Práctica 4

Cálculo de probabilidades y simulación

Contenido

1. Cálculo de probabilidades con R Commander

El menú *Distribuciones* de *R Commander* contiene un amplio conjunto de distribuciones de probabilidad, agrupadas en discretas y continuas. Para cada una de ellas, hay cuatro posibilidades:

- ***Cuantiles***. Es el menor valor c tal que $\Pr(X \leq c) \geq p$ (cola inferior o izquierda) o que $\Pr(X > c) \leq p$ (cola superior o derecha).
- ***Probabilidades***. En variables discretas, da los valores de la función de probabilidad, es decir, $\Pr(X = k)$ para cierto k.
- ***Probabilidades acumuladas***. Dado un valor k, calcula la probabilidad $\Pr(X \leq k)$ (cola izquierda) o $\Pr(X > k)$ (cola derecha).
- ***Gráfica***. Dibuja la gráfica de la función de densidad (para variables continuas) o de probabilidad (para discretas), o bien la función de distribución.
- ***Muestra***. Permite generar un nuevo conjunto de datos aleatorio indicando los parámetros de la distribución y las cantidades de filas y columnas deseadas (*simular una muestra de la v.a.*).

La tabla siguiente muestra los nombres en *R* de varias funciones junto con argumentos adicionales y por defecto.

Distribución	Nombre R	Argumentos adicionales	Argumentos por defecto
Binomial	binom	Size (n), prob (p)	
Poisson	pois	lambda (λ)	
Uniforme continua	unif	min (a), max (b)	min = 0, max = 1
Normal	norm	mean (μ), sd (σ)	mean = 0, sd = 1
Exponencial	exp	rate (λ= 1/μ)	rate = 1
Weibull	weibull	shape (α), scale (β)	scale = 1

A cada nombre de función dado por *R* se le agrega un prefijo “d” para obtener el valor de la función de densidad (v.a. continua) o de la función de probabilidad (v.a. discreta), “p” para calcular el valor de la función de distribución, “q” para obtener los percentiles y “r” para generar variables pseudo-aleatorias. La sintaxis es la siguiente:

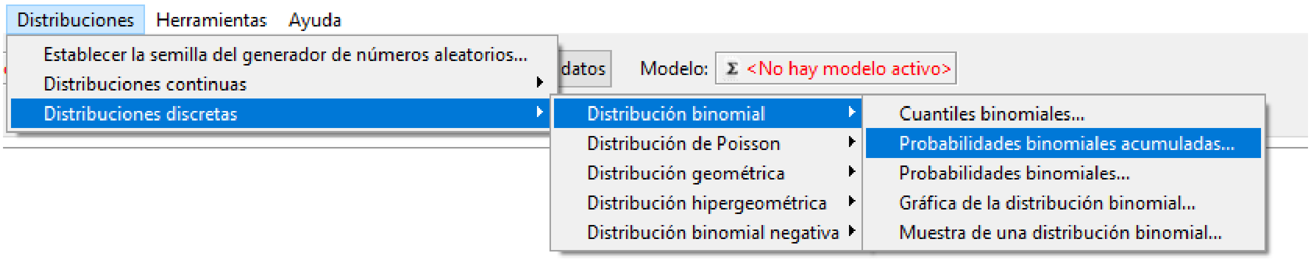
dnombre(c, ...) # calcula el valor de la función de probabilidad o de la función de densidad en c

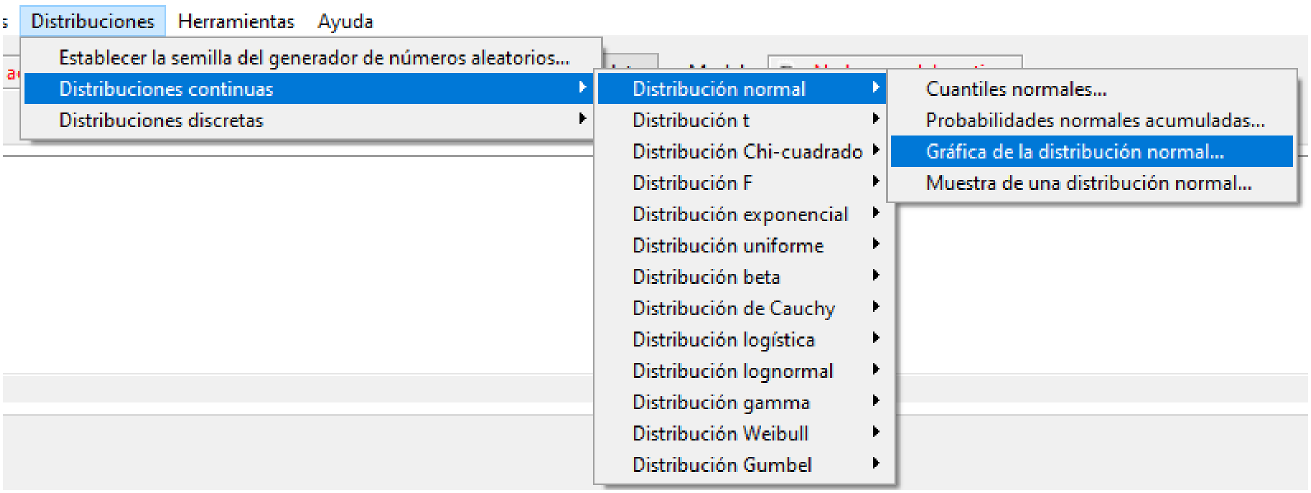
pnombre (c, ...) # calcula el valor de la función de distribución en c

qnombre(p, ...) # proporciona el p-ésimo percentil

rnombre(n, ...) # simula n observaciones

donde *nombre* indica el nombre de cualquiera de las distribuciones, *c* es un vector de valores de la v.a., *p* es un vector de probabilidades y *n* es un valor entero.





