Dados, Elecciones y otras Incertidumbres

Rogelio Ramos Quiroga

rramosq@cimat.mx

TALLER DE CIENCIA PARA JÓVENES
28 de Julio de 2015

Problemas con incertidumbre

- ¿Cómo saber si un dado está bien balanceado?
- Una imágen dice más que mil palabras
- El problema de Monty Hall
- ¿Cómo saber si una medicina funciona?
- El problema de los cumpleaños
- ¿Cómo saber quién va a ganar las elecciones?

Monedas y Dados

¿Cómo saber si una moneda está bien balanceada?

- ¿*A S A S A S A S A S*?
- ¿*A A A A A S S S S S*?

¿Cómo saber si una moneda está bien balanceada?

Nuestro sentido común nos dice que está bien balanceada si:

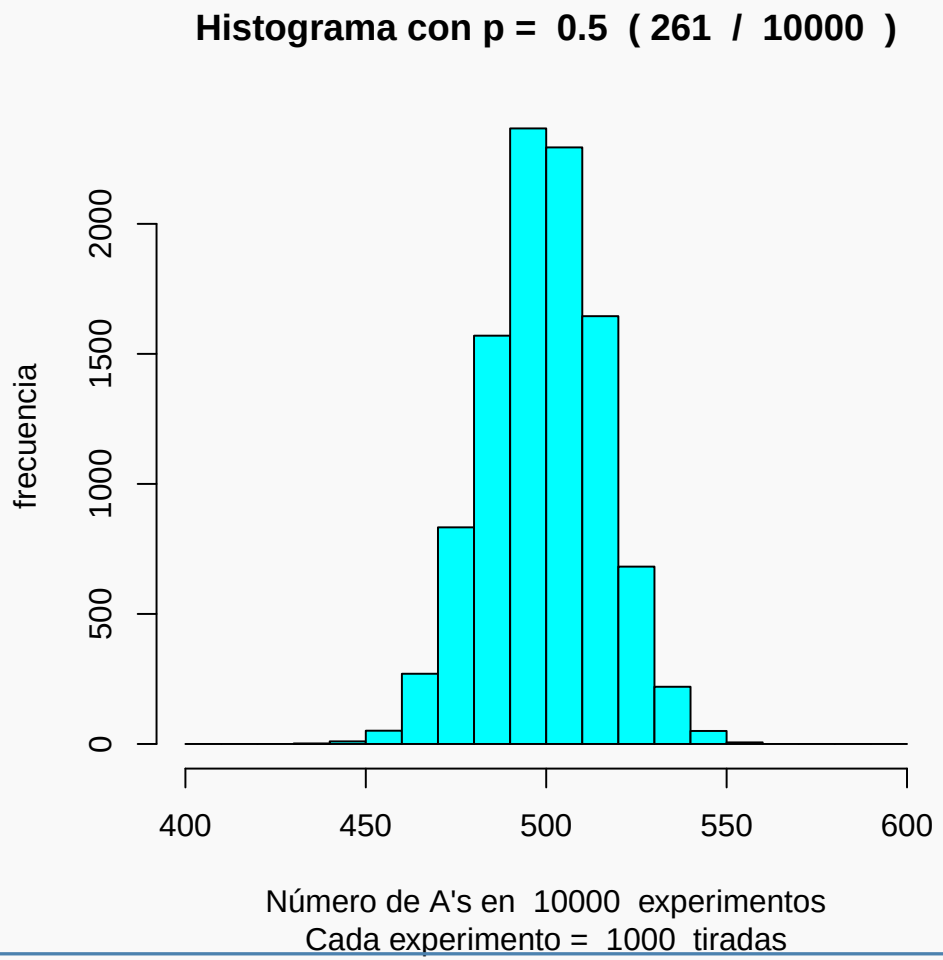
- Más o menos el 50% de las veces cae en A y más o menos en el 50% cae en S
- $\cdots$ y en orden "aleatorio"

en otras palabras:

* La probabilidad de A es 0.5.
* Las A's y S's ocurren en forma independiente

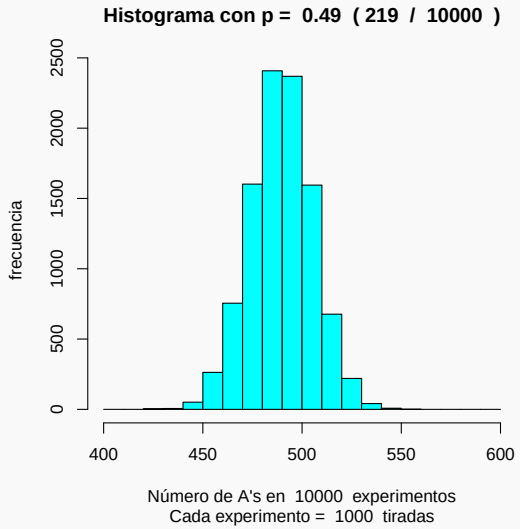
Simulación

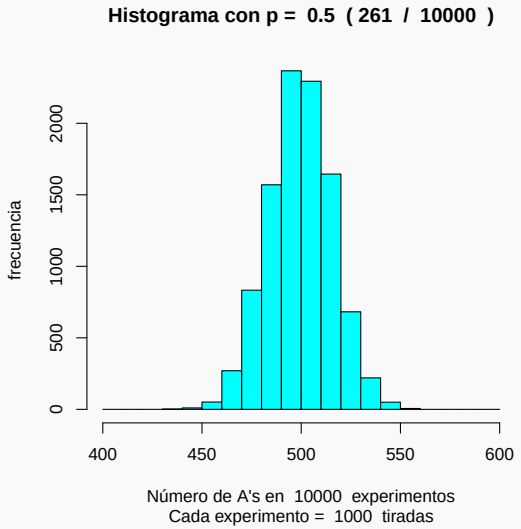
Es sumamente difícil probar empíricamente que una moneda está bien balanceada. Nos conformaremos con examinar de cerca el comportamiento de una moneda **bien** balanceada.

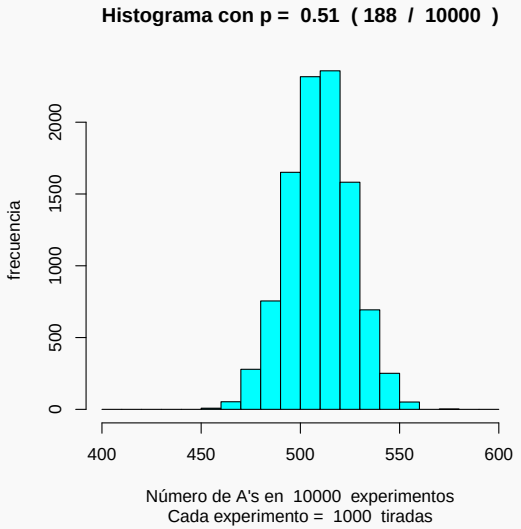


Experimento 1

¿Cómo se ven experimentos con monedas ligeramente sesgadas?



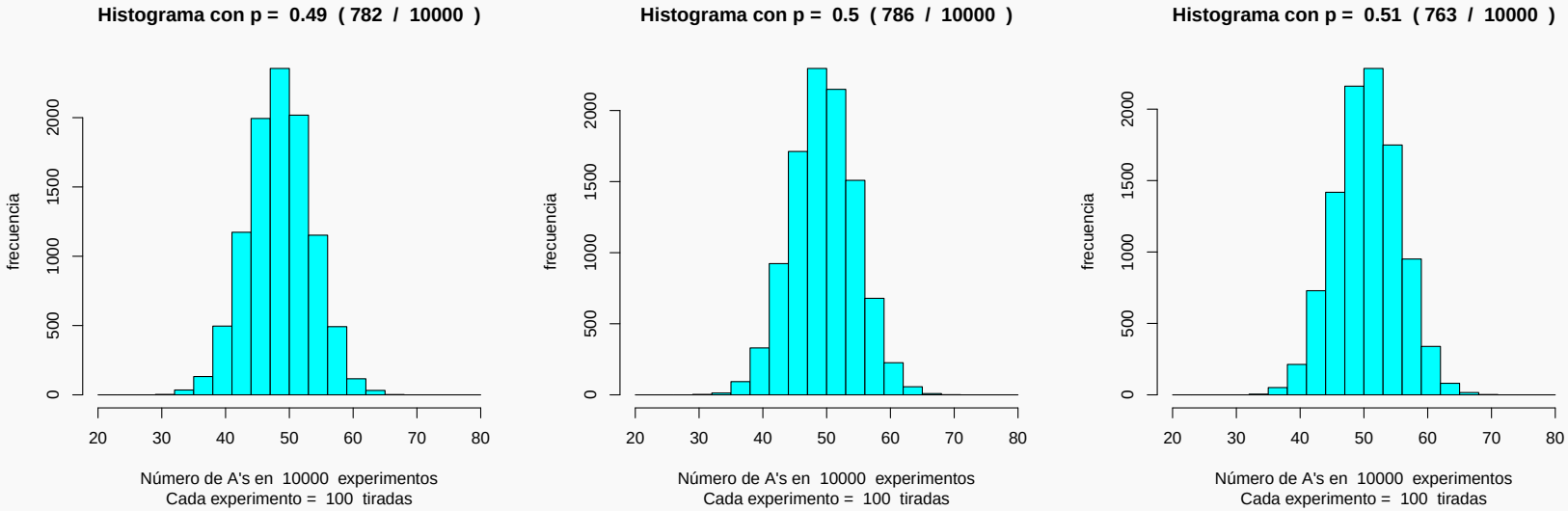




$\cdots$ pues si se ven diferencias $\cdots$ (pero estamos lanzando una moneda 10 millones de veces!)

Experimento 2

¿Qué pasa si hacemos experimentos con sólo 100 tiradas?

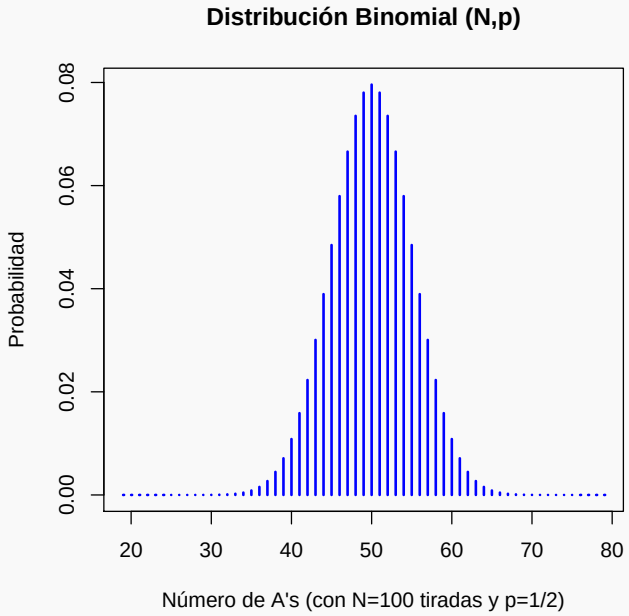


es extremadamente complicado decidir si hay desviaciones $\cdots$ (aquí la estamos lanzando "sólo" un millón de veces).

Moraleja: Para propósitos prácticos, no hay que preocuparse por monedas ligeramente desbalanceadas (sobre todo porque típicamente las usamos en un número pequeño de veces) (por supuesto, los casinos o apostadores profesionales, a la larga, si encuentran beneficios!).

Prueba de Hipótesis

Ya estamos de acuerdo en que es complicado, pero $\cdots$ ¿Cómo le hacemos?

