R / Bioconductor: Curso Intensivo

Leonardo Collado Torres
Licenciatura en Ciencias Genómicas, UNAM
www.lcg.unam.mx/~lcollado/index.php

Cuernavaca, México
Oct-Nov, 2008

Pruebas Estadísticas

Unos conceptos

Recuerden algunos conceptos:

- La inferencia estadística es el proceso de hacer juicios sobre una población con base en una muestra.
- Una variable aleatoria puede tener un valor observado o un potencial de valores descritos por su distribución de probabilidades.
- La media poblacional de la variable aleatoria X es igual a μ y al valor esperado de X.
- La varianza poblacional se denomina σ^2 y la desviación estándar es su raíz: σ.
- Si $f(x)$ es la función de densidad de probabilidad[1] para X, para toda b, $P(X \geq b)$ es igual al área bajo f que está a la izquierda de b.

[1] Es función de distribución de probabilidad si X es discreta; o función de densidad de probabilidad si X es contínua.

Muestreo

- Para hacer inferencia estadística necesitamos una muestra de la población; osea, una secuencia de variables aleatorias.
- Estas pueden estar idénticamente distribuidas si tienen la misma distribución. Además, generalmente asumimos que son independientes.
- En `R` podemos obtener muestras de una población dada con la función sample. Puede ser con o sín remplazo dependiendo del valor del argumento `replace`.
- Por ejemplo, tiramos 10 veces un dado de 20 lados[2]

```
> sample(1:20, size = 10, replace = TRUE)

 [1] 18  6  2 20  4  9 11 19  5 12
```

Muestreo

- En realidad trabajaremos con distribuciones muestrales. Estas pueden ser muy complicadas pero unas se relacionan con distribuciones poblacionales. Por ejemplo, la desviación estándar muestral es igual a $\sigma/\sqrt{n}$.

[2] Se usan en DnD

Funciones sobre distribuciones

- En R hay toda una gama de distribuciones. Estas generalmente van a tener 4 funciones:
  - Con la función d obtenemos la función función de densidad de probabilidad.
  - p nos regresa la función de densidad de probabilidad acumulada.
  - q nos da los los cuantiles de una distribución.
  - r nos da valores aleatorios que siguen a la distribución especificada.
- Por ejemplo, aquí jugamos un poco con la distribución uniforme en $[0, 3]$:

```
> dunif(x = 1, min = 0, max = 3)

[1] 0.3333333

> punif(q = 2, min = 0, max = 3)
```

Funciones sobre distribuciones


```
[1] 0.6666667
> qunif(p = 1/2, min = 0, max = 3)
[1] 1.5
> runif(n = 1, min = 0, max = 3)
[1] 0.6101635
```

Distribuciones

- R tiene ya varias poblaciones con las que podemos jugar :) como pueden ver con el siguiente comando:

```
> help.search("distribution", package = "stats")
```

- Para la Bernoulli pueden usar la función `sample`.
- La binomial usa binom. Por ejemplo:

```
> dbinom(5, size = 10, prob = 1/2)
```

```
[1] 0.2460938
```

- Para la normal, usamos norm.
- La uniforme ya la conocen: unif.
- La distribución lognormal es con lnorm. Tiene un sesgo importante hacia la derecha.

Distribuciones

- Las distribuciones t, F y χ^2 sirven para describir distribuciones muestrales. Se usan con t, f y chisq.
- Cada una tiene argumentos diferentes, así que chequen la sección de ayuda :)

Teorema del límite central

Recordando:

- Ley de los grandes números: *En un contexto estadístico, las leyes de los grandes números implican que el promedio de una muestra al azar de una población de gran tamaño tenderá a estar cerca de la media de la población completa.*[3]
- Con esta ley en mente, el teorema del límite central nos dice que si tienes muchos datos, estos se aproximarán a una normal.

Teorema del límite central

- Más rigurosamente, este teorema dice que para cualquier población padre con media μ y desviación estándar σ, la distribución muestral de $\overline{X}$ con n grande satisface:

$$P(\frac{\overline{X} - \mu}{\sigma/\sqrt{n}} \leq b) \approx P(Z \leq b)$$

  donde Z es una variable aleatoria normal estándar. Osea, que si n es suficientemente grande, la distribución de $\overline{X}$ una vez estandarizada es aproximadamente una distribución normal estándar.
- Pueden reemplazar a σ con la desviación estándar muestral s y funciona.

