Distribución exponencial y de Poisson

Fabiola Vázquez

29 de septiembre de 2020

---

1. Introducción

Los objetivos de este estudio, que se realiza con el software estadístico R versión 4.0.2 [4] en el IDE **R Studio** [5], son encontrar relaciones entre las distribuciones exponencial y uniforme con la distribución de Poisson con algún parámetro λ. Siguiendo con el análisis de la novela *Alice's Adventures in Wonderland* [1], un objetivo más es encontrar algún dato, palabra, frase, longitud, etcétera, que siga una distribución de Poisson.

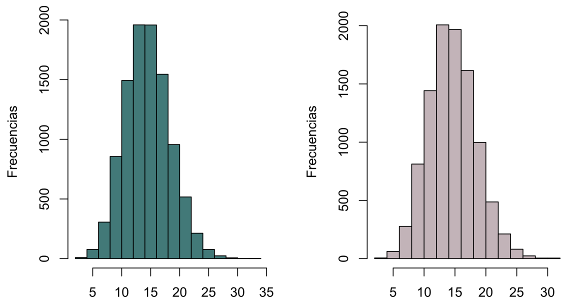
2. Uso de la distribución exponencial

En el primer estudio, se generan números distribuidos exponencialmente con parámetro `lambda` y se calcula la cantidad necesaria para que su suma exceda a una `meta`. Dicho proceso, se realiza una cierta cantidad de veces. El objetivo es ver si esos números que se producen y guardan en el vector `ce` se distribuyen Poisson con algún parámetro λ, el algoritmo fue realizado por la Dra. Elisa Schaeffer.

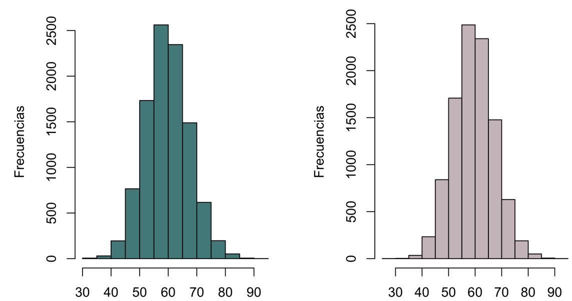
```
for (replica in 1:replicas) {
    de = numeric()

    while (sum(de) < meta) {
        de = c(de, rexp(1, lambda))
    }
    ce = c(ce, length(de)-1)
}
```

La media de la distribución exponencial es $\frac{1}{\lambda}$ [2]. Se sabe que la distribución de Poisson tiene como parámetro λ que es la tasa de llegadas en un intervalo de tiempo [6]. Entonces, **se espera** que la tasa de llegadas sea de `meta`·`lambda`.

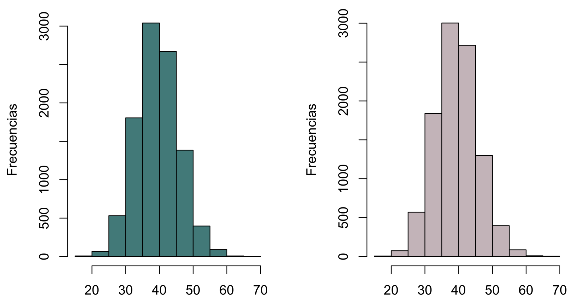
Se realiza unas pruebas, para ver si se va por buen camino, variando los valores de los parámetros `meta` y `lambda`. En la figura 1 se compara el histograma de los números generados, lado izquierdo,



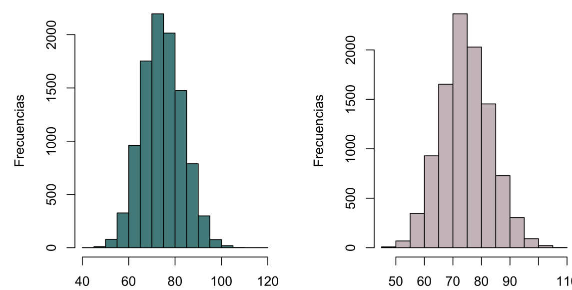
(a) Números generados con `meta`=3 y `lambda`=5 (izq) contra los generados con `rpois` (der).



(b) Números generados con `meta`=10 y `lambda`=6 (izq) contra los generados con `rpois` (der).



(c) Números generados con `meta`=4 y `lambda`=10 (izq) contra los generados con `rpois` (der).



(d) Números generados con `meta`=15 y `lambda`=5 (izq) contra los generados con `rpois` (der).

Figura 1: Pruebas usando el algoritmo para crear números distribuidos Poisson.

contra el histograma de valores generados con `rpois(replica,me*la)`, lado derecho. Visualmente los histogramas son idénticos.

