Victor Resende - Desafio de aprendizado de máquina

Alelo

Introdução

Esta é uma pequena série de exercícios para avaliar seu conhecimento em aprendizado de máquina. Responda às perguntas detalhando as etapas executadas para resolver cada tarefa. Todas as perguntas são simples, mas esta é sua chance de nos mostrar seus conhecimentos técnicos.

Também esperamos receber seu código com sua solução e análise, incluindo quaisquer etapas de pré-processamento. Sinta-se à vontade para usar a linguagem de programação com a qual você se sentir mais confortável, mas forneça-nos instruções para executar seu código. Nem é preciso dizer, mas você não precisa codificar os algoritmos do zero. Você pode usar qualquer biblioteca disponível, como o scikit-learn. Lembre-se de que a organização e a legibilidade do seu código também serão avaliadas.

Queremos que este desafio (e esperamos que seu trabalho conosco) seja divertido, então selecionamos um conjunto de dados de super-heróis para você jogar! Você pode baixar os dados do Kaggle em https://www.kaggle.com/claudiodavi/superhero-set/data. É composto por dois arquivos csv, super hero powers.csv e heroes information.csv, e você usará os dois.

Por fim, projetamos esse desafio para durar no máximo três dias. Obviamente, não estamos cronometrando você, mas você não deve demorar muito mais para terminar este exercício. Lembre-se de que não estamos procurando os modelos de melhor desempenho, pois não é uma competição e principalmente avaliaremos sua abordagem para resolver essas tarefas. Portanto, invista seu tempo detalhando e explicando suas soluções, em vez de tentar o melhor modelo. Além disso, observe que não estamos definindo nenhuma divisão de dados ou métricas de avaliação. Isso depende inteiramente de você e será avaliado.

Se tiver algum problema com o exercício, não hesite em nos contactar, mas não iremos comentar as suas soluções nem aprofundar os detalhes técnicos das questões.

Clustering

Questão 1

Primeiro, queremos agrupar nossos super-heróis de acordo com seus poderes e informações. Execute um método de cluster não supervisionado usando o número de clusters que você julgar mais apropriado.

1. Qual algoritmo você escolheu e por quê?

Foi escolhido o algoritmo **K-Means** para nossas tarefas de clusterização devido à sua eficácia e simplicidade na formação de grupos com base na proximidade aos centroides. Após uma análise cuidadosa de várias alternativas, como DBSCAN e HDBSCAN, optamos por descartar esses métodos por motivos específicos relacionados à natureza dos nossos dados e aos objetivos da análise.

O DBSCAN, que é bem adaptado para identificar clusters de forma arbitrária e lidar com outliers, foi descartado principalmente porque nossos dados tendem a ser distribuídos de maneira

esférica. Assim, a necessidade de avaliar a densidade para definir clusters não se mostrou uma vantagem nesse contexto, tornando o DBSCAN menos apropriado.

Por outro lado, o HDBSCAN, conhecido por sua capacidade de lidar com dados que apresentam uma estrutura hierárquica e variada densidade, também foi considerado inadequado. Nosso conjunto de dados não demonstrou padrões hierárquicos claros que justificassem o uso desse método mais complexo, que também possui maior exigência computacional em comparação com o K-Means.

2. Quais recursos você usou e por quê? Explique qualquer pré-processamento ou engenharia de recursos (seleção) que você executou.

Devido à grande dimensionalidade dos dados de poderes, cerca de 168 colunas, foi necessária a implementação de técnicas de redução de dimensionalidade para simplificar o modelo sem perder informações cruciais, visando um modelo mais parcimonioso. Para isso, foi escolhido a técnica **PCA** que faz um arranjo linear entre as variáveis originais. Para a escolho dos componentes principais, decidiu-se pela quantidade que representasse 80% (variância explicada) da informação dos dados, o que resultou em 2 componentes.

Além disso, devido ao KMEans necessitar previamente a quantidade de clusters, foi realizada uma análise do **método da silhueta**, que retorna a quantidade ideal de cluster. O índice de silhueta varia de -1 a 1, onde valores próximos a 1 indicam clusters muito homogêneos e valores próximos a -1 indicam clusters heterogêneos. Este método nos ajudou a identificar o número de clusters que melhor segmenta os dados, garantindo que os grupos formados sejam internamente coesos e claramente distintos entre si.