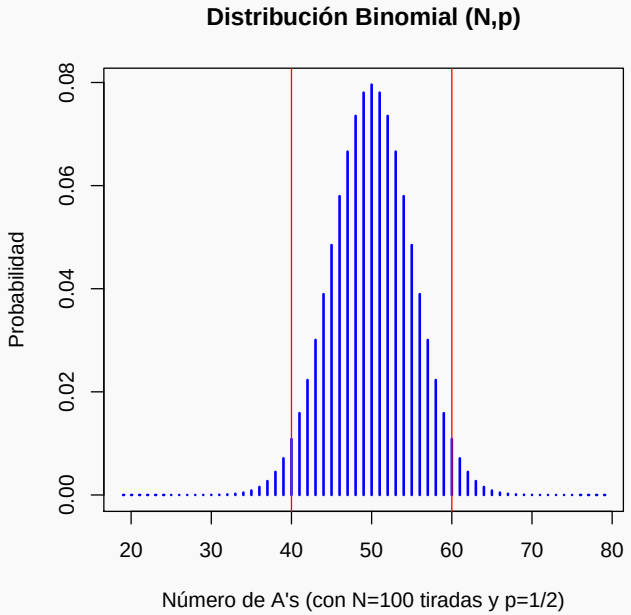


Esta figura muestra la distribución de probabilidad teórica del número de A's en $N = 100$ tiradas.

$$P(X = k) = \left(\begin{array}{c} N \\ k \end{array} \right) p^k (1-p)^{N-k}$$

Prueba de Hipótesis

Entonces, la idea es decidir que la moneda **no** está bien balanceda si el número observado de A's está en una región improbable.



Puede verse que $P(X \leq 40 \text{ ó } X \geq 60) = 0.057$, de aquí se obtiene la región de "rechazo" de la hipótesis $H_0 : p = 0.5$.

¿y, qué pasa con los dados?

- ¿Cuándo diríamos que un dado no está bien balanceado?
- Por lo que discutimos: Cuando sus resultados sean improbables.

Simulación de tiradas de un dado balanceado

Aquí presentamos 10 simulaciones de 600 tiradas de un dado. Lo esperado es que cada lado aparezca 100 veces. ¿Cómo cuantificar la discrepancia aleatoria natural y la discrepancia sistemática si el dado fuera desbalanceado?

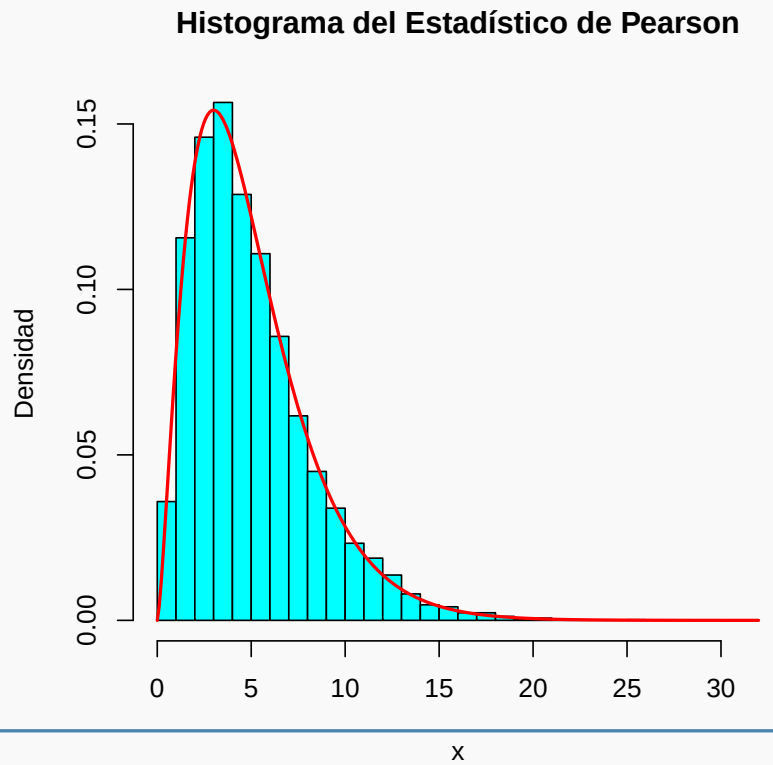
```
      [,1] [,2] [,3] [,4] [,5] [,6]
 [1,]   98   89  106  106   94  107
 [2,]  112   90   89  114   90  105
 [3,]   95  105   95   99  104  102
 [4,]  103   91   99  108   97  102
 [5,]  119  104   93  100   92   92
 [6,]  102   99  102  103   97   97
 [7,]  102  114   98   80  107   99
 [8,]  107   86  114  101  102   90
 [9,]  105  103   95  113   91   93
[10,]  105  111   97  106   94   87
```

Estadístico Ji-cuadrada de Pearson

Hace más de un siglo, Karl Pearson propuso el estadístico

$$\chi^2 = \sum_{j=1}^{K} \frac{(o_j - e_j)^2}{e_j}$$

para medir discrepancias entre observaciones y sus valores esperados.



Estadístico Ji-cuadrada de Pearson

El primer renglón son resultados de 600 tiradas de un dado

1	2	3	4	5	6
91	118	90	96	109	96
95	134	86	105	93	87

y el segundo son tiradas de uno desbalanceado que tiene probabilidades 0.154 0.231 0.154 0.154 0.154 0.154. En el primer caso:

$$\chi^2 = \sum_{j=1}^{K} \frac{(o_j - e_j)^2}{e_j} = 6.18$$

con un p-valor de 0.289, y, en el segundo

$$\chi^2 = \sum_{j=1}^{K} \frac{(o_j - e_j)^2}{e_j} = 16.2$$

con un correspondiente p-valor de 0.006 (estos p-valores son calculados bajo la distribución límite χ^2_5). ¿Cómo se interpreta esto?.

Apéndice: Monedas y Dados, R.1

```
# Cómo saber si una moneda está bien balanceada?
# Es muy difícil... Pero podemos simular el
# comportamiento y ver que pasa en un número grande de tiradas

N   = 1000
M   = 10000
p   = .50
res = rep(0,M)
for(i in 1:M){
  x  = sample(c("A","S"),size=N,replace=T,prob=c(p,1-p))
  x  = factor(x,levels=c("A","S"))
  a  = table(x)
  res[i] = a[1] }
n500 = sum(res==500)
hist(res,col="cyan",
 xlab=paste("Número de A's en ",M," experimentos"),
 ylab="frecuencia", breaks=seq(400,600,by=10),
 main=paste("Histograma con p = ",p," (",n500," / ",M ," )"),
 sub=paste("Cada experimento = ",N," tiradas") )
```

Apéndice: Monedas y Dados, R.2

```
# Probabilidades Binomiales

N  = 100
x  = 0:N
p  = 0.5
pb = dbinom(x,size=N,prob=p)
rr = 20:80
plot(x[rr],pb[rr],type="h",lwd=2,col="blue",
  xlab="Número de A's (con N=100 tiradas y p=1/2)",ylab="Probabilidad",
  main="Distribución Binomial (N,p)")
abline(v=c(40,60),col="red")

pbinom(40,size=N,prob=p) + 1-pbinom(59,size=N,prob=p) # 0.05688793
```

Apéndice: Monedas y Dados, R.3

```
# Prueba ji-cuadrada de Pearson:
# Mide la discrpancia entre observados y esperados

N   = 600
M   = 10000
p   = 1/6
obs = matrix(0,M,6)
for(i in 1:M){
  x  = sample(1:6,size=N,replace=T,prob=rep(1,6)/6)
  x  = factor(x,levels=1:6)
  a  = table(x)
  obs[i,] = a}
jicuad = rep(0,M)
esp    = N*p*rep(1,6)
for(i in 1:M){ jicuad[i] = sum(((obs[i,]-esp)^2)/esp) }

hist(jicuad,col="cyan",freq=F,breaks=seq(0,32,by=1),
 main="Histograma del Estadístico de Pearson",
 xlab="x",ylab="Densidad",ylim=c(0,0.155))
xx = seq(0,32,length=300)
yy = dchisq(xx,df=5)           # Densidad ji-cuadrada con 5 g.l.
lines(xx,yy,lwd=2,col="red")
```

Apéndice: Monedas y Dados, R.4

```
# Este es un problema que Samuel Pepys le planteó a Issac Newton:
# Qué es mas probable:
# a) obtener al menos un   seis   en  6 tiradas
# b) obtener al menos dos  seises en 12 tiradas
# c) obtener al menos tres seises en 18 tiradas ?

M <- 100000
A <- matrix(sample(1:6,size=M*6,replace=T),ncol=6)
A <- ifelse(A==6,1,0)
sum(rowSums(A)>0)/M     # 0.664882

B <- matrix(sample(1:6,size=M*12,replace=T),ncol=12)
B <- ifelse(B==6,1,0)
sum(rowSums(B)>1)/M     # 0.618965

C <- matrix(sample(1:6,size=M*18,replace=T),ncol=18)
C <- ifelse(C==6,1,0)
sum(rowSums(C)>2)/M     # 0.597351

# Cálculos exactos:
1 - pbinom(0,size=6,p=1/6)  # 0.665102
1 - pbinom(1,size=12,p=1/6) # 0.6186674
1 - pbinom(2,size=18,p=1/6) # 0.5973457
```

Gráficas

Una imágen dice más que mil palabras

Datos de Anscombe

Los siguientes son cuatro conjuntos de datos. Deseamos hacer los correspondientes análisis de regresión.

x_1	y_1	x_2	y_2	x_3	y_3	x_4	y_4
10	8.04	10	9.14	10	7.46	8	6.58
8	6.95	8	8.14	8	6.77	8	5.76
13	7.58	13	8.74	13	12.74	8	7.71
9	8.81	9	8.77	9	7.11	8	8.84
11	8.33	11	9.26	11	7.81	8	8.47
14	9.96	14	8.10	14	8.84	8	7.04
6	7.24	6	6.13	6	6.08	8	5.25
4	4.26	4	3.10	4	5.39	19	12.50
12	10.84	12	9.13	12	8.15	8	5.56
7	4.82	7	7.26	7	6.42	8	7.91
5	5.68	5	4.74	5	5.73	8	6.89

