**Uma Introdução Visual para *Machine Learning***

Em *machine learning*, os computadores aplicam técnicas de **aprendizado estatístico** para identificarem padrões em dados de forma automática. Estas técnicas podem ser utilizadas para fazer previsões altamente precisas.

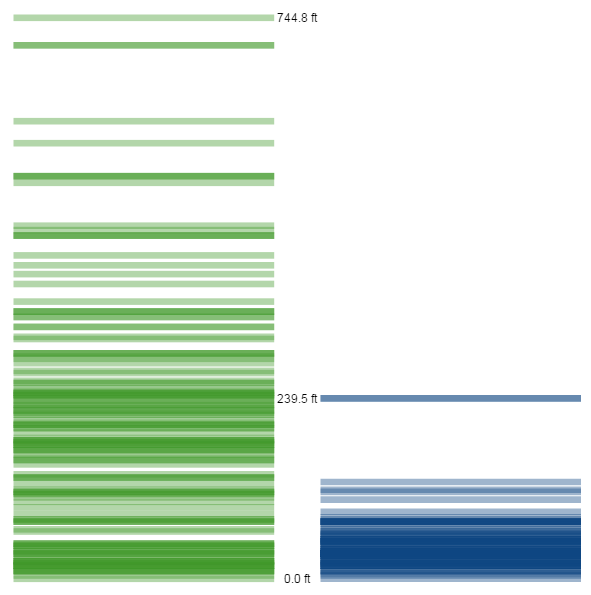
Fazendo uso de dados sobre moradias, criaremos um modelo de *machine learning* para distinguir casas em Nova York e em São Francisco.

**Primeiramente, um pouco de intuição**

Digamos que você tivesse que determinar se uma moradia está em São Francisco ou em Nova York. Em termos de *machine learning*, categorizar dados é uma tarefa de **classificação.**

Uma vez que São Francisco é relativamente montanhoso, a elevação de uma moradia pode ser uma boa forma de distinguir as duas cidades.

Baseando-se na elevação da moradia abaixo, você poderia afirmar que uma moradia acima de 73 metros deveria ser **classificada** como uma em São Francisco.

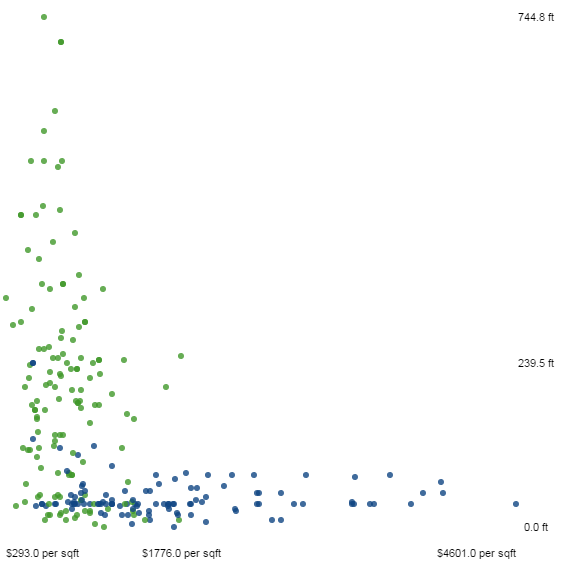


**Acrescentando nuances**

Adicionar outra **dimensão** permite mais nuances. Por exemplo, apartamentos em Nova York podem apresentar o metro quadrado extremamente caro.

Então visualizar elevação e preço por metro quadrado em um **gráfico de dispersão** nos ajuda a distinguir moradias de elevação baixa.

Os dados sugerem que, entre as casas que estão abaixo de e inclusive 73 metros, aquelas que custam mais de U$ 19,116.7 por metro quadrado estão na cidade de Nova York. Dimensões em um conjunto de dados são chamados de **características (*features),* preditores (*predictors*)** ou **variáveis (*variables*)**.

