

3858 Tutorial 8 new

Yifan Li, yli2763@uwo.ca

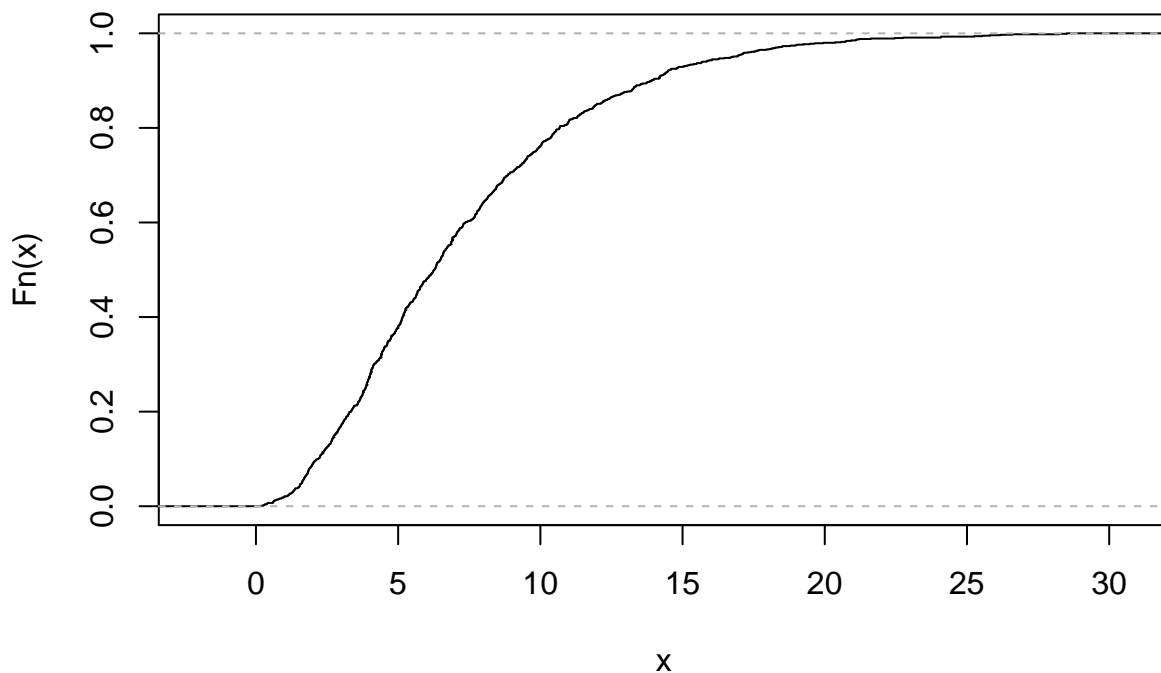
2017-03-21

Parametric vs Non-parametric Bootstrap for Gamma distribution

Here we use the method of moments to get the estimators. Using Delta method, we can prove that the limiting distributions of these estimators are normal distribution (here we make sample size large enough to clarify this point). Furthermore, even if the limiting distribution is not normal, we can still use bootstrap (which is quite robust) to do the estimation.

```
#rain.dat <- scan("http://www.stats.uwo.ca/faculty/kulperger/SS3858/Data/illinois60.txt")
set.seed(31415926)
rain.dat <- rgamma(1000, shape = 2.5, rate = 1/3)
plot(ecdf(rain.dat))
```

ecdf(rain.dat)



```
#method of moment
gamma.est <- function(x){
  mu.1 <- mean(x)
  mu.2 <- mean(x^2)
  var.x <- mu.2-mu.1^2
  lam.hat <- mu.1/(mu.2-mu.1^2)
  alpha.hat <- mu.1*lam.hat
  c(lam.hat, alpha.hat)
}
```

```

est <- gamma.est(rain.dat)
lam <- est[1] #lam and alpha are estimated values
alpha <- est[2]
#try to get better estimate at first (it needs more data)

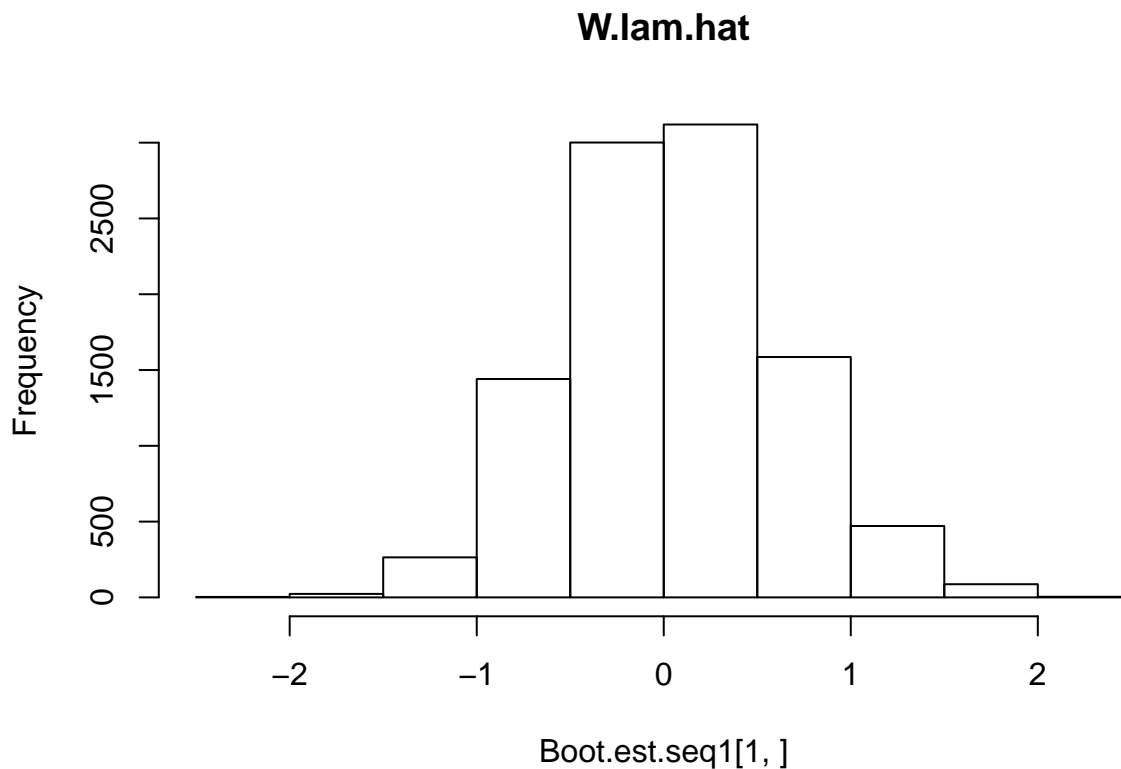
#bootstrap
M <- 1e4
n <- length(rain.dat)

#how to compute the variance of the method of moments
#it is hard to compute, so we just remove the denominator (variance),
#then the limiting distribution will depend on n, but bootstrap will still work
Boot.est <- function(x){
  theta <- gamma.est(x)
  W.lam <- sqrt(n)*(theta[1]-lam)
  W.alpha <- sqrt(n)*(theta[2]-alpha)
  c(W.lam, W.alpha)
}

#parametric
Boot.est.seq1 <- replicate(M, Boot.est(rgamma(n, shape = alpha, rate = lam)))

hist(Boot.est.seq1[1,], main = "W.lam.hat")

```

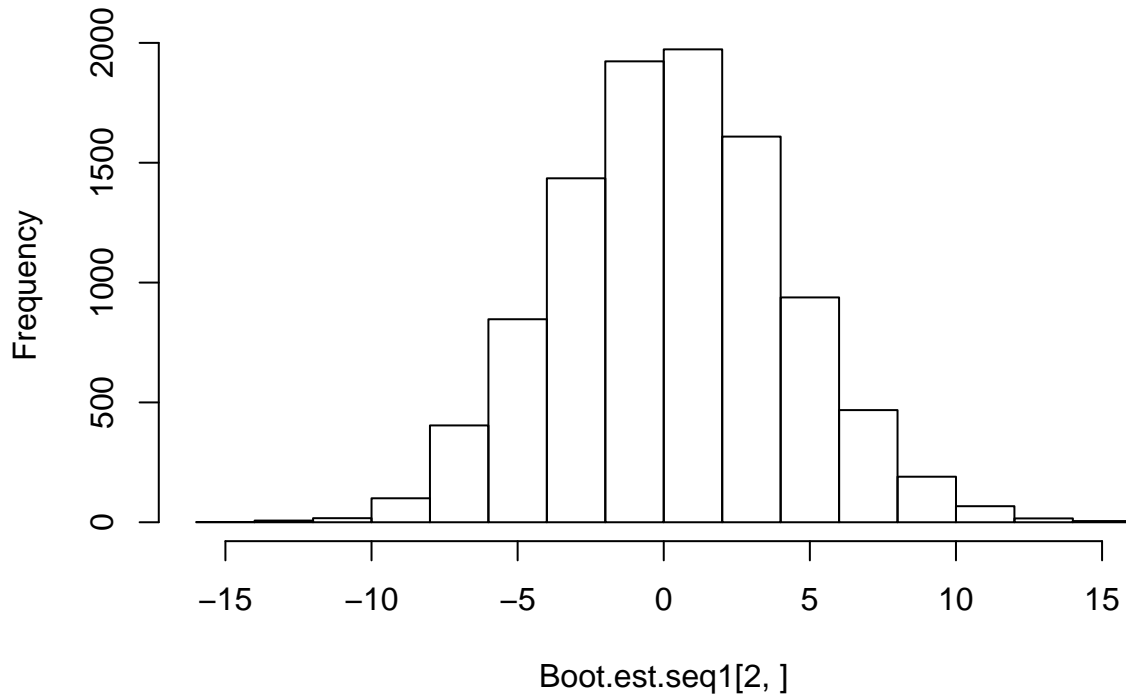


```

hist(Boot.est.seq1[2,], main = "W.alpha.hat")

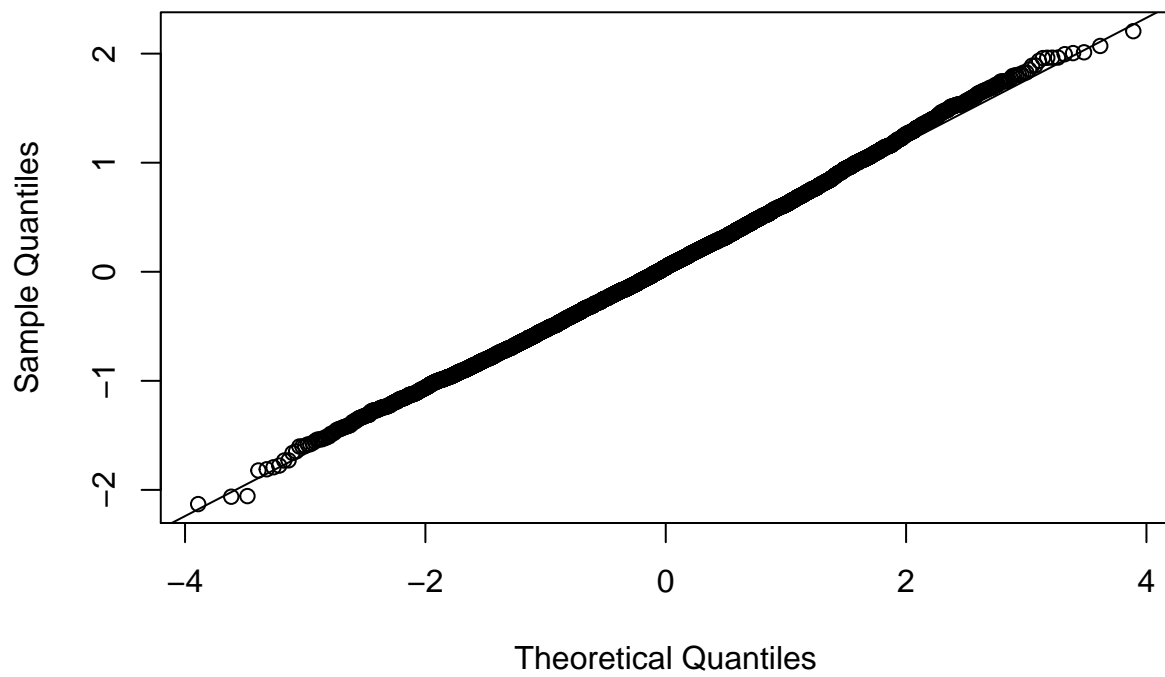
```

W.alpha.hat



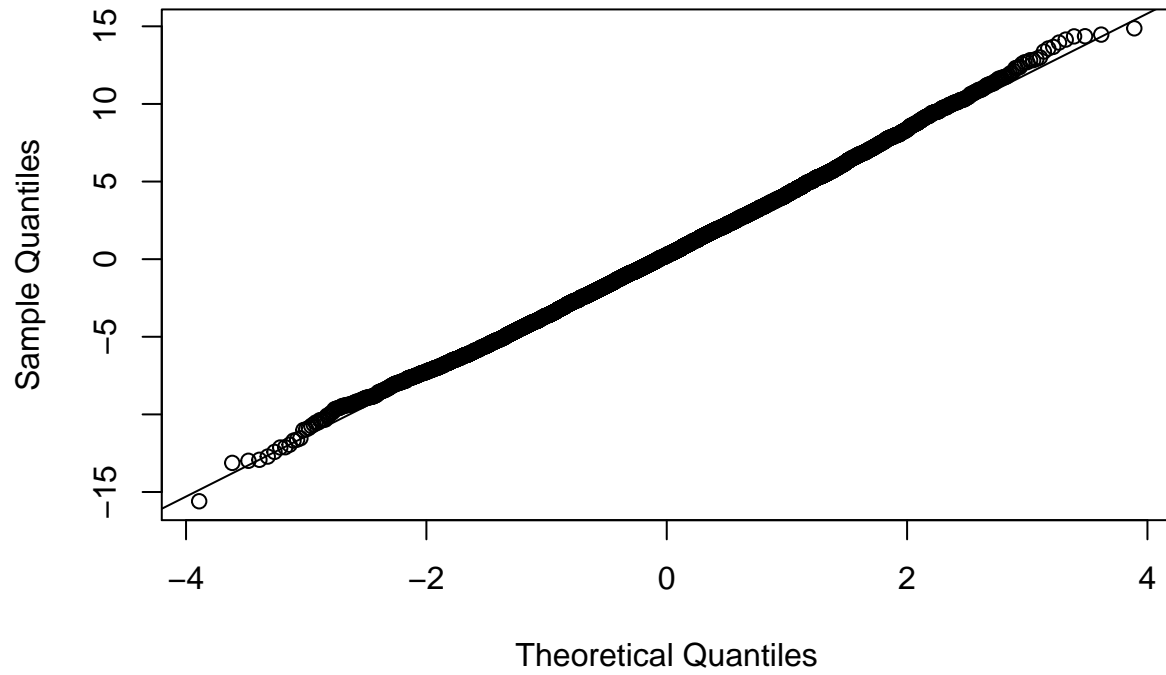
```
qqnorm(Boot.est.seq1[1,]); qqline(Boot.est.seq1[1,])
```

