UNIVERSITA' DEGLI STUDI DI PADOVA

FACOLTA' DI SCIENZE STATISTICHE

CORSO DI LAUREA IN STATISTICA E GESTIONE DELLE IMPRESE

TESI DI LAUREA

IMPLEMENTAZIONE DELLA CURVA ROC IN R

CON APPLICAZIONE A CASI DI STUDIO

Relatore: Ch.ma Prof.ssa Laura Ventura

Laureando: Federico Rosina

Matricola: 579101

ANNO ACCADEMICO 2010-2011

Alla mia famiglia

Indice

INTRODUZIONE

La curva di ROC (*Receiver operating characteristic curve*) è una tecnica statistica attualmente utilizzata in una grande varietà di campi scientifici.

Questa tecnica trae origine nell'ambito della teoria della rivelazione del segnale. Si tratta di una metodologia che è stata utilizzata per la prima volta da alcuni ingegneri, durante la seconda guerra mondiale, per l'analisi delle immagini radar e lo studio del rapporto segnale/disturbo. Più tardi, tra il 1970 e il 1980, diventò evidente l'importanza dell'utilizzo di questa tecnica nella valutazione dei test diagnostici, in campi quali la radiologia, cardiologia, chimica clinica ed epidemiologia. Recentemente è entrata anche nell'ambito del *data mining* (Metz, 1978; Pepe, 2003; Zou, 2002).

L'uso così estensivo di questa tecnica può essere spiegato grazie alla sua relativa semplicità di costruzione, e per la sua facile applicazione come tecnica di valutazione della bontà dei test discriminatori.

In base alla tipologia di responso fornito, i test si possono dividere in qualitativi e quantitativi. I primi restituiscono un output di tipo dicotomico (vero/falso, si/no, positivo/negativo), i secondi producono risultati sotto forma di variabili numeriche di tipo discrete o continue. Per i test di tipo quantitativo è necessario individuare un valore di soglia che permetta di discriminare i risultati in "positivi" e "negativi", e questo valore viene chiamato *cut-off* (*cut-point*, *threshold*), cioè quel valore assunto dalla variabile misurata nel test al di sopra del quale il soggetto viene dichiarato positivo e al di sotto del quale viene definito negativo.

Prendendo come esempio un test medico in realtà accade che, quando si sottopone un campione ad una procedura diagnostica, per un determinato valore di *cut-off*, non tutti i soggetti malati risulteranno positivi al test, così come non tutti i soggetti sani risulteranno negativi. Questo genera incertezza nell'interpretazione del test perché, nella maggioranza dei casi, esiste una zona di sovrapposizione dei

risultati del test applicato a pazienti sani e a pazienti malati. In particolare si possono ottenere queste tre situazioni:

1. il test ideale dovrebbe consentire di discriminare completamente tra pazienti sani e malati come mostrato nella Figura 1. In questo caso è immediato individuare sull'asse delle ascisse il valore di *cut-off* che permette di discriminare, in questo caso e solo in questo con precisione assoluta, tra pazienti sani e pazienti malati;

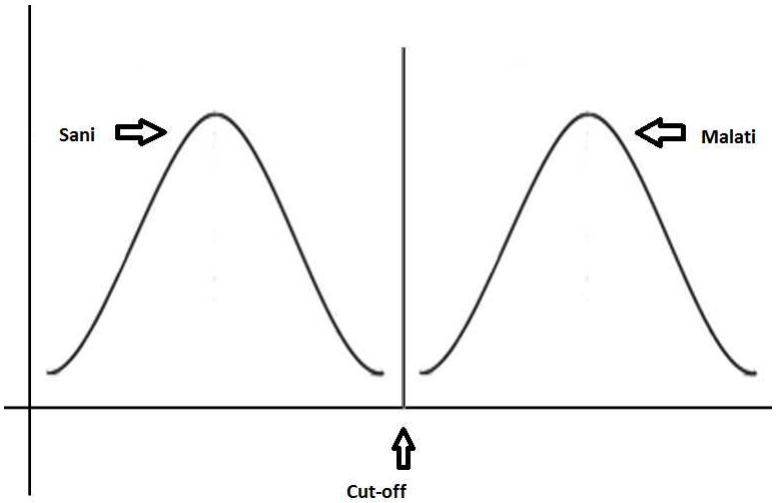

Figura 1

2. il caso opposto è quello mostrato nella Figura 2. I risultati non si possono interpretare e non si riesce ad attribuire il paziente al gruppo dei sani o al gruppo dei malati. Si dice che il test non ha potere diagnostico (o di classificazione);

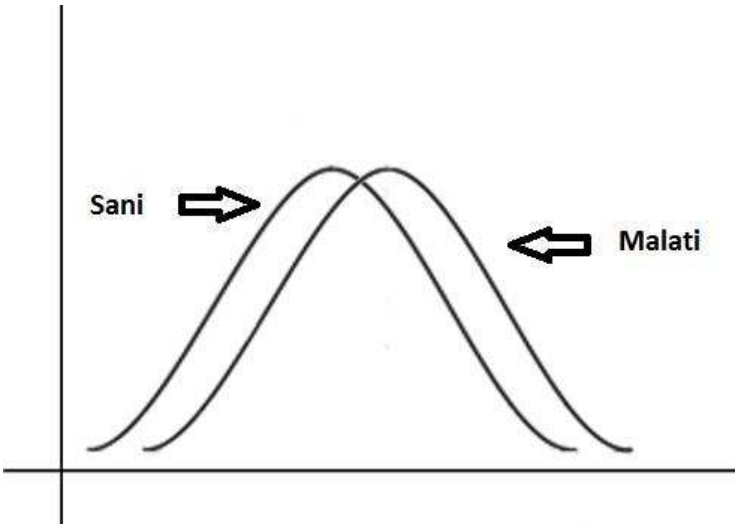

Figura 2

3. nella pratica si verifica sempre una sovrapposizione più o meno ampia delle due distribuzioni, come mostrato nella Figura 3. Si avrà sempre un certo numero di soggetti sani che risulteranno positivi al test ("falsi positivi", FP), e un certo numero di soggetti malati che erroneamente verranno classificati come sani ("falsi negativi", FN). Dunque, nella realtà, è impossibile individuare sull'asse delle ascisse un valore di *cut-off* che consenta una classificazione perfetta, ossia da azzerare i falsi positivi e i falsi negativi.

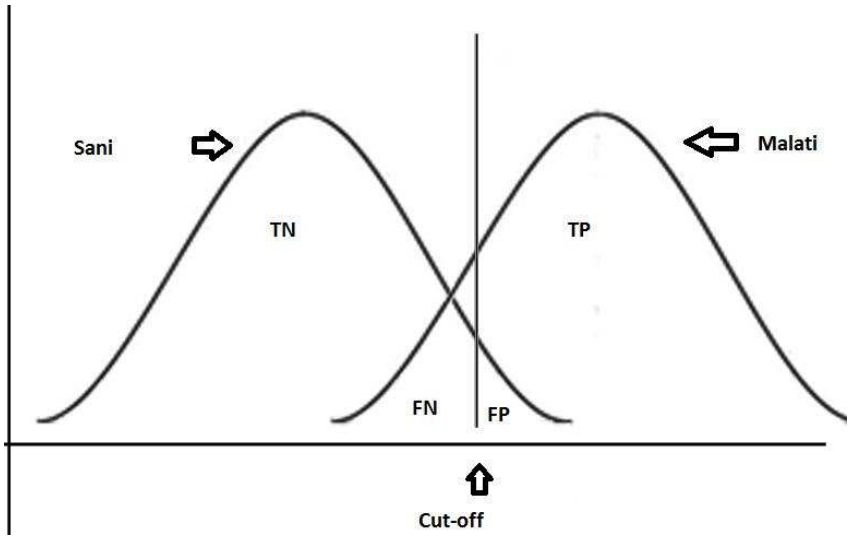

Figura (3)

Per semplicità, nella trattazione si farà sempre riferimento a test applicati per determinare lo stato di "salute" o "malattia" di un singolo paziente. Tuttavia questa analisi può essere utilizzata anche per test quantitativi capaci di accertare qualsiasi altra condizione (discriminare tra rumore e segnali contenenti informazioni, classificazione binaria dei dati e successiva possibilità di valutare la qualità del risultato prodotto etc.).

Una tecnica ampiamente utilizzata per valutare il comportamento di un test diagnostico in una popolazione è l'analisi tramite la curva ROC. Infatti, attraverso la costruzione della curva ROC e del calcolo dell'area sottesa ad essa (AUC, *Area under the ROC curve*), è possibile stimare la probabilità di assegnare un'unità statistica al suo reale gruppo di appartenenza e quindi valutare la bontà del metodo usato per la classificazione.

Nei capitoli successivi verranno presentati i seguenti argomenti. Il primo capitolo presenterà alcuni concetti generali riguardanti la teoria della curva ROC e della valutazione della capacità discriminatoria di un test attraverso l'indicatore AUC. Il secondo capitolo tratterà del *software* statistico R (http://www.r-project.org/) e della possibilità di analisi della curva ROC in R. Nell'ultimo capitolo si stimerà la curva ROC e le quantità ad essa collegate, sulla base di dati riguardanti dei casi di studio, e si analizzeranno i risultati.

Capitolo 1. La curva ROC

In questo capitolo presenteremo come valutare la bontà di un modello di classificazione. Verranno presentati la matrice di confusione, la curva ROC e l'indice AUC per confrontare i risultati attesi con quelli ottenuti. Per una trattazione approfondita di tali argomenti si rinvia a Krzanowski, Hand (2009).

1.1 La matrice di confusione: sensibilità e specificità

Un classificatore può essere decritto come una funzione che mappa gli elementi di un insieme in certe classi o gruppi. In un caso medico, vengono confrontati l'output del test con il vero stato del paziente. Quest'ultimo può essere già noto in partenza, oppure può essere stabilito mediante quello che viene definito un *golden test*, ossia un test con una alta attendibilità. Quindi i risultati del *golden test* corrispondono alla realtà.

In un problema di classificazione binaria l'insieme dei dati da classificare è suddiviso in due classi che possiamo indicare convenzionalmente in positivi P e negativi N (ad esempio malati e sani). Gli esiti predetti dal classificatore binario li indicheremo con positivi "p" e negativi "n", rispettivamente. Sono possibili quattro risultati a seconda del valore di *cut-off*:

- il classificatore produce il valore "p" partendo da un dato appartenente alla classe P. Si dice che il classificatore ha prodotto un vero positivo (TP);

- il classificatore produce il valore "p" partendo da un dato appartenente alla classe N. Si dice che il classificatore ha prodotto un falso positivo (FP);

- il classificatore produce il valore “n” partendo da un dato appartenente alla classe N. Si dice che il classificatore ha prodotto un vero negativo (TN);

- il classificatore produce il valore “n” partendo da un dato appartenente alla classe P. Si dice che il classificatore ha prodotto un falso negativo (FN).

I quattro valori identificati (TP, FP, TN, FN) possono essere rappresentati in una tabella a doppia entrata che conta il numero di unità classificate correttamente o meno per ciascuna delle due modalità possibili. Questa matrice è detta matrice di confusione (o tabella di errata classificazione); si veda la Tabella 1.

		Valori predetti	
		n	***p***
Valori reali	**N**	Veri negativi **(TN)**	Falsi positivi **(FP)**
	P	Falsi negativi **(FN)**	Veri positivi **(TP)**

Tabella 1. Matrice di confusione

I numeri sulla diagonale della matrice di confusione rappresentano le unità statistiche correttamente classificate; gli altri sono gli errori.

A partire da tale classificazione, si possono ottenere due importanti indici sintetici della qualità della classificazione: la sensibilità e la specificità. La sensibilità è definita come

$$\text{Sensibilità} = \text{Se} = \frac{TP}{TP+FN} ,$$

ed esprime la proporzione di Veri Positivi rispetto al numero totale di positivi effettivi.

La sensibilità è condizionata negativamente dalla quota di falsi negativi: pertanto un test molto sensibile dovrà associarsi ad una quota molto bassa di falsi negativi, ovvero di soggetti malati che "sfuggono" all'identificazione attraverso il test. Il calcolo della sensibilità considera esclusivamente la popolazione dei malati (ovvero la seconda riga della Tabella 1), in funzione dell'identificazione come positivi e negativi del test.

La specificità è definita come

$$\text{Specificità} = \text{Sp} = \frac{TN}{FP+TN} ,$$

ed esprime la proporzione di Veri Negativi rispetto al numero totale di negativi effettivi. La specificità è influenzata in particolare dalla quota di falsi positivi; ovvero un test sarà tanto più specifico quanto più bassa risulterà la quota dei falsi positivi, cioè di soggetti sani identificati dal test come malati. Un test molto specifico consente di limitare la possibilità che un soggetto sano risulti positivo al test. Per calcolare la specificità si fa riferimento esclusivamente al gruppo dei sani ed alla loro distribuzione fra positivi e negativi al test (ovvero la prima riga della Tabella 1). Un test altamente specifico sarà dunque un test che produrrà una bassa quota di falsi positivi.

Si dice che il test è sensibile al 100% quando tutti i malati sono risultati positivi; si dice che il test è specifico al 100% quando tutti i sani risultano negativi.

E' facile verificare che i valori di sensibilità e specificità sono fra loro inversamente correlati in rapporto alla scelta del valore di *cut-off*. Infatti, modificando quest'ultimo, si può ottenere uno dei seguenti effetti:

- aumento della sensibilità e diminuzione della specificità;

- aumento della specificità e diminuzione della sensibilità.

E' possibile dimostrare che, quando la distribuzione dei valori delle due classi malati-sani è di tipo normale, la "soglia discriminante ottimale", ossia il valore di *cut-off* che minimizza gli errori di classificazione, è pari al valore in ascissa corrispondente al punto di intersezione delle due distribuzioni (Bottarelli, Parodi, 2003). Tuttavia, la scelta di tale valore non si può basare solo su teorie probabilistiche. Ad esempio, nel caso di malattie ad alta contagiosità, potrebbe essere opportuno minimizzare la quota di falsi negativi e quindi privilegiare la sensibilità a scapito della specificità. Nel caso contrario, si privilegia la specificità a scapito della sensibilità.

A tali difficoltà è da sovrapporre un ulteriore elemento che ostacola sia la scelta del *cut-off* ottimale per un singolo test che il raffronto fra le performance di test diversi. Tale elemento è costituito dal fatto che i valori predittivi (Appendice 1) dipendono, oltre che dalla Se e Sp del test, anche dalla prevalenza della malattia nella popolazione studiata. Infatti è intuitivo che, all'aumentare della frazione dei malati nel campione sottoposto al test, la proporzione dei malati positivi aumenti nell'insieme dei positivi al test. Al contrario, per una patologia poco rappresentata, tenderà ad aumentare la frazione dei falsi positivi sul totale dei positivi al test. Nel complesso, tali osservazioni comportano tre importanti implicazioni:

1. è possibile scegliere un valore di *cut-off* che corrisponda ad un predeterminato valore di Se o di Sp, ma non è detto che tale valore sia ottimale per gli scopi contingenti;

2. la Se e la Sp associate ad un singolo valore di *cut-off* non rappresentano descrittori esaurienti della performance del test potenzialmente ottenibile adottando altri valori di *cut-off*;

3. i valori predittivi, in quanto dipendenti dalla prevalenza della malattia nella popolazione studiata, non sono caratteristiche intrinseche del test e quindi non possono essere utilizzati come descrittori esaurienti della performance dei test.

