UNIVERSIDAD DE COSTA RICA
SISTEMA DE ESTUDIOS DE POSGRADO

EXPLORACIÓN DE LA IDONEIDAD DE LAS ALTERNATIVAS DE RESPUESTA EN ÍTEMS DE ELECCIÓN ÚNICA DE PRUEBAS ESTANDARIZADAS

Trabajo final de investigación aplicada sometido a la consideración de la Comisión del Programa de Estudios de Posgrado en Estadística para optar por el grado y título de Maestría Profesional en Estadística

LETICIA VÁSQUEZ GUTIÉRREZ

Ciudad Universitaria Rodrigo Facio, Costa Rica

2022

Agradecimientos

Agradezco a Dios por darme la oportunidad de llegar hasta acá; a mis padres, Viviana Gutiérrez y Fernando Vásquez, por ser mi inspiración en todo momento y a toda mi familia, en general, por el apoyo que me han dado a lo largo de este proceso. A mis compañeros y amigos de la carrera, por hacer más amena esta etapa. A los profesores que nos han compartido no solo su conocimiento sino también su calidad humana. A mi profesor tutor Guaner, por su paciencia y apoyo en la finalización de este trabajo.

"Este trabajo final de investigación aplicada fue aceptado por la Comisión del Programa de Estudios de Posgrado en Estadística de la Universidad de Costa Rica, como requisito parcial para optar al grado y título de Maestría Profesional en Estadística."

GUANER DAVID ROJAS ROJAS (FIRMA)
Firmado digitalmente por GUANER DAVID ROJAS ROJAS (FIRMA)
Fecha: 2022.06.02 15:25:51 -06'00'

Dr. Guaner Rojas Rojas

Profesor Guía

EILIANA MONTERO ROJAS (FIRMA)
Firmado digitalmente por EILIANA MONTERO ROJAS (FIRMA)
Fecha: 2022.06.02 21:47:27 -06'00'

Dra. Eiliana Montero Rojas

Lectora

LUIS MIGUEL ROJAS TORRES (FIRMA)
PERSONA FISICA, CPF-01-1278-0625.
Fecha declarada: 01/06/2022 08:00:39 AM
Esta representación visual no es fuente
de confianza. Valide siempre la firma.

Dr. Luis Rojas Torres

Lector

LETICIA VASQUEZ GUTIERREZ (FIRMA)
Firmado digitalmente por LETICIA VASQUEZ GUTIERREZ (FIRMA)
Fecha: 2022.06.03 12:57:38 -06'00'

Leticia Vásquez Gutiérrez

Sustentante

Tabla de contenido

Resumen

El objetivo del presente trabajo es determinar los factores que influyen en el funcionamiento óptimo de las alternativas de respuesta en ítems de elección única, utilizando un modelo politómico de respuesta nominal de teoría de respuesta al ítem o NRM, por sus siglas en inglés, mediante un estudio de simulación, utilizando tres métodos para la disminución de las categorías de respuesta a partir de los valores de los parámetros de la pendiente en cada categoría del ítem. El primer método consiste en utilizar la posición del valor de la pendiente más baja, mientras que el segundo método plantea utilizar aleatoriamente cualquier posición de la pendiente y el tercer método, utilizar el valor de la posición de la pendiente más alta, mediante dos escenarios: uno aplicando análisis de factores y el otro sin aplicar análisis de factores.

Entre los principales resultados se obtuvo que utilizando cuatro categorías de respuesta se presenta mejor ajuste en la estimación tanto de los coeficientes de dificultad como de los coeficientes de discriminación. Además, se observó que el promedio de la raíz del error cuadrático medio converge a cero a medida que se aumenta el tamaño de muestra en cada método y presenta mejor ajuste en las estimaciones de los parámetros cuando se utiliza un análisis de factores exploratorio en comparación con cuando no se utiliza.

Lista de cuadros

Lista de gráficos

UNIVERSIDAD DE COSTA RICA

SEP Sistema de **Estudios de Posgrado**

Autorización para digitalización y comunicación pública de Trabajos Finales de Graduación del Sistema de Estudios de Posgrado en el Repositorio Institucional de la Universidad de Costa Rica.

Yo, Leticia Vásquez Gutiérrez, con cédula de identidad 503940901, en mi condición de autor del TFG titulado Exploración de la idoneidad de las alternativas de respuesta en ítems de elección única de pruebas estandarizadas

Autorizo a la Universidad de Costa Rica para digitalizar y hacer divulgación pública de forma gratuita de dicho TFG a través del Repositorio Institucional u otro medio electrónico, para ser puesto a disposición del público según lo que establezca el Sistema de Estudios de Posgrado. **SI** X **NO** * ☐

***En caso de la negativa favor indicar el tiempo de restricción: ______________ año (s).**

Este Trabajo Final de Graduación será publicado en formato PDF, o en el formato que en el momento se establezca, de tal forma que el acceso al mismo sea libre, con el fin de permitir la consulta e impresión, pero no su modificación.

Manifiesto que mi Trabajo Final de Graduación fue debidamente subido al sistema digital Kerwá y su contenido corresponde al documento original que sirvió para la obtención de mi título, y que su información no infringe ni violenta ningún derecho a terceros. El TFG además cuenta con el visto bueno de mi Director (a) de Tesis o Tutor (a) y cumplió con lo establecido en la revisión del Formato por parte del Sistema de Estudios de Posgrado.

LETICIA VASQUEZ GUTIERREZ (FIRMA)

Firmado digitalmente por LETICIA VASQUEZ GUTIERREZ (FIRMA)
Fecha: 2022.06.03 12:59:12 -06'00'

FIRMA ESTUDIANTE

Nota: El presente documento constituye una declaración jurada, cuyos alcances aseguran a la Universidad, que su contenido sea tomado como cierto. Su importancia radica en que permite abreviar procedimientos administrativos, y al mismo tiempo genera una responsabilidad legal para que quien declare contrario a la verdad de lo que manifiesta, puede como consecuencia, enfrentar un proceso penal por delito de perjurio, tipificado en el artículo 318 de nuestro Código Penal. Lo anterior implica que el estudiante se vea forzado a realizar su mayor esfuerzo para que no sólo incluya información veraz en la Licencia de Publicación, sino que también realice diligentemente la gestión de subir el documento correcto en la plataforma digital Kerwá.

I. Introducción

En Costa Rica, actualmente, se utilizan diferentes pruebas o test para medir conocimiento, habilidades, actitudes, comportamientos u otro tipo de características de las personas. Las pruebas funcionan como un mecanismo para seleccionar personas para un puesto, midiendo cuál está más capacitada o cuál cumple con los requerimientos solicitados, e incluso para seleccionar personas para ingresar a alguna universidad pública.

En el país se requiere de pruebas o exámenes como parte de los procesos de admisión para ingresar a las universidades públicas. Por ejemplo, la Universidad de Costa Rica (UCR), el Instituto Tecnológico de Costa Rica (TEC) y la Universidad Nacional de Costa Rica (UNA) son universidades estatales que usan una prueba de aptitud académica cuyo propósito es medir el potencial intelectual para el aprendizaje y no lo que el examinado ya sabe.

El Estado de la Educación (2017) indica que las universidades públicas reciben cada año en conjunto más de cinco solicitudes de admisión por cada cupo disponible para nuevos estudiantes. Además, según Montero, Rojas, Salazar y Negrín (2015), el proceso de ingreso a la UCR es muy selectivo y gran cantidad de aspirantes se quedan por fuera, ya que por año reporta aproximadamente 50.000 aspirantes, de los cuales un máximo de 8.000 logra obtener cupo.

Por otro lado, también se utilizan las pruebas psicométricas para la contratación de personal. Mediante la medición de aptitudes, habilidades, inteligencia y tendencias de conducta de las personas, entre otras características, se intenta conocer el potencial o la competencia de un candidato en relación con las tareas que debe desempeñar.

Por lo tanto, a partir de estos ejemplos, se puede constatar que las pruebas son importantes, ya que permiten seleccionar mediante filtros las personas con mayor capacidad de análisis para optar para una carrera o la persona más competente para optar por un puesto de trabajo. Debido a la importancia que representan, es necesario que dichos test o pruebas estén bien diseñados y que midan adecuadamente el constructo o fenómeno de análisis.

Es común que las pruebas estén conformadas por ítems de dos o más opciones de respuesta, de las cuales algunas sirven de distractores para la persona examinada. La calidad de los ítems podría verse afectada por la cantidad de opciones, por lo que es necesario determinar cuántas opciones de respuesta son óptimas para que la prueba esté bien diseñada. Este trabajo es de mucho valor pues permitirá determinar la cantidad óptima de respuestas por ítem de un test mediante el modelo de respuesta nominal.

1.1 Objetivos de la investigación

1.1.1 Objetivo general

Determinar los factores que influyen en el funcionamiento óptimo de las alternativas de respuesta en ítems de elección única utilizando un modelo politómico de respuesta nominal.

1.1.2 Objetivos específicos

- Establecer la relación de la discriminación y la dificultad del ítem sobre el funcionamiento de las alternativas de respuesta en pruebas estandarizadas.
- Generar un proceso sistemático de análisis de alternativas de respuesta con el modelo de respuesta nominal en una prueba estandarizada.

1.2 Justificación y planteamiento del problema

Las pruebas estandarizadas, por lo general, se construyen con ítems de elección única debido a la versatilidad o facilidad de su aplicación a los individuos. Una de las teorías comúnmente empleadas para los análisis corresponde a los modelos dicotómicos de teoría de respuesta al ítem, los cuales permiten obtener la precisión de las medidas, pero no llegan a determinar cuál es el funcionamiento óptimo de las alternativas de respuesta de los ítems.

Aunque algunos estudios han determinado que tres o cuatro opciones funcionan como óptimas en la calidad de los ítems (Asún y Zúñiga, 2008; Baguaeil y Amrahi, 2011), los estudios no fundamentan dicha decisión con base en un estudio sistemático bajo la teoría de respuesta al ítem unidimensional, por lo cual surge la necesidad de determinar qué factores influyen en la calidad técnica de las alternativas de respuesta, mediante un estudio sistemático y su aplicación a datos empíricos.

1.3 Pregunta de investigación

¿Cuáles son los factores que influyen en el funcionamiento óptimo de las alternativas de respuesta en ítems de elección única utilizando un modelo politómico de respuesta nominal?

II. Marco teórico

En esta sección se describen los antecedentes y conceptos teóricos.

2.1 Antecedentes

A pesar de que la aplicación de los modelos politómicos de la teoría de respuesta al ítem (TRI) tienen muchas ventajas en comparación con los de la teoría clásica de los test (TCT), hay pocos estudios en donde se hayan aplicado a la medición de actitudes o personalidad. Además, uno de los principales aspectos a estudiar es la determinación del número óptimo de alternativas de respuesta y los estudios realizados respecto a ese tema han mostrado que, por lo general, utilizar cuatro alternativas de respuesta, o incluso tres, podría ser más adecuado, siendo poco relevante la información aportada por las otras opciones de respuesta (Asún y Zúñiga, 2008). Este aspecto es el de mayor importancia para esta investigación.

En el estudio de Asún y Zúñiga (2008), se utilizaron diferentes tipos de modelos dicotómicos y politómicos de TRI y se obtuvo que no presentan un ajuste satisfactorio en tests no cognitivos, además de que las opciones intermedias son poco informativas y que cuatro alternativas de respuesta son suficientes para modelos de TRI politómicos. Dichos resultados indican que concuerdan con estudios previos.

Baguaeil y Amrahi (2011) mencionan que tradicionalmente se recomienda el uso de cuatro o cinco opciones de respuesta por ítem, para reducir el efecto de adivinar al azar. Además, indican que la mayoría de los estudios realizados para investigar el número óptimo de opciones recomiendan el uso de ítems de tres opciones. En su estudio, utilizaron 3 formularios diferentes con 30 ítems, uno con 5 opciones de respuesta por ítem, otro con 4 opciones y el último con 3 opciones; se utilizaron diez ítems de anclaje y los distractores se eliminaron de forma aleatoria. Entre los principales resultados, no se encontraron diferencias significativas con respecto a las estimaciones de la dificultad de los ítems y se determinó que los comportamientos de respuesta de las personas eran idénticos en los tres formularios con diferentes números de opciones por ítem.

2.2 Conceptos teóricos

En este apartado se definen algunos términos de gran importancia para el estudio, como lo son: test, Teoría Clásica de los Tests, Teoría de Respuesta al Ítem, ítems politómicos, dificultad del ítem, discriminación del ítem, entre otros.

El test, también conocido como prueba o cuestionario, ha sido utilizado para hacer referencia a un tipo de examen o evaluación que puede desarrollarse en el marco de dos grandes modelos: Teoría Clásica de los Tests (TCT) y Teoría de Respuesta al Ítem (TRI).

Según Muñiz (2018), el principal objetivo de la teoría clásica de los test es identificar un modelo estadístico que permita fundamentar adecuadamente las puntuaciones de los test y la estimación de los errores de medida asociados al proceso de medición. Además, Muñiz (2010) indica que hay ciertos problemas de medición que no quedan resueltos con la TCT. Entre los problemas más importantes, el primero consiste en que las mediciones no resultan invariantes respecto al instrumento utilizado y el segundo problema es la ausencia de invariancia de las propiedades de los test respecto de las personas utilizadas para su estimación. Dichos problemas van a ser resueltos mediante la TRI, sin embargo, Muñiz menciona que se tiende a combinar el uso de ambos métodos, TRI y TCT, en el desarrollo de los test.

Por lo tanto, la TRI surge como una alternativa a los problemas y limitaciones que se presentan en la TCT. En varios estudios (citados por Montero, 2000), se menciona que la TRI intenta corregir las debilidades de la TCT, puesto que intenta obtener estimaciones de los parámetros del ítem que sean independientes de la muestra de examinados, así como puntuaciones que sean independientes del instrumento utilizado. Los principales supuestos de la TRI son la unidimensionalidad, que significa que una aptitud o rasgo (θ) es suficiente para explicar los resultados de los examinados, y la relación entre los ítems y la independencia local, que significa que no existe relación entre las respuestas de los examinados a diferentes ítems, dentro del mismo nivel de aptitud (Martínez, Hernández, y Hernández, 2006).

Debido a que muchos de los instrumentos psicométricos para evaluar actitudes, personalidad, aptitudes, entre otros están formados por ítems politómicos, resulta relevante el utilizar la TRI como una alternativa a la TCT.

Un ítem politómico (Peréz, 2004) se define como una variable de respuesta con más de dos categorías (*k* categorías), las cuales podrían clasificarse en dos tipos de ítems:

- Ítems politómicos nominales: las categorías de respuesta no están ordenadas.
- Ítems politómicos ordinales: las categorías de respuesta están ordenadas.

Por otra parte, Asún y Zúñiga (2008) mencionan que para el caso de los ítems politómicos no existe un grado de dificultad del ítem como tal, sino un parámetro de posición de las alternativas de respuesta, dentro del rasgo latente o habilidad, para responder cada alternativa con una alta probabilidad.

También es importante considerar la diferencia entre la dificultad y la discriminación de un ítem. Al respecto, Martínez et al. (2006) definen la dificultad del ítem como la ubicación del ítem en la escala de aptitud o habilidad (θ), es decir, el nivel de habilidad que requiere el examinado para que el ítem sea resuelto con éxito, y es, por lo tanto, un índice de la posición o localización del ítem en la escala de medición del rasgo o aptitud. Así mismo, Asún y Zúñiga (2008) lo definen como el grado de habilidad o conocimiento que se requiere para tener altas probabilidades de escoger la opción correcta.

Por otra parte, la discriminación del ítem es la capacidad de un ítem de diferenciar entre sujetos que tienen un buen rendimiento en el test, con respecto a los sujetos que tienen mal rendimiento (Martínez et al., 2006), por ejemplo, en la TRI el rendimiento se entiende como la habilidad. Los ítems que presentan mayor pendiente discriminan mejor, mientras que los ítems con curvas más planas son ítems que discriminarán peor, puesto que la probabilidad de seleccionar la respuesta correcta es casi igual independientemente el nivel de aptitud o habilidad. Se recomienda que los ítems presenten un alto poder de discriminación, para que distinga mejor entre los sujetos que poseen un nivel de habilidad suficiente para responder correctamente de los que no.

En los ítems politómicos, al tener varias categorías de respuesta, una de las opciones es correcta y las demás son consideradas incorrectas o distractores. ‘Distractor’ es el nombre que se le da a las alternativas de respuestas incorrectas. Un distractor óptimo es considerado como tal cuando es seleccionado por examinados con bajos niveles de habilidad. Este es un punto importante en este trabajo, el definir la cantidad óptima de distractores.

Es importante considerar que hay diferentes tipos de modelos de respuesta para ítems politómicos. Los modelos de respuesta a ítems politómicos son una generalización de los modelos de respuesta para ítems dicotómicos. Los principales modelos para ítems politómicos ordinales son los siguientes:

- Modelo de respuesta graduada (Samejima, 1969)
- Modelo de respuesta graduada modificado (Muraki, 1990)
- Modelo de escala de estimación (Andrioch, 1978)
- Modelo de crédito parcial (Masters, 1982)
- Modelo de crédito parcial generalizado (Muraki, 1992)
- Modelo secuencial (Tutz, 1990)

Por su parte, los principales modelos para ítems politómicos nominales son los siguientes:

- Modelo de respuesta nominal (Bock, 1972)
- Modelo nominal modificado (Samejima, 1979)
- Modelo de elección múltiple (Thissen y Steinberg, 1984)

Para efectos de este trabajo, el modelo a utilizar es el modelo de respuesta nominal, por lo que es el único que se va a explicar en detalle.

3.3 Modelo de respuesta nominal

El modelo de respuesta nominal fue propuesto por Bock en el año 1972, mediante el uso de ítems nominales, en los que no existe un orden en las opciones de respuesta. Asún y Zúñiga (2008) indican que en este modelo se incorporan muchos parámetros, por lo que se requiere una muestra grande para realizar estimaciones precisas, sin embargo, tiende a ajustar frecuentemente a las respuestas de los individuos, porque establece pocas restricciones a los datos. En este modelo el método de estimación implementado es por máxima verosimilitud.

La formulación original del modelo propuesto por Bock (1972) fue la siguiente:

$$\mathrm{T}(u = k \mid \theta; a, c) = T(k) = \frac{\exp{(z_k\ )}}{\sum_i \exp{(z_k\ )}}$$

con

$$z_{k=}\ a_k\,\theta +\ c_k$$

Al combinar ambas ecuaciones se obtiene:

$$T(k) = \frac{\exp{(a_k\,\theta +\ c_k\ )}}{\sum_i \exp{(a_i\,\theta +\ c_i\ )}}$$

donde,

$\mathrm{T}(k)$ es la curva que traza la probabilidad de que la respuesta al ítem u esté en la categoría *k* es una función de la variable latente *θ* con los parámetros vectoriales a y c.

θ = nivel de habilidad del sujeto

a =discriminación del ítem *i* en la categoría *k*

c = intercepto del ítem *i* en la categoría *k*

Rivera (2019) menciona que a pesar de que en la práctica se suele asumir que el número de opciones de respuesta es la misma en todos los ítems, en el modelo de respuesta nominal el número de alternativas de respuesta puede ser distinto de un ítem a otro.

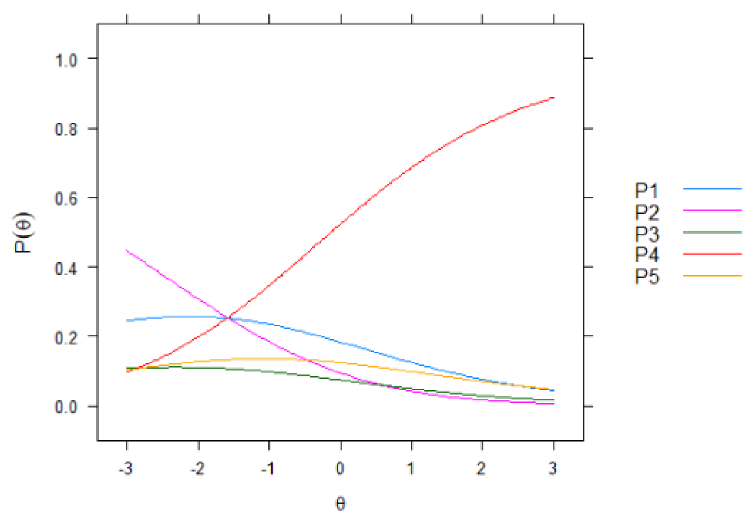
Según Attorresi, Lozzia, Abal, Galibert, y Aguerri (2009) la relación que existe entre el comportamiento de un sujeto frente a un ítem y la habilidad que genera esta conducta se define como la Curva Característica de Ítem (CCI) o en algunos casos también se le conoce como Función de Respuesta al Ítem (FRI). Por esto, a medida que aumenta el nivel de aptitud de un sujeto, aumenta la probabilidad de acertar correctamente un ítem (Muñiz, 2018).

La curva CCI permite observar la relación entre la habilidad o rasgo latente que está siendo evaluada (eje de las abscisas o eje X) y la probabilidad de un examinado de responder correctamente al ítem (eje Y). Entre más acentuada es la pendiente de la curva, mayor es la discriminación del ítem y entre más se desplaza la curva de la categoría correcta a la derecha en la escala de habilidad, mayor dificultad posee el ítem.

Un ítem tiene tantas CCI como categorías de respuesta. En el gráfico 1 se observa la curva característica del ítem 1, donde se observa que la categoría de respuesta correcta es la categoría 4, ya que es la que posee mayor valor de la pendiente "*a*". Además, se observa que entre mayor habilidad tiene el examinado, mayor probabilidad va a tener de seleccionar la respuesta correcta del ítem, por lo cual dicha curva siempre será creciente.

Para este caso se observan 5 curvas, debido a que el ítem cuenta con 5 opciones de respuesta. También se observa que entre menor sea la habilidad del sujeto, mayor probabilidad se tiene de seleccionar una de las categorías incorrectas o distractores.

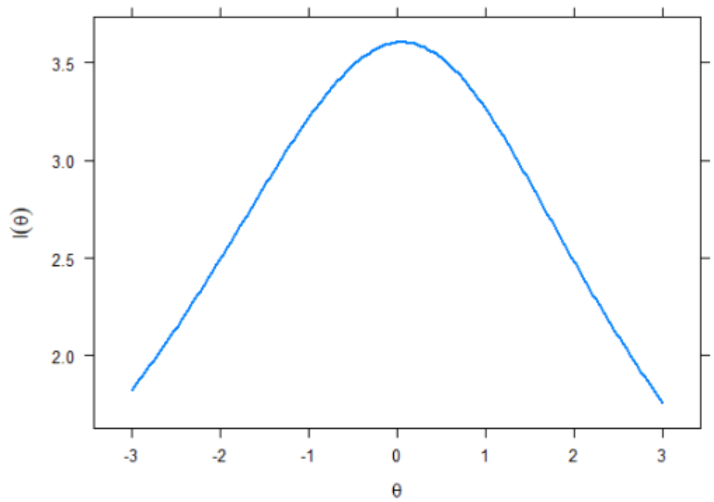
Gráfico 1. Curva característica del Ítem 1.



Adicionalmente, la curva característica del test (CCT) es la suma de las curvas características de los ítems que conforman el test, donde se indica en qué nivel de habilidad la evaluación es más precisa (Hidalgo y French, 2016).

La función de información es un indicador de la precisión del test, entre mayor sea $I(\theta)$ menor es el error típico de estimación, lo que indica que mayor es la información que las estimaciones poseen sobre θ. Según el gráfico 2, el test aporta información máxima en el valor de $\theta = 0$.

Gráfico 2. Función de información del test.



III. Metodología

Seguidamente de describe la metodología empleada en el estudio.

3.1 Diseño de simulación

Se utilizó la elaboración de diferentes bases mediante una simulación con diferentes opciones de respuesta por ítem, pasando de cinco opciones de respuesta por ítem a cuatro opciones y, por último, a tres opciones, utilizando tres métodos, los cuales se explican a continuación.

Para explicar los métodos se utilizará como ejemplo el ítem 1, donde la estimación de los parámetros de la pendiente (discriminación) son los siguientes:

Cuadro 1. Estimación de los coeficientes de los parámetros de la pendiente según categoría de respuesta para el ítem 1.

Categoría de respuesta	Coeficiente
1	-0,07
2	-0,49
3	-0,10
4	0,58
5	0,08

Si ordenamos de forma ascendente la estimación de los coeficientes del ítem 1, quedarían de la siguiente forma:

Cuadro 2. Estimación de los coeficientes de los parámetros de la pendiente según categoría de respuesta de forma ascendente para el ítem 1.

Categoría de respuesta	Coeficiente
4	0,58
5	0,08
1	-0,07
3	-0,10
2	-0,49

Como se logra observar en el Cuadro 2, la categoría 2 es la que posee el valor de la estimación del coeficiente de la pendiente más bajo, es decir, presenta menor discriminación; seguida de la categoría 3, hasta llegar al valor de la pendiente más alto para la categoría 4, la cual es la opción correcta.

A continuación, se procede a explicar cada uno de los métodos:

Método 1: Se selecciona la posición del siguiente valor de la pendiente más baja. Utilizando este método, al pasar de 5 a 4 opciones, a todos los sujetos que hayan seleccionado la categoría 2 (categoría con valor de la pendiente más baja) para el ítem 1 se les asignará la categoría 3 (siguiente categoría con el valor de la pendiente más baja) y para pasar de 4 a 3 opciones, a todos los sujetos que hayan seleccionado la categoría 3 se les asignará la categoría 1.

Método 2: Se selecciona la posición aleatoriamente. En este método, al pasar de 5 a 4 opciones, se selecciona aleatoriamente una de las categorías 3, 1, 5 y 4. Y para pasar de 4 a 3 opciones de respuesta, se selecciona aleatoriamente una de las categorías 1, 5 y 4.

Método 3: Se selecciona la posición del valor de la pendiente más alta. En este caso, al pasar de 5 a 4 opciones, a todos los sujetos que hayan seleccionado la categoría 2 se le asignará la categoría 4 (categoría con mayor valor de la pendiente) y al pasar de 4 a 3 opciones de respuesta, a los que hayan seleccionado la categoría 3 (segundo valor de la pendiente más baja) también se le asignará la categoría 4.

