Assignment 6: Generalized Linear Models

Xin Zhang

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] "C:/Users/Xin Zhang/Desktop/EDA"
```

```
library(tidyverse)
```

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

## v ggplot2 3.1.0     v purrr   0.3.0
## v tibble  2.0.1     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 ---------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(viridis)
```

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

```
library(RColorBrewer)
library(colormap)
Ecotox <- read.csv("./Data/Raw/ECOTOX_Neonicotinoids_Mortality_raw.csv")
ChemPhy <- read.csv("./Data/Raw/NTL-LTER_Lake_ChemistryPhysics_Raw.csv")
```

```
#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
summary(Ecotox$Chemical.Name)
```

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

```
nlevels(Ecotox$Chemical.Name)
```

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

```
#9 different chemicals are listed

#4
class(Ecotox$Pub..Year)
```

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

```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Acetamiprid"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Acetamiprid"]
## W = 0.90191, p-value = 5.706e-08
```

```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Clothianidin"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Clothianidin"]
## W = 0.69577, p-value = 4.287e-11
```

```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Dinotefuran"])
```

```
##
##  Shapiro-Wilk normality test
```

```
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Dinotefuran"]
## W = 0.82848, p-value = 8.83e-07
```

```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Imidacloprid"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Imidacloprid"]
## W = 0.88178, p-value < 2.2e-16
```

```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Imidaclothiz"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Imidaclothiz"]
## W = 0.68429, p-value = 0.00093
```

```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Nitenpyram"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Nitenpyram"]
## W = 0.79592, p-value = 0.0005686
```

```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Nithiazine"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Nithiazine"]
## W = 0.75938, p-value = 0.0001235
```

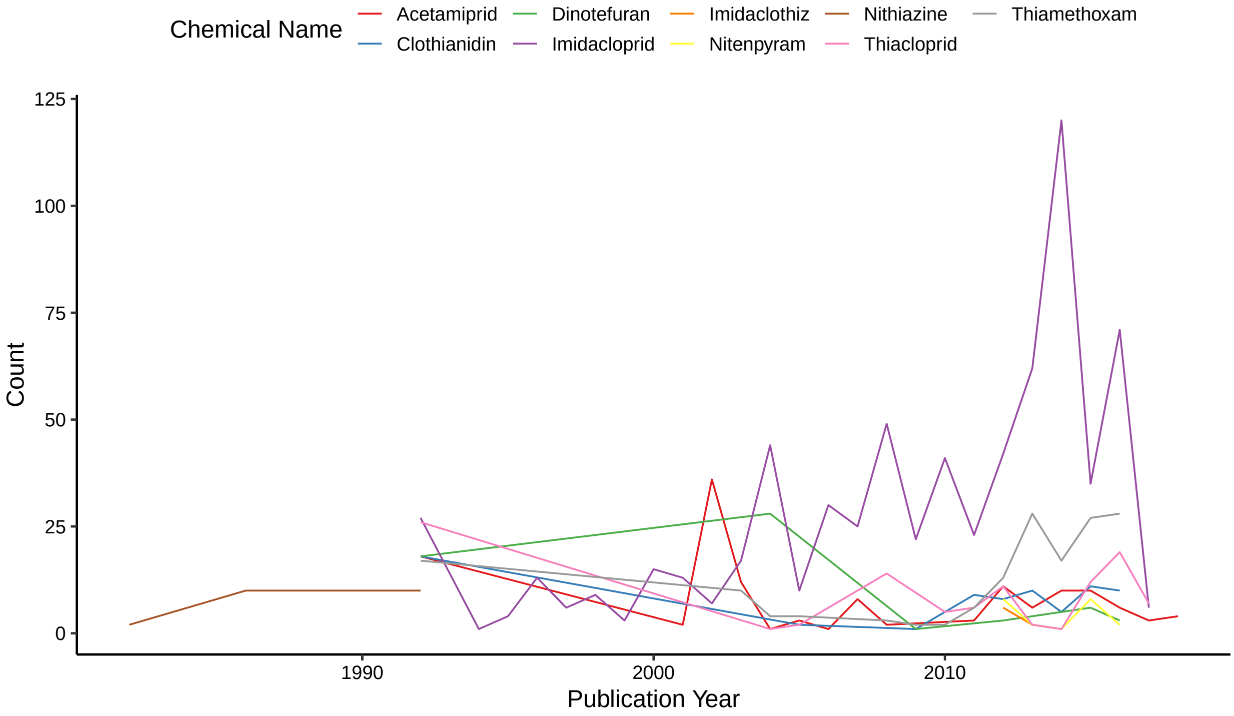
```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Thiacloprid"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Thiacloprid"]
## W = 0.7669, p-value = 1.118e-11
```

```
shapiro.test(Ecotox$Pub..Year[Ecotox$Chemical.Name == "Thiamethoxam"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Ecotox$Pub..Year[Ecotox$Chemical.Name == "Thiamethoxam"]
## W = 0.7071, p-value < 2.2e-16
```

```
ggplot(Ecotox, aes(x = Pub..Year, color = Chemical.Name)) +
  geom_freqpoly(stat = "count")+
  labs(x ="Publication Year", y="Count", color ="Chemical Name")+
   scale_color_brewer(palette = "Set1", direction = 1)
```



```
#5
bartlett.test(Ecotox$Pub..Year ~ Ecotox$Chemical.Name)
```

