

Manual de Estadística Bayesiana

con R y Stan



Alfredo Sánchez Alberca
asalber@ceu.es
<https://aprendeconalf.es>

Índice de contenidos

1	Manual de Estadística Bayesiana	3
1.1	Presentación	3
1.2	Contenido del manual	3
1.3	Requisitos	4
1.4	Licencia	4
2	Repaso de probabilidad	5
2.1	Frecuencias muestrales	5
2.2	Distribuciones de frecuencias	5
2.3	Distribuciones de frecuencias bidimensionales	10
2.3.1	Distribuciones de frecuencias marginales	11
2.4	Distribuciones de frecuencias condicionales	13
2.5	Experimentos aleatorios	15
2.6	Espacio muestral	16
2.6.1	Diagrama de árbol	16
2.7	Sucesos aleatorios	17
2.8	Operaciones entre sucesos	18
2.8.1	Unión de sucesos	18
2.8.2	Intersección de sucesos	18
2.8.3	Contrario de un suceso	19
2.8.4	Diferencia de sucesos	20
2.9	Definiciones de probabilidad	21
2.9.1	Definición clásica de probabilidad	21
2.9.2	Definición frecuentista de probabilidad	22
2.9.3	Definición subjetiva de probabilidad	23
2.9.4	Definición axiomática de probabilidad	24
2.9.5	Interpretación de la probabilidad	24
2.10	Probabilidad condicionada	24
2.10.1	Probabilidad del suceso intersección	26
2.11	Independencia de sucesos	26
2.11.1	Árboles de probabilidad con variables dependientes	27
2.11.2	Árboles de probabilidad con variables independientes	27
2.12	Variables aleatorias	28
2.12.1	Tipos de variables aleatorias	29
2.13	Variables aleatorias discretas	29
2.13.1	Estadísticos de una variable aleatoria discreta	29

2.13.2	Distribución uniforme discreta	30
2.13.3	Distribución binomial	31
2.13.4	Distribución geométrica	34
2.13.5	Distribución binomial negativa	37
2.13.6	Distribución de Poisson	40
2.13.7	Resumen de distribuciones discretas	42
2.14	Variables aleatorias continuas	43
2.14.1	Estadísticos de una variable aleatoria continua	44
2.14.2	Distribución uniforme continua	44
2.14.3	Distribución beta	45
2.14.4	Distribución normal	48
2.14.5	Distribución exponencial	51
2.14.6	Distribución gamma	53
2.14.7	Distribución gamma inversa	56
2.14.8	Resumen de distribuciones continuas	58
2.15	Muestreo	60
2.15.1	Muestreo en variables aleatorias discretas	61
2.15.2	Muestreo en el modelo de distribución binomial	63
2.15.3	Muestreo de la distribución geométrica	66
2.15.4	Muestreo de la distribución de Poisson	67
2.15.5	Muestreo de la distribución uniforme continua	68
2.15.6	Muestreo de la distribución beta	69
2.15.7	Muestreo de la distribución normal	70
2.15.8	Muestreo de la distribución exponencial	71
2.15.9	Muestreo de la distribución gamma	73
3	Fundamentos de la estadística bayesiana	83
3.1	¿Por qué pensar en bayesiano?	83
3.2	Probabilidad condicional	84
3.2.1	El jardín de los senderos que se bifurcan	85
3.3	El teorema de Bayes	85
3.3.1	Los tres ingredientes	86
3.3.2	La posterior como compromiso	87
3.4	Bayesiano frente a frecuentista	88
3.5	Aplicación: pruebas diagnósticas	89
3.5.1	Aplicando Bayes al cribado	90
3.5.2	Frecuencias naturales	90
3.5.3	El papel crucial de la prevalencia	91
3.5.4	Actualización secuencial	91
3.6	Priors: el ingrediente bayesiano	92
3.6.1	Simulación predictiva a priori	93
3.7	Ideas clave	94
3.8	Lecturas recomendadas	94

4	Regla de Bayes	95
4.1	Sistema completo de sucesos	95
4.2	Teorema de la probabilidad total	95
4.3	Teorema de Bayes	97
4.4	Regla de Bayes	99
4.5	Función de verosimilitud	99
4.5.1	Interpretación de la función de verosimilitud	100
4.6	Probabilidad a posteriori	100
4.7	Simulación de la regla de Bayes	101
4.7.1	Muestreo de la distribución a priori	102
4.7.2	Muestreo de la distribución conjunta	103
4.7.3	Muestreo de la distribución a posteriori	103
4.8	Distribuciones conjugadas	109
4.8.1	Distribución conjugada Beta-Binomial	109
4.8.2	Distribución conjugada Gamma-Poisson	110
4.8.3	Distribución conjugada Normal-Normal	111
4.9	Modelos bayesianos	111
4.9.1	Elementos de un modelo bayesiano	112
4.9.2	Distribuciones a priori discretas y continuas	112
4.9.3	Distribuciones de los datos condicionadas por las hipótesis	112
4.9.4	Tipos de modelos bayesianos	113
4.10	Modelo beta-binomial	114
4.11	Simulación de la distribución a posteriori	118
4.11.1	Simulación mediante muestreo de rejilla	118
4.11.2	Modelo gamma-Poisson	122
4.12	Independencia del orden de obtención de los datos	129
4.13	Modelo Normal-Normal	130
4.14	Explosión combinatoria del muestreo de rejilla	133
4.15	Métodos de Monte Carlo	134
4.16	El lenguaje Stan	134
4.16.1	Estructura de un programa en Stan	134
4.16.2	El paquete rstan	135
4.17	Análisis de las cadenas de Markov	139
4.17.1	Análisis de la traza	140
4.17.2	Análisis de la autocorrelación	141
4.17.3	Análisis de la eficiencia	142
4.17.4	Análisis de la consistencia	143
4.18	Dependencia de los resultados de la distribución a priori	146
4.19	Distribuciones a priori poco informativas	146
4.20	Tipos de distribuciones a priori	148
4.20.1	Distribuciones a priori no informativas	148
4.20.2	Distribuciones a priori poco informativas	149
4.20.3	Distribuciones a priori informativas	149
4.21	Distribuciones a priori recomendadas	150

4.22	Distribuciones a priori en epidemiología	150
4.23	Referencias del capítulo	153
5	Inferencia bayesiana	154
5.1	Estimación de parámetros	154
5.1.1	Estimación frecuentista	154
5.1.2	Interpretación de un intervalo de confianza	155
5.1.3	Estimación bayesiana	156
5.1.4	Intervalo de credibilidad	156
5.2	Contrastes de hipótesis	161
5.2.1	Hipótesis estadística	161
5.2.2	Contrastes de hipótesis paramétricos en estadística frecuentista	161
5.2.3	Errores en la toma de decisiones	162
5.2.4	Aceptación o rechazo de la hipótesis nula	162
5.2.5	Interpretación de la decisión	163
5.2.6	Hipótesis bayesianas	163
5.2.7	Contrastes de hipótesis bayesianos	163
5.2.8	Odds a posteriori	165
5.2.9	Factor de Bayes	166
5.3	Predicciones	167
5.3.1	Predicciones en estadística frecuentista	167
5.3.2	Predicciones en estadística bayesiana	168
5.3.3	Distribución predictiva posterior	168
5.3.4	Simulación de la distribución predictiva posterior	168
5.3.5	Resumen de la distribución predictiva posterior	170
5.4	Referencias del capítulo	172
6	Regresión bayesiana	173
6.1	Modelos de regresión	173
6.1.1	Diagrama de dispersión	173
6.1.2	Tipos de modelos de regresión simple	175
6.2	Regresión lineal simple	176
6.2.1	Regresión lineal en estadística frecuentista	176
6.2.2	Ajuste de modelos de regresión con R	177
6.3	Regresión lineal en estadística bayesiana	179
6.3.1	Supuestos del modelo de regresión lineal simple	179
6.3.2	Distribuciones a priori de los parámetros del modelo	179
6.3.3	Modelo de regresión lineal simple con distribuciones a priori	180
6.3.4	Validación de las distribuciones a priori	180
6.4	Simulación de las distribuciones a posteriori	182
6.5	Simulaciones de las distribuciones a posteriori con el paquete <code>rstanarm</code>	184
6.6	Distribuciones a priori por defecto	187
6.7	Intervalos de credibilidad de los parámetros	189
6.8	Dibujo de las rectas de regresión a posteriori	190

6.9	Contraste de hipótesis sobre los coeficientes de regresión	191
6.10	Predicciones con modelos de regresión	193
6.10.1	Predicción de la media de la variable dependiente	193
6.10.2	Predicción de un nuevo valor de la variable dependiente	195
6.11	Evaluación de modelos de regresión	200
6.11.1	Métricas de evaluación	201
6.12	Evaluación del modelo de regresión en datos nuevos	204
6.12.1	Validación cruzada	204
6.13	Regresión lineal múltiple	206
6.13.1	Evaluación de las distribuciones a priori	207
6.13.2	Ajuste del modelo de regresión lineal múltiple	209
6.13.3	Intervalos de credibilidad de los parámetros	210
6.13.4	Dibujo de las rectas de regresión a posteriori	211
6.13.5	Predicciones del modelo de regresión múltiple	212
6.13.6	Distribuciones predictivas del modelo de regresión múltiple	213
6.13.7	Evaluación del modelo de regresión múltiple	214
6.14	Comparación de modelos de regresión	215
6.15	Densidad logarítmica predictiva esperada (ELPD)	215
6.16	Modelos de regresión con interacciones	218
6.17	Regresión logística	221
6.17.1	Variables dependientes categóricas	221
6.17.2	El modelo de regresión logística	221
6.17.3	La función logística	222
6.17.4	Interpretación de los coeficientes	223
6.17.5	Construcción del modelo de regresión logística	223
6.17.6	Distribuciones a priori de los parámetros	223
6.17.7	Ajuste del modelo de regresión logística bayesiana	225
6.17.8	Validación de las distribuciones a priori	226
6.17.9	Ajuste de las distribuciones a posteriori	227
6.17.10	Intervalos de credibilidad	229
6.17.11	Razón de odds	229
6.17.12	La curva logística a posteriori	230
6.17.13	Estimación de la prevalencia	231
6.17.14	Predicción de la probabilidad para nuevos casos	232
6.17.15	Clasificación de nuevos casos	233
6.17.16	Evaluación del modelo: matriz de confusión	234
6.17.17	Ajuste del umbral de clasificación	235
6.17.18	Evaluación mediante validación cruzada	236
6.18	Regresión logística múltiple	237
6.19	Referencias del capítulo	238

1 Manual de Estadística Bayesiana

Con R y Stan

1.1. Presentación

Este manual introduce la estadística bayesiana desde sus fundamentos hasta los modelos de regresión, con ejemplos aplicados a datos de salud y con R como herramienta de cálculo. Está pensado para estudiantes y profesionales de ciencias de la salud y de otras disciplinas que quieran entender y aplicar el razonamiento bayesiano.

El objetivo no es memorizar fórmulas, sino comprender cómo se actualiza el conocimiento sobre un fenómeno a partir de los datos, y saber interpretar las distribuciones de probabilidad que resultan de ello.

1.2. Contenido del manual

- **Capítulo 1. Repaso de probabilidad:** probabilidad, frecuencias, distribuciones y simulación de muestras aleatorias.
- **Capítulo 2. Fundamentos de la estadística bayesiana:** por qué pensar en bayesiano, probabilidad condicionada, el teorema de Bayes, priors, verosimilitud y posterior, y pruebas diagnósticas.
- **Capítulo 3. Regla de Bayes:** distribuciones a priori, funciones de verosimilitud y distribuciones a posteriori. Distribuciones conjugadas.
- **Capítulo 4. Inferencia bayesiana:** estimación de parámetros, intervalos de credibilidad, contraste de hipótesis y predicciones.
- **Capítulo 5. Regresión bayesiana:** modelos de regresión lineales y no lineales, evaluación de modelos y predicciones.

1.3. Requisitos

Los ejemplos se han escrito con R. Para reproducir el código de los capítulos de regresión e inferencia son necesarios, entre otros, los paquetes `tidyverse`, `rstanarm`, `tidybayes`, `bayesplot` y `loo`.

1.4. Licencia

Esta obra está bajo una licencia [Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional \(CC BY-NC-SA 4.0\)](#).

Con esta licencia eres libre de:

- Compartir: copiar y redistribuir el material en cualquier medio o formato.
- Adaptar: remezclar, transformar y crear a partir del material.

Bajo las siguientes condiciones:

- **Reconocimiento.** Debes dar crédito de manera adecuada, proporcionar un enlace a la licencia e indicar si se han realizado cambios.
- **No comercial.** No puedes utilizar el material con fines comerciales.
- **Compartir igual.** Si remezclas, transformas o creas a partir del material, debes distribuir tu contribución bajo la misma licencia.

El texto legal completo de la licencia está en el fichero [LICENSE](#) del repositorio.

2 Repaso de probabilidad

La estadística bayesiana se apoya en el cálculo de probabilidades. Este capítulo repasa los conceptos básicos que necesitaremos: frecuencias muestrales, distribuciones de frecuencias, sucesos aleatorios, las distintas definiciones de probabilidad, la probabilidad condicionada y las variables aleatorias con sus distribuciones más importantes.

2.1. Frecuencias muestrales

Definición 2.1 (Frecuencias muestrales). Dada una muestra de tamaño n de una variable X , para cada valor x_i de la variable observado en la muestra se define

- **Frecuencia absoluta** n_i : número de veces que el valor x_i aparece en la muestra.
- **Frecuencia relativa** f_i : proporción de veces que el valor x_i aparece en la muestra,

$$f_i = \frac{n_i}{n}$$

Ejemplo 2.1. Si tomamos una muestra aleatoria de 10 personas en la que hay 6 mujeres, la frecuencia absoluta de mujeres es $n_i = 6$ y la frecuencia relativa es

$$f_i = \frac{6}{10} = 0,6.$$

2.2. Distribuciones de frecuencias

Las frecuencias muestrales se organizan habitualmente en *tablas de frecuencias*, que muestran para cada valor observado x_i su frecuencia absoluta n_i o su frecuencia relativa f_i .

Ejemplo 2.2. Número de hijos en una muestra de 25 familias.

```
# Data frame número de hijos
df <- tibble(
  hijos = rep(c(0,1,2,3,4),
             c(2,6,14,2,1))
)
# Tabla de frecuencias
df |>
  tabyl(hijos) |>
  kable()
```

hijos	n	percent
0	2	0.08
1	6	0.24
2	14	0.56
3	2	0.08
4	1	0.04

```
# Diagrama de barras
df |> ggplot(aes(x = hijos)) +
  geom_bar(width = 0.2, fill = myblue) +
  theme_minimal() +
  labs(x = "Número de hijos", y = "Frecuencia absoluta")
```

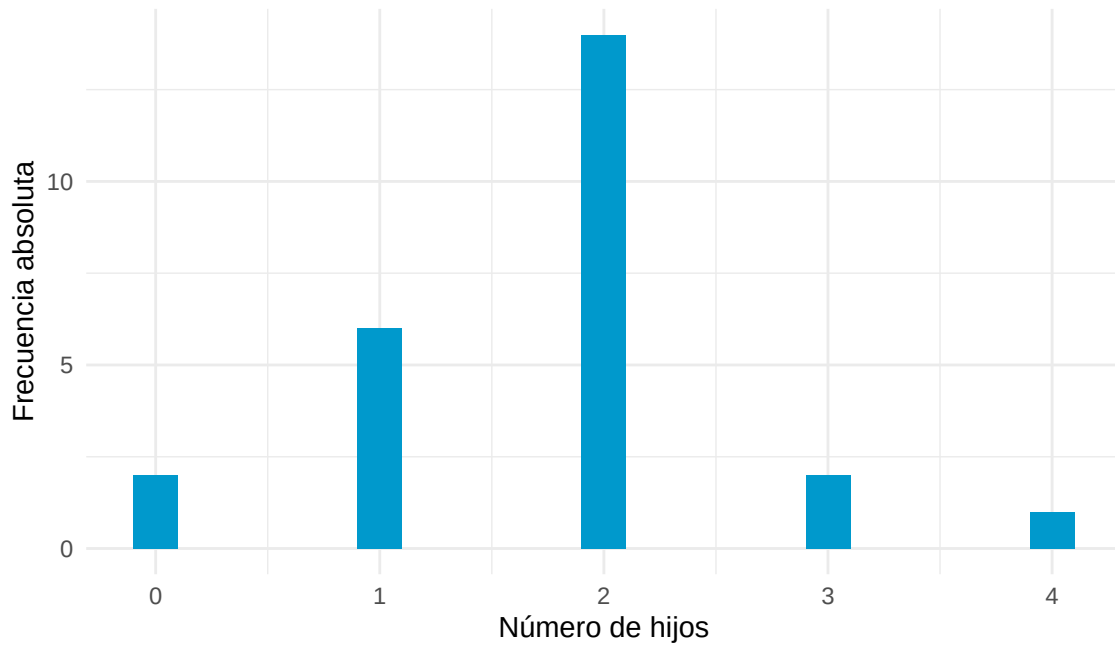


Figura 2.1: Distribución de frecuencias absolutas del número de hijos.

Ejemplo 2.3. Grupos sanguíneos de una muestra de 30 personas.

```
# Data frame grupo sanguíneo
df <- tibble(
  grupo = rep(c("0", "A", "B", "AB"),
             c(5, 14, 8, 3))
)
# Tabla de frecuencias
df |>
  tabyl(grupo) |>
  kable()
```

grupo	n	percent
0	5	0.1666667
A	14	0.4666667
AB	3	0.1000000
B	8	0.2666667

Ejercicio 2.1. El conjunto de datos `neonatos` contiene información sobre 320 recién nacidos en varios hospitales de una ciudad.

- Crear una tabla de frecuencias absolutas y relativas para la variable `sexo`.
- Crear una tabla de frecuencias absolutas y relativas para la variable `tabaco`.
- Representar gráficamente la distribución de frecuencias absolutas de la variable `tabaco`.
- Crear una tabla de frecuencias absolutas y relativas para la variable `apgar1`.
- Representar gráficamente la distribución de frecuencias relativas de la variable `apgar1`.

Solución

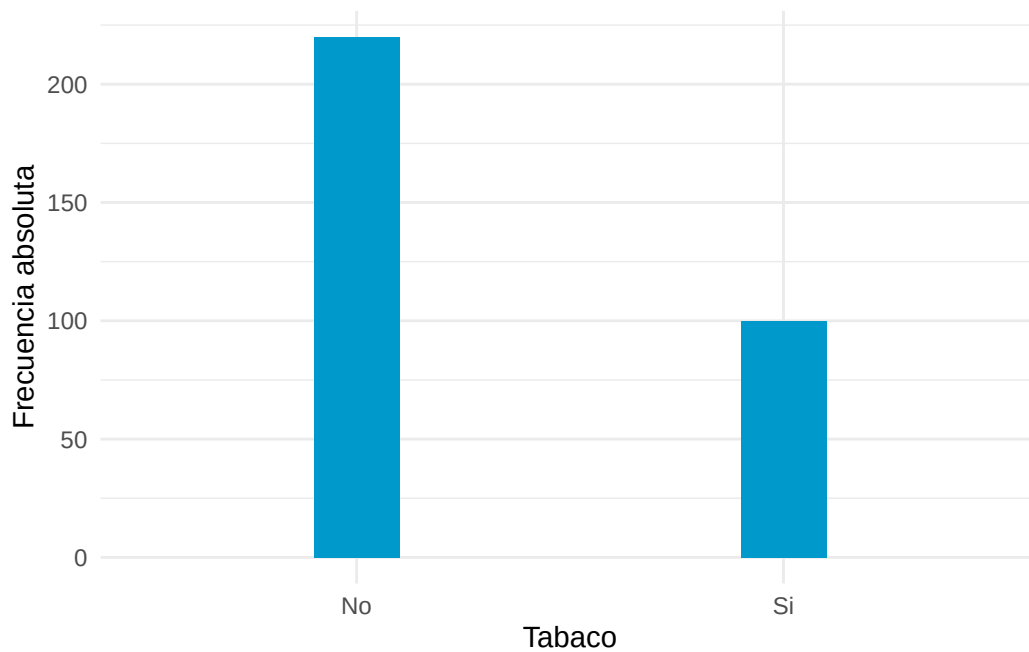
```
df_neonatos <- read_csv("datos/neonatos.csv", show_col_types = FALSE)
tabyl(df_neonatos, sexo) |> kable()
```

sexo	n	percent
hombre	157	0.490625
mujer	163	0.509375

```
tabyl(df_neonatos, tabaco) |> kable()
```

tabaco	n	percent
No	220	0.6875
Si	100	0.3125

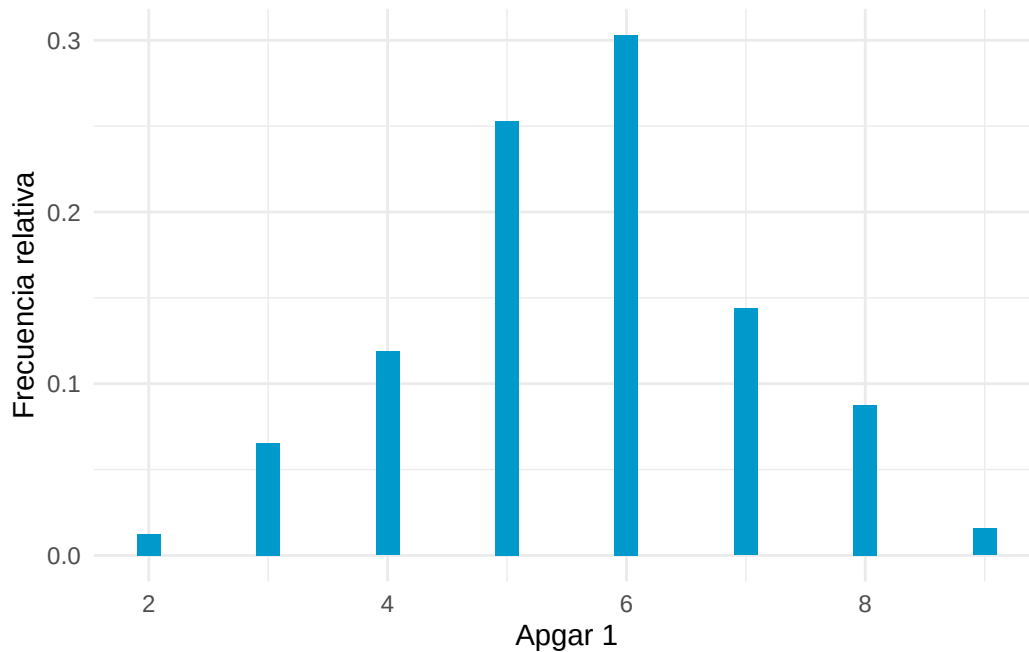
```
df_neonatos |> ggplot(aes(x = tabaco)) +
  stat_count(width = 0.2, fill = myblue) +
  theme_minimal() +
  labs(x = "Tabaco", y = "Frecuencia absoluta")
```



```
tabyl(df_neonatos, apgar1) |> kable()
```

apgar1	n	percent
2	4	0.012500
3	21	0.065625
4	38	0.118750
5	81	0.253125
6	97	0.303125
7	46	0.143750
8	28	0.087500
9	5	0.015625

```
df_neonatos |> ggplot(aes(x = apgar1, y = after_stat(prop))) +
  stat_count(width = 0.2, fill = myblue) +
  theme_minimal() +
  labs(x = "Apgar 1", y = "Frecuencia relativa")
```



2.3. Distribuciones de frecuencias bidimensionales

Cuando se consideran dos variables, sus frecuencias se organizan en una *tabla de contingencia*, que muestra para cada combinación de valores de las variables su frecuencia absoluta o relativa.

Ejemplo 2.4. Sexo y grupo sanguíneo de una muestra de 30 personas.

```
# Data frame sexo y grupo sanguíneo
df <- tibble(
  sexo = rep(c("Hombre", "Mujer"),
             c(16, 14)),
  grupo = c(rep("O", 3), rep("A", 8), rep("B", 4), rep("AB", 1),
            rep("O", 2), rep("A", 6), rep("B", 4), rep("AB", 2))
)
# Tabla bidimensional de frecuencias absolutas
df |>
  tabyl(grupo, sexo) |>
  kable()
```

grupo	Hombre	Mujer
0	3	2
A	8	6
AB	1	2
B	4	4

```
# Tabla bidimensional de frecuencias relativas
df |>
  tabyl(grupo, sexo) |>
  adorn_percentages("all") |>
  adorn_pct_formatting() |>
  kable()
```

grupo	Hombre	Mujer
0	10.0 %	6.7 %
A	26.7 %	20.0 %
AB	3.3 %	6.7 %
B	13.3 %	13.3 %

2.3.1. Distribuciones de frecuencias marginales

Sumando por filas o por columnas de una tabla de contingencia se obtienen las distribuciones marginales de cada variable, que muestran la frecuencia de cada valor de una variable sin tener en cuenta la otra.

Ejemplo 2.5.

```
# Distribución marginal del sexo
df |>
  tabyl(grupo, sexo) |>
  adorn_totals("row") |>
  kable()
```

grupo	Hombre	Mujer
0	3	2
A	8	6
AB	1	2
B	4	4

grupo	Hombre	Mujer
Total	16	14

```
# Distribución marginal del grupo sanguíneo
df |>
  tabyl(grupo, sexo) |>
  adorn_totals("col") |>
  kable()
```

grupo	Hombre	Mujer	Total
0	3	2	5
A	8	6	14
AB	1	2	3
B	4	4	8

Ejercicio 2.2. El conjunto de datos `neonatos` contiene información sobre 320 recién nacidos en varios hospitales de una ciudad.

- Crear una tabla de frecuencias absolutas bidimensional para las variables `edad` y `tabaco`. Añadir las frecuencias marginales de cada variable.
- Crear una tabla de frecuencias relativas bidimensional para las variables `tabaco` y `apgar1` para cada grupo de `edad`.

Solución

```
df_neonatos |>
  tabyl(edad, tabaco) |>
  adorn_totals(c("row", "col")) |>
  kable()
```

edad	No	Si	Total
mayor de 20	140	78	218
menor de 20	80	22	102
Total	220	100	320

```
df_neonatos |>
  tabyl(tabaco, apgar1, edad) |>
  adorn_percentages("all") |>
  adorn_pct_formatting() |>
  kable()
```

tabaco	2	3	4	5	6	7	8	9
No	0.0 %	1.4 %	3.7 %	11.5 %	23.9 %	13.3 %	8.7 %	1.8 %
Si	0.9 %	3.2 %	6.4 %	12.8 %	7.8 %	2.3 %	2.3 %	0.0 %
tabaco	2	3	4	5	6	7	8	9
No	1.0 %	2.9 %	9.8 %	24.5 %	24.5 %	10.8 %	3.9 %	1.0 %
Si	1.0 %	7.8 %	5.9 %	2.9 %	2.9 %	1.0 %	0.0 %	0.0 %

2.4. Distribuciones de frecuencias condicionales

Cada fila o columna de una tabla de contingencia es una distribución de frecuencias condicional, que muestra la frecuencia de cada valor de una variable para un valor fijo de la otra.

Ejemplo 2.6.

```
# Distribución condicional del grupo sanguíneo para hombres
df |>
  tabyl(grupo, sexo) |>
  kable() |>
  column_spec(2, color = myblue)
```

grupo	Hombre	Mujer
0	3	2
A	8	6
AB	1	2
B	4	4

```
# Distribución condicional del sexo para el grupo A
df |>
  tabyl(grupo, sexo) |>
  kable() |>
  row_spec(2, color = myblue)
```

grupo	Hombre	Mujer
0	3	2
A	8	6
AB	1	2
B	4	4

grupo	n	percent	grupo	n	percent
0	3	0.1875	0	2	0.1428571
A	8	0.5000	A	6	0.4285714
AB	1	0.0625	AB	2	0.1428571
B	4	0.2500	B	4	0.2857143

El siguiente código permite obtener la distribución de los grupos sanguíneos de las mujeres, y el que le sigue, las distribuciones condicionales para todas las categorías del **sexo**.

```
# Filtramos las mujeres
filter(df, sexo == "Mujer") |>
  tabyl(grupo) |>
  kable()
```

grupo	n	percent
0	2	0.1428571
A	6	0.4285714
AB	2	0.1428571
B	4	0.2857143

```
df |>
  split(~sexo) |>
  map(~tabyl(.x, grupo)) |>
  kable()
```

Ejercicio 2.3. El conjunto de datos **neonatos** contiene información sobre 320 recién nacidos en varios hospitales de una ciudad.

- Crear las tablas de frecuencias condicionales de **tabaco** para cada grupo de **edad**.
- Crear las tablas de frecuencias condicionales de **apgar1** para cada grupo de **tabaco**.
- Crear la tabla de frecuencias condicionales de **apgar1** para hijos de madres fumadoras de menos de 20 años.

Solución

```
df_neonatos |>
  split(~edad) |>
  map(~tabyl(.x, tabaco)) |>
  kable()
```

tabaco	n	percent	tabaco	n	percent
No	140	0.6422018	No	80	0.7843137
Si	78	0.3577982	Si	22	0.2156863

apgar1	n	percent	apgar1	n	percent
2	1	0.0045455	2	3	0.03
3	6	0.0272727	3	15	0.15
4	18	0.0818182	4	20	0.20
5	50	0.2272727	5	31	0.31
6	77	0.3500000	6	20	0.20
7	40	0.1818182	7	6	0.06
8	23	0.1045455	8	5	0.05
9	5	0.0227273			

```
df_neonatos |>
  split(~tabaco) |>
  map(~tabyl(.x, apgar1)) |>
  kable()
```

```
filter(df_neonatos, tabaco == "Si", edad == "menor de 20") |>
  tabyl(apgar1) |>
  kable()
```

apgar1	n	percent
2	1	0.0454545
3	8	0.3636364
4	6	0.2727273
5	3	0.1363636
6	3	0.1363636
7	1	0.0454545

2.5. Experimentos aleatorios

Definición 2.2 (Experimento aleatorio). Un *experimento aleatorio* es un experimento que cumple dos condiciones:

1. El conjunto de posibles resultados es conocido.
2. Es *no determinista*, es decir, no se puede predecir con absoluta certeza el resultado del experimento.

En algunos contextos se añade la condición de que el experimento sea *reproducible* bajo las mismas condiciones, aunque esto no siempre es posible. Podemos repetir el lanzamiento de una moneda en las mismas condiciones, pero no podemos repetir el nacimiento de un niño en las mismas condiciones.

2.6. Espacio muestral

Definición 2.3 (Espacio muestral). Al conjunto Ω de todos los posibles resultados de un experimento aleatorio se le llama *espacio muestral*.

Ejemplo 2.7. Algunos ejemplos de espacios muestrales son:

- Lanzamiento de una moneda: $\Omega = \{C, X\}$.
- Lanzamiento de un dado: $\Omega = \{1, 2, 3, 4, 5, 6\}$.
- Grupo sanguíneo de un individuo seleccionado al azar: $\Omega = \{A, B, AB, 0\}$.
- Estatura de un individuo seleccionado al azar: $\Omega = \mathbb{R}^+$.

2.6.1. Diagrama de árbol

Ejemplo 2.8. El siguiente diagrama de árbol representa el espacio muestral de un experimento aleatorio en el que se observan el sexo y el grupo sanguíneo de un individuo al azar.

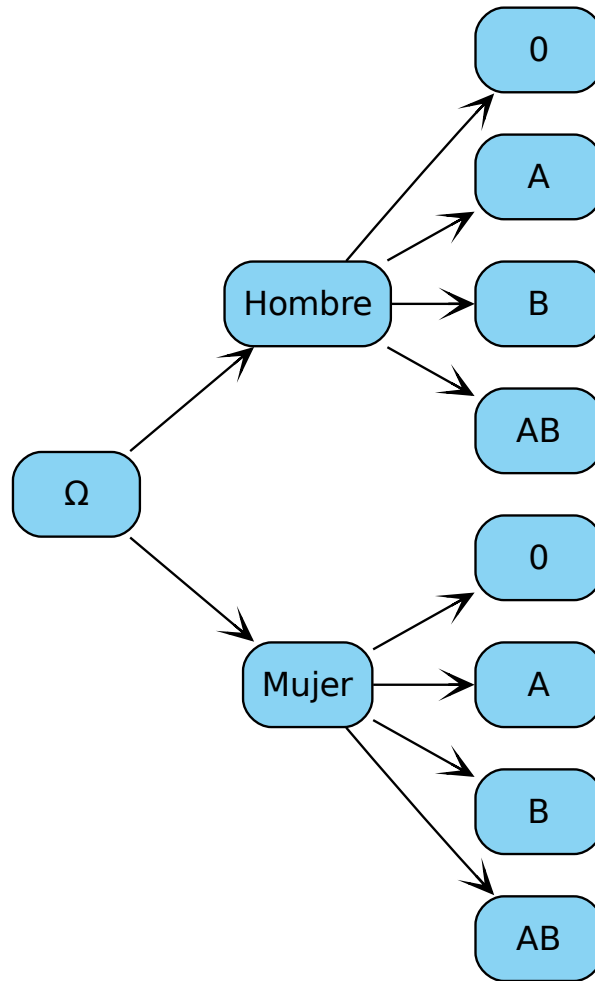


Figura 2.2: Diagrama de árbol del espacio muestral del sexo y el grupo sanguíneo.

2.7. Sucesos aleatorios

Definición 2.4 (Suceso aleatorio). Un *suceso aleatorio* es cualquier subconjunto del espacio muestral Ω de un experimento aleatorio.

Ejemplo 2.9. Algunos ejemplos de sucesos aleatorios son:

- En el lanzamiento de un dado, el suceso A de obtener un número par: $A = \{2, 4, 6\}$.

- En la medición del nivel de colesterol en sangre de un individuo, el suceso B de que el nivel de colesterol sea superior a 200 mg/dL: $B = \{x \in \mathbb{R}^+ : x > 200\}$.

2.8. Operaciones entre sucesos

Puesto que los sucesos son conjuntos, mediante la teoría de conjuntos se pueden definir las siguientes operaciones entre sucesos: unión, intersección, complementario y diferencia.

2.8.1. Unión de sucesos

Definición 2.5 (Suceso unión). Dados dos sucesos $A, B \subseteq \Omega$, se llama *suceso unión* de A y B , y se denota $A \cup B$, al suceso formado por los elementos de A junto con los elementos de B :

$$A \cup B = \{x \mid x \in A \text{ o } x \in B\}.$$

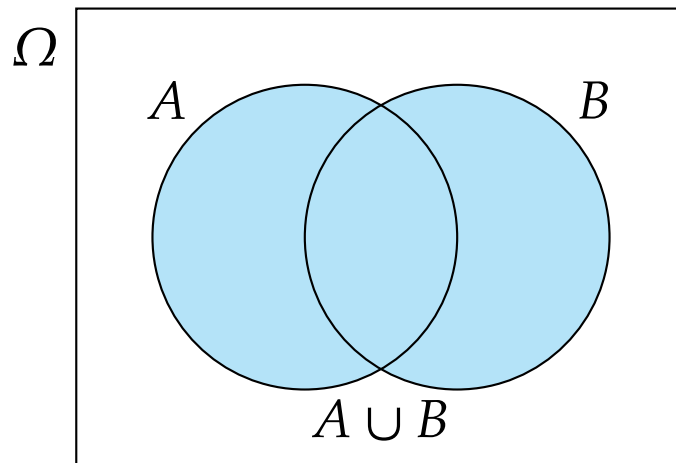


Figura 2.3: Unión de dos sucesos.

El suceso unión $A \cup B$ ocurre siempre que ocurre A o B .

2.8.2. Intersección de sucesos

Definición 2.6 (Suceso intersección). Dados dos sucesos $A, B \subseteq \Omega$, se llama *suceso intersección* de A y B , y se denota $A \cap B$, al suceso formado por los elementos comunes de A y B :

$$A \cap B = \{x \mid x \in A \text{ y } x \in B\}.$$

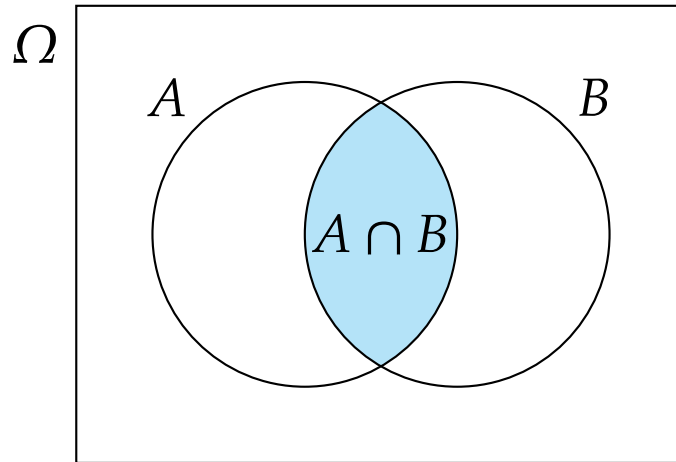


Figura 2.4: Intersección de dos sucesos.

El suceso intersección $A \cap B$ ocurre siempre que ocurren A y B .

Diremos que dos sucesos son **incompatibles** si su intersección es vacía.

2.8.3. Contrario de un suceso

Definición 2.7 (Suceso contrario). Dado un suceso $A \subseteq \Omega$, se llama *suceso contrario* o *complementario* de A , y se denota $\overline{A}$, al suceso formado por los elementos de Ω que no pertenecen a A :

$$\overline{A} = \{x \mid x \notin A\}.$$

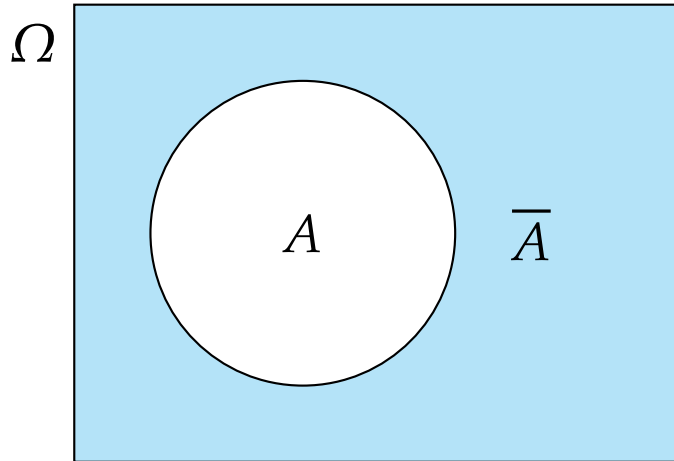


Figura 2.5: Contrario de un suceso.

El suceso contrario $\bar{A}$ ocurre siempre que **no** ocurre A .

2.8.4. Diferencia de sucesos

Definición 2.8 (Suceso diferencia). Dados dos sucesos $A, B \subseteq \Omega$, se llama *suceso diferencia* de A y B , y se denota $A - B$, al suceso formado por los elementos de A que no pertenecen a B :

$$A - B = \{x \mid x \in A \text{ y } x \notin B\} = A \cap \bar{B}.$$

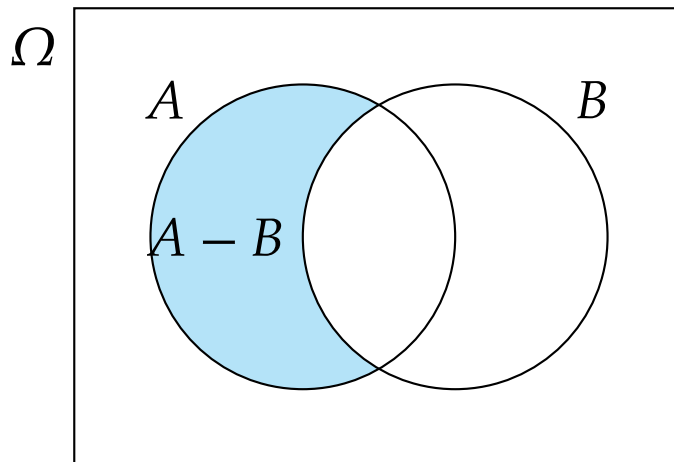


Figura 2.6: Diferencia de sucesos.

El suceso diferencia $A - B$ ocurre siempre que ocurre A pero no ocurre B .

Ejemplo 2.10. Dado el espacio muestral correspondiente al lanzamiento de un dado, $\Omega = \{1, 2, 3, 4, 5, 6\}$, y los sucesos $A = \{2, 4, 6\}$ y $B = \{1, 2, 3, 4\}$:

- La unión de A y B es $A \cup B = \{1, 2, 3, 4, 6\}$.
- La intersección de A y B es $A \cap B = \{2, 4\}$.
- El contrario de A es $\bar{A} = \{1, 3, 5\}$.
- Los sucesos A y $\bar{A}$ son incompatibles.
- La diferencia de A y B es $A - B = \{6\}$, y la diferencia de B y A es $B - A = \{1, 3\}$.

2.9. Definiciones de probabilidad

2.9.1. Definición clásica de probabilidad

Definición 2.9 (Probabilidad de Laplace). Dado un espacio muestral Ω de un experimento aleatorio en el que todos los elementos de Ω son *equiprobables*, la *probabilidad* de un suceso $A \subseteq \Omega$ es el cociente entre el número de elementos de A y el número de elementos de Ω :

$$P(A) = \frac{|A|}{|\Omega|} = \frac{\text{n}^\circ \text{ casos favorables a } A}{\text{n}^\circ \text{ casos posibles}}$$

Esta definición es ampliamente utilizada, pero tiene importantes restricciones:

- Es necesario que todos los elementos del espacio muestral tengan la misma probabilidad de ocurrir (*equiprobabilidad*).
- No puede utilizarse con espacios muestrales infinitos, ni con espacios de los que no se conoce el número de casos posibles.

Precaución

Estas condiciones no se cumplen en muchos experimentos aleatorios reales.

Ejemplo 2.11. Dado el espacio muestral correspondiente al lanzamiento de un dado, $\Omega = \{1, 2, 3, 4, 5, 6\}$, y el suceso $A = \{2, 4, 6\}$, la probabilidad de A es

$$P(A) = \frac{|A|}{|\Omega|} = \frac{3}{6} = 0,5.$$

Sin embargo, si consideramos el espacio muestral correspondiente a observar el grupo sanguíneo de un individuo al azar, $\Omega = \{0, A, B, AB\}$, no podemos usar la definición clásica para calcular la probabilidad de que tenga grupo sanguíneo A ,

$$P(A) \neq \frac{|A|}{|\Omega|} = \frac{1}{4} = 0,25,$$

porque los grupos sanguíneos no son igualmente probables en las poblaciones humanas.

2.9.2. Definición frecuentista de probabilidad

Teorema 2.1 (Ley de los grandes números). *Cuando un experimento aleatorio se repite un gran número de veces, las frecuencias relativas de los sucesos del experimento tienden a estabilizarse en torno a cierto número, que es precisamente su probabilidad.*



Figura 2.7: Lanzamiento de una moneda.

Definición 2.10 (Probabilidad frecuentista). Dado un espacio muestral Ω de un experimento aleatorio *reproducible*, la *probabilidad* de un suceso $A \subseteq \Omega$ es la frecuencia relativa del suceso A en infinitas repeticiones del experimento:

$$P(A) = \lim_{n \rightarrow \infty} \frac{n_A}{n}$$

Aunque esta definición es muy útil en experimentos científicos reproducibles, tiene serios inconvenientes:

- Sólo se puede calcular una aproximación de la probabilidad real.
- La repetición del experimento debe hacerse en las mismas condiciones.

Ejemplo 2.12. Dado el espacio muestral correspondiente al lanzamiento de una moneda, $\Omega = \{C, X\}$, si después de lanzar la moneda 100 veces obtenemos 54 caras, la probabilidad de C es aproximadamente

$$P(C) \approx \frac{n_C}{n} = \frac{54}{100} = 0,54.$$

Ejemplo 2.13. Si consideramos el espacio muestral correspondiente a observar el grupo sanguíneo de un individuo al azar, $\Omega = \{0, A, B, AB\}$, y tomamos una muestra aleatoria de 1000 personas en la que 412 tienen grupo sanguíneo A , la probabilidad del grupo sanguíneo A es aproximadamente

$$P(A) \approx \frac{n_A}{n} = \frac{412}{1000} = 0,412.$$

2.9.3. Definición subjetiva de probabilidad

Definición 2.11 (Probabilidad subjetiva). Dado un espacio muestral Ω de un experimento aleatorio, la probabilidad de un suceso $A \subseteq \Omega$ es una medida de la confianza que un individuo tiene en que el suceso A ocurra, y se expresa mediante un número real entre 0 y 1.

- Se basa en la experiencia personal, el conocimiento previo y la información disponible, por lo que es distinta para cada individuo y puede cambiar con nueva información o experiencias.
- No existe una fórmula matemática para calcularla.

Es la definición de probabilidad más usada en nuestras decisiones cotidianas, pero también tiene inconvenientes:

- Es subjetiva, por lo que puede tener sesgos.
- No se puede calcular de forma objetiva, así que no es posible contrastar su valor con la realidad.

Ejemplo 2.14. Basándome en el tiempo de los días previos, puedo hacer la predicción de que es moderadamente probable que llueva mañana. Sin embargo, mi abuelo puede hacer esta otra predicción: *me duele la rodilla, así que mañana va a llover*.

2.9.4. Definición axiomática de probabilidad

Definición 2.12 (Probabilidad de Kolmogórov). Dado un espacio muestral Ω de un experimento aleatorio, una función de *probabilidad* es una aplicación que asocia a cada suceso $A \subseteq \Omega$ un número real $P(A)$ que cumple los siguientes axiomas:

1. La probabilidad de un suceso cualquiera es positiva o nula: $P(A) \geq 0$.
2. La probabilidad del suceso seguro es igual a la unidad: $P(\Omega) = 1$.
3. La probabilidad de la unión de dos sucesos incompatibles ($A \cap B = \emptyset$) es igual a la suma de sus probabilidades: $P(A \cup B) = P(A) + P(B)$.

Teorema 2.2. Si P es una función de probabilidad de un espacio muestral Ω , entonces para cualesquiera sucesos $A, B \subseteq \Omega$ se cumple:

1. $P(\bar{A}) = 1 - P(A)$.
2. $P(\emptyset) = 0$.
3. Si $A \subseteq B$ entonces $P(A) \leq P(B)$.
4. $P(A) \leq 1$.
5. $P(A - B) = P(A) - P(A \cap B)$.
6. $P(A \cup B) = P(A) + P(B) - P(A \cap B)$.
7. Si $A = \{e_1, e_2, \dots, e_n\}$, entonces $P(A) = \sum_{i=1}^n P(e_i)$.

2.9.5. Interpretación de la probabilidad

Dos interpretaciones

- **Frecuentista:** frecuencia relativa a largo plazo de un suceso en un experimento aleatorio reproducible.
 - **Bayesiana:** verosimilitud o plausibilidad de un suceso, dada la información disponible.
-
- La mayor incertidumbre corresponde a $P(A) = 0,5$: es igual de probable que ocurra A que no ocurra.
 - La menor incertidumbre corresponde a $P(A) = 1$ (A sucederá con absoluta certeza) y a $P(A) = 0$ (A no sucederá con absoluta certeza).

2.10. Probabilidad condicionada

En algunas ocasiones tenemos información sobre el experimento antes de realizarlo. Habitualmente esa información se da en forma de un suceso B del mismo espacio muestral que sabemos que ha ocurrido. En tal caso, B es un suceso *condicionante*, y la probabilidad de otro suceso A se expresa como

$$P(A \mid B),$$

que se lee *probabilidad de A dado B* o *probabilidad de A bajo la condición de B*.

Los condicionantes suelen cambiar el espacio muestral del experimento y, por tanto, las probabilidades de sus sucesos.

Ejemplo 2.15. Supongamos que tenemos una muestra de 100 hombres y 100 mujeres con las siguientes frecuencias:

	No fumadores	Fumadores
Mujeres	80	20
Hombres	60	40

Usando la definición frecuentista de probabilidad, la probabilidad de que una persona elegida al azar sea fumadora es

$$P(\text{Fumadora}) = \frac{60}{200} = 0,3.$$

Sin embargo, si sabemos que la persona elegida es mujer, la muestra se reduce a la primera fila, y la probabilidad de que sea fumadora es

$$P(\text{Fumadora} \mid \text{Mujer}) = \frac{20}{100} = 0,2.$$

Definición 2.13 (Probabilidad condicionada). Dado un espacio muestral Ω de un experimento aleatorio y dos sucesos $A, B \subseteq \Omega$, la *probabilidad de A condicionada por B* es

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)},$$

siempre que $P(B) \neq 0$.

Esta definición permite calcular probabilidades condicionadas sin tener que alterar el espacio muestral original del experimento.

Ejemplo 2.16. En el ejemplo anterior,

$$P(\text{Fumadora} \mid \text{Mujer}) = \frac{P(\text{Fumadora} \cap \text{Mujer})}{P(\text{Mujer})} = \frac{20/200}{100/200} = \frac{20}{100} = 0,2.$$

2.10.1. Probabilidad del suceso intersección

A partir de la definición de probabilidad condicionada se obtiene la fórmula para calcular la probabilidad de la intersección de dos sucesos:

$$P(A \cap B) = P(A)P(B \mid A) = P(B)P(A \mid B).$$

Ejemplo 2.17. En una población hay un 30 % de fumadores, y se sabe que el 40 % de los fumadores tiene cáncer de pulmón. La probabilidad de que una persona elegida al azar sea fumadora y tenga cáncer de pulmón es

$$\begin{aligned} P(\text{Fumadora} \cap \text{Cáncer}) &= P(\text{Fumadora}) P(\text{Cáncer} \mid \text{Fumadora}) \\ &= 0,3 \times 0,4 = 0,12. \end{aligned}$$

2.11. Independencia de sucesos

En ocasiones, la ocurrencia del suceso condicionante no cambia la probabilidad original del suceso principal.

Definición 2.14 (Sucesos independientes). Dado un espacio muestral Ω de un experimento aleatorio, dos sucesos $A, B \subseteq \Omega$ son *independientes* si la probabilidad de A no se ve alterada al condicionar por B , y viceversa, es decir,

$$P(A \mid B) = P(A) \quad \text{y} \quad P(B \mid A) = P(B),$$

si $P(A) \neq 0$ y $P(B) \neq 0$.

Esto significa que la ocurrencia de un suceso no aporta información relevante para cambiar la incertidumbre sobre el otro. Cuando dos sucesos son independientes, la probabilidad de su intersección es igual al producto de sus probabilidades:

$$P(A \cap B) = P(A)P(B).$$

2.11.1. Árboles de probabilidad con variables dependientes

Ejemplo 2.18. Sea una población en la que el 30 % de las personas fuma, y en la que la incidencia del cáncer de pulmón es del 40 % en fumadores y del 10 % en no fumadores.

El espacio probabilístico del experimento aleatorio que consiste en elegir una persona al azar y medir las variables *fumar* y *cáncer de pulmón* se muestra a continuación.

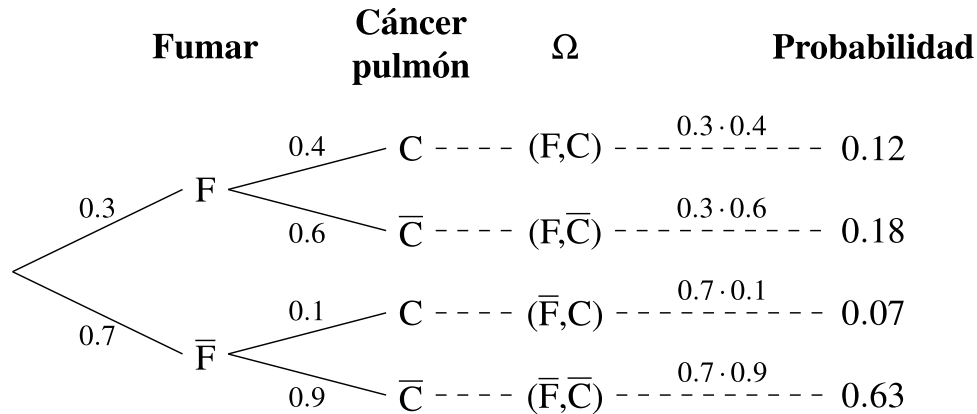


Figura 2.8: Diagrama de árbol del espacio probabilístico de fumar y tener cáncer de pulmón.

2.11.2. Árboles de probabilidad con variables independientes

Ejemplo 2.19. El árbol de probabilidad asociado al experimento aleatorio que consiste en el lanzamiento de dos monedas se muestra a continuación.

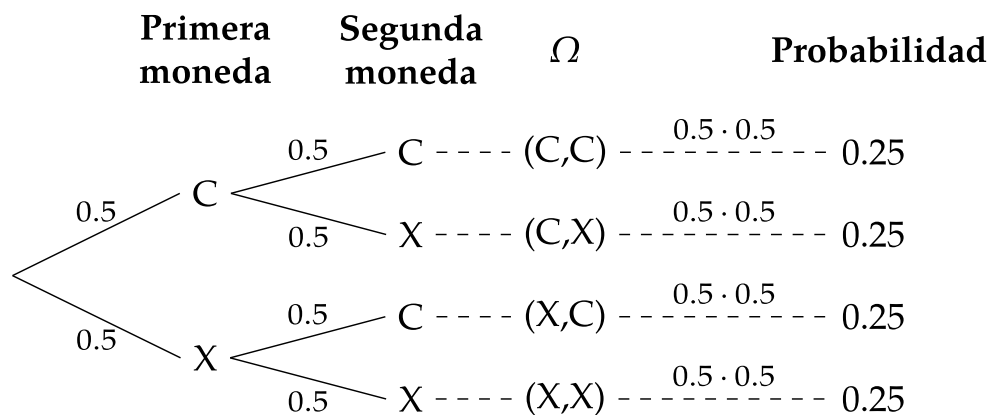


Figura 2.9: Diagrama de árbol del espacio probabilístico del lanzamiento de dos monedas.

Ejemplo 2.20. Dada una población en la que hay un 40 % de hombres y un 60 % de mujeres, el experimento aleatorio que consiste en tomar una muestra aleatoria de tres personas tiene el árbol de probabilidad que se muestra a continuación.

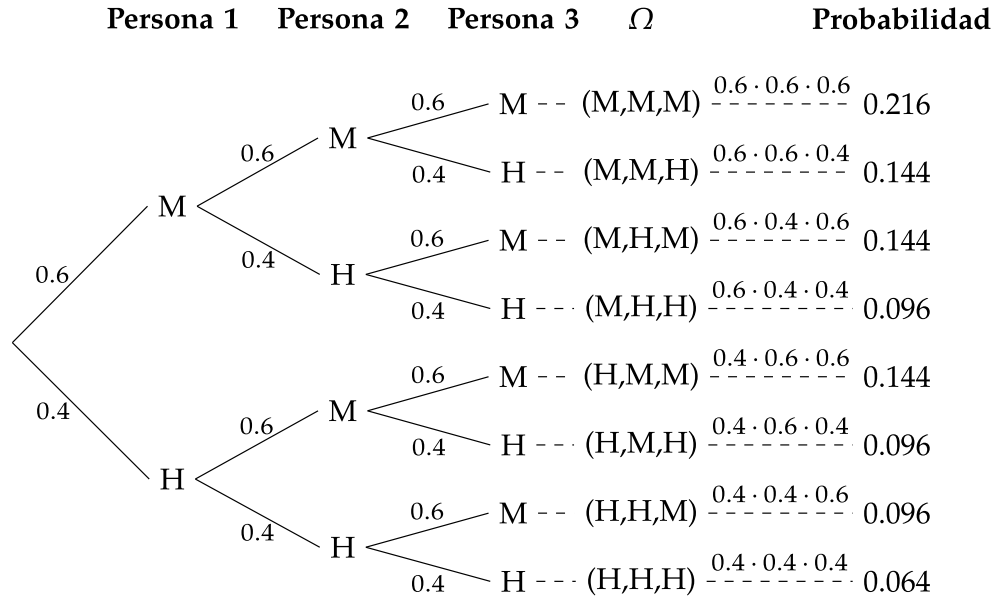


Figura 2.10: Diagrama de árbol del espacio probabilístico del sexo de tres individuos elegidos al azar.

2.12. Variables aleatorias

Definición 2.15 (Variable aleatoria). Una *variable aleatoria* X es una función que asocia un número real a cada elemento del espacio muestral de un experimento aleatorio:

$$X : \Omega \rightarrow \mathbb{R}.$$

Al conjunto de posibles valores que puede tomar la variable se le llama *rango* o *recorrido* de la variable, y se representa por $\text{Ran}(X)$.

En el fondo, una variable aleatoria es una variable cuyos valores provienen de la realización de un experimento aleatorio, y por tanto tendrá asociada una determinada distribución de probabilidad.

Ejemplo 2.21. La variable X que mide el número de personas infectadas con un virus en una muestra aleatoria de 10 personas es una variable aleatoria, cuyo rango es el conjunto de números enteros entre 0 y 10:

$$\text{Ran}(X) = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}.$$

2.12.1. Tipos de variables aleatorias

Las variables aleatorias se clasifican en dos tipos:

- **Discretas.** Toman valores aislados y su rango es numerable. Por ejemplo, el número de hijos, el número de cigarrillos o el número de asignaturas aprobadas.
- **Continuas.** Toman valores en un intervalo real y su rango no es numerable. Por ejemplo, el peso, la estatura, la edad o el nivel de colesterol.

2.13. Variables aleatorias discretas

Definición 2.16 (Función de probabilidad). La *función de probabilidad* de una variable aleatoria discreta X es una función $f(x)$ que asocia a cada valor de la variable su probabilidad:

$$f(x_i) = P(X = x_i).$$

Ejemplo 2.22. La función de probabilidad de la variable que mide el número de caras obtenidas al lanzar una moneda tres veces es

x	0	1	2	3
$f(x)$	$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{1}{8}$

2.13.1. Estadísticos de una variable aleatoria discreta

La **media** o esperanza de X es

$$E(X) = \sum_i x_i f(x_i),$$

y la **varianza** es

$$\text{Var}(X) = \sum_i (x_i - E(X))^2 f(x_i) = \sum_i x_i^2 f(x_i) - E(X)^2.$$

2.13.2. Distribución uniforme discreta

Definición 2.17 (Distribución uniforme discreta). Se dice que una variable aleatoria discreta X sigue un *modelo de distribución uniforme discreto*, $X \sim U(n)$, si su rango es $\text{Ran}(X) = \{1, \dots, n\}$ y su función de probabilidad es constante:

$$P(X = k) = \frac{1}{n} \quad \forall k = 1, \dots, n.$$

El modelo uniforme discreto se utiliza para modelar experimentos aleatorios con un número finito de resultados *equiprobables*, como el lanzamiento de un dado o la selección aleatoria de un individuo de una población finita.

Ejemplo 2.23. El modelo de distribución de probabilidad del resultado de lanzar un dado equilibrado es uniforme discreto $U(6)$.

```
df <- tibble(  
  x = 1:6, # Rango  
  p = rep(1/6, 6) # Probabilidad  
)  
# Diagrama de barras  
ggplot(df, aes(x = x, y = p)) +  
  geom_bar(stat = "identity", width = 0.2, fill = myblue) +  
  scale_x_continuous(breaks = 1:6) +  
  labs(x = "X", y = "Probabilidad") +  
  theme_minimal()
```

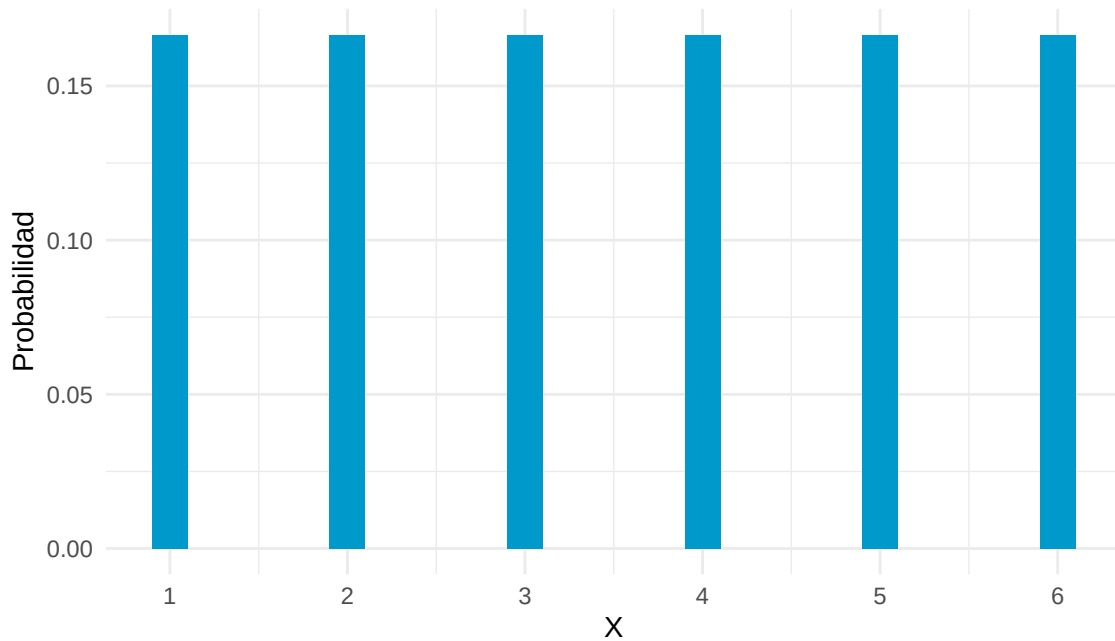


Figura 2.11: Distribución uniforme discreta $U(6)$.

2.13.3. Distribución binomial

Definición 2.18 (Distribución binomial). Se dice que una variable aleatoria discreta X sigue un *modelo de distribución binomial*, $X \sim B(n, p)$, si su rango es $\text{Ran}(X) = \{0, 1, \dots, n\}$ y su función de probabilidad es

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k} \quad k = 0, 1, \dots, n,$$

donde n es el *número de ensayos* y p la *probabilidad de éxito* en cada ensayo.

El modelo binomial se utiliza para modelar el número de éxitos en n ensayos independientes con dos posibles resultados (éxito/fracaso), como el número de caras al lanzar n monedas o el número de individuos enfermos en una muestra de tamaño n .

Sus estadísticos son

$$E(X) = np \quad \text{y} \quad \text{Var}(X) = np(1 - p).$$

Ejemplo 2.24. En una población hay el mismo número de hombres que de mujeres. La variable que mide el número de mujeres en una muestra aleatoria de 10 personas sigue una distribución binomial $B(10, 0.5)$.

```
df <- tibble(
  x = 0:10, # Rango
  p = dbinom(0:10, size = 10, prob = 0.5) # Probabilidad
)
# Diagrama de barras
ggplot(df, aes(x = x, y = p)) +
  geom_bar(stat = "identity", width = 0.2, fill = myblue) +
  scale_x_continuous(breaks = 0:10) +
  labs(x = "Número de mujeres", y = "Probabilidad") +
  theme_minimal()
```

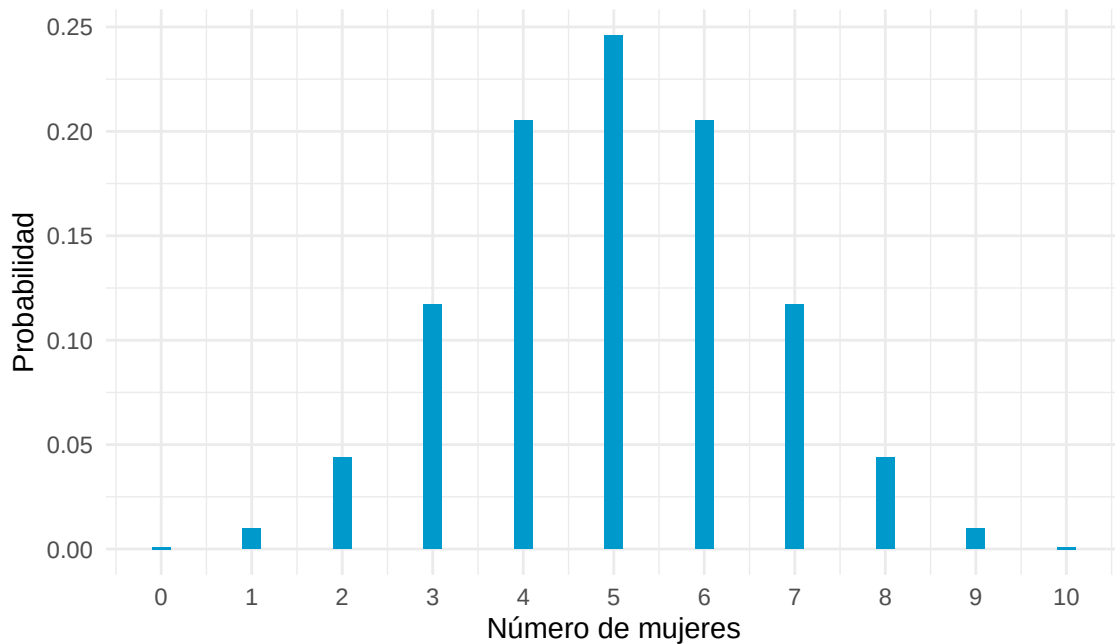


Figura 2.12: Distribución binomial $B(10, 0.5)$.

2.13.3.1. Forma de la distribución binomial para distintos valores de n y p

```
params <- expand_grid(n = c(10, 30, 100), p = c(0.1, 0.5, 0.9))
df_binom <- params |>
  rowwise() |>
  mutate(x = list(0:n), prob = list(dbinom(0:n, size = n, prob = p))) |>
  unnest(cols = c(x, prob)) |>
  mutate(
    label = factor(
```

```

    paste0("n = ", n, ", p = ", p),
    levels = paste0("n = ", rep(c(10, 30, 100), 3), ", p = ",
      ↪ rep(c(0.1, 0.5, 0.9), each = 3))
  )
)

df_binom |>
  ggplot(aes(x = x, y = prob)) +
  geom_bar(stat = "identity", width = 0.5, fill = myblue) +
  facet_wrap(~ label, scales = "free", nrow = 3) +
  labs(x = "X", y = "Probabilidad") +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))

```

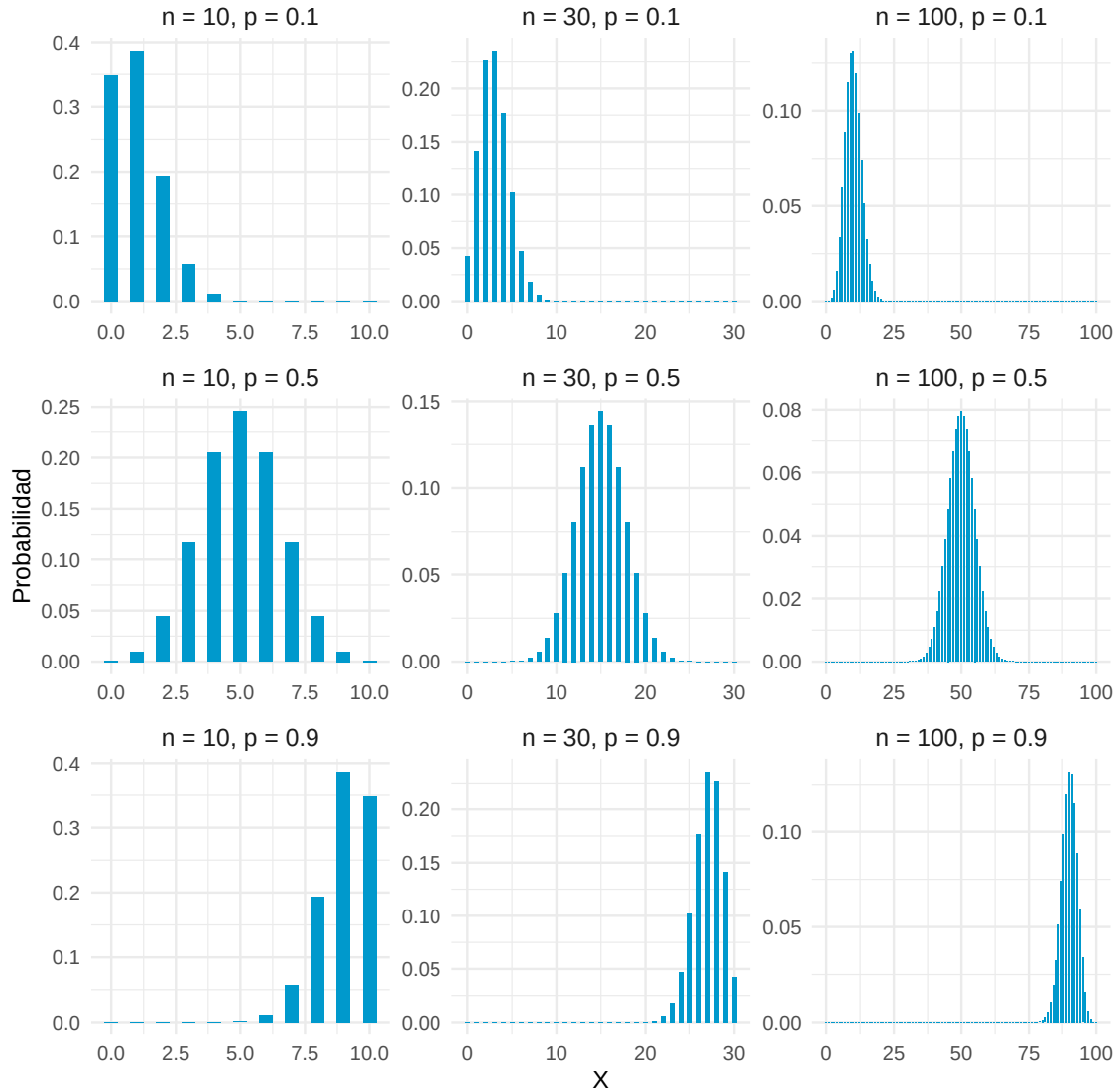


Figura 2.13: Forma de la distribución binomial para distintos valores de n y p .