Questão 2

Um dos desafios do clustering é definir o número certo de clusters. Como você escolheu esse número? Como você avalia a qualidade dos clusters finais?

O número de clusters foi determinado através do método da silhueta, o qual varia de -1 a 1, onde valores próximos a 1 indicam clusters muito homogêneos e valores próximos a -1 indicam clusters heterogêneos. Foi utilizada esta medida para identificar o número ótimo de clusters, escolhendo aquele que maximiza o índice de silhueta médio, indicando um equilíbrio ideal entre coesão interna e separação entre os clusters.

Para avaliar a qualidade dos clusters após a formação, empregamos a técnica de visualização t-Distributed Stochastic Neighbor Embedding (t-SNE), que nos permite observar a disposição dos clusters em um espaço bidimensional. Esta visualização nos ajudou a verificar intuitivamente se os clusters formados são distintos e agrupados de maneira compacta, o que é um indicativo de boa qualidade de clustering.

Identificando os bandidos

Nesta seção, lidaremos com o problema de aprendizagem supervisionada. Mais concretamente, iremos formular uma tarefa de classificação, e nosso alvo é o alinhamento dos super-heróis (bom ou mau).

Questão 3

Primeiro, usaremos o algoritmo Naive Bayes. Execute o algoritmo nos dados dos super-heróis para prever a variável de alinhamento e avaliar os resultados. Novamente, detalhe qualquer pré-processamento e engenharia de recursos que você aplicou no processo.

1. Quais hipóteses assumimos ao usar o algoritmo Naive Bayes?

O algoritmo Naive Bayes é amplamente utilizado para tarefas de classificação devido à sua eficiência e simplicidade. Ao aplicar este método, algumas hipóteses fundamentais são assumidas que devem ser compreendidas e avaliadas:

- **Independência das variáveis**: Uma das suposições mais críticas do Naive Bayes é que as variáveis preditoras (atributos) são independentes entre si, dado o rótulo da classe. Isso significa que a presença ou ausência de uma característica particular em um dado não afeta a presença ou ausência de outra característica, dentro do contexto da classificação.
- **Ausência de Correlação entre variáveis**: Reforçando o pressuposto de independência, o algoritmo assume que não há correlação entre as variáveis. A consideração de que as variáveis são não correlacionadas permite que o modelo simplifique os cálculos, tratando cada atributo como se contribuísse de forma independente para a probabilidade da classe, sem interação entre eles.
- **Distribuição dos Dados**: Dependendo da variante do Naive Bayes aplicada, diferentes suposições sobre a distribuição dos dados são feitas:
    - **Naive Bayes Gaussiano**: Assume que os dados numéricos seguem uma distribuição normal (gaussiana). Este modelo é adequado quando os atributos são contínuos. Para isso, utiliza-se o teste de Shapiro-Wilk.
    - **Naive Bayes Bernoulli**: Usado para dados binários, assume que os atributos são binários e seguem uma distribuição de Bernoulli, o que é comum em situações onde características podem ser descritas como presentes ou ausentes.
    - **Naive Bayes Multinomial**: Frequente em problemas de classificação de texto, onde as características são contagens ou frequências de eventos.

Estas hipóteses são essenciais para o correto funcionamento e eficácia do algoritmo. Ao implementar o Naive Bayes, é importante verificar se essas condições são razoavelmente satisfeitas nos dados em uso, pois violações desses pressupostos podem levar a resultados ótimos e interpretações enganosas.

2. Como as características específicas deste conjunto de dados influenciam suas escolhas e resultados de modelagem?

Ao abordar o conjunto de dados em questão, várias características específicas tiveram um impacto significativo nas decisões de modelagem e nos resultados subsequentes. Algumas características foram identificadas durante a etapa inicial de Exploração de Dados (EDA), como o desbalanceamento de classes, do qual é crucial para entender a natureza dos dados e criação de modelos.