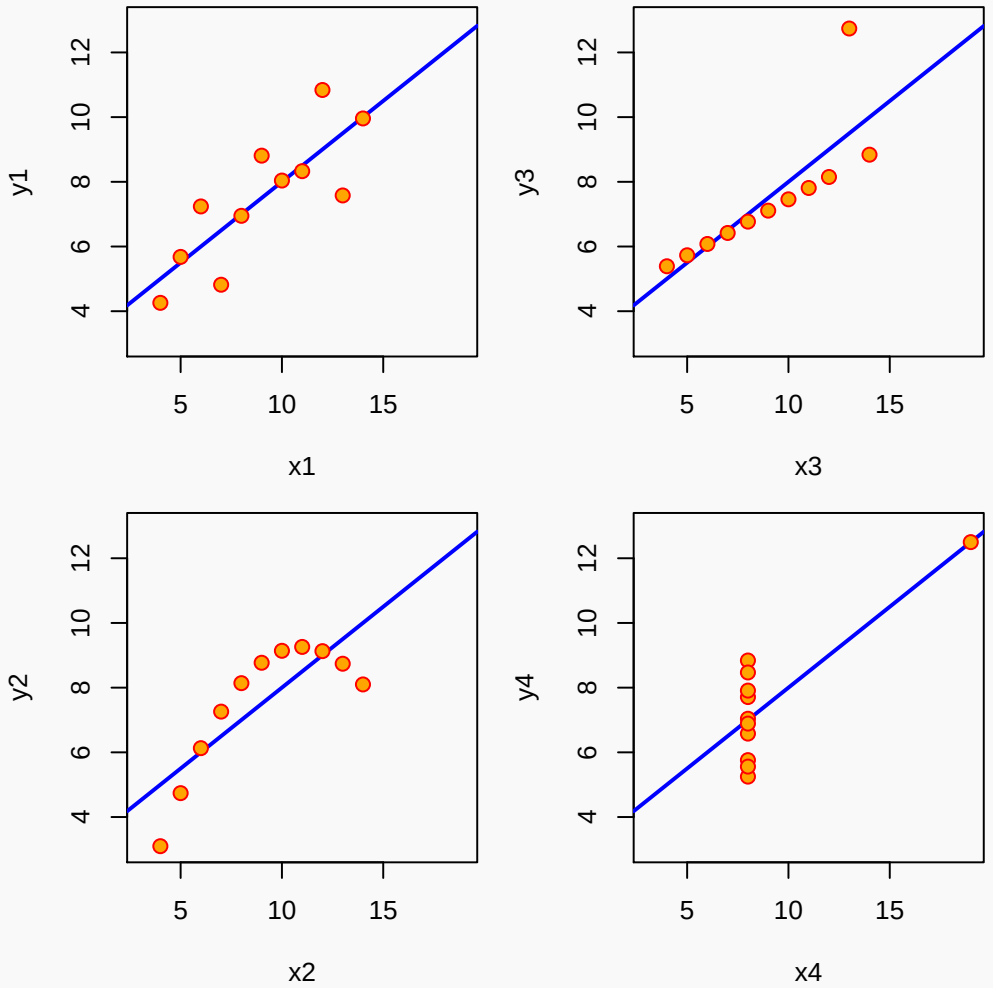
Es fácil ver que todas las x's tienen las mismas medias y desviaciones estándar y que pasa lo mismo con las y's: tienen también mismas medias y desviaciones estándar.

Datos de Anscombe: Análisis de Regresión

```
y1 vs x1 : Multiple R-squared:  0.6665
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.0001     1.1247   2.667  0.02573 *
x1            0.5001     0.1179   4.241  0.00217 **
Residual standard error: 1.237 on 9 degrees of freedom

y2 vs x2 : Multiple R-squared:  0.6662
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    3.001      1.125   2.667  0.02576 *
x2             0.500      0.118   4.239  0.00218 **
Residual standard error: 1.237 on 9 degrees of freedom

y3 vs x3 : Multiple R-squared:  0.6663
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.0025     1.1245   2.670  0.02562 *
x3            0.4997     0.1179   4.239  0.00218 **
Residual standard error: 1.236 on 9 degrees of freedom

y4 vs x4 : Multiple R-squared:  0.6667
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.0017     1.1239   2.671  0.02559 *
x4            0.4999     0.1178   4.243  0.00216 **
Residual standard error: 1.236 on 9 degrees of freedom
```

Conclusión: Las regresiones son iguales

Pero si son diferentes!



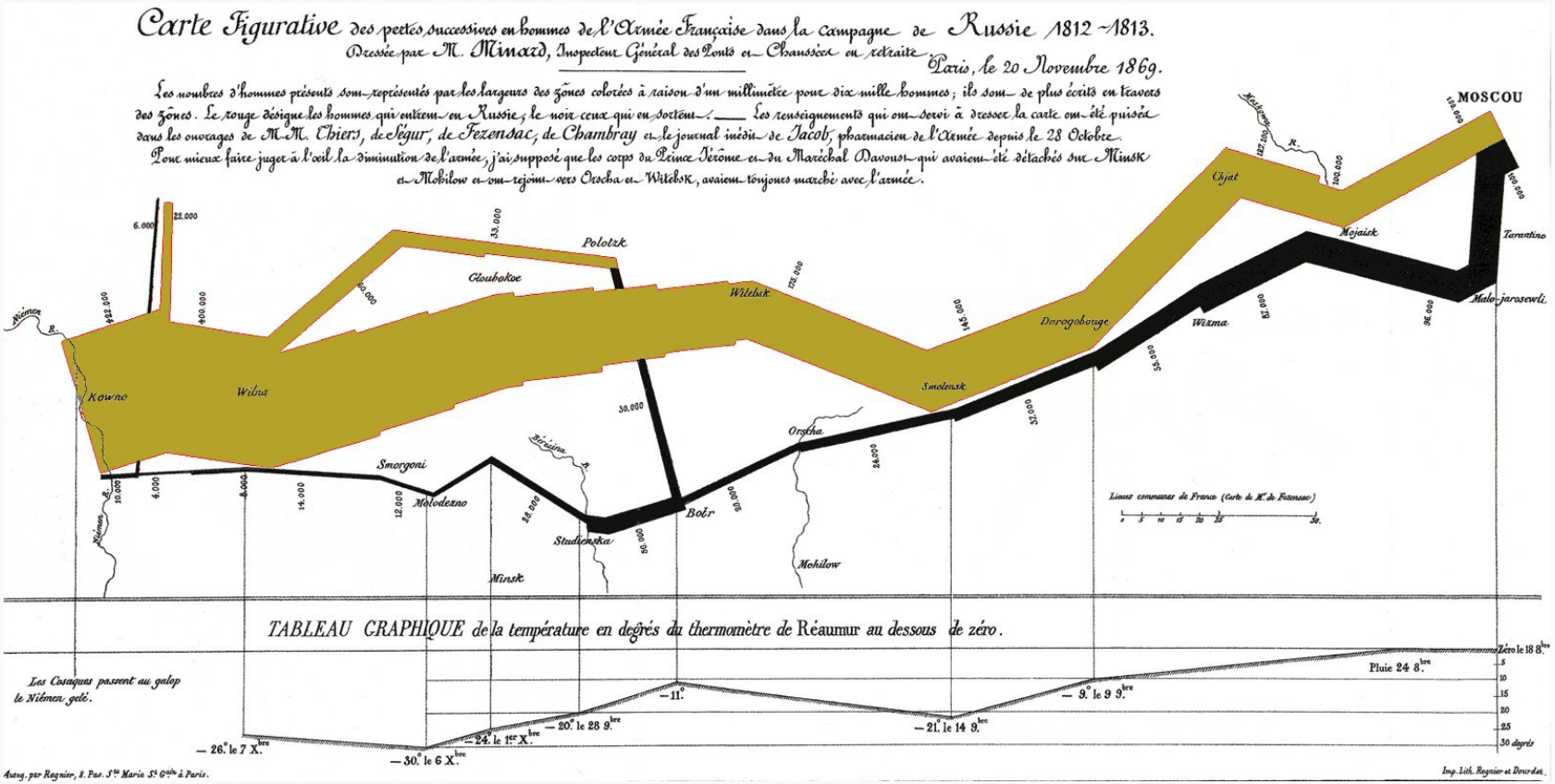
La valía de la información visual es clara. Siempre que sea posible, hay que dar una representación gráfica de los datos.

1812: La Campaña Rusa de Napoleón

En Junio de 1812, Napoleón y su ejército iniciaron la invasión de Rusia. La campaña militar fue un fracaso y marcó el inicio del fin del poder de Napoleón.

La siguiente lámina, de M. Minard, muestra el desarrollo de la campaña. Es considerada como un muy buen ejemplo de cómo transmitir información cuantitativa en forma visual (Tufte (2001) *The Visual Display of Quantitative Information*)

Una imágen dice más que mil palabras



Fronteras Actuales

(unos 900 Km de Lituania a Moscú)

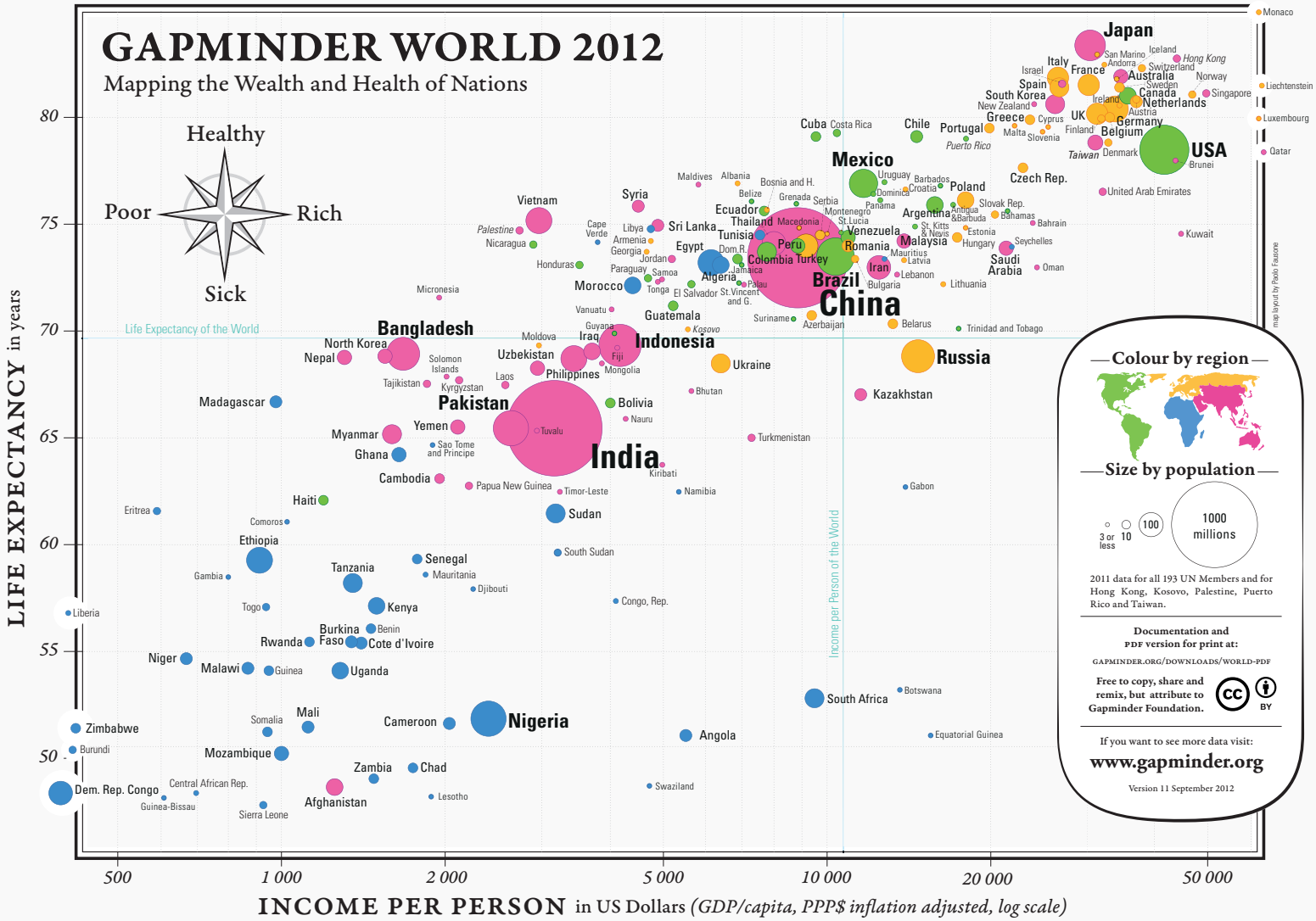
Hans Rosling

Recomendamos accesar alguna de las pláticas de Hans Rosling.

`http://www.ted.com/talks/hans_rosling_shows_the_best_stats_you_ve_ever_seen`

Mostramos en la siguiente lámina un ejemplo del tipo de visualización promovida por Rosling. Se presenta información de muchos países sobre población, ingreso, esperanza de vida y ubicación geográfica.

Una imágen del mundo en 2012



Gapminder

(Gráficas Dinámicas de Rosling)

Apéndice: Gráficas, R.1

```
# Análisis de datos de Anscombe
attach(anscombe)

reg1 = lm(y1~x1); summary(reg1)
reg2 = lm(y2~x2); summary(reg2)
reg3 = lm(y3~x3); summary(reg3)
reg4 = lm(y4~x4); summary(reg4)

par(mfcol=c(2,2),mar=c(4,4,1,1))
plot(x1,y1, col = "red", pch = 21, bg = "orange", cex = 1.2,
       xlim = c(3, 19), ylim = c(3, 13),type="n")
abline(reg1, col = "blue",lwd=2)
points(x1,y1, col = "red", pch = 21, bg = "orange", cex = 1.2)

plot(x2,y2, col = "red", pch = 21, bg = "orange", cex = 1.2,
       xlim = c(3, 19), ylim = c(3, 13),type="n")
abline(reg2, col = "blue",lwd=2)
points(x2,y2, col = "red", pch = 21, bg = "orange", cex = 1.2)

plot(x3,y3, col = "red", pch = 21, bg = "orange", cex = 1.2,
       xlim = c(3, 19), ylim = c(3, 13),type="n")
abline(reg3, col = "blue",lwd=2)
points(x3,y3, col = "red", pch = 21, bg = "orange", cex = 1.2)

plot(x4,y4, col = "red", pch = 21, bg = "orange", cex = 1.2,
       xlim = c(3, 19), ylim = c(3, 13),type="n")
abline(reg4, col = "blue",lwd=2)
points(x4,y4, col = "red", pch = 21, bg = "orange", cex = 1.2)
```

Apéndice: Gráficas, R.2

```
# Código R tomado de: http://www.animatedgraphs.co.uk/
# desarrollado por Robert Grant
# Recreated Hans Rosling bubble plot 1950-2011
# This involves way points with interpolation
# Colours for continents and radius for sqrt(population)

###### PRELIMINARIES #####
library(foreign)
setwd("C:\\...\\Taller2015")
iframes  <- 12   # number of interpolated frames
w        <- 800
h        <- 600  # dimensions in pixels
# get population data
data_pop  <- read.csv("C:\\...\\Taller2015\\gapminder_population.csv")
data_pop  <- data_pop[!is.na(data_pop$pop_1950),]
dimnames(data_pop)[[2]][2] <- "co"
data_pop  <- data.frame(data_pop[,1:2],(data_pop[,3:15]/1000000))  # 196 x 15

# linear interpolation for each year
newdata_pop                <- as.data.frame(matrix(rep(NA,196*62),nrow=196))
dimnames(newdata_pop)[[2]] <- paste("pop_",1950:2011,sep="")
newdata_pop[,(1:13*5)-4]   <- data_pop[,1:13+2]
select   <- 1:61
select   <- select[-seq(1,61,5)]
prevpop  <- ((((select-1)%/%5)+1)*5)-4
nextpop  <- ((((select-1)%/%5)+1)*5)+1
newdata_pop[,select] <- newdata_pop[,prevpop]+(t(matrix(rep((1:4)/5,
 196),nrow=4))*(newdata_pop[,nextpop]-newdata_pop[,prevpop]))
newdata_pop[,62]     <- newdata_pop[,61]+(newdata_pop[,61]-newdata_pop[,60])
data_pop <- data.frame(data_pop[,1:2],newdata_pop)
```

Apéndice: Gráficas, R.3

```
# get life expectancy and GDP data
data_life  <- read.csv("C:\\...\\Taller2015\\gapminder_life.csv")
data_gdp   <- read.csv("C:\\...\\Taller2015\\gapminder_gdp.csv")
gap        <- merge(data_pop,data_life,by="country")
gap        <- merge(gap,data_gdp,by="country")
gap        <- gap[!is.na(gap$gdp_1951),]
gap        <- gap[gap$country!="Russia" & gap$country!="Kuwait",]
# these vectors will be receive interpolation later:
lifei      <- gap$life_1950
gdpi       <- gap$gdp_1950

####### DRAW THE GRAPHS ######
for (i in 1:61) {
  lifei     <- gap[,(64+i)]
  gdpi      <- gap[,(126+i)]
  popi      <- 0.5+(sqrt(1+gap[,(2+i)])/9)
  framefile <- paste("HansR",as.character(((i-1)*(iframes+1))+1),".png",sep="")
  png(filename=framefile,width=w,height=h)
  par("xlog")
  plot(gdpi[gap$co==1],lifei[gap$co==1],cex=popi[gap$co==1],
    col="blue",log="x",xlim=c(300,50000),ylim=c(25,85),
    xlab="GDP",ylab="Life expectancy")
  points(gdpi[gap$co==2],lifei[gap$co==2],cex=popi[gap$co==2],
    col="red")
  points(gdpi[gap$co==3],lifei[gap$co==3],cex=popi[gap$co==3],
    col="orange")
  points(gdpi[gap$co==4],lifei[gap$co==4],cex=popi[gap$co==4],
    col="darkgreen")
  title(main=as.character(i+1949))
  dev.off()
```

Apéndice: Gráficas, R.4

```
  for (j in 1:iframes) {
  # interpolate
    lifei <- gap[,(64+i)]+((j/(iframes+1))*(gap[,(65+i)]-gap[,(64+i)]))
    gdpi  <- gap[,(126+i)]+((j/(iframes+1))*(gap[,(127+i)]-gap[,(126+i)]))
  # draw graph
    framefile <-paste("HansR",as.character(((i-1)*(iframes+1))+1+j),
    ".png",sep="")
    png(filename=framefile,width=w,height=h)
    par("xlog")
    plot(gdpi[gap$co==1],lifei[gap$co==1],cex=popi[gap$co==1],
      col="blue",log="x",xlim=c(300,50000),ylim=c(25,85),
      xlab="GDP",ylab="Life expectancy")
    points(gdpi[gap$co==2],lifei[gap$co==2],cex=popi[gap$co==2],
      col="red")
    points(gdpi[gap$co==3],lifei[gap$co==3],cex=popi[gap$co==3],
      col="orange")
    points(gdpi[gap$co==4],lifei[gap$co==4],cex=popi[gap$co==4],
      col="darkgreen")
    title(main=as.character(i+1949))
    dev.off()
      }
}
```

Apéndice: Gráficas, R.5

```
# then the final graph
  i <- 62
  lifei <- gap[,(64+i)]
  gdpi  <- gap[,(126+i)]
  popi  <- 0.5+(sqrt(1+gap[,(2+i)])/9)
  framefile<-paste("HansR",as.character(((i-1)*(iframes+1))+1),".png",sep="")
  png(filename=framefile,width=w,height=h)
  par("xlog")
  plot(gdpi[gap$co==1],lifei[gap$co==1],cex=popi[gap$co==1],
    col="blue",log="x",xlim=c(300,50000),ylim=c(25,85),
    xlab="GDP",ylab="Life expectancy")
  points(gdpi[gap$co==2],lifei[gap$co==2],cex=popi[gap$co==2],
    col="red")
  points(gdpi[gap$co==3],lifei[gap$co==3],cex=popi[gap$co==3],
    col="orange")
  points(gdpi[gap$co==4],lifei[gap$co==4],cex=popi[gap$co==4],
    col="darkgreen")
  title(main=as.character(i+1949))
  dev.off()

###### MAKE THE VIDEO! ######

shell("C:\\...\\Taller2015\\ffmpeg\\bin\\ffmpeg.exe -report -i
       C:\\...\\Taller2015\\HansR%d.png -b:v 2048k new_video2.mpg",
       mustWork=FALSE)
```

Monty Hall

El problema de Monty Hall

Supongamos que tu estas en un programa de juegos en televisión. Se te ofrece elegir una de 3 puertas y ganarás lo que se encuentre detrás de la puerta de tu elección. Detrás de una de las puertas hay un auto último modelo y detrás de las otras dos hay cabras.

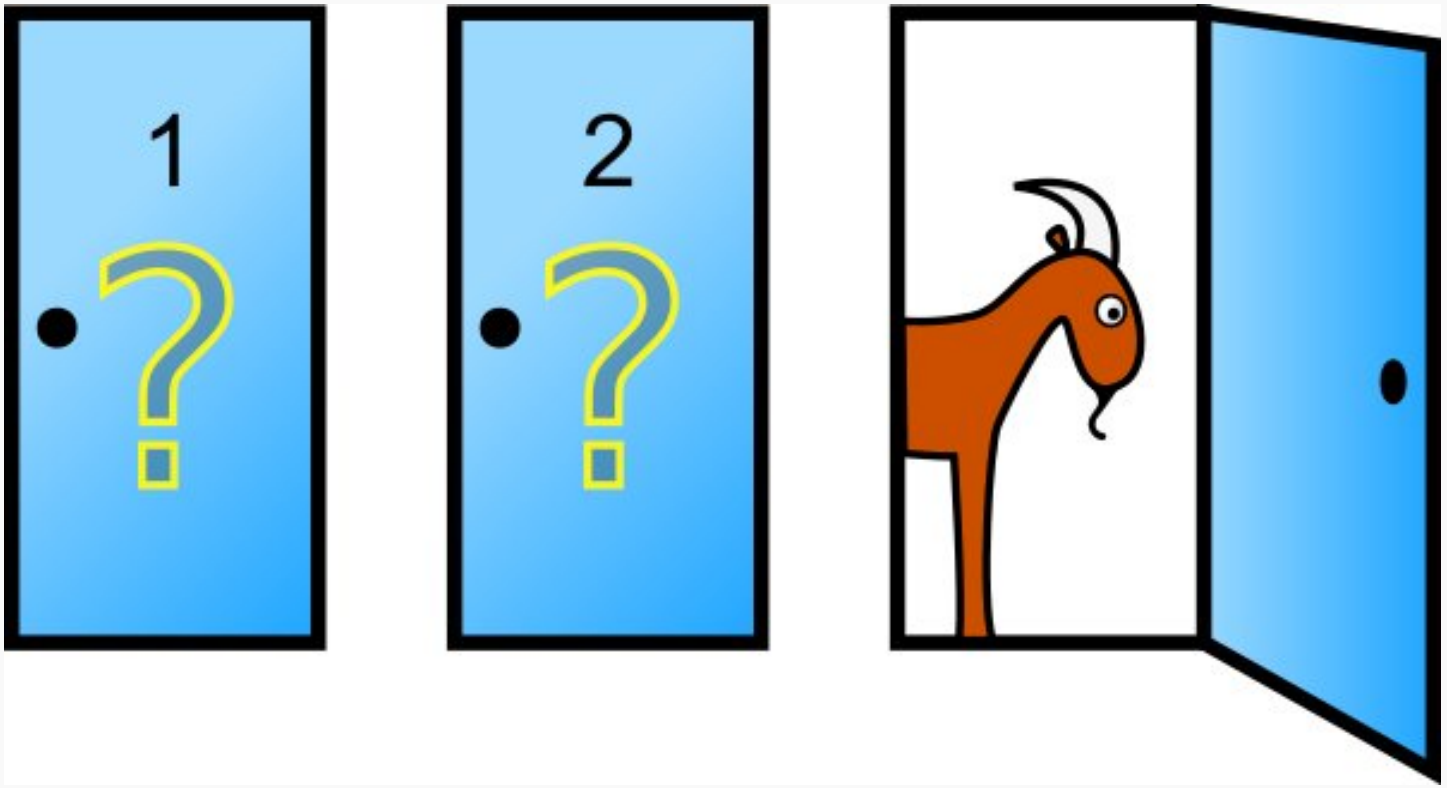
Digamos que tu eliges, digamos, la puerta No. 1; el conductor del programa (el cual sabe lo que hay detrás de las puertas) entonces abre, digamos, la puerta No. 3 la cual tiene una cabra.

El conductor, entonces, te hace la pregunta:

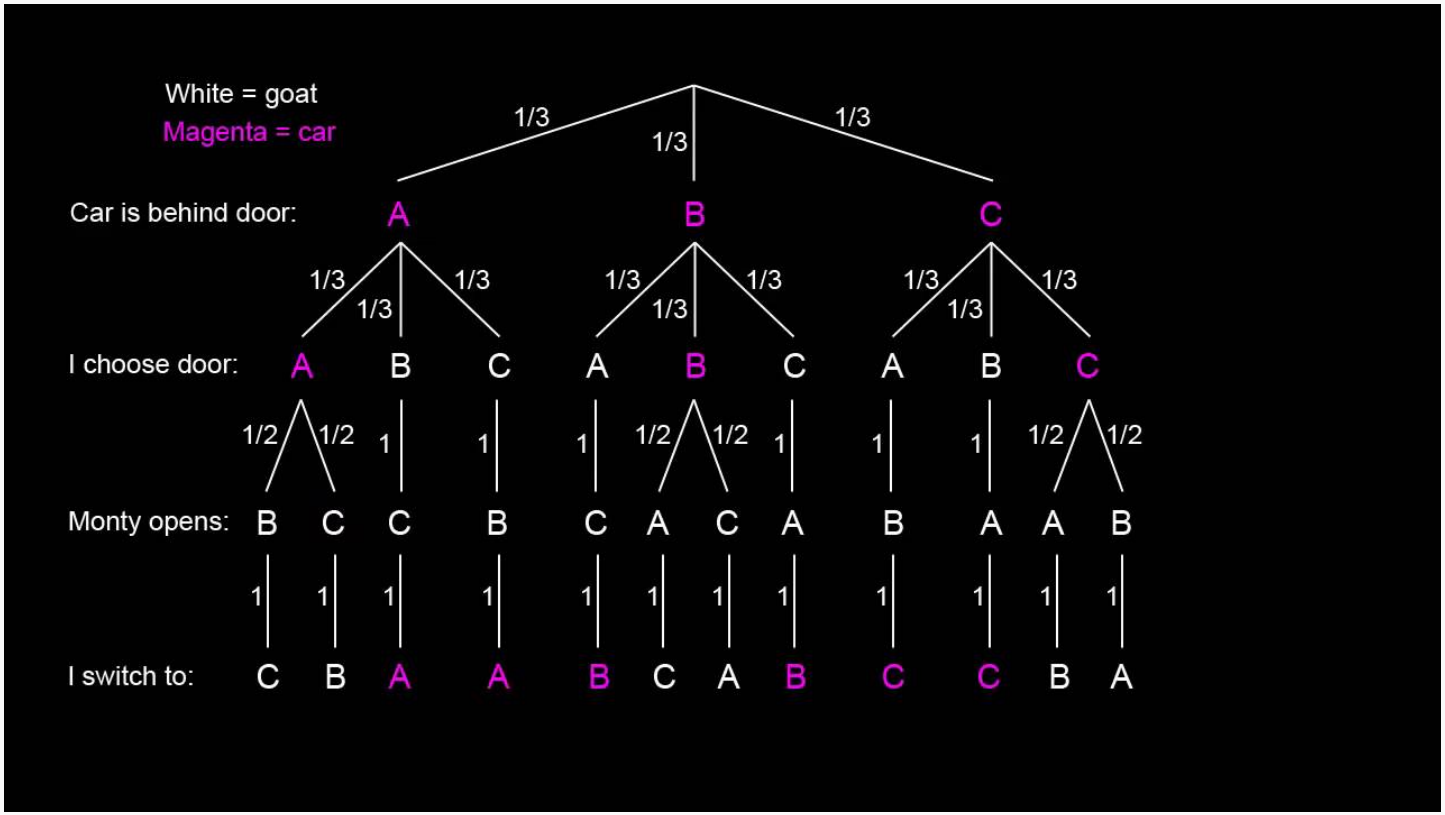
¿Deseas cambiar de puerta?

¿Qué debes hacer, cambiar de puerta o quedarte con la que elgiste primero?

El problema de Monty Hall



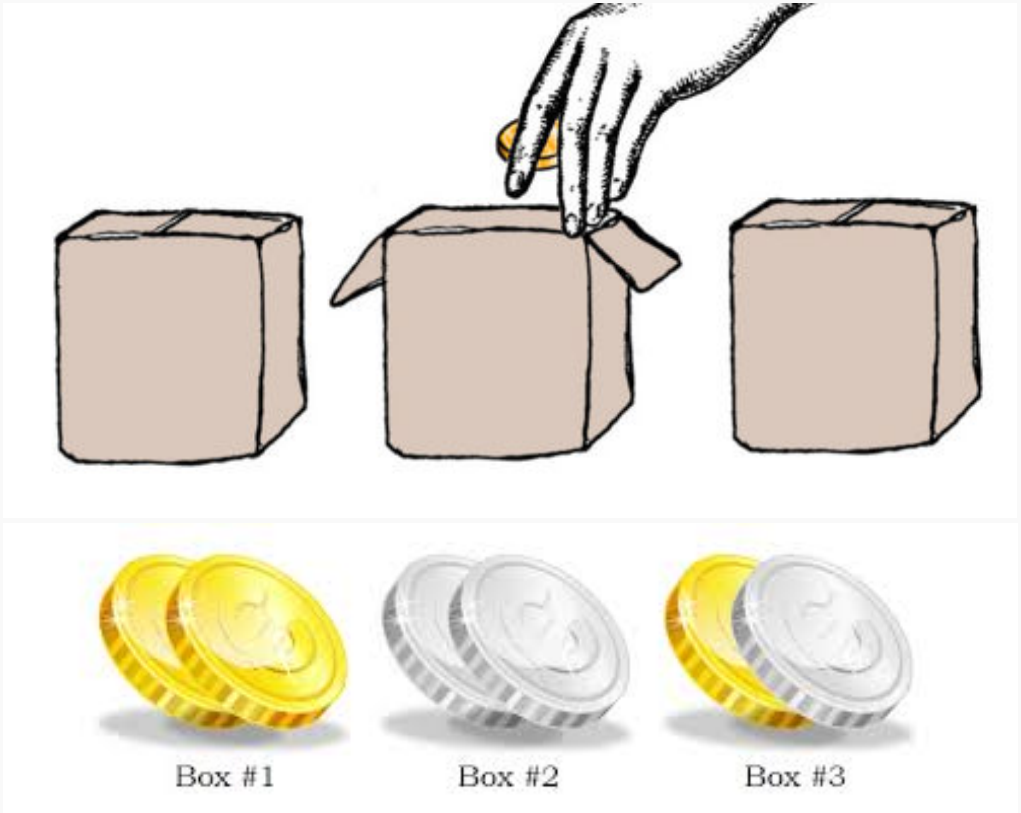
Solución al Problema



Si cambio de puerta, la probabilidad de ganar el auto es:

$$P(\text{auto}) = \frac{1}{3}\cdot\frac{1}{3} + \frac{1}{3}\cdot\frac{1}{3} + \frac{1}{3}\cdot\frac{1}{3} + \frac{1}{3}\cdot\frac{1}{3} + \frac{1}{3}\cdot\frac{1}{3} + \frac{1}{3}\cdot\frac{1}{3} = \frac{6}{9} = \frac{2}{3}$$

La Paradoja de Bertrand



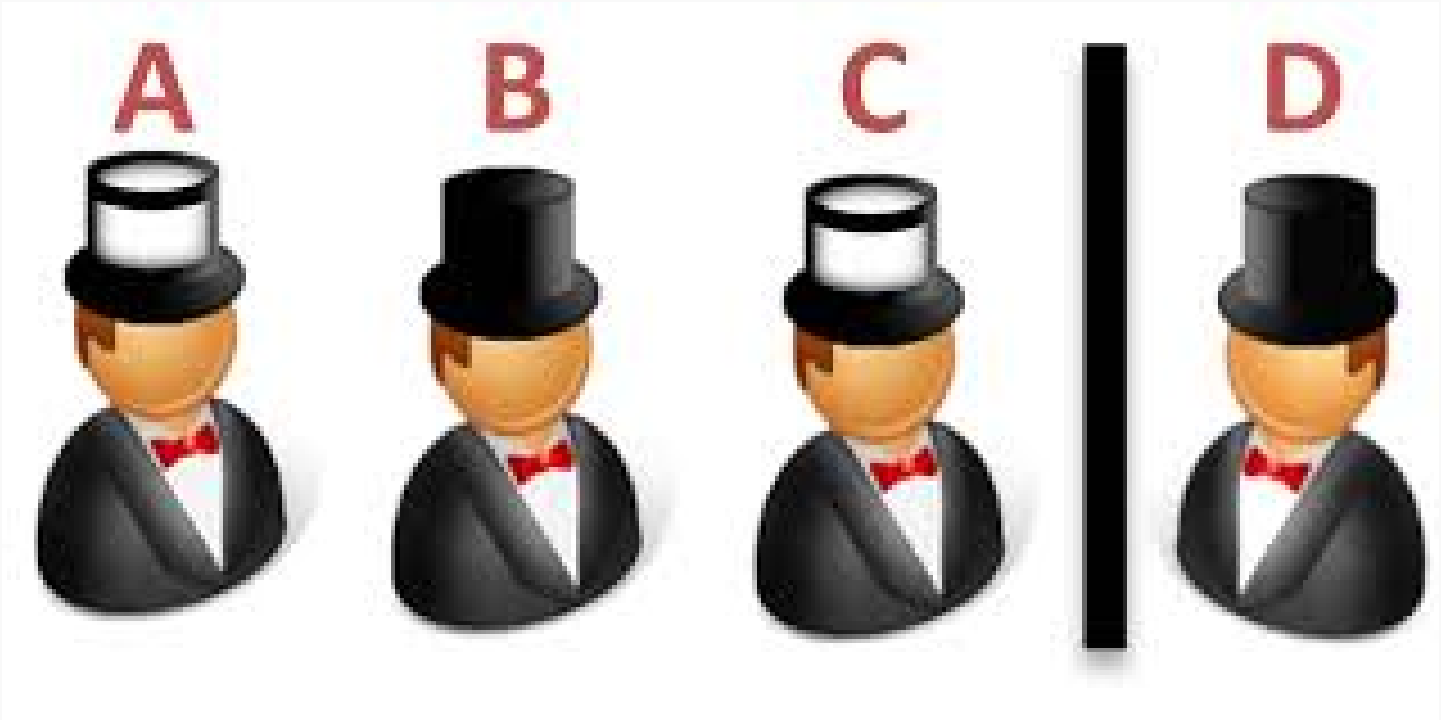
Selecciono una caja al azar. Extraigo una moneda de la caja elegida y esta resultó de Oro. ¿Cuál es la probabilidad de que la otra moneda de la caja sea también de Oro?

La Paradoja de Bertrand

Selecciono una caja al azar. Extraigo una moneda de la caja elegida y esta resultó de Oro. ¿Cuál es la probabilidad de que la otra moneda de la caja sea también de Oro?

$$
\begin{aligned}
P(\text{2a O}|\text{1a O}) &= \frac{P(\text{1a O} \wedge \text{2a O})}{P(\text{1a O})} \\
&= \frac{P(C_1)P(\text{1aO 2aO}|C_1) + P(C_2)P(\text{1aO 2aO}|C_2) + P(C_3)P(\text{2aO 2aO}|C_3)}{P(C_1)P(\text{1a O}|C_1) + P(C_2)P(\text{1a O}|C_2) + P(C_3)P(\text{1a O}|C_3)} \\
&= \frac{\frac{1}{3}(1) + \frac{1}{3}(0) + \frac{1}{3}(0)}{\frac{1}{3}(1) + \frac{1}{3}(0) + \frac{1}{3}\left(\frac{1}{2}\right)} = \frac{2}{3}
\end{aligned}
$$

El Problema de los Tres Prisioneros



El Problema de los Tres Prisioneros

El **Problema de los tres prisioneros**, publicado por Martin Gardner en la revista Scientific American en 1959, es equivalente al problema de Monty Hall. Este problema involucra a tres prisioneros condenados a muerte, uno de los cuales (elgido secretamente al azar) ha sido perdonado.

Uno de los prisioneros le ruega al guardia que le diga el nombre de uno de los otros dos prisioneros que será ejecutado, argumentando que esa información no revela ninguna información acerca de su propia suerte, pero que incrementa su probabilidad de ser perdonado de $1/3$ a $1/2$. El guardia accede y le da el nombre de uno que será ejecutado.

La pregunta aquí es si el conocer la respuesta del guardia, cambia la probabilidad del prisionero de ser perdonado.

El Problema de los Tres Prisioneros

El conocer la respuesta del guardia, ¿cambia la probabilidad del prisionero de ser perdonado?.

Este problema es equivalente al problema de Monty Hall: El prisionero que hace la pregunta sigue teniendo una probabilidad de 1/3 de ser perdonado, pero su compañero (que no fue mencionado por el guardia) tiene ahora una probabilidad de 2/3 de ser perdonado!

Apéndice: Monty Hall, R.1

```
# Monty Hall

set.seed(7272)
M  <- 1000000
aa <- sample(1:3,size=M,replace=TRUE) # puerta donde realmente está el carro
bb <- sample(1:3,size=M,replace=TRUE) # puerta que escogemos
pp <- mean(aa==bb)                    # proporción de veces que acertamos
pp

# cbind(aa[1:21],bb[1:21])

res <- rep("gana",M)
for(i in 1:M){
 if(bb[i]==aa[i]){ res[i] <- "pierde" }else{next}}

table(res)
```

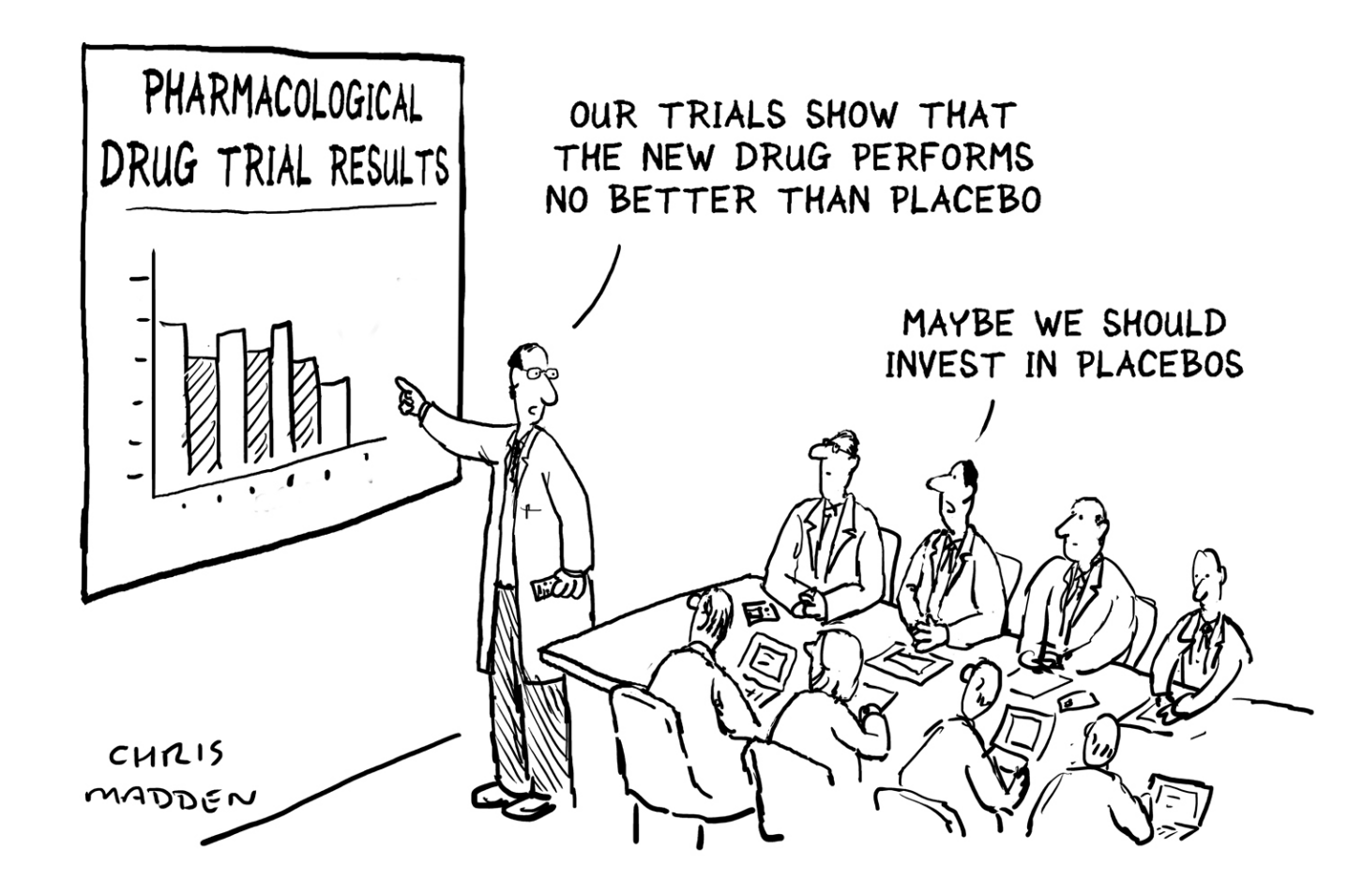
Apéndice: Monty Hall, R.2

```
# Paradoja de Bertrand

M        = 10000
seguir   = TRUE
cajas    = matrix( c("O","O","P","P","P","O"), ncol=2, byrow=T )
NumOro1a = 0
NumOro2a = 0
while( seguir ){
 caja = sample(1:3,size=1)
 if( caja==2 ){next}
 mon = sample(1:2,size=1)
 if( cajas[caja,mon] == "O" ){
  NumOro1a = NumOro1a+1
  if( caja == 1 ){ NumOro2a = NumOro2a+1 }
  if( NumOro1a == M ){ seguir=FALSE }}}
NumOro2a/NumOro1a
```

Medicinas

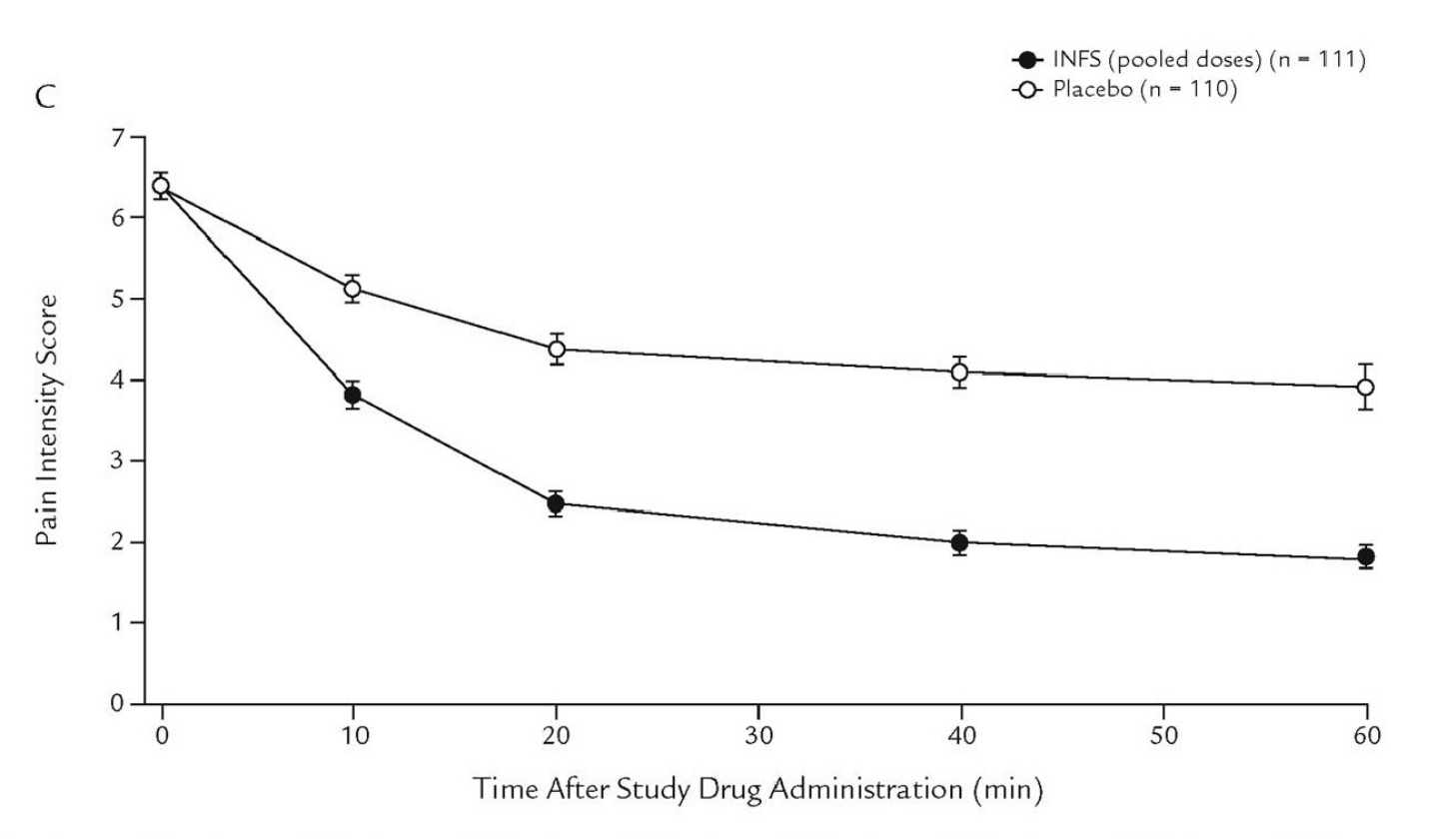
¿Cómo saber si una medicina funciona?

- Tomamos un cierto número de pacientes. Les damos la medicina y vemos si funciona.



¿Cómo saber si una medicina funciona?

- Ok,tomamos dos grupos de pacientes. A unos les damos el medicamento y a otros les damos un placebo y vemos como se comparan.



¿Cómo saber si una medicina funciona?

- Tomamos dos grupos de pacientes. A unos les damos el medicamento y a otros les damos un placebo y vemos como se comparan.
- Estudio ciego: Los pacientes no saben cual tratamiento estan recibiendo.
- Estudio doble ciego: Tampoco los médicos que administran el medicamento saben cual tratamiento se esta aplicando.
- Ahora si, los comparamos.

