Data analysis challenge

Peter Thorpe

26th Oct 2022

1 the problem

Data recived however, missing the coloumn names - data from Antartica.

2 time series data

105032 obervations recorded in 5 different ways . . . lets try and understand the properties: problems, trends corrolation . . . and so on.

Load the library needed

```
library(knitr)
```

```
Warning: package 'knitr' was built under R version 3.6.3
```

```
library(tidyr)
```

```
Warning: package 'tidyr' was built under R version 3.6.3
```

```
library(dplyr)
```

```
Warning: package 'dplyr' was built under R version 3.6.3
```

```
library(ggplot2)
library(readr)
```

```
Warning: package 'readr' was built under R version 3.6.3
```

```
library(magrittr)
```

```
Warning: package 'magrittr' was built under R version 3.6.3
```

```
library(devtools)
```

```
Warning: package 'usethis' was built under R version 3.6.3
```

```
# install_github('vqv/ggbiplot')
library(ggbiplot)
```

```
Warning: package 'scales' was built under R version 3.6.3
```

```
library(Hmisc)
```

```
Warning: package 'Formula' was built under R version 3.6.3
```

```
library(corrplot)
```

Load the data

counts were already generated using salmon and counts.matrix generated using trinity.

```
setwd("C:/Users/pjt6/Documents/DAG_challenge")

# check it
getwd()
```

```
[1] "C:/Users/pjt6/Documents/DAG_challenge"
```

The counts data in contained in the counts.matrix, each gene has a digital count per condition/ rep

```
# load in the data
time_series_data <- read.table("data/data.csv", sep = ",", header = TRUE, row.names = NULL)


summary(time_series_data)
```

```
       A                B                 C                 D         
 Min.   :-7.600   Min.   : 0.0000   Min.   :   0.00   Min.   : 963.8  
 1st Qu.: 5.300   1st Qu.: 0.0000   1st Qu.:   0.00   1st Qu.: 997.4  
 Median : 9.200   Median : 0.0000   Median :   5.00   Median :1007.4  
 Mean   : 9.115   Mean   : 0.8566   Mean   :  80.18   Mean   :1006.1  
 3rd Qu.:13.200   3rd Qu.: 1.0000   3rd Qu.:  80.00   3rd Qu.:1016.0  
 Max.   :27.300   Max.   :13.0000   Max.   :1098.00   Max.   :1036.4  
       E                id        
 Min.   : 34.00   Min.   :     1  
 1st Qu.: 80.00   1st Qu.: 26259  
 Median : 89.00   Median : 52517  
 Mean   : 86.15   Mean   : 52517  
 3rd Qu.: 96.00   3rd Qu.: 78774  
 Max.   :100.00   Max.   :105032  
```

have a quick look at the data:

```
head(time_series_data)
```

```
     A B C      D  E id
1 -1.1 0 0 1027.7 93  1
2 -1.1 0 0 1027.7 93  2
3 -1.1 0 0 1027.8 94  3
4 -1.1 0 0 1027.8 93  4
5 -1.1 0 0 1027.8 92  5
6 -1.1 0 0 1027.8 92  6
```

cleanup if required

```
# Using na.omit and assigning to another data frame
time_series_data_without_NAs <- na.omit(time_series_data)
```

check for NA in the dataset.

```
# Checking the dimensions with the dim() function (rows then columns)
dim(time_series_data)
```

```
[1] 105032      6
```

```
dim(time_series_data_without_NAs)
```

```
[1] 105032      6
```

There appears to be no NAs in the data. now check for duplicated rows.

```
# duplicated(time_series_data) # dont run this!
sum(duplicated(time_series_data))
```

```
[1] 0
```

there are no duplicated rows.

corrolation matrix

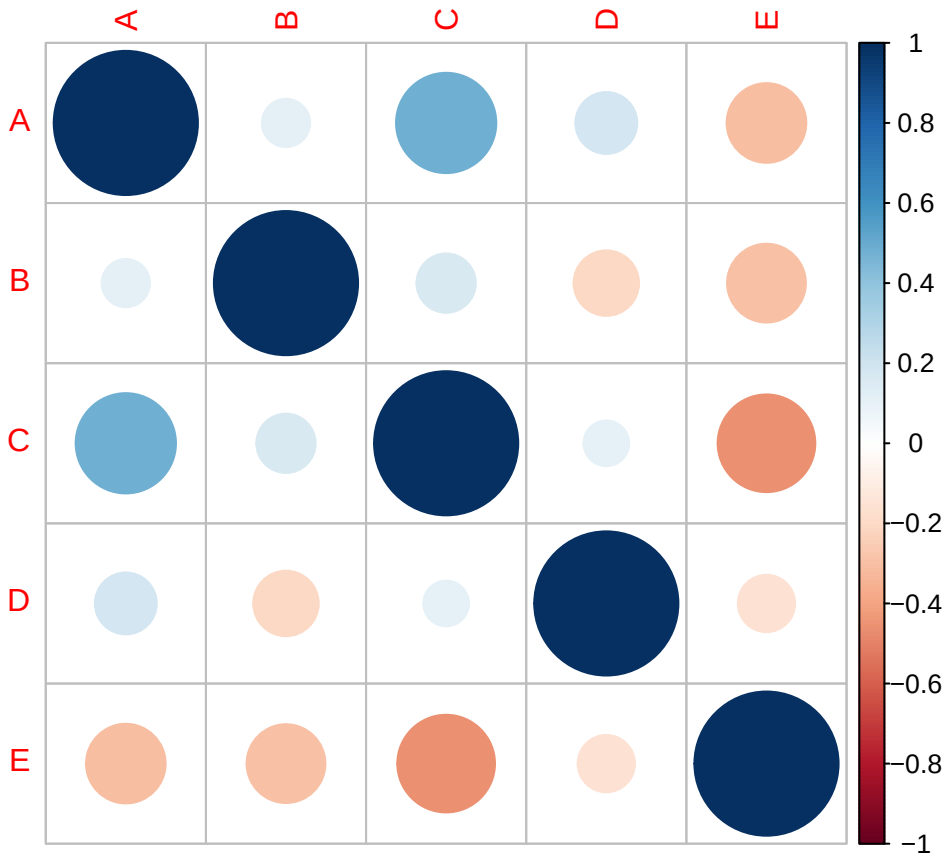
lets have a look at some corrolation matrix between these

```
# get the first 5 cols.
time_series_data_w_last_clo <- time_series_data[, c(1, 2, 3, 4, 5)]

# get a corrolation matrix with P values
time_series.rcorr = rcorr(as.matrix(time_series_data_w_last_clo))

# see the results
time_series.rcorr
```

```
      A     B     C     D     E
A  1.00  0.11  0.46  0.21 -0.37
B  0.11  1.00  0.13 -0.22 -0.26
C  0.46  0.13  1.00  0.13 -0.56
D  0.21 -0.22  0.13  1.00 -0.16
E -0.37 -0.26 -0.56 -0.16  1.00

n= 105032


P
  A  B  C  D  E
A     0  0  0  0
B  0     0  0  0
C  0  0     0  0
D  0  0  0     0
E  0  0  0  0
```

