Assignment 6: Generalized Linear Models

Caroline Reents

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
getwd()
```

```
## [1] "/Users/carolinereents/Desktop/Data Analytics/EnvironmentalDataAnalytics"
```

```
library(tidyverse)
```

```
## -- Attaching packages ----------------------------------------------------------------------- tidyv
```

```
## 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   1.1.1     v forcats 0.3.0
```

```
## -- Conflicts -------------------------------------------------------------------------- tidyverse_c
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
library(readr)
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(RColorBrewer)
library(colormap)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(stringr)
library(dplyr)

ECOTOX_Neonicotinoids_Mortality_raw <- read.csv("./Data/Raw/ECOTOX_Neonicotinoids_Mortality_raw.csv")
NTL_LTER_Lake_ChemistryPhysics_Raw <- read_csv("./Data/Raw/NTL-LTER_Lake_ChemistryPhysics_Raw.csv")
```

```
## Parsed with column specification:
## cols(
##   lakeid = col_character(),
##   lakename = col_character(),
##   year4 = col_integer(),
##   daynum = col_integer(),
##   sampledate = col_character(),
##   depth = col_double(),
##   temperature_C = col_double(),
##   dissolvedOxygen = col_double(),
##   irradianceWater = col_double(),
##   irradianceDeck = col_integer(),
##   comments = col_character()
## )
```

```
## Warning in rbind(names(probs), probs_f): number of columns of result is not
## a multiple of vector length (arg 1)
```

```
## Warning: 3232 parsing failures.
## row # A tibble: 5 x 5 col     row col         expected        actual file
## ... ................. ... ............................................................................
## See problems(...) for more details.
```

```
#2
mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
theme_set(mytheme)
```

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 there are 9
summary(ECOTOX_Neonicotinoids_Mortality_raw$Chemical.Name)
```

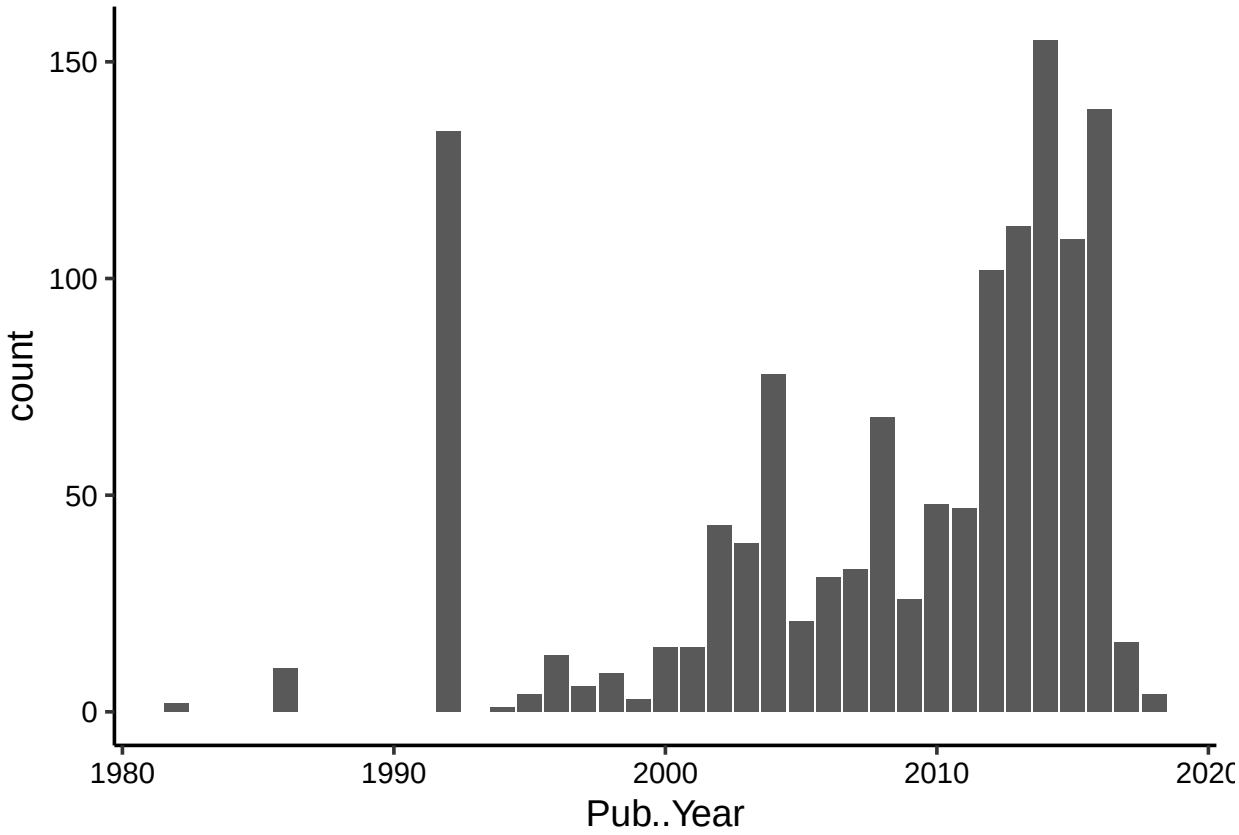
```
##  Acetamiprid Clothianidin  Dinotefuran Imidacloprid Imidaclothiz
##          136           74           59          695            9
##   Nitenpyram   Nithiazine  Thiacloprid Thiamethoxam
##           21           22          106          161
```

```
#4
shapiro.test(ECOTOX_Neonicotinoids_Mortality_raw$Pub..Year)
```