1.1. Variables aleatorias discretas

R te proporciona las probabilidades $\Pr(X \leq k)$ (cola izquierda, incluye el valor *k*) y $\Pr(X > k)$ (cola derecha, no incluye el valor *k*). Si *X* es una variable aleatoria discreta, para obtener las probabilidades $\Pr(X < k)$ o $\Pr(X \geq k)$ debes tener en cuenta que $\Pr(X < k) = \Pr(X \leq k\text{-}1)$ y $\Pr(X \geq k) = \Pr(X > k\text{-}1)$.

Ejercicio 1. **Sea *X* una v.a. con distribución binomial de parámetros n=8 y p=0.4 ($X \equiv$ binom(8, 0.4)).**

a) Para obtener $\Pr(X = 7)$ sigue la secuencia:
Distribuciones → Distribuciones discretas→ Distribución binomial → Probabilidades binomiales
y rellena la ventana emergente de la siguiente forma: *Ensayos binomiales* = 8, *Probabilidad de éxito* = 0.4 y acepta.

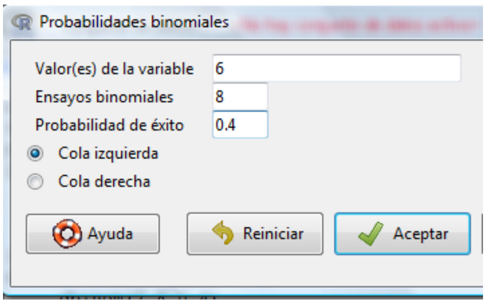
En la ventana de resultados obtienes:

	Probability	
0	0.01679616	
1	0.08957952	
2	0.20901888	
3	0.27869184	
4	0.23224320	
5	0.12386304	
6	0.04128768	
7	0.00786432	$\boxed{Pr(X = 7)}$
8	0.00065536	

- También puedes obtener este resultado escribiendo en la ventana de instrucciones:

```
dbinom(7,size=8,prob=0.4)  o   dbinom(7,8,0.4)
```

b) Si quieres calcular Pr($X \leq 6$) sigue la secuencia:
Distribuciones → Distribuciones discretas→ Distribución binomial → Probabilidades binomiales acumuladas
y rellena la ventana emergente, como sigue:



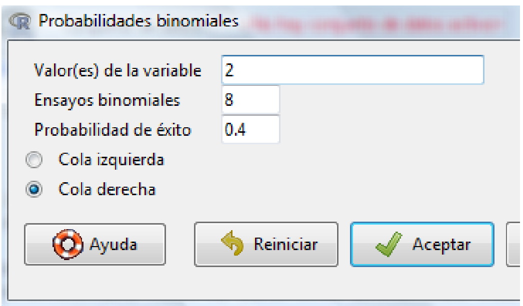
En la ventana de resultados se muestra:

> pbinom(c(6), size=8, prob=0.4, lower.tail=TRUE)

[1] 0.9914803 ⇒ Pr($X \leq 6$) = 0’9914803

c) Para calcular con *R* la probabilidad Pr($X \geq 3$) tenemos que obtener la probabilidad Pr($X > 2$), para ello seguimos el proceso:

Distribuciones → Distribuciones discretas→ Distribución binomial → Probabilidades binomiales acumuladas
y rellena la ventana emergente, como sigue:



En la ventana de resultados aparece:

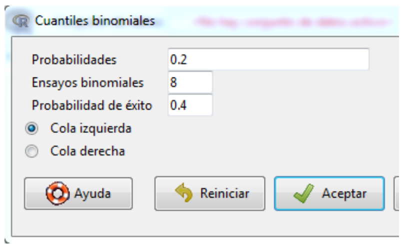
> pbinom(c(2), size=8, prob=0.4, lower.tail=FALSE)

[1] 0.6846054 $\Rightarrow$ Pr($X \geq 3$) = 0'6846054

d) ¿Qué valor de *X* deja al menos al 20% de los valores por debajo? Nos piden el valor c tal que Pr($X \leq c$) $\geq$ 0'20 (cola inferior). Para calcular este valor sigue la secuencia:

Distribuciones → Distribuciones discretas→ Distribución binomial → cuantiles

y rellena la ventana emergente, como sigue:



En la ventana de resultados figura:

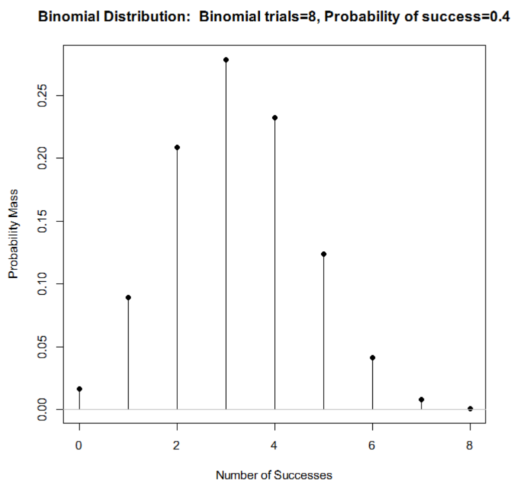
> qbinom(c(0.2), size=8, prob=0.4, lower.tail=TRUE)