Se utiliza la función `goodfit` del paquete `vcd` para hacer una prueba de bondad de ajuste Chi cuadrada, para analizar si el conjunto de datos generados se distribuye de forma Poisson. La función nos regresa también el valor del parámetro λ necesario.

En el cuadro 1 se ve que en las pruebas realizadas, el valor obtenido mediante `goodfit` se asemeja, en efecto, al valor de `meta`·`lambda`.

Otra perspectiva al experimento, es ver que los números distribuidos exponencialmente que se generan son las diferencias entre los eventos de Poisson, como se ve en la figura 2.

Teorema 1. *El tiempo necesario para que ocurra el primer evento de Poisson tiene distribución exponencial, con parámetro λt.*

Demostración. Se considera que $X \sim \text{Poisson}(\lambda t)$. La probabilidad de obtener n llegadas en un

Cuadro 1: Comparación de los parámetros usados y el lambda dado por `goodfit`.

Prueba	Meta	Lambda	Lambda (goodfit)
1	3	5	15.019
2	10	6	59.869
3	4	10	39.982
4	15	5	74.995

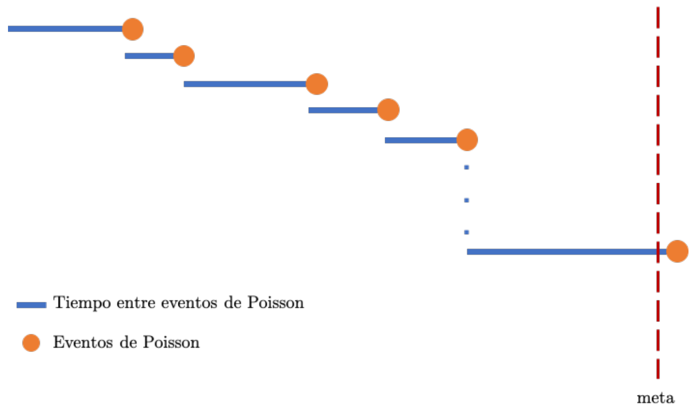


Figura 2: Visualización del experimento

intervalo, se obtiene como

$$P(X = n) = \frac{(\lambda t)^n}{n!} e^{-\lambda t} \tag{1}$$

Entonces, la probabilidad de que no haya llegadas en el intervalo sería

$$P(X = 0) = e^{-\lambda t} \tag{2}$$

La probabilidad de que T exceda x es la misma que la probabilidad de que no ocurra ningún evento de Poisson en x.

$$P(T > x) = e^{-\lambda t} \tag{3}$$

Entonces,

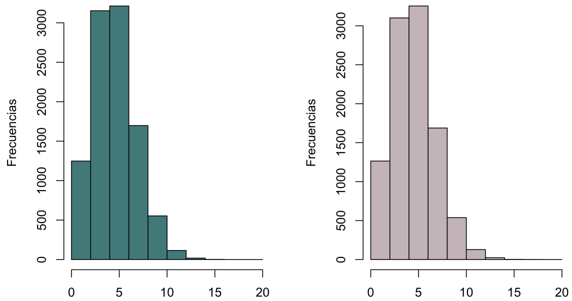
$$P(T \leqslant x) = 1 - P(T > x) = 1 - e^{-\lambda t} \tag{4}$$

□

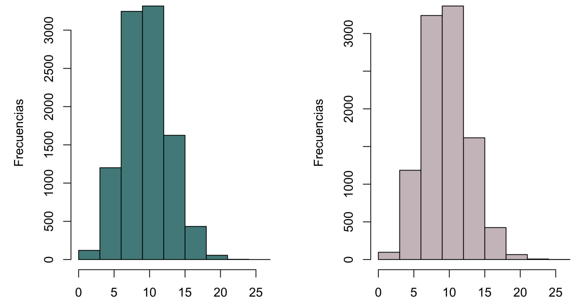
3. Uso de la distribución uniforme

Se generan números distribuidos uniformemente y con el *algoritmo de Knuth* [3] que calcula el producto de dichos números mientras sea mayor que `e**-lambda`, se guarda la cantidad de números. Se quiere ver si los números generados siguen una distribución de Poisson con parámetro `lambda`.

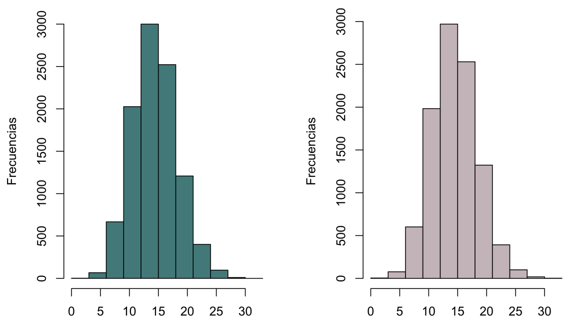
Se realizan pruebas variando los valores de `lambda`. Los histogramas se aprecian en la figura 2 y las comparaciones con los valores de `lambda` obtenidos con `goodfit` se encuentran en el cuadro 2.



