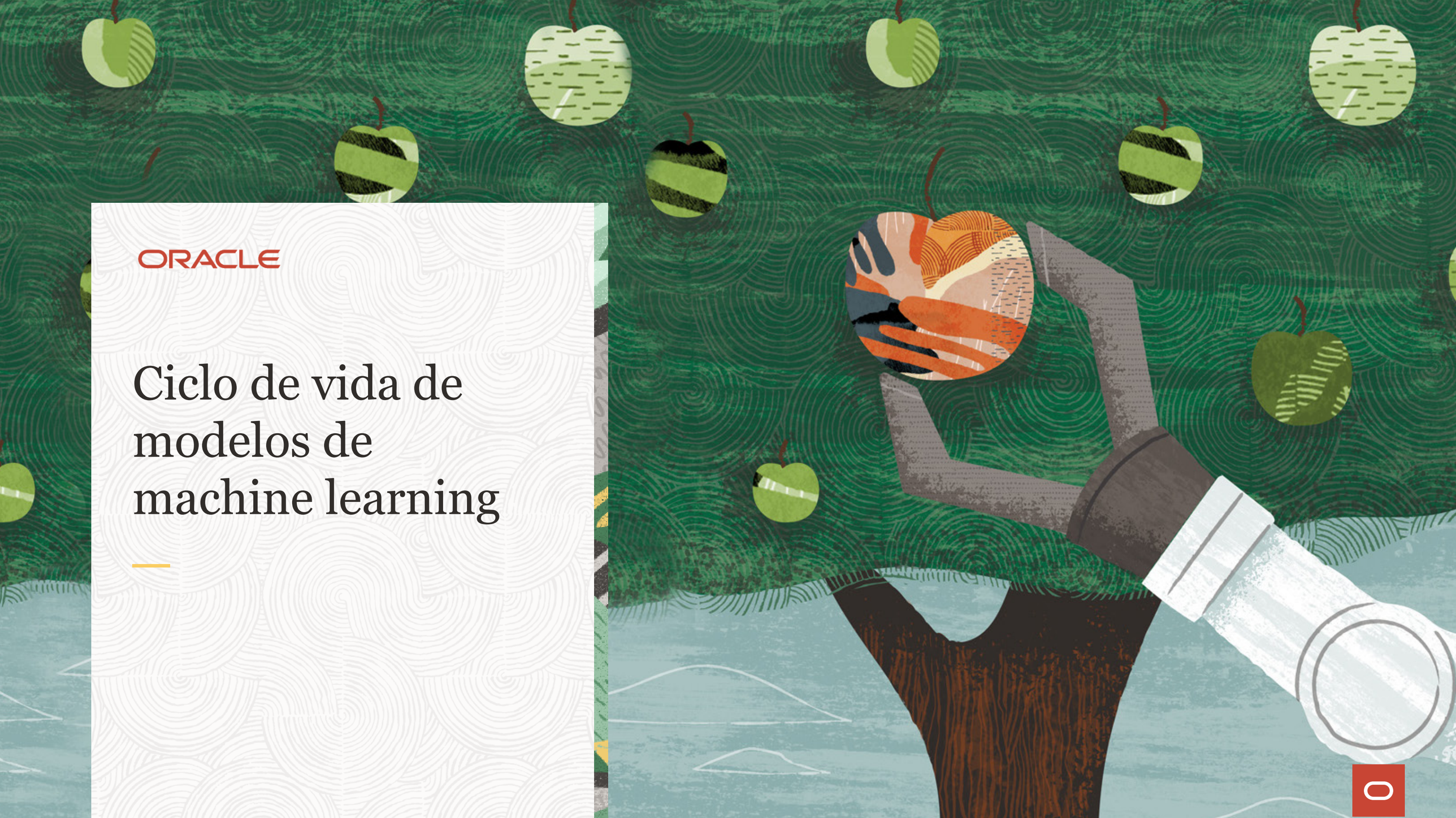
ORACLE

Ciclo de vida de modelos de machine learning

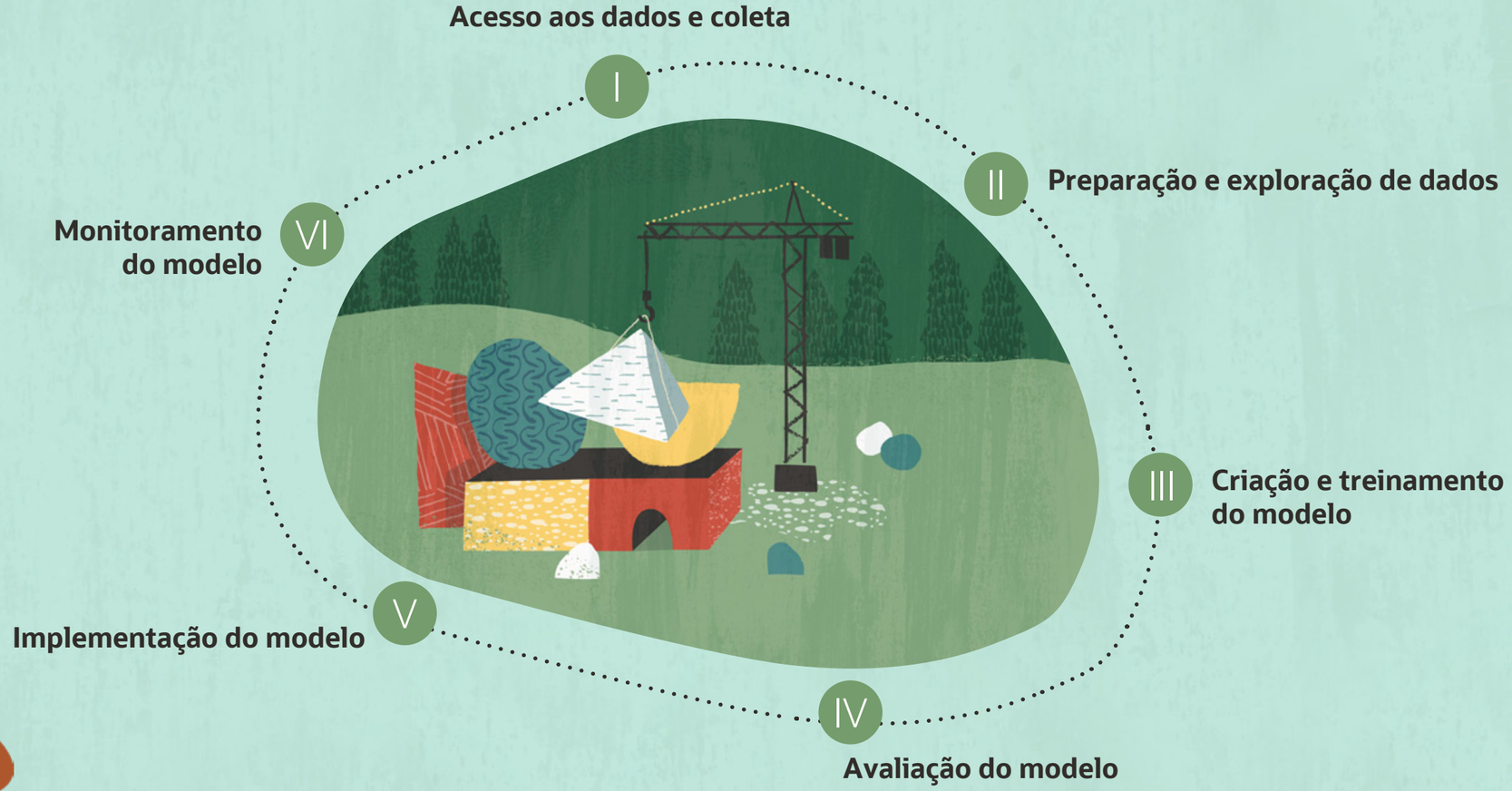


Introdução

O machine learning é uma área na qual as empresas estão cada vez mais investindo ou identificando potencial de crescimento. As empresas investem em machine learning por diversos motivos, indo desde a capacidade de aproveitar dados para obter insights sobre os clientes até a possibilidade de tornar os processos mais eficientes. Neste ebook, detalhamos como os modelos de machine learning são criados em seis etapas: acesso aos dados e coleta; preparação e exploração de dados; criação e treinamento de modelo; avaliação; implementação; e monitoramento.

A criação de um modelo de machine learning é um processo iterativo. Muitas das etapas necessárias para criar um modelo de machine learning são reiteradas e modificadas até que os cientistas de dados estejam satisfeitos com o desempenho do modelo. Esse processo requer muita exploração, visualização e experimentação de dados, pois cada etapa deve ser explorada, modificada e auditada de forma independente.

