writeup

Probability and Inference — Deliverable 06

Order Statistics

Rastko Stojsin

Which quantiles of a continuous distribution can be estimated with more precision?

The median is an important quantity in data analysis. It represents the middle value of the data distribution. Estimates of the median, however, have a degree of uncertainty because . . .

(a) the estimates are calculated from a finite sample and

(b) the data distribution of the underlying data is generally unknown.

One important roles of a data scientist is to quantify and to communicate the degree of uncertainty in his or her data analysis.

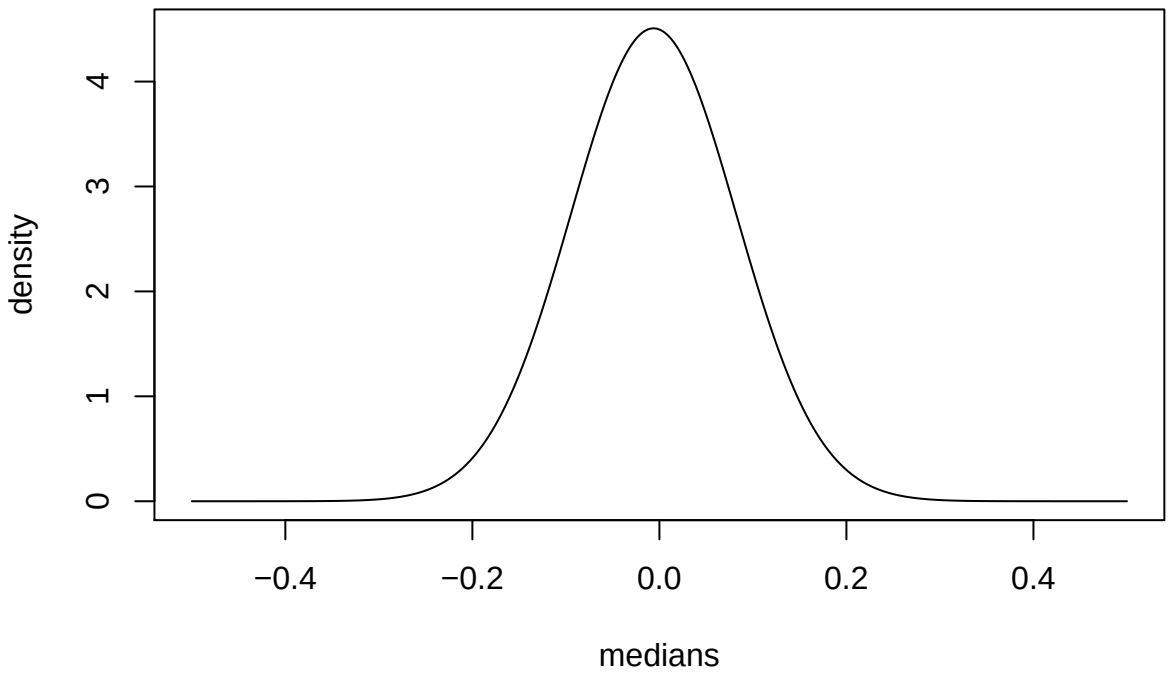
In this assignment, I will answer a series of questions related to the variation of the median and range other quantiles.

Questions

Q1

Begin with the median from a sample of N=200 from the standard normal distribution. Write an R function that is the density function for the median in this sample. Note that the 100th order statistic is approximately the median, and use the order statistic formula discussed in class. Generate a plot of the function.

```
dorder <- function(x){
  100*
    choose(200, 100)*
    pnorm(x)^(100-1)*
    (1-pnorm(x))^(200-100)*
    dnorm(x)
}
# median of standard normal distribution
plot(seq(-0.5,0.5, by=0.0001), dorder(seq(-0.5,0.5, by=0.0001)), type = "l", xlab = "medians", ylab = "
```



This graph shows us the density function for the median of a standard normal distribution.

