
ANNO ACCADEMICO 2024/2025

Apprendimento Automatico

Teoria

Altair's Notes



UNIVERSITÀ
DI TORINO



DIPARTIMENTO DI INFORMATICA

CAPITOLO 1

INTRODUZIONE

PAGINA 5

1.1	Le basi del machine learning Tasks — 7 • Modelli — 8 • Features — 9	5
1.2	Tasks classificazione — 10 • Scoring e ranking — 15	9

CAPITOLO 2

TEST2

PAGINA 17

Premessa

Licenza

Questi appunti sono rilasciati sotto licenza Creative Commons Attribuzione 4.0 Internazionale (per maggiori informazioni consultare il link: <https://creativecommons.org/version4/>).



Formato utilizzato

Box di "Concetto sbagliato":

Concetto sbagliato 0.1: Testo del concetto sbagliato

Testo contenente il concetto giusto.

Box di "Corollario":

Corollario 0.0.1 Nome del corollario

Testo del corollario. Per corollario si intende una definizione minore, legata a un'altra definizione.

Box di "Definizione":

Definizione 0.0.1: Nome delle definizioni

Testo della definizione.

Box di "Domanda":

Domanda 0.1

Testo della domanda. Le domande sono spesso utilizzate per far riflettere sulle definizioni o sui concetti.

Box di "Esempio":

Esempio 0.0.1 (Nome dell'esempio)

Testo dell'esempio. Gli esempi sono tratti dalle slides del corso.

Box di "Note":

Note:-

Testo della nota. Le note sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive.

Box di "Osservazioni":

Osservazioni 0.0.1

Testo delle osservazioni. Le osservazioni sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive. A differenza delle note le osservazioni sono più specifiche.

1

Introduzione

1.1 Le basi del machine learning

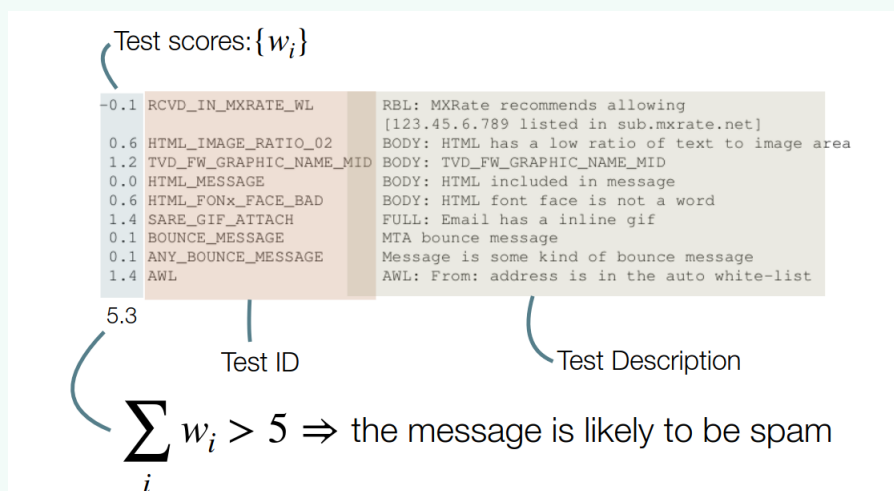
Gli ingredienti del machine learning:

- ⇒ *Task*: specifica di cosa si vuole fare;
- ⇒ *Modelli*: il modello matematico per affrontare un determinato task;
- ⇒ *Features*: il modo con cui sono descritti gli esempi.

Note:-

L'*apprendimento automatico* ruota attorno all'idea di estrarre una regola generale per risolvere un problema a partire da problemi già risolti.

Esempio 1.1.1 (Etichettatura delle email spam)



SpamAssassin è un filtro open-source usato per filtrare lo spam. Esso non lavora sul testo, ma su alcune *feature* della mail.

E-mail	x_1	x_2	Spam?	$4x_1 + 4x_2$
1	1	1	1	8
2	0	0	0	0
3	1	0	0	4
4	0	1	0	4

Discrimination rule example: $\text{Spam}(x) = 4x_1 + 4x_2 > 5$

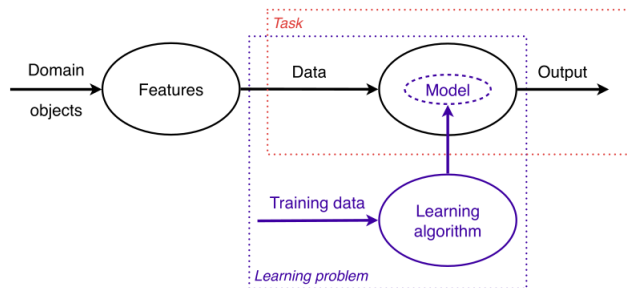
Definizione 1.1.1: Apprendimento automatico

L'apprendimento automatico è lo studio sistematico di algoritmi e sistemi che migliorano le loro conoscenze e performance con l'esperienza.

