



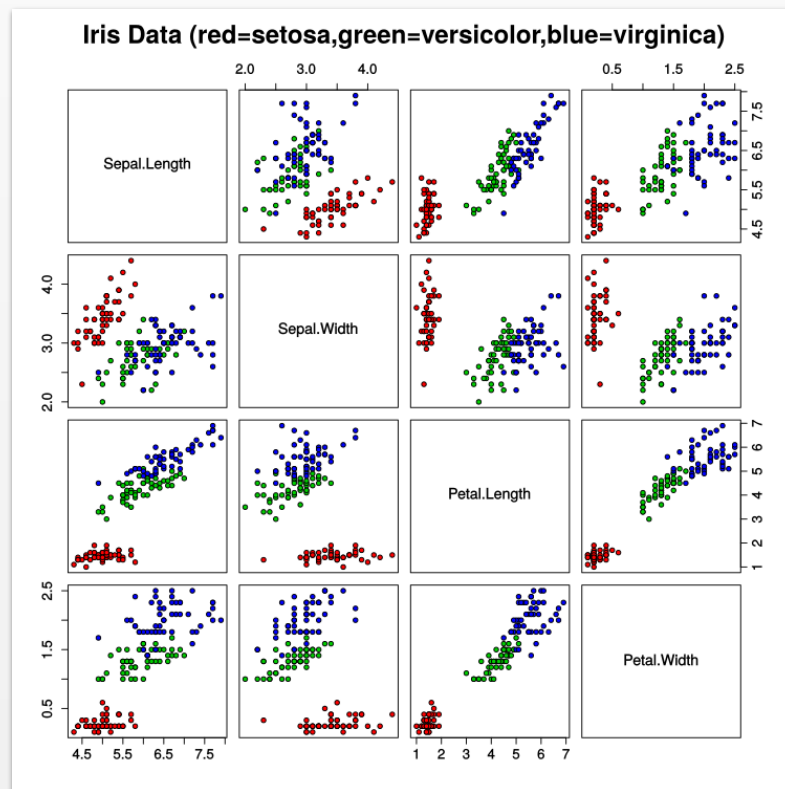
# Dimensionality Reduction

Shantanu Jain

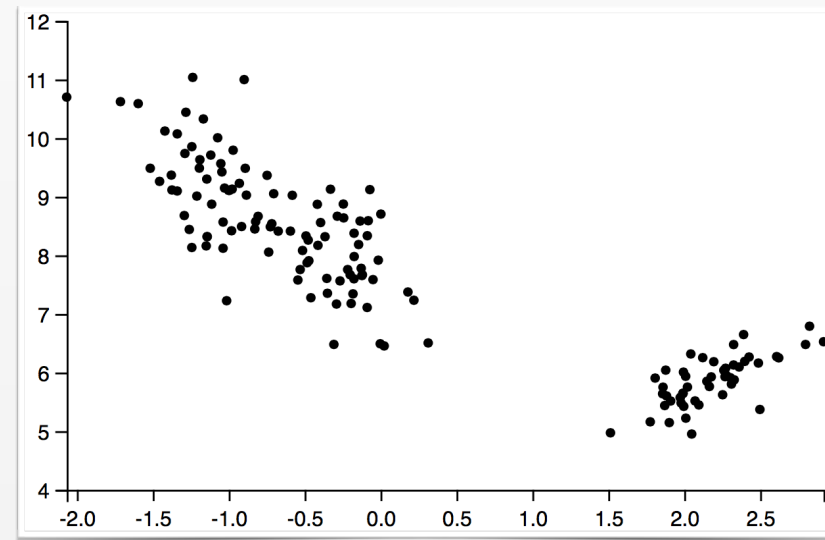
# Dimensionality Reduction

**Goal:** Map high dimensional data onto lower-dimensional data in a manner that preserves *distances/similarities*

**Original Data (4 dims)**



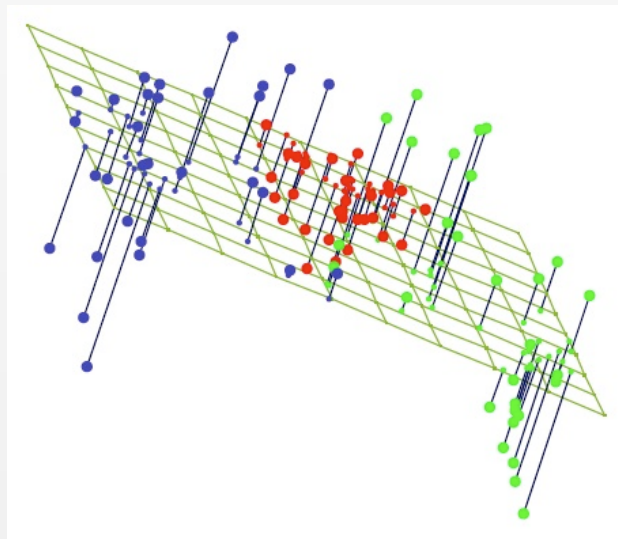
**Projection with PCA (2 dims)**



Objective: projection should “preserve” relative distances

# Linear Dimensionality Reduction

*Idea:* Project high-dimensional vector  
onto a lower dimensional space



$$\begin{array}{c} \mathbf{x} \in \mathbb{R}^{361} \\ \downarrow \mathbf{z} = \mathbf{U}^T \mathbf{x} \\ \mathbf{z} \in \mathbb{R}^{10} \end{array}$$

# Problem Setup

Given  $n$  data points in  $d$  dimensions:  $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d$

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

# Problem Setup

Given  $n$  data points in  $d$  dimensions:  $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d$

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

Want to reduce dimensionality from  $d$  to  $k$

Choose  $k$  directions  $\mathbf{u}_1, \dots, \mathbf{u}_k$

$$\mathbf{z} = \mathbf{U}^\top \mathbf{x} \quad \mathbf{U} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_k \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times k}$$

# Problem Setup

Given  $n$  data points in  $d$  dimensions:  $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d$

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

Want to reduce dimensionality from  $d$  to  $k$

Choose  $k$  directions  $\mathbf{u}_1, \dots, \mathbf{u}_k$

$$\mathbf{z} = \mathbf{U}^\top \mathbf{x} \quad \mathbf{U} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_k \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times k}$$

For each  $\mathbf{u}_j$ , compute “similarity”  $z_j = \mathbf{u}_j^\top \mathbf{x}$

# Problem Setup

Given  $n$  data points in  $d$  dimensions:  $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d$

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

Want to reduce dimensionality from  $d$  to  $k$

Choose  $k$  directions  $\mathbf{u}_1, \dots, \mathbf{u}_k$

$$\mathbf{U} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_k \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times k}$$

For each  $\mathbf{u}_j$ , compute “similarity”  $z_j = \mathbf{u}_j^\top \mathbf{x}$

Project  $\mathbf{x}$  down to  $\mathbf{z} = (z_1, \dots, z_k)^\top = \mathbf{U}^\top \mathbf{x}$

How to choose  $\mathbf{U}$ ?

# Background: Changes of Basis

Data

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

$\bar{\mathbf{z}} = \bar{\mathbf{U}}^T \mathbf{x}$  is a representation of  $\mathbf{x}$  w.r.t. the basis vectors in  $\bar{\mathbf{U}}$

Orthonormal Basis

$$\bar{\mathbf{U}} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_d \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times d}$$

$$\langle \mathbf{u}_i, \mathbf{u}_j \rangle = \delta_{i,j} = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases}$$

$$\bar{\mathbf{U}}^T \bar{\mathbf{U}} = \mathbf{I}_{d \times d}$$



# Background: Changes of Basis

Data

$$\mathbf{X} = \left( \begin{array}{c|c|c} & & \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ & & \end{array} \right) \in \mathbb{R}^{d \times n}$$

Orthonormal Basis

$$\bar{\mathbf{U}} = \left( \begin{array}{c|c|c} & & \\ \mathbf{u}_1 & \cdots & \mathbf{u}_d \\ & & \end{array} \right) \in \mathbb{R}^{d \times d}$$

$$\langle \mathbf{u}_i, \mathbf{u}_j \rangle = \delta_{i,j}$$

# Background: Changes of Basis

Data

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

Orthonormal Basis

$$\bar{\mathbf{U}} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_d \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times d}$$

Change of basis

$$\bar{\mathbf{z}} = (z_1, \dots, z_d)^\top$$

$$z_j = \mathbf{u}_j^\top \mathbf{x}$$

$$\bar{\mathbf{z}} = \bar{\mathbf{U}}^\top \mathbf{x}$$

Inverse Change of basis

$$\mathbf{x} = \bar{\mathbf{U}} \bar{\mathbf{z}} = \sum_{j=1}^d z_j \mathbf{u}_j$$

# Properties of orthonormal matrices

For an orthonormal matrix  $\bar{U} \in \mathbf{R}^{d \times d}$

$$\bar{U}^T \bar{U} = \bar{U} \bar{U}^T = I_{d \times d}$$

An orthonormal matrix has  $d$  orthogonal vectors of dimension  $d$  and unit length

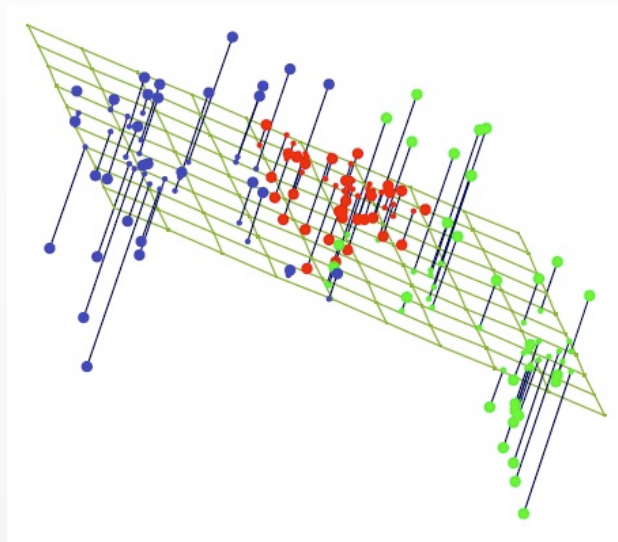
For a semi-orthonormal matrix  $U \in \mathbf{R}^{d \times k}$ , where  $k < d$

$$U^T U = I_{k \times k}$$

$$U U^T \neq I_{d \times d}$$

An semi-orthonormal matrix has  $k$  orthogonal vectors of dimension  $d$  and unit length

# Principal Component Analysis



$$\mathbf{x} \in \mathbb{R}^{361}$$

$$\mathbf{z} = \mathbf{U}^T \mathbf{x} \quad \mathbf{U} \text{ is } d \times k$$

$$\mathbf{z} \in \mathbb{R}^{10}$$

We are back to the PCA setting with  $\mathbf{U}$  containing fewer than  $d$  columns

## Optimize two equivalent objectives

1. Minimize the reconstruction error
2. Maximizes the projected variance

# PCA Objective 1: Reconstruction Error

**U** serves two functions:

- Encode:  $\mathbf{z} = \mathbf{U}^\top \mathbf{x}$ ,  $z_j = \mathbf{u}_j^\top \mathbf{x}$

# PCA Objective 1: Reconstruction Error

$\mathbf{U}$  serves two functions:

- Encode:  $\mathbf{z} = \mathbf{U}^\top \mathbf{x}$ ,  $z_j = \mathbf{u}_j^\top \mathbf{x}$
- Decode:  $\tilde{\mathbf{x}} = \mathbf{U}\mathbf{z} = \sum_{j=1}^k z_j \mathbf{u}_j$

# PCA Objective 1: Reconstruction Error

$\mathbf{U}$  serves two functions:

- Encode:  $\mathbf{z} = \mathbf{U}^\top \mathbf{x}$ ,  $z_j = \mathbf{u}_j^\top \mathbf{x}$
- Decode:  $\tilde{\mathbf{x}} = \mathbf{U}\mathbf{z} = \sum_{j=1}^k z_j \mathbf{u}_j$

Want reconstruction error  $\|\mathbf{x} - \tilde{\mathbf{x}}\|$  to be small

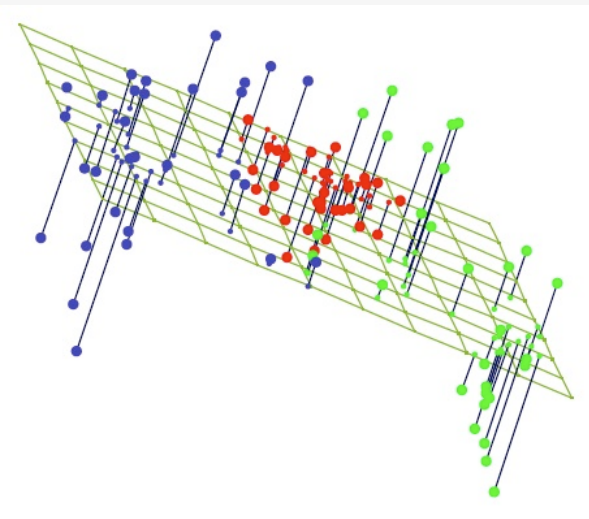
# PCA Objective 1: Reconstruction Error

**U** serves two functions:

- Encode:  $\mathbf{z} = \mathbf{U}^\top \mathbf{x}$ ,  $z_j = \mathbf{u}_j^\top \mathbf{x}$
- Decode:  $\tilde{\mathbf{x}} = \mathbf{U}\mathbf{z} = \sum_{j=1}^k z_j \mathbf{u}_j$

Want reconstruction error  $\|\mathbf{x} - \tilde{\mathbf{x}}\|$  to be small

**Objective:** minimize total squared reconstruction error



$$\min_{\mathbf{U} \in \mathbf{R}^{d \times k}, \mathbf{U}^\top \mathbf{U} = \mathbf{I}} \text{RE}(\mathbf{U})$$

$$\begin{aligned} \text{RE}(\mathbf{U}) &= \frac{1}{n} \sum_{i=1}^n \|x_i - \mathbf{U}\mathbf{U}^\top x_i\|^2 \\ &= \frac{1}{n} \sum_{i=1}^n \|(I - \mathbf{U}\mathbf{U}^\top)x_i\|^2 \\ &= \hat{\mathbf{E}}[\|(I - \mathbf{U}\mathbf{U}^\top)x\|^2] \end{aligned}$$

Mathematically, the expectation is w.r.t the empirical distribution of the data that gives an equal probability of  $1/n$  to each point.



# Total Variance

- Define the **Total Variance** of  $x \in \mathbf{R}^d$  as the **sum of variances across all dimensions**.
- It is estimated from the observed data as the sum of the diagonal elements of the covariance matrix

$$\text{Var}_T(x) = \text{tr} \left( \frac{1}{n} X X^T \right)$$

- It can also be expressed as

$$\begin{aligned} \text{Var}_T(x) &= \frac{1}{n} \sum_{i=1}^n \|x_i\|^2 \\ &= \hat{\mathbf{E}}[\|x\|^2] \end{aligned}$$

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

Assuming that the matrix  $\mathbf{X}$  is centered;  $\hat{\mathbf{E}}[x] = \frac{1}{n} \sum_{i=1}^n x_i = 0$

$$\|x_i\|^2 = x_{i1}^2 + x_{i2}^2 + \dots + x_{id}^2$$

Variance across dimension  $j$

$$\text{Var}(x_{\cdot j}) = \frac{1}{n} \sum_{i=1}^n x_{ij}^2$$

This is because the mean for each dimension is  $0$ .

# Projected Variance

- Let  $z = U^T x$  be the projection of  $x$

$$\begin{aligned}\text{Var}_T(z) &= \text{tr} \left( \frac{1}{n} Z Z^T \right) \\ &= \text{tr} \left( \frac{1}{n} U^T X X^T U \right)\end{aligned}$$

- It can also be expressed as

$$\begin{aligned}\text{Var}_T(z) &= \frac{1}{n} \sum_{i=1}^n \| U^T x_i \|^2 \\ &= \hat{\mathbf{E}} [\| U^T x \|^2]\end{aligned}$$

$$\max_{U \in \mathbf{R}^{d \times k}, U^T U = I} \text{Var}_T(z; U)$$

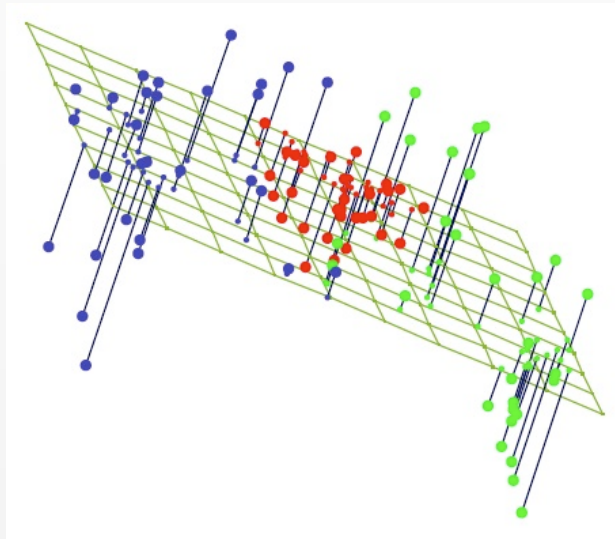
$Z = U^T X$ : contains the projections of all points

$$Z = [z_1, z_2, \dots, z_n], z_i \in \mathbf{R}^k$$

Note that the variance formulas are true for the  $Z$  matrix as well since  $\hat{\mathbf{E}}[z] = \hat{\mathbf{E}}[U^T x] = U^T \hat{\mathbf{E}}[x] = 0$

The steps above come from linearity of expectation and because we have assumed that  $X$  is centered; i.e.,  $\hat{\mathbf{E}}[x] = 0$

# Projected Variance



$$\begin{array}{c} \mathbf{x} \in \mathbb{R}^{361} \\ \downarrow \mathbf{z} = \mathbf{U}^T \mathbf{x} \\ \mathbf{z} \in \mathbb{R}^{10} \end{array}$$

# Equivalence of two objectives

Key intuition:

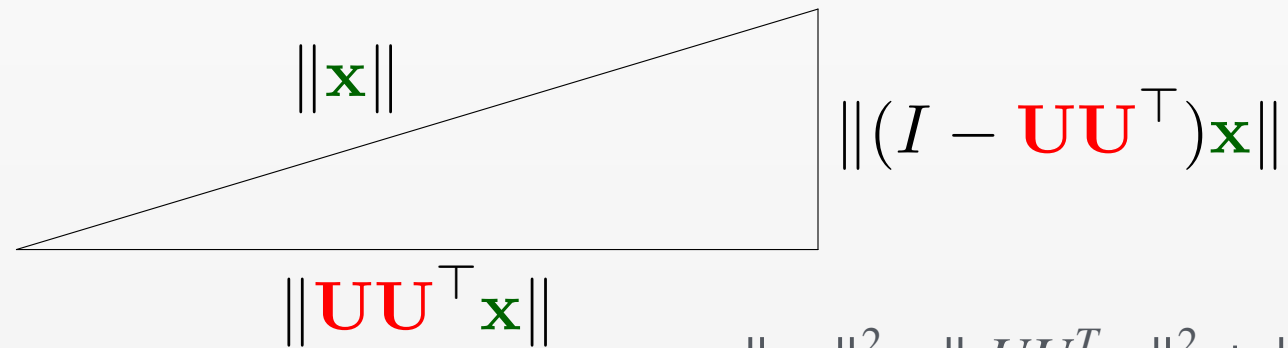
$$\underbrace{\text{variance of data}}_{\text{fixed}} = \underbrace{\text{captured variance}}_{\text{want large}} + \underbrace{\text{reconstruction error}}_{\text{want small}}$$

# Equivalence of two objectives

Key intuition:

$$\underbrace{\text{variance of data}}_{\text{fixed}} = \underbrace{\text{captured variance}}_{\text{want large}} + \underbrace{\text{reconstruction error}}_{\text{want small}}$$

Pythagorean decomposition:  $\mathbf{x} = \mathbf{UU}^\top \mathbf{x} + (I - \mathbf{UU}^\top) \mathbf{x}$



$$\begin{aligned} \|x\|^2 &= \|UU^\top x\|^2 + \|(I - UU^\top)x\|^2 \\ &= \|U^\top x\|^2 + \|(I - UU^\top)x\|^2 \end{aligned}$$

Take expectations;

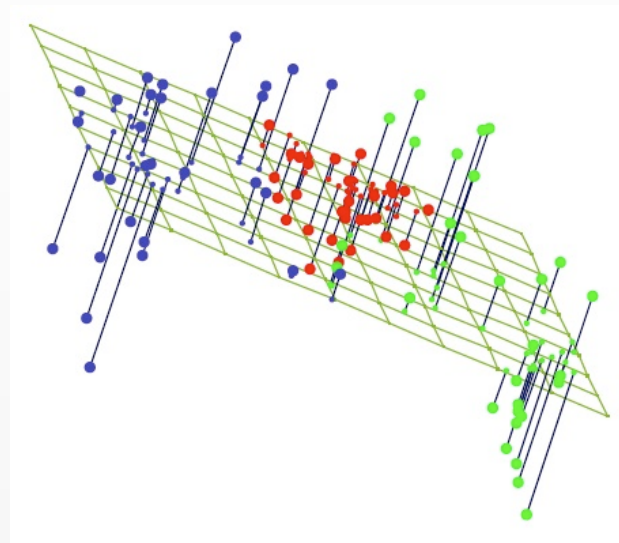
$$\hat{\mathbb{E}}[\|\mathbf{x}\|^2] = \hat{\mathbb{E}}[\|\mathbf{U}^\top \mathbf{x}\|^2] + \hat{\mathbb{E}}[\|\mathbf{x} - \mathbf{UU}^\top \mathbf{x}\|^2]$$



# Dimensionality Reduction

Shantanu Jain

# Principal Component Analysis



$$\begin{array}{c} \mathbf{x} \in \mathbb{R}^{361} \\ \downarrow \mathbf{z} = \mathbf{U}^\top \mathbf{x} \\ \mathbf{z} \in \mathbb{R}^{10} \end{array}$$

## Optimize two equivalent objectives

1. Minimize the reconstruction error

$$\hat{\mathbb{E}}[||\mathbf{x} - \mathbf{U}\mathbf{z}||^2] = \hat{\mathbb{E}}[||(I - \mathbf{U}\mathbf{U}^\top)\mathbf{x}||^2]$$

2. Maximizes the projected variance

$$\hat{\mathbb{E}}[\mathbf{z}^\top \mathbf{z}] = \hat{\mathbb{E}}[\mathbf{x}^\top \mathbf{U}\mathbf{U}^\top \mathbf{x}]$$

# Total variance unaltered by basis change

$$\bar{\mathbf{z}}^t \bar{\mathbf{z}} = x^t (\bar{\mathbf{U}} \bar{\mathbf{U}}^t) x = x^t x$$

$$\bar{\mathbf{U}} \bar{\mathbf{U}}^T = I_{d \times d}$$

$\bar{\mathbf{U}}^{-1} = \bar{\mathbf{U}}^T$  when  $\bar{\mathbf{U}}$  contains all  $d$  orthonormal basis, otherwise the inverse is undefined

$$\hat{\mathbf{E}}[\|\bar{\mathbf{z}}\|^2] = \hat{\mathbf{E}}[\|x\|^2]$$

$$\text{Var}_T(x) = \text{Var}_T(\bar{\mathbf{z}})$$

$$\text{tr}\left(\frac{1}{n} X X^T\right) = \text{tr}\left(\frac{1}{n} \bar{\mathbf{U}}^T X X^T \bar{\mathbf{U}}\right)$$

Data

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

Orthonormal Basis

$$\bar{\mathbf{U}} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_d \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times d}$$

Change of basis

$$\mathbf{z} = \bar{\mathbf{U}}^T \mathbf{x} \quad \mathbf{x} = \bar{\mathbf{U}} \mathbf{z}$$

$$\bar{\mathbf{U}}^T \bar{\mathbf{U}} = I_{d \times d}$$



# Eigenvectors of the Covariance

Data

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

Orthonormal Basis

$$\bar{\mathbf{U}} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_d \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times d}$$

Eigenvectors of Covariance

$$\mathbf{C} = \frac{1}{n} \sum_{j=1}^n \mathbf{x}_j \mathbf{x}_j^\top = \frac{1}{n} \mathbf{X} \mathbf{X}^\top$$

$$\mathbf{C} \mathbf{u}_j = \lambda_j \mathbf{u}_j$$

$$\mathbf{C} \bar{\mathbf{U}} = \bar{\mathbf{U}} \mathbf{\Lambda}$$

$$\mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \cdots & \\ & & & \lambda_d \end{pmatrix}$$

*Claim:* Eigenvectors of a symmetric matrix are orthogonal

# Proof: Eigenvectors are Orthogonal

For any real matrix  $A$  and any vectors  $\mathbf{x}$  and  $\mathbf{y}$ , we have

$$\langle A\mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, A^T \mathbf{y} \rangle.$$

Now assume that  $A$  is symmetric, and  $\mathbf{x}$  and  $\mathbf{y}$  are eigenvectors of  $A$  corresponding to distinct eigenvalues  $\lambda$  and  $\mu$ . Then

$$\lambda \langle \mathbf{x}, \mathbf{y} \rangle = \langle \lambda \mathbf{x}, \mathbf{y} \rangle = \langle A\mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, A^T \mathbf{y} \rangle = \langle \mathbf{x}, A\mathbf{y} \rangle = \langle \mathbf{x}, \mu \mathbf{y} \rangle = \mu \langle \mathbf{x}, \mathbf{y} \rangle.$$

Therefore,  $(\lambda - \mu) \langle \mathbf{x}, \mathbf{y} \rangle = 0$ . Since  $\lambda - \mu \neq 0$ , then  $\langle \mathbf{x}, \mathbf{y} \rangle = 0$ , i.e.,  $\mathbf{x} \perp \mathbf{y}$ .

Now find an orthonormal basis for each eigenspace; since the eigenspaces are mutually orthogonal, these vectors together give an orthonormal subset of  $\mathbb{R}^n$ . Finally, since symmetric matrices are diagonalizable, this set will be a basis (just count dimensions). The result you want now follows.

[share](#) [cite](#) [improve this answer](#)

answered Nov 15 '11 at 21:18



[Arturo Magidin](#)

219k ● 20 ■ 479 ▲ 780

*(from stack exchange)*

# Eigenvectors of the Covariance

Data

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

Orthonormal Basis

$$\bar{\mathbf{U}} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_d \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times d}$$

Eigenvectors of Covariance

$$\mathbf{C} = \frac{1}{n} \sum_{j=1}^n \mathbf{x}_j \mathbf{x}_j^\top = \frac{1}{n} \mathbf{X} \mathbf{X}^\top$$
$$\mathbf{C} \mathbf{u}_j = \lambda_j \mathbf{u}_j$$

Eigen-decomposition

$$\mathbf{C} = \bar{\mathbf{U}} \mathbf{\Lambda} \bar{\mathbf{U}}^\top$$
$$\mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \cdots & \\ & & & \lambda_d \end{pmatrix}$$

$$\begin{aligned} \mathbf{C} \bar{\mathbf{U}} &= \bar{\mathbf{U}} \mathbf{\Lambda} \\ \Rightarrow \mathbf{C} \bar{\mathbf{U}} \bar{\mathbf{U}}^\top &= \bar{\mathbf{U}} \mathbf{\Lambda} \bar{\mathbf{U}}^\top \\ \Rightarrow \mathbf{C} &= \bar{\mathbf{U}} \mathbf{\Lambda} \bar{\mathbf{U}}^\top \end{aligned}$$

Because  $\bar{\mathbf{U}}$  is  
orthonormal

# Total variance

$$\begin{aligned}\mathrm{tr}\left(\frac{1}{n}XX^T\right) &= \mathrm{Var}_T(x) \\ &= \mathrm{Var}_T(\bar{z}) \\ &= \mathrm{tr}\left(\frac{1}{n}\bar{U}^TXX^T\bar{U}\right) \\ &= \mathrm{tr}(\bar{U}^T\bar{U}\Lambda\bar{U}^T\bar{U}) \\ &= \mathrm{tr}(\Lambda) \\ &= \sum_{i=1}^n \lambda_i\end{aligned}$$

For  $\bar{z} = \bar{U}^T x$ , where  $\bar{U}$  contains all  $d$  eigenvectors of  $\frac{1}{n}XX^T$ , which are orthonormal by definition.

The total variance can be expressed as sum of the eigenvalues of the covariance matrix.

# Principal Component Analysis

*Idea:* Take **top-k** eigenvectors to maximize variance

1) Sort the eigenvalues in descending order

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$$

2) Sort the corresponding eigenvectors accordingly.

3) Construct a projection matrix with the top-k eigenvectors

$$\mathbf{U} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_k \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times k}$$

# Principal Component Analysis

Data

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

**Truncated** Basis

$$\mathbf{U} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_k \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times k}$$

Eigenvectors of Covariance

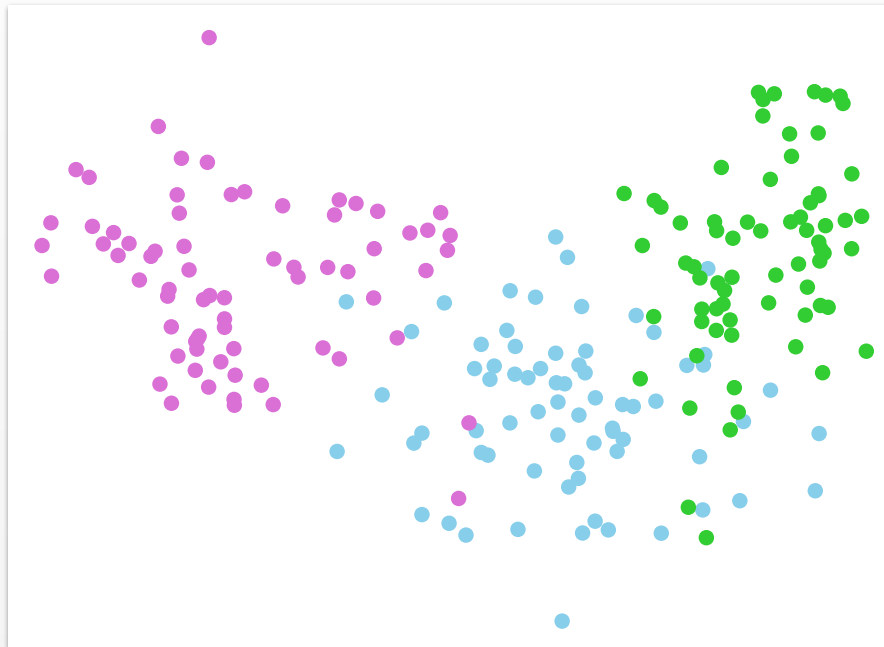
$$\mathbf{C} = \frac{1}{n} \sum_{j=1}^n \mathbf{x}_j \mathbf{x}_j^\top = \frac{1}{n} \mathbf{X} \mathbf{X}^\top$$
$$\mathbf{C} \mathbf{u}_j = \lambda_j \mathbf{u}_j$$

**Truncated** decomposition

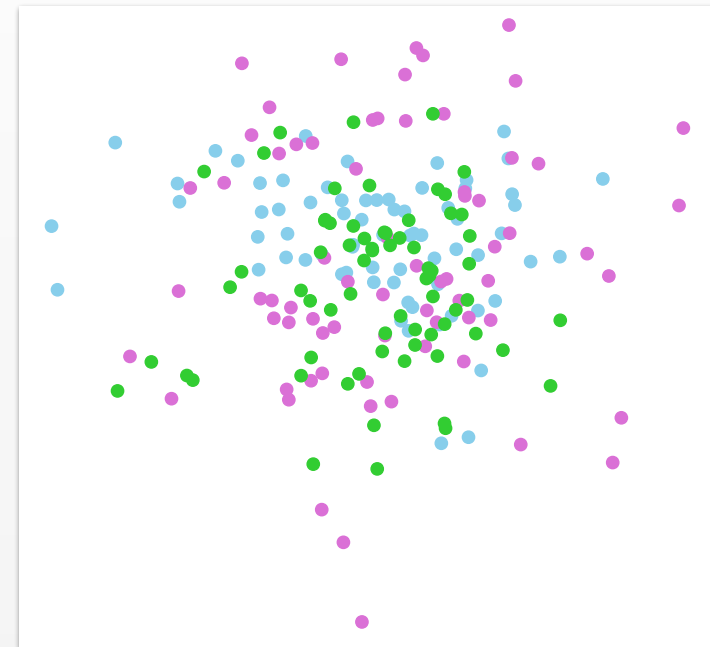
$$\mathbf{C} \simeq \mathbf{U} \mathbf{\Lambda}^{(k)} \mathbf{U}^\top$$
$$\mathbf{\Lambda}^{(k)} = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \cdots & \\ & & & \lambda_k \end{pmatrix}$$

# Principal Component Analysis

Top 2 components



Bottom 2 components



**Data:** three varieties of wheat: Kama, Rosa, Canadian

**Attributes:** Area, Perimeter, Compactness, Length of Kernel, Width of Kernel, Asymmetry Coefficient, Length of Groove

# PCA: Complexity

Data

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

$$\mathbf{C} = \frac{1}{n} \sum_{j=1}^n \mathbf{x}_j \mathbf{x}_j^\top = \frac{1}{n} \mathbf{X} \mathbf{X}^\top$$

Truncated Basis

$$\mathbf{U} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_k \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times k}$$

$$\mathbf{C} \mathbf{u}_j = \lambda_j \mathbf{u}_j$$

Using eigen-value decomposition

- Computation of covariance  $\mathbf{C}$ :  $O(n d^2)$
- Eigen-value decomposition:  $O(d^3)$
- Total complexity:  $O(n d^2 + d^3)$



# PCA: Complexity

Data

$$\mathbf{X} = \begin{pmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times n}$$

