Universidade do Minho
Departamento de Informática
Mestrado [integrado] em Engenharia Informática

N° _____ CURSO _____

NOME eduardovski

Dados e Aprendizagem Automática
1° Ano, 1° Semestre
Edição 2022/2023

Prova escrita, 5 de janeiro, 2023

GRUPO 1
(4 valores)

RESPONDA ÀS QUESTÕES DESTE GRUPO NO ESPAÇO RESERVADO PREENCHENDO OS ESPAÇOS VAZIOS COM AS EXPRESSÕES CORRETAS.

QUESTÃO 1

Numa metodologia de análise de dados como o CRISP-DM, a preparação de dados é uma tarefa anterior à modelagem e é preponderante visto que os dados recolhidos do mundo real podem apresentar-se incompletos.

QUESTÃO 2

Algoritmos de *Clustering*, tais como k-means e dbscan, implementam uma técnica de aprendizagem nao supervisionada com o objetivo de agrupar um conjunto de casos de estudo, de tal forma que os objetos no mesmo grupo apresentam mais semelhanças entre si do que com outros grupos.

QUESTÃO 3

Feature Engineering permite a criação de novas colunas / variaveis a partir da informação disponível, como forma de auxiliar o modelo a realizar previsões mais precisas.

GRUPO 2
(4 valores)

RESPONDA ÀS QUESTÕES DESTE GRUPO EM FOLHA DE TESTE SEPARADA.

QUESTÃO 1

Em alguns algoritmos de *Machine Learning* é usada a técnica de descida por gradiente (*gradient descent*) no processo de otimização dos parâmetros do algoritmo.

a) Quais poderão ser os motivos para esta convergir lentamente ou não convergir?

b) Indique 2 exemplos de algoritmos de *Machine Learning* que façam uso desta técnica.

Questão 1 (a)

Quais poderão ser os motivos para a descida por gradiente convergir lentamente ou não convergir?

1. **Taxa de aprendizado inadequada (learning rate)**:
   - Uma taxa de aprendizado muito **baixa** pode tornar a convergência muito lenta.
   - Uma taxa de aprendizado muito **alta** pode fazer com que o algoritmo oscile ou nunca alcance o mínimo.
2. **Paisagem do erro não suave**:
   - Presença de **múltiplos mínimos locais**, saddle points ou gradientes muito pequenos (vanishing gradients) pode dificultar a convergência.
3. **Problemas de escala nas features**:
   - Dados não normalizados ou mal escalados podem levar a trajetórias ineficientes no espaço dos parâmetros.
4. **Superfície de erro mal condicionada**:
   - Matrizes de segunda ordem (Hessiana) mal condicionadas podem gerar direções instáveis para o gradiente.

Questão 1 (b)

Indique 2 exemplos de algoritmos de Machine Learning que façam uso da técnica de descida por gradiente:

1. **Regressão Logística**
2. **Redes Neurais Artificiais**

N° _____

GRUPO 3
(4 valores)

PARA CADA AFIRMAÇÃO, RESPONDA ASSINALANDO A SUA VERACIDADE (**V**) OU FALSIDADE (**F**).
JUSTIFIQUE A RESPOSTA EXCLUSIVAMENTE NO ESPAÇO DISPONIBILIZADO.
NÃO SÃO CONSIDERADAS RESPOSTAS PARA AS QUAIS NÃO EXISTA JUSTIFICAÇÃO.

QUESTÃO 1 ☐ O algoritmo de aprendizagem *Decision Tree* apresenta normalmente um melhor desempenho quando comparado com o algoritmo *Random Forest*, apresentando características que possibilitem mitigar o problema de *overfit* de dados.

Resposta: F (Falso)

- **Justificação:** O Random Forest combina várias árvores de decisão, reduzindo o risco de overfitting devido à agregação (ensemble). Isso geralmente o torna mais robusto e eficaz do que uma única Decision Tree.

QUESTÃO 2 ☐ A *Off-Policy Learning* verificada nos algoritmos de *Reinforcement Learning* considera a avaliação e a otimização da respetiva *policy* aplicada para a seleção das ações do algoritmo inteligente.

Conceito de Off-Policy Learning

- No Reinforcement Learning, **Off-Policy Learning** significa que o algoritmo **aprende sobre uma política (policy) enquanto segue outra**.
- A política que o agente segue para coletar dados (denominada **behavior policy**) é diferente da política que o agente está tentando otimizar (denominada **target policy**).

Um exemplo clássico de algoritmo Off-Policy é o **Q-Learning**, que atualiza a política target independentemente de como os dados são coletados.

Resposta: V (Verdadeiro)

- **Justificação:** Em Off-Policy Learning, a política usada para gerar os dados de treinamento pode ser diferente da política que está sendo avaliada ou otimizada.

QUESTÃO 3 ☐ Uma matriz de confusão é uma métrica de avaliação de desempenho de modelos de *Reinforcement Learning*.

Resposta: F (Falso)

- **Justificação:** A matriz de confusão é uma métrica usada para avaliar modelos de classificação, não diretamente associada ao desempenho de modelos de Reinforcement Learning.

QUESTÃO 4 ☐ Em todos os algoritmos de *clustering* é necessário justificar a quantidade de *clusters* a procurar nos dados.

Resposta: F (Falso)

- **Justificação:** Nem todos os algoritmos de clustering requerem a definição prévia da quantidade de clusters. Por exemplo, o DBSCAN determina o número de clusters com base em densidades de pontos, sem necessidade de especificar previamente um valor.

