Classificazione: tecniche per la validazione

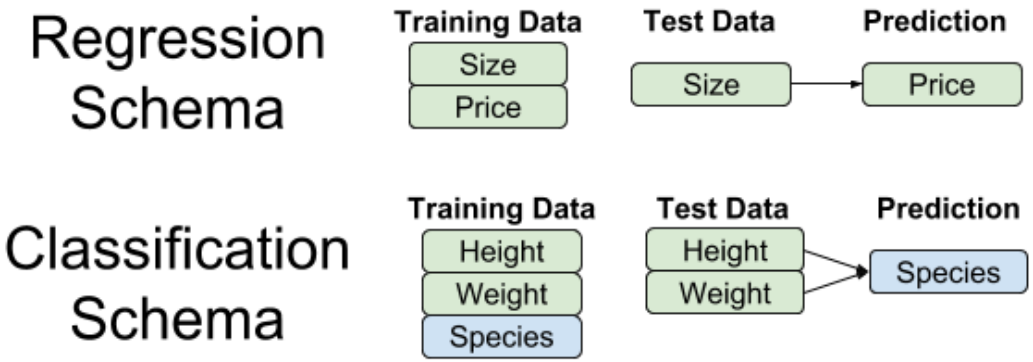
Algoritmi per l'intelligenza artificiale

Vincenzo Bonnici
Corso di Laurea Magistrale in Scienze Informatiche
Dipartimento di Scienze Matematiche, Fisiche e Informatiche
Universitá degli Studi di Parma

2025-2026

Classificazione vs regressione

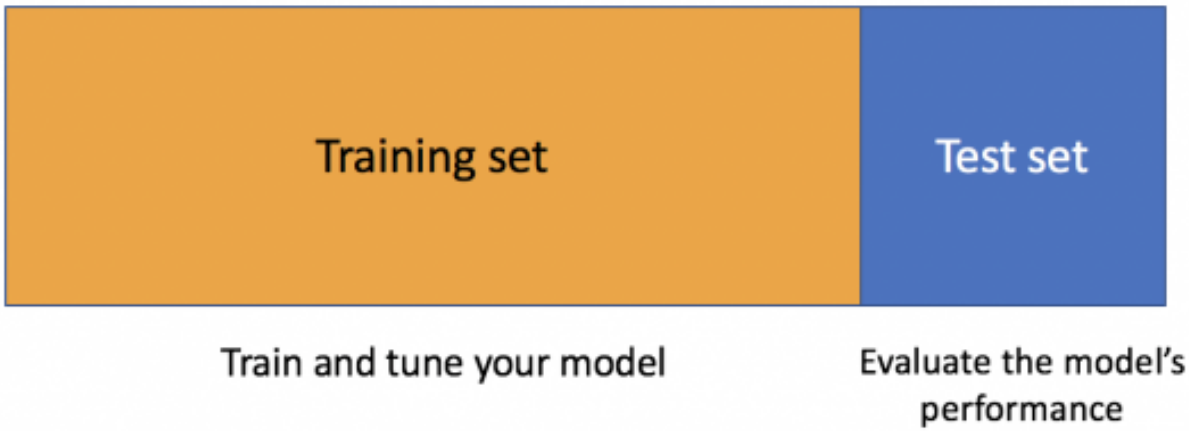
Piccola nota: nella regressione cerchiamo di costruire un modello per predirre un determinato valore numerico di output dato uno o più valori numerici di input.
Nella classificazione vogliamo predirre la classe di un oggetto dato uno o più dati di input di qualsivoglia natura.



Di conseguenza gli strumenti che possiamo applicare per valutare le performance di un modello predittivo di classificazione saranno sostanzialmente diversi dalle metriche utilizzare per valutare i modelli di regressione (tipicamente basi sul calcolo numerico dell'errore).

Modello di validazione base

Il modello base utilizzato per valutare un sistema di classificazione è quello descritto precedentemente, dove dividiamo i dati disponibili in training set e test set, secondo una certa **percentuale**.



Nel training set i dati delle label vengono utilizzati per addestrare il classificatore. Mentre l'informazione sulla label del test set viene utilizzate per valutarne la bontà. Quindi cerchiamo di rispondere alla domanda: il classificatore ha correttamente assegnato le label si dati del test set?

