

Un test adaptativo informatizado de vocabulario inglés: Descripción del programa

Javier Revuelta, Vicente Ponsoda y Julio Olea

Universidad Autónoma de Madrid

El trabajo describe la estructura y funcionamiento de un programa de ordenador, que es el soporte de un test adaptativo, basado en la TRI, que mide el nivel de vocabulario inglés. Cada ítem consiste en una palabra inglesa y cinco posibles traducciones, que son las alternativas. El lenguaje utilizado es Turbo Pascal 6.0.

El test de vocabulario es una prueba adaptativa presentada por ordenador, basada en la Teoría de Respuesta al Ítem (una introducción a la TRI puede encontrarse en Muñiz, 1990). Mediante los tests adaptativos se pretende presentar a cada sujeto la secuencia de ítems más adecuada para estimar su nivel de habilidad. Esto suele hacerse de modo iterativo, utilizando los procedimientos de la TRI, y representa seguramente su principal ventaja en relación a los tests psicométricos tradicionales, que se presentan en su totalidad y en la misma secuencia para todos los niveles de habilidad. Los principales ingredientes de un test adaptativo presentado por ordenador son tres: a) un amplio banco de ítems con propiedades conocidas, b) un procedimiento para estimar el nivel de habilidad del sujeto y c) un procedimiento para escoger el ítem más adecuado en cada momento.

En la actualidad, el banco de ítems del test adaptativo de vocabulario está formado por 221 ítems de cinco alternativas. Cada uno de ellos consta de una palabra inglesa y cinco posibles traducciones en castellano. La tarea del sujeto es elegir la traducción correcta de la palabra inglesa. En una aplicación previa, se han obtenido los tres parámetros ("a", "b", y "c") de cada uno de los ítems, de acuerdo con el modelo logístico. El test adaptativo estima iterativamente el nivel de vocabulario. En cada paso del proceso, se estima un nivel de habilidad y se presenta el ítem más informativo para dicho nivel del conjunto de ítems remanentes. La estimación final se consigue cuando se cumple algunos de los criterios de terminación, que se expondrán posteriormente.

El soporte del test adaptivo es un programa de ordenador que llamamos "TEST". El objetivo de estas páginas es describir su estructura y funcionamiento. Para que el programa funcione se necesita un ordenador

IBM o compatible, con 640 Kb de RAM y tarjeta gráfica CGA o superior; no se necesita coprocesador matemático pero es recomendable para mayor velocidad.

ESTRUCTURA DEL PROGRAMA

TEST ha sido creado con Turbo Pascal 6.0 para DOS (Borland, 1990a, b), utilizando la técnica de programación orientada a objetos; que consiste en la división del código en diversos módulos (u objetos) relativamente independientes entre sí, cada uno de los cuales sirve a una determinada función. Estos módulos pueden ser sumamente complejos, pudiendo incluir: a) variables, b) procedimientos (o métodos), c) otros módulos y d) cualquier tipo de estructura de datos. Entre los módulos pueden establecerse relaciones de jerarquía, por las cuales un objeto hereda las propiedades del anterior. La utilización de esta técnica permite crear un código más claro y, sobre todo, la incorporación y reciclado del mismo en futuros programas.

Los principales núcleos del programa son cuatro: a) La jerarquía "**estimador -> selector -> ejecutor**". b) El banco de ítems. c) La jerarquía "**pantalla -> ayudas -> control_de_ayuda**". y d) La pila. Todos ellos se comentan a continuación

a) La jerarquía "**estimador -> selector -> ejecutor**".

La espina dorsal del programa está compuesta por la jerarquía de objetos: "**estimador -> selector -> ejecutor**". de la cual dependen todas las demás estructuras del mismo. La función de estos objetos se describe a continuación.

El objeto "**Estimador**" estima el nivel de habilidad del sujeto según sus respuestas a los ítems previos y calcula el error típico de dicha estimación. También determina su valor inicial al comienzo del test.

El objeto "**Selector**" escoge el ítem más informativo a partir del nivel de habilidad estimado por el procedimiento anterior.

El objeto "**Ejecutor**" es el que lleva el control del programa. Contiene el algoritmo principal, que guía todo el proceso. Sus principales procedimientos son cinco, que se corresponden con las fases expuestas a continuación:

- 1) Portada: Dibuja la portada inicial y carga en la memoria el banco de ítems.
- 2) Iniciales: Toma las iniciales del sujeto.
- 3) Ejemplos: Pregunta al sujeto si desea ver los ejemplos y, en caso afirmativo, se los presenta.
- 4) Ejecutar: Administra el test adaptativo

5) Resultados: Muestra los resultados finales del test y, eventualmente, llama al objeto **"Informar_al_final"** para que muestre las tres siguientes pantallas de información:

- Información sobre los ítems presentados.
- Curva característica del sujeto.
- Posición del sujeto en la distribución poblacional del nivel de vocabulario, que se supone normal.

b) El banco de ítems.

