HW3_sliang

Liang Shi

9/26/2020

Problem 3

I'm familiar with almost of these rules. What impress me is the rule "Don't use attach()". So I make some research and find out if we have several different datasets, the cold will become messy.

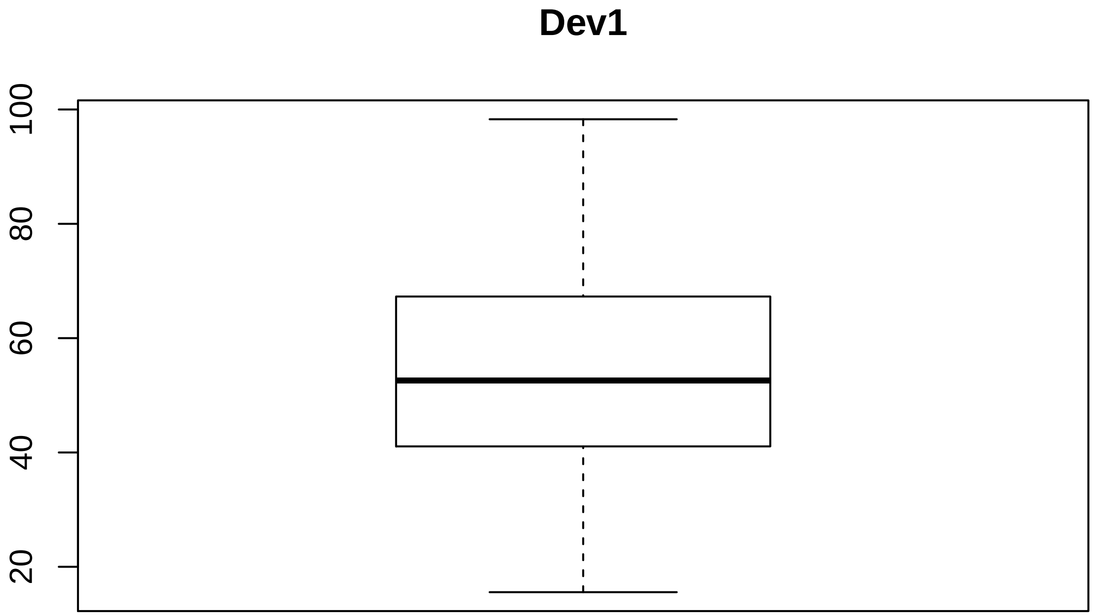
Problem 5

```
df <- readRDS("HW3_data.rds")

get_result <- function(df){
  mean_1 <- mean(df$dev1)
  mean_2 <- mean(df$dev2)
  sd_1 <- sd(df$dev1)
  sd_2 <- sd(df$dev2)
  corr_12 <- cor(df$dev1, df$dev2)
  return (c(mean_1, mean_2, sd_1, sd_2, corr_12))
}

df_organized <- data.frame(Observer = NULL, Mean_1 = NULL, Mean_2 = NULL, SD_1 = NULL, SD_2 = NULL, Cor

for (i in 1:13){
  result <- get_result(df[df$Observer == i, 2:3])
  df_cur_observer = data.frame(Observer = i, Mean_1 = result[1], Mean_2 = result[2], SD_1 = result[3], 
  df_organized <- rbind(df_organized, df_cur_observer)
}
kable(df_organized)
```

