Assignment 5: Data Visualization

Xin Wang

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics (ENV872L) on data wrangling.

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_A04_DataWrangling.pdf") prior to submission.

The completed exercise is due on Thursday, 14 February, 2019 before class begins.

Set up your session

1. Set up your session. Upload the NTL-LTER processed data files for chemistry/physics for Peter and Paul Lakes (tidy and gathered), the USGS stream gauge dataset, and the EPA Ecotox dataset for Neonicotinoids.

2. Make sure R is reading dates as date format, not something else (hint: remember that dates were an issue for the USGS gauge data).

```
#1
getwd()
```

```
## [1] "Y:/19spring/872/Environmental_Data_Analytics/Assignments"
```

```
setwd("Y:/19spring/872/Environmental_Data_Analytics/Assignments")
PPchemphy <- read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv")
#Sorry, I didn't upload the gathered one.
USGSgage <- read.csv("../Data/Raw/USGS_Site02085000_Flow_Raw.csv")
ecotox <- read.csv("../Data/Raw/ECOTOX_Neonicotinoids_Mortality_raw.csv")
library(tidyverse)
```

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

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

```
## 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   1.3.1     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

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

## Warning: package 'stringr' was built under R version 3.5.2

## Warning: package 'forcats' was built under R version 3.5.2

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

```
#2
USGSgage$datetime <- as.Date(USGSgage$datetime,"%m/%d/%y")
USGSgage$datetime <- format(USGSgage$datetime,"%y%m%d")
creat.early.dates <- function(d) paste0(ifelse(d > 181231,"19","20"),d)
USGSgage$datetime <- creat.early.dates(USGSgage$datetime)
USGSgage$datetime <- as.Date(USGSgage$datetime,"%Y%m%d")
PPchemphy$sampledate <- as.Date(PPchemphy$sampledate,"%Y-%m-%d")
```

Define your theme

3. Build a theme and set it as your default theme.

```
#3
wang <- theme_classic() +
  theme(plot.title=element_text(size = 20,hjust = 0.5),
        panel.background=element_rect(fill="white",color="grey30"),
        axis.title = element_text(size = 15),
        legend.title = element_text(size = 15), legend.text = element_text(size = 12),
        legend.margin=margin(6,6,6,6))
```

Create graphs

For numbers 4-7, create graphs that follow best practices for data visualization. To make your graphs "pretty," ensure your theme, color palettes, axes, and legends are edited to your liking.

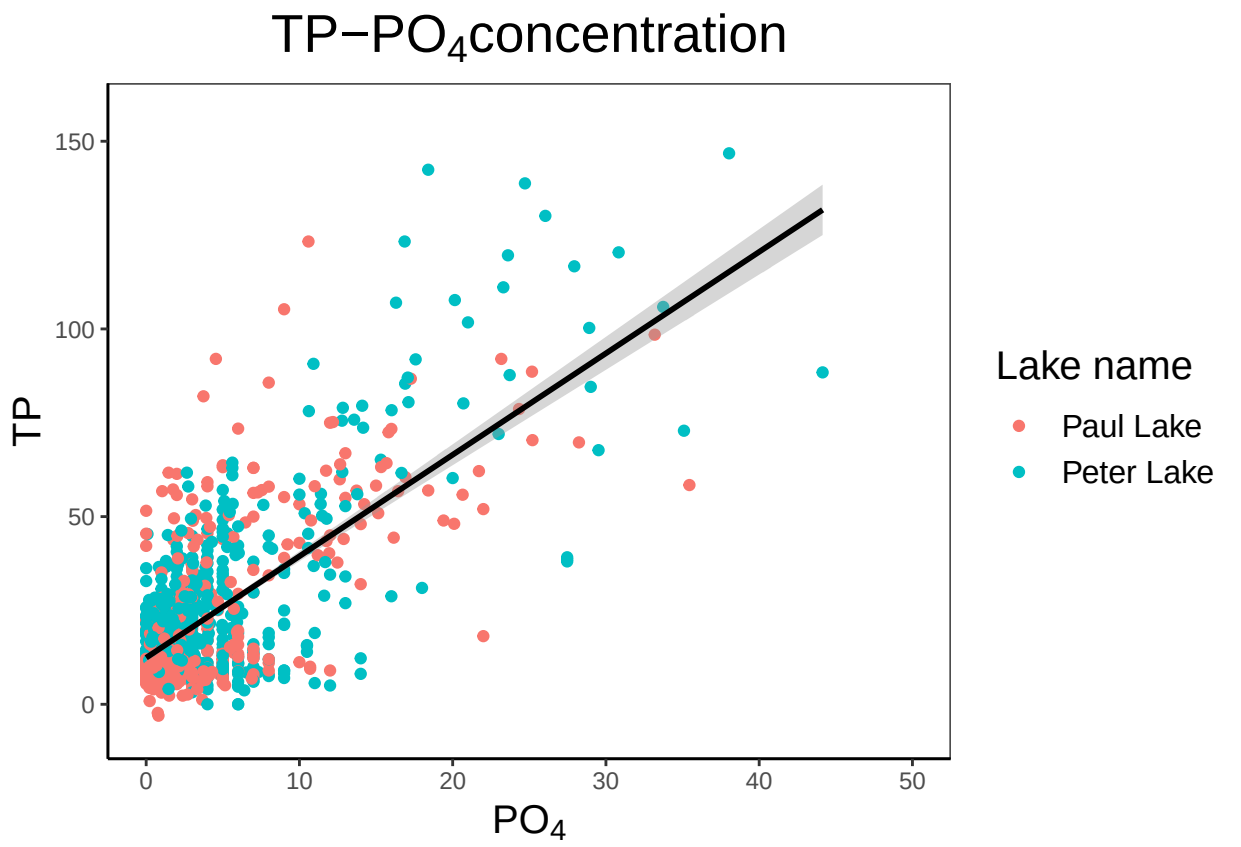
Hint: a good way to build graphs is to make them ugly first and then create more code to make them pretty.

4. [NTL-LTER] Plot total phosphorus by phosphate, with separate aesthetics for Peter and Paul lakes. Add a line of best fit and color it black.

