

TRABAJO FIN DE GRADO

APLICACIÓN DE SIMULACIÓN DE MONTE CARLO A LA PROPAGACIÓN DE INCERTIDUMBRES EN ANÁLISIS PROBABILISTA DE ELECTROMIOGRAFÍA

TRABAJO FIN DE GRADO PARA
LA OBTENCIÓN DEL TÍTULO DE
GRADUADO EN INGENIERÍA EN
TECNOLOGÍAS INDUSTRIALES

SEPTIEMBRE 2025

**Claudia María Domínguez
Núñez**

DIRECTOR DEL TRABAJO FIN DE GRADO:
José Manuel Mira Mcwilliams

“Para mí todavía es una fuente interminable de sorpresa ver cómo unos pocos garabatos en una pared o sobre una hoja de papel pueden cambiar el curso de los asuntos humanos.”

Stanislaw Ulam

Agradecimientos

A mi familia, amigos y todas las personas que me han apoyado tanto en la elaboración de este trabajo como durante mi etapa universitaria. Gracias por la paciencia, la confianza y por ayudarme a levantarme cuando más me costaba hacerlo.

A mi tutor, José Manuel Mira McWilliams, por darme la oportunidad y ofrecerme este tema tan interesarme que me ha acercado a una de las muchas situaciones en las que se puede aplicar la estadística.

A Fadi Hallal-Peche, médico especialista en neurofisiología clínica en el Hospital Central de la Defensa Gómez Ulla, por su colaboración y participación en este trabajo.

Por último, a la Universidad Politécnica de Madrid, por hacerme crecer tanto en lo académico, como en lo profesional, como en lo personal.

Resumen

Introducción

La electromiografía (EMG) es una técnica que trabaja con músculos de contracción voluntaria (esqueléticos estriados). Su objetivo es analizar el estado del músculo y de las raíces nerviosas que lo controlan. Para hacerlo, se mide la actividad eléctrica del músculo en reposo y durante la contracción usando una aguja fina. Cada raíz nerviosa inerva a más de un músculo, y cada músculo es inervado por más de una raíz.

La EMG intraoperatoria sirve para ubicar los nervios y protegerlos durante la operación. Sin embargo, es común que la interpretación de los resultados de esta esté acompañada de incertidumbre.

Este TFG va a tratar de cuantificar esta incertidumbre para poder ayudar a la interpretación mencionada. Para conseguirlo, se hará uso principalmente de una simulación de Monte Carlo.

Objetivos

Se tomarán como referencia los resultados de Schirmer et al. (2011) para plantear un ejercicio que incluye tres raíces nerviosas y dos músculos. A grandes rasgos, el objetivo principal de este trabajo es estudiar la relación entre la estimulación de diferentes nervios y la respuesta muscular que se ve en la EMG. Se toman $P(N2|M1)$ y $P(N1|M2)$ como las variables de salida, con la simulación de Monte Carlo como hilo conductor.

Se trabajará con dos fuentes principales de incertidumbre. La primera es inherente al propio modelo por el hecho de usar el Teorema de Bayes y su ya de por sí carácter probabilístico. La segunda deriva de asignarle variabilidad a las probabilidades de entrada (probabilidades a priori y condicionadas) y propagarlas usando simulaciones de Monte Carlo.

Como se ha mencionado, se creará un modelo basado en el Teorema de Bayes para medir cómo la incertidumbre de las probabilidades a priori ($P(N_i)$) y los parámetros ($P(M_j|N_i)$) se propaga hacia las probabilidades de salida. También se identificarán los parámetros que más afectan a los resultados y se analizarán diferentes fuentes de variabilidad. Por último, se compararán los resultados obtenidos mediante Monte Carlo con los del método de perturbación de Taylor de primer orden.

Metodología

La simulación de Monte Carlo es un método que usa un conjunto grande de números aleatorios para crear varios escenarios. Con estas simulaciones, obtiene una representación de las distribuciones de las variables de salida. Normalmente se usa en Estadística cuando el comportamiento del modelo es complejo.

Se han realizado tres experimentos computacionales usando RStudio.

El primer experimento estudia la propagación de incertidumbre de las probabilidades a priori. Para ello, a estas (las probabilidades $P(N_i)$) se les da una incertidumbre en el intervalo $[0, 1]$. Los parámetros (probabilidades condicionadas $P(M_j|N_i)$) se mantienen fijos. El método de Monte Carlo será el que se use para ver cómo se comportan las salidas ($P(N_2|M_1)$ y $P(N_1|M_2)$).

En el segundo experimento se hace un análisis de sensibilidad para ver qué parámetros (probabilidades condicionadas $P(M_j|N_i)$) son los que más influyen en la incertidumbre. Las probabilidades a priori (probabilidades $P(N_i)$) se vuelven a generar de forma aleatoria en un intervalo $[0, 1]$. Esta vez, también se les da una variación del $\pm 10\%$ respecto a los valores originales a los parámetros. Se realiza un diseño factorial de 3 factores (probabilidades condicionadas $P(M_1|N_1)$, $P(M_1|N_2)$ y $P(M_1|N_3)$) y 3 niveles, un total de 27 escenarios, con un ANOVA. Dentro de cada uno de los 27 escenarios, se ejecuta una simulación de Monte Carlo con 10.000 iteraciones. Así, se obtiene la influencia e interacciones de las probabilidades condicionadas sobre las variables de salida $P(N_2|M_1)$ y $P(N_1|M_2)$.

Para terminar, en el último experimento se busca comparar Monte Carlo con el método de perturbación de Taylor de primer orden. En este caso, las probabilidades a priori ($P(N_i)$) se mantienen constantes y solo se varían en un $\pm 10\%$ respecto de sus valores originales las probabilidades condicionadas. Monte Carlo actúa como método de referencia para ver cómo de eficiente es en este caso, la aproximación de Taylor.

En los tres experimentos se incluyó además una diagnosis del modelo, aparte de sus correspondientes análisis de resultados. La diagnosis se realiza tanto al modelo ANOVA como a los modelos de Monte Carlo. En el primer caso se analizan las hipótesis de homocedasticidad y normalidad de residuos, mientras que en el segundo se estudian las hipótesis de normalidad, convergencia e independencia.

Resultados

En el primer experimento se observa una distribución de $P(N_2|M_1)$ con asimetría positiva, no uniforme y con sesgo hacia los valores bajos. Sin embargo, la distribución de $P(N_1|M_2)$ es más uniforme, sin sesgos. La correlación entre $P(N_2|M_1)$ y $P(N_1|M_2)$ es fuertemente negativa. La conclusión que se puede obtener en este caso es que la incertidumbre de las probabilidades a priori ($P(N_i)$) genera una gran varianza en las salidas (CV de 79.2% y de 59.1% para las dos probabilidades de salida). En cuanto a la diagnosis, se cumplen convergencia e independencia, pero no normalidad.

Según el análisis de sensibilidad del segundo experimento, para $P(N_2|M_1)$ el orden de influencia de mayor a menor es: $P(M_1|N_2)$, $P(M_1|N_1)$ y $P(M_1|N_3)$. Para $P(N_1|M_2)$ el orden de influencia (de mayor a menor) es: $P(M_1|N_1)$, $P(M_1|N_2)$ y $P(M_1|N_3)$. En el caso de $P(N_2|M_1)$ hay que tener en cuenta las interacciones entre $P(M_1|N_1) - P(M_1|N_2)$ y $P(M_1|N_1) - P(M_1|N_3)$, ninguna para $P(N_1|M_2)$. En este caso, la correlación entre $P(N_2|M_1)$ y $P(N_1|M_2)$ es positiva. La diagnosis valida que los residuos cumplen tanto normalidad, como homocedasticidad e independencia.

El Experimento 3 demuestra que los resultados de los métodos de Monte Carlo y Taylor son prácticamente iguales para una incertidumbre pequeña como la de $\pm 10\%$. Además, para esta misma situación, evidenció que esta incertidumbre pequeña produce una varianza diminuta (CV de 4.88% y de 3.1% para las probabilidades de salida).

Como conclusión general, lo que más incertidumbre genera es la falta de conocimiento de las $P(N_i)$, mucho más que las $P(M_j|N_i)$. También se puede ver que si la incertidumbre viene de las probabilidades $P(N_i)$ la correlación será negativa y que, si viene de las probabilidades condicionadas, será positiva.

Palabras clave

Simulación de Monte Carlo, Electromiografía (EMG), propagación de incertidumbre, Teorema de Bayes, análisis de sensibilidad, neurofisiología.

Códigos UNESCO

1209.03(Análisis de datos), 1209.05 (Análisis y diseño de experimentos), 1209.10 (Teorías y técnicas de muestreo), 2411.11 (Neurofisiología).

Tabla de contenidos

| | |
|--|-----------|
| Agradecimientos | 3 |
| Resumen | 4 |
| 1. Introducción | 9 |
| 1.1. El Método de Monte Carlo..... | 9 |
| 1.1.1. Origen del método de Monte Carlo | 9 |
| 1.2. Introducción a la electromiografía (EMG)..... | 10 |
| 1.2.1. Principios básicos de la EMG..... | 10 |
| 2. Objetivos | 11 |
| 3. Fundamentos teóricos | 13 |
| 3.1. Axiomas y propiedades de la probabilidad | 13 |
| 3.1.1. Ley de la Probabilidad Total | 14 |
| 3.1.2. Teorema de Bayes | 14 |
| 3.2. Definición y fundamentos del método de Monte Carlo | 15 |
| 3.2.1. Procedimiento de generación de números aleatorios | 16 |
| 3.2.2. Propagación de incertidumbres con el método de Monte Carlo..... | 17 |
| 3.2.3. Diagnóstico de la simulación de Monte Carlo..... | 18 |
| 3.3. Método de perturbación de Taylor | 19 |
| 3.3.1. Aproximación de Primer Orden | 21 |
| 3.4. Análisis de Sensibilidad | 23 |
| 3.4.1. Diseño de Experimentos..... | 23 |
| 3.4.2. Análisis de varianza (ANOVA)..... | 25 |
| 3.4.3. Diagnóstico del modelo ANOVA..... | 26 |
| 4. Experimentos computacionales en R | 27 |
| 4.1. Experimentos a realizar..... | 27 |
| 4.1.1. Experimento 1..... | 27 |
| 4.1.2. Experimento 2..... | 28 |
| 4.1.3. Experimento 3..... | 30 |
| 4.2. Implementación del código en R..... | 33 |
| 4.2.1. Experimento 1..... | 33 |
| 4.2.2. Experimento 2..... | 39 |

| | |
|---|-----------|
| 4.2.3. Experimento 3..... | 46 |
| 5. Resultados y discusión | 55 |
| 5.1. Experimento 1 | 55 |
| 5.1.1. Diagnósis de la simulaci3n de Monte Carlo..... | 58 |
| 5.2. Experimento 2 | 60 |
| 5.2.2. Diagnósis del modelo ANOVA..... | 65 |
| 5.3. Experimento 3 | 67 |
| 5.3.1. Diagnósis de la simulaci3n de referencia | 70 |
| 5.4. Discusi3n general de los resultados | 72 |
| 6. Conclusiones | 74 |
| 7. LÍneas futuras..... | 75 |
| 8. Bibliografía | 76 |
| 9. Planificaci3n temporal y presupuesto | 78 |
| 10. Responsabilidad social..... | 81 |
| 10.1. Evaluaci3n de impactos..... | 81 |
| 10.2. Análisis de aspectos legales y éticos | 82 |
| 10.3. Objetivos de Desarrollo Sostenible..... | 82 |
| 11. Índice de figuras | 84 |
| 12. Índice de tablas..... | 85 |
| 13. Unidades, abreviaturas y acr3nimos | 86 |
| 14. Glosario..... | 88 |
| 15. Anexos | 90 |
| 15.1. Anexo 1: C3digo en R del Experimento 1 | 90 |
| 15.2. Anexo 2: C3digo en R del Experimento 2 | 95 |
| 15.3. Anexo 3: C3digo en R del Experimento 3 | 100 |
| 15.4. Anexo 4: Desarrollo de las derivadas para el m3todo de Taylor | 106 |
| 15.5. Anexo 5: Tablas de los IC (Experimento 2)..... | 109 |

1. Introducción

1.1. El Método de Monte Carlo

1.1.1. Origen del método de Monte Carlo

El método de Monte Carlo es un método de muestreo estadístico que nació en la década de 1940, en plena Segunda Guerra Mundial. Fue desarrollado por Stanislaw Marcin Ulam, matemático polaco, y por John von Neumann, matemático y físico húngaro-estadounidense, cuando trabajaban en el Proyecto Manhattan en Los Álamos en Nuevo México. Este proyecto era dirigido por el físico Robert Oppenheimer, y llevó a la creación de las primeras armas nucleares, como la bomba atómica y de hidrógeno.

La idea surgió de una pregunta de Ulam en 1946 cuando jugaba al solitario Canfield, recuperándose de una enfermedad. Ulam se preguntó cuál sería la probabilidad de ganar la partida al repartir las cartas. Como él mismo explicó: “Después de pasar mucho tiempo tratando de estimarla mediante cálculos combinatorios puros, me pregunté si un método más práctico que el “pensamiento abstracto” no sería disponerlo, digamos, cien veces y simplemente observar y contar el número de partidas exitosas.” [1]. De este modo, se establecieron las bases del método Monte Carlo. Su nombre proviene del famoso casino de Mónaco, haciendo alusión a la naturaleza aleatoria de los métodos que utiliza. Para simular diferentes escenarios, se necesita una gran base de números aleatorios. Estos números permiten obtener resultados a través de simulaciones repetidas.

Ulam compartió su idea con von Neumann, quien vio su potencial para estudiar las reacciones en cadena de neutrones en dispositivos de fisión utilizando computadoras. Juntos, desarrollaron la parte matemática del método. Para generar los números aleatorios que necesitaban, von Neumann propuso una técnica de aceptación-rechazo, que consiste en seleccionar dos distribuciones uniformes independientes y aceptar o rechazar valores en función de una función de densidad deseada. Esto aseguraba que los números generados se ajustaran a esa distribución concreta que se necesitaba según el problema en cuestión.

En Los Álamos, el método de Monte Carlo se utilizó para calcular las propiedades del plutonio y otros materiales relacionados con las bombas atómicas, debido a la complejidad de las ecuaciones diferenciales y a la necesidad de hacer muchas integraciones en varias dimensiones. Además, más adelante se usó para calcular valores singulares de la ecuación de Schrödinger.

Hoy en día, el método de Monte Carlo podría aplicarse a un número de situaciones prácticamente infinito. Tiene cabida desde la microelectrónica o telecomunicaciones hasta las finanzas, meteorología o, incluso, en los casinos para descubrir sesgos de las ruletas.

1.2. Introducción a la electromiografía (EMG)

La electromiografía (EMG) es una técnica que mide la actividad eléctrica de los músculos de contracción voluntaria (esqueléticos estriados), tanto en reposo como durante la contracción. Su objetivo es analizar la coherencia entre la corteza motora y los músculos para ver el estado de las raíces nerviosas que los controlan. La coherencia es una medida de la dependencia lineal entre dos señales en el dominio de frecuencia, y la diferencia de fase puede estimar el retraso temporal entre la corteza y la EMG. La electromiografía es útil para investigar los impulsos fisiológicos hacia el músculo y las oscilaciones características de diferentes actividades funcionales en circuitos motores específicos [2].

Los miotomas son un grupo de fibras musculares inervadas por los axones motores de cada raíz. La mayoría de las raíces nerviosas inerva más de un músculo y la mayoría de los músculos son inervados por más de una raíz [3]. Es importante entender esta inervación para poder interpretar los resultados de la EMG y tomar una decisión acertada. Numerosos estudios han investigado a fondo los miotomas funcionales a través de la estimulación radicular durante cirugías espinales.

1.2.1. Principios básicos de la EMG

Los principios básicos de la EMG convencional son los siguientes [4]:

- *Registro de la actividad eléctrica en estado de reposo.* En un músculo en reposo, no se registra ninguna actividad eléctrica. Cuando se inserta la aguja electromiográfica, se percibe una señal eléctrica de unos milisegundos por la irritación de las fibras musculares. Si existiese alguna patología, podrían registrarse en reposo potenciales de denervación, fibrilaciones y ondas positivas de denervación.
- *Registro de la actividad eléctrica durante la contracción voluntaria.* Durante la contracción del músculo, se registra el potencial de la unidad motora (PUM). Este es el resultado de la actividad combinada de las fibras musculares de una unidad motora. El PUM se basa en parámetros como la duración, amplitud y forma, que ayudan a evaluar la salud y el funcionamiento del músculo.
- *Reclutamiento de unidades motoras.* Las unidades motoras se activan siguiendo un orden específico, empezando por las más pequeñas y terminando con las mayores, siguiendo el principio del tamaño de Henneman. Cuando se alcanza la máxima contracción, se puede observar un patrón de interferencia, que muestra la activación de múltiples unidades motoras.
- *Patrones de actividad eléctrica.* Los patrones de actividad eléctrica se analizan en términos de la frecuencia de descarga de las unidades motoras y el número de unidades activas. Estos patrones ayudan a determinar la estabilidad y la eficiencia del reclutamiento muscular.

2. Objetivos

En este Trabajo de Fin de Grado se va a estudiar la relación entre las raíces nerviosas y la respuesta muscular que provoca su estimulación, usando la probabilidad y propagando la incertidumbre para después analizarla. Este estudio va a estar basado en los resultados de la electromiografía (EMG) y en los hallazgos de Schirmer et al. (2011), quien registró el número de veces que un músculo respondía positivamente cuando se estimulaba un nervio concreto. Se estableció que la mejor respuesta para un músculo es la que se consigue con la menor estimulación (mA) de la raíz [5].

Para llevar a cabo este objetivo, el hilo conductor de todo el trabajo va a ser el Teorema de Bayes y el método de Monte Carlo, que se va a usar especialmente para ver cómo responden las variables de salida a la incertidumbre de las entradas.

El escenario sobre el que vamos a trabajar plantea tres posibles nervios que pueden estimularse (N1, N2 y N3) durante una intervención neurológica, pero en cada intervención solo se puede estimular un único nervio. Se contemplan dos músculos distintos (M1 y M2) y se observan sus respuestas, pudiendo contraerse más de un músculo con cada estimulación de la raíz. Tomando como referencia los datos que obtuvo Schirmer, se va a definir que las probabilidades a priori de estimulación de N1, N2 y N3 son 0.3, 0.2 y 0.5 respectivamente; y que las probabilidades condicionadas de que se contraiga M1 (antes que M2) habiendo estimulado N1, N2 y N3 son 0.4, 0.5 y 0.2 respectivamente. Las dos probabilidades a posteriori que se van a calcular en todos los casos son: la probabilidad de que, habiéndose contraído el M1, haya sido porque se estimuló el nervio N2 ($P(N2|M1)$); y la probabilidad de que, habiéndose contraído el M2, haya sido por estimular el N1 ($P(N1|M2)$).

El **primer experimento** estará centrado en las probabilidades a priori, siendo las únicas a las que se les va a introducir variabilidad, manteniendo por tanto fijas las probabilidades condicionadas. Estas últimas son las que se consideran los parámetros fisiológicos de nuestro modelo. La incertidumbre se estudiará mediante una simulación de Monte Carlo, analizando después las métricas estadísticas de las distribuciones que obtengamos.

En el **segundo experimento** se realizará un análisis de sensibilidad para identificar cuáles son los parámetros que más afectan a las dos variables de salida. Se usará un diseño factorial, donde las probabilidades condicionadas van a variar en los niveles bajo, medio y alto; considerando una variación del $\pm 10\%$ respecto a sus valores originales (ya mencionados). Los datos que obtengamos de los 27 escenarios resultantes se analizarán con un análisis de varianza (ANOVA), para determinar qué parámetros y posibles interacciones entre ellos son los que dominan la variabilidad de las probabilidades $P(N2|M1)$ y $P(N1|M2)$.

Finalmente, en un **tercer experimento** se comparará la efectividad del método de Monte Carlo frente a un método analítico: el método de perturbación de Taylor de primer orden. Se van a comparar las medias y varianzas de los resultados con los obtenidos mediante una simulación de Monte Carlo (considerada como referencia), en la que las probabilidades a priori de los

2. Objetivos

nervios se mantendrán fijas y la incertidumbre provendrá exclusivamente de las probabilidades condicionadas, también con una variación del $\pm 10\%$.

3. Fundamentos teóricos

3.1. Axiomas y propiedades de la probabilidad

Un espacio muestral es el conjunto de todos los resultados posibles de un experimento aleatorio. Para cualquier espacio muestral, una función de probabilidad debe cumplir unas condiciones mínimas, conocidas como los axiomas de Kolmogórov:

Para que una función sea considerada una medida de probabilidad, debe cumplir unas condiciones mínimas. A estas condiciones se las llama “axiomas de Kolmogórov”, y son las siguientes:

Axioma 1. La probabilidad de cualquier evento del espacio muestral es no negativa.

Axioma 2. La probabilidad de la totalidad del espacio muestral es 1.

Axioma 3. Para una secuencia de eventos mutuamente excluyentes, la probabilidad de que ocurra alguno de ellos es igual a la suma de sus probabilidades por separado.

A partir de estos axiomas, se presentan algunas propiedades básicas de la probabilidad de eventos [6]:

1. La probabilidad de que ocurra un evento E_i y/o E_j es menor o igual a la suma de las probabilidades por separado de E_i y E_j .
2. E_i y E_j son eventos mutuamente excluyentes si y solo si la ocurrencia de E_i implica que E_j no ocurre y viceversa. Si E_i y E_j son mutuamente excluyentes, entonces:
 - La probabilidad de que ocurran ambos eventos simultáneamente es nula.

$$P(E_i \cap E_j) = 0$$

- La probabilidad de que ocurra E_i o E_j es la suma de sus probabilidades individuales.

$$P(E_i \cup E_j) = P(E_i) + P(E_j)$$

3. Una clase completa de eventos puede ser mutuamente excluyente para i y j . Cuando la clase es exhaustiva, es decir, se han enumerado todos los eventos posibles, la suma de las probabilidades de todos los eventos es 1.
4. La probabilidad de que ocurra el resultado específico (E_i, F_j) es p_{ij} , llamada la “probabilidad conjunta para E_i y F_j ”.
5. Dos eventos son independientes si la ocurrencia de uno no afecta a la probabilidad del otro, es decir, $P(F_j | E_i) = P(F_j)$.
6. Si los eventos E_i y F_j son independientes, la probabilidad conjunta de que ambos ocurran es igual al producto de sus probabilidades individuales.
7. La probabilidad conjunta de eventos que dependen entre sí se descompone en una probabilidad marginal y una condicional. La primera es la probabilidad de que ocurra

E_i independientemente del segundo evento. La probabilidad condicional es la probabilidad de que ocurra el evento F_j , dado que ha ocurrido el evento E_i .

$$P(F_j | E_i) = \frac{P(E_i \cap F_j)}{P(E_i)}, \text{ siempre que } P(E_i) > 0$$

8. La suma de todas las probabilidades condicionales debe ser igual a uno para cualquier evento E_i .

Estas propiedades y axiomas son los que sustentan tanto el método de Monte Carlo como el Teorema de Bayes.

3.1.1. Ley de la Probabilidad Total

Suponiéndose que el espacio muestral se divide en k subpoblaciones $S_1, S_2, S_3, \dots, S_k$, que son mutuamente excluyentes y exhaustivas (es decir, todas juntas conforman todo el espacio muestral) y expresando un evento A como

$$A = (A \cap S_1) \cup (A \cap S_2) \cup (A \cap S_3) \cup \dots \cup (A \cap S_k)$$

Entonces

$$P(A) = P(A \cap S_1) + P(A \cap S_2) + P(A \cap S_3) + \dots + P(A \cap S_k)$$

Usando la regla de la multiplicación para escribir $P(A \cap S_i)$ como $P(S_i) \cdot P(A|S_i)$ para $i = 1, 2, \dots, k$, se obtiene la Ley de la Probabilidad Total [7]:

$$P(A) = P(S_1) \cdot P(A|S_1) + P(S_2) \cdot P(A|S_2) + \dots + P(S_k) \cdot P(A|S_k)$$

3.1.2. Teorema de Bayes

El Teorema de Bayes es una fórmula matemática que calcula probabilidades condicionadas. Una probabilidad condicionada se escribe como $P(A|B)$, y representa la probabilidad de que ocurra el evento A sabiendo que B ha ocurrido [8].

El Teorema de Bayes relaciona la probabilidad a priori con la probabilidad a posteriori mediante la siguiente fórmula:

$$P(A_i | B) = \frac{P(B | A_i) \cdot P(A_i)}{P(B)}$$

Donde:

- $P(A_i|B)$ es la probabilidad a posteriori de A_i sabiendo que ha ocurrido B .
- $P(B|A_i)$ es la probabilidad de que se dé B dado que A_i ha ocurrido.
- $P(A_i)$ es la probabilidad a priori del evento A_i .
- $P(B)$ es la probabilidad total de B . Se calcula usando la Ley de la Probabilidad Total, en la que se suman las probabilidades conjuntas de B con todos los eventos posibles de A :

$$P(B) = P(B | A_1) \cdot P(A_1) + P(B | A_2) \cdot P(A_2) + \dots + P(B | A_i) \cdot P(A_i)$$

En este TFG, el Teorema de Bayes se va a usar para calcular las probabilidades $P(N2|M1)$ y $P(N1|M2)$.

3.2. Definición y fundamentos del método de Monte Carlo

En términos generales, la simulación de Monte Carlo usa un gran número de valores aleatorios, y, después, calcula el valor medio de los resultados que ha obtenido. Este procedimiento es justo lo contrario de lo que hacen otros métodos, que sustituyen la variable con incertidumbre por un valor promedio y trabajan sobre él. Monte Carlo, al trabajar incorporando la aleatoriedad, permite un rango más amplio de posibles resultados y, por tanto, da una visión mucho más completa de cómo se comporta la incertidumbre del modelo.

El fundamento teórico que le da validez al método es la Ley de los Grandes Números. Este teorema establece que la media de los resultados observados (\bar{Y}) tras un gran número de simulaciones (T), tiende a estabilizarse sobre el valor esperado real ($E[Y]$) del experimento aleatorio; es decir:

$$E[Y] \approx \bar{Y} = \frac{1}{T} \sum_{i=1}^T Y_i$$

A medida que el número de simulaciones (T) tiende a infinito, la aproximación se convierte en una igualdad, asegurando que el método converge al resultado correcto.

Cabe destacar la diferencia entre eventos aleatorios elementales y compuestos para entender cómo se construyen y analizan las simulaciones. Los primeros son aquellos posibles resultados individuales de un experimento aleatorio que no pueden descomponerse en otros más simples (son los bloques básicos de la simulación). Por otro lado, los eventos aleatorios compuestos son los que se definen a partir de eventos elementales y que permiten modelar sistemas con muchas variables que interactúan entre sí.

Monte Carlo funciona generando números pseudoaleatorios que simulan valores de las variables de entrada. Con un gran número de simulaciones, se consigue una representación de la distribución de las variables de salida. Los pasos fundamentales del método, como bien explican los autores de referencia Cox y Siebert, son los siguientes [9]:

1. Se define el modelo de medición. En el primer paso se describe cómo se combinan las variables de entrada para crear las de salida mediante una relación matemática.
2. Se especifican las distribuciones de las variables de entrada. Cada variable de entrada se describe con una distribución de probabilidad que refleja su incertidumbre.
3. En el tercer paso, se crean números pseudoaleatorios que simulan, según sus distribuciones, los valores de las variables de entrada. Con esto se obtienen los valores de entrada representativos.

4. Propagación a través del modelo. Los valores que se han simulado de estas variables de entrada se introducen en el modelo que describe el sistema de estudio.
5. En el quinto paso, se calcula la representación discreta de las salidas. Se hace una recopilación de los resultados de las simulaciones para construir la distribución de las variables de salida.
6. El último paso es el análisis estadístico. Se realizan los cálculos relevantes, como la media, la desviación estándar, los intervalos de confianza de las variables de salida o la matriz de covarianza.

3.2.1. Procedimiento de generación de números aleatorios

Normalmente, los números pseudoaleatorios se prefieren frente a los números puramente aleatorios. La razón es que estos últimos son difíciles de generar a una velocidad relativamente baja y necesitan (para cálculos grandes) una tabla donde estén todos recogidos, lo que no es práctico. Los números pseudoaleatorios, aunque no son puramente aleatorios, son secuencias de números entre $[0, 1]$ fáciles de generar y que satisfacen algunas pruebas de aleatoriedad, lo que suele hacerlos más adecuados.

El generador de números pseudoaleatorios (*Pseudo-Random Number Generator*, PRNG) debe cumplir las pruebas estadísticas más relacionadas con el problema de estudio [6].

A continuación, se describen algunos de los principales generadores de PRN [10]:

- *Generadores Congruenciales Lineales (LCG)*: generan una secuencia de números de la siguiente forma:

$$x_{i+1} = a_0x_i + a_1x_{i-1} + \dots + a_jx_{i-j} + b \pmod{P}$$

La “semilla” es el valor inicial (x_0) y los parámetros enteros positivos “multiplicador”, “incremento” y “módulo” se corresponden con a , b y P en la fórmula, respectivamente. Este último determina el rango de los números generados. La secuencia se repetirá al cabo de un periodo, que es la longitud máxima de la secuencia antes de comenzar a repetirse, y será como máximo P . Para alcanzar este máximo, es necesario que los parámetros cumplan con ciertos criterios. La semilla es importante porque, al fijar el mismo valor inicial, la secuencia de números generada será siempre idéntica, garantizando así la reproducibilidad de la simulación.

- *Generadores Congruenciales Multiplicativos (MCG)*: constituyen un caso particular del LCG, que requiere $a_j = 0$, j sea mayor o igual que uno y que $b = 0$.

$$x_{i+1} = a \cdot x_i \pmod{P}$$

La elección de un número primo muy grande como valor de P puede asegurar una secuencia con un periodo suficientemente largo.

- *Generadores Congruenciales de Fibonacci Retardados*: Estos generadores son una generalización de una secuencia simple de Fibonacci $x_{i+2} = x_{i+1} + x_i$ cuya fórmula de recurrencia es

$$x_i = x_{i-p} + x_{i-q} \pmod{P}$$

donde $p > q$ son enteros positivos que determinan el retardo.

- *Desplazamiento de Registros con Retroalimentación Lineal*: Otro método consiste en generar dígitos binarios, donde $b_i \in \{0,1\}$.

$$b_i = a_1 b_{i-1} + \dots + a_r b_{i-r} \pmod{2}$$

En la mayoría de los generadores, los a_j son una secuencia de ceros y unos correspondiente a

$$b_i = b_{i-p} \oplus b_{i-(p-q)}$$

donde \oplus denota la suma con desplazamiento entre bits.

