Session on Linear Algebra

Contents

Matrix multiplication as a Transformation

- A is a transformation that acts on vector x to produce a new vector y.

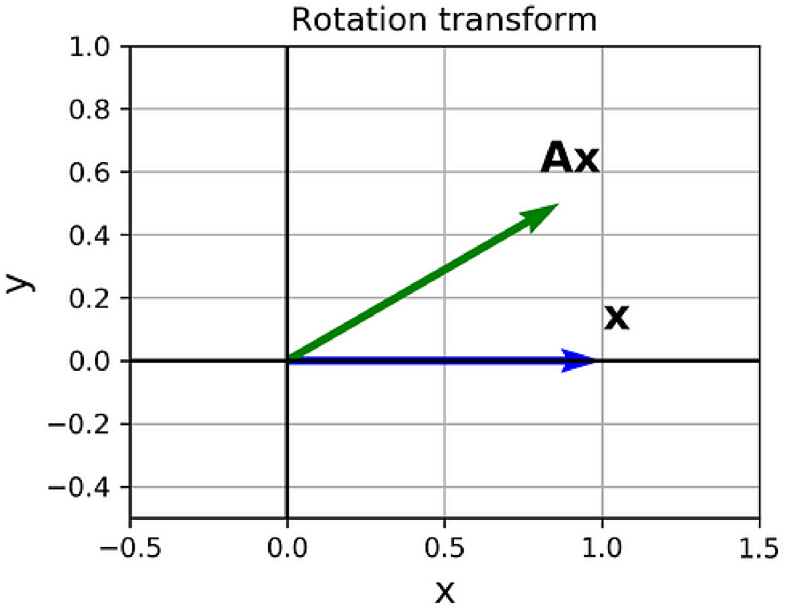
(m x n)

(n x 1)

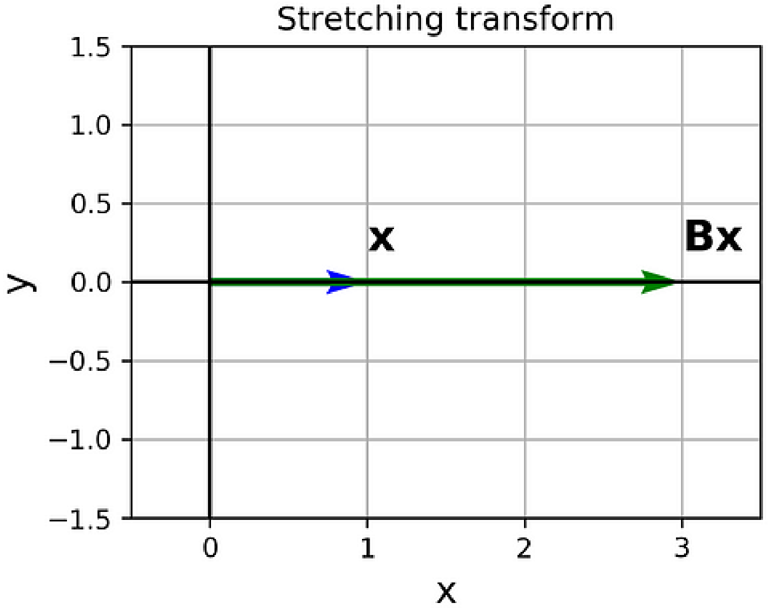
(m x 1) ← y = Ax

A is changing the direction and magnitude of vector x and we call this changed vector as a new vector y

$\theta = 30^0$



$k = 3$



$$x = \begin{bmatrix}1\\0\end{bmatrix}$$

$$A = \begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}$$

y = Ax

$$x = \begin{bmatrix}1\\0\end{bmatrix}$$

$$B = \begin{bmatrix}k & 0\\ 0 & 1\end{bmatrix}$$

y = Bx

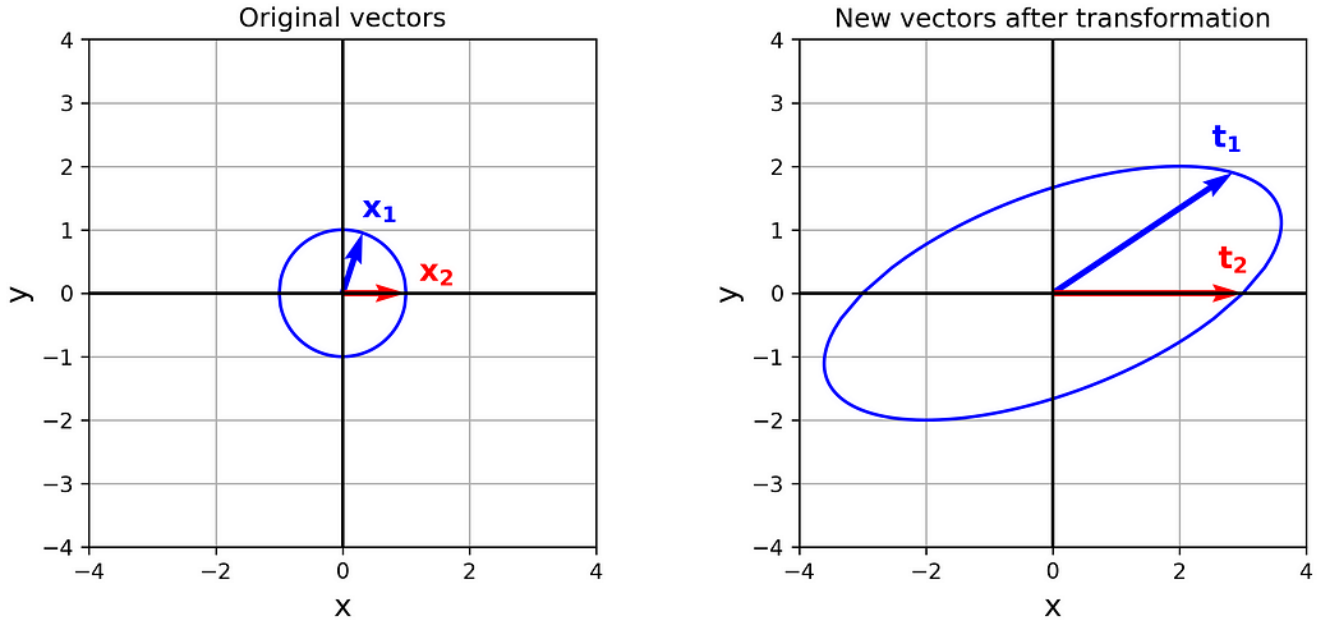
Reference: Reza Bagheri @ tds

Geometric significance of y = Ax

- Let $A = \begin{bmatrix} 3 & 2 \\ 0 & 2 \end{bmatrix}$ and $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ be the family of vectors that satisfy $x_1^2 + x_2^2 = 1$

A circle of unit radius in $\mathbb{R}^2$

- What happens if we apply the transformation of A on x, i.e. t = Ax?



Transforming both magnitude and direction!

The circle of unit radius in $\mathbb{R}^2$ is getting stretched to form an ellipse !!!

Causing a change in scale and rotation!

Geometric significance of y = Ax (Contd.)

