Relatório <10> - <Prática: Lidando com Dados do Mundo Real (II) (LEIA A DESCRIÇÃO COMPLETA)>

<Jonas Correia>

1. K-Nearest-Neighbors: Concepts

O algoritmo K-Nearest Neighbors (KNN), uma técnica simples de aprendizado supervisionado e mineração de dados. KNN é utilizado para classificar novos pontos de dados com base nos pontos de dados já classificados. A ideia central é que, ao receber um novo dado, o algoritmo verifica os K vizinhos mais próximos, medidos por uma métrica de distância (como em um gráfico de dispersão), e utiliza o "voto" desses vizinhos para determinar a classificação do novo ponto.

Por exemplo, se tivermos dados de filmes, onde quadrados azuis representam filmes de ficção científica e triângulos vermelhos representam dramas, e precisarmos classificar um novo filme, o algoritmo verifica os K vizinhos mais próximos e faz a decisão com base nas classificações dos vizinhos. Se K = 3, por exemplo, e houver 2 filmes dramáticos e 1 de ficção científica, o novo filme seria classificado como drama.

O valor de K é importante: ele deve ser pequeno o suficiente para evitar incluir vizinhos irrelevantes, mas grande o suficiente para garantir uma amostra representativa. O texto enfatiza que a escolha de K muitas vezes requer experimentação.

No exemplo prático, a ideia é aplicar KNN para encontrar filmes semelhantes entre si, usando metadados como classificações e gêneros. Com isso, seria possível criar recomendações de filmes, como os "clientes que também assistiram" da Amazon, e prever a classificação de novos filmes com base nos vizinhos mais próximos.

2. [Activity] Using KNN to predict a rating for a movie

3. Dimensionality Reduction; Principal Component Analysis (PCA)

Redução de dimensionalidade, uma técnica essencial para lidar com dados complexos e de alta dimensionalidade, que pode ser comparada a tentar capturar a essência de algo vasto em uma forma mais simples e compreensível, sem perder sua essência. Através de métodos como a Análise de Componentes Principais (PCA) e a Decomposição de Valor Singular (SVD), é possível comprimir e projetar dados em um espaço de menor dimensão, preservando ao máximo sua variação original.

A PCA, por exemplo, funciona como um filtro matemático, encontrando os hiperplanos que melhor representam os dados e projetando-os em uma nova estrutura mais manejável. Isso é essencial não só para facilitar a visualização dos dados, mas também para aplicações práticas, como compressão de imagens e reconhecimento facial. A redução da dimensionalidade é como desvelar as partes mais importantes de um todo, destacando apenas aquilo que é indispensável.

O exemplo da flor Iris é utilizado para ilustrar a aplicação prática dessa técnica, onde quatro características (dimensões) são resumidas em duas, sem sacrificar as nuances necessárias para classificação. No entanto, por mais intrincada que seja a matemática subjacente, a verdadeira beleza da redução de dimensionalidade reside na simplicidade de sua execução, com ferramentas como o scikit-learn, que tornam sua implementação acessível em poucas linhas de código. Assim, o processo revela um equilíbrio entre sofisticação teórica e praticidade funcional.

4. [Activity] PCA Example with the Iris data set

5. Data Warehousing Overview: ETL and ELT

Ormazenamento de dados e técnicas de processamento de grandes volumes de informações, como ETL (Extract, Transform, Load) e ELT (Extract, Load, Transform). O objetivo principal é fornecer uma visão geral dessas abordagens no contexto de data warehouses e big data.

Um armazém de dados (data warehouse) é descrito como um banco de dados gigante que integra informações de diferentes fontes, facilitando a análise de dados em grande escala. Empresas podem vincular dados de compras, logs de navegação e sistemas de atendimento ao cliente para obter insights valiosos sobre o comportamento dos consumidores. O desafio está na normalização dos dados, garantindo que campos de diferentes fontes sejam comparáveis, além de lidar com dados ausentes, corrompidos ou de baixa qualidade.

A abordagem tradicional de ETL envolve três fases: extrair dados de diferentes sistemas, transformá-los em um formato estruturado e, então, carregá-los no data warehouse. No entanto, à medida que o volume de dados cresce, a fase de transformação pode se tornar um gargalo, especialmente para grandes empresas que lidam com quantidades massivas de dados, como a Amazon ou o Google.