3.2 Generación de datos

La fuente de datos utilizada consiste en 35 valores de parámetros reales de ítems de una prueba piloto para la elaboración del folleto de práctica para la Prueba de Aptitud Académica de la Universidad de Costa Rica, con cinco alternativas de respuesta, en el 2011. En el Cuadro 3, se presenta la estimación de los parámetros de la pendiente y del intercepto para cada uno de los ítems mediante el modelo de respuesta nominal.

Cuadro 3. Parámetros estimados según el modelo de respuesta nominal para los ítems de estudio.

Ítem	Parámetros de discriminación					Parámetros del intercepto				
	a_1	a_2	a_3	a_4	a_5	c_1	c_2	c_3	c_4	c_5
1	-0,07	-0,49	-0,10	0,58	0,08	0,18	-0,48	-0,72	1,23	-0,21
2	-0,54	0,94	0,17	-0,35	-0,22	-1,24	1,16	0,42	0,16	-0,51
3	-0,19	-0,55	0,60	0,15	-0,01	-0,60	0,16	0,49	-0,19	0,14
4	-0,03	-0,04	-0,07	-0,14	0,29	0,33	0,18	-0,13	-0,05	-0,33
5	-0,17	-0,50	0,16	0,37	0,15	-1,25	-0,87	-0,04	1,50	0,65
6	-0,42	-0,33	0,87	0,04	-0,15	-0,36	0,23	0,59	0,10	-0,57
7	-0,55	-0,30	0,81	-0,25	0,29	0,22	-0,09	0,45	-0,08	-0,51
8	0,19	-0,14	-0,69	0,75	-0,12	0,37	-0,72	0,99	-0,62	-0,02
9	0,00	-0,07	-0,22	-0,42	0,72	1,41	0,62	-0,83	-1,15	-0,05
10	0,09	-0,61	0,28	-0,12	0,36	0,38	0,00	-0,30	-0,36	0,28
11	-0,33	-0,53	0,02	0,51	0,33	0,49	-0,25	0,13	-0,01	-0,36
12	0,11	-0,18	-0,35	0,17	0,25	-0,04	-0,32	0,23	0,03	0,09
13	-0,05	0,41	-0,06	-0,04	-0,26	0,12	0,42	-0,15	0,20	-0,59
14	0,30	0,05	0,37	-0,24	-0,48	0,57	0,17	0,17	-0,62	-0,30
15	-0,23	-0,08	0,00	0,06	0,26	0,50	0,07	0,14	-0,06	-0,65
16	-0,16	0,00	0,31	-0,11	-0,04	-0,04	-0,30	0,61	-0,37	0,10
17	-0,08	0,06	-0,14	0,13	0,02	-1,06	0,77	-0,32	0,30	0,31
18	-0,33	0,46	0,05	-0,05	-0,13	-0,89	0,60	0,46	-0,64	0,47
19	0,02	-0,29	-0,03	-0,29	0,59	-0,56	-0,83	-0,40	0,18	1,60
20	-0,32	0,58	-0,51	0,20	0,05	-0,88	-0,32	0,29	0,19	0,72
21	-0,03	0,00	0,13	0,04	-0,15	-0,19	0,22	0,36	-0,27	-0,12
22	-0,28	0,25	0,04	-0,03	0,02	-0,10	-0,19	0,15	-0,22	0,36
23	-0,21	-0,23	0,71	-0,31	0,05	-0,48	0,12	0,76	0,21	-0,62
24	-0,08	0,22	-0,12	-0,09	0,07	-0,34	0,25	0,04	-0,05	0,10
25	-0,28	0,04	0,31	-0,08	0,00	0,21	-0,40	0,92	0,23	-0,96
26	-0,02	0,18	0,06	0,20	-0,41	-0,24	0,18	0,16	0,15	-0,26
27	-0,39	-0,15	-0,07	0,09	0,52	0,52	0,15	0,01	-0,40	-0,28
28	0,03	0,20	-0,15	0,00	-0,09	-0,40	0,17	0,36	0,10	-0,23
29	-0,63	-0,23	0,74	0,19	-0,08	0,15	-0,57	0,49	-0,12	0,05
30	-0,08	0,32	0,12	-0,15	-0,21	0,03	-0,14	0,33	0,16	-0,38
31	-0,07	-0,10	-0,05	0,49	-0,27	0,70	0,33	-0,35	0,06	-0,74
32	-0,16	0,14	0,14	0,28	-0,41	-0,46	0,12	0,57	0,02	-0,25
33	-0,21	0,01	-0,20	-0,12	0,52	-0,51	-0,30	-0,12	1,09	-0,15
34	-0,22	0,34	0,39	0,25	-0,75	-0,66	-0,99	0,42	0,85	0,38
35	-0,11	0,26	0,08	0,19	-0,42	0,06	0,34	-0,47	0,27	-0,20

Las bases utilizadas fueron generadas mediante un conjunto de datos reales donde se utilizaron tres tamaños de muestra: 500, 750 y 1000, y se utilizaron los 35 ítems. Para cada uno de los tamaños de muestra, se realizaron 100 réplicas para realizar los análisis.

Además de los tres métodos mencionados anteriormente para la disminución en las categorías de respuesta, también se consideraron dos escenarios: el primer escenario sin realizar análisis de factorial exploratorio y el segundo escenario realizando análisis de factorial exploratorio para excluir los ítems que generaban problemas en las estimaciones. El análisis factorial exploratorio permite determinar la cantidad de factores o componentes que están presentes en el test o instrumento (Montero y Jiménez, 2013).

Primero se realizó la estimación del modelo mediante el paquete MIRT (*Multidimensional Item Response Theory*) (Chalmers, 2012), que se encuentra en el programa R. MIRT ajusta un modelo de análisis factorial de máxima verosimilitud a cualquier mezcla de datos dicotómicos y politómicos bajo el paradigma de la teoría de respuesta al ítem, utilizando para este caso el algoritmo EM (*Expectation Maximization*). Para la extracción de los coeficientes, debe especificar la conversión de los parámetros de intersección y de pendientes en parámetros tradicionales de TRI.

Posteriormente a la estimación de los coeficientes de los modelos mediante MIRT y una vez realizado el proceso de ambos escenarios (con AFE y sin AFE) y los métodos para la disminución de las alternativas de respuesta, se dicotomizaron las bases mediante la función key2binary (contenida en la librería MIRT), la cual convierte los datos del patrón de respuesta a un formato dicotómico, donde se le asignaba valor uno a los sujetos que contenían la respuesta correcta y, en caso contrario, se le asignaba cero.

Una vez dicotomizadas las bases, se realiza el análisis mediante el modelo logístico de dos parámetros, el cual tiene la siguiente fórmula (Muñiz, 2018):

$$P_i(\theta) = \frac{e^{Da_i(\theta - b_i)}}{1 + e^{Da_i(\theta - b_i)}}$$

donde:

$P_i(\theta)$ =Probabilidad de acertar el ítem i para un valor θ.

θ = Valores de la variable medida.

b_i: Índice de dificultad del ítem i.

a_i: Índice de discriminación del ítem i.

e: Base de los logaritmos neperianos (2,72).

D: Constante. Cuando toma el valor 1,702, la función logística se aproxima a la normal acumulada.

Por lo tanto, se cuenta con las siguientes bases para ambos escenarios:

- Base 1: Base dicotomizada con 5 opciones de respuesta.
- Base 2: Base dicotomizada con 4 opciones de respuesta utilizando el método 1 (seleccionando la posición del siguiente valor de la pendiente más baja).
- Base 3: Base dicotomizada con 3 opciones de respuesta utilizando el método 1 (seleccionando la posición del siguiente valor de la pendiente más baja).
- Base 4: Base dicotomizada con 4 opciones de respuesta utilizando el método 2 (seleccionando la posición aleatoriamente).
- Base 5: Base dicotomizada con 3 opciones de respuesta utilizando el método 2 (seleccionando la posición aleatoriamente).
- Base 6: Base dicotomizada con 4 opciones de respuesta utilizando el método 3 (seleccionando la posición del valor de la pendiente más alta).
- Base 7: Base dicotomizada con 3 opciones de respuesta utilizando el método 3 (seleccionando la posición del valor de la pendiente más alta).

Una vez dicotomizadas cada una de las 7 bases, se generaron los modelos utilizando la librería *ltm* de R (Rizopoulos, 2006), la cual genera modelos de variable latente para datos binarios.

3.3 Análisis de datos

Se determinó el valor promedio de la raíz del error cuadrático medio (RECM), que representa la media del error estándar de las estimaciones de los parámetros de cada ítem, para cada modelo, en cada uno de los tamaños de muestra, el cual tiene la siguiente fórmula:

$$RECM = \sqrt{\sum_{l=1}^{k} \frac{(\beta_l - \widehat{\beta_l})^2}{k}}$$

donde β_l representa la estimación del coeficiente utilizando 5 categorías de respuesta, $\widehat{\beta_l}$ el parámetro estimado mediante 4 y 3 categorías y k el número de réplicas utilizadas en cada caso, puesto que, en algunos escenarios, a pesar de que se utilizaron 100 réplicas puede disminuir dicho valor por la falta de convergencia del modelo en alguna réplica en particular, las cuales fueron excluidas del análisis.

Además, se calculó el valor del criterio de información de Akaike (AIC) para cada uno de los modelos generados en la simulación y luego este se promedió entre todas las réplicas.

Se utilizó el programa R-3.6.2 para el análisis de datos, mediante la librería MIRT y Excel (versión 2013) para la elaboración de cuadros.

IV. Resultados

A continuación, se presentan los principales resultados, tanto de la base empírica como de la simulación utilizadas en esta investigación.

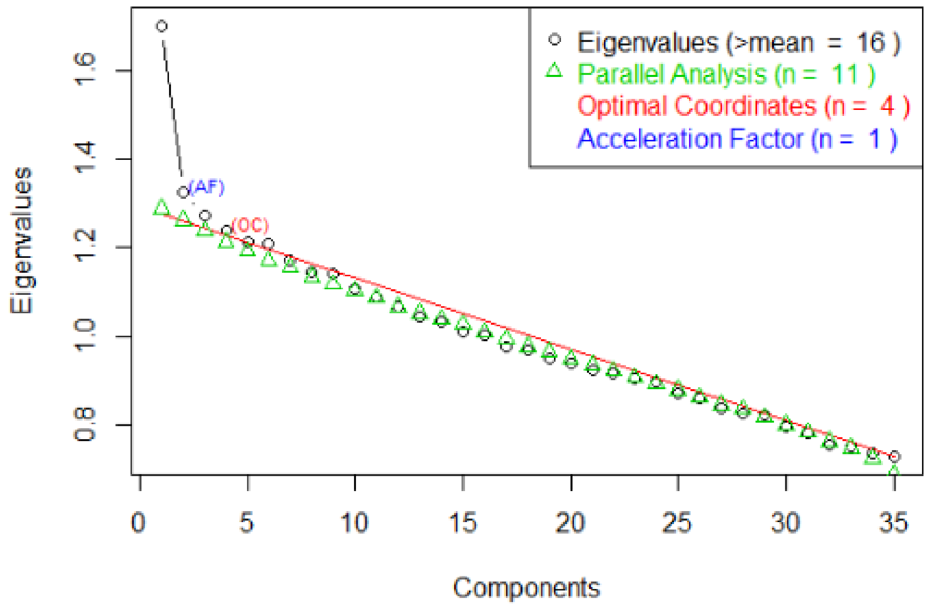
4.1 Base empírica

La base empírica utilizada contiene información de 1.263 sujetos y cuenta con 35 ítems. Para analizar el supuesto de unidimensionalidad previo al análisis factorial exploratorio, se realizaron los test de la medida de adecuación muestral de Kaiser-Meyer-Olkin y la prueba de esfericidad de Bartlett. El índice del KMO debe de rondar por valores superiores a 0,70 para que el análisis de factores pueda ser justificado (Hernández, 2013). En este caso, se obtuvo un valor de 0,56, el cual se considera como inapropiado para realizar el análisis de factores.

A su vez, se realizó la prueba de esfericidad de Bartlett, la cual contrasta la hipótesis nula de que la matriz de correlaciones es igual a la matriz de identidad. En esta, con una significancia del 5 % hay suficiente evidencia estadística como para rechazar la hipótesis planteada, ya que la probabilidad asociada es de 0,00.

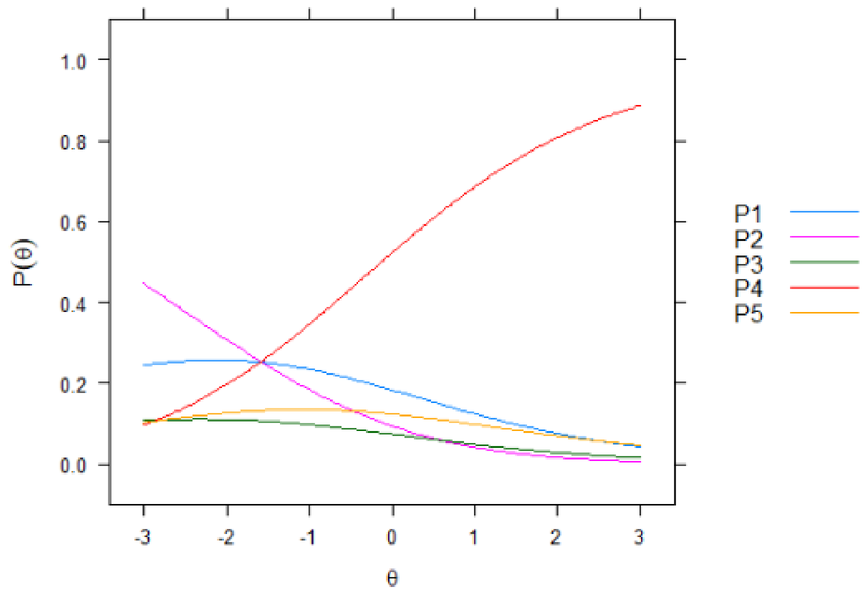
Posteriormente, para determinar el número de factores, se utilizó el criterio de análisis paralelo, donde se logra ver en el gráfico 3 que muy levemente en apariencia hay un factor que sobresale.

Gráfico 3. Análisis paralelo de los autovalores observados y los autovalores de las réplicas.



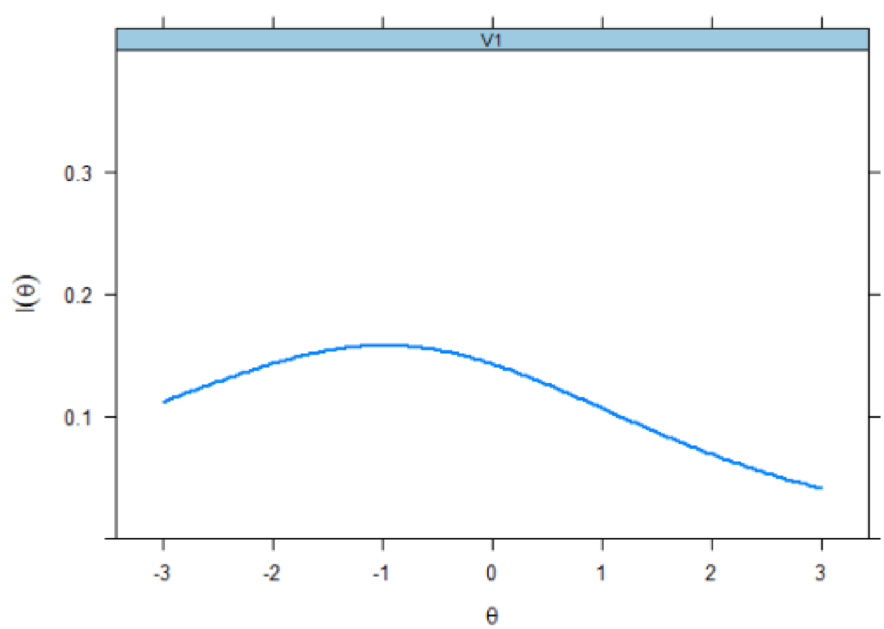
Por lo tanto, se asume unidimensionalidad y se procede con el análisis de la estimación de los parámetros mediante el modelo de respuesta nominal. En el gráfico 4 se presenta la curva característica de respuesta del ítem 1, que permite describir la probabilidad que tiene una persona de elegir cada una de las categorías de respuesta en función de su nivel de habilidad (θ); por ejemplo, para el ítem 1, la categoría de respuesta 4 es la que posee mayor valor de la pendiente y además se observa que conforme aumenta el valor de la habilidad del sujeto en dicho ítem, aumenta la probabilidad de seleccionar la respuesta correcta. En el anexo 1 se encuentran los gráficos de las curvas características para los demás ítems.

Gráfico 4. Curva característica del ítem 1.



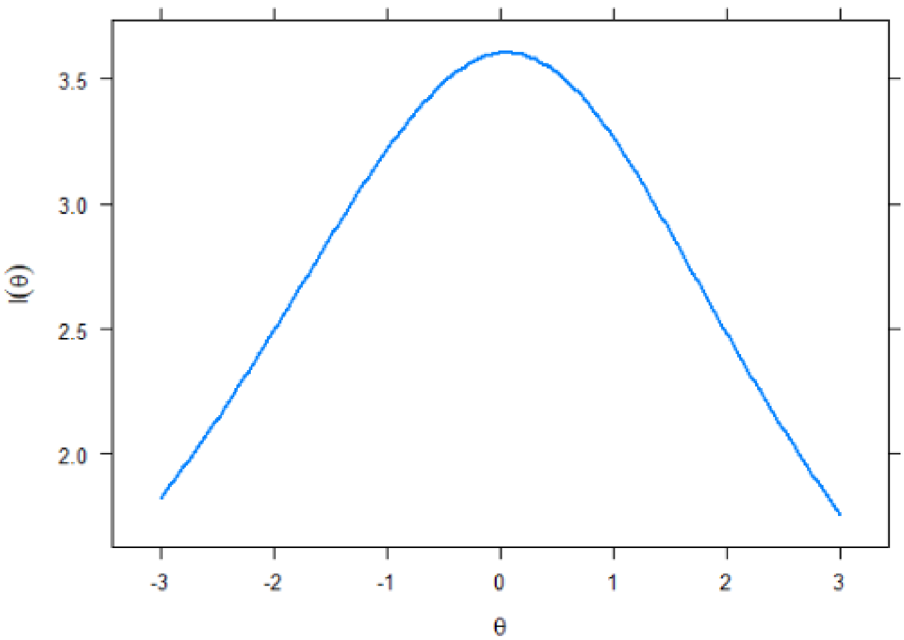
En el gráfico 5 se observa la cantidad de información que el ítem 1 puede explicar en los diferentes niveles del rasgo latente. Los ítems que proporcionan más información (es decir, donde está la curva más alta) son los ítems 2,6,7, 8, 2, 29 y 34, mientras que los demás ítems proporcionan menor cantidad de información, es decir, poseen la curva con el pico más bajo (ver anexo 2).

Gráfico 5. Función de información de ítem 1.



Sumando todas las funciones de información de los ítems, se obtiene la cantidad total de información disponible de la prueba, en el gráfico 6 se informa en qué nivel de habilidad la evaluación es más precisa. Entre más información tiene un ítem o un test, más precisa es la estimación de la habilidad para el examinado.

Gráfico 6. Función de información del test.



Posteriormente, se procedió a calcular la raíz del error cuadrático medio en cada uno de los escenarios mediante los métodos utilizados; es decir, sin AFE (incorporando en el análisis todos los ítems sin determinar si todos los ítems miden el mismo constructo o no) y con AFE (excluyendo los ítems que no miden el mismo constructo y presentan problemas es las estimaciones), en el cuadro 5 se observa que, independientemente del escenario, es decir, si se aplica o no previamente el AFE, el método que resulta con mejor ajuste tanto para el parámetro de dificultad como para el de discriminación es cuando se asigna la posición de la siguiente pendiente más baja utilizando 4 categorías de respuesta. Asimismo, en todos los métodos posee mejor ajuste cuando se utilizan 4 categorías de respuesta que cuando se utilizan 3.

Además, sobre la RECM, se observa que cuando se utilizan 4 categorías con el método de asignación a la posición de la siguiente pendiente más baja, al comparar el escenario sin

AFE con el escenario con AFE, disminuye únicamente el promedio de RECM para el parámetro de dificultad, en cambio para el parámetro de discriminación más bien aumenta.

Un dato importante para considerar es que para el método de asignación de la posición de la siguiente pendiente más baja, se pasa de 35 a 28 ítems porque se dejan únicamente los casos donde las estimaciones de los coeficientes oscilan entre -6 y 6, debido a que hubo 7 ítems que presentaron problemas de convergencia. Por su parte, para el escenario donde se realiza AFE, se pasa de 35 ítems a 12 ítems, por lo cual se considera que se pierde mucha información.

Cuadro 4. RECM entre los coeficientes estimados del parámetro de dificultad y discriminación en cada escenario para cada uno de los métodos.

Método de asignación de posición	**Escenario 1 sin AFE**		**Escenario 2 con AFE**	
	Dificultad	**Discriminación**	**Dificultad**	**Discriminación**
Siguiente pendiente más baja				
4 categorías de respuesta	**0,08**	**0,10**	**0,03**	**0,15**
3 categorías de respuesta	0,32	0,21	0,78	0,24
Posición aleatoriamente				
4 categorías de respuesta	0,71	0,20	0,24	0,24
3 categorías de respuesta	0,93	0,29	1,16	0,34
Pendiente más alta				
4 categorías de respuesta	2,52	0,44	1,36	0,41
3 categorías de respuesta	3,06	0,59	4,28	0,49

En el cuadro 6 se observan los valores del criterio de información de Akaike para cada uno de los modelos generados. El modelo que presenta menor AIC es el de 4 categorías de respuesta con el método de asignación de la posición de la siguiente pendiente más baja en ambos escenarios. Además, se observa que en el escenario cuando se realiza AFE el valor del AIC se reduce en más de la mitad en comparación con cuando no se realiza el AFE.

Cuadro 5. Valor del AIC para cada modelo utilizado para cada uno de los métodos.

Modelo	**Escenario 1 sin AFE**	**Escenario 2 con AFE**
5 categorías	47532,49	22026,32
Siguiente pendiente más baja		
4 categorías de respuesta	**46254,12**	**20743,29**
3 categorías de respuesta	48309,81	21575,68
Posición aleatoriamente		
4 categorías de respuesta	50612,43	22531,77
3 categorías de respuesta	54704,63	24385,78
Pendiente más alta		
4 categorías de respuesta	54729,97	23713,73
3 categorías de respuesta	53340,70	22373,34

4.2 Simulación

Se analizó la comparación entre los coeficientes estimados utilizando los tres métodos con 4 y 3 opciones de respuesta con los coeficientes teóricos (5 opciones de respuesta). Para hacer esta comparación, se calculó la resta de los coeficientes estimados con los coeficientes teóricos, por lo que mientras más cercano a cero sea el resultado de esta resta, mejor estimado se encuentra el coeficiente. A continuación, se procede a explicar los dos escenarios utilizados.

4.2.1 Primer escenario: Sin AFE

En este caso no se realizó análisis de factores, es decir, se incluyen todos los ítems sin corroborar si miden o no el mismo constructo y antes del cálculo del promedio de la raíz del error cuadrático medio (RECM) en cada uno de los 3 métodos se consideraron únicamente las réplicas donde la estimación de los coeficientes, tanto del parámetro de discriminación como de los parámetros de dificultad, oscilara entre -6 y 6, esto debido a que algunas réplicas presentaron problemas de convergencia en la estimación de los modelos.

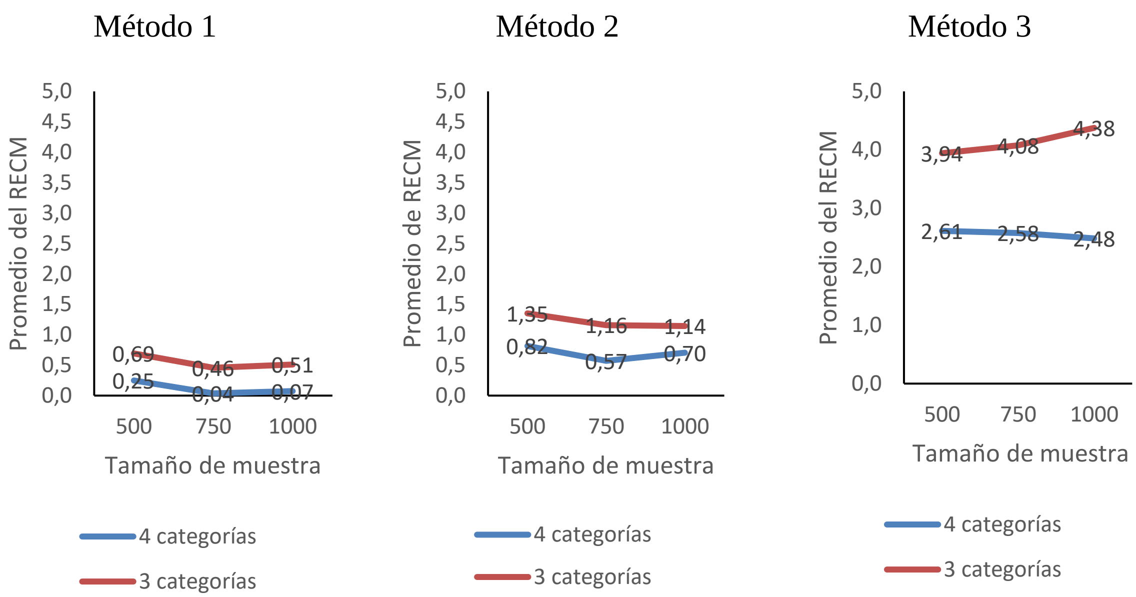
4.2.1.1 Estimación de los coeficientes del parámetro de dificultad

En el grafico 8 se observa que, para el Método 1 (seleccionando la posición de la siguiente probabilidad más baja), utilizar 4 opciones de respuesta posee mejor ajuste que cuando se utilizan 3 opciones, ya que la diferencia calculada converge a cero prácticamente utilizando una muestra de 500. Además, se observa que cuando se pasa de 750 a un tamaño de muestra de 1.000 sujetos, aumentaba el valor de la RECM, sin embargo, dicha diferencia era menor a 0,05 en ambos casos.

