Introducción al Equating

Simulación Estadística

Kevin Steven Garcia Chica Cod. 1533173
Cesar Andres Saavedra Vanegas Cod. 1628466

Universidad Del Valle
Facultad De Ingenieria
Estadistica
Febrero
2018

EQUATING

Equating (equiparando) de forma general, es un proceso estadístico cuyo objetivo o finalidad es ajustar los puntajes de dos formas distintas de una misma prueba; con esto, se busca relacionar el puntaje de una forma de una prueba y su equivalente en la otra forma con la cual se quiere comparar o equiparar, en otras palabras, lo que se busca aplicando Equating es que los puntajes en los formularios de prueba se puedan usar indistintamente.

Debemos tener en cuenta que condiciones o supuestos se debe cumplir para poder aplicar Equating. Para comparar dos formas diferentes de un test se deben cumplir básicamente los siguientes 5 supuestos:

- Simetría: Esta propiedad nos dice que la función utilizada para transformar una puntuación en la Forma X a la escala de la Forma Y sea la inversa de la función utilizada para transformar un puntaje en la Forma Y a la escala de la forma X. Por ejemplo, esta propiedad implica que si un puntaje bruto de 85 en la Forma X se convierte a un puntaje bruto de 90 en la Forma Y, luego un puntaje bruto de 90 en la Forma Y debe convertirse a un puntaje bruto de 85 en la forma X.
- Igual o cercana confiabilidad.
- Equidad: Lord (1980) definió esta propiedad específicamente. La propiedad es válida si los examinados con un puntaje verdadero dado tienen la misma distribución de puntajes en el Formulario X como lo harían en el Formulario Y.
  En nuestras palabras, esto quiere decir que la media, y la desviación estándar en las dos formas del test deben ser aproximadamente iguales. Por ejemplo, si vemos que la mayoría de examinados expuestos a la forma X del test tienen puntajes mucho mas altos que los expuestos a la forma Y, entonces esta propiedad no se cumpliría, ya que las medias van a ser muy diferentes (la media de los puntajes de la forma X va a ser mucho mayor que la de los puntajes de la forma Y).
- Invarianza poblacional.
- Igual constructo: Esta condición nos dice que ambas formas del test deben medir el mismo constructo o las mismas características.

En general, las situaciones en las que se requiere el uso de Equating son en la aplicación de distintas formas de una misma prueba o test. Por ejemplo, en un examen de ingreso a estudios superiores en el que se convoca a los aspirantes para distintas fechas resulta extremadamente conveniente disponer de formas alternativas de la prueba o del examen, por razones estrictamente de seguridad (evitar plagio o conocimiento previo de la prueba). También es necesario disponer de distintas formas de un test cuando se desea medir en repetidas ocasiones a un mismo individuo o colectivo con el fin de evaluar, por ejemplo, su progreso académico o un posible cambio en sus actitudes. En cualquiera de estos casos, para poder comparar las puntuaciones obtenidas en las distintas formas del test es necesario ponerlas previamente en la misma escala, y eso lo logramos mediante el uso adecuado del Equating.

En cuanto al problema matemático o estadístico como tal en la equiparación ó el Equating consiste en modelar la relación entre un puntaje en un formulario de prueba y su puntaje correspondiente en otra forma. Matemáticamente, esto significa que se debe definir una función que tome valores en X y da como resultado un valor en Y.
Un ejemplo de esto, es cuando convertimos grados celsius a grados farenheit, la función en este caso es:

$$F = (\frac{9}{5} * C) + 32$$

Esto significa que 0^oC equivalen a 32^oF. Exactamente lo mismo busca el Equating, hallar una función que dado un valor o un puntaje de la forma X nos arroje un puntaje en la escala de la forma Y.

Considerando solo dos tipos de pruebas o dos formas del test (debemos tener en cuenta que no necesariamente son dos formas del test, pueden ser muchas más), X y Y, supongamos que la prueba X se le aplica a n_x personas y la prueba Y se le aplica a n_y personas al azar, los puntajes obtenidos de la forma X y Y de los test se pueden ubicar en un vector X_i $(i = 1, ..., n_x)$ y Y_j $(j = 1, ..., n_y)$; además, como X y Y son resultados de cada uno de los test, estos pueden tomar cualquier valor en la escala del test, por lo cuál, consideramos X y Y como variables aleatorias, y estas a su vez, por ser variables aleatorias, se les pude asignar o siguen una distribución ya sea discreta o continua, F_x y F_y.

