Lista 4: Clebson Cardoso Alves de Sá

Code

Usage Examples: Binomial Distribution

Hide

```
rbinom(13, 20, 0.15)
```

```
 [1] 4 2 3 5 1 0 4 1 1 2 4 3 2
```

Hide

```
dbinom(13, 20, 0.15)
```

```
[1] 4.836516e-07
```

Hide

```
pbinom(13, 20, 0.15)
```

```
[1] 1
```

Hide

```
pbinom(0.20, 20, 0.15)
```

```
[1] 0.03875953
```

Usage Examples: Poisson Distribution

Hide

```
help(rpois)
```

Usage Examples: Gaussian Distribution

Hide

```
rnorm(100, 10, 2)
```

```
  [1]  5.790882 10.305302  7.892930 16.701383  9.002037  9.447400 10.981194 10.455710
  8.342838  5.560062  9.708634 10.570141 11.603880 10.473353 12.421074
 [16] 10.395502 11.755496 11.802836  9.118220  8.551303  9.505029  9.350180  9.935088
  8.327182  7.349706  7.932593  9.415753  8.531964  7.833598 13.434966
 [31]  6.395001  9.007763  9.840763  9.987228 10.008898 12.092404  7.108858 11.429791
10.489193  9.699970  9.515180  8.311093  9.175166 12.616959 11.019788
 [46] 12.060119  8.618155 12.114560 10.288170  7.994297 13.402495 11.222548 12.900251
  7.676207 11.815346 12.405508 11.166235 10.650999 10.710162 13.140483
 [61] 10.032650  9.182950  7.221198  9.783837  7.437410  7.201554  8.081093  9.678963
  7.592966  7.763269 11.226649 13.361370 11.374998 11.675522  8.149506
 [76]  9.749078  9.486495  8.862849  9.149494  9.805743 12.289213  5.955485  7.823000
12.138447  9.495102 10.104813  9.476591 11.042683  8.495195 10.020698
 [91] 10.830635  9.030373 13.181944 15.926320 11.372029  6.964228 15.903252  8.542636
10.244358 10.025495
```

Hide

```
dnorm(11.25, 10, 2)
```

```
[1] 0.1640805
```

Hide

```
pnorm(11.25, 10, 2)
```

```
[1] 0.7340145
```

Hide

```
pnorm(0.20, 10, 2)
```

```
[1] 4.791833e-07
```

Usage Examples: Exponential Distribution

Hide

```
help(rexp)
```

Other distributions

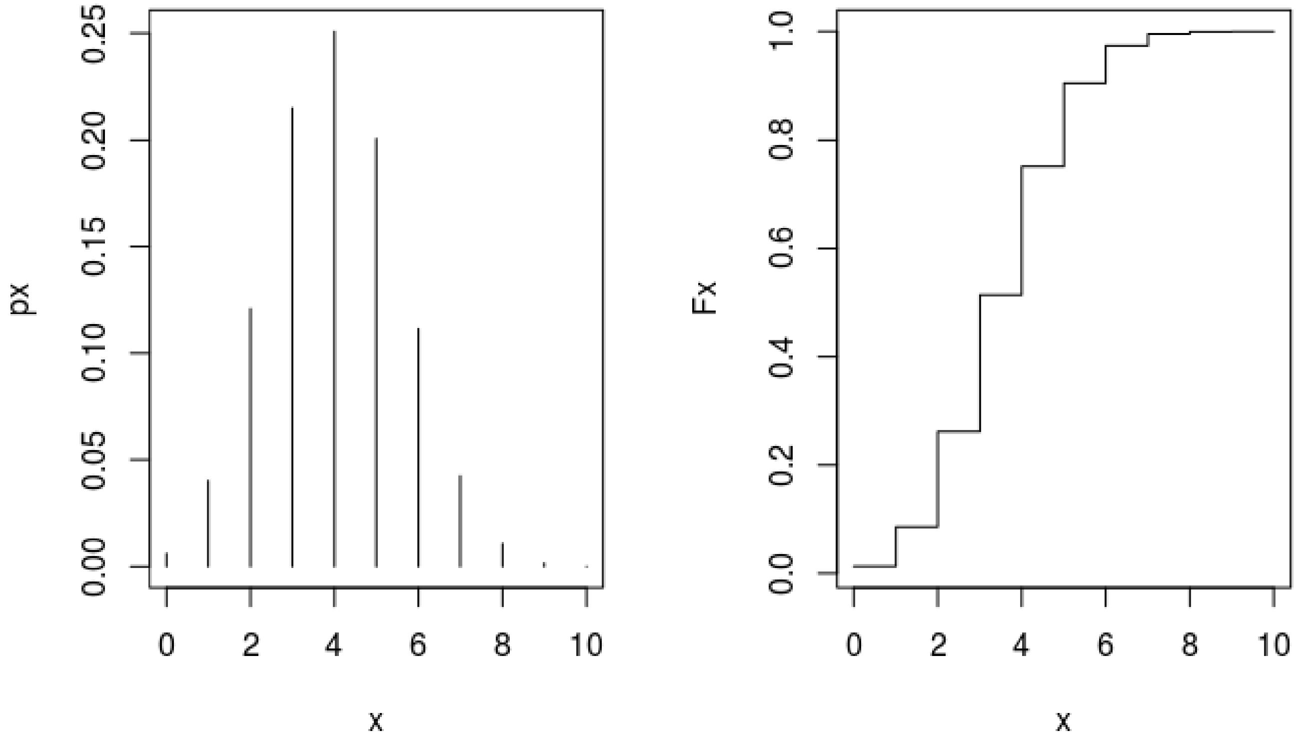
Hide

```
help(distributions)
```

Question 1

Hide

```
x <- 0:10
px <- dbinom(x, 10, 0.40)
par(mfrow=c(1,2)) # janela grafica com uma linha de 2 plots
plot(x, px, type = "h") # para usar linhas verticais at\'{e} os pontos (x,px)
Fx <- pbinom(x, 10, 0.35)
plot(x, Fx, type = "s") # o argumento "s"
```