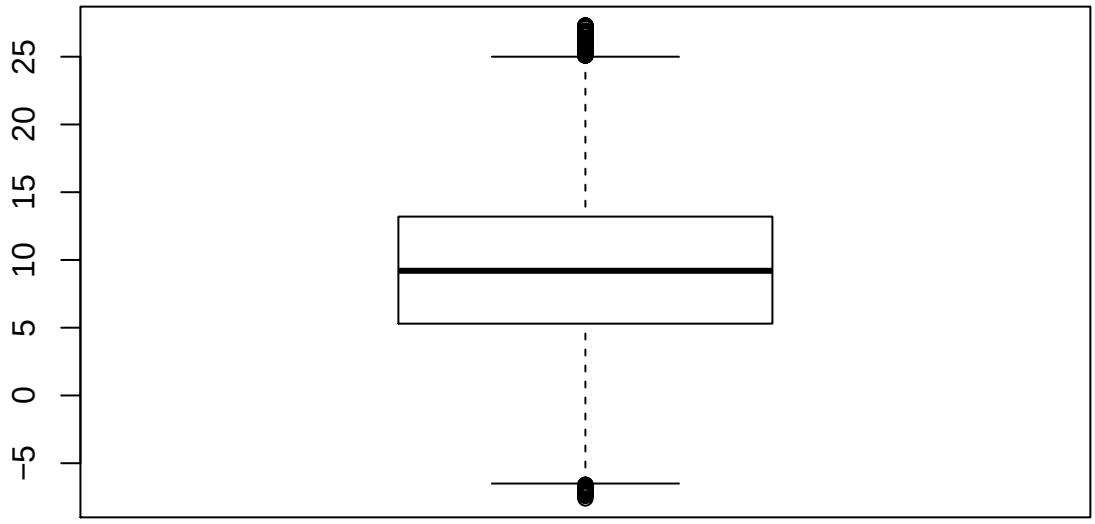
none of the corrolations are significant, see plot below. However, there is some loose corrolation.

```
time_series.cor = cor(time_series_data_w_last_clo, method = c("spearman"))

corrplot(as.matrix(time_series.cor))
```



Lets start by looking at colA

```
# this doesnt help
boxplot(time_series_data$A)
```

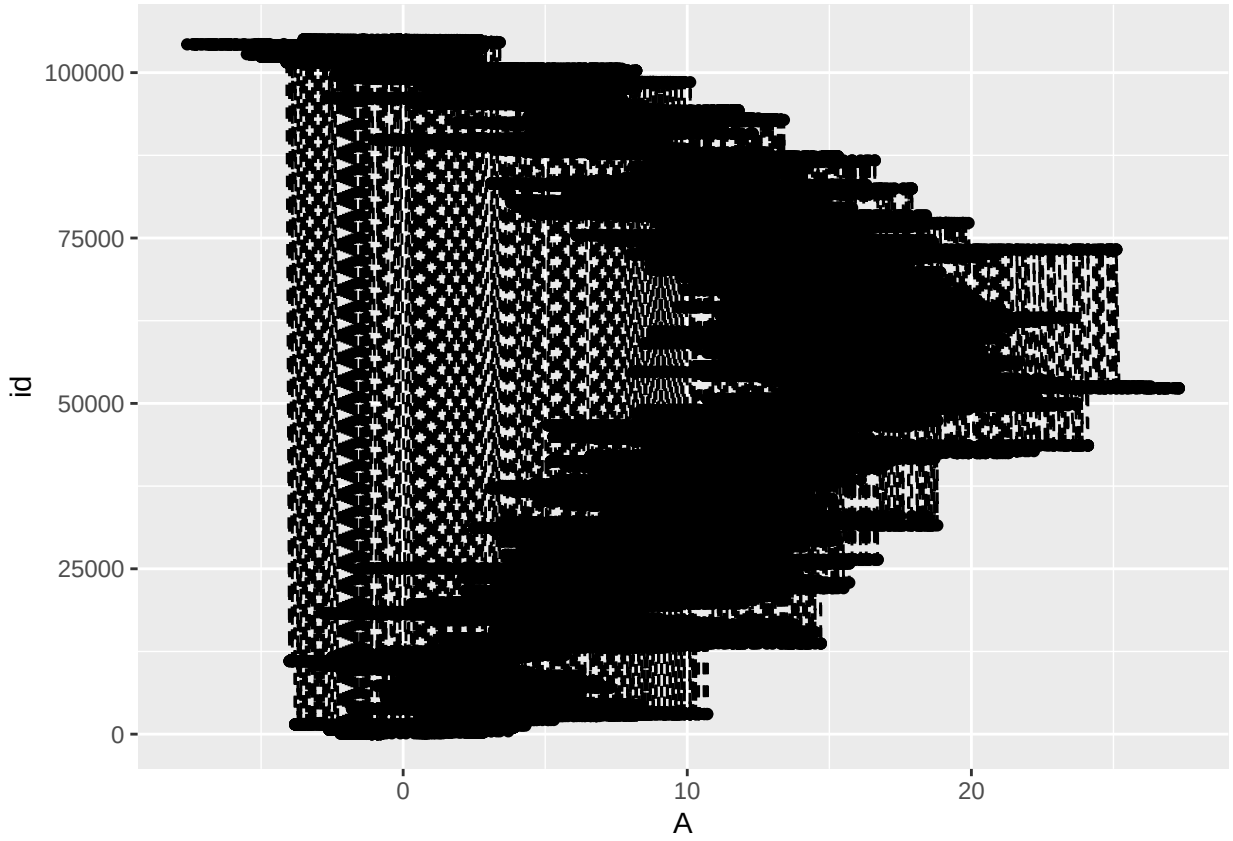


lets do a PCA to see how the variables separate out.

```
time_series_data.pca <- prcomp(time_series_data[, c(1, 2, 3, 4, 5)], center = TRUE,
    scale. = TRUE)


# ggbiplot(time_series_data.pca) this wont install and run on my windows
# laptop!

############################################## library/3.6/gtable/R/gtable.rdb'
############################################## is corrupt
```

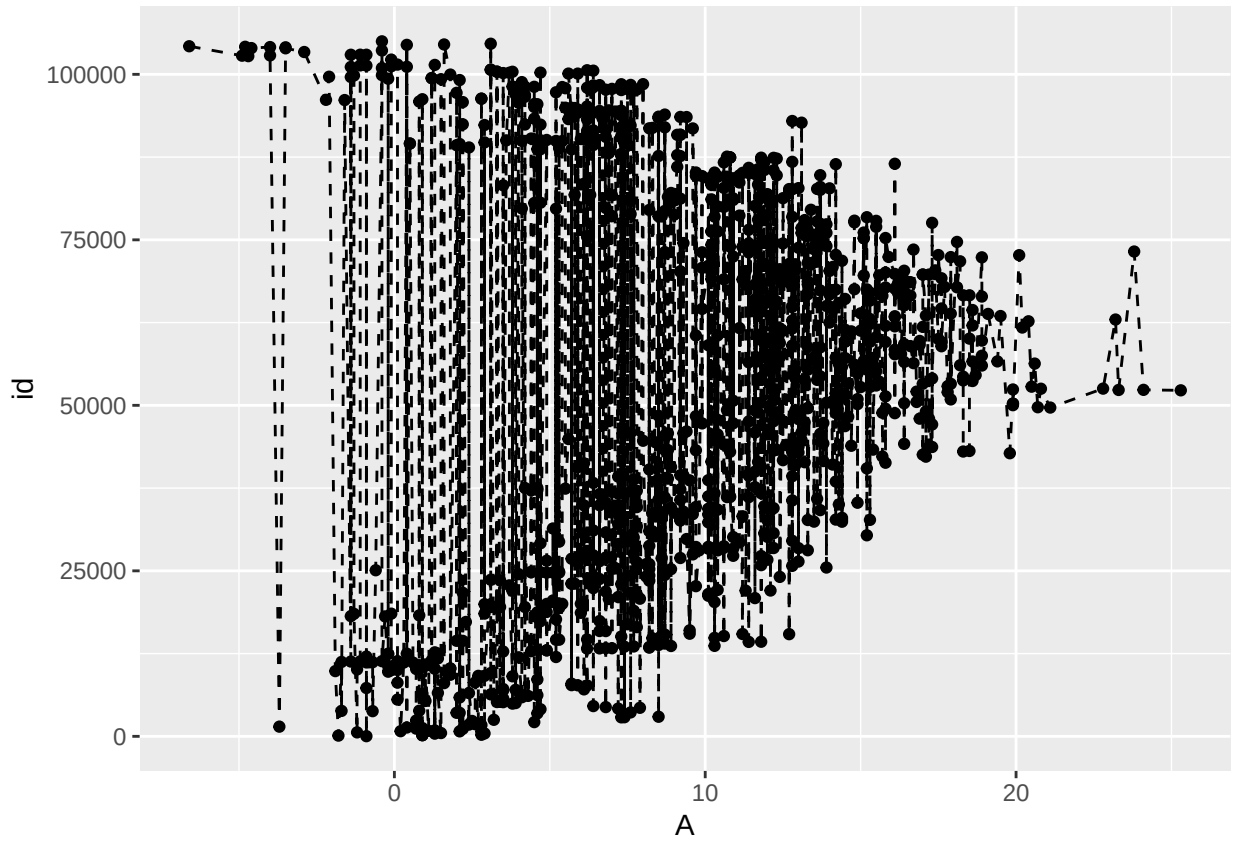
look at a quick plot A vs ID

```
ggplot(data = time_series_data, aes(x = A, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```



Ok, so this is not a nice plot. Lets randomly subsample this to see if we can see any better trends.