Modello di validazione base

Siamo però interessati a rispondere anche ad altre domande. Quali, per esempio:
il nostro classificatore performa in modo bilanciato nel valutare tutti i tipi di label?
o ha delle preferenze?
e tali preferenze da cosa dipendono?

In linea generale, siamo interessati a valutare i classificatori secondo indici matematici che ci permettono sia di avere delle stime oggettive sia di automatizzare anche altre fasi della progettazione o sviluppo del classificatore stesso.

Modello di validazione base

Tra questi indici quelli dall'aspetto più computazionale sono:

- **Accuratezza** del classificatore: bontà nel predire le etichette delle classi
- **Robustezza**: capacità di manipolare dati con rumore e mancanti
- **Velocità**
  - Tempo necessario per costruire il modello (training time)
  - Tempo necessario per usare il modello (classification/prediction time)
- **Scalabilità**: efficienza su dati presenti in memoria secondaria
- **Interpretabilità** dei risultati

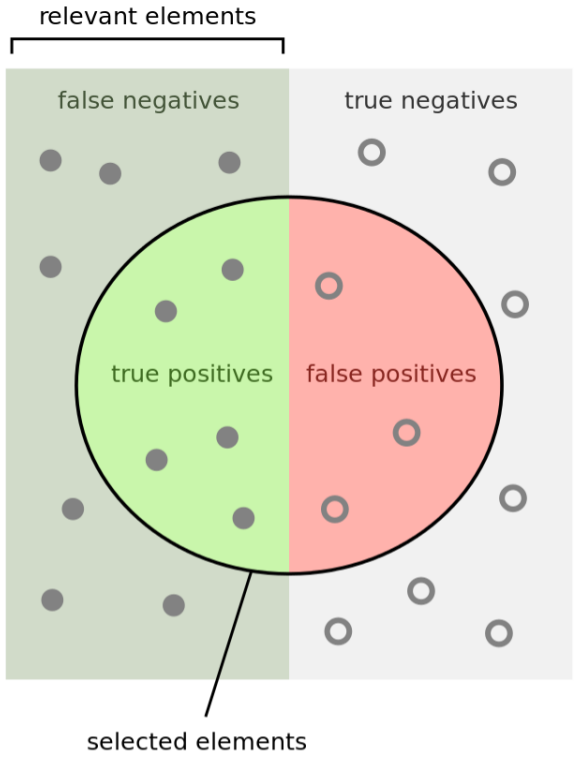
Precision e recall

Dividiamo i dati del training set in quattro insiemi secondo i risultati ottenuti dalla validazione del classificatore.

Ammettiamo che l'insieme delle classi $\mathbb{C}$ del nostro esperimento abbia cardinalità 2, quindi $\mathbb{C} = \{A, B\}$.

Allora, relativamente ad una delle due classi possiamo definire alcune misure per calcolare la bontà dell"algoritmo in valutare tale classe.

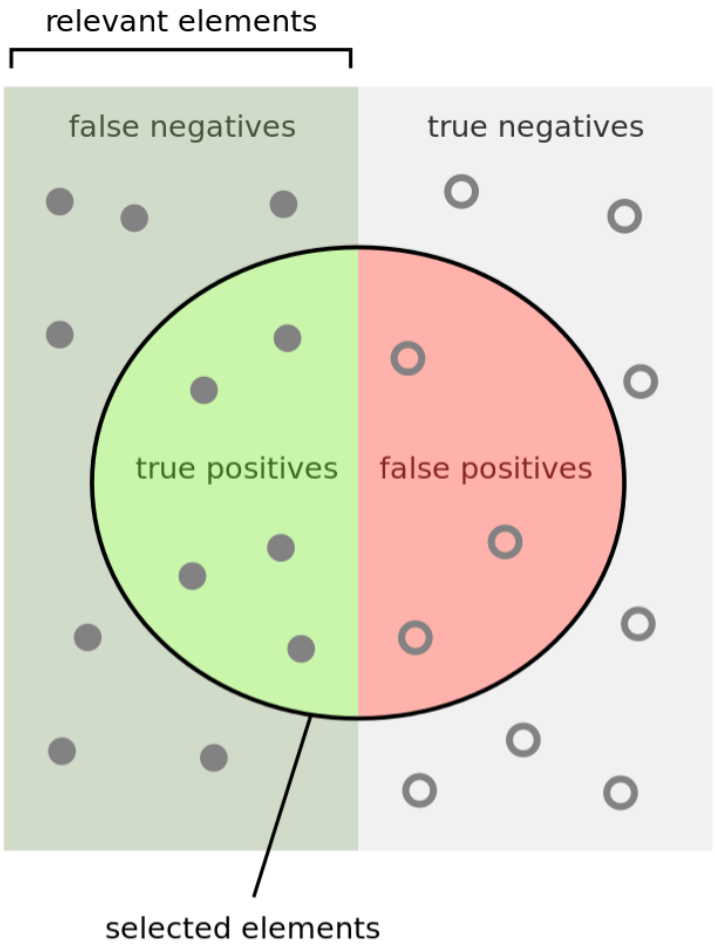
L'immagine a destra rappresenta tutti e soli i record appartenenti a CS.