2.13.4. Distribución geométrica

Definición 2.19 (Distribución geométrica). Se dice que una variable aleatoria discreta X sigue un *modelo de distribución geométrica*, $X \sim G(p)$, si su rango es $\text{Ran}(X) = \{0, 1, 2, \dots\}$ y su función de probabilidad es

$$P(X = k) = (1 - p)^k p \quad k = 0, 1, 2, \dots,$$

donde p es la *probabilidad de éxito* en cada ensayo.

El modelo geométrico se utiliza para modelar el número de fracasos necesarios hasta obtener el primer éxito en una serie de ensayos independientes con dos posibles resultados, como el número de pacientes sanos que se examinan hasta encontrar un paciente enfermo.

Sus estadísticos son

$$E(X) = \frac{1-p}{p} \quad y \quad Var(X) = \frac{1-p}{p^2}.$$

Ejemplo 2.25. En una población hay aproximadamente un 30 % de fumadores. La variable que mide el número de personas no fumadoras que hay que examinar hasta encontrar la primera persona fumadora sigue una distribución geométrica $G(0,3)$.

```
df <- tibble(
  x = 0:14, # Rango truncado (número de fracasos)
  p = dgeom(0:14, prob = 0.3) # Probabilidad
)
# Diagrama de barras
ggplot(df, aes(x = x, y = p)) +
  geom_bar(stat = "identity", width = 0.2, fill = myblue) +
  scale_x_continuous(breaks = 0:14) +
  labs(x = "Número de no fumadoras hasta la primera fumadora", y =
    ↪ "Probabilidad") +
  theme_minimal()
```

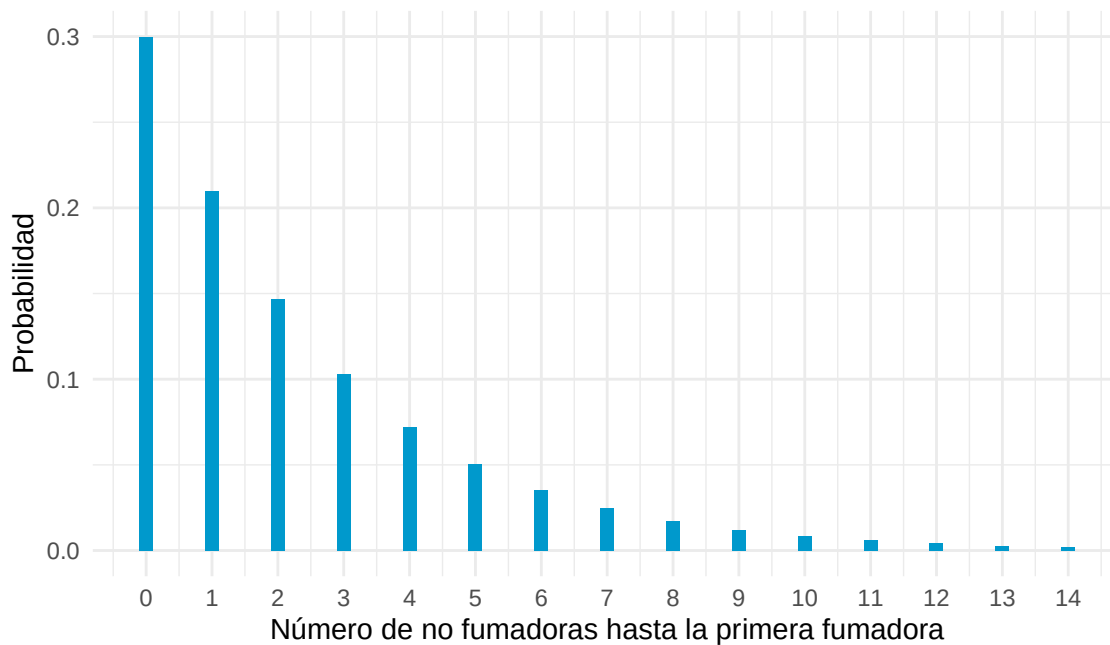


Figura 2.14: Distribución geométrica $G(0,3)$, truncada en 14 fracasos.

2.13.4.1. Forma de la distribución geométrica para distintos valores de p

```
ps <- c(0.1, 0.5, 0.9)
df_geom <- tibble(p = ps) |>
  rowwise() |>
  mutate(x = list(0:14), prob = list(dgeom(0:14, prob = p))) |>
  unnest(cols = c(x, prob)) |>
  mutate(label = factor(paste0("p == ", p),
                        levels = paste0("p == ", ps)))

df_geom |>
  ggplot(aes(x = x, y = prob)) +
  geom_bar(stat = "identity", width = 0.5, fill = myblue) +
  facet_wrap(~ label, nrow = 1, labeller = label_parsed) +
  labs(x = "X", y = "Probabilidad") +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))
```

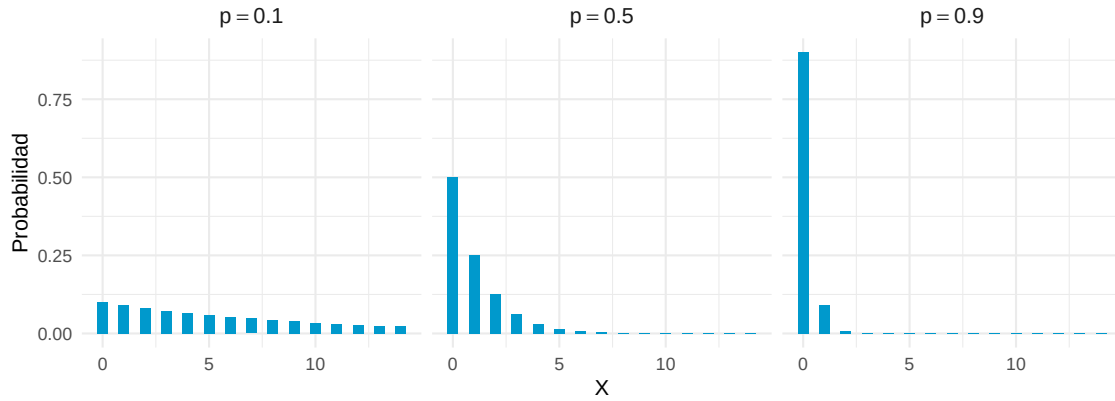


Figura 2.15: Forma de la distribución geométrica para distintos valores de p .

2.13.5. Distribución binomial negativa

Definición 2.20 (Distribución binomial negativa). Se dice que una variable aleatoria discreta X sigue un *modelo de distribución binomial negativa*, $X \sim BN(r, p)$, si su rango es $\text{Ran}(X) = \{0, 1, 2, \dots\}$ y su función de probabilidad es

$$P(X = k) = \binom{r + k - 1}{k} p^r (1 - p)^k \quad k = 0, 1, 2, \dots,$$

donde r es el *número de éxitos* y p la *probabilidad de éxito* en cada ensayo.

El modelo binomial negativo se utiliza para modelar el número de fracasos necesarios para obtener r éxitos en una serie de ensayos independientes con dos posibles resultados, como el número de pacientes sanos que se examinan hasta encontrar el tercer paciente enfermo.

Sus estadísticos son

$$E(X) = \frac{r(1-p)}{p} \quad \text{y} \quad \text{Var}(X) = \frac{r(1-p)}{p^2}.$$

Ejemplo 2.26. Se sabe que en una población hay aproximadamente un 30 % de fumadores. La variable que mide el número de personas no fumadoras hasta obtener 3 personas fumadoras sigue una distribución $BN(3, 0.3)$.

```
df <- tibble(
  x = 0:25, # Rango truncado (número de fracasos)
  p = dnbinom(0:25, size = 3, prob = 0.3) # Probabilidad
)
# Diagrama de barras
ggplot(df, aes(x = x, y = p)) +
  geom_bar(stat = "identity", width = 0.2, fill = myblue) +
  scale_x_continuous(breaks = seq(0, 25, by = 5)) +
  labs(x = "Número de no fumadoras hasta la tercera fumadora", y =
    ↪ "Probabilidad") +
  theme_minimal()
```

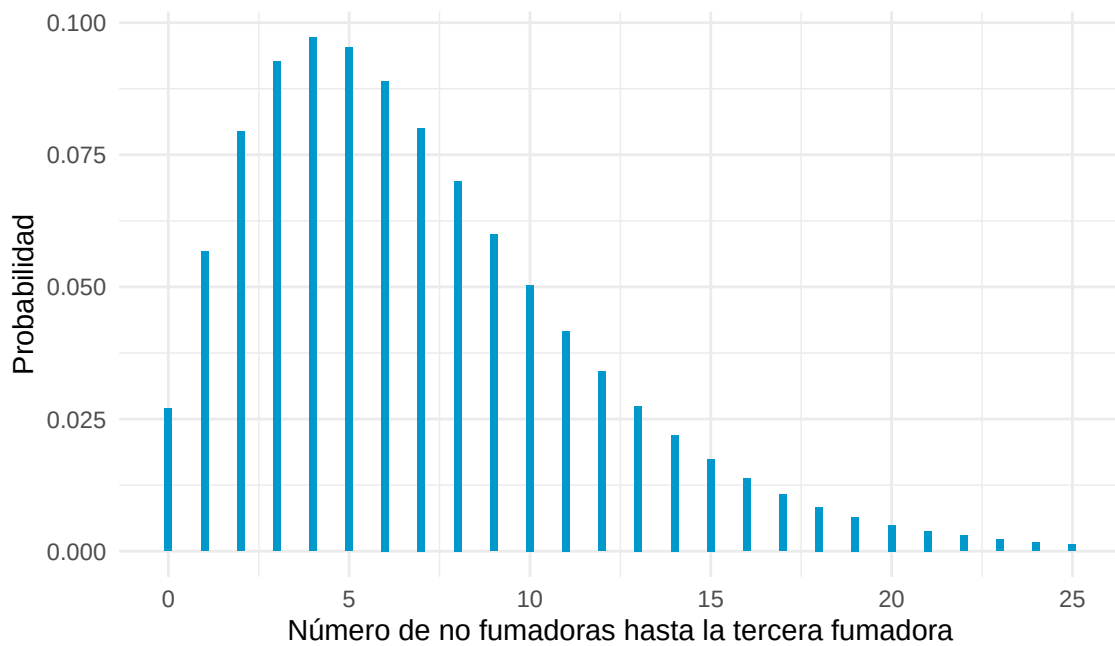


Figura 2.16: Distribución binomial negativa $BN(3, 0.3)$.

2.13.5.1. Forma de la distribución binomial negativa para distintos valores de r y p

```
params <- expand.grid(r = c(1, 3, 5), p = c(0.2, 0.5, 0.8))
df_binom <- params |>
  rowwise() |>
  mutate(x = list(0:30), prob = list(dnbinom(0:30, size = r, prob = p)))
  ↪ |>
  unnest(cols = c(x, prob)) |>
```

```

mutate(
  label = factor(
    paste0("r = ", r, ", p = ", p),
    levels = paste0("r = ", rep(c(1, 3, 5), 3), ", p = ", rep(c(0.2,
    ↪ 0.5, 0.8), each = 3))
  )
)

df_binom |>
  ggplot(aes(x = x, y = prob)) +
  geom_bar(stat = "identity", width = 0.5, fill = myblue) +
  facet_wrap(~ label, scales = "free", nrow = 3) +
  labs(x = "X", y = "Probabilidad") +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))

```

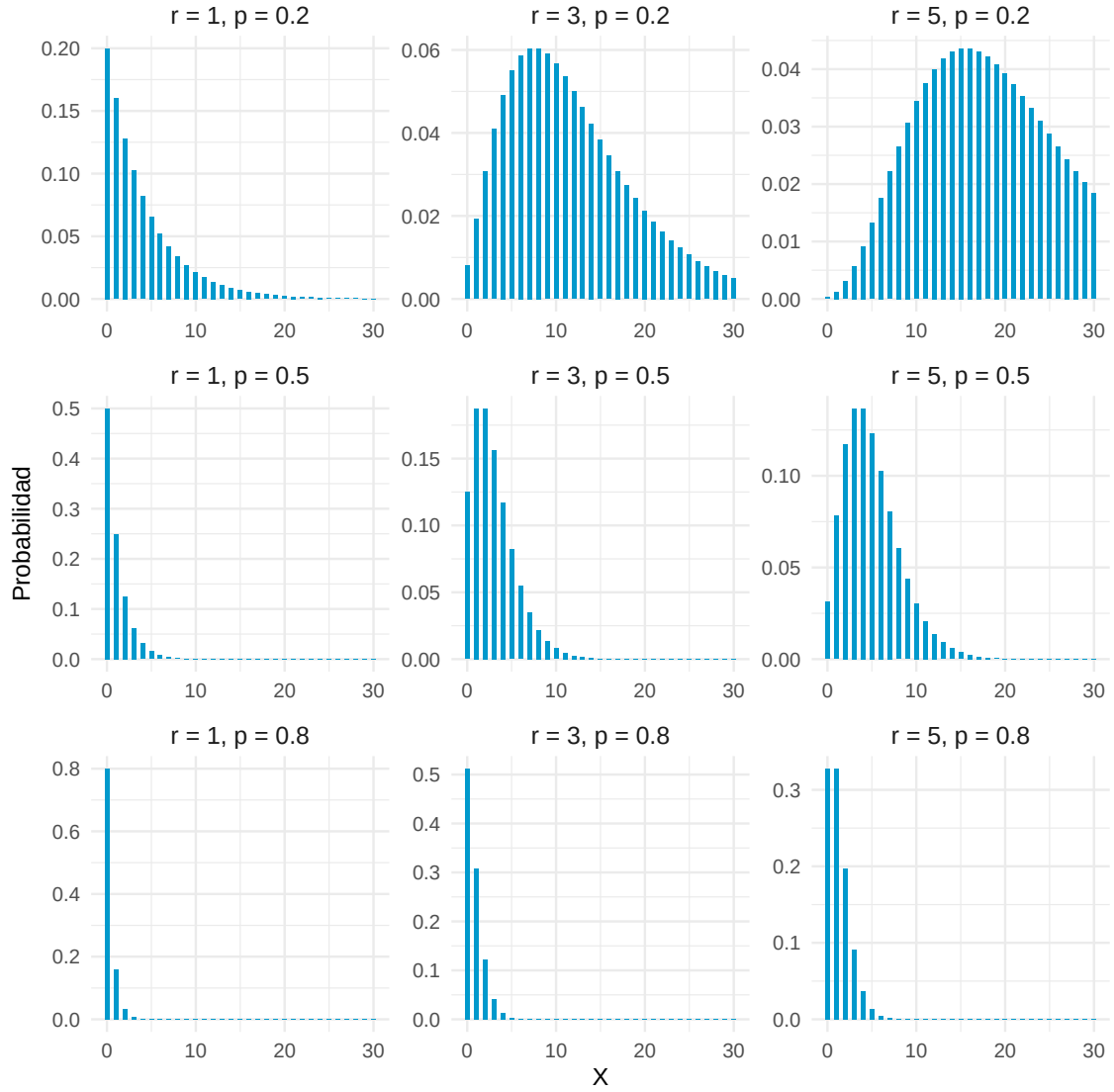


Figura 2.17: Forma de la distribución binomial negativa para distintos valores de r y p .

2.13.6. Distribución de Poisson

Definición 2.21 (Distribución de Poisson). Se dice que una variable aleatoria discreta X sigue un *modelo de distribución de Poisson*, $X \sim P(\lambda)$, si su rango es $\text{Ran}(X) = \{0, 1, 2, \dots\}$ y su función de probabilidad es

$$P(X = k) = \frac{e^{-\lambda} \lambda^k}{k!} \quad k = 0, 1, 2, \dots,$$

donde $\lambda > 0$ es el *número medio de ocurrencias* del suceso en el intervalo de referencia.

El modelo de Poisson se utiliza para modelar el número de ocurrencias de un suceso en un intervalo de tiempo o espacio fijo, como el número de llamadas que recibe una centralita en una hora o el número de mutaciones en un fragmento de ADN.

Sus estadísticos son

$$E(X) = Var(X) = \lambda.$$

Ejemplo 2.27. En una ciudad, el número medio de nacimientos diarios es 4. Entonces, el número de nacimientos en un día sigue una distribución de Poisson $P(4)$.

```
df <- tibble(  
  x = 0:12, # Rango truncado  
  p = dpois(0:12, lambda = 4) # Probabilidad  
)  
# Diagrama de barras  
ggplot(df, aes(x = x, y = p)) +  
  geom_bar(stat = "identity", width = 0.2, fill = myblue) +  
  scale_x_continuous(breaks = 0:12) +  
  labs(x = "Número de nacimientos", y = "Probabilidad") +  
  theme_minimal()
```

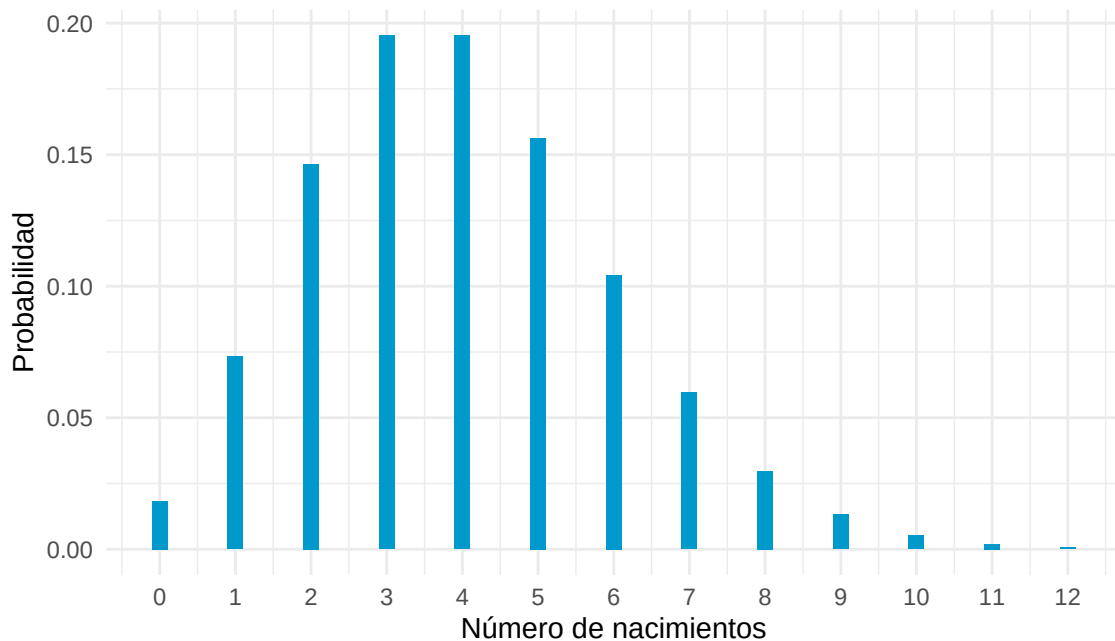


Figura 2.18: Distribución de Poisson $P(4)$.

2.13.6.1. Forma de la distribución de Poisson para distintos valores de λ

```
lambdas <- c(2, 10, 30)
df_pois <- tibble(lambda = lambdas) |>
  rowwise() |>
  mutate(x = list(0:qpois(0.999, lambda)), prob =
    ↪ list(dpois(0:qpois(0.999, lambda), lambda))) |>
  unnest(cols = c(x, prob)) |>
  mutate(
    label = factor(paste0("lambda == ", lambda),
      levels = paste0("lambda == ", lambdas))
  )

df_pois |>
  ggplot(aes(x = x, y = prob)) +
  geom_bar(stat = "identity", width = 0.5, fill = myblue) +
  facet_wrap(~ label, nrow = 1, labeller = label_parsed) +
  labs(x = "X", y = "Probabilidad") +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))
```

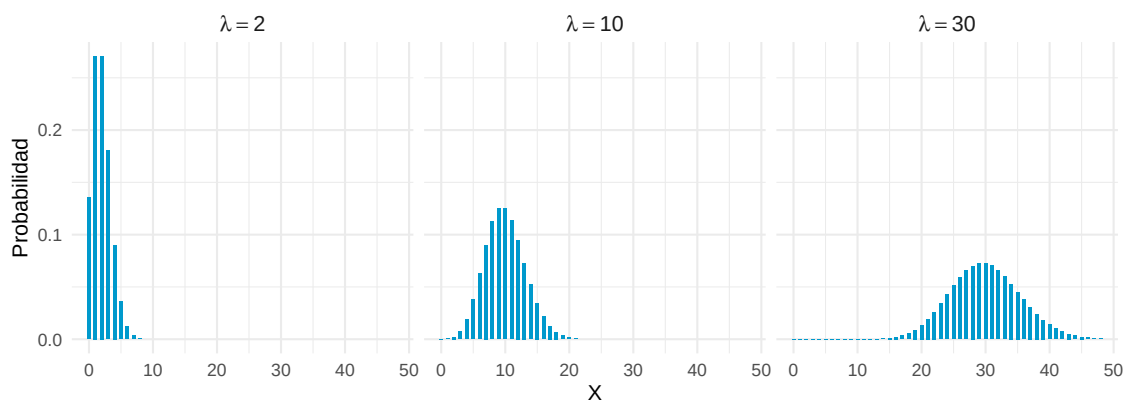


Figura 2.19: Forma de la distribución de Poisson para distintos valores de λ .

2.13.7. Resumen de distribuciones discretas

Distribución	Parámetros	Media	Varianza	Usos habituales
Bernoulli(p)	$p \in (0, 1)$	p	$p(1 - p)$	Resultado binario individual: test positivo/negativo, vivo/muerto, enfermo/sano.
Binomial(n, p)	$n \in \mathbb{N}, p \in (0, 1)$	np	$np(1 - p)$	Número de éxitos en n ensayos: respondedores a tratamiento, casos en cribado.
Geométrica(p)	$p \in (0, 1)$	$\frac{1 - p}{p}$	$\frac{1 - p}{p^2}$	Fracasos hasta el primer éxito: contactos hasta contagio, consultas hasta diagnóstico.
Binomial negativa(r, p)	$r > 0, p \in (0, 1)$	$\frac{r(1 - p)}{p}$	$\frac{r(1 - p)}{p^2}$	Fracasos hasta obtener r éxitos; conteos sobredispersos: recaídas, reingresos, eventos adversos con varianza mayor que la media.
Poisson(λ)	$\lambda > 0$	λ	λ	Conteos de eventos raros: casos semanales de enfermedad, llamadas a urgencias.
Hipergeométrica(N, K, n)	$N \in \mathbb{N}, K \in \mathbb{N}, n \in \mathbb{N}$	$\frac{nK}{N}$	$\frac{nK(N - K)(N - n)}{N^2(N - 1)}$	Muestreo sin reemplazamiento: prevalencia en poblaciones pequeñas.

2.14. Variables aleatorias continuas

Definición 2.22 (Función de densidad). La *función de densidad* de una variable aleatoria continua X es una función $f(x)$ que cumple las siguientes condiciones:

1. $f(x) \geq 0$ para todo $x \in \mathbb{R}$.
2. $\int_{-\infty}^{+\infty} f(x) dx = 1$.
3. $P(a \leq X \leq b) = \int_a^b f(x) dx$ para cualesquiera $a, b \in \mathbb{R}$ con $a \leq b$.

Ejemplo 2.28. La función

$$f(x) = \begin{cases} e^{-x} & x \geq 0 \\ 0 & x < 0 \end{cases}$$

es una función de densidad, ya que cumple las condiciones anteriores:

1. $f(x) \geq 0$ para todo $x \in \mathbb{R}$.
2. $\int_{-\infty}^{+\infty} f(x) dx = \int_0^{+\infty} e^{-x} dx = [-e^{-x}]_0^{+\infty} = 1$.
3. $P(a \leq X \leq b) = \int_a^b e^{-x} dx = [-e^{-x}]_a^b = e^{-a} - e^{-b}$ para $0 \leq a \leq b$.

2.14.1. Estadísticos de una variable aleatoria continua

La **media** de X es

$$E(X) = \int_{-\infty}^{+\infty} x f(x) dx,$$

y la **varianza**,

$$Var(X) = \int_{-\infty}^{+\infty} (x - E(X))^2 f(x) dx = \int_{-\infty}^{+\infty} x^2 f(x) dx - E(X)^2.$$

2.14.2. Distribución uniforme continua

Definición 2.23 (Distribución uniforme continua). Se dice que una variable aleatoria continua X sigue un *modelo de distribución uniforme continuo*, $X \sim U(a, b)$, si su rango es $\text{Ran}(X) = [a, b]$ y su función de densidad es constante en ese intervalo:

$$f(x) = \frac{1}{b-a} \quad \forall x \in [a, b],$$

donde a y b son el *mínimo* y el *máximo* de los valores que puede tomar la variable, respectivamente.

El modelo uniforme continuo se utiliza para modelar experimentos aleatorios en los que todos los valores del intervalo $[a, b]$ son *equiprobables*, como el tiempo de espera de un autobús que pasa a intervalos fijos.

Sus estadísticos son

$$E(X) = \frac{a+b}{2} \quad \text{y} \quad Var(X) = \frac{(b-a)^2}{12}.$$

Ejemplo 2.29. El tiempo de espera de un autobús que pasa cada 20 minutos sigue una distribución uniforme continua $U(0, 20)$.

```
df <- tibble(x = c(0, 0, 20, 20), p = c(0, 1/20, 1/20, 0))
# Diagrama de densidad de probabilidad
ggplot(df, aes(x = x, y = p)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  labs(x = "Tiempo de espera (minutos)", y = "Densidad") +
  theme_minimal()
```

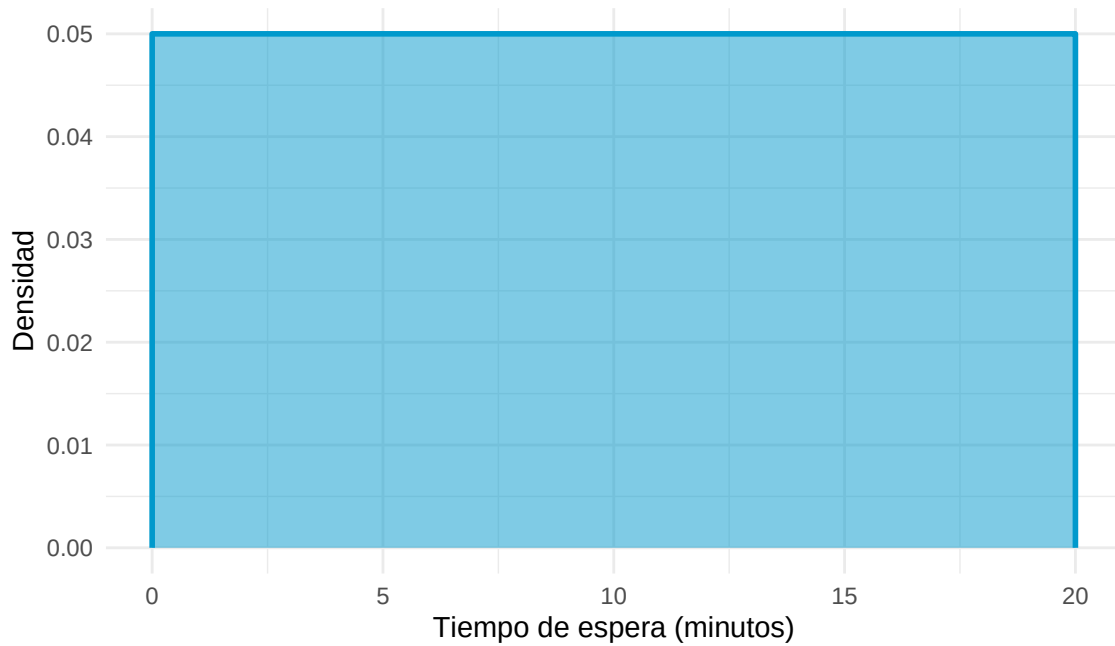


Figura 2.20: Distribución uniforme continua $U(0, 20)$.

2.14.3. Distribución beta

Definición 2.24 (Distribución beta). Se dice que una variable aleatoria continua X sigue un *modelo de distribución beta*, $X \sim \text{Beta}(\alpha, \beta)$, si su rango es $\text{Ran}(X) = [0, 1]$ y su función de densidad es

$$f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)} \quad x \in [0, 1],$$

donde $\alpha, \beta > 0$ son los parámetros de *forma* y $B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ es la función beta.

El modelo beta se utiliza para modelar proporciones o probabilidades, ya que su rango está acotado en $[0, 1]$. Según los valores de α y β puede tomar formas muy variadas (simétricas, asimétricas, en forma de U, etc.). Su media y su varianza son

$$E(X) = \frac{\alpha}{\alpha + \beta}, \quad \text{Var}(X) = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}.$$

i Nota

La **función gamma** es una generalización del factorial a los números reales positivos, definida como $\Gamma(\alpha) = \int_0^{+\infty} t^{\alpha-1} e^{-t} dt$. Para enteros positivos cumple $\Gamma(n) = (n-1)!$.

Ejemplo 2.30. La proporción de individuos que responden positivamente a un tratamiento puede modelarse con una distribución Beta(2, 5).

```
# Función de densidad Beta(2, 5)
df <- tibble(x = seq(0, 1, length.out = 300),
             p = dbeta(x, shape1 = 2, shape2 = 5)) # Densidad de
             ↪ probabilidad
# Diagrama de densidad de probabilidad
ggplot(df, aes(x = x, y = p)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  labs(x = "X", y = "Densidad") +
  theme_minimal()
```

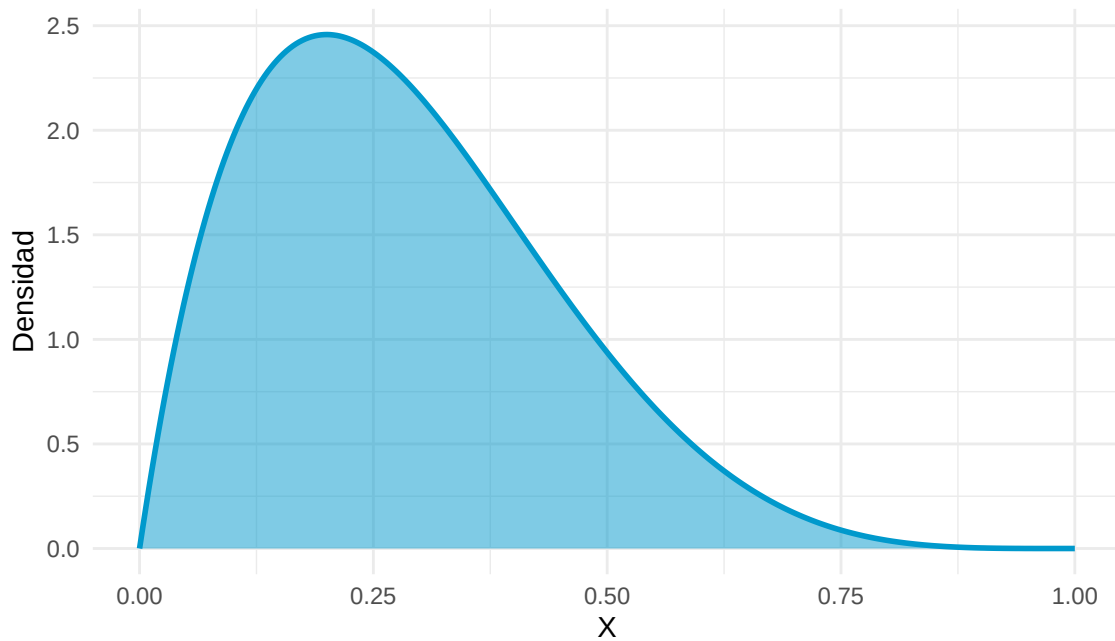


Figura 2.21: Distribución beta Beta(2, 5).

2.14.3.1. Forma de la distribución beta para distintos valores de α y β

```
beta_params <- expand.grid(alpha = c(1, 10, 100), beta = c(1, 10, 100))
df_beta <- pmap_dfr(beta_params, function(alpha, beta) {
  tibble(x = seq(0.001, 0.999, length.out = 300),
    y = dbeta(x, shape1 = alpha, shape2 = beta),
    label = factor(paste0("Beta(", alpha, ", ", beta, ")"),
      levels = paste0("Beta(", beta_params$alpha, ", ",
        ↪ ", beta_params$beta, ")"))))
})

df_beta |>
  ggplot(aes(x = x, y = y)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  facet_wrap(~ label, scales = "free_y", nrow = 3) +
  labs(x = "x", y = NULL) +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))
```

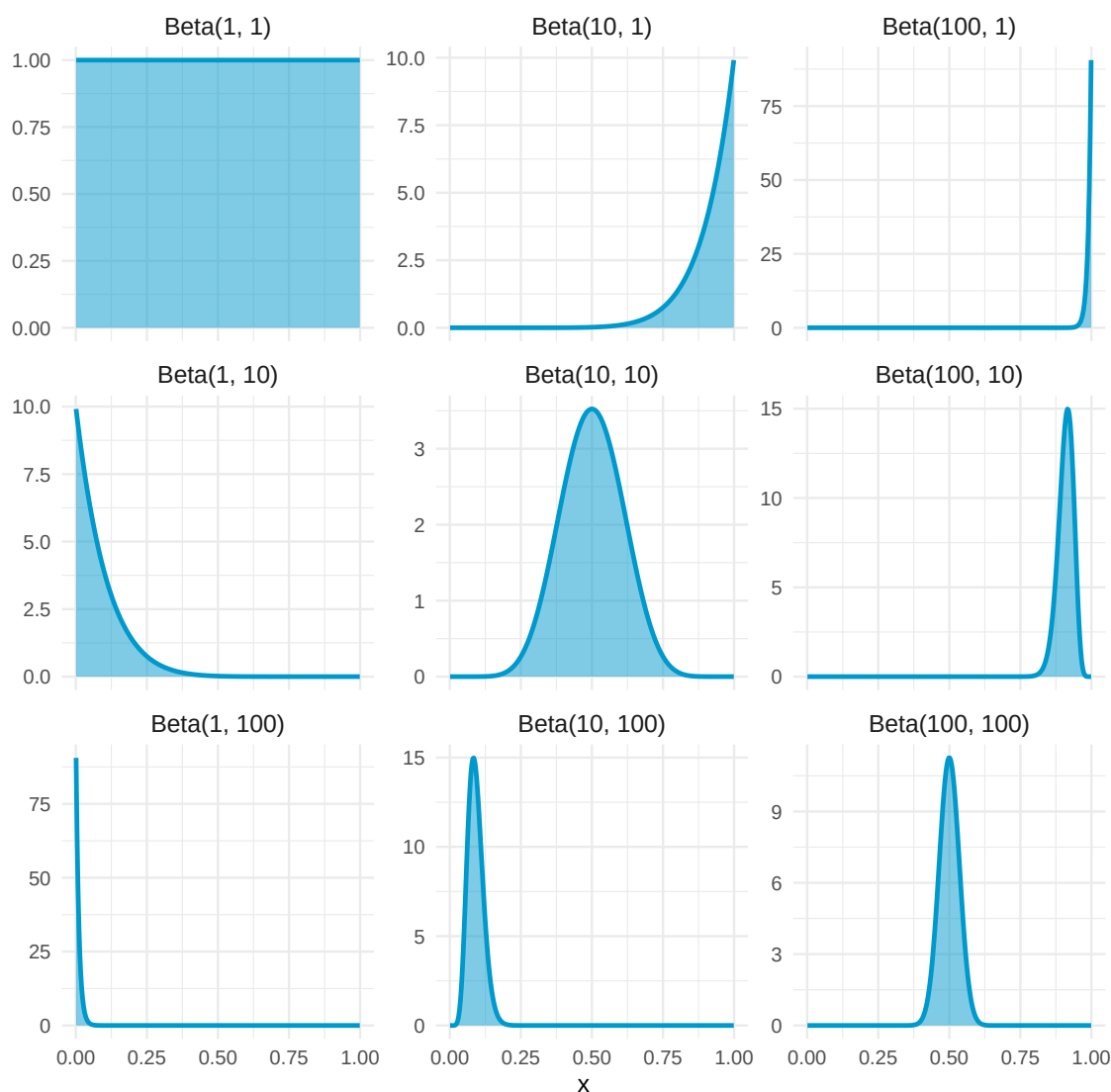


Figura 2.22: Forma de la distribución beta para distintos valores de α y β .

2.14.4. Distribución normal

Definición 2.25 (Distribución normal). Se dice que una variable aleatoria continua X sigue un *modelo de distribución normal*, $X \sim N(\mu, \sigma)$, si su rango es $\text{Ran}(X) = (-\infty, +\infty)$ y su función de densidad es

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad x \in \mathbb{R},$$

donde $\mu \in \mathbb{R}$ es la *media* y $\sigma > 0$ la *desviación típica*.

El modelo normal es el más importante de la estadística. Describe muchos fenómenos naturales y, por el *teorema central del límite*, es la distribución límite de la media de muestras grandes de casi cualquier distribución. Sus estadísticos son

$$E(X) = \mu \quad y \quad Var(X) = \sigma^2.$$

Ejemplo 2.31. La estatura de los adultos de una población suele seguir una distribución normal $N(170, 8)$ (en cm).

```
# Función de densidad N(170, 8)
df <- tibble(x = seq(140, 200, length.out = 300),
             p = dnorm(x, mean = 170, sd = 8)) # Densidad de
             ↪ probabilidad
# Diagrama de densidad de probabilidad
ggplot(df, aes(x = x, y = p)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  scale_x_continuous(breaks = seq(140, 200, by = 10)) +
  labs(x = "Estatura (cm)", y = "Densidad") +
  theme_minimal()
```

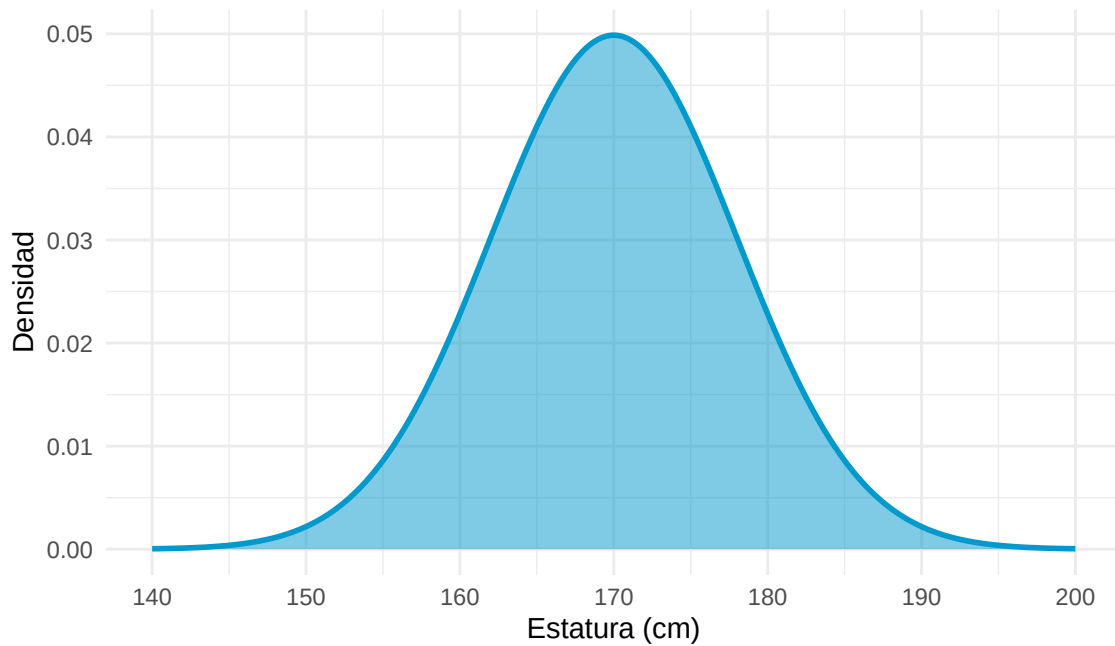


Figura 2.23: Distribución normal $N(170, 8)$.

2.14.4.1. Forma de la distribución normal para distintos valores de μ y σ

```
norm_params <- expand.grid(mu = c(0, 20, 100), sigma = c(1, 5, 10))
df_norm <- pmap_dfr(norm_params, function(mu, sigma) {
  tibble(x = seq(mu - 4*sigma, mu + 4*sigma, length.out = 300),
    y = dnorm(x, mean = mu, sd = sigma),
    mu = factor(paste0(" = ", mu), levels = paste0(" = ",
      ↪ c(0, 20, 100))),
    sigma = factor(paste0(" = ", sigma), levels = paste0(" = ",
      ↪ c(1, 5, 10))))
})

df_norm |>
  ggplot(aes(x = x, y = y)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  facet_grid(sigma ~ mu, scales = "free") +
  labs(x = "x", y = NULL) +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))
```

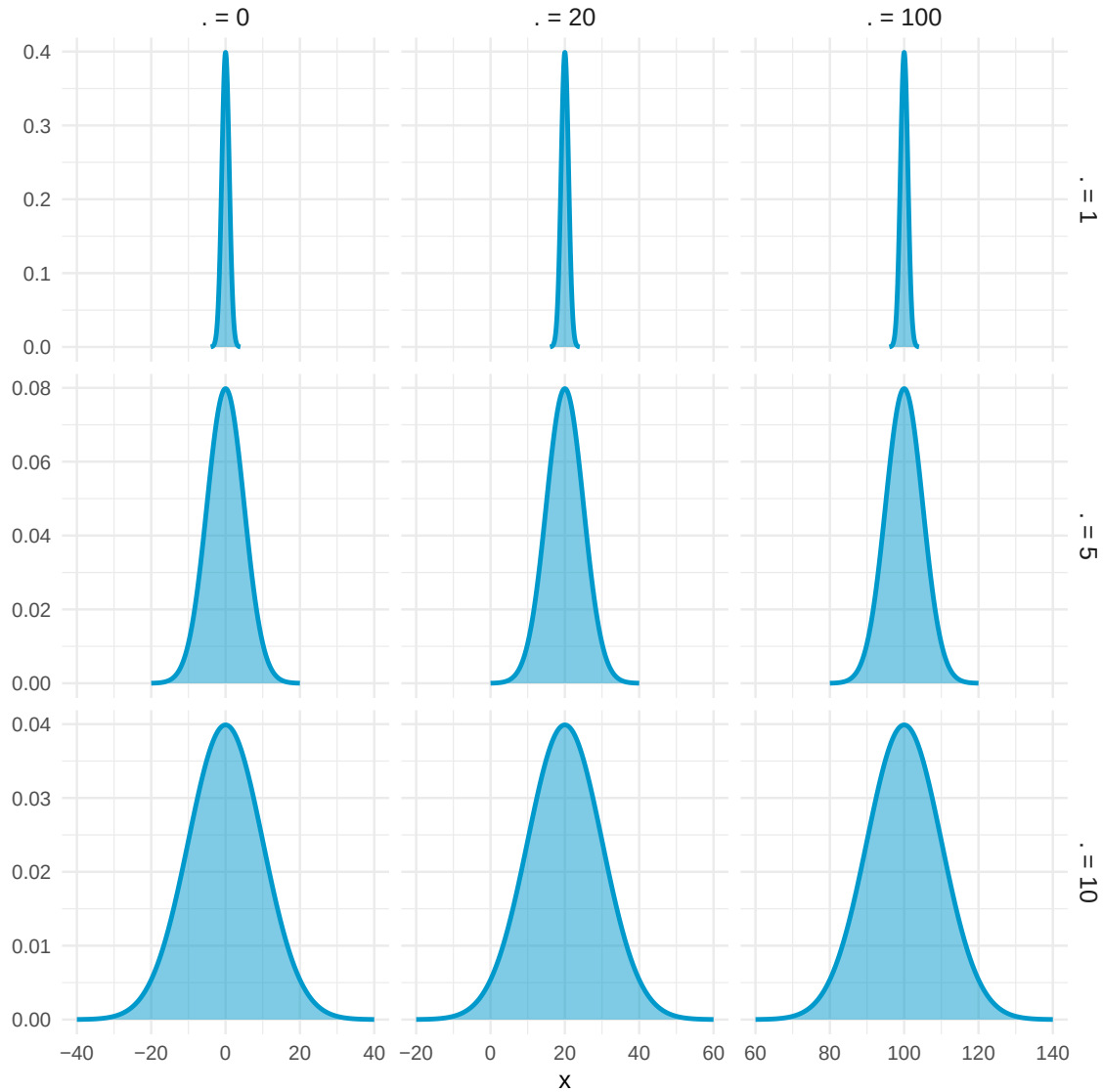


Figura 2.24: Forma de la distribución normal para distintos valores de μ y σ .

2.14.5. Distribución exponencial

Definición 2.26 (Distribución exponencial). Se dice que una variable aleatoria continua X sigue un *modelo de distribución exponencial*, $X \sim \text{Exp}(\lambda)$, si su rango es $\text{Ran}(X) = (0, +\infty)$ y su función de densidad es

$$f(x) = \lambda e^{-\lambda x} \quad x > 0,$$

donde $\lambda > 0$ es la *tasa de ocurrencia* del suceso.

El modelo exponencial se utiliza para modelar tiempos de espera o duraciones positivas, como el tiempo entre llamadas a una centralita o el tiempo de vida de un paciente. Sus estadísticos son

$$E(X) = \frac{1}{\lambda} \quad \text{y} \quad Var(X) = \frac{1}{\lambda^2}.$$

Ejemplo 2.32. Se sabe que la tasa de infecciones por un virus es de 0,2 por día. Entonces, el tiempo hasta que haya una infección sigue una distribución exponencial $Exp(0,2)$.

```
# Función de densidad Exp(0.2)
df <- tibble(x = seq(0, 30, length.out = 300),
             p = dexp(x, rate = 0.2)) # Densidad de probabilidad
# Diagrama de densidad de probabilidad
ggplot(df, aes(x = x, y = p)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  labs(x = "Tiempo hasta la infección (días)", y = "Densidad") +
  theme_minimal()
```

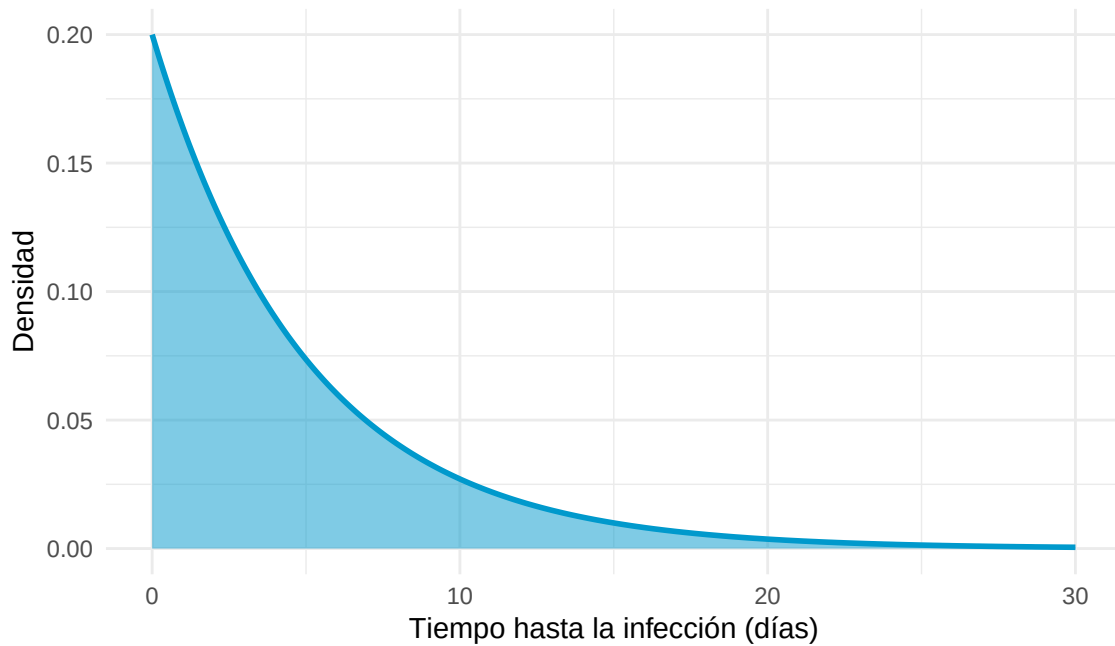


Figura 2.25: Distribución exponencial $Exp(0,2)$.

2.14.5.1. Forma de la distribución exponencial para distintos valores de λ

```
lambdas <- c(0.1, 0.3, 0.5)
df_exp <- tibble(lambda = lambdas) |>
  rowwise() |>
  mutate(x = list(seq(0, qexp(0.99, rate = lambda), length.out = 300)),
         y = list(dexp(x, rate = lambda))) |>
  unnest(cols = c(x, y)) |>
  mutate(label = factor(paste0(" = ", lambda), levels = paste0(" = ",
    ↪ lambdas)))

df_exp |>
  ggplot(aes(x = x, y = y)) +
  geom_line(color = myblue) +
  geom_area(fill = myblue, alpha = 0.5) +
  labs(x = "X", y = "Densidad") +
  facet_wrap(~ label, nrow = 1) +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))
```

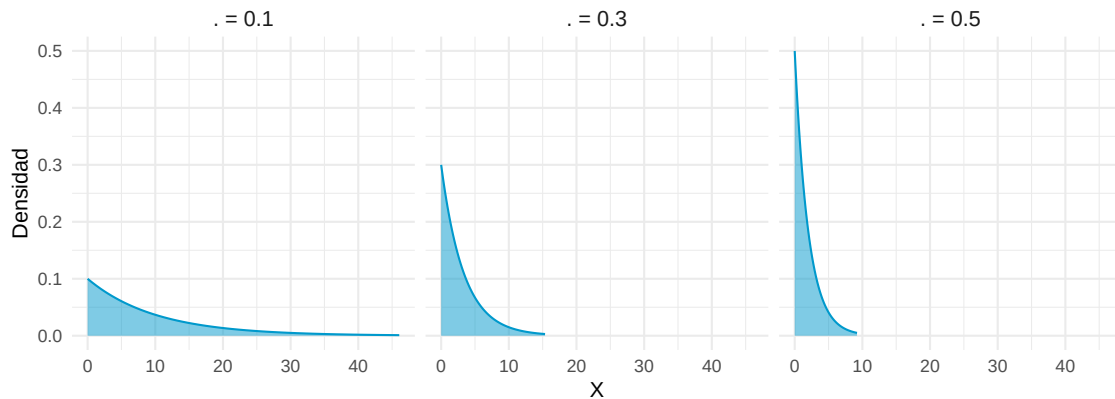


Figura 2.26: Forma de la distribución exponencial para distintos valores de λ .

2.14.6. Distribución gamma

Definición 2.27 (Distribución gamma). Se dice que una variable aleatoria continua X sigue un *modelo de distribución gamma*, $X \sim \text{Gamma}(\alpha, \beta)$, si su rango es $\text{Ran}(X) = (0, +\infty)$ y su función de densidad es

$$f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x} \quad x > 0,$$

donde $\alpha > 0$ es el parámetro de *forma*, $\beta > 0$ el parámetro de *tasa* y $\Gamma(\alpha)$ la función *gamma*.

El modelo gamma se utiliza para modelar tiempos de espera o duraciones positivas. Incluye como caso particular la distribución exponencial ($\alpha = 1$) y, para valores grandes de α , se aproxima a la normal. Sus estadísticos son

$$E(X) = \frac{\alpha}{\beta} \quad \text{y} \quad \text{Var}(X) = \frac{\alpha}{\beta^2}.$$

Ejemplo 2.33. El tiempo hasta el tercer contagio de un virus, si los contagios ocurren a una tasa de 2 por hora, sigue una distribución Gamma(3,2).

```
# Función de densidad Gamma(3, 2)
df <- tibble(x = seq(0, 5, length.out = 300),
             p = dgamma(x, shape = 3, rate = 2))
# Diagrama de densidad de probabilidad
ggplot(df, aes(x = x, y = p)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  labs(x = "Tiempo hasta el tercer contagio (horas)", y = "Densidad") +
  theme_minimal()
```

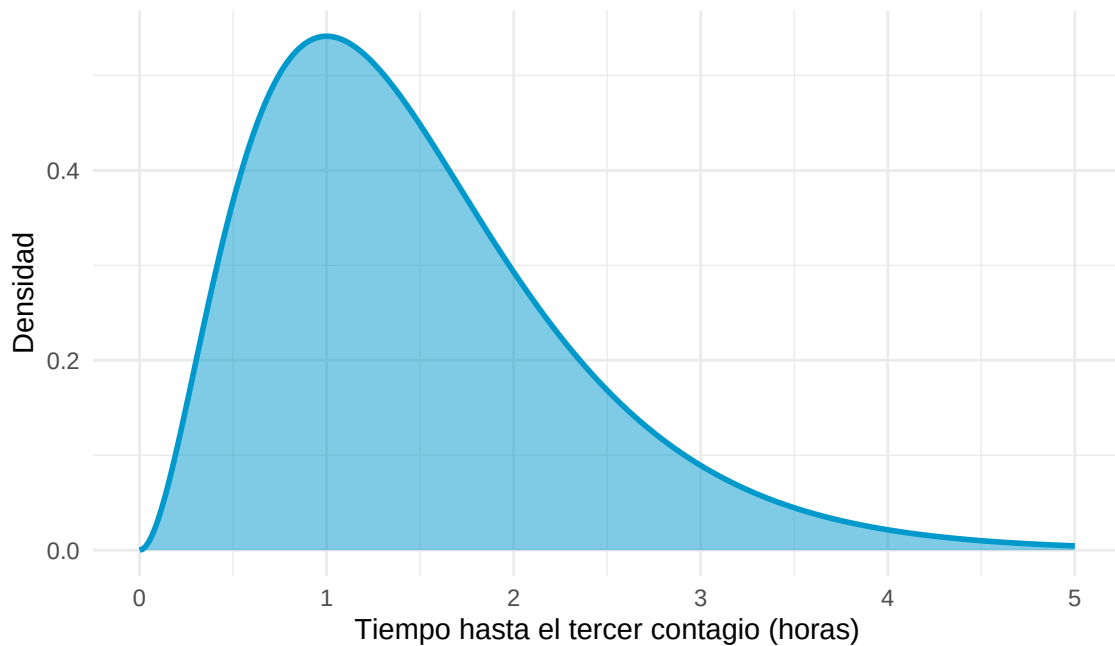


Figura 2.27: Distribución gamma Gamma(3,2).

2.14.6.1. Forma de la distribución gamma para distintos valores de α y β

```
library(ggh4x) # Para facet_grid2
alpha_vals <- c(1, 10, 30)
beta_vals <- c(0.1, 0.2, 0.3)

df_gamma <- expand_grid(alpha = alpha_vals, beta = beta_vals) |>
  pmap_dfr(function(alpha, beta) {
    x_max <- qgamma(0.9999, shape = alpha, rate = beta)
    tibble(x = seq(0.001, x_max, length.out = 300),
           y = dgamma(x, shape = alpha, rate = beta),
           alpha = factor(paste0(" = ", alpha), levels = paste0(" = ",
             ↪ alpha_vals)),
           beta = factor(paste0(" = ", beta), levels = paste0(" = ",
             ↪ beta_vals)))
  })

df_gamma |>
  ggplot(aes(x = x, y = y)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  ggh4x::facet_grid2(alpha ~ beta, scales = "free", independent = "y") +
  labs(x = "X", y = "Densidad") +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))
```

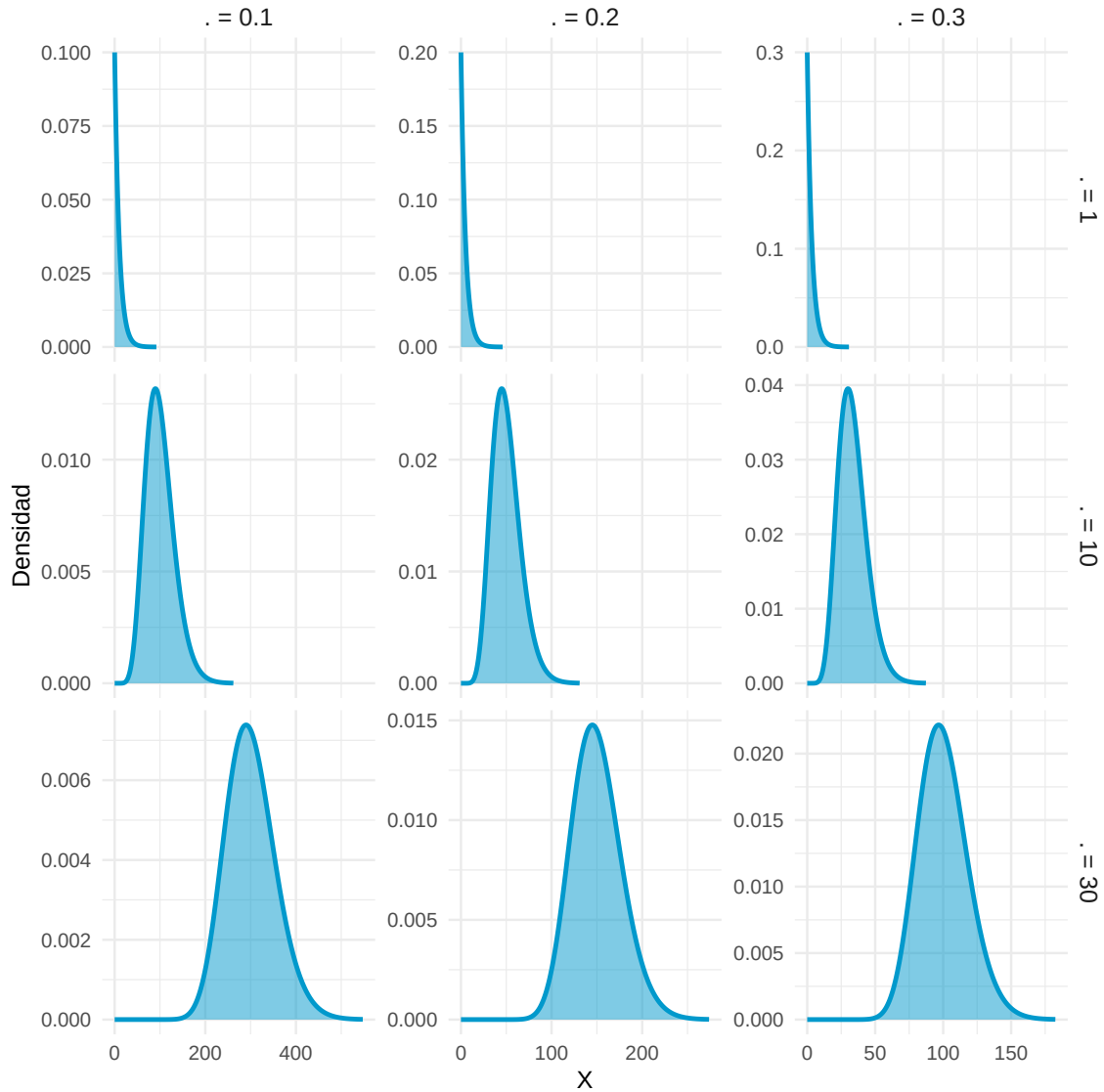


Figura 2.28: Forma de la distribución gamma para distintos valores de α y β .

2.14.7. Distribución gamma inversa

Definición 2.28 (Distribución gamma inversa). Se dice que una variable aleatoria continua X sigue un *modelo de distribución gamma inversa*, $X \sim \text{Gamma}^{-1}(\alpha, \beta)$, si su rango es $\text{Ran}(X) = (0, +\infty)$ y su función de densidad es

$$f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} e^{-\beta/x}, \quad x > 0,$$

donde $\alpha > 0$ es el parámetro de *forma*, $\beta > 0$ el parámetro de *escala* y $\Gamma(\alpha)$ la función *gamma*.

Si $X \sim \text{Gamma}(\alpha, \beta)$ (con β como tasa), entonces $1/X \sim \text{Gamma}^{-1}(\alpha, \beta)$. La distribución gamma inversa se utiliza habitualmente para modelar varianzas en estadística bayesiana. Sus estadísticos son

$$E(X) = \frac{\beta}{\alpha - 1} \quad \text{y} \quad \text{Var}(X) = \frac{\beta^2}{(\alpha - 1)^2(\alpha - 2)} \quad \text{para } \alpha > 2.$$

2.14.7.1. Forma de la distribución gamma inversa para distintos valores de α y β

```
library(extraDistr)
library(ggh4x) # Para facet_grid2
alpha_vals <- c(5, 10, 15)
beta_vals <- c(1, 5, 10)

df_gamma_inv <- expand.grid(alpha = alpha_vals, beta = beta_vals) |>
  pmap_dfr(function(alpha, beta) {
    x_max <- qinvgamma(0.99, alpha = alpha, beta = beta)
    tibble(x = seq(0.001, x_max, length.out = 300),
           y = dinvgamma(x, alpha = alpha, beta = beta),
           alpha = factor(paste0(" = ", alpha), levels = paste0(" = ",
             ↪ alpha_vals))),
           beta = factor(paste0(" = ", beta), levels = paste0(" = ",
             ↪ beta_vals)))
  })

df_gamma_inv |>
  ggplot(aes(x = x, y = y)) +
  geom_area(fill = myblue, alpha = 0.5) +
  geom_line(color = myblue, linewidth = 1) +
  ggh4x::facet_grid2(alpha ~ beta, scales = "free", independent = "y") +
  labs(x = "X", y = "Densidad") +
  theme_minimal() +
  theme(strip.text = element_text(size = 11))
```

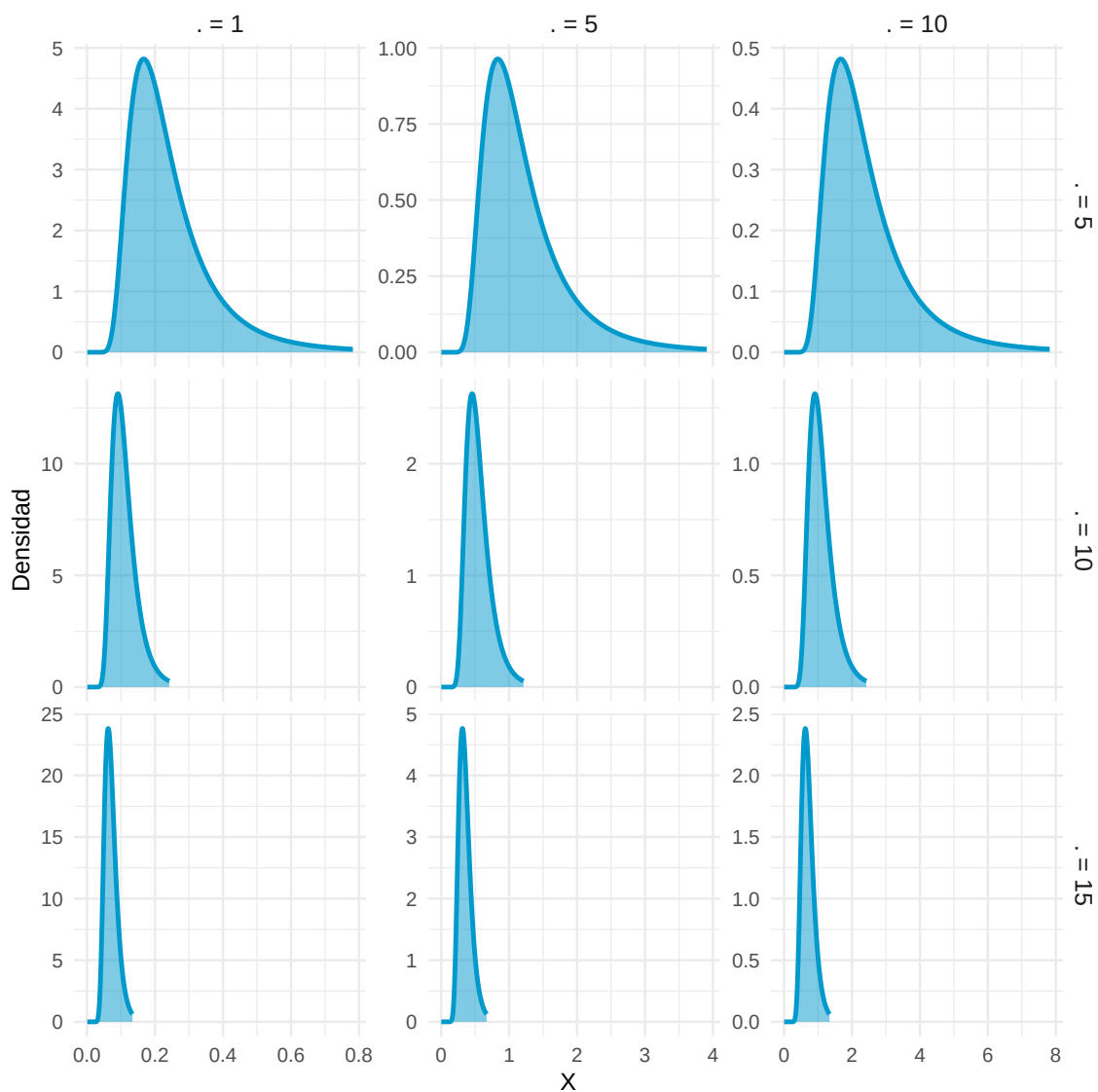


Figura 2.29: Forma de la distribución gamma inversa para distintos valores de α y β .

2.14.8. Resumen de distribuciones continuas

Distribución	Parámetros	Media	Varianza	Usos habituales
Uniforme(a, b)	$a < b$	$\frac{a + b}{2}$	$\frac{(b - a)^2}{12}$	Prior no informativo sobre parámetros acotados: probabilidades, proporciones.

Distribución	Parámetros	Media	Varianza	Usos habituales
Beta(α, β)	$\alpha, \beta > 0$	$\frac{\alpha}{\alpha + \beta}$	$\frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}$	Prior conjugado para proporciones: prevalencia, sensibilidad, especificidad.
Dirichlet($\alpha_1, \dots, \alpha_k$)	$\alpha_i > 0$	$\frac{\alpha_i}{\alpha_0}$	$\frac{\alpha_i(\alpha_0 - \alpha_i)}{\alpha_0^2(\alpha_0 + 1)}$	Prior conjugado para vectores de probabilidad: serotipos, mezclas de poblaciones.
Normal(μ, σ^2)	$\mu \in \mathbb{R}, \sigma^2 > 0$	μ	σ^2	Prior para medias y coeficientes: presión arterial, colesterol, efectos de tratamiento.
t de Student(ν, μ, σ^2)	$\nu > 0, \mu \in \mathbb{R}, \sigma^2 > 0$	μ ($\nu > 1$)	$\frac{\nu\sigma^2}{\nu - 2}$ ($\nu > 2$)	Prior robusto con colas pesadas; marginal de μ en el modelo normal-gamma inversa.
Cauchy(μ, σ)	$\mu \in \mathbb{R}, \sigma > 0$	no existe	no existe	Prior de colas muy pesadas para coeficientes en regresión logística.
Exponencial(λ)	$\lambda > 0$	$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$	Prior débilmente informativo para σ y τ ; tiempo entre eventos con riesgo constante.
Gamma(α, β)	$\alpha, \beta > 0$	$\frac{\alpha}{\beta}$	$\frac{\alpha}{\beta^2}$	Prior conjugado para tasas Poisson: incidencia, mortalidad.
Gamma inversa(α, β)	$\alpha, \beta > 0$	$\frac{\beta}{\alpha - 1}$ ($\alpha > 1$)	$\frac{\beta^2}{(\alpha - 1)^2(\alpha - 2)}$ ($\alpha > 2$)	Prior conjugado para varianzas σ^2 ; varianza de efectos aleatorios.
Lognormal(μ, σ^2)	$\mu \in \mathbb{R}, \sigma^2 > 0$	$e^{\mu + \sigma^2/2}$	$(e^{\sigma^2} - 1)e^{2\mu + \sigma^2}$	Parámetros positivos asimétricos: dosis, concentraciones, tiempos de supervivencia.
Weibull(k, λ)	$k, \lambda > 0$	$\lambda \Gamma(1 + \frac{1}{k})$	$\lambda^2 \left[\Gamma(1 + \frac{2}{k}) - \Gamma(1 + \frac{1}{k})^2 \right]$	Modelos de supervivencia: tiempo hasta la muerte, recaída o fracaso de tratamiento.

Ejercicio 2.4. ¿Qué tipo de distribución de probabilidad utilizarías para modelar las siguientes variables aleatorias?

- a. Número de casos de gripe por semana.
- b. Eficacia de una vacuna.
- c. Tiempo entre hospitalizaciones de un paciente con insuficiencia cardíaca.
- d. Número de lesiones de acné por dm^2 de piel en una persona.
- e. Nivel de glucosa en sangre de un paciente diabético.
- f. Número de contactos hasta el primer contagio en una cadena de transmisión.
- g. Tasa de incidencia de la gripe por semana.

Solución

- a. Poisson.
- b. Beta.
- c. Exponencial.
- d. Poisson.
- e. Normal.
- f. Geométrica.
- g. Gamma.

2.15. Muestreo

El proceso de obtener valores de una variable aleatoria mediante la realización de un experimento aleatorio se conoce como *muestreo*. Cuando la selección de los valores se hace de manera independiente y con la misma distribución, se dice que se ha realizado un *muestreo aleatorio simple*.

