

Davis et al. R-Markdown

Dylan S. Davis

October 18, 2019

Introduction

This R Markdown details the distributional analysis of archaeological data discussed in the text. The goal of this analysis was to ascertain the conformity of the data to different continuous distributional functions to provide evidence to support or reject an ideal free distribution. This document also contains confidence envelopes (95%) for the archaeological data and each simulated dataset.

```
library(stats)
library(readxl)

## Warning: package 'readxl' was built under R version 3.5.3

library(emdbook)

## Warning: package 'emdbook' was built under R version 3.5.3

library(sfsmisc)

## Warning: package 'sfsmisc' was built under R version 3.5.3

##
## Attaching package: 'sfsmisc'

## The following object is masked from 'package:emdbook':
##
##   lseq

library(rmarkdown)

## Warning: package 'rmarkdown' was built under R version 3.5.3

setwd("C:/Users/Dylan/Documents/School-Work/Dissertation/RS_project")

#Load excel file containing artiact probability values

prob <- read_excel("Survey_results_probability.xlsx", sheet = "Material_Prob_Scores_R")

#create random sample of uniformly distributed values

set.seed(1)
```

```

unif_prob <- runif(245, min = 0, max = 23)

#K-S distribution test between archaeological data and uniform distribution with same max and min values
ks.test(prob$Prob, unif_prob)

## Warning in ks.test(prob$Prob, unif_prob): p-value will be approximate in
## the presence of ties

##
## Two-sample Kolmogorov-Smirnov test
##
## data: prob$Prob and unif_prob
## D = 0.13469, p-value = 0.02348
## alternative hypothesis: two-sided

#create random sample of normally distributed values

set.seed(1)

norm_dist <- rnorm(245)

#K-S distribution test between archaeological data and normal distribution data

ks.test(prob$Prob, norm_dist)

## Warning in ks.test(prob$Prob, norm_dist): p-value will be approximate in
## the presence of ties

##
## Two-sample Kolmogorov-Smirnov test
##
## data: prob$Prob and norm_dist
## D = 0.83673, p-value < 2.2e-16
## alternative hypothesis: two-sided

##create gamma distributed data

set.seed(1)

gamma_dist <- rgamma(n=246, shape = 1)

ks.test(prob$Prob, gamma_dist)

## Warning in ks.test(prob$Prob, gamma_dist): p-value will be approximate in
## the presence of ties

##
## Two-sample Kolmogorov-Smirnov test
##

```

```

## data: prob$Prob and gamma_dist
## D = 0.7879, p-value < 2.2e-16
## alternative hypothesis: two-sided

##poisson distributed data
set.seed(1)

pois_dist <- rpois(n=246, lambda = 1)

ks.test(prob$Prob,pois_dist)

## Warning in ks.test(prob$Prob, pois_dist): p-value will be approximate in
## the presence of ties

##
## Two-sample Kolmogorov-Smirnov test
##
## data: prob$Prob and pois_dist
## D = 0.80823, p-value < 2.2e-16
## alternative hypothesis: two-sided

##create sample of logarithmically distributed values (using emdbook library)
set.seed(1)
log_dist <- lseq(0.1, 24, 245)

##K-S distribution between archaeological data and log distribution data

ks.test(prob$Prob, log_dist)

## Warning in ks.test(prob$Prob, log_dist): p-value will be approximate in th
e
## presence of ties

##
## Two-sample Kolmogorov-Smirnov test
##
## data: prob$Prob and log_dist
## D = 0.5102, p-value < 2.2e-16
## alternative hypothesis: two-sided

##plotting KS distribution tests

##KS test with UNIFORM DIstribution

U_group <- c(rep(prob$Prob, length(prob$Prob)), rep(unif_prob, length(unif_pr
ob)))

U_dat <- data.frame(KSD = c(prob$Prob, unif_prob), group=U_group)

# create ECDF of data

