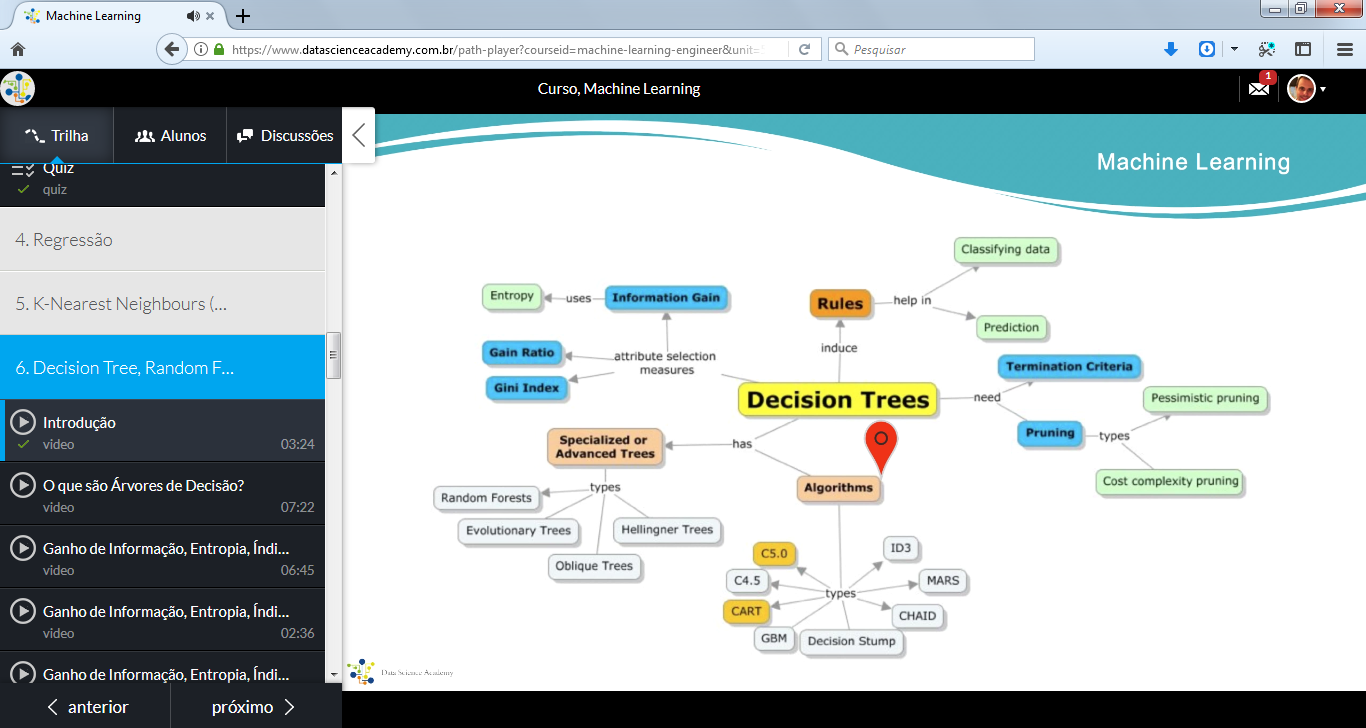
**Decision Tree, Random Forest e Métodos Ensemble**



**Árvores de Decisão**

Podem ser usadas para problemas de:

Classificação (para variáveis categóricas)

Regressão (para variáveis quantitativas/numéricas)



**Processo de aprendizagem**

**Greedy Search**

O algoritmo procura maximizar o passo atual sem olhar para o passo seguinte, a fim de alcançar uma otimização global.

**Verificação dos dados de saída**

Índice Gini

Ganho de informação (entropia)

Redução da variância (para problemas de regressão)

**Como definir o nó raiz e como realizar a divisão do conjunto de dados?**

**Estratégia Gulosa (Greedy Selection)**

Necessita da medida da “impureza” do nó, através das técnicas de Entropia, Índice de Gini e erro de classificação ou redução de variância para problemas de regressão.

Métodos:

Divisão baseada em atributos nominais

Divisão Binária

Divisão Múltipla

Divisão baseada em atributos contínuos

Decisão Binária

Discretização

Estática

Dinâmica

**Entropia**

Na física entropia é a medida de grau de desordem de um sistema.

