EST-46115: Modelación Bayesiana

Profesor: Alfredo Garbuno Iñigo — Primavera, 2022 — Muestreo predictivo.
Objetivo: Estudiaremos métodos de validación de supuestos, evaluación y crítica de modelos de acuerdo a las distribuciones predictivas (previa y/o posterior). Estableceremos ciertas conexiones con temas clásicos de inferencia frecuentista.
Lectura recomendada: Puedes leer la documentación de `Stan` sobre el tema aqui. También el capítulo 6 de [5] es una excelente referencia sobre el tema. Adicional, podrías checar la sección 2.3 de [7].

1. INTRODUCCIÓN

Como hemos visto podemos utilizar la distribución predictiva para traducir nuestros supuestos distribucionales (estado de conocimiento) en cantidades observables. En esta sección del curso usaremos la `distribución predictiva previa` para evaluar consistencia de cómo matematizamos nuestro conocimiento previo con respecto al de un experto. Por otro lado, utilizaremos la `distribución predictiva posterior` con el objetivo de evaluar y criticar el ajuste del modelo propuesto.

2. MUESTREO PREDICTIVO – PREVIA

Usualmente definimos la distribución predictiva previa de manera que refleje lo que sabemos del problema que estamos modelando. Esperaríamos que fuera fácil en situaciones sencillas. Pero incluso en modelos medianamente complejos es difícil 1) poder definir dicha distribución y 2) entender las consecuencias de esa elección.

2.1. Muestreo predictivo

Recordemos que la distribución predictiva previa es

$$\pi(\tilde{y}) = \int \pi(\tilde{y}|\theta)\,\pi(\theta)\,\mathrm{d}\theta\,. \tag{1}$$

El mecanismo que utilizamos para generar muestras de dicha distribución es el siguiente:

- Para $i = 1, \ldots, N$:
  1. Generar $\tilde{\theta}_i \sim \pi(\theta)$.
  2. Generar $\tilde{y}_i \sim \pi(\tilde{y}|\tilde{\theta}_i)$.

Al final de este proceso podemos descartar las simulaciones de $\tilde{\theta}$ y obtener una colección de realizaciones aleatorias de $\tilde{y}_i \sim \pi(\tilde{y})$.

2.1.1. Para pensar: ¿Por qué este proceso iterativo nos deja muestras de la marginal? *Hint*: considera el caso para dos variables y consider los gráficos de dispersión y los histogramas individuales.

2.2. Interpretación

El objetivo de utilizar la distribución predictiva previa es evaluar la interacción de la previa con la verosimilitud.

2.3. Ejemplo: cantantes

Usualmente consideramos algún resumen informativo de los datos. Es decir, un resumen a través de un `estadístico` de nuestra distribución, y en consecuencia lo que nos interesaría es evaluar la `distribución de muestreo` de dicho estadístico. La idea es poder definir regiones donde esperaríamos ver nuestra simulación de posibles estadísticos.

Por ejemplo, consideremos el caso de los cantantes de ópera. El modelo asume una distribución previa de la forma

$$\mu|\sigma \sim \text{Normal}\left(\mu_0, \frac{\sigma}{\sqrt{n_0}}\right), \tag{2}$$

$$\sigma^{-2} \sim \text{Gamma}(a_0, b_0)\,. \tag{3}$$

Donde los parámetros tienen la siguiente especificación:

```
previa.params  ← within(list(),
{
  mu0 ← 175
  n0  ← 5
  a0  ← 3
  b0  ← 147
})
```

Una manera práctica es hacer remuestreo de la distribución predictiva previa para poder evaluar ciertos estadísticos de resumen. La idea es que podamos definir una región donde la previa genere datos razonables. Por ejemplo, podemos definir una zona donde esperamos observar el promedio de la altura de 200 cantantes. Ver Fig. 1.

```
data {
  real mu0;
  real<lower=0> n0;
  real<lower=0> a0;
  real<lower=0> b0;
}
generated quantities {
  real tau = gamma_rng(a0, b0);
  real sigma = 1/sqrt(tau);
  real mu  = normal_rng(mu0, sigma/sqrt(n0));
  real y_tilde = normal_rng(mu, sigma);
}
```

```
replica ← function(id){
  previa ← modelo$sample(previa.params,
                         fixed_param = TRUE,
                         refresh = 0, chains = 1,
                         show_messages = FALSE)
  list(mean = mean(previa$draws(format = "df")$y_tilde),
       sd   = sd(previa$draws(format = "df")$y_tilde),
       min  = min(previa$draws(format = "df")$y_tilde),
       max  = max(previa$draws(format = "df")$y_tilde))
  }

resultados ← tibble(id = 1:200) ▷
  mutate(results = map_df(id, replica))
```

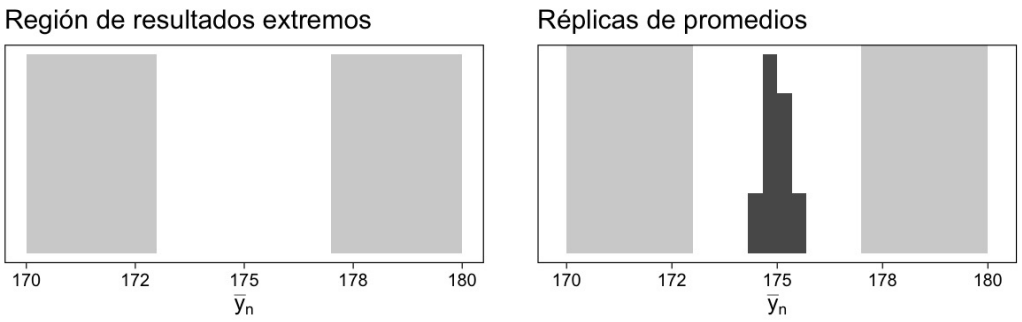

FIGURA 1. *Definimos una región donde consideremos resultados extremos de una estadística resumen para poder evaluar la distribución muestral de nuestro estadístico producto de la distribución predictiva previa.*

Por supuesto escoger el resumen correcto es complicado pero con la respuesta clara que queremos responder nos indicará cuál escoger. Podríamos haber escogido cualquier otro resumen estadístico y graficar histogramas para evaluar las simulaciones bajo la distribución predictiva previa. Ver Fig. 2.

