

Comandos R - Práctica 6-7

Paquete PASWR2

Muchas de las funciones utilizadas en esta práctica están dentro de la librería PASWR2. Por tanto, deberemos siempre cargar la librería antes de comenzar.

```
library(PASWR2)
```

Estudio de la normalidad de los datos

Como herramienta para estudiar la normalidad de los datos podemos usar un QQ-plot. Si asumimos que nuestra muestra está en un objeto llamado `datos`:

```
qqnorm(datos)
qqline(datos)
```

Para poder asumir normalidad, deberemos comprobar que los puntos que representan los cuantiles de la muestra, se ajustan aproximadamente a la línea que representa los cuantiles de una distribución normal teórica.

Otra alternativa para estudiar la normalidad y que es más fácil de interpretar es realizar un test de normalidad de Shapiro-Wilk. Las hipótesis que establece el test son:

- H_0 : los datos de la muestra provienen de una población que sigue una distribución normal
- H_1 : los datos de la muestra provienen de una población que NO sigue una distribución normal

```
shapiro.test(datos)
```

Como resultado del comando obtenemos un p-valor que podemos interpretar para rechazar o no la hipótesis nula y por tanto para decidir si podemos o no asumir normalidad. Si no rechazamos H_0 asumiremos normalidad y si rechazamos H_0 no podremos asumir normalidad.

Alternativamente podemos usar la función `eda()` que nos da información sobre el QQ-plot y sobre el test de Shapiro-Wilk:

```
eda(datos)
```

Cálculo de la potencia

Cuando trabajamos con un estadístico de contraste que sigue una distribución de probabilidad t , podemos usar el comando `power.t.test` de R para calcular la potencia. A continuación vemos distintos valores de los parámetros que podemos usar:

- **type:**
 - Para test de una muestra `type = "one.sample"`
 - Para test de dos muestras `type = "two.sample"`
 - Para test de muestras dependientes `type = "paired"`

En el caso de calcular la potencia con dos muestras (`type = "two.sample"`), es decir para el contraste para la diferencia de medias, las dos muestras tienen que tener el mismo tamaño y además las dos poblaciones deben tener la misma varianza. Si tienen diferente tamaño y/o distinta varianza, no podemos usar `power.t.test` para obtener la potencia.

- **alternative** (valor por defecto `"two.sided"`)
 - Contraste bilateral (Desigualdad en H1) `alternative = "two.sided"`
 - Contraste unilateral (`<` ó `>` en H1) `alternative = "one.sided"`
- **sig.level:** nivel de significación (por defecto es 0.05)
- **delta:** diferencia (en valor absoluto) entre el valor del parámetro (media o diferencia de medias) bajo la hipótesis alternativa menos el valor del parámetro bajo la hipótesis nula.
- **sd:** desviación típica, σ_X en el caso que `type = "one.sample"` o $\sqrt{\sigma_X^2 + \sigma_Y^2}$ en el caso que `type = "two.sample"`. En problemas reales, es habitual que no se tenga información sobre las varianzas poblacionales para el cálculo de la potencia. En esos casos, podremos usar las varianzas de la muestra como estimadores de las varianzas poblacionales (s_X en el caso que `type = "one.sample"` o $\sqrt{s_X^2 + s_Y^2}$ en el caso que `type = "two.sample"`).

Teniendo en cuenta esos parámetros podemos calcular la potencia de test:

```
power.t.test(n=tamaño.muestra, delta=numerador.del.parametro.de.no-centralidad,
             sd=valor.de.sigma, sig.level=0.05, type="one.sample",
             alternative="one.sided", strict=TRUE)
```

También podemos usar R para obtener el tamaño de muestra necesario para tener una potencia dada:

```
power.t.test(power=valor.de.la.potencia, delta=numerador.del.parametro.de.no-centralidad,
             sd=valor.de.sigma, sig.level=0.05, type="one.sample",
             alternative="one.sided", strict=TRUE)
```

COMANDOS PARA REALIZAR LOS CONTRASTES

Para poder utilizar los comandos que se describen en esta sección, deberemos poder asumir que la población o poblaciones siguen una distribución Normal.

