

SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO-MATEMATIČKI FAKULTET
MATEMATIČKI ODSJEK

**Klasifikacija objekata pomoću
HOSVD-a**

Ante Čubela, Mirna Lovrić i Ermin Mustafić

Zagreb
17. veljače 2023.

Sadržaj

Uvod	0
1 Osnovno o tenzorima	1
2 HOSVD	1
3 Algoritam I	3
4 Algoritam II	4
5 Primjene	5
Literatura	10

Uvod

Već smo se upoznali s vektorima i matricama te njihovim brojnim primjenama u matematičkom modeliranju raznih pojava. Jedna zanimljiva primjena matrica jest modeliranje slike. Na primjer, crno-bijelu sliku rezolucije 1920×1080 reprezentiramo matricom iz prostora $\mathbb{R}^{1920 \times 1080}$, gdje a_{ij} označava numeričku vrijednost određenog piksela. Sljedeći korak je modeliranje videozapisa kao konačnog niza slika. Recimo da imamo videozapis u 60 FPS-a, također rezolucije 1920×1080 , koji traje jednu minutu. Tada dobivamo 3600 matrica dimenzija 1920×1080 . U tom se slučaju prirodno nameću tenzori kao trodimenzionalne matrice, odnosno kvadri ako matrice doživljavamo kao pravokutnike.

U ovom radu ćemo prezentirati dva algoritma za klasifikaciju objekata. Oba koriste HOSVD (High Order SVD). Prvi algoritam koristi HOSVD za računanje baznih matrica koje razapinju dominantan podprostor za određenu klasu. Kod znamenaka ćemo imati 10 tenzora jer za svaku znamenku radimo poseban tenzor na kojeg se primjeni HOSVD. Kod drugog algoritma prvo vektoriziramo podatke te sve posložimo u jedan tenzor. HOSVD računamo samo jednom.

1 Osnovno o tenzorima

Općenito, *tenzor* reda N je objekt s N indeksa. Prostor tenzora reda N označavamo s $\mathbb{R}^{I_1 \times \dots \times I_N}$. U ovom će radu biti razmatrani samo tenzori reda 3.

Neka su $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{I \times J \times K}$. Definiramo *skalarni produkt*

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^K a_{ijk} b_{ijk}.$$

Za tenzore \mathcal{A} i \mathcal{B} kažemo da su *ortogonalni* ako je

$$\langle \mathcal{A}, \mathcal{B} \rangle = 0.$$

Norma tenzora \mathcal{A} je definirana kao

$$\|\mathcal{A}\| = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}.$$

Slično kao što imamo vektorizaciju matrice, uvodimo matricizaciju tenzora. Za to su nam potrebne *niti u modu l* , $l \in \{1, 2, 3\}$, koje predstavljaju dio tenzora u kojem su svi indeksi fiksirani osim l -tog. *Odsječak* je dio tenzora u kojem je fiksiran samo jedan indeks. Postoje horizontalni, bočni i frontalni odsjecci. Matricizacija tenzora u n -tom modu $\mathcal{A}_{(n)}$ se dobije tako da se niti u n -tom modu poslože kao stupci matrice u nekom unaprijed zadanom fiksnom poretku. Preciznije, za tenzor $\mathcal{A} \in \mathbb{R}^{I \times m \times n}$ definiramo

$$\begin{aligned} \mathcal{A}_{(1)} &= \text{unfold}_1(\mathcal{A}) = [\mathcal{A}(:, 1, :) \dots \mathcal{A}(:, m, :)], \\ \mathcal{A}_{(2)} &= \text{unfold}_2(\mathcal{A}) = [\mathcal{A}(:, :, 1)^T \dots \mathcal{A}(:, :, n)^T], \\ \mathcal{A}_{(3)} &= \text{unfold}_3(\mathcal{A}) = [\mathcal{A}(1, :, :)^T \dots \mathcal{A}(l, :, :)^T]. \end{aligned}$$

Definicija 1.1. (Tenzorsko matrično množenje u n -tom modu) Za tenzor $\mathcal{T} \in \mathbb{R}^{J_1 \times \dots \times J_{n-1} \times J_n \times J_{n+1} \times \dots \times J_N}$ i matricu $M \in \mathbb{R}^{I \times J_n}$ definiramo $\mathcal{T} \times_n M \in \mathbb{R}^{J_1 \times \dots \times J_{n-1} \times I \times J_{n+1} \times \dots \times J_N}$ za kojeg vrijedi

$$(\mathcal{T} \times_n M)_{j_1 \dots j_{n-1} i j_{n+1} \dots j_N} = \sum_{j=1}^{J_n} a_{ij} t_{j_1 \dots j_{n-1} j j_{n+1} \dots j_N}, \quad i = 1, \dots, I.$$

2 HOSVD

High Order Singular Value Decomposition je generalizacija matričnog SVD-a na tenzore.

