Geometrica

Hernández Martínez Oscar Gerardo

29/6/2020

```
import os
os.environ['QT_QPA_PLATFORM_PLUGIN_PATH'] = 'C:/Users/jxsje/anaconda3/Library/plugins/platforms'
```

Función de Densidad

Sea $X \sim Geom(p = 0.1)$ la distribución que modela la probabildad de intentar abrir una puerta hasta cosneguirlo.

$$f(k) = (1-p)^{k-1}p$$

En R

```
library(Rlab)
```

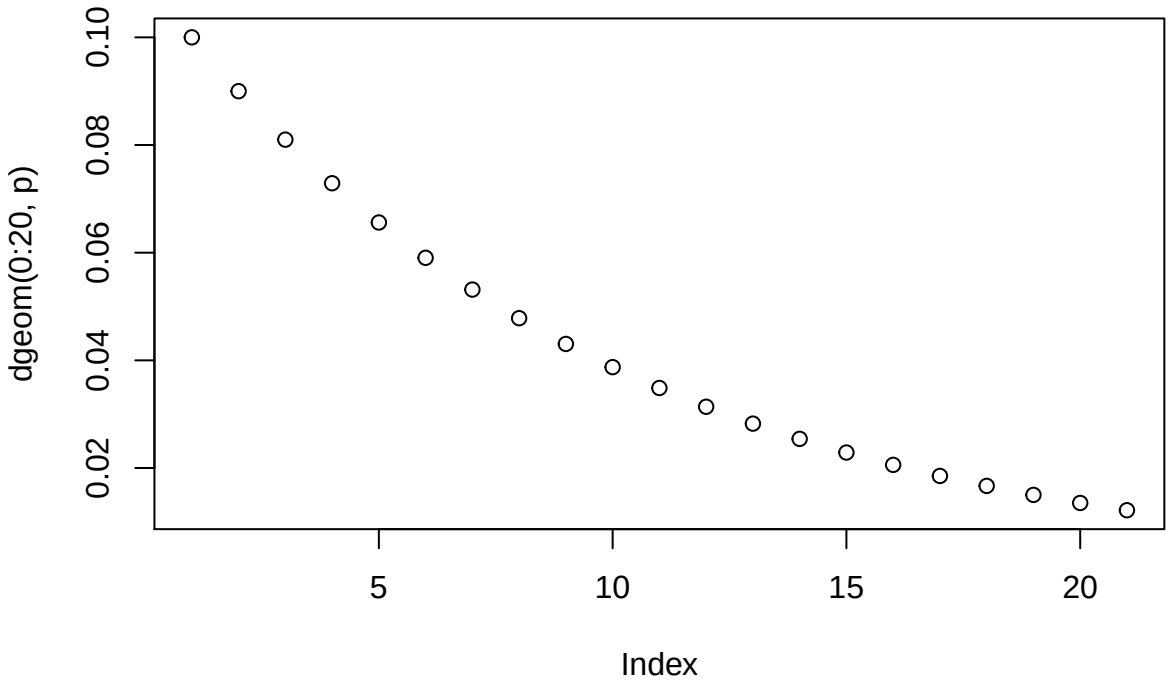
```
## Rlab 2.15.1 attached.
```

```
##
## Attaching package: 'Rlab'
```

