

Data Mining Project

Maryam Mohammadabadi

Tir 1402

—

—

Dr Jamal Zarepour

فهرست

3	توضیح کد.....
3	تجزیه SVD.....
7	فشرده سازی عکس (Compression).....
11	اجرای کد و خروجی ها.....

مراحل تجزیه (SVD) :

این الگوریتم برای بدست آوردن مقادیر ویژه و بردارهای ویژه یک ماتریس و تجزیه مقادیر تکین

(Singular Value Decomposition) ماتریس A بکار می‌رود. مراحل این الگوریتم به شرح زیر است:

۱. محاسبه ترانهاده ماتریس A و $A^T A$.

۲. محاسبه مقادیر تکین (singular values)

۳. ساخت ماتریس قطری S و معکوس آن (S^{-1}).

۴. ساخت V . همچنین، محاسبه ترانهاده آن V^T .

۵. ساخت ماتریس U و تجزیه مقادیر تکین ماتریس A .

با استفاده از این الگوریتم، می‌توان تجزیه مقادیر تکین ماتریس A را به دست آورد و از آن برای حل مسائل خطی و فشرده‌سازی داده‌ها استفاده کرد.

```
function [U,S,V] = newSvd(A)
```

مرحله ۱: محاسبه ترانهاده A و ضرب آن با خودش

```
%مرحله 1
```

```
AT = A';
```

```
ATA = AT * A;
```

در این مرحله، ترانهاده ماتریس A محاسبه می‌شود و سپس ضرب داخلی ترانهاده A با خود ماتریس A به دست می‌آید. این عملیات به منظور محاسبه بردارهای ویژه و مقادیر ویژه ماتریس A استفاده می‌شود.


مرحله ۲: محاسبه مقادیر ویژه و بردارهای ویژه ماتریس $A^T A$

```
[eig_vec, eig_val] = eig(ATA);  
[eig_val_sorted, ind] = sort(diag(eig_val), 'descend');  
singular_values = sqrt(eig_val_sorted);
```

- محاسبه بردارهای ویژه و مقادیر ویژه ماتریس $A^T A$: در این بخش با استفاده از تابع `eig`، مقادیر ویژه و بردارهای ویژه ماتریس $A^T A$ محاسبه می‌شود.

eig_vec = بردار ویژه

eig_val = مقدار ویژه

- مرتب‌سازی مقادیر ویژه: در این بخش، مقادیر ویژه محاسبه شده در مرحله قبل به ترتیب نزولی مرتب می‌شوند. این کار به دلیل آن است که مقادیر ویژه بزرگتر، نشان‌دهنده اهمیت بیشتر برای تجزیه مقادیر منفرد هستند. 
- برای مرتب‌سازی مقادیر ویژه در ترتیب نزولی، از تابع `sort` استفاده می‌شود و بردار اندیس مرتب‌شده در متغیر `ind` ذخیره می‌شود. در ادامه، بردارهای ویژه $A^T A$ با استفاده از اندیس‌های مرتب شده از مقادیر ویژه در ترتیب نزولی مرتب می‌شوند.

- محاسبه جذر مقادیر ویژه به عنوان مقادیر تکین (singular values)

در این مرحله، مقادیر تکین با جذر مقادیر ویژه مرتب شده محاسبه می‌شوند.

مرحله ۳: ساختن ماتریس قطری S و S^{-1}

```
% Step 3.  
S = diag(singular_values);  
S_inv=inv(S);  
%S_inv = diag(1./singular_values);
```

ساختن ماتریس قطری S با قرار دادن مقادیر تکین به ترتیب نزولی در قطر آن و محاسبه ماتریس وارون آن (از ماتریس وارون آن برای به دست آوردن ماتریس U استفاده میشود)

❖ تابع diag در متلب : برای ساخت ماتریس قطری

با گرفتن یک بردار ورودی (مقادیر منفرد)، یک ماتریس قطری با اندازه برابر با طول بردار ورودی ایجاد می کند. مقادیر بردار ورودی در قطر این ماتریس قرار می گیرند و سایر مقادیر برابر با صفر خواهند بود.

