ENSEMBLES LEARNING OU APRENDIZAGEM EM CONJUNTO

Cristiane Neri Nobre

Ensembles

- Métodos que geram muitos classificadores e combinam os seus resultados

  "Consultar vários especialistas para tomada de decisão"

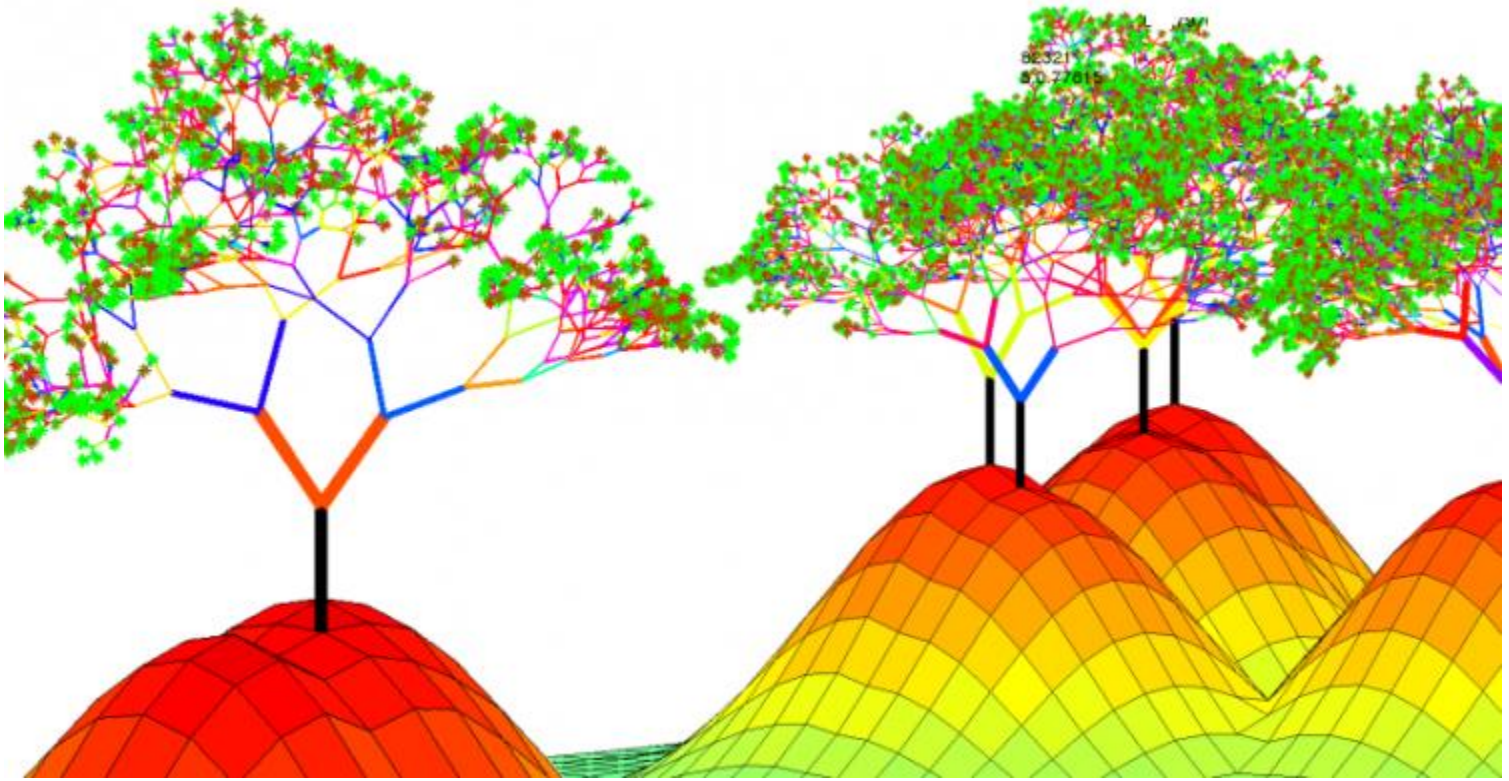
- É amplamente aceito que o desempenho de um conjunto de muitos classificadores fracos é geralmente melhor do que um único classificador, dada a mesma quantidade de informação de treinamento

Ensembles

- Como exemplos clássicos de métodos ensemble pode-se citar: **boosting**, **bagging** e recentemente **random forest**

Por quê **Forests?**
Muitas árvores!

Por quê Random **Forests?**
Muitas árvores...



