Tarea 1

Jorge Rotter

30 de agosto de 2018

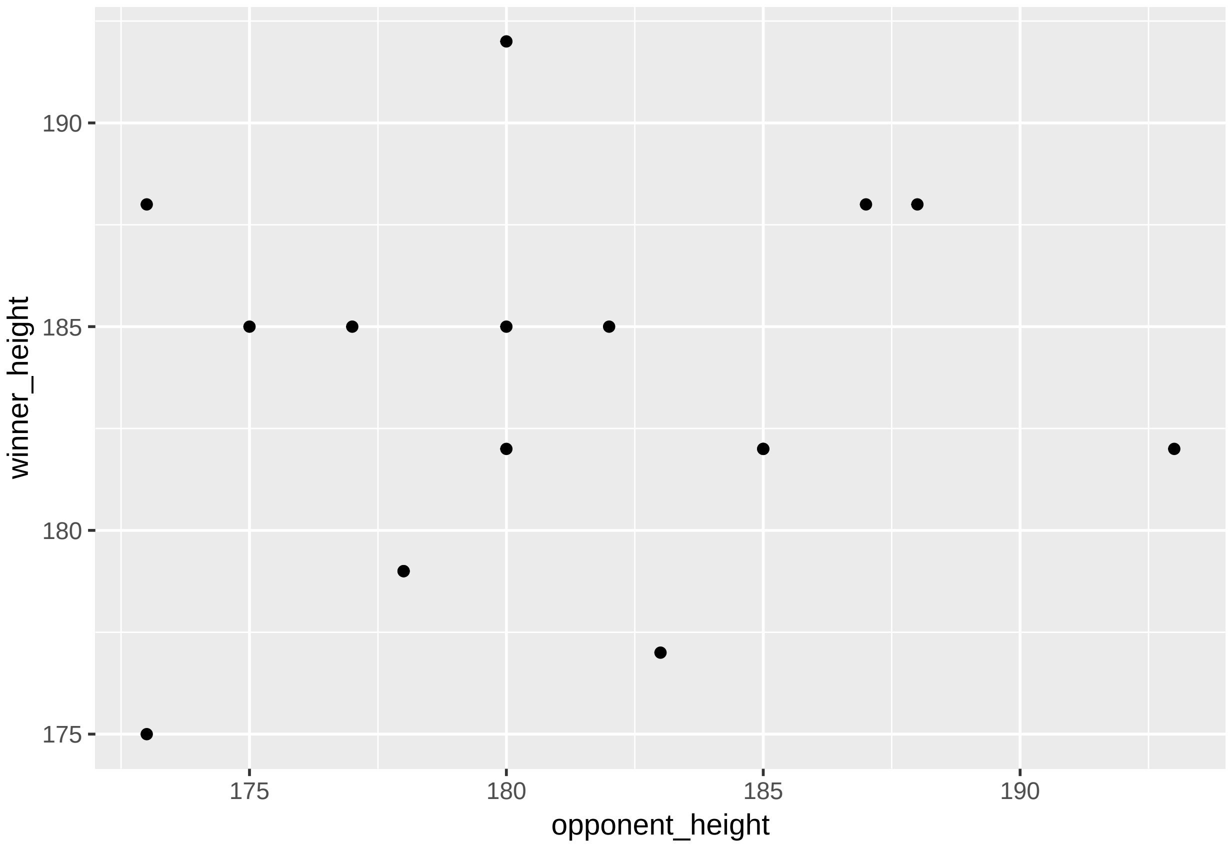
1. Sea X el número de 'unos' obtenido en doce lanzamientos de un dado honesto. Entonces, X tiene una distribución binomial. Calcule una tabla con los valores de lal función de distribución para $x \in \{0, \cdots, 12\}$ por dos métodos: usando la función `cumsum` y la función `pbinom`. Calcule también $\mathbb{P}\{X > 7\}$.

```
# Sólo por claridad, uso los nombres convencionales de los parámetros
n <- 12
p <- 1/6


# Calcular tabla
probabilities <- data_frame(x=0:12,
           using_cumsum = cumsum(choose(n,x)*p^x*(1-p)^(n-x)),
           using_pbinom = pbinom(x,n,p))

# Calcular P{X>7}
1 - pbinom(7,n,p)
```

```
## [1] 0.0001555443
```

2. En un artículo de Wikipedia se reportan las estaturas de lso presidentes de los Estados Unidos y la de sus oponentes en elecciones. Se ha notado que típicamente, el candidato más alto gana la elección. Haga una gráfica de dispersión de puntos con la estatura del ganador contra la del perdedor.

```
read_csv('./data/us_presidents_heights.csv',col_types = cols()) %>%
  ggplot(aes(x=opponent_height, y=winner_height)) +
  geom_point()
```