```
#4
ggplot(PPchemphy,aes(x=po4,y=tp_ug,color=lakename)) +
  geom_point(size=1.5) +
  wang +
  xlim(c(0,50)) +
  labs(title = expression("TP-PO"[4]*"concentration"),x=expression("PO"[4]),y="TP",color="Lake name") +
  geom_smooth(aes(x=po4,y=tp_ug),method = lm,color="black")
```

```
## Warning: Removed 22310 rows containing non-finite values (stat_smooth).

## Warning: Removed 22310 rows containing missing values (geom_point).
```



5. [NTL-LTER] Plot nutrients by date for Peter Lake, with separate colors for each depth. Facet your graph by the nutrient type.

```
#5
peter <- filter(PPchemphy,lakename=="Peter Lake")
```

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

```
NTL.gathered <- gather(peter,"nutrient","concentration",tn_ug:po4)
NTL.gathered$sampledate <- as.Date(NTL.gathered$sampledate,"%Y-%m-%d")
library(RColorBrewer)
```

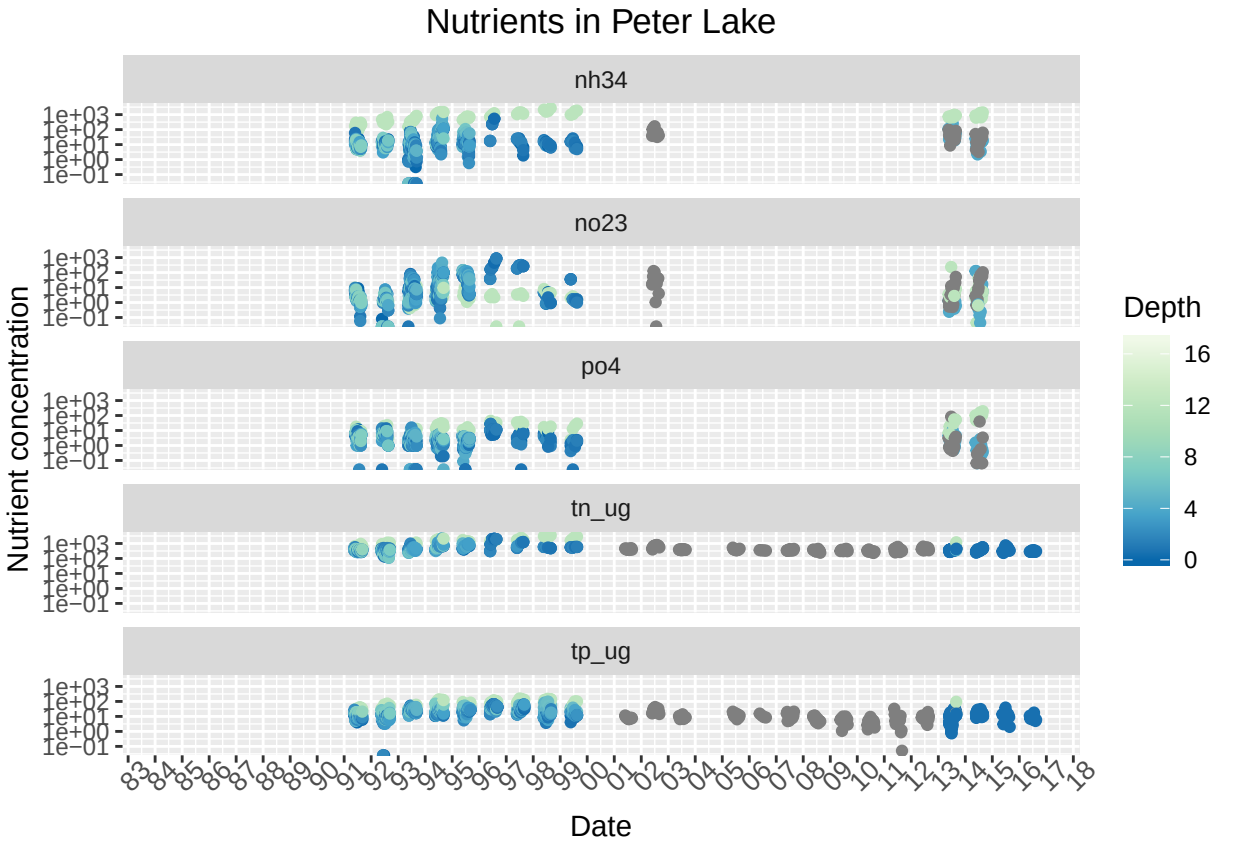
```
## Warning: package 'RColorBrewer' was built under R version 3.5.2
```

```
ggplot(NTL.gathered,aes(x=sampledate,y=concentration,color=depth)) +
  geom_point(size=1.5) +
  facet_wrap(vars(nutrient),nrow = 5) +
  scale_color_distiller("Depth",palette = "GnBu",direction = -1) +
  scale_x_date(name = "Date",date_breaks = "1 year",date_labels = "%y") +
  scale_y_log10("Nutrient concentration") +
  labs(title = "Nutrients in Peter Lake") +
  theme(axis.text.x = element_text(size=10,angle = 45),plot.title=element_text(hjust = 0.5))
```