Q2

Write an R function that is the probability function for the median in this sample. Use the order statistic formula discussed in class. Generate a plot of the function.

```
porder <- function(x){
  pbinom(100-1, 200, pnorm(x), lower.tail = FALSE)
}
```

Q3

Write an R function that is the quantile function for the median in this sample. (You have several options for how to write this function.) Generate a plot of the function.

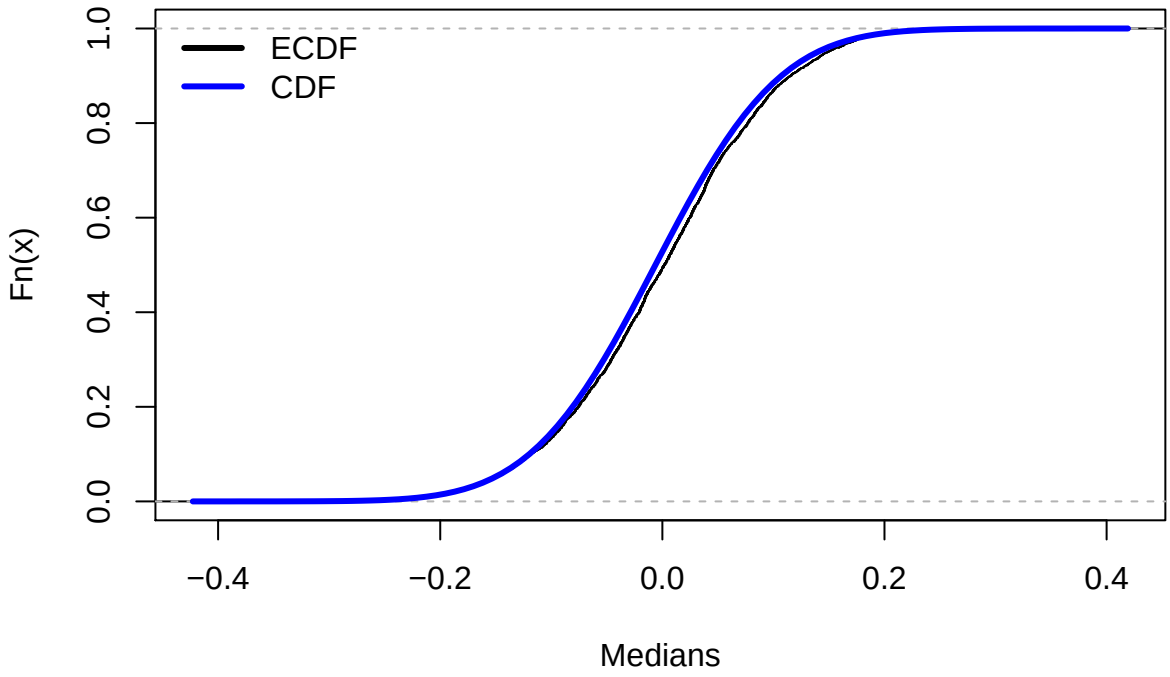
```
qorder <- function(p){
  out <- p
  for(i in seq_along(p)){
    out[i] <- uniroot(function(x){porder(x) - p[i]}, c(-100,100))$root
  }
  out
}
```

Q4

Simulate the sampling distribution for the median as you did in the previous deliverable. Create a plot of the empirical CDF (ECDF). Overlay the plot of the ECDF with a plot of the CDF.

```
N <- 200
M <- 5000
out <- array(rnorm(M*N), c(M,N))
meds <- apply(out,1,median)
e1 <- ecdf(meds)

plot(e1, xlab = "Medians", main = "")
curve(porder, add = TRUE, col = "blue", lwd = 3)
legend("topleft", c("ECDF","CDF"), lwd = 3, col = c("black","blue"), bty = "n")
```

