M5: Dimensionality Reduction via Principal Component Analysis (**PCA**) - [another popular (unsupervised ML) appn. of linear algebra]

Manikandan Narayanan

Week 9 (Sep 22-, 2025)

PRML Jul-Nov 2025 (Grads Section)

Acknowledgment of Sources

- Slides based on content from related
  - Courses:
    - IITM – Profs. Arun/Harish/Chandra's PRML offerings (slides, quizzes, notes, etc.), Prof. Ravi's "Intro to ML" slides – cited respectively as [AR], [HR]/[HG], [CC], [BR] in the bottom right of a slide.
    - India – NPTEL PR course by IISc Prof. PS. Sastry (slides, etc.) – cited as [PSS] in the bottom right of a slide.

  - Books:
    - PRML by Bishop. (content, figures, slides, etc.) – cited as **[CMB]**
    - Pattern Classification by Duda, Hart and Stork. (content, figures, etc.) – [DHS]
    - Mathematics for ML by Deisenroth, Faisal and Ong. (content, figures, etc.) – [DFO]
    - Information Theory, Inference and Learning Algorithms by David JC MacKay – [DJM]

Outline for Module M5

- M5. Dimensionality Reduction
  - **M5.0 Introduction**
  - M5.1 PCA
    - Intuition for two formulations
    - Maximizing variance formulation
    - Minimizing error formulation
  - M5.2 PCA applications/extensions/variants (very brief)

Context: Two unsupervised ML problems

- Unsupervised ML: Recognize patterns in the dataset (set of n data points in $\mathbb{R}^d$) without any labels on the data points.

- Dimensionality reduction:
  - Transforms data points from high to low dimensions without much loss of "information", assuming the data points lie "effectively in/close to" a low-dim. manifold of the original space.
  - Many approaches possible: **<u>PCA</u>**, t-SNE, UMAP, **Laplacian Eigenmaps (spectral)**, etc.

- Clustering:
  - Grouping n objects into k clusters based on their similarity
  - Again, many approaches possible: k-means, hierarchical, **spectral**, etc.

Dimensionality Reduction

- Transforms data points from high to low dimensions without much loss of "information", assuming the data points lie "effectively in/close to" a low-dim. manifold of the original space.

- A popular unsupervised PR/ML task like clustering with many appns.:
    - Visualization of data (in different domains)
    - Feature extraction/engineering (preprocessing step for classification/regression)
    - Lossy compression/Denoising
    - Preprocessing step for clustering (think spectral)
    - Identifying unknown confounding factors

- Our approach in this lecture: **Mostly [CBM]**, and (deterministic/hard) PCA

(but just be aware that probabilistic/soft PCA (based on linear Gaussian) also exists).

Principal Component Analysis or PCA

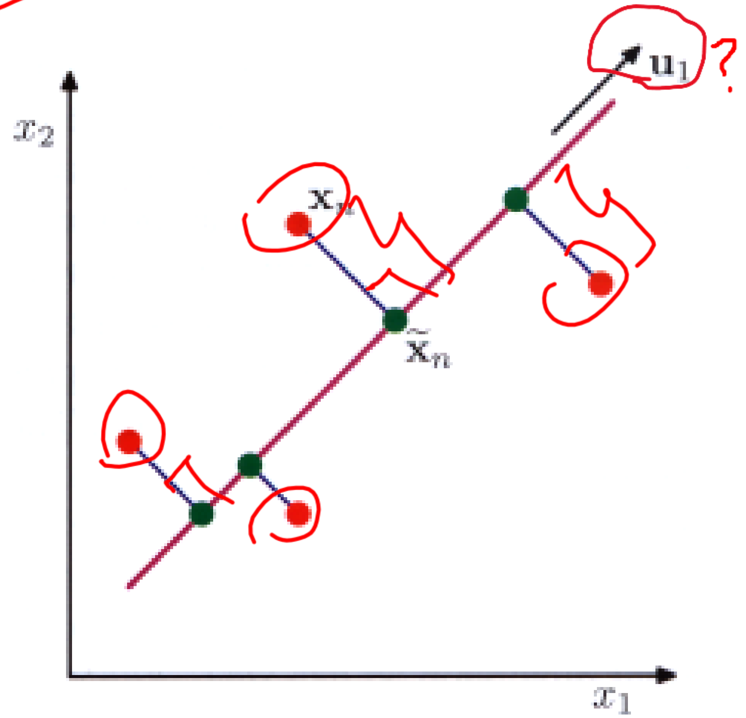
- A widely-used dimensionality reduction method, aka Karhunen-Loeve transform
  - PCA represents D-dimensional data points using M-dimensional vectors with M < D, while maximizing variance (Hotelling, 1933) or equivalently minimizing error (Pearson, 1901), using the spectrum of the data matrix.

- Why is it called **linear** PCA?
  - Orthogonal projection of data onto a low-dimensional **linear** subspace, known as the principal subspace, s.t. the variance of the projected data is maximized or equivalently the mean squared distance between data pts and their projections is minimized.
  - Can be viewed as a continuous latent variable model (LVM) with **linear** Gaussian assumptions (with links to EM algo, which may be seen later during mixture density estimation if time permits)!

[CBM]

PCA in pictures and notations

$D=2 \rightarrow M=1$



I/p: $x_1, \ldots\ldots, x_N \in \mathbb{R}^D$

O/p: $\tilde{x}_1, \ldots\ldots, \tilde{x}_N \in \mathbb{R}^D$

s.t.: 1) $\tilde{x}_n$ "approx." x_n

2) $\tilde{x}$ can be repr. using $M<D$ dimensions.

