

Evaluación y Selección de Modelos

Aprendizaje Automático II-2021

Prof. Ricardo Ñanculef Alegría
jnancu@inf.utfsm.cl

Departamento de Informática UTFSM
Campus San Joaquín - II - 2021

Outline

- ▶ Técnicas de Evaluación & Selección de Modelos.
- ▶ Métricas de Evaluación en Clasificación.

EVALUACIÓN & SELECCIÓN DE MODELOS

Evaluación

- ▶ Evaluar un modelo de aprendizaje consiste, en general, en estimar su desempeño futuro, es decir, el desempeño que tendrá una vez que esté operando en la tarea para la que fue construido.
- ▶ Formalmente, lo que nos interesa es estimar su **error de predicción**

$$R(f) = \mathbb{E}(L(f(x), y)) = \sum_{x,y} L(f(x), y)p(x, y), \quad (1)$$

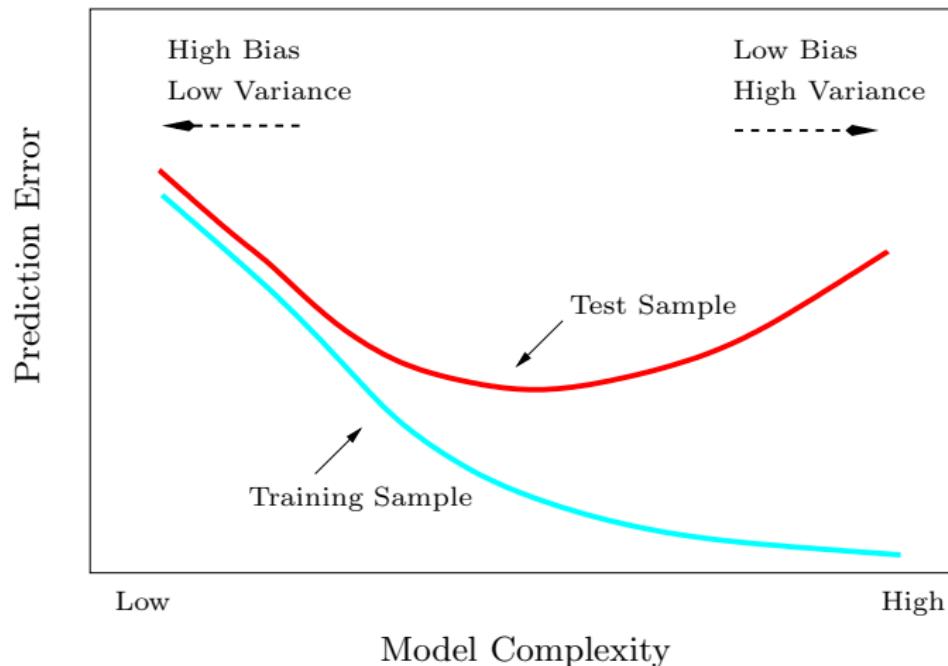
error que no necesariamente coincide con el **error de entrenamiento**,

$$\hat{R}(f) = \frac{1}{n} \sum_{i=1}^n L(f(x^{(i)}), y^{(i)}) \quad (2)$$

que es lo que usualmente terminan optimizando los algoritmos (veremos excepciones pronto).

Overfitting (Sobreajuste)

- ▶ Es muy común que el error de entrenamiento termine siendo muy pequeño, pero el error de predicción no.

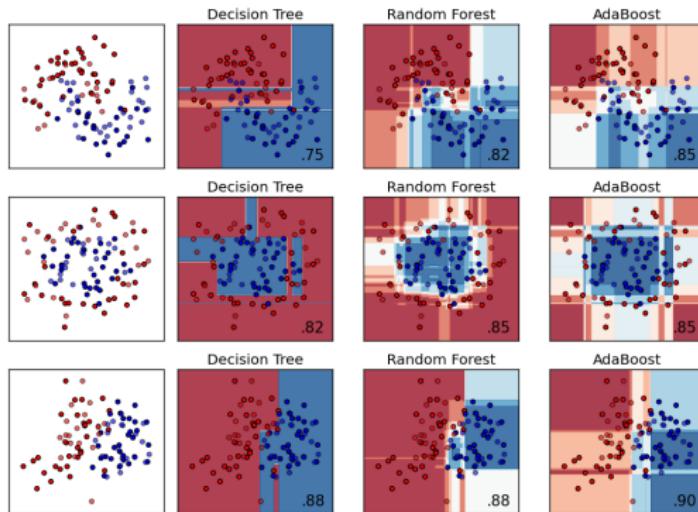


Evaluación vía Test Set

- ▶ Por lo anterior, hemos dicho que una práctica básica al momento de aplicar un método de aprendizaje a un problema es **evaluar** el resultado sobre un conjunto **independiente** del conjunto de entrenamiento.
- ▶ Este conjunto se denomina **conjunto de pruebas (test set)** y generalmente se reserva separando un subconjunto de datos del total disponible.
- ▶ La independencia de esos datos del modelo permite obtener un estimador realista del error de predicción.

Selección de Modelos

- Con frecuencia en una aplicación real será necesario elegir entre varios métodos/soluciones candidatas. Esa tarea se denomina selección de modelos.



Selección de Modelos

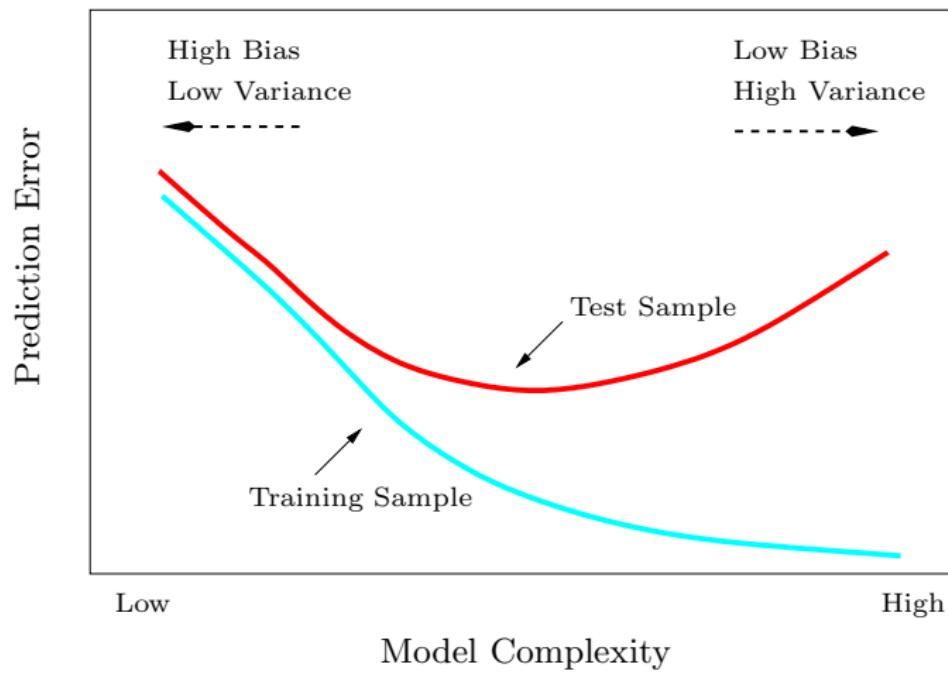
- ▶ La **selección** puede darse entre técnicas diferentes o entre variantes de una misma técnica (e.g. con diferentes atributos / representaciones).
- ▶ En cualquier caso, buscaremos elegir la técnica que tenga un menor error de predicción.



