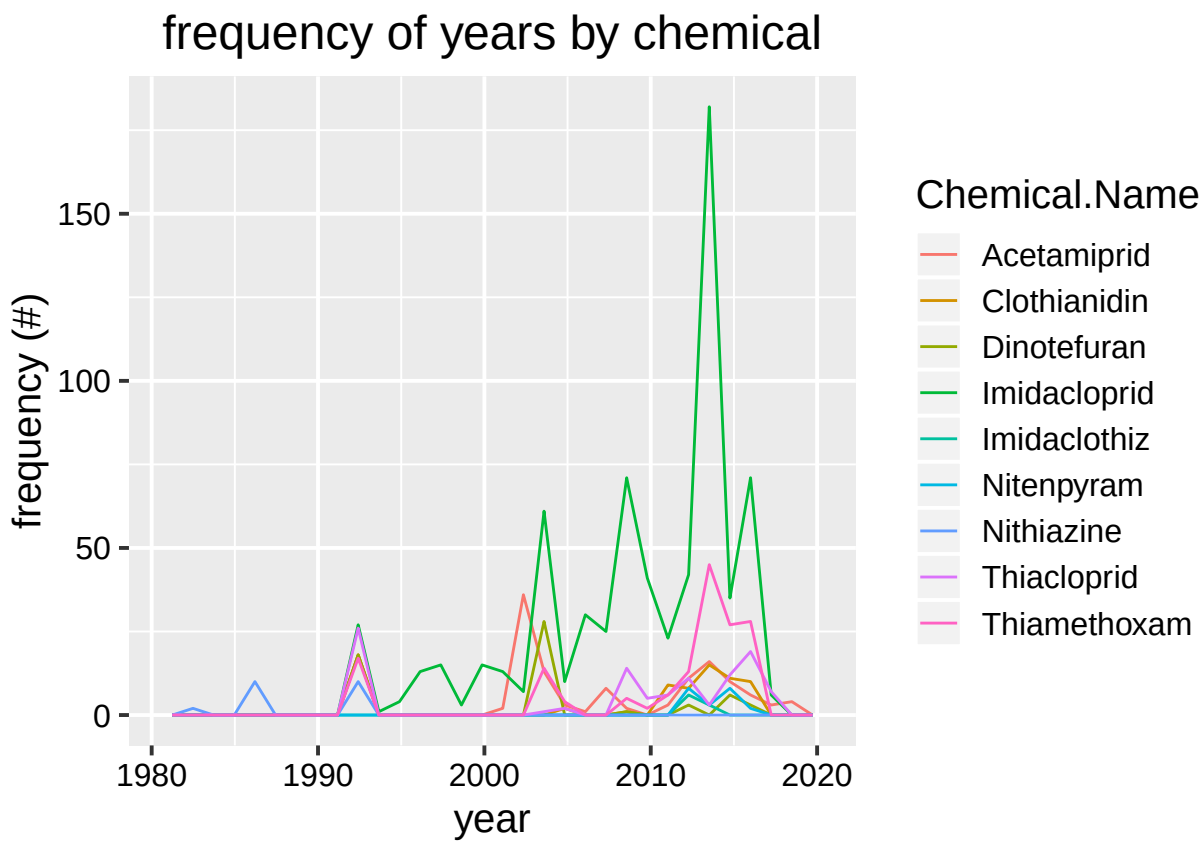
```
## # A tibble: 9 x 3
##   Chemical.Name statistic  p.value
##   <fct>             <dbl>    <dbl>
## 1 Acetamiprid       0.902 5.71e- 8
## 2 Clothianidin      0.696 4.29e-11
## 3 Dinotefuran       0.828 8.83e- 7
## 4 Imidacloprid      0.882 1.38e-22
## 5 Imidaclothiz      0.684 9.30e- 4
## 6 Nitenpyram        0.796 5.69e- 4
## 7 Nithiazine        0.759 1.24e- 4
## 8 Thiacloprid       0.767 1.12e-11
```

```
## 9 Thiamethoxam      0.707 1.57e-16
```

```
# the p values for each of the 9 chemicals is < 0.001, therefore the null hypotheses (H0 = data follows

# plot
year.plot.neonico <- ggplot(Neonicotinoids.Raw) +
  geom_freqpoly(aes(x = Pub..Year, color = Chemical.Name)) +
  xlab("year") +
  ylab("frequency (#)") +
  ggtitle("frequency of years by chemical") +
  theme(legend.position = "right")
print(year.plot.neonico)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#5

bartlett.test(Neonicotinoids.Raw$Pub..Year ~ Neonicotinoids.Raw$Chemical.Name)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Neonicotinoids.Raw$Pub..Year by Neonicotinoids.Raw$Chemical.Name
## Bartlett's K-squared = 139.59, df = 8, p-value < 2.2e-16
```

```
# Test results: K^2 = 139.59, df = 8, p < 0.001
# Since p < 0.05, this is evidence that the variance in frequency of years is significantly different f
```

6. Based on your results, which test would you choose to run to answer your research question?

   ANSWER: A one-way anova is appropriate because there is 1 categorical explanatory variable (chemical) with more than 2 categories (there are 9 chemicals), and a continous response (count of publications).

