

Meta-análisis de correlaciones y meta-regresión en R: Guía práctica

Juan David Leongómez 

14 abril, 2023

EvoCo: Laboratorio de Evolución y Comportamiento Humano, Facultad de Psicología, Universidad El Bosque, Bogotá, Colombia. Email: jleongomez@unbosque.edu.co. Web: jdleongomez.info.

Resumen

El meta-análisis es un método ampliamente utilizado para sintetizar los datos de diferentes estudios. Sin embargo estudiantes, profesionales e investigadores a menudo carecemos de conocimientos prácticos para hacer e interpretar un meta-análisis. Esta guía presenta una variedad de herramientas para realizar meta-análisis de correlaciones en R, mediante el uso de ejemplos reales. Incluye desde análisis simples y su interpretación, hasta el análisis de moderadores (meta-regresión), usando los paquetes **metafor** (Viechtbauer, 2010) y **metaviz** (Kossmeier et al., 2020). También incluye explicaciones para la transformación de coeficientes r de Pearson a z de Fisher (y viceversa), creación de gráficos de bosque (*forest plots*) y gráficos de embudo (*funnel plots*), análisis de heterogeneidad y diagnósticos de influencia, así como estrategias para detectar posibles sesgos de publicación utilizando el paquete **weightr** (Coburn & Vevea, 2019), y para determinar el poder estadístico de un meta-análisis utilizando **metameta** (Quintana, 2022). Con esta guía, los lectores podrán adquirir las habilidades necesarias para realizar meta-análisis de correlaciones de manera efectiva en R y obtener resultados confiables.



Conocimientos necesarios: Esta guía asume (1) un manejo básico de R, suficiente para al menos instalar paquetes y correr funciones; (2) comprensión de correlaciones y regresiones; y (3) un entendimiento de los principios del meta-análisis. De ser necesario, para adquirir la comprensión básica y necesaria del meta-análisis, recomiendo ver [este video](#) introductorio al meta-análisis en *jamovi* (Leongómez, 2021) publicado en mi canal de YouTube [Investigación Abierta](#).

Versión de R: Para poder correr el código tal cual aparece en esta guía, se requiere R versión 4.1 o superior, pues he usado el *pipe* nativo de R (`|>`). En versiones anteriores de R, el *pipe* nativo puede ser reemplazado por el pipe de **magrittr** (`%>%`), cargando ese paquete (Bache & Wickham, 2022). Para más información, puedes ver la sección 11.1.5 de Gil Bellosta (2023).

Cita esta guía como

Leongómez, J. D. (2022). Meta-análisis de correlaciones y meta-regresión en R: Guía práctica. *MetaArXiv*. <https://doi.org/10.31222/osf.io/yaxd4>

Índice

1. Introducción	3
1.1. ¿Qué es un meta-análisis y por qué es importante?	4
1.2. Sesgos de publicación	4
1.3. Motivación	5
2. Base de datos de ejemplo	6
2.1. Variables de la base de datos	7
2.2. Preparación de los datos: transformación de coeficientes r de Pearson a z de Fisher	8
3. Cómo hacer el meta-análisis	9
3.1. Interpretación de los resultados del meta-análisis	10
3.1.1. Reporte del meta-análisis básico	11
3.1.2. Alternativa: reportar el estimado como r de Pearson en vez de z de Fisher	11
3.2. Más información sobre heterogeneidad	12
3.2.1. Reporte del meta-análisis con intervalos de confianza para medidas de heterogeneidad	13
3.3. Diagnóstico de influencia	13
3.4. <i>Forest plot</i> (diagrama de bosque)	16
3.5. <i>Funnel plot</i> (diagrama de embudo), regresión de Egger y sesgo de estudios pequeños	19
3.5.1. Errores en la interpretación del <i>Funnel plot</i> y la regresión de Egger	19
3.5.2. <i>Funnel plot</i>	19
3.5.3. Regresión de Egger	23
3.6. Sesgo de publicación (<i>publication bias</i>)	23
3.6.1. Método <i>trim and fill</i> (recorte y relleno)	23
3.6.2. Modelo de función de peso de Vevea y Hedges	25
3.7. Poder estadístico del meta-análisis	27
3.7.1. Instalación de metameta	27
3.7.2. Análisis de poder	28
3.7.2.1. Poder mediano del meta-análisis para diferentes tamaños de efecto.	28
3.7.2.2. Poder de cada estudio para diferentes tamaños de efecto.	29
3.7.3. Visualización del análisis de poder (<i>Firepower plot</i>)	30
3.8. Más información sobre influencia y meta-análisis combinatorio	32
3.8.1. <i>Leave-one-out</i>	32
3.8.2. Meta-análisis combinatorio	33
3.8.2.1. Identificación de estudios que contribuyen a los patrones de heterogeneidad a partir del <i>GOSH plot</i>	37
4. Meta-análisis de correlación con moderador (meta-regresión)	41
4.1. Ejemplo 1: Moderación de una variable continua (edad promedio de los participantes)	41
4.1.1. Más información e interpretación de la moderación	42
4.1.1.1. <i>Meta-Analytic Scatter Plot</i> (Gráfico de dispersión meta-analítico)	43
4.1.2. <i>Forest plot</i> y <i>funnel plot</i>	44
4.1.3. Reporte del meta-análisis con moderador continuo	46
4.2. Ejemplo 2: Moderación de una variable categórica (controles usados en cada estudio meta-analizado)	46
4.2.1. Más información e interpretación de la moderación	47
4.2.1.1. <i>Meta-Analytic Scatter Plot</i> (Gráfico de dispersión meta-analítico)	48
4.2.2. <i>Forest plot</i> y <i>funnel plot</i>	50
4.2.3. Reporte del meta-análisis con moderador categórico	51
5. Recomendaciones generales	52
5.1. Pasos para hacer tu meta-análisis	52
5.2. Conclusión	53
Agradecimientos	53
Referencias	53

APÉNDICES	55
Apéndice A. Alternativas a <code>metafor</code>	55
A1. Ejemplo: curva de valores p (p -curve)	56
Apéndice B. Citas y referencias de paquetes de R	56
Apéndice C. Paquetes de R usados para crear este documento	57

1. Introducción

En investigación, es frecuente que busquemos medir la magnitud de un efecto en una población. Por ejemplo, podríamos tener interés en determinar cosas tan diversas como qué tan fuerte es la correlación entre la masa de una estrella y su luminosidad, o entre el nivel socioeconómico y el índice de masa corporal (IMC) en un país determinado. No obstante, es muy poco probable que podamos obtener información sobre la población entera (de estrellas en el universo o de personas del país), por lo que debemos hacer uso de muestras de esa población para inferir la magnitud del efecto.

Sin embargo, a menudo, diferentes estudios obtienen coeficientes de correlación ligeramente diferentes (en algunos casos, muy diferentes). Esta variabilidad no es sorprendente, ya que cada muestra es solo una visualización parcial (y de alguna manera sesgada) de la correlación en la población entera.

Por ejemplo mira la simulación a continuación (Fig. 1). En el panel **A** he simulado una población con un N de 6095 elementos (por ejemplo, personas, estrellas, o lo que sea relevante para ti). En esa población hipotética, la correlación entre dos variables es de $r = 0.25$.

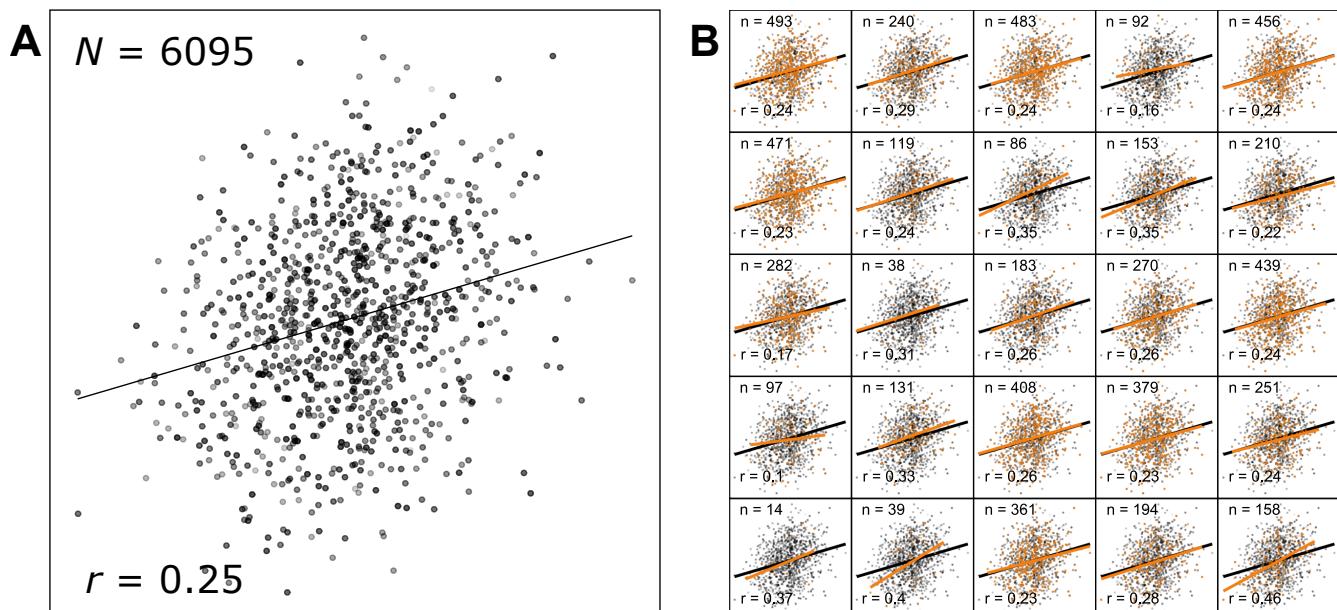


Figura 1. Simulación de una población hipotética de $N = 6095$ elementos, en la que dos variables se correlacionan positivamente ($r = 0.25$). **A.** Simulación de la población y correlación entre dos variables. **B.** 25 muestras aleatorias de diferentes tamaños (n) de esa población, y la correlación encontrada en cada una (r). En naranja se resaltan los elementos seleccionados aleatoriamente y la correlación encontrada en cada muestra, superpuestos sobre el total de la población y la correlación real (en negro, como referencia).

Ahora imagina que tú quieres establecer esa correlación real ($r = 0.25$), pero no la conoces, y no tienes acceso a la población completa de 6095 elementos. Tu mejor apuesta es tomar una muestra aleatoria, que en principio tendería a darte un resultado *similar* al real. En el panel **B** hay 25 muestras aleatorias de esos 6095 elementos, y en cada una se reporta tanto el tamaño de la muestra (n , que va desde una muestra muy pequeña de apenas 14 observaciones, hasta una muy respetable de 493), como la correlación encontrada en esa muestra (r , que va desde 0.10 hasta 0.46).

Entre las 25 muestras aleatorias de nuestra población (Fig. 1B), hay varias que encuentran una correlación similar a la real (que, como es simulada, sabemos que es exactamente de $r = 0.25$), pero hay otras que se alejan bastante.

Analiza detalladamente esos resultados y trata de ver si las que más se alejan (estando por debajo o por encima de 0.25), tienen algo en común.

Si miras con cuidado, verás algo claro: las muestras más grandes tienden a encontrar resultados similares al de nuestra población simulada ($r = 0.25$). Sin embargo, las muestras pequeñas son menos confiables: aunque algunas muestras pequeñas encuentran correlaciones muy cercanas a la real, otras se alejan mucho. Para ilustrar esto, voy a graficar la correlación encontrada en cada muestra, y cómo se asocia con su tamaño de muestra (Fig. 2).

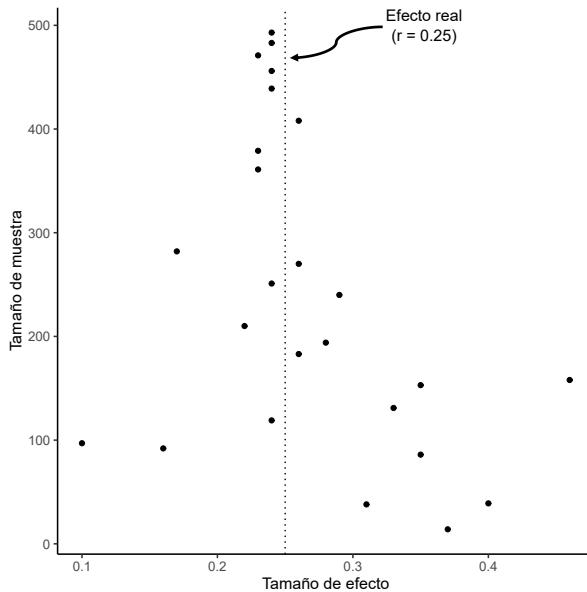


Figura 2. Relación entre el efecto encontrado (eje X) y el tamaño de muestra (eje Y) en 25 muestras aleatorias de una población simulada donde la correlación entre las variables es de $r = 0.25$. Mientras que las muestras grandes (arriba) son confiables y tienden a encontrar resultados similares al del efecto real, muestras pequeñas (abajo) son menos confiables y pueden producir estimaciones del efecto que se alejan mucho del real.

1.1. ¿Qué es un meta-análisis y por qué es importante?

De manera más general, se puede entender así: aunque hay técnicas de muestreo que tienden a hacer que las muestras se *parezcan* más a la población, cada muestra tiene características particulares que conducen a resultados diferentes. Entonces, cuando debemos hacer inferencia con base en muestras porque no podemos tener acceso a la población entera, ¿cómo podemos saber cuál es el efecto real si cada estudio obtiene un resultado distinto?

Una herramienta útil para abordar este problema es el meta-análisis, que es una técnica estadística que nos permite obtener una estimación más precisa del efecto *real* de la correlación. Por supuesto, la palabra clave acá es *estimación*, pues aunque el efecto real es algo que rara vez podremos determinar con absoluta certeza, sí es algo que podemos acercarnos a conocer.

El meta-análisis se puede entender como una especie *promediación* de los efectos encontrados en diversos estudios, dando mayor peso a estudios que tienen más credibilidad (es decir, mayor probabilidad de mostrar un resultado cercano a la verdad). ¿Cómo se hace esto? Como se ve en la Figura 2, el tamaño de una muestra afecta la confiabilidad de un resultado.

Para decirlo de manera más técnica, el tamaño de la muestra afecta el poder estadístico así como el error en la estimación del efecto (Leongómez, 2020a, 2020b); entre mayor sea una muestra, mayor será el poder estadístico y menor tenderá a ser su variación con respecto al efecto real (en últimas, muestras más grandes tienden a parecerse más a la población de la que provienen y por tanto son más confiables). Por esto, podemos usar estas medidas como indicadores de la credibilidad del efecto encontrado por un estudio.

1.2. Sesgos de publicación

Sin embargo, además de las diferencias que pueda tener cada estudio debido a las peculiaridades y tamaño de su muestra, con frecuencia al hacer meta-análisis nos enfrentamos a un riesgo más sutil y menos predecible: los sesgos

de publicación.

Los sesgos de publicación son una amenaza importante para la integridad de la investigación científica. Se refieren a la tendencia de los investigadores y editores de revistas a no publicar estudios que no encuentran un efecto significativo o que muestran resultados contrarios a las hipótesis de sus autores. Estos sesgos pueden afectar a todas las áreas de la investigación, pasando por ejemplo por la medicina, la psicología y la biología (e.g., Baker, 2016; Klein et al., 2018).

Como resultado, los estudios publicados pueden no ser representativos de la verdadera relación entre las variables que se están investigando, lo que puede llevar a conclusiones erróneas y a decisiones equivocadas, incluyendo áreas tan críticas como la práctica clínica y la toma de decisiones políticas, entre otras.

Imagina, por ejemplo, que frente a un efecto estudiado se publicaron preferentemente estudios que encontraron una correlación positiva y significativa, mientras que otros estudios —que no encontraron una correlación o encontraron una correlación negativa— tendieron a no ser publicados por los autores, o a ser rechazados por editores. Al hacer un meta-análisis de los estudios publicados, nuestra estimación del efecto *real* estaría sesgada, sobre-estimando el efecto (Fig. 3).

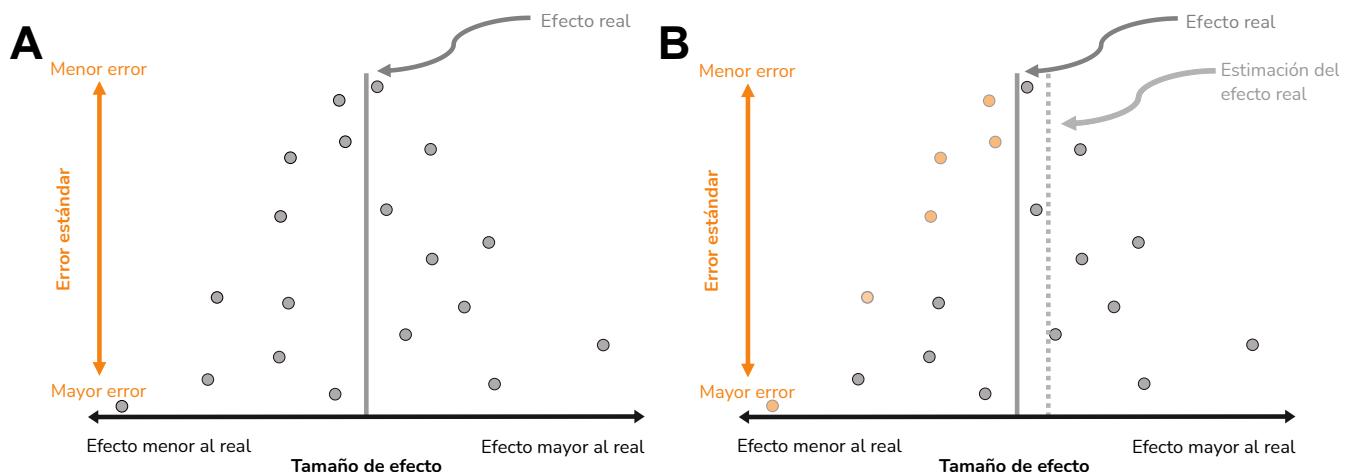


Figura 3. Ejemplo hipotético de distribución de resultados de diferentes estudios (puntos grises) en relación a un efecto real (línea gris vertical). Los estudios realizados con diferentes muestras tienden a estar a ambos lados del efecto real (efectos mayores al real a la derecha y efectos menores a la izquierda). Sin embargo, entre mayor sea la muestra (arriba) los estudios tienden a estar más cerca del efecto real (tienen menor error), pero estudios con muestras pequeñas (abajo) son más variables (y en promedio tienen mayor error), y pueden tener efectos muy alejados del efecto real. Esto forma una especie de triángulo simétrico, con el efecto real en el centro. Esta idea, aunque simplificada en este ejemplo, está en la base misma del meta-análisis, y es similar a lo que comúnmente se representa en un diagrama de embudo (*funnel plot*). **A.** Distribución cuando no hay sesgo de publicación. **B.** Distribución de resultados en el caso hipotético de que algunos estudios con efectos bajos no fueran publicados (puntos naranja claros). Debido a esto, un meta-análisis estaría sesgado, y el efecto estimado (línea gris punteada) sería mayor al efecto real (línea gris sólida).

Por suerte, existen técnicas poderosas para buscar evidencia de sesgos de publicación en los estudios incluidos en un meta-análisis, y hacer estimaciones corregidas del efecto *real* que tengan estos sesgos en cuenta para obtener resultados más precisos.

1.3. Motivación

Teniendo en cuenta la importancia de estas técnicas, y la relativa escasez de material de buena calidad en español, accesible tanto para estudiantes como para investigadores establecidos, he decidido hacer esta guía. He incluido información actualizada, soportada siempre en fuentes de alta calidad que pueden ser consultadas por las personas que quieran profundizar en cada tema.

A lo largo del documento daré ejemplos e instrucciones precisas de cómo hacer un meta-análisis básico en R (sección 3), incluyendo cómo hacer análisis de heterogeneidad (sección 3.2) y diagnósticos de influencia (sección 3.3 y, para técnicas más avanzadas, sección 3.8), creación de gráficos de bosque (*forest plots*; sección 3.4) y gráficos de embudo (*funnel plots*; sección 3.5), así como estrategias para detectar posibles sesgos de publicación (sección 3.6), y para

determinar el poder estadístico de un meta-análisis (sección 3.7). Finalmente, mostraré técnicas de meta-regresión para investigar los factores que pueden influir en los resultados de los estudios incluidos en un meta-análisis (sección 4).

Adicionalmente, he incluido algunos apéndices, donde encontrarás información concreta sobre otras rutas alternativas para hacer meta-análisis en R (Apéndice A), incluyendo instrucciones para hacer una curva de valores *p* (*p-curve*; Apéndice A1), así como la manera correcta de citar paquetes de R (Apéndice B) y, con fines de garantizar la reproducibilidad de este documento, una lista de los paquetes usados para la creación de esta guía, con sus respectivas versiones (Apéndice C).

2. Base de datos de ejemplo

A lo largo de esta guía usaré la base de datos `dat.molloy2014`, tomada de Molloy et al. (2014). Esta base de datos viene incluida con el paquete `metafor` de R.

Básicamente, Molloy et al. (2014) estudiaron si existe una asociación entre la escrupulosidad (*conscientiousness*¹) y la adherencia a la medicación. En otras palabras, ¿las personas más *escrupulosas* tienden a cumplir más con la medicación prescrita?

Primero, debemos cargar los principales paquetes que usaré a lo largo de esta guía: `metafor` (Viechtbauer, 2010) y `metaviz` (Kossmeier et al., 2020) para hacer e ilustrar los resultados del meta-análisis, así como `dplyr` (Wickham et al., 2023) y `forcats` (Wickham, 2023) para manipular y organizar la base de datos, y `ggplot2` (Wickham, 2016) para hacer algunas gráficas más adelante².

```
library(metafor)
library(metaviz)
library(dplyr)
library(forcats)
library(ggplot2)
```

Como ya hemos cargado el paquete `metafor`, ya podemos cargar la base de datos `dat.molloy2014`. En este caso, para poder *llamarla* cuando sea necesario, la asignaré a un objeto que llamaré simplemente `dat`.

```
dat <- get(data(dat.molloy2014))
```

Tras asignar la base de datos a este objeto (`dat`), la base de datos se puede ver como *output*³ en la consola de R sencillamente usando como comando el nombre que le dimos al objeto al que la asignamos (en este caso, `dat`).

```
dat
```

Consola de R: Output 1

	authors	year	ni	ri	controls	design	a_measure
1	Axelsson et al.	2009	109	0.187	none	cross-sectional	self-report
2	Axelsson et al.	2011	749	0.162	none	cross-sectional	self-report
3	Bruce et al.	2010	55	0.340	none	prospective	other
4	Christensen et al.	1999	107	0.320	none	cross-sectional	self-report
5	Christensen & Smith	1995	72	0.270	none	prospective	other
6	Cohen et al.	2004	65	0.000	none	prospective	other
7	Dobbels et al.	2005	174	0.175	none	cross-sectional	self-report
8	Ediger et al.	2007	326	0.050	multiple	prospective	self-report
9	Insel et al.	2006	58	0.260	none	prospective	other
10	Jerant et al.	2011	771	0.010	multiple	prospective	other
11	Moran et al.	1997	56	-0.090	multiple	prospective	other

¹En el contexto de los rasgos de personalidad, la dimensión de la *escrupulosidad* (también llamada conciencia o “voluntad de logro”), se refiere al auto-control, incluyendo la planificación y ejecución de tareas. Para una definición detallada del concepto de escrupulosidad en este contexto, ver John y Srivastava (1999).

²Los paquetes `dplyr`, `forcats` y `ggplot2` hacen parte de `tidyverse` (Wickham et al., 2019), una colección de paquetes para ciencia de datos que comparten un mismo diseño y gramática. Por esto, en vez de cargar cada paquete independientemente, si instalas `tidyverse`, puedes cargar los paquetes de esta colección usando `library(tidyverse)`.

³En R y otros lenguajes de programación, se llama “*output*” a la *salida* o resultado visible de una función o proceso, en este caso en la consola (que en RStudio es normalmente el panel que está a la izquierda en la parte inferior).

```

12 O'Cleirigh et al. 2007 91 0.370    none    prospective self-report
13 Penedo et al. 2003 116 0.000    none cross-sectional self-report
14 Quine et al. 2012 537 0.150    none    prospective self-report
15 Stilley et al. 2004 158 0.240    none    prospective      other
16 Wiebe & Christensen 1997 65 0.040    none    prospective      other
c_measure meanage quality
1   other  22.00     1
2    NEO  53.59     1
3    NEO  43.36     2
4   other  41.70     1
5    NEO  46.39     2
6    NEO  41.20     2
7    NEO  52.30     1
8    NEO  41.00     3
9   other  77.00     2
10   NEO  78.60     3
11   NEO  57.20     2
12   NEO  37.90     2
13   NEO  39.20     1
14   other  69.00     2
15    NEO  46.20     3
16    NEO  56.00     1

```

2.1. Variables de la base de datos

Por favor mira la base de datos, y familiarízate con sus columnas. ¿Qué información contiene cada variable? Explicaré cada una a continuación.

Primero, hay unas columnas que contienen información absolutamente necesaria para poder hacer un meta-análisis:

- **authors**: los autores de cada estudio a meta-analizar
- **year**: año de publicación de cada estudio
- **ni**: tamaño de muestra de cada estudio
- **ri**: coeficiente de correlación de Pearson de cada estudio

Esas son las variables más importantes y son necesarias para hacer un meta-análisis de correlación simple: alguna columna o columnas con información que permita identificar cada estudio (en este caso **authors** y **year**), una columna con el tamaño de muestra de cada estudio (en este caso, **ni**), y finalmente una columna con los coeficientes de correlación (en este caso la columna **ri**). Si tu meta-análisis no requiere moderadores (ver sección 4), con una estructura como esta sería suficiente.

Después hay unas columnas que contienen variables específicas para este ejemplo, y que son una serie de posibles moderadores que agrupan características de los estudios a meta-analizar:

- **controls**: cantidad de variables controladas (“**none**”: ninguna o “**multiple**”: múltiples)
- **design**: si se utilizó un diseño transversal (“**cross-section**”) o prospectivo (“**prospective**”) (para más información acerca de estos tipos de diseño, ver por ejemplo Manterola et al., 2019)
- **a_measure**: tipo de medida de adherencia (“**self-report**”: autoinforme u “**other**”: otro tipo de medida más *objetivo*)
- **c_measure**: tipo de medida de escrupulosidad (si se midió con alguna versión del inventario de personalidad NEO o con alguna otra escala)
- **meanage**: edad promedio de la muestra de cada estudio
- **quality**: calidad metodológica (la calidad metodológica fue calificada por los autores del meta-análisis en una escala de 1 a 4, donde las puntuaciones más altas indican una mayor calidad; para información respecto a cómo se obtuvo esta puntuación, ver el artículo original de Molloy et al. (2014)).

Por supuesto, el *output* de la consola no es la forma más clara de ver la base de datos, pero para una versión más legible, se puede usar la función **View**, y el nombre de la base de datos o tabla como argumento (en este caso

`View(dat)`). Sin embargo, de aquí en adelante mostraré algunas tablas en un formato de impresión (y no como sale en la consola de R), para que sean más fáciles de leer.

Voy a volver a cargar la base de datos (sobrescribiendo el objeto `dat`), para organizarla un poco mejor. Primero, agregaré una nueva columna llamada `study_id`, en la que numeraré los estudios del 1 al 16, lo que será útil para identificar cada estudio en algunas gráficas y análisis. Después, reorganizaré las columnas para que `study_id` sea la primera, en vez de la última columna. Finalmente, las columnas de variables que son factores (`controls`, `design`, `a_measure`, `c_measure` y `quality`), pueden ser definidas como tal, para evitar pasos adicionales más adelante. En este caso, además, voy a reorganizar los niveles de la variable `controls` (primero `none` y después `multiple`); esto, aunque no es un paso necesario, puede ayudar con la interpretación de resultados.

```
dat <- get(data(dat.molloy2014)) |>
  mutate(study_id = 1:16) |> # crear columna 'study_id' y agregar números del 1 al 16
  select(study_id, authors:quality) |> # mover 'study_id' como primera columna
  mutate_at(c("controls", # transformar variables relevantes en factores
            "design",
            "a_measure",
            "c_measure",
            "quality"),
            as.factor) |>
  mutate(controls = fct_relevel(controls, # reorganizar niveles de la variable 'controls'
                                "none", "multiple"))
```

Con esto, la base de datos tiene ahora la siguiente estructura (Tabla 1):

Tabla 1

Estructura de la base de datos con estudios numerados

study_id	authors	year	ni	ri	controls	design	a_measure	c_measure	meanage	quality
1	Axelsson et al.	2009	109	0.187	none	cross-sectional	self-report	other	22.00	1
2	Axelsson et al.	2011	749	0.162	none	cross-sectional	self-report	NEO	53.59	1
3	Bruce et al.	2010	55	0.340	none	prospective	other	NEO	43.36	2
4	Christensen et al.	1999	107	0.320	none	cross-sectional	self-report	other	41.70	1
5	Christensen & Smith	1995	72	0.270	none	prospective	other	NEO	46.39	2
6	Cohen et al.	2004	65	0.000	none	prospective	other	NEO	41.20	2
7	Dobbels et al.	2005	174	0.175	none	cross-sectional	self-report	NEO	52.30	1
8	Ediger et al.	2007	326	0.050	multiple	prospective	self-report	NEO	41.00	3
9	Insel et al.	2006	58	0.260	none	prospective	other	other	77.00	2
10	Jerant et al.	2011	771	0.010	multiple	prospective	other	NEO	78.60	3
11	Moran et al.	1997	56	-0.090	multiple	prospective	other	NEO	57.20	2
12	O' Cleirigh et al.	2007	91	0.370	none	prospective	self-report	NEO	37.90	2
13	Penedo et al.	2003	116	0.000	none	cross-sectional	self-report	NEO	39.20	1
14	Quine et al.	2012	537	0.150	none	prospective	self-report	other	69.00	2
15	Stilley et al.	2004	158	0.240	none	prospective	other	NEO	46.20	3
16	Wiebe & Christensen	1997	65	0.040	none	prospective	other	NEO	56.00	1

Nota: Datos tomados de Molloy et al. (2014).

2.2. Preparación de los datos: transformación de coeficientes r de Pearson a z de Fisher

Los coeficientes de Pearson no se distribuyen normalmente, lo que podría llevar a calcular varianzas incorrectas, especialmente cuando se trata de correlaciones con tamaños de muestra pequeño (Shadish & Haddock, 2009). Por esto, lo mejor es transformar los coeficientes r de Pearson a z de Fisher (1915), que no tienen este problema⁴.

⁴Para decirlo de manera más precisa, el problema es que la distribución del coeficiente de correlación de Pearson (r), es muy sesgada cuando se trata de variables altamente correlacionadas (positiva, o negativamente). Esto dificulta la estimación de los intervalos de confianza y por tanto la aplicación de las pruebas de significación para coeficientes r (e.g., Shadish & Haddock, 2009). La transformación de Fisher de valores r a z —que es la [tangente hiperbólica inversa](#) de r — resuelve este problema, pues los coeficientes z tienen una distribución aproximadamente normal, y una varianza estable a lo largo de diferentes valores posibles de r (para una demostración en español, ver Sánchez-Bruno & Borges del Rosal, 2005).

Para transformar los coeficientes r de Pearson a coeficientes z de Fisher, usaré la función `escalc` del paquete `metafor`, que sirve para calcular diversos tamaños de efecto que se utilizan comúnmente en meta-análisis. Los argumentos que requiere esta función, además del tipo de transformación a realizar (en este caso `measure = "ZCOR"`), son los coeficientes de correlación (`ri`), el tamaño de muestra de cada correlación (`ni`), y la base de datos que contiene estos valores (`data`). En nuestro caso, las columnas donde están estos valores, tienen los mismos nombres (`ri, ni`). En este ejemplo, asignaré el resultado de esta función al mismo objeto `dat`, que contiene la base de datos, para sobrescribirlo y no crear objetos adicionales.

```
dat <- escalc(measure = "ZCOR", ri = ri, ni = ni, data = dat)
```

Esta función agrega dos nuevas variables: `yi`, que es el tamaño de efecto (en valores z de Fisher), y `vi` que es la varianza (Tabla 2).

