

****Nume: SAMSON Alexandru-Valentin****

****Grupă: 312CC****

Tema 2 – Metode Numerice pentru funcții

Descriere:

Task 1

* Se aplică algoritmul de descompunere în valori singulare (SVD) asupra matricei "photo", rezultând trei matrice: "U", "S" și "V".

* Matricile "U", "S" și "V" sunt utilizate pentru a calcula matricile reduse: "U_k" (conține primele k coloane ale lui U), "S_k" (conține submatricea de dimensiune $k \times k$ din partea stânga-sus a lui S) și "V_k" (conține primele k coloane ale lui V).

* Aproximarea matricei inițiale "photo", denumită "new_X", este calculată prin înmulțirea matricilor "U_k", "S_k" și transpusa lui "V_k".

Task 2

* Se calculează media fiecărui rând al matricei "photo" și se stochează rezultatul în variabila "mean_row".

* Matricea "photo" este ajustată prin scăderea mediei de pe fiecare rând.

* Se construiește matricea "Z" ca transpusa matricei ajustate "photo".

* Se aplică descompunerea în valori singulare (SVD) matricei "Z", obținându-se trei matrici: "U", "S" și "V".

* Se construiește matricea "W" din primele "pcs" coloane ale matricei "V".

* Se calculează matricea "Y" ca produsul dintre transpusa matricei "W" și matricea ajustată "photo".

* Se aproximează matricea inițială, "new_X", prin înmulțirea matricelor "W" și "Y" și adăugarea mediei pe fiecare rând.

Task 3

* Se calculează media fiecărui rând al matricei "photo" și se stochează rezultatul în variabila "mean_row".

* Matricea "photo" este ajustată prin scăderea mediei de pe fiecare rând.

* Se calculează matricea de covarianță "Z" prin înmulțirea matricei ajustate "photo" cu transpusa sa și împărțirea rezultatului la $(n - 1)$, unde "n" reprezintă numărul de coloane al matricei.

* Se calculează vectorii și valorile proprii ale matricei de covarianță "Z" utilizând funcția "eig".

* Valorile proprii sunt ordonate în ordine descrescătoare, iar matricea "V" este construită astfel încât să conțină vectorii proprii corespunzători valorilor proprii ordonate.

* Se păstrează doar primele "pcs" coloane din matricea "V".

* Se construiește matricea "Y" prin înmulțirea transpusa matricei "V" cu matricea ajustată "photo".

* Se calculează matricea "new_X" care reprezintă aproximarea matricei inițiale prin înmulțirea matricelor "V" și "Y".

* Se adaugă înapoi media pe fiecare rând, care a fost anterior scăzută.

Task 4

* Se calculează media fiecărei coloane a matricei "train_mat" și rezultatul este stocat în variabila "miu".

* Matricea "train_mat" este ajustată prin scăderea mediei de pe fiecare coloană.

* Se calculează matricea de covarianță "cov_matrix" prin înmulțirea matricei ajustate "train_mat" cu transpusa sa și împărțirea rezultatului la $(m - 1)$, unde "m" reprezintă numărul de rânduri al matricei.

* Se calculează vectorii și valorile proprii ale matricei de covarianță "cov_matrix" utilizând funcția "eig".

* Valorile proprii sunt ordonate în ordine descrescătoare, iar matricea "V" este construită astfel încât să conțină vectorii proprii corespunzători valorilor proprii ordonate.

* Valorile proprii ordonate descrescător și indicii corespunzători sunt stocate în variabilele "sorted_eigenvalues" și "indices".

* Matricea "Vk" este actualizată cu primele "pcs" coloane ale matricei "V" în funcție de indicii sortați.

* Se păstrează doar primele "pcs" coloane din matricea "Vk".

* Se construiește matricea "Y" prin înmulțirea matricei ajustate "train_mat" cu matricea "Vk".

* Se calculează matricea "train" care reprezintă aproximarea matricei inițiale prin înmulțirea matricilor "Y" și "Vk" transpusă.

* Pentru fiecare rând în matricea "Y", se calculează distanța euclidiană între acel rând și vectorul de test primit ca argument. Distanțele sunt stocate în matricea "distance".

* Se ordonează distanțele în ordine crescătoare și se păstrează indicii corespunzători în variabila "indices".

* Se selectează primele "k" etichete (labels) asociate celor mai mici distanțe și se stochează în vectorul "k_labels".

* Se calculează predicția ca mediană a valorilor din vectorul "k_labels".

Comentarii asupra temei:

* Tema această mi-a aprofundat cunoștințele în Octave și m-a învățat mai multe metode de compresie a imaginilor și a recunoașterii cifrelor scrise de mână.

* Alegerea unui număr mai mic de componente principale poate duce la apariția unor artefacte sau pierderea detaliilor fine în imagine. În schimb, alegerea unui număr mai mare de componente principale poate păstra mai multă informație și detalii, dar poate duce și la o dimensiune mai mare a datelor comprimate.

Punctajul obținut la teste local:

* Total = [100 / 100]