Lab5_Standard_Curve_Lab

BillyMcLean

2024-09-20

```
ugL <- read_csv("stdcurvedata1.csv")
```

```
## Rows: 26 Columns: 2
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## dbl (2): ug_L_DIN, Area
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
SampleData <- read_csv("stdcurvedata.csv")
```

```
## Rows: 104 Columns: 6
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (2): Sample_Name, Analysis_Inj
## dbl (4): sample_order, Inj_Number, Area, Excluded
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
SampleData <- SampleData %>%
  filter(Excluded != 1)
```

Insert the graph of your standard curve. For full credit, please give your graph a title and label your axis approximately (including units as needed). Your graph should show the points of the standard curve (all that you chose to include) and the regression line fit to those points.

```
library(plotly)
```

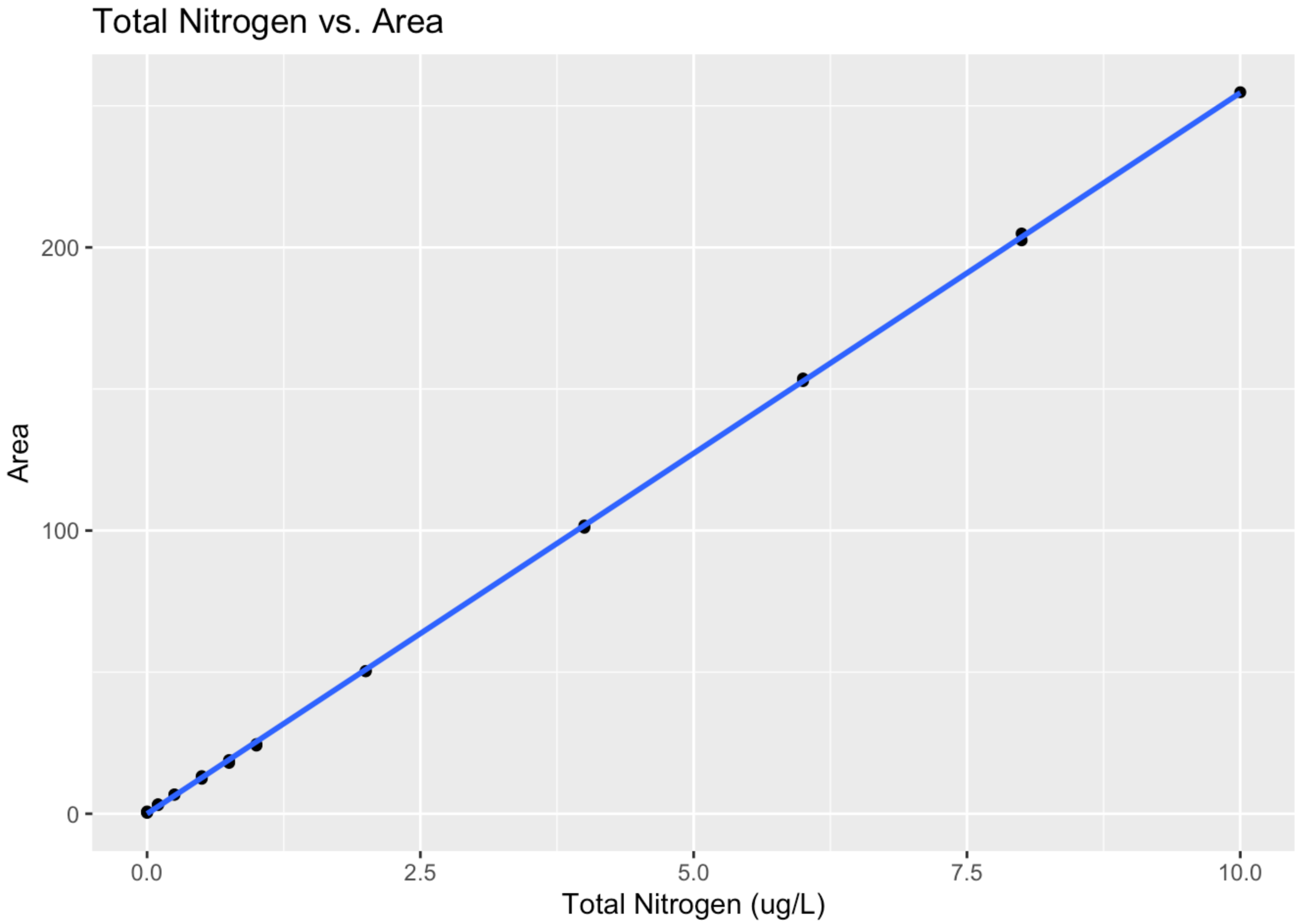
```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
TN <- ggplot(data = filter(ugL, Area != 249.1000), mapping = aes(x = ug_L_DIN, y = Are
a))+
  geom_point()+
  geom_smooth(method="lm")+
  labs(
    x = "Total Nitrogen (ug/L)",
    y = "Area",
    title = "Total Nitrogen vs. Area"
  )


TN
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



What is the equation of the standard curve?

