rv_05_mai

2023-05-06

```
# source our backend code,
# If you encounter "file not found": make sure to set the working directory correctly
source("./ducharme_methods.R")
```

```
##
## Attaching package: 'rmutil'
```

```
## The following object is masked from 'package:stats':
##
##     nobs
```

```
## The following objects are masked from 'package:base':
##
##     as.data.frame, units
```

Statistic with data from Professor

```
source("./ducharme_methods.R")

X=c(0.645612, 0.66022, 0.555223, 0.368246, 0.801863, 0.188038, 0.837897,
     0.872886, 0.179029, 0.312767, 0.184966, 0.782871, 0.196528, 0.111166,
     0.950258, 0.0409111, 0.0230874, 0.737858, 0.381488, 0.693835,
     0.271724, 0.779804, 0.624836, 0.0160502, 0.831894, 0.850674,
     0.624769, 0.214307, 0.590515, 0.273547, 0.716782, 0.510014, 0.608093,
     0.342415, 0.458901, 0.969161, 0.0731037, 0.445583, 0.939654,
     0.918209, 0.186333, 0.29723, 0.741534, 0.802549, 0.71817, 0.848047,
     0.430392, 0.26375, 0.697366, 0.165481)

Y=c(0.65977, 0.0316515, 0.872726, 0.944487, 0.595143, 0.770168,
     0.923727, 0.256481, 0.372617, 0.430102, 0.721855, 0.244513, 0.635611,
     0.651448, 0.52978, 0.516194, 0.351899, 0.360599, 0.800771, 0.533902,
     0.342824, 0.14494, 0.442621, 0.903258, 0.508918, 0.70013, 0.941215,
     0.0815881, 0.971915, 0.918777, 0.43683, 0.814787, 0.711265,
     0.0630177, 0.209283, 0.212617, 0.532894, 0.734362, 0.777764,
     0.162219, 0.428761, 0.430543, 0.571107, 0.568565, 0.666056, 0.96149,
     0.886778, 0.00261489, 0.541221, 0.717969)

D(X,Y, num_e = 1000, d=1, k=2)
```

```
##          [,1]
## [1,] 1.964824
```

```
# BUG NOTE:
# whan calculating M for equidistal points (x), should we then divide by num_e (1000)
# or by n = 50. For me using num_e seemed more resonable, but then we obtained a different result
```

Calculating the quantiles

```
source("./ducharme_methods.R")

M = 10000
n = 50
H0_data = D_dist_H0(M,n,0.95)
```

```
## [1] "Generating data, this might take some time ..."
```

```
print(paste(" quantile M=1000 ", H0_data$sc_s[1000]))
```

```
## [1] " quantile M=1000  23.8004750929811"
```

```
print(paste(" quantile M=5000 ", H0_data$sc_s[5000]))
```

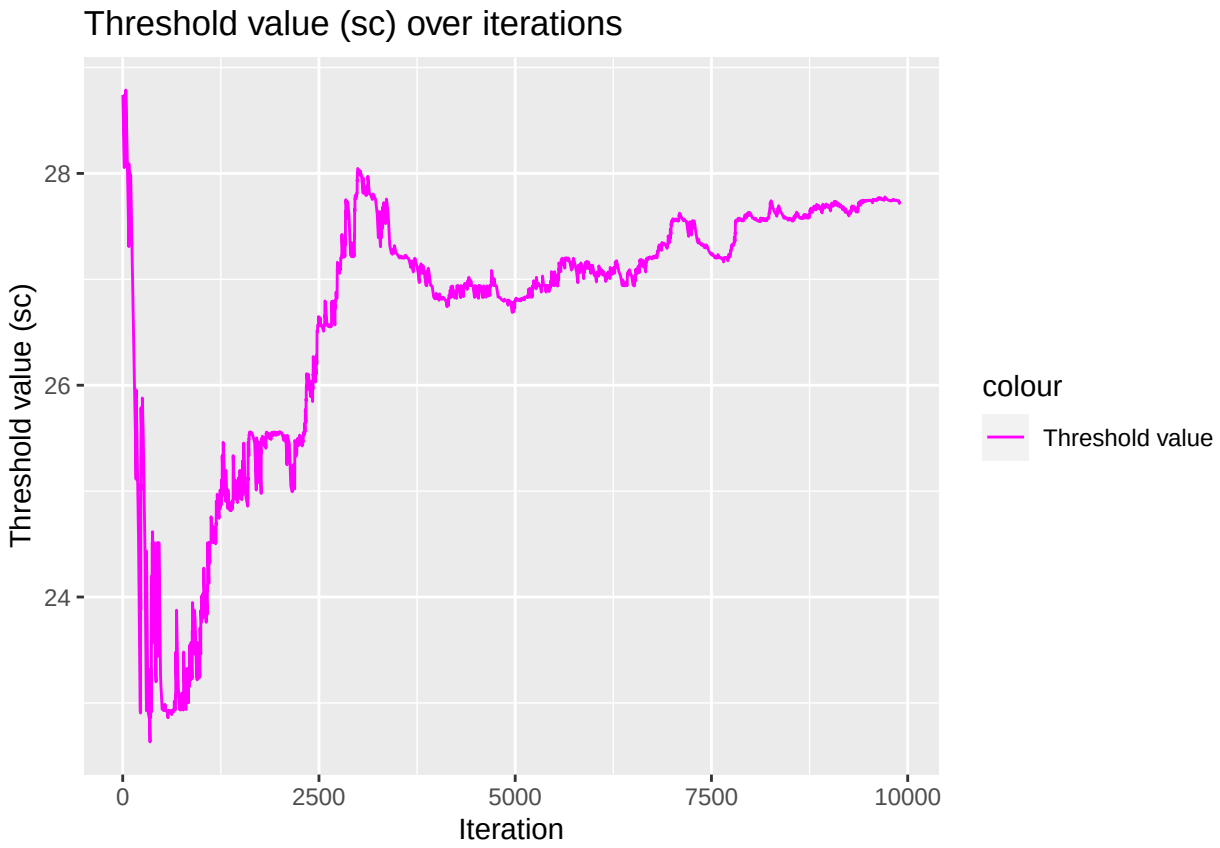
```
## [1] " quantile M=5000  26.7895091216073"
```

```
print(paste(" quantile M=10000 ", H0_data$sc_s[10000]))
```

```
## [1] " quantile M=10000  27.741281083301"
```

