Computer Vision: Fall 2022 — Lecture 2

Dr. Karthik Mohan

Univ. of Washington, Seattle

October 4, 2022

Weekly Logistics

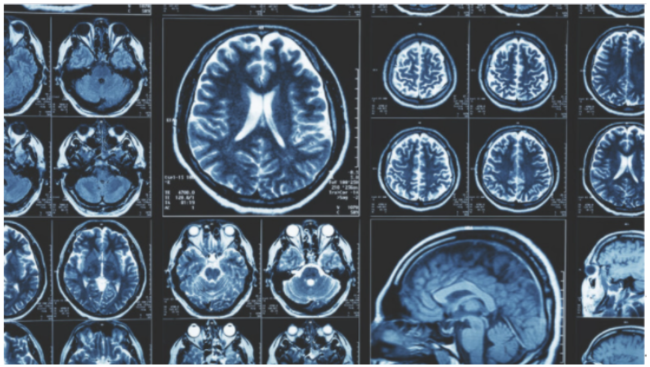
	Day	Timings	Class type
Lecture 1 (In-person)	T	4 pm - 6 pm	(In-person)
Lecture 2	Th	4 pm - 6 pm	Zoom
Office Hours Karthik	T	6 - 6:30 pm	In-person/Zoom
Calendly 15 min Karthik	October		Zoom
Office Hours Ayush	Fri	5-6 pm	Zoom
Quiz Section Ayush	Mon	5-6 pm	Zoom

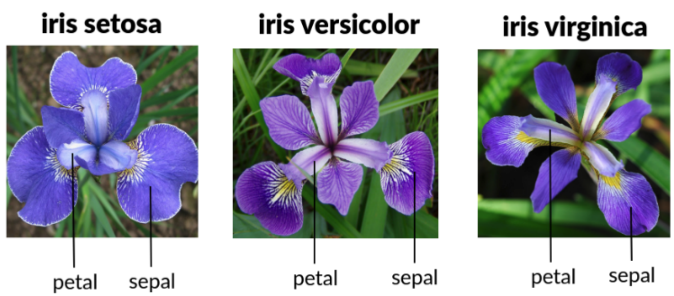
References for Lecture

1. Image Compression with SVD
2. kMeans Demo
3. Deep Learning TextBook by Yoshua Bengio et al

Find a buddy in the room!