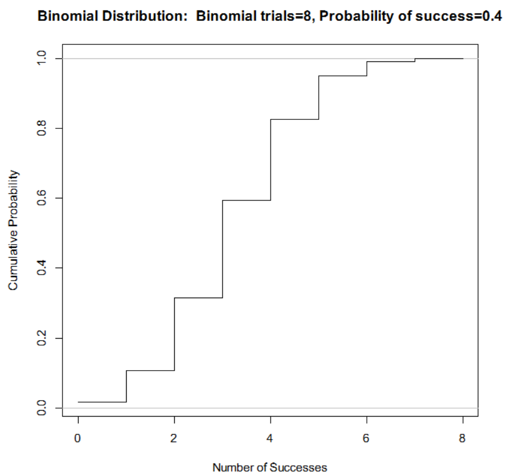
[1] 2 $\Rightarrow$ $c = 2$

e) Para representar gráficamente la función de probabilidad y la función de distribución acumulada de *X* procede como sigue:

Distribuciones → Distribuciones discretas→ Distribución binomial → Gráfica de la distribución binomial

rellena la ventana emergente con los valores de *n* y *p*, y selecciona la gráfica que te interesa obtener.





Ejercicio 2. **Sea *X* una v.a. con distribución de Poisson de parámetro 5 ($X \equiv$ pois(5)).**

Para hallar la probabilidad Pr($2 \leq X < 7$), en primer lugar debemos expresarla en la forma Pr($a < X \leq b$):

$$\Pr(2 \le X < 7) = \Pr(1 < X \le 6) = F_X(6) - F_X(1) = \Pr(X \le 6) - \Pr(X \le 1)$$

Para calcular esta probabilidad sigue la siguiente secuencia dos veces, una con *valor de la variable* = 6 y otra con *valor de la variable* = 1 y calcula la diferencia entre los resultados obtenidos:

Distribuciones → Distribuciones discretas→ Distribución Poisson → Probabilidades de Poisson acumuladas

también puedes hacerlo escribiendo en la ventana de instrucciones:

```
probs <- ppois(c(6, 1), lambda=5, lower.tail=TRUE)

prob <- probs[1] - probs[2]; prob
```

o bien:

```
ppois(6, 5) – ppois(1,5)
```

En la ventana de resultados se muestra:

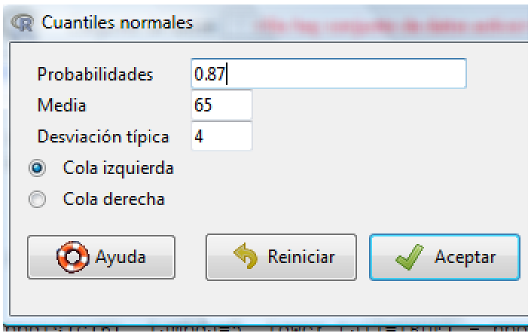
[1] 0.7217558 ⇒ $\Pr(2 \le X < 7) = 0'7217558$

1.2. Variables aleatorias continuas

Recuerda que si X es una variable aleatoria continua entonces $\Pr(X = k) = 0$.

Ejercicio 3. **Sea X una v.a. con distribución normal ($X \equiv$ norm(65, 4)).**

a) ¿Cuál es el valor de c tal que $\Pr(X < c)=0'87$?
Como X es una v.a. continua buscaremos el valor c tal que $\Pr(X \le c) = 0'87$. La secuencia para calcularla es:
Distribuciones → Distribuciones continuas → Distribución normal → Cuantiles normales…



La salida de *R* es:
> qnorm(c(0.87), mean=65, sd=4, lower.tail=TRUE)
[1] 69.50556 ⇒ $c = 69'50556$

b) Halla $\Pr(52 < X < 75)$. Ya que $\Pr(52 < X < 75) = \Pr(52 < X \le 75) = \Pr(X \le 75) - \Pr(X \le 52)$

Escribe en la ventana de instrucciones:

```
probs <- pnorm(c(75, 52), mean=65, sd=4, lower.tail=TRUE)
prob <- probs[1] - probs[2]; prob
```

o bien:

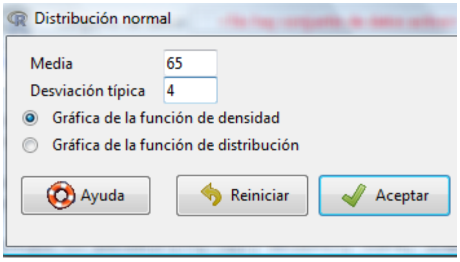
```
pnorm(75, 65, 4) - pnorm(52, 65, 4)
```

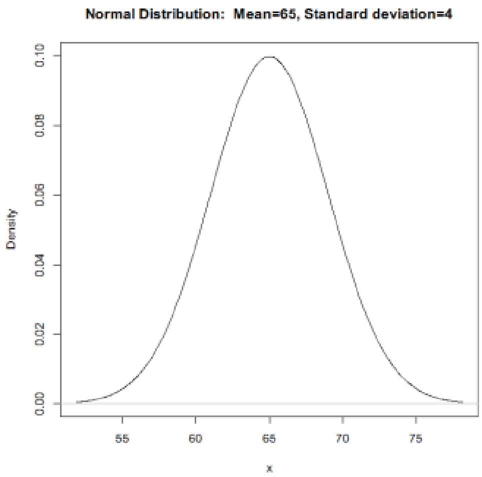
En la ventana de resultados se muestra:

[1] 0.9932133 ⇒ Pr(52 < X < 75) = 0’9932133

c) Representa gráficamente la función de densidad de *X.* Secuencia:

Distribuciones → Distribuciones continuas→ Distribución normal → Gráfica de la distribución normal