Normal Q-Q Plot



```
qqnorm(Boot.est.seq1[2,]); qqline(Boot.est.seq1[2,])
```

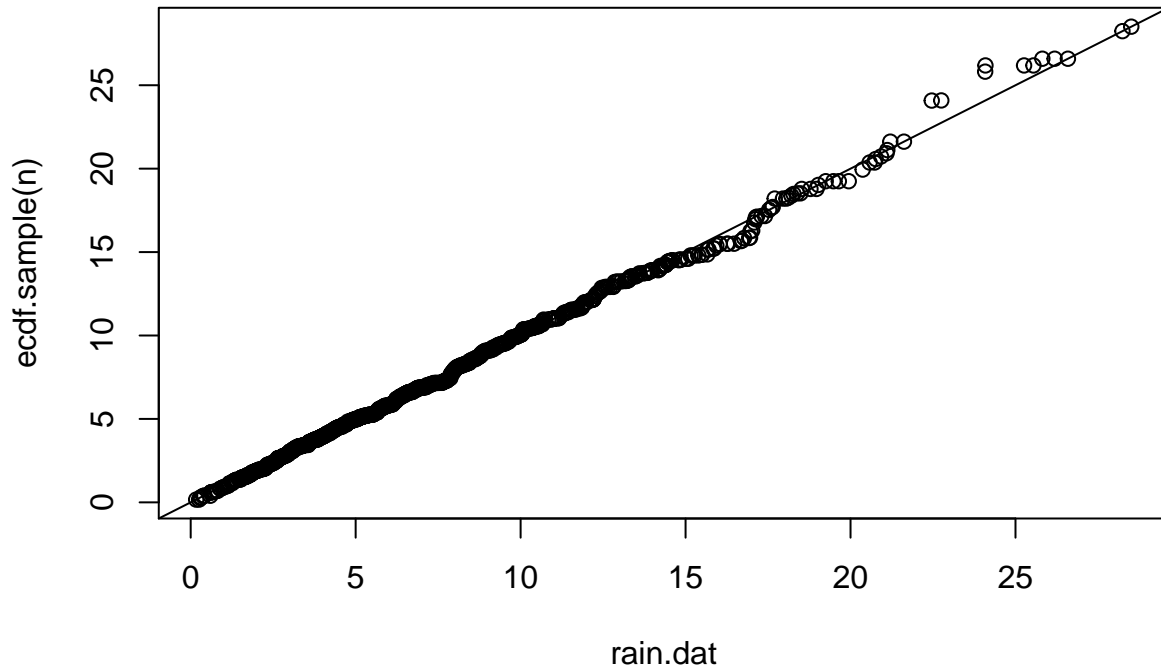
Normal Q-Q Plot



```
#non parametric

ecdf.sample <- function(num=n, data.sample=rain.dat){
  #n=length(rain.dat)
  x <- data.sample
  i.sample = sample(1:n , size = num , replace = T)
  x[i.sample]
}

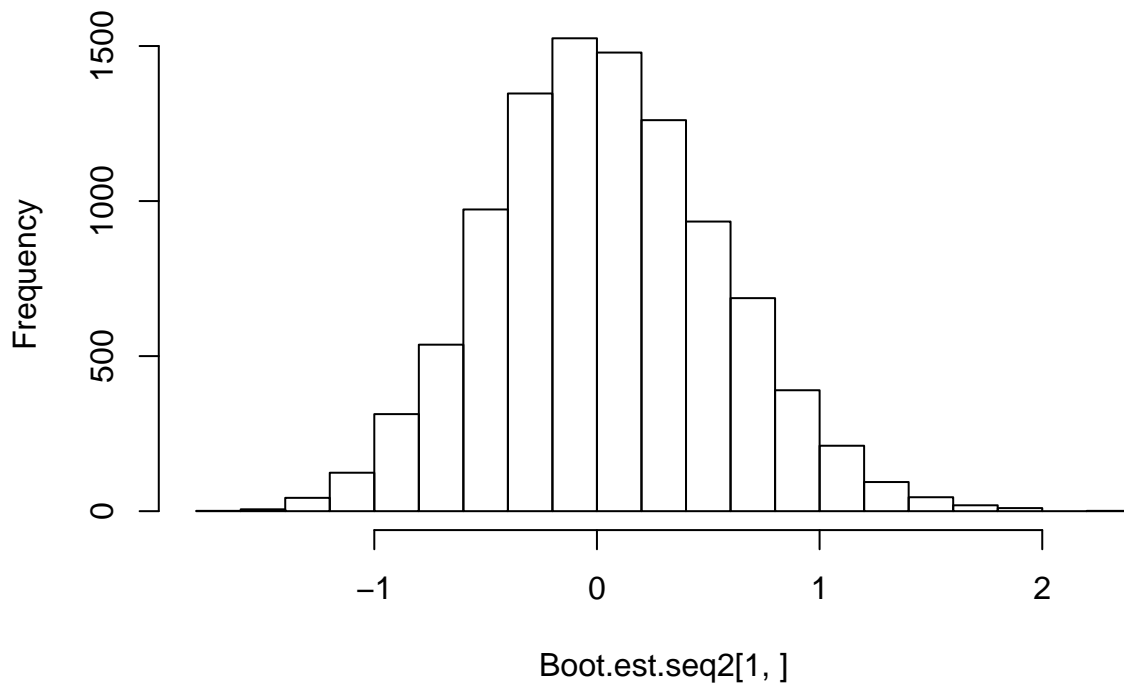
#compare data with X.star
qqplot(rain.dat, ecdf.sample(n)) #n is so small
abline(0,1)
```



```
Boot.est.seq2 <- replicate(M, Boot.est(ecdf.sample(n)))
```