➢ Let $A = \begin{bmatrix} 3 & 2 \\ 0 & 2 \end{bmatrix}$ and $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ be the family of vectors that satisfy $x_1^2 + x_2^2 = 1$

$$x^1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^1 \qquad x^2 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^2$$

$$\downarrow \qquad\qquad \downarrow$$

$$t^1 = Ax^1 \qquad t^2 = Ax^2$$

$$\downarrow \qquad\qquad \downarrow$$

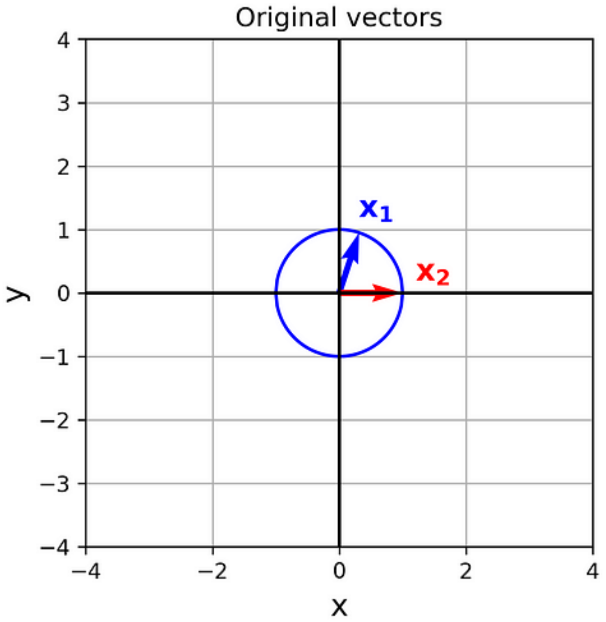
Changing both direction and magnitude

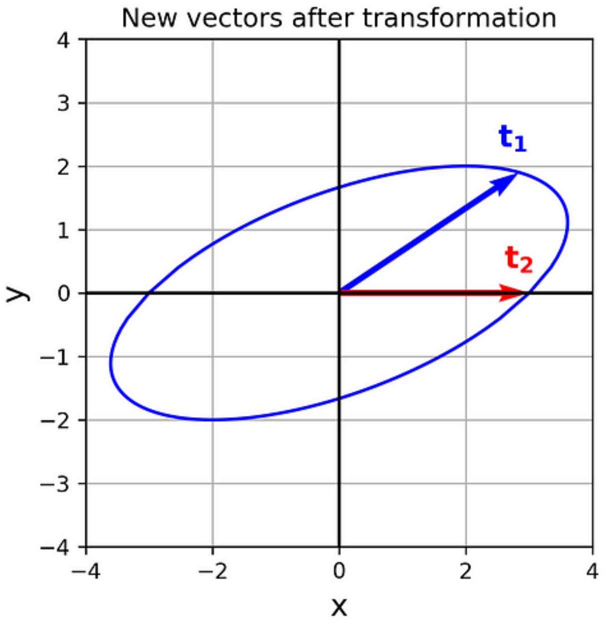
Changing only the magnitude

$$\downarrow$$

$$t^2 = \lambda x^2$$

λ is a scaler.





$$Ax^2 = \lambda x^2$$

Applying the transformation A on vector x^2 is resulting in only change in its magnitude

x^2 is a special vector

- Obviously linked to A.
- And also related to A with the scaler λ
- x^2 is Eigen Vector &
- λ is the Eigen Value of A

Eigen Values and Eigen Vectors of A

$$Au = \lambda u$$

Applying the transformation A on vector u is resulting in only change in its magnitude

Eigen Vector of a matrix A is the "non-zero" vector u, which when transformed using A results only in change of its magnitude and not its direction.

The quantity of the change is given by corresponding Eigen value λ.

What is the use of this Eigen Vector and Eigen Value?

One very important application is as follows:

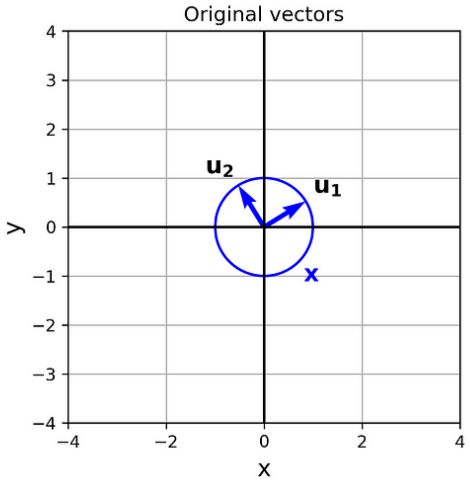
Matrix multiplication is transformed into scaler multiplication!!

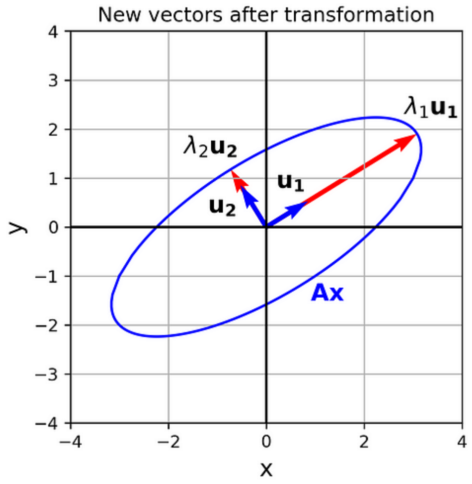
Geometric Significance of Eigen Vectors and Eigen Values

- For $A = \begin{bmatrix} 3 & 2 \\ 0 & 2 \end{bmatrix}$ we have two Eigen vectors and corresponding Eigen values

$$u_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } \lambda_1 = 3$$

$$u_2 = \begin{bmatrix} -0.89 \\ 0.44 \end{bmatrix} \text{ and } \lambda_2 = 2$$



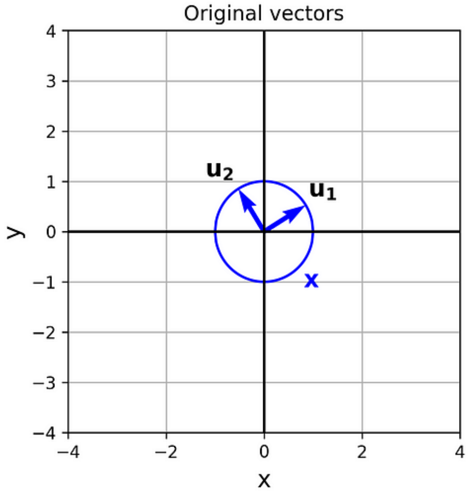


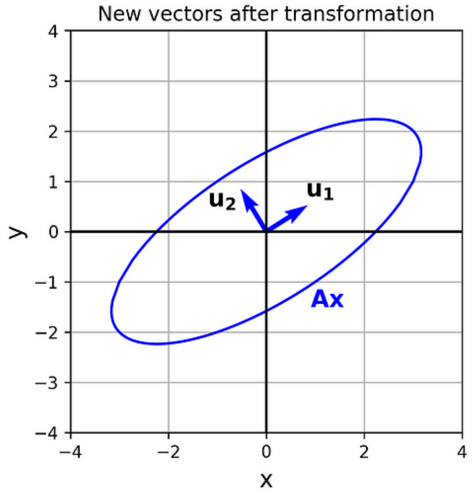
Only applicable for Symmetric matrices.

- However, for $A = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$

$$u_1 = \begin{bmatrix} 0.85 \\ 0.52 \end{bmatrix} \text{ and } \lambda_1 = 3.6$$

$$u_2 = \begin{bmatrix} -0.52 \\ 0.85 \end{bmatrix} \text{ and } \lambda_2 = 1.3$$





The circle of unit radius in $\mathbb{R}^2$ is getting stretched to form an ellipse along u_1 and u_2.

Eigen Vectors are the principle axes of Ellipse.

Eigen Value Decomposition

- A symmetric matrix A is orthogonally diagonalizable.

$$\mathrm{A} = \mathrm{PDP}^{\mathrm{T}}$$

(n x n) (n x n) (n x n)

- A will have n Eigen vectors. P is the set of Eigen vectors and D is diagonal matrix with n Eigen values as its diagonal.