Applications



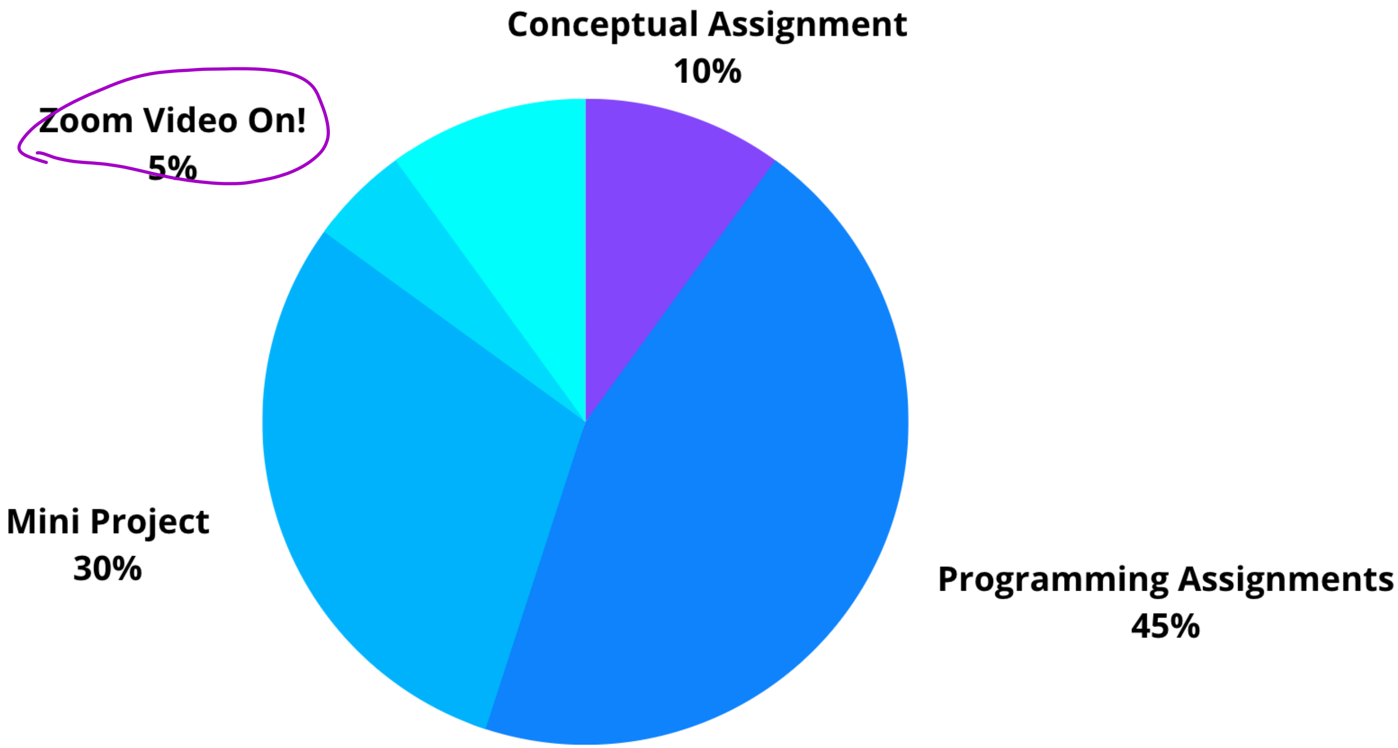


Syllabus

Week by Week

Week	Topic
1	Motivation and applications of CV
2	Transforms, Convolutions and feature extraction
3	Machine Learning for CV
4	Machine Learning for CV
5	Neural Networks & CNN
6	Pytorch Tutorial and libraries