[CBM]

Why dimension reduction works?: Latent space; latent variables/features and their extraction

100

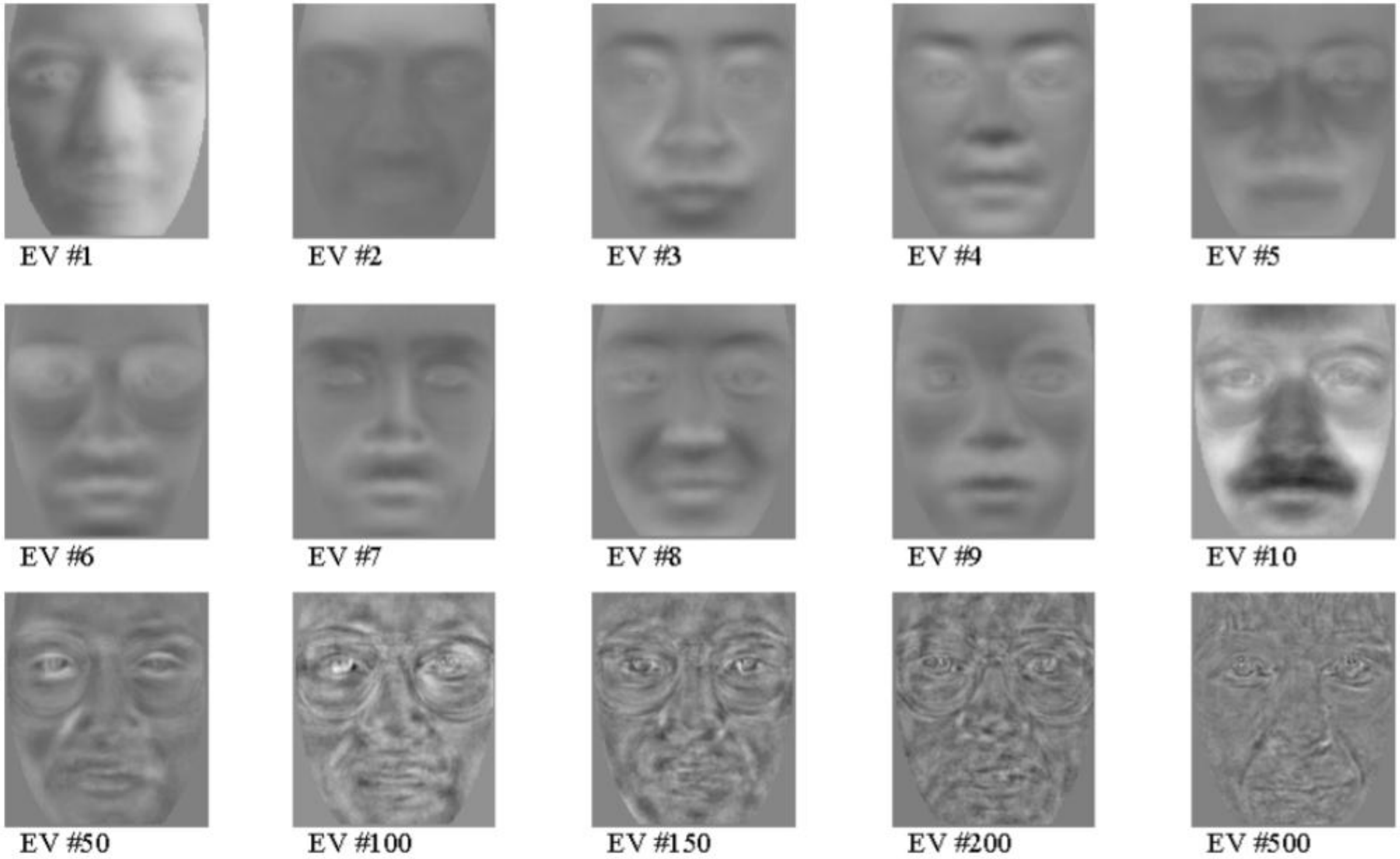
100

D = 10,000

M = 900 (30×30)

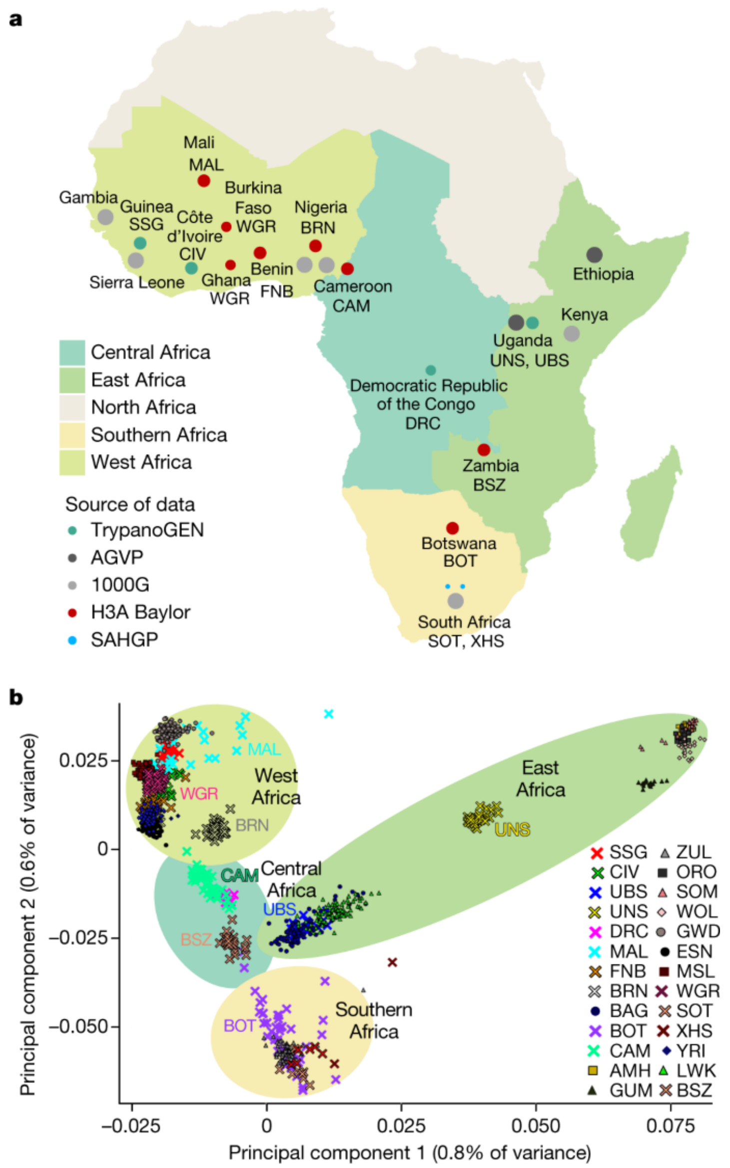
[CBM]