[3] De acuerdo a wiki

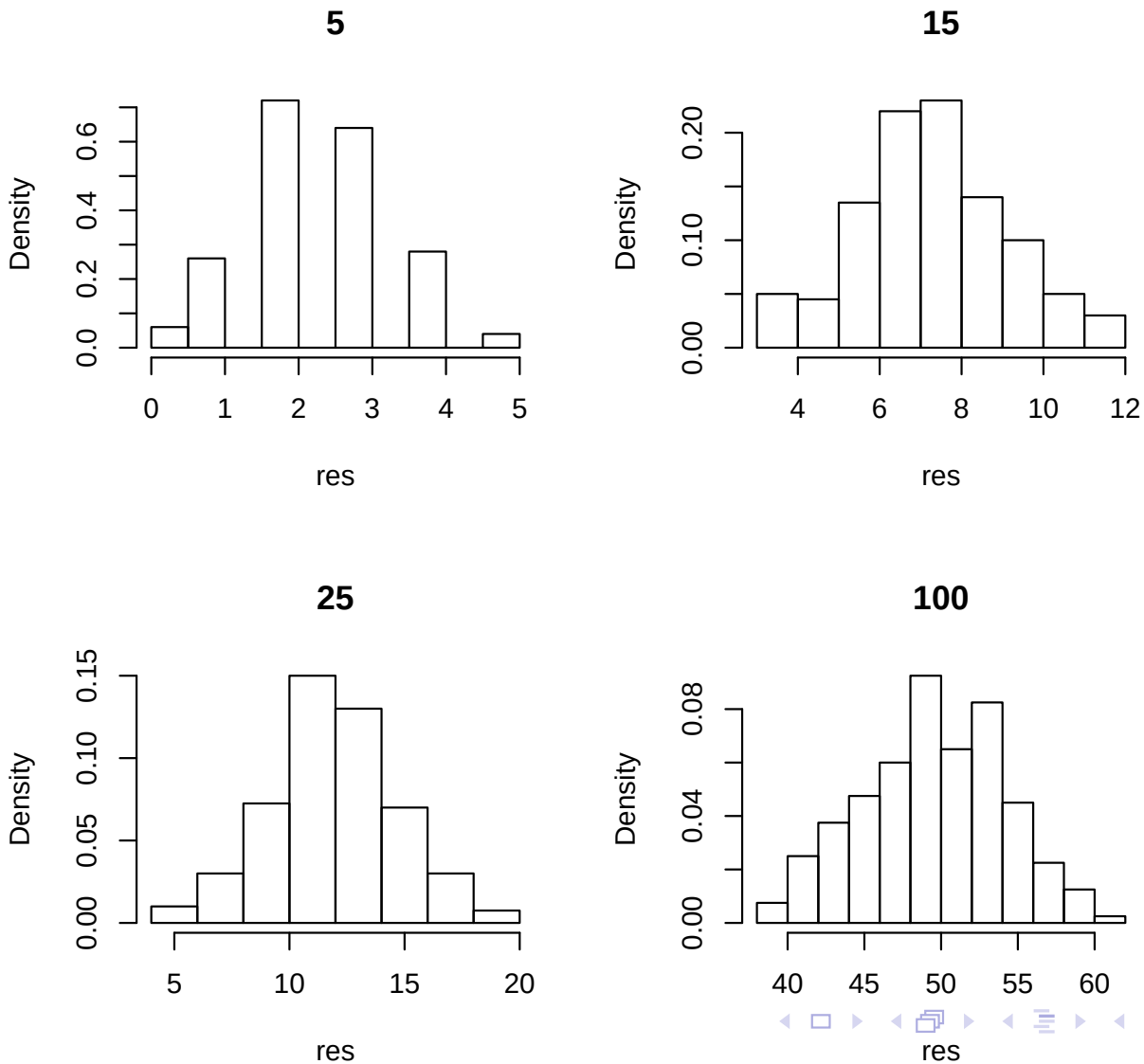
Simulaciones

- Algo que podría parecer muy loco es repetir lo mismo muchas veces esperando obtener resultados diferentes. Bueno, en estadística luego es bueno hacer simulaciones.
- Con simulaciones podemos obtener información la forma, las colas, la media y la varianza de una distribución.
- A continuación hacemos simulaciones para justificar la n del teorema del límite central.

TLC: simulación


```
> m <- 200
> p <- 1/2
> n <- c(5, 15, 25, 100)
> par(mfrow = c(2, 2))
> for (i in 1:4) {
+     res <- rbinom(m, n[i], p)
+     hist(res, prob = TRUE, main = n[i])
+ }
> par(mfrow = c(1, 1))
```