Selección de Modelos vía Test Set

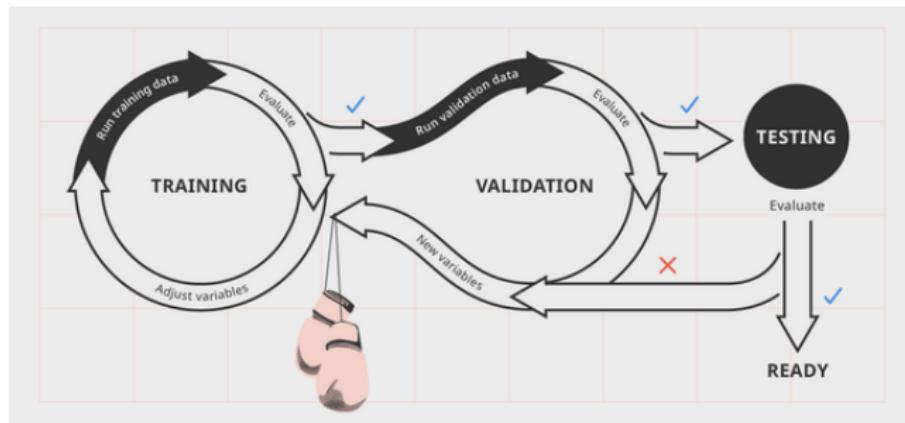
- ▶ Un error muy común es utilizar el conjunto de pruebas para **seleccionar** entre varios modelos y al final evaluar el modelo también en ese conjunto. Esto es **metodológicamente incorrecto**.
- ▶ El proceso de selección implica, hasta cierto punto, un segundo entrenamiento. Si la selección se hace en base al conjunto de pruebas, éste deja de ser independiente del modelo y se transforma en un segundo conjunto de entrenamiento.
- ▶ Si se usa el mismo conjunto de datos para **seleccionar** y para luego **evaluar**, se estará sub-estimando el error de predicción del modelo seleccionado.

Selección de Modelos



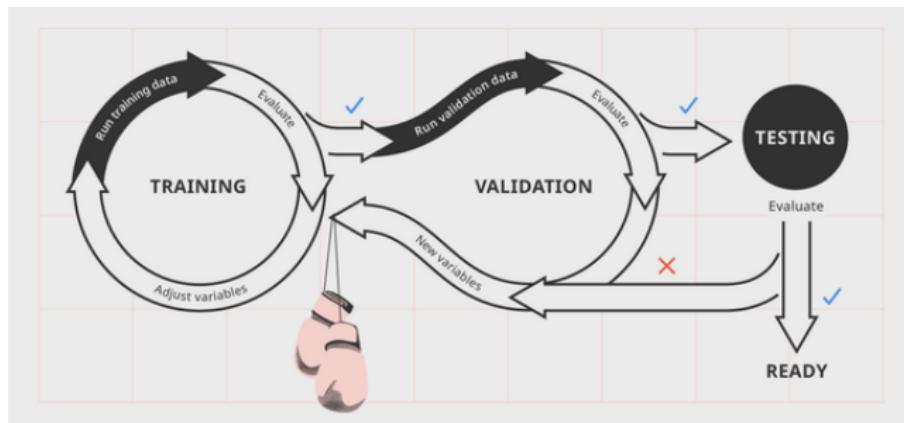
Conjunto de Validación

- ▶ Para poder hacer **selección de modelos** sin comprometer la independencia del conjunto de pruebas, con frecuencia reservaremos un **conjunto de validación**.
- ▶ La idea es entrenar los diferentes modelos candidatos y evaluarlos en el conjunto de validación para elegir aquel que finalmente se implementará.



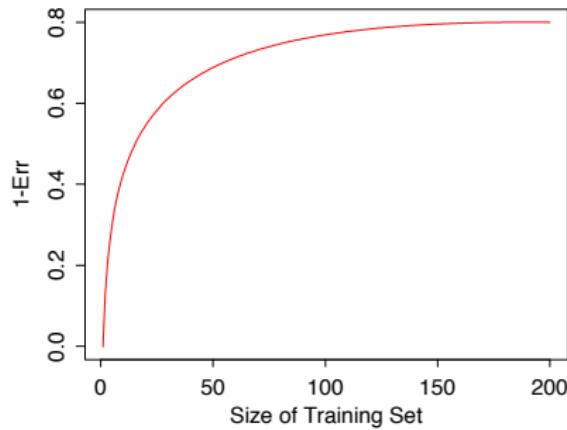
Evaluación versus Selección

- ▶ Sólo una vez que se ha terminado de seleccionar y entrenar un modelo, podemos evaluarlo en el conjunto de pruebas para estimar cómo se comportará en el futuro.



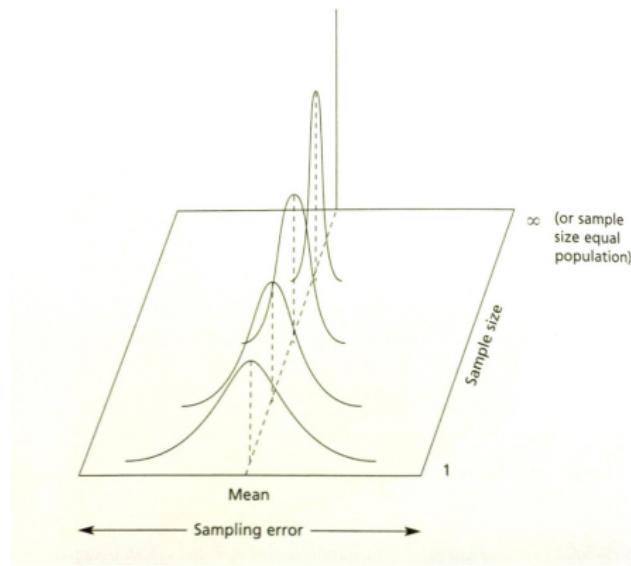
Dilema

- ▶ Mientras más ejemplos reservemos para validación (selección) o pruebas (evaluación) menos tendremos para entrenar el modelo.
- ▶ Mientras más ejemplos usemos para entrenar el modelo, mejores resultados obtendremos.



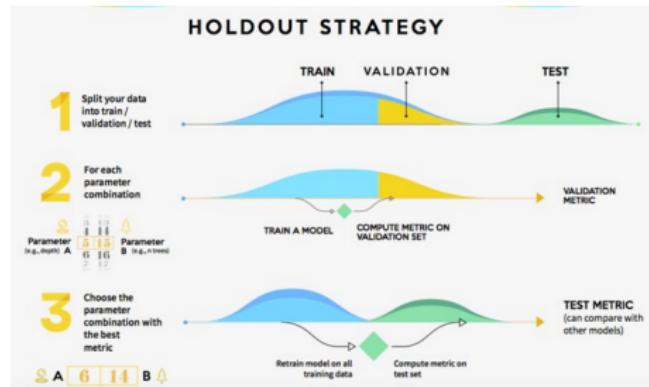
Dilema

- Mientras menos ejemplos reservemos para validación (selección) o pruebas (evaluación), menos precisa será nuestra estimación del error de predicción (alta varianza y por lo tanto, muy probable que, con una mala partición, encontremos un valor alejado del valor esperado).



Selección & Re-entrenamiento

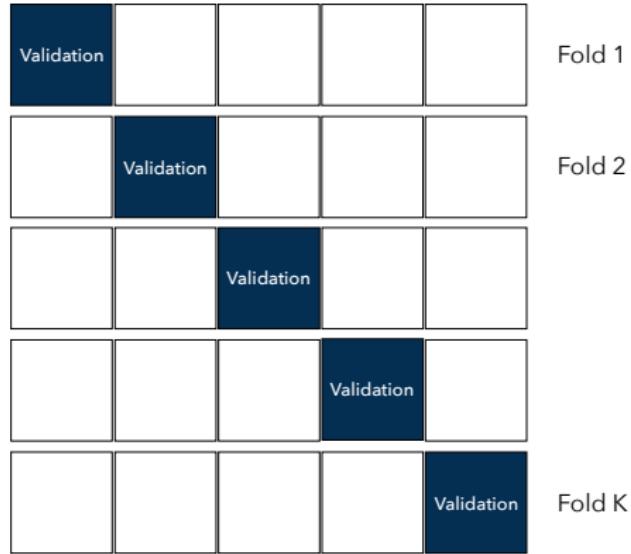
- ▶ Una práctica común consiste en re-entrenar el modelo con toda la data una vez que se ha completado el proceso de selección.



- ▶ Esto no elimina la posibilidad de haber hecho una mala/incorrecta selección: quizás con más datos era otro el modelo ganador.

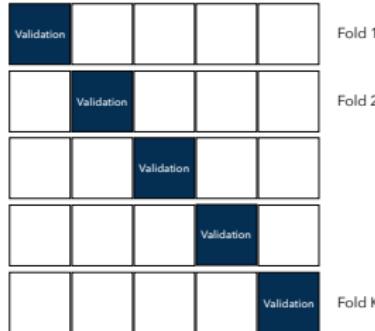
Cross-Validation

- ▶ Una idea simple para resolver este dilema es validación cruzada (**K-fold cross-validation**).