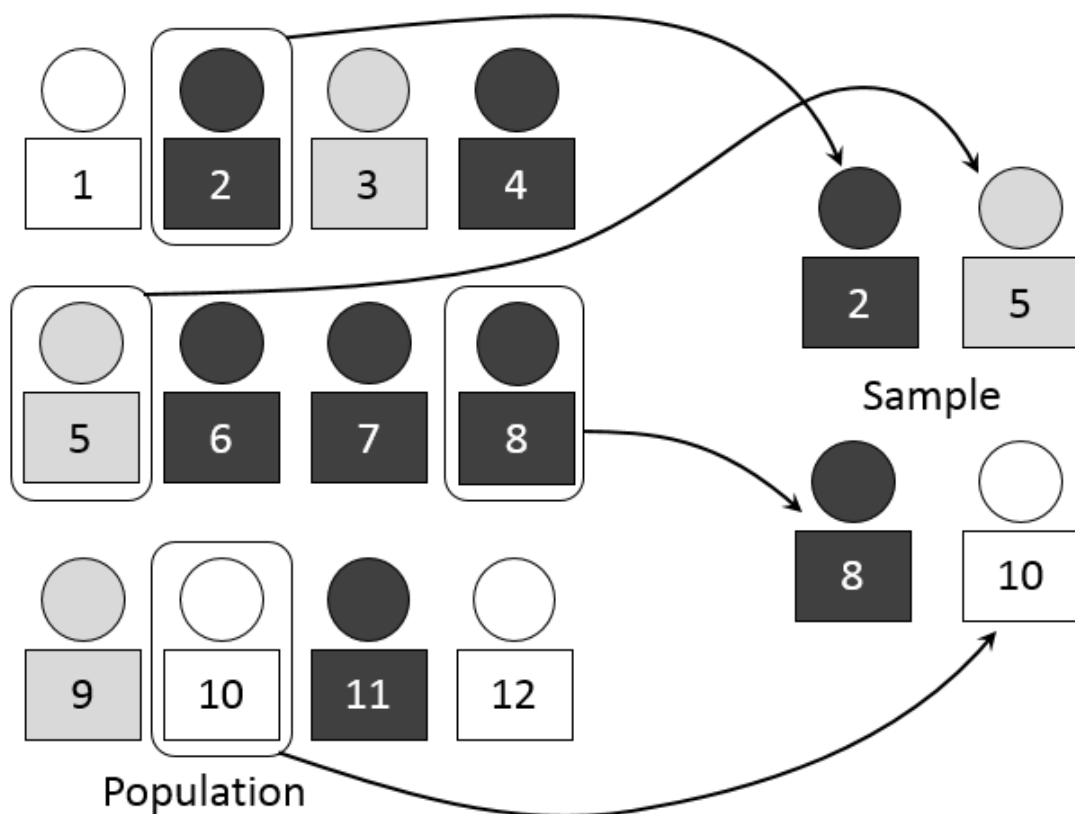


Figura 2.30: Muestreo aleatorio simple (Fuente: [Wikimedia Commons](#), CC BY-SA 4.0, Dan Kernler).

2.15.1. Muestreo en variables aleatorias discretas

Para realizar un muestreo aleatorio simple de una variable aleatoria discreta con un número finito de valores se puede usar la función `sample` de R.

i `sample`

Devuelve una muestra aleatoria de los elementos de un vector dado, con o sin reemplazamiento, y con probabilidades asociadas a cada elemento.

Parámetros:

- **x**: vector de valores a muestrear.
- **size**: número de valores a muestrear.
- **prob**: vector de probabilidades asociadas a cada valor de **x**. Si no se especifica, se asume que todos los valores tienen la misma probabilidad.

- `replace`: TRUE indica que el muestreo es con reemplazo, mientras que FALSE implica que el muestreo es sin reemplazo.

Ejemplo 2.34. El siguiente código realiza un muestreo aleatorio simple de tamaño 10 del resultado de lanzar una moneda equilibrada, es decir, con probabilidades iguales para cara y cruz.

```
set.seed(123) # Para reproducibilidad
# Valores posibles: Cara ("C") y Cruz ("X")
x <- c("C", "X")
# Probabilidades asociadas a cada valor
p <- c(0.5, 0.5)
# Muestreo aleatorio simple de tamaño 10
sample(x, size = 10, prob = p, replace = TRUE)
```

```
[1] "X" "C" "X" "C" "C" "X" "C" "C" "C" "X"
```

Si la moneda estuviera trucada, por ejemplo con una probabilidad de cara de 0,8 y de cruz de 0,2, se tendría:

```
# Probabilidades asociadas a cada valor
p <- c(0.8, 0.2)
# Muestreo aleatorio simple de tamaño 10
sample(x, size = 10, prob = p, replace = TRUE)
```

```
[1] "X" "C" "C" "C" "C" "X" "C" "C" "C" "X"
```

Otra opción para tomar muestras aleatorias de un data frame es la función `slice_sample` del paquete `dplyr`.

i slice_sample

Devuelve una muestra aleatoria de tamaño determinado de un data frame o tibble.
Parámetros:

- `.data`: data frame o tibble del que se quiere tomar la muestra.
- `n`: número de filas a muestrear.
- `weight_by`: nombre de la columna que contiene las probabilidades asociadas a cada fila. Si no se especifica, se asume que todas las filas tienen la misma probabilidad.
- `replace`: TRUE indica que el muestreo es con reemplazo, mientras que FALSE

implica que el muestreo es sin reemplazo.

El mismo ejemplo anterior puede replicarse con este código:

```
set.seed(123)
tibble(x = c("C", "X"), p = c(0.8, 0.2)) |>
  slice_sample(n = 10, weight_by = p, replace = TRUE)
```

```
# A tibble: 10 x 2
```

	x	p
	<chr>	<dbl>
1	C	0.8
2	C	0.8
3	C	0.8
4	X	0.2
5	X	0.2
6	C	0.8
7	C	0.8
8	X	0.2
9	C	0.8
10	C	0.8

2.15.2. Muestreo en el modelo de distribución binomial

Para realizar un muestreo aleatorio simple de una variable aleatoria con distribución binomial se puede usar la función `rbinom` de R.

i `rbinom`

Devuelve una muestra aleatoria de una distribución binomial.

Parámetros:

- **n**: tamaño de la muestra.
- **size**: número de repeticiones o ensayos.
- **prob**: probabilidad de éxito para cada ensayo.

Ejemplo 2.35. Se sabe que el porcentaje de fumadores en una población es del 20 %. Vamos a realizar un muestreo aleatorio simple de tamaño 20 del número de fumadores en muestras aleatorias de 10 personas.

```

set.seed(123) # Para reproducibilidad
n <- 10 # Número de repeticiones
p <- 0.2 # Probabilidad de éxito (fumador)
# Muestreo aleatorio simple de la distribución binomial
df <- tibble(x = rbinom(n = 20, size = n, prob = p))
df$x

```

```
[1] 1 3 2 4 4 0 2 4 2 2 4 2 2 2 0 4 1 0 1 4
```

```

# Distribución binomial teórica
df_binom <- tibble(x = 0:n, p = dbinom(0:n, size = n, prob = p))
ggplot(df, aes(x = x)) +
  # Diagrama de barras de la distribución muestral
  geom_bar(aes(y = after_stat(count / sum(count))), fill = myblue, width
    ↪ = 0.2) +
  # Distribución de probabilidad del modelo binomial
  geom_point(data = df_binom, aes(x = x, y = p), color = myred, size =
    ↪ 2) +
  scale_x_continuous(breaks = 0:n) +
  labs(x = "Número de fumadores", y = "Frecuencia relativa /
    ↪ Probabilidad") +
  theme_minimal()

```

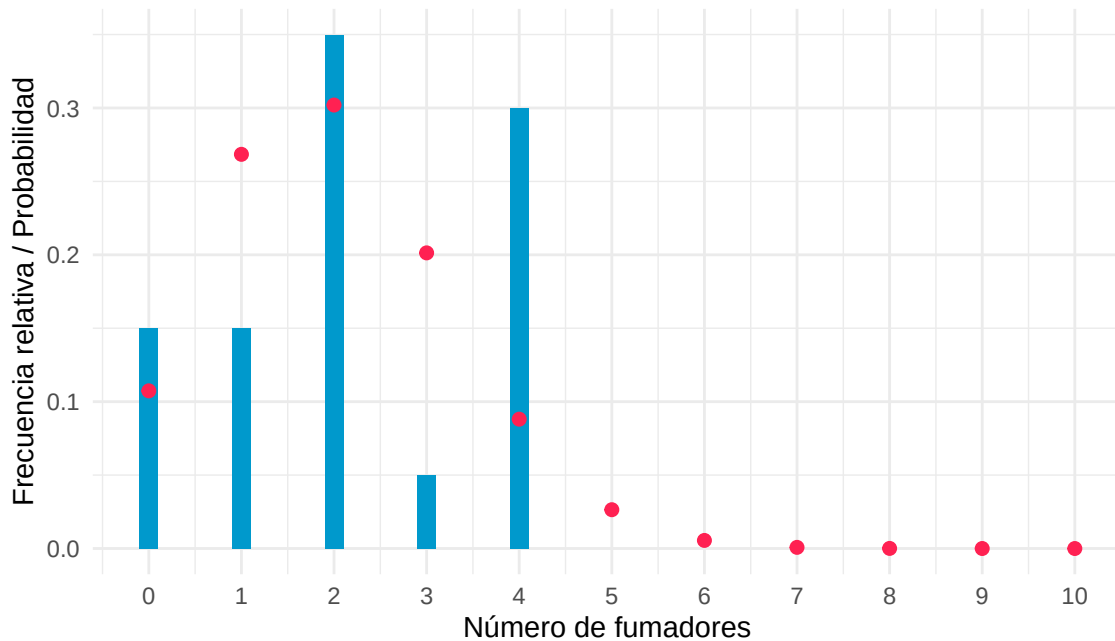


Figura 2.31: Muestra aleatoria de tamaño 20 de la distribución $B(10,0,2)$ frente al modelo teórico.

Para tamaños muestrales pequeños, la distribución de frecuencias muestral es muy diferente a la distribución de probabilidad del modelo binomial $B(10,0,2)$. Pero para tamaños muestrales grandes, la distribución de frecuencias del muestreo se aproxima a la distribución de probabilidad del modelo:

```
df <- tibble(x = rbinom(n = 10000, size = n, prob = p))
# Distribución binomial teórica
df_binom <- tibble(x = 0:n, p = dbinom(0:n, size = n, prob = p))
ggplot(df, aes(x = x)) +
  geom_bar(aes(y = after_stat(count / sum(count))), fill = myblue, width
    ↪ = 0.2) +
  geom_point(data = df_binom, aes(x = x, y = p), color = myred, size =
    ↪ 2) +
  scale_x_continuous(breaks = 0:n) +
  labs(x = "Número de fumadores", y = "Frecuencia relativa /
    ↪ Probabilidad") +
  theme_minimal()
```

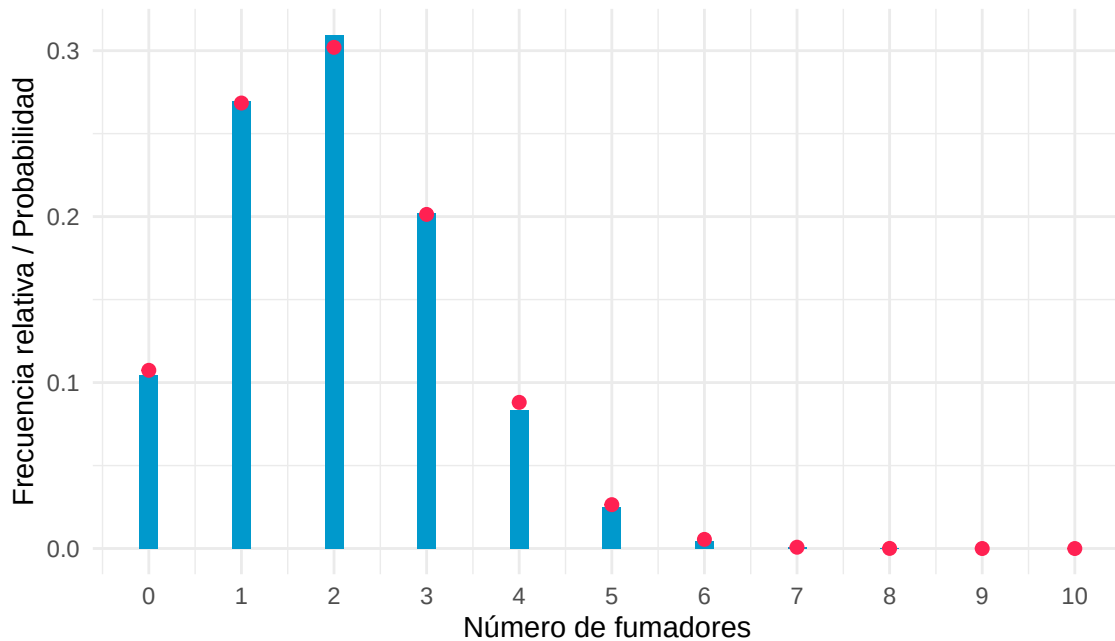


Figura 2.32: Muestra aleatoria de tamaño 10000 de la distribución $B(10, 0.2)$ frente al modelo teórico.

2.15.3. Muestreo de la distribución geométrica

Para realizar un muestreo aleatorio simple de una variable aleatoria con distribución geométrica se puede usar la función `rgeom` de R, cuyos parámetros son `n` (tamaño de la muestra) y `prob` (probabilidad de éxito en cada ensayo).

Ejemplo 2.36. La probabilidad de que un paciente responda favorablemente a un tratamiento es del 30 %. Tomamos una muestra aleatoria de tamaño 10000 del número de pacientes no respondedores tratados hasta obtener el primer éxito.

```
set.seed(123) # Para reproducibilidad
p <- 0.3 # Probabilidad de éxito
# Muestreo aleatorio simple de la distribución geométrica
df <- tibble(x = rgeom(n = 10000, prob = p))
# Distribución geométrica teórica
x_max <- max(df$x)
df_geom <- tibble(x = 0:x_max, p = dgeom(0:x_max, prob = p))
ggplot(df, aes(x = x)) +
  geom_bar(aes(y = after_stat(count / sum(count))), fill = myblue, width
    ↪ = 0.2) +
  geom_point(data = df_geom, aes(x = x, y = p), color = myred, size = 2)
    ↪ +
```

```
scale_x_continuous(breaks = 0:x_max) +
labs(x = "Número de fracasos antes del primer éxito", y = "Frecuencia
↪ relativa / Probabilidad") +
theme_minimal()
```

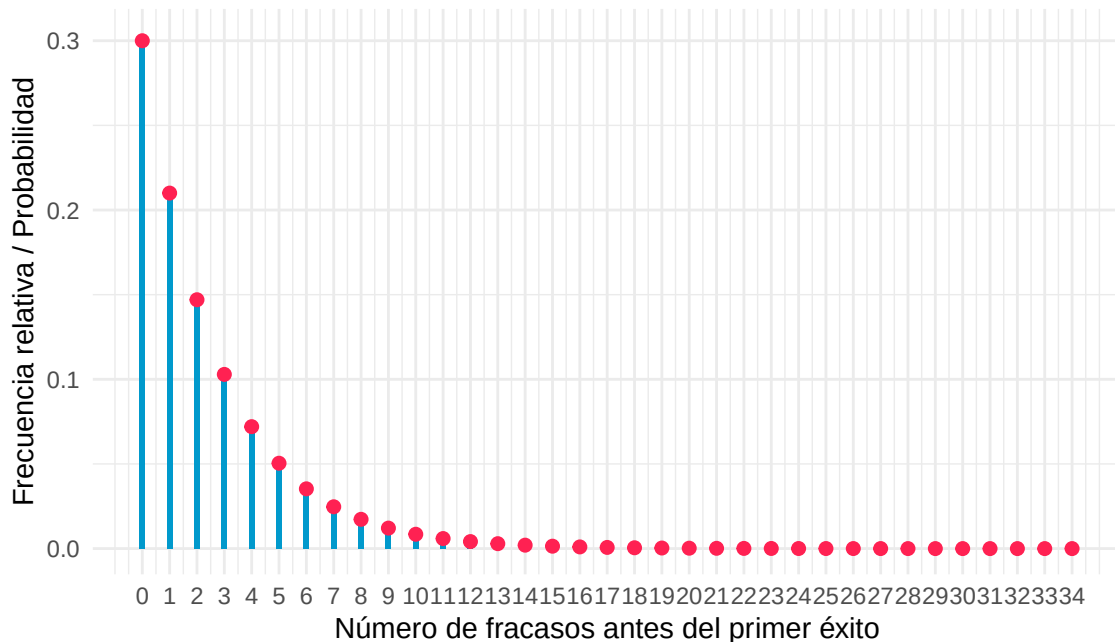


Figura 2.33: Muestra aleatoria de la distribución $G(0,3)$ frente al modelo teórico.

2.15.4. Muestreo de la distribución de Poisson

Para realizar un muestreo aleatorio simple de una variable aleatoria con distribución de Poisson se puede usar la función `rpois` de R, cuyos parámetros son `n` (tamaño de la muestra) y `lambda` (número medio de ocurrencias).

Ejemplo 2.37. El número medio de infartos que se producen diariamente en una ciudad es 5. Tomamos una muestra aleatoria de tamaño 10000 del número de infartos diarios.

```
set.seed(123)
lambda <- 5 # Media
df <- tibble(x = rpois(n = 10000, lambda = lambda))
x_max <- max(df$x)
df_pois <- tibble(x = 0:x_max, p = dpois(0:x_max, lambda = lambda))
ggplot(df, aes(x = x)) +
  geom_bar(aes(y = after_stat(count / sum(count))), fill = myblue, width
↪ = 0.2) +
```

```
geom_point(data = df_pois, aes(x = x, y = p), color = myred, size = 2)
  ↪ +
scale_x_continuous(breaks = 0:x_max) +
labs(x = "Número de infartos", y = "Frecuencia relativa /
  ↪ Probabilidad") +
theme_minimal()
```

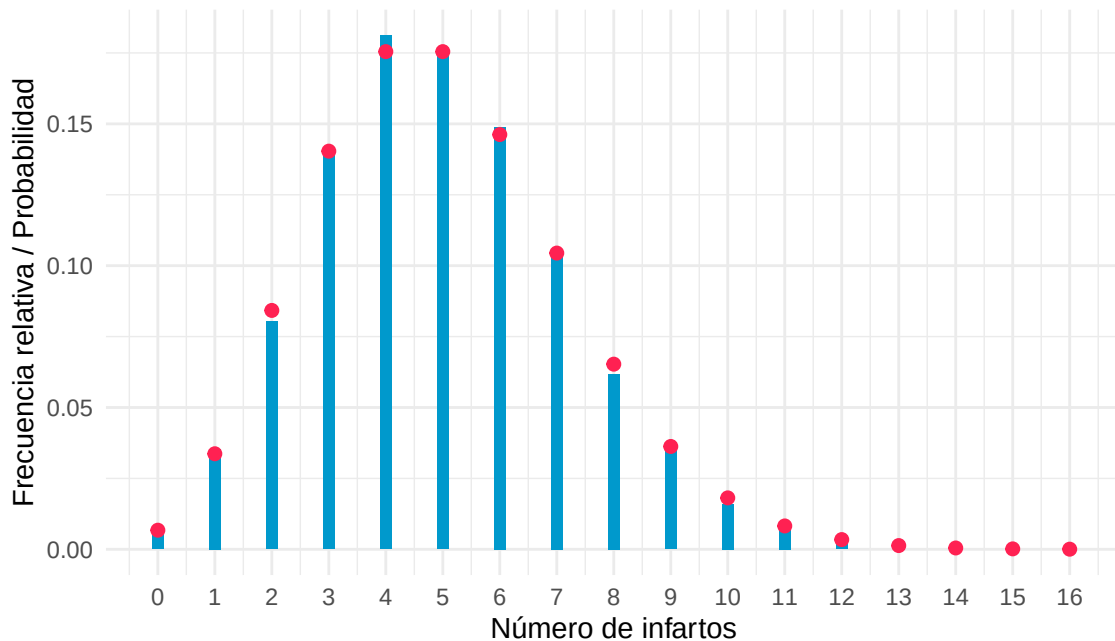


Figura 2.34: Muestra aleatoria de la distribución $P(5)$ frente al modelo teórico.

2.15.5. Muestreo de la distribución uniforme continua

Para muestrear una variable con distribución uniforme continua se usa la función `runif`, con los parámetros `n`, `min` y `max`.

Ejemplo 2.38. El tiempo de espera de un autobús que pasa cada 20 minutos sigue una distribución $U(0, 20)$. Tomamos una muestra aleatoria de tamaño 10000 del tiempo de espera.

```
set.seed(123)
df <- tibble(x = runif(n = 10000, min = 0, max = 20))
ggplot(df, aes(x = x)) +
  geom_area(stat = "density", fill = myblue, alpha = 0.5, bounds = c(0,
  ↪ 20)) +
```

```
geom_line(stat = "density", color = myblue, linewidth = 1, bounds =
  ↪ c(0, 20)) +
geom_hline(yintercept = 1/20, color = myred, linewidth = 1) +
labs(x = "Tiempo de espera (minutos)", y = "Densidad") +
theme_minimal()
```

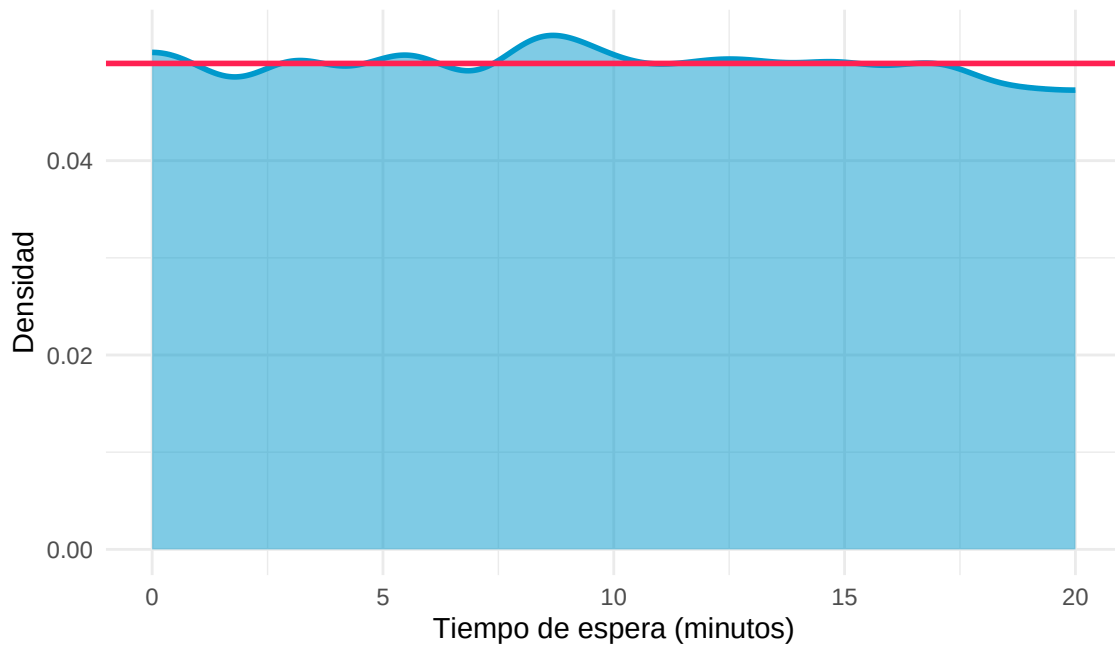


Figura 2.35: Muestra aleatoria de la distribución $U(0, 20)$ frente al modelo teórico.

2.15.6. Muestreo de la distribución beta

Para muestrear una variable con distribución beta se usa la función `rbeta`, con los parámetros `n`, `shape1` (α) y `shape2` (β).

Ejemplo 2.39. La proporción de individuos que responden positivamente a un tratamiento puede modelarse con una distribución Beta(2, 5). Tomamos una muestra aleatoria de tamaño 10000 de esta distribución.

```
set.seed(123)
n <- 10000 # Tamaño de la muestra
shape1 <- 2 # Parámetro de forma
shape2 <- 5 # Parámetro de forma
# Muestreo aleatorio simple de la distribución beta
df <- tibble(x = rbeta(n = n, shape1 = shape1, shape2 = shape2))
```

```
# Distribución beta teórica
df_beta <- tibble(x = seq(0, 1, length.out = 300),
                  p = dbeta(x, shape1 = shape1, shape2 = shape2))
ggplot(df, aes(x = x)) +
  geom_area(stat = "density", fill = myblue, alpha = 0.5) +
  geom_line(stat = "density", color = myblue, linewidth = 1) +
  geom_line(data = df_beta, aes(x = x, y = p), color = myred, linewidth
    ↪ = 1) +
  labs(x = "Proporción de respuestas positivas", y = "Densidad") +
  theme_minimal()
```

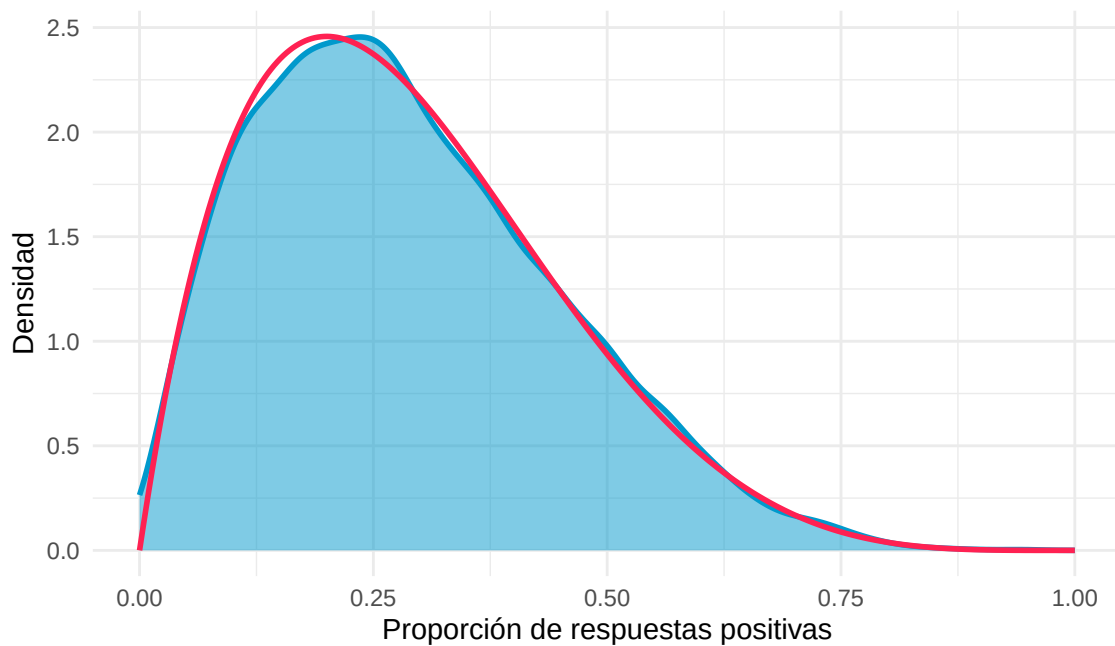


Figura 2.36: Muestra aleatoria de la distribución Beta(2,5) frente al modelo teórico.

2.15.7. Muestreo de la distribución normal

Para muestrear una variable con distribución normal se usa la función `rnorm`, con los parámetros `n`, `mean` (μ) y `sd` (σ).

Ejemplo 2.40. La estatura de los adultos de una población suele seguir una distribución $N(170,8)$ (en cm). Tomamos una muestra aleatoria de tamaño 10000 de esta distribución.

```

set.seed(123)
n <- 10000 # Tamaño de la muestra
mean <- 170 # Media
sd <- 8    # Desviación típica
# Muestreo aleatorio simple de la distribución normal
df <- tibble(x = rnorm(n = n, mean = mean, sd = sd))
# Distribución normal teórica
df_norm <- tibble(x = seq(140, 200, length.out = 300),
                  p = dnorm(x, mean = mean, sd = sd))
ggplot(df, aes(x = x)) +
  geom_area(stat = "density", fill = myblue, alpha = 0.5) +
  geom_line(stat = "density", color = myblue, linewidth = 1) +
  geom_line(data = df_norm, aes(x = x, y = p), color = myred, linewidth
    ↪ = 1) +
  scale_x_continuous(breaks = seq(140, 200, by = 10)) +
  labs(x = "Estatura (cm)", y = "Densidad") +
  theme_minimal()

```

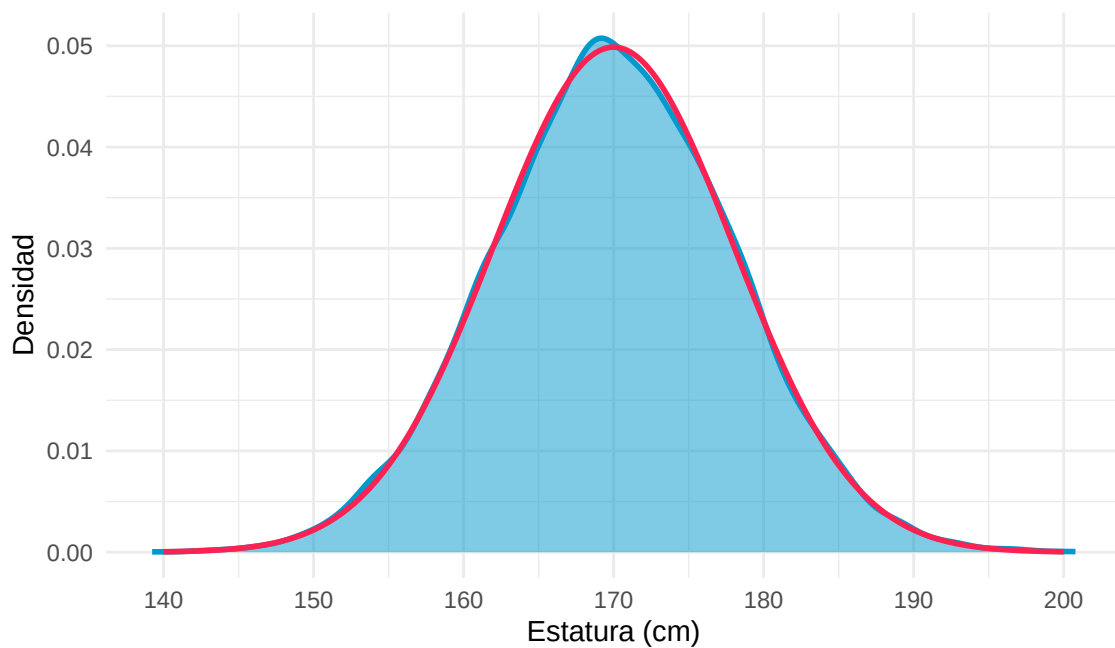


Figura 2.37: Muestra aleatoria de la distribución $N(170, 8)$ frente al modelo teórico.

2.15.8. Muestreo de la distribución exponencial

Para muestrear una variable con distribución exponencial se usa la función `rexp`, con los parámetros `n` y `rate` (λ).

Ejemplo 2.41. El tiempo hasta la infección por un virus sigue una distribución $Exp(0,2)$. Tomamos una muestra aleatoria de tamaño 10000 de esta distribución.

```
set.seed(123)
n <- 10000 # Tamaño de la muestra
rate <- 0.2 # Tasa de ocurrencia
# Muestreo aleatorio simple de la distribución exponencial
df <- tibble(x = rexp(n = n, rate = rate))
# Distribución exponencial teórica
x_max <- qexp(0.999, rate = rate)
df_exp <- tibble(x = seq(0, x_max, length.out = 300), p = dexp(x, rate
  ↪ = rate))
ggplot(df, aes(x = x)) +
  geom_area(stat = "density", fill = myblue, alpha = 0.5, bounds = c(0,
    ↪ x_max)) +
  geom_line(stat = "density", color = myblue, linewidth = 1, bounds =
    ↪ c(0, x_max)) +
  geom_line(data = df_exp, aes(x = x, y = p), color = myred, linewidth =
    ↪ 1) +
  labs(x = "Tiempo hasta la infección (días)", y = "Densidad") +
  theme_minimal()
```

Warning: Some data points are outside of `bounds`. Removing them.
Some data points are outside of `bounds`. Removing them.

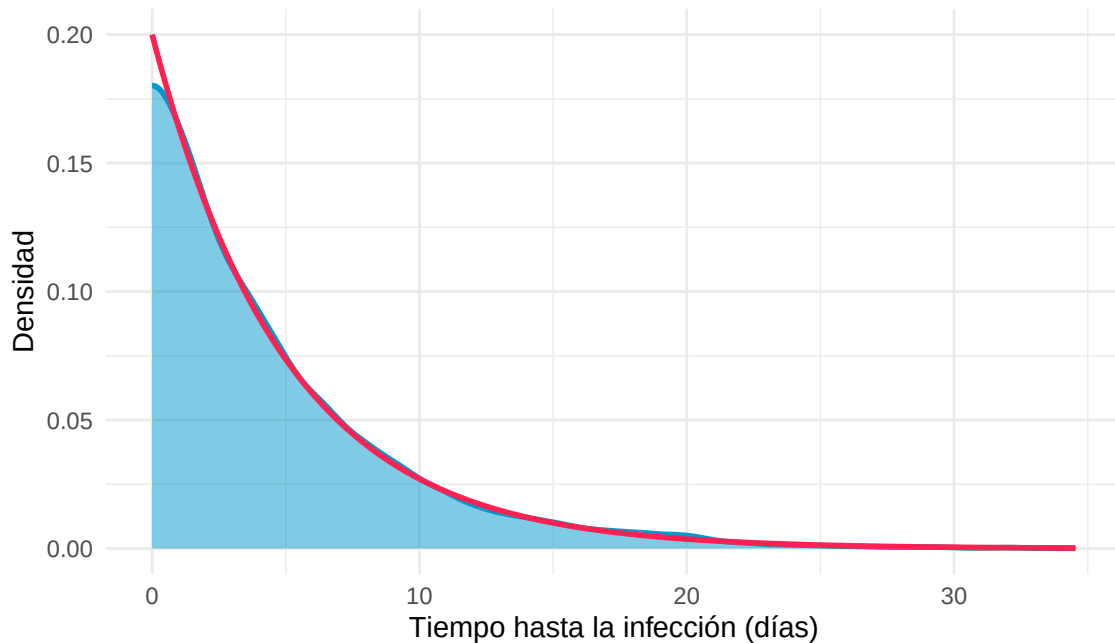


Figura 2.38: Muestra aleatoria de la distribución $Exp(0,2)$ frente al modelo teórico.

2.15.9. Muestreo de la distribución gamma

Para muestrear una variable con distribución gamma se usa la función `rgamma`, con los parámetros `n`, `shape` (α) y `rate` (β).

Ejemplo 2.42. El tiempo hasta el tercer contagio de un virus, si los contagios ocurren a una tasa de 2 por hora, sigue una distribución $\text{Gamma}(3,2)$. Tomamos una muestra aleatoria de tamaño 10000 de esta distribución.

```
set.seed(123)
n <- 10000 # Tamaño de la muestra
shape <- 3 # Parámetro de forma
rate <- 2 # Parámetro de tasa
# Muestreo aleatorio simple de la distribución gamma
df <- tibble(x = rgamma(n = n, shape = shape, rate = rate))
# Distribución gamma teórica
x_max <- qgamma(0.999, shape = shape, rate = rate)
df_gamma <- tibble(x = seq(0, x_max, length.out = 300),
                  p = dgamma(x, shape = shape, rate = rate))
ggplot(df, aes(x = x)) +
  geom_area(stat = "density", fill = myblue, alpha = 0.5) +
  geom_line(stat = "density", color = myblue, linewidth = 1) +
```

```
geom_line(data = df_gamma, aes(x = x, y = p), color = myred, linewidth
↪ = 1) +
labs(x = "Tiempo hasta el tercer contagio (horas)", y = "Densidad") +
theme_minimal()
```

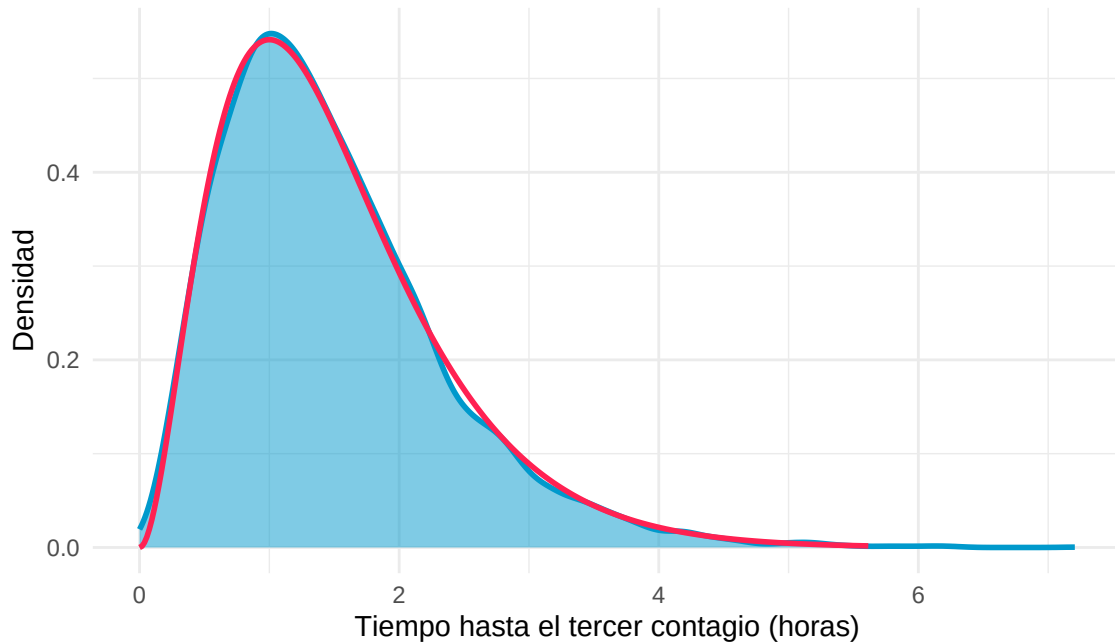


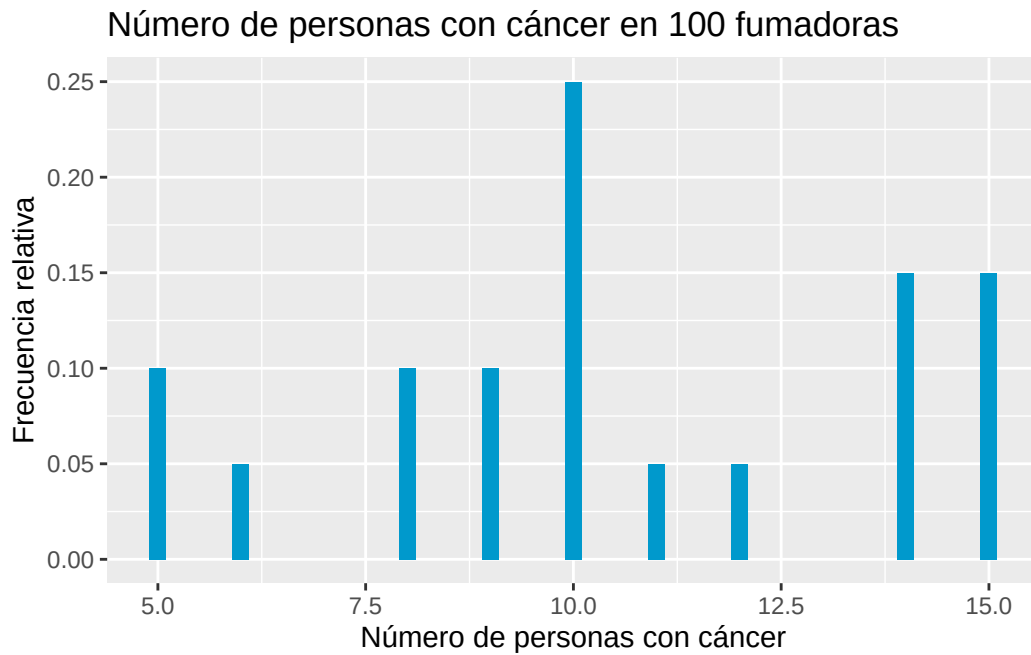
Figura 2.39: Muestra aleatoria de la distribución $\text{Gamma}(3, 2)$ frente al modelo teórico.

Ejercicio 2.5. En una población hay un 30 % de personas fumadoras. Se sabe que la prevalencia de cáncer de pulmón es del 10 % en fumadoras y del 1 % en no fumadoras.

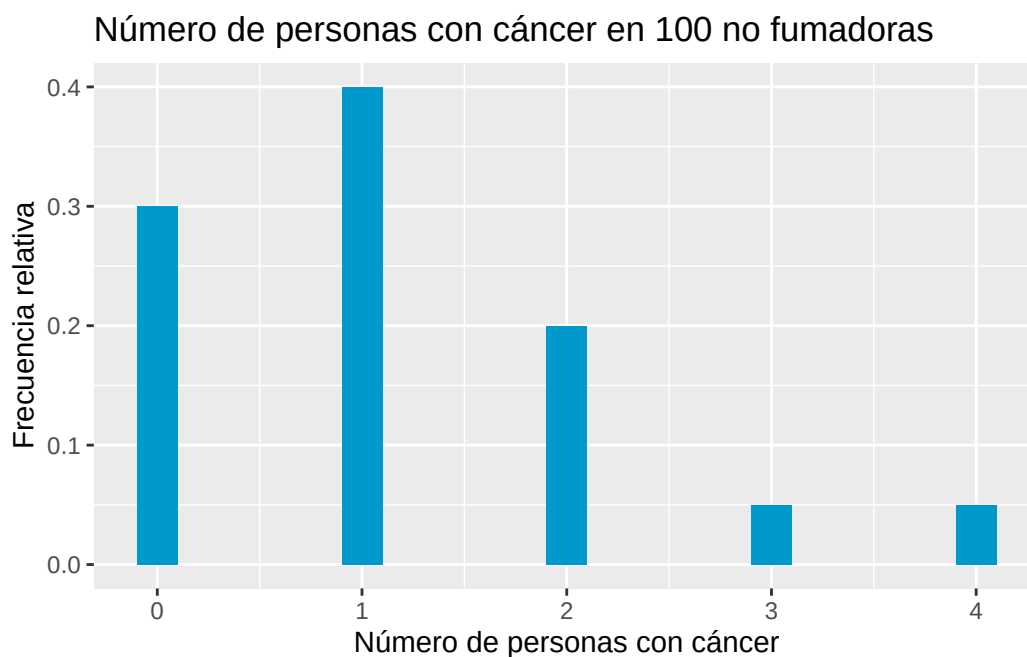
- Obtener una muestra aleatoria de tamaño 20 del número de personas con cáncer en muestras de 100 fumadoras y dibujar la distribución de frecuencias.
- Obtener una muestra aleatoria de tamaño 20 del número de personas con cáncer en muestras de 100 no fumadoras y dibujar la distribución de frecuencias.
- Obtener una muestra aleatoria de tamaño 20 del número de personas con cáncer en muestras de 100 personas de la población general y dibujar la distribución de frecuencias.
- Obtener una muestra aleatoria de tamaño 10000 del número de personas con cáncer en muestras de 100 personas de la población general y dibujar la distribución de frecuencias.

Solución

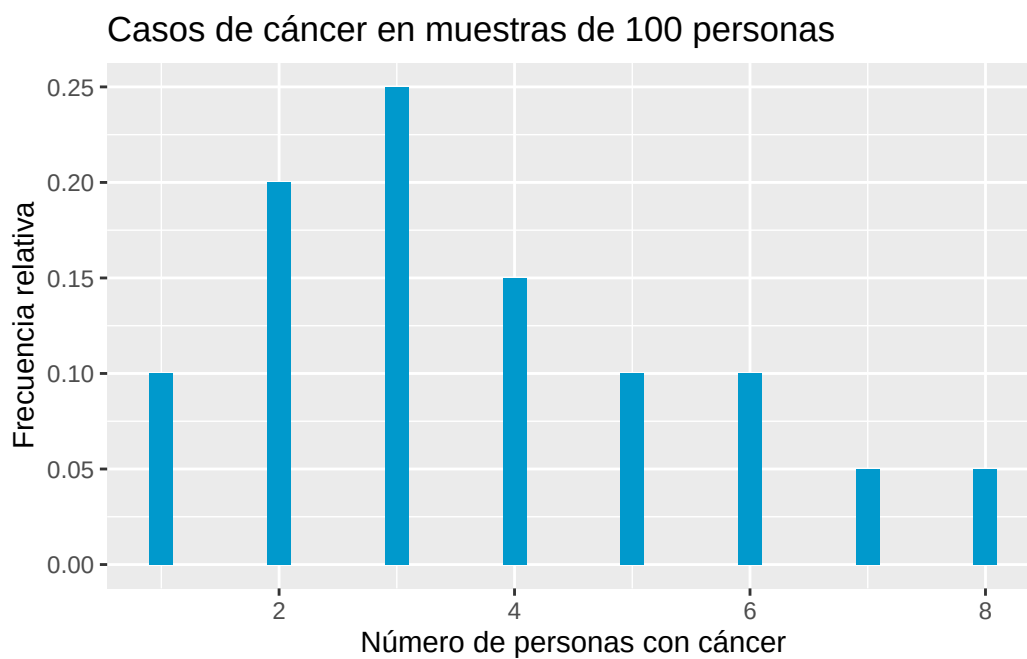
```
set.seed(123)
tibble(x = rbinom(n = 20, size = 100, prob = 0.1)) |>
  ggplot(aes(x = x, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  labs(title = "Número de personas con cáncer en 100 fumadoras",
       x = "Número de personas con cáncer", y = "Frecuencia relativa")
```



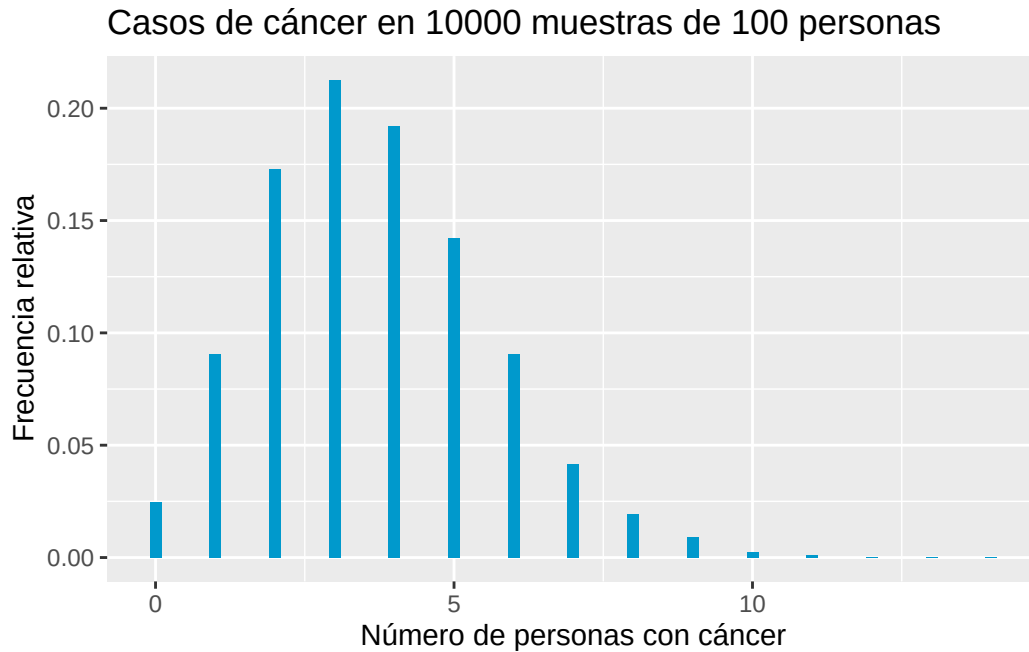
```
tibble(x = rbinom(n = 20, size = 100, prob = 0.01)) |>
  ggplot(aes(x = x, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  labs(title = "Número de personas con cáncer en 100 no fumadoras",
       x = "Número de personas con cáncer", y = "Frecuencia relativa")
```



```
# Cada muestra de 100 personas: primero el número de fumadoras y después  
  ↪ los casos en cada grupo  
set.seed(123)  
tibble(fumadoras = rbinom(n = 20, size = 100, prob = 0.3)) |>  
  mutate(x = rbinom(n = n(), size = fumadoras, prob = 0.1) +  
          rbinom(n = n(), size = 100 - fumadoras, prob = 0.01)) |>  
  ggplot(aes(x = x, y = after_stat(prop))) +  
  stat_count(fill = myblue, width = 0.2) +  
  labs(title = "Casos de cáncer en muestras de 100 personas",  
        x = "Número de personas con cáncer", y = "Frecuencia relativa")
```



```
set.seed(123)
tibble(fumadoras = rbinom(n = 10000, size = 100, prob = 0.3)) |>
  mutate(x = rbinom(n = n(), size = fumadoras, prob = 0.1) +
         rbinom(n = n(), size = 100 - fumadoras, prob = 0.01)) |>
  ggplot(aes(x = x, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  labs(title = "Casos de cáncer en 10000 muestras de 100 personas",
       x = "Número de personas con cáncer", y = "Frecuencia relativa")
```

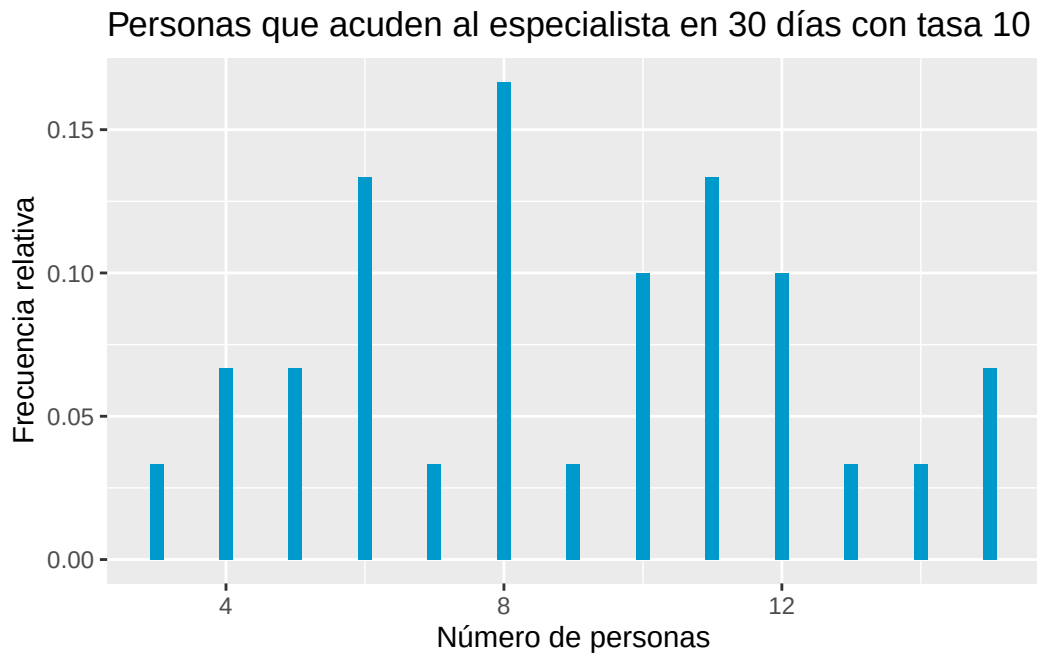


Ejercicio 2.6. La tasa diaria de personas que acuden a un especialista en un hospital sigue una distribución normal de media 10 y desviación típica 2.

- Obtener una muestra aleatoria de 30 días del número de personas que acuden al especialista, tomando una tasa fija de 10, y dibujar la distribución de frecuencias.
- Obtener una muestra aleatoria de 30 días del número de personas que acuden al especialista, tomando una tasa diaria aleatoria $\sim N(10, 2)$, y dibujar la distribución de frecuencias.
- Repetir el apartado a con 10000 días.
- Repetir el apartado b con 10000 días.

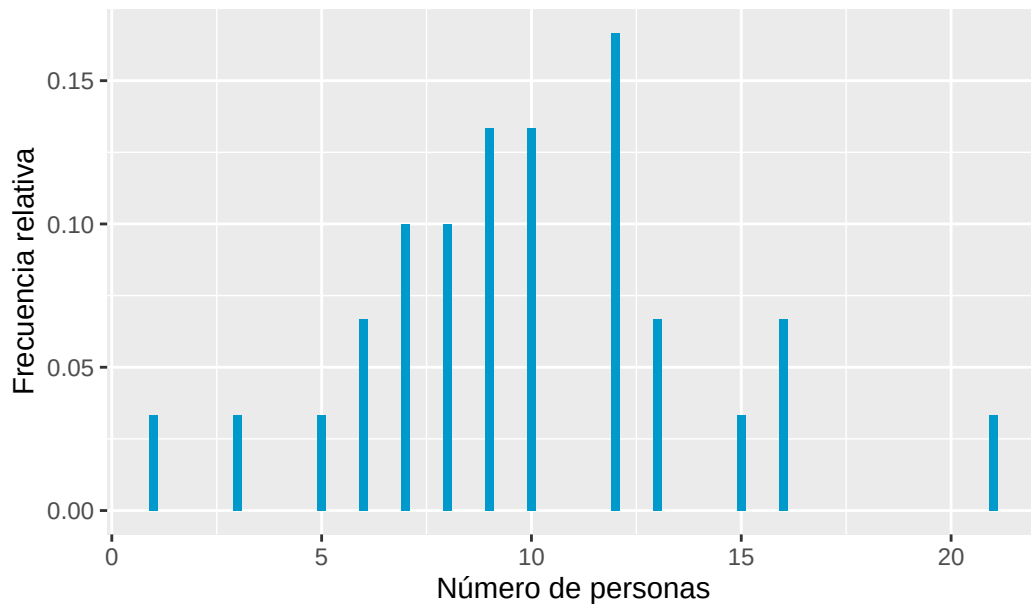
Solución

```
set.seed(123)
tibble(x = rpois(n = 30, lambda = 10)) |>
  ggplot(aes(x = x, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  labs(title = "Personas que acuden al especialista en 30 días con tasa
↪ 10",
       x = "Número de personas", y = "Frecuencia relativa")
```



```
set.seed(123)
tibble(lambda = rnorm(n = 30, mean = 10, sd = 2)) |>
  mutate(x = rpois(n = n(), lambda = lambda)) |>
  ggplot(aes(x = x, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  labs(title = "Personas que acuden al especialista en 30 días con tasa
↪ aleatoria",
       x = "Número de personas", y = "Frecuencia relativa")
```

Personas que acuden al especialista en 30 días con tasa alea



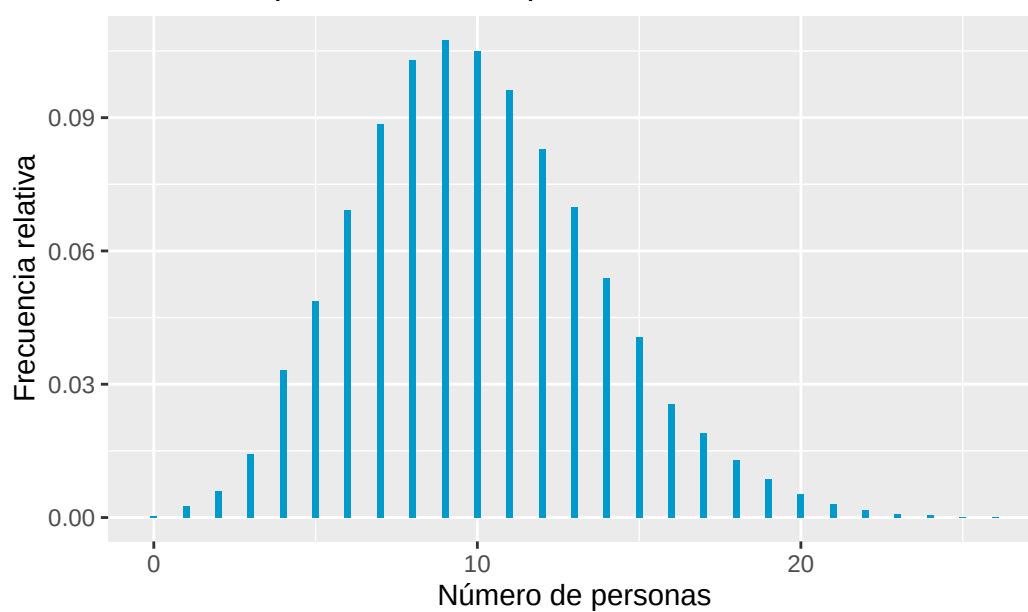
```
set.seed(123)
tibble(x = rpois(n = 10000, lambda = 10)) |>
  ggplot(aes(x = x, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  labs(title = "Personas que acuden al especialista en 10000 días con
    ↪ tasa 10",
    x = "Número de personas", y = "Frecuencia relativa")
```

Personas que acuden al especialista en 10000 días con tasa :



```
set.seed(123)
tibble(lambda = rnorm(n = 10000, mean = 10, sd = 2)) |>
  mutate(x = rpois(n = n(), lambda = lambda)) |>
  ggplot(aes(x = x, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  labs(title = "Personas que acuden al especialista en 10000 días con
    ↪ tasa aleatoria",
        x = "Número de personas", y = "Frecuencia relativa")
```

Personas que acuden al especialista en 10000 días con tasa λ



3 Fundamentos de la estadística bayesiana

Este capítulo presenta las ideas que sustentan toda la estadística bayesiana antes de entrar en las fórmulas y los modelos concretos. Se trata de responder a tres preguntas: por qué resulta útil un enfoque bayesiano, qué reglas de probabilidad lo sostienen y cómo se diferencia de la estadística frecuentista.

3.1. ¿Por qué pensar en bayesiano?

McElreath (*Statistical Rethinking*, Cap. 1) utiliza la metáfora del **golem de Praga**: una criatura de arcilla que sigue las instrucciones al pie de la letra, sin comprenderlas. Los modelos estadísticos son golems: potentes, pero ciegos. Siguen sus instrucciones exactamente, para bien o para mal.

Los **tests clásicos** (t de Student, chi-cuadrado, ANOVA, etc.) son golems preprogramados. Son útiles, pero rígidos: sirven para los problemas para los que fueron diseñados y no siempre para los nuestros. La estadística bayesiana permite **construir nuestros propios golems**, adaptados a cada problema.

i Mundos pequeños y mundos grandes

McElreath (*Statistical Rethinking*, Cap. 2) distingue entre:

- **Mundo pequeño** (*small world*): el mundo lógico y autocontenido del modelo. Dentro de él, la inferencia bayesiana es óptimamente eficiente: ningún otro método aprovecha mejor la información disponible.
- **Mundo grande** (*large world*): la realidad compleja donde desplegamos el modelo. No hay garantía de optimalidad, así que el buen rendimiento debe **demostrarse**, no deducirse.

En ciencias de la salud nuestros datos viven en un mundo grande: aparecen datos incompletos, censurados o con valores perdidos, estructuras jerárquicas (pacientes dentro de hospitales, hospitales dentro de regiones) y heterogeneidad entre poblaciones. El enfoque bayesiano permite **formalizar la incertidumbre** sobre ese mundo grande.

Según McElreath, para construir buenos golems necesitamos tres herramientas:

1. **Grafos causales (DAG)**: representar las relaciones causales antes de modelar.

2. **Modelos generativos:** simular datos bajo hipótesis concretas.
3. **Validación estadística:** contrastar el modelo con los datos reales.

 Idea clave

El análisis bayesiano no comienza con fórmulas, sino con una **historia generativa** sobre cómo se produjeron los datos (Gelman et al., *BDA3*, Cap. 1).

3.2. Probabilidad condicional

La inferencia bayesiana se construye sobre dos reglas básicas de la probabilidad.

Definición 3.1 (Regla del producto). La probabilidad conjunta de A y B es la probabilidad de B multiplicada por la probabilidad de A dado que B ha ocurrido:

$$P(A, B) = P(A | B) \cdot P(B)$$

Definición 3.2 (Regla de la suma). La probabilidad marginal de A se obtiene sumando (o integrando) sobre todos los posibles valores de B :

$$P(A) = \sum_B P(A, B) = \sum_B P(A | B) \cdot P(B)$$

 Dato clave

Toda la maquinaria bayesiana se deriva de estas dos reglas. No se necesita nada más (Gelman et al., *BDA3*, Cap. 1).

Ejemplo 3.1. Consideremos un hospital en el que se han ingresado 1000 pacientes en UCI, distribuidos según el tratamiento recibido y su evolución:

	Supervivencia (+)	Fallecimiento (−)	Total
Tratamiento A	180	120	300
Tratamiento B	490	210	700
Total	670	330	1000

La probabilidad de supervivencia condicionada a cada tratamiento es

- $P(\text{Supervivencia} | \text{Trat. A}) = 180/300 = 0,60$
- $P(\text{Supervivencia} | \text{Trat. B}) = 490/700 = 0,70$

 Advertencia

¿Es el tratamiento B realmente mejor? Puede haber confusión por la gravedad de los pacientes (paradoja de Simpson), ya que los pacientes más graves podrían haber recibido mayoritariamente el tratamiento A. Un modelo bayesiano jerárquico permitiría ajustar por estas variables.

3.2.1. El jardín de los senderos que se bifurcan

McElreath introduce la inferencia bayesiana con la metáfora del **jardín de senderos que se bifurcan**, inspirada en Borges. En esencia, la inferencia bayesiana consiste en **contar**: contar todas las formas en que los datos podrían haber ocurrido según cada hipótesis.

Ejemplo 3.2. Tenemos una bolsa con 4 canicas, cada una azul o blanca, pero no sabemos cuántas hay de cada color. Sacamos 3 canicas con reemplazo y obtenemos, en este orden, azul, blanca, azul.

Para cada composición posible de la bolsa contamos los caminos compatibles con esa observación. Un número de azules b produce $b \cdot (4 - b) \cdot b$ caminos, por ejemplo, 3 para $b = 1$, 8 para $b = 2$ y 9 para $b = 3$. Las composiciones con más caminos compatibles son más plausibles:

Composición	Caminos	Plausibilidad
4 blancas	0	0
1 azul, 3 blancas	3	0,067
2 azules, 2 blancas	8	0,178
3 azules, 1 blanca	9	0,200
4 azules	0	0

Estas plausibilidades, normalizadas para sumar 1, **son probabilidades**: constituyen la distribución a posteriori de la composición de la bolsa.

3.3. El teorema de Bayes

El teorema de Bayes proporciona la forma de actualizar nuestro conocimiento sobre un parámetro θ a partir de los datos y :

$$P(\theta | y) = \frac{P(y | \theta) \cdot P(\theta)}{P(y)}$$

Sus términos reciben nombres específicos:

$$\underbrace{P(\theta | y)}_{\text{Posterior}} = \frac{\overbrace{P(y | \theta)}^{\text{Verosimilitud}} \cdot \overbrace{P(\theta)}^{\text{Prior}}}{\underbrace{P(y)}_{\text{Evidencia}}}$$

Como la evidencia $P(y)$ no depende de θ , a menudo se usa la forma proporcional (Gelman et al., *BDA3*, Cap. 1):

$$P(\theta | y) \propto P(y | \theta) \cdot P(\theta)$$

i En palabras

Lo que sabemos **después** de ver los datos es proporcional a **cuán compatibles son los datos con cada valor del parámetro**, multiplicado por **lo que sabíamos antes** de ver los datos.

3.3.1. Los tres ingredientes

Distribución a priori $P(\theta)$. Refleja nuestro conocimiento (o ignorancia) sobre θ **antes** de observar los datos.

- Puede basarse en estudios previos, en opinión experta o ser deliberadamente vaga.
- Gelman recomienda priors **débilmente informativos**: descartan valores absurdos sin dominar los datos.
- McElreath enfatiza la **simulación predictiva a priori** para comprobar que la prior tiene sentido.

Por ejemplo, si modelamos la prevalencia de diabetes tipo 2 en una población, una prior $\text{Beta}(2, 15)$ centra la creencia en torno al 12%, en línea con datos epidemiológicos previos.

Función de verosimilitud $P(y | \theta)$. Es la probabilidad de observar los datos y para un valor concreto del parámetro θ .

- Es el **modelo generativo**: la historia de cómo se produjeron los datos.
- Es la misma función que se maximiza en la estimación frecuentista (máxima verosimilitud).
- En el contexto bayesiano se combina con la prior en lugar de maximizarse.

Si y pacientes de n dan positivo en un test, y suponemos independencia entre ellos,

$$P(y \mid \theta) = \binom{n}{y} \theta^y (1 - \theta)^{n-y}$$

Distribución a posteriori $P(\theta \mid y)$. Es nuestro conocimiento actualizado sobre θ tras observar los datos.

- Es una **distribución completa**, no un único número.
- Combina la información de la prior y la de los datos.
- A medida que acumulamos datos, la posterior se concentra en los valores más compatibles.
- Es un **compromiso** entre la prior y la verosimilitud (Gelman et al., *BDA3*).

Una consecuencia directa es que podemos afirmar que la probabilidad de que θ esté en un intervalo $[a, b]$ es, por ejemplo, 0,95. Esto es un **intervalo de credibilidad**, algo que **no podemos** afirmar con un intervalo de confianza frecuentista.

3.3.2. La posterior como compromiso

La distribución a posteriori es un **promedio ponderado** entre la información previa y los datos observados. Veámoslo con una prior Beta(2, 15), centrada en torno al 12 %, y 15 positivos de 120 pacientes testados:

```
# Prior: creencia inicial (prevalencia ~12%)
# Verosimilitud normalizada: 15 positivos de 120 testados
# Posterior (analítica por conjugación): Beta(2 + 15, 15 + 105)
rejilla <- seq(0, 0.35, length.out = 500)
df_compromiso <- tibble(
  theta = rep(rejilla, 3),
  densidad = c(dbeta(rejilla, 2, 15), dbeta(rejilla, 16, 106),
    ↪ dbeta(rejilla, 17, 120)),
  distribucion = rep(c("Prior", "Verosimilitud", "Posterior"), each =
    ↪ length(rejilla))
)

ggplot(df_compromiso, aes(theta, densidad, color = distribucion)) +
  geom_line(linewidth = 1) +
  scale_color_manual(values = c(Prior = greyceu, Verosimilitud =
    ↪ orangeceu, Posterior = blueceu)) +
  labs(x = expression(theta), y = "Densidad", color = NULL) +
  theme_minimal()
```

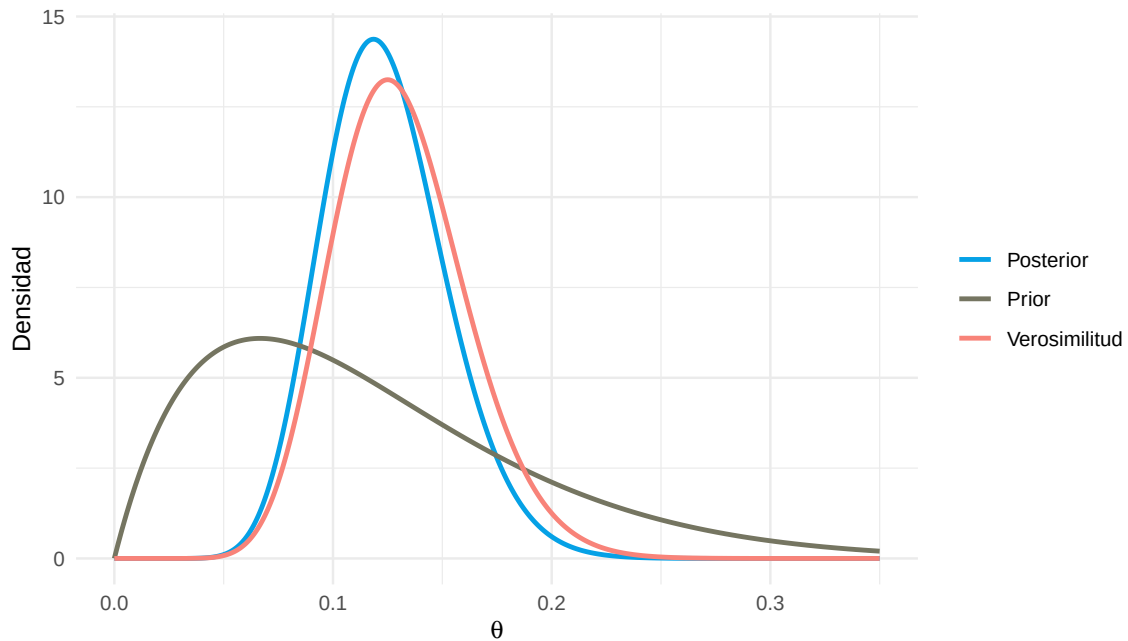


Figura 3.1: Distribución a priori, verosimilitud normalizada y distribución a posteriori de la prevalencia θ .

💡 Casos límite (Gelman et al., *BDA3*, Cap. 2)

- **Pocos datos:** la posterior se parece a la prior.
- **Muchos datos:** la posterior converge a la verosimilitud.
- **Prior fuerte y pocos datos:** la prior domina, así que hay que tener cuidado con las priors demasiado informativas.

3.4. Bayesiano frente a frecuentista

El enfoque frecuentista y el bayesiano son dos filosofías distintas para responder a las mismas preguntas.

Enfoque frecuentista.

- Los parámetros son **constantes fijas**, pero desconocidas.
- La probabilidad se define como **frecuencia a largo plazo**.
- La inferencia se basa en la **distribución muestral** del estimador.
- Un IC del 95 % significa que, si repitiéramos el estudio infinitas veces, el 95 % de los intervalos contruidos contendría el verdadero valor.

