Universidad Nacional de Educación a Distancia
Escuela Técnica Superior de Informática
Máster en Ingeniería y Ciencia de Datos

Trabajo Fin de Máster

Utilización de técnicas multivariantes para el estudio del aprendizaje de la mejora de la accesibilidad en el subtitulado de vídeos

Autor: Javier Pérez Arteaga
Directores: Emilio Letón Molina
Jorge Pérez Martín
Fecha de realización: 2023-10-03

This document is reproducible thanks to:

- LaTeXand its class memoir (http://www.ctan.org/pkg/memoir).
- R (http://www.r-project.org/) and RStudio (http://www.rstudio.com/)
- bookdown (http://bookdown.org/) and memoiR (https://ericmarcon.github.io/memoiR/)

Name of the owner of the logo

http://www.company.com

Resumen

TODO: Incluir un resumen del trabajo.

Agradecimientos

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed malesuada nulla augue, ac facilisis risus pretium a. Ut bibendum risus id ex fermentum, at accumsan erat vulputate. In hac habitasse platea dictumst. Sed lobortis est a enim bibendum, ac pulvinar nulla aliquam. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Pellentesque efficitur justo id suscipit pretium. Proin iaculis sit amet nibh vel euismod. Aenean tincidunt faucibus ex, non vehicula ipsum tristique in. Fusce vel tincidunt lectus, vel rutrum nisi. Suspendisse malesuada lectus ac enim vehicula rhoncus. Nullam convallis justo in bibendum eleifend.

Phasellus vitae magna nec mi sagittis luctus vitae eu augue. Donec scelerisque laoreet arcu, eget tempor mi ultricies vel. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum at blandit ex. Vestibulum eu sagittis mauris. In hac habitasse platea dictumst. Duis eget ante vel lacus sollicitudin convallis quis eu velit. Sed auctor sem non nisi hendrerit, vel tincidunt tortor bibendum.

Índice

Resumen iii

Agradecimientos v

Índice vi

Índice de cuadros vii

Índice de figuras ix

Glosario xi

1 Introducción 1
- 1.1 Motivación . . . 1
- 1.2 Propuesta y objetivo . . . 1
- 1.3 Estructura del documento . . . 1

2 Estado del arte 3

3 Materiales y métodos 5

4 Métodos. 7
- 4.1 Fuente de datos. . . . 7
- 4.2 Características del diseño del experimento. . . . 9
- 4.3 Objetivo. . . . 10
- 4.4 Preprocesamiento. . . . 10

5 Modelo. 13
- 5.1 Variables del modelo. . . . 13

6 Exploración inicial. 17
- 6.1 Análisis de la calidad de los datos. . . . 17
- 6.2 Comparación de los tratamientos A y B entre grupos. . . . 20
- 6.3 Análisis de las preguntas. . . . 24

7 Análisis estadístico. 27
- 7.1 Agrupamientos de preguntas. . . . 27
- 7.2 Análisis de tablas de contingencia. . . . 31

8 Modelado estadístico. **37**
8.1 Árboles de inferencia condicional. 37
8.2 Regresión ordinal. 39
8.3 Modelado Bayesiano. 59

9 Resultados **65**

10 Conclusiones y trabajo futuro **67**

Referencias **69**

Apéndices 71

A Preprocesado de los ficheros suministrados. **71**

B Creación de los `dataframes` `df_all` y `df_clean`. **77**

C Efecto secuencia e interacción tratamiento vs. periodo. **81**
C.1 Preparación. 81
C.2 Análisis con un solo factor (tratamiento). 82
C.3 Análisis con un dos factores (tratamiento y periodo). 84
C.4 Factor secuencia. 85

Índice de cuadros

4.1 Niveles de los items de la escala de Likert. 8
4.2 Items de la escala de Likert. 8

5.1 Descripción de las variables más importantes 13
5.2 Muestra del dataframe preparado para el modelado estadístico en formato largo. 15

6.1 Tiempos de realización de la segunda actividad de duración inferior a 2 minutos. 18
6.2 Test en los que todas las preguntas se contestan el mismo valor de respuesta. 19
6.3 Los 5 test con más respuestas 'No sé/No contesto' 20
6.4 Tablas de contingencia de la información socioeconómica de los estudiantes. 21
6.9 Estudiantes que tienen diferencias en sus respuestas muy alejadas de la tendencia de su grupo. 21
6.10 Resumen de frecuencias de respuesta. 22

7.1 Valor del coeficiente alpha de Cronbach si se elimina una pregunta. . 28
7.2 Relación de cada pregunta con el índice alpha de Cronbach. 28
7.3 Tabla de contingencia de preguntas y respuestas. 29

8.1 Probabilidades de respuesta para el modelo ordinal Response ~ Treat 45
8.2 Intercepto y pendiente de Question en el modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question) 58
8.3 Comparación frecuentista/bayesiano de coeficientes estimados en el modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question). 60
8.4 Distribuciones a priori del modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question). 61

C.1 Ajuste del modelo Response ~ Treat con contrasts treatment. 82
C.2 Ajuste del modelo Response ~ Treat * Period con contrasts treatment. 84
C.3 Ajuste del modelo Response ~ Treat + Period + Seq con contrasts treatment. 85
C.4 Ajuste del modelo Response ~ Treat + Period + Seq con contrasts sum. 86

C.5 Ajuste del modelo Response ~ Treat * Period con contrasts sum. . . 86

Índice de figuras

6.1 Estudiantes asignados a cada grupo. 17
6.2 Número de respuestas diferentes en un mismo test. 18
6.3 Número de respuestas diferentes entre los test para cada estudiante. . 19
6.4 Frecuencias absolutas de las diferencias en las respuestas entre test por estudiante y grupo. 22
6.5 Frecuencias relativas de las respuestas al test. 23
6.6 Frecuencias relativas de las respuestas por pregunta. 25
6.7 Preguntas ordenadas por valoración. 26

7.1 Dendograma de aglomeramiento jerárquico de preguntas en función de la tabla de contingencia de respuestas. 30
7.2 Mosaico de tratamientos y secuencias. 32
7.3 OR entre tratamiento y grupo por nivel de respuesta. 33
7.4 OR entre tratamiento y periodo por nivel de respuesta. 35

8.1 Modelo con árboles de inferencia condicional (Response ~ Treat + Cluster + Period + Seq). 38
8.2 Modelo con árboles de inferencia condicional (Level ~ Treat + Cluster + Period + Seq). 39
8.3 Función latente en una regresión ordinal acumulativa. 41
8.4 Probabilidades de respuesta para el modelo ordinal Response ~ Treat * Period 48
8.5 Probabilidades de respuesta para el modelo ordinal no proporcional Response ~ Treat * Period 50
8.6 Probabilidades de respuesta para el modelo ordinal Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question) 59
8.7 Distribuciones a priori del modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question). 62
8.8 MCMC trazado del modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question). 63
8.9 Comparación de los valores reales con los obtenidos a partir de la función predictiva a posteriori del modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question). 64

Capítulo 1

Introducción

1.1 Motivación

1.2 Propuesta y objetivo

1.3 Estructura del documento

Capítulo 2

Estado del arte

Capítulo 3

Materiales y métodos

Capítulo 4

Métodos.

4.1 Fuente de datos.

Los datos proceden de la edición de 2022 del curso MOOC Materiales digitales accesibles de la UNED. Concretamente a los estudiantes matriculados se les propuso que realizaran una actividad voluntaria consistente en evaluar la calidad del subtitulado de dos vídeos. Los vídeos eran idénticos y se diferenciaban únicamente en la calidad del subtitulado. Los subtítulos de uno de los vídeos se realizaron (ver Pérez Martín et al. 2021; Molanes-López et al. 2021) siguiendo las guía Web Content Accessibility Guidelines 2.1 (WCAG 2.1) del W3C (World Wide Web Consortium). El otro vídeo tenía un subtitulado similar pero se introdujeron pequeñas deficiencias inapreciables para alguien que carezca de conocimientos sobre accesibilidad. Los estudiantes fueron clasificados en dos grupos. Al primer grupo se le presentó primero el vídeo correctamente subtitulado y luego el otro. El segundo grupo realizó la actividad cruzada: primero evaluó el vídeo mal subtitulado y luego el bien subtitulado. Tras ver cada uno de los vídeos, los estudiantes tuvieron la oportunidad de valorar la calidad del subtitulado realizando un test en escala de Likert de 18 items y 5 niveles cada item [1]. Los 18 items de Likert pretenden asegurar los criterios de la norma UNE 153010 (ver AENOR 2012).

En la Tabla 4.1 se muestran los 5 niveles de cada uno de los items de la escala de Likert:

[1] Para una descripción sobre cómo se debe realizar una escala de Likert consultar Guerra et al. (2016).

Cuadro 4.1: Niveles de los items de la escala de Likert.

values	levels
0	No sé / No contesto
1	Muy en desacuerdo
2	En desacuerdo
3	Neutral
4	De acuerdo
5	Muy de acuerdo

En la Tabla 4.2 se muestran los 18 items de la escala de Likert que se propuso a los alumnos para que evaluaran cada uno de los vídeos:

Cuadro 4.2: Items de la escala de Likert.

Item	Texto
Q01	La posición de los subtítulos.
Q02	El número de líneas por subtítulo.
Q03	La disposición del texto respecto a la caja donde se muestran los subtítulos.
Q04	El contraste entre los caracteres y el fondo.
Q05	La corrección ortográfica y gramatical.
Q06	La literalidad.
Q07	La identificación de los personajes.
Q08	La asignación de líneas a los personajes en los diálogos.
Q09	La descripción de efectos sonoros.
Q10	La sincronización de las entradas y salidas de los subtítulos.
Q11	La velocidad de exposición de los subtítulos.
Q12	El máximo número de caracteres por línea.
Q13	La legibilidad de la tipografía.
Q14	La separación en líneas diferentes de sintagmas nominales, verbales y preposicionales.
Q15	La utilización de puntos suspensivos.
Q16	La escritura de los números.
Q17	Las incorrecciones en el habla.
Q18	Los subtítulos del vídeo cumplen en general con los requisitos de accesibilidad.

Los datos personales de los estudiantes se suministraron anonimizados para evitar ninguna referencia a su identidad. Del estudio se han eliminado a aquellos estudiantes que, a pesar de haber realizado la actividad, no dieron su autorización para que sus datos se utilizaran en un estudio científicos.

Se dispuso de los siguientes ficheros `csv`:

- El fichero `grade` contiene el identificador de estudiante y el grupo al que pertenece (campo `cohort`).
- El fichero `abo` es la información socioeconómica que voluntariamente ha aportado el estudiante: sexo, año nacimiento, nivel de estudios, ocupación .
- El fichero `conoc` contiene el test de evaluación inicial de conocimientos del estudiante.
- El fichero `exp` es la evaluación del curso realizada por cada estudiante.
- El fichero `acc` contiene la información sobre accesibilidad que utiliza el estudiante.
- Los ficheros `test1` y `test2` son las repuestas al test de Likert sobre la calidad del subtitulado del primer y del segundo vídeo realizado por cada grupo respectivamente.

4.2 Características del diseño del experimento.

El diseño del experimento es completamente aleatorizado, de respuesta ordinal, cruzado AB/BA y doble ciego. Es decir que la asignación de los estudiantes a cada grupo fue aleatoria; cada grupo vio los vídeos en orden inverso; los estudiantes no conocían a priori qué vídeo estaban viendo en cada momento y tampoco se disponía de esta información en el momento de realizar el análisis estadístico de los datos.

Un diseño completamente aleatorizado (Lawson 2015, pp. 18) «garantiza la validez del experimento contra sesgos causados por otras variables ocultas. Cuando las unidades experimentales se asignan aleatoriamente a los niveles de factor de tratamiento, se puede realizar una prueba exacta de la hipótesis de que el efecto del tratamiento es cero utilizando una prueba de aleatorización».

Siguiendo a Senn (2022), para que el ensayo sea de tipo cruzado no sería suficiente intercambiar las secuencias sino que debe ser objeto del ensayo el estudio de las diferencias entre los tratamientos individuales que componen las secuencias. Los principales problemas d eun diseño cruzado son el abandono, `drop-out`, de alguno de los participantes y la interacción entre el tratamiento y el periodo o `carry-over`. Además el análisis estadístico es más complicado y particularmente cuando la respuesta es ordinal y hay más de dos tratamientos. En la misma línea, Lui (2016) afirma que «el objetivo principal de un diseño cruzado es estudiar la diferencia entre tratamientos individuales (en lugar de la diferencia entre secuencias de tratamiento). Debido a que cada paciente sirve como su propio control, el diseño cruzado es una alternativa útil al diseño de grupos paralelos para aumentar la potencia».

Las respuestas a un test de Likert se realizan en escala ordinal. No es adecuado realizar operaciones aritméticas para calcular medias con este tipo de datos. Pero ellos los test estadísticos para analizar el efecto de un tratamiento con respuesta

continua como son *ANOVA* y *t*-test no son adecuados con datos ordinales. Según la investigación de Liddell y Kruschke (2018) ajustar datos ordinales con modelos cuantitativos puede producir los siguientes problemas:

- Se pueden encontrar diferencias significativas entre grupos cuando no las hay: Error tipo I.
- Se pueden obviar diferencias cuando en realidad sí existen: Error tipo II.
- Incluso se pueden invertir los efectos de un tratamiento.
- También puede malinterpretarse la interacción entre factores.

Una opción es tratar los datos ordinales como si se tratara de datos categóricos y utilizar técnicas no paramétricas como el test de $Kruskal - Wallis$. El problema de este tipo de técnicas es que ignoran que los datos tienen una escala y, en el caso particular del diseño que nos ocupa se trata de datos longitudinales, es decir, que se toman varias medidas de cada sujeto y, por lo tanto, los datos no son independientes. Agresti (2010) expone un catálogo de técnicas para analizar datos categóricos y ordinales.

4.3 Objetivo.

El objetivo del estudio es responder a la pregunta de investigación:

Son los estudiantes de un curso de accesibilidad capaces de encontrar los errores en el subtitulado de un vídeo. Para ello se propondrán diversos test y modelos estadísticos que tengan en consideración las características que se han comentado en el diseño del experimento (ver Sección 4.2). Particularmente se tendrá en cuenta que se trata de un diseño cruzado con variable respuesta ordinal y variables explicativas longitudinales.

4.4 Preprocesamiento.

Partiendo de los ficheros suministrados (ver Sección 4.2), se realiza el siguiente preprocesado (para ver el código ejecutado consultar Apéndice A):

- Se lee el fichero de perfil del usuario. El número de fila con el que el usuario aparece en el fichero se utilizará como identificador del usuario para mantener la trazabilidad y comprobar que las transformaciones realizadas son correctas.
- Se eliminan del estudio a los estudiantes que aún habiendo realizado la actividad, no han dado su consentimiento para participar en el estudio.
- El valor del campo `cohort` se sustituye por una letra *A* o *B* en función del grupo asignado. En este momento se desconoce qué vídeo vio primero cada grupo.

- Se lee el fichero `profile` y se añade a los usuarios información sobre el sexo, el año de nacimiento y el novel de estudios.
- Se lee el fichero `conoc` y se calcula cuántas preguntas acertó cada usuario en el test de evaluación de conocimientos previos. Se añade esta información al perfil del usuario.
- Se leen los ficheros de test y se procesan. Se utiliza el nombre del fichero (`test1` o `test2`) para saber de qué vídeo se está respondiendo el test [2].
- Se seleccionan las preguntas que contienen las respuestas y se renombran para que sea más fácil saber de qué pregunta se trata [3]. Se convierte el campo `LastTry`, que contiene la fecha y hora de realización del test, a formato fecha y hora.
- Se realizan algunas comprobaciones como la ausencia de valores nulos en la variables más relevantes o que no existan inconsistencias ni errores de procesado.
- Se eliminan los comentarios y se graban en fichero aparte para que no revelen información que podría descubrir el tipo de subtitulado que piensa que está evaluando el estudiante.
- Se almacenan los resultados de los test preprocesado en un fichero `csv`.

[2] Se reitera que en este momento se desconoce si el vídeo es el correctamente subtitulado o el otro. La única información que se almacena es si se está respondiendo al vídeo que se voy primero o al que se vio después.

[3] En los fichero suministrados pla respuesta a cada pregunta ocupa varios campos y se selecciona en cada pregunta el que contiene el valor de la respuesta y se convierte a numérico.

Capítulo 5

Modelo.

5.1 Variables del modelo.

En la Tabla 5.1 se describen las características más relevantes de las principales variables que se utilizarán en en modelado y en el análisis estadístico.

Cuadro 5.1: Descripción de las variables más importantes

Nombre	Descripción	Tipo	Valores
Response	Respuesta a las preguntas del test.	Factor ordenado	De 0 a 5[1]
Level	Valoración de la respuesta.	Factor ordenado	Negative, Neutral, Positive[2]
Treat	Subtítulos	Factor	A o B[3]
Period	Periodo	Factor	1 ó 2[4]
Seq	Secuencia de aplicación de los tratamientos.	Factor	AB o BA
Subject	Identificación del estudiante	Factor	Numérico
Question	Número de la pregunta	Factor	Q01, Q02, ..., Q18[5]
Cluster	Grupo de la pregunta	Factor	1, 2, ó 3[6]

[1] Se ha hecho una rotación sobre los valores originales. 0 = No sé, 1 = Muy en desacuerdo, ..., 5 Muy de acuerdo.
[2] Positive cuando Response sea 4 ó 5, Negative cuando sea 1 ó 2 y Neutral para 3.
[3] No se conoce si el tratamiento A es el subtitulado bueno o lo es el B.
[4] 1 para el primer vídeo visto y 2 el segundo.
[5] Se ha reorganizado de tal forma que Q18, que es la pregunta resumen, sea el valor primero y de referencia.
[6] Se aplicará una técnica estadística de agrupamiento para agregar las preguntas.

Partiendo del `dataframe` que se construyó en el preprocesado (ver Sección 4.4) construimos el `dataframe` que usaremos a partir de este momento. Las operaciones principales que se han realizado han sido:

- Renombrar las variables para que se correspondan con las de nuestro modelo (ver Tabla 5.1).

- Eliminar del estudio los usuarios que solo han realizado uno de los test como se explica en Sección 6.1.
- Transformar las variables que lo requieran en factores. La pregunta 18 se usará como referencia en el factor `Question`.
- Rotar los valores de respuesta para que «No sé / No contesto» tenga valor 0 y el resto de 1 a 5 desde «Muy en desacuerdo», 1, hasta «Muy de acuerdo», 5.
- Agrupar las preguntas por similitud de respuesta (ver Sección 7.1).
- Crear el factor `Level` con los niveles `positive`, `neutral` y `negative` dependiendo de si la respuesta es 4 ó 5, 3, 1 ó 2 respectivamente.
- Transformar el `dataframe` de formato ancho a largo: los ficheros de respuestas se suministran en formato ancho. Es decir, que cada fila es un test que contiene 18 columnas para las respuestas a cada pregunta. Los nombres de las columnas son $Q01$, $Q02$, …, $Q18$ y tendrán valores de 0 a 6 con las respuestas. La mayoría de los paquetes de R que vamos a usar requieren que los datos estén en formato largo. Esto que quiere decir que cada fila tendrá una única respuesta por lo que habrá únicamente dos columnas, *Question* y *Response*. En la primera se almacenará el identificador de la pregunta ($Q01$, $Q02$, …, $Q18$) y en la segunda el valor de la respuesta (de 0 a 6). De esta forma un test pasará de ocupar una fila y 18 columnas en el formato ancho a 18 filas y dos columnas en el largo.

En Apéndice B se puede consultar el código en R para realizar el proceso descrito anteriormente. Con estas transformaciones se crean los dos `dataframes` que se usarán en el análisis estadístico de los datos:

- `df_all` contiene en formato largo todas las respuestas a los test.
- `df_clean` tiene la misma estructura que `df_all` pero en él se han eliminado las respuestas «No sé / No contesto».

`df_all` se utilizará cuando se traten las respuestas como categóricas y, por lo tanto, como no ordenadas. `df_clean` se utilizará cuando se traten las respuestas como ordenadas y por ello no contiene las respuestas con valor «No sé / No contesto».

La estructura de estos `dataframes` es la siguiente:

```
tibble [2,980 x 8] (S3: tbl_df/tbl/data.frame)
 $ Seq     : Factor w/ 2 levels "AB","BA": 1 1 1 1 1 1 1 1 1 1 ...
 $ Period  : Factor w/ 2 levels "1","2": 1 1 1 1 1 1 1 1 1 1 ...
 $ Treat   : Factor w/ 2 levels "A","B": 1 1 1 1 1 1 1 1 1 1 ...
 $ Subject : Factor w/ 87 levels "4","33","35",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Question: Factor w/ 18 levels "Q18","Q01","Q02",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ Cluster : Factor w/ 3 levels "1","2","3": 1 2 2 2 2 1 1 1 1 1 ...
 $ Response: Ord.factor w/ 5 levels "1"<"2"<"3"<"4"<..: 3 3 3 3 3 3 3 3 3 3 ...
 $ Level   : Ord.factor w/ 3 levels "Negative"<"Neutral"<..: 2 2 2 2 2 2 2 2 2..
```

En el Tabla 5.2 se muestran algunos ejemplos de estos `dataframes`.

Cuadro 5.2: Muestra del dataframe preparado para el modelado estadístico en formato largo.

Seq	Period	Treat	Subject	Question	Cluster	Response	Level
BA	1	B	229	Q17	3	4	Positive
BA	2	A	1023	Q18	1	4	Positive
BA	1	B	765	Q15	3	2	Negative
BA	2	A	229	Q13	2	4	Positive
BA	2	A	320	Q11	2	4	Positive
AB	1	A	75	Q12	2	4	Positive
BA	1	B	220	Q03	2	3	Neutral
AB	1	A	1153	Q07	1	4	Positive
AB	1	A	1011	Q12	2	4	Positive
BA	2	A	535	Q01	2	4	Positive

Capítulo 6

Exploración inicial.