L'apprendimento automatico è interessato a usare le giuste features per costruire il giusto modello per ottenere buone performance sul giusto task.

Domanda 1.1

L'apprendimento automatico come può aiutarci a risolvere un task?



Dal dominio dell'applicazione arrivano degli oggetti descritti tramite features che vengono utilizzate per creare dei *training data* e un *dataset*. Questi vengono usati per costruire un modello per calcolare un output.

Note:-

Per risolvere un task bisogna sfruttare un modello. Per risolvere un problema di apprendimento bisogna trovare un algoritmo di apprendimento.

1.1.1 Tasks

Definizione 1.1.2: Tasks predittivi

Un task predittivo è focalizzato sul predire una variabile sulla base degli esempi. Si parte da problemi vecchi per trovare la soluzione a *nuovi* problemi.

Corollario 1.1.1 Overfitting

L'Overfitting è un adattamento eccessivo al dataset di allenamento per cui, messi di fronte a nuovi problemi, non si riesce a trovare una soluzione soddisfacente.

I tasks predittivi possono essere:

- *binari e multi-classe*: di categorizzazione;
- *Regressivi*: con un target numerico;
- *Clustering*: un target sconosciuto.

Note:-

IL Clustering fa anche parte dei tasks descrittivi.

Definizione 1.1.3: Tasks descrittivi

Un task descrittivo si concentra sul fornire regolarità nel dataset.

	Films			
Users	1	0	1	0
	0	2	2	2
	0	0	0	1
	1	2	3	2
	1	0	1	1
	0	2	2	3
	The Shawshank Redemption	The Usual Suspects	The Godfather	The Big Lebowski

Questa matrice rappresenta i voti dati da utenti a dei film. Si vogliono estrapolare le caratteristiche di questi film che hanno generato questi voti. Guardando questa matrice individualmente è difficile, per cui si compone con altre matrici.

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 1 \\ 1 & 2 & 3 & 2 \\ 1 & 0 & 1 & 1 \\ 0 & 2 & 2 & 3 \end{pmatrix} = \begin{matrix} \text{Users} \\ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \end{matrix} \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{matrix} \text{Films} \\ \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{matrix}$$

drama
crime
comedy

The Shawshank Redemption
The Usual Suspects
The Godfather
The Big Lebowski

1.1.2 Modelli

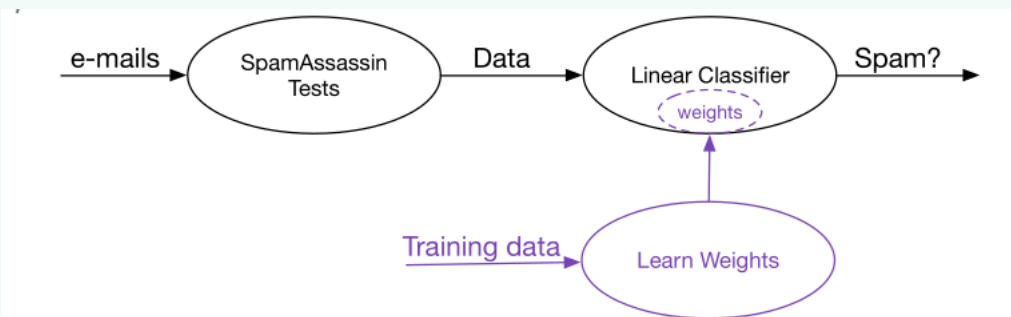
Ci sono 3 possibili tipi di modelli:

- *Geometrici*: modelli che usano l'intuizione dalla geometria per risolvere il problema;
- *Probabilistici*: usano il calcolo delle probabilità;
- *Logici*.

Definizione 1.1.4: Modelli geometrici

Nei modelli geometrici gli esempi sono punti di uno spazio vettoriale e la loro classificazione corrisponde a trovare un iperpiano che separi i punti positivi da quelli negativi.

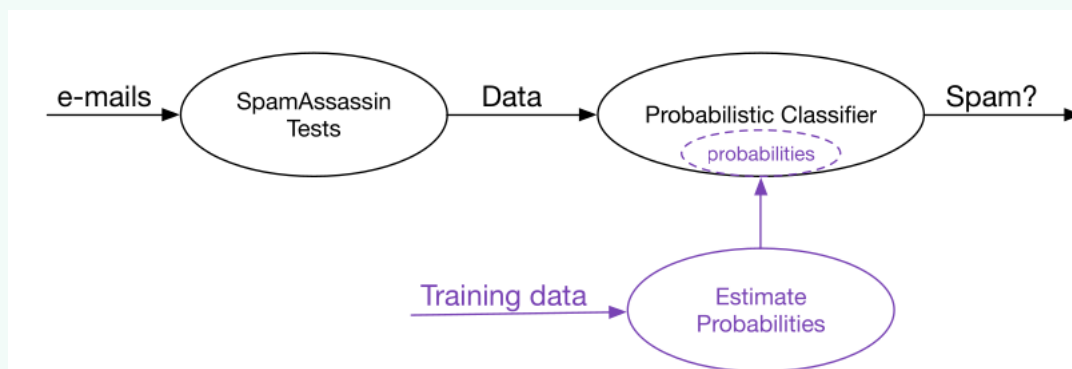
Esempio 1.1.2 (Modello geometrico)



Definizione 1.1.5: Modelli probabilistici

Nei modelli probabilistici si fanno delle stime con dei classificatori probabilistici. Dopo di che si usano delle regole di decisione.

Esempio 1.1.3 (Modello probabilistico)

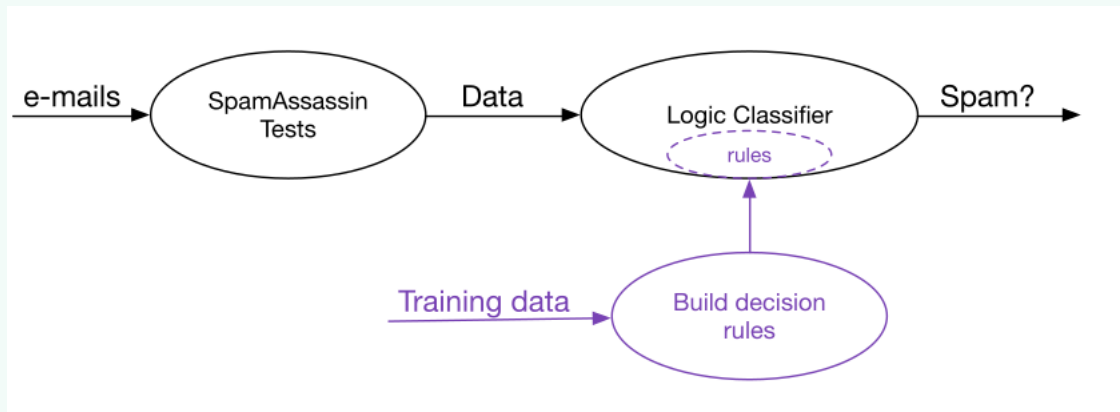


Note:-

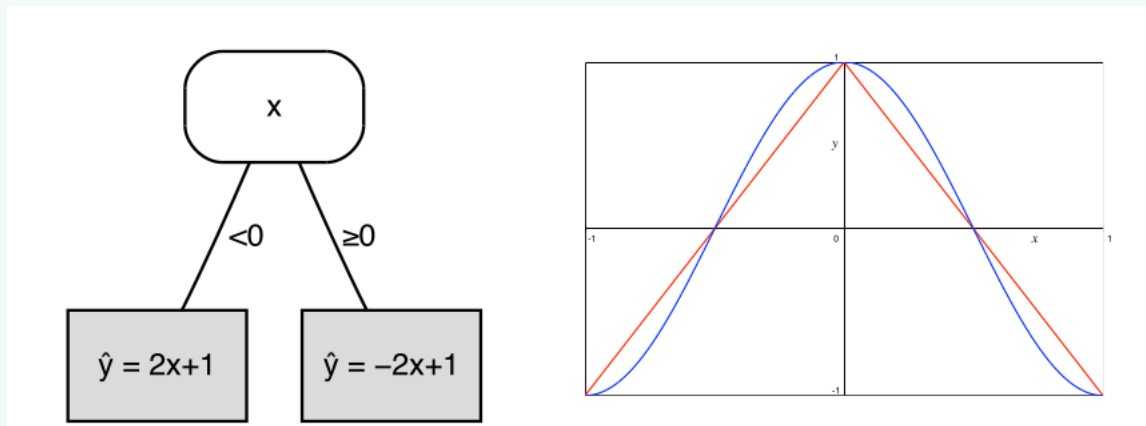
Uno degli algoritmi più semplici che si utilizza con i modelli probabilistici è l'assunzione di Naive Bayes. Si assume che x_1 e x_2 siano indipendenti tra loro per cui si possono calcolare solo i valori di x_1 e di x_2 individualmente.