Na teoria da informação, entropia é a medida da incerteza dos dados. Pode ser definida como o grau de impureza de um conjunto.

****

O algoritimo realiza o cálculo da entropia antes e após o cálculo das partições e decide se mantém o nó se houver ganho de informação. O modelo vai sendo reajustado até que os nós que apresentam a maior redução na entropia sejam selecionados.

**Ganho de informação**

É a redução esperada da entropia, ou seja, busca-se saber o nível de pureza dos dados, o que significa dados melhor organizados que permitem extrair as regras que eles representam.

**Índice de Gini**

O Índice de Gini é usado para medir a probabilidade de dois itens aleatórios pertencentes à mesma classe. No caso de um conjunto de dados real, esse valor de probabilidade é 1. A medida de Gini de um nó é a soma dos quadrados das proporções das classes.

O algoritmo usa os índice acima para determinar os nós de uma decision tree. Quando temos muito ruído nos dados, pode ser interessante mudar o parâmetro do algoritmo de “entropia” para “gini”.

**Regras de Parada (Stopping Rules)**

Técnicas utilizadas pelos algoritmos:

Índice Gini

Qui-quadrado

Ganho de informação

Redução de Variância

**Pruning (Poda da árvore)**

**Abordagens**

- A árvore de decisão é concluída antes que uma classificação perfeita dos dados de treinamento seja alcançada.

- Ocorre o excesso de ajuste nos dados gerando um modelo e, em seguida, a árvore é podada (Pruning) para se tornar generalizável.

**Algoritmo ID3 (Iterative Dichotomizer 3)**

Como funciona sua heurística:

- Começa com todos os exemplos de treino;

- Escolhe o teste (atributo) que melhor divide os exemplos, ou seja agrupa exemplos da mesma classe ou exemplos semelhantes;

- Para o atributo escolhido, é criado um nó filho para cada valor possível do atributo;

- Transporta os exemplos para cada filho considerando o valor do filho;

- Repete o procedimento para cada filho não "puro".

E como o algoritmo sabe o melhor atributo a escolher?

Através do Ganho de Informação e Entropia.

**Evoluções do ID3**

C4.5

C5.0

CART (Classification and Regression Trees)

**Notas:**

O R no momento possui ferramentas mais intuitivas e organizadas para implementação da Decision Tree.

Para problemas de classificação e regressão o algoritmo funciona melhor se trabalhar com valores numéricos em vez de strings. Potanto, sempre que possível, converta variáveis categóricas para numéricas (de-para).

Bootstrap: Várias amostragens com reposição

**Randon Forest**

Conjunto de árvores de decisão aplicados à uma solução (Método ensemble). O modelo faz estimativas utilizando vários modelos e define, através de pesos ou votação a melhor resposta.

**Randon Forest Classifier -** Composto por árvores de decisão de classificação.

**Randon Forest Regressor** - Composto por árvores de decisão de regressão.

**Métodos Ensemble**

Nada mais é que o agrupamento de modelos de machine learning. Define-se um estimador base, que é na verdade o modelo de machine learning onde o algoritmo vai construir uma série de estimadores, que pode ser controlado via parâmetros, para depois combinar cada estimativa, através de votação ou pesos, para gerar o resultado final.

