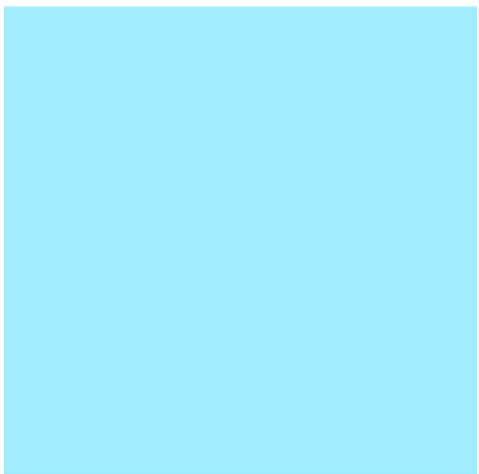


Método para el cálculo de potencia biológica a partir de datos de inmunoensayo según el modelo paralelo-lineal. Aplicación y desarrollo de una librería en R



Universitat Oberta
de Catalunya



UNIVERSITAT DE
BARCELONA

Rocío Lorenzo López

Estudios clínicos y epidemiológicos

Máster en Bioinformática y Bioestadística

Nombre de la tutora de TF:

Nuria Pérez Álvarez

Nombre de la PRA:

Laia Subirats Maté

13 de Enero de 2024



Esta obra esta sujeta a una licencia de Reconocimiento
<https://creativecommons.org/licenses/by-nc/3.0/es/>

Agradecimientos

En primer lugar, gracias a Nuria Pérez Álvarez, por su inestimable ayuda para sacar adelante este proyecto.

Agradecer en particular a Martín Montes Hernández; no solo por su valiosa orientación sino, especialmente, por haberme propuesto este desafío que ha resultado en una gran fuente de motivación.

Gracias a Leticia y Esperanza, por acogerme en su equipo hace ya unos cuantos años. A Jose, por compartir su saber tan desinteresadamente y por nuestras chaquetillas imaginarias de *stability team*. En general, a mis compañeros de Pharma Dev., por hacer más llevaderas las horas. En especial, a mis *compis* queridos: Samu, Rachel, Bego, Zori, Nuria, Kitty y Male; por la generosidad con su tiempo y por seguir bailando juntos aunque la vida nos lleve por diferentes caminos. A Dani, por todos esos ratos batallando sobre ensayos y demás dramas.

A mi otra mitad, *ó meu roxonciño*, a mi sevillana preferida, a mi querida *hufflepuff*, a mi vasquiño y a las habitantes de mi isla bonita. Por perdonar mis ausencias, por ser y estar incondicionalmente. A mi familia madrileña, por ser siempre casa. A mis padres, por vuestro apoyo; por tener siempre una sonrisa para mí, aún en los peores días. A todos vosotros, por vuestra paciencia; una vez más, mi humilde forma de devolver el tiempo que no os he podido dedicar. A Javi, por ayudarme a entender como “piensan” las máquinas, por salvarme innumerables veces del “valle de la desolación” y, sobre todo, por ser mi paz.

A todos aquellos que no he nombrado y con pequeños detalles, ayudados por el efecto mariposa, me han traído hasta aquí.

A Susa, Nati y J. Antonio; por ser mi estrella polar.

Ficha Del Trabajo Final

Título del trabajo:	Método para el cálculo de potencia biológica a partir de datos de inmunoensayo según el modelo paralelo-lineal. Aplicación y desarrollo de una librería en R
Nombre del autor/a:	Rocío Lorenzo López
Nombre de la tutora de TF:	Nuria Pérez Álvarez
Nombre de la PRA:	Laia Subirats Maté
Fecha de entrega:	13 de Enero de 2024
Titulación o programa:	Máster en Bioinformática y Bioestadística
Área del trabajo final:	Estudios clínicos y epidemiológicos
Idioma del trabajo:	Castellano
Palabras clave:	ParallelLineModel, EstimatedPotency, Immunoassay, Statistics

Resumen del trabajo

La medida de la actividad biológica de un compuesto o preparación es uno de los principales objetivos cuando se realiza un bioensayo, especialmente dentro del área de la farmacología. El enfoque más ampliamente utilizado para la determinación de potencia biológica es la aplicación del modelo paralelo-lineal. En este texto se desarrolla una solución para el cálculo de potencia aplicando el modelo paralelo-lineal.

Además, dentro de los bioensayos, un caso de especial atención son los inmunoensayos. Este tipo de ensayo genera curvas dosis-respuestas amplias, con aspecto sigmoidal, que no coinciden con un ajuste lineal, requisito necesario para emplear el modelo propuesto. La solución planteada ha sido diseñada especialmente para el caso de datos de inmunoensayo en el que, previamente, habrá que definir un rango de trabajo, dentro de la curva dosis-respuesta, que se ajuste a un modelo lineal.

En el presente documento se pretende dar una solución completa para el análisis de datos de inmunoensayos. De tal manera que, no solo se ha tenido en cuenta la fase de análisis de datos en sí misma, si no todo el proceso en conjunto. Se propone un análisis de forma secuencial, que va desde el diseño de un formato para la entrada de los datos obtenidos en el ensayo. Seguido del análisis en sí mismo, que se compone tanto del desarrollo de una librería **R** como de un script en RMarkdown. Éste último servirá, además, para la generación de un informe final donde los resultados obtenidos se puedan observar e interpretar fácilmente.

Abstract

Measuring the biological activity of a compound or preparation is one of the main objectives when performing a bioassay, especially within the area of pharmacology. The most widely used approach to the determination of biological potency is the application of the parallel-line model. In this text, a solution is developed for the calculation of potency by applying the parallel-line model.

In addition, within bioassays, a case of special attention are immunoassays. This type of assay generates wide dose-response curves, with sigmoidal appearance, that do not comply with a linear fit, which is a necessary requirement to use the proposed model. The proposed solution has been specially designed for the case of immunoassay data in which, previously, it will be necessary to define a working range, within the dose-response curve, that fits a linear model.

This paper aims to provide a complete solution for the analysis of immunoassay data. In this way, not only the data analysis phase itself has been taken into account, but the entire process as a whole. A sequential analysis is proposed, ranging from the design of a format for the input of the data obtained in the assay. Followed by the analysis itself, which is made up of both the development of a **R** library and a script in RMarkdown. The latter will also be used to generate a final report where the results obtained can be easily observed and interpreted.

Índice

1. Introducción	1
1.1. Contexto y justificación del trabajo	2
1.2. Objetivos del trabajo	3
1.3. Impacto en sostenibilidad, ético-social y de diversidad	4
1.4. Enfoque y método seguido	5
1.5. Planificación del trabajo	6
1.6. Breve resumen de productos obtenidos	9
1.7. Breve descripción de los otros capítulos de la memoria	10
2. Estado del arte	11
2.1. Inmunoensayo	11
2.1.1. ELISA	11
2.1.2. Análisis estadístico de datos de inmunoensayo	15
2.2. Curva dosis-respuesta	17
2.3. Determinación de la potencia biológica estimada	18
2.3.1. Modelo paralelo lineal	19
2.3.2. Modelo logístico de cuatro parámetros	21

2.4. Alergia	22
3. Materiales y Métodos	23
3.1. Obtención de los datos	23
3.1.1. ELISA IgE de Competición	23
3.2. Análisis estadístico y determinación de potencia. Modelo paralelo-lineal	25
3.2.1. Filtrado de datos	25
3.2.2. Selección del rango de dosis para ajustar el modelo	26
3.2.3. Determinación de potencia biológica (PLM)	26
3.3. Disposiciones legales y normas aplicadas	30
3.3.1. Datos de ejemplo para validación	30
3.4. Software empleado	31
3.4.1. Software R	31
4. Resultados y Discusión	33
4.1. Plantilla <i>.xlsx</i> para registro de datos obtenidos.	33
4.2. Librería <i>plm.crd</i>	36
4.2.1. Funciones lectura de datos en R	36
4.2.2. Funciones para tratamiento y filtrado de datos	38
4.2.3. Funciones Modelo paralelo-lineal	40
4.2.4. Funciones cálculo de potencia	43
4.3. Documento RMarkdown para presentación de resultados	45
4.3.1. <i>Script</i> RMarkdown	45
4.3.2. Informe de resultados	46

4.4. Validación de la solución propuesta y discusión final	47
5. Conclusiones y trabajos futuros	49
5.1. Conclusiones	49
5.2. Limitaciones del estudio	50
5.3. Líneas de futuro	50
5.4. Seguimiento de la planificación	51
6. Glosario	52
6.1. Abreviaturas	52
6.2. Definiciones	54
7. Bibliografía	55
A. Anexo: Datos de ejemplo para validación	59
B. Anexo: Plantilla entrada de datos obtenidos	64
C. Anexo: Código R librería plm.crd	66
C.1. Función <i>load.info</i>	66
C.2. Función <i>load.data</i>	67
C.3. Función <i>assay.data</i>	68
C.3.1. Función auxiliar <i>square</i>	68
C.4. Función <i>filter.data</i>	69
C.4.1. Función auxiliar <i>cv.calc</i>	70
C.4.2. Función auxiliar <i>pos.max.diff</i>	70

C.5. Función <i>pl.model</i>	71
C.6. Función <i>best.model</i>	72
C.6.1. Función auxiliar <i>select.range</i>	73
C.6.2. Función auxiliar <i>check.model</i>	74
C.7. Función <i>com.slope</i>	75
C.8. Función <i>rel.pot</i>	76
C.9. Función <i>est.pot</i>	77
D. Anexo: Código RMarkdown	78
E. Anexo: Informe de resultados generado	87

Índice de figuras

1.1. Diagrama de Gantt	7
2.1. Tipos de ELISA	13
2.2. Curva Log Dosis vs Respuesta Esperada	17
2.3. Modelo paralelo-lineal (ensayo 4x3)	19
2.4. Modelo logístico 4 parámetros (ensayo 10x3)	21

Índice de tablas

3.1. Fórmulas modelo paralelo lineal	27
3.2. Tabla ANOVA	28
3.3. Resultados de potencia e intervalos de confianza 95 %	30
4.1. Resultados de potencia según el método seguido	48
A.1. Datos ejemplo para validación	59
A.2. Medias de las ln(DOs) por muestra y nivel	61
A.3. Tabla ANOVA con los datos de ejemplo	62

Capítulo 1

Introducción

Un inmunoensayo es una técnica ampliamente empleada en laboratorio para detectar y/o medir la cantidad de una molécula específica en una muestra, generalmente de carácter biológico [1]. En el ámbito de aplicación en el que se desarrolla este trabajo, se emplean diferentes tipos de inmunoensayos para medir la potencia biológica de diferentes productos farmacéuticos.

La potencia es una medida de la actividad biológica. Declarar y realizar un control de la potencia o actividad biológica es un requisito obligatorio en todos los productos farmacéuticos [2]. En general, la potencia de una muestra desconocida se determina en función a un estándar de referencia conocido, que se analiza en las mismas condiciones [3]. Se analizan varias concentraciones o dosis, iguales para muestra y estándar, que generarán dos curvas dosis-respuesta, que idealmente serán similares (paralelas). En caso de que se cumpla lo anterior, el desplazamiento horizontal de la curva de la muestra en relación con la curva del estándar permitirá el cálculo de la potencia estimada. A este método de determinación de la potencia biológica se le conoce como modelo paralelo lineal (PLM, por sus siglas en inglés) [4].

El presente trabajo tiene como objetivo principal desarrollar una solución, pensando en todos los pasos del proceso de análisis una vez se han obtenido los datos, que permita aplicar el modelo paralelo lineal para el cálculo de potencia biológica a partir de datos de inmunoensayo.

1.1. Contexto y justificación del trabajo

En el ámbito de un laboratorio, tanto si se trata de un laboratorio de control de calidad, como si se refiere a un laboratorio de investigación; es importante optimizar todas las etapas del proceso de obtención de datos de ensayos, con el fin de reducir tiempos y minimizar desviaciones entre ensayos. Para ello se propone realizar una solución en lenguaje de programación **R** que realice todo el análisis estadístico y obtención de resultados, de los inmunoensayos realizados, de la forma más automatizada posible.

Además, es interesante que los datos brutos sean manipulados por los operadores lo mínimo imprescindible, con el fin de evitar sesgos involuntarios. Por ello, se propone el desarrollo de una solución que seleccione de forma automática el rango de trabajo dentro de las curvas dosis-respuesta obtenidas en el ensayo.

Existe una librería de funciones en **R**, llamada *pla*, en la que se implementa una solución al problema planteado, pero actualmente está descatalogado del repositorio de la Red integral de archivos de R (CRAN) y no está disponible para las versiones más recientes del software. La información que figura en el propio repositorio, con fecha 9 de abril de 2020, “el paquete fue archivado del repositorio CRAN porque el autor no corrigió los problemas que se le señalaron a pesar de haber recibido recordatorios” [5].

1.2. Objetivos del trabajo

Con la finalidad de abordar la determinación de potencia biológica a partir de datos de inmunoensayo, este proyecto se propone lograr varios objetivos. A continuación se detallan los objetivos generales y específicos que se pretenden alcanzar.

Objetivos generales:

- Desarrollar una solución para el cálculo de potencia biológica.
- Presentar un método de presentación de los resultados del análisis.
- Validar si los resultados obtenidos satisfacen los requerimientos de la norma europea.

Objetivos específicos:

- Desarrollar un análisis estadístico para datos de inmunoensayos.
- Exponer una función que realice el cálculo de potencia biológica.
- Elaborar una función/ algoritmo que automatice la elección de rango (cuantos puntos de la recta obtenida se emplearán en el análisis).
- Elaborar un método de entrada de datos que resulte amigable a usuarios sin formación previa en programación ni software específico.
- Obtener un reporte de resultados fácilmente comprensible.
- Validar el proyecto con la legislación vigente en cuanto a fabricación y control de medicamentos en Europa.

1.3. Impacto en sostenibilidad, ético-social y de diversidad

A priori, es difícil establecer el impacto en estas tres dimensiones de un trabajo técnico donde el producto obtenido es, en líneas generales, una herramienta para análisis de datos. Esto es debido, en gran medida, a que la finalidad que pueda llegar a tener depende en gran medida de los propósitos para los que, una vez desarrollado, se emplee dicha solución aportada. Aún así, se tratará de especificar en que objetivos de desarrollo sostenible propuestos por la Organización de las Naciones Unidas (ONU) puede encuadrarse.

En materia de sostenibilidad, este trabajo ofrece una mejora en la metodología para la obtención de resultados y reproducibilidad de los mismos, con un rango de aplicación muy amplio dentro de los estudios biológicos, como puede ser: la investigación de nuevos fármacos, el seguimiento y control de los fármacos existentes, la detección y caracterización de enfermedades infecciosas, entre otros. En este sentido, la solución presentada incide especialmente en una mejora en la gestión de los recursos disponibles, puesto que ofrece una solución más rápida para el análisis de los datos y evita la posterior generación de informes (lo genera automáticamente a la vez que realiza el análisis). Además, la selección automática del rango de trabajo para el posterior análisis y la generación automática del informe de resultados supone un incremento en la fiabilidad de los resultados obtenidos, dado que elimina posibles sesgos involuntarios; tanto a la hora del análisis, como a la hora de extraer conclusiones.

En relación con el impacto en aspectos ético-sociales, tal y como se ha mencionado en el párrafo anterior, este trabajo supone una mejora en gestión de recursos. Esto, a medio-largo plazo puede suponer una mejora en los recursos destinados a investigación y desarrollo de nuevos medicamentos. Por consiguiente, supone un avance en los recursos destinados a la salud de las personas.

Las mejoras que proporcionan esta propuesta, a su vez, pueden incidir en materia de diversidad si se considera una aplicación en el ámbito biosanitario, como puede ser el desarrollo de nuevos fármacos. En este sentido, la solución propuesta, impacta en materia de diversidad e inclusión de ensayos clínicos, ya que en ocasiones la falta de recursos ocasiona una falta de representación de la diversidad de la población mundial.

1.4. Enfoque y método seguido

Para la ejecución de este proyecto se pueden seguir varias estrategias. Desde el uso de diferentes tipos de software comercial para el análisis estadístico, la elección del formato de salida de los resultados o el formato de entrada de los datos.

Para el desarrollo de este proyecto se empleará el lenguaje de programación **R**. El formato de entrada de los datos será preferiblemente Excel, desde el cual se llamará a **R** para que ejecute el código propuesto. Esto se ha pensado de la manera anterior teniendo en cuenta que un usuario, aún sin formación específica en estadística ni lenguajes de programación, pueda aprovechar los beneficios de esta solución.

Para la salida de datos se empleará preferiblemente un formato de documentos portátiles (pdf), generado con RMarkdown o similar. Para validar el script realizado se emplearán los datos de ejemplo que figuran en la monografía *European Pharmacopeia* (Ph.Eur.) para cumplir con los estándares requeridos.

A lo largo del proyecto se irán definiendo con más detalle los retos a los que se pretende dar solución y explicando, más en detalle, porqué se ha elegido una metodología en concreto.

1.5. Planificación del trabajo

El presente proyecto se ha llevado a cabo siguiendo la siguiente planificación.

Tareas:

A continuación se detalla un listado de las tareas principales que es necesario realizar para la ejecución exitosa del trabajo propuesto:

- Estudio del modelo paralelo-lineal.
- Desarrollo del análisis estadístico según el modelo paralelo-lineal.
- Creación del algoritmo de decisión de rango empleado.
- Desarrollo función para cálculo de potencia.
- Desarrollo función para filtrado de datos.
- Construcción librería funciones para implementar el modelo paralelo-lineal.
- Generación de plantilla Excel y enlace con **R**.
- Diseño de documento de salida de datos.
- Validación de los resultados.
- Desarrollo de funciones para modelo logístico de cuatro parámetros (opcional según avance del proyecto).
- Redacción del informe final.

Planificación:

La planificación temporal del proyecto se detalla en el siguiente diagrama de Gantt (Figura 1.1).

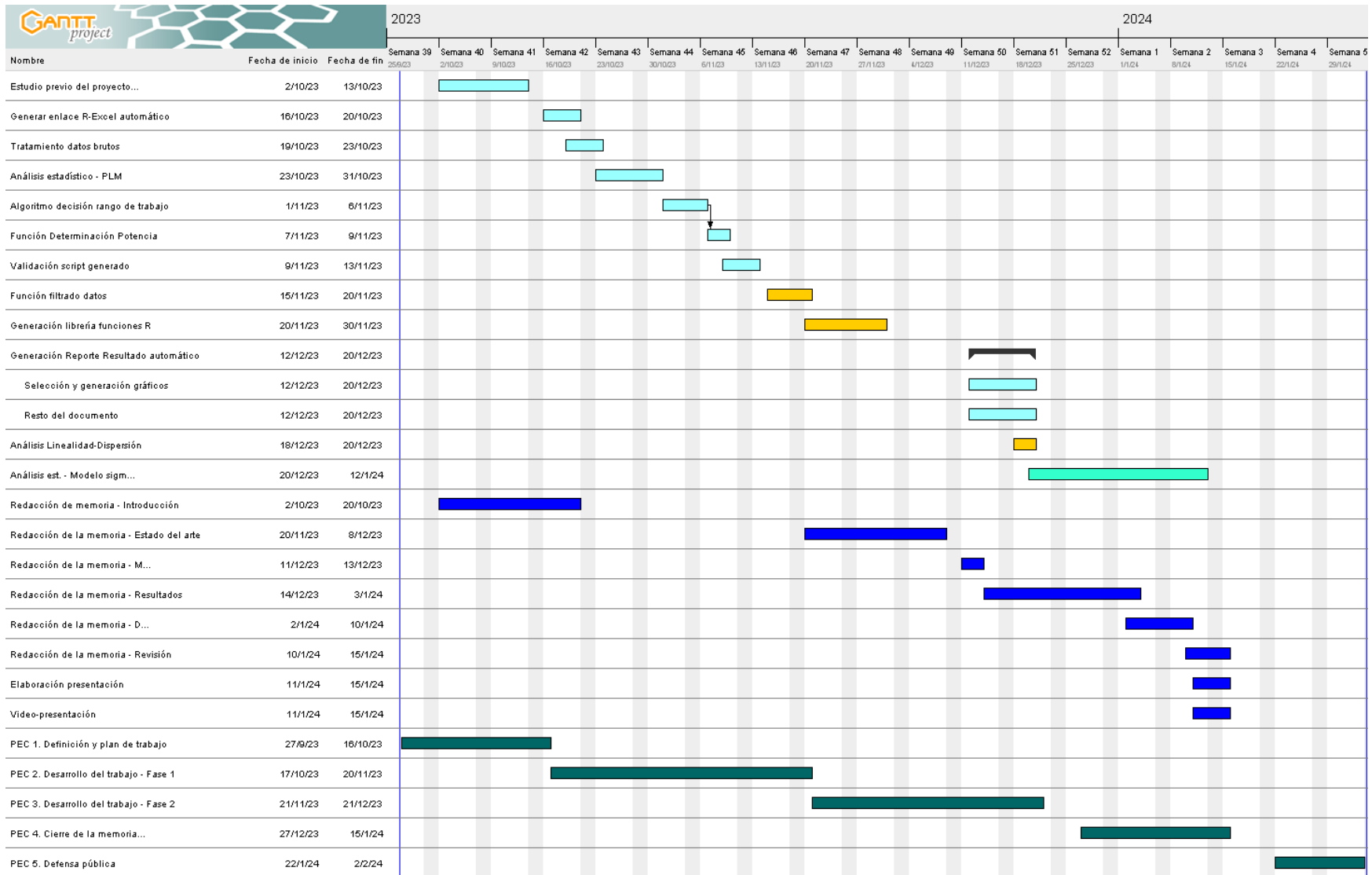


Figura 1.1: Diagrama de Gantt con la planificación del proyecto para la consecución de los objetivos propuestos. Realizado con Gantt-Project. En **azul claro** las tareas propias para la consecución de los objetivos. En **naranja** nuevas tareas que iban surgiendo según avanzaba el proyecto. En **verde claro** tareas opcionales. En **azul oscuro** tareas de redacción. Y por último, en **verde oscuro** tareas para actividades de evaluación

Hitos:

Seguidamente se detallan los hitos principales del trabajo propuesto. Se muestran cronológicamente de forma que permiten explicar la planificación anterior y, a su vez, se relacionan con las entregas parciales existentes:

- Entrega plan de trabajo.
- Generación funciones para análisis básico de datos.
- Automatizar la elección del rango de la curva.
- Entrega informe de seguimiento 1.
- Elaboración librería de funciones.
- Inclusión de nuevas funcionalidades según las necesidades observadas.
- Redacción de la memoria.
- Entrega informe de seguimiento 2.
- Validación de la solución.
- Redacción resultados y conclusiones.
- Entrega de la memoria.
- Elaboración de la presentación.
- Defensa pública.

