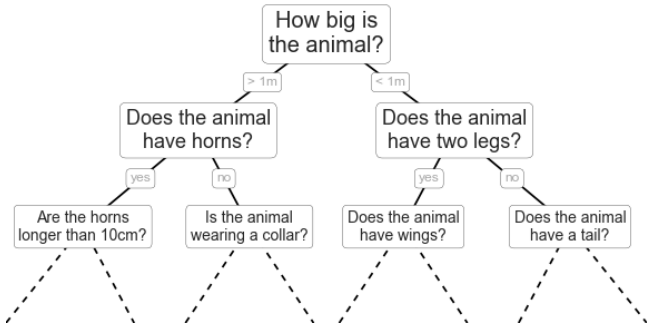
Decision Trees e Random Forests

Docente: Tommaso Muraca

Decision Trees e Random Forests

Gli **alberi decisionali sono modi estremamente intuitivi** per **classificare** o etichettare gli oggetti: basta porre una **serie di domande progettate per concentrarsi sulla classificazione**. Ad esempio, se volessi costruire un albero decisionale per **classificare un animale** che incontri durante un'escursione, potresti costruire quello mostrato nel grafico →

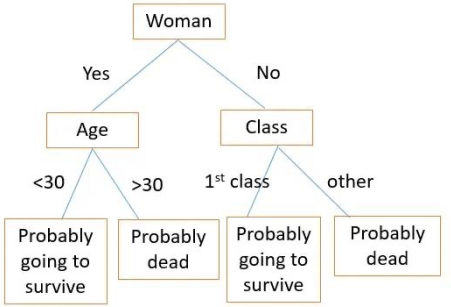


La **suddivisione binaria** lo rende **estremamente efficiente**: in un albero ben costruito, **ogni domanda ridurrà il numero di opzioni di circa la metà**, restringendo molto rapidamente le opzioni anche tra un gran numero di classi. Il **trucco**, ovviamente, sta nel **decidere quali domande porre ad ogni passaggio**. Nelle implementazioni di machine learning degli alberi decisionali, le domande generalmente assumono la forma di suddivisioni dei dati allineate agli assi: ovvero, ciascun nodo dell'albero divide i dati in due gruppi utilizzando un valore di interruzione all'interno di una delle funzionalità.

Esistono modi per **quantificare il guadagno di informazioni** in modo da poter **valutare essenzialmente ogni possibile suddivisione** dei dati di addestramento e massimizzare il guadagno di informazioni per ogni suddivisione. In questo modo possiamo **prevedere** ogni etichetta o valore nel **modo più efficiente possibile**.

Decision Trees e Random Forests

Su **Kaggle** è disponibile un **set di dati del Titanic** https://www.kaggle.com/c/titanic che viene **spesso utilizzato nelle lezioni introduttive di machine learning** e che abbiamo infatti già visto. Quando il Titanic affondò, morirono 1.502 dei 2.224 passeggeri e membri dell'equipaggio, anche se c'era un po' di casualità e confusione**, le donne, i bambini e le classi superiori avevano maggiori probabilità di sopravvivere**. Se guardiamo **all'albero decisionale a destra**, vedremo che riflette in qualche modo questa variabilità tra genere, età e classe.



Scelta delle suddivisioni in un albero decisionale

L'entropia è la **quantità di disordine in un insieme**, se i valori sono davvero contrastanti, c'è molta entropia; se puoi dividere in modo pulito i valori, non c'è entropia. Per ogni divisione in un nodo genitore, vogliamo che i nodi figli siano il più puri possibile: minimizzare l'entropia. Ad esempio, nel **Titanic**, il **genere** è un **grande fattore determinante** per la **sopravvivenza**, quindi ha senso che questa caratteristica venga utilizzata nella **prima divisione** poiché è quella che porta al maggior guadagno di informazioni.

Diamo un'occhiata alle nostre variabili Titanic -->

Data Dictionary

Variable	Definition
survival	Survival
pclass	Ticket class
sex	Sex
Age	Age in years
sibsp	# of siblings / spouses aboard the Titanic
parch	# of parents / children aboard the Titanic
ticket	Ticket number
fare	Passenger fare
cabin	Cabin number
embarked	Port of Embarkation

Key

0 = No, 1 = Yes

1 = 1st, 2 = 2nd, 3 = 3rd

C = Cherbourg, Q = Queenstown, S = Southampton

Decision Trees e Random Forests

Quindi possiamo costruire un albero selezionando una di queste variabili e suddividendo il set di dati in base a essa -->

La prima suddivisione separa il nostro set di dati in uomini e donne. Quindi, il ramo femminile viene nuovamente diviso in base all'età (la divisione che minimizza l'entropia). Allo stesso modo, il ramo maschile viene diviso per classe, **seguendo l'albero possiamo utilizzarlo per indovinare le probabilità di sopravvivenza del passeggero.**

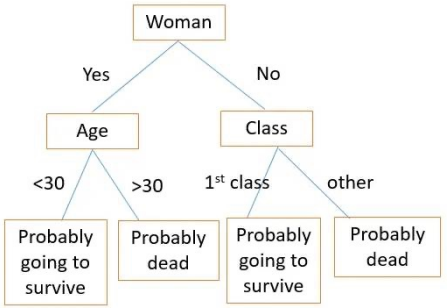
L'esempio del Titanic sta risolvendo quindi un **problema di classificazione ("sopravvivere" o "morire").**

Se utilizzassimo gli alberi decisionali per la **regressione** – ad esempio, per prevedere i **prezzi delle case** – creeremmo delle divisioni sulle **caratteristiche più importanti che determinano i prezzi delle case.**

Quanti metri quadrati: più o meno di ___? **Quante camere da letto e bagni**: più o meno di ___?