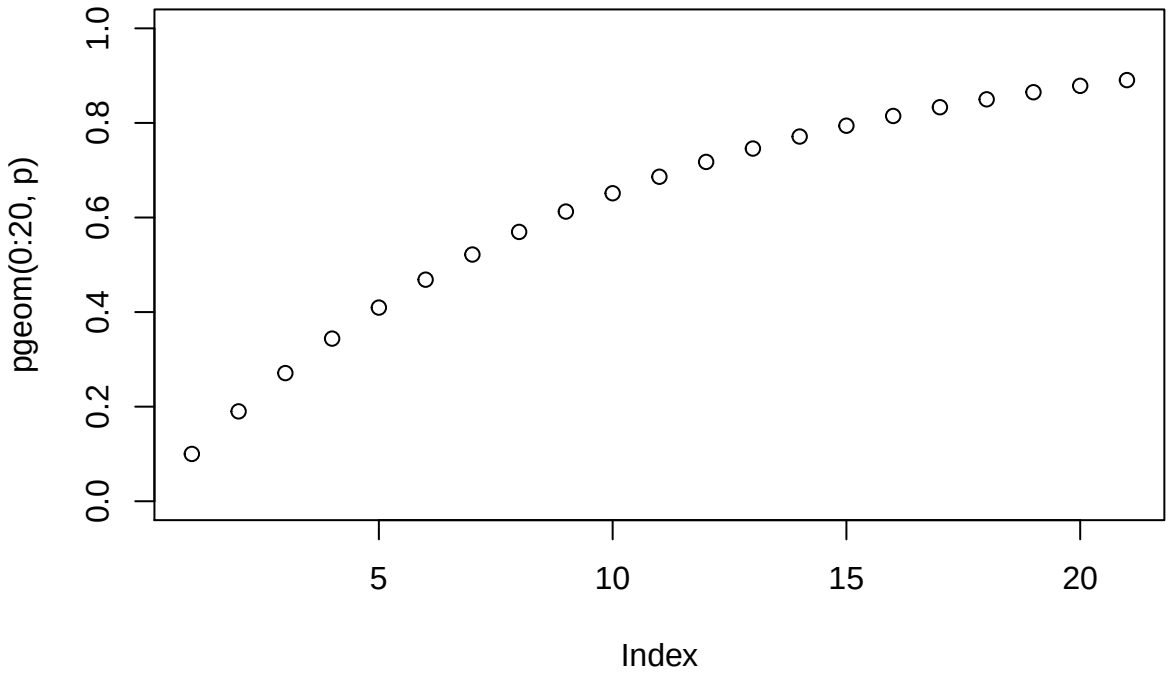
```
## The following objects are masked from 'package:stats':
##
##     dexp, dgamma, dweibull, pexp, pgamma, pweibull, qexp, qgamma,
##     qweibull, rexp, rgamma, rweibull
```

```
## The following object is masked from 'package:datasets':
##
##     precip
```

```
p = 0.1
plot(dgeom(0:20, p))
```




```
plot(pgeom(0:20, p), ylim = c(0,1))
```



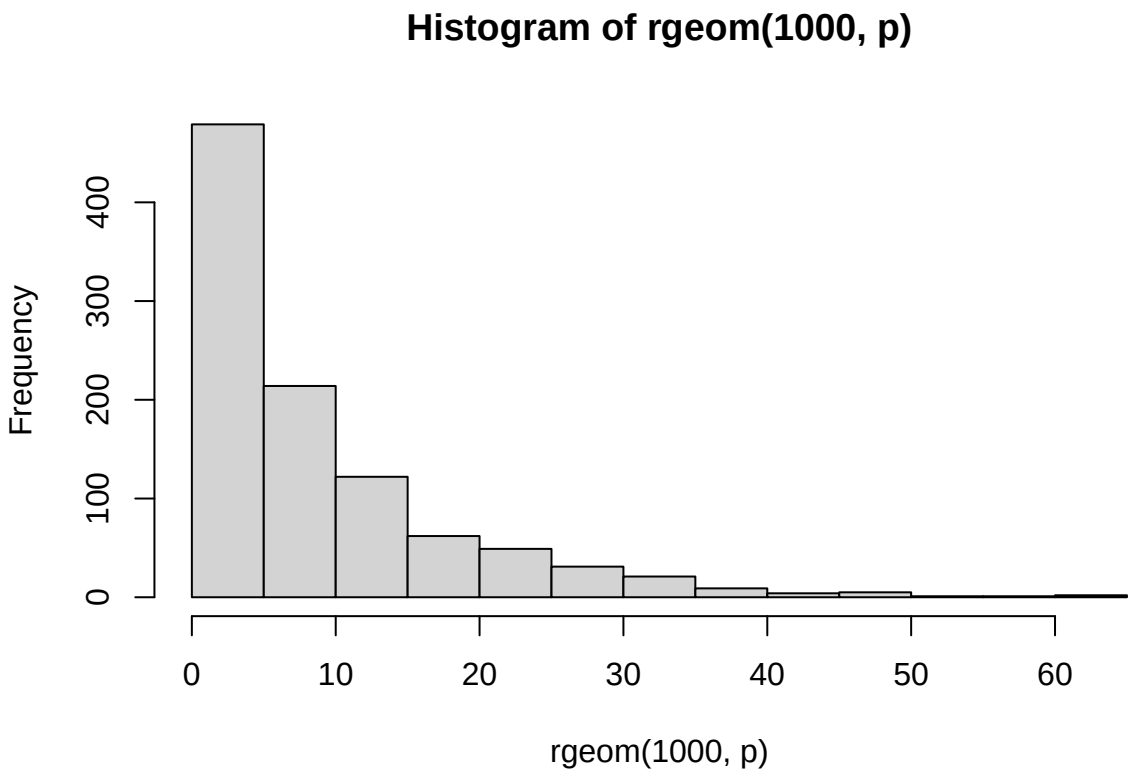

```
qgeom(0.5, p)
```

```
## [1] 6
```

```
qgeom(0.75, p)
```

```
## [1] 13
```

```
hist(rgeom(1000, p))
```



