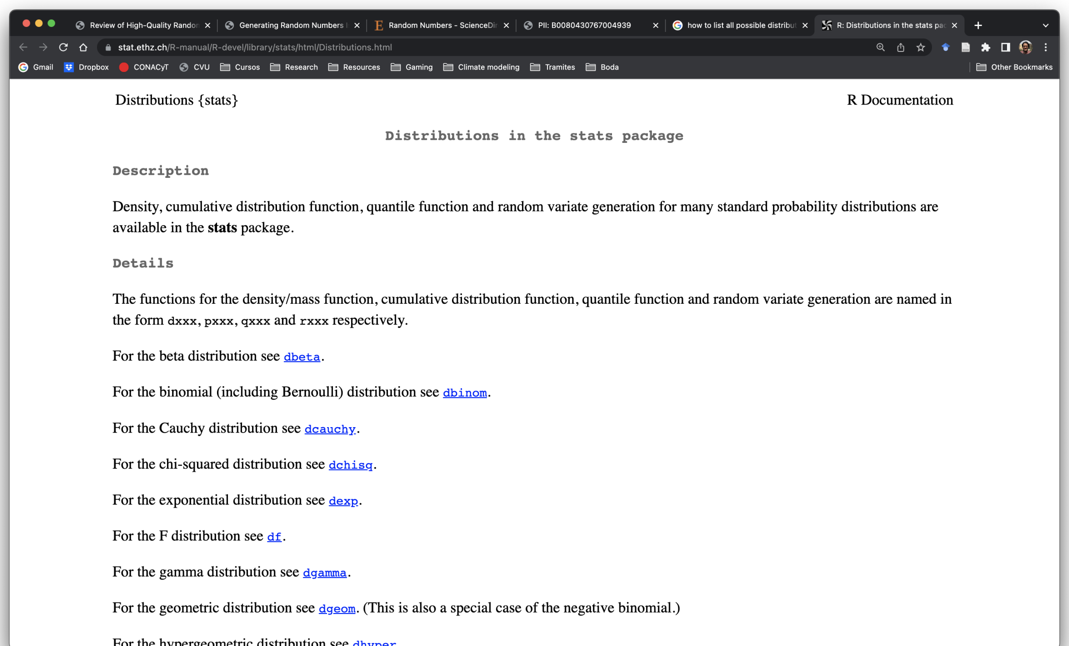
EST-24107: Simulación

Profesor: Alfredo Garbuno Iñigo — Primavera, 2022 — Números no uniformes.
Objetivo: Que veremos.
Lectura recomendada: Referencia.

1. GENERACIÓN DE VARIABLES NO UNIFORMES

`R`, por ejemplo, tiene distintos generadores de variables aleatorias. Un catálogo nos muestra que podemos generar mucho mas variables que solamente una variable uniforme.



Distributions {stats} R Documentation

Distributions in the stats package

Description

Density, cumulative distribution function, quantile function and random variate generation for many standard probability distributions are available in the **stats** package.

Details

The functions for the density/mass function, cumulative distribution function, quantile function and random variate generation are named in the form `dxxx`, `pxxx`, `qxxx` and `rxxx` respectively.

For the beta distribution see `dbeta`.

For the binomial (including Bernoulli) distribution see `dbinom`.

For the Cauchy distribution see `dcauchy`.

For the chi-squared distribution see `dchisq`.

For the exponential distribution see `dexp`.

For the F distribution see `df`.

For the gamma distribution see `dgamma`.

For the geometric distribution see `dgeom`. (This is also a special case of the negative binomial.)

For the hypergeometric distribution see `dhyper`.

FIGURA 1. *Catálogo de distribuciones en* `R`.

En esta sección exploraremos los mecanismos tradicionales para generar números pseudo-aleatorios de distribuciones un poco mas generales que una distribución uniforme.

1.1. Método de transformada inversa

2. DISTRIBUCIONES CONTINUAS

2.1. Prueba `KS`

3. DISTRIBUCIONES DISCRETAS

4. PRUEBA χ^2

Podemos usar otro mecanismo para probar estadísticamente si nuestros números pseudo aleatorios siguen la distribución que deseamos.

Podemos pensar en esta alternativa como la versión `discreta` de la prueba `KS`.