- **Desbalanceamento de Classes**: Uma das descobertas mais críticas foi o desbalanceamento significativo na variável de alinhamento do herói. Este desbalanceamento pode levar a modelos que são enviesados em favor das classes mais frequentes, afetando negativamente a precisão e a generalização do modelo em classes

menos representadas. Entretanto, como explicado anteriormente, essa é uma situação normal em uma base de dados assim, o qual no mundo real há uma tendência de haver mais heróis que vilões. Por fim, recomenda-se uma análise para o balanceamento das classes, como SMOTE.

- **Alta Dimensionalidade**: Outro desafio foi a grande dimensionalidade dos dados, especialmente com muitas variáveis relativas a poderes e características dos heróis. Então, para identificar as variáveis mais importantes utilizou-se o método de seleção RFE com uma árvore simples de decisão.

Essas características específicas do conjunto de dados guiaram a escolha de técnicas e estratégias de modelagem. O entendimento profundo adquirido através da EDA permitiu implementar soluções direcionadas que abordassem esses desafios de maneira eficaz, resultando em modelos mais robustos e interpretações mais confiáveis dos dados.

3. Como você avalia os resultados?

Os resultados de um modelo de classificação são geralmente avaliados com base nas seguintes métricas:

- **F1-Score**: É a média harmônica entre Precision e Recall. Um valor baixo de F1-Score sugere que uma das duas métricas, ou ambas, estão igualmente baixas. Está métrica é particularmente útil quando se deseja encontrar um equilíbrio entre Precision e Recall, especialmente em cenários onde as classes são desbalanceadas (nosso caso).
- **Precision**: Está métrica indica a precisão do modelo ao prever uma classe específica. Em outras palavras, quantifica a proporção de identificações positivas que foram realmente corretas. Uma Precision alta indica que a maioria das previsões positivas feitas pelo modelo está correta, ou seja, o herói possui alinhamento Bom.
- **Recall**: O Recall mede a capacidade do modelo de identificar todas as categorias relevantes dentro de uma classe específica. Uma alta revocação indica que o modelo capturou a maioria das amostras positivas.
- **Acurácia**: Está métrica indica a proporção de previsões corretas (tanto positivas quanto negativas) em relação ao total de previsões feitas. A acurácia é útil para uma visão geral da eficácia do modelo, mas pode ser enganosa em situações com classes desproporcionais (nosso caso).

Cada uma dessas métricas fornece uma visão diferente da performance do modelo, permitindo uma análise abrangente de sua eficiência em diversos aspectos da classificação, das quais podem ser calculadas através da matriz de confusão, caso necessário. Por fim, vale ressaltar que, dada as políticas públicas dos quais os países aplicam ao redor do mundo, é correta o desbalanceamento de classes referente ao alinhamento, uma vez que há muito mais heróis que vilões.

Questão 4

Agora sinta-se à vontade para executar o algoritmo de classificação que julgar mais adequado para essa tarefa.

1. O que motivou sua escolha do algoritmo?

A principal escolha por outro algoritmo de classificação se da por ser mais generalista em comparação ao Naive Bayes, como a Random Forest. Esta escolha foi motivada por várias razões estratégicas que destacam a superioridade do Random Forest para o contexto específico dos nossos dados:

- **Robustez contra o Overfitting**: Diferentemente de modelos como o Naive Bayes, o Random Forest é conhecido por sua robustez e capacidade de generalizar melhor para dados não vistos. Isso se deve ao seu mecanismo de construção de múltiplas árvores de decisão e a agregação de seus resultados (ensemble learning), o que reduz o risco de overfitting que é frequentemente observado em modelos mais simples ou menos complexos.
- **Não Exigência de Independência entre Variáveis**: Ao contrário do Naive Bayes, que requer que as variáveis preditoras sejam independentes entre si, o Random Forest não possui tal exigência. Isso é crucial em nosso caso, onde as variáveis podem ter interações complexas que são importantes para a classificação correta dos heróis baseado em seus alinhamentos e poderes.
- **Interpretabilidade e Análise de Importância de Variáveis**: Apesar de ser um modelo de caixa preta relativamente complexo, o Random Forest oferece insights úteis sobre a importância das variáveis envolvidas na classificação. Isso nos permite entender melhor quais características são mais decisivas na determinação do alinhamento do herói, enriquecendo nossa análise.