3. La función `rpois` genera observaciones aleatorias de una distribución Poisson. Úsela para simular un número grande ($n = 1000$ y $n = 10000$) de muestras Poisson con parámetro $\lambda = 0.61$. Encuentre la función de masa de probabilidad, media y varianza para la muestra, y compare con los valores teóricos.

```
x <- rpois(10000, 0.61)
# media
mean(x)
```

```
## [1] 0.6064
```

```
# varianza
var(x)
```

```
## [1] 0.6153406
```

```
# masa de probabilidad
data_frame(x) %>%
  group_by(x) %>%
  summarise(p_muestra = n()/1000) %>%
  mutate(p_teorica = 0.61^x*exp(-0.61)/factorial(x))
```

```
## # A tibble: 6 x 3
##       x p_muestra p_teorica
##   <int>     <dbl>     <dbl>
## 1     0     5.49   0.543
## 2     1     3.25   0.331
## 3     2     1.02   0.101
## 4     3     0.201  0.0206
## 5     4     0.036  0.00313
## 6     5     0.004  0.000382
```

La media fue $\bar{x} = 0.623$, un poco arriba del 0.61 teórico. La varianza (con la fórmula de R, que es el insesgado, no el máximo verosimil) es 0.605, un poco abajo del teórico pero en menor medida. Usando, el máximo verosimil, la estimación sería aún menor.

Con $n = 10,000$, la media muestral es 0.622 y la varianza 0.606, no mejoran demasiado.

4. Escriba una función en R llamada `sd_n` que regres el valor estimado de $\hat{sigma}$ de una muestra de tamaño n utilizando la fórmula máximo verosimil.

```
sd_n <- function(x){
  sqrt(sum((x-mean(x))^2)/length(x))
}
```

5. Escriba una función `norma` que calcule la norma euclideana de un vector numérico. Evalúe la norma de los vectores $(0,0,0,1)$, $(2,5,2,4)$ y $(1,2,3,4,5,6,7,8,9,10)$.

```
norma <- function(x){
  sqrt(sum(x^2))
}

norma(c(0,0,0,1))
```

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

```
norma(c(2,5,2,4))
```

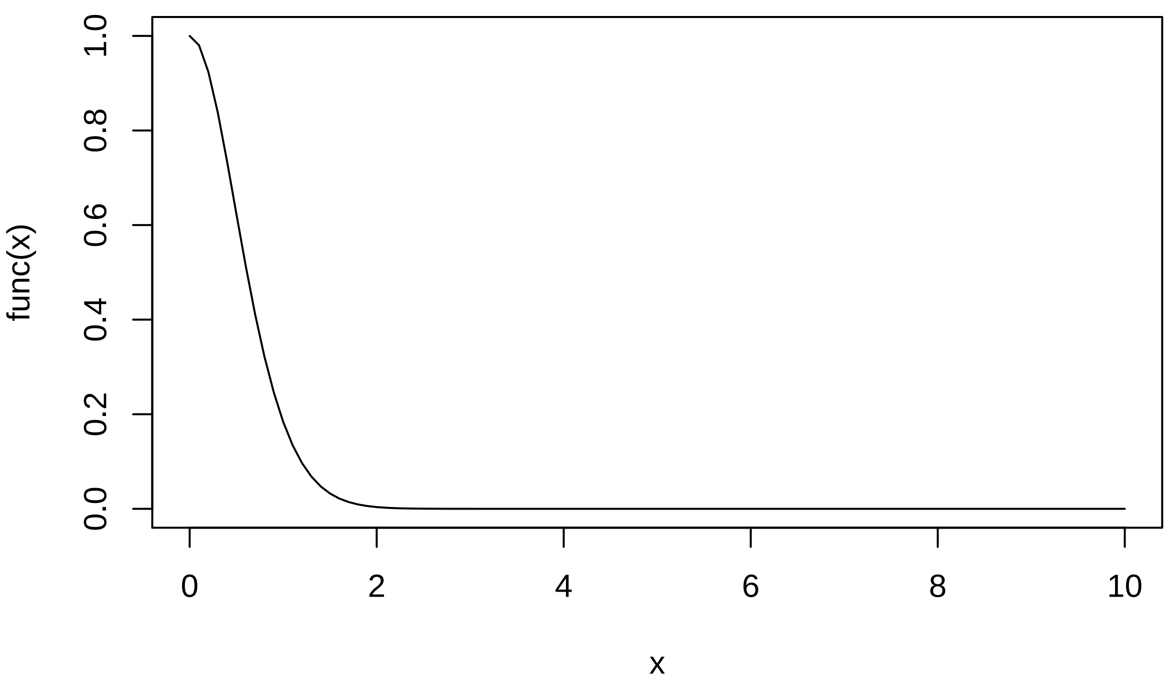
```
## [1] 7
```

```
norma(1:10)
```

```
## [1] 19.62142
```

6. Use la función `curve` para graficar la función $f(x) = e^{-x^2}/(1+x^2)$ en el intervalo $0 \leq x \leq 10$. Use la función `ingtegrate` para integrarla en $[0, \infty)$.

```
func <- function(x) exp(-x^2)/(1+x^2)
curve(expr = func,from = 0,to = 10)
```



```r
integrate(func, 0, Inf)
```

```
## 0.6716467 with absolute error < 8.3e-05
```