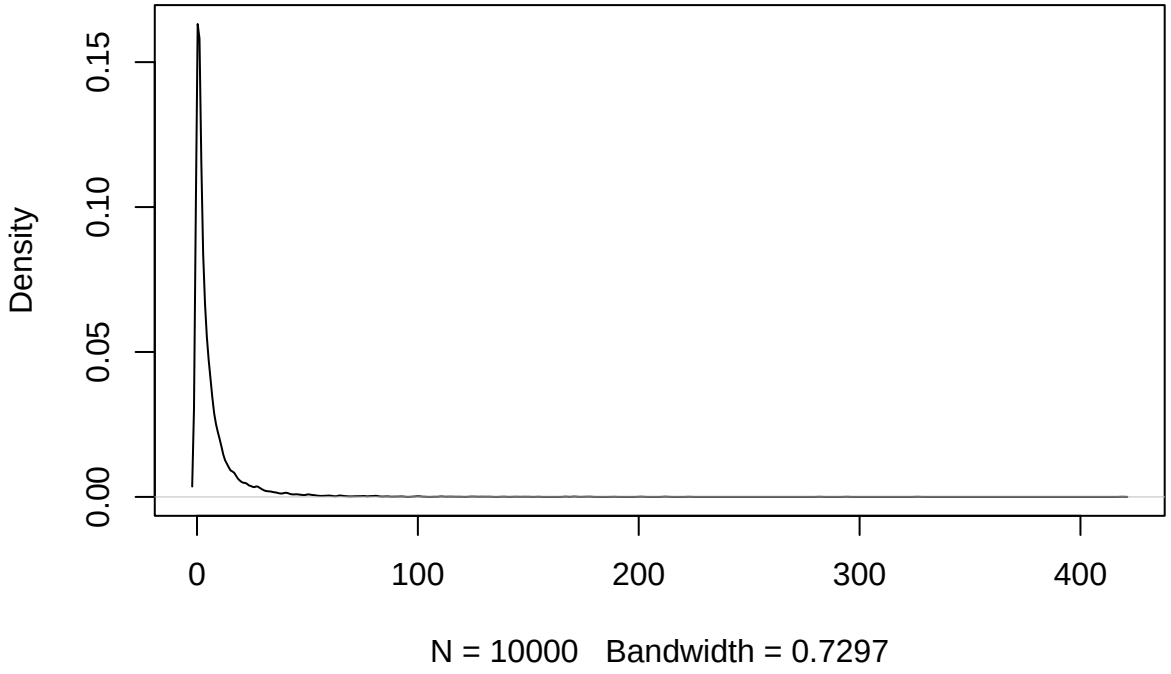
Plot convergence of quantile

```
ggplot(data = data.frame(H0_data$sc_s[100:M])) +
  geom_line(aes(x = seq_along(H0_data$sc_s[100:M]), y = H0_data$sc_s[100:M], color = "Threshold value"))
  xlab("Iteration") +
  ylab("Threshold value (sc)") +
  ggtitle("Threshold value (sc) over iterations") +
  scale_color_manual(values = c("Threshold value" = "magenta"))
```



Plot density of statistic in H_0 case

```
density_data <- density(H0_data$Dobs)
plot(density_data)
```


density.default(x = H0_data$Dobs)
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
N = 10000 Bandwidth = 0.7297