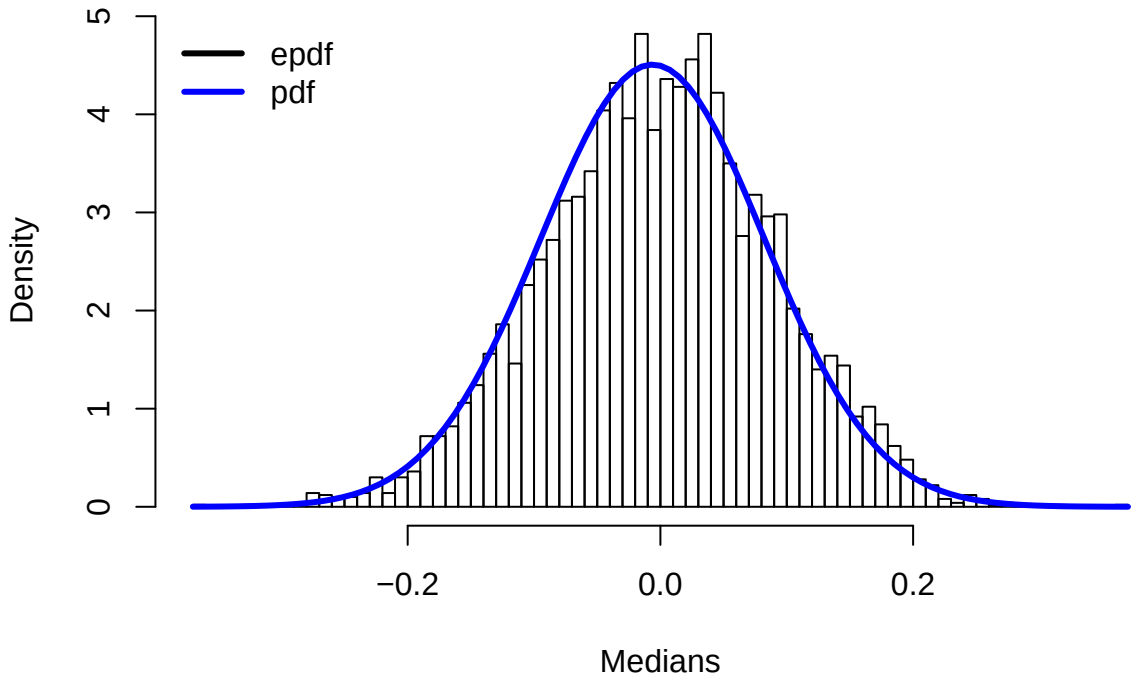


The black line is the empirical CDF calculation from the distribution of the medians. The CDF is mathmatically determined and is in blue. As we can see we can get the CDF curve quite well empirically.

Q5

Using the simulated sampling distribution from the previous question, create a histogram (on the density scale). Overlay the histogram with a plot of the density function.

```
hist(meds, xlab = "Medians", main = "", breaks = 100, freq = FALSE)
curve(dorder, add = TRUE, col = "blue", lwd = 3)
legend("topleft", c("epdf","pdf"), lwd = 3, col = c("black","blue"), bty = "n")
```



The bars indicate our simulated medians histogram. This is ploted against the actual mathmetically determined PDF line. Again the simulated medians histogram lines up pretty well with where it is supposed to be mathmatically.

Q6

One very common way to compare a random sample to a theoretical candidate distribution is the QQ plot. It is created by ploting quantiles of the theoretical distribution on the x-axis and empirical quantiles from the sample on the y-axis.

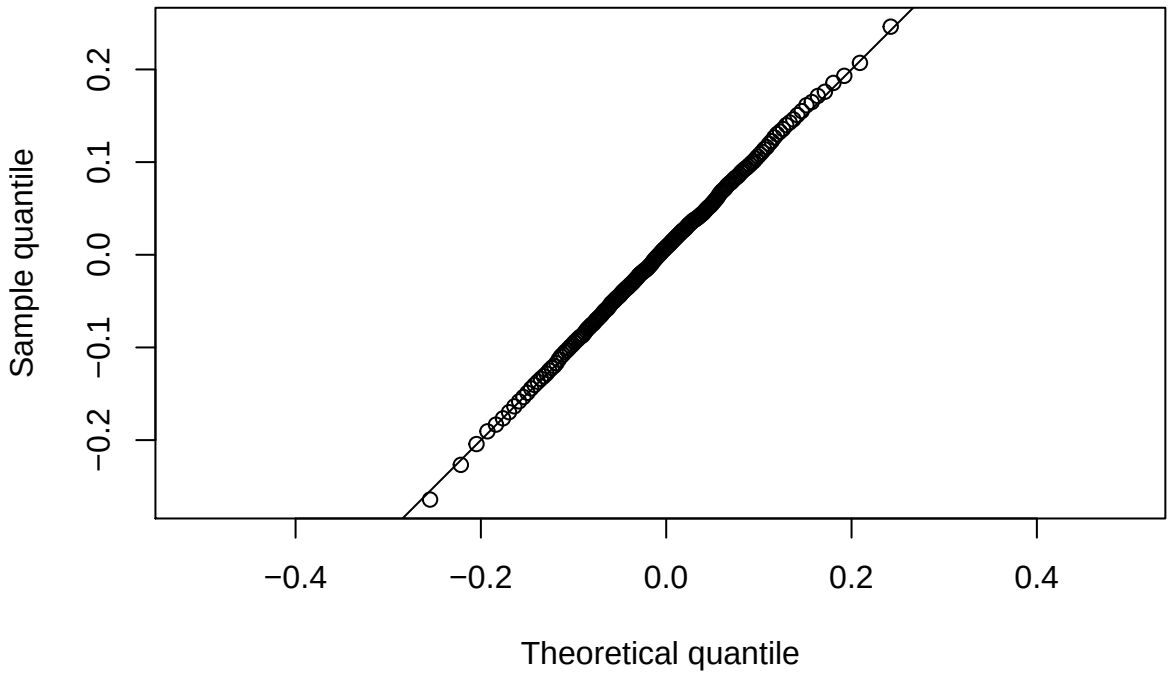
If sample and theoretical quantiles come from the same distribution, then the plotted points will fall along the line y=x, approximately.

For the assigment, generate a QQ plot for the simulated data of the median relative to the known sampling distribution of the median.

```
set.seed(9123)
p <- ppoints(200)
x <- qorder(p)
```

```r
y <- quantile(meds, probs = p)

plot(x,y, asp = 1, xlab = "Theoretical quantile", ylab = "Sample quantile")
abline(0,1)
```



The simulated medians line up to the sampling distribution of the mean which indicates that we choose the correct underlying distribution.

Q7

Modify the dorder, porder, and qorder functions so that the functions take a new parameter k (for the kth order statistic) so that the functions will work for any order statistic and not just the median.**

```r
# dorder with k
dorder <- function(x, k){
  k*
    choose(200, k)*
    pnorm(x)^(k-1)*
    (1-pnorm(x))^(200-k)*
    dnorm(x)
}

# porder with k
porder <- function(x, k){
```

```
  pbinom(k-1, 200, pnorm(x), lower.tail = FALSE)
}

# qorder with k
qorder <- function(p, k){
  out <- p
  for(i in seq_along(p)){
    out[i] <- uniroot(function(x){porder(x, k) - p[i]}, c(-100,100))$root
  }
  out
}
```