Comparación de Dos Poblaciones

La prueba t es muy usada para comparar dos grupos. Aquí comentaremos sobre su análogo noparamétrico: La prueba de Mann y Whitney.

Supongamos dos grupos de 4 pacientes cada uno. El grupo A es el grupo tratado y el B es el grupo control (o placebo). Se les administra el tratamiento respectivo y se registra, digamos, el nivel de dolor en alguna escala. Los datos son:

A :	7	4	9	17
B :	11	6	21	14

¿Hay evidencia de que A es menor que B?

Prueba de Mann y Whitney

- Primero, ordenamos las observaciones en orden ascendente

4	6	7	9	11	14	17	21
A	B	A	A	B	B	A	B

- Ahora, nos fijamos en un grupo, digamos A. Para cada elemento de A contamos el número de B's que la anteceden. Para la primer A no hay B's antes que ella, así que llevamos 0 B's.
- Para la segunda A, hay una B antes que ella, así que llevamos $0 + 1 = 1$ B, etc.
- En total tenemos que el número de B's que anteceden a las A's es $U = 0 + 1 + 1 + 3 = 5$; ahora, si este número fuera muy pequeño, esto sería evidencia de que las A's están por abajo que las B's y, si U fuera grande entonces sería evidencia de que las A's están por arriba de las B's.
- Listo!, ya tenemos un estadístico de prueba.

Prueba de Mann y Whitney

- Para poder usar el estadístico U, necesitamos saber su distribución nula; esto es, ¿Cuál es el comportamiento de U cuando en realidad no hay diferencias entre ambas poblaciones?.
- Los dos conjuntos de 4 observaciones pueden arreglarse de 70 formas diferentes; desde $AAAABBBB$ hasta $BBBBAAAA$, en el primer caso $U = 0$ y en el segundo $U = 16$, todos los arreglos tienen, bajo H_0, la misma probabilidad de ocurrencia $(1/70)$.
- Ahora solo nos faltaría obtener los otros 68 arreglos, calcular sus respectivos valores de U, para así tener la distribución nula de U.
- $\cdots$ eso es lo que se debería hacer, pero es medio engorroso, así que tomaremos un camino fácil (simulación) que no es del todo correcto, pero que nos puede dar una buena aproximación.

Prueba de Mann y Whitney

- Hagamos:

```
r      <- rep(0,M)     # con M suficientemente grande
for( i in 1:M){
 orden <- sample( c(0,0,0,0,1,1,1,1) )
 r[i]  <- cuenta( orden ) }
```

  donde `cuenta` es una función que cuenta el número de B's que anteceden a las A's (i.e. calcula el valor de U).

- Al final `r` es un vector de longitud `M` y con ello nos puede dar una buena idea de la distribución de U; la región de rechazo se obtendría de las colas de esa distribución.

Apéndice: Medicinas, R.1

```
# Mann-Whitney

cuenta <- function(a){
 p0  <- which(a==0)
 n0  <- length(p0)
 U   <- 0
 for(i in 1:n0){
  U <- U + sum(a[1:p0[i]]==1)}
 return(U)}

barplot( table(r) )

aa <- table(r)/M
```

Cumpleaños

El Problema de los Cumpleaños

- En un grupo de n personas, ¿Cuál es la probabilidad de que al menos dos personas tengan el mismo cumpleaños?
- ¿Qué tan grande debe ser el grupo, de modo que esta probabilidad sea de al menos 1/2?

El Problema de los Cumpleaños

Consideremos un grupo de n personas,

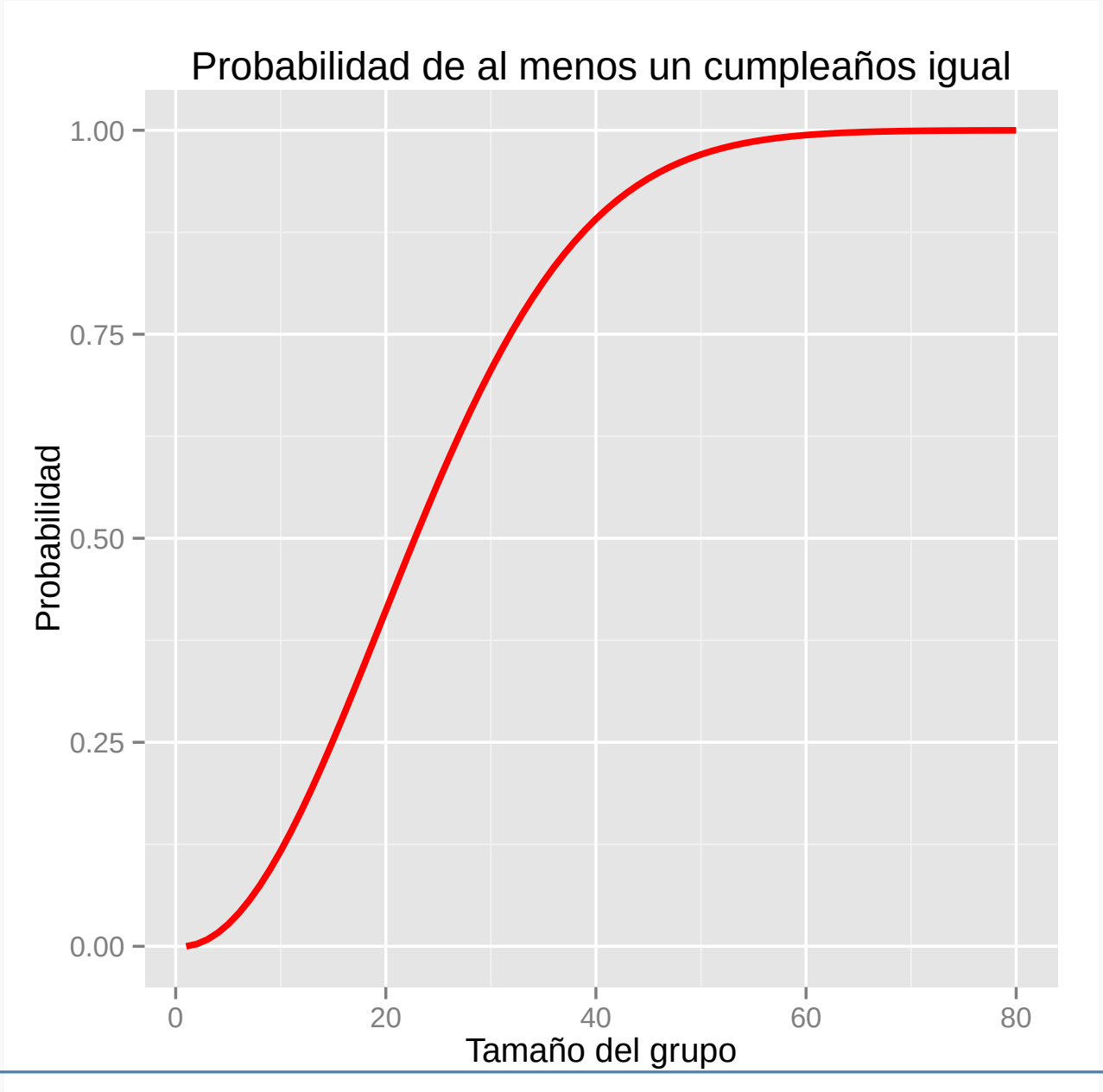
$$P(\text{dos iguales}) = 1 - P(\text{no hay dos iguales})$$

y

$$P(\text{no hay dos iguales}) = \left(\frac{364}{365}\right) \cdot \left(\frac{363}{365}\right) \cdots \left(\frac{365-(n-1)}{365}\right) \cdot$$

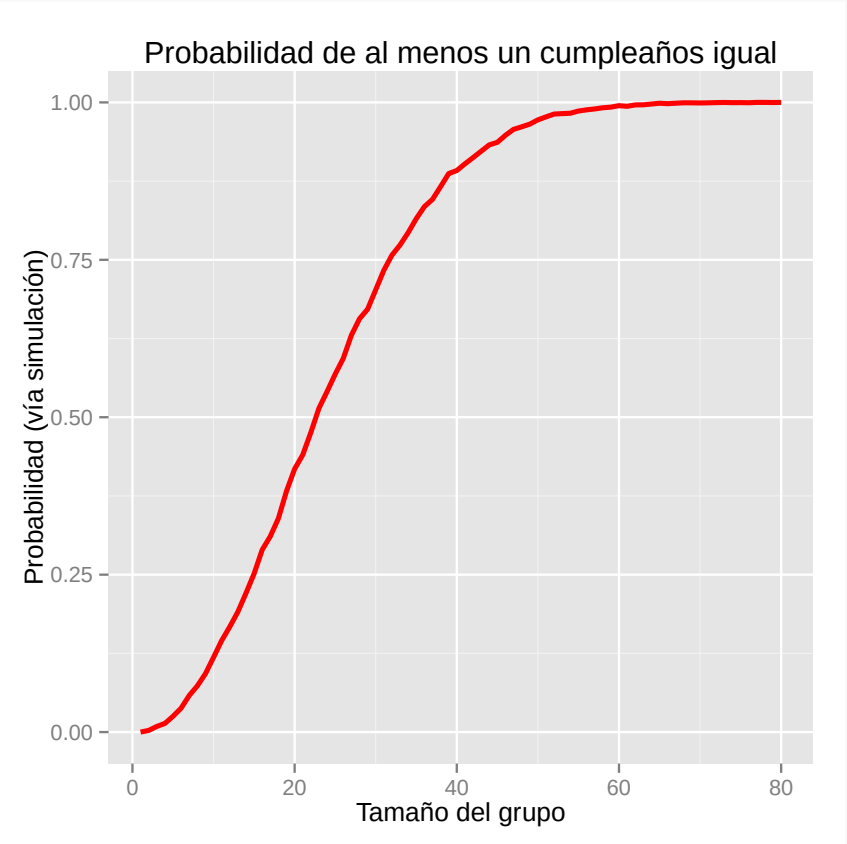
Es fácil ver que, para un grupo de 23 personas, la probabilidad de una coincidencia sobrepasa el 50%!

El Problema de los Cumpleaños



Usando Simulación

Para el problema de los cumpleaños tenemos una expresión cerrada para la probabilidad buscada. En ocasiones esto no es posible, sin embargo, algunas veces es posible usar "'fuerza bruta"'(i.e. computadora), para tener una idea aproximada de la solución. Aquí mostramos el resultado de tal ejercicio:



Apéndice: Cumpleaños, R.1

```
# library(ggplot2)
# Probabilidad de cumpleaños iguales. Expresión analítica:

Pr <- function(n){
  a <- 0
  for(i in 1:n){ a <- a + log((365-i+1)/365) }
  return(1-exp(a)) }

N  <- 80
nn <- 1:N
yy <- rep(0,N)
for(i in 1:N){ yy[i] <- Pr(i) }

qplot( nn, yy, xlab="Tamaño del grupo", ylab="Probabilidad",
  main="Probabilidad de al menos un cumpleaños igual",
  geom="path", colour=I("red"), size=I(1.1) )
```

Apéndice: Cumpleaños, R.2

```
## Usando simulación
# library(ggplot2)

N   <- 80
M   <- 10000
nn  <- 1:N
p   <- rep(0,M)
pr  <- rep(0,N)

for( k in 1:N ){
  for( i in 1:M ){
    sim  <- sample(1:365,size=nn[k],replace=TRUE)
    p[i] <- ( length(unique(sim)) < nn[k] ) }
  pr[k] <- mean(p) }

qplot( nn, pr, xlab="Tamaño del grupo", ylab="Probabilidad (vía simulación)",
  main="Probabilidad de al menos un cumpleaños igual",
  geom="path", colour=I("red"), size=I(1.1) )
```

Apéndice: Cumpleaños, R.3

```
# Cuál es la probabilidad de que al menos una triada de personas
# cumplan años el mismo día?

r   <- 3
N   <- 200
M   <- 1000
nn  <- 1:N
p   <- rep(0,M)
pr  <- rep(0,N)

for( k in 1:N ){
  for( i in 1:M ){
    sim  <- sample(1:365,size=nn[k],replace=TRUE)
    bb   <- as.vector( table(sim) )
    p[i] <- ( any( bb > r-1 ) ) }
  pr[k] <- mean(p) }

plot( nn, pr, xlab="Tamaño del grupo", ylab="Probabilidad",
  main="Probabilidad de al menos una triada de cumpleaños",
  type="l", col="red", lwd=2 )
grid()
```

Elecciones

¿Cómo saber quien va a ganar las elecciones?

