



Métricas de evaluación en clasificación



Métricas de evaluación

UNIVERSIDAD DE CÓRDOBA

Introducción

Como se introdujo en la lección anterior, tras entrenar un clasificador, tendremos que evaluarlo en base a ciertas métricas que consideremos oportunas en cada problema concreto. El objetivo detrás de esta evaluación es poder conocer una aproximación del rendimiento de nuestro modelo a la hora de aplicarlo en producción sobre nuevos datos no conocidos. En esta lección, vamos a introducir distintas métricas y técnicas utilizadas para la evaluación de clasificadores.

Matriz de confusión

Dado un clasificador binario y la predicción de varias instancias de los datos, las predicciones pueden considerarse correctas o incorrectas, acorde a la matriz de confusión que se observa en la Tabla 1. Las predicciones pueden encuadrarse dentro de uno de los siguientes tipos:

- Verdadero positivo (*True Positive*, TP). La instancia se predice como positiva y su clase real era positiva. Acierto.
- Verdadero negativo (*True Negative*, TN). La instancia se predice como negativa y su clase real era negativa. Acierto.
- Falso positivo (*False Positive*, FP). La instancia se predice como positiva, mientras que su clase real era negativa. Error.
- Falso negativo (*False Negative*, FN). La instancia se predice como negativa, mientras que su clase real era positiva. Error.

Tabla 1. Matriz de confusión binaria.

		Predicha	
		Positiva	Negativa
Real	Positiva	TP	FN
	Negativa	FP	TN

En base a esta matriz, se pueden calcular multitud de métricas de evaluación del rendimiento del clasificador. En el escenario de clasificación multi-clase, la matriz se extiende para cubrir todas las posibles clases, como se puede observar en la Tabla 2. Como en el caso anterior, la diagonal principal incluye los datos correctamente clasificados, mientras que el resto contiene los errores cometidos.

Tabla 2. Matriz de confusión para un problema con k clases.

	Predicha			
	$C1$	$C2$	\dots	Ck
Real	$C1$	$P_{1,1}$	$P_{1,2}$	$P_{1,k}$
	$C2$	$P_{2,1}$	$P_{2,2}$	$P_{2,k}$
	\dots			
	Cn	$P_{n,1}$	$P_{n,2}$	$P_{n,k}$

Métricas de evaluación

3.1. Clasificadores discretos

En primer lugar, vamos a describir varias métricas de evaluación para clasificadores que proporcionan una salida discreta, es decir, cuya salida es la clase a la que pertenece un patrón concreto. Para ello, las métricas se basan en la matriz de confusión. Se indicará junto a la métrica con \uparrow o \downarrow si la métrica es a maximizar (mejor cuanto mayor sea su valor) o a minimizar (mejor cuanto menor sea su valor).

La métrica más simple para conocer la bondad de nuestro clasificador es el porcentaje de patrones correctamente clasificados, conocido como accuracy. Para calcularla, habrá que sumar el número de instancias correctamente clasificadas, divididas entre el número total de instancias.

$$\uparrow accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

El accuracy es una métrica muy general y que puede darnos una idea del rendimiento del clasificador, pero en ciertos casos (donde las clases están muy desbalanceadas, una de las clases es más crítica, etc.), otras métricas de evaluación son útiles.

El recall, también llamado ratio de verdaderos positivos (TPR), o sensibilidad (sensitivity), es el número de verdaderos positivos comparado con el número de instancias realmente positivas. Por otro lado, precision se calcula como el número de verdaderos positivos entre el número total de instancias predichas como positivas.

$$\uparrow recall = \frac{TP}{TP + FN}$$

$$\uparrow precision = \frac{TP}{TP + FP}$$

Basada en ambas métricas, el *FMeasure* (también llamado *F1-score*), es la media armónica de *precisión* y *recall*. De este modo, muestra una combinación de ambas métricas, que son contrapuestas (por lo general, al optimizar y por tanto aumentar una de ellas, suele disminuir la otra).

