**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 utilizado o algoritmo *Kmeans* para construção dos clusters por sua fácil implementação, sua garantia de convergência e sua generalização para clusters de diferentes tamanhos e densidade. Apesar das dificuldades do algoritmo na sua utilização em dados multidimensionais, a base de super-heróis possui apenas 178 *features*, o que não afetou a performance do algoritmo.

O algoritmo de *Kmeans* também tem dificuldades em determinar clusters em formas complexas, por exemplo, o *dataset* de *two moons*. Assim, também foi utilizado o algoritmo de DBSCAN a nível de comparação de resultados.

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

Utilizando o *sklearn-pipelines* é possível testar facilmente diferentes tipos de pré-processamento e analisar o resultado final do algoritmo. Nas variáveis categóricas, os valores faltantes foram ignorados, pois não era possível inferir as informações faltantes. As variáveis numéricas foram centralizadas e o algoritmo de redução de dimensão *PCA* foi aplicado nas variáveis booleanas. Pois, 168 variáveis booleanas, podem ter 95% da variância explicada por 89 variáveis.

**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?

Foi utilizado o método de *Elbow* para determinação dos clusters do algoritmo de Kmeans e o algoritmo DBSCAN faz a própria determinação do número ideal de cluster, porém é necessário otimizar o parâmetro de *eps*. Para isso, foi utilizado o método que determina o valor do *eps* por meio do ponto máximo de curvatura através dos vizinhos mais próximos.

Para avaliação da qualidade dos cluster foi utilizado *Silhoette Score* e visualizada a distribuição dos clusters através do t-SNE.

**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 Teorema de Bayes assume a independência entre as *features*. Em outras palavras, um classificador Naive Bayes assume que a presença de uma determinada *feature* em uma classe não está relacionada à presença de qualquer outra *feature*.

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

Se a variável categórica tiver uma categoria (no conjunto de dados de teste), que não foi observada no conjunto de dados de treinamento, o modelo atribuirá uma probabilidade 0 (zero) e não será capaz de fazer uma previsão. Esse problema é conhecido como “Freqüência Zero”. Para resolver isso, podemos usar a técnica de suavização. Uma das técnicas de suavização mais simples é chamada de estimativa de Laplace.

Foi utilizado BernoulliNB adequado para dados binários, o classificador BernoulliNB conta quão frequente cada feature não tem o valor de 0.

3. Como você avalia os resultados?

Os resultados foram avaliados utilizando a matriz de confusão, por serem dados booleanos. Nela é possível identificar em quais classes o modelo tem pior ou melhor performance. Para o problema de classificação em questão, o erro tipo I é o mais importante (vilões classificados como heróis). Nesse quesito, o algoritmo classificou 12% dos dados errados.

Também foi utilizada a técnica de validação cruzada dos dados (*Kfold*). Os dados foram divididos em 10 pastas e a variável alvo foi *stratified*, ou seja, devido ao seu desbalanceamento (a base possui mais heróis do que vilões) cada pasta foi dividia mantendo a proporção dos dados de “*good*” e “*bad*”.

**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?

Para classificação foi escolhido o modelo *ensemble* Random *Forest Classifier* por ser robusto a *outliers*, boa performance em dados não lineares, baixo risco de *overfitting* e no geral tem melhor performance que a maioria dos algoritmos.

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

É esperado uma melhor performance do modelo de *Random Forest* pela sua capacidade de generalização dos dados através das diversas árvores. Como algumas categorias possuem somente um exemplar, para este caso foi utilizado o parâmetro de min\_samples\_leaf = 2, que determina o número mínimo de exemplos para criação do ramo.

Para o modelo de Naive Bayes obter o mesmo desempenho seria necessária a otimização do parâmetro *alpha*, usando técnica de *Random Search* e *Grid Search*, tais técnicas não foram utilizadas no trabalho. Também foi utilizada a técnica de validação cruzada no *Random Forest*.

Os resultados condizem com o esperado, o *Random Forest* obteve melhores resultados que o Naive Bayes, sendo o erro tipo I igual 0.

**Além do bem e do mal**

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 algoritmo escolhido foi o *Random Forest Regressor* devido as mesmas características apresentadas e acima com funcionamento similar ao classificador, sendo a diferença, o retorno do erro médio absoluto de cada árvore. Os resultados do modelo de classificação corroboraram a escolha para o modelo de regressão.

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

O modelo foi avaliado pelas seguintes métricas: R2 que mede o quanto da variabilidade da variável dependente foi explicada pelo modelo, *RMSE* que mede o *fit* do modelo e *MAE* para análise dos erros considerando que todos têm o mesmo peso.

**Análise**

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

Grande parte das categorias estão desbalanceadas: a *feature* de *Race* 3 classes correspondiam a 81% dos dados e das 46 raças, 25 possuíam somente um exemplar. O mesmo ocorreu com a *feature* de *Publisher,* 81% dos dados estavam divididos entre *Marvel/DC* e 9 *Publishers* possuíam somente um personagem dos 24 tipos. Foram utilizadas técnicas de *sampling* (KNN para determinação dos valores faltantes) para resolução dos problemas, porém os melhores resultados foram obtidos usando as técnicas apresentadas na questão 1.

**Bônus**

Se você gostou de brincar com o conjunto de dados de super-heróis, esta seção é para você mostrar quaisquer outros aspectos dos dados que não exploramos nas perguntas. Como uma seção de bônus, isso é totalmente opcional, mas adoraríamos ver os insights que você pode obter desses dados.

* Existe um perfil comum de personagem: homem com olhos azuis e cabelo preto (72% dos personagens são do gênero masculino, 40% possuem olhos azuis e 28% possuem cabelo preto).
* A Marvel é responsável por 54% dos personagens.
* Existem 496 heróis para combater 207 vilões (os vilões precisam ser 2.4x mais fortes que heróis para haver equilíbrio).
* *Stamina* e *Super Strength* força são as duas habilidades balanceadas entre os personagens (distribuição próxima de 50%)

PS: Os resultados e análises foram baseados em evidências empíricas e conhecimento prévio, uma vez que não haveria tempo suficiente para levantamento bibliográfico.