```

```

U_cdf1 <- ecdf(prob$Prob)
U_cdf2 <- ecdf(unif_prob)

# find min and max statistics to draw line between points of greatest distance
U_minMax <- seq(min(prob$Prob, unif_prob), max(prob$Prob, unif_prob), length.out=length(prob$Prob))
U_x0 <- U_minMax[which( abs(U_cdf1(U_minMax) - U_cdf2(U_minMax)) == max(abs(U_cdf1(U_minMax) - U_cdf2(U_minMax))) )]
U_y0 <- U_cdf1(U_x0)
U_y1 <- U_cdf2(U_x0)

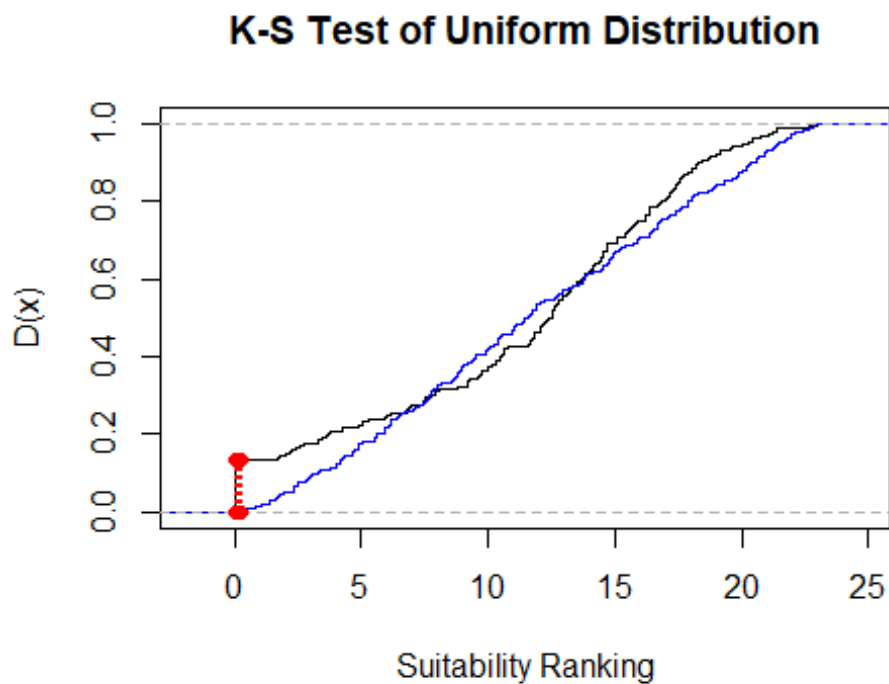
##Plot distribution curves without ggplot

plot(U_cdf1, verticals=TRUE, do.points=FALSE, col="black", main = "K-S Test of Uniform Distribution", xlab="Suitability Ranking", ylab="D(x)")
plot(U_cdf2, verticals=TRUE, do.points=FALSE, col="blue", add=TRUE)

## creates line for maximum difference between both datasets

points(c(U_x0, U_x0), c(U_y0, U_y1), pch=16, col="red")
segments(U_x0, U_y0, U_x0, U_y1, col="red", lty="dotted")