Face recogn.: Eigenfaces (latent space)



[From https://www.cs.colostate.edu/evalfacerec/papers/eemcvcsu.pdf]

An example application: bioinf. visualization



[From Chowdhury et al. Nature 2020]

Single-cell revolution in biology – visualized in reduced dimensions

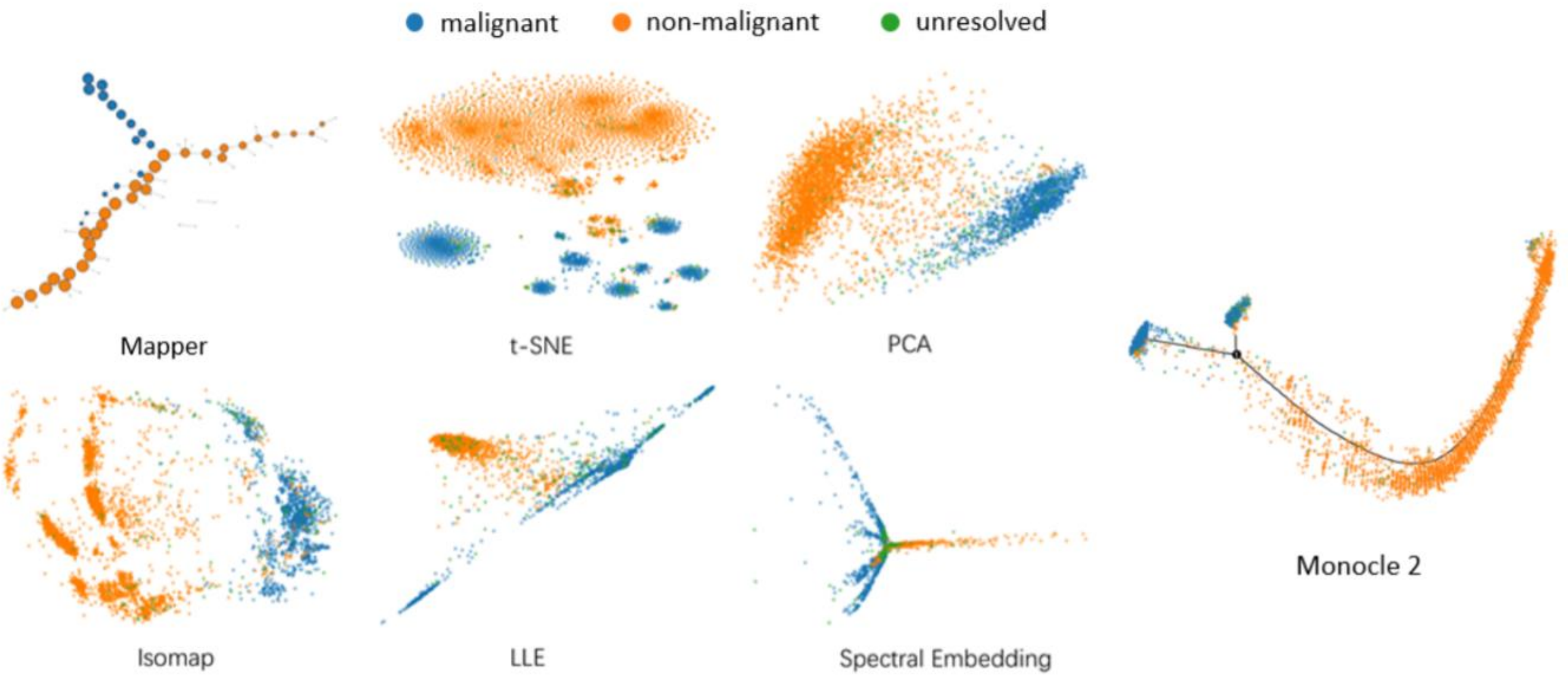


Figure 2. Visualization of melanoma cells.

[From Wang et al. Pac Symp Biocomput. 2019; 24:350-361]

Outline for Module M5

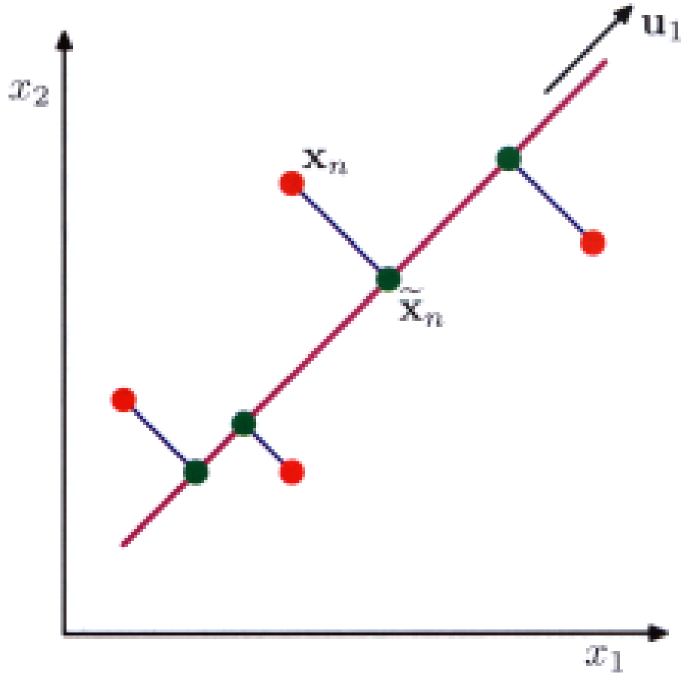
- M5. Dimensionality Reduction
  - M5.0 Introduction
  - **M5.1 PCA**
    - **Intuition for two formulations**
    - Maximizing variance formulation
    - Minimizing error formulation
  - M5.2 PCA applications/extensions/variants (very brief)

1) The M-dimensional subspace has to pass thru’ the center of the data cloud (average $\bar{x}$)!

- What if M=0?
- What if M=1?
- Illustrate in board (that u_1 passes thru’ $\bar{x}$)

2) PCA in pictures, and notation $(M=1)$

(Given that u_1 passes thru' $\bar{x}$, what is the optimal direction (rotation angle) of u_1?)



