

## Tema 2 – Metode Numerice

Jugaru Marius-Gabriel 312CB

### Task1

Pentru acest task aplicam comanda SVD asupra imaginii, care returneaza matricele U, S, V. Din U luam primele m linii si primele k coloane, din S primele k linii si primele k coloane, iar din V primele n linii si primele k coloane. Astfel obtinem noua aproximare  $\text{new\_X} = U_{\text{red}} * S_{\text{red}} * V_{\text{red}}'$ .

### Task2

Normalizam fotografia prin scaderea mediei fiecarui rand din ea. Construim matricea Z care reprezinta transpusa fotografiei impartita la radicalul numarului de coloane. Calculam matricele U, S si V cu algoritmul SVD. Apoi construim matricea W care reprezinta primele m linii si primele pcs coloane din matricea V.

Transpusa acesteia este inmultita cu fotografia pentru a obtine matricea Y, de care ne folosim pentru a calcula noua aproximare  $\text{new\_X} = W * Y + \text{media}$ .

### Task3

Calculam media pentru fiecare vector, pe rand, si scadem o scadem din poza. Apoi calculam matricea de covarianta pentru care aflam vectorii si valorile proprii, folosind functia eig. Apoi ordonam descrescator valorile proprii si tinem minte pozitiile mutate pentru a actualiza si vectorii proprii. Pastram primele pcs coloane si realizam o schimbare de baza a matricei initiale. Calculam noua matrice  $\text{new\_X}$  si apoi ii adaugam media randurilor scazute anterior.

## Task4

### Prepare data

Pentru pregătirea datelor încarcăm în `d` datele și din `d` luăm în `X` matricea `trainX`, iar în `y` vectorul `Y`.

### Visualise image

Citim într-o variabilă linia number din matricea de antrenament și o modelăm într-o matrice pătratică de gradul 28.

### Magic with pca

Mai întâi calculăm media coloanelor din matricea de antrenament și apoi scădem această medie din matrice. Calculăm matricea de covarianță pentru care îi aflăm valorile și vectorii proprii. Aplicăm asupra valorilor o sortare și actualizăm vectorii proprii asemenea. Alegem doar primele `pcs` coloane din matricea obținută și efectuăm o schimbare de bază asupra matricei de antrenament. Într-un final calculăm matricea `train` care este o aproximare a matricei inițiale.

### Prepare photo

Inițializăm sirul final. Pentru a inversa pixelii unei imagini trebuie doar să scădem din 255 valoarea acestora.

### KNN

Calculăm distanța euclidiană a fiecărui rând și apoi sortăm crescător valorile distanțelor. Actualizăm asemenea label-urile datelor de antrenament și aplicăm predicția mediană primelor `k` valori.

### ClassifyImage

Determinam  $\mu$ ,  $\Sigma$  si  $V_k$  prin folosirea functiei implementate anterior, `magic_with_pca`. Scadem  $\mu$  din imagine si realizam o schimbare de baza inmultind imaginea cu  $V_k$ . Apoi apelam functia KNN cu label-urile, matricea  $Y$ , imaginea ca si matrice de test si  $k = 5$ .