TEST ERROR RATE – TRAINING ERROR TEST

Supongamos que tenemos una muestra $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ de determinado vector (X, Y) donde $X \in \mathbb{R}^p$, $Y \in \{1, 2, 3, ..., k\}$.

A partir de $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ construimos con algún método estadístico (por ejemplo naive Bayes, KNN, LDA,QDA, SVM, logística o el que sea) un clasificador que le podemos llamar $\widehat{f}_n$.

La función $\widehat{f}_n$ depende de la muestra $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ y nos sirve para clasificar un determinado valor $x \in \mathbb{R}^p$, es decir que para cada x nos predecirá su etiqueta.

Formalmente tenemos que $\widehat{f}_n = \widehat{f}_n((x_1, y_1), (x_2, y_2), ..., (x_n, y_n), x)$ que vamos a abreviar como $\widehat{y} = \widehat{f}_n(x) \in \{1, 2, 3, ..., k\}$.

¿Qué tan bueno será nuestro clasificador $\widehat{f}_n$?

1. Podemos preguntarnos qué tan bien funciona nuestro clasificador $\widehat{f}_n$ en los puntos observados. Es decir que podemos llamarle $\widehat{y}_i = \widehat{f}_n(x_i)$ y ver si tiene la misma etiqueta que la observada para x_i (es decir y_i). En ese caso calculamos lo que se llama el "training error rate" definido como el porcentaje de valores mal clasificados en la muestra, es decir

$$\frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{\{\widehat{y}_i \neq y_i\}} = \frac{cantidaddeobservacionesmalclasificadas}{n}.$$

2. Podemos preguntarnos qué tan bien funciona nuestro clasificador $\widehat{f}_n$ en puntos distintos a los observados. En ese caso debemos considerar otros puntos, digamos $(x'_1, y'_1), (x'_2, y'_2), ..., (x'_m, y'_m)$ calculamos los $\widehat{y'}_i = \widehat{f}_n(x'_i)$ y consideramos lo que se llama "test error rate" definido naturalmente como

$$\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}_{\{\widehat{y'}_i \neq y'_i\}} = \frac{cantidaddeobservacionesmalclasificadas(entrelosmnuevospuntos)}{m}.$$

OBSERVACIONES Y COMENTARIOS

1. Un buen clasificador debería ser aquel que minimice el test error rate (nos interesa clasificar lo mejor posible otros puntos que no sean los observados, a los observados ya les conocemos su etiqueta).

2. En la práctica, si construimos el clasificador con los n datos de la muestra, no podremos conocer los y'_j para otros puntos x'_j por lo que podremos construir el clasificador $\widehat{f}_n$ pero no sabremos qué tan bien funciona en términos de su test error rate. Lo que se hace en la práctica es (si n es grande, utilizar una parte de la muestra para construir el clasificador y con la otra calculamos el test error rate).

3. Formalizando el punto anterior, supongamos que la muestra es

$$(x_1, y_1), (x_2, y_2), ..., (x_n, y_n), (x_{n+1}, y_{n+1}), (x_{n+2}, y_{n+2}), ..., (x_{n+m}, y_{n+m})$$

, entonces construimos el clasificador con las primeras n observaciones $\widehat{f_n}$ y luego utilizamos $(x_{n+1}, y_{n+1}), (x_{n+2}, y_{n+2}), ..., (x_{n+m}, y_{n+m})$ en lugar de $(x'_1, y'_1), (x'_2, y'_2), ..., (x'_m, y'_m)$ con los que calculamos el test error rate.

4. Cuando partimos la muestra en dos, a la primer parte de la muestra $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ se le llama muestra de entrenamiento (training) y a la segunda parte $(x_{n+1}, y_{n+1}), (x_{n+2}, y_{n+2}), ..., (x_{n+m}, y_{n+m})$ se le suele llamar validación ya que elegiremos n y m de modo que la estimación del test error rate sea razonablemente bajo.

5. El test error rate de un conjunto de m observaciones, es un estimador de $P(F(X) \neq Y)$ que sabemos que es minimizada cuando consideramos el clasificador de Bayes.