1.6. Breve resumen de productos obtenidos

Como resultado del trabajo realizado se ha obtenido una solución integral para la determinación de potencia biológica a partir de datos de inmunoensayo empleando el modelo paralelo-lineal.

Esto implica: desde el diseño de una plantilla para el volcado de los datos obtenidos en el ensayo, la generación de funciones y rutinas adecuadas para llevar a cabo el análisis propuesto (desde el software **R**) y, por último; el diseño de un documento, tipo informe, que resuma los resultados obtenidos de una forma clara y concisa.

En este sentido, además de la memoria y presentación elaboradas para documentar el estudio realizado; se han obtenido los siguientes productos, listados tal y como se pretende su uso:

- Plantilla (formato .xlsx) de entrada de datos “brutos”.
- Librería **R** de análisis estadístico y cálculo de potencia biológica.
- *Script* RMarkdown para la ejecución del análisis.
- Reporte (formato .pdf) de los resultados obtenidos.

1.7. Breve descripción de los otros capítulos de la memoria

En el segundo capítulo de esta memoria se hace una reseña de los puntos importantes que ponen en contexto, y explican, el marco teórico en el que se desarrolla el trabajo realizado; desde qué es un inmunoensayo y como se realiza, hasta presentar el modelo paralelo-lineal y la determinación de la potencia con este modelo.

Posteriormente, se describen los materiales y métodos. Se explica, de forma resumida, la fase previa de obtención de datos, y de forma más extensa se explican los métodos que ilustran la parte técnica de este proyecto.

Seguidamente se exponen los resultados obtenidos, se realiza una breve discusión sobre la solución presentada y se muestran las conclusiones que se extraen del trabajo realizado.

En los apéndices posteriores se puede encontrar la información (código, plantillas,...) que forman parte del resultado de este trabajo; pero que puede resultar demasiado extensa y, en ocasiones, molesta para la fácil comprensión del texto.

Capítulo 2

Estado del arte

2.1. Inmunoensayo

Las primeras publicaciones que se refieren a un sistema mediado por proteínas naturales datan de 1960, con Rosalyn Yalow y Solomon Berson como autores. Describen una nueva técnica conocida como radioinmunoensayo (RIA) para medir la insulina plasmática endógena [6], [7]. Descubrimiento que otorgaría el Premio Nobel en Medicina en el año 1977 a Rosalyn Yalow [7].

Un inmunoensayo es una técnica ampliamente empleada en laboratorio. Se realiza principalmente para detectar y/o medir la cantidad de una molécula específica en una muestra, generalmente de carácter biológico [3]. Hay diferentes tipos de inmunoensayos y diferentes formas de clasificarlos, pero en líneas generales, el principio que rige un inmunoensayo es la reacción antígeno-anticuerpo [8].

De forma general, consiste en dos pasos: la reacción y la detección de la misma [9]. El primer paso de reacción es la unión de un anticuerpo con un antígeno; generalmente este antígeno es la molécula de interés, aunque en algunos tipos de inmunoensayo lo que se precisa medir es el anticuerpo. En la segunda fase de detección; el antígeno (o anticuerpo), que deberá estar marcado (unido a una molécula detectable por fluorescencia, colorimetría o radioactividad) se mide en un equipo adecuado según la naturaleza del marcaje [9].

2.1.1. ELISA

Dentro de las técnicas que engloban los inmunoensayos es de nuestro interés el ensayo por inmunoabsorción ligado a enzimas (ELISA, por sus siglas en inglés). Esta técnica se desarrolló al mismo tiempo por dos grupos independientes en el año 1971 [10]. De una parte, por Peter Perlmann y Eva Engvall desde la Universidad de Estocolmo; que emplearon el método ELISA

para cuantificar la cantidad de Inmunoglobulina G (IgG) en suero de conejo. [6] [7]. Y de otra parte, Anton Schuurs y Bauke van Weemen, por medio de un inmunoensayo enzimático (EIA) lograron cuantificar las cantidades de gonadotropina coriónica humana en la orina [6], [10].

Desde sus primeras aplicaciones, se ha desarrollado un amplio rango de ELISA específicos para innumerables propósitos [1]. Las características que lo hacen ventajoso son, en gran medida, la gran especificidad y sensibilidad, unido a un procedimiento simple y su capacidad para cuantificar la molécula de interés [11] [12]. Hoy en día se emplea en una gran parte de los laboratorios a lo largo del mundo y con distintos propósitos; se emplea en diferentes disciplinas que van desde la neurociencia o la farmacología [1] [12].

De manera similar al esquema general de un inmunoensayo; el procedimiento regular de un ELISA incluye varios pasos de incubación donde se utilizan diferentes antígenos, anticuerpos y enzimas como reactivos, que se van uniendo secuencialmente a una fase sólida. Estas fases de formación de uniones entre moléculas, se intercalan con ciclos de lavado donde se eliminan los excedentes de reactivos libres (no unidos) [11], [12], [13]. Todo ello para detectar y, en ocasiones, cuantificar una sustancia específica en una muestra; en la mayoría de los casos un antígeno [10].

Hay diferentes tipos de ELISA y diferentes formas de clasificarlos. Según la literatura consultada la clasificación puede variar ligeramente y no ser exacta a la que sigue. En este caso se ha considerado la propuesta por John M Walker y JR Crowther en su texto *“The ELISA guidebook”*. Atendiendo a la secuencia de fases de incubación y anticuerpos empleados se pueden diferenciar: directo, indirecto y sándwich [12]. En la figura 2.1 se puede ver el esquema de cada uno de los tipos de ELISA mencionados.

- ELISA Directo, figura 2.1(a): es el más sencillo. En él, el analito (generalmente un antígeno) se inmoviliza en la fase sólida por adsorción pasiva. En un segundo paso, un anticuerpo marcado se une al antígeno. Posteriormente, se agrega un sustrato que reaccione con el enzima de marcaje empleado, que cambiará de estado y desarrollará color [10], [12], [13].
- ELISA Indirecto, figura 2.1(b): de la misma forma que el caso anterior; el antígeno se inmoviliza en la fase sólida y en una segunda fase se une con un anticuerpo (en este caso sin marcar). El sistema antígeno/anticuerpo formado se detecta en una tercera fase con un anticuerpo anti-especie marcado. Este segundo anticuerpo será el que reaccione con el sustrato [14]. Este tipo de ELISA es más sensible que el anterior, pero también tiene más riesgo de reactividad cruzada [10], [11].
- ELISA Sándwich (o ELISA de captura): en este caso lo que se inmoviliza en la fase sólida es un anticuerpo. Posteriormente, este anticuerpo se incubará con un antígeno (la molécula de interés). El antígeno capturado, se incubará después con otro anticuerpo [10], [11]. Este segundo anticuerpo puede estar marcado para reaccionar con el sistema sustrato/cromógeno (figura 2.1(c)); o bien, unirse a un segundo anticuerpo anti-especie marcado (figura 2.1(d)). Diferenciando así entre ELISA Sándwich directo o indirecto [12]. Para que este tipo de ELISA sea factible, es necesario que los antígenos tengan dos lugares

antigénicos o epítomos, ya que es necesario que se unan primero los anticuerpos de captura y después los de detección [3] [12].

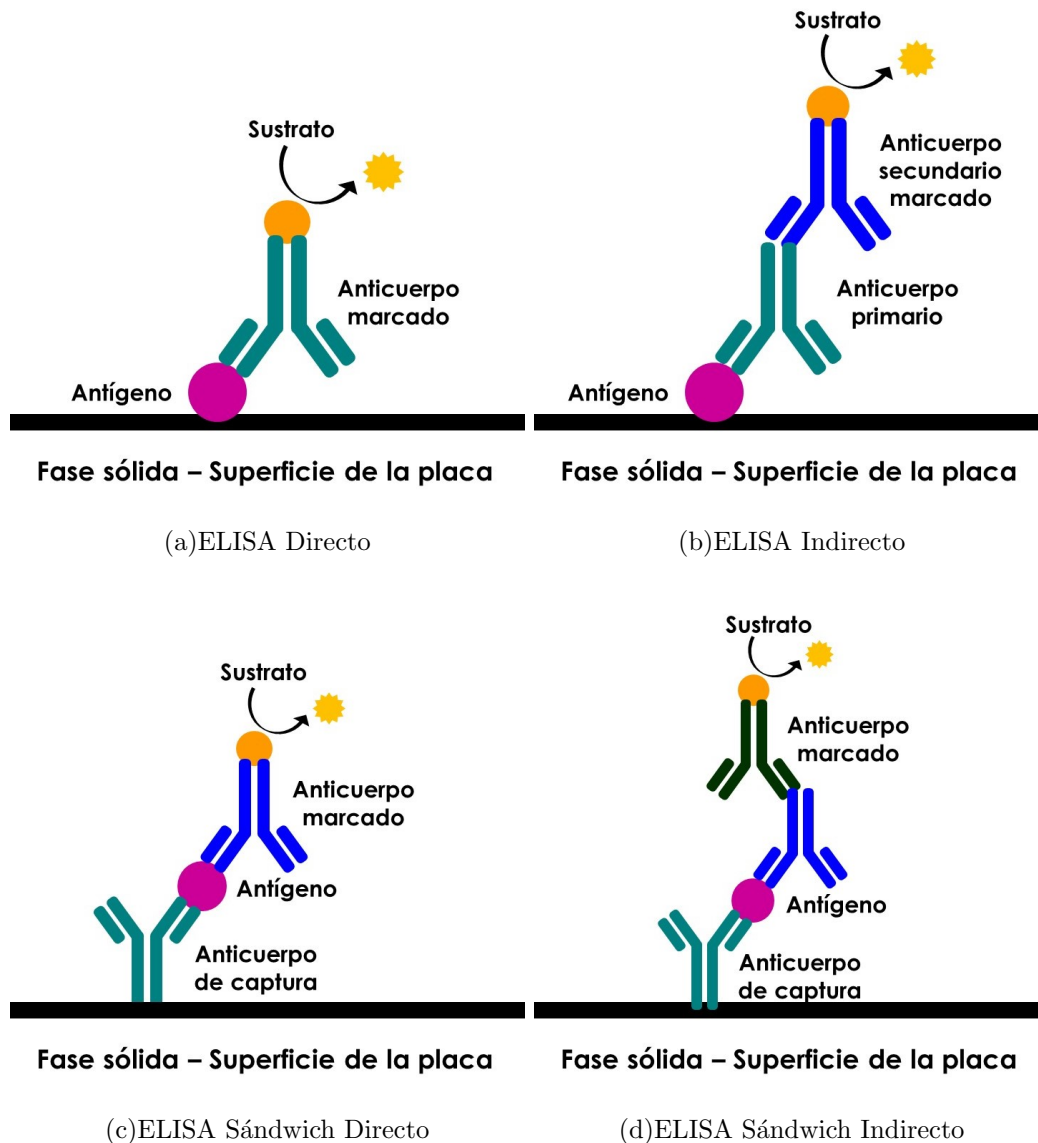


Figura 2.1: Tipos de ELISA

Dentro de las diferentes metodologías para cuantificar sustancias con un sistema ELISA surge otra clasificación, donde se diferencia entre ELISA de competición y ELISA de inhibición. Los tres tipos de ELISA mencionados anteriormente son la base de estos dos sistemas [12].

En los ELISA de competición, en alguna de las fases de incubación un anticuerpo sin marcar compete con un anticuerpo marcado por un número limitado de sitios de unión [3]. De tal forma que se medirá la cantidad de anticuerpo marcado que es capaz de unirse al sistema; y esto sucederá en función de la cantidad de anticuerpo sin marcar que contenga la mezcla [6] [11]. Este sistema se puede adaptar a todos los tipos de ELISA anteriormente mencionados. Es

importante tener en cuenta que, para que el sistema funcione, es necesario pre-titular uno de los dos ligandos, generalmente se titula la concentración del ligando sin marcar, y se emplea una concentración fija de ligando marcado; para que resulte en una respuesta gradual en función de la proporción de las dos sustancias en la fase de competición [3], [10], [12].

El sistema de ELISA Inhibición, funciona de manera similar, pero aquí sucede en dos fases, en una fase previa se inhibe o bloquea, con un anticuerpo, la posterior unión del anticuerpo marcado. Como en el caso anterior se puede adaptar a todos los tipos de ELISA anteriormente mencionados, y es necesario pre-titular una de las dos fases para generar un gradiente en la respuesta [12].

Teniendo en cuenta todo lo anterior y entrando un poco más en detalle, el procedimiento general de un ensayo ELISA Directo es el que sigue:

1. Adsorción de antígeno a la fase sólida. Habitualmente, la fase sólida empleada es una placa de pocillos de microtitulación, al que se une un antígeno por un proceso denominado adsorción pasiva. A las placas de ensayo con antígeno inmovilizado se les conoce comúnmente como placas tapizadas o sensibilizadas [12], [13].
2. Adicción de anticuerpo marcado. El anticuerpo se unirá al antígeno de forma específica formando un complejo. Además, este anticuerpo se encuentra unido a algún enzima (a este se le conoce como anticuerpo marcado). En cuanto a la enzima empleada para el marcaje, suelen emplearse moléculas detectables como fosfatasa alcalina (AP), peroxidasa de rábano picante (HRP) o biotina [11]. Generalmente la molécula de preferencia es HRP, debido a su límite de detección más bajo [13].
3. Adicción de un sistema de detección. En un siguiente paso, se añadirá un sustrato que reaccione con el enzima de marcaje empleada y que desarrollará algún cambio óptico (por ejemplo, color) en la medida en que tenga complejos con los que reaccionar [11].
4. Lectura del ensayo en un equipo adecuado. Dependerá de la naturaleza del marcaje; en el caso de desarrollar color se leerá en un espectrofotómetro. Este cambio óptico mencionado en el punto anterior es cuantificable y proporcional a la cantidad de analito de interés, por tanto, es el que permite medir la cantidad de analito de forma indirecta [10], [11].
5. Análisis estadístico de los resultados. Este punto es el de interés en el presente trabajo y por tanto, se explicará mas en detalle a lo largo del documento.

Cada uno de los pasos anteriores necesita de un tiempo y unas condiciones de incubación específicas para que se formen los complejos antígeno/anticuerpo; estas condiciones variarán en función de múltiples factores [6]. Entre cada período de incubación se eliminan los excedentes de moléculas libres (aquellos que no se encuentran formando complejos de unión) mediante uno o varios ciclos de lavado [12].

Para una mayor facilidad en la redacción de este texto, hasta este momento; se habla de un antígeno (como molécula de la cual interesa conocer su presencia o cantidad) y de anticuerpos

marcados. Es necesario aclarar que, para propósitos específicos, es posible que este esquema general se modifique para obtener resultados diferentes: puede emplearse un antígeno marcado en vez de un anticuerpo, en otros casos la molécula de interés puede ser un anticuerpo.

2.1.2. Análisis estadístico de datos de inmunoensayo

Cuando se habla de análisis estadístico de datos de inmunoensayo se puede referir a cosas muy diferentes. Se puede realizar un análisis estadístico sobre el propio ensayo, estudiando parámetros como su precisión, sensibilidad y exactitud para, por ejemplo, determinar su robustez [12]. De igual manera, se puede realizar un análisis estadístico sobre los datos generados en un ensayo, para analizar resultados de muestras específicas [12]; por ejemplo si una muestra es positiva o negativa frente a un test específico o, como se tratará más adelante en este texto, para determinar la potencia biológica relativa de una muestra concreta. Dentro del análisis de datos generados en un inmunoensayo también cabe distinguir enfoques diferentes, como los formulados en los ejemplos anteriores, en función de la finalidad de un inmunoensayo.

La finalidad puede ser obtener una respuesta binaria, es decir, identificar muestras o individuos como positivos o negativos frente a la presencia de un analito en concreto [15]. En este caso se trataría de determinar un umbral positivo/negativo (valor de corte o *cutoff*). Para ello, lo habitual es trabajar con muestras previamente conocidas y determinar la capacidad del inmunoensayo de evitar Falsos Negativos y Falsos Positivos, es decir; estudiar su sensibilidad y especificidad, con la intención de fijar el valor de corte. Desde este punto de vista, las curvas de Característica Operativa del Receptor (ROC) puede ser un enfoque interesante [12].

Por otra parte, la finalidad de un inmunoensayo puede ser la determinación de la dosis media efectiva, conocido como ED_{50} . En este caso, es necesario generar una curva dosis-respuesta, que se refiere a la relación entre la dosis de un analito de interés y la respuesta medida por el ensayo [16]. En un inmunoensayo, una curva dosis-respuesta se genera cuando se analizan diferentes dosis del analito de interés (generalmente varias diluciones con un ratio de dilución constante entre una dosis y la siguiente). Cuando se representa gráficamente la respuesta frente al logaritmo de la dosis suele generar una curva suave, simétrica y sigmoidea (en forma de S) [17] [18]. Para el análisis de la curva dosis-respuesta se han utilizado dos enfoques: la distribución acumulativa gaussiana (análisis probit) y los modelos logísticos [17].

Cuando la finalidad de un inmunoensayo es la determinación de actividad, o potencia, biológica se habla de inmunoensayos cuantitativos [15] y el análisis estadístico se hace más necesario y, a su vez, puede resultar más complejo. El estadístico David John Finney presentó una formulación general de métodos estadísticos para la estimación de la bioactividad en bioensayos indirectos [19]. La técnica desarrollada por Finney trataba de comparar la potencia de una preparación en función de un estándar conocido [18], para ello empleó el modelo paralelo lineal basado en regresión lineal. Otra estrategia, más compleja, es emplear una versión extendida: el modelo logístico de cuatro parámetros [4].

Dicho todo lo anterior, es fácil deducir, que es necesario establecer cuál será el análisis estadístico con el que se obtendrán los datos reales de cada ensayo cuantitativo particular para, posteriormente, poder realizar un análisis estadístico del propio ensayo y evaluar los diferentes parámetros de sensibilidad, exactitud, etc. Además, puesto que una prueba estadística es una medida de la probabilidad de un evento, los datos obtenidos serán más veraces cuanto más robusto sea el cálculo de dicha probabilidad [12].

2.2. Curva dosis-respuesta

Como se ha indicado anteriormente en este texto, la curva dosis-respuesta se refiere a la relación entre las diferentes dosis ensayadas y las respuestas generadas en un inmunoensayo (u otro ensayo biológico) [18]. Generalmente, la representación gráfica de la respuesta obtenida frente a la dosis, o el logaritmo de la dosis, generará una curva sigmoide característica [17], tal y como se puede observar en la Figura 2.2.

En este sistema se cumple que: si la respuesta es igual al valor esperado de la misma, existe una relación funcional entre la respuesta y la dosis, que viene dado por la ecuación 2.1 [16], [18].

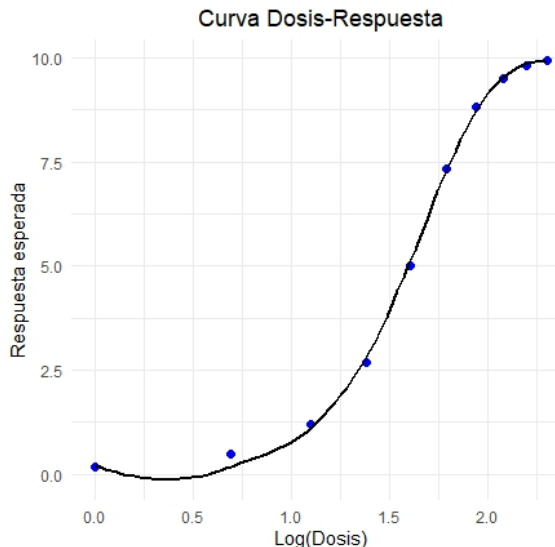


Figura 2.2: Curva Log Dosis vs Respuesta Esperada. Datos simulados en software R

$$y = F(x), y \equiv E(Y) \tag{2.1}$$

Siendo y la respuesta, x la dosis, y $F(x)$ una función real de valor único de x para todas las dosis en el rango de interés.

Aunque ha habido cierta discusión sobre la función que mejor se ajusta a esta curva y hay pequeñas variaciones en la función matemática que define la curva dosis-respuesta según la bibliografía consultada, la más empleada es la función logística de 3 o 4 parámetros (ecuación 2.2), [20], [21], [22]:

$$Y = \frac{d - a}{1 + e^{-b(c - \log(x))}} + a \tag{2.2}$$

Donde Y es la respuesta y x el valor de la dosis; a y d son, respectivamente, el valor mínimo y el valor máximo del rango de valores para la respuesta, c es la dosis resultante de la respuesta que está en el punto medio entre a y d (ED_{50}) y, por último, b es la pendiente relativa que determina la inclinación de la curva alrededor del valor de c . Si para este modelo el valor del límite inferior es igual a 0, $a = 0$, se convierte en un modelo logístico de 3 parámetros [21].

2.3. Determinación de la potencia biológica estimada

La potencia es la medida de la bioactividad de un compuesto. Como ya se ha comentado anteriormente en este texto, una de las finalidades de un inmunoensayo es la determinación de la potencia biológica estimada [19]. En los años 1930, J.H Burn llegó a la conclusión de que: era mucho más fiable determinar las estimaciones de potencia obtenidas comparando el material que se va a ensayar con un estándar, que intentar medir la potencia absoluta [18]. Esto es en gran medida debido a la gran variabilidad inherente a los bioensayos [23]. Partiendo de esta premisa, se desarrollaron varios métodos para estimar la potencia biológica en comparación con un estándar de referencia. Históricamente, el más estudiado es el modelo paralelo lineal [19]; aunque el modelo logístico de 4 parámetros, que estudia la totalidad de la curva dosis-respuesta, puede ser un enfoque más interesante [23].