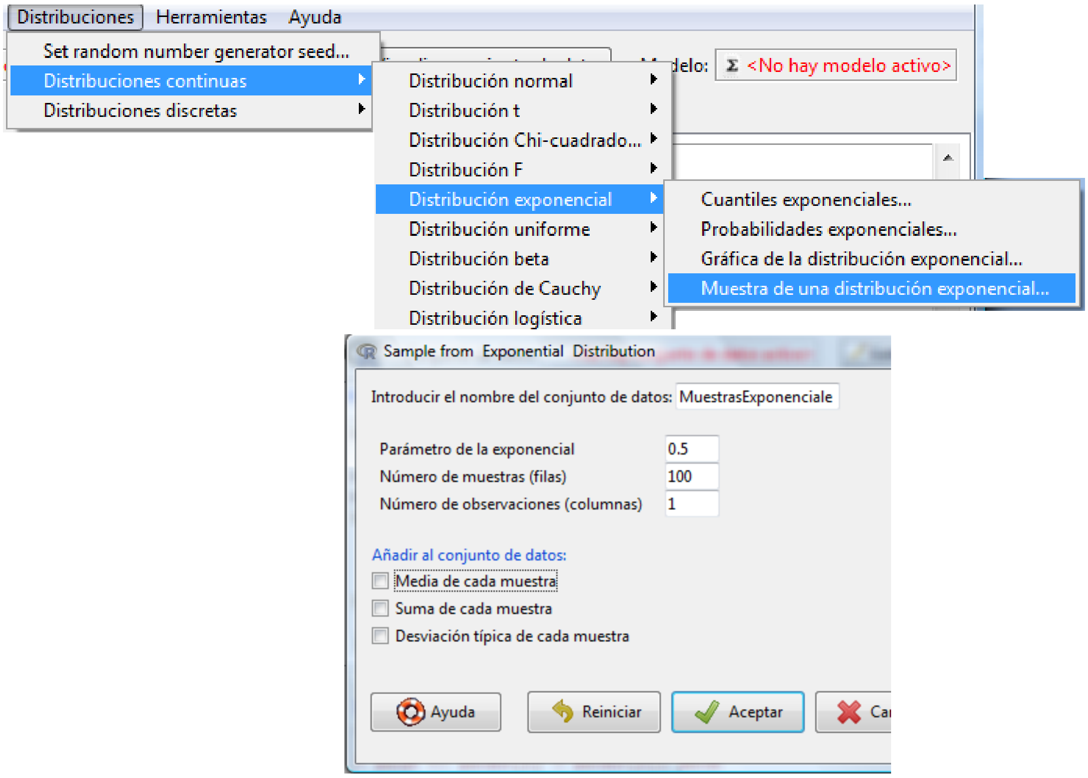
2. Simulación

Cuando un experimento es demasiado costoso, o físicamente difícil o imposible de realizar, y cuando la distribución de probabilidad de los datos que serían generados por el experimento es aproximadamente conocida, los números aleatorios generados por el ordenador a partir de la distribución apropiada se pueden utilizar en lugar de los verdaderos datos del experimento. A los datos generados de esta forma se les denomina simulados y al proceso que genera números aleatorios se le denomina **simulación**.

2.1. Simulación de muestras de variables aleatorias

En esta sección vamos a ver cómo podemos obtener muestras de variables aleatorias mediante simulación.

Ejercicio 4. Si la distribución de la v.a. en cuestión figura entre las contenidas en el menú *Distribuciones*, el proceso es muy sencillo. Veamos el procedimiento a seguir si queremos una muestra aleatoria de una distribución exponencial de parámetro 0.5.



Las instrucciones *R* son:

MuestrasExponenciales <- as.data.frame(matrix(rexp(100*1, rate=0.5), ncol=1)) *# crea el conjunto de datos MuestrasExponenciales, éste es el nombre que asigna R por defecto en la ventana anterior*
rownames(MuestrasExponenciales) <- paste("sample", 1:100, sep="") *# denomina sample i al individuo i-ésimo de la muestra*
colnames(MuestrasExponenciales) <- "obs" *# denomina obs a la muestra*

Podemos cambiar los nombres de las filas, por ejemplo si queremos numerarlas del 1 al 100 escribimos:

rownames(MuestrasExponenciales) <- 1:100

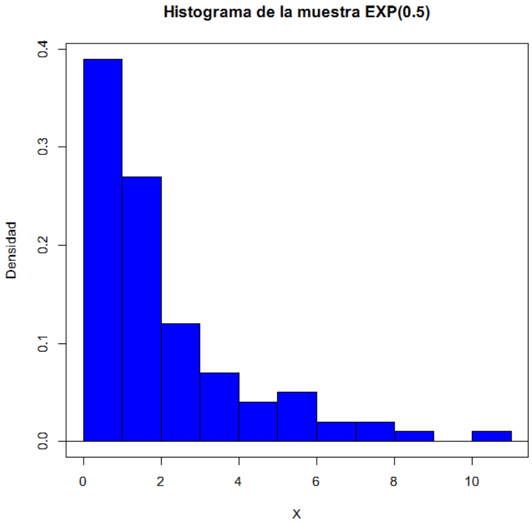
Activando el conjunto de datos *MuestrasExponenciales*, podemos obtener los resúmenes numéricos (o la distribución de frecuencias si la variables es discreta) y también las representaciones gráficas adecuadas de los datos muestrales.