```
## Warning in self$trans$transform(x): NaNs produced
```

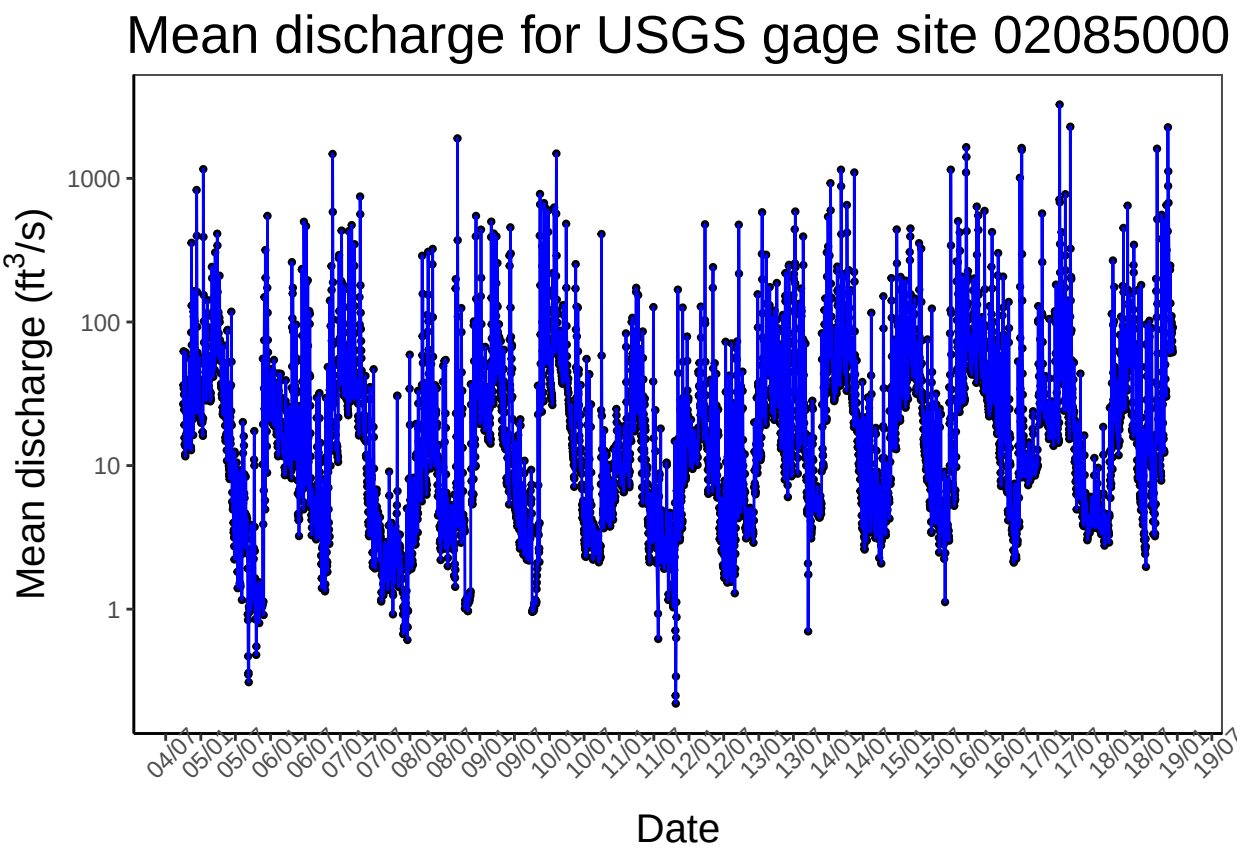
```
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## Warning: Removed 56671 rows containing missing values (geom_point).
