

TIP8311 - Reconhecimento de Padrões

Programa de Pós-Graduação em Engenharia de Teleinformática
Universidade Federal do Ceará (UFC)

Responsável: Prof. Guilherme de Alencar Barreto

2º. Trabalho Computacional - 26/09/2019

Questão Única – Acesse através do link abaixo o conjunto de dados para uso neste trabalho.

<https://archive.ics.uci.edu/ml/datasets/CMU+Face+Images>

Pede-se:

1. Comparar os desempenhos dos seguintes classificadores: (i) vizinho mais próximo (NN); (ii) distância mínima ao centróide (DMC); (iii) classificador quadrático (CQ); e (iv) linear de mínimos quadrados (LMQ). Preencher a tabela de resultados abaixo. Número de rodadas independentes de treino/teste: 100. Separação treino/teste: 80/20% por classe (i.e. indivíduo). Do número total de imagens por indivíduo, selecione 80% delas para treino e o restante deixar para teste.

Classificador	Média	Mediana	[mínimo, máximo]	Desvio-padrão
NN				
DMC				
CQ				
LMQ				

2. Construa as matrizes de confusão para as classes para o melhor e pior caso ao longo das 100 rodadas. Quais classes são mais fáceis de classificar? Quais são mais difíceis? Elencar possíveis explicações para os resultados

Obs. 1: Experimente com vários tamanhos de redução das imagens originais, por exemplo, 50 x 50, 40 x 40, 30 x 30, 20 x 20. Escolha aquele que levar ao melhor compromisso (*tradeoff*) entre acurácia e custo computacional.

Obs. 2: Para estimar as matrizes de covariância das classes, podem ser usados comandos prontos disponíveis no Matlab/Octave e em bibliotecas e pacotes de outras linguagens de programação.

Obs. 3: Para o CQG, avaliar a invertibilidade das matrizes de covariância das classes através dos seus postos. Usar comandos `rank` e `rcond` no Octave/Matlab ou seus equivalentes em outras linguagens de programação.

Obs. 4: Em caso de problemas com a invertibilidade, usar a matriz de covariância agregada (*pooled*) ou o método de regularização de Friedman. Indicar na tabela apenas a opção que produzir melhor resultado.

Boa Sorte!