Precision e recall

Sia $c : CS \mapsto \mathbb{C}$ la reale classe di ogni record $x \in CS$ e sia $\tilde{c} : CS \mapsto \mathbb{C}$ la classe assegnata dal nostro classificatore.

Data una classe, ad esempio A, dividiamo CS nei seguenti 4 insiemi:



Precision e recall

True Positive (**TP**) ovvero i record $x \in CS$ classificati **correttamente**, e la cui classe reale è A, per cui $\tilde{c}(x) = c(x) = A$

True Negative(**TN**) ovvero i record $x \in CS$ classificati **correttamente**, e la cui classe reale è B, per cui $\tilde{c}(x) = c(x) = B$

False Positive(**FP**) ovvero i record $x \in CS$ classificati **erroneamente**, e la cui classe reale è A, per cui $\tilde{c}(x) = B \neq A = c(x)$

False Negative(**FP**) ovvero i record $x \in CS$ classificati **erroneamente**, e la cui classe reale è B, per cui $\tilde{c}(x) = A \neq B = c(x)$

Precision e recall

Definiamo i record **rilevanti** tutti e soli i record in CS aventi classe reale eguale a quella investigata, che sono quindi rilevanti per la nostra valutazione. Nel nostro caso $\{x \in CS : c(x) = A\}$.

Definiamo **precision** (precisione) la frazione di elementi rilevanti tra tutti gli elementi classificati come A. $Precision = \frac{TP}{TP+FP}$.

Definiamo **recall** (recupero o sensitività) la frazione di elementi classificati come A tra tutti gli elementi rilevanti. $Recall = \frac{TP}{TP+FN}$.

Precision e recall

Più alti sono i valori di precision e recall, migliore sarà la bontà del nostro classificatore.
Tuttavia, in alcune situazioni è preferibile avere un solo indice anziché due.

L'**F_1-score** è una misura che combina precision e recall facendone una media armonica.

$$F_1 = 2\frac{Precision \cdot Recall}{Precision + Recall} = \frac{2TP}{2TP + FP + FN}$$

Come precision e recall, l'F_1-score ha un valore compreso tra 0 e 1, e più alto è il suo valore maggiore è la bontà del nostro classificatore. Tuttavia, valori prossimo ad 1 possono indicare **overfitting**.

Accuratezza (accuracy)

L'**accuratezza** (in inglese accuracy) misura la quantità di oggetti correttamente classificati rispetto al totale degli oggetti del CS. Ovvero quanto buono è il classificatore nell'identificare o escludere correttamente gli oggetti ad una determinata classe. E' definita come

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Accuratezza per data set sbilanciati

L'accuratezza è poco indicata da utilizzare in dataset sbilanciati. Ad esempio, se consideriamo un campione con 95 negativi (classe) e 5 positivi (classe), un classificatore che assegna sempre la classe negativa avrà il 95% (0.95) di accuratezza.
In questi casi si preferisce la formula dell'accuratezza bilanciata:

$$Balanced\ accuracy = \frac{TPR + TNR}{2}$$

dove:

- TPR (**True Posive Rate**) è $TPR = \frac{TP}{TP+FN}$, detta anche sensitività perché cattura quanto il classificatore è sensibile nel riconoscere una classe
- TNR (**True Negative Rate**) è $TNR = \frac{TN}{TN+FP}$, detta anche specificità perché cattura quanto il classificatore è specifico nel distinguere la classe specifica dalle altre

