

UNIVERSIDADE REGIONAL DE BLUMENAU - INSTITUTO FURB

CURSO DE ESPECIALIZAÇÃO EM DATA SCIENCE

DISCIPLINA: Aprendizado de Máquina II - Aprendizado Não Supervisionado

PROFESSOR: Márcio Koch

Entrega: 17/01/2021

Trabalho Final

Aplicar a Análise das Componentes Principais (PCA) na base de dados ORL disponibilizada em aula (<https://github.com/lobokoch/unsupervised-learning/blob/main/dataset/ORL.rar>).

O objetivo é efetuar reconhecimento facial da mesma forma como foi feito em sala de aula, porém em uma linguagem diferente de Java.

Como método de amostragem, deve ser utilizado holdout de 70% das imagens para treinamento e 30% para os testes (as imagens tanto de treino, como de testes, devem ser sorteadas aleatoriamente).

Pode ser utilizada qualquer linguagem de programação, com exceção de Java, a qual já foi utilizada em aula (sugere-se Python). A implementação pode ser paço a paço conforme foi feito em aula, como o aluno pode utilizar bibliotecas específicas para PCA oferecidas pela linguagem escolhida, como Python e OpenCV, mas é opcional, fica a critério do aluno qual linguagem e biblioteca PCA utilizar.

O aluno deve acrescentar 10 imagens da sua própria face no conjunto de imagens já existentes ORL (caso não queira da sua própria face, pode ser de outra pessoa). As 10 novas imagens devem ser em tons de cinza e devem ser nomeadas de 401_41.jpg até 410_41.jpg. Onde 401 a 410 são os ids das imagens e 41 é o rótulo, ou classe desta pessoa. Essas 10 novas imagens devem ser entregues juntamente com o código como material para avaliação.

Como saída final do algoritmo de reconhecimento facial, deve ser exibido um **resumo com os resultados** da acurácia do reconhecimento das execuções utilizando desde as 10 componentes principais até as 20 componentes principais. Para cada iteração deve exibir uma linha de texto da seguinte forma (não deve ser nem mais nem menos):

k componentes principais, acurácia: n%.

Onde **k** é o número de componentes principais utilizadas na iteração e **n** é a acurácia obtida nessa iteração. A Acurácia será o total das imagens de teste reconhecidas corretamente pela implementação feita pelo aluno, dividido pela quantidade total de imagens de teste, multiplicado por 100. Segue exemplo de saída, exibindo as iterações da componente principal 10 até a 20:

10 componentes principais, acurácia: 90%.

11 componentes principais, acurácia: 94%.

12 componentes principais, acurácia: 95%.

13 componentes principais, acurácia: 95%.

14 componentes principais, acurácia: 95%.

15 componentes principais, acurácia: 95%.

16 componentes principais, acurácia: 95%.

17 componentes principais, acurácia: 95%.
18 componentes principais, acurácia: 96%.
19 componentes principais, acurácia: 96,5%.
20 componentes principais, acurácia: 98%.

Materiais que devem ser entregues no AVA:

O link para este projeto de avaliação no GitHub. O projeto no GitHub deve ser público, o professor deve conseguir baixá-lo.

O que deve ter no projeto do GitHub:

- Todo o código fonte, só baixar e rodar.
- As imagens da base ORL + as 10 imagens com as novas faces (bem importante).
- Um arquivo README.md explicando um pouco sobre o projeto, do que se trata, técnicas que utiliza, linguagem e bibliotecas que usa. Um passo a passo de como proceder para baixar e rodar o código do projeto.
- O resumo da acurácia obtida de uma execução completa utilizando das 10 até as 20 componentes principais, conforme exemplo descrito acima.