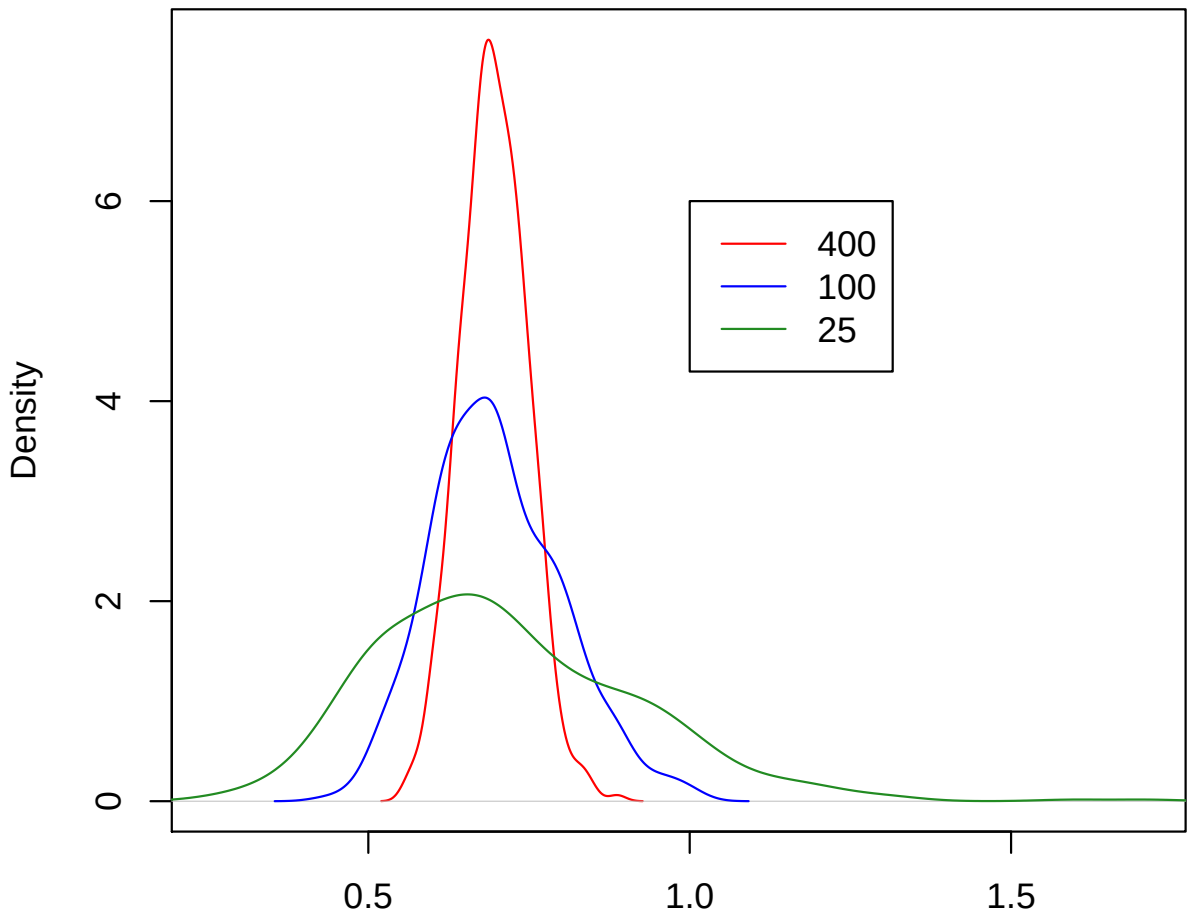
TLC: simulación



Ahora con medianas de exp


```
> m <- 500
> res.25 <- c()
> res.100 <- c()
> res.400 <- c()
> f <- function(n) median(rexp(n))
> for (i in 1:m) res.25[i] <- f(25)
> for (i in 1:m) res.100[i] <- f(100)
> for (i in 1:m) res.400[i] <- f(400)
> plot(density(res.400), xlim = range(res.25),
+     type = "l", main = "", xlab = "Medianas")
> lines(density(res.100))
> lines(density(res.25))
```

Medianas de exp



Distribuciones muestrales para la mediana con n=25, 100, 400

Bootstrap

- La idea es crear una nueva muestra del mismo tamaño que la original.
- Lo podemos hacer con un muestro con remplazo usando la función sample.
- Con este tipo de muestras podemos estimar ciertos parámetros, como la media y la varianza. Usemos el set de datos `bycatch` de `UsingR`.

```
> library(UsingR)
> data(bycatch)
> hauls <- rep(bycatch$no.albatross,
+     bycatch$no.hauls)
> n <- length(hauls)
> xbarstar <- c()
> for (i in 1:1000) {
```

Bootstrap


```
+     boot.samp <- sample(hauls,
+         n, replace = TRUE)
+     xbarstar[i] <- mean(boot.samp)
+ }
> mean(xbarstar)

[1] 0.2776544

> sd(xbarstar)

[1] 0.04091449
```

Para que sirven

- Un intervalo de confianza nos da un rango donde se encuentra el valor de un parámetro con cierta probabilidad.
- Por ejemplo, de acuerdo a `age.universe` de `UsingR`, la edad del universo es de 13.7 mil millones de años con un margen de error de 1
- Los intervalos de confianza nos van a servir para estimar parámetros desconocidos. Para más info chequen en wikipedia.
- Podemos usar varias funciones: prop.test para proporciones donde $np > 5$ y $n(1-p) > 5$. binom.test también puede servirnos si queremos usar la distribución binomial.

prop.test

Example

Imaginemos que tenemos 10000 bolas con valores 1 ó 0; 5600 tienen 1s. 1000 veces seleccionamos 100 de estas al azar y contamos cuantos 1s encontramos. Queremos encontrar el intervalo de confianza usando `prop.test` de 95% para la proporción de bolas marcadas con 1 en la población. Luego comparenlo con proporción real.

```
> pop <- rep(0:1, c(10000 - 5600,
+     5600))
> res <- c()
> for (i in 1:1000) res[i] <- sum(sample(pop,
+     100))
> prop.test(mean(res), 100, conf.level = 0.95)
```

Una sola cola

- En `R` podemos usar el argumento `alt` para especificar si queremos un intervalo de confianza de una sola cola. Puede ser `"less"` o `"greater"`.
- Además de las funciones que vimos, está la t.test.
- Para que usar t.test? Acuérdense de que la distribución t nos sirve si n es pequeña.

t.test

Example

Le dicen a Pepe que la temperatura ideal para servir el café es de 180 grados Farenheit. Tenemos 5 mediciones y queremos encontrar un intervalo de 90% de la forma $(-\infty, b]$ para la temperatura media:

Así lo pueden resolver:

```
> x <- c(175, 185, 170, 184, 175)
> t.test(x, conf.level = 0.9, alt = "less")
```

t.test


```
One Sample t-test

data:  x
t = 61.5671, df = 4, p-value = 1
alternative hypothesis: true mean is less than 0
90 percent confidence interval:
     -Inf 182.2278
sample estimates:
mean of x
    177.8
```

Más intervalos

- Pueden usar la χ^2 para encontrar valores de intervalo de varianzas, en vez de medias como lo veníamos haciendo.
- Para comparar dos proporciones y checar si vienen de la misma población, podemos volver a usar `prop.test`. Noten que usa una correción por continuidad.

