Multi CDF Final

Md Ashiqul Amin (ma3359)

January 30, 2020

"'— title: "CDF" author: "Md Ashiqul Amin (ma3359)" date: "October 2, 2019" output: pdf_document —

```
### get data and calculate key summary statistics
#Read data
#Mention the path of the data file
#header value will be true if there any header otherwise false

data <- read.table("C:/Users/Robin/Desktop/LoRaWAN/Throughput.csv", header = TRUE, sep = ",")
head(data)
```

```
##   Th_1 Th_2 Th_3 Th_4 Th_5 Th_11 Th_22 Th_33 Th_44 Th_55
## 1   26   26   21   24   23   208   208   168   192   184
## 2   26   26   21   21   23   208   208   168   168   184
## 3   26   26   20   23   23   208   208   160   184   184
## 4   23   26   23   23   24   184   208   184   184   192
## 5   23   26   23   23   24   184   208   184   184   192
## 6   21   26   23   23   24   168   208   184   184   192
```

```
#Select specific data from the dataset
data_1= data$Th_11
data_2 = data$Th_22
data_3 = data$Th_33
data_4 = data$Th_44
data_5 = data$Th_55

#Count the number of row conatining data
n = sum(!is.na(data_1))
m = sum(!is.na(data_2))
i = sum(!is.na(data_3))
j = sum(!is.na(data_4))
k = sum(!is.na(data_5))

#summary (optional)
summary(fivenum(data_1))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   128.0   168.0   184.0   179.2   208.0   208.0
```

```
summary(fivenum(data_2))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   128.0   176.0   184.0   177.6   192.0   208.0
```

```
summary(fivenum(data_3))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     128     184     192     184     208     208
```

```
summary(fivenum(data_4))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
##   128.0   168.0   184.0   177.6   200.0   208.0
```

```
summary(fivenum(data_5))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   136.0   184.0   192.0   185.6   208.0   208.0
```

```
# ordering the data
data.ordered = sort(data_1)
head(data.ordered)
```

```
## [1] 128 160 160 160 168 168
```

```
data.ordered_1 = sort(data_2)
data.ordered_2 = sort(data_3)
data.ordered_3 = sort(data_4)
data.ordered_4 = sort(data_5)

#create the image in png form

png('C:/Users/Robin/Desktop/LoRaWAN/Th.png',width = 300, height = 300, units = "px", bg = "white")

# plot the possible values of probability (0 to 1) against the ordered data
# notice the option type = '' for plotting the functions

#data_1
plot(data.ordered, (1:n)/n, type = 's',do.points=F, ylim = c(0, 1), xlab = 'Throughput', ylab = 'CDF')
```

```
## Warning in plot.window(...): "do.points" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "do.points" is not a graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "do.points" is
## not a graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "do.points" is
## not a graphical parameter
```

```
## Warning in box(...): "do.points" is not a graphical parameter
```

```
## Warning in title(...): "do.points" is not a graphical parameter
```

```
#data_2
#points(data.ordered_1, (1:m)/m, do.points=F, col="red", pch="*")
lines(data.ordered_1, (1:m)/m,do.points=F, col="red",lty=2)
```

```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "do.points" is not a
## graphical parameter
```

```
#data_3
#points(data.ordered_2, (1:i)/i, col="green", pch="+")
lines(data.ordered_2, (1:i)/i, col="green",lty=3)

#data_4
#points(data.ordered_3, (1:j)/j, col="orange", pch="o")
lines(data.ordered_3, (1:j)/j, col="orange",lty=4)

#data_5
#points(data.ordered_4, (1:k)/k, col="blue", pch="x")
lines(data.ordered_4, (1:k)/k, col="blue",lty=5)
```

```r
legend('topleft',
       legend=c("Node 1","Node 2","Node 3","Node 4", "Node 5"),  # text in the legend
       col=c("black","red","green","orange","blue"),  # point colors
       pch=15)  # specify the point type to be a square



dev.off()
```

```
## pdf
##   2
```