En Python

```
from scipy.stats import geom
import matplotlib.pyplot as plt
import numpy as np

fig, ax = plt.subplots(1,1)
p = 0.3
mean, var, skew, kurt = geom.stats(p, moments = 'mvsk')
print("Media %f"%mean)
```

```
## Media 3.333333
```

```
print("Varianza %f"%var)
```

```
## Varianza 7.777778
```

```
print("Sesgo %f"%skew)
```

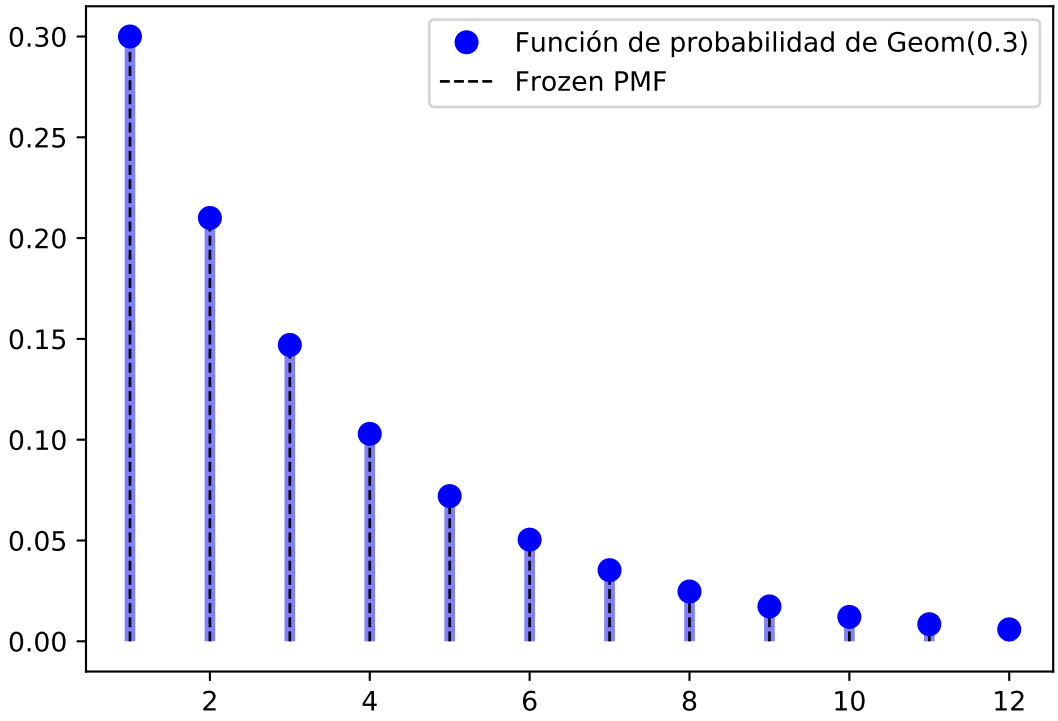
```
## Sesgo 2.031889
```

```
print("Curtosis %f"%kurt)
```

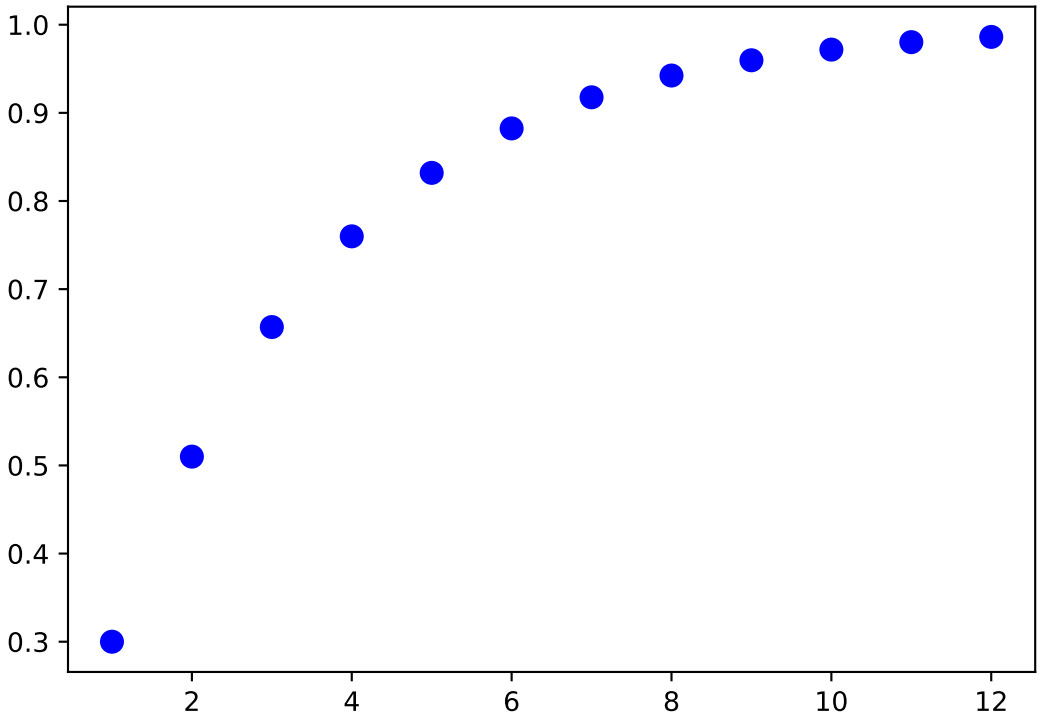
```
## Curtosis 6.128571
```

```
x = np.arange(geom.ppf(0.01,p), geom.ppf(0.99, p))
ax.plot(x, geom.pmf(x, p), 'bo', ms = 8, label = "Función de probabilidad de Geom(0.3)")
ax.vlines(x,0,geom.pmf(x,p),  colors = 'b', lw = 4, alpha = 0.5)
```

```
rv = geom(p)
ax.vlines(x,0,rv.pmf(x), colors = 'k', linestyles = '--', lw = 1, label = "Frozen PMF")
ax.legend(loc = 'best')
plt.show()
```