Em resumo, a escolha do Random Forest foi baseada na necessidade de um modelo que não apenas performasse bem em termos de acurácia, mas que também fosse robusto, flexível e informativo, adequando-se às complexidades do nosso conjunto de dados. Esta abordagem nos permite maximizar tanto a precisão quanto a interpretabilidade dos resultados da classificação.

2. Como esse algoritmo se compara ao Naive Bayes em relação às suposições e resultados da modelagem?

O algoritmo de Random Forest Classifier se destaca em relação ao Naive Bayes nos seguintes pontos:

- **Nenhuma Suposição de Independência**: Ao contrário do Naive Bayes, o Random Forest não assume independência entre as características. Ele pode capturar interações complexas entre variáveis através de múltiplas árvores de decisão, tornando-o mais robusto e geralmente mais preciso em conjuntos de dados com relações complexas.
- **Flexibilidade com Distribuições de Dados**: Random Forest não faz suposições explícitas sobre a distribuição dos dados, podendo trabalhar eficazmente tanto com características numéricas quanto categóricas.
- **Robustez**: Geralmente, proporciona um alto desempenho em uma ampla gama de tarefas de classificação devido à sua capacidade de construir muitas árvores de decisão e obter a média de seus resultados, o que ajuda a reduzir a variância e o overfitting.

- **Manuseio de Alta Dimensionalidade e Análise de Importância de Variáveis**: Efetivo no manuseio de dados com muitas variáveis preditoras e é capaz de identificar as variáveis mais importantes, o que é vantajoso para interpretação e redução de dimensionalidade.

Em resumo, enquanto o Naive Bayes pode ser ideal para situações em que a independência entre variáveis é uma suposição razoável e a eficiência é crucial, o Random Forest é frequentemente preferido em cenários que requerem uma modelagem mais robusta e capaz de lidar com complexidades e interdependências entre variáveis. Porém, como demonstrado na análise, o Ranfom Forest utilizou apenas a variável IMC como preditora, após a análise da importância de variáveis (da qual continha Peso e Altura também, porém visando a não multicolinearidade, usou-se apenas o IMC) e não obteve métricas melhores.

Além do bem e do mal

Questão 5

Vamos transformar nosso problema em uma tarefa de regressão e tentar prever o peso dos super-heróis dados os outros recursos.

1. Qual algoritmo você escolheu e por quê?

O modelo escolhido foi o Random Forest Regressor, um modelo que pode ser bastante generalista e alcançar ótimos resultados. Esta escolha foi motivada por várias razões estratégicas que destacam a superioridade do Random Forest para o contexto específico dos nossos dados:

- **Interpretabilidade e Análise de Importância de Variáveis**: Apesar de ser um modelo de caixa preta relativamente complexo, o Random Forest oferece insights úteis sobre a importância das variáveis envolvidas na classificação. Isso nos permite entender melhor quais características são mais decisivas na determinação do alinhamento do herói, enriquecendo nossa análise.
- **Robustez contra o Overfitting**: Diferentemente de modelos como o Naive Bayes, o Random Forest é conhecido por sua robustez e capacidade de generalizar melhor para dados não vistos. Isso se deve ao seu mecanismo de construção de múltiplas árvores de decisão e a agregação de seus resultados (ensemble learning), o que reduz o risco de overfitting que é frequentemente observado em modelos mais simples ou menos complexos.

