CSI 2300: Intro to Data Science

In-Class Exercise 09: Exploratory Data Analysis

Evelyn Pan

The data for today's exercises come from the `mowater` library, the `eml` dataset. This data are about measurements of the properties of water in the Eagle Mountain Lake reservoir in North Texas.

1. Load the `mowater` library, and then load the `eml` dataset. In the RMarkdown document, show the commands you use to do this, but not the output of those commands (`message=FALSE` as an option in the code chunk header is a good way to do this).

```
library(mowateR)
data(eml)
```

2. Next, inspect the dataset.

   - How many variables are there? **There are 7 variables**
   - How many observations are there? **There are 35532 observations**
   - Are the data already sorted in time order? **Yes, the data is sorted**
   - What do the variables represent, do you think? You may want to consult the `help` for the dataset to understand it better, including the units of measurement. **Date-Time; Date and two-hour time measurement was taken. Depth; Profile depth, measured in meters. Temp; water temperature, measured in degree Celsius. DO; Dissolved Oxygen, measured in mg/L. DOsat; Dissolved Oxygen Saturation, measured as the percentage of DO relative to what the concentration would be in equilibrium with the atmosphere, calcualted as actual/expected. pH; pH, measured as standard pH from 0 (basic) to 14 (acidic). Cond; Conductivity in water, measured in (micro Siemens per centimeter).**
   - What are the ranges of the values?

```
#View(eml)

max(eml$Date.Time) - min(eml$Date.Time)
# Time difference of 140.9167 days
max(eml$Depth) - min(eml$Depth)
# [1] 10
max(eml$Temp) - min(eml$Temp)
# [1] 16.435
max(eml$DO) - min(eml$DO)
```

```
# [1] 15.508
max(eml$DOsat) - min(eml$DOsat)
# [1] 218.645
max(eml$pH) - min(eml$pH)
# [1] 2.613
```

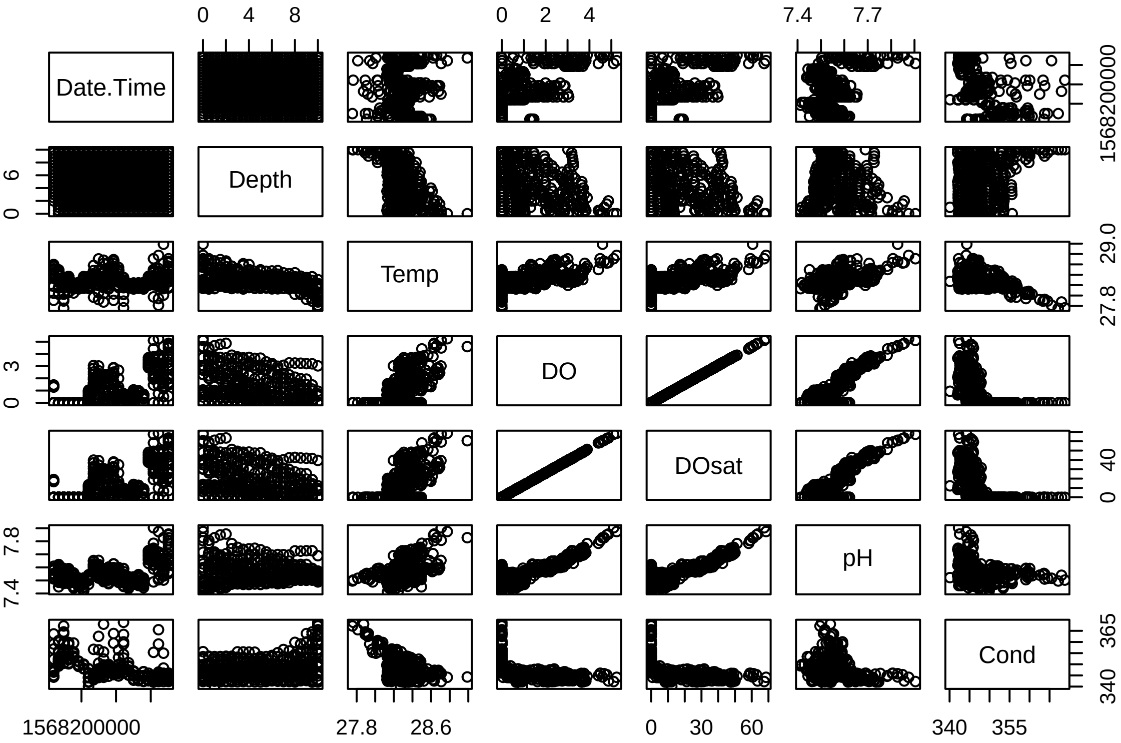
- Are there any missing values (NA values)? **No**