```



```

##TEST WITH NORMAL DISTRIBUTION

N_group <- c(rep(prob$Prob, length(prob$Prob)), rep(norm_dist, length(norm_dist)))

N_dat <- data.frame(KSD = c(prob$Prob, norm_dist), group=N_group)

# create ECDF of data

N_cdf1 <- ecdf(prob$Prob)
N_cdf2 <- ecdf(norm_dist)

# find min and max statistics to draw line between points of greatest distance
N_minMax <- seq(min(prob$Prob, norm_dist), max(prob$Prob, norm_dist), length.out=length(prob$Prob))
N_x0 <- N_minMax[which( abs(N_cdf1(N_minMax) - N_cdf2(N_minMax)) == max(abs(N_cdf1(N_minMax) - N_cdf2(N_minMax))) )]
N_y0 <- N_cdf1(N_x0)
N_y1 <- N_cdf2(N_x0)

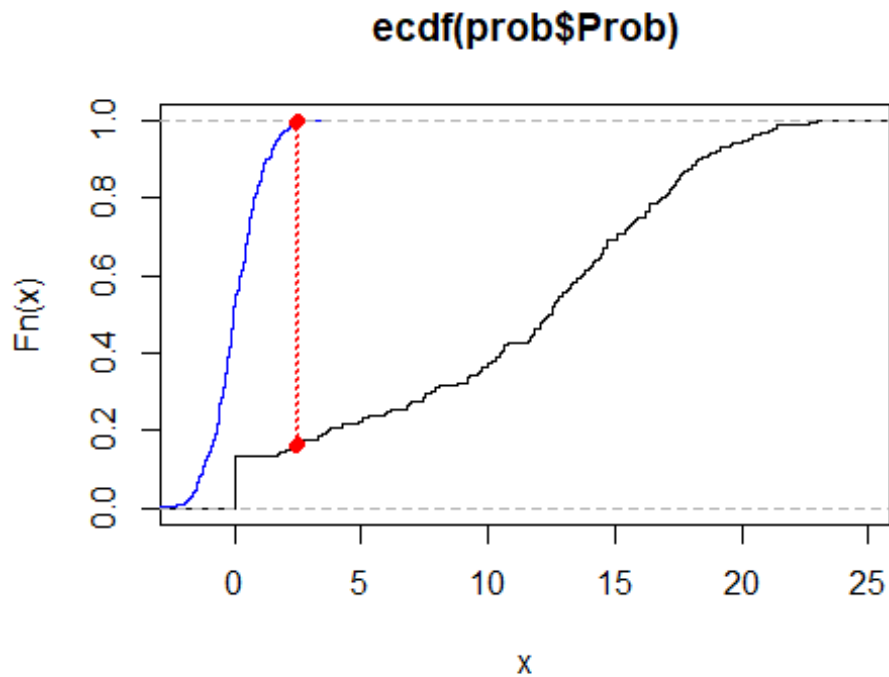
##PLOT

plot(N_cdf1, verticals=TRUE, do.points=FALSE, col="black")
plot(N_cdf2, verticals=TRUE, do.points=FALSE, col="blue", add=TRUE)

# draw line between points of greatest distance

points(c(N_x0, N_x0), c(N_y0, N_y1), pch=16, col="red")
segments(N_x0, N_y0, N_x0, N_y1, col="red", lty="dotted")

