CEFET/RJ
Programa de Pós-graduação em Ciência da Computação
Aprendizado de Máquina (CIC1205) - 2023
Trabalho 2

Prof. Eduardo Bezerra (ebezerra@cefet-rj.br)

30 de novembro de 2023

Sumário

1 Engenharia de *Features* (2 pts)

Considere novamente o problema de classificação apresentado na parte 1 do Trabalho 1. Nesta parte do trabalho, você deve usar os mesmos conjuntos de dados (treino e teste) que usou naquela ocasião. Sua tarefa aqui é investigar outras técnicas para codificação de atributos categóricos nesses conjuntos de dados.

Em aula, estudamos a aplicação das técnicas One-Hot encoding, Ordinal Encoding e Target Encoding para codificar atributos categóricos. Considere a biblioteca Categories Encoder[1]. Estude a documentação fornecida por essa biblioteca para produzir uma boa combinação de técnicas de codificação dos atributos categóricos contidos no conjunto de dados fornecido. Você deve usar ao menos duas técnicas diferentes das que usou no Trabalho 1. Em seguida, execute o processo de construção do modelo de classificação. Utilize apenas um algoritmo de aprendizado dentre os listados na parte 1 do Trabalho 1[2]; a escolha de qual algoritmo usar fica a seu critério. Apresente uma análise comparativa dos novos resultados frente aos resultados correspondentes que você produziu no Trabalho 1.

2 Validação Cruzada (1 pt)

Nesta parte, você deve revisitar os seguintes arquivos fornecidos no Trabalho 1: A602.pickle, A621.pickle, A627.pickle, A636.pickle, A652.pickle. Você deve novamente construir modelos de classificação binária usando esses conjuntos. Contudo, dessa vez você deve realizar a busca de hiperparâmetros por meio de validação cruzada com 5 partes (*5-fold cross-validation*).

Visto que os conjuntos de dados fornecidos estão divididos em treino, validação e teste, para aplicar a validação cruzada, você deve inicialmente unir os conjuntos de treino e de validação para gerar um único arquivo de treino. O arquivo resultante desta união deverá ser usado no procedimento de validação cruzada. Repare que os conjuntos de teste não devem ser alterados.

Outro aspecto que você deve considerar é que, dado o desbalanceamento dos conjuntos de dados fornecidos, você deve usar a classe `StratifiedKFold`[3] para realizar a validação cruzada durante a busca de hiperparâmetros. Essa classe mantém

[1] `https://contrib.scikit-learn.org/category_encoders/`

[2] Repare que em um contexto de aplicação real, o mais adequado seria experimentar todas as opções de algoritmos (via seleção de modelo) para determinar qual a melhor opção.

[3] `https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold.html`

a proporção das classes positiva e negativa ao fazer a divisão dos exemplos durante a validação cruzada.

Após executar a busca de hiperparâmetros, aplique os melhores modelos gerados sobre os conjuntos de testes correspondentes. Realize uma análise comparativa desses novos resultados com relação aos resultados correspondentes que você obteve no Trabalho 1.

3 Classificação Multi-classes (1 pt)

Nesta parte do trabalho, você deve revisitar os seguintes arquivos fornecidos no Trabalho 1: A602.pickle, A621.pickle, A627.pickle, A636.pickle, A652.pickle. No Trabalho 1, você enquadrou o problema como uma tarefa de classificação binária. Desta vez, você deve enquadrar o problema como uma tarefa de classificação multi-classes. Concretamente, desta vez, você deve discretizar os valores de precipitação (medidos em mm/h) produzindo cinco níveis (classes), conforme o seguinte mapeamento:

- $0 \rightarrow$ NONE
- $(0, 5] \rightarrow$ WEAK
- $(5, 25] \rightarrow$ MODERATE
- $(25,50] \rightarrow$ STRONG
- $(50, \infty] \rightarrow$ EXTREME

Crie modelos de classificação para cada um dos conjuntos de dados fornecidos. Utilize o algoritmo `GradientBoostingClassifier`, dessa vez equipado com a técnica de regularização denominada *Early Stopping*. Apresente a curva de aprendizado correspondente ao treinamento dos modelos. Apresente seus resultados (medidos sobre os respectivos conjuntos de teste) na forma de matrizes de confusão e utilizando a função `classification_report` do Scikit-Learn. Apresente também uma análise comparativa com os resultados correspondentes que você produziu no Trabalho 1.

4 SHAP Values (2 pts)

Considere novamente o problema de classificação multi-classes apresentado na parte 3 deste trabalho. Considere também novos arquivos fornecidos cujo padrão de nomenclatura é `AXXX_train.parquet.gzip`, `AXXX_val.parquet.gzip` e `AXXX_test.parquet.gzip`.