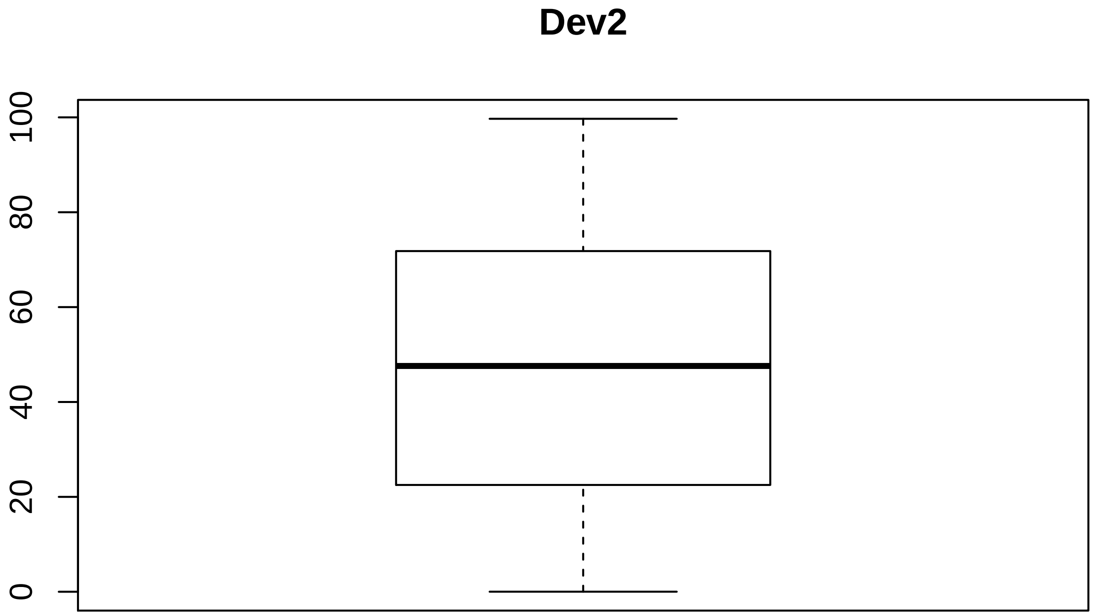
Observer	Mean_1	Mean_2	SD_1	SD_2	Corr_12
1	54.26610	47.83472	16.76983	26.93974	-0.0641284
2	54.26873	47.83082	16.76924	26.93573	-0.0685864
3	54.26732	47.83772	16.76001	26.93004	-0.0683434
4	54.26327	47.83225	16.76514	26.93540	-0.0644719
5	54.26030	47.83983	16.76774	26.93019	-0.0603414
6	54.26144	47.83025	16.76590	26.93988	-0.0617148
7	54.26881	47.83545	16.76670	26.94000	-0.0685042
8	54.26785	47.83590	16.76676	26.93610	-0.0689797
9	54.26588	47.83150	16.76885	26.93861	-0.0686092
10	54.26734	47.83955	16.76896	26.93027	-0.0629611
11	54.26993	47.83699	16.76996	26.93768	-0.0694456
12	54.26692	47.83160	16.77000	26.93790	-0.0665752

Observer	Mean_1	Mean_2	SD_1	SD_2	Corr_12
13	54.26015	47.83972	16.76996	26.93000	-0.0655833

```r
boxplot(df$dev1, main = "Dev1")
```

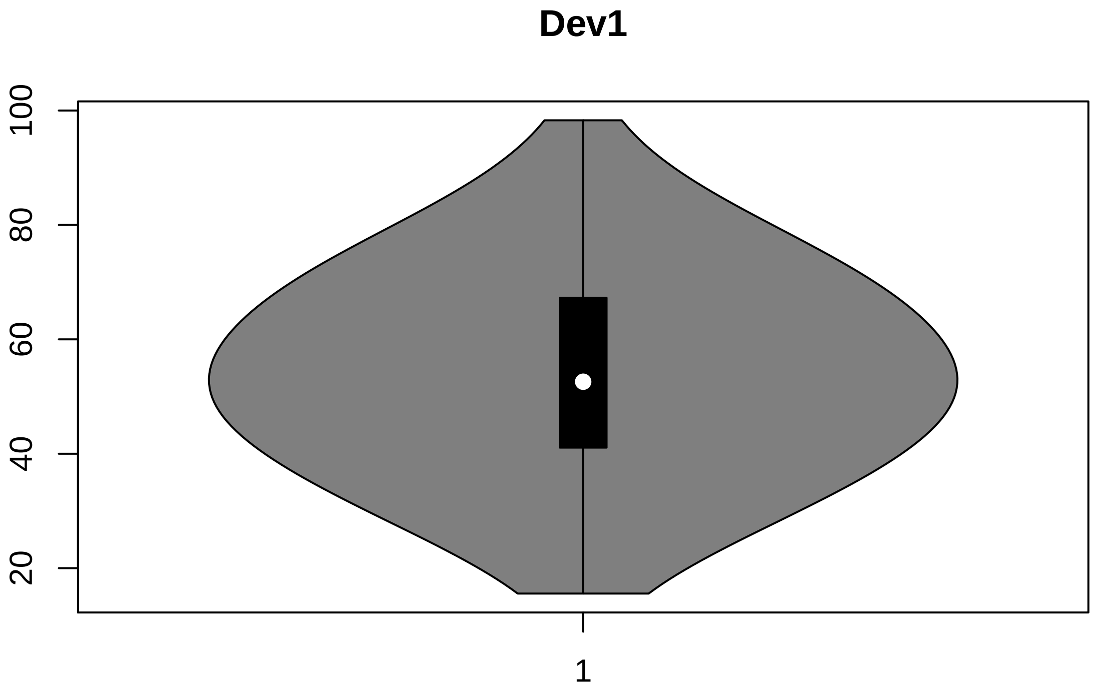


```r
boxplot(df$dev2, main = "Dev2")
```