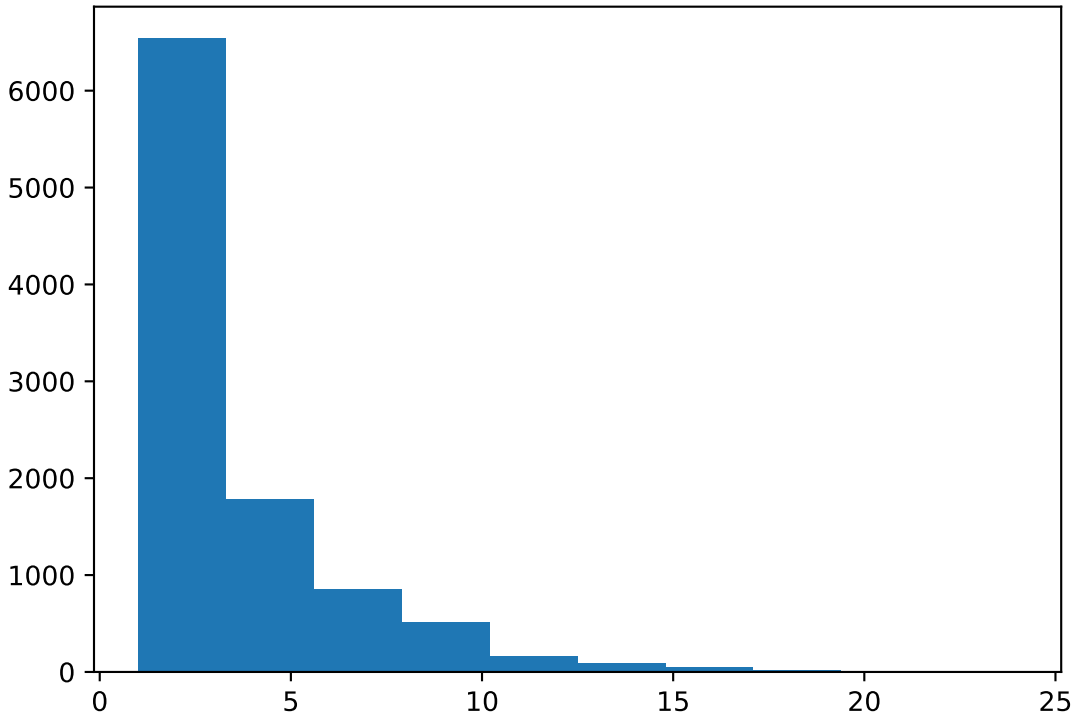

```
fig, ax = plt.subplots(1,1)
prob = geom.cdf(x,p)
ax.plot(x, prob, 'bo', ms = 8, label = "Función de distribución acumulada")
plt.show()
```




```
fig, ax = plt.subplots(1,1)
r = geom.rvs(p, size = 10000)
plt.hist(r)
```

```
## (array([6.540e+03, 1.783e+03, 8.560e+02, 5.080e+02, 1.570e+02, 8.600e+01,
##         4.500e+01, 1.200e+01, 8.000e+00, 5.000e+00]), array([ 1. ,  3.3,  5.6,  7.9, 10.2, 12.5, 14.8
```

```
plt.show()
```


6000
5000
4000
3000
2000
1000
0
0
5
10
15
20
25