I/p: $x_1, \ldots, x_N \in \mathbb{R}^D$

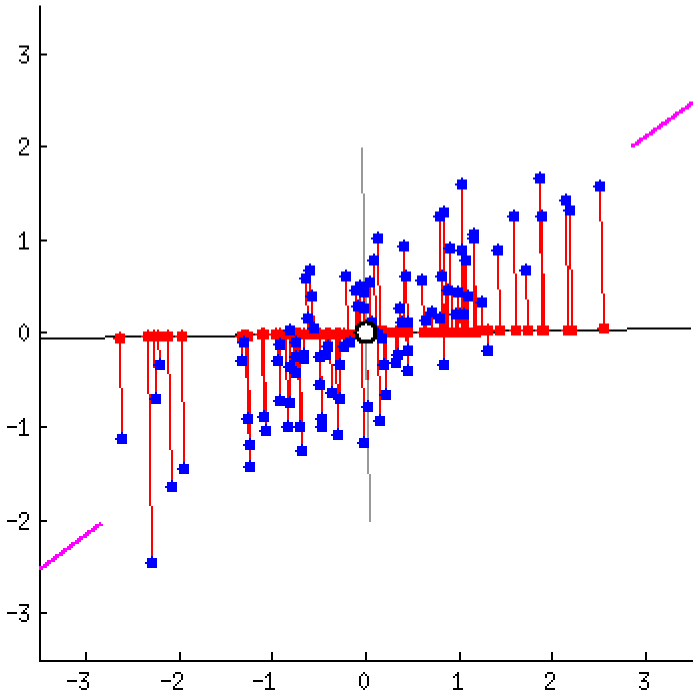
O/p: Find $u_1 \in \mathbb{R}^D$ s.t.

$$\boxed{\mathrm{Var}(x_1^T u_1, \ldots, x_N^T u_1) \text{ is maximized } (\& \ \|u_1\| = 1)}$$

$$\Longleftrightarrow \boxed{\min \frac{1}{N} \sum_{n=1}^{N} \| \tilde{x}_n - x_n \|^2}$$

[CBM]

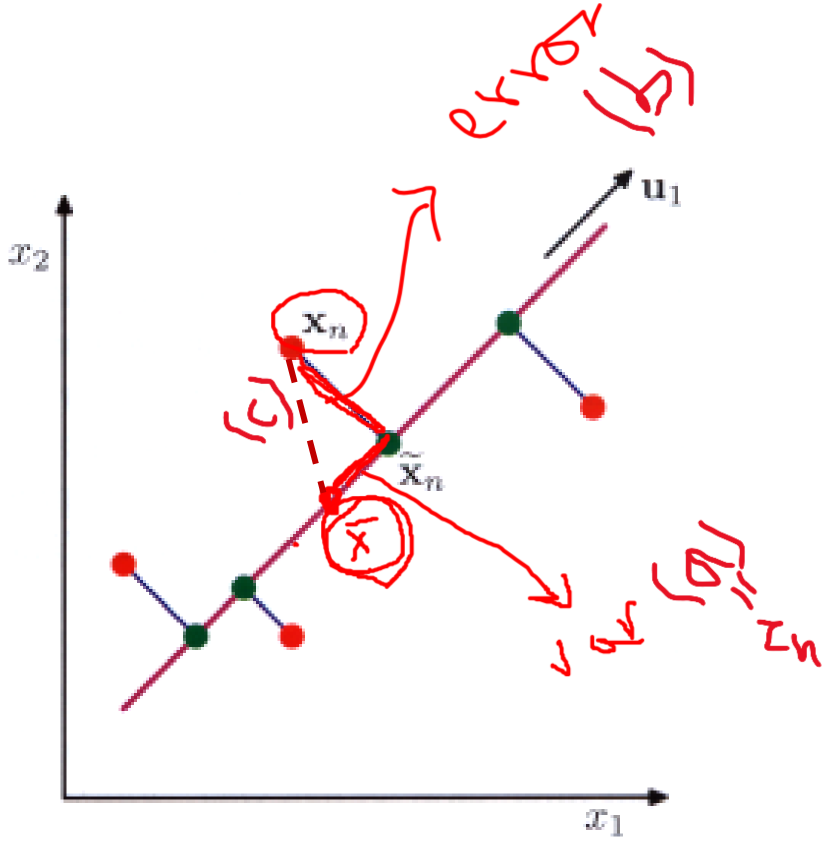
3) Animation of the two equiv. objectives!



https://stats.stackexchange.com/a/140579

(Also, show book example for 3D -> 2D reduction!)

4) Formal statement of the two objectives (M=1)



I/p: $x_1, \dots, x_N \in \mathbb{R}^D$

O/p: Find $u_1 \in \mathbb{R}^D$ s.t. $\mathrm{Var}(x_1^T u_1, \dots, x_N^T u_1)$ is maximized (& $\|u_1\|=1$)

$$\max \ \mathrm{Var}(\{z_n\}) = \mathrm{Var}(\{x_n^T u_1\}) \iff \min \frac{1}{N}\sum_{n=1}^{N} \| \underbrace{\tilde{x}_n}_{\bar{x} + z_n u_1} - x_n \|^2$$

$z_n \rightarrow (x_n - \bar{x})^T u_1$

"Avg. of original data pts = Avg. of projected data pts" for the opt. u_1 vector. So only vectors for which $\bar{x} = \bar{\tilde{x}}$ need to be considered. So, c is a constant, & minimizing a^2 or maximizing b^2 are the same by Pythagoras thm. ($c^2 = a^2 + b^2$).