7	Object detection and instance segmentation
8	Deep Learning applications in CV
9	Image to Text and Text to Image
10	More Deep Learning applications in CV

Assessments Breakdown



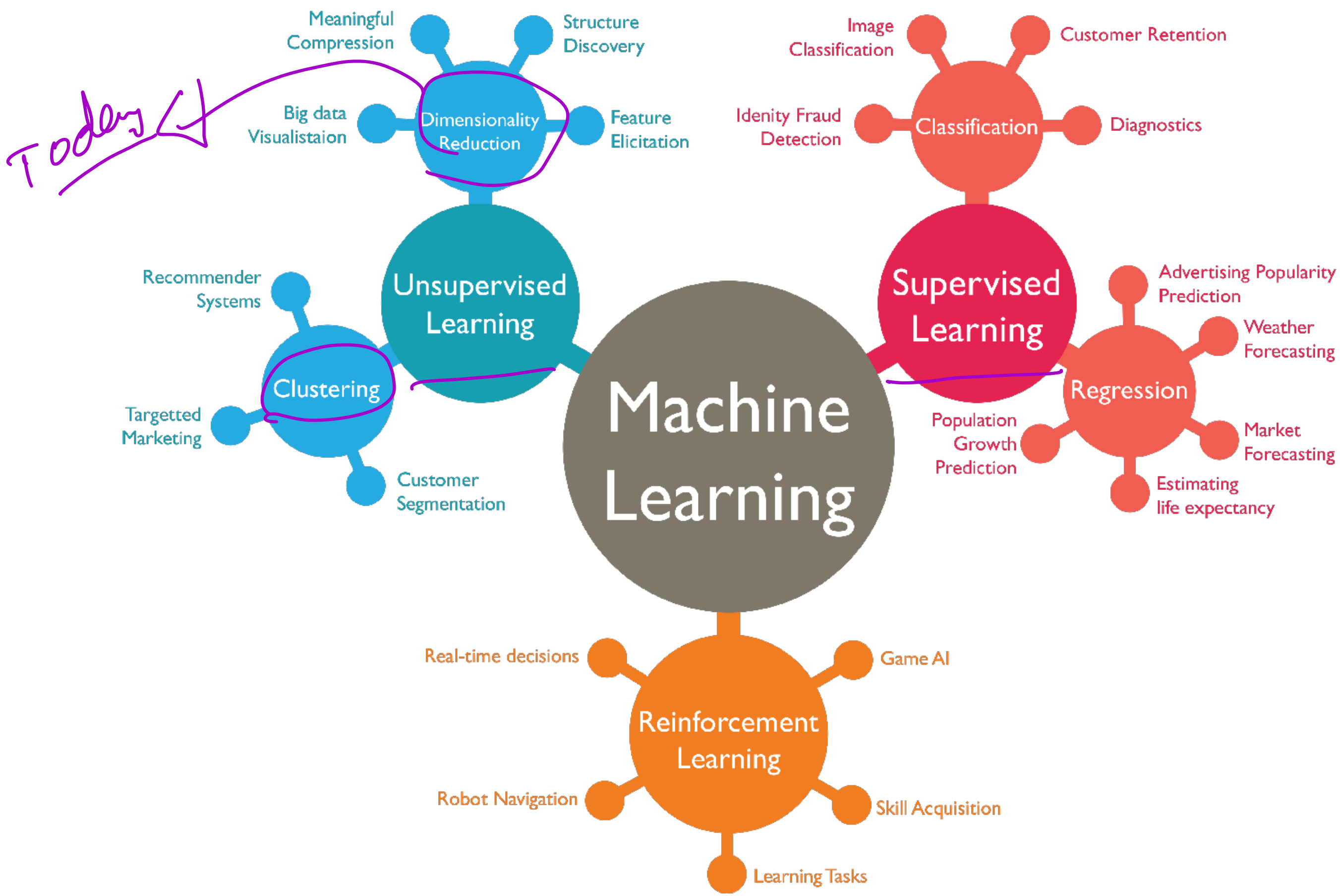
Computer Vision Problem Spaces we will tocuh on