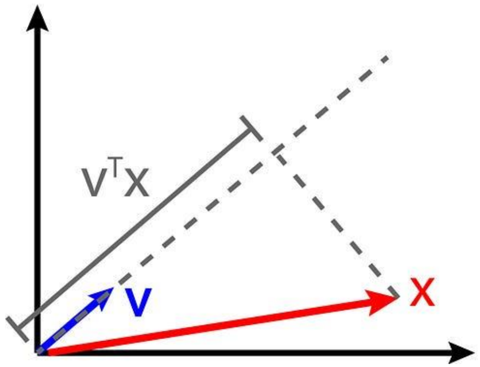
$$\mathrm{A} = \begin{bmatrix} u_1 & u_2 & u_n \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_n \end{bmatrix} \begin{bmatrix} u_1^T \\ u_2^T \\ u_n^T \end{bmatrix}$$

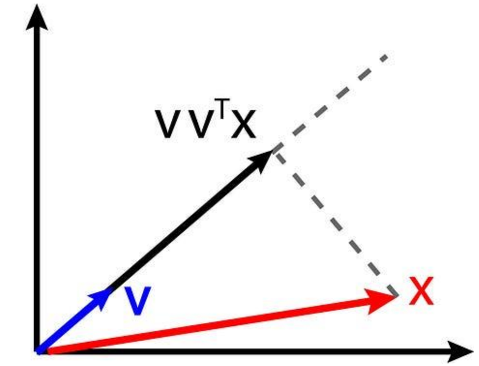
$$\mathrm{A}\mathrm{x} = \lambda_1 u_1 u_1^T \mathrm{x} + \lambda_2 u_2 u_2^T \mathrm{x} + \cdots + \lambda_n u_n u_n^T \mathrm{x}$$

Orthogonal projection of x in the direction of u_1

i.e. stretching in direction of u_1

Magnitude of stretching

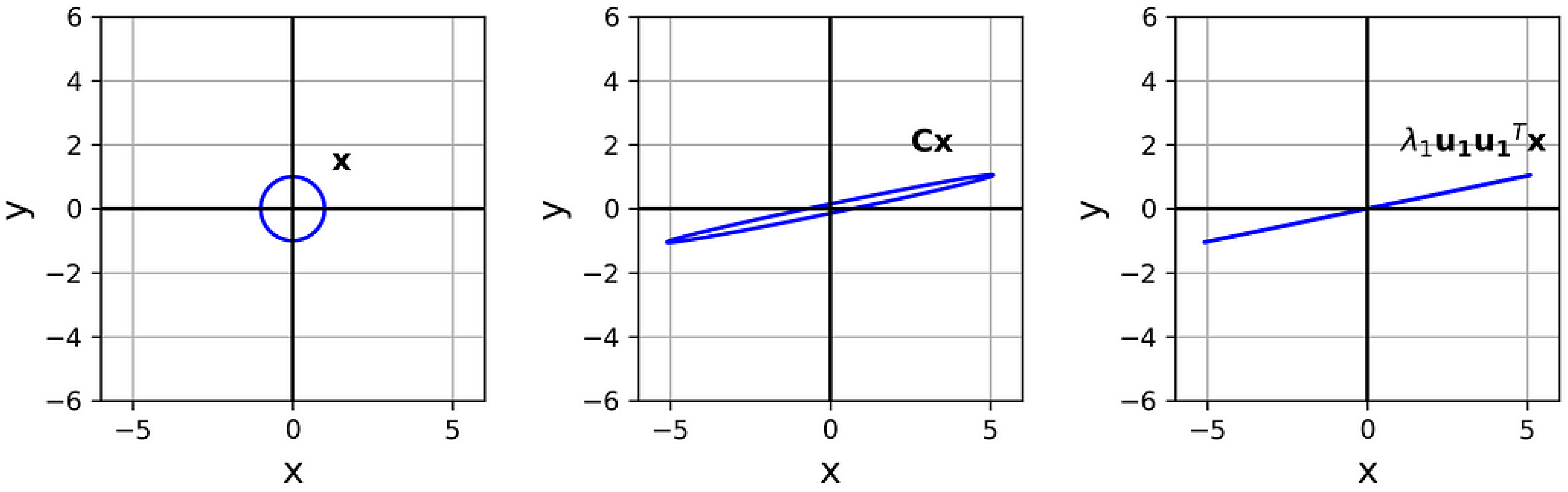




Why Eigen Value Decomposition of A?

$$\mathrm{C}\mathrm{x} = \lambda_1 u_1 u_1^T \mathrm{x} + \lambda_2 u_2 u_2^T \mathrm{x} + \cdots + \lambda_n u_n u_n^T \mathrm{x}$$

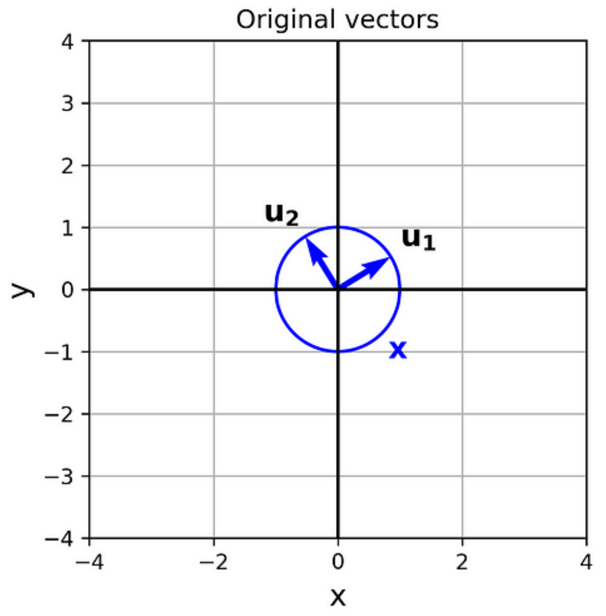
And $\lambda_1 \gg$ all other Eigen Values, I can approximate $\mathrm{C}\mathrm{x} \approx \lambda_1 u_1 u_1^T \mathrm{x}$

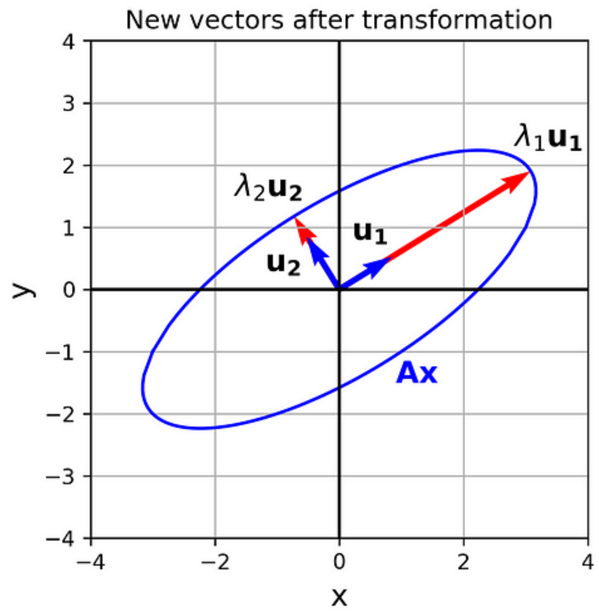


And thus, save a lot of computation. As a result we have a lot of applications !!!

Is Eigen Value Decomposition of A sufficient?

- Though Eigen Value Decomposition is extremely useful, it is only applicable when A is symmetric matrix.
- This is not the case in the real world. Thus, we need a more generic method that can "extend" the concepts of Eigen Value Decomposition to a non-symmetric matrix.