False discovery rate

Il **False discovery rate** (**FDR**) è una misura molto utilizzata in statistica per analizzare il tasso di errori di tipo I (errore nel rigetta una ipotesi nulla vera) quando si vuole testare una ipotesi nulla in un test con confronti multipli.
Nel caso della classificazione essa è definita come

$$FDR = \frac{FP}{FP + TP} = 1 - \frac{TP}{TP + FP}$$

Matrice di confusione e indici: resumè

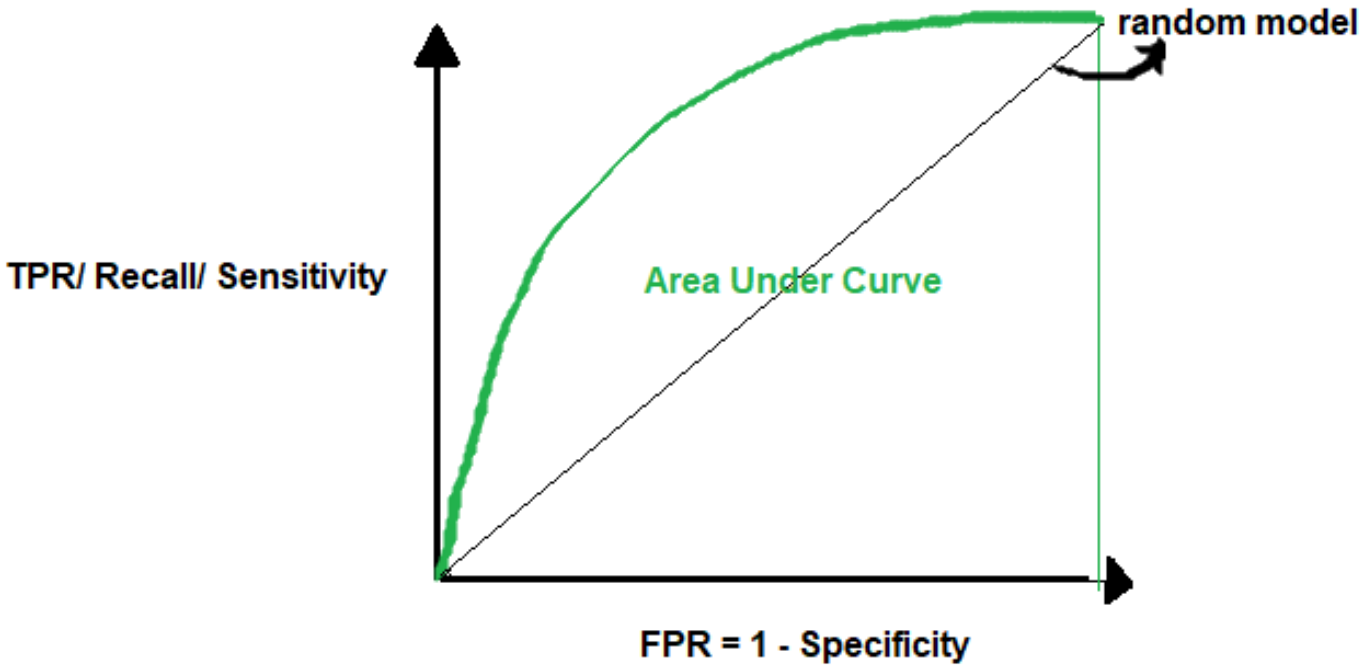
Matrice di confusione per classificatori binari e indici annessi.

		Actual class		
		Positive	Negative	
Predicted class	Positive	**TP:** True Positive	**FP:** False Positive (Type I Error)	**Precision:** $\frac{TP}{(TP + FP)}$
	Negative	**FN:** False Negative (Type II Error)	**TN:** True Negative	**Negative Predictive Value:** $\frac{TN}{(TN+FN)}$
		Recall or Sensitivity: $\frac{TP}{(TP + FN)}$	**Specificity:** $\frac{TN}{(TN + FP)}$	**Accuracy:** $\frac{TP + TN}{(TP + TN + FP + FN)}$

ATTENZIONE: sensitività, precisione, specificità e NPR sono specifiche delle casse prese in considerazione. L'accuratezza invece è invariante per classe.

AUC

Il valore di **AUC** (**Area Under the Curve**) è basato sulla curva **ROC** (curve di **Receiver Operating Characteristics** utilizzate per la prima volta durante la seconda guerra mondiale, da alcuni ingegneri elettrotecnici che volevano individuare i nemici utilizzando il radar durante le battaglie aeree), misura l'area al di sotto della curva ROC appunto.



AUC

Una curva ROC è il grafico dell'insieme delle coppie (FP, TP) al variare di un parametro del classificatore. Per esempio, in un classificatore a soglia, si calcola la frazione di veri positivi e quella di falsi positivi per ogni possibile valore della soglia; tutti i punti così ottenuti nello spazio FP-TP descrivono la curva ROC.

Attraverso l'analisi delle curve ROC si valuta la capacità del classificatore di discernere, ad esempio, tra un areo amico ed uno nemico, o tra una popolazione sana ed una malata.

Il valore di AUC, compreso tra 0 e 1, equivale alla probabilità che il risultato del classificatore applicato ad un individuo estratto a caso dal gruppo dei malati sia superiore a quello ottenuto applicandolo ad un individuo estratto a caso dal gruppo dei sani.

AUC

Le curve ROC passano per i punti (0,0) e (1,1), avendo inoltre due condizioni che rappresentano due curve limite:

- una che taglia il grafico a 45 gradi passando per l'origine. Questa retta rappresenta il caso del classificatore casuale e l'area sottesa AUC è pari a 0,5.
- la seconda curva è rappresentata dal segmento che dall'origine sale al punto (0,1) e da quello che congiunge il punto (0,1) a (1,1), avendo un'area sottesa di valore pari a 1, ovvero rappresenta il classificatore perfetto.

Classificazione a più di due classi

Le misure viste fin ora sono state sviluppate per la classificazione a due classi e spesso non sono di facile applicazione nel caso di classificatori multi-classe.
Se abbiamo un classificatore binario, possiamo calcolare precision e recall di ogni classe e avere una idea delle performance del classificatore. Ma cosa succede se abbiamo 10 o più classi? Ogni misura ci dirà quanto il classificatore è bravo nel assegnare ogni singola classe ma perderemo la visione di performance globale.

Nel caso della $F_1 - score$ possiamo mediare i singoli valori di F_1 ottenuti per le varie classi, in lamno due modi diversi, micro e macro.

Classificazione a più di due classi

Per ogni classe $g_i \in G = \{1, \ldots, K\}$, costruiamo una matrice di confusione centrata sulla classe g_i. Ovvero dove g_i è il caso positivo e tutte le altre classi le uniamo in un'unica grande classe che rappresenta il caso negativo.

Per ogni classe possiamo quindi calcolare gli specifici valori TP_i, FP_i e FN_i.

Micro average

La **micro average** prende il nome dal fatto che riassume le prestazioni sull'unità più piccola possibile (ovvero su tutti i campioni). Quindi

$$P_{micro} = \frac{\sum_{i=1}^{|G|} TP_i}{\sum_{i=1}^{|G|} (TP_i + FP_i)}$$

$$R_{micro} = \frac{\sum_{i=1}^{|G|} TP_i}{\sum_{i=1}^{|G|} (TP_i + FN_i)}$$

da cui

$$F1_{micro} = 2\frac{P_{micro} \cdot R_{micro}}{P_{micro} + R_{micro}}$$

Più grande è il valore migliore è il classificatore. Tuttavia questa misura non è sensibile alle prestazioni predittive per le singole classi. Di conseguenza, la micro-media può essere particolarmente fuorviante quando la distribuzione per classi è sbilanciata.

Macro average

La **macro average** prende il nome dal fatto che fa la media su gruppi più grandi, cioè sulla performance per le singole classi piuttosto che sulle osservazioni). Quindi:

$$P_{macro} = \frac{1}{|G|} \sum_{i=1}^{|G|} \frac{TP_i}{(TP_i + FP_i)} = \frac{\sum_{i=1}^{|G|} P_i}{|G|}$$

