Comparación de modelos

Table of contents

0.1 ¿Necesitamos comparar modelos?

Dado un conjunto de modelos, definir cuál es el mejor depende fuertemente del contexto de aplicación. En algunos casos, puede que ni siquiera se necesite formular más de un modelo, y en otros, la comparación misma puede ser parte del resultado.

Dada la necesidad de comparar modelos, más allá de los criterios particulares de cada contexto, en general se busca que los modelos puedan representar bien aquellos datos que no hayan sido utilizados para el ajuste. Esto es necesario para evitar el sobreajuste.

0.2 Generalidades

Así como la estimación de parámetros puede basarse en minimizar funciones de costo (error entre lo predicho y lo observado) o maximizando la probabilidad de los datos (i.e., máxima verosimilitud), la comparación de modelos puede basarse en minimizar el error sobre datos de prueba (no usados para la estimación) o maximizar la probabilidad de esos datos de prueba bajo el modelo estimado.

Las métricas más sencillas para comparar modelos se basan en el cálculo de alguna diferencia (o función de costo) entre lo predicho por el modelo y lo

observado en datos independientes.

Como obtener datos independientes a veces no es factible, o implica reducir el N destinado al ajuste, se han desarrollado métricas para intentar aproximar ese error o probabilidad. Los criterios de información siguen ese espíritu (e.g., AIC, BIC, DIC, WAIC).

0.3 La (log) densidad predicitiva esperada sobre nuevos datos

Buscamos un modelo bajo el cual nuevos conjuntos de datos tomados bajo los mismos valores de las variables predictoras resulten probables según el modelo. Esta probabilidad se define como la densidad predictiva posterior por punto esperada para un nuevo conjunto de datos (*expected log pointwise predictive density, elpd*):

$$\text{elpd} = \sum_{i=1}^{N} \int p_t(\tilde{y}_i) \ \log[p(\tilde{y}_i \mid y)] \ \mathrm{d}\tilde{y}_i,$$

donde $p_t(\tilde{y}_i)$ es la verdadera distribución que genera los datos (desconocida e imposible de conocer; con t de *true*) y $p(\tilde{y}_i \mid y)$ es la distribución predictiva posterior para la observación i (usando los correspondientes valores de las predictoras), condicionada en los datos observados y.

0.4 Leave-One-Out Cross Validation

Como no conocemos $p_t(\tilde{y}_i)$, la estimamos usando *Leave-One-Out Cross Validation* (LOO-CV), para obtener el estimador

$$\widehat{\text{elpd}}_{\text{loo}} = \sum_{i=1}^{N} \log \int p(y_i \mid \theta) \ p(y_i \mid y_{-i}) \ \mathrm{d}y_i,$$

donde $p(y_i \mid \theta)$ es la verosimilitud de la observación y_i y $p(y_i \mid y_{-i})$ es la distribución predictiva posterior de y_i con los parámetros θ estimados excluyendo dicha observación.

0.5 Leave-One-Out Cross Validation

En la práctica, si trabajamos con muestras de la distribución posterior, θ^s (con s indexando las muestras), la integral de arriba se aproxima mediante el promedio:

$$\widehat{\text{elpd}}_{\text{loo}} = \sum_{i=1}^{N} \log \left(\frac{1}{S} \sum_{s=1}^{S} p(y_i \mid \theta^{-i,s}) \right),$$

donde $\theta^{-i,s}$ es la muestra s de la posterior estimada excluyendo la observación i.

Pero esto implica estimar la posterior N veces, lo cual puede ser extremadamente costoso.

0.6 Aproximación de N LOO-posteriores mediante muestreo por importancia

Si asumimos que las observaciones son condicionalmente independientes, la verosimilitud es factorizable (producto de las verosimilitudes por observación). Esto permite aproximar las N posteriores leave-one-out ponderando las muestras que ya tenemos de la posterior completa (i.e., la que se estimó con todas las observaciones), lo que se llama muestreo por importancia.

Para obtener una muestra de la posterior que excluye el la observación i, se requiere ponderar las muestras de la posterior completa con los siguientes pesos no normalizados:

$$r_i^s = \frac{1}{p(y_i \mid \theta^s)},$$

con $s \in \{1, ...S\}$ indexando muestras de la posterior completa.

0.7 Pareto Smoothed Importance Sampling - LOO

Sin embargo, estos pesos crudos pueden ser muy inestables si la observación i resulta extraña para el modelo. Esto lleva a que la aproximación de la posterior LOO de la observación i basada en muestreo por importancia no sea confiable.

