Arborii de decizie (Decision trees)

Ce ne așteaptă?

1. Impuritatea Gini
2. Divizarea caracteristicilor categoriale binare
3. Divizarea caracteristicilor numerice
4. Divizarea caracteristicilor categoriale multiclass
5. Crearea arborelui de decizii
6. Instrumente Scikit-Learn

1. Impuritatea Gini

- **Impuritatea Gini un instrument matematic ce permite aprecierea gradului de uniformitatea a datelor unui grup**

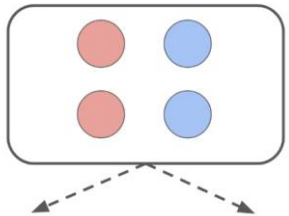
$$G(Q) = \sum_{c} p_c(1 - p_c)$$

unde: c – o clasa a datelor, p_c - probabilitatea clasei c

- **Impuritatea Gini se utilizează pentru determinarea caracteristicilor ce asigură ce mai bună divizare a datelor în procesul de elaborare a arborelui de decizie**

- **Impuritatea Gini a nodurilor finale trebuie să fie minim posibilă ceea ce ar însemna o divizare efectivă a claselor**

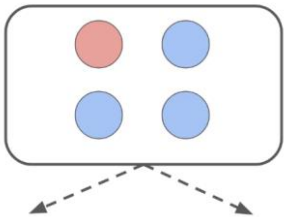
Determinarea impurității Gini pentru un grup de date



$$G(rosu) = p_{rosu}(1 - p_{rosu}) = \frac{2}{4}\left(1 - \frac{2}{4}\right) = 0{,}25$$

$$G(albastru) = p_{albastru}(1 - p_{albastru}) = \frac{2}{4}\left(1 - \frac{2}{4}\right) = 0{,}25$$

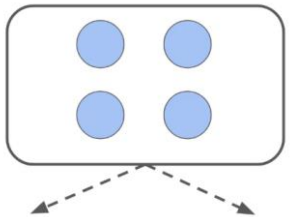
$$G(grup) = G(rosu) + G(albastru) = 0{,}5$$



$$G(rosu) = p_{rosu}(1 - p_{rosu}) = \frac{1}{4}\left(1 - \frac{1}{4}\right) = 0{,}1875$$

$$G(albastru) = p_{albastru}(1 - p_{albastru}) = \frac{3}{4}\left(1 - \frac{3}{4}\right) = 0{,}1875$$

$$G(grup) = 0{,}3755$$



$$G(rosu) = p_{rosu}(1 - p_{rosu}) = \frac{0}{4}\left(1 - \frac{0}{4}\right) = 0$$

$$G(albastru) = p_{albastru}(1 - p_{albastru}) = \frac{4}{4}\left(1 - \frac{4}{4}\right) = 0$$

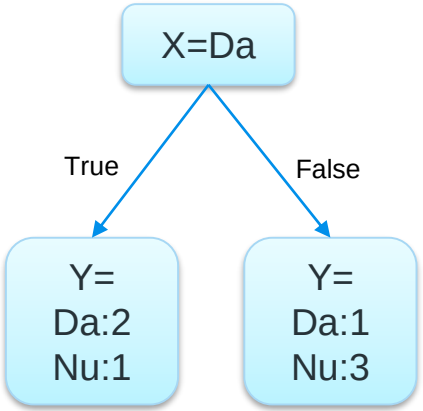
$$G(grup) = 0$$

2. Divizarea caracteristicilor categoriale binare

- Primul pas în algoritmul Decision Trees este determinarea caracteristicii nod rădăcină
- Pentru determinarea nodului rădăcină se determina caracteristica cu cea mai mică impuritate Gini
- Pentru determinarea impurității Ginii se va considera cazurile:
  - Caracteristici categoriale binare
  - Caracteristici numerice
  - Caracteristici categoriale cu clase multiple

Impuritatea Gini pentru caracteristicile categoriale binare

X	y
Da	Da
Da	Da
Nu	Nu
Nu	Nu
Nu	Da
Nu	Nu
Da	Nu



Impuritatea Gini pentru nodul final din stânga (când X=Da)

$$G(stânga) = p_{y=Da}(1 - p_{y=Da}) + p_{y=Nu}(1 - p_{y=Nu}) = \\ = \frac{2}{3}\left(1 - \frac{2}{3}\right) + \frac{1}{3}\left(1 - \frac{1}{3}\right) = 0{,}44$$

Impuritatea Gini pentru nodul final din dreapta(când X≠Da)

$$G(dreapta) = p_{y=Da}(1 - p_{y=Da}) + p_{y=Nu}(1 - p_{y=Nu}) = \\ = \frac{1}{4}\left(1 - \frac{1}{4}\right) + \frac{3}{4}\left(1 - \frac{3}{4}\right) = 0{,}375$$

Impuritatea Gini pentru caracteristica X – media ponderată a impurității nodurilor rezultate din această caracteristică

$$G(X) = p_{x=Da} * G(stânga) + p_{x=Nu} * G(dreapta) = \frac{3}{7} * 0{,}44 + \frac{4}{7} * 0{,}375 = 0{,}403$$

3. Divizarea caracteristicilor numerice

- **Selectarea criteriului de divizare**

- **Se reorganizează datele în ordinea crescătoarea a valorilor caracteristicii X**

- **Se determina valorile medii dintre datele vecine ale caracteristicii X**

- **Se utilizează valorile medii (N) în calitatea de criterii de divizarea**

X	y
10	Da
40	Nu
20	Da
50	Nu
30	Nu

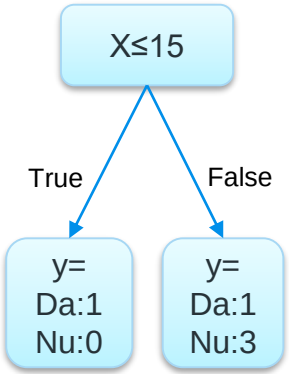
X	y
10	Da
20	Da
30	Nu
40	Nu
50	Nu

X	y
10	Da
15	
20	Da
25	
30	Nu
35	
40	Nu
45	
50	Nu

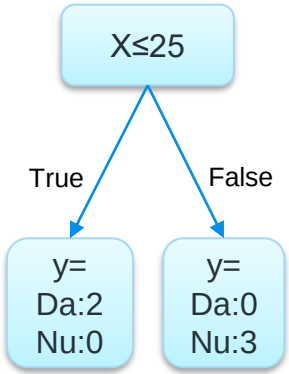
X≤N

- **Determinarea criteriului de divizare cu cea mai mică valoarea a impurității Gini**

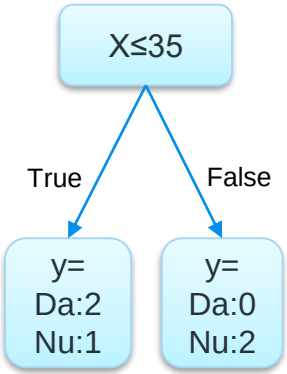
X	y
10	Da
15	
20	Da
25	
30	Nu
35	
40	Nu
45	
50	Nu



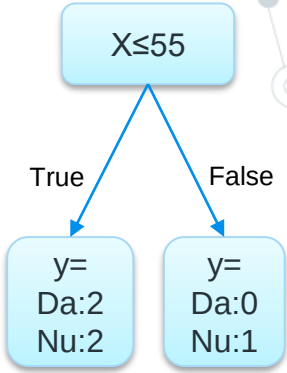
$G(X) = 0{,}3$



$G(X) = 0$



$G(X) = 0{,}26$



$G(X) = 0{,}4$

- **Se selectează criteriul de divizare cu cea mai mică valoarea a impurității Gini, iar această valoarea a impurității se va considera impuritatea Gini a caracteristicii X**

4. Divizarea caracteristicilor categoriale multiclass

- **Selectarea criteriilor de divizare**
  - **Criterii de divizare se vor considera valorile claselor dar și combinații dintre acestea**

X	y
A	Da
B	Da
C	Nu
A	Nu
B	Nu

X==A

X==B

X==C

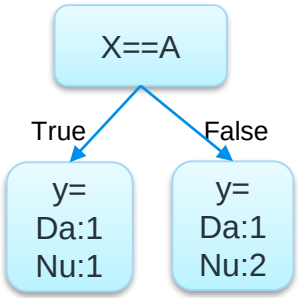
X==A or B

X==A or C

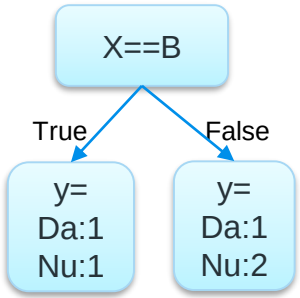
X==B or C

- **Determinarea criteriului de divizare cu cea mai mică valoarea a impurității Gini**

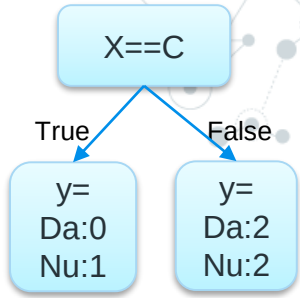
X	y
A	Da
B	Da
C	Nu
A	Nu
B	Nu



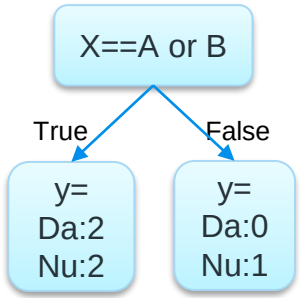
$G(X) = 0{,}46$



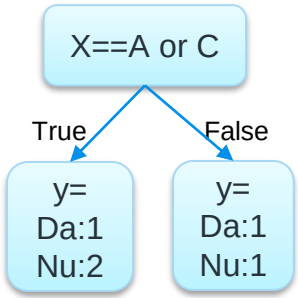
$G(X) = 0{,}46$



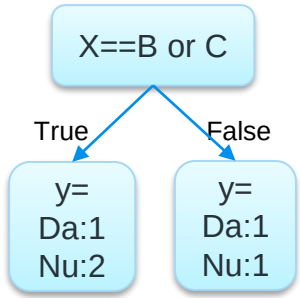
$G(X) = 0{,}4$



$G(X) = 0{,}4$



$G(X) = 0{,}46$



$G(X) = 0{,}46$

- **Se selectează criteriul de divizare cu cea mai mică valoarea a impurității Gini, iar această valoarea a impurității se va considera impuritatea Gini a caracteristicii X**

5. Crearea arborelui de decizii

- **Etapele de crearea a arborelui**

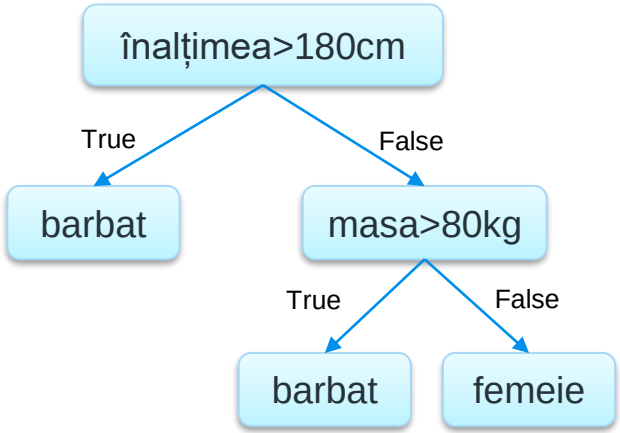
1. **Determinarea impurității Gini a tuturor caracteristicilor**
2. **Selectarea caracteristicii cu cea mai mică valoarea a impurității Gini drept nodul rădăcină**
3. **Pe baze criteriului selectat a caracteristicii nodului rădăcină se divizează datele și se crearea nodurile copii ai nodului rădăcină**
4. **Se repetă etapele 1 – 3 pentru fiecare dintre nodurile create**
5. **Procesul de divizarea se va opri atunci când:**
   - **Se va obține valoarea nulă a impurității Gini**
   - **Se va atinge valoarea numărul de divizări dacă se fixează valoarea acestuia**
   - **Se va atinge valoarea limită a impurității gini dacă se fixează această valoarea**
   - **Se atinge numărul maxim de noduri terminale dacă se fixează numărul acestora**

Clasificare a datelor noi

- **În cazul datelor noi se va parcurge arborele de decizie începând cu nodul rădăcină spre nodul terminal corespunzător clasei**

- **Exemplu:**

1. *O persoana cu înălțimea 183 cm și masa 77 kg va fi clasificată ca bărbat*
2. *O persoana cu înălțimea 177 cm și masa 73 kg va fi clasificată ca femeie*
3. *O persoana cu înălțimea 177 cm și masa 83 kg va fi clasificată ca bărbat*



6. Instrumente Scikit-Learn

- Se importa clasa algoritmului DecisionTreeClassifier

```
from sklearn.tree import DecisionTreeClassifier
```

- Se creează un model cu fixarea valorilor hiper-parametrilor:
  - criterion – criteriul de determinarea a uniformității grupului de date (valori posibile = {'gini', 'entropy'}, implicit ='gini')
  - max_depth - numărul maxim de divizări la crearea arborelui (valori posibile = int, implicit =None)
  - max_leaf_node – numărul maxim al nodurilor terminale ale arborelui (valori posibile =int, implicit=None)
  - min_impurity_decrease – valoarea minima de descreștere a impurității datelor la divizarea datelor (valori posibile = float, implicit=0,0)

```
model = DecisionTreeClassifier()
```

- Se realizează trainingul modelului pe datele de training

```
model.fit(X_train, y_train)
```

- Se vizualizează grafic arborele

```
from sklearn.tree import plot_tree
plt.figure(figsize=(12,8),dpi=250)
plot_tree(model,filled=True,feature_names=X.columns);
```

- Se realizează predicția pe datele de test

```
y_pred = model.predict(X_test)
```

- Se realizează predicția pe datele de test cu afișarea probabilităților

```
y_pred = model.predict_proba(X_test)
```