O conceito mais moderno, ELT, inverte esse processo. Ele sugere carregar primeiro os dados brutos no sistema e depois usar a capacidade de processamento de clusters distribuídos (como Hadoop, Spark e MapReduce) para realizar a transformação diretamente no data warehouse. Isso é viabilizado pelo advento de tecnologias de big data e computação em nuvem, que permitem uma escalabilidade muito maior.

6. Reinforcement Learning

O conceito de aprendizado por reforço, como descrito, é uma técnica poderosa para a criação de agentes inteligentes, como no exemplo do Pac-Man. A ideia central é que o agente, como o Pac-Man, explora seu ambiente (um labirinto) e aprende com as consequências de suas ações, seja por obter recompensas (como comer uma pílula de poder) ou punições (como ser comido por um fantasma).

7. [Activity] Reinforcement Learning & Q-Learning with Gym

8. Understanding a Confusion Matrix

A matriz de confusão é uma ferramenta poderosa usada para avaliar o desempenho de um modelo de classificação. Ela nos permite visualizar a relação entre as previsões feitas pelo modelo e os resultados reais, fornecendo informações sobre onde o modelo acerta ou erra.

Componentes da Matriz de Confusão:

Verdadeiro Positivo (TP): Quando o modelo prevê "sim" e o valor real também é "sim".

Falso Positivo (FP): Quando o modelo prevê "sim", mas o valor real é "não" (também chamado de erro tipo I).

Falso Negativo (FN): Quando o modelo prevê "não", mas o valor real é "sim" (também chamado de erro tipo II).

Verdadeiro Negativo (TN): Quando o modelo prevê "não" e o valor real também é "não".

Exemplo:

Imaginemos um modelo que classifica imagens como contendo um gato ou não.

TP (Verdadeiro Positivo): O modelo prevê que há um gato, e realmente há um gato (50 vezes).

FP (Falso Positivo): O modelo prevê que há um gato, mas na verdade não há (5 vezes).

FN (Falso Negativo): O modelo prevê que não há um gato, mas na verdade há (10 vezes).

TN (Verdadeiro Negativo): O modelo prevê que não há um gato, e de fato não há (100 vezes).

A matriz de confusão é organizada em uma tabela com os valores previstos (sim ou não) em uma dimensão e os valores reais na outra. Idealmente, queremos que a maioria dos valores esteja ao longo da diagonal principal da matriz (que corresponde a TP e TN), enquanto os FP e FN devem ser valores baixos.

A precisão do modelo, que pode ser medida como o número de previsões corretas sobre o total, nem sempre reflete o verdadeiro desempenho em contextos como a detecção de doenças raras. Um modelo pode ter alta precisão apenas por adivinhar que ninguém tem a doença, mas isso não é útil. A matriz de confusão ajuda a identificar tais cenários.

Às vezes, veremos versões com classes múltiplas (como reconhecimento de dígitos de 0 a 9) ou representações visuais como mapas de calor, onde cores mais escuras indicam um maior número de acertos ou erros.

A matriz de confusão nos permite entender onde o modelo acerta e onde erra, fornecendo insights mais profundos do que apenas a precisão isolada. É essencial em contextos onde os erros têm consequências diferentes, como diagnósticos médicos ou detecção de fraudes.

9. Measuring Classifiers (Precision, Recall, F1, ROC, AUC)

A matriz de confusão é uma ferramenta fundamental para avaliar o desempenho de modelos de classificação, sendo possível derivar várias métricas importantes a partir dela. O *recall* (ou sensibilidade) mede a capacidade do modelo de identificar corretamente instâncias positivas, calculado como a razão entre verdadeiros positivos e a soma dos verdadeiros positivos com os falsos negativos. É especialmente útil quando o foco está em minimizar falsos negativos, como em cenários de detecção de fraudes ou diagnóstico médico.

A precisão, por outro lado, avalia a proporção de instâncias corretamente classificadas como positivas em relação ao total de classificações positivas, e é mais relevante quando se busca minimizar falsos positivos. Um exemplo clássico seria um teste de drogas, onde classificar alguém erroneamente como positivo pode ter consequências graves.

