Random Forest

Esembles:

Métodos que geram muitos classificadores e combinam seus resultados. É amplamente aceito que o desempenho de um conjunto de muitos classificadores fracos é geralmente melhor do que um único classificador.

Como exemplos clássicos pode-se citar:

- Boosting
- Bagging
- Stacking
- Esembles de árvores

Condições necessárias para um bom desempenho:

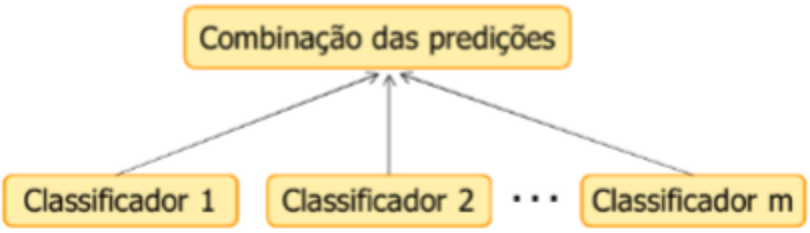
- Diversidade: classificadores base devem ser independentes é ideal cometer erros diferentes.
- Acurácia: desempenho dos classificadores base devem ser melhor que classificação aleatória.

Exemplo:

Seja três classificadores induzidos para os mesmos dados, com acurácia 0.6. Se eles cometem os mesmos erros a acurácia do ensemble será 0,6. Se eles são completamente independentes o ensembles erra classificação apenas se pelo menos 2 classificadores erram na predição.

Combinação de predições:

- Voto (média)
- Voto (média) ponderado



Bagging:

Cada classificador é induzido por uma amostra diferente do conjunto de treinamento.

- Mesmo tamanho do conjunto inicial
- Usa bootstrap

Classe definida por votação.

Tende a reduzir variância associada com classificadores base.

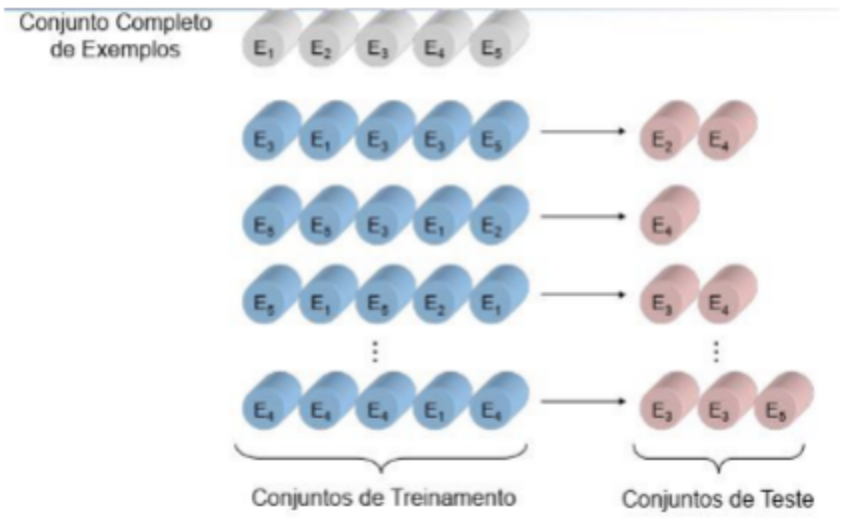
Técnica de amostragem Booststrap:

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 teste 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.

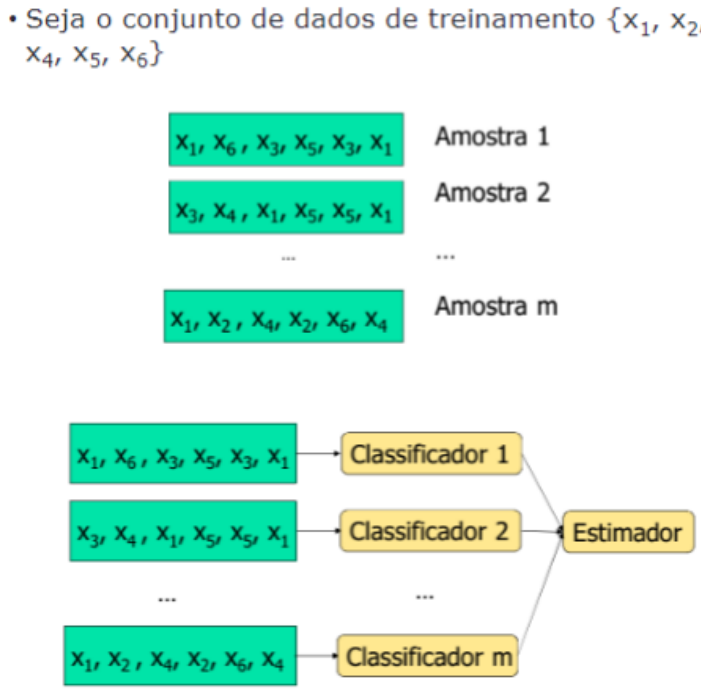
Normalmente adota r ≥ 100. A ideia é repetir o experimento um número alto de vezes e estimar o desempenho nesses experimentos replicados. Por isso, passa a ser um procedimento custoso.



Como se avalia o resultado final?

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

Bagging:



Boosting:

Adaboost é uma das técnicas mais conhecidas.

A cada interação:

- Induz o classificador
- Pondera cada exemplo do conjunto de dados completo pelo desempenho do classificador base. Quanto mais dificil ser o aprendizado, maior o peso associado ao exemplo.

Boosting funciona de forma semelhante a minimização por gradiente descendente.

Seja o conjunto de dados de treinamento $\{x_1, x_2, x_3, x_4, x_5\}$

Exemplos:	x_1	x_2	x_3	x_4	x_5
Pesos atuais:	0.2	0.2	0.2	0.2	0.2
Classificação:	C	I	C	C	I
Novos pesos:	0.2	0.4	0.2	0.2	0.4

Soma dos pesos = 1.0

C: correta
I: incorreta

Exemplos:	x_1	x_2	x_3	x_4	x_5
Pesos atuais:	0.2	0.4	0.2	0.2	0.4
Classificação:	C	I	C	I	C
Novos pesos:	0.2	0.6	0.2	0.4	0.4

Indicado para classificadores base fracos.

- Acurácia ligeiramente melhor que palpite aleatório.

Convergência rápida.

Pouco indicado para dados com ruídos e pequenos conjuntos de dados.

- Por focar em exemplos dificeis de serem classificados.

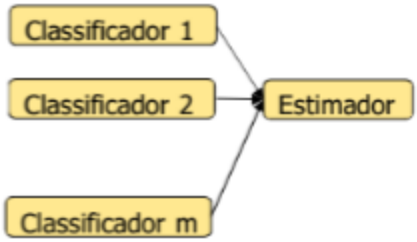
Stacking

Um algoritmo estimador aprende a combinar predições de modelos base.

- Modelos gerados por algoritmos base
- Saídas combinadas por algoritmo estimador (Algoritmo de AM)

Algoritmos base podem ser:

- Homogêneos
- Heterogêneos



Ensembles de árvores de decisão:

Combina a predição de várias árvore de decisão.

Duas principais abordagens:

- Extreme Gradient Boosting
- Random Forest

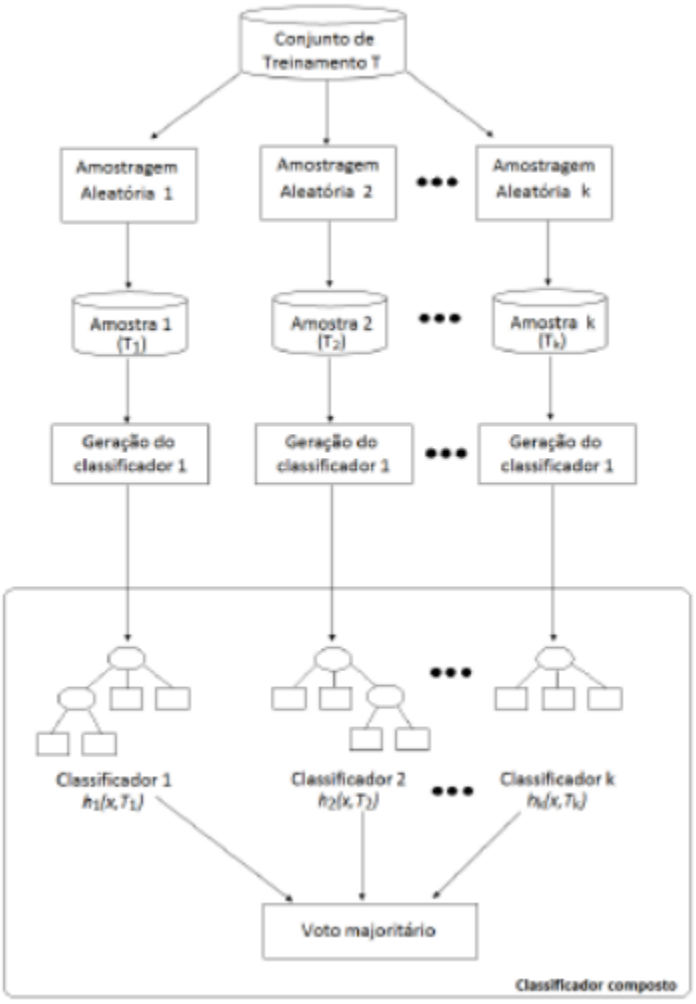
Extreme Gradient Boosting:

- XGBoost
- Combina árvores geradas pelo algoritmo CART (Gini)
- Pondera a resposta de cada árvore para reduzir complexidade do modelo final.