Enfoque bayesiano.

- Los parámetros tienen **distribuciones de probabilidad**.
- La probabilidad expresa **grado de incertidumbre**.
- La inferencia se basa en la **distribución a posteriori**.
- Un intervalo de credibilidad del 95 % significa que hay un 95 % de probabilidad de que el parámetro esté dentro de ese intervalo.

Perspectiva de Gelman

Gelman defiende una posición pragmática: los métodos bayesianos no son superiores por filosofía, sino por su **flexibilidad práctica** para modelar datos complejos como los que aparecen en salud (*BDA3*, Cap. 1).

Aspecto	Frecuentista	Bayesiano
Incorporar información previa	Difícil	Natural (priors)
Muestras pequeñas	Problemas	Robusto con buena prior
Datos jerárquicos	Modelos mixtos	Modelos multinivel naturales
Interpretación directa	Indirecta	Probabilidad del parámetro
Propagación de la incertidumbre	Aproximada	Completa
Predicción	Plug-in	Distribución predictiva
Coste computacional	Bajo	Mayor (MCMC)
Necesidad de priors	No	Sí (subjetividad)

¿Cuándo elegir un enfoque bayesiano? (Gelman y Hill)

Cuando los datos son escasos, las estructuras son complejas, necesitamos cuantificar la incertidumbre de forma directa o queremos incorporar conocimiento previo de manera formal.

3.5. Aplicación: pruebas diagnósticas

El cribado con pruebas diagnósticas es un escenario que todo profesional de salud debería dominar.

Ejemplo 3.3. Consideremos una mamografía de cribado con los siguientes datos:

- **Prevalencia** de cáncer de mama en la población cribada: $P(C) = 0,01$.
- **Sensibilidad** (tasa de verdaderos positivos): $P(+ | C) = 0,90$.
- **Especificidad** (tasa de verdaderos negativos): $P(- | \neg C) = 0,93$.

Una mujer obtiene un resultado **positivo**. ¿Cuál es la probabilidad de que realmente tenga cáncer?

Muchos profesionales responden “alrededor del 90 %”. La respuesta real es muy diferente.

3.5.1. Aplicando Bayes al cribado

$$P(C \mid +) = \frac{P(+ \mid C) \cdot P(C)}{P(+)}$$

El denominador se calcula con la regla de la suma:

$$P(+) = P(+ \mid C) \cdot P(C) + P(+ \mid \neg C) \cdot P(\neg C)$$

$$P(+) = 0,90 \times 0,01 + 0,07 \times 0,99 = 0,009 + 0,0693 = 0,0783$$

Por tanto,

$$P(C \mid +) = \frac{0,90 \times 0,01}{0,0783} \approx 0,115$$

! Resultado

Solo hay un **11,5 %** de probabilidad de cáncer tras un resultado positivo. Esto se debe a la baja prevalencia: la mayoría de los positivos son **falsos positivos**.

3.5.2. Frecuencias naturales

Otra forma de ver el mismo resultado es contar personas. Para 10.000 mujeres cribadas:

Grupo	Número	Resultado	Número de mujeres
Con cáncer (1 %)	100	Test positivo (sensibilidad 90 %)	90
Con cáncer (1 %)	100	Test negativo (falso negativo)	10

Grupo	Número	Resultado	Número de mujeres
Sin cáncer (99 %)	9.900	Test positivo (falso positivo, especificidad 93 %)	693
Sin cáncer (99 %)	9.900	Test negativo	9.207

De los $90 + 693 = 783$ tests positivos, solo **90** corresponden a mujeres con cáncer. Por tanto, el valor predictivo positivo es

$$VPP = \frac{90}{783} \approx 0,115$$

3.5.3. El papel crucial de la prevalencia

💡 Idea clave (Gelman et al., *BDA3*, Cap. 1)

La misma prueba diagnóstica tiene un valor predictivo muy diferente según la **prevalencia** (la prior) de la enfermedad en la población.

Prevalencia	VPP (test +)	VPN (test −)
0,1 %	1,3 %	99,99 %
1 %	11,5 %	99,9 %
5 %	40,4 %	99,4 %
20 %	76,3 %	97,4 %
50 %	92,8 %	90,3 %

La prevalencia actúa como una **distribución a priori**: en poblaciones de bajo riesgo, incluso las pruebas excelentes generan muchos falsos positivos. Esto tiene implicaciones directas para las políticas de cribado poblacional.

3.5.4. Actualización secuencial

Una propiedad fundamental del enfoque bayesiano es que, como dicen Gelman et al. (*BDA3*, Cap. 1), **la posterior de hoy es la prior de mañana**.

Si tras un primer test positivo hacemos un **segundo test independiente**, la probabilidad de cáncer se actualiza usando $P(C \mid +_1) = 0,115$ como nueva prior:

$$P(C \mid +_1, +_2) = \frac{P(+_2 \mid C) \cdot P(C \mid +_1)}{P(+_2 \mid +_1)} = \frac{0,90 \times 0,115}{0,90 \times 0,115 + 0,07 \times 0,885} \approx 0,626$$

i Resultado clínico

Tras **dos positivos consecutivos**, la probabilidad sube del 11,5 % al 62,6 %. Esta es la lógica que hay detrás de los protocolos de confirmación diagnóstica.

3.6. Priors: el ingrediente bayesiano

Elegir la distribución a priori es el paso más discutido del análisis bayesiano. Podemos distinguir tres tipos.

Priors no informativas. Pretenden dejar que los datos “hablen por sí mismos”.

- Prior uniforme: $P(\theta) \propto 1$.
- Prior de Jeffreys: invariante bajo reparametrización.

Sin embargo, las priors “no informativas” a menudo **sí son informativas** en el espacio de las predicciones. Una uniforme para un odds ratio implica que valores extremos ($OR = 1000$) son tan plausibles como valores moderados.

Priors débilmente informativas. Son la recomendación de Gelman: priors que **regularizan** sin dominar.

- Descartan valores absurdos.
- Estabilizan la estimación con muestras pequeñas.
- Permiten que los datos dominen cuando hay suficiente información.

Por ejemplo, para un coeficiente de regresión logística, $\beta \sim \text{Normal}(0, 2,5)$ descarta efectos gigantescos pero permite un rango amplio de valores.

Priors informativas. Se basan en conocimiento previo concreto: metaanálisis, estudios piloto u opinión experta formalizada (*prior elicitation*).

- Son apropiadas cuando existe evidencia previa sólida.
- Son fundamentales en ensayos clínicos con muestras pequeñas.
- Requieren un **análisis de sensibilidad**: ¿cuánto cambian los resultados si cambiamos la prior?

Por ejemplo, si cinco estudios previos estiman la eficacia de un fármaco en torno a 0,30, podemos usar $\theta \sim \text{Beta}(36, 84)$, cuya media es 0,30.

3.6.1. Simulación predictiva a priori

McElreath (*Statistical Rethinking*, Cap. 4) enfatiza un paso crucial que a menudo se omite:

Antes de ver los datos, **simula predicciones** desde tu prior. Si las predicciones a priori son absurdas, tu prior es mala.

```
# ¿Qué implica nuestra prior para la prevalencia?  
# Prior: Beta(2, 15) → centrada en torno al 12%  
prior_samples <- rbeta(1e4, 2, 15)  
  
# Simular datos bajo la prior (n = 100 pacientes)  
y_sim <- rbinom(1e4, size = 100, prob = prior_samples)  
  
# Visualizar: ¿son razonables los números de positivos simulados?  
tibble(y_sim) |>  
  ggplot(aes(y_sim)) +  
  geom_histogram(binwidth = 1, fill = "blue", color = "white") +  
  labs(x = "Número de positivos entre 100 pacientes", y = "Frecuencia")  
  ↪ +  
  theme_minimal()
```

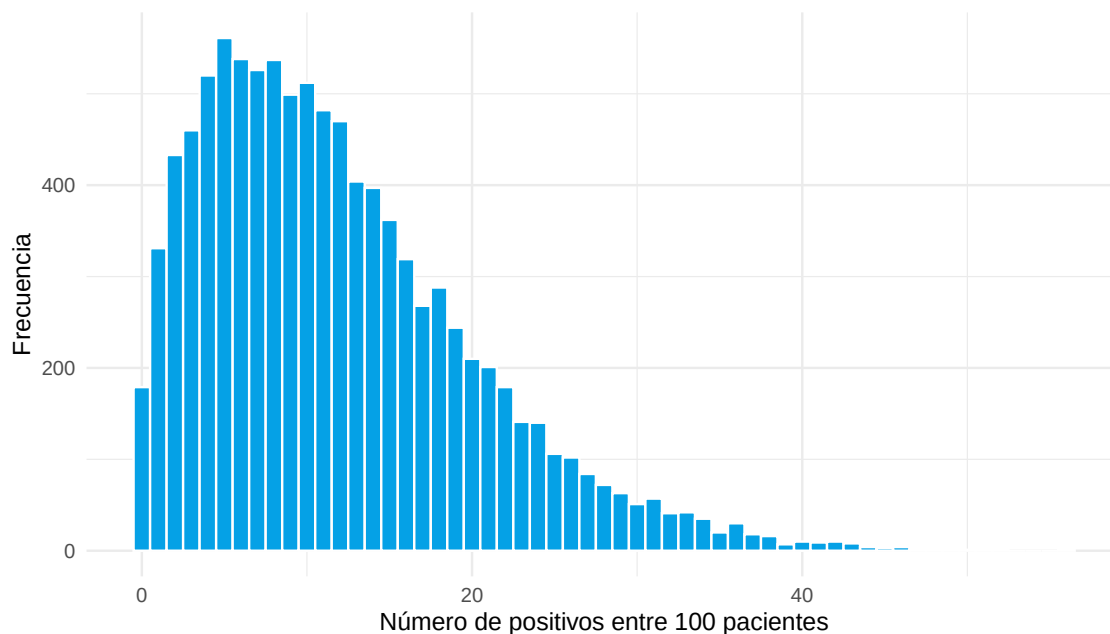


Figura 3.2: Distribución predictiva a priori del número de positivos entre 100 pacientes, con prior Beta(2, 15).

Regla práctica

Si la simulación predictiva a priori genera datos que serían imposibles o ridículos en nuestro contexto, hay que ajustar la prior.

3.7. Ideas clave

1. **La inferencia bayesiana es contar caminos** compatibles con los datos, ponderados por nuestro conocimiento previo (McElreath).
2. **El teorema de Bayes** combina prior, verosimilitud y evidencia para obtener la posterior: $P(\theta | y) \propto P(y | \theta) \cdot P(\theta)$.
3. **La posterior es un compromiso** entre lo que sabíamos (prior) y lo que observamos (datos). La balanza se inclina hacia los datos a medida que acumulamos evidencia (Gelman).
4. **La prevalencia es la prior** en diagnóstico: el mismo test tiene significados muy distintos según la población.
5. **La actualización es secuencial**: la posterior de un análisis puede ser la prior del siguiente.
6. **Las priors importan**: conviene elegir priors débilmente informativas y validarlas con simulación predictiva.

3.8. Lecturas recomendadas

Textos principales

- Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A., y Rubin, D.B. (2013). *Bayesian Data Analysis*, 3.^a ed. CRC Press. Capítulos 1-3.
- Gelman, A. y Hill, J. (2006). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press. Capítulos 1-2.
- McElreath, R. (2020). *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*, 2.^a ed. CRC Press. Capítulos 1-3.

Lecturas complementarias

- Gelman, A. (2008). Objections to Bayesian statistics. *Bayesian Analysis*, 3(3), 445-449.
- Greenland, S. (2006). Bayesian perspectives for epidemiological research. *International Journal of Epidemiology*, 35(3), 765-775.
- van de Schoot, R. et al. (2021). Bayesian statistics and modelling. *Nature Reviews Methods Primers*, 1, 1.

4 Regla de Bayes

La regla de Bayes es el núcleo de la estadística bayesiana. Permite actualizar la probabilidad de una hipótesis a partir de la evidencia observada. En este capítulo se presenta el teorema de Bayes sobre sistemas completos de sucesos, se introducen los conceptos de distribución a priori, función de verosimilitud y distribución a posteriori, y se simula el proceso de actualización con R. Al final se estudian las distribuciones conjugadas, que simplifican mucho el cálculo de la posterior.

4.1. Sistema completo de sucesos

Definición 4.1 (Sistema completo de sucesos). Una colección de sucesos $A_1, A_2, \dots, A_n$ de un mismo espacio muestral Ω es un *sistema completo* si cumple las siguientes condiciones:

1. La unión de todos ellos es el espacio muestral: $A_1 \cup \dots \cup A_n = \Omega$.
2. Son incompatibles dos a dos: $A_i \cap A_j = \emptyset$ para todo $i \neq j$.

En realidad, un sistema completo de sucesos es una partición del espacio muestral de acuerdo con algún atributo, como el sexo o el grupo sanguíneo.

4.2. Teorema de la probabilidad total

Conocer las probabilidades de un determinado suceso en cada una de las partes de un sistema completo puede ser útil para calcular su probabilidad global.

Teorema 4.1 (Probabilidad total). *Dado un sistema completo de sucesos $A_1, \dots, A_n$ de un espacio muestral Ω y un suceso B de ese mismo espacio muestral, la probabilidad de B se puede calcular mediante la fórmula*

$$P(B) = \sum_{i=1}^n P(A_i \cap B) = \sum_{i=1}^n P(A_i)P(B|A_i).$$

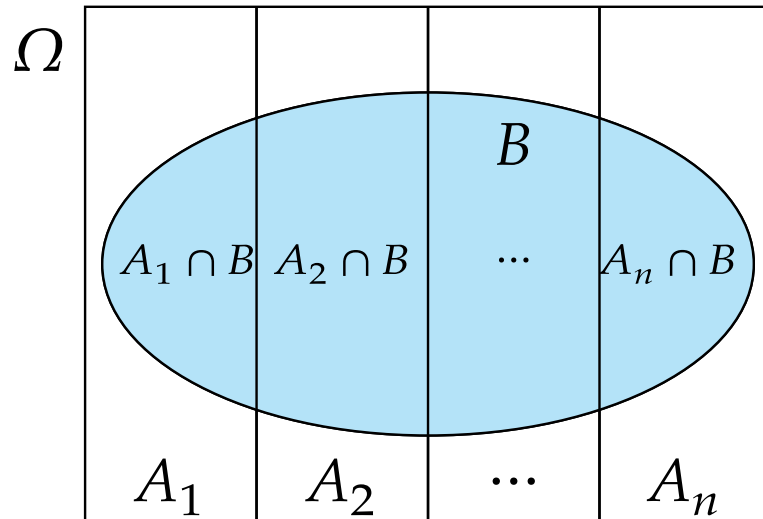


Figura 4.1: Teorema de la probabilidad total.

Ejemplo 4.1. Un test diagnóstico para una enfermedad E tiene una sensibilidad del 90 % y una especificidad del 95 %, es decir,

$$P(+|E) = 0,9 \quad \text{y} \quad P(-|\overline{E}) = 0,95.$$

Si la prevalencia de la enfermedad en la población es del 10 %, ¿qué porcentaje de positivos se espera en una muestra aleatoria de la población?

La siguiente tabla de probabilidades conjuntas resume la situación:

	Enfermo	No enfermo	Total
+	0,090	0,045	0,135
−	0,010	0,855	0,865
Total	0,100	0,900	1

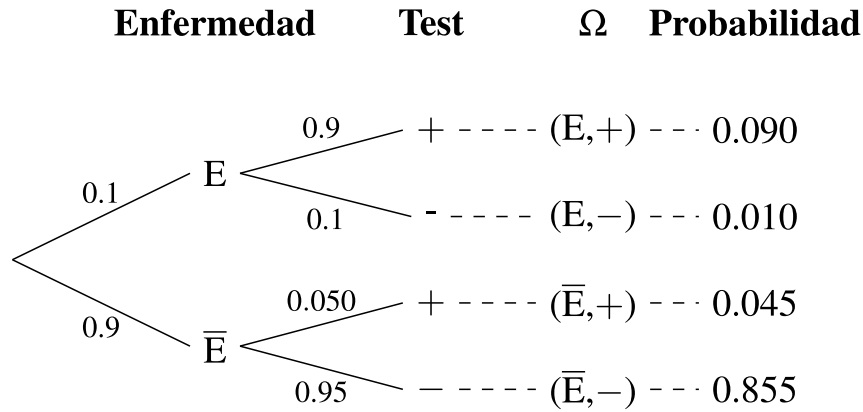


Figura 4.2: Diagrama de árbol del test diagnóstico.

Para responder a la pregunta se aplica el teorema de la probabilidad total con el sistema completo $\{E, \bar{E}\}$:

$$P(+) = P(E)P(+|E) + P(\bar{E})P(+|\bar{E}) = 0,1 \cdot 0,9 + 0,9 \cdot 0,05 = 0,135.$$

Es decir, alrededor del 13,5 % de los resultados del test serán positivos.

En el fondo se trata de una media ponderada de probabilidades.

4.3. Teorema de Bayes

Los sucesos de un sistema completo $A_1, \dots, A_n$ también pueden verse como las distintas hipótesis ante un determinado hecho B . En estas condiciones resulta útil poder calcular las probabilidades a posteriori $P(A_i|B)$ de cada una de las hipótesis.

Teorema 4.2 (Teorema de Bayes). *Dado un sistema completo de sucesos $A_1, \dots, A_n$ y un suceso B de un mismo espacio muestral Ω , la probabilidad de cada suceso A_i condicionada por B puede calcularse con la fórmula*

$$P(A_i|B) = \frac{P(A_i \cap B)}{P(B)} = \frac{P(A_i)P(B|A_i)}{\sum_{j=1}^n P(A_j)P(B|A_j)}.$$

Ejemplo 4.2. En el ejemplo anterior, una pregunta más interesante es qué diagnosticar a una persona que ha dado positivo en el test. Podemos interpretar E y $\bar{E}$ como las dos posibles hipótesis ante el resultado positivo del test.

Probabilidades a priori (sin información sobre el resultado del test)

$$P(E) = 0,1 \quad \text{y} \quad P(\overline{E}) = 0,9.$$

Sin información, el diagnóstico sería que la persona no tiene la enfermedad.

Sin embargo, si al realizar el test a una persona se obtiene un resultado positivo, esa información condiciona las hipótesis. Para decidir entre ellas hay que calcular sus probabilidades a posteriori, $P(E|+)$ y $P(\overline{E}|+)$:

$$P(E|+) = \frac{P(E)P(+|E)}{P(E)P(+|E) + P(\overline{E})P(+|\overline{E})} = \frac{0,1 \cdot 0,9}{0,1 \cdot 0,9 + 0,9 \cdot 0,05} = \frac{0,09}{0,135} = 0,667,$$

$$P(\overline{E}|+) = \frac{P(\overline{E})P(+|\overline{E})}{P(E)P(+|E) + P(\overline{E})P(+|\overline{E})} = \frac{0,9 \cdot 0,05}{0,1 \cdot 0,9 + 0,9 \cdot 0,05} = \frac{0,045}{0,135} = 0,333.$$

Con un resultado positivo, la probabilidad de tener la enfermedad ha aumentado de 0,10 a 0,667, y el diagnóstico pasa a ser que la persona tiene la enfermedad.

Ejercicio 4.1. Se dispone de dos test, A y B , para diagnosticar una enfermedad con una prevalencia del 1 % en la población. El test A tiene una sensibilidad del 98 % y una especificidad del 80 %, mientras que el test B tiene una sensibilidad del 75 % y una especificidad del 99 %.

- ¿Cuál de los dos test funcionaría mejor como test de cribado?
- Si aplicamos el test de cribado a una persona y da positivo, ¿cuál es la probabilidad de que esté enferma? ¿Qué diagnosticaríamos?
- Si aplicamos a una persona el test de cribado y, después, el otro test, y ambos dan positivo, ¿cuál es la probabilidad de que esté enferma? ¿Qué diagnosticaríamos?

Solución

- Para cribado se prioriza una alta sensibilidad, es decir, detectar a todos los enfermos minimizando los falsos negativos. Por tanto, el test A es mejor para cribado. Su baja especificidad (80 %) produce falsos positivos, pero estos se filtran con una prueba confirmatoria posterior.
- Aplicando el teorema de Bayes al resultado positivo del test A :

$$P(E|+_A) = \frac{P(+_A|E) \cdot P(E)}{P(+_A|E) \cdot P(E) + P(+_A|\overline{E}) \cdot P(\overline{E})}$$

$$= \frac{0,98 \times 0,01}{0,98 \times 0,01 + 0,20 \times 0,99} = \frac{0,0098}{0,2078} \approx 0,047$$

Como $P(E|+_A) = 0,047 < 0,5$, diagnosticaríamos sano, aunque derivaríamos a la persona a una prueba confirmatoria.

- c. Ahora usamos $P(E|+_A) = 0,047$ en lugar de $P(E) = 0,01$ como nueva probabilidad a priori, y aplicamos Bayes con el test B :

$$\begin{aligned} P(E | +_A, +_B) &= \frac{P(+_B | E) \cdot P(E | +_A)}{P(+_B | E) \cdot P(E | +_A) + P(+_B | \bar{E}) \cdot P(\bar{E} | +_A)} \\ &= \frac{0,75 \times 0,047}{0,75 \times 0,047 + 0,01 \times 0,953} \approx 0,788. \end{aligned}$$

Como $P(E|+_A, +_B) \approx 0,79 > 0,5$, diagnosticaríamos enfermedad.

4.4. Regla de Bayes

En terminología bayesiana, si H representa una hipótesis y D los datos observados, el teorema de Bayes se puede expresar como una regla de actualización de la probabilidad de la hipótesis H a la luz de la evidencia D :

$$\underbrace{P(H|D)}_{\text{Probabilidad a posteriori}} = \frac{\overbrace{P(H)}^{\text{Probabilidad a priori}} \times \overbrace{P(D|H)}^{\text{Verosimilitud}}}{\underbrace{P(D)}_{\text{Evidencia marginal}}}$$

El denominador $P(D)$ actúa como normalizador: garantiza que las probabilidades a posteriori de todas las hipótesis sumen 1.

4.5. Función de verosimilitud

En la regla de Bayes, la probabilidad $P(D|H)$ se conoce como *verosimilitud* de la hipótesis H dada la evidencia D .

Cuando tenemos varias hipótesis, la verosimilitud de cada una de ellas se puede comparar con la de las demás para determinar cuál es la más compatible con los datos observados.

Definición 4.2 (Función de verosimilitud). Dado un sistema completo de sucesos con hipótesis $H_1, H_2, \dots, H_n$ y un suceso D que representa la evidencia observada, la *función de verosimilitud* asigna a cada hipótesis H_i la probabilidad de observar la evidencia D si esa hipótesis fuera cierta:

$$L(H_i|D) = P(D|H_i).$$

! Importante

Los valores de la función de verosimilitud son probabilidades condicionadas, pero la variable de la función de verosimilitud es el condicionante, es decir, la hipótesis H_i , y no la evidencia D :

$$L(x|D) = P(D|x).$$

Ejemplo 4.3. Consideremos de nuevo el test diagnóstico con sensibilidad del 90 % y especificidad del 95 %. Dada la evidencia de que el resultado del test ha sido positivo, la función de verosimilitud de las dos hipótesis E y $\bar{E}$ es

$$L(E|+) = P(+|E) = 0,9 \quad \text{y} \quad L(\bar{E}|+) = P(+|\bar{E}) = 1 - 0,95 = 0,05.$$

En este caso, la hipótesis E es mucho más compatible con la evidencia $+$ que la hipótesis $\bar{E}$.

4.5.1. Interpretación de la función de verosimilitud

La función de verosimilitud se puede interpretar como una medida de la compatibilidad de cada hipótesis con los datos observados: *cuanto mayor sea la verosimilitud de una hipótesis dada la evidencia, más plausible será la hipótesis.*

Es importante destacar que la función de verosimilitud **no es una función de probabilidad**: la suma de las verosimilitudes de todas las hipótesis no tiene por qué ser igual a 1.

Ejemplo 4.4. La siguiente tabla recoge la probabilidad a priori y la función de verosimilitud de las hipótesis E y $\bar{E}$ dada la evidencia $+$ en el ejemplo del test diagnóstico.

Hipótesis	Probabilidad a priori $P(H)$	Función de verosimilitud $L(H +)$
E	0,1	0,9
$\bar{E}$	0,9	0,05

4.6. Probabilidad a posteriori

Si disponemos de la probabilidad a priori de cada hipótesis y de su función de verosimilitud dada la evidencia, podemos calcular la probabilidad a posteriori de cada hipótesis con la regla de Bayes:

$$P(H_i|D) = \frac{P(H_i) \times P(D|H_i)}{P(D)}.$$

La probabilidad del denominador $P(D)$ es un factor común para todas las hipótesis, por lo que la probabilidad a posteriori de cada hipótesis es proporcional al producto de su probabilidad a priori por su verosimilitud:

$$P(H_i|D) \propto P(H_i) \times P(D|H_i).$$

Ejemplo 4.5. En el ejemplo del test diagnóstico, dado que el test ha resultado positivo, obtenemos las siguientes plausibilidades a posteriori:

Hipótesis	Probabilidad a priori $P(H)$	Verosimilitud $L(H +)$	Plausibilidad a posteriori $P(H) \times L(H +)$
E	0,1	0,9	0,090
$\bar{E}$	0,9	0,05	0,045

La suma de las plausibilidades a posteriori no es igual a 1, por lo que no se pueden interpretar como probabilidades. Sin embargo, sí podemos compararlas: la hipótesis E es el doble de plausible que la hipótesis $\bar{E}$ dada la evidencia de que el test ha resultado positivo.

¿Cómo podemos transformar las plausibilidades a posteriori en probabilidades a posteriori? Normalizándolas: dividiendo cada una de ellas por la suma de todas las plausibilidades.

Hipótesis	Plausibilidad a posteriori	Probabilidad a posteriori $P(H +)$
E	0,090	$0,090/(0,090+0,045) = 0,667$
$\bar{E}$	0,045	$0,045/(0,090+0,045) = 0,333$

4.7. Simulación de la regla de Bayes

Podemos simular el procedimiento de actualización de la probabilidad de una hipótesis a la luz de la evidencia que realiza la regla de Bayes. Para ello necesitamos tomar muestras aleatorias de:

- La distribución a priori de la hipótesis H .
- La distribución conjunta de la hipótesis H y la evidencia D , teniendo en cuenta la función de verosimilitud $L(H|D)$.

Al filtrar después la muestra conjunta por la evidencia observada, obtenemos una muestra de la distribución a posteriori.

4.7.1. Muestreo de la distribución a priori

Ejercicio 4.2. Obtener una muestra aleatoria de tamaño $n = 10000$ de la distribución a priori de la hipótesis H , con probabilidades $P(E) = 0,1$ y $P(\bar{E}) = 0,9$, y representar gráficamente la distribución de frecuencias muestral.

Solución

```
df_H = tibble(hipótesis = c("E", "NoE")) # Hipótesis
prior <- c(0.1, 0.9) # P(E) y P(No E)
set.seed(123) # Para reproducibilidad
n <- 10000 # Tamaño de la muestra
# Muestreo de la distribución a priori
muestra_prior <- slice_sample(df_H, n = n, weight_by = prior, replace =
  ↪ TRUE)
# Diagrama de barras de la distribución de frecuencias
ggplot(muestra_prior, aes(x = hipótesis)) +
  stat_count(fill = myblue) +
  labs(x = "Hipótesis", y = "Frecuencia absoluta") +
  theme_minimal()
```

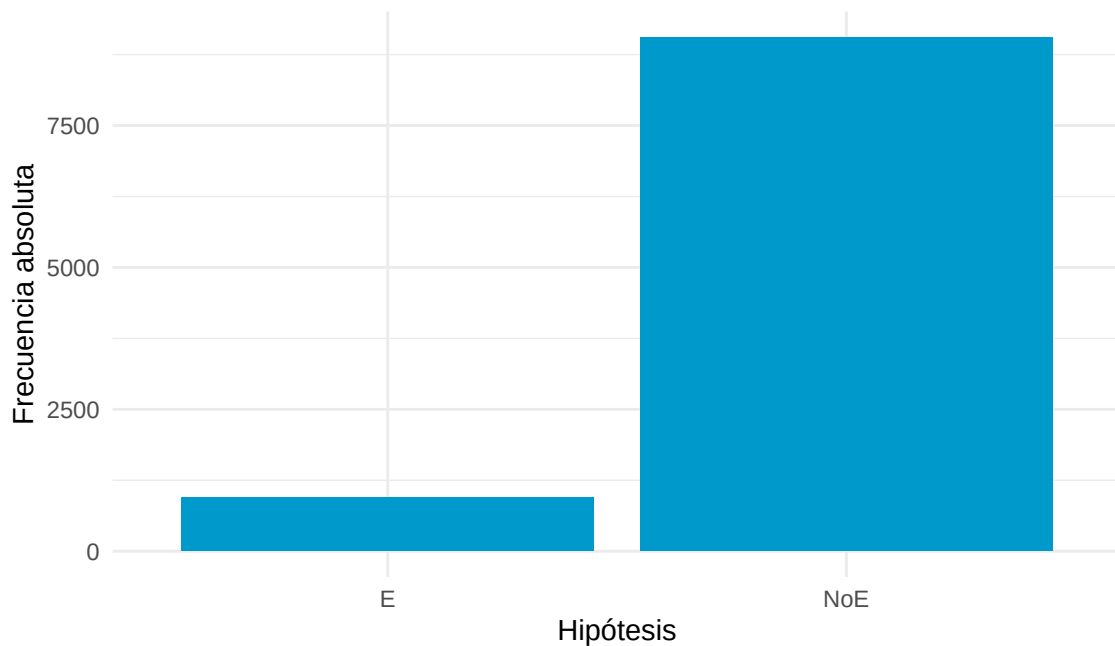


Figura 4.3: Distribución de frecuencias de una muestra de la distribución a priori.

4.7.2. Muestreo de la distribución conjunta

Para simular la distribución conjunta de la hipótesis y el resultado del test, generamos para cada individuo de la muestra un resultado del test según la función de verosimilitud: $P(+|E) = 0,9$ y $P(+|\bar{E}) = 0,05$.

```
# Muestreo de la función de verosimilitud (resultado del test)
muestra_prior <- muestra_prior |>
  mutate(test = ifelse(runif(n()) < ifelse(hipótesis == "E", 0.9, 0.05),
    ↪ "+", "-"))
# Diagrama de barras de la distribución de frecuencias del test
ggplot(muestra_prior, aes(x = hipótesis, fill = test)) +
  stat_count(position = "stack") +
  labs(x = "Hipótesis", y = "Frecuencia absoluta") +
  theme_minimal()
```

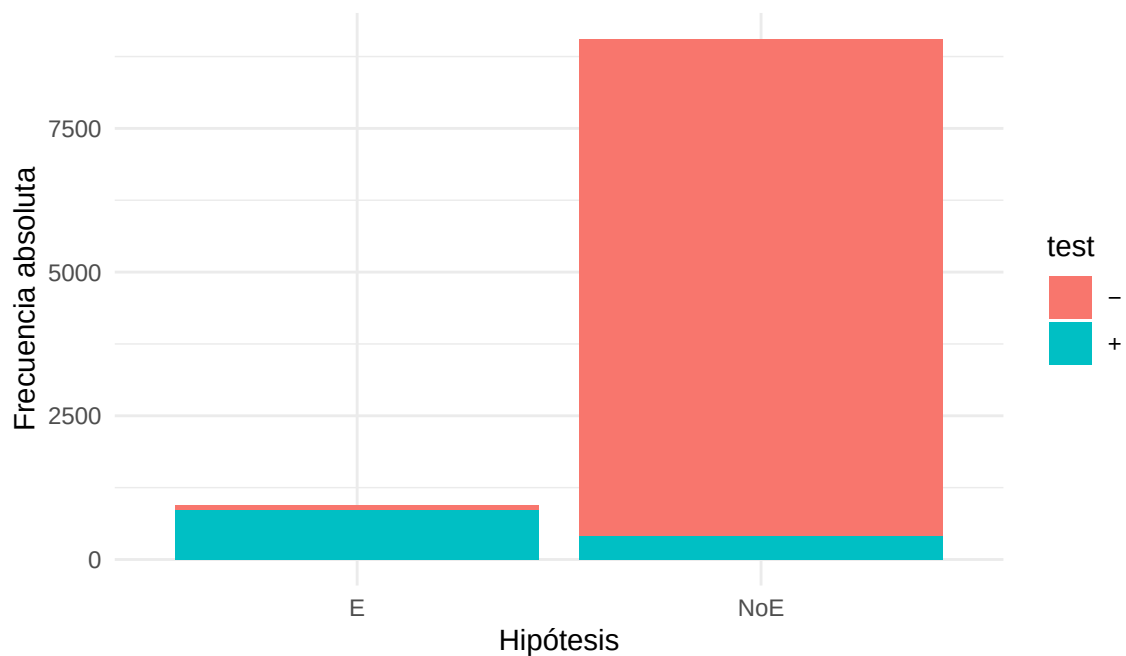


Figura 4.4: Distribución de frecuencias conjunta de la hipótesis y el resultado del test.

4.7.3. Muestreo de la distribución a posteriori

Una vez tenemos la distribución conjunta, podemos obtener una muestra de la distribución a posteriori filtrando los casos en los que el resultado del test ha sido positivo.

```
# Filtrar casos con test positivo
muestra_posterior <- muestra_prior |> filter(test == "+")
# Diagrama de barras de la distribución de frecuencias de la muestra a
  ↪ posteriori
ggplot(muestra_posterior, aes(x = hipótesis, y = after_stat(prop), group
  ↪ = 1)) +
  stat_count(fill = myblue) +
  labs(x = "Hipótesis", y = "Frecuencia relativa") +
  theme_minimal()
```

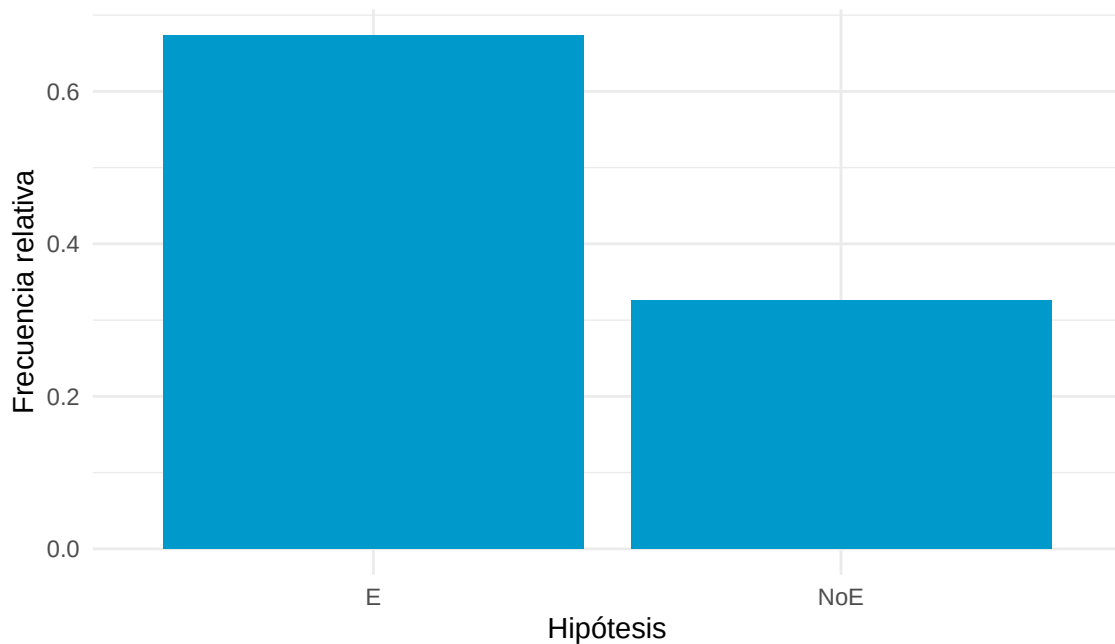


Figura 4.5: Distribución de frecuencias de una muestra de la distribución a posteriori.

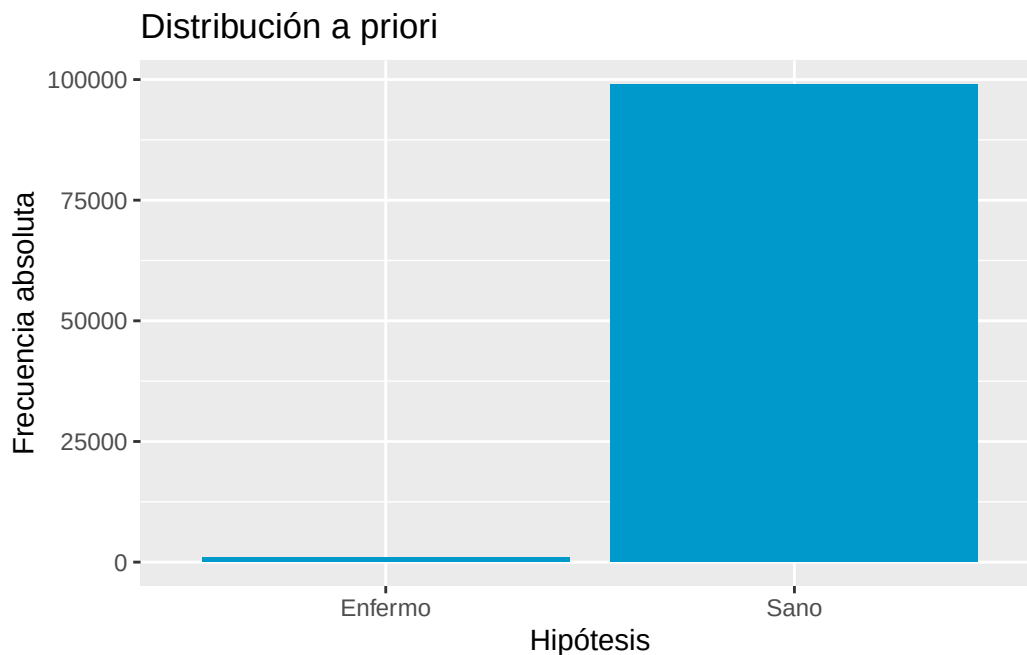
Ejercicio 4.3. Repetimos la simulación con 100000 individuos. La prevalencia de la enfermedad es del 1 % (prior $P(E) = 0,01$), el test A tiene sensibilidad 0,98 y especificidad 0,80, y el test B tiene sensibilidad 0,75 y especificidad 0,99.

- Obtener una muestra aleatoria de tamaño 100000 de la distribución a priori de la hipótesis y representarla gráficamente.
- Obtener una muestra aleatoria de tamaño 100000 de la distribución conjunta de las hipótesis y el resultado del test A y representarla gráficamente.
- Obtener una muestra aleatoria de la distribución a posteriori de las hipótesis dada la evidencia de que el test A ha resultado positivo y representarla gráficamente.
- Obtener una muestra aleatoria de la distribución conjunta de las hipótesis y el resultado del test B y representarla gráficamente.

- e. Obtener una muestra aleatoria de la distribución a posteriori de las hipótesis dada la evidencia de que el test B ha resultado positivo y representarla gráficamente.

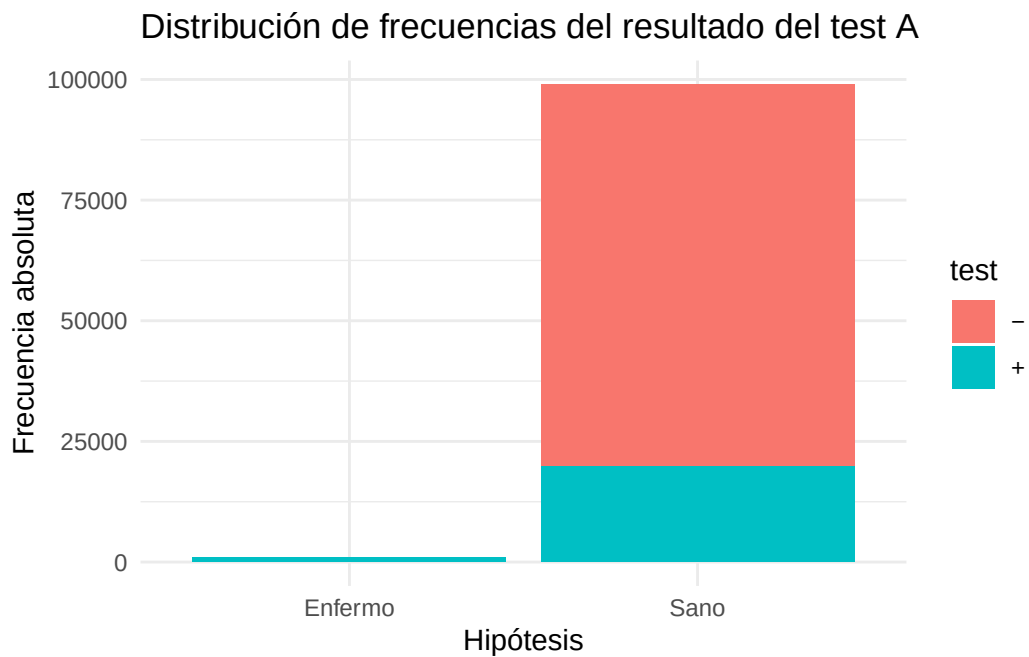
Solución

```
df_H = tibble(hipótesis = c("Enfermo", "Sano")) # Hipótesis
prior <- c(0.01, 0.99) # P(Enfermo) y P(Sano)
set.seed(123) # Para reproducibilidad
n <- 100000 # Tamaño de la muestra
# Muestreo de la distribución a priori
muestra_prior <- slice_sample(df_H, n = n, weight_by = prior, replace =
  ↪ TRUE)
muestra_prior |> ggplot(aes(x = hipótesis)) +
  stat_count(fill = myblue) +
  labs(title = "Distribución a priori", x = "Hipótesis", y = "Frecuencia
  ↪ absoluta")
```



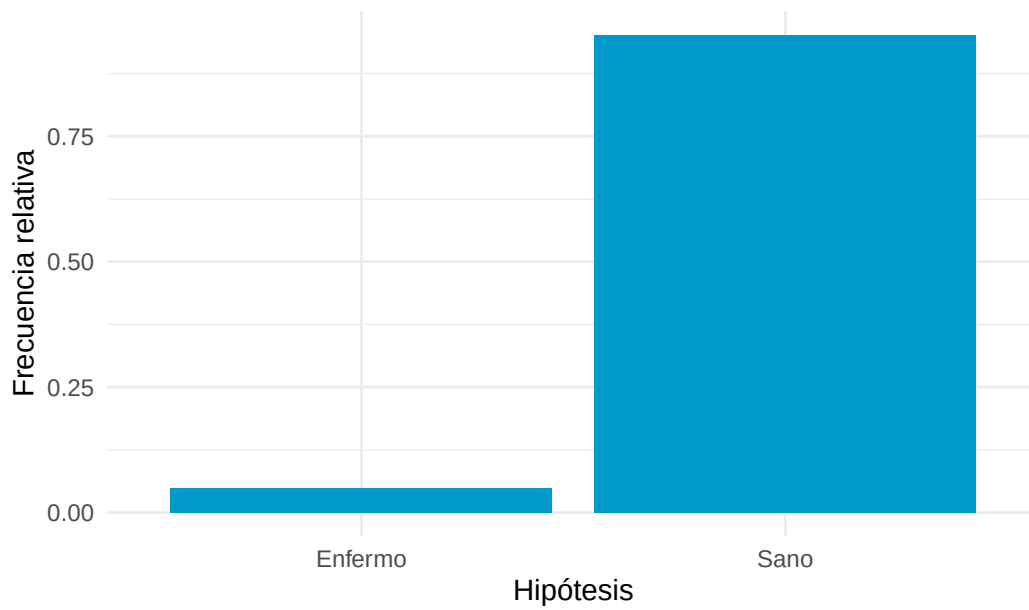
```
# Muestreo de la distribución conjunta con el test A
muestra_prior <- muestra_prior |>
  mutate(test = ifelse(runif(n()) < ifelse(hipótesis == "Enfermo", 0.98,
  ↪ 0.2), "+", "-"))
ggplot(muestra_prior, aes(x = hipótesis, fill = test)) +
  stat_count(position = "stack") +
  labs(title = "Distribución de frecuencias del resultado del test A",
```

```
x = "Hipótesis", y = "Frecuencia absoluta") +  
theme_minimal()
```

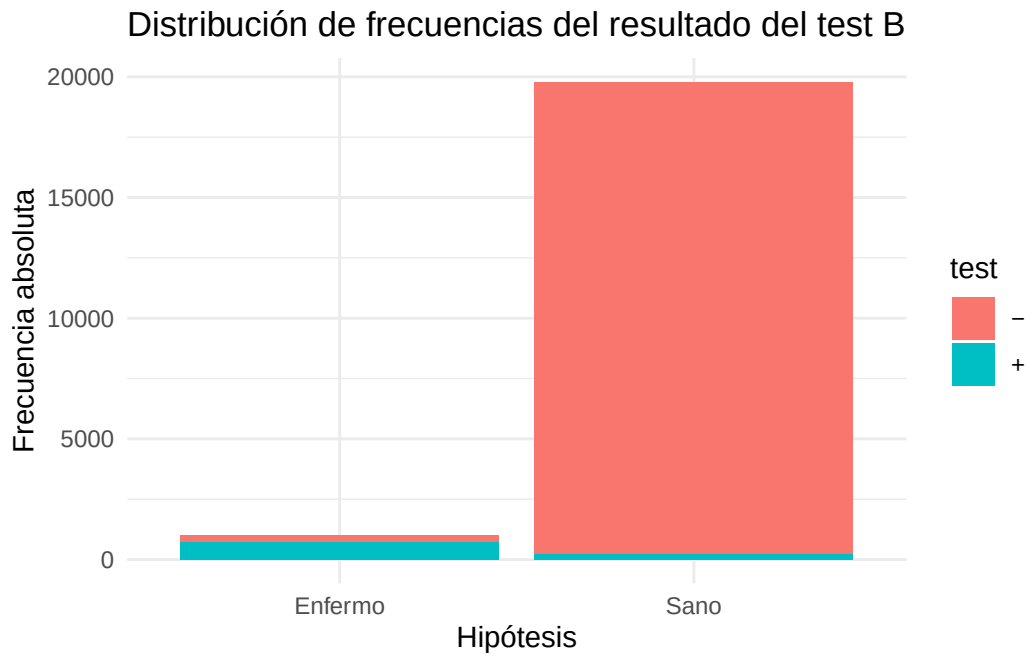


```
# Filtrar casos con test A positivo  
muestra_posterior <- muestra_prior |> filter(test == "+")  
ggplot(muestra_posterior, aes(x = hipótesis, y = after_stat(prop), group  
↪ = 1)) +  
  stat_count(fill = myblue) +  
  labs(title = "Distribución a posteriori tras un test A positivo",  
        x = "Hipótesis", y = "Frecuencia relativa") +  
  theme_minimal()
```

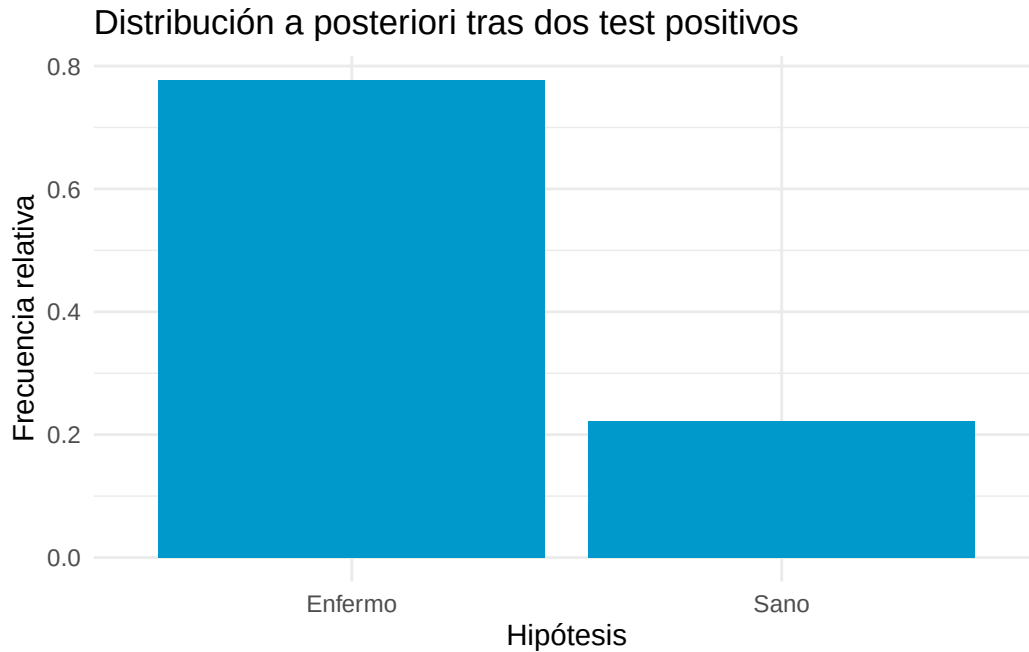
Distribución a posteriori tras un test A positivo



```
# Muestreo de la distribución conjunta con el test B sobre los positivos  
↪ de A  
muestra_prior2 <- muestra_posterior |>  
  mutate(test = ifelse(runif(n()) < ifelse(hipótesis == "Enfermo", 0.75,  
↪ 0.01), "+", "-"))  
ggplot(muestra_prior2, aes(x = hipótesis, fill = test)) +  
  stat_count(position = "stack") +  
  labs(title = "Distribución de frecuencias del resultado del test B",  
        x = "Hipótesis", y = "Frecuencia absoluta") +  
  theme_minimal()
```



```
# Filtrar casos con test B positivo
muestra_posterior <- muestra_prior2 |> filter(test == "+")
ggplot(muestra_posterior, aes(x = hipótesis, y = after_stat(prop), group
↵ = 1)) +
  stat_count(fill = myblue) +
  labs(title = "Distribución a posteriori tras dos test positivos",
        x = "Hipótesis", y = "Frecuencia relativa") +
  theme_minimal()
```



4.8. Distribuciones conjugadas

Hemos visto que, para obtener la distribución de probabilidad a posteriori de una hipótesis dada la evidencia, hay que combinar la distribución a priori de la hipótesis con la función de verosimilitud dada la evidencia, mediante la regla de Bayes.

Existen familias de distribuciones de probabilidad que, al combinarse con determinadas verosimilitudes, mantienen su forma funcional. Estas familias se denominan *distribuciones conjugadas* y son de gran utilidad en la inferencia bayesiana, porque la posterior se obtiene en forma cerrada sin necesidad de calcular la integral del denominador. Las parejas más comunes son:

- Beta-Binomial.
- Beta-Geométrica.
- Gamma-Poisson.
- Gamma-Exponencial.
- Normal-Normal.

4.8.1. Distribución conjugada Beta-Binomial

Si $p \sim \text{Beta}(\alpha, \beta)$ y $X \sim B(n, p)$, entonces la distribución a posteriori de p dado $X = k$ es también una distribución beta:

$$p|X = k \sim \text{Beta}(\alpha + k, \beta + n - k).$$

Ejemplo 4.6. Supongamos que queremos estimar la proporción de fumadores en nuestra ciudad y que tenemos una distribución a priori de p dada por una distribución beta con parámetros $\alpha = 2$ y $\beta = 5$, es decir, $p \sim \text{Beta}(2, 5)$. Si tomamos una muestra aleatoria de $n = 30$ personas y observamos que $k = 10$ de ellas son fumadoras, la distribución a posteriori de p dado $X = 10$ es

$$p|X = 10 \sim \text{Beta}(2 + 10, 5 + 30 - 10) = \text{Beta}(12, 25).$$

4.8.2. Distribución conjugada Gamma-Poisson

Si $\lambda \sim \text{Gamma}(\alpha, \beta)$ y $X \sim P(\lambda)$, entonces la distribución a posteriori de λ dado $X = k$ es también una distribución gamma:

$$\lambda|X = k \sim \text{Gamma}(\alpha + k, \beta + 1).$$

Si se toman n muestras independientes de la misma distribución $P(\lambda)$, la distribución a posteriori de λ dado $X_1 = x_1, \dots, X_n = x_n$ es también una distribución gamma:

$$\lambda|X_1 = x_1, \dots, X_n = x_n \sim \text{Gamma}\left(\alpha + \sum_{i=1}^n x_i, \beta + n\right).$$

Ejemplo 4.7. Supongamos que queremos estimar la tasa de nacimientos diarios en una ciudad. Tenemos una distribución a priori de λ dada por una distribución gamma con parámetros $\alpha = 2$ y $\beta = 5$, es decir, $\lambda \sim \text{Gamma}(2, 5)$. Si observamos que en una muestra de 5 días se han producido $X_1 = 10, X_2 = 8, X_3 = 9, X_4 = 11, X_5 = 7$ nacimientos, la distribución a posteriori de λ es

$$\begin{aligned} \lambda|X_1 = 10, X_2 = 8, X_3 = 9, X_4 = 11, X_5 = 7 &\sim \text{Gamma}(2 + 10 + 8 + 9 + 11 + 7, 5 + 5) \\ &= \text{Gamma}(47, 10). \end{aligned}$$

4.8.3. Distribución conjugada Normal-Normal

Si $\mu \sim N(\theta, \tau^2)$ y $X \sim N(\mu, \sigma^2)$, entonces la distribución a posteriori de μ dado $X = x$ es también una distribución normal:

$$\mu|X = x \sim N\left(\frac{\theta\sigma^2 + x\tau^2}{\sigma^2 + \tau^2}, \frac{\sigma^2\tau^2}{\sigma^2 + \tau^2}\right).$$

Si se toman n muestras independientes de la misma distribución $N(\mu, \sigma^2)$, la distribución a posteriori de μ dado $X_1 = x_1, \dots, X_n = x_n$ es también normal:

$$\mu|\{X_1 = x_1, \dots, X_n = x_n\} \sim N\left(\frac{\theta\sigma^2 + \bar{x}n\tau^2}{\sigma^2 + n\tau^2}, \frac{\sigma^2\tau^2}{\sigma^2 + n\tau^2}\right),$$

donde $\bar{x}$ es la media muestral.

Ejemplo 4.8. Supongamos que el peso perdido con una dieta sigue una distribución normal $N(\mu, \sigma^2 = 4)$ y queremos estimar el peso medio μ perdido con la dieta. Tenemos una distribución a priori para μ dada por una normal con media $\theta = 10$ y varianza $\tau^2 = 1$, es decir, $\mu \sim N(10, 1)$. Si tomamos una muestra aleatoria de $n = 20$ personas que han seguido la dieta y observamos que el peso medio perdido es $\bar{x} = 8$, la distribución a posteriori de μ es

$$\mu|\{X_1, \dots, X_{20}\} \sim N\left(\frac{10 \cdot 4 + 8 \cdot 20 \cdot 1}{4 + 20 \cdot 1}, \frac{4 \cdot 1}{4 + 20 \cdot 1}\right) = N(8,33; 0,17).$$

4.9. Modelos bayesianos

Un modelo bayesiano es un modelo estadístico que incorpora la incertidumbre sobre los parámetros desconocidos mediante distribuciones de probabilidad a priori, y que utiliza la regla de Bayes para actualizar estas distribuciones a posteriori a medida que se obtiene nueva información a través de los datos observados.

Los modelos bayesianos pueden considerarse modelos de aprendizaje automático, porque aprenden distribuciones con la experiencia que se va adquiriendo a partir de los datos observados.

4.9.1. Elementos de un modelo bayesiano

Para definir un modelo bayesiano es necesario especificar:

1. El espacio de las hipótesis o parámetros desconocidos.
2. La distribución de probabilidad a priori de las hipótesis o parámetros desconocidos, que refleja la incertidumbre inicial sobre ellos antes de observar los datos.
3. La distribución de probabilidad de los datos observados para cada hipótesis o valor de los parámetros desconocidos, que viene dada por la función de verosimilitud.



Figura 4.6: Esquema de un modelo bayesiano.

4.9.2. Distribuciones a priori discretas y continuas

Dependiendo de la naturaleza de las hipótesis o parámetros desconocidos, el espacio de hipótesis puede ser discreto o continuo, lo que da lugar a dos tipos de distribuciones a priori:

- **Distribuciones a priori discretas.** Se utilizan cuando el espacio de hipótesis es discreto, es decir, cuando los parámetros toman un número finito o contable de valores, como en el caso del test diagnóstico con las hipótesis E y $\bar{E}$.
- **Distribuciones a priori continuas.** Se utilizan cuando el espacio de hipótesis es continuo, es decir, cuando los parámetros pueden tomar cualquier valor dentro de un intervalo, como en la estimación de la media o de una proporción de una población.

4.9.3. Distribuciones de los datos condicionadas por las hipótesis

Del mismo modo, según la naturaleza de los datos observados, la función de verosimilitud puede modelarse mediante distribuciones de probabilidad discretas o continuas:

- **Distribuciones de probabilidad discretas.** Se utilizan cuando los datos son discretos, como el número de éxitos en una serie de ensayos independientes.
- **Distribuciones de probabilidad continuas.** Se utilizan cuando los datos son continuos, como la altura o el peso de una persona.

4.9.4. Tipos de modelos bayesianos

Según el tipo de la distribución a priori y de la distribución de los datos que se usa para calcular la función de verosimilitud, tenemos distintos tipos de modelos bayesianos.

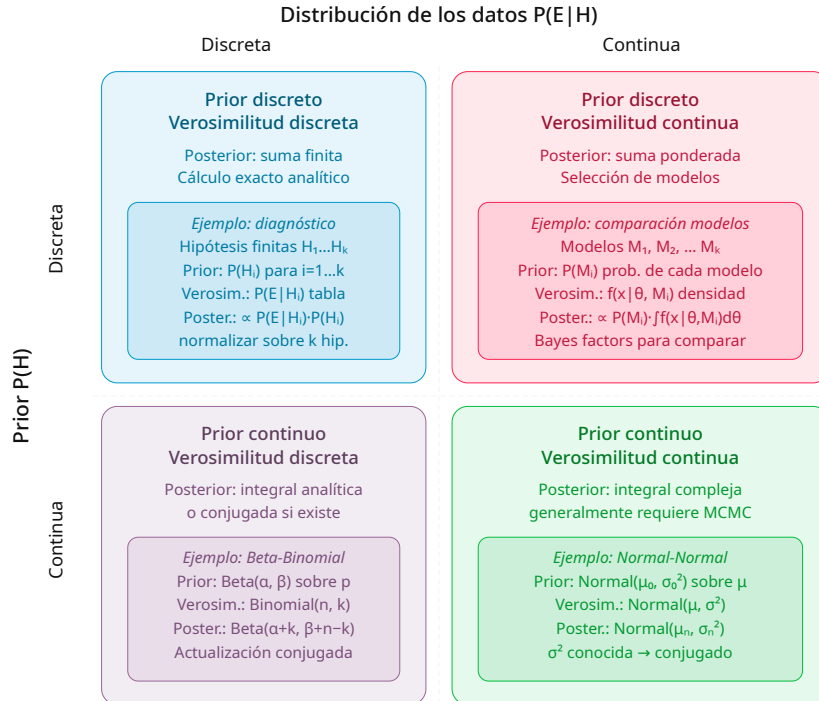


Figura 4.7: Taxonomía de modelos bayesianos.

- **Prior discreto y verosimilitud discreta.** Las parejas más habituales son la uniforme discreta con la binomial o la de Poisson, y las priors definidas por un experto con la multinomial o la categórica. Es el caso del diagnóstico médico con hipótesis finitas.
- **Prior discreto y verosimilitud continua.** Aparece en la comparación y selección de modelos. La pareja más importante es la uniforme sobre modelos con verosimilitud normal o Weibull, para calcular factores de Bayes. El caso *spike-and-slab* con normal es la base de la selección bayesiana de variables en regresión.
- **Prior continuo y verosimilitud discreta.** Aquí están los pares conjugados más importantes de la epidemiología: beta-binomial para proporciones, Dirichlet-multinomial para categorías, gamma-Poisson para conteos y beta-geométrica para el tiempo hasta el primer evento.
- **Prior continuo y verosimilitud continua.** Es el caso más general. Los pares conjugados más importantes son normal-normal, normal-inversa-gamma y gamma-exponencial para tasas. El resto (regresión con covariables, modelos mixtos,

supervivencia) no tiene una forma analítica cerrada y requiere aproximaciones por MCMC.

4.10. Modelo beta-binomial

Cuando el parámetro de interés es una proporción θ , como la proporción de fumadores en una población, y los datos observados son el número de éxitos X en una serie de n ensayos independientes, como el número de personas fumadoras en una muestra, el modelo bayesiano más utilizado es el modelo beta-binomial:

- Prior: $\theta \sim \text{Beta}(\alpha, \beta)$.
- Verosimilitud: $X|\theta \sim B(n, \theta)$.

Ejercicio 4.4. Supongamos que queremos estimar la proporción de fumadores en nuestra ciudad. En varias muestras tomadas en ciudades similares, la proporción media estaba alrededor del 33 %, y la menor y la mayor proporción observadas fueron del 19 % y del 48 %, respectivamente.

¿Qué distribución beta podríamos utilizar para modelizar la distribución a priori?

Tip

Puedes usar la aplicación [distribuciones](#) para probar distintas distribuciones beta y encontrar la que mejor se ajuste a la información.

Solución

Una distribución $\text{Beta}(30, 60)$ tiene media $30/90 \approx 0,33$ y se concentra en el rango de proporciones observado.

```
library(bayesrules)
plot_beta(alpha = 30, beta = 60) +
  labs(x = "Proporción de fumadores ( )", y = "Densidad")
```

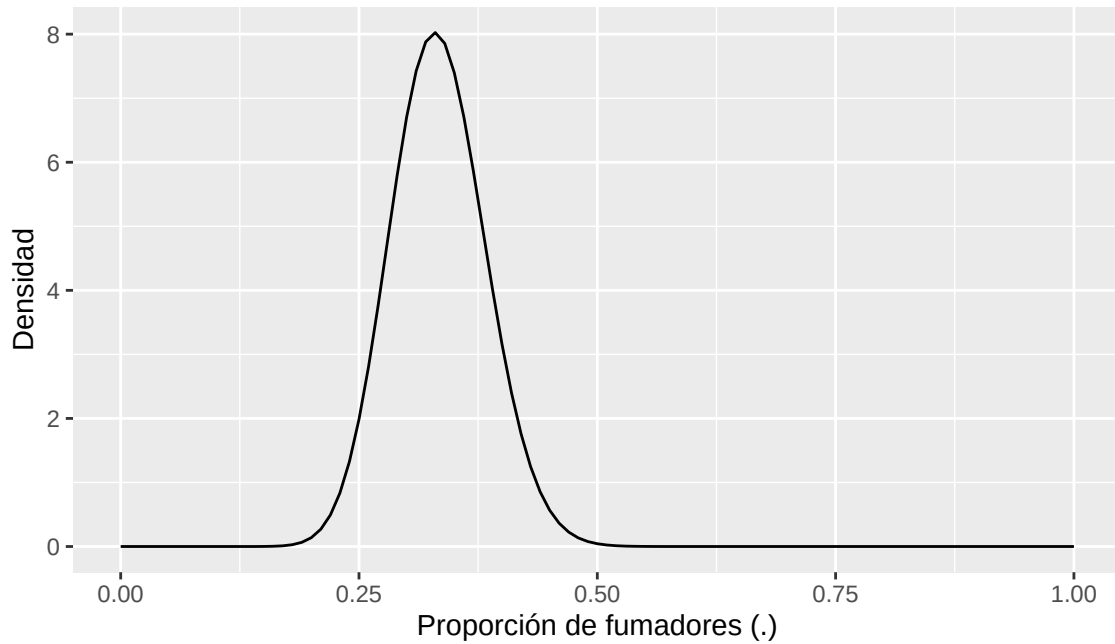


Figura 4.8: Distribución a priori Beta(30, 60) de la proporción de fumadores.

Ejercicio 4.5. Tras realizar un nuevo sondeo en nuestra ciudad, hemos observado que en una muestra de 100 personas, 20 son fumadoras.

¿Cuál es la función de verosimilitud para la proporción de fumadores en la ciudad, θ ?

Solución

Llamemos Y al número de personas fumadoras en una muestra de tamaño $n = 100$. Suponiendo que la proporción de personas fumadoras en la ciudad es θ , Y sigue una distribución binomial con parámetros $n = 100$ y θ :

$$Y|\theta \sim B(n = 100, \theta),$$

por lo que la función de verosimilitud es

$$L(\theta|Y = 20) = P(Y = 20|\theta) = \binom{100}{20} \theta^{20} (1 - \theta)^{80}.$$

```
library(bayesrules)
plot_binomial_likelihood(y = 20, n = 100) +
  labs(x = "Proporción de fumadores (.)", y = "Verosimilitud")
```

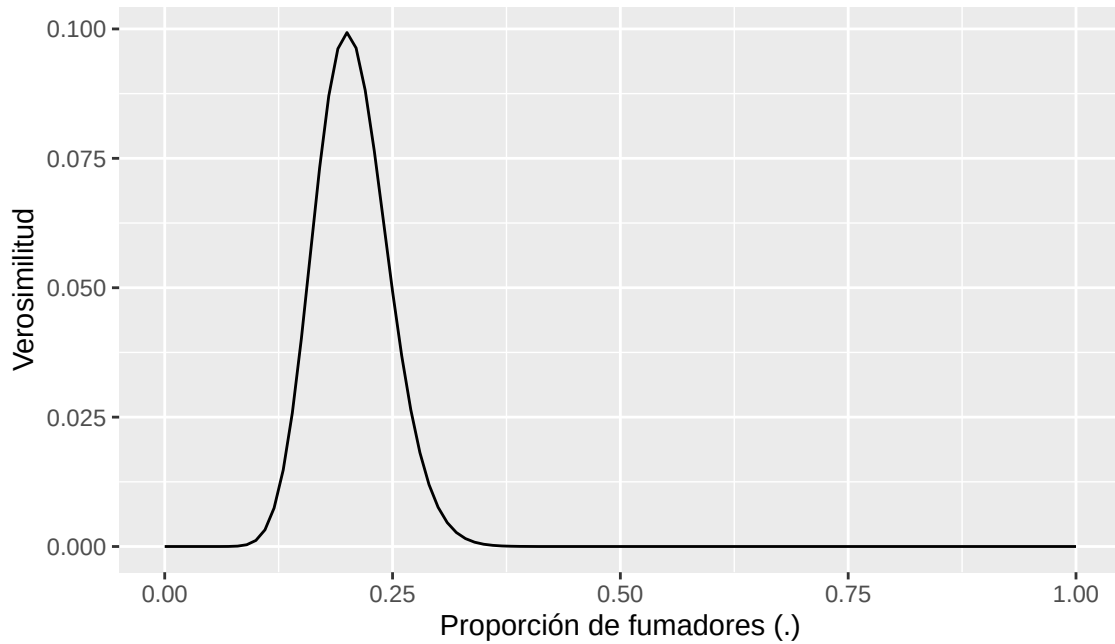


Figura 4.9: Función de verosimilitud para $Y = 20$ fumadoras en 100 personas.

Una vez que tenemos la distribución a priori y la función de verosimilitud, podemos calcular la distribución a posteriori con la regla de Bayes:

$$f(\theta|Y = 20) = \frac{f(\theta)L(\theta|Y = 20)}{P(Y = 20)}.$$

Sin embargo, la constante normalizadora $P(Y = 20)$ es difícil de calcular, porque requiere acumular las probabilidades $f(\theta)L(\theta|Y = 20)$ para cualquier θ . Como θ es una variable continua, esta suma se convierte en la integral

$$P(Y = 20) = \int_0^1 f(\theta)L(\theta|Y = 20) d\theta.$$

Afortunadamente, gracias a la conjugación beta-binomial, sabemos que la distribución a posteriori de θ dado $Y = 20$ es también una distribución beta, con parámetros $\alpha = 30 + 20 = 50$ y $\beta = 60 + 100 - 20 = 140$, es decir, $\theta|Y = 20 \sim \text{Beta}(50, 140)$.

```
plot_beta(alpha = 50, beta = 140) +
  labs(x = "Proporción de fumadores (.)", y = "Densidad")
```

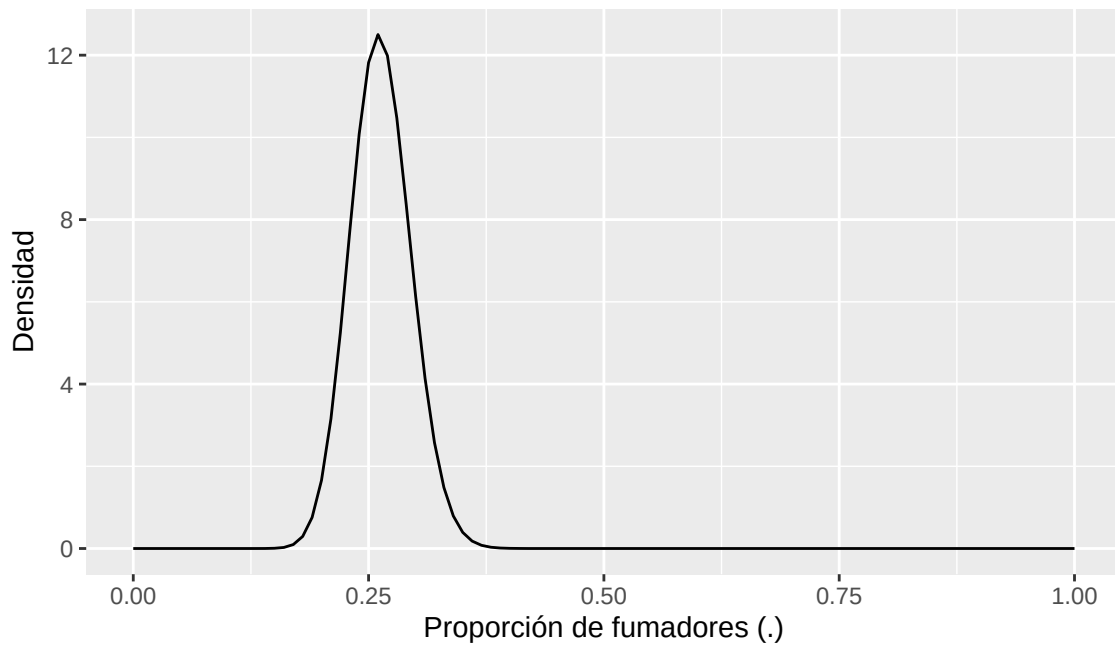


Figura 4.10: Distribución a posteriori $\text{Beta}(50, 140)$ de la proporción de fumadores dado $Y = 20$.