A pontuação F1 é a média harmônica entre precisão e recall, útil quando é necessário equilibrar ambos, sem priorizar um sobre o outro. Ela oferece uma visão geral do desempenho do modelo quando tanto a precisão quanto o recall são importantes. A especificidade, ou taxa de verdadeiros negativos, é outra métrica derivada da matriz de confusão que se concentra em quantos dos negativos foram corretamente identificados, sendo relevante em situações onde falsos positivos devem ser minimizados.

A curva ROC (Receiver Operating Characteristic) e a AUC (Área Sob a Curva) medem o desempenho do modelo em diferentes limiares de decisão. A curva ROC traça o recall versus a taxa de falsos positivos, e a AUC oferece uma métrica numérica que varia entre 0.5 (classificação aleatória) e 1 (classificação perfeita). Juntas, essas métricas fornecem uma visão abrangente do desempenho de classificadores e ajudam na comparação entre diferentes modelos.

10. Bias/Variance Tradeoff

O trade-off entre viés e variância é um conceito central ao lidar com dados do mundo real, refletindo o equilíbrio entre ajuste excessivo (overfitting) e ajuste insuficiente (underfitting) em modelos preditivos. O viés refere-se ao quanto as previsões de um modelo se afastam dos valores corretos, ou seja, se o modelo consistentemente erra em uma direção específica. Já a variância mede o quanto as previsões estão espalhadas, indicando se elas variam muito de uma instância para outra. Em um modelo com baixo viés, as previsões estão, em média, próximas do valor correto, enquanto um modelo com alta variância apresenta previsões dispersas.

O objetivo é encontrar um equilíbrio entre esses dois fatores, pois um modelo com baixo viés pode ter alta variância, sendo sensível ao ruído dos dados, enquanto um modelo com baixa variância pode ter alto viés, não capturando bem a complexidade dos dados. Por exemplo, em K-Nearest Neighbors (K-NN), aumentar o valor de K reduz a variância ao considerar mais vizinhos, mas pode introduzir viés ao incorporar informações menos relevantes. Modelos complexos, como árvores de decisão, tendem a ter alta variância e baixo viés, enquanto modelos mais simples apresentam o oposto. Florestas aleatórias equilibram esse trade-off, reduzindo a variância ao combinar várias árvores.

11. [Activity] K-Fold Cross-Validation to avoid overfitting

12. Data Cleaning and Normalization

A limpeza de dados é uma das etapas mais importantes e desafiadoras no processo de ciência de dados, frequentemente demandando mais tempo do que a análise em si. Dados brutos são muitas vezes sujos e poluídos, o que pode distorcer significativamente os resultados de um modelo se não forem adequadamente tratados. Isso torna a limpeza e a preparação dos dados essenciais para garantir a qualidade dos resultados.

Existem diversos problemas que podem ocorrer com os dados, como outliers (valores fora do padrão esperado), dados ausentes, dados maliciosos (exemplos incluem ataques de sistemas automatizados), dados errôneos (causados por bugs ou falhas de software), dados irrelevantes, dados inconsistentes (onde informações são representadas de diferentes maneiras), e formatação inadequada (como diferenças regionais em datas e números de telefone). Cada um desses problemas precisa ser resolvido para evitar que interfiram no desempenho do modelo e conduzam a decisões erradas.

Um modelo bem ajustado com dados limpos pode muitas vezes superar modelos mais complexos que utilizam dados sujos. Além disso, é crucial questionar constantemente os resultados, mesmo que pareçam corretos, para evitar viéses não intencionais e garantir a confiabilidade das análises. Portanto, a qualidade e a quantidade dos dados têm um impacto direto na eficácia dos algoritmos e nos insights gerados.

13. [Activity] Cleaning web log data

14. Normalizing numerical data

A normalização e a escalabilidade dos dados são fundamentais ao preparar entradas para algoritmos de aprendizado de máquina. É crucial garantir que diferentes atributos numéricos estejam na mesma escala e sejam comparáveis, ajudando a evitar que atributos com valores mais altos dominem o modelo e introduzam vieses.