Tabla 2

Estructura de la base de datos, con transformación de los r de Pearson a z de Fisher

study_id	authors	year	ni	ri	controls	design	a_measure	c_measure	meanage	quality	yi	vi
1	Axelsson et al.	2009	109	0.187	none	cross-sectional	self-report	other	22.00	1	0.1892266	0.0094340
2	Axelsson et al.	2011	749	0.162	none	cross-sectional	self-report	NEO	53.59	1	0.1634399	0.0013405
3	Bruce et al.	2010	55	0.340	none	prospective	other	NEO	43.36	2	0.3540925	0.0192308
4	Christensen et al.	1999	107	0.320	none	cross-sectional	self-report	other	41.70	1	0.3316471	0.0096154
5	Christensen & Smith	1995	72	0.270	none	prospective	other	NEO	46.39	2	0.2768638	0.0144928
6	Cohen et al.	2004	65	0.000	none	prospective	other	NEO	41.20	2	0.0000000	0.0161290
7	Dobbels et al.	2005	174	0.175	none	cross-sectional	self-report	NEO	52.30	1	0.1768200	0.0058480
8	Ediger et al.	2007	326	0.050	multiple	prospective	self-report	NEO	41.00	3	0.0500417	0.0030960
9	Insel et al.	2006	58	0.260	none	prospective	other	other	77.00	2	0.2661084	0.0181818
10	Jerant et al.	2011	771	0.010	multiple	prospective	other	NEO	78.60	3	0.0100003	0.0013021
11	Moran et al.	1997	56	-0.090	multiple	prospective	other	NEO	57.20	2	-0.0902442	0.0188679
12	O'Clairigh et al.	2007	91	0.370	none	prospective	self-report	NEO	37.90	2	0.3884231	0.0113636
13	Penedo et al.	2003	116	0.000	none	cross-sectional	self-report	NEO	39.20	1	0.0000000	0.0088496
14	Quine et al.	2012	537	0.150	none	prospective	self-report	other	69.00	2	0.1511404	0.0018727
15	Stilley et al.	2004	158	0.240	none	prospective	other	NEO	46.20	3	0.2447741	0.0064516
16	Wiebe & Christensen	1997	65	0.040	none	prospective	other	NEO	56.00	1	0.0400214	0.0161290

Nota: Las nuevas columnas creadas usando la función `escalc` (`yi` como tamaño de efecto y `vi` como varianza) están resaltadas en naranja

3. Cómo hacer el meta-análisis

Para hacer el meta-análisis, usaré la función `rma` del paquete `metafor`. Esta función requiere especificar los tamaños de efecto (`yi`) y varianzas (`vi`) de los estudios a meta-analizar. En nuestro caso, las columnas donde están estos valores, tienen los mismos nombres (`yi, vi`). Asignaré los resultados del meta-análisis a un nuevo objeto llamado `res`.

```
res <- rma(yi = yi, vi = vi, data = dat)
```

Los resultados, son los siguientes:

```
res
```

Consola de R: Output 2

```
Random-Effects Model (k = 16; tau^2 estimator: REML)

tau^2 (estimated amount of total heterogeneity): 0.0081 (SE = 0.0055)
tau (square root of estimated tau^2 value):      0.0901
I^2 (total heterogeneity / total variability):   61.73%
H^2 (total variability / sampling variability): 2.61

Test for Heterogeneity:
Q(df = 15) = 38.1595, p-val = 0.0009

Model Results:

estimate      se     zval    pval   ci.lb   ci.ub
0.1499  0.0316  4.7501 <.0001  0.0881  0.2118 ***
```

 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

3.1. Interpretación de los resultados del meta-análisis

Vamos a analizar estos resultados de la consola de R por partes:

Primero, nos confirman que ajustamos un modelo con efectos aleatorios (**Random-Effects Model**), a partir de 16 estudios ($k = 16$), y que para estimar τ^2 (tau cuadrado) usamos el método de **máxima verosimilitud restringida**⁵ (τ^2 estimator: **REML**), que se designa como *REML* por sus siglas en inglés .

Posteriormente, nos provee los valores de una serie de estimadores de heterogeneidad:

- τ^2 : tau^2 (estimated amount of total heterogeneity): 0.0081 (SE = 0.0055)
- τ : tau (square root of estimated tau^2 value): 0.0901
- I^2 : I^2 (total heterogeneity / total variability): 61.73%, y
- H^2 : H^2 (total variability / sampling variability): 2.61

La tercera parte, reporta otra prueba de heterogeneidad, usando el estadístico Q :

- Test for Heterogeneity:

$Q(df = 15) = 38.1595$, p-val = 0.0009

De todos estos, los más comúnmente reportados son τ^2 , τ , I^2 y Q . Cada una de estas medidas tiene ventajas y desventajas, por lo cual tiene sentido reportarlas todas.

- I^2 : tiene la ventaja de ser sencillo de interpretar, pues hay criterios generales para heterogeneidad baja, moderada y alta (típicamente 25%, 50%, and 75%, respectivamente). Sin embargo, es muy sensible a los tamaños de muestra de los estudios meta-analizados (por ejemplo, si en tu meta-análisis hay estudios con tamaños de muestra muy grandes, esto va a sesgar tu I^2).
- Q : aunque no es sensible al tamaño de muestra, es sensible al número de estudios meta-analizados. Tiene la ventaja de ser un test de hipótesis, y como tal, puede ser interpretado a partir de su valor p .
- τ^2 : no tiene problemas de sensibilidad a los tamaños de muestra o número de estudios meta-analizados, pero es más difícil de interpretar. τ^2 es una estimación de la varianza de los tamaños de los efectos reales entre los estudios meta-analizados. Se usa, principalmente, para asignar pesos a cada estudio (para más información, ver Borenstein et al., 2009).

En nuestro caso, el estadístico Q sugiere que hay una heterogeneidad significativa en los estudios meta-analizados ($p = 0.0009$). I^2 , sugiere una heterogeneidad moderada, lo que quiere decir que se estima que más de la mitad (61.73 %) de la varianza se deriva de diferencias en los tamaños de efecto.

Más abajo, el *output* de la consola de R nos muestra los resultados de nuestro meta-análisis (**Model results**); en otras palabras, ¿cuál es el tamaño de efecto de la asociación entre escrupulosidad (*conscientiousness*) y la adherencia a la medicación, según nuestro meta-análisis?

Esta parte nos provee varios resultados:

- **Estimate** (0.1499): estimado de la correlación entre escrupulosidad y adherencia a la medicación, en este caso en valores z de Fisher, pues así transformamos los coeficientes de cada estudio
- **se** (0.0316): error estándar del estimado de la asociación (en valores z de Fisher)
- **zval** (4.7501): estadístico Z (mayúscula) que comprueba la media de una distribución. No se debe confundir con la transformación de coeficientes de correlación a z de Fisher (minúscula); este estadístico no nos provee una estimación de la asociación entre las variables correlacionadas, sino, de manera similar a una prueba t ,

⁵Hay varios métodos disponibles como estimador, además de **máxima verosimilitud restringida** (REML). Sin embargo, si tienes dudas, REML es una buena opción. Cada método tiene ventajas y desventajas que, si tienes interés en mirar, están descritas en la documentación de la función **rma**.

nos sugiere si nuestra media (para el caso, el resultado de nuestro meta-análisis), se diferencia de 0 (o una correlación nula). Cuando Z es mayor a 1.96 (o menor a -1.96), nuestro resultado está en el 5% extremo de la distribución Z y sería significativo con un α tradicional de 0.05 (dos colas)

- **pval (<.0001):** valor p de la correlación meta-analizada
- **ci.lb (0.0881):** límite inferior del intervalo de confianza (*confidence interval lower bound*) de la correlación meta-analizada (en valores z de Fisher)
- **ci.ub (0.2118):** límite superior del intervalo de confianza (*confidence interval upper bound*) de la correlación meta-analizada (en valores z de Fisher)
- Nivel de significación (***) representación con asteriscos (o un punto) del nivel de significación
- **Signif. codes:** 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1: Clave para interpretar los niveles de significación. Aunque puede parecer complejo, básicamente, quiere decir que tres asteriscos (***) representan un valor p entre 0 y 0.001 (lo que comúnmente se representa como $p < .001$); dos asteriscos (**) un valor p entre 0.001 y 0.01 ($p < .01$); un asterisco (*) un valor p entre 0.01 y 0.05 ($p < .05$); un punto(.) un valor p entre 0.05 y 0.1 ($p < .1$, que ya no es significativo); y si no hay ningún símbolo, un valor p entre 0.1 y 1 ($p > .1$, no significativo)

En este caso, el meta-análisis nos sugiere que en efecto existe una asociación positiva entre escrupulosidad y adherencia a la medicación (coeficiente de correlación transformado en z de Fisher = .1499), con un error estándar de 0.0316. Así mismo, sugiere que esa asociación es significativa ($Z = 4.7501, p < .0001$), y nos muestra el intervalo de confianza del 95% (o IC 95%); los intervalos de confianza (en este caso del 95%) lo que estiman es que, si hicieramos 100 muestras independientes, 95 de estas contendrían una asociación que estaría entre los límites inferior ($z = .0881$) y superior ($z = .2118$) de estos intervalos de confianza.

3.1.1. Reporte del meta-análisis básico

Esto se podría resumir, por ejemplo, como:

Ejemplo de reporte básico

Este meta-análisis examinó la relación entre la escrupulosidad y la adherencia a la medicación a través de 16 estudios. Los resultados indicaron una asociación significativa y positiva entre ambos factores ($z \pm se = 0.15 \pm 0.032$, IC 95% [0.09, 0.21]; $Z = 4.75, p < .0001$). Sin embargo, se encontró una moderada pero significativa heterogeneidad en cuanto al tamaño del efecto entre los estudios incluidos ($\tau^2 \pm se = 0.0081 \pm 0.0055$; $\tau = 0.0901$; $Q(15) = 38.16, p < .001$; $I^2 = 61.7\%$). Por tanto, es importante interpretar los resultados con precaución debido a la amplia variabilidad en el tamaño del efecto entre los estudios. A pesar de estas limitaciones, los hallazgos sugieren que la escrupulosidad puede tener un efecto positivo en la adherencia a la medicación.

Siempre es buena idea citar tablas y figuras relevantes junto al reporte de los resultados que ilustran. Aunque aún no hemos creado ninguna figura, es buena idea al menos citar (e incluir) un *Forest plot* (diagrama de bosque; 3.4), y un *Funnel plot* (diagrama de embudo; sección 3.5), elementos que explicaré más adelante.

3.1.2. Alternativa: reportar el estimado como r de Pearson en vez de z de Fisher

Para reportar la correlación, si prefieres reportar coeficientes r de Pearson en vez de la transformación a z de Fisher, puedes transformar los valores z de Fisher de vuelta a r de Pearson. Para esto existen múltiples opciones en R, incluyendo simplemente usar la función **tanh**, que calcula la tangente hiperbólica, o la función **fisherz2r** del paquete **psych** (Revelle, 2023). Por ejemplo, para transformar el estimado de nuestro meta-análisis a r de Pearson, sólo debo usar alguna de esas funciones, y agregar el valor z (0.1499 en nuestro caso) como único argumento:

```
tanh(0.1499)
```

Consola de R: Output 3

```
[1] 0.1487872
```

```
library(psych)
fisherz2r(0.1499)
```

Consola de R: Output 4

```
[1] 0.1487872
```

Cualquiera de estas opciones nos da un valor de coeficiente de correlación de Pearson ($r = 0.1487872$), muy similar al obtenido como z de Fisher ($z = 0.1499$). Esto se debe a que, para coeficientes r de Pearson entre -0.4 y 0.4, la transformación a valores z de Fisher produce resultados muy similares (Fig. 4).

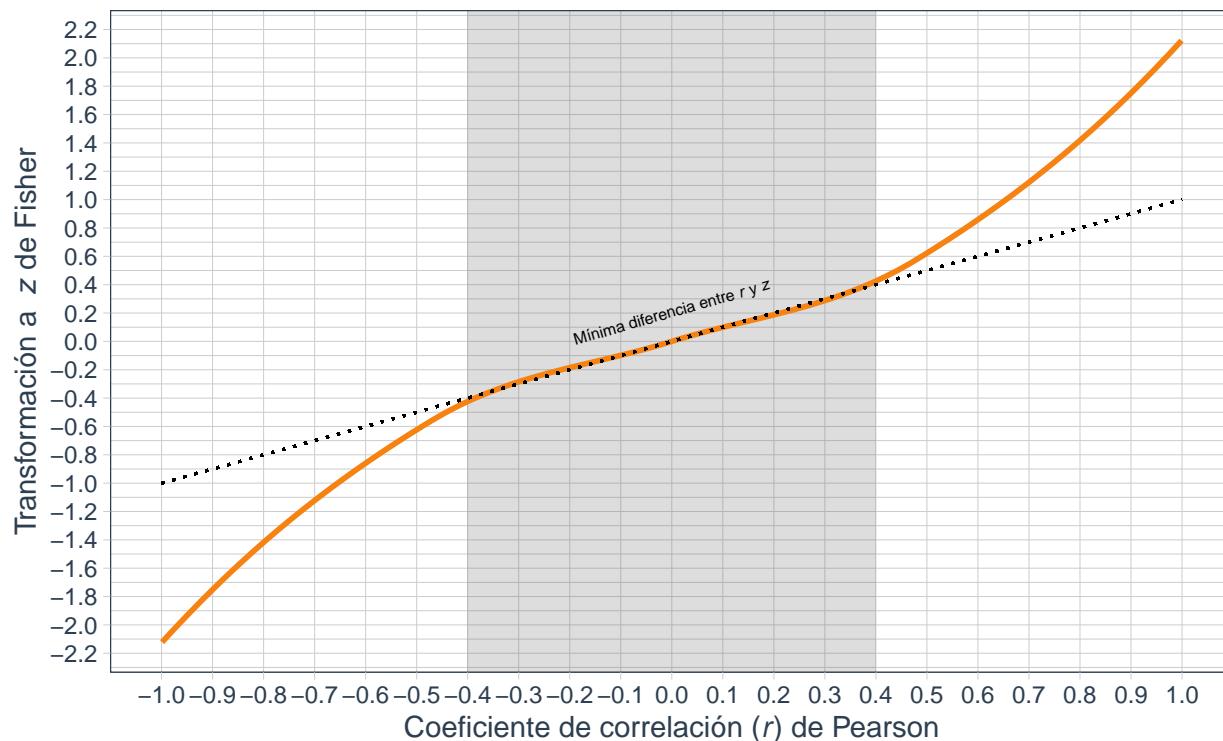


Figura 4. Asociación entre coeficientes de correlación r de Pearson (eje X), y su transformación a z de Fisher (eje Y). La línea naranja representa la asociación entre valores r y z ; como referencia, la línea negra punteada representa igualdad entre ejes ($y = x$). Como se puede ver, cuando r está aproximadamente entre -0.4 y 0.4 (rectángulo gris), los valores r y z son casi idénticos. Para valores más extremos, el valor de z se aleja progresivamente del valor de r .

Por supuesto, si decides reportar los resultados de tu meta-análisis en coeficientes r de Pearson, siempre puedes hacer lo mismo con el error estándar y los límites del intervalo de confianza del 95 % (todos valores en z de Fisher, pues fue el tamaño de efecto que meta-analizamos).

3.2. Más información sobre heterogeneidad

Es importante tener en cuenta que la heterogeneidad no es un *supuesto* que se deba cumplir al hacer un meta-análisis, y por ende una heterogeneidad moderada o alta no invalida sus resultados. Sencillamente es información útil que se debe reportar y tener en cuenta al interpretar el resultado de un meta-análisis. En este caso, la presencia de heterogeneidad sugiere que los estudios meta-analizados varían y no suelen reportar resultados similares.

De manera general, y para decirlo de manera más técnica, la presencia de heterogeneidad estadística es indicativa de una variación entre los estudios en la magnitud y la dirección de la estimación del efecto estudiado (para más información y ejemplos, ver Sedgwick, 2015).

Por esto, reportar información detallada acerca de la heterogeneidad de los estudios meta-analizados es siempre útil. De hecho, además de reportar los estadísticos τ^2 , τ , I^2 y Q (como expliqué en la sección 3.1), podemos fácilmente calcular los intervalos de confianza para τ^2 , τ , e I^2 (además de H^2 , que no he usado) con la función `confint`.

```
confint(res)
```

Consola de R: Output 5

	estimate	ci.lb	ci.ub
tau^2	0.0081	0.0017	0.0378
tau	0.0901	0.0412	0.1944
I^2(%)	61.7324	25.2799	88.2545
H^2	2.6132	1.3383	8.5139

Para τ^2 , el hecho de que los límites del intervalo de confianza no crucen el 0 (es decir, no hay un límite negativo y otro positivo; en nuestro caso, ambos son positivos: IC 95% [0.0017, 0.0378]), también sugiere que hay heterogeneidad entre los estudios que meta-analizamos.

3.2.1. Reporte del meta-análisis con intervalos de confianza para medidas de heterogeneidad

Estos intervalos de confianza que también pueden ser reportados junto a sus correspondientes estadísticos. Entonces, esto se podría resumir, por ejemplo, como:

Ejemplo de reporte con intervalos de confianza para heterogeneidad

El meta-análisis incluyó 16 estudios que examinaron la relación entre la escrupulosidad y la adherencia a la medicación. Los estudios incluidos variaron en cuanto al diseño, la población de pacientes y la intervención evaluada. Los resultados del meta-análisis indicaron una asociación positiva significativa entre la escrupulosidad y la adherencia a la medicación ($z \pm se = 0.15 \pm 0.032$, IC 95% [0.09, 0.21]; $Z = 4.75, p <.0001$). La heterogeneidad en torno al tamaño medio del efecto entre los estudios incluidos fue moderada pero significativa ($\tau^2 \pm se = 0.0081 \pm 0.0055$, IC 95% [0.0017, 0.0378]; $\tau = 0.0901$, IC 95% [0.0412, 0.1944]; $Q(15) = 38.16, p <.001$; $I^2 = 61.7\%$, IC 95% [25.28%, 88.25%]). Se observó una amplia variabilidad en el tamaño del efecto entre los estudios, lo que sugiere que los resultados deben interpretarse con precaución. A pesar de estas limitaciones, los hallazgos sugieren que es probable que la escrupulosidad tenga un efecto positivo en la adherencia a la medicación. Es importante tener en cuenta que la calidad metodológica de los estudios incluidos en el meta-análisis fue variable, lo que puede afectar la validez y generalización de las conclusiones.

De nuevo, recuerda que siempre es buena idea citar tablas y figuras relevantes junto al reporte de los resultados que ilustran. Aunque aún no hemos creado ninguna figura, al reportar un meta-análisis es buena idea al menos citar (e incluir) un *Forest plot* (diagrama de bosque; 3.4), y un *Funnel plot* (diagrama de embudo; sección 3.5). Más adelante explicaré cómo crear e interpretar estas figuras.

3.3. Diagnóstico de influencia

Otro aspecto importante de un meta-análisis, es determinar si alguno o algunos de los estudios meta-analizados es o son particularmente influyentes en nuestro resultado⁶. Para esto, se suele usar la técnica conocida como *leave-one-out*, que se refiere al resultado del meta-análisis cuando se excluye cada estudio; al estimar cómo y cuánto cambia el resultado del modelo de meta-análisis al excluir cada estudio, podemos estimar su influencia en el resultado. Dicho de otra manera, si al excluir un estudio el resultado cambia mucho, sabemos que ese estudio tiene gran influencia en el meta-análisis y sería mejor excluirlo.

Para hacer un análisis de influencia, podemos usar la función `influence` del paquete `metafor`, cuyo resultado, en este caso, asignaré a un objeto llamado `inf.res`.

```
inf.res <- influence(res)
```

Dado que lo asigné a un objeto (`inf.res`), para ver el resultado, debo usar como comando el nombre que le di al objeto.

```
inf.res
```

⁶Por ejemplo, si estuviésemos meta-analizando 20 estudios, de los cuales 19 tienen un n de 100, pero el otro tiene un n de 10.000, este último tendrá una influencia enorme en nuestro resultado. Sería preocupante que nuestro meta-análisis sea dependiente principalmente de un único estudio.

Consola de R: Output 6

	rstudent	dffits	cook.d	cov.r	tau2.del	QE.del	hat	weight	dfbs	inf
1	0.2918	0.0485	0.0025	1.1331	0.0091	37.7109	0.0568	5.6776	0.0481	
2	0.1196	-0.0031	0.0000	1.2595	0.0100	36.7672	0.1054	10.5396	-0.0032	
3	1.2740	0.2595	0.0660	0.9942	0.0075	35.3930	0.0364	3.6432	0.2623	
4	1.4711	0.3946	0.1439	0.9544	0.0068	33.5886	0.0562	5.6195	0.3994	
5	0.8622	0.1838	0.0339	1.0505	0.0082	36.5396	0.0441	4.4069	0.1837	
6	-0.9795	-0.2121	0.0455	1.0639	0.0084	37.1703	0.0411	4.1094	-0.2112	
7	0.2177	0.0296	0.0010	1.1740	0.0094	37.6797	0.0714	7.1362	0.0296	
8	-0.9774	-0.3120	0.1001	1.1215	0.0084	36.1484	0.0889	8.8886	-0.3128	
9	0.7264	0.1392	0.0195	1.0561	0.0083	37.0495	0.0379	3.7886	0.1387	
10	-1.8667	-0.5861	0.2198	0.8502	0.0047	25.0661	0.1058	10.5826	-0.5430	
11	-1.4985	-0.2771	0.0756	1.0073	0.0077	35.6617	0.0369	3.6922	-0.2791	
12	1.8776	0.4918	0.2148	0.8819	0.0059	31.9021	0.0511	5.1150	0.5059	
13	-1.1892	-0.2939	0.0859	1.0550	0.0080	36.3291	0.0587	5.8732	-0.2941	
14	-0.0020	-0.0423	0.0021	1.2524	0.0100	37.7339	0.0998	9.9778	-0.0434	
15	0.8066	0.2126	0.0459	1.0907	0.0083	35.8385	0.0684	6.8403	0.2125	
16	-0.7160	-0.1656	0.0280	1.0853	0.0087	37.7017	0.0411	4.1094	-0.1642	

Esto me muestra gran cantidad de datos de cada estudio (en este caso, lo presento como una tabla sin formato, tal cual se ve en la consola de R), y contiene información que puede parecer muy compleja; sin embargo, como verás, la última columna resume la interpretación general de estos resultados de influencia, facilitando muchísimo su interpretación.

Las columnas que incluye este *output* son las siguientes:

- **rstudent**: residuos estandarizados externamente. Esto **no** corresponde al coeficiente de correlación de cada estudio, sino a la diferencia entre el tamaño de efecto observado en cada estudio, y la predicción de dicho valor cuando dicho estudio se elimina del meta-análisis
- **dffits**: diferencia de ajuste(s) (en inglés *difference in fit(s)*). Es una medida diagnóstica de la influencia de cada punto en una regresión (en este caso, cada estudio en un meta-análisis) propuesta originalmente por Belsley et al. (1980)
- **cook.d**: Distancia de Cook (en inglés, *Cook's distance*). Es otra medida diagnóstica de la influencia, propuesta originalmente por Cook (1977). Es conceptualmente idéntica a DFFITS (**dffits**), y de hecho existe una fórmula para convertir una medida en la otra (Henry et al., 2003). Aunque no hay un acuerdo absoluto al respecto, comúnmente se asume que valores mayores a 1 ($D_i > 1$), representan puntos (o, en este caso, estudios) influyentes en un modelo (Cook & Weisberg, 1982), que probablemente deban ser excluidos
- **cov.r**: relación (o proporción) de covarianza. Es el determinante de la matriz de varianza-covarianza de las estimaciones de los parámetros basadas en el conjunto de datos, cuando cada estudio se elimina del meta-análisis, dividido por el mismo determinante de la matriz de varianza-covarianza cuando se incluyen todos los estudios. Sé que suena complejo, pero básicamente, un valor menor a 1 indica que la eliminación de ese estudio produce estimaciones más precisas de los coeficientes del modelo
- **tau2.del**: es la heterogeneidad (residual) τ^2 cuando se elimina cada estudio
- **QE.del**: similar al resultado anterior, este se refiere al estadístico *Q* obtenido cuando se excluye cada estudio
- **hat**: los valores *hat* (h) son una medida estandarizada de la distancia de cada punto a la media de la variable predictora. Mientras valores cercanos a 0 indican que no hay un valor preocupante, valores cercanos a 1, aunque no indican directamente alta influencia de algún punto, ciertamente nos indican que vale la pena investigar más. Los valores *hat* están abiertos a la interpretación, pero un valor de corte que es común es el doble de la media de los todos valores *hat* (\bar{h}); cualquier estudio con un valor mayor debe ser examinado con cuidado
- **weight**: peso asignado a cada estudio
- **dfbs**: el valor de **dfbs** (o *DFBETAS*) indica cuántas desviaciones estándar cambia el coeficiente estimado después de excluir cada estudio del modelo de meta-análisis
- **inf**: por suerte, esta columna **inf** resume esta información por nosotros. Cualquier estudio que se considere influyente, teniendo en cuenta la diferencia de ajuste, la distancia de Cook, los valores *hat* o *DFBETAS*, será

señalado acá como influyente, usando asteriscos

Aunque hay mucha información, lo más importante ahora es mirar la última columna, llamada `inf`. Si ahí aparecieran asteriscos para algún estudio meta-analizado (que no es nuestro caso), sugeriría que ese estudio es particularmente influyente y podría ser necesario eliminarlo del meta-análisis.

Por último, podemos también ver esta información que tenemos guardada en el objeto `inf.res`, de manera gráfica, usando la función `plot` (Fig. 5).

```
plot(inf.res)
```

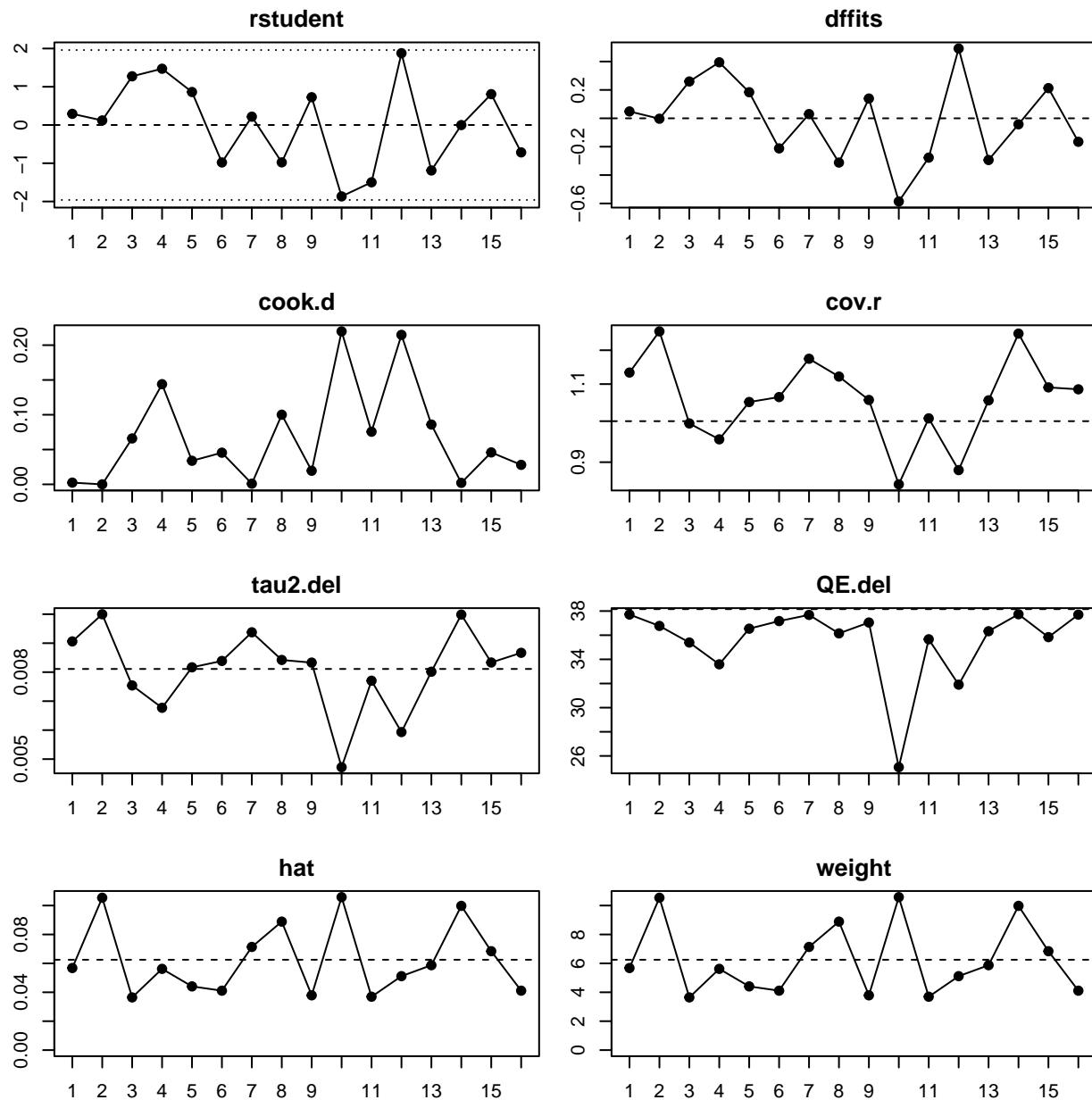


Figura 5. Diagnóstico de influencia. Estudios particularmente influyentes serían representados con un punto rojo. Los números 1 a 16 en el eje X representan cada estudio, como lo definimos en columna `study_id` de la Tabla 1. En este caso, no hay ningún estudio que se considere demasiado influyente, por lo que este análisis sugiere que no es necesario excluir ningún estudio de nuestro meta-análisis.

Más adelante, en la sección 3.8, daré más información y técnicas más completas y avanzadas para identificar estudios atípicos que podrían sesgar nuestra estimación del efecto *real*.

3.4. Forest plot (diagrama de bosque)

Hacer un diagrama de bosque (*forest plot*) que resuma nuestro meta-análisis usando `metafor` es muy fácil; sólo tenemos que usar la función `forest`, usando como argumento el objeto al que asignamos los resultados de nuestro meta-análisis (en nuestro caso, `res`).

```
forest(res)
```

Esto produce la Figura 6.

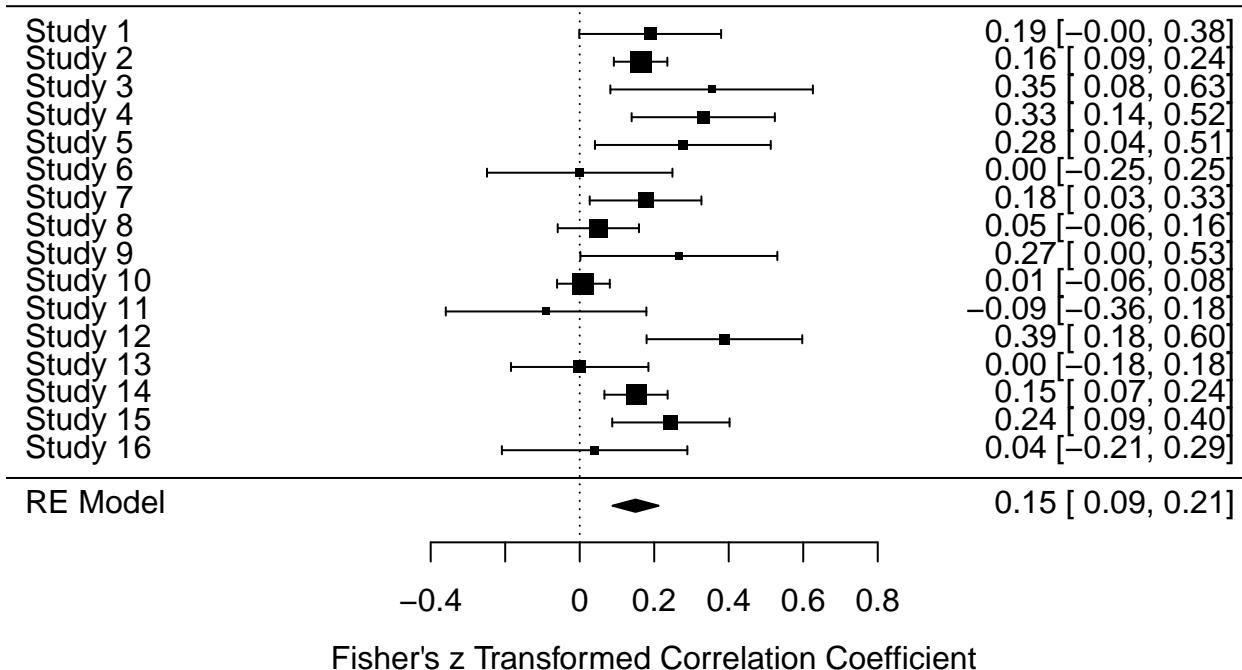


Figura 6. Forest plot básico de `metafor`. Para cada estudio meta-analizado, tenemos el efecto (correlación, en este caso en valores z de Fisher), así como sus intervalos de confianza entre corchetes. Esta misma información está representada gráficamente, con los cuadrados representando el efecto de cada estudio así como sus intervalos de confianza como barras de error, y el tamaño de muestra representado por el tamaño del cuadrado. Bajo estos resultados, tenemos nuestro meta-análisis, con el mismo formato en texto, pero representando el efecto y sus intervalos de confianza con un diamante.