```
time_series_data_subsample <- time_series_data[sample(nrow(time_series_data), 1000),
    ]

ggplot(data = time_series_data_subsample, aes(x = A, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```



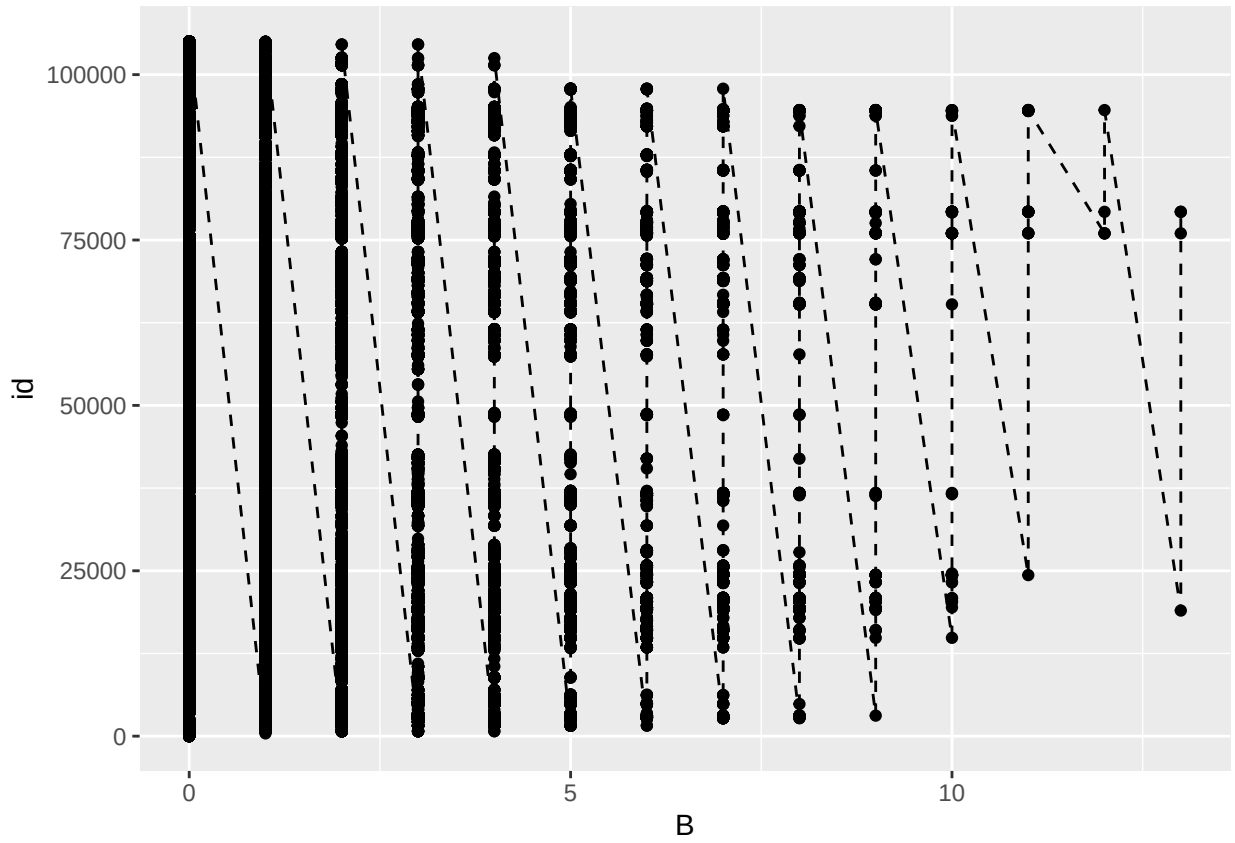
I dont know if there may be multiple peaks in this data set. So I will try a Fourier Transform. I know from my NMR days that I should be able to see peaks if there are any in the data here, but my code is not working.

```
# apply(time_series_data, 2,function(x) fft(as.numeric(x)))

# A_fft <- as.data.frame(fft(as.numeric(time_series_data$A)))

# A_fft <- unlist(A_fft)


# ggplot(data = A_fft, aes(x = A_fft, y = time_series_data$id)) +
# geom_line(linetype = 'dashed')+ geom_point()
```