Con respecto al Método 2 (seleccionando la posición aleatoriamente), se observa el mismo comportamiento que con el Método 1: utilizar 4 opciones de respuesta posee mejor ajuste que cuando se utilizan 3 opciones, sin embargo, la convergencia a cero no es tan notoria como en el Método 1. Para ambos casos se observa que conforme se aumenta el tamaño de la muestra disminuye el promedio de la raíz del error cuadrático medio.

Para el caso del Método 3 (seleccionando la posición de la probabilidad más alta), se aprecia que utilizar 4 categorías de respuesta posee mejor ajuste que cuando se utilizan 3 opciones; además, con 3 categorías se observa que conforme aumenta el tamaño de la muestra aumenta el promedio de la raíz del error cuadrático medio.

Gráfico 4. Promedio de RECM entre los coeficientes estimados del parámetro de dificultad con cada uno de los métodos.



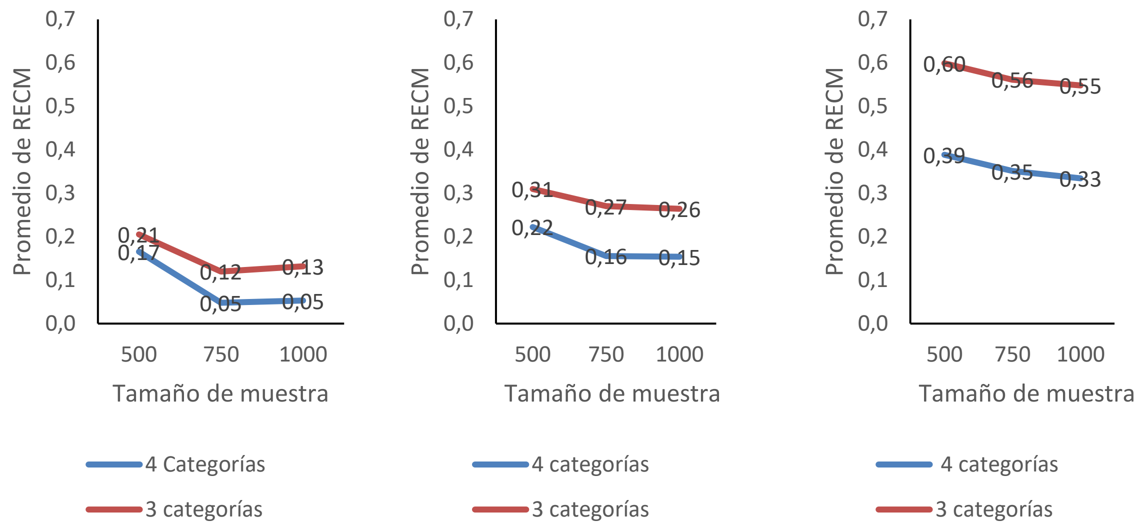
4.2.1.2 Estimación de los coeficientes del parámetro de discriminación

Con respecto a la estimación de los coeficientes de los parámetros de discriminación, en el gráfico 9 se observa que el comportamiento de los tres métodos es muy similar: cuando se utilizan 4 opciones de respuesta se da un mejor ajuste que cuando se utilizan 3 opciones, sin embargo, la convergencia a cero es más notoria utilizando el Método 1. Asimismo, en los 3 métodos se observa que conforme se aumenta el tamaño de la muestra disminuye el promedio de la raíz del error cuadrático medio.

En el método 1 se aprecia que utilizando ya sean 4 o 3 categorías de respuesta y cuando el tamaño de muestra es de 1.000 sujetos aumenta el promedio de la raíz del error cuadrático medio en comparación con el tamaño de muestra de 750, sin embargo, dicho promedio es muy similar, para el caso de 4 categorías de respuesta el valor de la RECM es aproximadamente 0,05 con ambos tamaños de muestra, y cuando se utilizan 3 opciones de respuesta el valor de la RECM pasa de 0,12 con un tamaño de muestra de 750 a 0,13 con una muestra de 1.000.

Gráfico 5. Promedio de RECM entre los coeficientes estimados del parámetro de discriminación con cada uno de los métodos.

Método 1 Método 2 Método 3



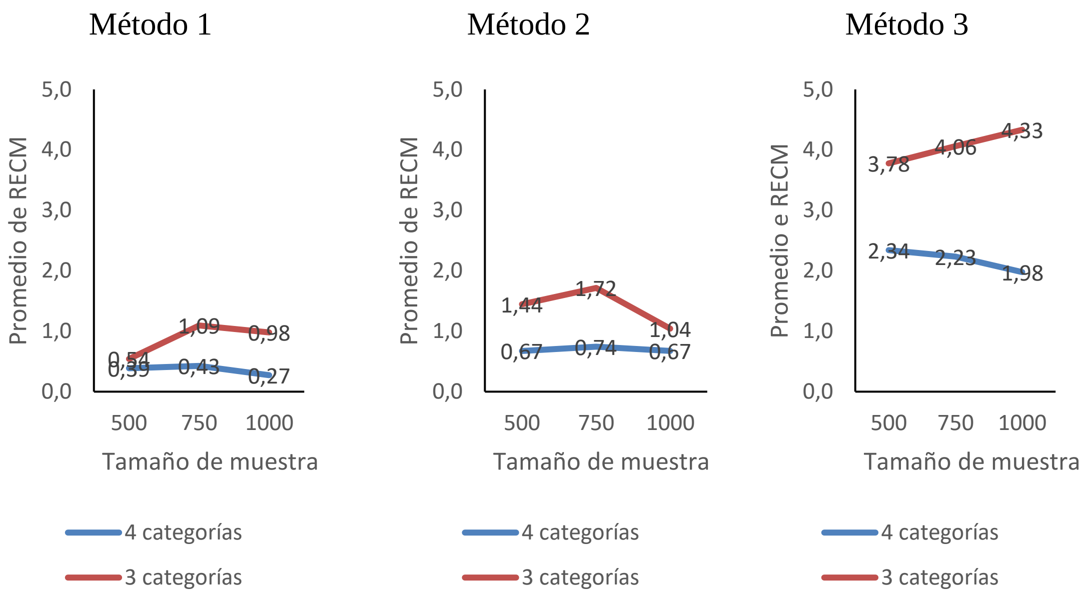
4.2.2 Segundo escenario: con AFE

En este escenario se realizó análisis de factores exploratorio (AFE) para determinar los ítems que estaban generando problemas y se dejaron únicamente los ítems que estaban en el 80 % o más del total de las réplicas. Anterior al cálculo del promedio de la raíz del error cuadrático medio (RECM) en cada uno de los 3 métodos se consideraron únicamente las réplicas donde la estimación de los coeficientes, tanto del parámetro de dificultad como de los parámetros de discriminación, oscilara entre -6 y 6, esto debido a que algunas réplicas presentaron problemas de convergencia en la estimación de los modelos a pesar de la exclusión de los ítems que presentaban problemas según el AFE.

4.2.2.1 Estimación de los coeficientes del parámetro de dificultad

En el gráfico 10 se observa que para los tres métodos se presenta menor promedio de la RECM cuando se utilizan 4 categorías de respuesta en comparación con 3 categorías. También se nota que cuando se utilizan tres opciones de respuesta en el método 1 y 2, aumenta el promedio de RECM cuando el tamaño de muestra pasa de 500 a 750 y para el caso del método 3, el promedio de la RECM con 3 categorías de respuesta va aumentando conforme aumenta el tamaño de muestra. Dicho comportamiento es diferente cuando se utilizan 4 opciones de respuesta ya que, en los tres métodos, dicho valor disminuye conforme aumenta el tamaño de muestra.

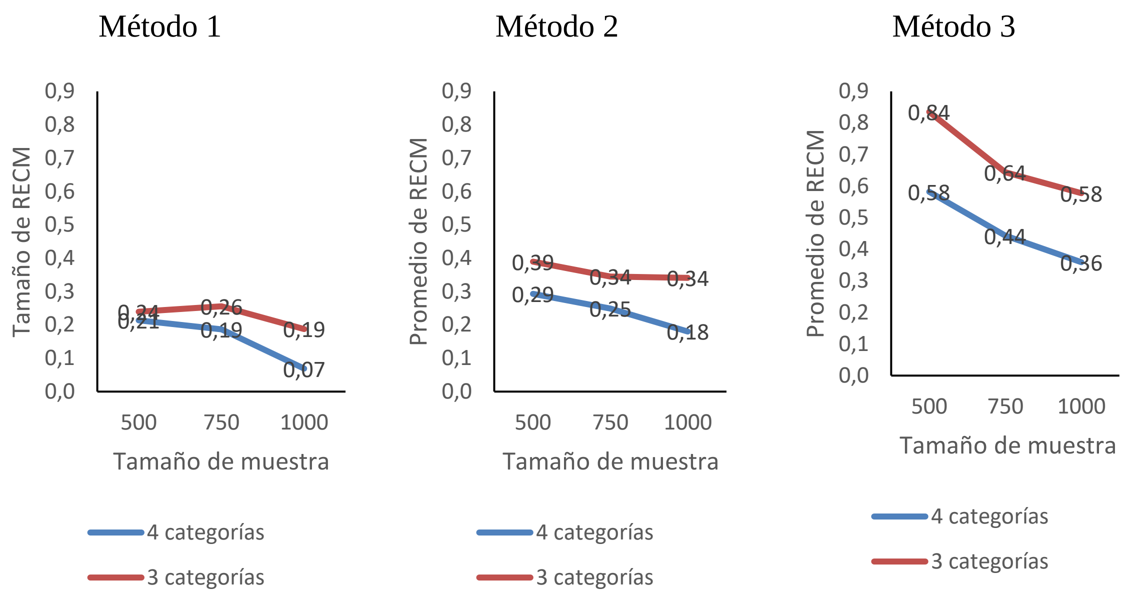
Gráfico 6. Promedio de RECM entre los coeficientes estimados del parámetro de dificultad del escenario 2 con cada uno de los métodos.



4.2.2.2 Estimación de los coeficientes del parámetro de discriminación

El gráfico 11 se muestra que el comportamiento de los tres métodos es muy similar: al utilizar 4 opciones de respuesta se da mejor ajuste que cuando se utilizan 3 opciones, sin embargo, la convergencia a cero es más notoria utilizando el Método 1. También, en los 3 métodos se observa que conforme se aumenta el tamaño de la muestra disminuye el promedio de la raíz del error cuadrático medio.

Gráfico 7. Promedio de RECM entre los coeficientes estimados del parámetro de discriminación del escenario 2 con cada uno de los métodos.



En el cuadro 7 se observan los valores del promedio del criterio de información de Akaike para cada uno de los modelos generados en la simulación. Al comparar en cada uno de los tamaños de muestra, el valor promedio del AIC es menor cuando se usa AFE en comparación con cuando no se usa AFE, donde dicho valor se reduce en más de la mitad. Además, se observa que dicho valor es menor cuando se utiliza el método 1 (asignando la posición de la siguiente pendiente más baja) en comparación con los demás métodos.

Al comparar por la cantidad de categorías de respuesta, el resultado varía dependiendo del tamaño de muestra. Para tamaños de muestra de 500 y 1000, el valor promedio del AIC es menor utilizando 3 categorías en lugar de 4; sin embargo, para el tamaño de muestra de 750, el valor promedio del AIC es menor utilizando 4 categorías en lugar de 3.

Cuadro 6. Promedio del criterio de información de Akaike de cada uno de los modelos generados en la simulación según método y tamaño de muestra.

Modelo	**Tamaño de muestra**					
	500		**750**		**1.000**	
	sin AFE	**con AFE**	**sin AFE**	**con AFE**	**sin AFE**	**con AFE**
5 categorías de respuesta	18.839	9.207	28.253	13.515	37.641	17.573
Siguiente pendiente más baja						
4 categorías de respuesta	16.222	8.243	28.033	**12.681**	**36.705**	17546
3 categorías de respuesta	**15.602**	**8.076**	**27.493**	12.736	36.777	**17482**
Posición aleatoriamente						
4 categorías de respuesta	17.600	8.913	30.516	13.773	40.020	19.034
3 categorías de respuesta	17.998	9.196	31.850	14.412	42.208	19.735
Pendiente más alta						
4 categorías de respuesta	18.692	9.349	32.860	14.487	43.014	20.104
3 categorías de respuesta	16.831	8.499	30.074	13.074	39.790	17.716

V. Conclusiones

Al comparar los 3 métodos utilizados en el estudio de simulación, se observa que cuando se utiliza el método de asignar la posición de la siguiente pendiente más baja, este presentó mejor ajuste en los modelos en comparación con los otros dos métodos, por lo que el método óptimo es cuando el distractor a eliminar es siempre el menos probable de ser seleccionado. Además, el método que presentó en todos los casos valores más altos en RECM fue el método 3, que consistió en asignar siempre la posición de la pendiente más alta, por lo que este método no se debe considerar como una opción en futuras investigaciones.

Se determinó que se presentan mejores resultados en los ajustes de los modelos cuando se realiza análisis de factores exploratorio en comparación con cuando no se realiza el AFE, independientemente del tamaño de muestra y del método empleado. Por lo tanto, es importante utilizarlo para determinar los ítems que realmente aportan información a la habilidad o rasgo latente que está siendo evaluada y utilizar ítems o preguntas que contribuyan con un único factor en el modelo y ofrezcan mayor precisión de los parámetros de los ítems.

Los coeficientes estimados del parámetro de discriminación presentaban en promedio valores de la RECM más cercanos a cero en los tres métodos utilizados y en los dos escenarios (sin AFE y con AFE); de igual forma, en la mayoría de los casos se observó que conforme se aumentaba el tamaño de muestra el promedio de la RECM disminuía. Además, se evidenció que dicho valor siempre fue más pequeño cuando se utilizan 4 categorías de respuesta en comparación con cuando se utilizan 3 categorías.

Con respecto a la estimación del parámetro de dificultad, el comportamiento observado refleja que utilizar 4 opciones de respuesta posee mejor ajuste en comparación con cuando se utilizan 3 opciones, independientemente del tamaño de la muestra y del escenario utilizado. Adicionalmente, se observó que en el método 1, cuando no se utiliza AFE, ya sea con 4 o 3 categorías de respuesta y cuando se pasaba de 750 a un tamaño de muestra de 1.000 sujetos, aumentaba el valor de la RECM, sin embargo, dicha diferencia era menor a 0,05 en ambos casos.

Asimismo, se encontró que, para todos los coeficientes calculados de dificultad y discriminación en el primer escenario, el promedio de la raíz del error cuadrático medio converge a cero a medida que se aumenta el tamaño de muestra en cada réplica. Se determinó que utilizando 4 categorías de respuesta se ajustan mejor los coeficientes en ambos escenarios en comparación con utilizar 3 categorías de respuesta.

De acuerdo con los resultados de la metodología de simulación empleada, basada en datos reales, mediante la TRI con el modelo de respuesta nominal, lo más adecuado es utilizar 4 opciones de respuesta, sin embargo, en el estudio de Rodriguez (2005), hay otra perspectiva mediante metodologías de la TCT donde más bien se recomienda utilizar 3 opciones de respuesta, sin embargo, dichas metodologías y enfoques utilizados son diferentes.

Es importante considerar su implementación en la reducción de opciones de categorías de respuesta en pruebas estandarizadas, ya que esta es una metodología complementaria, novedosa y útil.

Bibliografía

Asún, R., y Zúñiga, C. (2008). Ventajas de los Modelos Politómicos de Teoría de Respuesta al Ítem en la Medición de Actitudes Sociales. El Análisis de un Caso. *Psykhe*, 12(2), pp. 103-115. http://dx.doi.org/10.4067/S0718-22282008000200009

Attorresi, H. F., Lozzia, G. S., Abal, F. J., Galibert, M. S., y Aguerri, M. E. (2009). Teoría de respuesta al ítem. Conceptos básicos y aplicaciones para la medición de constructos psicológicos. *Revista Argentina de Clínica Psicológica*, 18(2), pp. 179-188. https://www.redalyc.org/pdf/2819/281921792007.pdf

Baghaei, P., y Amrahi, N. (2011). The effects of the number of options on the psychometric characteristics of multiple choice items. *Psychological Test and Assessment Modeling*, 53(2), pp. 192-211. https://www.researchgate.net/publication/266737879_The_effects_of_the_number_of_options_on_the_psychometric_characteristics_of_multiple_choice_items

Chalmers, R. P. (2012). mirt: A Multidimensional Item Response Theory Package for the R Environment. *Journal of Statistical Software*, 48(6), 1-29. https://doi.org/10.18637/jss.v048.i06

Hernández Rodríguez, O. (2013). *Temas de análisis estadístico multivariante*. Editorial UCR.

Hidalgo Montesinos, M. D., y French, B. F. (2016). Una introducción didáctica a la Teoría de Respuesta al Ítem para comprender la construcción de escalas. *Revista de Psicología Clínica con Niños y Adolescentes*, 3(2), pp. 13-21. https://dialnet.unirioja.es/servlet/articulo?codigo=5590670

Jiménez Alfaro, K., y Montero Rojas, E. (2013). Aplicación del modelo de Rasch, en el análisis psicométrico de una prueba. *Revista digital Matemática, Educación e Internet*, 13(1). https://tecdigital.tec.ac.cr/revistamatematica/ARTICULOS_V13_N1_2012/RevistaDigital_Montero_V13_n1_2012/Screen_RevistaDigital_Montero_V13_n1_2012.pdf

Martínez Arias, M. R., Hernández Lloreda, M. V., y Hernández Lloreda, M. J. (2006). *Psicometría*. Alianza Editorial.

Montero Rojas, E. (2000). Teoría de Respuesta a los Ítemes versus Tería Clásica de los Test: Análisis empírico comparativo. *Revista Educación*, 24(2), pp. 183-188. https://www.researchgate.net/publication/277236856_Teoria_de_respuesta_a_los_itemes_versus_teoria_clasica_de_los_tests_analisis_empirico_comparativo

Montero Rojas, E. (2010). Excelencia y equidad en prueba de admisión una propuesta emergente para la Universidad de Costa Rica. *Actualidades Investigativas en Educación*, pp. 1-19. https://revistas.ucr.ac.cr/index.php/aie/article/view/10119/17971

Montero Rojas, E., Rojas Rojas, G., Francis Salazar, S., y Negrín Hernández, M. (2015). Efecto de una capacitación sobre los puntajes de la prueba de admisión de la Universidad de Costa Rica: una aproximación bayesiana. *Actualidades en Psicología*, 10(2), pp. 115-130. http://dx.doi.org/10.15517/ap.v29i119.19283

Muñiz, J. (2010). Las teorías de los tests: teoría clásica y teoría de respuesta a los ítems. *Papeles del Psicólogo*, 31(1), pp. 57-66. https://dialnet.unirioja.es/servlet/articulo?codigo=3150824

Muñiz, J. (2018). Introducción a la Psicometría Teoría clásica y TRI. Ediciones Pirámide.

Pérez Gil, J. A. (2004). Modelos para ítems politómicos de respuestas discretas. Departamento de Psicología Experimental, Universidad de Sevilla.

Preston, K., Reise, S., Cai, L., y Hays, R. D. (2011). Using the Nominal Response Model to Evaluate Response Category Discrimination in the PROMIS Emotional Distress Item Pools. *Educational and Psychological Measurement*, 71(3), pp. 523-550. https://doi.org/10.1177/0013164410382250

Rivera Espejo, J. M. (2019). *El Modelo de Respuesta Nominal: Aplicación a datos educacionales*. [Tesis inédita Para optar el título de Magíster en Estadística]. Pontificia Universidad Católica del Perú. http://hdl.handle.net/20.500.12404/14600

Rizopoulos, D. (2006). ltm: An R Package for Latent Variable Modeling and Item Response Theory Analyses. *Journal of Statistical Software*, 17(5), pp. 1-25. https://doi.org/10.18637/jss.v017.i05

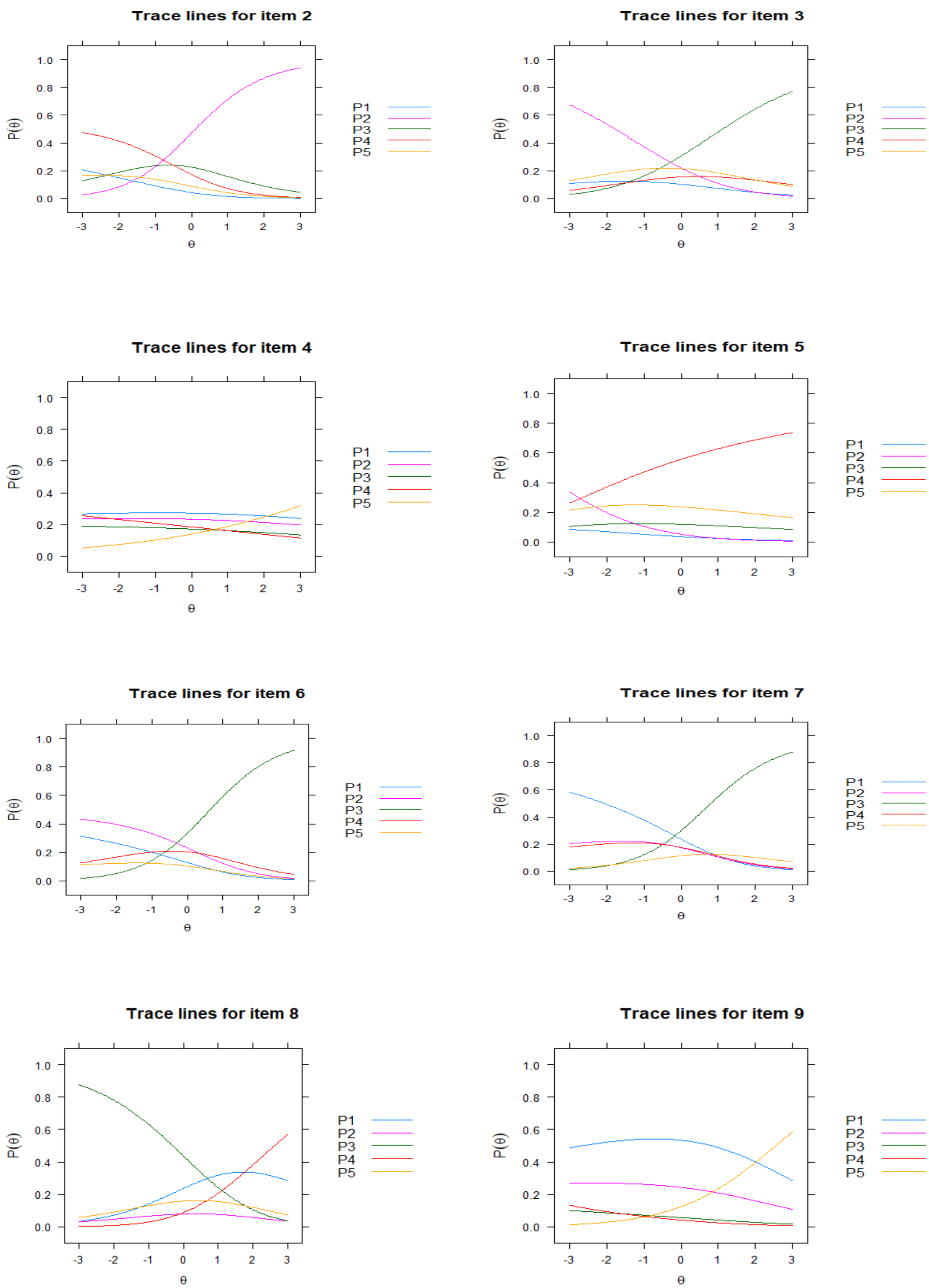
Rodríguez, M. C. (2005). Three options are optimal for multiple-choice items: A meta-analysis of 80 years of research. *Educational Measurement*, 3(1), pp. 3-13.