Otro elemento importante de TEST es la estructura formada por el banco de ítems. Cada ítem en concreto es un caso particular del objeto **"Ítem"**, conteniendo la siguiente información:

- Respuesta del sujeto.
- Alternativa correcta.
- Indicación de si ya ha sido presentado o no.
- Contenido del ítem.
- Parámetros del mismo.
- Puntero al siguiente ítem.
- Procedimientos encargados de dibujar el ítem en pantalla y tomar la respuesta.

Los ítems forman una lista enlazada por punteros, el último de los cuales es la constante *nil* (nulo). Además, existen en el objeto **"Selector"** otros dos punteros, uno fijo al primer ítem y otro que recorre la lista.

c) La jerarquía **"pantalla -> ayudas -> control_de_ayuda"**.

Junto con la jerarquía **"estimador -> selector -> ejecutor"** y la estructura formada por el banco de ítems, existe una segunda jerarquía de objetos formada por: **"pantalla -> ayudas -> control_de_ayuda"**. Como su nombre indica, se encarga de gestionar las ventanas de ayuda que pueden aparecer en distintos momentos:

El objeto **"Pantalla"** se encarga de las funciones generales a todas las pantallas de ayuda, excepto la correspondiente a la obtención de la posición del sujeto en la distribución poblacional, que requiere modo gráfico. Lee la memoria de vídeo en el momento de llamar a la ayuda y almacena su contenido. A continuación, dibuja el fondo de la ventana con las dimensiones especificadas. También se encarga de restaurar la pantalla a su situación original.

El objeto **"Ayudas"** contiene la información específica de las siete pantallas de ayuda que pueden ser llamadas.

El objeto **"Control_de_ayuda"** controla todo el proceso. Cuando se solicita ayuda, se llama a un procedimiento de este objeto, al que se pasa un

número entero que indica desde qué punto del programa se ha realizado la llamada. Dependiendo de este número, selecciona y presenta la pantalla adecuada.

d) La pila.

Aparte de los objetos mencionados es de gran importancia la pila, en la que el programa va almacenando la información específica del test que se está ejecutando. Esta pila se llama "**Apuntado**" y consiste en una cadena de registros. En ella se almacena de cada ítem presentado la siguiente información:

- Los parámetros "a", "b" y "c"
- La respuesta (acierto o error).
- El nivel de habilidad estimado, una vez que se ha emitido la respuesta.
- Error típico de estimación del nivel de habilidad.

La capacidad de la pila es de 36 ítems, ya que no se pueden presentar más de 34. Los dos ítems iniciales están almacenados en los dos primeros registros (véase el siguiente apartado "Funcionamiento").

Esta pila se utiliza para estimar el nivel de habilidad y para calcular el error típico de estimación; también se usa para informar del proceso de estimación y para crear el archivo de datos específico del sujeto.

FUNCIONAMIENTO

El test tiene cinco partes diferenciadas que se corresponden con los cinco procedimientos del objeto "**Ejecutor**"

Mientras está la portada en pantalla, se lee el banco de ítems del archivo "INGLES.STR". La lectura del banco de ítems sólo se produce una vez al ejecutar el programa, mientras que las demás fases se repiten para cada sujeto que pasa el test.

A continuación, se pide al sujeto que teclee sus iniciales, que formarán la extensión de su archivo de datos (cuya creación es opcional). Es necesario introducir tres iniciales para que el programa continúe.

En tercer lugar, se ofrece la posibilidad de responder a tres preguntas de ejemplo, cuyo formato es idéntico a las preguntas reales. La única diferencia es que se concede mayor tiempo para responder (30 segundos), mientras que en la fase de test, quince segundos es el tiempo máximo disponible para responder a cada ítem.

A partir de este momento comienza el test, que se basa en la repetición sucesiva de estimaciones de habilidad y presentaciones de ítems hasta que se cumple el criterio de terminación. Dicho criterio consiste en haber alcanzado un error típico de estimación menor de 0.3 o haber pasado al sujeto 34 ítems. Como nivel de habilidad inicial, se toma un valor aleatorio entre 1 y 1. a

partir del cual se escoge y presenta el ítem que mayor cantidad de información aporta para dicho nivel. A continuación, se estima el nivel de habilidad según la TRI, se escoge y presenta un nuevo ítem, y así sucesivamente (ver figura 1).

El nivel de habilidad se estima según el procedimiento de máxima verosimilitud. Fundamentalmente consiste en encontrar el nivel de habilidad del sujeto que hace más plausible las respuestas de éste a los ítems presentados (Hambleton y Swaminatan, 1985). Para ello, es imprescindible conocer los tres parámetros ("a", "b" y "c") de cada ítem del banco.