Q8

Modify the dorder, porder, and qorder functions so that the functions take new parameters dist and ... so that the functions will work for any continuous distribution that has d and p functions defined in R.

```
# dorder with k and dist type
dorder <- function(x, k, n, dist = "norm", ...){
  pf <- get(paste0("p", dist))
  df <- get(paste0("d", dist))
  k*
    choose(n, k)*
    pf(x, ...)^(k-1)*
    (1-pf(x, ...))^(n-k)*
    df(x, ...)
}

# porder with k and dist type
porder <- function(x, k, n, dist = "norm", ...){
  pf <- get(paste0("p", dist))
  pbinom(k-1, n, pf(x, ...), lower.tail = FALSE)
}

# qorder with k and dist type
qorder <- function(p, k, n, dist = "norm", ...){
  out <- p
  for(i in seq_along(p)){
    out[i] <- uniroot(function(x){porder(x, k, n, dist, ...) - p[i]}, c(-100,100))$root
  }
  out
}
```

Q9

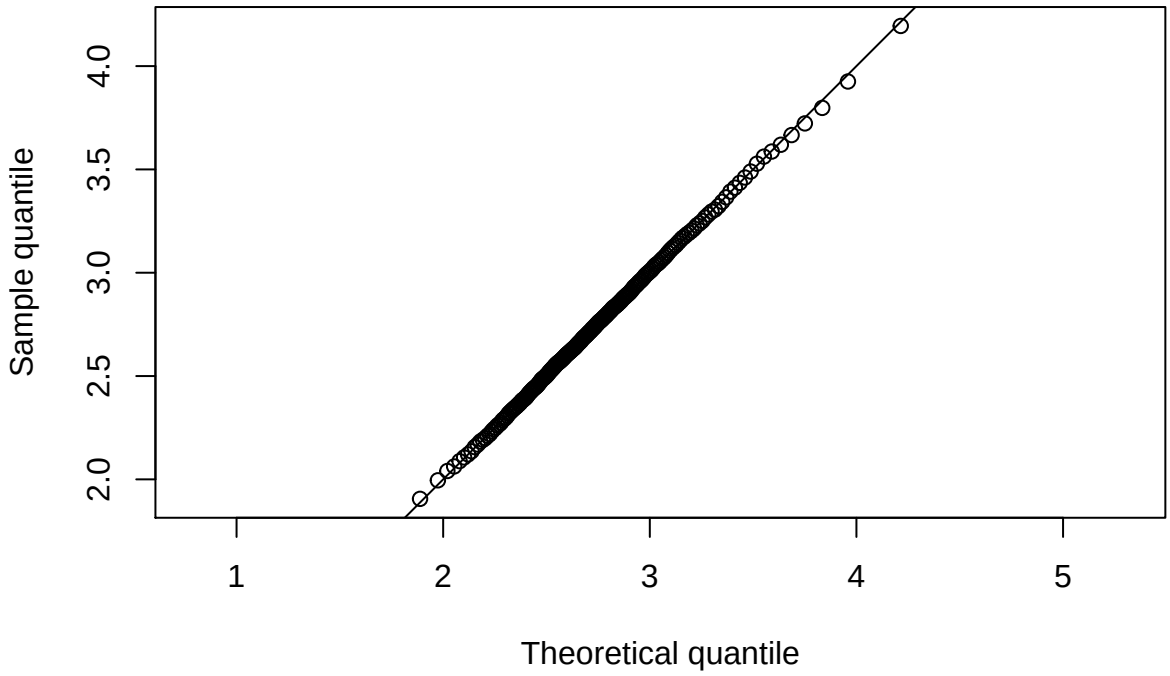
Generate the QQ plot for simulated data from the sampling distribution of the sample max and the theoretical largest order statistic distribution.

```
set.seed(810)
N <- 200
M <- 5000
```

```r
out <- array(rnorm(M*N), c(M,N))
maxs <- apply(out,1,max)

p <- ppoints(200)
x <- qorder(p, 200, 200)
y <- quantile(maxs, probs = p)

plot(x,y, asp = 1, xlab = "Theoretical quantile", ylab = "Sample quantile")
abline(0,1)
```