Para cumplir con el objetivo del Equating (transformar los puntajes X en la escala de Y), se trabaja con la función Equating:

$$\varphi_Y(x) = F_Y^{-1}(F_X(x))$$

Una de la dificultades mas importantes que se tienen al utilizar o aplicar Equating es que los puntajes de las formas X y Y de un test, suelen ser discretos (casi siempre se trabaja con número total de preguntas buenas), esto implica que la distribución de los puntajes F_X y F_Y de cada una de las formas no tenga inversa o sean muy difícil de encontrar. Este problema se suele resolver aproximando esas distribuciones discretas a continuas, por medio de los métodos de continuidad para poder utilizar la función Equating (note que necesitamos la inversa de la distribución de Y).

Ejemplo de Equating con R:
Para ejemplificar el método del Equating se siguió la metodología implantada por Anthony Albano en la cual se hace uso de la libreria .equate" que este provee para el software R en el articulo publicado el 12 de enero del 2017 y en el cual se describe los metodos.

- **Identity, Mean, Linear, Equipercentile With loglinear, Composite of mean and identity**

  Por medio de una base de datos proporcionada por Antnthony Albano se procedio a realizar el análisis de Equating por los 5 métodos anteriormente mencionados y se obtuvieron los siguientes resultados, en los cuales se evidencia la conversión del puntaje tras aplicar los métodos de equating y en para los que se asume un margen de error muy bajo al realizar la transformación .

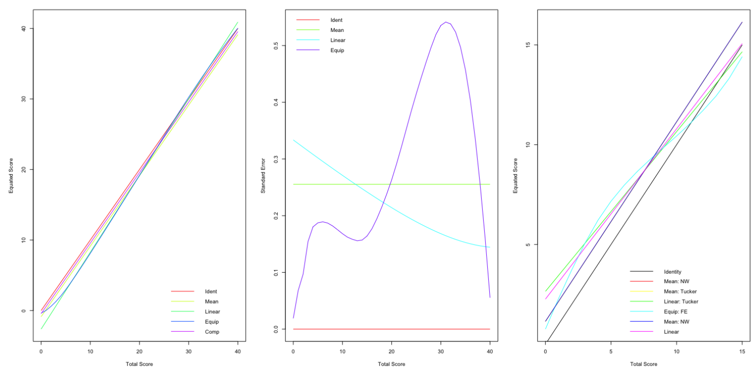

Figura 1: Equating para puntajes de una prueba por distintos métodos

Referencias

- Inés María Varas Cáceres, 2018. Taller: Introducción al Equating.
- Jorge González, Marie Wiberg, 2017. Applying Test Equating Methods Using R.
- Michael J. Kalen, Robert L. Brennan,2004 .Test Equating, Scaling,and Linking Methods and Practices, Second Edition.
- Fang Chen, Xiaomin Huang y David MacGregor, 2009. EQUATING OR LINKING: BASIC CONCEPTS AND A CASE STUDY.
- Lady Catheryne Lancheros Florian, 2013, Universidad Nacional de Colombia. MÉTODOS DE EQUIPARACIÓN DE PUNTUACIONES: LOS EXÁMENES DE ESTADO EN POBLACIÓN CON Y SIN LIMITACIÓN VISUAL.
- Neil J. Dorans, Tim P. Moses, and Daniel R. Eignor, 2010. Principles and Practices of Test Score Equating.
- Navas, M. J. ,2000. Equiparación de puntuaciones: Exigencias actuales y retos de cara al futuro.
- Jorge González, Facultad de Matemáticas, Pontificia Universidad Católica de Chile, 2017. Equating: Una breve introducción.
- Constanza Rojo Alfaro, Jorge González Burgos, 2012. EL MÉTODO KERNEL DE EQUATING Y SU CONTRAPARTE BAYESIANA NO PARAMÉTRICA: UN ESTUDIO DE COMPARACIÓN BAJO EL DISEÑO DE GRUPOS EQUIVALENTES.
- Anthony Albano, 2012, Package equate, Observed-Score Linking and Equating.