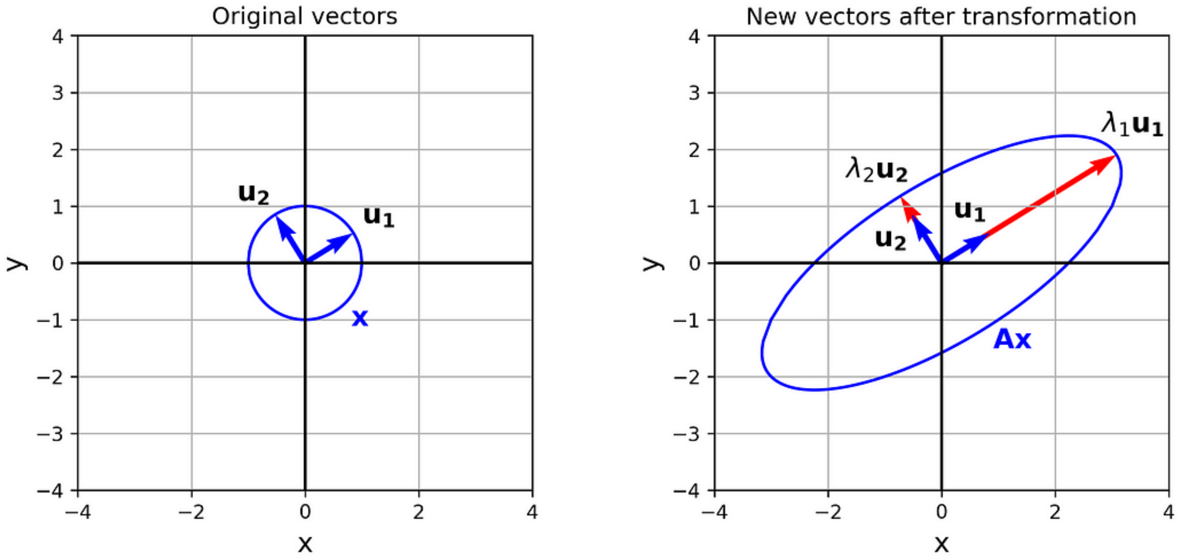




- When a symmetric matrix A is applied on the family of vectors x such that $||x|| = 1$, the result is the family of vectors t which is nothing but the circle stretched along Eigen Vectors u_i by a magnitude of Eigen values λ_i.
- It is also applicable to any number of dimensions.
- We use this geometric significance and some linear algebra to derive the directions of stretching & magnitudes when A is non-symmetric.

Singular Values

The situation when A is symmetric



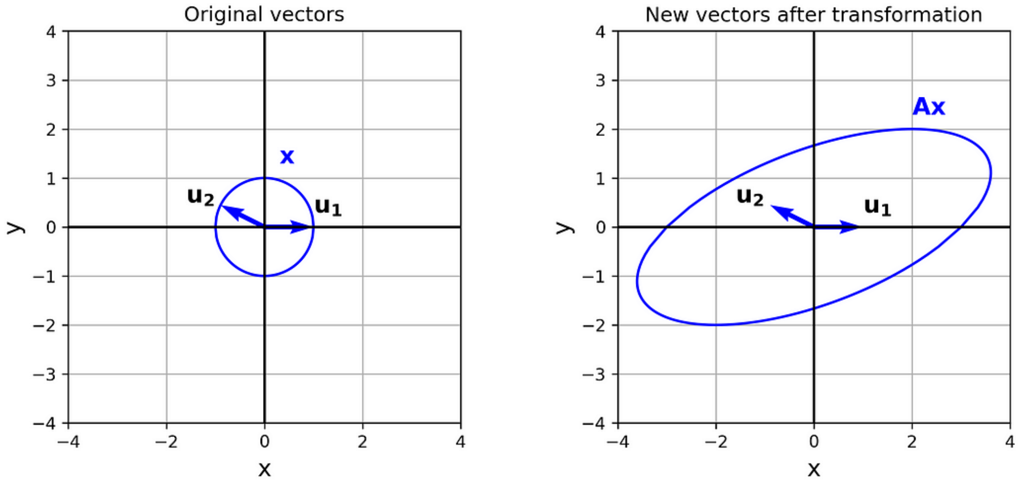
- We know, for any type of matrix A, the transformation $\mathrm{A^TA}$ is symmetric.
- Let the Eigen vectors of $\mathrm{A^TA}$ be $\mathrm{v_1, v_2, \dots v_n}$. Then, for the family of vectors x such that $||\mathrm{x}|| = 1$, it can be shown that, $\mathrm{Av_i}$ gives the direction of stretching as a result of Ax and the magnitude of stretching in that direction is given by $\sigma_\mathrm{i} = \sqrt{\lambda_i}$, where λ_i is the Eigen value of $\mathrm{A^TA}$ corresponding to its Eigen vector $\mathrm{v_i}$. σ_i is called the singular value.

Geometric Significance of Eigen Vectors and Eigen Values

- For $A = \begin{bmatrix} 3 & 2 \\ 0 & 2 \end{bmatrix}$ we have two Eigen vectors and corresponding Eigen values

$$\mathrm{u}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } \lambda_1 = 3$$

$$\mathrm{u}_2 = \begin{bmatrix} -0.89 \\ 0.44 \end{bmatrix} \text{ and } \lambda_2 = 2$$

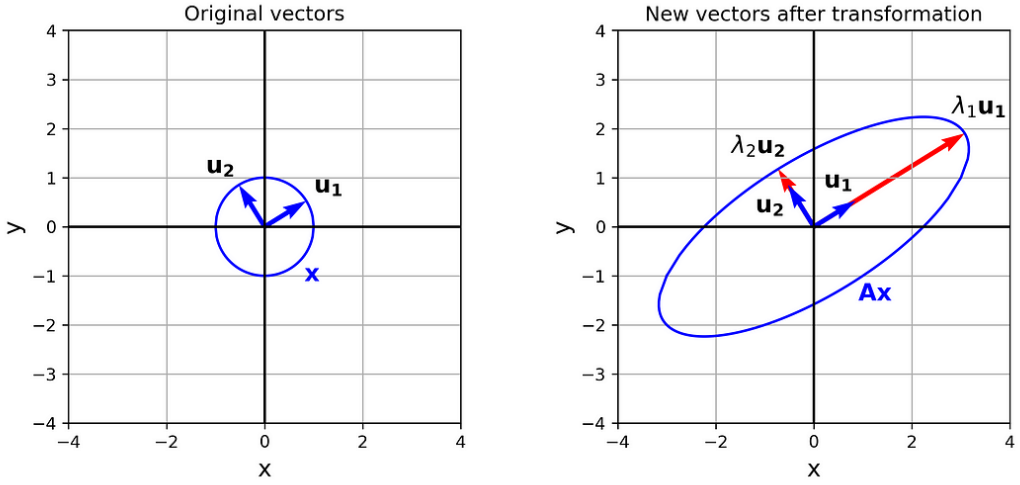


Only applicable for Symmetric matrices.

- However, for $A = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$

$$\mathrm{u}_1 = \begin{bmatrix} 0.85 \\ 0.52 \end{bmatrix} \text{ and } \lambda_1 = 3.6$$

$$\mathrm{u}_2 = \begin{bmatrix} -0.52 \\ 0.85 \end{bmatrix} \text{ and } \lambda_2 = 1.3$$



The circle of unit radius in $\mathbb{R}^2$ is getting stretched to form an ellipse along u_1 and u_2.

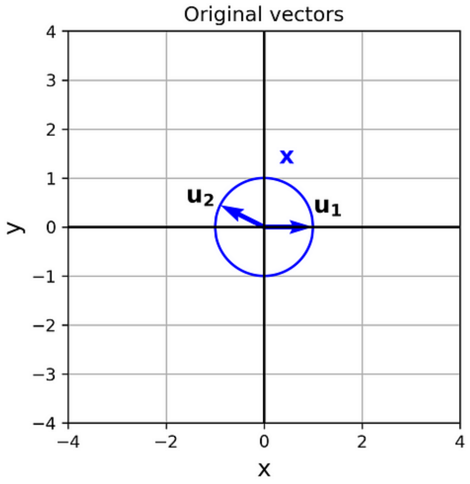
Eigen Vectors are the principle axes of Ellipse.