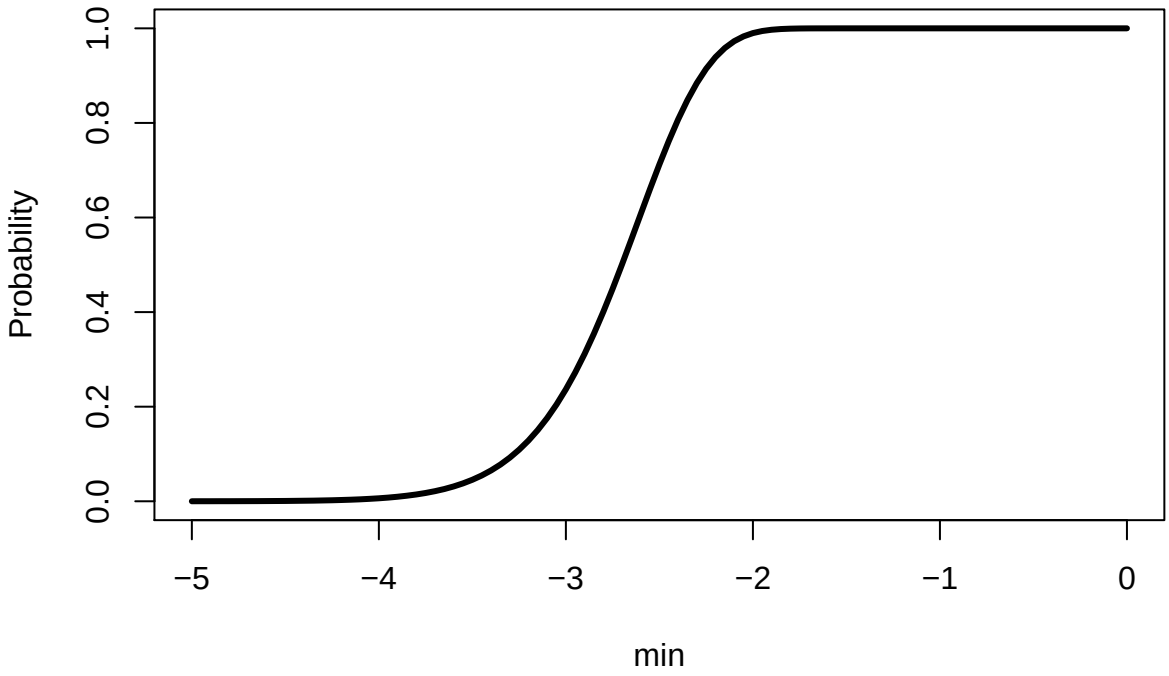


The simulated maxs line up to the sampling distribution of the maximum order statistic which indicates that we choose the correct underlying distribution. Albeit this one is a little bit worse fit than the theoretical vs sample mean - this might be because there is little data at ends of distribution leading to higher variance.