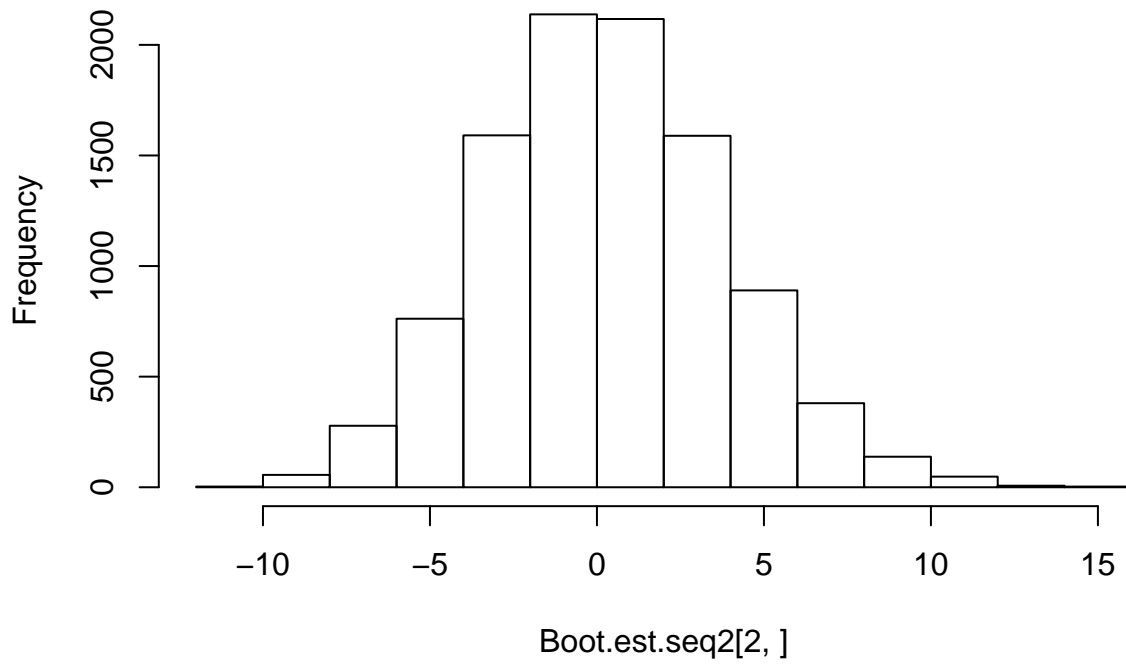
```
hist(Boot.est.seq2[1,], main = "W.lam.hat") #non-par makes it like normal
```

W.lam.hat



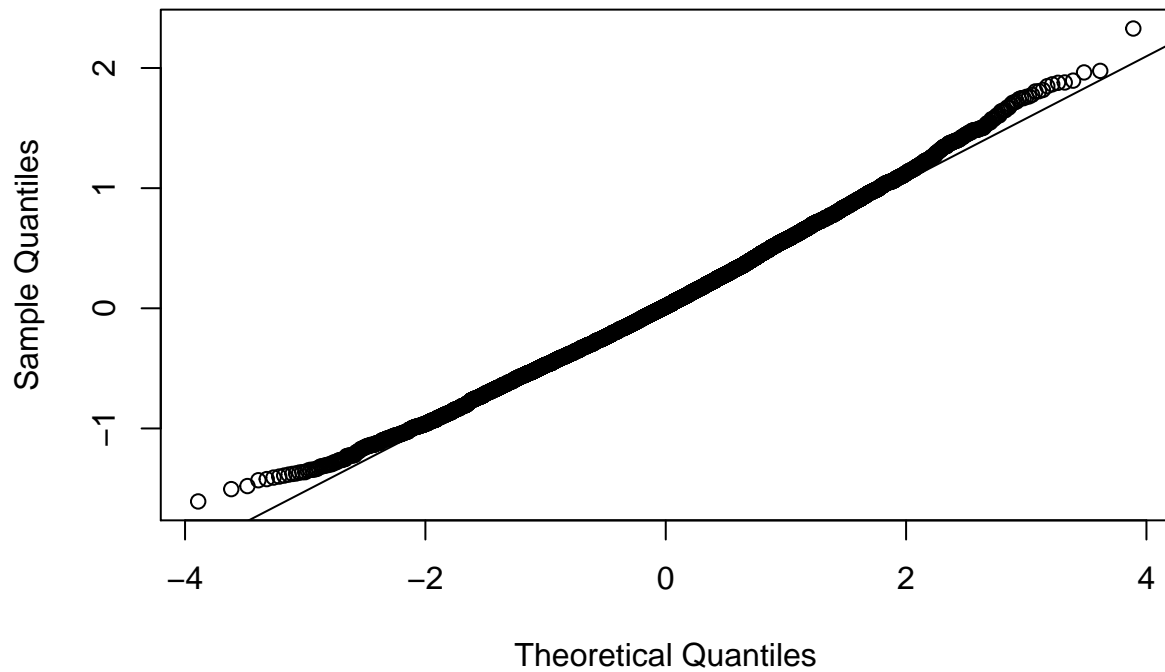
```
hist(Boot.est.seq2[2,], main = "W.alpha.hat") #non-par make it like normal
```

W.alpha.hat



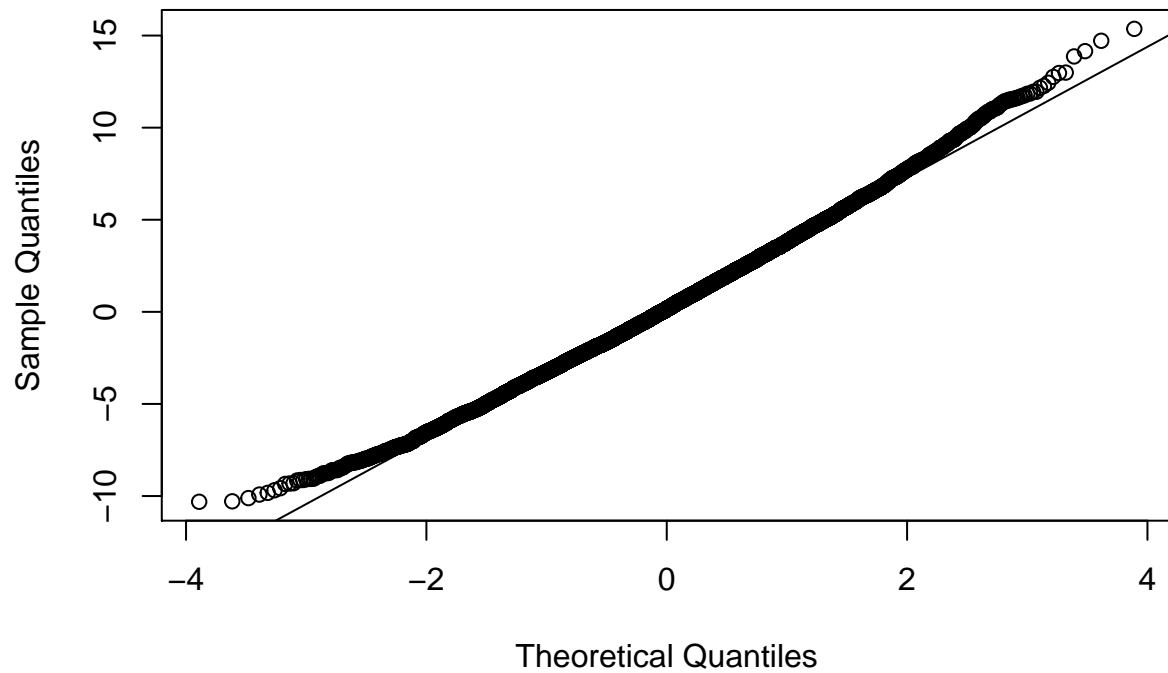
```
#non-par vs normal  
qqnorm(Boot.est.seq2[1,]); qqline(Boot.est.seq2[1,])
```