3. The size of our dataset is rather large for easy (and fast) manipulation. Pare it down by creating a new data frame with only the last 500 observations in it. This is throwing away a lot of information, and we should be careful any time we do this. But if we try to use all the data, it may be too time-consuming for an in-class exercise.

```
new_eml = tail(eml, 500)
```

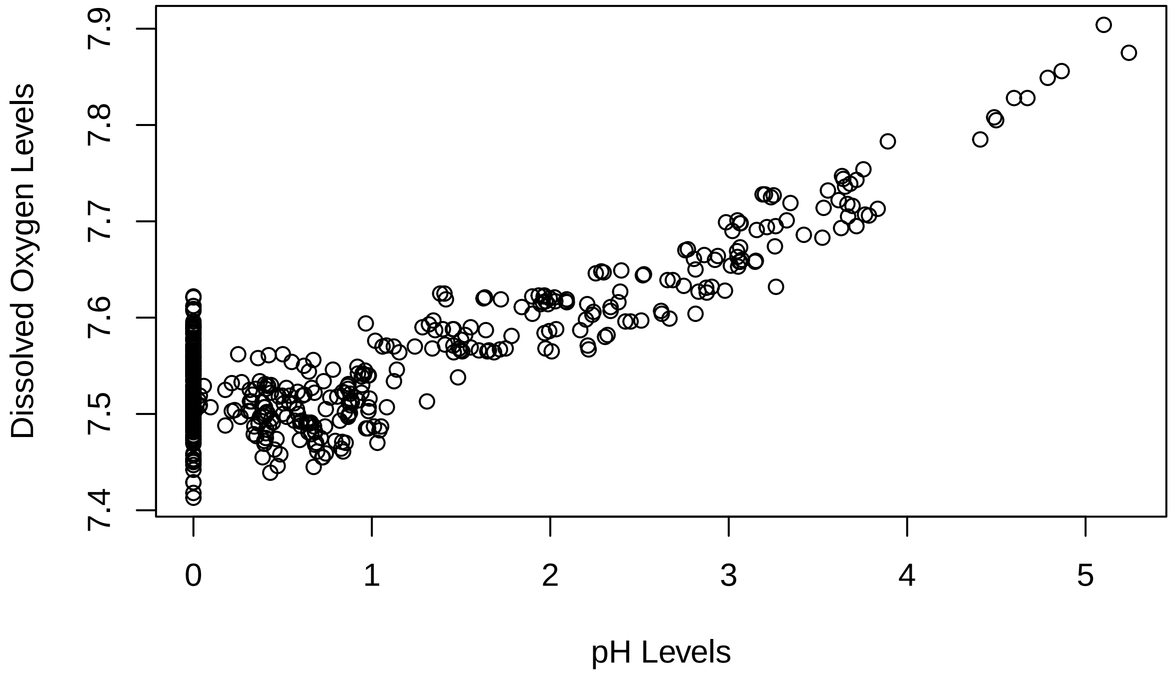
4. Now that the data are of a manageable size for visualizing it, plot all of the variables against each other. Try calling `plot(eml_small)`, but replace `eml_small` with the name of the *small* data frame you just created. This creates a matrix of pairwise scatterplots.
(If this takes a very long time, you may have made the mistake of trying to plot the original dataset, which is quite large for this task.)

```
plot(new_eml)
```



5. Look at the plot and consider which pairs of variables appear to be linearly related. Choose DO and pH. Make a scatter plot of DO on the y-axis and pH on the x-axis. Which one are you thinking of as the independent variable, and which one are you thinking of as the dependent variable?

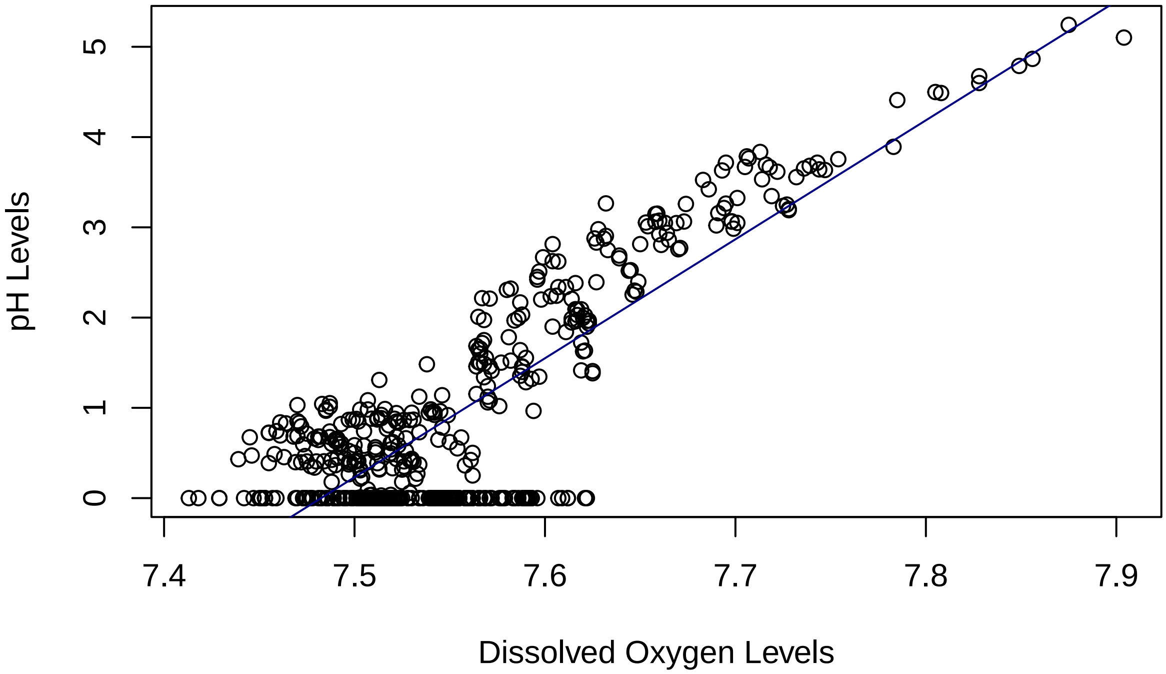
```
plot(new_eml$DO, new_eml$pH,
     xlab = "pH Levels",
     ylab = "Dissolved Oxygen Levels")
```



The x axis is always the independent variable. Thus, the pH Levels depend on the acidity (pH levels) of the water.

6. Run a linear regression by calling `lm(DO ~ pH, data=eml_small)` where again `eml_small` is the small data frame. Plot the regression line on the scatter plot using `abline` on the model that `lm` returns. If the line doesn't appear to follow the data, you may have switched the variables (for the plot versus the linear regression).

```
reg = lm(DO ~ pH, data=new_eml)
plot(new_eml$pH, new_eml$DO,
     ylab = "pH Levels",
     xlab = "Dissolved Oxygen Levels")
abline(reg, col = "navyblue")
```