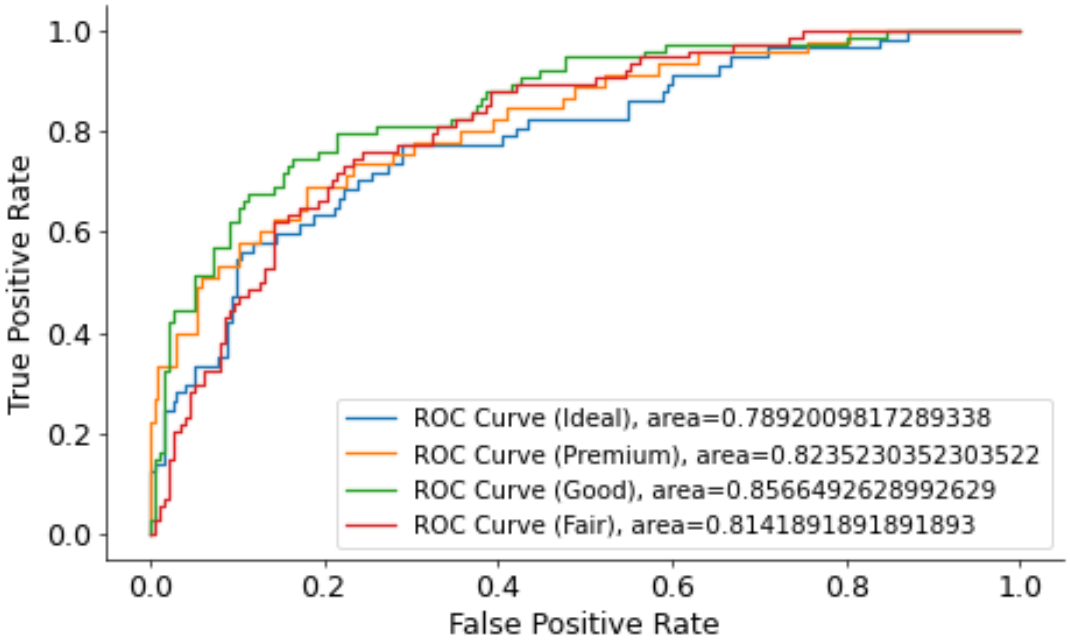
$$R_{macro} = \frac{1}{|G|} \sum_{i=1}^{|G|} \frac{TP_i}{(TP_i + FN_i)} = \frac{\sum_{i=1}^{|G|} R_i}{|G|}$$

da cui

$$F1_{macro} = 2 \frac{P_{macro} \cdot R_{macro}}{P_{macro} + R_{macro}}$$

Se ha un valore grande, questo indica che un classificatore funziona bene (in media) per ogni singola classe. La macro-media è quindi più adatta per dati con una distribuzione per classi sbilanciata.

Generalizzazione di AUC



(Metodo Hand & Till, Machine Learning, 45, 171–186, 2001)

Assumiamo che le classi siano etichettate come $0, 1, 2, \ldots, c-1$ con $c > 2$ e consideriamo le coppie di classi (i, j).
Un buon classificatore assegna con alta probabilità la classe corretta e con bassa probabilità la classe sbagliata.

Generalizzazione di AUC

Sia $\hat{A}(i|j)$ la probabilità che un elemento scelto a caso per la classe j abbia una probabilità di appartenere alla classe i inferiore rispetto a un elemento scelto a caso dalla classe i stessa. Sia $\hat{A}(j|i)$ definita di conseguenza.

Possiamo calcolare $\hat{A}(i|j)$ usando le seguenti definizioni:

- $\hat{p}(X_l)$ è la probabilità stimata che l'osservazione x_l sia originata dalla classe i
- per tutte le osservazioni x_l della classe i, sia $f_l = \hat{p}(x_l)$ la probabilità stimata di appartenere alla classe i
- per tutte le osservazioni x_l della classe j, sia $g_l = \hat{p}(x_l)$ la probabilità stimata di appartenere alla classe i

Generalizzazione di AUC

Ordiniamo i valori ottenuti $\{g_1, \ldots, g_{n_j}, f_1, \ldots, f_{n_i}\}$ in modo crescente.

Sia r_l il rango della l-esima osservazione dalla classe i.

Allora, il numero totale di coppie di punti in cui il punto della classe j ha un valore di probabilità stimato di appartenere alla classe i minore del punto di classe i è dato da

$$\sum_{l=1}^{n_i}(r_l - l) = \sum_{l=1}^{n_i} r_l - \sum_{l=1}^{n_i} l = S_i - \frac{n_i(n_i+1)}{2}$$

dove S_i è la somma rei ranghi delle osservazioni appartenenti alla classe i.

