

Tarefa 3: Regredir e Classificar

Resgate de Vítimas de Catástrofes Naturais, Desastres ou Grandes Acidentes

1 Objetivo da tarefa

O socorrista (A_s) recebe os dados de sinais vitais de cada uma das vítimas:

- **qPA**: qualidade da pressão arterial; resulta da avaliação da relação entre a pressão sistólica e a diastólica;
- **pulso**: pulsação ou Batimento por Minuto (pulso) e
- **frm**: frequência da respiração por minuto (respiração).

Nesta tarefa, imagine você tem um histórico de sinais que foram coletados de outros acidentes e analisados por um corpo médico. Estes médicos construíram uma fórmula para calcular o valor de gravidade do estado de saúde das vítimas e, também, estabeleceram intervalos que definem as seguintes classes de gravidade:

1 = crítico, **2** = instável, **3** = potencialmente estável e **4** = estável.

O problema é que a fórmula de cálculo e os limites de classificação de gravidade dos ferimentos foram perdidos. Portanto, você deve utilizar técnicas de aprendizado de máquina para reconstituir o cálculo (regressão) e a classificação a partir dos exemplos que estão nos arquivos de histórico. Com base no modelo que for aprendido, o desempenho final será calculado com dados de teste que não foram utilizados no treinamento (serão passados mais tarde para avaliação da tarefa).

Portanto, a tarefa tem dois objetivos:

- 1) comparar os resultados produzidos por duas técnicas diferentes capazes de fazerem regressão;
- 2) comparar os resultados produzidos por duas técnicas diferentes capazes de realizarem classificação.

A comparação deve ser feita utilizando-se as métricas usuais durante as fases de treinamento/validação e testes (e.g. raiz quadrada do erro quadrático médio (RMSE), precision, recall, f-measure, acurácia, matriz de confusão).

2 Arquivos de treinamento/validação e testes

2.1 Arquivo sinaisvitalis_hist.txt

Este arquivo contém os dados históricos de sinais vitais de vítimas de outros acidentes. Cada linha representa uma vítima.

Para uma vítima i do histórico temos 5 sinais vitais (s_1 até s_5) que resultam a gravidade g_i da vítima. Todos os valores são números reais criados de modo randômico dentro dos intervalos apresentados.

$i \quad s_{i1} \quad s_{i2} \quad s_{i3} \quad s_{i4} \quad s_{i5} \quad g_i \quad y_i$

i : identificação da vítima (número sequencial)

s_{i1} : pressão sistólica (pSist): [5, 22] - não usar, é utilizada no cálculo de s_{i3}

- s_{i2} : pressão diastólica (pDiast): [0, 15] - não usar, é utilizada no cálculo de s_{i3}
- s_{i3} : **qualidade da pressão (qPA)**: [-10,10] onde 0 é a qualidade máxima -10 é a pior qualidade quando a pressão está excessivamente baixa, +10 é a pior qualidade quando a pressão está excessivamente alta
- s_{i4} : **pulso**: [0,200] bpm
- s_{i5} : **respiração**: [0,22] FpM (frequência de respiração)
- g_i : **gravidade**: deve ser inferido pela técnica escolhida
- y_i : **rótulo que representa a classe de saída**: deve ser inferida com base na gravidade (pós-processamento) ou produzida diretamente pela técnica (e.g. árvore de decisão produz diretamente).