Etapas da criação de modelos de machine learning



I. Acesso aos dados e coleta

A primeira etapa de um problema de machine learning é acessar os dados. Geralmente, os cientistas de dados obtêm os dados dos problemas de negócios nos quais estão trabalhando por meio da consulta aos bancos de dados de suas empresas. Além disso, também são valiosos alguns conjuntos de dados não estruturados que não se encaixam em bancos de dados relacionais (por exemplo, logs, textos brutos, imagens, vídeos etc.). Esses conjuntos são altamente processados por meio de pipelines de extrair, transformar, carregar (ETL) escritos por engenheiros e cientistas de dados. Eles residem em um data lake ou em um banco de dados (relacional ou não). Quando os cientistas não têm os dados necessários para resolver seus problemas, eles podem obtê-los extraindo dados de sites, comprando-os de provedores de dados ou coletando os dados de pesquisas, sequência de cliques, sensores, câmeras e outras origens.

II. Preparação e exploração de dados

Depois de obter os dados, os cientistas precisam preparar os dados brutos, executar a exploração, visualização e transformação e possivelmente repetir as etapas até que esses dados estejam prontos para ser utilizados na criação de modelos. A preparação de dados é a limpeza e o processamento de dados brutos antes da análise. Antes de criar qualquer modelo de machine learning, os cientistas de dados precisam entender os dados disponíveis. Os dados brutos podem estar desorganizados, duplicados ou imprecisos. Os cientistas exploram os dados disponíveis para eles e, depois, os depuram, identificando dados corrompidos, imprecisos e incompletos e substituindo-os ou excluindo-os.

Além disso, os cientistas precisam determinar se os dados têm rótulos ou não. Por exemplo, se você tem uma série de imagens e deseja desenvolver um modelo de detecção para determinar se há um carro em uma delas, é preciso ter um conjunto de imagens com rótulos identificando se há um carro nelas e, muito provavelmente, são necessárias caixas delimitadoras ao redor dos carros nas imagens. Se as imagens não tiverem rótulos, os cientistas de dados terão que as rotular. Há ferramentas de código-fonte aberto e fornecedores comerciais que oferecem plataformas para rotulagem de dados, além disso, é possível contratar pessoas que realizam esse trabalho.

Após a depuração, os cientistas exploram as características (ou as variáveis) em seu conjunto de dados e identificam relações entre as transformações dessas características. Há várias ferramentas que os cientistas de dados podem usar para análise exploratória em bibliotecas de código-fonte aberto e plataformas de análise/ciência de dados. Nesta etapa, é útil ter uma ferramenta que realize a análise estatística do conjunto de dados e crie visualizações de dados para gerar plotagens das características.

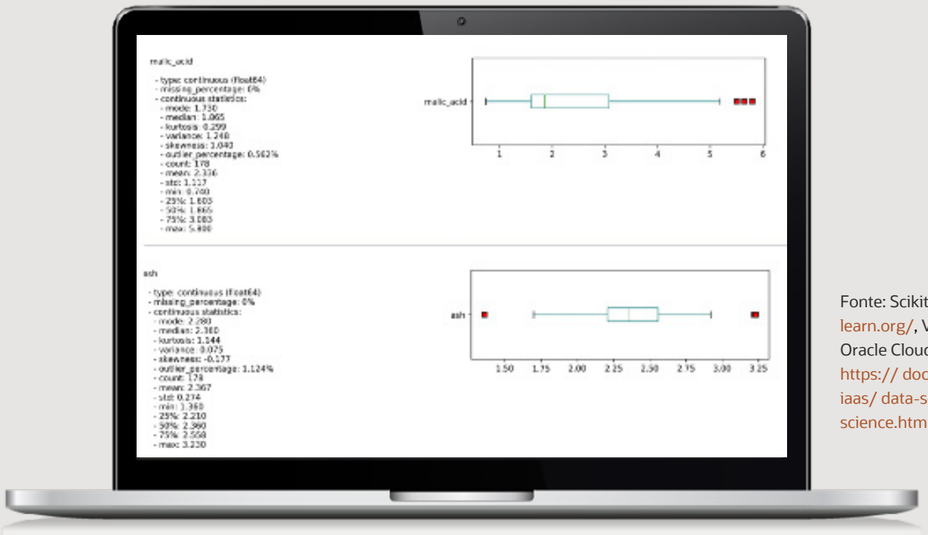
É importante verificar quais tipos de características estão presentes no conjunto de dados. Os valores de característica podem ser numéricos, como um ponto flutuante ou um número inteiro. As características de categoria têm um número finito de valores possíveis, geralmente atribuindo dados a grupos. Por exemplo, se você tiver um conjunto de dados de uma pesquisa de clientes, o sexo do respondente (masculino ou feminino) será uma característica de categoria. As características de ordem são categóricas e têm uma ordem ou escala definida. Por exemplo, a resposta de satisfação do cliente: muito satisfeito, satisfeito, indiferente, insatisfeito e muito insatisfeito tem uma ordem definida. Você pode converter essa ordenação em uma escala de números inteiros (1->5). Depois de determinar que tipos de característica existem, a próxima etapa é obter uma distribuição de valores de cada uma delas e as estatísticas individuais resumidas. Isso ajuda a responder às seguintes perguntas sobre o conjunto de dados:

- O conjunto de dados está distorcido para um intervalo de valores ou um subconjunto de categorias?
- Quais são os valores mínimo, máximo, médio, mediano e de modo da característica?
- Existem valores ausentes ou inválidos, por exemplo, "nulo"? Se sim, quantos?
- Há outliers no conjunto de dados?

Durante a etapa de exploração de dados, é útil plotar as características, bem como plotá-las em relação umas às outras para identificar padrões no conjunto de dados. Isso ajuda a determinar a necessidade de transformação de dados. Algumas perguntas que precisam ser respondidas:

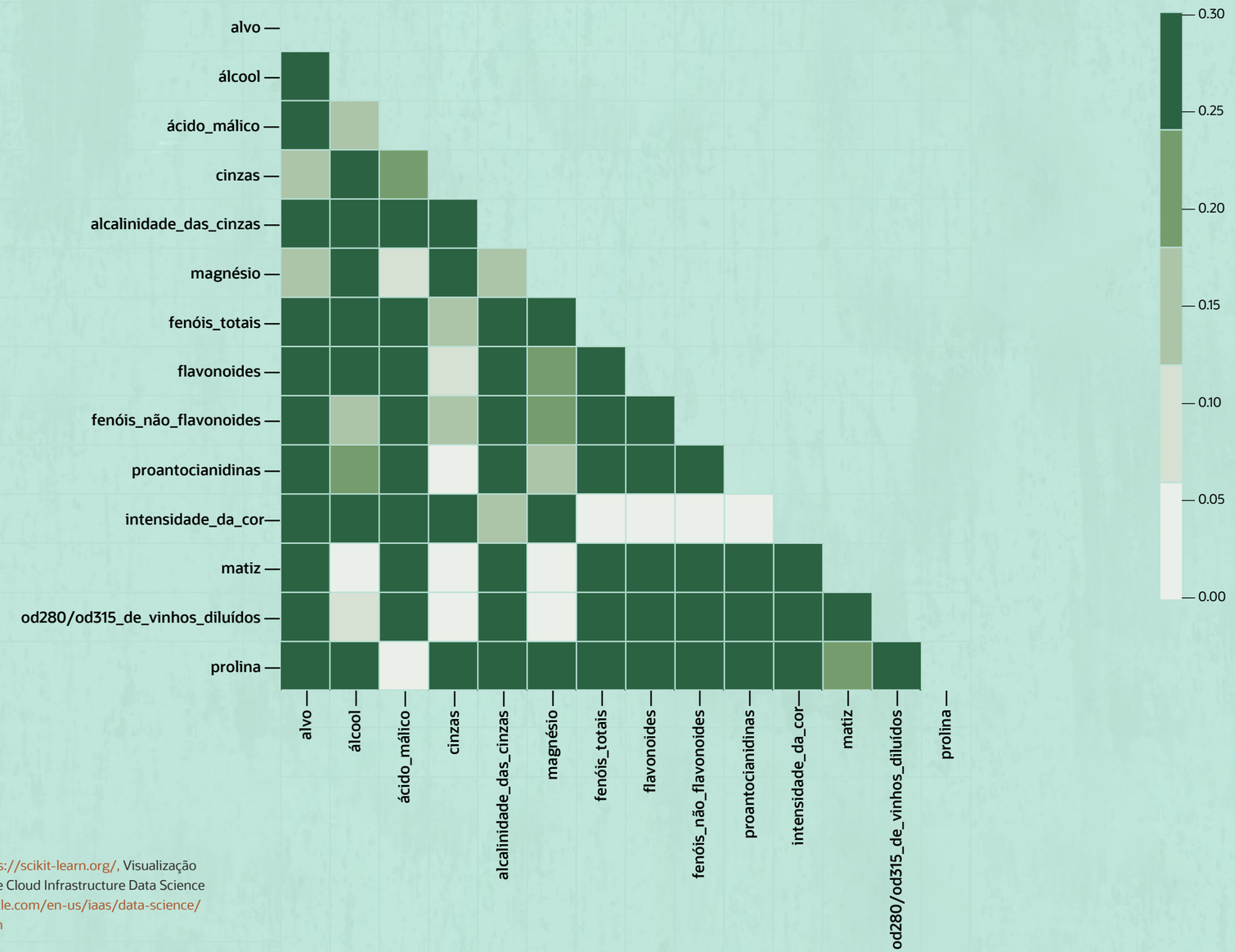
- Como você lida com valores ausentes? Você quer preencher os valores? Se sim, qual a abordagem que você pretende adotar para preencher o valor ausente? Algumas abordagens incluem considerar o valor médio, o mediano, o modo, o valor da entrada próxima e a média dos valores das entradas próximas.
- Como você lidará com outliers?
- Algumas características estão correlacionadas?
- Você precisa normalizar o conjunto de dados ou executar alguma outra transformação para redimensionar os dados (por exemplo, transformação de log)?
- Qual é a sua abordagem para "long tail" de valores categóricos? Você os usa como estão, os agrupa de alguma forma significativa ou ignora um subconjunto deles?

Estatísticas resumidas e visualização de características em um conjunto de dados de três tipos de vinho e de cada vinho.



Fonte: Scikit Learn https:// scikit-learn.org/, Visualização executada com o Oracle Cloud Infrastructure Data Science https:// docs.cloud.oracle.com/en-us/iaas/ data-science/using/data-science.htm

Mapa de calor de quanto as características estão correlacionadas umas às outras, em um conjunto de dados com três tipos de vinho e características de cada vinho.



Fonte: Scikit Learn https://scikit-learn.org/, Visualização executada com o Oracle Cloud Infrastructure Data Science https://docs.cloud.oracle.com/en-us/iaas/data-science/using/data-science.htm

Durante a etapa de exploração de dados, você consegue identificar padrões no seu conjunto para ter ideias de novas características que representariam melhor esse conjunto. Isso é conhecido como engenharia de características. Por exemplo, se você tiver um conjunto de dados de trânsito referente ao número de veículos que passam por uma interseção principal a cada hora, poderá criar uma nova característica para categorizar a hora em diferentes partes do dia, como início da manhã, meio da manhã, início da tarde, fim da tarde e noite.

Para características de categoria, geralmente é necessário aplicar o método "one-hot-encoding". Nesse método, cada categoria da característica categórica é transformada em uma característica binária. Por exemplo, vamos supor que haja um conjunto de dados de clientes e uma característica que indique o estado de origem do cliente: Washington, Óregon e Califórnia. O método "one-hot-encoding" geraria duas características binárias, uma para Washington ou não, e a outra para Óregon ou não. Pressupõe-se que, se o cliente não for de Washington nem de Óregon, ele será da Califórnia, assim sendo, não há necessidade de uma terceira característica.

III. Criação e treinamento do modelo

A criação de modelo consiste em escolher os modelos de machine learning corretos para resolver os problemas e as características que estarão neles. Na primeira etapa da criação do modelo, os cientistas de dados precisam decidir qual pode ser o modelo de machine learning adequado para resolver o problema. Há dois tipos principais de modelos de machine learning: supervisionado e não supervisionado. O aprendizado supervisionado envolve a modelagem de um conjunto de dados de entrada para uma saída ou um rótulo. A classificação e a regressão são problemas de aprendizado supervisionado. O aprendizado não supervisionado envolve a modelagem de um conjunto de dados de entrada sem rótulo. Por exemplo, a segmentação de clientes é um problema de aprendizado não supervisionado. Você não sabe a priori a qual segmento de clientes um cliente pertence. O segmento será atribuído pelo modelo.

São utilizadas classes diferentes de modelos de machine learning para resolver problemas de aprendizado supervisado e não supervisionado. Geralmente, os cientistas de dados testam vários modelos e algoritmos e geram diversos candidatos a modelos. Como não sabem qual modelo terá o melhor desempenho no conjunto de dados, eles testam vários. Durante o treinamento do modelo, um cientista de dados pode fazer a seleção de características, que é o processo de selecionar apenas um subconjunto delas como entrada para o modelo de machine learning. Benefícios de diminuir o número de variáveis de entrada: reduz o custo computacional do treinamento do modelo; torna o modelo mais generalizável; e possivelmente aprimora seu desempenho.