```



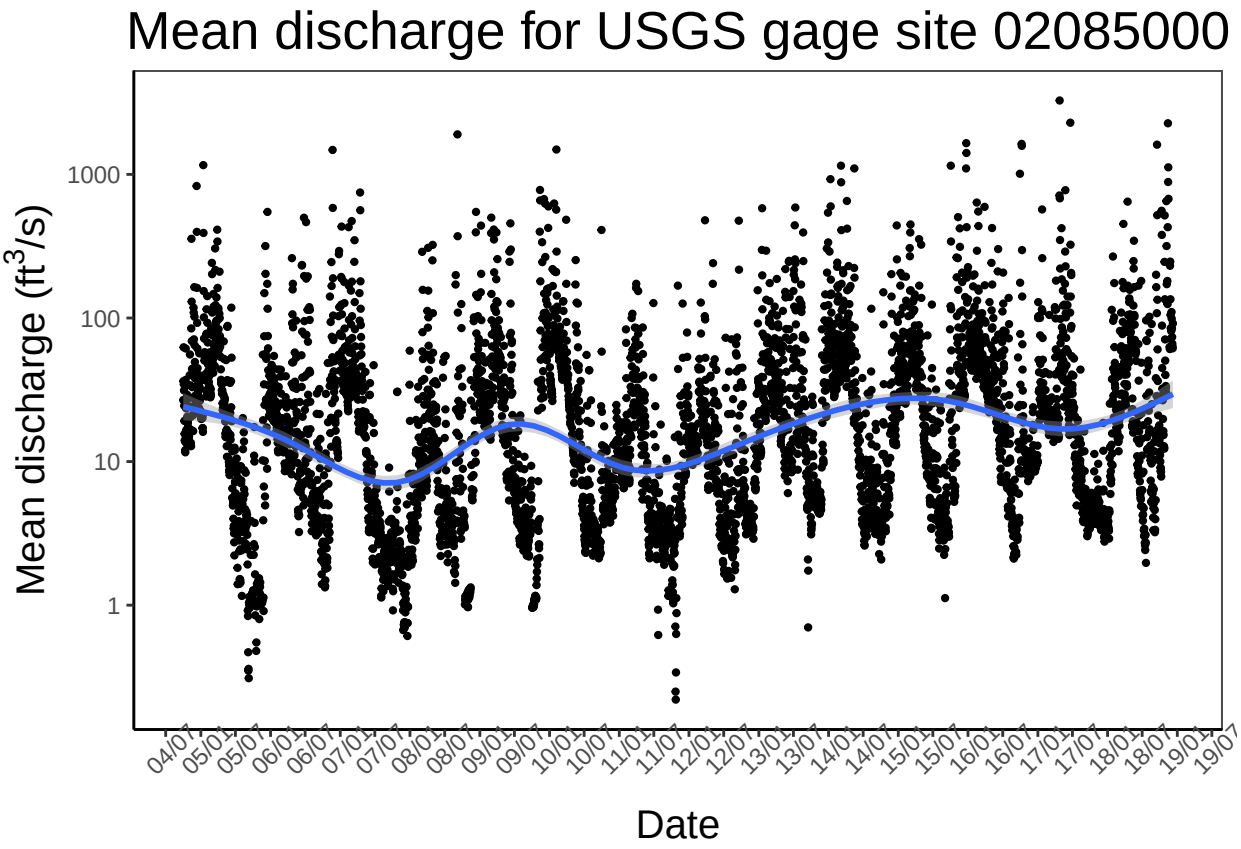
6. [USGS gauge] Plot discharge by date. Create two plots, one with the points connected with geom_line and one with the points connected with geom_smooth (hint: do not use method = "lm").

