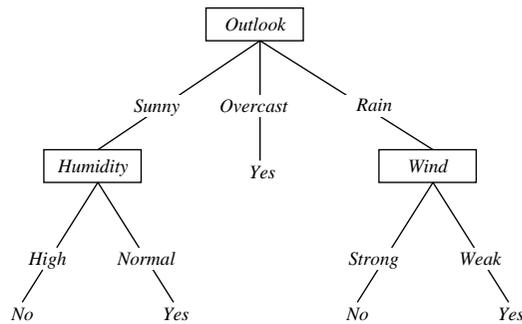


Alberi di Decisione

Decidere se è la giornata ideale per giocare a Tennis !!



Es. ingresso: (Outlook=Sunny, Temperature=Hot, Humidity=High, Wind=Strong).
attributo valore attributo

In un Albero di decisione:

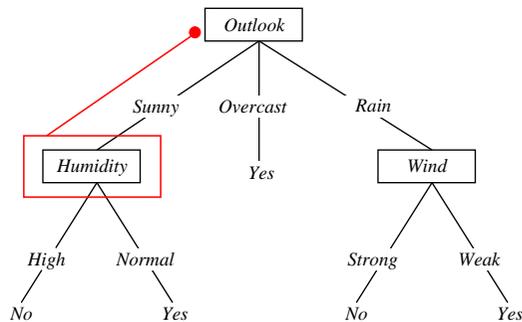
- Ogni nodo interno effettua un test su un attributo;
- Ogni ramo uscente da un nodo corrisponde ad uno dei possibili valori che l'attributo può assumere;
- Ogni foglia assegna una classificazione.

Per classificare una istanza con un Albero di Decisione bisogna

1. partire dalla radice;
2. selezionare l'attributo della istanza associato al nodo corrente;
3. seguire il ramo associato al valore assegnato a tale attributo nella istanza;
4. se si raggiunge una foglia restituire l'etichetta associata alla foglia, altrimenti a partire dal nodo corrente ripetere dal passo 2.

Alberi di Decisione: Classificazione

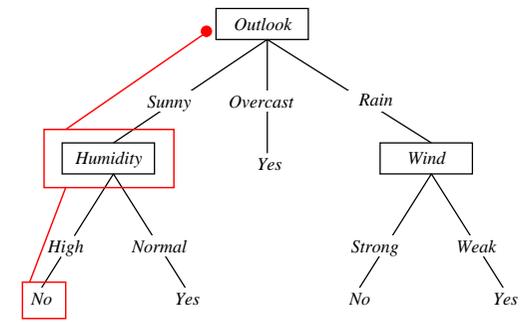
Alla radice è associato Outlook e quindi bisogna seguire il ramo Sunny



Es. ingresso: (Outlook=Sunny, Temperature=Hot, Humidity=High, Wind=Strong).

Alberi di Decisione: Classificazione

Al nodo raggiunto è associato Humidity e quindi bisogna seguire il ramo High



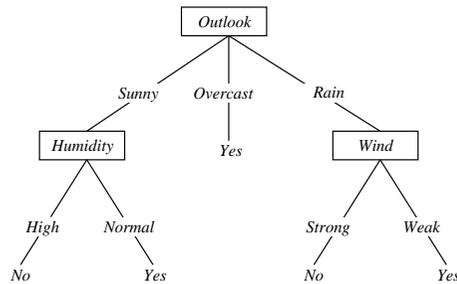
Es. ingresso: (Outlook=Sunny, Temperature=Hot, Humidity=High, Wind=Strong).