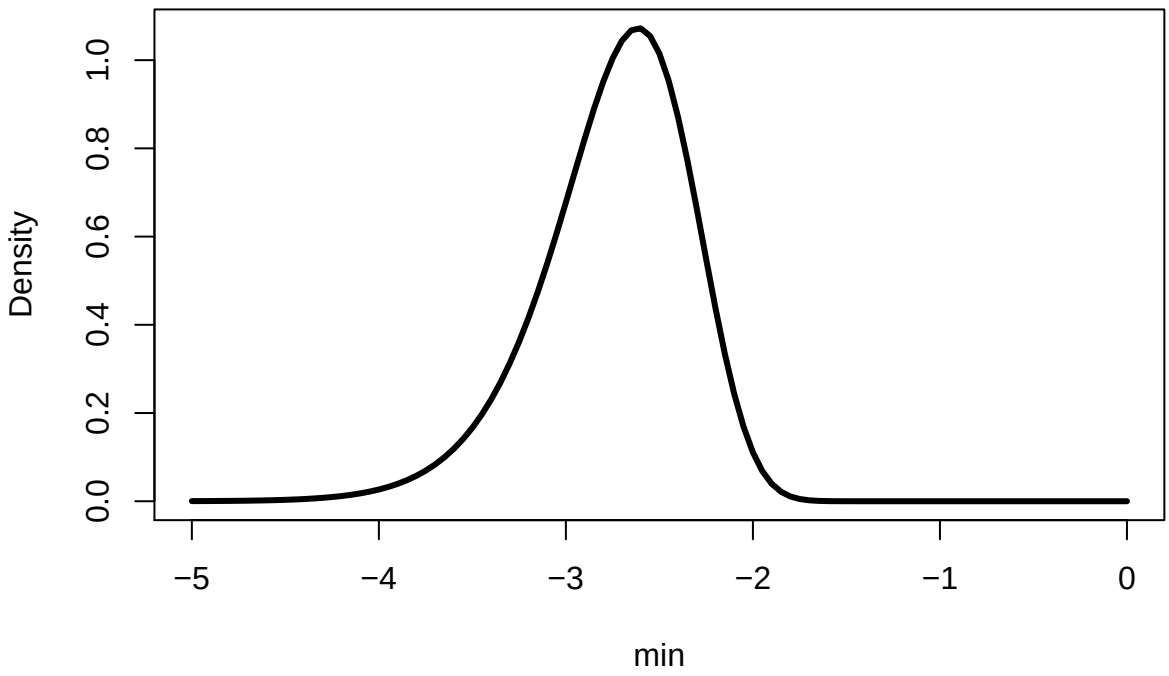
Q10

Use the newly modified functions to plot the probability and density functions for the sample min (N=200).

```r
curve(porder(x, 1, 200), -5,0, ylab = "Probability", xlab = "min", lwd = 3)
```

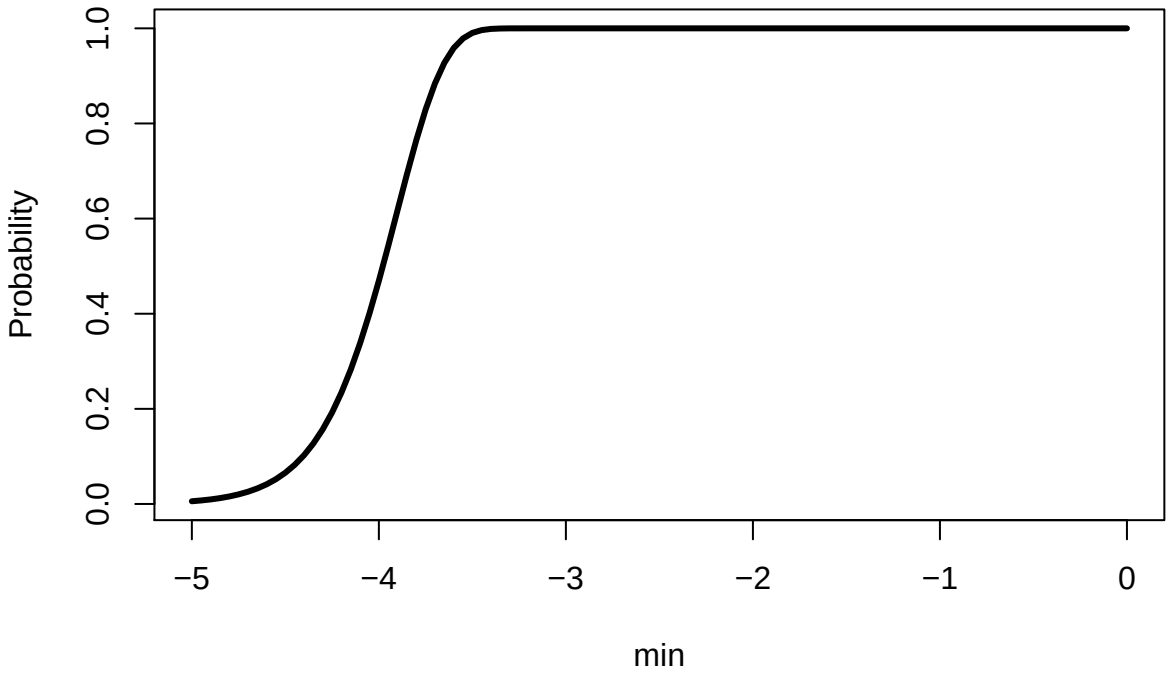



```
curve(dorder(x, 1, 200), -5,0, ylab = "Density", xlab = "min", lwd = 3)
```