Como se puede ver en las Figuras 6, 7 y 8 (que son versiones del mismo *forest plot*), no es una sorpresa que el análisis nos sugiera bastante heterogeneidad; las correlaciones encontradas entre los diferentes estudios varían mucho (están entre -0.09 y 0.37), y aunque son positivas en la mayoría de los casos, en algunos son prácticamente 0 o incluso negativas.

Para una versión más completa y anotada, también usando el *plot* básico de `metafor`, pero agregando encabezados de cada columna en español, nombres de los estudios meta-analizados⁷ así como una columna con los pesos dados a cada estudio, y detalles del modelo final⁸, podemos hacer algo como esto:

```
# forest plot con anotaciones adicionales
forest(res,
       cex = 0.75, xlim = c(-1.6, 1.6),
       slab = paste(dat$authors, dat$year, sep = ", "), showweights = TRUE,
       xlab = "Coeficiente de correlación (z de Fisher)", digits = c(2,3L),
       mlab = bquote(paste("Modelo EA: Q(", .(res$k - res$p), ") = ",
                           .(formatC(res$QE, digits = 2, format = "f")), ", p ",
                           .(scales::pvalue(res$pval)), "; ", I^2, " = ",
                           .(formatC(res$I2, digits = 1, format = "f"))), "%")))
```

⁷En este caso, dado que tenemos los autores y los años de publicación de cada estudio en columnas separadas, pegué las columnas `authors` y `year` separadas por una coma y un espacio: `paste(dat$authors, dat$year, sep = ", ")` como argumento `slab`.

⁸Estas opciones están explicadas [acá](#).

```
# agregar encabezados a las columnas (valores de X y Y deben ser ajustados)
op <- par(cex = 0.8, font = 2)
text(x = -1.6, y = 18, labels = "Autor(es), Año", pos = 4)
text(x = 0, y = 18, labels = "Efecto e IC", pos = 4)
text(x = 1, y = 18, labels = "Peso", pos = 2)
text(x = 1.6, y = 18, labels = "Corr. [95% IC]", pos = 2)
```

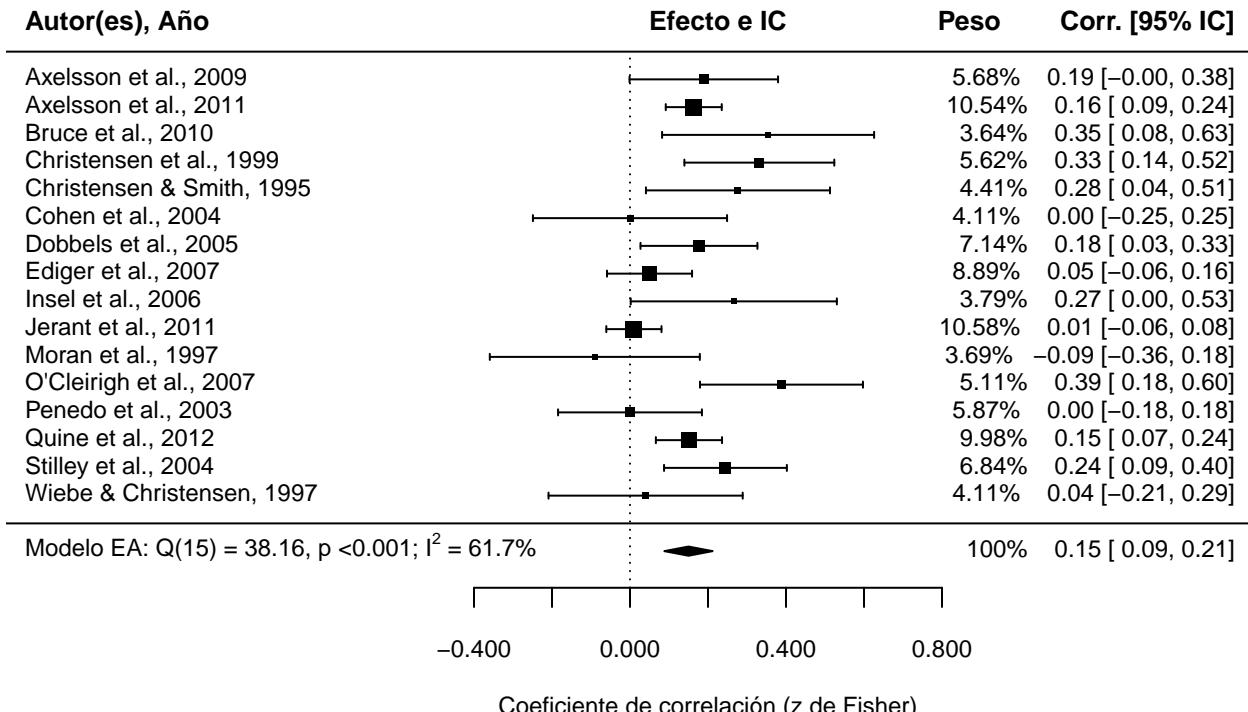


Figura 7. Forest plot anotado, creado con `metafor`. En esta versión agregué algunos encabezados en español, así como estadísticos generales del modelo de meta-análisis. Modelo EA se refiere al modelo meta-analizado, de efectos aleatorios.

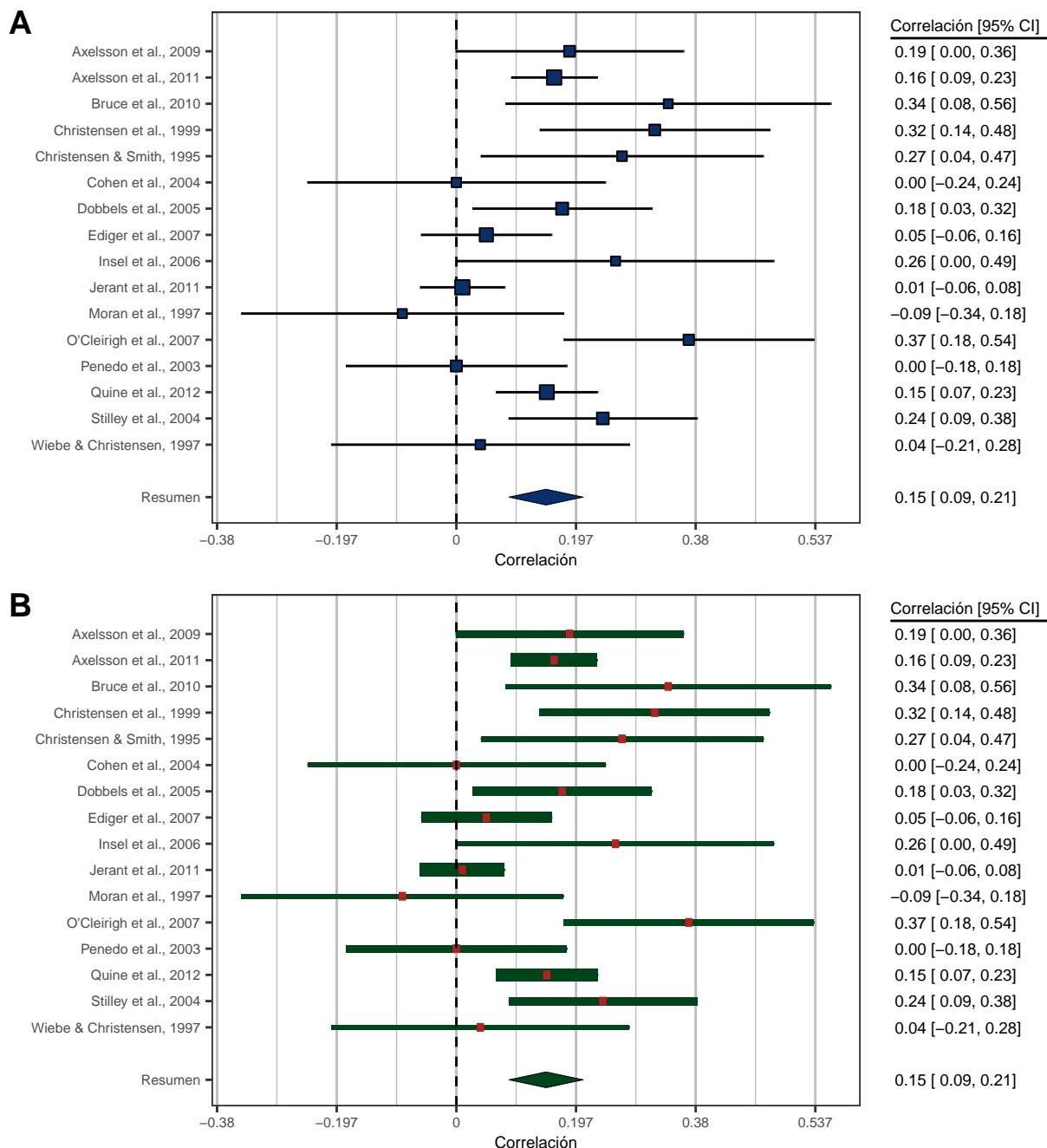
O, para un *forest plot* quizás más sofisticado, se puede usar la función `viz_forest` del paquete `metaviz`, en alguna de sus variantes y colores.

```
# A. Variante "classic" (no tiene que ser definida, pues es la opción por defecto)
viz_forest(res,
           study_labels = paste(dat$authors, dat$year, sep = ", "),
           xlab = "Correlación",
           annotate_CI = TRUE,
           summary_label = "Resumen",
           text_size = 2.6,
           x_trans_function = tanh)

# B. Variante "thick"
viz_forest(res,
           study_labels = paste(dat$authors, dat$year, sep = ", "),
           xlab = "Correlación",
           variant = "thick",
           col = "Greens",
           annotate_CI = TRUE,
           summary_label = "Resumen",
           text_size = 2.6,
           x_trans_function = tanh)
```

```
# C. Variante "rain"
viz_forest(res,
  study_labels = paste(dat$authors, dat$year, sep = ", "),
  xlab = "Correlación",
  variant = "rain",
  col = "Oranges",
  annotate_CI = TRUE,
  summary_label = "Resumen",
  text_size = 2.6,
  x_trans_function = tanh)
```

Con el código anterior genero las siguientes tres versiones del mismo *forest plot* (Fig. 8) usando diferentes variantes y escalas de colores, y transformando de vuelta los coeficientes de z de Fisher a r de Pearson (con el argumento `x_trans_function = tanh`). Por supuesto, es cuestión de gusto cuál usar.



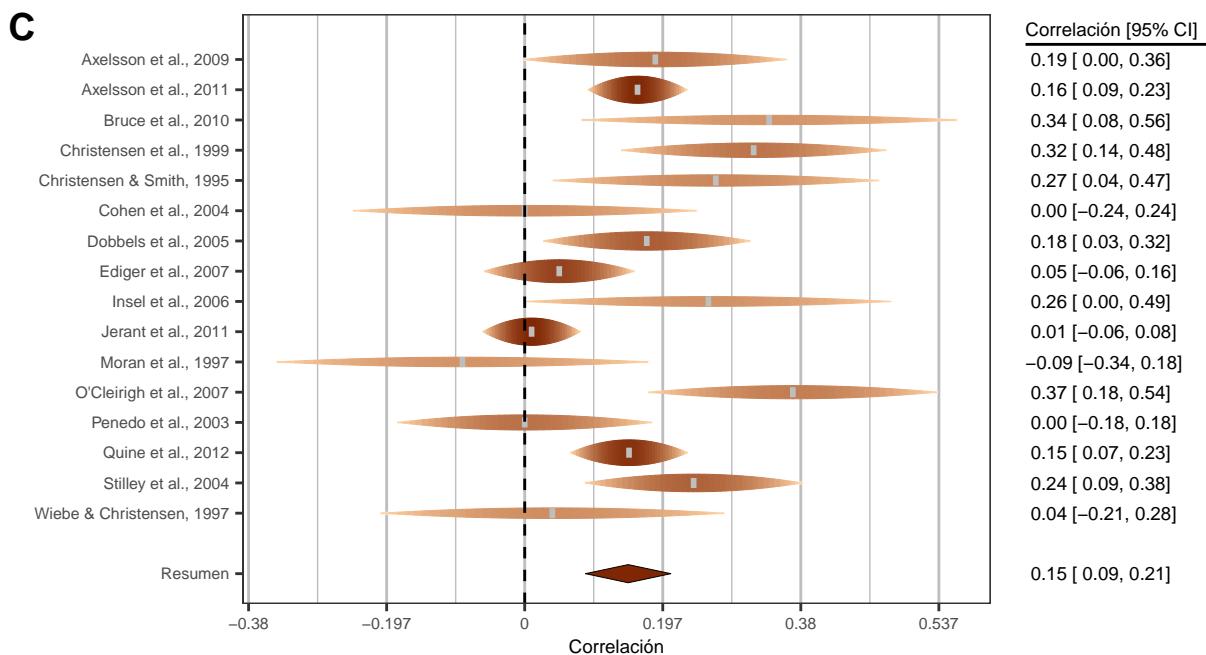


Figura 8. Variantes de *forest plots* creados con `metaviz`. **A.** Variante clásica (opción por defecto). **B.** Variante “thick” y escala de colores “Greens”. **C.** Variante “rain” y escala de colores “Oranges”.

3.5. *Funnel plot* (diagrama de embudo), regresión de Egger y sesgo de estudios pequeños

La creación e interpretación del *Funnel plot* es crucial, y aporta información muy importante acerca de un meta-análisis. Sin embargo, es en donde más errores se cometan.

3.5.1. Errores en la interpretación del *Funnel plot* y la regresión de Egger

El principal error que la mayoría de los investigadores podemos cometer⁹, es concluir que un meta-análisis tiene (o no) riesgo de sufrir de sesgos de publicación, simplemente basándonos en la evaluación de la asimetría en el *funnel plot* y la regresión de Egger (Egger et al., 1997) (sección 3.5.3).

Estos métodos, sin embargo, no son pruebas exclusivas de sesgo de publicación, sino de sesgo de estudios de tamaño muestral pequeño (ver e.g. Schwarzer et al., 2015b), que pueden incluir sesgo de publicación, pero no se centran exclusivamente en esto (explico algunas técnicas específicas para estimar el sesgo de publicación en la sección 3.6).

A pesar de esto, tanto la regresión de Egger como el *funnel plot* son interesantes, pues es importante considerar el sesgo que podrían generar estudios con tamaño de muestra pequeño en nuestra estimación del efecto *real* (o, para decirlo más técnicamente, el sesgo que podrían introducir estudios con poder estadístico bajo).

3.5.2. *Funnel plot*

Para crear un *funnel plot* de nuestro meta-análisis con `metafor` sólo tenemos que usar la función `funnel`, usando como argumento el objeto al que asignamos los resultados de nuestro meta-análisis (`res`).

```
funnel(res)
```

Con esto, generamos la Figura 9.

⁹Me incluyo pues por supuesto también soy culpable de haber cometido este error en el pasado, incluyendo algunas ideas que expuse en mi video sobre meta-análisis en *jamovi* (Leongómez, 2021).

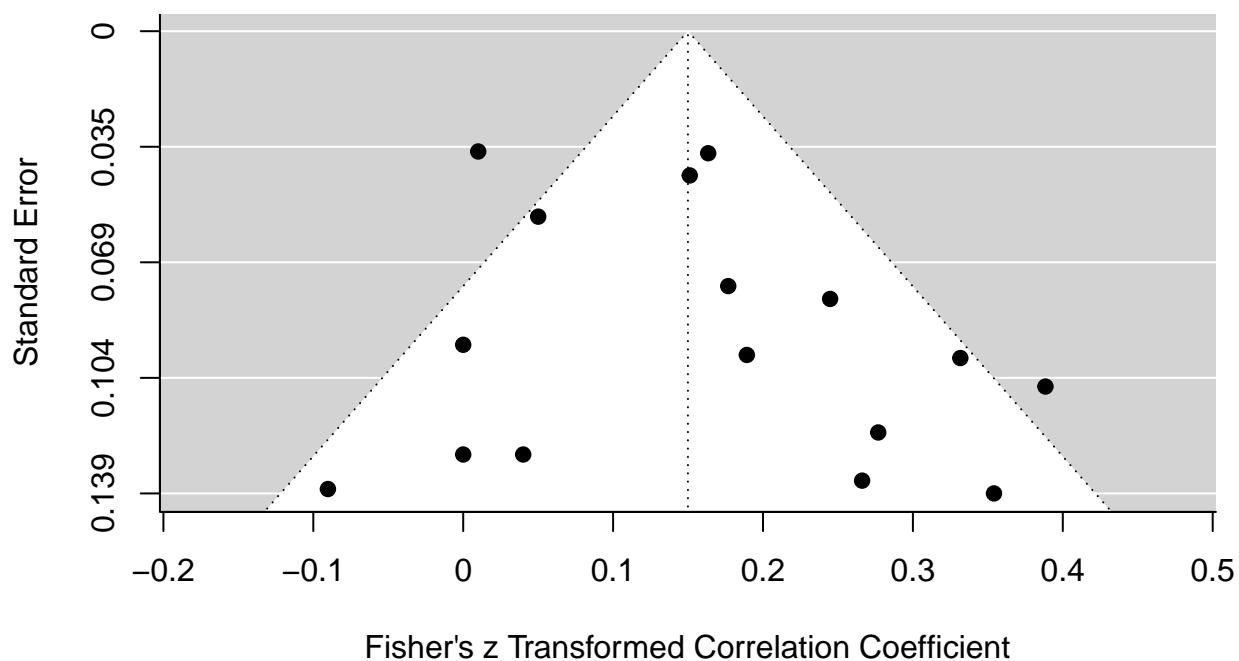


Figura 9. Funnel plot básico de `metafor`. Para cada estudio meta-analizado, tenemos el efecto (correlación, en este caso en valores z de Fisher) en el eje X , así como su error estándar en el eje Y . La línea punteada vertical representa el efecto real estimado mediante nuestro meta-análisis ($z = 0.15$), así que podemos ver los estudios que encontraron un efecto mayor (derecha) o menor (izquierda) que éste. A primera vista no parece haber mucha asimetría, pero es importante tener en cuenta que es un análisis muy subjetivo.

O, si queremos poner los títulos de los ejes en español, podemos hacerlo agregando los argumentos `xlab` (para el eje X) y/o `ylab` (para el eje Y), como se ve en la Figura 10.

```
funnel(res,
       xlab = "Coeficiente de correlación (z de Fisher)",
       ylab = "Error estándar")
```

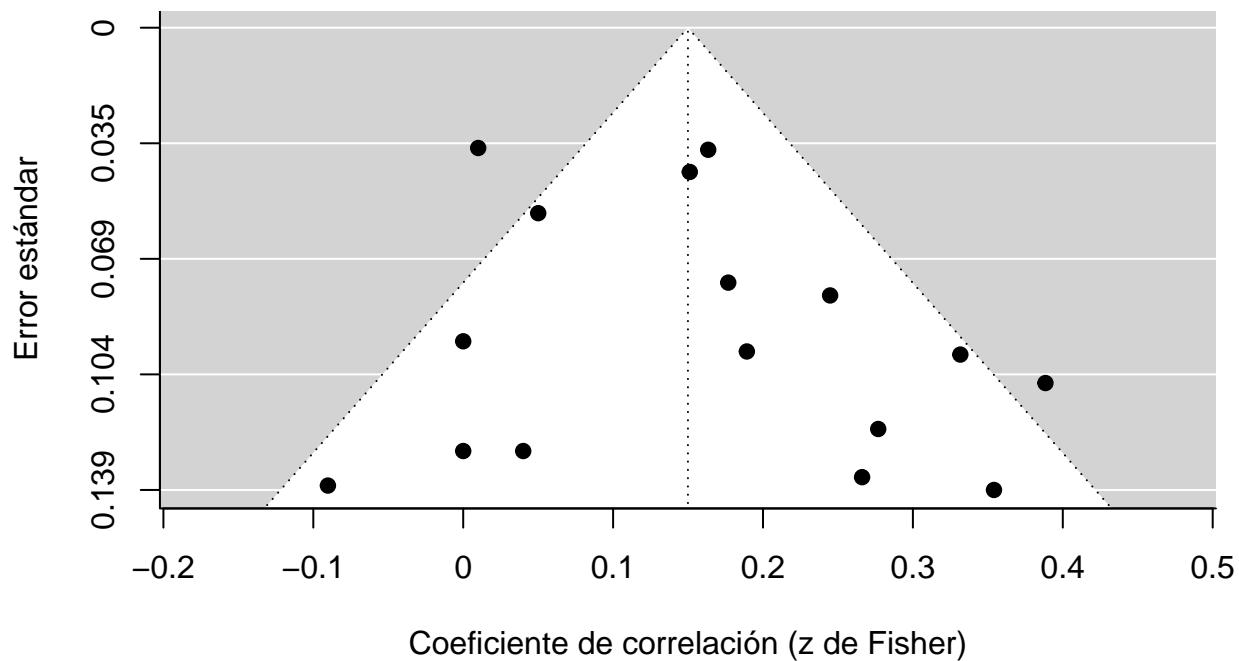


Figura 10. Funnel plot básico de `metafor`, idéntico al de la Figura 9, pero con títulos de ejes en español.

De nuevo, para una versión más sofisticada, se puede usar el paquete `metaviz`, usando la función `viz_funnel`. Hay muchas opciones, pero como ejemplo, usaré la versión por defecto, agregando sólo la línea de la regresión de Egger (`egger = TRUE`; ver sección 3.5.3, más adelante), transformando los tamaños de efecto de regreso a r de Pearson (`x_trans_function = tanh`), y con los títulos de los ejes en español (Fig. 11).

```
viz_funnel(res,
            egger = TRUE,
            x_trans_function = tanh,
            ylab = "Error estándar",
            xlab = "Coeficiente de correlación")
```

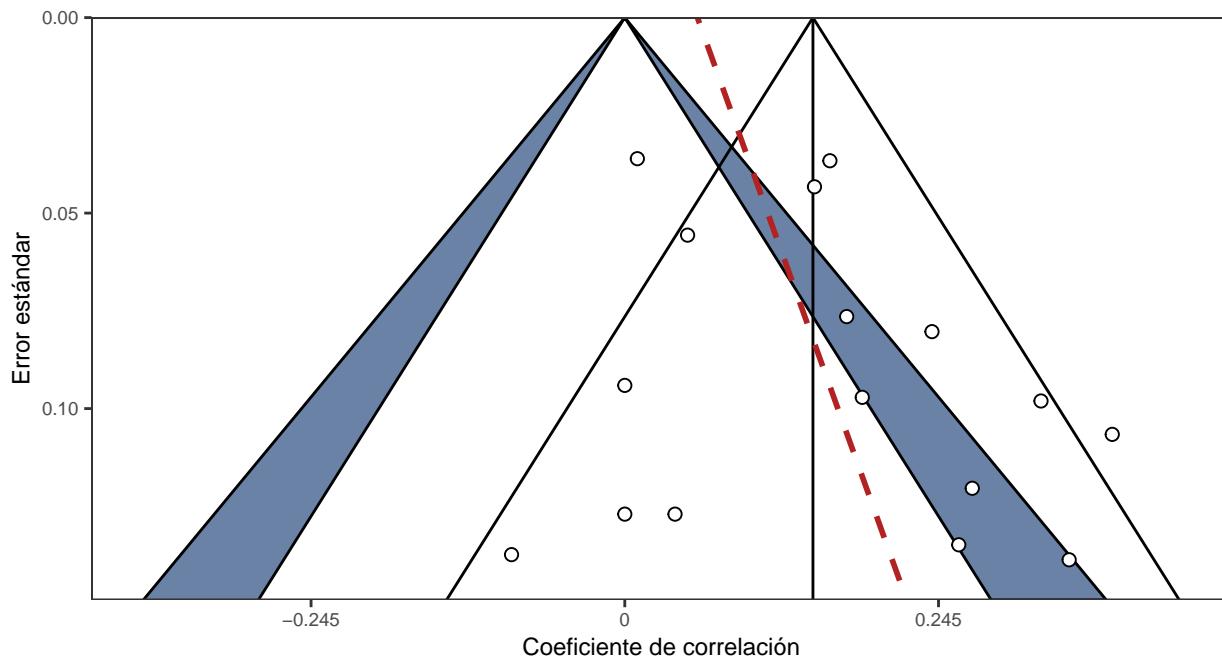


Figura 11. *Funnel plot* creado con `metaviz`. En azul, se representa el área donde estudios, según su error (y su tamaño de muestra), tendrían un efecto significativo al 5% (i.e. $p < 0.05$), y fuera de esta, donde tendrían un efecto significativo al 1% (i.e. $p < 0.01$). La línea negra vertical representa el efecto meta-analizado, y el triángulo a partir de su inicio, el área donde se ubican los estudios que no se diferencian significativamente del resultado del meta-análisis. La línea roja punteada, representa la regresión de Egger.

Alternativamente, el paquete `metaviz` tiene la función `viz_sunset`, que permite además mostrar el poder estadístico (o potencia) de los estudios meta-analizados para detectar un efecto de interés mediante una prueba de Wald de dos colas. De ser necesario, para entender las bases del poder estadístico, recomiendo ver [esta serie de videos](#) (Leongómez, 2020b) y/o, para mayor profundidad, leer [esta guía](#) (Leongómez, 2020a) que publiqué anteriormente.

A continuación, muestro dos versiones de *funnel plots* creados con la función `viz_sunset` (Fig. 12). En ambos casos, agregué el efecto *real* estimado a partir del meta-análisis (`contours = TRUE`), y transformé los tamaños de efecto de regreso a r de Pearson (`x_trans_function = tanh`).

```
# A. Escala de poder discreta
viz_sunset(res,
            contours = TRUE,
            x_trans_function = tanh)

# B. Escala de poder continua
viz_sunset(res,
            contours = TRUE,
            x_trans_function = tanh,
            power_contours = "continuous")
```

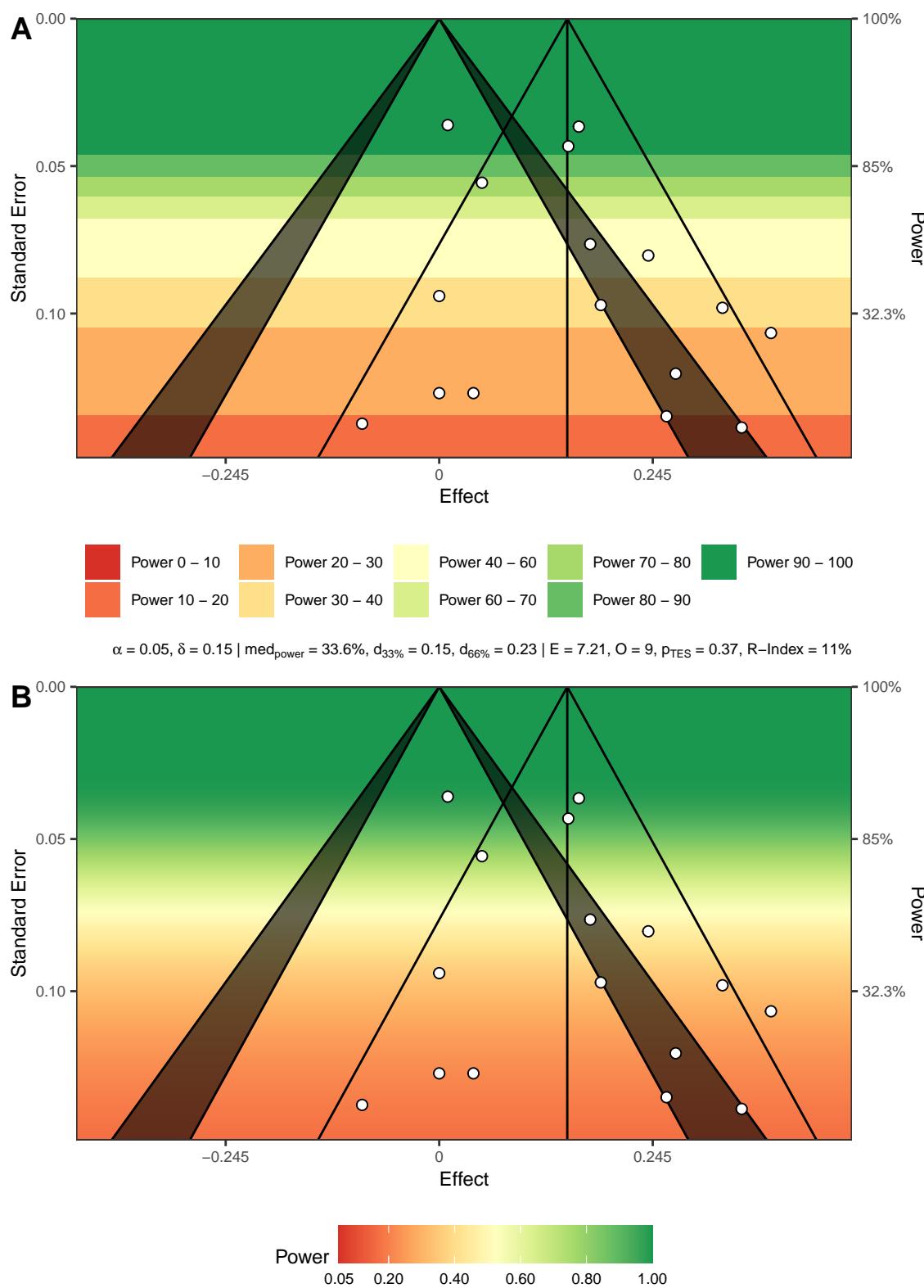


Figura 12. Dos versiones de *funnel plot* creados con [metaviz](#), usando la función `viz-sunset`, que estima el poder de cada estudio para detectar un efecto de interés. **A.** Poder representado por bandas discretas de color. **B.** Poder representado de manera continua en una escala de color. En ambos casos, y tal como en la Figura 11, el efecto real está representado como una línea vertical, y el triángulo a partir de su inicio representa el área donde se ubican los estudios que no se diferencian significativamente del resultado del meta-análisis.

3.5.3. Regresión de Egger

Para hacer una prueba formal de sesgo de estudios pequeños (Schwarzer et al., 2015b; Sterne et al., 2000, 2011), a partir de la asimetría en el *funnel plot*, podemos hacer una prueba o regresión de Egger (Egger et al., 1997).

Es importante tener en cuenta que la regresión de Egger puede carecer de poder para detectar sesgos cuando el número de estudios es pequeño; Sterne et al. (2011), de hecho, recomiendan no usar este técnica cuando tenemos menos de 10 estudios ($k < 10$). En nuestro ejemplo, dado que tenemos 16 estudios ($k = 16$), esto no debería ser un problema.

En **metafor**, podemos hacer una regresión de Egger con la función **regtest**, de nuevo usando como único argumento el objeto al que asignamos el resultado de nuestro meta-análisis (**res**).

```
regtest(res)
```

Como se puede ver, en este caso la prueba de Egger no muestra un resultado significativo (Test for Funnel Plot Asymmetry: $z = 1.0216$, $p = 0.3070$), lo que sugiere que no hay sesgo de estudios pequeños.

Consola de R: Output 7

```
Regression Test for Funnel Plot Asymmetry

Model: mixed-effects meta-regression model
Predictor: standard error

Test for Funnel Plot Asymmetry: z = 1.0216, p = 0.3070
Limit Estimate (as sei -> 0): b = 0.0790 (CI: -0.0686, 0.2266)
```

Con base en esto, y la inspección visual subjetiva del *funnel plot*, muchos investigadores podrían concluir, sin sustento adecuado, que en el meta-análisis no hay sesgo de publicación. Sin embargo, como mencioné antes (sección 3.5.1), estas pruebas no se centran en el sesgo de publicación sino en el sesgo de estudios pequeños. En otras palabras, con base en esto, lo único que podemos concluir correctamente, es que no hay sesgo de estudios pequeños (en la sección 3.6 a continuación, explicaré cómo evaluar si hay sesgo de publicación).