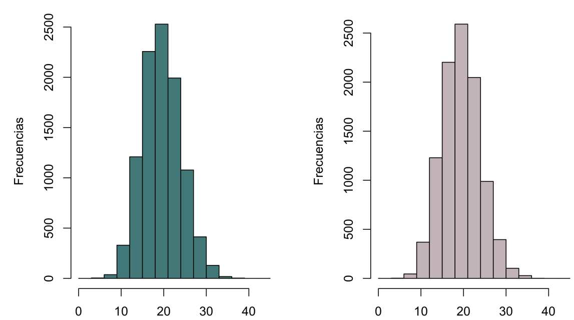
(a) Números generados con `lambda`=5 (izq) contra los generados con `rpois` (der).



(b) Números generados con `lambda`=10 (izq) contra los generados con `rpois` (der).



(c) Números generados con `lambda`=15 (izq) contra los generados con `rpois` (der).

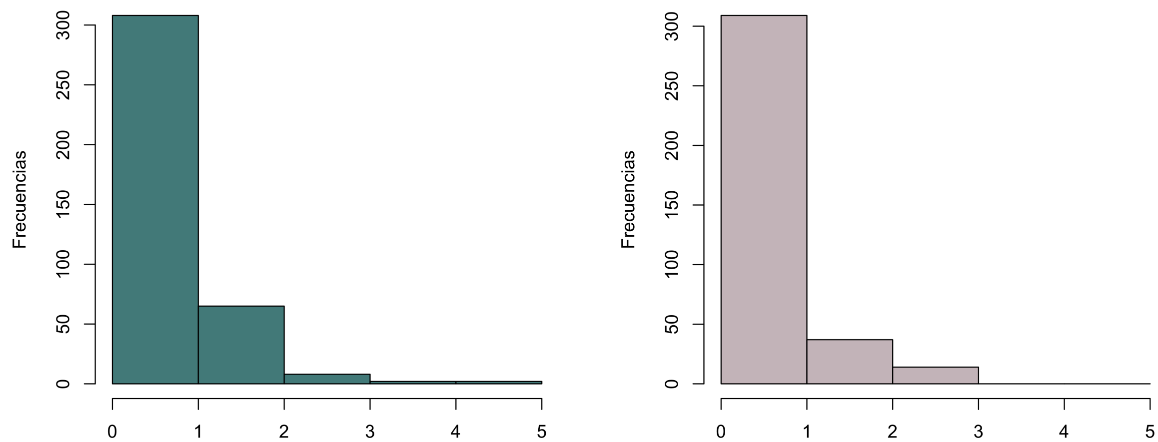


(d) Números generados con `lambda`=20 (izq) contra los generados con `rpois` (der).

Figura 3: Experimentos usando el algoritmo de Knuth.

Cuadro 2: Comparación del lambda usado y el lambda dado por `goodfit`.

Prueba	Lambda	Lambda (goodfit)
1	5	4.987
2	10	10.009
3	15	15.0388
4	20	20.013



(a) Diferencia entre los tiempos de aparición de la palabra *Alice*.

(b) Números generados con distribución Poisson.

Figura 4: Tiempos entre apariciones de la palabra *Alice*.

4. Distribución de Poisson en el libro de texto

Continuando con el estudio de la novela *Alice's Adventures in Wonderland* [1], se busca en qué tiempos aparece la palabra `Alice` y comparar su distribución con la de números generados con `rpois`.

Consideramos una unidad de tiempo como cien palabras. Se obtiene el promedio de eventos en una unidad de tiempo, es decir, en promedio cuantas veces aparece `Alice` en cien palabras. Se encuentra que dichos valores siguen una distribución de Poisson con λ igual al promedio de los datos. En la figura 4 se muestra en la izquierda del histograma de los datos generados con el algoritmo y en la derecha el histograma de los números generados con `rpois`.

Referencias

[1] Lewis Carroll. *Alice's Adventures in Wonderland*. Macmillan, Oxford, England, 1865.

[2] George Casella and Roger L. Berger. *Statistical inference*. Thomson Learning, Australia ; Pacific Grove, CA, 2nd edition, 2002.

[3] Donald Ervin Knuth. *The art of computer programming*. Addison-Wesley, 3rd ed edition, 1997.

[4] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020.

[5] RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio, PBC., Massachusetts, United States, 2020. http://www.rstudio.com/.

[6] Elisa Schaeffer. Modelos probabilistas aplicados. https://elisa.dyndns-web.com/teaching/prob/pisis/prob.html#t5.