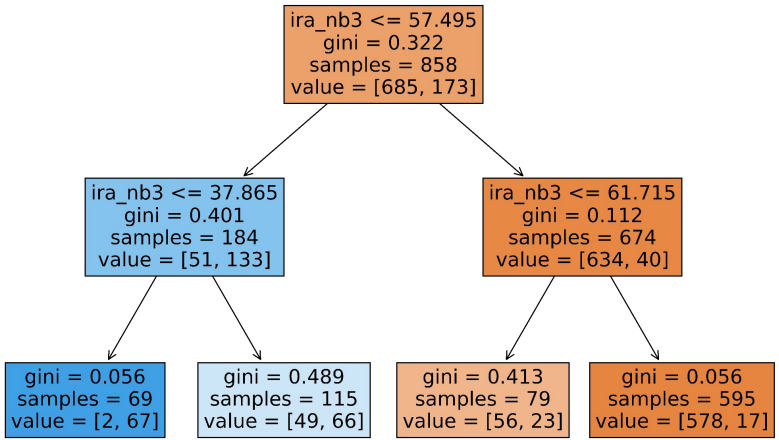
Floresta de árvores aleatórias no contexto da estrutura de dados e algoritmos

Ewerton L. Costadelle
Rafael N. de Carvalho

Antes da floresta havia a árvore



Algoritmos

- AID (Automatic Interaction Detection)
- THAID (THeta Automatic Interaction Detection)
- CART (Classification And Regression Trees)
- ID3 (Iterative Dichotomiser 3)
- C4.5