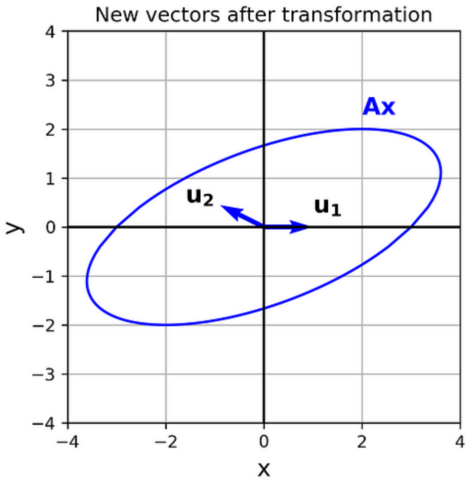
Geometric Significance of Eigen Vectors and Eigen Values

- For $A = \begin{bmatrix} 3 & 2 \\ 0 & 2 \end{bmatrix}$ we have two Eigen vectors and corresponding Eigen values

$$u_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } \lambda_1 = 3$$

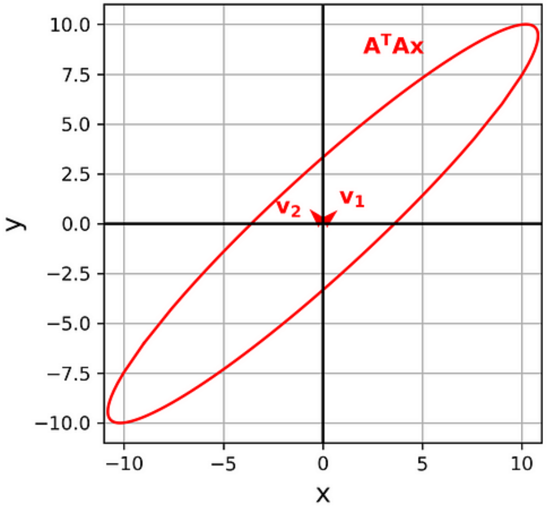
$$u_2 = \begin{bmatrix} -0.89 \\ 0.44 \end{bmatrix} \text{ and } \lambda_2 = 2$$

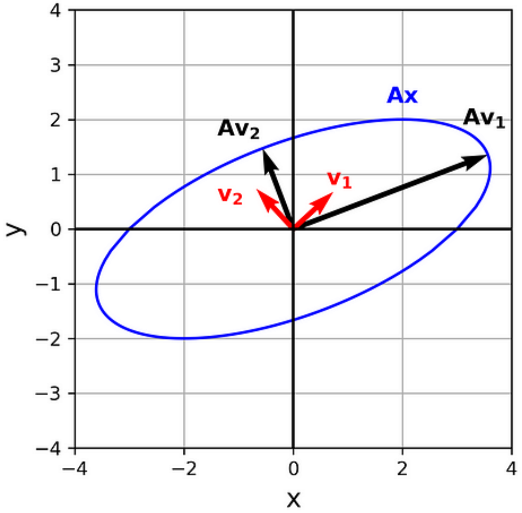




Eigen vectors of A were not pointing towards any significant direction

Eigen vectors of A^TA (v_i) are pointing towards the directions of stretching due to A^TAx because A^TA is symmetric

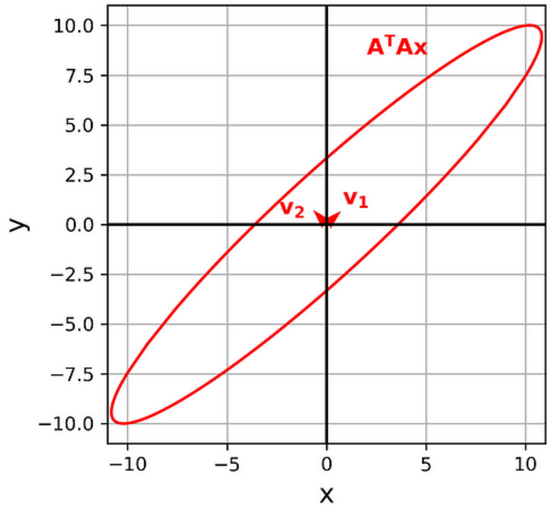


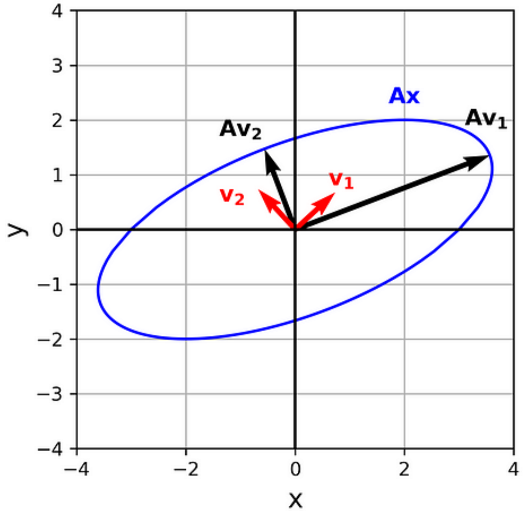


Using the Eigen vectors of A^TA, we obtain Av_1 and Av_2 which are pointing towards the directions of stretching due to Ax

Singular Value Decomposition

Eigen vectors of $\mathrm{A^TA}$ $(\mathrm{v_i})$ are pointing towards the directions of stretching due to $\mathrm{A^TAx}$ because $\mathrm{A^TA}$ is symmetric





Using the Eigen vectors of $\mathrm{A^TA}$, we obtain $\mathrm{Av_1}$ and $\mathrm{Av_2}$ which are pointing towards the directions of stretching due to Ax

- If A is symmetric, then what happens to this result? A being symmetric $\Longrightarrow \mathrm{A^T = A}$
- Let λ_i be the Eigen value corresponding to Eigen vector u_i of the matrix A
- Then, $(\mathrm{A^TA})u_i$
- This means, u_i is also Eigen vector of $\mathrm{A^TA}$ and its corresponding Eigen Value is λ_i^2
- That means, u_i and v_i will point in same direction of stretching.
- Thus, a more generic approach is established than Eigen Value Decomposition.

This approach is called as Singular Value Decomposition

Singular Value Decomposition (Contd.)

$$\mathrm{A} = \mathrm{U\Sigma V^T}$$

- A is (m x n) matrix.
- V is the square matrix of Eigen Values of $\mathrm{A^TA}$. It will be (n x n)
- Each column vector in V is orthogonal to each other and normalized, i.e. orthonormal. Thus, the matrix V is orthogonal matrix.
- Σ is (m x n) diagonal matrix of n singular values $\sigma_1, \sigma_2, \ldots \sigma_n$.
- U is (m x m) orthogonal matrix (refer to previous lecture for explanation of its derivation)

$$\boldsymbol{\Sigma} = \left.\underbrace{\begin{bmatrix} \sigma_1 & 0 & \ldots & 0 & 0 & \ldots & 0 \\ 0 & \sigma_2 & \ldots & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & \sigma_r & 0 & \ldots & 0 \\ 0 & 0 & \ldots & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & 0 & 0 & \ldots & 0 \end{bmatrix}}_{\text{n columns}}\right\}\text{m rows}$$