```
model_data <- ugL %>%
  filter(Area != 249.1000)

AreaModel <- lm(ug_L_DIN ~ Area, model_data)
summary(AreaModel)
```

```
##
## Call:
## lm(formula = ug_L_DIN ~ Area, data = model_data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.047935 -0.021585 -0.007223  0.022731  0.053688
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.810e-04  7.365e-03   0.079    0.938
## Area        3.927e-02  7.792e-05 504.055   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0292 on 23 degrees of freedom
## Multiple R-squared:  0.9999, Adjusted R-squared:  0.9999
## F-statistic: 2.541e+05 on 1 and 23 DF,  p-value: < 2.2e-16
```

y = 0.03927x - 0.0005810

What is the R2 of your curve?

R-Squared = 0.9999

Did you remove any of the points from your curve? Why or why not?

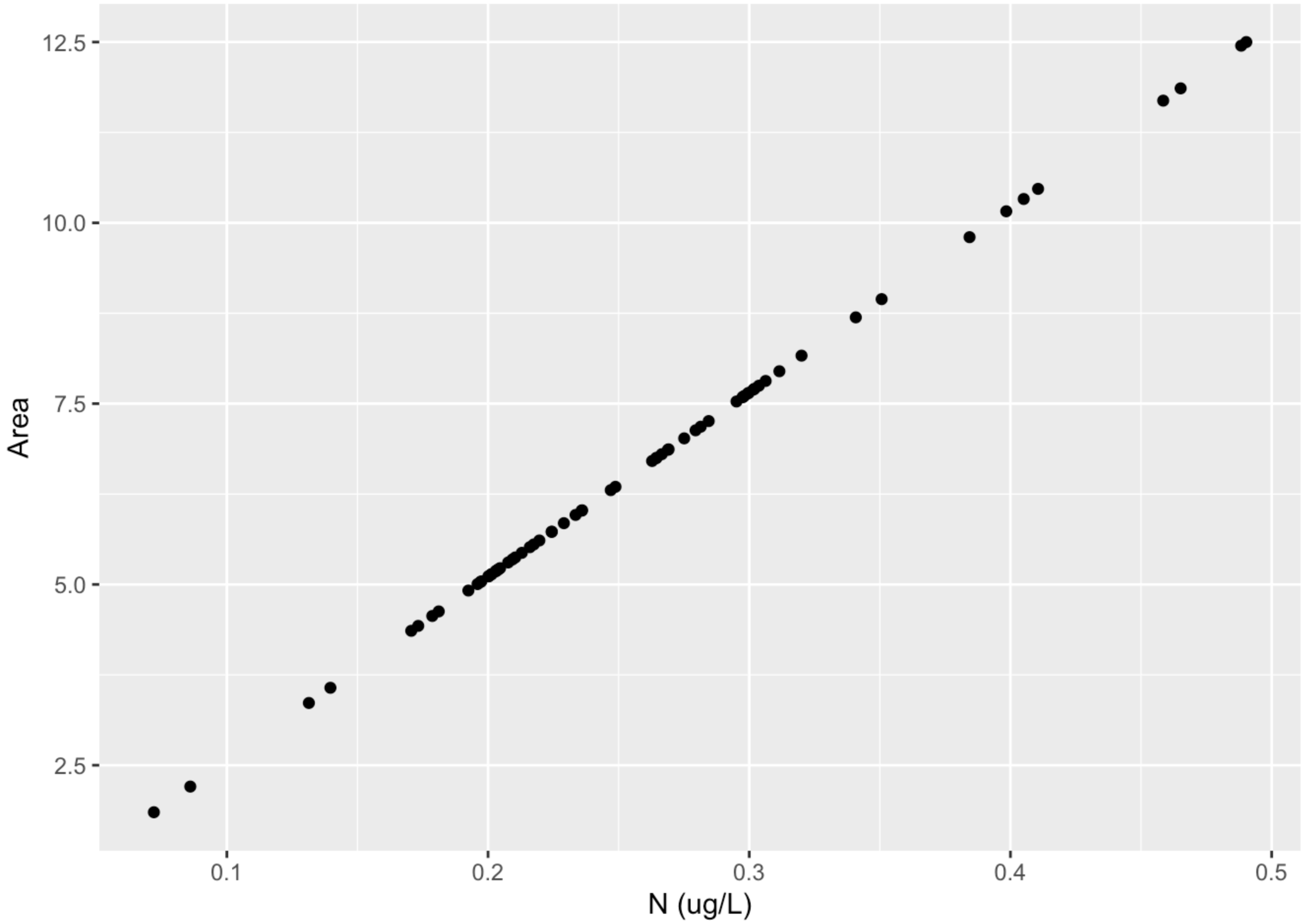
Yes, I removed the point where the Area value was 249.1000 because it appeared to be an outlier. Doing so resulted in a change in my R-squared value from 0.9998 to 0.9999

```
x <- SampleData$Area

SampleData <- SampleData %>%
  mutate(N = 0.03927*x - 0.0005810)
```

Were you able to determine the value of all of the samples? Why or why not?

Yes, all of the area values for our Sample Data fell within the standard curve dataset so I was able to accurately calculate the N values for our field data

```
#Filter out MQ and Check data
SampleData <- SampleData %>%
  filter(!Sample_Name %in% c("MQ", "check"))

ggplot(data = SampleData, mapping = aes(x = N, y = Area))+
  geom_point()+
  labs(x = "N (ug/L)",
       y = "Area")
```



What is the mean surface samples (across all groups)?

```
#Filter to just epilimnion "E" samples, leaving out samples that do not designate locati
on in the water column
Surface <- SampleData %>%
  filter(str_detect(Sample_Name, "-E-"))


MeanN_Surface <- mean(Surface$N)

MeanN_Surface
```

```
## [1] 0.2637103
```

What is the standard deviation of surface samples (across all groups)?

```
sd <- sd(Surface$N)

sd
```

```
## [1] 0.08642294
```

What is the mean N in deep water samples (across all groups)?

```
#Assuming this is the G#-H- series of samples
HypoSamples <- SampleData %>%
  filter(str_detect(Sample_Name, "-H-"))

MeanNHypo <- mean(HypoSamples$N)

MeanNHypo
```

```
## [1] 0.2677313
```

What is the standard deviation of deep water samples (across all groups)?

```
# sd_AreaHypo <- sd(HypoSamples$Area)
#
# sd_AreaHypo

sd_NHypo <- sd(HypoSamples$N)

sd_NHypo
```

```
## [1] 0.08776034
```

How do your groups samples compare to the other groups’ samples? Please provide specific values [type them below].

```
#Sum total N by Groups
GroupTotals <- SampleData %>%
  separate(Sample_Name, into = c("Group_Number", "Sample_Number"), sep = "-", extra = "merge", fill = "right")



GroupTotals1 <- GroupTotals %>%
  group_by(Group_Number) %>%
  summarise(Total_N = sum(N))

print(GroupTotals1)
```

```
## # A tibble: 9 × 2
##   Group_Number Total_N
##   <chr>          <dbl>
## 1 G1              1.00
## 2 G10             1.77
## 3 G2              2.56
## 4 G3              1.92
## 5 G4              2.65
## 6 G5              1.78
## 7 G6              2.32
## 8 G7              1.92
## 9 G8              1.74
```

```
#Average Total N
MeanTotalN <- GroupTotals1 %>%
  filter(!Group_Number %in% c("check", "MQ"))