6.1 Análisis de la calidad de los datos.

Respuestas a los test.

Como se explica en la Tabla 5.1, al subtitulado le denominamos tratamiento y a sus niveles (correcto e incorrecto) los hemos llamado *A* y *B* sin hacer ninguna conjetura de cual de los dos es el subtitulado correcto. El grupo con secuencia *AB* será el que primero vio el vídeo con subtitulado *A* y luego el *B*. Análogamente, el grupo con secuencia *BA* vio los vídeos en orden inverso. Recuérdese que el nivel 0 de respuesta se corresponde con «No sé / No contesto» (ver Tabla 4.1).

Hay 24 estudiantes que no realizaron el segundo test. De ellos 9 pertenecen al grupo AB y 15 al grupo BA. Debido a que no son muchos y a que los grupos se mantienen balanceados, se ha decidido eliminar los test de estos estudiantes.

Tras eliminar a los estudiantes que no realizaron uno de los test, constatamos (ver Figura 6.1) que los grupos están balanceados en el número de estudiantes y que disponemos de suficientes datos para realizar el análisis estadístico.

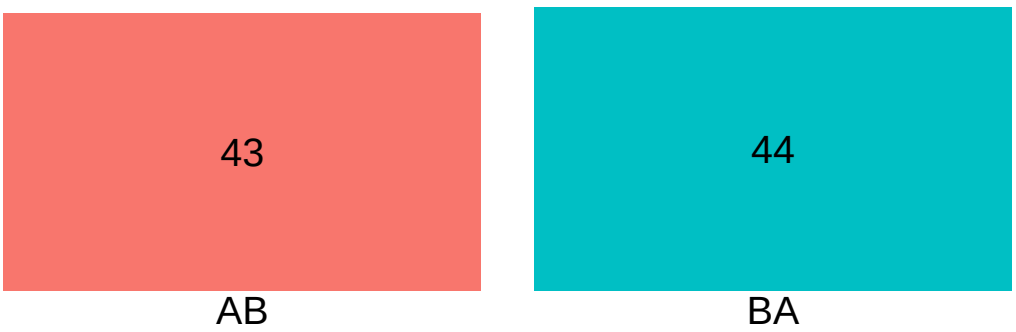


Figura 6.1: Estudiantes asignados a cada grupo.

El campo `LastTry` contiene la fecha y hora de realización del test. Con esta información podemos conocer el tiempo que empleó cada estudiante entre subtitulados. La Tabla 6.1 muestra que hay algunos test que se hicieron demasiado rápido [1].

Cuadro 6.1: Tiempos de realización de la segunda actividad de duración inferior a 2 minutos.

Minutes
0.93
1.3
1.7
1.72
1.78
1.97

La Figura 6.2 muestra que hay 28 test en los que el estudiante contestó a todas las preguntas usando únicamente 2 respuestas diferentes. Además hay 13 test en los que se contestaron todas las preguntas con 1 respuesta.

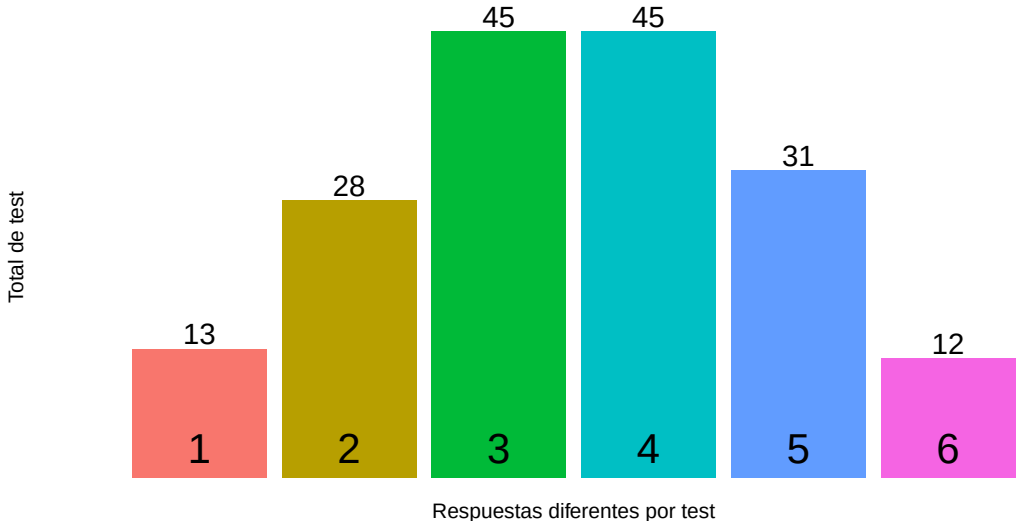


Figura 6.2: Número de respuestas diferentes en un mismo test.

[1] Hay que tener en cuenta que la duración de vídeo es de algo más de 40 segundos y que los estudiantes tienen que contestar un test de 18 preguntas.

La tabla Tabla 6.2 muestra la respuesta utilizada, el grupo y el periodo de los test con respuesta única.

Cuadro 6.2: Test en los que todas las preguntas se contestan el mismo valor de respuesta.

Response	Seq	Test
2	AB	01
2	AB	02
3	BA	01
3	BA	02
3	BA	02
3	BA	02
4	AB	01
4	AB	01
4	AB	02
4	BA	01
4	BA	02
4	BA	02
4	BA	02

La Figura 6.3 presenta la distribución de la cantidad de respuestas cuyo valor cambia entre los dos test que realiza cada estudiante.

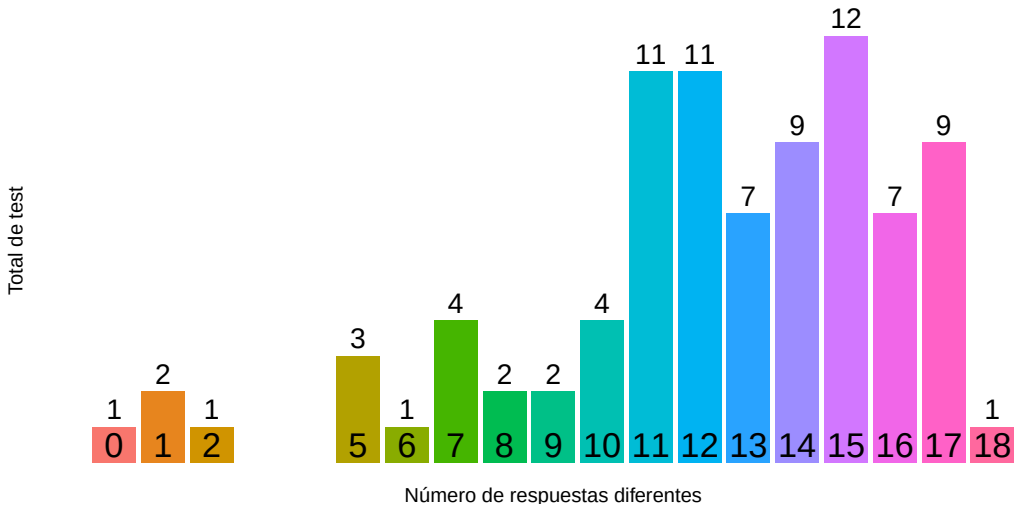


Figura 6.3: Número de respuestas diferentes entre los test para cada estudiante.

Tan solo 1 estudiante respondió a todas las preguntas con el mismo valor en los dos test. Por otro lado, no hay test que tengan un número excesivo de contestaciones «No sé/No contesto» (ver Tabla 6.3).

Cuadro 6.3: Los 5 test con más respuestas 'No sé/No contesto'

Test	Total respuesta por test
01	5
01	5
02	5
02	5
01	4

Conclusiones.

No parece razonable realizar la actividad en menos de 2 minutos. Se observa que en algunos test hay poca variabilidad. Sin embargo, no son muchos los test con estas características así que se ha decidido mantener estos datos a pesar de que se pueda dubar de si en ellos los estudiantes contestaron con la debida atención y diligencia.

Valores nulos o erróneos.

En los test no se ha detectado ningún valor nulo ni erróneo. Sin embargo tenemos algunos de estos valores en la información socioeconómica de los estudiantes (ver Tabla 7.3).

6.2 Comparación de los tratamientos A y B entre grupos.

La Figura 6.4 presenta una forma de comparar los dos test que realizados por los estudiantes. Para cada estudiante se comparó pregunta a pregunta sus dos test y se contabilizó la diferencia entre el número de preguntas en que la puntuación en el segundo vídeo fue superior y en las que lo fue inferior (las que no variaron de puntuación no se consideraron). En el eje x se muestra la diferencia entre preguntas. Cantidades negativas indican que hay más respuestas en el segundo de los test que han empeorado respecto al primero de las que han mejorado. En el eje y se representa el número de estudiantes para cada diferencia. Esta frecuencia se representa en negativo cuando la diferencia es negativa [2]. Esto es una forma de evaluar si el estudiante valoró mejor o no el segundo vídeo que el primero.

Vemos que en el grupo AB las diferencias tienden a ser negativas y en el BA positivas. Esto estaría indicando que los estudiantes valoran mejor el subtitulado

[2]En la comparación se han omitido aquellas preguntas en las que el estudiante contestó «No sé/No contesto» en la pregunta correspondiente de uno de los test.

Cuadro 6.4: Tablas de contingencia de la información socioeconómica de los estudiantes.

gender	Freq
f	92
m	38
NA	44

Estudiantes por sexo.

year_of_birth	Freq
None	44
NA	2

Estudiantes con valor nulo en el campo año de nacimiento.

level_of_education	Freq
a	50
b	16
hs	4
m	30
other	4
p	20
NA	50

Estudiantes por nivel educativo.

level_of_knowledge	Freq
4	2
6	4
7	30
8	44
9	40
10	32
NA	22

Estudiantes en función del número de preguntas acertadas en el test de conocimiento.

de nivel *A*. Por ello es esperable que las respuestas de los estudiantes del grupo *AB* hayan empeorado y que las diferencias sean negativas y que lo contrario haya sucedido con las del grupo *BA*. La diferencia más frecuente en el grupo *AB* es 12 y en el grupo *BA* este valor es 11.

Resulta llamativo que haya estudiantes cuyas contestaciones estén tan alejadas de la tendencia de su grupo. En la Tabla 6.9 se muestran los tiempos que han transcurrido entre la realización de los test de aquellos estudiantes cuyas respuestas difieren de forma importante de su grupo. Se observa que casi todos son tiempos entre actividades muy cortos. En cualquier caso y, como no son muchos, se ha decidido no eliminarlos y realizar el análisis con ellos.

Cuadro 6.9: Estudiantes que tienen diferencias en sus respuestas muy alejadas de la tendencia de su grupo.

Seq	Diff	Minutes
AB	17	1.3
AB	7	3.33
BA	-10	50345.95
BA	-12	1.7

En la Figura 6.5 representamos la frecuencia relativa del valor de respuesta para

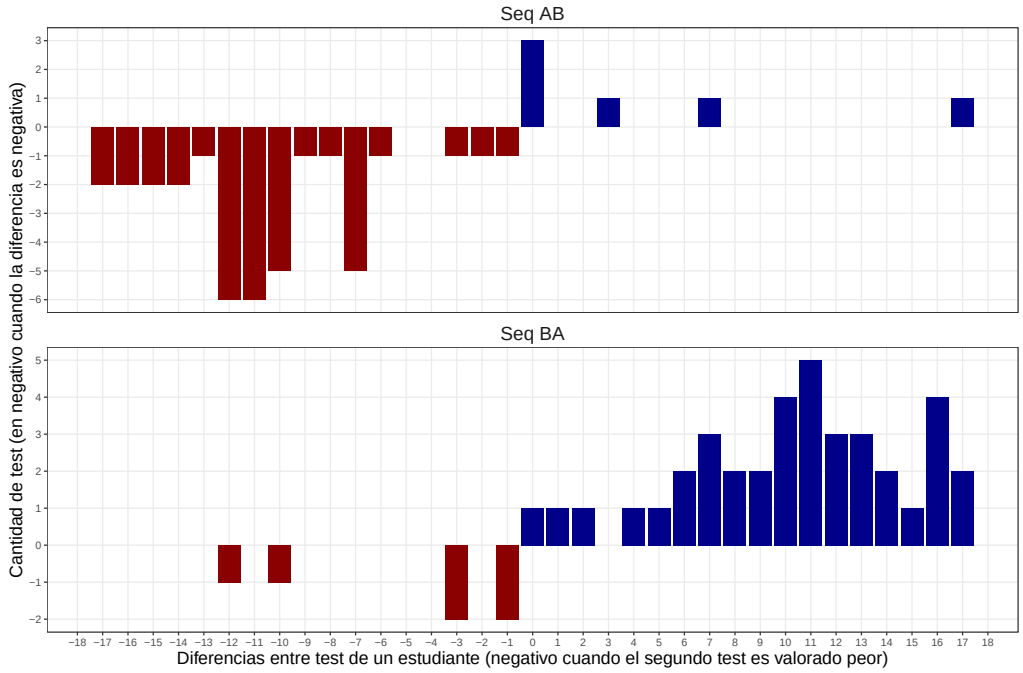


Figura 6.4: Frecuencias absolutas de las diferencias en las respuestas entre test por estudiante y grupo.

cada grupo y test en todas la preguntas [3]. Esta es otra forma de comparar los niveles de subtitulado.

Cuadro 6.10: Resumen de frecuencias de respuesta.

				Response				
Seq	Period	Treat	0	1	2	3	4	5
AB	1	A	39	2	25	71	203	434
AB	2	B	43	87	185	121	172	166
BA	1	B	40	76	174	127	237	138
BA	2	A	30	2	30	64	345	321

La Figura 6.5 muestra algunas cuestiones interesantes:

- El tratamiento (subtitulado) con nivel *A* presenta claramente mayores valores de respuesta que el *B* como ya habíamos visto (ver Figura 6.4). Si en este momento tuviéramos que decidir qué subtitulado es cada uno parece claro que sería el de nivel *A*. No obstante, ni en el análisis exploratorio ni en el modelado estadístico se hará ninguna suposición.

- En general los dos grupos muestran bastante acuerdo en el subtitulado en ambos niveles: En el nivel de tratamiento *A* los dos grupos tienen una frecuencia relativa similar de respuestas positivas (valores 4 y 5). El grupo *AB* tiene un 82% de respuestas positivas frente a un 84% el grupo *BA*. No

[3] En el Tabla 6.10 se presenta la misma información con los valores absolutos.

obstante, el grupo *AB* tiene más respuestas con valor 5 que el grupo *BA* (56% frente a 41%). La valoración es también similar entre grupos en el nivel de tratamiento *B*: el grupo *AB* tiene 44% de respuestas positivas y 47% el grupo *BA*. Las valoraciones negativas (1, 2), la neutra (3) y la "No sé / No contesto" (0) son también muy similares.

- Las respuestas son similares entre periodos aunque ligeramente más negativas en el segundo. Así un 65% de las respuestas son positivas en el primer periodo frente a un 64% en el segundo.

Figura 6.5: Frecuencias relativas de las respuestas al test.

El análisis marginalizado de tratamiento, secuencia y periodo tiene estos resultados referidos a las preguntas con contestación positiva (4, 5):

- El tratamiento *A* tiene un 83% marginalizado de respuestas positivas frente al 46% del tratamiento *B*.
- El periodo 1 tiene un 65% marginalizado de respuestas positivas frente al 64% del periodo 2.
- Finalmente, la secuencia *AB* tiene un 63% de respuestas positivas frente 66% de la secuencia *BA*. Analizado por respuestas individuales, la respuesta 4 pasa de 24% en la secuencia *AB* a 37% en la *BA* y, de forma contraria,

en la respuesta 5 pasa de 39% en *AB* a 29% en *BA*. En las respuestas negativas y no contestadas y neutra no se aprecian estas variaciones.

6.3 Análisis de las preguntas.

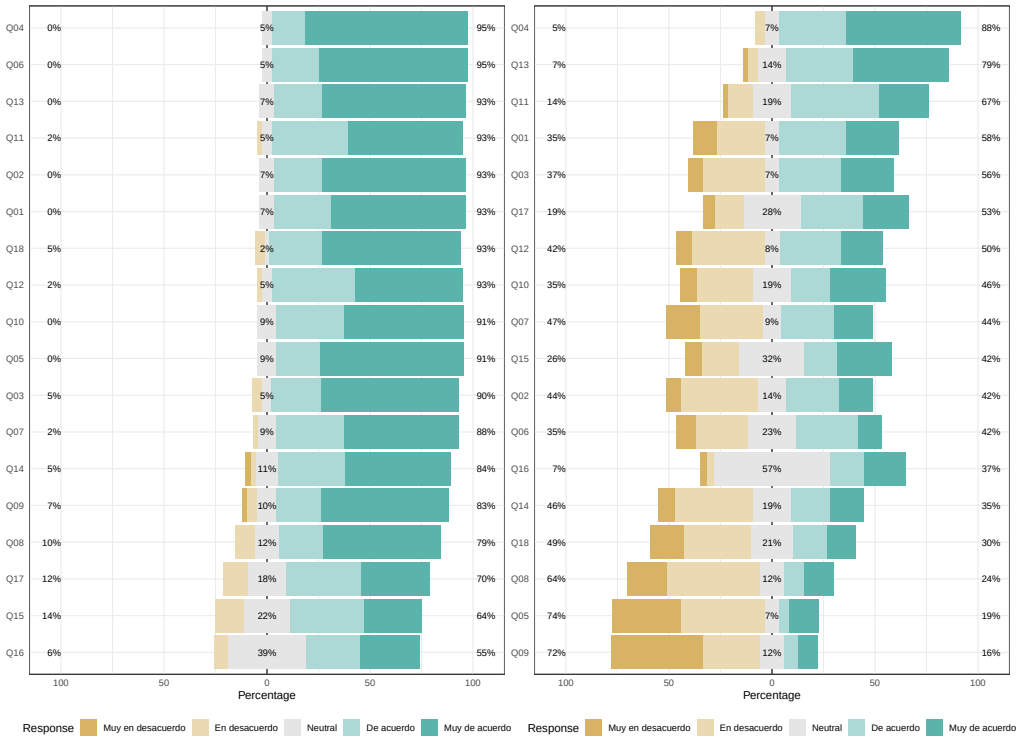
El gráfico Figura 6.6 muestra la frecuencia relativa por grupo y por test de las preguntas clasificadas por niveles de respuesta, considerando que:

- Los niveles 1 y 2 se consideran valoraciones negativas.
- El nivel 3 se considera neutro.
- Los niveles 4 y 5 se consideran positivos.
- El nivel 0 («No sé / No contesto») se excluye en este análisis.

Se muestra en primer lugar la pregunta 18 por ser una valoración global del subtitulado y que resume la opinión que sobre el mismo tiene el estudiante. Volvemos a constatar que el subtitulado *A* es mejor valorado por los estudiantes, pero ahora vemos que en las 18 preguntas ambos grupos tienen mas puntuaciones positivas y menos negativas en el subtitulado *A* que el *B*. También volvemos a encontrar que los dos grupos valoran de forma muy similar los dos niveles de subtitulado en todas la preguntas. En el nivel de subtitulado *A* las preguntas $Q15$, $Q16$ y $Q17$ obtienen relativamente peores valoraciones (consultar la Tabla 4.2 para ver los valores) y estas son similares en ambos subtitulados. Hay algunas preguntas que son valoradas de forma positiva incluso en el nivel de subtitulado *B* (por ejemplo $Q04$ o $Q13$) y que, por lo tanto, su valoración es similar en ambos subtitulados. Por último, las preguntas $Q05$ y $Q09$ (también la $Q14$ pero solo para el grupo *BA*) tienen una valoración muy negativa en el nivel de subtitulado *B*.

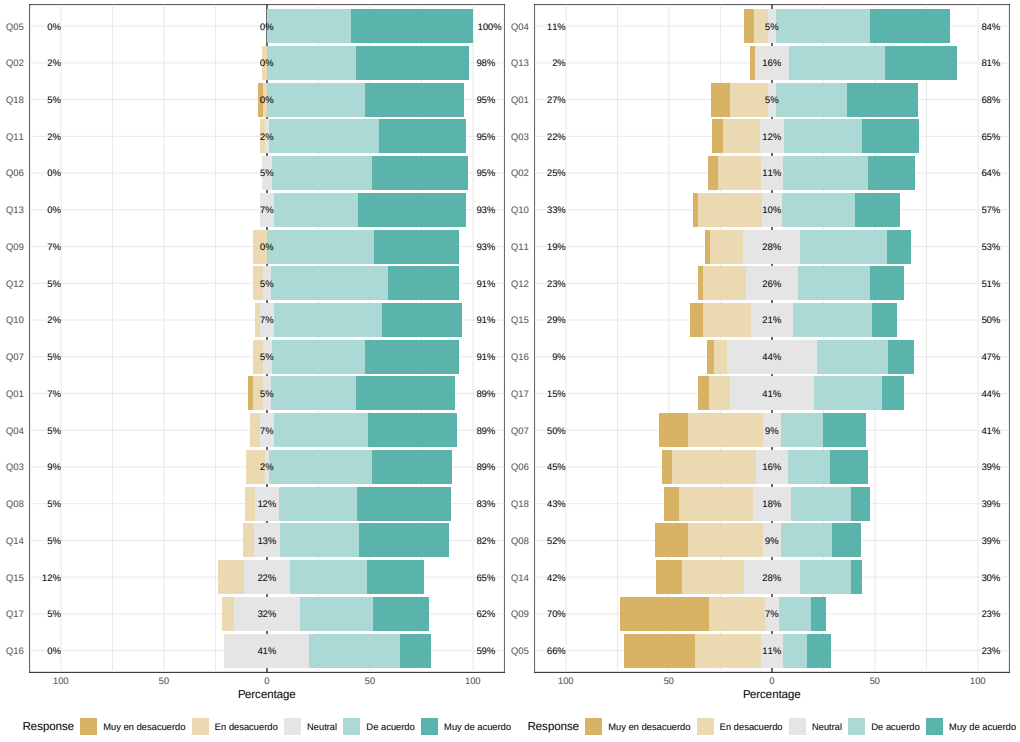
La figura Figura 6.7 clasifica la preguntas por valoración y permite constatar lo que ya habíamos visto en el párrafo anterior con mayor comodidad.