Lo que estamos poniendo a prueba es

$$H_0 : \mathbb{P}(x) = \mathbb{P}_0(x) \ \forall x \quad \text{contra} \quad H_1 : \mathbb{P}(x) \neq \mathbb{P}_0(x) \text{ para alguna } x\,. \tag{1}$$

4.1. Procedimiento de la prueba χ^2

1. Hacemos una partición del rango de la distribución supuesta en k subintervalos con límites $\{a_0, a_1, \ldots, a_k\}$, y definimos N_j como el número de observaciones (de nuestro generador de pseudo-aleatorios) en cada subintervalo.
2. Calculamos la proporción esperada de observaciones en el intervalo $(a_{j-1}, a_j]$ como

$$p_j = \int_{a_{j-1}}^{a_j} \mathrm{d}\mathbb{P}(x)\,. \tag{2}$$

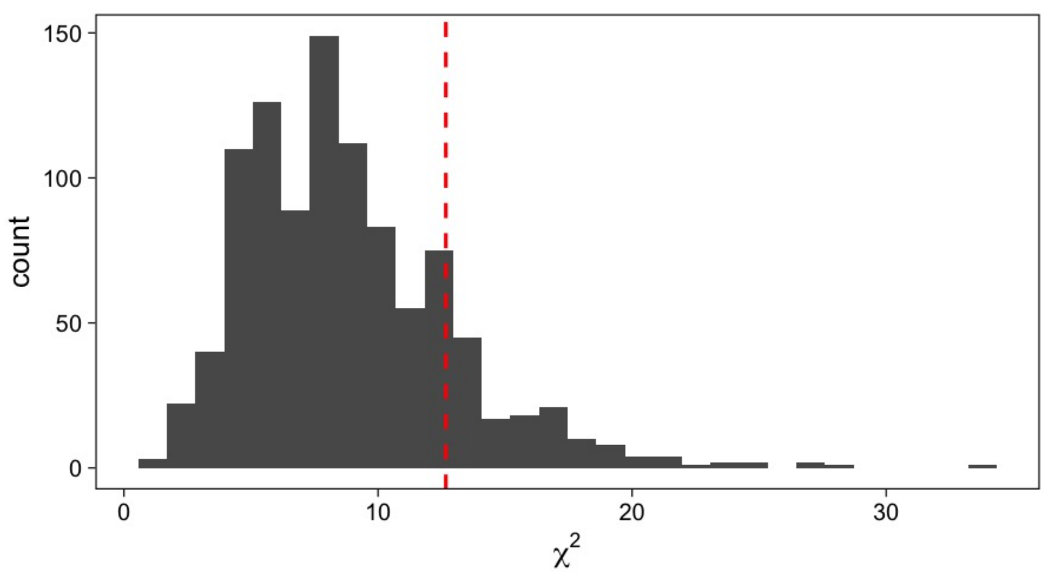
3. La estadística de prueba es

$$\chi^2 = \sum_{j=1}^{k} \frac{(N_j - p_j)^2}{np_j}\,. \tag{3}$$

> Nota que estamos comparando dos histogramas. El histograma observado que construimos a partir de nuestros números pseudo-aleatorios contra el histograma que esperaríamos de la distribución. ¿Puedes pensar en algún problema con esta prueba?

4.1.1. Pregunta: ¿Qué esperaríamos de nuestro estadístico χ^2 si nuestro generado de pseudo-aleatorios es incorrecto?

4.2. Aplicación de la prueba



Por lo tanto, la probabilidad de haber observador una estadístico χ^2 tan extremo como el que observamos si el generador hubiera sido el que suponemos es:

```
[1] "Estadistico: 12.6667, Probabilidad: 0.164"
```

Que podemos comparar contra el que obtenemos de una prueba ”tradicional”:

```
counts.obs <- Fn*nsamples
chisq.test(counts.obs, p = rep(1, nbreaks)/nbreaks, simulate.p.value = TRUE)
```

```

        Chi-squared test for given probabilities with simulated p-value (based
        on 2000 replicates)

data:  counts.obs
X-squared = 13, df = NA, p-value = 0.2
```

- La prueba χ^2 pues usualmente no es buena cuando el número de observaciones es menor a 50.
- La prueba `KS` tiene mejor potencia que la prueba χ^2:

```
ks.test(samples$x, "punif")
```

```

        Exact one-sample Kolmogorov-Smirnov test

data:  samples$x
D = 0.16, p-value = 0.4
alternative hypothesis: two-sided
```

4.3. Aplicación de pruebas

En la práctica se utiliza una colección de pruebas pues cada una es sensible a cierto tipo de desviaciones. La bateria de pruebas mas utilizada es la colección de pruebas `DieHARD` que desarrolló George Marsaglia y que se ha ido complementando con los años.

REFERENCIAS