LEAVE ONE OUT CROSS VALIDATION (LOOCV)

La idea intuitiva de separar la muestra en dos conjuntos, el de entrenamiento y el de validación tiene dos problemas.

1. La variabilidad de la estimación del test error rate puede ser alta en función de los valores que tomemos de n y m y también en función de cuáles observaciones tomamos para entrenar y cuáles para validar.

2. Para ajustar razonablemente necesitamos un n grande, pero para estimar razonablemente bien la probabilidad del error (test error rate) necesitamos un m grande por lo que tanto para ajustar con las n observaciones como para estimar el error con las otras m observaciones, estamos "desperdiciando" una buena cantidad de observaciones en ambos casos.

La validación cruzada "dejando uno afuera" intenta tener esto en cuenta y consiste en dejar sólo un dato afuera, entrenar con todos los demás y usar ese único dato como conjunto de validación, y eso repetirlo quitando cada uno de los datos afuera y validando con el mismo. De modo que si tenemos n observaciones $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ quitamos (x_1, y_1), entrenamos con $(x_2, y_2), ..., (x_n, y_n)$ (es decir que construimos el clasificador con $(x_2, y_2), ..., (x_n, y_n)$) y luego vemos si con este clasificador la etiqueta y_1 es correctamente obtenida por nuestro clasificador (es decir vemos si se cumple que $\widehat{y}_1 = y_1$ o no). Este procedimiento lo repetimos n veces quitando cada una de las observaciones y viendo si su etiqueta es correctamente clasificada por nuestro clasificador.

Si para cada $i = 1, 2, ..., n$, le llamamos $f_n^{(i)}$ al clasificador utilizando todas las observaciones salvo la $i-$ésima $((x_i, y_i))$ y le llamamos $\widehat{y}_i = f_n^{(i)}(x_i)$. De esta forma construimos nuestro estimador del error de clasificación mediante

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{\{\widehat{y}_i\neq y_i\}}=\frac{cantidaddeobservacionesmalclasificadas}{n}.$$

Notemos que de esta forma, si tenemos n observaciones estamos estimando el test error rate utilizando n predicciones de etiquetas y estamos entrenando n veces con $n-1$ datos cada vez.

NOTAS Y COMENTARIOS

1. Por construcción este método de LOOCV tiene en cuenta (y por lo tanto resuelve en algún sentido) los dos problemas anteriores planteados sobre con cuántos y con cuales entrenar y con cuántos y cuales validar.

2. Se llama también cross validation si por ejemplo en lugar de quitar de a una observación, quitamos de a dos o más, simplemente que no es leave one out cross validation (LOOCV).

3. Esta idea de cross validation, es muy útil en distintos contextos y suele ser útil cuando tenemos un método como KNN para la elección de un valor adecuado de K. La idea es la siguiente: dado el conjunto de observaciones $(x_1,y_1),(x_2,y_2),...,(x_n,y_n)$ utilizamos KNN con $K=1$ y calculamos por LOOCV el error de predicción $E_1=\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{\{\widehat{y}_i\neq y_i\}}$, luego utilizamos KNN con $K=2$ y calculamos por LOOCV el error de predicción $E_2=\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{\{\widehat{y}_i\neq y_i\}}$ y así sucesivamente con $K=1,2,3,...,H$ y por lo tanto elegimos el K óptimo tomando aquel K donde se minimice E_K. Es decir que elegimos
$$\widehat{K}=\arg\min_{K\in\{1,2,...,H\}}E_K.$$

4. Ideas como en el punto anterior se utilizan con muy buenos resultados a otros casos donde se tiene tiene un método estadístico para abordar determinado problema (puede ser tanto de estimación como de modelación) que dependa de algún parámetro que haya que definir de antemano.

5. El concepto de validación cruzada fue introducido por Meryvn Stone "Cross-validatory assessment in statistical prediction" (with discussion) en Journal of the Royal Society, B y "Cross-validation and multinomial predictions" en Biometrika, ambos en 1974. Previamente existieron algunas ideas planteadas en Lunts y Brailovsky (1967) "Evaluation of attributes obtained in statistical decision rules" en Engineering Cibernetics y en Allen "The relationship between variable selection and data augmentation and a method for prediction" en Technometrics en 1974.