Teorem 2.1. (HOSVD) Za tenzor $\mathcal{A} \in \mathbb{R}^{I \times J \times K}$ postoji tenzor $\mathcal{S} \in \mathbb{R}^{I \times J \times K}$ te ortogonalne matrice $U \in \mathbb{R}^{I \times I}$, $V \in \mathbb{R}^{J \times J}$, $W \in \mathbb{R}^{K \times K}$ takve da je

$$\mathcal{A} = \mathcal{S} \times_1 U \times_2 V \times_3 W.$$

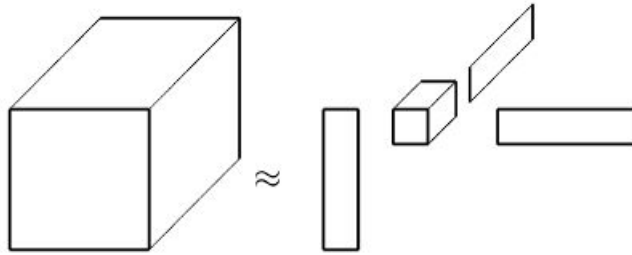
Nadalje, vrijedi svojstvo potpune ortogonalnosti, tj.

$$\begin{aligned}\langle \mathcal{S}(i, :, :), \mathcal{S}(j, :, :) \rangle &= 0, \\ \langle \mathcal{S}(:, i, :), \mathcal{S}(:, j, :) \rangle &= 0, \\ \langle \mathcal{S}(:, :, i), \mathcal{S}(:, :, j) \rangle &= 0,\end{aligned}$$

za sve $i \neq j$. Singularne vrijednosti u modu 1 su

$$\sigma_i^{(1)} = \|\mathcal{S}(i, :, :)\|, \quad i = 1, \dots, l$$

i uređene su u padajućem poretku. Analogno se definiraju singularne vrijednosti u ostalim modovima.



Slika 2.1 Ilustracija HOSVD-a.

Norme $\sigma_i^{(n)}$ u prethodnom teoremu su zapravo singularne vrijednosti matrica $\mathcal{A}_{(n)}$ te vrijedi

$$\mathcal{S} = \mathcal{A} \times_1 U^T \times_2 V^T \times_3 W^T.$$

Algoritam 2.2. (Računanje HOSVD-a)

1. `[U1, ~, ~] = svd(unfold(A, 1))`,
2. `[U2, ~, ~] = svd(unfold(A, 2))`,
3. `[U3, ~, ~] = svd(unfold(A, 3))`,
4. `S = ten_mat_m(ten_mat_m(ten_mat_m(A, U1', 1), U2', 2), U3', 3)`.

Kao i kod matrica, HOSVD služi za aproksimaciju tenzora nižim rangom. Odnosno $\tilde{\mathcal{A}} \approx \mathcal{A}$, gdje je

$$\tilde{\mathcal{A}} = \tilde{\mathcal{S}} \times_1 \tilde{U} \times_2 \tilde{V} \times_3 \tilde{W},$$

$$\tilde{\mathcal{S}} \in \mathbb{R}^{\tilde{I} \times \tilde{J} \times \tilde{K}}, \quad \tilde{U} \in \mathbb{R}^{I \times \tilde{I}}, \quad \tilde{V} \in \mathbb{R}^{J \times \tilde{J}}, \quad \tilde{W} \in \mathbb{R}^{K \times \tilde{K}} \quad \text{za } \tilde{I} \leq I, \quad \tilde{J} \leq J, \quad \tilde{K} \leq K.$$

Vrijedi rastav na ortonormirane bazne matrice u modu 3:

$$\mathcal{A} = \sum_{k=1}^K \mathcal{A}_k \times_3 w_k,$$

gdje je $\mathcal{A}_k = \mathcal{S}(:, :, k) \times_1 U \times_2 V$, a w_k k -ti stupac matrice W .

3 Algoritam I

Trening faza. Promatramo N sličnih objekata. Svaki objekt tipa n , $n = 1, \dots, N$, je dimenzije $I \times J$ te ih ima K_n . Prvo napravimo tenzor $\mathcal{A}^{(n)} \in \mathbb{R}^{I \times J \times K_n}$, zatim izračunamo HOSVD tog tenzora. To znači da je svaki objekt u $\mathcal{A}^{(n)}$ jedinstvena linearna kombinacija ortonormiranih baznih matrica $\mathcal{A}_k^{(n)}$, $k = 1, \dots, K_n$. Ako singularne vrijednosti značajno opadaju nakon nekog indeksa \tilde{K}_n , uzimat ćemo u obzir samo prvih \tilde{K}_n baznih matrica. Očito je $\tilde{K}_n \leq IJ$ jer je $\dim \mathbb{R}^{I \times J} = IJ$.

Test faza. Novi objekt $X \in \mathbb{R}^{I \times J}$ prvo normaliziramo, potom metodom najmanjih kvadrata računamo najmanju udaljenost između potprostora $[\{\mathcal{A}_1^{(1)}, \dots, \mathcal{A}_{\tilde{K}_1}^{(1)}\}], \dots, [\{\mathcal{A}_1^{(N)}, \dots, \mathcal{A}_{\tilde{K}_N}^{(N)}\}]$. Drugim riječima, minimiziramo

$$\begin{aligned} f_n(x_1^{(n)}, \dots, x_{\tilde{K}_n}^{(n)}) &= \frac{1}{2} \|X - \sum_{k=1}^{\tilde{K}_n} x_k^{(n)} \mathcal{A}_k^{(n)}\|^2 \\ &= \frac{1}{2} \left\langle X - \sum_{k=1}^{\tilde{K}_n} x_k^{(n)} \mathcal{A}_k^{(n)}, X - \sum_{k=1}^{\tilde{K}_n} x_k^{(n)} \mathcal{A}_k^{(n)} \right\rangle \\ &= \frac{1}{2} \|X\|^2 - \sum_{k=1}^{\tilde{K}_n} x_k^{(n)} \langle X, \mathcal{A}_k^{(n)} \rangle + \frac{1}{2} \sum_{k=1}^{\tilde{K}_n} (x_k^{(n)})^2 \|\mathcal{A}_k^{(n)}\|^2. \end{aligned}$$

Nadalje, vrijedi

$$\frac{\partial f_n(x_1^{(n)}, \dots, x_{\tilde{K}_n}^{(n)})}{\partial x_l^{(n)}} = x_l^{(n)} \|\mathcal{A}_l^{(n)}\|^2 - \langle X, \mathcal{A}_l^{(n)} \rangle.$$

Iz teorije optimizacije znamo da je jedinstveni minimum točka

$$\hat{x}^{(n)} = \left(\langle X, \mathcal{A}_1^{(n)} \rangle, \dots, \langle X, \mathcal{A}_{\tilde{K}_n}^{(n)} \rangle \right)$$

jer je $\|\mathcal{A}_k^{(n)}\| = 1$ za svaki $k = 1, \dots, \tilde{K}_n$. Konačno, objekt X pripada klasi $\hat{n} = \underset{n}{\operatorname{argmin}} f_n(\hat{x}^{(n)})$.

Algoritam 3.1. (Algoritam I)

Trening faza:

1. sortirati zadane podatke u tenzore istih tipova,
2. izračunati HOSVD tih tenzora,
3. izračunati i spremiti bazne matrice $(\mathcal{A}_k^{(n)})_{k=1}^{\tilde{K}_n}$, $n = 1, \dots, N$.

Test faza:

4. normalizirati ulazni podatak,
5. izračunati $\hat{x}^{(n)}$, $n = 1, \dots, N$,
6. $\hat{n} = \underset{n}{\operatorname{argmin}} f_n(\hat{x}^{(n)})$ je tražena klasa.

4 Algoritam II

Trening faza. Podatke iz $\mathbb{R}^{I \times J}$ vektoriziramo u stupce. Za ovaj algoritam je bitno imati isti broj zadanih podataka, odnosno zahtijevamo da je $K_1 = \dots = K_N$. Radi jednostavnosti ćemo taj broj označavati s K . Sve složimo u jedan tenzor $\mathcal{D} \in \mathbb{R}^{IJ \times K \times N}$. Nakon računanja samo jednog HOSVD-a dobivamo da je

$$\mathcal{D} = \mathcal{S} \times_1 U \times_2 V \times_3 W \approx \mathcal{F} \times_1 U_p \times_2 V_q,$$

gdje je $U_p = U(:, 1 : p)$, $V_q = V(:, 1 : q)$ te $\mathcal{F} = \mathcal{S}(1 : p, 1 : q, :) \times_3 W$. Ovdje smo, za razliku od prvog algoritma, reducirali i dimenziju samih ulaznih podataka. Drugim riječima, gledamo njihovu projekciju na potprostor razapet stupcima matrice U_p . Odnosno, stupci u tenzoru

$$\mathcal{D}_p = \mathcal{D} \times_1 U_p^T = \mathcal{F} \times_2 V_q$$

reprezentiraju neki objekt s p elemenata, a frontalni odsječci u \mathcal{F} sadrže q baznih vektora za različite klase. Njih računamo preko SVD-a. Preciznije, imamo

$$F^{(n)} = \mathcal{F}(:, :, n) = [B^{(n)}(B^{(n)})^\perp] \Sigma^{(n)}(Q^{(n)})^T, \quad n = 1, \dots, N,$$

gdje je $B^{(n)} \in \mathbb{R}^{p \times k}$, $k \leq p$.

Test faza. Ulazni podatak $d \in \mathbb{R}^{IJ}$ projiciramo na nižedimenzionalni potprostor i dobivamo $d_p = U_p^T d$. Ponovno riješavamo, ovaj put puno manji, problem najmanjih kvadrata, tj. tražimo $\hat{x}^{(n)}$ takav da je

$$\hat{x}^{(n)} = \underset{x^{(n)}}{\operatorname{argmin}} \|d_p - B^{(n)}x^{(n)}\|.$$

Jer je $B^{(n)}$ ortogonalna, slijedi da je

$$\hat{x}^{(n)} = (B^{(n)})^T d_p.$$

Kao i ranije, d pripada onoj klasi \hat{n} za koju je greška najmanja.

Algoritam 4.1. (Algoritam II)

Trening faza:

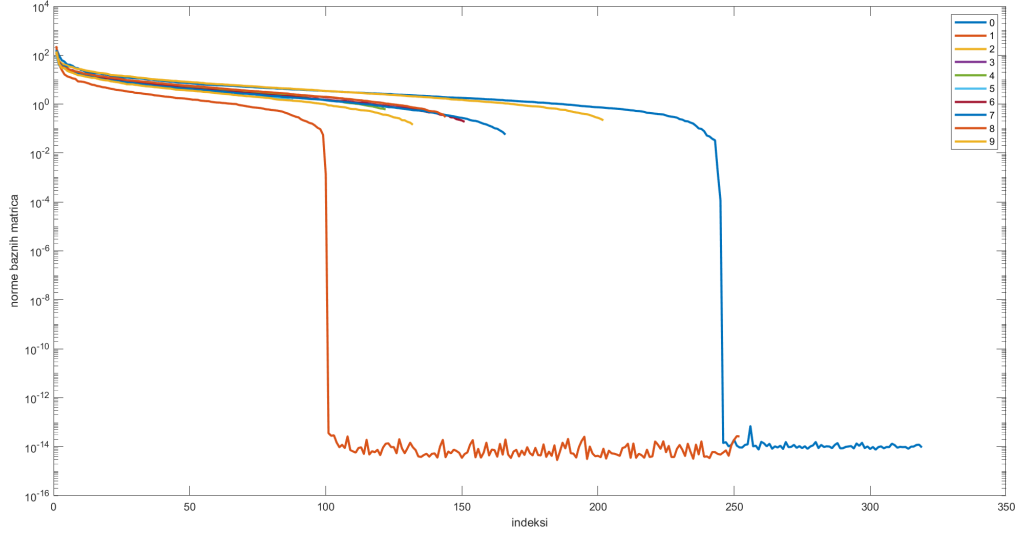
1. vektorizirati i posložiti podatke u tenzor \mathcal{D} ,
2. izračunati HOSVD tenzora \mathcal{D} ,
3. izračunati reducirani tenzor \mathcal{F} ,
4. izračunati i spremiti bazne matrice $B^{(n)}$ za svaku klasu.

Test faza:

5. reducirati dimenziju ulaznog podatka d ,
6. izračunati rezidualne,
7. klasa kojoj pripada najmanji rezidual je tražena klasa za objekt d .

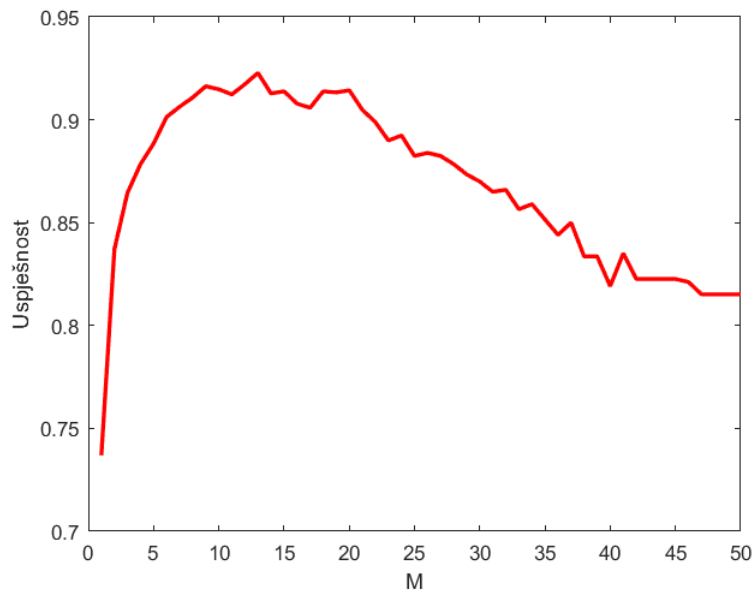
5 Primjene

Primjer 5.1. Isprobati ćemo naše algoritme na skupu podataka crno-bijelih slika znamenki koje smo koristili na kolegiju *Matrične i tenzorske metode u analizi podataka*. Skup podataka za treniranje sadrži 1707 klasificiranih slika znamenki od 0 do 9 rezolucije 16×16 piksela koji su reprezentirani kao matrice dimenzija 16×16 . Skup podataka za testiranje sadrži 2007 klasificiranih slika znamenki u istome formatu.



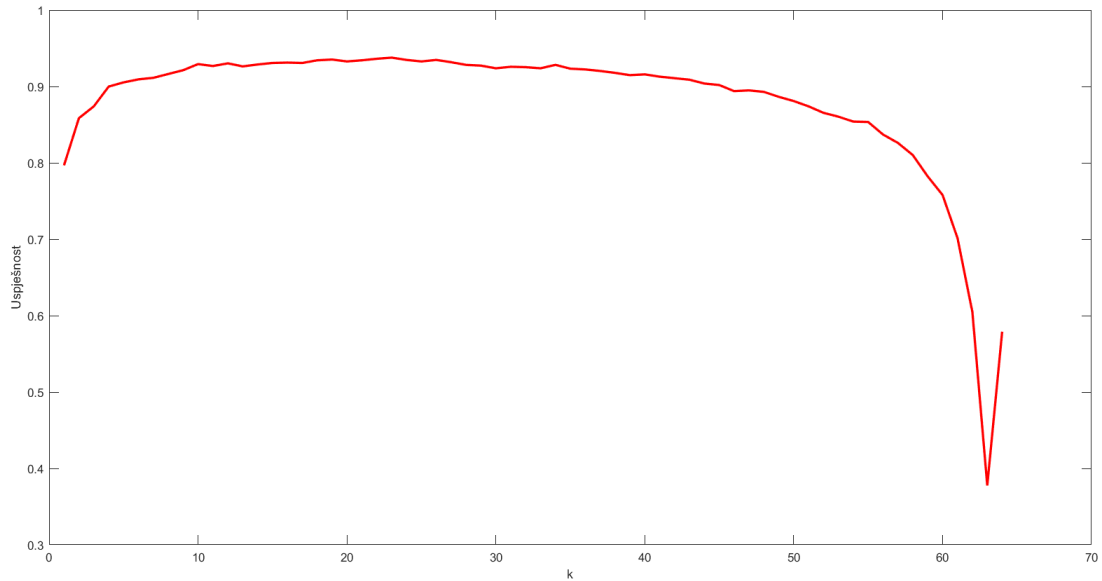
Slika 5.1 Slika prikazuje norme svih baznih matrica za sve znamenke.

Uzimanjem baznih matrica s normom većom od neke vrijednosti M , s prvim algoritmom smo postigli točnost od 92.3%.



Slika 5.2 Slika prikazuje uspješnost prvog algoritma u ovisnosti o M .

Testirali smo drugi algoritam za neke različite vrijednosti p i q iz prethodnog opisa. Za fiksirane vrijednosti p i q smo mijenjali stupčanu dimenziju matrice $B^{(n)}$ tako da smo ju postavili na $k \in \{1, \dots, q\}$. Na primjer, za $p = q = 64$ smo uspjeli postići uspješnost od 93.77%.



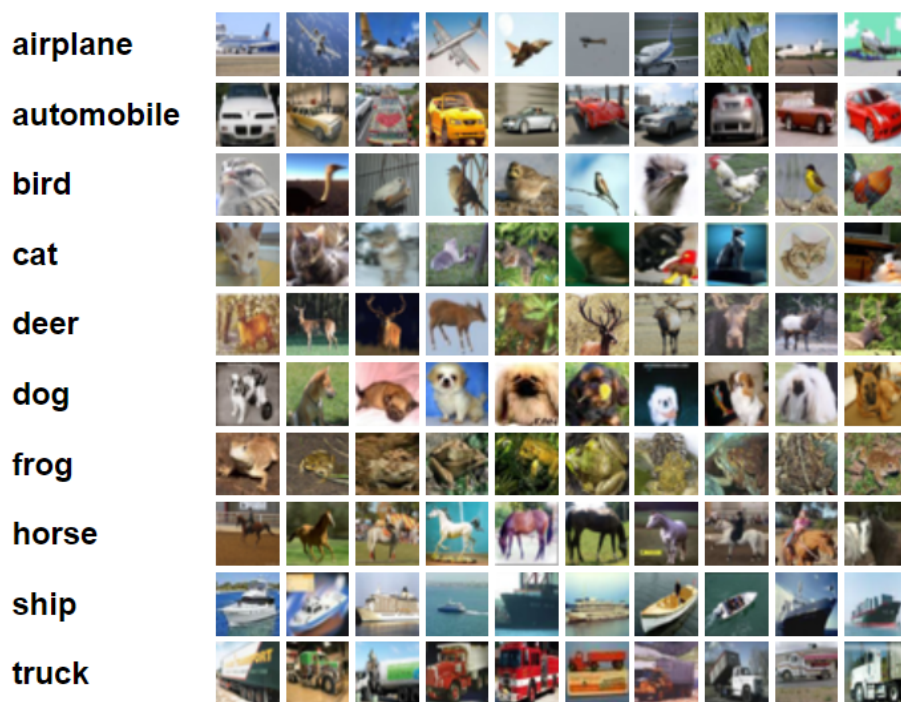
Slika 5.3 Slika prikazuje uspješnost drugog algoritma u ovisnosti o k za $p = q = 64$.

Sveukupno, uspjeli smo postići točnost od oko 93.9%.

p	q	Uspješnost
32	32	91.98%
48	48	93.52%
64	64	93.77%
88	88	93.87%

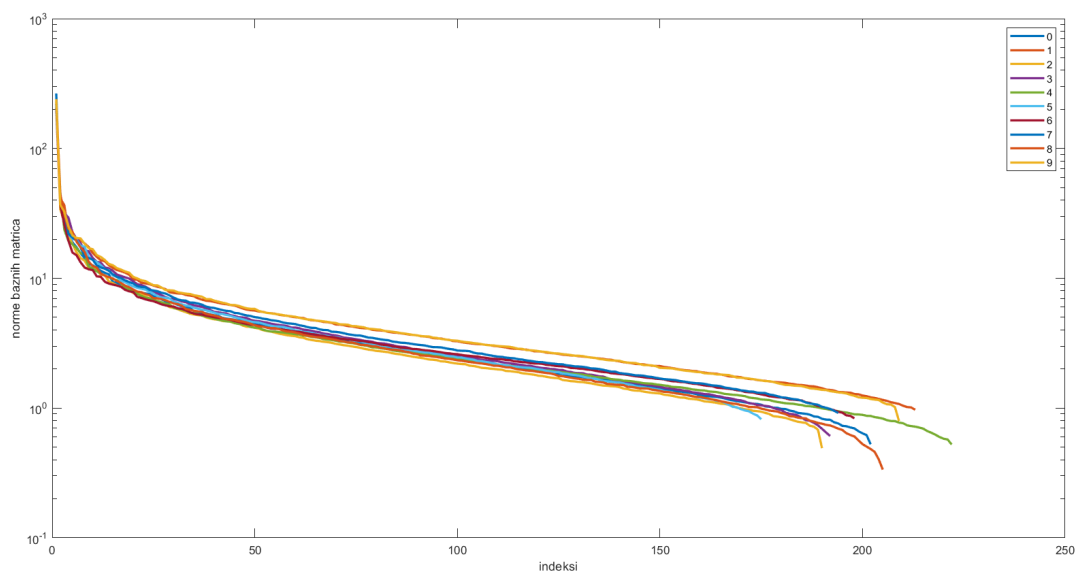
Tablica 5.1 Uspješnost drugog algoritma za različite varijacije parametara p i q .

Primjer 5.2. Koristit ćemo naše algoritme na *CIFAR-10* setu podataka. Za testiranje smo koristili podset od 2000 slika koje su klasificirane u 10 raznih klasa. Set sadrži RGB slike rezolucije 32×32 piksela. Uzeli smo podset od 1000 slika za testiranje točnosti algoritma.