1. Image processing
2. Image de-noising
3. Image smoothing
4. Image Classification
5. Object Detection
6. Semantic Segmentation
7. Instance Segmentation (maybe)
8. Image Embeddings
9. Convolutional Neural Networks (CNNs)
10. Image to text
11. Image Captioning
12. Text to Image (high-level)

Today!

1. Machine Learning Introduction
2. Unsupervised Learning for Images

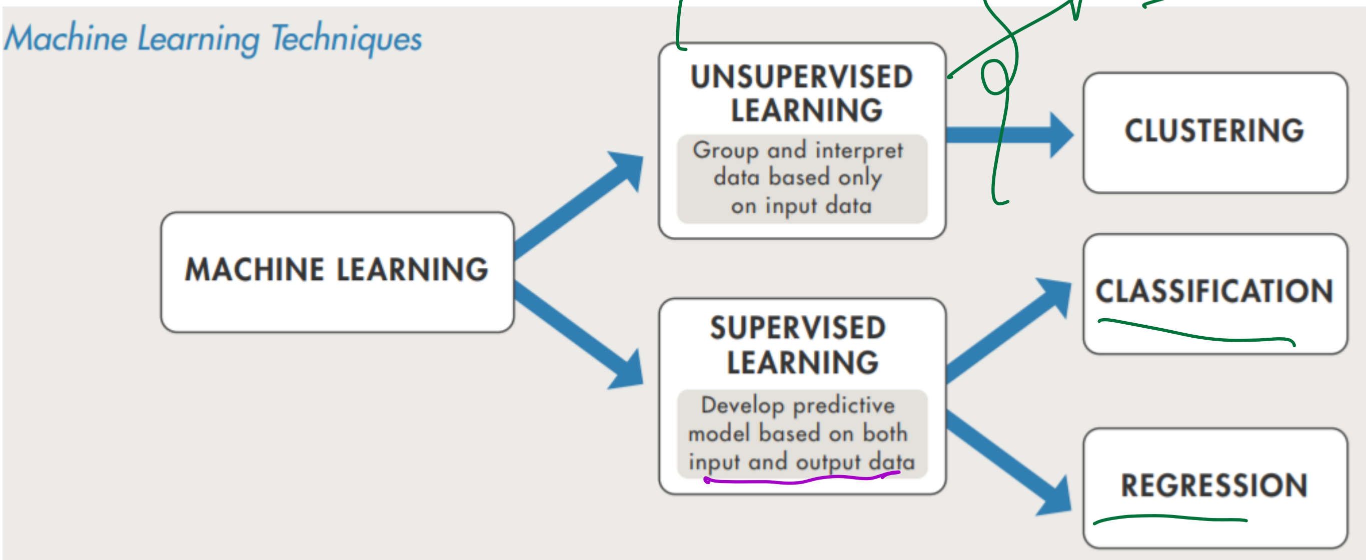
What is Machine Learning?



Supervised vs Unsupervised Learning

$i_1, i_2, i_3, \ldots$

Dim. Reduction



(Input , output)

Train Set $\leftarrow [(i_1, o_1)\ (i_2, o_2)\ (i_3, o_3) \ldots\ldots]$

Supervised Learning

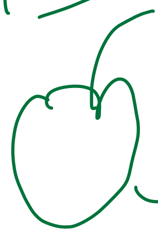
Input / Features

i

o

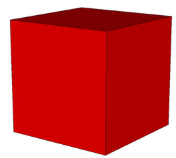
Target / Label

Test Example

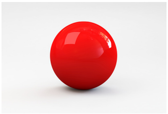


Apple

Apple



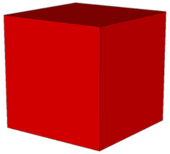
cube

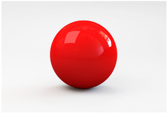


Ball

Apple

Un-Supervised Learning





ICE #1

Blurring Convolution

Consider the 3x3 blur convolution matrix - where every entry of the matrix is $\frac{1}{9}$. When applied to an image - It blurs the image. Is this an example of (pick all that apply):

