HW_1 Grigoreva Elizaveta

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(ggpubr)
```

```
## Loading required package: magrittr
```

#1. Measures of center # 1.0 create own sample or use given vector and write mode, median, mean functions/one-liners

```
x <- c(175, 176, 180, 165, 167, 172, 175, 146, 158, 178)
#Get mode
mymode <- function(x){
  un <- unique(x)
  r <- tabulate(match(x, un))
  return(un[r == max(r)])
}
#Median
mymedian <- function(x) {
  n <- length(x)
  s <- sort(x)
  ifelse(n%%2==1,s[(n+1)/2],mean(s[n/2+0:1]))
}
#Mean
mymean <- sum(x)/length(x)
```

1.1 calculate mode, median and mean for the sample.

#Compare results for own and built-ins for median and mean

```
mymode(x)
```

```
## [1] 175
```

```
mymedian(x)
```

```
## [1] 173.5
```

```r
median(x)
```

```
## [1] 173.5
```

```r
mymean
```

```
## [1] 169.2
```

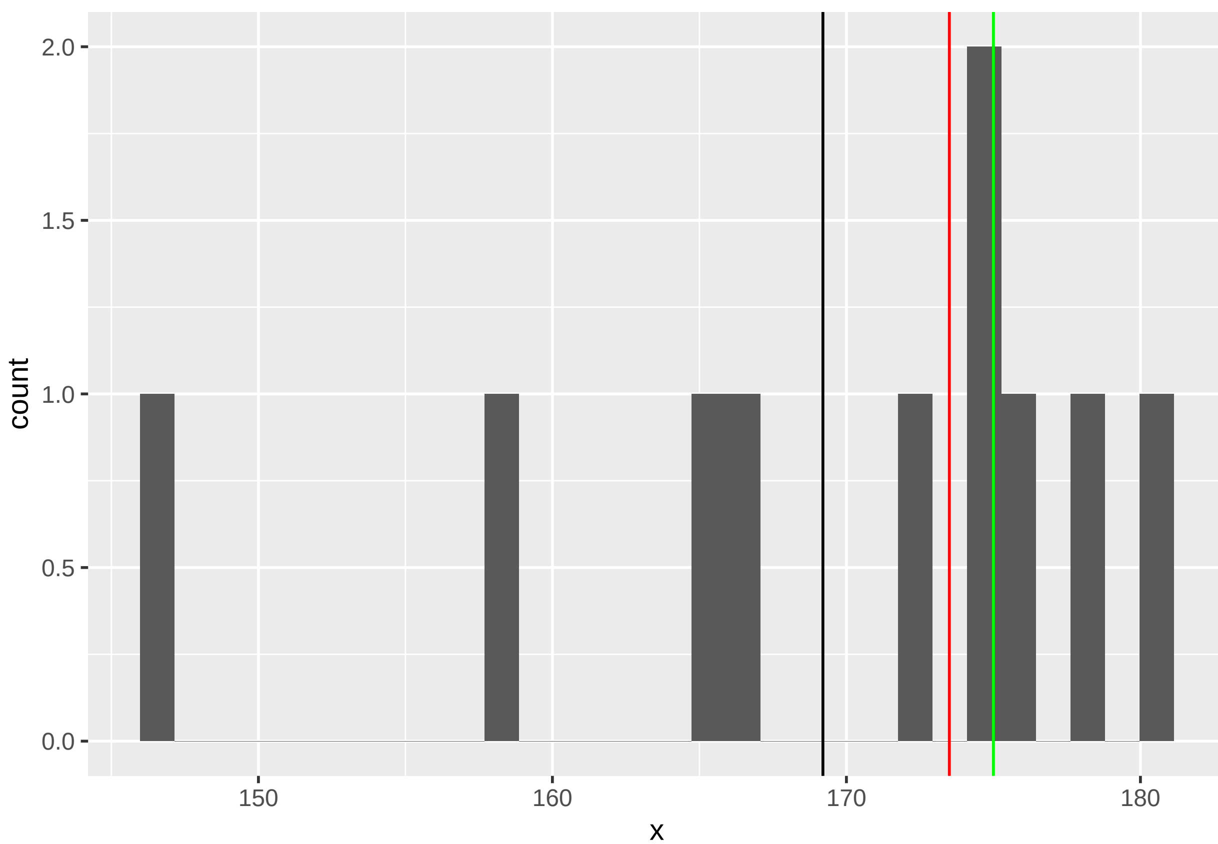
```r
mean(x)
```

```
## [1] 169.2
```

1.2 visualize histogram with 3 vertical lines for measures of center

```r
ggplot(as.data.frame(x), aes(x = x)) +
  geom_histogram() +
  geom_vline(xintercept = mymean, color = 'black') +
  geom_vline(xintercept = mymedian(x), color = 'red') +
  geom_vline(xintercept = mymode(x), color = 'green')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



1.3 spoil your sample with the outlier - repeat steps 1.1 and 1.2

```r
x[length(x) + 1] <- 15
## 1.1. repeat
mymode(x)
```

```
## [1] 175
```

```r
mymedian(x)
```

```
## [1] 172
```

```r
median(x)
```

```
## [1] 172
```

```r
mymean2 <- sum(x)/length(x)
mymean2
```

```
## [1] 155.1818
```