3.6. Sesgo de publicación (*publication bias*)

Existen diferentes opciones para estimar el sesgo de publicación de un meta-análisis. En esta sección, me centraré en dos: el método de recorte y relleno (*trim and fill*; sección 3.6.1), y el modelo de función de peso de Vevea y Hedges (sección 3.6.2), que es uno de los modelos de selección más conocidos para evaluar sesgos de publicación.

3.6.1. Método *trim and fill* (recorte y relleno)

El método de recorte y relleno (*trim and fill*) es una técnica no paramétrica para aumentar datos (Duval & Tweedie, 2000a, 2000b; ver también Duval, 2005). Este método estima los estudios potencialmente ausentes debido al sesgo de publicación en el gráfico de embudo, y ajusta la estimación del efecto *real*. Aunque muy útil, es importante tener en cuenta que este método tiene algunas limitaciones:

- No debe entenderse como una forma de obtener una estimación más “válida” del efecto *real*, sino como una forma de examinar qué tan sensibles son los resultados a una forma particular de sesgo de publicación
- No se puede usar en modelos con moderadores, como los que explico en la sección 4

En **metafor**, el método de recorte y relleno (*trim and fill*) se hace simplemente con la función **trimfill**, de nuevo usando como argumento el objeto al que asignamos el resultado de nuestro meta-análisis (**res**). En este caso, asignaré el resultado de esta función a un objeto que llamaré **tf**.

```
tf <- trimfill(res)
tf
```

Consola de R: Output 8

```
Estimated number of missing studies on the left side: 2 (SE = 2.7118)

Random-Effects Model (k = 18; tau^2 estimator: REML)
```

```

tau^2 (estimated amount of total heterogeneity): 0.0112 (SE = 0.0066)
tau (square root of estimated tau^2 value):      0.1061
I^2 (total heterogeneity / total variability):   67.50%
H^2 (total variability / sampling variability):  3.08

Test for Heterogeneity:
Q(df = 17) = 46.3990, p-val = 0.0002

Model Results:

estimate     se    zval   pval   ci.lb   ci.ub
 0.1288  0.0333  3.8628  0.0001  0.0635  0.1942 ***

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Esto nos muestra unos resultados equivalentes y muy similares a los de nuestro meta-análisis original (sección 3) y que se deben interpretar de manera igual. Sin embargo, hay unos pequeños cambios.

Primero, comienza diciendo que el número estimado de estudios que faltan en el lado izquierdo es 2 (**Estimated number of missing studies on the left side**: 2 (SE = 2.7118)), lo que quiere decir que estimó que faltaban dos estudios al lado izquierdo de nuestro *funnel plot*, y no se recortó ninguno. Por supuesto, al sumar estos dos estudios hipotéticos, el meta-análisis se hace con base en $k = 18$ estudios, en vez de 16 (**Random-Effects Model** ($k = 18$; **tau^2 estimator**: REML)). Como los dos estudios hipotéticos que se agregaron están a la izquierda de nuestro *funnel plot*, el efecto que estimamos usando el método *trim and fill* es menor ($z = 0.13$, como se ve en los resultados de nuestro meta-análisis (**Model results: Estimate** (0.1288)).

Podemos hacer un *funnel plot* del meta-análisis con el método de recorte y relleno (*trim and fill*), usando como argumento el objeto al que asignamos los resultados (tf). Con esto, generamos la Figura 13.

```

funnel(tf,
       xlab = "Coeficiente de correlación (z de Fisher)",
       ylab = "Error estándar")

```

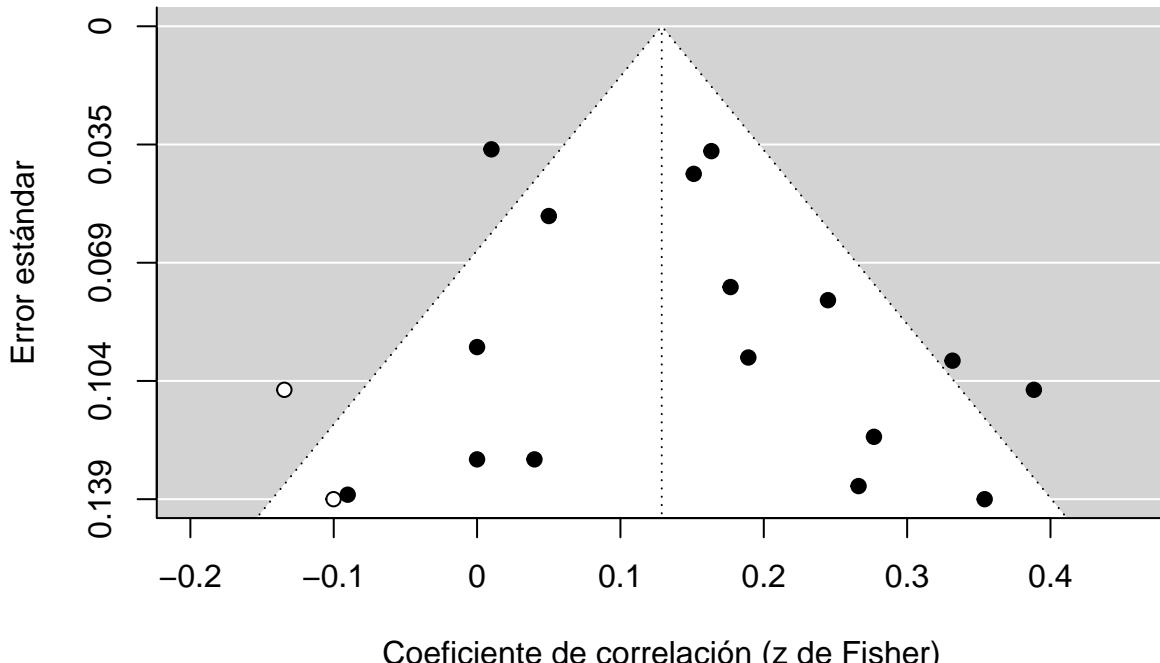


Figura 13. Funnel plot básico de `metafor` usando el método de recorte y relleno (*trim and fill*). En negro los estudios meta-analizados; en blanco, los estudios *rellenados*.

De nuevo, alternativamente podemos usar la función `viz_funnel` del paquete `metaviz`, para crear este *funnel*

plot representando los estudios *rellenados* con el método de recorte y relleno. Sin embargo, para esto usaremos como argumento el meta-análisis original (*res*), pero agregando los argumentos *trim_and_fill = TRUE* y *trim_and_fill_side = "left"* ("left" dado que sabemos que los estudios *faltantes* están a la izquierd).

```
viz_funnel(res,
            contours_col = "Oranges",
            trim_and_fill = TRUE,
            trim_and_fill_side = "left", #IMPORTANTE
            egger = TRUE,
            x_trans_function = tanh,
            ylab = "Error estándar",
            xlab = "Coeficiente de correlación") +
geom_vline(xintercept = 0, linetype = "dotted")
```

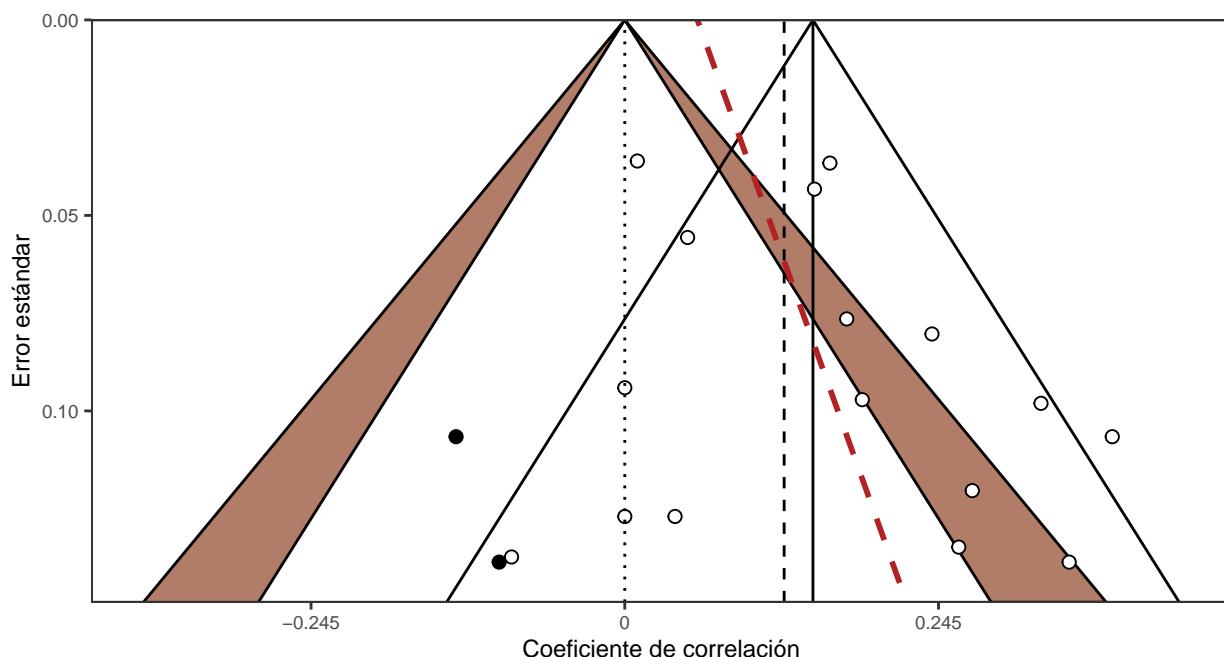


Figura 14. Funnel plot creado con `metaviz` usando el método de recorte y relleno (*trim and fill*). En blanco los estudios meta-analizados; en negro, los estudios *rellenados*.

El resultado nos muestra que el análisis aumentó dos estudios hipotéticos a la izquierda. Es decir, sugiere que hay *cierto* nivel de sesgo de publicación. De hecho, la estimación del efecto al aumentar estos estudios hipotéticos para que la distribución sea realmente simétrica, disminuye a 0.1288, algo menor que el 0.1499 que nos sugería el análisis inicial. Se debe tener en cuenta que los estudios generados por el método *trim and fill*, son hipotéticos. Y que estos estudios se utilizan para estimar el tamaño y la dirección del sesgo de publicación, pero no son datos reales. Por esto, los resultados deben ser interpretados con precaución y no constituyen evidencia definitiva de sesgo de publicación.

Si quieras profundizar en el método de recorte y relleno (*trim and fill*), su interpretación, así como sus fortalezas y limitaciones, te recomiendo leer el artículo de L. Shi y Lin (2019) al respecto.

3.6.2. Modelo de función de peso de Vevea y Hedges

El modelo de función de peso de Vevea y Hedges (1995; ver también 2005) es uno de los modelos de selección (en inglés, *selection models*) más conocidos¹⁰.

¹⁰El otro modelo de selección muy conocido, es el modelo de Copas y Shi (2000, 2001). Aunque no lo incluyo en esta guía, es importante mencionar que hay evidencia que muestra que es preferible a métodos como el de *trim and fill* (Schwarzer et al., 2010). En R, este método se puede implementar con la función `copas` del paquete `metasens` (Schwarzer et al., 2023; ver también Capítulo 5 en Schwarzer et al., 2015a).

Los modelos de selección intentan describir el mecanismo de supresión del efecto y combinarlo con un modelo de tamaño del efecto que describa la distribución de los efectos en ausencia de sesgo de publicación.

El modelo de Vevea y Hedges da más peso a ciertos tamaños de efecto. La realidad de la literatura científica es que es más probable que algunos estudios sean publicados, dependiendo de sus valores p (Coburn & Vevea, 2015; para una descripción sencilla y general, ver por ejemplo Gómez-Restrepo, 2008). En particular, es más probable que sean publicados los estudios con resultados *significativos* (típicamente, $p < 0,05$, usando un α tradicional de 0.05), en comparación con estudios con resultados no significativos (es decir, con resultados que no permiten rechazar la hipótesis nula). Estos sesgos pueden por supuesto sesgar la literatura publicada, y afectar nuestro meta-análisis.

En R, para estimar el sesgo de publicación de un meta-análisis usando el modelo de función de peso de Vevea y Hedges, se puede usar la función `weightfunct` del paquete `weighttr` (Coburn & Vevea, 2019).

Tal como se describe en la [documentación](#) de la función `weightfunct`, esta función nos permite “estimar tanto el modelo de función de peso para el sesgo de publicación que se publicó originalmente en Vevea y Hedges (1995) como la versión modificada presentada en Vevea y Woods (2005)”.

En este caso, usaré esta función, asignando el resultado a un objeto que llamaré `wf`.

```
library(weighttr)
wf <- weightfunct(effect = dat$yi, v = dat$vi, table = TRUE)
wf
```

Consola de R: Output 9

```
Unadjusted Model (k = 16):
tau^2 (estimated amount of total heterogeneity): 0.0070 (SE = 0.0051)
tau (square root of estimated tau^2 value):  0.0834

Test for Heterogeneity:
Q(df = 15) = 38.1595, p-val = 0.001436053

Model Results:

      estimate std.error z-stat     p-val   ci.lb   ci.ub
Intercept  0.1486    0.03073  4.835 1.3288e-06 0.08837 0.2088

Adjusted Model (k = 16):
tau^2 (estimated amount of total heterogeneity): 0.0056 (SE = 0.0045)
tau (square root of estimated tau^2 value):  0.0750

Test for Heterogeneity:
Q(df = 15) = 38.1595, p-val = 0.001436053

Model Results:

      estimate std.error z-stat     p-val   ci.lb   ci.ub
Intercept  0.09153   0.04464  2.050 0.040341  0.00403 0.1790
0.025 < p < 1  0.24121   0.20122  1.199 0.230626 -0.15317 0.6356

Likelihood Ratio Test:
X^2(df = 1) = 2.98493, p-val = 0.084043

Number of Effect Sizes per Interval:

      Frequency
p-values <0.025          9
0.025 < p-values < 1       7
```

Los resultados tienen una estructura similar a los del meta-análisis original (descrito y explicado en la sección 3) y a los del meta-análisis usando el método *trim and fill* (sección 3.6.1). Sin embargo, se organizan en dos partes; primero presenta los resultados del modelo no ajustado (bajo el encabezado `Unadjusted Model (k = 16)`), y después presenta los resultados del modelo con pesos ajustados de acuerdo al modelo de función de peso de Vevea y Hedges (bajo el encabezado `Adjusted Model (k = 16)`).

El modelo no ajustado nos da un estimado muy similar (pero no idéntico) al del estudio original (0.1486), dado que usa un método ligeramente diferente.

Como antes, también nos da valores de heterogeneidad τ^2 , τ y Q .

Pero lo más importante, es que en la segunda parte (el modelo ajustado) los resultados del meta-análisis se calculan de acuerdo con los pesos dados a cada efecto.

Esta técnica, a diferencia del método *trim and fill* no “recorta” ni “rellena” estudios, por lo que el análisis se hace con el mismo número de estudios (en este caso, 16; por eso el encabezado dice **Adjusted Model** ($k = 16$)).

Sin embargo, la función **weightfunct** incrementa el peso de estudios que tienen menos probabilidad de ser publicados (típicamente resultados no significativos), y reduce el peso de estudios con mayor probabilidad de ser publicados (los que encuentran resultados más extremos). Por esto, al usar esta técnica, estás asumiendo que, de hecho, en el efecto que tratas de encontrar en tu meta-análisis hay un sesgo de publicación, lo que a menudo es una suposición bastante apropiada.

Al usar esta técnica, tenemos un resultado bastante distinto. Mientras que el meta-análisis original nos daba como resultado un efecto de ~ 0.15 , esta técnica nos estima un efecto de ~ 0.09 . Ha *encogido* nuestro tamaño de efecto, asumiendo que nuestro resultado original estaba sesgado.

Al final, está el *Likelihood ratio test* (en español, “Prueba de la razón de verosimilitud”), que evalúa la bondad del ajuste de dos modelos estadísticos que compiten entre sí, con base en la relación de su verosimilitud¹¹. En este caso, comparando el modelo original, con este modelo con pesos ajustados.

Este resultado nos da una tendencia no descartable ($p\text{-val} = 0.084043$; menor que 0.10 y por tanto significativa si asumimos un análisis de una cola), que nos da evidencia de que en efecto hay un sesgo de publicación, a pesar de que el *funnel plot* (Figs. 9, 10, 11 y 12) y la regresión de Egger (sección 3.5.3), sugerían lo contrario.

3.7. Poder estadístico del meta-análisis

En esta sección explicaré cómo hacer un análisis de poder de un meta-análisis. Como mencioné anteriormente, de ser necesario, para entender las bases del poder estadístico, recomiendo ver [esta serie de videos](#) (Leongómez, 2020b) y/o, para mayor profundidad, leer [esta guía](#) (Leongómez, 2020a) que publiqué anteriormente.

La idea de esto es saber si nuestro meta-análisis tiene un poder suficiente para detectar el efecto *real* estimado mediante nuestro análisis (en nuestro caso $z = 0.15$ para el meta-análisis original “res”, o $z = 0.09$ el meta-análisis con pesos ajustados “wf”).

Para este ejemplo, asumiré que el efecto *real* es el estimado en nuestro análisis original ($z = 0.15$), pues este efecto es mayor. Si no tuviésemos el poder suficiente para detectar confiablemente ese efecto, menos lo tendríamos para un efecto menor, como el detectado en nuestro meta-análisis con pesos ajustados.

Para hacer esto, usaré el paquete **metameta** (Quintana, 2022, 2023), que permite calcular y visualizar el poder estadístico de un meta-análisis para detectar un rango de posibles efectos *reales* estimados.

3.7.1. Instalación de **metameta**

El paquete **metameta** se debe instalar desde GitHub ya que, al día de hoy, no está aún disponible en CRAN¹².

Para esto, debemos tener instalado el paquete **devtools**, y usar la función **install_github** que nos permite instalar paquetes directamente desde GitHub.

¹¹Aunque no voy a entrar en detalles respecto a técnicas de inferencia estadística basadas en verosimilitud, si quieres conocer más acerca de estos métodos, así como de estadística Bayesiana (que tiene mucho que ver con verosimilitud) y frequentista (la que solemos usar, que hace inferencia a partir de valores p), con énfasis en las fortalezas y limitaciones de cada enfoque, recomiendo encarecidamente el libro “*Improving Your Statistical Inferences*” de Daniël Lakens (2022). Este libro, completamente gratuito y abierto, integra además los contenidos de dos cursos gratuitos del mismo autor (“[Improving Your Statistical Inferences](#)” y “[Improving Your Statistical Questions](#)”), que también recomiendo hacer a cualquier persona que quiera dedicarse a hacer investigación y use técnicas estadísticas.

¹²**CRAN** (siglas de *Comprehensive R Archive Network*) es una red de servidores de todo el mundo que almacenan versiones idénticas y actualizadas del código y la documentación de R (incluyendo el lenguaje mismo de R, paquetes aceptados por el equipo de CRAN, y toda la documentación). Es el repositorio oficial y dedicado a R, desde el que normalmente instalas cualquier paquete. **GitHub**, en cambio, es un repositorio general y abierto para proyectos (típicamente programación) en el que, entre otras cosas, suelen estar alojados todos los paquetes de R en versiones de desarrollador, y antes de ser aceptados en CRAN. Por supuesto, a diferencia de CRAN, GitHub es un repositorio de código general, no específico para R.

```
#se debe tener instalado el paquete devtools
library(devtools)
install_github("dsquintana/metameta")
```

3.7.2. Análisis de poder

Una vez hemos instalado `metameta`, podemos cargar el paquete.

```
library(metameta)
```

Como datos, necesitamos no solamente los tamaños de efecto a meta-analizar (r de Pearson transformado a z de Fisher), sino además los intervalos de confianza, tal como fueron reportados en varios de nuestros *Forest plots*.

Como mencioné, voy a asumir un efecto *real* de $r = 0,15$, tal como en nuestro meta-análisis original. Sin embargo, el efecto *real* no es algo que podamos saber con absoluta certeza (es, de hecho, lo que queremos acercarnos a conocer a través del meta-análisis), así que la función `mapower_ul` del paquete `metameta` calcula el poder de cada meta-análisis para un rango de posibles efectos reales.

Para esto, la función `mapower_ul` requiere una base de datos que contenga información concreta, y con columnas con nombres específicos: el efecto encontrado en cada estudio (en nuestro caso `yi`, que es la columna en la que guardamos los coeficientes de correlación transformados a z de Fisher), así como sus intervalos de confianza. Estas columnas deben llamarse `yi`, `lower` y `upper`, respectivamente.

Esto lo puedo obtener de manera sencilla con la función `summary`, que agrega a la base de datos (`dat`) los límites inferior (`ci.lb`) y superior (`ci.ub`) de los intervalos de confianza. Sin embargo, aunque la columna con el tamaño de efecto (`yi`) ya tiene el nombre requerido, debo renombrar las columnas que tienen los límites de los intervalos de confianza (`ci.lb` y `ci.ub`) como `lower` y `upper`, usando la función `rename`.

```
dat.poder <- summary(dat) |>
  rename(lower = ci.lb, upper = ci.ub)
```

Asigné esta versión modificada de la base de datos a un objeto que llamé `dat.poder`. Ya con esta base de datos lista y con el formato correcto para usar la función `mapower_ul`, podemos seguir adelante.

La función `mapower_ul` requiere que asignemos la base de datos (`dat.poder`) al argumento `dat`, el tamaño de efecto que estimamos como efecto *real* (en este caso, 0.15) al argumento `observed_es`, así como un nombre que le daremos al meta-análisis al argumento `name` (en este caso lo llamaré “Molloy et al. 2014”, pero por supuesto tú puedes darle el nombre que consideres adecuado, siempre entre comillas). Asignaré el resultado de este análisis a un objeto que simplemente llamaré `poder`.

```
poder <- mapower_ul(dat = dat.poder, observed_es = 0.15, name = "Molloy et al. 2014")
```

Como veremos a continuación, este objeto tiene varios elementos muy importantes.

3.7.2.1. Poder mediano del meta-análisis para diferentes tamaños de efecto. El objeto generado por la función `mapower_ul` tiene varios elementos. Uno particularmente relevante es la lista de poder mediano estimado para diferentes tamaños de efecto (`$power_median_dat`), que asignaré como lista a un objeto que llamaré `lista_poder` (guardar este objeto es sumamente útil para hacer visualizaciones del análisis de poder, como explico más adelante, en la sección 3.7.3).

```
lista_poder <- list(poder$power_median_dat)
```

Para ver el objeto, como siempre, solo tengo que llamarlo por el nombre que le haya dado (en este caso, `lista_poder`). Muestra el poder estadístico del meta-análisis para detectar diferentes tamaños de efecto (que llama `es`, por las siglas en inglés de *effect size*).

```
lista_poder
```

Consola de R: Output 10

```
[1]
power_es_observed      es01      es02      es03      es04      es05      es06
1          0.1762492 0.5356629 0.8673657 0.9837594 0.9992182 0.9999858
       es07 es08 es09 es1 es_observed meta_analysis_name
1 0.9999999   1   1   1  0.3364197 Molloy et al. 2014
```

Por ejemplo, nos muestra que para detectar un efecto de $z = 0.1$ (**es01**), se estima un poder de 0.1762492; para un poder de $z = 0.2$ (**es02**), se estima un poder de 0.5356629, y continúa así hasta un poder de $z = 1.0$ (**es1**).

Finalmente, nos muestra el poder estimado para detectar un efecto del tamaño del que estimamos como *real* (en nuestro caso, $z = 0.15$), bajo el encabezado **es_observed**. Como se ve, el poder es de tan solo 0.3364197, lo que quiere decir que, si el efecto *real* de la asociación entre escrupulosidad y la adherencia a la medicación fuese $z = 0.15$, un meta-análisis como el nuestro sólo detectaría esa correlación como significativa en el 33.64 % de los casos.

Lo que es más, podemos concluir que solo tenemos suficiente poder estadístico para detectar como significativos efectos desde aproximadamente $z = 0.3$ (**es03** = 0.8673657), donde por fin tenemos un poder mayor a 0.8, o lo que es lo mismo, 80 %. Sin embargo, nuestras estimaciones del efecto *real* están por debajo de este umbral, ya que en nuestro meta-análisis original fue de $z = 0.15$ y en el meta-análisis con pesos ajustados fue de $z = 0.09$.

3.7.2.2. Poder de cada estudio para diferentes tamaños de efecto. El otro elemento generado es una base de datos modificada que, de manera similar a la lista de poder mediano, contiene el poder estimado para diferentes tamaños de efecto, pero en este caso no como una mediana, sino para cada estudio (**\$dat**). Asignaré este elemento a un objeto que llamaré **poder_full_dat**.

```
poder_full_dat <- poder$dat
poder_full_dat
```

Consola de R: Output 11

	study_id	authors	year	ni	ri	controls	design	a_measure	c_measure	meanage	quality	yi	vi	sei
1	1	Axelsson et al.	2009	109	0.187	none	cross-sectional							
2	2	Axelsson et al.	2011	749	0.162	none	cross-sectional							
3	3	Bruce et al.	2010	55	0.340	none	prospective							
4	4	Christensen et al.	1999	107	0.320	none	cross-sectional							
5	5	Christensen & Smith	1995	72	0.270	none	prospective							
6	6	Cohen et al.	2004	65	0.000	none	prospective							
7	7	Dobbels et al.	2005	174	0.175	none	cross-sectional							
8	8	Ediger et al.	2007	326	0.050	multiple	prospective							
9	9	Insel et al.	2006	58	0.260	none	prospective							
10	10	Jerant et al.	2011	771	0.010	multiple	prospective							
11	11	Moran et al.	1997	56	-0.090	multiple	prospective							
12	12	O' Cleirigh et al.	2007	91	0.370	none	prospective							
13	13	Penedo et al.	2003	116	0.000	none	cross-sectional							
14	14	Quine et al.	2012	537	0.150	none	prospective							
15	15	Stilley et al.	2004	158	0.240	none	prospective							
16	16	Wiebe & Christensen	1997	65	0.040	none	prospective							
								yi	vi	sei				
1	self-report	other	22.00		1	0.18922664	0.009433962	0.09712680						
2	self-report	NEO	53.59		1	0.16343992	0.001340483	0.03661193						
3	other	NEO	43.36		2	0.35409253	0.019230769	0.13867250						
4	self-report	other	41.70		1	0.33164711	0.009615385	0.09805627						
5	other	NEO	46.39		2	0.27686382	0.014492754	0.12038364						
6	other	NEO	41.20		2	0.00000000	0.016129032	0.12699779						
7	self-report	NEO	52.30		1	0.17682002	0.005847953	0.07647051						
8	self-report	NEO	41.00		3	0.05004173	0.003095975	0.05564047						
9	other	other	77.00		2	0.26610841	0.018181818	0.13483749						
10	other	NEO	78.60		3	0.01000033	0.001302083	0.03608373						
11	other	NEO	57.20		2	-0.09024419	0.018867925	0.13735804						
12	self-report	NEO	37.90		2	0.38842310	0.011363636	0.10659840						
13	self-report	NEO	39.20		1	0.00000000	0.008849558	0.09407036						
14	self-report	other	69.00		2	0.15114044	0.001872659	0.04327344						
15	other	NEO	46.20		3	0.24477411	0.006451613	0.08032046						
16	other	NEO	56.00		1	0.04002135	0.016129032	0.12699779						
	zi	pval	lower		upper	power_es_observed	power_es01							

1	1.9482075	5.139014e-02	-0.001141888	0.37959517	0.3390835	0.1774836
2	4.4640346	8.043057e-06	0.091680539	0.23519930	0.9837035	0.7797623
3	2.5533975	1.066777e-02	0.082294427	0.62589063	0.1910727	0.1113715
4	3.3821502	7.192082e-04	0.139456828	0.52383739	0.3337559	0.1750148
5	2.2998036	2.145935e-02	0.040911886	0.51281576	0.2383028	0.1320187
6	0.0000000	1.000000e+00	-0.248915675	0.24891567	0.2188783	0.1234921
7	2.3122218	2.076546e-02	0.026937828	0.32670221	0.5006730	0.2576556
8	0.8993600	3.684609e-01	-0.059013584	0.15909704	0.7691106	0.4354589
9	1.9735128	4.843717e-02	0.001826917	0.53038990	0.1994161	0.1149996
10	0.2771374	7.816746e-01	-0.060723775	0.08072444	0.9859910	0.7914240
11	-0.6569876	5.111889e-01	-0.359465946	0.17897757	0.1938536	0.1125800
12	3.6437317	2.687136e-04	0.179490237	0.59735596	0.2905754	0.1553001
13	0.0000000	1.000000e+00	-0.184377902	0.18437790	0.3575910	0.1861296
14	3.4926197	4.783072e-04	0.066324499	0.23595637	0.9340134	0.6371865
15	3.0474131	2.308202e-03	0.087346017	0.40220221	0.4632371	0.2379950
16	0.3151285	7.526641e-01	-0.208894321	0.28893703	0.2188783	0.1234921
	power_es02	power_es03	power_es04	power_es05	power_es06	power_es07
1	0.5395394	0.8705053	0.9845502	0.9992836	0.9999877	0.9999999
2	0.9997697	1.0000000	1.0000000	1.0000000	1.0000000	1.0000000
3	0.3026621	0.5806100	0.8223954	0.9500825	0.9910281	0.9989921
4	0.5317864	0.8642261	0.9829686	0.9991528	0.9999840	0.9999999
5	0.3827659	0.7026651	0.9135189	0.9858616	0.9987531	0.9999421
6	0.3502733	0.6562694	0.8829174	0.9759855	0.9971497	0.9998088
7	0.7439048	0.9751837	0.9994638	0.9999977	1.0000000	1.0000000
8	0.9489275	0.9997002	0.9999999	1.0000000	1.0000000	1.0000000
9	0.3170763	0.6044851	0.8429297	0.9597856	0.9936099	0.9993842
10	0.9998300	1.0000000	1.0000000	1.0000000	1.0000000	1.0000000
11	0.3074782	0.5886809	0.8294859	0.9535367	0.9919839	0.9991443
12	0.4666848	0.8035418	0.9634686	0.9968384	0.9998781	0.9999980
13	0.5659845	0.8904902	0.9890522	0.9996035	0.9999950	1.0000000
14	0.9961139	0.9999997	1.0000000	1.0000000	1.0000000	1.0000000
15	0.7019697	0.9620572	0.9987365	0.9999900	1.0000000	1.0000000
16	0.3502733	0.6562694	0.8829174	0.9759855	0.9971497	0.9998088
	power_es09	power_es1				
1	1.0000000	1.0000000				
2	1.0000000	1.0000000				
3	0.9999971	0.9999999				
4	1.0000000	1.0000000				
5	1.0000000	1.0000000				
6	0.9999999	1.0000000				
7	1.0000000	1.0000000				
8	1.0000000	1.0000000				
9	0.9999988	1.0000000				
10	1.0000000	1.0000000				
11	0.9999978	0.9999999				
12	1.0000000	1.0000000				
13	1.0000000	1.0000000				
14	1.0000000	1.0000000				
15	1.0000000	1.0000000				
16	0.9999999	1.0000000				

Básicamente, esta tabla contiene información que nos permite evaluar si cada estudio incluido en nuestro meta-análisis tenía o no un poder estadístico suficiente para detectar un rango de tamaños de efecto que, como antes van desde $z = 0.1$ (columna power_es01), hasta $z = 1.0$ (power_es1).

También, nos permite ver el poder estadístico de cada estudio para detectar un tamaño de efecto *real* como el que estimamos ($z = 0.15$), en la columna power_es_observed (la información de esta columna es análoga a la de los funnel plot creados con la función viz_funnel del paquete metaviz; Fig. 12A y B).

3.7.3. Visualización del análisis de poder (*Firepower plot*)

Estos análisis muestran que, en promedio, nuestro meta-análisis sólo alcanza un poder estadístico suficiente (mayor a 0.8 u 80 %) para detectar de manera confiable efectos mayores a ~0.3, lo que está muy por encima de nuestras estimaciones del efecto real (0.15 en nuestro meta-análisis original, 0.09 en nuestro meta-análisis con pesos ajustados). De hecho, el poder estimado para detectar un efecto de 0.15 es de apenas 33.64 % (reportado como es_observed, con un valor de 0.3364197 en el output 10 de la sección 3.7.2.1).

Puedo fácilmente crear una visualización de esto con la función `firepower` del paquete `metameta`. Solo tengo que agregar, como único argumento, el objeto en el que guardé la lista de poder mediano estimado para diferentes tamaños de efecto (que en este caso asigné a un objeto que llamé `lista_poder` en la sección 3.7.2.1).

```
poder.plot <- firepower(lista_poder)
```

En este caso asigné el `fireplot` resultante a un objeto, para poder modificar después, que llamé `poder.plot`. Entonces, para ver el resultado, debo correr el objeto (Fig. 15).

```
poder.plot
```

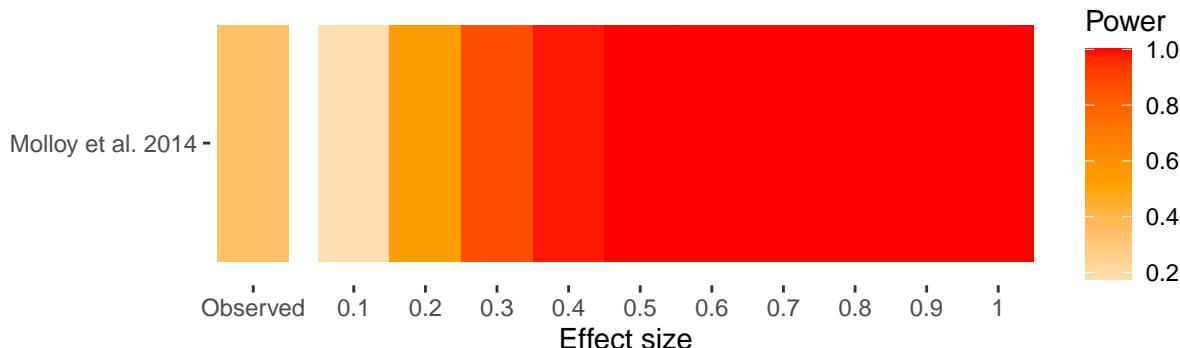


Figura 15. Fireplot básico de `metameta`, para un análisis de poder de nuestro meta-análisis. *Observed* hace referencia al tamaño de efecto observado en nuestro meta-análisis original; en este caso, 0.15.

El objeto `poder.plot` contiene dos elementos: los datos (`$dat`) y la gráfica propiamente dicha (`$fp_plot`). Este último elemento es de clase `ggplot`, por lo que podemos usar funciones de `ggplot2` para cambiar, por ejemplo, el título del eje *X* a “Tamaño de efecto”, el título de la leyenda a “Poder”, y el efecto observado de “*Observed*” a “Observado”¹³. Por ejemplo:

```
poder.plot$fp_plot +
  xlab("Tamaño de efecto") +
  guides(fill = guide_legend(title = "Poder",
                             reverse = TRUE)) +
  scale_x_discrete(labels = c("es_observed" = "Observado",
                            "es01" = 0.1, "es02" = 0.2, "es03" = 0.3, "es04" = 0.4,
                            "es05" = 0.5, "es06" = 0.6, "es07" = 0.7, "es08" = 0.8,
                            "es09" = 0.9, "es1" = 1))
```

Este código produce la Figura 16, con todos los textos en español.

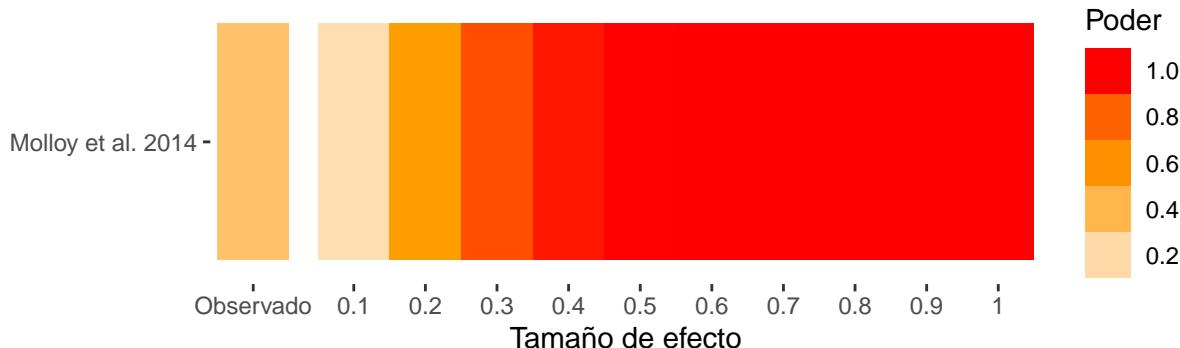


Figura 16. Fireplot básico de `metameta`, para un análisis de poder de nuestro meta-análisis, con el texto traducido a español y con la leyenda en una escala discreta para facilitar su lectura. *Observado* hace referencia al tamaño de efecto observado en nuestro meta-análisis original (en este caso, 0.15).

¹³Para cambiar el título del eje *X* usé la función `xlab`; para el título de la leyenda la función `guides` (opción `fill = guide_legend`); y para los valores del eje *X*, la función `scale_x_discrete`.

La Figura 16 nos permite comprobar, de nuevo, que el meta-análisis solamente alcanza un poder mínimo recomendado de 0.8 (80 %) cuando se trata de efectos $z = 0.3$ (o mayores), y que para nuestro efecto *observado* ($z = 0.15$), el poder apenas alcanza cerca de 0.4 (40 %).

Si el poder estadístico de nuestro meta-análisis para detectar ese efecto ($z = 0.15$) es demasiado bajo, por supuesto sería aún más deficiente para detectar un efecto menor, como el estimado en nuestro modelo con pesos corregidos ($z = 0.09$).

3.8. Más información sobre influencia y meta-análisis combinatorio

Como hemos visto, al realizar un meta-análisis, es fundamental tomar decisiones que pueden influir significativamente en los resultados y conclusiones. Una de las preguntas más importantes es qué estudios incluir en el análisis y si es necesario excluir algún estudio. Aunque nuestro diagnóstico inicial de influencia (sección 3.3) no mostró la presencia de estudios atípicos en nuestro meta-análisis, en ocasiones se requiere un análisis más exhaustivo.

En esta sección, que se basa por completo en la explicación y ejemplos de Quintana (2020), explicaré dos técnicas: *leave-one-out* (sección 3.8.1), que consiste en realizar una serie de meta-análisis excluyendo uno de los estudios incluidos en cada iteración, y meta-análisis combinatorio (sección 3.8.2), que permite crear todas las combinaciones posibles de estudios y, por lo tanto, evaluar la robustez de los resultados en diferentes escenarios.

3.8.1. Leave-one-out

La técnica de *leave-one-out* realiza una serie de meta-análisis dejando fuera cada uno de los estudios incluidos. En nuestro caso, dado que tenemos 16 estudios, esta técnica ajustará 16 modelos (uno excluyendo cada estudio).

Esta técnica no sólo se puede hacer como parte de un diagnóstico de influencia (como se muestra en la sección 3.3), sino que también se puede hacer para ver la estimación del efecto meta-analizado al excluir cada uno de los estudios de nuestra base de datos.

Hacerlo sólo requiere usar la función `leave1out` del paquete `metafor`, usando como argumento el resultado de nuestro meta-análisis original (en este caso, `res`):

```
leave1out(res)
```

Consola de R: Output 12

	estimate	se	zval	pval	ci.lb	ci.ub	Q	Qp	tau2	I2
1	0.1483	0.0336	4.4157	0.0000	0.0825	0.2142	37.7109	0.0006	0.0091	64.9684
2	0.1500	0.0354	4.2355	0.0000	0.0806	0.2194	36.7672	0.0008	0.0100	62.2717
3	0.1418	0.0315	4.5064	0.0000	0.0801	0.2035	35.3930	0.0013	0.0075	61.1741
4	0.1379	0.0308	4.4739	0.0000	0.0775	0.1984	33.5886	0.0024	0.0068	58.1050
5	0.1441	0.0323	4.4550	0.0000	0.0807	0.2075	36.5396	0.0009	0.0082	62.8995
6	0.1566	0.0326	4.8119	0.0000	0.0928	0.2205	37.1703	0.0007	0.0084	63.5769
7	0.1489	0.0342	4.3555	0.0000	0.0819	0.2160	37.6797	0.0006	0.0094	65.1901
8	0.1599	0.0334	4.7841	0.0000	0.0944	0.2254	36.1484	0.0010	0.0084	61.4011
9	0.1455	0.0324	4.4863	0.0000	0.0819	0.2091	37.0495	0.0007	0.0083	63.4803
10	0.1647	0.0291	5.6601	0.0000	0.1077	0.2218	25.0661	0.0339	0.0047	43.6587
11	0.1586	0.0317	5.0069	0.0000	0.0965	0.2207	35.6617	0.0012	0.0077	61.6732
12	0.1353	0.0296	4.5647	0.0000	0.0772	0.1934	31.9021	0.0041	0.0059	55.0102
13	0.1592	0.0324	4.9100	0.0000	0.0956	0.2227	36.3291	0.0009	0.0080	62.0683
14	0.1514	0.0353	4.2858	0.0000	0.0821	0.2206	37.7339	0.0006	0.0100	63.6942
15	0.1432	0.0330	4.3432	0.0000	0.0786	0.2078	35.8385	0.0011	0.0083	62.6114
16	0.1552	0.0329	4.7203	0.0000	0.0908	0.2196	37.7017	0.0006	0.0087	64.3368
	H2									
1	2.8546									
2	2.6505									
3	2.5756									
4	2.3869									
5	2.6954									
6	2.7455									
7	2.8727									
8	2.5907									
9	2.7382									
10	1.7749									
11	2.6091									
12	2.2227									

13	2.6363
14	2.7544
15	2.6746
16	2.8040

Como puedes ver, esta función crea una tabla con las columnas `estimate`, `se`, `zval`, `pval`, `ci.lb`, `ci.ub`, `Q`, `Qp`, `tau2`, `I2` y `H2` (explicadas en la sección 3.1). Sin embargo, tenemos 16 filas, cada una de las cuales corresponde a los resultados cuando se excluye ese estudio del meta-análisis.

En este caso, los tamaños del efecto estimado (columna `estimate`) para los meta-análisis que eliminaron un único estudio están entre $z = 0.14$ y $z = 0.16$, lo que muestra que la eliminación de un estudio no influye mucho en el tamaño del efecto original estimado, que fue de $z = 0.15$. En otras palabras, podríamos eliminar cualquier estudio de nuestro análisis y el resultado cambiaría muy poco.

Aunque este análisis es bastante informativo, podría haber un subgrupo de estudios —en lugar de un único estudio— que influye en el resultado. Para investigar esto, podemos hacer un meta-análisis combinatorio.

3.8.2. Meta-análisis combinatorio

Aunque la técnica de *leave-one-out* es útil para determinar la influencia de cada estudio en un meta-análisis, no nos dice nada de cuánto cambia nuestra estimación del efecto real cuando excluimos dos o más estudios. Por lo tanto, el meta-análisis combinatorio puede ser una técnica complementaria para detectar subgrupos de estudios que estén afectando nuestros resultados.

Por ejemplo, si estamos realizando un meta-análisis sobre el efecto del consumo de café en la salud cardiovascular, es posible que encontremos varios estudios que no controlan adecuadamente el consumo de tabaco de los participantes. Si eliminamos uno solo de estos estudios, es posible que aún quede un sesgo importante en nuestra estimación del efecto *real*. En este caso, el meta-análisis combinatorio nos permitiría evaluar todas las posibles combinaciones de estudios que incluyen o excluyen los estudios que no controlan el consumo de tabaco, lo que nos ayudaría a comprender mejor el impacto de estos estudios en nuestra estimación del efecto *real*.

Un meta-análisis combinatorio ajusta una serie de meta-análisis a partir de todas las combinaciones posibles de estudios. Esta técnica es poderosa, pero es importante tener en cuenta que el número de posibles combinaciones aumenta exponencialmente a medida que aumenta el número de estudios incluidos en nuestro meta-análisis. De hecho, el número de análisis a realizar puede calcularse usando la fórmula $2^k - 1$, donde k es el número de estudios meta-analizados (Tabla 3).

Tabla 3

Número de posibles combinaciones según número de estudios (k)

Número de estudios (k)	Posibles combinaciones ($2^k - 1$)	Número de estudios (k)	Posibles combinaciones ($2^k - 1$)
5	31	18	262 143
6	63	19	524 287
7	127	20	1 048 575
8	255	21	2 097 151
9	511	22	4 194 303
10	1 023	23	8 388 607
11	2 047	24	16 777 215
12	4 095	25	33 554 431
13	8 191	26	67 108 863
14	16 383	27	134 217 727
15	32 767	28	268 435 455
16	65 535	29	536 870 911
17	131 071	30	1 073 741 823

Como se puede ver en la Tabla 3, el número de combinaciones aumenta drásticamente a medida que aumenta el número de estudios en nuestro meta-análisis. Por ejemplo, si tuviéramos 5 estudios, habría solo 31 posibles combinaciones ($2^5 - 1 = 31$). Sin embargo, si tuviéramos 20 estudios, habría más de un millón de combinaciones ($2^{16} - 1 = 1\,048\,575$) y si tuviéramos 30 estudios, habría más de mil millones de combinaciones (ver Tabla 3). Dado que esto puede consumir muchos recursos computacionales, es importante tener precaución al realizar un meta-análisis combinatorio con un gran número de estudios. El análisis puede tardar mucho tiempo e incluso fallar si se usa toda la memoria RAM del ordenador.

En nuestro ejemplo, estamos meta-analizando 16 estudios, lo que significa que necesitamos ajustar 65 535 modelos ($2^{16} - 1$). Ajustar todos estos modelos puede llevar bastante tiempo, aunque esto dependerá de las especificaciones de tu equipo (en mi portátil, tarda alrededor de 5 minutos, pero ten en cuenta que en equipos viejos podría tardar 20, 30 minutos o incluso más).

Para visualizar la heterogeneidad entre los estudios, podemos utilizar una técnica llamada *visualización gráfica de la heterogeneidad de los estudios* (en inglés, *Graphical Display of Study Heterogeneity* o simplemente *GOSH*), propuesta por Olkin et al. (2012). En R, podemos realizar esta visualización utilizando la función `gosh` del paquete `metafor`. Simplemente necesitamos proporcionar el resultado de nuestro meta-análisis original (`res`) como argumento. Opcionalmente, también podemos agregar el argumento `progbar = TRUE` para ver el progreso del análisis en la consola. Guardaré el resultado de este análisis en un objeto llamado `gp` para su posterior uso.

```
gp <- gosh(res, progbar = TRUE)
```

Una vez que hayamos realizado el análisis *GOSH* podemos utilizar la función `plot` para visualizar los 65 535 modelos. Simplemente necesitamos proporcionar el resultado de `gosh` (`gp` en este caso) como argumento para `plot`, pero en este ejemplo utilizaré los argumentos `breaks` y `labels` para personalizar la visualización. El argumento `breaks` nos permite definir el número de puntos de corte (y, por lo tanto, el número de barras) en los histogramas, mientras que el argumento `labels` nos permite cambiar los títulos de los ejes (en este caso para ponerlos en español). Esto nos permitirá crear una visualización más clara y fácil de entender de la heterogeneidad entre los estudios.

```
plot(gp, breaks = 100,
      labels = c("Coeficiente de correlación (z de Fisher)",
                expression(I^2)))
```

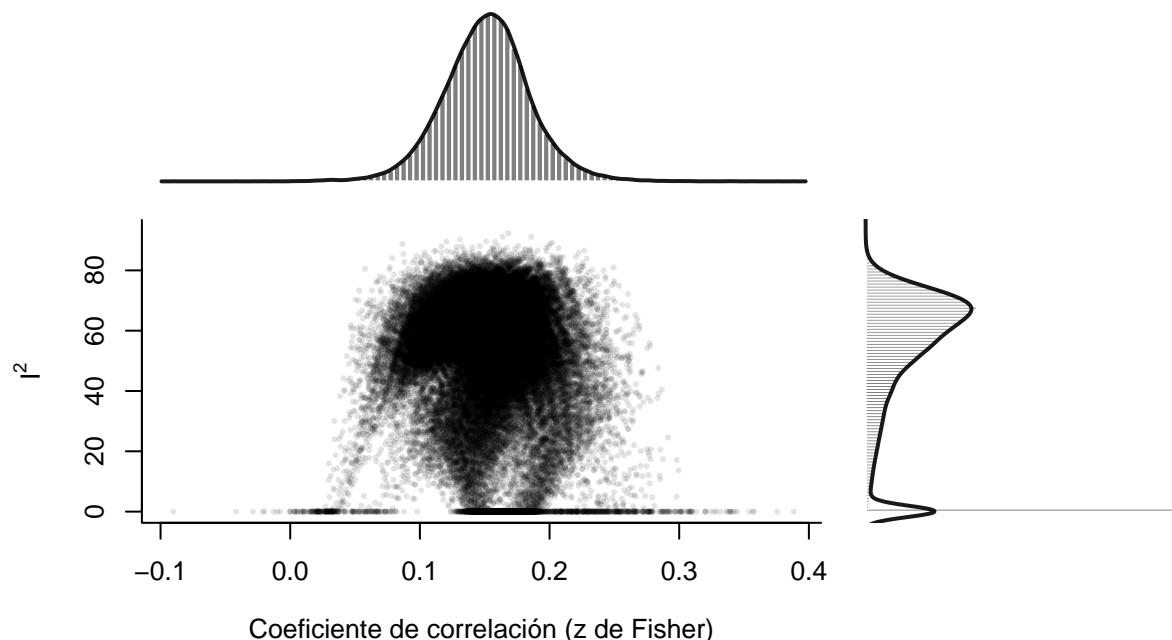


Figura 17. *COSH plot* creado con la función `gosh` del paquete `metafor`. Un *COSH plot* (en inglés, *Graphical Display of Study Heterogeneity*) es una visualización gráfica de la heterogeneidad de todos los posibles modelos de meta-análisis ajustados a partir de todas las combinaciones posibles de estudios. Muestra los efectos estimados para cada uno de los modelos en el eje horizontal, y su respectiva heterogeneidad en el eje vertical. La distribución de los efectos (parte superior de la gráfica) y la heterogeneidad (lado derecho de la gráfica) están representadas como densidades e histogramas.

Los tamaños del efecto que se obtienen de los estudios incluidos en un meta-análisis de correlaciones varían ampliamente, generalmente entre $z = 0.05$ y $z = 0.25$. Esta variación subraya cómo la elección de los estudios incluidos en el meta-análisis puede influir en la estimación del tamaño del efecto. Sin embargo, no hay un subgrupo de resultados que se distinga significativamente del resto.

Para ilustrar mejor esto, voy a manipular manualmente el tamaño de efecto de uno de los estudios. Por ejemplo, en nuestro caso, podemos cambiar el tamaño del efecto del estudio 12, que inicialmente tenía un valor z de Fisher de 0.39, a un valor mucho mayor, de 0.70. Aunque esta base de datos alterada pierde validez para realizar cualquier análisis, nos permite visualizar el efecto de valores atípicos en una base de datos¹⁴.

Para evitar afectar nuestra base de datos original, creamos una copia de los datos en un objeto llamado `dat_mo`. Luego, utilizamos las funciones `mutate` y `replace` del paquete `dplyr` (Wickham et al., 2023) para cambiar los tamaños de efecto del estudio 12 en la nueva base de datos.

```
dat_mo <- dat |>
  mutate(yi = replace(yi, list = 12, values = 0.7))
```

Para visualizar cómo el estudio 12 se convierte en un valor atípico después de la manipulación de los datos, podemos realizar un meta-análisis de la nueva base de datos alterada (`dat_mo`) y representar los resultados en un diagrama de bosque o *forest plot*. Como se puede observar en la Figura 18, el valor atípico introducido (estudio 12) es claramente visible en el *forest plot*.

```
res_mo <- rma.uni(yi, vi, data = dat_mo)
forest(res_mo)
```

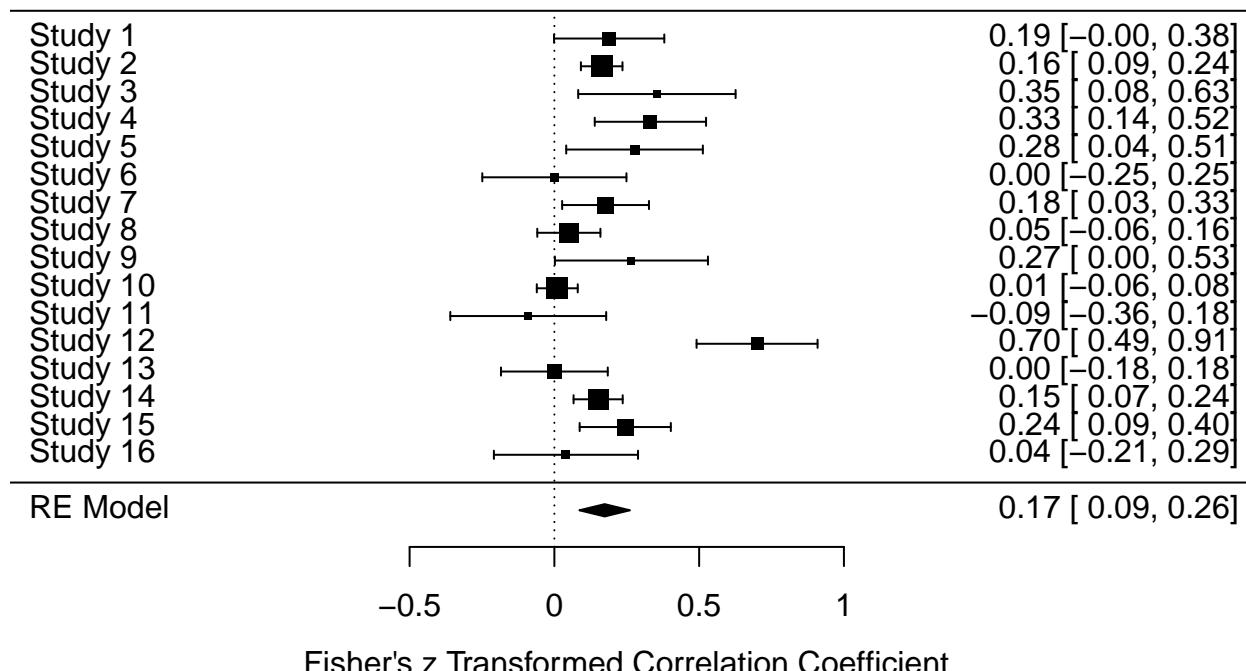


Figura 18. Forest plot básico de `metafor` de la base de datos con valores atípicos introducidos manualmente para el estudio 12. Como se puede ver, este estudio tiene un tamaño de efecto muy diferente al de los demás estudios.

Hagamos el meta-análisis combinatorio con esta base de datos alterada que contiene el resultado atípico, usando de nuevo la función `plot` (Fig. 19) para ilustrar el resultado.

```
gp_mo <- gosh(res_mo)
plot(gp_mo, breaks = 100,
     labels = c("Coeficiente de correlación (z de Fisher)",
               expression(1^2)))
```

¹⁴Normalmente identificamos formalmente estudios con resultados atípicos a partir de un diagnóstico de influencia (3.3) y técnicas tipo *leave-one-out* (sección 3.8.1), apoyándonos informalmente en el *forest plot* (sección 3.4) y el *funnel plot* (sección 3.5).

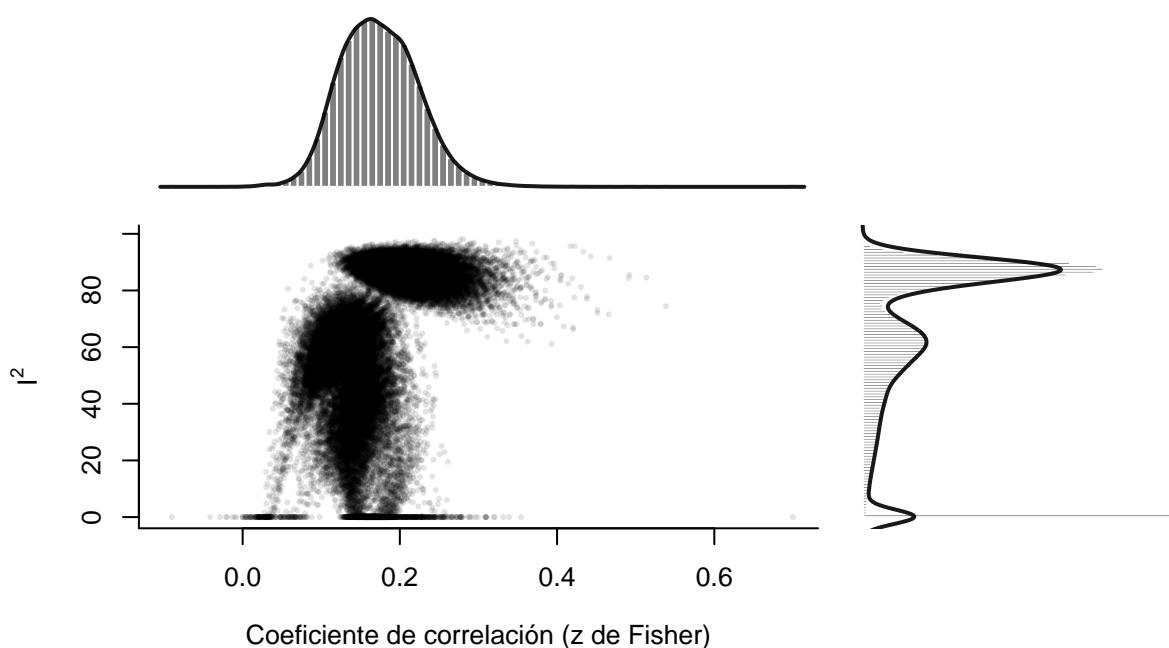


Figura 19. GOSH plot de un meta-análisis combinatorio que contiene un estudio con resultado atípico. En la parte superior del gráfico, se observa claramente un grupo de tamaños de efecto elevados con una alta heterogeneidad.

En la parte superior de la Figura 19 hay un claro grupo de tamaños de efecto relativamente grandes, con alta heterogeneidad. Podemos destacar específicamente los modelos del meta-análisis que incluyeron el estudio atípico con el argumento `out`:

```
plot(gp_mo, breaks = 100, out = 12,
     labels = c("Coeficiente de correlación (z de Fisher)",
               expression(I^2)))
```

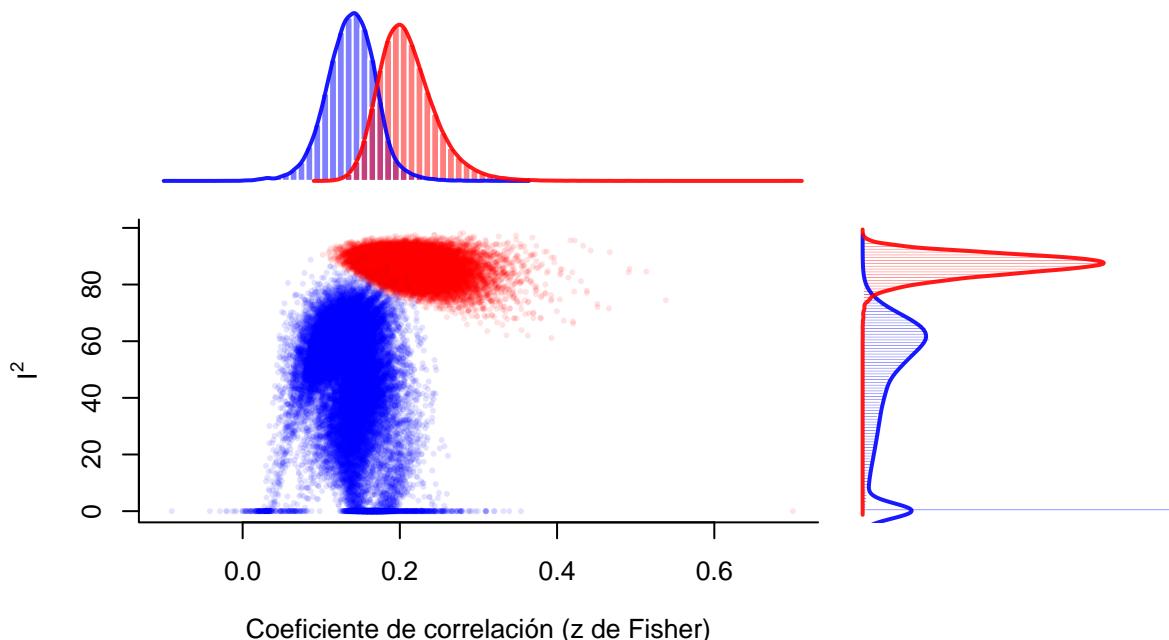


Figura 20. GOSH plot de un meta-análisis combinatorio que contiene un estudio con resultado atípico. En este caso, a diferencia de la Figura 19, los resultados combinatorios que no contienen el estudio atípico están en azul, mientras que los meta-análisis que sí lo incluyen están en rojo. Los meta análisis que incluyen el estudio con un resultado atípico tienden a arrojar estimados más fuertes para la correlación, sesgando el resultado hacia valores más extremos, y son siempre altamente heterogéneos.

La inclusión del estudio 12 en los modelos de meta-análisis provoca un cambio notable en la distribución de las estimaciones del tamaño del efecto *real*. Este cambio indica la importancia de considerar cuidadosamente la inclusión de cada estudio en un meta-análisis, ya que la inclusión de valores atípicos —o estudios que no cumplen los criterios de calidad— puede tener un impacto significativo en los resultados.

En resumen, el meta-análisis combinatorio, junto con la visualización de datos mediante gráficos *GOSH*, permite demostrar cómo las opciones de análisis pueden influir en los resultados y conclusiones de un meta-análisis. Es fundamental realizar una selección cuidadosa de los estudios incluidos y llevar a cabo un análisis crítico y riguroso de los resultados obtenidos para garantizar la fiabilidad y validez del meta-análisis.

3.8.2.1. Identificación de estudios que contribuyen a los patrones de heterogeneidad a partir del *GOSH plot* Una limitación del meta-análisis combinatorio (y el *GOSH plot*), es que no necesariamente nos permite identificar qué estudios están influyendo de manera preocupante en nuestro resultado. En el ejemplo pasado, sabíamos que el estudio 12 era el *culpable* pues así lo manipulamos. Sin embargo, al hacer un meta-análisis no necesariamente sabemos cuáles estudios son sospechosos. Por esto, tener herramientas que nos permitan identificar estudios potencialmente influyentes, es importante.

Por ejemplo, la función `gosh.diagnostics` del paquete `dmetar` (Harrer et al., 2019) se puede usar para crear análisis y visualizaciones muy completas a partir de los objetos generados con la función `gosh` del paquete `metafor`, como los que creamos en la sección 3.8.2, inmediatamente anterior ¹⁵.

La función `gosh.diagnostics` utiliza tres algoritmos de aprendizaje automático (*machine learning*) para detectar patrones en nuestros datos: el algoritmo *k-means* (Hartigan & Wong, 1979), la agrupación de densidad alcanzable y la conectividad, o *DBSCAN* (Schubert et al., 2017), y los modelos de mezcla gaussiana, o *GMM* (Fraley & Raftery, 2002).

Básicamente, usa esos tres métodos para buscar grupos de resultados (o *clusters*) entre todas las combinaciones posibles de estudios que creamos con la función `gosh`. Con esto, y como explicaré en esta sección, ayuda a identificar estudios influyentes en nuestro resultado a partir de una identificación de posibles valores atípicos.

Es importante tener en cuenta que, al menos por ahora, y tal como sucede con `metameta`, el paquete `dmetar` debe instalarse desde GitHub:

```
#se debe tener instalado el paquete devtools
devtools::install_github("MathiasHarrer/dmetar")
```

Una vez hemos instalado `dmetar`, podemos cargar el paquete.

```
library(dmetar)
```

Como ejemplo, voy a usar el objeto `gp_mo` que creamos antes, a partir de la base de datos alterada que contiene el resultado atípico (estudio 12). Voy a usar la función `gosh.diagnostics` usando ese objeto como argumento.

Sin embargo, es posible agregar parámetros a cada algoritmo como argumentos. Si quieres información detallada y más profunda, puedes ver la [documentación](#) de la función `gosh.diagnostics`, y los ejemplos en el capítulo 5 del libro *Doing Meta-Analysis in R: A Hands-on Guide* (Harrer et al., 2021).

En este caso, voy a agregar el argumento `km.params = list(centers = 2)` para especificar que el algoritmo *k-means* debe buscar dos *clusters* (`centers`) en nuestros datos (esto no es estrictamente necesario, pero de manera automática el algoritmo buscará tres; decidí poner dos, pues visiblemente hay dos grupos de resultados, como se ve en la Fig. 19 y especialmente en la Fig. 20).

Adicionalmente, incluiré el argumento `db.params = list(eps = 0.1, MinPts = 50)`, para especificar parámetros para el algoritmo *DBSCAN*, pues los *clusters* requieren un número mínimo de puntos (`MinPts`: *minimum number of points* que se refiere mínimo número de puntos por *cluster*) dentro de una distancia máxima (`eps`: ϵ o épsilon) alrededor de uno de sus miembros (o “semilla”) ¹⁶. En este caso, agregué `eps = 0.1` para especificar que la “distancia

¹⁵El libro *Doing meta-analysis with R: a hands-on guide* (Harrer et al., 2021) se acompaña del paquete `dmetar` (Harrer et al., 2019), y explica opciones para hacer meta-análisis tanto a partir de `metafor`, como a partir de otros paquetes (más información en el Apéndice A).

¹⁶Elegir los parámetros `eps` y `MinPts` no es necesariamente una tarea sencilla. No hay una única forma de elegir estos parámetros, ni una manera automática y sencilla de hacerlo. Las técnicas existentes son relativamente complejas, por lo que no las incluyo acá. Si quieres información detallada sobre estos parámetros, te recomiendo leer Ester et al. (1996) y Schubert et al. (2017), así como buscar diferentes maneras prácticas de elegir los valores para estos parámetros propuestas en *blogs* como [este](#) en inglés, o [este](#) en español.

de *alcanzabilidad*” (en inglés, *reachability distance*) es de 0.1, y `MinPts = 50` para determinar el número mínimo de puntos en cada *cluster* (en este caso, 50).

En este caso, asignaré el resultado a un objeto que llamaré `gp_diag`.

```
gp_mo_diag <- gosh.diagnostics(gp_mo,
                                km.params = list(centers = 2),
                                db.params = list(eps = 0.1, MinPts = 50))
```

De manera conveniente, en la consola se puede ver el progreso de la función.

Consola de R: Output 13

```
Perform Clustering...
|=====| DONE
```

Para ver un resumen, solo tengo que usar la función `summary` agregando como argumento el objeto generado con la función `gosh.diagnostics` (`gp_mo_diag`):

```
summary(gp_mo_diag)
```

Consola de R: Output 14

```
GOSH Diagnostics
=====
- Number of K-means clusters detected: 2
- Number of DBSCAN clusters detected: 3
- Number of GMM clusters detected: 9

Identification of potential outliers
-----
- K-means: Study 12
- DBSCAN: Study 10, Study 12
- Gaussian Mixture Model: Study 10, Study 12
```

En el *output*, vemos el número de *clusters* que ha detectado cada algoritmo. En este caso, el modelo *K-means* detectó 2 *clusters*, *DBSCAN* detectó 3, y *GMM* detectó 9. Dado que cada enfoque utiliza una estrategia matemática diferente para segmentar los datos, es normal que el número de *clusters* no sea el mismo.

De manera importante, el *output* también nos reporta explícitamente los estudios identificados como potencialmente atípicos, bajo el encabezado “*Identification of potential outliers*”. En este caso, mientras que el modelo *K-means* sugiere que el estudio 12 es atípico (e influyente), los modelos *DBSCAN* y *GMM* detectaron tanto el estudio 12, como el estudio 10 como potencialmente atípicos. En otras palabras, los tres modelos coinciden en señalar el estudio 12 como atípico.

Para ver esto gráficamente, puedo sencillamente usar la función `plot` incluyendo como argumento el objeto que creamos con la función `gosh.diagnostics` (en este caso, `gp_mo_diag`).

```
plot(gp_mo_diag)
```

Esto creará una serie de figuras; primero, una figura por cada algoritmo (*K-means*, *DBSCAN* y *GMM*), y después una figura por cada estudio potencialmente atípico que se haya identificado (en nuestro caso, los estudios 12 y 10; si no se identificara ninguno, estas gráficas no se harían). Si estás usando RStudio, deberás moverte entre las figuras usando las flechas que están sobre la pestaña *Plots* del panel donde aparecen, pues sólo será visible la última figura.

En este caso, voy a producir cada figura independientemente para analizarla.

Primero, tenemos la Figura 21 que muestra los resultados del algoritmo *K-means*. Incluye 3 paneles, llamados *K-means Algorithm*, que es un *GOSH plot* pero resaltando en colores las combinaciones de estudios asignadas a cada *cluster* (izquierda); *Cluster imbalance (K-means algorithm)*, que representa el porcentaje de cambio entre los *clusters* producto de cada estudio (derecha, arriba); y *Cluster imbalance (Cook's Distance)*, que nos muestra la distancia de Cook producto de cada estudio para cada *cluster* (derecha, abajo).

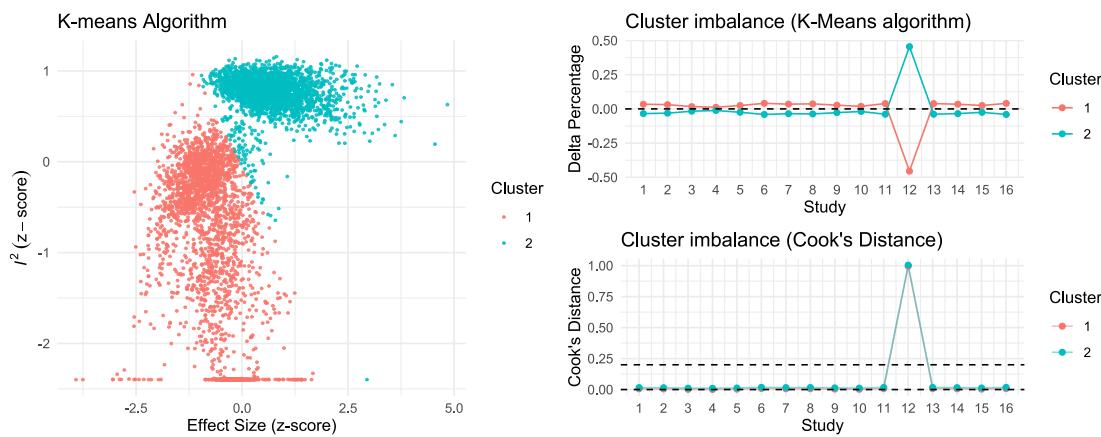


Figura 21. *GOSH plot*, desequilibrio de *clusters* y distancia de Cook estimados usando el algoritmo *K-means*. Los *clusters* identificados se representan con diferentes colores, y sus efectos se muestran en los 3 paneles. Este algoritmo identificó 2 *clusters*, pues así fue especificado.

¿Qué nos muestra la Figura 21? Primero, en el *GOSH plot* podemos ver que los dos *clusters* coinciden casi perfectamente con los que encontramos en la Figura 19, cuando especificamos explícitamente que se debían agrupar las combinaciones que incluían el estudio 12. Esto es importante pues acá no especificamos qué estudios son sospechosos, y el algoritmo mismo está identificando un *cluster* que sabemos, contiene un estudio con un resultado atípico e influyente.

Pero, de manera aún más clara, vemos que hay un gran desbalance entre *clusters* al incluir el estudio 12. Si medimos el desbalance entre *clusters* como porcentaje de cambio, podemos ver que los dos *clusters* se alejan muchísimo en el estudio 12; si medimos el desbalance como distancia de Cook, podemos ver que hay un pico muy importante para ambos *clusters* en el estudio 12, lo que quiere decir que ese estudio influye muchísimo en la distancia entre esos dos *clusters*.

Luego, tenemos la Figura 22 que muestra los resultados del algoritmo *DBSCAN*. Incluye también 3 paneles, equivalentes a los de la Figura 21, pero acá llamados *DBSCAN Algorithm (black dots are outliers)*, *Cluster imbalance (Density-Based Clustering)*, y al igual que en la Figura 21, *Cluster imbalance (Cook's Distance)*.

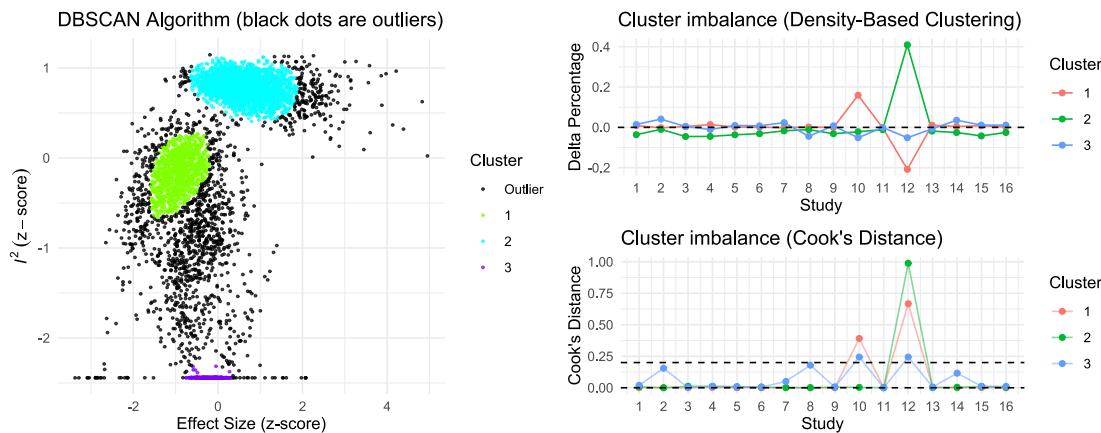


Figura 22. *GOSH plot*, desequilibrio de *clusters* y distancia de Cook estimados usando el algoritmo *DBSCAN*. Los *clusters* identificados se representan con diferentes colores, y sus efectos se muestran en los 3 paneles. En el *GOSH plot*, las combinaciones de estudios que no pudieron ser asignadas a ningún *cluster* están en negro. Este algoritmo identificó 3 *clusters*.

En el *GOSH plot* de la Figura 22 muestra los tres clusters detectados por el algoritmo *DBSCAN* (verde, azul claro y morado), así como las combinaciones de estudios que —dados los parámetros `eps` y `MinPts` que usamos en la función `gosh.diagnostics`— no fueron clasificadas en ningún *cluster* (*Outlier*, en negro). El *cluster* 2 (azul claro), corresponde en buena medida al *cluster* 2 de la Figura 21, que a su vez coincide ampliamente con la Figura 19.

Adicionalmente, el desbalance entre *clusters* muestra picos para el estudio 12, tanto cuando es medido como porcentaje de cambio, como cuando es medido como distancia de Cook. Sin embargo, acá también encontramos un pico de menor tamaño para el Estudio 10, que explica por qué fue señalado como potencialmente atípico.

Finalmente, tenemos la Figura 23 que muestra los resultados del algoritmo *GMM*. Incluye de nuevo tres paneles equivalentes, llamados *Gaussian Mixture Model*, *Cluster imbalance (GMM)*, y al igual que en la Figuras 21 y 22, *Cluster imbalance (Cook's Distance)*.

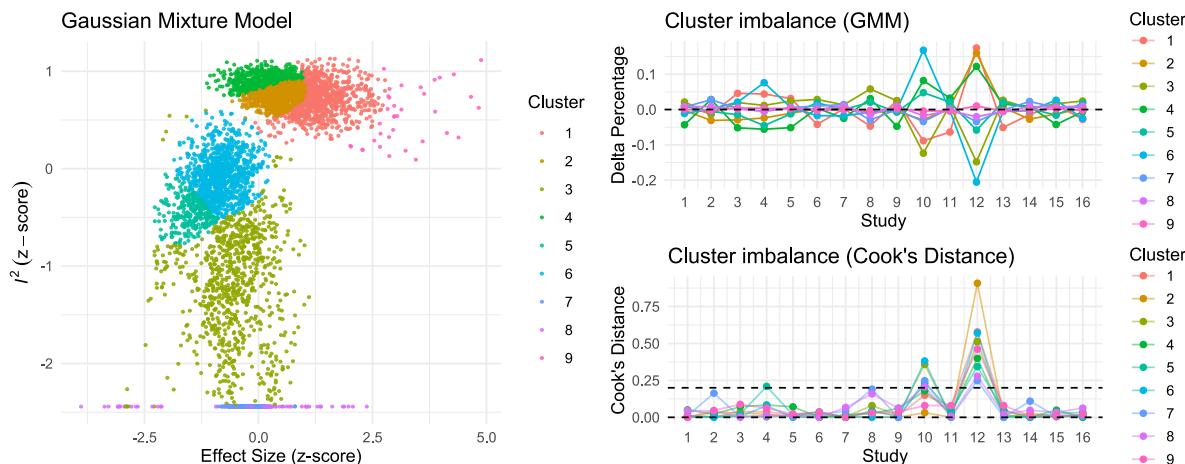


Figura 23. *GOSH plot*, desequilibrio de *clusters* y distancia de Cook estimados usando el algoritmo de modelos de mezcla gaussiana (en inglés, *Gaussian Mixture Model* o *GMM* por sus siglas). Los *clusters* identificados se representan con diferentes colores, y sus efectos se muestran en los 3 paneles. Este algoritmo identificó 9 *clusters*.

El *GOSH plot* de la Figura 23 muestra los nueve clusters detectados por el algoritmo *GMM*. Aunque en este caso no hay ningún *cluster* que corresponda tan claramente con el *cluster* 2 de las Figuras 21 o 22, o con la Figura 19, sí vemos de nuevo picos muy importantes para el estudio 12, especialmente cuando el desbalance entre *clusters* es medido como distancia de Cook. En últimas, los tres algoritmos coinciden en señalar que el estudio 12 es potencialmente atípico y podría tener demasiada influencia en nuestra estimación del efecto *real*.

Como mencioné antes, la función `plot` también crea una figura por cada estudio potencialmente atípico que se haya identificado (en nuestro caso, los estudios 12 y 10; Fig 24).

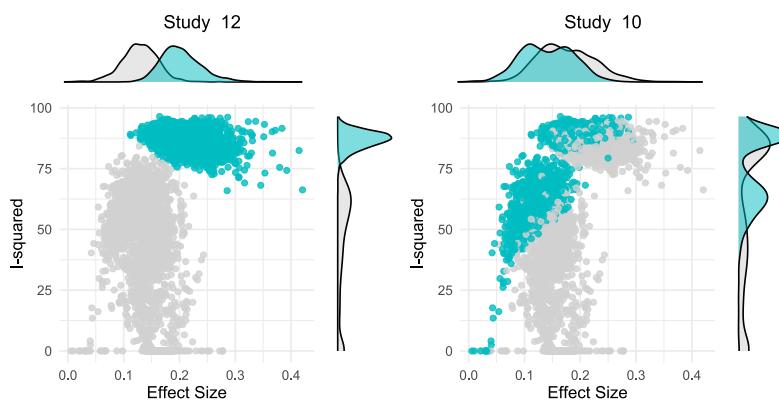


Figura 24. *GOSH plot* de cada estudio identificado como potencialmente atípico por los algoritmos de la función `gosh.diagnostics`. En este caso, los estudios 12 (izquierda) y 10 (derecha). En azul claro se resaltan los resultados de las combinaciones de estudios que incluyen cada uno de esos estudios potencialmente atípicos.

Como verás, la Figura 24 es simplemente una serie *GOSH plots*, señalando en azul claro las combinaciones de estudios que incluyen cada uno de los estudios identificados como potencialmente atípicos: el estudio 12 (izquierda) y el estudio 10 (derecha). En este caso, por supuesto, la figura que resalta las combinaciones que incluyen el estudio 12 corresponde completamente a lo mostrado en la Figura 19.

En conjunto, estas técnicas incluidas en la función `gosh.diagnostics` nos permiten identificar estudios potencialmente atípicos, incluso cuando no tenemos indicación o sospecha de algún o algunos estudios. En este caso, la función indicó dos estudios como potencialmente atípicos (estudios 12 y 10), y los tres algoritmos, a pesar de ser distintos, coincidieron en señalar especialmente al estudio 12 como influyente (algo que en este caso sabemos que es verdad, pues manipulamos su resultado).

4. Meta-análisis de correlación con moderador (meta-regresión)

La inclusión de moderadores en un meta-análisis mediante una técnica llamada meta-regresión, combina principios de regresión meta-analítica y lineal. Esta técnica nos permite determinar si existe una asociación lineal entre el resultado de nuestro meta-análisis y una o más covariables; es decir, variables que pueden estar influyendo en los resultados de los estudios que estamos analizando. Al incluir estos moderadores en nuestro análisis, podemos controlar mejor la heterogeneidad y entender cómo factores específicos pueden estar afectando los resultados del meta-análisis.

Por ejemplo, en el caso de la relación entre la altura de una planta y la cantidad de luz que recibe, el tipo de suelo puede actuar como variable moderadora. Es decir, el efecto de la cantidad de luz en la altura de la planta puede ser diferente dependiendo del tipo de suelo, y estudios hechos en diferentes tipos de suelo podrían tender a encontrar resultados distintos. Por lo tanto, en este caso, el tipo de suelo actuaría como una variable moderadora de la relación entre la cantidad de luz y la altura de la planta, que deberíamos tener en cuenta en cualquier meta-análisis.

En nuestro caso, variables como la cantidad de variables controladas (`controls`), el diseño de cada estudio (`design`), la forma en la que se midieron la adherencia al tratamiento (`a_measure`) y la escrupulosidad (`c_measure`), la edad de los participantes (`meanage`) y/o la calidad metodológica de los estudios (`quality`), son características que podrían haber afectado el coeficiente de correlación que encontraron (todas estas variables se explican con mayor detalle en la sección 2.1). Por ejemplo, estudios con menor calidad metodológica podrían haber tendido a sobreestimar el efecto, encontrando correlaciones más fuertes y positivas entre escrupulosidad y adherencia al tratamiento, sesgando así nuestra estimación del efecto *real*.

Los moderadores pueden ser tanto variables continuas, como categóricas, y en las siguientes secciones daré ejemplos para cada caso.

4.1. Ejemplo 1: Moderación de una variable continua (edad promedio de los participantes)

Primero, y como ejemplo, vamos a ver si la edad promedio de los participantes de un estudio (en nuestros datos, `meanage`) modera el resultado. Esto es importante, pues hay una enorme variación entre las edades medias de los participantes de los diferentes estudios¹⁷, lo que podría moderar (afectar) la asociación entre escrupulosidad (`conscientiousness`) y adherencia a la medicación prescrita.

Para esto, de nuevo podemos usar la función `rma` de paquete `metafor` de la misma manera que en la sección 3, pero agregando nuestra variable moderadora (`meanage`) al argumento `mods`. En este caso voy a asignar a un objeto llamado `res.modage`, para diferenciarlo del objeto `res` al que asigné el meta-análisis básico, sin moderadores.

```
res.modage <- rma(yi = yi, vi = vi,
                     mods = ~meanage,
                     data = dat)
```

Los resultados, son los siguientes:

```
res.modage
```

¹⁷De hecho, mientras que en el estudio de Axelsson et al. (2009) la edad promedio fue de 22, en el estudio de Jerant et al. (2011) la edad promedio fue de 78.6.

Consola de R: Output 15

```
Mixed-Effects Model (k = 16; tau^2 estimator: REML)

tau^2 (estimated amount of residual heterogeneity): 0.0072 (SE = 0.0054)
tau (square root of estimated tau^2 value): 0.0846
I^2 (residual heterogeneity / unaccounted variability): 56.50%
H^2 (unaccounted variability / sampling variability): 2.30
R^2 (amount of heterogeneity accounted for): 11.76%

Test for Residual Heterogeneity:
QE(df = 14) = 30.9050, p-val = 0.0057

Test of Moderators (coefficient 2):
QM(df = 1) = 1.4286, p-val = 0.2320

Model Results:

estimate      se      zval     pval    ci.lb    ci.ub
intrcpt   0.2741  0.1090  2.5147  0.0119  0.0605  0.4877 *
meanage   -0.0024  0.0020 -1.1952  0.2320 -0.0063  0.0015

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Estos resultados tienen la misma organización que los del análisis sin moderadores (sección 3). Este resultado nos muestra que, a pesar de la gran diferencia de edad entre estudios, la edad no tiene un efecto significativo, como se puede ver en la sección “Test of Moderators (coefficient 2)” (al final nos muestra el valor p como “p-val = 0.2320”), así como los resultados para el efecto de `meanage` en la tabla `Model Results` (donde nos da el mismo resultado: “0.2320”).

En la tabla de `Model results` los estimados (columna `estimate`) se pueden leer tal cual como si fueran una regresión, pero teniendo en cuenta que acá nuestra variable dependiente sería la correlación entre escrupulosidad y adherencia al tratamiento (en valores z). Entonces, como el intercepto (`intrcpt`) tiene un valor estimado de 0.2741, para estudios con participantes que tuviesen una edad media de 0 años, se estimaría que la asociación entre escrupulosidad y adherencia al tratamiento sería de $z = 0.2741$. Por su parte el efecto de la edad media (`meanage`), nuestro moderador, es de -0.0024, lo que indica que por cada año de aumento en la edad promedio de los participantes, la correlación entre escrupulosidad y adherencia disminuye $z = -0.0024$. Para decirlo de manera sencilla, en nuestra muestra de estudios, la correlación entre adherencia y escrupulosidad tiende a disminuir con la edad, pero esta tendencia no es significativa.

4.1.1. Más información e interpretación de la moderación

Para más información, podemos *predecir* el efecto (en este caso, la correlación entre la escrupulosidad y la adherencia a la medicación), a diferentes edades, usando la función `predict`.

En este ejemplo, como argumentos de esta función incluiré el objeto que contiene el meta-análisis con la edad como variable moderadora (`res.modage`), y los los valores para los cuales quiero saber el coeficiente de correlación predicho (argumento `newmods`); en este ejemplo voy a usar “`seq(20, 80, by = 10)`”, lo que produce la secuencia de números de 20 a 80, cada 10 (es decir: 20, 30, 40, 50, 60, 70, 80), que serán las edades para las cuales obtendré el coeficiente de correlación estimado por el modelo¹⁸.

Adicionalmente, voy a convertir esta tabla a un objeto clase `data.frame` (usando la función `as.data.frame()`), crearé una nueva variable llamada `meanage` que contiene las edades para las cuales he calculado la predicción (`mutate(meanageage = seq(20, 80, by = 10))`¹⁹), y voy a reorganizar el orden de las columnas (`select(7, 1:6)`). Asignaré esta tabla a un objeto llamado `pred.res.modage`.

¹⁸Escogí edades entre 20 y 80 como referencia, pues están cerca al rango de valores de las edades promedio de los estudios meta-analizados (22 a 78.6), y con edades cada 10 años, para obtener un número manejable pero informativo de predicciones.

¹⁹Para agregar los valores de la variable moderadora para los cuales se hace la predicción, alternativamente se puede agregar el argumento `addx = TRUE` a la función `predict` (en este caso, por ejemplo, `predict(res.modage, newmods = seq(20, 80, by = 10), addx = TRUE)`). Sin embargo, para evitar crear columnas adicionales y tener una tabla en un formato más sencillo y claro, he decidido crear la columna `meanage` manualmente. Si usara el argumento `addx = TRUE`, crearía las columnas `X.intrcpt` con el intercepto, y `X.meanage` con los valores de `meanage`.

```
# Calcular efecto ajustado para diferentes edades
pred.res.modage <- predict(res.modage, newmods = seq(20, 80, by = 10)) |>
  as.data.frame() |>
  mutate_all(~round(., 3)) |>
  mutate(meanage = seq(20, 80, by = 10)) |>
  select(7, 1:6)
# Ver la tabla
pred.res.modage
```

Consola de R: Output 16

	meanage	pred	se	ci.lb	ci.ub	pi.lb	pi.ub
1	20	0.226	0.071	0.086	0.366	0.009	0.443
2	30	0.202	0.054	0.097	0.307	0.006	0.398
3	40	0.178	0.039	0.102	0.254	-0.005	0.360
4	50	0.154	0.031	0.094	0.214	-0.022	0.330
5	60	0.130	0.034	0.063	0.197	-0.049	0.309
6	70	0.106	0.047	0.013	0.198	-0.084	0.296
7	80	0.082	0.064	-0.043	0.207	-0.126	0.290

Con esto, el objeto `pred.res.modage`, al que he asignado esta predicción, tiene las siguientes columnas:

- **meanage**: edad promedio de los participantes de los estudios meta-analizados
- **pred**: valor del efecto predicho (en este caso, coeficiente de correlación transformado en z de Fisher), para cada edad promedio
- **se**: error estándar (en inglés *standard error*) del efecto predicho
- **ci.lb**: límite inferior del intervalo de confianza del 95 % (en inglés *confidence interval lower bound*) del efecto predicho
- **ci.ub**: límite superior del intervalo de confianza del 95 % (en inglés *confidence interval upper bound*) del efecto predicho
- **pi.lb**: límite inferior del intervalo de predicción (en inglés *prediction interval lower bound*), y
- **pi.ub**: límite superior del intervalo de predicción (en inglés *prediction interval upper bound*)

Las columnas `age` (edad) y `pred` (efecto predicho), nos muestran que, para estudios donde los participantes son en promedio más jóvenes, se esperaría una correlación más fuerte; por ejemplo, mientras que para estudios con personas de 20 años de edad en promedio se esperaría una correlación de $z = 0.226$, para estudios con participantes con una edad promedio de 80 años, el modelo estima un efecto de $z = 0.082$. En otras palabras, en nuestra muestra de estudios, la asociación entre escrupulosidad y adherencia a la medicación tiende a reducirse ligeramente con la edad promedio de la muestra.

Sin embargo, este efecto moderador de la edad no es significativo, por lo que no tenemos evidencia suficiente para inferir que se da más allá de nuestra muestra de estudios.

4.1.1.1. Meta-Analytic Scatter Plot (Gráfico de dispersión meta-analítico) Alternativamente, cuando tenemos un modelo con moderador, también podemos ver la asociación entre la variable moderadora, y el efecto de cada estudio meta-analizado, a modo de regresión. La función `regplot` hace precisamente esto (Fig. 25).

```
regplot(res.modage,
        ylab = "Coeficiente de correlación (z de Fisher)",
        xlab = "Edad promedio de muestra")
```

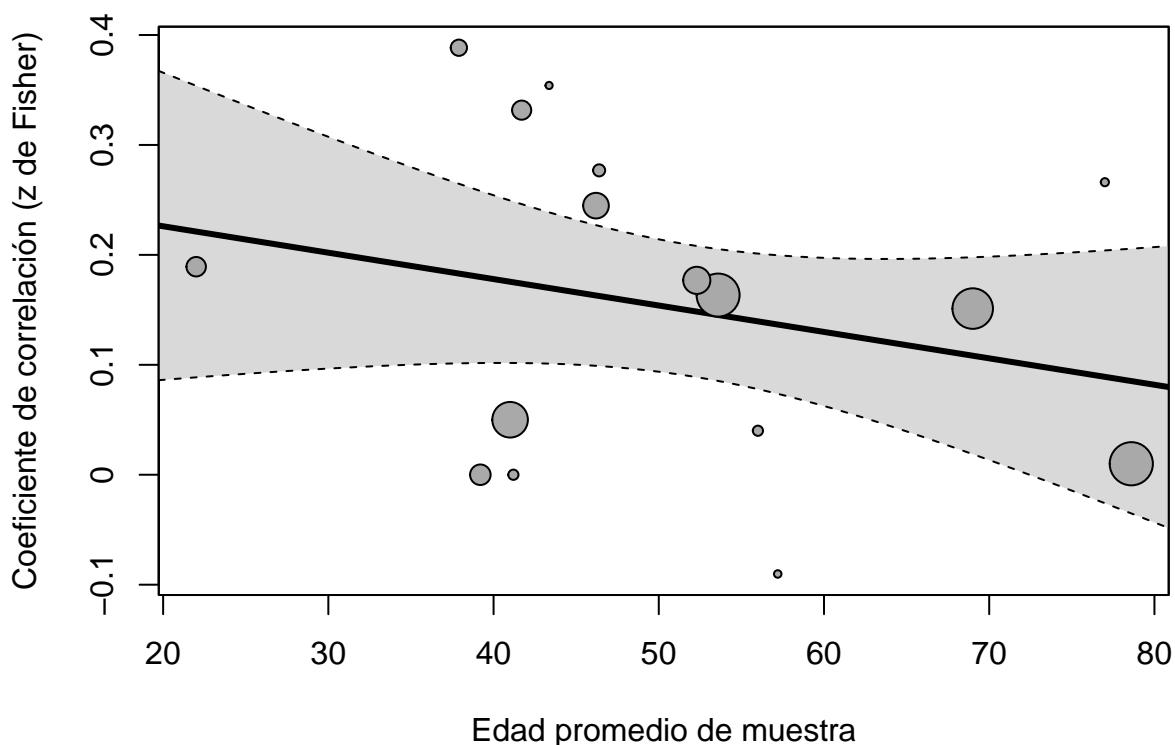


Figura 25. Gráfico de dispersión meta-analítico (*Meta-Analytic Scatter Plot*). El tamaño de los puntos es proporcional al peso que recibieron los estudios en el meta-análisis (puntos más grandes para los estudios con más peso, pues tienen un tamaño de muestra mayor y con un menor error estimado). La línea negra representa el efecto previsto en función del predictor (en este caso `meanage`, edad promedio), que por supuesto coincide con las predicciones del objeto `pred.res.modage` (Output 16, en la sección 4.1.1); la banda gris delimitada por líneas punteadas representa el intervalo de confianza del 95 %.

4.1.2. Forest plot y funnel plot

Por supuesto, de estos resultados también puedo crear *forest plots* y *funnel plots*, siguiendo los ejemplos y código de la sección 3. Para el *forest plot*, hago a continuación un ejemplo anotado y mejorado²⁰ (Fig. 26, con un código similar al usado como ejemplo en la Fig. 7). Sin embargo, es importante tener en cuenta que esta opción no creará un resumen del meta-análisis, ya que no tenemos un solo efecto *real* estimado como producto del meta-análisis.

```
# forest plot con anotaciones adicionales
forest(res.modage, cex = 0.75, xlim = c(-1.6, 1.6),
       slab = paste(dat$authors, dat$year, sep = ", "),
       showweights = TRUE,
       xlab = "Coeficiente de correlación (z de Fisher)",
       digits = c(2,3L))

# agregar encabezados a las columnas (valores de X y Y deben ser ajustados)
par(cex = 0.8, font = 2)
text(x = -1.6, y = 18, labels = "Autor(es), Año", pos = 4)
text(x = 0, y = 18, labels = "Efecto e IC", pos = 4)
text(x = 1, y = 18, labels = "Peso", pos = 2)
text(x = 1.6, y = 18, labels = "Corr. [95% IC]", pos = 2)
```

²⁰La función `viz_forest` del paquete `metaviz` no puede crear un *forest plot* de un meta-análisis ajustado con `metafor` cuando este modelo contiene variables moderadoras continuas, o cuando tiene como variable moderadora más de una variable categórica. En otras palabras, sólo podrá hacer el *forest plot* si nuestro meta-análisis no tiene moderadores, o tiene un único moderador categórico. Como en este ejemplo nuestro moderador es una variable continua, no es posible usar esta función.