Durante o treinamento do modelo, o conjunto de dados é dividido em conjuntos de treinamento e de teste. O conjunto de dados de treinamento é usado para treinar o modelo, e o conjunto de dados de teste é usado para verificar o desempenho do modelo com outros dados, que ele não analisou. A avaliação do modelo será discutida em mais detalhes abaixo.

O ajuste de hiperparâmetro do modelo é uma tarefa importante no processo de treinamento. Modelos são algoritmos, e hiperparâmetros são os botões que um cientista de dados pode ajustar para melhorar o desempenho do modelo. Por exemplo, a profundidade de uma árvore de decisão é um hiperparâmetro.

Você pode optar por ter uma árvore de decisão muito profunda ou muito superficial. Isso afetará o viés e a variação do seu modelo. Viés é o erro decorrente do subajuste ou da não captura da relação entre as características e as saídas. Variação é um erro de sobreajuste, no qual o modelo funciona bem no conjunto de dados de treinamento, mas não funciona para dados não analisados. É possível automatizar parcialmente o ajuste dos hiperparâmetros de um modelo, mas os cientistas de dados devem sempre estar envolvidos no processo.

Os cientistas de dados também precisam decidir quais tipos de recurso de computação são necessários para treinar seus modelos. Você pode preparar os dados e treinar os modelos de forma local no seu computador. No entanto, dependendo do volume de dados a ser preparado e utilizado para treinar o modelo, seu computador pode não ser suficiente. Talvez você precise fazer a transição da carga de trabalho para a nuvem, onde terá acesso a uma seleção mais ampla de recursos de computação, incluindo GPUs.

Alguns modelos podem ser treinados mais rapidamente em hardware especializado (por exemplo, perceptrons de treinamento/modelos de rede neural profunda em GPUs). É possível também explorar ambientes de treinamento distribuídos que podem acelerar o processo, especialmente quando a quantidade de dados não pode se ajustar à memória da maior máquina disponível, dividindo e distribuindo os dados em várias máquinas, ou quando você deseja treinar simultaneamente vários candidatos de modelo em paralelo em máquinas separadas.

AutoML

O AutoML ganhou bastante atenção nos últimos anos devido à promessa de tornar o machine learning mais acessível para um público maior. AutoML significa machine learning automatizado. Ele automatiza o processo de seleção de características/modelos/algoritmos e o ajuste de hiperparâmetros. Esse recurso está presente em todas as principais plataformas de ciência de dados. Os usuários podem alimentar o AutoML com um conjunto de dados, e ele treinará vários modelos de machine learning, ajustará os hiperparâmetros para esses modelos e avaliará o desempenho entre eles.

O AutoML pode aumentar a produtividade dos cientistas de dados automatizando o processo de treinamento. Ele também permite que desenvolvedores e analistas de dados criem modelos de machine learning sem ajustar cada aspecto do processo de treinamento de modelo que vem com a experiência em ciência de dados. A maioria dos recursos de AutoML suporta dados tabulares para problemas de classificação e regressão, enquanto outros têm ofertas mais avançadas que suportam imagens e dados de texto, bem como previsão de séries temporais.

O inconveniente do AutoML, ou de qualquer modelo complicado, é que pode parecer uma solução caixa preta, que torna difícil para os usuários entender como os modelos chegam às previsões. Os usuários devem buscar a oferta de explicabilidade do modelo do sistema AutoML para ver qual recurso existe para ajudar os usuários a interpretar os modelos, bem como entender como os modelos selecionados chegam às previsões.

As explicações de modelo geralmente são globais e locais. A explicação global é entender o comportamento geral de um modelo de machine learning como um todo. Isso inclui explicar a importância de contribuição de cada característica para as previsões do modelo. A explicação local ajuda a entender por que o modelo de machine learning fez uma previsão específica para uma amostra de dados. Por exemplo, por que um algoritmo de detecção de fraude previu uma transação específica como fraudulenta?