Escribe en la ventana de instrucciones:

print(summary(MuestrasExponenciales))

la salida es:

mean	sd	IQR	0%	25%	50%	75%	100%	n
2.019353	2.065647	2.269049	0.001240272	0.5556427	1.36122	2.824691	10.08371	100



2.2. Simulación de experimentos aleatorios

En primer lugar vamos a presentar algunas instrucciones en *R* para realizar determinadas acciones.
sample (x, tamaño, replace=TRUE,prob=NULL) *# x = población, tamaño = número de valores simulados, replace =TRUE (FALSE) si (no) hay reemplazamiento, prob = vector de pesos a asignar a cada uno de los posibles valores que se extraen del conjunto especificado por x. Por defecto, todos los valores resultantes de x tienen la misma probabilidad.*

replicate (n, código a repetir n veces) *# devuelve un vector con n valores de código*

data.frame (A) *# hace visible desde Rcmdr el conjunto A*

set.seed (número) *# permite seleccionar la semilla para generar números pseudoaleatorios*

A continuación vamos a ver cómo podemos simular experimentos aleatorios.

Ejercicio 5. **Simula el lanzamiento de 4 dados perfectos.**
Solución: Escribe el siguiente código en la ventana de instrucciones:

```
muestra<-sample(1:6,4, replace=TRUE); muestra  # lanzamiento de cuatro dados
```

La salida es:
> muestra<-sample(1:6,4, replace=TRUE);muestra # lanzamiento de cuatro dados
[1] 5 3 4 1

Ejercicio 6. **Simula la distribución de la suma de los números que se obtienen al lanzar 4 dados**, para ello simula 10000 valores de la suma y después represéntalos gráficamente y obtén la tabla de frecuencias y porcentajes y los resúmenes numéricos.

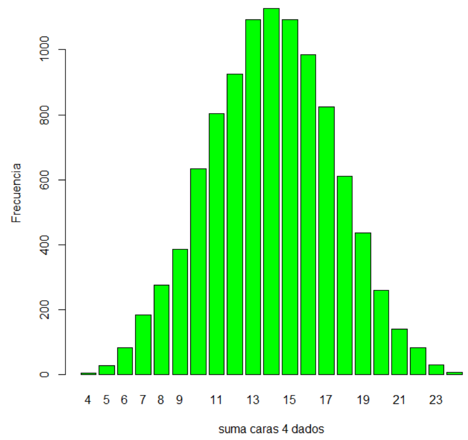
Solución: Escribe el siguiente código en la ventana de instrucciones:

```
sumas <- replicate(10000,sum(sample(1:6,4, replace = TRUE))) # devuelve vector con 10000 valores resultado de la suma de las caras obtenidas al lanzar 10000 veces cuatro dados (sample(1:6,4, replace = TRUE))
```

```
datosSuma <- data.frame(sumas)  # para poder seleccionarlo como conjunto de datos activo y así poder representarlos gráficamente (gráfico barras) y obtener los resúmenes numéricos

numSummary(datosSuma$sumas, statistics=c("mean", "sd", "IQR", "quantiles"), quantiles=c(0,.25,.5,.75,1))  #resúmenes numéricos

fsumas <- datosSuma$fsumas <- as.factor(datosSuma$sumas)  #convierte sumas en factor para dibujar el diagrama de barras

frecuencia <- table(datosSuma$fsumas); frecuencia

porcentaje <- 100*frecuencia/sum(frecuencia);porcentaje

barplot(frecuencia,xlab="suma caras 4 dados", ylab="Frecuencia", col="green")
```

Salida:

```
> mean     sd       IQR  0%  25%  50%  75%  100%   n
  13.9958  3.402079 4    4   12   14   16   24     10000

> frecuencia <- table(datosSuma$fsumas); frecuencia
 4  5  6  7  8   9  10  11   12  13   14   15   16  17  18  19  20  21  22 23 24
 4 26 83 183 274 386 634 803 926 1093 1127 1092 986 824 610 436 259 139 82 28  5

> porcentaje <- 100*frecuencia/sum(frecuencia);porcentaje
 4    5    6    7    8    9    10   11   12   13    14    15    16   17   18   19   20   21
0.04 0.26 0.83 1.83 2.74 3.86 6.34 8.03 9.26 10.93 11.27 10.92 9.86 8.24 6.10 4.36 2.59 1.39
22   23   24
0.82 0.28 0.05
```

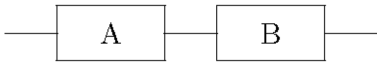


3. Fiabilidad

En esta sección vamos a estudiar la fiabilidad de sistemas utilizando la simulación para generar los tiempos de vida de las componentes del sistema.

Cuando un sistema se compone de varios aparatos conectados, el tiempo de vida del sistema depende del tiempo de vida de las diferentes componentes y de la forma en que éstas se componen. Suponiendo siempre independencia entre la duración de las distintas componentes,

- si dos componentes están conectadas en serie,

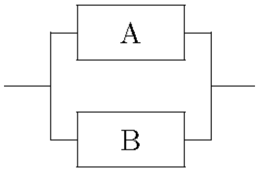


la probabilidad de que el sistema dure más de un tiempo t es la probabilidad de que cada componente dure más de ese tiempo t:

Pr(sistema > t) = Pr(min(A,B)>t)= Pr(A>t)Pr(B>t)

```
sistema<-pmin(A,B) #vector de mínimos
prob<-mean(sistema >t)
```