Estos parámetros, las respuestas del sujeto y su nivel de habilidad (a estimar) constituyen los ingredientes esenciales de la ecuación de verosimilitud, en cuya resolución se utiliza el método iterativo de Newton-Raphson. El número de iteraciones requeridas no suele ser muy elevado (entre cuatro y seis). No obstante, la rapidez de la convergencia depende del patrón de respuestas y del número de ítems presentados, por lo que, en previsión, se ha fijado un máximo de 200 iteraciones. Transcurridas éstas, si no se ha conseguido resolver la ecuación, se considera el nivel de habilidad como la media aritmética de las cuatro últimas estimaciones. No obstante, la probabilidad de tener que recurrir a este mecanismo de seguridad es prácticamente nula.

Para facilitar la estimación cuando se han pasado pocos ítems y evitar los problemas relacionados con patrones de respuesta consistentes en sólo aciertos o errores, hemos adaptado la estrategia sugerida por Herrando (1989), que consiste en incluir en los cálculos dos ítems adicionales que el sujeto, en realidad, no ha pasado. Uno de ellos tiene un gran nivel de dificultad ($b = 4,2$) y se considera fallado, el otro es sumamente fácil ($b = -4,2$) y se considera acertado. Este procedimiento resuelve el problema antes comentado, no modifica la estimación final de habilidad, ni incrementa el número total de ítems presentados.

A continuación, se calcula el error típico de estimación. El cálculo es directo, por lo que no se necesitan procedimientos iterativos. Si el error resulta ser menor del tope predeterminado, el test se da por concluído.

Estas tareas las realizan dos procedimientos del objeto "**Estimador**" que, como ya se ha dicho, utilizan los datos almacenados en la pila "**Apuntado**", añadiendo en el registro correspondiente al último ítem la habilidad y error típico obtenidos tras responder al mismo.

A partir del nivel de habilidad estimado del sujeto, se calcula la cantidad de información proporcionada por cada uno de los ítems restantes, y se escoge el más informativo: este proceso es el más costoso en términos de tiempo. El ítem escogido se presenta al sujeto y se toma su respuesta. Todo el proceso se repite hasta que se cumple alguno de los criterios de terminación de la prueba.