Definizione 1.1.6: Modelli logici

Nei modelli logici si utilizza la logica. Si hanno una serie di regole.

Esempio 1.1.4 (Modello logico)**1.1.3 Features****Definizione 1.1.7: Features**

Il modo in cui si descrivono i propri dati. Possono facilitare il lavoro di apprendimento se correttamente usate.

Esempio 1.1.5 (Coseno)

Due rappresentazioni della funzione coseno: a sinistra si utilizza una variabile di regressione, a destra un'approssimazione lineare.

1.2 Tasks

I task più comuni sono:

- *classificazione*;
- *Punteggio e classifica*;

- *Stima probabilistica;*
- *Regressione.*

1.2.1 classificazione

Definizione 1.2.1: classificazione

La classificazione è il task in cui si ha come obiettivo la costruzione di un modello $\hat{c}: \mathbf{X} \rightarrow \mathbf{C}$ in cui $\mathbf{C} = \{C_1, C_2, \dots, C_k\}$. Questo modello è un'approssimazione del mondo reale. Un esempio è una coppia $(x, c(x)) \in \mathbf{X} \times \mathbf{C}$.

Osservazioni 1.2.1 Il problema dell'induzione

L'induzione partendo dai dati di un dataset è generalmente infondata senza ulteriori informazioni.

Note:-

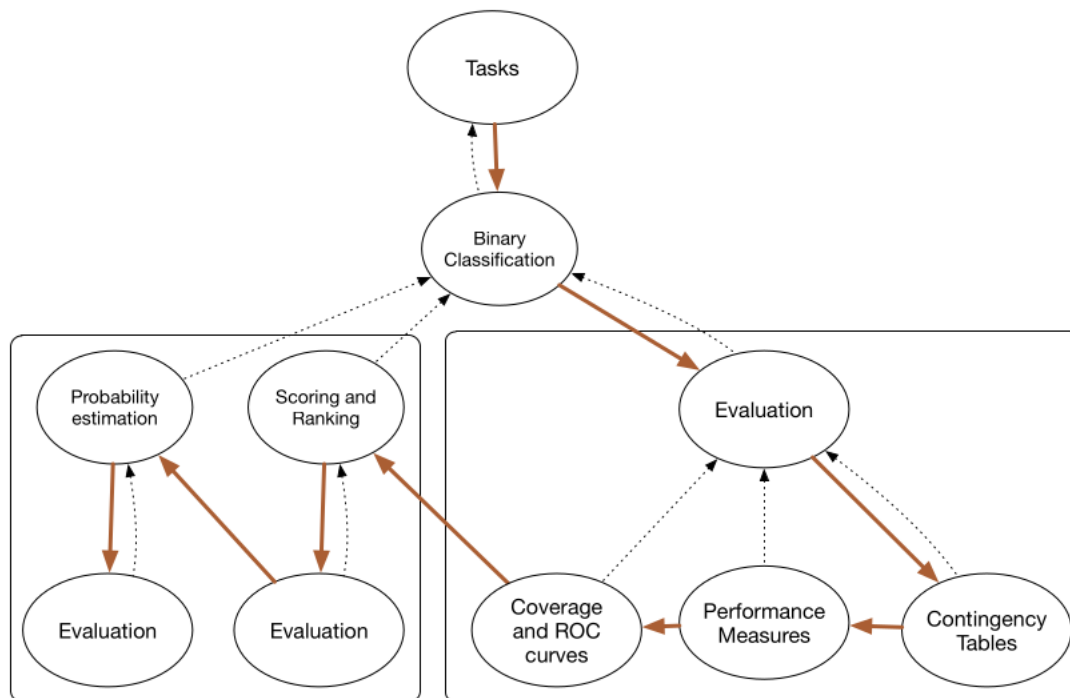
Il mondo non è semplice, per cui il rasoio di Occame non sempre funziona. Spesso però si utilizzano preconcetti e bias induttivi per avere apprendimento automatico.

Definizione 1.2.2: classificazione binaria

La classificazione binaria è il caso in cui si hanno solo 2 opzioni (spesso 0 e 1).

Note:-

Dalla classificazione binaria si può passare alla classificazione multi-classe senza sviluppare nuovi algoritmi.