Example

En el transcurso de dos semanas se hizo la misma encuesta. En la primera 560 de 1000 digeron que sí, en la segunda 570 de 1200 digeron que sí. Cual es el intervalo de confianza para la diferencia de proporciones?

- Concluyen que hay o no hay una diferencia real en los parametros de la población?

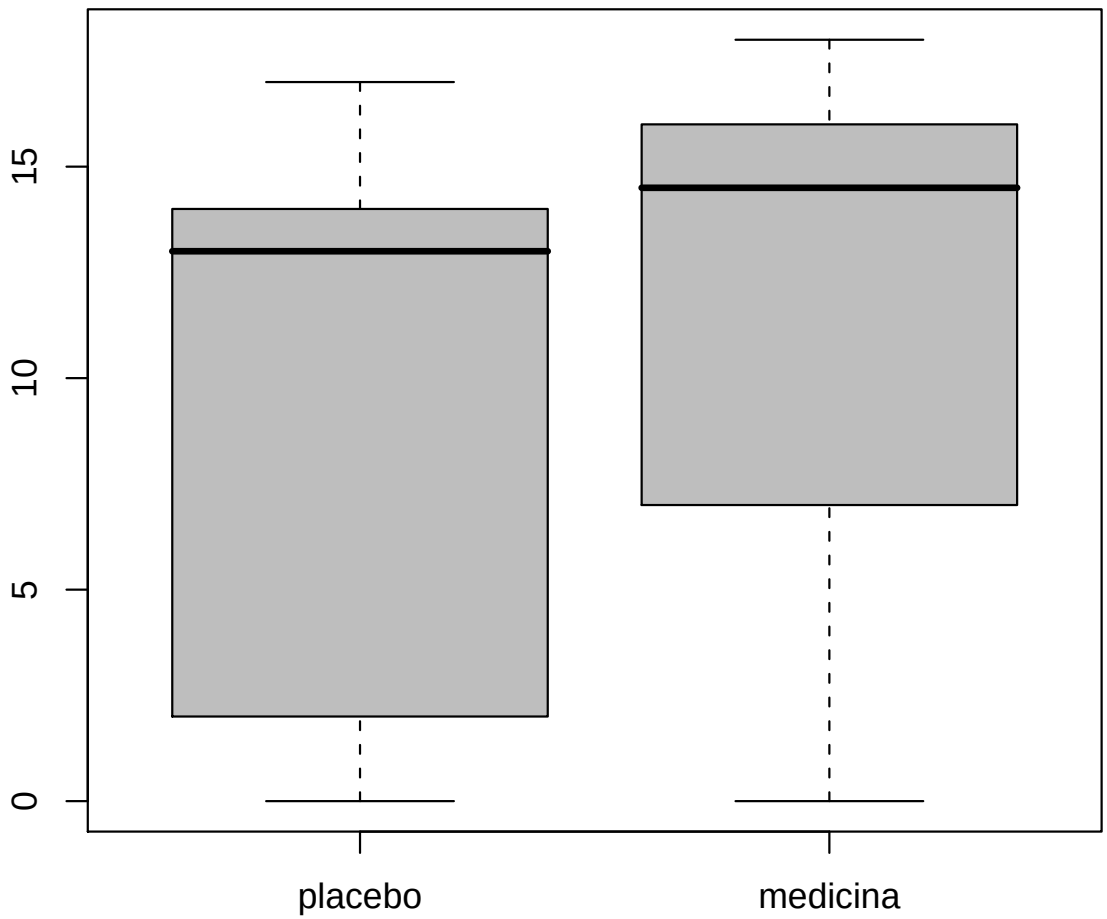
Diferencia de medias

- Si quieren encontrar un intervalo de confianza para una diferencia de medias, tienen que usar la distribución t.
- Acuérdense que al hacer esto vamos a tener que escoger entre suponer que las varianzas son iguales o que son diferentes.
- Eso afecta a nuestra fórmula del error estándar y de los grados de libertad.
- A parte, hay que asumir que las variables son independientes o dependientes en cuyo caso deben usar el argumento `paired` o usar la notación de fórmula.
- Aquí les muestro un ejemplo de variables independientes.

Diferencia de medias


```
> x <- c(0, 0, 0, 2, 4, 5, 13, 14,
+     14, 14, 15, 17, 17)
> y <- c(0, 6, 7, 11, 13, 16, 16,
+     16, 17, 18)
> boxplot(list(placebo = x, medicina = y))
> t.test(x, y, var.equal = TRUE)
```

Checando igualdad de varianza



Unas no paramétricas

- Queremos encontrar el intervalo de confianza para la mediana, lo cual es útil si nuestros datos tienen un sesgo considerable.
- Por un lado, podemos usar la binomial y ordenar nuestros datos de la siguiente forma:

```
> x <- c(110, 12, 2.5, 98, 1017,
+      540, 54, 4.3, 150, 432)
> n <- length(x)
> j <- qbinom(0.05, n, 1/2)
> sort(x)[c(j, n + 1 - j)]
```

```
[1]   4.3 540.0
```

Unas no paramétricas

- Otra opción es usar la estadística de signo ordenada de Wilcoxon. Esta la pueden usar con signrank aunque tienen que especificar el argumento `conf.int=TRUE`.
- Con esta prueba tenemos que asumir simetría con respecto a la mediana. Claro, si no son simétricas podemos transformar los datos y luego transformar el intervalo de regreso como a continuación:

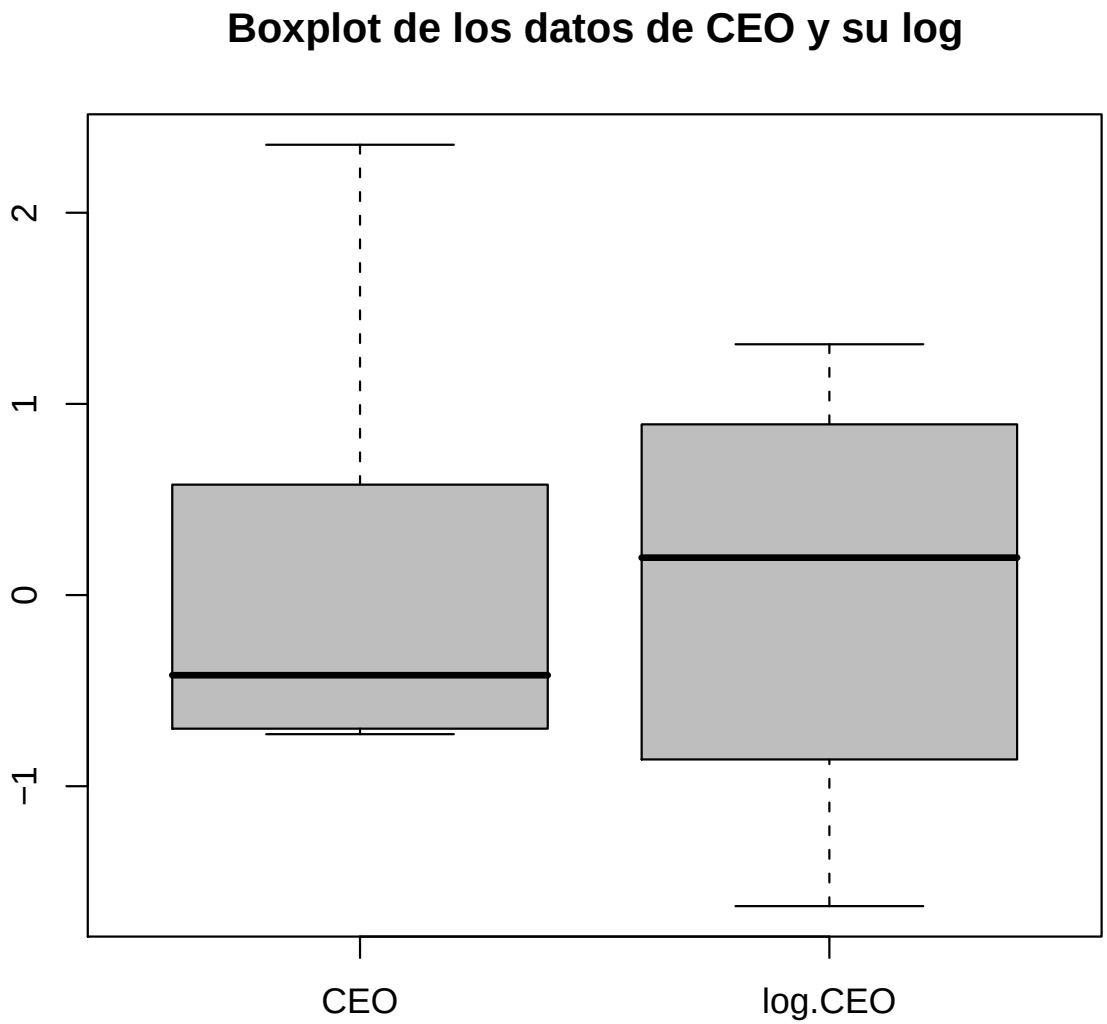
```
> boxplot(scale(x), scale(log(x)),
+     names = c("CEO", "log.CEO"))

> xx <- wilcox.test(log(x), conf.int = TRUE,
+     conf.level = 0.9)
> exp(xx$conf.int)
```

Unas no paramétricas


```
[1]  19.36492 254.55844
attr(,"conf.level")
[1] 0.9
```

- Comparen los dos intervalos de confianza!

Cual es simétrica vs la mediana?



Definiciones

- Los intervalos de confianza son una forma de inferencia estadística. Las pruebas de significancia o pruebas de hipótesis son otra que asumen un valor exacto para el parámetro de la población en vez de un rango.
- Ya con el valor, estas pruebas calculan una probabilidad basado en una muestra dado el valor asumido.
- Vamos a tener que definir una hipótesis nula H_0 que contrastamos con la hipótesis alternativa H_A. Con la prueba de hipótesis queremos ver si H_0 es razonable dado los datos que tenemos. Nunca aceptamos H_A pero podemos rechazar H_0 en favor de H_A.