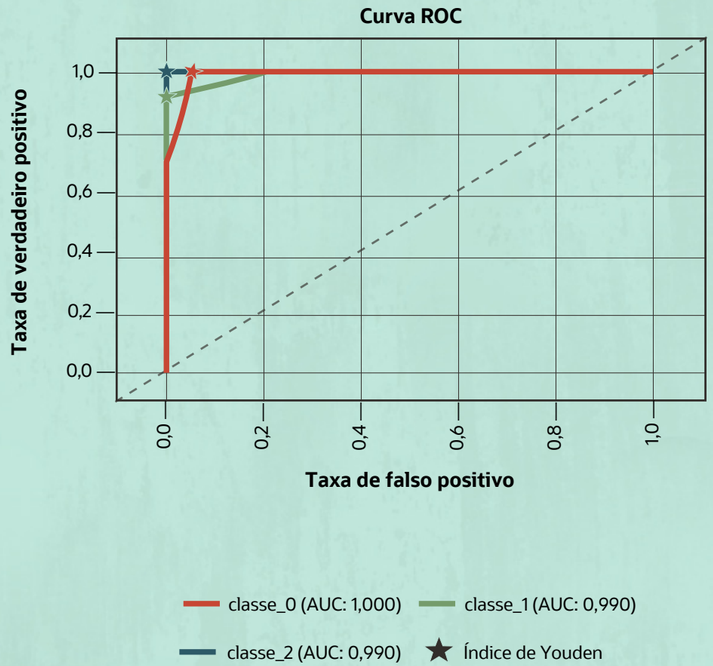
IV. Avaliação do modelo

Há muitas ferramentas de código-fonte aberto que ajudam os cientistas de dados a calcular e visualizar as métricas usadas para avaliar modelos de machine learning (por exemplo, curva AUC-ROC, gráficos de ganho e lift). Ao avaliar modelos de machine learning, os cientistas de dados precisam decidir quais métricas são importantes para o problema de negócios que estão tentando resolver.

Para problemas de classificação, é possível usar precisão para avaliação de modelo, mas às vezes pode não ser a escolha de métrica ideal. Se um problema envolve detectar se alguém tem uma doença rara, uma métrica melhor pode ser o número de pessoas com a doença que são diagnosticadas com precisão dividido por todas as pessoas com a doença. Nesse caso, seria mais útil olhar para uma matriz de confusão que mostra o número de verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos, e calcular precisão e revocação. Para problemas de regressão, você pode usar métricas como erro médio quadrático, erro médio absoluto ou calcular o coeficiente de determinação r2. Para problemas não supervisionados, um conjunto de clusters com alta coesão dentro dos clusters e separação entre eles é considerado ideal. Isso pode ser medido com métricas como índice Silhouette e o coeficiente de Calinski-Harabasz.

Matriz de confusão de classificação multiclasse para os resultados de um modelo de floresta aleatório sobre a previsão do tipo de vinho com base nas características de um vinho, a partir de um conjunto de dados com três tipos de vinho e as características de cada um.

Curva ROC de classificação multiclasse criada para os resultados de um modelo de floresta aleatório sobre a previsão do tipo de vinho a partir de um conjunto de dados com três tipos de vinho e as características de cada um.



Fonte: Scikit Learn Library https://scikit-learn.org/, Visualização executada com o Oracle Cloud Infrastructure Data Science https://docs.cloud.oracle.com/en-us/iaas/data-science/using/data-science.htm

V. Implementação do modelo

Depois que os processos de treinamento e avaliação do modelo são concluídos, os melhores candidatos a modelo são salvos. Os modelos geralmente são salvos no formato Pickle, ONNX e PMML. Dependendo dos objetivos, os cientistas de dados podem trabalhar em um problema de machine learning para prova de conceito, experimentação ou implementação na produção. A implementação do modelo está consumindo as previsões feitas pelo modelo de machine learning de alguma forma. Provavelmente, o pipeline de transformações de dados também precisa ser implantado. Geralmente, os cientistas de dados trabalham com engenheiros na implementação do modelo.

Dependendo de como pretende consumir as previsões, você pode implantá-la para consumo em lote ou em tempo real. Para o consumo em lote, as previsões podem ser programadas (por exemplo, a cada hora, todos os dias). Posteriormente, as previsões podem ser armazenadas em um banco de dados e consumidas por outras aplicações. Em geral, a quantidade de dados que você processa é maior do que a previsão em tempo real. Um caso de uso seria se você executasse um site de eCommerce e quisesse enviar um e-mail semanal aos clientes sobre produtos recomendados para eles com base em compras anteriores. Os modelos de machine learning podem ser programados para serem executados com antecedência.