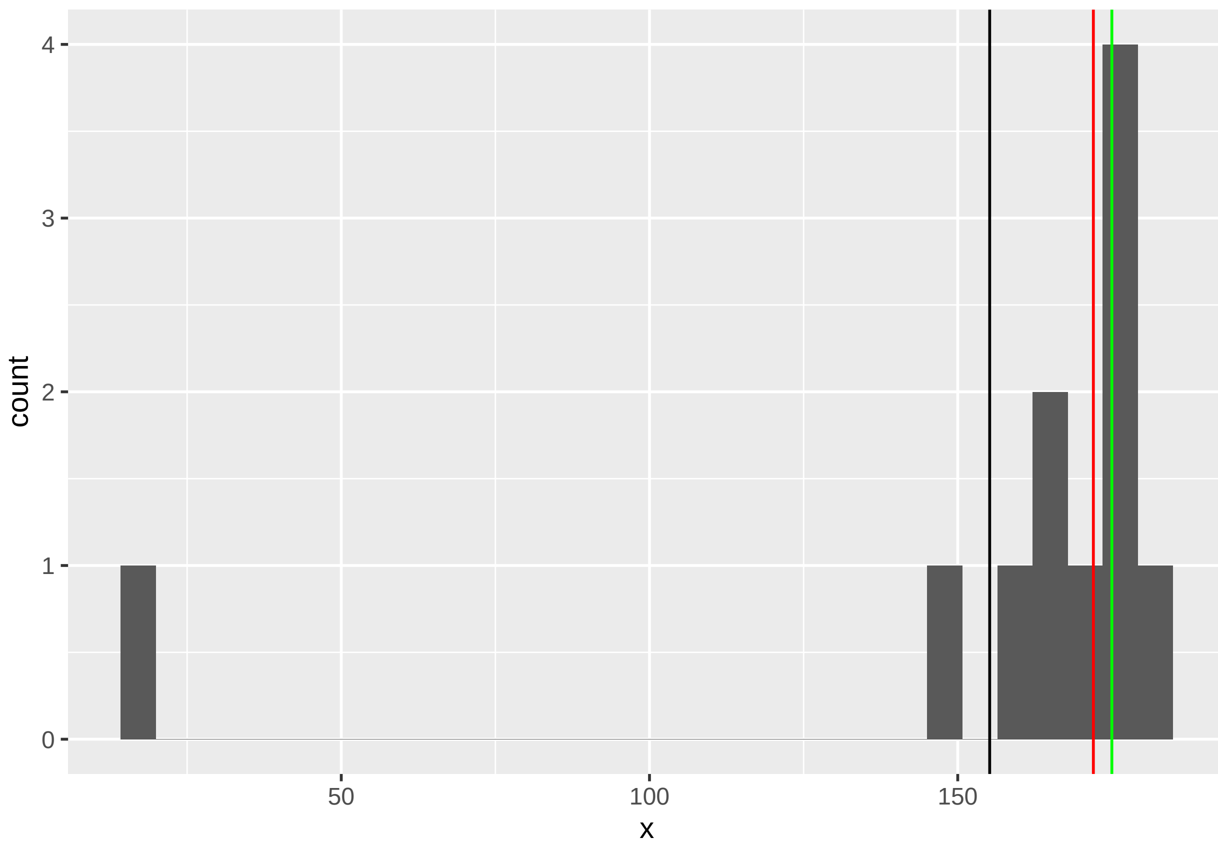
```r
mean(x)
```

```
## [1] 155.1818
```

Vizualize it

```r
ggplot(as.data.frame(x), aes(x = x)) +
  geom_histogram() +
  geom_vline(xintercept = mymean2, color = 'black') +
  geom_vline(xintercept = mymedian(x), color = 'red') +
  geom_vline(xintercept = mymode(x), color = 'green')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



#Measures of spread # 2.0 write the functions/one-liners for variance and sd, calculate result, compare with the built-ins

```r
x <- c(175, 176, 182, 165, 167, 172, 175, 196, 158, 172)
var_one_line <- mean((x-mean(x))^2)
sd_one_line <- sqrt(sum((x-mean(x))^2/(length(x)-1)))
var(x)
```

```
## [1] 105.2889
```

```r
var_one_line
```

```
## [1] 94.76
```

```r
sd(x)
```

```
## [1] 10.26104
```

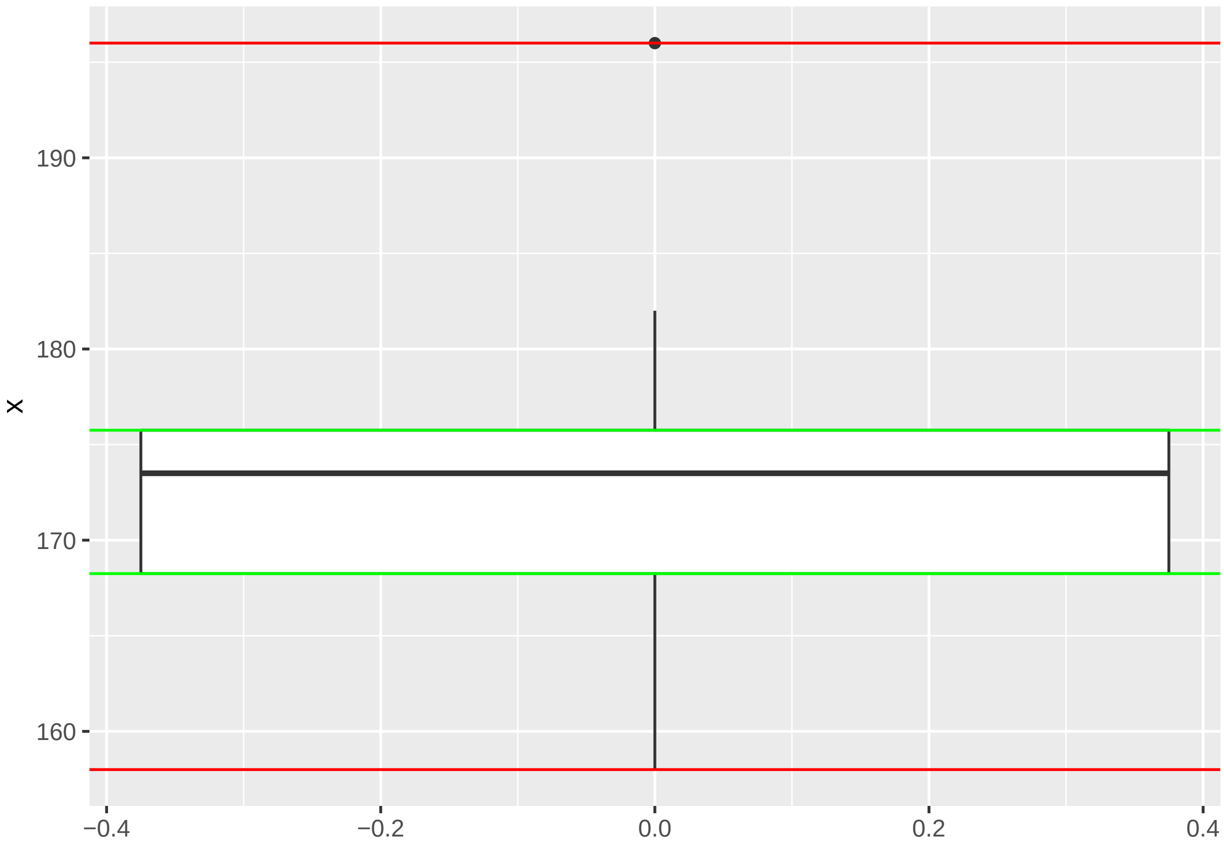
```r
sd_one_line
```