Podemos dibujar las tres distribuciones, a priori, verosimilitud y a posteriori, en la misma gráfica con la función `plot_beta_binomial()` del paquete `bayesrules`.

i `plot_beta_binomial()`

Dibuja la distribución beta a priori, la distribución binomial de la función de verosimilitud y la distribución beta a posteriori dado el número de éxitos observado en la muestra.

Parámetros:

- **alpha**: parámetro α de la distribución beta a priori.
- **beta**: parámetro β de la distribución beta a priori.
- **y**: número de éxitos observados en la muestra.
- **n**: tamaño de la muestra.

```
plot_beta_binomial(alpha = 30, beta = 60, y = 20, n = 100) +
  labs(x = "Proporción de fumadores (.)", y = "Densidad / Verosimilitud")
```

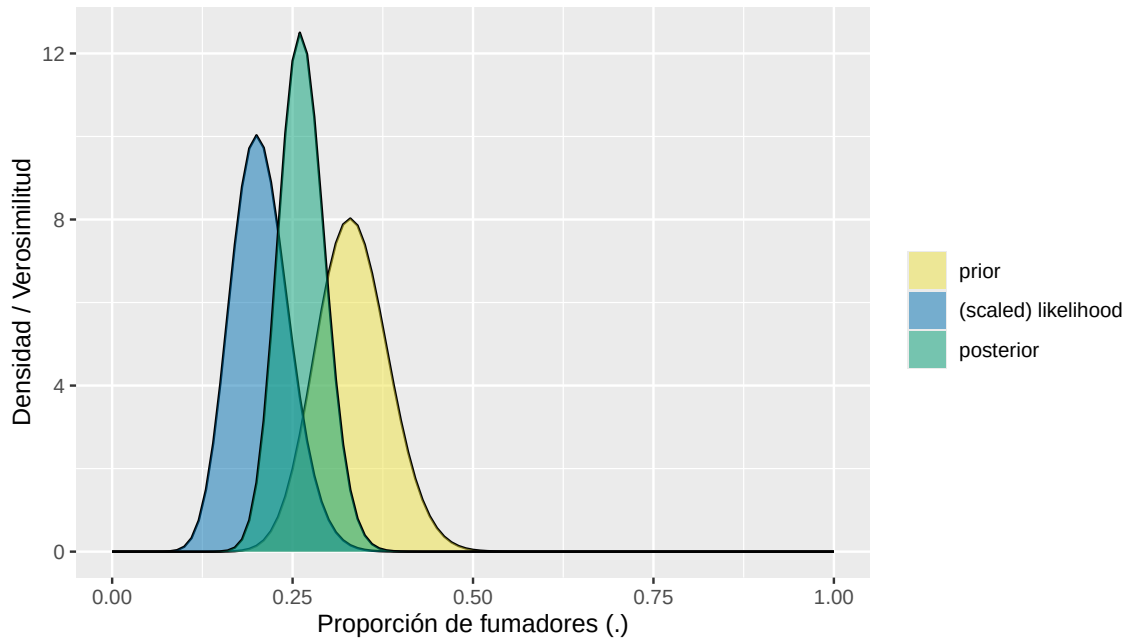


Figura 4.11: Distribución a priori, función de verosimilitud y distribución a posteriori del modelo beta-binomial.

4.11. Simulación de la distribución a posteriori

Cuando no es posible determinar analíticamente la distribución a posteriori, como ocurre con modelos bayesianos con distribuciones no conjugadas, puede simularse mediante técnicas de muestreo. Las técnicas más habituales, por orden de aparición, son:

- Muestreo de rejilla (*grid sampling*).
- Cadenas de Markov Monte Carlo (MCMC).
- Monte Carlo Hamiltoniano (HMC).

4.11.1. Simulación mediante muestreo de rejilla

La técnica de muestreo de rejilla es la más sencilla y consiste en:

1. Definir una rejilla de valores posibles para la hipótesis o parámetro desconocido.
2. Calcular la función de verosimilitud para cada uno de estos valores.
3. Multiplicarla por la probabilidad a priori correspondiente a cada valor.
4. Normalizar el resultado para obtener una distribución a posteriori aproximada.

Ejemplo 4.9. Partiendo de la distribución a priori $\theta \sim \text{Beta}(30, 60)$ y de la función de verosimilitud $L(\theta|Y = 20) = \binom{100}{20}\theta^{20}(1 - \theta)^{80}$, el siguiente código simula la distribución a posteriori de θ dado $Y = 20$ mediante muestreo de rejilla.

```
# Rejilla de 100 posibles valores de entre 0 y 1
rejilla <- seq(0, 1, length.out = 100)
# Función de verosimilitud para cada valor de
verosimilitud <- dbinom(20, size = 100, prob = rejilla)
# Probabilidad a priori para cada valor de
prior <- dbeta(rejilla, shape1 = 30, shape2 = 60)
# Plausibilidad a posteriori para cada valor de
posterior <- prior * verosimilitud
# Normalizamos la plausibilidad a posteriori para obtener una
  ↪ distribución de probabilidad
df_posterior <- tibble( = rejilla, posterior = posterior /
  ↪ sum(posterior))
# Gráfica de la distribución a posteriori
ggplot(df_posterior, aes(x = , y = posterior)) +
  geom_line(color = myblue, linewidth = 1.5) +
  labs(x = "Proporción de fumadores (.)", y = "Probabilidad a posteriori
  ↪ (rejilla)") +
  theme_minimal()
```

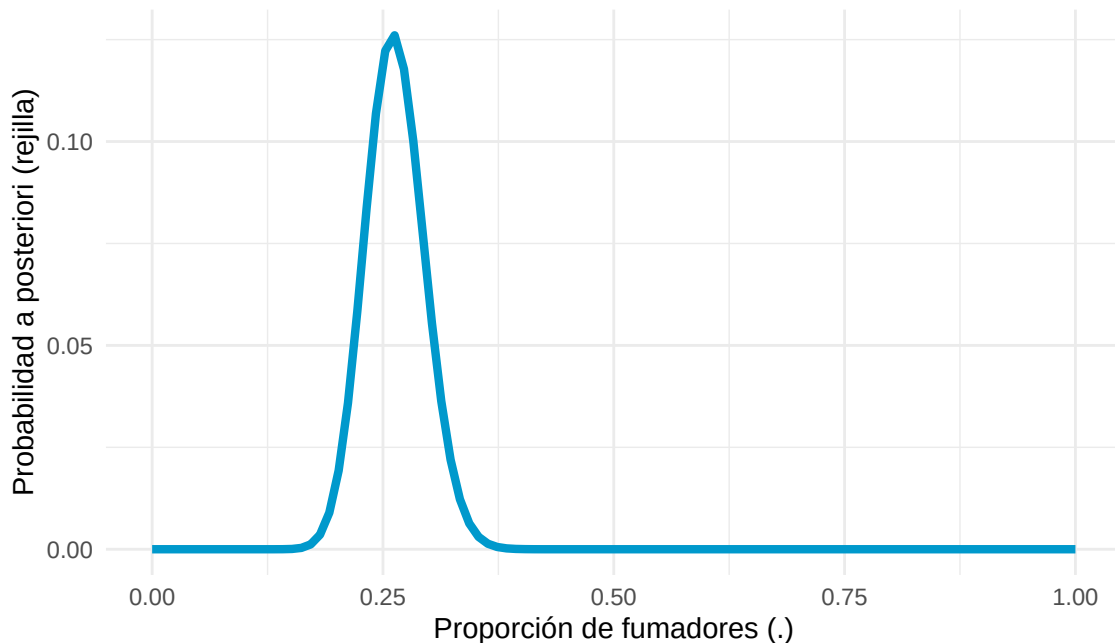


Figura 4.12: Distribución a posteriori de θ dado $Y = 20$ obtenida por muestreo de rejilla.

Para ver si la distribución simulada se parece a la distribución teórica $\text{Beta}(50, 140)$, podemos extraer de ella una muestra aleatoria suficientemente grande y dibujar su histograma:

```
set.seed(123)
# Muestra aleatoria de tamaño 10000
muestra_posterior <- slice_sample(df_posterior, n = 10000, weight_by =
  ↪ posterior, replace = TRUE)
# Histograma de la muestra a posteriori
ggplot(muestra_posterior, aes(x = )) +
  geom_histogram(aes(y = after_stat(density)), binwidth = 0.01, alpha =
    ↪ 0.7, fill = myblue, color = "white") +
  stat_function(fun = dbeta, args = list(50, 140), color = myred,
    ↪ linewidth = 1) +
  labs(x = "Proporción de fumadores (.)", y = "Densidad") +
  theme_minimal()
```

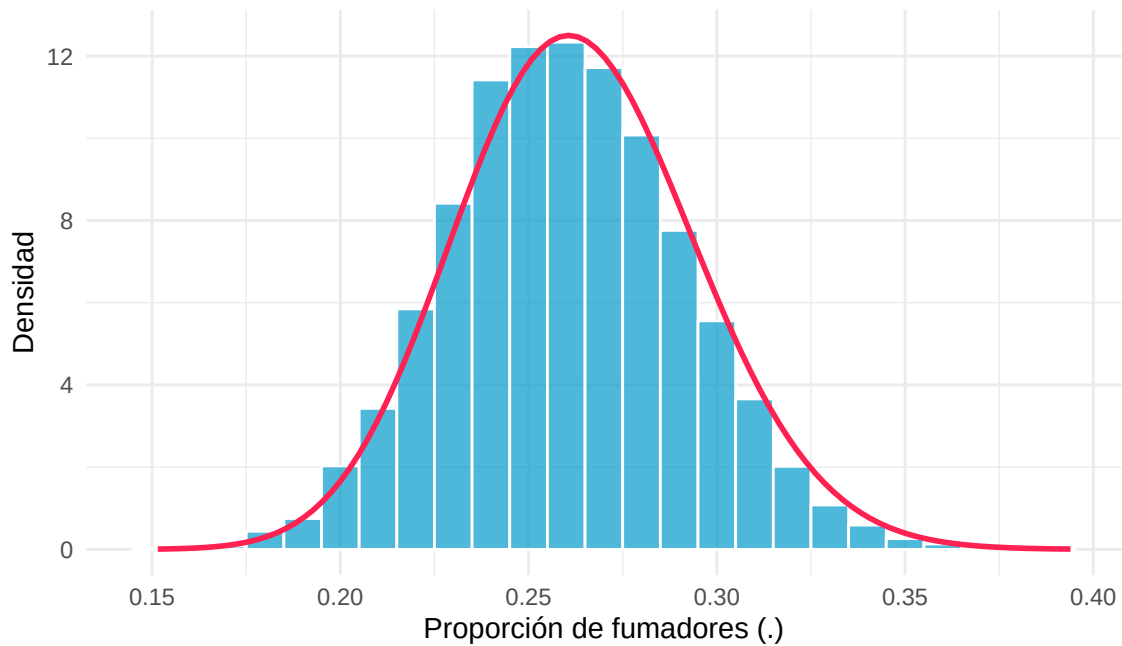


Figura 4.13: Histograma de una muestra de la distribución a posteriori simulada por rejilla, con la densidad teórica $\text{Beta}(50, 140)$.

La distribución de frecuencias de la muestra simulada se ajusta bastante bien a la función de densidad de la distribución teórica $\text{Beta}(50, 140)$.

Ejercicio 4.6. Partiendo de la distribución a posteriori anterior, $\theta \sim \text{Beta}(50, 140)$, simula la distribución a posteriori si se toma otra muestra de tamaño 300 en la que aparecen 140 personas fumadoras.

Solución

```
# Rejilla de 100 posibles valores de  entre 0 y 1
rejilla <- seq(0, 1, length.out = 100)
# Función de verosimilitud para cada valor de  con la nueva muestra
verosimilitud <- dbinom(140, size = 300, prob = rejilla)
# Probabilidad a priori para cada valor de  (distribución a posteriori
  ↪ anterior)
prior <- dbeta(rejilla, shape1 = 50, shape2 = 140)
# Plausibilidad a posteriori para cada valor de
posterior <- prior * verosimilitud
# Normalización de la plausibilidad a posteriori
df_posterior <- tibble( = rejilla, posterior = posterior /
  ↪ sum(posterior))
set.seed(123)
# Muestra aleatoria de tamaño 10000
muestra_posterior <- slice_sample(df_posterior, n = 10000, weight_by =
  ↪ posterior, replace = TRUE)
# Histograma de la muestra a posteriori
ggplot(muestra_posterior, aes(x = )) +
  geom_histogram(aes(y = after_stat(density)), binwidth = 0.01, alpha =
    ↪ 0.7, fill = myblue, color = "white") +
  stat_function(fun = dbeta, args = list(50 + 140, 140 + 300 - 140),
    ↪ color = myred, linewidth = 1) +
  labs(x = "Proporción de fumadores ( )", y = "Densidad") +
  theme_minimal()
```

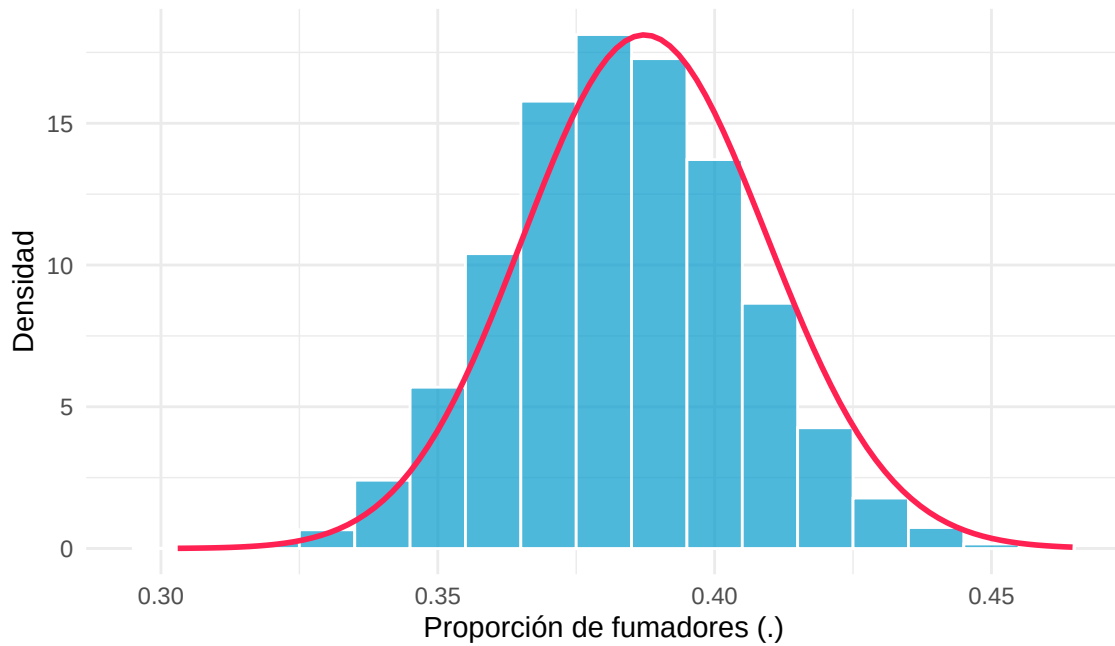


Figura 4.14: Distribución a posteriori tras la nueva muestra de 300 personas, con 140 fumadoras.

4.11.2. Modelo gamma-Poisson

Cuando el parámetro de interés es una tasa λ , como la tasa de nacimientos diarios en una ciudad, y los datos observados son el número de eventos X en un intervalo de tiempo dado, como el número de nacimientos en un día, el modelo bayesiano más utilizado es el modelo gamma-Poisson:

- Prior: $\lambda \sim \text{Gamma}(\alpha, \beta)$.
- Verosimilitud: $X|\lambda \sim P(\lambda)$.

Ejercicio 4.7. Supongamos que queremos estimar la tasa de infartos diarios por cada millón de habitantes. Sabemos que en varios países similares la tasa media de infartos es de 5 por cada 1000 habitantes al año, y que la menor y la mayor tasas observadas fueron de 2 y 10 por cada 1000 habitantes al año, respectivamente.

¿Qué distribución gamma podríamos utilizar para modelizar la distribución a priori?

💡 Tip

Puedes usar la aplicación [distribuciones](#) para probar distintas distribuciones gamma y encontrar la que mejor se ajuste a la información.

Solución

Una distribución $\text{Gamma}(10, 2)$ tiene media $10/2 = 5$ y un rango de valores compatible con la información disponible.

```
plot_gamma(shape = 10, rate = 2) +  
  labs(x = "Tasa de infartos (.)", y = "Densidad")
```

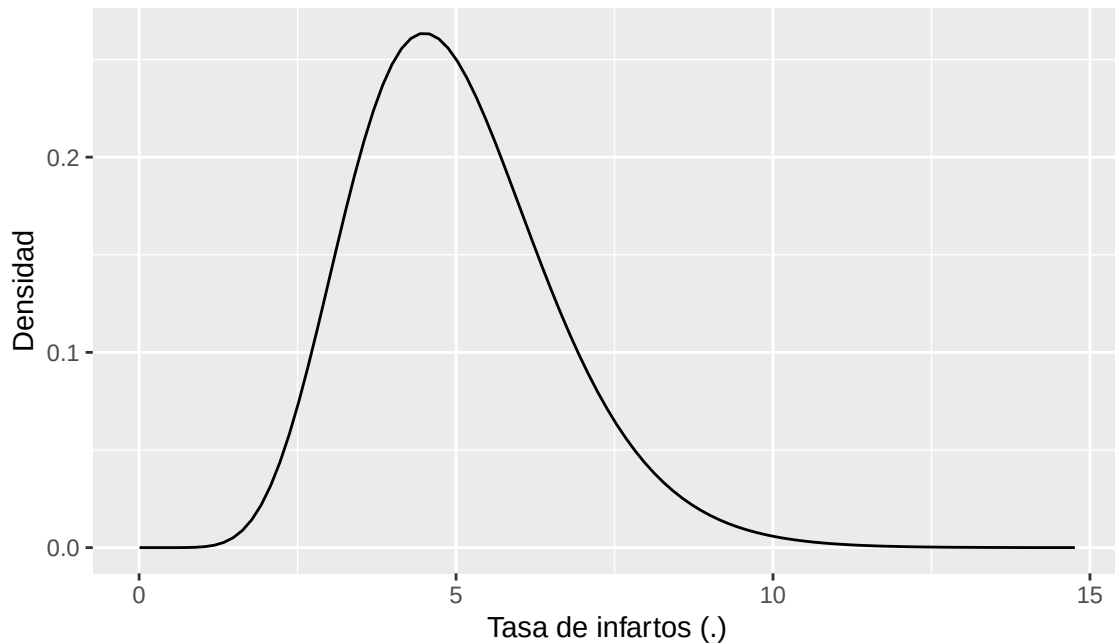


Figura 4.15: Distribución a priori $\text{Gamma}(10, 2)$ de la tasa de infartos.

Ejercicio 4.8. Tras realizar un nuevo sondeo, hemos observado que en una muestra de 5 días se han producido $X_1 = 10, X_2 = 8, X_3 = 9, X_4 = 11, X_5 = 7$ infartos.

¿Cuál es la función de verosimilitud para la tasa de infartos λ ?

Solución

Llamemos Y al número de infartos diarios. Suponiendo que la tasa de infartos diarios es λ , el número de infartos de cada día sigue una distribución de Poisson de parámetro λ , por lo que la función de verosimilitud para λ dados los cinco días observados es

$$\begin{aligned} L(\lambda|Y = (10, 8, 9, 11, 7)) = & P(Y_1 = 10|\lambda)P(Y_2 = 8|\lambda)P(Y_3 = 9|\lambda) \\ & P(Y_4 = 11|\lambda)P(Y_5 = 7|\lambda), \end{aligned}$$

es decir,

$$L(\lambda|Y = (10, 8, 9, 11, 7)) = \frac{e^{-\lambda}\lambda^{10}}{10!} \frac{e^{-\lambda}\lambda^8}{8!} \frac{e^{-\lambda}\lambda^9}{9!} \frac{e^{-\lambda}\lambda^{11}}{11!} \frac{e^{-\lambda}\lambda^7}{7!}$$

$$= \frac{e^{-5\lambda}\lambda^{45}}{10!8!9!11!7!}.$$

```
plot_poisson_likelihood(y = c(10, 8, 9, 11, 7), lambda_upper_bound = 15)
↪ +
  labs(x = "Tasa de infartos (.)", y = "Verosimilitud")
```

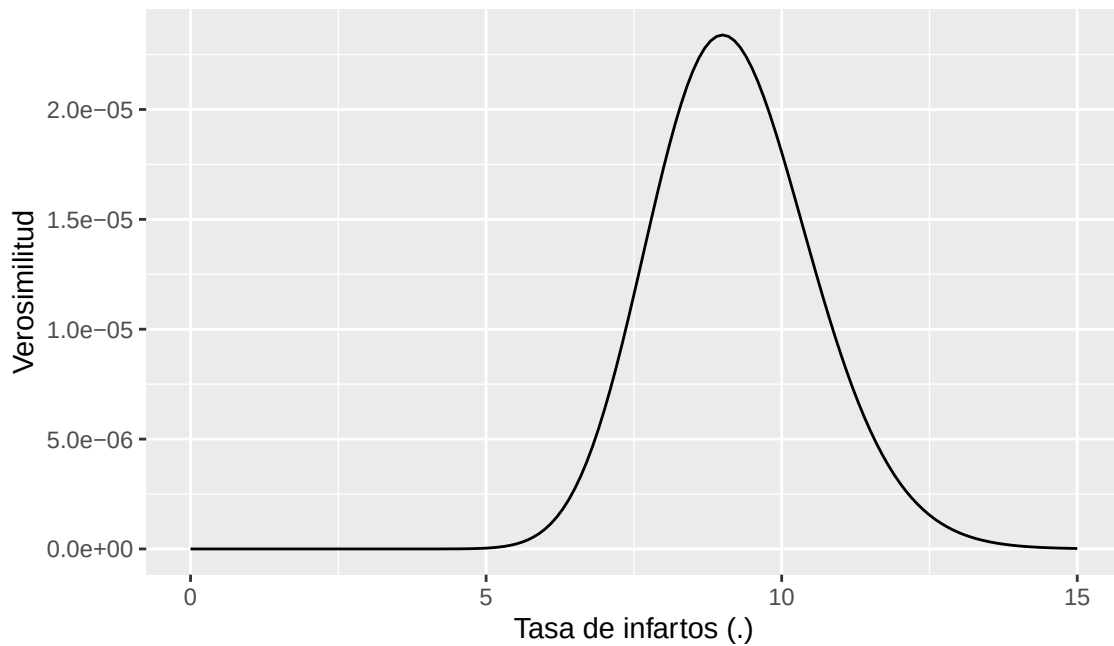


Figura 4.16: Función de verosimilitud de la tasa de infartos para la muestra de 5 días.

Una vez que tenemos la distribución a priori y la función de verosimilitud, la distribución a posteriori se obtiene con la regla de Bayes:

$$f(\lambda|Y = (10, 8, 9, 11, 7)) = \frac{f(\lambda)L(\lambda|Y = (10, 8, 9, 11, 7))}{P(Y = (10, 8, 9, 11, 7))}.$$

De nuevo, la constante normalizadora $P(Y = (10, 8, 9, 11, 7))$ es difícil de calcular, porque requiere la integral

$$P(Y = (10, 8, 9, 11, 7)) = \int_0^{\infty} f(\lambda) L(\lambda | Y = (10, 8, 9, 11, 7)) d\lambda.$$

Pero, gracias a la conjugación gamma-Poisson, la distribución a posteriori de λ dada la muestra es también una distribución gamma con parámetros $\alpha = 10 + 45 = 55$ y $\beta = 2 + 5 = 7$, es decir, $\lambda \sim \text{Gamma}(55, 7)$.

```
plot_gamma(shape = 55, rate = 7) +  
  labs(x = "Tasa de infartos (.)", y = "Densidad")
```

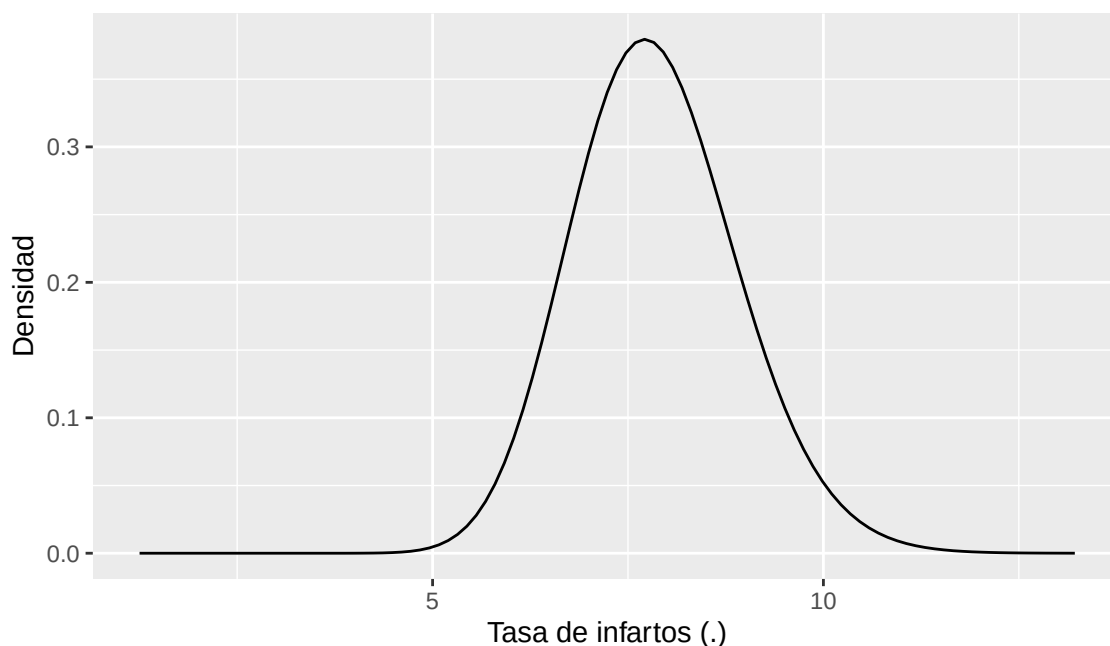


Figura 4.17: Distribución a posteriori $\text{Gamma}(55, 7)$ de la tasa de infartos.

Podemos dibujar las tres distribuciones, a priori, verosimilitud y a posteriori, en la misma gráfica con la función `plot_gamma_poisson()` del paquete `bayesrules`.

i `plot_gamma_poisson()`

Dibuja la distribución gamma a priori, la distribución de Poisson de la función de verosimilitud y la distribución gamma a posteriori, dada una muestra de conteos. Parámetros:

- **shape**: parámetro α de la distribución gamma a priori.
- **rate**: parámetro β de la distribución gamma a priori.
- **sum_y**: suma de los conteos observados en la muestra.

- n : tamaño de la muestra.

```
plot_gamma_poisson(shape = 10, rate = 2, sum_y = 45, n = 5) +
  labs(x = "Tasa de infartos (.)", y = "Densidad / Verosimilitud")
```

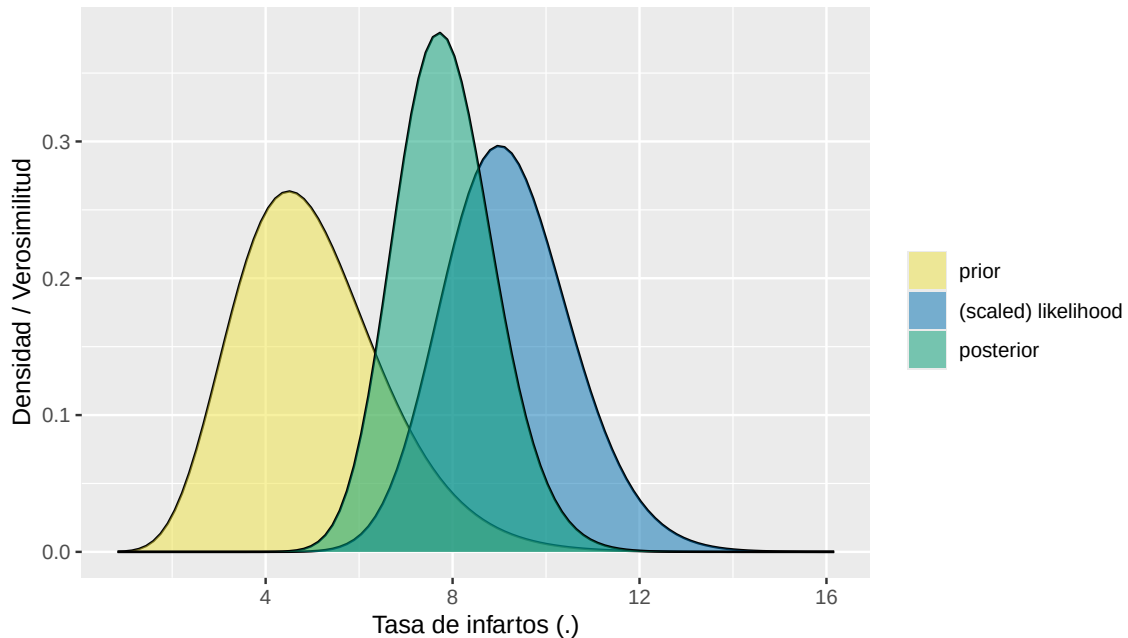


Figura 4.18: Distribución a priori, función de verosimilitud y distribución a posteriori del modelo gamma-Poisson.

4.11.2.1. Simulación de la distribución a posteriori mediante muestreo de rejilla

Ejemplo 4.10. Partiendo de la distribución a priori $\lambda \sim \text{Gamma}(10, 2)$ y de la función de verosimilitud $L(\lambda|Y = (10, 8, 9, 11, 7))$, el siguiente código simula la distribución a posteriori de λ mediante muestreo de rejilla.

```
# Rejilla de 100 posibles valores de entre 1 y 15
rejilla <- seq(1, 15, length.out = 100)
# Función de verosimilitud para cada valor de
verosimilitud <- sapply(rejilla, function(l) prod(dpois(c(10, 8, 9, 11,
  ↪ 7), lambda = l)))
# Probabilidad a priori para cada valor de
prior <- dgamma(rejilla, shape = 10, rate = 2)
# Plausibilidad a posteriori para cada valor de
posterior <- prior * verosimilitud
```

```
# Normalizamos la plausibilidad a posteriori para obtener una
  ↳ distribución de probabilidad
df_posterior <- tibble( = rejilla, posterior = posterior /
  ↳ sum(posterior))
set.seed(123)
# Muestra aleatoria de tamaño 10000
muestra_posterior <- slice_sample(df_posterior, n = 10000, weight_by =
  ↳ posterior, replace = TRUE)
# Histograma de la muestra a posteriori
ggplot(muestra_posterior, aes(x = )) +
  geom_histogram(aes(y = after_stat(density)), binwidth = 0.5, alpha =
  ↳ 0.7, fill = myblue, color = "white") +
  stat_function(fun = dgamma, args = list(55, 7), color = myred,
  ↳ linewidth = 1) +
  labs(x = "Tasa de infartos ( )", y = "Densidad") +
  theme_minimal()
```

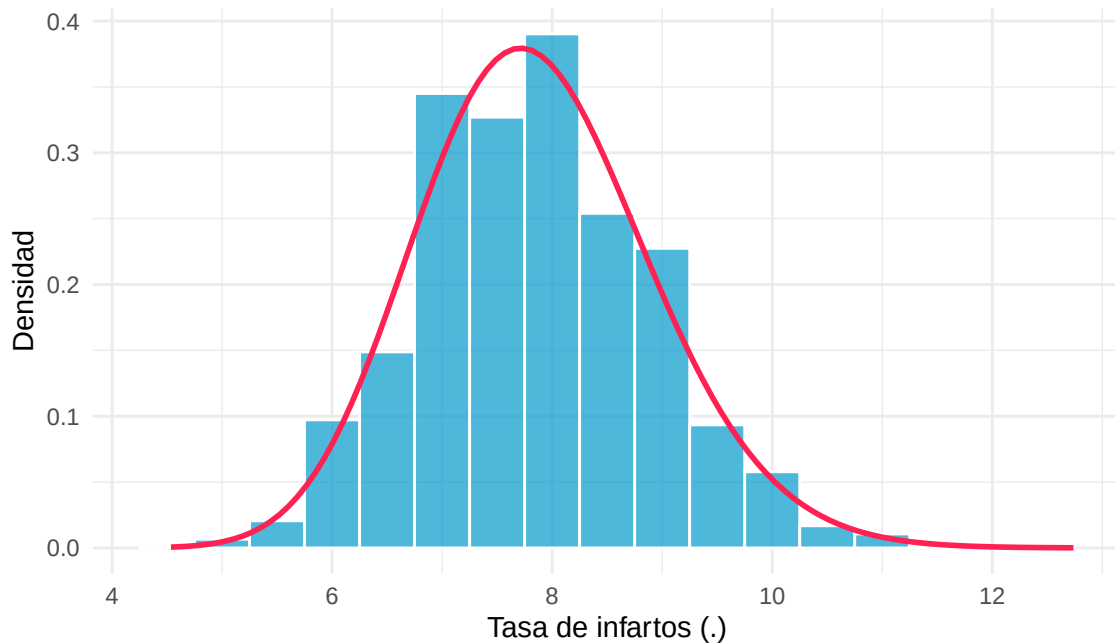


Figura 4.19: Histograma de una muestra de la distribución a posteriori de λ por rejilla, con la densidad teórica $\text{Gamma}(55, 7)$.

Ejercicio 4.9. Partiendo de la distribución a posteriori anterior, $\lambda \sim \text{Gamma}(55, 7)$, simula la distribución a posteriori si se toma otra muestra de tamaño 5 en la que se han producido $X_1 = 5, X_2 = 7, X_3 = 4, X_4 = 6, X_5 = 3$ infartos.

Solución

```
# Rejilla de 100 posibles valores de entre 1 y 15
rejilla <- seq(1, 15, length.out = 100)
# Función de verosimilitud para cada valor de con la nueva muestra
verosimilitud <- sapply(rejilla, function(l) prod(dpois(c(5, 7, 4, 6,
  ↪ 3), lambda = l)))
# Probabilidad a priori para cada valor de (distribución a posteriori
  ↪ anterior)
prior <- dgamma(rejilla, shape = 55, rate = 7)
# Plausibilidad a posteriori para cada valor de
posterior <- prior * verosimilitud
# Normalización de la plausibilidad a posteriori
df_posterior <- tibble( = rejilla, posterior = posterior /
  ↪ sum(posterior))
set.seed(123)
muestra_posterior <- slice_sample(df_posterior, n = 10000, weight_by =
  ↪ posterior, replace = TRUE)
# Histograma de la muestra a posteriori
ggplot(muestra_posterior, aes(x = )) +
  geom_histogram(aes(y = after_stat(density)), binwidth = 0.5, alpha =
    ↪ 0.7, fill = myblue, color = "white") +
  stat_function(fun = dgamma, args = list(55 + 25, 7 + 5), color =
    ↪ myred, linewidth = 1) +
  labs(x = "Tasa de infartos ( )", y = "Densidad") +
  theme_minimal()
```

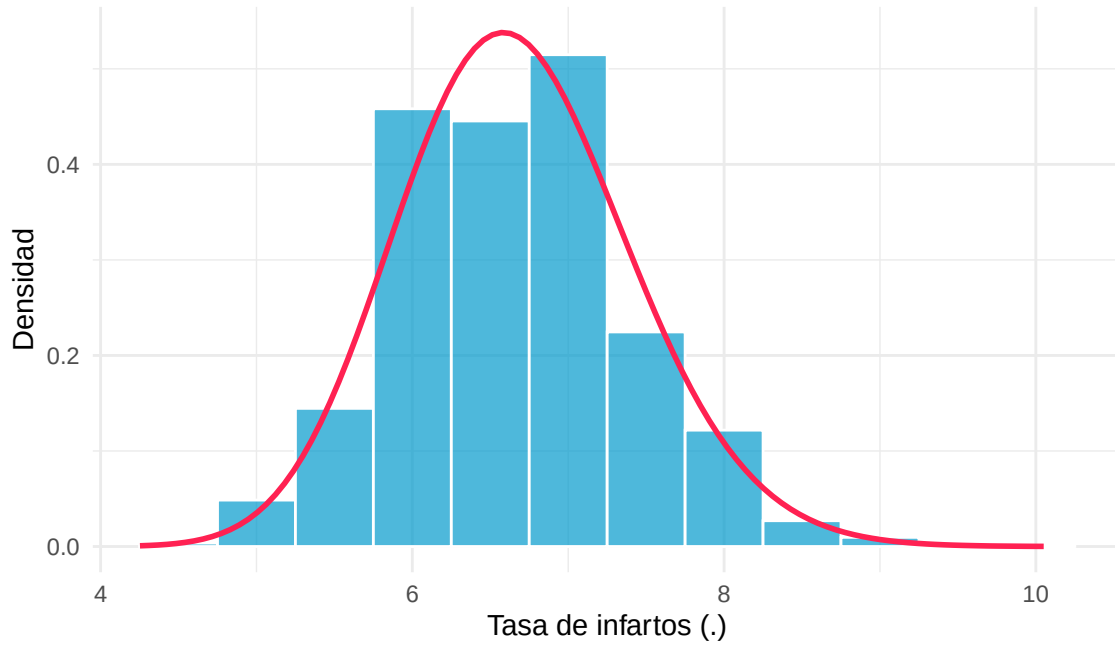


Figura 4.20: Distribución a posteriori de λ tras la nueva muestra de 5 días.

4.12. Independencia del orden de obtención de los datos

Una de las propiedades más importantes de la regla de Bayes es que el orden en el que se obtienen los datos no afecta a la distribución a posteriori final:

$$f(\theta|Y_1, Y_2) = f(\theta|Y_2, Y_1).$$

La demostración es sencilla. Si primero se observa Y_1 ,

$$f(\theta|Y_1) = \frac{f(\theta)L(\theta|Y_1)}{P(Y_1)},$$

y después se observa Y_2 ,

$$\begin{aligned} f(\theta|Y_1, Y_2) &= \frac{\frac{f(\theta)L(\theta|Y_1)}{P(Y_1)}L(\theta|Y_2)}{P(Y_2)} = \frac{f(\theta)L(\theta|Y_1)L(\theta|Y_2)}{P(Y_1)P(Y_2)} \\ &= \frac{f(\theta)L(\theta|Y_2)L(\theta|Y_1)}{P(Y_2)P(Y_1)} = f(\theta|Y_2, Y_1). \end{aligned}$$

4.13. Modelo Normal-Normal

Cuando el parámetro de interés es una media μ , como el nivel medio de colesterol en sangre, y los datos X siguen una distribución normal con desviación típica conocida, el modelo bayesiano más utilizado es el modelo normal-normal:

- Prior: $\mu \sim N(\theta, \tau^2)$.
- Verosimilitud: $X|\mu \sim N(\mu, \sigma^2)$.

Ejercicio 4.10. Supongamos que queremos estimar el nivel medio de colesterol en sangre en una población. Sabemos que en poblaciones similares el nivel medio de colesterol está en un rango que va de 191 a 209 mg/dL.

¿Qué distribución normal podríamos utilizar para modelizar la distribución a priori?

Tip

Puedes usar la aplicación [distribuciones](#) para probar distintas distribuciones normales y encontrar la que mejor se ajuste a la información.

Solución

Una distribución $N(200, 3^2)$ tiene media 200 y desviación típica 3, de modo que aproximadamente el 95 % de su masa cae entre 194 y 206 mg/dL, dentro del rango de la información disponible.

```
library(bayesrules)
plot_normal(mean = 200, sd = 3) +
  labs(x = "Nivel de colesterol (mg/dL)", y = "Densidad")
```

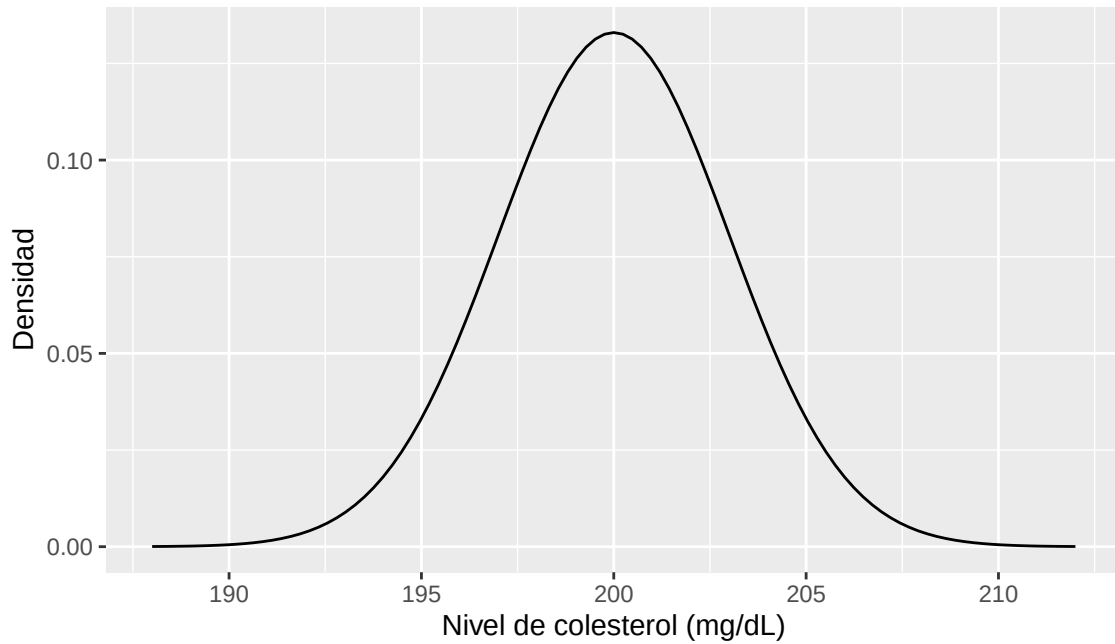


Figura 4.21: Distribución a priori $N(200, 3^2)$ del nivel medio de colesterol.

Ejercicio 4.11. Tras realizar un nuevo sondeo, hemos observado que en una muestra de 10 personas el nivel de colesterol en sangre es 230, 226, 245, 228, 232, 235, 232, 244, 251 y 247 mg/dL. Suponiendo que el nivel de colesterol sigue una distribución normal con desviación típica conocida de 40 mg/dL, algo razonable en una variable biológica continua, ¿cuál es la función de verosimilitud para el nivel medio de colesterol μ ?

Solución

Llamemos Y al nivel de colesterol en sangre. Si $Y \sim N(\mu, 40^2)$, la función de verosimilitud para μ dados los valores observados $y = (y_1, \dots, y_n)$ es

$$L(\mu|y) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \mu)^2\right), \quad \sigma^2 = 40^2.$$

```
muestra <- c(230, 226, 245, 228, 232, 235, 232, 244, 251, 247)
plot_normal_likelihoood(y = muestra, sigma = 40) +
  labs(x = "Nivel medio de colesterol ( )", y = "Verosimilitud")
```

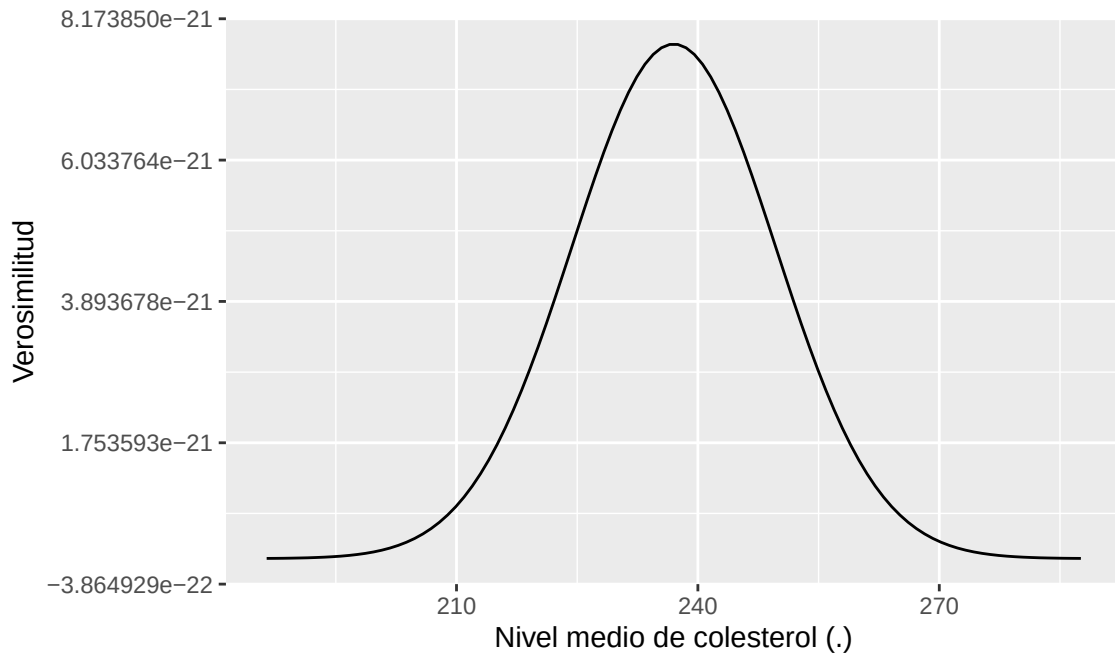


Figura 4.22: Función de verosimilitud para el nivel medio de colesterol con la muestra de 10 personas.

Una vez definidas la verosimilitud y la distribución a priori, la distribución a posteriori se obtiene con la regla de Bayes. La función `plot_normal_normal()` del paquete `bayesrules` dibuja las tres distribuciones en la misma gráfica:

i `plot_normal_normal()`

Dibuja la distribución normal a priori, la distribución normal de la función de verosimilitud y la distribución normal a posteriori, dada una muestra de datos.
Parámetros:

- **mean**: media de la distribución normal a priori.
- **sd**: desviación típica de la distribución normal a priori.
- **y_bar**: media muestral.
- **n**: tamaño de la muestra.
- **sigma**: desviación típica conocida de la distribución normal de la función de verosimilitud.

```
media <- mean(c(230, 226, 245, 228, 232, 235, 232, 244, 251, 247))
plot_normal_normal(mean = 200, sd = 3, y_bar = media, n = 10, sigma =
↪ 40) +
  labs(x = "Nivel medio de colesterol (.)", y = "Densidad /
↪ Verosimilitud")
```

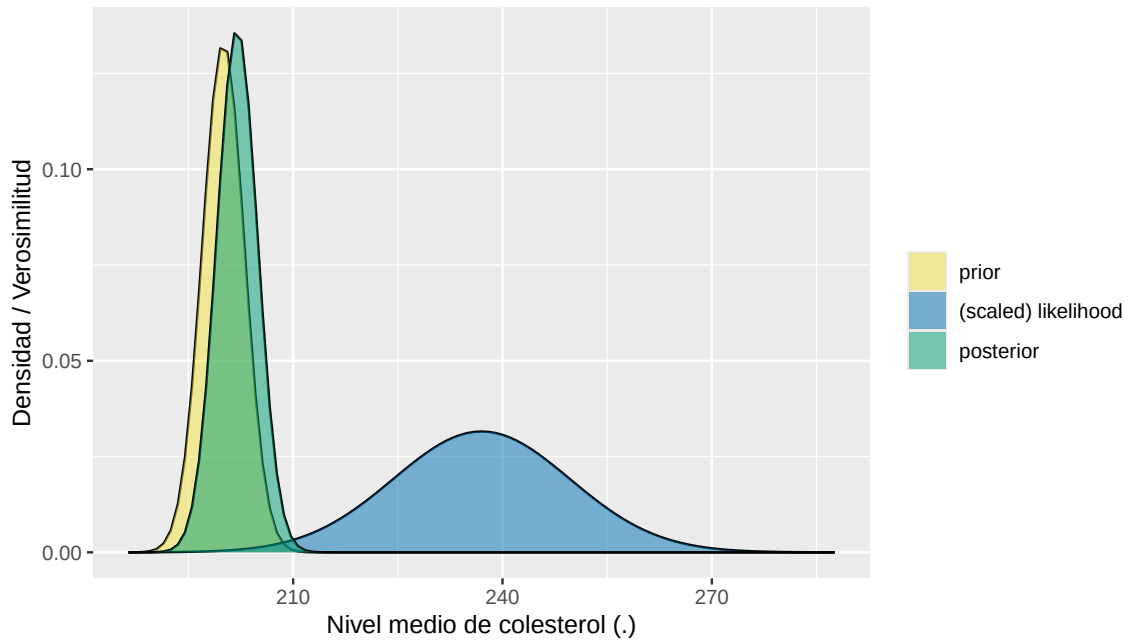


Figura 4.23: Distribución a priori, función de verosimilitud y distribución a posteriori del modelo normal-normal.

4.14. Explosión combinatoria del muestreo de rejilla

El muestreo de rejilla es sencillo, pero no es factible para modelos con muchos parámetros, porque el número de puntos de la rejilla crece exponencialmente con el número de parámetros. Con 100 puntos por parámetro:

Número de parámetros	Número total de puntos en la rejilla
1	100
2	10.000
3	1.000.000
4	100.000.000
5	10.000.000.000

Por esta razón, para modelos con muchos parámetros se utilizan técnicas de muestreo más eficientes, como MCMC o HMC.

4.15. Métodos de Monte Carlo

Los métodos de Monte Carlo generan muestras representativas de la distribución a posteriori con un coste computacional mucho menor que el muestreo de rejilla. Los dos tipos principales son:

- **Markov Chain Monte Carlo (MCMC)**. Es una familia de algoritmos que generan cadenas de Markov cuya distribución estacionaria es la distribución a posteriori deseada. Parten de un valor inicial de cada parámetro y, en cada iteración, proponen un nuevo valor aleatorio cercano al anterior, que se acepta o rechaza según una regla que garantiza que la cadena converja a la distribución a posteriori sin necesidad de conocer su constante de normalización. El algoritmo más conocido de esta familia es Metropolis-Hastings.
- **Hamiltonian Monte Carlo (HMC)**. Refina esta idea usando el gradiente de la log-posterior para proponer saltos grandes y coherentes con la geometría de la distribución, evitando el paseo aleatorio poco eficiente de algoritmos más simples. Es especialmente potente en modelos con muchos parámetros correlacionados.

4.16. El lenguaje Stan

Stan es un lenguaje de programación específico para el modelado y la inferencia bayesiana que implementa algoritmos de muestreo eficientes, como HMC.

- Es un lenguaje declarativo que permite definir modelos bayesianos complejos de manera sencilla.
- Compila los modelos a código C++ optimizado, por lo que genera muestras de distribuciones a posteriori multiparamétricas de forma rápida y eficiente.
- Permite también analizar la convergencia y la calidad de las muestras generadas.

4.16.1. Estructura de un programa en Stan

Un programa en Stan se compone de varios bloques declarativos:

- **data**: declara las variables observadas en los datos, sus dimensiones, tipos y rango posible de valores.
- **parameters**: declara los parámetros del modelo que queremos estimar, sus dimensiones, tipos y rango posible de valores.
- **model**: define la estructura del modelo bayesiano, especificando la distribución a priori de los parámetros y la función de verosimilitud de los datos dados los parámetros.

- **generated quantities:** permite definir variables adicionales que se calculan a partir de los parámetros y los datos, como predicciones o medidas de ajuste del modelo.

Ejemplo 4.11. A continuación se muestra un programa en Stan para el modelo beta-binomial de la proporción de fumadores, con $Y = 20$ fumadoras en una muestra de 100 personas.

```
// modelo-beta-binomial.stan
// Datos
data {
  int<lower = 0, upper = 100> Y;
}
// Parámetros
parameters {
  real<lower = 0, upper = 1> pi;
}
// Modelo
model {
  Y ~ binomial(100, pi);
  pi ~ beta(30, 60);
}
// Posterior predictivo: simular nuevas observaciones
generated quantities {
  int y_rep = binomial_rng(100, pi);
}
```

4.16.2. El paquete rstan

El paquete **rstan** es la interfaz de R para Stan. Permite compilar y ejecutar modelos escritos en Stan desde R, así como analizar los resultados. Los programas se escriben como una cadena de caracteres y se compilan y ejecutan con la función **stan()**.

i stan()

Compila un programa de Stan y simula el muestreo de la distribución a posteriori dados los datos.

- **model_code:** cadena de caracteres con el código del programa de Stan.
- **data:** lista con los datos que se pasan al programa.
- **chains:** número de cadenas de Markov que se simulan.
- **iter:** número de iteraciones de cada cadena.
- **seed:** semilla para la generación de números aleatorios.

Ejemplo 4.12. El siguiente código define un modelo gamma-Poisson para la tasa de infartos diarios, con una distribución a priori $\lambda \sim \text{Gamma}(10, 2)$ y una función de verosimilitud de Poisson. Lo compila y simula el muestreo de la distribución a posteriori.

El resultado es una muestra de 4 cadenas de Markov de 5000 valores cada una, es decir, un total de 20.000 muestras de la distribución a posteriori de λ dado el número de infartos diarios observado.

```
as.data.frame(tasa_infartos_post) |> head()
```

```
      lambda      lp__
1 8.450915 58.22870
2 6.963386 57.99292
3 8.179375 58.33324
4 7.334871 58.25109
5 7.943305 58.37498
6 8.568749 58.16545
```

Podemos visualizar las trazas de las cuatro cadenas con la función `mcmc_trace` del paquete `bayesplot`.

```
library(bayesplot)
mcmc_trace(tasa_infartos_post, pars = "lambda", size = 0.2) +
  labs(x = "Iteración", y = "Valor de ") +
  theme_gray()
```

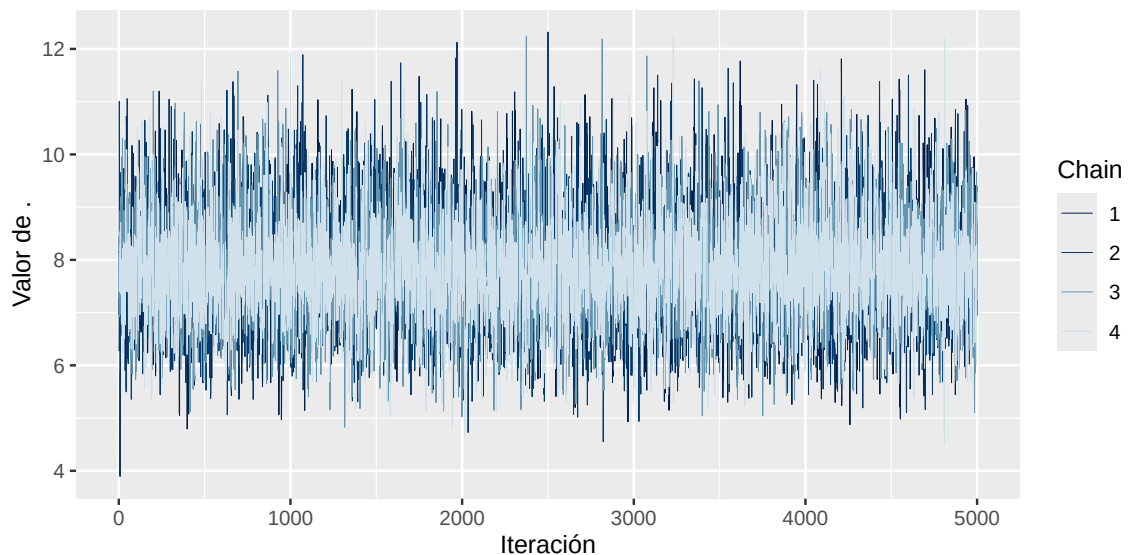


Figura 4.24: Trazas de las cuatro cadenas de Markov para λ .

Si hacemos zoom sobre una de estas cadenas, podemos observar cómo recorre los distintos valores de la distribución a posteriori.

```
mcmc_trace(as.array(tasa_infartos_post)[, 1, , drop = FALSE], pars =
  ↪ "lambda", window = c(1, 100)) +
  labs(x = "Iteración", y = "Valor de ") +
  theme_gray()
```

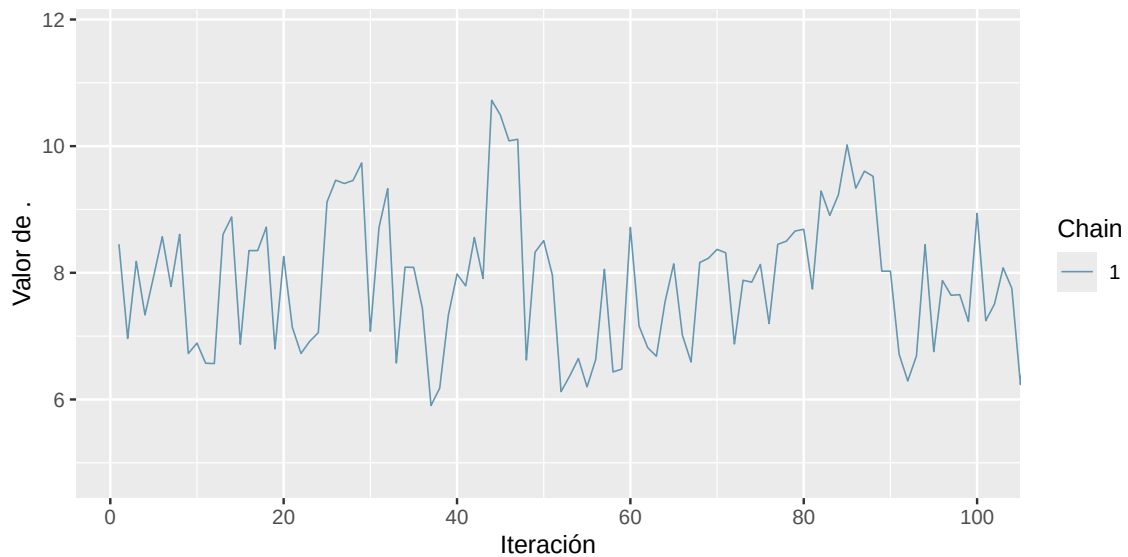


Figura 4.25: Zoom sobre las primeras 100 iteraciones de una de las cadenas de Markov para λ .

Podemos visualizar la distribución a posteriori de λ con la función `mcmc_dens` del paquete `bayesplot`, y compararla con la distribución teórica `Gamma(55, 7)`:

```
mcmc_dens(tasa_infartos_post, pars = "lambda") +
  stat_function(fun = dgamma, args = list(55, 7), color = myred) +
  labs(x = "Tasa de infartos ( )", y = "Densidad") +
  theme_gray()
```

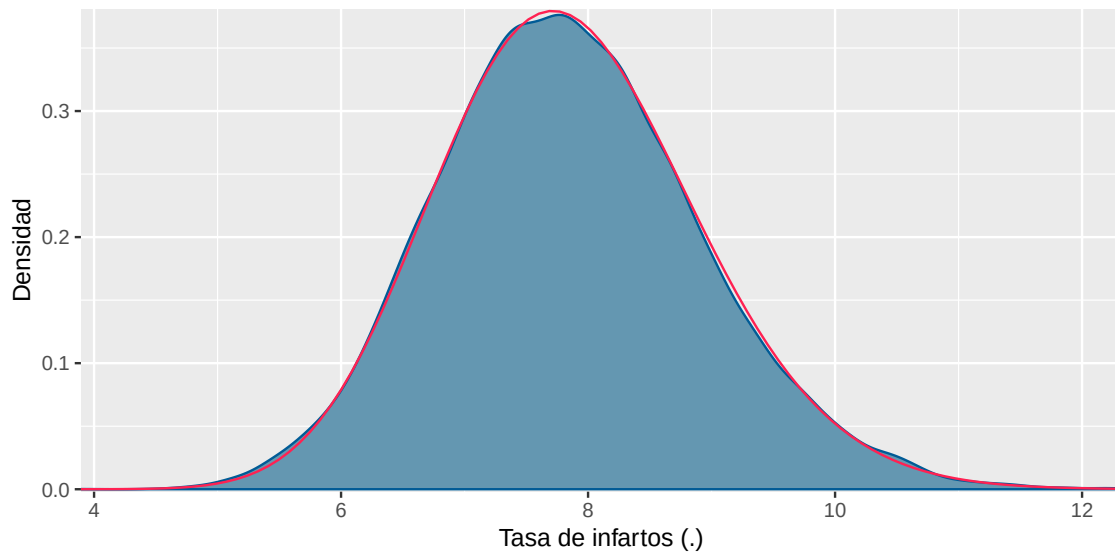


Figura 4.26: Distribución a posteriori de λ simulada con Stan y densidad teórica $\text{Gamma}(55, 7)$.

Ejercicio 4.12. Define en Stan un modelo normal-normal para el nivel medio de colesterol en sangre, con una distribución a priori $N(200, 3^2)$ y una función de verosimilitud normal con desviación típica conocida de 40 mg/dL. Compíllalo y simula el muestreo de la distribución a posteriori dada la muestra de 10 personas del ejercicio anterior.

Solución

```
mcmc_dens(colesterol_post, pars = "mu") +
  stat_function(fun = dnorm, args = list(201.97, sqrt(8.52)), color =
    ↪ myred) +
  labs(x = "Nivel medio de colesterol ( )", y = "Densidad") +
  theme_gray()
```

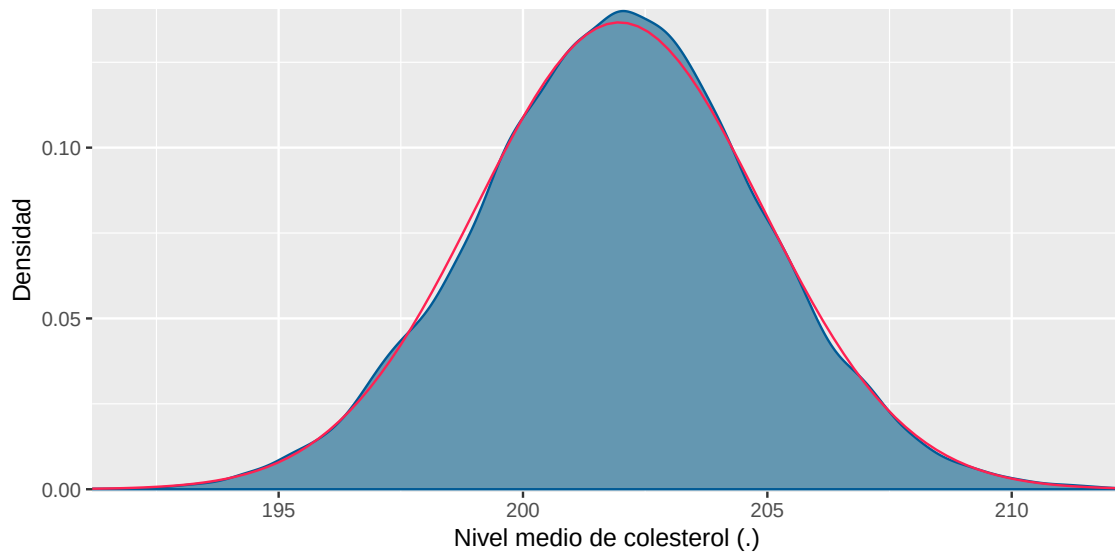


Figura 4.27: Distribución a posteriori de μ simulada con Stan y densidad teórica normal de media 201,97 y varianza 8,52.

4.17. Análisis de las cadenas de Markov

El principal problema de las muestras generadas con MCMC o HMC es que los valores de una cadena no son independientes: cada valor depende del anterior. Aunque el algoritmo busca que la muestra se ajuste a la distribución a posteriori, a veces puede fallar y generar muestras poco representativas.

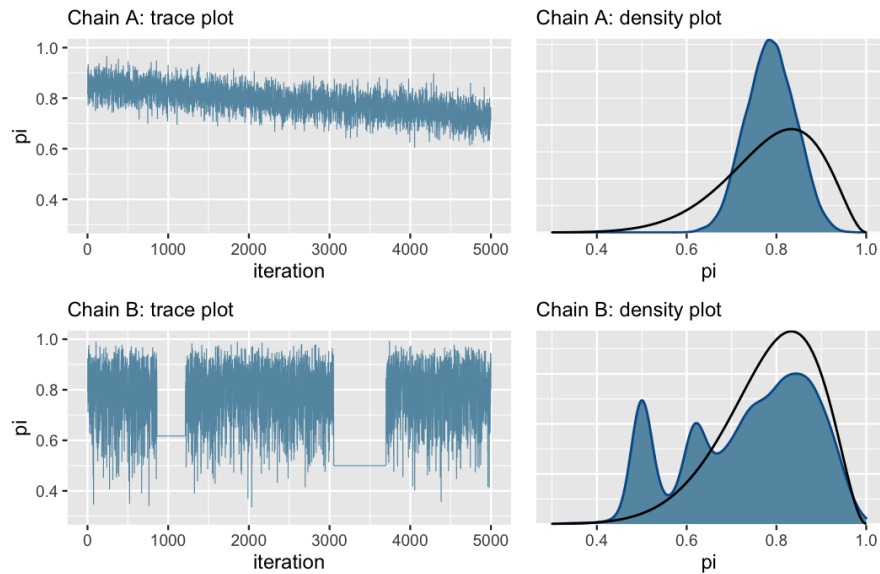


Figura 4.28: Ejemplo de cadenas de Markov anómalas.

Para analizar la calidad de las muestras se utilizan, entre otras, las siguientes técnicas:

- **Análisis de la traza.** Consiste en visualizar las trazas de las cadenas para detectar patrones anómalos, como cadenas que no convergen a la misma distribución, cadenas que no exploran toda la distribución a posteriori o cadenas atrapadas en un valor concreto.
- **Análisis de la autocorrelación.** Consiste en medir la dependencia entre valores separados por un número determinado de iteraciones.
- **Análisis de la eficiencia.** Consiste en calcular el número efectivo de muestras independientes que equivalen a las muestras autocorrelacionadas generadas.
- **Análisis de la consistencia.** Consiste en comparar la variabilidad entre cadenas con la variabilidad dentro de cada cadena.

4.17.1. Análisis de la traza

Para analizar la distribución de frecuencias de cada cadena de Markov podemos usar la función `mcmc_dens_overlay` del paquete `bayesplot`.

i `mcmc_dens_overlay()`

Dibuja las distribuciones de frecuencias de cada cadena de Markov generada.
Parámetros:

- **fit:** objeto de clase `stanfit` con el resultado del muestreo de la distribución

a posteriori.

- **pars**: vector de caracteres con los nombres de los parámetros que se analizan.

Ejemplo 4.13.

```
mcmc_dens_overlay(tasa_infartos_post, pars = "lambda") +  
  labs(x = "Tasa de infartos ( )", y = "Densidad") +  
  theme_gray()
```

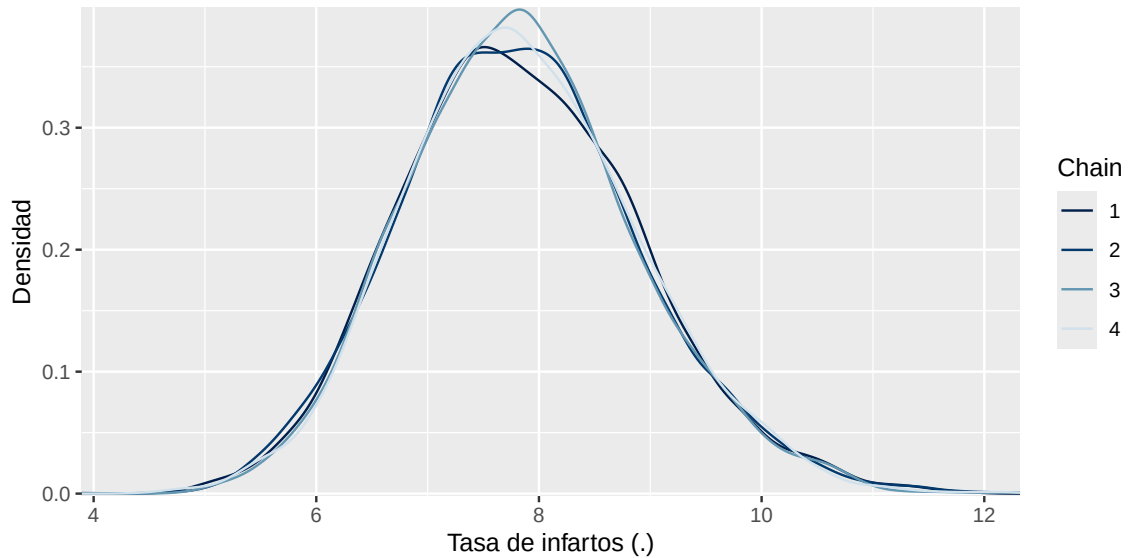


Figura 4.29: Distribución de frecuencias de cada una de las cadenas de Markov para λ .