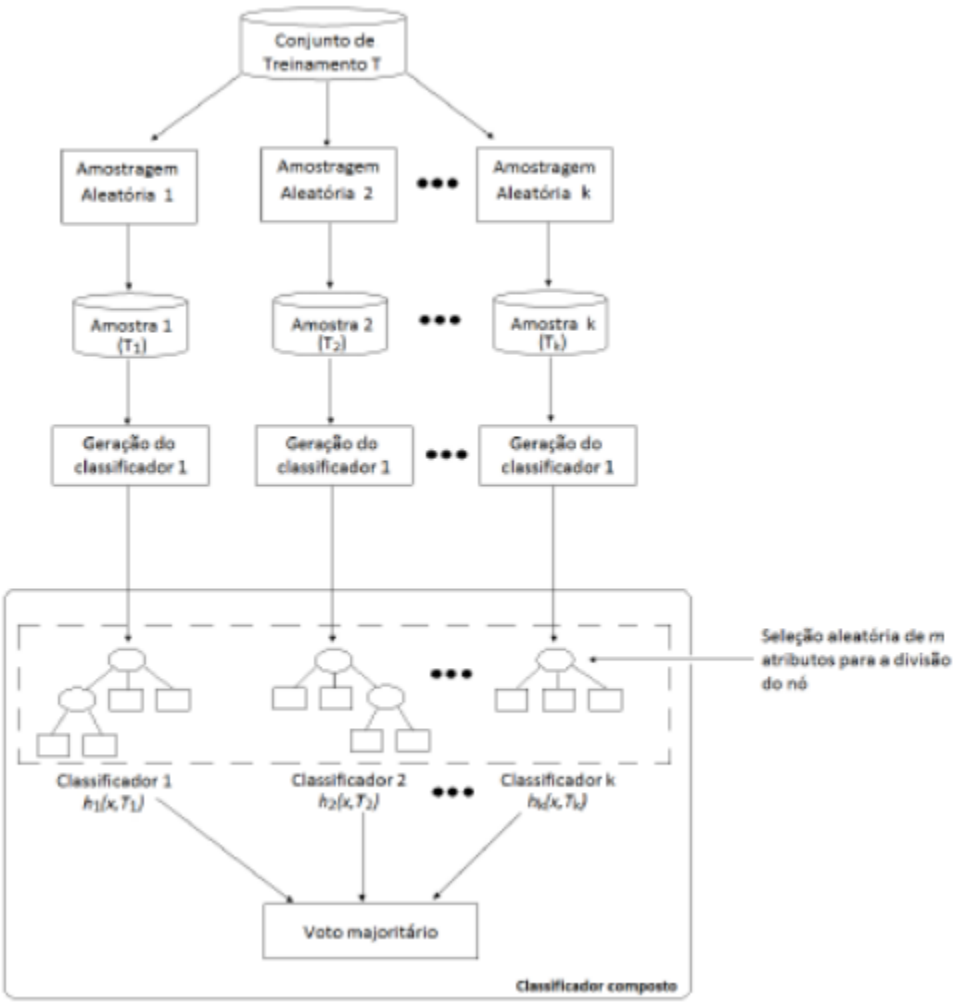
Random Forests:

Algoritmo que constrói muitas árvores para classificação. Ou seja, é um termo geral para métodos de ensemble utilizando classificadores do tipo árvore. Utiliza a amostragem Bootstrap.

Funcionamento do Bagging:



Funcionamento do Random Forest:



Algoritmo Random Forest:

- Treina cada árvore com amostras geradas a partir do método de amostragem Bootstrap.
- Para cada conjunto de instâncias, considera-se somente m variáveis selecionadas aleatoriamente do conjunto de dados.
- Não faz poda, pois seleciona um número m de amostras.
- O resultado final obtido a partir das árvores é dado por:
  - Problemas de classificação: voto majoritário
  - Problemas de regressão: média dos valores preditos

Desvantagem: Difícil extrair o conhecimento da árvore, apesar do método exibir atributos mais relevantes.

Como definir o número de atributos a serem usados:

- Alguns métodos usam a raiz quadrada da quantidade de atributos.
- Outros utilizam o log da quantidade de atributos.