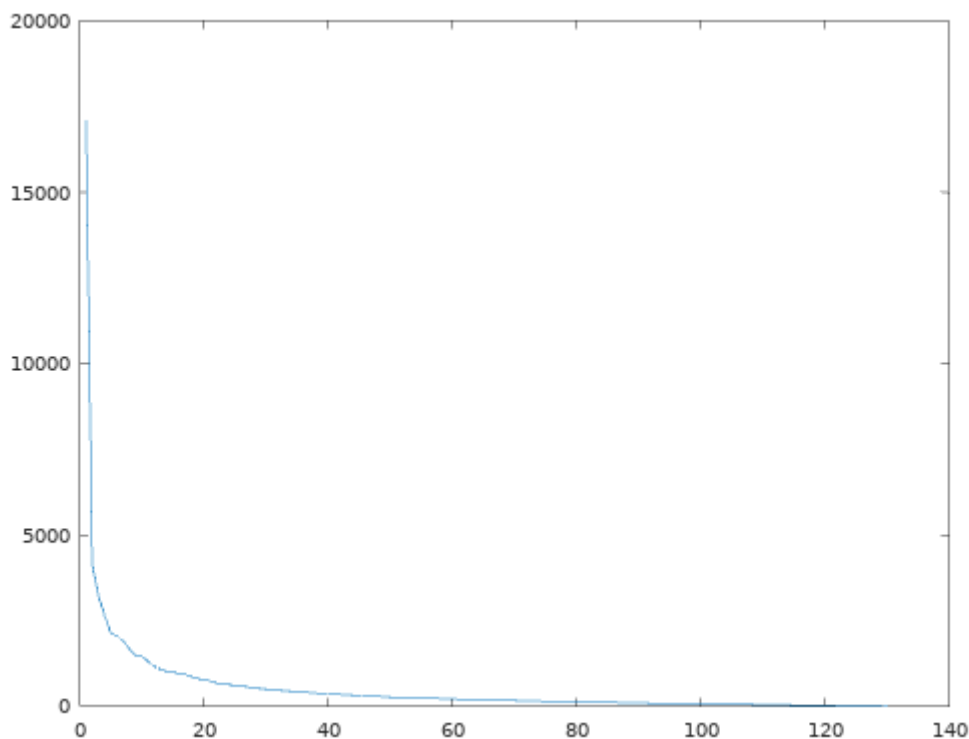


## README – Tema 2 -Metode Numerice

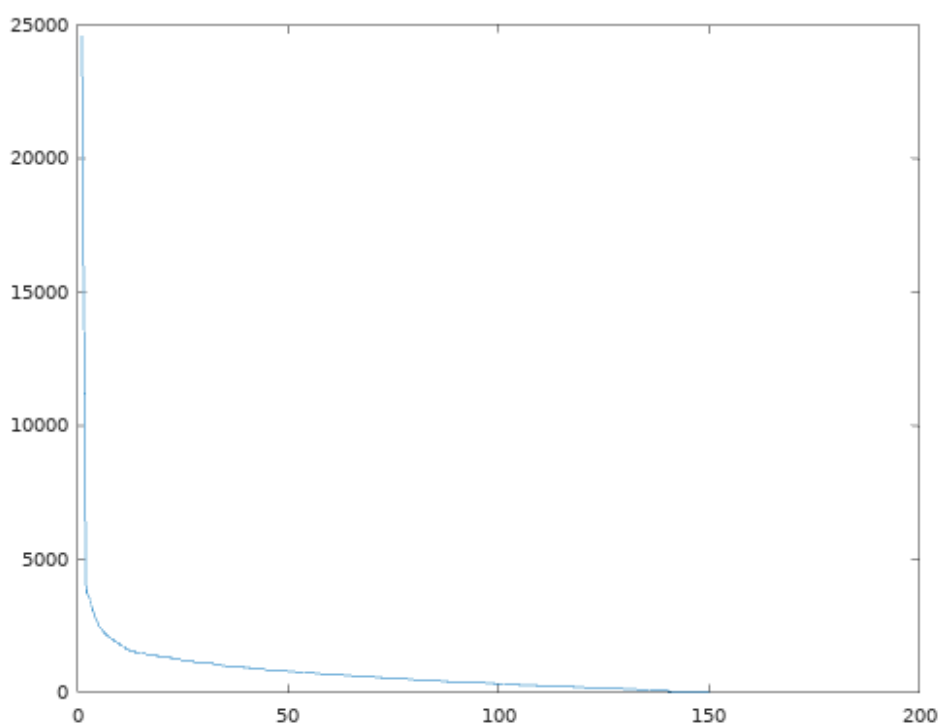
**Cerinta 1** – La aceasta cerinta am implementat descompunerea redusa a valorilor singulare. Am convertit imaginea data de calea image intr-o matrice **A** pe care am salvat-o folosind functia 'double(imread(image))'. Am apelat functia **svd** returnand matricile:U, S si V. Am transpus matricea V. Pe urma conform desenului de la pagina 5 am eliminat portiunea alba atribuind tuturor linilor/coloanelor incepand de la k+1 la end "=[]". La final am atribuit lui A\_k inmultirea celor 3 matrici modificate.

**Cerinta 2** – La inceputul acestei functii am dat adresa imaginii ca parametru de intrare . Am convertit imaginea data de calea image intr-o matrice **A** pe care am salvat-o folosind functia 'double(imread(image))'. Am definit k=[1:19 20:20:99 100:30:min(m,n)]. Pentru primul grafic am facut svd pe matricea **A** si am salvat diagonala matricei intr-un vector **x** caruia i-am dat plot .

**Graficul 1 pentru imaginile 3 :**

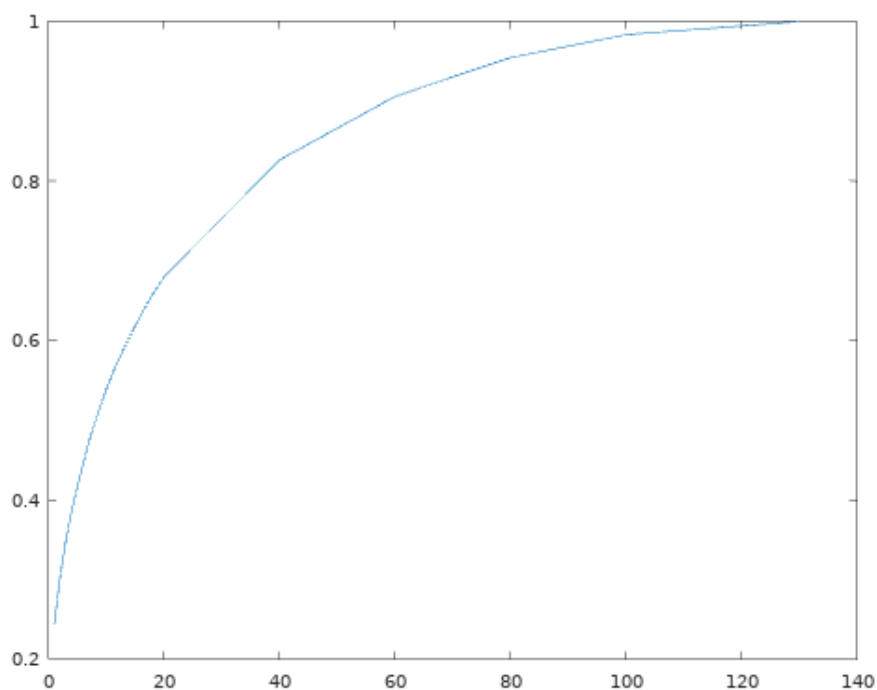


**Graficul 1 pentru imaginile 4 :**

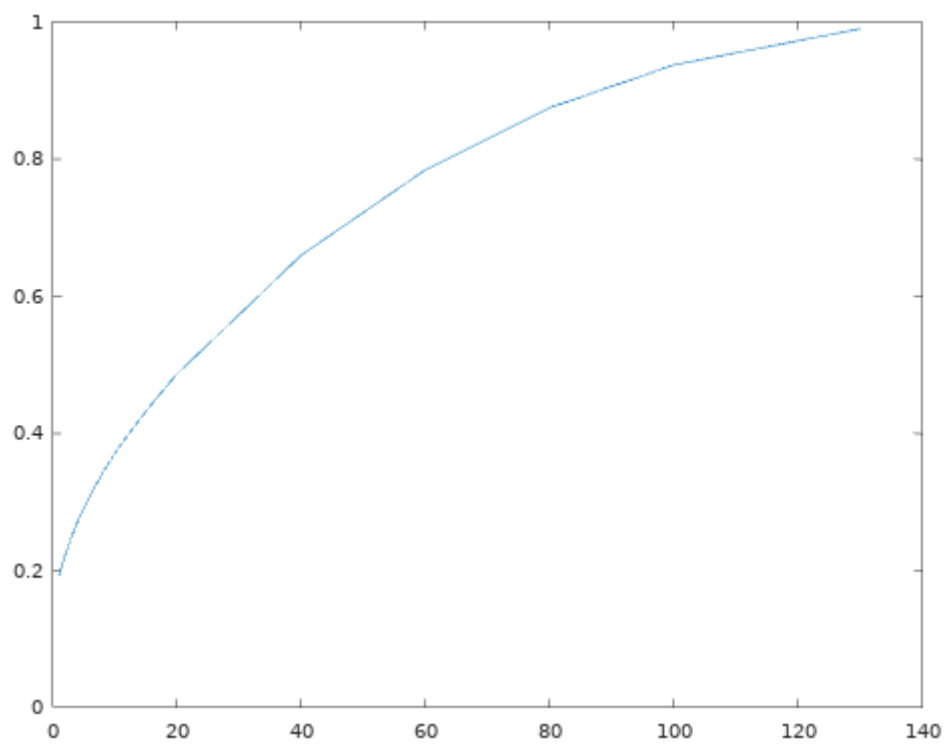


Pentru graficul 2 am facut un for care merge de la 1 la  $\text{length}(k)$  si am apelat pentru fiecare valoare  $k(i)$  functia `task1` de la cerinta 1 ,la fiecare grafic ce urmeaza am folosit acelasi for si acelasi apel al functiei `task1`,formam pe rand cele 2 sume ,iar apoi salvam raportul de fiecare data intr-un vector caruia ii dam `plot` dupa ce iesim din for alaturi de vectorul `k`.

**Graficul 2 pentru imaginea 3:**

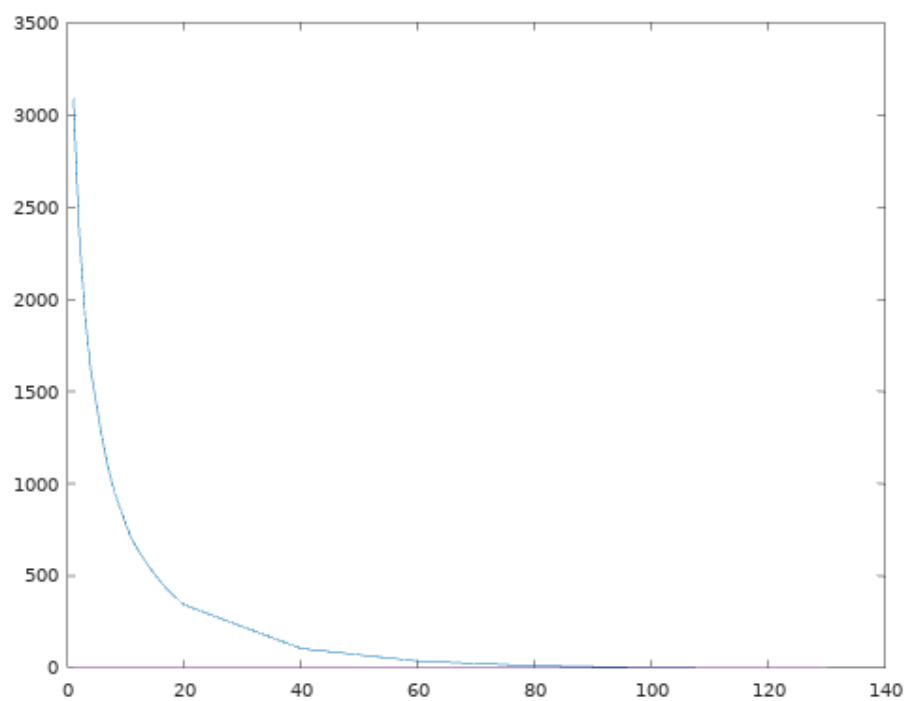


**Graficul 2 pentru imaginea 4:**

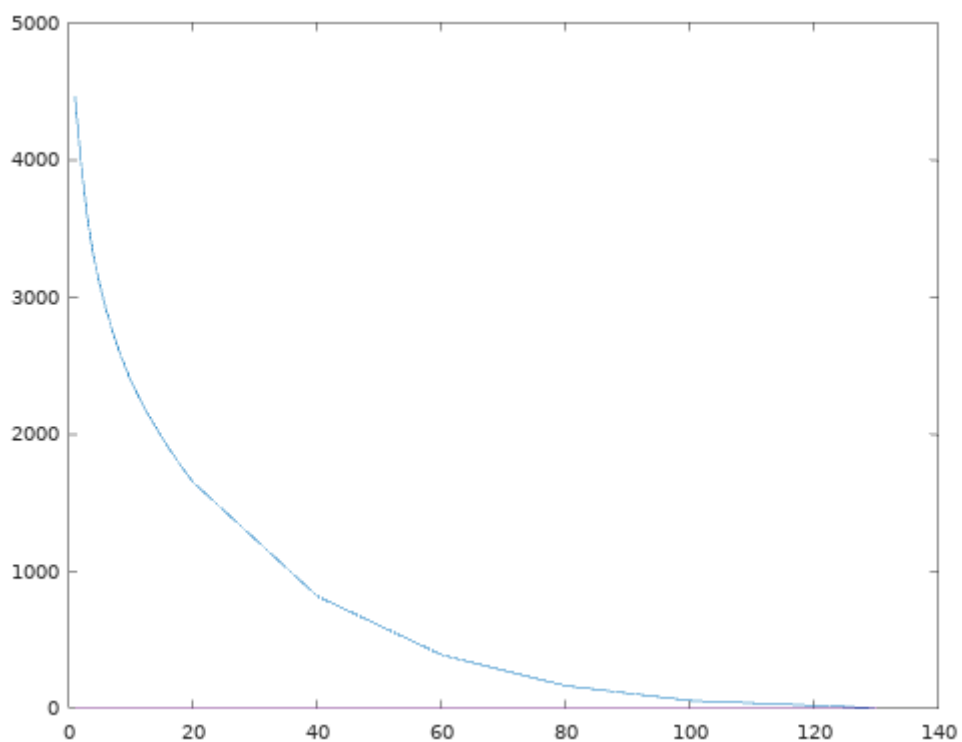


Pentru gaficul 3 facem sumele din formula in acelasi for ca la graficul 2 si salvam de fiecare data raportul dintre sume si  $m*n$  intr-un vector .La iesirea din for dam plot la vector si vectorul k .

**Graficul 3 pentru imaginea 3:**

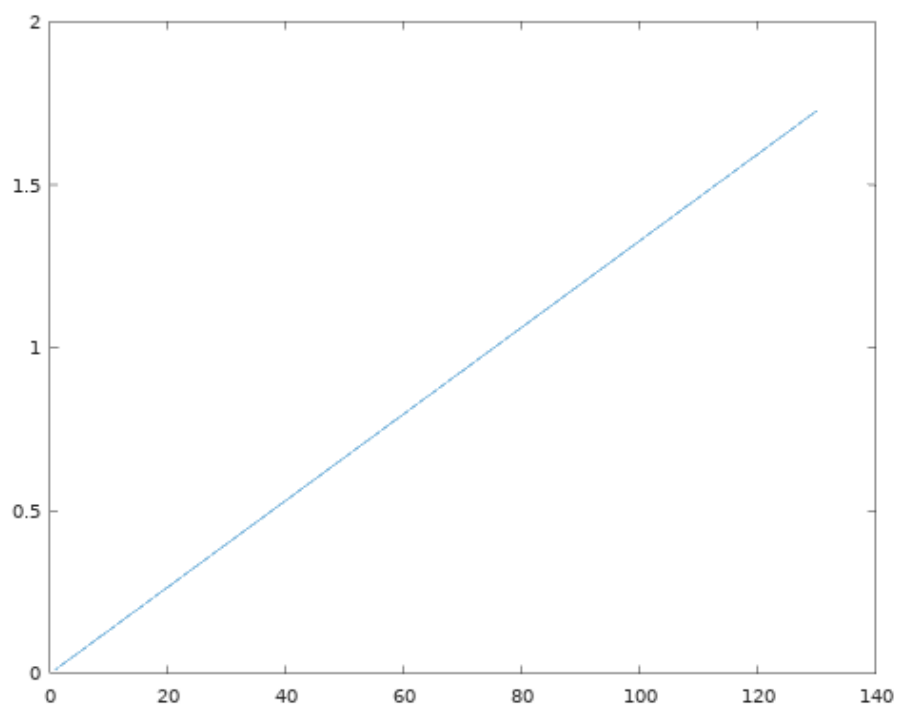


**Graficul 3 pentru imaginea 4:**

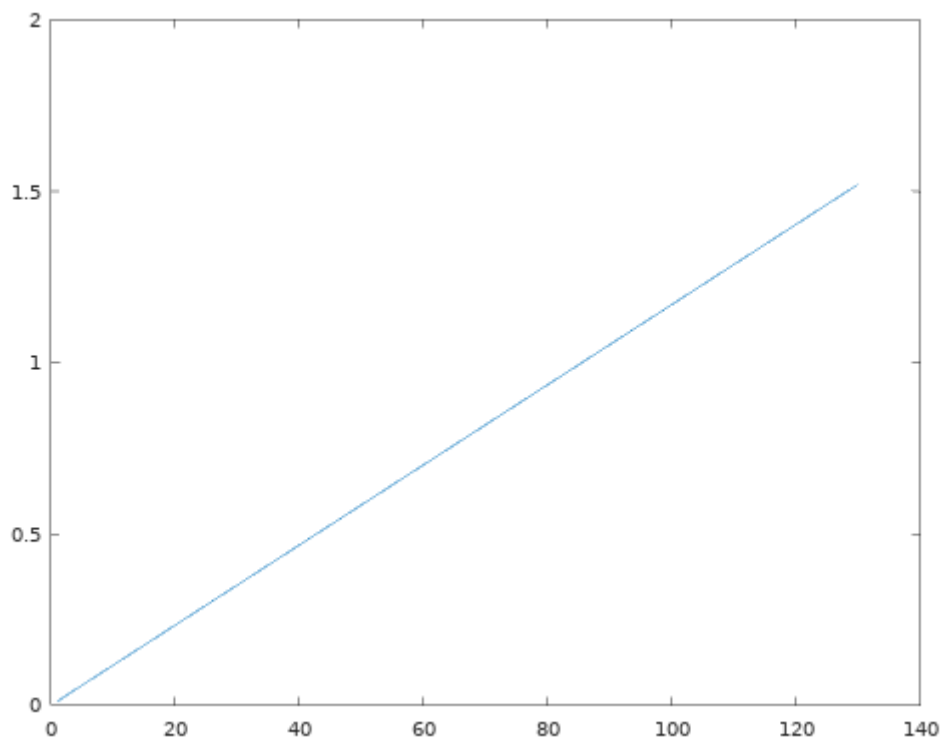


Pentru graficul 4 am implementat formula in acelsi for pentru  $k(i)$  si am salvat totul intr-un vector caruia i-am dat plot la finalul for-ului alturi de vectorul  $k$ .

**Graficul 4 pentru imaginea 3:**



**Graficul 4 pentru imaginea 4:**



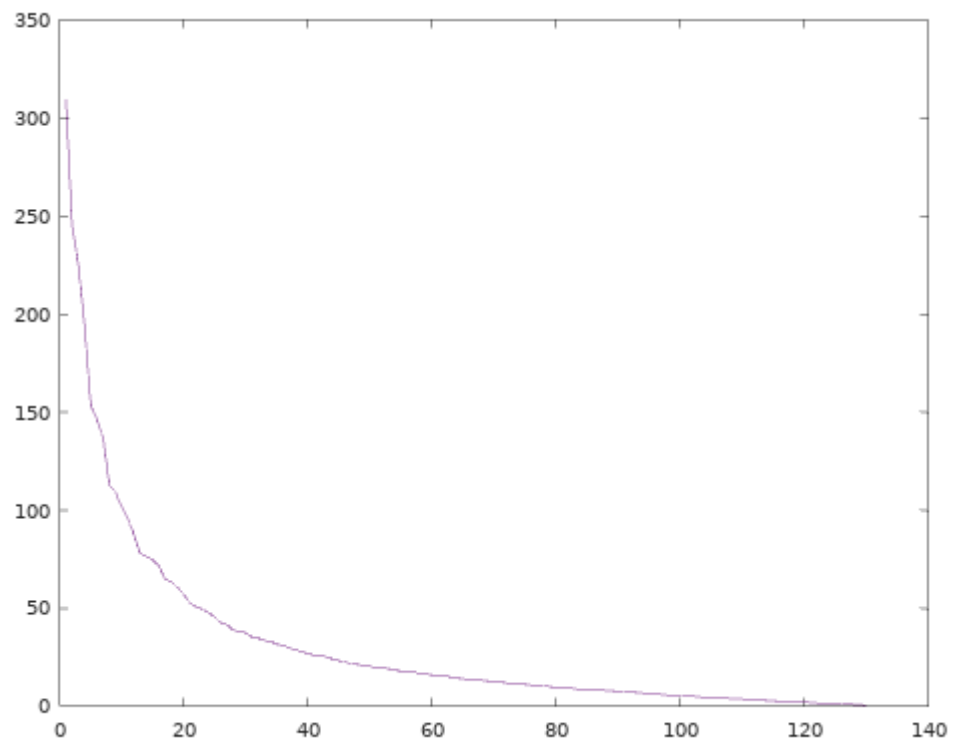
**Cerinta 3** – La aceasta cerinta am calculat media pt fiecare vector  $\mathbf{a}_i$  si am salvat intr-un vector  $\mathbf{u}$ . Transpun vectoru  $\mathbf{u}$  ca sa fie un vector coloana .Actualizez valoarea lui  $\mathbf{a}_i$  dupa formula  $\mathbf{a}_i = \mathbf{a}_i - \mathbf{u}_i$ . Construiesc matricea  $\mathbf{Z}$  dupa formula .Aplic **svd** pentru matricea  $\mathbf{Z}$  .Formez spatiul k-dimensional al componentelor principale salvandu-le in matricea  $\mathbf{W}$ . Calculez proiectia lui  $\mathbf{A}$  pe spatiul componetelor principale. Aproximez matricea initiala.

**Cerinta 4** – La aceasta cerinta am calculat media pt fiecare vector  $\mathbf{a}_i$  si am salvat intr-un vector  $\mathbf{u}$ . Transpun vectoru  $\mathbf{u}$  ca sa fie un vector coloana .Actualizez valoarea lui  $\mathbf{a}_i$  dupa formula  $\mathbf{a}_i = \mathbf{a}_i - \mathbf{u}_i$ . Construiesc matricea  $\mathbf{Z}$  dupa formula .Aplic functia **eig** pe matricea  $\mathbf{Z}$  . Formez spatiul k-dimensional al componentelor principale salvandu-le in matricea  $\mathbf{W}$ . Calculez proiectia lui  $\mathbf{A}$  pe spatiul componetelor principale. Aproximez matricea initiala.

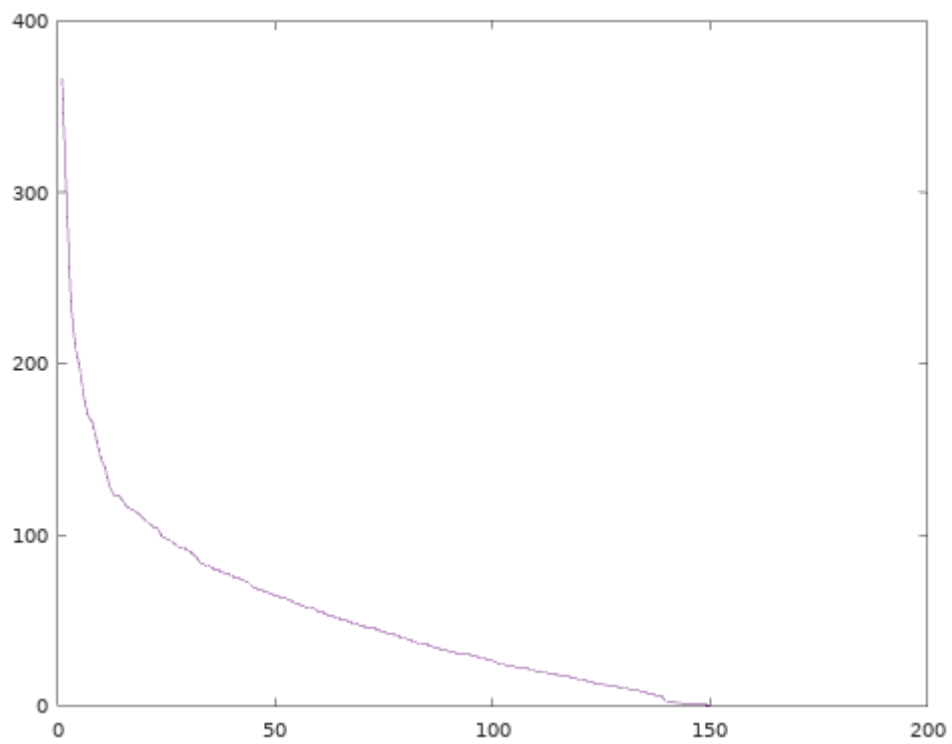
**Cerinta 5** – La inceputul acestei functi am dat adresa imaginii ca parametru de intrare . Am convertit imaginea data de calea image intr-o matrice  $\mathbf{A}$  pe care am salvat-o folosind functia 'double(imread(image))' .Am definit  $k=[1:19\ 20:20:99\ 100:30:\min(m,n)]$ ;

Pentru primul grafic am facut un for care merge de la 1 la length(k) si am apelat in interiorul acestuia functia task3 de la cerinta 3, in interiorul acestui for fac toate graficele. Graficul 1 il formez punct cu punct dand plot pt fiecare  $k(i)$  la **diag(S)** alaturi de **hold on** .

**Graficul 1 pentru imaginea 3:**

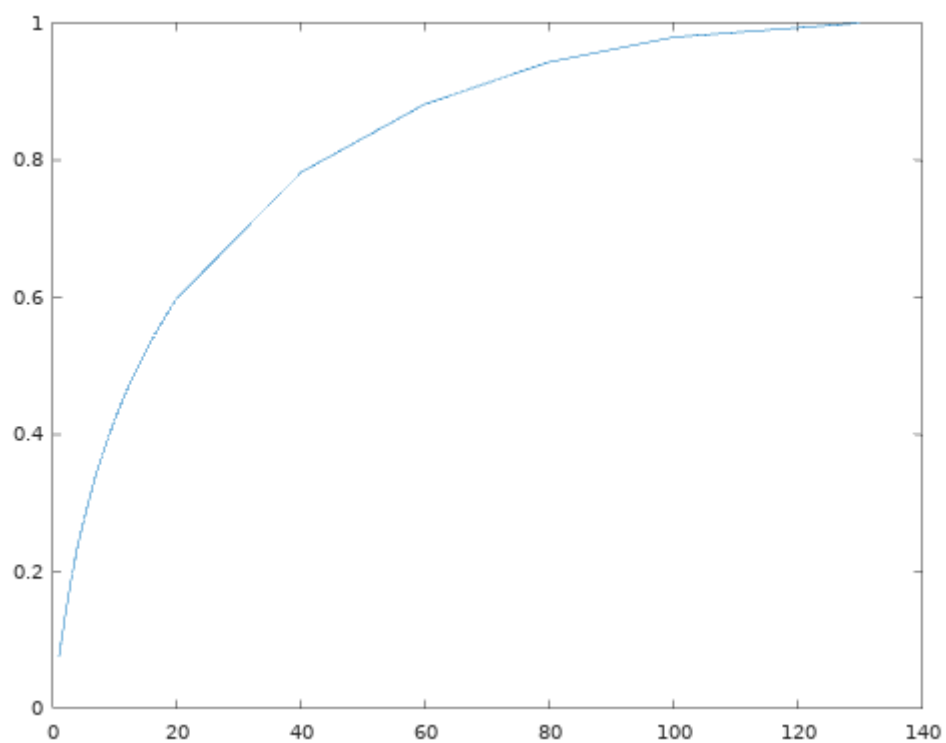


**Graficul 1 pentru imaginea 4:**

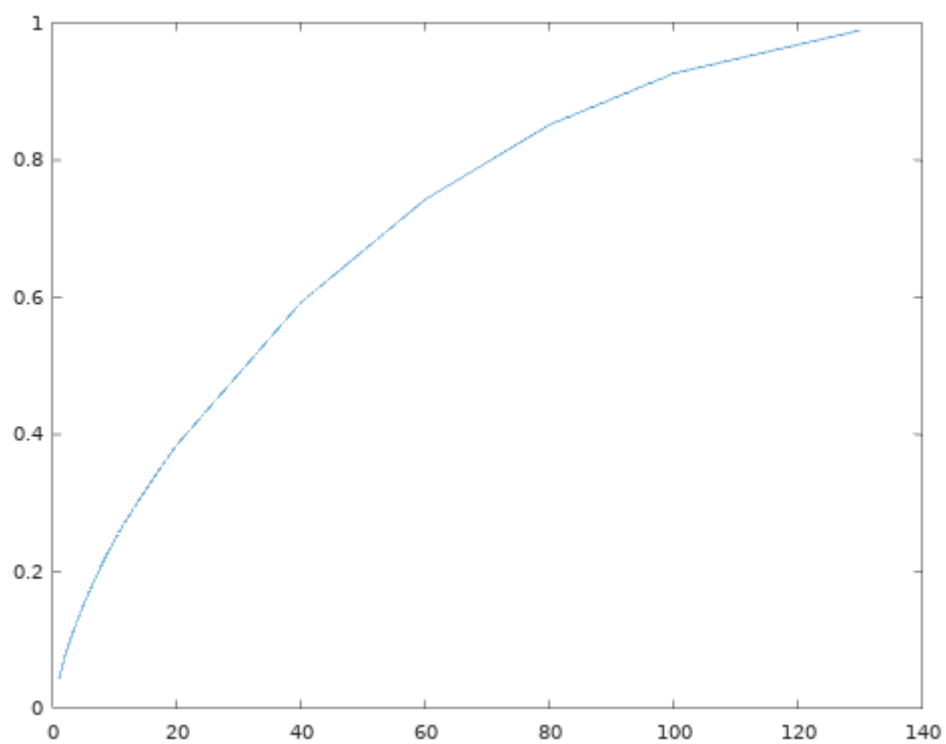


Pentru graficul 2 ,in acelasi for ca la grafic 1 , fac cele 2 sume din formula si pun raportul lor intr-un vector caruia ii dau plot la iesirea din for alaturi de vectorul k.

**Graficul 2 pentru imaginea 3:**

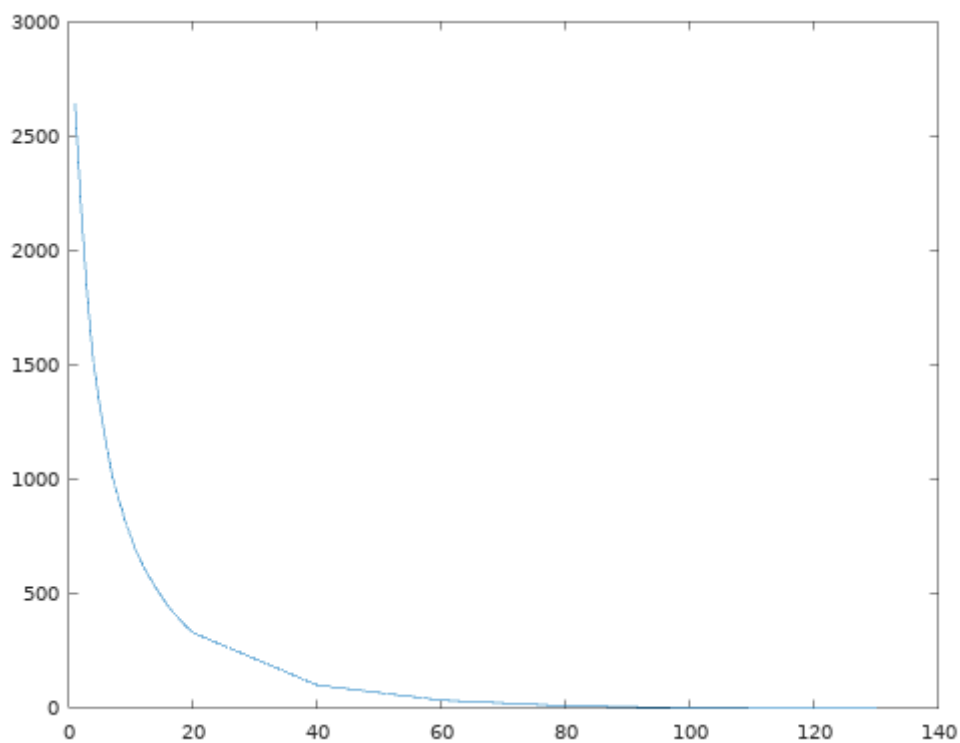


**Graficul 2 pentru imaginea 4:**

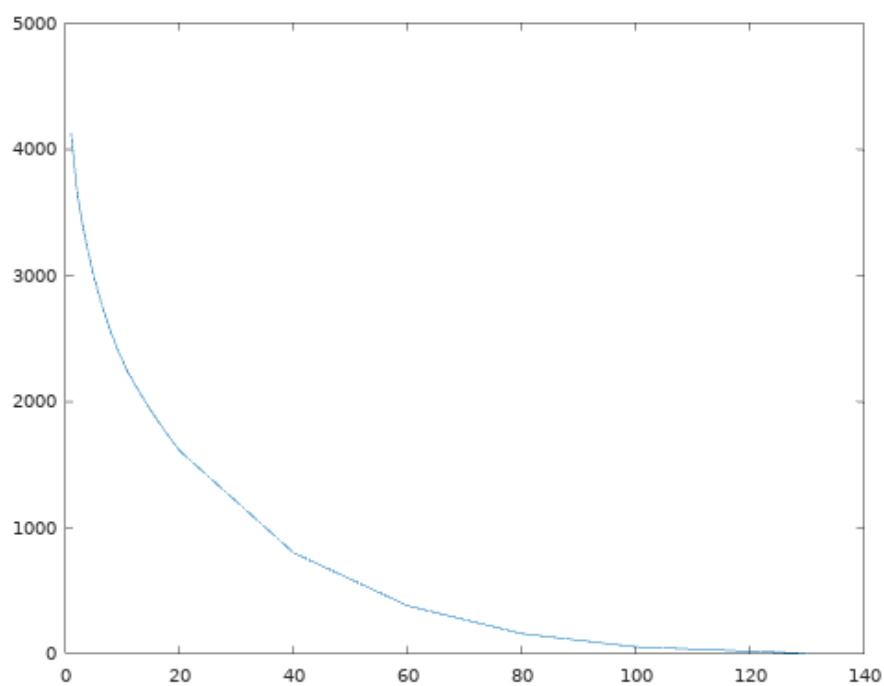


Pentru graficul 3 facem sumele din formula in acelasi for ca la graficul 2 si salvam de fiecare data raportul dintre sume si  $m*n$  intr-un vector .La iesirea din for dam plot la vector si vectorul k.

**Graficul 3 pentru imaginea 3 :**



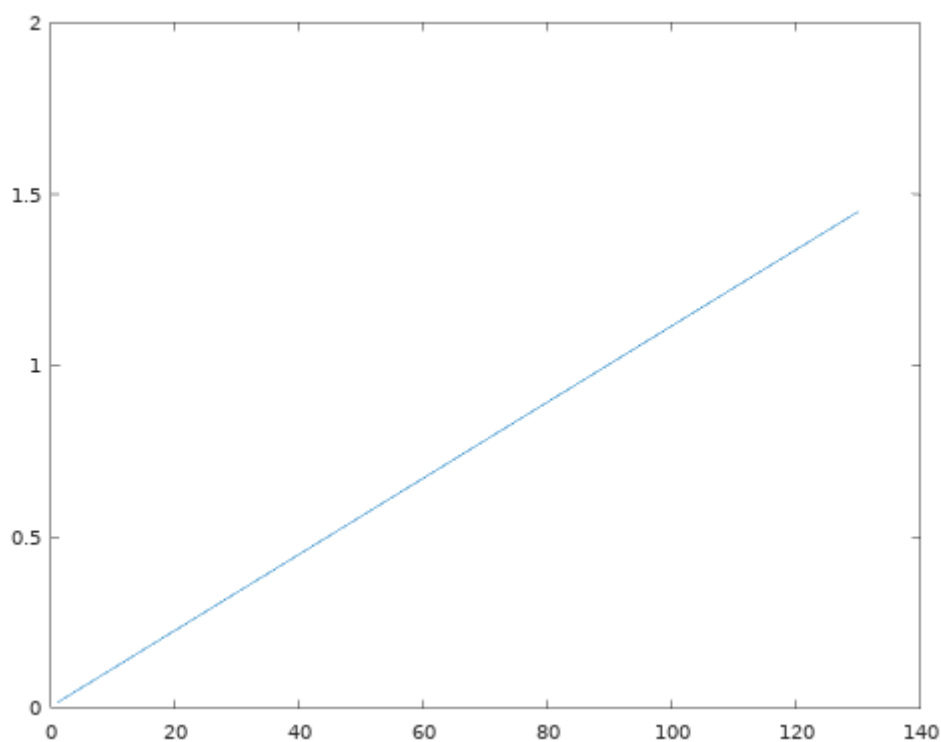
**Graficul 3 pentru imaginea 4 :**



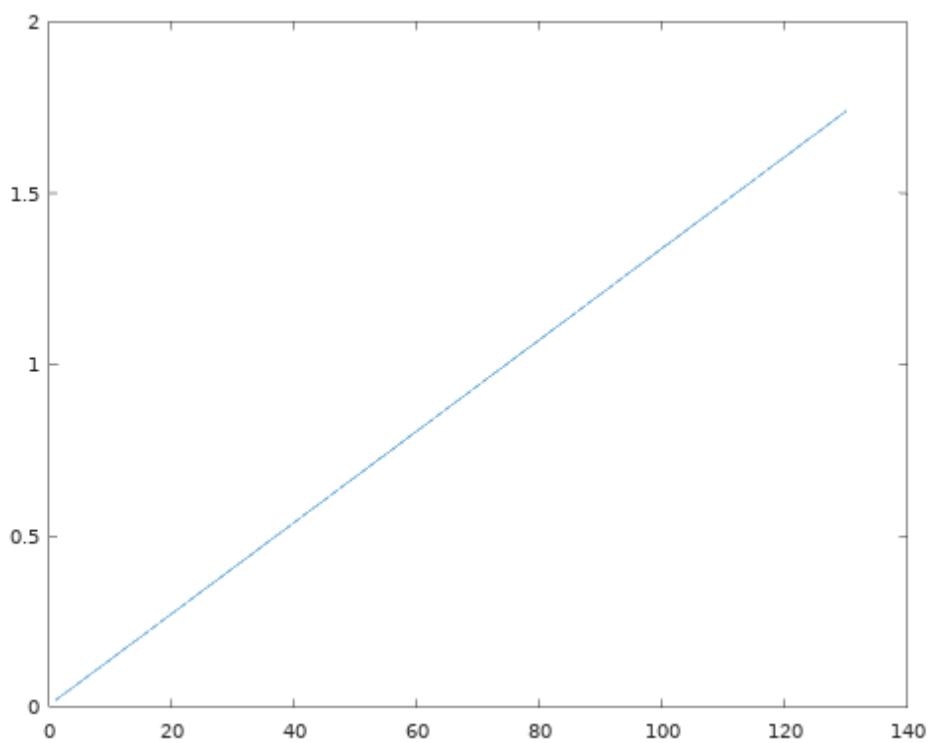


Pentru graficul 4 am implementat formula in acelasi for pentru  $k(i)$  si am salvat totul intr-un vector caruia i-am dat plot la finalul for-ului alturi de vectorul  $k$ .

**Graficul 4 pentru imaginea 3:**



**Graficul 4 pentru imaginea 4:**



**Cerinta 6** – La aceasta cerinta am implementat-o in 2 functii **eigenface\_core** si **face\_recognition**

In prima functie citesc intr-un for de la 1 la 10 imaginile ,le transform in vectori coloana pe care ii pun in matricea T. Fac media pe lini utilizand functia **mean(T')** si o salvez in m .Calculez matricea A ca fiind diferenta dintre T si m' (m initial e un vector linie ) .Salvez in matricea B  $A^*A$  si apelez functia eig si construiesc o matrice V care contine vectorii proprii mai mari ca 1 .Calculez matricea cu fetele proprii ca produs dintre  $A*V$ .Calculez proiectia ca produs dintre **eigenfaces'** $*A$  .

In a doua functie citesc imaginea fetei si o transform in vector coloana si scad m' .Calculez proiectia imaginii de test in spatiul fetelor .Calculez cea mai mica distanta intre **pr\_test\_image** si **pr\_img** .Distanta intre vectori o calculez cu norma diferentei dintre ei pe care o pun intr-un vector vect .La final apelez functia **min** pentru determinarea celei mai mici distante din vect si indicele imaginii.