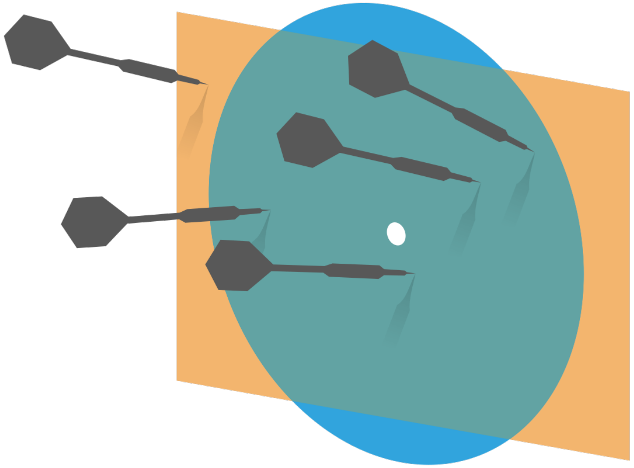
Probability and Statistics for Computer Science



Credit: wikipedia

Principal Component Analysis --- Exploring the data in less dimensions

Hongye Liu, Teaching Assistant Prof, CS361, UIUC, 10.27.2020

Last time

- Review of Bayesian inference
- Visualizing high dimensional data & Summarizing data
- The covariance matrix

Objectives

- Principal Component Analysis
- Examples of PCA

Diagonalization of a symmetric matrix

- If A is an **n×n** symmetric square matrix, the eigenvalues are real.
- If the eigenvalues are also distinct, their eigenvectors are orthogonal
- We can then scale the eigenvectors to unit length, and place them into an orthogonal matrix $U = [\mathbf{u}_1\ \mathbf{u}_2\ \ldots.\ \mathbf{u}_n]$
- We can write the diagonal matrix $\Lambda = U^T A U$ such that the diagonal entries of Λ are $\lambda_1, \lambda_2 \ldots \lambda_n$ in that order.

Diagonalization example

- For

$$A = \begin{bmatrix} 5 & 3 \\ 3 & 5 \end{bmatrix}$$

Covariance for a pair of components in a data set

- For the jth and kth components of a data set {x}

$$cov(\{x\}; j, k) = \frac{\sum_i (x_i^{(j)} - mean(\{x^{(j)}\}))(x_i^{(k)} - mean(\{x^{(k)}\}))^T}{N}$$

Covariance matrix

Data set $\{\mathbf{x}\}$ 7×8

$cov(\{\mathbf{x}\}; 3, 5)$

	1	2	3	4	5	6	7	8
1	*	*	*	*	*	*	*	*
2	*	*	*	*	*	*	*	*
3	*	*	*	*	*	*	*	*
4	*	*	*	*	*	*	*	*
5	*	*	*	*	*	*	*	*
6	*	*	*	*	*	*	*	*
7	*	*	*	*	*	*	*	*

Covmat($\{\mathbf{x}\}$) 7×7

	1	2	3	4	5	6	7
1	*	*	*	*	*	*	*
2	*	*	*	*	*	*	*
3	*	*	*	*	*	*	*
4	*	*	*	*	*	*	*
5	*	*	*	*	*	*	*
6	*	*	*	*	*	*	*
7	*	*	*	*	*	*	*

Properties of Covariance matrix

$$cov(\{x\}; j, j) = var(\{x^{(j)}\})$$

Covmat($\{\mathbf{x}\}$) 7×7

	1	2	3	4	5	6	7
1	*	*	*	*	*	*	*
2	*	*	*	*	*	*	*
3	*	*	*	*	*	*	*
4	*	*	*	*	*	*	*
5	*	*	*	*	*	*	*
6	*	*	*	*	*	*	*
7	*	*	*	*	*	*	*

- The diagonal elements of the covariance matrix are just variances of each jth components
- The off diagonals are covariance between different components

Properties of Covariance matrix

$$cov(\{x\}; j, k) = cov(\{x\}; k, j)$$

Covmat$(\{\mathbf{x}\})$ 7×7

	1	2	3	4	5	6	7
1	*	*	*	*	*	*	*
2	*	*	*	*	*	*	*
3	*	*	*	*	*	*	*
4	*	*	*	*	*	*	*
5	*	*	*	*	*	*	*
6	*	*	*	*	*	*	*
7	*	*	*	*	*	*	*

- The covariance matrix is **symmetric**!
- And it's **positive semi-definite**, that is all $\lambda_i \geq 0$
- Covariance matrix is diagonalizable

Properties of Covariance matrix

- If we define $\mathbf{x_c}$ as the mean centered matrix for dataset {x}

$$Covmat(\{x\}) = \frac{\mathrm{x}_c \times \mathrm{x}_c^T}{N}$$

- The covariance matrix is a d×d matrix

$\text{Covmat}(\{\mathbf{x}\})$ 7×7

	1	2	3	4	5	6	7
1	*	*	*	*	*	*	*
2	*	*	*	*	*	*	*
3	*	*	*	*	*	*	*
4	*	*	*	*	*	*	*
5	*	*	*	*	*	*	*
6	*	*	*	*	*	*	*
7	*	*	*	*	*	*	*

d =7

Example: covariance matrix of a data set

(I)

$$A_0 = \begin{array}{c} \begin{bmatrix} 5 & 4 & 3 & 2 & 1 \\ -1 & 1 & 0 & 1 & -1 \end{bmatrix} \end{array} \begin{array}{l} X^{(1)} \\ X^{(2)} \end{array}$$

What are the dimensions of the covariance matrix of this data?

A) 2 by 2
B) 5 by 5
C) 5 by 2
D) 2 by 5

Example: covariance matrix of a data set

(I)

Mean centering

$$A_0 = \begin{bmatrix} 5 & 4 & 3 & 2 & 1 \\ -1 & 1 & 0 & 1 & -1 \end{bmatrix}$$

$$A_1 = \begin{bmatrix} 2 & 1 & 0 & -1 & -2 \\ -1 & 1 & 0 & 1 & -1 \end{bmatrix}$$

(II) $A_2 = A_1 A_1^T$

Inner product of each pairs:

A_2 [1,1] = 10

A_2 [2,2] = 4

A_2 [1,2] = 0

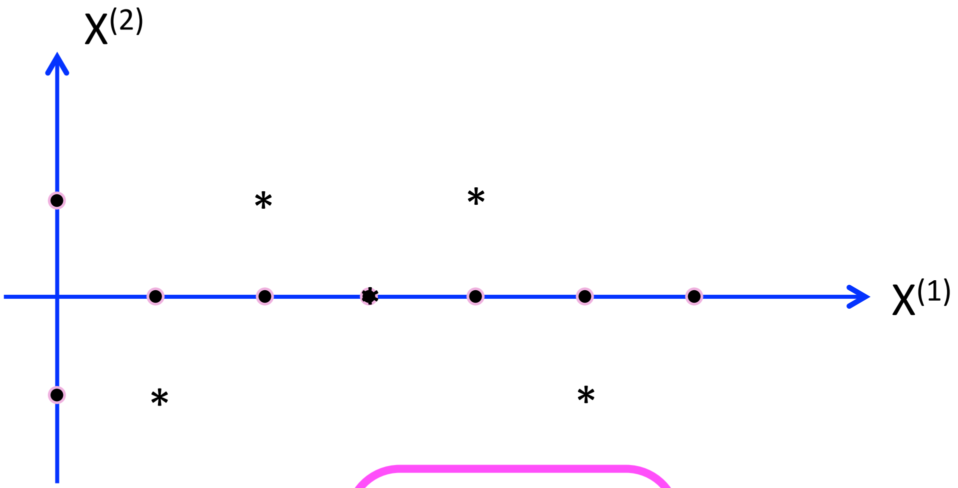
(III)

Divide the matrix with N – the number of data poits

$$\text{Covmat}(\{\mathbf{x}\}) = \frac{1}{N} A_2 = \frac{1}{5} \begin{bmatrix} 10 & 0 \\ 0 & 4 \end{bmatrix} = \boxed{\begin{bmatrix} 2 & 0 \\ 0 & 0.8 \end{bmatrix}}$$

What do the data look like when Covmat({x}) is diagonal?

$$A_0 = \begin{array}{cc} \begin{bmatrix} 5 & 4 & 3 & 2 & 1 \\ -1 & 1 & 0 & 1 & -1 \end{bmatrix} & \begin{array}{c} x^{(1)} \\ x^{(2)} \end{array} \end{array}$$



$$\text{Covmat}(\{\mathbf{x}\}) = \frac{1}{N} A_2 = \frac{1}{5} \begin{bmatrix} 10 & 0 \\ 0 & 4 \end{bmatrix} = \boxed{\begin{bmatrix} 2 & 0 \\ 0 & 0.8 \end{bmatrix}}$$

What is the correlation between the 2 components for the data m?

$$Covmat(\boldsymbol{m}) = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix}$$

Q. Is this true?

Transforming a matrix with orthonormal matrix only rotates the data

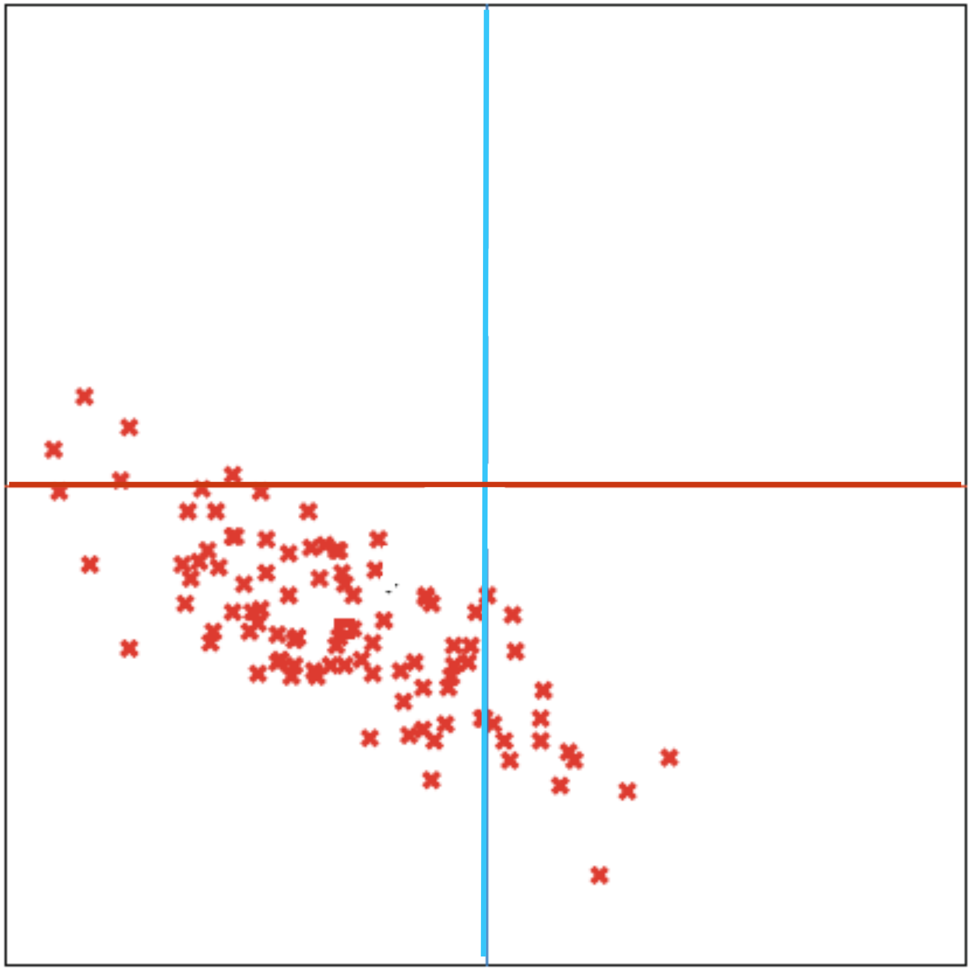
A. Yes

B. No

Dimension Reduction

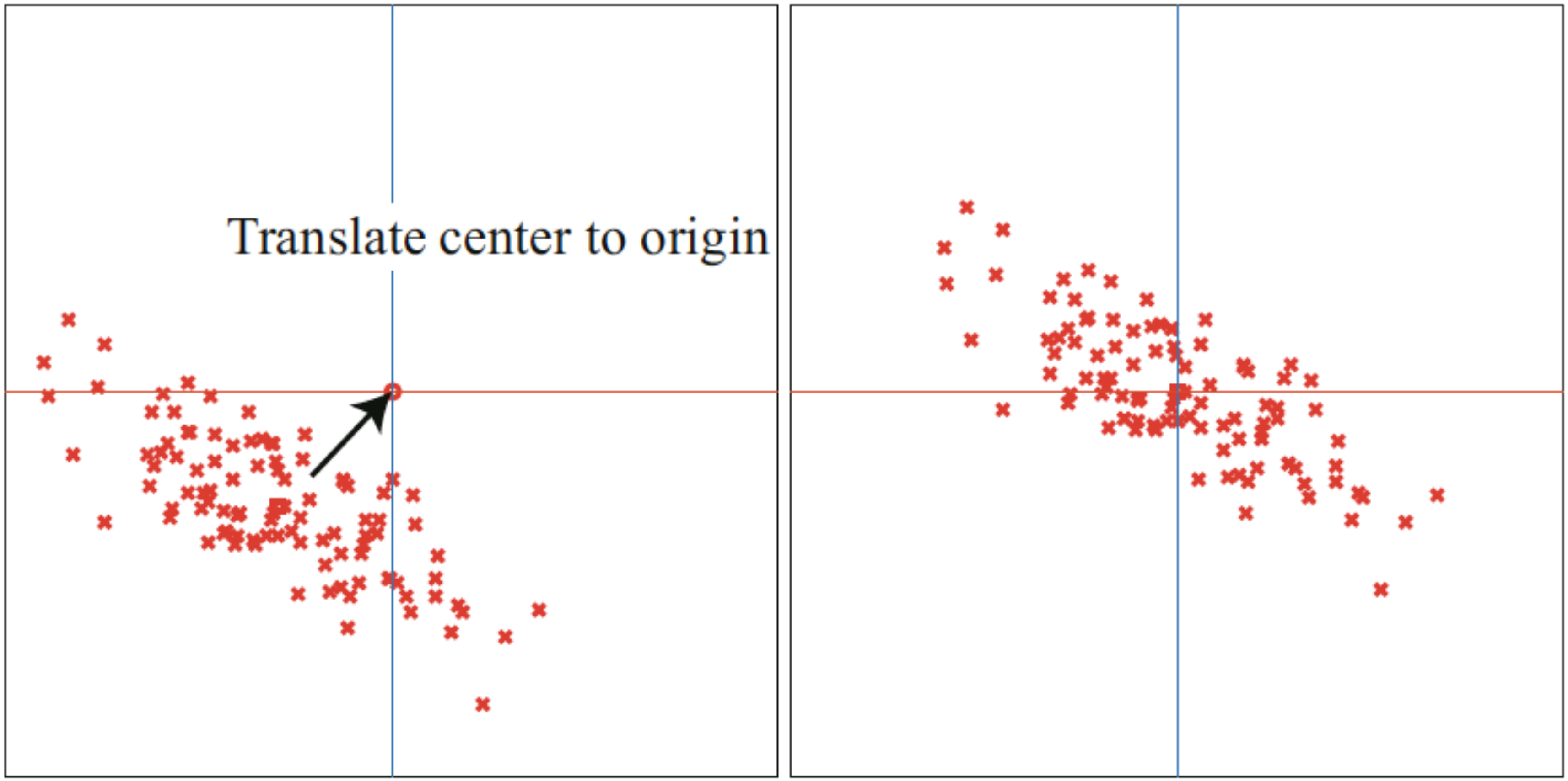
- In stead of showing more dimensions through visualization, it's a good idea to do dimension reduction in order to see the major features of the data set.
- For example, principal component analysis help find the major components of the data set.
- PCA is essentially about finding eigenvectors of the covariance matrix of the data set {x}

Dimension reduction from 2D to 1D



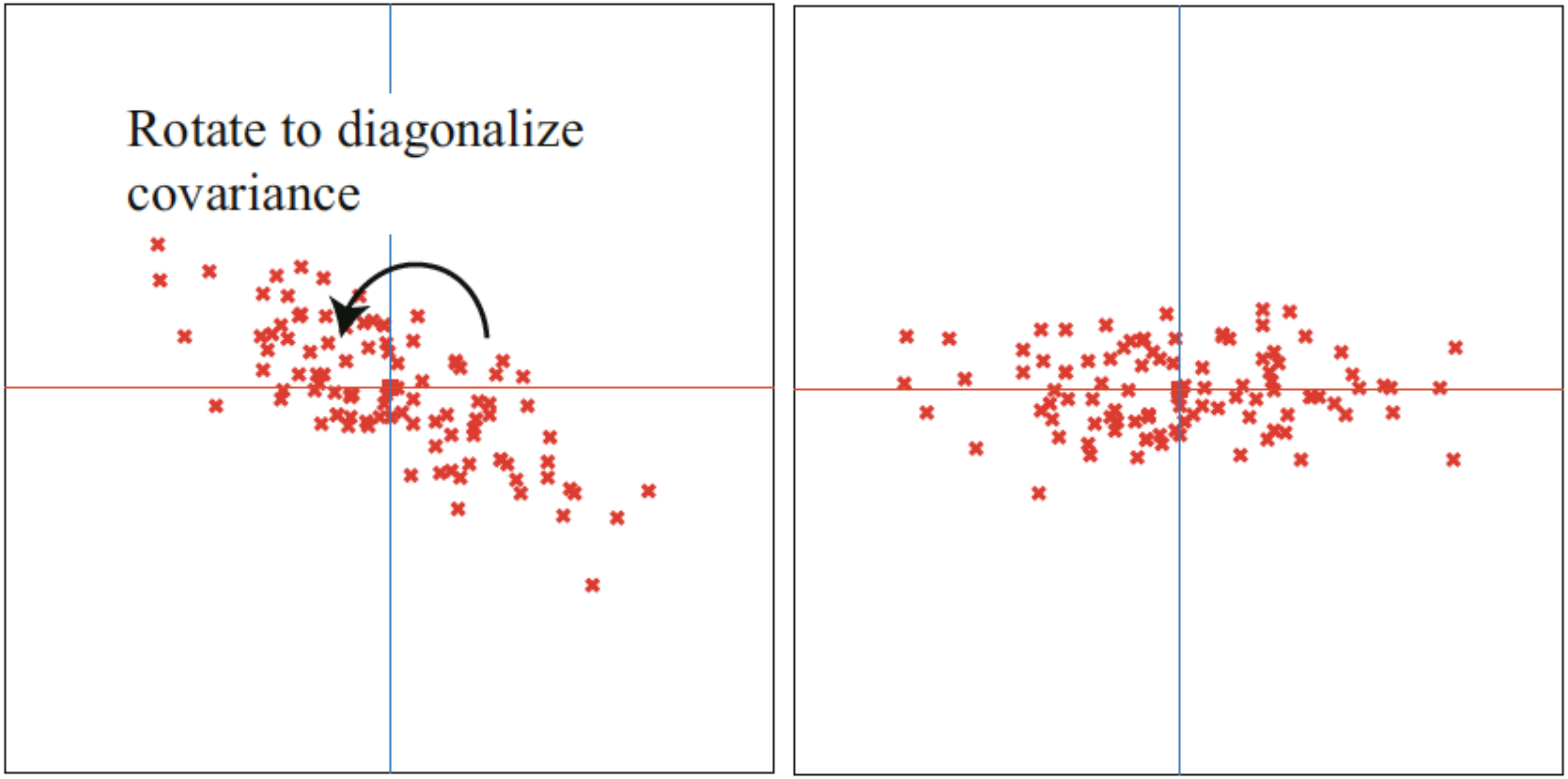
Credit: Prof. Forsyth

Step 1: subtract the mean



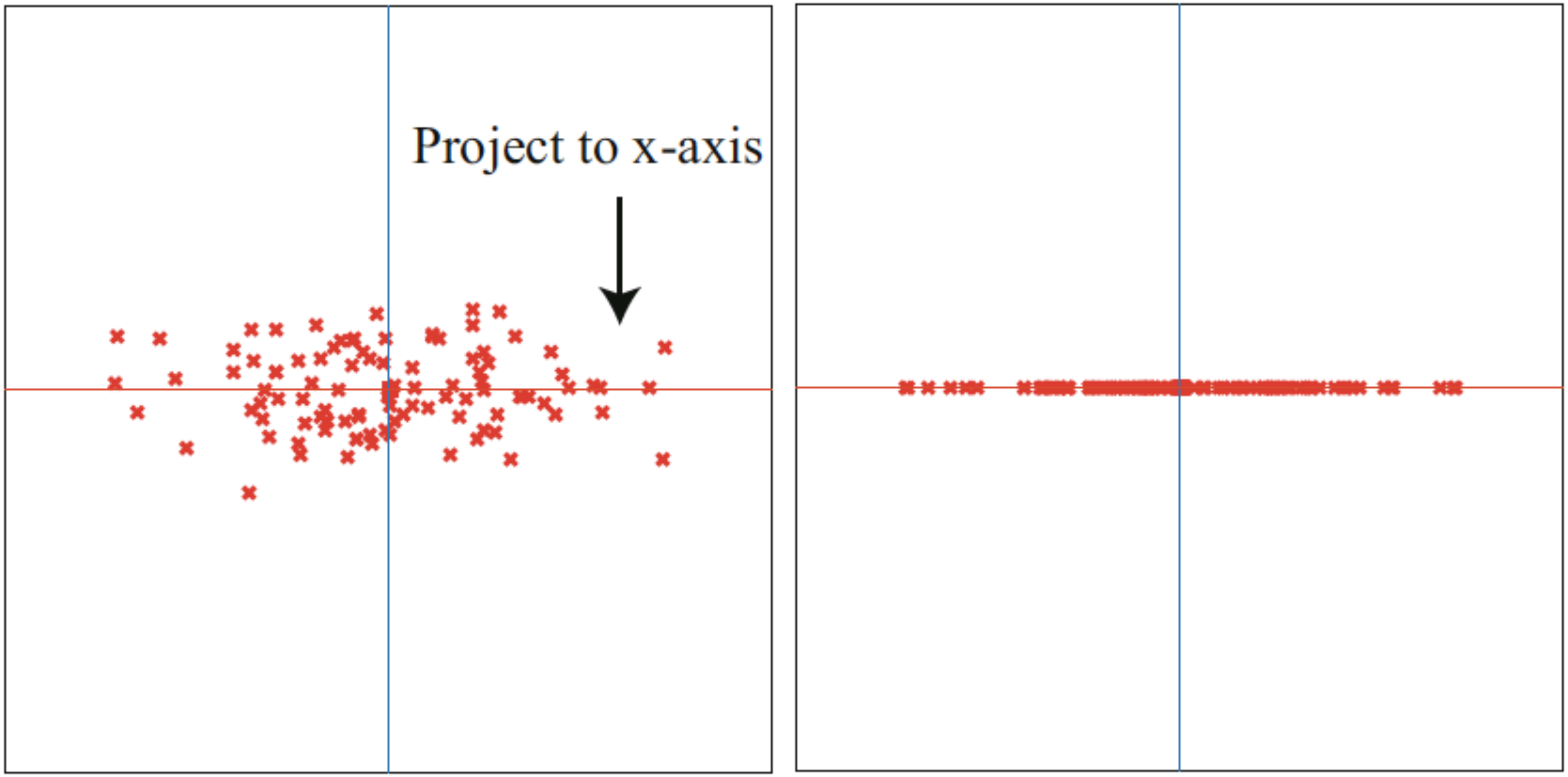
Credit: Prof. Forsyth

Step 2: Rotate to diagonalize the covariance



Credit: Prof. Forsyth

Step 3: Drop component(s)



Credit: Prof. Forsyth

Principal Components

- The columns of U are the normalized eigenvectors of the Covmat({x}) and are called the **principal components** of the data {x}

Principal components analysis

- We reduce the dimensionality of dataset {**x**} represented by matrix $\boldsymbol{D}_{d\times n}$ from d to s (s < d).
- Step 1. define matrix $\boldsymbol{m}_{d\times n}$ such that $\boldsymbol{m} = \boldsymbol{D} - mean(\boldsymbol{D})$
- Step 2. define matrix $\boldsymbol{r}_{d\times n}$ such that $\boldsymbol{r}_i = \boldsymbol{U}^T \boldsymbol{m}_i$

  Where $\boldsymbol{U}^T$ satisfies $\boldsymbol{\Lambda} = \boldsymbol{U}^T\ Covmat(\{\boldsymbol{x}\})\boldsymbol{U}$, $\boldsymbol{\Lambda}$ is the diagonalization of $Covmat(\{\boldsymbol{x}\})$ with the eigenvalues sorted in decreasing order, $\boldsymbol{U}$ is the orthonormal eigenvectors' matrix
- Step 3. Define matrix $\boldsymbol{p}_{d\times n}$ such that $\boldsymbol{p}$ is $\boldsymbol{r}$ with the last d-s components of $\boldsymbol{r}$ made zero.

What happened to the mean?

- Step 1.

$$mean(\boldsymbol{m}) = mean(\boldsymbol{D} - mean(\boldsymbol{D})) = 0$$

- Step 2.

$$mean(\boldsymbol{r}) = \boldsymbol{U}^T mean(\boldsymbol{m}) = \boldsymbol{U}^T 0 = 0$$

- Step 3.

$$mean(\boldsymbol{p_i}) = mean(\boldsymbol{r}_i) = 0 \;\; while\ i \in 1:s$$

$$mean(\boldsymbol{p}_i) = 0 \quad while\ i \in s+1:d$$

What happened to the covariances?

- Step 1.

$$Covmat(\boldsymbol{m}) = Covmat(\boldsymbol{D}) = Covmat(\{\boldsymbol{x}\})$$

- Step 2.

$$Covmat(\boldsymbol{r}) = \boldsymbol{U}^T Covmat(\boldsymbol{m})\boldsymbol{U} = \boldsymbol{\Lambda}$$

- Step 3. $Covmat(\boldsymbol{p})$ is $\boldsymbol{\Lambda}$ with the last/smallest d-s diagonal terms turned to 0.

Sample covariance matrix

- In many statistical programs, the sample covariance matrix is defined to be

$$Covmat(\boldsymbol{m}) = \frac{\boldsymbol{m}\ \boldsymbol{m}^T}{N-1}$$

- Similar to what happens to the unbiased standard deviation

PCA an example

- Step 1.

$$D = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix} \Rightarrow mean(\boldsymbol{D}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\boldsymbol{m} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix}$$

- Step 2.

- Step 3.

PCA an example

- Step 1.

$$D = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix} \Rightarrow mean(\boldsymbol{D}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\boldsymbol{m} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix}$$

- Step 2.

$$Covmat(\boldsymbol{m}) = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix} \Rightarrow \lambda_1 \simeq 57; \quad \lambda_2 \simeq 3$$

$$\Rightarrow \boldsymbol{U} = \begin{bmatrix} 0.5606288 & -0.8280672 \\ 0.8280672 & 0.5606288 \end{bmatrix} \qquad \boldsymbol{U^T} = \begin{bmatrix} 0.5606288 & 0.8280672 \\ -0.8280672 & 0.5606288 \end{bmatrix}$$

- Step 3.

PCA an example

- Step 1.

$$D = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix} \Rightarrow mean(\boldsymbol{D}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\boldsymbol{m} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix}$$

- Step 2.

$$Covmat(\boldsymbol{m}) = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix} \Rightarrow \lambda_1 \simeq 57; \quad \lambda_2 \simeq 3$$

$$\Rightarrow \boldsymbol{U} = \begin{bmatrix} 0.5606288 & -0.8280672 \\ 0.8280672 & 0.5606288 \end{bmatrix} \qquad \boldsymbol{U^T} = \begin{bmatrix} 0.5606288 & 0.8280672 \\ -0.8280672 & 0.5606288 \end{bmatrix}$$

$$\Rightarrow \boldsymbol{r} = \boldsymbol{U}^T \boldsymbol{m} = \begin{bmatrix} 7.478 & -7.211 & 10.549 & -0.267 & -3.071 & -7.478 \\ 1.440 & -0.052 & -1.311 & -1.389 & 2.752 & -1.440 \end{bmatrix}$$

- Step 3.

PCA an example

- Step 1.

$$D = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix} \Rightarrow mean(\boldsymbol{D}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\boldsymbol{m} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix}$$

- Step 2.

$$Covmat(\boldsymbol{m}) = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix} \Rightarrow \lambda_1 \simeq 57; \quad \lambda_2 \simeq 3$$

$$\Rightarrow \boldsymbol{U} = \begin{bmatrix} 0.5606288 & -0.8280672 \\ 0.8280672 & 0.5606288 \end{bmatrix} \qquad \boldsymbol{U^T} = \begin{bmatrix} 0.5606288 & 0.8280672 \\ -0.8280672 & 0.5606288 \end{bmatrix}$$

$$\Rightarrow \boldsymbol{r} = \boldsymbol{U}^T\boldsymbol{m} = \begin{bmatrix} 7.478 & -7.211 & 10.549 & -0.267 & -3.071 & -7.478 \\ 1.440 & -0.052 & -1.311 & -1.389 & 2.752 & -1.440 \end{bmatrix}$$

- Step 3. $\Rightarrow \boldsymbol{p} = \begin{bmatrix} 7.478 & -7.211 & 10.549 & -0.267 & -3.071 & -7.478 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

What is this matrix for the previous example?

$$\boldsymbol{U}^T Covmat(\boldsymbol{m})\boldsymbol{U} = ?$$

What is this matrix for the previous example?

$$\boldsymbol{U}^T Covmat(\boldsymbol{m})\boldsymbol{U} = ?$$

$$\boxed{\begin{bmatrix} 57 & 0 \\ 0 & 3 \end{bmatrix}}$$

The Mean square error of the projection

- The mean square error is the sum of the smallest d-s eigenvalues in $\mathbf{\Lambda}$

$$\frac{1}{N-1}\sum_i \|r_i - p_i\|^2 = \frac{1}{N-1}\sum_i \sum_{j=s+1}^{d} (r_i^{(j)})^2$$

The Mean square error of the projection

- The mean square error is the sum of the smallest d-s eigenvalues in $\mathbf{\Lambda}$

$$\frac{1}{N-1}\sum_i \|r_i - p_i\|^2 = \frac{1}{N-1}\sum_i \sum_{j=s+1}^{d} (r_i^{(j)})^2 \quad = \sum_{j=s+1}^{d} \sum_i \frac{1}{N-1}(r_i^{(j)})^2$$

The Mean square error of the projection

- The mean square error is the sum of the smallest d-s eigenvalues in $\mathbf{\Lambda}$

$$\frac{1}{N-1}\sum_i \|r_i - p_i\|^2 = \frac{1}{N-1}\sum_i \sum_{j=s+1}^{d} (r_i^{(j)})^2 \quad = \sum_{j=s+1}^{d} \sum_i \frac{1}{N-1}(r_i^{(j)})^2$$

$$= \sum_{j=s+1}^{d} var(r_i^{(j)})$$

The Mean square error of the projection

- The mean square error is the sum of the smallest d-s eigenvalues in $\mathbf{\Lambda}$

$$\frac{1}{N-1}\sum_i \|r_i - p_i\|^2 = \frac{1}{N-1}\sum_i \sum_{j=s+1}^{d} (r_i^{(j)})^2 \quad = \sum_{j=s+1}^{d} \sum_i \frac{1}{N-1}(r_i^{(j)})^2$$

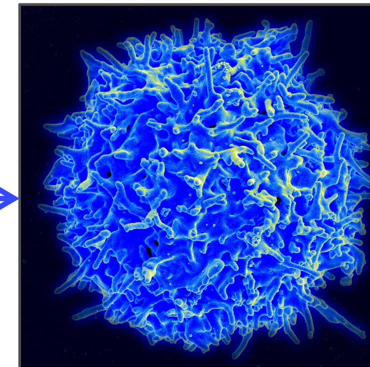
$$= \sum_{j=s+1}^{d} var(r_i^{(j)})$$

$$= \sum_{j=s+1}^{d} \lambda_j$$

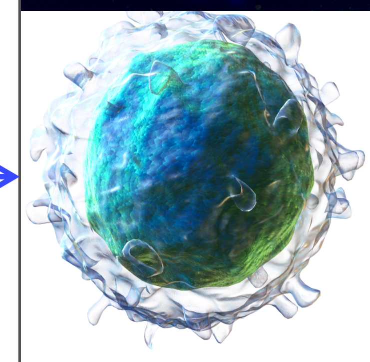
Examples: Immune Cell Data

- There are 38816 white blood immune cells from a mouse sample
- Each immune cell has 40+ features/ components
- Four features are used as illustration.
- There are at least 3 cell types involved

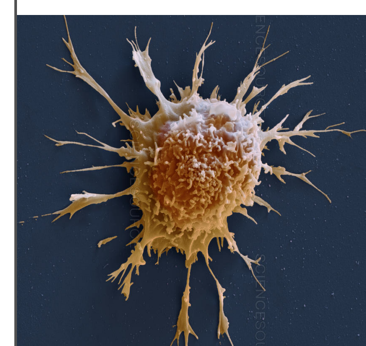
T cells



B cells

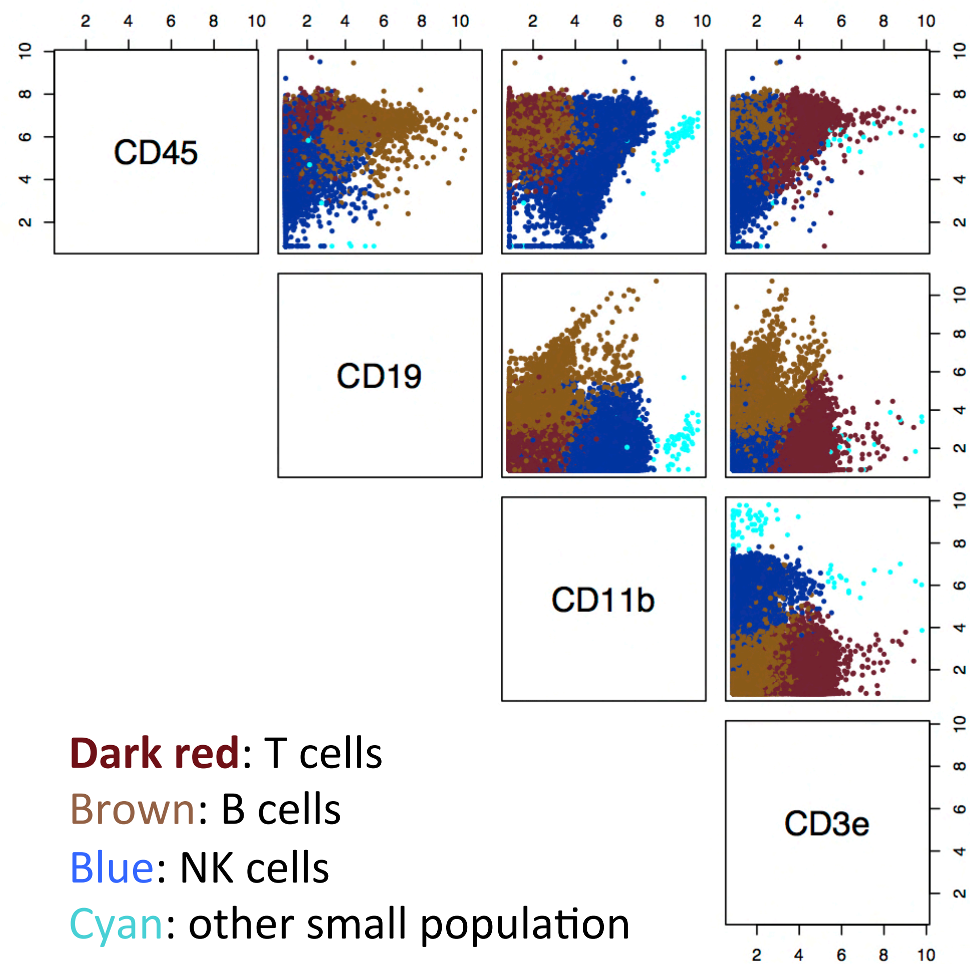


Natural killer cells



Scatter matrix of Immune Cells

- There are 38816 white blood immune cells from a mouse sample
- Each immune cell has 40+ features/ components
- Four features are used for the illustration.
- There are at least 3 cell types involved

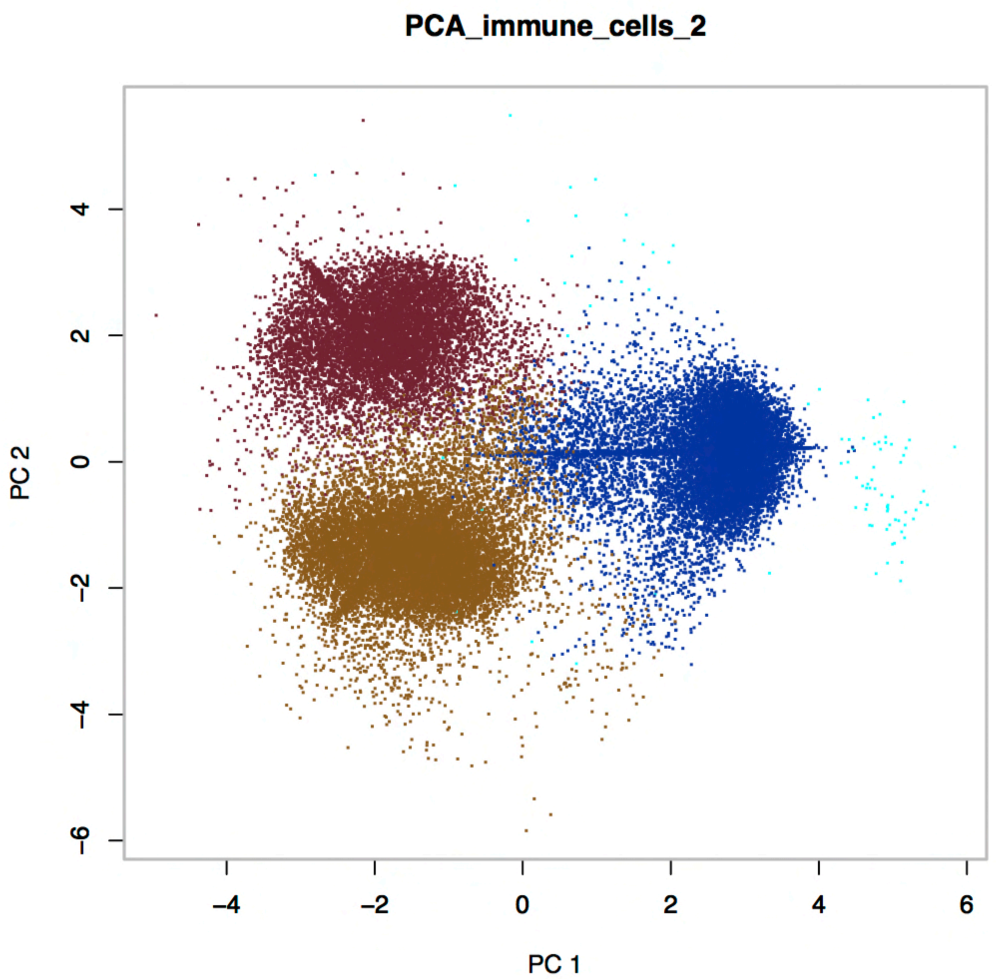


Dark red: T cells
Brown: B cells
Blue: NK cells
Cyan: other small population

PCA of Immune Cells

```
> res1
$values   Eigenvalues
[1] 4.7642829 2.1486896 1.3730662
0.4968255

          Eigenvectors
$vectors
          [,1]       [,2]       [,3]       [,4]
[1,]  0.2476698  0.00801294 -0.6822740
0.6878210
[2,]  0.3389872 -0.72010997 -0.3691532
-0.4798492
[3,] -0.8298232  0.01550840 -0.5156117
-0.2128324
[4,]  0.3676152  0.69364033 -0.3638306
-0.5013477
```



What is the percentage of variance that PC1 covers?

Given the eigenvalues: 4.7642829 2.1486896 1.3730662 0.4968255, what is the percentage that PC1 covers?

A. 54%
B. 16%
C. 25%

Reconstructing the data

- Given the projected data $\boldsymbol{p}_{d\times n}$ and mean({x}), we can approximately reconstruct the original data

$$\widehat{\boldsymbol{D}} = \boldsymbol{U}\boldsymbol{p} + mean(\{\boldsymbol{x}\})$$

- Each reconstructed data item $\widehat{\boldsymbol{D}_i}$ is a linear combination of the columns of $\boldsymbol{U}$ weighted by $\boldsymbol{p}_i$

- The columns of $\boldsymbol{U}$ are the normalized eigenvectors of the Covmat({x}) and are called the **principal components** of the data {x}

End-to-end mean square error

- Each $\boldsymbol{x}_i$ becomes $\boldsymbol{r}_i$ by translation and rotation
- Each $\boldsymbol{p}_i$ becomes $\widehat{\boldsymbol{x}}_i$ by the opposite rotation and translation
- Therefore the end to end mean square error is:

$$\frac{1}{N-1}\sum_i \|\widehat{\boldsymbol{x}}_i - \boldsymbol{x}_i\|^2 = \frac{1}{N-1}\sum_i \|\boldsymbol{r}_i - \boldsymbol{p}_i\|^2 = \sum_{j=s+1}^{d} \lambda_j$$

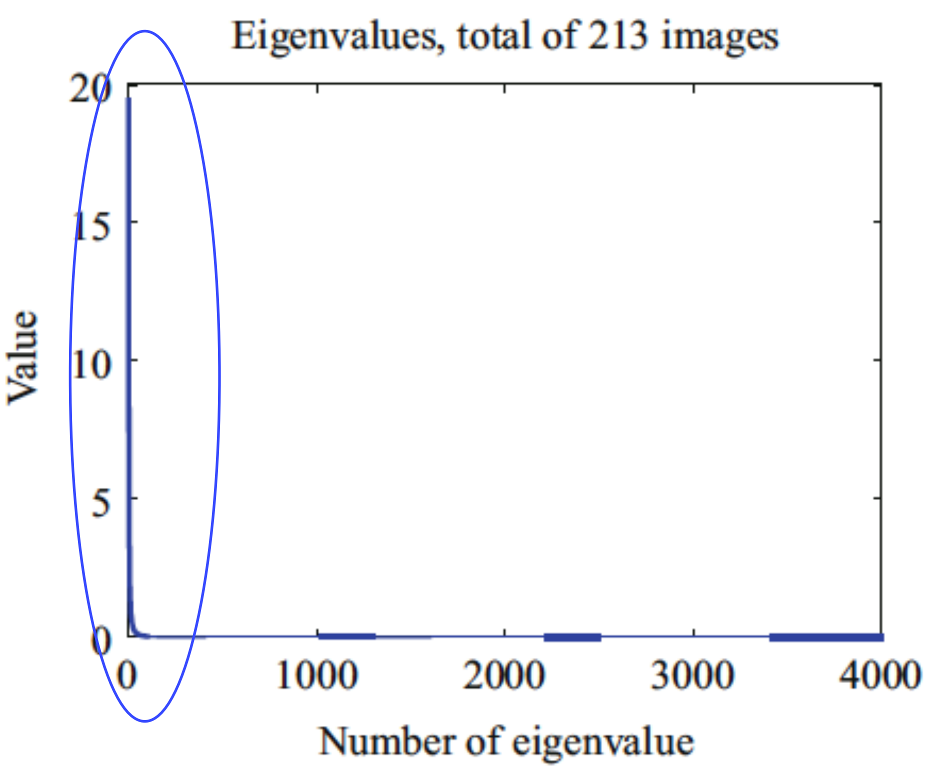
- $\lambda_{s+1}, \ldots, \lambda_d$ are the smallest d-s eigenvalues of the Covmat({x})

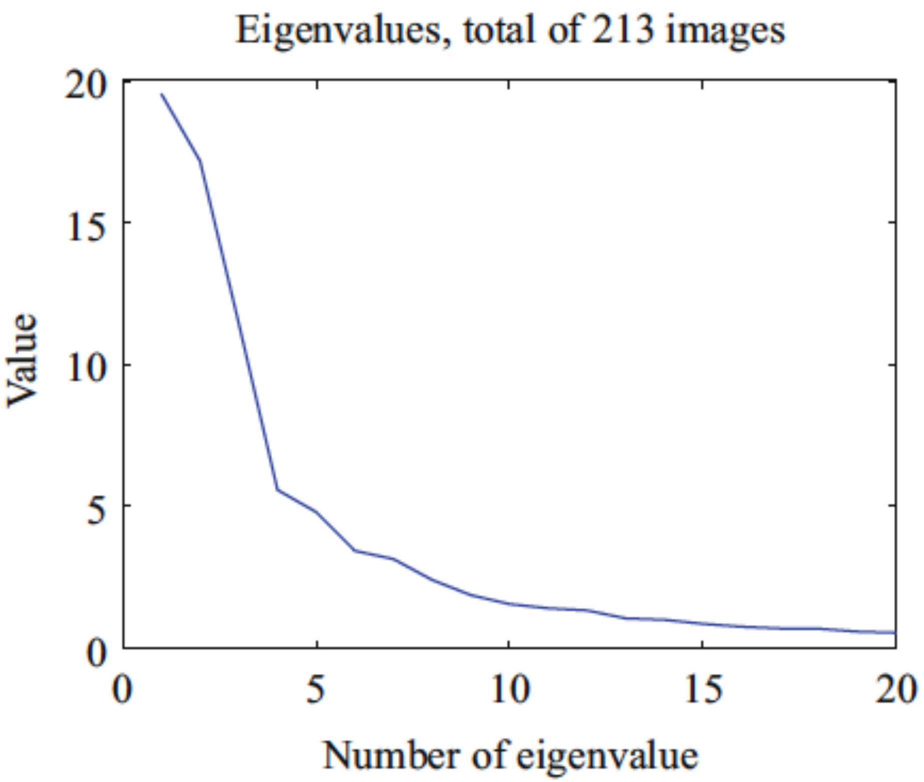
PCA: Human face data

- The dataset consists of 213 images
- Each image is grayscale and has 64 by 64 resolution
- We can treat each image as a vector with dimension $d = 4096$

Credit: Prof. Forsyth

How quickly the eigenvalues decrease?





Credit: Prof. Forsyth

What do the principal components of the images look like?

Mean image

The first 16 principal components arranged into images

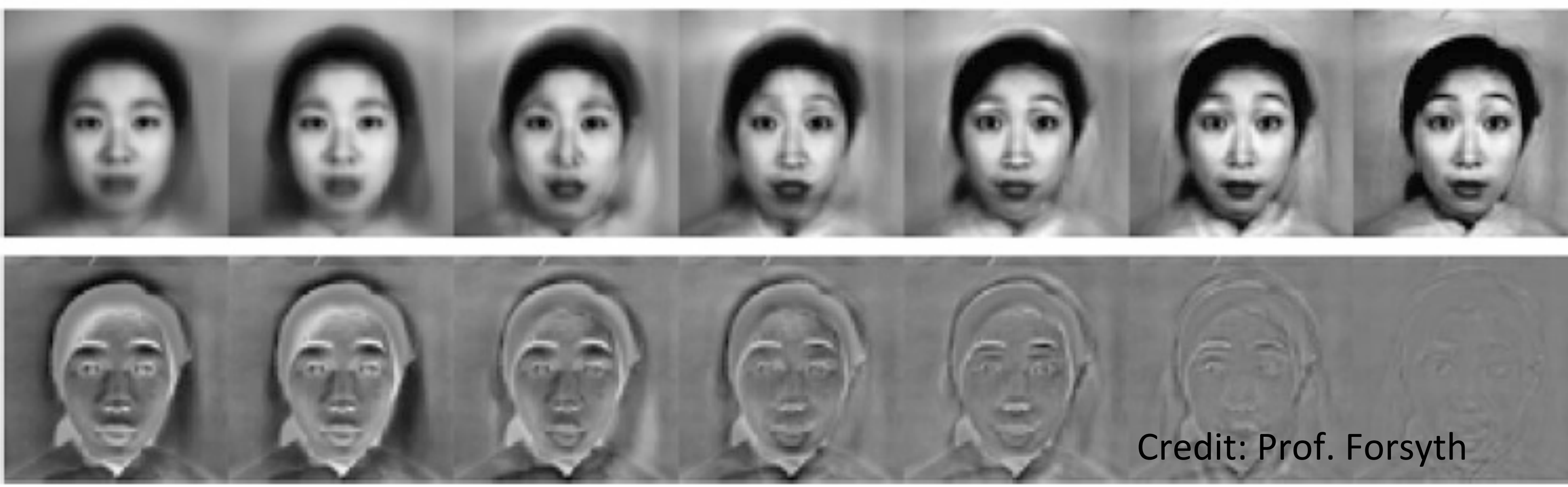
Credit: Prof. Forsyth

Reconstruction of the image

The original

1st row show the reconstructions using some number of principal components
2nd row show the corresponding errors

Mean 1 5 10 20 50 100



Credit: Prof. Forsyth

Q. Which are true?

A . PCA allows us to project data to the direction along which the data has the biggest variance

B. PCA allows us to compress data

C. PCA uses linear transformation to show patterns of data

D. PCA allows us to visualize data in lower dimensions

E. All of the above

Assignments

- Read Chapter 10 of the textbook
- Next time: Intro to classification

Additional References

- Robert V. Hogg, Elliot A. Tanis and Dale L. Zimmerman. “Probability and Statistical Inference”
- Morris H. Degroot and Mark J. Schervish "Probability and Statistics”

See you next time

See You!