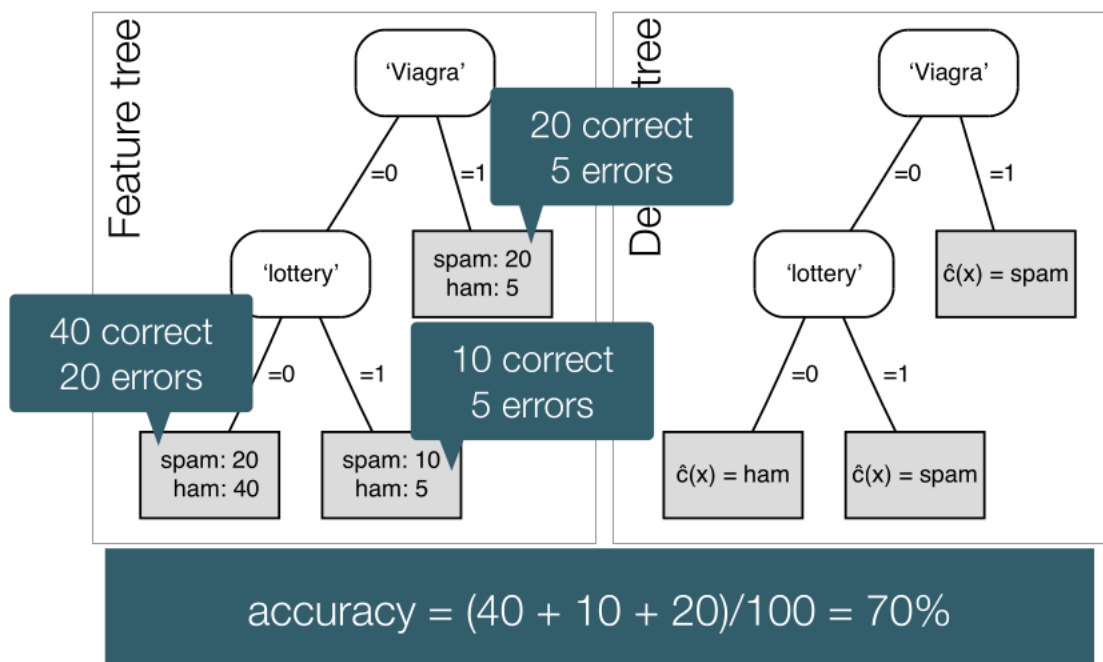


Definizione 1.2.3: Alberi di decisione

Alberi per visualizzare i dati. Ogni nodo corrisponde a una features.

Definizione 1.2.4: Alberi di Features

Alberi per visualizzare i dati. Si ha una suddivisione dei vari esempi divisi per etichette.



Definizione 1.2.5: Tavola di contingenza

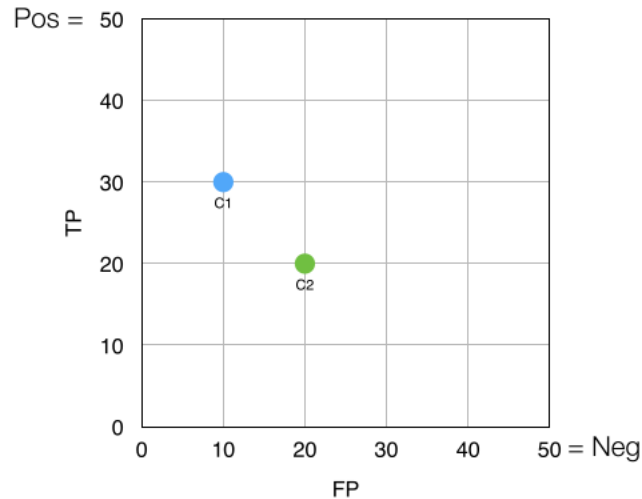
Tavola in cui le colonne corrispondono alle predizioni e le righe al mondo reale. Nella loro intersezione si ha il numero di esempi predetti in un certo modo e hanno una certa etichetta (TP, TN, FT, FN).

	Feature tree	Decision tree	
	Predicted \oplus	Predicted \ominus	
Actual \oplus	30	20	50
Actual \ominus	10	40	50
	40	60	100