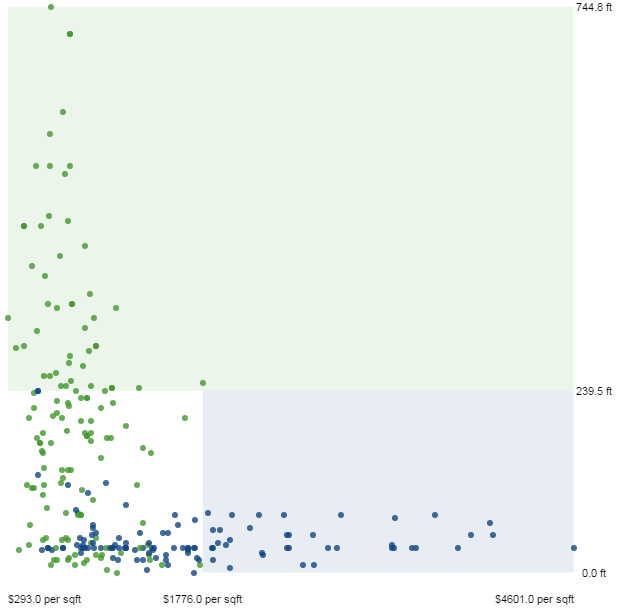


**Desenhando fronteiras**

Você pode visualizar sua elevação (acima de 73 metros) e preço por metro quadrado (acima de U$ 19,116.7) como fronteiras de regiões em seu gráfico de dispersão. Moradias plotadas nas regiões verdes e azuis estariam em São Francisco e Nova York respectivamente.

Identificar fronteiras em dados usando matemática é a essência do aprendizado estatístico.

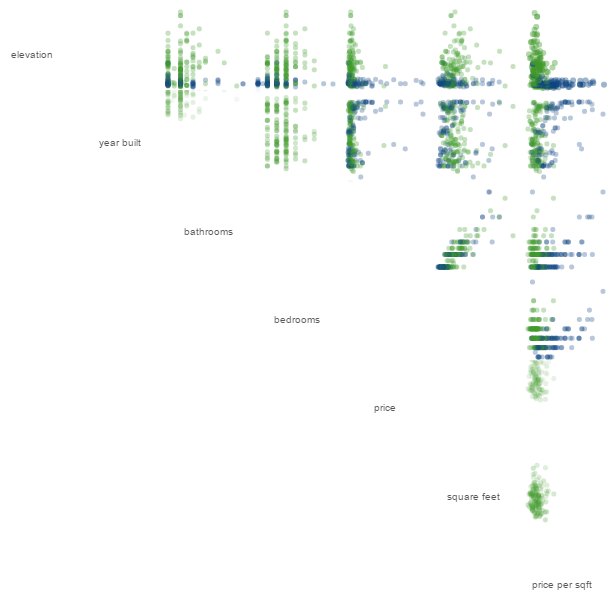
Obviamente, você precisará de informação adicional para distinguir as moradias com valores baixos tanto em relação à elevação e ao metro quadrado.

****

O conjunto de dados que estamos utilizando para construir o modelo tem 7 diferentes dimensões. Criar um modelo também é chamado de **treinar *(training)*** um modelo.

Na figura abaixo, nós visualizamos as variáveis em uma **matriz de gráficos de dispersão** **(*scatterplot matrix*)** para mostrar os relacionamentos entre cada par de dimensões.

Há padrões claros nos dados, mas as fronteiras para delineá-los não são óbvias.



**E agora, *machine learning***

Encontrar padrões em dados é onde entra o *machine learning*. Métodos de *machine learning* usam aprendizado estatístico para identificar fronteiras.

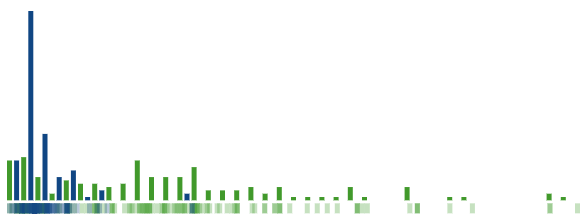
Um exemplo de método de *machine learning* é a **árvore de decisão (*decision tree*)**. Árvores de decisão olham para uma variável de cada vez e são métodos razoavelmente acessíveis, embora rudimentares.

**Encontrando melhores fronteiras**

Vamos revisitar a fronteira de elevação de 73 metros proposta anteriormente para ver como podemos melhorar além da nossa intuição. Obviamente, isso requer uma perspectiva diferente.