Random Forests

- É um algoritmo ensemble proposto por Breiman (veja o artigo **Random Forests)**[1] que constrói muitas árvores de decisão as quais são utilizadas para classificar um novo exemplo por meio do voto majoritário

- Ou seja, o algoritmo de Random Forest (RF) é um termo geral para métodos de ensemble utilizando classificadores do tipo árvore

- Utiliza a amostragem **Bootstrap**

- Utilizado pelo Kinect da Microsoft

Disponível em https://link.springer.com/content/pdf/10.1023%2FA%3A1010933404324.pdf

Random Forests

- No método **booststrap**, *r* subconjuntos de treinamento são gerados a partir do conjunto de exemplos original. Os exemplos são amostrados aleatoriamente desse conjunto, com reposição. Logo um exemplo pode estar presente em um determinado subconjunto de treinamento mais de uma vez.

- Os subconjuntos de testes são formados pelas sequências que não estão no conjunto de treino

  Cerca de um terço das instâncias são deixados de fora da amostra de bootstrap e não são usados na construção da k-árvore.

Random Forests

Como se avalia o resultado final?

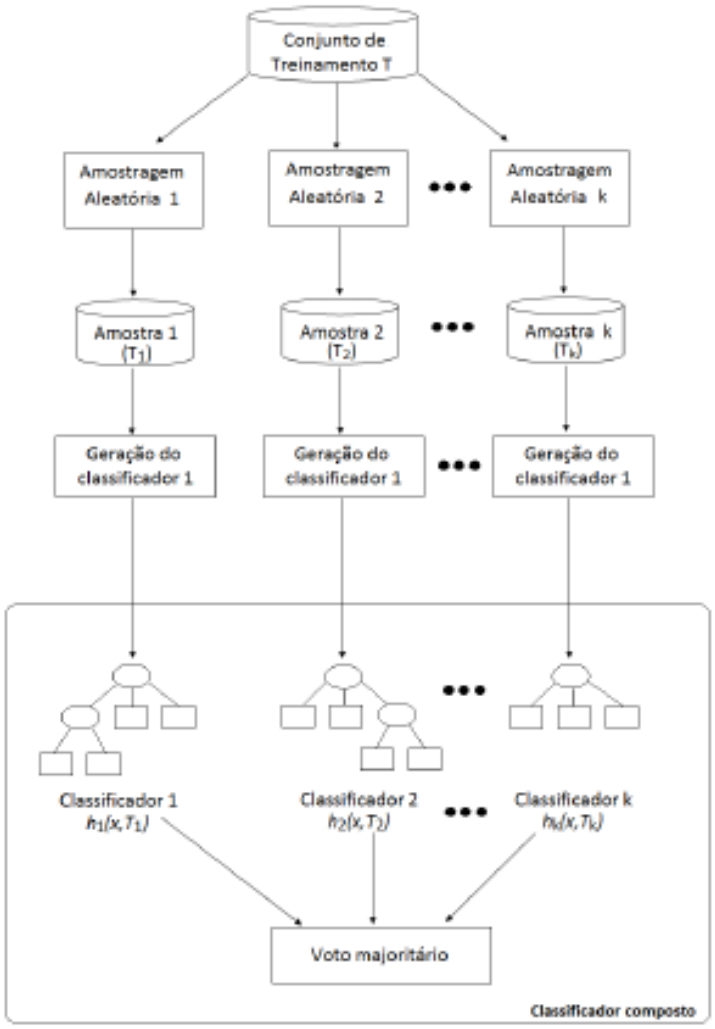
- O resultado final é dado pela média do desempenho observado em cada subconjunto de teste.