Cross-Validation

- ▶ Dividimos el conjunto de datos (que no son de pruebas) en K bloques de igual tamaño $m = n/K$.
- ▶ Para cada partición, entrenamos el modelo dejando fuera el bloque de validación y luego evaluamos el modelo en el bloque que dejamos aparte.
- ▶ Finalmente, tomamos como estimador del error de predicción el promedio de los errores de validación calculados sobre cada partición.
- ▶ Una vez elegido el modelo, **se re-entrena usando todos los bloques**.



Cross-Validation

Más formalmente,

- ▶ Sea $k : \{1, 2, \dots, n\} \rightarrow \{1, 2, \dots, K\}$ una función que mapea cada dato a su correspondiente bloque y f^{-k} la función/modelo obtenido entrenando con data la data excepto el bloque k .
- ▶ El estimador de validación cruzada del error de predicción (riesgo) del modelo se define como

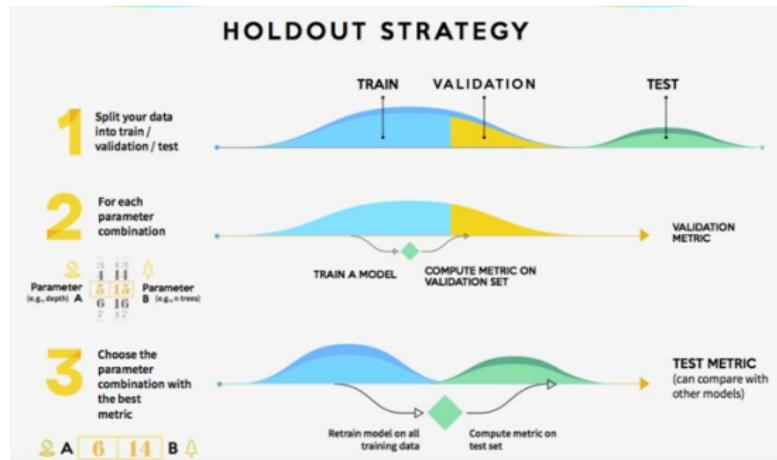
$$CV(f) = \frac{1}{n} \sum_{i=1}^n L\left(f^{-k(i)}(x^{(i)}), y^{(i)}\right) \quad (3)$$

donde $\{(x^{(i)}, y^{(i)})\}_{i=1}^n$ es el conjunto de ejemplo de entrenamiento.

- ▶ Típicamente $K = 10$ ó $K = 5$.

Cross-Validation & Re-Training

- Una vez elegido el modelo, se re-entrena usando todos los bloques.



Cross-Validation

Ejemplo

Supongamos que tenemos 20 predictores (atributos) X_1, \dots, X_{20} generados IID con distribución uniforme en $[0, 1]$. Y se define como $Y = I(\sum_{j=1}^{10} X_i > 5)$, es decir, 1 si $\sum_{j=1}^{10} X_i > 5$ y 0 en otro caso.

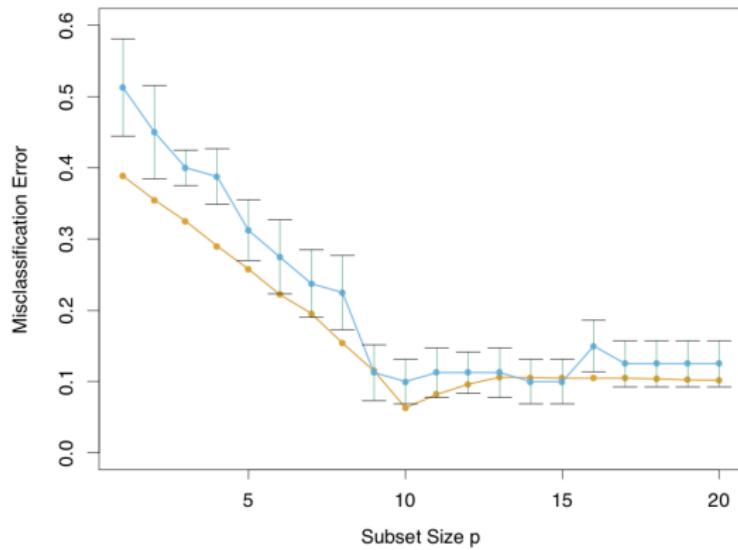
¿Cuál es el número adecuado de predictores en esta tarea?

¿Permite el error de entrenamiento detectar esto?

¿Logra cross-validation detectar esto?

Cross-Validation

Naranja: CV Error ($K = 10$) usando $n = 80$ ejemplos. Barras de error calculadas como la desviación estándar del error de cada fold. Celeste: Error de predicción real.



Errores Comunes usando CV

- ▶ La sencillez de CV hace que sea común encontrar errores metodológicos serios en su aplicación.
- ▶ Aplicar CV en una parte de la “pipeline” de procesamiento de datos pero usar todos los datos para otra (típicamente previa).
- ▶ Usar dos CV independientes: uno para hacer selección de modelos (“ahorra” el validation set) y otro para evaluar el modelo final (“ahorra” el test set).

Errores Comunes usando CV

Ejemplo

Tenemos 5000 predictores (atributos) X_1, \dots, X_{5000} generados IID con distribución normal estándar. Y se define como una v.a. independiente de los predictores con probabilidad 0.5 de ser 1 y probabilidad 0.5 de ser 0.

¿Cuál es el error de predicción para este problema?

¿Cuántos predictores son útiles para hacer la predicción?

¿Cuál debiese ser la correlación de los predictores con Y ?

¿Logra cross-validation detectar esto?

Errores Comunes usando CV

Ejemplo

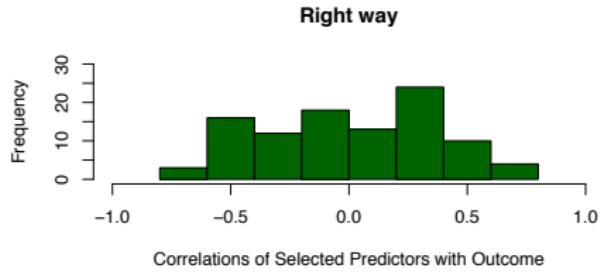
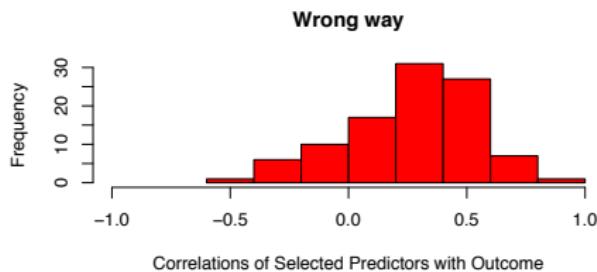
Simulamos $n = 80$ datos y aplicaremos dos versiones de CV, una mal hecha y una bien hecha.

Mal hecho: Primero elegimos los 100 predictores (X_j) más correlacionados con la clase (Y) usando todos los ejemplos. Luego ejecutamos CV con $K = 5$ y medimos, en cada fold, la correlación entre los predictores (X_j) y la clase (Y) usando los datos de validación para ese fold.

Bien hecho: Ejecutamos CV con $K = 5$. En cada fold, elegimos los 100 predictores (X_j) más correlacionados con la clase (Y) usando los ejemplos de entrenamiento de ese fold. Luego, medimos la correlación entre los predictores (X_j) y la clase (Y) usando los datos de validación para ese fold.

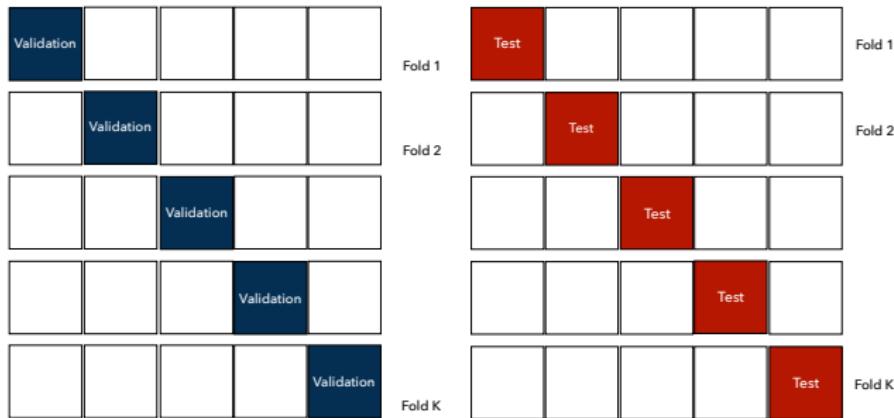
Errores Comunes usando CV

Resultados del experimento.