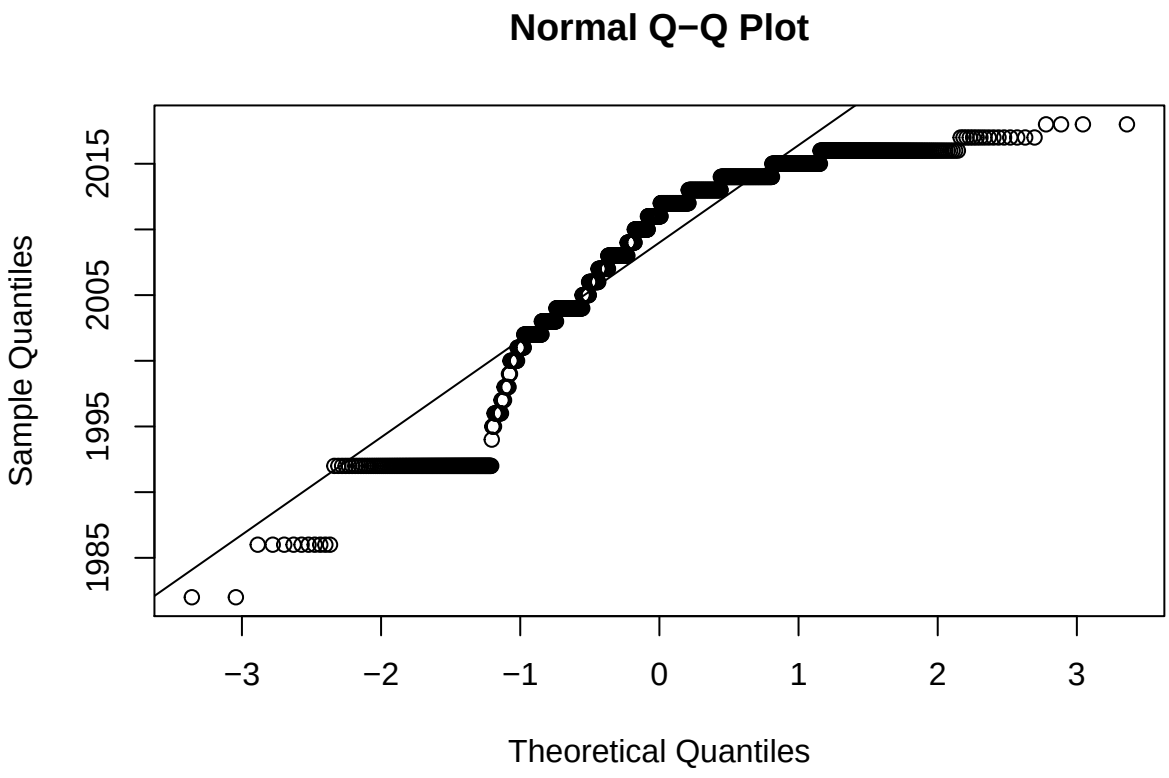
```
##
##  Shapiro-Wilk normality test
##
## data:  ECOTOX_Neonicotinoids_Mortality_raw$Pub..Year
## W = 0.85472, p-value < 2.2e-16
```

```
ggplot(ECOTOX_Neonicotinoids_Mortality_raw, aes(x = Pub..Year)) +
  geom_histogram(stat = "count")
```