El principio es el mismo para los dos modelos: si se realiza un ensayo con dos preparaciones (una conocida de referencia o estándar, y otra desconocida), y las curvas dosis-respuesta de cada una de las dos preparaciones caen una encima de la otra, quiere decir que ambas preparaciones proporcionan exactamente la misma respuesta biológica [15], [24]. Es decir, la potencia relativa de la preparación de prueba es de 1 respecto al estándar de referencia [24]. Si, por el contrario, la curva de la preparación desconocida está a la derecha (gráficamente) del estándar: entonces será menos potente que este. De la misma forma, si la preparación desconocida se encuentra a la izquierda será más potente [15]. Estos ensayos son concebidos como ensayos de dilución; donde la preparación T debe, en teoría, derivar de la preparación S, por dilución con componentes inertes. Si lo anterior no se cumple: las curvas dosis-respuesta de ambas preparaciones diferirán significativamente una de la otra y esta aproximación no será válida. [19].

Si se tiene en cuenta la ecuación 2.1 anterior; la ecuación aplicada a la preparación estándar, a partir de ahora también se le llamará S, se formularía de la siguiente forma (ecuación 2.3) [19]:

$$Y_S \equiv E(y|x) = F(x) \quad (2.3)$$

Según lo anterior una preparación de prueba o *test*, a partir de ahora T, tiene ρ veces la potencia de la preparación S. Por lo tanto, Si una dosis x de T es equivalente a una dosis ρx de S, cualquiera que sea el valor de x; la función de regresión correspondiente para T es (ecuación 2.4) [24]:

$$Y_T = F(\rho x) \quad (2.4)$$

Por lo tanto, determinando el valor de ρ se puede estimar la potencia biológica. Este parámetro mide, en realidad, la distancia horizontal entre las dos líneas, que debe ser igual para todo el rango de dosis, y proporciona una estimación de la potencia de T [16].

En este modelo simplificado para su explicación, solo se tienen en cuenta preparaciones S y T, pero puede darse el caso de más de una preparación desconocida que determinar. A lo largo de este texto se hablará principalmente de preparación T para la preparación desconocida; cuando sea necesario diferenciar entre más de una preparación desconocida se emplearán las denominaciones, U y V respectivamente.

2.3.1. Modelo paralelo lineal

El modelo paralelo lineal se emplea cuando la relación entre dosis y respuestas es, en realidad, lineal (Figura 2.3) [15], [19], [25]. En el caso de los inmunoensayos esta situación no es muy habitual. En la práctica, se aplica este modelo teniendo en cuenta la siguiente consideración: el modelo paralelo lineal es aplicable a curvas dosis-respuesta sigmoides si se selecciona un rango de dosis para el cual las respuestas o, respuestas adecuadamente transformadas (por ejemplo el logaritmo neperiano, \ln , de la respuestas), sean aproximadamente lineales cuando se representan frente a $\ln(dosis)$ [4].

Dentro de la curva dosis-respuesta presentada en el sección 2.2 anterior (Figura 2.2); si se eliminan las regiones pertenecientes a la asíntota superior e inferior, se obtendría una región donde se podría aplicar el modelo paralelo-lineal. En este caso las rectas generadas no serán estrictamente lineales, sino *cuasi*-lineales.

Para poder aplicar este modelo es necesario que se cumplan una serie de consideraciones previas [4], [19], [25]:

1. Aleatoriedad. Los diferentes tratamientos (muestras) deben asignarse de forma aleatoria a las unidades experimentales.
2. Normalidad. Las respuestas obtenidas para cada muestra deben distribuirse según la distribución normal.

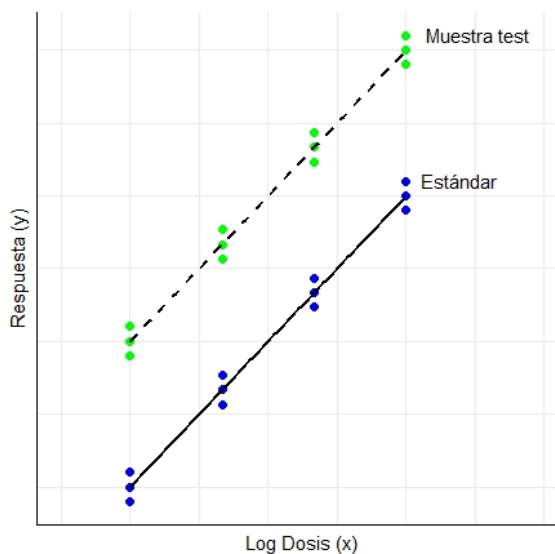


Figura 2.3: Ejemplo de modelo paralelo-lineal para un ensayo de 4 puntos y 3 réplicas por punto. Datos simulados en software R

3. Homogeneidad de varianzas. Las desviaciones estándar de las respuestas de la muestra y el estándar no deben diferir significativamente una de la otra.

La condición 1, en el caso de inmunoensayos de tipo ELISA, se cumple en tanto que las diferentes muestras se distribuyen de manera uniforme en la placa de ensayo.

Las condiciones 2 y 3 es necesario tenerlas en cuenta con los datos de varios ensayos, puesto que habitualmente, los datos generados en un solo ensayo no son suficientes para determinar si se cumplen estas condiciones [25]. En ocasiones, el no cumplimiento de las condiciones 2 y 3 se puede solucionar con transformaciones de las respuestas. Algunas de las transformaciones más comunes son: $\ln(y)$, \sqrt{y} ó y^2 [4].

Por otra parte, en la literatura consultada también se cita que, en ensayos simétricos balanceados, pequeñas desviaciones de las condiciones 1 y 2 no introducen grandes fallos en el análisis posterior, en tanto que se incluyan varias réplicas por cada nivel de tratamiento [4] [25]. Un modelo simétrico balanceado es aquel que tiene el mismo número de dosis para cada muestra y el mismo número de réplicas para cada dosis [25].

Otras dos consideraciones son necesarias [4], [15], [19], [25]:

4. Linealidad. La relación entre el logaritmo de la dosis y la respuesta se puede representar con una línea recta sobre el rango de dosis empleadas (es necesario realizar el ensayo con más de dos dosis por tratamiento).
5. Paralelismo. La línea recta de cada preparación desconocida ensayada, T/U/V/..., debe ser paralela a la línea recta representada por el estándar, S.

Por lo tanto, la ecuación que define el modelo paralelo-lineal viene dada por la ecuación 2.5 [15], [19]:

$$Y_{ijk} = \alpha_i + \beta \log(x_{ij}) + \epsilon_{ijk}, \epsilon_{ijk} \sim N(0, \sigma^2) \quad (2.5)$$

Donde Y_{ijk} es la k respuesta medida en la x_{ij} dosis, siendo i la preparación y j el número de dosis para cada preparación. α es el intercepto de la recta y β la pendiente común (todas las rectas deben ser paralelas, es decir comparten una pendiente común), y ϵ es el residuo, que se asume independiente y normalmente distribuido.

En este caso, una vez comprobado que se cumplen las suposiciones del modelo, la estimación de la potencia viene dada por la ecuación 2.6 [15]:

$$\log(\hat{\rho}) = \frac{\hat{\alpha}_T - \hat{\alpha}_S}{\hat{\beta}} \quad (2.6)$$

2.3.2. Modelo logístico de cuatro parámetros

Otro enfoque, y en ocasiones mucho más interesante, es el modelo logístico de cuatro parámetros. El modelo paralelo-lineal surge como una simplificación para poder realizar el análisis de una forma más sencilla, pero exige trabajar en un número limitado de dosis alrededor del valor de máxima pendiente [26]. Sin embargo, en ocasiones puede ser preferible emplear un diseño de ensayo con más dosis, distribuidas en el rango completo de la curva dosis-respuesta; este diseño permitirá estimaciones más robustas ante la variabilidad entre ensayos y las potencias impredecibles [26].

El concepto alrededor de este modelo se puede ver gráficamente en la Figura 2.4 y se basa en la curva dosis-respuesta presentada en la Sección 2.2.

Generalmente se necesita un software informático que realice los cálculos, pues, a diferencia del diseño anterior, manualmente puede ser muy laborioso [4]. En este caso, además de testar un modelo válido para una pendiente común, deben también compartir los valores para el mínimo y máximo de respuesta. Con esto; solo será diferente el parámetro c , anteriormente citado en la ecuación 2.2, [22].

Una de las aproximaciones tiene en cuenta lo siguiente: en las asíntotas el modelo se asemeja a un modelo de relación de pendientes (*slope-ratio*, en inglés) y en la región central a un modelo paralelo-lineal y se desarrolla en la referencia [26]. Las estimaciones de dichos parámetros en este caso se puede realizar con el método de máxima verosimilitud [22] [26]. En la práctica, este modelo plantea diferentes dilemas estadísticos que probablemente requieran diferentes soluciones según el caso particular planteado. A lo largo de la bibliografía hay numerosos enfoques de este mismo problema. Dicho lo anterior, es posible que para la implementación de este método de análisis a un ensayo particular, sea necesaria la supervisión por un estadístico experimentado [4].

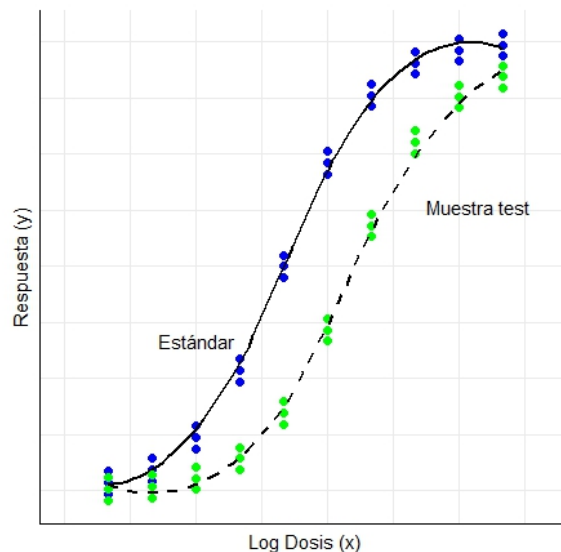


Figura 2.4: Ejemplo de modelo logístico de 4 parámetros para un ensayo de 10 diluciones y 3 réplicas. Datos simulados en software R

2.4. Alergia

El término “alergia” lo introdujo von Pirquet en 1906, hoy en día se emplea para referirse a un amplio rango de reacciones que se producen cuando el organismo entra en contacto con una sustancia, que en principio es inocua, y el sistema inmunológico la interpreta como nociva; desarrollando una respuesta inmunológica exagerada conocida como reacción alérgica [27]. A esa sustancia se le conoce como alérgeno. Hoy en día, enfermedades como la rinitis alérgica, el asma o el eccema son las mayores causas de enfermedad crónica y su prevalencia está en aumento [27] [28]. Estas enfermedades, en conjunto, se conocen como enfermedades atópicas, y su mecanismo en común es que son enfermedades mediadas por Inmunoglobulina E (IgE). [27].

La inmunoterapia con alérgenos específicos es reconocido por la Organización Mundial de la Salud, como el único tratamiento capaz de modificar el curso natural de las enfermedades atópicas [28] [29]. A su vez, los inmunoensayos como ELISA, y las técnicas para la determinación de potencia son importantes en este campo porque permiten determinar la actividad alérgica de un extracto alérgico.

La regulación Europea al respecto no proporciona estándares de referencia para medir la potencia alérgica de un extracto. Sin embargo, en 1989, las directrices nórdicas proporcionaron recomendaciones para la estandarización de extractos alérgicos mediante pruebas cutáneas y estas has sido adaptadas por los principales fabricantes de extractos alérgicos para establecer un estándar de referencia interna [30]. Estos estándar de referencia interna son establecidos a partir de estudios *in-vivo* en pacientes. Posteriormente los extractos comerciales son analizados con técnicas *in-vitro*, como ELISA, frente al estándar de referencia para asignar su potencia relativa [30].

Capítulo 3

Materiales y Métodos

3.1. Obtención de los datos

Aunque el presente trabajo pretende dar una solución general, para la estimación de potencia biológica a partir de datos de inmunonensayo (ELISA), y se pueda emplear para este propósito por cualquier usuario; se realizará una breve reseña del método particular de obtención de los datos.

Se han empleado datos de diversos ensayos ELISA para la realización del mismo. Entre ellos un ELISA IgE de Competición [31] y un ELISA IgG de Inhibición [32], en sus correspondientes referencias se puede encontrar un procedimiento detallado de ambos métodos. Pese a que no se describirán todos ellos, por no resultar imprescindible para la comprensión y seguimiento del presente documento, se formulará el procedimiento (en líneas generales) de uno de ellos, como ejemplo, para una mayor comprensión del contexto del trabajo realizado.

3.1.1. ELISA IgE de Competición

El ensayo seleccionado para detallar el proceso de obtención de datos de inmunoensayo, que posteriormente se puedan analizar con las soluciones planteadas en este trabajo; es un ELISA IgE de Competición empleado para determinar la actividad alérgica total en extractos alérgicos y sus derivados [31].

Se emplean placas de microtitulación de poliestireno como fase sólida del ensayo, que en un primer paso se recubren/sensibilizan con un anticuerpo monoclonal anti-IgE humana. Posteriormente, después de varios ciclos de lavado, se añadirá a la placa una solución diluida de muestras de suero humano positivas para el alérgeno que se pretende medir y se incubará durante un período de tiempo determinado. Durante este período de incubación la IgE presente

en las muestras de suero humano se unirá a los sitios de unión del anticuerpo anti-IgE humana previamente unido a la placa.

Después de los ciclos de lavado necesarios; la placa de ensayo se incuba con una solución que contiene el alérgeno libre y el marcado en proporción variable, esta es la fase de competición. Esta solución se prepara con anterioridad y contiene una dilución seriada de alérgeno libre junto con una dilución constante de alérgeno marcado en proporción 1:1. Con esto se conseguirá una relación inversa de la respuesta frente a la cantidad de muestra presente. Esto es debido a que hay un número limitado de sitios de unión para ambos antígenos; el alérgeno marcado y el libre.

Cuando termina la segunda fase de incubación, se realizan los ciclos de lavado determinados y se incuba con una solución de desarrollo adecuada. Esta solución contiene 3,3'-4,4'- tetrametilbenzidina como cromógeno y H_2O_2 como sustrato de la reacción, que serán los responsables del desarrollo de color cuando reaccionan con la HRP del alérgeno marcado. La reacción se para con la adición de ácido sulfúrico diluido. Finalmente, los datos de absorbancia se obtienen mediante la lectura de la placa en un espectrofotómetro una longitud de onda de 450nm. Los resultados se expresan en unidades de densidad óptica (OD). Este procedimiento se realiza en paralelo para estándar y muestra.

3.2. Análisis estadístico y determinación de potencia. Modelo paralelo-lineal

Antes de realizar el análisis estadístico mediante el modelo paralelo lineal es necesario examinar los datos y, si resulta necesario, filtrarlos adecuadamente.

3.2.1. Filtrado de datos

Cada punto de la curva dosis-respuesta es ensayado por triplicado. Para determinar el nivel de dispersión entre replicados se determinará el coeficiente de variación (CV) para cada grupo de replicados. El CV en este caso se calcula como el ratio entre la desviación estándar y la media.

En este sentido, un coeficiente de variación alto no es deseable, ya que indica un alto nivel de dispersión entre replicados, lo que es lo mismo que una mala calidad del ensayo realizado. Sin embargo, en ocasiones, se pueden producir contaminaciones puntuales en pocillos concretos, obteniendo finalmente un valor anormalmente alto o bajo de respuesta (OD). Este paso de filtrado pretende que, estas desviaciones puntuales no relacionadas con la naturaleza de la muestra, no intervengan en el posterior análisis pudiendo afectar al resultado obtenido.

Para este caso particular se ha fijado un CV del 15 % como valor de corte, en base a estudios previos. Si para alguno de los grupos de replicados, el valor del CV de las ODs sobrepasa el 15 %, se identificará el valor responsable del aumento de la dispersión. Teniendo en cuenta que el diseño planteado del modelo paralelo-lineal es válido para ensayos simétricos con el mismo número de dosis y el mismo número de réplicas por cada dosis ensayada [4] [25]; esto quiere decir que dicho valor no se puede eliminar sin sustituirlo por otro. En este caso se sustituirá por la media de los dos valores restantes.

Por otra parte, el máximo de datos filtrados del total de datos obtenidos no podrá superar el 10 %. Por lo que este procedimiento se llevará a cabo un máximo dos veces (para los dos valores más altos de CV, siempre y cuando sobrepase el 15 %) para cada curva dosis-respuesta (o muestra).

Esto es así teniendo en mente este máximo de valores filtrados y el diseño de placa en el que, como máximo se pueden ensayar 7 dosis. Hay que tener en cuenta que, si en un paso posterior del análisis, el rango de dosis-respuestas empleadas no tiene más de tres niveles, quizá no sería aconsejable emplear los datos filtrados si dos de los tres niveles contienen datos filtrados para alguna de las muestras.

3.2.2. Selección del rango de dosis para ajustar el modelo

Según el modelo paralelo-lineal: seleccionado para este análisis debido a su menor complejidad, es necesario ensayar el rango de la curva dosis-respuesta donde la pendiente es máxima, y se puede aceptar linealidad. Por ello, se ha realizado un paso previo de elección del rango de dosis ensayadas que serán empleadas para el análisis en base a este principio.

Para determinar cual es el rango seleccionado, se han testado las suposiciones del modelo (regresión, linealidad y paralelismo) en todos los diferentes rangos posibles dentro del número total de dosis ensayadas. Finalmente se ha seleccionado el rango de dosis que cumple los criterios anteriores y que a su vez:

- Supone el mayor número de dosis posibles.
- Prioriza aquel rango de dosis que contiene la dosis intermedia. Lo habitual es ajustar previamente el ensayo para que el valor de máxima pendiente de la curva dosis-respuesta (denominado ED_{50}) corresponda al valor de dosis intermedio en todo el rango de dosis, por eso se ha tenido en cuenta este criterio.

3.2.3. Determinación de potencia biológica (PLM)

Modelo paralelo-lineal

Una vez seleccionado el rango de dosis y realizado el ANOVA con el que se comprueban las suposiciones del modelo y la validez del ensayo. A continuación se muestra el análisis estadístico propuesto por el texto de referencia [4] para el análisis del cumplimiento del modelo, para ensayos simétricos de una o más muestras, también desarrollado y disponible para su consulta en la siguiente referencia, [25].

En la Tabla 3.1 se muestran las fórmulas para determinar el valor de la preparación total (P) y del contraste lineal (L) para cada preparación. El valor de la preparación total (P, ecuación 3.1) para la muestra i , es igual al sumatorio de la respuesta media para cada una de las j dosis. El valor para el contraste lineal (L, ecuación 3.2) para la muestra i , es igual al sumatorio del producto del valor de j por la respuesta media para cada una de las j dosis, menos un medio por el número de dosis (d) más 1 y por el valor de P para esa misma muestra.

Tabla 3.1: Fórmulas para el contraste modelo paralelo lineal para cada preparación [4], [25]

	Estándar (i = S)	Muestra 1 (i = T)	Muestra 2 (i = U, ...)
Respuesta media de la dosis más baja (j = 1)	S ₁	T ₁	U ₁
Respuesta media de la 2ª dosis (j = 2)	S ₂	T ₂	U ₂
...
Respuesta media de la dosis más alta (j = d)	S _d	T _d	U _d
Preparación Total (P)	$P_s = \sum S_j$	$P_t = \sum T_j$	$P_u = \sum U_j$ (3.1)
Contraste lineal (L)	$L_S = \sum jS_j - \frac{1}{2}(d+1)P_S$	$L_T = \sum jT_j - \frac{1}{2}(d+1)P_T$	$L_U = \dots$ (3.2)

Tres fórmulas adicionales (ecuaciones 3.3, 3.4, 3.5) son necesarias para la posterior construcción de la tabla de análisis de varianza (ANOVA), [4], [25].

$$H_P = \frac{n}{d} \tag{3.3}$$

$$H_L = \frac{12n}{d^3 - d} \tag{3.4}$$

Donde n es el número de réplicas para cada nivel (dosis) ensayada y d , como ya se ha mencionado anteriormente, es el número total de dosis.

$$K = \frac{n(\sum P_i)^2}{hd} \tag{3.5}$$

Donde P_i es el sumatorio del valor de P para todas las muestras ensayadas y h , es el número total de preparaciones diferentes (S, T,...).

Con todo lo anterior, se puede construir la tabla ANOVA (Tabla 3.2) con la que testar la validez del ensayo. En esta tabla y se refiere a cada valor de Y_{ijk} , mientras que \bar{y} se refiere a la media de todas las respuestas obtenidas en el ensayo.

Con los datos de la tabla ANOVA es posible calcular los cuadrados medios, dividiendo el valor de la suma de cuadrados por sus grados de libertad correspondientes. Los valores resultantes se pueden expresar a su vez como un ratio frente al valor del error residual (F-ratio). Los valores para los F-ratio son evaluados frente a los valores críticos de la distribución F de Fisher para las probabilidades $P[X > x]$ con un nivel específico de α .

Tabla 3.2: Tabla ANOVA [4], [25]

Fuente de variación	Grados de libertad (<i>f</i>)	Suma de cuadrados
Preparaciones	$h-1$	$SS_{prep} = H_p(\sum P_i^2) - K$
Regresión lineal	1	$SS_{reg} = \frac{1}{h} H_L(\sum L_i)^2$
No-paralelismo	$h-1$	$SS_{par} = H_L(\sum L_i^2) - SS_{reg}$
No-linealidad	$h(d-2)$	$SS_{lin} = SS_{treat} - SS_{prep} - SS_{reg} - SS_{par}$
Tratamientos	$hd-1$	$SS_{trat} = n(\sum S_j^2 + \sum T_j^2 + \dots + \sum i_j^2) - K$
Error residual	$hd(n-1)$	$SS_{res} = SS_{tot} - SS_{treat}$
Total	$nhd-1$	$SS_{tot} = \sum (y - \bar{y})^2$

Los resultados de este análisis son estadísticamente válidos si se cumple lo siguiente [4]:

- El término de regresión lineal es significativo
- Los términos de paralelismo y linealidad se cumplen. Esto es así cuando los test resultan no significativos, ya que la hipótesis nula es que existe linealidad y paralelismo. Si no resulta significativo, no podemos rechazar la hipótesis nula, esto es, existe paralelismo y linealidad.