Generalizzazione di AUC

Siccome ci sono coppie di punti del tipo $n_0 n_1$ dalle due classi, la probabilità che un punto della classe j scelto a caso abbia una probabilità stimata di appartenere alla classe i inferiore di un punto scelto a caso da i, è pari a

$$\hat{A}(i|j) = \frac{S_i - n_i(n_i + 1)/2}{n_i n_j}$$

Siccome non possiamo distinguere $\hat{A}(i|j)$ da $\hat{A}(j|i)$, definiamo

$$\hat{A}(i|j) = \frac{\hat{A}(i|j) + \hat{A}(j|i)}{2}$$

come la misura di separabilità delle classi i e j.

Generalizzazione di AUC

Il valore di AUC globale di un classificatore multi-classe è quindi dato dalla media dei valori di $\hat{A}(i|j)$:

$$M = \frac{2}{c(c-1)} \sum_{i<j} \hat{A}(i|j)$$

Applichiamo il fattore $\frac{2}{c(c-1)}$ perché ci sono $c(c-1)$ modi diversi con cui costruire coppie distinte tenendo conto i possibili differenti ordini. Inoltre, siamo interessati solo a metà di tali coppie.

Generalizzazione di AUC

Esempio.
Consideriamo due classi i e j tale che $n_j = 6$ ed $n_i = 7$.
Calcoliamo le probabilità condizionali:
$\hat{f}_i(x_l|y_l = j) = \{0.42, 0.15, 0.30, 0.01, 0.04, 0.23\}$
$\hat{f}_i(x_l|y_l = i) = \{0.89, 0.67, 0.39, 0.57, 1, 0.96, 0.92\}$

Ordiniamo:
$0.01, 0.04, 0.15, 0.23, 0.30, 0.39(6),$
$0.42, 0.57(8), 0.67(9), 0.89(10), 0.92(11), 0.96(12), 1(13)$

Sommiamo i ranghi per $y_l = i$
$Si = \sum_{l=1}^{n_1} r_l = 6 + 8 + 9 + 10 + 11 + 12 + 13 = 69$

Quindi calcoliamo la probabilità che un punto della classe j abbiamo un probabilità stimata inferiore ad un punto della classe i
$\hat{A}(i|j) = \frac{S_i - n_i(n_i+1)/2}{n_i n_j} = \frac{69-(7\cdot 8)/2}{7\cdot 6} = 0.976$

Cross-validazione

La **cross-validazione** (**cross-validation**) è una tecnica per validare modelli statistici per valutare come i risultati di un'analisi statistica si generalizzeranno a un insieme di dati indipendente.

L'obbiettivo principale è quello di testare la capacità del modello di prevedere nuovi dati che non sono stati utilizzati durante la fase di addestramento o creazione del modello di classificazione.

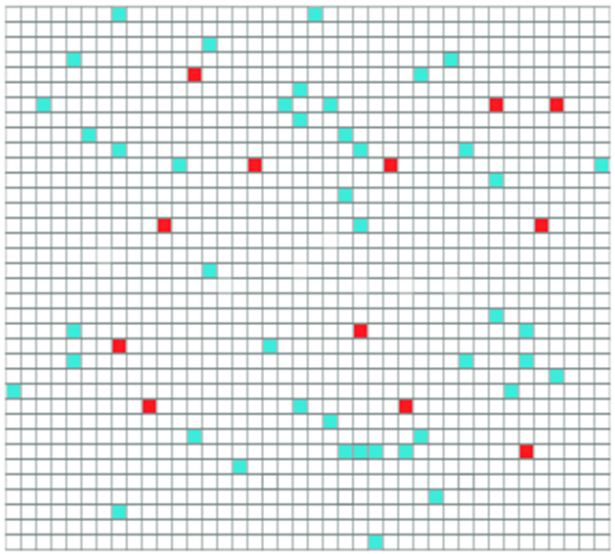
Serve principalmente a valutare problemi di overfitting o di **selection bias**.

Può anche essere utilizzata per decidere quale tra i modelli di classificazione disponibili è quello più adatto ai nostri scopi. In questo caso è importante che tutti i metodi vengano verificati sugli stessi TS+CS!!!