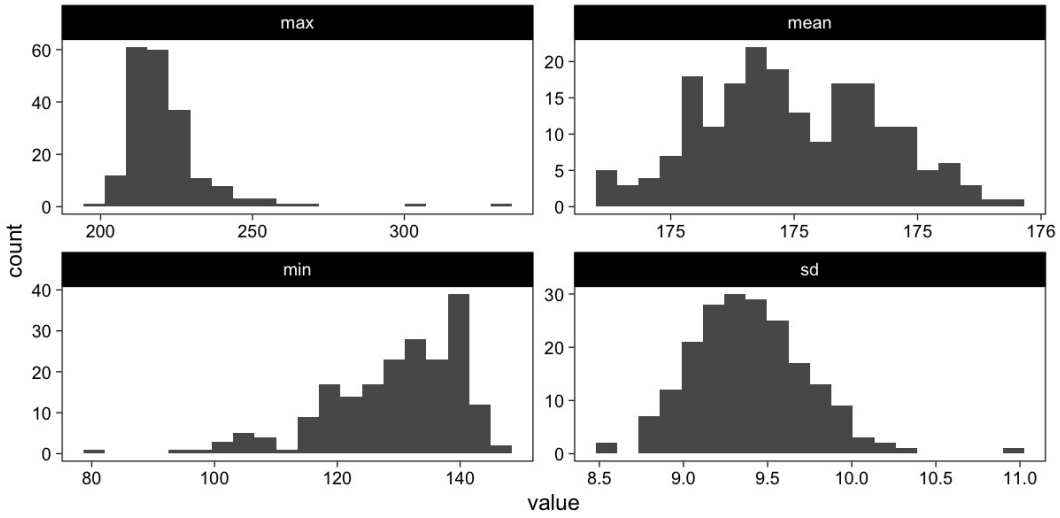

FIGURA 2. *Estadisticos bajo réplicas de la distirbución predictiva previa.*

2.4. Ejemplo: juegos de soccer

Este ejemplo lo hemos tomado de la documentación de Stan. Consideremos que estamos modelando partidos de *soccer* en una liga que tiene J equipos. Cada equipo tiene una tasa de goles λ_j. Además, asumimos que cada equipo mete goles de acuerdo a una distribución Poisson con tasa λ_j.

Utilizaremos, para ilustrar, una distribución a priori

$$\lambda_j \sim \mathsf{Gamma}(\epsilon_1, \epsilon_2)\,, \tag{4}$$

donde los parámetros ϵ_i se escogen de acuerdo a recomendación en [6]. Lo cual corresponde a una previa `no informativa`.

`Definición (Distribución no informativa)`: Decimos que una distribución previa es **no informativa** si dicha distribución aporta poca información relativa al experimento [2].

Supongamos que la liga juega un torneo *round-robin* (todos contra todos). El modelo siguiente genera una simulación del torneo.

```
data {
  int<lower=0> J;
  array[2] real<lower=0> epsilon;
}

generated quantities {
  array[J] real<lower=0> lambda;
  array[J, J] int y;
  // Generamos las lambdas
  for (j in 1:J) lambda[j] = gamma_rng(epsilon[1], epsilon[2]);
  // Generamos de la predictiva
  for (i in 1:J) {
    for (j in 1:J) {
      y[i, j] = poisson_rng(lambda[i]) - poisson_rng(lambda[j]);
    }
  }
}
```

Nota que estamos permitiendo algunas cosas sin sentido, pero obviaremos esto. Podríamos ser mas cuidadosos con la combinatoria y sólo permitir los $\binom{J}{2}$ juegos posibles.

```
params.previa ← within(list(),{
  J ← 18
  epsilon ← c(0.5, 0.00001)
})
ppredia ← modelo$sample(params.previa, fixed_param = TRUE,
                        refresh = 0, seed = 10872791)
```

Con la distribución previa definida tenemos las siguientes 20 simulaciones de los partidos entre los dos primeros equipos.

```
        [,1]    [,2]   [,3]    [,4]   [,5]   [,6]   [,7]  [,8]   [,9]   [,10]
[1,]   -45106 -224328 -28857  -32947 106527  33672 -14938 -8644 -14235 -109005
[2,] -202066   20687  -2540 -112032  -6899 105781     84  2742 -56941  -26355
```

Lo cual no tiene mucho sentido. Los partidos usualmente no pasan de tener mas de 10 goles en una liga profesional. El modelo previo que tenemos asigna con alta probabilidad una diferencia de mas de 100 goles. Ver Fig. 3.