Le problematiche appena accennate, ad eccezione della 3, possono essere risolte con l'analisi della curva ROC (Bottarelli, Parodi, 2003).

1.2 La curva ROC

Il modello di classificazione sarebbe ottimale se massimizzasse contemporaneamente sia la sensibilità che la specificità. Questo tuttavia non è possibile: infatti elevando il valore della specificità, diminuisce il valore di falsi positivi , ma si aumentano i falsi negativi, il che comporta una diminuzione della sensibilità. Si può quindi osservare che esiste un *trade-off* tra i due indici.

La relazione tra i suddetti parametri può essere rappresentata attraverso una linea che si ottiene riportando, in un sistema di assi cartesiani e per ogni possibile valore di *cut-off*, la proporzione di veri positivi in ordinata e la proporzione di falsi positivi in ascissa. Se il risultato del test è calcolato su scala continua, si possono calcolare i valori di Se e il complemento a uno della specificità, 1-Sp (probabilità di ottenere un falso positivo nella classe dei non-malati). L'unione dei punti ottenuti riportando nel piano cartesiano ciascuna coppia di Se e 1-Sp genera una spezzata, la curva ROC (Figura 4).

Piccoli spostamenti lungo la curva informano sulle variazioni reciproche di sensibilità e di specificità per piccole variazioni del *cut-off.* In questo senso è importante la pendenza locale della curva; ad esempio, una forte pendenza significa un buon incremento di sensibilità con piccola perdita di specificità.

Un test perfetto dal punto di vista discriminatorio (assenza di sovrapposizione tra i due gruppi) è rappresentato da una curva ROC che passa per l'angolo superiore sinistro degli assi cartesiani (massima specificità e sensibilità). Al contrario, la curva ROC per un test assolutamente privo di valore informativo è rappresentata dalla bisettrice ("*chance line*") (Bamber, 1975; Zweig, Campbell, 1993).

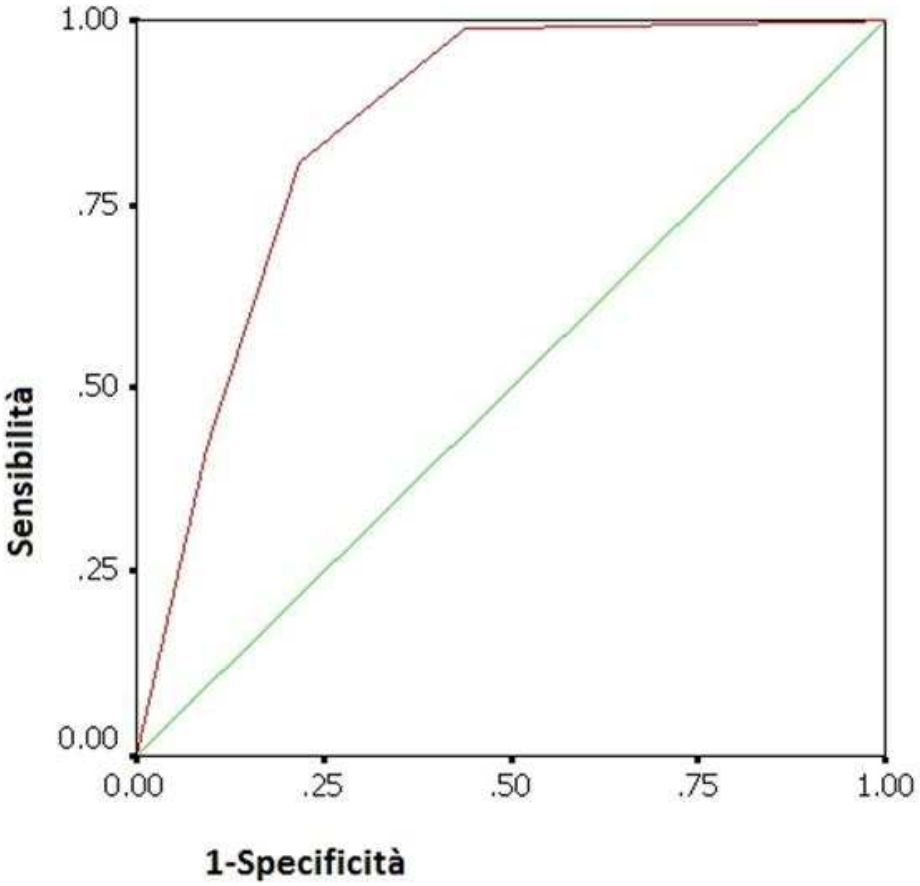

Figura 4.Grafico della curva ROC. La linea verde rappresenta la "*chance line*".

1.3 Area sottesa alla curva ROC

La curva ROC può essere utilizzata per confrontare test diagnostici diversi che fanno riferimento ad uno stesso gruppo di unità statistiche.

Uno degli indici più utilizzati per valutare la bontà della regola di classificazione è l'AUC (*Area under the ROC curve*). Il calcolo dell'AUC per una curva empirica (Figura 4), può essere effettuata semplicemente connettendo i diversi punti della curva ROC all'asse delle ascisse con segmenti verticali e sommando le aree dei risultanti poligoni generati nella zona sottostante. Questa tecnica, detta regola trapezoidale, può fornire però risultati distorti.

Sia X la variabile che rappresenta la misura nel gruppo dei pazienti sani e Y quella nel gruppo dei

pazienti malati. Una seconda possibilità prevede di esprimere l'AUC come

$$\text{AUC}= \text{P}(X<Y).$$

Questa espressione è anche nota come modello sollecitazione-resistenza (*stress-strenght model*) (vedi, ad esempio, Kotz. et al., 2003). Si possono fare varie assunzioni parametriche e non parametriche sulle variabili *X* e *Y*.

Sotto ipotesi non parametriche, la quantità AUC è collegata alla statistica test U di Mann-Whitney (Bottarelli, Parodi, 2003). La statistica U, rappresenta una delle più note tecniche di statistica non parametrica. Viene utilizzata per il confronto della distribuzione di una variabile continua tra due gruppi e per testare l'ipotesi nulla che i due gruppi presentino la stessa mediana. Tale ipotesi è equivalente a testare che un soggetto estratto a caso da un gruppo *X* abbia la stessa probabilità di presentare un valore della variabile inferiore ad un valore predefinito di quello di un soggetto estratto a caso dall'altro gruppo *Y* (Bottarelli, Parodi, 2003).

La stima campionaria di U è

$$U= n_1 n_2 + \frac{n_1(n_1+1)}{2} - R_1 ,$$

con n_1 e n_2 numerosità campionarie dei due gruppi e R_1 somma dei ranghi del gruppo con numerosità n_1. Il suo valore atteso, sotto l'ipotesi nulla, è

$$E(U)= \frac{n_1 n_2}{2}.$$

Bamber (1975) ha mostrato l'equivalenza tra l'area sottesa ad una curva ROC, costruita per dati su

scala continua, e la statistica U. La relazione che lega i due parametri è la seguente

$$AUC=1-\frac{U}{n_1 n_2},$$

da cui segue che (sempre sotto l'ipotesi nulla)

$$\mu_{AUC}= E[AUC]= 0.5.$$

Sotto ipotesi parametriche, si assume in genere che X e Y siano variabili casuali con funzioni di densità, rispettivamente, $f_x(x; \Theta_x)$ e $f_y(y; \Theta_y)$, con $\Theta_x \in \Theta_x \subseteq R^{px}$ *e* $\Theta_y \in \Theta_y \subseteq R^{py}$. L'AUC può essere quindi espressa come

$$AUC = AUC(\Theta)= \quad P(X<Y)=\int_{-\infty}^{+\infty} F_x(t; \Theta_x)\, dF_y(t; \Theta_y),$$

con $F_x(t; \Theta_x)$ e $F_y(t; \Theta_y)$ funzioni di ripartizione di X e Y, rispettivamente, e $\Theta = (\Theta_x , \Theta_y)$.

Esistono diverse espressioni dell'AUC a seconda delle distribuzioni delle due variabili. Nel caso più comune, in cui le variabili X e Y si distribuiscono come $X\sim N(\mu_x ,\sigma^2_x)$ e $Y\sim N(\mu_y ,\sigma^2_y)$, si ottiene

$$AUC = AUC(\Theta)= P(X<Y)= \phi\left(- \frac{\mu_x - \mu_y}{\sqrt{\sigma_x^2+ \sigma_y^2}} \right),$$

con $\Theta =(\mu_x, \mu_y, \sigma^2_x, \sigma^2_y)$.

Oppure, se $X\sim Exp(\alpha)$ e $Y\sim Exp(\beta)$ si ottiene

$$AUC = AUC(\Theta) = P(X<Y) = \frac{\alpha}{\alpha+\beta} \ ,$$

con $\Theta = (\alpha,\beta)$.

Una stima parametrica dell'AUC si può ottenere utilizzando la proprietà di equivarianza degli stimatori di massima verosomiglianza (cfr. ad esempio Pace, Salvan, 2001, Cap 3). Sia $\hat{\theta}$ la stima di massima verosomiglianza di Θ, ottenuta massimizzando la funzione di verosomiglianza completa

$$L(\Theta) = L(\Theta_x, \Theta_y) = \prod_{i=1}^{nx} f_x(x_i;\theta_x) \prod_{j=1}^{ny} f_y(y_j;\theta_y) \ ,$$

con $x = (x_1, \ldots, x_{nx})$ e $y = (y_1, \ldots, y_{ny})$ campioni casuali semplici di numerosità n_x e n_y tratti, rispettivamente, da X e Y, variabili casuali indipendenti.

La stima di massima verosomiglianza dell'AUC è

$$\widehat{AUC} = AUC(\hat{\theta}) = AUC(\hat{\theta}_x, \hat{\theta}_y).$$

Nel caso di un test perfetto, ossia che non restituisce alcun falso positivo né falso negativo (capacità discriminante = 100%), la curva ROC passa attraverso le coordinate (0,1) ed il valore dell'AUC corrisponde all'area dell'intero quadrato delimitato dai punti di coordinate (0,0), (0,1), (1,0), (1,1), che assume valore 1, corrispondendo ad una probabilità del 100% di una corretta classificazione. Al contrario la curva ROC per un test assolutamente privo di valore informativo è rappresentata dalla bisettrice ("*chance line*"), con AUC =0.5.

Per l'interpretazione del valore dell'AUC, si può tenere presente la classificazione della capacità discriminante di un test proposta da Swets (1988). Essa è basata su criteri largamente soggettivi ed avviene secondo lo schema seguente:

- AUC=0.5 test non informativo
- 0.5<AUC≤0.7 test poco accurato
- 0.7<AUC≤0.9 test moderatamente accurato
- 0.9<AUC≤1.0 test altamente accurato
- AUC=1.0 test perfetto

In alcuni casi si potrebbe essere interessati al calcolo dell'area sottesa alla curva ROC per un intervallo di specificità minore rispetto all'intero intervallo. In questo caso si parla di PAUC (*Partial Area under the ROC curve*). Sembrerebbe un modo perfettamente ragionevole di restringere l'attenzione su un intervallo di specificità, che indicheremo con *(a,b)*. Tuttavia, il problema è che sia il valore massimo che il valore minimo di PAUC dipendono dall'intervallo considerato, il che rende difficile l'interpretazione del PAUC per valutare la capacità discriminante di un test. Questo problema di interpretazione può essere risolto utilizzando la correzione di McClish (1989). Con questa correzione, il PAUC assume valori compresi tra 0.5 e 1, per qualsiasi intervallo *(a,b)*, il che rende possibile utilizzare, ad esempio, la classificazione proposta da Swets (1988) per valutare la capacità di classificazione di un test, utilizzando il PAUC al posto dell'AUC.

La correzione di McClish è la seguente

$$\frac{1}{2}\left(1 + \frac{PAUC(a,b) - m}{M - m}\right),$$

con $M = (b-a)$ e $m = \frac{(b-a)(b+a)}{2}$.

1.4 Valutazione della performance di un singolo test

L'area sottesa alla curva ROC rappresenta un parametro fondamentale per la valutazione della performance di un test, in quanto costituisce una misura dell'accuratezza non dipendente dalla prevalenza. Poiché l'AUC rappresenta una stima da popolazione campionaria finita, risulta quasi sempre necessario testare la significatività della capacità discriminante del test, ovvero se l'area sotto la curva ROC eccede significativamente il suo valore atteso nullo 0.5. Tale procedura corrisponde a verificare se la proporzione dei veri positivi è superiore a quella dei falsi positivi (Bottarelli, Parodi, 2003).

In generale, l'AUC può essere considerata una variabile normale, per cui si può costruire un test alla Wald come

$$Z=\frac{\widehat{AUC}-0.5}{\sqrt{\sigma^2_{AUC}}} \ ,$$

dove σ^2_{AUC} è la varianza di $\widehat{AUC}$.

Secondo Hanley e McNeil (1983), sotto ipotesi non parametriche, la varianza di $\widehat{AUC}$ può essere stimata con la seguente formula

$$\sigma^2_{AUC}= \frac{\widehat{AUC}\ (1-\widehat{AUC})+ (n_1-1)(Q_1-\widehat{AUC}^2)+ (n_2-1)(Q_2-\widehat{AUC}^2)}{n_1\, n_2} ,$$

dove n_1 e n_2 rappresentano la numerosità dei due gruppi a confronto e Q_1 e Q_2 sono date da

$$Q_1 = \frac{\widehat{AUC}}{2-\widehat{AUC}} \qquad \text{e} \qquad Q_2 = \frac{2\widehat{AUC}^2}{1+\widehat{AUC}} \ .$$

Se, ad esempio, il valore di |Z| eccede il valore critico 1.96, si può affermare che a livello α=0.05 il test diagnostico presenta una performance significativamente superiore rispetto ad un test non discriminante.

Attraverso la stima non parametrica di σ^2_{AUC}, è possibile ottenere intervalli di confidenza per l'AUC. L'intervallo di confidenza a livello nominale (1-α) per l'AUC è dato da

$$\widehat{AUC} \ \pm \sigma_{AUC}\, z_{1-\alpha/2} \ .$$

1.5 Comparazione di due test mediante l'analisi della curva di ROC

Due test possono essere comparati tra loro confrontando le stime dell'area sottesa alle corrispondenti curve ROC (Figura 5).