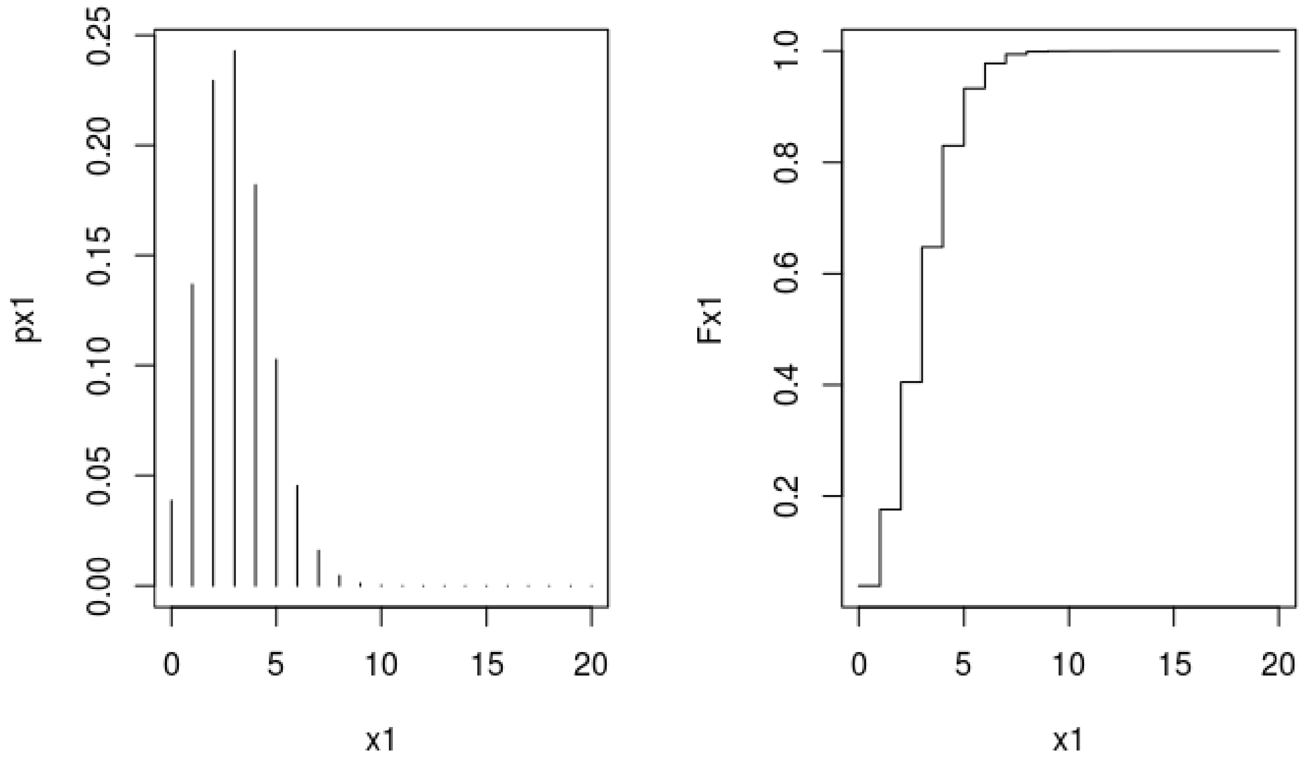


1.a)

Obtenha o gráfico das probabilidades P(X = k) e da função de probabilidade acumulada F(x) para uma v.a. X ~ Bin(n = 20, θ = 0.15).

Hide

```
x1 <- 0:20
px1 <- dbinom(x1, 20, 0.15)
par(mfrow=c(1,2)) # janela grafica com uma linha de 2 plots
plot(x1, px1, type = "h") # para usar linhas verticais at\'{e} os pontos (x,px)
Fx1 <- pbinom(x1, 20, 0.15)
plot(x1, Fx1, type = "s") # o argumento "s"
```



1.b)

Qual o valor k em que P(X = k) é máxima? Quanto é esta probabilidade máxima?

Hide

```
max_index = 0
max_value = px1[1]
for(i in x1){
  if (px1[i+1] > max_value){
    max_value = px1[i+1]
    max_index = i+1;
  }
}
sprintf("k é máximo em %d com probabilidade de %f.", max_index, max_value)
```

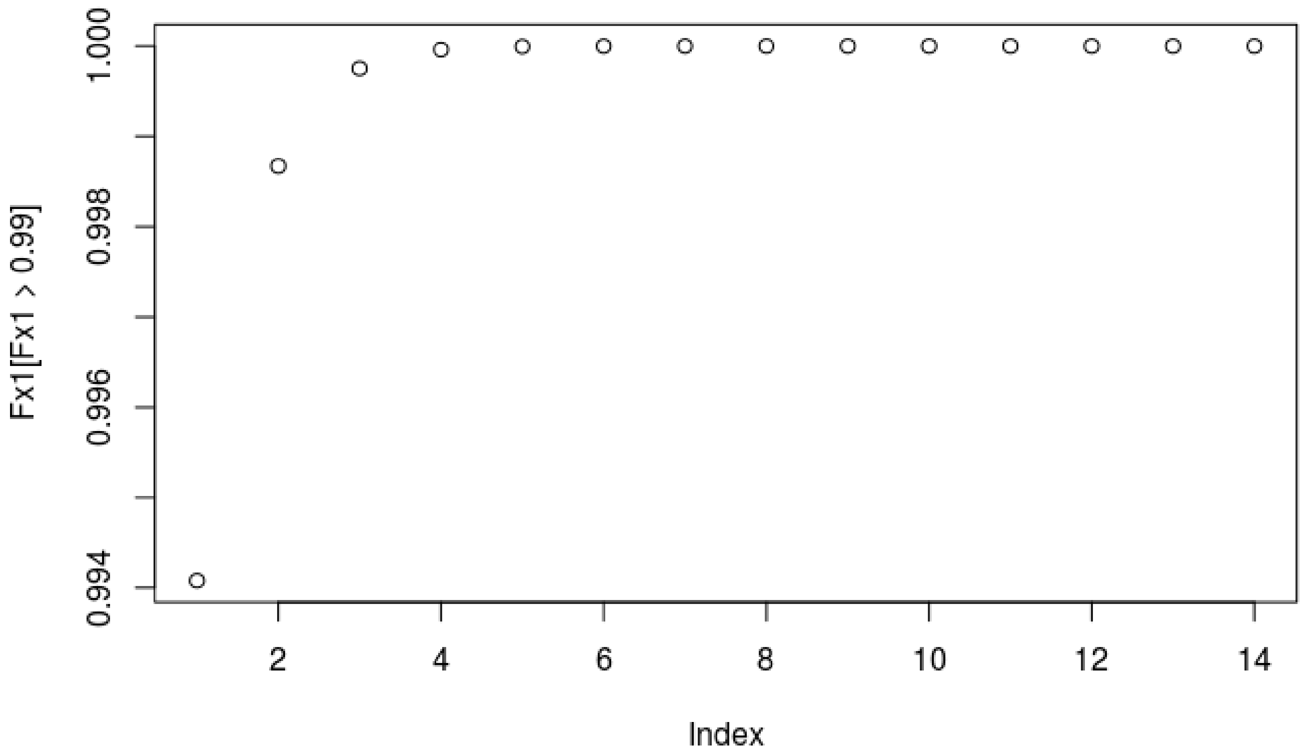
```
[1] "k é máximo em 4 com probabilidade de 0.242829."
```

1.c)

VISUALMENTE, obtenha uma faixa de valores (a, b) na qual a probabilidade de X ∈ (a, b) seja próxima de 1. Procure grosseiramente obter a faixa mais estreita possível.

Hide

```
plot(Fx1[Fx1 > 0.99])
```



1.d)

O valor (teórico) de E(X) no caso de uma binomial é nθ. Como é o comportamento da função P(X = k) no entorno deste valor E(X)? Ela tem valores P(X = k) relativamente altos?

Hide

```
ex = 20 * 0.15
print(px1[ex])
```

```
[1] 0.2293384
```

Conforme pode-se observar, a a probabilidade em torno de E(X) é de 0.22. Este valor é relativamente alto quando comparado com o valor máximo de 0.24.