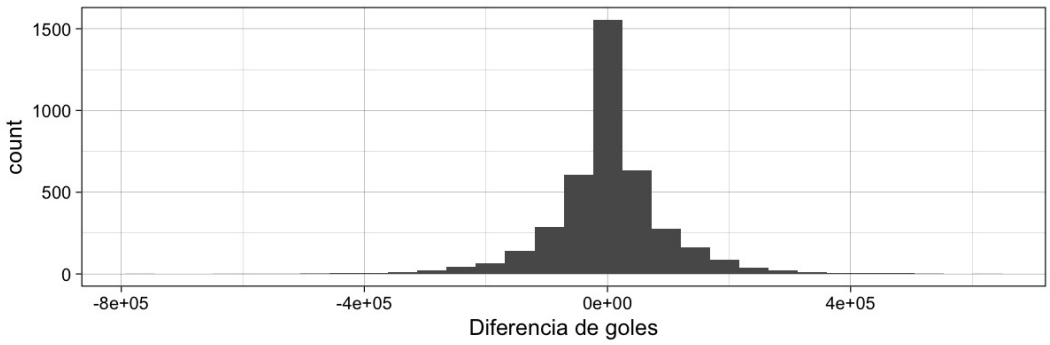

FIGURA 3. *Histograma de la distribución predictiva previa.*

Naturalmente la interpretación no es tan fácil en escenarios con mas parámetros. La distribución predictiva previa nos permite evaluar la incidencia de nuestros supuestos del modelo en cantidades observables.

El **objetivo** *no es* poder replicar los datos con la distribución predictiva previa. Pues esto implicaría ajustar la posterior y convertirla en una distribución previa. El objetivo *es* asegurarnos que nuestra distribución inicial no asigna regiones de alta probabilidad a valores que no tienen sentido en el contexto del problema que estamos modelando. Esto es sumamente relevante cuando tenemos pocas observaciones o cuando los datos no son completamente informativos sobre ciertos parámetros. Por ejemplo, en modelos jerárquicos usualmente los parámetros de escala son mas difíciles de ajustar [4]. Un caso práctico adicional con respecto a un modelo de concentración de contaminantes se puede encontrar en [3]. En esta última referencia el modelo previo asigna una concentración de contaminantes mas densa que un hoyo negro (??).

Para una discusión mas profunda sobre el estado del arte en elicitación y prácticas para definir las distribuciones previas consultar [8].

2.4.1. Tarea: Define una región que creas que sea razonable para observar el promedio de la diferencia de número de goles. Para esto, replica algo parecido a lo que hicimos para los cantantes.

3. MUESTREO PREDICTIVO – POSTERIOR

La distribución predictiva posterior es la distribución sobre nuevas realizaciones que podríamos observar dado que ya hemos actualizado nuestra distribución previa con datos.

La distribución predictiva posterior para datos hipotéticos $\tilde{y}$ condicional en los observados y está definida como

$$\pi(\tilde{y}|y) = \int \pi(\tilde{y}|\theta) \cdot \pi(\theta|y)\, \mathrm{d}\theta\,. \tag{5}$$

3.1. Ejemplo: cantantes

Ajustaremos la distribución posterior y generaremos observaciones hipotéticas bajo la distribución predictiva posterior.

```
data {
  int N;
  real y[N];
  real mu0;
  real<lower=0> n0;
  real<lower=0> a0;
  real<lower=0> b0;
}
parameters {
  real<lower=0> tau;
  real mu;
}
transformed parameters {
  real sigma = 1/tau;
}
model {
  tau ~ gamma(a0, b0);
  mu  ~ normal(mu0, sigma/sqrt(n0));
  y   ~ normal(mu, sigma);
}
generated quantities {
```

```
  array[N] real y_tilde = normal_rng(rep_array(mu, N), rep_array(sigma, N));
}
```

Nota la forma `vectorizada` para generar las simulaciones de un conjunto de datos hipotético del mismo tamaño que el conjunto original.

```
data.list ← within(list(), {
  N ← 42
  y ← cantantes$estatura_cm
  })
posterior ← modelo$sample(append(previa.params, data.list), refresh = 0)
```

3.2. Procesamiento de conjunto de datos ficticios

En las secciones anteriores hemos utilizado un poco de posprocesamiento de las muestras y las réplicas para evaluar estadísticos de interés en nuestro problema. Ahora utilizaremos `Stan` para poder generar dichos resúmenes *dentro* de la simulación.

```
data {
  int N;
  real y[N];
  real mu0;
  real<lower=0> n0;
  real<lower=0> a0;
  real<lower=0> b0;
}
parameters {
  real<lower=0> tau;
  real mu;
}
transformed parameters {
  real sigma = 1/tau;
}
model {
  tau ~ gamma(a0, b0);
  mu  ~ normal(mu0, sigma/sqrt(n0));
  y   ~ normal(mu, sigma);
}
generated quantities {
  array[N] real y_tilde = normal_rng(rep_array(mu, N), rep_array(sigma, N));
  real mean_y_tilde = mean(to_vector(y_tilde));
  real sd_y_tilde = sd(to_vector(y_tilde));
}
```

Mejor aún, podemos utilizar gráficos de `bayesplot` para verificar nuestras simulaciones contra los datos. Ver Fig. 4, Fig. 5 y Fig. 6.

Adicional a esto, podemos hacer nuestras comparaciones gráficas con ciertos estadísticos. Ver Fig. 9, Fig. 10 y Fig. 11.

Incluso podemos graficar comparaciones bivariadas (dos estadísticas el mismo tiempo) como se muestra en Fig. 12.

3.2.1. Tarea: Replica algunos de estos gráficos para los modelos Poisson-Gamma y Binomial-Negativo para los conteos de reclamos atendidos en *twitter* por las aerolíneas.