$$\uparrow FMeasure = 2 \frac{precision * recall}{precision + recall}$$

Se considera *specificity*, también llamado especificidad o ratio de verdaderos negativos, al número de instancias negativas correctamente clasificadas, entre el número total de patrones realmente negativos. Se llama ratio de falsos positivos, o *FPR*, al número de instancias incorrectamente predichas como positivas, de entre el total de instancias negativas.

$$\uparrow specificity = \frac{TN}{TN + FP}$$

$$\downarrow FPR = \frac{FP}{TN + FP}$$

Todas estas métricas se definen para clasificación binaria, pero pueden extenderse fácilmente para clasificación multi-clase. El enfoque seguido comúnmente es calcular la métrica para cada clase, y calcular el promedio entre todas ellas. Por ejemplo, dada la matriz de la Tabla 2, la suma de los valores de una columna sería el denominador para la *precision* ($TP + FP$); mientras que la suma de los valores de una fila serviría para calcular el *recall* ($TP + FN$). Nótese que *accuracy*, que es la suma de los valores de la diagonal principal entre el total de instancias, puede calcularse directamente.



Además de las métricas descritas, en el escenario multi-clase cabe considerar el coeficiente *Kappa de Cohen* (κ). Esta métrica pretende compensar las clasificaciones que puedan deberse al azar, y se define como se observa a continuación, donde $M_{i.}$ es la suma de elementos de la i -ésima columna de la matriz de confusión, $M_{.i}$ la suma de elementos en la i -ésima fila, y n el número total de patrones en evaluación.


$$\uparrow \kappa = \frac{n \sum_{i=1}^k M_{ii} - \sum_{i=1}^k M_{i \cdot} \cdot M_{\cdot i}}{n^2 - \sum_{i=1}^k M_{i \cdot} \cdot M_{\cdot i}}$$

3.1.1. Coste de los errores

No en todos los problemas, el coste de cometer un error de un tipo (FP) u otro (FN) es el mismo. A continuación, se muestran dos ejemplos de ambos casos.

En primer lugar, consideremos un hospital, donde se trata de predecir si un paciente sufrirá o no una enfermedad de manera severa, como puede ser el Covid-19. En este caso, el error cometido por predecir a un paciente como que tendrá dicha enfermedad severa cuando realmente no será así (falso positivo), es mucho menos importante que aquellos cometidos al predecir que un paciente no tendrá la enfermedad severa, y que luego sí la sufra sin recibir tratamiento. En la Figura 1 se muestra un ejemplo de este caso.

Case 1
COVID 19 = 1 
Healthy = 0 



Cost of FN > Cost of FP

Actual









	Diagnosed COVID 19 (1)	Diagnosed Healthy (0)
Predict COVID 19 (1)	  TP	  FP Healthy predicted as sick
Predict Healthy (0)	  FN Sick predicted as healthy	  TN

Figura 1. Ejemplo donde el coste de los FN es mayor al coste de los FP.

Por otro lado, imaginemos un sistema de detección de correo basura o spam. En este caso, el coste de los falsos positivos (un correo que no es spam, detectado como tal, y que por tanto podemos perder información importante) es mucho mayor que los falsos negativos (no detectar un correo que realmente es spam, lo recibiremos y simplemente tendremos que borrarlo). Se ilustra el ejemplo en la Figura 2.

Case 2

Spam = 1

Not Spam = 0



Cost of FP > Cost of FN

Actual

Not spam predicted as spam

Predict

	Spam (1)	Not Spam (0)
Spam (1)	TP 	FP
Not Spam (0)	FN 	TN

Spam predicted as not spam

Figura 2. Ejemplo donde el coste de los FP es mayor al de los FN.

Mientras que en el primer caso, utilizar la métrica *recall* puede ser más interesante (incluye los FN en el denominador), en el segundo, es más interesante utilizar *precisión* (donde los FP están en el denominador).