2. Como você avalia o desempenho do seu algoritmo neste caso?

Os resultados de um modelo de regressão são geralmente avaliados com base nas seguintes métricas:

- **R^2 (Coeficiente de Determinação)**: Fornece uma indicação de quão bom é o ajuste do modelo aos dados observados. Um valor de R^2 mais próximo de 1 indica que o modelo explica uma grande parte da variância nos dados, enquanto um valor próximo de 0 indica que o modelo não explica a variância dos dados de forma eficaz. Entretanto, um valor muito próximo de 1 pode indicar overfitting, bem como um valor próximo de 0 indicar underfitting.
- **MAPE (Erro Percentual Absoluto Médio)**: Mede a precisão do modelo de regressão como uma porcentagem e é calculado como a média dos valores absolutos dos erros percentuais. Está métrica é particularmente útil porque ajuda na interpretabilidade de

metrificação do modelo para pessoa mais leigas. Por exemplo, um MAPE de 10% significa que, em média, a previsão do modelo está dentro de 10% do valor real

- **MAE (Erro Absoluto Médio)**: Fornece uma média dos erros absolutos entre os valores previstos e os valores observados. Ao contrário do erro quadrático médio, o MAE mede os erros absolutos sem elevar ao quadrado, evitando, portanto, a penalização excessiva de erros grandes.

Em resumo, O R^2 fornece uma visão geral da adequação do modelo nos dados, o MAPE ajuda a entender a precisão em termos percentuais, e o MAE oferece uma medida direta dos erros de previsão. Ao comparar essas métricas, você pode fazer ajustes no modelo para melhorar seu desempenho ou escolher o modelo mais adequado para a tarefa em questão.

Análise

Questão 6

Quais aspectos desse conjunto de dados apresentam problemas para agrupamento, classificação e regressão? Como você resolveu esses problemas?

Os principais problemas encontrar no decorrer da criação dos modelos e análises foram:

Dados nulos ou sujos: Principalmente na base de dados de informações dos heróis, variáveis importantes como Peso e Altura, possuíam dados nulos ou sujos, dos quais necessitaram tratamento. Assim, o tratamento para as variáveis de Peso e Altura se deu pela média olhando o agrupamento do respectivo gênero e raça. Por fim, a porcentagem de dados nulos no restante das colunas era abaixo de 5%, decidindo pelo descarte.

Alta dimensionalidade: A base de dados com as informações dos poderes de cada super-herói possuía cerca de 168 colunas, das quais uma análise particular levaria bastante esforço. Para isso, principalmente visando a criação dos modelos e escolha das melhores variáveis, foram utilizados métodos como RFE, PCA e Features importances, visando reduzir a dimensionalidade ou escolher apenas as melhores dentre todas.

Streamlit

Desenvolva uma aplicação interativa usando Streamlit que permita aos usuários explorar o conjunto de dados de super-heróis e interagir com os modelos de machine learning desenvolvidos nas questões anteriores.

Exploração de Dados: A aplicação deve permitir que os usuários visualizem e explorem os conjuntos de dados super hero powers.csv e heroes information.csv. Implemente funcionalidades para exibir estatísticas descritivas, distribuições de variáveis, e a capacidade de filtrar super-heróis por diferentes critérios (alinhamento, gênero, editora).

Resultados do Clustering (Questão 1): Integre o modelo de clustering desenvolvido na Questão 1. A aplicação deve permitir que os usuários visualizem os clusters de super-heróis gerados e explorem as características comuns dentro de cada cluster. Se possível, inclua uma visualização interativa como um gráfico de dispersão ou um mapa de calor para representar os clusters.

Classificação do Alinhamento (Questão 3): Incorpore o modelo de Naive Bayes (ou o modelo de classificação de sua escolha da Questão 4) na aplicação. Os usuários devem ser capazes de

selecionar ou inserir informações de um super-herói e receber uma previsão do alinhamento (bom ou mau).

Previsão de Peso (Questão 5): Adicione funcionalidade para que os usuários possam prever o peso de um super-herói com base em suas características. Isso inclui a integração do modelo de regressão desenvolvido na seção correspondente do desafio.

Documentação e Instruções: Forneça comentários claros no código e uma seção na aplicação que explique como usar as diferentes funcionalidades.

Interatividade e Design: A interface deve ser intuitiva e amigável. Utilize as capacidades de personalização do Streamlit para criar uma experiência visual agradável para os usuários.

Streamlit - Implementação

Visando uma interface amigável e interativa, a HeroStats decidiu por a público sua plataforma de análise dos super-heróis. Acesse em: https://hero-stats.streamlit.app/.