4.17.2. Análisis de la autocorrelación

Cada valor de la cadena de Markov depende del anterior, lo que genera una cadena de dependencias $\theta_1 \rightarrow \theta_2 \rightarrow \theta_3 \rightarrow \dots \rightarrow \theta_n$. Es de esperar que la dependencia entre θ_i y θ_{i+k} disminuya a medida que k aumenta. Esta dependencia entre los valores de la cadena se conoce como *autocorrelación*.

Definición 4.3 (Autocorrelación de una cadena de Markov). Dada una cadena de Markov θ_i con $i = 1, \dots, n$, la autocorrelación de orden k se define como

$$\rho_k = \frac{\text{Cov}(\theta_i, \theta_{i+k})}{\text{Var}(\theta_i)},$$

donde $\text{Cov}(\theta_i, \theta_{i+k})$ es la covarianza entre el valor θ_i y el valor θ_{i+k} , y $\text{Var}(\theta_i)$ es la varianza de los valores θ_i .

Para representar la autocorrelación de las muestras generadas podemos usar la función `mcmc_acf` del paquete `bayesplot`.

i `mcmc_acf()`

Dibuja la función de autocorrelación de las muestras generadas por MCMC o HMC.
Parámetros:

- **fit**: objeto de clase `stanfit` con el resultado del muestreo de la distribución a posteriori.
- **pars**: vector de caracteres con los nombres de los parámetros que se analizan.
- **lags**: número de retardos para calcular la autocorrelación.

Ejemplo 4.14.

```
mcmc_acf(tasa_infartos_post, pars = "lambda", lags = 10) +  
  labs(x = "Orden o retardo (k)", y = "Autocorrelación (_k)") +  
  theme_gray()
```

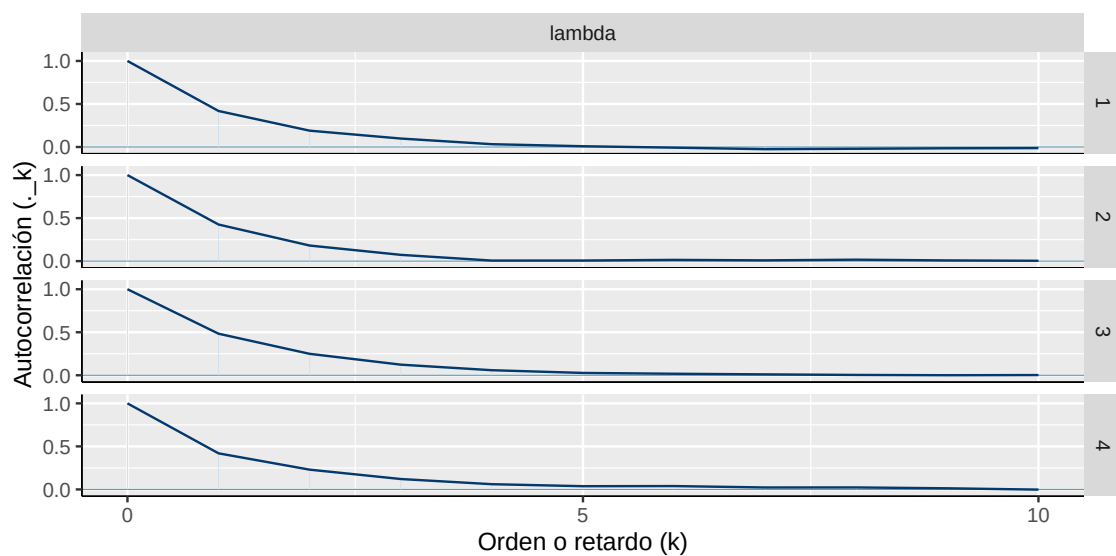


Figura 4.30: Función de autocorrelación de las muestras de MCMC para λ .

4.17.3. Análisis de la eficiencia

La eficiencia de una cadena de Markov se puede evaluar calculando el *número efectivo de muestras independientes* (ESS), es decir, el número de muestras independientes a las que equivale la cadena autocorrelada.

Podemos analizar la eficiencia de las muestras con la función `neff_ratio` del paquete `bayesplot`.

i `neff_ratio()`

Calcula la proporción entre el número efectivo de muestras independientes (ESS) y el número total de muestras generadas por MCMC o HMC.

Parámetros:

- **fit**: objeto de clase `stanfit` con el resultado del muestreo de la distribución a posteriori.
- **pars**: vector de caracteres con los nombres de los parámetros que se analizan.

Ejemplo 4.15.

```
neff_ratio(tasa_infartos_post, pars = "lambda")
```

```
[1] 0.3731105
```

Cuanto más cercano a 1 sea este cociente, más eficiente es el muestreo.

4.17.4. Análisis de la consistencia

La consistencia de las muestras se puede evaluar comparando la variabilidad de los valores muestreados en todas las cadenas combinadas con la variabilidad dentro de cada cadena individual.

Definición 4.4 (Métrica $\hat{R}$). Dada una colección de cadenas de Markov generadas por un mismo proceso (MCMC o HMC), la métrica $\hat{R}$ es la raíz cuadrada de la razón entre la variabilidad total de los valores muestreados en todas las cadenas combinadas y la variabilidad dentro de cada cadena individual:

$$\hat{R} = \sqrt{\frac{\text{Var}_{total}}{\text{Var}_{individual}}}.$$

Si la variabilidad entre cadenas es similar a la variabilidad dentro de cada cadena, $\hat{R} \approx 1$, lo que indica que las cadenas han convergido a la misma distribución a posteriori y las muestras son consistentes.

Podemos calcular $\hat{R}$ con la función `rhat` del paquete `bayesplot`.

i `rhat()`

Calcula la métrica $\hat{R}$ para evaluar la consistencia de las muestras generadas por MCMC o HMC.

Parámetros:

- `fit`: objeto de clase `stanfit` con el resultado del muestreo de la distribución a posteriori.
- `pars`: vector de caracteres con los nombres de los parámetros que se analizan.

Ejemplo 4.16.

```
rhat(tasa_infartos_post, pars = "lambda")
```

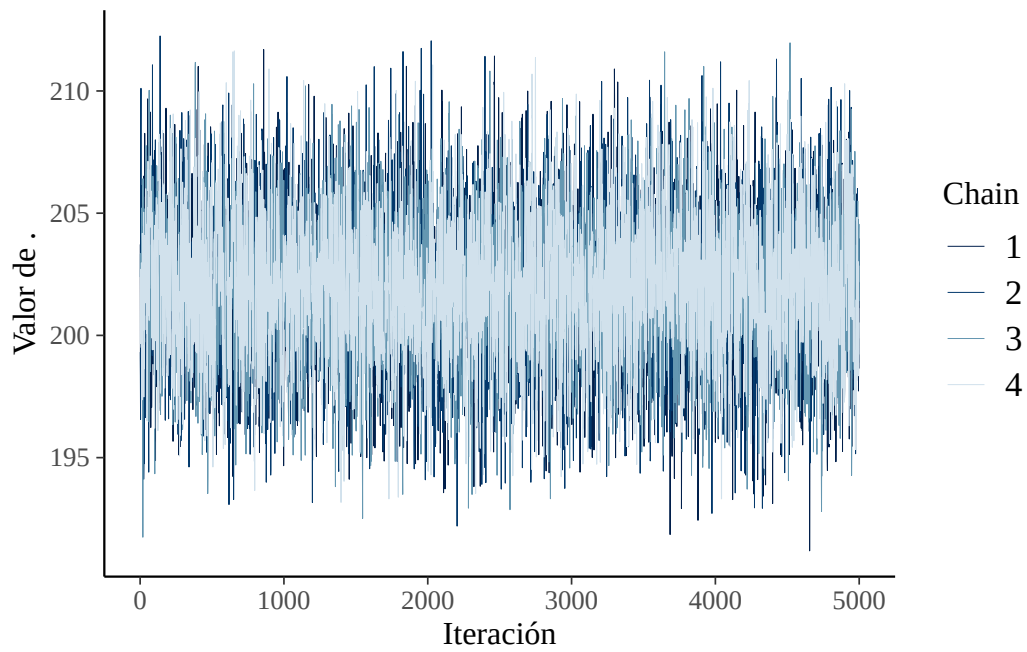
```
[1] 1.000579
```

Si el valor es prácticamente 1, las cuatro cadenas de Markov generadas son consistentes entre sí.

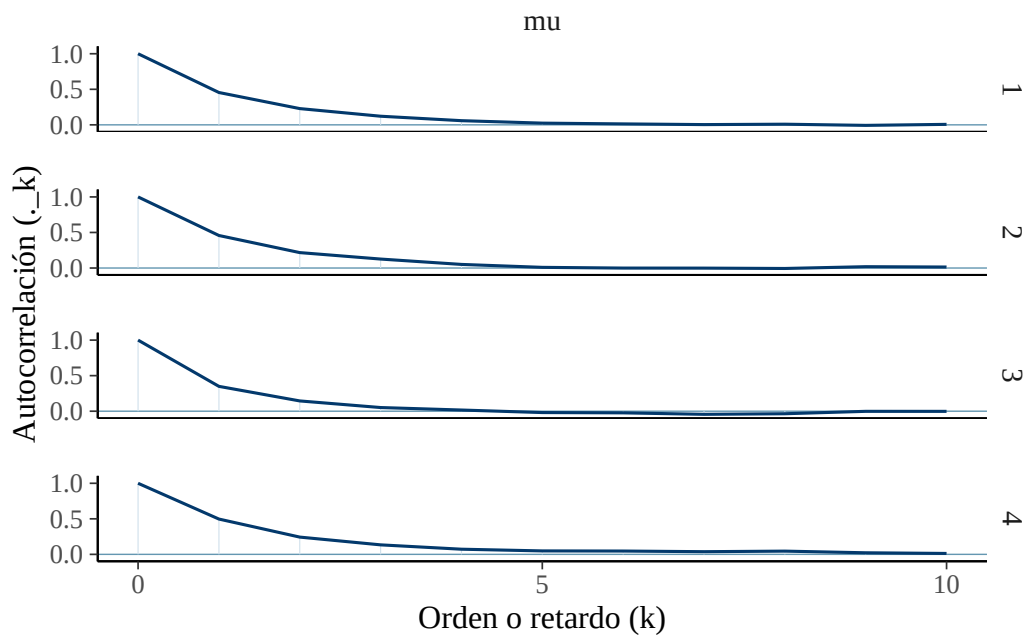
Ejercicio 4.13. Analiza la traza, la autocorrelación, la eficiencia y la consistencia de las cadenas de Markov generadas con Stan para la distribución a posteriori del nivel medio de colesterol en sangre del ejercicio anterior.

Solución

```
# Análisis de la traza
mcmc_trace(colesterol_post, pars = "mu", size = 0.2) +
  labs(x = "Iteración", y = "Valor de ")
```



```
# Análisis de la autocorrelación
mcmc_acf(colesterol_post, pars = "mu", lags = 10) +
  labs(x = "Orden o retardo (k)", y = "Autocorrelación (._k)")
```



```
# Eficiencia
neff_ratio(colesterol_post, pars = "mu")
```

```
[1] 0.3672184
```

```
# Consistencia
rhat(colesterol_post, pars = "mu")
```

```
[1] 1.000358
```

4.18. Dependencia de los resultados de la distribución a priori

Una de las críticas más comunes a la inferencia bayesiana es que los resultados pueden depender de la elección de la distribución a priori, lo que puede introducir un sesgo subjetivo en el análisis.

Los defensores de la estadística bayesiana argumentan que la elección de la distribución a priori debe basarse en el conocimiento previo disponible sobre el fenómeno estudiado, y no en una preferencia personal o arbitraria. En cualquier caso, es importante realizar un **análisis de sensibilidad**: evaluar cómo cambian los resultados a posteriori al cambiar la distribución a priori. Así se identifica si los resultados son robustos o si dependen en gran medida de la elección.

4.19. Distribuciones a priori poco informativas

Una forma de reducir la influencia de la distribución a priori es usar distribuciones poco informativas, que asignan una probabilidad similar a un amplio rango de valores del parámetro. Como ejemplo, consideremos la estimación del nivel medio de colesterol con una distribución a priori uniforme continua $U(150, 300)$.

Ejemplo 4.17. Veamos cómo cambia la distribución a posteriori del nivel medio de colesterol con la muestra de 10 personas, usando una distribución a priori uniforme continua.

```
mcmc_dens(posterior_sim, pars = "mu") +
  stat_function(fun = dnorm, args = list(201.97, sqrt(8.52)), color =
    ↪ myred) +
  labs(x = "Nivel medio de colesterol ( )", y = "Densidad") +
  theme_gray()
```

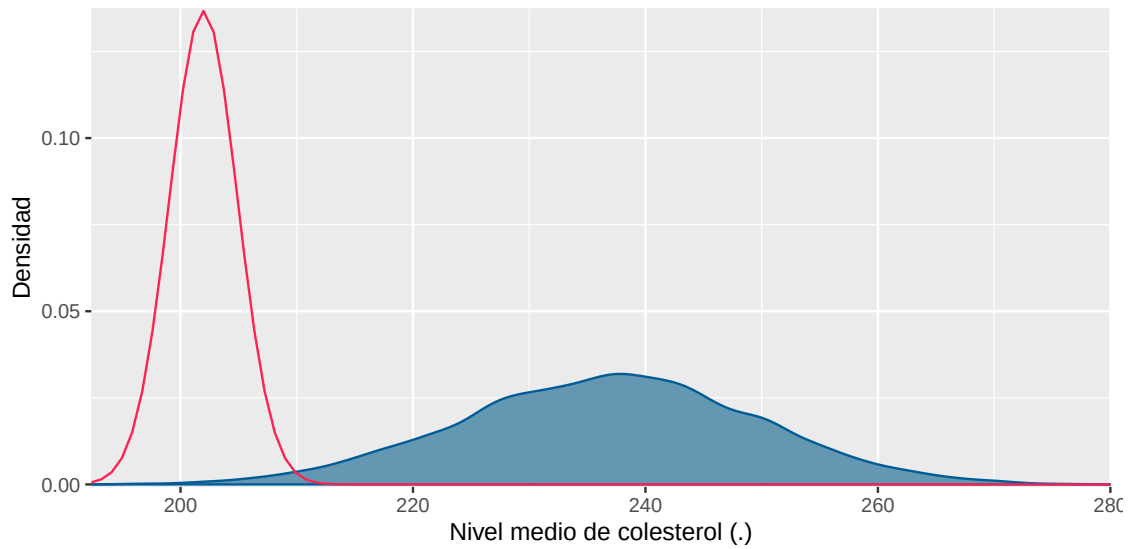


Figura 4.31: Distribución a posteriori de μ con prior uniforme $U(150, 300)$ y 10 observaciones.

Con una muestra más grande, de 100 personas, simulada a partir de la misma distribución, la distribución a posteriori es prácticamente la misma con cualquier prior razonable:

```
mcmc_dens(posterior_sim, pars = "mu") +
  labs(x = "Nivel medio de colesterol (.)", y = "Densidad") +
  theme_gray()
```

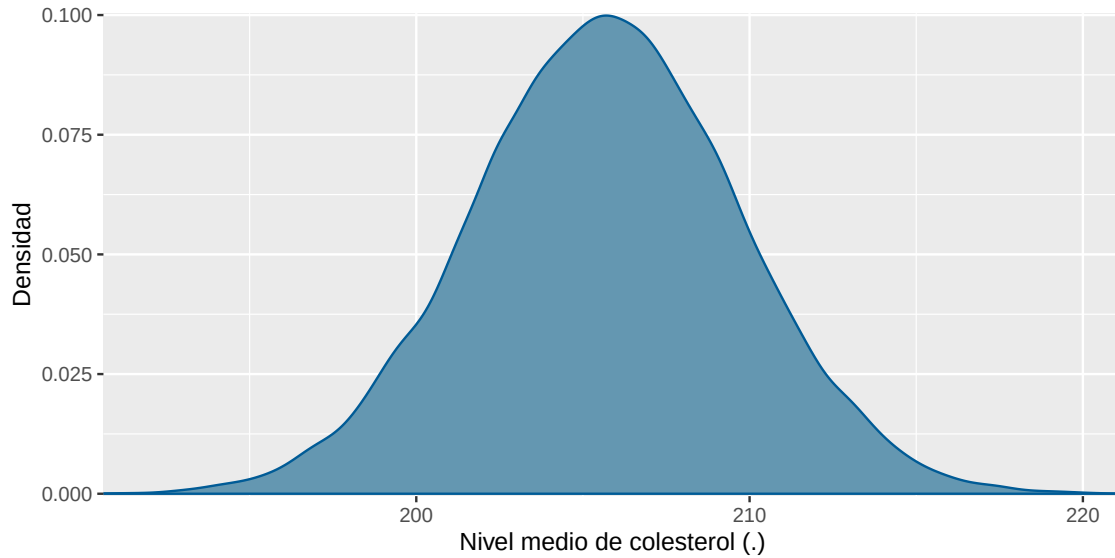


Figura 4.32: Distribución a posteriori de μ con prior uniforme $U(150, 300)$ y 100 observaciones.

Para muestras pequeñas puede haber bastante discrepancia entre las distribuciones a posteriori generadas con distintas distribuciones a priori, pero esa diferencia se disipa a medida que tenemos más datos.

4.20. Tipos de distribuciones a priori

Cuando la muestra es pequeña, la elección de la distribución a priori es determinante. Según el grado de información que incorporan, se distinguen tres tipos:

- **No informativas.** Asignan la misma plausibilidad a todos los valores del parámetro, incluso a valores que no tienen sentido.
- **Poco informativas.** Asignan una probabilidad similar a un amplio rango de valores del parámetro, pero descartan los valores que no tienen sentido.
- **Informativas.** Reflejan un conocimiento previo sólido sobre el parámetro, lo que puede mejorar la precisión de las estimaciones cuando hay poca información en los datos.

4.20.1. Distribuciones a priori no informativas

Una distribución a priori no informativa asigna la misma plausibilidad a todo el rango de valores del parámetro. El caso más extremo es el modelo uniforme impropio $f(\theta) \propto 1$,

que no es una distribución de probabilidad porque no integra 1, pero que produce distribuciones a posteriori adecuadas cuando hay datos suficientes.

¿Cuándo usarla? Cuando no tenemos información previa y los datos son abundantes.

¿Cuándo no usarla? No debería usarse con muestras pequeñas, porque la posterior puede ser inestable o producir estimaciones absurdas.

Ejemplo 4.18. Para estimar la tensión arterial sistólica media de una población se utiliza un modelo uniforme $f(\mu) \propto 1$, que asigna la misma plausibilidad a cualquier valor de $\mu > 0$.

4.20.2. Distribuciones a priori poco informativas

Una distribución a priori poco informativa asigna una probabilidad similar a un amplio rango de valores del parámetro, pero descarta los valores que no tienen sentido. Suele utilizarse una distribución uniforme continua con el rango restringido a los valores plausibles, o una distribución normal con desviación típica grande para reflejar la incertidumbre sobre el parámetro.

¿Cuándo usarla? Cuando no tenemos información previa pero queremos evitar que la posterior sea inestable con muestras pequeñas. Es la opción más habitual cuando no hay información sólida sobre la incertidumbre del parámetro.

¿Cuándo no usarla? No debería usarse cuando tenemos información previa sólida, porque no aprovecha esa información para mejorar la precisión de las estimaciones, especialmente con pocos datos.

Ejemplo 4.19. Para estimar la tensión arterial sistólica media de una población se utiliza un modelo uniforme $\mu \sim U(70, 200)$, que asigna la misma plausibilidad a cualquier valor de μ entre 70 y 200 mmHg, y descarta los valores menores de 70 o mayores de 200 mmHg, que no tienen sentido en este contexto.

4.20.3. Distribuciones a priori informativas

Una distribución a priori informativa refleja un conocimiento previo sólido sobre el parámetro de interés (resultados de estudios anteriores, valores de referencia clínica u opinión de expertos). Puede mejorar la precisión de las estimaciones cuando hay poca información en los datos. Suelen elegirse distribuciones conjugadas con la distribución de los datos.

¿Cuándo usarla? Cuando existe evidencia previa sólida (metaanálisis, registros históricos, valores fisiológicos bien establecidos). Siempre debe documentarse y justificarse.

¿Cuándo no usarla? No debería usarse cuando no existe información previa sólida, porque puede introducir un sesgo subjetivo y producir estimaciones poco fiables, especialmente con pocos datos.

Ejemplo 4.20. Para estimar la tensión arterial sistólica media de una población se utiliza un modelo normal $\mu \sim N(120, 15^2)$, que refleja el conocimiento previo sobre la tensión arterial sistólica media en poblaciones similares, con una media de 120 mmHg y una desviación típica de 15 mmHg para reflejar la incertidumbre sobre ese valor.

4.21. Distribuciones a priori recomendadas

Situación	Prior recomendado
Media poblacional	<code>normal(valor_referencia, incertidumbre)</code>
Desviación típica	<code>half_t(3, 0, 1)</code> o <code>exponential(1)</code>
Proporción sin información previa	<code>beta(1, 1)</code> o <code>beta(0.5, 0.5)</code>
Proporción con información previa	<code>beta()</code> calibrada con la media y la varianza esperadas
Coefficientes de regresión (datos estandarizados)	<code>normal(0, 1)</code>
Coefficientes de regresión (datos sin estandarizar)	<code>normal(0, 2.5)</code> , recomendación de Gelman
Muchos predictores, selección de variables	<code>horseshoe(df=1)</code>
Regresión logística, coeficientes	<code>normal(0, 2.5)</code> o <code>student_t(7, 0, 2.5)</code>

4.22. Distribuciones a priori en epidemiología

Parámetro	Distribución prior	Parámetros habituales	Conjugada con	Aplicación en epidemiología
Prevalencia, proporción	$\text{Beta}(\alpha, \beta)$	$\text{Beta}(1,1)$ uniforme · $\text{Beta}(0,5;0,5)$ Jeffreys · $\text{Beta}(2,5)$ informativa	Binomial / Bernoulli	Prevalencia de enfermedad, tasa de ataque, sensibilidad y especificidad de un test

Parámetro	Distribución prior	Parámetros habituales	Conjugada con	Aplicación en epidemiología
Vector de proporciones	$\text{Dirichlet}(\alpha_1, \dots, \alpha_k)$	$\text{Dir}(1, \dots, 1)$ uniforme · $\text{Dir}(0, 5, \dots, 0, 5)$ Jeffreys	Multinomial / Categorica	Distribución de serotipos, variantes, grupos de riesgo
Tasa de incidencia, tasa de mortalidad	$\text{Gamma}(\alpha, \beta)$	$\text{Gamma}(0, 01; 0, 01)$ difusa · $\text{Gamma}(2; 0, 5)$ débil	Poisson / Exponencial	Incidencia de casos por 100.000 habitantes, mortalidad, tasa de hospitalización
Número reproductivo R_0 , $R(t)$	$\text{Gamma}(\alpha, \beta)$	$\text{Gamma}(1; 0, 2)$ · $\text{Gamma}(2; 0, 5)$	Poisson (ecuación de renovación)	Transmisibilidad de enfermedades infecciosas, estimación de $R(t)$
Media poblacional (tensión arterial, colesterol, glucemia)	$\text{Normal}(\mu_0, \sigma_0^2)$	N(valor de referencia, σ amplio)	Normal (σ^2 conocida)	Estimación de la media de biomarcadores clínicos
Media y varianza poblacional	Normal-inversa-gamma($\mu_0, \lambda, \alpha, \beta$)	Hiperparámetros basados en valores de referencia	Normal completa	Biomarcadores con media y varianza desconocidas
Log-odds, coeficiente de regresión logística	$\text{Normal}(0, \sigma^2)$	$N(0; 2, 5^2)$ de Gelman · $N(0, 1)$ estandarizado	— (MCMC)	Factores de riesgo en estudios de casos y controles, OR ajustado
Log-hazard ratio, coeficiente de supervivencia	$\text{Normal}(0, \sigma^2)$	$N(0; 2, 5^2)$	— (MCMC)	Análisis de supervivencia: tiempo hasta el evento, censura

Parámetro	Distribución prior	Parámetros habituales	Conjugada con	Aplicación en epidemiología
Desviación típica, varianza residual	half-normal($0, \sigma$) · half- $t(\nu, 0, \sigma)$	half-N($0,1$) · half- $t(3, 0, 1)$ de Gelman	— (MCMC)	Variabilidad residual en regresión, modelos mixtos
Varianza de efectos aleatorios	half- $t(\nu, 0, \sigma)$ · gamma inversa(α, β)	half- $t(3, 0, 1)$ recomendado · IG($0,01; 0,01$) clásica	— (MCMC)	Heterogeneidad entre centros, hospitales o regiones en modelos jerárquicos
Correlación entre efectos aleatorios	LKJ(η)	LKJ(1) uniforme · LKJ(2) concentrada	— (MCMC)	Correlación entre interceptos y pendientes en modelos mixtos
Eficacia vacunal (escala log-OR)	Normal($0, \sigma^2$) · Beta (escala de probabilidad)	N(log($0,5$), $0,5^2$) informativa	— (MCMC)	Ensayos clínicos de vacunas, estudios de efectividad vacunal
Sensibilidad y especificidad de un test	Beta(α, β)	Beta(18, 2) para Se 90 % · Beta(38, 2) para Sp 95 %	Binomial	Evaluación de pruebas diagnósticas, modelos de clasificación errónea
Coeficientes con muchas variables	horseshoe(df) · Normal($0,1$)	horseshoe(df=1) · N($0,1$) estandarizado	— (MCMC)	Selección de factores de riesgo, estudios de exposición múltiple
Parámetro de forma de Weibull (supervivencia)	Gamma(α, β) · half-normal	Gamma(1,1) · half-N($0,1$)	— (MCMC)	Modelos paramétricos de supervivencia, tiempo hasta recaída o muerte

4.23. Referencias del capítulo

- Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A., y Rubin, D.B. (2013). *Bayesian Data Analysis*, 3.^a ed. CRC Press.
- McElreath, R. (2020). *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*, 2.^a ed. CRC Press.
- Carpenter, B. et al. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1).

5 Inferencia bayesiana

La inferencia estadística consiste en extraer conclusiones sobre una población a partir de una muestra. En este capítulo se compara la estimación y el contraste de hipótesis frecuentistas con sus equivalentes bayesianos, y se termina con la predicción de nuevas observaciones. En el enfoque bayesiano, el resultado de la inferencia es siempre una distribución de probabilidad sobre los parámetros o sobre las predicciones.

5.1. Estimación de parámetros

5.1.1. Estimación frecuentista

En estadística frecuentista, la estimación de un parámetro desconocido es un valor que se obtiene a partir de la muestra observada y que se utiliza como aproximación del valor real del parámetro. La función que se aplica a la muestra para obtener la estimación se llama *estimador*.

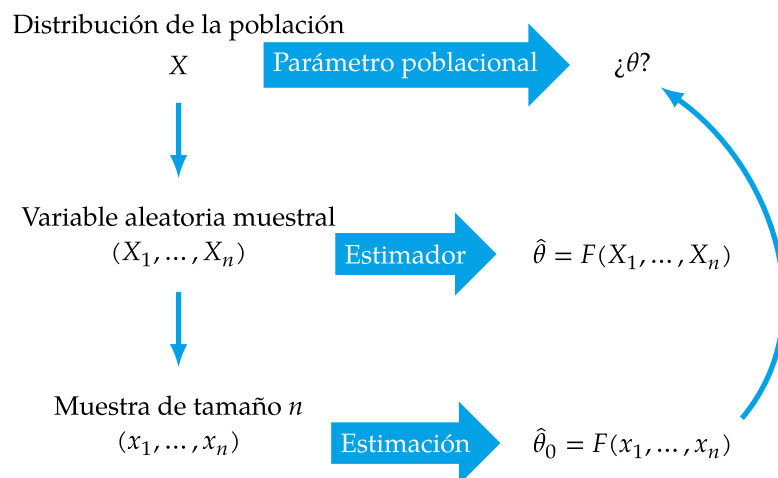


Figura 5.1: Estimación frecuentista.

La estimación de un parámetro puede ser puntual (un único valor) o por intervalos (intervalos de confianza).



Figura 5.2: Estimación puntual y por intervalos.

Definición 5.1 (Intervalo de confianza). Dados dos estimadores $li(X)$ y $ls(X)$, se dice que el intervalo $[li(X), ls(X)]$ es un *intervalo de confianza* para el parámetro θ con nivel de confianza $1 - \alpha$ si

$$P(li(X) \leq \theta \leq ls(X)) = 1 - \alpha.$$

Obsérvese que aquí no hay una distribución de probabilidad que refleje la incertidumbre sobre el parámetro θ , sino sobre los límites del intervalo, que son aleatorios porque dependen de la muestra.

5.1.2. Interpretación de un intervalo de confianza

Si repitiéramos el experimento infinitas veces y construyéramos un intervalo a partir de cada muestra, el 95 % de esos intervalos contendría el verdadero valor de θ .

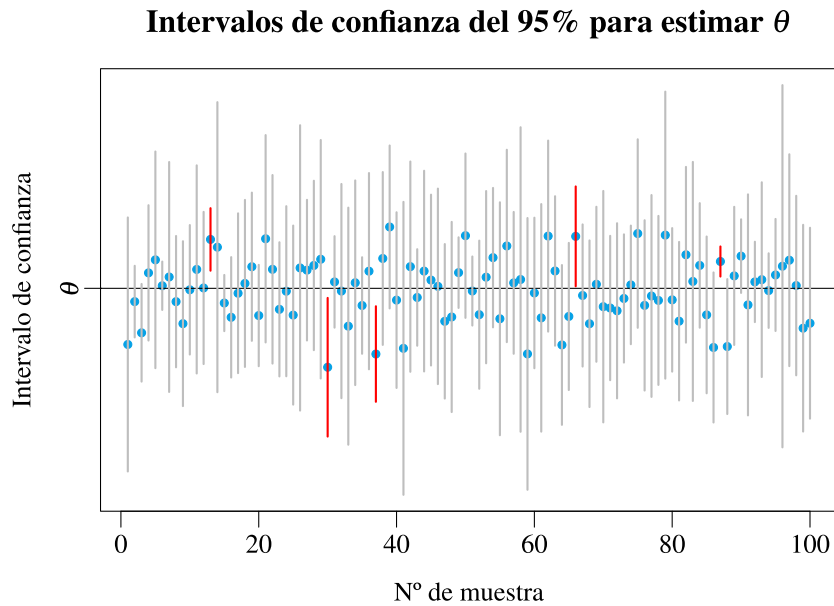


Figura 5.3: Simulación de 100 intervalos de confianza.

 Advertencia

Esta interpretación es contrafactual, es decir, se refiere a experimentos que no se han realizado, y es fácil de malinterpretar. Lo que no podemos afirmar es que el intervalo obtenido contenga el parámetro con probabilidad 0,95.

5.1.3. Estimación bayesiana

En estadística bayesiana, la estimación de un parámetro desconocido es la propia distribución a posteriori del parámetro, que refleja la incertidumbre sobre su valor después de haber observado los datos. Esta distribución se describe habitualmente con distintas medidas:

- **Medidas de tendencia central:** media, mediana o moda.
- **Medidas de posición:** cuantiles.
- **Medidas de dispersión:** varianza, desviación típica o intervalo de credibilidad.

5.1.4. Intervalo de credibilidad

Definición 5.2 (Intervalo de credibilidad). Dada una distribución a posteriori de un parámetro $\theta|x$, un intervalo de credibilidad con nivel $1 - \alpha$ es un intervalo $[li, ls]$ tal que

$$P(\theta \in [li, ls]|x) = 1 - \alpha.$$

La interpretación de un intervalo de credibilidad es mucho más natural: dado un intervalo de credibilidad $[li, ls]$ del 95 %, la probabilidad de que el parámetro θ esté dentro de ese intervalo es del 95 %.

Habitualmente, para construir un intervalo de credibilidad de nivel $1 - \alpha$ se toman los percentiles $\alpha/2$ y $1 - \alpha/2$ de la distribución a posteriori.

Ejemplo 5.1. Para estimar la prevalencia de la hipertensión en mayores de 18 años usamos un modelo beta-binomial con prior $\text{Beta}(4, 6)$. Si tomamos una muestra de 200 personas y encontramos 70 con hipertensión, la distribución a posteriori de la prevalencia es $\text{Beta}(4 + 70, 6 + 200 - 70) = \text{Beta}(74, 136)$.

Podemos obtener esta distribución a posteriori mediante simulación con Stan:

```

modelo <- "
data
{
  int<lower=0> n; // tamaño de la muestra
  int<lower=0> k; // número de éxitos
}
parameters
{
  real<lower=0, upper=1> p; // prevalencia de la hipertensión
}
model
{
  p ~ beta(4, 6); // prior
  k ~ binomial(n, p); // verosimilitud
}
"
tension_post <- stan(model_code = modelo, data = list(n = 200, k = 70),
  chains = 4, iter = 5000*2, seed = 123)

summary(tension_post, pars = "p", probs = c(0.05, 0.95))$summary |>
  kable(digits = 3)

```

	mean	se_mean	sd	5 %	95 %	n_eff	Rhat
p	0.352	0	0.033	0.298	0.408	7634.017	1

Podemos obtener también el intervalo de credibilidad con las funciones `spread_draws` y `mean_qi` del paquete `tidybayes`.

i `spread_draws()`

Extrae las muestras de la distribución a posteriori obtenidas con MCMC y las organiza en un formato tidy para su análisis posterior.

Parámetros:

- **post**: objeto que contiene las muestras de la distribución a posteriori obtenidas con MCMC.
- **p**: nombre del parámetro del que queremos extraer las muestras.

i mean_qi()

Calcula la media y el intervalo de credibilidad de una distribución a posteriori a partir de las muestras obtenidas con MCMC.

Parámetros:

- `.data`: data frame con las muestras de la distribución a posteriori.
- `.width`: nivel de credibilidad.

```
tension_post |>
  spread_draws(p) |>
  mean_qi(p, .width = 0.9) |>
  kable(digits = 3)
```

p	.lower	.upper	.width	.point	.interval
0.352	0.298	0.408	0.9	mean	qi

5.1.4.1. Dibujo de los intervalos de credibilidad

Podemos usar la función `mcmc_intervals` del paquete `bayesplot` para dibujar los intervalos de credibilidad de la distribución a posteriori.

i mcmc_intervals()

Dibuja intervalos de credibilidad para los parámetros de una distribución a posteriori a partir de las muestras obtenidas por simulación de Monte Carlo.

Parámetros:

- `post`: objeto que contiene las muestras de la distribución a posteriori.
- `pars`: vector de caracteres con los nombres de los parámetros que se quieren dibujar.
- `prob`: nivel de credibilidad del intervalo interior.
- `prob_outer`: nivel de credibilidad del intervalo exterior (opcional).
- `point_est`: método para calcular el punto de estimación central (media, mediana o moda).

Ejemplo 5.2. Dibujamos los intervalos de credibilidad calculados en el ejemplo anterior para la prevalencia de la hipertensión.

```
mcmc_intervals(tension_post, pars = "p",
  prob = 0.5, prob_outer = 0.9,
  point_est = "mean") +
  labs(x = "Prevalencia de la hipertensión (p)") +
  theme_gray()
```

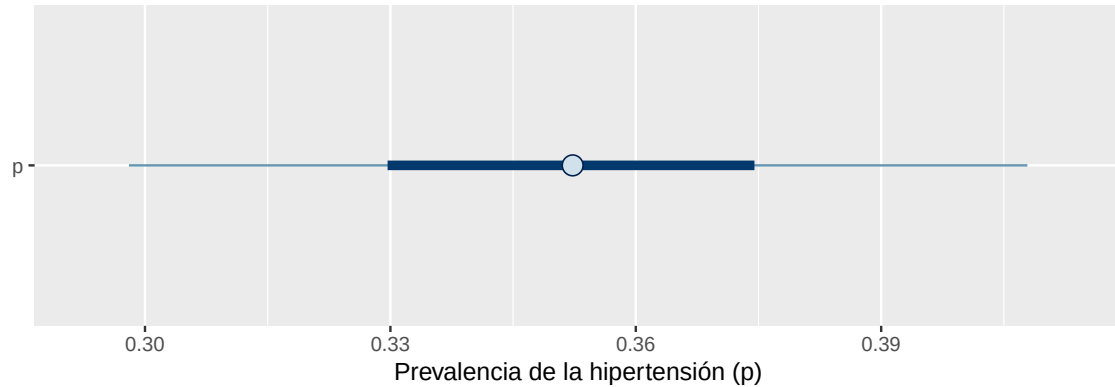


Figura 5.4: Intervalos de credibilidad del 50 % y del 90 % para la prevalencia de la hipertensión.

Ejercicio 5.1. Los datos históricos del sistema de vigilancia epidemiológica sugieren que la incidencia de la tuberculosis en una población está entre 8 y 15 casos por cada 100.000 habitantes y año. Si en el último año se han registrado 48 nuevos casos en una población de 300.000 habitantes, ¿cuál es el intervalo de credibilidad del 95 % para la incidencia de la tuberculosis en esa población?

Solución

Calibrando una distribución gamma para que su rango 95 % sea aproximadamente $[8, 15]$, obtenemos una distribución a priori $\text{Gamma}(27,5; 2,5)$, cuyos percentiles 2,5 % y 97,5 % son 7,3 y 15,5.

Sea λ la incidencia por cada 100.000 habitantes y año, y sea Y el número de casos observados en 3 unidades de exposición (300.000 habitantes), de modo que $Y|\lambda \sim \text{Poisson}(3\lambda)$. Con $Y = 48$, la distribución a posteriori es

$$\lambda|Y = 48 \sim \text{Gamma}(27,5 + 48, 2,5 + 3) = \text{Gamma}(75,5; 5,5).$$

```
modelo <- "
data {
  int<lower=0> y; // número de casos de tuberculosis
  real<lower=0> t; // exposición: población en unidades de 100000
  ↪ habitantes-año
```

```

}
parameters {
  real<lower=0> lambda; // incidencia por 100000 habitantes-año
}
model {
  y ~ poisson(lambda * t); // función de verosimilitud
  lambda ~ gamma(27.5, 2.5); // distribución a priori
}
"
tuberculosis_post <- stan(model_code = modelo,
  data = list(y = 48, t = 3),
  chains = 4, iter = 5000*2, seed = 123)

```

Solución (continuación)

```

tuberculosis_post |>
  spread_draws(lambda) |>
  mean_qi(lambda, .width = 0.95) |>
  kable(digits = 2)

```

lambda	.lower	.upper	.width	.point	.interval
13.75	10.83	17.02	0.95	mean	qi

```

mcmc_intervals(tuberculosis_post, pars = "lambda",
  prob = 0.5, prob_outer = 0.95,
  point_est = "mean") +
  labs(x = "Incidencia de tuberculosis (casos por 100.000
    ⇨ habitantes-año)") +
  theme_gray()

```

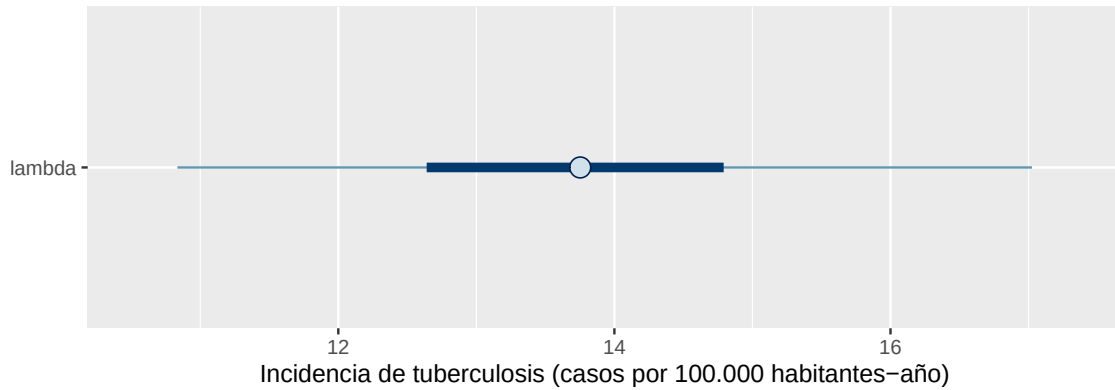


Figura 5.5: Intervalos de credibilidad del 50 % y del 95 % para la incidencia de la tuberculosis.

Con una media a posteriori de unos 13,7 casos por cada 100.000 habitantes y año, el intervalo de credibilidad del 95 % está entre unos 10,8 y 17,0 casos.

5.2. Contrastes de hipótesis

5.2.1. Hipótesis estadística

Definición 5.3 (Hipótesis estadística). Una *hipótesis estadística* es cualquier afirmación o conjetura que determina, total o parcialmente, la distribución de una o varias variables de la población.

Cuando las hipótesis se formulan sobre los parámetros que describen una distribución, se dice que son *hipótesis paramétricas*.

Ejemplo 5.3. Para valorar la gravedad de una epidemia de gripe podríamos estar interesados en contrastar si la incidencia por cada 100.000 personas-día es mayor que 20.

5.2.2. Contrastes de hipótesis paramétricos en estadística frecuentista

En estadística frecuentista, un contraste de hipótesis se realiza eligiendo entre dos hipótesis antagónicas:

- H_0 : hipótesis nula.
- H_1 : hipótesis alternativa.

La hipótesis nula asigna al parámetro un valor concreto, mientras que la alternativa suele ser una hipótesis abierta que, aunque opuesta a la nula, no fija el valor del parámetro. Esto da lugar a tres tipos de contrastes:

Bilateral	Unilateral menor	Unilateral mayor
$H_0: \theta = \theta_0$ $H_1: \theta \neq \theta_0$	$H_0: \theta = \theta_0$ $H_1: \theta < \theta_0$	$H_0: \theta = \theta_0$ $H_1: \theta > \theta_0$

5.2.3. Errores en la toma de decisiones

En un contraste de hipótesis frecuentista se pueden cometer dos tipos de errores:

- **Error de tipo I:** rechazar la hipótesis nula siendo esta verdadera.
- **Error de tipo II:** aceptar la hipótesis nula siendo esta falsa.

Decisión	H_0 cierta	H_1 cierta
Aceptar H_0	Decisión correcta	Error de tipo II
Rechazar H_0	Error de tipo I	Decisión correcta

La probabilidad de cometer un error de tipo I es α , el *nivel de significación* del contraste, que fijamos de antemano. La probabilidad de cometer un error de tipo II es β .

5.2.4. Aceptación o rechazo de la hipótesis nula

La decisión sobre la hipótesis nula se basa en el cálculo de un *estadístico de prueba* a partir de la muestra observada y su comparación con una distribución de referencia bajo la hipótesis nula.

El *p-valor* es la probabilidad de obtener un valor del estadístico de prueba tan extremo como o más extremo que el observado, suponiendo que la hipótesis nula es cierta. Habitualmente, la regla de decisión es

p-valor	Decisión
$p \leq \alpha$	Rechazar H_0
$p > \alpha$	No rechazar H_0

Habitualmente se toma $\alpha = 0,05$. Así, la metodología favorece H_0 , que es la hipótesis conservadora, y solo se rechaza si la evidencia en su contra es suficientemente fuerte.

5.2.5. Interpretación de la decisión

Cuando el p-valor es menor que α , se rechaza la hipótesis nula y se concluye que hay suficiente evidencia para apoyar la hipótesis alternativa.

Un p-valor menor que α no significa que la hipótesis alternativa sea cierta, sino que los datos observados son poco compatibles con la hipótesis nula: si la hipótesis nula fuera cierta y repitiéramos el experimento muchas veces, obtendríamos un resultado tan extremo como el observado en menos de un $\alpha\%$ de las ocasiones. Como ocurre con los intervalos de confianza, esta interpretación es contrafactual y genera muchos errores.

5.2.6. Hipótesis bayesianas

En estadística bayesiana, las hipótesis se pueden formular de manera más flexible, porque no es necesario que la hipótesis nula asigne un valor concreto al parámetro. Normalmente, la hipótesis nula plantea un intervalo de valores para el parámetro y la alternativa contempla el resto:

- **Contraste bilateral:** $H_0 : \theta \in [\theta_1, \theta_2]$ frente a $H_1 : \theta \notin [\theta_1, \theta_2]$.
- **Contraste unilateral mayor:** $H_0 : \theta \leq \theta_0$ frente a $H_1 : \theta > \theta_0$.
- **Contraste unilateral menor:** $H_0 : \theta \geq \theta_0$ frente a $H_1 : \theta < \theta_0$.

5.2.7. Contrastes de hipótesis bayesianos

En estadística bayesiana, un contraste de hipótesis se realiza de forma natural calculando la probabilidad a posteriori de las hipótesis que se quieren contrastar:

$$P(\theta_1 \leq \theta \leq \theta_2 \mid x) = \sum_{\theta \in [\theta_1, \theta_2]} f(\theta \mid x)$$

para parámetros discretos, o

$$P(\theta_1 \leq \theta \leq \theta_2 \mid x) = \int_{\theta_1}^{\theta_2} f(\theta \mid x) d\theta$$

para parámetros continuos. Casi siempre no calcularemos esta integral, sino que la aproximaremos con las muestras de la distribución a posteriori obtenidas por simulación.

Esta metodología no da ventaja a ninguna de las hipótesis: cada una recibe la probabilidad que le corresponde según la distribución a posteriori.

Ejemplo 5.4. En el ejemplo de la prevalencia de la hipertensión podríamos estar interesados en contrastar si la prevalencia es mayor que el 30 %. Para ello hay que calcular

$$P(p > 0,3 \mid x) = \int_{0,3}^1 f(p|x) dp,$$

que puede aproximarse con las muestras de Stan:

```
tension_post |>
  spread_draws(p) |>
  summarise(prob = mean(p > 0.3)) |>
  kable(digits = 3)
```

prob
0.943

Con la distribución Beta(74, 136) la probabilidad es, aproximadamente, 0,94: los datos son muy favorables a que la prevalencia supere el 30 %.

Ejercicio 5.2. En el ejemplo de la incidencia de la tuberculosis, contrasta si la incidencia es mayor que 15 casos por cada 100.000 habitantes y año.

¿Cómo plantearías el contraste para ver si la incidencia es 18?

Solución

Planteamos el contraste como $H_0 : \lambda \leq 15$ frente a $H_1 : \lambda > 15$. Calculamos la probabilidad a posteriori de la hipótesis alternativa:

```
tuberculosis_post |>
  spread_draws(lambda) |>
  summarise(prob = mean(lambda > 15)) |>
  kable(digits = 3)
```

prob
0.213

Esta probabilidad es de aproximadamente 0,21. Como H_0 tiene una probabilidad a posteriori de unos 0,79, los datos son más compatibles con H_0 . Sin embargo, una

probabilidad de 0,21 para H_1 no es despreciable, así que la conclusión debe presentarse con cautela y no como una certeza.

Para contrastar si la incidencia es 18 podemos mirar si ese valor está dentro del intervalo de credibilidad:

```
tuberculosis_post |>
  spread_draws(lambda) |>
  mean_qi(lambda, .width = 0.99) |>
  kable(digits = 2)
```

lambda	.lower	.upper	.width	.point	.interval
13.75	10.02	18.17	0.99	mean	qi

El intervalo de credibilidad del 99% va de unos 10,0 a 18,2 casos por cada 100.000 habitantes y año. El valor 18 queda dentro del intervalo, aunque cerca de su límite superior, por lo que los datos no permiten descartarlo con ese nivel de credibilidad.

5.2.8. Odds a posteriori

En los contrastes bayesianos resulta habitual comparar las probabilidades a posteriori de las hipótesis mediante las *odds a posteriori*:

$$\text{Odds}(H_0|x) = \frac{P(H_0|x)}{P(H_1|x)}.$$

Si las odds a posteriori son mayores que 1, la hipótesis nula es más probable que la alternativa; si son menores que 1, la alternativa es más probable que la nula.

Ejemplo 5.5. En el ejemplo de la prevalencia de la hipertensión, las odds a posteriori de que la prevalencia sea mayor que el 30% se calculan con

```
tension_post |>
  spread_draws(p) |>
  summarise(odds = mean(p > 0.3) / mean(p <= 0.3)) |>
  kable(digits = 1)
```

odds
16.7

Es decir, la hipótesis de que la prevalencia supera el 30% es casi 17 veces más probable que la alternativa.

5.2.9. Factor de Bayes

En ocasiones interesa comparar las odds de la hipótesis nula según la distribución a posteriori con las odds según la distribución a priori. Para ello se utiliza el *factor de Bayes*.

Definición 5.4 (Factor de Bayes). El factor de Bayes es la razón entre las odds a posteriori y las odds a priori de una hipótesis H_0 frente a una hipótesis H_1 :

$$\text{BF} = \frac{\text{Odds}(H_0|x)}{\text{Odds}(H_0)}.$$

Esta medida cuantifica la evidencia que aportan los datos a favor de una hipótesis frente a otra, teniendo en cuenta la información previa que se tenía sobre el parámetro.

- Si $\text{BF} = 1$, la plausibilidad de H_0 no ha cambiado tras observar los datos.
- Si $\text{BF} < 1$, la plausibilidad de H_0 ha disminuido tras observar los datos, es decir, los datos aportan más evidencia a favor de H_1 que de H_0 .
- Si $\text{BF} > 1$, la plausibilidad de H_0 ha aumentado tras observar los datos, es decir, los datos aportan más evidencia a favor de H_0 que de H_1 .

Ejemplo 5.6. En el ejemplo de la prevalencia de la hipertensión, el factor de Bayes a favor de que la prevalencia sea mayor que el 30 % se calcula con

```
tension_post |>
  spread_draws(p) |>
  summarise(odds_prior = (1 - pbeta(0.3, 4, 6)) / pbeta(0.3, 4, 6),
            odds_post = mean(p > 0.3) / mean(p <= 0.3),
            BF = odds_post / odds_prior) |>
  kable(digits = 1)
```

odds_prior	odds_post	BF
2.7	16.7	6.2

En este caso, el factor de Bayes es de aproximadamente 6,2. Es decir, los datos multiplican por 6,2 las odds a favor de que la prevalencia supere el 30 %, lo que indica que aportan bastante más evidencia a favor de esa hipótesis que a favor de la alternativa. Como hay que tener en cuenta que la prior ya daba una probabilidad alta a ese suceso (0,73), este factor debe leerse junto con las odds a posteriori.

Ejercicio 5.3. En el ejemplo de la incidencia de la tuberculosis, calcula el factor de Bayes a favor de que la incidencia sea mayor que 15 casos por cada 100.000 habitantes y año, es decir, de $H_1 : \lambda > 15$ frente a $H_0 : \lambda \leq 15$.

Solución

```
tuberculosis_post |>
  spread_draws(lambda) |>
  summarise(odds_prior = (1 - pgamma(15, 27.5, 2.5)) / pgamma(15, 27.5,
    ↪ 2.5),
            odds_post = mean(lambda > 15) / mean(lambda <= 15),
            BF = odds_post / odds_prior) |>
  kable(digits = 2)
```

odds_prior	odds_post	BF
0.04	0.27	6.89

La probabilidad a priori de que la incidencia supere 15 era de apenas 0,04, mientras que a posteriori es de unos 0,21. Por eso, el factor de Bayes a favor de H_1 es de unos 6,9: los datos multiplican casi por 7 las odds de la alternativa. Aun así, las odds a posteriori siguen siendo de unos 0,27, es decir, H_0 sigue siendo más probable que H_1 .

5.3. Predicciones

5.3.1. Predicciones en estadística frecuentista

En estadística frecuentista, las predicciones suelen ser puntuales: se obtiene un único valor como predicción para una nueva observación de la variable de interés. Este valor suele ser una medida de tendencia central de la distribución de la variable, determinada por los parámetros estimados a partir de la muestra.

- **Media:** valor esperado de la variable. Es la medida de tendencia central más informativa, pero también la más sensible a los datos atípicos.
- **Mediana:** valor que deja el mismo número de observaciones por debajo y por encima de él. Es más robusta que la media porque no se ve afectada por los datos atípicos.
- **Moda:** valor más frecuente. Puede ser útil para variables categóricas o discretas, pero es poco informativa para variables continuas.

5.3.2. Predicciones en estadística bayesiana

Para realizar predicciones en estadística bayesiana hay que tener en cuenta no solo la incertidumbre sobre la variable que queremos predecir, sino también la incertidumbre sobre los parámetros que describen su distribución. En general, hay dos fuentes de variabilidad:

- **Variabilidad del muestreo:** la variabilidad que se produce al tomar una muestra aleatoria de la población. Por ejemplo, el número de personas con hipertensión en una muestra de 10 personas sigue una distribución binomial $B(10, p)$ cuando conocemos p .
- **Variabilidad de los parámetros:** la variabilidad que se produce al estimar los parámetros de la distribución. Por ejemplo, p no se conoce con exactitud y se describe mediante su distribución a posteriori.

5.3.3. Distribución predictiva posterior

Ambas fuentes de variabilidad deben combinarse. Si $\tilde{X}$ representa la variable a predecir, la probabilidad de obtener una predicción $\tilde{X} = \tilde{x}$ es la verosimilitud $f(\tilde{x}|\theta)$ ponderada por la probabilidad de cada valor de θ según la distribución a posteriori $f(\theta|x)$.

Definición 5.5 (Distribución predictiva posterior). Dada una variable aleatoria $\tilde{X}$ cuya distribución depende de un parámetro θ , y la distribución a posteriori $f(\theta|x)$ del parámetro una vez observados los datos x , la *distribución predictiva posterior* de $\tilde{X}$ es

$$P(\tilde{X} = \tilde{x}|x) = \int_{\Theta} f(\tilde{x}|\theta)f(\theta|x) d\theta,$$

donde $f(\tilde{x}|\theta)$ es la función de verosimilitud de la nueva observación $\tilde{x}$ dado el parámetro θ .

Esta distribución nos da una predicción completa, que refleja todos los valores posibles de la nueva observación y su probabilidad.

5.3.4. Simulación de la distribución predictiva posterior

Para simular la distribución predictiva posterior basta con simular una muestra de valores de θ a partir de la distribución a posteriori $f(\theta|x)$ y, para cada uno de esos valores, simular un valor de $\tilde{X}$ a partir de la verosimilitud $f(\tilde{x}|\theta)$.

Ejemplo 5.7. En el ejemplo de la prevalencia de la hipertensión, simulamos la distribución predictiva posterior del número de personas con hipertensión en una nueva muestra de 10 personas:

```
tension_pred <- tension_post |>
  spread_draws(p) |> # Extraemos las muestras de la distribución a
    ↪ posteriori de p
  # Generamos una predicción para cada valor de p obtenido en la muestra
    ↪ (20000 en total)
  mutate(pred = rbinom(n(), size = 10, prob = p)) |>
  select(p, pred)
head(tension_pred, 5) |> # Mostramos las 5 primeras predicciones
  kable(digits = 3)
```

p	pred
0.309	3
0.375	6
0.315	2
0.420	4
0.330	3

Solo se muestran las 5 primeras de las 20000 predicciones obtenidas por simulación.

Podemos dibujar un diagrama de barras con la distribución predictiva posterior:

```
tension_pred |>
  ggplot(aes(x = pred, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  scale_x_continuous(breaks = 0:10) +
  labs(x = "Número de personas con hipertensión en 10", y = "Frecuencia
    ↪ relativa") +
  theme_gray()
```

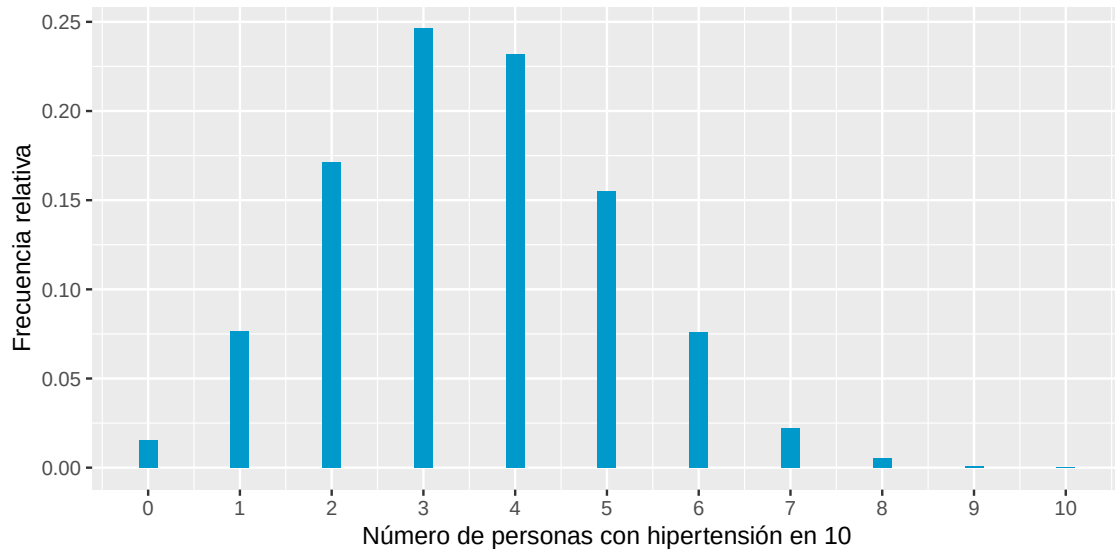


Figura 5.6: Distribución predictiva posterior del número de personas con hipertensión en una nueva muestra de 10 personas.

5.3.5. Resumen de la distribución predictiva posterior

La predicción que obtenemos no es un valor puntual, sino toda una distribución que refleja la incertidumbre sobre los valores posibles. Si necesitamos un valor puntual, podemos calcular cualquier medida de tendencia central sobre esta distribución, aunque es preferible dar un intervalo de credibilidad para la predicción.

Ejemplo 5.8. Continuando con el ejemplo anterior, calculamos la media, la desviación típica y el intervalo de credibilidad del 90 % de la distribución predictiva posterior:

```
tension_pred |>
  summarise(mean = mean(pred),
            sd = sd(pred),
            li = quantile(pred, 0.05),
            ls = quantile(pred, 0.95)) |>
  kable(digits = 2)
```

mean	sd	li	ls
3.52	1.55	1	6

Ejercicio 5.4. Dibuja la distribución predictiva posterior del número de casos de tuberculosis en una nueva muestra de 10.000 personas del ejemplo anterior, y resúmela con la media, la desviación típica y el intervalo de credibilidad del 95 %.

Solución

Una muestra de 10.000 personas equivale a 0,1 unidades de 100.000 habitantes, así que el número de casos se simula con una Poisson de media $0,1\lambda$.

```
tuberculosis_pred <- tuberculosis_post |>
  spread_draws(lambda) |> # Extraemos las muestras de la distribución a
  ↪ posteriori de lambda
  # Generamos una predicción para cada valor de lambda (20000 en total)
  mutate(pred = rpois(n(), lambda = lambda / 10)) |>
  select(lambda, pred)
tuberculosis_pred |>
  ggplot(aes(x = pred, y = after_stat(prop))) +
  stat_count(fill = myblue, width = 0.2) +
  scale_x_continuous(breaks = 0:10) +
  labs(x = "Número de casos de tuberculosis en 10.000 personas", y =
  ↪ "Frecuencia relativa") +
  theme_gray()
```

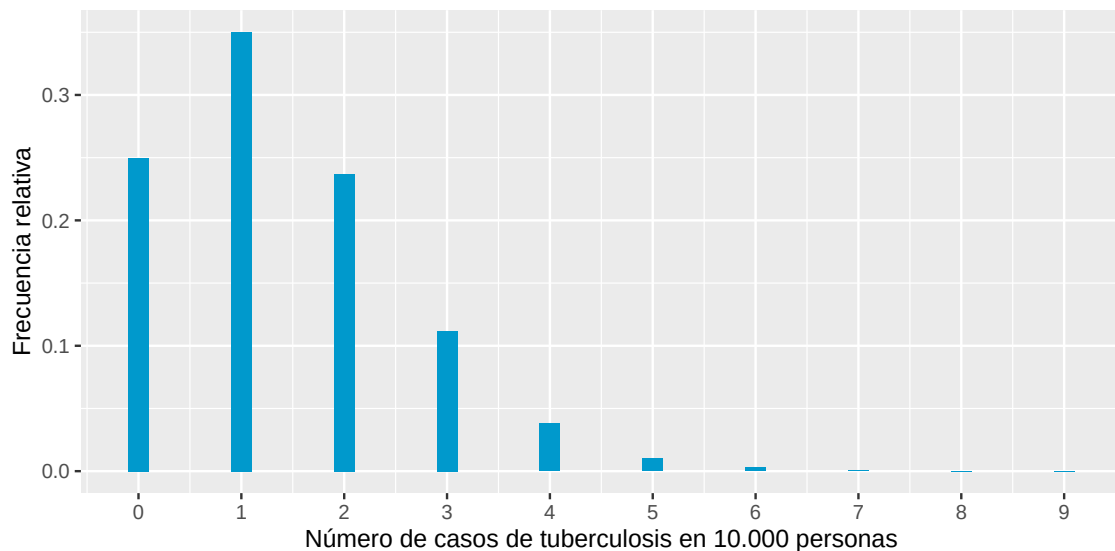


Figura 5.7: Distribución predictiva posterior del número de casos de tuberculosis en una nueva muestra de 10.000 personas.

```
tuberculosis_pred |>
  summarise(mean = mean(pred),
            sd = sd(pred),
            li = quantile(pred, 0.025),
            ls = quantile(pred, 0.975)) |>
```

```
kable(digits = 2)
```

mean	sd	li	ls
1.38	1.17	0	4

5.4. Referencias del capítulo

- Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A., y Rubin, D.B. (2013). *Bayesian Data Analysis*, 3.^a ed. CRC Press. Capítulos 2 y 6.
- Kruschke, J.K. (2015). *Doing Bayesian Data Analysis*, 2.^a ed. Academic Press. Capítulos 11-12.

6 Regresión bayesiana

Los modelos de regresión son el instrumento más utilizado para estudiar cómo se relaciona una variable con otras. En este capítulo se introduce la regresión lineal simple desde el enfoque bayesiano, se extiende a la regresión múltiple y a los modelos con interacciones, y se termina con la evaluación de los modelos, la comparación entre ellos y la predicción.

6.1. Modelos de regresión

En estadística se utilizan *modelos de regresión* para describir la relación entre una variable dependiente Y y una o varias variables independientes $X_1, X_2, \dots, X_n$.

En el caso de una sola variable independiente se habla de *regresión simple*, y el modelo se expresa como

$$Y = f(X) + \varepsilon,$$

donde $f(X)$ es una función de X que describe la relación entre Y y X , y ε es un término de error que refleja la variabilidad de Y que no se explica por X . Se asume que el error sigue una distribución normal con media cero y desviación típica σ :

$$\varepsilon \sim \mathcal{N}(0, \sigma).$$

6.1.1. Diagrama de dispersión

Para analizar la relación entre dos variables cuantitativas hay que estudiar la variabilidad conjunta de ambas. Esta variabilidad conjunta se representa habitualmente con un *diagrama de dispersión*: sobre unos ejes cartesianos, la variable independiente se sitúa en el eje X y la dependiente en el eje Y , y por cada par de valores (x_i, y_j) de la muestra se dibuja un punto.

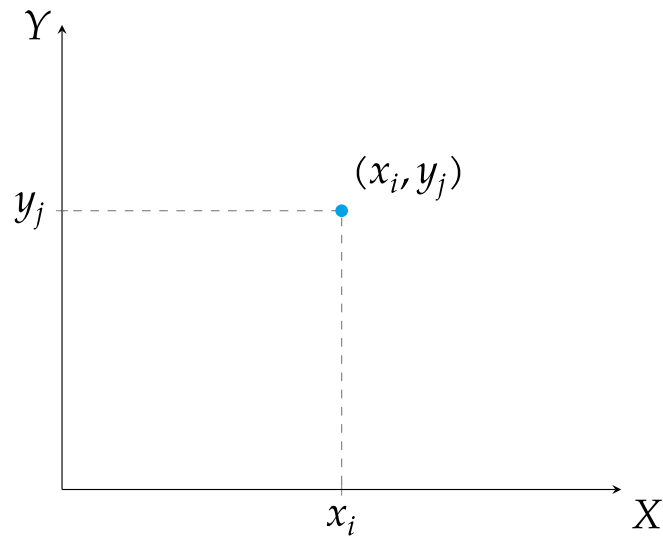


Figura 6.1: Diagrama de dispersión.

Ejemplo 6.1. El conjunto de datos **peso-estatura** contiene el peso y la estatura de una muestra de 200 personas mayores de 18 años. El siguiente código muestra el diagrama de dispersión del peso frente a la estatura.

```
df <- read_csv("datos/peso-estatura.csv", show_col_types = FALSE)
ggplot(df, aes(x = estatura, y = peso)) +
  geom_point(color = myblue) +
  labs(x = "Estatura (cm)", y = "Peso (kg)") +
  theme_minimal()
```

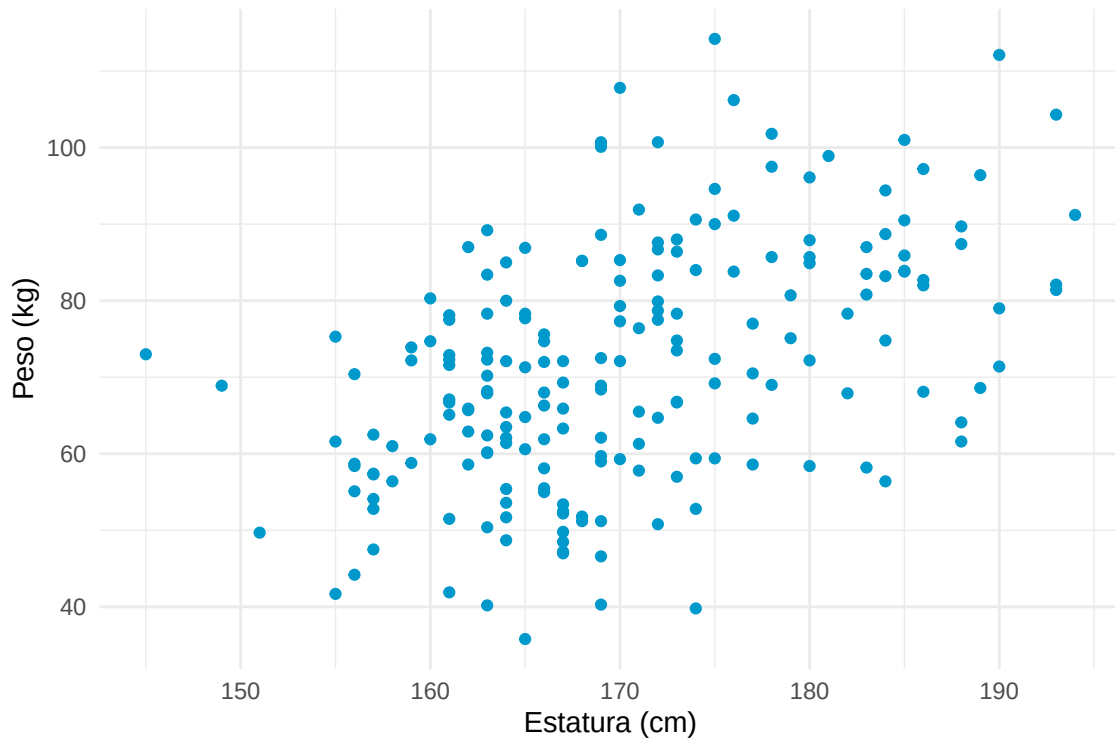


Figura 6.2: Diagrama de dispersión del peso frente a la estatura de 200 personas.

6.1.2. Tipos de modelos de regresión simple

La forma de la nube de puntos nos da una idea de la forma de la función f que describe la relación entre Y y X .

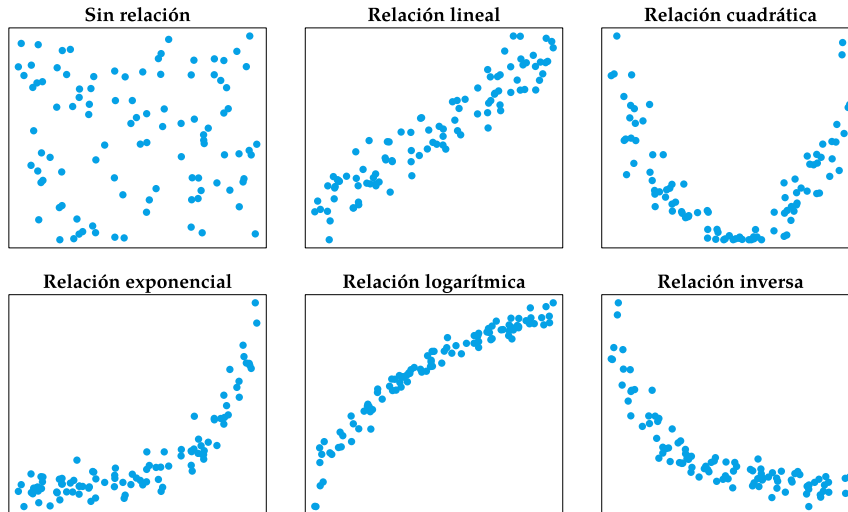


Figura 6.3: Tipos de relación en un diagrama de dispersión.

6.2. Regresión lineal simple

Cuando f es una recta, el modelo es un modelo de *regresión lineal simple*:

$$Y = \beta_0 + \beta_1 X + \varepsilon,$$

donde $f(x) = \beta_0 + \beta_1 x$ es la recta que expresa la relación lineal entre Y y X , y $\varepsilon \sim N(0, \sigma)$.