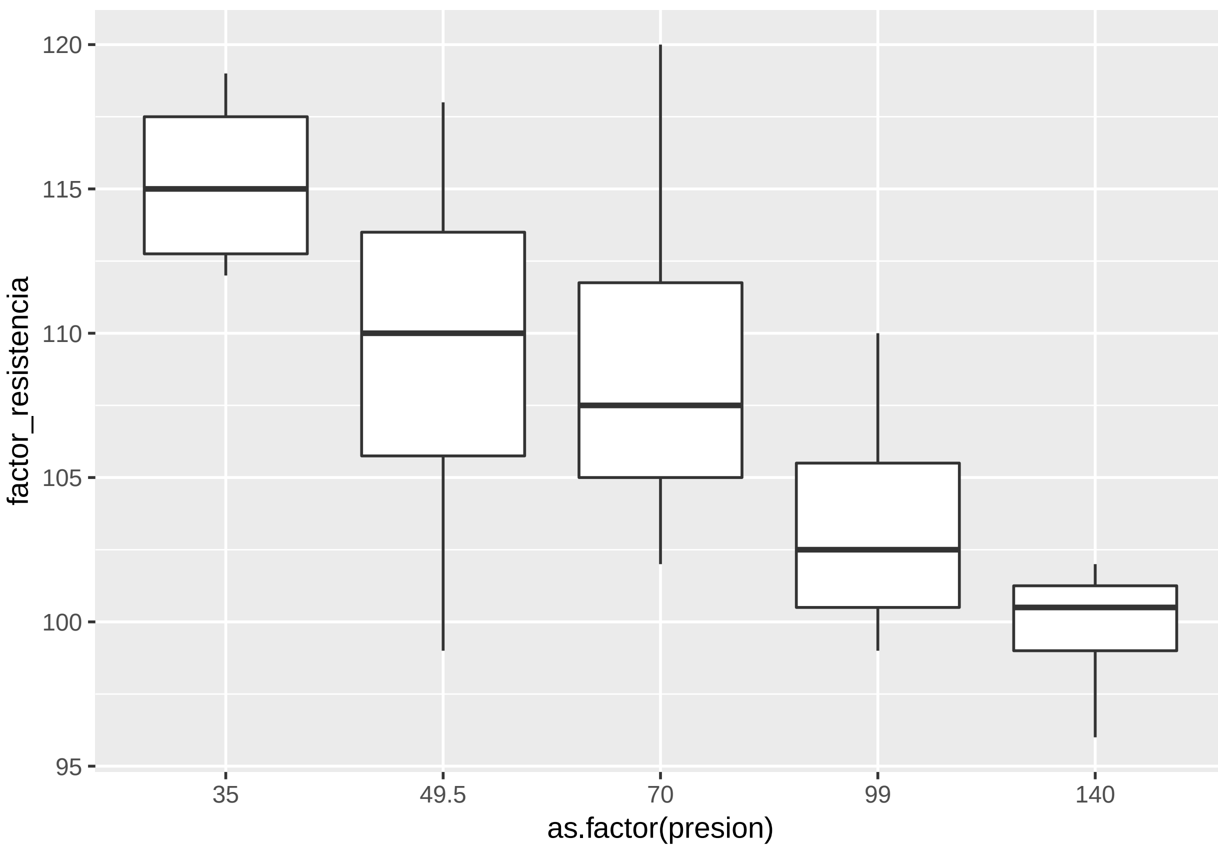
7. Construya una muestra de 10 observaciones de una normal bivariada. Use la función `apply` y la función `norma` del ejercicio 5 para calcular la norma de cada observación.

```r
x <- matrix(rnorm(20), 10, 2)
apply(x, 1, norma)
```

```
##  [1] 1.0556663 2.3776384 0.5097332 1.2289221 0.3110820 0.7539694 1.5104855
##  [8] 1.6566984 0.9153918 1.4567038
```

8. Haga un boxplot de los siguientes datos para compaar los factores de resistencia en cada presión.

```r
data_frame(presion = rep(c(35,49.5,70,99,140),4),
           factor_resistencia = c(112,108,120,110,100,
                                  119,99,106,101,102,
                                  117,112,102,99,96,
                                  113,118,109,104,101)) %>%
  ggplot(aes(x=as.factor(presion), y=factor_resistencia)) +
  geom_boxplot()
```



9. Este ejercicio da continuidad a la cola M/M/1 que programamos antes.

a. Modifique su código para incluir las siguientes medidas:

- El tiempo total promedio de los n clientes en el sistema
- La longitud máxima de la cola
- La máxima espera en la cola

b. Ejecute el modelo 100 veces para $\lambda_A = 5, \lambda_S = 4, n = 1000$ y haga un histograma para cada una de las medidas de desempeño. Calculee estadísticas descriptivas.

```
resultados <- replicate(100, mm1(5,4,1000), simplify=FALSE) %>%
  bind_rows()

resultados %>%
  gather() %>%
  ggplot(aes(value)) +
  geom_histogram(bins=15)+
  facet_wrap(~key, scales='free_x')
```