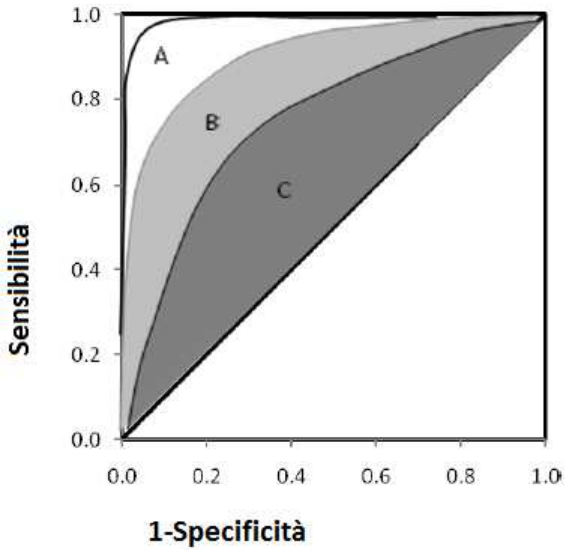

Figura 5. Confronto tra tre test diagnostici mediante analisi ROC. Risulta evidente la superiorità discriminatoria del test A rispetto ai test B e C.

Un test alla Wald può essere eseguito rapportando la differenza di due aree all'errore standard di tale differenza.

Un test (Hanley, McNeil, 1983) per il confronto tra due curve ROC indipendenti (i test a confronto sono stati applicati a gruppi di soggetti diversi) è dato da

$$Z= \frac{\widehat{AUC_1}-\widehat{AUC_2}}{\sqrt{\sigma_1^2+\sigma_2^2}} \quad ,$$

dove σ^2_1 e σ^2_2 sono le varianze di $\widehat{AUC_1}$ e $\widehat{AUC_2}$, rispettivamente.

Se i due test non sono indipendenti (situazione che viene a verificarsi quando vengono applicati agli stessi soggetti), l'errore standard della differenza delle due aree viene a dipendere anche dalla correlazione *r* esistente tra esse, ossia

$$Z= \frac{\widehat{AUC_1}-\widehat{AUC_2}}{\sqrt{\sigma_1^2+ \sigma_2^2-2r\sigma_1\,\sigma_2}} .$$

La stima di *r* è stata illustrata in dettaglio da Hanley e McNeil (1982).

1.6 Scelta del valore soglia ottimale

In una curva ROC esistono due segmenti di scarsa importanza ai fini della valutazione dell'attitudine discriminante del test in esame. Essi sono rappresentati dalle frazioni di curva sovrapposte rispettivamente all'asse delle ascisse ed all'asse delle ordinate. Infatti, i corrispondenti valori possono essere scartati in quanto esistono altri valori di *cut-off* che forniscono una migliore Sp senza perdita di

Se o, viceversa, una migliore Se senza perdita di Sp. Infine, è da ricordare che la valutazione di un test attraverso l'AUC viene compiuta attribuendo ugual importanza alla Se e alla Sp, mentre in molti casi è necessario differenziare il peso da assegnare a tali parametri.

Nella maggioranza degli studi, l'individuazione del *cut-off* ottimale viene effettuata assumendo una distribuzione normale per la variabile oggetto di studio e si raggiunge adottando un valore pari a [media aritmetica + 2 deviazione standard] dei risultati generati dal gruppo dei pazienti sani. Questo approccio consente di ottenere una specificità pari al 97.5% (Barajas-Rojas, 1993), ma trascura completamente il valore della sensibilità.

Un metodo empirico comunemente utilizzato per la scelta del *cut-off* consiste nel fissare a priori il valore desiderato di specificità (generalmente ≥0.9) e, quindi, nel calcolare la corrispondente sensibilità del test nella suddetta condizione. Questo approccio genera tuttavia due effetti negativi. Il primo è rappresentato dall'evenienza che il test in questione possa produrre risultati complessivamente migliori attraverso l'adozione di un *cut-off* diverso da quello assunto. Il secondo è legato all'impossibilità di effettuare un raffronto affidabile fra la performance di due o più test valutati in base ad un singolo valore di *cut-off*.

Come regola generale si può affermare che il punto sulla curva ROC più vicino all'angolo superiore sinistro rappresenta il miglior compromesso fra sensibilità e specificità. Infatti, la distanza di ogni punto della curva ROC al punto (0,1) è pari a

$$\text{d}= \sqrt{(1-Se)^2 + (1-Sp)}^2.$$

Per ottenere il punto di *cut-off* ottimale si calcola questa distanza per ogni combinazione di Se e Sp: il valore soglia ottimale sarà il punto con distanza minore (Perkins, Schisterman, 2006).

Il valore ottimo può essere calcolato anche attraverso l'indice J di Youden (Perkins, Schisterman, 2006) , che massimizza la distanza verticale tra la *change line* e il generico punto (x,y) della curva

ROC (Figura 6). In altre parole, l'indice J di Youden, è il punto sulla curva ROC più lontano dalla diagonale ottenuto massimizzando la funzione

$$[\text{Se} + \text{Sp} - 1].$$

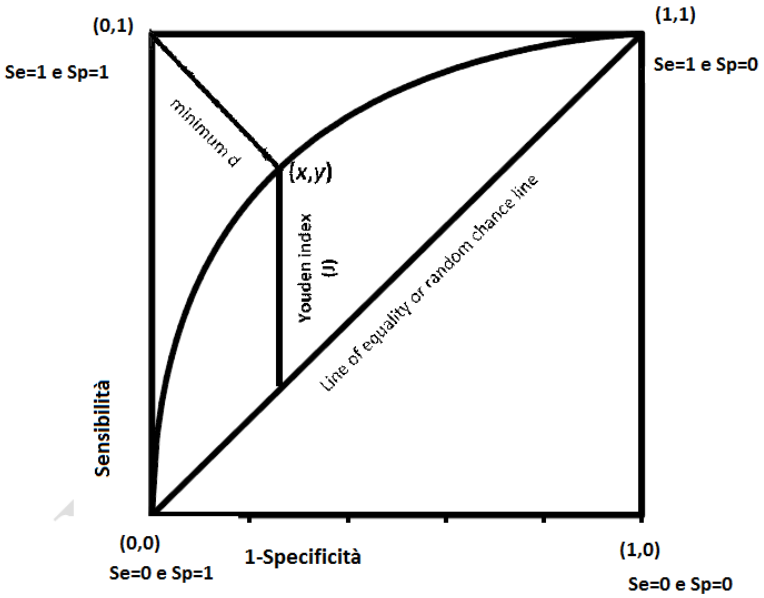

Figura 6. Criterio della minima distanza d e indice di Youden J.

Infine si ricorda che tra le misure che si possono ottenere dalla matrice di confusione, la più immediata è l'accuratezza, che rappresenta la frazione totale di casi classificati correttamente, data da

$$\text{Accuratezza} = \frac{TP+TN}{TP+TN+FN+FP} .$$

Riportando in un grafico i valori dell'accuratezza del test rispetto a diversi valori di *cut-off,* si sceglie come valore ottimo di *cut-off* il valore al quale corrisponde il massimo valore di accuratezza (Figura 7).

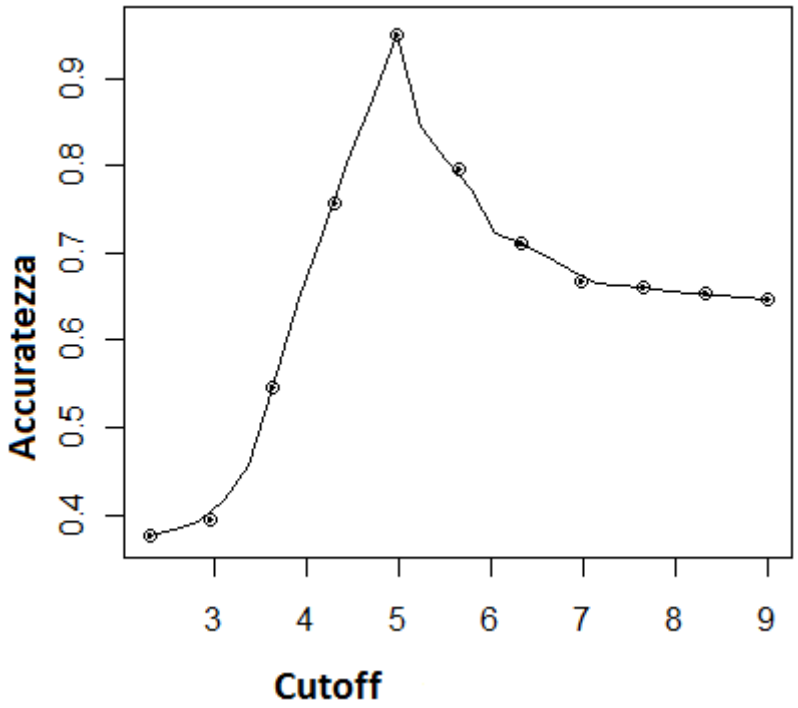

Figura7. Esempio di scelta del valore di soglia ottimale attraverso l' accuratezza del test. Dal grafico si nota che il valore massimo di accuratezza lo si raggiunge per un valore di *cut-off* pari a 5 (*cut-off* ottimale).

Appendice 1: Valori predittivi di un test e prevalenza della malattia

Sensibilità e specificità sono parametri definibili a priori, perché sono caratteristiche intrinseche del test che dipendono esclusivamente dalla tipologia del test adottato. Esse ci informano su qual è la probabilità di reclutare soggetti malati o sani da una certa popolazione di partenza (di malati e di sani), mentre nulla ci dicono sulla probabilità che, di fronte ad un singolo risultato positivo, quel soggetto sia realmente malato. Soprattutto nel campo dell'epidemiologia clinica, cioè quando i test vengono utilizzati a scopo diagnostico e non in operazioni di screening, ancor più interessanti risultano altri due parametri: il valore predittivo positivo VPP (probabilità che un soggetto scelto casualmente dalla popolazione, risultato positivo al test, sia effettivamente malato) e il valore predittivo negativo VPN (probabilità che un soggetto risultato negativo ad un test sia effettivamente sano).

La relazione tra valori predittivi di un test, sensibilità, specificità e prevalenza della malattia può essere ricavata analiticamente mediante il Teorema di Bayes. Dati due eventi A e B, definendo con P(A|B) e P(B|A), rispettivamente, la probabilità che si verifichi l'evento A dato che si è verificato l'evento B e viceversa, il Teorema di Bayes pone in relazione le due probabilità nel seguente modo

$$P(A|B)= \frac{P(B|A)P(A)}{P(B)} \ .$$

Ricordando la definizione di valore predittivo, di sensibilità e di specificità, si può scrivere, formalmente,

$$VPP= P(M+|T+), \quad VPN= P(M-|T-), \quad Se= P(T+|M+),$$

$$Sp= P(T-|M-) \ ,$$

dove M+ e M- indicano il vero stato del soggetto, e T+ T- se il test è risultato positivo o negativo, rispettivamente.

Si indichi inoltre con OP l'OR (*Odds Ratio*) di Prevalenza, definito formalmente come il rapporto tra la probabilità di osservare un soggetto malato rispetto a quella di osservare un soggetto non malato, ossia

$$OP = \frac{P(M+)}{P(M-)} .$$

Tale indice viene spesso impiegato per comodità, in quanto aumenta con la prevalenza e può essere stimato molto semplicemente come rapporto tra il numero dei malati e quello dei non malati nel campione.

Applicando il Teorema di Bayes alle definizioni sopra riportate si ricava immediatamente la relazione tra valori predittivi, prevalenza, sensibilità e specificità, data da

$$VPP = P(M+|T+) = \frac{P(T+|M+)P(M+)}{P(T+)} .$$

Si consideri inoltre che la probabilità di un test positivo P(T+) è pari alla somma delle probabilità di un test positivo nei malati e della probabilità di un test positivo nei non malati, ossia

$$VPP = P(M+|T+) = \frac{P(T+|M+)P(M+)}{P(T+|M+)P(M+) + P(T+|M-)P(M-)} ,$$

da cui si ottiene la seguente relazione

$$\text{VPP}= P(M+|T+)= \frac{Se}{Se+\frac{(1-Sp)}{OP}} .$$

Analogamente, si dimostra che

$$\text{VPN}= P(M-|T-)= \frac{P(T-|M-)P(M-)}{P(T-|M-)P(M-)+ P(T-|M+)P(M+)} ,$$

da cui si ottiene

$$\text{VPN}= P(M-|T-)= \frac{Sp}{Sp+(1-Se)OP} .$$

Dal punto di vista pratico risulta evidente dal Teorema di Bayes che:

- in condizioni di bassa prevalenza diminuisce il valore predittivo del test positivo;
- in condizioni di bassa specificità del test diminuisce il valore predittivo del test positivo;
- in condizioni di bassa specificità e di bassa prevalenza aumenta il valore predittivo del test negativo, quindi un test diventa utile soprattutto per escludere la malattia.

Altri parametri spesso impiegati per valutare la performance di un test diagnostico sono i rapporti di verosomiglianza (LR, dall'inglese "*likelihood ratio*"), così definiti

$$\text{LR+} = \frac{P(T+|M+)}{P(T+|M-)} = \frac{Se}{1-Sp}, \qquad \text{LR-} = \frac{P(T-|M+)}{P(T-|M-)} = \frac{1-Se}{Sp} .$$

Il rapporto di verosomiglianza di un risultato positivo (LR+) esprime la probabilità di un risultato positivo in un soggetto malato rispetto alla medesima probabilità in un soggetto sano. Analogamente, il rapporto di verosomiglianza di un risultato negativo (LR-) esprime la probabilità di un risultato negativo in un soggetto malato rispetto alla medesima probabilità in un soggetto sano. In termini di rapporti di verosomiglianza, le relazioni sopra illustrate diventano

$$\text{VPP}= \frac{(\text{LR}+)\text{OP}}{(\text{LR}+)\text{OP}+1} \qquad \text{VPN}= \frac{1}{1+\text{OP}(\text{LR}-)} \ .$$

Illustrando tali relazioni in un grafico (Figura 8), si può notare che il VPP tende ad aumentare in modo non lineare con la prevalenza e con maggiore rapidità per valori elevati di LR+, mentre VPN tende a diminuire con la prevalenza, tanto più rapidamente quanto più elevato è LR-.

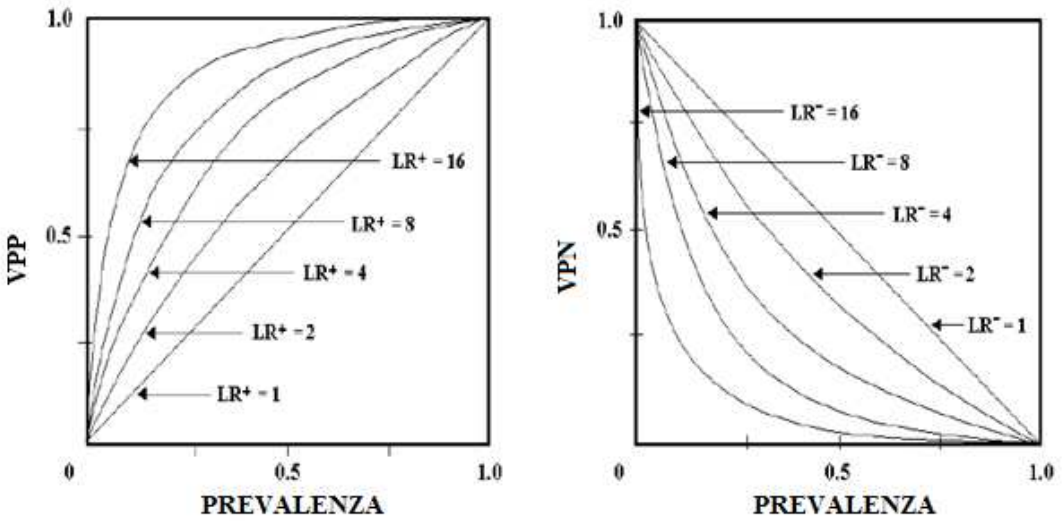

Figura 8. Relazione tra valori predittivi e prevalenza.