Nesses arquivos, **XXX** representa o identificador da estação meteorológica na qual foram observados os dados. Os nomes das colunas nesses arquivos permitem que você entenda o significado de cada coluna.

Para cada modelo gerado, apresente uma análise de explicabilidade do comportamento desse modelo sobre os exemplos do conjunto de testes correspondente. Você deve usar a biblioteca SHAP[4] para dar suporte à sua análise. Apenas como sugestão (você deve refletir sobre as perguntas que entende fazerem sentido), seguem algumas perguntas para guiar a sua análise.

a. Qual a importância de cada feature para uma predição específica?

b. Como uma feature específica impacta a predição em geral?

c. Como cada feature contribui para o desempenho do modelo?

d. Quais instâncias da classe EXTREME são as mais influenciadas por uma determinada feature?

e. Quais features têm contribuições consistentes ou inconsistentes em instância da classe EXTREME?

f. Como as predições mudam com variações nas features?

g. Quais features estão mais correlacionadas com outras em termos de contribuição para as predições?

5 Redução de dimensionalidade (2 pts)

Considere novamente o problema de classificação multi-classes apresentado na parte 3 deste trabalho. Nesta parte, você irá realizar um experimento usando o algoritmo PCA. Por simplificação, considere apenas o conjunto de dados cujo código é A652. Inicialmente, usando validação cruzada, escolha o melhor valor para a quantidade de componentes principais. Para fazer essa escolha, se assegure de que você não está usando a variável alvo como *feature*. Em seguida, ajuste novamente um modelo de classificação multi-classe sobre o conjunto de dados resultante da redução de dimensionalidade. Finalmente, apresente uma análise comparativa entre esse novo modelo e o anteriormente ajustado (usando todas dimensões originais).

[4] `https://shap.readthedocs.io/en/latest/`

6 Agrupamento (2 pts)

Considere novamente o problema de classificação multi-classes apresentado na parte 3 deste trabalho. Nesta parte, você irá realizar um experimento usando dois algoritmos de agrupamento disponíveis no Scikit-Learn, o K-Means e o DBSCAN. Durante essa tarefa, espera-se que você utilize validação cruzada para determinar valores adequados dos hiperparâmetros desses algoritmos.

Os grupos criados por algoritmos de agrupamento tradicionalmente carecem de uma forma direta de interpretação. Nesta parte do trabalho, você irá implementar uma forma simples de interpretar os grupos gerados por um algoritmo de agrupamento, conforme especificado a seguir.

Por simplificação, considere apenas o conjunto de dados cujo código é A652. Execute os dois algoritmos de agrupamento sobre esse conjunto de dados. Para fazer essa execução, se assegure de que você não está usando a variável alvo como *feature*. Em seguida, use os rótulos encontrados pelo algoritmo de agrupamento para treinar um modelo `DecisionTreeClassifier`. Finalmente, se aproveitando do código disponível em `https://mljar.com/blog/extract-rules-decision-tree/`, crie regras para explicar cada grupo gerado. Apresente uma análise das regras geradas. Elas parecem fazer sentido? Qual sua interpretação do resultado?

Especificação da entrega

- Você deve produzir um notebook Jupyter que deve apresentar as **implementações e os resultados de execução** de cada parte desse trabalho. Nesse notebook, descreva **em detalhes** de que forma implementou cada parte desse trabalho. Já é fornecido junto com este enunciado um notebook para você usar como ponto de partida neste trabalho.

- O único arquivo a ser submetido é o notebook Jypyter. Esse arquivo deve ser nomeado com o seguinte padrão: AM_T2_SEU_NOME_COMPLETO.ipynb. Um exemplo: AM_T2_EDUARDO_BEZERRA_DA_SILVA.ipynb. Siga à risca essa convenção de nomenclatura.

- Você deve também elaborar um vídeo (cuja duração aproximada foi especificada no primeiro dia de aula) no qual você deve explicar os aspectos mais importantes de cada parte do seu trabalho. Nesse vídeo, você também deve demonstrar a execução de cada parte e apresentar uma análise dos resultados obtidos. O link para acesso a esse vídeo deve estar contido na primeira célula (de texto) do notebook Jupyter.

- A entrega aqui especificada deve ser realizada pela plataforma MS Teams, até a data estabelecida. Trabalhos entregues com atraso irão sofrer desconto na nota (20% a cada dia de atraso).

- Esse trabalho é individual. Você é livre para discutir com seus colegas de turma sobre as partes desse trabalho, mas deve manter para si as suas soluções. Eventuais cópias em quaisquer partes do trabalho serão penalizadas com nota zero.