- si las componentes están conectadas en paralelo,

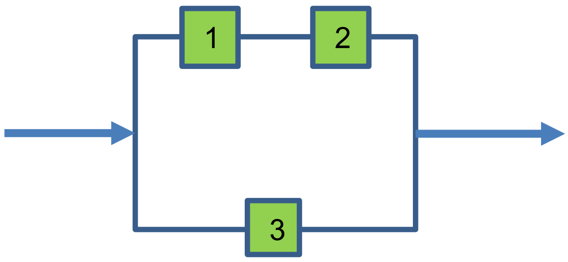


la probabilidad de que el sistema dure más de un tiempo t es la probabilidad de que alguna de las dos componentes dure más de ese tiempo t:

Pr(sistema > t) =Pr(max(A,B)>t) = 1-Pr(max(A,B)< t)=1-Pr(A < t)Pr(B < t)

```
sistema<-pmax(A,B)      #vector de máximos
prob<-mean(sistema >t)
```

Ejercicio 7. Estima la vida esperada del sistema siguiente en el que los tiempos de vida (en años) de las componentes se distribuyen de acuerdo a: vida1 ≡ exp(0’1), vida2 ≡ exp(0’15) y vida3 ≡ weibull(1,0’13). Proporciona una estimación de la probabilidad de que el sistema no falle antes de 5 años.



Solución: Escribe el código siguiente:

```
n <- 1000
vida1 <- rexp(n, rate=0.1)  # simulamos n valores del tiempo de vida de la componente 1
vida2 <- rexp(n, rate=0.15) # simulamos n valores del tiempo de vida de la componente 2
vida3 <- rweibull(n, shape=1, scale=0.13) # simulamos n valores del tiempo de vida de la componente 3
vidaS <- pmax(vida3,pmin(vida1,vida2)) # simulamos n valores del tiempo de vida del sistema

vida_media <- mean(vidaS); vida_media      #estimación puntual de la vida media del sistema
probabilidad <- mean(vidaS >= 5); probabilidad
```

Salida:

```
> vida_media <- mean(vidaS); vida_media
[1] 4.015712
> probabilidad <- mean(vidaS > 5); probabilidad
[1] 0.28913
```

4. Ejercicios propuestos

Ejercicio 1. Un examen tipo test tiene 10 preguntas y cada pregunta tiene 4 respuestas, siendo correcta sólo una de ellas. Si un estudiante no conoce la respuesta correcta de ninguna cuestión y contesta aleatoriamente, queremos saber:

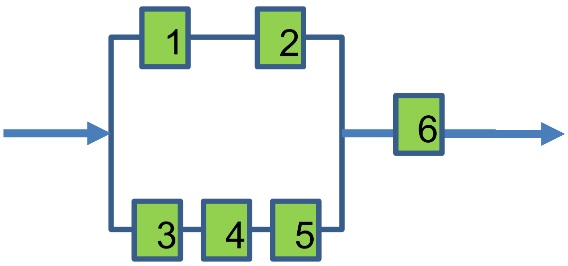
a) ¿Cuál es la probabilidad de responder correctamente a todas las preguntas?
b) ¿Cuál es la probabilidad de responder correctamente como mucho a 8 preguntas?
c) Si se aprueba el examen cuando se responden correctamente al menos 5 cuestiones, ¿cuál es la probabilidad de que pase un alumno que ha contesta aleatoriamente?
d) Si cada respuesta correcta vale 1 punto ¿a partir de que puntuación está como mucho el 2% de las notas más altas?
e) Representa gráficamente la función de probabilidad y la función de distribución acumulada de *X*.

Ejercicio 2. El número de peticiones por minuto a un servidor horario NTP sigue un proceso de Poisson de media 2.

a) ¿Cuál es la probabilidad de que en un minuto cualquiera el servidor reciba por lo menos 4 peticiones?
b) ¿Cuál es la probabilidad de que en un minuto cualquiera el servidor reciba al menos una petición?
c) ¿Cuál es la probabilidad de que en un minuto cualquiera el número de peticiones recibidas sea superior a 2 e inferior a 6?

Ejercicio 3. Sabiendo que la demanda de gasolina durante un cierto período de tiempo se comporta con arreglo a la ley normal de media 150000 litros y desviación típica 10000 litros, determinar la cantidad que hay que tener dispuesta a la venta en dicho período para poder satisfacer la demanda con una probabilidad de 0.95.

Ejercicio 4. Estima la vida esperada del sistema siguiente en el que los tiempos de vida (en años) de las componentes se distribuyen de acuerdo a: vida1 ≡ exp(1/30), vida2 ≡ weibull(0'5,14), vida3 ≡ weibull(0'25,1), vida4 ≡ exp(1/35), vida5 ≡ exp(1/32) y vida6 ≡ weibull(10,30). Estima la probabilidad de que el sistema falle antes de 3 años.



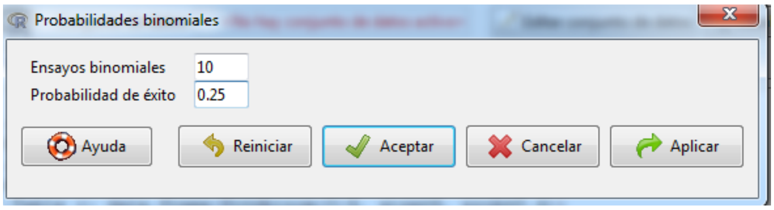
Soluciones

Ejercicio 1.