Ao transformar nossa visualização em um **histograma (*histrogram*)**, nós podemos melhor visualizar a frequência que as moradias aparecem em cada elevação.

Enquanto a moradia de maior elevação em Nova Yorké de 73 metros, a maioria delas parece ter uma elevação bem menor.

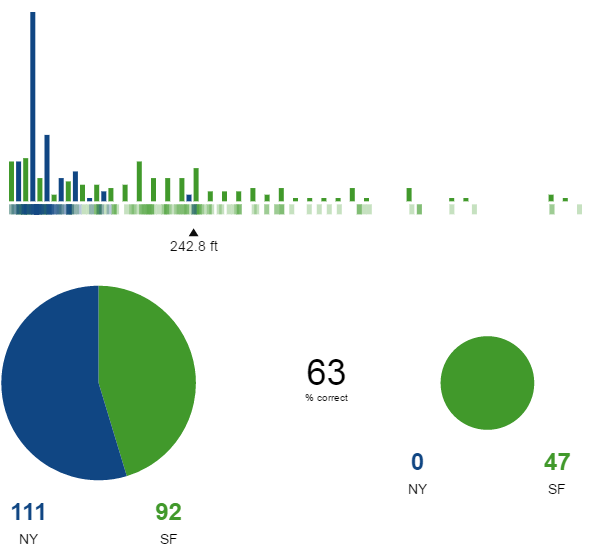


**Sua primeira bifurcação (*fork*)**

Uma árvore de decisão utiliza declarações **se-então** para definir padrões em dados. Por exemplo, **se** a elevação de uma moradia é acima de um número, **então** a casa está provavelmente em São Francisco.

Em *machine learning*, estas declarações são chamadas de **bifurcações (*forks*)**, e elas dividem os dados em dois **ramos (*branches*)** baseando-se em algum valor.

O valor entre os ramos é chamado de **ponto de divisão (s*plit ploint*)*.*** Moradias à esquerda deste ponto são categorizadas de uma forma, enquanto aquelas à direita são categorizadas de outra. Um ponto de divisão é a fronteira para uma árvore de decisão.

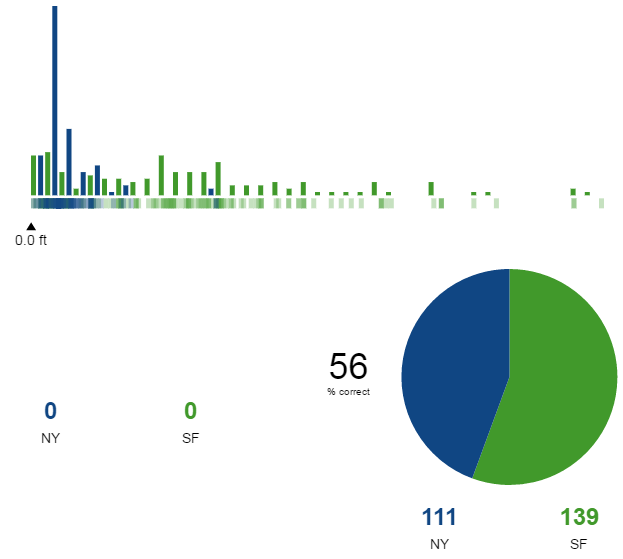


***Tradeoffs***

Selecionar um ponto de decisão tem *tradeoffs*. Nossa divisão inicial (73 metros) classifica de forma incorreta algumas moradias em São Francisco como se fossem de Nova York.

Olhe abaixo para a maior fatia verde no gráfico de torta, todas elas são moradias em São Francisco que foram classificadas de forma incorreta. Elas são chamadas de **falso-negativo (*false negatives*)**.