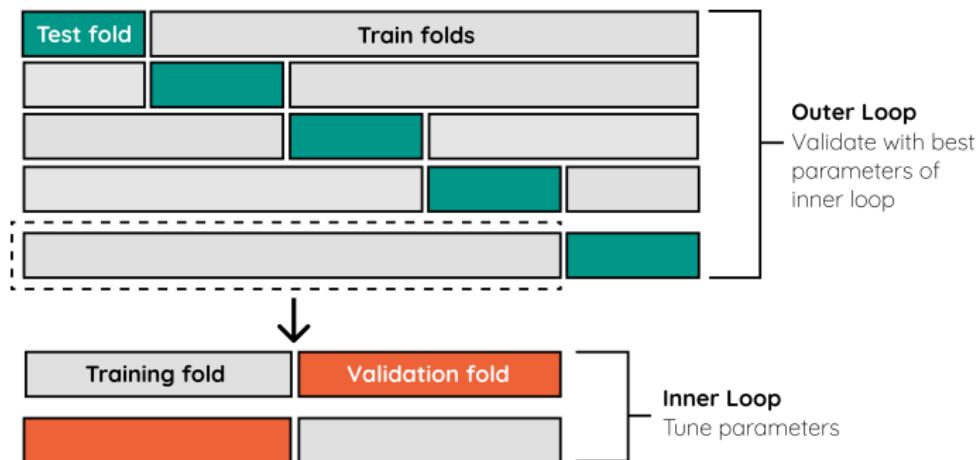
Errores Comunes usando CV

La misma situación se da cuando se usan dos CV independientes: uno para sustituir la extracción del validation set y otro para sustituir la extracción del test set.



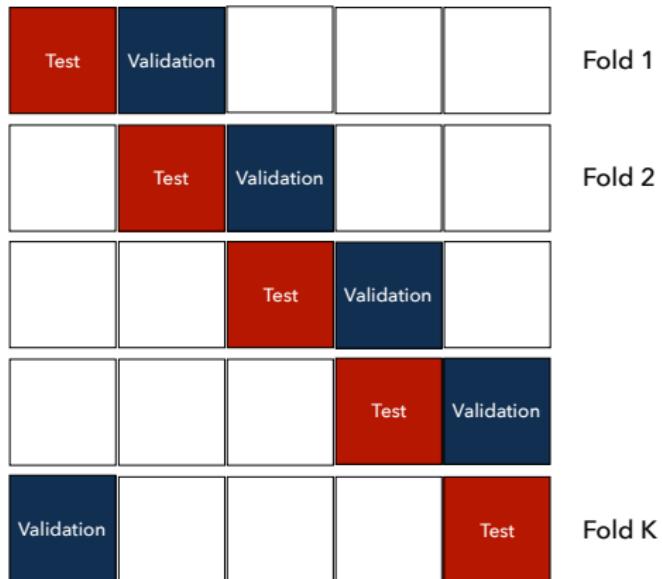
Cross-Validation Anidado

Si se desea usar CV para selección y también para evaluación, es necesario garantizar que en cada iteración (fold) de evaluación se usan datos nunca vistos ni para entrenamiento ni para validación.



Cross-Validation con Validación y Test

Una alternativa “económica” a CV Anidado:



Alternativas a Cross-Validation

- ▶ K-fold cross-validation es por lejos el estándar en la comunidad.
- ▶ Entre las pocas alternativas disponibles, vale la pena mencionar
 - Leave-one-out cross-validation (LOOCV).
 - Criterio de Akaike (AIC).
 - Criterio BIC (Bayesian Information Criterion).
 - Bootstrap.

LOOCV

- ▶ K-fold cross-validation con $K = n$.
- ▶ Extremadamente costoso en la práctica.
- ▶ Para algunos modelos y algunos criterios de entrenamiento (e.g. modelo estándar de regresión lineal) es posible mostrar que el sesgo y la varianza de CV como estimador del error de predicción decrece aumentando K (1*).
- ▶ Para algunos modelos y algunos criterios de entrenamiento, es posible aproximar analíticamente LOOCV.

(1*) Zhang, Yongli, and Yuhong Yang. "Cross-validation for selecting a model selection procedure." *Journal of Econometrics* 187.1 (2015): 95-112.

LOOCV

- ▶ Por ejemplo, en cualquier técnica en que las predicciones se obtengan como $\hat{Y} = S \cdot Y$, se verifica que (2*).

$$\frac{1}{n} \sum_i \left(\hat{f}^{-i}(x^{(i)} - y^{(i)}) \right)^2 = \frac{1}{n} \sum_i \left[\frac{\left(\hat{f}(x^{(i)} - y^{(i)}) \right)^2}{1 - S_{ii}} \right]. \quad (4)$$

Cuando S_{ii} es difícil de calcular, es posible usar la aproximación $\text{Tr}(S)/n$.

(2*) Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. The Elements of Statistical Learning. Springer. 2009.

Criterio de Akaike (AIC)

Hirotsugu Akaike



Criterio de Akaike (AIC)

- ▶ Supongamos que podemos poner la tarea como el problema de estimar una densidad de probabilidad $p(z; \theta)$ sobre los datos z ($\theta \in \mathbb{R}^p$ denota el vector de parámetros), de modo que la loss $L(f_\theta(x), y) = -\log p(z; \theta)$. Es posible mostrar que si la distribución real de los datos $p(z)$ está en la familia $(\exists \theta_0, p(z) = p(z; \theta_0))$

$$\mathbb{E}(L(f_{\hat{\theta}}(x), y)) \approx -\frac{1}{n} \sum_i \log p(z^{(i)}; \hat{\theta}) + \frac{p}{n} \quad (5)$$

donde $\{z^{(i)}\}_{i=1}^n$ es el conjunto de ejemplos de entrenamiento y p es el número total de parámetros.

- ▶ Por razones históricas, el estimador se multiplica por 2

$$AIC(f) = -2 \frac{1}{n} \sum_i \log p(z^{(i)}; \hat{\theta}) + \frac{2p}{n} \quad (6)$$

Criterio de Akaike (AIC)

- ▶ Para regresión de mínimos cuadrados (modelo gausiano), el criterio toma la forma

$$\mathbb{E}(L(\hat{f}_\theta(x), y)) \propto \log(RSS/n) + 2p/n \quad (7)$$

- ▶ La aproximación se basa en un gran número de supuestos y requiere una especialización al modelo de interés.
- ▶ Aún si la aproximación es asintótica, recomendable sólo cuando se tiene un número pequeño de ejemplos (porque en este caso CV tiende a comportarse más mal).

Criterio BIC (Bayesian Information Criterion)

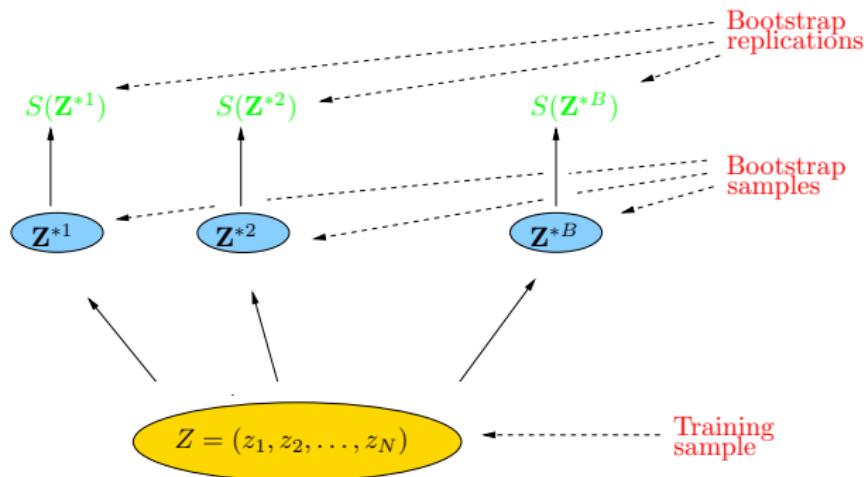
- ▶ Aparentemente similar a Akaike,

$$BIC(f) = -\frac{2}{n} \sum_i \log p(z^{(i)}; \hat{\theta}) + \log(n) \frac{p}{n} \quad (8)$$

donde $\{z^{(i)}\}_{i=1}^n$ es el conjunto de ejemplos de entrenamiento y p es el número total de parámetros.

- ▶ Aproximación a $p(\bar{X}|f_\theta) = \int p(\bar{X}|f, \theta)p(\theta|f)$.
- ▶ Mucho más conservador que Akaike.

Bootstrap



Bootstrap

- ▶ Supongamos que entrenamos un modelo/clasificador f^{*b} en la b-ésima muestra bootstrap (de un total de B).
- ▶ Una forma ingenua de estimar el error de predicción sería

$$\hat{E}_{B_1}(f) = \frac{1}{nB} \sum_{i=1}^n \sum_{b=1}^B L\left(y^{(i)}, f^{*b}(x^{(i)})\right) \quad (9)$$

- ▶ Este estimador tiende a ser demasiado optimista porque la muestra de entrenamiento (donde se evalúa el modelo) y las muestras bootstrap tienen elementos en común (63.2 % en valor esperado).

Bootstrap

- ▶ Una estimación que intenta “imitar” cross-validation estaría dada por

$$\hat{E}_{B_2}(f) = \frac{1}{n} \sum_{i=1}^n \frac{1}{|C^{-i}|} \sum_{b \in C^{-i}} L\left(y^{(i)}, f^{*b}(x^{(i)})\right) \quad (10)$$

donde $C^{-i} \subset [B]$ es el conjunto de índices de las muestras bootstrap que no contienen el ejemplo $x^{(i)}$.

- ▶ En la práctica este estimador tiende a ser demasiado pesimista: en promedio, una muestra bootstrap contiene 63.2 % del dataset original ($\approx CV-2$).
- ▶ Una corrección que suele funcionar bien cuando el error de entrenamiento no es demasiado pequeño es:

$$\hat{E}_{0.632}(f) = 0.632 \hat{E}_{B_2}(f) + 0.328 \hat{R}(f) \quad (11)$$

donde $\hat{R}(f)$ es el error de entrenamiento.

Bootstrap

Leo Breiman (izquierda) y Bradley Efron (izquierda)



EVALUACIÓN DE CLASIFICADORES

Evaluación

- ▶ El propósito de la evaluación es predecir el desempeño futuro del modelo.
- ▶ Una vez elegida una estrategia de entrenamiento, validación y pruebas consistente, ¿Qué medimos en el test set?
- ▶ Hasta ahora hemos asumido que tanto para seleccionar como para evaluar usaremos la función de costo (loss) con la que entrenamos el clasificador.
- ▶ Como no todos los clasificadores usan la *misclassification loss* para entrenarse, tiene sentido también evaluar la *accuracy* del modelo.
- ▶ En muchos casos es útil emplear métricas adicionales que nos permitan hacer un análisis más fino del modelo.
- ▶ Estas métricas adicionales podrían también emplearse para seleccionar el modelo (y hacer correcciones al comportamiento observado). En este caso, esto debe hacerse de manera consistente con las recomendaciones ya mencionadas.

Problemas Desbalanceados

- ▶ Utilizar métricas alternativas es particularmente importante en:
 - Problemas **desbalanceados**, en que la clases no están igualmente representadas.
 - Problemas en que nos interesan más algunas clases que otras o los **costos de un error** no son uniformes.

Ejemplo

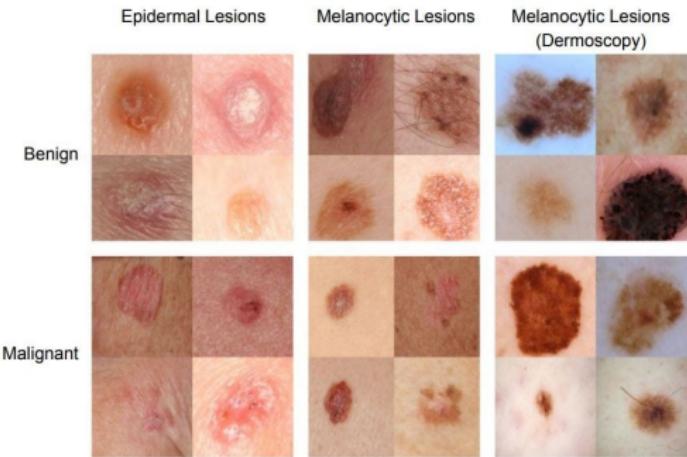
Imaginemos que necesitamos distinguir entre imágenes casos de cáncer maligno y benigno a partir de una imagen.

Si la clase maligno representa el 5 % de los casos y la otra clase el 95 %, ¿Cuál sería el error de clasificación esperado de un clasificador que predice siempre la clase benigno?

Problemas Desbalanceados

Ejemplo

Como la clase maligno representa sólo el 5 % de los casos, el 95 % de los casos serán manejados correctamente.



Matrices de Confusión

- Una de las herramientas más utilizadas para evaluar clasificadores es la matriz de confusión (es posible normalizar por el número total de datos).

		True/Actual		
		Cat (img alt="Cat icon" data-bbox="438 388 481 428")/)	Fish (img alt="Fish icon" data-bbox="578 388 621 428"/>)	Hen (img alt="Hen icon" data-bbox="718 388 761 428"/>)
Predicted	Cat (img alt="Cat icon" data-bbox="218 458 311 498")/)	4	6	3
	Fish (img alt="Fish icon" data-bbox="218 538 311 578")/)	1	2	0
	Hen (img alt="Hen icon" data-bbox="218 618 311 658")/)	1	2	6

(en sklearn, las clases verdaderas se ponen en las filas)

Matrices de Confusión Binarias

- ▶ Si consideramos una clase de interés c , denominaremos
 - **Positivos**, a todos los datos que son de la clase c .
 - **Negativos**, a todos los datos que no son de la clase c .

		True/Actual	
		Positive (🐶)	Negative
Predicted	Positive (🐶)	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

(en sklearn, las clases verdaderas se ponen en las filas)

Falsos y Verdaderos Positivos

- ▶ Si consideramos una clase de interés c , denominaremos
 - **Verdaderos Positivos**, a todos los datos que son de la clase c y el modelo/clasificador predice como pertenecientes a la clase c .
 - **Falsos Positivos (FP)**, a todos los datos que NO son de la clase c , pero el modelo/clasificador predice como pertenecientes a la clase c .

		True/Actual	
		Positive (🐶)	Negative
Predicted	Positive (🐶)	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

Falsos y Verdaderos Negativos

- ▶ Si consideramos una clase de interés c , denominaremos
 - **Verdaderos Negativos (TN)**, a todos los datos que NO son de la clase c y el modelo/clasificador predice como NO pertenecientes a la clase c .
 - **Falsos Negativos (FN)**, a todos los datos que son de la clase c y el modelo/clasificador predice como NO pertenecientes a la clase c .

		True/Actual	
		Positive (🐶)	Negative
Predicted	Positive (🐶)	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

Precisión

- La **precisión** se define como la fracción de predicciones correctas que hace el clasificador respecto a una clase c .

$$P = \frac{TP}{TP + FP} \quad (12)$$

		True/Actual	
		Positive (好人)	Negative
Predicted	Positive (好人)	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

Ejercicio: determine la precisión de cada clase en la matriz de más arriba.

Recall o Sensibilidad

- El **recall** se define como la fracción ejemplos de una clase c (positivos) que el clasificador efectivamente reconoce.

$$R = \frac{TP}{TP + FN} \quad (13)$$

- También se denomina sensibilidad o tasa de verdaderos positivos.

		True/Actual	
		Positive (好人)	Negative
Predicted	Positive (好人)	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

Ejercicio: determine el recall de cada clase en la matriz de más arriba.

Especificidad

- La **especificidad** se define como la fracción ejemplos negativos (que no son de la clase c) que el clasificador efectivamente reconoce. Es equivalente al recall de la “clase” \bar{c} .

$$S = \frac{TN}{TN + FP} \quad (14)$$

- La tasa de falsos positivos se define como $1 - S$.

		True/Actual	
		Positive (深厚的)	Negative
Predicted	Positive (深厚的)	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

Ejercicio: determine la especificidad de cada clase en la matriz de más arriba.

Accuracy (Exactitud)

- La fracción de predicciones correctas del clasificador (exactitud) se puede calcular a partir de la matriz.

$$A = \frac{TN + TP}{TN + TP + FP + FN} \quad (15)$$

		True/Actual	
		Positive (好人)	Negative
Predicted	Positive (好人)	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

Ejercicio: determine la exactitud de cada clase en la matriz de más arriba.