Para cada valor x_i de la variable independiente, el modelo asume que la variable dependiente Y sigue una distribución normal con media $\mu_i = \beta_0 + \beta_1 x_i$ y desviación típica σ . La relación lineal queda determinada por dos coeficientes:

- **Término independiente** β_0 . Es la media de Y cuando $X = 0$.
- **Pendiente** β_1 . Es el cambio que experimenta la media de Y por cada unidad que aumenta X .

6.2.1. Regresión lineal en estadística frecuentista

En estadística frecuentista, el modelo de regresión lineal simple se estima con el método de los mínimos cuadrados, que consiste en encontrar los valores de β_0 y β_1 que minimizan la suma de los cuadrados de las diferencias entre los valores observados y_i y los valores predichos $\hat{y}_i = \beta_0 + \beta_1 x_i$.

Esto da lugar a la fórmula de la recta de regresión lineal simple:

$$y = \bar{y} + \frac{\sigma_{xy}}{\sigma_x^2}(x - \bar{x}),$$

donde $\bar{x}$ y $\bar{y}$ son las medias de X e Y , σ_{xy} es la covarianza entre X e Y , y σ_x^2 es la varianza de X . Los coeficientes son, por tanto, las estimaciones puntuales

$$\beta_0 = \bar{y} - \beta_1 \bar{x} \quad \text{y} \quad \beta_1 = \frac{\sigma_{xy}}{\sigma_x^2}.$$

6.2.2. Ajuste de modelos de regresión con R

En R existen varias funciones para ajustar modelos de regresión. Las más habituales son:

- `lm()`. Se utiliza para variables dependientes continuas con distribución normal. Ajusta el modelo por mínimos cuadrados ordinarios.
- `glm()`. Se utiliza para variables dependientes con distribuciones de la familia exponencial (normal, binomial, Poisson, gamma, etc.). Ajusta el modelo por máxima verosimilitud.

`lm()`

Ajusta un modelo de regresión lineal por el método de los mínimos cuadrados ordinarios.

Parámetros:

- **formula**: fórmula que especifica el modelo de regresión lineal.
- **data**: conjunto de datos que se utiliza para ajustar el modelo.

La fórmula para especificar un modelo de regresión en R tiene la sintaxis

```
var_dependiente ~ var_independientes
```

donde, en los modelos de regresión múltiple, las variables independientes se combinan con los operadores

- **+** para incluir varias variables independientes sin interacción.
- ***** para incluir varias variables independientes con interacción.
- **:** para incluir solo la interacción entre varias variables independientes.

Ejemplo 6.2. Recordemos el ajuste de un modelo de regresión lineal simple del peso sobre la estatura por mínimos cuadrados con la función `lm()`.

```
modelo_lineal <- lm(peso ~ estatura, data = df)
tidy(modelo_lineal) |> kable(digits = 3)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-59.676	17.113	-3.487	0.001
estatura	0.770	0.100	7.675	0.000

i glm()

Ajusta un modelo de regresión generalizada por el método de máxima verosimilitud. Parámetros:

- **formula**: fórmula que especifica el modelo.
- **family**: familia de distribuciones del error aleatorio y función de enlace.
- **data**: conjunto de datos que se utiliza para ajustar el modelo.

La familia de distribuciones **family** puede ser cualquiera de las siguientes:

family	Variable dependiente	Función de enlace	Uso típico
gaussian()	Continua	identidad μ	Equivalente a <code>lm</code>
binomial()	0/1 o proporción	logit $\log\left(\frac{\mu}{1-\mu}\right)$	Regresión logística
poisson()	Conteos	logaritmo $\log(\mu)$	Modelos de conteo
Gamma()	Continua positiva	inversa $1/\mu$	Tiempos, costes
inverse.gaussian()	Continua positiva	inversa cuadrada $1/\mu^2$	Tiempos de espera

Ejemplo 6.3. El mismo modelo lineal simple puede ajustarse por máxima verosimilitud con `glm()`, usando la familia gaussiana:

```
modelo_lineal <- glm(peso ~ estatura, data = df, family = gaussian())
tidy(modelo_lineal) |> kable(digits = 3)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-59.676	17.113	-3.487	0.001
estatura	0.770	0.100	7.675	0.000

6.3. Regresión lineal en estadística bayesiana

En estadística bayesiana, los dos coeficientes β_0 y β_1 son parámetros más del modelo, a los que se asocia una distribución que refleja la incertidumbre sobre su valor.

Si asumimos que el término de error sigue una distribución normal, el modelo de los datos es

$$Y_i|\beta_0, \beta_1, \sigma \sim N(\beta_0 + \beta_1 x_i, \sigma).$$

Como con otros parámetros, asignamos una distribución a priori a cada coeficiente y a la desviación típica σ del error, y actualizamos estas distribuciones con los datos para obtener las distribuciones a posteriori de todos los parámetros del modelo.

6.3.1. Supuestos del modelo de regresión lineal simple

El modelo de regresión lineal simple se basa en los siguientes supuestos:

1. **Linealidad.** La relación entre Y y X es lineal: la función f es una recta.
2. **Independencia.** Las observaciones son independientes entre sí.
3. **Normalidad.** El término de error ε sigue una distribución normal con media cero y desviación típica σ .
4. **Homocedasticidad.** La varianza del término de error es constante para todos los valores de X .

6.3.2. Distribuciones a priori de los parámetros del modelo

Si disponemos de información previa sobre los parámetros del modelo, debemos utilizar distribuciones informativas que reflejen esa información. En otro caso, podemos usar distribuciones poco informativas.

Como β_0 y β_1 son parámetros de localización, es decir, pueden tomar cualquier valor real, una distribución a priori adecuada para ellos es la normal:

$$\beta_0 \sim N(\theta_0, \tau_0), \quad \beta_1 \sim N(\theta_1, \tau_1).$$

Para la desviación típica σ del término de error, que es un parámetro de escala y solo puede tomar valores positivos, una distribución a priori adecuada es la exponencial (o la gamma inversa):

$$\sigma \sim \text{Exp}(\lambda).$$

6.3.3. Modelo de regresión lineal simple con distribuciones a priori

El modelo completo queda así:

Distribución de los datos: $Y_i | \beta_0, \beta_1, \sigma \sim N(\beta_0 + \beta_1 x_i, \sigma)$

Distribuciones a priori: $\beta_0 \sim N(\theta_0, \tau_0)$, $\beta_1 \sim N(\theta_1, \tau_1)$, $\sigma \sim \text{Exp}(\lambda)$

Ejemplo 6.4. Para modelizar la relación entre el peso y la estatura podemos utilizar un modelo de regresión lineal simple con las siguientes distribuciones a priori:

$$\beta'_0 \sim N(70, 10)$$

$$\beta_1 \sim N(2, 1)$$

$$\sigma \sim \text{Exp}(0,5)$$

Hemos cambiado β_0 , que sería el peso de una persona de estatura 0 cm, algo que no tiene sentido en este caso, por un término independiente centrado, β'_0 , que representa el peso típico de una persona de estatura media.

- β'_0 es el peso de una persona de estatura media. Podemos suponer que está entre 45 y 95 kg (media de 70 kg y desviación típica de 10 kg).
- β_1 es el incremento de peso por cada centímetro de estatura. Podemos suponer que no es negativo y que podría estar entre 0 y 4 kg/cm (media de 2 kg/cm y desviación típica de 1 kg/cm).
- σ mide la variabilidad de los pesos de personas de la misma estatura. No puede ser negativa, y parece razonable suponer desviaciones típicas entre 0 y 10 kg.

6.3.4. Validación de las distribuciones a priori

Cuando utilizamos distribuciones a priori informativas es importante validar que reflejan realmente la información previa que tenemos. Para ello podemos simular valores de los parámetros a partir de las distribuciones a priori y comprobar que son razonables. Con Stan, la opción `algorithm = "Fixed_param"` permite simular a partir de las distribuciones a priori sin ajustar el modelo a los datos.

Ejemplo 6.5. Para validar las distribuciones a priori del ejemplo anterior, simulamos 100 valores de β'_0 , β_1 y σ a partir de sus distribuciones a priori:

```

set.seed(123)
modelo <- "
data {
  int<lower=1> n;    // número de simulaciones (100)
}

generated quantities {
  real beta0 = normal_rng(70, 10);
  real beta1 = normal_rng(2, 1);
  real sigma = exponential_rng(0.5);
}
"
prior <- stan(model_code = modelo, data = list(n = 100),
  chains = 1, iter = 200, algorithm = "Fixed_param")

```

A continuación representamos las rectas de regresión que implican estas simulaciones, para comprobar que son razonables. Cada recta se dibuja con β'_0 como altura en la estatura media de 170 cm:

```

library(tidybayes)
prior |>
  spread_draws(beta0, beta1, sigma) |>
  expand_grid(x = seq(120, 220, length.out = 2)) |>
  mutate(y = beta0 + beta1 * (x - 170)) |>
  ggplot(aes(x = x, y = y, group = .draw)) +
  geom_line(color = myred, alpha = 0.2) +
  labs(x = "Estatura (cm)", y = "Peso (kg)") +
  theme_minimal()

```

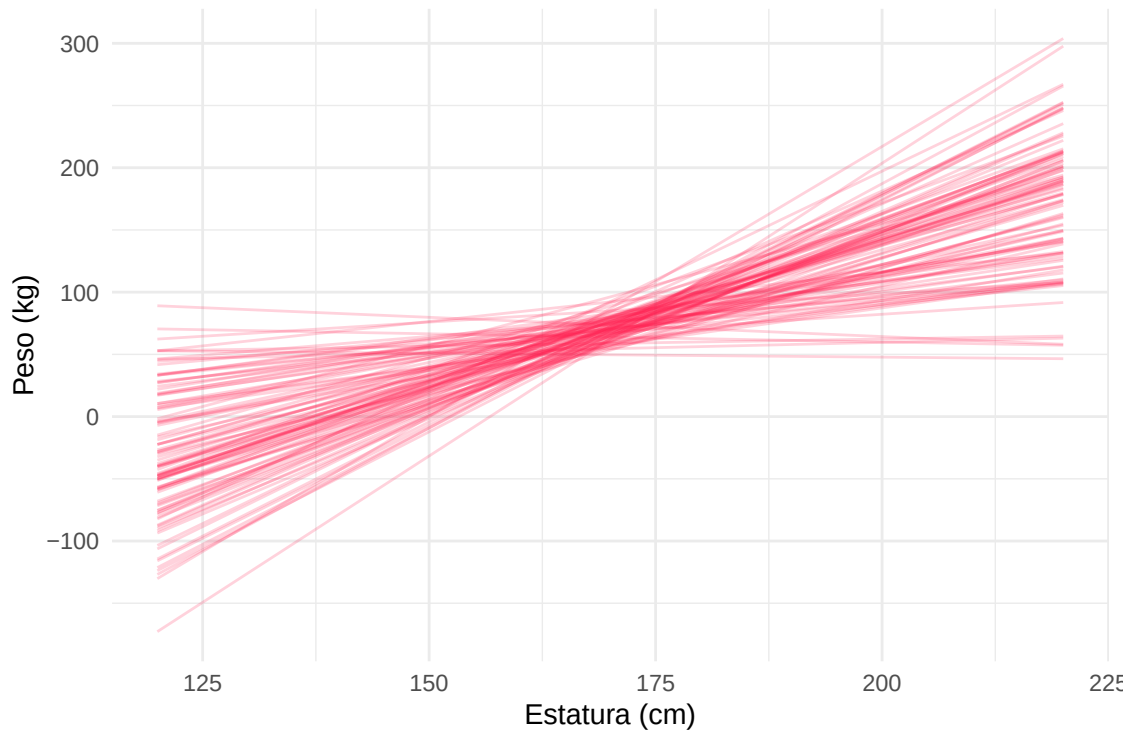


Figura 6.4: Cien rectas de regresión simuladas a partir de las distribuciones a priori.

6.4. Simulación de las distribuciones a posteriori

Una vez validadas las distribuciones a priori, definimos la función de verosimilitud $L(\beta_0, \beta_1, \sigma | X, Y)$ y aplicamos la regla de Bayes para obtener las distribuciones a posteriori de los parámetros:

$$f(\beta_0, \beta_1, \sigma | X, Y) \propto f(\beta_0, \beta_1, \sigma) \cdot L(\beta_0, \beta_1, \sigma | X, Y).$$

Ejemplo 6.6. Veamos cómo ajustar el modelo de regresión lineal simple a los datos de pesos y estaturas. El programa de Stan define los datos, los parámetros, el modelo y las distribuciones a priori:

```
modelo <- "
data {
  int<lower = 0> n;
  vector[n] estatura;
  vector[n] peso;
}
```

```

transformed data {
  // Centramos la estatura para que beta0 sea el peso medio de una
  ↪ persona de estatura media
  vector[n] estatura_c = estatura - mean(estatura);
}
parameters {
  real beta0;
  real beta1;
  real<lower = 0> sigma;
}
model {
  peso ~ normal(beta0 + beta1 * estatura_c, sigma); // verosimilitud
  beta0 ~ normal(70, 10);                          // prior
  beta1 ~ normal(2, 1);                             // prior
  sigma ~ exponential(0.5);                         // prior
}
"
ajuste_peso <- stan(model_code = modelo,
  data = list(n = nrow(df), estatura = df$estatura, peso = df$peso),
  chains = 4, iter = 2000, seed = 123)

```

Podemos comprobar la convergencia de las cadenas de Markov del ajuste anterior con la traza del parámetro de la pendiente:

```

library(bayesplot)
mcmc_trace(ajuste_peso, pars = "beta1", size = 0.2) +
  labs(x = "Iteración", y = expression(beta[1])) +
  theme_gray()

```

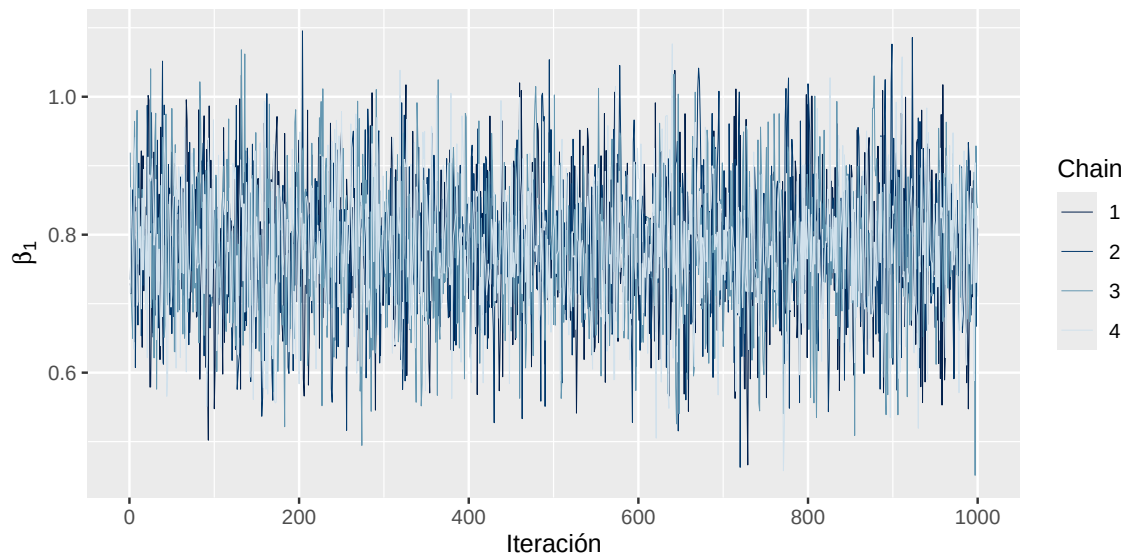


Figura 6.5: Trazas de las cadenas de Markov de la pendiente β_1 del modelo ajustado directamente con Stan.

6.5. Simulaciones de las distribuciones a posteriori con el paquete `rstanarm`

Aunque ya hemos visto cómo ajustar un modelo de regresión directamente con `rstan`, la definición del modelo en el lenguaje de Stan puede resultar compleja, sobre todo en modelos más elaborados. Para facilitar el ajuste de modelos de regresión lineales y generalizados podemos utilizar la función `stan_glm()` del paquete `rstanarm`, que ofrece una interfaz más sencilla con la misma sintaxis que `glm()`.

i `stan_glm()`

Ajusta un modelo de regresión lineal generalizado mediante inferencia bayesiana.
Parámetros:

- **formula**: fórmula que especifica el modelo, con la misma sintaxis que `glm()`.
- **data**: conjunto de datos que se utiliza para ajustar el modelo.
- **family**: familia de distribuciones que se utiliza para modelar la variable dependiente. Para la regresión lineal simple se usa `gaussian()`.
- **prior**: distribución a priori de los coeficientes de regresión (las pendientes).
- **prior_intercept**: distribución a priori del término independiente.
- **prior_aux**: distribución a priori de la desviación típica del término de error.

Ejemplo 6.7. Ajustamos el modelo de regresión lineal simple a los datos de pesos y estaturas con `stan_glm()`, usando las distribuciones a priori que hemos validado:

```
library(rstanarm)
set.seed(123)
modelo_peso_estatura <- stan_glm(peso ~ estatura, data = df, family =
  ↪ gaussian(),
  prior = normal(2, 1),
  prior_intercept = normal(70, 10),
  prior_aux = exponential(0.5),
  iter = 5000*2, chains = 4, seed = 123)
```

⚠ Advertencia

`stan_glm()` centra la variable independiente en su media antes de ajustar el modelo. Por eso el término independiente β_0 representa el peso medio de una persona de estatura media, y no el peso de una persona de estatura 0.

Analizamos las cadenas de Markov generadas por el ajuste:

```
mcmc_trace(modelo_peso_estatura, size = 0.1)
```

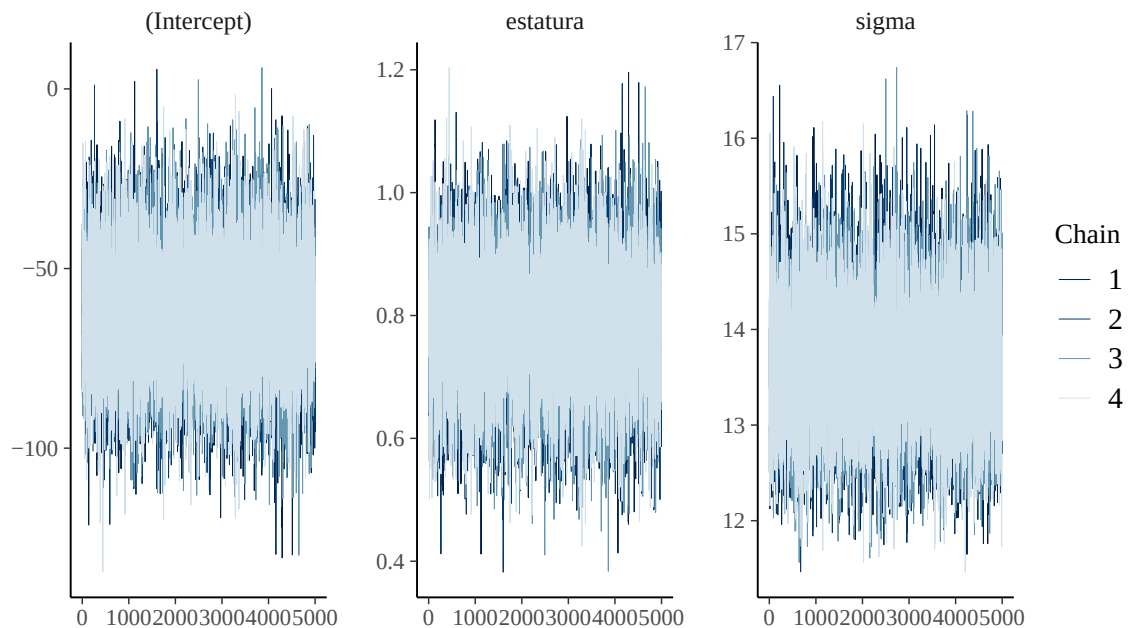


Figura 6.6: Trazas de las cadenas de Markov de los parámetros del modelo ajustado con `stan_glm()`.

```
mcmc_dens_overlay(modelo_peso_estatura)
```

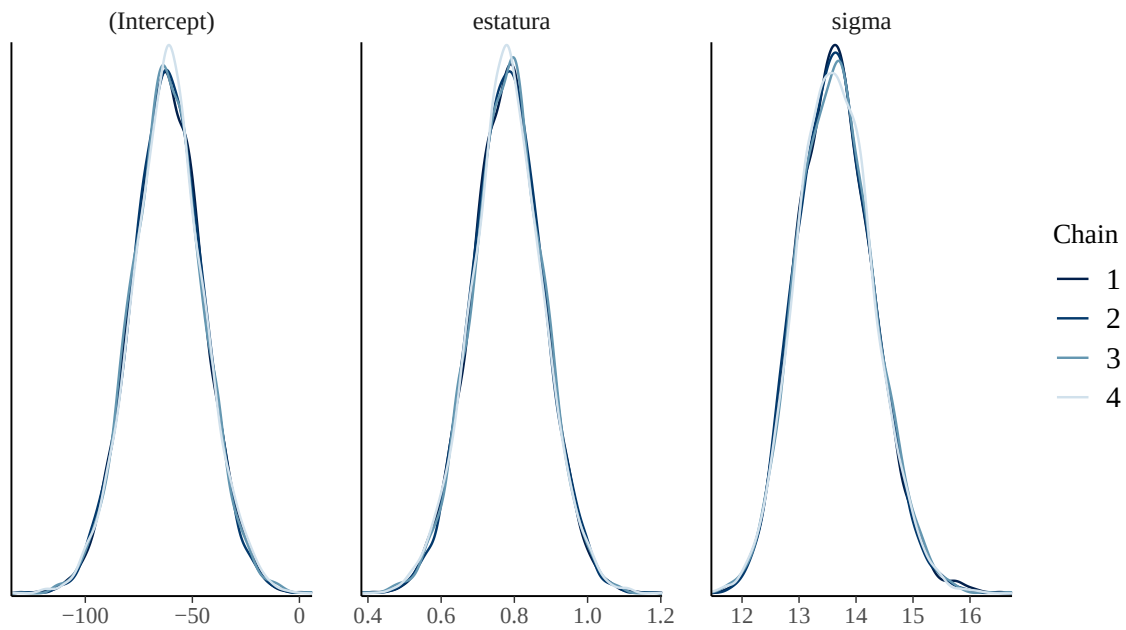


Figura 6.7: Distribuciones a posteriori de los parámetros del modelo, por cadena.

Las cadenas de Markov de los tres parámetros son estables y convergentes. Además, el número efectivo de simulaciones y el valor de $\hat{R}$ confirman la convergencia:

```
neff_ratio(modelo_peso_estatura) |> t() |> kable(digits = 3)
```

(Intercept)	estatura	sigma
0.966	0.966	0.957

```
rhat(modelo_peso_estatura) |> t() |> kable(digits = 3)
```

(Intercept)	estatura	sigma
1	1	1

6.6. Distribuciones a priori por defecto

Cuando no tenemos evidencias fundadas sobre los posibles valores de los parámetros, es preferible usar distribuciones a priori poco informativas. La función `stan_glm()` puede escalar automáticamente las distribuciones a priori a los datos cuando se indica `autoscale = TRUE`.

Ejemplo 6.8. Veamos qué distribuciones a priori se usan en el modelo de regresión lineal simple del peso sobre la estatura cuando las especificamos como poco informativas con escalado automático:

```
modelo_peso_estatura_2 <- stan_glm(peso ~ estatura, data = df, family =  
  ↪ gaussian(),  
  prior_intercept = normal(0, 2.5, autoscale = TRUE),  
  prior = normal(0, 2.5, autoscale = TRUE),  
  prior_aux = exponential(1, autoscale = TRUE),  
  iter = 5000*2, chains = 4, seed = 123)
```

```
prior_summary(modelo_peso_estatura_2)
```

```
Priors for model 'modelo_peso_estatura_2'
```

```
-----
```

```
Intercept (after predictors centered)
```

```
  Specified prior:
```

```
    ~ normal(location = 0, scale = 2.5)
```

```
  Adjusted prior:
```

```
    ~ normal(location = 0, scale = 39)
```

```
Coefficients
```

```
  Specified prior:
```

```
    ~ normal(location = 0, scale = 2.5)
```

```
  Adjusted prior:
```

```
    ~ normal(location = 0, scale = 4)
```

```
Auxiliary (sigma)
```

```
  Specified prior:
```

```
    ~ exponential(rate = 1)
```

```
  Adjusted prior:
```

```
    ~ exponential(rate = 0.064)
```

```
-----
```

```
See help('prior_summary.stanreg') for more details
```

Ejercicio 6.1. Utiliza la función `stan_glm()` para ajustar el modelo de regresión del peso sobre el sexo, usando el conjunto de datos `peso-estatura` y distribuciones a priori poco informativas para los parámetros del modelo. ¿Qué distribuciones a priori propone `stan_glm()`? ¿Qué representan los parámetros del modelo?

 Tip

Cuando la variable independiente es categórica, `stan_glm()` crea automáticamente variables indicadoras (dummy) tomando una categoría como referencia. En el caso del sexo, toma H como referencia y le asigna el valor 0, mientras que a M le asigna el valor 1. El coeficiente `sexoM` es, por tanto, la diferencia de peso medio de las mujeres respecto a los hombres.

Solución

```
modelo_peso_sexo <- stan_glm(peso ~ sexo, data = df, family =  
  ↪ gaussian(),  
  prior_intercept = normal(0, 2.5, autoscale = TRUE),  
  prior = normal(0, 2.5, autoscale = TRUE),  
  prior_aux = exponential(1, autoscale = TRUE),  
  chains = 4, iter = 5000*2, seed = 123)
```

```
prior_summary(modelo_peso_sexo)
```

Priors for model 'modelo_peso_sexo'

Intercept (after predictors centered)

Specified prior:

~ normal(location = 0, scale = 2.5)

Adjusted prior:

~ normal(location = 0, scale = 39)

Coefficients

Specified prior:

~ normal(location = 0, scale = 2.5)

Adjusted prior:

~ normal(location = 0, scale = 79)

Auxiliary (sigma)

Specified prior:

~ exponential(rate = 1)

Adjusted prior:

```

~ exponential(rate = 0.064)
-----
See help('prior_summary.stanreg') for more details

```

El término independiente es el peso medio de los hombres, el coeficiente `sexoM` es la diferencia de peso medio entre mujeres y hombres, y `sigma` es la desviación típica del peso dentro de cada grupo.

6.7. Intervalos de credibilidad de los parámetros

Igual que en el capítulo anterior, podemos describir las distribuciones a posteriori mediante estadísticos de tendencia central e intervalos de credibilidad. Para extraer las muestras de los parámetros generadas por `stan_glm()` podemos usar la función `gather_draws` del paquete `tidybayes`.

i `gather_draws()`

Devuelve un data frame con las muestras de los parámetros en formato *tidy*, es decir, con una fila por cada simulación de cada parámetro.

Parámetros:

- `modelo`: modelo ajustado con `stan_glm()`.
- `...`: nombres de los parámetros que se quieren extraer.

Ejemplo 6.9. Calculamos la media y los intervalos de credibilidad del 90 % de las distribuciones a posteriori de los parámetros del modelo de regresión del peso sobre la estatura:

```

modelo_peso_estatura |>
  gather_draws(`(Intercept)`, estatura, sigma) |>
  mean_qi(.width = 0.9) |>
  kable(digits = 2)

```

.variable	.value	.lower	.upper	.width	.point	.interval
(Intercept)	-61.60	-89.38	-33.79	0.9	mean	qi
estatura	0.78	0.62	0.94	0.9	mean	qi
sigma	13.65	12.59	14.79	0.9	mean	qi

Según estos resultados, por cada centímetro más de estatura se espera un incremento del peso de entre 0,62 y 0,94 kg, con una probabilidad del 90 %.

Ejercicio 6.2. Obtén los intervalos de credibilidad del 90 % de las distribuciones a posteriori de los parámetros del modelo de regresión del peso sobre el sexo. ¿Cómo se interpretan?

Solución

```
modelo_peso_sexo |>
  gather_draws(`(Intercept)`, sexoM, sigma) |>
  mean_qi(.width = 0.9) |>
  kable(digits = 2)
```

.variable	.value	.lower	.upper	.width	.point	.interval
(Intercept)	82.85	80.78	84.93	0.9	mean	qi
sexoM	-20.61	-23.42	-17.83	0.9	mean	qi
sigma	11.92	10.96	12.96	0.9	mean	qi

Con una probabilidad del 90 %, el peso medio de los hombres está en el intervalo que proporciona el término independiente, mientras que el peso medio de las mujeres es menor que el de los hombres en la cantidad indicada por el intervalo de `sexoM`.

6.8. Dibujo de las rectas de regresión a posteriori

Cada muestra de la distribución a posteriori de los parámetros define una recta de regresión distinta, y todas ellas reflejan la incertidumbre sobre el modelo. Podemos dibujar algunas de estas rectas para hacernos una idea visual del ajuste con la función `add_epred_draws` del paquete `tidybayes`.

i `add_epred_draws()`

Añade al conjunto de datos las predicciones de la media de un modelo de regresión bayesiano ajustado con `stan_glm()`, una por cada simulación de la distribución a posteriori.

Parámetros:

- **newdata**: conjunto de datos en el que se hacen las predicciones (por defecto, los datos del modelo).
- **modelo**: modelo ajustado con `stan_glm()`.
- **ndraws**: número de simulaciones de la distribución a posteriori que se quieren dibujar.
- **value**: nombre de la columna donde se guardarán las predicciones.

Ejemplo 6.10. Dibujamos 50 de las rectas de regresión obtenidas a partir de la simulación de las distribuciones a posteriori de los parámetros:

```
df |>
  add_epred_draws(modelo_peso_estatura, ndraws = 50, value =
    ↪ "Predicción_media") |>
  ggplot(aes(x = estatura, y = peso)) +
  geom_point(data = df, color = myblue) +
  geom_line(aes(y = Predicción_media, group = .draw), color = myred,
    ↪ alpha = 0.2) +
  labs(x = "Estatura (cm)", y = "Peso (kg)") +
  theme_minimal()
```

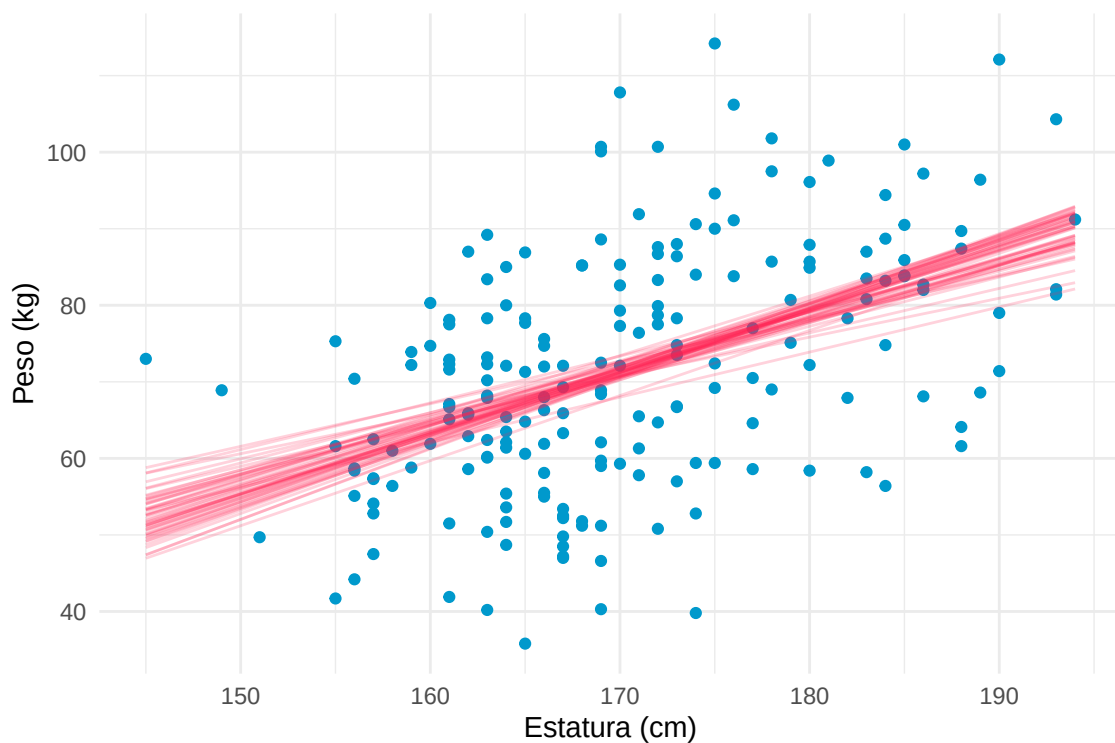


Figura 6.8: Cincuenta rectas de regresión de la distribución a posteriori sobre el diagrama de dispersión de peso y estatura.

6.9. Contraste de hipótesis sobre los coeficientes de regresión

En estadística frecuentista, después de ajustar un modelo de regresión lineal se suele contrastar si cada coeficiente es distinto de cero. Para la pendiente:

$$H_0 : \beta_1 = 0 \quad \text{frente a} \quad H_1 : \beta_1 \neq 0.$$

Este contraste se realiza con el test t de Student, y proporciona un p -valor que indica la probabilidad de obtener un valor tan extremo como el observado si la hipótesis nula fuera cierta.

En estadística bayesiana, en cambio, el contraste se realiza comparando las distribuciones a posteriori de los parámetros con un valor de interés, calculando la probabilidad a posteriori de cada hipótesis.

Ejemplo 6.11. Calculamos la probabilidad a posteriori de que la pendiente del modelo de regresión lineal del peso sobre la estatura sea positiva:

$$H_0 : \beta_1 \leq 0 \quad \text{frente a} \quad H_1 : \beta_1 > 0.$$

```
modelo_peso_estatura |>
  spread_draws(estatura) |>
  summarise(prob = mean(estatura > 0)) |>
  kable(col.names = "P( 1 > 0 | datos)", digits = 3)
```

$$\frac{P(1 > 0 \mid \text{datos})}{1}$$

Es decir, podemos afirmar, con total certeza según el modelo, que la pendiente es positiva: a mayor estatura, mayor peso.

Ejercicio 6.3. Realiza un contraste bayesiano para ver si el peso medio de los hombres es mayor que el de las mujeres, y calcula la probabilidad de que esto sea cierto a partir de las distribuciones a posteriori de los parámetros del modelo.

Solución

En el modelo del sexo, `sexoM` mide la diferencia entre el peso medio de las mujeres y el de los hombres, de modo que la hipótesis de interés es $H_1 : \mu_H > \mu_M$, que equivale a $\beta_{\text{sexoM}} < 0$. Su probabilidad a posteriori es la proporción de muestras de la distribución a posteriori en que `sexoM` es negativo:

```
modelo_peso_sexo |>
  spread_draws(sexoM) |>
  summarise(prob_H_mayor_M = mean(sexoM < 0)) |>
  kable(digits = 3)
```

prob_H_mayor_M
1

6.10. Predicciones con modelos de regresión

Una de las principales utilidades de los modelos de regresión es predecir nuevos casos a partir de los valores de las variables independientes.

Al hacer predicciones con un modelo de regresión hay que tener en cuenta dos fuentes de variabilidad:

- **Variabilidad de los parámetros del modelo**, que se refleja en las distribuciones a posteriori de los parámetros obtenidas por simulación.
- **Variabilidad en el muestreo**, asociada al término de error, que viene dada por el parámetro σ y que también tiene una distribución a posteriori.

Dado un nuevo caso con un valor concreto de la variable independiente $\hat{x}$, podemos estar interesados en dos tipos de predicciones:

- **Predicción de la media** de la variable dependiente para ese caso, $E[Y|\hat{x}]$. Solo tiene en cuenta la variabilidad de los parámetros del modelo.
- **Predicción de un nuevo valor** de la variable dependiente para ese caso, $Y|\hat{x}$. Tiene en cuenta tanto la variabilidad de los parámetros como la variabilidad en el muestreo.

6.10.1. Predicción de la media de la variable dependiente

Para predecir la media de la variable dependiente para un nuevo caso basta con aplicar

$$E[Y|\hat{x}] = \beta_0 + \beta_1 \hat{x}$$

a cada una de las simulaciones de los parámetros del modelo. Para ello podemos usar la función `epred_draws` del paquete `tidybayes`.

i `epred_draws()`

Genera una simulación de la distribución a posteriori de la predicción de la media de la variable dependiente para nuevos casos, a partir de un modelo de regresión bayesiano ajustado con `stan_glm()`.

Parámetros:

- **modelo**: modelo ajustado con `stan_glm()`.

- **newdata**: nuevos casos para los que se quiere hacer la predicción, en un data frame con las mismas variables independientes que el modelo.
- **value**: nombre de la columna donde se guardarán las predicciones.

Ejemplo 6.12. Obtenemos la distribución a posteriori de la predicción del peso medio para una persona de estatura 180 cm:

```
datos_nuevos <- tibble(estatura = c(180))
pred_media_180 <- epred_draws(modelo_peso_estatura, newdata =
  ↪ datos_nuevos, value = "Predicción_media")
pred_media_180
```

```
# A tibble: 20,000 x 6
# Groups:   estatura, .row [1]
  estatura .row .chain .iteration .draw Predicción_media
    <dbl> <int> <int>      <int> <int>      <dbl>
1      180     1    NA         NA     1        78.1
2      180     1    NA         NA     2        82.4
3      180     1    NA         NA     3        81.6
4      180     1    NA         NA     4        79.0
5      180     1    NA         NA     5        78.5
6      180     1    NA         NA     6        77.0
7      180     1    NA         NA     7        80.1
8      180     1    NA         NA     8        77.2
9      180     1    NA         NA     9        80.6
10     180     1    NA         NA    10        81.1
# i 19,990 more rows
```

Obtendríamos lo mismo calculando directamente la fórmula con las simulaciones de los parámetros:

```
modelo_peso_estatura |>
  spread_draws(`(Intercept)`, estatura) |>
  mutate(Predicción_media = `(Intercept)` + estatura * 180) |>
  head() |>
  kable(digits = 2)
```

.chain	.iteration	.draw	(Intercept)	estatura	Predicción_media
1	1	1	-73.52	0.84	78.14
1	2	2	-83.63	0.92	82.36

.chain	.iteration	.draw	(Intercept)	estatura	Predicción_media
1	3	3	-81.76	0.91	81.56
1	4	4	-37.58	0.65	78.95
1	5	5	-75.57	0.86	78.55
1	6	6	-44.35	0.67	76.95

Describimos la distribución a posteriori de la predicción con su media y su intervalo de credibilidad:

```
mean_qi(pred_media_180, .width = 0.9) |>
  kable(digits = 2)
```

estatura	.row	Predicción_me- dia	.lower	.upper	.width	.point	.interval
180	1	79.03	76.8	81.29	0.9	mean	qi

Es decir, para una persona de 180 cm el peso medio estaría entre unos 77 y 81 kg, con una probabilidad del 90 %.

6.10.2. Predicción de un nuevo valor de la variable dependiente

Para predecir un nuevo valor de la variable dependiente, debemos tener en cuenta tanto la variabilidad de los parámetros como la del muestreo:

$$Y|\hat{x}, \beta_0, \beta_1, \sigma \sim N(\beta_0 + \beta_1 \hat{x}, \sigma).$$

Para obtener una muestra de esta distribución basta con generar un valor normal para cada simulación de los parámetros:

$$\begin{aligned} \beta_{0,1}, \beta_{1,1}, \sigma_1 &\longrightarrow \hat{y}_1 = \text{rnorm}(1, \beta_{0,1} + \beta_{1,1} \hat{x}, \sigma_1) \\ \beta_{0,2}, \beta_{1,2}, \sigma_2 &\longrightarrow \hat{y}_2 = \text{rnorm}(1, \beta_{0,2} + \beta_{1,2} \hat{x}, \sigma_2) \\ &\vdots \\ \beta_{0,n}, \beta_{1,n}, \sigma_n &\longrightarrow \hat{y}_n = \text{rnorm}(1, \beta_{0,n} + \beta_{1,n} \hat{x}, \sigma_n) \end{aligned}$$

Podemos hacerlo con la función `predicted_draws` del paquete `tidybayes`.

i predicted_draws()

Genera una simulación de la distribución a posteriori de la predicción de un nuevo valor de la variable dependiente para nuevos casos, a partir de un modelo de regresión bayesiano ajustado con `stan_glm()`.

Parámetros:

- **modelo**: modelo ajustado con `stan_glm()`.
- **newdata**: nuevos casos para los que se quiere hacer la predicción, en un data frame con las mismas variables independientes que el modelo.
- **value**: nombre de la columna donde se guardarán las predicciones.

Ejemplo 6.13. Obtenemos la distribución a posteriori de la predicción de un nuevo valor del peso para una persona de 180 cm:

```
pred_180 <- predicted_draws(modelo_peso_estatura, newdata =  
  ↪ datos_nuevos, value = "Predicción")  
mean_qi(pred_180, .width = 0.9) |> kable(digits = 1)
```

estatura	.row	Predicción	.lower	.upper	.width	.point	.interval
180	1	79	56.2	101.4	0.9	mean	qi

Es decir, el rango de pesos factibles para una persona de 180 cm está entre unos 56 y 101 kg, con una probabilidad del 90 %. No todos los valores de esta distribución son igualmente probables, por lo que a menudo es más útil representarla gráficamente:

```
ggplot(pred_180, aes(x = Predicción)) +  
  stat_halfeye(.width = 0.9, fill = myblue, alpha = 0.5) +  
  labs(x = "Peso (kg)", y = "Densidad")
```

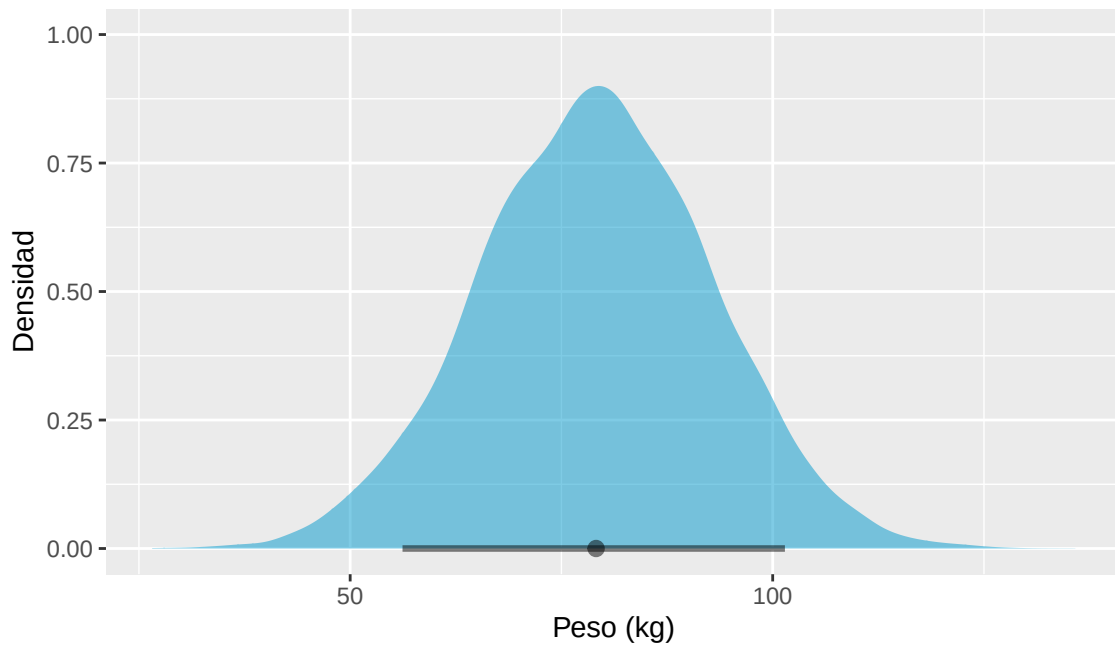


Figura 6.9: Distribución a posteriori de la predicción del peso de una persona de 180 cm.

La siguiente gráfica muestra sobre el diagrama de dispersión el intervalo de credibilidad para la predicción de la media (en verde) y para la predicción de un nuevo valor (en rojo) de una persona de 180 cm:

```
ggplot(df, aes(x = estatura, y = peso)) +
  geom_point(color = myblue) +
  stat_pointinterval(
    data = pred_180,
    aes(x = estatura, y = Predicción),
    .width = 0.9, color = myred, size = 5
  ) +
  stat_pointinterval(
    data = pred_media_180,
    aes(x = estatura, y = Predicción_media),
    .width = 0.9, color = mygreen, size = 5
  ) +
  labs(x = "Estatura (cm)", y = "Peso (kg)") +
  theme_minimal()
```

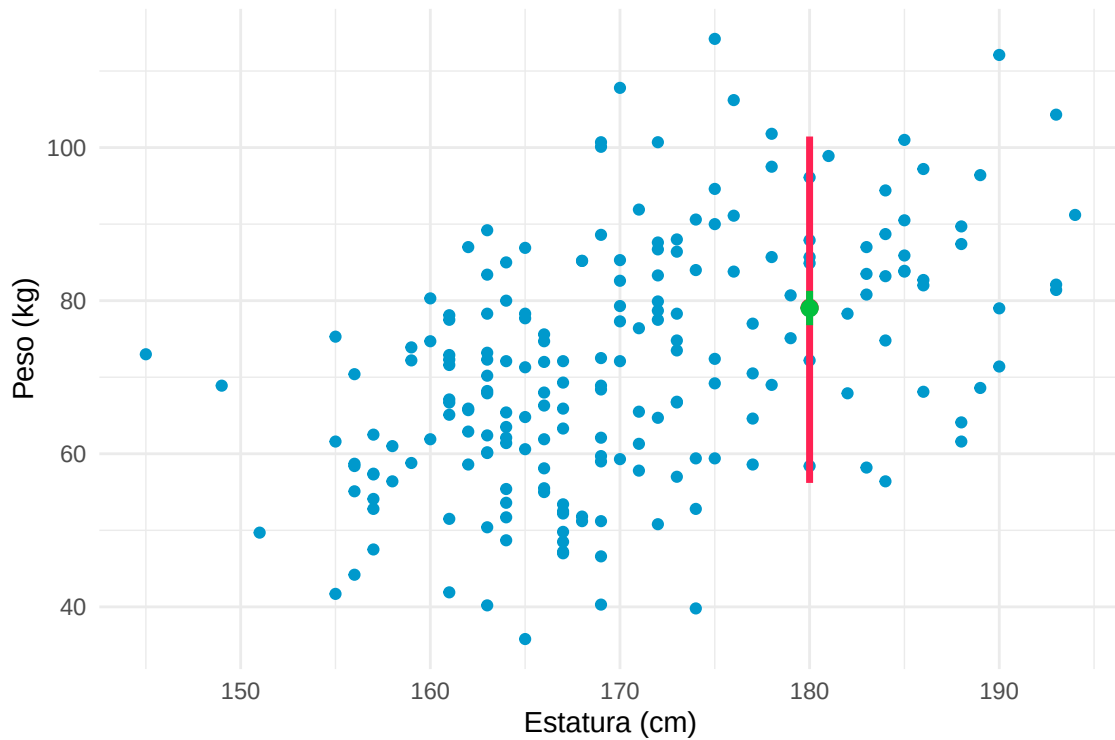


Figura 6.10: Intervalos de credibilidad del 90 % para la predicción de la media (verde) y de un nuevo valor (rojo) para una persona de 180 cm.

Ejercicio 6.4. Calcula la predicción de la media y de un nuevo valor del peso para un hombre y para una mujer, usando el modelo de regresión del peso sobre el sexo. Obtén los intervalos de credibilidad del 90 % de cada predicción y representalos gráficamente sobre el diagrama de dispersión de los datos.

Solución

```
datos_nuevos_sexo <- tibble(sexo = c("H", "M"))
pred_media_sexo <- epred_draws(modelo_peso_sexo, newdata =
  ↪ datos_nuevos_sexo, value = "Predicción_media")
mean_qi(pred_media_sexo, .width = 0.9) |> kable(digits = 1)
```

sexo	.row	Predicción_media	.lower	.upper	.width	.point	.interval
H	1	82.8	80.8	84.9	0.9	mean	qi
M	2	62.2	60.4	64.1	0.9	mean	qi

```

pred_sexo <- predicted_draws(modelo_peso_sexo, newdata =
  ↪ datos_nuevos_sexo, value = "Predicción")
mean_qi(pred_sexo, .width = 0.9) |> kable(digits = 1)

```

sexo	.row	Predicción	.lower	.upper	.width	.point	.interval
H	1	82.8	63.2	102.5	0.9	mean	qi
M	2	62.3	42.6	82.2	0.9	mean	qi

```

ggplot(df, aes(x = sexo, y = peso)) +
  geom_jitter(color = myblue, alpha = 0.4, width = 0.1) +
  stat_pointinterval(
    data = pred_media_sexo,
    aes(x = sexo, y = Predicción_media),
    .width = 0.9, color = mygreen, linewidth = 3, size = 5
  ) +
  stat_pointinterval(
    data = pred_sexo,
    aes(x = sexo, y = Predicción),
    .width = 0.9, color = myred, linewidth = 1, size = 3
  ) +
  labs(x = "Sexo", y = "Peso (kg)") +
  theme_minimal()

```

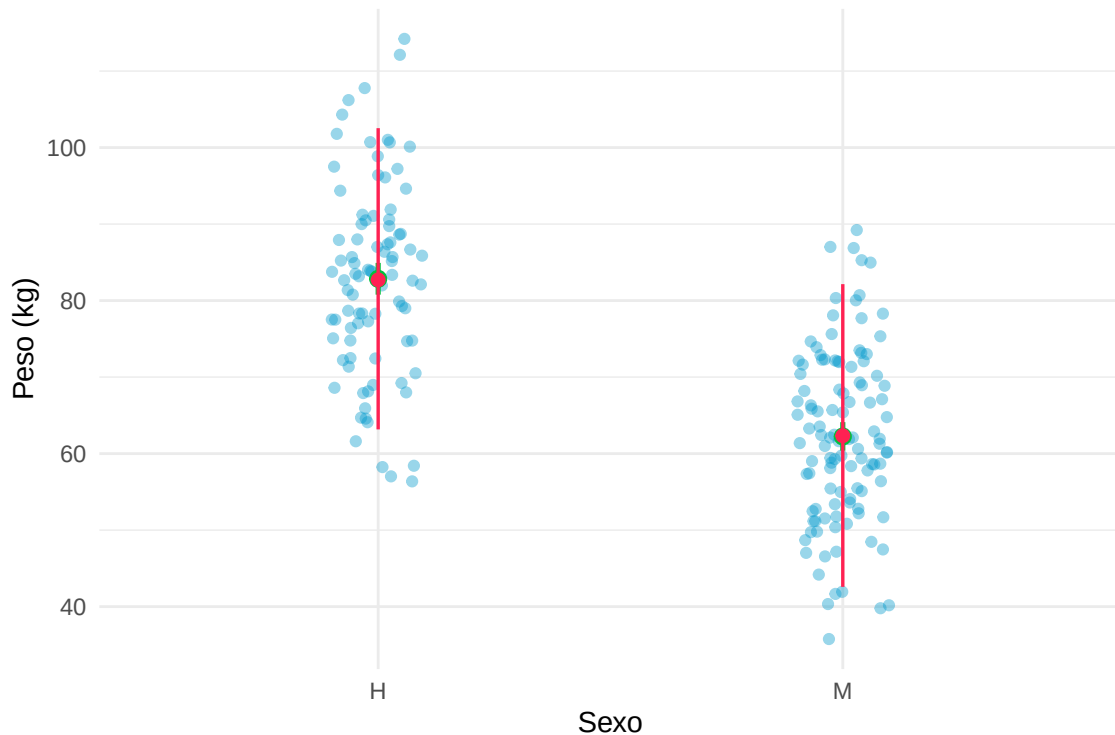


Figura 6.11: Intervalos de credibilidad del 90 % para la predicción de la media (verde) y de un nuevo valor (rojo) del peso de hombres y mujeres.

6.11. Evaluación de modelos de regresión

Todos los modelos de regresión son aproximaciones de la realidad. Por eso es importante evaluar la calidad del ajuste, sobre todo si queremos hacer predicciones con el modelo. La forma más lógica de hacerlo es comparar las predicciones con los valores observados y medir la discrepancia entre ambos.

Para ello, primero obtenemos la distribución a posteriori de la predicción para cada caso de la muestra con `predicted_draws()`:

Ejemplo 6.14. Calculamos los intervalos de credibilidad de las predicciones para cada una de las estaturas del conjunto de datos:

```
post_pred <- predicted_draws(modelo_peso_estatura, newdata = df, value =
  ↪ "Predicción")
ggplot(post_pred, aes(x = estatura, y = Predicción)) +
  stat_pointinterval(.width = c(0.5, 0.95), color = myred) +
  geom_point(data = df, aes(y = peso), color = myblue) +
  labs(x = "Estatura (cm)", y = "Peso (kg)")
```

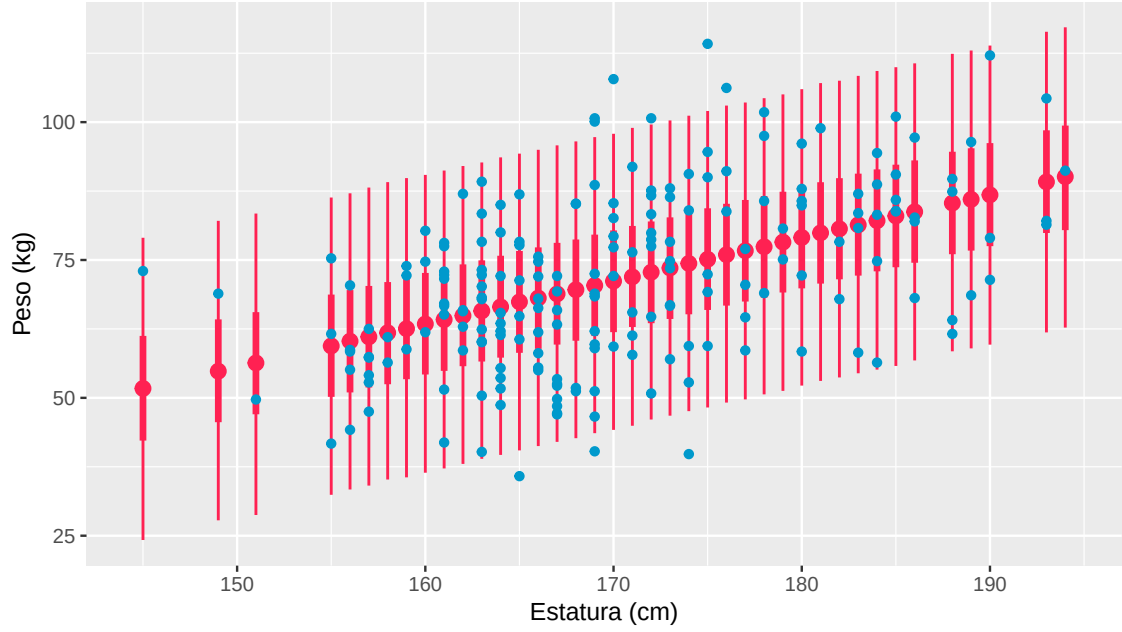


Figura 6.12: Intervalos de credibilidad del 50 % y del 95 % de las predicciones del modelo para cada estatura, con los pesos observados.

6.11.1. Métricas de evaluación

Una vez generadas las distribuciones a posteriori de las predicciones para cada caso de la muestra, podemos usar las siguientes métricas para evaluar la precisión del modelo:

- **Error absoluto mediano (MAE).** Mediana de los valores absolutos de las diferencias entre los valores observados y los predichos:

$$MAE = \text{Mediana}\{|y_i - \hat{y}_i|\}.$$

- **Error absoluto mediano tipificado (MAE_t).** Igual que el anterior, pero dividiendo cada diferencia por la desviación típica de los valores observados:

$$MAE_t = \text{Mediana}\left\{\frac{|y_i - \hat{y}_i|}{\sigma_y}\right\}.$$

- **Raíz del error cuadrático medio (RMSE).** Raíz de la media de los cuadrados de las diferencias entre los valores observados y los predichos:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}.$$

- **Cobertura de los intervalos de credibilidad.** Proporción de casos para los que el valor observado cae dentro del intervalo de credibilidad de la predicción. Se calcula para distintos niveles, por ejemplo, 50 %, 90 % y 95 %.
- **Coefficiente de determinación bayesiano R^2 .** Proporción de la variabilidad total de los valores observados que explica el modelo:

$$R^2_{\text{Bayes}} = \frac{\text{Var}(\hat{y})}{\text{Var}(\hat{y}) + \text{Var}(y - \hat{y})}.$$

Para calcular el error absoluto mediano, el error absoluto mediano tipificado y la cobertura de los intervalos podemos usar la función `prediction_summary()` del paquete `bayesrules`.

i `prediction_summary()`

Devuelve el error absoluto mediano, el error absoluto mediano tipificado y las coberturas de los intervalos de las predicciones de un modelo de regresión bayesiano. Parámetros:

- **model:** modelo de regresión bayesiano ajustado con `stan_glm()`.
- **data:** conjunto de datos con la variable dependiente y las variables independientes.
- **prob_inner:** nivel de credibilidad del intervalo interior. Por defecto 50 %.
- **prob_outer:** nivel de credibilidad del intervalo exterior. Por defecto 90 %.

Ejemplo 6.15. Calculamos las métricas del modelo de regresión lineal simple del peso sobre la estatura:

```
library(bayesrules)
prediction_summary(modelo_peso_estatura, data = df, prob_outer = 0.9) |>
  ↪ kable(digits = 2)
```

mae	mae_scaled	within_50	within_90
9.15	0.66	0.5	0.92

Así, en los 200 individuos de la muestra, el peso observado se encuentra típicamente a unos 9,2 kg de la mediana de la distribución predictiva correspondiente a su estatura, es decir, a unas 0,66 desviaciones típicas. Además, el 50 % de los pesos cae dentro del intervalo de credibilidad del 50 % y el 92 % cae dentro del intervalo del 90 %.

Para calcular el coeficiente de determinación bayesiano R^2 podemos usar la función `bayes_R2()` del paquete `rstanarm`.

i bayes_R2()

Calcula la distribución a posteriori del coeficiente de determinación bayesiano R^2 de un modelo de regresión bayesiano ajustado con `stan_glm()`.

Parámetros:

- `object`: modelo de regresión bayesiano ajustado con `stan_glm()`.

Ejemplo 6.16. Calculamos la distribución a posteriori del coeficiente de determinación del modelo de regresión lineal simple:

```
bayes_R2(modelo_peso_estatura) |>
  mean_qi(.width = 0.9) |>
  kable(digits = 3, col.names = c("R2 medio", ".lower", ".upper",
    ↪ ".width", ".point", ".interval"))
```

R ² medio	.lower	.upper	.width	.point	.interval
0.238	0.159	0.319	0.9	mean	qi

El modelo explica en torno al 24 % de la variabilidad del peso y, con una probabilidad del 90 %, entre el 16 % y el 32 %.

Ejercicio 6.5. Calcula las métricas de evaluación del modelo de regresión del peso sobre el sexo. ¿Qué conclusiones se pueden extraer?

Solución

```
prediction_summary(modelo_peso_sexo, data = df, prob_outer = 0.9) |>
  ↪ kable(digits = 2)
```

mae	mae_scaled	within_50	within_90
8.03	0.67	0.5	0.9

```
bayes_R2(modelo_peso_sexo) |>
  mean_qi(.width = 0.9) |>
  kable(digits = 3, col.names = c("R2 medio", ".lower", ".upper",
    ↪ ".width", ".point", ".interval"))
```

R ² medio	.lower	.upper	.width	.point	.interval
0.426	0.345	0.501	0.9	mean	qi

El error absoluto mediano indica la distancia típica entre el peso observado y la mediana de la predicción para su sexo. El modelo explica en torno al 43 % de la variabilidad del peso, y el resto de la variabilidad se debe a otros factores, como la estatura, que no están incluidos en este modelo.

6.12. Evaluación del modelo de regresión en datos nuevos

En general, un modelo de regresión se ajusta mejor a los datos con los que se ha construido que a otros datos nuevos. Este problema se conoce como *sobreajuste* (*overfitting*).

Cuando hay sobreajuste, las medidas de precisión evaluadas sobre el propio conjunto de datos sobreestiman la capacidad predictiva del modelo. Por eso es importante evaluar la predicción en datos nuevos. Para ello se suele dividir el conjunto de datos en dos subconjuntos:

- **Conjunto de entrenamiento.** Se utiliza para ajustar el modelo y obtener las distribuciones a posteriori de sus parámetros.
- **Conjunto de prueba.** Se utiliza para evaluar la precisión del modelo, y no se usa para ajustarlo.

6.12.1. Validación cruzada

Para obtener una estimación más fiable de la precisión predictiva, es recomendable repetir el proceso con distintos conjuntos de entrenamiento y prueba, y promediar las métricas obtenidas. Esta es la idea de la *validación cruzada*: dividir los datos en k subconjuntos llamados *pliegues* (*folds*) y realizar k iteraciones. En cada una se reserva un pliegue como conjunto de prueba y se ajusta el modelo con los $k - 1$ restantes. Al final se promedian las métricas de las k iteraciones.



Figura 6.13: Validación cruzada con 4 pliegues.

Para evaluar un modelo de regresión mediante validación cruzada podemos usar la función `prediction_summary_cv()` del paquete `bayesrules`.

i `prediction_summary_cv()`

Devuelve el error absoluto mediano, el error absoluto mediano tipificado y las coberturas de los intervalos de las predicciones de un modelo de regresión bayesiano mediante validación cruzada.

Parámetros:

- **model**: modelo de regresión bayesiano ajustado con `stan_glm()`.
- **data**: conjunto de datos.
- **k**: número de pliegues.
- **prob_inner**: nivel de credibilidad del intervalo interior. Por defecto 50 %.
- **prob_outer**: nivel de credibilidad del intervalo exterior. Por defecto 90 %.

Ejemplo 6.17. Evaluamos el modelo de regresión lineal simple del peso sobre la estatura mediante validación cruzada con 10 pliegues:

```
set.seed(123)
eval_cv <- prediction_summary_cv(model = modelo_peso_estatura, data =
  ↪ df, k = 10, prob_outer = 0.9)
eval_cv$cv |> kable(digits = 2)
```

mae	mae_scaled	within_50	within_90
9.75	0.71	0.5	0.9

Comparadas con las métricas obtenidas sobre todo el conjunto de datos, las métricas por validación cruzada son un poco peores, pero más realistas.

Ejercicio 6.6. Evalúa el modelo de regresión del peso sobre el sexo mediante validación cruzada con 10 pliegues. ¿Qué conclusiones se pueden extraer?

Solución

```
set.seed(123)
eval_cv <- prediction_summary_cv(model = modelo_peso_sexo, data = df, k
  ↪ = 10, prob_outer = 0.9)
eval_cv$cv |> kable(digits = 2)
```

mae	mae_scaled	within_50	within_90
8.13	0.68	0.5	0.9

El error absoluto mediano por validación cruzada es ligeramente mayor que el obtenido sobre todo el conjunto de datos, lo que indica un ligero sobreajuste de los datos de entrenamiento.

6.13. Regresión lineal múltiple

Cuando la variable dependiente Y depende de varias variables independientes $X_1, X_2, \dots, X_p$ hablamos de *regresión lineal múltiple*:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + \varepsilon, \quad \varepsilon \sim N(0, \sigma).$$

El procedimiento para ajustar un modelo de regresión lineal múltiple con inferencia bayesiana es el mismo que para la regresión simple, solo que hay más parámetros y hay que especificar la distribución a priori de cada uno.

Ejercicio 6.7. ¿Qué modelo de regresión lineal múltiple ajustarías para modelizar el peso de una persona en función de su estatura y su sexo? ¿Qué distribuciones a priori utilizarías para los parámetros del modelo?

Solución

Como el sexo es una variable categórica, debemos codificarla numéricamente para introducirla en el modelo (por ejemplo, 0 para hombres y 1 para mujeres).

Si llamamos X_1 a la estatura y X_2 al sexo, el modelo sería

$$Y|\beta_0, \beta_1, \beta_2, \sigma \sim N(\beta_0 + \beta_1 X_1 + \beta_2 X_2, \sigma).$$

De esta manera, para cada sexo el modelo proporciona una recta de regresión distinta:

- Recta de regresión de los hombres ($X_2 = 0$): $Y = \beta_0 + \beta_1 X_1 + \varepsilon$.
- Recta de regresión de las mujeres ($X_2 = 1$): $Y = (\beta_0 + \beta_2) + \beta_1 X_1 + \varepsilon$.

Este modelo asume que la pendiente es la misma para hombres y mujeres, pero que el término independiente es distinto.

Para β_0 , β_1 y σ podemos usar las mismas distribuciones a priori que en el modelo simple. Para β_2 podemos usar una normal que refleje la diferencia de peso entre hombres y mujeres, que sabemos por experiencia que es negativa. Por ejemplo, $\beta_2 \sim N(-10, 5)$ refleja que, para una misma estatura, las mujeres pesan unos 10 kg menos, con una desviación típica de 5 kg.

6.13.1. Evaluación de las distribuciones a priori

Como en los modelos simples, antes de ajustar el modelo conviene comprobar que las distribuciones a priori son razonables. Podemos hacerlo con la misma función `stan_glm()`, añadiendo la opción `prior_PD = TRUE`, que simula a partir de las distribuciones a priori sin usar los datos.

Ejemplo 6.18. Simulamos las distribuciones a priori del modelo de regresión lineal múltiple del peso sobre la estatura y el sexo:

```
library(rstanarm)
set.seed(123)
modelo_peso_estatura_sexo_prior <- stan_glm(
  peso ~ estatura + sexo,
  data = df,
  family = gaussian(),
  prior_intercept = normal(70, 10),
  prior = normal(location = c(2, -10), scale = c(1, 5)),
  prior_aux = exponential(0.5),
  chains = 4, iter = 5000 * 2, seed = 123,
  prior_PD = TRUE
)
```

⚠ Advertencia

`stan_glm()` codifica las variables categóricas de forma automática: la primera categoría en orden alfabético se toma como referencia (código 0) y la segunda recibe el código 1.

Dibujamos algunas de las rectas de regresión que se obtienen de las distribuciones a priori para comprobar que son razonables:

```
df |>
  add_epred_draws(modelo_peso_estatura_sexo_prior, ndraws = 50, value =
    ↪ "Predicción_media") |>
  ggplot(aes(x = estatura, y = peso, color = sexo)) +
  geom_line(aes(y = Predicción_media, group = paste(sexo, .draw)), alpha
    ↪ = 0.2) +
  labs(x = "Estatura (cm)", y = "Peso (kg)") +
  theme_minimal()
```

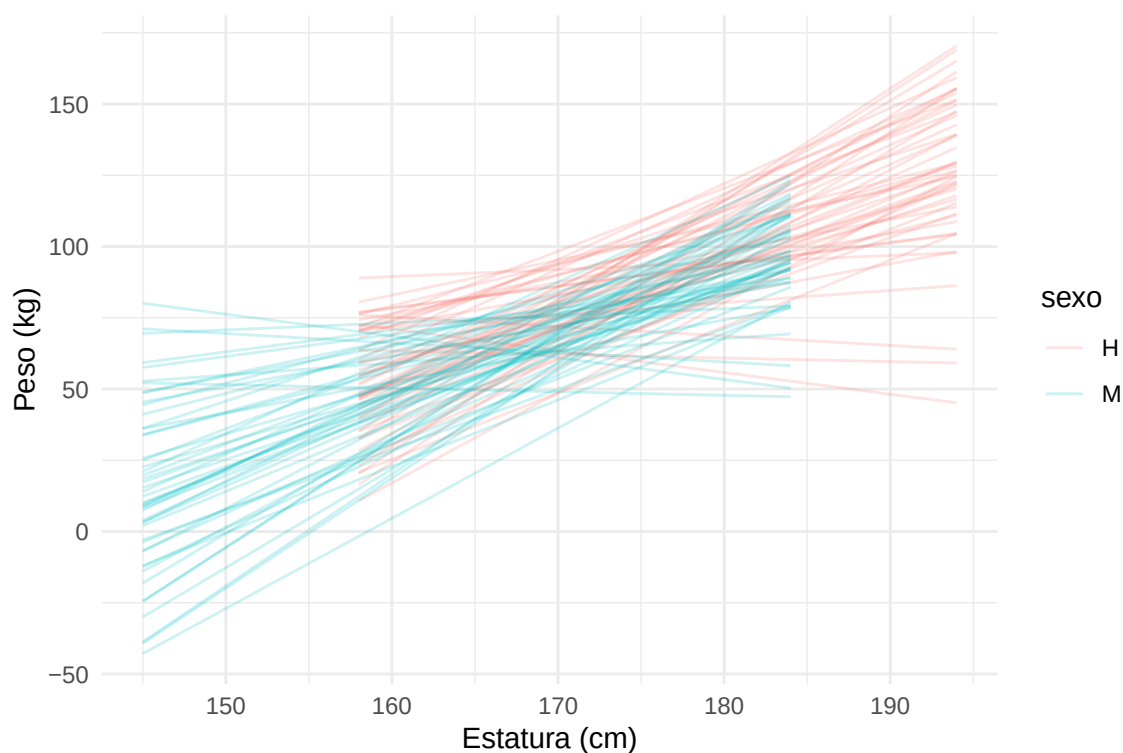


Figura 6.14: Rectas de regresión simuladas a partir de las distribuciones a priori del modelo múltiple, para hombres y mujeres.