3.2. Salidas probabilísticas

Pese a que todos los clasificadores deben tener la capacidad de proporcionar una salida discreta (es decir, la clase predicha), algunos de ellos también proporcionan una salida probabilística. Esta salida probabilística indica la probabilidad con que el patrón concreto pertenecería a cada una de las clases existentes.

Uno de los análisis más comunes en este caso es el análisis ROC (*Receiver Operating Characteristics*). El espacio ROC es un espacio bidimensional con los falsos positivos en el eje horizontal, y los verdaderos positivos en el eje vertical. Un punto en dicho espacio representa un clasificador con un valor x de ratio de falsos positivos, y un valor y de ratio de verdaderos positivos. Por tanto, el punto (1,0) representaría el peor clasificador posible, y el (0,1) el mejor posible. Se considera que un clasificador es mejor que otro si se sitúa más al noreste que otro en el espacio ROC. En la Figura 3 se muestra un ejemplo del espacio ROC.

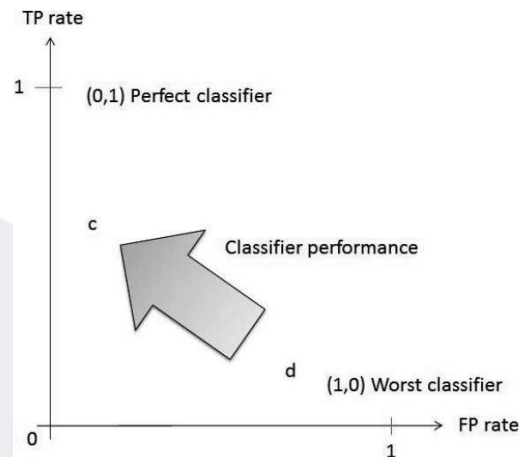


Figura 3. Espacio ROC. La flecha indica el sentido en que los clasificadores son considerados mejores. Fuente: [Agg15].

Los clasificadores probabilísticos necesitan un umbral para poder dar una decisión de la clase final. Para cada posible umbral, se obtendría un clasificador discreto distinto, con un valor distinto para los ratios de TP y FP. Considerando todos los posibles umbrales, y uniendo los puntos resultantes en el espacio ROC, se obtiene la curva ROC. Se observa en la Figura 4 un ejemplo con varias curvas ROC. La diagonal muestra la curva ROC que se espera de un clasificador aleatorio (en un escenario binario), donde la mitad de las predicciones son correctas. Cualquier curva por debajo de dicha diagonal sería un clasificador a desechar. Un clasificador ideal o perfecto tendría la forma que presenta en la figura como “*perfect classifier*”. El resto de curvas, se consideran mejor cuanto más cercanas a dicho clasificador perfecto; o lo que es lo mismo, se considera un clasificador mejor que otro cuando el área existente bajo su curva ROC es mayor.

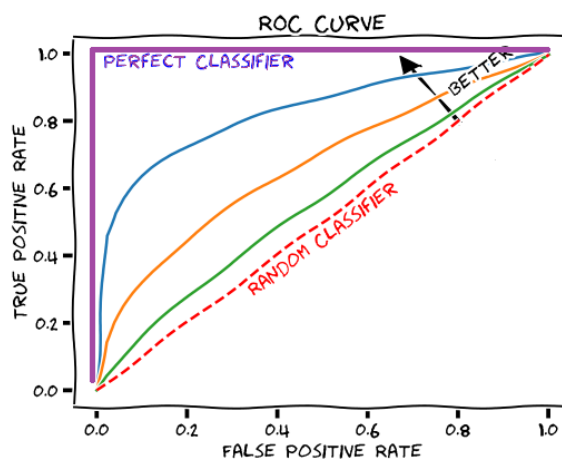


Figura 4. Ejemplos de curva ROC¹.