7. Investigate the coefficients and summary statistics of the model that `lm` gave you. Comment on the coefficient values, significance levels of the coefficients (are they significantly different from 0), and R^2 values. Does there appear to be a linear relationship between these two variables?

```
summary(reg)
#
# Call:
# lm(formula = DO ~ pH, data = new_eml)
#
# Residuals:
#      Min       1Q   Median       3Q      Max
# -1.83942 -0.41907  0.09877  0.48027  1.29376
#
# Coefficients:
#              Estimate Std. Error t value Pr(>|t|)
# (Intercept) -98.6337     2.8287  -34.87   <2e-16 ***
# pH           13.1820     0.3743   35.22   <2e-16 ***
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
```

```
# Residual standard error: 0.6543 on 498 degrees of freedom
# Multiple R-squared:  0.7135,  Adjusted R-squared:  0.713
# F-statistic:  1240 on 1 and 498 DF,  p-value: < 2.2e-16
```

Coefficient Values: The estimated Significance levels of coefficients: Linear relationship?

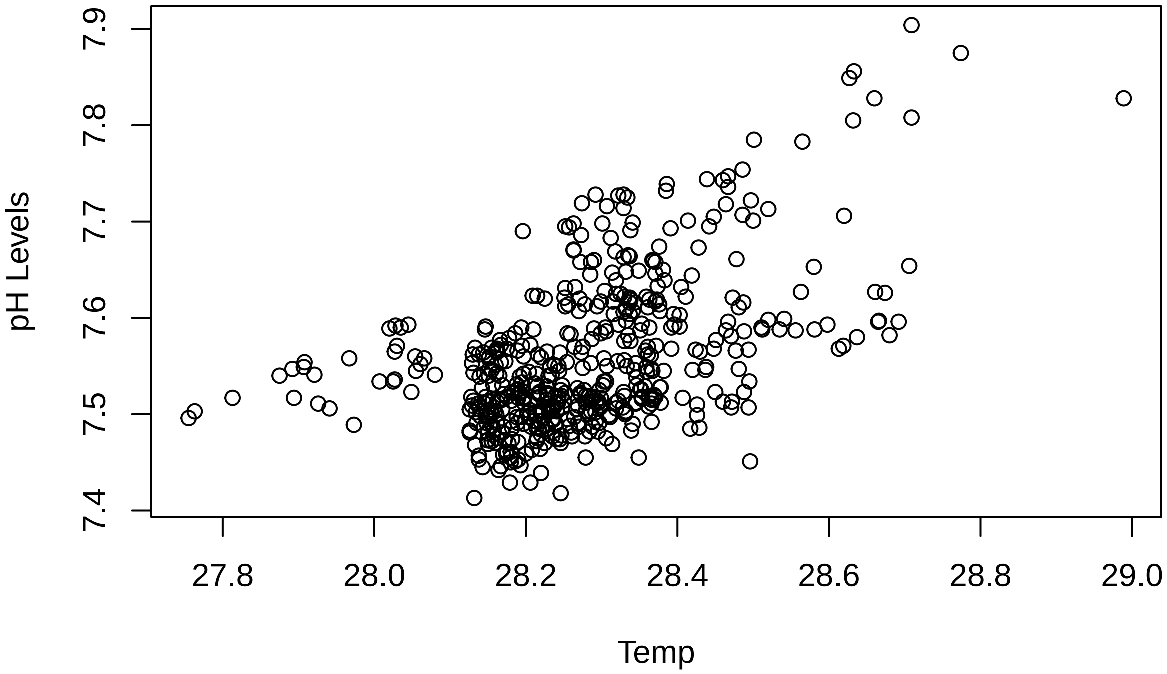
8. Use the model to make a prediction of the dependent variable for when the variable `pH` is 7.7. You can do this by direct computation if you want (rather than using any specialty R command). Hand-check your work on your plot. Is the predicted value close to the plotted data? **The predicted value of 7.7 DO level is close to the pH level on the y-axis of 2.8677**

```
13.182 * 7.7 - 98.6337
# [1] 2.8677
```