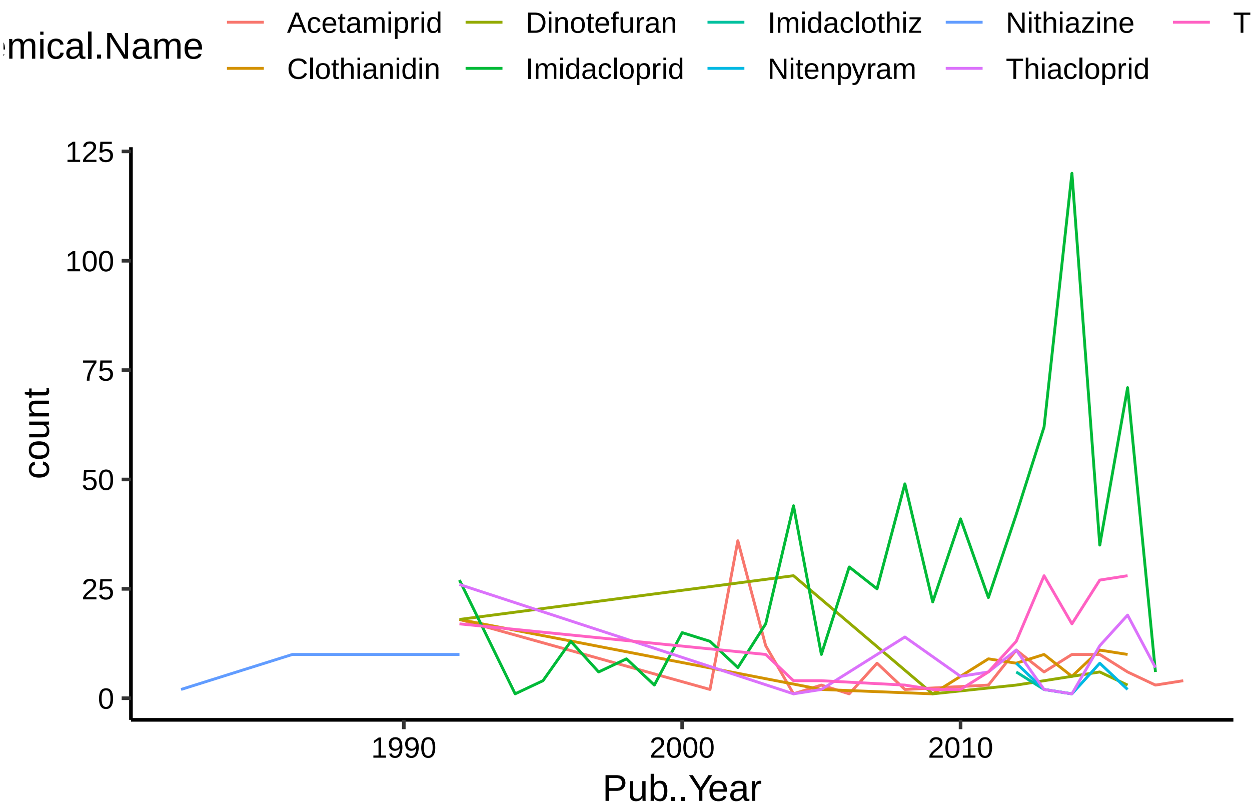
```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```




```
qqnorm(ECOTOX_Neonicotinoids_Mortality_raw$Pub..Year); qqline(ECOTOX_Neonicotinoids_Mortality_raw$Pub..
```




```
chem_poly_plot <- ggplot(ECOTOX_Neonicotinoids_Mortality_raw, aes(x = Pub..Year, color = Chemical.Name))
  geom_freqpoly(stat = "count")
print(chem_poly_plot)
```




```
#5

bartlett.test(ECOTOX_Neonicotinoids_Mortality_raw, Chemical.Name~ECOTOX_Neonicotinoids_Mortality_raw$Pul
```

```
## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.

## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.

## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.

## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.

## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.

## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.

## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.

## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.
```

```
## Warning in FUN(X[[i]], ...): Calling var(x) on a factor x is deprecated and will become an error.
##   Use something like 'all(duplicated(x)[-1L])' to test for a constant vector.
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  ECOTOX_Neonicotinoids_Mortality_raw
## Bartlett's K-squared = Inf, df = 12, p-value < 2.2e-16
```

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

   ANSWER: The vaariance is not equal between the pub year and each chemical. Also, the assumption of normality is not met. Therefore, I would run a kruskal-wallis test.

7. Run this test below.

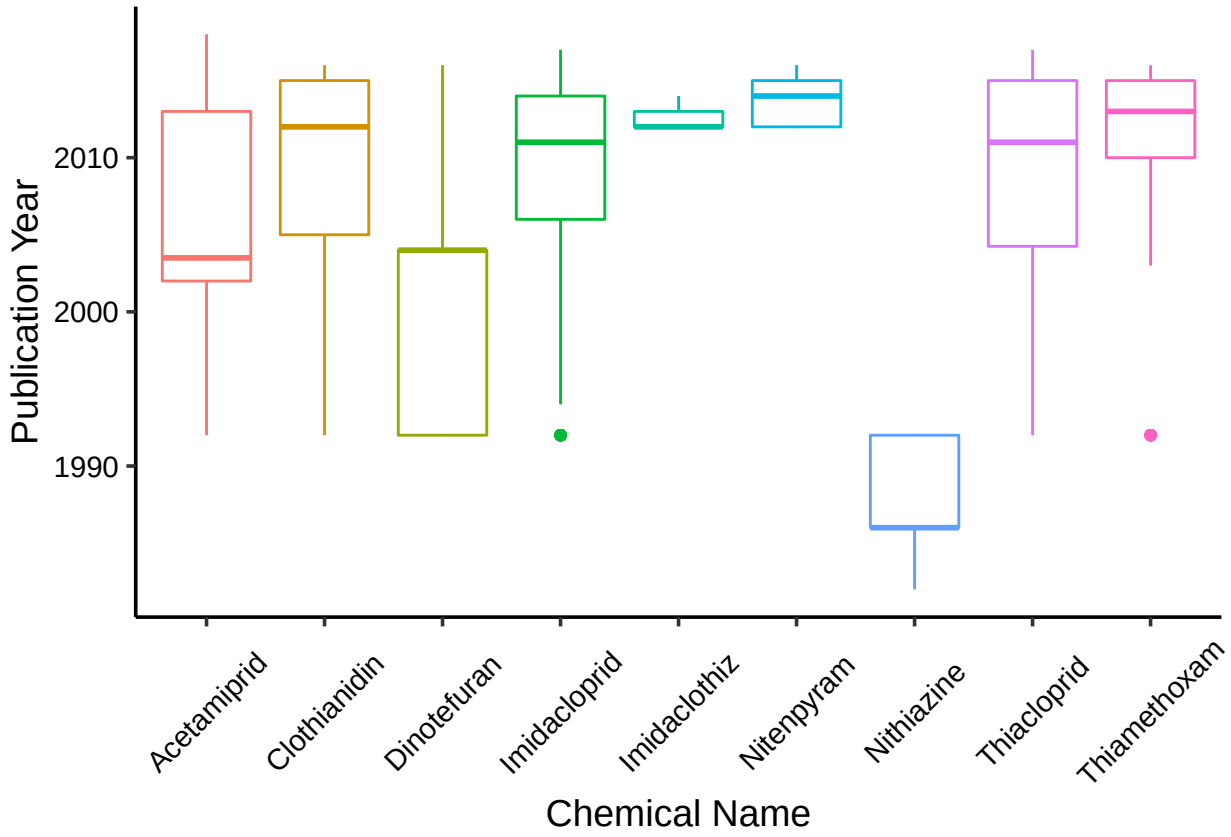
8. Generate a boxplot representing the range of publication years for each chemical. Adjust your graph to make it pretty.