7. Run this test below.

8. Generate a boxplot representing the range of publication years for each chemical. Adjust your graph to make it pretty.

```
#7

pub.year.anova <- lm(Neonicotinoids.Raw$Pub..Year ~ Neonicotinoids.Raw$Chemical.Name)
summary(pub.year.anova)
```

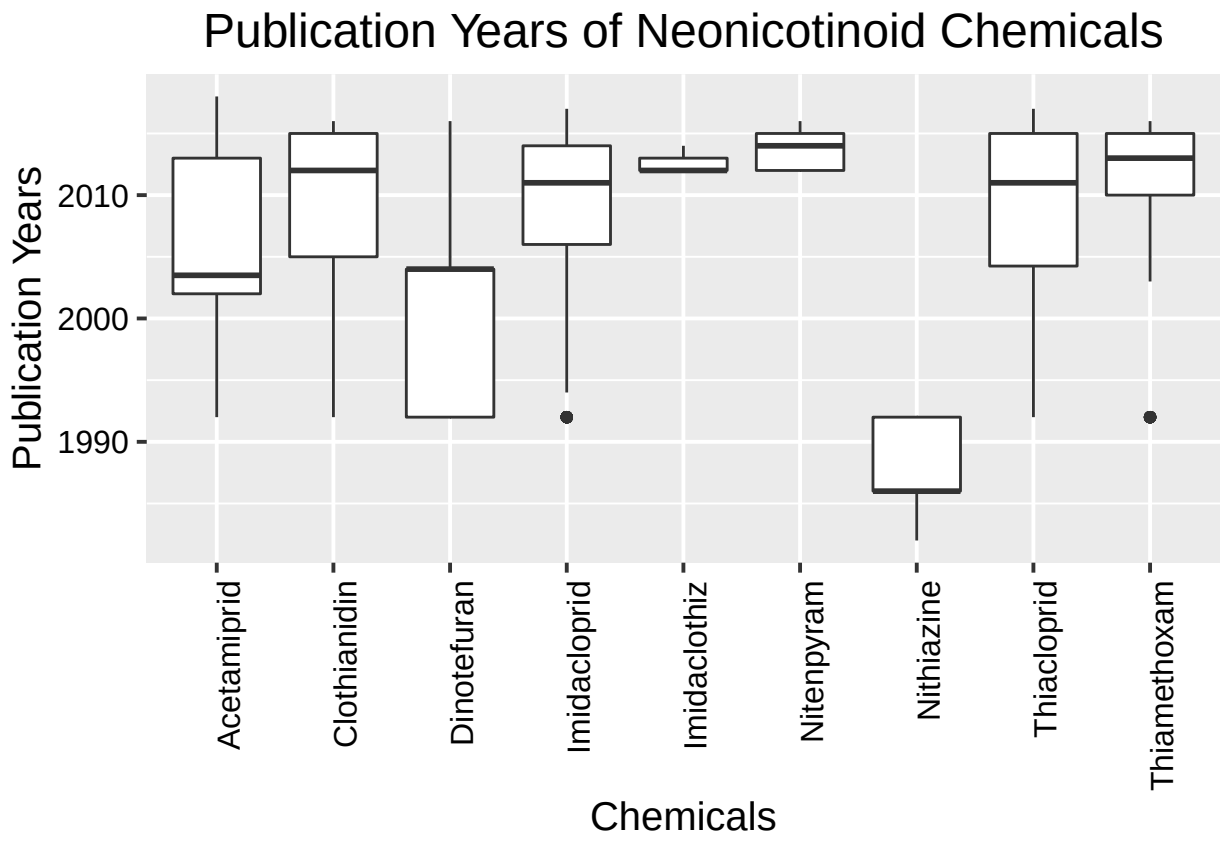
```
##
## Call:
## lm(formula = Neonicotinoids.Raw$Pub..Year ~ Neonicotinoids.Raw$Chemical.Name)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.366  -3.993   1.889   4.889  13.441
##
## Coefficients:
##                                            Estimate Std. Error  t value
## (Intercept)                                2005.9926     0.6082 3298.222
## Neonicotinoids.Raw$Chemical.NameClothianidin    2.0479     1.0246    1.999
## Neonicotinoids.Raw$Chemical.NameDinotefuran    -3.4333     1.1057   -3.105
## Neonicotinoids.Raw$Chemical.NameImidacloprid    3.1181     0.6651    4.689
## Neonicotinoids.Raw$Chemical.NameImidaclothiz    6.4518     2.4412    2.643
## Neonicotinoids.Raw$Chemical.NameNitenpyram      7.7216     1.6630    4.643
## Neonicotinoids.Raw$Chemical.NameNithiazine    -17.6290     1.6299  -10.816
## Neonicotinoids.Raw$Chemical.NameThiacloprid     1.6394     0.9190    1.784
## Neonicotinoids.Raw$Chemical.NameThiamethoxam    4.3738     0.8261    5.295
##                                              Pr(>|t|)
## (Intercept)                                   < 2e-16 ***
## Neonicotinoids.Raw$Chemical.NameClothianidin  0.04584 *
## Neonicotinoids.Raw$Chemical.NameDinotefuran   0.00194 **
## Neonicotinoids.Raw$Chemical.NameImidacloprid 3.05e-06 ***
## Neonicotinoids.Raw$Chemical.NameImidaclothiz  0.00832 **
## Neonicotinoids.Raw$Chemical.NameNitenpyram   3.78e-06 ***
## Neonicotinoids.Raw$Chemical.NameNithiazine    < 2e-16 ***
## Neonicotinoids.Raw$Chemical.NameThiacloprid   0.07467 .
## Neonicotinoids.Raw$Chemical.NameThiamethoxam 1.40e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.093 on 1274 degrees of freedom
## Multiple R-squared:  0.1726, Adjusted R-squared:  0.1674
## F-statistic: 33.21 on 8 and 1274 DF,  p-value: < 2.2e-16
```

```
# see the summary table for individual results (t values, p values)

#8

pub.year.boxplot <- ggplot(Neonicotinoids.Raw) +
```

```
  geom_boxplot(aes(x = Chemical.Name, y = Pub..Year)) +
  xlab("Chemicals") +
  ylab("Publication Years") +
  ggtitle("Publication Years of Neonicotinoid Chemicals") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
print(pub.year.boxplot)
```