[CBM]

Outline for Module M5

- M5. Dimensionality Reduction
  - M5.0 Introduction
  - **M5.1 PCA**
    - Intuition for two formulations
    - **Maximizing variance formulation**
    - Minimizing error formulation
  - M5.2 PCA applications/extensions/variants (very brief)

What is the variance of projected data when M=1?

$$Var(\{\boldsymbol{x}_n^T \boldsymbol{u}_1\}_{\boldsymbol{n}=1,\dots,N}) = \underbrace{\frac{1}{N}\sum_{n=1}^{N}\left\{\mathbf{x}_n^T\mathbf{u}_1 - \overline{\mathbf{x}}^{\mathrm{T}}\mathbf{u}_1\right\}^2}_{} = \mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1$$

$$= \frac{1}{N}\sum_n \left((x_n^T - \bar{x}^T)u_1\right)^2$$

$$= \frac{1}{N}\sum_n u_1^T (x_n - \bar{x})(x_n - \bar{x})^T u_1$$

$$= u_1^T\left[\frac{1}{N}\sum_n (x_n - \bar{x})(x_n - \bar{x})^T\right]u_1$$

What u_1 maximizes the above variance?

$$\overline{\mathbf{x}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n$$

$$(AB)^T = B^T A^T$$

$$(u^T v)^2 = \|u^T v\|^2 = (u^T v)(u^T v)^T$$

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \overline{\mathbf{x}})(\mathbf{x}_n - \overline{\mathbf{x}})^{\mathrm{T}}.$$

[CBM]

How do you maximize variance, subject to unit length constraint? Lagrangian multiplier

Maximize $\mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1 + \lambda_1\left(1 - \mathbf{u}_1^{\mathrm{T}}\mathbf{u}_1\right).$

yields: $\mathbf{S}\mathbf{u}_1 = \lambda_1\mathbf{u}_1$

with maximum value being: $\mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1 = \lambda_1$

$u_1^T \lambda_1 u_1$

$= \lambda_1 u_1^T u_1 = \lambda_1$

Ex: Prove S is real, symm. psd.

$\nabla_x(x^TAx) = (A+A^T)x$

$\nabla_x(x^Tx) = 2x$

[CBM]

M > 1, Problem statement for PCA (max. var.)

Given: $x_1, \ldots, x_N \in \mathbb{R}^D$

Find: $u_1, \ldots, u_M \in \mathbb{R}^D$ orthonorm.

$$u_i^T u_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

s.t. $J = \sum_{i=1}^{M} \mathrm{Var}(\{x_n^T u_i\}) = \sum_{i=1}^{M} u_i^T S u_i$ is maximized.

Global max of J achieved when $\{u_i\}$ are top M eigen vecs. of S with max value being $\sum_{i=1}^{M} \lambda_i$.

M > 1 (alternate) proof sketch

One proof very similar to that of spectral clustering M score vectors' proof.

Alternate proof sketch: It can also be shown that

- Q: What is the direction u_2 that maximizes variance of projected data along u_2, under the constraint that u_2 is orthogonal to the top eigenvector u_1?
- A: Eigenvector corresp. to the 2nd largest eigenvalue.

- Extending the argument by induction shows that "top" M eigen vectors of S are the "top" M PCs, and projection onto them maximizes the sum of variances of projected data along these directions.

[CBM, Exercise 12.1]

M > 1 proof (similar to spectral clustering M score vectors' proof)

Maximize $$\tilde{J} = \mathrm{Tr}\left\{\widehat{\mathbf{U}}^{\mathrm{T}}\mathbf{S}\widehat{\mathbf{U}}\right\} + \mathrm{Tr}\left\{\mathbf{H}(\mathbf{I} - \widehat{\mathbf{U}}^{\mathrm{T}}\widehat{\mathbf{U}})\right\}$$

$$\underset{N\times M}{\hat{U}} = \begin{bmatrix} | & & | \\ u_1 & \cdots & u_M \\ | & & | \end{bmatrix}$$

$$\tilde{J} = \sum_{i=1}^{M} u_i^T S u_i + \sum_{i,j=1}^{M} H_{ij}(\delta_{ij} - u_i^T u_j)$$

yields: $$\boldsymbol{S\hat{U}} = \boldsymbol{\hat{U}H}$$

One soln: $\mathbf{S}\mathbf{u}_i = \lambda_i \mathbf{u}_i$ for $i = 1, \ldots, M$,
with maximum value being: $\tilde{J} = \sum_{i=1}^{M} \lambda_i$.
(enough to consider this solution alone, as the symmetric $\boldsymbol{H}$ can be assumed to be diagonal ($\boldsymbol{H}_{ii} \coloneqq \lambda_i$) wlog)

[CBM, Exercise 12.2]

Outline for Module M5

- M5. Dimensionality Reduction
  - M5.0 Introduction
  - **M5.1 PCA**
    - Intuition for two formulations
    - Maximizing variance formulation
    - **Minimizing error formulation**
  - M5.2 PCA applications/extensions/variants (very brief)

Representing a datapoint (and its approxmn.) via a set of D ($M < D$) orthonormal vectors in $\mathbb{R}^D$

$$\mathbf{u}_i^{\mathrm{T}}\mathbf{u}_j = \delta_{ij}.$$

(orthonormality requirement) $u_1, \ldots, u_M, \ldots, u_D$

$$\mathbf{x}_n = \sum_{i=1}^{D} \alpha_{ni}\mathbf{u}_i \qquad \mathbf{x}_n = \sum_{i=1}^{D} \left(\mathbf{x}_n^{\mathrm{T}}\mathbf{u}_i\right)\mathbf{u}_i.$$

(perfect/loss-free reconstruction using D dimensions, i.e., along D u_is)

$$\widetilde{\mathbf{x}}_n = \sum_{i=1}^{M} z_{ni}\mathbf{u}_i + \sum_{i=M+1}^{D} b_i\mathbf{u}_i$$

(approximation using M<D dimensions; Note: b_i does **not** depend on n)

$\{z_{ni}\}$ $N \times M$

$x_1, \ldots, x_N$ $(N \times D)$ $\longleftrightarrow$ $\tilde{x}_1, \ldots, \tilde{x}_N$ $(N \times D)$ $\{u_i\}$ D^2 scalars $\{b_i\}$ $D-M$ scalars

[CBM]

Problem statement for PCA (min. error)

I/P: $\{x_n\}_{n=1}^{N} \in \mathbb{R}^D$

O/P: Find $\{z_{ni}\}$, $\{b_i\}$, $\{u_i\}$ orthonorm.

s.t. $\frac{1}{N}\sum_n \|x_n - \tilde{x}_n\|^2$ is minimized,

where $$\widetilde{\mathbf{x}}_n = \sum_{i=1}^{M} z_{ni}\mathbf{u}_i + \sum_{i=M+1}^{D} b_i\mathbf{u}_i$$