3 LOOK AT COL B

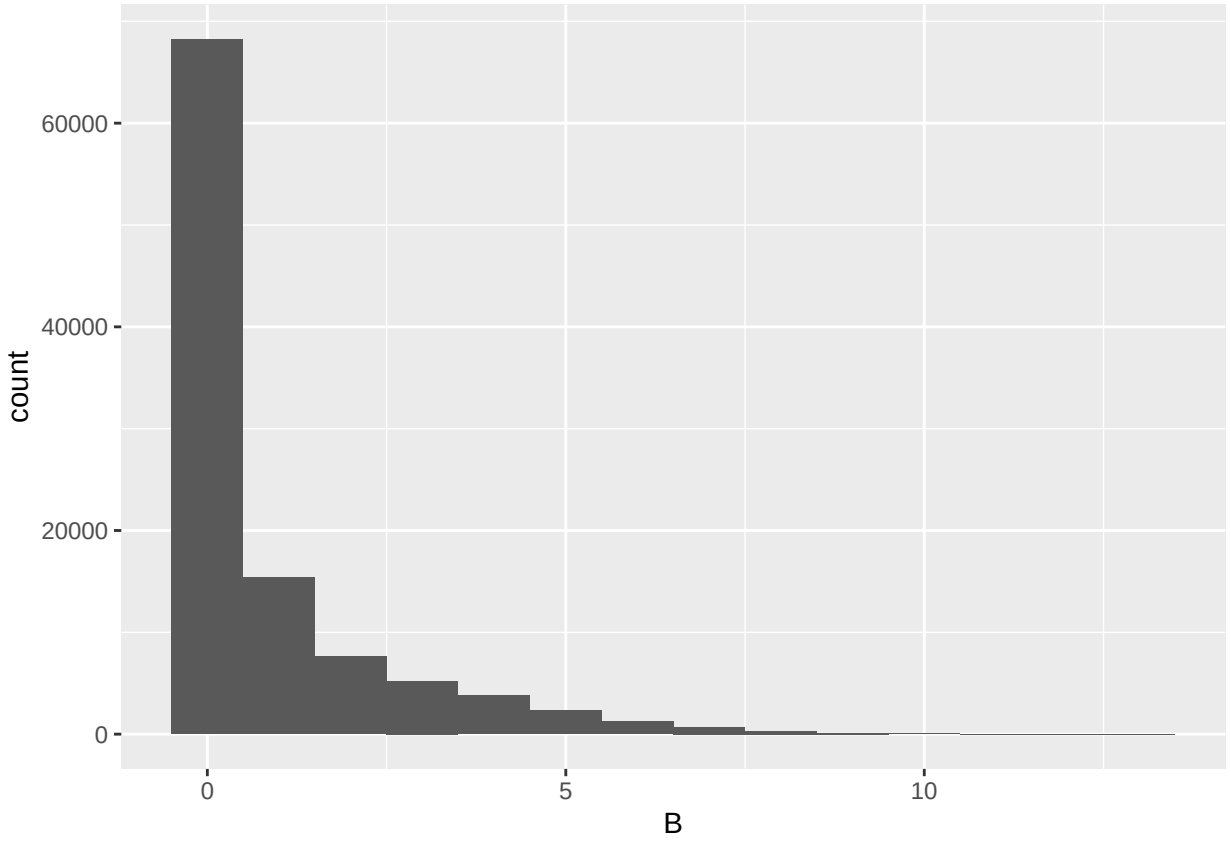
look at a quick plot B vs ID

```
ggplot(data = time_series_data, aes(x = B, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```



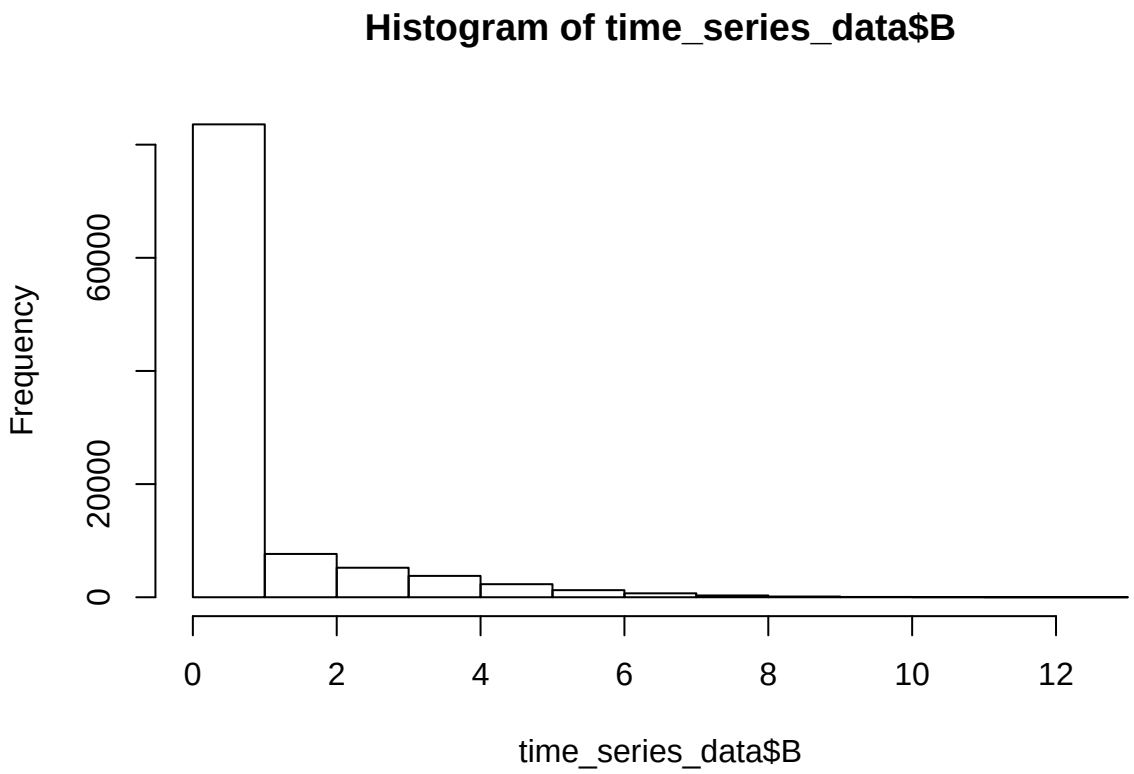
That doesn thelp - lets try a histogram

```
ggplot(data = time_series_data, aes(x = B)) + geom_histogram(binwidth = 1)
```



Ok so we have a poisson distribution. These could be counts of observations per time point.

```
# Extract histogram information
hist_vec <- hist(time_series_data$B)
```



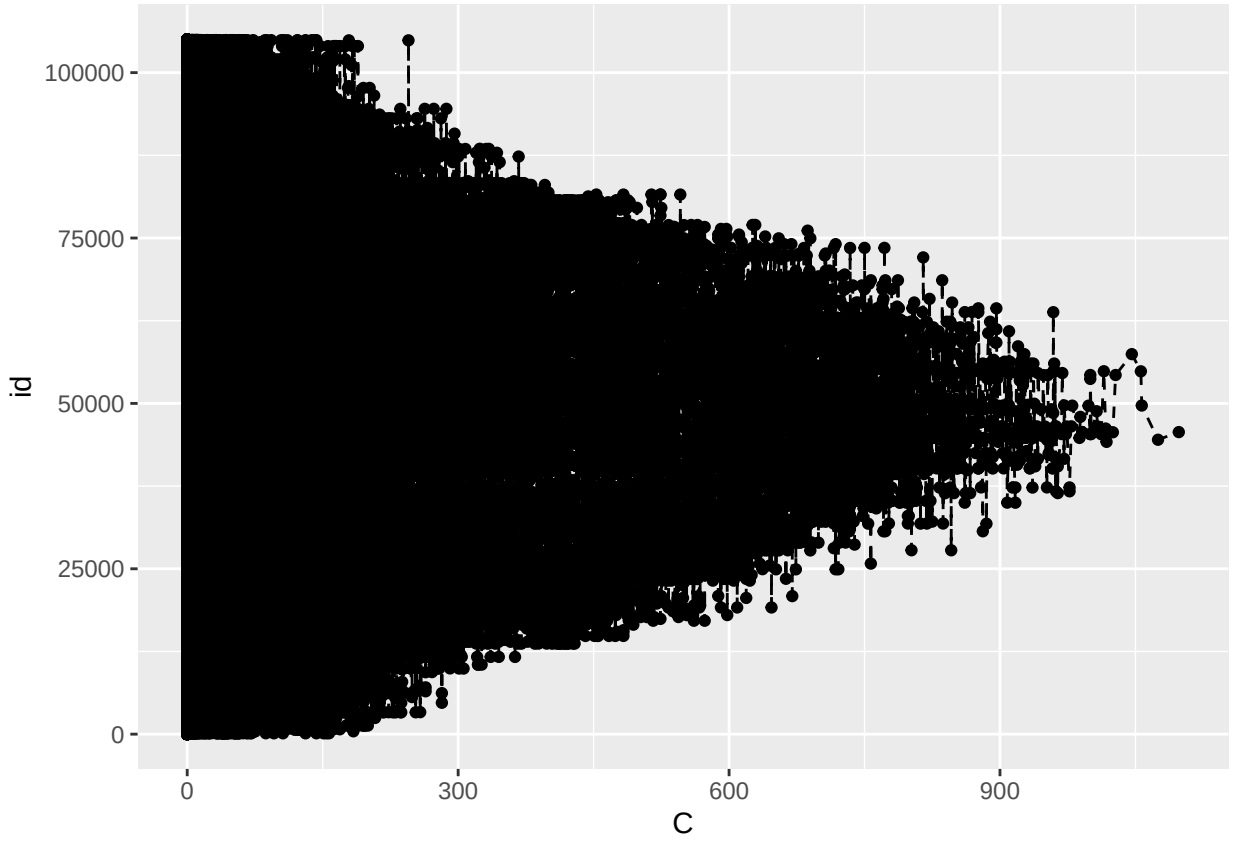

```
# Store histogram counts in frequency
frequency <- hist_vec$counts

frequency
```

```
 [1] 83590  7655  5220  3774  2312  1256   694   318   128    52    24     5
[13]     4
```

3 LOOK AT COL C

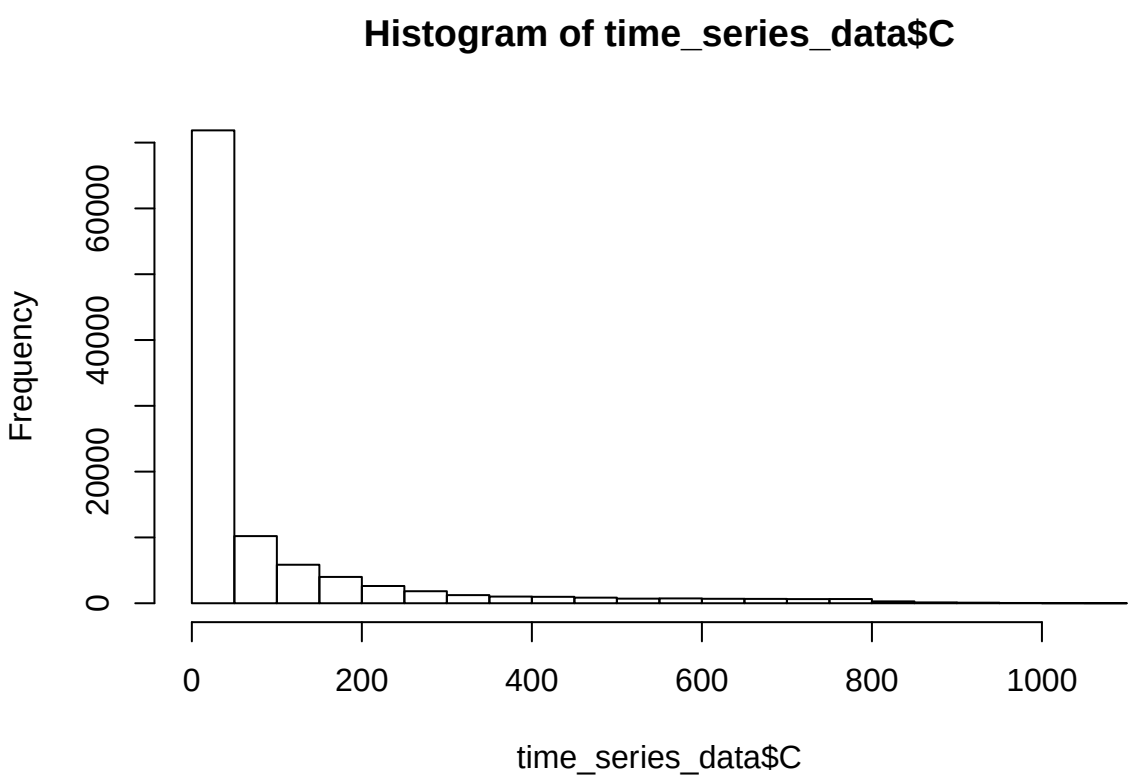
look at a quick plot B vs ID

```
ggplot(data = time_series_data, aes(x = C, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```



Ok, lets try a histogram of this.

```
# Extract histogram information
hist_vec <- hist(time_series_data$C)
```

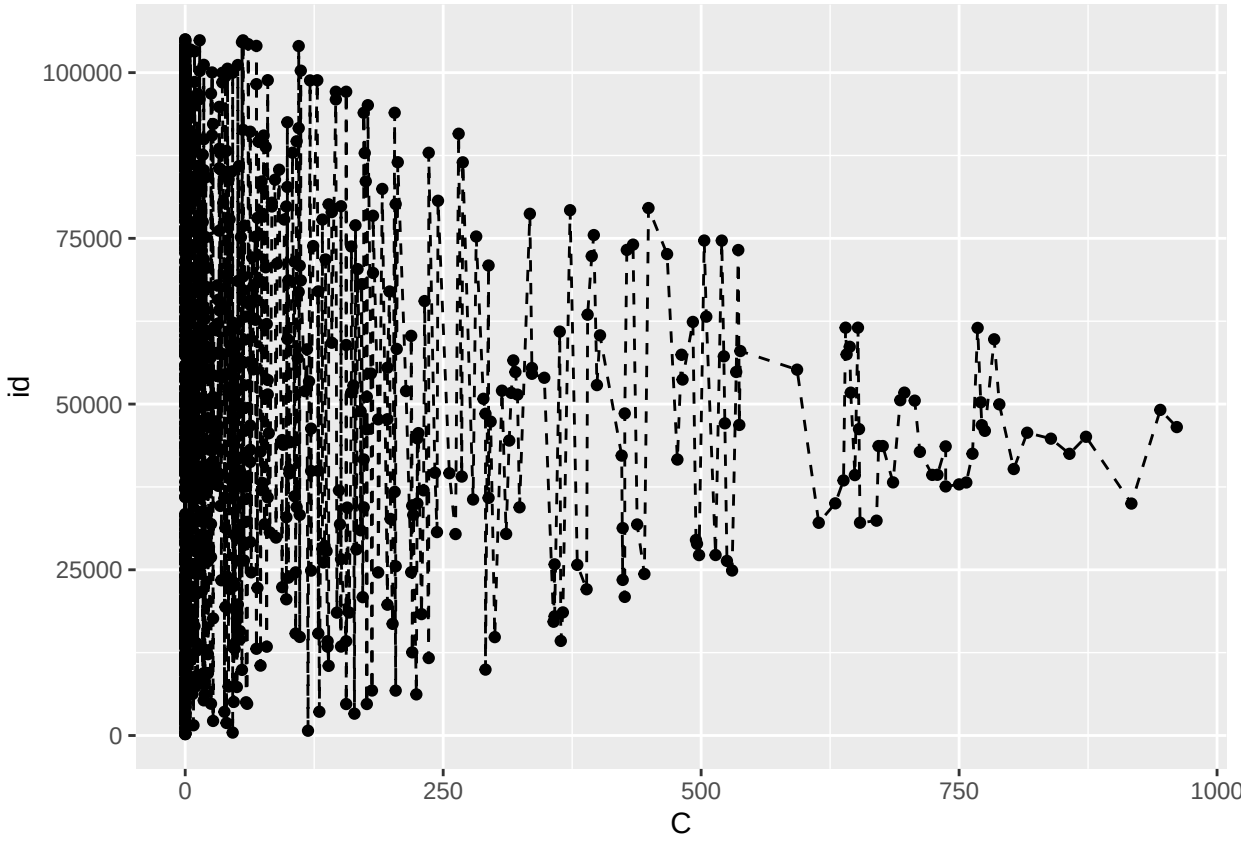



```
# Store histogram counts in frequency
frequency <- hist_vec$counts

frequency
```

```
 [1] 71862 10203  5853  3998  2621  1829  1234  1013   976   850   700   732
[13]   678   660   632   639   284   125    87    41    11     4
```

Lets randomly subsample this to see if we can see any better trends. (the number to subsample by could be better chosen)