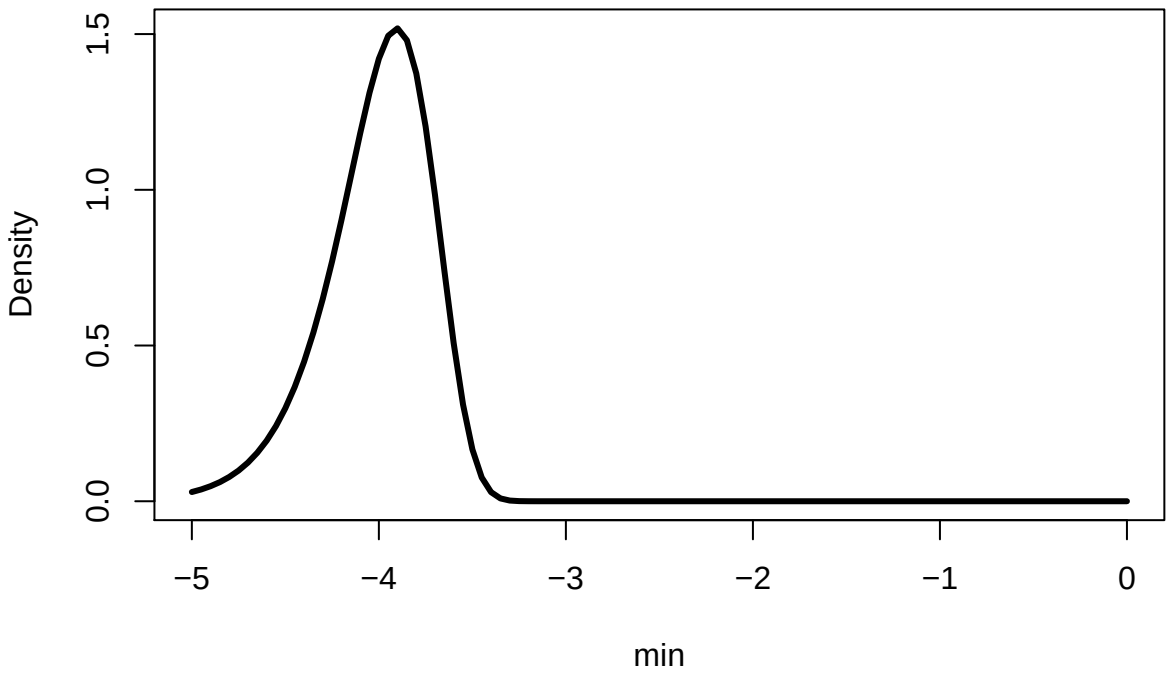


At 200 samples the minimum settles at around -2.8. If we increase N the minimum should slide to the left (more negative) because there are more oprotunities to get values at the extreme negative tail. Lets test this at 20000 samples below.

```
curve(porder(x, 1, 20000), -5,0, ylab = "Probability", xlab = "min", lwd = 3)
```




```
curve(dorder(x, 1, 20000), -5,0, ylab = "Density", xlab = "min", lwd = 3)
```



As expected the minimums have now shifted to near -4 as we have more posibility of extreme values.