Figura 6.6: Frecuencias relativas de las respuestas por pregunta.



(a) Seq AB , Treat A

(b) Seq AB , Treat B



(c) Seq BA , Treat A

(d) Seq BA , Treat B

Figura 6.7: Preguntas ordenadas por valoración.

Capítulo 7

Análisis estadístico.

7.1 Agrupamientos de preguntas.

Correlación entre preguntas con el alfa de Cronbach.

Normalmente las preguntas de un cuestionario pretenden medir una variable que está oculta o latente. En nuestro caso es la calidad del subtitulado. Las respuestas a estas preguntas relacionadas deben ser consistentes internamente, es decir, las respuestas deben correlacionarse fuerte y positivamente.

Un índice que se utiliza habitualmente para medir la consistencia interna de un cuestionario es el coeficiente `alfa de Cronbach`, ver Schweinberger (2020). Se define de esta forma:

$$\alpha = \frac{N}{N-1}\left(1 - \frac{\sum_{i=1}^{N} s_i^2}{s^2}\right) \tag{7.1}$$

Donde:

- α es el coeficiente `alfa de Cronbach`.
- N es el número de items de la escala de Likert.
- s_i^2 es la varianza de la puntuación del item i.
- s^2 es la varianza total de las puntuaciones de todos los items.

Valores cercanos 1 indican una fuerte correlación en las respuestas y se admite que las preguntas del cuestionario están midiendo la misma variable latente.

Para calcular en R este coeficiente podemos usar la función `alpha` del paquete `psych`:

Cuadro 7.1: Valor del coeficiente alpha de Cronbach si se elimina una pregunta.

(a)

Q18	Q01	Q02	Q03	Q04	Q05	Q06	Q07	Q08
0.91	0.92	0.92	0.92	0.92	0.91	0.91	0.91	0.91

(b)

Q09	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17
0.91	0.91	0.92	0.91	0.92	0.92	0.92	0.93	0.92

Cuadro 7.2: Relación de cada pregunta con el índice alpha de Cronbach.

(a)

Q18	Q05	Q06	Q09	Q08	Q07	Q10	Q12	Q02
0.86	0.81	0.79	0.79	0.77	0.75	0.73	0.72	0.71

(b)

Q03	Q14	Q01	Q11	Q15	Q13	Q04	Q16	Q17
0.68	0.66	0.65	0.64	0.56	0.51	0.46	0.44	0.42

```
alpha <- df_all %>%
    pivot_wider(
        names_from = Question,
        values_from = Response_v,
        id_cols = c(Treat, Subject)
    ) %>%
    dplyr::select(-c(Treat, Subject)) %>%
    psych::alpha()
```

Se obtiene un coeficiente alfa de `alfa de Cronbach` de 0.92 que indica una muy buena correlación entre las respuestas a todas las preguntas. Este valor apenas se ve alterado si se elimina una de las preguntas (ver Tabla 7.1).

En la Tabla 7.2 mostramos las preguntas que más contribuyen al índice `alpha de Cronbach`. Es interesante que la pregunta $Q18$, que es la valoración general del cuestionario, sea la que mejor contribución tiene al índice.

Agrupamiento jerárquico aglomerativo.

En en la Sección 7.1 y en la Sección 6.2 hemos visto que algunas de las preguntas tienen respuestas similares a otras pero diferentes del resto. Puede ser interesante aplicar una técnica de agrupamiento que nos permita crear grupos de preguntas que podremos analizar por separado.

Vamos a realizar una agrupación jerárquica aglomerativa de las preguntas en función de la tabla de contingencia de las respuestas utilizando la distancia euclidea como medida de distancia y el método de aglomeración de enlace completo para unir conglomerados [1]. Para ello primero calculamos la tabla de contingencia (ver Tabla 7.3) de preguntas y respuestas.

```
table <- df_all %>%
    xtabs(~ Question + Response, data = .)
```

Cuadro 7.3: Tabla de contingencia de preguntas y respuestas.

Question	Response_0	Response_1	Response_2	Response_3	Response_4	Response_5
Q18	0	11	33	18	52	60
Q01	0	10	20	10	59	75
Q02	0	5	26	14	58	71
Q03	5	5	26	11	60	67
Q04	0	2	7	10	61	94
Q05	1	29	31	12	34	67
Q06	1	6	29	21	53	64
Q07	0	13	32	14	54	61
Q08	4	15	41	19	40	55
Q09	1	39	29	12	42	51
Q10	8	4	24	18	59	61
Q11	3	2	14	23	75	57
Q12	5	4	26	18	69	52
Q13	1	2	2	19	62	88
Q14	21	9	29	27	44	44
Q15	26	5	25	36	47	35
Q16	47	2	5	57	39	24
Q17	29	4	15	44	49	33

Con la tabla de contingencia calculamos las distancias entre preguntas y realizamos el agrupamiento. En el dendograma se aprecian claramente tres conglomerados. Es muy interesante constatar que los tres grupos están formados por preguntas que en su mayor parte son correlativas. Esto es consistente con que al elaborar un test normalmente se colocan las preguntas por unidades temáticas y con que el encuestado también suele hacerlo teniendo en cuenta esta estructura y tiende a responder de forma similar a las preguntas correlativas.

```
dist <- dist(table, method = "euclidean")
cluster <- hclust(dist, method = "complete")
plot(cluster)
```

Podemos distinguir los siguientes grupos y subgrupos:

[1] El método de enlace completo usa la distancia máxima entre dos conglomerados para seleccionar los más cercanos a unir.

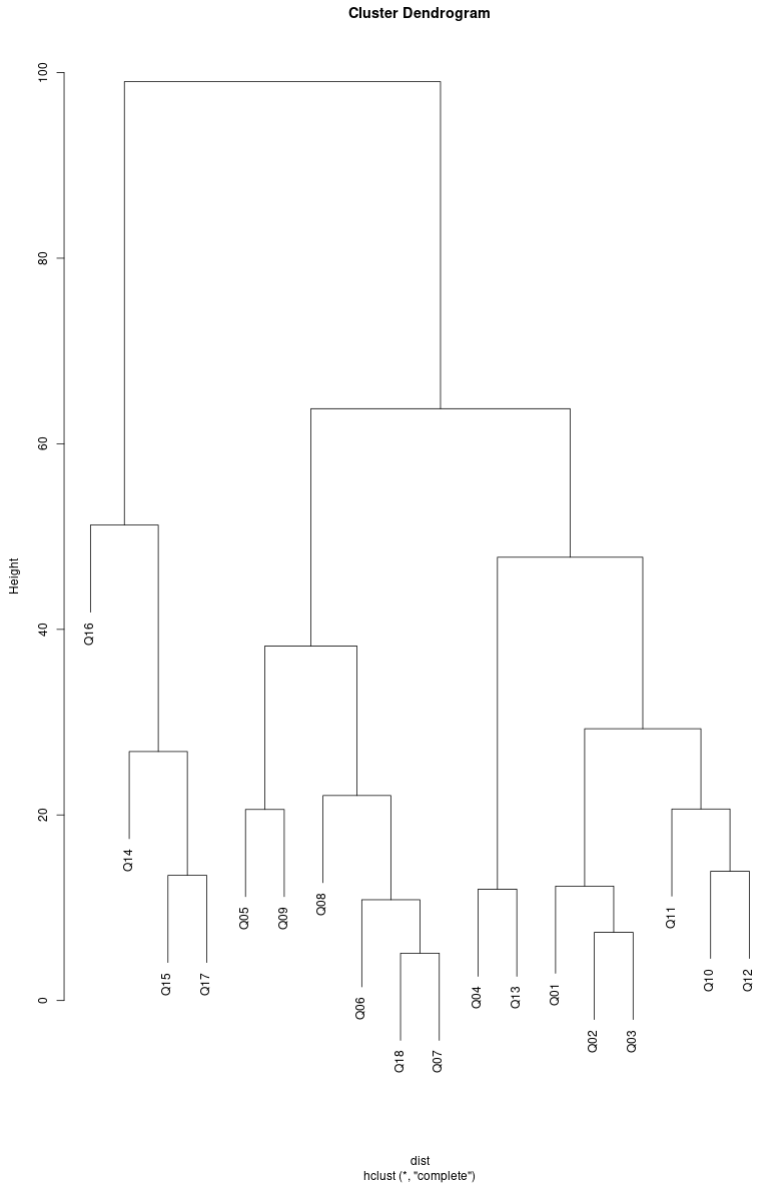


Figura 7.1: Dendograma de aglomeramiento jerárquico de preguntas en función de la tabla de contingencia de respuestas.

- Grupo 1: Trata sobre la corrección del subtítulo.
  - Subgrupo 05, 06, 07, 08, 09: preguntas sobre si la información que presenta el subtítulo es correcta y está bien escrita.
  - Pregunta 18: valoración general del subtitulado. El que esta pregunta esté incluida en este grupo estaría indicando que este es el apartado al que más importancia dan los estudiantes a la hora de valorar la calidad del subtitulado.

- Grupo 2: Es el más numeroso. En general está formado por preguntas sobre el grado de dificultad que presenta la lectura del subtítulo.

  - Subgrupo preguntas Q01, Q02, Q03: colocación de los subtítulos.
  - Subgrupo preguntas Q10, Q11, Q12: sincronización, velocidad y número de líneas.
  - Subgrupo preguntas Q04, Q13: contraste y legibilidad.

- Grupo 3: son preguntas que tratan también sobre la corrección del subtítulo pero con la diferencia sobre el grupo uno de que se trata de cuestiones más sutiles y presumiblemente más difíciles de valorar para un novato. Está formado por las preguntas Q14, Q15, Q16 y Q17.

7.2 Análisis de tablas de contingencia.

En esta sección se aplicarán técnicas estadísticas que se basan en tablas de contingencia. Una descripción teórica de este tipo de técnicas se pueden encontrar en Agresti (2018). Un tratamiento aplicado y basado en gráficos, que será el enfoque que seguiremos en este trabajo, es realizado en Friendly et al. (2015).

Asociación de variables con la prueba de homogeneidad χ^2.

Podemos usar la prueba de homogeneidad χ^2 para saber si las respuestas al cuestionario son independientes del nivel de subtitulado, del periodo y de la secuencia. Constatamos que según esta prueba ninguna de estas variables es independiente de la respuesta.

```
chisq.test(df_all$Treat, df_all$Response)
```

```
    Pearson's Chi-squared test

data:  df_all$Treat and df_all$Response
X-squared = 621.5, df = 5, p-value < 2.2e-16
```

```
chisq.test(df_all$Period, df_all$Response)
```

```
    Pearson's Chi-squared test

data:  df_all$Period and df_all$Response
X-squared = 15.039, df = 5, p-value = 0.0102
```

```
chisq.test(df_all$Seq, df_all$Response)
```

```
    Pearson's Chi-squared test

data:  df_all$Seq and df_all$Response
X-squared = 64.904, df = 5, p-value = 1.173e-12
```

Comparación mediante mosaicos.

En el Figura 7.2 se representan en forma de mosaico las tablas de contingencia de las respuestas por tratamiento y secuencia. La información mostrada es similar a la que presentamos en la Figura 6.5, aunque el gráfico es más intuitivo ya que la anchura y altura de los rectángulos son proporcionales a la frecuencia marginal de la secuencia y el tratamiento respectivamente y el área es proporcional a la frecuencia conjunta. En esta ocasión hemos decidido emparejar los tratamientos en lugar de hacerlo con la secuencia, como hicimos anteriormente. Esto permite una mejor comparación de las diferencias entre grupos. Con ello podemos ver fácilmente que el tratamiento *A* es mejor valorado por los estudiantes y que el grupo que realizó la secuencia *AB* tiene más respuestas 5 pero menor número de respuestas positivas totales que el grupo de secuencia *BA* en ambos niveles de tratamiento.

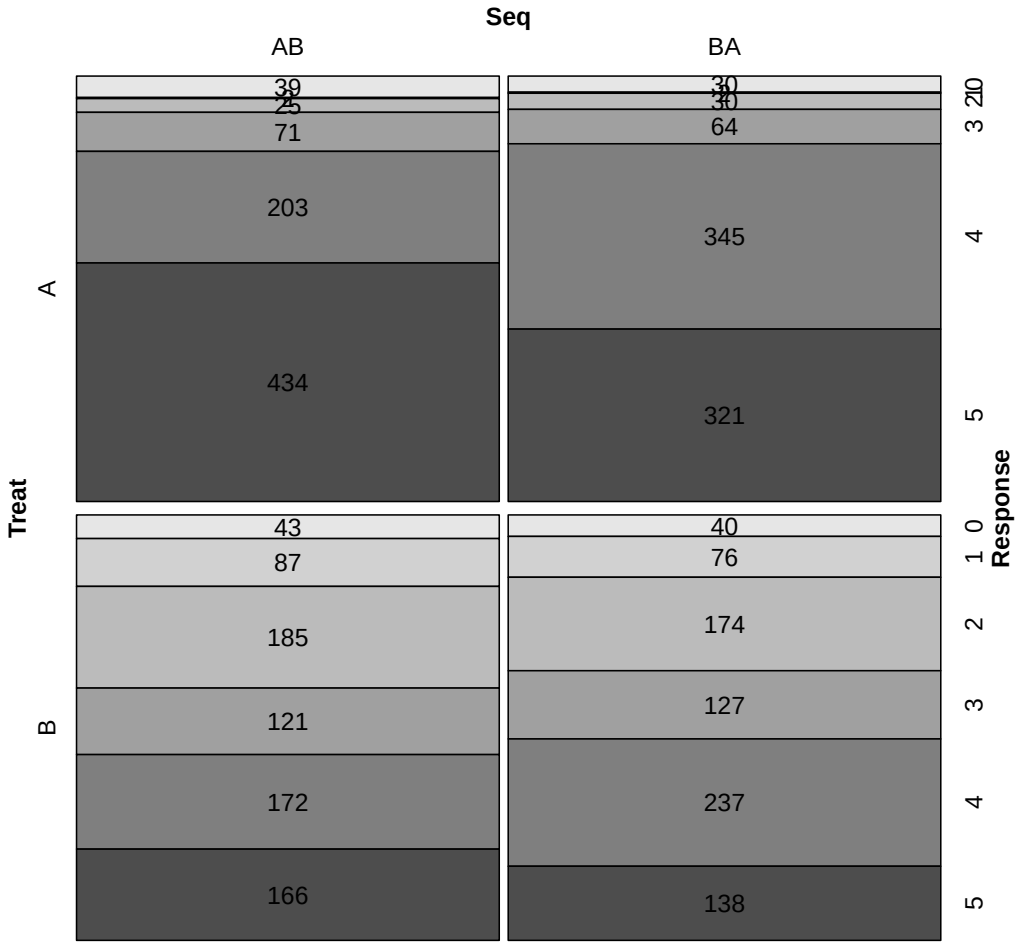


Figura 7.2: Mosaico de tratamientos y secuencias.

Comparación con *Odds Ratio*.

Hasta este momento ha quedado claro que el nivel de subtitulado *A* es preferido por los estudiantes y que las respuestas de ambos grupos son similares. Pero,

¿cuánto de similares son? Una forma de contestar esta pregunta es utilizar el `odds ratio` de tratamientos y grupos para cada nivel de respuesta.

Es decir, calcular:

$$OR_{(Treat,Seq|Response=r)} = \frac{\frac{P(Treat=A|Seq=AB,Response=r)}{P(Treat=B|Seq=AB,Response=r)}}{\frac{P(Treat=A|Seq=BA,Response=r)}{P(Treat=B|Seq=BA,Response=r)}} \tag{7.2}$$

Si los *OR* son similares en todos los niveles de respuesta, podemos afirmar que los grupos son homogéneos. Los resultados en R no producen significación estadística en ningún nivel de respuesta por lo que según esta prueba estadística la secuencia de subtitulado no influiría en la respuesta de los estudiantes (ver Figura 7.3).

```
summary(loddsratio(~ Treat + Seq + Response_l, data = df_all))
```

```
z test of coefficients:

                   Estimate Std. Error z value Pr(>|z|)
No sé / No contesto  0.19004    0.32746  0.5804   0.5617
Muy en desacuerdo   -0.13517    1.01225 -0.1335   0.8938
En desacuerdo       -0.24362    0.29066 -0.8382   0.4019
Neutral              0.15219    0.21412  0.7108   0.4772
De acuerdo          -0.20977    0.13363 -1.5698   0.1165
Muy de acuerdo       0.11687    0.13671  0.8549   0.3926
```

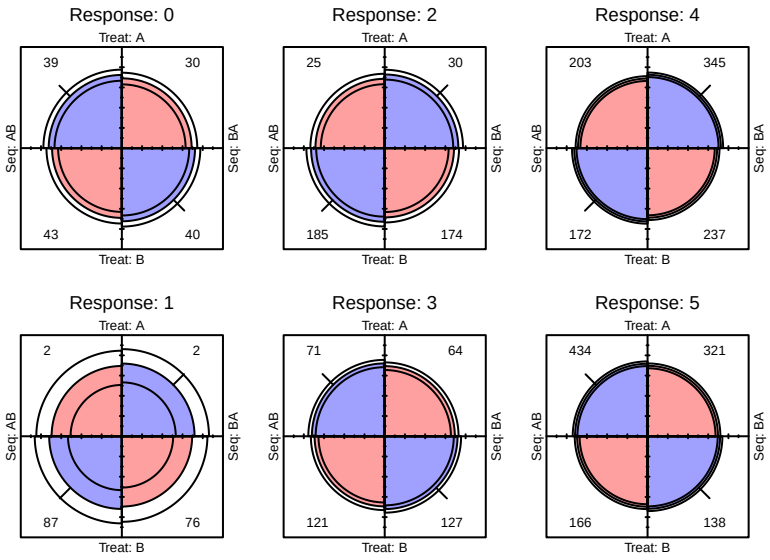


Figura 7.3: OR entre tratamiento y grupo por nivel de respuesta.

Sería interesante calcular el *OR* para cada nivel de respuesta y pregunta pero por desgracia la muestra es demasiado pequeña para hacerlo. Se ha calculado el *OR* sobre los agrupamientos de preguntas y se ha obtenido significación estadística tan solo en el agrupamiento 2 y nivel de respuesta 2:

```
summary(loddsratio(~ Treat + Seq + Cluster + Response_l, data = df_all))
```

```
z test of coefficients:

                      Estimate Std. Error z value Pr(>|z|)
1:No sé / No contesto -1.94591    1.75662 -1.1078  0.26797
2:No sé / No contesto  0.41074    1.12569  0.3649  0.71520
3:No sé / No contesto  0.29239    0.35972  0.8128  0.41632
1:Muy en desacuerdo   -0.12516    1.17012 -0.1070  0.91482
2:Muy en desacuerdo   -1.39488    1.66941 -0.8356  0.40341
3:Muy en desacuerdo    1.19870    1.69327  0.7079  0.47900
1:En desacuerdo        0.17829    0.49526  0.3600  0.71885
2:En desacuerdo       -1.34796    0.57253 -2.3544  0.01855 *
3:En desacuerdo        0.23740    0.50928  0.4661  0.64111
1:Neutral              0.62181    0.46172  1.3467  0.17807
2:Neutral              0.53248    0.39619  1.3440  0.17895
3:Neutral             -0.24146    0.31560 -0.7651  0.44421
1:De acuerdo          -0.35125    0.25963 -1.3529  0.17609
2:De acuerdo          -0.28064    0.18244 -1.5382  0.12399
3:De acuerdo           0.22503    0.30663  0.7339  0.46303
1:Muy de acuerdo       0.27860    0.26542  1.0497  0.29387
2:Muy de acuerdo       0.23441    0.17892  1.3101  0.19015
3:Muy de acuerdo      -0.61437    0.38071 -1.6137  0.10659
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Sin embargo no podemos asumir que esta significación no se deba al azar ya que estamos realizando 18 contrastes de hipótesis diferentes y cada uno tiene un error tipo I asociado, con lo que la probabilidad de encontrar una significación estadística por puro azar aumenta. Se han propuesto correcciones del *p*-value como la de Bonferroni para abordar este problema que no se aplican en este trabajo.

Otro *OR* que tiene interés calcular es el de tratamiento y periodo para evaluar si las respuestas son homogéneas. Mostramos tanto la tabla de resultados en R y también su representación visual (ver Figura 7.4).