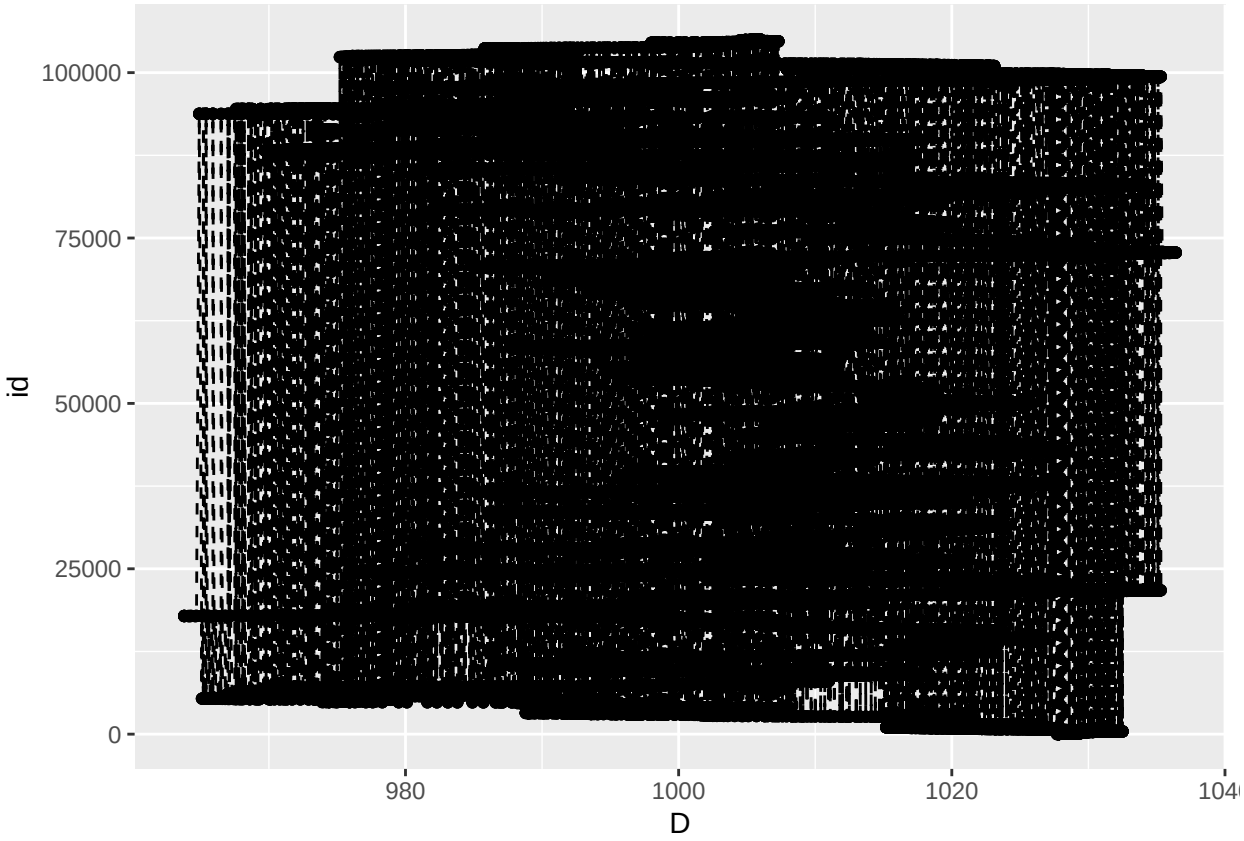
```
time_series_data_subsample <- time_series_data[sample(nrow(time_series_data), 1000),
    ]

ggplot(data = time_series_data_subsample, aes(x = C, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```



5 LOOK AT COL D

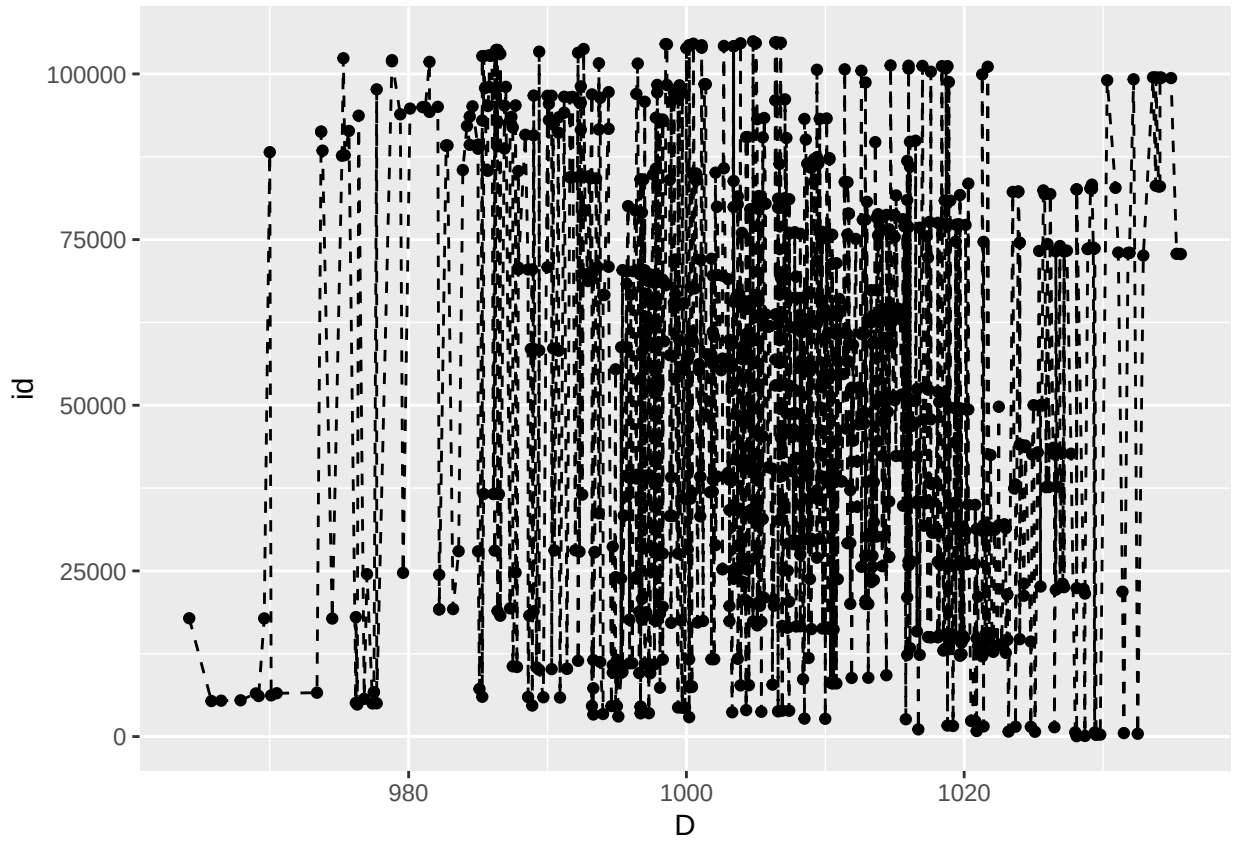
look at a quick plot D vs ID

```
ggplot(data = time_series_data, aes(x = D, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```