**Tipos**

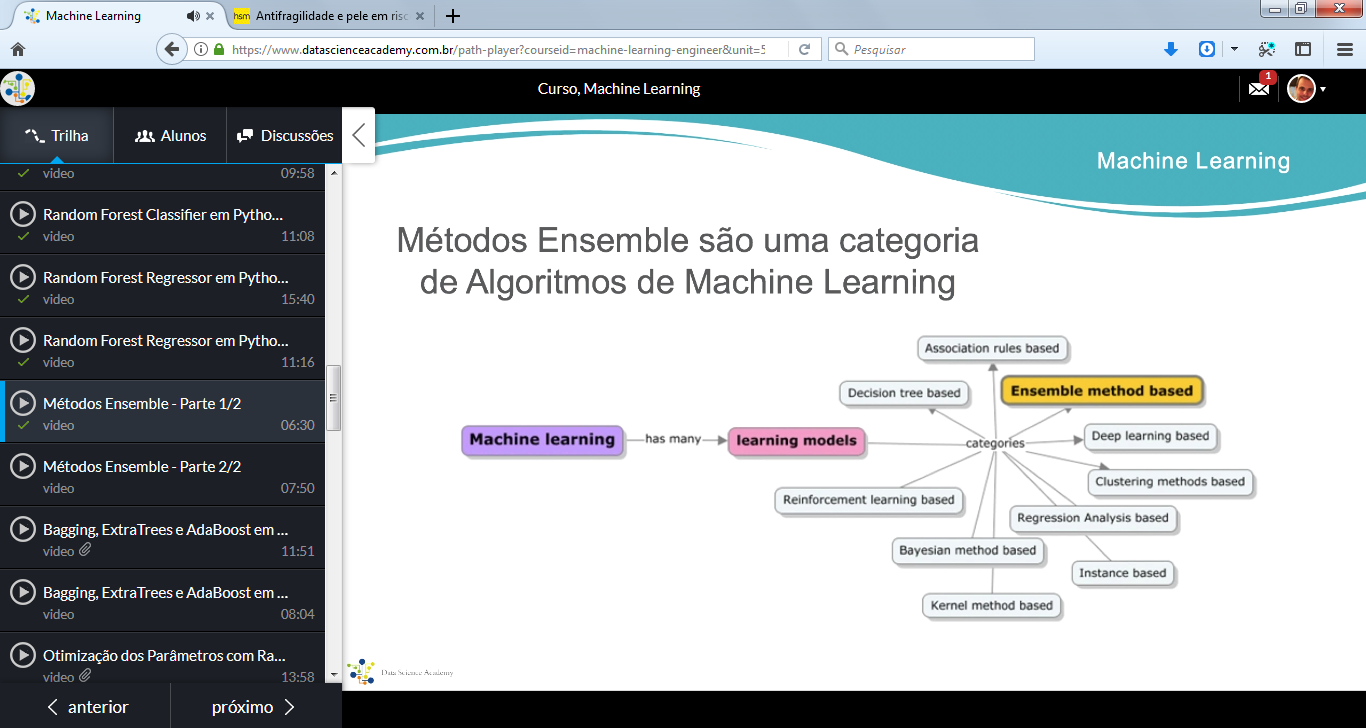
Random Forest

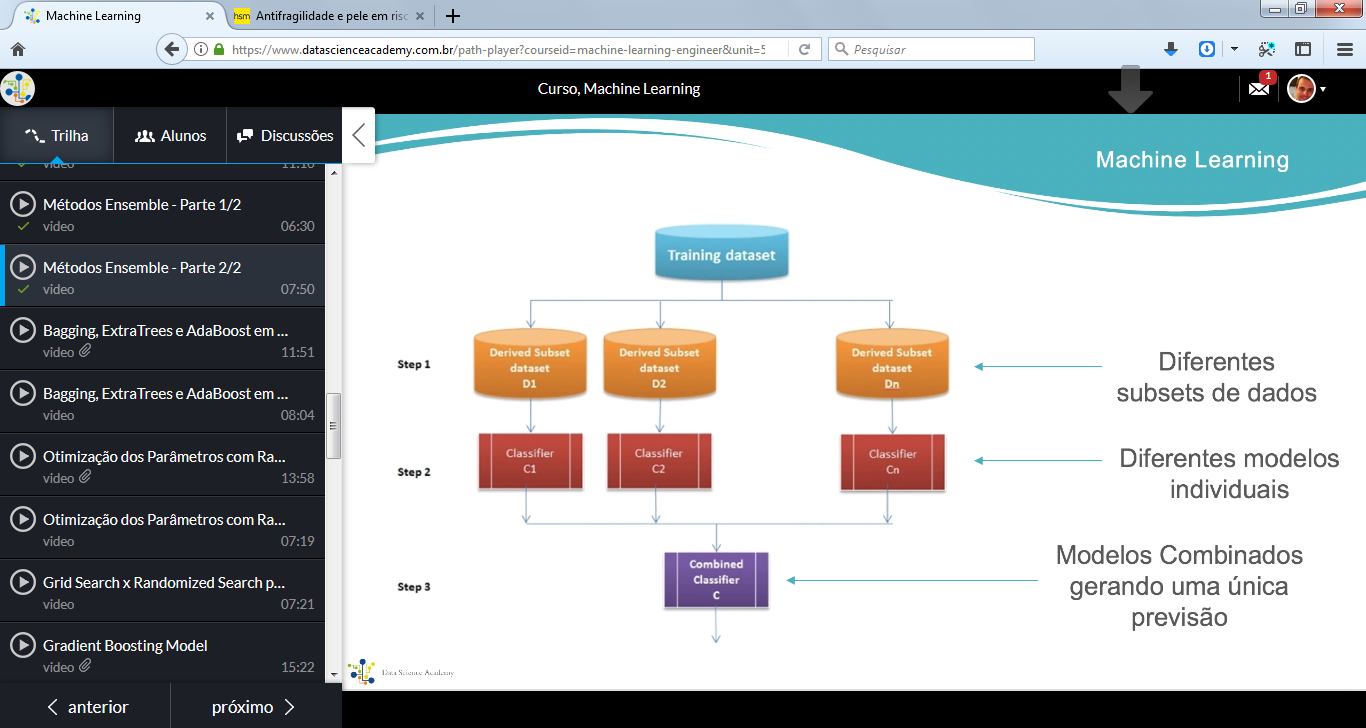
Bagging

AdaBoost

Gradient Boosting

Muitos modelos diferentes, quando reunidos, fortalecem os modelos fracos e geram um melhor desempenho global.



**Metodologia** 

1 - O dataset é dividio em vários subsets;

2 - Os subsets divididos irão alimentar modelos individuais;

3 - Cada modelo individual gerará uma estimativa que será agregada gerando um resultado final.

**Nota:** O processo acima é utilizado apenas ao dataset de treino. Após definido o modelo final, o dataset de teste passará somente por ele, gerando assim a avaliação final do modelo.

**Bagging (Bootstrap Aggregation)**

É uma técnica onde obtemos uma amostra do dataset original, recalculamos as previsões e retiramos uma média, no caso de regressões, ou votamos no caso de classificações. Essa técnica é capaz de criar um modelo com viés semelhante ao modelo original, porém com variância reduzida, principalmente com funções lineares.

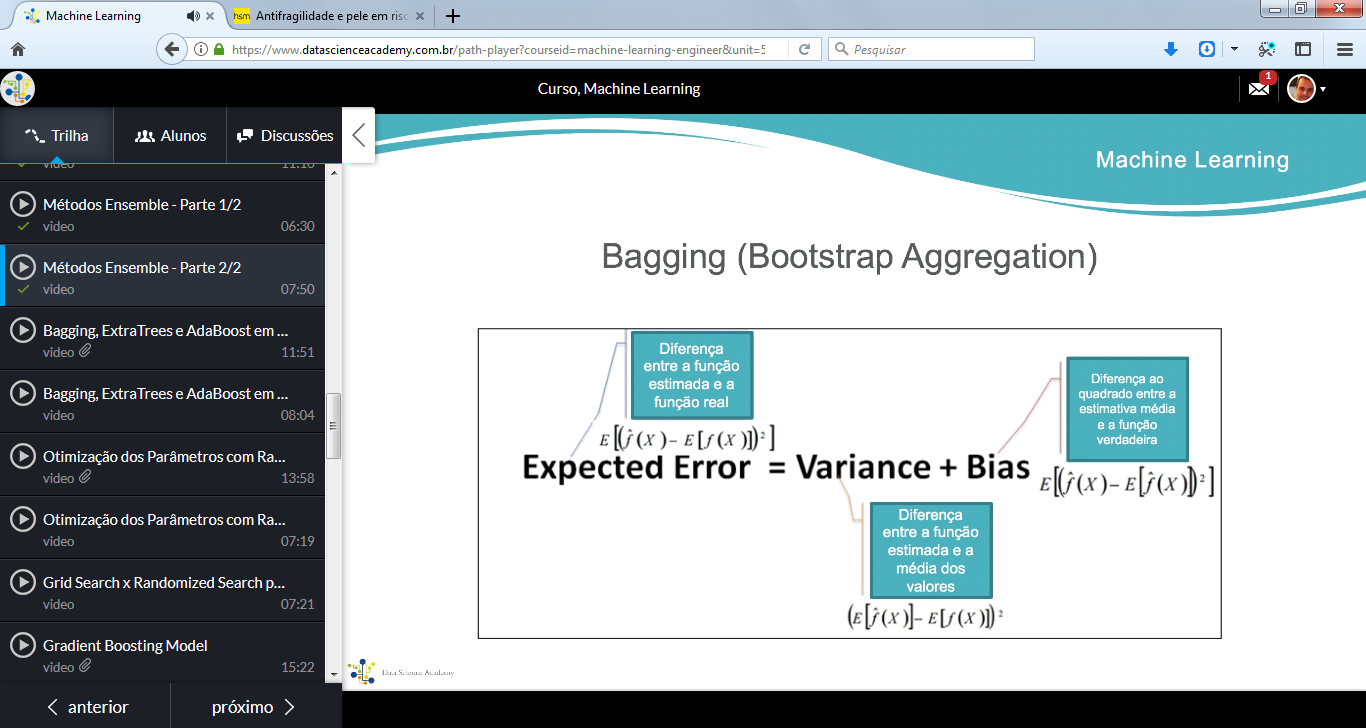
O método bagging gera multiplos subconjuntos a partir do dataset de treino original, escolhidos de maneira aleatória.