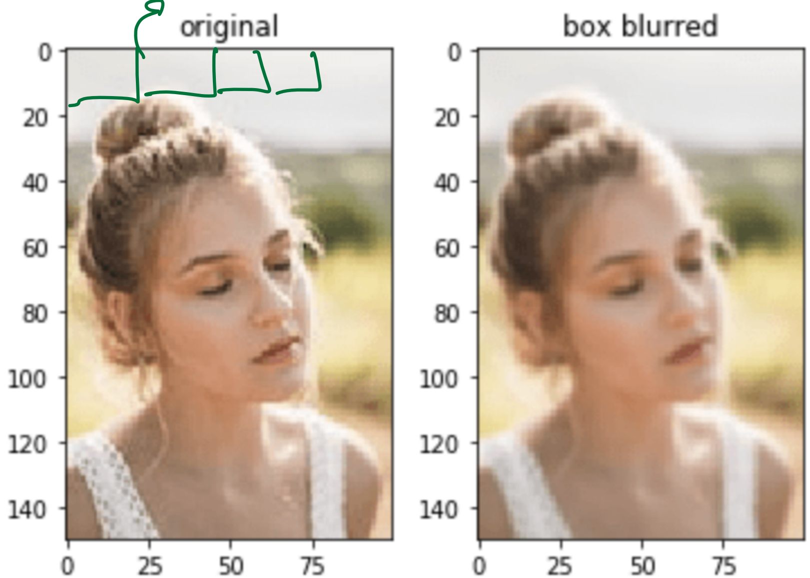
a. Supervised Learning
b. Unsupervised Learning
c. Semi-Supervised Learning
d. Image Processing Technique

Submit your answer on the POLL

Box Blur Convolutor

$$C = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

C



$* \, C$

ICE #2

Child Learning to identify an apple!

We looked at the example of a 3 year old kid learning to identify apple from different objects. What is this an example of?

a. Unsupervised Learning
b. Supervised Learning ✓
c. Neither
d. Both

SVD for Image Compression

(Transformation) "Decomposition"

SVD of a Matrix

Every matrix, $X \in^{m \times n}$ has a **Singular Value Decomposition (SVD)** given by three matrices U, Σ, V^T such that

Decomposed

$$X = U\Sigma V^T$$

Singular Vectors

orthogonal

$u_1 \cdot u_2 = 0$ $u_2 \cdot u_3 = 0$ $u_2 \cdot u_2 = 1$
$u_3 \cdot u_1 = 0$ $u_1^T u_1 = 1$
$u_3 \cdot u_3 = 1$

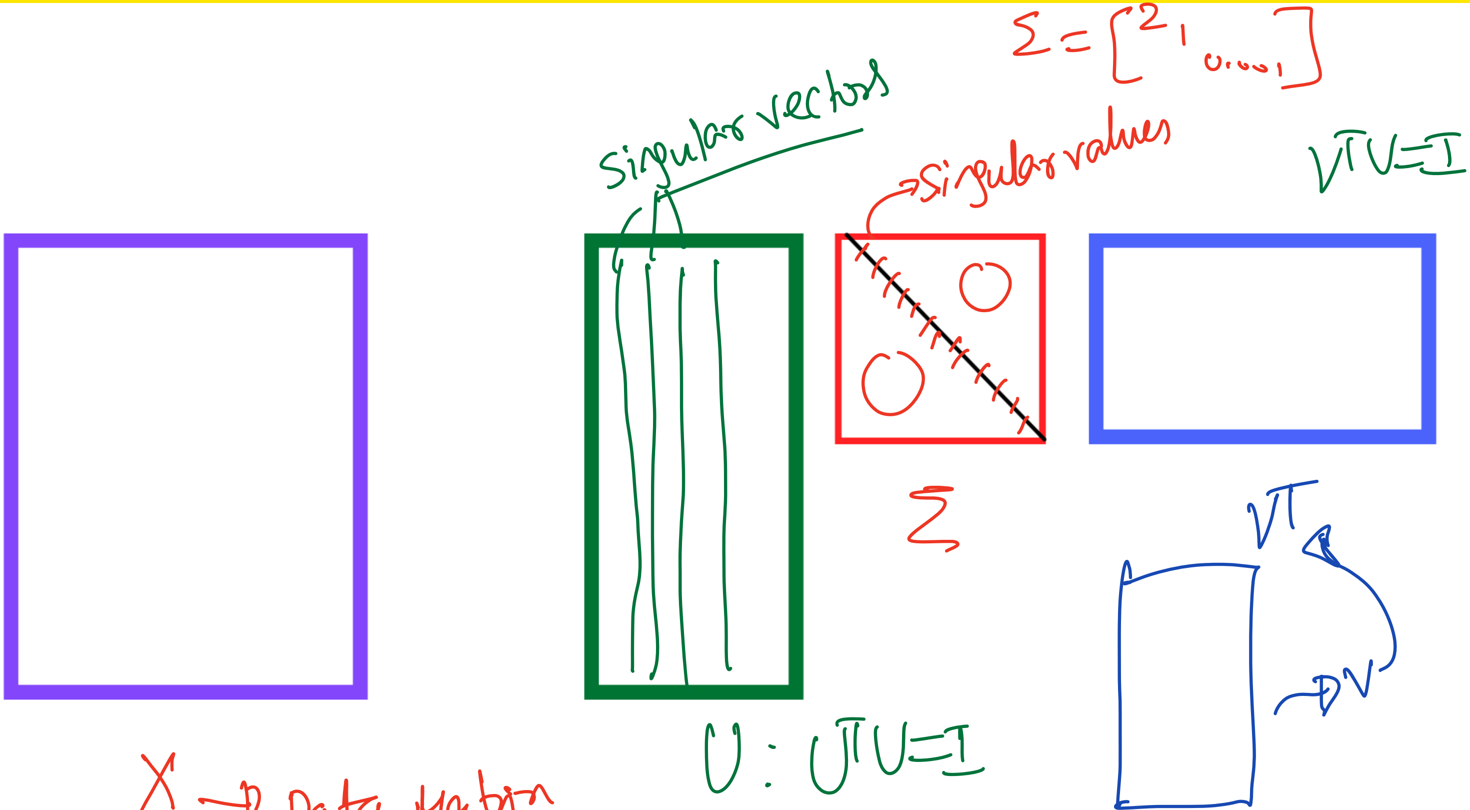
The matrices U, V are such that $U^T U = I$ and $V^T V = I$. So the columns of U and the columns of V are called the singular vectors.