En el caso que nos ocupa, la generación de números pseudoaleatorios permitirá simular las probabilidades de contracción muscular y la implicación de las raíces nerviosas. Para conseguirlo se utilizará la función “`runif()`” de RStudio como generador de números pseudoaleatorios que siguen una distribución uniforme en el intervalo $[0, 1]$. Esta función está basada en el Generador Congruencial Lineal (LCG), ya explicado anteriormente.

3.2.2. Propagación de incertidumbres con el método de Monte Carlo

Es importante distinguir entre los términos “incertidumbre” y “error”. El error es la diferencia entre el valor medido y el valor real de lo que se está midiendo, mientras que la incertidumbre es una forma de medir cuánto se duda del resultado de la medida [11].

Cuando se trata de evaluación de incertidumbres, Monte Carlo identifica la distribución de probabilidad de las variables de salida a partir de las que tienen las variables de entrada. La “propagación de incertidumbres” se refiere a estudiar cómo la variabilidad de las variables de entrada se transmite a las de salida.

El documento de referencia internacional para la evaluación de incertidumbres es la “Guía para la Expresión de la Incertidumbre en la Medición” o GUM. El método de Monte Carlo puede ser útil cuando la información que proporciona el GUM no es suficiente.

La forma en la que el método de Monte Carlo introduce incertidumbre, como ya se ha mencionado, es generando números de forma aleatoria. A partir de estos valores, se calcula la incertidumbre estándar y el intervalo de confianza del 95%. Monte Carlo no necesita que las funciones sean derivables y funciona para diferentes tipos de distribuciones, lo que permite obtener resultados más realistas [12]. Esto marca una ventaja respecto al GUM, que requiere que las funciones sean derivables y las incertidumbres se clasifiquen en tipo A (evaluadas por métodos estadísticos) y B (evaluadas por otros métodos).

Aun así, el método de Monte Carlo presenta también algunas limitaciones. Por ejemplo, es un método que depende de que los generadores de números pseudoaleatorios utilizados sean de calidad. También depende del número de simulaciones que se hagan.

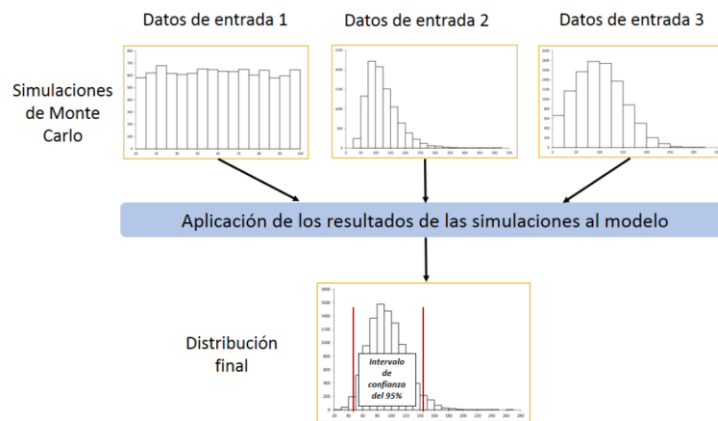


Figura 1. Esquema del método de Monte Carlo para la propagación de incertidumbre

3.2.3. Diagnóstico de la simulación de Monte Carlo

Para asegurar que se han obtenido resultados válidos y que la simulación se ha comportado como se esperaba, es necesario validar algunas hipótesis. Este proceso se hace a través de la diagnosis del modelo.

La hipótesis de normalidad afirma que los valores de la variable respuesta, y_{ij} , siguen una distribución normal con media del modelo μ y varianza de la componente aleatoria σ^2 . Para verificar la hipótesis de normalidad se pueden emplear gráficos como histogramas o Q-Q plots, también llamados *gráficos de probabilidad normal*, que comparan la distribución de los datos con una distribución normal teórica. Si los puntos se aproximan a una línea recta, podrá decirse que son normales, sin perder de vista que siempre tenderá a observarse una mayor desviación en los extremos.

Sin embargo, la limitación que presentan los métodos gráficos es que siempre implican mucha subjetividad; entre otras, porque es el investigador quien ha de juzgar en qué medida los puntos se ajustan o no a la recta. Por eso, para una mayor objetividad, se suelen emplear como soporte los contrastes de normalidad, entre los que destaca el de Shapiro-Wilk. En este último, se mide cuánto se ajustan las observaciones a la recta normal en el gráfico Q-Q. La hipótesis nula de normalidad se rechaza si el ajuste es malo; es decir, si los valores del estadístico de contraste son pequeños. El contraste de Shapiro-Wilk es el que se recomienda cuando el tamaño muestral es pequeño (menor que 50) y también requiere que los parámetros de la distribución estén especificados [13]. En los experimentos de este trabajo, las muestras son de 10.000 puntos, por lo que se ha optado por omitir este contraste. La razón es que, por su alta sensibilidad, puede detectar desviaciones de normalidad que son teóricamente irrelevantes.

La hipótesis de convergencia establece que los resultados se van a estabilizar alrededor del valor medio al aumentar el número de simulaciones. Si esta hipótesis se cumple, se puede decir que los resultados que se han obtenido representan bien el comportamiento del modelo.

Para comprobar la convergencia, se usa el gráfico de la media acumulada de las simulaciones. Este gráfico representa la media acumulada, que se calcula como el promedio de todos los

resultados, en función del número de observaciones. Si se observa que la curva se aplana al final, significa que no va a cambiar de forma significativa el valor promedio aunque se añadan más simulaciones.

La hipótesis de independencia comprueba que los eventos asociados a una variable no afectan a la probabilidad de que ocurran los eventos asociados a la otra variable. Si existiese autocorrelación, querría decir que no se están generando los números aleatorios correctamente, lo que le restaría validez a Monte Carlo. Aunque la hipótesis de independencia no se pueda probar de forma exacta, sí que puede comprobarse la ausencia de autocorrelación. En este TFG se usan para eso la función de autocorrelación (ACF) y la prueba de Durbin-Watson.

La ACF permite identificar patrones de dependencia entre observaciones separadas por diferentes retardos (*lags*). En un gráfico ACF, habrá un único pico significativo en el lag 0 y las demás autocorrelaciones irán cambiando aleatoriamente alrededor de cero para lags superiores, cuando la simulación sea independiente.

El test de Durbin-Watson contrasta la hipótesis nula de ausencia de autocorrelación de primer orden (es decir, en el *lag* 1) en los residuos de un modelo o en una secuencia de datos. El test genera un estadístico, d , cuya proximidad a 2 indicaría independencia. Un p-valor alto (superior a 0.05) confirmaría que no hay suficiente evidencia como para rechazar la hipótesis nula de independencia. Valores del estadístico d muy alejados de 2 y un p-valor bajo (inferior a 0.05) sugerirían que hay autocorrelación.

$$d = \frac{\sum_{t=2}^n (e_t - e_{t-1})^2}{\sum_{t=1}^n e_t^2}$$

donde e_t son los residuos del modelo.

Como ya se ha mencionado, si el valor de d es similar a 2 se puede decir que no hay autocorrelación; si es menor que 2, que existe autocorrelación positiva; y, si es mayor, que la autocorrelación es negativa.

3.3. Método de perturbación de Taylor

No solo con Monte Carlo se puede estudiar la incertidumbre asociada a una función de variables aleatorias. Otro de los métodos más conocidos es el de perturbación o de expansión en series de Taylor.

Con él se estudia la ecuación matemática del modelo. La media, varianza, asimetría, etc. se calculan expandiendo la ecuación del modelo en una serie de Taylor en torno a las medias. La expansión da como resultado una ecuación para los momentos estadísticos globales, en función de los momentos de las variables y de las derivadas parciales de la ecuación del propio modelo [14].

El método se basa en aproximar la función del modelo, $Y = f(X_1, X_2, \dots, X_n)$, con una expansión en series de Taylor alrededor del punto que definen los valores medios de las variables de entrada $(\mu_{X_1}, \mu_{X_2}, \dots, \mu_{X_n})$. El modelo puede ser lineal o no, con X_i son aleatorias.

Cómo de preciso sea el método depende del orden en el que se trunca la serie de Taylor. Las posibles aproximaciones son:

- *Aproximación de Primer Orden.* La serie se trunca después de los términos lineales, lo que da lugar a una aproximación del modelo lineal en el entorno del punto medio. Esta aproximación es la que más se usa en la práctica y es la base de la metodología descrita en el GUM.

La media se estima como:

$$\mu_Y \approx f(\mu_{X_1}, \dots, \mu_{X_n})$$

La varianza se estima como:

$$\sigma_Y^2 \approx \sum_{i=1}^n \sum_{j=1}^n \frac{\partial f}{\partial X_i} \left| \frac{\partial f}{\partial X_j} \right|_{\mu} \text{Cov}(X_i, X_j)$$

En los experimentos se va a asumir que las entradas son independientes. Se tomará una covarianza nula y se usará la fórmula simplificada:

$$\sigma_Y^2 \approx \sum_{i=1}^n \left(\frac{\partial f}{\partial X_i} \right)^2 \sigma_{X_i}^2$$

Esta es la forma más común en la que se usa la Ley de Propagación de Incertidumbre.

- *Aproximación de Segundo Orden.* Se incluyen también los términos cuadráticos de la serie (aquellos con segundas derivadas). Proporciona una aproximación cuadrática del modelo, que puede ser más precisa si la función presenta una no linealidad significativa o si las incertidumbres de entrada son grandes.

La media incorpora una corrección adicional:

$$\mu_Y \approx f(\mu_X) + \frac{1}{2} \sum_{i=1}^n \frac{\partial^2 f}{\partial X_i^2} \Big|_{\mu} \sigma_{X_i}^2$$

- *Órdenes superiores.* No suelen utilizarse porque suele aumentar la complejidad rápidamente. En comparación con el coste de analizarlas, la ganancia en cuanto a precisión no compensa.

En esencia, el método de Taylor propaga la incertidumbre mediante derivadas parciales, multiplicadas por la varianza y covarianza de las entradas. La pendiente local $\left(\frac{\partial f}{\partial X_i} \right)$ indica cuánta variación en Y se produce por una variación unitaria en X_i . Al multiplicar esa pendiente

por la varianza de X_i y combinar con los términos cruzados (covarianzas), se obtiene lo que contribuye cada entrada a la incertidumbre de la salida.

En este trabajo, se empleará la aproximación de primer orden del método de perturbación de Taylor en el Experimento 3 porque la variación de las entradas es pequeña y los efectos de segundo orden se consideran despreciables.

3.3.1. Aproximación de Primer Orden

La aproximación de Taylor de primer orden estima cómo se propaga la incertidumbre de las entradas a través de una función Y . Como se asume que las variaciones en estas variables son pequeñas, pueden usarse solo las derivadas parciales de primer orden (sensibilidades) de la función respecto a cada entrada.

Se puede observar cómo una función $f(x_1, x_2, \dots, x_n)$ cambia según pequeñas variaciones en sus variables de entrada δx_i usando:

$$f(x_1 + \delta x_1, \dots, x_n + \delta x_n) \approx f(x_1, \dots, x_n) \cdot \sum_{i=1}^n \frac{\partial f}{\partial x_i} \delta x_i$$

En el contexto de la propagación de incertidumbre, $f(x_1, x_2, \dots, x_n)$ representa el valor de la función que se calcula con los valores medios de las entradas. La suma de las derivadas parciales multiplicadas por las perturbaciones (δx_i) representa la variación que se estima en la salida.

Con esta aproximación lineal, se deriva la fórmula para la propagación de la incertidumbre, que estima la varianza de la salida (σ_Y^2). Si se asume que las variables de entrada son independientes, se calcula como:

$$\sigma_Y^2 \approx \sum_{i=1}^n \left(\frac{\partial f}{\partial X_i} \right)^2 \sigma_{X_i}^2$$

Esta ecuación se va a usar en el Experimento 3. Las funciones que se buscan analizar son $P(N2|M1)$ y $P(N1|M2)$, que dependen de las probabilidades condicionadas $P(M1|N1)$, $P(M1|N2)$ y $P(M1|N3)$.

Para calcular $P(N2|M1)$ hay que determinar previamente $P(M1)$, que se obtiene aplicando la Ley de la Probabilidad Total:

$$P(M1) = P(N1) \cdot P(M1 | N1) + P(N2) \cdot P(M1 | N2) + P(N3) \cdot P(M1 | N3)$$

La expansión de Taylor de primer orden para $P(M1)$ es:

$$pertP(M1) = P(M1) + \sum_{i=1}^3 \frac{\partial P(M1)}{\partial P(M1 | Ni)} \cdot \delta P(M1 | Ni)$$

3. Fundamentos teóricos

Donde las derivadas parciales indican la sensibilidad de $P(M1)$ respecto a cada probabilidad condicionada:

$$\frac{\partial P(M1)}{\partial P(M1 | N1)} = P(N1)$$
$$\frac{\partial P(M1)}{\partial P(M1 | N2)} = P(N2); \quad \frac{\partial P(M1)}{\partial P(M1 | N3)} = P(N3)$$

La función $P(N2|M1)$ se obtiene mediante la aplicación del Teorema de Bayes:

$$P(N2 | M1) = \frac{P(M1 | N2) \cdot P(N2)}{P(M1)}$$

Se usa de nuevo la expansión de Taylor de primer orden para ver cómo cambia su valor cuando lo hacen $P(M1|N1)$, $P(M1|N2)$ y $P(M1|N3)$:

$$pertP(N2 | M1) = P(N2 | M1) + \sum_{i=1}^3 \frac{\partial P(N2 | M1)}{\partial P(M1 | Ni)} \cdot \delta P(M1 | Ni)$$

Las derivadas parciales o sensibilidades enseñan cómo cambia $P(N2|M1)$ con cada probabilidad. Cada una se obtiene aplicándole a la función la regla del cociente.

$$\frac{\partial P(N2 | M1)}{\partial P(M1 | N1)} = - \frac{P(N2) \cdot P(M1 | N2) \cdot P(N1)}{P(M1)^2}$$
$$\frac{\partial P(N2 | M1)}{\partial P(M1 | N2)} = \frac{(P(N2) \cdot P(M1)) - (P(M1 | N2) \cdot P(N2) \cdot P(N2))}{P(M1)^2}$$
$$\frac{\partial P(N2 | M1)}{\partial P(M1 | N3)} = - \frac{P(N2) \cdot P(M1 | N2) \cdot P(N3)}{P(M1)^2}$$

Se realiza el mismo proceso con la probabilidad $P(N1|M2)$. Primero se define la función $P(M2)$:

$$P(M2) = P(N1) \cdot (1 - P(M1 | N1)) + P(N2) \cdot (1 - P(M1 | N2)) + P(N3) \cdot (1 - P(M1 | N3))$$

Y la función respuesta final $P(N1|M2)$:

$$P(N1 | M2) = \frac{(1 - P(M1 | N1)) \cdot P(N1)}{P(M2)}$$

Las correspondientes expansiones de Taylor de primer orden para estas funciones son:

$$pertP(M2) = P(M2) + \sum_{i=1}^3 \frac{\partial P(M2)}{\partial P(M1 | Ni)} \cdot \delta P(M1 | Ni)$$

$$pertP(N1 | M2) = P(N1 | M2) + \sum_{i=1}^3 \frac{\partial P(N1 | M2)}{\partial P(M1 | Ni)} \cdot \delta P(M1 | Ni)$$

Donde:

$$\frac{\partial P(M2)}{\partial P(M1|N1)} = -P(N1) \quad ; \quad \frac{\partial P(M2)}{\partial P(M1|N2)} = -P(N2)$$

$$\frac{\partial P(M2)}{\partial P(M1 | N3)} = -P(N3)$$

$$\frac{\partial P(N1 | M2)}{\partial P(M1 | N1)} = - \frac{P(N1) \cdot P(M2) + (1 - P(M1|N1)) \cdot P(N1) \cdot (-P(N1))}{P(M2)^2}$$

$$\frac{\partial P(N1 | M2)}{\partial P(M1 | N2)} = \frac{(1 - P(M1 | N1)) \cdot P(N1) \cdot P(N2)}{P(M2)^2}$$

$$\frac{\partial P(N1 | M2)}{\partial P(M1 | N3)} = \frac{(1 - P(M1 | N1)) \cdot P(N1) \cdot P(N3)}{P(M2)^2}$$

En los capítulos posteriores, se implementarán estas fórmulas para obtener estimaciones numéricas que serán directamente comparadas con los resultados de la simulación de Monte Carlo del Experimento 3.

3.4. Análisis de Sensibilidad

3.4.1. Diseño de Experimentos

El diseño de experimentos sirve para planificar qué pruebas se deben realizar y cómo, para obtener datos que, cuando se analicen, den evidencia objetiva que aclaren los aspectos inciertos de un proceso, resolver un problema o conseguir mejoras. [15]

Su principal objetivo es estudiar la relación que hay entre el resultado de un proceso y los factores que le afectan. Para conseguirlo, se manipulan las variables controladas y se ven los efectos en las respuestas.

Las variables controladas son las que se pueden fijar en un nivel específico. Se pueden ajustar durante el experimento y son conocidas también como factores controlables, variables de entrada, variables de diseño o parámetros del proceso. Por otro lado, las variables respuesta son aquellas en las que se observan los efectos o resultados del experimento.

Algunos de los principios básicos del diseño experimental son:

3. Fundamentos teóricos

- *Aleatorización*. Si los ensayos se hacen en orden aleatorio, aumenta la probabilidad de cumplir la hipótesis de independencia. También asegura que las diferencias en los resultados que se puedan observar sean por los factores que se han estudiado y no por otros que no se han controlado.
- *Repetición*. Se ejecuta con las mismas condiciones más de una combinación de factores.
- *Bloqueo*. Ya se ha mencionado que los factores controlados son los que se pueden manipular. Los no controlados, consecuentemente, son los que no pueden modificarse. Hay que tener en cuenta todos los factores que puedan afectar a la salida, controlados y no controlados, para disminuir la variabilidad de los segundos.

Los tipos de diseños de experimentos son:

- *Diseños factoriales*. Se investigan al mismo tiempo dos o más factores relevantes, cada uno con varios niveles. Los niveles son los valores que puede tomar un factor. Se utiliza para poder estudiar las interacciones que pueda haber entre los factores y cómo afectan a la variable de salida.
En las simulaciones, como las que se realizan en Monte Carlo, los diseños factoriales permiten medir cómo diferentes variables influyen en la incertidumbre de los resultados. Se tiene un factor con K niveles y otro con L niveles, y se plantea la posibilidad de que alguno de los dos afecte a las variables respuesta. Para evaluar el efecto, se fija un nivel de uno de los factores y se aplican los niveles del otro, viendo cómo influyen.
- *Diseños de superficie de respuesta (RSM)*. Técnicas estadísticas que modelan y analizan problemas donde una respuesta de interés está influenciada por otras varias. Busca optimizar esta respuesta, por lo que normalmente hay que encontrar los niveles óptimos de las variables controladas que la minimizan o maximizan.

El diseño factorial se va a usar en el segundo experimento para el análisis de sensibilidad. Se parte de la base de que todos los factores tienen que ser independientes entre sí. En este trabajo las probabilidades iniciales $P(N1)$, $P(N2)$ y $P(N3)$ son dependientes porque se les ha exigido que $P(N1) + P(N2) + P(N3) = 1$.

El Experimento 2 incluye tres factores con tres niveles. Se crean 27 escenarios.

- Factor 1: $P(M1|N1)$
 - Nivel 1: 0.36
 - Nivel 2: 0.4
 - Nivel 3: 0.44
- Factor 2: $P(M1|N2)$
 - Nivel 1: 0.45
 - Nivel 2: 0.5
 - Nivel 3: 0.55

- Factor 3: P(M1|3)
 - Nivel 1: 0.18
 - Nivel 2: 0.2
 - Nivel 3: 0.22

El diseño factorial 3^3 del Experimento 2 es un diseño no replicado. La salida de cada uno de los 27 escenarios no se obtiene con una única observación: es el valor que estima una simulación de Monte Carlo interna de 10.000 iteraciones. Con una simulación de Monte Carlo dentro de cada escenario, la varianza de cada punto se hace muy pequeña.

Por esto, se admite que a la variabilidad de los 27 valores medios la dominan los efectos de los factores y sus interacciones, y no el ruido aleatorio de la simulación. Como no se van a hacer repeticiones para estimar el error puro del experimento, el término de error para el modelo ANOVA se obtiene asumiendo que las interacciones de orden más alto (en este caso, la interacción de tercer orden) son despreciables.

3.4.2. Análisis de varianza (ANOVA)

El análisis de varianza (ANOVA) se centra en las medias de grupos diferentes y las compara para ver si existe alguna diferencia significativa entre ellas.

El ANOVA contrasta la hipótesis nula H_0 : “todas las medias son iguales” y la hipótesis H_1 : “al menos una es distinta”. Esto lo hace usando el estadístico F , que se obtiene haciendo: $F = \frac{MSA}{MSE}$. “MSA” es el cuadrado medio del factor de interés (variación entre medias de grupos) y “MSE” es el cuadrado medio del error (variación dentro de cada grupo). Esto es, compara la variabilidad explicada (por el modelo) con la no explicada (los errores o ruido). Si el valor de F es alto (p-valor asociado menor que 0.05), se puede rechazar la hipótesis nula (es decir, que al menos hay una media que es diferente al resto).

Esto viene a coalición de que la variabilidad total (SST) se puede dividir en la variabilidad que explica el modelo (SSA) y la residual (SSE): $SST = SSA + SSE$.

El factor es la variable de entrada que puedes controlar y cuyo efecto se busca en la salida. Para identificar el efecto que cada factor produce, es necesario que las variables de entrada tomadas sean independientes. Los niveles serían los distintos valores que puede tomar el factor.

Si se toman dos factores o más, se habla de un ANOVA factorial. En este caso, se podrían estudiar las interacciones entre los factores, además de los efectos principales. Los efectos principales serían los directos de cada factor por sí solo. Las interacciones explican cómo la combinación de los factores afecta a la salida. Si las interacciones son entre dos factores, se les llama “interacciones de segundo orden” y, si son entre tres o más, “interacciones de orden superior”.

Por último, un modelo ANOVA puede incluir (o no) repeticiones, dependiendo del modelo que se haya diseñado. Una replicación es el número de observaciones independientes que se

hacen para cada combinación de factor y nivel. Las replicaciones están muy relacionadas con el término de error, pues se hacen para poder reducir el error experimental.

3.4.3. Diagnóstico del modelo ANOVA

El análisis de varianza compara la variabilidad que explican los factores con la no explicada (error). Esta segunda se refleja en los residuos, por eso las hipótesis de la diagnóstico se prueban sobre ellos. Realmente se usan los residuos porque no se conocen los errores reales, por lo que se estiman sobre ellos.

Los residuos del ANOVA están sujetos a tres hipótesis que deben cumplirse:

- *Normalidad.* El estadístico F parte de la base de que los residuos siguen una distribución normal, por lo que hay que comprobarlo.
Esta hipótesis se comprueba con los gráficos Q-Q, que comparan los cuantiles de los residuos con los de una distribución normal. Si cumplen la hipótesis, los puntos se ajustarán a la recta teórica.
- *Homocedasticidad.* MSA y MSE se basan en que las varianzas de todos los grupos son iguales. Si no lo fueran, el valor de F podría dar un valor alterado o erróneo.
Para comprobar la homocedasticidad de los residuos, se usan gráficos de residuos frente a valores ajustados. Los residuos aparecen en el eje Y, y los valores ajustados, en el X. Si los puntos se distribuyen de manera aleatoria por el gráfico, significa que se cumple la hipótesis.
- *Independencia.* En ANOVA se asume que cada observación aporta información nueva. Si existen dependencias en los residuos, el error se subestima y el valor de F se engrosa, pudiendo dar por válidos resultados que en realidad no son relevantes.
En el Experimento 2 (que es en el que se hace el ANOVA) se han creado 27 escenarios. En cada uno de esos 27 escenarios, además, se ha llevado a cabo una simulación de Monte Carlo con 10.000 iteraciones (aleatorias y, por tanto, independientes). Por el propio diseño del modelo, se puede asumir que esta hipótesis se cumple.

4. Experimentos computacionales en R

4.1. Experimentos a realizar

4.1.1. Experimento 1

En este primer experimento se estudia cómo la incertidumbre de las probabilidades a priori se transmite a las a posteriori. Como ya se ha mencionado, las a priori son las probabilidades de estimular los nervios N1, N2 y N3. Las probabilidades a posteriori son $P(N2|M1)$ y $P(N1|M2)$; es decir, las probabilidades de que, sabiendo que se ha contraído un músculo, sea por la estimulación de dichos nervios.

Usando el método de Monte Carlo se va a modelar la incertidumbre de las probabilidades a priori. Las probabilidades condicionadas $P(M1|N1)$, $P(M1|N2)$ y $P(M1|N3)$ se van a mantener fijas y no se les va a introducir incertidumbre en este caso.

Se van a seguir los siguientes pasos:

1. *Generación de probabilidades iniciales.*

Se simulan 10.000 valores aleatorios para “ $probN1$ ” y “ $probN2$ ” usando una distribución uniforme en el intervalo $[0, 1]$. Para garantizar que las probabilidades de los tres nervios sumen 1, se ajusta “ $probN2$ ” tal que:

$$probN2 = \min(probN2, 1 - probN1)$$

Posteriormente, se calcula “ $probN3$ ”, asegurando que no sea negativa:

$$probN3 = 1 - probN1 - probN2$$

2. *Probabilidades condicionadas de contracción de cada músculo.*

Se asignan los valores de las probabilidades condicionadas usando los resultados de Schirmer et al. (2011) como referencia.

$$P(M1|N1) = 0.4 \quad , \quad P(M2|N1) = 1 - P(M1|N1)$$

$$P(M1|N2) = 0.5 \quad , \quad P(M2|N2) = 1 - P(M1|N2)$$

$$P(M1|N3) = 0.2 \quad , \quad P(M2|N3) = 1 - P(M1|N3)$$

La probabilidad total de que M1 se contraiga primero se obtiene mediante la Ley de la Probabilidad Total:

$$P(M1) = (probN1 \cdot 0.4) + (probN2 \cdot 0.5) + (probN3 \cdot 0.2)$$

De forma análoga, se calcula $P(M2)$.

3. *Cálculo de las probabilidades condicionadas de interés.*

Posteriormente, haciendo uso del Teorema de Bayes, se obtienen las probabilidades de interés:

$$P(N2 | M1) = \frac{probN2 \cdot 0.5}{P(M1)} \quad , \quad P(N1 | M2) = \frac{probN1 \cdot P(M2|N1)}{P(M2)}$$

donde $P(M2|N1)$ es conocida y constante.

4. *Análisis estadístico de los resultados.*

Para cada probabilidad de salida se calculan la media, la varianza y los cuartiles (25%, 50% y 75%). También se crean histogramas que muestran la distribución de $P(N2|M1)$ y $P(N1|M2)$ y se calcula la correlación entre ambas.

5. *Diagnosis de la simulación de Monte Carlo.*

Por último, se usan gráficos Q-Q para estudiar si las distribuciones de $P(N2|M1)$ y $P(N1|M2)$ se aproximan a la de la normal. Se realiza un análisis de la media acumulada para comprobar que los datos que se han obtenido se estabilizan conforme aumenta el número de simulaciones. Para terminar esta diagnosis se confirma la ausencia de autocorrelación con la función ACF y el test de Durbin-Watson.

4.1.2. Experimento 2

Al igual que en el primero, se van a generar 10.000 valores aleatorios para $N1$, $N2$ y $N3$ usando una distribución uniforme en el intervalo $[0, 1]$. Para este Experimento 2, se va a contemplar una incertidumbre adicional en las probabilidades condicionadas $P(M1|N1)$, $P(M1|N2)$ y $P(M1|N3)$.

Con el Experimento 2 se busca hacer un análisis de sensibilidad para reconocer cuáles son los parámetros que más influyen en las variables de salida. Con esto, se examina cuál es el efecto que tienen las probabilidades de entrada sobre $P(N2|M1)$ y $P(N1|M2)$.

En este caso, las tres probabilidades condicionadas

$$P(M1|N1) = 0.4 \quad , \quad P(M1|N2) = 0.5 \quad , \quad P(M1|N3) = 0.2$$

se toman considerando una variación del $\pm 10\%$ respecto a estos valores. Se implementa mediante un diseño factorial 3^3 , en el que cada probabilidad condicionada toma tres niveles:

$$P(M1|N1) \in \{0.36, 0.40, 0.44\}$$

$$P(M1|N2) \in \{0.45, 0.50, 0.55\}$$

$$P(M1|N3) \in \{0.18, 0.20, 0.22\}$$

Para hacer el análisis de sensibilidad, se utiliza un modelo ANOVA para cada probabilidad de salida, $P(N2|M1)$ y $P(N1|M2)$. Se implementará un diseño factorial 3^3 sin replicaciones porque se ha optado por incluir la fuente de “ruido” (que es la incertidumbre de las probabilidades a

priori $P(N_i)$) dentro de cada escenario con una simulación de Monte Carlo. Las probabilidades condicionadas $P(M1|N1)$, $P(M1|N2)$ y $P(M1|N3)$ son los factores con tres niveles cada uno.

El término del error residual se estima a partir de la interacción de tercer orden, que se asume que es despreciable. El modelo cuenta con 26 grados de libertad, de los cuales 6 se destinan a los efectos principales, 12 a cada interacción de segundo orden y los 8 restantes, que son los que le corresponderían a la interacción de tercer orden, son los que se le asignan al error residual.

El diseño factorial genera 27 combinaciones posibles de parámetros condicionados. Aunque los niveles de los factores son numéricos, se tratan como variables cualitativas usando “`factor()`” porque el ANOVA no busca una relación matemática, sino comparar las medias de los tres grupos. Se incluyen las interacciones de segundo orden para que la interacción de orden superior, que no se considera relevante, estime la varianza del error residual, calculando los estadísticos F y los p-valores.

Dentro de cada uno de los 27 escenarios, se ejecuta una simulación de Monte Carlo con 10.000 iteraciones. Se seguirán los siguientes pasos:

1. *Generación de probabilidades iniciales.*

Al igual que en el primer experimento, se simulan “`probN1`” y “`probN2`” de forma independiente con distribución uniforme en $[0, 1]$. Se ajusta “`probN2`” para garantizar que las probabilidades de los tres nervios sumen 1 y se calcula “`probN3`” de la forma: $probN3 = 1 - probN1 - probN2$.