Ao trabalhar com atributos de diferentes escalas, como idade variando de 0 a 100 anos e renda de 0 a bilhões, a normalização se torna essencial, especialmente em modelos que não lidam bem com escalas variadas, como aqueles que utilizam distâncias, como KNN ou SVM. Ferramentas como Scikit-Learn oferecem métodos automáticos para normalizar e escalar dados. Por exemplo, o PCA (Análise de Componentes Principais) possui opções para normalizar os dados automaticamente.

Além disso, dados categóricos, como respostas sim/não, devem ser convertidos em formatos numéricos, como 0 e 1, para serem utilizados em modelos. Após a normalização, é importante reverter o processo antes de apresentar os resultados, garantindo que sejam interpretáveis na escala original.

15. [Activity] Detecting outliers

16. Feature Engineering and the Curse of Dimensionality

Engenharia de características é o processo de aplicar o conhecimento que você tem sobre os dados para selecionar, criar ou transformar características que serão usadas para treinar seu modelo. Essas características são os atributos do conjunto de dados, como idade, altura, peso, entre outros, que podem ou não ser relevantes para a previsão que você deseja realizar.

Por exemplo, se o objetivo é prever a renda das pessoas, características como idade, altura, peso, endereço e tipo de carro podem ser consideradas. O desafio é identificar quais dessas características são realmente importantes e se as transformações são necessárias, como normalização ou codificação de dados ausentes.

A engenharia de características envolve também a criação de novas características a partir das existentes, como aplicar operações matemáticas para capturar melhor as tendências dos dados. A prática não se resume apenas em seguir um conjunto de passos, mas envolve conhecimento do domínio e experimentação.

Um conceito chave na engenharia de características é a "maldição da dimensionalidade". Isso se refere ao problema de que, à medida que você adiciona mais características, o espaço em que os dados existem se torna mais esparso, dificultando a identificação de padrões. Por exemplo, se você representar dados apenas pela idade, terá um vetor unidimensional. Ao adicionar características como altura e renda, o espaço se torna tridimensional, e assim por diante. Quanto mais dimensões você tiver, mais difícil se torna encontrar soluções eficazes.

Para mitigar a maldição da dimensionalidade, é vital selecionar apenas as características mais relevantes. Além de melhorar a eficiência do modelo, isso também facilita a construção de algoritmos, como redes neurais, que se tornariam extremamente complexos com muitas entradas. A experiência e o

senso comum são essenciais nesse processo, permitindo que você identifique quais características ajudam ou prejudicam a performance do modelo.

Existem métodos, como a Análise de Componentes Principais (PCA) e o agrupamento K-Means, que ajudam a reduzir a dimensionalidade. O PCA, por exemplo, destila múltiplas características em um número menor de dimensões, preservando a informação essencial. Ambas as técnicas são não supervisionadas, permitindo que você as aplique sem precisar rotular os dados.

17. Imputation Techniques for Missing Data

A engenharia de características é o processo de seleção, transformação e criação de atributos a partir dos dados de treinamento, visando melhorar a eficácia de um modelo. A escolha e a transformação adequadas de características são cruciais para prever resultados, e esse processo pode ajudar a reduzir a maldição da dimensionalidade, que torna a análise mais complexa quando há muitos atributos.

No que diz respeito à imputação de dados ausentes, é comum encontrar valores faltantes nos dados do mundo real. Uma abordagem simples é a substituição média, onde os valores ausentes são substituídos pela média da coluna correspondente. Essa técnica é rápida, mas pode ser insatisfatória em presença de outliers ou correlações importantes entre características. Uma alternativa é a substituição pela mediana, que pode ser mais adequada em conjuntos de dados com valores extremos.

Outra técnica é o KNN (K-Nearest Neighbors), que consiste em encontrar as linhas mais semelhantes que possuem dados completos e usar a média desses valores para imputar os ausentes. Modelos de aprendizado de máquina também podem ser utilizados para prever os valores ausentes com base em outras características, utilizando redes neurais ou regressões. MICE, ou Imputação Múltipla por Equações Encadeadas, é uma técnica avançada que considera a incerteza em imputações.