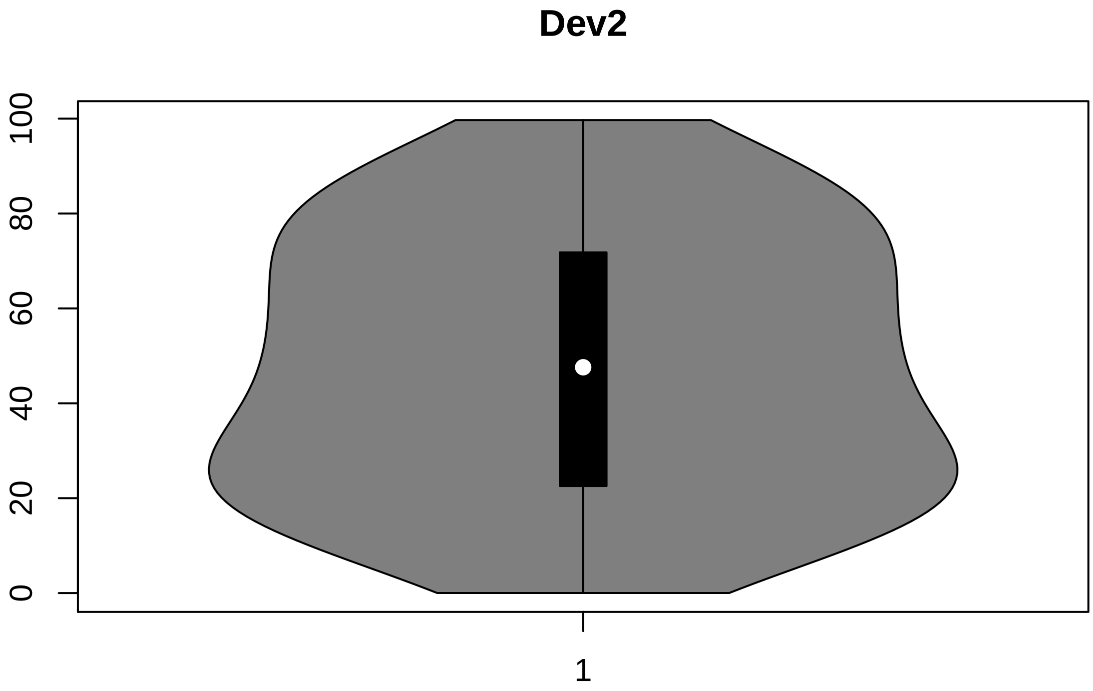


These two distribution are almost symmertic, and no outliers here.

```r
vioplot(df$dev1, main = "Dev1")
```