Para resolver este problema, Vehtari et al. (2017) implementan el *Pareto Smoothed Importance Sampling* (PSIS-LOO), donde los pesos r_i^s son suavizados utilizando los cuantiles de una distribución de Pareto ajustada a los pesos r_i^s. A la vez que vuelve los pesos más confiables para estimar la posterior LOO, con este método se obtiene un diagnóstico para la confiabilidad de los pesos, el parámetro k de la distribución de Pareto. Cuando este valor es mayor a 0.7, se recomienda muestrear directamente la posterior LOO correspondiente a la observación i, no aproximarla mediante muestreo por importancia.

Llamando w_i^s a los pesos suavizados, podemos aproximar la densidad predictiva posterior por punto de la siguiente manera:

$$\widehat{\text{elpd}}_{\text{psis-loo}} = \sum_{i=1}^{N} \log \left(\frac{\sum_{s=1}^{S} w_i^s \; p(y_i \mid \theta^s)}{\sum_{s=1}^{S} w_i^s} \right),$$

donde S es el total de muestras tomadas de la posterior completa.

0.8 Validación Cruzada en K particiones (*K-fold CV*)

Si el diagnóstico k es alto para muchas observaciones, y muestrear las posteriores LOO esa cantidad de veces no es factible, podemos recurrir a una validación cruzada en K particiones de los datos, de la siguiente manera

Dividimos el conjunto de datos en K partes iguales (o similares), con $K \approx 10$ siendo un valor sensato. Ajustamos el modelo K veces, cada vez dejando fuera del ajuste el subconjunto k. En cada vez que un subconjunto de datos se deja afuera, calculamos la densidad predictiva posterior por punto de las observadciones pertenecientes a ese grupo:

$$\widehat{\text{elpd}}_{\text{xval}} = \sum_{i=1}^{N} \log \left(\frac{1}{S} \sum_{s=1}^{S} p(y_i \mid \theta^{-k,s}) \right),$$

donde $\theta^{-k,s}$ es la muestra s de la posterior estimada quitando el grupo de observaciones k, en el cual está incluida la observación i.

Otro criterio de información fácil de calcular es el WAIC (Watanabe Information Criterion, o Widely Applicable Information Criterion). Sin embargo, cuando PSIS-LOO no es confiable (altos valores del parámetro k), tampoco lo es WAIC. Por lo tanto, se recomienda intentar primero con PSIS-LOO, y si no es confiable, realizar LOO-CV exacto o K-fold CV.

Detalles en

Vehtari, A., Gelman, A., & Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. Statistics and computing, 27, 1413-1432.

En resumen, para comparar modelos estimamos la densidad de hipotéticas nuevas observaciones según nuestro modelo. Los modelos con mayor densidad son preferidos.

Para estimar esta densidad, podemos hacer validación cruzada, ya sea LOO o K-fold, lo cual implica ajustar el modelo dejando 1 o varias observaciones afuera

y calcular la verosimilitud de las observaciones dejadas afuera. Idealmente, esto se puede eviar usando importance sampling.

La aproximación de PSIS-LOO está implementada en el paquete `loo` de `R`.

0.9 Comparación de modelos en la práctica

Para comparar modelos necesitamos calcular la matriz de N observaciones $\times S$ muestras con la log verosimilitud. Si la calculamos en `Stan` (en `generated quantities`), es conveniente llamarla `log_lik`.

```
data {
  int N;
  array[N] int y; // Número de incendios
  vector[N] x; // FWI
}

// Parámetros a muestrear
parameters {
  real alpha;
  real beta;
}

// Calculamos las cantidades derivadas
transformed parameters {
  vector[N] lambda = exp(alpha + beta * x);
}

// Acá definimos la log densidad posterior (o la que sea)
model {
  // Densidad previa
  alpha ~ normal(0, 1);
  beta ~ normal(0, 0.1);

  // Verosimilitud
  y ~ poisson(lambda);
}

generated quantities {
  vector[N] log_lik;
  for (i in 1:N) {
    log_lik[i] = poisson_lpmf(y[i] | lambda[i]);
  }
}
```

0.10 Ejemplo: modelos de fuego

Comparamos dos modelos para el número de incendios por verano. Supongamos que queremos un modelo predictivo. Podemos usar un modelo univariado, con el FWI, o incluir más predictoras (precipitación, temperatura, VDP). Pero incluir más predictoras puede generar sobreajuste, ya que tenemos pocos datos. Entonces, comparamos la capacidad predictiva de un modelo simple y uno más complejo.

Ver código `<modelos/nfuegos_comparacion.R>`