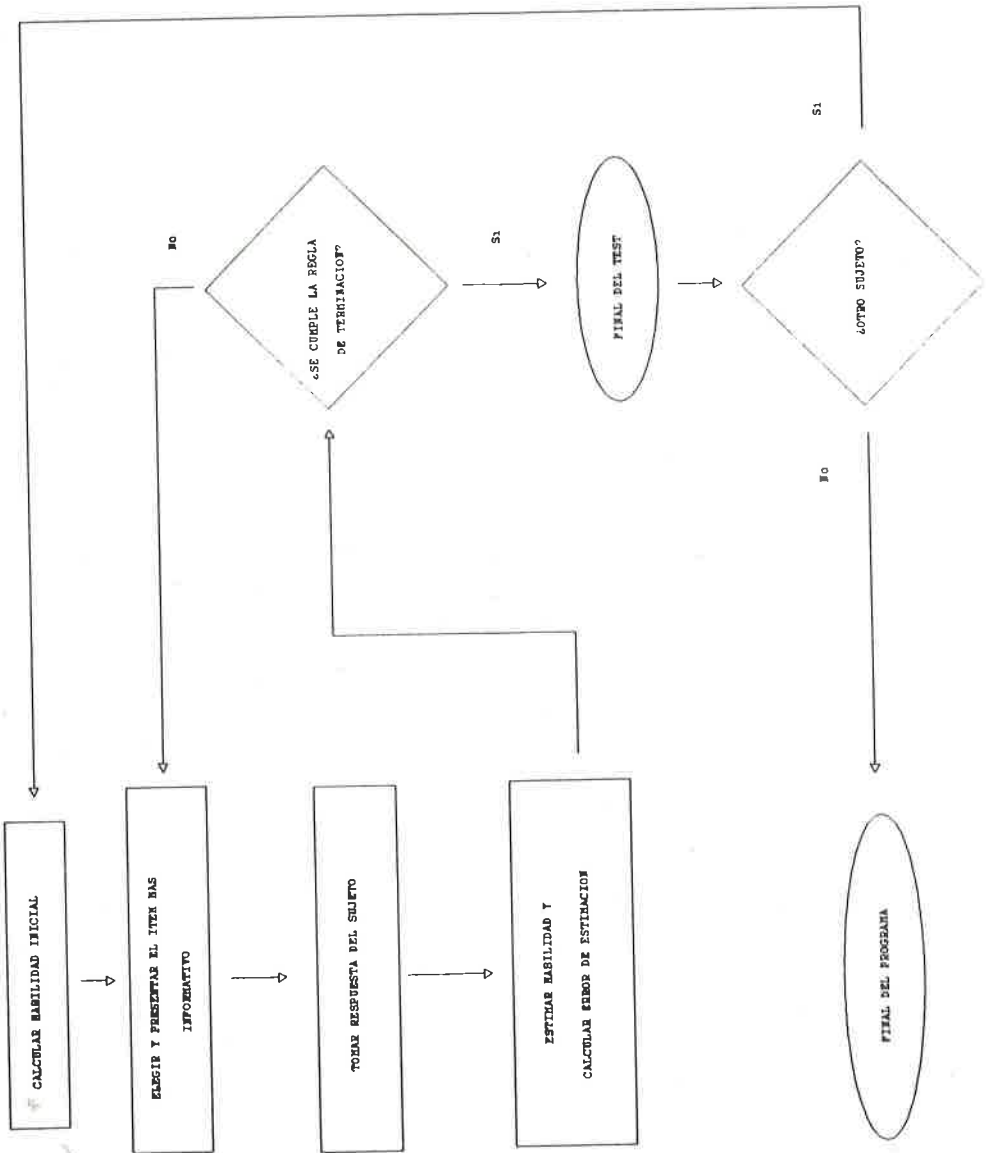


Figura 1. Diagrama de flujo del programa TEST (adaptado de Wainer 1990, p. 108).

Una vez finalizado el test aparece una pantalla de resultados con el nivel de habilidad del sujeto y el error de estimación del mismo; en este momento, se añade esta información al archivo "ZETAS" junto con las iniciales del sujeto, o se crea si no existe. También se presenta un menú con cuatro opciones:

a) **Salir.** Termina el programa y vuelve al DOS.

b) **Repeticir.** Se repite el test desde las iniciales.

c) **Información.** Escogiendo esta opción pueden verse las siguientes pantallas:

c1) **Items presentados.** Parámetros de los mismos, respuesta (acierto/error), habilidad estimada y error típico tras cada uno de ellos.

c2) **Curva característica del sujeto.** Representa su probabilidad de acertar un ítem para cada uno de los niveles de dificultad que pudiera tener (en intervalos de 0,25). esta curva puede ser:

Teórica: Se calcula a partir del nivel de habilidad. Aparece en pantalla formada por asteriscos rojos.

Empírica: Obtenida según las respuestas del sujeto a los ítems presentados, agrupándolos por niveles de dificultad. Es la formada por rectángulos grises.

c3) **Posición que corresponde al nivel de habilidad.**

Asumiendo que la distribución del nivel de habilidad es normal con media cero y varianza uno, se muestra el lugar del sujeto en la misma y el centil, aproximado, que ocupa.

d) **Crear fichero.** Este fichero es específico para el sujeto en cuestión. Tiene como nombre "DATOS" y como extensión sus iniciales. Contiene la misma información que la pantalla "**Items presentados**".

Obtención del programa: Una versión "demostración" del test informatizado de vocabulario de inglés puede obtenerse sin cargo alguno, mandando un disco de 3.5 o 5.25 pulgadas, con formato MSDOS, y con un sobre franqueado, a V. Ponsoda, Facultad de Psicología, Universidad Autónoma, 28049-Madrid. La versión reducida difiere sólo de la expuesta aquí en el tamaño del banco de ítems, que consta sólo de 25 ítems.

ABSTRACT

This work describes the structure and operation of an IRT-based computer adaptive test to measure the ability level of English vocabulary. The stem of each item is an English word and the options are its Spanish translation and four more Spanish words. The program is coded in Turbo Pascal 6.0.

REFERENCIAS

- Borland (1990a) *Turbo Pascal 6.0, Programmer's guide*. Borland International, Scotts Valley, CA.
- Borland (1990b) *Turbo Pascal 6.0, Users guide*. Borland International, Scotts Valley, CA.
- Hambleton R.K. y Swaminathan H. (1985) *Item response theory. Principles and applications*. Boston, M.A., Kluwer-Nijhoff Pub.
- Herrando, S. (1989) Tests adaptativos computerizados: Una sencilla solución al problema de la estimación con puntuaciones perfecta y cero. *II Conferencia española de Biometría*. Biometric Society, Segovia.
- Muñiz, J. (1990) *Teoría de respuesta a los ítems, un nuevo enfoque en la evaluación psicológica y educativa*. Madrid, Pirámide.
- Wainer H. (1990) *Computerized adaptive testing: a primer*. Hillsdale, New Jersey, LEA.

(Revisión aceptada: 21/9/93)