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

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

   ANSWER: Since they are not normal distributions (sharpiro.test pvalue<0.0001), and there are not equal variance (bartlett.test, df=8, pvalue<0.0001), I will choose to run a Non-parametric equivalent of ANOVA: Kruskal-Wallis Test.

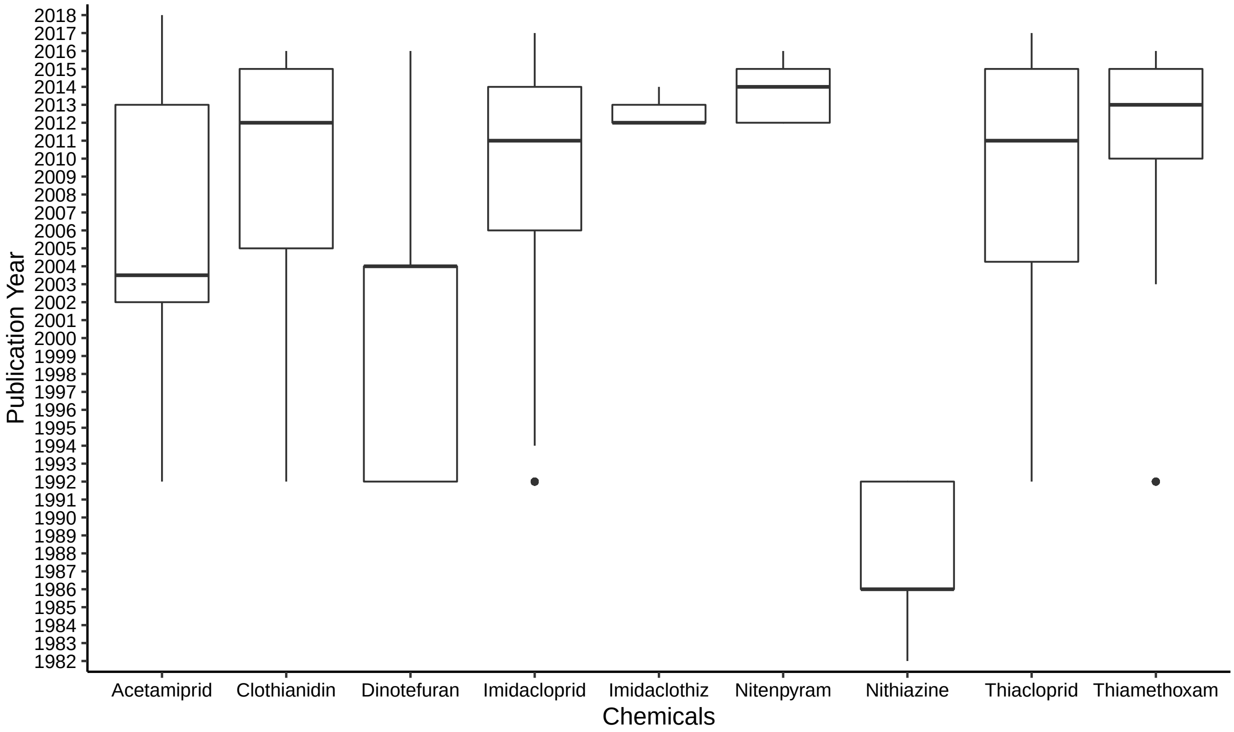
7. Run this test below.