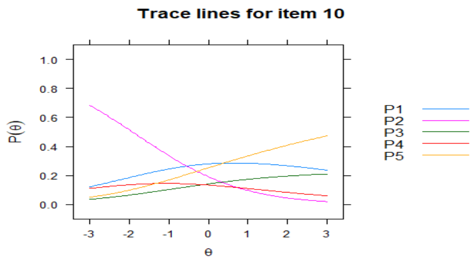
https://doi.org/10.1111/j.1745-3992.2005.00006.x

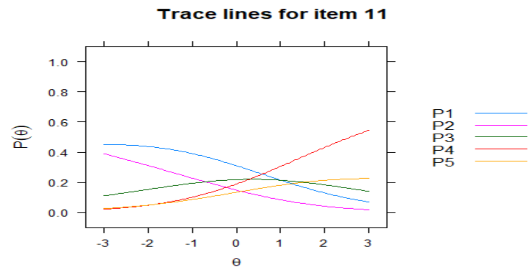
Ximénez, M. C., y Revuelta, J. (1999). Aplicación de un modelo politómico de TRI para la obtención de medidas conmensurables en una escala de ajuste persona-organización. *Psicológica*, 20(2), pp. 135-150. https://dialnet.unirioja.es/servlet/articulo?codigo=72281

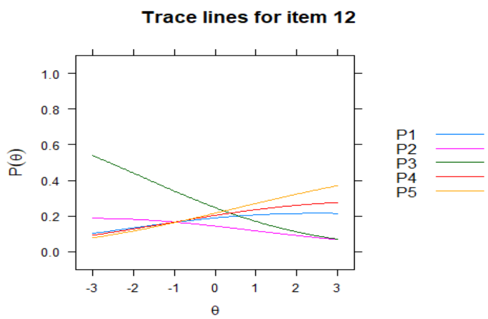
Anexos

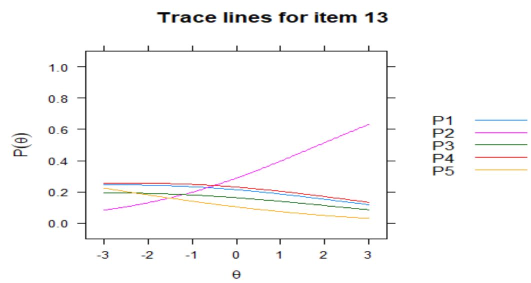
Anexo 1. Curva Característica del Ítem

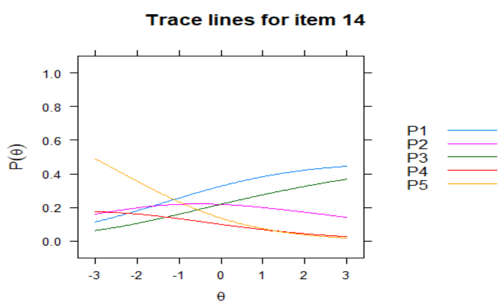


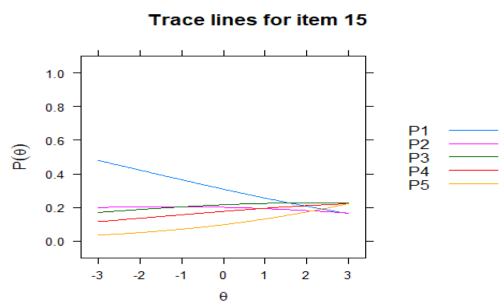

Trace lines for item 10
P(θ)
θ
P1
P2
P3
P4
P5

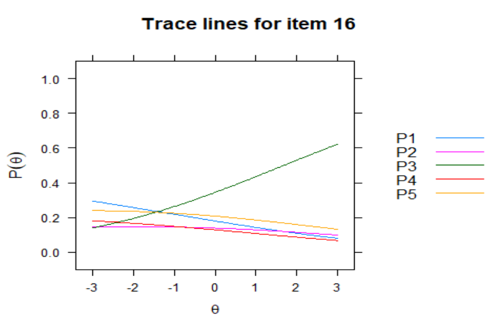

Trace lines for item 11
P(θ)
θ
P1
P2
P3
P4
P5

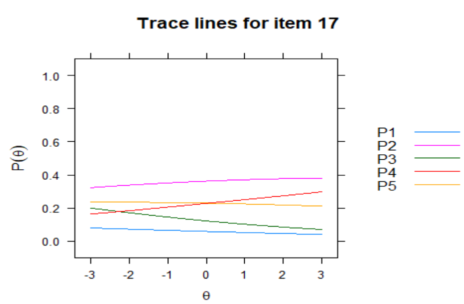

Trace lines for item 12
P(θ)
θ
P1
P2
P3
P4
P5

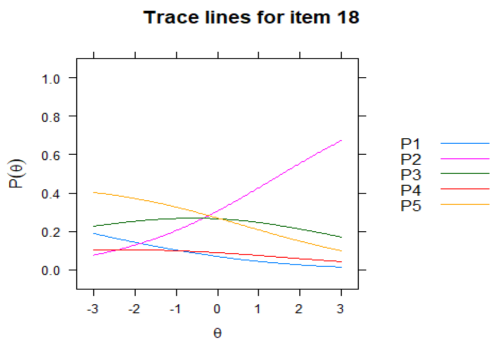

Trace lines for item 13
P(θ)
θ
P1
P2
P3
P4
P5

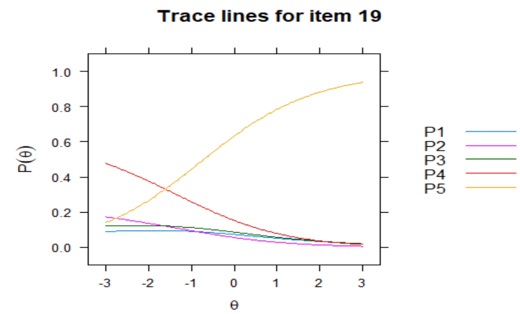

Trace lines for item 14
P(θ)
θ
P1
P2
P3
P4
P5

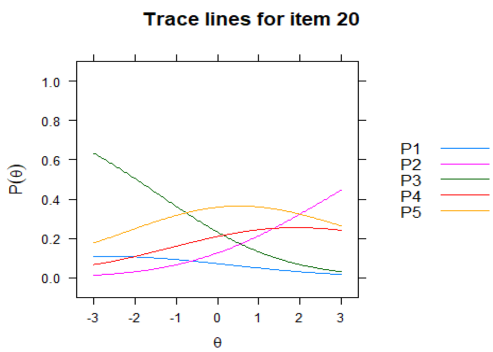

Trace lines for item 15
P(θ)
θ
P1
P2
P3
P4
P5

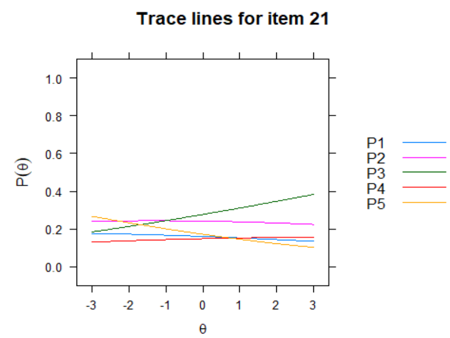

Trace lines for item 16
P(θ)
θ
P1
P2
P3
P4
P5

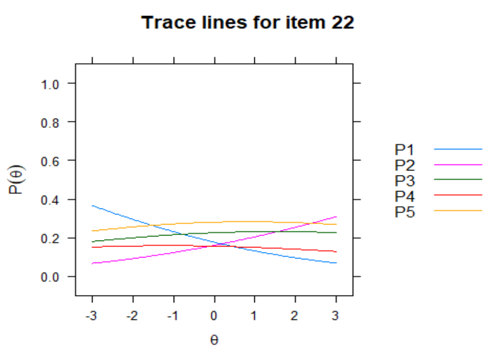

Trace lines for item 17
P(θ)
θ
P1
P2
P3
P4
P5

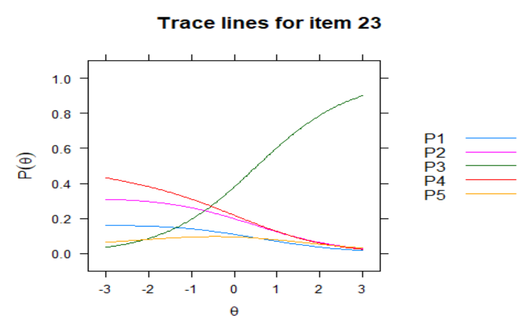

Trace lines for item 18
P(θ)
θ
P1
P2
P3
P4
P5

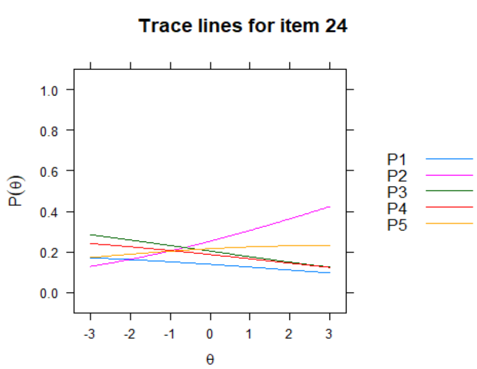

Trace lines for item 19
P(θ)
θ
P1
P2
P3
P4
P5

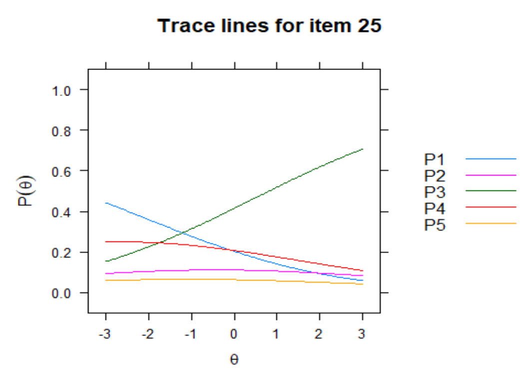

Trace lines for item 20
P(θ)
θ
P1
P2
P3
P4
P5

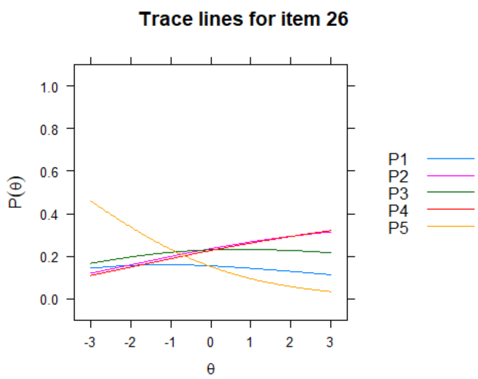

Trace lines for item 21
P(θ)
θ
P1
P2
P3
P4
P5

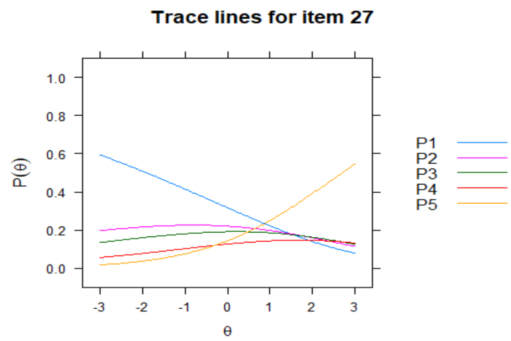

Trace lines for item 22
P(θ)
θ
P1
P2
P3
P4
P5

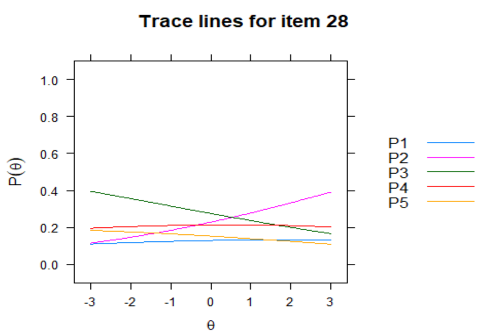

Trace lines for item 23
P(θ)
θ
P1
P2
P3
P4
P5

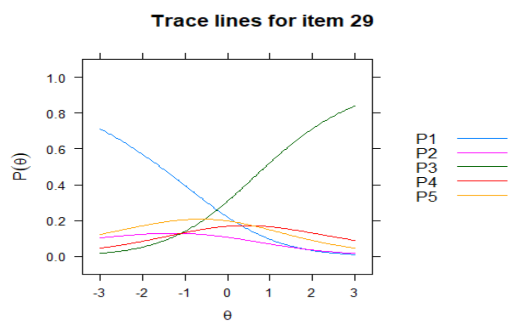

Trace lines for item 24
P(θ)
θ
P1
P2
P3
P4
P5

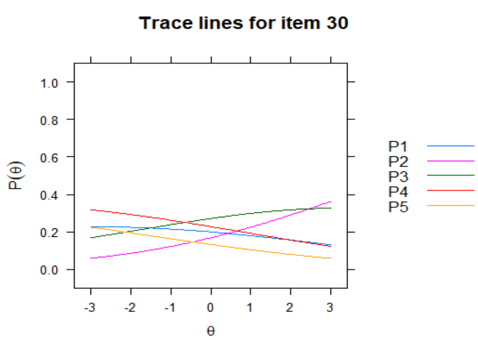

Trace lines for item 25
P(θ)
θ
P1
P2
P3
P4
P5

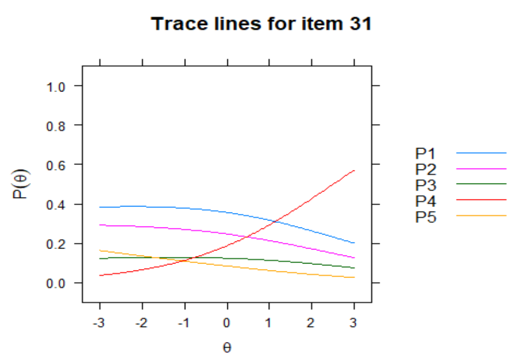

Trace lines for item 26
1.0
0.8
0.6
0.4
0.2
0.0
P(θ)
-3 -2 -1 0 1 2 3
θ
P1
P2
P3
P4
P5

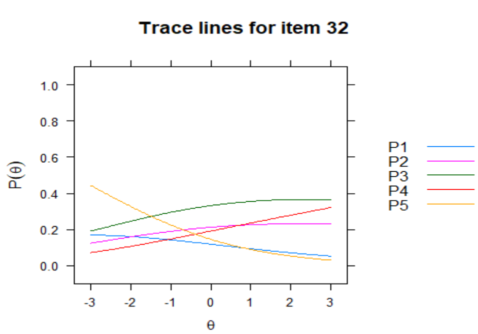

Trace lines for item 27
1.0
0.8
0.6
0.4
0.2
0.0
P(θ)
-3 -2 -1 0 1 2 3
θ
P1
P2
P3
P4
P5

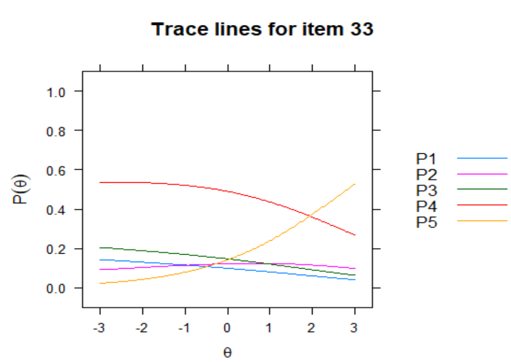

Trace lines for item 28
1.0
0.8
0.6
0.4
0.2
0.0
P(θ)
-3 -2 -1 0 1 2 3
θ
P1
P2
P3
P4
P5

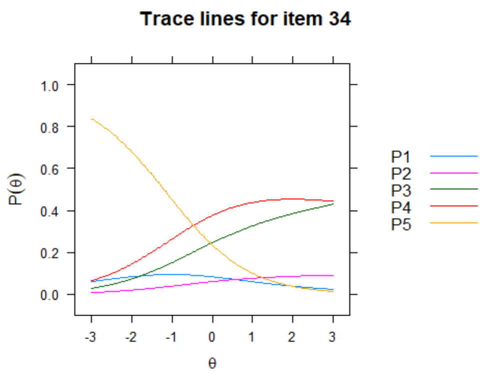

Trace lines for item 29
1.0
0.8
0.6
0.4
0.2
0.0
P(θ)
-3 -2 -1 0 1 2 3
θ
P1
P2
P3
P4
P5

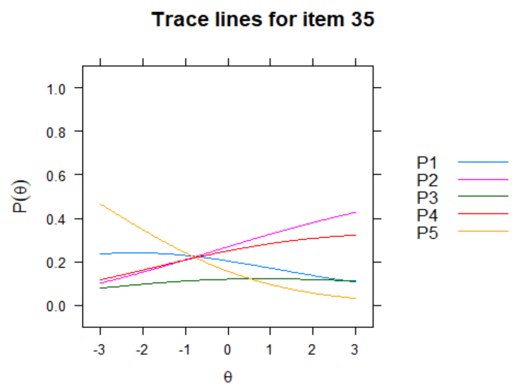

Trace lines for item 30
1.0
0.8
0.6
0.4
0.2
0.0
P(θ)
-3 -2 -1 0 1 2 3
θ
P1
P2
P3
P4
P5


Trace lines for item 31
1.0
0.8
0.6
0.4
0.2
0.0
P(θ)
-3 -2 -1 0 1 2 3
θ
P1
P2
P3
P4
P5

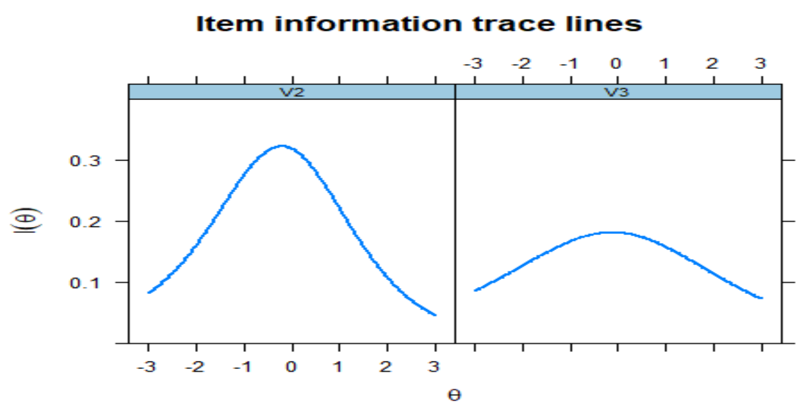

Trace lines for item 32
1.0
0.8
0.6
0.4
0.2
0.0
P(θ)
-3 -2 -1 0 1 2 3
θ
P1
P2
P3
P4
P5

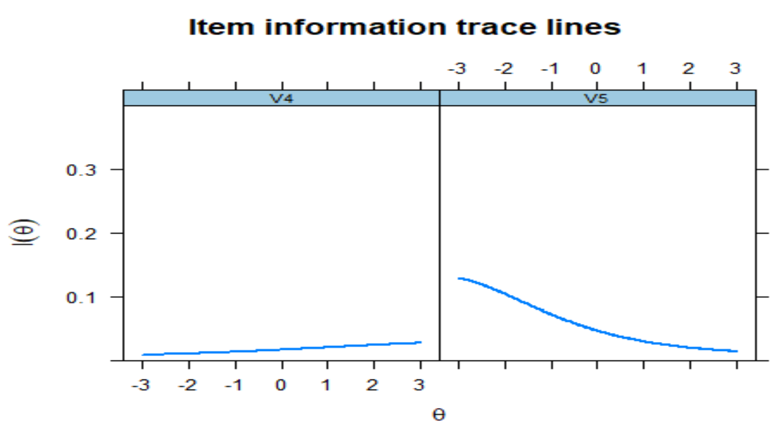

Trace lines for item 33
1.0
0.8
0.6
0.4
0.2
0.0
P(θ)
-3 -2 -1 0 1 2 3
θ
P1
P2
P3
P4
P5

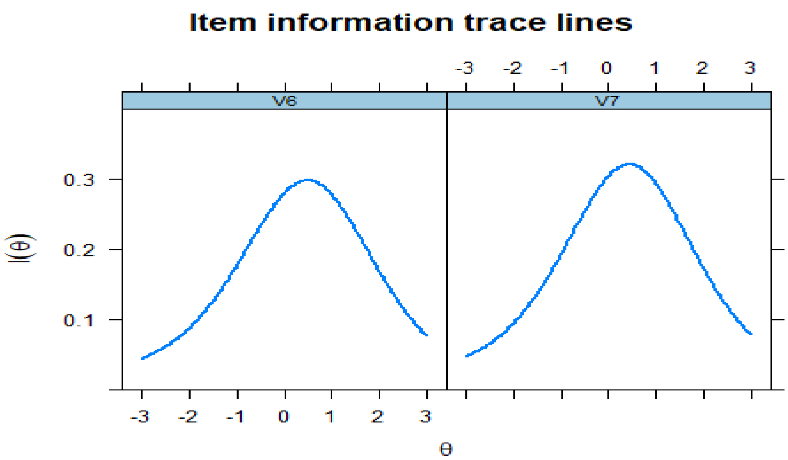


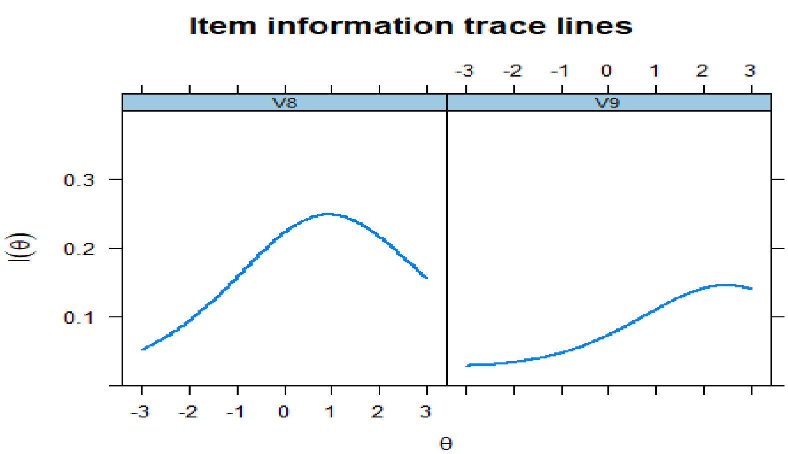


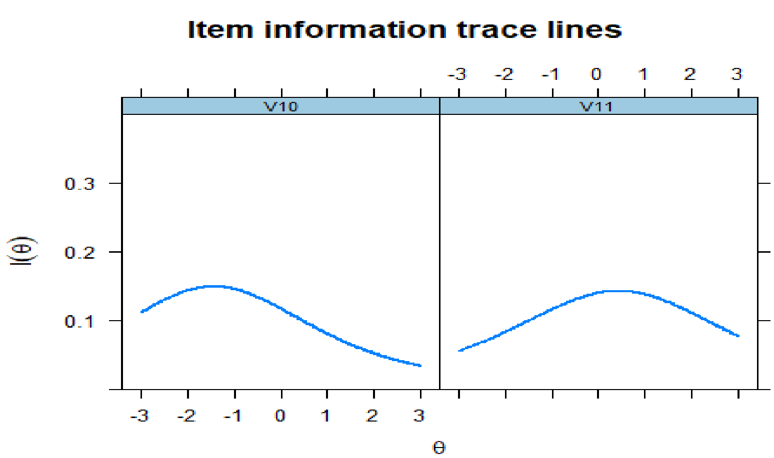
Anexo 2. Función de información de los ítems.

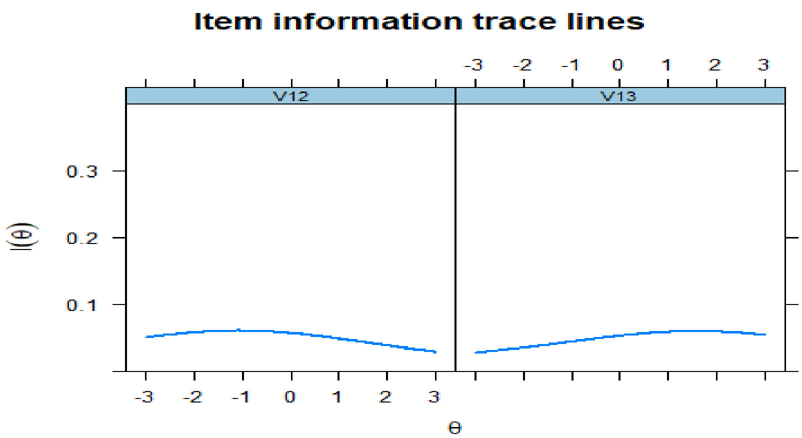


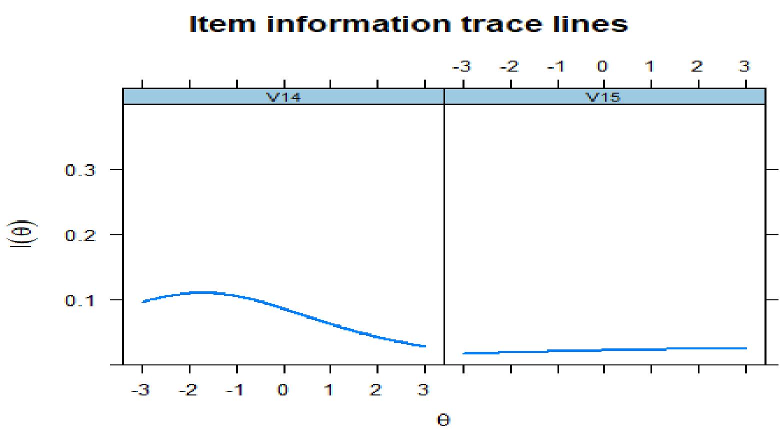


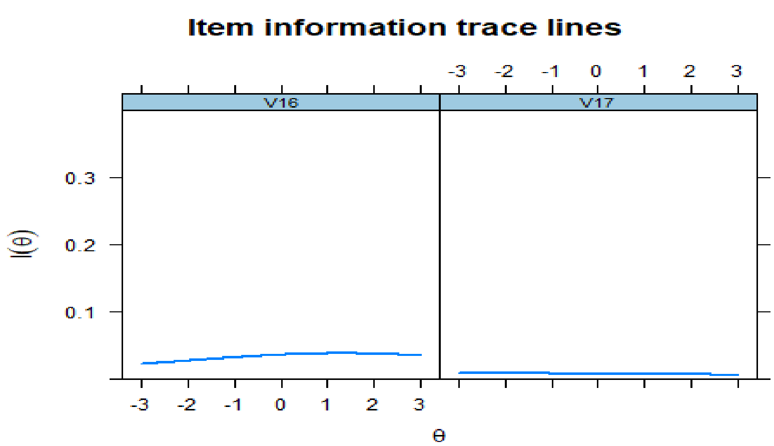

Item information trace lines
V6
V7
I(θ)
θ

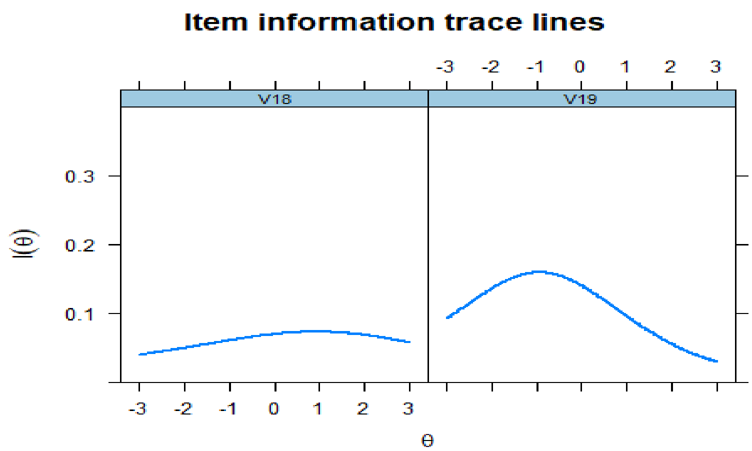

Item information trace lines
V8
V9
I(θ)
θ

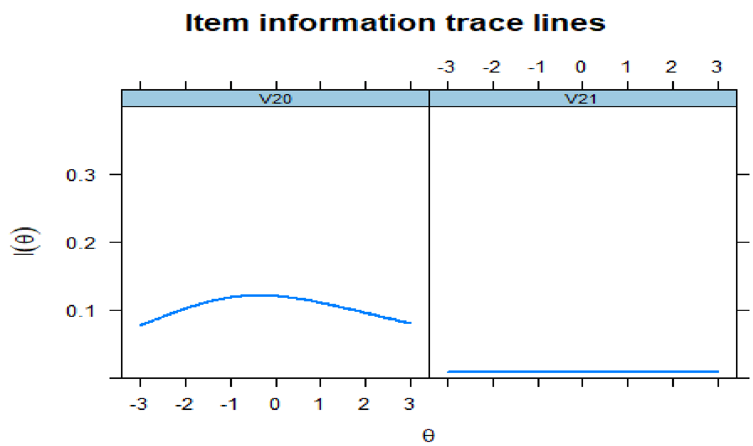

Item information trace lines
V10
V11
I(θ)
θ

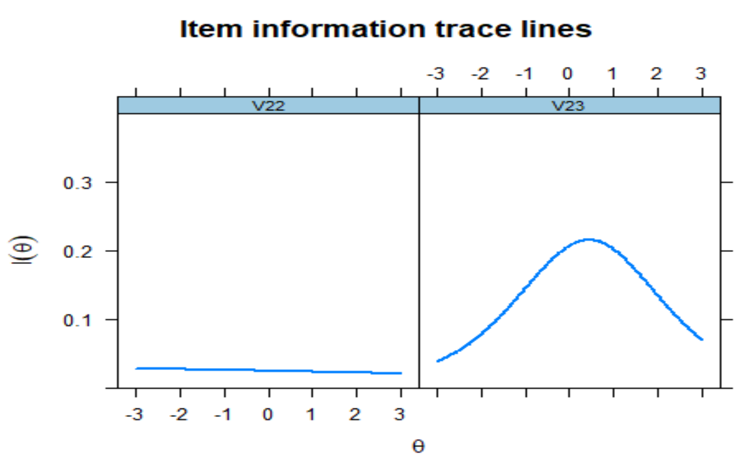

Item information trace lines
V12
V13
I(θ)
0.3
0.2
0.1
-3
-2
-1
0
1
2
3
θ

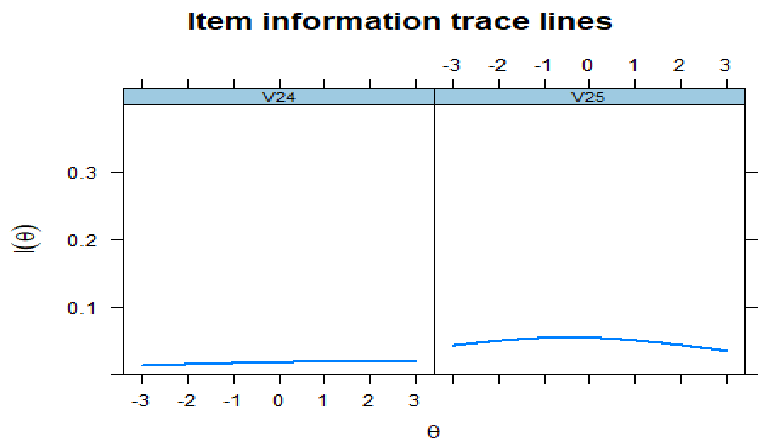

Item information trace lines
V14
V15
I(θ)
0.3
0.2
0.1
-3
-2
-1
0
1
2
3
θ

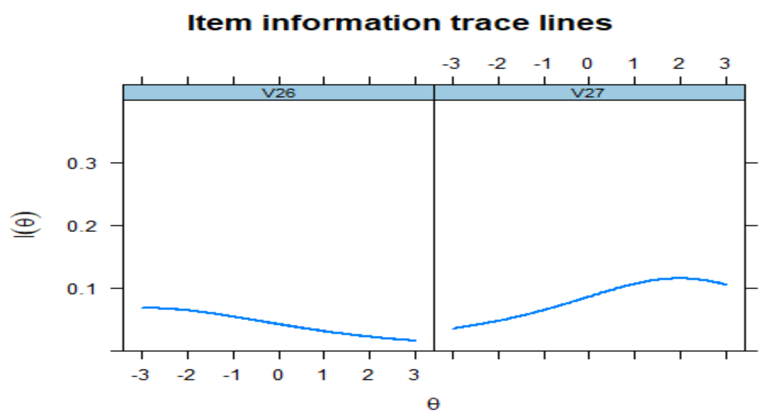

Item information trace lines
V16
V17
I(θ)
0.3
0.2
0.1
-3
-2
-1
0
1
2
3
θ

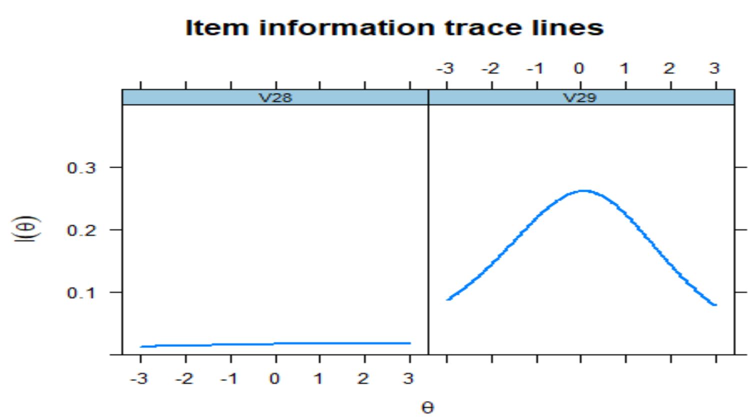

Item information trace lines
V18
V19
I(θ)
0.1
0.2
0.3
-3
-2
-1
0
1
2
3
θ

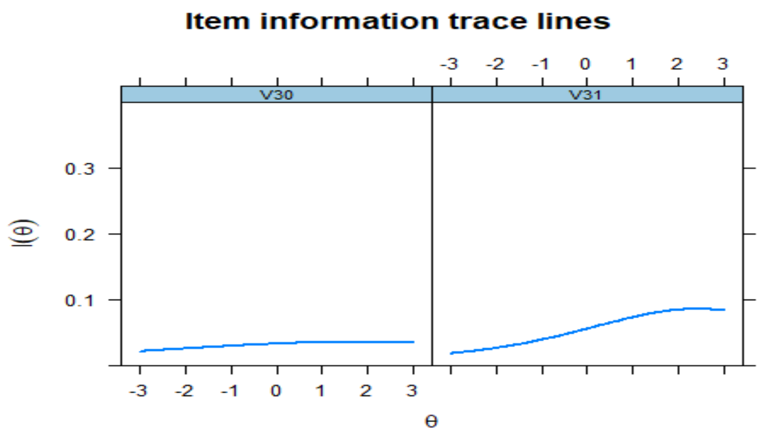

Item information trace lines
V20
V21
I(θ)
0.1
0.2
0.3
-3
-2
-1
0
1
2
3
θ

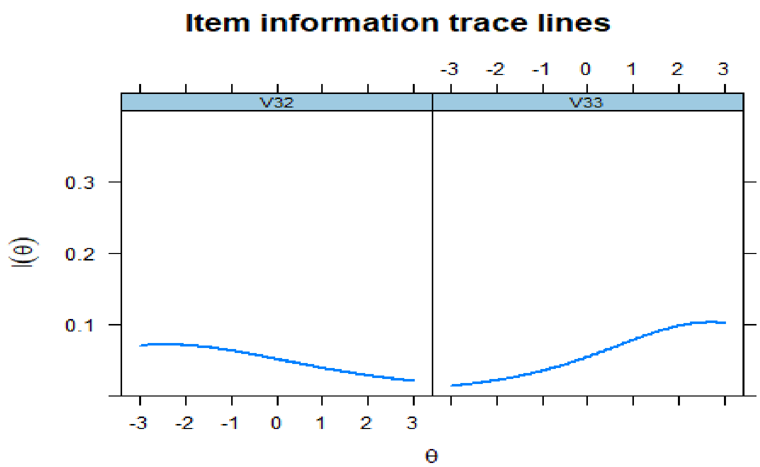

Item information trace lines
V22
V23
I(θ)
0.1
0.2
0.3
-3
-2
-1
0
1
2
3
θ

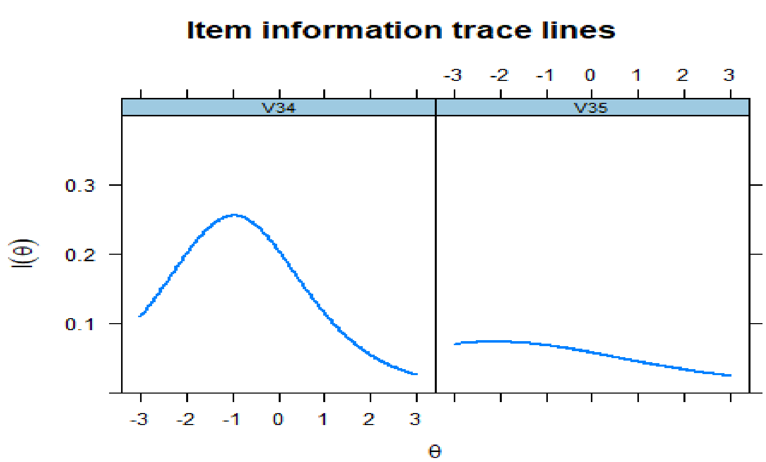

Item information trace lines
V24
V25
I(θ)
0.1
0.2
0.3
-3 -2 -1 0 1 2 3
θ


Item information trace lines
V26
V27
I(θ)
0.1
0.2
0.3
-3 -2 -1 0 1 2 3
θ


Item information trace lines
V28
V29
I(θ)
0.1
0.2
0.3
-3 -2 -1 0 1 2 3
θ


Item information trace lines
V30
V31
I(θ)
0.3
0.2
0.1
-3 -2 -1 0 1 2 3
θ


Item information trace lines
V32
V33
I(θ)
0.3
0.2
0.1
-3 -2 -1 0 1 2 3
θ


Item information trace lines
V34
V35
I(θ)
0.3
0.2
0.1
-3 -2 -1 0 1 2 3
θ

Anexo 3. Código de la simulación

```
#librerias
library(mirt)
library(dplyr)
library(ltm)
library(purrr)
library(stats)
library(openxlsx)
#------------------------ leer bases ---------------------------------------------#
base <- read.table("prueba_leticia.txt",header = F)
base1 <- base
base1 <- replace(base1,base1==8,0) # un? opcion 8 con la de 0 xq son muy pocos casos
base1 <- replace(base1,base1==0,6)
sum(base1 == "0")
sum(base1 == "1")
sum(base1 == "2")
sum(base1 == "3")
sum(base1 == "4")
sum(base1 == "5")
sum(base1 == "6")
rowSums (base1[ , 1:35]=="6")
colSums (base1[ , 1:35]=="6")

base2 <- base1
base2[base2 == 6] <- NA
base3 <- na.omit(base2)
sum(base3 == "0")
sum(base3 == "1")
sum(base3 == "2")
sum(base3 == "3")
sum(base3 == "4")
sum(base3 == "5")
sum(base3 == "6")

# Pasa de 1467 registros a 1263, habian 204 registros con valor 6
base1 <- base3
save(base1, file="base1.rda")
load("base1.rda")
#-------------------------------- Opcion correcta ---------------------------------------#
opcion1 <- read.table("cla.txt",header = F)
table(opcion1)
item <- length(base1) #35
opcion<- rep('nominal',item)
opcion1 <- mutate(opcion1,id=1:item)
########################## Generacion de datos
set.seed(8010)
#numero de items
items<-length(base1)
#numero de examinados
n<-500 # cambiar por 500, 750 y 1000
#numero de categorias de respuesta
opciones<-5
replica <- 100
#--------------------------------generando las bases----------------------------------------#
muestra.base<- list()
muestra<- list()
opcion.muestra <- list()
for (m in 1:replica){
```

```
  muestra.base[[m]]<- sample(1:nrow(base1),size=n,replace=FALSE)
  muestra[[m]] <- base1[muestra.base[[m]],]
  opcion.muestra[[m]] <- opcion1
}
save(muestra, file = "muestra500.rda")
save(muestra, file = "muestra750.rda")
save(muestra, file = "muestra1000.rda")
save(opcion.muestra, file = "opcion.muestra.rda")
load("opcion.muestra.rda")
load("muestra500.rda")
#----------- Carpetas -------------------#
#-------- 500 -----------
setwd("C:/Users/leticia.vasquezg/Documents/Practica/N 500")
load("muestra500.rda")
load("opcion.muestra.rda")
n<-500
opciones<-5
replica <- 100
items<-35
save(mod, file = "mod.500.rda")
save(coef.mod, file = "coef.mod.500.rda")
save(pend, file = "pend.500.rda")
save(inter, file = "inter.500.rda")
load("mod.500.rda")
load("coef.mod.500.rda")
load("pend.500.rda")
load("inter.500.rda")
#------------ 750
setwd("C:/Users/leticia.vasquezg/Documents/Practica/N 750")
load("muestra750.rda")
load("opcion.muestra.rda")
n<-750
opciones<-5
replica <- 100
items<-35
save(mod, file = "mod.750.rda")
save(coef.mod, file = "coef.mod.750.rda")
save(pend, file = "pend.750.rda")
save(inter, file = "inter.750.rda")
load("mod.750.rda")
load("coef.mod.750.rda")
load("pend.750.rda")
load("inter.750.rda")
#--------------- 1000
setwd("C:/Users/leticia.vasquezg/Documents/Practica/N 1000")
load("muestra1000.rda")
load("opcion.muestra.rda")
n<-1000
opciones<-5
replica <- 100
items<-35
save(mod, file = "mod.1000.rda")
save(coef.mod, file = "coef.mod.1000.rda")
save(pend, file = "pend.1000.rda")
save(inter, file = "inter.1000.rda")

load("mod.1000.rda")
load("coef.mod.1000.rda")
load("pend.1000.rda")
load("inter.1000.rda")
```

```
#-------------------------------------
#genrando el modelo
mod <- list()
coef.mod <- list()
pend <- list()
inter <- list()

t <- proc.time()
for (v in 1:replica){
  #generando el modelo TRI
  mod[[v]]<-mirt(muestra[[v]],1, itemtype=rep('nominal',items),draws=5000)

  #extrayendo los coeficientes
  coef.mod[[v]]<-   as.data.frame(coef(mod[[v]],   IRTpars=TRUE,   simplify=TRUE)$items,method   =
"EM",draws = 10000) #IRTpars, Convertir parámetros de intersección de pendiente en parámetros
tradicionales de IRT
  pend[[v]]<-coef.mod[[v]][1:5]
  inter[[v]]<-coef.mod[[v]][6:10]
  colnames(pend[[v]]) <- 1:5

}
proc.time() - t

#colnames(muestra[[1]]) <- 1:items
#a<-mirt(muestra[[1]],1, itemtype=rep('nominal',items),draws=5000,key=opcion.muestra[[1]][,1])
#b<- as.data.frame(coef(a, IRTpars=TRUE, simplify=TRUE)$items,method = "EM",draws = 10000)
#c <- b[1:5]
#apply(c,1,function(x) names(sort(x)))
#c(opcion1)
#opcion.muestra[[1]][1,]
#itemplot(a,1,theta_lim = c(-3,3))

####
opcion.muestra[[1]][1,]
itemplot(mod[[1]],1,theta_lim = c(-3,3))

#####
pend2<-mapply(function(data.pend){
  apply(data.pend,1,function(x) names(sort(x)))},
  data.pend=pend, SIMPLIFY = FALSE)

pend3<-mapply(function(data.pend){
  apply(data.pend,1,function(x) names(sort(x,decreasing = TRUE)))},
  data.pend=pend, SIMPLIFY = FALSE)

pend3[[1]][,1]

pend.order2<-pend3
pend.order2[[5]]


#------------------------- Sin quitar los items malos con el AF -----------------------------#

#---------------------- GENERANDO LAS BASE DE 4 Y 3 OPCIONES ----------------------------------#

#----------------- Usando la siguiente opcion más baja --------------------------#

#Pasando a 4 opciones
base4 <- muestra
for (c in 1:replica){
```

```
  for (j in 1:ncol(base4[[c]])){
    for(r in 1:n){
      if                 (base4[[c]][r,j]==as.numeric(pend.order2[[c]][,j][1])){base4[[c]][r,j]<-
as.numeric(pend.order2[[c]][,j][2])}else
        if (base4[[c]][r,j]!=as.numeric(pend.order2[[c]][,j][1])){base4[[c]][r,j]}
    }
  }
}

pend.order2[[1]][,1]
table(muestra[[1]][,1]);table(base4[[1]][,1])

#Pasando a 3 opciones
base3 <- base4
for (e in 1:replica){
  for (k in 1:ncol(base3[[e]])){
    for(m in 1:n){
      if                 (base3[[e]][m,k]==as.numeric(pend.order2[[e]][,k][2])){base3[[e]][m,k]<-
as.numeric(pend.order2[[e]][,k][3])}else
        if (base3[[e]][m,k]!=as.numeric(pend.order2[[e]][,k][2])){base3[[e]][m,k]}
    }
  }
}

pend.order2[[1]][,1]
table(muestra[[1]][,1]);table(base4[[1]][,1]);table(base3[[1]][,1])
opcion.muestra[[1]][1,]
#---------------------------- Opcion 2 mayor proba -------------------------------------------#
#Pasando a 4 opciones
base4.2 <- muestra
for (c in 1:replica){
  for (j in 1:ncol(base4.2[[c]])){
    for(r in 1:n){
      if             (base4.2[[c]][r,j]==as.numeric(pend.order2[[c]][,j][1])){base4.2[[c]][r,j]<-
as.numeric(pend.order2[[c]][,j][5])}else
        if (base4.2[[c]][r,j]!=as.numeric(pend.order2[[c]][,j][1])){base4.2[[c]][r,j]}
    }
  }
}

pend.order2[[1]][,1]
table(muestra[[1]][,1]);table(base4.2[[1]][,1])

#Pasando a 3 opciones
base3.2 <- base4.2
for (e in 1:replica){
  for (k in 1:ncol(base3.2[[e]])){
    for(m in 1:n){
      if             (base3.2[[e]][m,k]==as.numeric(pend.order2[[e]][,k][2])){base3.2[[e]][m,k]<-
as.numeric(pend.order2[[e]][,k][5])}else
        if (base3.2[[e]][m,k]!=as.numeric(pend.order2[[e]][,k][2])){base3.2[[e]][m,k]}
    }
  }
}

pend.order2[[1]][,1]
table(muestra[[1]][,1]);table(base4.2[[1]][,1]);table(base3.2[[1]][,1])
opcion.muestra[[1]][1,]

#--------------------------------- Opcion 3 aleatoreamente ------------------------------------#
```

```
#Pasando a 4 opciones
base4.3 <- muestra
for (c in 1:replica){
  for (j in 1:ncol(base4.3[[c]])){
    for(r in 1:n){
      if              (base4.3[[c]][r,j]==as.numeric(pend.order2[[c]][,j][1])){base4.3[[c]][r,j]<-
sample(as.numeric(pend.order2[[c]][,j][2:5]),1)}else
        if (base4.3[[c]][r,j]!=as.numeric(pend.order2[[c]][,j][1])){base4.3[[c]][r,j]}
    }
  }
}

pend.order2[[1]][,1]
table(muestra[[1]][,1]);table(base4.3[[1]][,1])

#Pasando a 3 opciones
base3.3 <- base4.3
for (e in 1:replica){
  for (k in 1:ncol(base4[[e]])){
    for(m in 1:n){
      if              (base3.3[[e]][m,k]==as.numeric(pend.order2[[e]][,k][2])){base3.3[[e]][m,k]<-
sample(as.numeric(pend.order2[[e]][,k][3:5]),1)}else
        if (base3.3[[e]][m,k]!=as.numeric(pend.order2[[e]][,k][2])){base3.3[[e]][m,k]}
    }
  }
}

pend.order2[[1]][,1]
table(muestra[[1]][,1]);table(base4.3[[1]][,1]);table(base3.3[[1]][,1])
opcion.muestra[[1]][1,]


#-------------------------- Dicotomizando las bases --------------------------------------#
base5   <- list()
base4pb <- list()
base3pb <- list()
base4pm <- list()
base3pm <- list()
base4a  <- list()
base3a  <- list()

for (w in 1:replica){
  base5[[w]]   <- key2binary(muestra[[w]], key=opcion.muestra[[w]][1], score_missing = FALSE)
  base4pb[[w]] <- key2binary(base4[[w]], key=opcion.muestra[[w]][1], score_missing = FALSE)
  base3pb[[w]] <- key2binary(base3[[w]], key=opcion.muestra[[w]][1], score_missing = FALSE)
  base4pm[[w]] <- key2binary(base4.2[[w]], key=opcion.muestra[[w]][1], score_missing = FALSE)
  base3pm[[w]] <- key2binary(base3.2[[w]], key=opcion.muestra[[w]][1], score_missing = FALSE)
  base4a[[w]]  <- key2binary(base4.3[[w]], key=opcion.muestra[[w]][1], score_missing = FALSE)
  base3a[[w]]  <- key2binary(base3.3[[w]], key=opcion.muestra[[w]][1], score_missing = FALSE)
}

prop.table(table(base5[[1]]))
prop.table(table(base4pb[[1]]))
prop.table(table(base3pb[[1]]))
prop.table(table(base4pm[[1]]))
prop.table(table(base3pm[[1]]))
prop.table(table(base4a[[1]]))
prop.table(table(base3a[[1]]))

table(base4pb[[1]])
```

```
table(base3pb[[1]])
table(base4pm[[1]])
table(base3pm[[1]])
table(base4a[[1]])
table(base3a[[1]])

#-------------------Analisis con el modelo de 2PL------------------------------------#

mod5   <- list()
mod4pb <- list()
mod3pb <- list()
mod4pm <- list()
mod3pm <- list()
mod4a  <- list()
mod3a  <- list()

coef.mod5   <- list()
coef.mod4pb <- list()
coef.mod3pb <- list()
coef.mod4pm <- list()
coef.mod3pm <- list()
coef.mod4a  <- list()
coef.mod3a  <- list()

dif.5   <- list()
dif.4pb <- list()
dif.3pb <- list()
dif.4pm <- list()
dif.3pm <- list()
dif.4a  <- list()
dif.3a  <- list()

discri.5   <- list()
discri.4pb <- list()
discri.3pb <- list()
discri.4pm <- list()
discri.3pm <- list()
discri.4a  <- list()
discri.3a  <- list()

AIC.5   <- list()
AIC.4pb <- list()
AIC.3pb <- list()
AIC.4pm <- list()
AIC.3pm <- list()
AIC.4a  <- list()
AIC.3a  <- list()

t <- proc.time()
for (i in 1:replica){

  #generando los modelos
  mod5[[i]]  =ltm(base5[[i]]  ~z1)#nivel de habilidad unidimensional
  mod4pb[[i]]=ltm(base4pb[[i]]~z1)#nivel de habilidad unidimensional
  mod3pb[[i]]=ltm(base3pb[[i]]~z1)#nivel de habilidad unidimensional
  mod4pm[[i]]=ltm(base4pm[[i]]~z1)#nivel de habilidad unidimensional
  mod3pm[[i]]=ltm(base3pm[[i]]~z1)#nivel de habilidad unidimensional
  mod4a[[i]] =ltm(base4a[[i]] ~z1)#nivel de habilidad unidimensional
  mod3a[[i]] =ltm(base3a[[i]] ~z1)#nivel de habilidad unidimensional
```

```
  #Coeficientes
  coef.mod5[[i]] <- coef(mod5[[i]])
  coef.mod4pb[[i]] <- coef(mod4pb[[i]])
  coef.mod3pb[[i]] <- coef(mod3pb[[i]])
  coef.mod4pm[[i]]  <- coef(mod4pm[[i]])
  coef.mod3pm[[i]] <- coef(mod3pm[[i]])
  coef.mod4a[[i]]  <- coef(mod4a[[i]])
  coef.mod3a[[i]]  <- coef(mod3a[[i]])

  #Coeficientes de dificultad
  dif.5[[i]]    <- coef.mod5[[i]][,1]
  dif.4pb[[i]] <- coef.mod4pb[[i]][,1]
  dif.3pb[[i]] <- coef.mod3pb[[i]][,1]
  dif.4pm[[i]] <- coef.mod4pm[[i]][,1]
  dif.3pm[[i]] <- coef.mod3pm[[i]][,1]
  dif.4a[[i]]  <- coef.mod4a[[i]][,1]
  dif.3a[[i]]  <- coef.mod3a[[i]][,1]

  #Coeficientes de discriminacion
  discri.5[[i]]   <- coef.mod5[[i]][,2]
  discri.4pb[[i]] <- coef.mod4pb[[i]][,2]
  discri.3pb[[i]] <- coef.mod3pb[[i]][,2]
  discri.4pm[[i]] <- coef.mod4pm[[i]][,2]
  discri.3pm[[i]] <- coef.mod3pm[[i]][,2]
  discri.4a[[i]]  <- coef.mod4a[[i]][,2]
  discri.3a[[i]]  <- coef.mod3a[[i]][,2]

  #Generando los AIC
  AIC.5[[i]]   <-AIC(mod5[[i]])
  AIC.4pb[[i]] <- AIC(mod4pb[[i]])
  AIC.4pm[[i]] <- AIC(mod4pm[[i]])
  AIC.4a [[i]] <- AIC(mod4a[[i]])
  AIC.3pb[[i]] <- AIC(mod3pb[[i]])
  AIC.3pm[[i]] <- AIC(mod3pm[[i]])
  AIC.3a [[i]] <- AIC(mod3a[[i]])
  }
proc.time() - t


#--------------------------------Guardando bases -------------------------------------#
save(coef.mod5   ,file="coef.mod5.rda")
save(coef.mod4pb ,file="coef.mod4pb.rda")
save(coef.mod3pb ,file="coef.mod3pb.rda")
save(coef.mod4pm ,file="coef.mod4pm.rda")
save(coef.mod3pm ,file="coef.mod3pm.rda")
save(coef.mod4a  ,file="coef.mod4a.rda")
save(coef.mod3a  ,file="coef.mod3a.rda")

load("coef.mod5.rda")
load("coef.mod4pb.rda")
load("coef.mod3pb.rda")
load("coef.mod4pm.rda")
load("coef.mod3pm.rda")
load("coef.mod4a.rda")
load("coef.mod3a.rda")

save(dif.5   ,file="dif.5.rda")
save(dif.4pb ,file="dif.4pb.rda")
save(dif.3pb ,file="dif.3pb.rda")
save(dif.4pm ,file="dif.4pm.rda")
```

```
save(dif.3pm ,file="dif.3pm.rda")
save(dif.4a  ,file="dif.4a.rda")
save(dif.3a  ,file="dif.3a.rda")

load("dif.5.rda")
load("dif.4pb.rda")
load("dif.3pb.rda")
load("dif.4pm.rda")
load("dif.3pm.rda")
load("dif.4a.rda")
load("dif.3a.rda")

save(discri.5  ,file= "discri.5.rda")
save(discri.4pb,file= "discri.4pb.rda")
save(discri.3pb,file= "discri.3pb.rda")
save(discri.4pm,file= "discri.4pm.rda")
save(discri.3pm,file= "discri.3pm.rda")
save(discri.4a ,file= "discri.4a.rda")
save(discri.3a ,file= "discri.3a.rda")

load("discri.5.rda")
load("discri.4pb.rda")
load("discri.3pb.rda")
load("discri.4pm.rda")
load("discri.3pm.rda")
load("discri.4a.rda")
load("discri.3a.rda")

save(AIC.5  ,file="AIC.5.rda")
save(AIC.4pb,file="AIC.4pb.rda")
save(AIC.3pb,file="AIC.3pb.rda")
save(AIC.4pm,file="AIC.4pm.rda")
save(AIC.3pm,file="AIC.3pm.rda")
save(AIC.4a ,file="AIC.4a.rda")
save(AIC.3a ,file="AIC.3a.rda")

load("AIC.5.rda")
load("AIC.4pb.rda")
load("AIC.3pb.rda")
load("AIC.4pm.rda")
load("AIC.3pm.rda")
load("AIC.4a.rda")
load("AIC.3a.rda")

#--------------------- convirtiendo en DF las listas de AIC ----------------------------------

base.aic.5<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.5, SIMPLIFY = FALSE)

base.aic.4pb<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.4pb, SIMPLIFY = FALSE)

base.aic.4pm<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.4pm, SIMPLIFY = FALSE)

base.aic.4a<-mapply(function(data.aic){
  as.data.frame(data.aic)},
```

```
  data.aic=AIC.4a, SIMPLIFY = FALSE)

base.aic.3pb<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.3pb, SIMPLIFY = FALSE)

base.aic.3pm<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.3pm, SIMPLIFY = FALSE)

base.aic.3a<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.3a, SIMPLIFY = FALSE)

aic.5  <-do.call(rbind.data.frame,base.aic.5)
aic.4pb<-do.call(rbind.data.frame,base.aic.4pb)
aic.4pm<-do.call(rbind.data.frame,base.aic.4pm)
aic.4a <-do.call(rbind.data.frame,base.aic.4a )
aic.3pb<-do.call(rbind.data.frame,base.aic.3pb)
aic.3pm<-do.call(rbind.data.frame,base.aic.3pm)
aic.3a <-do.call(rbind.data.frame,base.aic.3a )

mean(aic.5  $data.aic)
mean(aic.4pb$data.aic)
mean(aic.4pm$data.aic)
mean(aic.4a $data.aic)
mean(aic.3pb$data.aic)
mean(aic.3pm$data.aic)
mean(aic.3a $data.aic)

#--------------convirtiendo en DF las listas de dificultad----------------------------#

base.coef.dif.5<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=dif.5, SIMPLIFY = FALSE)

base.coef.dif.4pb<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=dif.4pb, SIMPLIFY = FALSE)

base.coef.dif.4pm<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=dif.4pm, SIMPLIFY = FALSE)

base.coef.dif.4a<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=dif.4a, SIMPLIFY = FALSE)

base.coef.dif.3pb<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=dif.3pb, SIMPLIFY = FALSE)

base.coef.dif.3pm<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=dif.3pm, SIMPLIFY = FALSE)

base.coef.dif.3a<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=dif.3a, SIMPLIFY = FALSE)
```

```
# uniendo en DF los coeficientes de dificultad
c.dif.5  <-data.frame(t(do.call(cbind.data.frame,base.coef.dif.5)))
c.dif.4pb<-data.frame(t(do.call(cbind.data.frame,base.coef.dif.4pb)))
c.dif.4pm<-data.frame(t(do.call(cbind.data.frame,base.coef.dif.4pm)))
c.dif.4a <-data.frame(t(do.call(cbind.data.frame,base.coef.dif.4a )))
c.dif.3pb<-data.frame(t(do.call(cbind.data.frame,base.coef.dif.3pb)))
c.dif.3pm<-data.frame(t(do.call(cbind.data.frame,base.coef.dif.3pm)))
c.dif.3a <-data.frame(t(do.call(cbind.data.frame,base.coef.dif.3a )))

rownames(c.dif.5  ) <- c(1:replica)
rownames(c.dif.4pb) <- c(1:replica)
rownames(c.dif.4pm) <- c(1:replica)
rownames(c.dif.4a ) <- c(1:replica)
rownames(c.dif.3pb) <- c(1:replica)
rownames(c.dif.3pm) <- c(1:replica)
rownames(c.dif.3a ) <- c(1:replica)

colnames(c.dif.5  ) <- c(1:items)
colnames(c.dif.4pb) <- c(1:items)
colnames(c.dif.4pm) <- c(1:items)
colnames(c.dif.4a ) <- c(1:items)
colnames(c.dif.3pb) <- c(1:items)
colnames(c.dif.3pm) <- c(1:items)
colnames(c.dif.3a ) <- c(1:items)

save(c.dif.5   ,file="c.dif.5.rda")
save(c.dif.4pb ,file="c.dif.4pb.rda")
save(c.dif.3pb ,file="c.dif.3pb.rda")
save(c.dif.4pm ,file="c.dif.4pm.rda")
save(c.dif.3pm ,file="c.dif.3pm.rda")
save(c.dif.4a  ,file="c.dif.4a.rda")
save(c.dif.3a  ,file="c.dif.3a.rda")

load("c.dif.5.rda")
load("c.dif.4pb.rda")
load("c.dif.3pb.rda")
load("c.dif.4pm.rda")
load("c.dif.3pm.rda")
load("c.dif.4a.rda")
load("c.dif.3a.rda")
#---------------------------------------

prom.dif.5  <- data.frame(NA,1,items)
prom.dif.4pb<- data.frame(NA,1,items)
prom.dif.4pm<- data.frame(NA,1,items)
prom.dif.4a <- data.frame(NA,1,items)
prom.dif.3pb<- data.frame(NA,1,items)
prom.dif.3pm<- data.frame(NA,1,items)
prom.dif.3a <- data.frame(NA,1,items)

for (d in 1:items){