```



```
####TEST WITH GAMMA DISTRIBUTION

##plotting KS distribution tests

G_group <- c(rep(prob$Prob, length(prob$Prob)), rep(gamma_dist, length(gamma_
dist)))

# create ECDF of data

G_cdf1 <- ecdf(prob$Prob)
G_cdf2 <- ecdf(gamma_dist)

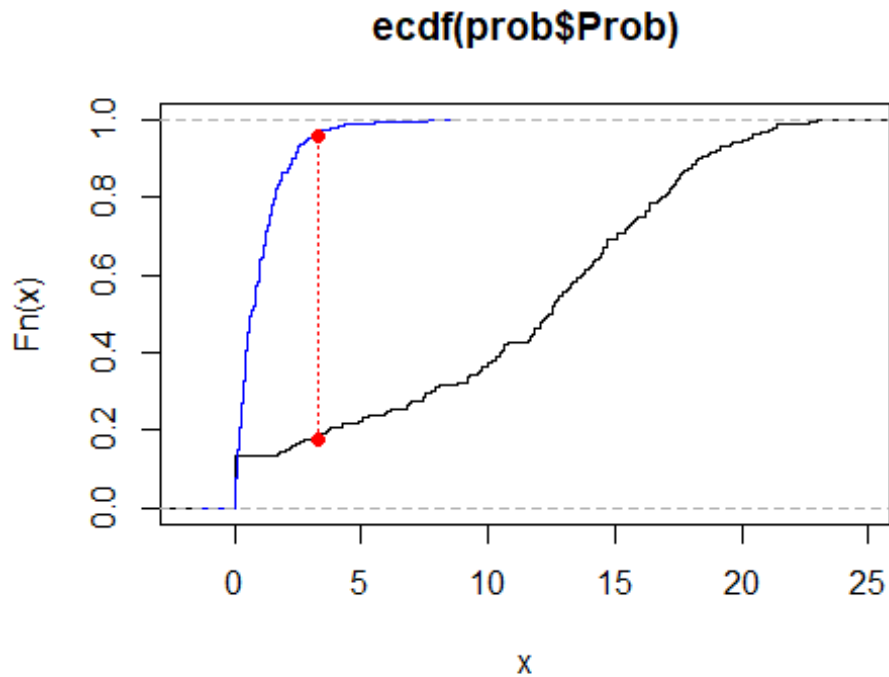
# find min and max statistics to draw line between points of greatest distanc
e
G_minMax <- seq(min(prob$Prob, gamma_dist), max(prob$Prob, gamma_dist), lengt
h.out=length(prob$Prob))
G_x0 <- G_minMax[which( abs(G_cdf1(G_minMax) - G_cdf2(G_minMax)) == max(abs(G
_cdf1(G_minMax) - G_cdf2(G_minMax))) )]
G_y0 <- G_cdf1(G_x0)
G_y1 <- G_cdf2(G_x0)

##PLOT

plot(G_cdf1, verticals=TRUE, do.points=FALSE, col="black")
plot(G_cdf2, verticals=TRUE, do.points=FALSE, col="blue", add=TRUE)
```

```
# draw line between points of greatest distance
```

```
points(c(G_x0, G_x0), c(G_y0, G_y1), pch=16, col="red")  
segments(G_x0, G_y0, G_x0, G_y1, col="red", lty="dotted")
```



```
####TEST WITH POISSION DISTRIBUTION
```

```
##plotting KS distribution tests
```

```
P_group <- c(rep(prob$Prob, length(prob$Prob)), rep(pois_dist, length(pois_dist)))
```

```
# create ECDF of data
```

```
P_cdf1 <- ecdf(prob$Prob)
```

```
P_cdf2 <- ecdf(pois_dist)
```

```
# find min and max statistics to draw line between points of greatest distance
```

```
P_minMax <- seq(min(prob$Prob, pois_dist), max(prob$Prob, pois_dist), length.out=length(prob$Prob))
```

```
P_x0 <- P_minMax[which( abs(P_cdf1(P_minMax) - P_cdf2(P_minMax)) == max(abs(P_cdf1(P_minMax) - P_cdf2(P_minMax))) )]
```

```
P_y0 <- P_cdf1(G_x0)
```

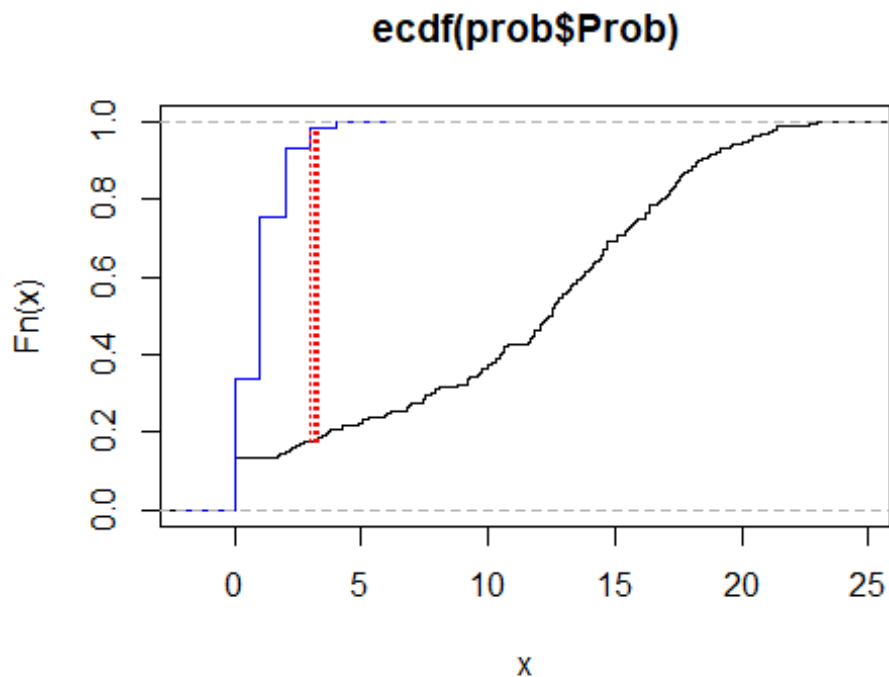
```
P_y1 <- P_cdf2(G_x0)
```

```
##PLOT

plot(P_cdf1, verticals=TRUE, do.points=FALSE, col="black")
plot(P_cdf2, verticals=TRUE, do.points=FALSE, col="blue", add=TRUE)

# draw line between points of greatest distance

segments(P_x0, P_y0, P_x0, P_y1, col="red", lty="dotted")
```



```
##CREATE PANNELED FIGURE
# 4 figures arranged in 2 rows and 2 columns

attach(mtcars)

## The following object is masked from package:ggplot2:
##
##      mpg

par(mfrow=c(2,2))

#UNIFORM
plot(U_cdf1, verticals=TRUE, do.points=FALSE, col="black",
      main = "K-S Test of Uniform Distribution", xlab="Suitability Ranking", y
```



```

lab="D(x)")
plot(U_cdf2, verticals=TRUE, do.points=FALSE, col="blue", add=TRUE)
points(c(U_x0, U_x0), c(U_y0, U_y1), pch=16, col="red")
segments(U_x0, U_y0, U_x0, U_y1, col="red", lty="dotted")

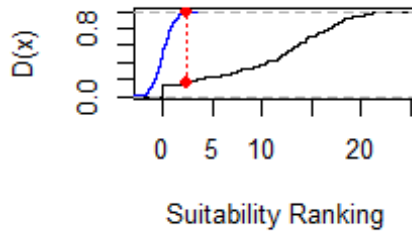
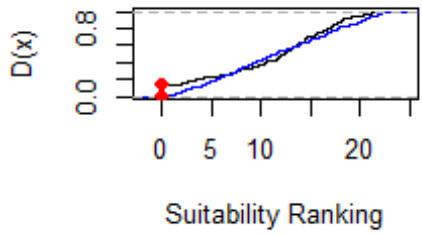
#NORMAL
plot(N_cdf1, verticals=TRUE, do.points=FALSE, col="black",
      main = "K-S Test of Normal Distribution", xlab="Suitability Ranking", ylab="D(x)")
plot(N_cdf2, verticals=TRUE, do.points=FALSE, col="blue", add=TRUE)
points(c(N_x0, N_x0), c(N_y0, N_y1), pch=16, col="red")
segments(N_x0, N_y0, N_x0, N_y1, col="red", lty="dotted")

#GAMMA
plot(G_cdf1, verticals=TRUE, do.points=FALSE, col="black",
      main = "K-S Test of Gamma Distribution", xlab="Suitability Ranking", ylab="D(x)")
plot(G_cdf2, verticals=TRUE, do.points=FALSE, col="blue", add=TRUE)
points(c(G_x0, G_x0), c(G_y0, G_y1), pch=16, col="red")
segments(G_x0, G_y0, G_x0, G_y1, col="red", lty="dotted")

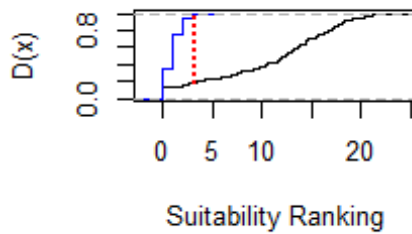
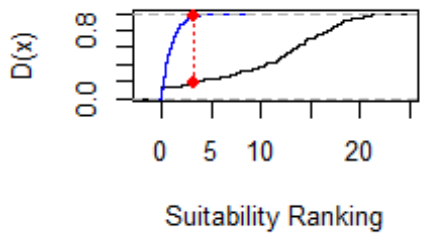
#POISSON
plot(P_cdf1, verticals=TRUE, do.points=FALSE, col="black",
      main = "K-S Test of Poisson Distribution", xlab="Suitability Ranking", ylab="D(x)")
plot(P_cdf2, verticals=TRUE, do.points=FALSE, col="blue", add=TRUE)
segments(P_x0, P_y0, P_x0, P_y1, col="red", lty="dotted")