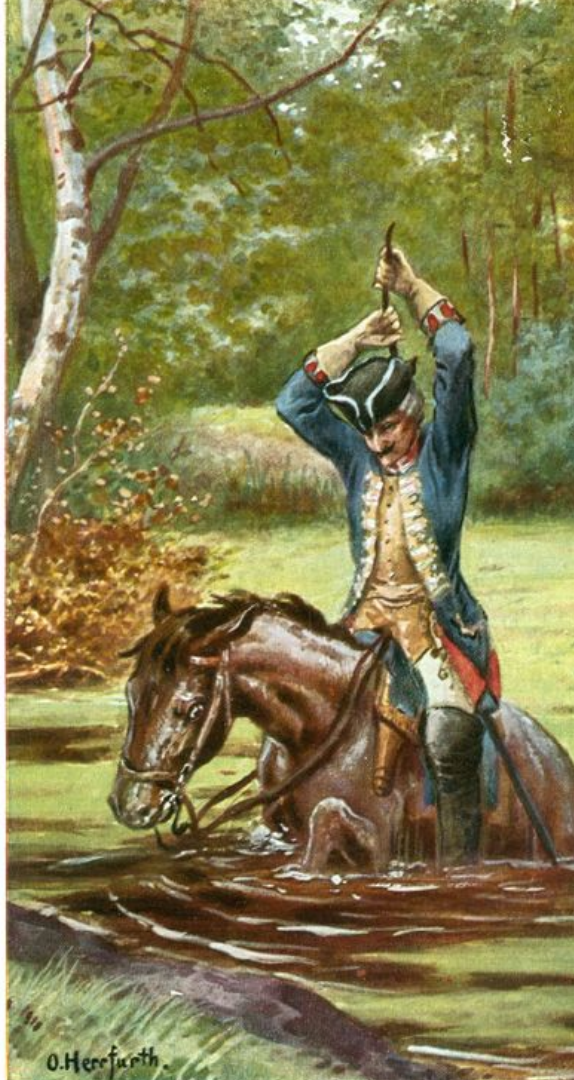
Entropia e ganho de informação

$$Entropia(S) \equiv \sum_{i=1}^{c} -p_i \log_2 p_i$$

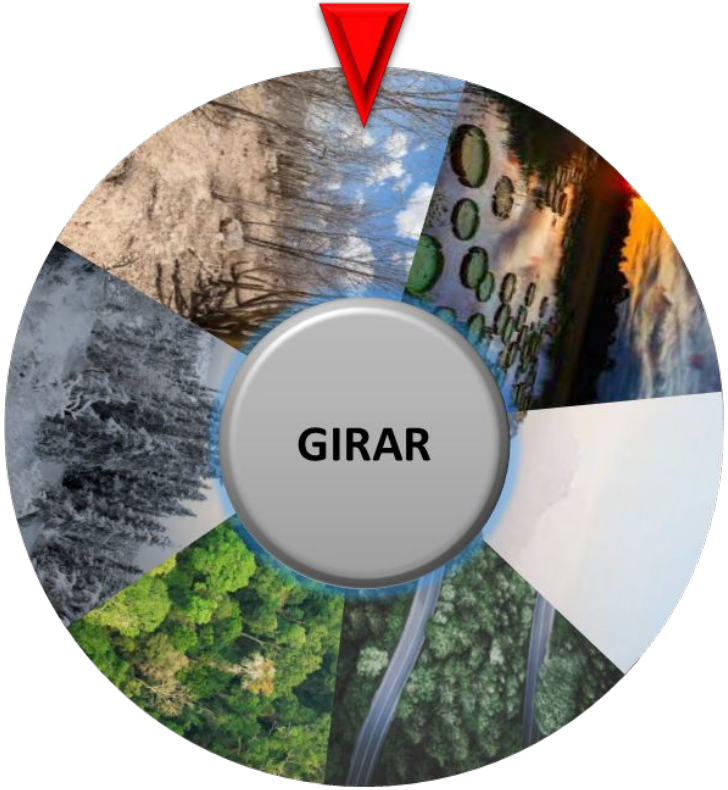
$$Gain(S, A) \equiv Entropia(S) - \sum_{v \in valores(A)} \frac{|S_v|}{|S|} Entropia(S_v)$$

Bagging e Bootstrap

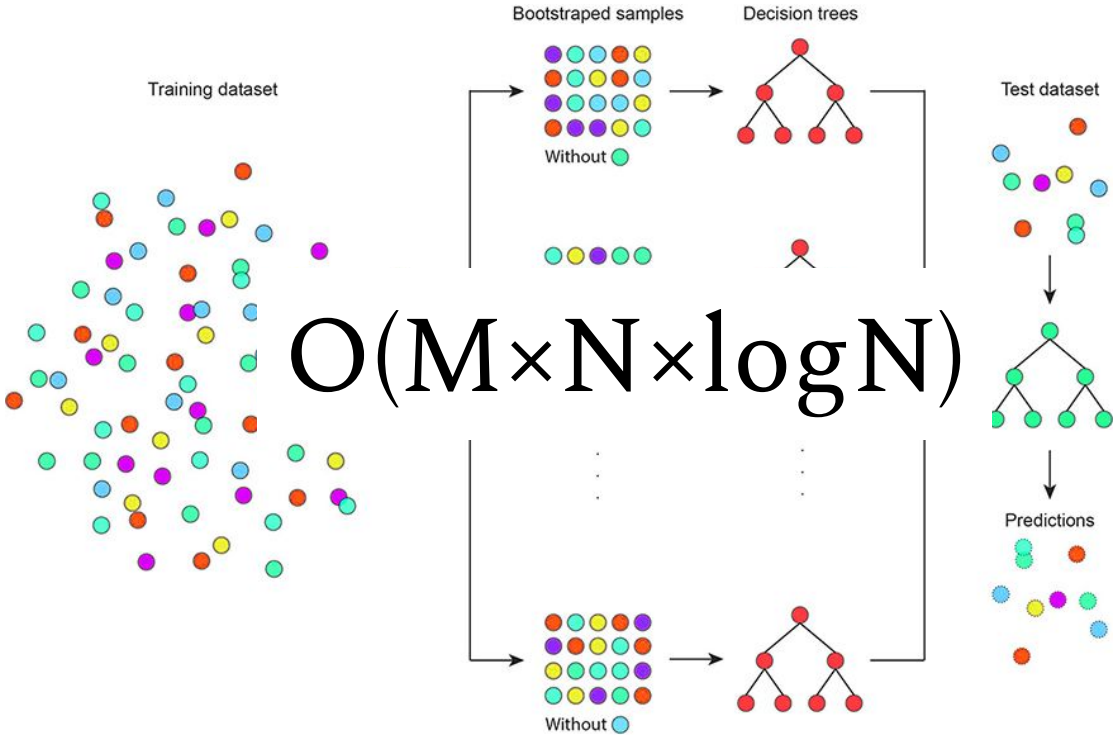
```
import numpy as np
porosity = [30.3, 21.0, 19.2, 29.1, 21.9, 23.1]
Nboot = 400
def bootstrap(x, Nboot, statfun):
    x = np.array(x)
    resampled_stat = []
    for b in range(Nboot):
        index = np.random.randint(0,len(x),len(x))
        sample = x[index]
        bstatistic = statfun(sample)
        resampled_stat.append(bstatistic)
    return np.array(resampled_stat)
porosity_bootstrap = bootstrap(porosity, Nboot, np.mean)
```



Random Forest



Random Forest



$$O(M \times N \times \log N)$$

scikit-learn