Pendiente común, potencia e intervalos de confianza

Finalmente, cuando se cumplen los criterios de validez expuestos, se puede calcular el valor de la pendiente común y la estimación de la potencia estimada de la muestra frente al estándar. Todo ello según se indica en la Ph. Eur. [4]. Para realizar la determinación de la potencia se necesita el valor para la pendiente común del modelo anterior, β (ecuación 3.6), tal y como se puede deducir de la ecuación 2.6,[4].

$$\beta = \frac{H_L(\sum L_i)}{Inh} \tag{3.6}$$

Donde H_L viene definida por la anterior ecuación 3.4, el sumatorio de L_i se refiere al sumatorio de todas los contrastes lineales calculados según las ecuaciones 3.2, I es el ln del ratio entre dosis adyacentes, n es el número de réplicas para cada nivel (dosis) y h es el número de preparaciones diferentes (S, T, ...)

Con el valor de β anterior, el cálculo de potencia para la preparación T (como ejemplo) resulta en (ecuación 3.7) [4]:

$$\ln(\text{Potencia}_{\text{estimada}})_T = M'_T = \frac{P_T - P_S}{d\beta} \quad (3.7)$$

Y, finalmente los intervalos de confianza para este valor se calculan con la siguiente ecuación (3.8) [4]:

$$CI = CM'_T \pm \sqrt{(C - 1)(CM'^2_T + 2V)} \quad (3.8)$$

$$\text{Donde } C = \frac{SS_{reg}}{SS_{reg} - s^2 t^2} \quad \text{y } V = \frac{SS_{reg}}{\beta^2 dn}$$

El valor para t se puede consultar en las tablas para la distribución del estadístico t de Student.

En ocasiones, en función de la potencia asignada al estándar, es necesario aplicar un factor: para convertir la potencia relativa de la muestra frente al estándar en unidades de potencia, cualquiera que sea la unidad empleada para definirla [4].

3.3. Disposiciones legales y normas aplicadas

El proyecto se lleva a cabo cumpliendo con las normas de Trabajos Finales establecidas por el Máster U. en Bioinformática y Bioestadística UOC-UB y la Universitat Oberta de Catalunya.

Aunque la aplicación de las soluciones presentadas en este trabajo es universal para cualquier ensayo ELISA cuya finalidad sea determinar la potencia biológica de una muestra. Como ya se ha comentado anteriormente, en el ámbito de aplicación en el que se desarrolla el proyecto, es necesario que este análisis se haga siguiendo las directrices de la Dirección Europea de Calidad de los Medicamentos y la Atención Sanitaria (EDQM) recogidas en la Pharmacopoeia Europea. Por eso, se ha empleado este texto como referencia para llevar a cabo el análisis; al igual que, se emplearán sus datos de ejemplo como datos para la validación del análisis propuesto.

3.3.1. Datos de ejemplo para validación

Para la validación del análisis estadístico propuesto, se emplearán los datos de ejemplo que proporciona la Ph. Eur. en su capítulo 5.3, específicamente en el punto 5.1.4 de este capítulo. En este caso se trata de un ejemplo de bioensayo *in-vitro* de tres vacunas para la hepatitis B frente a un estándar. Para ello se realiza un ensayo con cinco dosis y un diseño completamente aleatorizado (crd). Los datos proporcionados se documentan en el Anexo A. En ese mismo anexo se documentarán los cálculos manuales para validar que los resultados obtenidos, mediante la solución adoptada, son veraces.

A continuación, tabla 3.3, se muestran los valores estimados de potencia para las 3 vacunas según el cálculo manual para, posteriormente, poder compararlos con los datos obtenidos con la solución propuesta.

Tabla 3.3: Resultados de potencia e intervalos de confianza 95 %

	Potencia	Límite inferior	Límite superior
Vacuna T	40.5	43.4	46.5
Vacuna U	32.9	35.2	37.6
Vacuna V	36.8	39.4	42.2

3.4. Software empleado

Para la elaboración del presente documento se empleó un editor de textos basado en el sistema tipográfico \LaTeX . La planificación temporal del proyecto se ha llevado a cabo con la ayuda de la herramienta GanttProject.

Para la elaboración de la plantilla de entrada de datos se eligió una hoja de cálculo (formato *.xlsx*) elaborada en el editor Excel de Microsoft. Se ha seleccionado este sistema para la entrada de los datos brutos por ser un sistema ampliamente conocido y presentar un interfaz amigable para usuarios menos experimentados.

Para la realización de la parte técnica de este proyecto se utilizó el lenguaje de programación **R** (versión 4.3.2) dentro del entorno de desarrollo integrado (IDE) RStudio, que facilita la programación [33]. Asimismo, dentro de la herramienta RStudio, se empleó el formato R Markdown para la elaboración de un documento de salida en formato *.pdf* donde se documente el análisis PLM realizado sobre un conjunto de datos particular. Se ha decidido este sistema porque la presentación de los resultados en este formato minimiza la posible manipulación de los resultados obtenidos, algo que insiste en el cumplimiento de las buenas prácticas de laboratorio. Además, un formato *.pdf* también permite su supervisión (y firma correspondiente) de manera electrónica; lo que reduce la necesidad de documentación en papel y lo hace una solución más sostenible.

3.4.1. Software R

R es un entorno gráfico y estadístico potente y flexible, que se distribuye gratuitamente bajo la Licencia Pública GNU. Inicialmente fue desarrollado por Ross Ihaka y Robert Gentleman, pero que hoy en día sigue en desarrollo constante y consta de una cooperativa internacional para la toma de decisiones [34].

Se ha elegido esta herramienta por ser altamente adaptable, con herramientas básicas de análisis integradas, extremadamente útiles, y además por disponer de una amplia cantidad de librerías con funciones específicas para multitud de propósitos. También es importante su orientación a la interpretación y visualización de los datos, para lo que cuenta con gran cantidad de herramientas para elaborar gráficos igualmente útiles.

Dentro de R se han utilizado funciones de las librerías base como pueden ser *base*, *stats* o *graphics* (versión 4.3.2). Además se han empleado las siguientes librerías más específicas:

- La librería “*openxlsx*” (versión 4.2.5.2) para la lectura de datos provenientes de Excel [35].
- La librería “*xtable*” (versión 1.8.4) para convertir las tablas y data frame de formato **R** a formato \LaTeX con la intención de incluirlas en este documento.
- La librería “*devtools*” (versión 2.4.5) ya que contiene herramientas para facilitar el desarrollo de librerías **R** [36].
- La librería “*roxygen2*” (versión 7.2.3) ya que facilita la generación de la documentación de la librería [37].

Capítulo 4

Resultados y Discusión

4.1. Plantilla *.xlsx* para registro de datos obtenidos.

Con el fin de facilitar el posterior análisis y desarrollo del *script* se ha diseñado una plantilla donde se volcarán los datos obtenidos del inmunoensayo en cuestión (ver plantilla en Apéndice B). Esta plantilla se ha diseñado teniendo en cuenta lo siguiente:

- Formato amigable para usuarios con conocimientos limitados en informática.
- Será necesario destinar un número determinado de “rellenables” para contener la información del ensayo necesaria para identificarlo y para indicar las condiciones de análisis posterior.
- Zona delimitada donde volcar los datos de ODs (respuesta obtenida) del ensayo.

Teniendo en cuenta lo anterior se ha optado por un diseño en formato hoja de cálculo, concretamente un formato *.xlsx* de Microsoft Excel. Se ha decidido emplear este sistema por resultar un entorno amigable, de fácil uso y ampliamente conocido y empleado; tanto en el ámbito laboral como de investigación.

La plantilla Excel generada está diseñada de tal forma que se destinan celdas específicas para introducir los datos necesarios para el posterior análisis, tales como: potencia asignada, condiciones de análisis, etc. Así como celdas específicas para asegurar la trazabilidad de los datos, nombre de las muestras, identificación del ensayo, etc.

A su vez, teniendo en mente el diseño habitual de una placa ELISA (una placa de 96 pocillos, de 12 columnas por 8 filas), se ha destinado una área de 96 celdas para contener los resultados obtenidos del ensayo. La mayoría de *software* informáticos diseñados para la lectura de placas

de micro-titulación suelen tener diversos formatos de presentación de los resultados, y en la mayoría una de las opciones predeterminadas es que los resultados se muestren en una hoja de cálculo, con lo que tan solo será necesario copiar los datos de dicha hoja a la plantilla.

El ensayo realizado y el diseño de la placa de ensayo determina, en parte, el diseño de la plantilla. Para este caso particular, a la hora de realizar el ensayo las muestras se distribuyen de la siguiente manera: la primera fila de la placa irá destinada a los blancos y controles que el ensayo requiera (blanco de placa, control positivo y negativo); las filas de 2 a 8 contendrán la titulación de la muestra (estándar o test) por cada columna. Es decir, entre los pocillos 2 y 8 de cada columna se obtendrá una curva dosis-respuesta de una de las muestras ensayadas, y cada curva correspondiente a una muestra particular se ensayará en 3 columnas distintas (dando lugar a 3 réplicas por nivel de la curva o dosis).

Como resultado del diseño propuesto anterior, teniendo en cuenta que la placa es de 12 columnas, el número máximo de muestras a ensayar es de 4, un estándar y tres muestras problema o test; y el número máximo de dosis o niveles es de 7.

Los datos correspondientes a los resultados obtenidos (ODs), que se vuelcan en el área anterior de 96 celdas; se volcarán de forma automática en una segunda tabla, con un formato tipo *dataframe*, para su posterior exportación a **R**. Esta segunda tabla tiene 5 columnas que corresponden con: Respuesta, Muestra, Dosis, Replicado y Nivel. La primera columna, Respuesta, es la que recoge los datos de las ODs volcadas. En la plantilla Excel presentada, esta segunda tabla no figura puesto que, al ser una tabla que se autorellena con los datos introducidos y para evitar su manipulación se encuentra oculta en la hoja de cálculo. La disposición y características son similares a las presentadas en la Tabla A.1 del Anexo A.

Una celda de especial atención es la que determina el rango de análisis. En ella se indicará, del total de muestras ensayadas cuales son las que se quieren incluir en el análisis posterior. Esto es interesante para los casos en los que, por ejemplo, se sepa previamente que ha sucedido un error de ejecución de ensayo en una muestra concreta, o una columna correspondiente a una muestra tenga valores atípicos que hagan aconsejable eliminar dicha muestra del análisis. Según el valor determinado en esta celda, se seleccionará posteriormente en **R** el rango de análisis indicado.

Por otra parte, se ha tenido en cuenta la posibilidad de ensayar menos dosis de 7 (es decir, no emplear todas las filas de la placa). En ese caso, el valor para la respuesta en las dosis “faltantes” se igualará a 0. En un paso posterior, en el script en **R**, se eliminarán las filas para las que la respuesta sea 0.

Todas estas consideraciones han sido tomadas para adaptar la plantilla de entrada de datos a un caso particular de un inmunoensayo con un diseño específico. Aunque se han tenido en cuenta ciertas variaciones; si el diseño es diferente a lo indicado será necesario adaptar esta plantilla al diseño elegido.

Para una mayor facilidad en un paso posterior de exportación de datos de Excel a **R**, las

regiones o celdas individuales que contienen la información a exportar se han alojado en celdas con nombres asignados o, en caso de la tabla de datos, en regiones con nombres asignados.

4.2. Librería plm.crd

Para la importación de los datos al entorno de trabajo, así como para la aproximación estadística propuesta, se han diseñado una serie de funciones que, finalmente, se recogen en una librería a la que se ha denominado “plm.crd”. En este capítulo de resultados se presentarán las funciones elaboradas y su funcionamiento. En el Apéndice C se puede consultar el código fuente de cada una de ellas.

La descripción del paquete es la siguiente:

- Librería: plm.crd.
- Tipo: *Package*.
- Título: *Parallel-line model. Completely randomised design*.
- Versión: 0.1.0.
- Autor: Rocío Lorenzo.
- Mantenedor: Rocío Lorenzo (r.lorenzolopez@hotmail.com).
- Descripción: *Parallel-line Model Analysis of immunoassay data with a completely randomized design*.
- Licencia: GPL-3.
- Codificación: UTF-8.

Las funciones que contiene se describen a continuación. Además de ello, se emplean unos datos como ejemplo, los mismos que servirán para la validación de la librería y que pertenecen al texto de referencia empleado[4], que se cargan con el mismo y figuran bajo en nombre de *hepatitis_example*.

4.2.1. Funciones lectura de datos en R

Independientes al resto del análisis, y para facilitar la carga de datos desde la plantilla Excel generada; se han desarrollado dos funciones muy enfocadas al propósito y desarrollo del presente trabajo. Estas funciones, a su vez, emplean la función *readWorkbook* (propia de la

librería *openxlsx*) [35]. Una de las funciones creadas toma los valores de las celdas nombradas y otra los de la región donde se alojan los datos brutos y (ambas) los almacenan en el entorno de trabajo.

Función *load.info*

Descripción: la función *load.info* descarga y almacena en distintas variables en el entorno de trabajo de **R** los valores de cada celda nombrada de la plantilla Excel (formato *.xlsx*) que se le proporcione como argumento.

Argumentos:

- *filename*: como argumento toma únicamente el nombre completo del archivo tipo *.xlsx* de donde debe extraer la información necesaria.

La función devuelve, tal y como se indicó, en el entorno de trabajo tantas variables como regiones nombradas existan en la plantilla de datos, así mismo, los nombres de las variables son iguales al nombre de las celdas nombradas del archivo de origen, y el contenido igual al contenido de dichas celdas.

Esta función presenta problemas a la hora de extraer datos de una región de más de una celda en excel, por lo que se ha desarrollado una segunda función para este propósito.

Función *load.data*

Descripción: la función *load.data* descarga y almacena en una variable de tipo *dataframe* en el entorno de trabajo de **R** los valores para la región nombrada concreta del archivo tipo *.xlsx* que se le proporcione como argumento.

Argumentos:

- *filename*: se refiere al nombre del archivo tipo *.xlsx* de donde debe extraer la información necesaria.
- *data*: se refiere al nombre de la región en el archivo *.xlsx* donde están contenidos los datos.

La función devuelve, al entorno de trabajo, un *dataframe* con las mismas dimensiones que el rango de celdas nombradas del archivo de origen. El nombre que asignará al *dataframe* generado será *rawData*.

4.2.2. Funciones para tratamiento y filtrado de datos

Como se ha comentado, las funciones anteriores son independientes del análisis paralelo que se propone en el presente texto. Una vez cargados los datos, de la manera que resulte más conveniente, el resto de funciones se pueden emplear independientemente de la procedencia de los mismos.

Antes de realizar el análisis, es necesario preparar o adaptar los datos, para ello se ha generado la función *assay.data*. Además, si se requiere un filtrado de los datos tal y como se ha indicado en el apartado 3.2.1 de Materiales y métodos se puede utilizar la función *filter.data*, desarrollada para tal fin.

Función *assay.data*

Descripción: la función *assay.data* adapta un *dataframe* a un formato específico, en función de los valores asignados a los distintos argumentos.

Argumentos:

- *data*: se refiere al *dataframe* o matriz sobre la que realizar las operaciones indicadas.
- *y.treat*: aquí se debe indicar si a la variable Respuesta (*Response* en el *dataframe*) se le debe aplicar algún tratamiento. Por ejemplo si en vez de la respuesta tal cual, es preferible emplear el logaritmo de la misma. Este argumento es nulo por defecto, es decir, no se aplica ningún tratamiento sobre la variable Respuesta.
- *dilutionRatio*: aquí se debe expresar cual es el ratio existente entre dosis adyacentes en el ensayo en cuestión. Por defecto se ha establecido en 1/2 (0.5) ya que es, *de facto*, una de las ratios más empleadas en este tipo de ensayos.
- *dilution*: calcula la dilución para cada punto. Para ello emplea el valor del ratio de dilución y la dosis para calcular la dilución en escala logarítmica. Por defecto los valores que toma esta variable es igual a: $dilutionRatio + \log(Dosis)/\log(dilutionRatio)$

Esta función devuelve el mismo *dataframe* de entrada con los cambios indicados. Para que esta función funcione correctamente el *dataframe* de origen debe de tener 3 variables mínimo: Respuesta, Dosis, y Muestra (nombradas como *Response*, *Dose* y *Sample*). En esta tercera se debe indicar la muestra a la que pertenece cada dato individual, para que funcione como una variable tipo factor en el resto del análisis. Además, los datos presentados, tienen dos variables adicionales que corresponden al Replicado y al Nivel (nombradas como *Replicate* y *Level* en **R**). La variable Replicado se refiere a las distintas réplicas o valores de respuesta para cada valor de Dosis en una muestra concreta. La variable Nivel toma valores enteros positivos,

específicamente 1, 2, 3, ..., y proporciona una representación discreta de la magnitud relativa de la dosis.

Esta función se ha elaborado con varios propósitos: por una parte, la de añadir la variable *Dilution* necesaria para el posterior ajuste del modelo, y por otra parte, la de realizar de forma sencilla cualquier transformación sobre la variable de interés. Además sirve para adaptar los datos al formato requerido en la mayoría de las funciones desarrolladas a continuación.

Para añadir una característica más a esta función se ha generado una función auxiliar que calcula el cuadrado de la respuesta en caso de ser necesario. Esto se ha realizado así, puesto que la manera de calcular cuadrados predefinida en **R** no hace posible emplearla como función.

Función auxiliar *square* Descripción: Esta función calcula el cuadrado de un número o un vector de números. Argumentos: *x*: puede ser un número o un vector numérico. La función devuelve el valor del cuadrado del número o vector numérico proporcionado.

Función *filter.data*

Descripción: Esta función filtra los datos brutos de la Respuesta con las siguientes condiciones: Filtra aquellas réplicas para cada valor que difieran en CV más de un 15 % hasta un máximo de dos datos filtrados por curva dosis-respuesta o muestra individual.

Argumentos:

- *data*: el *dataframe* de origen donde filtrar los datos

Esta función devuelve el mismo *dataframe* de origen con los datos filtrados que proceda. Para que funcione correctamente el *dataframe* de entrada tiene que tener las variables Respuesta, Muestra, Replicado y Nivel (nombradas como *Response*, *Sample*, *Replicate*, *Level*). El procedimiento de esta función es como sigue: se calcula el CV de la variable respuesta entre réplicas para una misma muestra y nivel, si sobrepasa el 15 % identifica el valor que más se aleja del promedio y lo sustituye por el promedio de los valores restantes. Si más de dos niveles por muestra tienen un CV superior al 15 % la función solo realizará las sustituciones para los dos valores más altos; dejando el resto tal cual.

Se han generado dos funciones auxiliares para facilitar la elaboración de esta función principal, y que pueden resultar de interés en algunos casos.

Función *cv.calc* Descripción: Esta función calcula el CV entre varios números (vector numérico). Argumentos: *vector*: vector numérico sobre el que realizar el cálculo. *FUN*: se puede indicar

mean o *median* dependiendo de con que valor se pretende calcular el CV (media o mediana, respectivamente); el valor por defecto es la media. El resultado de esta función es el CV en % entre los valores proporcionados.

Función *pos.max.diff* Descripción: Esta función devuelve la posición del valor con mayor diferencia respecto a la media. Argumentos: *vector*: vector numérico sobre el que realizar la comprobación. *FUN*: se puede indicar *mean* o *median* dependiendo de con que valor se pretende revisar la máxima diferencia (media o mediana, respectivamente); el valor por defecto es la media. El resultado de esta función es la posición del valor más diferente.

4.2.3. Funciones Modelo paralelo-lineal

Una vez los datos brutos están debidamente cargados en el software, se han tratado y/o transformado convenientemente y, si resulta necesario, se han filtrado; el siguiente paso es ajustar los datos al modelo paralelo-lineal propuesto para, posteriormente, poder determinar el cumplimiento o no de dicho modelo. Para ello se elabora una función sencilla, *pl.model*, que ajusta los datos a un modelo lineal. Además, se elabora una segunda función *best.model* que comprueba, para todos los posibles rangos dentro de las curvas dosis-respuesta, aquellos que cumplen el modelo propuesto y devuelve el mejor de ellos.

Función *pl.model*

Descripción: la función ajusta los datos a un modelo de regresión lineal. Para ello se emplea la función *lm* de la librería base *stats*.

Argumentos:

- *data*: la matriz o *dataframe* donde figuran los datos.

La función devuelve un objeto de clase “*lm*” con un modelo ajustado que presenta las mismas características que el modelo ajustado que propone el texto empleado como referencia. Algo que se ha podido deducir del código fuente de la antigua librería “*pl*” [5] y de los conocimientos explicados y adquiridos en la asignatura de Regresión, Modelos y Métodos, del profesor Francesc Carmona. Un resumen de lo anterior se puede consultar en [38] y [39]

Puesto que esta función es la base de todo el análisis posterior se desarrollará más en detalle el uso de la función *lm* dentro de esta función. La fórmula que toma la función *lm* es $Response \sim factor(Sample) * Dilution + factor(Sample):factor(Dilution)$. En este modelo la variable *y* es la respuesta obtenida en el ensayo, como variables explicativas se toman la variable numérica

Dilución y la variable Muestra como factor (toma cada curva dosis-respuesta como un conjunto de datos individual), además de su interacción y la interacción de las dos mismas variables como factor.

Es importante señalar que para que esta función actúe adecuadamente es necesario adaptar el *dataframe* que se le da como argumento, con la función anteriormente descrita *assay.data*.

Al realizar un ANOVA (función *anova* de la librería *stats*) sobre los datos de este ajuste los valores obtenidos son idénticos a los obtenidos con el análisis manual representado en este texto, anexo A. La variable predictora Muestra tomada como factor representa el test de Preparación. La variable predictora Dilución representa el test de Regresión. La interacción representa los test de no-parallelismo y no-linealidad en función de si la variable Dilución es tomada como variable numérica o como variable de tipo factor, respectivamente.