```

K-S Test of Uniform Distributi **K-S Test of Normal Distributi**



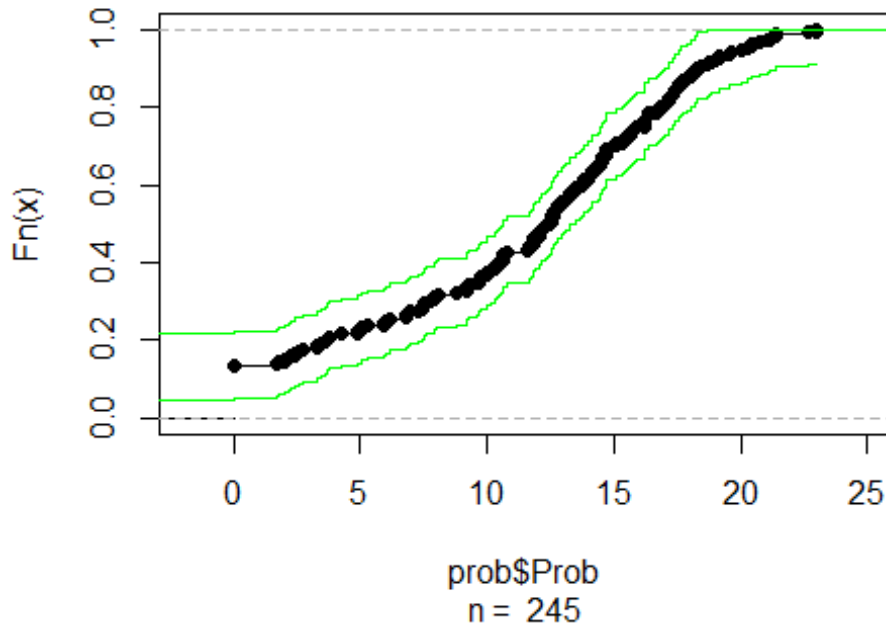
K-S Test of Gamma Distributi **K-S Test of Poisson Distributi**



##CONFIDENCE ENVELOPES FOR KS, ECDF distribution (uses sfsmisc package)

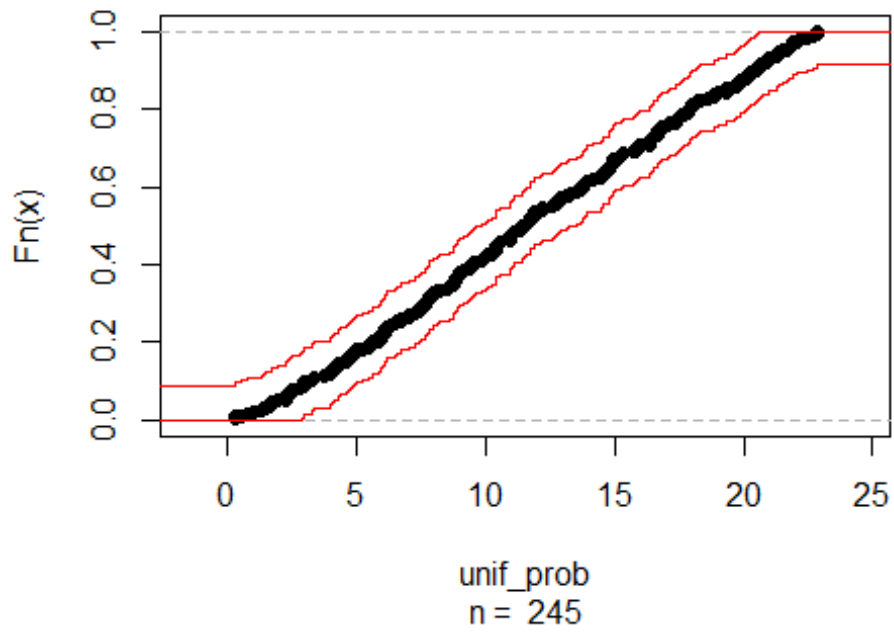
```
ecdf.ksCI(prob$Prob, ci.col = "green")
```