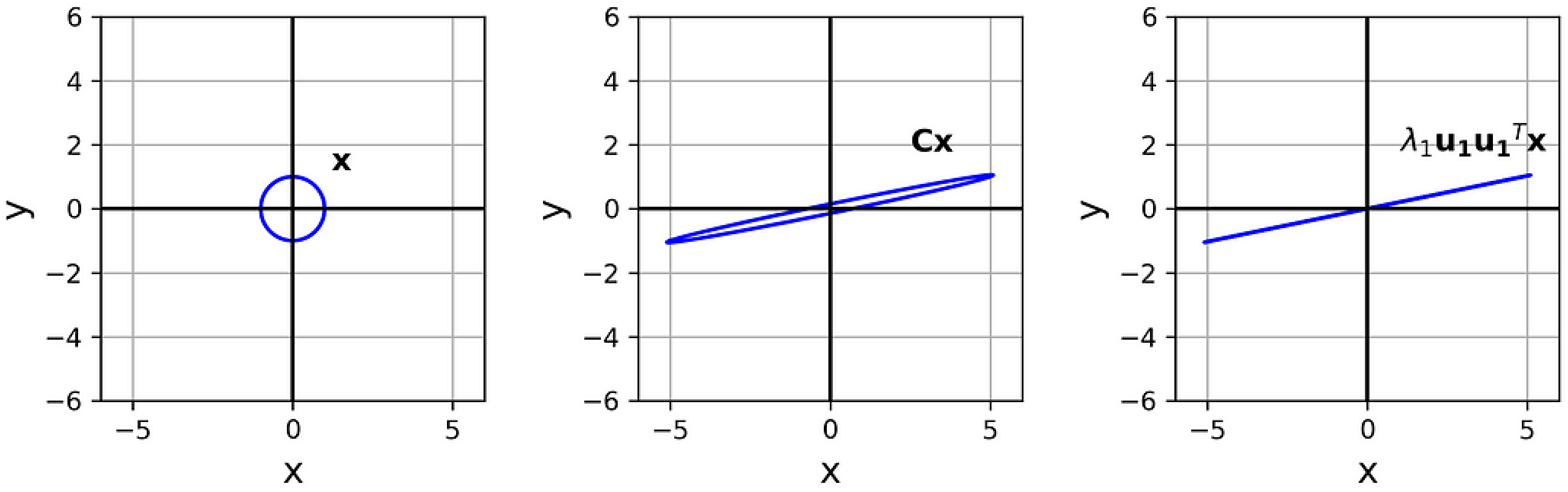
$$\mathrm{A}\mathbf{x} = \sigma_1 u_1 \mathrm{v}_1^T \mathbf{x} + \sigma_2 u_2 \mathrm{v}_2^T \mathbf{x} + \cdots + \sigma_r u_r \mathrm{v}_r^T \mathbf{x}$$

The singular values are always arranged in decreasing order of significance

Why Singular Value Decomposition of A?

$$\mathrm{C}\,\mathrm{x} = \lambda_1 u_1 u_1^T \mathrm{x} + \lambda_2 u_2 u_2^T\, \mathrm{x} + \cdots + \lambda_n u_n u_n^T\, \mathrm{x}$$

And $\lambda_1 \gg$ all other Eigen Values, I can approximate $\mathrm{Cx} \approx \lambda_1 u_1 u_1^T \mathrm{x}$



And thus, save a lot of computation. As a result we have a lot of applications !!!

Why Singular Value Decomposition of A?

$$\mathrm{C}\mathrm{x} = \lambda_1 u_1 u_1^T \mathrm{x} + \lambda_2 u_2 u_2^T \mathrm{x} + \cdots + \lambda_n u_n u_n^T \mathrm{x}$$

And $\lambda_1 \gg$ all other Eigen Values, I can approximate $\mathrm{Cx} \approx \lambda_1 u_1 u_1^T \mathrm{x}$

$$\mathrm{A}\mathrm{x} = \sigma_1 u_1 \mathrm{v}_1^T \mathrm{x} + \sigma_2 u_2 \mathrm{v}_2^T \mathrm{x} + \cdots + \sigma_r u_r \mathrm{v}_r^T \mathrm{x}$$

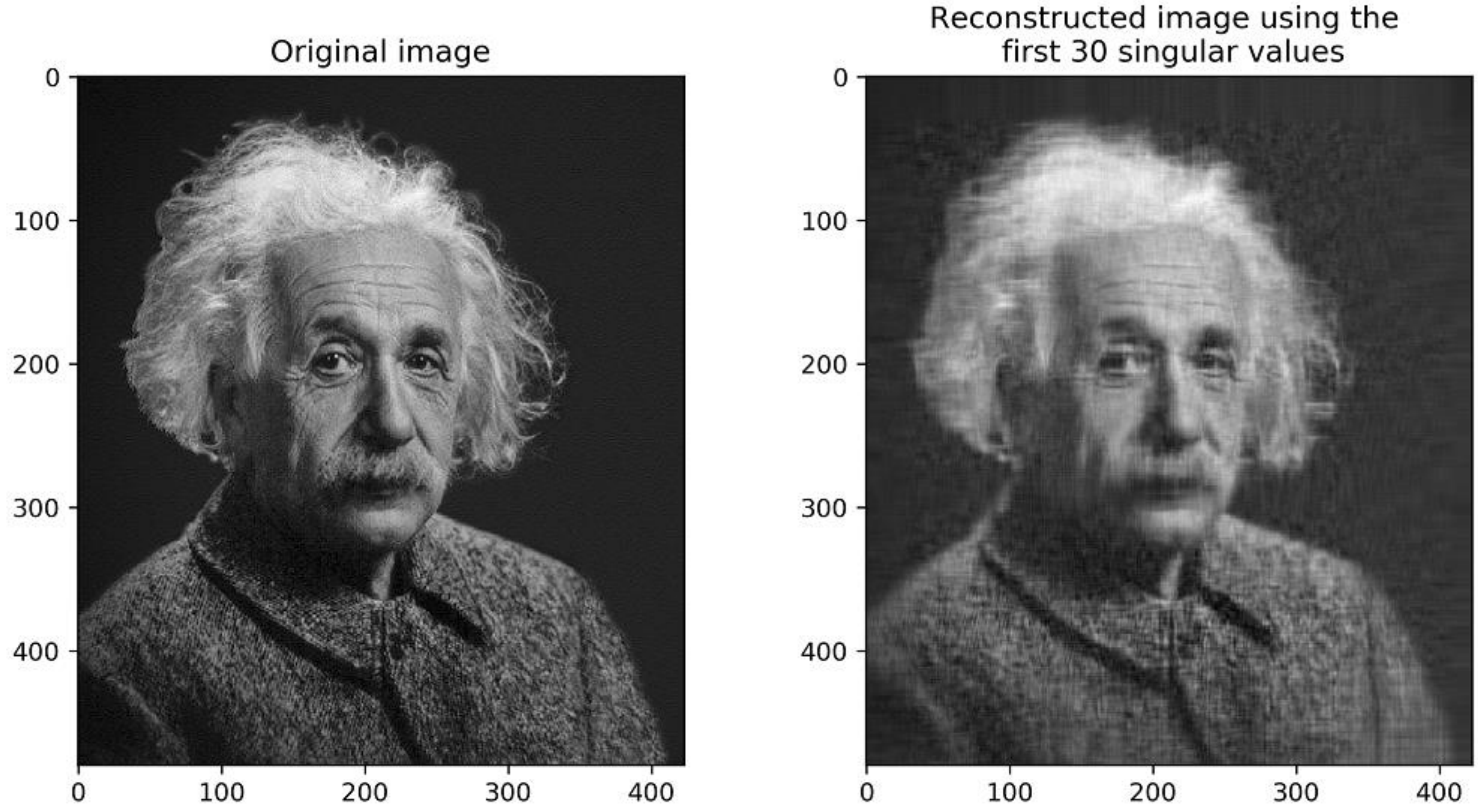
And $\sigma_1 \gg$ all other Singular Values, I can approximate $\mathrm{Ax} \approx \sigma_1 u_1 \mathrm{v}_1^T \mathrm{x}$

$$\mathrm{A} = \sigma_1 u_1 \mathrm{v}_1^T + \sigma_2 u_2 \mathrm{v}_2^T + \cdots + \sigma_r u_r \mathrm{v}_r^T$$