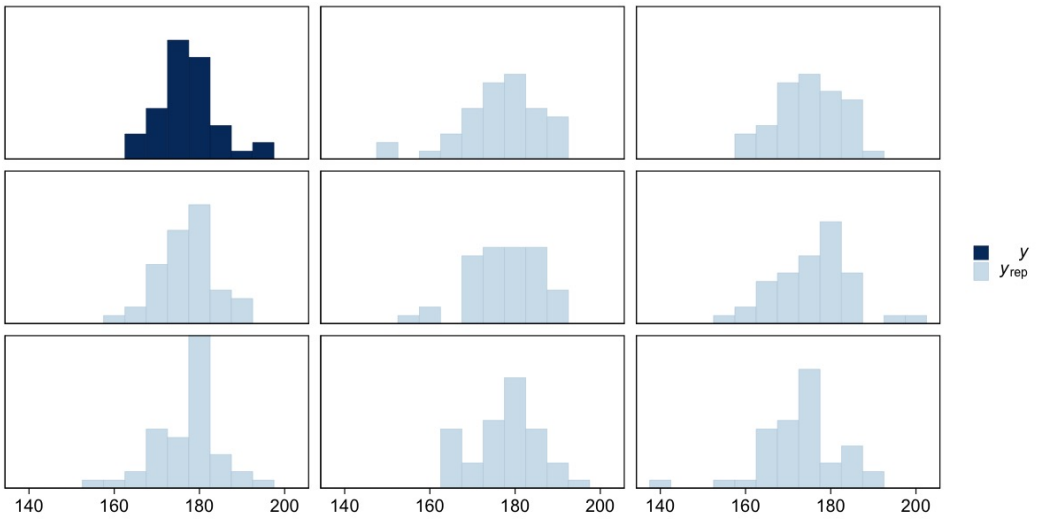

FIGURA 4. *Comparación de histogramas con respecto a los datos y las simulaciones bajo la distirbución predictiva posterior. Utiliza* `ppc_hist`.

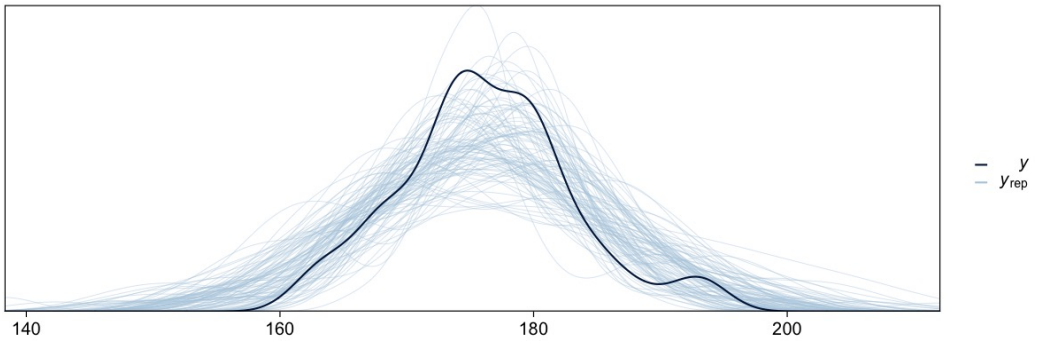

FIGURA 5. *Gráfico espaguetti que compara la densidad de datos ficticios contra observados. Utiliza* `ppc_dens_overlay`.

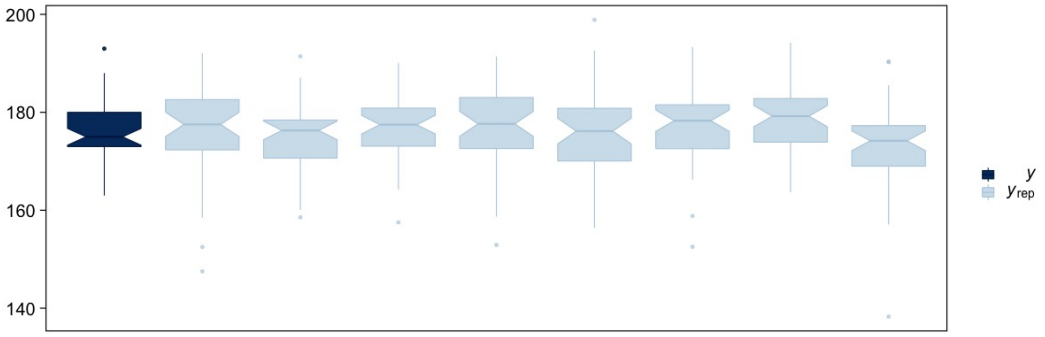

FIGURA 6. *Comparación de* boxplots *entre datos ficticios contra observados. Utiliza* `ppc_boxplot`.

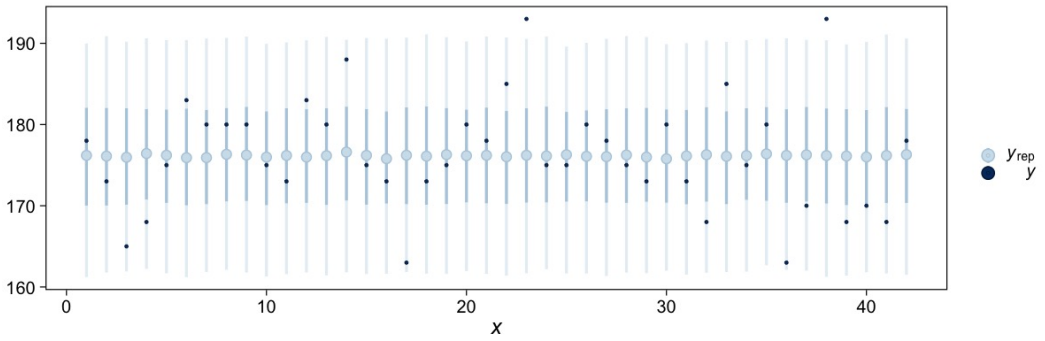

FIGURA 7. *Comparación de* intervalos *entre datos ficticios contra observados. Utiliza* `ppc_intervals`.