```
## [1] 10.26104
```

2.1 visualize with the box plot.

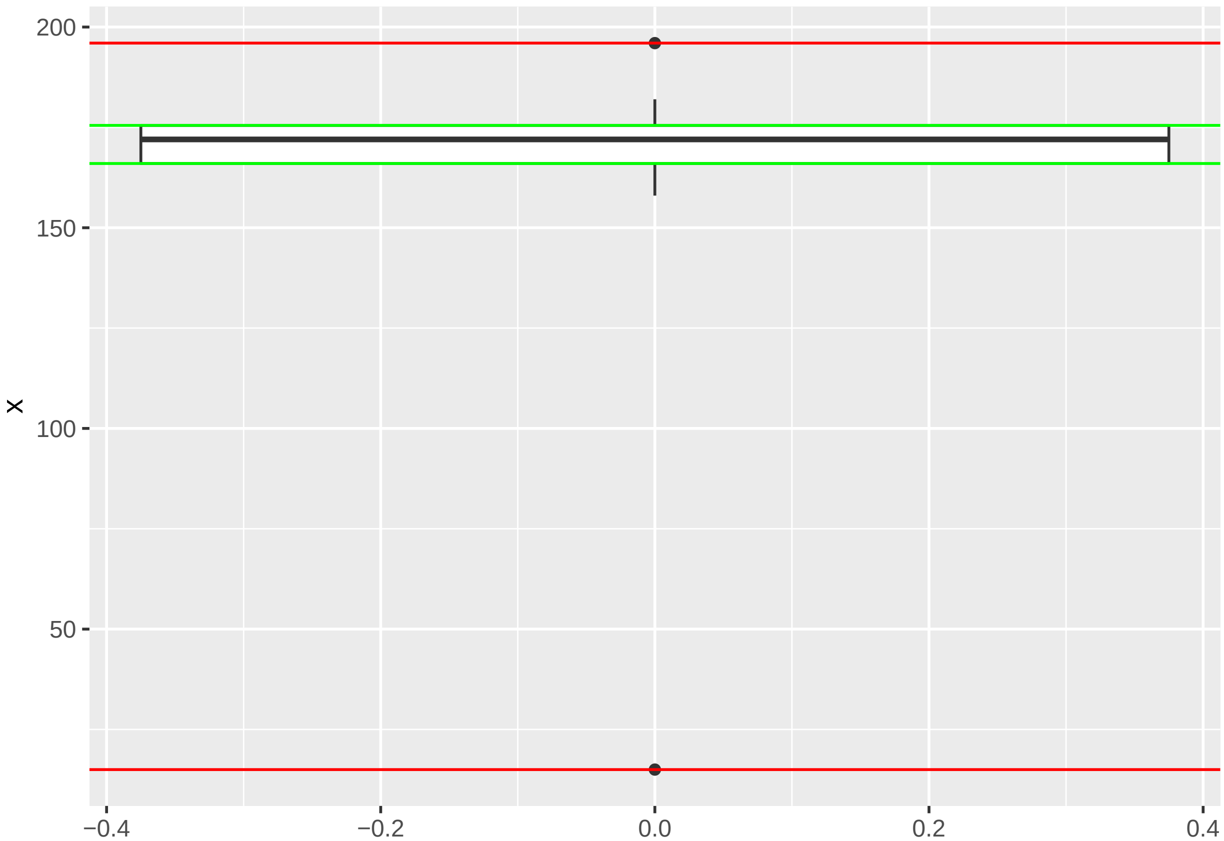
Add horizontal lines for range, IQR, 1-sd borders (use built-ins)

```r
ggplot(as.data.frame(x), aes(y = x)) +
  geom_boxplot() +
  geom_hline(yintercept = min(x), color = 'red') +
  geom_hline(yintercept = max(x), color = 'red') +
  geom_hline(yintercept = quantile(x, 3/4), color = 'green') +
  geom_hline(yintercept = quantile(x, 1/4), color = 'green')
```



2.2 spoil your sample with the outlier, repeat step 2.1 and back vector

```
x[length(x) + 1] <- 15
ggplot(as.data.frame(x), aes(y = x)) +
  geom_boxplot() +
  geom_hline(yintercept = min(x), color = 'red') +
  geom_hline(yintercept = max(x), color = 'red') +
  geom_hline(yintercept = quantile(x, 3/4), color = 'green') +
  geom_hline(yintercept = quantile(x, 1/4), color = 'green')
```