Definizione 1.2.6: Grafico di copertura

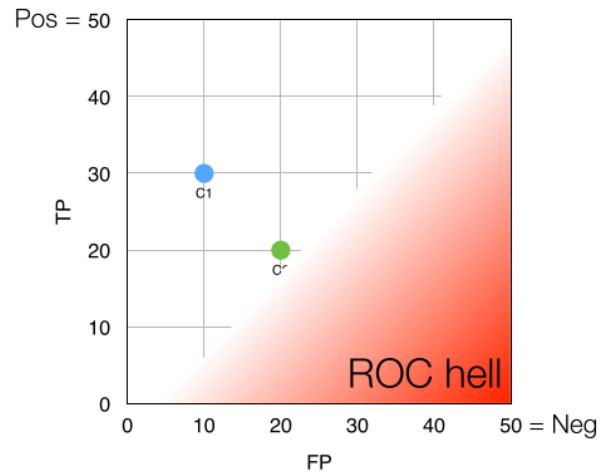
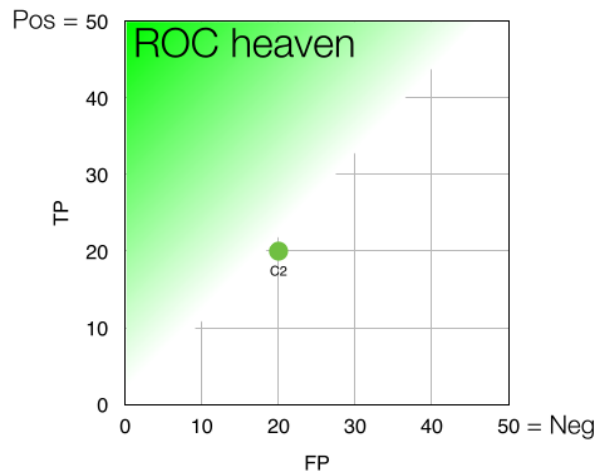
Grafico per visualizzare le informazioni della tavola di contingenza.

	Predicted \oplus	Predicted \ominus			Predicted \oplus	Predicted \ominus		
Actual \oplus	30	20	50		Actual \oplus	20	30	50
Actual \ominus	10	40	50		Actual \ominus	20	30	50
	40	60	100			40	60	100



Note:-

I classificatori che si trovano sulla bisettrice del piano cartesiano sono imprevedibili e quindi poco interessanti. Più un classificatore ha la coordinata x bassa e y alta più è preciso.



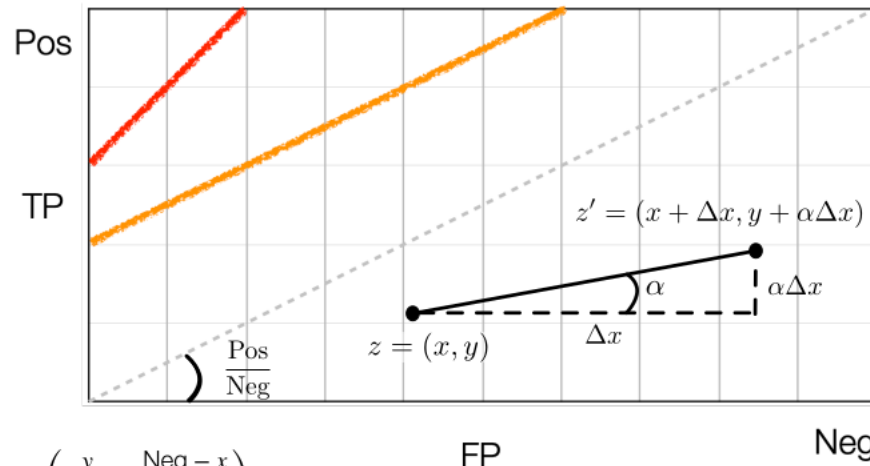
Note:-

Tutti i classificatori che stanno su una retta con pendenza 1 hanno la stessa *accuratezza*.

Definizione 1.2.7: Avg recall

$$\text{avg recall} = (\text{recall} + \text{specificity}) / 2 = (TP/POS + TN/NEG)/2$$

Se due classificatori hanno la stessa avg recall allora sono su linee parallele alla diagonale principale.