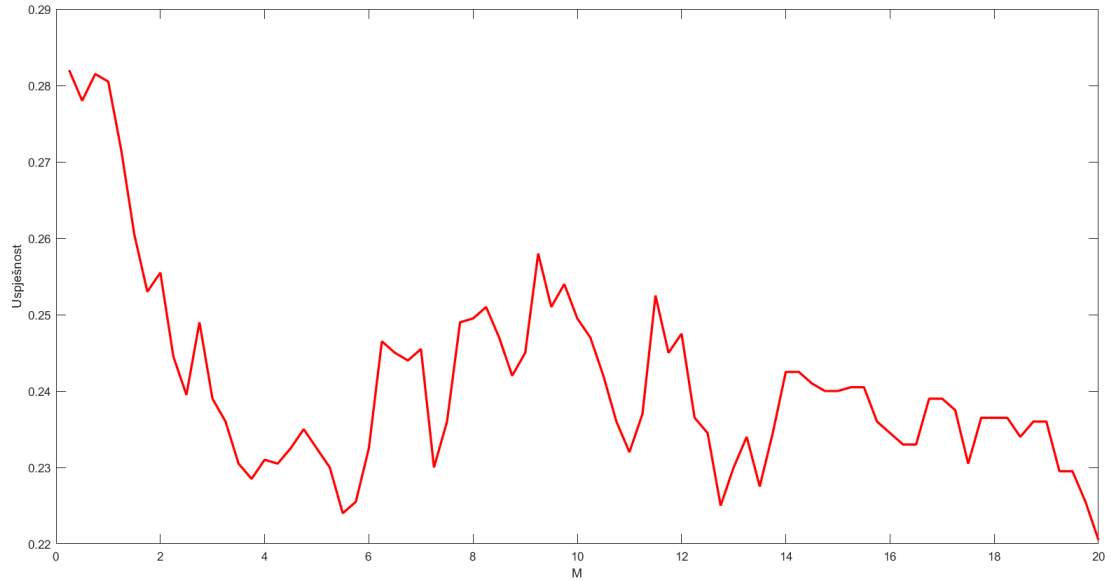
Slika 5.4 Primjeri slika iz *CIFAR-10* seta podataka.

Prvi algoritam smo primijenili na dva načina. Prvi pristup je bio da RGB slike "greyscale"-amo kako bi dobili crno-bijele slike, s kojima smo kasnije radili slično kao i sa znamenkama.



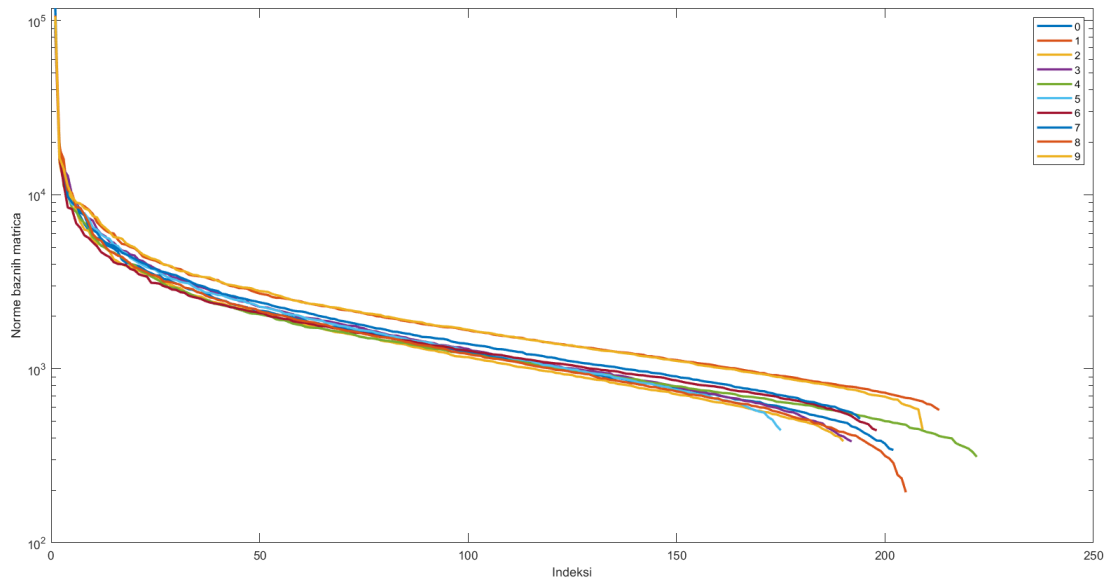
Slika 5.5 Slika prikazuje norme svih baznih matrica za sve klase.

Uzimanjem baznih matrica s normom većom od neke vrijednosti M smo postigli točnost od oko 28.2%.



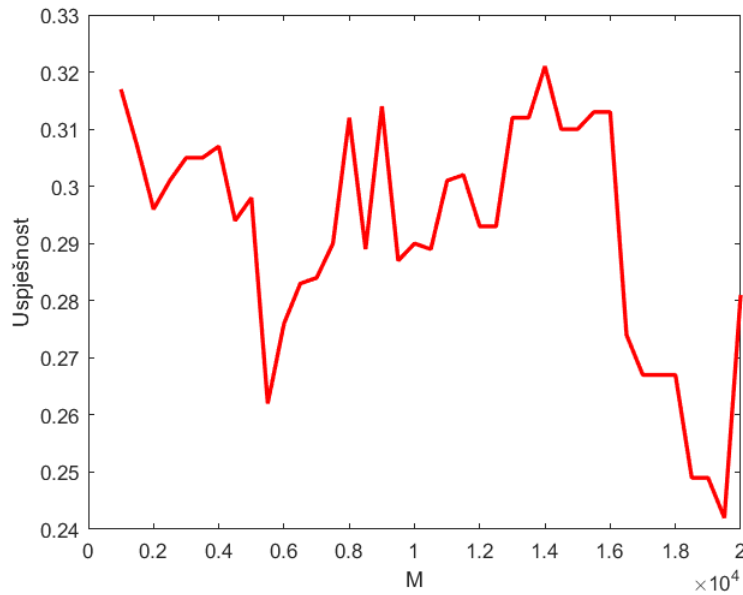
Slika 5.6 Slika prikazuje uspješnost prvog algoritma u ovisnosti o M .

Drugi pristup je bio da radimo s RGB slikama rezolucije 32×32 koje smo spremili kao 96×32 matrice, te na njih primijenili prvi algoritam.



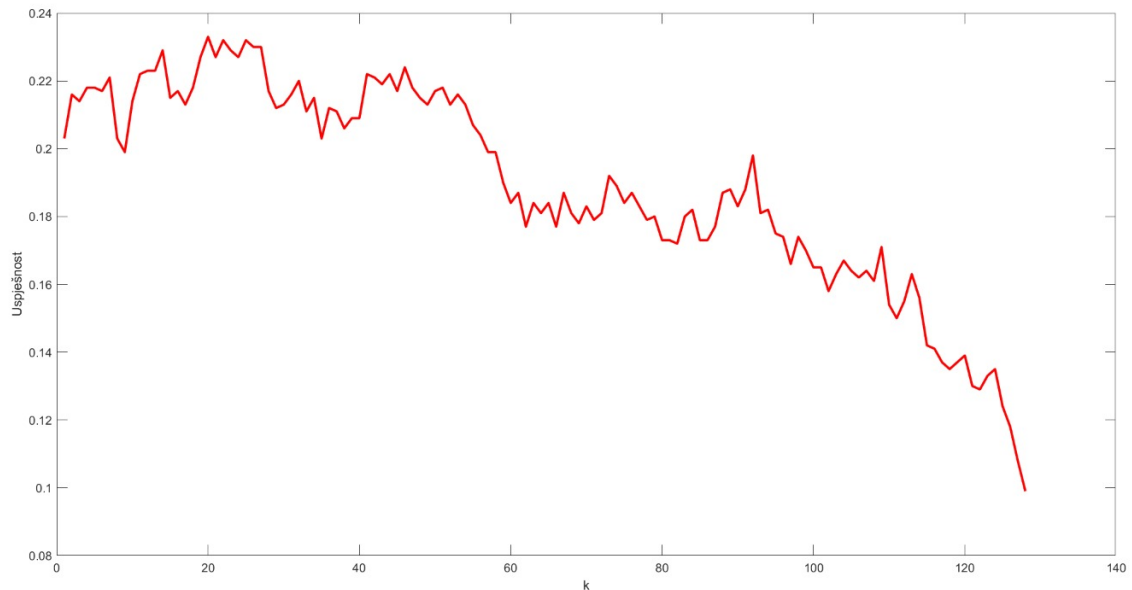
Slika 5.7 Slika prikazuje norme svih baznih matrica za sve klase.

Uzimanjem baznih matrica s normom većom od neke vrijednosti M smo postigli točnost od oko 32.1%.



Slika 5.8 Slika prikazuje uspješnost prvog algoritma u ovisnosti o M .

Pokretanje drugog algoritma na istome setu nije bilo moguće jer su rezultirajući tenzori bili prevelikih dimenzija, što se dovelo do manjka radne memorije prije izvođenju HOSVD-a. Uspjeli smo provesti drugi algoritam za skup od 400 podataka gdje je najveća moguća vrijednost q bila 29. Time smo dobili točnost od 23.3%.



Slika 5.9 Slika prikazuje uspješnost drugog algoritma u ovisnosti o k za $p = 128$, $q = 29$.

Literatura

- [1] L. Eldén, B. Savas: *Handwritten digit classification using higher order singular value decomposition*. Science Direct, 2006.
- [2] Z. Drmač: *Matrične i tenzorske metode u analizi podataka*. PMF-MO Zagreb, 2022.
- [3] CIFAR-10 dataset: <http://www.cs.toronto.edu/~kriz/cifar.html>