Existem 3 importantes fases na construção de modelos baseados nessa técnica:

1- É construída uma amostra baseada em Bootstrap (reamostragem) que contém aproximadamente 60% dos registros originais no dataset.

2- Os classificadores são treinados usando cada amostra.

3- É usada votação da maioria para identificar a classe final do classificador ensemble.

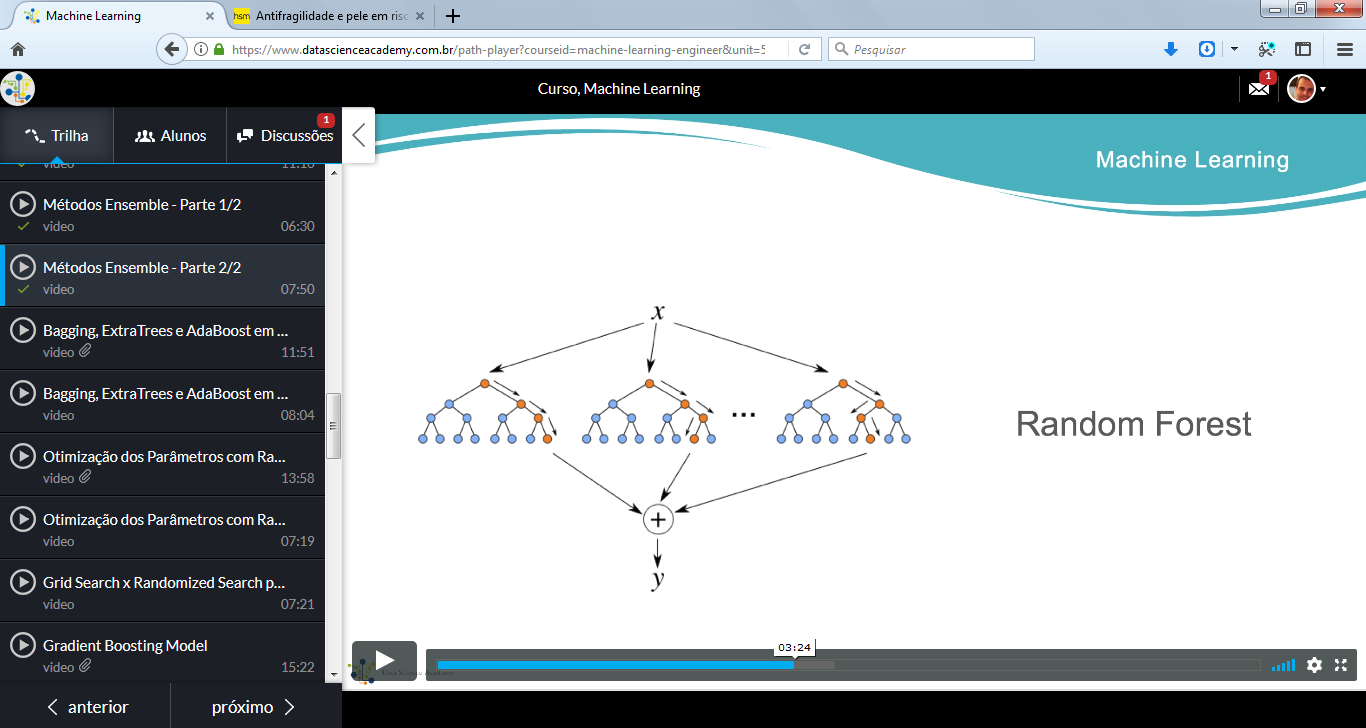


**Extratrees Classifier**

Segue a mesma metodoligia do bagging, porém ele randomiza os estimadores base, que são as árvores de decisão.

**Random Forest**

É uma extensão do método bagging. Amostras do dataset de treino são retiradas com reposição, mas as árvores são criadas de modo que diminua a correlação entre os classificadores individuais.



Uma boa característica do Random Forest é determinar a importância de cada atributo para a variável de saída estimada, podendo ser usada para feature selection.

Random Forest se propõe a criar pequenas árvores de decisão, que apesar de individualmente não terem uma boa acurácia, quando unidas tornam o método robusto.

**Boosting**

Esse método utiliza a média ponderada da saída gerada por múltiplos modelos, isso o torna ideal para trabalhar com modelos fracos. Essa técnica permite trabalhar com grandes conjuntos de dados.

**Metodologia**

- Os modelos fracos são treinados sequencialmente.

- Os dados usados para treinar cada classificador são baseados na performance do classificador anterior.

- Cada classificador vota e contribui para o resultado final.

A cada iteração, os classificadores que tiverem respostas incorretas, terão o seu peso reduzido enquanto os classificadores corretos receberão mais peso, aumentando sua importância no sistema.

**Adaboost**

São algoritmos baseados no método boosting que cria uma sequencia de modelos que tenta corrigir os erros dos modelos anteriores dentro de sequencia. Os modelos recebem pesos de acordo com sua acurácia e ao final do processo são agregados para dar um modelo final.

O modelo também atribui peso às instâncias do dataset, classificando o quão fácil ou difícil elas são para o modelo de classificação, permitindo que o algoritmo tenha mais ou menos atenção às instâncias durante a criação do modelo.

**Otimização de Parâmetros**

**Randomized Search**

Constrói vários modelos com diferentes combinações de parâmetros, para em seguida apresentar a combinação que deu o melhor resultado.

**Grid Search**

Realiza metodicamente combinações entre todos os parâmetros do algoritmo, criando uma tabela. Requer uma capacidade computacional maior que o Randomized Search.

**Gradient Boosting**

É o método mais avançado entre os métodos ensemble. Esse método reúne técnicas diferentes, gerado modelo que são complexos e altamente sofisticados. Também é chamado de Gradient Boosted Regression Trees (GBRT), utilizado para problemas de classificaçõe e regressão. Une os métodos Gradient Descent e o método Boosting.

O gradient boosting internamente só trabaha com variáveis quantitativas, portanto quando quisermos trabalhar com variáveis categóricas, devemos tranformá-las em variáveis numéricas.

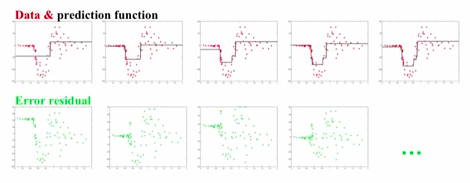
Basicamente são realizadas 3 etapas na construção do modelo:

1 - Gera um regressor (estimador base);

2 - Computa o erro residual (diferença entre o valor previsto e o observado);

3 - Aprende a prever o resíduo.

À medida que o algoritmo aprende através dos resíduos, ele ajusta sua função matemática para aprimorar os resultados:



**Parâmetros mais importantes:**

- Número de árvores de regressão (n\_estimators)

- Profundidade de cada árvore (max\_depth)

- Loss Function (loss)

**Reguralização (Evitar Overfitting)**

Quando trabalhamos com modelos de árvore de regressão, podemos evitar o overfitting aplicando algumas técnicas de regularização:

1 - Alterar a estrutura da árvore;

2 - Shrinkage (reduzir a árvore mudando a quantidade de estimadores);

3 - Stochastic Gradient Boosting;