Definiciones

- El valor p se calcula con las asunciones de H_0 y es la probabilidad de que la prueba estadística[4] es el valor observado o uno más extremo descrito con la hipótesis alternativa.
- valor p = P(probabilidad de obtener un valor como el observado o más extremo dado H_0)
- Si el valor p es pequeño, la prueba es estadísticamente significativa lo cual nos indica que es poco probable que H_0 genere valores más extremos del observado. Por lo tanto, terminamos rechazando H_0.
- R te pone símbolos a tus valores p si llegan a cierto rango de "pequeño". Por ejemplo, para valores p en $(0.01, 0.05]$ te pone un *.

Definiciones

- Cuando rechazamos una H_0 en realidad no hemos probado que H_0 está mal o que H_A está bien.
- Una vez que especificamos un nivel de significancia, podemos encontrar la región de rechazo y los valores críticos.
- Podemos cometer errores:
  - Si rechazamos H_0 cuando era cierta, es un error tipo I.
  - Si H_0 es falsa y la aceptamos, cometemos un error tipo II.

[4] Es generada por un experimento que reemplaza a nuestra evidencia

Diferencia de proporciones

- A veces conocemos una proporción para cierta variable. Luego medimos esta proporción para alguna muestra y queremos saber si es diferente a la conocida.
- Podemos usar como H_A que $p < p_0$, $p > p_0$ o $p \neq p_0$. Osea, podemos usar la cola izquierda, la derecha o las dos.
- Podemos hacer esta prueba con prop.test.

Example

En USA, 11.3% era pobre en el 2000 de acuerdo a un censo. Para el 2001 estimaron que el 11.7% era pobre. Digamos que el tamaño de su muestra fue de 50 000 personas. Queremos investigar si el valor de 11.7% representa un incremento entre el año 2000 y 2001. Usen `prop.test`.

Resolviendo el ejemplo

- Para resolver el ejemplo, usamos $H_0 : p = 0.113$ y $H_A : p > 0.113$
- Como es de solo una cola (la derecha), obtenemos el resultado así:

```
> xx <- prop.test(x = 0.117 * 50000,
+      n = 50000, p = 0.113, alt = "greater")
> xx$p.value
```

```
[1] 0.002415415
```

- Nuestro valor p es muy significativo. Por lo que rechazamos H_0 en favor de H_A.
- NOTA: hagan `attributes(xx)` para aprender más de la función `prop.test`.

Para la media

- Si asumen que sus datos se distribuyen como normal y no están fuertemente sesgados[5], con la función t.test podemos probar la $H_0 : \mu = \mu_0$ vs $H_A : \mu < \mu_0$, $\mu > \mu_0$ o $\mu \neq \mu_0$
- Por ejemplo, sospechan que su nueva SUV no da el kilometraje anunciado de 17 km por litro. Llenan su tanque 10 veces y obtienen los siguientes valores: 11.4, 13.1, 14.7, 15, 15.5, 15.6, 15.9, 16, 16.8. Hacemos una `t.test`:

```
> kpl <- c(11.4, 13.1, 14.7, 15,
+     15.5, 15.6, 15.9, 16, 16.8)
> xx <- t.test(kpl, mu = 17, alt = "less")
> xx$p.value
```

Para la media


```
[1] 0.002614081
```

- Dado que el valor p es significativo, rechazamos H_0 en favor de H_A.
- Si se fijaron, estamos usando las mismas funciones para obtener los intervalos de confianza y hacer las pruebas de hipótesis. Es que estamos usando la misma estadística para los dos.

[5] Grafiquen sus datos previamente

Prueba del signo

- Si sus datos no se distribuyen como normal (o cerca de), podemos hacer unas pruebas no paramétricas con la mediana.
- Por ejemplo, podemos hacer la prueba de signo donde solo asumimos que la distribución es contínua y positiva usando `sum` y `pbinom`. Nuestra hipótesis son:
  - H_0: mediana $= m$
  - H_A: mediana $< m$, mediana $> m$ o mediana $\neq m$
- Un ejemplo. Digamos que hicimos llamadas de 2, 1, 3, 3, 3, 3, 1, 3, 16, 2, 2, 12, 20, 3 y 1 minutos. Tenemos H_0: mediana $= 5$ y H_A: mediana < 5.

Prueba del signo


```
> llamadas <- c(2, 1, 3, 3, 3, 3,
+     1, 3, 16, 2, 2, 12, 20, 3,
+     1)
> obs <- sum(llamadas > 5)
> n <- length(llamadas)
> 1 - pbinom(n - obs - 1, n, 1/2)
```

```
[1] 0.01757812
```

- Rechazamos H_0 con $\alpha = 0.05$. Para dos colas podríamos hacer:

```
> k <- max(obs, n - obs)
> 2 * (1 - pbinom(k - 1, n, 1/2))
```

```
[1] 0.03515625
```

Wilcoxon signo ordenado

- Si sus datos son símetricos y contínuos pueden usar esta prueba con wilcox.test.