2. *Probabilidades musculares.*

Se aplica la Ley de la Probabilidad Total, esta vez teniendo en cuenta las fluctuaciones en las probabilidades condicionadas mencionadas anteriormente. Para cada simulación se obtienen:

$$P(M1) = (probN1 \cdot P(M1|N1)) + (probN2 \cdot P(M1|N2)) + (probN3 \cdot P(M1|N3))$$

$$P(M2) = (probN1 \cdot P(M2|N1)) + (probN2 \cdot P(M2|N2)) + (probN3 \cdot P(M2|N3))$$

siendo $P(M2|N_k) = 1 - P(M1|N_k)$ para cada nervio N_k .

3. *Cálculo de las probabilidades condicionadas de interés.*

Haciendo uso del Teorema de Bayes, se obtienen las probabilidades de interés.

Probabilidad condicionada de que N2 haya sido estimulado dado que M1 se contrae:

$$P(N2 | M1) = \frac{probN2 \cdot P(M1|N2)}{P(M1)}$$

Probabilidad condicionada de que N1 haya sido estimulado dado que M2 se contrae:

$$P(N1 | M2) = \frac{probN1 \cdot P(M2|N1)}{P(M2)}$$

4. *Obtención de resultados para cada escenario.*

Se calcula la media de las 10.000 simulaciones para cada variable respuesta y los valores medios de $P(N2|M1)$ y $P(N1|M2)$ se añaden a la matriz del diseño factorial.

5. *Análisis de sensibilidad con ANOVA.*

Se realizan dos ANOVA, uno para $P(N2|M1)$ y otro para $P(N1|M2)$. Se toman como factores las probabilidades condicionadas $P(M1|N1)$, $P(M1|N2)$ y $P(M1|N3)$, por ser las independientes. El modelo evalúa efectos principales y de interacción de segundo orden.

Se crea un gráfico de dispersión que estudia la correlación entre $P(N2|M1)$ y $P(N1|M2)$. Con él también pueden observarse si las combinaciones de parámetros que afectan a una de las salidas favorecen también a la otra; es decir, ver si las salidas tienden a aumentar o disminuir juntas.

Para terminar con el análisis, se añaden gráficos de efectos principales que acompañan visualmente al ANOVA para ver el efecto de cada factor por separado. También se usan gráficos de interacción con el fin de ver si un factor depende de otro o no. Además, se calculan los intervalos de confianza (95%).

6. *Diagnosis del modelo ANOVA.*

Finalmente, se usan gráficos Q-Q para evaluar la similitud con una distribución aproximadamente normal. Se utilizan gráficos de residuos frente a valores ajustados para comprobar que no existen patrones y demostrar la hipótesis de homocedasticidad.

4.1.3. Experimento 3

En el tercer y último experimento se van a comparar dos enfoques distintos para la propagación de incertidumbres. Por un lado, la simulación de Monte Carlo, que será la que se tome como método de referencia. Por otro, el método de perturbación de Taylor de primer orden, que aproxima la variabilidad de las entradas sobre las salidas con un desarrollo matemático directo.

Las dos probabilidades de interés vuelven a ser $P(N2|M1)$ y $P(N1|M2)$.

Con esta comparación se pretende medir hasta qué punto el método de Taylor reproduce correctamente la media, la varianza y el tiempo de ejecución obtenidos con Monte Carlo.

El contexto de este experimento vuelve a ser el mismo de los dos anteriores. Se estudia la respuesta de los músculos M1 y M2 cuando se estimulan los nervios N1, N2 y N3. Las probabilidades a priori se consideran fijas en este caso, de nuevo tomando los resultados de Schirmer et al. (2011) como referencia para asignar sus valores.

$$P(N1) = 0.3 \quad , \quad P(N2) = 0.2 \quad , \quad P(N3) = 0.5$$

Las probabilidades condicionadas se toman dentro del rango del $\pm 10\%$ respecto a sus valores originales:

$$P(M1|N1) \in [0.36, 0.44]$$

$$P(M1|N2) \in [0.45, 0.55]$$

$$P(M1|N3) \in [0.18, 0.22]$$

El motivo por el que en este experimento se mantienen constantes las probabilidades de estimulación de los nervios y solo se introduce incertidumbre en las condicionadas, es aislar una única fuente de incertidumbre. Haciendo el experimento de este modo, se pueden comparar los dos métodos sobre un mismo modelo controlado. Si las varianzas no coinciden, la diferencia que se dé podrá atribuirse al método que se ha usado, y no habrá duda de que la razón sea que hay mucha incertidumbre al mismo tiempo.

Otra causa es que el método de Taylor funciona cuando hay pequeñas variaciones alrededor de valores conocidos, y se puede complicar si existen dependencias como las que se dan entre $P(N1)$, $P(N2)$ y $P(N3)$. Como estas tres últimas probabilidades son dependientes, habría que tener en cuenta los términos de covarianza, y esto aumenta el nivel de dificultad cuando se calculan las derivadas parciales.

Los pasos que se van a seguir son:

1. *Simulación de Monte Carlo con 10.000 iteraciones.*

El primer paso es asignarles los valores correspondientes a las probabilidades de $N1$, $N2$ y $N3$. Igualmente, se define el rango en el que van a variar las probabilidades condicionadas, que va a ser un $\pm 10\%$ de sus valores medios, como se ha comentado al inicio. $P(M1|N1)$, $P(M1|N2)$ y $P(M1|N3)$ generarán valores aleatorios dentro de estos rangos, siguiendo una distribución uniforme.

Para cada iteración, se aplica la Ley de la Probabilidad Total para obtener $P(M1)$ y $P(M2)$:

$$P(M1) = (0.3 \cdot P(M1|N1)) + (0.2 \cdot P(M1|N2)) + (0.5 \cdot P(M1|N3))$$

$$P(M2) = (0.3 \cdot P(M2|N1)) + (0.2 \cdot P(M2|N2)) + (0.5 \cdot P(M2|N3))$$

siendo $P(M2|N_k) = 1 - P(M1|N_k)$ para cada nervio N_k .

Para las probabilidades a posteriori, se usa de nuevo el Teorema de Bayes:

$$P(N2|M1) = \frac{0.2 \cdot P(M1|N2)}{P(M1)}, \quad P(N1|M2) = \frac{0.3 \cdot P(M2|N1)}{P(M2)}$$

Para terminar, se guardan los resultados y se calculan la media, la varianza y el tiempo de ejecución.

2. Método de perturbación de Taylor de primer orden.

Se definen los valores medios de las probabilidades:

$$P(N1) = 0.3 \quad , \quad P(M1|N1) = 0.4$$

$$P(N2) = 0.2 \quad , \quad P(M1|N2) = 0.5$$

$$P(N3) = 0.5 \quad , \quad P(M1|N3) = 0.2$$

Al igual que en el paso anterior, las probabilidades $P(M1|N1)$, $P(M1|N2)$ y $P(M1|N3)$ van a tomar valores dentro del rango del $\pm 10\%$. También para implementar el método de Taylor, se definen las probabilidades condicionadas siguiendo una distribución uniforme.

La varianza para una distribución de este tipo, $U(a, b)$, es una fórmula estándar dada por la siguiente expresión:

$$\sigma_X^2 = \frac{(b - a)^2}{12}$$

donde X es igual a $P(MI|N_i)$. b es el valor máximo del rango (+10%) para esa probabilidad condicionada y a es el valor mínimo (-10%). Usando esta fórmula, se calculan las varianzas de entrada que se necesitan para propagar la incertidumbre.

El siguiente paso de este método es obtener las derivadas parciales de las funciones de salida, como se ha explicado en el capítulo 3.3. Se aplica la fórmula de propagación de incertidumbre de primer orden para estimar la varianza de las salidas:

$$\sigma_Y^2 \approx \sum_i \left(\frac{\partial f}{\partial X_i} \right)^2 \cdot \sigma_{X_i}^2$$

donde X_i son las probabilidades $P(MI|N_i)$. Las derivadas se evalúan en el punto medio.

Posteriormente, se registran y se guardan los valores de la media, varianza y el tiempo de ejecución.

3. Comparación de resultados.

Para comparar los resultados obtenidos con los dos métodos se construyen dos tablas, una para $P(N2|M1)$ y otra para $P(N1|M2)$. Habrá cuatro columnas: una que indica qué método es el que se ha usado, una segunda para la media y otras dos para la varianza y el tiempo de ejecución.

Como apoyo visual, para cada variable de salida se representa el histograma de Monte Carlo con la distribución simulada. Además, se superpone la curva normal obtenida con

los parámetros de Taylor: $N(\mu_{Taylor}, \sigma_{Taylor})$, donde $\sigma_{Taylor} = \sqrt{\sigma_{Taylor}^2}$.

De esta forma, se puede apreciar si el método de Taylor y el de Monte Carlo dan resultados similares. Con la varianza, puede analizarse si el método de perturbación de Taylor capta correctamente la dispersión de Monte Carlo. Comparando el tiempo de ejecución, se puede medir si existe una ganancia de eficiencia con el método de Taylor.

4. *Diagnosis de la simulación de referencia (Monte Carlo)*

La diagnosis del Experimento 3 es la misma que la del Experimento 1. Se usan gráficos Q-Q para examinar si se aproximan a una distribución normal las de $P(N2|M1)$ y $P(N1|M2)$. Se lleva a cabo un análisis de la media acumulada para revisar que las estimaciones se estabilizan al aumentar el número de simulaciones. Para terminar, se prueba la hipótesis de independencia con la función ACF y el test de Durbin-Watson.

El resultado que se obtiene con la simulación de Monte Carlo se consigue a base de un gran número de simulaciones aleatorias. Únicamente se le realiza una diagnosis al método de Monte Carlo porque deben validarse que sus resultados son factibles. Al final, este es el método que se toma como referencia en la comparación con la perturbación de Taylor, y se debe estar seguro de que es fiable.

Por el contrario, el método de Taylor es analítico. Esto quiere decir que su resultado deriva de usar una fórmula, es un cálculo matemático directo. No aplican convergencia ni independencia, se valida con la propia comparación con el método de Monte Carlo, ya validada.

4.2. Implementación del código en R

En esta sección se explican los códigos que se han utilizado para los tres experimentos. Se han implementado en R, que es un lenguaje de programación orientado al análisis estadístico.

Como se ha mencionado, para la manipulación y ejecución del código se ha utilizado la herramienta RStudio, el principal entorno de desarrollo integrado (IDE) para el lenguaje R. RStudio es de código abierto y está destinado a combinar los distintos componentes de R (consola, código fuente, gráficos, etc.) [16].

4.2.1. Experimento 1

El código del Experimento 1 se ha estructurado en tres bloques principales. En la primera parte se crean los valores aleatorios para las probabilidades de $N1$, $N2$ y $N3$ y se aplican la Ley de la Probabilidad Total y el Teorema de Bayes para calcular $P(N2|M1)$ y $P(N1|M2)$. En el segundo bloque se analizan los resultados con las medias de dispersión y los gráficos. La última parte es en la que se hace la diagnosis de Monte Carlo.

1. *Simulación de Monte Carlo*

4. Experimentos computacionales en R

El primer paso es fijar la semilla del generador para que los resultados que se obtengan se puedan reproducir. Gracias a ella, se puede crear exactamente la misma secuencia de números aleatorios cada vez que se ejecute el código.

```
set.seed(123)
```

Se crean los vectores “probN1” y “probN2” de 10.000 valores en el intervalo [0, 1] siguiendo una distribución uniforme, con la función “runif()”.

```
probN1 = runif(10000, min = 0, max = 1)
probN2 = runif(10000, min = 0, max = 1 - probN1)
probN3 = 1 - probN1 - probN2
```

El vector “probN2” se ajusta permitiéndole un valor máximo que depende de la probabilidad ya asignada para el N1. La razón por la que se hace es asegurar que la suma de ambas probabilidades no exceda nunca 1. Cuando se hayan fijado ya P(N1) y P(N2), se define “probN3” como la probabilidad restante.

Para comprobar visualmente que las probabilidades que se han generado son correctas, se combinan los tres vectores creando una matriz de 10.000×3. La suma de cada fila tiene que sumar 1. Esta matriz se genera usando la función “cbind()”, y se muestran las primera seis filas con “head()”.

```
head(cbind(probN1, probN2, probN3))
```

Para este primer experimento, se asumen conocidos los valores de las probabilidades condicionadas. Dado que M1 y M2 son eventos mutuamente excluyentes, se cumple $P(M2|N_i) = 1 - P(M1|N_i)$.

Todas estas probabilidades condicionadas se definen en el código de la siguiente forma:

```
P_M1_N1_dato = 0.4
P_M1_N2_dato = 0.5
P_M1_N3_dato = 0.2
P_M2_N1_dato = 1 - P_M1_N1_dato
P_M2_N2_dato = 1 - P_M1_N2_dato
P_M2_N3_dato = 1 - P_M1_N3_dato
```

Para el cálculo de P(M1), primero se crea un vector de longitud 10.000 inicializado en ceros mediante la función “numeric()”. En él se reservará espacio para los 10.000 valores simulados de P(M1).

Se implementa en R la Ley de la Probabilidad Total, descrita en la sección 3.1.1 de este trabajo, para calcular $P(M1)$. Para cada simulación i , se calcula mediante el bucle *for* cómo varía $P(M1)$ con diferentes combinaciones de probabilidades “probN1”, “probN2” y “probN3”.

De este modo, habremos propagado la incertidumbre de las probabilidades iniciales a $P(M1)$ por Monte Carlo.

```
P_M1 = numeric(10000)

for (i in 1:10000) {

  P_M1[i] = (probN1[i] * P_M1_N1_dato) + (probN2[i] *
    P_M1_N2_dato) + (probN3[i] * P_M1_N3_dato) }
```

Para el cálculo de la probabilidad $P(N2|M1)$, al igual que antes, previamente debe crearse un vector de longitud 10.000 para almacenar esta variable respuesta.

Se implementa el Teorema de Bayes para calcular la probabilidad de que se haya estimulado N2 dado que M1 se contrajo primero ($P(N2|M1)$). Para verificar que los valores son válidos, se muestran los primeros resultados con la función “head()” ya mencionada.

```
P_N2_M1 = numeric(10000)

for (i in 1:10000) {

  P_N2_M1[i] = (probN2[i] * P_M1_N2_dato) / P_M1[i] }

head(P_N2_M1)
```

Estos mismos pasos se repiten de forma análoga para calcular $P(M2)$ y $P(N1|M2)$, que es la segunda variable de interés:

```
P_M2 = numeric(10000)

for (i in 1:10000) {

  P_M2[i] = (probN1[i] * P_M2_N1_dato)
    + (probN2[i] * P_M2_N2_dato)
    + (probN3[i] * P_M2_N3_dato) }

P_N1_M2 = numeric(10000)

for (i in 1:10000) {

  P_N1_M2[i] = (probN1[i] * P_M2_N1_dato) / P_M2[i] }

head(P_N1_M2)
```

4. Experimentos computacionales en R

Después de haber obtenido las simulaciones para $P(N2|M1)$ y $P(N1|M2)$, se procede al análisis de los resultados.

2. *Análisis de resultados*

Se calculan la media, la varianza y los cuartiles. Con la varianza se mide el valor promedio de la distribución que se ha simulado y la varianza cuantifica la dispersión respecto a la media. Se calculan los percentiles 25, 50 (mediana) y 75 de los cuartiles para describir la forma de la distribución.

Estos cálculos se hacen para $P(N2|M1)$ y $P(N1|M2)$.

```
mean_P_N2_M1 = mean(P_N2_M1)

var_P_N2_M1 = var(P_N2_M1)

quantiles_P_N2_M1 = quantile(P_N2_M1, probs = c(0.25, 0.5, 0.75))

mean_P_N1_M2 = mean(P_N1_M2)

var_P_N1_M2 = var(P_N1_M2)

quantiles_P_N1_M2 = quantile(P_N1_M2, probs = c(0.25, 0.5, 0.75))
```

Una vez calculados, se muestran los resultados con la función “`print()`”. Para concatenar texto y valores numéricos para generar mensajes legibles, se usa “`paste()`”.

```
print(paste("Media de P(N2|M1):", mean_P_N2_M1))
print(paste("Varianza de P(N2|M1):", var_P_N2_M1))
print("Cuartiles de P(N2|M1):") print(quantiles_P_N2_M1)
print(paste("Media de P(N1|M2):", mean_P_N1_M2))
print(paste("Varianza de P(N1|M2):", var_P_N1_M2))
print("Cuartiles de P(N1|M2):") print(quantiles_P_N1_M2)
```

Además, se usa “`summary()`” para crear un resumen de los estadísticos descriptivos estándar y detectar valores atípicos.

```
summary (P_N2_M1)

summary (P_N1_M2)
```

Con “`hist()`” se representa la frecuencia de los valores en intervalos definidos. Se establecen 50 intervalos (`breaks = 50`) para poder tener más detalle de la forma de la distribución.

```
hist(P_N2_M1,
     breaks = 50, col = "lightblue",
```

```
main = "Distribución de P(N2|M1)", xlab = "P(N2|M1)")  
hist(P_N1_M2,  
     breaks = 50, col = "lightgreen",  
     main = "Distribución de P(N1|M2)", xlab = "P(N1|M2)")
```

Se estudia la relación lineal entre $P(N2|M1)$ y $P(N1|M2)$ con el coeficiente de correlación de Pearson ($cor(x, y)$).

Para revisar las posibles dependencias se usa el gráfico de dispersión o *scatter plot* utilizando “ $plot()$ ”. La función “ $rgb(r, g, b, alfa)$ ” de R crea un color basándose en el modelo RGB (Rojo, Verde, Azul) e introduce transparencia (alfa) en los puntos para que no haya tanto solapamiento visual.

```
cor_value = cor(P_N2_M1, P_N1_M2)  
  
print(paste("Correlación entre P(N2|M1) y P(N1|M2):", round(cor_value,  
4)))  
  
plot(P_N2_M1, P_N1_M2,  
     main = "Correlación entre P(N2|M1) y P(N1|M2)",  
     xlab = "P(N2|M1)", ylab = "P(N1|M2)",  
     pch = 20, col = rgb(0.1, 0.3, 0.7, 0.3))
```

Se calcula asimismo el coeficiente de variación (CV). Este coeficiente mide la dispersión, expresada como un porcentaje, de un conjunto de datos respecto a su media. Sirve para sopesar cómo de grande o pequeña es la variabilidad. La fórmula del CV es: $CV = \frac{\text{Desviación típica}}{\text{Media}}$.

Con la función “ $sd()$ ” se calcula la desviación típica de las probabilidades de salida. Después, se imprimen los resultados como porcentaje.

```
CV_N2_M1 = sd(P_N2_M1) / mean(P_N2_M1)  
CV_N1_M2 = sd(P_N1_M2) / mean(P_N1_M2)  
cat("CV para P(N2|M1):", CV_N2_M1 * 100, "%\n")  
cat("CV para P(N1|M2):", CV_N1_M2 * 100, "%\n")
```

Para terminar, se calculan los intervalos de confianza para $P(N2|M1)$ y $P(N1|M2)$. Para conseguir un intervalo de confianza del 95% se usa “ $quantile()$ ”, que calcula los percentiles 2.5% y 97.5%. El primero representa el límite inferior y el segundo, el límite superior. Esto se repite para las dos probabilidades a posteriori. Después, se imprimen los resultados.

```
IC_95_N2_M1 = quantile(P_N2_M1, probs = c(0.025, 0.975))  
IC_95_N1_M2 = quantile(P_N1_M2, probs = c(0.025, 0.975))  
print(IC_95_N2_M1)
```

4. Experimentos computacionales en R

```
print(IC_95_N1_M2)
```

3. *Diagnosis de la simulación de Monte Carlo*

Con la diagnosis del Experimento 1 se pretende probar convergencia e independencia. La normalidad se estudia para ver qué forma tiene la incertidumbre de salida, pero no porque sea un requisito que se tenga que cumplir. Es solo una posible característica del resultado de Monte Carlo.

Los gráficos Q-Q se construyen con “`qqnorm()`” y con “`qqline()`”. El primero grafica los cuantiles muestrales frente a los teóricos de una normal estándar. El segundo dibuja la línea de referencia que, si los puntos se alinean, puede asegurar que la distribución es aproximadamente normal. Esto se realiza para $P(N2|M1)$ y $P(N1|M2)$.

Para mostrar los cuatro gráficos a la vez, se configura la ventana gráfica con la función “`par()`”.

```
par(mfrow = c(2, 2))

qqnorm(P_N2_M1, main = "Gráfico Q-Q para P(N2|M1)", pch = 20)

qqline(P_N2_M1, col = "steelblue", lwd = 2)

qqnorm(P_N1_M2, main = "Gráfico Q-Q para P(N1|M2)", pch = 20)

qqline(P_N1_M2, col = "maroon3", lwd = 2)
```

“`cumsum()`” es una función que obtiene la suma acumulada de los valores simulados. “`seq_along()`” genera una secuencia desde 1 hasta la longitud del vector. Si se divide la suma acumulada por el número de simulaciones, se consigue la media acumulada, que es la que se usa para demostrar la convergencia.

Posteriormente, se representa esta media acumulada en función del número de simulaciones. Si se ve una línea que se estabiliza, quiere decir que el número de simulaciones es suficiente para estimar el valor esperado.

Se restablece la ventana gráfica a un único gráfico haciendo nuevamente uso de la función “`par()`”.

```
media_acum_N2_M1 = cumsum(P_N2_M1) / seq_along(P_N2_M1)

plot(media_acum_N2_M1, type = "l", col = "steelblue", lwd = 2,
      main = "Convergencia de la Media Acumulada de P(N2|M1)",
      xlab = "Número de Simulaciones", ylab = "Media Acumulada")
```

```
media_acum_N1_M2 = cumsum(P_N1_M2) / seq_along(P_N1_M2)
plot(media_acum_N1_M2, type = "l", col = "maroon3", lwd = 2,
      main = "Convergencia de la Media Acumulada de P(N1|M2)",
      xlab = "Número de Simulaciones", ylab = "Media Acumulada")

par(mfrow = c(1, 1))
```

“`acf()`” calcula y grafica la autocorrelación de una serie. En una simulación de Monte Carlo, se espera que solo haya un pico importante en el lag 0. Si hay en lags mayores, significaría que existen simulaciones dependientes.

```
par(mfrow = c(1, 2))
acf(P_N2_M1, main = "Análisis Independencia P(N2|M1)")
acf(P_N1_M2, main = "Análisis Independencia P(N1|M2)")
par(mfrow = c(1, 1))
```

Para complementar la función ACF, se realiza el test de Durbin-Watson “`dwtest()`”. Esta función estudia la autocorrelación de los residuos. Si el estadístico Durbin-Watson es cercano a 2, sugiere ausencia de autocorrelación.

Para poder hacer uso de la función “`dwtest()`”, es necesario cargar previamente la librería “`lmtest()`”. En este caso, se ha ajustado a un modelo trivial (~ 1) para aplicar el test directamente a las series simuladas.

```
library(lmtest)
dw_test_N2_M1 = dwtest(P_N2_M1 ~ 1)
print(dw_test_N2_M1)

dw_test_N1_M2 = dwtest(P_N1_M2 ~ 1)
print(dw_test_N1_M2)
```

En el capítulo 5 de la memoria se comentan los resultados que se han obtenido.

4.2.2. Experimento 2

Como en el Experimento 1, el código del Experimento 2 se va a dividir en tres bloques. En el primer bloque se crea el diseño factorial 3^3 . Para cada combinación se ejecuta una simulación de Monte Carlo interna con la que se obtienen $P(N2|M1)$ y $P(N1|M2)$. En la segunda parte se realiza un ANOVA sobre estas medias, con el que se estudia el impacto de cada factor y sus interacciones. Para terminar, en el último bloque es donde se lleva a cabo la diagnosis del ANOVA.

1. Diseño factorial y simulación Monte Carlo

4. Experimentos computacionales en R

El primer paso es fijar la semilla del generador, al igual que en el primer experimento.

```
set.seed(123)
```

Se construye la estructura del diseño factorial 3^3 . Los factores son las probabilidades condicionadas y a cada uno se le asigna tres niveles que se corresponden con la variación del $\pm 10\%$. Esto último se implementa usando los multiplicadores 0.9 y 1.1:

```
niveles_PM1N1 = c(0.4 * 0.9, 0.4, 0.4 * 1.1)
niveles_PM1N2 = c(0.5 * 0.9, 0.5, 0.5 * 1.1)
niveles_PM1N3 = c(0.2 * 0.9, 0.2, 0.2 * 1.1)
```

La matriz de diseño se construye con “`expand.grid()`”. Esta función es la que crea los 27 escenarios posibles. Con “`factor()`” cada columna se trata como cualitativa dentro del diseño; es decir, como un conjunto de grupos distintos y no como una variable numérica.

```
design_matriz = expand.grid(
  F_PM1N1 = factor(niveles_PM1N1),
  F_PM1N2 = factor(niveles_PM1N2),
  F_PM1N3 = factor(niveles_PM1N3) )
```

Antes de iniciar la simulación de Monte Carlo, se crean dos vectores, uno por cada variable respuesta, que almacena la media resultante de las 10.000 simulaciones para cada escenario. El tamaño 27 corresponde al número total de combinaciones factoriales.

Además, se define el número de repeticiones para asegurar que para cada escenario se ejecutarán 10.000 simulaciones de Monte Carlo.

```
resultados_N2_M1 = numeric(27)
resultados_N1_M2 = numeric(27)
n_sims = 10000
```

Se guarda el tiempo de inicio antes de entrar en el bucle principal, para poder calcular la duración total de la simulación.

```
start_time_montecarlo = Sys.time()
```

A continuación, se implementa el bucle principal de los 27 escenarios. Este bucle externo recorre cada fila de la matriz de diseño o, lo que es lo mismo, cada combinación de niveles.

```
for (i in 1:nrow(design_matriz)) {
```

Para cada escenario:

- Se extraen las probabilidades condicionadas fijas para ese escenario. Las funciones “`as.character()`” y “`as.numeric()`” se usan para convertir los factores en valores numéricos reales. Este proceso de dos pasos es necesario porque R almacena

internamente los factores como números enteros y una conversión directa a numérico devolvería estos índices en lugar de los valores originales.

```
P_M1_N1_fixed = as.numeric(as.character(design_matriz$F_PM1N1[i]))
P_M1_N2_fixed = as.numeric(as.character(design_matriz$F_PM1N2[i]))
P_M1_N3_fixed = as.numeric(as.character(design_matriz$F_PM1N3[i]))
```

- Se calculan las probabilidades complementarias sabiendo que $P(M2|N_i) = 1 - P(M1|N_i)$.

```
P_M2_N1_fixed = 1 - P_M1_N1_fixed
P_M2_N2_fixed = 1 - P_M1_N2_fixed
P_M2_N3_fixed = 1 - P_M1_N3_fixed
```

En cada iteración del bucle externo se ejecuta una simulación de Monte Carlo independiente con 10.000 repeticiones:

- Se generan las probabilidades base para los nervios tal y como se planteó en el Experimento 1, haciendo uso de la función “runif()”.

```
probN1_dos = runif(n_sims, min = 0, max = 1)
probN2_dos = runif(n_sims, min = 0, max = 1 - probN1_dos)
probN3_dos = 1 - probN1_dos - probN2_dos
```

- Se inicializan los vectores que almacenarán los resultados de cada simulación antes de calcular la media. Se usará la función “numeric()”, también mencionada en el experimento anterior.

```
P_M1_dos = numeric(n_sims)
P_M2_dos = numeric(n_sims)
P_N2_M1_dos = numeric(n_sims)
P_N1_M2_dos = numeric(n_sims)
```

- Se crea el bucle interno de la simulación de Monte Carlo. En él se implementan la Ley de la Probabilidad Total para calcular $P(M1)$ y $P(M2)$, y el Teorema de Bayes para conseguir las probabilidades $P(N2|M1)$ y $P(N1|M2)$.

```
for (j in 1:n_sims) {
  # Ley de probabilidad total
  P_M1_dos[j] = (probN1_dos[j] * P_M1_N1_fixed)
               + (probN2_dos[j] * P_M1_N2_fixed)
               + (probN3_dos[j] * P_M1_N3_fixed)
  P_M2_dos[j] = (probN1_dos[j] * P_M2_N1_fixed)
```

4. Experimentos computacionales en R

```
      + (probN2_dos[j] * P_M2_N2_fixed)
      + (probN3_dos[j] * P_M2_N3_fixed)

# Teorema de Bayes
P_N2_M1_dos[j] = (probN2_dos[j] * P_M1_N2_fixed) / P_M1_dos[j]
P_N1_M2_dos[j] = (probN1_dos[j] * P_M2_N1_fixed) / P_M2_dos[j]
}
```

Una vez completadas las 10.000 simulaciones de un escenario, se guarda la media (`mean()`) como estimador central para cada probabilidad inversa.

Además, se imprime un mensaje de seguimiento con la función “`cat()`”. Después de esto, se finaliza el bucle externo.

```
resultados_N2_M1[i] = mean(P_N2_M1_dos)
resultados_N1_M2[i] = mean(P_N1_M2_dos)
cat(paste("Completado escenario", i, "de 27\n"))
}
```

Después de cerrar el bucle principal, se calcula el tiempo total de ejecución para conocer la duración exacta de todo el proceso de simulación.

```
end_time_montecarlo = Sys.time()
montecarlo_time = end_time_montecarlo - start_time_montecarlo
print(paste("Tiempo de ejecución Monte Carlo:", montecarlo_time))
```

Por último, se añaden dos nuevas columnas con las medias obtenidas a la matriz de diseño factorial y se imprime la tabla final con las 27 combinaciones y sus resultados.

```
design_matriz$P_N2_M1_mean = resultados_N2_M1
design_matriz$P_N1_M2_mean = resultados_N1_M2
print("Resultados del Diseño Factorial (Medias)")
print(design_matriz)
```

2. *Análisis de resultados*

Como se ha mencionado a lo largo del trabajo, la finalidad del ANOVA es reconocer qué parámetros son los que influyen más en las probabilidades a posteriori.

Para ajustar el modelo de análisis de varianza se usa la función “`aov()`”.

Empezando por la probabilidad de interés $P(N2|M1)$, la fórmula $((F_{PM1N1} + F_{PM1N2} + F_{PM1N3})^2)$ indica los efectos principales de los tres factores y todas las interacciones de dos factores. No incluye la interacción triple para simplificar y evitar un modelo saturado,

permitiendo que la interacción de orden superior se utilice para estimar el término de error residual.

Para especificar que las variables están en la matriz factorial se usa “data=design_matriz”.

```
aov_N2_M1 = aov(P_N2_M1_mean ~ (F_PM1N1 + F_PM1N2 + F_PM1N3)^2,  
               data = design_matriz)
```

El resumen del análisis se consigue con “summary()”, que devuelve una lista donde el primer elemento ([[1]]) es la tabla ANOVA con sumas de cuadrados, grados de libertad, valores F y p-valores.