Promedio Macro & Micro

Supongamos que tenemos varias clases c_1, c_2, \dots, c_K y nos interesa evaluarlas en conjunto. ¿Cómo agregamos las métricas anteriores?

- ▶ **Macro Average:** Tomar el promedio aritmético (de la métrica de interés) sobre todas las clases.
- ▶ **Macro Average Pesado:** Tomar un promedio con pesos sobre todas las clases, usando pesos que e.g. reflejen la probabilidad a-priori de cada una.
- ▶ **Micro Average:** Construir una única matriz de confusión sumando los FP, FN, TP, TN correspondientes a cada clase. Sobre esta matriz global se calcula la métrica de interés.

Promedio Macro & Micro

Ejemplo

		True/Actual		
		Cat (😺)	Fish (🐠)	Hen (🐓)
Predicted	Cat (😺)	4	6	3
	Fish (🐠)	1	2	0
	Hen (🐓)	1	2	6

		True	
		+	-
Pred	+	+	-
	-	-	+

		True	
		+	-
Pred	+	+	-
	-	-	+

		True	
		+	-
Pred	+	+	-
	-	-	+

Promedio Macro & Micro

Ejemplo

		True/Actual		
		Cat (😺)	Fish (🐠)	Hen (🐓)
Predicted	Cat (😺)	4	6	3
	Fish (🐠)	1	2	0
	Hen (🐓)	1	2	6

		True	
		+	-
Pred	+	4	9
	-	2	10

		True	
		+	-
Pred	+	2	1
	-	8	14

		True	
		+	-
Pred	+	6	3
	-	3	13

Promedio Macro & Micro

Ejemplo

		True	
		+	-
Pred	+	4	9
	-	2	10

		True	
		+	-
Pred	+	2	1
	-	8	14

		True	
		+	-
Pred	+	6	3
	-	3	13

		True	
		+	-
Pred	+		
	-		

Promedio Macro & Micro

Ejemplo

		True	
		+	-
Pred	+	4	9
	-	2	10

		True	
		+	-
Pred	+	2	1
	-	8	14

		True	
		+	-
Pred	+	6	3
	-	3	13

		True	
		+	-
Pred	+	12	13
	-	13	37

F-Score

- ▶ Una forma bastante clásica de agregar precisión y recall consiste en tomar la media armónica entre ambos índices

$$F_1 = 2 \frac{P \cdot R}{P + R} \quad (16)$$

- ▶ Recordemos que la media armónica H de n valores x_1, x_2, \dots, x_n satisface

$$\min(x_1, x_2, \dots, x_n) \leq H(x_1, x_2, \dots, x_n) \leq n \cdot \min(x_1, x_2, \dots, x_n) \quad (17)$$

- ▶ Es posible asignar también un peso al recall respecto a la precisión

$$F_\beta = (1 + \beta)^2 \frac{P \cdot R}{(\beta^2 P) + R} \quad (18)$$

(para $\beta > 1$ el recall es más importante)

F-Score

- ▶ Una forma bastante clásica de agregar precisión y recall consiste en tomar la media armónica entre ambos índices

$$F_1 = 2 \frac{P \cdot R}{P + R} \quad (19)$$

- ▶ Recordemos que la media armónica H de n valores x_1, x_2, \dots, x_n satisface

$$\min(x_1, x_2, \dots, x_n) \leq H(x_1, x_2, \dots, x_n) \leq n \cdot \min(x_1, x_2, \dots, x_n) \quad (20)$$

- ▶ Es posible asignar también un peso al recall respecto a la precisión

$$F_\beta = (1 + \beta)^2 \frac{P \cdot R}{(\beta^2 P) + R} \quad (21)$$

(para $\beta > 1$ el recall es más importante)

Índice de Fowlkes–Mallows

- ▶ Otra forma de agregar precisión y recall se denomina índice de Fowlkes–Mallows y consiste en tomar la media geométrica entre ambos índices

$$FM = \sqrt{PR} \quad (22)$$

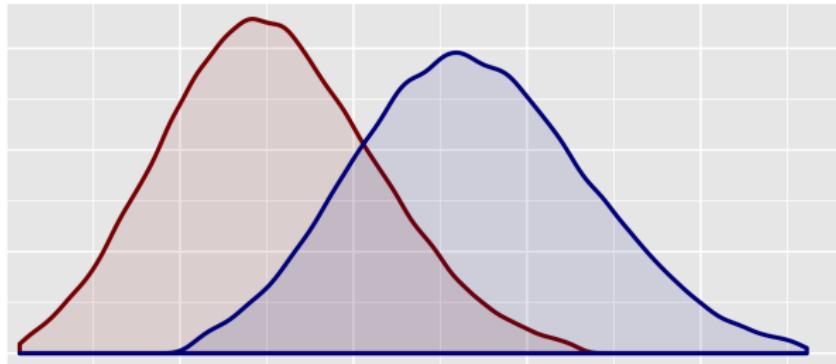
- ▶ Recordemos que la media armónica H de n valores x_1, x_2, \dots, x_n , la media geométrica G y la media aritmética M se relacionan como

$$H \leq G \leq M \quad (23)$$

- ▶ Fowlkes–Mallows penaliza menos un desequilibrio entre precisión y recall.

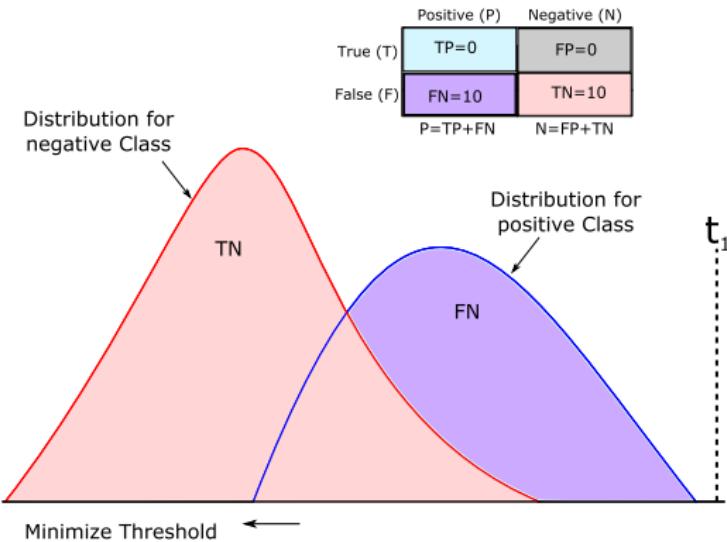
Bayes, Precisión & Recall

- ▶ Usualmente existirá un tradeoff entre precisión y recall: si somos muy “selectivos” para predecir una clase tiende a aumentar la precisión, pero disminuir el recall y viceversa.
- ▶ Esto está estrechamente relacionado con el concepto de *error de Bayes* que hemos discutido.



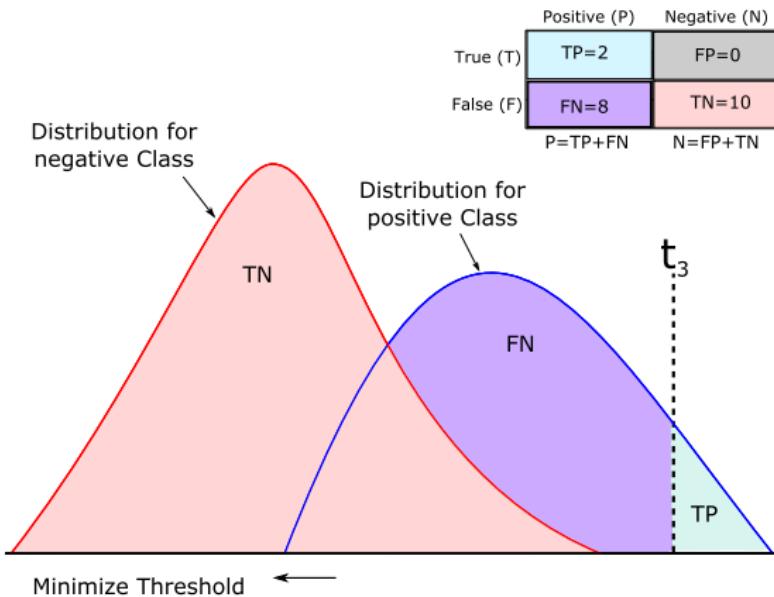
Bayes, Precisión & Recall

- ▶ Es posible modificar la regla de elegir la clase más probable *a posteriori* para modificar la matriz de confusión obtenida.
- ▶ Asumiendo un clasificador de tipo probabilístico se predice la clase de interés (c) cuando $p(y = c|x) > \theta$.