Random Forests

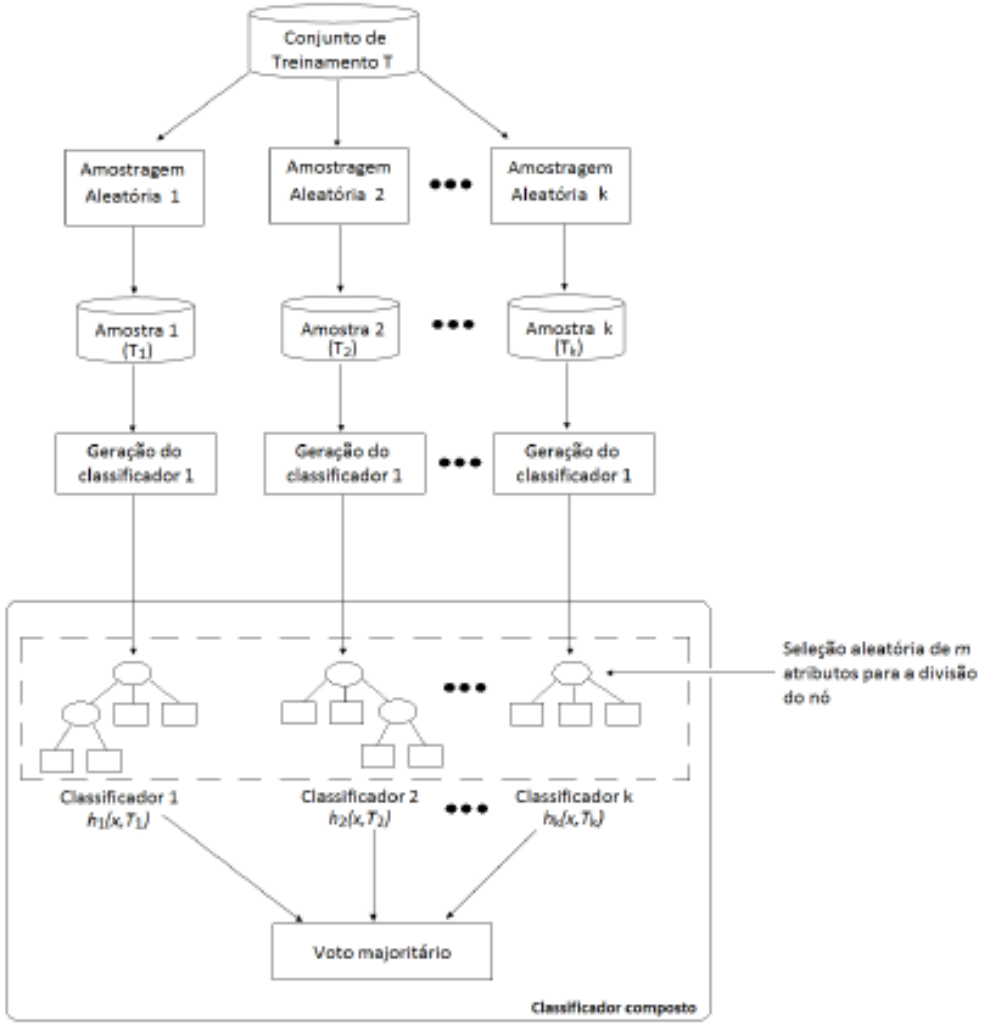
Normalmente adota-se $r \geq 100$. A ideia básica é repetir o experimento um número alto de vezes e estimar o desempenho nesses experimentos replicados.

Por este motivo, o bootstrap é um procedimento custoso.

Há vários estimadores bootstrap, e o mais comum é o e_0. Neste cada conjunto de treinamento tem *n* exemplos, amostrados com reposição do conjunto original, sendo *n* o número total de exemplos nesse conjunto



Funcionamento do Bagging



Funcionamento do Random Forest

The Random Forest Algorithm

1. For $b = 1$ to B:
   (a) Draw a bootstrap sample $\mathbf{Z}^*$ of size N from the training data.
   (b) Grow a random-forest tree T_b to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size n_{min} is reached.
      i. Select m variables at random from the p variables.
      ii. Pick the best variable/split-point among the m.
      iii. Split the node into two daughter nodes.
2. Output the ensemble of trees $\{T_b\}_1^B$.

To make a prediction at a new point x:

Regression: $\hat{f}_{\text{rf}}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$.

Classification: Let $\hat{C}_b(x)$ be the class prediction of the bth random-forest tree. Then $\hat{C}_{\text{rf}}^B(x) = \textit{majority vote}\ \{\hat{C}_b(x)\}_1^B$.

Resumo do algoritmo Random Forest

- Treina cada árvore com amostras geradas a partir do método de amostragem *bootstrap*
- Para cada conjunto de instâncias, o Random Forest considera somente *m* variáveis selecionadas aleatoriamente do conjunto de dados
- Random Forest não faz poda

O resultado final obtido a partir das árvores é dado por:

- Para problemas de **classificação**: voto majoritário
- Para problemas de **regressão**: Média dos valores preditos

Desvantagem da Random Forest

- Difícil extrair o conhecimento das árvores, apesar de o método exibir os atributos mais relevantes

  Verificar no Weka como este método funciona!

Como definir o número de atributos a serem utilizados?

- Alguns métodos utilizam **raiz quadrada** da quantidade de atributos

- Outros utilizam o log da quantidade de atributos

Referências

Breiman, L. **Random Forest**. Machine Learning (2001) 45: 5. https://doi.org/10.1023/A:1010933404324
https://link.springer.com/article/10.1023%2FA%3A1010933404324

https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#workings