Normal Q-Q Plot



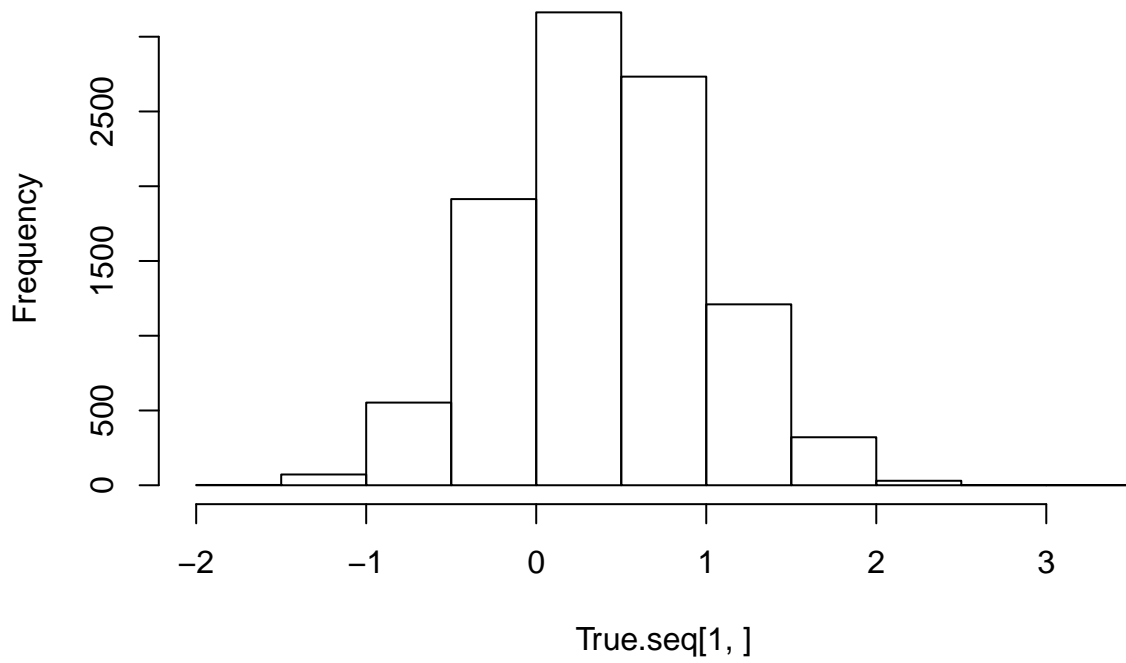
```
qqnorm(Boot.est.seq2[2,]); qqline(Boot.est.seq2[2,])
```

Normal Q-Q Plot



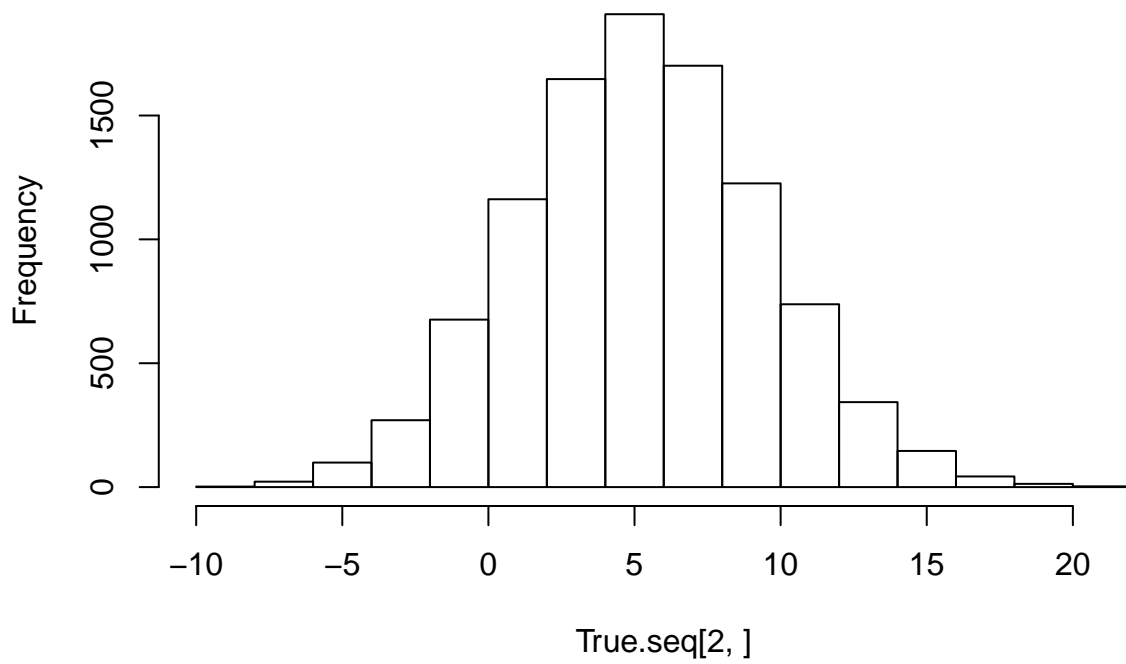
```
#but the true distribution are not normal  
True.seq <- replicate(M, Boot.est(rgamma(n, shape = 2.5, rate = 1/3))) #with true parameter  
hist(True.seq[1,], main = "W.lam.hat true")
```

W.lam.hat true



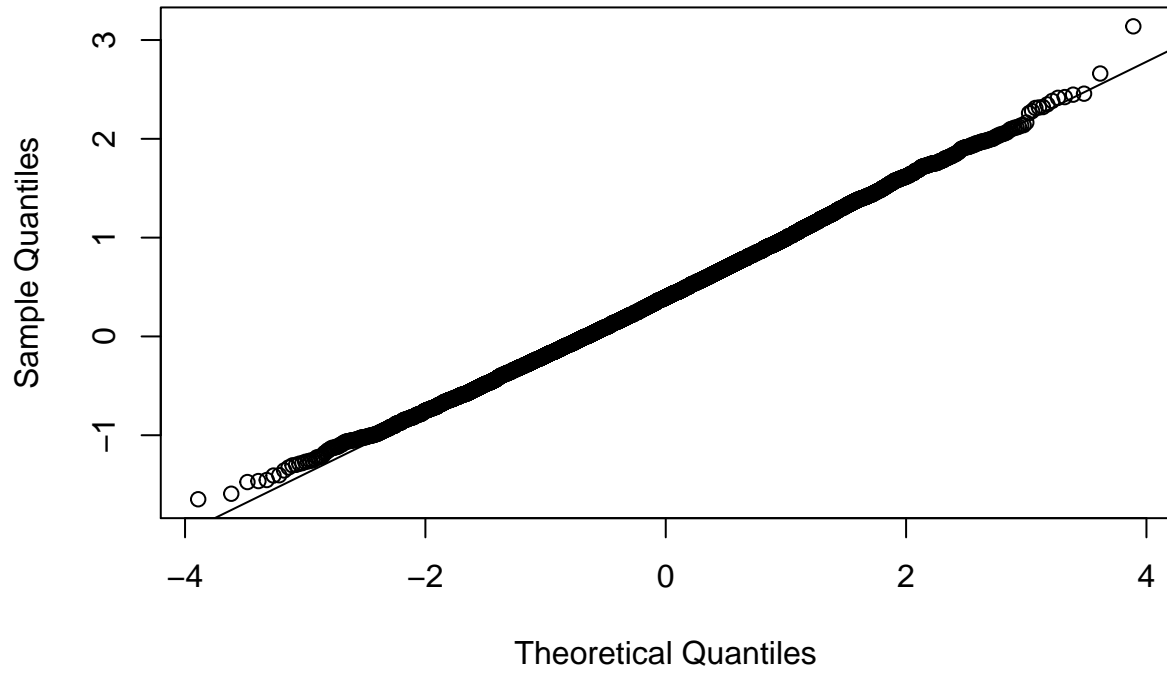
```
hist(True.seq[2,], main = "W.alpha.hat true")
```