```
#7
kruskal.test(Chemical.Name~Pub..Year, ECOTOX_Neonicotinoids_Mortality_raw)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Chemical.Name by Pub..Year
## Kruskal-Wallis chi-squared = 164.61, df = 27, p-value < 2.2e-16
```

```
#8

chemplot <- ggplot(ECOTOX_Neonicotinoids_Mortality_raw, aes(Chemical.Name, Pub..Year, color=Chemical.Na
  labs(x="Chemical Name", y="Publication Year")+
  theme(axis.text.x = element_text(angle=45, vjust=0.5),  legend.position="none")+
  geom_boxplot()
print(chemplot)
```



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: There were significant differences in the publication years for the studies of neonicotinoids (kruskal-wallis; chi-squared = 164.61, df = 27, p-value < 0.001).

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.

```
#11
class(NTL_LTER_Lake_ChemistryPhysics_Raw$sampledate)
```

```
## [1] "character"
```

```
View(NTL_LTER_Lake_ChemistryPhysics_Raw)
NTL_LTER_Lake_ChemistryPhysics_Raw$sampledate <- as.Date(NTL_LTER_Lake_ChemistryPhysics_Raw$sampledate,
```

```r
july_NTL_LTER <- NTL_LTER_Lake_ChemistryPhysics_Raw %>%
  select("lakename", "year4", "daynum", "depth", "temperature_C") %>%
  mutate(month=month(NTL_LTER_Lake_ChemistryPhysics_Raw$sampledate)) %>%
  na.omit() %>%
  filter(month=="7")


View(july_NTL_LTER)

#12
class(july_NTL_LTER$year4)
```

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

```r
july_NTL_LTER$year4 <- as.numeric(july_NTL_LTER$year4)
july_NTL_LTER$daynum <- as.numeric(july_NTL_LTER$daynum)

lake_AIC <- lm(data = july_NTL_LTER, temperature_C ~ year4 + daynum + depth)
step(lake_AIC)
```

```
## Start:  AIC=26065.53
## temperature_C ~ year4 + daynum + depth
##
##          Df Sum of Sq    RSS   AIC
## <none>                141687 26066
## - year4   1       101 141788 26070
## - daynum  1      1237 142924 26148
## - depth   1    404475 546161 39189
```

```
##
## Call:
## lm(formula = temperature_C ~ year4 + daynum + depth, data = july_NTL_LTER)
##
## Coefficients:
## (Intercept)        year4       daynum        depth
##    -8.57556      0.01134      0.03978     -1.94644
```

```r
model1 <- lm(data = july_NTL_LTER, temperature_C ~ year4 + daynum)
summary(model1)
```

```
##
## Call:
## lm(formula = temperature_C ~ year4 + daynum, data = july_NTL_LTER)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.279  -7.158  -2.591   8.072  21.402
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.827705  16.944033  -0.167    0.867
## year4        0.003779   0.008439   0.448    0.654
## daynum       0.040484   0.008475   4.777 1.81e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 7.494 on 9725 degrees of freedom
## Multiple R-squared:  0.002363,   Adjusted R-squared:  0.002158
## F-statistic: 11.52 on 2 and 9725 DF,  p-value: 1.007e-05
```

```
model2 <- lm(data = july_NTL_LTER, temperature_C ~ depth)
summary(model2)
```

```
##
## Call:
## lm(formula = temperature_C ~ depth, data = july_NTL_LTER)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5173 -3.0192  0.0633  2.9365 13.5834
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 21.95597    0.06792   323.3   <2e-16 ***
## depth       -1.94621    0.01174  -165.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.835 on 9726 degrees of freedom
## Multiple R-squared:  0.7387, Adjusted R-squared:  0.7387
## F-statistic: 2.75e+04 on 1 and 9726 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: temp = 21.96 + -1.95 (depth) + 3.84; 73.87% of the variance