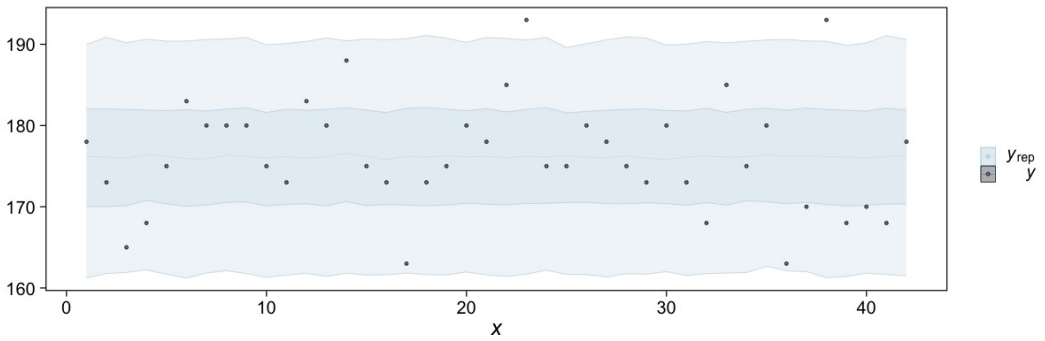

FIGURA 8. *Comparación de bandas entre datos ficticios contra observados. Utiliza* `ppc_ribbons`.

3.3. "Valores p" bayesianos

Si el modelo captura bien los datos, entonces estadísticos basados en tendencias centrales –como media y desviación estándar– deberían de tener valores similares tanto en conjuntos hipotéticos (muestras de la distribución predictiva posterior) como en los datos mismos.

Esto puede ser evaluado por medio de un estadístico que asemeja el concepto frecuentista de valor-p. Es decir, para un estadístico $s(\cdot)$ comparamos los valores de acuerdo a

$$\mathbb{P}\left[s(\tilde{y}) \geq s(y)|y\right] = \int I[s(\tilde{y}) \geq s(y)] \cdot \pi(\tilde{y}|y)\, \mathrm{d}\tilde{y}\,. \tag{6}$$

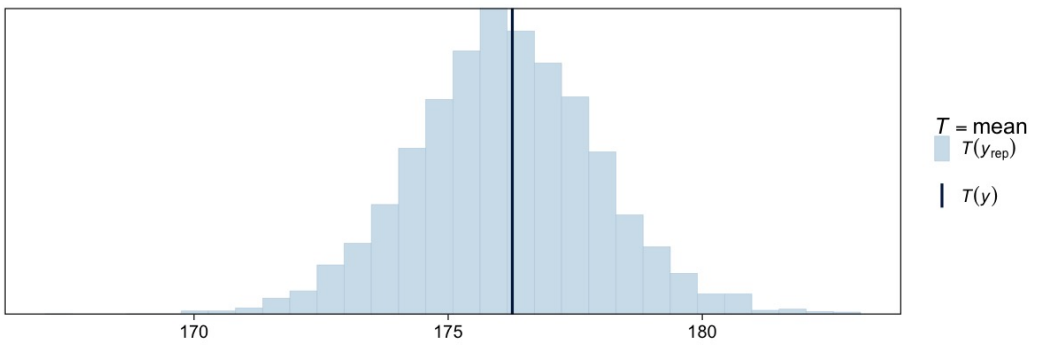

FIGURA 9. *Comparación entre datos ficticios contra observados por medio de medias. Utiliza* `ppc_stat`.

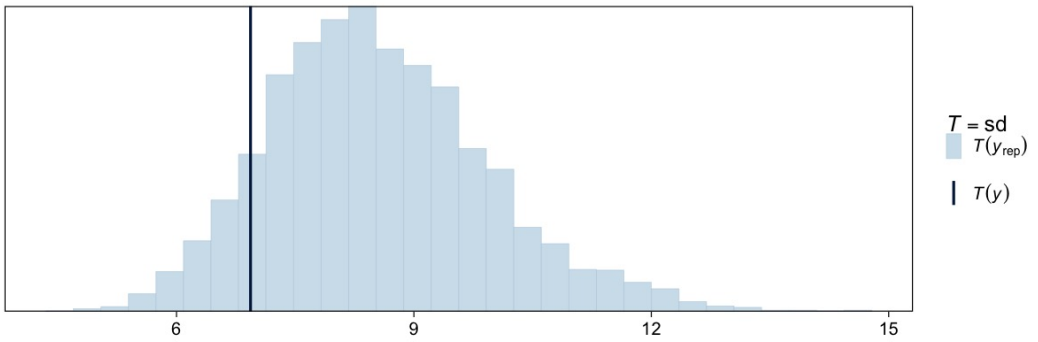

FIGURA 10. *Comparación entre datos ficticios contra observados por medio de desviación estándar. Utiliza* `ppc_stat`.

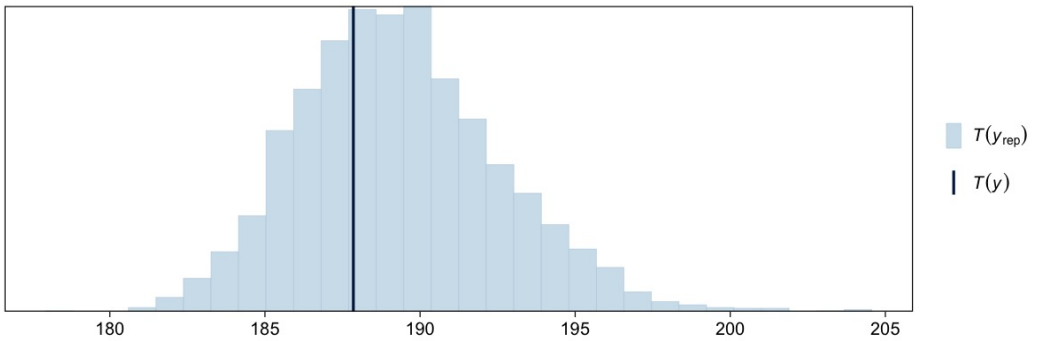

FIGURA 11. *Comparación entre datos ficticios contra observados por medio del percentil* 95 %. *Utiliza* `ppc_stat`.