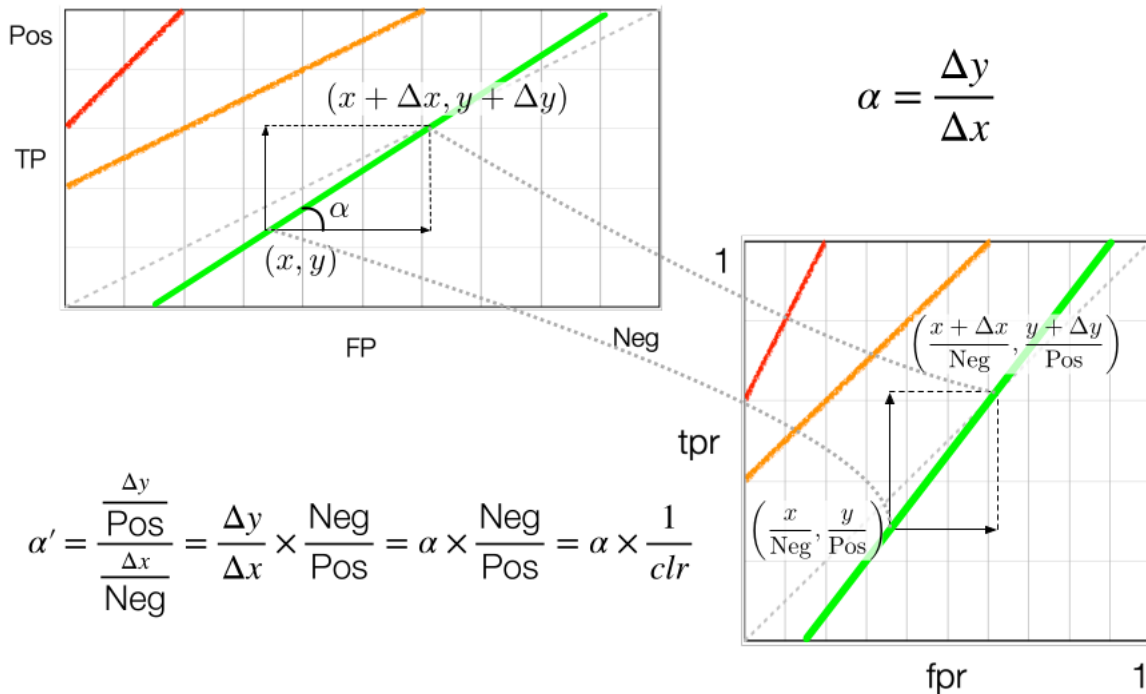
$$\text{avgrec}(z) = \left(\frac{y}{\text{Pos}} + \frac{\text{Neg} - x}{\text{Neg}} \right) / 2$$

$$\text{avgrec}(z') = \left(\frac{y + \alpha\Delta x}{\text{Pos}} + \frac{\text{Neg} - x - \Delta x}{\text{Neg}} \right) / 2 = \left(\frac{y}{\text{Pos}} + \alpha \frac{\Delta x}{\text{Pos}} + \frac{\text{Neg} - x}{\text{Neg}} - \frac{\Delta x}{\text{Neg}} \right) / 2$$

$$z \text{ and } z' \text{ have the same avg-rec if and only if } \alpha = \frac{\text{Pos}}{\text{Neg}}$$

Roc Plots Properties

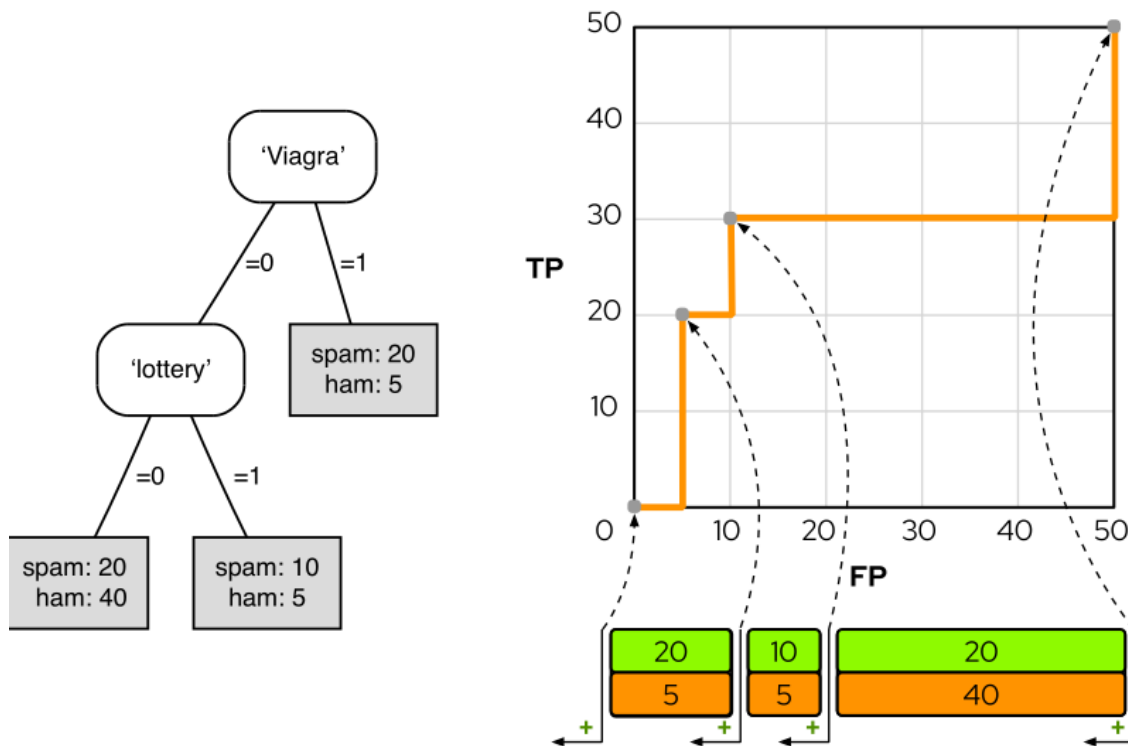
Se si vogliono confrontare le performance di un classificatore su un dataset o su un altro si deve *normalizzare* gli assi dividendo l'asse x per il numro di esempi negativi e l'asse y per il numero di esempi positivi. Così facendo si otterrà un quadrato con gli assi compresi tra 0 e 1.