❖ برای محاسبه ماتریس وارون S ، ابتدا مقادیر ویژه غیر صفر S را معکوس کرده و در قطر ماتریس قرار دهیم. (میتوانیم از تابع inv خود متلب هم استفاده کنیم ولی به دلیل وجود مقادیر صفر زیاد این روش را انجام میدهیم).

مرحله ۴ : ساخت ماتریس V

```
%Step 4.  
V = eig_vec(:, ind);  
VT = V';
```

محاسبه بردارهای ویژه $A^T A$ و قرار دادن آن ها در ستون های ماتریس V . همچنین، محاسبه ترانهاده ماتریس V ، به عنوان V^T .
در این مرحله، بردارهای ویژه ماتریس $A^T A$ مرتب شده با استفاده از اندیس های مرتب شده از مقادیر ویژه در مرحله قبل محاسبه می شوند. سپس با استفاده از این بردارهای ویژه، ماتریس V ساخته می شود و ترانهاده آن (V^T) محاسبه می شود.

مرحله ۵ : محاسبه ماتریس U

```
U = A * V * S_inv;  
A_reconstructed = U * S * VT;  
end
```

در این مرحله، با استفاده از ماتریس های A ، V و ماتریس وارون $S(S^{-1})$ ، ماتریس U محاسبه می شود.

سپس با استفاده از ماتریس های U ، S و V^T بدست آمده در مراحل قبل، تجزیه مقادیر منفرد کامل از ماتریس ورودی A به دست می آید.

```

1
2 function [U,S,V] = newSvd(A)
3
4 %Step 1.
5 AT = A';
6 ATA = AT * A;
7
8 %Step 2.
9 [eig_vec, eig_val] = eig(ATA);
10 [eig_val_sorted, ind] = sort(diag(eig_val), 'descend');
11 singular_values = sqrt(eig_val_sorted);
12
13 % Step 3.
14 S = diag(singular_values);
15 S_inv = diag(1./singular_values);
16
17 %Step 4.
18 V = eig_vec(:, ind);
19 VT = V';
20
21 %Step 5.
22 U = A * V * S_inv;
23 A_reconstructed = U * S * VT;
24 end
25

```

فشرده سازی عکس (Compression)

استفاده از SVD برای فشرده سازی تصویر

$$A = U \cdot \Sigma \cdot V^T$$

ابتدا با استفاده از تابع SVD که در قسمت قبل توضیح داده شد U, S, V را به دست میاوریم و سپس یک عدد k (که در اینجا ما از اعداد ۱۰ و ۲۰ و ۳۰ و ۵۰ استفاده میکنیم) از مقادیر مفرد را انتخاب می کنیم که می خواهیم برای تقریب استفاده کنیم.

هرچه این عدد بیشتر باشد، کیفیت تقریب بهتر می شود، اما همچنین داده های بیشتری برای رمزگذاری آن مورد نیاز است.

ما اکنون فقط اولین k ستون U و V و مربع سمت چپ بالایی $(k \times k)$ از Σ را می گیریم که حاوی k بزرگترین (و بنابراین مهم ترین) مقادیر مفرد است. پس از آن داریم

$$A \approx \begin{matrix} k \\ U \end{matrix} \cdot \begin{matrix} k & k \\ \Sigma \end{matrix} \cdot \begin{matrix} k \\ V^T \end{matrix}$$

```
function [] = Compression(img)
[U, S, V] = newSvd(img);
```

(۱) به دست آوردن U,S,V

(۲) یک آرایه سلولی تعریف میکنیم برای ذخیره عکس ها

```
% تعریف آرایه برای ذخیره عکس های فشرده شده
images = cell(1, 4);
```