Objective #1: choosing z_{ni} and b_i to minimize error (given an orthonormal set $\{u_i\}$)

$$J = \frac{1}{N} \sum_{n=1}^{N} \|\mathbf{x}_n - \widetilde{\mathbf{x}}_n\|^2.$$

$$= \frac{1}{N} \sum_{n} \left[\sum_{i=1}^{M} (\alpha_{ni} - z_{ni})^2 + \sum_{i=M+1}^{D} (\alpha_{ni} - b_i)^2 \right]$$

minimized when

$$z_{ni} = \mathbf{x}_n^{\mathrm{T}} \mathbf{u}_i \qquad b_i = \overline{\mathbf{x}}^{\mathrm{T}} \mathbf{u}_i$$

Minimum value:

$$J = \frac{1}{N} \sum_{n=1}^{N} \sum_{i=M+1}^{D} \left(\mathbf{x}_n^{\mathrm{T}} \mathbf{u}_i - \overline{\mathbf{x}}^{\mathrm{T}} \mathbf{u}_i \right)^2 = \sum_{i=M+1}^{D} \mathbf{u}_i^{\mathrm{T}} \mathbf{S} \mathbf{u}_i$$

[CBM, HR]

Objective #2: choosing $\{u_i\}$ to minimize error (when M=1 & D=2 again)

Minimize $\widetilde{J} = \mathbf{u}_2^{\mathrm{T}}\mathbf{S}\mathbf{u}_2 + \lambda_2\left(1 - \mathbf{u}_2^{\mathrm{T}}\mathbf{u}_2\right)$

yields: $\mathbf{S}\mathbf{u}_2 = \lambda_2\mathbf{u}_2$

with minimum value being: $\mathbf{u}_2^{\mathrm{T}}\mathbf{S}\mathbf{u}_2 = \lambda_2$

[CBM]

General D,M (i.e., when (D-M) > 1, proof similar to spectral clustering multiple score vectors’ proof)

Minimize $$\tilde{J} = \mathrm{Tr}\left\{\widehat{\mathbf{U}}^{\mathrm{T}}\mathbf{S}\widehat{\mathbf{U}}\right\} + \mathrm{Tr}\left\{\mathbf{H}(\mathbf{I} - \widehat{\mathbf{U}}^{\mathrm{T}}\widehat{\mathbf{U}})\right\}$$

$\hat{U} \in \mathbb{R}^{D \times D-M}$

$$\tilde{J} = \sum_{i=M+1}^{D} u_i^T S u_i + \sum_{i,j=M+1}^{D} H_{ij}(\delta_{ij} - u_i^T u_j)$$

$H_{ij} \to H_{(i-M)(j-M)}$

$$\hat{U} = D\left\{\overbrace{\begin{bmatrix} u_{M+1} & \cdots & u_D \end{bmatrix}}^{D-M}\right.$$

yields: $$\boldsymbol{S\hat{U}} = \boldsymbol{\hat{U}H}$$

One soln: $\mathbf{S}\mathbf{u}_i = \lambda_i \mathbf{u}_i$ for $i = M+1, \ldots, D$,

with minimum value being: $\tilde{J} = \sum_{i=M+1}^{D} \lambda_i$.

(enough to consider this solution alone, as the symmetric $\boldsymbol{H}$ can be assumed to be diagonal ($\boldsymbol{H}_{ii} \coloneqq \lambda_i$) wlog)

[CBM, Exercise 12.2]

Outline for Module M5

- M5. Dimensionality Reduction
  - M5.0 Introduction
  - M5.1 PCA
    - Intuition for two formulations
    - Maximizing variance formulation
    - Minimizing error formulation
  - **M5.2 PCA applications/extensions/variants (very brief)**

$\{x_n\}$ $\quad X \in R^{N \times D}$ (mean-centered) $\rightarrow S = \frac{X^T X}{N} \xrightarrow{EVD}$ Top M ev (PCs) $\rightarrow \{\tilde{x}_n\}$ $(N \times D)$

$(N \times M, D^2, D-M)$

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \overline{\mathbf{x}})(\mathbf{x}_n - \overline{\mathbf{x}})^{\mathrm{T}}.$$

Data compression: but first, what is all the talk about mean-subtracting?