```
#6
#I'm going to plot the mean discharge (84936_00060_00003).
discharge <- filter(USGSgage,X84936_00060_00003>0)
ggplot(discharge,aes(x=datetime,y=X84936_00060_00003)) +
  geom_point(size=0.8,na.rm = T) +
  geom_line(color="Blue") +
  scale_x_date(name = "Date",date_breaks = "6 months",date_labels = "%y/%m") +
  scale_y_log10(expression("Mean discharge (ft"^"3"*"/s)")) +
  labs(title = "Mean discharge for USGS gage site 02085000") +
  wang +
  theme(axis.text.x = element_text(angle = 45))
```




```
ggplot(discharge,aes(x=datetime,y=X84936_00060_00003)) +
  geom_point(size=0.8,na.rm = T) +
  geom_smooth() +
  scale_x_date("Date",date_breaks = "6 months",date_labels = "%y/%m") +
  scale_y_log10(expression("Mean discharge (ft"^"3"*"/s)")) +
  labs(title = "Mean discharge for USGS gage site 02085000") +
  wang +
  theme(axis.text.x = element_text(angle = 45))
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

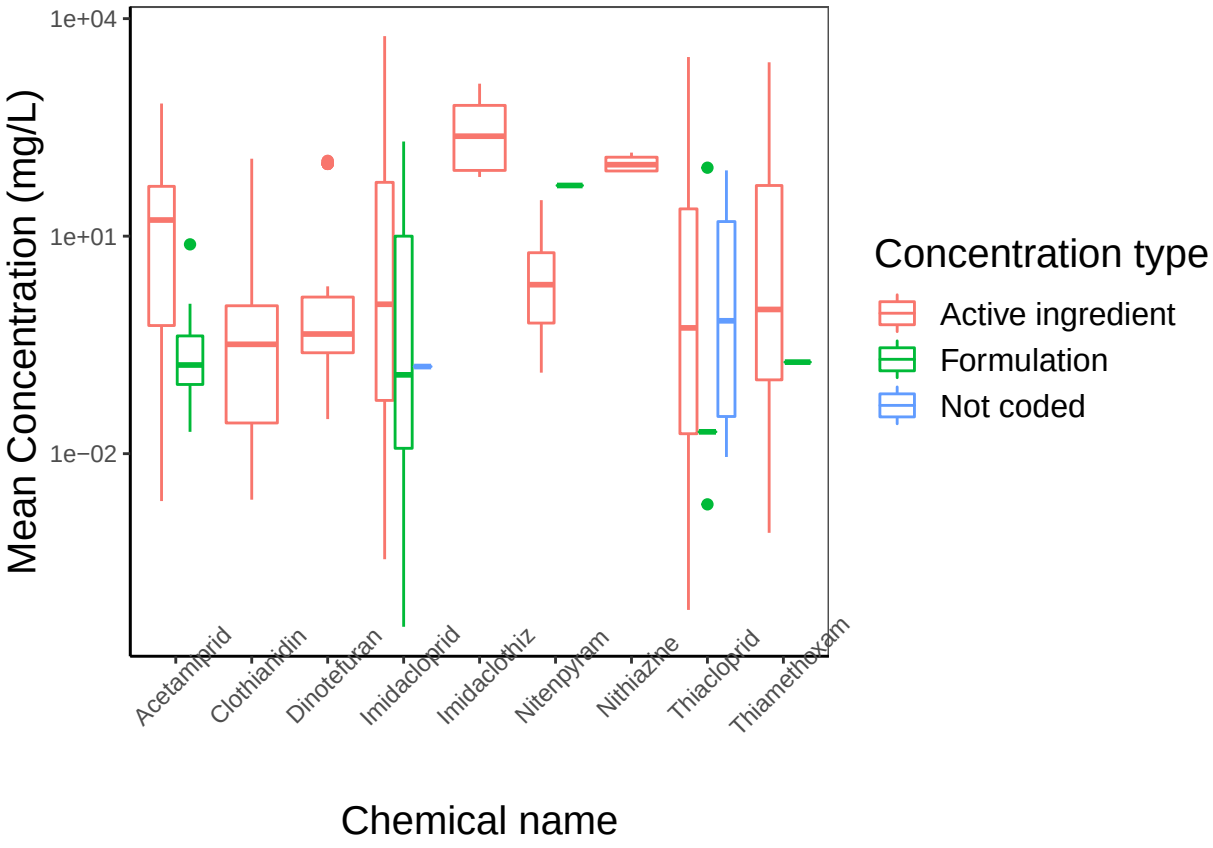


Question: How do these two types of lines affect your interpretation of the data?

Answer: The former one (geom_line) reflects the finest daily variation without any smoothing, while the latter (geom_smooth) shows the overall trend over time on a yearly temporal scale.

7. [ECOTOX Neonicotinoids] Plot the concentration, divided by chemical name. Choose a geom that accurately portrays the distribution of data points.

```
#7
attach(ecotox)
ecotox <- filter(ecotox,Conc..Units..Std.=="AI mg/L")
ecotox <- select(ecotox,Chemical.Name,Dur..Std.:Conc..Mean..Std.)
ggplot(ecotox,aes(x=Chemical.Name,y=Conc..Mean..Std.,color=Conc..Type)) +
  geom_boxplot() +
  scale_y_log10("Mean Concentration (mg/L)") + xlab("Chemical name") +
  wang +
  scale_color_discrete("Concentration type") +
  theme(axis.text.x = element_text(angle = 45))
```


Mean Concentration (mg/L)
1e+04
1e+01
1e−02
Acetamiprid
Clothianidin
Dinotefuran
Imidacloprid
Imidaclothiz
Nitenpyram
Nithiazine
Thiacloprid
Thiamethoxam
Chemical name
Concentration type
Active ingredient
Formulation
Not coded