(۳) درصد های فشرده سازی را تعریف کرده در یک آرایه و با استفاده از حلقه هر درصد را روی عکس مورد نظر اعمال کرده و در پایان عکس را داخل آرایه قرار میدهد.

```
% تعریف درصدهای فشرده سازی
compression_ratios = [10, 20, 30, 50];
for i = 1:length(compression_ratios)
    % محاسبه تعداد مقادیر تکین جدید
    n = round(compression_ratios(i)/100*size(S,1));
    % بازسازی ماتریس با تعداد مقادیر تکین جدید
    A_new = U(:,1:n)*S(1:n,1:n)*V(:,1:n)';
    images{i}=A_new;
end
```

در این کد،

`compression_ratios(i)` نسبت فشرده سازی را برای فشرده سازی تصویر مشخص می کند.

با توجه به نسبت فشرده سازی و اندازه ی ماتریس `S`، اندازه برش `n` با استفاده از کد

`n = round(compression_ratios(i)/100*size(S,1))` محاسبه می شود.

سپس با استفاده از برش مناسب، ماتریس اصلی با استفاده از تجزیه SVD ماتریس جدید `A_new` تقریب زده شده است.

۴) و در پایان عکسای موجود را با عنوان درصدهاشون چاپ میکند.

```
% رسم تصاویر فشرده شده جدید
figure;
subplot(2,2,1);imshow(images{1},[]); title('10%');
subplot(2,2,2);imshow(images{2},[]); title('20%');
subplot(2,2,3);imshow(images{3},[]); title('30%');
subplot(2,2,4);imshow(images{4},[]); title('50%');
end
```

کد کامل فشرده سازی:



```
function [] = Compression(img)
[U, S, V] = newSvd(img);
% تعریف ارایه برای ذخیره عکس های فشرده شده
images = cell(1, 4);
% تعریف درصد های فشرده سازی
compression_ratios = [10, 20, 30, 50];
for i = 1:length(compression_ratios)
    % محاسبه تعداد مقادیر تکین جدید
    n = round(compression_ratios(i)/100*size(S,1));
    % بازسازی ماتریس با تعداد مقادیر تکین جدید
    A_new = U(:,1:n)*S(1:n,1:n)*V(:,1:n)';
    images{i}=A_new;
end
% رسم تصاویر فشرده شده جدید
figure;
subplot(2,2,1);imshow(images{1},[]); title('10%');
subplot(2,2,2);imshow(images{2},[]); title('20%');
subplot(2,2,3);imshow(images{3},[]); title('30%');
subplot(2,2,4);imshow(images{4},[]); title('50%');
end
```

dataMiningProject.m

```
1 clear all;
2 clc;
3
4 n=4;
5 for i=1:n
6     image=imread(['image' num2str(i) '.jpg']);
7     A = double(rgb2gray(image));
8
9     [U,S,V]=newSvd(A);
10    noisy_image = imnoise(image, 'gaussian');
11    A2 = double(rgb2gray(noisy_image));
12    [U2,S2,V2]=newSvd(A2);
13
14    figure(i);
15    set(gcf, 'Name', ['image' num2str(i) '.jpg']);
16    subplot(1,2,1);plot(diag(S));title('Original photo');
17    subplot(1,2,2); plot(diag(S2));title('with noise');
18
19 end
20
21 Compression(double(rgb2gray(imread('image1.jpg'))));
22 Compression(double(rgb2gray(imread('image2.jpg'))));
23 Compression(double(rgb2gray(imread('image3.jpg'))));
24 Compression(double(rgb2gray(imread('image4.jpg'))));
```

در این اسکریپت توابع مذکور روی عکس ها اجرا میشود.

برای راحتی کار یک حلقه تعریف شده که ابتدا تعداد عکس ها را خوانده و عکس (ساده یا نویزی) را به تابع SVD میدهد و خروجی را به عنوان یک نمودار نشان میدهد.

در قسمت دوم تابع فشرده سازی روی عکس های پیاده میشود.

