

// Tudor Maria - Elena, 311CA

Tema 2 - Metode Numerice

Task 1

Primul task consta in compresia imaginilor folosind algoritmul SVD, care este un algoritm de factorizare ce descompune matricea imagine in trei matrici: U, S si V, ce contin valori si vectori singulare. Parametrul k controleaza la ce nivel se face compresia:

- k mai mic => compresie mai buna, dar se pot pierde detalii despre imagine
- k mai mare => pastreaza mai multe detalii, dar realizeaza o compresie mai mica.

Task 2

Al doilea task implementeaza o tehnica de reducere a dimensionalității și reconstrucție a imaginii utilizând analiza componentelor principale. Initial, imaginea este normalizata prin scaderea mediei fiecarui rand, avand ca scop centrarea datelor in jurul originii. Se aplica SVD, si se selecteaza componentele principale (primele pcs coloane ale matricei V), care au cea mai mare contributie la variatia datelor. In final, se reconstruieste matricea initiala, folosind noile matrici obtinute si media fiecarui rand.

Task 3

Al treilea task implementeaza, de asemenea, analiza componentelor principale, insa de data asta folosind matricea de covarianta. Dupa normalizarea matricei initiale, se calculeaza aceasta matrice ce prezinta relatiile dintre pixelii imaginii. Se calculeaza vectorii si valorile proprii ale acesteia si se sorteaza descrescator, in functie de importanta fiecărei componente principale. Se pastreaza primele 'pcs' coloane ale matricei V. Cu cat pcs este mai mare, cu atat imaginea rezultata va avea o claritate mai mare. In final, se reconstruieste matricea in mod asemanator cu metoda anterioara ce foloseste SVD pentru analiza componentelor principale.

Task 4

Functia prepare_data are rolul de a pregati datele pentru antrenarea unui model de recunoastere a imaginilor. Matricea train_mat stocheaza pixelii imaginilor de antrenament, iar vectorul train_val stocheaza etichetele imaginilor de antrenament.

Functia visualise_image are rolul de a vizualiza o imagine din setul de date de antrenament, prin selectarea liniei number din matricea de pixeli train_mat, si rearanjand valorile intr-o matrice patratica 28x28.

Functia magic_with_pca aplica analiza componentelor principale asupra matricei de antrenament, utilizand metoda care foloseste matricea de covarianta. Functia este asemanatoare cu cea de la task-ul 3 insa returneaza mai multi parametri, precum matricea train - aproximarea matricii initiale din setul de antrenament, miu - vectorul ce contine mediile pe fiecare coloana a train_mat, Vk - matricea primilor pcs vectori proprii sortati in ordinea descrescatoare a valorilor proprii ale matricei de covarianta, reprezentand componentele principale si Y - matricea obtinuta prin transformarea bazei matricei de antrenament utilizand Vk.

Functia prepare_photo inverseaza pixelii imaginii (negru devine alb si invers prin scaderea fiecărei valori din valoarea maxima a unui pixel(255, de obicei)) si o transforma intr-un vector linie.

Functia KNN implementeaza algoritmul K-Nearest Neighbors (KNN) pentru a realiza o predictie pe baza vecinilor cei mai apropiati. Se calculeaza distanta Euclidiana intre fiecare rand din matricea de antrenament Y si vectorul test. Apoi, sunt ordonate distantele în ordine crescatoare si pastrate primele k valori. In final, se calculeaza predictia ca mediana celor k valori cele mai apropiate.

Functia classify_image realizeaza clasificarea unei imagini utilizand setul de antrenament train_mat si train_val, pe care aplica algoritmul magic_with_pca. Se normeaza matricea imaginii initiale si se schimba baza inmultind cu matricea Vk obtinuta anterior si se calculeaza predictia folosind KNN.