Example

Usemos `salmon.rate` de `UsingR`. Hagan una gráfica para ver la distribución. Luego hagan una con los valores log. Tenemos H_0: mediana $=$ log(0.005) y H_A: mediana $>$ log(0.005). Qué concluyen?

```
> xx <- wilcox.test(log(salmon.rate),
+      mu = log(0.005), alt = "greater")
> xx$p.value

[1] 0.06499583
```

Dos proporciones

- Estábamos comparando un estimado de un parametro vs el poblacional. Ahora si queremos comparar dos parametros como podrían ser dos prociones podemos usar `prop.test`.
- Vamos a tener H_0: $p_1 = p_2$ y H_A: $p_1 < p_2$, $p_1 > p_2$ ó $p_1 \neq p_2$.
- Siguiendo el ejemplo de la pobreza, digamos que tenemos para el 2002 un porcentaje de 12.1% de pobres con una muestra de 60 000.

```
> phat <- c(0.117, 0.121)
> n <- c(50000, 60000)
> xx <- prop.test(n * phat, n, alt = "less")
> xx$p.value
```

```
[1] 0.0212056
```

Pruebas de centro

- Al igual que comparamos proporciones, ahora podemos comparar medidas centrales de dos poblaciones. Si asumimos:
  - que las 2 son independientes y normales podemos usar una prueba t.
  - si no están distribuidas como normales podemos usar la Wilcoxon.
  - si las muestras no son independientes y están pareadas de alguna forma podemos usar una prueba de pares.
- Tendremos H_0: $\mu_x = \mu_y$ y H_A $\mu_x < \mu_y$, $\mu_x > \mu_y$ ó $\mu_x \neq \mu_y$.
- Básicamente usamos `t.test` con diferentes valores para los argumentos `paired` y `var.equal` ó `wilcox.test`. La diferencia principal es que ahora usamos los argumentos x y y en vez de solo x.

Más estadística!

- Bueno, ahora regresamos a los datos categóricos. Ya no vamos a usar tablas de contingencia como vimos anteriormente.
- Este tipo de pruebas nos dicen qué tan bien se asemejan nuestros datos observados a unos valores esperados. Por eso llevan el nombre de "Bondad de Ajuste".

La famosa χ^2

- Primero necesitamos tener k categorías de datos donde la suma de sus probabilidades sea igual a 1; por ejemplo, $1/k$. Podemos escoger una categoría con esas probabilidades lo cual nos da un Y_i. Esto lo hacemos n veces, las cuales se distribuyen conjuntamente como multinomial.
- No es fácil usar la multinomial directamente para encontrar un valor p, pues nuestras variables Y_i van a estar correlacionadas.

La famosa χ^2

- Lo que sí podemos hacer es comparar el valor observado contra el esperado y luego normalizar de alguna forma para obtener algo que se parezca a una distribución conocida como la estadística χ cuadrada de Pearson:

$$\chi^2 = \sum_{i=1}^{k} \frac{(Y_i - np_i)^2}{np_i} = \sum \frac{(observado - esperado)^2}{esperado}$$

- Tendremos H_0: $p_1 = \pi_1, ..., p_k = \pi_k$ y H_A: $p_i \neq \pi_i$ para al menos i. Esta prueba la podemos hacer con la función chisq.test.
- También podemos hacer comparaciones entre solo dos categorías, pueden hacer la prueba "manualmente" o fijarse bien en los grados de libertad.

Un ejemplo

- Usemos `samhba` de `UsingR`. Queremos checar si las proporciones observadas (donde 1 es que fuma diario y 7 que nunca) están de acuerdo con nuestras probabilidades $p_1 = .15$, $p_2 = .05$, $p_3 = .05$, $p_4 = .05$, $p_5 = .1$, $p_6 = .2$, $p_7 = .4$

```
> y <- table(samhda$amt.smoke[samhda$amt.smoke <
+     98])
> p <- c(0.15, 0.05, 0.05, 0.05,
+     0.1, 0.2, 0.4)
> xx <- chisq.test(y, p = p)
> xx$p.value
```

```
[1] 0.2426627
```

Independencia

- Si luego quieren hacer una prueba de independencia para dos variables categóricas, pueden usar la χ^2.
- Tienen que encontrar la tabla de contingencia para sus datos y usar:
  - `chisq.test(x)` si `x` es una matriz o una tabla
  - `chisq.test(x,y)` si tienen las variables separadas y los valores x_i y y_i son del mismo individuo i.
  - O pueden resumir sus datos y usarlos así: `chisq.test(table(x,y))`.
- H_0 será que las 2 variables son independientes. H_A que sean dependientes.
- Pueden usar el argumento `simulate.p.value=TRUE` si quieren estimar su valor p con una simulación de Monte Carlo. Es útil en el caso de que sus valores esperados en frecuencia absoluta sean muy pequeños.

Prueba de homogeneidad

- Aunque H_0 sea que las 2 variables son iguales y H_A que son diferentes, en realidad se encuentra el valor p de la misma forma que en la prueba de independencia.