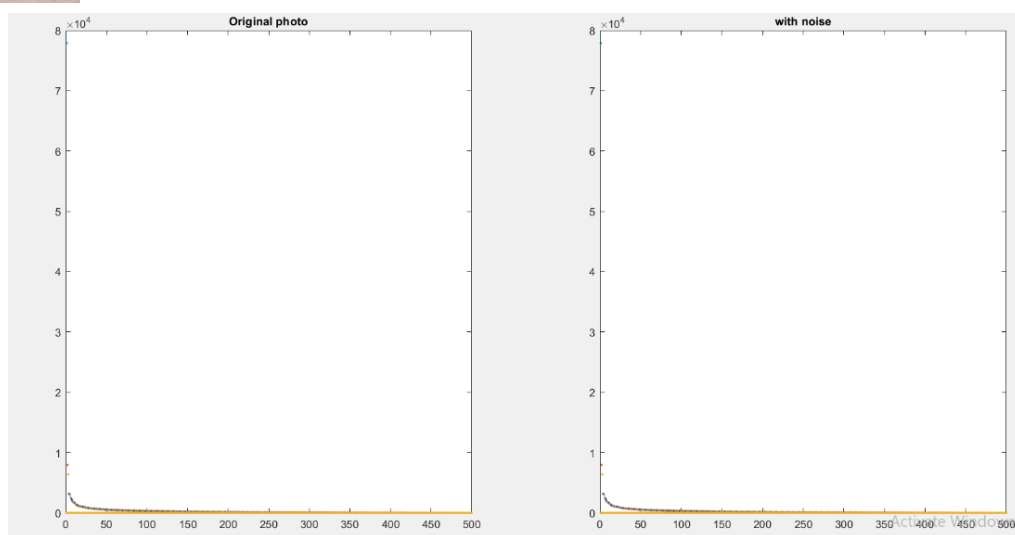
اجرای کد

عکس اول:



برای عکس اول نمودار مقادیر تکین به صورت زیر خواهد بود

که در کنارش نمودار همین عکس در صورتی که نویز داشته باشد نشان داده شده.



تاثیر درصد های فشرده سازی ۱۰ و ۲۰ و ۳۰ و ۵۰ رو این عکس:

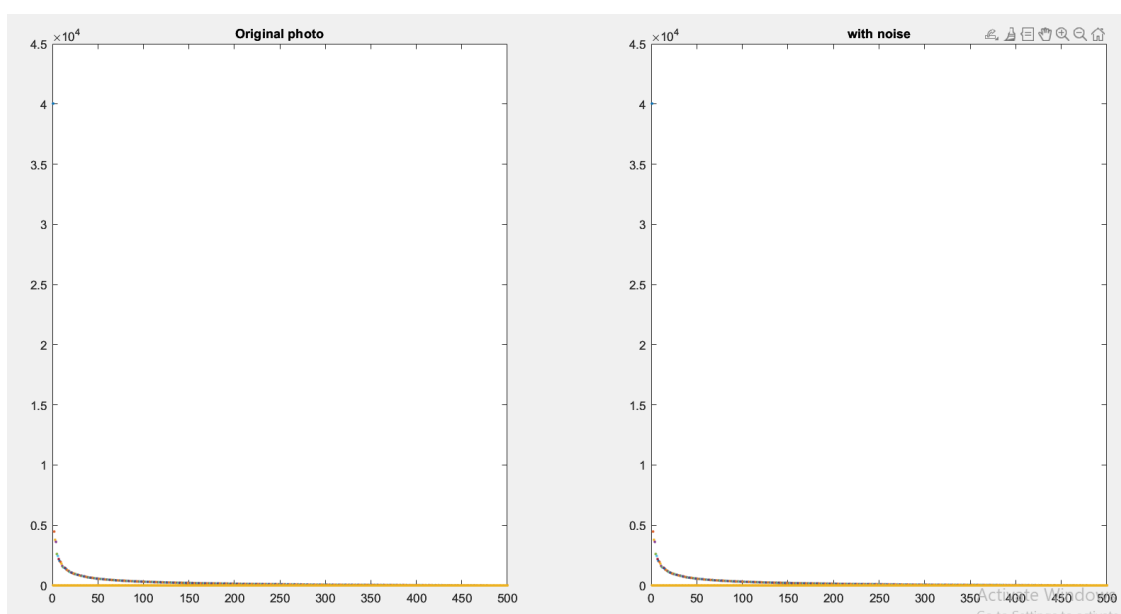


عکس دوم



برای عکس دوم نمودار مقادیر تکین به صورت زیر خواهد بود

که در کنارش نمودار همین عکس در صورتی که نویز داشته باشد نشان داده شده.



تاثیر درصد های ۱۰ و ۲۰ و ۳۰ و ۵۰ رو این عکس:

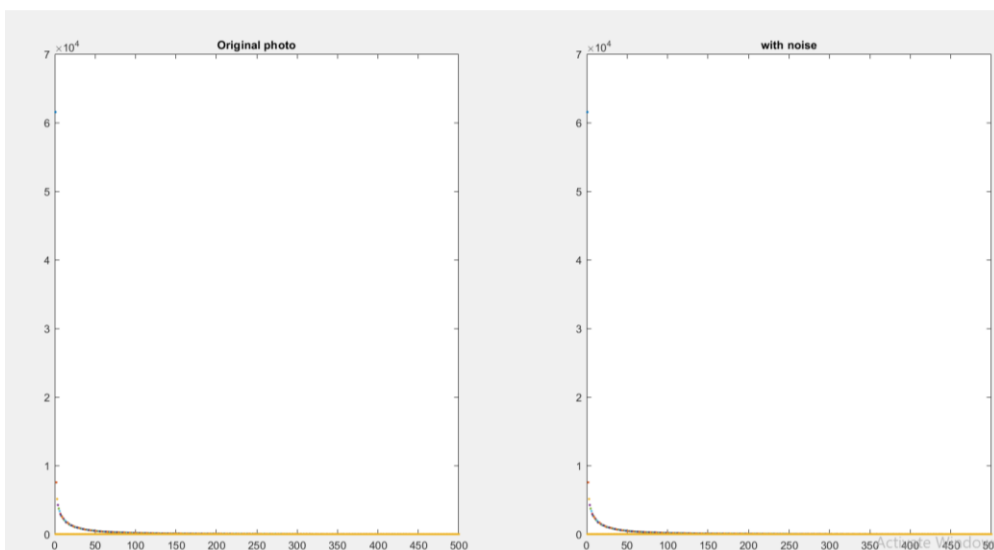


عکس سوم:

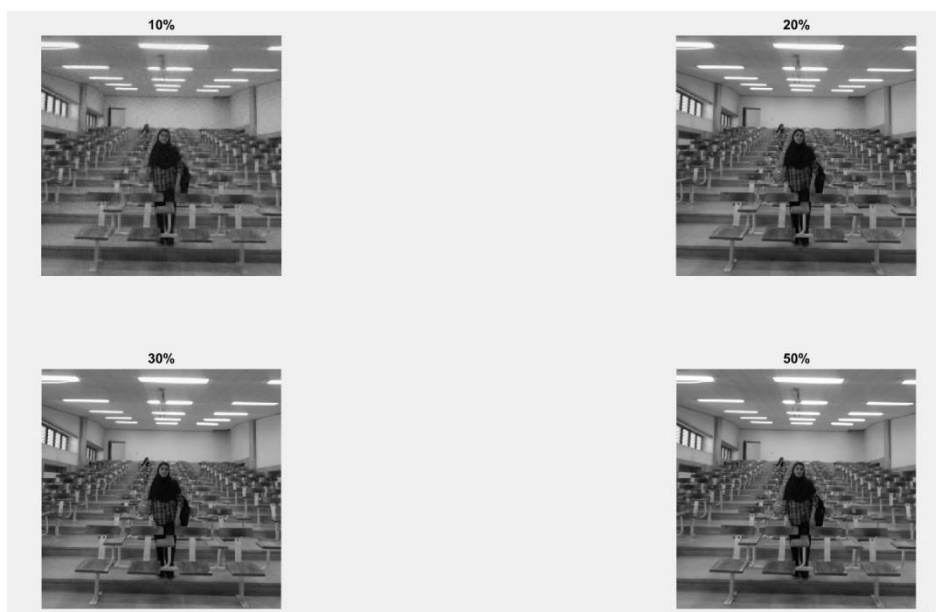


برای عکس سوم نمودار مقادیر تکین به صورت زیر خواهد بود

که در کنارش نمودار همین عکس در صورتی که نویز داشته باشد نشان داده شده.



تاثیر درصد های ۱۰ و ۲۰ و ۳۰ و ۵۰ رو این عکس:

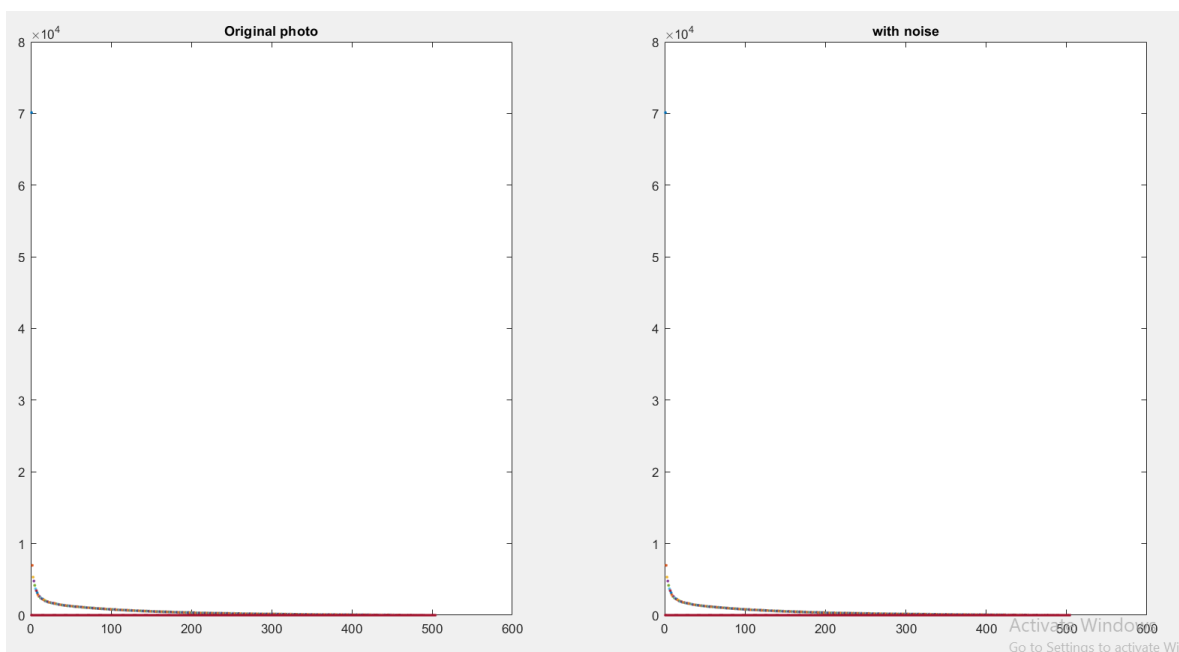


عکس چهارم:



برای عکس چهارم نمودار مقادیر تکین به صورت زیر خواهد بود

که در کنارش نمودار همین عکس در صورتی که نویز داشته باشد نشان داده شده.



تاثیر درصد های ۱۰ و ۲۰ و ۳۰ و ۵۰ رو این عکس:

