Uma Introducao Visual ao Aprendizado de Maquina

: Português

Em **aprendizado de máquina**, computadores aplicam **técnicas estatísticas** de aprendizado para automaticamente identificar padrões em dados. Estas técnicas podem ser utilizadas para realizar previsões com alta precisão.

Role a página para baixo. Utilizando um conjunto de dados de imóveis, nós criaremos um modelo de aprendizado de máquina para distinguir imóveis localizados em Nova Iorque/EUA de imóveis localizados em São Francisco/EUA.

Primeiramente, algumas intuições

Vamos supor que o objetivo seja determinar se um imóvel encontra-se em **São Francisco** ou em **Nova Iorque**. Em aprendizado de máquina, esta tarefa de categorizar pontos de dados é chamada de uma tarefa de **classificação**.

Uma vez que São Francisco é uma cidade tipicamente montanhosa, a altitude de um imóvel pode ser uma boa maneira de distinguir as duas cidades.

Baseado na informação do gráfico ao lado direito sobre a altitude dos imóveis, é possível **classificar** que um imóvel que esteja acima de 73 metros de altitude esteja localizado em São Francisco.

Adicionando variações

Ao adicionar uma outra dimensão conseguimos mais variações. Por exemplo, em Nova Iorque, o custo de um apartamento pode ser extremamente caro por metro quadrado.

Com isto, ao visualizar a altitude *e* o preço por metro quadrado de um imóvel em um **gráfico de dispersão** é possível distinguir imóveis localizados em uma baixa altitude.

Os dados sugerem que, entre imóveis que estejam até ou abaixo de 73 metros de altitude, aqueles que custam mais de $19,116.7 por metro quadrado estão localizados na cidade de Nova Iorque.

Dimensões em um conjunto de dados são chamados de **recursos**, **fatores preditivos**, ou **variáveis**. [1]

Desenhando limites

É possível visualizar a altitude (>73 metros) e o preço por metro quadrado (>$19,116.7) e observações como os limites das regiões no gráfico de dispersão. Os imóveis plotados em regiões verde e azul estariam em São Francisco e Nova Iorque respectivamente.

Identificar limites em dados utilizando matemática é a essência da aprendizagem estatística.

Fica claro que será necessário informação adicional para distinguir imóveis que estejam com baixa altitude e preços baixos por metro quadrado.

ROLE A PÁGINA PARA BAIXO

O conjunto de dados utilizado para criar o modelo possui 7 diferentes dimensões. A ação de criar um modelo é também conhecido como **treinar** um modelo.

No lado direito do gráfico, pode-se visualizar as variáveis em uma matriz de dispersão para mostrar os relacionamentos entre cada par de dimensões.

Há claramente padrões nos dados, mas os limites para delinear eles não são óbvios.

E finalmente, aprendizado de máquina

Localizar padrões em conjuntos de dados é onde o aprendizado de máquina entra. Métodos de aprendizado de máquina fazem uso de aprendizado estatístico para identificar limites.

Um exemplo típico de métodos aprendizado de máquina são **árvores de decisão**. Árvores de decisão olham para uma variável de cada vez e trata-se de um método de aprendizado de máquina relativamente acessível (embora rudimentar)

Identificando melhor limites

Vamos rever o limite de altitude de 73 metros proposto anteriormente para ver como é possível melhorar a intuição inicial.

Claramente, isto requer uma perspectiva diferente.

Ao transformar a visualização em um **histograma**, é possível visualizar melhor como os imóveis aparecem frequentemente a cada elevação.

Enquanto que o imóvel mais alto em Nova Iorque está a 73 metros de altitude, a maioria destes imóveis parecem estar em

altitudes mais baixas.

Nossa primeira bifurcação

Uma árvore de decisão utiliza cláusulas if-then para definir padrões em dados.

Por exemplo, **if** a elevação de um imóvel é maior que algum número pré definido, **then** o imóvel provavelmente está localizado em São Francisco.

Em aprendizado de máquina, estas cláusulas são chamadas de **bifurcações** (**forks** em inglês) e são responsáveis pela divisão dos dados em duas ramificações baseadas no mesmo valor.

O valor entre as ramificações é nomeado de ponto de divisão — split point em inglês. Imóveis localizados no lado esquerdo destes pontos são categorizados de uma forma enquanto que os imóveis localizados no lado direito são categorizados de outra. Um **ponto de divisão** é a árvore de decisão na versão de um limite.

Conflitos de escolhas

Escolher um ponto de divisão tem vantagens e desvantagens. Nossa divisão inicial (~ 73 metros) classifica incorretamente algumas casas em São Francisco como se fossem Nova Iorque.

Observe a grande fatia verde no gráfico de pizza à esquerda, estas são todas os imóveis de São Francisco que estão mal classificados. Estes dados são nomeados de **falsos negativos**.

No entanto, um ponto de divisão destinado a capturar todos os imóveis em São Francisco incluirá muitas casas de Nova Iorque também. Estes dados são nomeados de **falsos positivos**.

A melhor divisão

Na **melhor divisão** a se conseguir, os resultados de cada ramificação precisam estar homogêneos (ou puros) o quanto possível. Existem vários métodos matemáticos que podem ser escolhidos para calcular a melhor divisão a ser conseguida. [2]

Como podemos ver, mesmo a melhor divisão em um único recurso não separa completamente os imóveis de São Francisco dos imóveis localizados em Nova Iorque.

Recursão

Para adicionar outro ponto de divisão, o algoritmo repete o processo anterior sobre os subconjuntos de dados. Esta repetição é chamada de recursividade, e é um conceito que aparece com frequência nos modelos de treinamento. [3]

Os histogramas à esquerda exibem a distribuição de cada subconjunto de dados, repetido para cada variável.

A melhor divisão irá variar de acordo com o ramo da árvore o qual está se observando. [4]

Para imóveis localizados em baixa altitude, o preço por metro quadrado gira em torno de $1.061, que é a melhor variável para a próxima instrução if-then. Para imóveis localizados em uma altitude maior, o preço gira em torno de $514.500

.

O crescimento de uma árvore

Bifurcações adicionais adicionarão novas informações que podem aumentar a **precisão da previsão** da árvore.

Dividindo em uma camada mais profunda, a precisão da árvore melhora **84%**.

Ao adicionar várias camadas a mais, é possível alcançar uma precisão de **96%**.

É possível, inclusive, continuar a adicionar ramificações até as previsões da árvore serem **100% precisas**, de modo que no final de cada ramificação, identificar os imóveis localizados puramente em São Francisco ou puramente em Nova Iorque.

Estes ramos localizados no final da árvore são nomeados de nós folha. O modelo de árvore de decisão construído vai classificar os imóveis em cada nó folha segundo o qual classes de imóveis sejam a maioria.

Fazendo previsões

O modelo de árvore de decisão recém treinado determina se um imóvel está localizado em São Francisco ou em Nova Iorque, executando cada ponto de dados por meio de ramificações.

Aqui é possível visualizar os dados que foram usados para treinar o fluxo da árvore através da árvore.

Estes dados são nomeados **dados de treinamento** porque são utilizados para treinar o modelo.

Devido ao crescimento da árvore até que atingisse 100% de precisão, esta árvore mapeia cada ponto de dados de treinamento perfeitamente até que se localize a cidade.

Confrontação com a realidade

E claro, o fator mais importante é descobrir como a árvore se comportará sobre dados desconhecidos.

Para testar o desempenho da árvore com novos dados, é preciso aplicá-la a pontos de dados que ela nunca tenha visto anteriormente. Estes dados não utilizados anteriormente são chamados de **dados de teste**.

O ideal é que a árvore, seja executada de forma semelhante com ambos os dados, os conhecidos previamente e os desconhecidos por ela.

Sendo assim, isto não é o suficiente. [5]

Esses erros são devido ao **sobre-ajuste** (**overfitting**) em inglês. Nosso modelo aprendeu a tratar todos os detalhes no treinamento como dados importantes, mesmo os detalhes que foram identificados como irrelevantes.

Sobre-ajuste é parte de um conceito fundamental de aprendizado de máquina que será detalhado em nossa próxima publicação. [6]

Recapitulando

1. **Aprendizado de máquina** identifica padrões por meio de **aprendizado estatístico** e computadores **identificando limites** em conjuntos de dados. Com isto, é possível fazer previsões.
2. Um método utilizado para fazer previsões é conhecido como árvores de decisão, que fazem uso de uma série de cláusulas if-then para identificar limites e definir padrões em conjuntos de dados.

3. O **sobre-ajuste** (Overfitting) ocorre quando algum limite é baseado em distinções que não possuem diferença. É possível constatar se um modelo possui sobre-ajuste ao observar o fluxo dos dados de teste através de um modelo.

O que virá a seguir?

Em nossa próxima publicação, exploraremos o sobre-ajuste, e como se relaciona com os conflitos de interesses fundamentais em aprendizagem de máquina.

Perguntas? Idéias? Gostaríamos muito de ouvir de você. Envie um tweet em @r2d3us (https://twitter.com/r2d3us) ou escreva para team@r2d3.us (mailto:team@r2d3.us).

Finalmente, obrigado a Marcelo JS Costa (LinkedIn (https://www.linkedin.com/in/marcelojscosta), Twitter (https://twitter.com/marcelojscosta)) por sua atividade voluntária oferecendo-se a traduzir o texto para nós!

R2D3 is an experiment in expressing statistical thinking with interactive design. Find us at @r2d3us (https://twitter.com/r2d3us).

Questions? Check out the FAQs (/about/faqs/).

Stephanie interprets R2

Stephanie is currently at Stitch Fix (http://multithreaded.stitchfix.com/algorithms/) (**& hiring !!!** (http://multithreaded.stitchfix.com/careers/)). In the past, she's been at Cardiogram (https://cardiogr.am), Sift Science (https://siftscience.com/), Google, Bain & Company (http://www.bain.com/), and Vector Capital (http://www.vectorcapital.com/). She's got a MS in Statistics from Stanford.

Find Stephanie: LinkedIn (http://www.linkedin.com/in/stephaniejyee) Twitter (http://twitter.com/stephaniejyee) Email (mailto:yee@r2d3.us)

Tony visualizes with D3

Tony is a designer who loves data visualizations and information design. He is currently a Principal Designer Noodle Analytics (http://noodle.ai/). Prior to Noodle, Tony led user experience and product design at H2O (http://h2o.ai/) and at Sift Science (https://siftscience.com/). He holds an MFA in Interaction Design at the School of Visual Arts (http://interactiondesign.sva.edu/) in New York City, where he tried to change congress with a fancy infographic (http://letsfreecongress.org).

Find Tony: Portfolio (http://tonyhschu.ca/) Twitter (http://twitter.com/tonyhschu/) Blog (http://blog.tonyhschu.ca) LinkedIn (https://www.linkedin.com/in/tonyhschu) Email (mailto:chu@r2d3.us)