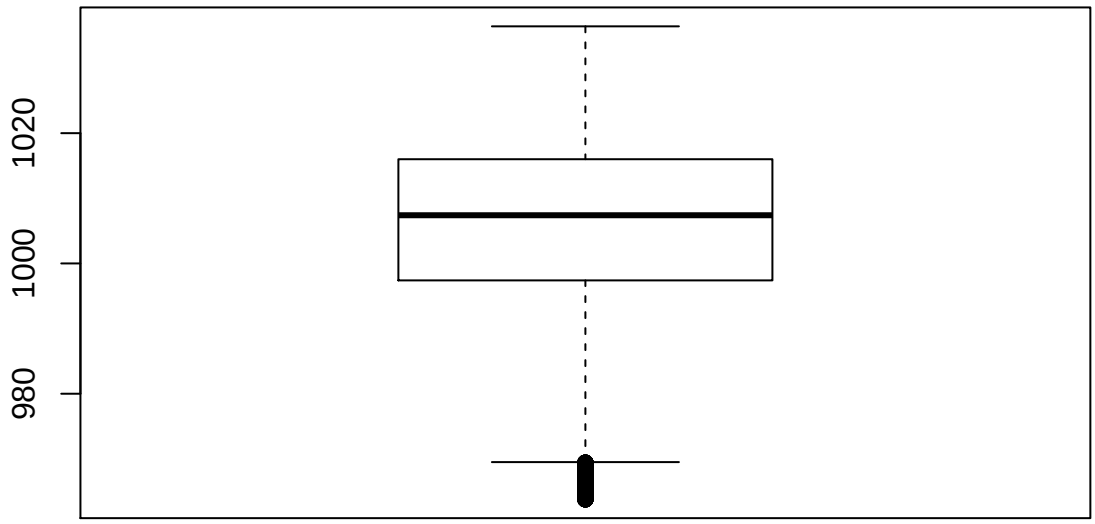


Lets randomly subsample this to see if we can see any better trends. (the number to subsample by could be better chosen)

```
time_series_data_subsample <- time_series_data[sample(nrow(time_series_data), 1000),
    ]

ggplot(data = time_series_data_subsample, aes(x = D, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```

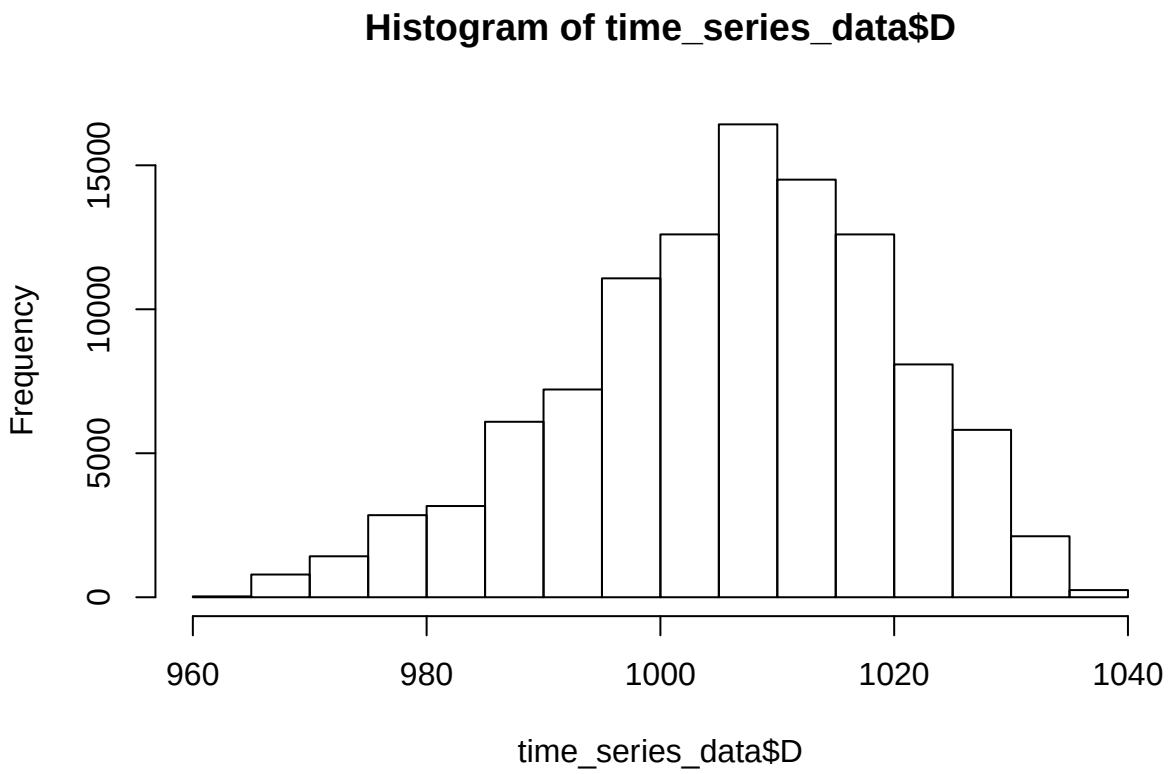



```
# this doesnt help
boxplot(time_series_data$D)
```



Ok, lets try a histogram of this.

```
# Extract histogram information
hist_vec <- hist(time_series_data$D)
```



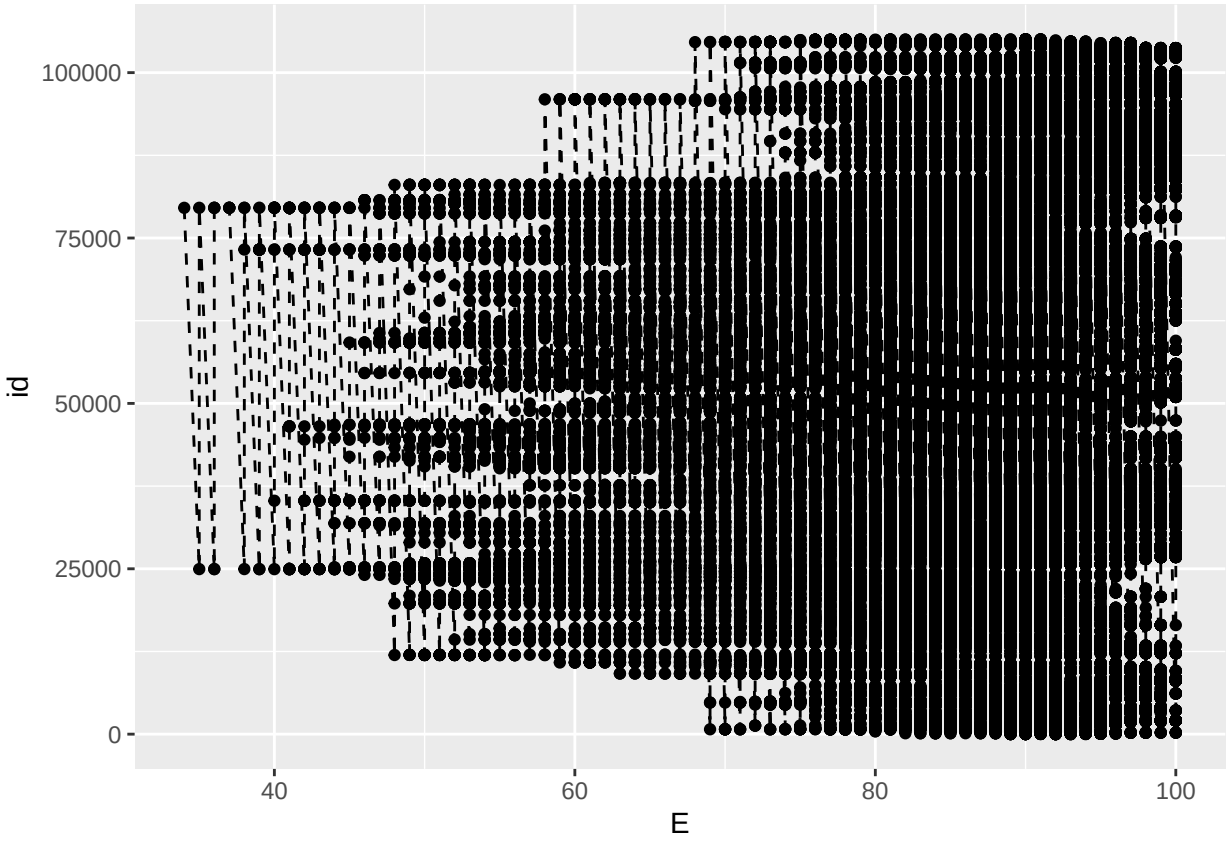

```
# Store histogram counts in frequency
frequency <- hist_vec$counts

frequency
```

```
 [1]    30   789  1422  2850  3168  6093  7215 11075 12601 16422 14502 12600
[13]  8085  5813  2118   249
```

6 LOOK AT COL E

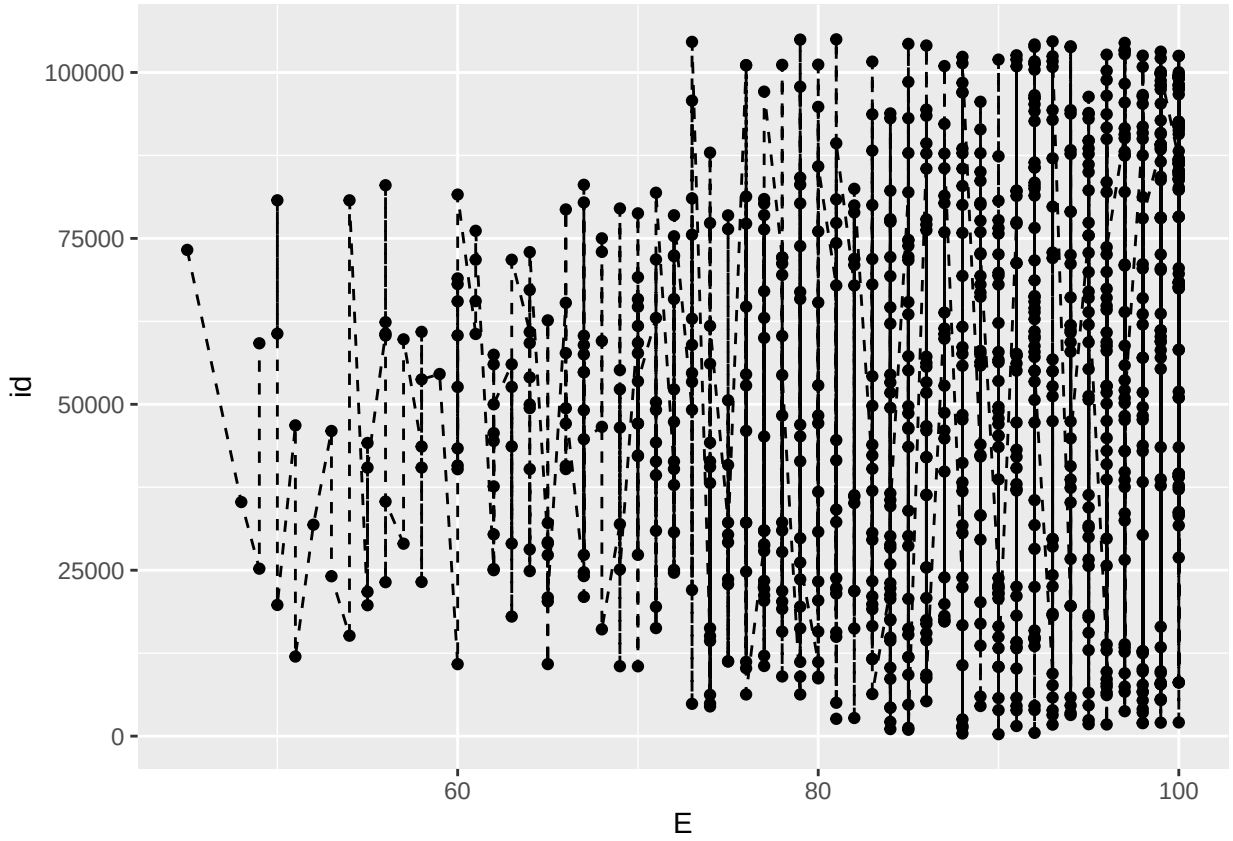
look at a quick plot E vs ID