Note:-

Il clr è il class ratio.

Più di un Classificatore per una Singola Feature.



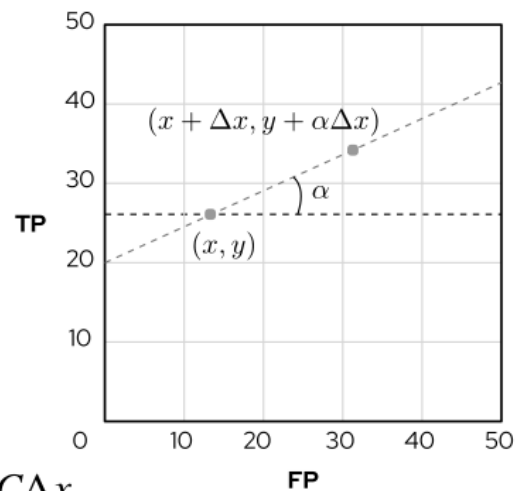
Domanda 1.2

Come si considera il caso in cui il costo per FP (falsi positivi) e FN (falsi negativi) sono differenti?

$$\text{acc}_C(x, y) = \frac{y + C \times \text{Neg} - Cx}{\text{Pos} + C \times \text{Neg}}$$

$$\begin{aligned} \text{acc}_C(x + \Delta x, y + \alpha \Delta x) &= \\ &= \frac{y + \alpha \Delta x + C \times \text{Neg} - C(x + \Delta x)}{\text{Pos} + C \times \text{Neg}} \\ &= \frac{y + C \times \text{Neg} - Cx}{\text{Pos} + C \times \text{Neg}} + \frac{\alpha \Delta x - C \Delta x}{\text{Pos} + C \times \text{Neg}} \end{aligned}$$

$$\text{if } \alpha = C, \text{ then: } \frac{\alpha \Delta x - C \Delta x}{\text{Pos} + C \times \text{Neg}} = \frac{C \Delta x - C \Delta x}{\text{Pos} + C \times \text{Neg}} = 0$$



In **ROC plots**, as is customary, one still needs to multiply by 1/clr.

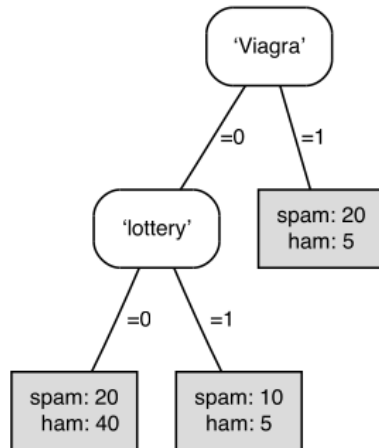
1.2.2 Scoring e ranking

Definizione 1.2.8: Scoring classifier

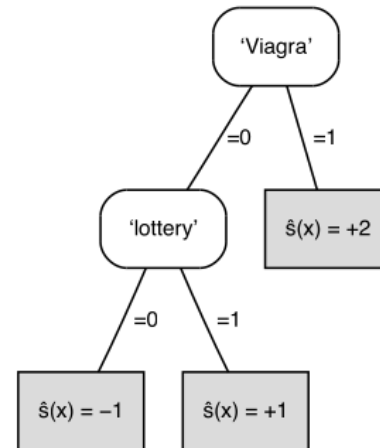
Uno scoring classifier è una mappatura $\hat{s}: \mathbf{X} \rightarrow \mathbb{R}^k$ il cui output è un vettore ($\hat{s}(x) = \hat{s}_1(x), \dots, \hat{s}_i(x)$) dove i -esimo componente è lo score assegnato alla classe C_i per l'istanza x .

Note:-

Se si hanno solo due classi si può considerare solo uno score. Gli score vanno interpretati nel contesto di un classificatore, sono misure della confidenza in una determinata predizione.



A **feature tree**
with training set
class distribution
on the leaves



A **scoring tree**
using the logarithm
of the class ratio
as scores

2

Test2