MeanTotalN <- mean(MeanTotalN$Total_N)

print(MeanTotalN)
```

```
## [1] 1.96117
```

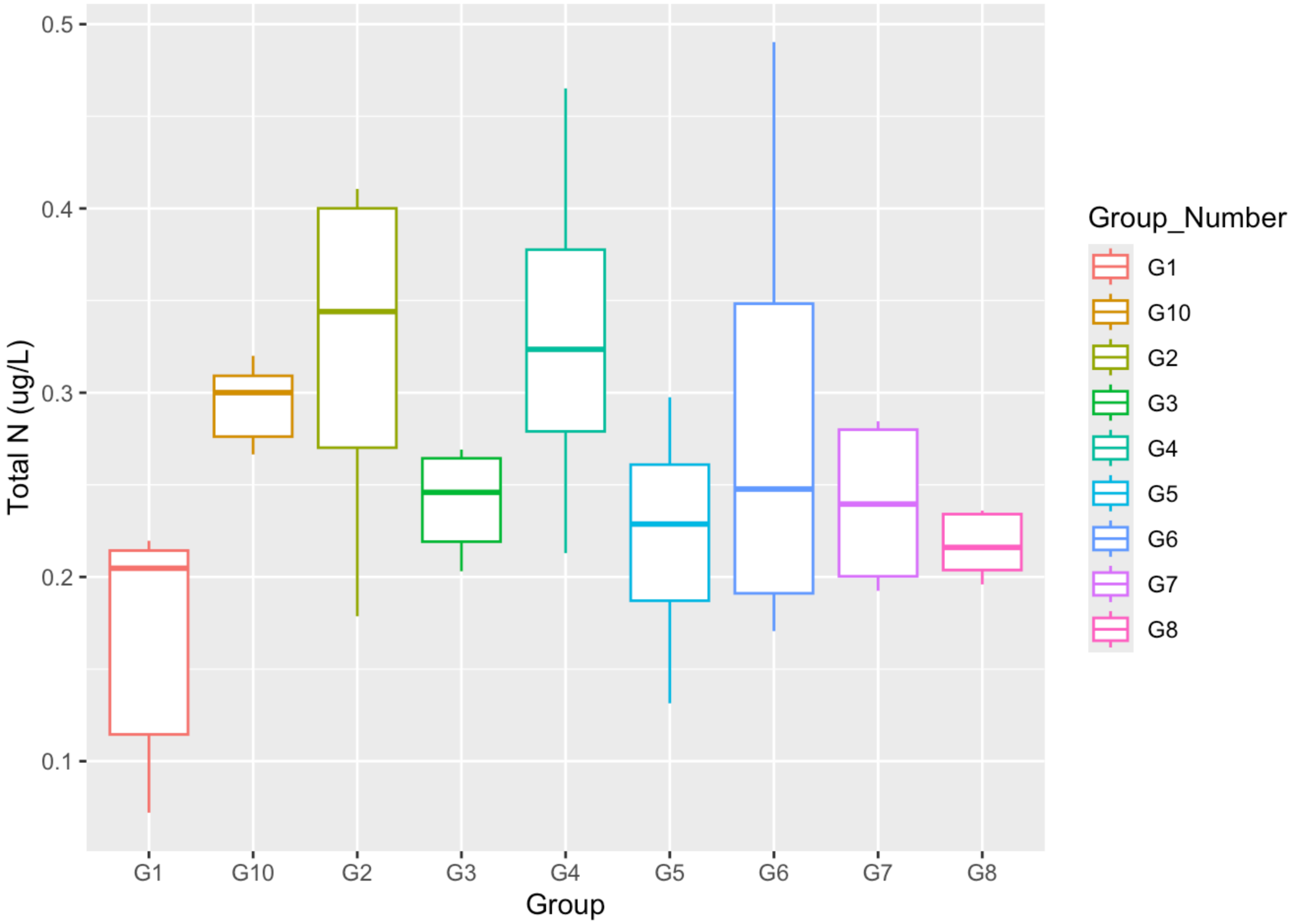
```
Group6 <- SampleData %>%
  filter(str_detect(Sample_Name,"G6"))
Group6_TotalN <- sum(Group6$N)

Group6_TotalN
```

```
## [1] 2.316602
```

Group 6 total N is ~2.32. This is higher than the average total N across all groups (excluding “check” and “MQ”), which is 1.96.

This could be linked to differing sample locations. We stopped at two separate locations on the field trip and Group 6 sampled using the Van Dorn at the first location while other groups sampled at the second.

```
ggplot(data = GroupTotals)+
  geom_boxplot(mapping = aes(x = Group_Number, y = N, group = Group_Number, color = Grou
p_Number))+
  labs(x = "Group",
       y = "Total N (ug/L)")
```


Total N (ug/L)
0.5
0.4
0.3
0.2
0.1
Group
G1
G10
G2
G3
G4
G5
G6
G7
G8
Group_Number
G1
G10
G2
G3
G4
G5
G6
G7
G8