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

```
#7
yr.chemical <- kruskal.test(Ecotox$Pub..Year ~ Ecotox$Chemical.Name)
yr.chemical
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Ecotox$Pub..Year by Ecotox$Chemical.Name
## Kruskal-Wallis chi-squared = 134.15, df = 8, p-value < 2.2e-16
```

```
#8
ggplot(Ecotox, aes(y = Pub..Year, x = Chemical.Name)) +
  geom_boxplot()+
  labs(x ="Chemicals", y="Publication Year")+
  scale_y_discrete(limits = (c(1982:2018)))
```



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 conducted in different years (Kruskal-Wallis rank sum test; Kruskal-Wallis chi-squared = 134.15, df = 8, p<0.0001)

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
#daynum cannot make it to choose July because 6/30 and 7/1 are both 182....
ChemPhy2<-cbind(Month = ChemPhy$sampledate, ChemPhy)
ChemPhy2$Month <- as.Date(ChemPhy2$Month, format = "%m/%d/%y")
ChemPhy2$Month <- format.Date(ChemPhy2$Month, format = "%m")
subchemphy<-
  #ChemPhy2 %>%
  #filter(Month == "07") %>%
  ChemPhy %>%
  filter(daynum >=182 & daynum<=213) %>%
  select(lakename, year4, daynum, depth, temperature_C) %>%
  na.omit()

subchemphy2<-
  ChemPhy2 %>%
  filter(Month == "07") %>%
  select(lakename, year4, daynum, depth, temperature_C) %>%
  na.omit()

#12
tempAIC <- lm(data = subchemphy, temperature_C ~ year4 + daynum + depth)
step(tempAIC)
```

```
## Start:  AIC=26781.56
## temperature_C ~ year4 + daynum + depth
##
##          Df Sum of Sq    RSS   AIC
## <none>                146054 26782
## - year4   1       154 146209 26790
## - daynum  1      1582 147636 26887
## - depth   1    414049 560103 40189

##
## Call:
## lm(formula = temperature_C ~ year4 + daynum + depth, data = subchemphy)
##
## Coefficients:
## (Intercept)        year4       daynum        depth
##   -14.33180      0.01386      0.04337     -1.94112
```

```
tempAIC2 <- lm(data = subchemphy2, temperature_C ~ year4 + daynum + depth)
step(tempAIC2)
```

```
## 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 = subchemphy2)
##
## Coefficients:
## (Intercept)        year4       daynum        depth
##    -8.57556      0.01134      0.03978     -1.94644
```

```
#full model has the smallest AIC: temperature_C ~ year4 + daynum + depth
summary(tempAIC)
```

```
##
## Call:
```

```
## lm(formula = temperature_C ~ year4 + daynum + depth, data = subchemphy)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
## -9.669 -3.014  0.091  2.977 13.606 
## 
## Coefficients:
##               Estimate Std. Error  t value Pr(>|t|)    
## (Intercept) -14.331802   8.582522   -1.670  0.09497 .  
## year4         0.013861   0.004274    3.243  0.00119 ** 
## daynum        0.043368   0.004173   10.393  < 2e-16 ***
## depth        -1.941121   0.011545 -168.135  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 3.827 on 9972 degrees of freedom
## Multiple R-squared:  0.7399, Adjusted R-squared:  0.7398 
## F-statistic:  9457 on 3 and 9972 DF,  p-value: < 2.2e-16
```