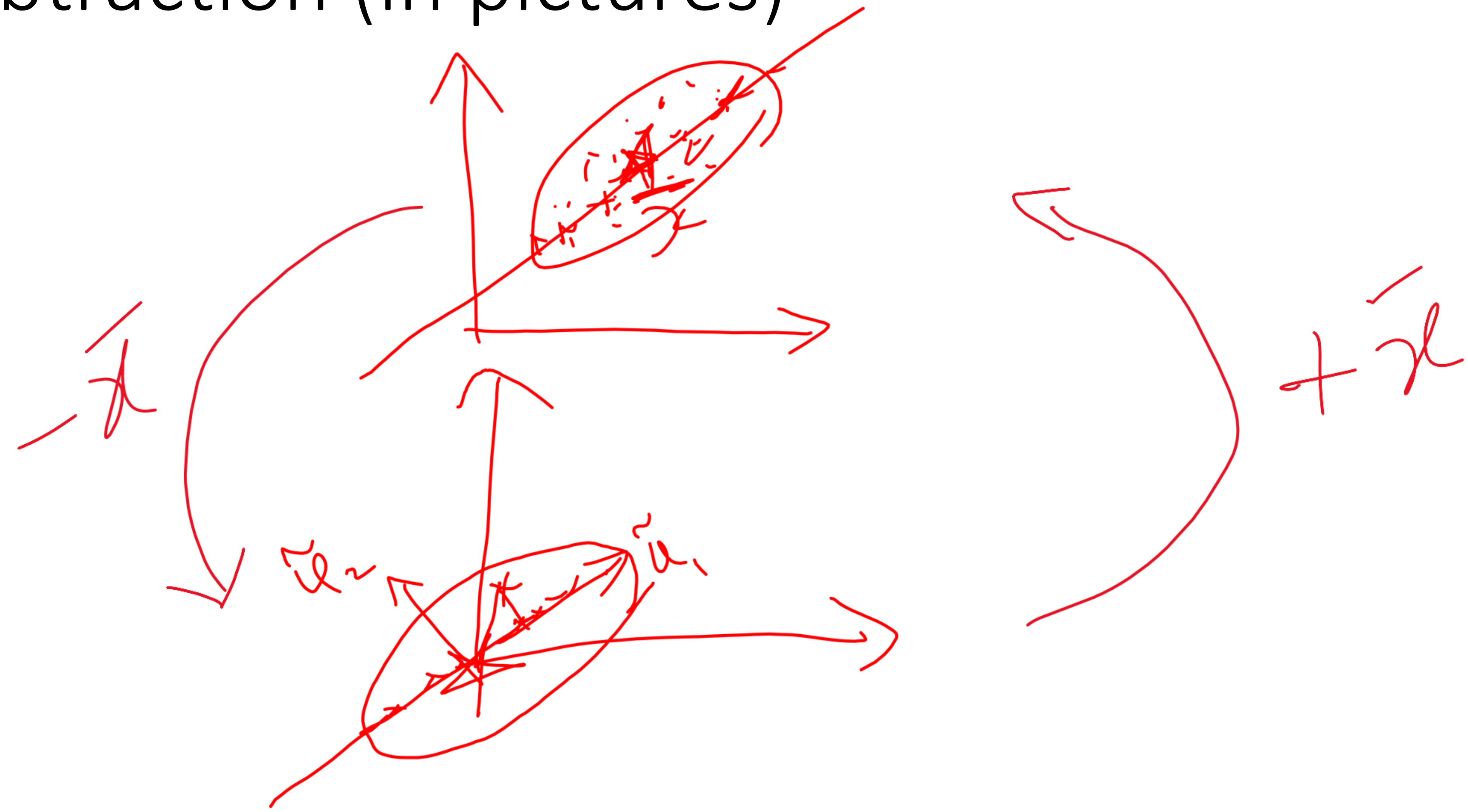
$$\widetilde{\mathbf{x}}_n = \sum_{i=1}^{M}(\mathbf{x}_n^{\mathrm{T}}\mathbf{u}_i)\mathbf{u}_i + \sum_{i=M+1}^{D}(\overline{\mathbf{x}}^{\mathrm{T}}\mathbf{u}_i)\mathbf{u}_i + \bar{x} - \bar{x}$$

$$= \overline{\mathbf{x}} + \sum_{i=1}^{M}\left(\mathbf{x}_n^{\mathrm{T}}\mathbf{u}_i - \overline{\mathbf{x}}^{\mathrm{T}}\mathbf{u}_i\right)\mathbf{u}_i = \bar{x} + \sum_{i=1}^{M}\left((x_n - \bar{x})^T u_i\right)u_i$$

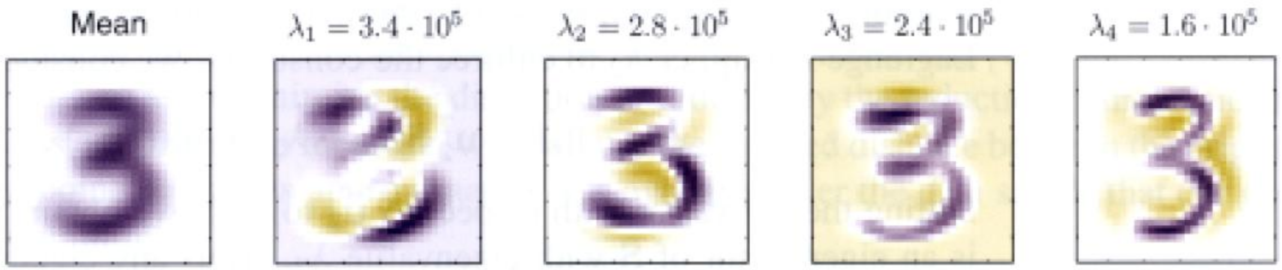
$$\overline{\mathbf{x}} = \sum_{i=1}^{D}\left(\overline{\mathbf{x}}^{\mathrm{T}}\mathbf{u}_i\right)\mathbf{u}_i$$

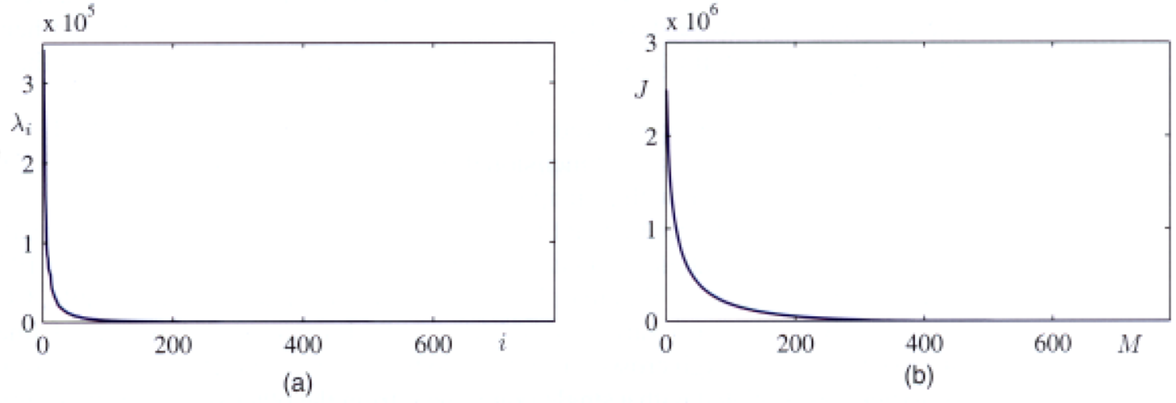
[CBM]

Mean-subtraction (in pictures)