```
summary(loddsratio(~ Treat + Period + Response_l, data = df_all))
```

```
z test of coefficients:

                    Estimate Std. Error z value  Pr(>|z|)
No sé / No contesto  0.33469    0.32746  1.0221 0.3067511
Muy en desacuerdo    0.13517    1.01225  0.1335 0.8937673
En desacuerdo       -0.12102    0.29067 -0.4164 0.6771467
Neutral              0.05540    0.21412  0.2587 0.7958414
De acuerdo          -0.85090    0.13363 -6.3674 1.922e-10 ***
Muy de acuerdo       0.48634    0.13671  3.5574 0.0003745 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Podemos constatar la existencia de un efecto periodo de signo contrario para las preguntas 4 y 5. La razón de que se produzca este efecto periodo es que algunas de las respuestas de valoración 5 en ambos niveles de subtitulado y grupos en

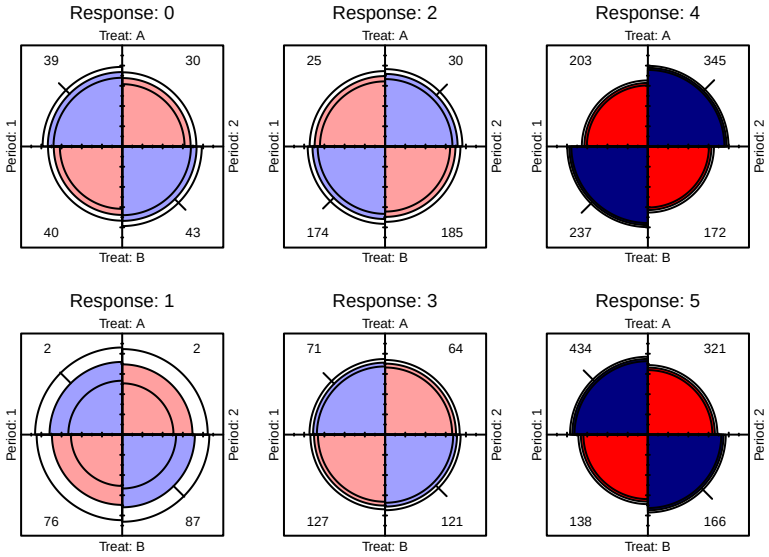


Figura 7.4: OR entre tratamiento y periodo por nivel de respuesta.

el primer periodo se convierten en valoración 4 en el segundo periodo. Esto indica que los estudiantes de ambos grupos prestaron más atención o fueron más exigentes en el segundo visionado y decidieron no otorgar la puntuación máxima incluso en algunos items al subtitulado correcto. Que el efecto periodo sea contrario en dos preguntas no debe sorprendernos en este diseño de experimento, ya que un test es un juego de suma cero: la valoraciones que se ganan o se pierden en un nivel de respuesta necesariamente provoca que el resto de niveles pierdan o ganen respectivamente la misma cantidad. En cualquier caso, vemos que el efecto periodo es cuantitativa y cualitativamente pequeño. Al afectar solo al intercambio de valoraciones entre los niveles 4 y 5, y ser las dos positivas, es simplemente una pequeña corrección en la valoración del subtitulado.

Capítulo 8

Modelado estadístico.

Vamos a construir diversos modelos para analizar la asociación de la variable respuesta sobre los dos niveles de subtitulado y la interacción con el periodo y la secuencia de tratamientos.

8.1 Árboles de inferencia condicional.

Los arboles de inferencia condicional (CIT) son un tipo de árbol de decisión en el que la selección de variables y de los puntos de división no se basan en medidas de homogeneidad como el índice de Gini, sino en un contrastes de hipótesis no paramétricos. El algoritmo que se utiliza es el siguiente, ver Levshina (2020):

El algoritmo consiste en contrastar la hipótesis nula de si la variable de respuesta Y es independiente de alguna variable explicativa $Y \mid X$. Para probar la hipótesis, se utiliza un algoritmo de permutación de la variable respuesta y se mide la asociación con la variable explicativa antes y después de la permutación. Si la asociación no varía significativamente, podemos asumir que las variables de respuesta y explicativa son independientes. De esta forma se selecciona la variable explicativa que más influye en la respuesta y que se utilizará en el particionado. Para elegir el valor de la variable explicativa que dividirá el conjunto de datos, se procede de forma análoga midiendo el cambio en la diferencia de asociación. De acuerdo con Friendly (2015), los CIT resuelven los problemas de sobreajuste de los árboles de decisión tradicionales.

Para realizar el particionado basado en CIT, vamos a usar la función `ctree` del paquete `party` de R. Presentamos aquí únicamente el modelo final elegido que incluye como variables explicativas `Treat`, `Period`, `Seq` y `Cluster` [1].

[1] Se han realizado simulaciones con otras combinaciones de variables explicativas que no se incluyen por no haber producido resultados relevantes.

En la Figura 8.1 podemos ver que el nivel de subtitulado es el efecto principal, seguido del grupo de preguntas y finalmente la secuencia. En este modelo el periodo no aparece por no estar asociado con la respuesta. Estos resultados son contradictorios con los que obtuvimos en el análisis con el OR (ver Sección 7.2) en el que el factor secuencia no era significativo pero sí lo era el factor periodo. Por otro lado, vemos que la asociación más fuerte es el nivel de respuesta 5 para subtitulado *A*, grupos de preguntas 1 y 2 y secuencia *AB* y de las respuestas 4 y 5 cuando la secuencia es *BA*. El tratamiento *B* está fuertemente asociado con el nivel de respuesta 1 para el grupo de preguntas 1. Por último, con este modelo no hay ninguna combinación de factores que prediga un nivel de respuesta 1.

```
tree.1 <- ctree(Response ~ Treat + Cluster + Period + Seq, data = df_clean)
```

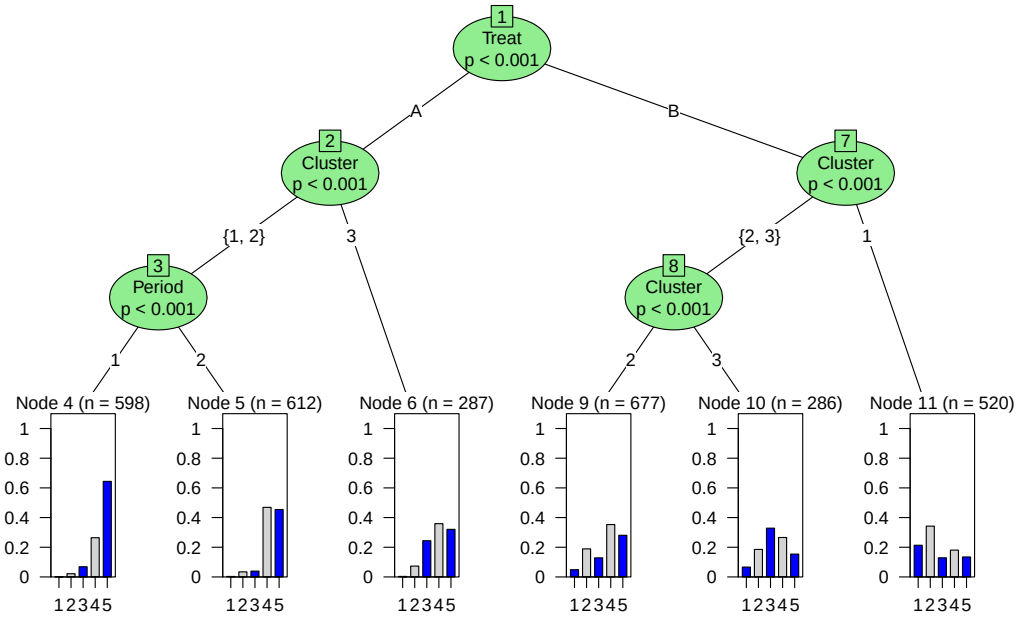


Figura 8.1: Modelo con árboles de inferencia condicional (Response ~ Treat + Cluster + Period + Seq).

Aunque no es el objetivo del trabajo, podemos usar este modelo para hacer predicciones. La matriz de contingencia resultante es la siguiente:

```
          Prediction
Reference   1   2   3   4   5
        1   0 111  19  36   1
        2   0 178  53 170  13
        3   0  67  94 181  41
        4   0  94  76 629 158
        5   0  70  44 560 385
```

Como habíamos anticipado, nunca se predice el nivel de respuesta 1. Las categorías que más probablemente predice nuestro modelo son la 4 y la 5 pero aún así hay mucha confusión entre ellas. La exactitud de predicción es 43%.

Un modelo alternativo sería usar las mismas variables explicativas pero cambiado `Response` por `Level` como variable de respuesta. Esta variable solo tiene tres

niveles: positivo, negativo y neutro. De esta forma no se producen confusiones entre los niveles 1 y 2 por un lado y 4 y 5 por otro:

```
tree.2 <- ctree(Level ~ Treat + Cluster + Period + Seq, data = df_clean)
```

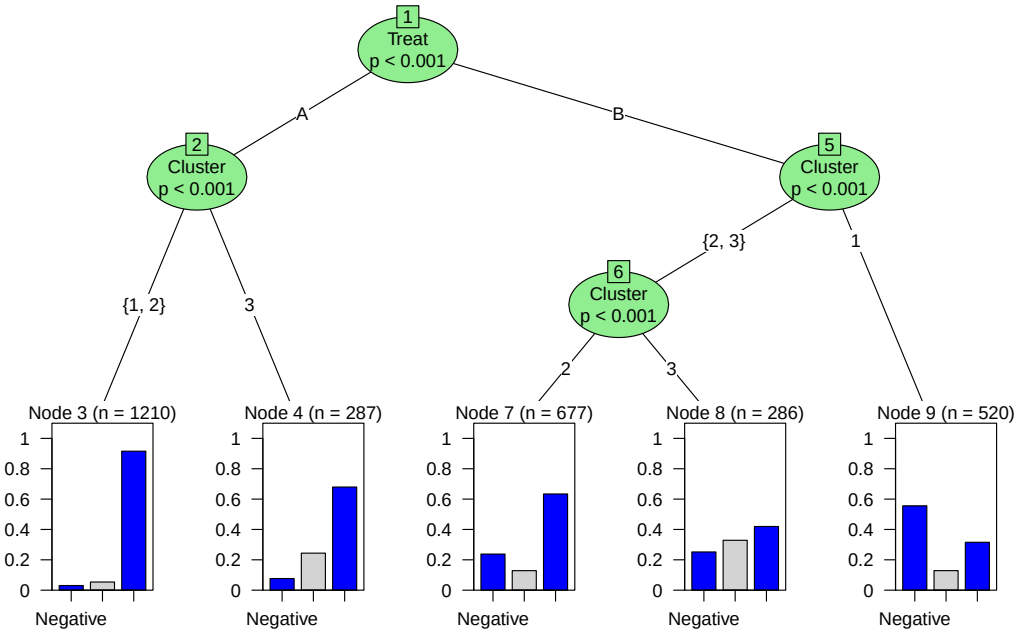


Figura 8.2: Modelo con árboles de inferencia condicional (Level ~ Treat + Cluster + Period + Seq).

El árbol obtenido con variable respuesta `Level` (ver Figura 8.2) es muy similar al otro (ver Figura 8.1) con la principal diferencia de que ahora la secuencia ha desaparecido como factor relevante. Por otro lado, el modelo siempre predice una respuesta positiva excepto para el subtitulado *B* y grupo de preguntas 1, que es negativa (el nivel neutro nunca se predice). La exactitud del modelo ha subido a 72%. En cualquier caso no es una gran mejora ya que un modelo que predijera siempre la categoría mayoritaria (positiva), habría obtenido una exactitud de 68%. Se han hecho simulaciones consistentes en incluir como factor las preguntas o usar como modelo un árbol de decisión convencional con resultados similares.

8.2 Regresión ordinal.

El test de Likert es una escala ordinal. Los test estadísticos ANOVA o MANOVA presuponen que la variable de respuesta es cuantitativa y con distribución normal. Tratar las respuestas a un test de Likert como si fueran cuantitativas no es correcto por las siguientes razones:

- Los niveles de respuesta no son necesariamente equidistantes: la distancia entre un par de opciones de respuesta puede no ser la misma para todos

los pares de opciones de respuesta. Por ejemplo, la diferencia entre «Muy en desacuerdo» y «En desacuerdo» y la diferencia entre «De acuerdo» y «Muy de acuerdo» es de un nivel, pero psicológicamente puede ser percibida de forma diferente por cada sujeto.

- La distribución de las respuestas ordinales puede ser no normal. En particular esto sucederá si hay muchas respuestas en los extremos del cuestionario.

- Las varianzas de las variables no observadas que subyacen a las variables ordinales observadas pueden diferir entre grupos, tratamientos, periodos, etc.

En Liddell y Kruschke (2018) se han analizado los problemas potenciales de tratar datos ordinales como si fueran cuantitativos constatando que se pueden presentar las siguientes situaciones:

- Se pueden encontrar diferencias significativas entre grupos cuando no las hay: error tipo I.
- Se pueden obviar diferencias cuando en realidad sí existen: error tipo II.
- Incluso se pueden invertir los efectos de un tratamiento.
- También puede malinterpretarse la interacción entre factores.

La Regresión Logística Multinomial es una extensión de la Regresión Logística cuando la variable de respuesta es nominal. La Regresión Ordinal tiene en consideración que los valores nominales de la variable de respuesta están ordenados y por eso será el modelo que utilizaremos.

Variantes de la Regresión Ordinal.

Los modelos lineales generalizados (*GLM*) son modelos en los que la variable respuesta no es normal. Para especificar un *GLM* son necesarios tres componentes (ver O'Connell 2006):

- Un componente aleatorio: será una distribución de probabilidad de la familia exponencial que se asume que sigue la variable respuesta (en la regresión logística será la distribución Binomial o la distribución de Bernoulli).
- Un componente lineal y aditivo de predictores.
- Una función de enlace que realiza transformación de los valores del componente lineal a los que puede tomar la variable respuesta. Por ejemplo en la función logística será la función $logit^{-1}(x)$. Esta función permite pasar de un rango de valores $(-\infty, +\infty)$ a un rango $(0, 1)$.

La Regresión Ordinal es una extensión de la Regresión Logística y, por lo tanto de *GLM*. Según Bürkner y Vuorre (2019) hay tres clases de Regresión Ordinal:

- Regresión ordinal acumulativa.

- Regresión ordinal secuencial.
- Regresión ordinal adyacente.

Nos centraremos en la primera ya que es la más habitual y adecuada para nuestro caso (ver ibíd., pp. 23-24). El modelo acumulativo, CM, presupone que la variable ordinal observada, Y, proviene de la categorización de una variable latente (no observada) continua, $\tilde{Y}$. Hay K umbrales τ_k que particionan $\tilde{Y}$ en $K+1$ categorías ordenadas observables (ver Figura 8.3). Si asumimos que $\tilde{Y}$ tiene una cierta distribución (por ejemplo, normal) con distribución acumulada F, se puede calcular la probabilidad de que Y sea la categoría k de esta forma:

$$Pr(Y = k) = F(\tau_k) - F(\tau_{k-1})$$

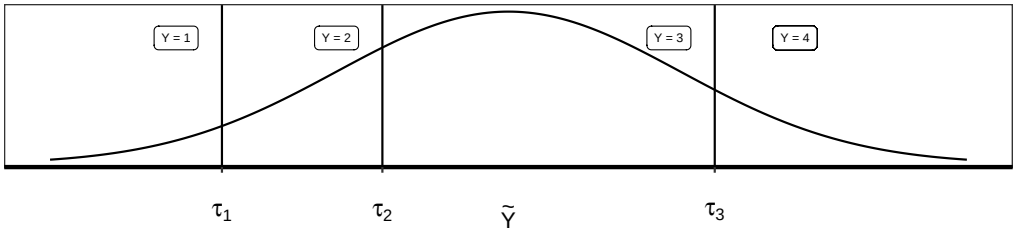


Figura 8.3: Función latente en una regresión ordinal acumulativa.

Por ejemplo en la Figura 8.3,

$$Pr(Y = 2) = F(\tau_2) - F(\tau_1)$$

Si suponemos que $\tilde{Y}$ tiene una relación lineal los predictores:

$$\tilde{Y} = \eta + \epsilon = \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p + \epsilon$$

,
y que los errores son $N(0, \sigma^2)$. Entonces la función de probabilidad acumulada de los errores tendrá la misma forma que la de $\tilde{Y}$:

$$\Pr(\epsilon \leq z) = F(z)$$

Y podremos calcular la distribución de probabilidad acumulada de Y:

$$\Pr(Y \leq k \mid \eta) = \Pr(\tilde{Y} \leq \tau_k \mid \eta) = \Pr(\eta + \epsilon \leq \tau_k) = \Pr(\epsilon \leq \tau_k - \eta) = F(\tau_k - \eta)$$

Por lo que asumiendo la normalidad de los errores:

$$\Pr(Y = k) = \Phi(\tau_k - \eta) - \Phi(\tau_{k-1} - \eta)$$

Donde hay que estimar los umbrales y los coeficientes de regresión. La función anterior es la conocida como la función de enlace `probit`. Otra función de enlace popular es la función `logit`. Es la que usaremos en este trabajo por ser más

fácil su interpretación [2]. Con esta función de enlace la interpretación de los coeficientes es parecida a de los coeficientes de la regresión logística. Se parte del supuesto de que el *logit* de la función de probabilidad es lineal:

$$logit[P(Y \leq k)] = \tau_k - \eta = \tau_k - (\beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p)$$

En ese caso, se puede demostrar fácilmente que, por ejemplo:

$$\frac{\frac{\Pr(Y \leq k|\eta)}{\Pr(Y > k|\eta)}}{\frac{\Pr(Y \leq k+1|\eta)}{\Pr(Y > k+1|\eta)}} = \exp(\tau_k - \tau_{k+1})$$

Y que [3]:

$$\frac{\frac{\Pr(Y \leq k|x_i=1)}{\Pr(Y > k|x_i=1)}}{\frac{\Pr(Y \leq k|x_i=0)}{\Pr(Y > k|x_i=0)}} = \exp(-\beta_i)$$

o, equivalentemente:

$$\frac{\frac{\Pr(Y > k|x_i=x+1)}{\Pr(Y \leq k|x_i=x+1)}}{\frac{\Pr(Y > k|x_i=x)}{\Pr(Y \leq k|x_i=x)}} = \exp(\beta_i)$$

Es decir, que $\exp(\beta_i)$ es el *OR* (cambio en *odds*) de que la variable respuesta esté por encima de una determinada categoría versus estar por debajo de ella para una unidad de incremento del predictor x_i. Este modelo se denomina proporcional ya que cada predictor se asume que tiene los mismos efectos sobre todas las categorías de la variable de respuesta ordinal (ver Liu 2022). Un valor del coeficiente β_i positivo indica que la relación entre el predictor x_i y la función de *logit* es positiva y, por lo tanto, se incrementa la posibilidad de un mayor valor de la variable respuesta. Como veremos, esta suposición se puede relajar y permitir que los coeficientes de todos o de algunos de los predictores sean diferentes para cada pareja consecutiva de valores de respuesta. Tendríamos entonces más parámetros a estimar con una interpretación más compleja.

Ajuste del modelo ordinal `Response ~ Treat`.

Existen varios paquetes en R que permiten ajustar una regresión ordinal logística. El más popular es el paquete `Ordinal` (Christensen 2022). El paquete `VGAM` (Yee 2023) es más flexible y potente. Otra posibilidad es usar la función `polr` del paquete `MASS` (Venables y Ripley 2002). Finalmente la función `orm` del paquete `rms` también permite hacerlo (ver Harrell 2015). En este trabajo usaremos el paquete `Ordinal` por permitir también incluir efectos aleatorios que utilizaremos en un apartado posterior. Comenzamos con un modelo simple que tiene como

[2] En la práctica los coeficientes estimados con las funciones de enlace `probit` y `logit` suelen similares.

[3] En el siguiente apartado se demuestra esta fórmula.

único predictor el nivel de subtitulado por ser la variable objetivo de nuestro modelo:

$$\text{logit}(P(Response_i \leq k)) = \tau_k - \beta_1 \text{Treat}_i,$$

```
clm_treat <-
    clm(
        Response ~ Treat,
        data = df_clean, link = "logit"
    )
summary(clm_treat)
```

```
formula: Response ~ Treat
data:    df_clean

 link  threshold nobs logLik   AIC     niter max.grad cond.H
 logit flexible  2980 -3966.11 7942.21 5(0)  1.64e-10 3.1e+01

Coefficients:
       Estimate Std. Error z value Pr(>|z|)
TreatB  -1.7206     0.0731  -23.54   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2 -3.97230    0.09678 -41.045
2|3 -2.45446    0.06812 -36.029
3|4 -1.66453    0.05936 -28.042
4|5 -0.10547    0.04946  -2.132
```

El método `summary()` muestra la información resumen. Para su interpretación vamos a seguir Christensen (2018). El número de condición Hessiano es inferior a 10^4 lo que es indicativo de que no hay problemas de optimización [4]. La sección de coeficientes es la más importante. Se muestra la estimación de parámetros, el error estándar y la significación estadística de acuerdo al test de Wald [5]. Comprobamos que el valor es claramente significativo. Es decir, que los estudiantes han valorado de forma diferente la calidad del subtitulado en ambos vídeos. El estimador de maxima verosimilitud del coeficiente `TreatB` es -1.72. Siguiendo la deducción de Bruin (2011) podemos, por ejemplo, hacer la siguiente interpretación del significado de este coeficiente referido a dos niveles consecutivos de respuesta:

$$\begin{aligned} logit[P(Y \leq 1)] &= -3.97 - (-1.72x_1) \\ logit[P(Y \leq 2)] &= -2.45 - (-1.72x_1) \end{aligned}$$

Por lo tanto los *odds* serían:

[4] El número de condición de Hessiano es una medida de la curvatura de una función en un punto. Si el número de condición de Hessiano es grande, la función es muy sensible a pequeñas perturbaciones y puede ser difícil de optimizar.

[5] El test de Wald es un contraste de hipótesis estadístico en el que se evalúa si el valor estimado es cero suponiendo que $W = \left(\frac{\hat{\theta}-\theta_0}{se(\hat{\theta})}\right)^2 \sim \chi^2$.

$$\begin{aligned}
\frac{P(Y \leq 1 \mid x_1 = B)}{P(Y > 1 \mid x_1 = B)} &= exp(-3.97)/exp(-1.72) \\
\frac{P(Y \leq 1 \mid x_1 = A)}{P(Y > 1 \mid x_1 = A)} &= exp(-3.97) \\
\frac{P(Y \leq 2 \mid x_1 = B)}{P(Y > 2 \mid x_1 = B)} &= exp(-2.45)/exp(-1.72) \\
\frac{P(Y \leq 2 \mid x_1 = A)}{P(Y > 2 \mid x_1 = A)} &= exp(-2.45)
\end{aligned}$$

Y los *OR*:

$$\begin{aligned}
\frac{P(Y \leq 1|x_1 = B)}{P(Y > 1|x_1 = B)} \Big/ \frac{P(Y \leq 1|x_1 = A)}{P(Y > 1|x_1 = A)} &= 1/exp(-1.72) = 5.59 \\
\frac{P(Y \leq 2|x_1 = B)}{P(Y > 2|x_1 = B)} \Big/ \frac{P(Y \leq 2|x_1 = A)}{P(Y > 2|x_1 = A)} &= 1/exp(-1.72) = 5.59
\end{aligned}$$

Se comprueba que el *OR* es equivalente en todos los niveles de respuesta al cuestionario. Esta es una de las suposiciones de la regresión ordinal acumulativa. El *odds* de respuesta al cuestionario entre los niveles inferiores y superiores a uno dado, k, es 5.59 veces en el subtitulado B que en el A. Esto indica que el subtitulado B es percibido por los estudiantes como de peor calidad que el subtitulado A. Concretamente, el coeficiente β para `Treat` es el `log odds` de observar una mejor respuesta en una pregunta del test es 5.59 veces superior en el nivel de subtitulado A que en el B. Aunque no suele ser de interés la interpretación de los coeficientes de los umbrales (`Threshold coefficients`), se pueden utilizar para estimar las probabilidades de respuesta. Por ejemplo, para el nivel de subtitulado B:

$$\begin{aligned}
logit[P(Y \leq 1)] &= -3.97 - (-1.72) = -2.25 \\
odds(P(Y \leq 1)) &= exp(logit[P(Y \leq 1)]) = 0.11 \\
P(Y \leq 1) &= \frac{exp(-2.25)}{1 + exp(-2.25)} = 0.10 \\
P(Y \leq 2) &= \frac{exp(-0.73)}{1 + exp(-0.73)} = 0.32 \\
P(Y = 2) &= P(Y \leq 2) - P(Y \leq 1) = 0.23
\end{aligned}$$

Para el subtitulado A no se tiene en cuenta el coeficiente $TreatB$ ya que el valor x_1 es cero:

$$\begin{aligned}
logit[P(Y \leq 1)] &= -3.97 \\
odds(P(Y \leq 1)) &= exp(logit[P(Y \leq 1)]) = 0.02 \\
P(Y \leq 1) &= \frac{exp(-3.97)}{1 + exp(-3.97)} = 0.02
\end{aligned}$$

En Tabla 8.1 se muestran las probabilidades para ambos niveles de subtitulado y todos los posibles valores de respuesta.

Cuadro 8.1: Probabilidades de respuesta para el modelo ordinal Response ~ Treat

	1	2	3	4	5
A	0.018	0.061	0.08	0.315	0.526
B	0.095	0.229	0.19	0.320	0.166

Ajuste del modelo ordinal `Response ~ Treat + Period`.

Para saber si existe un efecto periodo, añadimos como predictor la variable `Period`.

$$\text{logit}(P(Response_i \leq k)) = \tau_k - \beta_1 \text{Treat}_i - \beta_2 \text{Period}_i$$

```
clm_treat_period <-
    clm(
        Response ~ Treat + Period,
        data = df_clean, link = "logit"
    )
summary(clm_treat_period)
```

```
formula: Response ~ Treat + Period
data:    df_clean

 link  threshold nobs logLik   AIC     niter max.grad cond.H
 logit flexible  2980 -3957.88 7927.76 5(0)  1.94e-10 4.1e+01

Coefficients:
        Estimate Std. Error z value Pr(>|z|)
TreatB  -1.74090    0.07339  -23.72  < 2e-16 ***
Period2 -0.27560    0.06805   -4.05 5.12e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2 -4.13085    0.10507 -39.314
2|3 -2.60905    0.07872 -33.143
3|4 -1.81652    0.07073 -25.681
4|5 -0.25187    0.06153  -4.093
```

Vemos que ambos coeficientes son significativos y con signo negativo. Un signo negativo en el efecto periodo está asociado con que la valoración del subtitulado empeora en el segundo periodo independientemente de si se trata del subtitulado correcto o incorrecto. Aplicando el mismo razonamiento del apartado anterior, el *OR* del efecto periodo es $1/exp(-0.28) = 1.32$. Lo que quiere decir que una vez controlado el efecto principal del tratamiento, el subtitulado en el segundo periodo es valorado como de inferior calidad que en el primero. Esto estaría indicando que los estudiantes son más exigentes con el subtitulado en la segunda actividad independientemente de su calidad real.

Ajuste del modelo ordinal `Response ~ Treat * Period`.

Añadimos al modelo la interacción entre subtitulado y periodo. Esta interacción corresponde al efecto secuencia. Se puede demostrar que los modelos `Response ~ Treat*Period` y `Response ~ Treat + Period + Seq` son equivalentes si se cambia el contraste por defecto utilizado en R, que es `treatment`, a `sum` [6].

```
options(contrasts = rep("contr.sum", 2))
clm_treat_period_seq.sum <-
    clm(
        Response ~ Treat + Period + Seq,
        data = df_clean, link = "logit"
    )
coef(clm_treat_period_seq.sum)
```

```
      1|2       2|3       3|4       4|5     Treat1    Period1       Seq1
-3.1266457 -1.6083838 -0.8184857  0.7499332  0.8739547  0.1396182  0.1062749
```

```
options(contrasts = rep("contr.sum", 2))
clm_treat.period.sum <-
    clm(
        Response ~ Treat * Period,
        data = df_clean, link = "logit"
    )
coef(clm_treat.period.sum)
```

```
        1|2          2|3          3|4          4|5        Treat1
 -3.1266457   -1.6083838   -0.8184857    0.7499332    0.8739547
    Period1 Treat1:Period1
  0.1396182    0.1062749
```

Vemos que los coeficientes `Seq1` y `Treat1:Period1` son iguales y, por lo tanto, queda demostrado que la secuencia es la interacción entre periodo y tratamiento. Sin embargo los coeficientes son diferentes si el contraste es `treatment` [7]:

```
options(contrasts = rep("contr.treatment", 2))
clm_treat_period_seq <-
    clm(
        Response ~ Treat + Period + Seq,
        data = df_clean, link = "logit"
    )
coef(clm_treat_period_seq)
```

```
      1|2       2|3       3|4       4|5     TreatB    Period2      SeqBA
-4.2464935 -2.7282315 -1.9383335 -0.3699146 -1.7479094 -0.2792363 -0.2125498
```

```
options(contrasts = rep("contr.treatment", 2))
clm_treat.period <-
    clm(
        Response ~ Treat * Period,
        data = df_clean, link = "logit"
```

[6] Ver Apéndice C para una discusión sobre el significado y la interpretación de los contrastes `treatment` y `sum`.

[7] Los interceptores sí son iguales.

```
    )
coef(clm_treat.period)
```

```
       1|2        2|3        3|4        4|5     TreatB
-4.2464935 -2.7282315 -1.9383335 -0.3699146 -1.9604592
   Period2 TreatB:Period2
-0.4917861  0.4250996
```

En el Apéndice C se explica como se pueden obtener los coeficientes de un modelo a partir de los coeficientes de otro modelo. Es decir, que se pueden obtener los coeficientes del modelo `clm_treat.period` a partir de los coeficientes del modelo `clm_treat.period.sum`. Por ejemplo, el coeficiente *TreatB* del modelo `clm_treat.period` se calcula:

```
(-2 * (coef(clm_treat.period.sum)["Treat1"] + coef(clm_treat.period.sum)["Treat1:Period1"]))
```

```
   Treat1
-1.960459
```

```
coef(clm_treat.period)["TreatB"]
```

```
   TreatB
-1.960459
```

Sin embargo la interpretación de los coeficientes del segundo modelo, `clm_treat.period`, es más sencilla ya que es la que estamos habituados a utilizar en R. Por ello en este análisis se utilizará el modelo `clm_treat.period`. El resumen del ajuste es:

```
summary(clm_treat.period)
```

```
formula: Response ~ Treat * Period
data:    df_clean

 link  threshold nobs logLik   AIC     niter max.grad cond.H
 logit flexible  2980 -3953.01 7920.03 5(0)  2.14e-10 8.1e+01

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
TreatB         -1.96046    0.10229 -19.166  < 2e-16 ***
Period2        -0.49179    0.09744  -5.047 4.49e-07 ***
TreatB:Period2  0.42510    0.13638   3.117  0.00183 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2 -4.24649    0.11182 -37.977
2|3 -2.72823    0.08821 -30.928
3|4 -1.93833    0.08167 -23.732
4|5 -0.36991    0.07308  -5.062
```

Vemos que los tres coeficientes son significativos. El principal efecto es el nivel de subtitulado obteniendo mejores puntuaciones el nivel A; el efecto periodo es negativo por lo que el primer periodo obtiene mejores puntuaciones; por último, el efecto secuencia es positivo pero de menor valor absoluto que el efecto periodo. Esto quiere decir que el subtitulado de nivel B en el periodo 2 (secuencia AB), tiene un efecto periodo inferior que el subtitulado A en el mismo periodo. Matemáticamente:

$$
\begin{aligned}
logit[P(Y \leq 1 \mid Treat = A, Period = 1)] &= -4.25 \\
logit[P(Y \leq 1 \mid Treat = B, Period = 1)] &= -4.25 - (-1.96) \\
logit[P(Y \leq 1 \mid Treat = A, Period = 2)] &= -4.25 - (-0.49) \\
logit[P(Y \leq 1 \mid Treat = B, Period = 2)] &= -4.25 - (-1.96 - 0.49 + 0.43)
\end{aligned}
$$

En definitiva, en el nivel de subtitulado B apenas encontramos diferencias entre periodos, sin embargo, en el nivel de subtitulado A existe un efecto periodo cuyo valor en `logits` es -0.49. Es decir, que la valoración del subtitulado de nivel A es inferior en el segundo periodo que en el primero. En la Figura 8.4 podemos ver las predicciones del modelo.

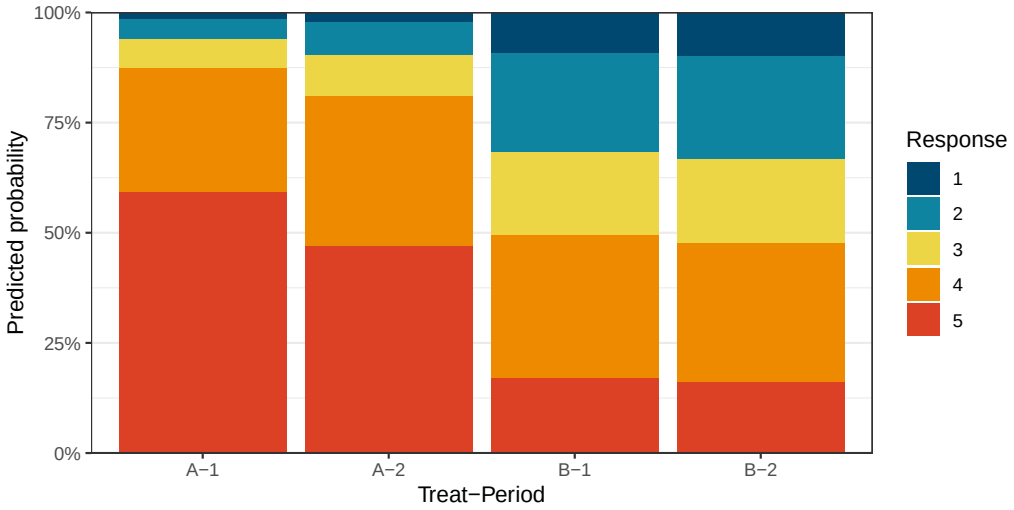


Figura 8.4: Probabilidades de respuesta para el modelo ordinal Response ~ Treat * Period

Elección del modelo ordinal mediante el test de razón de verosimilitud.

Al ser los tres modelos anidados, podemos compararlos con la prueba de razón de verosimilitud. Comprobamos que el tercer modelo (el que incorpora la interacción entre los subtítulos y el periodo) reduce significativamente el logaritmo de la función de verosimilitud y, por lo tanto, debe ser aceptado:

```
anova(clm_treat, clm_treat_period, clm_treat.period)
```

```
Likelihood ratio tests of cumulative link models:

                 formula:                  link: threshold:
clm_treat        Response ~ Treat          logit flexible
clm_treat_period Response ~ Treat + Period logit flexible
clm_treat.period Response ~ Treat * Period logit flexible

                 no.par    AIC  logLik LR.stat df Pr(>Chisq)
clm_treat             5 7942.2 -3966.1
clm_treat_period      6 7927.8 -3957.9  16.448  1      5e-05 ***
clm_treat.period      7 7920.0 -3953.0   9.738  1   0.001805 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Comprobación de las hipótesis del modelo.

La principal hipótesis de un modelo de regresión logística ordinal proporcional acumulativa es que los coeficientes son iguales entre cualesquiera dos niveles de respuestas correlativos. Se han propuesto diversas fórmulas para comprobar esta hipótesis. El paquete `Ordinal` dispone de la función `nominal_test()` que lo que hace es realizar un test de razón de verosimilitud para cada predictor ajustando un modelo en el que se ha relajado la condición de proporcionalidad. Se constata que el test resulta significativo para `Treat` y para `Treat:Period`, por lo que para estas dos variables no se puede asumir que los coeficientes estimados se mantengan constantes en todos los niveles de respuesta.

```
nominal_test(clm_treat.period)
```

```
Tests of nominal effects

formula: Response ~ Treat * Period
            Df  logLik    AIC     LRT Pr(>Chi)
<none>         -3953.0 7920.0
Treat        3 -3904.4 7828.9  97.172   <2e-16 ***
Period       3 -3951.4 7922.7   3.307   0.3467
Treat:Period 9 -3884.8 7801.6 136.408   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Lo que procede es ajustar el modelo relajando la constante de proporcionalidad de esas variables. Se ha realizado esto utilizando la función `vglm` del paquete `VGAM`. Vemos que ahora hay cuatro coeficientes para cada una de las variables `Treat` y `Treat:Period` [8].

```
vglm_treat.period <- vglm(
    Response ~ Treat * Period,
    VGAM::cumulative(link = "logit", parallel = F ~ Treat + Treat:Period, reverse = T),
    data = df_clean
)
```

[8] Los umbrales tienen los mismo valores pero de signo contrario debido a diferencias en la parametrización del modelo en cada función utilizada.

```
coef(vglm_treat.period) %>% data.frame()
```

```
                      .
(Intercept):1     6.2295614
(Intercept):2     3.5001281
(Intercept):3     2.2121438
(Intercept):4     0.2998788
TreatB:1         -4.0441016
TreatB:2         -2.8029889
TreatB:3         -2.2174630
TreatB:4         -1.7926200
Period2          -0.5345919
TreatB:Period2:1  0.3509228
TreatB:Period2:2  0.3607009
TreatB:Period2:3  0.3891474
TreatB:Period2:4  0.8024952
```

En la Figura 8.5 se muestran las probabilidades de respuesta de este modelo.

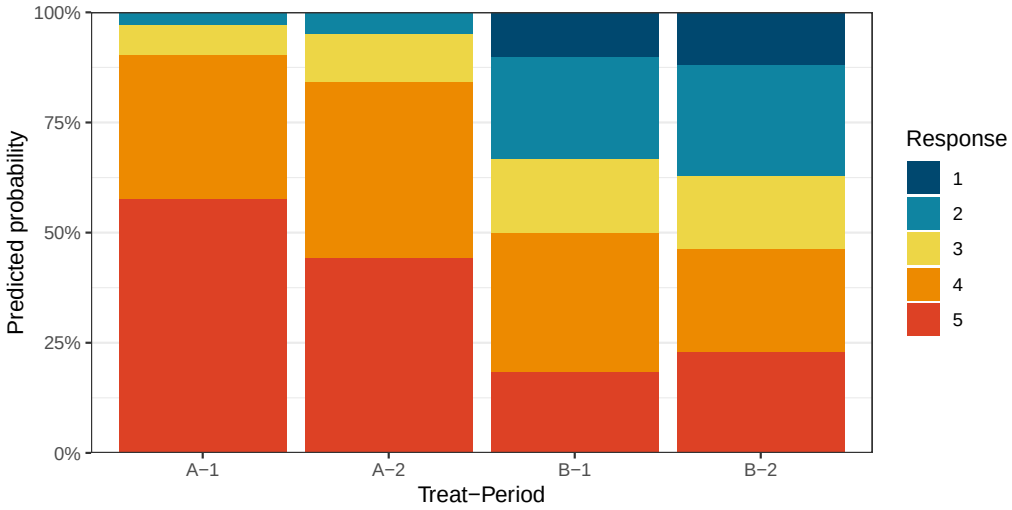


Figura 8.5: Probabilidades de respuesta para el modelo ordinal no proporcional Response ~ Treat * Period

Introducción a los modelos multinivel.

Un modelo multinivel, jerárquico o mixto es un modelo en el que tenemos datos de un nivel inferior anidados en estructuras de un nivel superior. Por ejemplo, si quisiéramos evaluar el rendimiento de varios métodos de enseñanza, poríamos seleccionar aleatoriamente varios colegios participantes y en cada uno de ellos elegir varias clases en las que se impartiría uno de los métodos de enseñanza. Los modelos multinivel se utilizan cuando se incumple la hipótesis de independencia de entre las observaciones. En el caso de los métodos de enseñanza, los alumnos de una clase no son independientes de los alumnos de otra clase del mismo colegio y tampoco lo son los alumnos de dos colegios diferentes. Otra situación en la que se viola la condición de independencia entre observaciones es cuando se toman varias medidas del mismo sujeto. Este tipo de experimentos se llaman

de medidas repetidas o longitudinales. En este caso se consideran que las medidas están anidadas en el sujeto (ver Liu 2022). En un modelo multinivel no es necesario que todas las variables tengan una estructura jerárquica. Distinguimos entonces dos tipos de variables. Las conocidas como de efectos fijos son aquellas variables que se consideran que tienen el mismo efecto en toda la población y, por lo tanto, estimamos un único coeficiente. Las que llamamos como variables de efectos aleatorios tienen un coeficiente diferente para cada elemento de la población y se supone que son una muestra de una población mucho mayor, como el caso de seleccionar aleatoriamente una muestra de colegios. Normalmente el coeficiente particular de cada elemento no es de interés para el investigador y se supone que tienen una media centrada en cero. El mayor interés de los efectos aleatorios es la estimación de su matriz de varianzas-covarianzas.

La ecuación general de un modelo multinivel con dos niveles y un solo predictor con efectos aleatorios es (ver D.-G. Chen y J. Chen 2021, pp. 40):

$$\begin{aligned} Level\ 1: y_{ij} &= \beta_{0j} + \beta_{1j}x_{1ij} + \epsilon_{ij} \\ Level\ 2: \beta_{0j} &= \beta_0 + U_{0j} && (intercepto\ aleatorio) \\ \beta_{1j} &= \beta_0 + U_{1j} && (pendiente\ aleatoria) \end{aligned}$$

Donde los errores del modelo se distribuyen,

$$Error\ intra\ grupo: \epsilon_{ij} \sim N(0, \sigma^2)$$

$$Error\ entre\ grupos: \begin{pmatrix} U_{0j} \\ U_{1j} \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \tau_0^2 & \tau_0\tau_1\rho_{01} \\ \tau_0\tau_1\rho_{01} & \tau_1^2 \end{pmatrix}\right)$$

donde j son los grupos que varían $j = 1, ..., J$ (J es el número de grupos); i es la observación i del grupo j ($i = 1, ..., n_j$, n_j es el número de observaciones del grupo j). El modelo se compone de una parte fija $\beta_0 + \beta_1 x_{ij}$ y una aleatoria $U_{0j} + U_{1j}x_{1ij} + \epsilon ij$. Los parámetros de este modelo son el intercepto y la pendiente de efectos fijos (β_0 y β_1), la varianza intra-grupos (σ^2), la varianza inter-grupos del intercepto aleatoria (τ_0) y de la pendiente aleatoria (τ_1), y la correlación entre intercepto y pendiente aleatorias (ρ_{01}). Cuando se introduce una estructura multinivel se pueden omitir tanto el intercepto como la pendiente aleatoria.

En Gelman et al. (2013) se evalúan tres posibilidades a la hora de definir un modelo:

- *Complete pooloing*: Consiste en estimar un único parámetro para todas las observaciones. Es equivalente a un modelo con efectos fijos.
- *No pooling*: Se estiman tantos paŕametros como grupos haya de forma independiente.
- *Partial pooling*: Es el modelo jerárquico. Es una mezcla de ambos, ya que aunque se estima un parámetro para cada grupo, esta estimación no es independiente, sino que se supone que las observaciones de un mismo

grupo proceden de una misma distribución de probabilidad. Esto se traduce en que se produce una contracción (*shrinkage*) en la estimación de los parámetros. Al influir la estimación de unas observaciones en otras, la estimación es de menor valor absoluto que la que resultaría en un modelo de *no pooling*. De esta forma podemos ver el *complete pooling* y el *no pooling* como dos casos particulares extremos del *no pooling*. La contracción de coeficientes en los modelos multinivel actúa como una regularización que puede evitar el sobreajuste.