```r
summary(tempAIC2)
```

```
## 
## Call:
## lm(formula = temperature_C ~ year4 + daynum + depth, data = subchemphy2)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -9.6536 -3.0000  0.0902  2.9658 13.6123 
## 
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)    
## (Intercept) -8.575564   8.630715   -0.994  0.32044    
## year4        0.011345   0.004299    2.639  0.00833 ** 
## daynum       0.039780   0.004317    9.215  < 2e-16 ***
## depth       -1.946437   0.011683 -166.611  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 3.817 on 9724 degrees of freedom
## Multiple R-squared:  0.7412, Adjusted R-squared:  0.7411 
## F-statistic:  9283 on 3 and 9724 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: Final linear equation: temperature_C = -14.33 + 0.01*year4 + 0.04*daynum - 1.94*depth. (linear regression, R2=0.74, df=9972, p<0.0001(except for the intercept)). This model explains 74% variance.The coefficient of the intercept -14.33 means that when year4, daynum and depth are are 0, the temperature will be -14.33 celsius. The coefficient of year4 means that when year4 increases by 1, the temperature will increase by 0.01.The coefficient of daynum means that when daynum increases by 1, the temperature will increase by 0.04.The coefficient of depth means that when depth increases by 1, the temperature will decrease by 1.94.

    If choose July directly instead of using daynum, the result will be the following: temperature_C = -8.58 + 0.01*year4 + 0.04*daynum - 1.95*depth. (linear regression, R2=0.74, df=9724, p<0.0001(except for the intercept)).The coefficient of the intercept -14.33 means that when year4, daynum and depth are are 0, the temperature will be -8.58 celsius. The coefficient of year4 means that when year4 increases by 1, the temperature will increase by 0.01.The coefficient of daynum means that when daynum increases by 1, the temperature will increase by 0.04.The coefficient of depth means that when depth increases by 1, the temperature will decrease by 1.95.

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

```
#14
temp2 <- lm(data = subchemphy, temperature_C ~ depth * lakename)
summary(temp2)
```

```
##
## Call:
## lm(formula = temperature_C ~ depth * lakename, data = subchemphy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6410 -2.9075 -0.2944  2.7531 16.3358
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     22.8748     0.5658  40.427  < 2e-16 ***
## depth                           -2.5543     0.2331 -10.960  < 2e-16 ***
## lakenameCrampton Lake            2.5625     0.6518   3.932 8.49e-05 ***
## lakenameEast Long Lake          -4.2925     0.5993  -7.163 8.46e-13 ***
## lakenameHummingbird Lake        -2.1903     0.8044  -2.723 0.006483 **
## lakenamePaul Lake                0.7115     0.5784   1.230 0.218684
## lakenamePeter Lake               0.3862     0.5770   0.669 0.503250
## lakenameTuesday Lake            -2.8635     0.5857  -4.889 1.03e-06 ***
## lakenameWard Lake                2.4887     0.8299   2.999 0.002718 **
## lakenameWest Long Lake          -2.4193     0.5959  -4.060 4.94e-05 ***
## depth:lakenameCrampton Lake      0.7704     0.2379   3.238 0.001208 **
## depth:lakenameEast Long Lake     0.9181     0.2353   3.902 9.60e-05 ***
## depth:lakenameHummingbird Lake  -0.6738     0.2831  -2.380 0.017323 *
## depth:lakenamePaul Lake          0.3716     0.2341   1.587 0.112452
## depth:lakenamePeter Lake         0.5503     0.2338   2.354 0.018612 *
## depth:lakenameTuesday Lake       0.6486     0.2345   2.766 0.005687 **
## depth:lakenameWard Lake         -0.7207     0.2796  -2.578 0.009962 **
## depth:lakenameWest Long Lake     0.7928     0.2351   3.373 0.000747 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.475 on 9958 degrees of freedom
## Multiple R-squared:  0.7859, Adjusted R-squared:  0.7855
## F-statistic:  2150 on 17 and 9958 DF,  p-value: < 2.2e-16
```