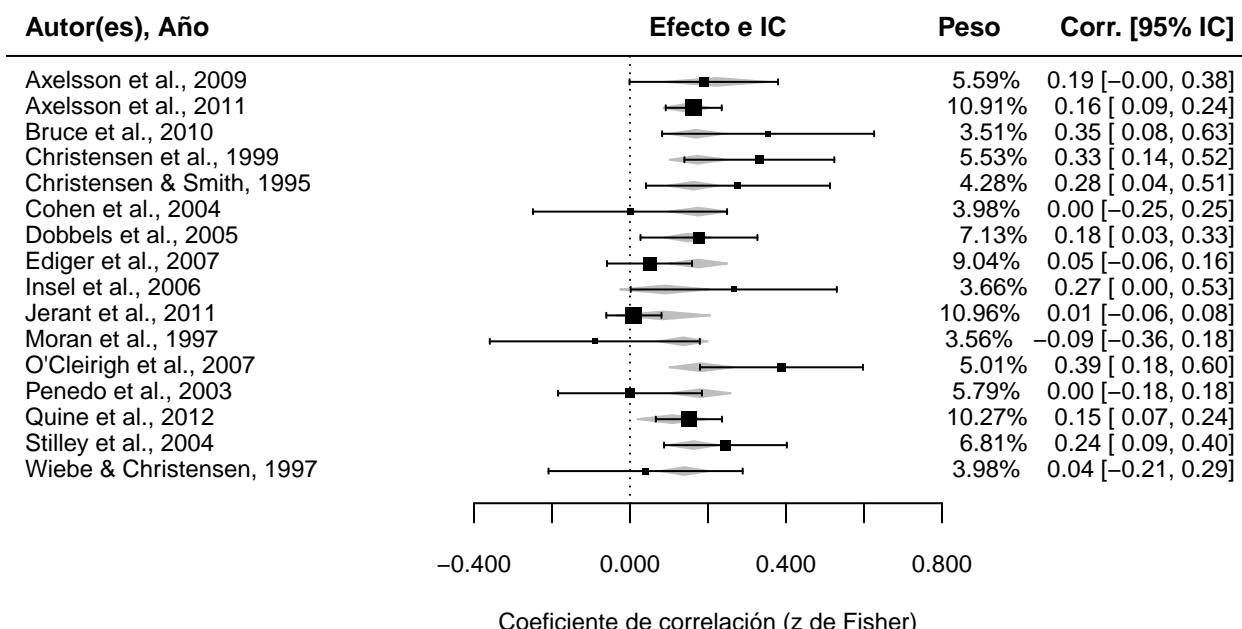


Figura 26. Forest plot básico de `metafor`, para un meta-análisis incluyendo la edad promedio de los participantes como moderador. En la ilustración gráfica, además de los efectos originales, se puede ver el efecto de cada estudio estimado cuando se incluye el moderador como polígonos (diamantes) de color gris. Sin embargo, ya no obtenemos una fila al final representando el efecto promediado del meta-análisis, ya que no tenemos un solo efecto.

De manera similar, podemos obtener un *funnel plot* de nuestro meta-análisis con moderador. Sin embargo, tanto el paquete `weightr` como el paquete `metafor` tienen funciones llamadas `funnel`, lo que puede crear un conflicto. Para evitar esto, podemos explícitamente pedir a R que use la función `funnel` del paquete `metafor` con el comando `metafor::funnel`:

```
metafor::funnel(res.modage, xlab = "Valor residual", ylab = "Error estándar")
```

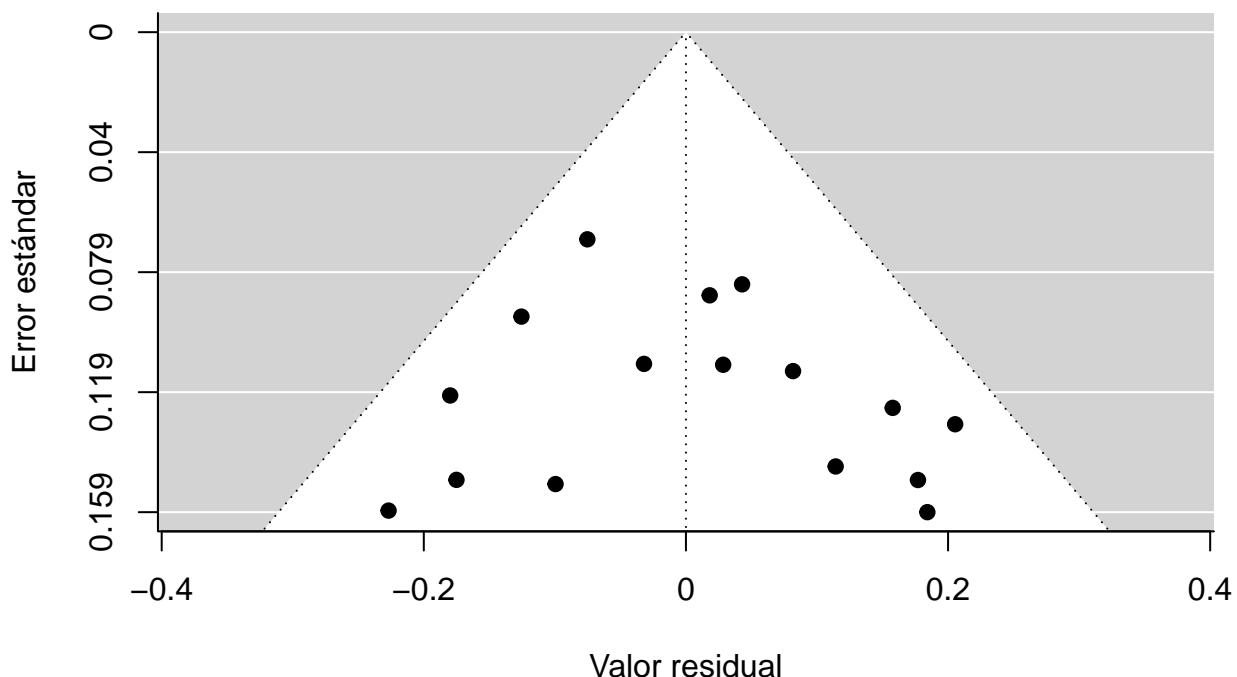


Figura 27. Funnel plot básico de `metafor`, para un meta-análisis incluyendo la edad promedio de los participantes como moderador, y con títulos de los ejes en español. La línea punteada vertical representa el efecto meta-analizado que hemos encontrado, así que podemos ver los estudios que encontraron un efecto mayor (derecha de la línea punteada) o menor (izquierda) de éste.

Este *funnel plot* (Fig. 9), a diferencia los los anteriores, nos muestra los valores residuales de cada estudio en vez de los coeficientes de correlación (transformados a z de Fisher). Es decir, nos muestra qué tanto se aleja cada estudio del resultado de nuestro meta-análisis, que tiene un valor residual de 0 (pues no se aleja de sí mismo).

4.1.3. Reporte del meta-análisis con moderador continuo

Por supuesto, al igual que como se explica en la sección 3.2, también podemos estimar intervalos de confianza de las medidas de heterogeneidad, con la función `confint(res.modage)`. Incluyendo esos intervalos de confianza el modelo se podría resumir, por ejemplo, así:

Ejemplo de reporte con moderador continuo

Dado que hay una considerable variación en la edad media de los participantes entre los estudios incluídos (con un rango entre 22 y 78.6), volvimos a ajustar el mismo meta-análisis, pero incluyendo la edad media de los participantes de cada estudio como variable moderadora. Esta meta-regresión reveló que, aunque en los estudios incluidos la asociación entre escrupulosidad y adherencia disminuye al aumentar la edad media de las muestras, esta moderación no es significativa (intercepto: $z \pm se = 0.27 \pm 0.11$, IC 95% [0.06, 0.49]; $Z = 2.51$, $p = .012$; edad promedio: $z \pm se = -0.002 \pm 0.002$, IC 95% [-0.006, 0.002]; $Z = -1.195$, $Q(1) = 1.429$, $p = .232$).

Adicionalmente, la heterogeneidad moderada y significativa se mantuvo, incluso al incluir este moderador ($\tau^2 \pm se = 0.0072 \pm 0.0054$, IC 95% [0.0010, 0.0403]; $\tau = 0.0846$, IC 95% [0.0315, 0.2007]; $Q(14) = 30.91$, $p = .006$; $I^2 = 56.50\%$, IC 95% [15.24%, 87.97%]). Por esto, no podemos inferir que la edad promedio de las muestras tenga un efecto en la estimación de la asociación entre escrupulosidad y adherencia al tratamiento.

Por supuesto, este resultado está pensado acá para ser reportado tras el reporte del meta-análisis básico (ejemplos en las secciones 3.1.1 y 3.2.1).

De nuevo, ten en cuenta que siempre es buena idea citar tablas y figuras relevantes junto al reporte de los resultados que ilustran, pues esto facilitará la comunicación de resultados complejos. En este caso, yo citaría el gráfico de dispersión meta-analítico (*Meta-Analytic Scatter Plot*; sección 4.1.1.1) al presentar los resultados de la edad promedio, y el *forest plot* y el *funnel plot* de este meta-análisis con moderador (sección 4.1.2) al hablar de heterogeneidad.

Aunque estoy siendo extremadamente exhaustivo en la manera de reportar toda la información, hay diferentes formas de reportar tus resultados. Puedes por supuesto usar ejemplos de artículos que reporten meta-análisis para usarlos como modelo; por ejemplo, Salsman et al. (2015) y L. Shi et al. (2010) hacen reportes distintos pero claros, y el estudio de Hidalgo-Fuentes (2022) es un ejemplo sencillo del reporte de un meta-análisis de correlación, con y sin moderador, en español.

4.2. Ejemplo 2: Moderación de una variable categórica (controles usados en cada estudio meta-analizado)

Como segundo ejemplo, voy a mirar si el hecho de que los estudios tengan variables que fueron controladas, modera la asociación entre escrupulosidad (*conscientiousness*) y adherencia a la medicación prescrita. Siguiendo los mismos pasos, voy hacer este análisis, pero voy a asignar este meta-análisis a un objeto llamado `res.contr`.

```
res.contr <- rma(yi = yi, vi = vi, mods = ~controls, data = dat)
res.contr
```

Consola de R: Output 17

```
Mixed-Effects Model (k = 16; tau^2 estimator: REML)

tau^2 (estimated amount of residual heterogeneity): 0.0000 (SE = 0.0015)
tau (square root of estimated tau^2 value): 0.0002
I^2 (residual heterogeneity / unaccounted variability): 0.00%
H^2 (unaccounted variability / sampling variability): 1.00
R^2 (amount of heterogeneity accounted for): 100.00%

Test for Residual Heterogeneity:
QE(df = 14) = 18.0370, p-val = 0.2051
```

```

Test of Moderators (coefficient 2):
QM(df = 1) = 20.1221, p-val < .0001

Model Results:

            estimate      se     zval   pval    ci.lb    ci.ub
intrcpt       0.1788  0.0208  8.6044 <.0001   0.1380   0.2195  ***
controlsmultiple -0.1621  0.0361 -4.4858 <.0001  -0.2329  -0.0913  ***

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

En este caso, a diferencia del ejemplo de moderación anteriores, la variable moderadora (`controls`) sí tiene un efecto significativo, como se puede ver en el “`Test of Moderators (coefficient 2)`” y en la columna `pval` para el efecto de `controlsmultiple` (<.0001), así como en los asteriscos que aparecen al final de esa fila (***)�.

De nuevo, en la tabla de `Model results` los estimados (columna `estimate`) se pueden leer tal cual como si fueran una regresión. Sin embargo es importante tener en cuenta que, dado que nuestra variable moderadora es categórica, el intercepto acá será en valor estimado de la correlación para el primer nivel (o categoría) de nuestra variable moderadora. Si recuerdas, como se explica en la sección 2.1 la variable que estamos usando como moderadora, `controls` contiene información de si cada estudio no controló ninguna variable (`none`), o controló múltiples variables (`multiple`). Es decir, es una variable categórica con 2 niveles (`none, multiple`).

Entonces, como el intercepto (`intrcpt`) tiene un valor estimado de 0.1788, se estima que la asociación entre escrupulosidad y adherencia al tratamiento fue de $z = 0.1788$ para estudios que no controlaron ninguna variable. Después, frente al efecto de nuestro moderador (`controlsmultiple`, que pone de manera conjunta la variable `controls` más el nivel `multiple`), nos da un estimado de -0.1621, lo que quiere decir que la correlación disminuye $z = -0.1621$ cuando se trata de estudios que controlaron múltiples variables, con relación al intercepto (`none`, en nuestro caso). De manera concreta, el modelo estimó una correlación de $z = 0.1788$ para estudios que no controlaron ninguna variable, y una correlación de $z = 0.0167$ (o, lo que es lo mismo, $0.1788 - 0.1621$) cuando los estudios usaron varios controles.

4.2.1. Más información e interpretación de la moderación

Al igual que en el ejemplo con un moderador continuo (sección 4.1.1), puedo *predicir* el efecto (en este caso, la correlación entre la escrupulosidad y la adherencia a la medicación), cuando hay o no controles, usando la función `predict`.

```

pred.res.contr <- predict(res.contr, newmods = c(0, 1)) |>
  as.data.frame() |>
  mutate_all(~round(., 3)) |>
  mutate(controls = levels(dat$controls)) |>
  rename(yi = pred) |>
  select(7, 1:6)
pred.res.contr

```

Consola de R: Output 18

	controls	yi	se	ci.lb	ci.ub	pi.lb	pi.ub
1	none	0.179	0.021	0.138	0.219	0.138	0.219
2	multiple	0.017	0.030	-0.041	0.075	-0.041	0.075

En este ejemplo, como argumentos de esta función incluí el objeto que contiene el meta-análisis con la edad como variable moderadora (`res.contr`), y los los valores para los cuales quiero saber el coeficiente de correlación predicho (argumento `newmods`); dado que la variable moderadora es categórica, el modelo genera variables *dummy* asignando valores de 0 y 1 a los niveles de esta variable (en este caso, 0 = ningún control; 1 = múltiples controles). Por esto, al asignar 0 y 1 al argumento `newmods` (i.e. `newmods = c(0, 1)`), la función `predijo` la correlación para estudios sin ningún control y para estudios con múltiples controles. También convertí la tabla a un objeto clase `data.frame` (usando la función `as.data.frame()`), creé una nueva variable llamada `controls` que contiene las categorías para

las cuales calculé la predicción (`mutate(controls = levels(dat$controls))`), que en este caso corresponden con los niveles de la variable `controls`), y reorganicé el orden de las columnas (`select(7, 1:6)`). El objeto resultante fue asignado a `pred.res.contr`.

4.2.1.1. Meta-Analytic Scatter Plot (Gráfico de dispersión meta-analítico) Al igual que en el ejemplo de meta-análisis con moderación de una variable continua (sección 4.1), es posible hacer un gráfico de dispersión meta-analítico (*meta-analytic scatter plot*) usando la función `regplot` del paquete `metafor`. Sin embargo, dado que la variable moderadora es categórica, y el modelo generó variables *dummy* asignando valores de 0 y 1 a los niveles de esta variable (0 = ningún control; 1 = múltiples controles), en el ejemplo siguiente agregué esta información a la descripción del eje *X*.

```
regplot(res.contr,
        ylab = "Coeficiente de correlación (z de Fisher)",
        xlab = "Controles (0 = ninguno; 1 = múltiples)")
```

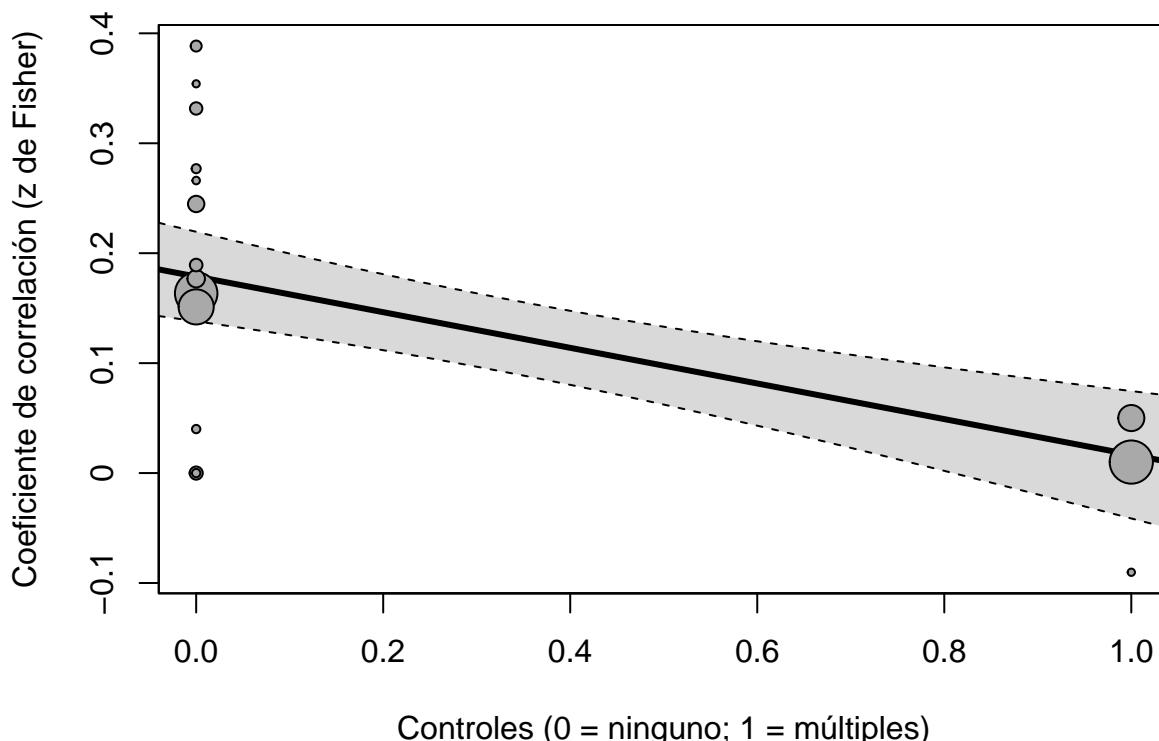


Figura 28. Gráfico de dispersión meta-analítico (*Meta-Analytic Scatter Plot*) básico de `metafor` creado con la función `regplot`. El tamaño de los puntos es proporcional al peso que recibieron los estudios en el meta-análisis (puntos más grandes para los estudios con más peso, pues tienen un tamaño de muestra mayor y con un menor error estimado). La línea negra representa el efecto previsto en función del predictor (en este caso `controls`, `controles`). La banda gris delimitada por líneas punteadas representa el intervalo de confianza del 95 %. Dado que la variable moderadora es categórica, el modelo genera variables *dummy* asignando valores de 0 y 1 a los niveles de esta variable (en este caso, 0 = ningún control; 1 = múltiples controles), tal y como se describe en el eje *X*. Para una versión más apropiada, ver Figura 29.

Aunque la Figura 28 presenta información correcta, tal vez no es la más adecuada ni la más clara cuando se trata de representar meta-análisis con moderadores categóricos. Por ejemplo, a pesar de que nuestro moderador es un factor (`controles`) con dos niveles (ninguno, múltiples), en el eje *X* se representa como una variable continua con valores entre 0 y 1, que además tiene valores intermedios (0.2, 0.4, 0.6, 0.8) que en nuestro caso no tienen sentido.

Sin embargo, de manera un poco más *artesanal*, es posible crear una versión más adecuada usando, por ejemplo, `ggplot2`. El código siguiente permite crear una figura mucho más adecuada usando este paquete.

```
# Definir base de datos, así como ejes X (controls) y Y (yi)
ggplot(dat, aes(x = controls, y = yi)) +
  # Agregar puntos para cada estudio, con tamaño y color según tamaño de muestra (ni)
```

```

geom_point(aes(size = ni, color = ni), alpha = 0.5) +
# Definir escala de colores
scale_colour_gradient(low = "red", high = "blue") +
# Definir rango de tamaño de los puntos
scale_size_continuous(range = c(1, 10)) +
# Combinar colores y tamaños de puntos en una sola leyenda
guides(color = guide_legend(), size = guide_legend()) +
# Traducir etiquetas del eje X en español
scale_x_discrete(labels = c("none" = "Ninguno", "multiple" = "Múltiples")) +
# Cambiar títulos de ejes a español
labs(x = "Controles", y = "Coeficiente de correlación (z de Fisher)") +
# Agregar barras de error para cada categoría, con base en la predicción de la sección 4.2.1
geom_errorbar(data = pred.res.contr, mapping = aes(ymin = ci.lb, ymax = ci.ub),
               width = 0.1, color = "black") +
# Agregar puntos blancos representando la predicción de la sección 4.2.1 para cada categoría
geom_point(data = pred.res.contr, shape = 21, size = 3, color = "black", fill = "white") +
# Cambiar título de leyenda
labs(color = "Tamaño de \nmuestra", size = "Tamaño de \nmuestra") # "\n" para salto de línea

```

El código anterior puede verse algo confuso para quien no haya usado `ggplot2` antes. Por esto, he agregado anotaciones, función por función, para ayudar a su interpretación²¹. Este código produce la Figura 29.

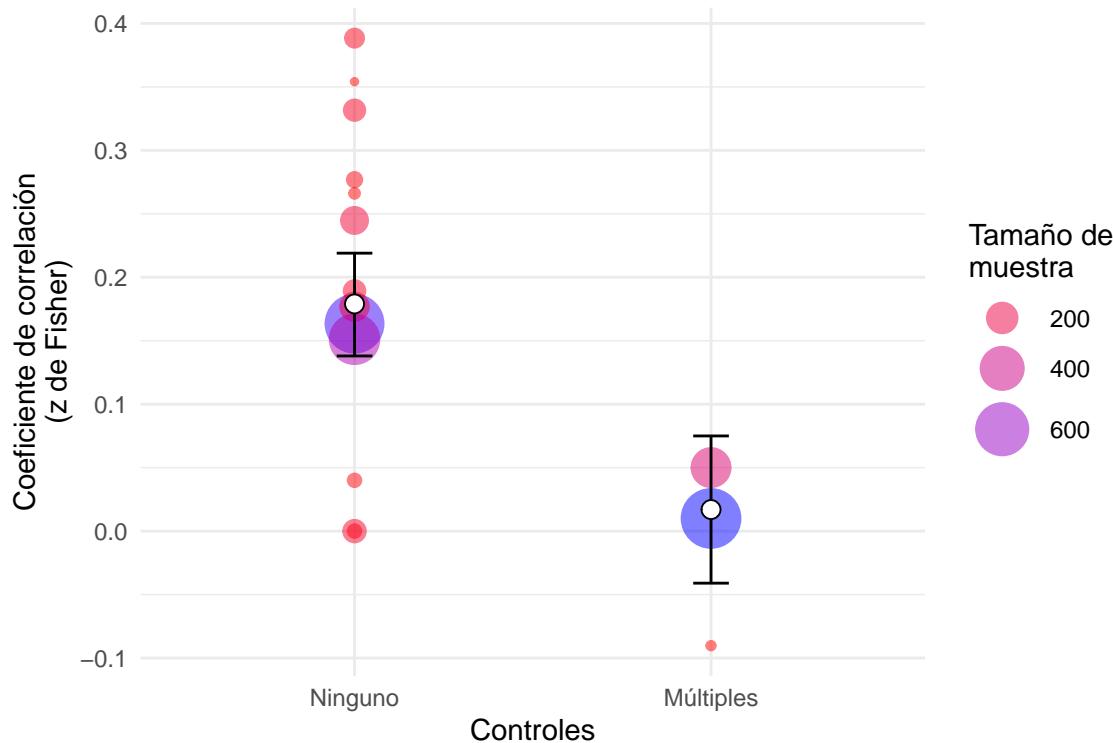


Figura 29. Gráfico de dispersión meta-analítico (*Meta-Analytic Scatter Plot*) creado manualmente con `ggplot2` para hacer una mejor representación de un moderador categórico. Los puntos de colores representan el coeficiente de correlación en función de la presencia o ausencia de controles. El tamaño de los puntos es proporcional al tamaño de muestra de los estudios incluidos en el meta-análisis (puntos más grandes y azules para los estudios con mayor tamaño de muestra). Los puntos blancos superpuestos representan el efecto estimado para cada categoría, y las barras de error representan los intervalos de confianza del 95 %.

²¹Debido a sus enormes y numerosas posibilidades, `ggplot2` es quizás la opción más poderosa para hacer gráficos estadísticos en R; por esto explicar sus bases supera por mucho el alcance de esta guía. Sin embargo, hay gran cantidad de opciones disponibles en internet. Por ejemplo, puedes leer este [tutorial](#) (López Peñaranda, 2019) o ver este [video](#) (Datademia, 2018).

4.2.2. Forest plot y funnel plot

Por supuesto, *forest plots* y *funnel plots* pueden ser creados, tal y como describí en la sección 4.1.2. Para el *forest plots* podríamos usar el siguiente código:

```
# forest plot con anotaciones adicionales
forest(res.contr, cex = 0.75, xlim = c(-1.6, 1.6),
       slab = paste(dat$authors, dat$year, sep = ", "),
       showweights = TRUE,
       xlab = "Coeficiente de correlación (z de Fisher)",
       digits = c(2,3L))

# agregar encabezados a las columnas (valores de X y Y deben ser ajustados)
par(cex = 0.8, font = 2)
text(x = -1.6, y = 18, labels = "Autor(es), Año", pos = 4)
text(x = 0, y = 18, labels = "Efecto e IC", pos = 4)
text(x = 1, y = 18, labels = "Peso", pos = 2)
text(x = 1.6, y = 18, labels = "Corr. [95% IC]", pos = 2)
```

Ese código produce la Figura 30.

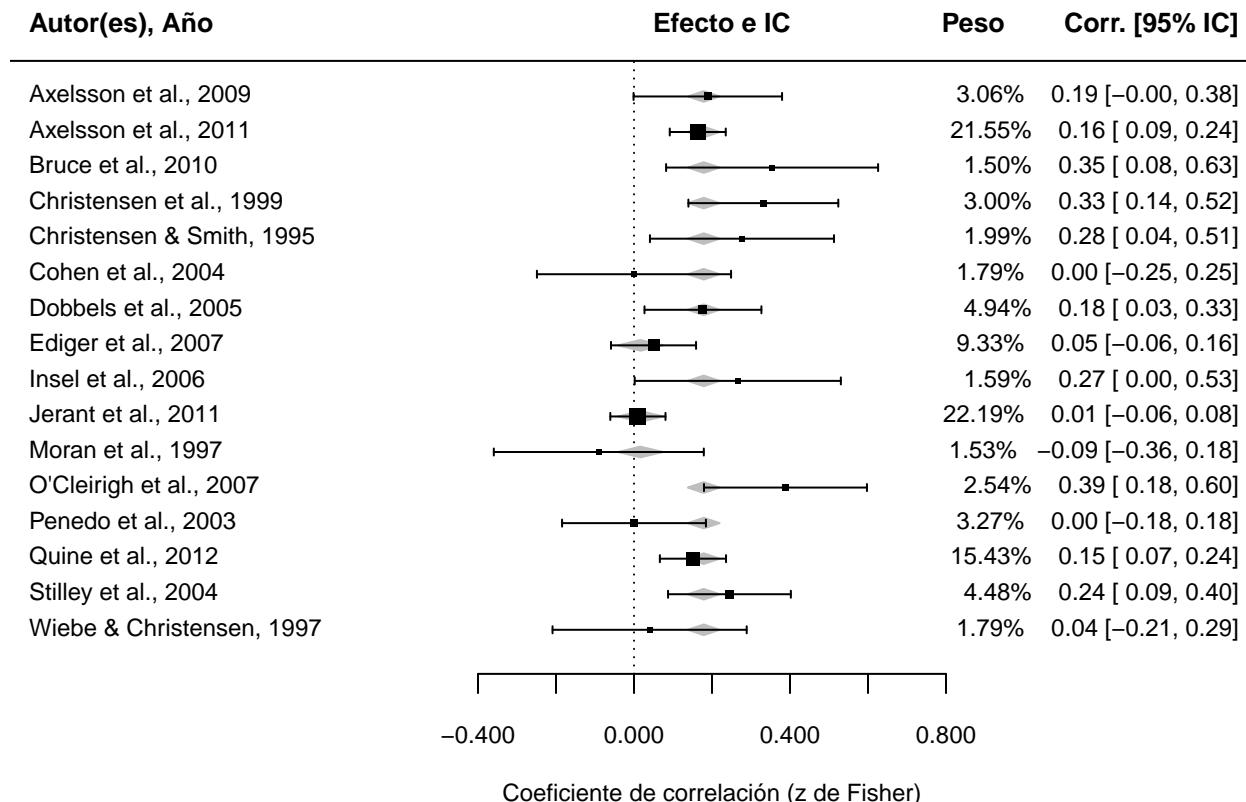


Figura 30. Forest plot básico de `metafor`, para un meta-análisis incluyendo como moderador si los estudios tuvieron en cuenta múltiples controles estadísticos o ninguno. En la ilustración gráfica, además de los efectos originales, se puede ver el efecto de cada estudio estimado cuando se incluye el moderador como polígonos (diamantes) de color gris. Sin embargo, ya no obtenemos una fila al final representando el efecto promediado del meta-análisis, ya que no tenemos un solo efecto.

De manera similar, podemos obtener un *funnel plot*, pero recuerda que dado que tanto el paquete `weightr` como el paquete `metafor` tienen funciones llamadas `funnel`, debemos pedirle a R explícitamente que use la función `funnel` del paquete `metafor` con el comando `metafor::funnel`. Este *funnel plot*, nos mostrará los valores residuales de cada estudio (es decir, qué tanto se alejan del resultado de nuestro meta-análisis, que tiene un valor residual de 0; Fig. 9), en vez de los coeficientes de correlación (transformados a *z* de Fisher).

Adicionalmente no olvides que este **funnel plot** nos mostrará los valores residuales de cada estudio, dado que se trata de un meta-análisis con moderador, tal como lo expliqué en la sección 4.1.2.