Hay ciertos parámetros que son comunes a todos los comandos que usaremos para resolver los tests. Por ejemplo el parámetro **alternative** con el que se determina el tipo de hipótesis alternativa:

- Contraste bilateral \rightarrow `alternative = "two.sided"`
- Contraste unilateral izquierdo (`<`) \rightarrow `alternative = "less"`
- Contraste unilateral derecho (`>`) \rightarrow `alternative = "greater"`

El parámetro `conf.level` realmente se usa sólo para construir el intervalo de confianza, para realizar el contraste no es necesario indicarlo.

NOTA: los comandos `zsum.test` o `tsum.test` se emplean cuando los datos de la muestra vienen en forma de resumen. Por ejemplo, si nos dan el valor de la media muestral y de la desviación típica de la población podremos usar el `zsum.test` o si nos dan el valor de la media muestral y de la desviación típica de la muestra podremos usar el `tsum.test`. Por otra parte, `z.test` o `t.test` se emplean cuando tenemos el conjunto de datos que compone la muestra.

CONTRASTE PARA LA MEDIA (SIGMA CONOCIDA)

```
## Si tenemos los valores resumidos de la muestra
zsum.test(mean.x=media.muestral, sigma.x=desviacion.estandar.poblacional, n.x = tamaño.muestra,
alternative = "two.sided", mu = valor.de.mu.en.H0, conf.level = 0.95)

## Si tenemos el conjunto de datos que compone la muestra
z.test(x=datos, sigma.x = sigma.poblacional ,alternative = "two.sided",
mu = valor.de.mu.en.H0, conf.level = 0.95)
```

CONTRASTE PARA LA MEDIA (SIGMA DESCONOCIDA)

```
## Si tenemos valores "resumidos" de la muestra
tsum.test(mean.x=media.muestral, s.x=desviacion.estandar.muestral, n.x = tamaño.muestra,
alternative = "two.sided", mu = Valor.de.mu.en.H0, conf.level = 0.95)

## si tenemos el conjunto de datos que compone la muestra
t.test(x=datos , alternative = "two.sided", mu = valor.de.mu.en.H0, conf.level = 0.95)
```

CONTRASTE PARA LA DIFERENCIA DE MEDIAS (SIGMAS CONOCIDAS)

```
## valores "resumidos" de la muestra
zsum.test(mean.x=media.muestral.x , sigma.x = sigma.poblacional.x, n.x = tamaño.muestra.x,
mean.y=media.muestral.y ,sigma.y = sigma.poblacional.y, n.y = tamaño.muestra.y,
alternative = "two.sided", mu = En.Ho.valor.de.la.resta.de.medias, conf.level = 0.95)

## tenemos el conjunto de datos que compone la muestra
z.test(x=datos.x, y=datos.y, sigma.x=sigma.poblacional.x, sigma.y=sigma.poblacional.y,
alternative = "two.sided",mu = Valor.en.Ho.para.la.resta.de.medias ,conf.level = 0.95)
```

DIFERENCIA DE MEDIAS (SIGMAS DESCONOCIDAS)

En el caso que las sigmas sean desconocidas e iguales usaremos `var.equal = TRUE`. Si por el contrario las sigmas son desconocidas y distintas, usaremos `var.equal = FALSE`

```

## valores "resumidos" de la muestra
tsum.test(mean.x=media.muestra.x , s.x = desviacion.estandar.muestra.x, n.x = tamaño.muestra.x,
mean.y = media.muestra.y , s.y = desviacion.estandar.muestra.y, n.y = tamaño.muestra.y,
alternative = "two.sided", mu = Valor.en.Ho.para.la.resta.de.medias, var.equal = TRUE,
conf.level = 0.95)

## tenemos el conjunto de datos que compone la muestra
t.test(x=datos.x, y=datos.y, var.equal=TRUE, conf.level = 0.95,
alternative = "two.sided", mu = Valor.en.Ho.para.la.resta.de.medias)

```

DIFERENCIA DE MEDIAS (DATOS RELACIONADOS)

```

t.test(x=datos.x, y=datos.y, paired=TRUE, conf.level = 0.95,
alternative = "two.sided", mu = Valor.en.Ho.para.la.resta.de.medias)

```

COCIENTE DE VARIANZAS

```

var.test(x=datos.x, y=datos.y, ratio = Valor.en.Ho.para.el.cociente.de.varianzas,
alternative="two.sided", conf.level = 0.95)

```

En el caso del `var.test`, habitualmente `ratio = 1` (Nos permite decidir si asumimos o no que las varianzas son iguales).

PROPORCION POBLACIONAL

Usando la aproximación mediante la normal basada en muestras grandes. En este caso, dado que estamos aproximando la distribución de probabilidad del estadístico mediante una normal, podemos usar una corrección (o no) para que esta aproximación sea más precisa. El test sin corrección corresponde al que hemos visto en clase.

```

#### Sin corregir
prop.test(x=Casos.favorables, n=tamaño.muestral, conf.level = 0.95, correct=FALSE,
p = Valor.de.pi.en.Ho, alternative = "two.sided")

#### Corregido
prop.test(x=Casos.favorables, n=tamaño.muestral, conf.level = 0.95, correct=TRUE,
p = Valor.de.pi.en.Ho, alternative = "two.sided")

```

Cuidado porque en los resultados que devuelve `prop.test`, el valor que nos da como el observado, no corresponde a z_{obs} sino a z_{obs}^2 . Por lo tanto tenemos que hacer la raíz cuadrada para obtener el z_{obs} . Se debe tener cuidado con el signo cuando tenemos una hipótesis unilateral ya que al hacer la raíz cuadrada, el resultado puede ser positivo o negativo. Para identificarlo nos tendremos que fijar en el signo de $(p - \pi_0)$.

Si el tamaño de muestra no es suficientemente grande, no podemos realizar la aproximación mediante la normal. En esos casos podemos hacer el test usando una distribución binomial:

```
binom.test(x=Casos.favorables, n=tamaño.muestral, p = Valor.de.pi.en.Ho, alternative = "two.sided",
           conf.level = 0.95)
```

TEST DE IGUALDAD DE PROPORCIONES (2 POBLACIONES INDEPENDIENTES)

Podemos usar el comando `prop.test` para resolver el test de hipótesis de la diferencia de proporciones. Al igual que antes, podremos aplicar corrección o no.

```
#### Sin corregir
prop.test(x=c(Casos.favorables.X, Casos.favorables.Y), n=c(tamaño.muestraX, tamaño.muestraY),
          conf.level = 0.95, correct=FALSE, alternative = "two.sided")

#### Corregido
prop.test(x=c(Casos.favorables.X, Casos.favorables.Y), n=c(tamaño.muestraX, tamaño.muestraY),
          conf.level = 0.95, correct=TRUE, alternative = "two.sided")
```

Nuevamente, en los resultados que devuelve `prop.test`, el valor que nos da como el observado, no corresponde a z_{obs} sino a z_{obs}^2 . Por lo tanto tenemos que hacer la raíz cuadrada para obtener el z_{obs} . Se debe tener cuidado con el signo cuando tenemos una hipótesis unilateral ya que al hacer la raíz cuadrada, el resultado puede ser positivo o negativo. Para identificarlo nos tendremos que fijar en el signo de $(p_x - p_y) - \delta_0$ donde δ_0 es el valor de $\pi_x - \pi_y$ bajo la hipótesis nula.