```
x <- x[1:length(x) - 1]
```

3. Properties

3.0 check the properties for mean and sd for your sample

```
x <- c(175, 176, 182, 165, 167, 172, 175,196, 158, 172)
var(x - 100)
```

```
## [1] 105.2889
```

```
var(x)
```

```
## [1] 105.2889
```

```
var(x / 100)
```

```
## [1] 0.01052889
```

```
var(x) / 10000
```

```
## [1] 0.01052889
```

```
sd(x / 100)
```

```
## [1] 0.1026104
```

```
sd(x) / 100
```

```
## [1] 0.1026104
```

```r
abs(sum(x) -mean(x) -0 ) < 0.000000001
```

```
## [1] FALSE
```

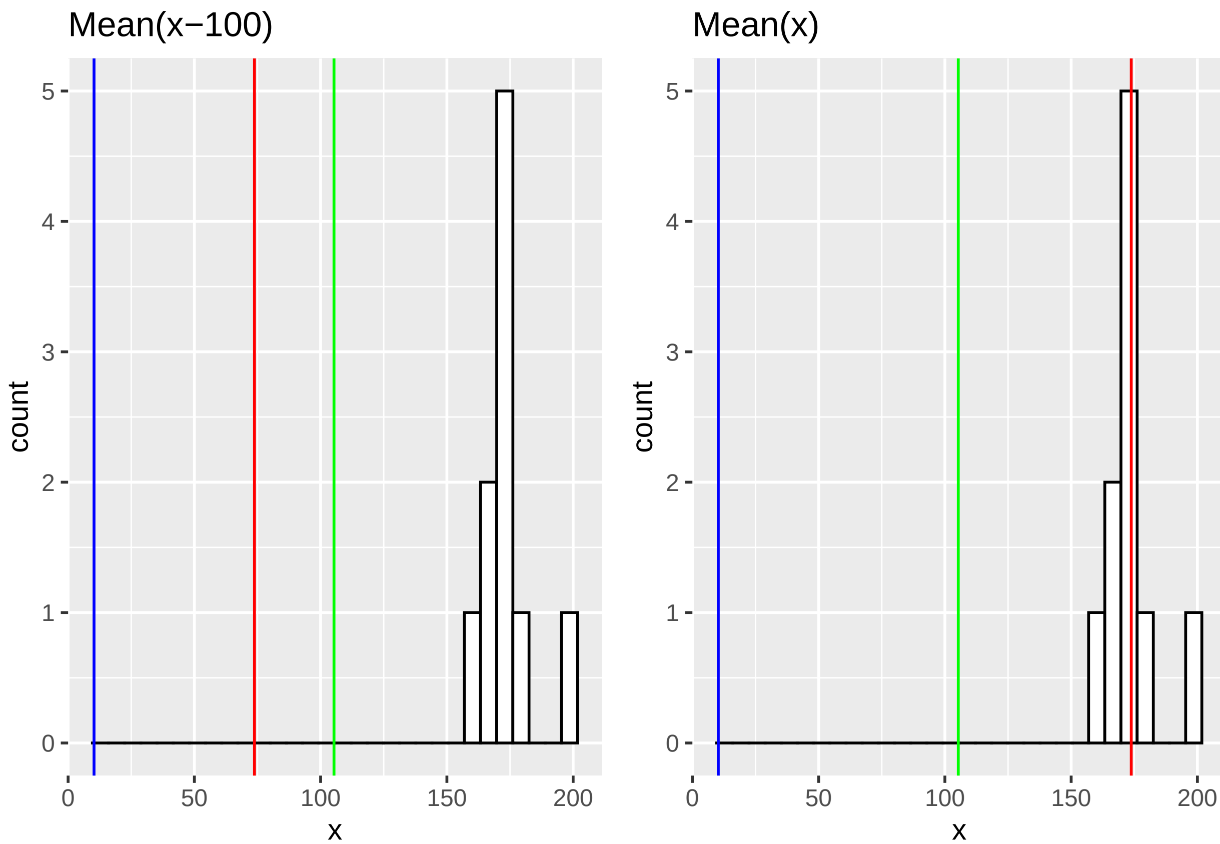
3.1 visualize result tabularly and graphically (maybe with facetting free scales?)

```r
#Vizualize tabulary
properties_table <- matrix(c(mean(x), mean(x - 100), mean(x/ 100),
                       var(x), var(x - 100),var(x / 100),
                       sd(x), sd(x - 100), sd(x / 100)), ncol = 3, byrow = TRUE)
colnames(properties_table ) <- c("x","x-100","x/100")
rownames(properties_table ) <- c("mean","var","sd")
as.table(properties_table)
```

```
##                 x        x-100        x/100
## mean 173.80000000  73.80000000   1.73800000
## var  105.28888889 105.28888889   0.01052889
## sd    10.26103742  10.26103742   0.10261037
```

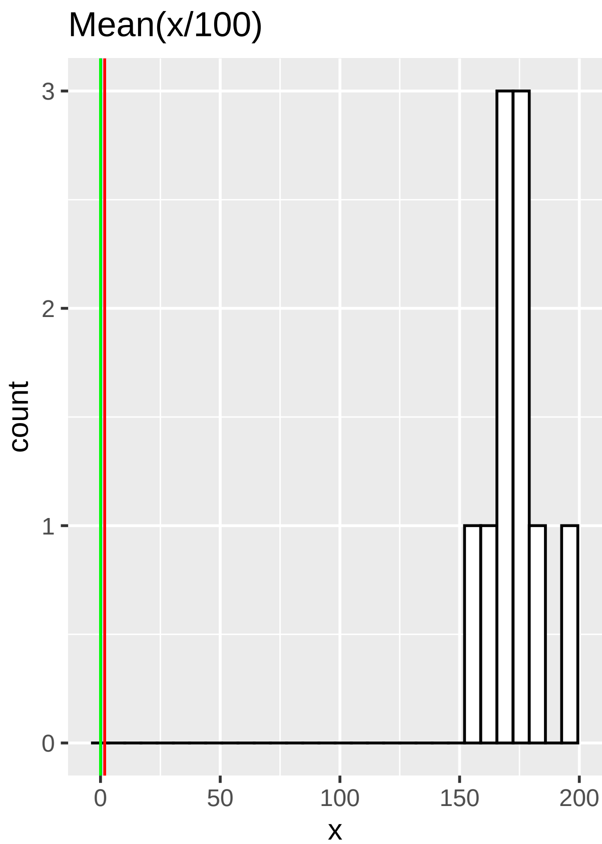
```r
#Vizualize for mean
a <- ggplot() +
  aes(x) +
  geom_histogram(colour="black", fill="white") +
  geom_vline(xintercept=mean(x-100), color="red") +
  geom_vline(xintercept=sd(x-100), color="blue") +
  geom_vline(xintercept=var(x-100), color="green") +
  ggtitle(label = 'Mean(x-100)')