1. Scripts

```
#--------------------------------------------------------------#

#Kevin Steven Garcia Chica - 1533173
#Cesar A. Saavedra Vanegas - 1628466
#Simulacion Estadistica
#Tarea 1
#Equating

#--------------------------------------------------------------#

library(equate)
# See vignette("equatevignette") and Albano (2016) for a
# description of methods and additional examples
# Random groups equating for (1) identity, (2) mean,
# (3) linear, (4) equipercentile with loglinear
# smoothing, and (5) a composite of mean and identity
rx <- as.freqtab(ACTmath[, 1:2])
ry <- as.freqtab(ACTmath[, c(1, 3)])
set.seed(2007)
req1 <- equate(rx, ry, type = "i", boot = TRUE, reps = 5)
req2 <- equate(rx, ry, type = "m", boot = TRUE, reps = 5)
req3 <- equate(rx, ry, type = "l", boot = TRUE, reps = 5)
req4 <- equate(rx, ry, type = "e", boot = TRUE, reps = 5,
               smooth = "loglin", degree = 3)
req5 <- composite(list(req1, req2), wc = .5, symmetric = TRUE)
# Compare equating functions
plot(req1, req2, req3, req4, req5[[1]], addident = FALSE)
# Compare boostrap standard errors
# Bootstrapping isn't supported for composite equating
plot(req1, req2, req3, req4, addident = FALSE, out = "se",
     legendplace = "topleft")
# Nonequivalent groups design for (1) Tucker linear,
# (2) frequency estimation , and (3) Braun/Holland linear
nx <- freqtab(KBneat$x, scales = list(0:36, 0:12))
ny <- freqtab(KBneat$y, scales = list(0:36, 0:12))
neq1 <- equate(nx, ny, type = "linear", method = "tuck", ws = 1)
neq2 <- equate(nx, ny, type = "equip", method = "freq", ws = 1)
neq3 <- equate(nx, ny, type = "linear", method = "braun", ws = 1)
# Compare equated scores
round(cbind(xscale = 0:36, tucker = neq1$conc$yx,
            fe = neq2$conc$yx, braun = neq3$conc$yx), 2)
# Multiple linkings using PISA reading booklet 6
# clusters 3a, 5, 6, and 7
r3 <- freqtab(PISA$totals$b6$r3a, scales = 0:15)
```

```
r5 <- freqtab(PISA$totals$b6$r5, scales = 0:15)
r6 <- freqtab(PISA$totals$b6$r6, scales = 0:15)
r7 <- freqtab(PISA$totals$b6$r7, scales = 0:14)
eqargs <- list(r3r5 = list(type = "linear", x = "r3", y = "r5",
                           name = "Linear Linking PISA r3 to r5"),
               r5r6 = list(type = "linear", x = "r5", y = "r6",
                           name = "Linear Linking PISA r5 to r6"),
               r6r7 = list(type = "linear", x = "r6", y = "r7",
                           name = "Linear Linking PISA r5 to r7"))
req <- equate(list(r3 = r3, r5 = r5, r6 = r6, r7 = r7), eqargs)
# Put PISA r3 on the scale of r7 using the linking chain
# Compare to a direct linking of r3 to r7
equate(equate(req$r3r5$conc$yx, req$r5r6), req$r6r7)
equate(r3, r7, "linear")$conc$yx
# Linking PISA cluster r3a to r5 with multiple anchors
m367 <- freqtab(PISA$totals$b6[1:198, c("r3a", "r6", "r7")],
                scales = list(0:15, 0:16, 0:14))
m567 <- freqtab(PISA$totals$b6[199:396, c("r5", "r6", "r7")],
                scales = list(0:15, 0:16, 0:14))
meq1 <- equate(m367, m567, type = "mean", method = "nom")
meq2 <- equate(m367, m567, type = "mean", method = "tuck")
meq3 <- equate(m367, m567, type = "lin", method = "tuck")
meq4 <- equate(m367, m567, type = "equip", method = "freq",
               smooth = "log", show = FALSE)
meq <- equate(m367, m567, type = "mean", method = "nom")
plot(meq1, meq2, meq3, meq4, meq, req[[1]])

#-----------------Graficos---------------------------------------------#
par(mfrow=c(1,3))
plot(req1, req2, req3, req4, req5[[1]], addident = FALSE)
plot(req1, req2, req3, req4, addident = FALSE, out = "se",
     legendplace = "topleft")
plot(meq1, meq2, meq3, meq4, meq, req[[1]])
```