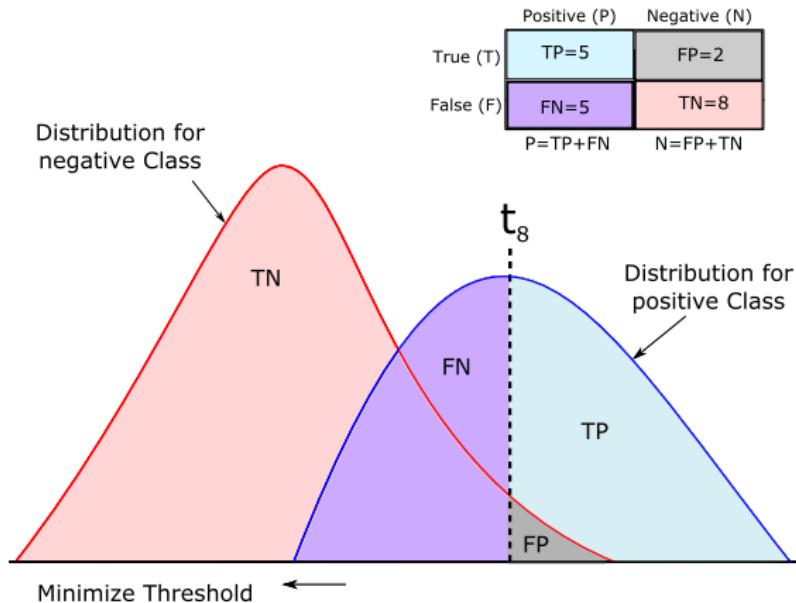
Bayes, Precisión & Recall

- ▶ Es posible modificar la regla de elegir la clase más probable *a posteriori* para modificar la matriz de confusión obtenida.



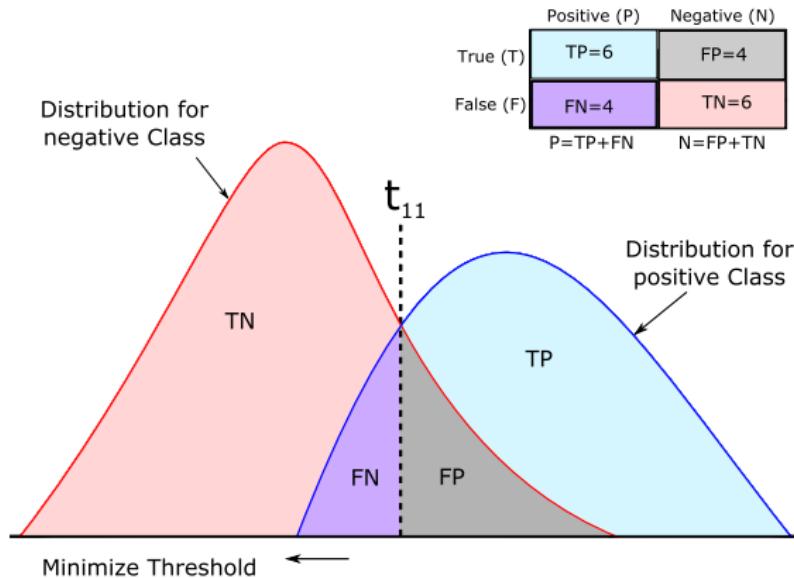
Bayes, Precisión & Recall

- ▶ Es posible modificar la regla de elegir la clase más probable *a posteriori* para modificar la matriz de confusión obtenida.



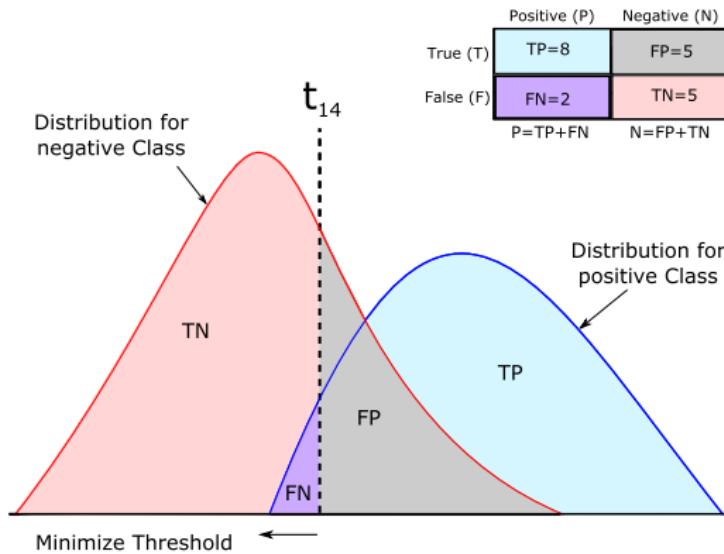
Bayes, Precisión & Recall

- ▶ Es posible modificar la regla de elegir la clase más probable *a posteriori* para modificar la matriz de confusión obtenida.



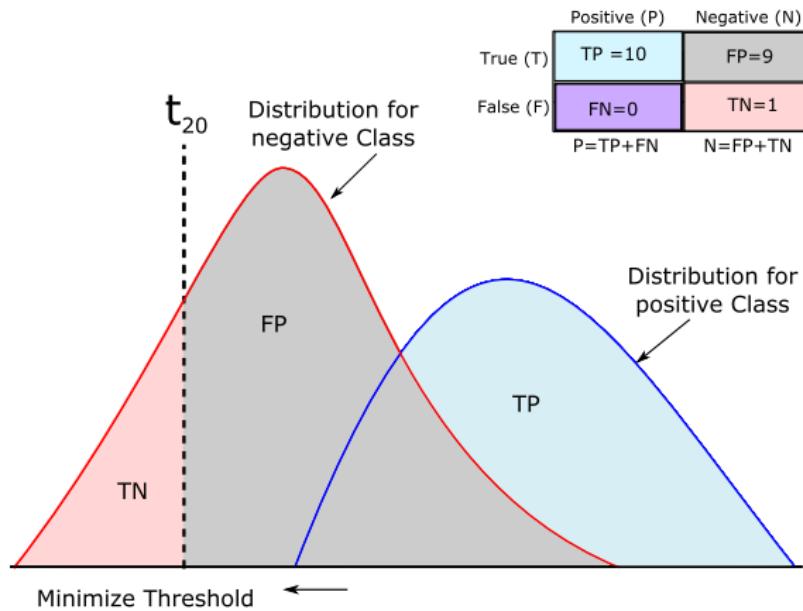
Bayes, Precisión & Recall

- ▶ Es posible modificar la regla de elegir la clase más probable *a posteriori* para modificar la matriz de confusión obtenida.



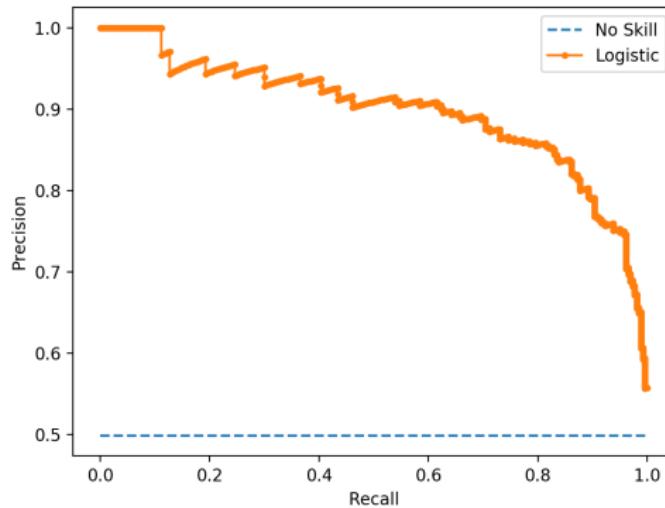
Bayes, Precisión & Recall

- ▶ Es posible modificar la regla de elegir la clase más probable *a posteriori* para modificar la matriz de confusión obtenida.