b <-  ggplot() +
  aes(x) +
  geom_histogram(colour="black", fill="white") +
  geom_vline(xintercept=mean(x), color="red") +
  geom_vline(xintercept=sd(x), color="blue") +
  geom_vline(xintercept=var(x), color="green") +
  ggtitle(label = 'Mean(x)')

ggarrange(a, b,  ncol = 2, nrow = 1)
```

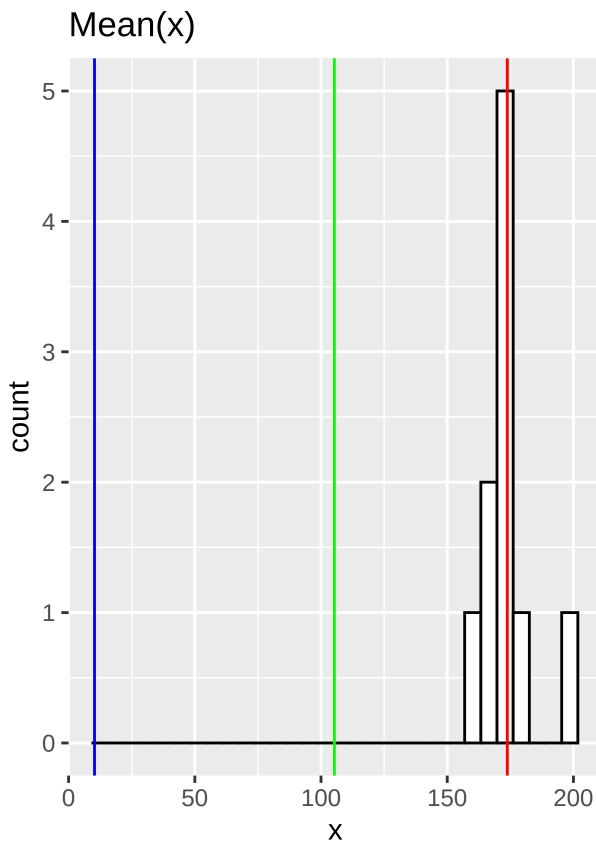
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
c <- ggplot() +
  aes(x) +
  geom_histogram(colour="black", fill="white") +
  geom_vline(xintercept=mean(x/100), color="red") +
  geom_vline(xintercept=sd(x/100), color="blue") +
  geom_vline(xintercept=var(x/100), color="green") +
  ggtitle(label = 'Mean(x/100)')

d <-  ggplot() +
  aes(x) +
  geom_histogram(colour="black", fill="white") +
  geom_vline(xintercept=mean(x), color="red") +
  geom_vline(xintercept=sd(x), color="blue") +
  geom_vline(xintercept=var(x), color="green") +
  ggtitle(label = 'Mean(x)')

ggarrange(c, d,  ncol = 2, nrow = 1)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```





4 Normal Distribution # 4.0 for the population N(175, 10) find the probability to be: # less than 156cm, # more than 198, # between 168 and 172 cm

```
pnorm(156,175,10, lower.tail = TRUE)
```

```
## [1] 0.02871656
```

```
pnorm(198,175,10, lower.tail = FALSE)
```

```
## [1] 0.01072411
```

```
pnorm(168,175,10, lower.tail = FALSE)-pnorm(172, mean = 175, sd = 10, lower.tail = TRUE)
```

```
## [1] 0.3759478
```

Standard normal distribution

4.1 check the properties of 1-2-3-sd's for standard normal distribution using pnorm()

```
pnorm(1) - pnorm(-1) # 68,2%
```

```
## [1] 0.6826895
```

```
pnorm(2) - pnorm(-2) # 95,4%
```

```
## [1] 0.9544997
```

```
pnorm(3) - pnorm(-3)# 99.7%
```

```
## [1] 0.9973002
```

4.3 standardize, find the same

```r
set.seed(42)
x <- rnorm(1000,175,10)
mean(x)
```

```
## [1] 174.7418
```

```r
sd(x)
```

```
## [1] 10.02521
```

```r
x1 <- (x-mean(x))/sd(x)
mean(x1)
```

```
## [1] -2.744457e-16
```

```r
sd(x1)
```

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

```r
x <- rnorm(1000, mean = 0, sd = 1)
mean(x)
```

```
## [1] -0.005317994
```

```r
sd(x)
```

```
## [1] 0.986061
```

5. Central Limit Theorem

5.0 Generate large population (n ~ 100 000 - 1 000 000) distributed as N(0, 1)

Sample from population k observations for 30 times - you will have set of 30 samples.

For each sample calculate mean. For the set calculate means of means, sd of means, SE.

Create table with k, mean of means, sd of means, SE.

Visualize distribution of means with histogram and lines for mean of means and SE.

5.1 k = 10

5.2 k = 50

5.3 k = 100

5.4 k = 500

5.5 Compare results

```
set.seed(42)
x <- rnorm(1000000, mean = 0, sd = 1)
#10,50,100,500
k_10 <- replicate(30, sample(x, 10))
k_50 <- replicate(30,sample(x,50))
k_100 <- replicate(30,sample(x,100))
k_500 <- replicate(30,sample(x,500))
#mean
means <- function(k){
  m <- c()
  for (i in 1:ncol(k)) {
    m[i] <- mean(k[,i])
  }
  return(m)
}
SE <- function(k){
  return(sd(k)/sqrt(length(k)))
}
#10
means_of_mean_K_10 <- means(k_10)
mean(means_of_mean_K_10)
```

```
## [1] -0.02034378
```

```r
sd(means_of_mean_K_10)
```

```
## [1] 0.3392652
```

```r
sd(means_of_mean_K_10)
```

```
## [1] 0.3392652
```

```r
SE(means_of_mean_K_10)
```

```
## [1] 0.06194107
```

```r
#50
means_of_mean_K_50 <- means(k_50)
mean(means_of_mean_K_50)
```

```
## [1] 0.00483748
```

```r
sd(means_of_mean_K_50)
```

```
## [1] 0.1299833
```

```r
 SE(means_of_mean_K_50)
```

```
## [1] 0.02373159
```

```r
#100
means_of_mean_K_100 <- means(k_100)
mean(means_of_mean_K_100)
```

```
## [1] 0.01439165
```

```r
sd(means_of_mean_K_100)
```

```
## [1] 0.1093768
```

```r
 SE(means_of_mean_K_100)
```

```
## [1] 0.01996938
```

```r
#500
means_of_mean_K_500 <- means(k_500)
mean(means_of_mean_K_500)
```

```
## [1] 0.003615403
```

```r
sd(means_of_mean_K_500)
```

```
## [1] 0.03664876
```

```r
SE(means_of_mean_K_500)
```

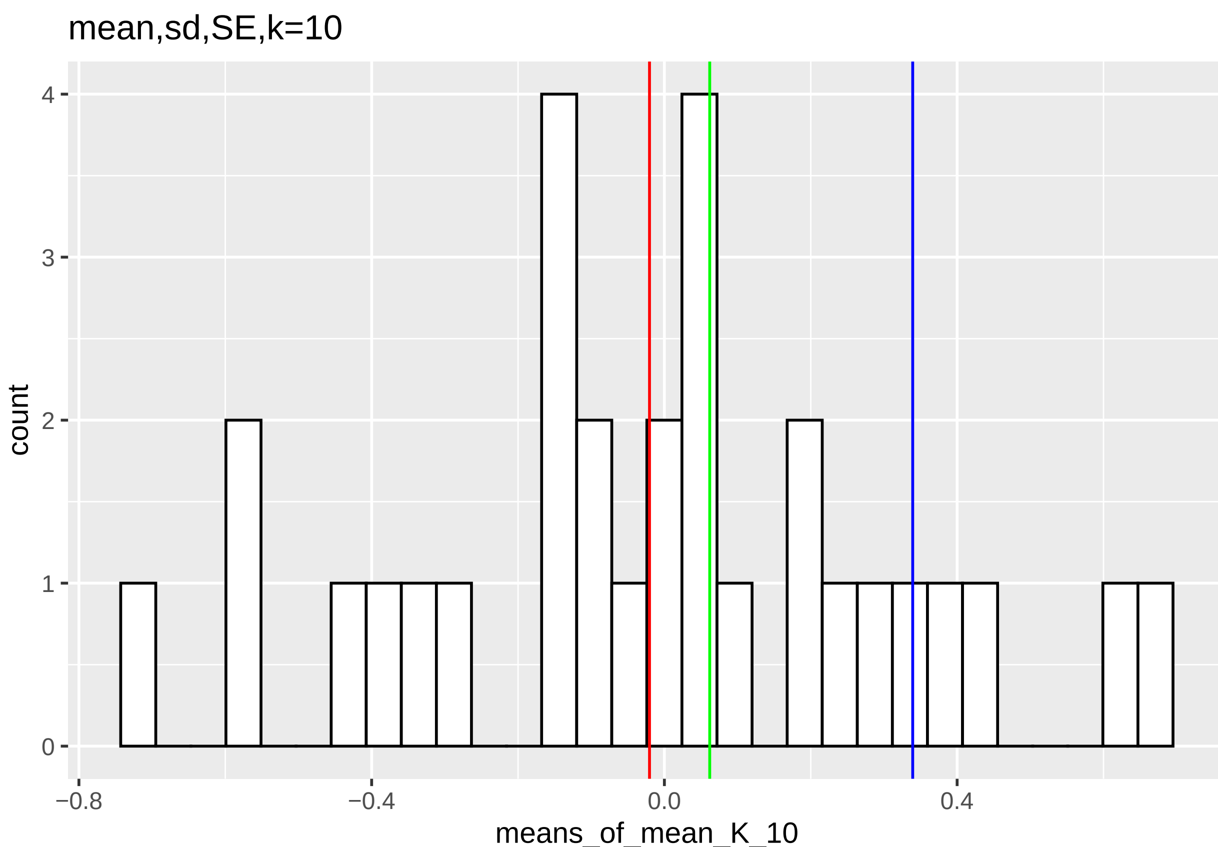
```
## [1] 0.006691118
```

```r
#Create table
table_samples <- matrix(c(mean(means_of_mean_K_10),sd(means_of_mean_K_10),SE(means_of_mean_K_10),
                    mean(means_of_mean_K_50), sd(means_of_mean_K_50), SE(means_of_mean_K_50),
                    mean(means_of_mean_K_100), sd(means_of_mean_K_100), SE(means_of_mean_K_100),
                    mean(means_of_mean_K_500), sd(means_of_mean_K_500), SE(means_of_mean_K_500)),ncol=3)
colnames(table_samples) <- c("mean","sd","SE")
rownames(table_samples) <- c("10","50","100", "500")
table_samples <- as.table(table_samples)
table_samples
```