ecdf(prob\$Prob) + 95% K.S. bands



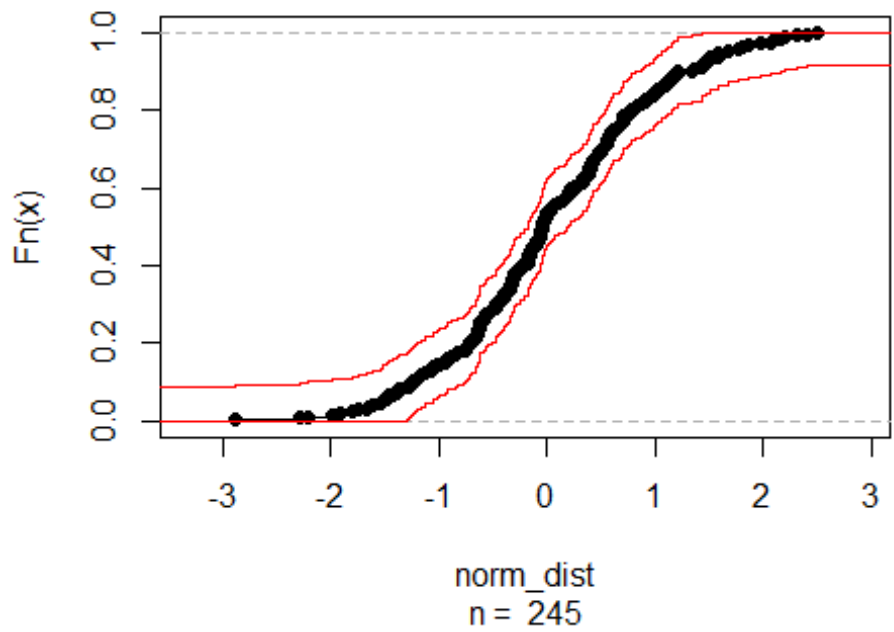
```
ecdf.ksCI(unif_prob)
```

ecdf(unif_prob) + 95% K.S. bands

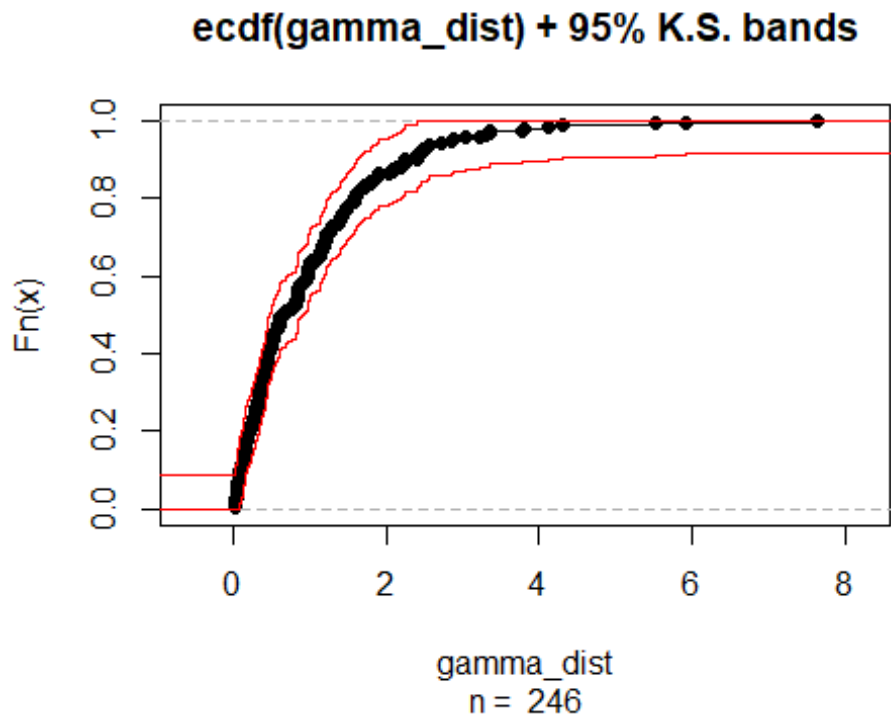


```
ecdf.ksCI(norm_dist)
```

ecdf(norm_dist) + 95% K.S. bands



```
ecdf.ksCI(gamma_dist)
```



```
ecdf.ksCI(pois_dist)
```