Para consumo em tempo real, um acionador iniciaria o processo de usar o modelo persistente para atender a uma previsão. Por exemplo, decidir se uma transação é fraudulenta ou não quando o pagamento é iniciado requer previsão em tempo real. Você precisa considerar a rapidez com que precisa atender às previsões (milissegundos, segundos?), o volume de demanda do serviço e o tamanho dos dados nos quais executará previsões. É importante minimizar a latência de atendimento à previsão. Você pode diminuir a latência de atendimento utilizando um modelo menor ou aceleradores (como GPU) e melhorando como as características relacionadas à entidade são recuperadas para previsão em tempo real (por exemplo, se você estiver recomendando produtos para um usuário durante a navegação em um site, melhorias na forma como são extraídas as informações sobre compras anteriores do usuário podem diminuir a latência).

Há diferentes ferramentas e ofertas de plataforma em nuvem para implementação de modelos, como plataformas Functions-as-a-service (FaaS), implementação totalmente gerenciada de modelos como pontos de extremidade HTTP, DIY com Flask ou Django em uma plataforma de orquestração de contêiner, como K8, Docker Swarm e outros.

VI. Monitoramento do modelo

O monitoramento do modelo é uma etapa desafiadora que, às vezes, é esquecida por organizações sem iniciativas maduras de machine learning e ciência de dados. O novo treinamento e a reimplementação do modelo requerem tempo da equipe de ciência e engenharia de dados e recursos de computação. O monitoramento do modelo ajuda a equipe a decidir se e quando é necessário treinar o modelo novamente e reimplantá-lo. O monitoramento do modelo pode ser dividido em dois componentes: monitoramento de divergência/estatístico do desempenho do modelo e monitoramento de operações.

Após a implementação dos modelos, as métricas pelas quais os modelos foram medidos e treinados diminuem em produção. Isso ocorre porque os dados não são estáticos. A dinâmica pode se manifestar de várias maneiras: as características nos dados de produção podem assumir valores fora do intervalo do conjunto de dados de treinamento; pode haver uma lenta divergência na distribuição dos valores etc.

Devido à degradação dos modelos, a qualidade deles precisa ser monitorada para decidir se e quando treiná-lo novamente e reimplantá-lo. Às vezes, não é possível obter imediatamente a precisão de previsão dos dados reais que vão para um sistema de produção. Por exemplo, pode levar algum tempo para que você possa decidir se um modelo de previsão de abandono ou um modelo de detecção de fraude forneceu uma previsão precisa. No entanto, é possível analisar as estatísticas e a distribuição dos dados de treinamento em relação aos dados reais, e também comparar a distribuição das previsões do modelo com os dados de treinamento e reais. Por exemplo, se você estiver trabalhando com um modelo de abandono de cliente, poderá comparar as características dos clientes usados para treinar seu modelo com aquelas dos clientes no sistema de produção. Além disso, você pode observar a porcentagem de clientes com previsão de abandono na amostra de treinamento em comparação com a produção real.

O monitoramento de operações do sistema de machine learning exigirá uma parceria entre os cientistas de dados e a equipe de engenharia. A latência de atendimento, o uso de memória/CPU, o ganho e a confiabilidade do sistema são alguns exemplos do que deve ser monitorado. É necessário configurar logs e métricas para acompanhamento e monitoramento. Os logs contêm registros de eventos, juntamente com o horário em que ocorreram. Eles podem ser usados para investigar incidentes específicos e descobrir a causa do incidente. O Kibana é uma ferramenta de código-fonte aberto usada para pesquisar e visualizar logs. As métricas medem o uso e o comportamento do sistema de machine learning. O Prometheus e o Grafana são ferramentas para monitorar métricas.

Conclusão

Esperamos que este livro tenha sido um guia útil com as etapas necessárias para criar um modelo de machine learning. Lembre-se de que o machine learning é um processo extremamente iterativo, e as etapas aqui descritas serão reiteradas e aprimoradas diversas vezes.

Estão disponíveis muitos recursos mais detalhados sobre cada uma das etapas abordadas neste livro, e você poderá aprender mais sobre elas conforme for tomando decisões sobre a estratégia de ciência de dados da sua empresa. Caso esteja pronto para "colocar a mão na massa", a Oracle oferece laboratórios práticos para que você crie seus próprios modelos de ciência de dados.

Oracle corporation

Sede mundial
500 Oracle Parkway, Redwood Shores, CA 94065, USA

Consultas globais
Telefone + 1.650.506.7000 + 1.800.ORACLE1
Fax + 1.650.506.7200

oracle.com

Fale conosco

Ligue para +1.800.ORACLE1 ou acesse oracle.com. Se não estiver na América do Norte, encontre seu escritório local em: oracle.com/contact.

blogs.oracle.com/oracle

facebook.com/oracle

twitter.com/oracle

Autora

Wendy Yip, cientista de dados.