Conteos Rápidos

Supongamos una población de $N = 100,000$ habitantes y organizamos un conteo rápido para darnos una idea de las tendencias del voto.

- ¿Qué tan grande debe ser la muestra para estar bastante confiados de nuestras estimaciones sobre quien va a ganar?.
- ¿Qué tanto depende la respuesta del punto anterior de N, el tamaño total de la población?.

Tamaño de Muestra

Un intervalo de confianza para la proporción de votantes está dado por:

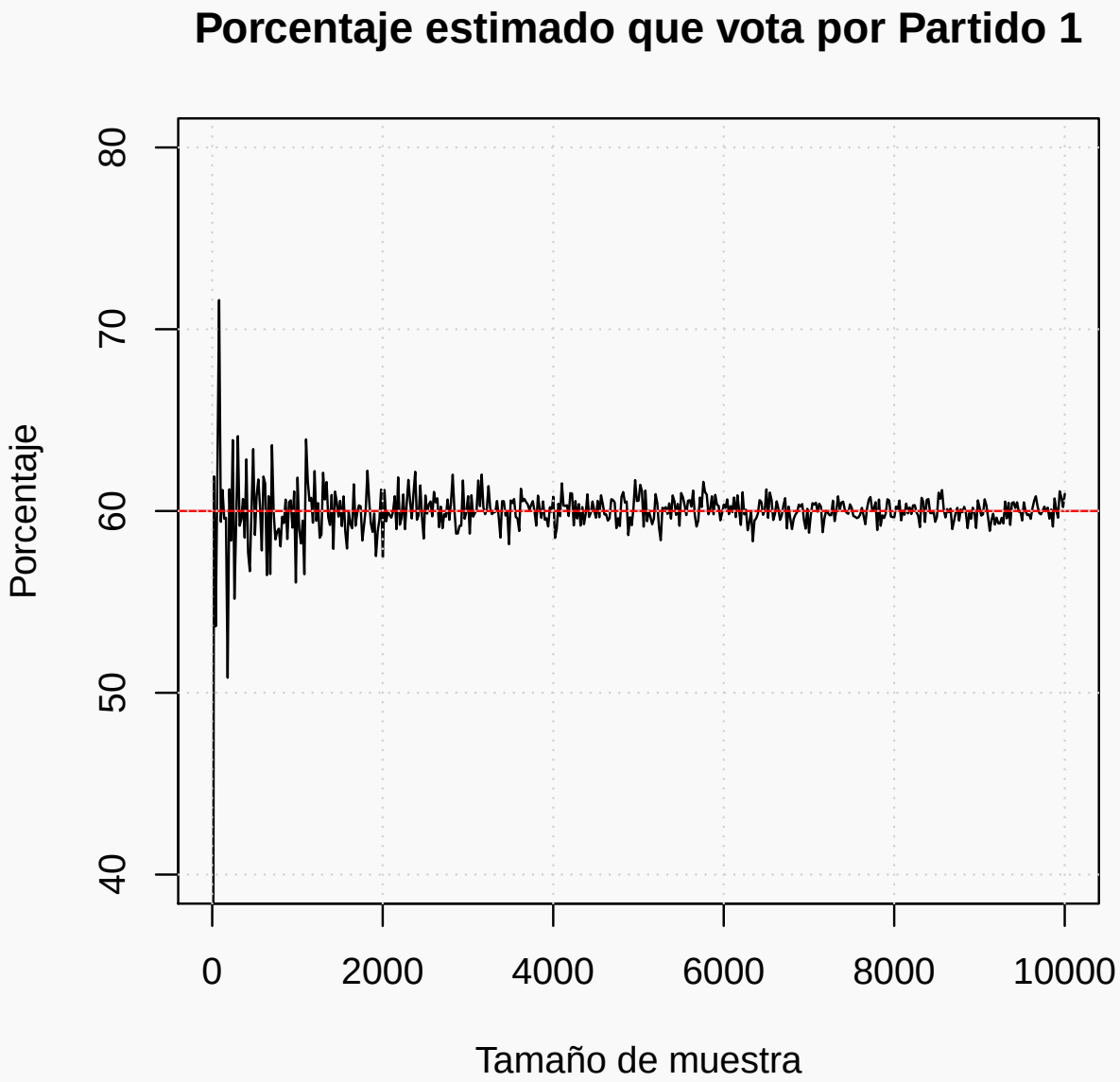
$$\widehat{p} \pm z\sqrt{\frac{1}{n}p(1-p)}$$

de aquí, es fácil ver que si deseamos un error máximo de tamaño e, entonces el tamaño de muestra debe ser tal que

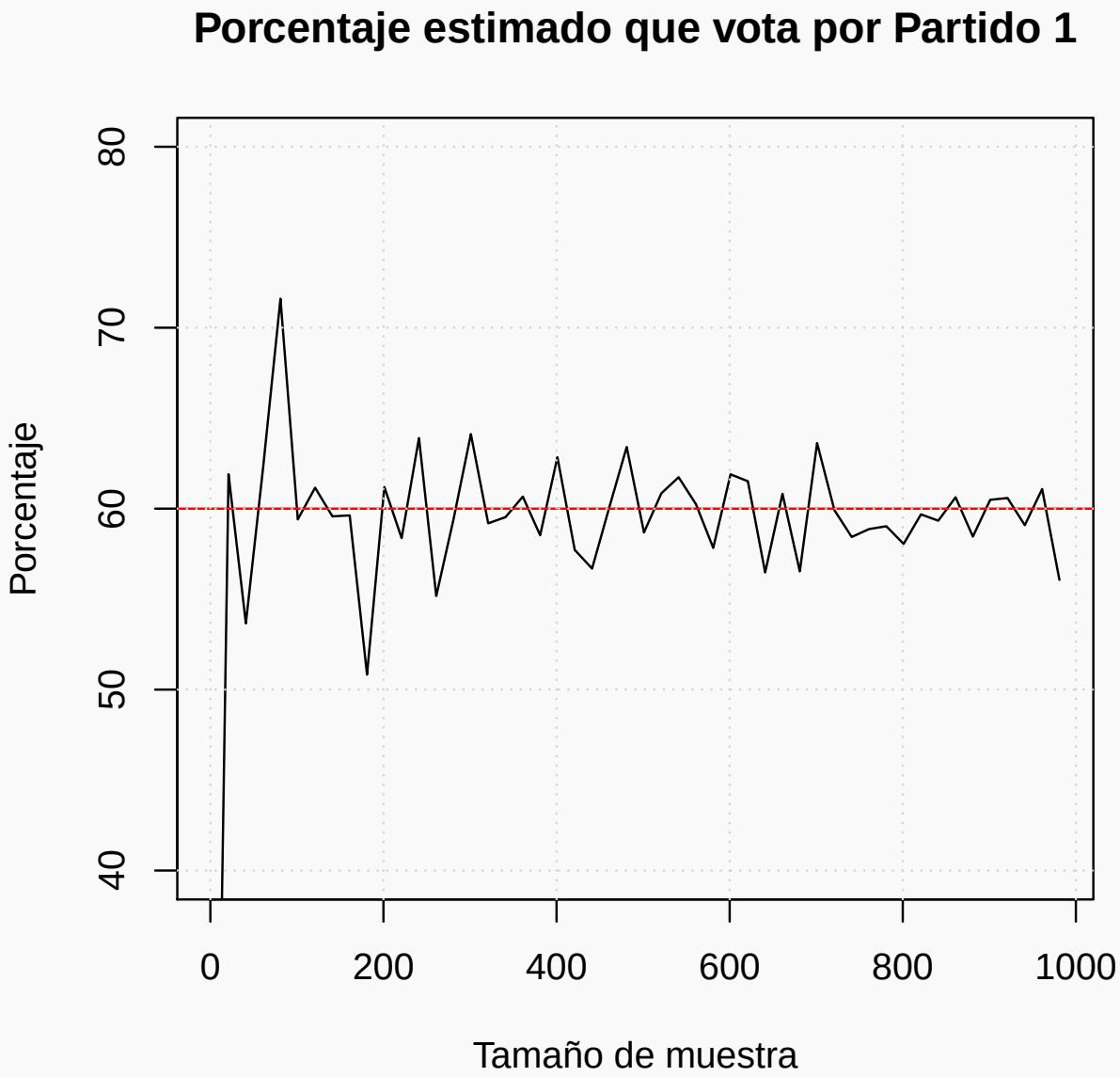
$$n > \frac{z^2 p(1-p)}{e^2}$$

(En la vida real, los supuestos de muestreo aleatorio simple no son fáciles de conseguir, sin embargo, en nuestro caso supondremos que esto es válido).

Simulando el proceso de muestreo



Fijándonos en muestras pequeñas



Conteos Rápidos IFE

- En México se instalan unas 130,000 casillas para votar.
- El Programa de Resultados Electorales Preliminares (PREP) selecciona unas 7,500 casillas (un poco menos del 6% del total) para realizar predicciones tempranas (estas casillas -seleccionadas de antemano- cuentan con sistemas especiales de transmisión electrónica).
- La elección de las casillas se basa en un "diseño estratificado", más complicado que un muestreo aleatorio simple, pero el punto básico es el mismo: Es posible dar, de forma confiable, resultados preliminares en una elección.

Apéndice: Elecciones, R.1

```
# Elecciones (suponemos dos partidos)
par1 <- .60
par2 <- .40

# Voto real de la población
N   <- 100000
pob <- c(rep(1,N*par1),rep(2,N*par2))

# los revolvemos (no hay necesidad, solo lo hacemos
# por razones psicológicas)
set.seed(84848)
pob <- pob[sample(1:N)]

# obtenemos una muestra aleatoria simple
# y vemos el efecto del tamaño de muestra

mm  <- seq(1,10001,by=20)
nn  <- length(mm)
vx1 <- rep(0,nn)   # propoción estimada de votantes por partido 1
for(i in 1:nn){
 aa     <- sample(1:N,size=mm[i],replace=FALSE)
 vx1[i] <- 100*mean(pob[aa]==1) }
```

Apéndice: Elecciones, R.2

```
# Gráficas
plot(mm,vx1,type="l",xlab="Tamaño de muestra",ylab="Porcentaje",
 main="Porcentaje estimado que vota por Partido 1",ylim=c(40,80))
abline(h=100*par1,col="red")
grid()

# zoom en muestras pequeñas
plot(mm[1:50],vx1[1:50],type="l",xlab="Tamaño de muestra",ylab="Porcentaje",
 main="Porcentaje estimado que vota por Partido 1",ylim=c(40,80))
abline(h=100*par1,col="red")
grid()

# Cómo contestar a la segunda pregunta acerca del efecto de N?
```

Aquí terminamos!