Quindi, durante il test, dovremmo seguire **tutte le suddivisioni** e prendere la **media di tutti i prezzi delle case nel nodo foglia finale** (nodo più in basso) dove la casa finisce come previsione per il prezzo di vendita.

Gli **alberi decisionali** sono efficaci perché **sono facili da leggere**, **potenti anche con dati disordinati** e **computazionalmente economici da implementare** una volta terminato l'addestramento. Gli alberi decisionali sono utili anche per **gestire dati misti** (numerici o categoriali).



Data Dictionary

Variable	Definition	K
survival	Survival	0
pclass	Ticket class	1
sex	Sex	
Age	Age in years	
sibsp	# of siblings / spouses aboard the Titanic	
parch	# of parents / children aboard the Titanic	
ticket	Ticket number	
fare	Passenger fare	
cabin	Cabin number	
embarked	Port of Embarkation	

Key

0 = No, 1 = Yes

1 = 1st, 2 = 2nd, 3 = 3rd

C = Cherbourg, Q = Queenstown, S = Southampton

Decision Trees e Random Forests

Detto questo, gli **alberi decisionali** sono però **computazionalmente costosi da addestrare**, comportano un **grosso rischio di overfitting** e tendono a trovare **valori ottimali locali perché non possono tornare indietro** dopo aver effettuato una divisione. **Per affrontare queste debolezze**, ci rivolgiamo a un metodo che illustra il potere di **combinare molti alberi decisionali** in un **unico modello:**

Random Forests: un insieme di alberi decisionali

Un modello composto da molti modelli è **chiamato modello d'insieme** e questa è solitamente una strategia vincente.

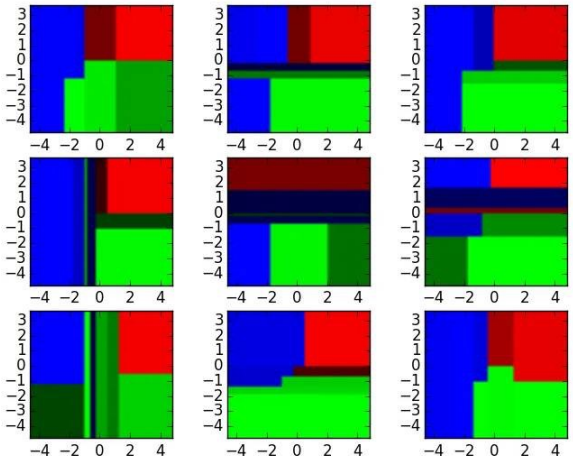
Un **singolo** albero decisionale può fare **molte scelte sbagliate** perché ha giudizi molto in bianco e nero. Una **foresta casuale** è un meta-stimatore che **aggrega molti alberi decisionali**, con alcune modifiche utili:

- Il **numero di funzionalità** che possono essere **suddivise** su **ciascun nodo è limitato** a una certa percentuale del totale, ciò garantisce che il modello d'insieme non faccia troppo affidamento su alcuna caratteristica individuale e faccia un uso corretto di tutte le caratteristiche potenzialmente predittive.

- **Ogni albero** estrae un **campione casuale dal set di dati** originale durante la generazione delle sue suddivisioni, aggiungendo un ulteriore elemento di casualità che impedisce l'adattamento eccessivo.

Queste modifiche impediscono inoltre che gli alberi siano troppo correlati. Senza i punti sopra, ogni albero sarebbe identico, poiché la suddivisione binaria ricorsiva è deterministica.

Nel grafico a destra vediamo con 9 alberi desicionali →



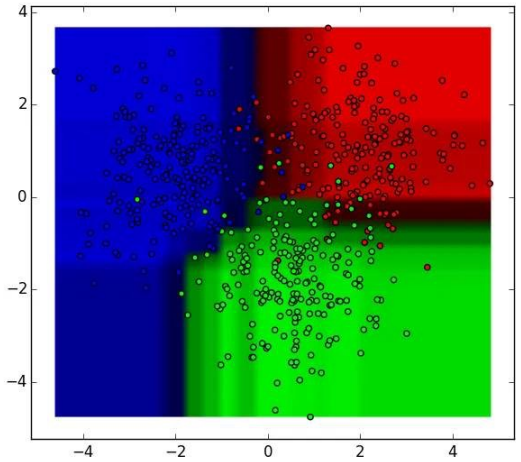
Decision Trees e Random Forests

Questi classificatori **dell'albero decisionale** possono essere **aggregati in un insieme di foreste casuali** che combina i loro input. Pensiamo agli **assi orizzontale e verticale di ciascun output** dell'albero decisionale come caratteristiche x1 e x2.

A determinati valori di ciascuna caratteristica, l'albero decisionale **restituisce una classificazione di "blu", "verde", "rosso", ecc.**

Questi **risultati vengono aggregati**, attraverso **voti modali o media**, in un unico modello d'insieme che finisce per sovraperformare l'output di qualsiasi singolo albero decisionale.

Le **foreste casuali** rappresentano un **ottimo punto di partenza per il processo di modellazione**, poiché tendono ad avere **prestazioni elevate con un'elevata tolleranza** per i dati meno puliti e possono essere **utili per capire** quali **caratteristiche** effettivamente contano tra molte caratteristiche.



Andiamo ora a vedere un po' di pratica: