QUADERNO #2 *Data mining*

Carlo MIGLIACCIO matricola: 272489

1. Decision tree (`minimal gain: 0.01, maximal depth: 10`)

Figura 1 - Albero generato con i parametri sopra (le foglie sono le etichette di classe)

a) L'attributo considerato dall'algoritmo il più selettivo ai fini dell'analisi è **`node-caps`** che è un booleano associato al fatto che il tumore resti arginato alla *capsula del linfonodo* (come si può leggere dalla descrizione in http://archive.ics.uci.edu/ml/datasets/Breast+Cancer: *"Node caps: if the cancer does metastasise to a lymph node, although outside the original site of the tumor it may remain "contained" by the capsule of the lymph node. However, over time, and with more aggressive disease, the tumor may replace the lymph node and then penetrate the capsule, allowing it to invade the surrounding tissues")*

b) Come si può vedere dall'immagine seguente l'albero riportato al passo precedente ha **altezza** pari a 7.

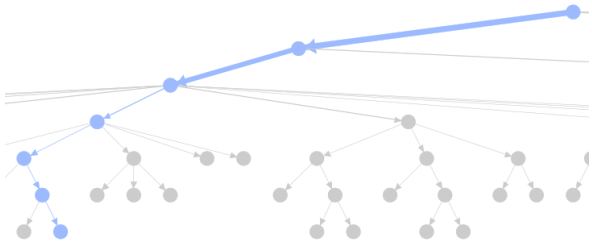


Figura 2 - Esempio di cammino radice-foglia

c) Di seguito si riporta un esempio di partizionamento **puro.** La purezza è associata al fatto che c'è una distribuzione netta delle etichette di classe. Esempio: (N_A: 0, N_B: 15), oppure (N_A: 15, N_B: 0) dove A e B sono le classi e N_A e N_B sono il numero di record associati alle rispettive classi.

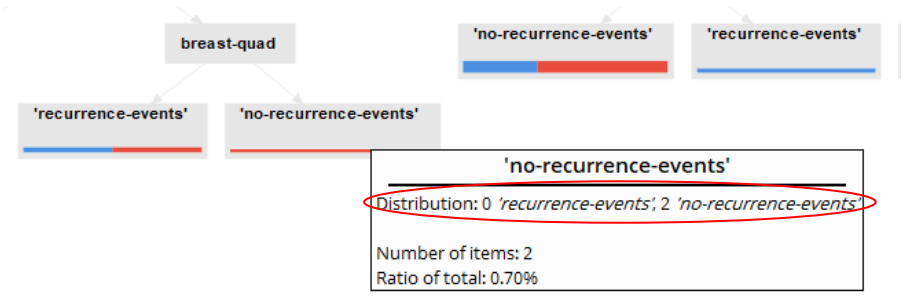


Figura 3 - Esempio di partizione pura

Essendo il dataset di dimensione ridotta (circa 300 record) e avendo fatto pre-pruning dell'albero, impostando un minimal gain non si riescono ad osservare casi in cui tutte le foglie arrivano da partizioni pure.

2. Decision tree: impatto del `minimal gain` e del `maximal depth`

Si riportano di seguito alcune immagini che mostrano come i due parametri di minimal gain e maximal depth influiscono sulle caratteristiche dell'albero generato (numero di nodi, altezza...).

`minimal gain: 0.01` `maximal depth: 6`

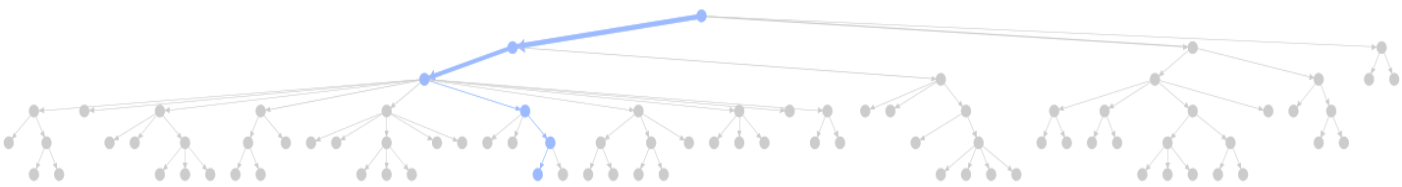


Figura 4 - Albero generato (mg=0.01, md=6)

minimal gain: 0.03 **maximal depth: 10**

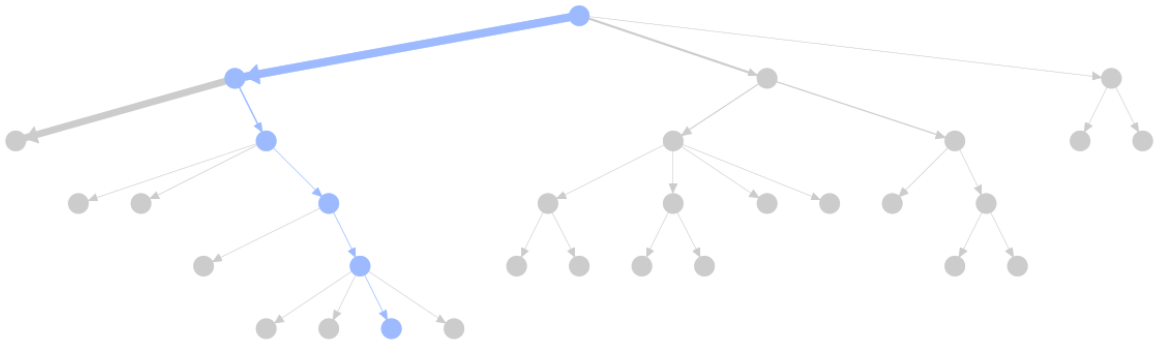


Figura 5 - Albero generato con mg=0.03, md=10)

minimal gain: 0.02 **maximal depth: 5**

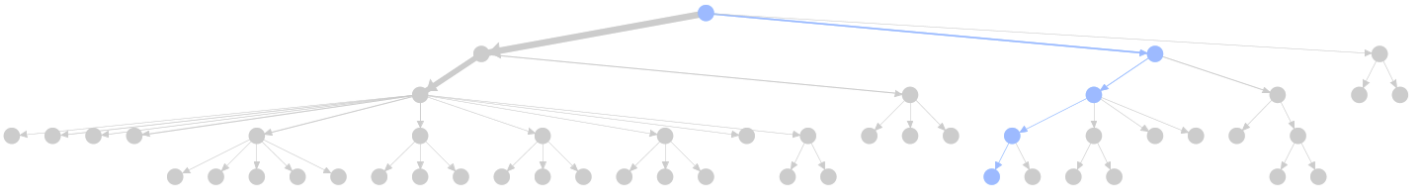


Figura 6 - Albero generato con mg=0.02, md=5

minimal gain: 0.03 **maximal depth: 5**

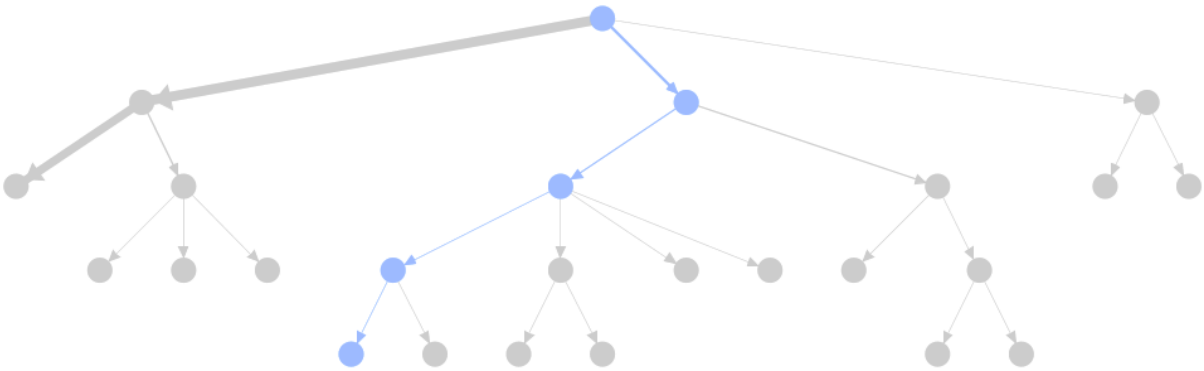


Figura 7 - Albero generato con mg=0.03, md=5

minimal gain: 0.03 **maximal depth: 4**

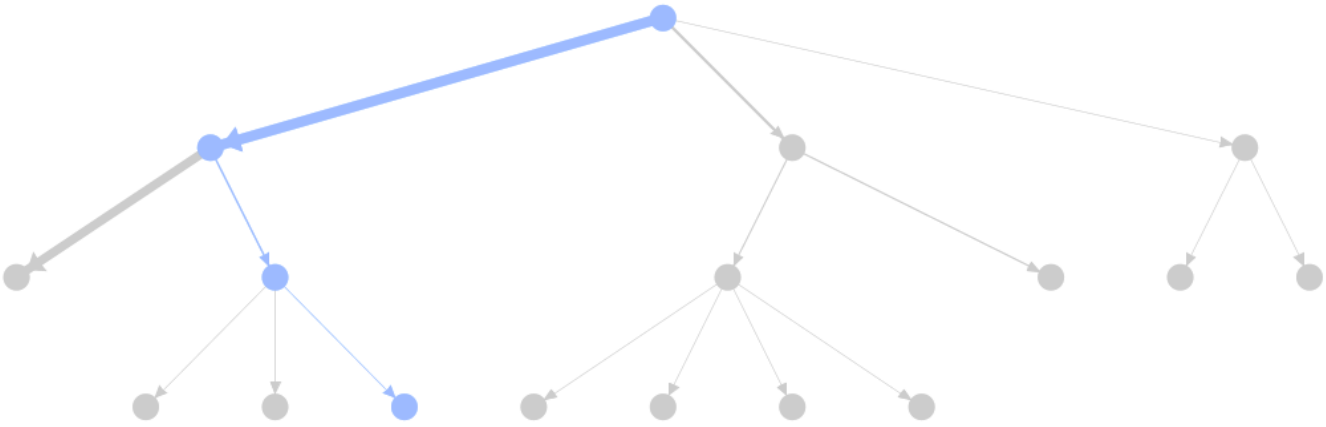


Figura 8 - Albero generato con mg=0.03, md=4

Nelle immagini riportate prima per praticità e chiarezza di visualizzazioe si sono state nascoste le etichette agli archi e ai nodi. Un esempio di albero completo di queste informazioni in precedenza omesse è riportato di seguito:

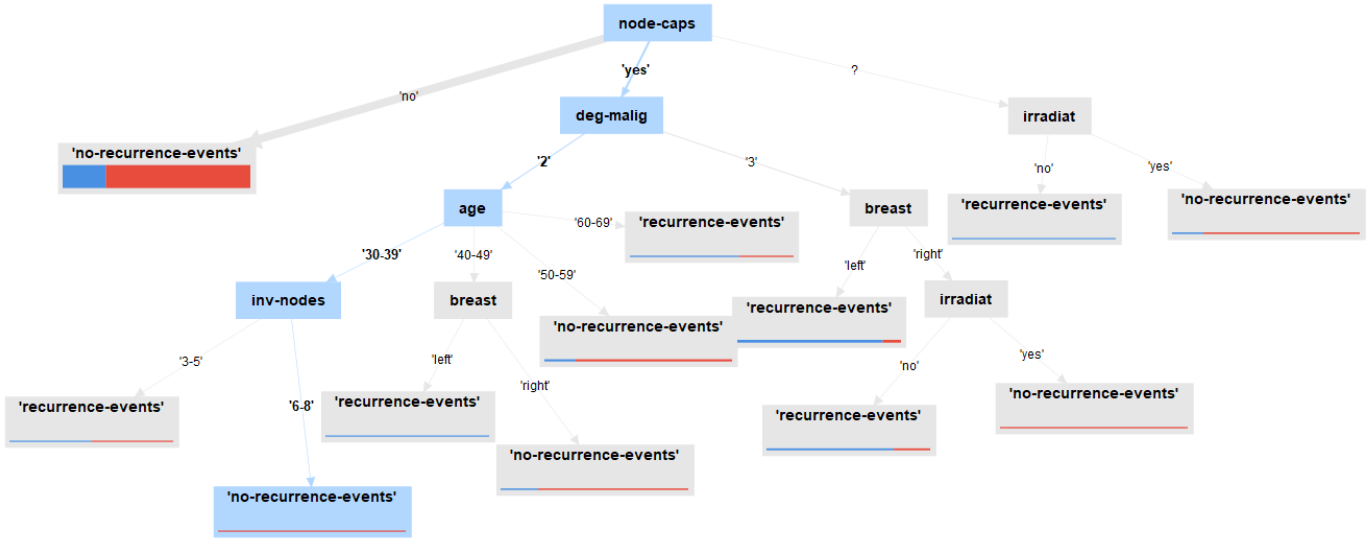


Figura 9 - Albero generato con mg=0.04, md=5 (completo di tutte le informazioni)

Ad ogni passo di un cammino (vedi nodi in azzurro) che porta dalla radice (**`node-caps`**) fino alla foglia (classe **`no-recurrence-events`**) si fa una scelta sul valore di un certo attributo.

3. **Decision tree:** validazione del modello tramite *10-fold Stratified Cross-Validation*

La tecnica di validazione ***k-fold Stratified Cross-Validation*** si applica soprattutto quando si hanno a disposizione dataset di dimensione ridotta. I record disponibili vengono divisi in **k insieme disgiunti** attraverso un *campionamento stratificato*[1], successivamente, k-1 insiemi vengono usati per addestrare il modello (training set) mentre la partizione rimanente viene utilizzata per la verifica del modello stesso (test set). Nel caso analizzato k=10.

Il tuning di entrambi i parametri, `minimal gain` e `maximal depth`, che danno vincoli sulla crescita dell'albero, influenzano le metriche di performance ricavabili dalle matrici di confusione. In particolare un modello troppo specifico, con minimal gain basso e profondità elevata tende ad essere affetto dal problema dell'overfitting, si ha quindi in generale una carenza sulle performance di generalizzazione del modello risultante.

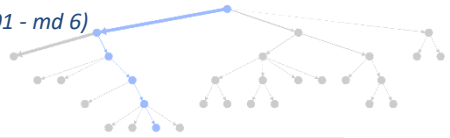
`minimal gain:` `0.01` **`maximal depth:`** `6`

accuracy: 68.55% +/- 8.74% (micro average: 68.53%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	35	40	46.67%
pred. 'no-recurrence-events'	50	161	76.30%
class recall	41.18%	80.10%	

Tabella 1 - Matrice di confusione (parametri albero mg 0.01 - md 6)

`minimal gain:` `0.03` **`maximal depth:`** `10`



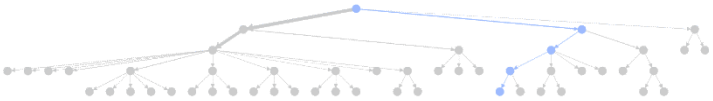
accuracy: 70.31% +/- 5.87% (micro average: 70.28%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	24	24	50.00%
pred. 'no-recurrence-events'	61	177	74.37%
class recall	28.24%	88.06%	

Tabella 2 - Matrice di confusione (parametri albero mg 0.03 - md 10)

[1] Con questa tecnica di campionamento si assicura che ogni campione abbia le etichette di classe distribuite in modo equo

minimal gain: 0.02 **maximal depth: 5**

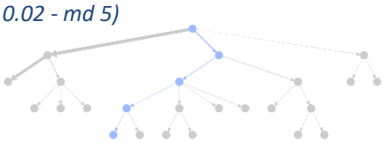


accuracy: 71.35% +/- 8.78% (micro average: 71.33%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	35	32	52.24%
pred. 'no-recurrence-events'	50	169	77.17%
class recall	41.18%	84.08%	

Tabella 3 - Matrice di confusione (parametri dell'albero mg 0.02 - md 5)

minimal gain: 0.03 **maximal depth: 5**

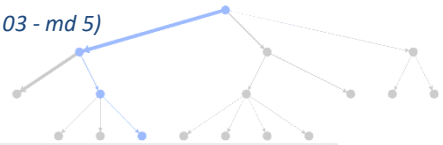


accuracy: 72.06% +/- 6.03% (micro average: 72.03%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	27	22	55.10%
pred. 'no-recurrence-events'	58	179	75.53%
class recall	31.76%	89.05%	

Tabella 4 - Matrice di confusione (parametri dell'albero mg 0.03 - md 5)

minimal gain: 0.03 **maximal depth: 4**



accuracy: 70.65% +/- 6.72% (micro average: 70.63%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	27	26	50.94%
pred. 'no-recurrence-events'	58	175	75.11%
class recall	31.76%	87.06%	

Tabella 5 - Matrice di confusione (parametri dell'albero mg 0.03 - md 4)

4. K-Nearest Neighbour (KNN) e Naïve Bayes: validazione e confronto

a) *KNN: 10-fold Stratified Cross validation, tuning del parametro K*

Come mostrano le matrici di confusione ottenute, facendo variare il parametro K da 2 a 10 l'accuratezza media del modello cresce, da 11 in poi comincia a diminuire in quanto probabilmente sono inclusi punti nell'intorno con etichetta di classe diversa da quella effettiva, che inficiano quindi la bontà della classificazione.

K=2

```
Weighted 2-Nearest Neighbour model for classification.
The model contains 286 examples with 9 dimensions of the following classes:
  'recurrence-events'
  'no-recurrence-events'
```

accuracy: 65.73% +/- 8.62% (micro average: 65.73%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	39	52	42.86%
pred. 'no-recurrence-events'	46	149	76.41%
class recall	45.88%	74.13%	

K=6

```
Weighted 6-Nearest Neighbour model for classification.
The model contains 286 examples with 9 dimensions of the following classes:
  'recurrence-events'
  'no-recurrence-events'
```

accuracy: 72.03% +/- 6.10% (micro average: 72.03%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	24	19	55.81%
pred. 'no-recurrence-events'	61	182	74.90%
class recall	28.24%	90.55%	

K=8

```
Weighted 8-Nearest Neighbour model for classification.
The model contains 286 examples with 9 dimensions of the following classes:
  'recurrence-events'
  'no-recurrence-events'
```

accuracy: 74.51% +/- 5.02% (micro average: 74.48%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	24	12	66.67%
pred. 'no-recurrence-events'	61	189	75.60%
class recall	28.24%	94.03%	

K=10

```
Weighted 10-Nearest Neighbour model for classification.
The model contains 286 examples with 9 dimensions of the following classes:
  'recurrence-events'
  'no-recurrence-events'
```

accuracy: 75.20% +/- 5.43% (micro average: 75.17%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	25	11	69.44%
pred. 'no-recurrence-events'	60	190	76.00%
class recall	29.41%	94.53%	

K=13

```
Weighted 13-Nearest Neighbour model for classification.
The model contains 286 examples with 9 dimensions of the following classes:
  'recurrence-events'
  'no-recurrence-events'
```

accuracy: 73.79% +/- 6.89% (micro average: 73.78%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	19	9	67.86%
pred. 'no-recurrence-events'	66	192	74.42%
class recall	22.35%	95.52%	

b) **Naïve Bayes*: 10-fold Stratified Cross validation, confronto con KNN***

accuracy: 72.45% +/- 7.70% (micro average: 72.38%)

	true 'recurrence-events'	true 'no-recurrence-events'	class precision
pred. 'recurrence-events'	41	35	53.95%
pred. 'no-recurrence-events'	44	166	79.05%
class recall	48.24%	82.59%	

Il classificatore *Naïve Bayes* ottiene prestazioni inferiori al KNN (per certi parametri di K), ma migliori rispetto alle prestazioni ottenute con alberi decisionali con i parametri scelti. Si ricorda che questo tipo di classificatore ha alla base il Teorema di Bayes e un'ipotesi (molto forte) di indipendenza delle varie features del dataset. Al passo seguente si valuta quanto corretta può essere questa ipotesi.

5. Matrice di correlazione

Quella riportata a fianco è la **matrice di correlazione** ricavata dal dataset analizzato. L'ipotesi di indipendenza tra gli attributi, utilizzata nel Naive Bayes, non risulta valida soprattutto per alcune coppie di feature che mostrano una tendenza di correlazione, in particolare correlazione negativa.

Attributes	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
age	1	0.241	-0.045	-0.001	0.052	-0.043	0.067	-0.024	-0.011
menopause	0.241	1	0.019	-0.011	0.130	-0.161	0.077	-0.096	-0.075
tumor-size	-0.045	0.019	1	-0.131	0.058	0.133	-0.022	-0.056	-0.022
inv-nodes	-0.001	-0.011	-0.131	1	-0.465	-0.213	0.040	0.063	0.399
node-caps	0.052	0.130	0.058	-0.465	1	0.098	0.024	-0.036	-0.197
deg-malig	-0.043	-0.161	0.133	-0.213	0.098	1	-0.073	0.018	-0.074
breast	0.067	0.077	-0.022	0.040	0.024	-0.073	1	0.175	-0.019
breast-quad	-0.024	-0.096	-0.056	0.063	-0.036	0.018	0.175	1	-0.005
irradiat	-0.011	-0.075	-0.022	0.399	-0.197	-0.074	-0.019	-0.005	1

La coppia di attributi **maggiormente (negativamente) correlata** è `(node-caps, inv-nodes)`.

Nella matrice sono stati inoltre riquadrati altri valori di correlazione significativi.