W.alpha.hat true



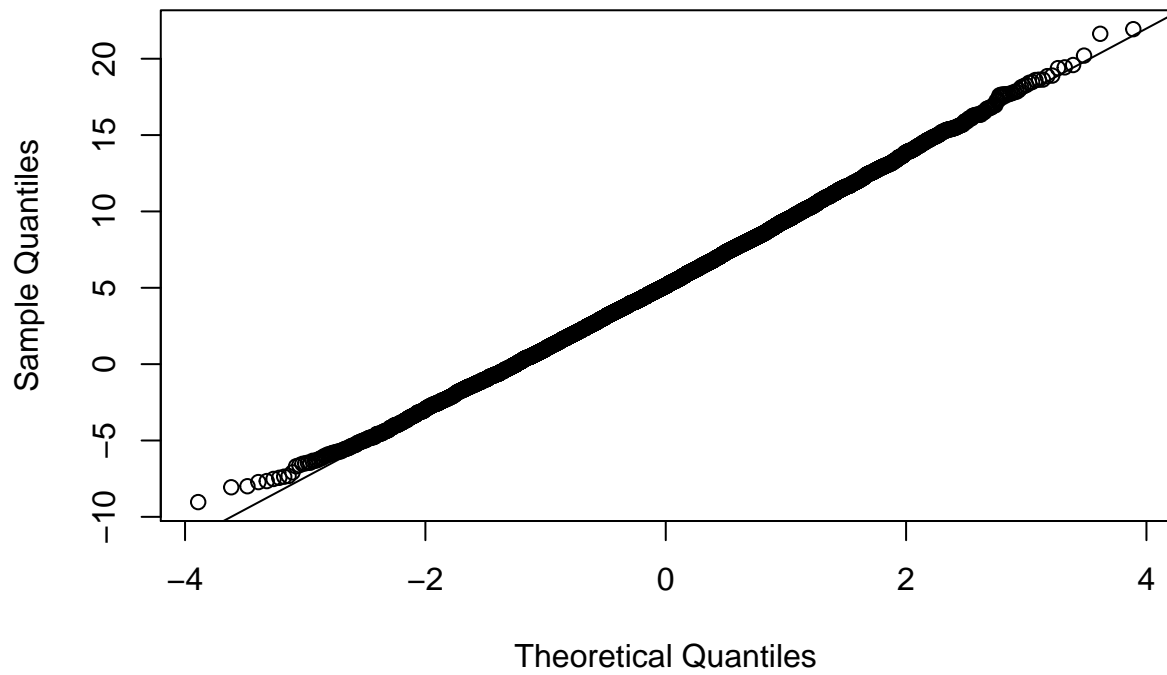
```
#true vs normal  
qqnorm(True.seq[1,]); qqline(True.seq[1,])
```


Normal Q-Q Plot

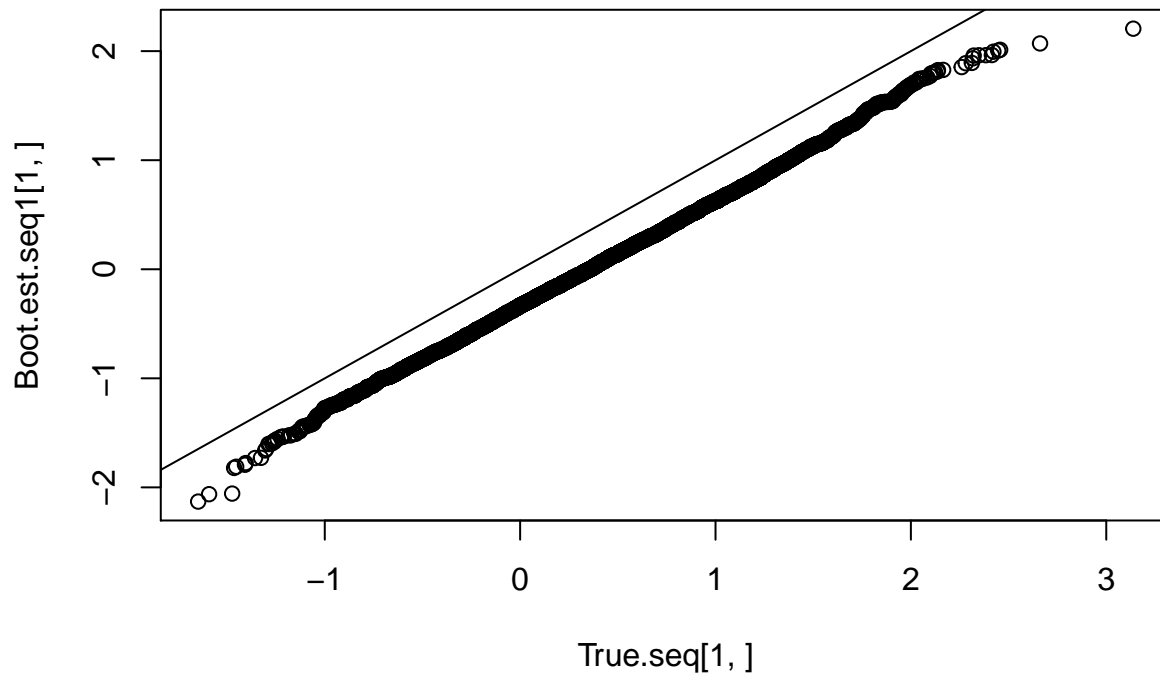


```
qqnorm(True.seq[2,]); qqline(True.seq[2,])
```

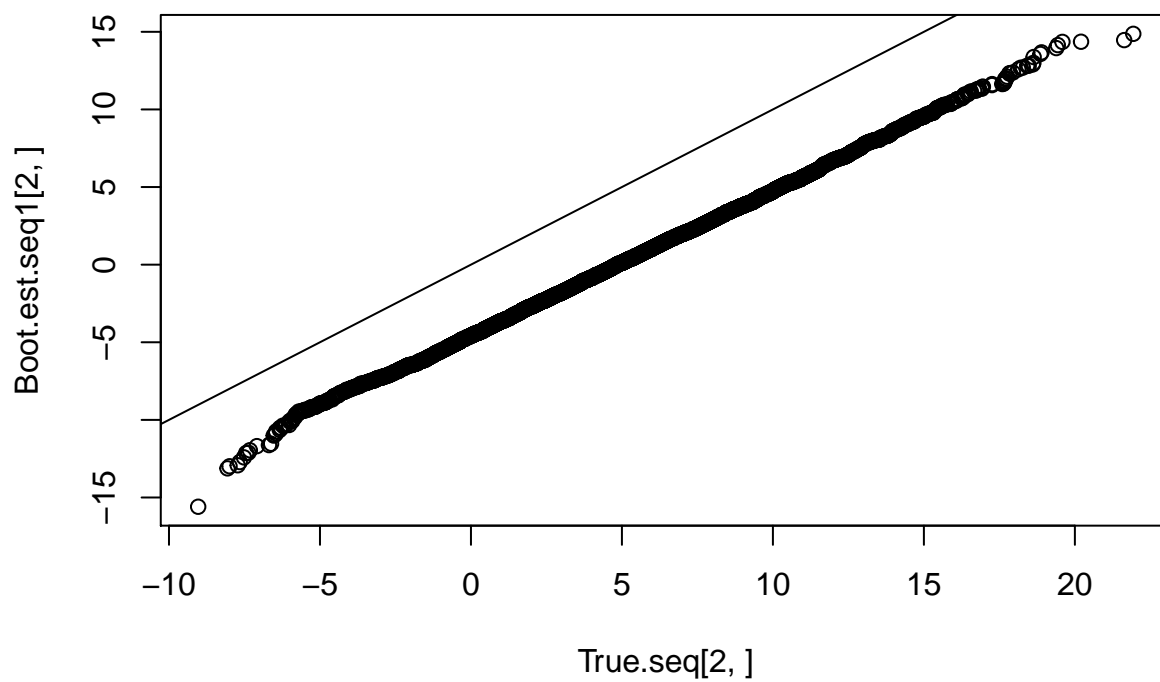
Normal Q-Q Plot



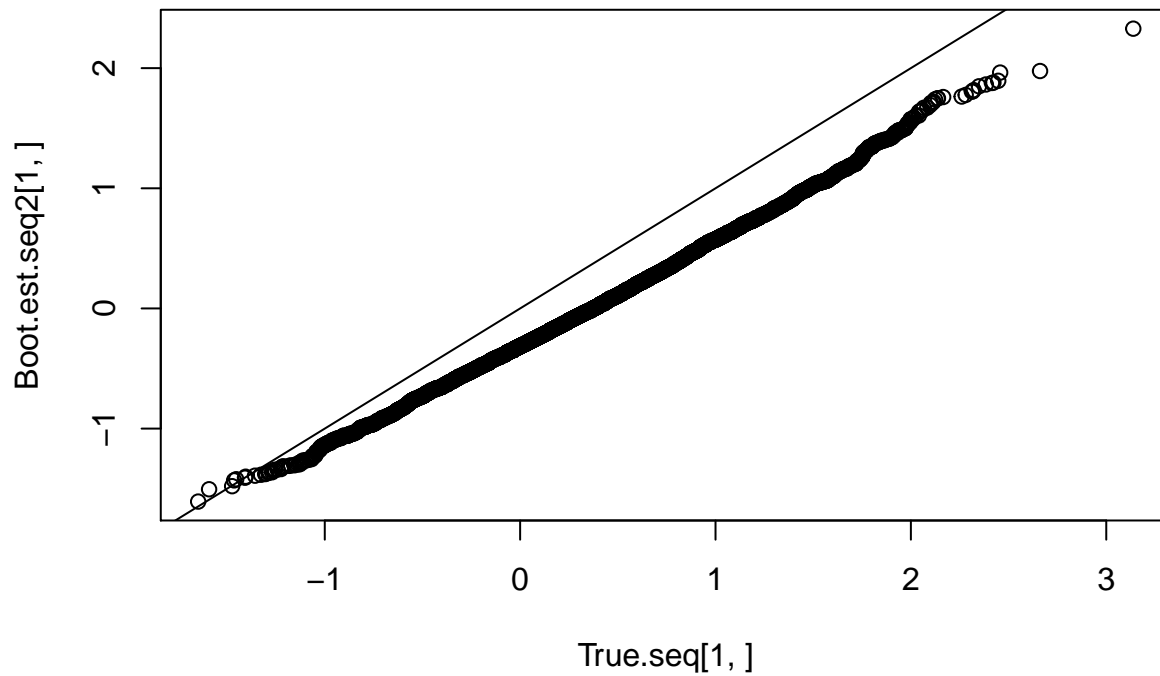
```
#true vs par  
#maybe some translation happens, due to the low accuracy of the estimates  
qqplot(True.seq[1,], Boot.est.seq1[1,]); abline(0,1)
```



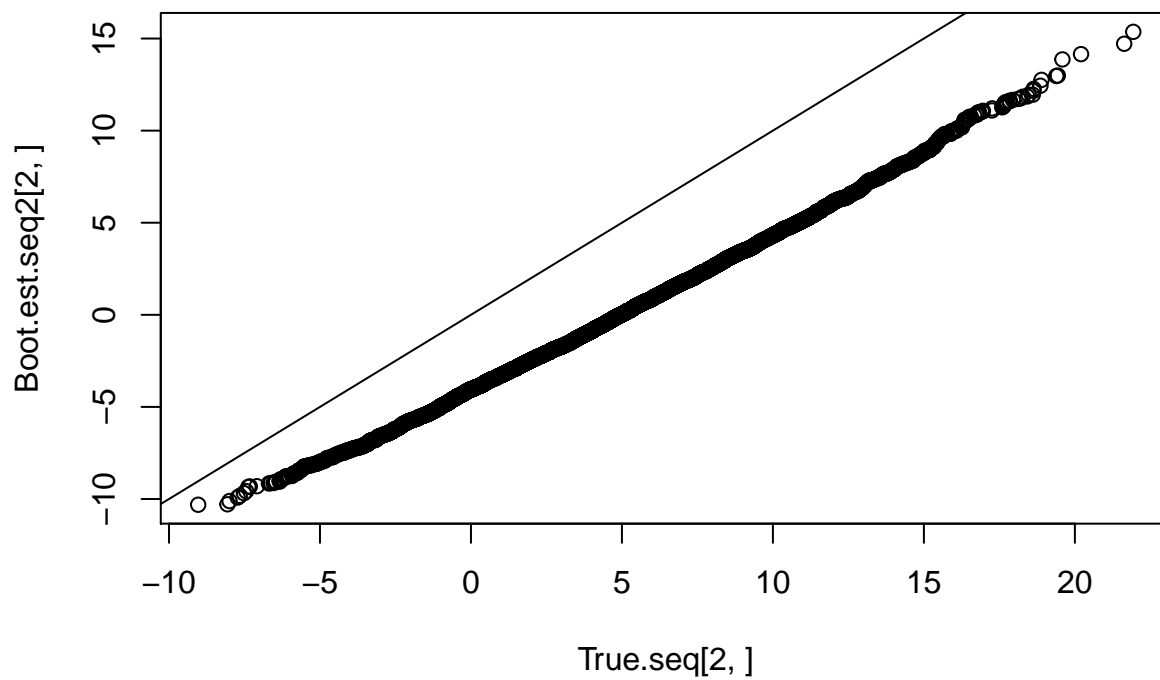
```
qqplot(True.seq[2,], Boot.est.seq1[2,]); abline(0,1)
```



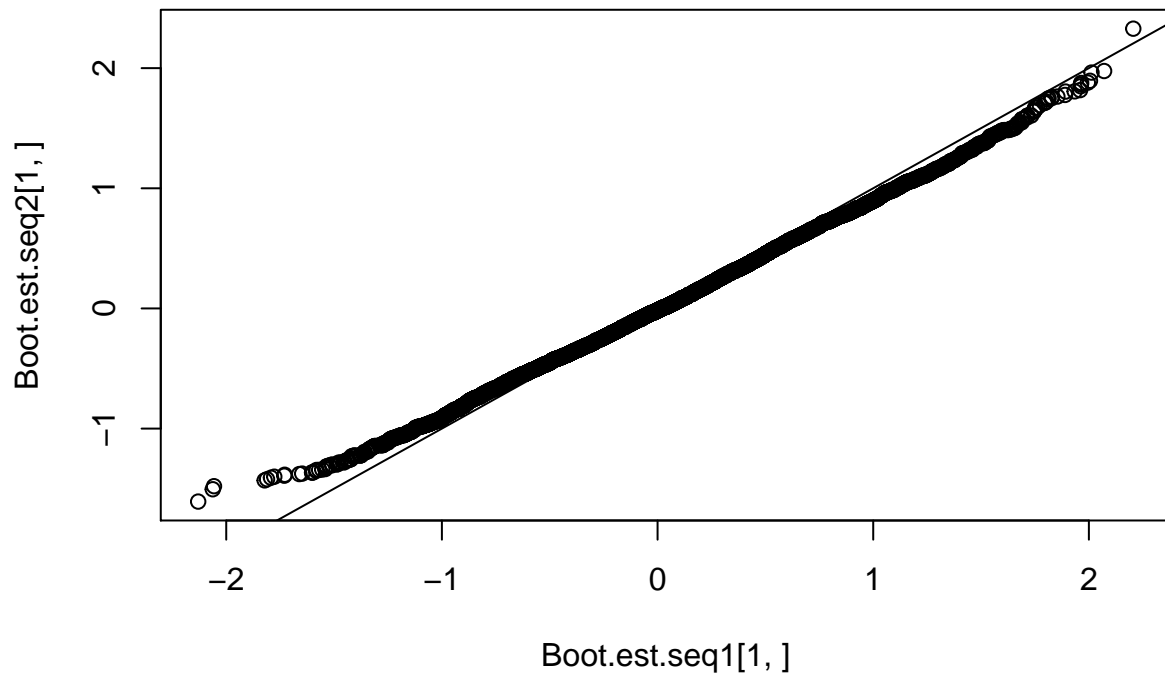
```
#true vs non-par  
qqplot(True.seq[1,], Boot.est.seq2[1,]); abline(0,1)
```



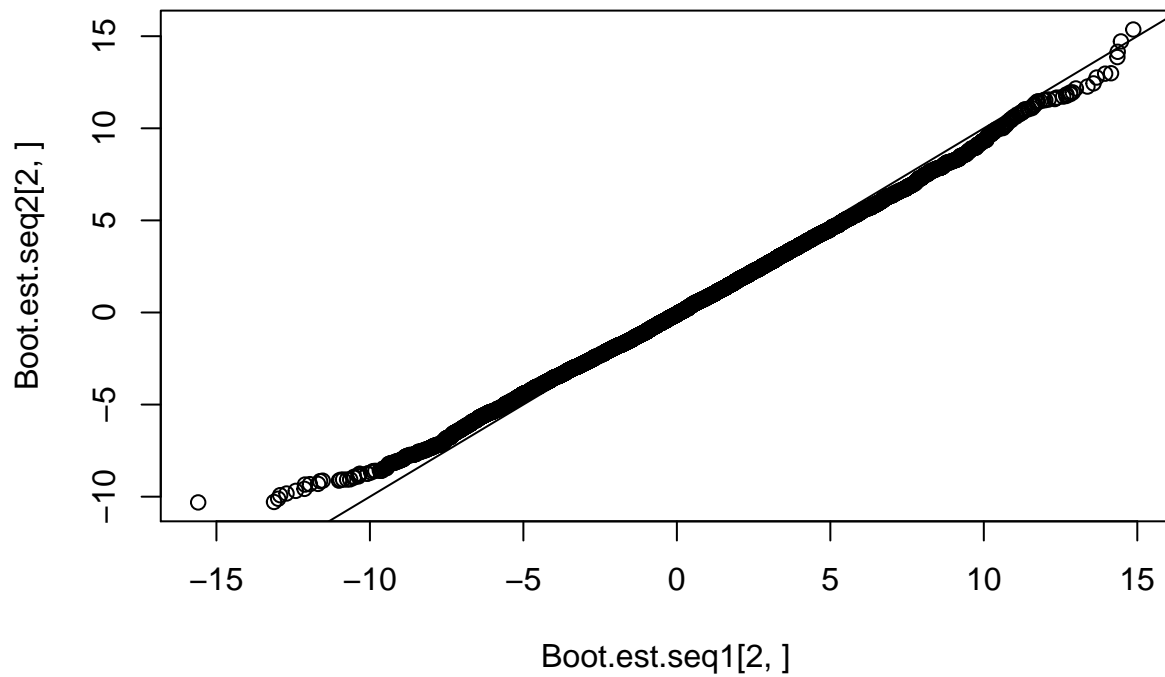
```
qqplot(True.seq[2,], Boot.est.seq2[2,]); abline(0,1)
```



```
#compare par vs non-par  
qqplot(Boot.est.seq1[1, ], Boot.est.seq2[1,]) #they are different  
abline(0 , 1)
```



```
qqplot(Boot.est.seq1[2, ], Boot.est.seq2[2,]) #they are different
abline(0 , 1)
```



```
#ks.test(Boot.est.seq1[2, ], Boot.est.seq2[2,])
```

```
#par quantile
pr <- c(.025 , .975)
q.lam <- quantile(Boot.est.seq1[1,], pr)
q.lam
```

```
##      2.5%      97.5%
```

```
## -1.032446 1.226292
```

```
q.alpha <- quantile(Boot.est.seq1[2,], pr)
q.alpha
```

```
##      2.5%      97.5%
## -7.134576  8.195958
```

```
#confidence interval
#unname(lam - q.lam[c(2,1)] * sqrt(var.lam)/sqrt(n))
#unname(alpha - q.alpha[c(2,1)] * sqrt(var.alpha)/sqrt(n))
```

```
#non-par quantile
pr <- c(.025 , .975)
q.lam <- quantile(Boot.est.seq2[1,], pr)
q.lam
```

```
##      2.5%      97.5%
## -0.9416078  1.0991828
```

```
q.alpha <- quantile(Boot.est.seq2[2,], pr)
q.alpha
```

```
##      2.5%      97.5%
## -6.434341  7.556508
```

```
#confidence interval
#unname(lam - q.lam[c(2,1)] * sqrt(var.lam)/sqrt(n))
#unname(alpha - q.alpha[c(2,1)] * sqrt(var.alpha)/sqrt(n))
```