Si raggiunge una foglia: quindi l'istanza è classificata No

Alberi di Decisione e Funzioni Booleane

Con Alberi di Decisione ogni funzione booleana può essere rappresentata:

- Ogni cammino dalla radice ad una foglia codifica una **congiunzione** di test su attributi;
- Più cammini che conducono allo stesso tipo di classificazione codificano una **disgiunzione** di congiunzioni;



(Outlook=Sunny **and** Humidity=Normal)
or
 Outlook=Overcast
or
 (Outlook=Rain **and** Wind=Weak)

L'apprendimento di Alberi di Decisione tipicamente procede attraverso una procedura di tipo **(divide et impera)** che costruisce l'albero top-down:

1. crea il nodo radice, $\hat{T}r \leftarrow Tr$ e inserisci tutti gli attributi nell'insieme \mathcal{A} ;
2. **se** gli esempi in $\hat{T}r$ sono tutti della stessa classe (- o +), assegna al nodo l'etichetta della classe e fermati;
 altrimenti
 (a) **se** \mathcal{A} è vuoto, assegna al nodo l'etichetta della classe più frequente e fermati;
 altrimenti assegna al nodo $A \leftarrow$ l'attributo "ottimo" in \mathcal{A} ;
3. partiziona $\hat{T}r$ secondo i possibili valori che A può assumere:
 $\hat{T}r_{A=val_1}, \dots, \hat{T}r_{A=val_{m(A)}}$, dove $m(A)$ = numero valori distinti di A ;
4. $\forall \hat{T}r_{A=val_j} = \emptyset$ crea una foglia figlio con l'etichetta della classe più frequente in $\hat{T}r$;
5. $\forall \hat{T}r_{A=val_i} \neq \emptyset$ crea nodo figlio e vai a 2 con $\hat{T}r \leftarrow \hat{T}r_{A=val_i}$ e $\mathcal{A} \leftarrow \mathcal{A} \setminus A$.

Esempio di Algoritmo di Apprendimento: ID3

L'apprendimento di Alberi di Decisione tipicamente procede attraverso una procedura di tipo **(divide et impera)** che costruisce l'albero top-down:

1. crea il nodo radice, $\hat{T}r \leftarrow Tr$ e inserisci tutti gli attributi nell'insieme \mathcal{A} ;
2. **se** gli esempi in $\hat{T}r$ sono tutti della stessa classe (- o +), assegna al nodo l'etichetta della classe e fermati;
 altrimenti
 (a) **se** \mathcal{A} è vuoto, assegna al nodo l'etichetta della classe più frequente e fermati;
 altrimenti assegna al nodo $A \leftarrow$ **l'attributo "ottimo" in \mathcal{A}** ;
3. partiziona $\hat{T}r$ secondo i possibili valori che A può assumere:
 $\hat{T}r_{A=val_1}, \dots, \hat{T}r_{A=val_{m(A)}}$, dove $m(A)$ = numero valori distinti di A ;
4. $\forall \hat{T}r_{A=val_j} = \emptyset$ crea una foglia figlio con l'etichetta della classe più frequente in $\hat{T}r$;
5. $\forall \hat{T}r_{A=val_i} \neq \emptyset$ crea nodo figlio e vai a 2 con $\hat{T}r \leftarrow \hat{T}r_{A=val_i}$ e $\mathcal{A} \leftarrow \mathcal{A} \setminus A$.

Esempio di Algoritmo di Apprendimento: ID3

L'apprendimento di Alberi di Decisione tipicamente procede attraverso una procedura di tipo **(divide et impera)** che costruisce l'albero top-down:

1. crea il nodo radice, $\hat{T}r \leftarrow Tr$ e inserisci tutti gli attributi nell'insieme \mathcal{A} ;
2. **se** gli esempi in $\hat{T}r$ sono tutti della stessa classe (- o +), assegna al nodo l'etichetta della classe e fermati;
 altrimenti
 (a) **se** \mathcal{A} è vuoto, assegna al nodo l'etichetta della classe più frequente e fermati;
 altrimenti assegna al nodo $A \leftarrow$ **l'attributo "ottimo" in \mathcal{A}** ;
3. partiziona $\hat{T}r$ secondo i possibili valori che A può assumere:
 $\hat{T}r_{A=val_1}, \dots, \hat{T}r_{A=val_{m(A)}}$, dove $m(A)$ = numero valori distinti di A ;
4. $\forall \hat{T}r_{A=val_j} = \emptyset$ crea una foglia figlio con l'etichetta della classe più frequente **in $\hat{T}r$** ;
5. $\forall \hat{T}r_{A=val_i} \neq \emptyset$ crea nodo figlio e vai a 2 con.. **se necessario risalire ai nodi avi** \uparrow