Capitolo 2. Implementazione della curva ROC in R

L'analisi tramite la curva ROC si può effettuare con diversi programmi statistici, uno di questi è R. Nei successivi paragrafi presenteremo in breve il programma *open-source* R e ci soffermeremo su alcuni dei *package* di R che implementano l'analisi ROC.

2.1 L'ambiente statistico R

Esiste una gamma assai vasta di software specializzati nell'analisi statistica dei dati, basta ricordare SAS, SPSS, STATA, STATGRAPHICS PLUS, SHAZAM, S PLUS, MINITAB, GAUSS, etc., solo per citarne alcuni che sono in commercio. Sono prodotti che costituiscono senz'altro un fondamentale ed insostituibile ausilio per il lavoro dello statistico. Tuttavia, molti di questi programmi sono anche alquanto costosi ed è consentito l'uso su licenza da parte del produttore.

Da alcuni anni a questa parte, soprattutto in ambito universitario (ma non solo), si sta sempre più diffondendo un nuovo *package* che merita di sicuro una debita trattazione e l'interessamento da parte degli statistici e di coloro che fanno analisi dei dati, e che costituisce anche una valida alternativa ai *software* commerciali. Ci stiamo riferendo al *software* R (http://www.r-project.org/). R è un ambiente statistico scaricabile gratuitamente dal sito di *The R Project for Statistical Computing*. Esso è il frutto del lavoro collettivo svolto da un gruppo, sempre più folto, di ricercatori in campo statistico ed informatico a livello mondiale. R più che un software statistico può essere definito come un ambiente, costituito da una varietà di strumenti, orientato alla gestione, all'analisi dei dati e alla produzione di grafici, basato sul linguaggio S creato da *AT&T Bell Laboratories*. R è contemporaneamente un linguaggio ed un software.

Le sue caratteristiche principali possono essere così riassunte:

a) semplicità nella gestione e manipolazione dei dati;

b) disponibilità di una suite di strumenti per calcoli su vettori, matrici ed altre operazioni complesse;

c) accesso ad un vasto insieme di strumenti integrati per l'analisi statistica;

d) produzione di numerose potenzialità grafiche particolarmente flessibili;

e) possibilità di adoperare un vero e proprio linguaggio di programmazione orientato ad oggetti che consente l'uso di strutture condizionali e cicliche, nonché di funzioni create dall'utente.

L'ambiente R è basato sul concetto di "*package*" tradotto di solito in italiano con pacchetto. Un *package* è un insieme di strumenti che svolgono determinate funzioni, ma può anche contenere solo dati, oppure solo documentazione. Attualmente è disponibile una vasta gamma di *packages* (http://cran.stat.unipd.it/) utilizzabili per la risoluzione di specifici problemi o per analisi statistiche molto particolareggiate. Il cuore di R è rappresentato dal modulo base (che offre gli strumenti fondamentali per l'analisi statistica) e attorno a questo modulo "ruotano" una serie di altre librerie addizionali, alcune delle quali sono già comprese nel programma R al momento in cui lo si installa, mentre altre librerie ancora, in relazione alle esigenze e necessità, possono essere aggiunte e installate dall'utente dopo averle scaricate dal sito.

In R sono disponibili diversi pacchetti che implementano la possibilità di analisi della curva ROC e più in generale valutare il potere discriminatorio di un test. Alcuni esempi sono: ROCR, HSROC, caTools, Analogue, pROC, nonbinROC, risksetROC, rocc, rocplus, survivalROC, verification.

Nei prossimi paragrafi analizzeremo in dettaglio alcuni di questi pacchetti.

2.2 ROCR

Il pacchetto ROCR è sicuramente il più noto e il più utilizzato e lo si può scaricare dall'URL http://rocr.bioinf.mpi-sb.mpg.de/.

ROCR è un *tool* flessibile che permette di costruire la curva ROC dai dati forniti dall'utente, con la possibilità di ulteriori analisi combinando tra loro due tra le 25 misure di performance possibili (le misure di performance possono essere ampliate usando l'interfaccia standard).

Tutto questo è possibile grazie all'utilizzo di tre semplici classi di comandi:

1. Prediction-class
2. Performance-class
3. Plot-methods

2.2.1 Prediction-class

Ogni valutazione di un classificatore utilizzando ROCR inizia con la creazione di un "oggetto" definito *prediction*. Questa funzione viene utilizzata per trasformare i dati in nostro possesso (che possono essere in formato vettoriale, matriciale, o sotto forma di un elenco) in formato standardizzato.

L'espressione della funzione *prediction* è la seguente

prediction(predictions, labels, label.ordering=NULL)

In questa funzione sono presenti tre argomenti:

- *predictions*: un vettore, una matrice, una lista o un data frame che contiene i valori predetti dal test;

- *labels*: un vettore, una matrice, una lista o un data frame che contiene la vera classificazione dei dati. Deve avere la stessa dimensione dell'argomento predictions;
- *label.ordering*: permette di modificare l'ordine predefinito da R dell'argomento *labels.* Se posto uguale a *NULL* verrà mantenuto l'ordinamento predefinito.

Attualmente ROCR supporta solo la classificazione binaria (estensioni verso la classificazione multi-classe sono in programma). Se ci sono più di due etichette distinte, l'esecuzione viene eseguita con un messaggio di errore.

L'oggetto prediction contiene informazioni relative alla matrice di confusione 2x2 (vedi Tabella 1) come *tp,fp,tn,fn,* insieme con le somme marginali *n.pos, n.neg, n.pos.pred, n.neg.pred*, poiché questi indici formano le basi di molte delle misure di performance che si possono derivare.

Assegnando un nome alla funzione, si ottiene l'oggetto

pred=*prediction(predictions, labels, laber.order=NULL)*

e poi, richiamandolo, si ottengono le seguenti informazioni:

- predictions: una lista dei valori contenuti nell'argomento *predictions*;
- labels: la lista delle etichette contenute nell'argomento *labels*;
- cutoffs: una lista dei valori di *cut-off* utilizzati per discriminare i dati. L'ordinamento è in ordine decrescente. Per ogni valore di *cut-off* si ottiene una diversa tabella di contingenza;
- fp: un vettore di numeri in cui ogni elemento rappresenta il numero di falsi positivi indotti dal corrispondente valore presente in cutoffs;
- tp: come fp, ma per i veri positivi;
- tn: come fp, ma per i veri negativi;
- fn: come fp, ma per i falsi negativi;
- n.pos: contiene il vero numero di positivi;

- n.neg: contiene il numero di veri negativi;
- n.pos.pred: un vettore di numeri in cui ogni elemento rappresenta il numero di positivi predetti dal test indotti dal corrispondente valore presente in cutoffs;
- n.pos.pred: come n.pos.pred, ma per il numero di negativi predetti.

2.2.2 Performance-class

Tutti i tipi di valutazione predittivi si ottengono utilizzando questa funzione. Dopo aver ottenuto l'oggetto pred, attraverso questa funzione è possibile il calcolo dei valori di sensibilità, specificità e accuratezza, rappresentare la curva ROC e stimare l'AUC.

L'espressione della funzione *performance* è la seguente

performance(prediction.obj, measure, x.measure)

All' interno sono presenti tre argomenti:

- *prediction.obj*: l'oggetto della classe *prediction* (es: "pred");
- *measure*: contiene una misura di performance da utilizzare per la valutazione. In seguito mostreremo alcune delle misure di performance più importanti tra quelle disponibili;
- *x.measure*: contiene una seconda misura di performance. Se diversa dal valore predefinito, *x.measure* rappresenta l'unità nell'asse delle ascisse, *measure* l'unità nell'asse delle ordinate, dell'eventuale curva bidimensionale che si potrà creare con la funzione *plot*, che analizzeremo nel seguito. Questa curva è parametrizzata con il valore di *cut-off*.

Un oggetto può essere creato per catturare le misure di valutazione che si possono ottenere con la funzione *performance.* Assegnando un nome alla classe *performance*, si ottiene l'oggetto

perf= *performance(* pred*, measure, x.measure)*

e richiamandolo vengono visualizzate le seguenti informazioni:

- x.name: una misura di performance utilizzata per l'asse x;
- y. name: la seconda misura di performance utilizzata per l'asse y;
- alpha.name: il nome dell'unità che viene utilizzata per parametrizzare la curva.

  Può assumere valore "none" oppure "cut-off";
- x.values: una lista in cui ogni voce contiene i valori dell'argomento *measure*;
- y.values: una lista in cui ogni voce contiene i valori dell'argomento *x.measure*;
- alpha.values: contiene la lista dei valori soglia con la quale verrà parametrizzata la curva.

I valori x.values, y.values e alpha.values sono in corrispondenza univoca.

Come detto precedentemente, all'interno delle classe performance sono presenti un'ampia varietà di misure di performance. Qui di seguito elencheremo le principali; per le altre si può consultare la manualistica della libreria ROCR.

Per facilitare l'esposizione, indicheremo con Y la vera classificazione,$\hat{Y}$la classificazione stimata dalla classe *prediction,* e con ⧋ e△ i valori positivi e negativi, rispettivamente. Ecco un elenco delle misure di performance:

- *acc*: accuratezza: P($\hat{Y}$=Y). Frazione totale di casi classificati correttamente. La sua stima è: $\frac{TP+TN}{P+N}$;
- *err*: *error rate*: P($\hat{Y}\neq$Y). Frazione totale di casi classificati scorrettamente. La sua stima è: $\frac{FP+FN}{P+N}$;
- *fpr*: *false positive rate*. Complemento a uno della specificità: P($\hat{Y}$=⧋|Y=△). La sua stima è: $\frac{FP}{N}$;
- *tpr*: *true positive rate*. Sensibilità: P($\hat{Y}$=⧋|Y=⧋). La sua stima: $\frac{TP}{P}$;

- *fnr*: *false negative rate*: P($\hat{Y}$=△|Y=△). La sua stima è: $\frac{FN}{P}$;
- *tnr*: *true negative rate*: Specificità. P($\hat{Y}$=△|Y=△). La sua stima: $\frac{TN}{N}$;
- *ppv*: *positive predictive value*: P(Y=△|$\hat{Y}$=△). La sua stima è: $\frac{TP}{TP+FP}$;
- *npv*: *negative predictive value*: P(Y=△|$\hat{Y}$=△). La sua stima è: $\frac{TN}{TN+FN}$;
- *rch*: curva ROC. Non può essere utilizzata in combinazione con altre misure di performance;
- *auc*: area sottesa alla curva ROC. Questo valore, come visto nel capitolo precedente, è collegata al valore della statistica test di Wilcoxon. Questa misura di performance non può essere utilizzata in combinazione con altre. L'area parziale sotto la curva può essere calcolata dato un certo valore di *fpr,* con il comando opzionale *fpr.stop=0.5* (o un qualsiasi altro valore tra 0 e 1) da aggiungere alla funzione *performance*.

2.2.3 Plot-methods

E' una classe di funzioni attraverso la quale è possibile creare il grafico di tutte le misure di performance contenute nella classe *performance*.

Si riportano nel seguito degli esempi di combinazioni di misure di performance per ottenere alcuni dei grafici più importanti:

- Curva ROC: perf= *performance*(pred, "*tpr*","*fpr*")
- *Plot* Sensibilità/Specificità: perf= *performance*(pred,"*sens*","*spec*")

Naturalmente è possibile ottenere molti altri diversi grafici di valutazione del test, combinando differenti misure di performance. L'espressione della funzione è la seguente

plot(x, avg="none",spread.estimate="none",

add=FALSE,colorize=FALSE,print.cutoffs.at=c(),downsampling=0)

in cui:

- x: oggetto della classe *performance* (nel nostro caso perf);
- *avg*: se l'oggetto *performance* descrive più curve, quest'ultime possono essere valutate in media. I valori consentiti sono: "*none*" (*plot* con tutte le curve separate), "*horizontal*" (in media orizzontale), "*vertical"* (in media verticale) e "*threshold*" (in media con il valore di soglia);
- *spread.estimate*: quando l'argomento *avg* è attivo, la variazione attorno la *average curve* può essere visualizzata come barre di *errore standard* ("*stderror*"), barre di *standard deviation* ("*stddev*"), o usando i *boxplot* ("*boxplot*"). L'uso di questo argomento può portare ad un messaggio di errore nel caso in cui, in una certa posizione, la variazione sia pari a zero;
- *colorize*: permette di colorare la curva in base alla variazione del valore di *cut-off*;
- *print.cutoffs.at*: permette di stampare lungo la curva i valori di *cut-off* selezionati dall'utente;
- *downsampling*: ROCR può calcolare misure di performance in modo efficiente anche per insiemi di dati con milioni di elementi. Tuttavia, la rappresentazione dei *plot* per grandi insiemi di dati può essere molto lenta e può portare alla creazione di file di dimensione elevata. In tali casi, una curva indistinguibile dall'originale, si può ottenere utilizzando solo una parte degli elementi a disposizione. Valori di *downsampling* compresi tra 0 e 1 rappresentano la frazione dei dati originali che viene utilizzata per la creazione dell'oggetto performance;
- *add*: se posto pari a *TRUE*, la curva sarà aggiunta ad un *plot* già esistente.

Per gli argomenti *spread.estimate* e *colorize* esistono delle opzioni, la cui trattazione è rimandata alla manualistica della libreria ROCR.

2.3 pROC

Il pacchetto pROC può essere scaricato al seguente indirizzo

http://cran.r-project.org/web/packages/pROC .

L'unità di base del pacchetto pROC è la funzione *roc*. Attraverso di essa si può costruire la curva ROC, eseguire l'operazione di *smoothing* (lisciamento), calcolare l'area sottesa ad essa (AUC), e calcolare gli intervalli di confidenza di alcuni parametri.

E' inoltre possibile confrontare due curve ROC (osservate su uno stesso campione) attraverso la funzione *roc.test*.

2.3.1 Roc-function

E' la funzione principale del pacchetto pROC e permette di costruire la curva ROC e di creare un oggetto "roc", che poi potrà essere richiamato dalle funzioni *plot, print, auc, ci* e *coords*. Inoltre, due oggetti creati attraverso la funzione *roc* possono essere confrontati attraverso il *roc.test*.

L' espressione della funzione è la seguente:

roc(response, predictor, percent=TRUE, auc=TRUE,na.rm=TRUE, plot=TRUE)

Gli argomenti all'interno della funzione hanno le seguenti utilità:

- *response*: vettore delle risposte solitamente di tipo numerico o di tipo *factor*, tipicamente codificate con "0" (controlli) e "1" (casi). Solo due classi possono essere utilizzate per la costruzione della curva ROC. Per questo, se il vettore contiene più di due valori distinti, attraverso l'argomento *levels* si può specificare quali valori devono essere utilizzati come

controlli e quali come casi;

- *predictor*: vettore numerico contenente il valore di ogni osservazione (valori osservati dal test);
- *levels*: permette all'utente di codificare i dati contenuti nell'argomento *response* in "0" e "1". Se la codificazione è già stata effettuata, questo argomento può essere omesso;
- *percent*: consente di ottenere i valori di sensibilità, specificità e AUC in percentuale (*TRUE*) o in frazione (*FALSE*, *default*);
- *na*.rm: se *TRUE* i valori Na (dati mancanti) vengono automaticamente rimossi;
- *auc*: calcola la stima non parametrica dell'area sottesa alla curva;
- *plot*: crea il *plot* della curva ROC.

Attraverso questa semplice funzione si può molto semplicemente rappresentare la curva ROC e calcolare l'area sottesa ad essa. Ma se vogliamo effettuare un analisi più approfondita, basta creare l'oggetto roc.obj e poi richiamarlo con le altre funzioni presenti nel pacchetto pROC.

2.3.2 AUC-function

Questa funzione viene tipicamente richiamata se si è posto l'argomento *auc=TRUE* all'interno della funzione *roc*, che permette di calcolare solamente il valore dell'AUC.

Invece, attraverso la funzione *auc,* è possibile il calcolo parziale dell'area e il calcolo dell'area parziale corretta.

La terminologia della funzione *auc* è la seguente

*auc(*roc.obj*, partial.auc=c(),partial.auc.focus=("sp","se"),partial.auc.correct=TRUE)*

con:

- roc.obj: oggetto della funzione *roc*;
- *partial.auc*: se *TRUE*, si deve inserire un vettore di dimensione due, contenente l'intervallo da considerare per il calcolo parziale dell'area;
- *partial.auc.focus*: se l'argomento *partial.auc* è attivo, è possibile specificare se l'intervallo inserito nell'argomento *partial.auc* è in termini di specificità (*default*) o in termini di sensibilità;
- *partial.auc.correct*: se *TRUE*, viene applicata la correzione di McClish al calcolo dell'area parziale. Con questa correzione, l'AUC parziale è pari a 0.5 se il test è non discriminante, pari a 1 se il test ha massimo potere discriminante, qualunque intervallo sia stato definito. Attraverso questa correzione si può valutare l'accuratezza di un test anche usando l'AUC parziale.

2.3.3 CI-function

Questa funzione permette di calcolare l'intervallo di confidenza per l'AUC, per la sensibilità, per la specificità e per i valori di *cut- off*. Questo è possibile grazie al metodo "*bootstrap*" (vedi Appendice 2). Per il calcolo di intervalli di confidenza al 95%, per *default*, vengono utilizzati 2000 campioni *bootstrap*.

2.3.3.1 CI-auc

Questa funzione permette di calcolare l'intervallo di confidenza per l'AUC.

La terminologia della funzione è la seguente

*ci.auc(*roc.obj*, conf.level=0.95,method=c("delong","bootstrap"), boot.n=2000,*

boot.stratified=TRUE, reuse.auc= TRUE)

Gli argomenti all'interno della funzione hanno il seguente significato:

- roc.obj: oggetto della funzione *roc*;
- *conf.level*: livello dell'intervallo di confidenza, da *default* pari a 0.95;
- *method*: due metodi possono essere utilizzati per il calcolo degli intervalli di confidenza, "*delong*" e "*bootstrap*". Da *default* viene utilizzato il metodo "*delong*", perché più flessibile e rapido, eccetto i casi di confronto di AUC parziali e di *smoothing* curve, come definito da Delong (1988). Per i casi in cui il metodo "*delong*" non è supportato, si usa il metodo "*bootstrap*" come definito da Carpenter e Bithell (2000);
- *boot.n*: numero di ricampionamenti utilizzati. Per default pari a 2000. Naturalmente maggiore sarà il numero di ricampionamenti, più precise saranno le stime;
- *boot.stratified*: la stratificazione del *bootstrap* può essere controllata con *boot.stratified*. Se posta *TRUE* (*default*), ogni replica contiene lo stesso numero di casi e di controlli rispetto al campione originale. Questo controllo è molto utile, perché senza questa impostazione potrebbe capitare che una o più repliche non contengano alcuna osservazione "caso" o "controllo", il che non permette il calcolo degli intervalli di confidenza;
- *reuse.auc*: se *TRUE* (default) e il roc.obj contiene l'argomento *auc* (nella funzione *roc*, l'argomento *auc=TRUE*), vengono riutilizzate queste informazioni per il calcolo degli intervalli. Se roc.obj non contiene l'argomento *auc*, basta utilizzare la funzione *auc* e creare un oggetto auc.obj e poi richiamarlo all'interno della funzione *ci.auc* in questo modo

ci.auc(roc.obj,auc.obj,...) .

2.3.3.2 CI- Sensibilità, specificità e cut-off

Come per l'AUC, è possibile calcolare intervalli di confidenza per la sensibilità per determinati valori di specificità, viceversa calcolare intervalli di confidenza per la specificità per determinati valori di sensibilità.

La terminologia delle due funzioni è la seguente

1. *ci.se*(roc.obj*, conf.level=0.95, specificities = seq(0, 1, .1) ,method=c("delong","bootstrap"), boot.n=2000, boot.stratified=TRUE)*
2. *ci.sp(*roc.obj*, conf.level=0.95, sensitivities = seq(0, 1, .1) ,method=c("delong","bootstrap"), boot.n=2000, boot.stratified=TRUE)*

Le due espressioni sono molto simili e differiscono solamente per gli argomenti *specifities* e *sensitivities.* Attraverso questi argomenti l'utente inserisce i valori di specificità o sensibilità in corrispondenza della quale verranno calcolati gli intervalli di confidenza della sensibilità o specificità, rispettivamente.

Se invece siamo interessati al calcolo degli intervalli di confidenza della Se e Sp in corrispondenza di determinati valori *di cut-off*, è sufficiente utilizzare questa funzione

*ci.thresholds(*roc.obj*, conf.level=0.95,thresholds="local maximas" ,boot.n=2000, boot.stratified=TRUE)*

Attraverso l'argomento *thresholds* è possibile inserire un vettore di numeri, o i caratteri: "*all*" (calcolo degli intervalli per ogni valore di *cut-off*), "*local maximas*" (calcolo per i valori di *cut-off* massimi locali), "*best*" (calcolo eseguito solamente per il valore di *cut-off* migliore).

2.3.3.3 Plot

E' la funzione definita nella libreria pROC che permette di aggiungere gli intervalli di confidenza di uno dei parametri visti precedentemente, ad un *plot* di una curva ROC già esistente. Naturalmente gli intervalli di confidenza per l'AUC non potranno essere rappresentati.
Dopo aver creato un oggetto (ci.se.obj, ci.sp.obj, ci.thresholds.obj) del parametro del quale siamo interessati, lo si potrà richiamare attraverso la funzione *plot*, come

*plot (*x*, type=c("bars", "shape",length=…).*

Gli argomenti all'interno della funzione hanno il seguente significato:

- x: uno degli oggetti precedentemente creati;
- *type*: scelta del tipo di *plot*, "*bars*" o "*shape*";
- *length*: la lunghezza delle "*bars*" *plot*.

2.3.4 Coords-function

Questa funzione di semplice utilizzo permette di individuare all'interno del ROC *plot* alcuni punti di interesse, come ad esempio il valore di *cut-off* ottimale. Più in generale questa funzione restituisce le coordinate della curva ROC per i punti specificati dall'utente.
La terminologia della funzione è la seguente

*coords(*roc.obj*, x, input=c("threshold", "specificity","sensitivity"), ret=c("threshold", "specificity", "sensitivity"), best.method=c("youden", "closest.topleft")*

Gli argomenti hanno il seguente significato:

- roc.obj: oggetto della funzione *roc*;
- *x*: coordinate alla quali siamo interessati. Possono essere numeriche (il significato dovrà essere però specificato nell'argomento *input*) o espresse attraverso uno di questi argomenti: "*all*" (tutti i punti della curva ROC), "*local maximas*" (il massimo locale della curva ROC), "*best*" (il punto che massimizza la somma tra sensibilità e specificità);
- *input*: se l'argomento x è numerico, bisogna specificare il significato delle coordinate inserite, cioè uno tra "*threshold*", "*specificità*" e "*sensitività*");
- *ret*: valori che vogliamo ci vengano restituiti. Uno o più tra "*threshold*", "*specificity*" e "*sensitivity*";
- *best.method*: se *x*= "*best*", specificare il criterio per l'individuazione del miglior valore di *cut-off*. I due metodi disponibili per l'individuazione del *cut-off* ottimale sono l'indice di Youden (*best.method*="*youden*"), e il metodo della minima distanza (*bestmethod*= "*closet.topleft*").

2.3.5 roc.test

Questa funzione permette di comparare due curve ROC, attraverso il confronto dell'AUC o dell'AUC parziale. Per poter applicare questa funzione, bisogna prima specificare se le due curve ROC che vogliamo analizzare sono correlate tra di loro.

Per far questo, la libreria pROC mette a disposizione la funzione *are.paired*.

Due curve ROC sono correlate se sono costruite su due variabili osservate sullo stesso campione, ossia, se il vettore di valori contenuti nell'argomento *response* di entrambe le curve ROC sono identici. Quindi la funzione *are.paired* non fa altro che verificare questa uguaglianza.

La terminologia della funzione è la seguente

*are.paired (*roc1.obj*,* roc2.obj*,return.paired.rocs=TRUE)*

Gli argomenti della funzione hanno il seguente significato:

- roc1.obj, roc2.obj: i due oggetti associati a due funzioni *roc*, che contengono le due curve ROC che vogliamo analizzare;
- *return.paired.rocs*: se *TRUE* e le due curve ROC sono correlate, questo argomento rappresenterà le due curve ROC in un unico *plot*.

Quando questa funzione viene eseguita in R, restituisce due tipi di valori: TRUE se le due curve ROC sono correlate, FALSE altrimenti.

Verificata la correlazione tra le due curve, è possibile comparare i due test discriminatori mediante la funzione *roc.test*, per capire quale sia il test migliore.

L'espressione della funzione è la seguente

*roc.test(*roc1.obj, roc2.obj*,*

method=c("delong","bootstrap","venkatraman"),alternative=c("two.sided","less",

"greater"),reuse.auc=TRUE, boot.n=2000, boot.stratified=TRUE,ties.method="first")

con

- roc1.obj, roc2.obj: i due oggetti associati a due funzioni *roc*, che contengono le due curve ROC che vogliamo comparare;
- *method*: sono disponibili tre metodi: "*delong*", "*bootstrap*" e "*venkatraman*". I primi due, per la comparazione dei due test, confrontano l'AUC delle due curve ROC; il terzo, invece, confronta direttamente le due curve. Da *default* viene utilizzato il metodo "*delong*" (DeLong, 1988), tranne per il confronto di AUC parziali, *smoothed* curve e curve con

direzione diversa, in cui viene utilizzato il metodo "*bootstrap*" (Carpenter, Bithell, 2000). Il metodo "*venkatraman*" viene eseguito come descritto da Venkatraman e Begg (1966) per curve ROC correlate, e Venkatraman (2000) per curve ROC indipendenti. Il vantaggio di questo ultimo metodo è quindi la possibilità di confrontare anche curve ROC indipendenti, cosa non possibile con i primi due metodi;

- *alternative*: specifica l'ipotesi alternativa del test: "*two.sided*" (test bilaterale), "*less*" (l'AUC di roc1.obj più piccola dell'AUC di roc2.obj), "*greater*" (l'AUC di roc1.obj più grande dell'AUC di roc2.obj);
- *reuse.auc*: se *TRUE* (*default*) e il roc.obj contiene l'argomento *auc* (nella funzione *roc*, l'argomento *auc=TRUE*), vengono riutilizzate queste informazioni per eseguire il test d'ipotesi;
- *boot.n*: indica il numero di repliche e di permutazioni utilizzate per il metodo "*bootstrap*" e "*venkatraman*", rispettivamente;
- *boot.stratified*: se posto pari a TRUE controlla la stratificazione di ogni replica;
- *ties.method*: solo per il metodo "*venkatraman*". Da default "*first*" (Venkatraman, 2000).

Eseguendo la funzione *roc.test* si ottiene la seguente lista di valori:

- p.value: il p-value del test;
- il valore della statistica Z (metodo "*delong*") o della statistica test D (metodo "*bootstrap*");
- alternative: l'ipotesi alternativa verificata;
- method: metodo scelto dall'utente per eseguire la verifica dell'ipotesi;
- null.value: valore atteso della statistica test sotto l'ipotesi nulla;
- estimate: stima dell'AUC per le due curve ROC;
- parameter: solo per il metodo "*bootstrap*", vengono riportati i valori degli argomenti *boot.n* e *boot.stratified*.

2.4 Verification

Il pacchetto verification, scaricabile dal sito

http://cran.r-project.org/web/packages/verification/index.html, sebbene non sia stato creato specificamente per questo scopo, permette la costruzione della curva ROC e di calcolare l'AUC. Un test, basato sulla statistica U di Wilcoxon, è implementato all'interno del pacchetto, per valutare la performance di un modello di classificazione, ma non è possibile il confronto attraverso l'analisi della curva ROC di due modelli di classificazione.

2.4.1 roc.plot

Questa funzione permette di costruire la curva ROC.

L'espressione della funzione è la seguente

roc.plot(obs, pred, thresholds=NULL,binormal=FALSE,plot= "emp", CI=FALSE, n.boot=2000, alpha=0.05)

Gli argomenti all'interno della funzione hanno il seguente significato:

- *obs*: osservazioni binarie codificate con 0 e 1;
- *pred*: valori predetti sull' intervallo [0,1]. Se si confrontano più modelli, viene costruita una matrice in cui ogni colonna rappresenta, rispettivamente, i valori predetti da ciascun modello;
- *thresholds:* attraverso questo argomento possiamo fornire i valori di *cut-off* che vogliamo utilizzare per la costruzione della curva ROC. Se posto pari a *NULL* (*default*), verranno utilizzati come *cut-off* tutti i valori dell'intervallo [0,1];
- *binormal*: se *TRUE*, in aggiunta alla curva ROC, viene calcolata la curva ROC sotto ipotesi

bi-normale;

- *plot*: è possibile scegliere tra tre tipi diversi di *plot*: "*emp*" (curva ROC empirica), "*binormal*" (curva ROC sotto ipotesi bi-normale), "*both*" (vengono costruite entrambe le curve);
- *CI*: intervalli di confidenza, calcolati con il metodo "*bootstrap*";
- *alpha*: livello degli intervalli di confidenza. (*Default alpha= 0.05*);
- *boot.*n: numero di ricampionamenti utilizzati per la costruzione degli intervalli di confidenza.

2.4.2 roc.area

Tale funzione permette il calcolo dell'AUC e la valutazione della performance di un singolo test di classificazione utilizzando la statistica test U di Wilcoxon.

La terminologia della funzione è la seguente

roc.area(obs, pred)

con gli stessi argomenti *obs* e *pred*, visti nella funzione precedente.

Una volta eseguita questa funzione si ottengono i seguenti valori:

- A: valore dell'AUC;
- n.total: numero totale di osservazioni binarie contenute nell'argomento *obs;*
- n.events: numero totale di 1 nell'argomento *obs*;
- n.noevents: numero totale di 0 nell'argomento *obs*;
- p-value: il p- value della statistica Z, ottenuto come visto nel paragrafo 1.4.

2.5 Tabella riassuntiva

In questo capitolo abbiamo analizzato tre dei pacchetti disponibili in R che implementano la possibilità di analizzare la curva ROC. Si riporta una tabella riassuntiva che mostra le differenze tra questi tre pacchetti.

Pacchetto	ROCR	Verification	pROC
Smoothing	NO	SI	SI
AUC	SI	SI	SI
AUC PARZIALE	SI	NO	SI
INTERVALLI DI CONFIDENZA	SI	SI	SI
TEST STATISTICO	NO	SI	SI
TEST (PIU' MODELLI)	NO	NO	SI

Tabella 2. Differenza tra i pacchetti ROCR, verification e pROC.

Appendice 2: Metodo Bootstrap e DeLong per comparare curve ROC

Il *bootstrap* è una tecnica statistica di ricampionamento utile per approssimare la distribuzione campionaria di una statistica. Permette perciò di approssimare media e varianza di uno stimatore, costruire intervalli di confidenza e calcolare livelli di significatività osservati di test quando, in particolare, non si conosce la distribuzione della statistica di interesse. L'idea alla base del *bootstrap* è quella di utilizzare la distribuzione empirica del campione, che è l'unica informazione che abbiamo sulla ignota distribuzione della popolazione $F^0(\bullet)$, generando numerosi campioni con una procedura di ricampionamento con ripetizione di *n* elementi dagli *n* dati campionari. In questo modo si ottengono diverse stime del parametro d'interesse con le quali, grazie all'aiuto del computer e senza utilizzare formule matematiche particolarmente complicate, si è in grado di ottenere misure di variabilità dello stimatore, quali errori standard, distorsione e intervalli di confidenza. Nel caso di campionamento casuale semplice, il funzionamento è il seguente: si consideri un campione effettivamente osservato di numerosità pari a *n*, indicato con $x=(x_1, \ldots, x_n)$. Da *x* si campionano *B* campioni di numerosità costante pari ad *n*, indicati con $x_1^*, \ldots, x_B^*$. In ciascuna estrazione *bootstrap*, i dati provenienti dal primo elemento del campione, cioè x_1, possono essere estratti più di una volta e ciascun dato ha probabilità pari a *1/n* di essere estratto. Sia *T* lo stimatore di Θ, diciamo $T(X) = \hat{\theta}$. Si calcola tale quantità per ogni campione *bootstrap*, ossia $T(x_1^*), \ldots, T(x_B^*)$. In questo modo si hanno a disposizione B stime di Θ, dalle quali è possibile calcolare la media *bootstrap*, la varianza *bootstrap* ecc., che sono approssimazioni dei corrispondenti valori ignoti e portano informazioni sulla distribuzione di *T(X)*.

Il metodo *bootsrap* per comparare l'AUC di due curve ROC correlate, utilizza la procedura descritta da Hanley e McNeil (1983), definendo

$$Z = \frac{\widehat{\theta_1} - \widehat{\theta_2}}{sd(\widehat{\theta_1} - \widehat{\theta_2})},$$

dove $\widehat{\theta_1}$ *e* $\widehat{\theta_2}$ sono i due valori AUC. Il calcolo di *sd* $(\widehat{\theta_1} - \widehat{\theta_2})$ viene effettuato con B ricampionamenti (*bootstrap*). Z segue approssimativamente una distribuzione normale standard.

Il secondo metodo per comparare due curve ROC attraverso l'AUC e stato introdotto da DeLong (DeLong et al., 1988). Questo test non richiede di utilizzare ricampionamenti dato che la varianza della differenza dei due valori di AUC può essere calcolata come

$$var(\widehat{\theta_1} - \widehat{\theta_2}) = var(\widehat{\theta_1}) + var(\widehat{\theta_2}) - 2cov(\widehat{\theta_1}, \widehat{\theta_2}).$$

Capitolo 3. Applicazioni a casi di studio

3.1 Marcatori tumorali

Si supponga che un nuovo marcatore tumorale A sia stato scoperto in uno studio in vitro e che si voglia valutare la sua efficacia nel separare individui malati (ad esempio di uno specifico tumore) da soggetti sani. Vengono reclutati 58 pazienti, afferenti ad un centro, di cui 30 sono risultati affetti da tale patologia e 28 no, grazie all'utilizzo di un *golden test* (test ad alta affidabilità).

I valori di tale marcatore, misurati nel plasma dei soggetti, appartenenti ai due gruppi sono riportati nella seguente tabella (Bottarelli, Parodi, 2003).

Malati		Non-Malati	
ID soggetto	Marcatore A	ID soggetto	Marcatore A
1	23.7	31	23.8
2	25.4	32	35.9
3	23.2	33	30.2
4	32.4	34	20.6
5	24.4	35	13.7
6	48.8	36	19.5
7	42.1	37	36.3
8	31.5	38	20.3
9	54.2	39	20.7
10	53.7	40	22.4
11	40.1	41	22.9
12	23.3	42	15.7
13	27.6	43	33.6
14	39.9	44	27.3
15	52.2	45	34.8
16	26.0	46	30.0
17	16.3	47	25.0
18	23.5	48	41.3
19	22.6	49	26.9
20	42.9	50	21.3
21	22.3	51	11.6
22	36.5	52	39.2
23	29.0	53	18.3
24	50.8	54	28.7
25	42.6	55	10.8
26	36.0	56	28.5
27	36.8	57	33.2
28	35.0	58	25.3
29	19.2		
30	40.6		

Per valutare la performance del nuovo marcatore viene utilizzata l'analisi ROC, utilizzando il *software* R.

Per prima cosa costruiamo in R le tre variabili: *labels_A*, *X_A* e *Y_A*. La prima conterrà il vero stato del paziente, codificato con 0 (non-malato) e con 1 (malato).

La variabile *X_A* è un vettore contenente i valori del marcatore A per i pazienti non-malati, e viceversa, la variabile *Y_A* contiene i valori del marcatore A per i pazienti malati. In R:

```
> X_A= c(23.8, 35.9, 30.2, 20.6, 13.7, 19.5, 36.3, 20.3, 20.7, 22.4, 22.9, 15.7, 33.6, 27.3, 34.8, 30, 25, 41.3, 26.9, 21.3, 11.6, 39.2, 18.3, 28.7, 10.8, 28.5, 33.2, 25.3)
```

```
> Y_A= c(23.7, 25.4, 23.2, 32.4, 24.4, 48.8, 42.1, 31.5, 54.2, 53.7, 40.1, 23.3, 27.6, 39.9, 52.2, 26, 16.3, 23.5, 22.6, 42.9, 22.3, 36.5, 29, 50.8, 42.6, 36, 36.8, 35, 19.2, 40.6)
```

```
> labels_A= c(rep(0,length(X_A)), rep(1,length(Y_A)))
[1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1
[39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

Come visto nel primo capitolo, quando si considerano i risultati di un particolare test diagnostico su due gruppi, un gruppo di pazienti malati e un secondo gruppo di pazienti non-malati, in genere è difficile osservare una separazione perfetta tra i due gruppi, ma in genere si osserva una sovrapposizione delle distribuzioni (Figura 9).

Attraverso il test di Shapiro-Wilk (1965) verifichiamo l'ipotesi di normalità per le variabili *X_A* e *Y_A,* ottenendo i seguenti valori:

```
> shapiro.test (X_A)
```

```
        Shapiro-Wilk normality test

data:  X_A
W = 0.9819, p-value = 0.8922

> shapiro.test (Y_A)
        Shapiro-Wilk normality test

data:  Y_A
W = 0.9406, p-value = 0.09431
```

che porta ad accettare l'ipotesi nulla per entrambe le variabili.

Nel nostro caso di studio, le distribuzioni delle misurazioni del test diagnostico nei due gruppi sono illustrate nella Figura 9. Le due distribuzioni normali sono stimate con

```
> boxplot(X_A,Y_A, names=c("non-malati", "malati"))
> curve(dnorm(x,mean(X_A), sd(X_A)), 0,60, ylab="", xlab="")
> curve(dnorm(x, mean(Y_A), sd(Y_A)), 0,60, add=TRUE, lty=2, ylab="", xlab="")
```

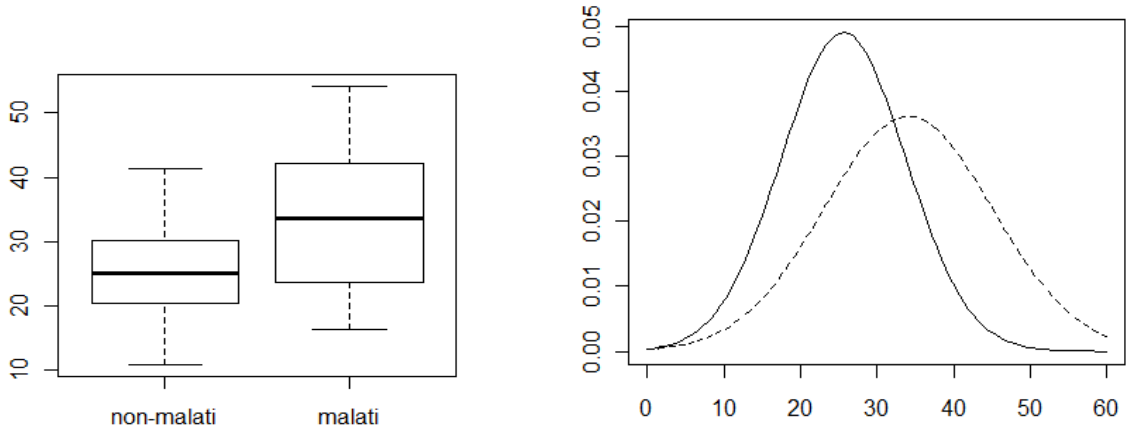

Figura 9. Distribuzione delle misurazioni del marcatore A nel gruppo dei pazienti non-malati (linea continua), e il gruppo dei pazienti malati (linea tratteggiata).

Dalla Figura 9 si nota che, per ogni valore di *cut-off* selezionato per discriminare tra i due gruppi di pazienti, ci saranno dei pazienti malati classificati come positivi (TP), ma alcuni dei pazienti malati saranno classificati come negativi (FN). Analogamente, alcuni pazienti non-malati saranno correttamente classificati come negativi (TN), ma alcuni non-malati saranno classificati come positivi (FP).

Utilizzando la libreria ROCR possiamo ottenere i quattro valori TP, TN, FP e FN per diversi valori di *cut-off*:

```
> library(ROCR)
> predictions_A=c(X_A,Y_A)
> pred = prediction(predictions_A, labels_A)
```

Richiamando l'oggetto pred si ottiene il seguente output di R:

```
> pred
An object of class "prediction"
```

Slot "predictions":

[1] 23.8 35.9 30.2 20.6 13.7 19.5 36.3 20.3 20.7 22.4 22.9 15.7 33.6 27.3 34.8

[16] 30.0 25.0 41.3 26.9 21.3 11.6 39.2 18.3 28.7 10.8 28.5 33.2 25.3 23.7 25.4

[31] 23.2 32.4 24.4 48.8 42.1 31.5 54.2 53.7 40.1 23.3 27.6 39.9 52.2 26.0 16.3

[46] 23.5 22.6 42.9 22.3 36.5 29.0 50.8 42.6 36.0 36.8 35.0 19.2 40.6

Slot "labels":

[1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1

[39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Levels: 0 < 1

Slot "cutoffs":

[1] Inf 54.2 53.7 52.2 50.8 48.8 42.9 42.6 42.1 41.3 40.6 40.1 39.9 39.2 36.8

[16] 36.5 36.3 36.0 35.9 35.0 34.8 33.6 33.2 32.4 31.5 30.2 30.0 29.0 28.7 28.5

[31] 27.6 27.3 26.9 26.0 25.4 25.3 25.0 24.4 23.8 23.7 23.5 23.3 23.2 22.9 22.6

[46] 22.4 22.3 21.3 20.7 20.6 20.3 19.5 19.2 18.3 16.3 15.7 13.7 11.6 10.8

Slot "fp":

[1] 0 0 0 0 0 0 0 0 0 1 1 1 1 2 2 2 3 3 4 5 6 7 7 7

[26] 8 9 9 10 11 11 12 13 13 13 14 15 15 16 16 16 16 16 17 17 18 18 19 20 21

[51] 22 23 23 24 24 25 26 27 28

Slot "tp":

[1] 0 1 2 3 4 5 6 7 8 8 9 10 11 11 12 13 13 14 14 15 15 15 15 16 17

[26] 17 17 18 18 18 19 19 19 20 21 21 21 22 22 23 24 25 26 26 27 27 28 28 28 28

[51] 28 28 29 29 30 30 30 30 30

Slot "tn":

[1] 28 28 28 28 28 28 28 28 28 27 27 27 27 26 26 26 25 25 24 24 23 22 21 21 21

[26] 20 19 19 18 17 17 16 15 15 15 14 13 13 12 12 12 12 12 11 11 10 10 9 8 7

[51] 6 5 5 4 4 3 2 1 0

Slot "fn":

[1] 30 29 28 27 26 25 24 23 22 22 21 20 19 19 18 17 17 16 16 15 15 15 15 14 13

[26] 13 13 12 12 12 11 11 11 10 9 9 9 8 8 7 6 5 4 4 3 3 2 2 2 2

[51] 2 2 1 1 0 0 0 0 0

Slot "n.pos":

[1] 30

Slot "n.neg":

[1] 28

Slot "n.pos.pred":

[1] 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

[26] 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49

[51] 50 51 52 53 54 55 56 57 58

Slot "n.neg.pred":

[1] 58 57 56 55 54 53 52 51 50 49 48 47 46 45 44 43 42 41 40 39 38 37 36 35 34

[26] 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9

[51] 8 7 6 5 4 3 2 1 0

Ad esempio, scegliendo come valore di *cut-off* k=30 (Figura 10, posizione 27 nello *slot* cutoffs), e prendendo i valori di TN, TP, FP, FN in corrispondenza della posizione 27, si ottiene la seguente matrice di confusione

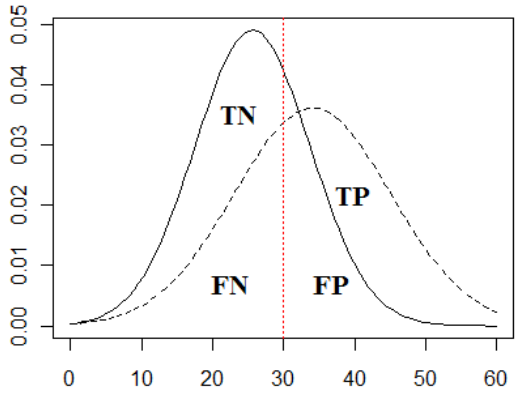

Figura 10. Distribuzioni delle misurazioni del marcatore A sui pazienti non-malati (linea continua) e malati (linea tratteggiata). La linea rossa rappresenta il valore di *cut-off* k=30.

	Paziente Malato	Paziente non-Malato	Totale
Test positivo	17	9	26
Test negativo	13	19	32
Totale	30	28	

mentre per k= 22.4 (Figura 11, posizione 46 nello *slot* cutoffs) si ottiene

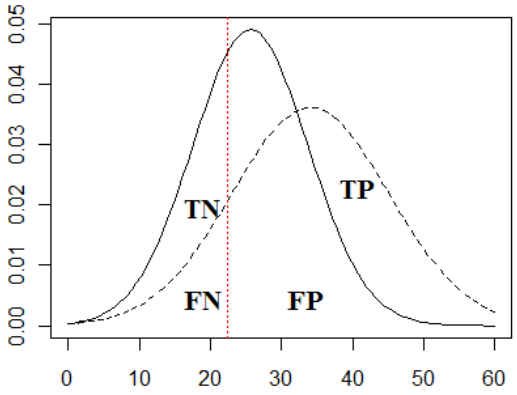

Figura 11. Distribuzioni delle misurazioni del marcatore A sui pazienti non-malati (linea continua) e malati (linea tratteggiata). La linea rossa rappresenta il valore di *cut-off k*=22.4.

	Paziente Malato	Paziente non-Malato	Totale
Test Positivo	27	18	45
Test negativo	3	10	13
Totale	30	28	

Ad ogni valore di *cut-off* corrisponde quindi, una matrice di confusione diversa.

A partire da queste classificazioni, si possono ottenere due importanti indici sintetici della qualità della classificazione: la sensibilità e la specificità. In R:

```
> perf = performance (pred, "tpr", "tnr")
```

L'output di R è il seguente:

```
> perf
An object of class "performance"
Slot "x.name":
"True negative rate"
```

Slot "y.name":

"True positive rate"

Slot "alpha.name":

"Cutoff"

Slot "x.values":

[1] 1.00000000 1.00000000 1.00000000 1.00000000 1.00000000 1.00000000

[7] 1.00000000 1.00000000 1.00000000 0.96428571 0.96428571 0.96428571

[13] 0.96428571 0.92857143 0.92857143 0.92857143 0.89285714 0.89285714

[19] 0.85714286 0.85714286 0.82142857 0.78571429 0.75000000 0.75000000

[25] 0.75000000 0.71428571 0.67857143 0.67857143 0.64285714 0.60714286

[31] 0.60714286 0.57142857 0.53571429 0.53571429 0.53571429 0.50000000

[37] 0.46428571 0.46428571 0.42857143 0.42857143 0.42857143 0.42857143

[43] 0.42857143 0.39285714 0.39285714 0.35714286 0.35714286 0.32142857

[49] 0.28571429 0.25000000 0.21428571 0.17857143 0.17857143 0.14285714

[55] 0.14285714 0.10714286 0.07142857 0.03571429 0.00000000

Slot "y.values":

[1] 0.00000000 0.03333333 0.06666667 0.10000000 0.13333333 0.16666667

[7] 0.20000000 0.23333333 0.26666667 0.26666667 0.30000000 0.33333333

[13] 0.36666667 0.36666667 0.40000000 0.43333333 0.43333333 0.46666667

[19] 0.46666667 0.50000000 0.50000000 0.50000000 0.50000000 0.53333333

[25] 0.56666667 0.56666667 0.56666667 0.60000000 0.60000000 0.60000000

[31] 0.63333333 0.63333333 0.63333333 0.66666667 0.70000000 0.70000000

[37] 0.70000000 0.73333333 0.73333333 0.76666667 0.80000000 0.83333333

[43] 0.86666667 0.86666667 0.90000000 0.90000000 0.93333333 0.93333333

[49] 0.93333333 0.93333333 0.93333333 0.93333333 0.96666667 0.96666667

[55] 1.00000000 1.00000000 1.00000000 1.00000000 1.00000000

Slot "alpha.values":

[1] Inf 54.2 53.7 52.2 50.8 48.8 42.9 42.6 42.1 41.3 40.6 40.1 39.9 39.2 36.8

[16] 36.5 36.3 36.0 35.9 35.0 34.8 33.6 33.2 32.4 31.5 30.2 30.0 29.0 28.7 28.5

[31] 27.6 27.3 26.9 26.0 25.4 25.3 25.0 24.4 23.8 23.7 23.5 23.3 23.2 22.9 22.6

[46] 22.4 22.3 21.3 20.7 20.6 20.3 19.5 19.2 18.3 16.3 15.7 13.7 11.6 10.8

Come descritto nel Capitolo 1, i valori di sensibilità e specificità dipendono ovviamente dalla soglia *k* fissata nella classificazione. Si noti che, quando si fissa un valore elevato di *k*, i Veri Positivi (TP) e la sensibilità decrescono, con la specificità e i Veri Negativi (TN) che aumentano. Viceversa, se si seleziona un valore basso di *k*, i Veri Positivi (TP) e la sensibilità aumentano, e i Veri Negativi e la specificità diminuiscono.

Cut-off	Specificità	Sensibilità
54.2	1.000	0.033
42.6	1.000	0.233
35.0	0.857	0.500
30.0	0.679	0.567
22.4	0.357	0.900
18.3	0.143	0.967
10.8	0.000	1.000

Tabella 3. Alcuni valori di sensibilità e specificità in corrispondenza dei valori di *cut-off*

Per costruire la curva ROC (Figura 12), per il calcolo dell'AUC e per valutare la performance discriminatoria del marcatore A, utilizzeremo il pacchetto verification, molto più rapido ed efficace rispetto a pROC e ROCR. In R:

```
> library (verification)
> roc.plot (labels_A, predictions_A, thresholds=NULL)
```

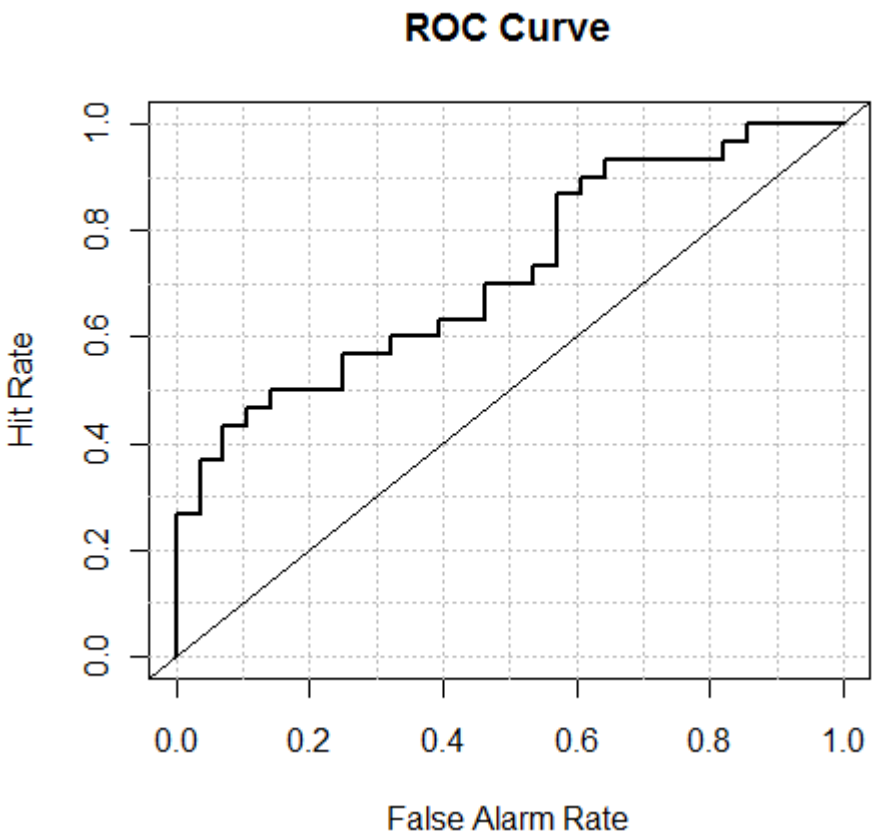

Figura 12. Curva ROC corrispondente al marcatore A.

```
> roc.area (labels, predictions)
```

A

[1] 0.7202381

n.total

[1] 58

n.events

[1] 30

n.noevents

[1] 28

p.value

[1] 0.001785681

L'area sotto la curva ROC risulta pari a 0.720, per cui il marcatore risulterebbe moderatamente accurato, secondo la classificazione di Swets (1998).

Si perviene allo stesso risultato utilizzando la relazione tra la statistica U di Wilcoxon e l'AUC:

> wilcox.test (X_A,Y_A)

Wilcoxon rank sum test

data: X_A and Y_A

W = 235, p-value = 0.003571

alternative hypothesis: true location shift is not equal to 0

L' AUC risulta essere

$$AUC = 1 - \frac{235}{30*28} = 0.7202381.$$

Sotto ipotesi parametriche si può ottenere una stima dell'AUC sfruttando la distribuzione normale delle due variabili X_A, Y_A:

```
> mux= mean(X_A)
> muy=mean(Y_A)
> mux
[1] 25.63571
> muy
[1] 34.08667
> sigma2y_hat= length(Y_A)*var(Y_A)/length(Y_A)
> sigma2y_hat
[1] 121.8419
> sigma2x_hat= length(X_A)*var(X_A)/length(X_A)
> sigma2x_hat
[1] 66.05497
> auc_hat= pnorm((muy-mux)/ sqrt( sigma2y_hat+sigma2x_hat))
> auc_hat
[1] 0.7312237
```

La funzione roc.area ci ha fornito inoltre il p-value del test per valutare la performance del marcatore A. Il p-value è pari a 0.0018, questo ci porta a rifiutare l'ipotesi nulla (H_0: AUC=0.5), vista nel paragrafo 1.4.

Si arriva agli stessi risultati calcolando a mano il valore della statistica test Z.

Occorre quindi calcolare σ^2_{AUC}, e applicando il metodo di Hanley e McNeil (vedi paragrafo 1.4) si ottiene:

$$Q_1 = \frac{0.720}{2-0.720} = 0.563 \qquad Q_2 = \frac{2*0.720\text{^}2}{1+0.720} = 0.603$$

$$\sigma^2_{AUC} = \frac{0.720*(1-0.720)+(30-1)*(0.563-0.720^2)+(28-1)*(0.603-0.720^2)}{30*28} = 0.00449$$

Il test Z è pari a

$$Z = \frac{0.720-0.5}{\sqrt{0.00449}} = 3.284 .$$

Il valore ottenuto eccede il valore 1.96 ($\alpha<0.05$), per cui si può affermare che i valori del marcatore differiscono significativamente nei due gruppi.

E' possibile ottenere anche intervalli di confidenza per l'AUC utilizzando la libreria pROC:

```
> library(pROC)
> roc_A= roc(labels_A, predictions_A, auc=TRUE)
> ci.auc(roc_A, conf.level=0.95, reuse.auc=TRUE)
        95% CI: 0.5895-0.851 (DeLong)
> ci.auc(roc_A, conf.level=0.95,method="bootstrap", boot.n=2000, boot.stratified=TRUE,
reuse.auc=TRUE)
        95% CI: 0.5928-0.8369 (2000 stratified bootstrap replicates)
```

Si supponga ora che il nuovo marcatore A che stiamo analizzando, possa essere utilizzato in sostituzione di un precedente marcatore B. Si applica quindi un confronto tra le due curve ROC, utilizzando i valori della Tabella 4.

Malati			Non-Malati		
ID soggetto	Marcatore A	Marcatore B	ID soggetto	Marcatore A	Marcatore B
1	23.7	63.0	31	23.8	48.2
2	25.4	63.1	32	35.9	38.5
3	23.2	59.2	33	30.2	22.5
4	32.4	71.4	34	20.6	22.2
5	24.4	68.5	35	13.7	33.1
6	48.8	69.9	36	19.5	24.3
7	42.1	59.7	37	36.3	39.0
8	31.5	67.1	38	20.3	18.1
9	54.2	74.5	39	20.7	32.4
10	53.7	93.3	40	22.4	31.3
11	40.1	58.5	41	22.9	36.3
12	23.3	40.8	42	15.7	18.4
13	27.6	60.0	43	33.6	53.7
14	39.9	82.6	44	27.3	22.1
15	52.2	88.2	45	34.8	42.4
16	26.0	75.7	46	30.0	43.3
17	16.3	53.6	47	25.0	28.6
18	23.5	80.8	48	41.3	71.0
19	22.6	60.1	49	26.9	45.5
20	42.9	95.3	50	21.3	57.2
21	22.3	74.0	51	11.6	28.0
22	36.5	63.4	52	39.2	69.2
23	29.0	61.5	53	18.3	17.9
24	50.8	80.6	54	28.7	23.5
25	42.6	80.8	55	10.8	16.0
26	36.0	73.8	56	28.5	33.7
27	36.8	68.2	57	33.2	15.1
28	35.0	89.7	58	25.3	31.0
29	19.2	53.2			
30	40.6	67.6			

Tabella 4. Valori dei marcatori tumorali A e B

Come per il marcatore A, anche per il marcatore B creiamo le variabili: *labels_B*, *X_B*, *Y_B*. In R:

```
> X_B= c(48.2, 38.5, 22.5, 22.2, 33.1, 24.3, 39.0, 18.1, 32.4, 31.3, 36.3, 18.4, 53.7, 22.1, 42.4, 43.3,
28.6, 71, 45.5, 57.2, 28, 69.2, 17.9, 23.5, 16, 33.7, 15.1, 31)
```

```
> Y_B= c(63, 63.1, 59.2, 71.4, 68.5, 69.9, 59.7, 67.1, 74.5, 93.3, 58.5, 40.8, 60, 82.6, 88.2, 75.7,
53.6, 80.8, 60.1, 95.3, 74, 63.4, 61.5, 80.6, 80.8, 73.8, 68.2, 89.7, 53.2, 67.6)
```

```
> labels_B= c( rep(0,length(X_B)), rep(1,length(Y_B)))
[1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1
```

[39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Le due curve ROC corrispondenti ai due marcatori sono riportate nella Figura 13. In R:

```
> predictions_B= c(X_B, Y_B)
> roc_A= roc(labels_A, predictions_A,plot=TRUE)
> roc_B= roc(labels_B, predictions_B,plot=TRUE, add=TRUE,lty=2)
```

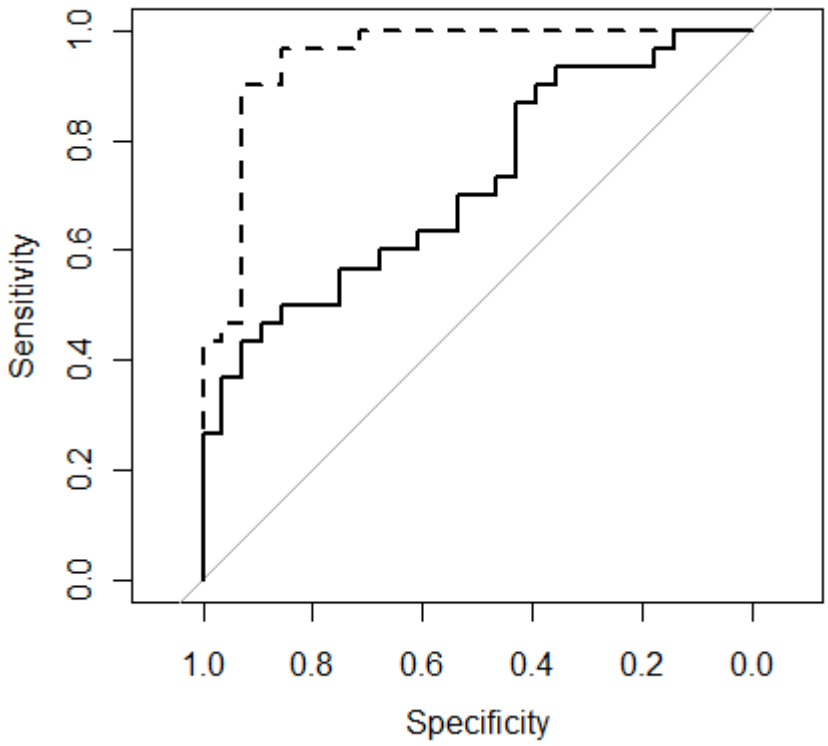

Figura 13. Confronto tra la curva ROC del marcatore A (linea continua), e la curva ROC del marcatore B (linea tratteggiata).

Il marcatore B, presenta quindi una migliore capacità discriminante, poiché la curva ROC corrispondente si trova interamente al di sopra di quella generata dal marcatore A. L'AUC del marcatore B risulta essere pari a:

```
> roc(labels_B, predictions_B,auc=TRUE)
```

Call:

roc.default(response = labels_B, predictor = predictions_B, auc = TRUE)

Data: predictions_B in 28 controls (response 0) < 30 cases (response 1).

Area under the curve: 0.9488

Verifichiamo se la performance del marcatore B supera, a livello α=0.05, la performance del marcatore A. Per il confronto tra i due marcatori, utilizziamo la libreria pROC, l'unica che implementa questa funzione. Il test d'ipotesi che andiamo a verificare è il seguente

- $H_0 : AUC_B - AUC_A = 0$
- $H_1 : AUC_B - AUC_A \neq 0$

In R:

```
> roc.test (roc_B,roc_A, alternative = "two.sided")

        DeLong's test for two correlated ROC curves

data:  roc_B and roc_A
Z = 3.8574, p-value = 0.0001146
alternative hypothesis: true difference in AUC is not equal to 0
sample estimates:
AUC of roc1 AUC of roc2
  0.9488095   0.7202381
```

Il valore della statistica che si ottiene è Z= 3.8574, con relativo p-value= 0.00011, il che ci porta a

rifiutare l'ipotesi nulla, per cui si può affermare che la *performance* del marcatore B supera quella del nuovo marcatore A (con α<0.05).

3.2 Dati Linfoma Anaplastico a Grandi Cellule.

I valori contenuti nel *dataset* oggetto di studio sono stati raccolti dal Centro Oncoematologico Pediatrico di Padova e si dividono in due gruppi: uno di malati (variabile *casi*) e uno di sani (variabile *controlli*). I dati sono:

```
> controlli=c (0.23, 0.44, 0.19, 0.08)
> casi= c(2.8, 1.4, 0.13, 0.2, 0.8, 0.56, 0.44, 5.2, 1.7, 1.14)
```

Il Centro voleva valutare se la quantità di una certa proteina (Hsp70) presente nel paziente potesse essere un indicatore in grado di discriminare i sani dai malati di Linfoma Anaplastico a Grandi Cellule, un particolare tipo di cancro. Non solo sembra che i pazienti malati abbiano un livello di Hsp70 maggiore rispetto ai sani (come mostra il *boxplot*, Figura 14), ma la presenza di tale proteina sembra diminuire l'efficacia della terapia.

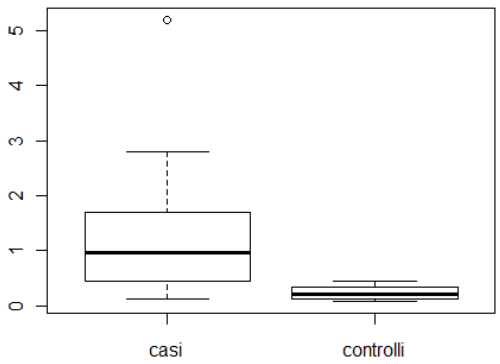

Figura 14. Boxplot dei livelli della proteina Hsp70 nei due gruppi di pazienti.

Un'ipotesi sperimentale d'interesse consiste nel valutare se il livello della proteina Hsp70 discrimina i due gruppi. Questo problema può essere riformulato in termini di AUC.

Il livello di proteina nel paziente è stato rilevato su scala continua e le osservazioni sono indipendenti.

Costruiamo la curva ROC (Figura 15), costruendo le variabili *labels* e *predictions*:

```
> predictions= c(controlli,casi)
> labels=c(rep(0,length(controlli)),rep(1,length(casi)))
> library(pROC)
> roc.(labels, predictions,plot=TRUE)
```

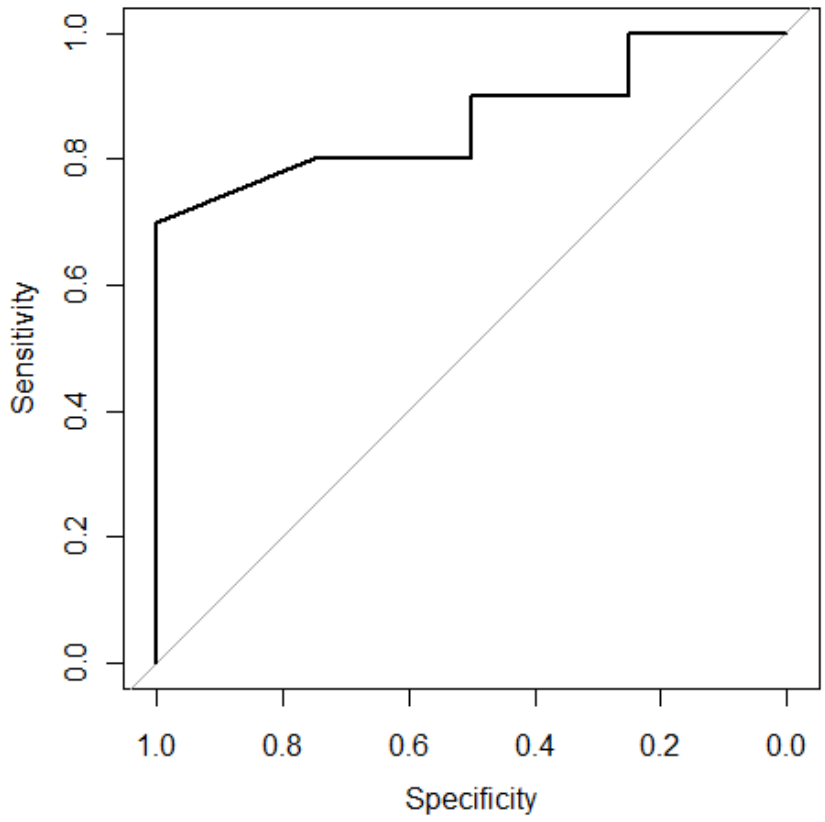

Figura 15. Analisi con la curva ROC per i livelli di proteina Hsp70.

Procediamo al calcolo dell'area sottesa alla curva:

```
> library(verification)
> roc.area(labels,predictions)
A
[1] 0.8625
n.total
[1] 14
n.events
[1] 10
n.noevents
[1] 4
p.value
[1] 0.02373524
```

L'AUC risulta pari a 0.8625, per cui la quantità di proteina Hsp70 presente nel paziente, risulta essere un indicatore moderatamente accurato (Swets, 1988).

Si perviene allo stesso risultato utilizzando la relazione tra la statistica U di Wilcoxon e l'AUC:

```
> wilcox.test(controlli,casi)
        Wilcoxon rank sum test with continuity correction
data:    controlli and casi
W = 5.5, p-value = 0.04747
alternative hypothesis: true location shift is not equal to 0
```

L'AUC risulta essere

$$\text{AUC} = 1 - \frac{5.5}{10*4} = 0.8625$$

Sotto ipotesi parametriche è possibile calcolare l'AUC sfruttando la distribuzione delle variabili casi e controlli. Attraverso il test non parametrico di Kolgomorov-Smirnov è possibile fare delle assunzioni sulla distribuzione delle variabili d'interesse.

In pratica il problema d'ipotesi è del tipo:

$$\begin{cases} H_0: F(x) = F_0(x) & per\ qualsiasi\ x \\ H_1: F(x) \neq F_0(x) & per\ qualche\ x \end{cases}$$

con $F_0(x)$ la distribuzione che vogliamo verificare.

Questo significa che l'ipotesi non si riferisce soltanto ad un parametro della variabile casuale *X*, ma l'intera sua funzione di ripartizione.

Nel nostro caso verifichiamo che le due variabili casi e controlli si distribuiscono in maniera esponenziale. In R:

```
> m_casi= mean(casi)
> 1/m_casi
[1] 0.6958942
> ks.test (casi,"pexp",0.695)
        One-sample Kolmogorov-Smirnov test
data:  casi
D = 0.1068, p-value = 0.999
alternative hypothesis: two-sided
> m_controlli= mean(controlli)
```

```
> 1/m_controlli
[1] 4.255319
> ks.test(controlli,"pexp",4.25)
        One-sample Kolmogorov-Smirnov test

data:  controlli
D = 0.304, p-value = 0.7582
alternative hypothesis: two-sided
```

Per entrambe le due variabili non si rifiuta l'ipotesi nulla a livello α= 0.05.

Sotto assunzioni di distribuzione esponenziale delle variabile casi e controlli, una stima parametrica dell'AUC si ottiene con:

```
> auc= m_casi/ (m_casi + m_controlli)
> auc
[1] 0.8594498
```

La funzione roc.area ci ha fornito inoltre il p-value del test per valutare la performance, della quantità di proteina Hsp70 nel paziente, come indicatore in grado di discriminare pazienti sani e malati di Linfoma Anaplastico a Grandi Cellule. Il p-value è pari a 0.024, questo ci porta a rifiutare l'ipotesi nulla (H_0 : AUC=0.5), vista nel paragrafo 1.4.

Si può pertanto concludere che il livello della proteina è diverso nei due gruppi e pertanto discrimina sufficientemente bene i due gruppi.

E' possibile determinare, inoltre, un valore soglia *k* che discrimini tra i due gruppi di pazienti, che risponda ad un qualche criterio ottimale visto nel paragrafo 1.6. In R:

```
> library(pROC)
> roc_Hsp70=roc(labels, predictions, auc=TRUE)
> coords(roc_Hsp70, x="best",best.method="youden")
          threshold specificity sensitivity
0.5               1.0               0.7
> coords(roc_Hsp70, x="best",best.method="closet.topleft")
          threshold specificity sensitivity
0.5               1.0               0.7
```

L'output di R ci mostra che il valore di *cut-off* ottimale è k=0.5, sia per l'indice di Youden, sia per il criterio della minima distanza.

BIBLIOGRAFIA

- Bamber, D. (1975), *The Area above the Ordinal Dominance Graph and the Area below the Receiver Operating Characteristic Graph*, J. Math. Psychol.
- Barajas-Rojas, J.A., Riemann, H.P., Franti, C.E. (1993), *Notes about determining the cut-off value in enzyme-linked immunosorbent assay (ELISA).* Prev . Vet. Med., 15, 231-3.
- Bland, M. (2009), *Statistica Medica*, Apogeo, Milano.
- Bottarelli, E., Parodi, S., (2003), *Un approccio per la valutazione della validità dei test diagnostici: le curve R.O.C. (Receiver Operating Characteristic)*.
- Carpenter, J., Bithell, J. (2000), *Bootstrap condence intervals: when, which, what? A practical guide for medical statisticians*, Statistics in Medicine 19, 1141 – 1164.
- DeLong, E.R., DeLong, D.M., Clarke-Pearson, D.L. (1988), *Comparing the areas under two or more correlated receiver characteristic curves: a nonparametric approach*, Biometrics 44, 837- 845.
- Erdreich, L.S. (1981), *Use of Relative Operating Characteristic analysis in Epidemiology*,Am. J.
- Goodenough, D.J., Rossmann, K., Lusted, L.B. (1974), *Radiographic applications of receiver operating characteristic (ROC) analysis*.
- Greiner, M., Pfeiffer, D., Smith, R.D. (2000), *Principles and practical application* of the receiver-operating characteristic analysis for diagnostic tests.
- Hanley, J., McNeil, B.J. (1982), *The meaning and use of the area under a receiver operating characteristic (ROC) curve*.
- Hanley, J.A., McNeil, B. J. (1983), *A method of comparing the areas under Receiver Operating Characteristic curves derived from the same cases.*
- Kotz, S., Lumelskii, Y., Pensky, M. (2003), *The Stress-Strength Model and its Generalizations. Theory and Applications.* World Scientific, Singapore.

- Krzanowski, W.J., Hand, D.J. (2009), *Roc Curves for Continuos Data*, Chapman & Hall, London.
- Metz, C.E (1978), Basic principles of ROC analysis, Seminar Nuclear Medicine; Vol VIII, No. 4:283-29.
- NCAR - Research Application Program (2010), *Package 'verification'*.
- Pace, L., Salvan, A. (2001), *Introduzione alla Statistica: Inferenza,verosomiglianza, modelli,* Cedam, Padova.
- Pepe,M.S. (2003), *The statistical evaluation of medical tests for classification and prediction*, Oxford University Press: New York; 28.
- Robin, X., Turck, N., Hainard, A., Tiberti, N., Lisacek, F., Sanchez, J.C., Müller, M. (2011), *Package 'pROC'*.
- Sing, T., Sander, O., Beerenwinkel, N., Lengauer, T. (2009), *Package 'ROCR'*.
- Swets, J.A. (1998), *Measuring the accuracy of diagnostic systems.*
- Venkatraman, E.S. (2000), *A Permutation Test to Compare Receiver Operating Characteristic Curves*, Biometrics 56, 1134 – 1138.
- Zhou, X.H., Obuchowsky, N.A., McClish, D.K. (2002), *Statistical methods in diagnostic medicine,* Wiley: New York.
- Zweig, H.H., Campbell, G. (1993), *Receiver Operating Characteristic (ROC) plots: a fundamental evolution tool in medicine.*