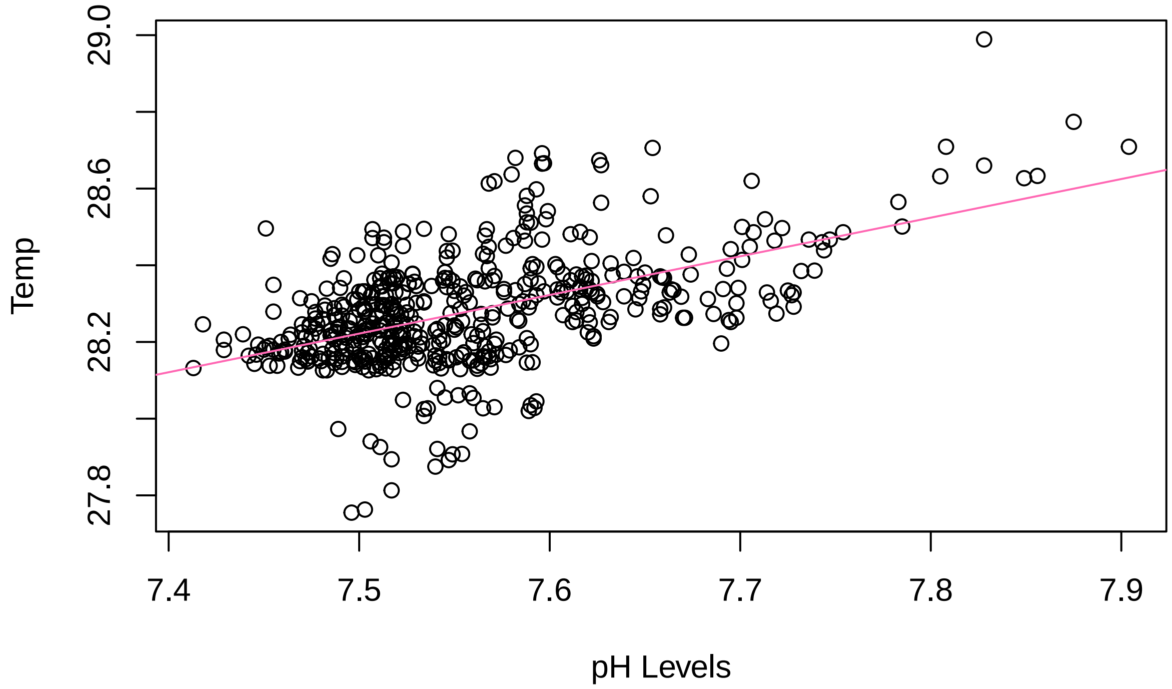
9. Do you trust this model to make a prediction of `DO` for a `pH` value of 3? Explain your answer. **No, because if you have a value that's out of the x range of 7.4 - 7.9, you can't extrapolate the data that's out of the x range.**

10. Extra credit: repeat steps 5-7, but for a different pair of variables than the pair you were just working with. Compare the models; does it appear that one pair of variables is more strongly linearly related than the other pair? Note, do not choose the pair `DO` and `DOsat` as they are the same variable but measured in different units. **Since it's less than 1, it proves evidence that it is less than zero.**

```
plot(new_eml$Temp, new_eml$pH,
     xlab = "Temp",
     ylab = "pH Levels")
```



```r
reg = lm(Temp ~ pH, data=new_eml)
plot(new_eml$pH, new_eml$Temp,
     ylab = "Temp",
     xlab = "pH Levels")
abline(reg, col = "hotpink")
```



```
summary(reg)
#
# Call:
# lm(formula = Temp ~ pH, data = new_eml)
#
# Residuals:
#      Min       1Q   Median       3Q      Max
# -0.46273 -0.06880 -0.00200  0.06676  0.43668
#
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept) 20.66346    0.55041   37.54   <2e-16 ***
# pH           1.00777    0.07283   13.84   <2e-16 ***
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# Residual standard error: 0.1273 on 498 degrees of freedom
# Multiple R-squared:  0.2777,  Adjusted R-squared:  0.2763
# F-statistic: 191.5 on 1 and 498 DF,  p-value: < 2.2e-16
```