No entanto, a melhor solução para lidar com dados ausentes muitas vezes é simplesmente obter mais dados. Coletar informações adicionais pode ser mais eficaz do que confiar apenas em técnicas de imputação, desde que se evite viés ao remover linhas com dados faltantes. Em suma, um entendimento sólido e prático dessas técnicas é fundamental para aumentar a qualidade e a precisão dos modelos de aprendizado de máquinas.

18. Handling Unbalanced Data: Oversampling, Undersampling, and SMOTE

O desafio do manuseio de dados desequilibrados na engenharia de características, especialmente em contextos como a detecção de fraudes, onde a maioria dos dados são negativos (não fraudulentos) e apenas uma pequena fração representa casos positivos (fraudulentos). Essa discrepância pode levar um modelo a prever sempre a classe majoritária, resultando em alta precisão aparente, mas sem eficácia real na detecção dos casos positivos.

Uma forma de abordar esse problema é por meio da sobreamostragem, que envolve a duplicação de amostras da classe minoritária para aumentar sua representação no conjunto de dados. Embora essa abordagem possa ser útil, ela também tem suas limitações e pode não ser a melhor solução, especialmente se resultar em perda de informações valiosas.

Outra alternativa é a subamostragem, que consiste em remover casos da classe majoritária. No entanto, essa abordagem geralmente não é recomendada, a menos que o volume de dados seja tão grande que justifique a exclusão de informações. Em vez disso, buscar mais poder computacional é uma solução mais eficaz.

Uma técnica superior a ambas é o SMOTE (Synthetic Minority Over-sampling Technique), que gera amostras artificiais da classe minoritária utilizando algoritmos de vizinhos mais próximos (KNN). Essa abordagem cria novos pontos de dados a partir de médias de amostras existentes, o que pode melhorar a performance do modelo ao preservar a estrutura dos dados.

Ajustar os limites de decisão durante a inferência é outra estratégia. Ao definir um limiar de probabilidade para classificar uma transação como fraudulenta, é possível controlar a taxa de falsos positivos e negativos. Aumentar o limiar pode reduzir falsos positivos, mas pode também resultar em mais falsos negativos, o que pode ser problemático em contextos onde é preferível perder alguns casos de fraude do que incomodar clientes com alerta falsos.

19. Binning, Transforming, Encoding, Scaling, and Shuffling

A fiação, ou binning, consiste em transformar dados numéricos em categóricos, agrupando valores em faixas. Por exemplo, as idades podem ser agrupadas em intervalos como 20-29 ou 30-39. Essa abordagem ajuda a lidar com incertezas nas medições e simplifica a modelagem ao reduzir a precisão dos dados. O quantile binning é uma forma de fiação que assegura que cada categoria tenha o mesmo número de amostras, melhorando a distribuição dos dados em cada grupo.

Transformações de dados, como a aplicação de logaritmos ou quadrados, são utilizadas para linearizar dados que apresentam padrões não lineares, facilitando a identificação de tendências pelos modelos. Um exemplo disso é o YouTube, que utiliza essas transformações para melhorar suas recomendações.

A codificação, especialmente a codificação one-hot, cria variáveis indicadoras para representar categorias. No caso de números de 0 a 9, por exemplo, são criadas 10 variáveis, onde apenas uma será "1" e as outras serão "0". Essa técnica é essencial para transformar dados categóricos em um formato que algoritmos de aprendizado de máquina, especialmente redes neurais, consigam processar.

O escalonamento e normalização são práticas que ajustam as magnitudes dos dados, assegurando que todas as características tenham

pesos comparáveis. Por exemplo, normalizar idades e rendas evita que características de maior magnitude tenham um impacto desproporcional nas previsões, melhorando a desempenho do modelo.

O embaralhamento envolve aleatorizar a ordem dos dados de treinamento para evitar padrões indesejados que influenciem o aprendizado. Isso ajuda a garantir que o modelo aprenda de forma generalizada, em vez de se adaptar a uma ordem específica dos dados. Essas técnicas são fundamentais para otimizar como os dados são utilizados no treinamento de modelos de aprendizado de máquina, maximizando a eficácia das previsões.