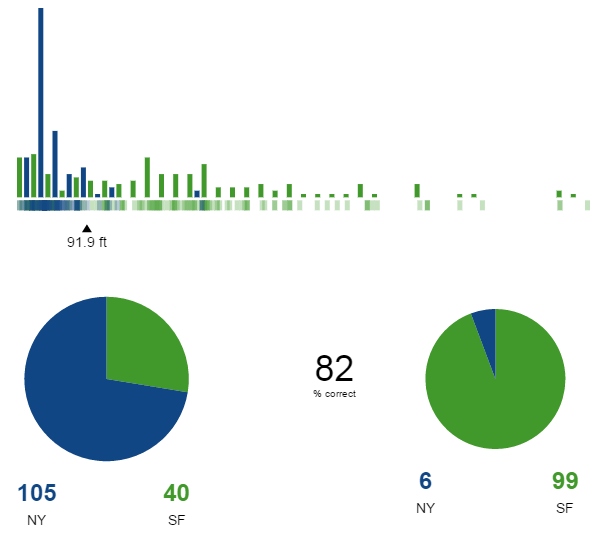
Entretanto, um ponto de divisão destinado a capturar toda moradia em São Francisco também incluirá muitas moradias de Nova York também. Elas são chamadas de **falso-positivo (*false positives*)**.



**A melhor divisão**

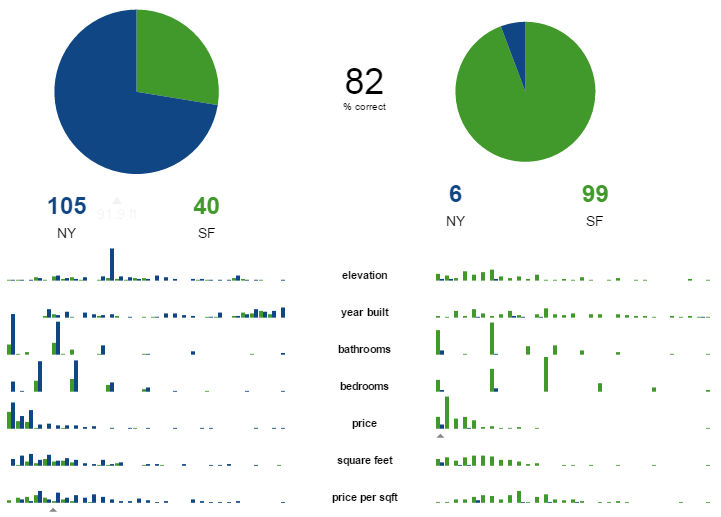
Na **melhor divisão (*best split*)**, os resultados de cada ramo deveriam ser tão homogêneos ou puros quanto possível. Há diversos métodos matemáticos à disposição para escolha para calcular a melhor divisão.

Conforme vemos abaixo, mesmo a melhor divisão em uma única *feature* não separa totalmente as moradias de São Francisco com as de Nova York.

****

**Recursão**

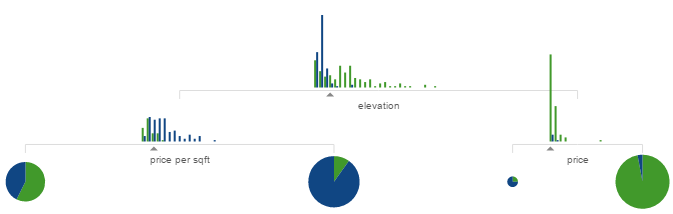
Para adicionar outro ponto de divisão, o algoritmo repete o processo acima nos subconjuntos dos dados. Essa repetição é chamada de **recursão (*recursion*)**e é um conceito que aparece frequentemente em modelos de treino. Os histogramas abaixo mostram a distribuição de cada subconjunto de dados, repetidos para cada variável.

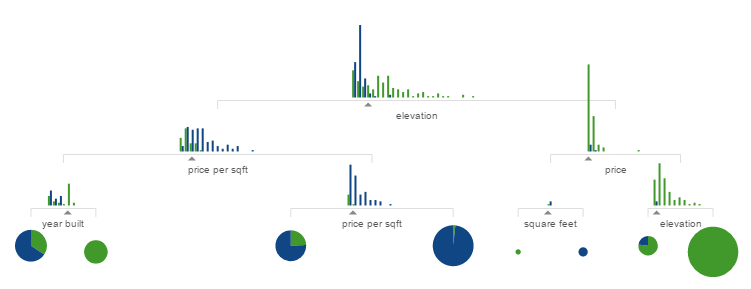
****

A melhor divisão variará baseada em qual ramo da árvore você está olhando. Para moradias de baixa elevação, o preço por metro quadrado no valor de U$ 1061 per sqft é a melhor variável para a próxima declaração **se-então**. Para moradias de elevação mais alta, o preço no valor de U$ 514,500 é a melhor variável.