Posteriormente, se imprime el resultado para visualizarlo.

```
summary_aov1 = summary(aov_N2_M1)  
tabla_anova_1 = summary_aov1[[1]]  
cat("\n Tabla ANOVA para P(N2|M1) \n")  
print(tabla_anova_1)
```

Este mismo procedimiento se repite para la segunda variable respuesta P(N1|M2).

```
aov_N1_M2 = aov(P_N1_M2_mean ~ (F_PM1N1 + F_PM1N2 + F_PM1N3)^2,  
               data = design_matriz)  
summary_aov2 = summary(aov_N1_M2)  
tabla_anova_2 = summary_aov2[[1]]  
cat("\n Tabla ANOVA para P(N1|M2) \n")  
print(tabla_anova_2)
```

De esta manera, se obtienen dos tablas independientes, una por cada variable respuesta.

Se estudia la correlación de P(N2|M1) y P(N1|M2). Como en el Experimento 1, se calcula primero el coeficiente de correlación de Pearson con “cor(x, y)”. Con el coeficiente ya calculado, se imprime el resultado redondeado a 4 decimales usando “round(cor_exp2, 4)”. Después, se vuelve a emplear un gráfico de dispersión para visualizar las posibles dependencias (plot()).

```
cor_exp2=cor(design_matriz$P_N2_M1_mean, design_matriz$P_N1_M2_mean)  
print(paste("Coeficiente de correlación:", round(cor_exp2, 4)))  
plot(design_matriz$P_N2_M1_mean, design_matriz$P_N1_M2_mean,  
     main = "Correlación entre Respuestas Promedio (Exp. 2)",  
     xlab = "Media de P(N2|M1)", ylab = "Media de P(N1|M2)",  
     pch = 20, col = "lightpink")
```

4. Experimentos computacionales en R

Para terminar con este bloque, se calculan los intervalos de confianza de los factores. Se van a utilizar dos funciones “`ICplot()`” e “`interIC()`”, que se proporcionaron en la asignatura de *Diseños de Experimentos y Modelos de Regresión* por la universidad. Obtienen automáticamente los gráficos y los intervalos de confianza de los efectos principales y las interacciones, respectivamente.

Los primeros, los gráficos de efectos principales, trazan el valor medio de la variable respuesta para cada nivel de un factor: una mayor diferencia de altura entre los puntos indica mayor influencia del factor sobre la variable.

Por otro lado, los gráficos de interacción muestran si el efecto de un factor depende del nivel de otro (interacción no aditiva). “`interIC()`” dibuja líneas para cada nivel del factor que corresponda. El no paralelismo de las líneas indica interacción.

Se generan los dos gráficos de interacción de $P(N2|M1)$. No se generan gráficos de interacción para $P(N1|M2)$ ya que el ANOVA (resultado presentado en la sección 5.2) no reveló interacciones estadísticamente significativas para esta variable respuesta.

Los dos tipos de gráficos, el de efectos principales y el de interacciones, añaden además los intervalos de confianza de cada nivel del factor.

Como paso previo, deben cargarse las dos funciones usando “`source()`”. Al terminar, se imprimen los resultados de las tablas.

```
source("ICplot.R")      source("interIC.R")

cat("IC para Efectos Principales sobre P(N2|M1)\n")

IC_F1_N2M1 = ICplot(aov_N2_M1, "F_PM1N1")
IC_F2_N2M1 = ICplot(aov_N2_M1, "F_PM1N2")
IC_F3_N2M1 = ICplot(aov_N2_M1, "F_PM1N3")

print(IC_F1_N2M1)      print(IC_F2_N2M1)      print(IC_F3_N2M1)

cat("IC para Interacciones sobre P(N2|M1)\n")

IC_int12_N2M1 = interIC(aov_N2_M1, "F_PM1N1", "F_PM1N2")
IC_int13_N2M1 = interIC(aov_N2_M1, "F_PM1N1", "F_PM1N3")

print(IC_int12_N2M1)  print(IC_int13_N2M1)

cat("IC para Efectos Principales sobre P(N1|M2)\n")

IC_F1_N1M2 = ICplot(aov_N1_M2, "F_PM1N1")
IC_F2_N1M2 = ICplot(aov_N1_M2, "F_PM1N2")
IC_F3_N1M2 = ICplot(aov_N1_M2, "F_PM1N3")
```

```
print(IC_F1_N1M2)      print(IC_F2_N1M2)      print(IC_F3_N1M2)
```

3. *Diagnosis del modelo ANOVA*

Como se comentó en la sección 3.4.3, los residuos del ANOVA deben cumplir normalidad y homocedasticidad (que las varianzas sean iguales).

Se vuelve a usar “`par()`” para mostrar los cuatro gráficos a la vez. Al terminar esta parte del código, vuelve a usarse para restablecer la ventana gráfica.

Para comprobar la normalidad de los residuos, se realiza un gráfico Q-Q para cada variable respuesta: “`qqnorm()`” genera un gráfico cuantil-cuantil comparando la distribución de los residuos con una normal teórica; mientras que “`qqline()`” añade la línea teórica de normalidad. Desviaciones grandes indican falta de ajuste.

```
par(mfrow = c(2, 2))

qqnorm(aov_N2_M1$residuals,
       main = "Q-Q Plot de los Residuos de P(N2|M1)")
qqline(aov_N2_M1$residuals, col = "darkolivegreen1")

qqnorm(aov_N1_M2$residuals,
       main = "Q-Q Plot de los Residuos de P(N1|M2)")
qqline(aov_N1_M2$residuals, col = "lightsteelblue1")
```

Con un gráfico de residuos frente a los valores ajustados para $P(N2|M1)$ y $P(N1|M2)$ se demuestra homocedasticidad. Si la dispersión de los residuos aumenta o disminuye con el valor ajustado, la varianza no es constante y existe heterocedasticidad.

La línea horizontal en cero (`abline(h = 0)`) ayuda a visualizar la simetría de los residuos.

```
plot(aov_N2_M1$fitted.values, aov_N2_M1$residuals,
     main = "Residuos vs. Valores Ajustados P(N2|M1)",
     xlab = "Valores Ajustados", ylab = "Residuos")
abline(h = 0, col = "darkolivegreen1", lty = 2)

plot(aov_N1_M2$fitted.values, aov_N1_M2$residuals,
     main = "Residuos vs. Valores Ajustados P(N1|M2)",
     xlab = "Valores Ajustados", ylab = "Residuos")
abline(h = 0, col = "lightsteelblue1", lty = 2)

par(mfrow = c(1, 1))
```

En el capítulo 5 de la memoria se comentarán los resultados que se han obtenido al ejecutar el código.

4.2.3. Experimento 3

Por último, el código del Experimento 3 se organiza en cuatro bloques. El primer bloque define las probabilidades de entrada y se calculan $P(N2|M1)$ y $P(N1|M2)$ con una simulación de Monte Carlo. En el segundo bloque se implementa la expansión de Taylor de primer orden para estimar la propagación de incertidumbre con este método. En el tercero, se comparan los resultados de los dos métodos. En la última parte se realiza la diagnosis de la simulación de Monte Carlo, tomada como referencia.

1. Simulación de Monte Carlo

Al igual que en los experimentos anteriores, el primer paso es fijar la semilla del generador de números aleatorios para que los resultados sean reproducibles (`set.seed()`).

Previo a la implementación propiamente dicha del código del experimento, se captura el tiempo de inicio para calcular posteriormente el tiempo de ejecución de Monte Carlo.

```
set.seed(123)
start_time_mc = Sys.time()
```

Se definen los valores de las probabilidades $P(N1)$, $P(N2)$ y $P(N3)$. Esta vez se mantienen fijas todo el proceso.

```
probN1_MC = 0.3
probN2_MC = 0.2
probN3_MC = 0.5
```

Se generan las probabilidades condicionadas usando nuevamente “`runif()`”. Se crean 10.000 valores para cada una en el rango de $\pm 10\%$ de sus valores originales. También se calculan las probabilidades complementarias $P(M2|N_i)$.

```
P_M1_N1_MC = runif(10000, min = 0.36, max = 0.44)
P_M1_N2_MC = runif(10000, min = 0.45, max = 0.55)
P_M1_N3_MC = runif(10000, min = 0.18, max = 0.22)
P_M2_N1_MC = 1 - P_M1_N1_MC
P_M2_N2_MC = 1 - P_M1_N2_MC
P_M2_N3_MC = 1 - P_M1_N3_MC
```

Como se ha realizado en experimentos anteriores, se inicializa primero un vector numérico de longitud 10.000 con ceros. En cada iteración, se aplica la Ley de la Probabilidad Total para obtener $P(M1)$. Para calcular la probabilidad de $P(N2|M1)$, se vuelve a hacer uso del Teorema de Bayes.

```
P_M1_MC = numeric(10000)
for (i in 1:10000) {
    P_M1_MC[i] = (probN1_MC * P_M1_N1_MC[i])
                + (probN2_MC * P_M1_N2_MC[i])
                + (probN3_MC * P_M1_N3_MC[i]) }
P_N2_M1_MC = numeric(10000)
for (i in 1:10000) {
    P_N2_M1_MC[i] = (probN2_MC * P_M1_N2_MC[i]) / P_M1_MC[i] }
```

Este mismo procedimiento se repite para las probabilidades $P(M2)$ y $P(N1|M2)$:

```
P_M2_MC = numeric(10000)
for (i in 1:10000) {
    P_M2_MC[i] = (probN1_MC * P_M2_N1_MC[i])
                + (probN2_MC * P_M2_N2_MC[i])
                + (probN3_MC * P_M2_N3_MC[i]) }
P_N1_M2_MC = numeric(10000)
for (i in 1:10000) {
    P_N1_M2_MC[i] = (probN1_MC * P_M2_N1_MC[i]) / P_M2_MC[i] }
```

Se calculan la media, la varianza y el tiempo de ejecución de la simulación de Monte Carlo.

```
mean_P_N2_M1_MC = mean(P_N2_M1_MC)
var_P_N2_M1_MC = var(P_N2_M1_MC)
mean_P_N1_M2_MC = mean(P_N1_M2_MC)
var_P_N1_M2_MC = var(P_N1_M2_MC)
end_time_mc = Sys.time()
time_mc = end_time_mc - start_time_mc
```

Al igual que se hizo en el Experimento 1, se calcula el coeficiente de varianza y se imprimen los resultados como porcentaje:

```
CV_N2_M1_MC = sd(P_N2_M1_MC) / mean(P_N2_M1_MC)
CV_N1_M2_MC = sd(P_N1_M2_MC) / mean(P_N1_M2_MC)
cat("CV (MC) para P(N2|M1):", CV_N2_M1_MC * 100, "%\n")
cat("CV (MC) para P(N1|M2):", CV_N1_M2_MC * 100, "%\n")
```

Para terminar con este primer bloque, se calculan los intervalos de confianza del 95% para $P(N2|M1)$ y $P(N1|M2)$. Se repite el código y la lógica que se usó en el primer experimento.

```
IC_95_N2_M1_MC = quantile(P_N2_M1_MC, probs = c(0.025, 0.975))
IC_95_N1_M2_MC = quantile(P_N1_M2_MC, probs = c(0.025, 0.975))
print(IC_95_N2_M1_MC)
print(IC_95_N1_M2_MC)
```

2. *Perturbación de Taylor (primer orden)*

Este bloque inicia con la definición de los valores (medias) de las probabilidades de estimulación de los nervios y de las probabilidades condicionadas.

De igual forma que con la simulación de Monte Carlo, se recoge el tiempo inicial para poder hacer posteriormente la comparación entre ambos métodos.

```
start_time_taylor = Sys.time()
P_N1_taylor = 0.3
P_N2_taylor = 0.2
P_N3_taylor = 0.5
P_M1_N1_taylor = 0.4
P_M1_N2_taylor = 0.5
P_M1_N3_taylor = 0.2
```

Se implementa la fórmula de la varianza de una distribución uniforme (ver sección 4.1.3 de este trabajo) para calcular la varianza de las entradas.

```
var_PM1N1_taylor = (0.44 - 0.36)^2 / 12
var_PM1N2_taylor = (0.55 - 0.45)^2 / 12
var_PM1N3_taylor = (0.22 - 0.18)^2 / 12
```

Se calculan primero las probabilidades $P(M1)$ y $P(M2)$ en el punto medio aplicando la Ley de la Probabilidad Total (P_{M1_taylor} y P_{M2_taylor}).

Después, se obtiene la media de $P(N2|M1)$ y $P(N1|M2)$ ($mean_N2_M1_taylor$ y $mean_N1_M2_taylor$) implementando el Teorema de Bayes. El resultado que se obtiene haciendo esto, es la media que el método de Taylor estima para las variables de salida.

```
P_M1_taylor = (P_N1_taylor * P_M1_N1_taylor)
              + (P_N2_taylor * P_M1_N2_taylor)
              + (P_N3_taylor * P_M1_N3_taylor)
```

```

P_M2_taylor = (P_N1_taylor * (1-P_M1_N1_taylor))
              + (P_N2_taylor * (1-P_M1_N2_taylor))
              + (P_N3_taylor * (1-P_M1_N3_taylor))
mean_N2_M1_taylor = (P_M1_N2_taylor * P_N2_taylor) / P_M1_taylor
mean_N1_M2_taylor = ((1-P_M1_N1_taylor) * P_N1_taylor) / P_M2_taylor

```

En este método se propaga la incertidumbre con las derivadas parciales de la función respecto a las variables de entrada. Se implementan en el código las derivadas parciales comentadas en la sección 3.3.1 para P(N2|M1) y P(N1|M2).

```

d_N2M1_d_p1 = - (P_M1_N2_taylor * P_N2_taylor * P_N1_taylor) /
              (P_M1_taylor^2)
d_N2M1_d_p2 = (P_N2_taylor * P_M1_taylor
              - P_M1_N2_taylor * P_N2_taylor * P_N2_taylor) /
              (P_M1_taylor^2)
d_N2M1_d_p3 = - (P_M1_N2_taylor * P_N2_taylor * P_N3_taylor) /
              (P_M1_taylor^2)

d_N1M2_d_p1 = (-P_N1_taylor * P_M2_taylor
              - (1-P_M1_N1_taylor) * P_N1_taylor * (-P_N1_taylor)) /
              (P_M2_taylor^2)
d_N1M2_d_p2 = ((1-P_M1_N1_taylor) * P_N1_taylor * P_N2_taylor) /
              (P_M2_taylor^2)
d_N1M2_d_p3 = ((1-P_M1_N1_taylor) * P_N1_taylor * P_N3_taylor) /
              (P_M2_taylor^2)

```

Teniendo ya la varianza de las entradas y las derivadas parciales calculadas, se implementa en el código la fórmula de propagación de incertidumbre: $\sigma_Y^2 \approx \sum_{i=1}^n \left(\frac{\partial f}{\partial X_i}\right)^2 \sigma_{X_i}^2$. Se evalúa en el punto medio de las entradas.

```

var_N2_M1_taylor = (d_N2M1_d_p1^2 * var_PM1N1_taylor)
                  + (d_N2M1_d_p2^2 * var_PM1N2_taylor)
                  + (d_N2M1_d_p3^2 * var_PM1N3_taylor)
var_N1_M2_taylor = (d_N1M2_d_p1^2 * var_PM1N1_taylor)
                  + (d_N1M2_d_p2^2 * var_PM1N2_taylor)
                  + (d_N1M2_d_p3^2 * var_PM1N3_taylor)

```

4. Experimentos computacionales en R

Al igual que en la simulación de Monte Carlo del primer bloque de esta sección, se calcula el tiempo de ejecución para el método de perturbación de Taylor con el objetivo de poder comparar ambos.

```
end_time_taylor = Sys.time()
time_taylor = end_time_taylor - start_time_taylor
```

3. Comparación de resultados

Se construyen dos tablas de comparación para que sea más fácil visualizar los resultados obtenidos con los dos métodos. “`data.frame()`” crea una tabla en R con columnas a las que se les ha llamado “Metodo”, “Media”, “Varianza” y “Tiempo_Ejecucion_s”.

Hay dos filas, una para cada método, que guardan los estadísticos que se han calculado. “`mean_P_N2_M1_MC`” y “`var_P_N2_M1_MC`” son la media y la varianza obtenidas con Monte Carlo, y “`mean_N2_M1_taylor`” y “`var_N2_M1_taylor`” son las obtenidas con Taylor; ambas para la probabilidad $P(N2|M1)$.

“`as.numeric(time_mc)`” y “`as.numeric(time_taylor)`” convierten a valores numéricos los valores que se han obtenido en cada caso para el tiempo de ejecución. Los tiempos se miden en segundos.

Se crean dos tablas, una para la variable $P(N2|M1)$ y otra para $P(N1|M2)$.

```
resultados_N2_M1_comp = data.frame(
  Metodo = c("Monte Carlo", "Taylor"),
  Media = c(mean_P_N2_M1_MC, mean_N2_M1_taylor),
  Varianza = c(var_P_N2_M1_MC, var_N2_M1_taylor),
  Tiempo_Ejecucion_s =
    c(as.numeric(time_mc), as.numeric(time_taylor))
)

resultados_N1_M2_comp = data.frame(
  Metodo = c("Monte Carlo", "Taylor"),
  Media = c(mean_P_N1_M2_MC, mean_N1_M2_taylor),
  Varianza = c(var_P_N1_M2_MC, var_N1_M2_taylor),
  Tiempo_Ejecucion_s =
    c(as.numeric(time_mc), as.numeric(time_taylor))
)
```

Para mostrar las tablas de datos que se acaban de crear, se emplea “`print()`”. La función “`cat()`” imprime un texto como encabezado para que se puedan distinguir ambas tablas.

```
cat("\n COMPARACIÓN PARA P(N2|M1) \n")
print(resultados_N2_M1_comp)
cat("\n COMPARACIÓN PARA P(N1|M2) \n")
print(resultados_N1_M2_comp)
```

Para completar la comparación visualmente, se crean los histogramas que representan el método de Monte Carlo con la curva normal de Taylor superpuesta.

El histograma con los valores simulados para $P(N2|M1)$ se construye con “`hist()`”. “`breaks = 50`” divide los datos en 50 intervalos o barras. “`freq = FALSE`” hace que el eje Y muestre la densidad y no la frecuencia absoluta para poder superponer una curva de densidad con la misma escala.

“`x_curve = seq(min(P_N2_M1_MC), max(P_N2_M1_MC), length.out = 100)`” crea una secuencia de 100 puntos equiespaciados que empieza en el valor mínimo obtenido por Monte Carlo y acaba en el máximo. Esto sirve de dominio para evaluar la densidad normal. Por otro lado, “`taylor_sd = sqrt(var_N2_M1_taylor)`” calcula la desviación estándar que se asocia con la varianza que ha aproximado Taylor.

“`y_curve = dnorm(x_curve, mean = mean_N2_M1_taylor, sd = taylor_sd)`” evalúa la densidad de la normal $N(\mu_{Taylor}, \sigma_{Taylor})$ en los puntos “`x_curve`”. Además, “`dnorm(x, mean, sd)`” devuelve la densidad normal:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

“`lines(x_curve, y_curve, ...)`” dibuja la curva sobre el histograma y, por último, “`legend()`” añade la leyenda en la esquina superior derecha.

El bloque análogo para $P(N1|M2)$ hace exactamente lo mismo, pero aplicándolo a esta otra probabilidad de interés.

```
hist(P_N2_M1_MC, breaks = 50,
     col = "paleturquoise2", border = "grey",
     main = "Comparación para P(N2|M1)",
     xlab = "P(N2|M1)", ylab = "Densidad", freq = FALSE)
x_curve = seq(min(P_N2_M1_MC), max(P_N2_M1_MC), length.out = 100)
taylor_sd = sqrt(var_N2_M1_taylor)
y_curve = dnorm(x_curve,
               mean = mean_N2_M1_taylor, sd = taylor_sd)
lines(x_curve, y_curve, col = "plum2", lwd = 2)
```

4. Experimentos computacionales en R

```
legend("topright", legend = c("Monte Carlo", "Aprox. Taylor"),
      fill = c("paleturquoise2", "plum2"))

hist(P_N1_M2_MC, breaks = 50,
     col = "darkseagreen2", border = "grey",
     main = "Comparación para P(N1|M2)",
     xlab = "P(N1|M2)", ylab = "Densidad", freq = FALSE)

x_curve_2 = seq(min(P_N1_M2_MC), max(P_N1_M2_MC), length.out = 100)
taylor_sd_2 = sqrt(var_N1_M2_taylor)
y_curve_2 = dnorm(x_curve_2,
                  mean = mean_N1_M2_taylor, sd = taylor_sd_2)

lines(x_curve_2, y_curve_2, col = "coral1", lwd = 2)

legend("topright", legend = c("Monte Carlo", "Aprox. Taylor"),
      fill = c("darkseagreen2", "coral1"))
```

Si la curva normal de Taylor se ajusta bien al histograma, significa que la aproximación lineal (Taylor) puede capturar la forma central de la distribución obtenida por Monte Carlo. Si se visualizase alguna asimetría o similares, podría significar que la aproximación de primer orden de Taylor no es suficiente.

4. *Diagnosis de la simulación de Monte Carlo*

La simulación de Monte Carlo es la que se toma como referencia, por lo que se le debe realizar una diagnosis.

La diagnosis del Experimento 3 sigue los mismos pasos que la del Experimento 1.

Para estudiar la normalidad, se crean gráficos Q-Q usando “`qqnorm()`” y “`qqline()`”. Si los puntos aparecen alineados, puede decirse que la distribución es aproximadamente normal. Este proceso se realiza para $P(N2|M1)$ y $P(N1|M2)$.

Previo a los gráficos, se configura la ventana gráfica con la función “`par()`” para mostrar los cuatro gráficos de la diagnosis simultáneamente.

```
par(mfrow = c(2, 2))

qqnorm(P_N2_M1_MC, main = "Gráfico Q-Q para P(N2|M1) (MC)", pch = 20)
qqline(P_N2_M1_MC, col = "paleturquoise3", lwd = 2)

qqnorm(P_N1_M2_MC, main = "Gráfico Q-Q para P(N1|M2) (MC)", pch = 20)
qqline(P_N1_M2_MC, col = "darkseagreen1", lwd = 2)
```

Para demostrar la hipótesis de convergencia se divide la suma acumulada (`cumsum()`) entre el número de simulaciones (`seq_along()`) y se obtiene la media acumulada. Seguidamente, se representa esta media acumulada en función del número de simulaciones.

Al terminar esta parte del código, se restablece la ventana gráfica a un único gráfico haciendo nuevamente uso de la función “`par()`”.

```
media_acum_N2_M1_MC = cumsum(P_N2_M1_MC)/seq_along(P_N2_M1_MC)
plot(media_acum_N2_M1_MC,
      type = "l", col = "paleturquoise3", lwd = 2,
      main = "Análisis Convergencia P(N2|M1) (MC)",
      xlab = "Número de Simulaciones", ylab = "Media Acumulada")

media_acum_N1_M2_MC = cumsum(P_N1_M2_MC) / seq_along(P_N1_M2_MC)
plot(media_acum_N1_M2_MC,
      type = "l", col = "darkseagreen1", lwd = 2,
      main = "Análisis convergencia P(N1|M2) (MC)",
      xlab = "Número de Simulaciones", ylab = "Media Acumulada")
par(mfrow = c(1, 1))
```

La hipótesis de independencia se prueba de nuevo con la función ACF usando “`acf()`”.

```
par(mfrow = c(1, 2))
acf(P_N2_M1_MC, main = "Independencia P(N2|M1) (MC)")
acf(P_N1_M2_MC, main = "Independencia P(N1|M2) (MC)")
par(mfrow = c(1, 1))
```

Asimismo, se realiza con un test de Durbin-Watson con la función “`dwtest()`”. Previo a esto, se debe cargar la librería “`library()`” si aún no se había hecho.

```
library(lmtest)
dw_test_N2_M1_MC = dwtest(P_N2_M1_MC ~ 1)
print(dw_test_N2_M1_MC)
dw_test_N1_M2_MC = dwtest(P_N1_M2_MC ~ 1)
print(dw_test_N1_M2_MC)
```

4. Experimentos computacionales en R

En el siguiente capítulo se comentarán los resultados obtenidos de cada método y de la diagnosis realizada a Monte Carlo como método de referencia.

5. Resultados y discusión

5.1. Experimento 1

El objetivo del primer experimento era estudiar cómo la variabilidad en las probabilidades a priori de estimulación nerviosa se propaga hacia las probabilidades $P(N2|M1)$ y $P(N1|M2)$, utilizando la simulación de Monte Carlo.

En la tabla siguiente se recogen los estadísticos descriptivos resultantes del Experimento 1.

| Variable respuesta | $P(N2 M1)$ | $P(N1 M2)$ |
|--|------------|------------|
| Media | 0.3193 | 0.4853 |
| Varianza | 0.0640 | 0.0823 |
| Mínimo | 0.0000 | 0.0001 |
| Cuartil 25% | 0.0992 | 0.2354 |
| Mediana (Cuartil 50%) | 0.2661 | 0.4747 |
| Cuartil 75% | 0.5078 | 0.7295 |
| Máximo | 0.9960 | 0.9999 |
| Correlación entre $P(N2 M1)$ y $P(N1 M2)$ | -0.6911 | |

Tabla 1. Resultados Experimento 1

- *Distribución de $P(N2|M1)$.*

La Figura 2, representa el histograma para la probabilidad de interés $P(N2|M1)$, que se interpretará a continuación.

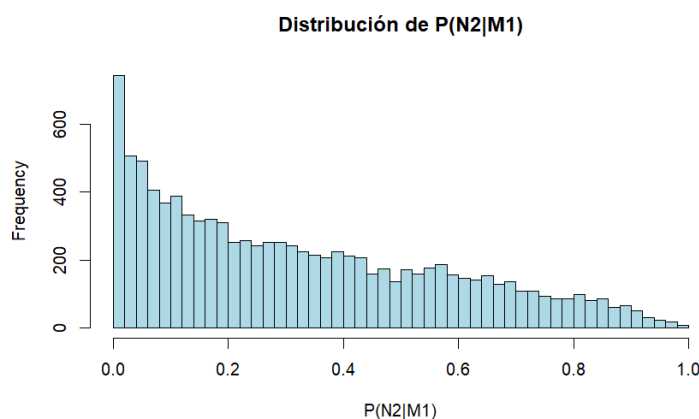


Figura 2. Histograma de la distribución de probabilidad para $P(N2|M1)$ (Exp. 1)

Los resultados muestran que la distribución de $P(N2|M1)$ presenta un comportamiento asimétrico (más concretamente, asimetría positiva) y no uniforme, con sesgo hacia los valores bajos. La media obtenida es aproximadamente 0.3193, y la mediana tiene un valor inferior (~ 0.2661), lo que confirma el sesgo. Esto indica que los valores por debajo de 0.3193 son mucho más probables que los valores que están por encima.

5. Resultados y discusión

Además, los valores cubren todo el rango $[0, 1]$, aunque la mayoría de las simulaciones están por debajo de 0.5, como puede verse en el histograma.

La combinación del Teorema de Bayes con la estructura de las probabilidades condicionadas son la consecuencia directa de los resultados que se han obtenido. Para que $P(N2|M1)$ tenga un valor alto, tiene que ocurrir que $P(N2)$ sea alta y que la probabilidad $P(M1)$ no aumente considerablemente por aportaciones de los nervios $N1$ y $N3$.

El valor de $P(M1|N2)$ es 0.5, no mucho mayor que el 0.4 de $P(M1|N1)$. $P(M1|N3)$ sí tiene un valor bajo (0.2) en comparación a estos. Como hay tan poca diferencia entre las probabilidades de que se contraiga $M1$ si se estimulan los nervios $N1$ o $N2$, va a ser normal que $P(N1)$ sea lo suficientemente grande como para que la contribución de $N1$ sea la que domina el resultado, provocando que $P(N2|M1)$ tenga un valor pequeño. Pero es que, además, las probabilidades a priori se han generado de forma aleatoria con la condición de que las tres deben sumar 1. Por eso $P(N2|M1)$ tiende a recibir valores pequeños en la mayoría de las simulaciones.

Se puede afirmar que las variaciones en las probabilidades a priori hacen que la probabilidad $P(N2|M1)$ pueda tomar valores que van desde cercanos a cero hasta casi 1.

- *Distribución de $P(N1|M2)$.*

La Figura 3 muestra el histograma para la segunda probabilidad de interés $P(N1|M2)$.

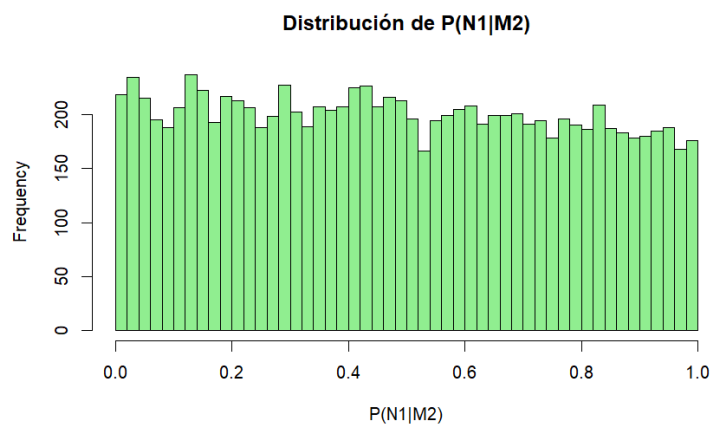


Figura 3. Histograma de la distribución de probabilidad para $P(N1|M2)$ (Exp. 1)

Al contrario que $P(N2|M1)$, $P(N1|M2)$ muestra una distribución más plana y uniforme. La media obtenida es aproximadamente 0.4853 y la mediana, en este caso, está muy próxima a este valor (~ 0.4747). Esto implica que el sesgo es menor (prácticamente inexistente).

En este caso, para el $M2$, la diferencia entre $N1$ y $N2$ no es tan alta (0.6 frente a 0.5), pero $N3$ es el más relevante de los tres (probabilidad de 0.8): aunque $N1$ sea más “fuerte” que $N2$, los valores altos de $P(N1|M2)$ se van a ver limitados cuando $N3$ tenga una probabilidad alta inicial.

Que el histograma sea más uniforme que el del caso anterior indica que la influencia de la probabilidad inicial y la condicionada del nervio N1 está más equilibrada frente a la competencia que le hace el nervio N3. Esto indica que la probabilidad $P(N1|M2)$ puede ser muy baja, muy alta o cualquier valor intermedio con una frecuencia homogénea.

Como consecuencia de esto, la incertidumbre inicial se traduce en una gran dispersión posterior, lo que da a entender que el modelo es muy sensible cuando no se toman probabilidades iniciales sin variabilidad.