With relevant Singular values, we can reconstruct A: $\mathrm{A}_{\mathrm{approx}} = \sigma_1 u_1 \mathrm{v}_1^T + \sigma_2 u_2 \mathrm{v}_2^T$

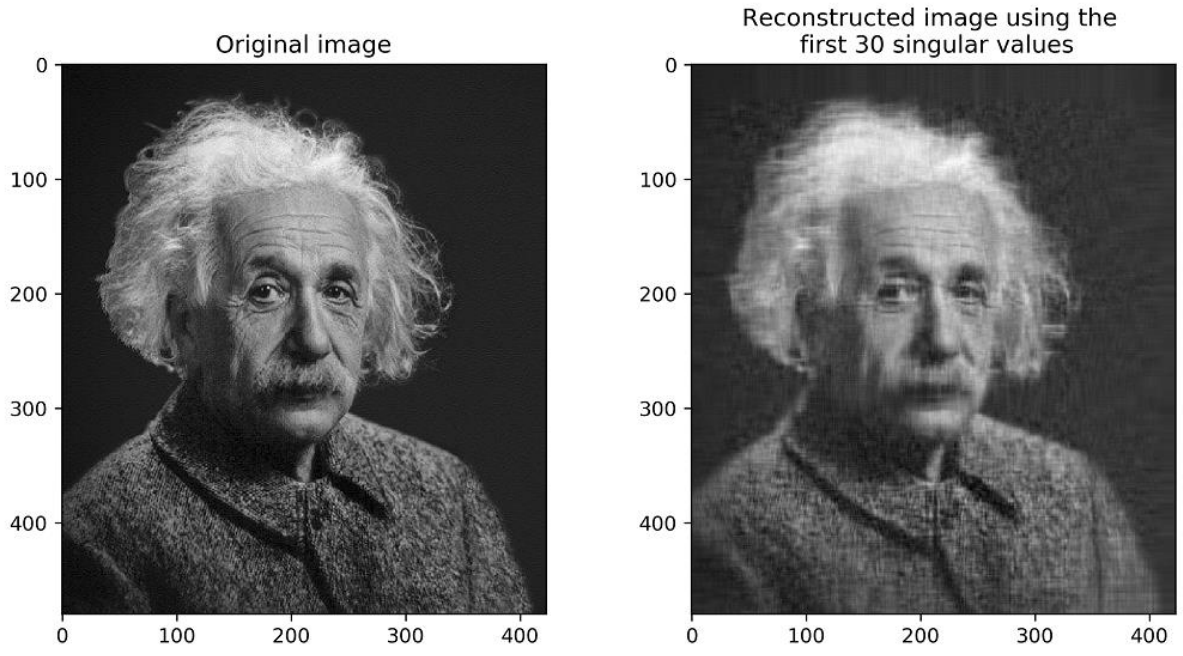
And thus, save a lot of computation. As a result we have a lot of applications !!!

Application 1: Image Compression



Reference: Reza Bagheri @ tds

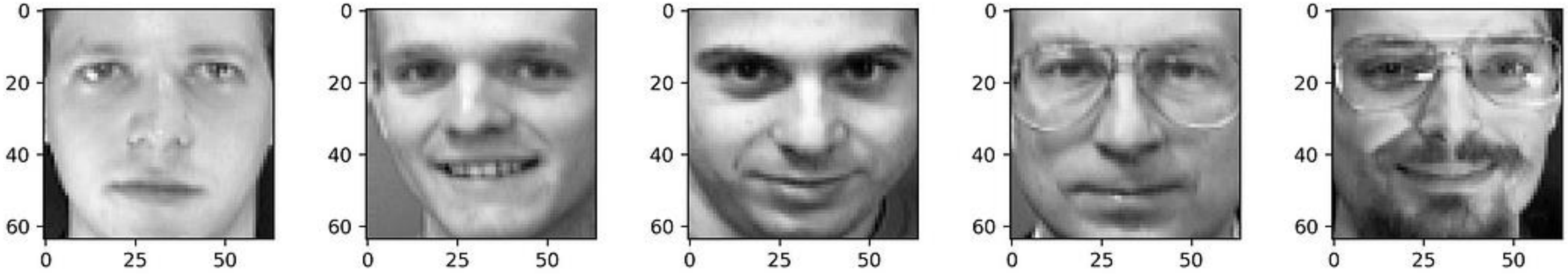
Application 1: Image Compression (Contd.)

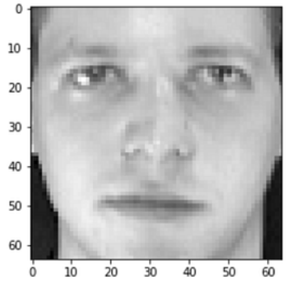


- The original matrix is 480×423.
- So we need to store 480×423=203040 values.
- After SVD each u_i has 480 elements and each v_i has 423 elements.
- To be able to reconstruct the image using the first 30 singular values we only need to keep the first 30 σ_i, v_i and u_i.
- It means storing 30×(1+480+423)=27120 values.
- **This is roughly 13% of the number of values required for the original image.**
- So using SVD we can have a good approximation of the original image and save a lot of memory.