1.e)

Confirme esta impressão calculando P(a ≤ X ≤ b) usando a função `dnorm` ou `pnorm` do R. Por exemplo, se eu quiser P(5 ≤ X ≤ 8), uso `sum(dnorm(5:8, 20, 0.15))` ou então `pbinom(8, 20, 0.15) - pbinom(5-0.01, 20, 0.15)` . Porque eu subtraio 0.01 de 5 na chamada da segunda função? Usando `dnorm` : Utilizando `pnorm` :

Hide

```
pn <- pnorm(20, 20, 0.15) - pnorm(1-0.001, 20, 0.15)
print(pn)
```

```
[1] 0.5
```

O exercício pede para calcular com $n = 20$ e $\theta = 15$, conforme pode ser observado a probabilidade do intervalo $[0, 14]$ é 0. Considerando o intervalo total, a probabilidade é de 0.5.

Utilizando `dnorm`

Hide

```
dn <- dnorm(1:20, 20, 0.15)
print(dn)
```

```
 [1]  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e
+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
[12]  0.000000e+00  0.000000e+00  0.000000e+00 1.412884e-241 1.020947e-154  3.680632e
-87  6.620105e-39  5.940600e-10  2.659615e+00
```

Hide

```
print(sum(dn))
```

```
[1] 2.659615
```

A subtração é dada para que seja considerado o intervalo fechado de 5 a 8.

1.f)

Use qbinom para obter o inteiro k tal que $F(k) = P(X \leq k) \approx 0.95$.

Hide

```
qbinom(0.95, 20,0.95)
```

```
[1] 20
```

1.g)

Verifique o valor da probabilidade acumulada exata F(k) obtida com o inteiro acima usando pbinom.

Hide

```
print(Fx1[20])
```

```
[1] 1
```

Hide

```
pbinom(20, 20, 1)
```

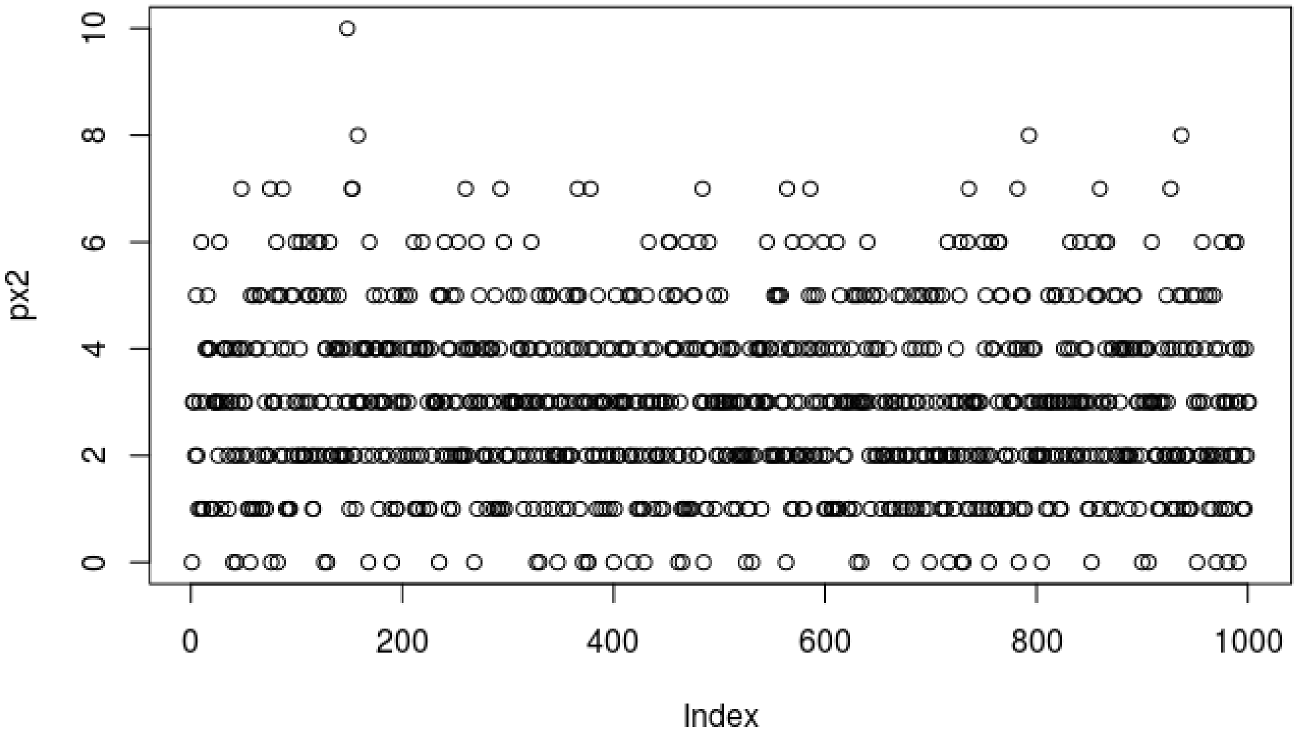
```
[1] 1
```

1.h)

Gere 1000 valores aleatórios independentes de X ∼ Bin(n = 20, θ = 0.15). Estes valores cairam, em sua maioria, na faixa que você escolheu mais acima? Qual a porcentagem de valores que caiu na faixa que você escolheu?

Hide

```
x2 <- 0:1000
px2 <- rbinom(x2, 20, 0.15)
plot(px2)
```



1.i)

Compare os valores das probabilidades P(X = k) para k = 0, . . . 6 e as frequências relativas destes inteiros nos 100 valores simulados.São parecidos?

Hide

```
w1 = dbinom(0:6, 20, 0.15)
print("Dbinomial")
```

```
[1] "Dbinomial"
```

Hide

```
print(w1)
```

```
[1] 0.03875953 0.13679835 0.22933840 0.24282890 0.18212167 0.10284518 0.04537287
```

Hide

```
w2 = rbinom(100, 20, 0.15)
print("Rbinomial")
```

```
[1] "Rbinomial"
```

Hide

```
print(w2)
```

```
  [1] 2 8 3 3 5 3 4 0 0 2 6 3 5 3 1 1 1 4 3 5 2 4 5 2 4 2 3 1 4 2 0 2 5 1 1 4 4 2 4 3
6 2 8 3 3 1 1 6 3 2 3 3 1 6 4 2 5 4 5 3 0 1 1 1 6 4 1 3 2 3 4 0 3 7 7 3 5
 [78] 3 3 3 2 3 1 4 1 4 2 3 3 3 1 2 5 3 5 2 5 5 2 4
```

Hide

```
for(i in 0:6){
  print(length(w2[w2 == i])/100)
}
```

```
[1] 0.05
[1] 0.16
[1] 0.17
[1] 0.26
[1] 0.15
[1] 0.12
[1] 0.05
```

Conforme pode ser observado, elas são muito parecidas sim w1: 0.03875953 0.13679835 0.22933840 0.24282890 0.18212167 0.10284518 0.04537287 w2: 0.05 0.14 0.15 0.23 0.23 0.13 0.04

Questão 2.

2.1

Obtenha o gráfico das probabilidades P(X = k) e da função de probabilidade acumulada F(x) para uma v.a. X ~ Poisson(λ) usando dois valores: λ = 0.73 e λ = 10.