```
ggplot(data = time_series_data, aes(x = E, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```



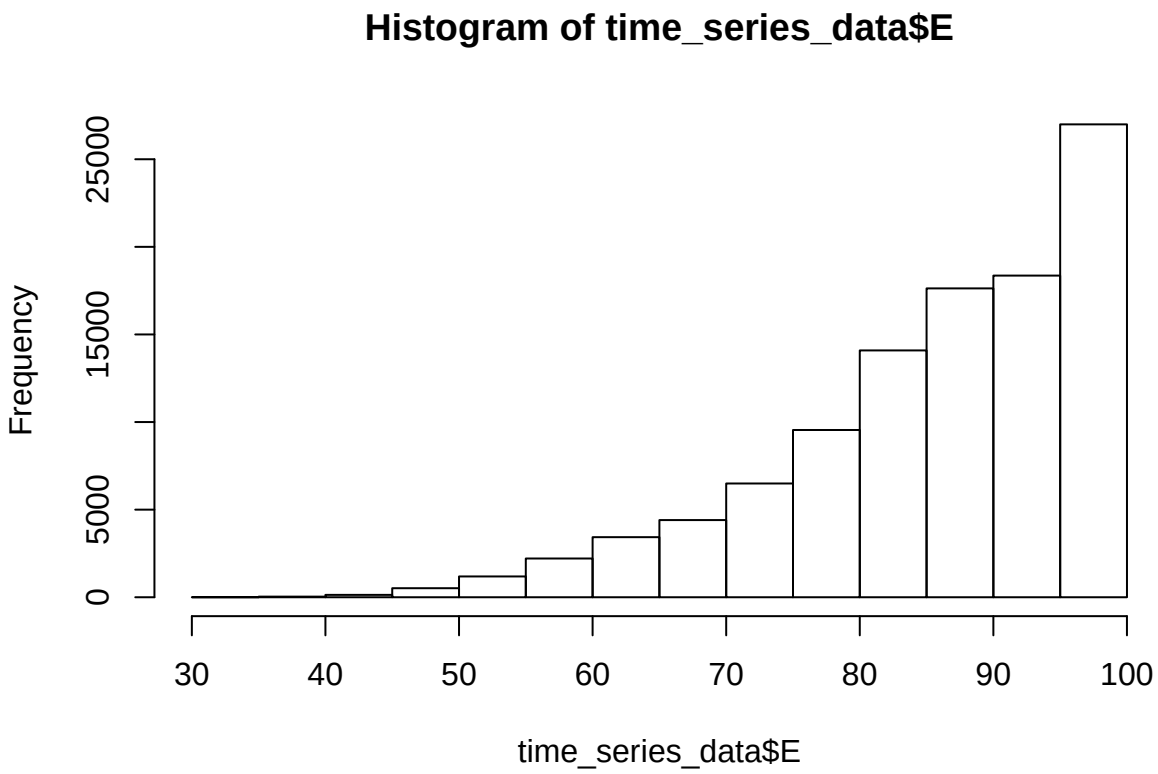
Lets randomly subsample this to see if we can see any better trends. (the number to subsample by could be better chosen)

```
time_series_data_subsample <- time_series_data[sample(nrow(time_series_data), 1000),
    ]

ggplot(data = time_series_data_subsample, aes(x = E, y = id)) + geom_line(linetype = "dashed") +
    geom_point()
```



Ok, lets try a histogram of this.

```
# Extract histogram information
hist_vec <- hist(time_series_data$E)
```




```
# Store histogram counts in frequency
frequency <- hist_vec$counts

frequency
```

```
 [1]     3    35   135   516  1188  2212  3430  4405  6493  9547 14088 17629
[13] 18359 26992
```

SUMMARY colA

My hyposthesis for ColA is that is it temperature data over a year, based on the shape of the graph (although this graph is horrible - changing the defaults would be better) and based on the values. Min -7, mean of 9. The shape of the graph suggests to me the change in temp over the year. Given that there are 3.154e+7 seconds in a year and 105032 obseravtions.

print(3.154e+7 / 105032) = 300

300/60 = 5.

Therefore there is an observation every 5 mins.

QUESTION TO THE REASEARCH TEAM/ problems: These temperatures (if it is temp data) seem a little high for Antartica. Is the equipment callibrated correctly? I know there are serious warming problems, how does this compare to last year?

SUMMARY colB

This is a poisson distribution. These are discreet counts of observations per time point, where the observation is less likely over time. ColA and ColB have a 0.11 corrolation, thus they are not corrolated.

problem: Looking at the plot there is a periodical increase then sharpe decrease. The pattern looks too regular to be biological data? Systematic error?

SUMMARY colC

```
summary(time_series_data)
```

```
       A                B                 C                 D          
 Min.   :-7.600   Min.   : 0.0000   Min.   :   0.00   Min.   : 963.8  
 1st Qu.: 5.300   1st Qu.: 0.0000   1st Qu.:   0.00   1st Qu.: 997.4  
 Median : 9.200   Median : 0.0000   Median :   5.00   Median :1007.4  
 Mean   : 9.115   Mean   : 0.8566   Mean   :  80.18   Mean   :1006.1  
 3rd Qu.:13.200   3rd Qu.: 1.0000   3rd Qu.:  80.00   3rd Qu.:1016.0  
 Max.   :27.300   Max.   :13.0000   Max.   :1098.00   Max.   :1036.4  
       E                id        
 Min.   : 34.00   Min.   :     1  
 1st Qu.: 80.00   1st Qu.: 26259  
 Median : 89.00   Median : 52517  
 Mean   : 86.15   Mean   : 52517  
 3rd Qu.: 96.00   3rd Qu.: 78774  
 Max.   :100.00   Max.   :105032  
```

ColA and C have a corroltation of 0.46. This is somewhat corrolated, although not statistically significant.

QUESTION TO THE REASEARCH TEAM/ Problems: I am concerned that an element from ColD has ended up in ColC (max: 1098) . Looking at the medians of both cols and the max in ColC. Although, there are a few other high enteries in ColC which may be out of place in ColD . . . This needs to be clarified.

SUMMARY colD

There is a slight negative screw to the data with several very low values. However, there does not seem to be a trend over time and looking at the subsampled plot - there is no trend over time.

Although if you look at the non subsampled plot you can see moving avaergaes over shorter time periods.

SUMMARY colE

The max value of colE is 100. With the highest number of observations in the historgam bin occuring at the top. I guess this is a percentage based count.