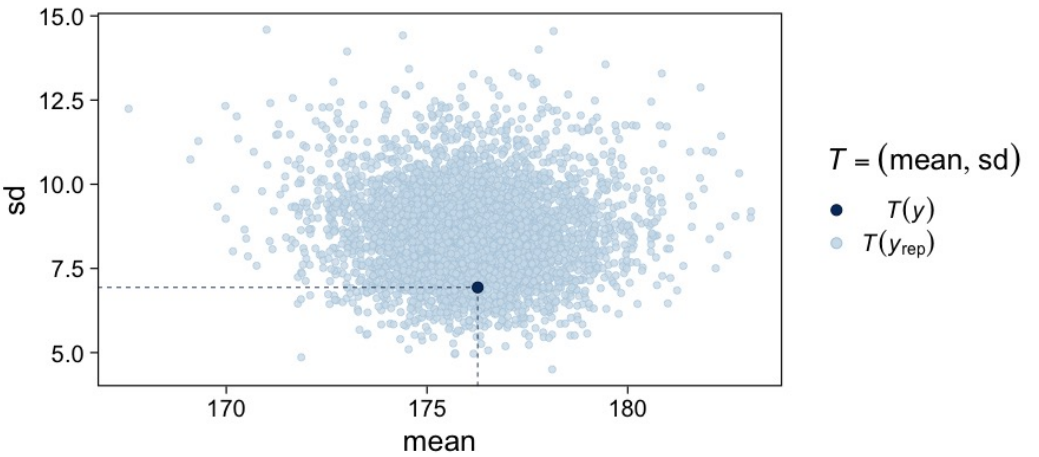

FIGURA 12. *Comparación entre datos ficticios contra observados por medio del percentil* 95 %. *Utiliza* `ppc_stat`.

Este concepto **no** es tal cual un valor-p (en el sentido frecuentista) pues estos estadísticos no están bien calibrados. Es decir, la cobertura nominal **no** corresponde al calculado. En general, no tendrán una distribución uniforme incluso si el modelo está bien especificado [1].

Valores cercanos a 0 ó 1 son motivo de alerta sobre el ajuste del modelo. Por ejemplo, para nuestro modelo de los cantantes de ópera vemos una situación *ideal* utilizando la media. Esto corresponde a que nuestro modelo está capturando bien el comportamiento promedio de las alturas. Sin embargo, para la dispersión nos indica que posiblemente haya problemas con el comportamiento con la dispersión aprendida por el modelo.

```
posterior$draws(variables = c("mean_y_tilde", "sd_y_tilde"), format = "df") ▷
  mutate(indicadora.mean = mean_y_tilde ≥ mean(cantantes$estatura_cm),
         indicadora.sd   = sd_y_tilde ≥ sd(cantantes$estatura_cm)) ▷
  summarise(p.value.mean = mean(indicadora.mean),
            p.value.sd   = mean(indicadora.sd)) ▷
   as.data.frame()
```

```
  p.value.mean p.value.sd
1         0.46       0.89
```

Esto ya lo habíamos graficado antes en Fig. 9, Fig. 10 y Fig. 11.

3.4. Ejemplo: Velocidad de la luz

Los datos provienen de un experimento por Simon Newcomb para medir la velocidad con la que viaja la luz.

> Simon Newcomb set up an experiment in 1882 to measure the speed of light. Newcomb measured the amount of time required for light to travel a distance of 7442 meters. —Gelman et al. [5].

```
data {
  int N;
  real y[N];
}
parameters {
  real<lower=0> sigma;
  real mu;
}
model {
  y   ~ normal(mu, sigma);
}
generated quantities {
  array[N] real y_tilde = normal_rng(rep_array(mu, N), rep_array(sigma, N));
  real mean_y_tilde = mean(to_vector(y_tilde));
  real sd_y_tilde = sd(to_vector(y_tilde));
}
```

```
Running MCMC with 4 sequential chains...

Chain 1 finished in 0.1 seconds.
Chain 2 finished in 0.0 seconds.
Chain 3 finished in 0.1 seconds.
Chain 4 finished in 0.1 seconds.
```

```

All 4 chains finished successfully.
Mean chain execution time: 0.1 seconds.
Total execution time: 0.5 seconds.
```

```
  p.value.mean p.value.sd
1          0.5       0.52
```

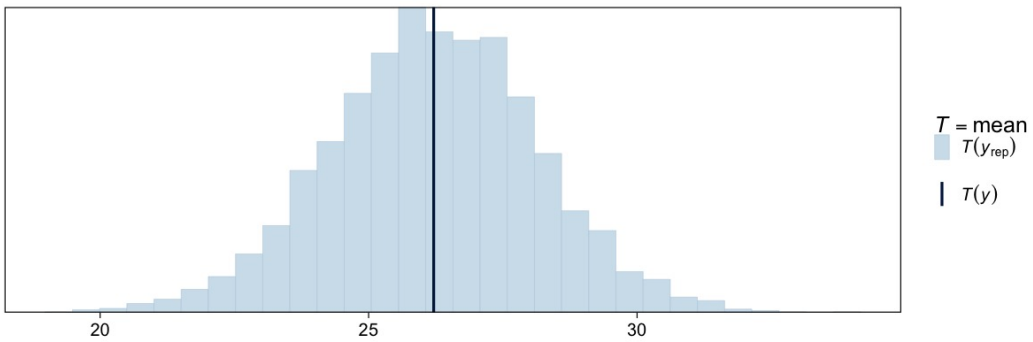

FIGURA 13. *Comparación entre datos ficticios contra observados por medio de medias. Utiliza* `ppc_stat`.