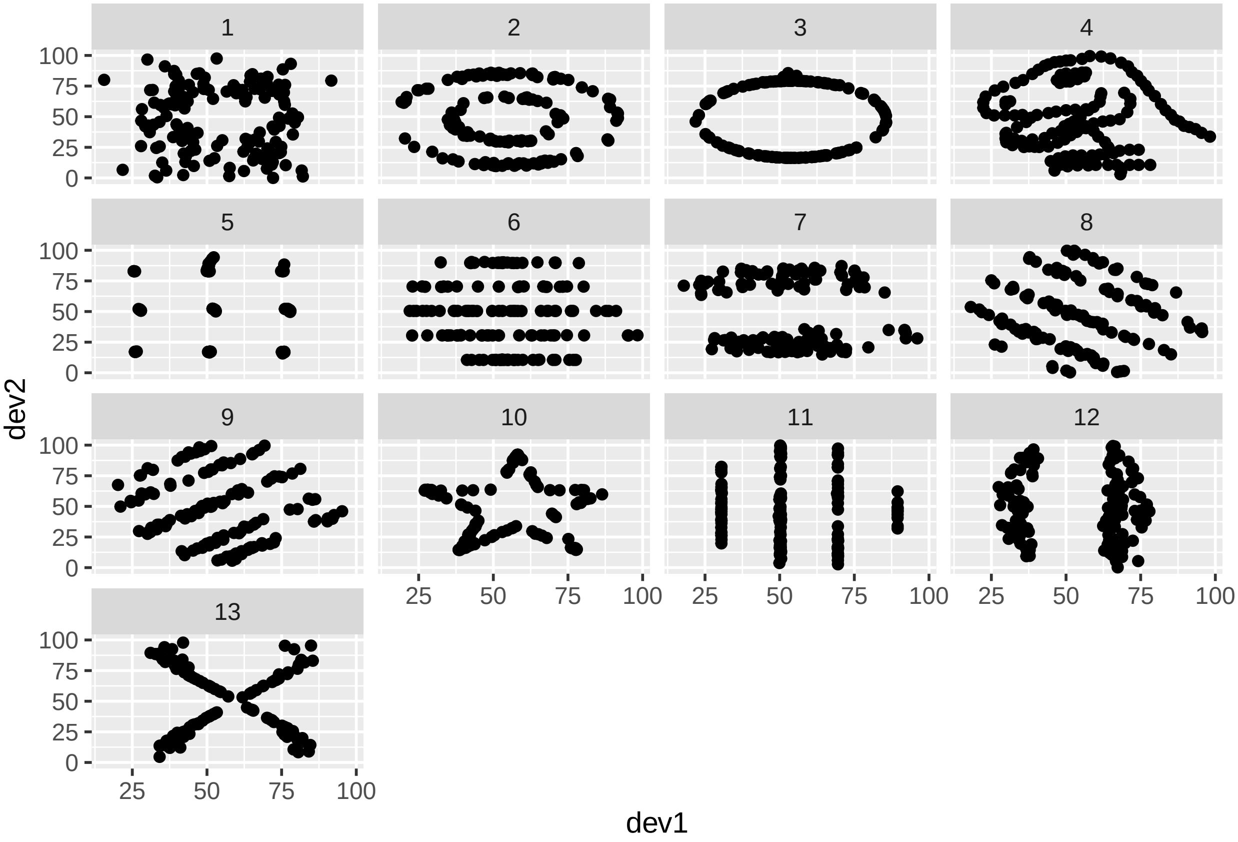


```r
vioplot(df$dev2, main = "Dev2")
```



From the vioplot, we know that in dev1, the mode is near the median, but in dev2, the mode is a little skew from the median. And we can also see that the standard deviation of dev2 is larger than of dev1.

```
ggplot(df, aes(x=dev1,y=dev2)) + geom_point() + facet_wrap(Observer~.)
```



I see many funny plots. From the scatter plot, we know the correlation between dev1 and dev2 is very weak, which agree with our data summary. I think plot the data is a very important way before analysis, because statistics only may misleading our approach.