- *Relación entre ambas probabilidades.*

La correlación entre $P(N2|M1)$ y $P(N1|M2)$ es -0.6911 ; es decir, fuertemente negativa. Esto significa que, cuando la probabilidad $P(N2|M1)$ es alta, $P(N1|M2)$ tiende a ser baja y viceversa. Este resultado es coherente con la condición que se les ha hecho cumplir a las probabilidades iniciales de que su suma tenía que ser igual a 1. Cuando $P(N2)$ reciba un valor alto, $P(N1)$ tiene que tomar necesariamente un valor más pequeño, y al contrario. Esta relación se traslada también a las variables de salida, porque tanto $P(N2|M1)$ como $P(N1|M2)$ dependen de las probabilidades a priori.

El gráfico de dispersión (Figura 4) confirma esta relación inversa. La densidad es alta en las esquinas superior izquierda e inferior derecha: la mayoría de los 10.000 puntos de la simulación de Monte Carlo se acumulan en esas zonas y son, por tanto, las combinaciones más probables (especialmente las que se dan cuando $P(N2|M1)$ toma valores pequeños y $P(N1|M2)$, grandes).

Por otro lado, la densidad es inexistente cuando los valores son altos tanto para $P(N2|M1)$ como para $P(N1|M2)$: es muy poco frecuente que una simulación genere un resultado donde ambas variables respuesta sean altas a la vez.

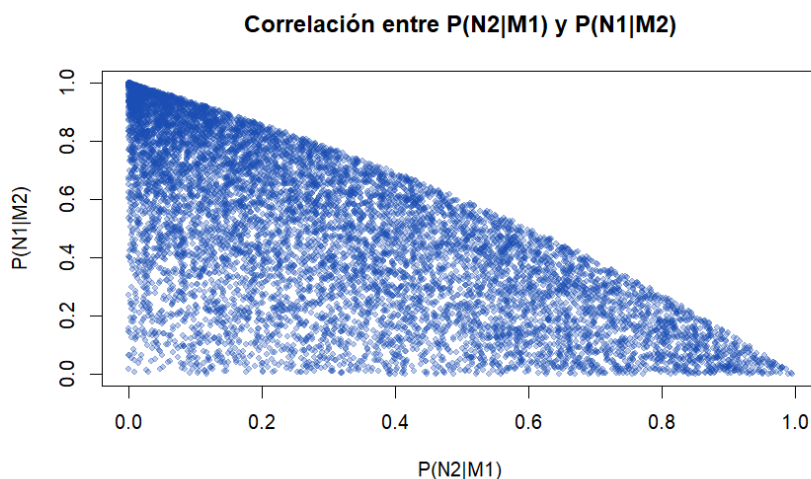


Figura 4. Gráfico de dispersión de las dos variables respuesta (Exp. 1)

Se puede ver así cómo la variabilidad que comparten las probabilidades iniciales induce dependencias entre las probabilidades de interés, incluso aunque se refieran a eventos distintos (activación de los músculos M1 y M2).

Para terminar, se van a comentar los valores del coeficiente de variación para $P(N2|M1)$ y $P(N1|M2)$.

Si el CV es menor al 15%, se puede decir que la dispersión es baja. Si está entre el 15 y el 30% es moderada y, si es mayor del 30%, se considera alta. Los valores que se han obtenido son del 79.2% para $P(N2|M1)$ y del 59.1% para $P(N1|M2)$. Es decir, son extremadamente altos.

Esta incertidumbre tan grande se refleja también en los intervalos de confianza, muy amplios en los dos casos. Para $P(N2|M1)$ hay un 95% de confianza de que el valor real esté en el intervalo [0.005, 0.864]. Para $P(N1|M2)$, el valor real está en [0.023, 0.972] con un 95% de confianza.

Esto último se muestra en la siguiente tabla:

| | Coeficiente de variación (CV) | Límite inferior IC (95%) | Límite superior IC (95%) |
|-----------------|--------------------------------------|---------------------------------|---------------------------------|
| P(N2 M1) | 79.2102 % | 0.0053 | 0.8637 |
| P(N1 M2) | 59.1017 % | 0.0227 | 0.9716 |

Tabla 2. CV e Intervalos de Confianza (Exp. 1)

Con este experimento se ha podido demostrar que el método de Monte Carlo es eficaz para estudiar la propagación de incertidumbre en modelos que usan el Teorema de Bayes. Se ha podido observar cómo la variabilidad de las probabilidades a priori transmite y genera una incertidumbre muy elevada en las probabilidades de salida, con rangos que prácticamente abarcan el intervalo de 0 a 1 y un CV que llega hasta el 79.2% para $P(N2|M1)$.

Además, produce distribuciones con formas que no son intuitivas y que reflejan el efecto no lineal del Teorema de Bayes: variaciones pequeñas en las variables de entrada pueden producir cambios grandes en las salidas, especialmente si el denominador (que, en este caso, es la probabilidad total) está influenciado por varias fuentes.

Por otro lado, la dispersión de ambas distribuciones y la fuerte correlación negativa hacen ver que el modelo es muy dependiente de los valores que toman las probabilidades iniciales.

El Experimento 1 demuestra que, si las probabilidades iniciales se desconocen, las conclusiones sobre qué nervio es el que se ha estimulado en función de las respuestas musculares, pueden variar mucho.

5.1.1. Diagnóstico de la simulación de Monte Carlo

Los gráficos Q-Q para las variables $P(N2|M1)$ y $P(N1|M2)$, recogidos en la Figura 5, muestran una desviación clara respecto a la recta teórica de normalidad, sobre todo en las colas. Por lo tanto, podría decirse que ninguna de las dos distribuciones sigue una distribución normal.

Monte Carlo no requiere que se cumpla la hipótesis de normalidad. De hecho, que las distribuciones no sean normales era algo que se podía esperar. $P(N2|M1)$ y $P(N1|M2)$ se calculan con el Teorema de Bayes, que introduce una relación no lineal entre las probabilidades a priori y las condicionadas. $P(N1)$, $P(N2)$ y $P(N3)$, además, se han generado de forma aleatoria, pero con la restricción de que las tres deben sumar como mucho 1, lo que crea una dependencia entre ellas.

Que las probabilidades a posteriori no cumplan la hipótesis de normalidad, lejos de ser un problema, lo que demuestra es que ha sido una decisión correcta usar el método de Monte Carlo para el experimento. Hay otros métodos que son más simples que Monte Carlo pero asumen que los datos son normales: si se hubiesen usado, se habrían obtenido conclusiones incorrectas.

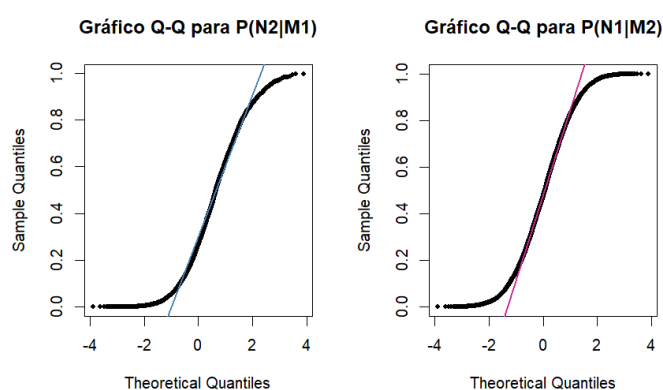


Figura 5. Normalidad (Exp.1)

Los gráficos de la Figura 6 muestran que, tanto para $P(N2|M1)$ como para $P(N1|M2)$, la media acumulada se estabiliza. Además, lo hace relativamente rápido, alrededor de las primeras simulaciones. Las demás variaciones que se observan hasta las 10.000 simulaciones son mínimas y la media acumulada se mantiene prácticamente plana. Significa que el número de simulaciones que se ha elegido es más que suficiente para que se cumpla la hipótesis de convergencia.

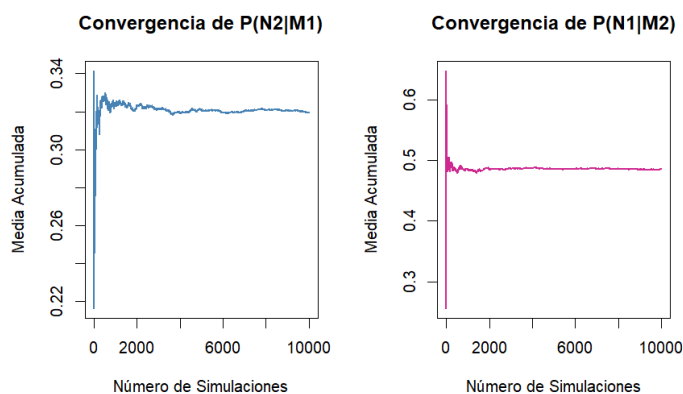


Figura 6. Convergencia (Exp. 1)

La función de autocorrelación (ACF) de la Figura 7 muestra que solo hay un pico significativo en el lag 0, pero que en el resto de lags las autocorrelaciones se mantienen próximas al cero. Las distribuciones son independientes, resultado que era previsible teniendo en cuenta que se han utilizado números aleatorios para este experimento.

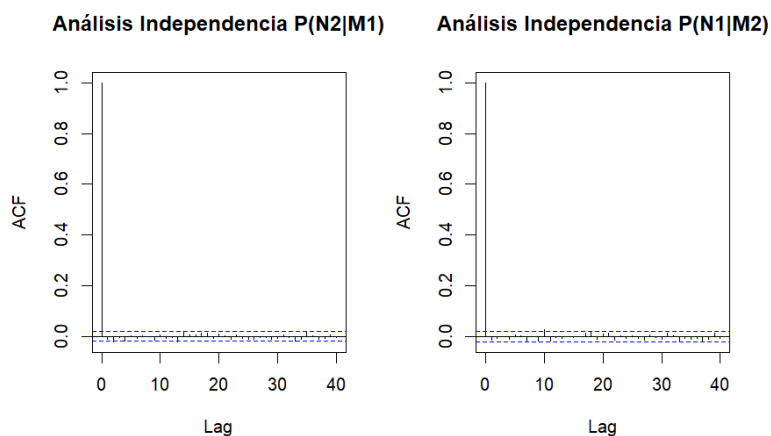


Figura 7. Independencia (Exp. 1)

El test de Durbin-Watson lo refuerza porque sus estadísticos (2.0206 para P(N2|M1) y 2.0249 para P(N1|M2)) son prácticamente iguales a 2. Sus p-valores son altos, superiores a 0.84. Esto indica que hay ausencia de autocorrelación relevante entre las variables de salida.

Los resultados del test se muestran en la siguiente tabla:

| Variable respuesta | P(N2 M1) | P(N1 M2) |
|-----------------------|----------|----------|
| Estadístico DW | 2.0206 | 2.0249 |
| p-valor | 0.8482 | 0.8938 |

Tabla 3. Test de Durbin-Watson (Exp. 1)

Es importante que se haya podido demostrar la hipótesis de independencia, pues el método de Monte Carlo se basa precisamente en esta suposición. Si hubiese dependencia, los resultados no serían fiables y no podría usarse este método.

5.2. Experimento 2

Para el análisis de sensibilidad del Experimento 2, como se explicó en la sección 4.1.2, se desarrolló un ANOVA para P(N2|M1) y P(N1|M2). Es un modelo factorial 3^3 sin repeticiones, con las tres probabilidades condicionadas (P(M1|N1), P(M1|N2) y P(M1|N3)) como factores de tres niveles cada uno. No se hacen repeticiones porque se incluye la incertidumbre de las probabilidades P(Ni) dentro de cada uno de los 27 escenarios usando una simulación de Monte Carlo.

Para el error residual, se asume que la interacción de tercer orden es despreciable. El modelo cuenta con 26 grados de libertad, de los cuales 6 se destinan a los efectos principales, 12 a cada

interacción de segundo orden y los 8 restantes, que son los que le corresponderían a la interacción de tercer orden, son los que se le asignan al error residual.

En la tabla 4 se presentan los valores obtenidos para cada uno de los 27 escenarios del Experimento 2. Estos datos se analizarán a continuación mediante las tablas ANOVA.

| | F_PM1N1 | F_PM1N2 | F_PM1N3 | P_N2_M1_mean | P_N1_M2_mean |
|-----------|----------------|----------------|----------------|---------------------|---------------------|
| 1 | 0.36 | 0.45 | 0.18 | 0.3193484 | 0.4869648 |
| 2 | 0.4 | 0.45 | 0.18 | 0.3044700 | 0.4792059 |
| 3 | 0.44 | 0.45 | 0.18 | 0.2965541 | 0.4678111 |
| 4 | 0.36 | 0.5 | 0.18 | 0.3346350 | 0.4953512 |
| 5 | 0.4 | 0.5 | 0.18 | 0.3242312 | 0.4891846 |
| 6 | 0.44 | 0.5 | 0.18 | 0.3134548 | 0.4748322 |
| 7 | 0.36 | 0.55 | 0.18 | 0.3535804 | 0.5019644 |
| 8 | 0.4 | 0.55 | 0.18 | 0.3371641 | 0.4936522 |
| 9 | 0.44 | 0.55 | 0.18 | 0.3246768 | 0.4840295 |
| 10 | 0.36 | 0.45 | 0.2 | 0.3121184 | 0.4896568 |
| 11 | 0.4 | 0.45 | 0.2 | 0.3032490 | 0.4836538 |
| 12 | 0.44 | 0.45 | 0.2 | 0.2930200 | 0.4707729 |
| 13 | 0.36 | 0.5 | 0.2 | 0.3240682 | 0.5024586 |
| 14 | 0.4 | 0.5 | 0.2 | 0.3195567 | 0.4855333 |
| 15 | 0.44 | 0.5 | 0.2 | 0.3082742 | 0.4745954 |
| 16 | 0.36 | 0.55 | 0.2 | 0.3450953 | 0.5072418 |
| 17 | 0.4 | 0.55 | 0.2 | 0.3357812 | 0.4947452 |
| 18 | 0.44 | 0.55 | 0.2 | 0.3210823 | 0.4855441 |
| 19 | 0.36 | 0.45 | 0.22 | 0.3102195 | 0.4946023 |
| 20 | 0.4 | 0.45 | 0.22 | 0.2991852 | 0.4809932 |
| 21 | 0.44 | 0.45 | 0.22 | 0.2877598 | 0.4733879 |
| 22 | 0.36 | 0.5 | 0.22 | 0.3218630 | 0.5045908 |
| 23 | 0.4 | 0.5 | 0.22 | 0.3123747 | 0.4909611 |
| 24 | 0.44 | 0.5 | 0.22 | 0.3018234 | 0.4814667 |
| 25 | 0.36 | 0.55 | 0.22 | 0.3384943 | 0.5050713 |
| 26 | 0.4 | 0.55 | 0.22 | 0.3288031 | 0.4960149 |
| 27 | 0.44 | 0.55 | 0.22 | 0.3177484 | 0.4860606 |

Tabla 4. Valores obtenidos de los 27 escenarios del ANOVA (Exp. 2)

La Figura 8 recoge el contenido de la tabla ANOVA para la variable de salida $P(N2|M1)$.

En esta primera tabla se observa que $P(M1|N2)$ es el factor que más impacto tiene ($F \sim 1194.93$ y $p \sim 1.24e-10$), seguido de $P(M1|N1)$ ($F \sim 594.51$, $p \sim 1.995e-09$) y, en menor medida, de $P(M1|N3)$ ($F \sim 126.14$, $p \sim 8.92e-07$). Estos resultados son coherentes porque en el cálculo de $P(N2|M1)$, $P(M1|N2)$ interviene directamente en el numerador y su efecto por tanto es mayor. Las otras dos probabilidades condicionadas intervienen indirectamente con $P(M1)$ y esto hace que su efecto, aunque sigue siendo grande, sea un poco menor.

5. Resultados y discusión

Si se observan las interacciones entre los factores se puede ver que para $P(N2|M1)$ solo son significativas las que se dan entre $P(M1|N1)$ y $P(M1|N2)$ ($p \sim 0.046$), y entre $P(M1|N1)$ y $P(M1|N3)$ ($p \sim 0.031$), especialmente esta segunda. Sin embargo, ninguna de las dos tiene un efecto especialmente grande porque sus p-valores se acercan a 0.05.

```
--- Tabla ANOVA para P(N2|M1) ---
      Df      Sum Sq      Mean Sq      F value      Pr(>F)
F_PM1N1      2  0.0021137  0.00105687    594.5057  1.995e-09 ***
F_PM1N2      2  0.0042485  0.00212426   1194.9265  1.239e-10 ***
F_PM1N3      2  0.0004485  0.00022425    126.1446  8.923e-07 ***
F_PM1N1:F_PM1N2  4  0.0000283  0.00000707     3.9759  0.04593 *
F_PM1N1:F_PM1N3  4  0.0000332  0.00000829     4.6629  0.03084 *
F_PM1N2:F_PM1N3  4  0.0000160  0.00000401     2.2552  0.15227
Residuals      8  0.0000142  0.00000178
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figura 8. Tabla ANOVA para $P(N2|M1)$ (Exp. 2)

Por otro lado, la Figura 9, muestra la tabla ANOVA para la variable de salida $P(N1|M2)$.

En el caso de la probabilidad $P(N1|M2)$ el factor más relevante es $P(M1|N1)$ (con $F \sim 251.04$, $p \sim 6.05e-08$), seguido de $P(M1|N2)$ ($F \sim 114.00$, $p \sim 1.32e-06$) y de $P(M1|N3)$, que, aunque influye, tiene un efecto algo menor ($F \sim 11.29$, $p \sim 0.0047$). Este orden vuelve a tener sentido porque $P(M2|N1) = 1 - P(M1|N1)$. $P(M1|N1)$ interviene directamente en el numerador al calcular $P(N1|M2)$ y le da cierta ventaja frente a los demás factores.

Para $P(N1|M2)$ las interacciones entre los factores no son relevantes en ningún caso al tener todos los p-valores valores altos cercanos a 0.3. Es decir, $P(N1|M2)$ depende de los tres factores principales, pero estos actúan de forma independiente.

```
--- Tabla ANOVA para P(N1|M2) ---
      Df      Sum Sq      Mean Sq      F value      Pr(>F)
F_PM1N1      2  0.00199298  0.00099649   251.0437  6.05e-08 ***
F_PM1N2      2  0.00090503  0.00045252   114.0013  1.32e-06 ***
F_PM1N3      2  0.00008966  0.00004483    11.2945  0.004678 **
F_PM1N1:F_PM1N2  4  0.00001892  0.00000473     1.1915  0.384514
F_PM1N1:F_PM1N3  4  0.00002440  0.00000610     1.5370  0.279921
F_PM1N2:F_PM1N3  4  0.00002143  0.00000536     1.3497  0.331898
Residuals      8  0.00003176  0.00000397
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figura 9. Tabla ANOVA para $P(N1|M2)$ (Exp. 2)

Para terminar, la Figura 10 muestra el gráfico de dispersión entre $P(N2|M1)$ y $P(N1|M2)$. Puede verse que la correlación entre las dos probabilidades es positiva porque los puntos suben de izquierda a derecha. Los escenarios que tienden a asignarle un valor medio más alto a $P(N2|M1)$, también tienden a dárselo a $P(N1|M2)$. El valor +0.8157 que se ha obtenido confirma lo que muestra el gráfico.

Que la correlación sea positiva puede ser por los efectos combinados de los factores. Por ejemplo, $P(M1|N1)$ tenía un efecto negativo en $P(N2|M1)$ y en $P(N1|M2)$ así que, en los escenarios con un valor bajo de $P(M1|N1)$, se va a tender a “empujar” hacia arriba el valor de las dos probabilidades. Es cierto que hay otros factores que se comportan diferente para

$P(N2|M1)$ que para $P(N1|M2)$, pero el efecto que comparten con $P(M1|N1)$ es lo suficientemente fuerte como para crear esa tendencia positiva general.

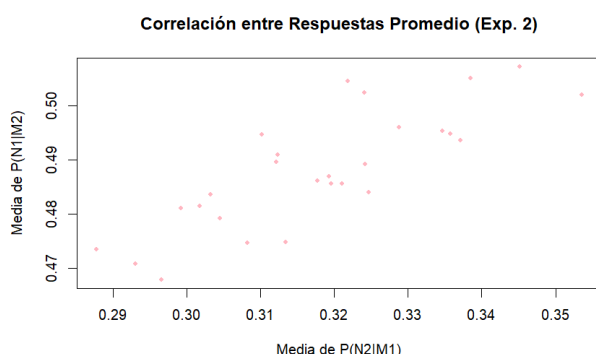


Figura 10. Gráfico de dispersión entre $P(N2|M1)$ y $P(N1|M2)$ (Exp. 2)

En cuanto a los gráficos de efectos principales, se confirma que, tanto para $P(N2|M1)$ como $P(N1|M2)$, hay factores significativos, como han indicado los ANOVAs. Esto se puede observar porque los intervalos de confianza, que se representan sobre cada punto con las barras de error verdes, prácticamente no se solapan entre sí.

Para $P(N2|M1)$ (Figura 11) el factor $P(M1|N2)$ es el más influyente por ser el que tiene la pendiente más pronunciada. Lo siguen $P(M1|N1)$ y $P(M1|N3)$, aunque con menos impacto. $P(M1|N1)$ y $P(M1|N3)$ tienen una pendiente negativa y la de $P(M1|N2)$ es positiva. Al aumentar los niveles de los dos primeros, el promedio disminuirá; pero si aumenta el nivel de $P(M1|N2)$, el resultado promedio aumentará también.

También se corroboran los valores obtenidos en la tabla ANOVA para $P(N1|M2)$ (Figura 12). El factor que más afecta es $P(M1|N1)$ con una pendiente negativa grande; seguido de $P(M1|N2)$ con una pendiente positiva considerable pero menor; y, por último, de $P(M1|N3)$ con una pendiente negativa casi plana.

En el Anexo 5 se han añadido las tablas con los valores numéricos que han generado “ICplot()” e “interIC()”. Estas tablas miden la magnitud de los efectos principales que se acaban de comentar. Por ejemplo, para $P(N2|M1)$, el factor F_M1N2 produce un aumento en la media de 0.03 cuando pasa del nivel bajo (0.30) al alto (0.33). También se muestra que las estimaciones son muy exactas, porque los intervalos de confianza que se obtienen son muy estrechos.

Sobre los intervalos de confianza, son más compactos para $P(N2|M1)$ que para $P(N1|M2)$. Tiene sentido porque el ANOVA de la primera variable mostró que el error residual para esta (0.00000178) era menor que para $P(N1|M2)$ (0.00000397). Todo esto se traduce en que es más precisa la explicación que da el diseño factorial de la probabilidad $P(N2|M1)$.

5. Resultados y discusión

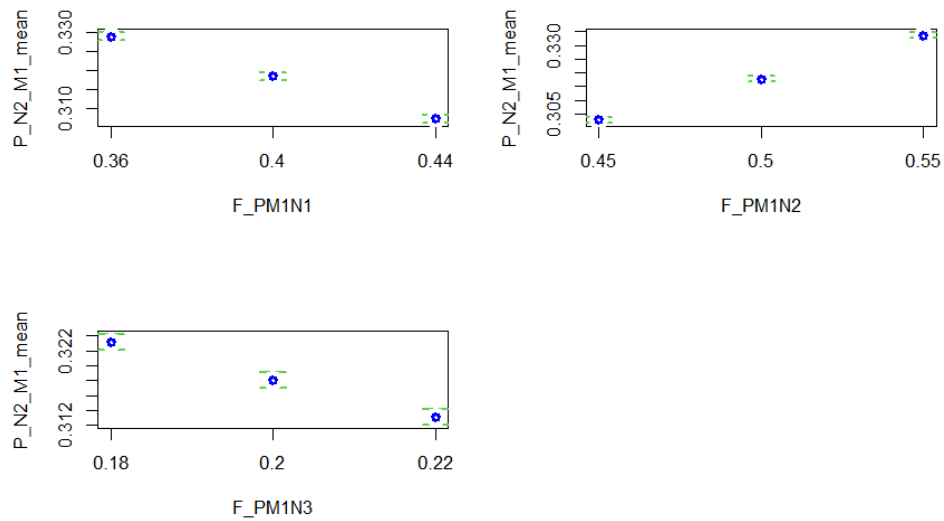


Figura 11. Gráficos de efectos principales para P(N2|M1) (Exp. 2)

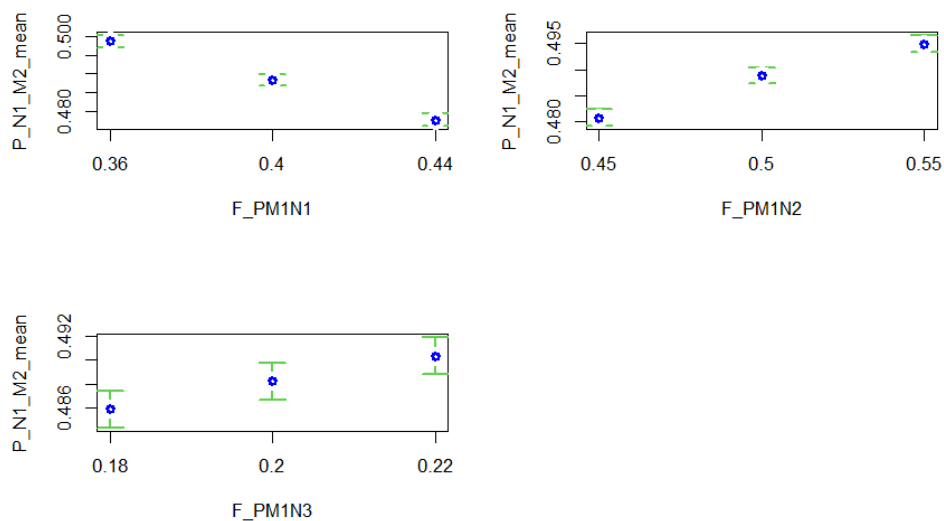


Figura 12. Gráficos de efectos principales para P(N1|M2) (Exp. 2)

Los gráficos de interacción se muestran en la Figura 13. No se han hecho para P(N1|M2) porque ninguna interacción ha resultado ser significativa. Para P(N2|M1) se corrobora lo que se ha comentado anteriormente, ya que las líneas no son paralelas en el caso de P(M1|N1) y P(M1|N2), y la diferencia es aún mayor para P(M1|N1) y P(M1|N3). Que las líneas no sean paralelas significa que hay una interacción entre esas variables, tal y como indicaba el ANOVA. El efecto del factor P(M1|N1) sobre el resultado depende del nivel en que se encuentren P(M1|N2) y P(M1|N3). No actúan de forma puramente independiente.

En el gráfico de la izquierda, las líneas de los niveles bajo, medio y alto del factor F_PM1N2 no son paralelas, pero por muy poco. Existe una interacción real entre los factores F_PM1N1

y F_{PM1N2} , aunque es muy sutil. Cabía esperar este resultado por lo que mostró la tabla ANOVA para esta combinación de factores: el p-valor es 0.046, que se encuentra muy próximo del 0.05 (umbral a partir del cual ya no se consideraría significativa).

Si se analiza el gráfico de la derecha, se puede ver que la línea roja ($P(M1|N3)$ bajo) tiene una pendiente mucho más pronunciada que la de la línea azul ($P(M1|N3)$ alto). Esto es lo mismo que decir que el efecto negativo que tiene $P(M1|N1)$ es más fuerte cuando $P(M1|N3)$ está en su nivel bajo.

Los intervalos de confianza para las interacciones entre los factores se incluyen igualmente en el Anexo 5. Puede observarse que, en este caso, las estimaciones vuelven a ser muy exactas porque los intervalos que se han obtenido son muy compactos.

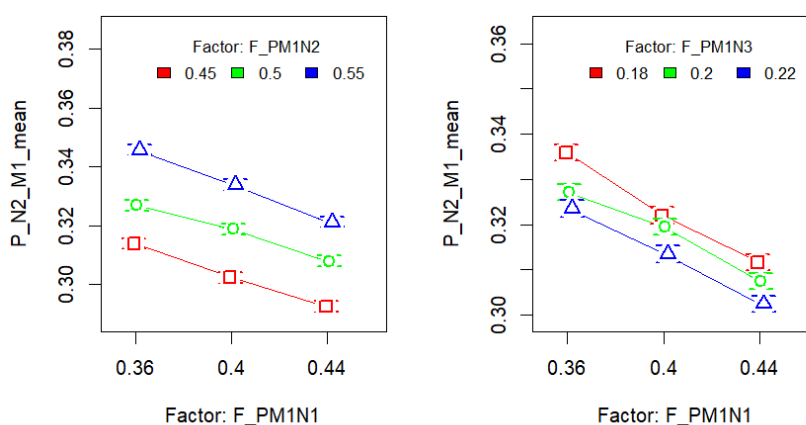


Figura 13. Gráficos de interacción para $P(N2|M1)$ (Exp. 2)

El Experimento 2 ha demostrado que los parámetros que están directamente relacionados con el nervio de estudio son los que dominan la incertidumbre del resultado. $P(N2|M1)$ es muy sensible a cambios que puedan darse en $P(M1|N2)$, e igual le ocurre a $P(N1|M2)$ con $P(M1|N1)$. Que $P(M1|N3)$ tenga un efecto menor en los dos casos era previsible porque $N3$ es un nervio distinto al que nos interesa e influye en las probabilidades a posteriori indirectamente.

Además, se ha obtenido que, mientras que la predicción de $P(N1|M2)$ es sencilla y solo tiene en cuenta los efectos principales, la de $P(N2|M1)$ es no aditiva y debe tenerse en cuenta que los factores se influyen entre sí.

5.2.2. Diagnóstico del modelo ANOVA

Los gráficos Q-Q de la Figura 14 reflejan la relación que hay entre los cuantiles teóricos de una distribución normal y los que se observan en los residuos. Tanto para $P(N2|M1)$ (gráfico de la izquierda) como para $P(N1|M2)$ (gráfico de la derecha) los puntos siguen una tendencia que es aproximadamente lineal con pequeñas desviaciones en los extremos. La curvatura que hay, aunque no es muy acusada, sugiere una pequeña asimetría que puede deberse a que el tamaño

5. Resultados y discusión

de la muestra no es muy grande (27 escenarios). Aun así, las desviaciones no son lo suficientemente pronunciadas como para negar la hipótesis de normalidad.

Puede que las pequeñas desviaciones que se han mencionado se deban a que los valores medios de las probabilidades calculadas son el resultado de simulaciones muy precisas, con 10.000 iteraciones por escenario: la dispersión residual va a disminuir y los puntos se van a concentrar alrededor de la recta teórica.

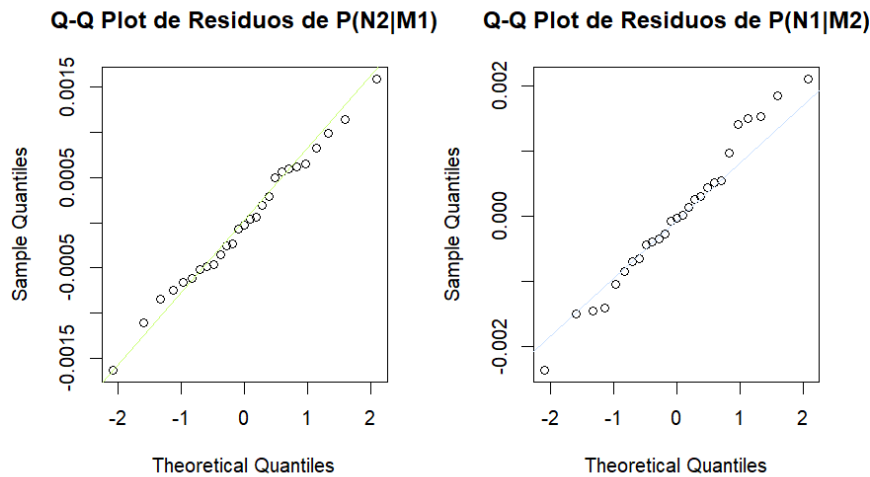


Figura 14. Normalidad de residuos (Exp. 2)

Por otro lado, los diagramas de residuos frente a los valores ajustados (Figura 15) muestran que hay una dispersión aleatoria alrededor de la línea horizontal cero, sin tendencias crecientes o decrecientes. El hecho de que no pueda observarse una forma de “embudo” (*funnel shape*) confirma que no existe heterocedasticidad. Sin embargo, para la variable P(N1|M2) (gráfico de la derecha) puede apreciarse un ligero estrechamiento alrededor del 0.47. El efecto es muy sutil y el resto de los residuos mantiene prácticamente la misma amplitud, por lo que no se considera que sea lo suficientemente grande como para invalidar homocedasticidad. Los residuos se distribuyen de forma homogénea para todo el rango de valores ajustados tanto en P(N2|M1) como en P(N1|M2), lo que verifica la hipótesis de homocedasticidad u homogeneidad de varianzas.

Este comportamiento tiene sentido, entre otras, porque las respuestas analizadas son las medias de simulaciones muy largas, lo que hace que el impacto del error aleatorio sea menor y contribuye a que la varianza se estabilice.

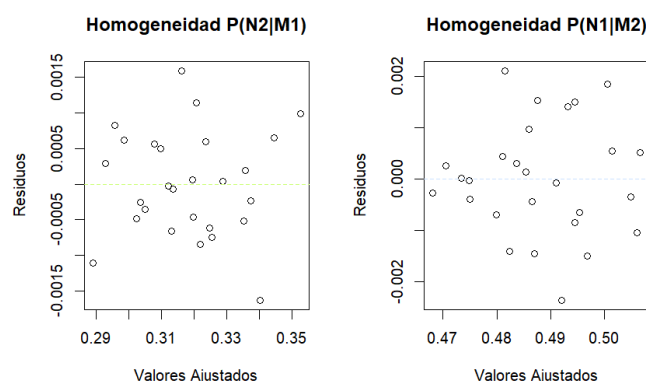


Figura 15. Homocedasticidad (Exp. 2)

Con los cuatro gráficos comentados, puede afirmarse que el modelo ANOVA cumple correctamente con las hipótesis básicas: los residuos siguen una distribución que se aproxima a la normal y tienen varianzas homogéneas, sin ninguna tendencia.

Ya se comentó en la sección 3.4.3 que los residuos deben también cumplir la hipótesis de independencia. Por la propia naturaleza del modelo y por que no se han observado resultados anómalos, se puede asumir que esta hipótesis también se cumple.

Esto da seguridad a la hora de interpretar los resultados que se han obtenido con este modelo en el Experimento 2.

5.3. Experimento 3

Para la comparación de Monte Carlo con el método de perturbación de Taylor de primer orden se ha tenido en cuenta una variación de $\pm 10\%$ respecto a los valores originales de las probabilidades condicionadas. Las probabilidades a priori sí que se han mantenido fijas con $P(N1) = 0.3$, $P(N2) = 0.2$ y $P(N3) = 0.5$.

Las tablas 5 y 6 recogen los resultados que se han obtenido para $P(N2|M1)$ y $P(N1|M2)$:

COMPARACIÓN PARA $P(N2|M1)$

| Método | Monte Carlo | Taylor |
|---------------------|-------------|-------------|
| Media | 0.3124 | 0.3125 |
| Varianza | 0.000232620 | 0.000231425 |
| Tiempo de ejecución | 0.5785 | 0.3991 |

Tabla 5. Comparación Monte Carlo vs. Taylor para $P(N2|M1)$ (Exp. 3)

COMPARACIÓN PARA $P(N1|M2)$

| Método | Monte Carlo | Taylor |
|---------------------|-------------|------------|
| Media | 0.2647 | 0.2647 |
| Varianza | 6.6202e-05 | 6.6226e-05 |
| Tiempo de ejecución | 0.5785 | 0.3991 |

Tabla 6. Comparación Monte Carlo vs. Taylor para $P(N1|M2)$ (Exp. 3)

En ambos casos, las medias que se han obtenido con Taylor son prácticamente iguales a las que se han estimado con Monte Carlo. Podría decirse que la aproximación de Taylor reproduce de manera casi exacta el valor que se espera de las probabilidades con el rango de variabilidad que hemos tomado.

Las varianzas de ambos métodos también son muy parecidas. Esto indica que, para la incertidumbre de $\pm 10\%$ (relativamente pequeña), el método de Taylor es suficiente para capturar la dispersión de las variables respuesta, lo cual podía esperarse porque las funciones que definen $P(N2|M1)$ y $P(N1|M2)$ son continuas y dependen de las probabilidades condicionadas (que son los parámetros del modelo).

Sí que hay algo más de diferencia en cuanto al tiempo de ejecución. El método de Taylor tarda menos tiempo en ejecutarse que el de Monte Carlo (0.3991 s frente a 0.5785 s). La diferencia no es demasiado importante en este experimento porque el tamaño de la simulación no es muy grande, pero es relevante a nivel conceptual. Si la simulación hubiese sido más compleja o con un número mayor de simulaciones o de parámetros, se notaría mucho más la eficiencia del método de Taylor.

En el gráfico de comparación para $P(N2|M1)$, en la Figura 16, puede observarse que la distribución obtenida con Monte Carlo es aproximadamente simétrica, centrada en torno a ~ 0.312 . La curva normal que se superpone y basada en Taylor, coincide notablemente con la forma del histograma, lo que corrobora la idea de que, para este experimento concreto, usar la aproximación de Taylor de primer orden es totalmente válido.

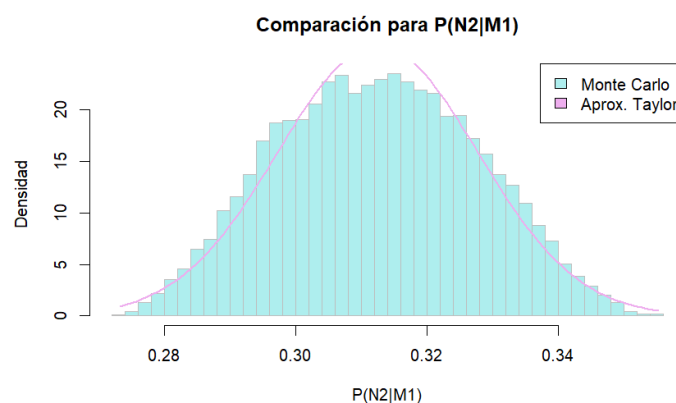


Figura 16. Gráfico Monte Carlo vs. Taylor para $P(N2|M1)$ (Exp. 3)

En el caso del gráfico de comparación para $P(N1|M2)$, mostrado en la Figura 17, se aprecia una situación similar: la curva normal de Taylor describe bien la forma de la distribución de Monte Carlo, lo que sugiere que puede usarse este método para estudiar la dispersión de los resultados para un rango de variación no muy grande y sin demasiadas dependencias lineales.

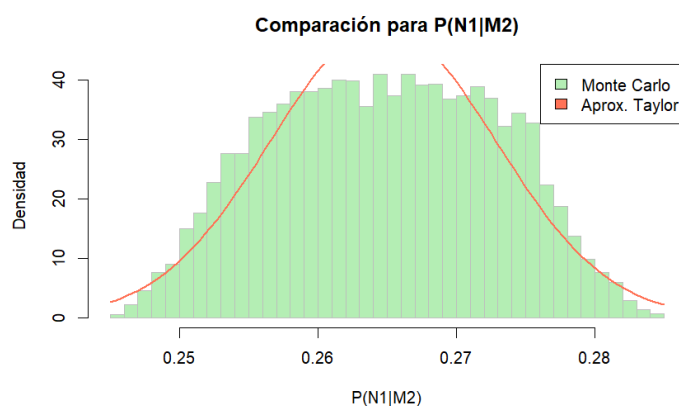


Figura 17. Gráfico Monte Carlo vs. Taylor para P(N1|M2) (Exp. 3)

En cuanto a la interpretación de los valores obtenidos en sí mismos, se va a realizar sobre los resultados según Monte Carlo por ser el método que hemos tomado como referencia (los valores son prácticamente idénticos en Taylor).

$P(N2|M1)$ presenta una media ~ 0.312 , lo que indica que, habiéndose contraído el músculo M1, la probabilidad de que se deba por la estimulación de N2 es del 31.2% aproximadamente.

Además, el valor de la varianza de $P(N2|M1)$ es ~ 0.0023 . Esto significa que, a pesar de la incertidumbre del $\pm 10\%$ en las probabilidades condicionadas iniciales, el resultado del modelo es estable. El histograma ya mencionado anteriormente para $P(N2|M1)$ confirma esta baja dispersión y muestra que los valores están concentrados alrededor del valor medio.

Por su parte, $P(N1|M2)$ tiene una media ~ 0.265 : la probabilidad de que el nervio N1 sea el responsable de que M2 se haya contraído es de un 26.5% aproximadamente.

La varianza de $P(N1|M2)$ es aún menor ($\sim 6.62e-05$), por lo que la estimación de esta probabilidad es todavía más precisa que la de $P(N2|M1)$. Dicho con otras palabras, que la varianza de $P(N2|M1)$ sea mayor que la de $P(N1|M2)$ puede significar que la primera es mucho más sensible a la incertidumbre de las probabilidades condicionadas que $P(N1|M2)$.

Los CV que se han obtenido son 4.9% para $P(N2|M1)$ y 3.1% para $P(N1|M2)$. Para un CV menor del 15% ya se considera que la dispersión es baja. Con esto puede decirse que, a pesar de la incertidumbre del $\pm 10\%$ de las probabilidades condicionadas, la variabilidad que se obtiene para los resultados es muy pequeña.

Los valores del CV son coherentes con los intervalos de confianza que se obtienen. Se tiene un 95% de confianza de que el valor real esté entre el 28.4% y el 34.1% para $P(N2|M1)$. En el caso de $P(N1|M2)$ hay un 95% de confianza de que el valor real se encuentre entre el 25% y el 27.9%. Como se puede observar, los intervalos son muy compactos. Además, las medias que se calcularon con Taylor quedan dentro de los dos intervalos.

Todo esto se muestra en la siguiente tabla:

| | Coefficiente de variación (CV) | Límite inferior IC (95%) | Límite superior IC (95%) |
|-----------------|---------------------------------------|---------------------------------|---------------------------------|
| P(N2 M1) | 4.8820 % | 0.2842 | 0.3412 |
| P(N1 M2) | 3.0737 % | 0.2501 | 0.2794 |

Tabla 7. CV e Intervalos de Confianza (Exp. 3)

Además, al ser ambas distribuciones simétricas, como se ve en los gráficos, no se observa ningún sesgo.

Recapitulando, lo principal es que con el Experimento 3 hemos demostrado que, para el rango de incertidumbre del $\pm 10\%$, el modelo se comporta de forma muy “lineal” y predecible. No hay curvaturas extrañas ni comportamientos no lineales que la aproximación de Taylor de primer orden no pueda capturar: para este problema en particular, la solución rápida (Taylor) no sacrifica prácticamente nada de precisión.

5.3.1. Diagnósis de la simulación de referencia

Al igual que se hizo en el Experimento 1, se realiza ahora la diagnósis del modelo de Monte Carlo por ser el método que se ha tomado como referencia en el Experimento 3.

Los gráficos Q-Q para P(N2|M1) y P(N1|M2), en la Figura 18, muestran una alineación de los puntos razonablemente buena sobre la recta teórica. Es cierto que en las colas (en los extremos superior e inferior) pueden verse pequeñas desviaciones, pero no es suficiente como para negar normalidad. Por la cercanía general de los puntos a la recta podría decirse que la distribución es casi normal.

En este caso, que la distribución cumpla la hipótesis de normalidad tiene sentido con que la comparación con el método de Taylor haya salido tan parecida. Cuando se usa la aproximación de Taylor de primer orden es porque el modelo tiene un comportamiento lineal. Cuando el comportamiento es lineal, tiende a producir una salida que es normal. Si Monte Carlo hubiera generado una distribución que fuese claramente asimétrica, no se podría comparar con la aproximación de Taylor de primer orden.

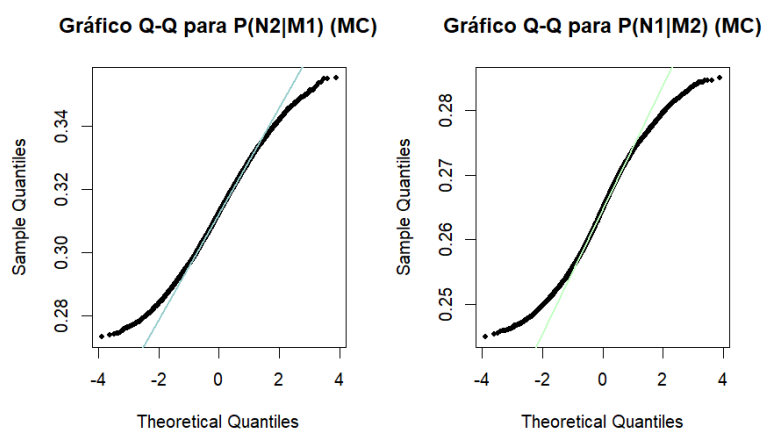


Figura 18. Normalidad (Exp. 3)

En cuanto al análisis de convergencia, las curvas de media acumulada para $P(N2|M1)$ y $P(N1|M2)$ (Fig. 19) tienen un comportamiento similar y se estabilizan relativamente rápido. Se confirma así la hipótesis de convergencia.

Al igual que en el Experimento 1, que esta hipótesis se cumpla confirma que el número de iteraciones que se ha elegido para Monte Carlo (10.000) es más que suficiente para garantizar una estimación estable de la media.

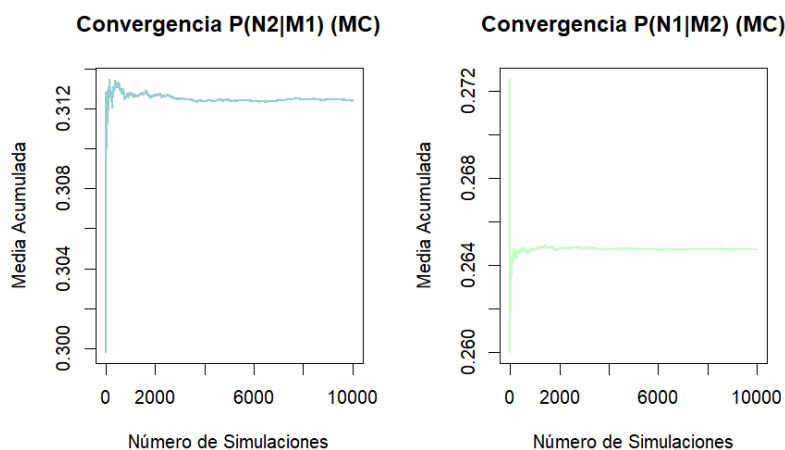


Figura 19. Convergencia (Exp. 3)

Los gráficos ACF (Figura 20) muestran en ambos casos que los coeficientes de autocorrelación están muy cerca de cero, la mayoría estando dentro del intervalo de confianza. Puede por tanto confirmarse la ausencia de autocorrelación (independencia).

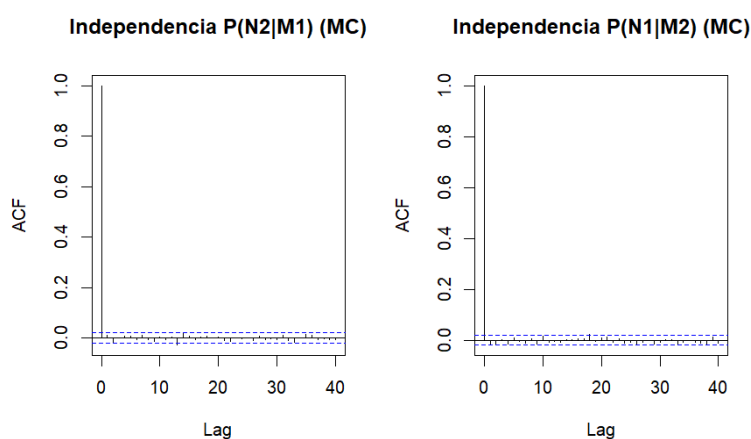


Figura 20. Independencia (Exp. 3)

Como se puede ver en la tabla 8, el estadístico DW para $P(N2|M1)$ es aproximadamente 1.982 ($p \sim 0.186$), y para $P(N1|M2)$ está alrededor de 2.031 ($p \sim 0.937$). Los dos valores se aproximan a 2, que es el indicador teórico de independencia perfecta. Además, los p-valores indican que

no hay evidencia para rechazar la hipótesis de independencia, confirmando lo que mostraban los gráficos ACF.

| Variable respuesta | P(N2 M1) | P(N1 M2) |
|--------------------|----------|----------|
| Estadístico DW | 1.9821 | 2.0306 |
| p-valor | 0.1858 | 0.9371 |

Tabla 8. Test de Durbin-Watson (Exp. 3)

Como hemos comprobado que la simulación de Monte Carlo cumple con las tres hipótesis, podemos afirmar que era una idea acertada el tomarlo como método de referencia para el Experimento 3.

5.4. Discusión general de los resultados

Con los resultados que se han obtenido de los tres experimentos, se puede ver cómo se propaga la incertidumbre según el modelo que se tenga.

Al comparar el Experimento 1 con el 3 nos damos cuenta de la gran diferencia que hay entre sus varianzas. En el primero, la variabilidad de las probabilidades iniciales provoca una dispersión muy elevada, con un CV para P(N2|M1) de 79.2% y de 59.1% para P(N1|M2). Sin embargo, en el tercer experimento obtuvimos una dispersión que apenas afecta al resultado (CV de 4.88% para P(N2|M1) y 3.1% para P(N1|M2)). Lo que esto da a entender es que la fuente más importante de incertidumbre no es la falta de conocimiento de los parámetros del modelo (probabilidades condicionadas), sino de las condiciones iniciales (las probabilidades de estimulación nerviosa).

También llama la atención las correlaciones opuestas de los experimentos 1 y 2 entre P(N2|M1) y P(N1|M2). En el primero, era fuertemente negativa con un valor de -0.691 y, en el segundo, se obtuvo que era fuertemente positiva con +0.82. En el Experimento 1 se debía a que se impuso la condición de que las tres probabilidades P(N1), P(N2) y P(N3) debían sumar 1. La correlación del Experimento 2 ocurría por los efectos combinados de los factores. Lo que se puede extraer de esto es que la dependencia de las variables de salida no es fija, sino que puede variar con la fuente de incertidumbre. Si la incertidumbre se debe a no saber qué nervio es el que se ha estimulado, la correlación va a ser negativa porque las variables van a competir entre ellas. No obstante, si la incertidumbre es por no conocer las probabilidades condicionadas, la correlación será positiva.

Además, en el Experimento 1 se generó una distribución para P(N2|M1) muy asimétrica y otra más uniforme para P(N1|M2). En el Experimento 2 se confirmó esta discrepancia cuando para P(N2|M1) sí se dieron interacciones significativas y para P(N1|M2), no. La diferencia entre las dos probabilidades a posteriori está en la “relación de fuerzas”. En el músculo M1, el nervio N2 ($P(M1|N2) = 0.5$) tiene un competidor muy fuerte ($P(M1|N1) = 0.4$) y otro que es más débil ($P(M1|N3) = 0.2$), y esto se refleja en forma de interacciones. Sin embargo, en el caso del músculo M2, el nervio N1 ($P(M2|N1) = 0.6$) compite con el N2 ($P(M2|N2) = 0.5$), pero a ambos

los opaca el N3 ($P(M2|N3) = 0.8$). La diferencia tan grande que hay entre el nervio N3 y el resto hace mucho más simple el comportamiento del sistema y explica por qué no hay interacciones para $P(N1|M2)$.

Otra discrepancia es que en el Experimento 1 la diagnosis reveló que el modelo no se ajustaba a una distribución normal y que tenía un comportamiento no lineal. Esta no linealidad se confirmó en el Experimento 2, con el que se obtuvieron interacciones significativas para $P(N2|M1)$. La existencia de interacciones demuestra que el modelo no es aditivo y que el efecto de un factor depende del nivel de otro. Ahora bien, en la diagnosis que se le hizo a la simulación de Monte Carlo en el Experimento 3, sí se cumplía la hipótesis de normalidad.

El distinto comportamiento que tienen los dos experimentos se puede explicar por las condiciones iniciales de los modelos: en el primero, la variabilidad se introducía en las probabilidades a priori en un rango grande (entre 0 y 1); mientras que en el tercer experimento se les daba variabilidad a las probabilidades condicionadas y en pequeñas cantidades. El Teorema del Límite Central nos dice que, en condiciones generales, las sumas y medias de muestras aleatorias obtenidas de una población tienden a tener una distribución aproximadamente normal [7]; y esto es justo lo que ha pasado en el Experimento 3.

6. Conclusiones

El principal objetivo de este Trabajo de Fin de Grado era estudiar la probabilidad de implicación de las raíces nerviosas a partir los resultados de la electromiografía (EMG), usando el Teorema de Bayes y expandiendo la incertidumbre con Monte Carlo.

De los tres estudios realizados, se destacan las conclusiones siguientes:

- La principal fuente de incertidumbre proviene de no conocer las probabilidades $P(N_i)$, mucho más que las condicionadas. Esto deriva de la enorme diferencia entre los coeficientes de variación del Experimento 1 y el Experimento 3, que llega a ser del $\sim 79\%$ en el primer caso y de $\sim 5\%$ en el segundo.
- Gracias al ANOVA, se ha podido ver que los factores que más impacto tienen en $P(N_2|M_1)$ y $P(N_1|M_2)$ son $P(M_1|N_2)$ y $P(M_1|N_1)$, respectivamente.
- Existen interacciones entre factores para la variable $P(N_2|M_1)$, pero no se da ninguna para $P(N_1|M_2)$. Esto puede llevar a que la EMG dé resultados menos precisos cuando haya raíces nerviosas que tengan respuestas parecidas, y que la interpretación sea mejor cuando sea una raíz la que domine claramente la respuesta.
- El método de perturbación de Taylor de primer orden es una buena alternativa al método de Monte Carlo cuando la incertidumbre es relativamente pequeña ($\pm 10\%$).
- El modelo de basa en el Teorema de Bayes. Esto significa que su comportamiento general es no lineal, lo que justifica que se usen métodos como el de Monte Carlo para poder estudiar cómo se comporta la incertidumbre en las probabilidades a posteriori.
- En función de qué variables de entrada se conozcan y cuáles no, la correlación de $P(N_2|M_1)$ y $P(N_1|M_2)$ puede ser positiva o negativa. Será negativa si la incertidumbre se da en las probabilidades a priori, y positiva si es por las condicionadas.

7. Líneas futuras

Este trabajo se ha desarrollado en torno a un escenario que incluía tres nervios y dos músculos. Una posible investigación futura podría tratar de abarcar más músculos y raíces nerviosas. De este modo, podría hacerse un modelo donde se evalúen cómo se comporta la incertidumbre y las posibles interacciones en una situación con más variables involucradas.

En el Experimento 1 se estudió la hipótesis de normalidad para ver la forma que tenía la incertidumbre de salida cuando la variabilidad provenía de las probabilidades a priori. Resultó que las distribuciones no se ajustaban a una normal. Por eso, otro paso futuro podría ser el de identificar a qué distribución corresponden los resultados obtenidos.

Por último, y en relación con el primer punto, si se añadiesen más variables podría resultar en unos efectos no lineales mayores. También podría tomarse una incertidumbre más grande que la del $\pm 10\%$. En los dos casos, podrían investigarse comparaciones con el método de Taylor de segundo orden.

8. Bibliografía

- [1] R. Eckhardt, "Stan Ulam, John von Neumann," *Los Alamos Science*, pp. 131, 1987.
- [2] P. Grosse, M. J. Cassidy and P. Brown, "EEG–EMG, MEG–EMG and EMG–EMG frequency analysis: physiological principles and clinical applications," *Clinical Neurophysiology*, vol. 113, (10), pp. 1523–1531, 2002.
- [3] J. Gallardo, "La inervación sensitiva segmentaria. Dermatomas, miotomas y esclerotomas," *Rev Chil Anest*, vol. 37, (1), pp. 26–38, 2008.
- [4] R. Álvarez Fiallo, C. Santos Anzorandia and E. Medina Herrera, "Desarrollo histórico y fundamentos teóricos de la electromiografía como medio diagnóstico," *Revista Cubana De Medicina Militar*, vol. 35, (4), pp. 0, 2006.
- [5] Hallal-Peche, F., Fernández-Lorente, J., Valentin, A., Mira, J., Escobar-Montalvo, J. M., Armas-Zurita, R., González-Mendoza, C., Gil-Ruiz, D., Rúa, L. M., Jiménez, C. M., & Aguilera-Vergara, M. (2025)., "AB-662. How to calculate the probability of nerve root involvement based on the EMG result?" *Clinical Neurophysiology*, vol. 172, pp. S100.
- [6] M. H. Kalos and P. A. Whitlock, *Monte Carlo Methods*. John Wiley & Sons, 2009.
- [7] W. Mendenhall *et al*, *Introducción a La Probabilidad Y Estadística*. 1982.
- [8] C. Condicional and E. Independencia, "Razonamiento sobre probabilidad condicional e implicaciones para la enseñanza de la estadística," *Epsilon*, vol. 59, pp. 245–260, 2005.
- [9] M. G. Cox and B. R. Siebert, "The use of a Monte Carlo method for evaluating uncertainty and expanded uncertainty," *Metrologia*, vol. 43, (4), pp. S178, 2006.
- [10] J. I. Illana, "Métodos Monte Carlo,".
- [11] J. M. O. Sevilla, "La incertidumbre en la medida de una magnitud y el método de Montecarlo," *Técnica Industrial*, vol. 306, pp. 54–64, 2014.
- [12] G. Delgado, "Metodología para la implementación del método adaptativo de Monte Carlo en la evaluación de la incertidumbre de la medición, utilizando el cálculo simbólico Maple. Aplicación a un experimento sencillo," *Universitas (León): Revista Científica De La UNAN León.*, vol. 3, (2), pp. 22–32, 2009.
- [13] N. G. Álvarez *et al*, "Normalidad y otros supuestos," *Modelización Con Estructuras De Covarianzas En Ciencias Sociales: Temas Esenciales, Avanzados Y Aportaciones Especiales*, vol. 31, (2), pp. 122–155, 2006.
-

[14] C. L. Smith, "Uncertainty propagation using Taylor series expansion and a spreadsheet," *Journal of the Idaho Academy of Science*, vol. 30, (2), pp. 93–105, 1994.

[15] H. Gutiérrez Pulido, "Análisis y diseño de experimentos," 2012.

[16] J. Allaire, "RStudio: integrated development environment for R," *Boston, MA*, vol. 770, (394), pp. 165–171, 2012.

[17] C. M. Schirmer *et al*, "Heuristic map of myotomal innervation in humans using direct intraoperative nerve root stimulation," *Journal of Neurosurgery: Spine*, vol. 15, (1), pp. 64–70, 2011.

9. Planificación temporal y presupuesto

A continuación, en la Figura 21, se presenta el diagrama de Gantt donde se refleja la planificación temporal de este trabajo.

El eje temporal del diagrama representa los meses de trabajo efectivo invertidos en este TFG. Los primeros cuatro meses se corresponden con el período de mayo de 2024 a septiembre de 2024 (sin incluir el mes de junio), cuando se inició este trabajo. El mes 5 se corresponde con noviembre de 2024. Posteriormente, se hizo una pausa por períodos de exámenes y otras obligaciones académicas. El trabajo se retomó en julio de 2025 (mes 6) hasta terminarlo en septiembre de 2025 (mes 8).

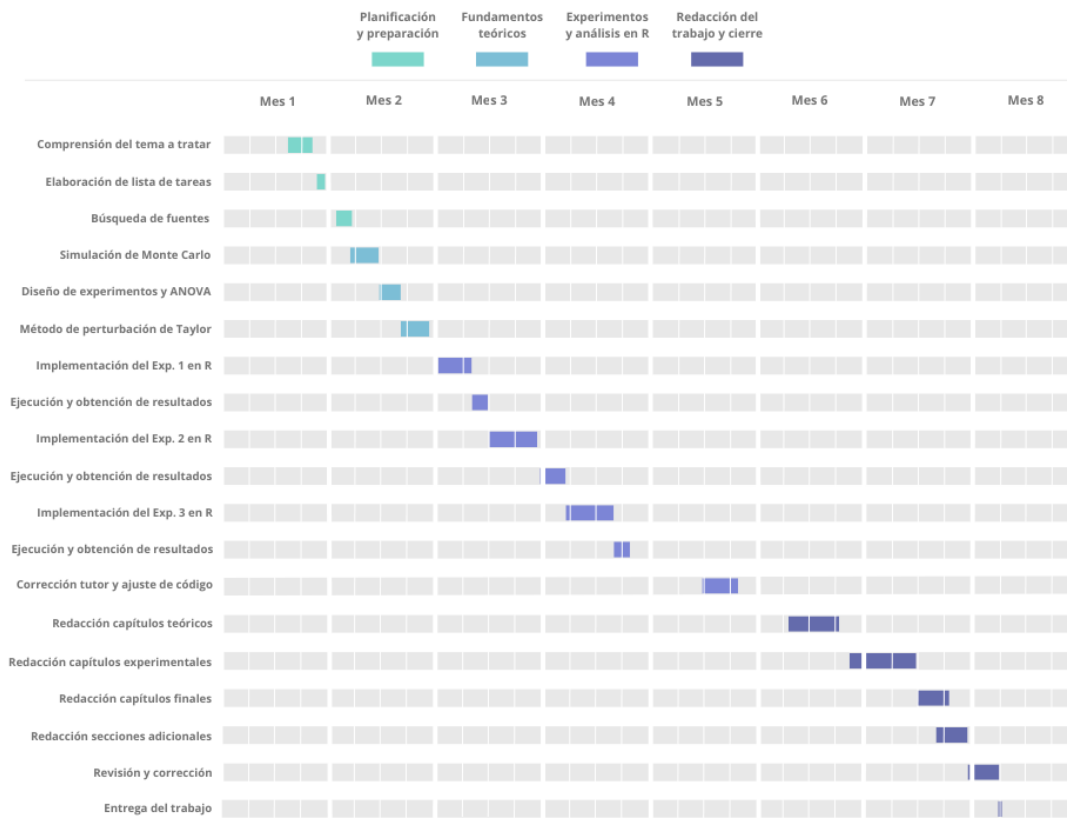


Figura 21. Diagrama de Gantt

En la siguiente tabla se listan con más detalle las tareas con su fecha de inicio y fin:

| Tarea | Fecha inicio | Fecha fin | Duración (días) |
|------------------------------------|-----------------|-----------------|-----------------|
| Planificación y preparación | 23/05/24 | 11/07/24 | 10 |
| Comprensión del tema a tratar | 23/05/24 | 28/05/24 | 5 |
| Elaboración de lista de tareas | 29/05/24 | 31/05/24 | 2 |
| Búsqueda de fuentes | 08/07/24 | 11/07/24 | 3 |
| Fundamentos teóricos | 10/07/24 | 28/07/24 | 16 |

Aplicación de simulación de Monte Carlo a la propagación de incertidumbres en análisis probabilista de electromiografía

| | | | |
|---|-----------------|-----------------|-----------|
| Simulación de Monte Carlo | 10/07/24 | 16/07/24 | 6 |
| Diseño de experimentos y ANOVA | 17/07/24 | 21/07/24 | 4 |
| Método de perturbación de Taylor | 22/07/24 | 28/07/24 | 6 |
| Experimentos y análisis en R | 01/08/24 | 20/11/24 | 54 |
| Implementación del Experimento 1 en R | 01/08/24 | 09/08/24 | 8 |
| Ejecución y obtención de resultados (Exp. 1) | 10/08/24 | 13/08/24 | 3 |
| Implementación del Experimento 2 en R | 14/08/24 | 27/08/24 | 13 |
| Ejecución y obtención de resultados (Exp. 2) | 31/08/24 | 04/09/24 | 4 |
| Implementación del Experimento 3 en R | 05/09/24 | 18/09/24 | 13 |
| Ejecución y obtención de resultados (Exp. 3) | 18/09/24 | 21/09/24 | 3 |
| Corrección del tutor y ajuste de código | 10/11/24 | 20/11/24 | 10 |
| Redacción del trabajo y cierre | 08/07/25 | 07/09/25 | 54 |
| Redacción de capítulos teóricos | 08/07/25 | 22/07/25 | 14 |
| Introducción y Objetivos | 08/07/25 | 12/07/25 | 4 |
| Fundamentos teóricos | 12/07/25 | 22/07/25 | 10 |
| Redacción de capítulos experimentales | 28/07/25 | 15/08/25 | 18 |
| Explicación de los experimentos | 28/07/25 | 09/08/25 | 12 |
| Resultados | 09/08/25 | 15/08/25 | 6 |
| Redacción capítulos finales (conclusiones, resumen ejecutivo, responsabilidad social) | 18/08/25 | 25/08/25 | 7 |
| Redacción secciones adicionales | 23/08/25 | 30/08/25 | 7 |
| Redacción y corrección | 30/08/25 | 06/09/25 | 7 |
| Entrega del trabajo | 06/09/25 | 07/09/25 | 1 |

Tabla 9. Planificación temporal

Como se puede ver, las partes del TFG que han requerido más tiempo el desarrollo e implementación de los tres experimentos en RStudio y la redacción de la memoria. En concreto, lo que más tiempo consumió fue redactar los experimentos y sus resultados, seguido de la redacción de los capítulos teóricos.

9. Planificación temporal y presupuesto

En lo que respecta al presupuesto, se han tenido en cuenta los costes del personal y de material. Los primeros se han contabilizado por horas dedicadas al trabajo, que para el caso de la alumna incluyen las tareas de documentación, planificación, diseño y desarrollo de los experimentos, reuniones y redacción. En el caso del tutor, se cuentan las horas invertidas en reuniones, revisión y corrección de la memoria. Se ha adoptado un precio por hora de 10 € en el caso de la alumna y de 35 € por hora del tutor, basándose en la experiencia y formación de cada uno de ellos.

En cuanto a costes de material, no se tiene en cuenta el software usado (RStudio, Microsoft Office) porque es proporcionado por la universidad. Se contabilizan los costes derivados de la impresión y encuadernación del trabajo, a lo que se le asocia un precio de 20 €.

Todo lo mencionado queda reflejado en la siguiente tabla:

| Concepto | | Descripción | Coste total (€) |
|---------------------|----------------------------|--------------------|-----------------|
| Personal | Claudia Domínguez (Alumna) | 380 horas a 10 €/h | 3.800 € |
| | José Manuel Mira (Tutor) | 28 horas a 35 €/h | 980 € |
| Material de oficina | | 20 € | 20 € |
| Total: | | | 4.800 € |

Tabla 10. Planificación del presupuesto

Por tanto, para este trabajo se estima un presupuesto de 4.800 € en total.

10. Responsabilidad social

10.1. Evaluación de impactos

Las conclusiones obtenidas en este TFG tienen una importancia especial en la monitorización neurofisiológica durante una cirugía (EMG intraoperatoria). El neurocirujano estimulará las raíces nerviosas con una pequeña corriente eléctrica, mientras que el neurofisiólogo observa qué músculos se contraen. Esto se hace para crear un “mapa” de los nervios y protegerlos durante la operación.

Es común que se dé la situación en la que el cirujano estimula lo que él cree que es el N2 (p. ej.). Entonces, el neurofisiólogo observa que el músculo M1 se contrae, pero no tanto como esperaba, e incluso puede que haya una pequeña contracción también en el músculo M2. En ese momento pueden surgir dudas sobre si realmente se está estimulando el nervio correcto, o si la corriente eléctrica ha estimulado al mismo tiempo otro nervio (entre otras). Con los experimentos que hemos hecho, se le podría poner un valor numérico a esas preguntas para que los cirujanos no tengan que tomar una decisión basándose únicamente en su experiencia e intuición. Incluso, gracias al Experimento 2, se podría dar información sobre qué interacciones entre los parámetros afectan más a la respuesta muscular, pudiéndose hacer una estimulación controlada para verificar las respuestas.

Basándonos en este contexto, en este capítulo se van a presentar los posibles impactos, tanto positivos como los que podrían considerarse riesgos, que pudieran estar ligados a este TFG.

Empezando por el impacto social, podría resultar en una mejor toma de decisiones clínicas por parte de los neurofisiólogos y cirujanos, al darle un valor numérico a la incertidumbre de los diagnósticos de la EMG. Igualmente, esto deriva en mejores resultados del paciente, porque se reduce el riesgo de dañar nervios sanos y su recuperación sería más rápida. Desde un punto de vista más académico, este TFG podría ser útil como herramienta para que futuros estudiantes y residentes entiendan la complejidad del modelo de diagnóstico con el que trabajan.

También es importante mencionar el riesgo de un exceso de confianza o una mala interpretación del modelo. Podría pasar, por ejemplo, que un sanitario lo utilice sin acabar de entender sus limitaciones (como la gran dependencia que tiene de las probabilidades a priori), lo que podría terminar en un mal diagnóstico.

También tiene un impacto económico por la reducción de los costes, tanto directos como indirectos. Se agilizaría el proceso y repercutiría también en la estancia hospitalaria del paciente (mejores resultados durante la operación suponen menos tiempo de hospitalización y menos rehabilitación). Sin embargo, debe tenerse en cuenta que podría haber costes asociados a la integración real del modelo, incluyendo, entre otros, la formación del personal sanitario.

Por último, como este trabajo se basa en su totalidad en el desarrollo y ejecución de software, el impacto medioambiental asociado se puede considerar que es prácticamente nulo.

10.2. Análisis de aspectos legales y éticos

En cuanto a los aspectos legales, debería tenerse en cuenta que, si se usase el código de los experimentos en la práctica, probablemente se consideraría un “Software como Dispositivo Médico” (SaMD). Esto significa que tendría que cumplir una serie de regulaciones estrictas que garantizan que su uso es seguro, como el Reglamento de Dispositivos Médicos de la UE. Además, se usan datos de pacientes (EMG) y se debe proteger su anonimato cumpliendo con las leyes de protección de datos. Por último, debería establecerse sobre quién recae la responsabilidad si el modelo da un resultado incorrecto que lleva a un mal resultado en la operación.

Siguiendo con los aspectos éticos, debe plantearse la pregunta sobre si le debe informar o no al paciente sobre que se está usando un modelo computacional para ayudar al médico a hacer su diagnóstico, al igual que la incertidumbre inherente del proceso.

Por otro lado, este trabajo está basado en el estudio de Schirmer et al. (2011), que hizo estimulaciones en un total de 2.295 ubicaciones de raíces nerviosas en 129 pacientes de entre 15 y 57 años, de las cuales 1.589 cumplieron con los criterios de calidad para análisis [17]. Se ha asumido que estos datos representan a toda la población, pero sería recomendable usar el modelo en un grupo demográfico diferente para reducir posibles sesgos.

Por último, se debe recalcar que el modelo es solo una herramienta para ayudar a tomar una decisión, pero que en ningún caso debe sustituir al juicio y la experiencia de los profesionales sanitarios. La decisión final la debe tomar el médico.

10.3. Objetivos de Desarrollo Sostenible

Los Objetivos de Desarrollo Sostenible (ODS), incluidos en la Agenda 2030, son promovidos por la Organización de las Naciones Unidas. Estos son un llamado a nivel global que consta de diecisiete objetivos destinados a impulsar un futuro más sostenible y justo.



Figura 22. Objetivos de Desarrollo Sostenible

En este trabajo, se puede considerar que están presentes los siguientes ODS:

ODS 3: Salud y bienestar. Las conclusiones obtenidas gracias a Monte Carlo facilitan la interpretación de los resultados de la EMG. Esto podría ayudar a los médicos a tomar decisiones

más objetivas y fundamentadas. A su vez, agilizaría el proceso en las operaciones que lo requieran y mejoraría los tratamientos médicos. En última instancia, esto resulta en una mejor calidad de vida del paciente.

ODS 4: Educación de calidad. El propio trabajo en sí es el resultado directo de una educación superior de calidad. Supone el fin de mi formación como ingeniera y busca el desarrollo académico de quienes participan. También se convierte en una potencial fuente de conocimiento que podría enriquecer la educación de futuros estudiantes o investigadores.

ODS 9: Industria, innovación e infraestructura. La meta de este objetivo de potenciar la investigación científica y la innovación está también relacionada con este TFG, pues él mismo es un ejercicio de investigación aplicada, donde se usan técnicas avanzadas como la simulación de Monte Carlo para resolver un problema real.

ODS 17: Alianzas para lograr objetivos. Haber trabajado codo a codo con un neurofisiólogo del Hospital Gómez Ulla (Madrid), ha sido fundamental para darle forma a este trabajo, creando un puente entre lo que es ingeniería y la medicina. No se trata solo de un ejercicio teórico puro, sino que también intenta traer a la práctica conocimientos técnicos para poder resolver una necesidad real que ha sido identificada por un profesional de la salud.

11. Índice de figuras

| | |
|---|----|
| Figura 1. Esquema del método de Monte Carlo para la propagación de incertidumbre..... | 18 |
| Figura 2. Histograma de la distribución de probabilidad para $P(N2 M1)$ (Exp. 1)..... | 55 |
| Figura 3. Histograma de la distribución de probabilidad para $P(N1 M2)$ (Exp. 1)..... | 56 |
| Figura 4. Gráfico de dispersión de las dos variables respuesta (Exp. 1) | 57 |
| Figura 5. Normalidad (Exp.1)..... | 59 |
| Figura 6. Convergencia (Exp. 1)..... | 59 |
| Figura 7. Independencia (Exp. 1)..... | 60 |
| Figura 8. Tabla ANOVA para $P(N2 M1)$ (Exp. 2) | 62 |
| Figura 9. Tabla ANOVA para $P(N1 M2)$ (Exp. 2) | 62 |
| Figura 10. Gráfico de dispersión entre $P(N2 M1)$ y $P(N1 M2)$ (Exp. 2) | 63 |
| Figura 11. Gráficos de efectos principales para $P(N2 M1)$ (Exp. 2)..... | 64 |
| Figura 12. Gráficos de efectos principales para $P(N1 M2)$ (Exp. 2)..... | 64 |
| Figura 13. Gráficos de interacción para $P(N2 M1)$ (Exp. 2)..... | 65 |
| Figura 14. Normalidad de residuos (Exp. 2)..... | 66 |
| Figura 15. Homocedasticidad (Exp. 2) | 67 |
| Figura 16. Gráfico Monte Carlo vs. Taylor para $P(N2 M1)$ (Exp. 3) | 68 |
| Figura 17. Gráfico Monte Carlo vs. Taylor para $P(N1 M2)$ (Exp. 3) | 69 |
| Figura 18. Normalidad (Exp. 3)..... | 70 |
| Figura 19. Convergencia (Exp. 3)..... | 71 |
| Figura 20. Independencia (Exp. 3)..... | 71 |
| Figura 21. Diagrama de Gantt..... | 78 |
| Figura 22. Objetivos de Desarrollo Sostenible | 82 |

12. Índice de tablas

| | |
|---|-----|
| Tabla 1. Resultados Experimento 1 | 55 |
| Tabla 2. CV e Intervalos de Confianza (Exp. 1)..... | 58 |
| Tabla 3. Test de Durbin-Watson (Exp. 1)..... | 60 |
| Tabla 4. Valores obtenidos de los 27 escenarios del ANOVA (Exp. 2)..... | 61 |
| Tabla 5. Comparación Monte Carlo vs. Taylor para $P(N2 M1)$ (Exp. 3) | 67 |
| Tabla 6. Comparación Monte Carlo vs. Taylor para $P(N1 M2)$ (Exp. 3) | 67 |
| Tabla 7. CV e Intervalos de Confianza (Exp. 3)..... | 70 |
| Tabla 8. Test de Durbin-Watson (Exp. 3)..... | 72 |
| Tabla 9. Planificación temporal | 79 |
| Tabla 10. Planificación del presupuesto | 80 |
| Tabla 11. Tabla IC del factor F_{PM1N1} de $P(N2 M1)$ (Exp. 2) | 109 |
| Tabla 12. Tabla IC del factor F_{PM1N2} de $P(N2 M1)$ (Exp. 2) | 109 |
| Tabla 13. Tabla IC del factor F_{PM1N3} de $P(N2 M1)$ (Exp. 2) | 109 |
| Tabla 14. Tabla IC del factor F_{PM1N1} de $P(N1 M2)$ (Exp. 2) | 109 |
| Tabla 15. Tabla IC del factor F_{PM1N2} de $P(N1 M2)$ (Exp. 2) | 109 |
| Tabla 16. Tabla IC del factor F_{PM1N3} de $P(N1 M2)$ (Exp. 2) | 109 |
| Tabla 17. Tabla IC de interacción $F_{PM1N1} - F_{PM1N2}$ de $P(N2 M1)$ (Exp. 2)..... | 110 |
| Tabla 18. Tabla IC de interacción $F_{PM1N1} - F_{PM1N3}$ de $P(N2 M1)$ (Exp. 2)..... | 110 |

13. Unidades, abreviaturas y acrónimos

- *Unidades.*

mA: Miliamperio. Unidad de corriente eléctrica utilizada para medir la estimulación de la raíz nerviosa.

- *Abreviaturas.*

CV: Coeficiente de Variación.

Et al.: Y otros (*et alii*).

Exp.: Experimento.

Fig.: Figura.

IC: Intervalo de Confianza.

P. ej.: Por ejemplo.

Vs.: versus.

- *Acrónimos.*

ACF: Función de autocorrelación.

ANOVA: Análisis de la varianza (*ANalysis Of VAriance*).

DoE: Diseño de experimentos (*Design of Experiments*).

DW: Durbin-Watson (test).

EMG: Electromiografía.

GUM: Guía para la expresión de la incertidumbre en la medición (*Guide to the expression of Uncertainty in Measurement*).

LCG: Generador congruente lineal (*Lineal Congruential Generator*).

MC: Monte Carlo.

MCG: Generador congruente multiplicativo (*Multiplicative Congruential Generator*).

MSA: Cuadrado medio asociado al factor (*Mean Square for the factor A*).

MSE: Cuadrado medio del error (*Mean Square for Error*).

ODS: Objetivos de Desarrollo Sostenible.

PRNG: Generador de números pseudoaleatorios (*Pseudo-Random Number Generator*).

SaMD: Software como dispositivo médico (*Software as Medical Device*).

SST: Suma total de cuadrados (*Total Sum of Squares*).

SSA: Suma de cuadrados del factor A (*Sum of Squares for factor A*).

SSE: Suma de cuadrados del error (*Sum of Squares for Error*).

TFG: Trabajo de Fin de Grado.

TLC: Teorema del Límite Central.

14. Glosario

ANOVA: Análisis de varianza. Es una prueba que se usa en Estadística para comparar las medias de varios grupos y decidir si existen diferencias significativas entre ellas.

Autocorrelación: Mide la relación de una variable consigo misma en diferentes momentos en el tiempo. Cuando se habla de una simulación, se utiliza para comprobar que cada iteración es independiente de la anterior.

Convergencia: Indica si las variables aleatorias se estabilizan alrededor de un cierto valor a medida que aumenta el tamaño de la muestra.

Correlación de Pearson: Coeficiente “r” que mide la intensidad y la dirección de la relación lineal entre dos variables y que varía entre -1 y 1 .

Diseño factorial: Un tipo de diseño de experimentos que estudia cómo afectan varios factores (variables independientes) a una variable de salida. Se prueban todas las combinaciones posibles de sus niveles al mismo tiempo y permite analizar el efecto de cada factor por separado y cómo interactúan entre sí.

Distribución uniforme: Una distribución de probabilidad continua donde cada resultado tiene la misma probabilidad de ocurrir.

Electromiografía intraoperatoria (EMG): Una técnica de monitoreo intraoperatorio que registra la actividad eléctrica de los músculos a tiempo real durante la operación. Se hace para conocer el estado de los nervios que los inervan.

Homocedasticidad: Establece que la varianza de los errores o residuos de un modelo es constante para todos los niveles de las variables independientes.

Inervar: Alcanzar un órgano o parte del cuerpo y transmitirle impulsos nerviosos.

Interacciones: Efectos que son el resultado de la combinación de dos o más factores y que no podrían explicarse sumando los efectos por separado de cada factor.

Intervalo de confianza: Rango de valores calculado a partir de unos datos que contiene el valor real del parámetro poblacional con una probabilidad concreta (normalmente 95%).

Normalidad: Propiedad de un conjunto de datos de seguir una distribución normal (campana de Gauss). Los valores se agrupan en el centro y disminuyen hacia los extremos.

Monte Carlo: Método no determinista que utiliza el muestreo aleatorio repetido. Así, obtiene una serie de resultados probables.

Perturbación de Taylor: Método analítico. Propaga la incertidumbre aproximando la media y la varianza de una función con un desarrollo en serie de Taylor de primer orden.

P-valor: Es la probabilidad de rechazar la hipótesis nula basándose en los datos que se han obtenido. Si el p-valor es menor que alfa (normalmente 0.05), se rechaza la hipótesis nula; y se acepta si es mayor.

Probabilidad condicionada: Probabilidad de que ocurra un evento, ya conociendo que otro ha ocurrido antes. Se escribe como $P(A|B)$.

Probabilidad a priori: Probabilidad de que ocurra un evento basándose en información previa.

Probabilidad a posteriori: Probabilidad actualizada de un evento cuando se han observado nuevos datos o evidencia.

Propagación de incertidumbre: Proceso en el que se mide cómo la variabilidad de las variables de entrada afecta a la variabilidad de los resultados de un modelo.

Residuos (en ANOVA): Diferencia entre el valor real observado y el que ha estimado el modelo. En ANOVA, los residuos se relacionan con la variabilidad que no explican los factores.

15. Anexos

15.1. Anexo 1: Código en R del Experimento 1

```
{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)

-----BLOQUE PREVIO-----
rm(list = ls()) # borrar la memoria
graphics.off() # borrar todos los gráficos abiertos en R

-----GENERACIÓN PROBABILIDADES ALEATORIAS PARA N1, N2 y N3-----
Fijamos la semilla para reproducibilidad
set.seed(123)

# PROBABILIDADES ALEATORIAS PARA N1 Y N2:

probN1 = runif(10000, min = 0, max = 1) # Probabilidad de estimular el
nervio N1

probN2 = runif(10000, min = 0, max = 1 - probN1) # Probabilidad de
estimular el nervio N2

probN3 = 1 - probN1 - probN2 # Probabilidad de estimular el nervio N3

# Mostramos las primeras filas para verificar
head(cbind(probN1, probN2, probN3))

-----Probabilidades condicionadas (dato)-----
P_M1_N1_dato = 0.4
P_M1_N2_dato = 0.5
P_M1_N3_dato = 0.2
P_M2_N1_dato = 1 - P_M1_N1_dato
P_M2_N2_dato = 1 - P_M1_N2_dato
P_M2_N3_dato = 1 - P_M1_N3_dato
```

```
-----PROBABILIDAD P(M1)-----
# Inicializamos el vector de resultados para P(M1) para guardar resultados
P_M1 = numeric(10000)
# Probabilidad total de que se contraiga primero M1
for (i in 1:10000) {
  P_M1[i] = (probN1[i] * P_M1_N1_dato)
  + (probN2[i] * P_M1_N2_dato) + (probN3[i] * P_M1_N3_dato) }

-----PROBABILIDAD P(N2|M1)-----
# Inicializamos el vector de resultados para P(N2|M1)
P_N2_M1 = numeric(10000)
# Teorema de Bayes P(N2|M1)
for (i in 1:10000) {
  P_N2_M1[i] = (probN2[i] * P_M1_N2_dato) / P_M1[i] }
# Mostrar las primeras filas del resultado
head(P_N2_M1)

-----PROBABILIDAD P(M2)-----
# Probabilidad total de que se contraiga primero M2
P_M2 = numeric(10000)
for (i in 1:10000) {
  P_M2[i] = (probN1[i] * P_M2_N1_dato)
  + (probN2[i] * P_M2_N2_dato) + (probN3[i] * P_M2_N3_dato) }

-----PROBABILIDAD P(N1|M2)-----
P_N1_M2 = numeric(10000)
# Teorema de Bayes P(N1|M2)
for (i in 1:10000) {
  P_N1_M2[i] = (probN1[i] * P_M2_N1_dato) / P_M2[i] }
# Mostrar las primeras filas del resultado
head(P_N1_M2)
```

15. Anexos

```
----- ANÁLISIS DE RESULTADOS -----  
  
# MEDIA, VARIANZA Y CUARTILES P(N2|M1)  
mean_P_N2_M1 = mean(P_N2_M1)  
var_P_N2_M1 = var(P_N2_M1)  
quantiles_P_N2_M1 = quantile(P_N2_M1, probs = c(0.25, 0.5, 0.75))  
print(paste("Media de P(N2|M1):", mean_P_N2_M1))  
print(paste("Varianza de P(N2|M1):", var_P_N2_M1))  
print("Cuartiles de P(N2|M1):") print(quantiles_P_N2_M1)  
  
# MEDIA, VARIANZA Y CUARTILES P(N1|M2)  
mean_P_N1_M2 = mean(P_N1_M2)  
var_P_N1_M2 = var(P_N1_M2)  
quantiles_P_N1_M2 = quantile(P_N1_M2, probs = c(0.25, 0.5, 0.75))  
print(paste("Media de P(N1|M2):", mean_P_N1_M2))  
print(paste("Varianza de P(N1|M2):", var_P_N1_M2))  
print("Cuartiles de P(N1|M2):")  
print(quantiles_P_N1_M2)  
  
# ESTADÍSTICAS DESCRIPTIVAS  
summary (P_N2_M1)  
summary (P_N1_M2)  
  
# HISTOGRAMAS  
hist(P_N2_M1, breaks = 50, col = "lightblue",  
      main = "Distribución de P(N2|M1)", xlab = "P(N2|M1)")  
hist(P_N1_M2, breaks = 50, col = "lightgreen",  
      main = "Distribución de P(N1|M2)", xlab = "P(N1|M2)")  
  
# CORRELACIÓN P(N2|M1) Y P(N1|M2)  
cor_value = cor(P_N2_M1, P_N1_M2)  
print(paste("Correlación entre P(N2|M1) y P(N1|M2):",
```

```
round(cor_value, 4)))  
  
# Scatter plot para visualizar la relación  
plot(P_N2_M1, P_N1_M2, main = "Correlación entre P(N2|M1) y P(N1|M2)",  
      xlab = "P(N2|M1)", ylab = "P(N1|M2)", pch = 20,  
      col = rgb(0.1, 0.3, 0.7, 0.3))  
  
# CÁLCULO DEL COEFICIENTE DE VARIACIÓN (CV)  
  
# Para P(N2|M1)  
CV_N2_M1 = sd(P_N2_M1) / mean(P_N2_M1)  
  
# Para P(N1|M2)  
CV_N1_M2 = sd(P_N1_M2) / mean(P_N1_M2)  
  
# Muestra los resultados como porcentaje  
cat("CV para P(N2|M1):", CV_N2_M1 * 100, "%\n")  
cat("CV para P(N1|M2):", CV_N1_M2 * 100, "%\n")  
  
# CÁLCULO DEL INTERVALO DE CONFIANZA (IC 95%)  
  
# Calcular IC para P(N2|M1)  
IC_95_N2_M1 = quantile(P_N2_M1, probs = c(0.025, 0.975))  
  
# Calcular IC para P(N1|M2)  
IC_95_N1_M2 = quantile(P_N1_M2, probs = c(0.025, 0.975))  
  
# Mostrar los resultados  
print(IC_95_N2_M1)  
print(IC_95_N1_M2)  
  
-----DIAGNOSIS DE LA SIMULACIÓN-----  
  
par(mfrow = c(2, 2))  
  
# ANÁLISIS DE NORMALIDAD (Gráficos Q-Q)  
# Gráfico Q-Q para P(N2|M1)  
qqnorm(P_N2_M1, main = "Gráfico Q-Q para P(N2|M1)", pch = 20)  
qqline(P_N2_M1, col = "steelblue", lwd = 2)  
  
# Gráfico Q-Q para P(N1|M2)
```

15. Anexos

```
qqnorm(P_N1_M2, main = "Gráfico Q-Q para P(N1|M2)", pch = 20)
qqline(P_N1_M2, col = "maroon3", lwd = 2)

# ANÁLISIS DE CONVERGENCIA (Media Acumulada)
# Convergencia para P(N2|M1)
media_acum_N2_M1 = cumsum(P_N2_M1) / seq_along(P_N2_M1)
plot(media_acum_N2_M1, type = "l", col = "steelblue", lwd = 2,
      main = "Convergencia Media Acumulada P(N2|M1)",
      xlab = "Número de Simulaciones", ylab = "Media Acumulada")

# Convergencia para P(N1|M2)
media_acum_N1_M2 = cumsum(P_N1_M2) / seq_along(P_N1_M2)
plot(media_acum_N1_M2, type = "l", col = "maroon3", lwd = 2,
      main = "Convergencia Media Acumulada P(N1|M2)",
      xlab = "Número de Simulaciones", ylab = "Media Acumulada")

# Resetear la ventana de gráficos a 1 solo gráfico
par(mfrow = c(1, 1))

# ANÁLISIS DE INDEPENDENCIA (Autocorrelación - ACF)
par(mfrow = c(1, 2))
# ACF para P(N2|M1)
acf(P_N2_M1, main = "Análisis Independencia P(N2|M1)")

# ACF para P(N1|M2)
acf(P_N1_M2, main = "Análisis Independencia P(N1|M2)")

# Resetear la ventana de gráficos a 1 solo gráfico
par(mfrow = c(1, 1))

# TEST DURBIN - WATSON COMO SOPORTE
library(lmtest)

# Test para P(N2|M1)
dw_test_N2_M1 = dwtest(P_N2_M1 ~ 1)
print(dw_test_N2_M1)

# Test para P(N1|M2)
dw_test_N1_M2 = dwtest(P_N1_M2 ~ 1)
print(dw_test_N1_M2)
```

15.2. Anexo 2: Código en R del Experimento 2

```
# Propagación de la variabilidad de las probabilidades condicionadas
P(M1|N1)=0.4, P(M1|N2)=0.5, P(M1|N3)=0.2

# Fijamos la semilla para reproducibilidad
set.seed(123)

-----PASO 1: Definir los factores y sus niveles-----

# Consideramos una incertidumbre del 10% en los valores de las probabilidades
condicionadas

# Niveles del diseño factorial (3 niveles por factor)
niveles_PM1N1 = c(0.4 * 0.9, 0.4, 0.4 * 1.1)
niveles_PM1N2 = c(0.5 * 0.9, 0.5, 0.5 * 1.1)
niveles_PM1N3 = c(0.2 * 0.9, 0.2, 0.2 * 1.1)

design_matriz = expand.grid(
  F_PM1N1 = factor(niveles_PM1N1),
  F_PM1N2 = factor(niveles_PM1N2),
  F_PM1N3 = factor(niveles_PM1N3) )

-----PASO 2: Ejecutar una simulación MC para cada combinación-----

# Inicializar vectores para guardar los resultados (medias)
resultados_N2_M1 = numeric(27)
resultados_N1_M2 = numeric(27)
n_sims = 10000

# Tiempo de ejecución Monte Carlo:
start_time_montecarlo = Sys.time()
# BUCLE EXTERNO (BUCLE PRINCIPAL)
for (i in 1:nrow(design_matriz)) {
  P_M1_N1_fixed = as.numeric(as.character(design_matriz$F_PM1N1[i]))
  P_M1_N2_fixed = as.numeric(as.character(design_matriz$F_PM1N2[i]))
```

15. Anexos

```
P_M1_N3_fixed = as.numeric(as.character(design_matriz$F_PM1N3[i]))
P_M2_N1_fixed = 1 - P_M1_N1_fixed
P_M2_N2_fixed = 1 - P_M1_N2_fixed
P_M2_N3_fixed = 1 - P_M1_N3_fixed

# INICIO SIMULACIÓN MONTE CARLO
# Generamos probabilidades aleatorias P(N1), P(N2), P(N3)
probN1_dos = runif(n_sims, min = 0, max = 1)
probN2_dos = runif(n_sims, min = 0, max = 1 - probN1_dos)
probN3_dos = 1 - probN1_dos - probN2_dos

# Inicializar vectores
P_M1_dos = numeric(n_sims)
P_M2_dos = numeric(n_sims)
P_N2_M1_dos = numeric(n_sims)
P_N1_M2_dos = numeric(n_sims)

# Simulación por Monte Carlo (BUCLE INTERNO)
for (j in 1:n_sims) {
  # Ley de probabilidad total
  P_M1_dos[j] = (probN1_dos[j] * P_M1_N1_fixed)
    + (probN2_dos[j] * P_M1_N2_fixed)
    + (probN3_dos[j] * P_M1_N3_fixed)
  P_M2_dos[j] = (probN1_dos[j] * P_M2_N1_fixed)
    + (probN2_dos[j] * P_M2_N2_fixed)
    + (probN3_dos[j] * P_M2_N3_fixed)

  # Teorema de Bayes
  P_N2_M1_dos[j] = (probN2_dos[j] * P_M1_N2_fixed) / P_M1_dos[j]
  P_N1_M2_dos[j] = (probN1_dos[j] * P_M2_N1_fixed) / P_M2_dos[j]
}

# GUARDAMOS MEDIA DE LAS 10.000 SIMULACIONES
resultados_N2_M1[i] = mean(P_N2_M1_dos)
resultados_N1_M2[i] = mean(P_N1_M2_dos)
cat(paste("Completado escenario", i, "de 27\n")) }
```

```
# Tiempo de ejecución final Monte Carlo
end_time_montecarlo = Sys.time()

montecarlo_time = end_time_montecarlo - start_time_montecarlo
print(paste("Tiempo de ejecución Monte Carlo:", montecarlo_time))

# Añadimos medias a matriz de diseño
design_matriz$P_N2_M1_mean = resultados_N2_M1
design_matriz$P_N1_M2_mean = resultados_N1_M2
print("Resultados del Diseño Factorial (Medias)")
print(design_matriz)

----- ANÁLISIS DE SENSIBILIDAD (ANOVA) SOBRE LAS MEDIAS-----

# ANÁLISIS PARA P(N2|M1)
aov_N2_M1 = aov(P_N2_M1_mean ~ (F_PM1N1 + F_PM1N2 + F_PM1N3)^2,
               data = design_matriz)

# Guardamos y extraemos la tabla completa
summary_aov1 = summary(aov_N2_M1)
tabla_anova_1 = summary_aov1[[1]]

cat("\n Tabla ANOVA para P(N2|M1) \n")
print(tabla_anova_1)

# ANÁLISIS PARA P(N1|M2)
aov_N1_M2 = aov(P_N1_M2_mean ~ (F_PM1N1 + F_PM1N2 + F_PM1N3)^2,
               data = design_matriz)

summary_aov2 = summary(aov_N1_M2)
tabla_anova_2 = summary_aov2[[1]]

cat("\n Tabla ANOVA para P(N1|M2) \n")
print(tabla_anova_2)

# GRÁFICO DE CORRELACIÓN ENTRE P(N2|M1) Y P(N1|M2)

# Coeficiente de correlación entre las medias
cor_exp2 = cor(design_matriz$P_N2_M1_mean, design_matriz$P_N1_M2_mean)
print(paste("Coeficiente de correlación:", round(cor_exp2, 4)))
```

15. Anexos

```
# Gráfico de dispersión para visualizar la relación
plot(design_matriz$P_N2_M1_mean, design_matriz$P_N1_M2_mean,
      main = "Correlación entre Respuestas Promedio (Exp. 2)",
      xlab = "Media de P(N2|M1)", ylab = "Media de P(N1|M2)",
      pch = 20, col = "lightpink")

-----INTERVALOS CONFIANZA (95%) EXPPERIMENTO-----

# Primero se cargan las funciones
source("ICplot.R")
source("interIC.R")

# Análisis de los efectos principales para P(N2|M1)
cat("IC para Efectos Principales sobre P(N2|M1)\n")
IC_F1_N2M1 = ICplot(aov_N2_M1, "F_PM1N1")
IC_F2_N2M1 = ICplot(aov_N2_M1, "F_PM1N2")
IC_F3_N2M1 = ICplot(aov_N2_M1, "F_PM1N3")
# Se imprimen las tablas que genera la función
print(IC_F1_N2M1)
print(IC_F2_N2M1)
print(IC_F3_N2M1)

# Análisis de las interacciones significativas para P(N2|M1)
cat("IC para Interacciones sobre P(N2|M1)\n")
IC_int12_N2M1 = interIC(aov_N2_M1, "F_PM1N1", "F_PM1N2")
IC_int13_N2M1 = interIC(aov_N2_M1, "F_PM1N1", "F_PM1N3")
# Se imprimen las tablas
print(IC_int12_N2M1)
print(IC_int13_N2M1)

# Análisis de los efectos principales para P(N1|M2)
cat("IC para Efectos Principales sobre P(N1|M2)\n")
```

```
IC_F1_N1M2 = ICplot(aov_N1_M2, "F_PM1N1")
IC_F2_N1M2 = ICplot(aov_N1_M2, "F_PM1N2")
IC_F3_N1M2 = ICplot(aov_N1_M2, "F_PM1N3")

# Se imprimen las tablas
print(IC_F1_N1M2)
print(IC_F2_N1M2)
print(IC_F3_N1M2)

-----DIAGNOSIS DEL MODELO ANOVA-----

par(mfrow = c(2, 2))

# NORMALIDAD DE RESIDUOS

# Para el ANOVA de P(N2|M1)
qqnorm(aov_N2_M1$residuals, main = "Q-Q Plot de los Residuos de P(N2|M1)")
qqline(aov_N2_M1$residuals, col = "darkolivegreen1")

# Para el ANOVA de P(N1|M2)
qqnorm(aov_N1_M2$residuals, main = "Q-Q Plot de los Residuos de P(N1|M2)")
qqline(aov_N1_M2$residuals, col = "lightsteelblue1")

# HOMOGENEIDAD DE VARIANZAS (HOMOCEDEASTICIDAD)

# Gráfico de Residuos vs. Valores Ajustados de P(N2|M1)
plot(aov_N2_M1$fitted.values, aov_N2_M1$residuals,
     main = "Residuos vs. Valores Ajustados P(N2|M1)",
     xlab = "Valores Ajustados", ylab = "Residuos")
abline(h = 0, col = "darkolivegreen1", lty = 2)

# Gráfico de Residuos vs. Valores Ajustados de P(N1|M2)
plot(aov_N1_M2$fitted.values, aov_N1_M2$residuals,
     main = "Residuos vs. Valores Ajustados P(N1|M2)",
     xlab = "Valores Ajustados", ylab = "Residuos")
abline(h = 0, col = "lightsteelblue1", lty = 2)

par(mfrow = c(1, 1))
```

15.3. Anexo 3: Código en R del Experimento 3

```
-----SIMULACIÓN DE MONTE CARLO-----  
  
set.seed(123)  
  
start_time_mc = Sys.time()  
  
# PROBABILIDADES ALEATORIAS PARA N1 Y N2  
# Se mantienen fijas: P(N1)=0.3, P(N2)=0.2, P(N3)=0.5  
P_M1_N1_MC = runif(10000, min = 0.36, max = 0.44) # +-10% de 0.4  
P_M1_N2_MC = runif(10000, min = 0.45, max = 0.55) # +-10% de 0.5  
P_M1_N3_MC = runif(10000, min = 0.18, max = 0.22) # +-10% de 0.2  
P_M2_N1_MC = 1 - P_M1_N1_MC  
P_M2_N2_MC = 1 - P_M1_N2_MC  
P_M2_N3_MC = 1 - P_M1_N3_MC  
  
# P(M1):  
P_M1_MC = numeric(10000)  
for (i in 1:10000) {  
  P_M1_MC[i] = (probn1_MC * P_M1_N1_MC[i])  
  + (probn2_MC * P_M1_N2_MC[i]) + (probn3_MC * P_M1_N3_MC[i]) }  
# (P(N2|M1)):  
P_N2_M1_MC = numeric(10000)  
for (i in 1:10000) {  
  P_N2_M1_MC[i] = (probn2_MC * P_M1_N2_MC[i]) / P_M1_MC[i] }  
# P(M2):  
P_M2_MC = numeric(10000)  
for (i in 1:10000) {  
  P_M2_MC[i] = (probn1_MC * P_M2_N1_MC[i])  
  + (probn2_MC * P_M2_N2_MC[i]) + (probn3_MC * P_M2_N3_MC[i]) }  
#P(N1|M2):  
P_N1_M2_MC = numeric(10000)  
for (i in 1:10000) {
```

```
P_N1_M2_MC[i] = (probN1_MC * P_M2_N1_MC[i]) / P_M2_MC[i] }

# Media y varianza de los resultados de Monte Carlo
mean_P_N2_M1_MC = mean(P_N2_M1_MC)
var_P_N2_M1_MC = var(P_N2_M1_MC)
mean_P_N1_M2_MC = mean(P_N1_M2_MC)
var_P_N1_M2_MC = var(P_N1_M2_MC)

end_time_mc = Sys.time()
time_mc = end_time_mc - start_time_mc

# CÁLCULO DEL COEFICIENTE DE VARIACIÓN (CV)
# Para P(N2|M1)_MC
CV_N2_M1_MC = sd(P_N2_M1_MC) / mean(P_N2_M1_MC)
# Para P(N1|M2)_MC
CV_N1_M2_MC = sd(P_N1_M2_MC) / mean(P_N1_M2_MC)
# Muestra los resultados como porcentaje
cat("CV (MC) para P(N2|M1):", CV_N2_M1_MC * 100, "%\n")
cat("CV (MC) para P(N1|M2):", CV_N1_M2_MC * 100, "%\n")

# CÁLCULO DEL INTERVALO DE CONFIANZA (IC 95%)
# IC para P(N2|M1)_MC
IC_95_N2_M1_MC = quantile(P_N2_M1_MC, probs = c(0.025, 0.975))
# IC para P(N1|M2)_MC
IC_95_N1_M2_MC = quantile(P_N1_M2_MC, probs = c(0.025, 0.975))
# Imprimir resultados
print(IC_95_N2_M1_MC)
print(IC_95_N1_M2_MC)

----- MÉTODO PERTURBACIÓN DE TAYLOR (Primer orden) -----
start_time_taylor = Sys.time()
```

15. Anexos

```
# Valores medios (MEDIAS) de las entradas
P_N1_taylor = 0.3
P_N2_taylor = 0.2
P_N3_taylor = 0.5
P_M1_N1_taylor = 0.4
P_M1_N2_taylor = 0.5
P_M1_N3_taylor = 0.2

# Definir las VARIANZAS de las entradas
var_PM1N1_taylor = (0.44 - 0.36)^2 / 12
var_PM1N2_taylor = (0.55 - 0.45)^2 / 12
var_PM1N3_taylor = (0.22 - 0.18)^2 / 12

# VALORES DE LAS FUNCIONES EN EL PUNTO MEDIO
P_M1_taylor = (P_N1_taylor * P_M1_N1_taylor)
              + (P_N2_taylor * P_M1_N2_taylor)
              + (P_N3_taylor * P_M1_N3_taylor)
P_M2_taylor = (P_N1_taylor * (1-P_M1_N1_taylor))
              + (P_N2_taylor * (1-P_M1_N2_taylor))
              + (P_N3_taylor * (1-P_M1_N3_taylor))

mean_N2_M1_taylor = (P_M1_N2_taylor * P_N2_taylor) / P_M1_taylor
mean_N1_M2_taylor = ((1-P_M1_N1_taylor) * P_N1_taylor) / P_M2_taylor

# DERIVADAS PARCIALES EN EL PUNTO MEDIO
# Para P(N2|M1):
d_N2M1_d_p1 = - (P_M1_N2_taylor * P_N2_taylor * P_N1_taylor) /
              (P_M1_taylor^2)
d_N2M1_d_p2 = (P_N2_taylor * P_M1_taylor
              - P_M1_N2_taylor * P_N2_taylor * P_N2_taylor) / (P_M1_taylor^2)
d_N2M1_d_p3 = - (P_M1_N2_taylor * P_N2_taylor * P_N3_taylor) /
```

```
(P_M1_taylor^2)

# Para P(N1|M2):
d_N1M2_d_p1 = (-P_N1_taylor * P_M2_taylor
  - (1-P_M1_N1_taylor) * P_N1_taylor * (-P_N1_taylor)) /
  (P_M2_taylor^2)
d_N1M2_d_p2 = ((1-P_M1_N1_taylor) * P_N1_taylor * P_N2_taylor) /
  (P_M2_taylor^2)
d_N1M2_d_p3 = ((1-P_M1_N1_taylor) * P_N1_taylor * P_N3_taylor) /
  (P_M2_taylor^2)

# FÓRMULA DE PROPAGACIÓN DE VARIANZA
var_N2_M1_taylor = (d_N2M1_d_p1^2 * var_PM1N1_taylor)
  + (d_N2M1_d_p2^2 * var_PM1N2_taylor)
  + (d_N2M1_d_p3^2 * var_PM1N3_taylor)
var_N1_M2_taylor = (d_N1M2_d_p1^2 * var_PM1N1_taylor)
  + (d_N1M2_d_p2^2 * var_PM1N2_taylor)
  + (d_N1M2_d_p3^2 * var_PM1N3_taylor)

end_time_taylor = Sys.time()
time_taylor = end_time_taylor - start_time_taylor

-----COMPARACIÓN DE RESULTADOS-----
# TABLA COMPARACIÓN
resultados_N2_M1_comp = data.frame(
  Metodo = c("Monte Carlo", "Taylor"),
  Media = c(mean_P_N2_M1_MC, mean_N2_M1_taylor),
  Varianza = c(var_P_N2_M1_MC, var_N2_M1_taylor),
  Tiempo_Ejecucion_s = c(as.numeric(time_mc), as.numeric(time_taylor)) )
resultados_N1_M2_comp = data.frame(
  Metodo = c("Monte Carlo", "Taylor"),
```

15. Anexos

```
Media = c(mean_P_N1_M2_MC, mean_N1_M2_taylor),
Varianza = c(var_P_N1_M2_MC, var_N1_M2_taylor),
Tiempo_Ejecucion_s = c(as.numeric(time_mc), as.numeric(time_taylor)) )
# Imprimimos las tablas
cat("\n COMPARACIÓN PARA P(N2|M1) \n")
print(resultados_N2_M1_comp)
cat("\n COMPARACIÓN PARA P(N1|M2) \n")
print(resultados_N1_M2_comp)

# COMPARACIÓN VISUAL
# Gráfico para P(N2|M1):
hist(P_N2_M1_MC, breaks = 50, col = "paleturquoise2",
      border = "grey", main = "Comparación para P(N2|M1)",
      xlab = "P(N2|M1)", ylab = "Densidad", freq = FALSE)
x_curve = seq(min(P_N2_M1_MC), max(P_N2_M1_MC), length.out = 100)
taylor_sd = sqrt(var_N2_M1_taylor)
y_curve = dnorm(x_curve, mean = mean_N2_M1_taylor, sd = taylor_sd)
lines(x_curve, y_curve, col = "plum2", lwd = 2)
legend("topright", legend = c("Monte Carlo", "Aprox. Taylor"),
      fill = c("paleturquoise2", "plum2"))

# Gráfico para P(N1|M2):
hist(P_N1_M2_MC, breaks = 50, col = "darkseagreen2",
      border = "grey", main = "Comparación para P(N1|M2)",
      xlab = "P(N1|M2)", ylab = "Densidad", freq = FALSE)
x_curve_2 = seq(min(P_N1_M2_MC), max(P_N1_M2_MC), length.out = 100)
taylor_sd_2 = sqrt(var_N1_M2_taylor)
y_curve_2 = dnorm(x_curve_2, mean = mean_N1_M2_taylor, sd = taylor_sd_2)
lines(x_curve_2, y_curve_2, col = "coral1", lwd = 2)
legend("topright", legend = c("Monte Carlo", "Aprox. Taylor"),
      fill = c("darkseagreen2", "coral1"))
```

```
----- DIAGNOSIS DE MONTE CARLO -----  
  
# La simulación de MC es el método de referencia.  
  
par(mfrow = c(2, 2))  
  
# ANÁLISIS DE NORMALIDAD  
  
# Gráfico Q-Q para P(N2|M1)  
  
qqnorm(P_N2_M1_MC, main = "Gráfico Q-Q para P(N2|M1) (MC)", pch = 20)  
qqline(P_N2_M1_MC, col = "paleturquoise3", lwd = 2)  
  
# Gráfico Q-Q para P(N1|M2)  
  
qqnorm(P_N1_M2_MC, main = "Gráfico Q-Q para P(N1|M2) (MC)", pch = 20)  
qqline(P_N1_M2_MC, col = "darkseagreen1", lwd = 2)  
  
# ANÁLISIS DE CONVERGENCIA (Media Acumulada)  
  
# Convergencia para P(N2|M1):  
  
media_acum_N2_M1_MC = cumsum(P_N2_M1_MC) / seq_along(P_N2_M1_MC)  
plot(media_acum_N2_M1_MC, type = "l",  
      col = "paleturquoise3", lwd = 2,  
      main = "Análisis Convergencia P(N2|M1) (MC)",  
      xlab = "Número de Simulaciones", ylab = "Media Acumulada")  
  
# Convergencia para P(N1|M2):  
  
media_acum_N1_M2_MC = cumsum(P_N1_M2_MC) / seq_along(P_N1_M2_MC)  
plot(media_acum_N1_M2_MC, type = "l",  
      col = "darkseagreen1", lwd = 2,  
      main = "Análisis convergencia P(N1|M2) (MC)",  
      xlab = "Número de Simulaciones", ylab = "Media Acumulada")  
  
par(mfrow = c(1, 1))  
  
# ANÁLISIS DE INDEPENDENCIA  
  
par(mfrow = c(1, 2))  
  
# ACF para P(N2|M1) y P(N1|M2):  
  
acf(P_N2_M1_MC, main = "Independencia P(N2|M1) (MC)")  
acf(P_N1_M2_MC, main = "Independencia P(N1|M2) (MC)")
```

```

par(mfrow = c(1, 1))

# TEST DURBIN - WATSON

library(lmtest)

dw_test_N2_M1_MC = dwtest(P_N2_M1_MC ~ 1)

print(dw_test_N2_M1_MC)

dw_test_N1_M2_MC = dwtest(P_N1_M2_MC ~ 1)

print(dw_test_N1_M2_MC)

```

15.4. Anexo 4: Desarrollo de las derivadas para el método de Taylor

- **Derivadas de P(M1).** Fórmula de la que se parte:

$$P(M1) = P(N1) \cdot P(M1|N1) + P(N2) \cdot P(M1|N2) + P(N3) \cdot P(M1|N3)$$

- *Derivada respecto a P(M1|N1):* tomamos como constantes a P(N1), P(N2), P(N3), P(M1|N2) y P(M1|N3).

Los términos $P(N2) \cdot P(M1|N2)$ y $P(N3) \cdot P(M1|N3)$ son constantes, por lo que su derivada es cero.

$P(N1) \cdot P(M1|N1)$ es como derivar “ $c \cdot x$ ” respecto a x , el resultado es $P(N1)$.

Así:

$$\frac{\partial P(M1)}{\partial P(M1|N1)} = P(N1)$$

- *Derivada respecto a P(M1|N2) y respecto a P(M1|N3):* mismo proceso que el anterior:

Para la derivada respecto a P(M1|N2) se consideran constantes P(N1), P(N2), P(N3), P(M1|N1) y P(M1|N3).

Para la derivada respecto a P(M1|N3) se consideran constantes P(N1), P(N2), P(N3), P(M1|N1) y P(M1|N2).

Así:

$$\frac{\partial P(M1)}{\partial P(M1|N2)} = P(N2) \quad ; \quad \frac{\partial P(M1)}{\partial P(M1|N3)} = P(N3)$$

- **Derivadas de P(N2|M1).** Fórmula de la que se parte:

$$P(N2|M1) = \frac{P(M1|N2) \cdot P(N2)}{P(M1)}$$

Ahora se tiene que usar la regla del cociente: $\left(\frac{f}{g}\right)' = \frac{f' \cdot g - f \cdot g'}{g^2}$

- *Derivada respecto a $P(M1|N1)$* : el numerador es $f = P(M1|N2) \cdot P(N2)$ y el denominador $g = P(M1)$.

Como $P(M1|N1)$ no aparece en f , su derivada es cero ($f' = 0$).

La derivada de g respecto a $P(M1|N1)$ se ha explicado anteriormente: $g' = P(N1)$.

Aplicando la regla del cociente:

$$\left(\frac{P(M1|N2) \cdot P(N2)}{P(M1)} \right)' = \frac{(0 \cdot P(M1)) - (P(M1|N2) \cdot P(N2) \cdot P(N1))}{P(M1)^2}$$

El resultado final quedaría:

$$\frac{\partial P(N2|M1)}{\partial P(M1|N1)} = - \frac{P(M1|N2) \cdot P(N2) \cdot P(N1)}{P(M1)^2}$$

- *Derivada respecto a $P(M1|N2)$* : el numerador es $f = P(M1|N2) \cdot P(N2)$ y el denominador $g = P(M1)$.

La derivada de f respecto a $P(M1|N2)$ es $f' = P(N2)$; y la de g es $g' = P(N2)$.

Aplicando de nuevo la regla del cociente, nos queda finalmente:

$$\frac{\partial P(N2|M1)}{\partial P(M1|N2)} = \frac{(P(N2) \cdot P(M1)) - (P(M1|N2) \cdot P(N2) \cdot P(N2))}{P(M1)^2}$$

- *Derivada respecto a $P(M1|N3)$* : Sigue la misma lógica que la ya explicada para la derivada respecto a $P(M1|N1)$, quedando:

$$\frac{\partial P(N2|M1)}{\partial P(M1|N3)} = - \frac{P(N2) \cdot P(M1|N2) \cdot P(N3)}{P(M1)^2}$$

- **Derivadas de $P(M2)$** . Fórmula de la que se parte:

$$P(M2) = P(N1) \cdot P(M2|N1) + P(N2) \cdot P(M2|N2) + P(N3) \cdot P(M2|N3)$$

Como $P(M2|Ni) = 1 - P(M1|Ni)$, se puede reescribir dejándola en función de las variables que nos interesan:

$$P(M2) = P(N1) \cdot (1 - P(M1|N1)) + P(N2) \cdot (1 - P(M1|N2)) + P(N3) \cdot (1 - P(M1|N3))$$

- *Derivada respecto a $P(M1|N1)$* : Nos centramos en el término que contiene a $P(M1|N1)$ (los otros dos términos de la suma se toman como constantes, por lo que su derivada es cero):

$$P(N1) \cdot (1 - P(M1|N1)) = P(N1) - P(N1) \cdot P(M1|N1)$$

La derivada de $P(N1)$ es cero por ser una constante. La derivada del segundo término es $-P(N1)$.

El resultado final, por tanto, queda:

$$\frac{\partial P(M2)}{\partial P(M1|N1)} = -P(N1)$$

- *Derivadas respecto a $P(M1|N2)$ y $P(M1|N3)$:* siguen exactamente la misma lógica que el caso anterior.

$$\frac{\partial P(M2)}{\partial P(M1|N2)} = -P(N2) \quad , \quad \frac{\partial P(M2)}{\partial P(M1|N3)} = -P(N3)$$

- **Derivadas de $P(N1|M2)$.** Fórmula de la que se parte:

$$P(N1|M2) = \frac{(1 - P(M1|N1)) \cdot P(N1)}{P(M2)}$$

De nuevo, para este caso se vuelve a hacer uso de la regla del cociente: $\left(\frac{f}{g}\right)' = \frac{f' \cdot g - f \cdot g'}{g^2}$; donde el numerador es $f = (1 - P(M1|N1)) \cdot P(N1)$ y el denominador es $g = P(M2)$.

- *Derivada respecto a $P(M1|N1)$:* El resultado de derivar f respecto a $P(M1|N1)$ es $f' = -P(N1)$; y el de derivar g respecto a la misma variable es $g' = -P(N1)$ (ya visto anteriormente).

Sustituyendo en la regla del cociente, obtenemos el resultado final:

$$\frac{\partial P(N1|M2)}{\partial P(M1|N1)} = \frac{(-P(N1)) \cdot P(M2) - ((1 - P(M1|N1)) \cdot P(N1)) \cdot (-P(N1))}{P(M2)^2}$$

- *Derivada respecto a $P(M1|N2)$:* Como esta variable no aparece en f , su derivada es cero ($f' = 0$). En el caso del denominador, como ya se vio antes, la derivada respecto a $P(M1|N2)$ es $g' = -P(N2)$.

Al aplicar la regla del cociente queda:

$$\begin{aligned} & \left(\frac{(1 - P(M1|N1)) \cdot P(N1)}{P(M2)} \right)' \\ &= \frac{(0 \cdot P(M2)) - ((1 - P(M1|N1)) \cdot P(N1)) \cdot (-P(N2))}{P(M2)^2} \end{aligned}$$

Operando un poco, tenemos el resultado final:

$$\left(\frac{\partial P(N1|M2)}{\partial P(M1|N2)} \right) = \frac{(1 - P(M1|N1)) \cdot P(N1) \cdot P(N2)}{P(M2)^2}$$

- *Derivada respecto a $P(M1|N3)$:* Repitiendo el mismo proceso anterior, tenemos que $f' = 0$ y que $g' = -P(N3)$.

Aplicando la regla del cociente y simplificando, queda:

$$\frac{\partial P(N1|M2)}{\partial P(M1|N3)} = \frac{(1 - P(M1|N1)) \cdot P(N1) \cdot P(N3)}{P(M2)^2}$$

15.5. Anexo 5: Tablas de los IC (Experimento 2)

- *Tablas de los IC (95%) para los factores de $P(N2|M1)$.*

FACTOR F_PM1N1

| Nivel | Media | 2.5% | 97.5% |
|-------|-------|------|-------|
| 0.36 | 0.33 | 0.33 | 0.33 |
| 0.4 | 0.32 | 0.32 | 0.32 |
| 0.44 | 0.31 | 0.31 | 0.31 |

Tabla 11. Tabla IC del factor F_{PM1N1} de $P(N2|M1)$ (Exp. 2)

FACTOR F_PM1N2

| Nivel | Media | 2.5% | 97.5% |
|-------|-------|------|-------|
| 0.45 | 0.30 | 0.30 | 0.30 |
| 0.5 | 0.32 | 0.32 | 0.32 |
| 0.55 | 0.33 | 0.33 | 0.33 |

Tabla 12. Tabla IC del factor F_{PM1N2} de $P(N2|M1)$ (Exp. 2)

FACTOR F_PM1N3

| Nivel | Media | 2.5% | 97.5% |
|-------|-------|------|-------|
| 0.18 | 0.32 | 0.32 | 0.32 |
| 0.2 | 0.32 | 0.32 | 0.32 |
| 0.22 | 0.31 | 0.31 | 0.31 |

Tabla 13. Tabla IC del factor F_{PM1N3} de $P(N2|M1)$ (Exp. 2)

- *Tablas de los IC (95%) para los factores de $P(N1|M2)$.*

FACTOR F_PM1N1

| Nivel | Media | 2.5% | 97.5% |
|-------|-------|------|-------|
| 0.36 | 0.50 | 0.50 | 0.50 |
| 0.4 | 0.49 | 0.49 | 0.49 |
| 0.44 | 0.48 | 0.48 | 0.48 |

Tabla 14. Tabla IC del factor F_{PM1N1} de $P(N1|M2)$ (Exp. 2)

FACTOR F_PM1N2

| Nivel | Media | 2.5% | 97.5% |
|-------|-------|------|-------|
| 0.45 | 0.48 | 0.48 | 0.48 |
| 0.5 | 0.49 | 0.49 | 0.49 |
| 0.55 | 0.49 | 0.49 | 0.50 |

Tabla 15. Tabla IC del factor F_{PM1N2} de $P(N1|M2)$ (Exp. 2)

FACTOR F_PM1N3

| Nivel | Media | 2.5% | 97.5% |
|-------|-------|------|-------|
| 0.18 | 0.49 | 0.48 | 0.49 |
| 0.2 | 0.49 | 0.49 | 0.49 |
| 0.22 | 0.49 | 0.49 | 0.49 |

Tabla 16. Tabla IC del factor F_{PM1N3} de $P(N1|M2)$ (Exp. 2)

- *Tablas de los IC (95%) para las interacciones de $P(N2|M1)$.*

INTERACCIÓN F_PM1N1 - F_PM1N2

| Nivel F_PM1N1 | Nivel F_PM1N2 | Media | 2.5% | 97.5% |
|------------------|------------------|-------|------|-------|
| 0.36 | 0.45 | 0.31 | 0.31 | 0.32 |
| 0.36 | 0.5 | 0.33 | 0.33 | 0.33 |
| 0.36 | 0.55 | 0.35 | 0.34 | 0.35 |
| 0.4 | 0.45 | 0.30 | 0.30 | 0.30 |
| 0.4 | 0.5 | 0.32 | 0.32 | 0.32 |
| 0.4 | 0.55 | 0.33 | 0.33 | 0.34 |
| 0.44 | 0.45 | 0.29 | 0.29 | 0.29 |
| 0.44 | 0.5 | 0.31 | 0.31 | 0.31 |
| 0.44 | 0.55 | 0.32 | 0.32 | 0.32 |

Tabla 17. Tabla IC de interacción F_PM1N1 - F_PM1N2 de $P(N2|M1)$ (Exp. 2)**INTERACCIÓN F_PM1N1 - F_PM1N3**

| Nivel F_PM1N1 | Nivel F_PM1N3 | Media | 2.5% | 97.5% |
|------------------|------------------|-------|------|-------|
| 0.36 | 0.18 | 0.34 | 0.33 | 0.34 |
| 0.36 | 0.2 | 0.33 | 0.33 | 0.33 |
| 0.36 | 0.22 | 0.32 | 0.32 | 0.33 |
| 0.4 | 0.18 | 0.32 | 0.32 | 0.32 |
| 0.4 | 0.2 | 0.32 | 0.32 | 0.32 |
| 0.4 | 0.22 | 0.31 | 0.31 | 0.32 |
| 0.44 | 0.18 | 0.31 | 0.31 | 0.31 |
| 0.44 | 0.2 | 0.31 | 0.31 | 0.31 |
| 0.44 | 0.22 | 0.30 | 0.30 | 0.30 |

Tabla 18. Tabla IC de interacción F_PM1N1 - F_PM1N3 de $P(N2|M1)$ (Exp. 2)



POLITÉCNICA

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES
UNIVERSIDAD POLITÉCNICA DE MADRID

José Gutiérrez Abascal, 2. 28006 Madrid

Tel.: 91 336 3060

info.industriales@upm.es

www.industriales.upm.es