¹ Fuente: <https://glassboxmedicine.com/2019/02/23/measuring-performance-auc-auroc/>

Ejemplo práctico

4.1. Clasificador binario

Consideremos un clasificador binario, entrenado para clasificar patrones en las clases P y N . Muestre la Tabla 3 las predicciones ($\Omega(i)$) y la clase real de varias instancias en un problema de clasificación binaria. A continuación, la Tabla 4 muestra la matriz de confusión derivada de dichas predicciones.

Tabla 3. Predicciones de un clasificador Ω de ejemplo para un problema binario.

	$i1$	$i2$	$i3$	$i4$	$i5$	$i6$	$i7$	$i8$	$i9$	$i10$
Real	P	P	N	P	N	P	P	N	P	N
$\Omega(i)$	P	P	P	P	N	P	N	N	P	P

Tabla 4. Matriz de confusión para el ejemplo práctico de clasificación binaria.

		Predicha	
		P	N
Real	P	5	1
	N	2	2

A continuación, se calculan algunas de las métricas de evaluación para este problema:

$$accuracy = \frac{5 + 2}{10} = 0.7 \equiv 70\%$$

$$recall = \frac{5}{5 + 1} = 0.833$$

$$precision = \frac{5}{5 + 2} = 0.714$$

$$FMeasure = 2 \frac{0.833 * 0.714}{(0.833 + 0.714)} = 0.769$$

$$specificity = \frac{2}{2 + 2} = 0.5$$

4.2. Clasificador multi-clase

Consideremos un clasificador multi-clase, entrenado para clasificar patrones en las clases A , B , y C . Muestre la Tabla 5 las predicciones ($\Omega(i)$) y la clase real de varias instancias en un problema de clasificación binaria. A continuación, la Tabla 6 muestra la matriz de confusión global derivada de dichas predicciones.

Tabla 5. Predicciones de un clasificador Ω de ejemplo para un problema multi-clase.

	$i1$	$i2$	$i3$	$i4$	$i5$	$i6$	$i7$	$i8$	$i9$	$i10$
Real	A	B	C	A	B	B	B	A	C	A
$\Omega(i)$	B	B	A	A	B	B	A	C	B	A

Tabla 6. Matriz de confusión para el ejemplo práctico de clasificación multi-clase.

		Predicha		
		A	B	C
Real	A	2	1	1
	B	1	3	1
	C	1	0	0

A continuación, se muestran algunas de las métricas para este ejemplo.

$$accuracy = \frac{2 + 3 + 0}{10} = 0.5 \equiv 50\%$$

$$recall = \left[\left(\frac{2}{4} \right) + \left(\frac{3}{4} \right) + \left(\frac{0}{2} \right) \right] / 3 = 0.417$$

$$precision = \left[\left(\frac{2}{5} \right) + \left(\frac{3}{5} \right) + \left(\frac{0}{1} \right) \right] / 3 = 0.367$$

$$\kappa = \frac{10 * (2 + 3 + 0) - (4 * 4 + 4 * 5 + 2 * 1)}{100 - (4 * 4 + 4 * 5 + 2 * 1)} = \frac{50 - 38}{62} = 0.194$$

4.3. Clasificador probabilístico

Consideremos un problema de clasificación binario, y un clasificador capaz de ofrecer una salida probabilística de pertenencia a una clase. Muestre la Tabla 7 las predicciones ($\Omega(i)$) y la clase real de varias instancias en un problema de clasificación binaria.

Tabla 7. Predicciones de un clasificador probabilístico Ω de ejemplo para un problema de clasificación binario.

	$i1$	$i2$	$i3$	$i4$	$i5$	$i6$	$i7$	$i8$	$i9$	$i10$
Real	P	N	N	P	N	N	P	P	N	P
$\Omega(i)$	0.27	0.12	0.67	0.66	0.73	0.08	0.94	0.51	0.49	0.94

De cara a producir una clasificación discreta, este clasificador necesita un umbral t que determine que si $\Omega(i) \geq t$ entonces la clase predicha es P, o la clase N en caso de que $\Omega(i) < t$. Como ejemplo, consideremos un umbral 0.5. Para dicho umbral, la salida del clasificador para la instancia 1 sería un falso positivo, ya que la clase real es P, y el clasificador predeciría N ($0.27 < t$); por otro lado, la salida para la instancia 2 sería un verdadero negativo.