N° _____

GRUPO 4
(6 valores)

RESPONDA ÀS QUESTÕES DESTE GRUPO NO ESPAÇO RESERVADO.

Considere o *dataset* "breast_cancer", usado diversas vezes no decorrer das aulas, com o intuito de treinar um modelo de classificação com capacidade de prever a existência de um tumor mamário, de acordo com alguns dados clínicos do paciente.

Considere, ainda, o excerto de código abaixo, onde se apresenta a construção e avaliação de um modelo de aprendizagem automática.

QUESTÃO 1

O excerto de código apresentado contém imprecisões. Identifique-as e corrija-as utilizando o espaço disponível ao lado do excerto (não deve copiar todo o excerto, mas apenas aquilo que corrigiu).

```
[1]  df = pandas.read_csv('breast_cancer_dataset.csv')
[2]  df['diagnosis'] =
            df['diagnosis'].substitute(['B', 'M'], [0, 1])
[3]  X = df.drop(['diagnosis', 'id'], axis=1)
[4]  y = df['diagnosis']
[5]  X_train, X_test, y_train, y_test =
         train_test_split(X, y, test_size=1, random_state=2023)
[6]  model = RandomForestClassifier(random_state=2023)
[7]  model.predict(X_train, y_train)
[8]  inferences = model.fit(X_test)
[9]  accuracy = accuracy_score(y_train, inferences)
[10] mse = MSE(y_test, inferences)
[11] print(classification_report(y_test, inferences))
[12] print(confusion_matrix(y_test, inferences))
```

1. **Substituir valores na coluna** `diagnosis`: Trocar `.substitute` por `.replace`.
2. **Divisão dos dados**: Corrigir `test_size=1` para `test_size=0.3` (30% para teste, 70% para treino).
3. **Treinamento do modelo**: Usar `model.fit` antes de qualquer inferência.
4. **Inferências**: Substituir `model.fit` por `model.predict` para obter predições.
5. **Métricas**: Importar `mean_squared_error` e corrigir os parâmetros das funções de avaliação.

```
# Linha 7: 'model.predict' está mal posicionado. Devemos usar 'model.fit' antes para treina
model.fit(X_train, y_train)  # Correção

# Linha 8: 'model.fit' não é usado para inferências. Usar 'model.predict' para gerar predi
inferences = model.predict(X_test)  # Correção

# Linha 9: O cálculo de acurácia está errado. Os parâmetros do 'accuracy_score' precisam s
accuracy = accuracy_score(y_test, inferences)  # Correção
```

Figura 1. Excerto de um modelo de aprendizagem.

GRUPO 5
(2 valores)

ASSINALE A VERACIDADE (**V**) OU FALSIDADE (**F**) DE CADA UMA DAS AFIRMAÇÕES QUE SE APRESENTAM.
UMA AFIRMAÇÃO INCORRETAMENTE ASSINALADA ANULA UMA RESPOSTA ASSINALADA CORRETAMENTE.

QUESTÃO 1

Qual o significado de '*boosting*' no contexto de modelos de previsão?

- ☐ Fazer diferentes modelos "votar" para obter uma solução final;
- ☐ Validar um modelo utilizando conjuntos de dados maiores;
- ☐ Treinar modelos iterativamente de acordo com os erros de classificação;
- ☐ Dividir aleatoriamente um conjunto de dados para produzir modelos alternativos.

QUESTÃO 2

Qual o significado de 'categórico' quando nos referimos a uma variável num conjunto de dados?

- ☐ Uma variável categórica não pode ser transformada;
- ☐ Não se usam valores numéricos para codificar uma variável categórica;
- ☐ Uma variável categórica não pode ser utilizada como variável dependente/*target*;
- ☐ Uma variável categórica não pode ser utilizada como um número/quantidade.

1. **Fazer diferentes modelos "votar" para obter uma solução final:**
   F (Falso)
   - Isto descreve **bagging**, como no Random Forest, mas não o conceito de boosting.
2. **Validar um modelo utilizando conjuntos de dados maiores:**
   F (Falso)
   - Boosting não está relacionado à validação de modelos ou tamanho de conjuntos de dados.
3. **Treinar modelos iterativamente de acordo com os erros de classificação:**
   V (Verdadeiro)
   - Boosting funciona ajustando modelos sequenciais, onde cada modelo tenta corrigir os erros do modelo anterior. Exemplo: **AdaBoost**.
4. **Dividir aleatoriamente um conjunto de dados para produzir modelos alternativos:**
   F (Falso)
   - Esta é uma descrição do bagging, não do boosting.

1. **Uma variável categórica não pode ser transformada:**
   F (Falso)
   - Variáveis categóricas podem ser transformadas em representações numéricas (ex.: one-hot encoding).
2. **Não se usam valores numéricos para codificar uma variável categórica:**
   F (Falso)
   - É comum codificar variáveis categóricas com números (ex.: label encoding).
3. **Uma variável categórica não pode ser utilizada como variável dependente/target:**
   F (Falso)
   - Muitas vezes, variáveis categóricas são usadas como target (ex.: classificação binária ou multi-classe).
4. **Uma variável categórica não pode ser utilizada como um número/quantidade:**
   V (Verdadeiro)
   - Uma variável categórica representa categorias e não possui um significado numérico intrínseco, mesmo se codificada numericamente.