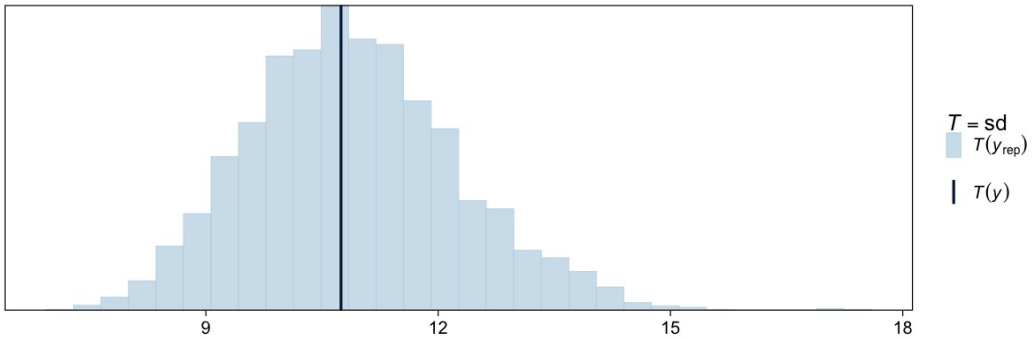

FIGURA 14. *Comparación entre datos ficticios contra observados por medio de desviación estándar. Utiliza* `ppc_stat`.

Los estadístico centrales se ven bien. Sin embargo, si comparamos con respecto al mínimo vemos una historia muy distinta. Lo cual nos indica junto con los gráficos de *lineup* que hay variabilidad en los datos que no es explicada por el modelo.

Una manera de arreglar esta deficiencia del modelo es incorporar un componente adicional que incorpore un proceso de contaminación de observaciones. Tal como es sugerido en [5].

3.5. Observaciones atípicas

En el contexto de los datos del experimentos de la estimación de la velocidad de la luz, se podría sugerir evaluar algún tipo de criterio que permita criticar si una observación es *típica* del modelo ajustado. Esto es, nos interesaría cuantificar si alguna observación tiene una baja probabilidad predictiva. Esto se *puede* lograr utilizando un concepto cercano a `validación cruzada` (que veremos mas adelante) el cual se llama *conditional predictive ordinate* (`CPO`)

$$\mathsf{CPO}_i = \pi(y_i|y_{-i})\,, \tag{7}$$

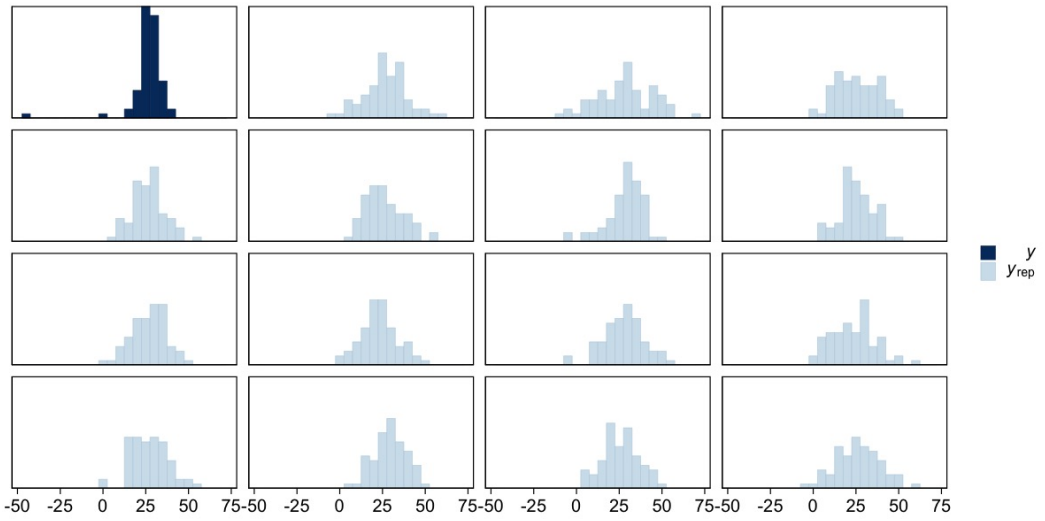

FIGURA 15. *Comparación entre datos ficticios contra observados por medio de* lineup. *Utiliza* `ppc_hist`.

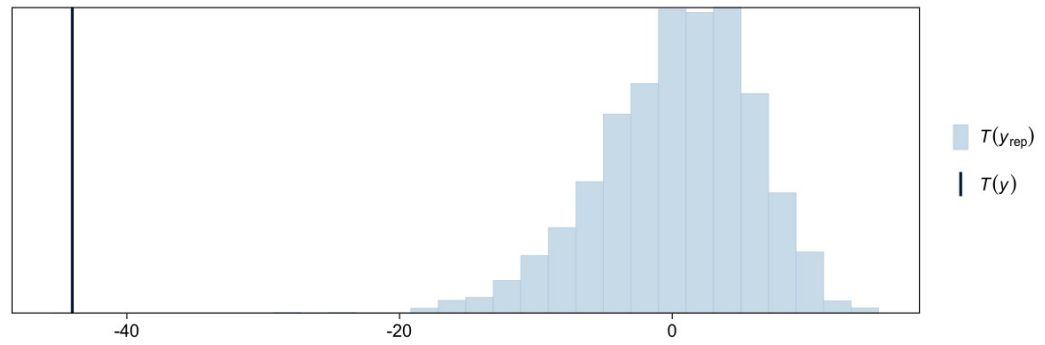

FIGURA 16. *Comparación entre datos ficticios contra observados por medio del mínimo. Utiliza* `ppc_stat`.