Singular Values

Σ is a diagonal matrix and the entries on the diagonal are called singular values.

All entries of $\Sigma \geq 0$

SVD



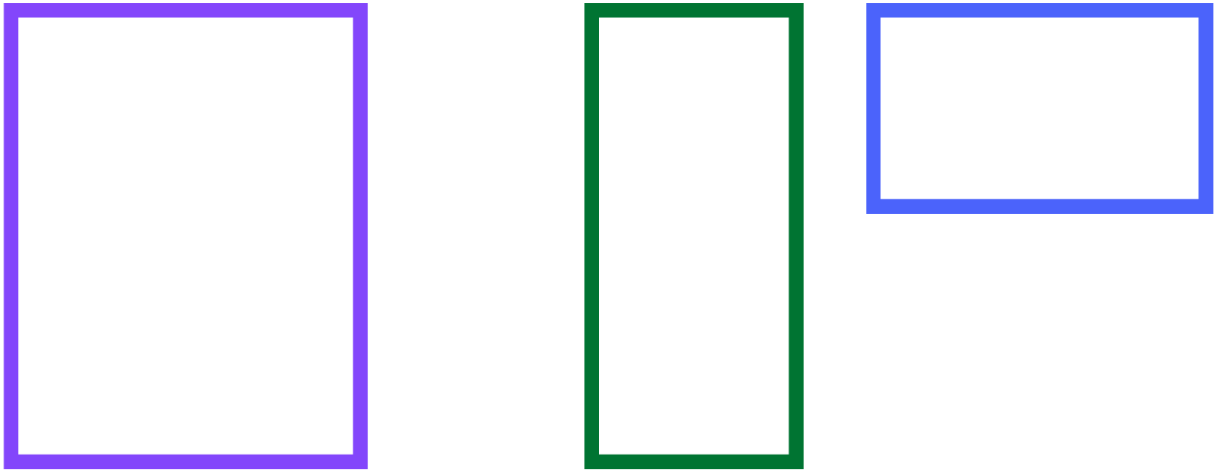
$X \rightarrow$ Data Matrix

1. (Image) 1000x1000 pixels

2. Set of Images $\rightarrow$ E.S. 1000x1000 pixels & 100 images

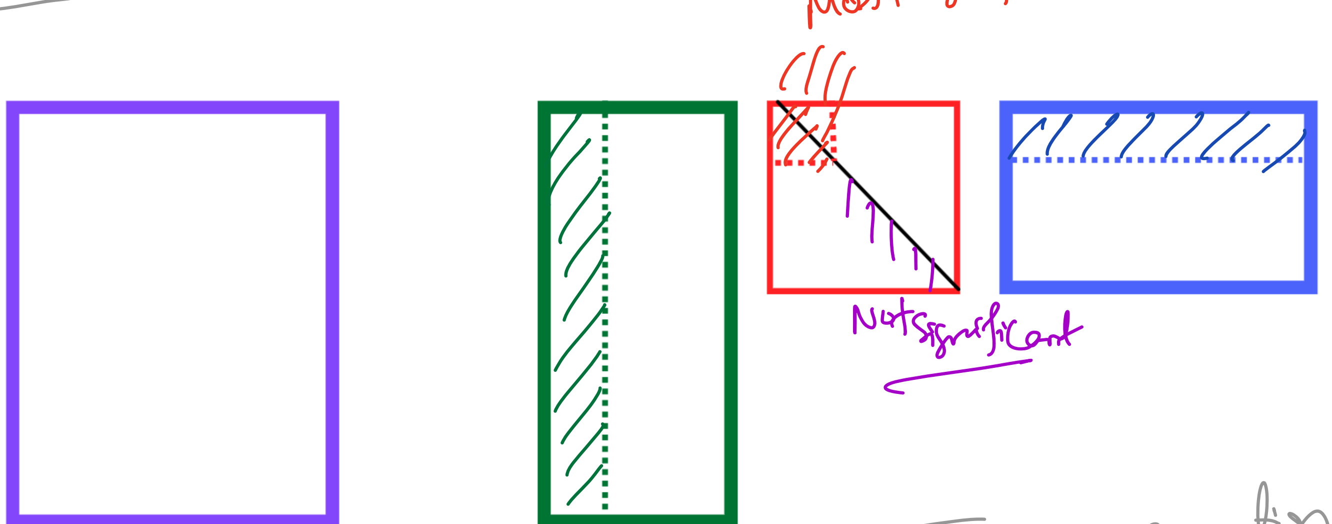
$\Rightarrow$ 1MM x 100 matrix

SVD and Two Factors

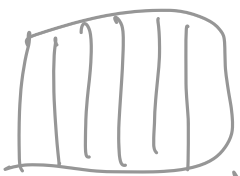


Reduced SVD or Low-Rank SVD && Image Compression

Matrix Rank :- # Non-zero singular values!



Shaded → Significant Information!



↳ Image of Low rank (Rank 1)

↳ 1 Column is sufficient to know the matrix!

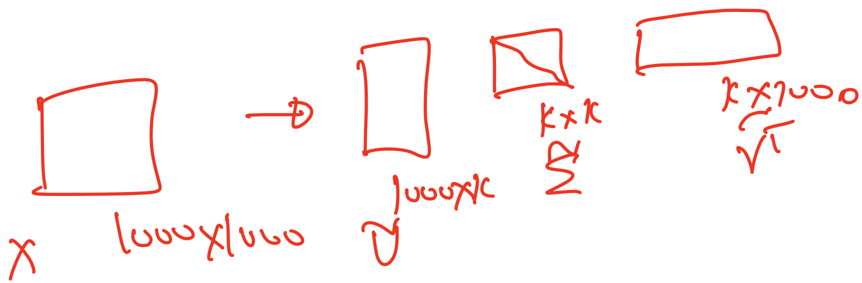
SVD based Image Compression

ICE #3

SVD based Image Compression

Consider a RGB image of size 1000×1000 pixels with a file size of $20MB$. You want to store it in a more compressed format and your file size limit for the compressed format is $5MB$. You decide to use SVD to do the compression of the image. What should be the number of singular vectors, k you pick for the SVD compression so you can achieve your desired compression?

- a) 500
- b) 200
- c) 125
- d) 100



Submit your answer on the POLL

No Compression

Info1 = 1000×1000 pixels

Compression

Info2 = $\underbrace{1000 \times K}_{U} + \underbrace{K \times 1000}_{V^T} + K =$

$\simeq 2 \times K \times 1000$

$$\frac{\text{Info1}}{\text{Info2}} = \frac{20\,MB}{5\,MB} = 4 = \frac{1000 \times 1000}{2 \times K \times 1000}$$

$$\Rightarrow K = \frac{1000}{8} = 125!$$

$\Rightarrow K < 125$ to get desired compression!