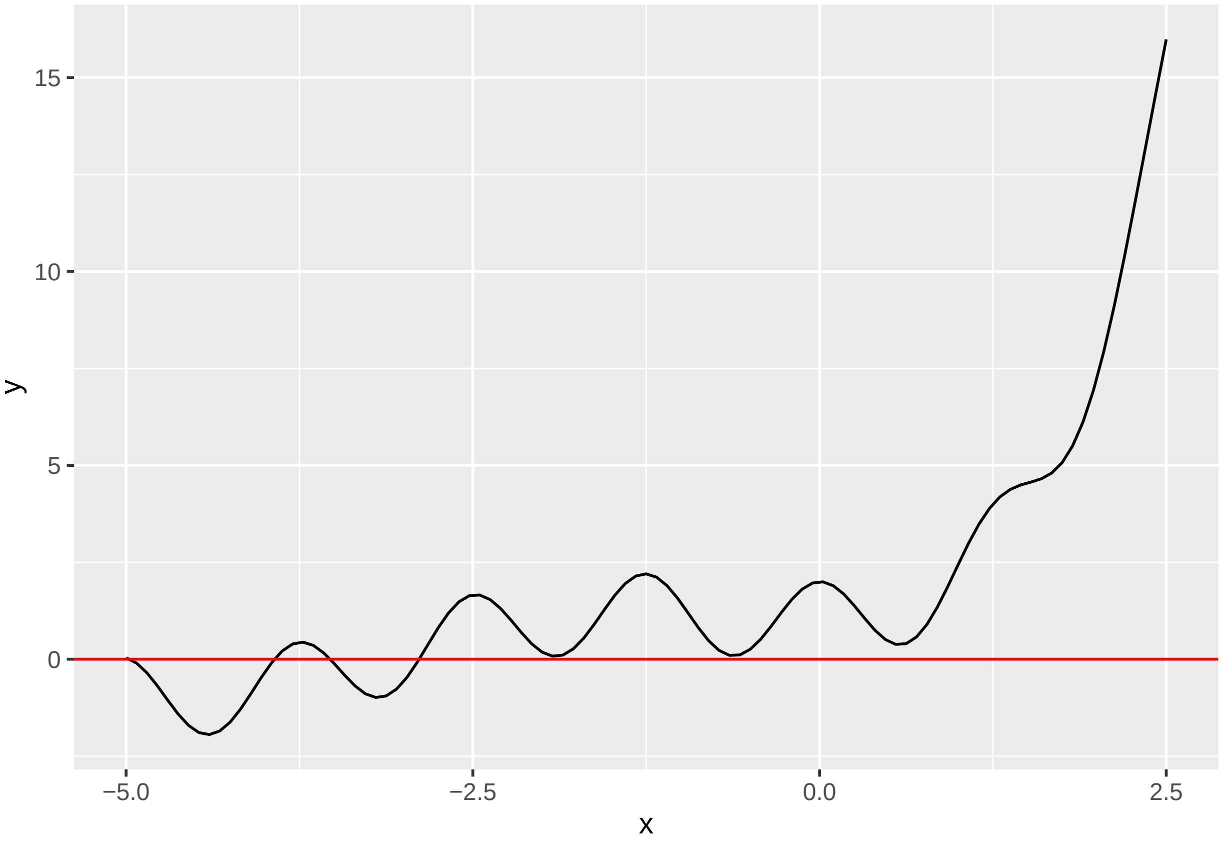
Problem 6

```
get_integral <- function(x){
  y <- exp(- (x^2 / 2))
  return(y)
}

x_range <- seq(0, 1, by = 1e-6)
sum(get_integral(x_range) * 1e-6)
```

```
## [1] 0.8556252
```

Problem 7

```
fx <- function(x){
  y <- 3^x - sin(x) + cos(5*x)
}

fdx <- function(x){
  y <- 3^x*log(3) - cos(x) - 5*sin(5*x)
}

ggplot(data = data.frame(x = 0,y = 0), mapping = aes(x = x)) +
  stat_function(fun = fx) +
  xlim(-5, 2.5) +
  geom_abline(intercept = 0, slope = 0, colour = "red")
```



```
x <- -2.5
while (abs(fx(x)-0) > 1e-6) {
  x <- x - fx(x)/fdx(x)
}
x
```

```
## [1] -3.930114
```

Problem 8

```
start <- Sys.time()
X <- cbind(rep(1,100),rep.int(1:10,time=10))
beta <- c(4,5)
y <- X%*%beta + rnorm(100)
I <- diag(100)
NJ_1 <- rep(1/100, 100)
y_A <- vector()
for (i in 1:100){
  a <- I[,i] - NJ_1
  num <- t(y)%*%a
  y_A[i] <- num
}
SST = t(y_A)%*%y
print(SST)
```

```
##           [,1]
## [1,] 20737.44
```

```
end <- Sys.time()
print(end - start)
```

```
## Time difference of 0.06573987 secs
```