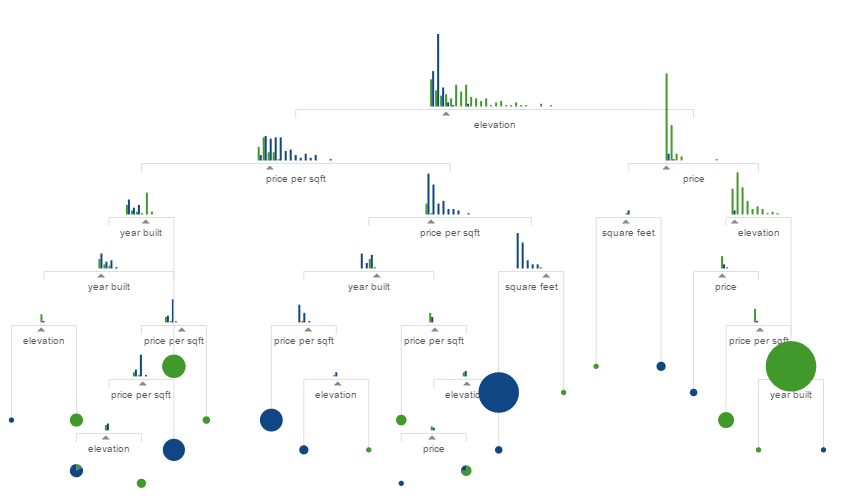
**Crescendo a árvore**

Bifurcações adicionais adicionarão novas informações que podem aumentar a **precisão de previsão** **(*prediction accuracy*)** de uma árvore. Dividindo em mais um nível, a precisão da árvore aumenta para 84%.





Adicionar mais camadas, chegamos a 96%:

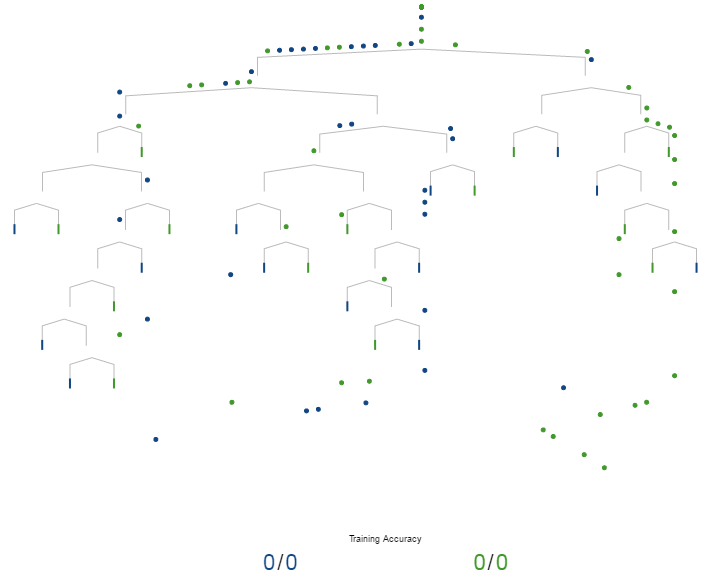
****

Você poderia até continuar a adicionar mais ramos até as previsões da árvore chegar a **100% de precisão**, de forma que no final de cada ramo, as moradias são puramente São Francisco ou puramente Nova York.

Estes ramos finais são chamados de **folhas (*leaf nodes*)**. Nosso modelo de árvore de decisão classificará as moradias em cada folha de acordo com quais classes de moradia está na maioria.