```
##              mean          sd          SE
## 10  -0.020343779  0.129983255  0.019969384
## 50   0.339265202  0.023731587  0.003615403
## 100  0.061941068  0.014391655  0.036648761
## 500  0.004837480  0.109376823  0.006691118
```

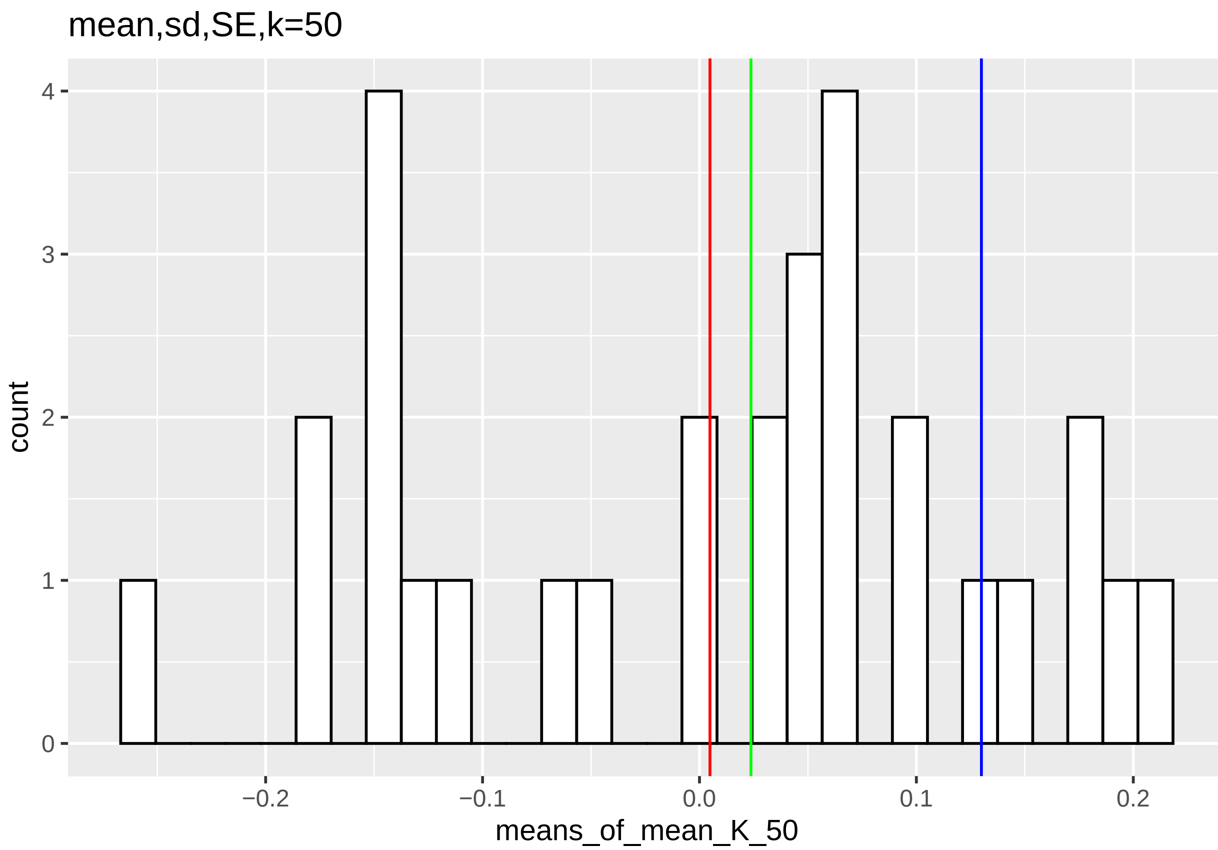
```r
#Vizualizing distribution
q <- ggplot() +
aes(means_of_mean_K_10) +
  geom_histogram(colour="black", fill="white") +
  geom_vline(xintercept=mean(means_of_mean_K_10), color="red") +
  geom_vline(xintercept=sd(means_of_mean_K_10),color="blue")+
  geom_vline(xintercept=SE(means_of_mean_K_10), color='green')+
  ggtitle(label = 'mean,sd,SE,k=10')
q
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