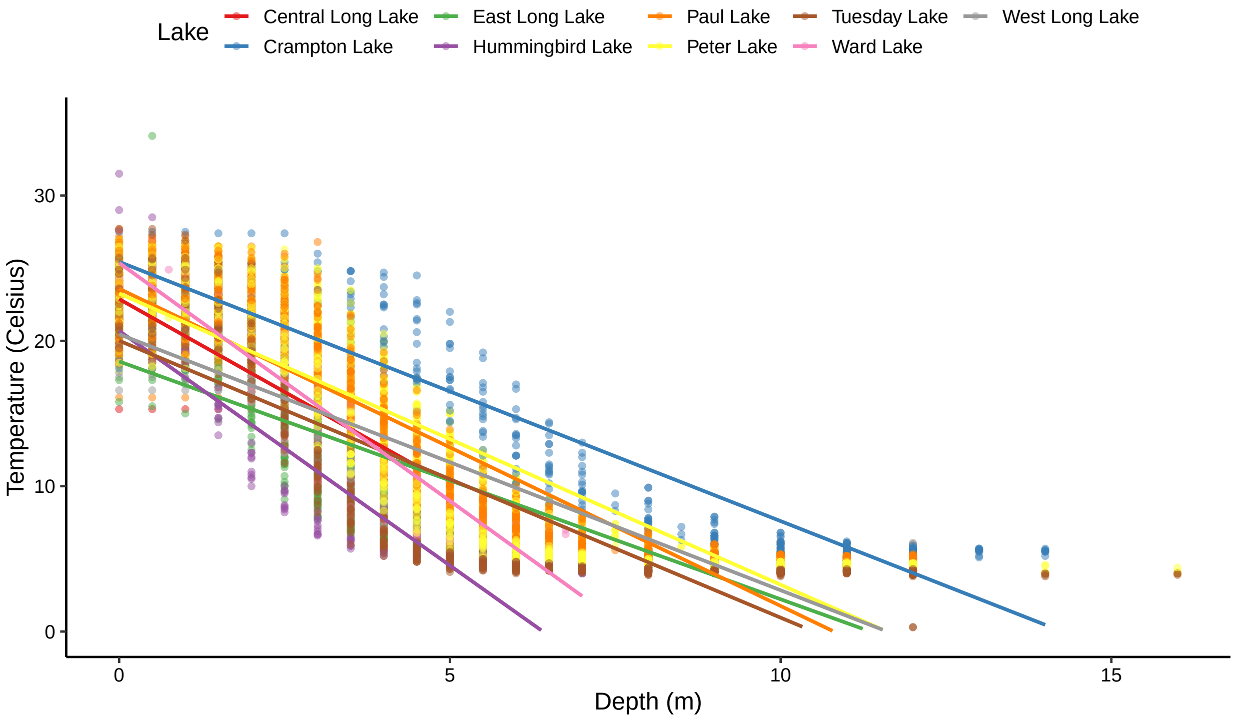
15. Is there an interaction between depth and lakename? How much variance in the temperature observations does this explain?

    ANSWER: Yes, there is an interatction between depth and lakename (interactive variables p<0.01). This model explain 79% variance. (ANVOVA, R2=0.79, df=9710, p<0.01)

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
ggplot(subchemphy, aes(x = depth, y = temperature_C, color = lakename)) +
  geom_point(alpha=0.5)+
  geom_smooth(method = "lm", se = FALSE)+
  ylim(0,35)+
  labs(x ="Depth (m)", y="Temperature (Celsius)", color ="Lake")+
 scale_color_brewer(palette = "Set1", direction = 1)
```

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


Lake
Central Long Lake
Crampton Lake
East Long Lake
Hummingbird Lake
Paul Lake
Peter Lake
Tuesday Lake
Ward Lake
West Long Lake
Temperature (Celsius)
Depth (m)