9. How would you summarize the conclusion of your analysis? Include a sentence summarizing your findings and include the results of your test in parentheses at the end of the sentence.

   ANSWER: Studies on various neonicotinoid chemicals were conducted in different years - this is concluded by statistically different means of the publication years of the chemicals. (Using a linear model, residual standard error: 7.093 on 1274 degrees of freedom, adjusted R-squared: 0.1674, F-statistic: 33.21, p-value: $<$ 2.2e-16)

NTL-LTER test

Research question: What is the best set of predictors for lake temperatures in July across the monitoring period at the North Temperate Lakes LTER?

11. Wrangle your NTL-LTER dataset with a pipe function so that it contains only the following criteria:

- Only dates in July (hint: use the daynum column). No need to consider leap years.
- Only the columns: lakename, year4, daynum, depth, temperature_C
- Only complete cases (i.e., remove NAs)

12. Run an AIC to determine what set of explanatory variables (year4, daynum, depth) is best suited to predict temperature. Run a multiple regression on the recommended set of variables.

```r
#11

# July is daynum 182-212

NTL.ChemPhys.processed <-
  NTL.ChemPhys %>%
  filter(daynum > 181 & daynum < 213) %>% # take only dates in July
  select(lakename, year4, daynum, depth, temperature_C) %>% # choose these columns
  na.omit() # remove any row with NA

#12

# run step AIC
NTL.AIC <- lm(data = NTL.ChemPhys.processed, temperature_C ~ year4 + daynum +
              depth)
step(NTL.AIC)
```

```
## Start:  AIC=26016.31
## temperature_C ~ year4 + daynum + depth
##
##          Df Sum of Sq    RSS   AIC
## <none>                141118 26016
## - year4   1        80 141198 26020
## - daynum  1      1333 142450 26106
## - depth   1    403925 545042 39151
```

```
##
## Call:
## lm(formula = temperature_C ~ year4 + daynum + depth, data = NTL.ChemPhys.processed)
##
## Coefficients:
## (Intercept)        year4       daynum        depth
##    -6.45556      0.01013      0.04134     -1.94726
```

```r
# run multiple regression on recommended model
NTL.model <- lm(data = NTL.ChemPhys.processed, temperature_C ~ year4 + daynum + depth)
summary(NTL.model)
```

```
##
## Call:
## lm(formula = temperature_C ~ year4 + daynum + depth, data = NTL.ChemPhys.processed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.6517 -2.9937  0.0855  2.9692 13.6171
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept) -6.455560   8.638808   -0.747   0.4549
## year4        0.010131   0.004303    2.354   0.0186 *
## daynum       0.041336   0.004315    9.580   <2e-16 ***
## depth       -1.947264   0.011676 -166.782   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3.811 on 9718 degrees of freedom
## Multiple R-squared:  0.7417, Adjusted R-squared:  0.7417
## F-statistic:  9303 on 3 and 9718 DF,  p-value: < 2.2e-16
```