Define la v.a. X = *nº de respuestas correctas en el test*. Como sólo es correcta una de las cuatro posibles respuestas, la probabilidad de responder correctamente a una pregunta es 0’25. Por tanto la v.a. X sigue una distribución binomial, $X \equiv$ binom(10, 0’25).

a) *¿Cuál es la probabilidad de responder correctamente a todas las preguntas?* Nos piden Pr(X = 10). Para calcular esta probabilidad sigue la secuencia:

Distribuciones → Distribuciones discretas→ Distribución binomial → Probabilidades binomiales
y rellena la ventana emergente, como sigue:



En la ventana de resultados obtienes:

```
> .Table <- data.frame(Pr=dbinom(0:10, size=10, prob=0.25))
> rownames(.Table) <- 0:10
> .Table
          Pr
0         5.631351e-02
1         1.877117e-01
2         2.815676e-01
3         2.502823e-01
4         1.459980e-01
5         5.839920e-02
6         1.622200e-02
7         3.089905e-03
8         3.862381e-04
9         2.861023e-05
10        9.536743e-07
```

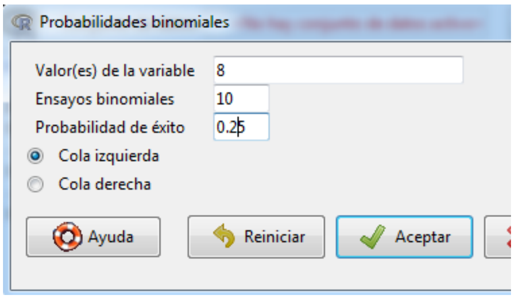
Pr(X = 10)

```
> remove(.Table)
```

- También puedes obtener este resultado escribiendo en la ventana de instrucciones:

  dbinom(10,size=10,prob=0.25) o dbinom(10,10,0.25)

b) *¿Cuál es la probabilidad de responder correctamente como mucho a 8 preguntas?* Nos piden Pr($X \leq 8$). Para calcular esta probabilidad sigue la secuencia:
Distribuciones → Distribuciones discretas→ Distribución binomial → Probabilidades binomiales acumuladas
y rellena la ventana emergente, como sigue:

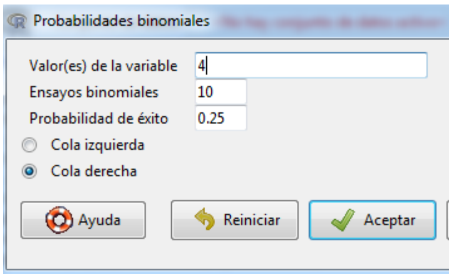


En la ventana de resultados obtienes:

> pbinom(c(8), size=10, prob=0.25, lower.tail=TRUE)

[1] 0.9999704

c) *Si se aprueba el examen cuando se responden correctamente al menos 5 cuestiones, ¿cuál es la probabilidad de que pase un alumno que ha probado suerte?* Nos piden $\Pr(X \geq 5)$, pero esta probabilidad es igual a $\Pr(X > 4)$. Para calcular esta probabilidad sigue la secuencia:
Distribuciones → Distribuciones discretas→ Distribución binomial → Probabilidades binomiales acumuladas
y rellena la ventana emergente, como sigue:



En la ventana de resultados obtienes:

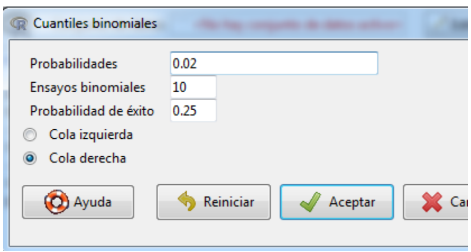
> pbinom(c(4), size=10, prob=0.25, lower.tail=FALSE)

[1] 0.07812691

d) *Si cada respuesta correcta vale 1 punto ¿a partir de que puntuación está como mucho el 2% de las notas más altas?* Nos piden la puntuación *c* que deja por encima el 2% de las notas más altas, $\Pr(X > c) \leq 0'02$ (cola superior o derecha). Para calcular esta puntuación sigue la secuencia:

Distribuciones → Distribuciones discretas→ Distribución binomial → cuantiles

y rellena la ventana emergente, como sigue:

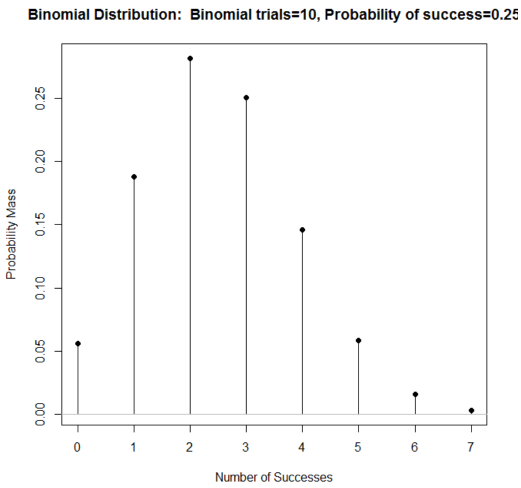


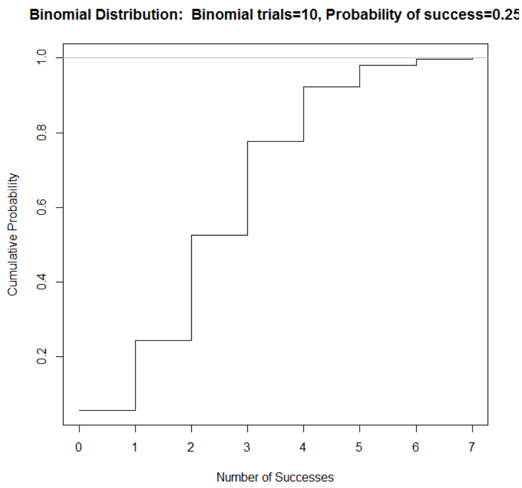
En la ventana de resultados obtienes:

> qbinom(c(0.02), size=10, prob=0.25, lower.tail=FALSE)

[1] 5 $\Rightarrow$ *c = 6 ya que el 6 es el primer valor de X > 5, se puede comprobar que* Pr(X > 5) = Pr($X \geq 6$) = 0'01972771 ≤ 0'02 y esta es la primera puntuación que verifica la condición del enunciado

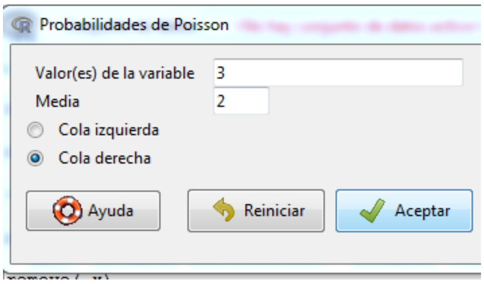
e) *Representa gráficamente la función de probabilidad y la función de distribución acumulada de X.* sigue la secuencia:
Distribuciones → Distribuciones discretas→ Distribución binomial → Gráfica de la distribución binomial y rellena la ventana emergente con los valores de n y p, y selecciona la gráfica que te interesa obtener.





Ejercicio 2. Define la v.a. *X = nº de peticiones por minuto a un servidor horario NTP*. Sabemos que *X* sigue una distribución de Poisson de media 2, es decir $X \equiv$ pois(2).

a) *¿Cuál es la probabilidad de que en un minuto cualquiera el servidor reciba por lo menos 4 peticiones?* Nos piden Pr($X \geq 4$). Para calcular esta probabilidad sigue la secuencia:
Distribuciones → Distribuciones discretas→ Distribución Poisson → Probabilidades de Poisson acumuladas
Como la cola superior nos proporciona Pr($X > k$) debes tomar k = 3 y rellenar la ventana emergente como sigue:



En la ventana de resultados obtienes:

> ppois(c(3), lambda=2, lower.tail=FALSE)

[1] 0.1428765

b) *¿Cuál es la probabilidad de que en un minuto cualquiera el servidor* reciba al menos una petición*?* Nos piden Pr($X \geq 1$) = Pr($X > 0$). Para calcular esta probabilidad sigue la secuencia:
Distribuciones → Distribuciones discretas→ Distribución Poisson → Probabilidades de Poisson acumuladas
y rellena la ventana emergente tomando como *valor(es) de la variable* : 0, *Media*:2 , marca *Cola derecha*. En la ventana de resultados obtienes:

> ppois(c(0), lambda=2, lower.tail=FALSE)
[1] 0.8646647

Otra forma de hacerlo: Pr($X \geq 1$) = 1 - Pr($X = 0$). En este caso la secuencia es:
Distribuciones → Distribuciones discretas→ Distribución Poisson → Probabilidades de Poisson
Rellena la ventana emergente con Media = 2. El resultado es:

> .Table <- data.frame(Pr=dpois(0:8 , lambda = 2))
> rownames(.Table) <- 0:8
> .Table

	Pr	
0	**0.1353352832**	Pr($X = 0$)
1	0.2706705665	
2	0.2706705665	
3	0.1804470443	
4	0.0902235222	
5	0.0360894089	
6	0.0120298030	
7	0.0034370866	
8	0.0008592716	

> remove(.Table)

Entonces Pr(X ≥ 1) = 1 - Pr(X = 0)

> 1- 0.1353352832
[1] 0.8646647

- También puedes obtener este resultado escribiendo en la ventana de instrucciones:
  1 - dpois(0,lambda=2) o 1 - dpois(0,2)

c) *¿Cuál es la probabilidad de que en un minuto cualquiera el número de peticiones recibidas sea superior a 2 e inferior a 6?* Nos piden Pr($2 < X < 6$) = Pr($X < 6$) – Pr($X \leq 2$) = Pr($X \leq 5$) – Pr($X \leq 2$).
Escribe en la ventana de instrucciones:

probs <- ppois(c(5, 2), lambda=5, lower.tail=TRUE)

prob <- probs[1] - probs[2]; prob

[1] 0.30676

Ejercicio 3. Sabiendo que la demanda de gasolina durante un cierto período de tiempo se comporta con arreglo a la ley normal de media 150.000 litros y desviación típica 10.000 litros, determinar la cantidad que hay que tener dispuesta a la venta en dicho período para poder satisfacer la demanda con una probabilidad de 0'95.

Sea X = *demanda de gasolina durante un cierto período de tiempo*, $X \equiv$ norm(150000,10000). Nos piden el valor c tal $\Pr(X \leq c) = 0'95$.
> qnorm(c(0.95), mean=150000, sd=10000, lower.tail=TRUE)

[1] 166448.5

Ejercicio 4. Escribe en la ventana de instrucciones el código siguiente:

```
n <- 100000
vida1 <- rexp(n, rate=1/30)
vida2 <- rweibull(n, shape=0.5, scale=14)
vida3 <- rweibull(n, shape=0.25, scale=1)
vida4 <- rexp(n, rate=1/35)
vida5 <- rexp(n, rate=1/32)
vida6 <- rweibull(n, shape=10, scale=30)
vidaS <- pmin(vida6, pmax(pmin(vida1,vida2), pmin (vida3,vida4,vida5)))
vida_media <- mean(vidaS); vida_media
probabilidad <- mean(vidaS < 3); probabilidad
```

Salida:

```
> vida_media <- mean(vidaS); vida_media
[1] 9.69932
> probabilidad <- mean(vidaS < 3); probabilidad
[1] 0.33371
```