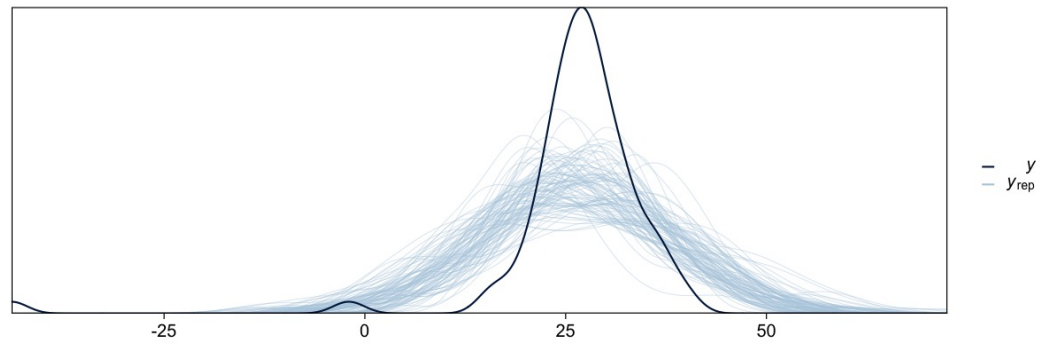

FIGURA 17. *Comparación entre datos ficticios contra observados por densidades. Utiliza* `ppc_dens_overlay`.

donde y_{-i} denota el conjunto de observaciones exceptuando la i -ésima. Este diagnóstico se puede utilizar *barriendo* sobre todas las observaciones buscando tener un resumen bajo todo el conjunto de datos.

Sin embargo, este es un estimador de la evidencia marginal de una observación (un estimador armónico) el cual tiende a tener severos problemas para ser estimado. Incluso en modelos sencillos, puede incurrir en una estimación con alta variabilidad y dar estimaciones sesgadas [7, 9].

El `CPO` es computacionalmente atractivo pues **no** necesita ajustar tantos modelos como observaciones tengamos. Puedes probar que

$$\pi(y_i|y_{-i}) = \frac{\pi(\underline{y}_n)}{\pi(y_{-i})}, \tag{8}$$

donde $\underline{y}_n$ denota la muestra completa. Y este cociente a su vez se puede calcular por medio de

$$\frac{\pi(\underline{y}_n)}{\pi(y_{-i})} = \left[\int \frac{1}{\pi(y_i|\theta)} \cdot \pi(\theta|\underline{y}_n)\, \mathrm{d}\theta\right]^{-1}. \tag{9}$$

Para el cual podemos proponer un estimador Monte Carlo basado en muestras de la posterior $\pi(\theta|\underline{y}_n)$.
Aunque atractivo, computacionalmente hablando, el `CPO` no es recomendable [7]. Pero veremos alternativas que tienen un comportamiento mejor estudiado y para el cual tenemos mejores estimadores.

4. CONCLUSIONES

Las posibilidades para escoger un estadístico resumen son muy extensas. La elección debe ser guiada por la pregunta que se quiere responder por el modelo. Aunque, idealmente, esperaríamos que sean estadísticos pivotales. Es decir, que pongan a prueba el ajuste del modelo.

REFERENCIAS

[1] M. J. Bayarri and J. O. Berger. P Values for Composite Null Models. *Journal of the American Statistical Association*, 95(452):1127–1142, 2000. ISSN 0162-1459. . 10

[2] G. E. P. Box and G. C. Tiao. *Bayesian Inference in Statistical Analysis*. John Wiley & Sons, jan 2011. ISBN 978-1-118-03144-5. 3

[3] J. Gabry, D. Simpson, A. Vehtari, M. Betancourt, and A. Gelman. Visualization in Bayesian workflow. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 182(2):389–402, feb 2019. ISSN 0964-1998, 1467-985X. . 5

[4] A. Gelman and J. Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press, 2006. 5

[5] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian Data Analysis*, volume 2. CRC press Boca Raton, FL, 2014. 1, 10, 11

[6] D. Lunn, C. Jackson, N. Best, A. Thomas, and D. Spiegelhalter. *The BUGS Book: A Practical Introduction to Bayesian Analysis*. CRC Press, oct 2012. ISBN 978-1-4665-8666-6. 3

[7] O. A. Martin, R. Kumar, and J. Lao. *Bayesian Modeling and Computation in Python*. Chapman and Hall/CRC, Boca Raton, first edition, nov 2021. ISBN 978-1-00-301916-9. . 1, 13

[8] P. Mikkola, O. A. Martin, S. Chandramouli, M. Hartmann, O. A. Pla, O. Thomas, H. Pesonen, J. Corander, A. Vehtari, S. Kaski, P.-C. Bürkner, and A. Klami. Prior knowledge elicitation: The past, present, and future. *arXiv:2112.01380 [stat]*, dec 2021. 5

[9] M. A. Newton and A. E. Raftery. Approximate Bayesian Inference with the Weighted Likelihood Bootstrap. *Journal of the Royal Statistical Society. Series B (Methodological)*, 56(1):3–48, 1994. ISSN 0035-9246. 13