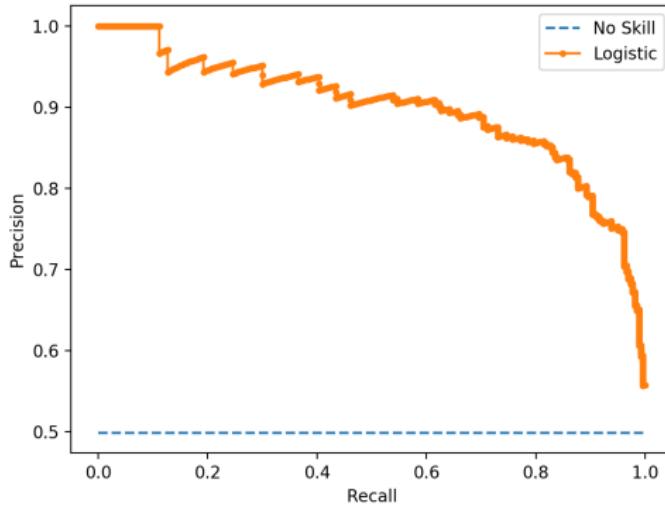
Curva Precisión-Recall

- ▶ Considerando diferentes umbrales de decisión, aparecerá una curva con diferentes niveles de precisión y recall.



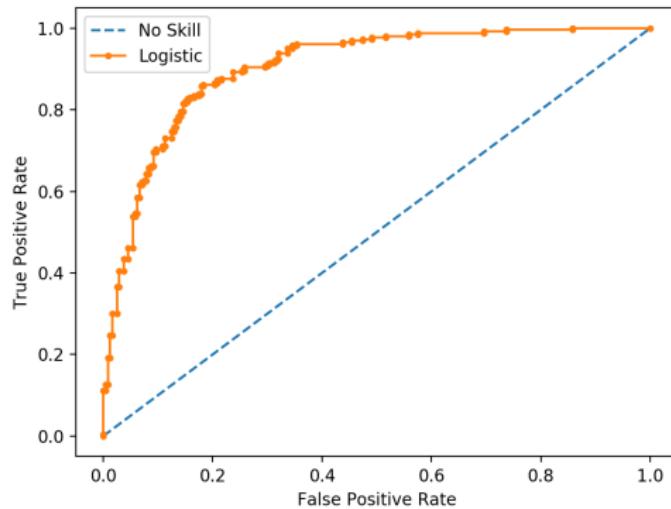
AUPR

- ▶ Una forma de evaluar globalmente un clasificador consiste en estimar el área bajo esa curva (AUPR) usando e.g. la regla del trapecio.
- ▶ Métrica muy utilizada cuando se tiene una clase de interés que está sub-representada u ocurre con menor frecuencia.



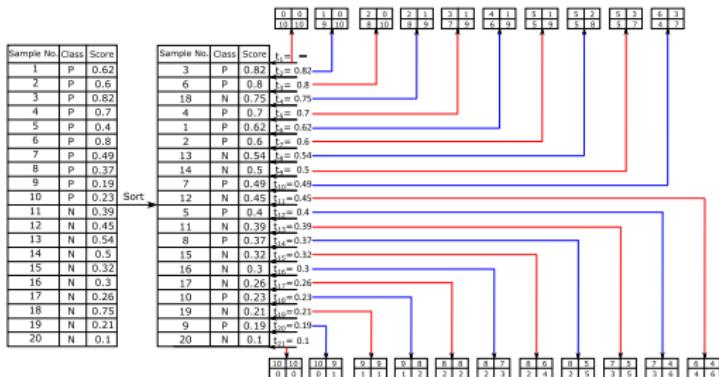
Curva ROC

- ▶ Una alternativa a la curva PR es la curva ROC. En el eje X ponemos la tasa de falsos positivos ($1 - S$, 1-especificidad) y en el eje Y la sensibilidad.
- ▶ Como en el caso anterior, es usual tomar el área como un indicador global.



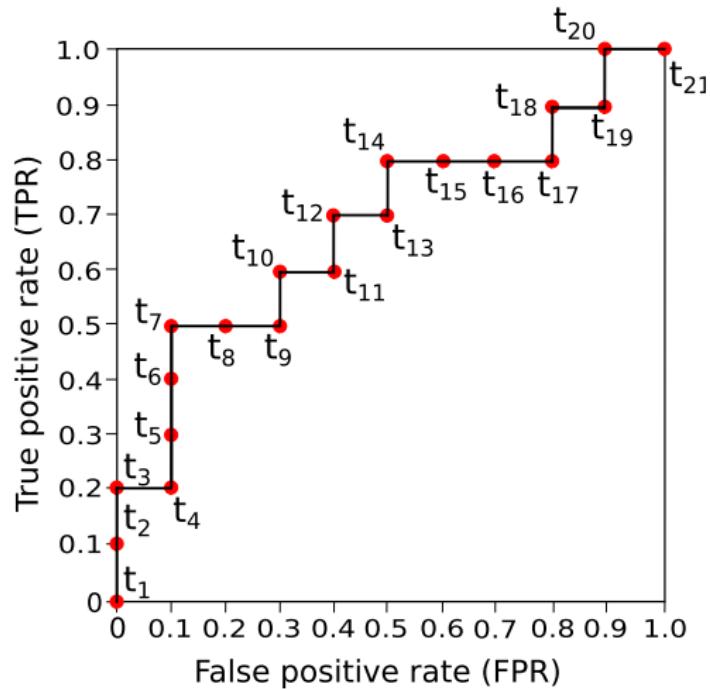
Construcción de las Curvas

- Asumiendo un clasificador de tipo probabilístico, tanto la curva ROC como la curva PR se suelen construir usando como umbrales de decisión posibles las diferentes probabilidades que predice el modelo sobre los datos usados para evaluar.



Construcción de las Curvas

- ▶ Considerar otros umbrales no cambia la curva.



ROC o PR?

- ▶ Existe aún en la comunidad una intensa discusión sobre qué curva reportar o qué curva usar para elegir un clasificador.
- ▶ Ambas curvas consideran el recall o sensibilidad en un eje: capacidad de detectar la clase positiva.
- ▶ La curva ROC contrasta esa habilidad contra la capacidad de detectar simultáneamente la otra clase (especificidad), es decir la capacidad de “dejar pasar (correctamente) casos de la otra” .
- ▶ La curva PR contrasta esa habilidad contra la contra la capacidad de que las predicciones positivas efectivamente lo sean.
- ▶ Si se invierte el etiquetado (positivo/negativo) la curva ROC se mantiene invariable en cambio la curva PR cambia.

ROC o PR?

- ▶ La observación anterior muestra, en general, la curva PR es más apropiada cuando se tiene un interés especial por detectar una de las clases.
- ▶ En el caso de problemas desbalanceados (una clase muy poco representada o difícil) suele ser aconsejado estudiar la curva PR asociada a esa clase.
- ▶ En problemas balanceados o donde todas las clases interesan por igual la curva ROC es más informativa.
- ▶ Cuando se hacen agregaciones sobre las diferentes clases balanceadas, suele usarse la curva ROC con una descomposición OVO.
- ▶ En ciertas áreas (medicina) suele utilizarse más la curva ROC, mientras que en otras (IR) es más común la curva PR.

ROC o PR?

- ▶ Recientemente se han demostrado algunas relaciones importantes entre las dos curvas
 - Una curva domina otra curva en el espacio ROC (todos sus puntos por encima) si y sólo si domina en el espacio PR.
 - Un clasificador puede tener mayor área que otro bajo la curva ROC, pero menor área bajo la curva PR.

The Relationship Between Precision-Recall and ROC Curves

Jesse Davis
Mark Goadrich

Department of Computer Sciences and Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, 1210 West Dayton Street, Madison, WI, 53706 USA

JDAVIS@CS.WISC.EDU

RICHM@CS.WISC.EDU

Abstract

Receiver Operator Characteristic (ROC) curves are commonly used to present results for binary decision problems in machine learning. However, when dealing with highly skewed datasets, Precision-Recall (PR) curves give a more informative picture of an algorithm's performance. We show that a deep connection exists between ROC space and PR space, such that a curve dominates in ROC space if and only if it dominates

in the class distribution. Drummond and Holte (2000; 2004) have recommended using cost curves to address this issue. Cost curves are an excellent alternative to ROC curves, but discussing them is beyond the scope of this paper.

Precision-Recall (PR) curves, often used in Information Retrieval (Manning & Schütze, 1999; Raghavan et al., 1989), have been cited as an alternative to ROC curves for tasks with a large skew in the class distribution (Bockhorst & Craven, 2005; Bunescu et al., 2004; Davis et al., 2005; Goadrich et al., 2004; Kok &