Hide

```
  par(mfrow=c(2,3))
    # λ = 0.73
  x <- 0:10
  px <- dpois(x, 0.73)
  plot(x, px, type = "h")
  Fx <- dpois(x, 0.73)
  plot(x, Fx, type = "s")
```

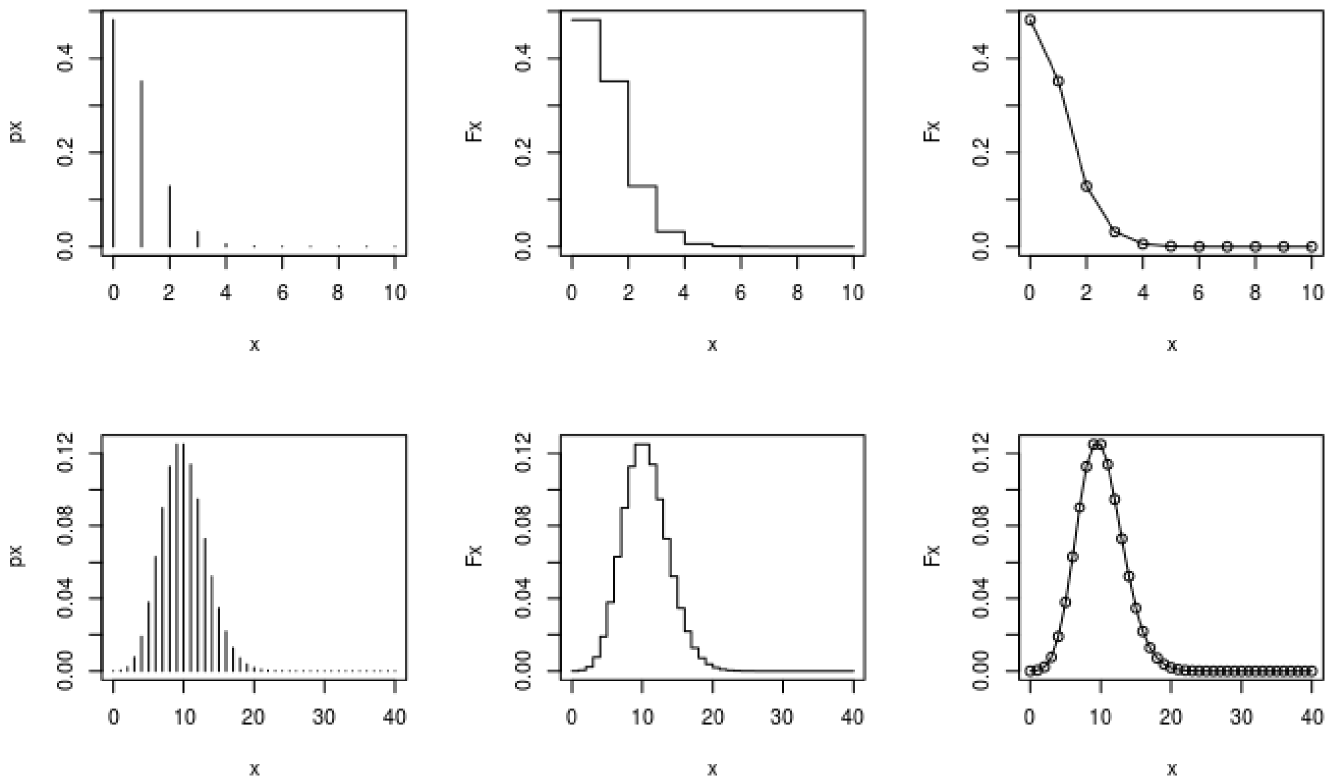
Hide

```
plot(x, Fx, type = "o")

  # λ = 10
x <- 0:40
px <- dpois(x, 10)
plot(x, px, type = "h")
```

Hide

```
Fx <- dpois(x, 10)
plot(x, Fx, type = "s")
plot(x, Fx, type = "o")
```



2.2

O valor k em que P(X = k) é máximo é próximo de E(X) = λ? com λ = 0.73:, p(X) esta maximo no ponto 0 e com λ = 10:, p(X) esta maximo no ponto 10 sim, está bem proximo de valor esperado E(X) = λ

2.3

Obtenha um intervalo de valores (a, b), o mais curto possível gosseiramente, para o qual $P(X \in (a, b)) \approx 1$ para λ = 0.73 X pertence intervalo de (0,3) para λ = 10 X pertence intervalo de (2,20)

2.4

Usando ppois do R, calcule $P(a \leq X \leq b)$. para λ = 0.73: ppois(3, 0.73) - ppois(0-0.01, 0.73) = 0.9933523

para λ = 10: ppois(20, 10) - ppois(2-0.01, 10) = 0.9979123

2.5

Gere 200 valores aleatórios independentes de X ∼ Poisson(λ) com os dois valores acima para λ. para λ = 0.73:

Hide

```
P1 = dpois(0:6, 0.73)
P1
```

```
[1] 0.4819089901 0.3517935628 0.1284046504 0.0312451316 0.0057022365 0.0008325265 0.0
001012907
```

Hide

```
P2 = rpois(100, 0.73)
P2
```

```
  [1] 1 1 0 1 0 0 1 0 1 2 1 0 1 2 1 1 2 0 1 1 0 1 1 0 1 0 1 1 0 1 2 0 1 0 0 4 0 1 2 3
0 0 0 2 0 0 1 0 2 1 2 1 0 0 0 1 1 0 0 0 2 0 2 0 1 0 1 0 1 0 0 0 1 1 0 0 1
 [78] 2 1 0 1 0 1 2 0 1 0 0 2 0 2 0 2 0 3 2 1 0 2 1
```

Hide

```
for(i in 0:6){
  print(length(P2[P2 == i])/100)
}
```

```
[1] 0.44
[1] 0.36
[1] 0.17
[1] 0.02
[1] 0.01
[1] 0
[1] 0
```

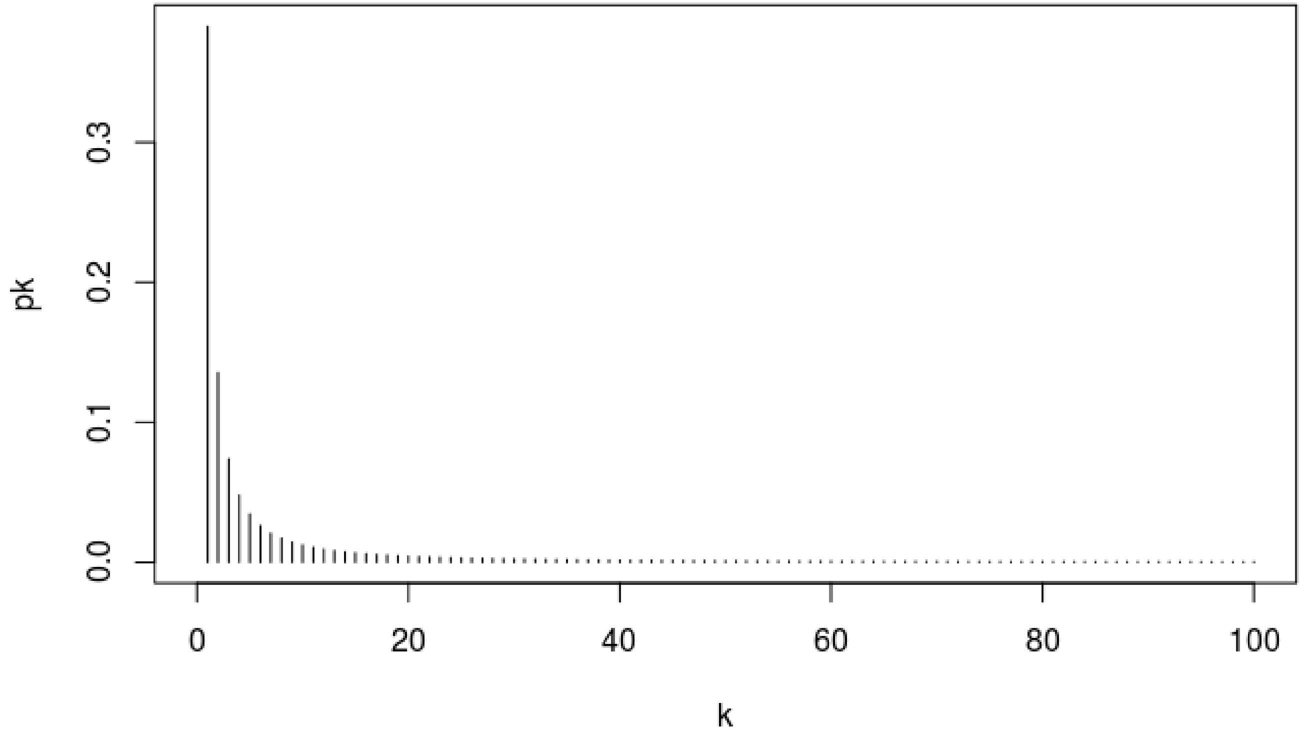
sao parecidos sim: P1: 0.4819089901 0.3517935628 0.1284046504 0.0312451316 0.0057022365 0.0008325265 0.0001012907 P2: 0.4 0.4 0.12 0.07 0.01 0 0

Questão 3

Questão 3.1

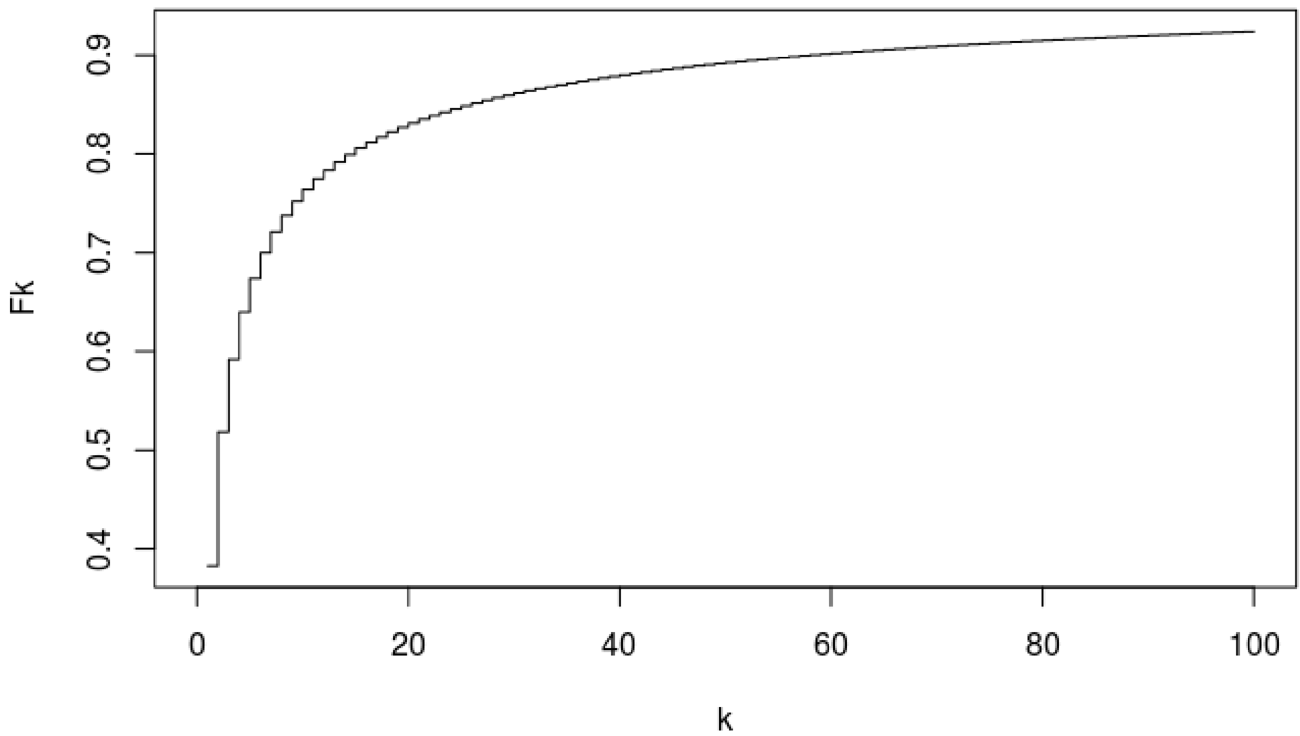
Hide

```
k  = 0:100
pk = 0.3828484 / (k^1.5)
plot(k, pk, type = "h")
```



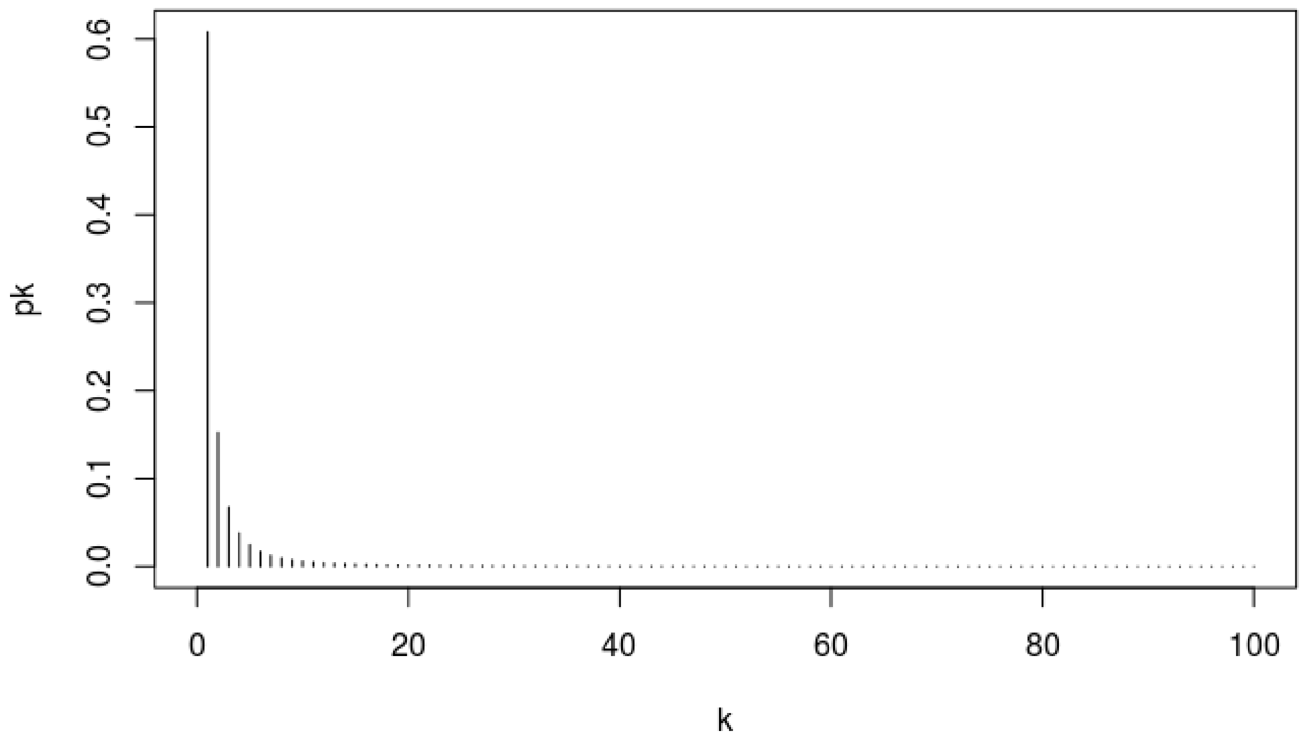
Hide

```
fk_list <- c()
for (k in 0:100){
  fk=0
  for(i in 1:k){
    fk = fk + (1/(i^1.5))
  }
  fk_list = append(fk_list, 0.3828484 * fk)
}
k  = 0:100
Fk = fk_list
plot(k, Fk, type = "s")
```



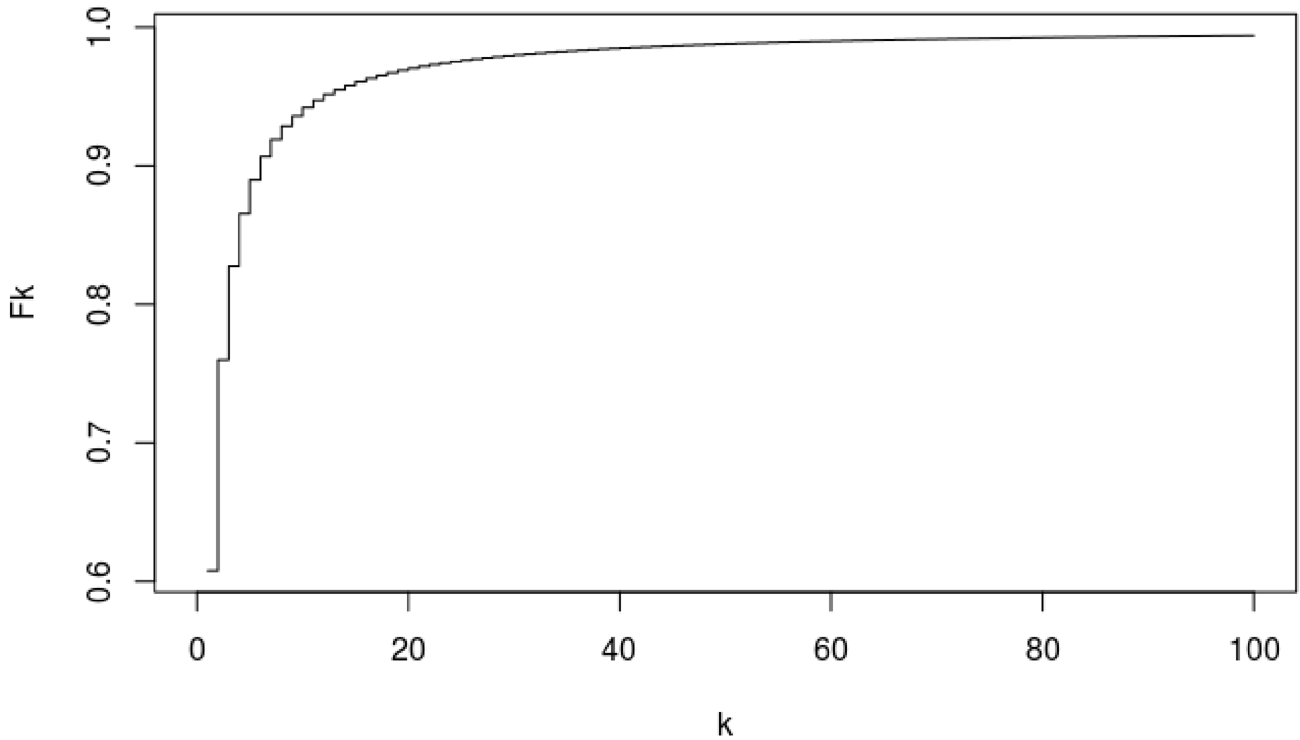
Hide

```
# α = 1 => C = 1/ζ(1+α) => C = 1/1.645 = 0.6079027
k  = 0:100
pk = 0.6079027 / (k^2)
plot(k, pk, type = "h")
```



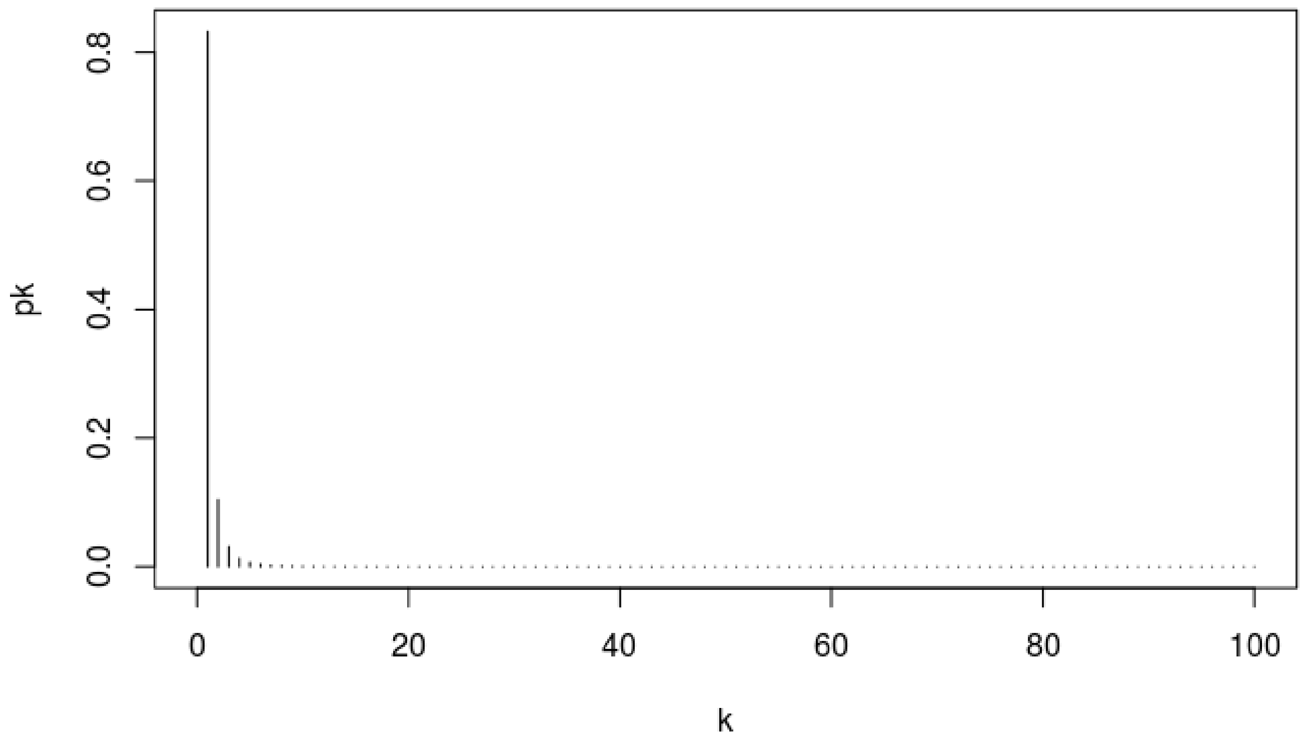
Hide

```
fk_list <- c()
for (k in 0:100){
  fk=0
  for(i in 1:k){
    fk = fk + (1/(i^2))
  }
  fk_list = append(fk_list, 0.6079027 * fk)
}
k  = 0:100
Fk = fk_list
plot(k, Fk, type = "s")
```