Función *best.model*

Descripción: dentro de un conjunto de datos de varias curvas dosis-respuesta esta función elige cual es el rango de dosis que es lineal y pasa los test necesarios para poder aplicar el modelo paralelo-lineal.

Argumentos:

- *data*: el conjunto de datos sobre el que es necesario determinar el rango de dosis que cumplen el modelo.

La función devuelve el número de nivel (que se corresponde con el número de dosis) más bajo y más alto entre los que cada curva dosis-respuesta cumple los test de linealidad y paralelismo. Devuelve los dos valores para cada muestra y en un formato *dataframe*. Esta función tiene en cuenta que el número de niveles para cada muestra debe ser igual y exige un mínimo de 3 niveles para cada una de ellas. Puesto que, con menos de 3 niveles, no tiene sentido testar linealidad.

Además; siempre mostrará el rango de puntos más amplio posible y, siempre y cuando sea posible, aquel que incluya la dosis intermedia, puesto que se supone el punto para el cual el ensayo está ajustado, tal y como se indica en el apartado 3.2.2.

De la misma manera, si no es posible conseguir ningún rango que cumpla las características anteriores se realizará una segunda comprobación, y se seleccionará el rango que cumple las características pero con un nivel de $\alpha = 0,01$ para el criterio de no-linealidad.

Como ya se ha mencionado, se trabaja con una curva dosis-respuesta, que sigue un modelo sigmoidal, de la que se extrae la región central, donde existe cierta linealidad y la pendiente es

máxima. Se ha observado que valores bajos de dispersión entre replicados afectan al cumplimiento de este criterio. Por esta razón se considera conveniente realizar esta “relajación” sobre el criterio de linealidad. Cabe destacar que, aunque se realice lo anterior, no se podrá indicar que el test de linealidad cumple según los criterios establecidos, con lo que los datos obtenidos en este caso deberían usarse con extrema cautela.

Al igual que en el caso anterior, para que funcione correctamente el conjunto de datos que se facilita como argumento, es necesario que previamente se adapte al formato requerido con la función *assay.data*. Además, para el correcto funcionamiento de esta función y con la finalidad de facilitar la elaboración de la misma se han generado tres funciones auxiliares.

Función *select.range* Descripción: Esta función devuelve todos los posibles rangos de dosis susceptibles de ser analizados en la función principal *best.model*. Argumentos: *data*: Toma como valor de entrada el conjunto de datos en formato matriz o *dataframe* que vayan a ser posteriormente analizados. El resultado de esta función es una *dataframe* con todos los posibles rangos, teniendo en cuenta:

- Mínimo de 3 niveles por curva dosis-respuesta
- Mismo número de niveles para cada curva.
- El desplazamiento entre la primera dosis elegida del estándar y el resto de muestras dependerá de las respuestas obtenidas. De tal forma que: se calculará la media de la respuesta por nivel (que equivale a la dosis) para la curva del estándar y se seleccionará el valor para el nivel intermedio. Posteriormente, se buscará cual es el nivel, de cada curva restante, en la que la media de respuesta difiere menos de la media en ese nivel intermedio. Ese será el desplazamiento máximo permitido. Esta consideración se ha realizado teniendo en cuenta lo indicado en el texto de referencia [4], donde se describe que lo más apropiado es que las curvas para muestras y estándar difieran en distancia horizontal entre sí lo mínimo (cuanto menos difieran, mejores serán las estimaciones obtenidas). Si por cualquier razón, la potencia pre-asignada a una muestra está infra- o sobre-estimada la curva se desplazará en el eje x más de lo deseado, de esta manera se corrige dicho desplazamiento y se reajusta la asignación inicial de potencia.

Función *check.model* Descripción: Esta función devuelve un valor booleano en función de si los datos de entrada cumplen el modelo propuesto. Se emplea la función anterior, *pl.model*, a un determinado conjunto de datos y esta función determina si los datos cumplen los criterios establecidos o no. Argumentos: *data*: Toma como valor de entrada el conjunto de datos en formato matriz o *dataframe*. *alpha*: valor determinado para la aceptación o no del criterio de linealidad. Por defecto es 0.05 como en el resto de test. El resultado de esta función es TRUE o FALSE en función de si el conjunto de datos cumplen o no los criterios establecidos. Para realizar dicha comprobación, dentro de la función se emplea la función *anova* de la librería *stats* (librería base de R).

4.2.4. Funciones cálculo de potencia

Una vez se ha determinado el cumplimiento o no del modelo propuesto, se puede realizar el cálculo de potencia estimada. Si el modelo no cumple los criterios anteriormente establecidos la determinación de potencia relativa y potencia estimada no se recomiendan.

Función *com.slope*

Descripción: la función calcula la pendiente común para todas las curvas dosis-respuesta proporcionadas.

Argumentos:

- *data*: El conjunto de datos sobre el que se quiere calcular la pendiente común.

El valor que devuelve dicha función es el valor para el término β o pendiente común a las curvas dosis-respuesta del conjunto de datos suministrados como argumento.

Función *rel.pot*

Descripción: la función calcula la potencia relativa de una muestra en relación con una muestra estándar.

Argumentos:

- *data*: El conjunto de datos sobre el que se quiere calcular la potencia relativa.
- *std*: El nombre asignado al estándar dentro de la variable *Sample* del conjunto de datos.
- *tst*: El nombre asignado a la muestra “problema” dentro de la variable *Sample* del conjunto de datos.
- *output.value*: el valor deseado de salida. La función calcula tanto la potencia relativa, como sus límites de confianza inferior y superior al 95 %. En este argumento se decide cuál de los 3 valores se pretende conocer. Por defecto está fijado en “*potency*” y calcula el valor de la potencia relativa, indicar “*lower*” o “*upper*” si se desean conocer los valores para el límite inferior o superior, respectivamente.

El valor que devuelve dicha función es el valor para la potencia relativa o sus límites de confianza se indique en los argumentos. La potencia relativa se expresa en función de la potencia

de la muestra estándar, es decir: la muestra “T” tiene una potencia “x” respecto del estándar. Para calcular la potencia en las unidades deseadas en función de la potencia asignada de la muestra se debe usar la función siguiente.

Función *est.pot*

Descripción: la función calcula la potencia estimada de una muestra en relación con una muestra estándar.

Argumentos:

- *data*: El conjunto de datos sobre el que se quiere calcular la potencia.
- *std*: El nombre asignado al estándar dentro de la variable *Sample* del conjunto de datos.
- *tst*: El nombre asignado a la muestra “problema” dentro de la variable *Sample* del conjunto de datos.
- *output.value*: el valor deseado de salida. La función calcula tanto la potencia, como sus límites de confianza inferior y superior al 95%. Por defecto está fijado en “*potency*” y calcula el valor de la potencia, indicar “*lower*” o “*upper*” si se desean conocer los valores para el límite inferior o superior, respectivamente.
- *factor*: un valor para el factor o potencia asignada de la muestra, que permita presentar el resultado en las unidades de potencia deseadas.
- *unit*: si se desea se pueden indicar en qué unidades se expresa la potencia de la muestra. Por ejemplo “mg/mL” o “UIgG/mL” Por defecto este argumento es nulo.

El valor que devuelve dicha función es el valor para la potencia biológica de la muestra o sus límites de confianza.

4.3. Documento RMarkdown para presentación de resultados

Con las funciones y herramientas que proporciona la librería desarrollada, se diseña un documento que realice el análisis propuesto en su conjunto y presente los resultados obtenidos. Para ello, se han tenido en cuenta los siguientes puntos:

- Documento con paginado, fecha-hora de impresión, información relativa al ensayo. Todo ello para asegurar la trazabilidad de los datos, siguiendo los principios de buenas prácticas de laboratorio.
- Gráficos con los datos antes y después del análisis.
- Resultado del análisis y valor para la potencia y su intervalo de confianza.

Teniendo en cuenta lo anterior se ha elegido un formato RMarkdown para diseñar dicho documento, ya que ofrece una gran versatilidad a la hora de presentar los resultados y al mismo tiempo se puede configurar fácilmente desde el IDE RStudio con el que se desarrolla el resto del trabajo.

4.3.1. *Script* RMarkdown

El código del documento *.rmd* se puede visualizar en el anexo D.

El documento se ha diseñado de tal forma que: tanto realiza el análisis de todos los datos de forma conjunta, o bien, de forma individual cada muestra o *test* frente al estándar; en función de lo indicado en el documento *.xlsx* donde se cargan los datos e información de partida.

Para el análisis de forma individual se forman subconjuntos de los datos obtenidos por cada muestra o *test*; de esta manera se analiza la muestra de forma individual frente al estándar y se generan tantos ajustes lineales y modelos como muestras contengan los datos.

Esto es especialmente útil si para una muestra concreta se observa un comportamiento extraño, o, por cualquier razón, se ha fallado en su análisis. De esta manera, si en un primer sondeo con un análisis de todos los datos de forma conjunta, se observa alguna interferencia; se puede decidir eliminar la muestra (o muestras) problemáticas del análisis. Otra opción, contemplada en la plantilla inicial, es analizar todos los datos pero de forma individual cada

muestra frente al estándar; para esto indicar en Rango de análisis: Toda la placa (individual) en la plantilla.

En lo que se refiere a gráficos, se ha decidido un gráfico que represente la relación entre dosis y respuesta tanto de los datos brutos obtenidos como, posteriormente, de los datos que forman parte del rango de dosis seleccionadas (y han sido filtrados, si así se indica) y subsiguiente ajuste lineal. Asimismo, se propone también el uso de un gráfico de residuos para analizar la calidad del ajuste de los datos al modelo propuesto.

4.3.2. Informe de resultados

El anterior código RMarkdown da como resultado un informe en formato *.pdf*, un ejemplo se puede observar en el anexo E. En este informe se puede observar como se han personalizado encabezados y pie de página, de tal forma que se incluya la fecha y hora de generación del informe así como el número de páginas totales en todas las hojas del documento.

La primera parte de este informe muestra los datos generales del ensayo realizado, que provienen de la plantilla inicial donde se vuelcan los datos obtenidos. Además de la ruta del *script* RMarkdown que genera dicho documento *.pdf*. Junto con los datos para la respuesta y la dosis se ha decidido incluir el valor para el CV entre réplicas como una primera aproximación a la dispersión de los datos obtenidos. De esta forma, si se decidiese filtrar los datos, con la función *filter.data* desarrollada para tal fin, también se podría ver cuales son los datos que la función va a modificar. En este caso sería deseable modificar el código de tal forma que imprima en este informe los datos iniciales que han resultado filtrados y los datos que resultan de la modificación realizada.

En una segunda parte se muestra el análisis propuesto según el modelo paralelo-lineal y la determinación de potencia.

Si el análisis es conjunto para todas las curvas dosis-respuesta obtenidas (como en el ejemplo mostrado) se podrá ver el rango de dosis (en forma de niveles) que han resultado adecuados para un ajuste paralelo-lineal y la tabla ANOVA del modelo. De esta tabla se extraen los 3 valores que resultan de interés: el valor de p para cada uno de los criterios que condicionan el cumplimiento del modelo. Seguidamente a la tabla, se muestra explícitamente el cumplimiento o no de dicho criterio. A continuación se muestra el valor para la pendiente común y los gráficos mencionados para cada muestra y su valor de potencia relativa y potencia estimada. También se muestran los intervalos de confianza del 95 % de ambos valores.

En el caso de que el análisis se haga de forma individual para cada muestra: el rango de dosis, la tabla ANOVA y la pendiente común será diferente para cada agrupación Estándar-Muestra. El resto del documento seguirá la misma estructura que la mostrada aquí.

4.4. Validación de la solución propuesta y discusión final

La librería diseñada, “*plm.crd*”, además de contener toda la información expuesta, tiene ejemplos para el uso de cada una de las funciones que ayuda a entender el funcionamiento de cada una de ellas. Además, se ha diseñado de tal forma que los datos obtenidos y su empleo sea fácilmente personalizable en función de la finalidad con la que se utilicen. La actual solución que presenta **R** para realizar este análisis, la librería *pla* [5]; a parte de no disponible para las nuevas versiones de **R**, presenta una salida muy poco personalizable, en ocasiones incluso difícil de interpretar. Por otra parte, a diferencia de la solución propuesta aquí, la librería *pla* considera otros diseños de ensayo, además del diseño completamente aleatorizado (*crd*); con lo que es una solución más extensa en ese sentido.

Es mucha la bibliografía que hace referencia a la determinación de potencia en bioensayos, tanto con aplicaciones del modelo paralelo lineal [15], [16], [17], [23], [40], [41]; o, en otros casos, con aplicaciones del modelo logístico de cuatro parámetros [22], [26]. Sin embargo, es difícil encontrar un texto realmente explicativo de la forma concreta de llevar a cabo este análisis desde el punto de vista estadístico. En [25] se hace una explicación estadística homóloga a la que se encuentra en [4] y que ha sido empleada en este trabajo.

En lo que se refiere a validación de los resultados obtenidos con el análisis propuesto, si se comparan los resultados obtenidos con el cálculo manual, resumidos en la tabla 3.3, con los que figura en el informe generado del análisis propuesto, anexo E. Se puede observar que los resultados para cada muestra son idénticos. Además, resultan idénticos a los resultados obtenidos del análisis con la librería “*pla*” mencionada.

Es necesario indicar que para la validación no se ha empleado la función *filter.data* propuesta para el filtrado de datos brutos. Esto se debe a que en el conjunto de datos presentado como ejemplo para la validación, no se realiza dicho paso y, de realizarlo, los datos no coincidirían con el cálculo manual mostrado en el anexo A. Para otros casos, en conjuntos de datos reales puede resultar de interés dicha función y en tal caso, sería necesario modificar el código del archivo RMarkdown (*.rmd*) presentado, para que realizase dicha función.

También resulta conveniente puntualizar que, aunque se emplea la función *best.model* (que proporciona el rango de dosis/niveles que se ajustan al modelo paralelo lineal), en este ejemplo su uso es meramente ilustrativo, ya que no modifica el rango inicial de dosis empleadas (puesto que la totalidad del rango proporcionado se ajusta al modelo). De todas formas, se ha querido incluir para aumentar la versatilidad del análisis y que realice dicha función en caso necesario.

En resumen, el informe que se genera es un modelo (ejemplo) para la casuística particular de los datos empleados en este texto. Este modelo se debe personalizar en función de la naturaleza

de los datos de origen y/o las preferencias del usuario para añadir información considerada relevante o por cualquier otra causa que lo requiera.

Finalmente, dado que el diseño para la entrada de datos propuesto nos permite elegir el rango de análisis (muestras ensayadas que se incluirán en el análisis) en la siguiente tabla 4.1 se muestran los resultados para la potencia estimada de las tres muestras; obtenidos en el análisis de todas las curvas dosis-respuesta de forma conjunta (rango de análisis: “toda la placa”) o de cada muestra frente al estándar de forma individual (rango de análisis: “toda la placa (individual)”). En esta tabla 4.1 se combinan también estos resultados con los de la tabla 3.3 provenientes del cálculo manual.

Tabla 4.1: Resultados de potencia según el método seguido

	Vacuna T	Vacuna U	Vacuna V
Potencia según cálculo manual	43.4	35.2	39.4
Potencia según análisis conjunto	43.4	35.2	39.4
Potencia según análisis individual	44.0	35.2	39.6

Como se puede observar, y como ya se había mencionado, el cálculo manual y el análisis de todas las muestras de forma conjunta arroja exactamente los mismos resultados, puesto que el cálculo manual propuesto también analiza todas las muestras de forma conjunta. En el caso de analizar las muestras de forma individual los resultados varían ligeramente.

Capítulo 5

Conclusiones y trabajos futuros

5.1. Conclusiones

A lo largo de este documento se presenta una solución para la determinación de potencia aplicado a datos de inmunoensayo. Se ha empleado el modelo paralelo-lineal propuesto por la bibliografía [4], [19] [15], [25]. El caso particular de los datos de inmunoensayo presenta un reto en cuanto que los datos, generalmente, no se ajustan al modelo paralelo-lineal. Por ello se ha diseñado una función (*best.model*) para la elección del rango lineal dentro de dos o más curvas dosis-respuesta.

Por otra parte la elección de un formato de entrada de los datos en forma de plantilla/hoja de cálculo *.xlsx* facilita la utilización de esta solución por parte de usuarios sin experiencia en **R**. De igual manera que la presentación de los resultados en un formato *.pdf* facilita su interpretación y dificulta una manipulación no deseada de los mismos. Además, el archivo generado ofrece una apariencia de informe con lo que fácilmente se puede emplear sin necesidad de edición posterior, cualquiera que sea su ámbito de aplicación.

Además, como ejemplo para la presentación de las soluciones propuestas se han empleado datos provenientes de la bibliografía, que a su vez sirven para la validación del mismo, observando que los cálculos y funciones propuestas coinciden con los cálculos manuales presentados [4].

Inicialmente se había propuesto la generación de un enlace entre Excel y **R**; de tal forma que, con una acción simple en Excel (como por ejemplo: pulsar un botón), se ejecutase el análisis de forma automática en **R** y se generase el documento *.pdf* con los resultados del análisis realizado. La metodología seguida para la consecución de este objetivo fue el empleo de código de Visual Basic para Aplicaciones (VBA) que ejecutase R y el Script en cuestión. Sin embargo, este punto no ha sido posible llevarlo a cabo por el momento; con lo que, actualmente es necesario especificar el nombre del archivo Excel de origen en el archivo *.rmd* de RMarkdown para que

se ejecute el análisis y se genere el documento con los resultados. Esto, interfiere con uno de los objetivos propuestos y además implica una mínima formación del entorno **R** o RStudio. Debido a esto, se han generado dos funciones que fácilmente extraen la información desde Excel (*load.info* y *load.data*).

Asimismo, aunque el presente proyecto se ha realizado con la finalidad de presentar un modelo de análisis y determinación de potencia para el caso de datos procedentes de inmunoensayos, la solución obtenida se puede generalizar para datos de cualquier bioensayo en el que la finalidad sea obtener una potencia estimada de una muestra desconocida en función de un estándar de referencia.

Las soluciones aportadas hasta el momento para abordar la determinación de la potencia biológica a partir de datos de inmunoensayo con el modelo paralelo-lineal, se centran: o bien en la metodología, [4], [25]; o en las funciones específicas que hacían posible el análisis [5]. La solución aquí planteada integra todos los pasos una vez se obtienen los datos del ensayo y, en este sentido, conlleva una propuesta novedosa, de fácil aplicación y completa.

5.2. Limitaciones del estudio

A continuación se describen los factores que pueden influir en la generalización los resultados presentados.

En este texto se propone la determinación de la potencia biológica a través del modelo paralelo-lineal, para lo cual es necesario una selección previa de aquellos datos obtenidos que cumplen con la premisa de linealidad. Esto se ha hecho así puesto que es una de las aproximaciones propuestas en la bibliografía más implementadas y documentadas. Además, el estudio de las curvas dosis-respuesta en su totalidad exige un análisis más complejo y no tan generalizable como la propuesta que aquí se describe. Sin embargo, la no utilización de parte de los datos obtenidos puede no ser deseable en algunos casos, haciendo necesario el uso de modelos más complejos como el modelo logístico de cuatro parámetros.

Asimismo, la solución que ofrece este texto se limita al caso de un diseño de ensayo completamente aleatorizado. Otros diseños son posibles y es necesario adaptar la solución propuesta para la determinación de potencia según el diseño elegido.

5.3. Líneas de futuro

Tal y como se indica previamente, una de las tareas pendientes de este trabajo es determinar si existe alguna forma de automatizar el análisis en **R** desde Excel y evitar el paso intermedio entre las dos herramientas. Una de las soluciones que podrían resolver este problema es la

inclusión del código **R** directamente en VBA, de tal manera que no resulte necesario ejecutar el programa **R** como tal, queda pendiente estudiar si esta solución es aplicable para la casuística concreta descrita.

Otra línea de investigación pendiente es la extensión de la librería actual para el estudio de otros diseños de ensayo, como puede ser el diseño aleatorizado por bloques, en cuadrados latinos o bien el diseño *cross-over* aumentaría la versatilidad de la librería elaborada y, además, serviría como alternativa total a la librería *pla* descatalogada.

Además, uno de los objetivos opcionales propuestos era el estudio del modelo logístico de cuatro parámetros para curvas dosis-respuesta extendidas. Con este modelo, no se seleccionarían las dosis dentro del rango lineal si no que, se emplearía la totalidad del rango de dosis ensayadas reduciendo la manipulación y tratamiento de los datos obtenidos. Resultaría de gran interés explorar esta línea de trabajo para aportar una alternativa al análisis aquí propuesto.

5.4. Seguimiento de la planificación

Para la realización de este trabajo se pretendía limitar a lo planificado inicialmente. Sin embargo, a lo largo del proyecto han surgido diversos problemas así como nuevas ideas que no han hecho posible seguir al nivel deseado la planificación propuesta.

En la última etapa, se había dejado tiempo suficiente para un posible estudio de la parte opcional de este proyecto. Las desviaciones temporales sucedidas se han mitigado eliminando esta parte opcional.

Capítulo 6

Glosario

6.1. Abreviaturas

ANOVA: Análisis de la varianza (*Analysis of variance*).

AP: Fosfatasas alcalinas (*Alkaline phosphatases*).

cdr: diseño completamente aleatorizado (*completely randomized design*)

CRAN: Red integral de archivos de R (*Comprehensive R Archive Network*).

CV: Coeficiente de Variación.

EDQM: Dirección Europea para la Calidad de los Medicamentos y la Atención Sanitaria (*European Directorate for the Quality of Medicines and HealthCare*).

ED_{50} : dosis media efectiva.

EIA: Inmunoensayo enzimático (*Enzyme Immunoassay*).

ELISA: ensayo por inmunoadsorción ligado a enzimas (*Enzyme-Linked ImmunoSorbent Assay*).

HRP: Peroxidasa de rábano picante. (*Horseradish peroxidase*).

IDE: Entorno de desarrollo integrado (*Integrated development environment*).

IgE: Inmunoglobulina E.

IgG: Inmunoglobulina G.

ln: Logaritmo neperiano.

nm: nanómetros.

OD: densidad óptica (*Optical density*).

ONU: Organización de las Naciones Unidas.

pdf: *Portable Document Format*.

Ph.Eur.: European Pharmacopoeia.

PLM: Modelo paralelo lineal (*parallel-line model*).

RIA: Radioinmunoensayo (*radioimmunoassay*).

ROC: Característica operativa del receptor (*Receiver Operating Characteristic*).

VBA: Visual Basic para aplicaciones (*Visual Basic for Application*).

6.2. Definiciones

Analito: Sustancia química presente en un material o sistema, que es objeto de detección, identificación y cuantificación en el análisis químico de una muestra [42].

Anticuerpo: o inmunoglobulina, es una proteína elaborada por las células plasmáticas (glóbulos blancos) en respuesta a un antígeno. El anticuerpo se une al antígeno para ayudar a destruirlo o, en ocasiones, marcarlo para su posterior destrucción [43].

Antígeno: cualquier sustancia que provoca la producción de anticuerpos específicos (respuesta inmune) en un animal. Cualquier molécula: proteína, azúcar, sustancia química, etc. puede considerarse antígeno si genera una respuesta inmune [44].

Enzima: proteína que realiza y/o modifica la velocidad de una reacción bioquímica en un organismo.

Epitopo: cada uno de los sitios de unión que presenta un antígeno y que son reconocidos de forma individual por un anticuerpo específico [12].

Potencia biológica/estimada: potencia calculada con el análisis de los datos del ensayo [4].

Script: secuencia de comandos, fragmentos de código con una funcionalidad específica.

Valor booleano: en ciencia computacional, representa un valor de lógica binaria, generalmente verdadero o falso.

Capítulo 7

Bibliografía

- [1]S. K. Vashist y J. H. Luong, “Immunoassays: an overview”, *Handbook of Immunoassay Technologies*, págs. 1-18, 2018.
- [2]J. Geigert, “Indispensable Potency (Biological Activity)”, en *The Challenge of CMC Regulatory Compliance for Biopharmaceuticals*, Springer, 2023, págs. 407-430.
- [3]K. L. Cox, V. Devanarayan, A. Kriauciunas, J. Manetta, C. Montrose y S. Sittampalam, “Immunoassay methods”, *Assay Guidance Manual [Internet]*, 2019.
- [4]EDQM Council of Europe, “5.3 Statistical Analysis of Results of Biological Assays and Tests”, *European Pharmacopoeia, 11st Edition*, 2023.
- [5]J. H. Badsberg, *pla: Parallel Line Assays*, R package version 0.2, 2015. dirección: <https://CRAN.R-project.org/package=pla>.
- [6]S. Aydin, “A short history, principles, and types of ELISA, and our laboratory experience with peptide/protein analyses using ELISA”, *Peptides*, vol. 72, págs. 4-15, 2015.
- [7]R. M. Lequin, “Enzyme immunoassay (EIA)/enzyme-linked immunosorbent assay (ELISA)”, *Clinical chemistry*, vol. 51, n.º 12, págs. 2415-2418, 2005.
- [8]T. Porstmann y S. Kiessig, “Enzyme immunoassay techniques an overview”, *Journal of immunological methods*, vol. 150, n.º 1-2, págs. 5-21, 1992.
- [9]D. W. Chan, “General principle of immunoassay”, *Immunoassay: a practical guide*, págs. 1-23, 1987.
- [10]K. Shah y P. Maghsoudlou, “Enzyme-linked immunosorbent assay (ELISA): the basics”, *British journal of hospital medicine*, vol. 77, n.º 7, págs. C98-C101, 2016.
- [11]G. N. Konstantinou, “Enzyme-linked immunosorbent assay (ELISA)”, *Food Allergens: Methods and Protocols*, págs. 79-94, 2017.

- [12] J. M. Walker y J. Crowther, “The ELISA guidebook”, en *Series Springer Protocols. Methods in Molecular Biology*, vol. 516, Springer, 2009.
- [13] A. V. Lin, “Direct ELISA”, *ELISA: Methods and Protocols*, págs. 61-67, 2015.
- [14] A. V. Lin, “Indirect Elisa”, *ELISA: methods and protocols*, págs. 51-59, 2015.
- [15] T. Mzolo, “Statistical methods for the analysis of bioassay data”, English, Proefschrift, Phd Thesis 1 (Research TU/e / Graduation TU/e), Mathematics y Computer Science, abr. de 2016, ISBN: 978-90-386-4051-8.
- [16] Z. Govindarajulu, *Statistical techniques in bioassay*. Karger Medical y Scientific Publishers, 2001.
- [17] A. DeLean, P. J. Munson y D. Rodbard, “Simultaneous analysis of families of sigmoidal curves: application to bioassay, radioligand assay, and physiological dose-response curves.”, *American Journal of Physiology-Endocrinology And Metabolism*, vol. 235, n.º 2, E97-E102, 1978.
- [18] D. J. Finney, “The principles of biological assay”, *Supplement to the Journal of the Royal Statistical Society*, vol. 9, n.º 1, págs. 46-91, 1947.
- [19] D. J. Finney et al., *Statistical method in biological assay*. Charles Griffin & Company., 1978.
- [20] M. J. Healy, “Statistical analysis of radioimmunoassay data”, *Biochemical Journal*, vol. 130, n.º 1, pág. 207, 1972.
- [21] S. Z. Knezevic, J. C. Streibig y C. Ritz, “Utilizing R software package for dose-response studies: the concept and data analysis”, *Weed Technology*, vol. 21, n.º 3, págs. 840-848, 2007.
- [22] J. N. Jonkman y K. Sidik, “Equivalence testing for parallelism in the four-parameter logistic model”, *Journal of biopharmaceutical statistics*, vol. 19, n.º 5, págs. 818-837, 2009.
- [23] S. Novick y H. Yang, “A fast and reliable test for parallelism in bioassay”, *Journal of biopharmaceutical statistics*, vol. 29, n.º 6, págs. 1011-1023, 2019.
- [24] D. J. Finney, “Bioassay and the Practice of Statistical Inference”, *International Statistical Review / Revue Internationale de Statistique*, vol. 47, n.º 1, págs. 1-12, 1979.
- [25] R. Mohamed y H. Mariam, “Using Completely Randomized Design of Parallel Linear Model for Estimating the Biological Potency of Human Insulin Drugs: An Empirical Study”, *Biostatistics and Biometrics Open Access Journal*, vol. 3, n.º 4, págs. 115-121, 2017.
- [26] A. VØlund, “Application of the four-parameter logistic model to bioassay: comparison with slope ratio and parallel line models”, *Biometrics*, págs. 357-365, 1978.
- [27] A. B. Kay, “Allergy and allergic diseases”, *New England Journal of Medicine*, vol. 344, n.º 1, págs. 30-37, 2001.
- [28] F. J. Sola Martinez, R. M. Barranco Jimenez, C. Martin Garcia et al., “Intradermal Phleum pratense allergoid immunotherapy. Double-blind, randomized, placebo-controlled trial”, *Clinical & Experimental Allergy*, vol. 50, n.º 12, págs. 1352-1361, 2020.

- [29] M. Gallego, V. Iraola, M. Himly et al., “Depigmented and polymerised house dust mite allergoid: allergen content, induction of IgG4 and clinical response”, *International archives of allergy and immunology*, vol. 153, n.º 1, págs. 61-69, 2010.
- [30] D. Larenas-Linnemann, L. S. Cox et al., “European allergen extract units and potency: review of available information”, *Annals of Allergy, Asthma & Immunology*, vol. 100, n.º 2, págs. 137-145, 2008.
- [31] M. Casanovas, F. Marañón e I. Bel, “Comparison of skin-prick test assay and reverse enzyme immunoassay competition (REINA-C) for biological activity of allergens”, *Clinical & Experimental Allergy*, vol. 24, n.º 2, págs. 134-139, 1994.
- [32] L. Boluda, M. Gallego, M. G. Romano, O. Cambra y L. Berrens, “Evaluation of the potency of allergenic extracts by inhibition of IgG-antibody binding”, *Biologicals*, vol. 22, n.º 2, págs. 171-177, 1994.
- [33] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2020.
- [34] M. Logan, *Biostatistical design and analysis using R: a practical guide*. John Wiley & Sons, 2011.
- [35] P. Schaubergger y A. Walker, *openxlsx: Read, Write and Edit xlsx Files*, R package version 4.2.5.2, 2023. dirección: <https://CRAN.R-project.org/package=openxlsx>.
- [36] H. Wickham, J. Hester, W. Chang y J. Bryan, *devtools: Tools to Make Developing R Packages Easier*, R package version 2.4.5, 2022. dirección: <https://CRAN.R-project.org/package=devtools>.
- [37] H. Wickham, P. Danenberg, G. Csárdi y M. Eugster, *roxygen2: In-Line Documentation for R*, R package version 7.2.3, 2022. dirección: <https://CRAN.R-project.org/package=roxygen2>.
- [38] F. Carmona, “Modelos lineales”, *Pub. Univ. de Barcelona, Barcelona*, 2005.
- [39] F. Carmona, “Análisis de la Covarianza con R”, [Acceso: 30/12/2023]. dirección: <https://erre-que-erre-paco.blogspot.com/2013/12/analisis-de-la-covarianza-con-r.html>.
- [40] M. Eissa, E. R. Rashed y D. E. Eissa, “Validation of symmetrical two-dose parallel line assay model for nystatin potency determination in pharmaceutical product”, *Journal of Advanced Pharmacy Research*, vol. 5, n.º 4, págs. 406-413, 2021.
- [41] G. D. Djira, “Relative potency estimation in parallel-line assays—method comparison and some extensions”, *Communications in Statistics—Theory and Methods*, vol. 39, n.º 7, págs. 1180-1189, 2010.
- [42] Real Academia Española, *Analito*, [Acceso: 28/12/2023]. dirección: <https://dle.rae.es/analito>.

- [43] Instituto Nacional del cáncer (EEUU), *Anticuerpo*, [Acceso: 28/12/2023]. dirección: <https://www.cancer.gov/espanol/publicaciones/diccionarios/diccionario-cancer/def/anticuerpo>.
- [44] R. Hnasko, *Elisa*. Springer, 2015.
- [45] manualdelatex.com, *Crear una bibliografía en LaTeX*, [Acceso: 19/10/2023]. dirección: <https://manualdelatex.com/tutoriales/bibliografia>.
- [46] M. Amelot, *VBA Excel 2016: programación en Excel: macros y lenguaje VBA*. Ediciones ENI, 2016.
- [47] A. Arenas, E. Labarga y J. L. Varona, “Una breve guía para redactar un trabajo de fin de grado con contenido matemático utilizando LaTeX”,
- [48] H. Wickham y J. Bryan, *R packages*. O’Reilly Media, Inc., 2023.
- [49] D. A. Schmitt, H. Cheng, S. J. Maleki y A. W. Burks, “Competitive inhibition ELISA for quantification of Ara h 1 and Ara h 2, the major allergens of peanuts”, *Journal of AOAC International*, vol. 87, n.º 6, págs. 1492-1497, 2004.

Apéndice A

Anexo: Datos de ejemplo para validación

Los datos para la validación de la solución propuesta se pueden consultar en la siguiente tabla A.1. Tal y como se indica en el apartado 3.3.1, se trata de un bioensayo en el que el objetivo es determinar la potencia biológica de 3 vacunas para la Hepatitis B frente a un estándar; datos que se muestran como ejemplo en la referencia [4].

La primera columna se refiere a la respuesta (el valor de Y), medida en unidades de densidad óptica (DO). La segunda indica la muestra a la que pertenecen esos datos, en este caso hay 4 muestras, STD se refiere al Estándar y las siguientes a las muestras T, U y V. La tercera columna indica la dosis, el valor de x. Hasta esta tercera columna son los datos proporcionados.

Para una mayor facilidad a la hora de trabajar con estos datos se han añadido la cuarta y quinta columna. La cuarta columna hace referencia al replicado: para cada valor de dosis (x) y muestra se han realizado tres medidas diferentes, estas tres medidas se corresponden a los valores 1/2/3 que figura en esta columna. La quinta columna es el nivel, entendido como tal el número de valores que toma la variable dosis (en este caso van de 1 a 5). Estas dos columnas serán de utilidad para agrupar los datos de determinada forma según las necesidades específicas en cada momento.

Tabla A.1: Datos ejemplo para validación. Datos facilitados por la bibliografía como ejemplo para el análisis y determinación de potencia con el modelo paralelo-lineal.

Respuesta	Muestra	Dosis	Replicado	Nivel
0.04300	STD	0.00125	1	1
0.09300	STD	0.00250	1	2
0.15900	STD	0.00500	1	3
0.28300	STD	0.01000	1	4
Continúa en la siguiente página				

Continuación de la tabla A.1				
Response	Sample	Dose	Replicate	Level
0.51400	STD	0.02000	1	5
0.04500	STD	0.00125	2	1
0.09900	STD	0.00250	2	2
0.15400	STD	0.00500	2	3
0.29500	STD	0.01000	2	4
0.53100	STD	0.02000	2	5
0.05100	STD	0.00125	3	1
0.08200	STD	0.00250	3	2
0.16600	STD	0.00500	3	3
0.36200	STD	0.01000	3	4
0.54500	STD	0.02000	3	5
0.09700	TST	0.00125	1	1
0.16700	TST	0.00250	1	2
0.32700	TST	0.00500	1	3
0.50100	TST	0.01000	1	4
1.14000	TST	0.02000	1	5
0.09700	TST	0.00125	2	1
0.15700	TST	0.00250	2	2
0.35500	TST	0.00500	2	3
0.66500	TST	0.01000	2	4
1.38600	TST	0.02000	2	5
0.09400	TST	0.00125	3	1
0.17800	TST	0.00250	3	2
0.34500	TST	0.00500	3	3
0.57600	TST	0.01000	3	4
1.05100	TST	0.02000	3	5
0.08600	UST	0.00125	1	1
0.12700	UST	0.00250	1	2
0.27700	UST	0.00500	1	3
0.58600	UST	0.01000	1	4
0.95700	UST	0.02000	1	5
0.07100	UST	0.00125	2	1
0.14600	UST	0.00250	2	2
0.26800	UST	0.00500	2	3
0.48900	UST	0.01000	2	4
0.86600	UST	0.02000	2	5
0.07300	UST	0.00125	3	1
0.13300	UST	0.00250	3	2
0.26900	UST	0.00500	3	3
0.54600	UST	0.01000	3	4
1.04500	UST	0.02000	3	5
0.08200	VST	0.00125	1	1

Continúa en la siguiente página

Continuación de la tabla A.1				
Response	Sample	Dose	Replicate	Level
0.14500	VST	0.00250	1	2
0.31800	VST	0.00500	1	3
0.55200	VST	0.01000	1	4
1.03700	VST	0.02000	1	5
0.08200	VST	0.00125	2	1
0.14400	VST	0.00250	2	2
0.30600	VST	0.00500	2	3
0.55100	VST	0.01000	2	4
1.03900	VST	0.02000	2	5
0.08600	VST	0.00125	3	1
0.17300	VST	0.00250	3	2
0.31600	VST	0.00500	3	3
0.62400	VST	0.01000	3	4
1.06800	VST	0.02000	3	5
Final de la tabla				

En **R** se empleará el valor de la respuesta transformado, el logaritmo de la respuesta con lo que aquí se pueden obtener los valores medios de cada nivel de respuesta y para cada muestra, tabla A.2:

Tabla A.2: Medias de las ln(DOs) por muestra y nivel

	STD	TST	UST	VST
Nivel 1	-3.075	-2.344	-2.572	-2.485
Nivel 2	-2.396	-1.789	-2.002	-1.874
Nivel 3	-1.835	-1.073	-1.305	-1.161
Nivel 4	-1.166	-0.550	-0.618	-0.554
Nivel 5	-0.635	0.169	-0.048	0.047

Los valores necesarios para la construcción de la tabla ANOVA son P (ecuación 3.1), L (ecuación 3.2), H_L (ecuación 3.4) y H_P (ecuación 3.3). En este caso son los siguientes:

$$\begin{aligned}
 P_S &= -9.108 & L_S &= 6.109 \\
 P_T &= -5.586 & L_T &= 6.264 \\
 P_U &= -6.544 & L_U &= 6.431 \\
 P_V &= -6.027 & L_V &= 6.384
 \end{aligned}$$

$$H_P = \frac{3}{5} = 0.6 \quad H_L = \frac{36}{120} = 0.3$$

Tabla A.3: Tabla ANOVA con los datos de ejemplo

Fuente de varianza	Grados de libertad	Suma de cuadrados	Media de cuadrados	F-ratio	Prob.
Preparación	3	4.475	1.492		
Regresión	1	47.58	47.58	7126	0.0000
No-paralelismo	3	0.0187	0.006	0.933	0.434
No-linealidad	12	0.0742	0.006	0.926	0.531
Tratamientos	19	52.152			
Error Residual	40	0.267	0.0067		
Total	59	52.42			

La tabla ANOVA (Tabla A.3) con los datos anteriores sería:

Por tanto, el test resulta significativo para el término de regresión y no significativo para paralelismo y linealidad. Por tanto el valor para la pendiente común, según la ecuación 3.6, es:

$$\beta = \frac{0,3 \times (6,109 + 6,264 + 6,431 + 6,384)}{\ln 2 \times 3 \times 4} = 0,90848$$

Por último, el valor de la potencia y su intervalo de confianza, en este caso de la muestra T viene dado por:

$$\ln(\text{Potencia}_{\text{estimada}})_T = M'_T = \frac{-5,586 - (-9,108)}{5 \times 0,90848 = 0,7752}, e^{0,7752} = 2,171$$

$$C = \frac{47,58}{47,58 - 0,0067 \times 2,021^2} = 1,00057 \quad \text{y} \quad V = \frac{47,58}{0,90848^2 \times 5 \times 3} = 3,8436$$

$$\text{CI} = 1.00057 \times 0,7752 \pm \sqrt{(0,00057)(1,00057 \times 0,7752^2 + 2 \times 3,8436)} = 0,7756 \pm 0,0689$$

$$\text{CI} = e^{0,7067} - e^{0,8445} = 2,027 - 2,327$$

En el texto indican que tanto muestras como estándar tienen una potencia asignada de 20 µg proteína/mL. En este caso para equiparar la potencia de la muestra a las unidades del estándar será necesario aplicar este factor de 20 a la potencia relativa obtenida. Por lo tanto,

la potencia estimada de la vacuna T es de 40.5 μg proteína/mL con un intervalo de confianza del 95 % entre 43.4 y 46.5.

Siguiendo el mismo procedimiento la vacuna U tiene una potencia de 32.9 μg proteína/mL (35.2 - 37.6) y la vacuna V tiene una potencia de 36.8 μg proteína/mL (39.4 - 42.2).

Apéndice B

Anexo: Plantilla entrada de datos obtenidos

A continuación se muestra un modelo de plantilla para la carga de los datos de ensayo.

Usuario - Fecha	R. Lorenzo - 20/12/2023
Tipo de ensayo	Otro
Antígeno	Hepatitis
Condiciones Ensayo	Bioensayo XXX

Dilución Inicial	0,0000625
Ratio Dilución	0,5
Número de muestras/placa	3
Rango de análisis	Toda la placa

STD posición	1
STD Identificación	Estándar
STD Unidades	µg/mL
STD potencia asignada	20

TST posición	2
TST Identificación	Vacuna T
TST Unidades	µg/mL
TST potencia asignada	20

UST posición	3
UST Identificación	Vacuna U
UST Unidades	µg/mL
UST potencia asignada	20

VST posición	4
VST Identificación	Vacuna V
VST Unidades	µg/mL
VST potencia asignada	20

Respuesta	STD			TEST T			TEST U			TEST V		
	1	2	3	4	5	6	7	8	9	10	11	12
A	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
B	0,043	0,097	0,086	0,082	0,045	0,097	0,071	0,082	0,051	0,094	0,073	0,086
C	0,093	0,167	0,127	0,145	0,099	0,157	0,146	0,144	0,082	0,178	0,133	0,173
D	0,159	0,327	0,277	0,318	0,154	0,355	0,268	0,306	0,166	0,345	0,269	0,316
E	0,283	0,501	0,586	0,552	0,295	0,665	0,489	0,551	0,362	0,576	0,546	0,624
F	0,514	1,140	0,957	1,037	0,531	1,386	0,866	1,039	0,545	1,051	1,045	1,068
G												
H												

Apéndice C

Anexo: Código R librería plm.crd

C.1. Función *load.info*

```
#' @title Load file information
#' @description Function to load assay additional information from excel file
#' to use in plm analysis. Only valid to single named cells.
#' @param filename Complete name of the excel file from which
#' information is to be extracted
#' @return all info located in single cells with 'named region'
#' configuration in excel
#' @export

load.info <- function(filename){

  nr <- openxlsx::getNamedRegions(filename)

  for (i in nr) {
    j <- openxlsx::readWorkbook(filename, namedRegion= i, colNames=F)[1,1]
    assign(i, j, envir = .GlobalEnv)
  }
}
```

Código C.1: Función load.info

C.2. Función *load.data*

```
#' @title Load file data
#' @description Function to load assay data from excel file to use in plm
  analysis
#' Only valid to named regions.
#' @param filename Complete name of the excel file from which information is to
  be extracted
#' @param data name of the named region that contains the data.
#' @return all cells range, which contain the data for the analysis.
#' @export

load.data <- function(filename , data){

  rawData <- openxlsx::readWorkbook(filename , namedRegion= data , colNames=T)
  assign("rawData" , rawData , envir = .GlobalEnv)
}
```

Código C.2: Función load.data

C.3. Función *assay.data*

```

#' @title Assay data preparation
#' @description Function to give a specific format to the data
#' @param data Complete name of the excel file from which information is to be
  extracted
#' @param y.treat if the answer (y) requires treatment indicate which.
  NULL by default
#' @param dilution the variable dilution. By default is equal to:
  dilutionRatio+log(Dose)/log(dilutionRatio). If not it should be a vector
  of equal length to data
#' @param dilutionRatio the ratio between dilutions By default is equal to: 1/2
#' @return data with the required changes
#' @export
#' @examples
#' assay.data(hepatitis_example, y.treat = log)
#' assay.data(hepatitis_example, y.treat = square)

assay.data <- function(data, y.treat = NULL, dilution = NULL,
  dilutionRatio = 0.5 ){

  if (!is.null(y.treat)){
    data$Response <- sapply(data$Response, FUN = y.treat)}

  if (is.null(dilution)){
    data <- cbind(data,
      Dilution = dilutionRatio+log(data$Dose)/log(dilutionRatio))}
  else{
    data <- cbind(data, Dilution = dilution)
  }
  data$Dose <- data$Dose[data$Sample == "STD"]
  return(data)
}

```

Código C.3: Función assay.data

C.3.1. Función auxiliar *square*

```

#' @title Square
#' @description This function Calculates the square of a given number,
#' @param x number/vector to calculate the square
#' @return the square of the given number
#' @examples
#' square(3)
#' square(1:3)
#' @export

square <- function(x) {
  return(x^2)
}

```

Código C.4: Función square

C.4. Función *filter.data*

```

#' @title Filtered data based on CV
#' @description Calculates the data filtered from a dataframe based on the
#' coeficiente of variation up to a maximum of 2 global data per sample or group
#' @param data dataframe to filter the response variable
#' @return Returns the same input dataframe with some filtered data changes
#' @examples
#' filter.data(hepatitis_example)
#' @export

filter.data <- function (data){

  groupnames <- unique(data$Sample)
  subgroups <- list()

  for (i in groupnames){
    subg <- matrix(data$Response[data$Sample == i],
                  nrow = max(data$Level),
                  ncol = max(data$Replicate))
    subg <- data.frame(subg)
    colnames(subg) <- c(1:ncol(subg))
    subgroups[[i]] <- subg
  }

  for (j in groupnames){
    subgroups[[j]] <- cbind(subgroups[[j]],
                          cv = apply(subgroups[[j]],
                                     MARGIN = 1,
                                     FUN = cv.calc),
                          max_dif = apply(subgroups[[j]],
                                         MARGIN = 1,
                                         FUN = pos.max.diff))

    for (k in 1:nrow(subgroups[[j]])){

      pos <- subgroups[[j]][k,5]
      if (subgroups[[j]][k,4] > 15 &
          (subgroups[[j]][k,4] == max(subgroups[[j]][,4]) |
           subgroups[[j]][k,4] == sort(subgroups[[j]][,4],
                                       decreasing = TRUE)[2])){
        val <- subgroups[[j]][k,1:(ncol(subgroups[[j]))-2)]
        subgroups[[j]][k,pos] <- median(unlist(val[1,-pos]))
      }
    }

    subgroups[[j]] <- subgroups[[j]][,1:(ncol(subgroups[[j]))-2)]
  }

  filter_data <- data
  filter_data$Response <- unlist(subgroups)
  return(filter_data)
}

```

Código C.5: Función *filter.data*

C.4.1. Función auxiliar *cv.calc*

```

#' @title Coefficient of variation
#' @description Function to calculates the coefficient of variation with the
  mean or median of a numerical vector
#' @param vector vector to calculate the cv
#' @param FUN if cv is calculated with mean or median. Mean by default
#' @return coefficient of variation of the data in percentage
#' @examples
#' cv.calc(c(105,205,150,175))
#' cv.calc(c(1:450), FUN = "median")
#' @export

cv.calc <- function(vector, FUN = "mean"){
  sd_vc <- sd(vector)
  if (FUN == "mean"){
    m_vc <- mean(vector)
  } else if (FUN == "median"){
    m_vc <- median(vector)
  }
  cv <- (sd_vc / m_vc) * 100
  return(cv)
}

```

Código C.6: Función cv.calc

C.4.2. Función auxiliar *pos.max.diff*

```

#' @title Position with maximum difference
#' @description Function that returns the position of the value with the maximum
  difference with respect to the calculated mean or median of the given
  numerical vector
#' @param vector vector to calculate the position with the maximum difference
#' @param FUN if value of max. difference is calculated with mean or median.
  Mean by default
#' @return The position of the value with maximum difference
#' @examples
#' pos.max.diff(c(100,105,405))
#' pos.max.diff(c(1:450), FUN = "median")
#' @export

pos.max.diff <- function(vector, FUN = "mean") {
  if (FUN == "mean"){
    desv <- abs(vector - mean(vector))
  } else if (FUN == "median"){
    desv <- abs(vector - median(vector))
  }
  maxim <- which.max(desv)
  return(maxim)
}

```

Código C.7: Función pos.max.diff

C.5. Función *pl.model*

```
#' @title Parallel-line analysis model
#' @description This function generates the appropriate model
#' to then perform parallel line analysis.
#' @param data matrix or dataframe that contains the information
#' @return list with lm() model
#' @examples
#' pl.model(assay.data(hepatitis_example))
#' @export

pl.model <- function(data){

  model <- lm(Response ~ factor(Sample)*Dilution +
              factor(Sample):factor(Dilution), data = data)

  return(model)
}
```

Código C.8: Función *pl.model*

C.6. Función *best.model*

```

#' @title Best model for PLM
#' @description This function generates a dataframe with the minimum and maximum
#' levels of each dose-response curve under study that generates the best model
#' for the plm analysis among all the possible ones. Prioritize those that
#' include the central point and the maximum number of points and complies all
#' the tests. If none are obtained, apply an alpha 0.01 to the linearity
#' criterion and search a second time.
#' @param data matrix or dataframe that contains the information
#' @return dataframe with minimum and maximum levels of each dose-response curve
#' @examples
#' best.model(assay.data(hepatitis_example))
#' @export

best.model <- function(data){

  pos <- select.range(data)
  best <- data.frame()
  final.pos <- data.frame()
  a <- mean(data$Level)

  for (j in 1:nrow(pos)){
    sbdt <- data.frame()

    for (i in unique(data$Sample)){
      sbdt1 <- data[data$Sample==i &
                    data$Level>= pos[j,paste0("Low", i)] &
                    data$Level<= pos[j,paste0("High", i)],]
      sbdt <- rbind (sbdt, sbdt1)
    }
    if (check.model(pl.model(sbdt)) == T){
      final.pos <- rbind(final.pos, pos[j,])
    }
  }

  if (nrow(final.pos) == 0) {
    for (j in 1:nrow(pos)){
      sbdt <- data.frame()

      for (i in unique(data$Sample)){
        sbdt1 <- data[data$Sample==i &
                      data$Level>= pos[j,paste0("Low", i)] &
                      data$Level<= pos[j,paste0("High", i)],]
        sbdt <- rbind (sbdt, sbdt1)
      }

      if (check.model(pl.model(sbdt), alpha = 0.01) == T){
        final.pos <- rbind(final.pos, pos[j,])
      }
    }
  }
}

```

```

    }
  }
}

for (j in 1:nrow(final.pos)){
  if (a-1&&a+1 %in% final.pos[j,1]:final.pos[j,2]){
    best <- rbind(best, final.pos[j,])
  }
}

best <- best[which.max(final.pos[,2]-final.pos[,1]), ]

if (nrow(best) == 0) {
  best <- best[which.max(final.pos[,2]-final.pos[,1]), ]
}

rownames(best) <- "Level"

return(best)
}

```

Código C.9: Función best.model

C.6.1. Función auxiliar *select.range*

```

#' @title All possible ranges for further analysis
#' @description This function calculates all possible ranges for later
#' plm analysis.
#' @param data dataframe to search all possible ranges
#' @return new dataframe with all possibilities Low and High Level for all
#' samples
#' @examples
#' select.range(hepatitis_example)
#' @export

select.range <- function (data){
  c <- vector()
  m <- vector()

  for (i in unique(data$Sample)){
    c <- c(c, paste0("Low", i), paste0("High", i))
  }

  pos <- expand.grid(replicate(length(unique(data$Sample))*2,
                              min(data$Level):max(data$Level),
                              simplify = FALSE))

  n <- ncol(pos)
  colnames(pos) <- c

  Means <- tapply(data$Response, list(data$Sample, data$Level), FUN = mean)

  for (i in 1:nrow(Means)){

```

```

  a <- mean(1:ncol(Means))
  m[[i]] <- which.min(abs(Means[i,] - Means[1,a]))
}

mx <- max(abs(m - m[1]))

for (k in (seq(1, n, 2))) {
  pos <- pos[pos[,k+1] - pos[,k] >= 2,]
}

pos <- pos[apply(pos[,seq(2, n, 2)] - pos[,seq(1, n, 2)], 1,
  function(x) all(x == x[1])), ]

if (n<5){
  pos <- pos[abs(pos[,3] - pos[,1]) <= mx,]
} else {
  pos <- pos[apply(abs(pos[,seq(3, n, 2)] - pos[,1]), 1,
    function(x) all(x <= mx)), ]
}

return(pos)
}

```

Código C.10: Función select.range

C.6.2. Función auxiliar *check.model*

```

#' @title Check for PLM test
#' @description This function check if the given model passes the regression
#' linearity and parallel tests.
#' @param model lm model to check
#' @param alpha alpha value to linearity test. 0.05 by default
#' @return boolean value TRUE or FALSE
#' @export

check.model <- function(model, alpha = 0.05){
  anv <- anova(model)
  ifelse(anv$'Pr(>F) '[2] <= 0.05 & anv$'Pr(>F) '[4] >= alpha &
    anv$'Pr(>F) '[3] >= 0.05,
    OK <- T, OK <- F )
  return(OK)
}

```

Código C.11: Función check.model

C.7. Función *com.slope*

```

#' @title Common slope between two or more dose-response curves
#' @description This function calculates the common slope between
#' two or more curves
#' @param data matrix or dataframe that contains the information.
#' @return the value for the common slope
#' @examples
#' com.slope(hepatitis_example)
#' @export

com.slope <- function(data){
  I <- log(data$Dose[2]/data$Dose[1])
  d <- length(unique(data$Dose[data$Sample=="STD"]))
  h <- length(unique(data$Sample))
  n <- length(unique(data$Replicate))
  H1 <- (12*n)/(d^3-d)

  Means <- tapply(data$Response, list(data$Sample,data$Level), FUN = mean)
  Means <- data.frame(matrix(na.omit(as.vector(t(Means))),
                             ncol = d, byrow = TRUE),
                      row.names = unique(data$Sample))
  P <- apply(Means, MARGIN = 1, FUN = sum)
  L <- c()
  for (i in rownames(Means)){
    e <- sum(1:length(Means[i,])*Means[i,1:length(Means[i,])]) -(1/2*(d+1)*P[i])
    L <- c(L, e)
  }

  b <- (H1* (sum(L))) / (I*n*h)

  return(b)
}

```

Código C.12: Función *com.slope*

C.8. Función *rel.pot*

```

#' @title Relative potency
#' @description This function calculates the relative potency of a given sample
#' and a given std to compare
#' @param data matrix or dataframe that contains the information.
#' @param std the std name in data
#' @param tst the tst name in data
#' @param output.value the value to return choose between potency, upper or
  lower
#' @return the value for relative potency or its confidence limit upper or low
#' @examples
#' rel.pot(assay.data(hepatitis_example), "STD", "TST", output.value = "potency
  ")
#' @export

rel.pot <- function(data, std, tst, output.value){
  anv <- anova(pl.model(data))
  d <- length(unique(data$Dose[data$Sample=="STD"]))
  n <- length(unique(data$Replicate))
  Means <- tapply(data$Response, list(data$Sample, data$Level), FUN = mean)
  Means <- data.frame(matrix(na.omit(as.vector(t(Means))),
                             ncol = d, byrow = TRUE),
                      row.names = unique(data$Sample))
  P <- apply(Means, MARGIN = 1, FUN = sum)
  t <- qt(0.05/2, anv$Df[5], lower.tail = F)
  s2 <- anv$`Mean Sq`[5]
  b <- com.slope(data)

  Mt <- ((P[tst]) - (P[std])) / (d*b)
  C <- anv$`Mean Sq`[2] / (anv$`Mean Sq`[2] - (s2 * t^2))
  V <- anv$`Mean Sq`[2] / (b^2 * d * n)

  pot <- exp(Mt)
  lower_cl <- exp((Mt*C) - sqrt((C-1)*((C*Mt)^2 + (2*V))))
  upper_cl <- exp((Mt*C) + sqrt((C-1)*((C*Mt)^2 + (2*V))))

  if (output.value == "potency"){
    return(pot)
  } else if (output.value == "lower"){
    return(lower_cl)
  } else if (output.value == "upper"){
    return(upper_cl)
  }
}

```

Código C.13: Función *rel.pot*

C.9. Función *est.pot*

```
#' @title Calculated potency
#' @description This function calculates the potency of a given sample
#' and a given std to compare
#' @param data matrix or dataframe that contains the information.
#' @param std the std name in data
#' @param tst the tst name in data
#' @param output.value the value to return choose between potency, upper or
#' lower.
#' @param factor a factor or estimated potency to calculate the potency.
#' 1 by default
#' @param unit if it is necessary to express the potency in certain units
#' @return the value for relative potency or its confidence limit upper or low
#' @examples
#' est.pot(assay.data(hepatitis_example, "STD", "TST", output.value = "potency",
#' unit=" g /mL"))
#' @export

est.pot <- function(data, std, tst, output.value, factor=1, unit=NULL){

  if (output.value == "potency"){
    value <- (rel.pot(data, std, tst, output.value = "potency"))
  } else if (output.value == "lower"){
    value <- (rel.pot(data, std, tst, output.value = "lower"))
  } else if (output.value == "upper"){
    value <- (rel.pot(data, std, tst, output.value = "upper"))
  }

  if (!is.null(unit)){
    return(paste(value*factor, unit))
  } else {
    return(value*factor)
  }
}
```

Código C.14: Función *est.pot*

Apéndice D

Anexo: Código RMarkdown

A continuación se presenta el código RMarkdown para el diseño del informe con los resultados obtenidos.

```

title: 'Determinacion de la potencia'
author: "Rocio Lorenzo Lopez"

output:
  pdf_document:
    latex_engine: pdflatex
    keep_tex: yes
    toc: no
header-includes: |
  \usepackage{fancyhdr}
  \usepackage[spanish]{babel}
  \usepackage{datetime}
  \usepackage{lastpage}
  \usepackage{float}
  \usepackage[T1]{fontenc}
  \floatplacement{figure}{H}
  \pagestyle{fancy}
  \fancyhf{}
  \lhead{\includegraphics[height=0.75cm]{logouoc.png}}
  \chead{\includegraphics[height=0.75cm]{logoub.png}}
  \rhead{Potencia biologica. Aplicacion y libreria R}
  \lfoot{Impreso el \today, \currenttime}
  \rfoot{P gina \thepage de \pageref{LastPage}}
  \renewcommand{\headrulewidth}{1pt}
  \renewcommand{\footrulewidth}{1pt}
  \setlength{\headheight}{25pt}
  \fancypagestyle{plain}{
    \fancyhf{}
    \lhead{\includegraphics[height=0.75cm]{logouoc.png}}
    \chead{\includegraphics[height=0.75cm]{logoub.png}}
  }

```

```

\thead{Potencia biológica. Aplicación y librería R}
\tfoot{Impreso el \today, \currenttime}
\rfoot{Página \thepage de \pageref{LastPage}}
\renewcommand{\headrulewidth}{1pt}
\renewcommand{\footrulewidth}{1pt}
}

---

““{r setup, include=FALSE}

knitr::opts_chunk$set(echo = TRUE, comment = NULL, warning = FALSE,
                      message = FALSE)
““

““{r packages, message=FALSE, warning=FALSE, echo=FALSE, include=FALSE}

# Package requires
#
require(openxlsx)
require(plm.crd)

““

““{r assignments, message=FALSE, warning=FALSE, echo=FALSE, include=FALSE}

# Work environment – EXCEL DATA
#
# File name and working directory (automatically with VBA)

fileNameRoot <- "Data_Plantilla"
FileName <- paste(fileNameRoot, ".xlsx", sep='')

““

““{r Data Import, warning=FALSE, message=FALSE, include=FALSE}

# Data Import for Script
#
load.info(FileName)
load.data(FileName, "rawData")

if (AnalysisRange != "Toda la placa" &
    AnalysisRange != "Toda la placa (individual)") {
  AnalysisRange_vector <- unlist(strsplit(AnalysisRange, "\\+"))
  rawData <- rawData[rawData$Sample %in% AnalysisRange_vector
                    | rawData$Sample == "STD", ]
} #AnalysisRange_vector with sample names to analyze

# Data preparation
#

rawData$Response <- rawData$Response/1000 #In pheu data

```

```

rawData <- subset(rawData, rawData$Response != 0) #Eliminates rows without
  response
assay_data <- assay.data(rawData, y.treat = log) # Data preparation and response
  transformation
““

# Informaci n del ensayo

| <!-- --> | <!-- --> |
|-----|:-----|
| R-script: | 'r normalizePath(knitr::current_input(), winslash =
| "/"') '|
| Archivo origen datos: | 'r FileName '|
| ID ensayo: | 'r paste(assayType, " - ", Antigen) '|
| Operador / Fecha: | 'r operatorDate '|
| Fecha/hora de impresion: | 'r format(Sys.time(), "%Y-%m-%d %H:%M:%S") '|
| ID STD : | 'r STDBatch '|
| Potencia estimada STD: | 'r STDEstimatedPotency '|
| Unidades STD: | 'r STDunits '|
| Rango de Analisis: | 'r AnalysisRange '|

# Datos Ensayo

““{r Data Table, warning=FALSE, message=FALSE, echo=FALSE, results='asis'}

# Data Import for Script
#-----

ind_data <- list()
names <- c()

for (i in unique(rawData$Sample)){
  dat <- data.frame(Dosis = unique(rawData$Dose))
  dat$ODs1 <- rawData$Response[rawData$Sample == i & rawData$Replicate == 1]
  dat$ODs2 <- rawData$Response[rawData$Sample == i & rawData$Replicate == 2]
  dat$ODs3 <- rawData$Response[rawData$Sample == i & rawData$Replicate == 3]
  dat$CV <- apply(dat[2:4], MARGIN = 1, FUN = cv.calc)
  ind_data[[i]] <- dat
  names <- c(names, get(paste(i, "Batch", sep = ")))
}

ind_data <- setNames(ind_data, names)

for (name in names(ind_data)) {
  cat(paste("\\paragraph{", name, "}\n", sep = ""))
  cat(knitr::kable(ind_data[[name]], format = "latex"))
  cat("\n\n")
}

# Print in pdf initial data for each sample and CV value

““

```

```

# Analisis PLM y Determinación Potencia

““{r Data Analysis Initial, warning=FALSE, message=FALSE, echo=FALSE, results=’
  asis’, fig.align=’center’}

# Data Analysis
#-----

# Perform the proposed analysis in two ways depending on whether
# all the curves are analyzed together or individually (each one
# against the standard).

if (AnalysisRange == "Toda la placa (individual)") {
  subgroup <- list()
  analysis_tab <- list()
  for (i in unique(assay_data$Sample[assay_data$Sample != "STD"])) {
    subgroup[[i]] <- rbind(assay_data[assay_data$Sample == "STD", ],
                          assay_data[assay_data$Sample == i, ])
  } #One dataframe standard-sample for each sample.

  for (j in names(subgroup)) {
    sample_pos <- best.model(subgroup[[j]])
    subrowData <- data.frame()
    for (k in unique(subgroup[[j]]$Sample)) {
      sbdt1 <- subgroup[[j]][subgroup[[j]]$Sample==k &
                            subgroup[[j]]$Level>= sample_pos[1, paste0("Low", k)] &
                            subgroup[[j]]$Level<= sample_pos[1, paste0("High", k)], ]
      subrowData <- rbind(subrowData, sbdt1)
    } #Selected range with best.model for each subgroup

    if (length(subrowData) == 0) {
      analysis_tab[[j]] <- anova(pl.model(subgroup[[j]]))
    } else {
      analysis_tab[[j]] <- anova(pl.model(subrowData))
    }
    # Perform model fit with best.model or all data if there is no one
    analysis_tab[[j]] <- anova(pl.model(subrowData)) # ANOVA for the model fit
    row.names(analysis_tab[[j]]) <- c("Test de preparacion",
                                     "Test de regresion",
                                     "Test No-Paralelismo",
                                     "Test No-Linealidad",
                                     "Residuos")

    cat("\subsection{", j, "-", get(paste(j, "Batch", sep="")), " }\n\n")

    cat("Rango de dosis/niveles para el mejor ajuste", "\n\n")

    cat(knitr::kable(t(sample_pos), format = "latex"))

#Plot1
plot(log(assay_data$Dose[assay_data$Sample=="STD"]),
      assay_data$Response[assay_data$Sample=="STD"], pch = 20,
      col = "darkblue", xlab = "Log(Dosis)", ylab = "Log(Respuesta)",
      main = paste(j, get(paste(j, "Batch", sep=""))),

```

```

xlim = c(min(log(assay_data$Dose)), max(log(assay_data$Dose))),
ylim = c(min(assay_data$Response), max(assay_data$Response))
abline(lm(assay_data$Response[assay_data$Sample=="STD"] ~
        log(assay_data$Dose[assay_data$Sample=="STD"])),
       col = "black", lwd = 1, lty = 1)

points(log(assay_data$Dose[assay_data$Sample== j]),
       assay_data$Response[assay_data$Sample== j],
       pch = 20, col = "yellowgreen")

abline(lm(assay_data$Response[assay_data$Sample== j] ~
        log(assay_data$Dose[rawData$Sample== j])),
       col = "black", lwd = 1, lty = 2)
legend("topleft", legend = c("STD", i),
       col = c("darkblue", "yellowgreen"), lwd = 2)

#ANOVA Table
cat("Tabla ANOVA", "\n\n")
cat("\n\n")

cat(knitr::kable(analysis_tab[[j]], format = "latex"))
cat("\n\n")

cat("Test de regresion (a = 0,05) para la muestra", j, ":",
    ifelse(analysis_tab[[j]]$'Pr(>F) '[2] <= 0.05,
          "Cumple.", "No cumple."), "(Valor de p:",
    analysis_tab[[j]]$'Pr(>F) '[2], ")", "\n\n")

cat("Test de linealidad (a = 0,05) para la muestra", j, ":",
    ifelse(analysis_tab[[j]]$'Pr(>F) '[4] >= 0.05,
          "Cumple.", "No cumple."), "(Valor de p:",
    analysis_tab[[j]]$'Pr(>F) '[4], ")", "\n\n")

cat("Test de paralelismo (a = 0,05) para la muestra", j, ":",
    ifelse(analysis_tab[[j]]$'Pr(>F) '[3] >= 0.05,
          "Cumple.", "No cumple."), "(Valor de p:",
    analysis_tab[[j]]$'Pr(>F) '[3], ")", "\n\n")

cat("Valor para la pendiente comun STD -", j, ":",
    com.slope(subgroup[[j]]), "\n\n")
#Common slope value for each STD-sample subgroup

cat("Potencia relativa de", j, ":",
    signif(rel.pot(subgroup[[j]], "STD", j, output.value = "potency"),
          digits = 7), ". ", "\n\n")

cat("Intervalo de confianza para el valor de potencia relativa de", j, ":", ("
,
    signif(rel.pot(subgroup[[j]], "STD", j, output.value = "lower"),
          digits = 7), " - ",
    signif(rel.pot(subgroup[[j]], "STD", j, output.value = "upper"),
          digits = 7), ").", "\n\n")

estpot <- get(paste(j, "EstimatedPotency", sep=""))

```

```

unitpot <- get(paste(j, "Units", sep = ""))

cat("Potencia estimada de", j, ":", estpot, unitpot, "\n\n")

cat("Potencia de", j, ":",
    signif(est.pot(subgroup[[j]], "STD", j, output.value = "potency",
        factor = estpot), digits = 7), unitpot,
    ". ", "\n\n") #Estimated potency for sample

cat("Intervalo de confianza para el valor de potencia de", j, ":", ("
    signif(est.pot(subgroup[[j]], "STD", j, output.value = "lower",
        factor = estpot), digits = 7), " - ",
    signif(est.pot(subgroup[[j]], "STD", j, output.value = "upper",
        factor = estpot), digits = 7), unitpot,
    "). ", "\n\n")

par(mfrow = c(1, 2))
#Residuals plot
plot(pl.model(subrawData)$residuals, pch = 20, col = "darkblue",
    xlab = "ndice ", ylab = "Residuos",
    main = paste("Gráfico de residuos -", j))
abline(h=0)
#Plot2
plot(log(subrawData$Dose[subrawData$Sample=="STD"]),
    subrawData$Response[subrawData$Sample=="STD"], pch = 20,
    col = "darkblue", xlab = "Log(Dosis)", ylab = "Log(Respuesta)",
    main = paste(j, get(paste(j, "Batch", sep=""))),
    xlim = c(min(log(subrawData$Dose)), max(log(subrawData$Dose))),
    ylim = c(min(subrawData$Response), max(subrawData$Response)))
abline(lm(subrawData$Response[subrawData$Sample=="STD"] ~
    log(subrawData$Dose[subrawData$Sample=="STD"])),
    col = "black", lwd = 1, lty = 1)

points(log(subrawData$Dose[subrawData$Sample== j]),
    subrawData$Response[subrawData$Sample== j],
    pch = 20, col = "yellowgreen")

abline(lm(subrawData$Response[subrawData$Sample== j] ~
    log(subrawData$Dose[subrawData$Sample== j])),
    col = "black", lwd = 1, lty = 2)
legend("topleft", legend = c("STD", i),
    col = c("darkblue", "yellowgreen"), lwd = 2)
par(mfrow = c(1, 1))
cat("\n\n\\pagebreak\n")
}
} else {
sample_pos <- best.model(assay_data)
subrawData <- data.frame()
for (1 in unique(assay_data$Sample)){
    sbdt1 <- assay_data[assay_data$Sample== 1 &
        assay_data$Level>= sample_pos[1, paste0("Low", 1)] &
        assay_data$Level<= sample_pos[1, paste0("High", 1)]
    ],]
    subrawData <- rbind(subrawData, sbdt1)
}
}

```

```

} #Selected range with best.model for all data

if (length(subrawData) == 0) {
  plm_table <- anova(pl.model(assay_data))
} else {
  plm_table <- anova(pl.model(subrawData))
} # Perform model fit with best.model or all data if there is no one and
# ANOVA for the model fit

row.names(plm_table) <- c("Test de preparacion",
                          "Test de regresion",
                          "Test No-Paralelismo",
                          "Test No-Linealidad",
                          "Residuos")

cat("\\subsection{An lisis PLM}\\n")

cat("Rango de dosis/niveles para el mejor ajuste", "\\n\\n")

cat(knitr::kable(t(sample_pos), format = "latex"))
cat("\\n\\n")

cat("Tabla ANOVA", "\\n\\n")
cat(knitr::kable(plm_table, format = "latex"))
cat("\\n\\n")

cat("Test de regresion para (a = 0,05): ",
     ifelse(plm_table$`Pr(>F)`[2] <= 0.05, "Cumple.", "No cumple. -"),
     "Valor de p:", plm_table$`Pr(>F)`[2], "\\n\\n")

cat("Test de linealidad para (a = 0,05): ",
     ifelse(plm_table$`Pr(>F)`[4] >= 0.05, "Cumple.", "No cumple. -"),
     "Valor de p:", plm_table$`Pr(>F)`[4], "\\n\\n")

cat("Test de paralelismo para (a = 0,05): ",
     ifelse(plm_table$`Pr(>F)`[3] >= 0.05, "Cumple.", "No cumple. -"),
     "Valor de p:", plm_table$`Pr(>F)`[3], "\\n\\n")

cat("Valor para la pendiente comun:", com.slope(subrawData), "\\n\\n")
#Common slope value for all curves

for (m in unique(assay_data$Sample[assay_data$Sample != "STD"])){

  cat("\\subsection{" , m, "-"}", get(paste(m, "Batch", sep="")), "}\\n\\n")

  plot(log(assay_data$Dose[assay_data$Sample=="STD"]),
        assay_data$Response[assay_data$Sample=="STD"],
        pch = 20, col = "darkblue",
        xlab = "Log(Dosis)", ylab = "Log(Respuesta)",
        main = paste(m, get(paste(m, "Batch", sep=""))),
        xlim = c(min(log(assay_data$Dose)), max(log(assay_data$Dose))),
        ylim = c(min(assay_data$Response), max(assay_data$Response)))
}

```



```

abline(lm(assay_data$Response[assay_data$Sample=="STD"] ~
        log(assay_data$Dose[assay_data$Sample=="STD"])),
       col = "black", lwd = 1, lty = 1)

points(log(assay_data$Dose[assay_data$Sample== m]),
       assay_data$Response[assay_data$Sample== m],
       pch = 20, col = "yellowgreen")

abline(lm(assay_data$Response[assay_data$Sample== m] ~
        log(assay_data$Dose[assay_data$Sample== m])),
       col = "black", lwd = 1, lty = 2)
legend("topleft", legend = c("STD", m),
       col = c("darkblue", "yellowgreen"), lwd = 2)

cat("Potencia relativa de", m, ":",
    signif(rel.pot(subrawData, "STD", m, output.value = "potency"),
           digits = 7), "\n\n")
cat("Intervalo de confianza para el valor de potencia relativa de", m, ": ("
,
    signif(rel.pot(subrawData, "STD", m, output.value = "lower"),
           digits = 7), " - ",
    signif(rel.pot(subrawData, "STD", m, output.value = "upper"),
           digits = 7), ")", "\n\n")

estpot <- get(paste(m, "EstimatedPotency", sep=""))
unitpot <- get(paste(m, "Units", sep = ""))

cat("Potencia estimada de", m, ":", estpot, unitpot, "\n\n")

cat("Potencia de", m, ":",
    signif(est.pot(subrawData, "STD", m, output.value = "potency",
                  factor = estpot), digits = 7), unitpot, "\n\n")

cat("Intervalo de confianza para el valor de potencia de", m, ": ("
,
    signif(est.pot(subrawData, "STD", m, output.value = "lower",
                  factor = estpot), digits = 7), " - ",
    signif(est.pot(subrawData, "STD", m, output.value = "upper",
                  factor = estpot), digits = 7), unitpot,
    ")", "\n\n")

par(mfrow = c(1, 2))
plot(pl.model(subrawData)$residuals, pch = 20, col = "darkblue",
     xlab = " ndice ", ylab = "Residuos",
     main = paste("Gráfico de residuos -", m))
abline(h=0)
plot(log(subrawData$Dose[subrawData$Sample=="STD"]),
     subrawData$Response[subrawData$Sample=="STD"], pch = 20,
     col = "darkblue", xlab = "Log(Dosis)", ylab = "Log(Respuesta)",
     main = paste(m, get(paste(m, "Batch", sep=""))),
     xlim = c(min(log(subrawData$Dose)), max(log(subrawData$Dose))),
     ylim = c(min(subrawData$Response), max(subrawData$Response)))

abline(lm(subrawData$Response[subrawData$Sample=="STD"] ~

```

```

        log(subrawData$Dose[subrawData$Sample=="STD"])),
col = "black", lwd = 1, lty = 1)

points(log(subrawData$Dose[subrawData$Sample== m]),
subrawData$Response[subrawData$Sample== m],
pch = 20, col = "yellowgreen")

abline(lm(subrawData$Response[subrawData$Sample== m] ~
log(subrawData$Dose[subrawData$Sample== m])),
col = "black", lwd = 1, lty = 2)
legend("topleft", legend = c("STD", m),
col = c("darkblue", "yellowgreen"), lwd = 2)
par(mfrow = c(1, 1))
cat("\n\n\\pagebreak\n")
}
}
'''

```

Código D.1: Código RMarkdown

Apéndice E

Anexo: Informe de resultados generado

A continuación se muestra el informe generado por el código del anterior anexo D.

Determinación de la potencia

Rocío Lorenzo López

Información del ensayo

R-script:	C:/Users/rlore/Documents/TFM/ScriptsR/Informe_final.Rmd
Archivo origen datos:	Data_Plantilla.xlsx
ID ensayo:	Otro - Hepatitis
Operador / Fecha:	R. Lorenzo - 20/12/2023
Fecha/hora de impresión:	2024-01-12 22:03:56
ID STD :	Estándar
Potencia estimada STD:	20
Unidades STD:	µg/mL
Rango de Análisis:	Toda la placa

Datos Ensayo

	Dosis	ODs1	ODs2	ODs3	CV
Estándar	6.25e-05	0.043	0.045	0.051	8.985609
	1.25e-04	0.093	0.099	0.082	9.439794
	2.50e-04	0.159	0.154	0.166	3.775186
	5.00e-04	0.283	0.295	0.362	13.586652
	1.00e-03	0.514	0.531	0.545	2.929090

	Dosis	ODs1	ODs2	ODs3	CV
Vacuna T	6.25e-05	0.097	0.097	0.094	1.804220
	1.25e-04	0.167	0.157	0.178	6.277271
	2.50e-04	0.327	0.355	0.345	4.144848
	5.00e-04	0.501	0.665	0.576	14.138840
	1.00e-03	1.140	1.386	1.051	14.553254

	Dosis	ODs1	ODs2	ODs3	CV
Vacuna U	6.25e-05	0.086	0.071	0.073	10.623297
	1.25e-04	0.127	0.146	0.133	7.176750
	2.50e-04	0.277	0.268	0.269	1.818016
	5.00e-04	0.586	0.489	0.546	9.021774
	1.00e-03	0.957	0.866	1.045	9.362363

	Dosis	ODs1	ODs2	ODs3	CV
Vacuna V	6.25e-05	0.082	0.082	0.086	2.771281
	1.25e-04	0.145	0.144	0.173	10.689661
	2.50e-04	0.318	0.306	0.316	2.051841
	5.00e-04	0.552	0.551	0.624	7.271722
	1.00e-03	1.037	1.039	1.068	1.655472

Análisis PLM y Determinación Potencia

Análisis PLM

Rango de dosis/niveles para el mejor ajuste

	Level
LowSTD	1
HighSTD	5
LowTST	1
HighTST	5
LowUST	1
HighUST	5
LowVST	1
HighVST	5

Tabla ANOVA

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Test de preparacion	3	4.4752218	1.4917406	223.3920338	0.0000000
Test de regresion	1	47.5841262	47.5841262	7125.8466834	0.0000000
Test No-Paralelismo	3	0.0186856	0.0062285	0.9327402	0.4338160
Test No-Linealidad	12	0.0742323	0.0061860	0.9263737	0.5307794
Residuos	40	0.2671072	0.0066777	NA	NA

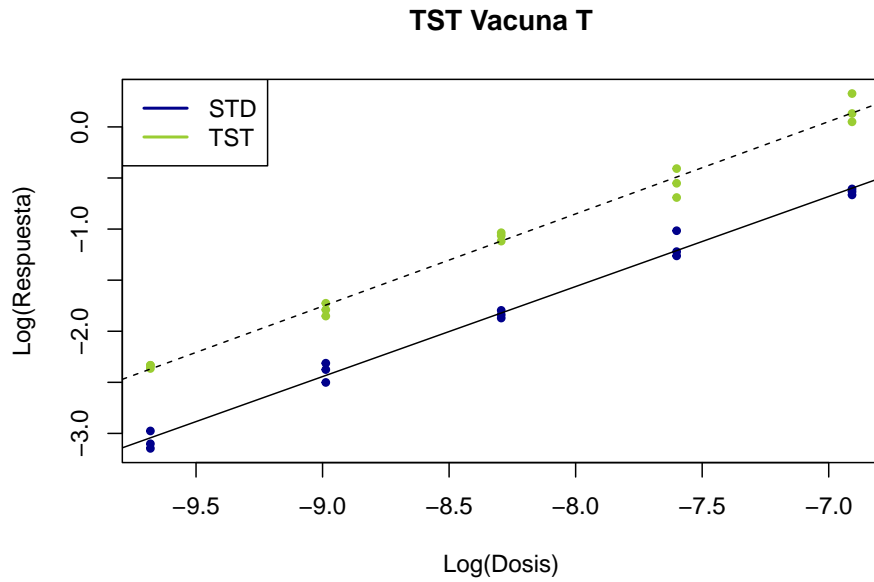
Test de regresión para ($\alpha = 0,05$): Cumple. Valor de p: 1.084434e-46

Test de linealidad para ($\alpha = 0,05$): Cumple. Valor de p: 0.5307794

Test de paralelismo para ($\alpha = 0,05$): Cumple. Valor de p: 0.433816

Valor para la pendiente común: 0.9084792

TST - Vacuna T



Potencia relativa de TST : 2.170981

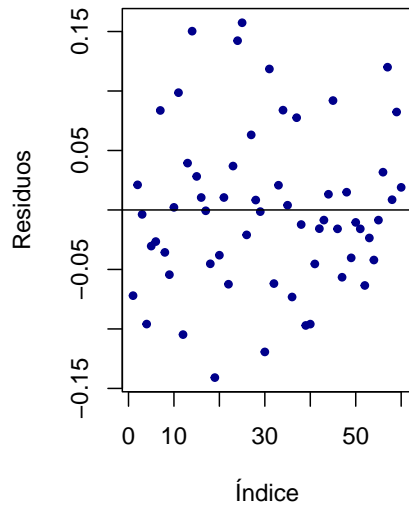
Intervalo de confianza para el valor de potencia relativa de TST : (2.027237 - 2.326986)

Potencia estimada de TST : 20 µg/mL

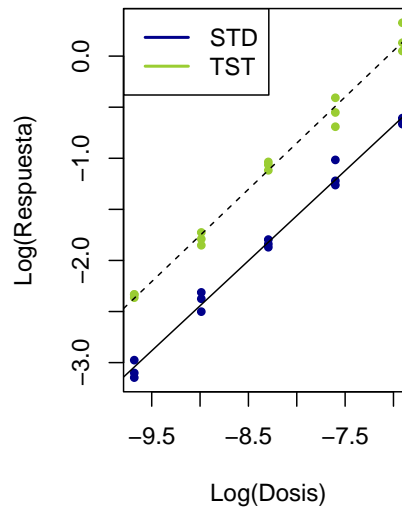
Potencia de TST : 43.41962 µg/mL

Intervalo de confianza para el valor de potencia de TST : (40.54473 - 46.53973 µg/mL)

Gráfico de residuos - TST

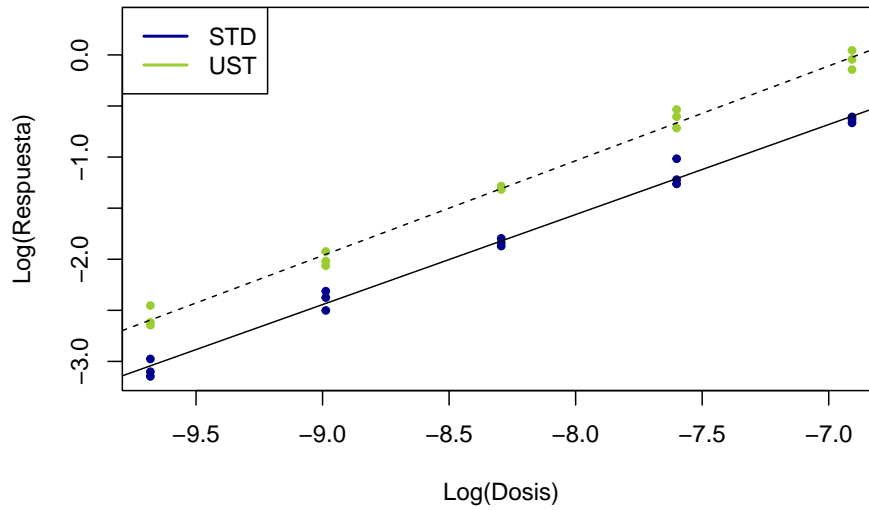


TST Vacuna T



UST - Vacuna U

UST Vacuna U



Potencia relativa de UST : 1.758149

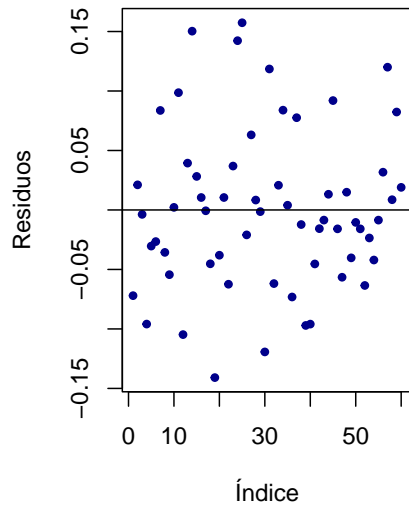
Intervalo de confianza para el valor de potencia relativa de UST : (1.643489 - 1.882026)

Potencia estimada de UST : 20 µg/mL

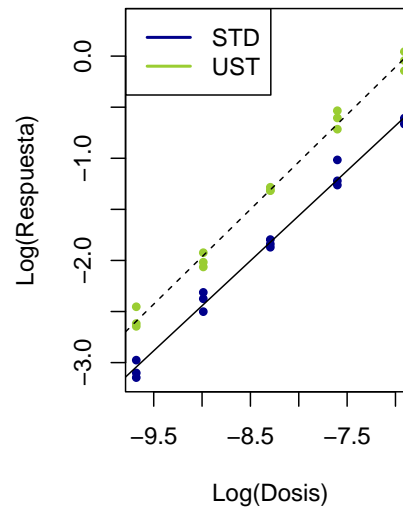
Potencia de UST : 35.16298 µg/mL

Intervalo de confianza para el valor de potencia de UST : (32.86978 - 37.64052 µg/mL)

Gráfico de residuos – UST

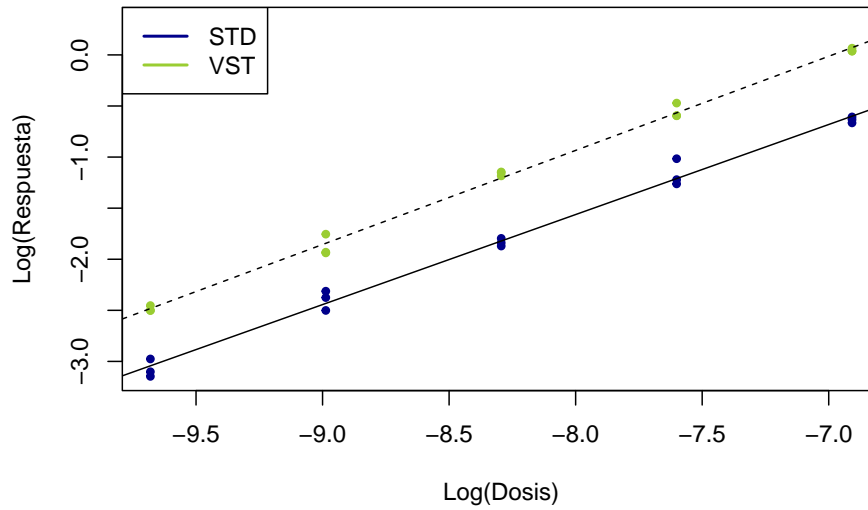


UST Vacuna U



VST - Vacuna V

VST Vacuna V



Potencia relativa de VST : 1.970084

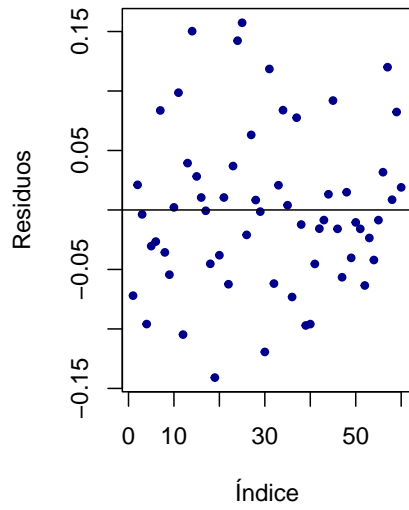
Intervalo de confianza para el valor de potencia relativa de VST : (1.840625 - 2.11029)

Potencia estimada de VST : 20 µg/mL

Potencia de VST : 39.40168 µg/mL

Intervalo de confianza para el valor de potencia de VST : (36.81249 - 42.20579 µg/mL)

Gráfico de residuos - VST



VST Vacuna V