13. What is the final linear equation to predict temperature from your multiple regression? How much of the observed variance does this model explain?

    ANSWER:

Temperature_C = -6.455560 + 0.010131*year4* + *0.041336*daynum -1.947264*depth

The adjusted R-squared value shows that 0.7417 of the observed variance is explained by this model.

14. Run an interaction effects ANCOVA to predict temperature based on depth and lakename from the same wrangled dataset.

```
#14

NTL.ancova <- lm(data = NTL.ChemPhys.processed, temperature_C ~ depth * lakename)
summary(NTL.ancova)
```

```
##
## Call:
## lm(formula = temperature_C ~ depth * lakename, data = NTL.ChemPhys.processed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6455 -2.9133 -0.2879  2.7567 16.3606
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      22.9455     0.5861  39.147  < 2e-16 ***
## depth                            -2.5820     0.2411 -10.711  < 2e-16 ***
## lakenameCrampton Lake             2.2173     0.6804   3.259  0.00112 **
## lakenameEast Long Lake           -4.3884     0.6191  -7.089 1.45e-12 ***
## lakenameHummingbird Lake         -2.4126     0.8379  -2.879  0.00399 **
## lakenamePaul Lake                 0.6105     0.5983   1.020  0.30754
## lakenamePeter Lake                0.2998     0.5970   0.502  0.61552
## lakenameTuesday Lake             -2.8932     0.6060  -4.774 1.83e-06 ***
## lakenameWard Lake                 2.4180     0.8434   2.867  0.00415 **
## lakenameWest Long Lake           -2.4663     0.6168  -3.999 6.42e-05 ***
## depth:lakenameCrampton Lake       0.8058     0.2465   3.268  0.00109 **
## depth:lakenameEast Long Lake      0.9465     0.2433   3.891  0.00010 ***
## depth:lakenameHummingbird Lake   -0.6026     0.2919  -2.064  0.03903 *
## depth:lakenamePaul Lake           0.4022     0.2421   1.662  0.09664 .
## depth:lakenamePeter Lake          0.5799     0.2418   2.398  0.01649 *
## depth:lakenameTuesday Lake        0.6605     0.2426   2.723  0.00648 **
## depth:lakenameWard Lake          -0.6930     0.2862  -2.421  0.01548 *
## depth:lakenameWest Long Lake      0.8154     0.2431   3.354  0.00080 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.471 on 9704 degrees of freedom
## Multiple R-squared:  0.7861, Adjusted R-squared:  0.7857
## F-statistic:  2097 on 17 and 9704 DF,  p-value: < 2.2e-16
```

15. Is there an interaction between depth and lakename? How much variance in the temperature observations does this explain?

ANSWER: The ancova model shows that there is an interaction between depth and lakename, as illustrated by the overall p-value < 2.2e-16. However, some of the p values corresponding to the interactions between depth and specific lakes are not < 0.05, which suggests that this interaction is not significant for all lakes. The adjusted R-squared value shows that 0.7857 of the observed temperature variance is explained by this model.

16. Create a graph that depicts temperature by depth, with a separate color for each lake. Add a geom_smooth (method = "lm", se = FALSE) for each lake. Make your points 50 % transparent. Adjust your y axis limits to go from 0 to 35 degrees. Clean up your graph to make it pretty.

```
#16

temp_by_depth <- ggplot(NTL.ChemPhys.processed, aes(x = depth, y = temperature_C)) +
  geom_point(aes(color = lakename), alpha=0.5, stroke=0) +
  geom_smooth(method=lm, se=FALSE, aes(color = lakename)) + # add line of best fit for each lake
  ylim(0, 35) + # zoom into concentration of points
  ggtitle("Temperature vs Depth, by Lake") + # add main title
  xlab("Depth (m)") + # format labels with UOM
  ylab("Temperature (C)")
print(temp_by_depth)
```

```
## Warning: Removed 73 rows containing missing values (geom_smooth).
```