  prom.dif.5  [,d] <- mean(subset(c.dif.5  [,d],c.dif.5  [,d]>-6 & c.dif.5  [,d]<6))
  prom.dif.4pb[,d] <- mean(subset(c.dif.4pb[,d],c.dif.4pb[,d]>-6 & c.dif.4pb[,d]<6))
  prom.dif.4pm[,d] <- mean(subset(c.dif.4pm[,d],c.dif.4pm[,d]>-6 & c.dif.4pm[,d]<6))
  prom.dif.4a [,d] <- mean(subset(c.dif.4a [,d],c.dif.4a [,d]>-6 & c.dif.4a [,d]<6))
  prom.dif.3pb[,d] <- mean(subset(c.dif.3pb[,d],c.dif.3pb[,d]>-6 & c.dif.3pb[,d]<6))
  prom.dif.3pm[,d] <- mean(subset(c.dif.3pm[,d],c.dif.3pm[,d]>-6 & c.dif.3pm[,d]<6))
  prom.dif.3a [,d] <- mean(subset(c.dif.3a [,d],c.dif.3a [,d]>-6 & c.dif.3a [,d]<6))
}
```

```
colnames(prom.dif.5  ) <- c(1:items)
colnames(prom.dif.4pb) <- c(1:items)
colnames(prom.dif.4pm) <- c(1:items)
colnames(prom.dif.4a ) <- c(1:items)
colnames(prom.dif.3pb) <- c(1:items)
colnames(prom.dif.3pm) <- c(1:items)
colnames(prom.dif.3a ) <- c(1:items)

prom.dif                                                                                  <-
rbind(prom.dif.5,prom.dif.4pb,prom.dif.4pm,prom.dif.4a,prom.dif.3pb,prom.dif.3pm,prom.dif.3a)
rownames(prom.dif) <- c("5 opciones","4 proba baja", "4 proba alta", "4 azar","3 proba baja", "3
proba alta", "3 azar")

library(openxlsx)

write.xlsx(prom.dif, "prom.dif.xlsx",colNames = TRUE, borders = "columns" )
write.xlsx(c.dif.5  ,"c.dif.5.xlsx"  ,colNames = TRUE, borders = "columns")
write.xlsx(c.dif.4pb,"c.dif.4pb.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.dif.3pb,"c.dif.3pb.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.dif.4pm,"c.dif.4pm.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.dif.3pm,"c.dif.3pm.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.dif.4a ,"c.dif.4a.xlsx" ,colNames = TRUE, borders = "columns")
write.xlsx(c.dif.3a ,"c.dif.3a.xlsx" ,colNames = TRUE, borders = "columns")


#-------------------------quitando los casos con coeficientes <-6 y >6 -----------------#
#------------------- convirtiendo en DF las listas de discriminacion -----------------------#

base.coef.discri.5<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=discri.5, SIMPLIFY = FALSE)

base.coef.discri.4pb<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=discri.4pb, SIMPLIFY = FALSE)

base.coef.discri.4pm<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=discri.4pm, SIMPLIFY = FALSE)

base.coef.discri.4a<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=discri.4a, SIMPLIFY = FALSE)

base.coef.discri.3pb<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=discri.3pb, SIMPLIFY = FALSE)

base.coef.discri.3pm<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=discri.3pm, SIMPLIFY = FALSE)

base.coef.discri.3a<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=discri.3a, SIMPLIFY = FALSE)

# uniendo en DF los coeficientes de discriminacion

c.discri.5  <-data.frame(t(do.call(cbind.data.frame,base.coef.discri.5)))
c.discri.4pb<-data.frame(t(do.call(cbind.data.frame,base.coef.discri.4pb)))
```

```
c.discri.4pm<-data.frame(t(do.call(cbind.data.frame,base.coef.discri.4pm)))
c.discri.4a <-data.frame(t(do.call(cbind.data.frame,base.coef.discri.4a )))
c.discri.3pb<-data.frame(t(do.call(cbind.data.frame,base.coef.discri.3pb)))
c.discri.3pm<-data.frame(t(do.call(cbind.data.frame,base.coef.discri.3pm)))
c.discri.3a <-data.frame(t(do.call(cbind.data.frame,base.coef.discri.3a )))

rownames(c.discri.5  ) <- c(1:replica)
rownames(c.discri.4pb) <- c(1:replica)
rownames(c.discri.4pm) <- c(1:replica)
rownames(c.discri.4a ) <- c(1:replica)
rownames(c.discri.3pb) <- c(1:replica)
rownames(c.discri.3pm) <- c(1:replica)
rownames(c.discri.3a ) <- c(1:replica)

colnames(c.discri.5  ) <- c(1:items)
colnames(c.discri.4pb) <- c(1:items)
colnames(c.discri.4pm) <- c(1:items)
colnames(c.discri.4a ) <- c(1:items)
colnames(c.discri.3pb) <- c(1:items)
colnames(c.discri.3pm) <- c(1:items)
colnames(c.discri.3a ) <- c(1:items)

save(c.discri.5   ,file="c.discri.5.rda")
save(c.discri.4pb ,file="c.discri.4pb.rda")
save(c.discri.3pb ,file="c.discri.3pb.rda")
save(c.discri.4pm ,file="c.discri.4pm.rda")
save(c.discri.3pm ,file="c.discri.3pm.rda")
save(c.discri.4a  ,file="c.discri.4a.rda")
save(c.discri.3a  ,file="c.discri.3a.rda")

load("c.discri.5.rda")
load("c.discri.4pb.rda")
load("c.discri.3pb.rda")
load("c.discri.4pm.rda")
load("c.discri.3pm.rda")
load("c.discri.4a.rda")
load("c.discri.3a.rda")
#-----------------------------------------------------#

prom.discri.5  <- data.frame(NA,1,items)
prom.discri.4pb<- data.frame(NA,1,items)
prom.discri.4pm<- data.frame(NA,1,items)
prom.discri.4a <- data.frame(NA,1,items)
prom.discri.3pb<- data.frame(NA,1,items)
prom.discri.3pm<- data.frame(NA,1,items)
prom.discri.3a <- data.frame(NA,1,items)

for (d in 1:items){
  prom.discri.5  [,d] <- mean(subset(c.discri.5  [,d],c.discri.5  [,d]>-6 & c.discri.5  [,d]<6))
  prom.discri.4pb[,d] <- mean(subset(c.discri.4pb[,d],c.discri.4pb[,d]>-6 & c.discri.4pb[,d]<6))
  prom.discri.4pm[,d] <- mean(subset(c.discri.4pm[,d],c.discri.4pm[,d]>-6 & c.discri.4pm[,d]<6))
  prom.discri.4a [,d] <- mean(subset(c.discri.4a [,d],c.discri.4a [,d]>-6 & c.discri.4a [,d]<6))
  prom.discri.3pb[,d] <- mean(subset(c.discri.3pb[,d],c.discri.3pb[,d]>-6 & c.discri.3pb[,d]<6))
  prom.discri.3pm[,d] <- mean(subset(c.discri.3pm[,d],c.discri.3pm[,d]>-6 & c.discri.3pm[,d]<6))
  prom.discri.3a [,d] <- mean(subset(c.discri.3a [,d],c.discri.3a [,d]>-6 & c.discri.3a [,d]<6))
}

colnames(prom.discri.5  ) <- c(1:items)
colnames(prom.discri.4pb) <- c(1:items)
colnames(prom.discri.4pm) <- c(1:items)
```

```
colnames(prom.discri.4a ) <- c(1:items)
colnames(prom.discri.3pb) <- c(1:items)
colnames(prom.discri.3pm) <- c(1:items)
colnames(prom.discri.3a ) <- c(1:items)


prom.discri                                                                                          <-
rbind(prom.discri.5,prom.discri.4pb,prom.discri.4pm,prom.discri.4a,prom.discri.3pb,prom.discri.3p
m,prom.discri.3a)
rownames(prom.discri) <- c("5 opciones","4 proba baja", "4 proba alta", "4 azar","3 proba baja",
"3 proba alta", "3 azar")

write.xlsx(prom.discri, "prom.discri.xlsx",colNames = TRUE, borders = "columns" )

#------------------recm de dificultad y discriminacion por replica--------------------------#

#----------------------------optimos dif y discri-----------------------------------------------#
dif.optimo5   <- list()
dif.optimo4pb <- list()
dif.optimo3pb <- list()
dif.optimo4pm <- list()
dif.optimo3pm <- list()
dif.optimo4a  <- list()
dif.optimo3a  <- list()

discri.optimo5   <- list()
discri.optimo4pb <- list()
discri.optimo3pb <- list()
discri.optimo4pm <- list()
discri.optimo3pm <- list()
discri.optimo4a  <- list()
discri.optimo3a  <- list()

for (d in 1:items){
  dif.optimo5[[d]] <- which(c.dif.5[,d]> -6 & c.dif.5[,d]<6)
  dif.optimo4pb[[d]] <- which(c.dif.4pb[,d]> -6 & c.dif.4pb[,d]<6)
  dif.optimo3pb[[d]] <- which(c.dif.3pb[,d]> -6 & c.dif.3pb[,d]<6)
  dif.optimo4pm[[d]] <- which(c.dif.4pm[,d]> -6 & c.dif.4pm[,d]<6)
  dif.optimo3pm[[d]] <- which(c.dif.3pm[,d]> -6 & c.dif.3pm[,d]<6)
  dif.optimo4a [[d]] <- which(c.dif.4a[,d]> -6 & c.dif.4a[,d]<6)
  dif.optimo3a [[d]] <- which(c.dif.3a[,d]> -6 & c.dif.3a[,d]<6)

  discri.optimo5[[d]] <- which(c.discri.5[,d]> -6 & c.discri.5[,d]<6)
  discri.optimo4pb[[d]] <- which(c.discri.4pb[,d]> -6 & c.discri.4pb[,d]<6)
  discri.optimo3pb[[d]] <- which(c.discri.3pb[,d]> -6 & c.discri.3pb[,d]<6)
  discri.optimo4pm[[d]] <- which(c.discri.4pm[,d]> -6 & c.discri.4pm[,d]<6)
  discri.optimo3pm[[d]] <- which(c.discri.3pm[,d]> -6 & c.discri.3pm[,d]<6)
  discri.optimo4a [[d]] <- which(c.discri.4a[,d]> -6 & c.discri.4a[,d]<6)
  discri.optimo3a [[d]] <- which(c.discri.3a[,d]> -6 & c.discri.3a[,d]<6)
}

dif.optimo5<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo5, SIMPLIFY = FALSE)

dif.optimo4pb<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo4pb, SIMPLIFY = FALSE)

dif.optimo3pb<-mapply(function(data){
```

```
  as.data.frame(data)},
  data=dif.optimo3pb, SIMPLIFY = FALSE)

dif.optimo4pm<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo4pm, SIMPLIFY = FALSE)

dif.optimo3pm<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo3pm, SIMPLIFY = FALSE)

dif.optimo4a<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo4a, SIMPLIFY = FALSE)

dif.optimo3a<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo3a, SIMPLIFY = FALSE)

discri.optimo5<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo5, SIMPLIFY = FALSE)

discri.optimo4pb<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo4pb, SIMPLIFY = FALSE)

discri.optimo3pb<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo3pb, SIMPLIFY = FALSE)

discri.optimo4pm<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo4pm, SIMPLIFY = FALSE)

discri.optimo3pm<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo3pm, SIMPLIFY = FALSE)

discri.optimo4a<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo4a, SIMPLIFY = FALSE)

discri.optimo3a<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo3a, SIMPLIFY = FALSE)

dif.optimo.pb <- list()
dif.optimo.pm <- list()
dif.optimo.a  <- list()
discri.optimo.pb <- list()
discri.optimo.pm <- list()
discri.optimo.a  <- list()

for (d in 1: items){
  dif.optimo.pb[[d]]     <-     Reduce(merge,     list(dif.optimo5[[d]],     dif.optimo4pb[[d]],
dif.optimo3pb[[d]])) # Arroja un DF con datos en comunes de todos ellos
  dif.optimo.pm[[d]]     <-     Reduce(merge,     list(dif.optimo5[[d]],     dif.optimo4pm[[d]],
dif.optimo3pm[[d]])) # Arroja un DF con datos en comunes de todos ellos
```

```
  dif.optimo.a    [[d]]    <-    Reduce(merge,    list(dif.optimo5[[d]],    dif.optimo4a[[d]], dif.optimo3a[[d]]))  # Arroja un DF con datos en comunes de todos ellos

  discri.optimo.pb[[d]]   <-   Reduce(merge,   list(discri.optimo5[[d]],   discri.optimo4pb[[d]], discri.optimo3pb[[d]])) # Arroja un DF con datos en comunes de todos ellos
  discri.optimo.pm[[d]]   <-   Reduce(merge,   list(discri.optimo5[[d]],   discri.optimo4pm[[d]], discri.optimo3pm[[d]])) # Arroja un DF con datos en comunes de todos ellos
  discri.optimo.a   [[d]]   <-   Reduce(merge,   list(discri.optimo5[[d]],   discri.optimo4a[[d]], discri.optimo3a[[d]]))  # Arroja un DF con datos en comunes de todos ellos
  }

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

#rm(optimo.pb)
#rm(optimo.pm)
#rm(optimo.a)
#rm(optimo5  )
#rm(optimo4pb)
#rm(optimo3pb)
#rm(optimo4pm)
#rm(optimo3pm)
#rm(optimo4a )
#rm(optimo3a )

recm.dif.4pb   <-list()
recm.dif.4pm   <-list()
recm.dif.4a    <-list()
recm.dif.3pb   <-list()
recm.dif.3pm   <-list()
recm.dif.3a    <-list()

recm.discri.4pb<-list()
recm.discri.4pm<-list()
recm.discri.4a <-list()
recm.discri.3pb<-list()
recm.discri.3pm<-list()
recm.discri.3a <-list()


t <- proc.time()
for (i in 1:items){
  #RECM de dificultad
  recm.dif.4pb[[i]]            <-             sqrt(sum((c.dif.4pb[unlist(dif.optimo.pb[[i]]),i]-c.dif.5[unlist(dif.optimo.pb[[i]]),i])^2)/length(unlist(dif.optimo.pb[[i]])))
  recm.dif.4pm[[i]]            <-             sqrt(sum((c.dif.4pm[unlist(dif.optimo.pm[[i]]),i]-c.dif.5[unlist(dif.optimo.pm[[i]]),i])^2)/length(unlist(dif.optimo.pm[[i]])))
  recm.dif.4a[[i]]                  <-          sqrt(sum((c.dif.4a[unlist(dif.optimo.a[[i]]),i]-c.dif.5[unlist(dif.optimo.a[[i]]),i])^2)/length(unlist(dif.optimo.a[[i]])))
  recm.dif.3pb[[i]]            <-             sqrt(sum((c.dif.3pb[unlist(dif.optimo.pb[[i]]),i]-c.dif.5[unlist(dif.optimo.pb[[i]]),i])^2)/length(unlist(dif.optimo.pb[[i]])))
  recm.dif.3pm[[i]]            <-             sqrt(sum((c.dif.3pm[unlist(dif.optimo.pm[[i]]),i]-c.dif.5[unlist(dif.optimo.pm[[i]]),i])^2)/length(unlist(dif.optimo.pm[[i]])))
  recm.dif.3a[[i]]                  <-          sqrt(sum((c.dif.3a[unlist(dif.optimo.a[[i]]),i]-c.dif.5[unlist(dif.optimo.a[[i]]),i])^2)/length(unlist(dif.optimo.a[[i]])))

  #RECM de discriminacion
  recm.discri.4pb[[i]]        <-        sqrt(sum((c.discri.4pb[unlist(discri.optimo.pb[[i]]),i]-c.discri.5[unlist(discri.optimo.pb[[i]]),i])^2)/length(unlist(discri.optimo.pb[[i]])))
  recm.discri.4pm[[i]]        <-        sqrt(sum((c.discri.4pm[unlist(discri.optimo.pm[[i]]),i]-c.discri.5[unlist(discri.optimo.pm[[i]]),i])^2)/length(unlist(discri.optimo.pm[[i]])))
```

```
  recm.discri.4a[[i]]          <-     sqrt(sum((c.discri.4a     [unlist(discri.optimo.a[[i]]),i]-
c.discri.5[unlist(discri.optimo.a[[i]]),i])^2)/length(unlist(discri.optimo.a[[i]])))
  recm.discri.3pb[[i]]         <-         sqrt(sum((c.discri.3pb[unlist(discri.optimo.pb[[i]]),i]-
c.discri.5[unlist(discri.optimo.pb[[i]]),i])^2)/length(unlist(discri.optimo.pb[[i]])))
  recm.discri.3pm[[i]]         <-         sqrt(sum((c.discri.3pm[unlist(discri.optimo.pm[[i]]),i]-
c.discri.5[unlist(discri.optimo.pm[[i]]),i])^2)/length(unlist(discri.optimo.pm[[i]])))
  recm.discri.3a[[i]]          <-     sqrt(sum((c.discri.3a     [unlist(discri.optimo.a[[i]]),i]-
c.discri.5[unlist(discri.optimo.a[[i]]),i])^2)/length(unlist(discri.optimo.a[[i]])))

}
proc.time() - t


#rm(recm.dif.4pb)
#rm(recm.dif.3pb)
#rm(recm.dif.4pm)
#rm(recm.dif.3pm)
#rm(recm.dif.4a)
#rm(recm.dif.3a)
#
#rm(recm.discri.4pb)
#rm(recm.discri.3pb)
#rm(recm.discri.4pm)
#rm(recm.discri.3pm)
#rm(recm.discri.4a)
#rm(recm.discri.3a)

#
base.recm.dif.4pb<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.4pb, SIMPLIFY = FALSE)

base.recm.dif.4pm<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.4pm, SIMPLIFY = FALSE)

base.recm.dif.4a<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.4a, SIMPLIFY = FALSE)

base.recm.dif.3pb<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.3pb, SIMPLIFY = FALSE)

base.recm.dif.3pm<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.3pm, SIMPLIFY = FALSE)

base.recm.dif.3a<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.3a, SIMPLIFY = FALSE)

# uniendo en DF los RECM dificultad

recm.dif.4pb<-do.call(rbind.data.frame,base.recm.dif.4pb)
recm.dif.4pm<-do.call(rbind.data.frame,base.recm.dif.4pm)
recm.dif.4a <-do.call(rbind.data.frame,base.recm.dif.4a )
recm.dif.3pb<-do.call(rbind.data.frame,base.recm.dif.3pb)
recm.dif.3pm<-do.call(rbind.data.frame,base.recm.dif.3pm)
recm.dif.3a <-do.call(rbind.data.frame,base.recm.dif.3a )
```

```
recm.dif                                                                                    <-
rbind(t(recm.dif.4pb),t(recm.dif.3pb),t(recm.dif.4pm),t(recm.dif.3pm),t(recm.dif.4a),t(recm.dif.3
a))
rownames(recm.dif) <- c("4 proba baja","3 proba baja","4 proba alta","3 proba alta","4 azar","3
azar")
colnames(recm.dif) <- 1:items

write.xlsx(recm.dif, "recm.dif.xlsx",colNames = TRUE, borders = "columns" )

save(recm.dif,file="recm.dif.rda")
load("recm.dif.rda")

mean.recm.dif <- apply(recm.dif, 1, mean, na.rm = TRUE)
save(mean.recm.dif,file="mean.recm.dif.rda")
load("mean.recm.dif.rda")
write.xlsx(mean.recm.dif, "mean.recm.dif.xlsx",colNames = TRUE, borders = "columns" )


#--------------------recm de discriminacion--------------------------------------------#
base.recm.discri.4pb<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.4pb, SIMPLIFY = FALSE)

base.recm.discri.4pm<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.4pm, SIMPLIFY = FALSE)

base.recm.discri.4a<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.4a, SIMPLIFY = FALSE)

base.recm.discri.3pb<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.3pb, SIMPLIFY = FALSE)

base.recm.discri.3pm<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.3pm, SIMPLIFY = FALSE)

base.recm.discri.3a<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.3a, SIMPLIFY = FALSE)

recm.discri.4pb<-do.call(rbind.data.frame,base.recm.discri.4pb)
recm.discri.4pm<-do.call(rbind.data.frame,base.recm.discri.4pm)
recm.discri.4a <-do.call(rbind.data.frame,base.recm.discri.4a )
recm.discri.3pb<-do.call(rbind.data.frame,base.recm.discri.3pb)
recm.discri.3pm<-do.call(rbind.data.frame,base.recm.discri.3pm)
recm.discri.3a <-do.call(rbind.data.frame,base.recm.discri.3a )

mean(na.omit(recm.discri.4pb$data.discri))
mean(na.omit(recm.discri.4pm$data.discri))
mean(na.omit(recm.discri.4a $data.discri))
mean(na.omit(recm.discri.3pb$data.discri))
mean(na.omit(recm.discri.3pm$data.discri))
mean(na.omit(recm.discri.3a $data.discri))
```

```
recm.discri                                                                                    <-
rbind(t(recm.discri.4pb),t(recm.discri.3pb),t(recm.discri.4pm),t(recm.discri.3pm),t(recm.discri.4
a),t(recm.discri.3a))
rownames(recm.discri) <- c("4 proba baja","3 proba baja","4 proba alta","3 proba alta","4 azar","3
azar")
colnames(recm.discri) <- 1:items

write.xlsx(recm.discri, "recm.discri.xlsx",colNames = TRUE, borders = "columns" )

save(recm.discri,file="recm.discri.rda")
load("recm.discri.rda")

mean.recm.discri <- apply(recm.discri, 1, mean, na.rm = TRUE)
save(mean.recm.discri,file="mean.recm.discri.rda")
load("mean.recm.discri.rda")
write.xlsx(mean.recm.discri, "mean.recm.discri.xlsx",colNames = TRUE, borders = "columns" )

#---------------------------------OPCION 2-------------------------------------------------------#

#---------------------- Quitando los items malos con el AF -------------------------#
# --------------------- Excluyendo items ------------------------------------------#

#------------------------- leer bases --------------------------------------------------#
#-------- 500 -----------
set.seed(8003)
load("muestra500.rda")
load("opcion.muestra.rda")
n<-500
opciones<-5
replica <- 100
items<-35

#------------ 750
load("muestra750.rda")
load("opcion.muestra.rda")
n<-750
opciones<-5
replica <- 100
items<-35

#--------------- 1000
load("muestra1000.rda")
load("opcion.muestra.rda")
n<-1000
opciones<-5
replica <- 100
items<-35

#---------------- Análisis de factores ----------------------------------------------------#

#--------------------------------------- primera ronda ---------------------------------------#

af <- list()
cargas <- list()
#no.excluir <- list()
excluir <- list()
for (y in 1:replica){
  af[[y]] <- factanal(muestra[[y]],1)
 cargas[[y]] <- af[[y]]$loadings[,1]
 cargas[[y]] <- data.frame(cargas[[y]])
```

```
 cargas[[y]] <- mutate(cargas[[y]],id=1:items)
 colnames(cargas[[y]]) <- c("carga", "id")
 excluir[[y]]<- subset(cargas[[y]],carga > -0.1 & carga < 0.1)$id
 #excluir[[y]] <- cargas[[y]][-no.excluir[[y]],2]
}

#confirmar
af[[6]]
cargas[[6]]
excluir[[6]]

#a <- subset(cargas[[6]],carga > -0.1 & carga < 0.1)$id
#b <- cargas[[6]][-a,2]
#
#which(cargas[[6]][,1] > -0.1 & cargas[[6]][,1] < 0.1)
#subset(cargas[[6]],carga > -0.1 & carga < 0.1)$id


#------------------ quietar los items malos (muestra, opcion.muestra, items)---------------------
#

base.e <- muestra
opcion.e <- opcion.muestra
for (c in 1:replica){
    if(length(excluir[[c]])!=  0){base.e[[c]]  <-base.e[[c]][,-c(excluir[[c]])];opcion.e[[c]]  <-
opcion.e[[c]][-c(excluir[[c]]),]}else
    {base.e[[c]] <-base.e[[c]];opcion.e[[c]] <-opcion.e[[c]]}
}

length(colnames(base.e[[6]]))
nrow(opcion.e[[6]])
#----------------------------------------- Segunda ronda ----------------------------#

#aplicar nuevamente AF para ver si sigue dando problemas
af2 <- list()
cargas2 <- list()
excluir2 <- list()
for (y in 1:replica){
  af2[[y]] <- factanal(base.e[[y]],1)
  cargas2[[y]] <- af2[[y]]$loadings[,1]
  cargas2[[y]] <- data.frame(cargas2[[y]])
  cargas2[[y]] <- mutate(cargas2[[y]],id=opcion.e[[y]]$id)
  colnames(cargas2[[y]]) <- c("carga", "id")
  rownames(cargas2[[y]]) <- opcion.e[[y]]$id # ver si cambiarlo por cargas2[[p]]$id
  excluir2[[y]]<- subset(cargas2[[y]],carga > -0.1 & carga < 0.1)$id
}

af2[[6]]
excluir2[[6]]
cargas2[[6]]

#------------------ quitar los items malos (muestra, opcion.muestra, items)--------------------#
# cambiar nombre a las columnas de la base
base.e2 <- base.e
for ( p in 1:replica){
  colnames(base.e2[[p]]) <- cargas2[[p]]$id
}

opcion.e2 <- opcion.e
for (c in 1:replica){
```

```
  if(length(excluir2[[c]])!=      0){base.e2[[c]]      <-base.e2[[c]][,!(names(base.e2[[c]])%in%
excluir2[[c]])];opcion.e2[[c]]                                                                  <-
filter(opcion.e2[[c]][!(rownames(opcion.e2[[c]][2])%in%excluir2[[c]]),])}else
  {base.e2[[c]] <-base.e2[[c]];opcion.e2[[c]] <-opcion.e2[[c]]}
}

for (p in 1:replica){
rownames(opcion.e2[[p]]) <- opcion.e2[[p]]$id
}

#base.e2[[6]]
#colnames(base.e[[6]])
#opcion.e2[[6]]
#filter(opcion.e2[[6]][!(rownames(opcion.e2[[6]][2])%in%excluir2[[6]]),])
length(colnames(base.e2[[6]]))
nrow(opcion.e2[[6]])
#rownames(opcion.e[[6]])

#Ejemplos especificos
#opcion.e2[[2]][!()%in%excluir2[[2]]),]
#base.e2[[2]][,!(names(base.e2[[2]])%in% excluir2[[2]])]
#filter(opcion.e2[[2]][!(rownames(opcion.e2[[2]][2])%in%excluir2[[2]]),])

#---------------------------- Tercera ronda --------------------------------------------------#
#aplicar nuevamente AF para ver si sigue dando problemas
af3 <- list()
cargas3 <- list()
excluir3 <- list()
for (y in 1:replica){
  af3[[y]] <- factanal(base.e2[[y]],1)
  cargas3[[y]] <- af3[[y]]$loadings[,1]
  cargas3[[y]] <- data.frame(cargas3[[y]])
  cargas3[[y]] <- mutate(cargas3[[y]],id=opcion.e2[[y]]$id)
  colnames(cargas3[[y]]) <- c("carga", "id")
  rownames(cargas3[[y]]) <- opcion.e2[[y]]$id # ver si cambiarlo por cargas2[[p]]$id
  excluir3[[y]]<- subset(cargas3[[y]],carga > -0.1 & carga < 0.1)$id
}

af3[[6]]
excluir3[[6]]
#------------------ quitar los items malos (muestra, opcion.muestra, items)---------------------#

# cambiar nombre a las columnas de la base
base.e3 <- base.e2
for ( p in 1:replica){
  colnames(base.e3[[p]]) <- cargas3[[p]]$id
}

opcion.e3 <- opcion.e2
for (c in 1:replica){
  if(length(excluir3[[c]])!=      0){base.e3[[c]]      <-base.e3[[c]][,!(names(base.e3[[c]])%in%
excluir3[[c]])];opcion.e3[[c]]                                                                  <-
filter(opcion.e3[[c]][!(rownames(opcion.e3[[c]][2])%in%excluir3[[c]]),])}else
  {base.e3[[c]] <-base.e3[[c]];opcion.e3[[c]] <-opcion.e3[[c]]}
}


for (p in 1:replica){
  rownames(opcion.e3[[p]]) <- opcion.e3[[p]]$id
}
```

```
length(colnames(base.e3[[6]]))
nrow(opcion.e3[[6]])

#----------------------------------------- Cuarta ronda --------------------------------------#
#aplicar nuevamente AF para ver si sigue dando problemas
af4 <- list()
cargas4 <- list()
excluir4 <- list()
for (y in 1:replica){
  af4[[y]] <- factanal(base.e3[[y]],1)
  cargas4[[y]] <- af4[[y]]$loadings[,1]
  cargas4[[y]] <- data.frame(cargas4[[y]])
  cargas4[[y]] <- mutate(cargas4[[y]],id=opcion.e3[[y]]$id)
  colnames(cargas4[[y]]) <- c("carga", "id")
  rownames(cargas4[[y]]) <- opcion.e3[[y]]$id # ver si cambiarlo por cargas2[[p]]$id
  excluir4[[y]]<- subset(cargas4[[y]],carga > -0.1 & carga < 0.1)$id
}

#----------------- quitar los items malos (muestra, opcion.muestra, items)---------------------#

# cambiar nombre a las columnas de la base
base.e4 <- base.e3
for ( p in 1:replica){
  colnames(base.e4[[p]]) <- cargas4[[p]]$id
}

opcion.e4 <- opcion.e3
for (c in 1:replica){
  if(length(excluir4[[c]])!=      0){base.e4[[c]]      <-base.e4[[c]][,!(names(base.e4[[c]])%in%
excluir4[[c]])];opcion.e4[[c]]                                                                 <-
filter(opcion.e4[[c]][!(rownames(opcion.e4[[c]][2])%in%excluir4[[c]]),])}else
  {base.e4[[c]] <-base.e4[[c]];opcion.e4[[c]] <-opcion.e4[[c]]}
}

for (p in 1:replica){
  rownames(opcion.e4[[p]]) <- opcion.e4[[p]]$id
}

excluir4[[6]]
length(colnames(base.e4[[6]]))
nrow(opcion.e4[[6]])

#----------------------------------------- Quinta ronda -----------------------------------#
#aplicar nuevamente AF para ver si sigue dando problemas
af5 <- list()
cargas5 <- list()
excluir5 <- list()
for (y in 1:replica){
  af5[[y]] <- factanal(base.e4[[y]],1)
  cargas5[[y]] <- af5[[y]]$loadings[,1]
  cargas5[[y]] <- data.frame(cargas5[[y]])
  cargas5[[y]] <- mutate(cargas5[[y]],id=opcion.e4[[y]]$id)
  colnames(cargas5[[y]]) <- c("carga", "id")
  rownames(cargas5[[y]]) <- opcion.e4[[y]]$id # ver si cambiarlo por cargas2[[p]]$id
  excluir5[[y]]<- subset(cargas5[[y]],carga > -0.1 & carga < 0.1)$id
}

#----------------- quitar los items malos (muestra, opcion.muestra, items)---------------------#
```

```
# cambiar nombre a las columnas de la base
base.e5 <- base.e4
for ( p in 1:replica){
  colnames(base.e5[[p]]) <- cargas5[[p]]$id
}

opcion.e5 <- opcion.e4
for (c in 1:replica){
  if(length(excluir5[[c]])!=      0){base.e5[[c]]      <-base.e5[[c]][,!(names(base.e5[[c]])%in%
excluir5[[c]])];opcion.e5[[c]]                                                                  <-
filter(opcion.e5[[c]][!(rownames(opcion.e5[[c]][2])%in%excluir5[[c]]),])}else
  {base.e5[[c]] <-base.e5[[c]];opcion.e5[[c]] <-opcion.e5[[c]]}
}

for (p in 1:replica){
  rownames(opcion.e5[[p]]) <- opcion.e5[[p]]$id
}

excluir5[[6]]
length(colnames(base.e5[[6]]))
nrow(opcion.e5[[6]])
#----------------------------------------- Sexta ronda ---------------------------------------#
#aplicar nuevamente AF para ver si sigue dando problemas
af6 <- list()
cargas6 <- list()
excluir6 <- list()
for (y in 1:replica){
  af6[[y]] <- factanal(base.e5[[y]],1)
  cargas6[[y]] <- af6[[y]]$loadings[,1]
  cargas6[[y]] <- data.frame(cargas6[[y]])
  cargas6[[y]] <- mutate(cargas6[[y]],id=opcion.e5[[y]]$id)
  colnames(cargas6[[y]]) <- c("carga", "id")
  rownames(cargas6[[y]]) <- opcion.e5[[y]]$id # ver si cambiarlo por cargas2[[p]]$id
  excluir6[[y]]<- subset(cargas6[[y]],carga > -0.1 & carga < 0.1)$id
}

#------------------ quitar los items malos (muestra, opcion.muestra, items)---------------------#

# cambiar nombre a las columnas de la base
base.e6 <- base.e5
for ( p in 1:replica){
  colnames(base.e6[[p]]) <- cargas6[[p]]$id
}

opcion.e6 <- opcion.e5
for (c in 1:replica){
  if(length(excluir6[[c]])!=      0){base.e6[[c]]      <-base.e6[[c]][,!(names(base.e6[[c]])%in%
excluir6[[c]])];opcion.e6[[c]]                                                                  <-
filter(opcion.e6[[c]][!(rownames(opcion.e6[[c]][2])%in%excluir6[[c]]),])}else
  {base.e6[[c]] <-base.e6[[c]];opcion.e6[[c]] <-opcion.e6[[c]]}
}

for (p in 1:replica){
  rownames(opcion.e6[[p]]) <- opcion.e6[[p]]$id
}

excluir6[[6]]
length(colnames(base.e6[[6]]))
nrow(opcion.e6[[6]])
```

```
#------------------------------------------------------------------------------------------#
#corroborar
ncol(base.e3[[1]])
nrow(opcion.e3[[1]])

#--------------------------------------generando el modelo -------------------------------------#
mod <- list()
coef.mod <- list()
pend <- list()
inter <- list()

t <- proc.time()
for (v in 1:replica){
  #generando el modelo TRI
  mod[[v]]<-mirt(base.e2[[v]],1, itemtype=rep('nominal',ncol(base.e2[[v]])),draws=5000)

  #extrayendo los coeficientes
  coef.mod[[v]]<-   as.data.frame(coef(mod[[v]],   IRTpars=TRUE,   simplify=TRUE)$items,method   =
"EM",draws = 10000) #IRTpars, Convertir parámetros de intersección de pendiente en parámetros
tradicionales de IRT
  pend[[v]]<-coef.mod[[v]][1:5]
  inter[[v]]<-coef.mod[[v]][6:10]
  colnames(pend[[v]]) <- 1:5
}
proc.time() - t


####
opcion.e2[[1]][1,]
itemplot(mod[[1]],1,theta_lim = c(-3,3))

#####
pend2<-mapply(function(data.pend){
  apply(data.pend,1,function(x) names(sort(x)))},
  data.pend=pend, SIMPLIFY = FALSE)

pend3<-mapply(function(data.pend){
  apply(data.pend,1,function(x) names(sort(x,decreasing = TRUE)))},
  data.pend=pend, SIMPLIFY = FALSE)

pend3[[1]][,1]

pend.order2<-pend3
pend.order2<-pend2
pend.order2[[1]]

pend2[[1]]
pend3[[1]]
opcion.e6[[1]]
#---------------------- GENERANDO LAS BASE DE 4 Y 3 OPCIONES -------------------------------#

#----------------- Usando la siguiente opcion más baja -------------------------------#

#Pasando a 4 opciones
base4 <- base.e2
for (c in 1:replica){
  for (j in 1:ncol(base4[[c]])){
    for(r in 1:n){
      if                 (base4[[c]][r,j]==as.numeric(pend.order2[[c]][,j][1])){base4[[c]][r,j]<-
as.numeric(pend.order2[[c]][,j][2])}else
```

```
        if (base4[[c]][r,j]!=as.numeric(pend.order2[[c]][,j][1])){base4[[c]][r,j]}
    }
  }
}

pend.order2[[1]][,1]
table(base.e2[[1]][,1]);table(base4[[1]][,1])

#Pasando a 3 opciones
base3 <- base4
for (e in 1:replica){
  for (k in 1:ncol(base3[[e]])){
    for(m in 1:n){
      if                 (base3[[e]][m,k]==as.numeric(pend.order2[[e]][,k][2])){base3[[e]][m,k]<-
as.numeric(pend.order2[[e]][,k][3])}else
        if (base3[[e]][m,k]!=as.numeric(pend.order2[[e]][,k][2])){base3[[e]][m,k]}
    }
  }
}

pend.order2[[1]][,1]
table(base.e2[[1]][,1]);table(base4[[1]][,1]);table(base3[[1]][,1])
opcion.e2[[1]][1,]
#------------------------- Opcion 2 mayor proba -----------------------------------------#
#Pasando a 4 opciones
base4.2 <- base.e2
for (c in 1:replica){
  for (j in 1:ncol(base4.2[[c]])){
    for(r in 1:n){
      if             (base4.2[[c]][r,j]==as.numeric(pend.order2[[c]][,j][1])){base4.2[[c]][r,j]<-
as.numeric(pend.order2[[c]][,j][5])}else
        if (base4.2[[c]][r,j]!=as.numeric(pend.order2[[c]][,j][1])){base4.2[[c]][r,j]}
    }
  }
}

pend.order2[[1]][,1]
table(base.e2[[1]][,1]);table(base4.2[[1]][,1])

#Pasando a 3 opciones
base3.2 <- base4.2
for (e in 1:replica){
  for (k in 1:ncol(base3.2[[e]])){
    for(m in 1:n){
      if             (base3.2[[e]][m,k]==as.numeric(pend.order2[[e]][,k][2])){base3.2[[e]][m,k]<-
as.numeric(pend.order2[[e]][,k][5])}else
        if (base3.2[[e]][m,k]!=as.numeric(pend.order2[[e]][,k][2])){base3.2[[e]][m,k]}
    }
  }
}

pend.order2[[1]][,1]
table(base.e2[[1]][,1]);table(base4.2[[1]][,1]);table(base3.2[[1]][,1])
opcion.e2[[1]][1,]

#------------------------------ Opcion 3 aleatoreamente ------------------------#
#Pasando a 4 opciones
base4.3 <- base.e2
for (c in 1:replica){
  for (j in 1:ncol(base4.3[[c]])){
```

```
    for(r in 1:n){
      if              (base4.3[[c]][r,j]==as.numeric(pend.order2[[c]][,j][1])){base4.3[[c]][r,j]<-
sample(as.numeric(pend.order2[[c]][,j][2:5]),1)}else
        if (base4.3[[c]][r,j]!=as.numeric(pend.order2[[c]][,j][1])){base4.3[[c]][r,j]}
    }
  }
}

pend.order2[[1]][,1]
table(base.e2[[1]][,1]);table(base4.3[[1]][,1])

#Pasando a 3 opciones
base3.3 <- base4.3
for (e in 1:replica){
  for (k in 1:ncol(base4[[e]])){
    for(m in 1:n){
      if              (base3.3[[e]][m,k]==as.numeric(pend.order2[[e]][,k][2])){base3.3[[e]][m,k]<-
sample(as.numeric(pend.order2[[e]][,k][3:5]),1)}else
        if (base3.3[[e]][m,k]!=as.numeric(pend.order2[[e]][,k][2])){base3.3[[e]][m,k]}
    }
  }
}

pend.order2[[1]][,1]
table(base.e2[[1]][,1]);table(base4.3[[1]][,1]);table(base3.3[[1]][,1])
opcion.e2[[1]][1,]

#-------------------- Dicotomizando las bases -----------------------------------------#
base5   <- list()
base4pb <- list()
base3pb <- list()
base4pm <- list()
base3pm <- list()
base4a  <- list()
base3a  <- list()

for (w in 1:replica){
  base5[[w]]   <- key2binary(base.e2[[w]], key=opcion.e2[[w]][,1], score_missing = FALSE)
  base4pb[[w]] <- key2binary(base4[[w]],   key=opcion.e2[[w]][,1], score_missing = FALSE)
  base3pb[[w]] <- key2binary(base3[[w]],   key=opcion.e2[[w]][,1], score_missing = FALSE)
  base4pm[[w]] <- key2binary(base4.2[[w]], key=opcion.e2[[w]][,1], score_missing = FALSE)
  base3pm[[w]] <- key2binary(base3.2[[w]], key=opcion.e2[[w]][,1], score_missing = FALSE)
  base4a[[w]]  <- key2binary(base4.3[[w]], key=opcion.e2[[w]][,1], score_missing = FALSE)
  base3a[[w]]  <- key2binary(base3.3[[w]], key=opcion.e2[[w]][,1], score_missing = FALSE)
}

length(base.e2[[6]]);length(opcion.e2[[6]][,1])

prop.table(table(base5[[1]]))
prop.table(table(base4pb[[1]]))
prop.table(table(base3pb[[1]]))
prop.table(table(base4pm[[1]]))
prop.table(table(base3pm[[1]]))
prop.table(table(base4a[[1]]))
prop.table(table(base3a[[1]]))

table(base4pb[[1]])
table(base3pb[[1]])
table(base4pm[[1]])
table(base3pm[[1]])
```

```
table(base4a[[1]])
table(base3a[[1]])

rm(base5)
rm(base4pb)
rm(base3pb)
rm(base4pm)
rm(base3pm)
rm(base4a)
rm(base3a)
#-----------------------Analisis con el modelo de 2PL---------------------------------------#
mod5   <- list()
mod4pb <- list()
mod3pb <- list()
mod4pm <- list()
mod3pm <- list()
mod4a  <- list()
mod3a  <- list()

coef.mod5   <- list()
coef.mod4pb <- list()
coef.mod3pb <- list()
coef.mod4pm <- list()
coef.mod3pm <- list()
coef.mod4a  <- list()
coef.mod3a  <- list()

dif.5   <- list()
dif.4pb <- list()
dif.3pb <- list()
dif.4pm <- list()
dif.3pm <- list()
dif.4a  <- list()
dif.3a  <- list()

discri.5   <- list()
discri.4pb <- list()
discri.3pb <- list()
discri.4pm <- list()
discri.3pm <- list()
discri.4a  <- list()
discri.3a  <- list()

AIC.5   <- list()
AIC.4pb <- list()
AIC.3pb <- list()
AIC.4pm <- list()
AIC.3pm <- list()
AIC.4a  <- list()
AIC.3a  <- list()

recm.dif.4pb   <-list()
recm.dif.4pm   <-list()
recm.dif.4a    <-list()
recm.dif.3pb   <-list()
recm.dif.3pm   <-list()
recm.dif.3a    <-list()

recm.discri.4pb<-list()
recm.discri.4pm<-list()
```

```
recm.discri.4a <-list()
recm.discri.3pb<-list()
recm.discri.3pm<-list()
recm.discri.3a <-list()

t <- proc.time()
for (i in 1:replica){

  #generando los modelos
  mod5[[i]]  =ltm(base5[[i]]  ~z1)#nivel de habilidad unidimensional
  mod4pb[[i]]=ltm(base4pb[[i]]~z1)#nivel de habilidad unidimensional
  mod3pb[[i]]=ltm(base3pb[[i]]~z1)#nivel de habilidad unidimensional
  mod4pm[[i]]=ltm(base4pm[[i]]~z1)#nivel de habilidad unidimensional
  mod3pm[[i]]=ltm(base3pm[[i]]~z1)#nivel de habilidad unidimensional
  mod4a[[i]] =ltm(base4a[[i]] ~z1)#nivel de habilidad unidimensional
  mod3a[[i]] =ltm(base3a[[i]] ~z1)#nivel de habilidad unidimensional

  #Coeficientes
  coef.mod5[[i]] <- coef(mod5[[i]])
  coef.mod4pb[[i]] <- coef(mod4pb[[i]])
  coef.mod3pb[[i]] <- coef(mod3pb[[i]])
  coef.mod4pm[[i]]  <- coef(mod4pm[[i]])
  coef.mod3pm[[i]] <- coef(mod3pm[[i]])
  coef.mod4a[[i]]  <- coef(mod4a[[i]])
  coef.mod3a[[i]]  <- coef(mod3a[[i]])

  #Coeficientes de dificultad
  dif.5[[i]]    <- coef.mod5[[i]][,1]
  dif.4pb[[i]] <- coef.mod4pb[[i]][,1]
  dif.3pb[[i]] <- coef.mod3pb[[i]][,1]
  dif.4pm[[i]] <- coef.mod4pm[[i]][,1]
  dif.3pm[[i]] <- coef.mod3pm[[i]][,1]
  dif.4a[[i]]  <- coef.mod4a[[i]][,1]
  dif.3a[[i]]  <- coef.mod3a[[i]][,1]

  #Coeficientes de discriminacion
  discri.5[[i]]   <- coef.mod5[[i]][,2]
  discri.4pb[[i]] <- coef.mod4pb[[i]][,2]
  discri.3pb[[i]] <- coef.mod3pb[[i]][,2]
  discri.4pm[[i]] <- coef.mod4pm[[i]][,2]
  discri.3pm[[i]] <- coef.mod3pm[[i]][,2]
  discri.4a[[i]]  <- coef.mod4a[[i]][,2]
  discri.3a[[i]]  <- coef.mod3a[[i]][,2]

  #Generando los AIC
  AIC.5[[i]]   <-AIC(mod5[[i]])
  AIC.4pb[[i]] <- AIC(mod4pb[[i]])
  AIC.4pm[[i]] <- AIC(mod4pm[[i]])
  AIC.4a [[i]] <- AIC(mod4a[[i]])
  AIC.3pb[[i]] <- AIC(mod3pb[[i]])
  AIC.3pm[[i]] <- AIC(mod3pm[[i]])
  AIC.3a [[i]] <- AIC(mod3a[[i]])
}
proc.time() - t


#--------------------------------Guardando bases ----------------------------------------------#
save(coef.mod5   ,file="coef.mod5.rda")
save(coef.mod4pb ,file="coef.mod4pb.rda")
save(coef.mod3pb ,file="coef.mod3pb.rda")
```

```
save(coef.mod4pm ,file="coef.mod4pm.rda")
save(coef.mod3pm ,file="coef.mod3pm.rda")
save(coef.mod4a  ,file="coef.mod4a.rda")
save(coef.mod3a  ,file="coef.mod3a.rda")

load("coef.mod5.rda")
load("coef.mod4pb.rda")
load("coef.mod3pb.rda")
load("coef.mod4pm.rda")
load("coef.mod3pm.rda")
load("coef.mod4a.rda")
load("coef.mod3a.rda")

save(dif.5   ,file="dif.5.rda")
save(dif.4pb ,file="dif.4pb.rda")
save(dif.3pb ,file="dif.3pb.rda")
save(dif.4pm ,file="dif.4pm.rda")
save(dif.3pm ,file="dif.3pm.rda")
save(dif.4a  ,file="dif.4a.rda")
save(dif.3a  ,file="dif.3a.rda")

load("dif.5.rda")
load("dif.4pb.rda")
load("dif.3pb.rda")
load("dif.4pm.rda")
load("dif.3pm.rda")
load("dif.4a.rda")
load("dif.3a.rda")

save(discri.5  ,file= "discri.5.rda")
save(discri.4pb,file= "discri.4pb.rda")
save(discri.3pb,file= "discri.3pb.rda")
save(discri.4pm,file= "discri.4pm.rda")
save(discri.3pm,file= "discri.3pm.rda")
save(discri.4a ,file= "discri.4a.rda")
save(discri.3a ,file= "discri.3a.rda")

load("discri.5.rda")
load("discri.4pb.rda")
load("discri.3pb.rda")
load("discri.4pm.rda")
load("discri.3pm.rda")
load("discri.4a.rda")
load("discri.3a.rda")

save(AIC.5  ,file="AIC.5.rda")
save(AIC.4pb,file="AIC.4pb.rda")
save(AIC.3pb,file="AIC.3pb.rda")
save(AIC.4pm,file="AIC.4pm.rda")
save(AIC.3pm,file="AIC.3pm.rda")
save(AIC.4a ,file="AIC.4a.rda")
save(AIC.3a ,file="AIC.3a.rda")

load("AIC.5.rda")
load("AIC.4pb.rda")
load("AIC.3pb.rda")
load("AIC.4pm.rda")
load("AIC.3pm.rda")
load("AIC.4a.rda")
load("AIC.3a.rda")
```

```
#------------------------- convirtiendo en DF las listas de AIC ----------------------

base.aic.5<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.5, SIMPLIFY = FALSE)

base.aic.4pb<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.4pb, SIMPLIFY = FALSE)

base.aic.4pm<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.4pm, SIMPLIFY = FALSE)

base.aic.4a<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.4a, SIMPLIFY = FALSE)

base.aic.3pb<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.3pb, SIMPLIFY = FALSE)

base.aic.3pm<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.3pm, SIMPLIFY = FALSE)

base.aic.3a<-mapply(function(data.aic){
  as.data.frame(data.aic)},
  data.aic=AIC.3a, SIMPLIFY = FALSE)

aic.5  <-do.call(rbind.data.frame,base.aic.5)
aic.4pb<-do.call(rbind.data.frame,base.aic.4pb)
aic.4pm<-do.call(rbind.data.frame,base.aic.4pm)
aic.4a <-do.call(rbind.data.frame,base.aic.4a )
aic.3pb<-do.call(rbind.data.frame,base.aic.3pb)
aic.3pm<-do.call(rbind.data.frame,base.aic.3pm)
aic.3a <-do.call(rbind.data.frame,base.aic.3a )

mean(aic.5  $data.aic)
mean(aic.4pb$data.aic)
mean(aic.4pm$data.aic)
mean(aic.4a $data.aic)
mean(aic.3pb$data.aic)
mean(aic.3pm$data.aic)
mean(aic.3a $data.aic)

#-----------convirtiendo en DF las listas de dificultad y discriminación------------------------#

base.coef.dif.5<-mapply(function(data.dif){
  t(as.data.frame(data.dif))},
  data.dif=dif.5, SIMPLIFY = FALSE)

base.coef.dif.4pb<-mapply(function(data.dif){
  t(as.data.frame(data.dif))},
  data.dif=dif.4pb, SIMPLIFY = FALSE)

base.coef.dif.4pm<-mapply(function(data.dif){
  t(as.data.frame(data.dif))},
  data.dif=dif.4pm, SIMPLIFY = FALSE)
```

```
base.coef.dif.4a<-mapply(function(data.dif){
  t(as.data.frame(data.dif))},
  data.dif=dif.4a, SIMPLIFY = FALSE)

base.coef.dif.3pb<-mapply(function(data.dif){
  t(as.data.frame(data.dif))},
  data.dif=dif.3pb, SIMPLIFY = FALSE)

base.coef.dif.3pm<-mapply(function(data.dif){
  t(as.data.frame(data.dif))},
  data.dif=dif.3pm, SIMPLIFY = FALSE)

base.coef.dif.3a<-mapply(function(data.dif){
  t(as.data.frame(data.dif))},
  data.dif=dif.3a, SIMPLIFY = FALSE)

base.coef.discri.5<-mapply(function(data.discri){
  t(as.data.frame(data.discri))},
  data.discri=discri.5, SIMPLIFY = FALSE)

base.coef.discri.4pb<-mapply(function(data.discri){
  t(as.data.frame(data.discri))},
  data.discri=discri.4pb, SIMPLIFY = FALSE)

base.coef.discri.4pm<-mapply(function(data.discri){
  t(as.data.frame(data.discri))},
  data.discri=discri.4pm, SIMPLIFY = FALSE)

base.coef.discri.4a<-mapply(function(data.discri){
  t(as.data.frame(data.discri))},
  data.discri=discri.4a, SIMPLIFY = FALSE)

base.coef.discri.3pb<-mapply(function(data.discri){
  t(as.data.frame(data.discri))},
  data.discri=discri.3pb, SIMPLIFY = FALSE)

base.coef.discri.3pm<-mapply(function(data.discri){
  t(as.data.frame(data.discri))},
  data.discri=discri.3pm, SIMPLIFY = FALSE)

base.coef.discri.3a<-mapply(function(data.discri){
  t(as.data.frame(data.discri))},
  data.discri=discri.3a, SIMPLIFY = FALSE)

#----------------------selecionar items que están en todas las replicas -------------------------
----#

# unir los DF
c.dif.5   <- Reduce(function(...)merge(..., all=TRUE),base.coef.dif.5)
c.dif.4pb <- Reduce(function(...)merge(..., all=TRUE),base.coef.dif.4pb)
c.dif.3pb <- Reduce(function(...)merge(..., all=TRUE),base.coef.dif.3pb)
c.dif.4pm <- Reduce(function(...)merge(..., all=TRUE),base.coef.dif.4pm)
c.dif.3pm <- Reduce(function(...)merge(..., all=TRUE),base.coef.dif.3pm)
c.dif.4a  <- Reduce(function(...)merge(..., all=TRUE),base.coef.dif.4a)
c.dif.3a  <- Reduce(function(...)merge(..., all=TRUE),base.coef.dif.3a)

c.discri.5   <- Reduce(function(...)merge(..., all=TRUE),base.coef.discri.5)
c.discri.4pb <- Reduce(function(...)merge(..., all=TRUE),base.coef.discri.4pb)
c.discri.3pb <- Reduce(function(...)merge(..., all=TRUE),base.coef.discri.3pb)
```

```
c.discri.4pm <- Reduce(function(...)merge(..., all=TRUE),base.coef.discri.4pm)
c.discri.3pm <- Reduce(function(...)merge(..., all=TRUE),base.coef.discri.3pm)
c.discri.4a  <- Reduce(function(...)merge(..., all=TRUE),base.coef.discri.4a)
c.discri.3a  <- Reduce(function(...)merge(..., all=TRUE),base.coef.discri.3a)

#a <- c.dif.5 [ , order(c(as.numeric(names(c.dif.5))))]
c.dif.5   <- c.dif.5  [ , order(c(as.numeric(names(c.dif.5  ))))]
c.dif.4pb <- c.dif.4pb[ , order(c(as.numeric(names(c.dif.4pb))))]
c.dif.3pb <- c.dif.3pb[ , order(c(as.numeric(names(c.dif.3pb))))]
c.dif.4pm <- c.dif.4pm[ , order(c(as.numeric(names(c.dif.4pm))))]
c.dif.3pm <- c.dif.3pm[ , order(c(as.numeric(names(c.dif.3pm))))]
c.dif.4a  <- c.dif.4a [ , order(c(as.numeric(names(c.dif.4a ))))]
c.dif.3a  <- c.dif.3a [ , order(c(as.numeric(names(c.dif.3a ))))]

c.discri.5   <- c.discri.5  [ , order(c(as.numeric(names(c.discri.5  ))))]
c.discri.4pb <- c.discri.4pb[ , order(c(as.numeric(names(c.discri.4pb))))]
c.discri.3pb <- c.discri.3pb[ , order(c(as.numeric(names(c.discri.3pb))))]
c.discri.4pm <- c.discri.4pm[ , order(c(as.numeric(names(c.discri.4pm))))]
c.discri.3pm <- c.discri.3pm[ , order(c(as.numeric(names(c.discri.3pm))))]
c.discri.4a  <- c.discri.4a [ , order(c(as.numeric(names(c.discri.4a ))))]
c.discri.3a  <- c.discri.3a [ , order(c(as.numeric(names(c.discri.3a ))))]

names(c.dif.5  )
names(c.dif.4pb)
names(c.dif.3pb)
names(c.dif.4pm)
names(c.dif.3pm)
names(c.dif.4a )
names(c.dif.3a )

names(c.discri.5  )
names(c.discri.4pb)
names(c.discri.3pb)
names(c.discri.4pm)
names(c.discri.3pm)
names(c.discri.4a )
names(c.discri.3a )

# Determinar la cantidad de items que quedan por replica
table(unlist(lapply(base.coef.dif.5,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.dif.4pb,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.dif.3pb,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.dif.4pm,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.dif.3pm,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.dif.4a,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.dif.3a,colnames), recursive = FALSE, use.names = TRUE))

p2 <- table(unlist(lapply(base.coef.dif.5,colnames), recursive = FALSE, use.names = TRUE))
p3 <- subset(p2,p2>= 80 ) # los items que estan en 80% de las replicas
p3 <- as.numeric(names(p3))
p3 <- sort(p3)
p4 <- subset(p2,p2 < 80 ) # los items que no estan en 80% de las replicas
p4 <- as.numeric(names(p4))
p4 <- sort(p4)

table(unlist(lapply(base.coef.discri.5,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.discri.4pb,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.discri.3pb,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.discri.4pm,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.discri.3pm,colnames), recursive = FALSE, use.names = TRUE))
```

```
table(unlist(lapply(base.coef.discri.4a,colnames), recursive = FALSE, use.names = TRUE))
table(unlist(lapply(base.coef.discri.3a,colnames), recursive = FALSE, use.names = TRUE))

p2.discri  <-  table(unlist(lapply(base.coef.discri.5,colnames),  recursive  =  FALSE,  use.names  =
TRUE))
p3.discri <- subset(p2.discri,p2.discri>= 80 ) # los items que estan en 80% de las replicas
p3.discri <- as.numeric(names(p3.discri))
p3.discri <- sort(p3.discri)
p4.discri <- subset(p2.discri,p2.discri < 80 ) # los items que no estan en 80% de las replicas
p4.discri <- as.numeric(names(p4.discri))
p4.discri <- sort(p4.discri)

#seleccionar items que estén en mas del 80% de las replicas

c.dif.5   <- c.dif.5  [ , -p4]
c.dif.4pb <- c.dif.4pb[ , -p4]
c.dif.3pb <- c.dif.3pb[ , -p4]
c.dif.4pm <- c.dif.4pm[ , -p4]
c.dif.3pm <- c.dif.3pm[ , -p4]
c.dif.4a  <- c.dif.4a [ , -p4]
c.dif.3a  <- c.dif.3a [ , -p4]

c.discri.5   <- c.discri.5  [ , -p4.discri]
c.discri.4pb <- c.discri.4pb[ , -p4.discri]
c.discri.3pb <- c.discri.3pb[ , -p4.discri]
c.discri.4pm <- c.discri.4pm[ , -p4.discri]
c.discri.3pm <- c.discri.3pm[ , -p4.discri]
c.discri.4a  <- c.discri.4a [ , -p4.discri]
c.discri.3a  <- c.discri.3a [ , -p4.discri]

library(openxlsx)
write.xlsx(c.dif.5  ,"c.dif.5.xlsx"  ,colNames = TRUE, borders = "columns")
write.xlsx(c.dif.4pb,"c.dif.4pb.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.dif.3pb,"c.dif.3pb.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.dif.4pm,"c.dif.4pm.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.dif.3pm,"c.dif.3pm.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.dif.4a ,"c.dif.4a.xlsx" ,colNames = TRUE, borders = "columns")
write.xlsx(c.dif.3a ,"c.dif.3a.xlsx" ,colNames = TRUE, borders = "columns")

write.xlsx(c.discri.5  ,"c.discri.5.xlsx"  ,colNames = TRUE, borders = "columns")
write.xlsx(c.discri.4pb,"c.discri.4pb.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.discri.3pb,"c.discri.3pb.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.discri.4pm,"c.discri.4pm.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.discri.3pm,"c.discri.3pm.xlsx",colNames = TRUE, borders = "columns")
write.xlsx(c.discri.4a ,"c.discri.4a.xlsx" ,colNames = TRUE, borders = "columns")
write.xlsx(c.discri.3a ,"c.discri.3a.xlsx" ,colNames = TRUE, borders = "columns")

#rm(c.dif.5)
#rm(c.dif.4pb)
#rm(c.dif.3pb)
#rm(c.dif.4pm)
#rm(c.dif.3pm)
#rm(c.dif.4a )
#rm(c.dif.3a )

# seleccionar replicas que esten entre los umbrales

dif.optimo5   <- apply(c.dif.5  ,2,function(x) which(x > -6 & x < 6))
dif.optimo4pb <- apply(c.dif.4pb,2,function(x) which(x > -6 & x < 6))
dif.optimo3pb <- apply(c.dif.3pb,2,function(x) which(x > -6 & x < 6))
```

```
dif.optimo4pm <- apply(c.dif.4pm,2,function(x) which(x > -6 & x < 6))
dif.optimo3pm <- apply(c.dif.3pm,2,function(x) which(x > -6 & x < 6))
dif.optimo4a  <- apply(c.dif.4a ,2,function(x) which(x > -6 & x < 6))
dif.optimo3a  <- apply(c.dif.3a ,2,function(x) which(x > -6 & x < 6))

discri.optimo5   <- apply(c.discri.5  ,2,function(x) which(x > -6 & x < 6))
discri.optimo4pb <- apply(c.discri.4pb,2,function(x) which(x > -6 & x < 6))
discri.optimo3pb <- apply(c.discri.3pb,2,function(x) which(x > -6 & x < 6))
discri.optimo4pm <- apply(c.discri.4pm,2,function(x) which(x > -6 & x < 6))
discri.optimo3pm <- apply(c.discri.3pm,2,function(x) which(x > -6 & x < 6))
discri.optimo4a  <- apply(c.discri.4a ,2,function(x) which(x > -6 & x < 6))
discri.optimo3a  <- apply(c.discri.3a ,2,function(x) which(x > -6 & x < 6))

# unir por metodo
dif.optimo5<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo5, SIMPLIFY = FALSE)

dif.optimo4pb<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo4pb, SIMPLIFY = FALSE)

dif.optimo3pb<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo3pb, SIMPLIFY = FALSE)

dif.optimo4pm<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo4pm, SIMPLIFY = FALSE)

dif.optimo3pm<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo3pm, SIMPLIFY = FALSE)

dif.optimo4a<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo4a, SIMPLIFY = FALSE)

dif.optimo3a<-mapply(function(data){
  as.data.frame(data)},
  data=dif.optimo3a, SIMPLIFY = FALSE)

discri.optimo5<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo5, SIMPLIFY = FALSE)

discri.optimo4pb<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo4pb, SIMPLIFY = FALSE)

discri.optimo3pb<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo3pb, SIMPLIFY = FALSE)

discri.optimo4pm<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo4pm, SIMPLIFY = FALSE)

discri.optimo3pm<-mapply(function(data){
  as.data.frame(data)},
```

```
  data=discri.optimo3pm, SIMPLIFY = FALSE)

discri.optimo4a<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo4a, SIMPLIFY = FALSE)

discri.optimo3a<-mapply(function(data){
  as.data.frame(data)},
  data=discri.optimo3a, SIMPLIFY = FALSE)

#union de optimos
dif.optimo.pb<-mapply(function(data1,data2,data3){
  Reduce(merge, list(data1, data2,data3))},
  data1=dif.optimo5,data2=dif.optimo4pb, data3=dif.optimo3pb, SIMPLIFY = FALSE)

dif.optimo.pm<-mapply(function(data1,data2,data3){
  Reduce(merge, list(data1, data2,data3))},
  data1=dif.optimo5,data2=dif.optimo4pm, data3=dif.optimo3pm, SIMPLIFY = FALSE)

dif.optimo.a<-mapply(function(data1,data2,data3){
  Reduce(merge, list(data1, data2,data3))},
  data1=dif.optimo5,data2=dif.optimo4a, data3=dif.optimo3a, SIMPLIFY = FALSE)

discri.optimo.pb<-mapply(function(data1,data2,data3){
  Reduce(merge, list(data1, data2,data3))},
  data1=discri.optimo5,data2=discri.optimo4pb, data3=discri.optimo3pb, SIMPLIFY = FALSE)

discri.optimo.pm<-mapply(function(data1,data2,data3){
  Reduce(merge, list(data1, data2,data3))},
  data1=discri.optimo5,data2=discri.optimo4pm, data3=discri.optimo3pm, SIMPLIFY = FALSE)

discri.optimo.a<-mapply(function(data1,data2,data3){
  Reduce(merge, list(data1, data2,data3))},
  data1=discri.optimo5,data2=discri.optimo4a, data3=discri.optimo3a, SIMPLIFY = FALSE)

#rm(dif.optimo.pb)
#rm(dif.optimo.pm)
#rm(dif.optimo.a)

#Seleccionar bases optimas

sqrt(sum((c.dif.4pb$`9`[unlist(dif.optimo.pb$`9`)]                                    -
c.dif.5$`9`[unlist(dif.optimo.pb$`9`)])^2)/length(unlist(dif.optimo.pb$`9`)))

sqrt(sum((c.dif.4pb[unlist(dif.optimo.pb[[1]]),1]                                     -
c.dif.5[unlist(dif.optimo.pb[[1]]),1])^2)/length(unlist(dif.optimo.pb[[1]])))

# Calculo de ECM

recm.dif.4pb   <-list()
recm.dif.4pm   <-list()
recm.dif.4a    <-list()
recm.dif.3pb   <-list()
recm.dif.3pm   <-list()
recm.dif.3a    <-list()

recm.discri.4pb<-list()
recm.discri.4pm<-list()
recm.discri.4a <-list()
recm.discri.3pb<-list()
```

```
recm.discri.3pm<-list()
recm.discri.3a <-list()


t <- proc.time()
for (i in 1:ncol(c.dif.5)){
  #RECM de dificultad
  recm.dif.4pb[[i]]             <-             sqrt(sum((c.dif.4pb[unlist(dif.optimo.pb[[i]]),i]-
c.dif.5[unlist(dif.optimo.pb[[i]]),i])^2)/length(unlist(dif.optimo.pb[[i]])))
  recm.dif.4pm[[i]]             <-             sqrt(sum((c.dif.4pm[unlist(dif.optimo.pm[[i]]),i]-
c.dif.5[unlist(dif.optimo.pm[[i]]),i])^2)/length(unlist(dif.optimo.pm[[i]])))
  recm.dif.4a[[i]]                   <-          sqrt(sum((c.dif.4a[unlist(dif.optimo.a[[i]]),i]-
c.dif.5[unlist(dif.optimo.a[[i]]),i])^2)/length(unlist(dif.optimo.a[[i]])))
  recm.dif.3pb[[i]]             <-             sqrt(sum((c.dif.3pb[unlist(dif.optimo.pb[[i]]),i]-
c.dif.5[unlist(dif.optimo.pb[[i]]),i])^2)/length(unlist(dif.optimo.pb[[i]])))
  recm.dif.3pm[[i]]             <-             sqrt(sum((c.dif.3pm[unlist(dif.optimo.pm[[i]]),i]-
c.dif.5[unlist(dif.optimo.pm[[i]]),i])^2)/length(unlist(dif.optimo.pm[[i]])))
  recm.dif.3a[[i]]                   <-          sqrt(sum((c.dif.3a[unlist(dif.optimo.a[[i]]),i]-
c.dif.5[unlist(dif.optimo.a[[i]]),i])^2)/length(unlist(dif.optimo.a[[i]])))

  #RECM de discriminacion
  recm.discri.4pb[[i]]        <-        sqrt(sum((c.discri.4pb[unlist(discri.optimo.pb[[i]]),i]-
c.discri.5[unlist(discri.optimo.pb[[i]]),i])^2)/length(unlist(discri.optimo.pb[[i]])))
  recm.discri.4pm[[i]]        <-        sqrt(sum((c.discri.4pm[unlist(discri.optimo.pm[[i]]),i]-
c.discri.5[unlist(discri.optimo.pm[[i]]),i])^2)/length(unlist(discri.optimo.pm[[i]])))
  recm.discri.4a[[i]]          <-     sqrt(sum((c.discri.4a     [unlist(discri.optimo.a[[i]]),i]-
c.discri.5[unlist(discri.optimo.a[[i]]),i])^2)/length(unlist(discri.optimo.a[[i]])))
  recm.discri.3pb[[i]]        <-        sqrt(sum((c.discri.3pb[unlist(discri.optimo.pb[[i]]),i]-
c.discri.5[unlist(discri.optimo.pb[[i]]),i])^2)/length(unlist(discri.optimo.pb[[i]])))
  recm.discri.3pm[[i]]        <-        sqrt(sum((c.discri.3pm[unlist(discri.optimo.pm[[i]]),i]-
c.discri.5[unlist(discri.optimo.pm[[i]]),i])^2)/length(unlist(discri.optimo.pm[[i]])))
  recm.discri.3a[[i]]          <-     sqrt(sum((c.discri.3a     [unlist(discri.optimo.a[[i]]),i]-
c.discri.5[unlist(discri.optimo.a[[i]]),i])^2)/length(unlist(discri.optimo.a[[i]])))

}
proc.time() - t

#
base.recm.dif.4pb<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.4pb, SIMPLIFY = FALSE)

base.recm.dif.4pm<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.4pm, SIMPLIFY = FALSE)

base.recm.dif.4a<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.4a, SIMPLIFY = FALSE)

base.recm.dif.3pb<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.3pb, SIMPLIFY = FALSE)

base.recm.dif.3pm<-mapply(function(data.dif){
  as.data.frame(data.dif)},
  data.dif=recm.dif.3pm, SIMPLIFY = FALSE)

base.recm.dif.3a<-mapply(function(data.dif){
  as.data.frame(data.dif)},
```

```
  data.dif=recm.dif.3a, SIMPLIFY = FALSE)

# uniendo en DF los RECM dificultad

recm.dif.4pb<-do.call(rbind.data.frame,base.recm.dif.4pb)
recm.dif.4pm<-do.call(rbind.data.frame,base.recm.dif.4pm)
recm.dif.4a <-do.call(rbind.data.frame,base.recm.dif.4a )
recm.dif.3pb<-do.call(rbind.data.frame,base.recm.dif.3pb)
recm.dif.3pm<-do.call(rbind.data.frame,base.recm.dif.3pm)
recm.dif.3a <-do.call(rbind.data.frame,base.recm.dif.3a )


recm.dif                                                                                     <-
rbind(t(recm.dif.4pb),t(recm.dif.3pb),t(recm.dif.4pm),t(recm.dif.3pm),t(recm.dif.4a),t(recm.dif.3
a))
rownames(recm.dif) <- c("4 proba baja","3 proba baja","4 proba alta","3 proba alta","4 azar","3
azar")
colnames(recm.dif) <-c(colnames(c.dif.5))
#colnames(recm.dif) <- p3

write.xlsx(recm.dif, "recm.dif.xlsx",colNames = TRUE, borders = "columns" )

save(recm.dif,file="recm.dif.rda")
load("recm.dif.rda")

mean.recm.dif <- apply(recm.dif, 1, mean, na.rm = TRUE)
save(mean.recm.dif,file="mean.recm.dif.rda")
load("mean.recm.dif.rda")
write.xlsx(mean.recm.dif, "mean.recm.dif.xlsx",colNames = TRUE, borders = "columns" )


#--------------recm de discriminacion--------------------------------------------#
base.recm.discri.4pb<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.4pb, SIMPLIFY = FALSE)

base.recm.discri.4pm<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.4pm, SIMPLIFY = FALSE)

base.recm.discri.4a<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.4a, SIMPLIFY = FALSE)

base.recm.discri.3pb<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.3pb, SIMPLIFY = FALSE)

base.recm.discri.3pm<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.3pm, SIMPLIFY = FALSE)

base.recm.discri.3a<-mapply(function(data.discri){
  as.data.frame(data.discri)},
  data.discri=recm.discri.3a, SIMPLIFY = FALSE)

recm.discri.4pb<-do.call(rbind.data.frame,base.recm.discri.4pb)
recm.discri.4pm<-do.call(rbind.data.frame,base.recm.discri.4pm)
recm.discri.4a <-do.call(rbind.data.frame,base.recm.discri.4a )
recm.discri.3pb<-do.call(rbind.data.frame,base.recm.discri.3pb)
```

```
recm.discri.3pm<-do.call(rbind.data.frame,base.recm.discri.3pm)
recm.discri.3a <-do.call(rbind.data.frame,base.recm.discri.3a )

mean(na.omit(recm.discri.4pb$data.discri))
mean(na.omit(recm.discri.4pm$data.discri))
mean(na.omit(recm.discri.4a $data.discri))
mean(na.omit(recm.discri.3pb$data.discri))
mean(na.omit(recm.discri.3pm$data.discri))
mean(na.omit(recm.discri.3a $data.discri))

recm.discri                                                                                   <-
rbind(t(recm.discri.4pb),t(recm.discri.3pb),t(recm.discri.4pm),t(recm.discri.3pm),t(recm.discri.4
a),t(recm.discri.3a))
rownames(recm.discri) <- c("4 proba baja","3 proba baja","4 proba alta","3 proba alta","4 azar","3
azar")
colnames(recm.discri) <- c(colnames(c.discri.5))

write.xlsx(recm.discri, "recm.discri.xlsx",colNames = TRUE, borders = "columns" )

save(recm.discri,file="recm.discri.rda")
load("recm.discri.rda")

mean.recm.discri <- apply(recm.discri, 1, mean, na.rm = TRUE)
save(mean.recm.discri,file="mean.recm.discri.rda")
load("mean.recm.discri.rda")
write.xlsx(mean.recm.discri, "mean.recm.discri.xlsx",colNames = TRUE, borders = "columns" )
```