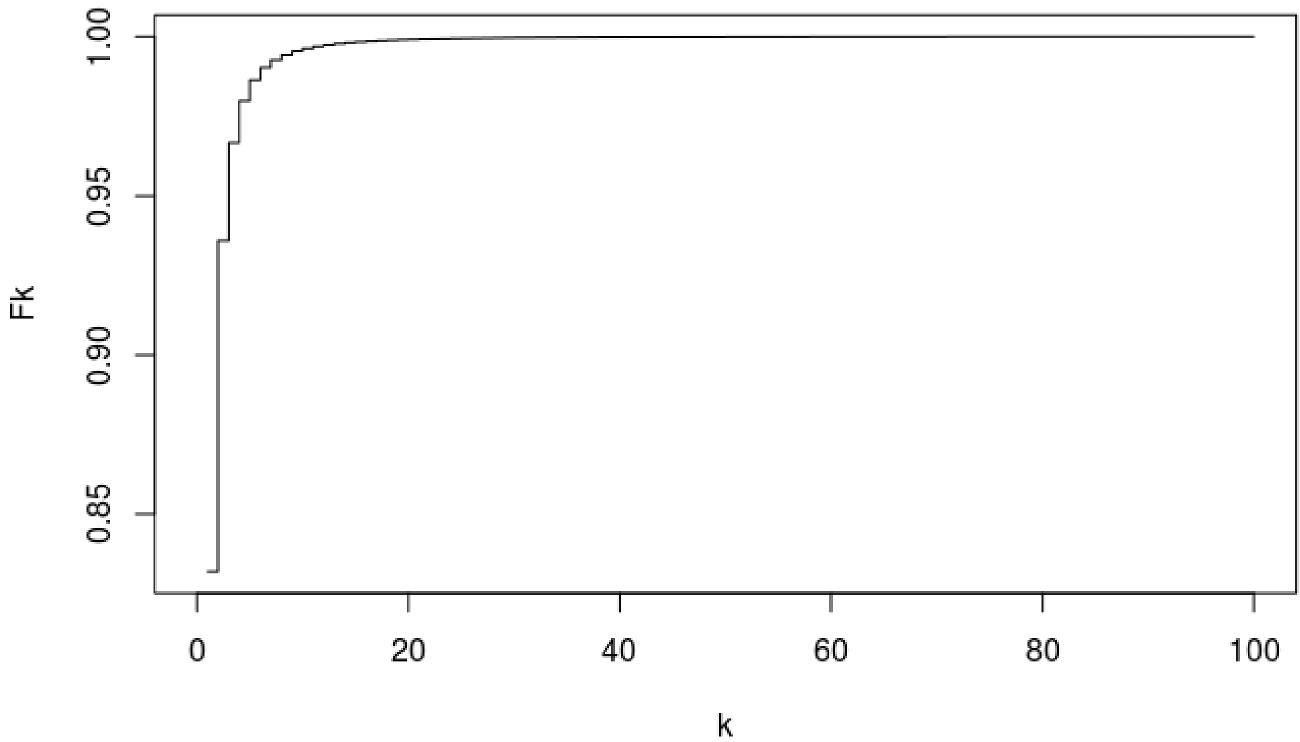
Hide

```
# α = 2 => C = 1/ζ(1+α) => C = 1/1.202 = 0.8319468
 k  = 0:100
 pk = 0.8319468 / (k^3)
 plot(k, pk, type = "h")
```



Hide

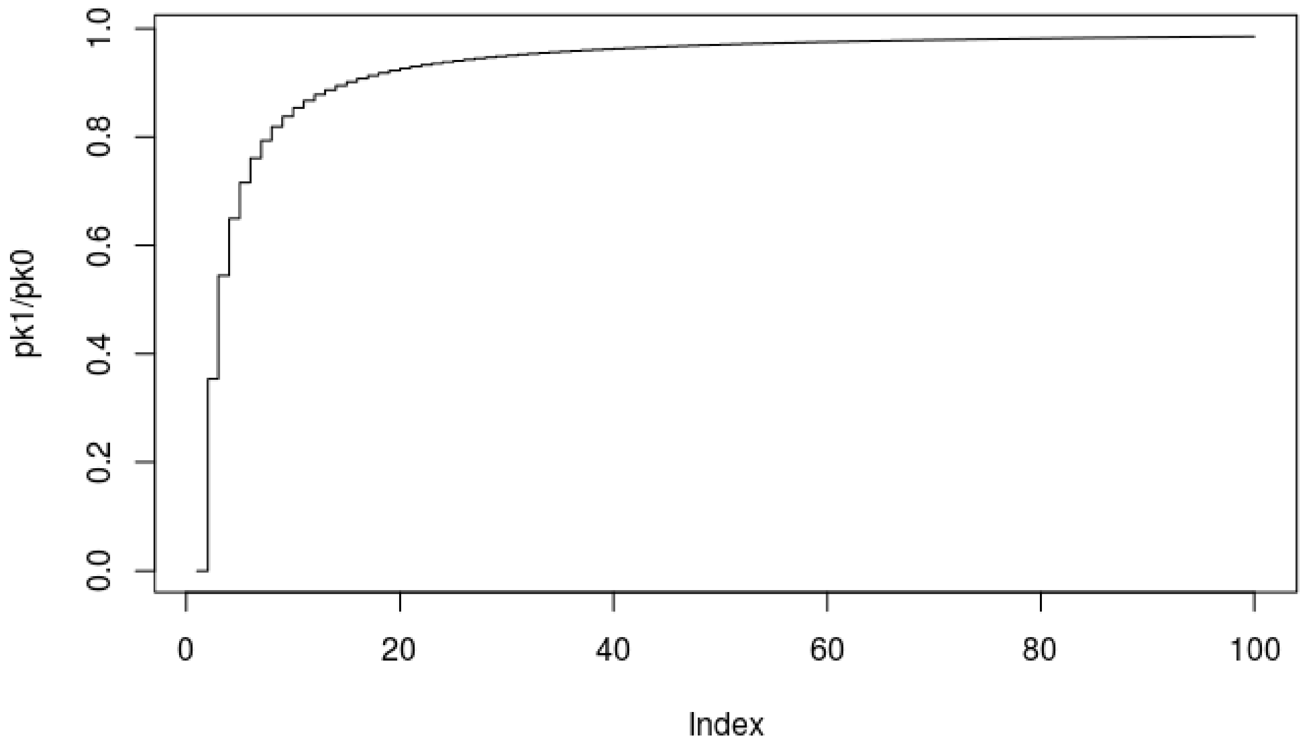
```
fk_list <- c()
for (k in 0:100){
  fk=0
  for(i in 1:k){
    fk = fk + (1/(i^3))
  }
  fk_list = append(fk_list, 0.8319468 * fk)
}
k  = 0:100
Fk = fk_list
plot(k, Fk, type = "s")
```