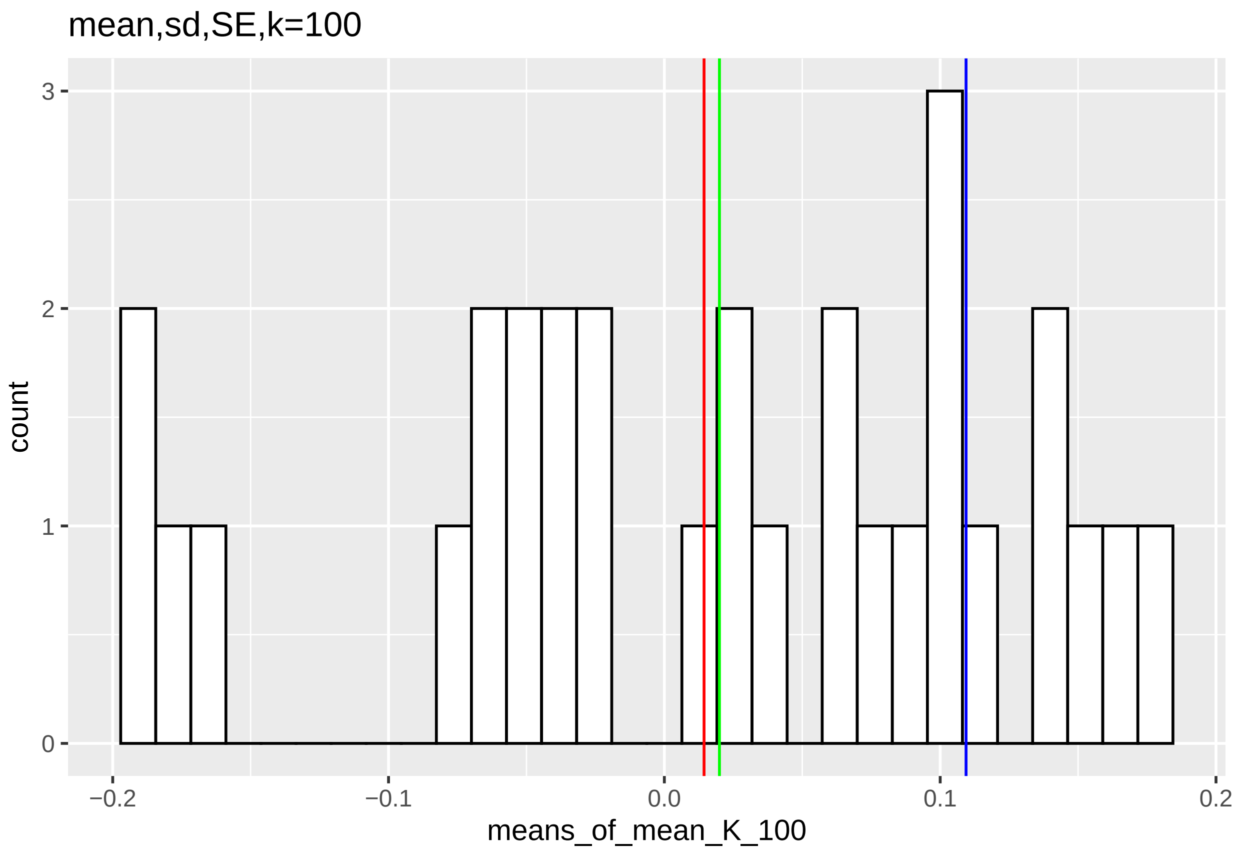

```r
z <- ggplot() +
aes(means_of_mean_K_50) +
  geom_histogram(colour="black", fill="white") +
  geom_vline(xintercept=mean(means_of_mean_K_50), color="red") +
  geom_vline(xintercept=sd(means_of_mean_K_50),color="blue")+
  geom_vline(xintercept=SE(means_of_mean_K_50), color='green')+
  ggtitle(label = 'mean,sd,SE,k=50')


z
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



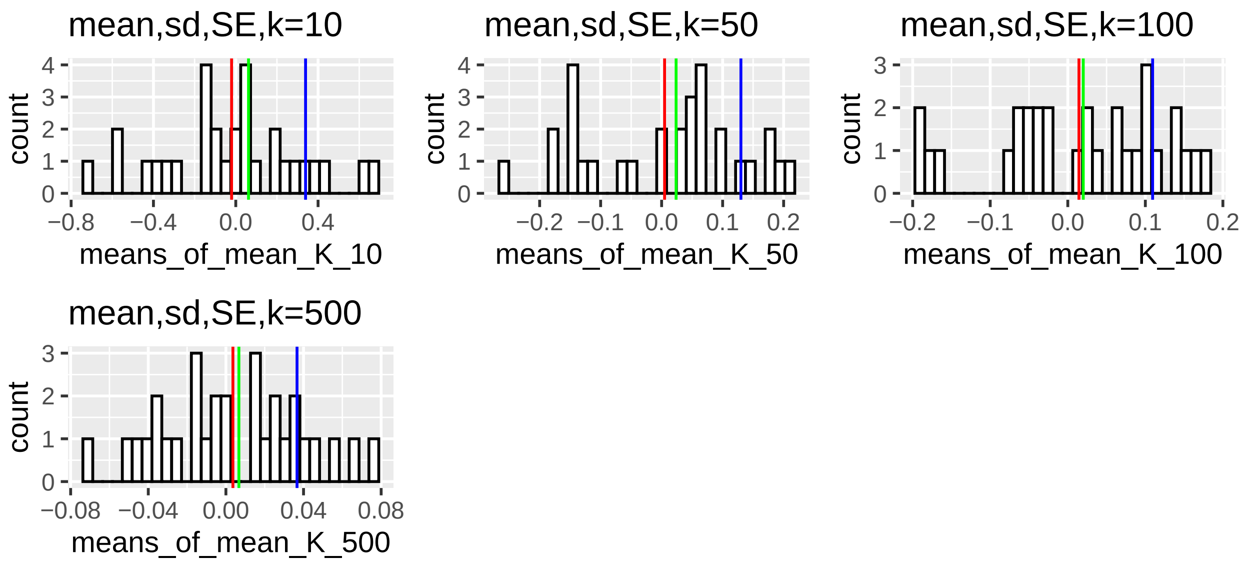

```r
t <- ggplot() +
aes(means_of_mean_K_100) +
  geom_histogram(colour="black", fill="white") +
  geom_vline(xintercept=mean(means_of_mean_K_100), color="red") +
  geom_vline(xintercept=sd(means_of_mean_K_100),color="blue")+
  geom_vline(xintercept=SE(means_of_mean_K_100), color='green')+
  ggtitle(label = 'mean,sd,SE,k=100')

t
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```r
u <- ggplot() +
aes(means_of_mean_K_500) +
  geom_histogram(colour="black", fill="white") +
  geom_vline(xintercept=mean(means_of_mean_K_500), color="red") +
  geom_vline(xintercept=sd(means_of_mean_K_500),color="blue")+
  geom_vline(xintercept=SE(means_of_mean_K_500), color='green') +
  ggtitle(label = 'mean,sd,SE,k=500')
ggarrange(q, z, t,u,  ncol = 3, nrow = 3)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
u
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```