Para distribuciones contínuas

- Digamos que obtenemos n muestras X^6 de una distribución contínua. Estas nos dan una distribución empírica con la cual podemos encontrar la probabilidad de que una muestra tenga un valor igual a x es $F_n(x) = \frac{i:X_i \leq x}{n}$.
- Podemos encontrar esta probabilidad usando la función ecdf.
- Tendremos H_0: $F(x) = F_0(x)$ y H_A: $F(x) \neq F_0(x)$. Podríamos usar una χ^2 pero su desempeño es pobre para esto, así que usamos una Kolmogorov-Smirnov donde $D =$ máximo en x de $|F_x(x) - F(x)|$
- En `R` hacemos esta prueba con ks.test(x, y="name", ...) donde `"name"` es el nombre de la función que regresa la función de probabilidad acumulada. Por ejemplo, `pnorm`.

Para distribuciones contínuas

- Si tenemos muestras X y Y de dos distribuciones contínuas, podemos utilizar `ks.test(x,y)`.

[6] sin empates

Shapiro

- La Kolmogorov-Smirnov funciona para datos univariados cuando la hipótesis nula es especificada antes de ver a los datos. Ninguna asunción de los parámetros debe depender de los datos ya que esto arruina a la prueba (cambia la distribución de muestro).
- Con la prueba Shapiro-Wilk podemos ver la distribución padre de nuestra muestra es normal.
- Tendremos H_0: distribución padre es normal vs H_A: la distribución padre no es normal.
- En `R` se usa con `shapiro.test(x)`.
- En el caso de que falle, todavía pueden usar la prueba t si su n no es muy grande. En estos caso, la t es más resistente a pequeños cambios en las asunciones de la distribución padre.

Una función útil!

- El paquete `MASS` tiene una función llamada fitdistr. Con esta podemos estimar parámetros de una muestra después de graficarlos.
- La idea es que con la gráfica puedes inferir un poco el tipo de distribución. `fitdistr` te regresa valores estimados con errores estándar los cuales te pueden servir para construir intervalos de confianza.
- Les recomiendo que hagan los siguientes comandos:

```
> library(MASS)
> `?`(fitdistr)
```

ANOVAs

Funciones

- oneway.test
- aov
- kruskal.test(x ∼ f, data=..., subset=...): para igualdad de medias. No paramétrica.
- lm junto con summary es un tipo de ANOVA.
- De forma parecida a la anterior o con anova pueden hacer ANCOVAs.
- Para ANOVAs de dos sentidos usen anova
- interaction.plot les puede servir para probar interacciones.

Un ejemplo

- Acuérdense de que las ANOVAs son una generalización de la prueba t.
- En fin, escogimos a 15 individuos y los separamos al azar en 3 grupos; 1 mes por grupo. Luego medimos cuantas calorías consumieron un día al azar los individuos de cada grupo.
- Queremos saber si las diferencias observadas se deben a una variación natural en nuestro muestreo o a una diferencia real entre las poblaciones originales. Usemos oneway.test.

Un ejemplo


```
> mayo <- c(2166, 1568, 2233, 1882,
+     2019)
> sep <- c(2279, 2075, 2131, 2009,
+     1793)
> dic <- c(2226, 2154, 2583, 2010,
+     2190)
> d <- stack(list(mayo = mayo, sep = sep,
+     dic = dic))
> oneway.test(values ~ ind, data = d,
+     var.equal = TRUE)
```

Un ejemplo


```
One-way analysis of means

data:  values and ind
F = 1.7862, num df = 2, denom df =
12, p-value = 0.2094
```

- El valor p no es significativo, por lo que las diferencias observadas se pueden explicar por simple variación en las muestras.

Que es

- Unos me han preguntado como hice las presentaciones. Bueno, en `R` hay una función que te pasa archivos a formato `LaTeK`. Una vez que tienes tu archivo en ese formato, puedes crear tus PDFs.
- La idea de los creadores fue unificar la forma en que se reportan los trabajos realizados en `R` y ha tenido tal impacto que practicamente todos los manuales de ayuda y reportes nuevos salen con ese formato.
- Empecemos con nuestro archivo `.Rnw`

Estructura básica

Un archivo básico tiene las siguientes cuatro etiquetas:

- \documentclass
- \usepackage
- \begin{document}
- \end{document}

Además, pueden usar el `begin` y `end` para:

- poner una ecuación con `equation`.
- poner alguna fórmula con `displaymath`.
- poner un listado con `itemize`.
- poner una enumeración con `enumerate`.

Los comandos!

- Ya con esta estructura básica, podemos hacer nuestros reportes en PDF.
- Bajen el archivo templado.Rnw a un folder vacio.
- Ahora corran los siguientes comandos (en orden):
  - `R CMD Sweave templado.Rnw`
  - `R CMD pdflatex templado.tex`
- Ya tienen su PDF listo! Chequen su folder que estaba vacio :P Por algo no es recomendable tener más de un archivo `.Rnw` en el mismo folder.
- Pueden cambiar el comando `Sweave` por `Stangle` para obtener un archivo `templado.R` que tiene el código `R` que usaron en su reporte.

Los comandos!

- Si quieren bajar algún archivo `.Rnw` de los que hice, usen la misma ruta para los PDFs pero cambien la extensión final. Yo aprendí comparando los `.Rnw` de James Bullard y sus presentaciones PDF. En el material de apoyo del curso hay documentos muy buenos para aprender más al respecto :).
- Disfrutenlo!
- Bueno, ahora hagan los Ejercicios 4. Suerte!