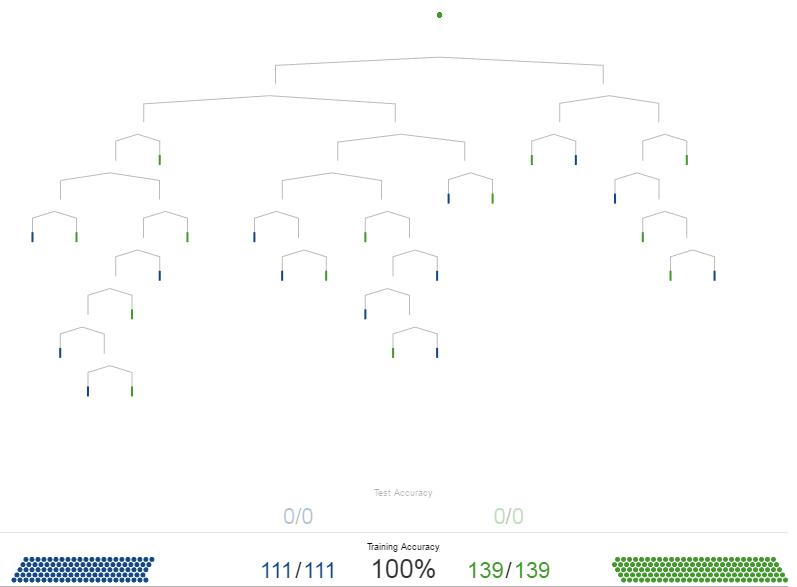
**Fazendo previsões**

O recém-treinado modelo de árvore de decisão determina se uma moradia está em São Francisco ou em Nova York ao processar cada ponto de dados através dos ramos.



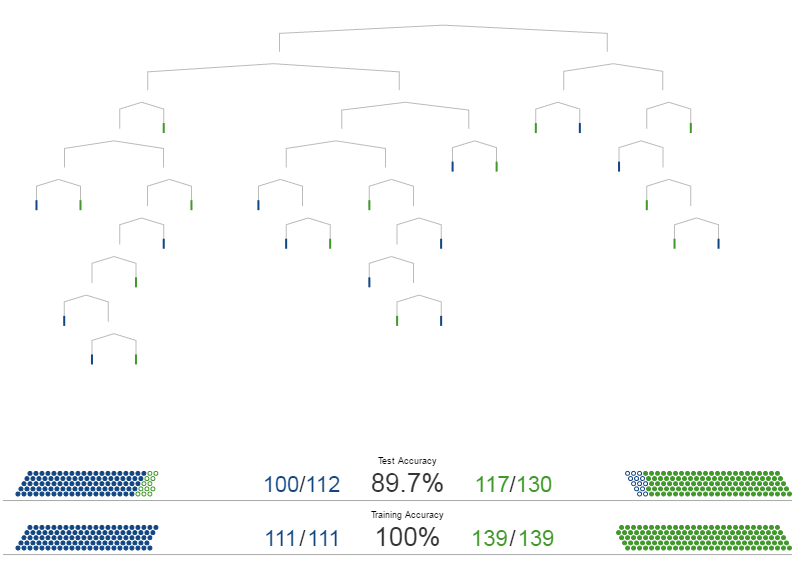
Aqui você pode visualizar os dados que foram utilizados para o treinamento da árvore fluir através dos ramos da árvore. Estes dados são chamados de **dados de treinamento (*training data*)** porque foram usados para treinar o modelo.

Por conta de que nós crescemos a árvore até os 100% de precisão, essa árvore mapeia cada ponto de dados de treinamento perfeitamente para cada cidade.



**Verificação da realidade**

É claro que o que mais interessa é como a árvore desempenha em dados nunca vistos anteriormente. Para **testar** o desempenho da árvore com novos dados, nós precisamos aplica-la a pontos de dados que nunca foram vistos. Estes dados são chamados de ***test data*.** De forma ideal, a árvore deveria desempenhar de forma similar tanto em dados conhecidos como não conhecidos. Na figura abaixo, a árvore está abaixo do ideal:



Estes erros são devidos ao ***overfitting***. Nosso modelo aprendeu a tratar cada detalhe no conjunto de dados de treino como importante, mesmo detalhes que são irrelevantes. *Overfitting* é parte de um conceito fundamental em *machine learning* que nós explicaremos em um próximo post.

**Recapitulando**

1 – *Machine Learning* identifica padrões utilizando **aprendizado estatístico** e computadores através do descobrimento de **fronteiras** em conjuntos de dados. Você pode utilizar para fazer previsões.

2 – Um método para fazer previsões é chamado de **árvore de decisão**, que utiliza uma série de declarações se-então para identificar fronteiras e definir padrões em dados.

3 – ***Overfitting*** acontece quando algumas fronteiras são baseadas em **distinções que não fazem a diferença**. Você pode ver se um modelo ocorre *overfitting* por meio de dados de teste fluindo através do modelo.