```
metafor::funnel(res.contr,
  xlab = "Valor residual",
  ylab = "Error estándar")
```

Esto producirá la siguiente Figura 11.

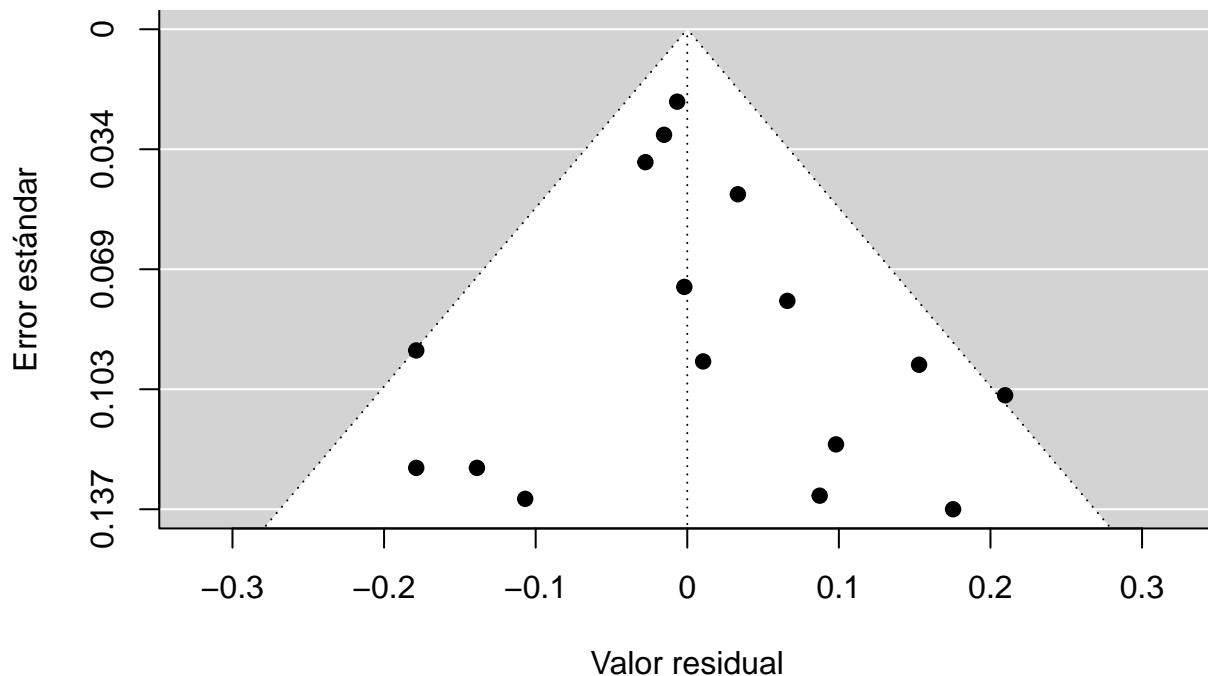


Figura 31. Funnel plot básico de `metafor`, para un meta-análisis incluyendo como moderador si los estudios tuvieron en cuenta múltiples controles estadísticos o ninguno. La línea punteada vertical representa el efecto meta-analizado que hemos encontrado, así que podemos ver los estudios que encontraron un efecto mayor (derecha de la línea punteada) o menor (izquierda) de éste.

4.2.3. Reporte del meta-análisis con moderador categórico

De nuevo, y tal como lo hicimos en la sección 3.2 y en la sección 4.1.3, es posible estimar intervalos de confianza de las medidas de heterogeneidad. Para ello, en este caso usamos la función `confint(res.contr)`. El reporte incluyendo esta información podría ser algo como esto:

Ejemplo de reporte con moderador categórico

Dado que hay estudios que controlaron variables y otros que no controlaron ninguna, ajustamos de nuevo el meta-análisis, pero incluyendo este factor como variable moderadora. Al incluir la variable moderadora de controles en el meta-análisis, se encontró que esta tuvo un efecto significativo en la estimación de la correlación entre la escrupulosidad y la adherencia al tratamiento ($Q(1) = 20.122, p < .0001$). De hecho, los estudios que no incorporaron controles estimaron una asociación más alta ($z \pm se = 0.179 \pm 0.021$, IC 95% [0.138, 0.219]) que los estudios que incluyeron múltiples controles ($z \pm se = 0.017 \pm 0.030$, IC 95% [-0.041, 0.075]).

Al incluir esta variable moderadora, se redujo enormemente la heterogeneidad entre los estudios ($\tau^2 \pm se = 0.00 \pm 0.0015$, IC 95% [0.0000, 0.0244]; $\tau = 0.0002$, IC 95% [0.000, 0.1562]; $Q(14) = 18.037, p = .205$; $I^2 = 0.00\%$, IC 95% [0.00%, 80.96%]), lo que indica que la heterogeneidad original se debía en gran parte a las diferencias metodológicas entre los estudios.

Como antes, es importante tener en cuenta que este resultado debería ser reportado después del meta-análisis básico, como se muestra en las secciones 3.1.1 y 3.2.1. Además, al presentar los resultados, te recomiendo citar las tablas y figuras pertinentes que los ilustren, ya que esto puede facilitar la comunicación de resultados complejos.

En particular, al reportar los resultados de la moderación, se podría hacer referencia al gráfico de dispersión meta-analítico (sección 4.2.1.1). Asimismo, para hablar de la heterogeneidad, se podría citar el *forest plot* y el *funnel plot* correspondientes a este meta-análisis con moderador (sección 4.2.2).

Aunque es importante proporcionar información detallada y exhaustiva en el reporte de los resultados de un meta-análisis, existen diferentes formas de hacerlo. Una recomendación es utilizar como modelos algunos artículos que hayan reportado meta-análisis previamente. Por ejemplo, el trabajo de Hidalgo-Fuentes (2022) es un ejemplo sencillo de cómo reportar un meta-análisis de correlación, tanto con como sin moderador, en español. Por otro lado, los estudios de Salsman et al. (2015) y L. Shi et al. (2010) presentan reportes de manera distinta pero clara (en inglés).

5. Recomendaciones generales

Hacer un meta-análisis de correlación en R no es necesariamente complejo, pero podría requerir acciones que pueden parecer algo laberínticas. Sin embargo, la confiabilidad del resultado que obtengas, dependerá de varios aspectos. Por esto, la realización de un meta-análisis requiere una comprensión de los métodos estadísticos involucrados, la selección y evaluación de estudios relevantes, y la identificación y consideración de posibles fuentes de heterogeneidad y sesgo.

5.1. Pasos para hacer tu meta-análisis

Primero, debes tener hacer una selección de artículos buena y no sesgada²². Esto mínimamente incluye:

Con esto deberías ya poder hacer una base de datos para un meta-análisis de correlación simple. Mientras tengas en tu base de datos algún identificador de cada estudio, el tamaño de efecto (r) y el tamaño de muestra (n), tendrías todo lo necesario para hacerlo. Por ejemplo:

estudio	ri	ni
Autores (año)	0.25	425
Autores (año)	0.65	45
Autores (año)	0.12	235

Adicionalmente, podrías definir o extraer información de posibles variables moderadoras; por ejemplo, si los estudios se pudieran agrupar según metodología usada o población estudiada, y sospechas que estas variables afectan la correlación encontrada en cada estudio, podrías agregar esta información en una o más variables que podrías usar como moderadores.

Una vez tengas tu base de datos lista, deberías, cuando menos:

1. Hacer el meta-análisis (sin moderador, sección 3; con moderador, sección 4)
 - 1.1 Reportar información sobre heterogeneidad (sección 3.2)
 - 1.2 Hacer un diagnóstico de influencia (sección 3.3; incluir, si es posible y no tienes un número grande de estudios, un meta-análisis combinatorio, sección 3.8.2, e idealmente la identificación de estudios que contribuyen a los patrones de heterogeneidad a partir del *GOSH plot*, sección 3.8.2.1)
 - 1.3 Reportar los resultados usando un *forest plot* (diagrama de bosque; sección 3.4)
 - 1.4 Mostrar la distribución de los resultados en un *funnel plot* (diagrama de embudo; sección 3.5) y estimar el sesgo de estudios pequeños (sección 3.5.3)

²²Para esto existen varias guías, e incluso la *Declaración PRISMA* que está diseñada para abordar los problemas en la publicación de revisiones sistemáticas y meta-análisis (para información en español, ver e.g., Hutton et al., 2016; Page et al., 2021).

- A. Formular la pregunta de investigación clara, con base en teoría
- B. Identificar la bibliografía pertinente
- C. Extraer y consolidar los datos de los estudios (en el caso de correlaciones, cuando menos r y n)

2. Estimar si hay evidencia de sesgo de publicación (sección 3.6)

2.1 Puedes usar el método *trim and fill* (recorte y relleno; sección 3.6.1) o, lo que es mejor, hacer una estimación del modelo de función de peso (sección 3.6.2)

Adicionalmente, podrías:

3. Estimar el poder estadístico del meta-análisis y los estudios meta-analizados (sección 3.7)

5.2. Conclusión

Es fundamental tener en cuenta que el meta-análisis, aunque es una herramienta poderosa, no es una solución mágica para resolver los problemas de investigación. La calidad de los estudios incluidos, la heterogeneidad de los datos y otras limitaciones pueden influir en los resultados y su interpretación. En última instancia, el éxito de un meta-análisis dependerá del rigor y la transparencia del investigador o investigadora en su implementación y en la interpretación de sus resultados.

Sin embargo, si tienes en cuenta y de manera rigurosa las recomendaciones de esta guía, tendrás una base sólida para hacer y comunicar tu meta-análisis de manera transparente, sin importar los resultados obtenidos.

Lamentablemente, en el mundo hispano, especialmente en Latinoamérica, todavía se presentan importantes rezagos en la producción científica, que son resultado en parte del insuficiente apoyo gubernamental y de un sistema global que desfavorece a los países en vías de desarrollo. Sin embargo, también son producto de una falta de confianza en las capacidades propias y de ciertos déficits reales en educación.

Mi objetivo con esta guía es inspirar a investigadores hispanohablantes, ya sean establecidos o en formación, y de diversas disciplinas, a conocer los fundamentos del meta-análisis y su implementación de forma transparente. El fortalecimiento de nuestras habilidades analíticas es un paso fundamental para mejorar nuestra formación y contribuir al conocimiento global.

Agradecimientos

Quiero agradecer especialmente el apoyo de mi gran amigo y colega [Andrés Castellanos-Chacón](#) por su dedicada revisión de este documento.

Referencias

- Bache, S. M., & Wickham, H. (2022). *magrittr: A Forward-Pipe Operator for R* [Paquete de R versión 2.0.3]. <https://CRAN.R-project.org/package=magrittr>
- Baker, M. (2016). 1,500 Scientists Lift the Lid on Reproducibility. *Nature*, 533(7604), 452. <https://doi.org/10.1038/533452a>
- Balduzzi, S., Rücker, G., & Schwarzer, G. (2019). How to perform a meta-analysis with R: a practical tutorial. *Evidence-Based Mental Health*, (22), 153-160. <https://doi.org/10.1136/ebmental-2019-300117>
- Belsley, D. A., Kuh, E., & Welsch, R. E. (1980, junio). *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. John Wiley & Sons, Inc. <https://doi.org/10.1002/0471725153>
- Borenstein, M., Hedges, L. V., Higgins, J. P. T., & Rothstein, H. R. (2009). Identifying and Quantifying Heterogeneity. En *Introduction to Meta-Analysis* (pp. 107-125). Wiley. <https://doi.org/10.1002/9780470743386.ch16>
- Coburn, K. M., & Vevea, J. L. (2015). Publication Bias as a Function of Study Characteristics. *Psychological Methods*, 20(3), 310-330. <https://doi.org/10.1037/met0000046>
- Coburn, K. M., & Vevea, J. L. (2019). *weightr: Estimating Weight-Function Models for Publication Bias* [Paquete de R versión 2.0.2]. <https://CRAN.R-project.org/package=weightr>
- Cook, R. D. (1977). Detection of Influential Observation in Linear Regression. *Technometrics*, 19(1), 15-18. <https://doi.org/10.2307/1268249>
- Cook, R. D., & Weisberg, S. (1982). *Residuals and Influence in Regression*. New York: Chapman and Hall. <https://hdl.handle.net/11299/37076>
- Copas, J., & Shi, J. Q. (2000). Meta-Analysis, Funnel Plots and Sensitivity Analysis. *Biostatistics*, 1(3), 247-262. <https://doi.org/10.1093/biostatistics/1.3.247>
- Copas, J., & Shi, J. Q. (2001). A sensitivity analysis for publication bias in systematic reviews. *Statistical Methods in Medical Research*, 10(4), 251-265. <https://doi.org/10.1177/096228020101000402>
- Datademia. (2018, 28 de abril). *Aprende Ggplot2 En R* [Video]. YouTube. https://youtu.be/BUzTAr_QqKs
- Duval, S. (2005). The Trim and Fill Method. En H. R. Rothstein, A. J. Sutton & M. Borenstein (Eds.), *Publication Bias*

- in *Meta-Analysis* (pp. 127-144). John Wiley & Sons. <https://doi.org/10.1002/0470870168.ch8>
- Duval, S., & Tweedie, R. (2000a). A Nonparametric “Trim and Fill” Method of Accounting for Publication Bias in Meta-Analysis. *Journal of the American Statistical Association*, 95(449), 89-98. <https://doi.org/10.1080/01621459.2000.10473905>
- Duval, S., & Tweedie, R. (2000b). Trim and Fill: A Simple Funnel-Plot-Based Method of Testing and Adjusting for Publication Bias in Meta-Analysis. *Biometrics*, 56(2), 455-463. <https://doi.org/10.1111/j.0006-341X.2000.00455.x>
- Egger, M., Smith, G. D., Schneider, M., & Minder, C. (1997). Bias in Meta-Analysis Detected by a Simple, Graphical Test. *BMJ*, 315(7109), 629-634. <https://doi.org/10.1136/bmj.315.7109.629>
- Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 226-231. <http://cs.fit.edu/~pkc/classes/ml-i/internet/papers/ester96kdd-dbscan.pdf>
- Fisher, R. A. (1915). Frequency Distribution of the Values of the Correlation Coefficient in Samples from an Indefinitely Large Population. *Biometrika*, 10(4), 507-521. <https://doi.org/10.2307/2331838>
- Fraley, C., & Raftery, A. E. (2002). Model-Based Clustering, Discriminant Analysis, and Density Estimation. *Journal of the American Statistical Association*, 97(458), 611-631. <https://doi.org/10.1198/016214502760047131>
- Gil Bellosta, C. J. (2023). *R para profesionales de los datos: Una introducción*. https://datanalytics.com/libro_r/
- Gómez-Restrepo, C. (2008). Acerca del sesgo de publicación en literatura sobre antidepresivos. *Revista Colombiana de Psiquiatría*, 37(1), 8-10. http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0034-74502008000100001
- Harrer, M., Cuijpers, P., A. F. T., & Ebert, D. D. (2021). *Doing Meta-Analysis With R: A Hands-On Guide*. Chapman & Hall/CRC Press. https://bookdown.org/MathiasHarrer/Doing_Meta_Analysis_in_R/
- Harrer, M., Cuijpers, P., Furukawa, T., & Ebert, D. D. (2019). *dmetar: Companion R Package For The Guide 'Doing Meta-Analysis in R'* [Paquete de R versión 0.0.9000]. <http://dmetar.protectlab.org/>
- Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1), 100-108. <https://doi.org/10.2307/2346830>
- Henry, N. W., Cohen, J., & Cohen, P. (2003). *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*. (3rd). Lawrence Erlbaum Associates. <https://doi.org/10.4324/9780203774441>
- Hidalgo-Fuentes, S. (2022). Uso problemático de Internet y procrastinación en estudiantes: un meta-análisis. *European Journal of Education and Psychology*, 15(1), 1-18. <https://doi.org/10.32457/ejep.v15i1.1815>
- Hutton, B., Catalá-López, F., & Moher, D. (2016). La extensión de la declaración PRISMA para revisiones sistemáticas que incorporan metaanálisis en red: PRISMA-NMA. *Medicina Clínica*, 147(6), 262-266. <https://doi.org/10.1016/j.medcli.2016.02.025>
- John, O. P., & Srivastava, S. (1999). The Big Five Trait Taxonomy: History, Measurement, and Theoretical Perspectives. En L. A. Pervin & O. P. John (Eds.), *Handbook of Personality: Theory and Research*, 2nd Ed (pp. 102-138). Guilford Press. http://jenni.uchicago.edu/econ-psych-traits/John_Srivastava_1995_big5.pdf
- Klein, R. A., Vianello, M., Hasselman, F., Adams, B. G., Adams, R. B., Alper, S., Aveyard, M., Axt, J. R., Babalola, M. T., Bahník, Š., Batra, R., Berkics, M., Bernstein, M. J., Berry, D. R., Bialobrzeska, O., Binan, E. D., Boocian, K., Brandt, M. J., Busching, R., ... Nosek, B. A.
- (2018). Many Labs 2: Investigating Variation in Replicability Across Samples and Settings. *Advances in Methods and Practices in Psychological Science*, 1(4), 443-490. <https://doi.org/10.1177/2515245918810225>
- Kossmeier, M., Tran, U. S., & Voracek, M. (2020). *metaviz: Forest Plots, Funnel Plots, and Visual Funnel Plot Inference for Meta-Analysis* [Paquete de R versión 0.3.1]. <https://CRAN.R-project.org/package=metaviz>
- Lakens, D. (2022). *Improving Your Statistical Inferences*. https://lakens.github.io/statistical_inferences/
- Leongómez, J. D. (2020a). Análisis de Poder Estadístico y Cálculo de Tamaño de Muestra En R: Guía Práctica. *Zenodo*. <https://doi.org/10.5281/zenodo.3988776>
- Leongómez, J. D. (2020b, 18 de septiembre). *Poder estadístico y tamaño de muestra en R* [Serie de videos]. YouTube. <https://youtube.com/playlist?list=PLHk7UNt35ccVdyHqnQ6oXVYAA6JBNFrE1x>
- Leongómez, J. D. (2021, 9 de mayo). *Hacer meta-análisis en jupyter es muy fácil* [Video]. YouTube. https://youtu.be/ntBbkOn9D_o
- López Peñaranda, A. (2019, 29 de diciembre). *Tutorial de ggplot2*. RPubs. Consultado el 14 de noviembre de 2022, desde <https://rpubs.com/anlope10/562981>
- Manterola, C., Quiroz, G., Salazar, P., & García, N. (2019). Metodología de los tipos y diseños de estudio más frecuentemente utilizados en investigación clínica. *Revista Médica Clínica Las Condes*, 30(1), 36-49. <https://doi.org/https://doi.org/10.1016/j.rmcl.2018.11.005>
- Molloy, G., O'Carroll, R., & Ferguson, E. (2014). Conscientiousness and Medication Adherence: A Meta-analysis. *Annals of Behavioral Medicine*, 47(1), 92-101. <https://doi.org/10.1007/s12160-013-9524-4>
- Olkkin, I., Dahabreh, I. J., & Trikalinos, T. A. (2012). GOSH – a Graphical Display of Study Heterogeneity. *Research Synthesis Methods*, 3(3), 214-223. <https://doi.org/10.1002/rsm.1053>
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., ... Alonso-Fernández, S. (2021). Declaración PRISMA 2020: una guía actualizada para la publicación de revisiones sistemáticas. *Revista Española de Cardiología*, 74(9), 790-799. <https://doi.org/10.1016/j.recesp.2021.06.016>
- Quintana, D. S. (2020, 15 de junio). *Oh My GOSH: Calculating All Possible Meta-Analysis Study Combinations* [Blog]. Hormones, brain, and behavior. Consultado el 17 de marzo de 2023, desde <https://www.dsquintana.blog/combinatorial-meta-analysis/>
- Quintana, D. S. (2022). *metameta: A Suite of Tools to Re-evaluate Published Meta-analyses* [Paquete de R versión 0.1.1]. <https://github.com/dsquintana/metameta>
- Quintana, D. S. (2023). A guide for calculating study-Level statistical power for meta-analyses. *Advances in Methods and Practices in Psychological Science*, 6(1), 1-18. <https://doi.org/10.1177/25152459221147260>
- Revelle, W. (2023). *psych: Procedures for Psychological, Psychometric, and Personality Research* [Paquete de R versión 2.3.3]. Northwestern University. Evanston, Illinois. <https://CRAN.R-project.org/package=psych>
- Salsman, J. M., Pustejovsky, J. E., Jim, H. S. L., Munoz, A. R., Merluzzi, T. V., George, L., Park, C. L., Danhauer, S. C., Sherman, A. C., Snyder, M. A., & Fitchett, G. (2015). A Meta-Analytic Approach to Examining the Correlation between Religion/Spirituality and Mental Health in Cancer. *Cancer*, 121 (21), 3769-3778. <https://doi.org/10.1002/cncr.29350>
- Sánchez-Bruno, A., & Borges del Rosal, Á. (2005). Transformación Z de Fisher para la determinación de intervalos de confianza del coeficiente de correlación de Pearson. *Psi-*

- cothema*, 17(1), 148-153. <https://www.redalyc.org/articulo.oa?id=72717124>
- Schubert, E., Sander, J., Ester, M., Kriegel, H. P., & Xu, X. (2017). DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Transactions on Database Systems*, 42(3), 19:1-19:21. <https://doi.org/10.1145/3068335>
- Schwarzer, G., Carpenter, J., & Rücker, G. (2010). Empirical Evaluation Suggests Copas Selection Model Preferable to Trim-and-Fill Method for Selection Bias in Meta-Analysis. *Journal of Clinical Epidemiology*, 63(3), 282-288. <https://doi.org/10.1016/j.jclinepi.2009.05.008>
- Schwarzer, G., Carpenter, J. R., & Rücker, G. (2015a). *Meta-Analysis with R*. Springer. https://doi.org/10.1007/978-78-3-319-21416-0_0
- Schwarzer, G., Carpenter, J. R., & Rücker, G. (2015b). Small-Study Effects in Meta-Analysis. En *Meta-Analysis with R* (pp. 107-141). Springer. https://doi.org/10.1007/978-3-319-21416-0_5
- Schwarzer, G., Carpenter, J. R., & Rücker, G. (2023). *metasens: Statistical Methods for Sensitivity Analysis in Meta-Analysis* [Paquete de R versión 1.5-2]. <https://CRAN.R-project.org/package=metasens>
- Sedgwick, P. (2015). Meta-Analyses: What Is Heterogeneity? *BMJ*, 350, h1435. <https://doi.org/10.1136/bmj.h1435>
- Shadish, W. R., & Haddock, C. K. (2009). Combining Estimates of Effect Size. En H. M. Cooper, L. V. Hedges & J. C. Valentine (Eds.), *The Handbook of Research Synthesis and Meta-Analysis* (2nd ed., pp. 257-277). Russell Sage Foundation.
- Shi, L., & Lin, L. (2019). The Trim-and-Fill Method for Publication Bias: Practical Guidelines and Recommendations Based on a Large Database of Meta-Analyses. *Medicine*, 98(23), e15987. <https://doi.org/10.1097/MD.00000000000015987>
- Shi, L., Liu, J., Fonseca, V., Walker, P., Kalsekar, A., & Pawaskar, M. (2010). Correlation between Adherence Rates Measured by MEMS and Self-Reported Questionnaires: A Meta-Analysis. *Health and Quality of Life Outcomes*, 8(1), 99. <https://doi.org/10.1186/1477-7525-8-99>
- Simonsohn, U., Nelson, L. D., & Simmons, J. P. (2014). P-Curve and Effect Size: Correcting for Publication Bias Using Only Significant Results. *Perspectives on Psychological Science*, 9(6), 666-681. <https://doi.org/10.1177/1745691614553988>
- Simonsohn, U., Nelson, L. D., & Simmons, J. P. (2019). P-Curve Won't Do Your Laundry, but It Will Distinguish Replicable from Non-Replicable Findings in Observational Research: Comment on Bruns & Ioannidis (2016). *PLOS ONE*, 14(3), e0213454. <https://doi.org/10.1371/journal.pone.0213454>
- Sterne, J. A., Gavaghan, D., & Egger, M. (2000). Publication and Related Bias in Meta-Analysis. *Journal of Clinical Epidemiology*, 53(11), 1119-1129. [https://doi.org/10.1016/S0895-4356\(00\)00242-0](https://doi.org/10.1016/S0895-4356(00)00242-0)
- Sterne, J. A., Sutton, A. J., Ioannidis, J. P. A., Terrin, N., Jones, D. R., Lau, J., Carpenter, J., Rucker, G., Harbord, R. M., Schmid, C. H., Tetzlaff, J., Deeks, J. J., Peters, J., Macaskill, P., Schwarzer, G., Duval, S., Altman, D. G., Moher, D., & Higgins, J. P. T. (2011). Recommendations for Examining and Interpreting Funnel Plot Asymmetry in Meta-Analyses of Randomised Controlled Trials. *BMJ*, 343(1), d4002. <https://doi.org/10.1136/bmj.d4002>
- Vevea, J. L., & Hedges, L. V. (1995). A General Linear Model for Estimating Effect Size in the Presence of Publication Bias. *Psychometrika*, 60(3), 419-435. <https://doi.org/10.1007/BF02294384>
- Vevea, J. L., & Woods, C. M. (2005). Publication Bias in Research Synthesis: Sensitivity Analysis Using a Priori Weight Functions. *Psychological Methods*, 10(4), 428-443. <https://doi.org/10.1037/1082-989X.10.4.428>
- Viechtbauer, W. (2010). Conducting Meta-Analyses in R with the metafor Package. *Journal of Statistical Software*, 36(3). <https://doi.org/10.18637/jss.v036.i03>
- Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer. <https://ggplot2.tidyverse.org>
- Wickham, H. (2023). *forcats: Tools for Working with Categorical Variables (Factors)* [Paquete de R versión 1.0.0]. <https://CRAN.R-project.org/package=forcats>
- Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., ... Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43), 1686. <https://doi.org/10.21105/joss.01686>
- Wickham, H., François, R., Henry, L., & Müller, K. (2023). *dplyr: A Grammar of Data Manipulation* [Paquete de R versión 1.1.1]. <https://CRAN.R-project.org/package=dplyr>

APÉNDICES

Apéndice A. Alternativas a `metafor`

A lo largo de esta guía he usado una ruta para hacer meta-análisis que parte de objetos generados con el paquete `metafor`. Sin embargo, existen rutas alternativas para realizar meta-análisis en R. El libro *Doing meta-analysis with R: a hands-on guide* (Harrer et al., 2021) se acompaña del paquete `dmetar` (Harrer et al., 2019), que contiene y explica opciones para hacer meta-análisis tanto a partir de `metafor`, como a partir de `meta` (Balduzzi et al., 2019; Schwarzer et al., 2015a).

Los objetos generados por `meta` al realizar un meta-análisis, permiten usar funciones de `dmetar` para hacer otros análisis como *risk of bias* (riesgo de sesgo), inferencia multi-modelo, detección de *outliers* (valores atípicos), y *p-curve* o curva de valores *p* (e.g., Simonsohn et al., 2014, 2019), y permiten hacer gráficos distintos.

Para una guía resumida y concreta de estas opciones (en inglés), recomiendo ver el sitio web del paquete `dmetar`, y en especial la página [Get Started](#).

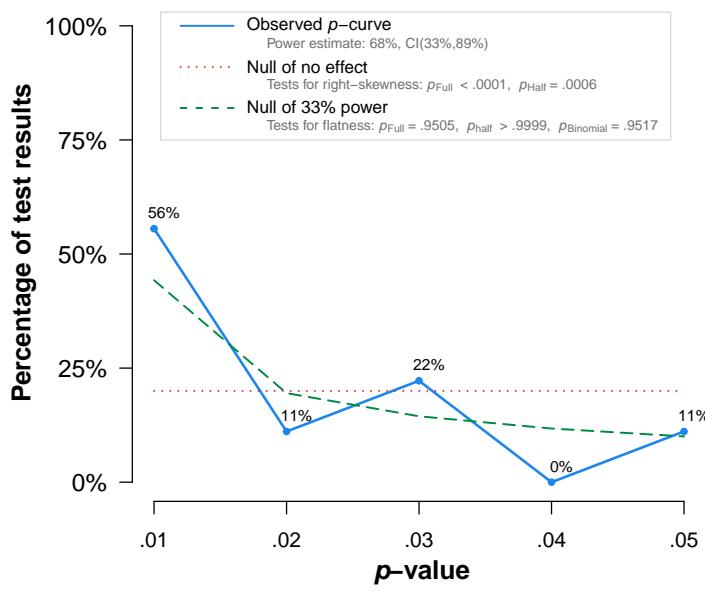
A1. Ejemplo: curva de valores p (p -curve)

Como ejemplo puntual, la curva de valores p (en inglés p -curve) de nuestro meta-análisis se podría hacer tras ajustar el modelo con la función `metacor` (que hace meta-análisis de correlaciones) del paquete `meta` (Balduzzi et al., 2019; Schwarzer et al., 2015a). Para este ejemplo asignaré el resultado a un objeto que llamaré `res.meta`.

```
library(meta)
res.meta <- metacor(cor = yi, n = ni,
                      studlab = paste(dat$authors, dat$year, sep = ", "),
                      data = dat, sm = "ZCOR")
```

El objeto resultante (`res.meta`) puede usarse directamente como argumento a la función `pcurve` del paquete `dmetar`, para obtener una gráfica con la curva de valores p (p -curve).

```
library(dmetar)
pcurve(res.meta)
```



Note: The observed p -curve includes 9 statistically significant ($p < .05$) results, of which 8 are $p < .025$.
There were 7 additional results entered but excluded from p -curve because they were $p > .05$.

Figura A1. Curva de valores p (p -curve) creada con `dmetar`. Para entender una curva de valores p , su relevancia y su interpretación, sugiero leer Simonsohn et al. (2019, 2014), o ver el sitio web del paquete `dmetar`: <http://dmetar.protectlab.org/>.

La función `pcurve` produce varios elementos relacionados la curva de valores p , incluyendo la Figura A1 que resume este análisis.

Es importante tener en cuenta que, al menos por ahora, el paquete `dmetar` debe instalarse desde GitHub tal como `metameta` (en este caso, con el comando `devtools::install_github("MathiasHarrer/dmetar")`).

Apéndice B. Citas y referencias de paquetes de R

Por supuesto, los paquetes de R que usemos deben ser citados. Una manera fácil de encontrar la cita que los autores de un paquete quieren que usemos, es la función `citation` en R. Simplemente debemos usar esta función, agregando como argumento el nombre del paquete que queremos citar entre comillas. Esto nos dará la referencia en un formato estándar, así como como en un formato BibTex que puede ser usado en documentos L^AT_EX, o por muchos gestores de referencia como `Zotero`. Alternativamente, el formato formato BibTex nos permite saber los campos como autores, título y demás, si vamos a crear las citas y referencias manualmente, o incluso crear referencias en diversos estilos usando, por ejemplo, `Online Bibtex Converter`.

Como ejemplo, en esta guía usé `dplyr` (Wickham et al., 2023) para manipular los datos. Usando la función `citation`:

```
citation("dplyr")
```

Obtengo la siguiente información sobre ese paquete:

Consola de R: Output 19

To cite package 'dplyr' in publications use:

```
Wickham H, François R, Henry L, Müller K, Vaughan D (2023). _dplyr: A Grammar of Data Manipulation_. R package version 1.1.1, <https://CRAN.R-project.org/package=dplyr>.
```

A BibTeX entry for LaTeX users is

```
@Manual{,
  title = {dplyr: A Grammar of Data Manipulation},
  author = {Hadley Wickham and Romain François and Lionel Henry and Kirill Müller and Davis Vaughan},
  year = {2023},
  note = {R package version 1.1.1},
  url = {https://CRAN.R-project.org/package=dplyr},
```

Apéndice C. Paquetes de R usados para crear este documento

R version 4.2.3 (2023-03-15 ucrt)

Platform: x86_64-w64-mingw32/x64 (64-bit)

attached base packages: *stats, graphics, grDevices, utils, datasets, methods and base*

other attached packages: *meta(v.6.2-1), dmetar(v.0.0.9000), metameta(v.0.1.1), metaviz(v.0.3.1), tidyquant(v.1.0.6), quantmod(v.0.4.20), TTR(v.0.24.3), PerformanceAnalytics(v.2.0.4), xts(v.0.13.0), zoo(v.1.8-11), psych(v.2.3.3), scales(v.1.2.1), kableExtra(v.1.3.4), ggpubr(v.0.6.0), lubridate(v.1.9.2), forcats(v.1.0.0), stringr(v.1.5.0), dplyr(v.1.1.1), purrr(v.1.0.1), readr(v.2.1.4), tidyverse(v.1.3.0), tibble(v.3.2.1), ggplot2(v.3.4.1), tidyverse(v.2.0.0), metafor(v.4.0-0), numDeriv(v.2016.8-1.1), metadat(v.1.2-0), Matrix(v.1.5-3), robumeta(v.2.1) and knitr(v.1.42)*

loaded via a namespace (and not attached): *minqa(v.1.2.5), colorspace(v.2.1-0), ggsignif(v.0.6.4), class(v.7.3-21), modeltools(v.0.2-23), mclust(v.6.0.0), rstudioapi(v.0.14), farver(v.2.1.1), ggrepel(v.0.9.3), flexmix(v.2.3-19), fansi(v.1.0.4), mathjaxr(v.1.6-0), xml2(v.1.3.3), splines(v.4.2.3), mnormt(v.2.1.1), robustbase(v.0.95-0), jsonlite(v.1.8.4), nloptr(v.2.0.3), broom(v.1.0.4), kernlab(v.0.9-32), cluster(v.2.1.4), compiler(v.4.2.3), httr(v.1.4.5), backports(v.1.4.1), fastmap(v.1.1.1), cli(v.3.6.1), htmltools(v.0.5.5), tools(v.4.2.3), gtable(v.0.3.3), glue(v.1.6.2), ggthemes(v.4.2.4), Rcpp(v.1.0.10), carData(v.3.0-5), vctrs(v.0.6.1), svglite(v.2.1.1), nlme(v.3.1-162), fpc(v.2.2-10), xfun(v.0.38), lme4(v.1.1-32), rvest(v.1.0.3), timechange(v.0.2.0), CompQuadForm(v.1.4.3), lifecycle(v.1.0.3), rstatix(v.0.7.2), DEoptimR(v.1.0-11), MASS(v.7.3-58.2), hms(v.1.1.3), parallel(v.4.2.3), RColorBrewer(v.1.1-3), yaml(v.2.3.7), curl(v.5.0.0), gridExtra(v.2.3), pander(v.0.6.5), MuMIn(v.1.47.5), stringi(v.1.7.12), highr(v.0.10), poibin(v.1.5), boot(v.1.3-28.1), rlang(v.1.1.0), pkgconfig(v.2.0.3), systemfonts(v.1.0.4), prabclus(v.2.3-2), evaluate(v.0.20), lattice(v.0.20-45), labeling(v.0.4.2), cowplot(v.1.1.1), tidyselect(v.1.2.0), magrittr(v.2.0.3), bookdown(v.0.33), R6(v.2.5.1), generics(v.0.1.3), pillar(v.1.9.0), withr(v.2.5.0), mgcv(v.1.8-42), abind(v.1.4-5), nnet(v.7.3-18), car(v.3.1-1), Quandl(v.2.11.0), utf8(v.1.2.3), tzdb(v.0.3.0), rmarkdown(v.2.21), grid(v.4.2.3), netmeta(v.2.8-1), diptest(v.0.76-0), digest(v.0.6.31), webshot(v.0.5.4), stats4(v.4.2.3), munsell(v.0.5.0), viridisLite(v.0.4.1), magic(v.1.6-1) and quadprog(v.1.5-8)*