SVD based Image Compression — Demo

SVD Demo

Understand Singular values Better

Keep | Throw away

$$\underline{8} = 4 + 3 + 0.5 + 0.2 + \; | \; 0.1 + \dots -$$

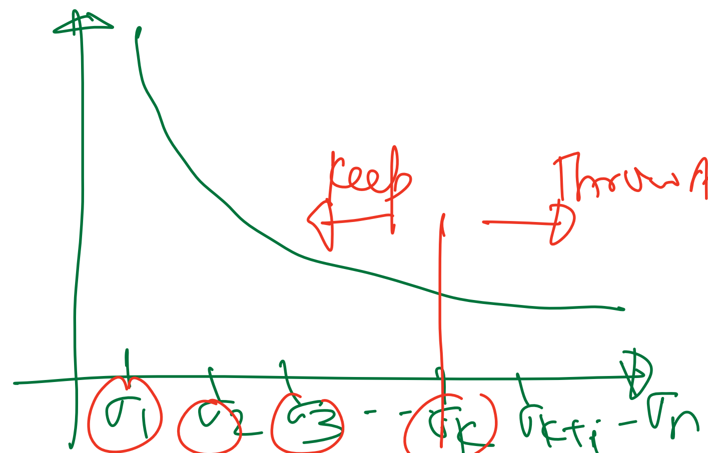
$\downarrow$

X

Remove 4 $\Rightarrow \tilde{X} < 4$

Remove 0.1 & lower $\Rightarrow \tilde{X} = 7.7$

Singular value curve



$[\; u_1^T \;] \begin{bmatrix} \\ \\ \end{bmatrix} v_1$

$u_1^T v_1$

Inner product

Matrix of Rank 1

$$X = \underbrace{\sigma_1 u_1 v_1^T} + \sigma_2 u_2 v_2^T + \sigma_3 u_3 v_3^T + \dots + \sigma_k u_k v_k^T + \dots + \sigma_n u_n v_n^T$$

outer product

$\begin{bmatrix} \\ \\ \end{bmatrix}_{u_1} [\; u_1^T \;] \qquad \begin{bmatrix} \\ \\ \end{bmatrix}_{u_2} [\; v_2^T \;]$

ICE #4

(Extra Credit)

SVD based Image Compression

Consider a RGB image of size $n \times n$ pixels. You want to compress it by a factor of α. You decide to use SVD to do the compression of the image. What should be the number of singular vectors, k you pick for the SVD compression so you can achieve your desired compression?

a. $\frac{n}{\alpha}$
b. $\frac{n}{2\alpha}$
c. $\frac{n}{3\alpha}$
d. $\frac{n}{4\alpha}$

Submit your answer on the POLL

SVD vs $\underbrace{\text{Eigen}}_{\text{Square matrices}}$ Decomposition

↳ Derived from SVD!

If $X = X^T$ (symmetric)

$$X = U \Lambda U^T \quad \Big| \quad Xu = \lambda u$$

Eigen Faces

x_2 x_3 x_4

Training Image with True Label (LFW people's dataset)

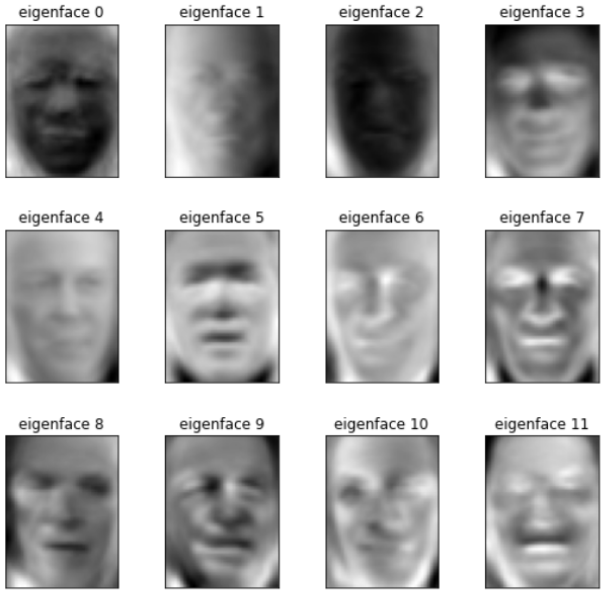
x_1

$$\underline{X} = \begin{bmatrix} | & | & | & | \\ x_1 & x_2 & x_3 & x_4 \\ | & | & | & | \end{bmatrix}$$

Each column is an image

Eigen Faces

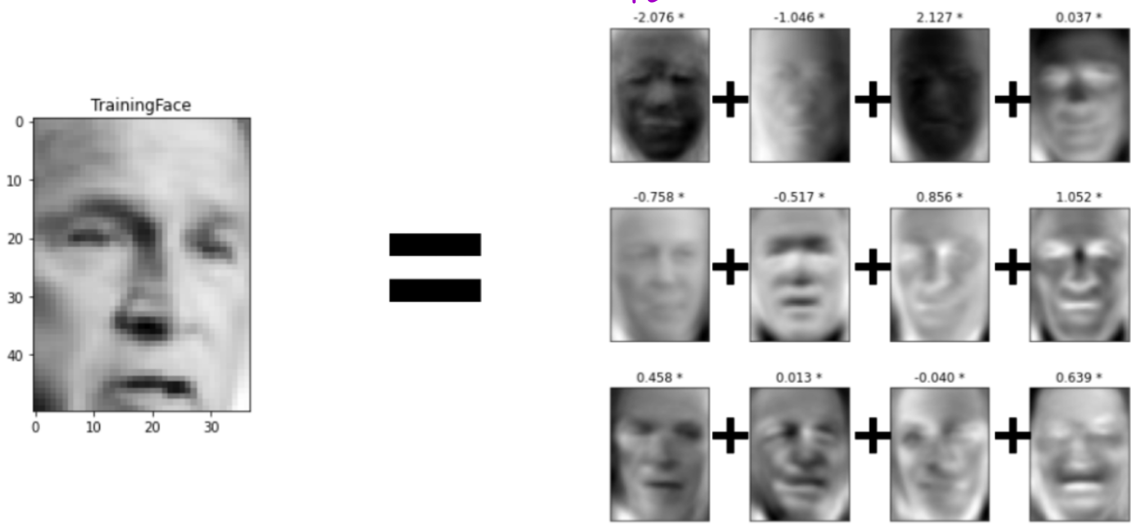
These u_j are called **EigenFaces**.