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

```
#14
model2_ancova <- lm(data = july_NTL_LTER, temperature_C ~ lakename + depth)
model3_ancova <- lm(data = july_NTL_LTER, temperature_C ~ lakename * depth)
summary(model2_ancova)
```

```
##
## Call:
## lm(formula = temperature_C ~ lakename + depth, data = july_NTL_LTER)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1062 -3.0182 -0.2145  2.8397 15.1605
##
## Coefficients:
##                            Estimate Std. Error  t value Pr(>|t|)
## (Intercept)                21.67335    0.31408   69.006  < 2e-16 ***
## lakenameCrampton Lake       4.53288    0.37298   12.153  < 2e-16 ***
## lakenameEast Long Lake     -1.44524    0.33500   -4.314 1.62e-05 ***
## lakenameHummingbird Lake   -4.87775    0.45450  -10.732  < 2e-16 ***
## lakenamePaul Lake           0.93875    0.32184    2.917  0.00354 **
## lakenamePeter Lake          1.40045    0.32179    4.352 1.36e-05 ***
## lakenameTuesday Lake       -1.39244    0.32746   -4.252 2.14e-05 ***
## lakenameWard Lake          -0.67149    0.45458   -1.477  0.13967
```

```
## lakenameWest Long Lake   -0.17061    0.33389   -0.511  0.60938
## depth                    -1.96509    0.01096 -179.268  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.544 on 9718 degrees of freedom
## Multiple R-squared:  0.777,  Adjusted R-squared:  0.7768
## F-statistic:  3762 on 9 and 9718 DF,  p-value: < 2.2e-16
```

```
summary(model3_ancova)
```

```
##
## Call:
## lm(formula = temperature_C ~ lakename * depth, data = july_NTL_LTER)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6470 -2.9129 -0.2949  2.7469 16.3606
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      22.8748     0.5660  40.412  < 2e-16 ***
## lakenameCrampton Lake             2.2881     0.6634   3.449 0.000565 ***
## lakenameEast Long Lake           -4.3176     0.6002  -7.194 6.76e-13 ***
## lakenameHummingbird Lake         -2.3418     0.8246  -2.840 0.004523 **
## lakenamePaul Lake                 0.7115     0.5786   1.230 0.218863
## lakenamePeter Lake                0.3884     0.5774   0.673 0.501146
## lakenameTuesday Lake             -2.8656     0.5864  -4.887 1.04e-06 ***
## lakenameWard Lake                 2.4887     0.8302   2.998 0.002728 **
## lakenameWest Long Lake           -2.3819     0.5983  -3.981 6.91e-05 ***
## depth                            -2.5543     0.2331 -10.956  < 2e-16 ***
## lakenameCrampton Lake:depth       0.7781     0.2388   3.258 0.001125 **
## lakenameEast Long Lake:depth      0.9189     0.2354   3.903 9.56e-05 ***
## lakenameHummingbird Lake:depth   -0.6303     0.2856  -2.207 0.027334 *
## lakenamePaul Lake:depth           0.3716     0.2342   1.587 0.112592
## lakenamePeter Lake:depth          0.5511     0.2339   2.356 0.018500 *
## lakenameTuesday Lake:depth        0.6472     0.2347   2.758 0.005826 **
## lakenameWard Lake:depth          -0.7207     0.2797  -2.577 0.009991 **
## lakenameWest Long Lake:depth      0.7892     0.2353   3.354 0.000800 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.476 on 9710 degrees of freedom
## Multiple R-squared:  0.7857, Adjusted R-squared:  0.7853
## F-statistic:  2094 on 17 and 9710 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: There are significant interactions between depth and lake name. It explains 78.57% of the variance in temperature

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.

```r
#16
summary(july_NTL_LTER$lakename)
```

```
##    Length     Class      Mode
##      9728 character character
```

```r
tempbydepth_plot <- ggplot(july_NTL_LTER, aes(x = depth, y = temperature_C, color = lakename)) +
  geom_point() +
  scale_colour_brewer(palette = "Set1")+
  geom_smooth(method = "lm", se = FALSE) +
  labs(x="Depth (m)", y= "Temperature (C)")+
  ylim(0, 35)

print(tempbydepth_plot)
```

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