Reference: Reza Bagheri @ tds

Application 2: Face detection





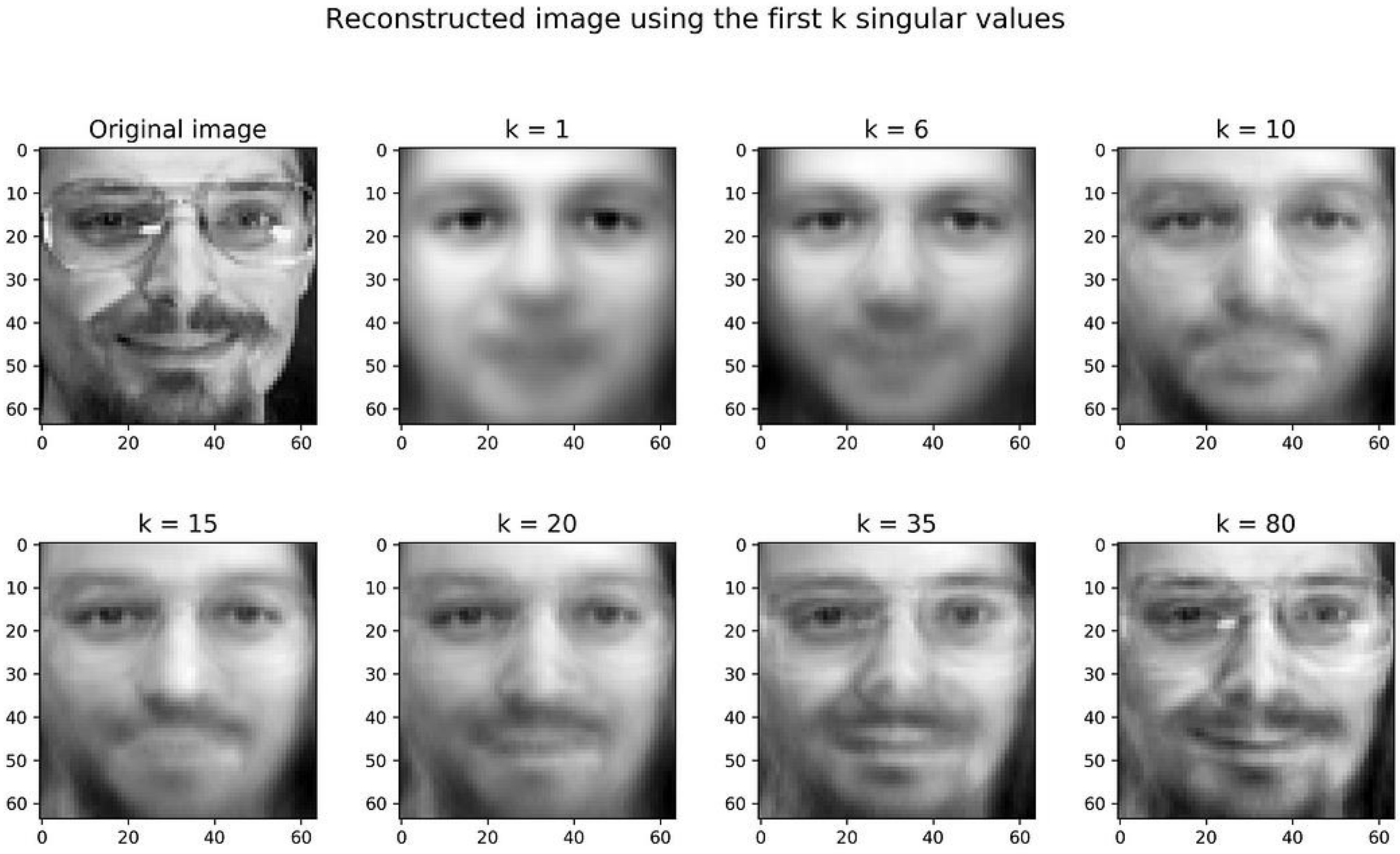
$$= \begin{bmatrix} 0.310 & 0.368 & \dots & 0.306 \\ 0.343 & 0.405 & \dots & 0.314 \\ \vdots & \vdots & \dots & \vdots \\ 0.202 & 0.207 & \dots & 0.157 \end{bmatrix} \rightarrow \mathbf{f}_1 = \begin{bmatrix} 0.310 \\ 0.368 \\ \vdots \\ 0.306 \\ 0.343 \\ 0.405 \\ \vdots \\ 0.314 \\ \vdots \\ 0.202 \\ 0.207 \\ \vdots \\ 0.157 \end{bmatrix}$$

$$\mathbf{M} = [\mathbf{f}_1 \quad \mathbf{f}_2 \quad \mathbf{f}_3 \quad \dots \quad \mathbf{f}_{400}]$$

- ➢ Decompose M using SVD and reconstruct the images with significant Singular Values.
- ➢ It is observed that each Singular Value corresponds to a specific feature in the space.
- ➢ These features are used for Face detection!!

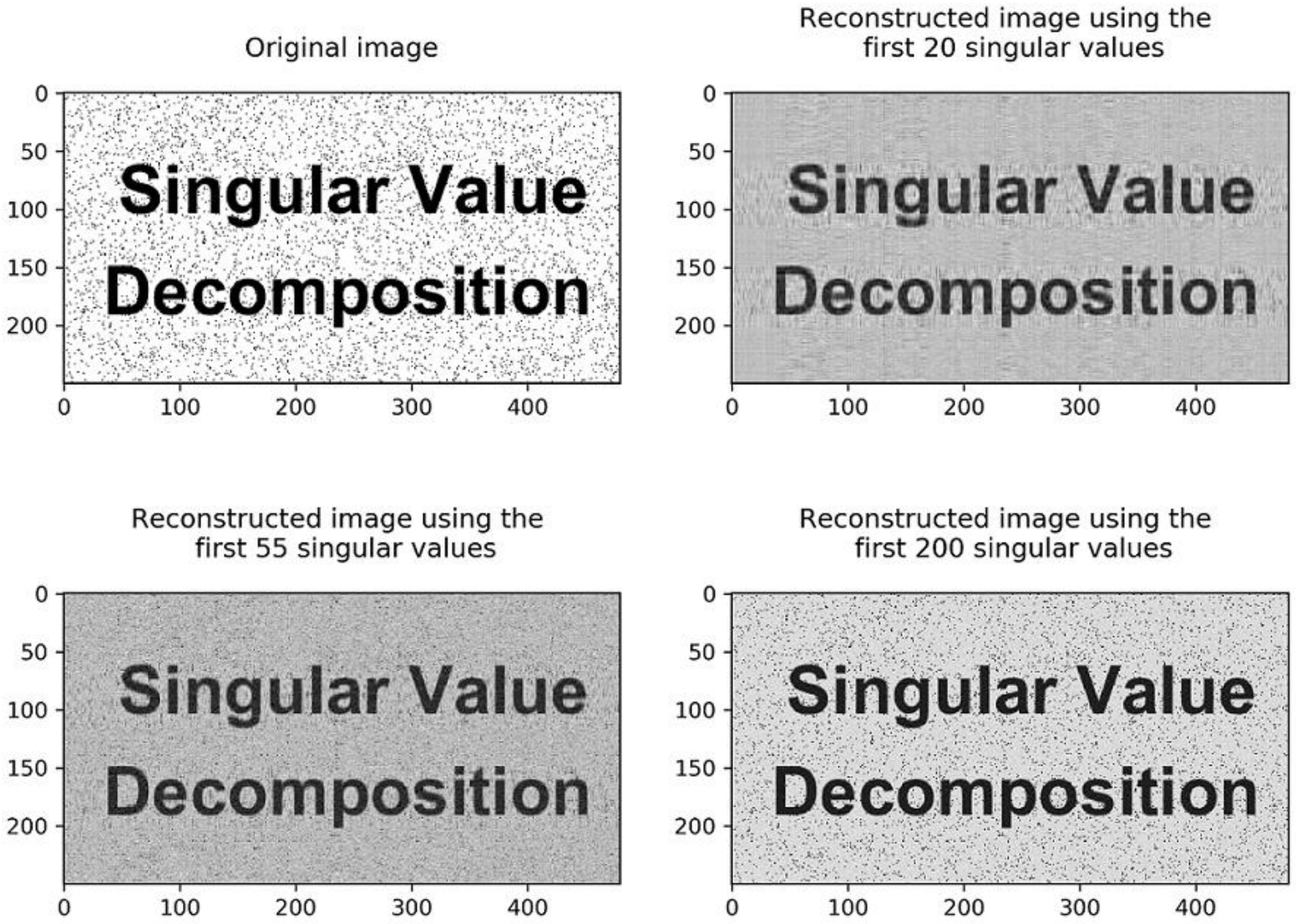
Reference: Reza Bagheri @ tds

Application 2: Face detection (Contd.)



Application 3: Noise Reduction

- Noise as a feature in the image has very less singular value.
- Thus, reconstruction of images with only significant singular values helps in filtering the noise.



[Reference: Reza Bagheri @ tds]()

Application 4: Background Removal

- ➢ Each frame is an image. We can flatten it and convert into columns.
- ➢ Multiple frames (Video) becomes the matrix A.
- ➢ We can apply SVD and reconstruct using significant singular values.
- ➢ Since background is common to all frames, it becomes the predominant feature.

[Reference: Background Extraction by Mubashir ul Islam | Medium]

Some more Applications of SVD

- Calculate inverse of A
- Matrix power calculation becomes simple: $A^m = PD^mP^T$
- PCA using SVD
- Drug target interactions
- $PM_{2.5}$ forecasting
- Recommender systems
- Topic Modelling
- Online Transactions

Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables (entities each of which takes on various numerical values) into a set of values of linearly uncorrelated variables called principal components.

From a simplified perspective, PCA transforms data linearly into new properties that are not correlated with each other. For ML, positioning PCA as feature extraction may allow us to explore its potential better than dimension reduction.

Thank you!