3.2

Hide

```
par(mfrow=c(1,1))
k0  = 0:99
k1  = 1:100
pk0 = 0.3828484 / (k0^1.5)
pk1 = 0.3828484 / (k1^1.5)
plot(pk1/pk0, type = "s")
```



Hide

```
NA
```

Quando k cresce, k/(k + 1) eh sempre menor que 1 mas cada vez mais proximo de 1

3.3

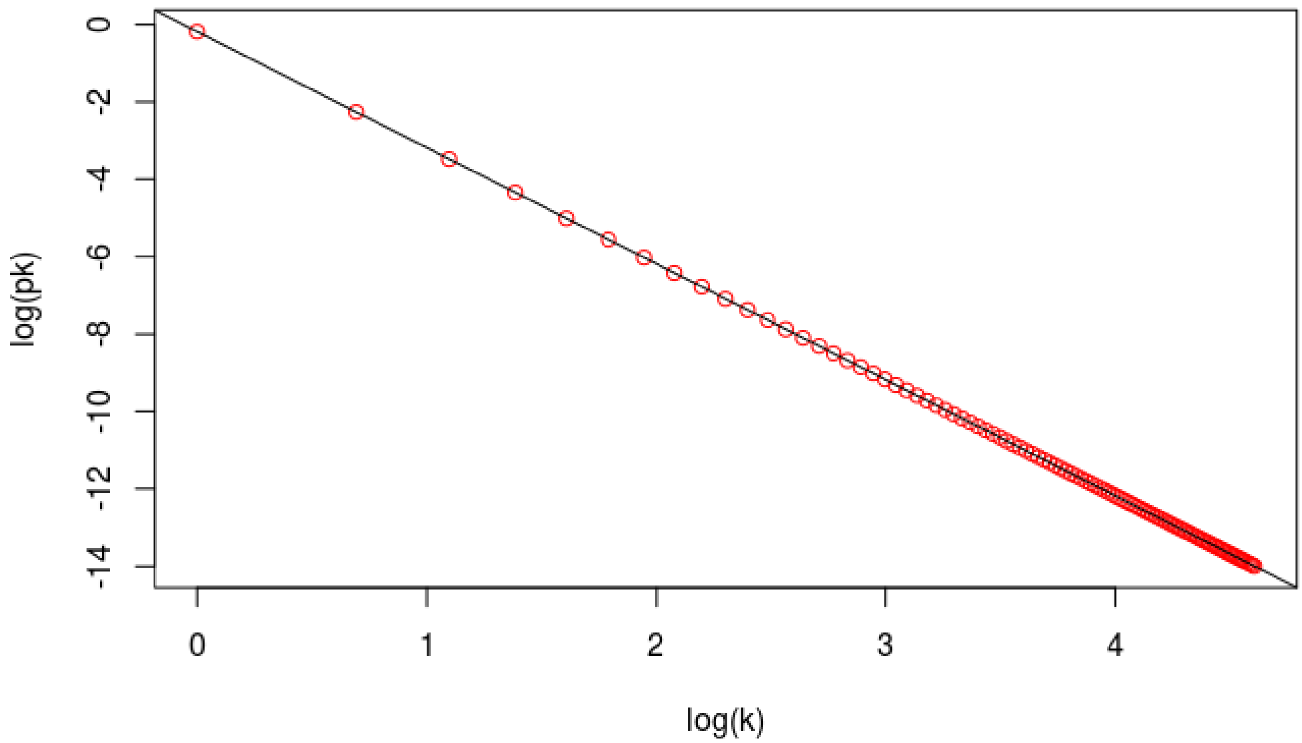
Quando α maior, crescimento eh mais rapido e tambem mais rapido, fracao de k/(k + 1) fica perto de 1.

3.4

alpha = 2 => C = 1/ζ(1+α) => C = 1/1.202 = 0.8319468

Hide

```
alpha=2
k  = 0:100
pk = 0.8319468 / (k^3)
plot(log(k), log(pk), col = "red")
abline(log(0.8319468), -(1 + alpha))
```



A linha é exatamente acima de plot anterior e isso significa que a linha mostra a inclinacao de plot log

3.5

Hide

```
rzipf = function(nsim = 1, alpha = 1, Cte = 1/1.645)
{
 res = numeric(nsim)
 for(i in 1:nsim){
   x = -1
   k = 1
   F = p = Cte
   U = runif(1)
   while( x == -1){
     if(U < F) x = k
     else{
       p = p * (k/(k+1))^(1+alpha)
       F = F + p
       k = k+1
     }
   }
   res[i] = x
 }
 res
}

# test
rzipf(400)
rzipf(400, 1/2, 1/2.62)
rzipf(400, 1, 1/1.645)
rzipf(400, 2, 1/1.202)
```

O resultado de rzipf(400, 1, 1/1.645)

Hide

```
rzipf(400, 1, 1/1.645)
```

```
  [1]    1    5    2    1    1    1    1    1    1    1    1    1    1    2    1    2
    2    2    1    1    1    5    1    1   11    2    3    1    3    2
 [31]    1    1    2    1    2    2    2    1    1    1    1    1    1    1    3    1
    1    1    2    1    1    1    1    2    1    1    1    1    1    1
 [61]    1    2    1    9    1    1    1    1    2    3   10    1    1    2    1    2
    2    1    1    1    1    1  120    1    1    1  147    1   27    2
 [91]    1    3    3    4    1   29    4    1    4    1    2    6    1    2    1    1
    1    1    1    1    1    1    3    1    3    1    1    1    2   38
[121]    1    1   20    8    5    1    3    1    2    1    1    4    6   34    2    4
    1    1    1    2    1    2    1    1    1    3    1    1    1    1
[151]    1    3    2    2    4    1    1    8    2    1    2    1    3   11    2    1
    1    3    1    1    1    1    2    1    1    2    2    1    1   11
[181]   80    1    1    8    3    2    7    1    2   94    1    5    1    1   14    5
    4    1    1    1   44    1   10    4    2   55    2    1    1    1
[211]    5   12    1    1    1    1    1    3    6    2    1    1    1    1    1    6
    8    1    1    3    3    1    2    1    2    1    1    7    1    1
[241]    1   12   17    1    1    2    1 3308    1    1    7    1    1    1   49    1
    2    8    1    1    1    6    1    2    1    2    1    1    1    1
[271]    1    1    7    2    1    1    2    1   21    1    1    2    3    1    1    1
    3    3    4    4    1    1    1    3    1    2    1    1    1    1
[301]    5    1    1    1    1    1    1    1    3    1    1    1    7    3    1   19
    1    1    1    1   24    1    1    5    1    1    1    1    1    2
[331]  176    1    1    1    1    2    3    1    2    6    1    1   10    1   16   66
    3    1    1    4    1    2    2    1   10    1   14    1    1    1
[361]    1    1    1    1    2    2    1    1    1    1    1    4    1    1    1    1
    1    1    7    1   15    1    1    1   12    2    1    1    9    2
[391]    5    1    3    1    1    2    1    1    8    1
```