Exemplo:

i	si1	si2	si3	si4	si5	g1	y1
	pSist	pDiast	qPA	pulso	resp	gravid	classe
1,	8.5806,	2.2791,	-8.4577,	56.8384,	9.2229,	33.5156,	2

2.2 Arquivo sinais_vitais_teste.txt

O *dataset* para o teste cego segue quase o mesmo formato dos dados históricos. No entanto, retiramos $si1$, $si2$, $g1$ e $y1$. Este arquivo vai ser utilizado somente na fase de teste cego do modelo aprendido para cada uma das duas técnicas escolhidas a ser fornecido no dia da entrega para podermos comparar as soluções dos diferentes grupos.

i	si3	si4	si5
	qPA	pulso	resp
1,	-8.5577,	56.8004,	9.0000

Para cada exemplo do teste cego, o programa deverá gerar um vetor de 3 colunas por n linhas separados por vírgulas com os resultados de predição numérica da gravidade e da classe.

i	gravid	classe
1,	33.5034,	2
2,	10.4034,	3
...		
n,	0.0399,	1

O professor fornecerá os valores conhecidos de gravidade e da classe e o grupo fará o cálculo de erro (RMSE) e de classificação (precision, recall, f-measure, acuracidade) para compararmos as soluções.

3 METODOLOGIA

Escolher as técnicas a serem utilizadas dentre ID3/J48, Fuzzy ou Redes Neurais. Podem ser utilizadas *Toolbox* (e.g. MatLab) ou programação com auxílio de bibliotecas existentes (e.g. Python com SciKit, Tensorflow). O importante é entender conceitualmente os parâmetros a serem definidos/implementados (não utilizar o software de maneira cega – sem entender os conceitos).

Para cada técnica escolhida, testar diferentes estruturas e parametrizações. Por exemplo, ao escolher sistema de inferência fuzzy (SIF), você pode variar as variáveis linguísticas que definem as entradas (o total de termos, as funções de pertinência que definem os conjuntos fuzzy). No caso particular de um SIF, as regras podem ser construídas manualmente ou você pode implementar o método de Wang-Mendel para gerá-las automaticamente.

Ainda, para extrair um comportamento médio independente da escolha dos dados de treinamento/validação, você deve fazer a validação cruzada várias vezes para cada configuração escolhida. Então, você seleciona o modelo que gerou o melhor resultado para efetuar os testes.

Isto é interessante porque permite analisar a capacidade de generalização do modelo aprendido. Portanto, pode haver casos em que um modelo com bom desempenho na etapa de treinamento/avaliação não seja tão bom na etapa de testes.

4 ENTREGA

- 1) Os códigos fonte. Caso utilize uma Toolbox, descrever qual foi utilizada, parametrização e scripts.
- 2) Um artigo PDF de até 8 páginas no [formato da SBC](#) com a estrutura abaixo

4.1 Estrutura do artigo

Introdução: dentro do problema como um todo, quais subproblemas atacará e por quais razões: quais são as motivações e justificativas para resolvê-los.

Fundamentação Teórica: as técnicas escolhidas com uma breve descrição

Metodologia: descreva como procedeu para cada uma das técnicas escolhidas, salientando as variações de parametrização e de estrutura (e.g. regras fuzzy, topologia da rede neural). Explicar a razão de tentar uma nova parametrização e/ou estrutura.

Resultados e análise: mostrar os resultados numéricos das métricas de desempenho para as etapas de treinamento/avaliação e para a etapa de testes. Fazer uma análise comparativa entre as técnicas escolhidas.

Conclusões: qual técnica apresentou o melhor desempenho e as razões que você crê que justificam o desempenho. Há algo a ser melhorado nas soluções apresentadas?

=== páginas contam até aqui; além daqui o nº de páginas é livre ===

Referências bibliográficas

Apêndice: instruções claras de como executar o código respeitando os formatos de arquivos de entrada e de configuração do enunciado; print das telas do programa se desejar (não colocar print das telas no corpo do artigo).

5 Critérios de correção dos projetos

- **Problema:** nível de dificuldade
- **Fundamentação Teórica:** emprego correto dos termos e conceitos
- **Abordagens Relacionadas:** qualidade e atualidade do levantamento bibliográfico
- **Proposta:** qualidade, detalhamento e correção da proposta
- **Comparação:** quais são as abordagens de comparação
- **Análise:** qualidade da análise dos resultados
- **Geral:** apresentação geral e qualidade da redação