Selection bias

Il **selection bias** si verifica quando le scelta del training set per l'addestramento risulta essere viziata da fattori esterni piuttosto che rispecchiare un modello di selezione non di parte.

TS e CS dovrebbero essere prodotti infatti tramite **campionamento uniforme** dell'universo delle possibili osservazioni.



Uniform sampling

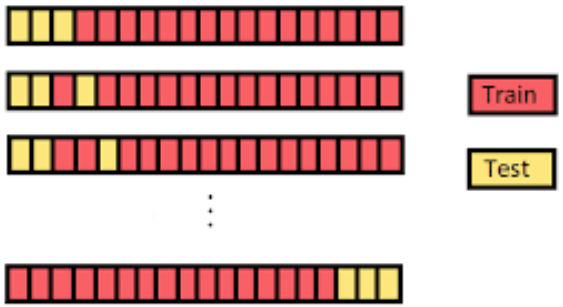
Cross-validazione

La **cross validazione** si divide in

- **esaustiva** che fa addestramento e verifica su tutti i possibile modi di dividere il data set in TS e CS
- **non esaustiva** in cui solo un sotto insieme dei possibili modi di divisione è preso in considerazione. Generalmente i sottoinsiemi sono scelti a caso dal data set

Cross-validazione esaustiva

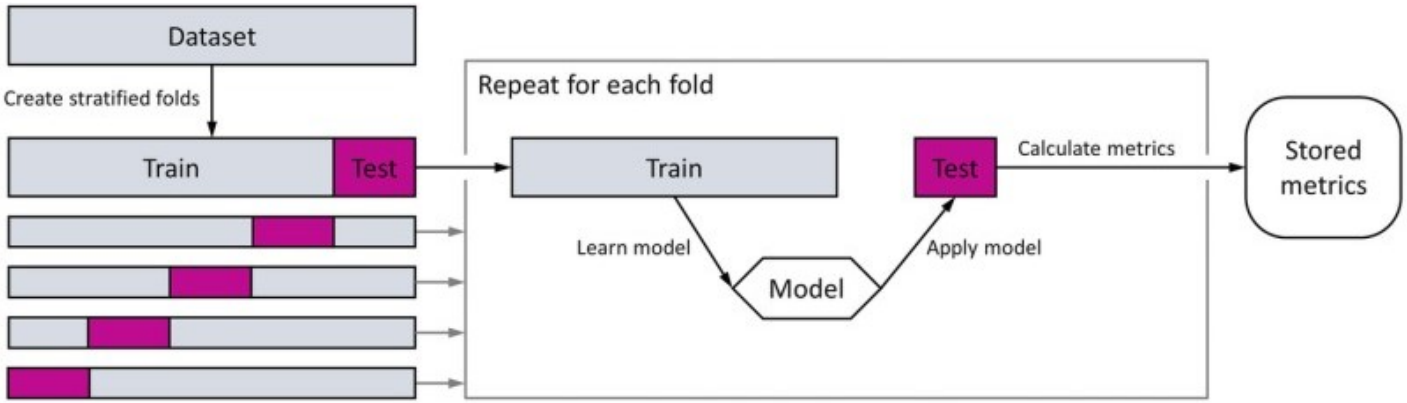
Dato un data set (DS) di N osservazioni, la **leave-p-out cross-validation** utilizza p osservazioni come CS e le rimanenti $N - p$ come TS.
La divisione in CS e TS viene ripetuta per tutti i possibili tagli. Questo implica che il modello verrà testato $\binom{n}{p}$ volte.



Per $p = 2$, viene utilizzata come come metodo unbiased per stimare la AUC in classificatori binari.
Molto utilizzata è anche la **leave-one-out** che risulta appropriata quando si dispone di un piccolo set di dati o quando una stima accurata delle prestazioni del modello è più importante del costo computazionale del metodo.

Cross-validazione non esaustiva

La più utilizzata tecnica di cross validazione non esaustiva è la **k-fold cross-validation**. In questa tecnica il DS viene diviso in k parti di egual dimensione. Quindi, a turno, ogni parte viene utilizzata come CS, mentre le restanti parti vengono tutte utilizzate come TS.



E' importante che ogni fold sia unbiased. Un metodo abbastanza utilizzato è mescolare in modo molto casuale il DS e verificare che le osservazioni delle varie classi sia distribuite in modo più o meno uniforme nel DS.