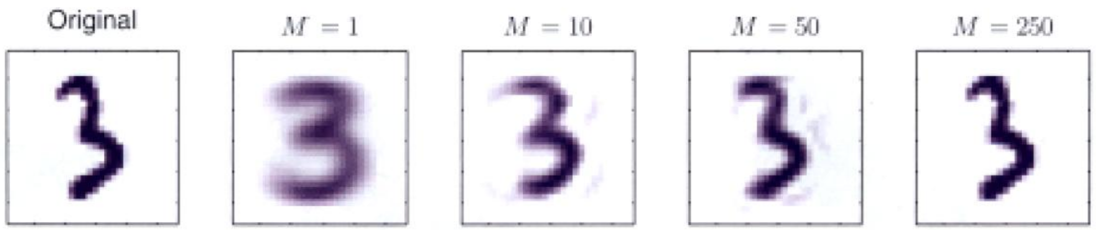
Data compression (on mean-subtracted data):





ND

$x_n \in \mathbb{R}^{900}$



NM

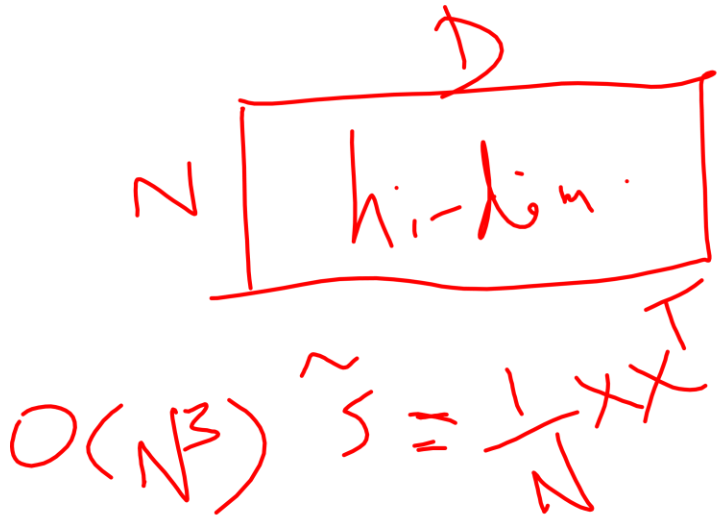
$\overline{x}$ D +

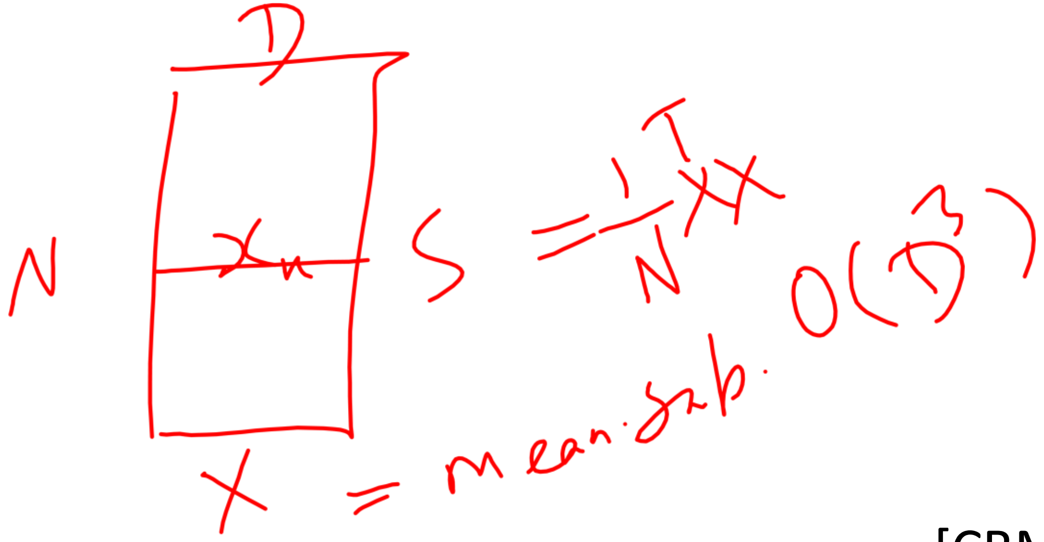
$\{u_i\}$ D^2

[CBM]

Other PCA extensions/variants

- PCA for High-dimensional Data
  - (transpose, compute Eigen vectors in less time, convert to Eigen vectors in original space – has connections to SVD) – see Assignment question!
- From Deterministic PCA to Probabilistic PCA to Bayesian PCA / Mixture of Probabilistic PCA / Factor analysis / etc.
- Kernel PCA





[CBM]

Summary of dimensionality reduction (PCA)

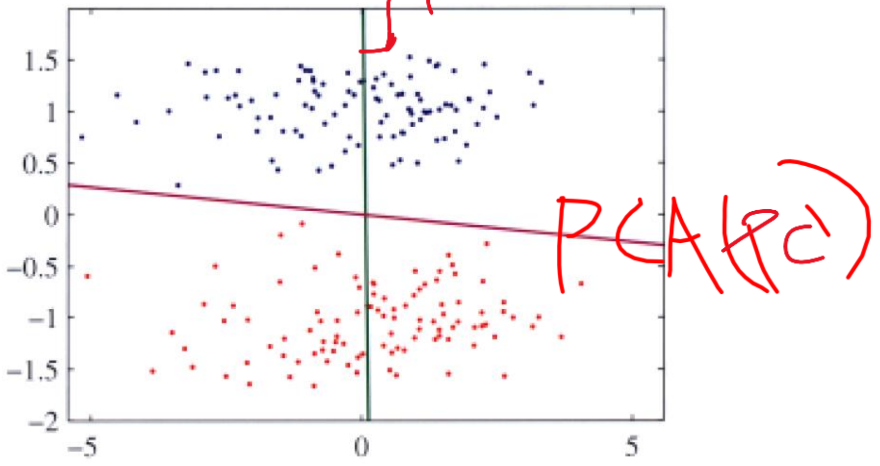
- PCA is a method to transform data points from a high-dimensional to a linear low-dimensional space, by maximizing variance or equiv. minimizing error.
  - Linear because low-dimensional subspace is a vector subspace (provided, the data is mean-centered i.e., the mean passes through zero making the subspace to include zero/origin as well; even otherwise, PCA holds by viewing the subspace as an affine subspace instead of a vector subspace)

- One of the many available unsupervised methods for dimension reduction.

- Supervised dim. redn. methods also exist (not covered here)

and may be better at finding relevant dimensions.

Fishers linear discriminant



[CBM]

Thank you!