Los modelos multinivel requieren supuestos adicionales en el nivel segundo y superiores que son similares a los supuestos para los modelos de efectos fijos en el primer y único nivel (ver D.-G. Chen y J. Chen 2021, pp. 43). Para estimar los parámetros en un modelo multinivel se utiliza el método de máxima verosimilitud restringida (RMLE) que es una variante de la estimación por máxima verosimilitud (MLE) en la que se hacen ajustes en los grados de libertad del modelo con efectos aleatorios.

Ajuste del modelo multinivel ordinal.

El modelo multinivel aleatorio más simple que podemos considerar es el que incorpora únicamente un interceptor aleatorio para los estudiantes del curso. Que los estudiantes sean considerados un efecto aleatorio está doblemente justificado. Por un lado, son una muestra de una población más amplia que estaría constituida por todos los estudiantes de todos los cursos de accesibilidad. Por otro, cada estudiante realiza el test de evaluación dos veces y, por lo tanto, las respuestas a estos cuestionarios no son independientes. La especificación del modelo será la siguiente:

$$\text{logit}(P(Response_{ij} \leq k)) = \tau_k + \tau_{kj} - \beta_1 \text{Treat}_{ij},$$

donde $Response_{ij}$ es la observación i del usuario j, τ_k es el interceptor común a todos los usuarios para el nivel de respuesta k y τ_{kj} es el interceptor específico para el usuario j. Para ajustar el modelo, vamos a utilizar la función `clmm()` del paquete `Ordinal` ya que permite la inclusión de efectos aleatorios.

```
clmm_subject <- clmm(Response ~ (1 | Subject), data = df_clean)
summary(clmm_subject)
```

```
Cumulative Link Mixed Model fitted with the Laplace approximation

formula: Response ~ (1 | Subject)
data:    df_clean

 link  threshold nobs logLik   AIC     niter     max.grad cond.H
 logit flexible  2980 -4053.61 8117.23 272(1093) 7.11e-04 8.3e+01

Random effects:
 Groups  Name        Variance Std.Dev.
 Subject (Intercept) 0.8385   0.9157
Number of groups:  Subject 87
```

```
No Coefficients

Threshold coefficients:
    Estimate Std. Error z value
1|2  -3.1597     0.1291 -24.481
2|3  -1.6869     0.1109 -15.212
3|4  -0.9383     0.1077  -8.713
4|5   0.6354     0.1068   5.951
```

Vemos que el parámetro $\widehat{\tau_0}$ tiene un valor 0.92 y que no hay coeficientes que estimar. El siguiente modelo en orden de complejidad es el que incorpora el predictor `Treat`:

```r
clmm_treat_subject <- clmm(Response ~ Treat + (1 | Subject), data = df_clean)
summary(clmm_treat_subject)
```

```
Cumulative Link Mixed Model fitted with the Laplace approximation

formula: Response ~ Treat + (1 | Subject)
data:    df_clean

 link  threshold nobs logLik   AIC     niter     max.grad cond.H
 logit flexible  2980 -3665.73 7343.47 395(1585) 4.97e-04 9.5e+01

Random effects:
 Groups  Name        Variance Std.Dev.
 Subject (Intercept) 1.265    1.125
Number of groups:  Subject 87

Coefficients:
       Estimate Std. Error z value Pr(>|z|)
TreatB  -2.0747     0.0793  -26.16   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2  -4.7243     0.1638 -28.839
2|3  -3.0518     0.1453 -21.008
3|4  -2.1256     0.1392 -15.269
4|5  -0.2068     0.1325  -1.561
```

En este modelo $\widehat{\tau_0}$ vale 1.12 y la pendiente del tratamiento, `TreatB` es -2.07. Podemos considerar un modelo en la que la valoración de cada sujeto sea diferente para cada tratamiento:

```r
clmm_treat.subject <- clmm(Response ~ Treat + (1 + Treat | Subject), data = df_clean)
summary(clmm_treat.subject)
```

```
Cumulative Link Mixed Model fitted with the Laplace approximation

formula: Response ~ Treat + (1 + Treat | Subject)
data:    df_clean

 link  threshold nobs logLik   AIC     niter     max.grad cond.H
 logit flexible  2980 -3431.27 6878.53 535(3741) 2.43e-03 1.7e+02

Random effects:
```

```
 Groups  Name        Variance Std.Dev. Corr
 Subject (Intercept) 2.691    1.641
         TreatB      4.295    2.072    -0.598
Number of groups:  Subject 87

Coefficients:
       Estimate Std. Error z value Pr(>|z|)
TreatB  -2.5864     0.2425  -10.67   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2  -5.5561     0.2207 -25.173
2|3  -3.6249     0.2024 -17.914
3|4  -2.5581     0.1968 -12.999
4|5  -0.3271     0.1900  -1.721
```

Ahora $\widehat{\tau_0}$ vale 1.64 y $\widehat{\tau_1}$ 2.07. La correlación, ρ_{01}, es -0.6. Podemos añadir el factor `Period` al modelo:

```r
clmm_treat.period.subject <- clmm(
    Response ~ Treat * Period + (1 + Treat | Subject),
    data = df_clean
)
summary(clmm_treat.period.subject)
```

```
Cumulative Link Mixed Model fitted with the Laplace approximation

formula: Response ~ Treat * Period + (1 + Treat | Subject)
data:    df_clean

 link  threshold nobs logLik   AIC     niter       max.grad cond.H
 logit flexible  2980 -3429.88 6879.76 1696(11676) 7.34e-05 6.4e+02

Random effects:
 Groups  Name        Variance Std.Dev. Corr
 Subject (Intercept) 2.588    1.609
         TreatB      4.168    2.042    -0.584
Number of groups:  Subject 87

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
TreatB         -2.8530     0.3795  -7.519 5.53e-14 ***
Period2        -0.5893     0.3685  -1.599    0.110
TreatB:Period2  0.5307     0.5855   0.906    0.365
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2  -5.8518     0.2892 -20.236
2|3  -3.9206     0.2754 -14.234
3|4  -2.8541     0.2714 -10.515
4|5  -0.6226     0.2654  -2.345
```

Vemos que, a diferencia de lo que sucedía en el modelo de efectos fijos, el periodo y la interacción del periodo con el subtitulado son ahora no significativos. Queda, por último, discutir cómo añadir las preguntas al modelo. Consideramos que las respuestas a las preguntas no son independientes unas de otras y que, por lo tanto,

deben ser consideradas efectos aleatorios. En Bürkner (2021) Bürkner y Vuorre (2019, pp. 19-20) podemos encontrar un ejemplo de esta solución. Las preguntas como efecto aleatorio se pueden añadir considerando únicamente el intercepto o el intercepto y la pendiente. Ajustamos ambos modelos:

```
clmm_treat.period.subject_question <- clmm(
    Response ~ Treat * Period + (1 + Treat | Subject) + (1 | Question),
    data = df_clean
)
summary(clmm_treat.period.subject_question)
```

```
Cumulative Link Mixed Model fitted with the Laplace approximation

formula: Response ~ Treat * Period + (1 + Treat | Subject) + (1 | Question)
data:    df_clean

 link  threshold nobs logLik   AIC     niter     max.grad cond.H
 logit flexible  2980 -3309.04 6640.07 960(7467) 3.85e-04 6.6e+02

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 Subject  (Intercept) 3.0249   1.739
          TreatB      4.8228   2.196    -0.591
 Question (Intercept) 0.4651   0.682
Number of groups:  Subject 87,  Question 18

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
TreatB         -3.0562     0.4062  -7.525 5.29e-14 ***
Period2        -0.6163     0.3966  -1.554    0.120
TreatB:Period2  0.5608     0.6266   0.895    0.371
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2  -6.2609     0.3506 -17.858
2|3  -4.1977     0.3375 -12.439
3|4  -3.0366     0.3334  -9.108
4|5  -0.6370     0.3276  -1.944
```

```
clmm_treat.period.subject.question <- clmm(
    Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question),
    data = df_clean
)
summary(clmm_treat.period.subject.question)
```

```
Cumulative Link Mixed Model fitted with the Laplace approximation

formula: Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat |
    Question)
data:    df_clean

 link  threshold nobs logLik   AIC     niter      max.grad cond.H
 logit flexible  2980 -3186.06 6398.11 1026(8128) 1.80e-04 6.1e+02

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 Subject  (Intercept) 3.1372   1.7712
          TreatB      5.4601   2.3367   -0.552
 Question (Intercept) 0.4474   0.6689
```

```
          TreatB      1.8621   1.3646   -0.471
Number of groups:  Subject 87,  Question 18

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
TreatB         -3.1435     0.5358  -5.867 4.43e-09 ***
Period2        -0.6255     0.4028  -1.553    0.120
TreatB:Period2  0.5590     0.6615   0.845    0.398
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2  -6.7481     0.3585 -18.821
2|3  -4.3947     0.3411 -12.884
3|4  -3.1106     0.3362  -9.252
4|5  -0.5610     0.3298  -1.701
```

Un modelo más simple que el anterior que podemos considerar es eliminar la pendiente del subtitulado en el efecto aleatorio `Subject`.

```
clmm_treat.period_subject_question <- clmm(
    Response ~ Treat * Period + (1 | Subject) + (1 | Question),
    data = df_clean
)
summary(clmm_treat.period_subject_question)
```

```
Cumulative Link Mixed Model fitted with the Laplace approximation

formula: Response ~ Treat * Period + (1 | Subject) + (1 | Question)
data:    df_clean

 link  threshold nobs logLik   AIC     niter     max.grad cond.H
 logit flexible  2980 -3559.48 7136.96 987(3952) 5.50e-04 5.9e+02

Random effects:
 Groups   Name        Variance Std.Dev.
 Subject  (Intercept) 1.4348   1.1978
 Question (Intercept) 0.3394   0.5826
Number of groups:  Subject 87,  Question 18

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
TreatB         -2.5366     0.2815  -9.012   <2e-16 ***
Period2        -0.6203     0.2795  -2.219   0.0265 *
TreatB:Period2  0.5958     0.5353   1.113   0.2657
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2  -5.3264     0.2655 -20.059
2|3  -3.5644     0.2530 -14.089
3|4  -2.5690     0.2488 -10.325
4|5  -0.5243     0.2434  -2.154
```

E incluso eliminar completamente el efecto aleatorio `Subject` y mantener solo las preguntas como efecto aleatorio.

```
clmm_treat.period_question <- clmm(
    Response ~ Treat * Period + (1 | Question),
```

```
    data = df_clean
)
summary(clmm_treat.period_question)
```

```
Cumulative Link Mixed Model fitted with the Laplace approximation

formula: Response ~ Treat * Period + (1 | Question)
data:    df_clean

 link  threshold nobs logLik   AIC     niter     max.grad cond.H
 logit flexible  2980 -3883.04 7782.07 741(2226) 1.08e-04 1.3e+02

Random effects:
 Groups   Name        Variance Std.Dev.
 Question (Intercept) 0.2274   0.4769
Number of groups:  Question 18

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
TreatB         -2.04502    0.10378 -19.706  < 2e-16 ***
Period2        -0.50200    0.09929  -5.056 4.29e-07 ***
TreatB:Period2  0.43361    0.13747   3.154  0.00161 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2  -4.4008     0.1610 -27.337
2|3  -2.8298     0.1446 -19.575
3|4  -1.9983     0.1401 -14.259
4|5  -0.3609     0.1349  -2.675
```

Mediante el test de razón de verosimilitud podemos seleccionar el modelo con menor función de verosimilitud:

```
anova(
    clmm_subject,
    clmm_treat.subject,
    clmm_treat.period.subject,
    clmm_treat.period.subject_question,
    clmm_treat.period.subject.question,
    clmm_treat.period_subject_question,
    clmm_treat.period_question
)
```

```
Likelihood ratio tests of cumulative link models:

                                  formula:
clmm_subject                      Response ~ (1 | Subject)
clmm_treat.subject                Response ~ Treat + (1 + Treat | Subject)
clmm_treat.period_question        Response ~ Treat * Period + (1 | Question)
clmm_treat.period_subject_question Response ~ Treat * Period + (1 | Subject) + (1 | Question)
clmm_treat.period.subject         Response ~ Treat * Period + (1 + Treat | Subject)
clmm_treat.period.subject_question Response ~ Treat * Period + (1 + Treat | Subject) + (1 | Question)
clmm_treat.period.subject.question Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Ques
                                  link: threshold:
clmm_subject                      logit flexible
clmm_treat.subject                logit flexible
clmm_treat.period_question        logit flexible
clmm_treat.period_subject_question logit flexible
clmm_treat.period.subject         logit flexible
clmm_treat.period.subject_question logit flexible
```

```
clmm_treat.period.subject.question logit flexible

                                  no.par    AIC  logLik LR.stat df Pr(>Chisq)
clmm_subject                           5 8117.2 -4053.6
clmm_treat.subject                     8 6878.5 -3431.3 1244.69  3  < 2.2e-16
clmm_treat.period_question             8 7782.1 -3883.0 -903.54  0
clmm_treat.period_subject_question     9 7137.0 -3559.5  647.12  1  < 2.2e-16
clmm_treat.period.subject             10 6879.8 -3429.9  259.20  1  < 2.2e-16
clmm_treat.period.subject_question    11 6640.1 -3309.0  241.69  1  < 2.2e-16
clmm_treat.period.subject.question    13 6398.1 -3186.1  245.96  2  < 2.2e-16

clmm_subject
clmm_treat.subject                 ***
clmm_treat.period_question
clmm_treat.period_subject_question ***
clmm_treat.period.subject          ***
clmm_treat.period.subject_question ***
clmm_treat.period.subject.question ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Vemos que el modelo más complejo, `clmm_treat.period.subject.question`, presenta una menor funcion de verosimilitud. Este modelo tiene un AIC menor que los modelos ordinales ajustados en el apartado anterior incluso si a esos modelos se les añade como factor predictor `Question`. En el Tabla 8.2 se muestran los interceptores y pendientes estimadas para el efecto aleatorio `Question`.

Cuadro 8.2: Intercepto y pendiente de Question en el modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question)

	(Intercept)	TreatB
Q18	0.3933757	-1.1021051
Q01	0.2726623	0.4752429
Q02	0.6197324	-0.3813147
Q03	0.0640464	0.5063518
Q04	0.5020005	1.9203520
Q05	0.7498895	-2.8118058
Q06	0.5528685	-0.8949249
Q07	-0.0106860	-0.4533266
Q08	-0.1980169	-0.8402323
Q09	0.0184714	-2.4698505
Q10	0.0228628	0.1615350
Q11	0.0782894	0.6223213
Q12	-0.2003100	0.3403381
Q13	0.5516281	1.5330216
Q14	-0.2392607	-0.4339893
Q15	-1.3380683	1.4804380
Q16	-1.3655878	1.7228905
Q17	-1.0044586	1.1760588

Las preguntas Q16, Q15, Q17, Q05, Q02 son las 5 cuyo `log odds` del intercepto tiene un valor mayor valor absoluto y, por lo tanto, las que nuestro modelo considera más diferentes del resto. Por otro lado, las preguntas Q05, Q09, Q04, Q16, Q13 son las que mayor valor absoluto tienen en el coeficiente `TreatB` y, por ello, las que presentan mayor diferencia entre tratamientos. En la Figura 8.6 se muestran las predicciones del modelo.

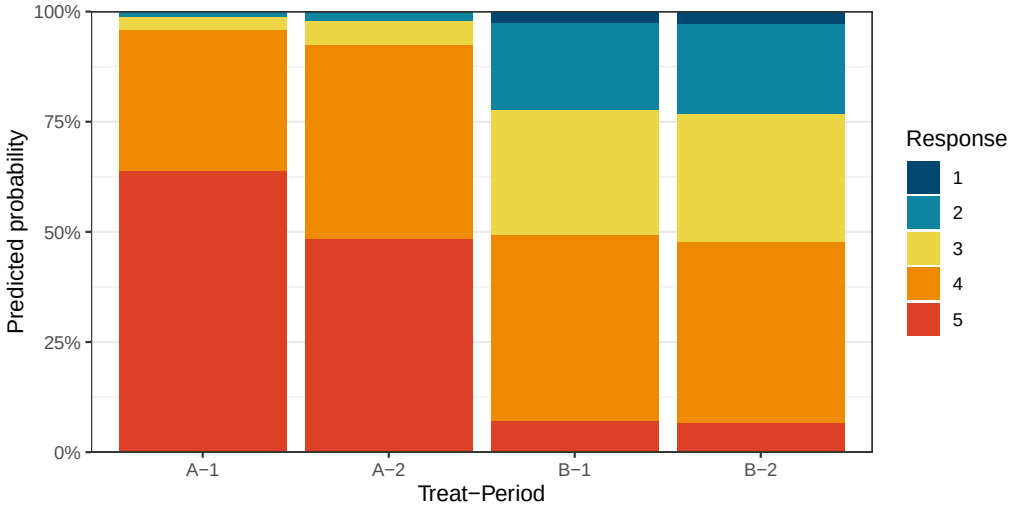


Figura 8.6: Probabilidades de respuesta para el modelo ordinal Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question)

8.3 Modelado Bayesiano.

En el apéndice se comparan diversas parametrizaciones de modelado bayesiano utilizando la función `brm()` del paquete `brms`. Analizamos aquí la que mejor resultado resultado produjo en la aproximación bayesiana a validación cruzada `leave-one-out`:

```
brm_treat.period.subject.question <- brm(
    Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question),
    data = df_clean,
    family = cumulative("logit"),
    sample_prior = TRUE,
    file = "models/brm_treat.period.subject.question",
    file_refit = "on_change"
)
```

Esta parametrización coincide con la que elegimos en el apartado de Regresión Ordinal con efectos mixtos. El modelo utiliza como factores con efectos fijos (`complete pooling` en terminología bayesiana) el nivel de subtitulado y el periodo y la interacción entre ambos; y como efectos aleatorios (`partial pooling`) los sujetos y las preguntas del test. Cada uno de ellos con un intercepto y un nivel de subtitulado variable. El resumen del modelo es el siguiente:

```
summary(brm_treat.period.subject.question)
```

```
 Family: cumulative
  Links: mu = logit; disc = identity
Formula: Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question)
   Data: df_clean (Number of observations: 2980)
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Group-Level Effects:
~Question (Number of levels: 18)
                     Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
sd(Intercept)            0.74      0.16     0.50     1.11 1.00     1213
sd(TreatB)               1.50      0.29     1.05     2.21 1.00     1194
cor(Intercept,TreatB)   -0.41      0.21    -0.74     0.07 1.01      931
                     Tail_ESS
sd(Intercept)            2155
sd(TreatB)               1803
cor(Intercept,TreatB)    1691

~Subject (Number of levels: 87)
                     Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
sd(Intercept)            1.82      0.18     1.50     2.21 1.00      841
sd(TreatB)               2.39      0.21     2.01     2.84 1.00      685
cor(Intercept,TreatB)   -0.54      0.09    -0.70    -0.35 1.01      420
                     Tail_ESS
sd(Intercept)            1266
sd(TreatB)               1487
cor(Intercept,TreatB)     719

Population-Level Effects:
              Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept[1]     -6.70      0.37    -7.43    -5.98 1.00      701     1417
Intercept[2]     -4.35      0.35    -5.03    -3.64 1.00      641     1231
Intercept[3]     -3.06      0.34    -3.73    -2.37 1.00      626     1271
Intercept[4]     -0.50      0.34    -1.17     0.18 1.00      623     1227
TreatB           -3.15      0.57    -4.28    -2.04 1.01      483     1112
Period2          -0.58      0.41    -1.41     0.21 1.01      492      943
TreatB:Period2    0.50      0.68    -0.82     1.83 1.00      428      888

Family Specific Parameters:
     Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
disc     1.00      0.00     1.00     1.00   NA       NA       NA

Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
and Tail_ESS are effective sample size measures, and Rhat is the potential
scale reduction factor on split chains (at convergence, Rhat = 1).
```

En el Tabla 8.3 se comparan las estimaciones puntuales que obtuvimos para este modelo con la función `clmm`. Se comprueba que son muy similares. También se añaden los intervalos de confianza. Vemos que los interceptores son claramente significativos y también el coeficiente de `TreatB`. Sin embargo los coeficientes correspondientes al efecto periodo, `Period2`, y al efecto secuencia, `TreatB:Period`, incluyen el cero y además tienen intervalos muy grandes por lo que hay mucha incertidumbre respecto a su verdadero valor.

Cuadro 8.3: Comparación frecuentista/bayesiano de coeficientes estimados en el modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question).

	ordinal::clmm			brms::brm			
Name	Estimation.clmm	2.5%	97.5%	Estimation.brm	std.error	conf.low	conf.high
1|2	-6.75	-7.45	-6.05	-6.71	0.37	-7.43	-5.98

2|3	-4.39	-5.06	-3.73	-4.35	0.35	-5.03	-3.64
3|4	-3.11	-3.77	-2.45	-3.06	0.34	-3.73	-2.37
4|5	-0.56	-1.21	0.09	-0.50	0.34	-1.17	0.18
TreatB	-3.14	-4.19	-2.09	-3.14	0.57	-4.28	-2.04
Period2	-0.63	-1.42	0.16	-0.57	0.41	-1.41	0.21
TreatB:Period2	0.56	-0.74	1.86	0.52	0.68	-0.82	1.83
Subject.sd(Intercept)	0.67			0.72	0.16	0.50	1.11
Subject.sd(TreatB)	1.36			1.47	0.29	1.05	2.21
Subject.cor(Intercept,TreatB)	1.77			1.82	0.18	1.50	2.21
Question.sd(Intercept)	2.34			2.38	0.21	2.01	2.84
Question.sd(TreatB)	-0.47			-0.44	0.21	-0.74	0.07
Question.cor(Intercept,TreatB)	-0.55			-0.54	0.09	-0.70	-0.35

No hemos dado valor a las distribuciones de probabilidad a priori con fiando que los valores por defecto que asigna `brm` son adecuados. En el Tabla 8.4 se muestran las distribuciones a priori de los parámetros aleatorios del modelo. En la Figura 8.7 se constata que toman valores razonables y no informativos.

Cuadro 8.4: Distribuciones a priori del modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question).

prior	class	coef	group	resp	dpar	nlpar	lb	ub	source
	b								defaul
	b	Period2							defaul
	b	TreatB							defaul
	b	TreatB:Period2							defaul
student_t(3, 0, 2.5)	Intercept								defaul
	Intercept	1							defaul
	Intercept	2							defaul
	Intercept	3							defaul
	Intercept	4							defaul
lkj_corr_cholesky(1)	L								defaul
	L		Question						defaul
	L		Subject						defaul
student_t(3, 0, 2.5)	sd						0		defaul
	sd		Question						defaul
	sd	Intercept	Question						defaul
	sd	TreatB	Question						defaul
	sd		Subject						defaul
	sd	Intercept	Subject						defaul
	sd	TreatB	Subject						defaul

Es importante asegurar que el entrenamiento del a convergido a su distribución a posteriori. En la tabla de resumen constatamos que el valor de `Rhat` es inferior a 1.1 y el de `ESS` superior a 400 en todos los parámetros, que son umbrales que no se deberían violar (**ver**). En la Figura 8.8 se comprueba que las cadenas de muestrado se mezclan correctamente y no muestran autocorrelación en ninguno de las parámetros. Por último, en la Figura 8.9 se muestra un a comparación de los histogramas obtenidos de modelo con los intervalos de confianza marginales de la función predictiva a posteriori del modelo. En la mayoría de las preguntas, el muestreo reproduce bastante bien el histograma en casi todas las preguntas

excepto en algunas como la `Q16` o la `Q17`.

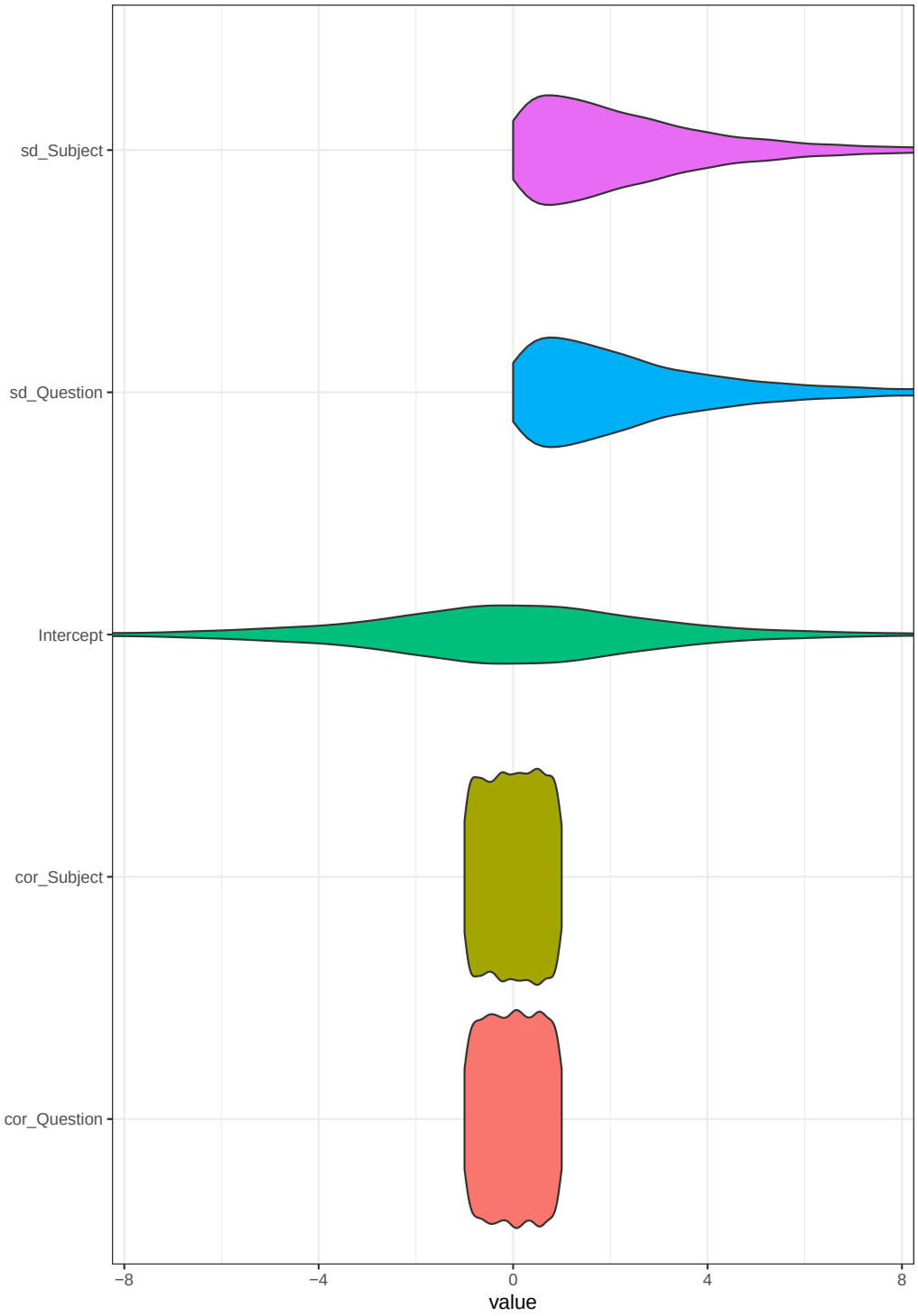


Figura 8.7: Distribuciones a priori del modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question).

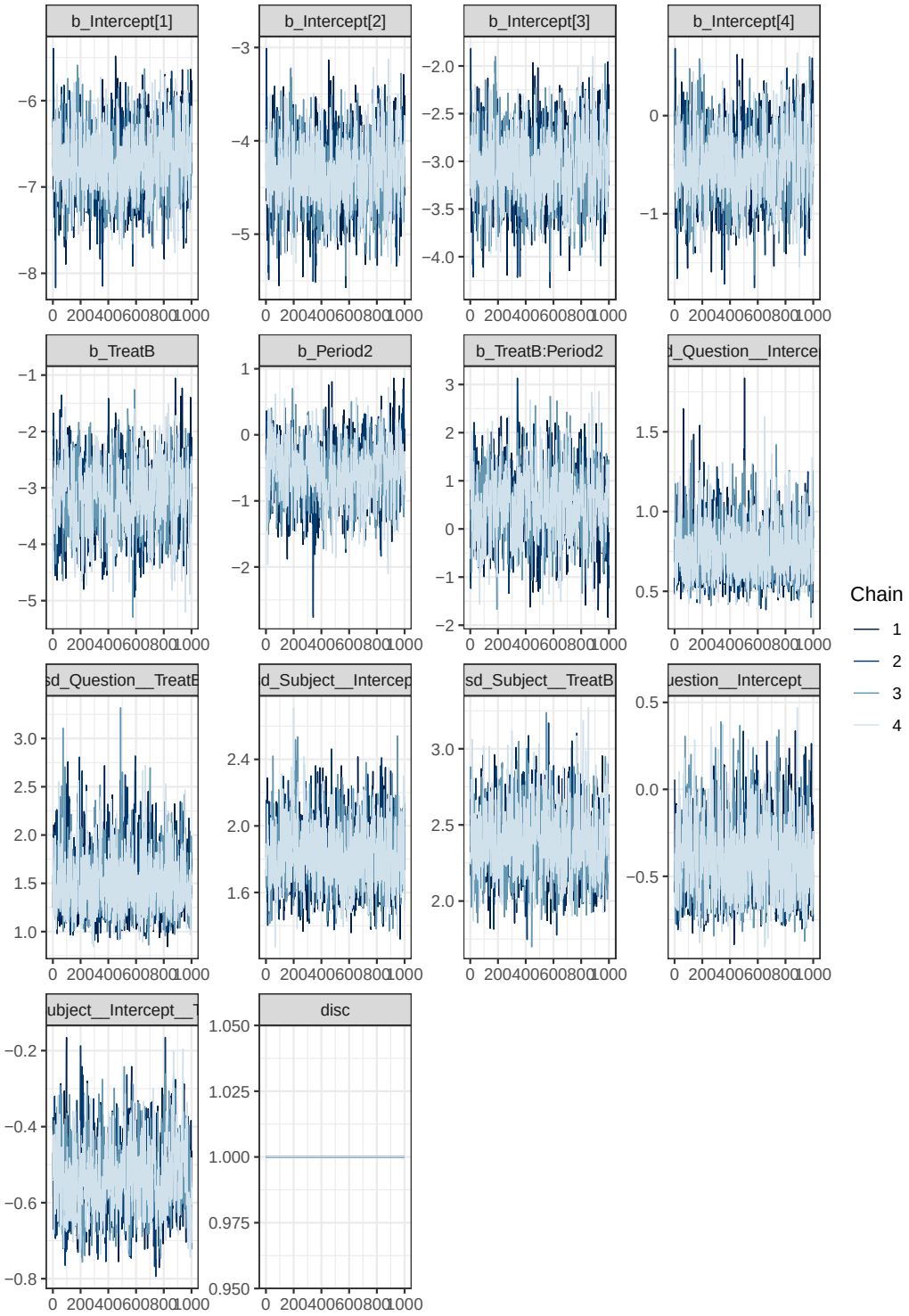


Figura 8.8: MCMC trazado del modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question).

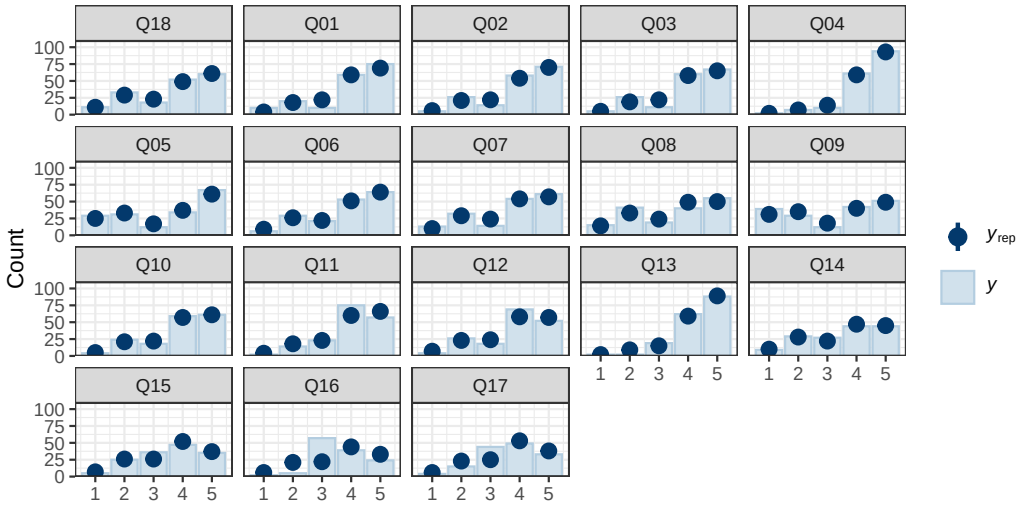


Figura 8.9: Comparación de los valores reales con los obtenidos a partir de la función predictiva a posteriori del modelo Response ~ Treat * Period + (1 + Treat | Subject) + (1 + Treat | Question).

Capítulo 9

Resultados

Capítulo 10

Conclusiones y trabajo futuro

Referencias

AENOR (2012). *UNE 153010 Subtitulado para personas sordas y personas con discapacidad auditiva*. Asociación Española de Normalización y Certificación.

Agresti, A. (2010). *Analysis of Ordinal Categorical Data*. DOI: 10.1002/9780470594001.

– (oct. de 2018). *An introduction to categorical data analysis, 3rd Edition*. URL: https://www.wiley.com/en-us/An+Introduction+to+Categorical+Data+Analysis%2C+3rd+Edition-p-9781119405283.

Bruin, J. (2011). *How do I interpret the coefficients in an ordinal logistic regression in R*. URL: https://stats.oarc.ucla.edu/r/faq/ologit-coefficients.

Bürkner, P.-C. (nov. de 2021). «Bayesian Item Response Modeling in R with brms and Stan». En: *Journal of Statistical Software* 100. DOI: 10.18637/jss.v100.i05.

Bürkner, P.-C. y M. Vuorre (feb. de 2019). «Ordinal Regression Models in Psychology: A Tutorial». En: *Advances in Methods and Practices in Psychological Science* 2, pág. 251524591882319. DOI: 10.1177/2515245918823199.

Chen, D.-G. y J. Chen (ene. de 2021). *Statistical Regression Modeling with R: Longitudinal and Multi-level Modeling*. DOI: 10.1007/978-3-030-67583-7.

Christensen, R. H. B. (2022). *ordinal—Regression Models for Ordinal Data*. R package version 2022.11-16. https://CRAN.R-project.org/package=ordinal.

Christensen, R. H. B. (2018). «Cumulative Link Models for Ordinal Regression with the R Package ordinal». En.

Friendly, M. (dic. de 2015). *Classification and regression trees*.

Friendly, M., D. Meyer y A. Zeileis (dic. de 2015). *Discrete Data Analysis with R: Visualization and Modeling Techniques for Categorical and Count Data*, págs. 1-525. DOI: 10.1201/b19022.

Gelman, A., J. Carlin, H. Stern, D. Dunson, A. Vehtari y D. Rubin (nov. de 2013). *Bayesian Data Analysis*. DOI: 10.1201/b16018.

Guerra, A., T. Gidel y E. Vezzetti (mayo de 2016). «Toward a common procedure using likert and likert-type scales in small groups comparative design observations». En.

Harrell, F. (ene. de 2015). *Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis*. DOI: 10.1007/978-3-319-19425-7.

Lawson, J. (2015). Ed. por Chapman y Hall/CRC. DOI: 10.1201/b17883. URL: https://www.taylorfrancis.com/books/mono/10.1201/b17883/design-analysis-experiments-john-lawson.

Levshina, N. (2020). «Conditional Inference Trees and Random Forests». En: *A Practical Handbook of Corpus Linguistics*. Ed. por M. Paquot y S. T. Gries. Cham: Springer International Publishing, págs. 611-643. DOI: 10.1007/978-3-030-46216-1_25. URL: https://doi.org/10.1007/978-3-030-46216-1_25.

Liddell, T. M. y J. K. Kruschke (2018). «Analyzing ordinal data with metric models: What could possibly go wrong?» En: *Journal of Experimental Social Psychology* 79, págs. 328-348. DOI: 10.1016/j.jesp.2018.08.009. URL: https://www.sciencedirect.com/science/article/pii/S0022103117307746.

Liu, X. (abr. de 2022). *Categorical Data Analysis and Multilevel Modeling Using R*. Ed. por S. P. Ltd.

Lui, K.-J. (ago. de 2016). *Crossover Designs: Testing, Estimation, and Sample Size*. DOI: 10.1002/9781119114710.

Molanes-López, E. M., A. Rodriguez-Ascaso, E. Letón y J. Pérez-Martín (2021). «Assessment of Video Accessibility by Students of a MOOC on Digital Materials for All». En: *IEEE Access* 9, págs. 72357-72367. DOI: 10.1109/ACCESS.2021.3079199.

O'Connell, A. (ene. de 2006). *Logistic Regression Models for Ordinal Response Variables*. DOI: 10.4135/9781412984812.

Pérez Martín, J., A. Rodríguez-Ascaso y E. Molanes-López (nov. de 2021). «Quality of the captions produced by students of an accessibility MOOC using a semi-automatic tool». En: *Universal Access in the Information Society* 20. DOI: 10.1007/s10209-020-00740-9.

Schweinberger, M. (2020). *Questionnaires and Surveys: Analyses with R*. 2020/12/11. https://slcladal.github.io/survey.html. The University of Queensland, Australia. School of Languages y Cultures. Brisbane.

Senn, S. (2022). Ed. por L. John Wiley. DOI: 10.1002/0470854596.

Venables, W. N. y B. D. Ripley (2002). *Modern Applied Statistics with S*. Fourth. ISBN 0-387-95457-0. New York: Springer. URL: https://www.stats.ox.ac.uk/pub/MASS4/.

Yee, T. W. (2023). *VGAM: Vector Generalized Linear and Additive Models*. R package version 1.1-8. URL: https://CRAN.R-project.org/package=VGAM.

Apéndice A

Preprocesado de los ficheros suministrados.

Este es el código en R con el que se transforman los ficheros que se suministran (ver Sección 4.2).

```r
library(readr)
library(purrr)
library(dplyr)
library(magrittr)
library(stringr)
library(forcats)
library(testit)
library(tidyr)

##### GRADE #####
## Usuarios que no quieren participar
no_want_users <- read_lines("data/original/ids_a_eliminar.txt")

# Leemos todos los archivos de grade CSV
grade_files <- list.files(
    "data/original", pattern = ".*grade.*.csv", full.names = TRUE
)

grade_df <- map_dfr(
    grade_files, ~ read_delim(.x, delim = ";", show_col_types = FALSE) %>%
        # Añadimos el número de fila para mantener la trazabilidad
        mutate(Userid = row_number() + 1) %>%
        # Movemos las columnas de identificación de fila a la primera posición
        relocate(Userid, .before = 2) %>%
        # Renombramos las columnas para que empiecen con mayúsculas
        rename_with(~ str_to_title(.), everything()) %>%
        # Renombramos para que sea más fácil procesar el campo Cohort Name
```

```
        rename("Cohort" = "Cohort Name") %>%
        # Eliminamos valores nulos y los que no quieren participar
        filter(!is.na(Cohort) & !Username %in% no_want_users)
)

assert("Comprobamos que no hay usuarios duplicados", grade_df %>%
    nrow() == grade_df %>%
    distinct(Username) %>%
    nrow())

# Creamos un tibble que tiene un campo con letras en lugar del valor de Cohorte
(groups <- grade_df %>%
    distinct(Cohort) %>%
    arrange(Cohort) %>%
    mutate(Group = LETTERS[1:n()]))

# Unimos los tibbles para asignar en grupo como letra en lugar de la cohorte
grade_df <- left_join(grade_df, groups) %>% dplyr::select(Username, Userid, Group)

##### PROFILE #####
profile_files <- list.files(
    "data/original", pattern = ".*student_profile.*.csv", full.names = TRUE
)

profile_df <- map_dfr(
    profile_files, ~ read_delim(.x, delim = ";", show_col_types = FALSE)
)

grade_df <- left_join(
    grade_df, profile_df %>% dplyr::select(-cohort), by = join_by(Username == username)
)


##### CONOC #####
conoc_files <- list.files(
    "data/original", pattern = ".*conoc.*.csv", full.names = TRUE)

conoc_df <- map_dfr(
    conoc_files, ~ read_delim(.x, delim = ";", show_col_types = FALSE)
)

conoc_df <- conoc_df %>%
    filter(Tries == 1) %>%
    rowwise() %>%
    mutate(
        level_of_knowledge =
            sum(c_across(starts_with(paste("Q", 1:10, "C", sep = ""))) == "correct")
    ) %>%
    dplyr::select(User, level_of_knowledge)

grade_df <- left_join(grade_df, conoc_df, by = join_by(Username == User))

##### TEST #####
# Leemos todos los archivos de test CSV
```

```r
test_files <- list.files(
    "data/original", pattern = ".*test.*.csv", full.names = TRUE
)

# Leer todos los archivos de test y los combinamos en un dataframe
test_df <- map_dfr(
    test_files, ~ read_delim(.x, delim = ";", show_col_types = FALSE) %>%
        # Añadimos un número de fila para mantener la trazabilidad
        mutate(Row = row_number() + 1) %>%
        # Añadimos la columna del número de test
        mutate(Test = sprintf("%02d", as.integer(str_extract(.x, "(?<=test)\\d+")))) %>%
        # Movemos las columnas de identificación de test y fila a la primera posición
        relocate(c(Test, Row), .before = 2)
) %>%
    # eliminamos los usuarios que no quieren participar
    filter(!User %in% no_want_users)

num_questions <- 18

# Nombre de los campos que contienen las respuestas al test
questions_original <- paste(
    "Q", seq(from = 1, by = 2, length.out = num_questions), "R", sep = ""
)

# Nombre de los campos que contienen las respuestas al test
comments_original <- paste(
    "Q", seq(from = 2, by = 2, length.out = num_questions - 1), "R", sep = ""
)

# Nombre de los campos que se usarán para renombrar los campos de respuesta al test
questions <- sprintf("Q%02d", seq(from = 1, by = 1, length.out = num_questions))
comments <- sprintf("C%02d", seq(from = 1, by = 1, length.out = num_questions - 1))
columns <- c(
    "Row", "Test", "User", "LastTry", questions_original, comments_original
)

# Procesamos el dataframe
# Con este operador del paquete magrittr hacemos las transformaciones in situ
test_df %<>%
    # Eliminamos las filas que no contienen información
    filter(Tries > 0) %>%
    # Convertimos LastTry a formato fecha
    mutate(LastTry = strptime(LastTry, format = "%Y-%m-%dT%H:%M:%SZ")) %>%
    # Seleccionamos las columnas que nos interesan
    dplyr::select(all_of(columns)) %>%
    # Extraemos la puntuación numérica de la pregunta
    mutate(across(questions_original, ~ if_else(
        startsWith(.x, "choice_"), as.integer(str_extract(.x, "\\d+")), NA_integer_)
    )) %>%
    # Renombramos los respuestas para que sean secuenciales
    rename(
        setNames(questions_original, questions),
        setNames(comments_original, comments)
    ) %>%
```

```
    # nos aseguramos de que el orden filas es el mismo que el de los ficheros.
    arrange("Test", "Row")

# Guardamos el número de filas para posterior comprobación
n_test <- test_df %>% nrow()

# Unimos los dataframes para tener el grupo y el UserID secuencial
test_df <- inner_join(
    test_df, grade_df, by = join_by(User == Username)
    ) %>% relocate(Group, .before = 2)

# Cambiamos los valores del campo User por los del UserID
test_df %<>%
    mutate(User = Userid) %>%
    dplyr::select(-Userid) %>%
    arrange(User, Test) # Ordenamos por usuario y test


##### CHECKS #####
assert(
    "Comprobamos que no hay preguntas duplicadas en el dataframe de test",
    n_test == test_df %>%
    distinct(Group, Test, User) %>%
    nrow()
)

assert(
    "Comprobamos que no hay valores nulos",
    test_df %>%
    dplyr::select(
        -c(comments, year_of_birth, gender, level_of_education, level_of_knowledge)
    ) %>% filter(if_any(everything(), is.na)) %>% nrow() == 0)


assert(
    "Comprobamos que no hay respuestas con valores incorrectos",
    sum(sort(unique(unlist(
        test_df %>% dplyr::select(all_of(questions))
    ))) == 0:5) == 6)


comments_df <- test_df %>%
    pivot_longer(
        cols = starts_with(c("Q", "C")),
        names_to = c(".value", "Question"),
        names_pattern = "(Q|C)(.*)") %>%
    rename(Response = Q, Comment = C) %>%
    filter(!is.na(Comment) & grepl("[a-zA-Z]", Comment)) %>%
    dplyr::select(Test, Row, Group, User, Question, Response, Comment) %>%
    arrange(Test, Group, Response, Row)


write_csv(comments_df, "./data/preprocess/comments.csv")
```

```
##### SAVE TO FILE #####
write_csv(
    test_df %>% dplyr::select(-all_of(comments)), "./data/preprocess/test_all.csv"
)
```

Apéndice B

Creación de los dataframes df_all y df_clean.

Código que transforma los datos preprocesados (ver Apéndice A) en los dataframes que se usan en el análisis estadístico.