En la Tabla 8 se muestra en detalle, para distintos umbrales seleccionados (t), los valores de TP, TN, FP, y FN, así como los ratios de FP y TP. En este problema, con estos umbrales es suficiente para obtener la curva completa; en problemas más complejos, cabría seleccionar umbrales en intervalos lo suficiente en detalle como para obtener la curva. La Figura 5 muestra la curva ROC resultante de dibujar los valores de ratios de FP y TP calculados en la Tabla 8. Por último, el área bajo la curva ROC de esta curva sería de un valor de 0.72.

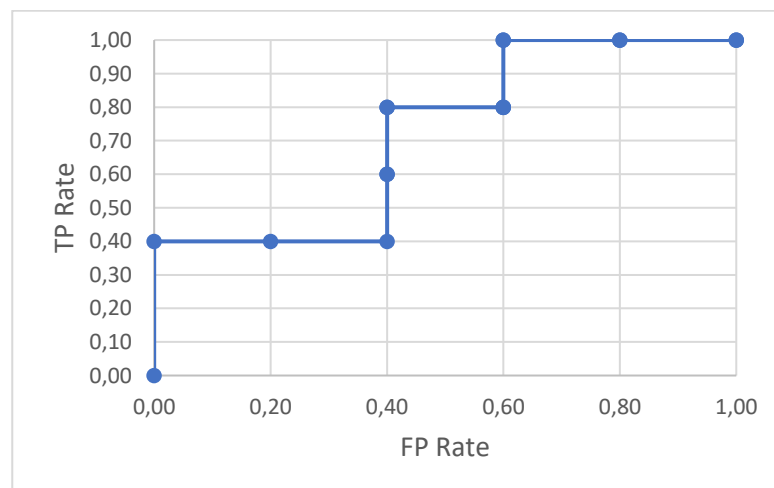


Figura 5. Curva ROC para el ejemplo de clasificador probabilístico.

Tabla 8. Valores de ratio de FP y TP para el ejemplo de clasificador probabilístico.

t	TP	TN	FP	FN	FP Rate	TP Rate
0,00	5	0	5	0	1,00	1,00
0,08	5	0	5	0	1,00	1,00
0,10	5	1	4	0	0,80	1,00
0,12	5	1	4	0	0,80	1,00
0,20	5	2	3	0	0,60	1,00
0,27	5	2	3	0	0,60	1,00
0,30	4	2	3	1	0,60	0,80
0,40	4	2	3	1	0,60	0,80
0,49	4	2	3	1	0,60	0,80
0,50	4	3	2	1	0,40	0,80
0,51	4	3	2	1	0,40	0,80
0,60	3	3	2	2	0,40	0,60
0,66	3	3	2	2	0,40	0,60
0,67	2	3	2	3	0,40	0,40
0,73	2	4	1	3	0,20	0,40
0,94	2	5	0	3	0,00	0,40
1,00	0	5	0	5	0,00	0,00

Referencias

- [Agg15] Aggarwal, C. C. (2015). Data Classification. Algorithms and Applications. *Chapman and Hall/CRC*.
- [Lar14] Larose, D. T., & Larose, C. D. (2014). *Discovering knowledge in data: an introduction to data mining* (Vol. 4). John Wiley & Sons.
- [Mai10] Maimon, O., & Rokach, L. (Eds.). (2010). Data mining and knowledge discovery handbook, 2nd edition. *Springer*.
- [Wit11] Witten, I. H., Frank, E., & Hall, M. A. (2011). Data mining: practical machine learning tools and techniques, 3rd edition. *Morgan Kaufmann*.