EigenFaces

Eigen Faces

α_1 α_2 α_3 α_4 - .



Linear Combination of EigenFaces

$X = U \Lambda U^T$ } Eigen Decomposition

$\hat{x} \rightarrow$ New Image

$\tilde{U} \rightarrow$ Set of Eigen Faces with $K = 10$

$(1000 \times 1000) \times 10$

$l(\alpha)$

$$\min_{\alpha} \|\hat{x} - \tilde{U}\alpha\|_2^2 \rightarrow \text{optimization problem}$$

$\alpha \rightarrow$ weighting vector

$\tilde{U}\alpha \approx \hat{x} \rightarrow$ New Image

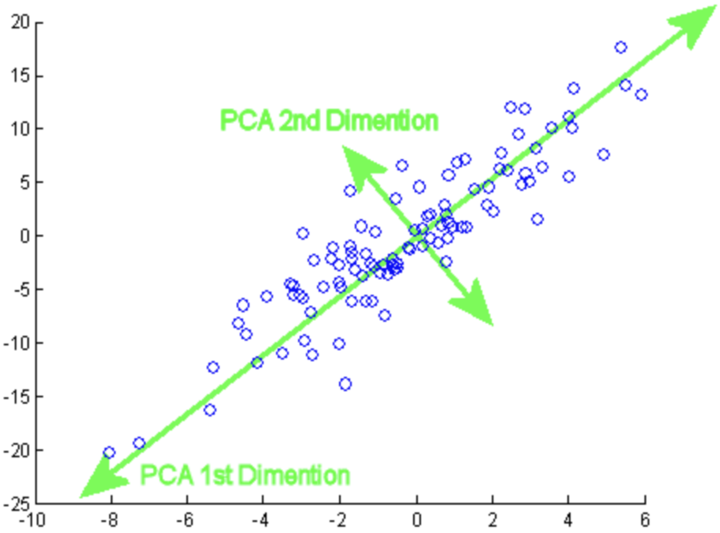
$\approx \rightarrow$ Approximates

$$l(\alpha) = \hat{x}^T\hat{x} + \alpha^T \tilde{U}^T \tilde{U} \alpha - 2\alpha^T \tilde{U}^T \hat{x}$$

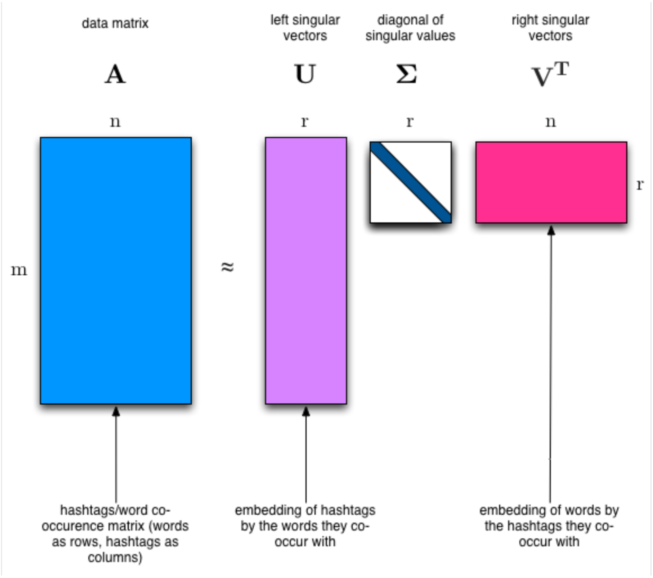
$\nabla l(\alpha) = 0$ | Gradient & set to zero

Solve this for α $\left\{ 2\tilde{U}^T\tilde{U}\alpha - 2\tilde{U}^T\hat{x} = 0 \right.$

SVD and PCA


PCA 2nd Dimention
PCA 1st Dimention
20
15
10
5
0
-5
-10
-15
-20
-25
-10
-8
-6
-4
-2
0
2
4
6

Matrix Factorization: SVD for Tweet embeddings and recommendations



Winter 2022 course on Recommender Systems

ICE #5

Submit your answer on the POLL