```
# Leemos el tibble preprocesado
test_all_df <- read_delim(
    "./data/preprocess/test_all.csv",
    delim = ",", show_col_types = FALSE
)

# Eliminamos aquellos usuarios que no han hecho uno de los test
test_df <- test_all_df %>%
    group_by(User) %>%
    mutate(Rows = n()) %>%
    filter(Rows > 1) %>%
    ungroup()

##### SAVE TO FILE #####
write_csv(test_df, "./data/preprocess/test.csv")

df <- test_df %>%
    mutate(
        Period = as.factor(
            if_else(Test == "01", 1, 2)
        ),
        Treat = as.factor(
            if_else(Group == "A" & Test == "01" | Group == "B" & Test == "02", "A", "B")
        ),
        Seq = as.factor(
            if_else(Group == "A", "AB", "BA")
        ),
        Subject = as.factor(User)
    ) %>%
    dplyr::select(
        Seq, Period, Treat, Subject,
        gender, year_of_birth, level_of_education, starts_with("Q")
    ) %>%
```

```
    mutate_at(
        vars(starts_with("Q")), ~ (. + 1) %% 6
    ) %>%
    pivot_longer(
        cols = all_of(starts_with("Q")),
        names_to = "Question",
        values_to = "Response"
    ) %>%
    mutate(
        Question = relevel(as.factor(Question), ref = "Q18"),
        Response = factor(Response, ordered = TRUE)
    ) %>%
    arrange(Subject, Period, Question)

response_labels <- c(
    "No sé / No contesto",
    "Muy en desacuerdo",
    "En desacuerdo",
    "Neutral",
    "De acuerdo",
    "Muy de acuerdo"
)

question_labels <- c(
    "Los subtítulos del vídeo cumplen en general con los requisitos de accesibilidad.",
    "La posición de los subtítulos.",
    "El número de líneas por subtítulo.",
    "La disposición del texto respecto a la caja donde se muestran los subtítulos.",
    "El contraste entre los caracteres y el fondo.",
    "La corrección ortográfica y gramatical.",
    "La literalidad.",
    "La identificación de los personajes.",
    "La asignación de líneas a los personajes en los diálogos.",
    "La descripción de efectos sonoros.",
    "La sincronización de las entradas y salidas de los subtítulos.",
    "La velocidad de exposición de los subtítulos.",
    "El máximo número de caracteres por línea.",
    "La legibilidad de la tipografía.",
    "La separación en líneas diferentes de sintagmas nominales, verbales y preposicionales.",
    "La utilización de puntos suspensivos.",
    "La escritura de los números.",
    "Las incorrecciones en el habla."
)

question_labels_reduced <- c(
    "Valoración general",
    "Posición",
    "Número líneas",
    "Texto dentro caja",
    "Contraste",
    "Corrección",
    "Literalidad",
    "Identificación personajes",
    "Líneas/personajes",
    "Efectos sonoros",
    "Sincronización",
    "Velocidad",
    "Caracteres x línea",
    "Tipografía",
    "Separación sintagmas",
    "Puntos suspensivos",
    "Escritura números",
    "Incorrecciones habla"
)
```

```r
df <- df %>% mutate(
    Response_v = as.numeric(Response) - 1,
    Response_l = ordered(Response_v, labels = response_labels),
    Question_l = factor(Question, labels = question_labels),
    Question_lr = factor(Question, labels = question_labels_reduced)
)

dist <- df %>%
    xtabs(~ Question + Response, data = .) %>%
    dist(x = ., method = "euclidean")

cluster <- hclust(dist, method = "complete")
cuts <- factor(cutree(cluster, k = 3))

# Añadimos la columna cluster al dataframe
df <- inner_join(
    df,
    data.frame(
        Question = factor(names(cuts),
            levels = levels(df$Question)
        ),
        Cluster = as.factor(cuts)
    ),
    by = "Question"
)

write_csv(df, "./data/preprocess/test_lg.csv")
df_all <- df

df_all$Y <- model.matrix(~ Response - 1, data = df_all)

df_clean <- df %>% filter(Response != 0)
df_clean <- df_clean %>% mutate(
    Response = factor(Response, levels = levels(Response)[-1]),
    Response_l = ordered(Response_l, levels = levels(Response_l)[-1]),
    Level = as.ordered(
        ifelse(
            Response %in% c(1, 2),
            "Negative",
            ifelse(
                Response %in% c(4, 5),
                "Positive",
                "Neutral"
            )
        )
    )
)
df_0 <- df %>% filter(Response == 0)
```

Apéndice C

Efecto secuencia e interacción tratamiento vs. periodo.

Vamos a demostrar que el efecto secuencia es equivalente a la interacción de los factores tratamiento y periodo.

C.1 Preparación.

Partimos del siguiente conjunto de datos generado aleatoriamente [1]:

```
set.seed(100)
n <- 1000
df <- data.frame(
    Response = rnorm(n),
    Treat = as.factor(sample(c("A", "B"), n, replace = TRUE)),
    Period = as.factor(sample(c(1, 2), n, replace = TRUE))
)

df$Seq <- as.factor(
    ifelse(
        df$Period == 1 & df$Treat == "A" | df$Period == 2 & df$Treat == "B",
        "AB",
        "BA"
    )
)

head(df, 10)
```

```
     Response Treat Period Seq
1 -0.50219235     B      2  AB
2  0.13153117     A      1  AB
```

[1] Obsérvese que se la variable `Response` en esta simulación es cuantitativa y no ordinal. Se ha realizado de esta forma para poder usar un ajuste de mínimos cuadrados en lugar de una regresión ordinal para facilitar el cálculo y su interpretación.

```
3  -0.07891709     A      2  BA
4   0.88678481     A      2  BA
5   0.11697127     A      1  AB
6   0.31863009     A      2  BA
7  -0.58179068     A      2  BA
8   0.71453271     A      1  AB
9  -0.82525943     B      2  AB
10 -0.35986213     B      1  BA
```

Calculamos las medias por cada nivel de factor y combinaciones de niveles que utilizaremos luego en la interpretación de los coeficientes de los modelos

```
M <- mean(df$Response) # 1 media de respuesta global

# 2 medias de respuesta para tratamientos A y B
mTreat <- with(df, tapply(Response, Treat, mean))

# 2 medias de respuesta para periodos 1 y 2
mPeriod <- with(df, tapply(Response, Period, mean))

# 2 medias de respuesta para secuencias AB y BA
mSeq <- with(df, tapply(Response, Seq, mean))

# 4 medias de respuesta para las cuatro combinaciones de tratamiento y periodo
m2 <- with(df, tapply(Response, list(Treat, Period), mean))

dTreat <- diff(mTreat) # diferencia de medias entre tratamientos A y B

dPeriod <- diff(mPeriod) # diferencia de medias entre periodos 1 y 2

d2 <- diff(m2) # diferencias entre niveles de tratamiento en cada nivel de periodo
```

C.2 Análisis con un solo factor (tratamiento).

```
l1 <- lm(Response ~ Treat, df)
data.frame(t(coef(l1))) %>% gt()
```

Cuadro C.1: Ajuste del modelo Response ~ Treat con contrasts treatment.

X.Intercept.	TreatB
0.03624217	-0.03966751

Vemos que el intercepto es la media de la respuesta en el nivel de tratamiento *A*:

```
mTreat[1]
```

```
         A
0.03624217
```

Que la pendiente (parámetro $TreatB$) es la diferencia entre las medias tratamientos:

```
dTreat
```

```
          B
-0.03966751
```

Por ello, para conocer el efecto del tratamiento en el nivel *B* hay que sumar intercepto y pendiente:

```
coef(l1)[[1]] + coef(l1)[[2]] - mTreat[[2]]
```

```
[1] 1.214306e-16
```

Esto es así ya que por defecto R utiliza el contraste conocido como `codificación de tratamiento`:

```
contr.treatment(2)
```

```
  2
1 0
2 1
```

Podemos ver la matriz ampliada añadiendo el intercepto, que siempre será una columna de 1's:

```
model.matrix(~Treat, expand.grid(Treat = c("A", "B")))
```

```
  (Intercept) TreatB
1           1      0
2           1      1
attr(,"assign")
[1] 0 1
attr(,"contrasts")
attr(,"contrasts")$Treat
[1] "contr.treatment"
```

Cada fila representa el nivel del tratamiento (fila 1 nivel *A* y fila 2 nivel *B*) y las columnas representan los parámetros del modelo. Los valores son los niveles de tratamiento (0 ó 1). Para obtener el significado de cada parámetro, multiplicamos el valor del contraste por el parámetro. Así:

- De la primera fila obtenemos que el efecto del tratamiento *A* es el intercepto: $A = 1 \cdot Intercept + 0 \cdot TreatB$.
- De la segunda fila obtenemos que el valor del parámetro $TreatB$ es la diferencia de los niveles de tratamiento. $B = 1 \cdot Intercept + 1 \cdot TreatB \Rightarrow TreatB = B - Intercept$.

Esto quiere decir que existe una variable para codificar el efecto tratamiento, y esta variable tiene el valor 0 para el nivel *A* por ser el de referencia y 1 para el nivel *B*. La pendiente se codifica como la diferencia del efecto de los dos niveles $(B - A)$.

C.3 Análisis con un dos factores (tratamiento y periodo).

```
l2 <- lm(Response ~ Treat * Period, df)
data.frame(t(coef(l2))) %>% gt()
```

Cuadro C.2: Ajuste del modelo Response ~ Treat * Period con contrasts treatment.

X.Intercept.	TreatB	Period2	TreatB.Period2
0.04138614	-0.1076137	-0.01125933	0.1343517

Vemos que el intercepto es la media del tratamiento A en el periodo 1 por ser estos los valores que R usa como referencia [2]:

```
m2["A", "1"]
```

```
[1] 0.04138614
```

El parámetro $TreatB$ es la diferencia de medias entre los tratamientos en el periodo 1:

```
m2["B", "1"] - m2["A", "1"]
```

```
[1] -0.1076137
```

El parámetro $Period2$ es la diferencia de medias entre los periodos en el nivel de tratamiento A:

```
m2["A", "2"] - m2["A", "1"]
```

```
[1] -0.01125933
```

Finalmente, $TreatB : Period2$ es la diferencia entre el segundo periodo y el primero del nivel de tratamiento B menos la diferencia entre periodos del nivel de tratamiento A:

```
m2["B", "2"] - m2["B", "1"] - (m2["A", "2"] - m2["A", "1"])
```

```
[1] 0.1343517
```

La matriz de contraste nos permite razonar por qué esto es así:

```
model.matrix(~ Treat * Period, expand.grid(Treat = c("A", "B"), Period = c("1", "2")))
```

[2] R utiliza como valor de referencia el nivel más bajo de factor.

```
  (Intercept) TreatB Period2 TreatB:Period2
1           1      0       0              0
2           1      1       0              0
3           1      0       1              0
4           1      1       1              1
attr(,"assign")
[1] 0 1 2 3
attr(,"contrasts")
attr(,"contrasts")$Treat
[1] "contr.treatment"

attr(,"contrasts")$Period
[1] "contr.treatment"
```

- La primera fila es el intercepto y corresponde con el tratamiento A y el periodo 1.
- La segunda fila es el efecto del tratamiento B en el periodo 1 y se calcula con la suma del intercepto y el parámetro $TreatB$. Luego $TreatB$ es la diferencia del efecto de los tratamientos en el periodo 1.
- Análogamente con la tercera fila concluimos que $Period2$ es la deferencia entre periodos para el tratamiento A.
- Finalmente, la cuarta fila, es el tratamiento B en el periodo 2 y, por lo tanto, $Treat2 : Period2$ es la diferencia el nivel B de tratamiento y el periodo 2 y el nivel de tratamiento A en el periodo 1, menos la diferencia de niveles de tratamiento para el periodo 1 y menos la diferencia de periodos para el tratamiento A.

Obsérvese que antes hemos calculado de forma diferente $TreatB : Period2$. Podemos aplicar la fórmula anterior y comprobar que produce el mismo resultado:

```
m2["B", "2"] - m2["A", "1"] - (m2["B", "1"] - m2["A", "1"]) - (m2["A", "2"] - m2["A", "1"])
```

```
[1] 0.1343517
```

C.4 Factor secuencia.

Vamos a incorporar la secuencia como factor para ver si es equivalente a la interacción entre periodo y tratamiento. En caso de serlo los coeficientes del modelo ajustado deberían coincidir. Sin embargo vemos que los modelos l2 (Tabla C.2) y l3 (Tabla C.3) tienen distintos coeficientes.

```
l3 <- lm(Response ~ Treat + Period + Seq, df)
data.frame(t(coef(l3))) %>% gt()
```

Cuadro C.3: Ajuste del modelo Response ~ Treat + Period + Seq con contrasts treatment.

X.Intercept.	TreatB	Period2	SeqBA
0.04138614	-0.04043786	0.05591654	-0.06717587

Los coeficientes no coinciden debido a que estamos usando el contraste con `codificación de tratamientos`. Pero si cambiamos a `codificación de sumas`:

```
options(contrasts = rep("contr.sum", 2))
```

Y volvemos a ajustar los modelos que ya usarán el `contraste suma`, podemos comprobar que ahora tienen los mismos coeficientes y el coeficiente $Seq1$ del modelo que incorpora el efecto secuencia (Tabla C.4) es igual que el coeficiente $Treat1 : Period1$ del modelo que incorpora la interacción entre tratamiento y periodo (Tabla C.5). Obsérvese que los nombres de los coeficientes han cambiado respecto al `contraste de tratamiento`. Esto sucede porque la interpretación de los coeficientes varía como se explica a continuación.

```
l4 <- lm(Response ~ Treat + Period + Seq, df)
data.frame(t(coef(l4))) %>% gt()
```

Cuadro C.4: Ajuste del modelo Response ~ Treat + Period + Seq con contrasts sum.

X.Intercept.	Treat1	Period1	Seq1
0.01553755	0.02021893	-0.02795827	0.03358794

```
l5 <- lm(Response ~ Treat * Period, df)
data.frame(t(coef(l5))) %>% gt()
```

Cuadro C.5: Ajuste del modelo Response ~ Treat * Period con contrasts sum.

X.Intercept.	Treat1	Period1	Treat1.Period1
0.01553755	0.02021893	-0.02795827	0.03358794

La interpretación de los contrastes es diferente. Para explicarlo, mostramos la matriz de contraste:

```
model.matrix(~ Treat * Period, expand.grid(Treat = c("A", "B"), Period = c("1", "2")))
```

```
  (Intercept) Treat1 Period1 Treat1:Period1
1           1      1       1              1
2           1     -1       1             -1
3           1      1      -1             -1
4           1     -1      -1              1
attr(,"assign")
[1] 0 1 2 3
attr(,"contrasts")
attr(,"contrasts")$Treat
[1] "contr.sum"

attr(,"contrasts")$Period
[1] "contr.sum"
```

Vemos que ahora los niveles son 1 y -1 [3] en vez de 0 y 1 que se utilizan en el contraste de tratamiento. La interpretación es la siguiente:

- El interceptor es la media de la media de cada uno de los niveles de factor. ¿Por qué?. El interceptor es el valor de la variable de respuesta cuando cuando todas las variables explicativas valen 0. Esto sucede en la media de la variable de respuesta ya que cero es el valor que está en la mitad de +1 y -1. Podemos comprobar que la media global coincide con el interceptor del modelo l4 (Tabla C.4):

```
mean(m2)
```

```
[1] 0.01553755
```

- El coeficiente $Treat1$ es la mitad la diferencia de la media entre niveles de tratamiento ($TreatA - TreatB$). La media de cada tratamiento se calcula como la media del tratamiento en cada periodo.

```
-diff(apply(m2, 1, mean)) / 2
```

```
         B
0.02021893
```

Otra forma de entender el coeficiente $Treat1$ es como la cuarta parte de la diferencia de los efectos de los tratamientos en cada periodo.

```
(m2["A", "1"] + m2["A", "2"] - (m2["B", "1"] + m2["B", "2"])) / 4
```

```
[1] 0.02021893
```

- El coeficiente $Period1$ es la mitad la diferencia de la media entre periodos($Period1 - Period2$). La media entre periodos se calcula como la media del periodo para cada tratamiento.

```
-diff(apply(m2, 2, mean)) / 2
```

```
          2
-0.02795827
```

Otra forma de entender el coeficiente $Period1$ es como la cuarta parte de la diferencia de los efectos del periodo en cada tratamiento.

[3] El nivel de referencia del factor tendrá valor 1 y el otro -1. Por ejemplo, en la variable `Treat`, A tendrá +1 y B tendrá valor -1.

```
(m2["A", "1"] + m2["B", "1"] - (m2["A", "2"] + m2["B", "2"])) / 4
```

```
[1] -0.02795827
```

- El coeficiente $Treat1:Period1$ es el coeficiente $Treat1$ menos la mitad de la diferencia de la media entre tratamientos para el periodo 2 ($TreatA - TreatB$):

```
-diff(apply(m2, 1, mean)) / 2 + diff(m2[, "2"]) / 2
```

```
        B
0.03358794
```

```
coef(l5)[2] + diff(m2[, "2"]) / 2
```

```
    Treat1
0.03358794
```

El coeficiente $Treat1:Period1$ también se puede calcular como $Period1$ menos la mitad de la diferencia de la media entre periodos para el para el tratamiento B ($Period1 - Period2$):

```
-diff(apply(m2, 2, mean)) / 2 + diff(m2["B", ]) / 2
```

```
        2
0.03358794
```

```
coef(l5)[3] + diff(m2["B", ]) / 2
```

```
   Period1
0.03358794
```

Un tercera forma de interpretar el coeficiente $Treat1:Period1$ es como la cuarta parte de la suma de la diferencia cruzada del efecto de cada tratamiento en cada periodo:

```
(m2["A", "1"] - m2["A", "2"] + m2["B", "2"] - m2["B", "1"]) / 4
```

```
[1] 0.03358794
```

O reorganizando los términos de otra forma, sería la cuarta parte de la suma de la diferencia cruzada del efecto de cada periodo en cada tratamiento:

```
(m2["B", "2"] - m2["A", "2"] + m2["A", "1"] - m2["B", "1"]) / 4
```

```
[1] 0.03358794
```

- Podemos obtener el coeficiente $TreatB$ del modelo $l2$ (Tabla C.2) como $-2 \cdot (Treat1 + Treat1:Period1)$:

```r
-2 * (coef(l5)["Treat1"] + coef(l5)["Treat1:Period1"])
```

```
   Treat1
-0.1076137
```

- Análogamente el coeficiente $Period2$ del modelo $l2$ (Tabla C.2) se obtiene $-2 \cdot (Period1 + Treat1 : Period1)$:

```r
-2 * (coef(l5)["Period1"] + coef(l5)["Treat1:Period1"])
```

```
   Period1
-0.01125933
```

- El coeficiente $TreatB : Period2$ se obtiene como $4 \cdot Treat1 : Period1$:

```r
4 * (coef(l5)["Treat1:Period1"])
```

```
Treat1:Period1
     0.1343517
```

```r
options(contrasts = rep("contr.treatment", 2))
l6 <- lm(Response ~ Treat + Period, df)
data.frame(t(coef(l6))) %>% gt()
```

X.Intercept.	TreatB	Period2
0.01120158	-0.04259114	0.05480989

```r
options(contrasts = rep("contr.sum", 2))
l7 <- lm(Response ~ Treat + Period, df)
data.frame(t(coef(l7))) %>% gt()
```

X.Intercept.	Treat1	Period1
0.01731095	0.02129557	-0.02740495

Abstract English abstract, on the last page.

This is a bookdown template based on LaTeX memoir class.

Keywords Keyword in English, As a list.