**HIPERPARÂMETROS DA FLORESTA ALEATÓRIA**

Fonte: <https://medium.com/machina-sapiens/o-algoritmo-da-floresta-aleat%C3%B3ria-3545f6babdf8>

**Diferenças entre Árvore de Decisão e Florestas Aleatórias**

Como mencionado anteriormente, Floresta Aleatória é uma coleção de árvores de decisão, mas há algumas diferenças.

Se você treinar uma árvore com um dataset de treinamento e rótulos, ela vai elaborar um conjunto de regras que serão utilizadas para realizar predições.

Por exemplo, se você quiser prever se uma pessoa vai clicar em um anúncio online, você pode colecionar anúncios que a pessoa clicou no passado e algumas características que descrevam esta decisão. Se você colocar as características e rótulos em uma árvore de decisão ela vai gerar nodos e algumas regras. Você poderá então prever ser o anúncio será clicado ou não. Quando uma árvore de decisão gera regras e nodos, normalmente utiliza calculos de *ganho de informação* e *gini index*. Por outro lado, a Floresta Aleatória faz isto de modo aleatório.

Outra diferença é que árvores de decisão profundas podem sofrer de sobreajuste (*overfitting*). Florestas Aleatórias evitam o sobreajuste na maioria dos casos, pois trabalha com subconjuntos aleatórios das características e constrói árvores menores a partir de tais subconjuntos. Depois do treinamento, as subárvores são combinadas. Esta abordagem torna a computação mais lenta, dependendo de quantas árvores serão construiídas pelo Floresta Aleatória.

**Hiperparâmetros Importantes**

Os parâmetros no Floresta Aleatória são utilizados ou para aumentar o poder preditivo do modelo ou para tornar o modelo mais rápido. Irei discutir os hiperparâmetros da função random forest do pacote [sklearn](http://scikit-learn.org/" \t "_blank).

**1. Aumentar o poder preditivo**

Primeiramente, há o hiperparâmetro n\_estimators, que indica o número de árvores construídas pelo algoritmo antes de tomar uma votação ou fazer uma média de predições. Em geral, uma quantidade elevada de árvores aumenta a performance e torna as predições mais estáveis, mas também torna a computação mais lenta.

Outor hiperparâmetro importante é max\_features, que indica o número máximo de características a serem utilizadas pelo Floresta Aleatória na construção de uma dada árvore. [Sklearn](http://scikit-learn.org/" \t "_blank) oferece várias opções, descritas na sua documentação.

Um ultimo hiperparâmetro que discutiremos é o min\_sample\_leaf. Este parâmetro indica o número mínimo de folhas que devem existir em uma dada árvore.

**2. Aumentar a velocidade do modelo**

O hiperparâmetro n\_jobs informa quantos processadores o algoritmo pode utilizar. Se ele tiver valor 1, pode utilizar apenas um processador. O valor -1 significa que não há limite na quantidade de processadores a ser utilizada.

O parâmetro random\_state torna o resultado do modelo replicável. O modelo será produzido do mesmo modo se ele tiver um valor definido de random\_state e se forem utilizados os mesmos parâmetros com o mesmos dados de treinamento.

Por último, há o oob\_score(também chamado de oob sampling), que é um método de validação cruzada para floresta aleatória. Neste tipo de amostragem (sampling), cerca de um terço dos dados não é utilizado no treinamento e pode ser utilizado para avaliar a performance. Estas amostras são chamadas out of the bag samples. É uma técnica similar ao método de validação cruzada leave one out, mas sem nenhum custo computacional extra.

**Vantagens e Desvantagens**

Uma das vantagens do Floresta Aleatória é que ele pode ser utilizado tanto para regressão quanto para classificação e é fácil visualizar a importância relativa que ele atribui para cada característica na suas entradas.

O Floresta Aleatória é também considerado um algoritmo muito fácil e acessível, pois seus hiperparâmetros com valores *default* geralmente produzem um bom resultado de predição. O número de hiperparâmetros não é tão grande e são muito fáceis de compreender.

Um dos grande problemas em aprendizagem de máquina é subreajuste (*overfitting*), mas a maior parte do tempo isto não ocorrerá tão facilmente com um classificador de Floresta Aleatória qualquer. Isto porquê, se há árvores suficiente na floresta, o classificador não irá sobreajustar o modelo.

A maior limitação do Floresta Aleatória é que uma quantidade grande de árvores pode tornar o algoritmo lento e ineficiente para predições em tempo real. Em geral, estes algoritmos são rápidos para treinar, mas muito lentos para fazer predições depois de treinados. Uma predição com mais acurácia requer mais árvores, o que faz o modelo ficar mais lento. Em muitas aplicações do mundo real o Floresta Alatória é rápido o suficiente, mas pode certamente haver situações onde a performance em tempo de execução é importante e outras abordagens são mais apropriadas.

E é claro, o Floresta Aleatória é uma ferramenta de modelagem preditiva e não descritiva. Isto significa que, se você está procurando uma descrição dos relacionamentos nos seus dados, você deve escolher outras abordagens.

**Aplicações**

O algorimo Floresta Aleatória é utilizado em muitas áreas diferentes, tal como setor bancário, mercado financeiro, medicina, comércio eletrônico. No setor bancário ele é utilizado, por exemplo, para detectar clientes que irão utilizar os serviços bancários mais frequentemente que outros e pagar suas dívidas pontualmente. Neste domínio ele também é utilizado para detectar fraudes de clientes que querem lesar o banco. No setor financeiro, ele é utilizado para determinar o desempenho futuro de uma ação. Na área de saúde ele é utilizado para identificar a correta combinação de componentes em medicina, e também é utilizado para analisar o histórico médico de um paciente para identificar doenças. Por último, em comércio eletrônico o Floresta Aleatória é utilizado para determinar se um cliente irá gostar do produto ou não.

**Resumo**

O Floresta Aleatória é um excelente algoritmo para ser utilizado nos primeiros estágios de criação do processo de desenvolvimento de um modelo, para se ter uma ideia de performance. Devido à sua simplicidade, é difícil construir um Floresta Aleatória “ruim”. Este algoritmo é também uma boa opção se você precisa desenvolver um modelo em curto espaço de tempo. Além disso, ele provê um bom indicador de importância para as características.

O algoritmo Floresta Aleatória é também quase imbatível em termos de performance. É claro que você pode provavelmente encontrar um modelo com melhor performance, talvez uma rede neural, mas estas levaram muito mais tempo para desenvolver. Outra vantagem é que ele pode trabalhar com diferentes dipos de entradas, tais como binárias, categóricas ou numéricas.

Em resumo, Floresta Aleatória é uma ferramenta (geralmente) rápida, simples e flexível, embora apresente algumas limitações.