6.13.2. Ajuste del modelo de regresión lineal múltiple

Una vez validadas las distribuciones a priori, no hace falta volver a especificarlas. Basta con actualizar el modelo anterior con `update()` y `prior_PD = FALSE`, para que las distribuciones a posteriori se ajusten a partir de los datos:

Ejemplo 6.19.

```
modelo_peso_estatura_sexo <- update(modelo_peso_estatura_sexo_prior,  
  ↪ prior_PD = FALSE)
```

Comprobamos la convergencia de las cadenas de Markov:

```
mcmc_trace(modelo_peso_estatura_sexo, size = 0.2)
```

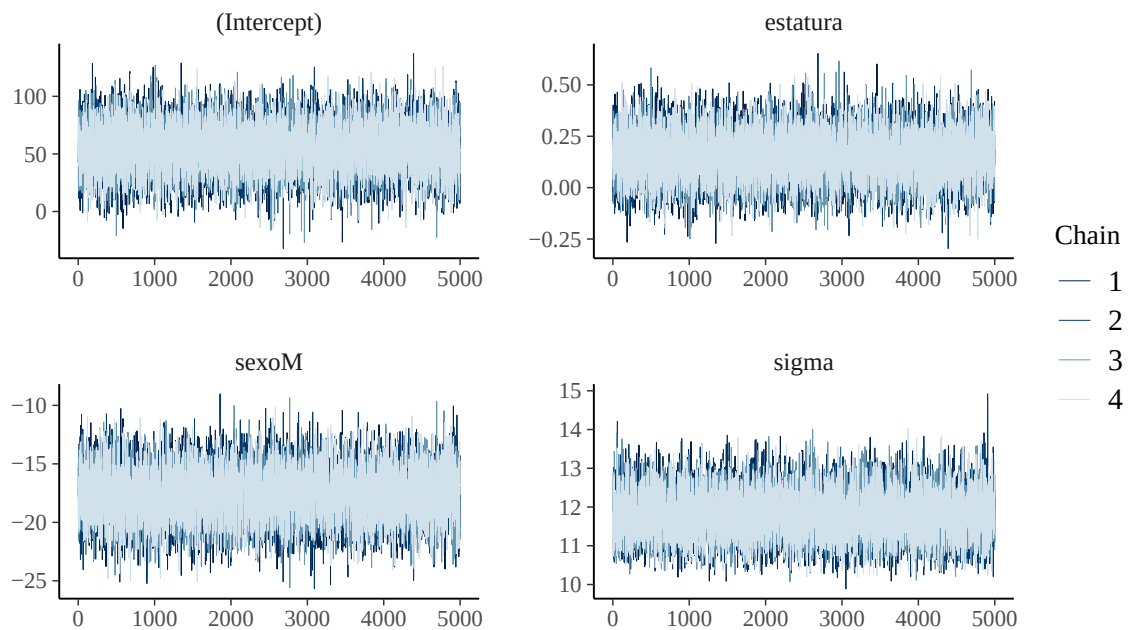


Figura 6.15: Trazas de las cadenas de Markov del modelo de regresión múltiple.

```
mcmc_dens_overlay(modelo_peso_estatura_sexo)
```

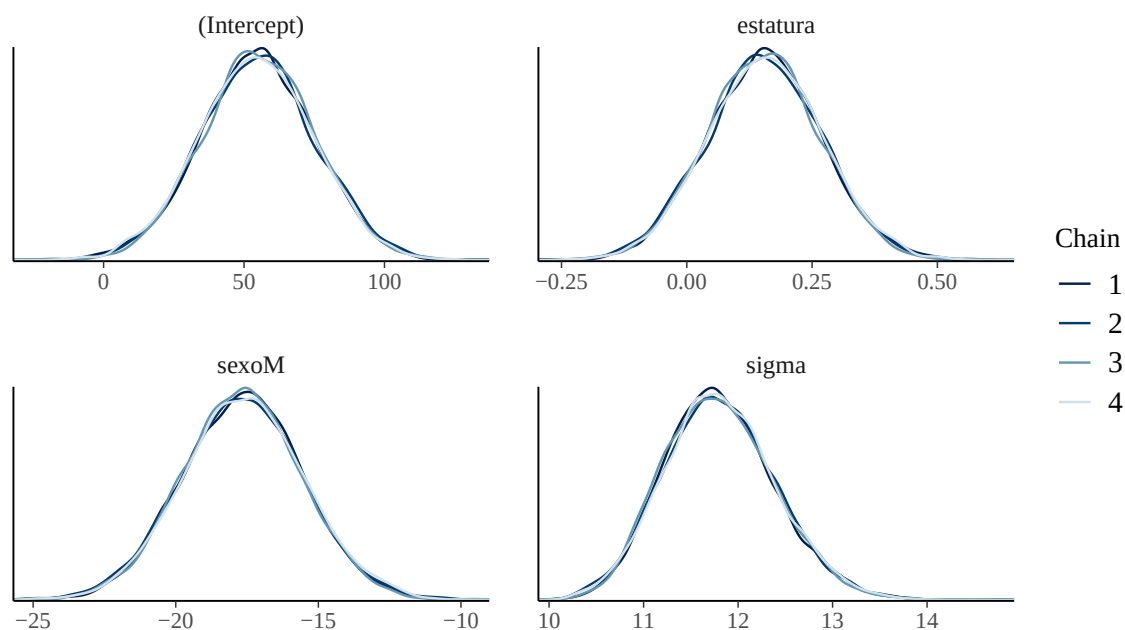


Figura 6.16: Distribuciones a posteriori de los parámetros del modelo de regresión múltiple, por cadena.

```
neff_ratio(modelo_peso_estatura_sexo) |> t() |> kable(digits = 3)
```

(Intercept)	estatura	sexoM	sigma
0.576	0.588	0.597	0.831

```
rhat(modelo_peso_estatura_sexo) |> t() |> kable(digits = 3)
```

(Intercept)	estatura	sexoM	sigma
1	1	1	1

Las cadenas son estables y convergentes: el número efectivo de simulaciones es razonablemente alto para todos los parámetros y el valor de $\hat{R}$ es prácticamente 1.

6.13.3. Intervalos de credibilidad de los parámetros

Ejemplo 6.20. Calculamos la media y los intervalos de credibilidad del 90% de los parámetros del modelo de regresión del peso sobre la estatura y el sexo:

```

modelo_peso_estaturasexo |>
  gather_draws(`(Intercept)`, estatura, sexoM, sigma) |>
  mean_qi(.width = 0.9) |>
  kable(digits = 2)

```

.variable	.value	.lower	.upper	.width	.point	.interval
(Intercept)	54.86	21.48	88.22	0.9	mean	qi
estatura	0.15	-0.03	0.34	0.9	mean	qi
sexoM	-17.65	-21.12	-14.16	0.9	mean	qi
sigma	11.80	10.88	12.81	0.9	mean	qi

Cuando se controla por el sexo, por cada centímetro más de estatura se espera un incremento de peso de entre $-0,03$ y $0,34$ kg, tanto para hombres como para mujeres. Por otro lado, cuando se controla por la estatura, se espera que las mujeres pesen entre $14,2$ y $21,1$ kg menos que los hombres.

6.13.4. Dibujo de las rectas de regresión a posteriori

Ejemplo 6.21. Dibujamos 50 de las rectas de regresión de la distribución a posteriori del modelo de regresión múltiple:

```

df |>
  add_epred_draws(modelo_peso_estaturasexo, ndraws = 50, value =
    ↪ "Predicción_media") |>
  ggplot(aes(x = estatura, y = peso, color = sexo)) +
  geom_point(data = df) +
  geom_line(aes(y = Predicción_media, group = paste(sexo, .draw)), alpha
    ↪ = 0.2) +
  labs(x = "Estatura (cm)", y = "Peso (kg)") +
  theme_minimal()

```

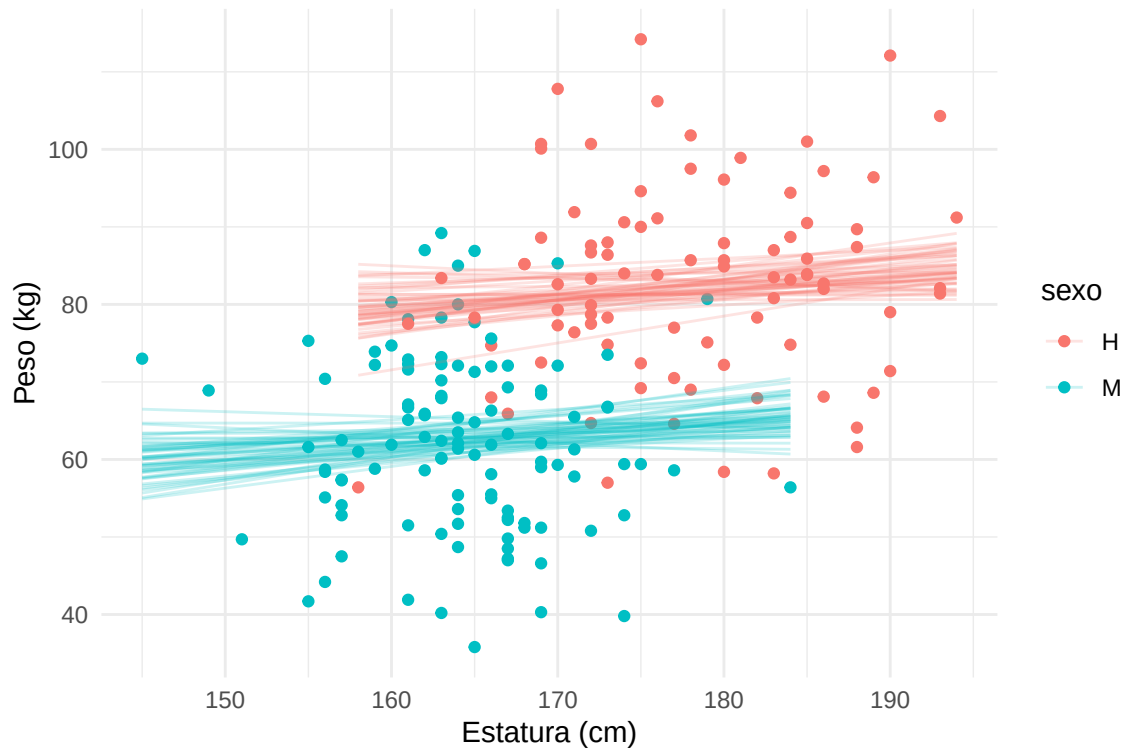


Figura 6.17: Cincuenta rectas de regresión a posteriori del modelo múltiple, para hombres y mujeres, sobre los datos.

Se observa una relación positiva entre la estatura y el peso, de menor magnitud que en el modelo simple, y que, para la misma estatura, las mujeres tienen un peso medio menor que los hombres, algo coherente con la experiencia.

6.13.5. Predicciones del modelo de regresión múltiple

Ejemplo 6.22. Obtenemos la distribución a posteriori de la predicción del peso para un hombre y una mujer de 160 cm:

```
datos_nuevos <- tibble(estatura = c(160, 160), sexo = c("H", "M"))
post_pred_160 <- predicted_draws(modelo_peso_estatura_sexo, newdata =
  ↪ datos_nuevos, value = "Predicción")
mean_qi(post_pred_160, .width = 0.9) |> kable(digits = 1)
```

estatura	sexo	.row	Predicción	.lower	.upper	.width	.point	.interval
160	H	1	79.7	60.0	99.6	0.9	mean	qi

estatura	sexo	.row	Predicción	.lower	.upper	.width	.point	.interval
160	M	2	62.0	42.6	81.4	0.9	mean	qi

Un hombre de 160 cm tendría un peso entre unos 60 y 100 kg con una probabilidad del 90 %, mientras que una mujer de la misma estatura tendría un peso entre unos 43 y 81 kg.

```
predicted_draws(modelo_peso_estatura_sexo, newdata = tibble(estatura =
  ↪ c(160, 160), sexo = c("H", "M")), value = "Predicción") |>
  ggplot(aes(x = Predicción, y = sexo, color = sexo)) +
  stat_halfeye(.width = c(0.5, 0.9), fill = "gray90") +
  scale_y_discrete(labels = c("H" = "Hombre", "M" = "Mujer")) +
  labs(x = "Peso (kg)", y = "Sexo") +
  theme_minimal()
```

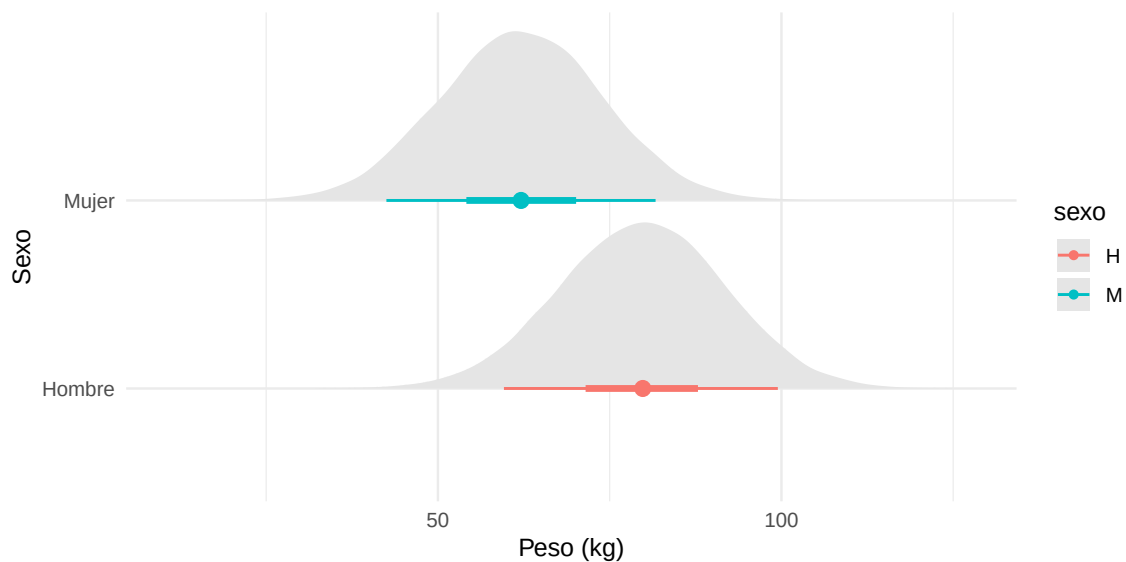


Figura 6.18: Distribuciones predictivas del peso de un hombre y de una mujer de 160 cm.

6.13.6. Distribuciones predictivas del modelo de regresión múltiple

Ejemplo 6.23. Obtenemos las distribuciones predictivas del peso para todas las personas de la muestra:

```

post_pred <- predicted_draws(modelo_peso_estatura_sexo, newdata = df,
  ↪ value = "Predicción")
ggplot(post_pred, aes(x = estatura, y = Predicción, color = sexo)) +
  stat_pointinterval(.width = c(0.5, 0.95), position =
    ↪ position_dodge(width = 0.5)) +
  geom_point(data = df, aes(y = peso), position = position_dodge(width =
    ↪ 0.5)) +
  labs(x = "Estatura (cm)", y = "Peso (kg)") +
  theme_minimal()

```

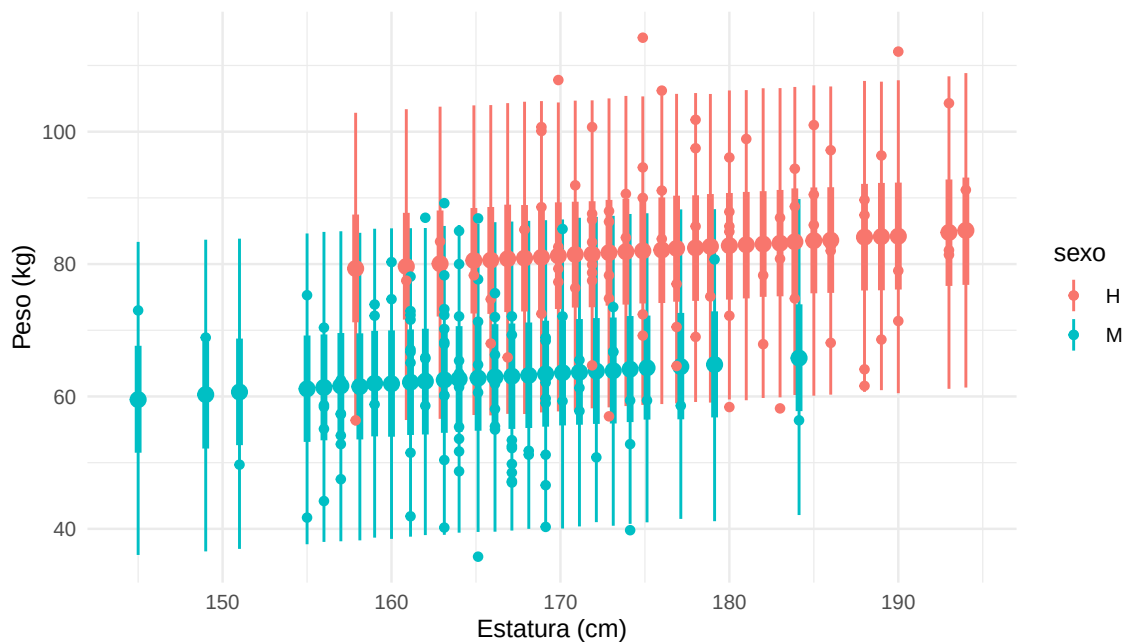


Figura 6.19: Intervalos predictivos del 50 % y del 95 % para cada persona de la muestra, con sus pesos observados.

6.13.7. Evaluación del modelo de regresión múltiple

Ejemplo 6.24. Evaluamos el modelo de regresión múltiple mediante validación cruzada con 10 pliegues:

```

set.seed(123)
eval_cv <- prediction_summary_cv(model = modelo_peso_estatura_sexo, data
  ↪ = df, k = 10, prob_outer = 0.9)
eval_cv$cv |> kable(digits = 2)

```

mae	mae_scaled	within_50	within_90
8.28	0.7	0.5	0.88

El error absoluto mediano por validación cruzada baja de unos 9,7 kg (modelo simple con la estatura) a unos 8,3 kg. Es prácticamente el mismo valor que el del modelo con solo el sexo (unos 8,1 kg, como vimos antes), así que la mejora procede sobre todo del sexo.

El coeficiente de determinación bayesiano del modelo múltiple es:

```
bayes_R2(modelo_peso_estatura_sexo) |>
  mean_qi(.width = 0.9) |>
  kable(digits = 3, col.names = c("R² medio", ".lower", ".upper",
    ↪ ".width", ".point", ".interval"))
```

R² medio	.lower	.upper	.width	.point	.interval
0.415	0.336	0.488	0.9	mean	qi

El modelo de regresión múltiple explica alrededor del 42 % de la variabilidad del peso, algo más que el modelo de regresión simple.

6.14. Comparación de modelos de regresión

En general, cuantas más variables independientes añadimos a un modelo de regresión, mayor es el coeficiente de determinación, porque la variabilidad no explicada disminuye. Sin embargo, eso no significa que el modelo sea mejor, ya que al aumentar el número de variables también aumenta el sobreajuste.

Para comparar la capacidad predictiva de distintos modelos, podemos usar la *densidad logarítmica predictiva esperada* (ELPD), que penaliza la complejidad del modelo y permite comparar modelos con distinto número de variables independientes.

6.15. Densidad logarítmica predictiva esperada (ELPD)

La ELPD mide el logaritmo de la densidad predictiva de los valores observados según el modelo ajustado con los datos de entrenamiento. Se calcula normalmente con validación cruzada de n pliegues, conocida como *leave-one-out cross-validation* (LOO-CV): en cada iteración se reserva un caso como prueba y se ajusta el modelo con los demás.

Para cada caso y_i de la muestra $(y_1, y_2, \dots, y_n)$ se calcula el logaritmo de su densidad predictiva según el modelo ajustado sin ese caso:

$$\log p(y_i|y_{-i}) = \log \int p(y_i|\theta) p(\theta|y_{-i}) d\theta,$$

donde y_{-i} es el conjunto de datos sin el caso y_i , $p(y_i|\theta)$ es la densidad de y_i dados los parámetros θ , y $p(\theta|y_{-i})$ es la distribución a posteriori de los parámetros ajustada sin ese caso.

Definición 6.1 (Densidad logarítmica predictiva esperada (ELPD)). Dada una muestra de datos $(y_1, \dots, y_n)$, la *densidad logarítmica predictiva esperada* de un modelo, calculada mediante LOO-CV, es

$$\text{elpd} = \sum_{i=1}^n \log p(y_i|y_{-i}) = \sum_{i=1}^n \log \int p(y_i|\theta) p(\theta|y_{-i}) d\theta.$$

Cuanto mayor es el valor de la ELPD, mejor es la capacidad predictiva del modelo, por lo que podemos comparar modelos mediante esta métrica.

i loo()

Calcula la ELPD de un modelo de regresión bayesiano ajustado con `stan_glm()` mediante validación cruzada LOO-CV. Se encuentra en el paquete `loo`.

Parámetros:

- **x**: modelo de regresión bayesiano ajustado con `stan_glm()`.

Ejemplo 6.25. Calculamos la ELPD del modelo de regresión lineal simple y del múltiple para comparar su capacidad predictiva:

```
library(loo)
# ELPD del modelo de regresión lineal simple
loo(modelo_peso_estatura)$estimates[1,] |> t() |> kable(digits = 1)
```

Estimate	SE
-810.6	9.8

```
# ELPD del modelo de regresión lineal múltiple
loo(modelo_peso_estatura_sexo)$estimates[1,] |> t() |> kable(digits = 1)
```

Estimate	SE
-781.6	9.7

La diferencia no se interpreta en términos absolutos, sino en relación con su error estándar.

La función `loo_compare()` del paquete `loo` permite comparar la ELPD de varios modelos ajustados con `stan_glm()`.

i `loo_compare()`

Compara la ELPD de varios modelos de regresión bayesianos a partir de los objetos calculados con `loo()`.

Parámetros:

- **x**: objetos `loo` de los modelos que se quieren comparar.

Ejemplo 6.26. Comparamos la ELPD de los modelos de regresión lineal simple y múltiple:

```
loo1 <- loo(modelo_peso_estatura)
loo2 <- loo(modelo_peso_estatura_sexo)
loo_compare(loo1, loo2)[, c("elpd_diff", "se_diff")] |> kable(digits =
  ↪ 2)
```

	elpd_diff	se_diff
modelo_peso_estatura_sexo	0.00	0.00
modelo_peso_estatura	-28.99	6.37

La diferencia tipificada de ELPD es el cociente entre la diferencia de ELPD y su error estándar, que en este caso es de alrededor de 4,6 (28,99 entre 6,37).

Se suele utilizar la siguiente regla, propuesta por Vehtari:

- Si $\Delta\text{ELPD}/\text{SE}(\text{ELPD}) < 2$, la diferencia no es sustancial.
- Si $2 \leq \Delta\text{ELPD}/\text{SE}(\text{ELPD}) < 4$, la diferencia es moderada.
- Si $\Delta\text{ELPD}/\text{SE}(\text{ELPD}) \geq 4$, la diferencia es sustancial.

Según esta regla, el modelo de regresión múltiple es sustancialmente mejor que el modelo de regresión simple.

Ejercicio 6.8. Compara la ELPD del modelo de regresión del peso sobre el sexo con la del modelo de regresión múltiple del peso sobre la estatura y el sexo. ¿Qué conclusiones se pueden extraer?

Solución

```
loo1 <- loo(modelo_peso_sexo)
loo2 <- loo(modelo_peso_estatura_sexo)
loo_compare(loo1, loo2)[, c("elpd_diff", "se_diff")] |> kable(digits =
  ↪ 2)
```

	elpd_diff	se_diff
modelo_peso_sexo	0.00	0.00
modelo_peso_estatura_sexo	-1.05	1.41

La diferencia tipificada de ELPD es de alrededor de 0,74 (1,05 entre 1,41), por debajo de 2. La diferencia no es sustancial: el modelo que solo incluye el sexo es ligeramente mejor que el múltiple, por lo que añadir la estatura al modelo del sexo no mejora la capacidad predictiva. Esto es coherente con la pequeña mejora que hemos visto antes al pasar de un modelo a otro, y puede deberse a que la estatura y el sexo están correlacionados, de modo que la estatura aporta poca información adicional cuando ya se conoce el sexo.

6.16. Modelos de regresión con interacciones

En un modelo de regresión múltiple sin interacción,

$$Y|\beta_0, \beta_1, \beta_2, \sigma \sim N(\beta_0 + \beta_1 X_1 + \beta_2 X_2, \sigma),$$

se asume que la pendiente β_1 es la misma para cualquier valor de X_2 . En el ejemplo anterior, esto quiere decir que el incremento de peso por centímetro de estatura es el mismo para hombres y mujeres. Pero esto no tiene por qué ser cierto: puede haber *interacción* entre las variables independientes, es decir, el efecto de una depende del valor de la otra.

Cuando no hay interacción entre X_1 y X_2 , las rectas de regresión correspondientes a cada valor de X_2 son paralelas. Si hay interacción, no lo son.

Para modelar la interacción introducimos un nuevo parámetro β_3 , coeficiente del producto de X_1 y X_2 :

$$Y|\beta_0, \beta_1, \beta_2, \beta_3, \sigma \sim N(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2, \sigma).$$

Ejemplo 6.27. Ajustamos el modelo con interacción entre la estatura y el sexo:

```
modelo_peso_estatura_sexo_interaccion <- stan_glm(
  peso ~ estatura * sexo,
  data = df,
  family = gaussian(),
  prior_intercept = normal(70, 10),
  prior = normal(location = c(2, -10, 0), scale = c(1, 5, 0.5)),
  prior_aux = exponential(0.5),
  chains = 4, iter = 5000 * 2, seed = 123
)
```

```
modelo_peso_estatura_sexo_interaccion |>
  gather_draws(`(Intercept)`, estatura, sexoM, `estatura:sexoM`, sigma)
  ↪ |>
  mean_qi(.width = 0.9) |>
  kable(digits = 2)
```

.variable	.value	.lower	.upper	.width	.point	.interval
(Intercept)	62.66	29.03	96.70	0.9	mean	qi
estatura	0.11	-0.08	0.30	0.9	mean	qi
estatura:sexoM	-0.06	-0.11	-0.01	0.9	mean	qi
sexoM	-9.24	-17.38	-1.16	0.9	mean	qi
sigma	11.77	10.84	12.78	0.9	mean	qi

Dibujamos las rectas de regresión a posteriori del modelo con interacción:

Ejemplo 6.28.

```
df |>
  add_epred_draws(modelo_peso_estatura_sexo_interaccion, ndraws = 50,
  ↪ value = "Predicción_media") |>
  ggplot(aes(x = estatura, y = peso, color = sexo)) +
  geom_point(data = df) +
  geom_line(aes(y = Predicción_media, group = paste(sexo, .draw)), alpha
  ↪ = 0.2) +
  labs(x = "Estatura (cm)", y = "Peso (kg)") +
  theme_minimal()
```

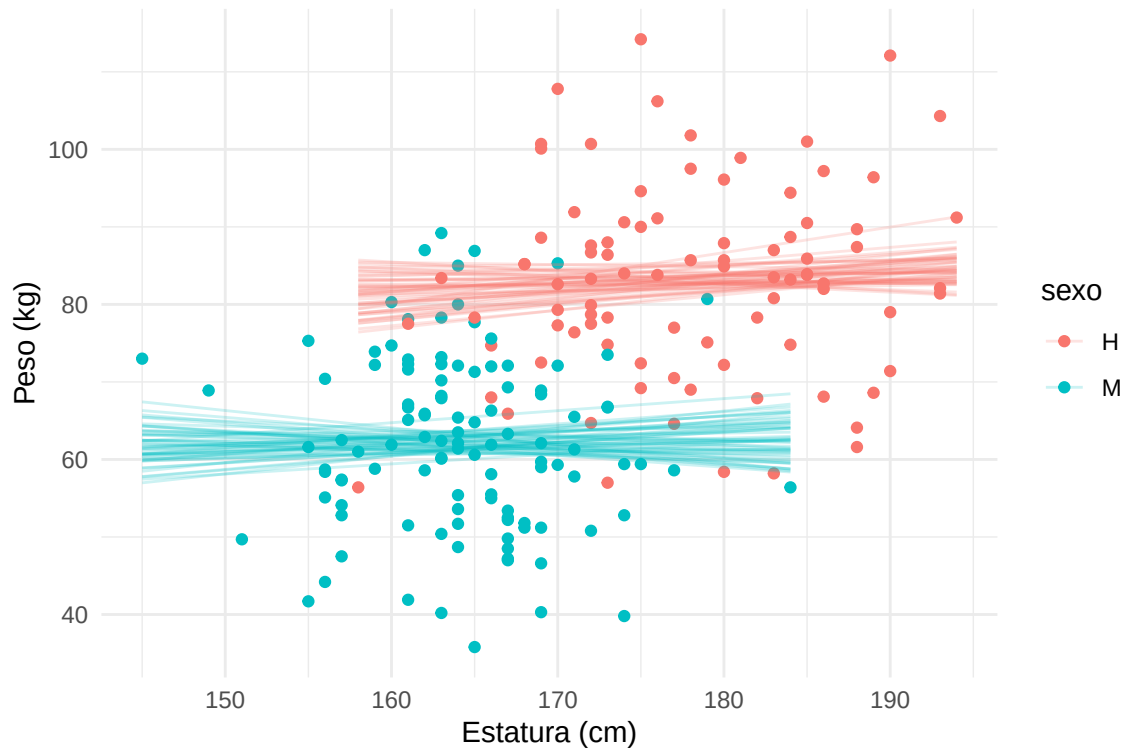


Figura 6.20: Cincuenta rectas de regresión a posteriori del modelo con interacción, para hombres y mujeres.

Aunque el modelo incluye la interacción, las rectas de regresión son prácticamente paralelas, lo que indica que casi no hay interacción entre la estatura y el sexo.

Ejercicio 6.9. Compara la ELPD del modelo múltiple sin interacción con la del modelo con interacción. ¿Qué conclusiones se pueden extraer?

Solución

```
loo1 <- loo(modelo_peso_estatura_sexo)
loo2 <- loo(modelo_peso_estatura_sexo_interaccion)
loo_compare(loo1, loo2)[, c("elpd_diff", "se_diff")] |> kable(digits =
  ↪ 2)
```

	elpd_diff	se_diff
modelo_peso_estatura_sexo_interaccion	0.00	0.00
modelo_peso_estatura_sexo	-0.41	0.83

La diferencia tipificada de ELPD es de alrededor de 0,5 (0,41 entre 0,83), por debajo de 2, así que no hay una diferencia sustancial entre los dos modelos. Por el principio de parsimonia, nos quedaríamos con el modelo sin interacción.

6.17. Regresión logística

6.17.1. Variables dependientes categóricas

En todos los modelos vistos hasta ahora, la variable dependiente Y era cuantitativa continua (el peso). Sin embargo, en muchos problemas la variable dependiente es *categorica* y lo que queremos modelizar es la *probabilidad* de que ocurra un suceso.

En epidemiología, por ejemplo, es muy habitual estudiar la probabilidad de que una persona padezca una enfermedad (enfermo / sano) en función de factores de riesgo como la edad, el sexo, el índice de masa corporal o los hábitos. En estos casos, la variable dependiente es dicotómica:

$$Y = \begin{cases} 1 & \text{si la persona padece la enfermedad,} \\ 0 & \text{si la persona no la padece.} \end{cases}$$

Un modelo de regresión lineal no es adecuado aquí, porque podría predecir valores de Y por debajo de 0 o por encima de 1, que no tienen sentido como probabilidades.

6.17.2. El modelo de regresión logística

Cuando la variable dependiente es dicotómica, suponemos que sigue una distribución de **Bernoulli** con probabilidad de éxito π_i , que es la probabilidad de padecer la enfermedad para el individuo i :

$$Y_i \mid \pi_i \sim \text{Bernoulli}(\pi_i).$$

Como π_i debe estar entre 0 y 1, no podemos modelizarla directamente como combinación lineal de las variables independientes. En su lugar, modelizamos una transformación conocida como *logaritmo de las odds* (*log-odds* o *logit*):

$$\text{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 x_i,$$

donde el cociente $\frac{\pi_i}{1-\pi_i}$ es la *odds* de padecer la enfermedad, que expresa cuántas veces es más probable padecerla que no padecerla.

6.17.3. La función logística

Despejando π_i se obtiene la *función logística*, que da nombre al modelo. Transforma cualquier valor real del predictor lineal en una probabilidad entre 0 y 1:

$$\pi_i = \frac{e^{\beta_0 + \beta_1 x_i}}{1 + e^{\beta_0 + \beta_1 x_i}} = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_i)}}.$$

```
tibble(x = seq(-10, 10, 0.1)) |>
  mutate(pi = plogis(x)) |>
  ggplot(aes(x = x, y = pi)) +
  geom_line(color = myblue, linewidth = 1.2) +
  geom_hline(yintercept = c(0, 1), linetype = "dashed", color = mygray)
  ↪ +
  labs(x = expression(beta[0] + beta[1] * x), y = expression(pi)) +
  theme_minimal()
```

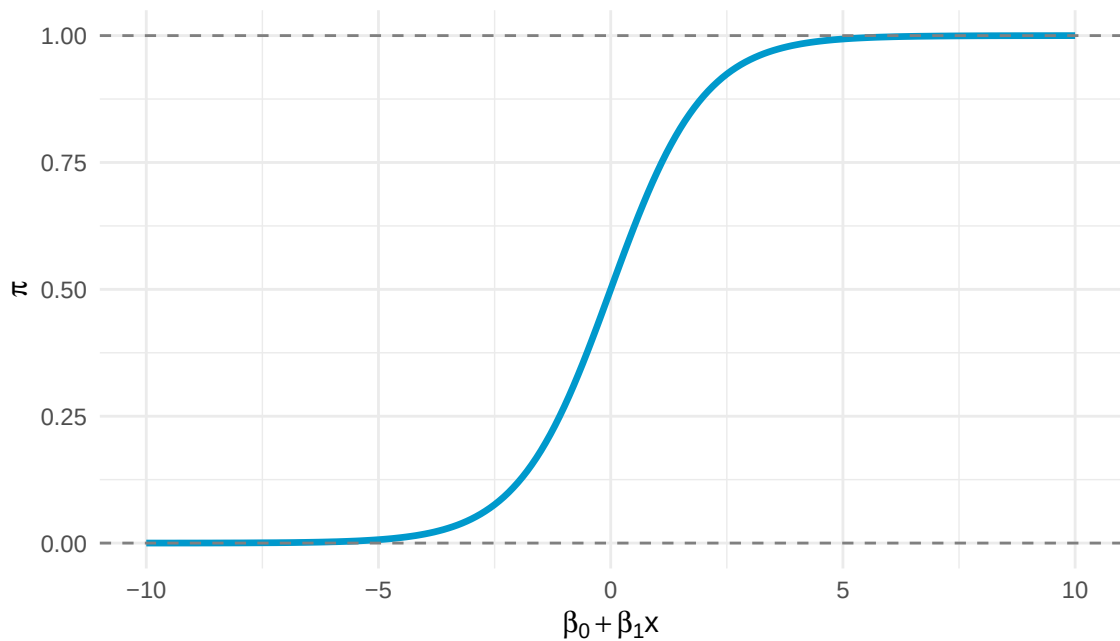


Figura 6.21: Función logística, que transforma el predictor lineal en una probabilidad.

La función logística tiene forma de S, es creciente y toma siempre valores entre 0 y 1, por lo que es ideal para modelizar probabilidades.

6.17.4. Interpretación de los coeficientes

En la regresión logística los coeficientes se interpretan en la escala de las *odds*:

- **Término independiente** β_0 . Es el logaritmo de la odds de padecer la enfermedad cuando $X = 0$. Es decir, e^{β_0} es la odds base.
- **Pendiente** β_1 . Es el cambio en el logaritmo de la odds por cada unidad que aumenta X . Su exponencial e^{β_1} , la *razón de odds* (*odds ratio*), indica por qué factor se multiplica la odds de padecer la enfermedad por cada unidad que aumenta X :
 - Si $e^{\beta_1} > 1$, la variable X es un **factor de riesgo**.
 - Si $e^{\beta_1} < 1$, la variable X es un **factor de protección**.
 - Si $e^{\beta_1} = 1$, la variable X no influye en la probabilidad de enfermedad.

6.17.5. Construcción del modelo de regresión logística

El modelo se especifica como

$$Y_i | \beta_0, \beta_1 \sim \text{Bernoulli}(\pi_i) \\ \text{logit}(\pi_i) = \beta_0 + \beta_1 x_i$$

Advertencia

A diferencia de la regresión lineal, no hay un parámetro σ para la desviación típica del error, porque la variabilidad de una distribución de Bernoulli queda determinada por su probabilidad π_i .

Como siempre, asignamos distribuciones a priori a los coeficientes y las actualizamos con los datos para obtener sus distribuciones a posteriori.

6.17.6. Distribuciones a priori de los parámetros

Como β_0 y β_1 son parámetros de localización que pueden tomar cualquier valor real (están en la escala logarítmica de las odds), una distribución a priori adecuada para ambos es la normal:

$$\beta_0 \sim N(\theta_0, \tau_0), \quad \beta_1 \sim N(\theta_1, \tau_1).$$

Conviene tener presente que, en la escala *logit*, valores aparentemente pequeños de los coeficientes pueden corresponder a cambios grandes en la probabilidad. Por ejemplo, una distribución a priori $\beta_1 \sim N(0, 0.5)$ ya es bastante informativa, porque asume que la

razón de odds e^{β_1} está, con un 95 % de probabilidad, aproximadamente entre $e^{-1} \approx 0,37$ y $e^1 \approx 2,72$.

Ejemplo 6.29. Supongamos que queremos estudiar la **prevalencia de diabetes tipo 2** en una población adulta en función de la edad. Disponemos de los datos de una cohorte de 250 personas en las que se ha registrado su edad y si padecen o no diabetes.

Para este ejemplo simulamos los datos a partir de un modelo conocido, lo que nos permitirá comprobar después si el modelo bayesiano recupera la prevalencia real:

```
set.seed(123)
n <- 250
edad <- round(runif(n, 30, 85))
imc <- round(rnorm(n, 27, 4), 1)
# Modelo real: logit(pi) = -9 + 0.07 * edad + 0.12 * imc
prob_real <- plogis(-9 + 0.07 * edad + 0.12 * imc)
diabetes <- rbinom(n, size = 1, prob = prob_real)
df <- tibble(edad = edad, imc = imc, diabetes = diabetes)
```

La *prevalencia* es la proporción de individuos de la población que padecen la enfermedad. En la muestra, la prevalencia observada es

```
df |> summarise(prevalencia = mean(diabetes)) |> kable(digits = 3)
```

prevalencia
0.18

Representamos la presencia de diabetes frente a la edad. Como la variable dependiente solo toma los valores 0 y 1, añadimos un pequeño ruido vertical (*jitter*) para visualizar mejor la densidad de puntos:

```
ggplot(df, aes(x = edad, y = diabetes)) +
  geom_jitter(height = 0.05, alpha = 0.4, color = myblue) +
  labs(x = "Edad (años)", y = "Diabetes (0 = no, 1 = sí)") +
  theme_minimal()
```

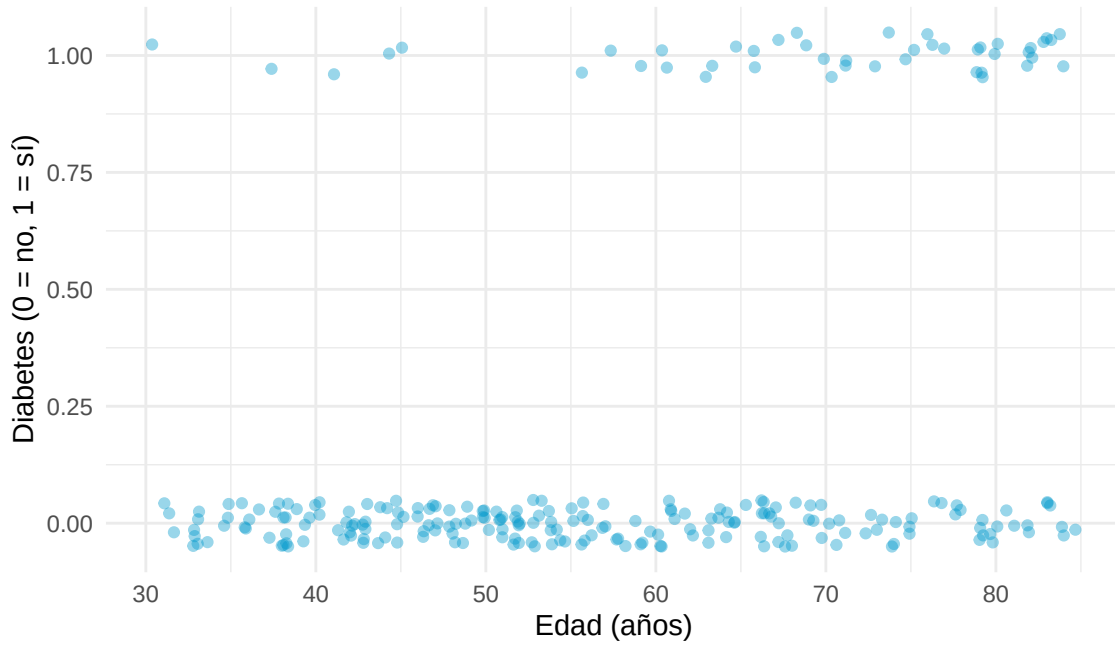


Figura 6.22: Presencia de diabetes (0 = no, 1 = sí) según la edad en la cohorte simulada.

En este contexto usamos distribuciones a priori poco informativas, para que los datos tengan un peso importante en la estimación. Para β_0 asumimos una normal de media $-1,5$ y desviación típica $1,5$, lo que implica que la odds de padecer diabetes es baja para una persona de edad media:

$$\beta_0 \sim N(-1,5; 1,5).$$

Para β_1 asumimos una normal de media 0 y desviación típica $0,5$, lo que implica un efecto de la edad moderado y de signo desconocido:

$$\beta_1 \sim N(0; 0,5).$$

6.17.7. Ajuste del modelo de regresión logística bayesiana

Para ajustar un modelo de regresión logística bayesiano usamos de nuevo `stan_glm()`, indicando la familia `binomial()`, que utiliza por defecto la función de enlace *logit*.

6.17.8. Validación de las distribuciones a priori

Como con la regresión lineal, conviene comprobar que las distribuciones a priori son razonables antes de ajustar el modelo. Para ello ajustamos el modelo con `prior_PD = TRUE`, que simula los parámetros a partir de sus distribuciones a priori sin usar los datos.

Ejemplo 6.30. Obtenemos las distribuciones a priori del modelo de regresión logística de la probabilidad de diabetes sobre la edad:

```
library(rstanarm)
set.seed(123)
modelo_diabetes_prior <- stan_glm(
  diabetes ~ edad, data = df, family = binomial(),
  prior_intercept = normal(-1.5, 1.5),
  prior = normal(0, 0.5),
  chains = 4, iter = 5000 * 2, seed = 123,
  prior_PD = TRUE
)
```

Dibujamos algunas de las curvas logísticas generadas a partir de las distribuciones a priori para comprobar que cubren un rango razonable de posibilidades:

```
tibble(edad = seq(30, 85, length.out = 50)) |>
  add_epred_draws(modelo_diabetes_prior, ndraws = 100, value = "prob")
  ↪ |>
  ggplot(aes(x = edad, y = prob, group = .draw)) +
  geom_line(color = myred, alpha = 0.2) +
  labs(x = "Edad (años)", y = "Probabilidad de diabetes") +
  theme_minimal()
```

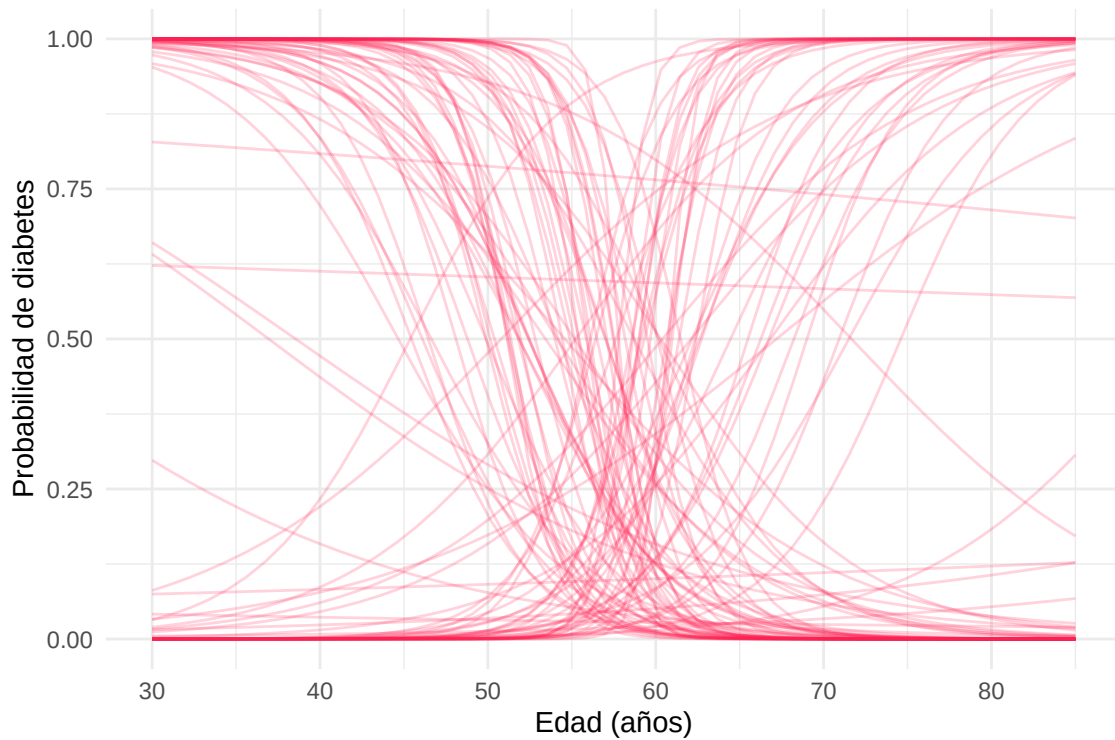


Figura 6.23: Cien curvas logísticas simuladas a partir de las distribuciones a priori.

Las curvas a priori contemplan tanto relaciones crecientes como decrecientes con la edad y cubren todo el rango de probabilidades, lo que indica que las distribuciones a priori son poco informativas y razonables.

6.17.9. Ajuste de las distribuciones a posteriori

Una vez validadas las distribuciones a priori, ajustamos el modelo a los datos actualizando el modelo anterior con `prior_PD = FALSE`:

Ejemplo 6.31.

```
modelo_diabetes <- update(modelo_diabetes_prior, prior_PD = FALSE)
```

Comprobamos la convergencia de las cadenas de Markov:

```
library(bayesplot)
mcmc_trace(modelo_diabetes, size = 0.2)
```

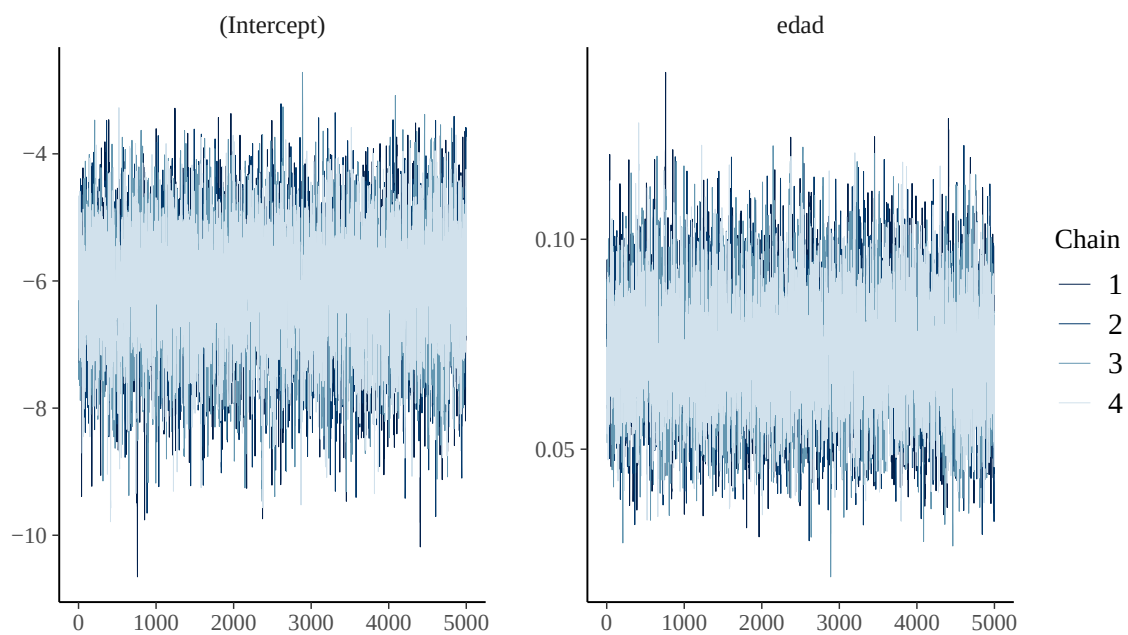


Figura 6.24: Trazas de las cadenas de Markov del modelo de regresión logística.

```
mcmc_dens_overlay(modelo_diabetes)
```

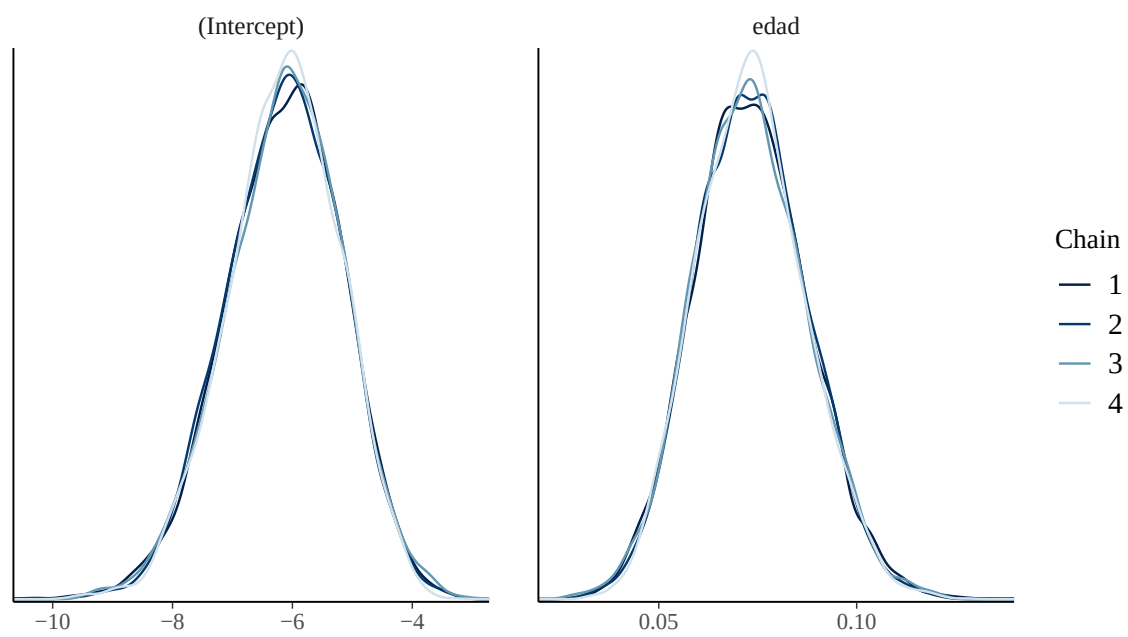


Figura 6.25: Distribuciones a posteriori de los parámetros del modelo logístico, por cadena.

```
neff_ratio(modelo_diabetes) |> t() |> kable(digits = 3)
```

(Intercept)	edad
0.403	0.427

```
rhat(modelo_diabetes) |> t() |> kable(digits = 3)
```

(Intercept)	edad
1	1

Las cadenas son estables y convergentes: el número efectivo de simulaciones es razonable para ambos parámetros y el valor de $\hat{R}$ es prácticamente 1.

6.17.10. Intervalos de credibilidad

Ejemplo 6.32. Obtenemos la media y los intervalos de credibilidad del 90 % de los parámetros:

```
modelo_diabetes |>
  gather_draws(`(Intercept)`, edad) |>
  mean_qi(.width = 0.9) |>
  kable(digits = 3)
```

.variable	.value	.lower	.upper	.width	.point	.interval
(Intercept)	-6.137	-7.768	-4.627	0.9	mean	qi
edad	0.073	0.051	0.097	0.9	mean	qi

El coeficiente de la edad es positivo, lo que indica que, a mayor edad, mayor probabilidad de diabetes.

6.17.11. Razón de odds

Para interpretar el coeficiente de la edad en términos más intuitivos, calculamos la razón de odds e^{β_1} :

```

modelo_diabetes |>
  gather_draws(edad) |>
  mutate(odds_ratio = exp(.value)) |>
  mean_qi(odds_ratio, .width = 0.9) |>
  kable(digits = 3)

```

.variable	odds_ratio	.lower	.upper	.width	.point	.interval
edad	1.076	1.052	1.102	0.9	mean	qi

Es decir, por cada año más de edad, la odds de padecer diabetes se multiplica por aproximadamente 1,07 (un incremento del 7 % en la odds por año), y con una probabilidad del 90 % el factor está entre 1,05 y 1,1.

6.17.12. La curva logística a posteriori

Cada muestra de la distribución a posteriori de los parámetros define una curva logística distinta. Podemos dibujar algunas de ellas sobre los datos para visualizar el ajuste y la incertidumbre:

```

tibble(edad = seq(30, 85, length.out = 100)) |>
  add_epred_draws(modelo_diabetes, ndraws = 50, value = "prob") |>
  ggplot(aes(x = edad, y = prob)) +
  geom_line(aes(group = .draw), color = myred, alpha = 0.2) +
  geom_jitter(data = df, aes(y = diabetes), height = 0.03,
    alpha = 0.3, color = myblue) +
  labs(x = "Edad (años)", y = "Probabilidad de diabetes") +
  theme_minimal()

```

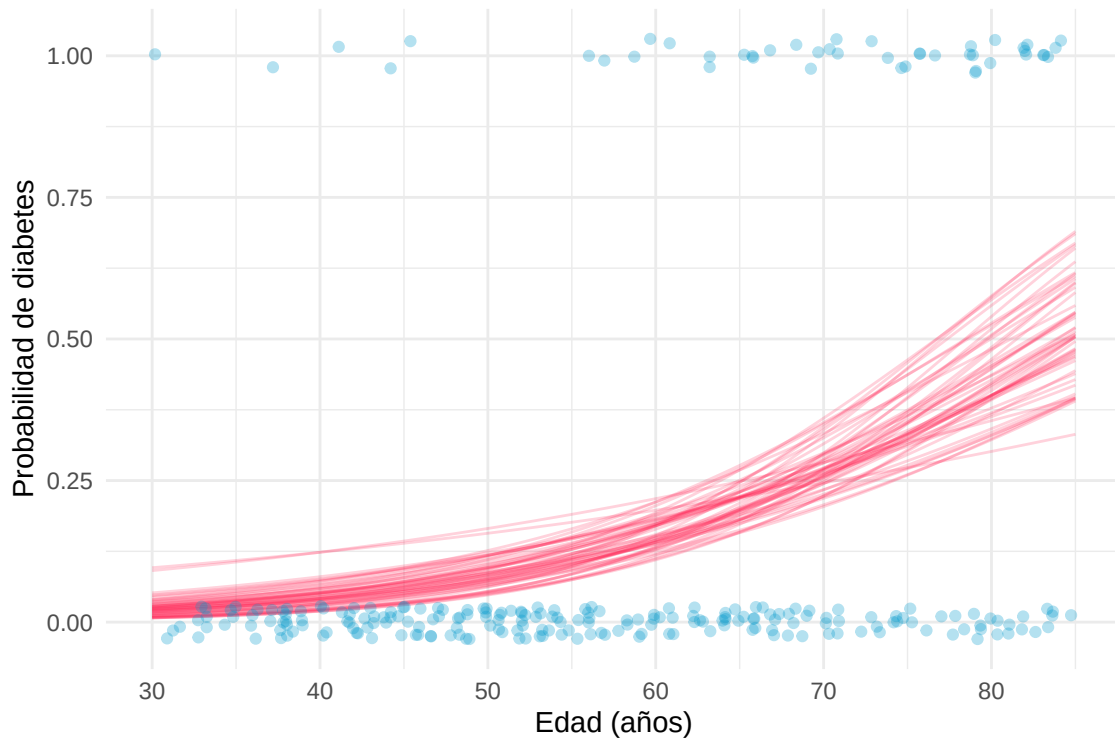


Figura 6.26: Cincuenta curvas logísticas a posteriori sobre los datos de diabetes.

Se observa cómo la probabilidad de padecer diabetes aumenta con la edad, siguiendo una curva en forma de S.

6.17.13. Estimación de la prevalencia

En estadística bayesiana, la estimación de un parámetro es su propia distribución a posteriori, que describimos habitualmente con su media o mediana y su intervalo de credibilidad.

Ejemplo 6.33. Para estimar la prevalencia de la diabetes calculamos, para cada muestra de la distribución a posteriori, la probabilidad media de diabetes en toda la muestra:

```
df |>
  add_epred_draws(modelo_diabetes, value = "prob") |>
  group_by(.draw) |>
  summarise(prevalencia = mean(prob)) |>
  mean_qi(prevalencia, .width = 0.9) |>
  kable(digits = 3)
```

prevalencia	.lower	.upper	.width	.point	.interval
0.181	0.145	0.218	0.9	mean	qi

La prevalencia estimada de diabetes en la población es de alrededor del 18 %, con un intervalo de credibilidad del 90 % que va del 15 % al 22 %.

6.17.14. Predicción de la probabilidad para nuevos casos

Para predecir la probabilidad de padecer la enfermedad de nuevos individuos usamos `epred_draws()`, que genera la distribución a posteriori de la probabilidad π para cada nuevo caso.

Ejemplo 6.34. Calculamos la probabilidad de padecer diabetes para personas de 40, 60 y 80 años:

```
nuevos <- tibble(edad = c(40, 60, 80))
epred_draws(modelo_diabetes, newdata = nuevos, value = "prob") |>
  mean_qi(.width = 0.9) |>
  kable(digits = 3)
```

edad	.row	prob	.lower	.upper	.width	.point	.interval
40	1	0.042	0.019	0.073	0.9	mean	qi
60	2	0.152	0.111	0.196	0.9	mean	qi
80	3	0.435	0.334	0.539	0.9	mean	qi

Para una persona de 40 años, la probabilidad de padecer diabetes es baja, mientras que para una de 80 años es considerablemente mayor.

Representamos gráficamente estas distribuciones a posteriori de la probabilidad:

```
epred_draws(modelo_diabetes, newdata = nuevos, value = "prob") |>
  mutate(edad = factor(edad)) |>
  ggplot(aes(x = prob, y = edad, fill = edad)) +
  stat_halfeye(.width = 0.9, alpha = 0.6) +
  labs(x = "Probabilidad de diabetes", y = "Edad (años)") +
  theme_minimal() +
  theme(legend.position = "none")
```

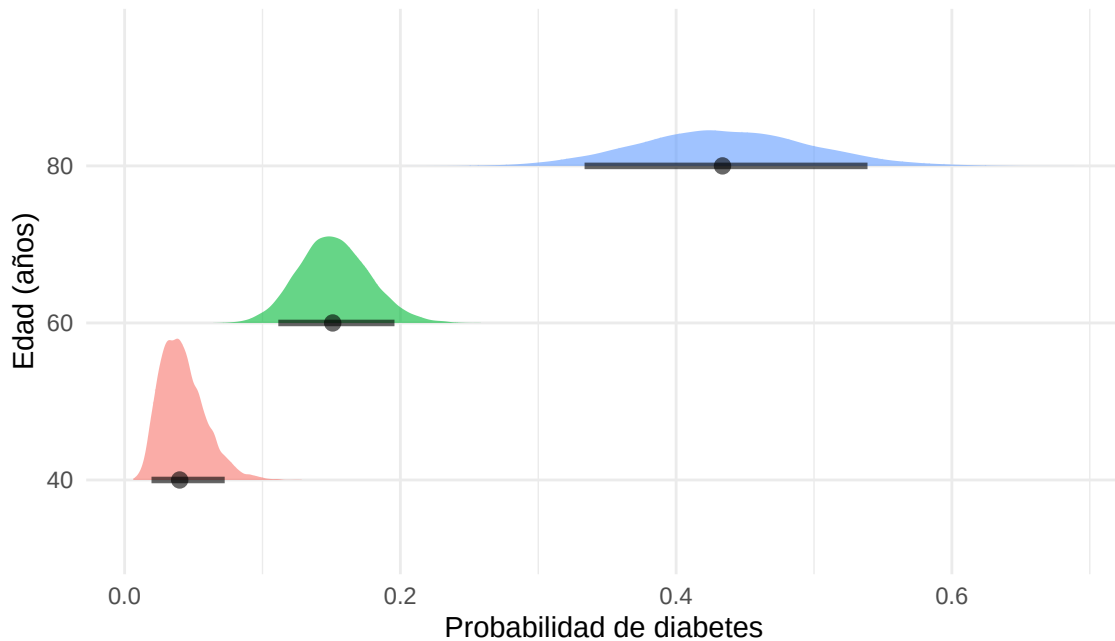


Figura 6.27: Distribución a posteriori de la probabilidad de diabetes para personas de 40, 60 y 80 años.

6.17.15. Clasificación de nuevos casos

Además de predecir la probabilidad de padecer la enfermedad, a menudo queremos *clasificar* a cada individuo como enfermo o sano. Para ello fijamos un *umbral de clasificación* (*cutoff*): si la probabilidad estimada supera el umbral, clasificamos al individuo como enfermo.

El umbral más habitual es 0,5, pero puede ajustarse según el coste relativo de los errores. En cribados de enfermedades graves se suele bajar el umbral para no dejar pasar a posibles enfermos, es decir, para aumentar la sensibilidad.

Para obtener las clases predichas usamos `predicted_draws()`, que simula valores 0/1 de la distribución de Bernoulli para cada caso. La proporción de unos es una estimación de la probabilidad:

Ejemplo 6.35. Clasificamos a las personas de 40, 60 y 80 años con un umbral de 0,4:

```
predicted_draws(modelo_diabetes, newdata = nuevos, value = "clase") |>
  group_by(edad) |>
  summarise(prob_diabetes = mean(clase)) |>
  mutate(clasificacion = if_else(prob_diabetes > 0.4, "Enfermo",
    ↪ "Sano")) |>
  kable(digits = 3)
```

edad	prob_diabetes	clasificacion
40	0.039	Sano
60	0.152	Sano
80	0.438	Enfermo

Con el umbral de 0,4, solo las personas de 80 años serían clasificadas como enfermas.

6.17.16. Evaluación del modelo: matriz de confusión

Para evaluar la capacidad de clasificación del modelo comparamos las clases predichas con las observadas, y construimos una **matriz de confusión** que clasifica cada caso como:

- **Verdadero positivo (VP)**: enfermo clasificado como enfermo.
- **Verdadero negativo (VN)**: sano clasificado como sano.
- **Falso positivo (FP)**: sano clasificado como enfermo.
- **Falso negativo (FN)**: enfermo clasificado como sano.

A partir de la matriz de confusión se calculan:

- **Sensibilidad** $= \frac{VP}{VP + FN}$: proporción de enfermos correctamente clasificados.
- **Especificidad** $= \frac{VN}{VN + FP}$: proporción de sanos correctamente clasificados.
- **Exactitud** (*accuracy*) $= \frac{VP + VN}{VP + VN + FP + FN}$: proporción total de aciertos.

Para calcular la matriz de confusión y las métricas de clasificación de un modelo de regresión logística bayesiano podemos usar la función `classification_summary()` del paquete `bayesrules`.

i `classification_summary()`

Devuelve la matriz de confusión y las métricas de sensibilidad, especificidad y exactitud de un modelo de clasificación bayesiano.

Parámetros:

- **model**: modelo ajustado con `stan_glm()`.
- **data**: conjunto de datos con la variable dependiente y las variables independientes.
- **cutoff**: umbral de clasificación. Por defecto 0,5.

Ejemplo 6.36. Calculamos la matriz de confusión y las métricas de clasificación del modelo de diabetes sobre la edad, con un umbral de 0,4:

```
library(bayesrules)
set.seed(123)
clasificacion <- classification_summary(model = modelo_diabetes, data =
  ↪ df, cutoff = 0.4)
clasificacion$confusion_matrix |> kable()
```

y	0	1
0	185	20
1	29	16

```
clasificacion$accuracy_rates |> kable(digits = 3)
```

sensitivity	0.356
specificity	0.902
overall_accuracy	0.804

La exactitud global es de 0,80, pero la sensibilidad es baja (0,36): con el umbral de 0,4, el modelo clasifica bien a los sanos (especificidad de 0,90), pero deja sin detectar a muchos enfermos. Esto es habitual cuando la enfermedad es poco prevalente.

6.17.17. Ajuste del umbral de clasificación

Cuando interesa detectar el mayor número posible de enfermos, por ejemplo en un cribado, podemos bajar el umbral para aumentar la sensibilidad a costa de reducir la especificidad.

Ejemplo 6.37. Repetimos la evaluación con un umbral de 0,2:

```
set.seed(123)
classification_summary(model = modelo_diabetes, data = df, cutoff =
  ↪ 0.2)$accuracy_rates |>
  kable(digits = 3)
```

sensitivity	0.733
-------------	-------

specificity	0.712
overall_accuracy	0.716

Al bajar el umbral aumenta la sensibilidad (detectamos más enfermos) y disminuye la especificidad (clasificamos como enfermos a más sanos). La elección del umbral depende del coste relativo de los falsos positivos y de los falsos negativos.

6.17.18. Evaluación mediante validación cruzada

Al igual que en la regresión lineal, las métricas calculadas sobre el propio conjunto de datos tienden a ser optimistas por el sobreajuste. Para obtener una estimación más realista usamos la validación cruzada con la función `classification_summary_cv()` del paquete `bayesrules`.

i `classification_summary_cv()`

Devuelve las métricas de clasificación de un modelo bayesiano mediante validación cruzada.

Parámetros:

- `model`: modelo ajustado con `stan_glm()`.
- `data`: conjunto de datos.
- `k`: número de pliegues.
- `cutoff`: umbral de clasificación. Por defecto 0,5.

Ejemplo 6.38. Evaluamos el modelo mediante validación cruzada con 10 pliegues:

```
set.seed(123)
cv_diabetes <- classification_summary_cv(model = modelo_diabetes, data
  ↪ = df, k = 10, cutoff = 0.4)
cv_diabetes$cv |> kable(digits = 3)
```

sensitivity	specificity	overall_accuracy
0.196	0.908	0.792

Las métricas obtenidas por validación cruzada son algo peores que las obtenidas sobre el conjunto de entrenamiento, lo que da una idea más realista de la capacidad predictiva del modelo con datos nuevos.

6.18. Regresión logística múltiple

Al igual que en la regresión lineal, podemos incluir varias variables independientes en el modelo logístico para mejorar la predicción de la probabilidad de la enfermedad:

$$\text{logit}(\pi_i) = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_p x_{pi}.$$

Cada coeficiente β_j es el cambio en el logaritmo de la odds por cada unidad que aumenta X_j , manteniendo constantes las demás variables. Su exponencial e^{β_j} es la razón de odds ajustada por el resto de factores de riesgo.

Ejercicio 6.10. Ajusta un modelo de regresión logística múltiple para predecir la probabilidad de diabetes en función de la edad y del índice de masa corporal (IMC), usando los datos simulados anteriormente. Compara su capacidad predictiva con la del modelo que solo incluye la edad mediante la ELPD.

Solución

Ajustamos el modelo añadiendo el IMC como variable independiente:

```
set.seed(123)
modelo_diabetes_imc <- stan_glm(
  diabetes ~ edad + imc, data = df, family = binomial(),
  prior_intercept = normal(-1.5, 1.5),
  prior = normal(0, 0.5),
  chains = 4, iter = 5000 * 2, seed = 123
)
```

```
modelo_diabetes_imc |>
  gather_draws(`(Intercept)` , edad, imc) |>
  mean_qi(.width = 0.9) |>
  kable(digits = 3)
```

.variable	.value	.lower	.upper	.width	.point	.interval
(Intercept)	-9.378	-12.303	-6.619	0.9	mean	qi
edad	0.075	0.052	0.099	0.9	mean	qi
imc	0.114	0.038	0.191	0.9	mean	qi

Tanto la edad como el IMC tienen coeficientes positivos, lo que indica que ambos son factores de riesgo para la diabetes.

Comparamos los dos modelos con la ELPD mediante validación cruzada LOO-CV:

```

loo_edad <- loo(modelo_diabetes)
loo_edad_imc <- loo(modelo_diabetes_imc)
loo_compare(loo_edad, loo_edad_imc)[, c("elpd_diff", "se_diff")] |>
  ↪ kable(digits = 2)

```

	elpd_diff	se_diff
modelo_diabetes_imc	0.00	0.00
modelo_diabetes	-1.92	2.66

El modelo con IMC tiene una ELPD ligeramente mayor, pero la diferencia es pequeña (1,9 con un error estándar de 2,7; cociente de unos 0,7), así que no es sustancial. Aun así, el IMC aparece con un coeficiente positivo, coherente con que hayamos generado los datos a partir de un modelo que depende de la edad y del IMC.

6.19. Referencias del capítulo

- Gelman, A., y Hill, J. (2006). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press.
- Vehtari, A., Gelman, A., y Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27, 1413-1432.
- Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., y Gelman, A. (2019). Visualization in Bayesian workflow. *Journal of the Royal Statistical Society: Series A*, 182(2), 389-402.