Example1.2.5.R

ATE

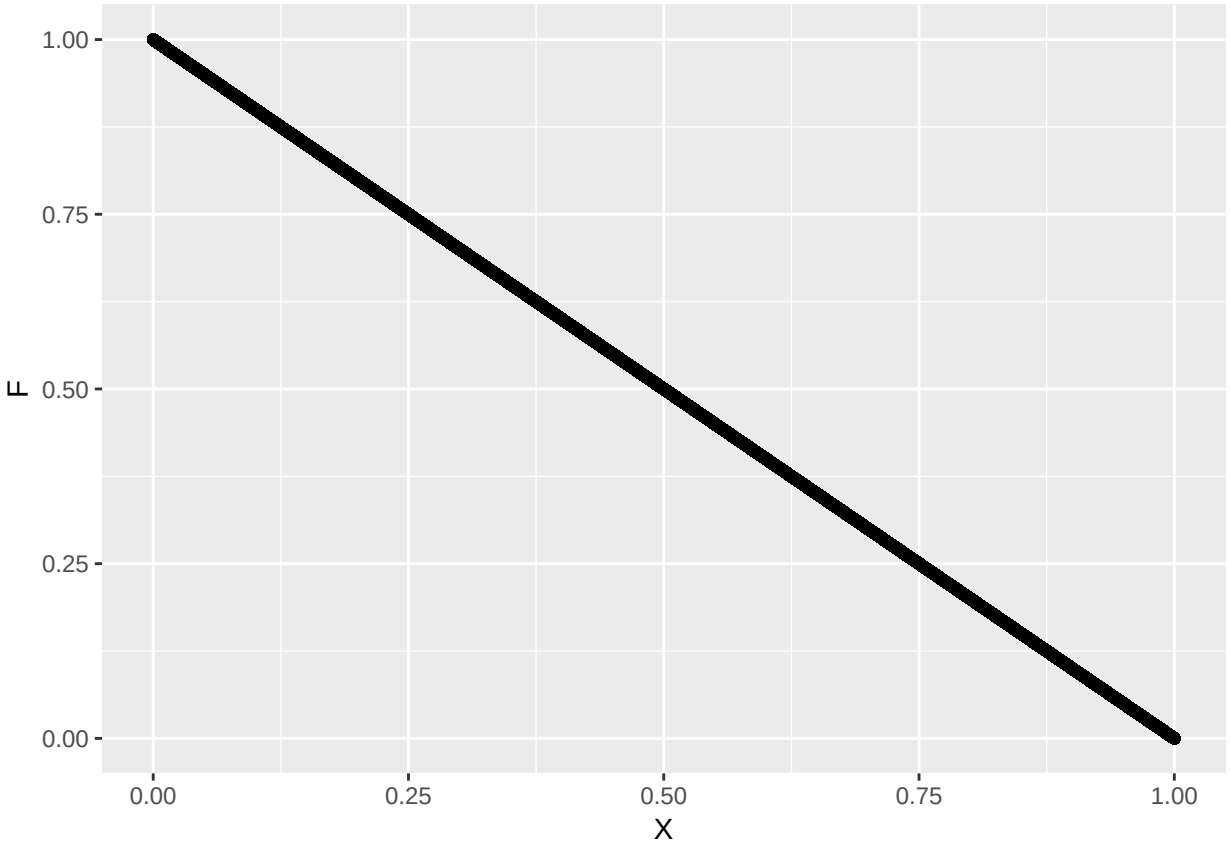
Thu Jul 13 16:12:32 2017

```
######################################################################
# Example 1.2.5: Another random variable uniformly distributed.
######################################################################
library(ggplot2)
# W random variables included various value from 0 to one.
# All UNIFORMALY distributed.
W <- seq(from = 0,
         to = 1,
         by = .0001)
#
# Define the probability measure p & q
p <- 1/2
q <- 1 - p
# number of exeriments: expe
expe <- 100000
# Sample size:
n <- 100

##
# Construction of the theoretical distribution of X
#
# Not consider the infinite sample space but instead a rather large one
#
#   * [n]: It gives the number of columns.
#   * [2^n]: Gives the number of rows.
##
n <- 20
Y <- expand.grid(
  rep(
    list(c(1,0)),
    n)
)
Y <- t(apply(Y, MARGIN = 1, FUN = function(x) x/2^(1:n)))
X <- sort(rowSums(Y), decreasing = T)

distrib_X <- data.frame('X' = X,
                        'P' = 1/length(X),
                        'F' = seq(1/length(X), 1, by = 1/length(X)))

ggplot(distrib_X, aes(X, F)) +
  geom_point()
```




```
# Find the expected value:
prob_X <- 1/length(X)
sum(X * prob_X)
```

```
## [1] 0.4999995
```

```
##
# Next: Convergence of Integrals.
##

#
# Y should be constructed according to some random experiment.
Y = list()
for(i in 1:expe){
  Y[[i]] <- rbinom(n = n,
                   size = 1,
                   prob = p)
}
# According to Y, construction of X:
# X should be a random variable construct upon a vector
# Transformation of Y
YPrime <- Y
for(i in 1:expe)
  for(j in 1:n)
    YPrime[[i]][j] <- Y[[i]][j] / 2^j

X <- vector()
```

```r
for(i in 1:expe)
  X[i] <- sum(YPrime[[i]])

# Probability that the value of X fall between 4/3000 and 5/3000:
ProbX <- 1 / 2 ^ n

#########################################################
#  Check of the theory
 #########################################################

# Probability that X is in the interval:
# [0, 3e29/2^n]:
probI <- (3e29 - 0)/2^n
# Check:
length(X[X < 3e29/2^n]) / length(X)
```

```
## [1] 1
```

```r
# Probability that X is in the interval:
# [3e28/2^n, 3e29/2^n]:
probI <- (3e29 - 3e28)/2^n
# Check:
length(X[X > 3e28/2^n & X < 3e29/2^n]) / length(X)
```

```
## [1] 0
```