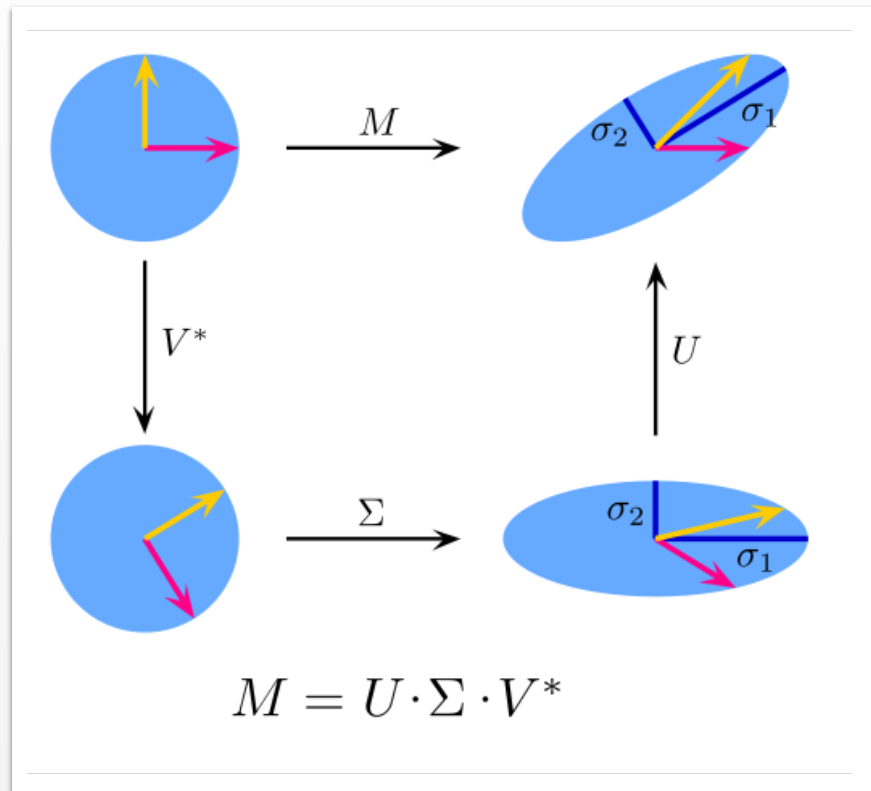
Truncated Basis

$$\mathbf{U} = \begin{pmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_k \\ | & & | \end{pmatrix} \in \mathbb{R}^{d \times k}$$

Using singular-value decomposition

- Full decomposition:  $O(\min\{nd^2, n^2d\})$
- Rank-k decomposition:  $O(k d n \log(n))$   
(with power method)

# Singular Value Decomposition



*Idea:* Decompose the  $d \times n$  matrix  $\mathbf{X}$  into

1. A  $n \times n$  basis  $\mathbf{V}$   
(unitary matrix)
2. A  $d \times n$  matrix  $\Sigma$   
(diagonal projection)
3. A  $d \times d$  basis  $\mathbf{U}$   
(unitary matrix)

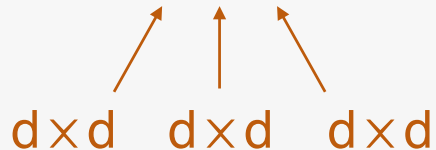
$$\mathbf{X} = \mathbf{U}_{d \times d} \Sigma_{d \times n} \mathbf{V}_{n \times n}^T$$

# Relationship Between SVD and PCA


The eigen-vectors of  $\frac{1}{n}XX^T$  can be obtained as the left singular vectors of  $X$

PCA (all  $d$  components)

SVD (all  $d$  components)

$$\frac{1}{n}XX^T = U\Lambda U^T$$


$d \times d$     $d \times d$     $d \times d$

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


$d \times d$     $d \times n$     $n \times n$

$$\begin{aligned}\frac{1}{n}XX^T &= \frac{1}{n}U\Sigma V^T V\Sigma^T U^T \\ &= \frac{1}{n}U\Sigma I \Sigma^T U^T \\ &= \frac{1}{n}U\Sigma\Sigma^T U^T\end{aligned}$$

Relationship  $\Lambda$  and  $\Sigma$

$$\Lambda = \frac{1}{n}\Sigma\Sigma^T$$

# Computing Principal Components

Method 1: eigendecomposition

**U** are eigenvectors of covariance matrix  $C = \frac{1}{n} \mathbf{X} \mathbf{X}^\top$

# Computing Principal Components

Method 1: eigendecomposition

**U** are eigenvectors of covariance matrix  $C = \frac{1}{n} \mathbf{X} \mathbf{X}^\top$

Computing  $C$  already takes  $O(nd^2)$  time (very expensive)

# Computing Principal Components

Method 1: eigendecomposition

$\mathbf{U}$  are eigenvectors of covariance matrix  $C = \frac{1}{n}\mathbf{X}\mathbf{X}^\top$

Computing  $C$  already takes  $O(nd^2)$  time (very expensive)

Method 2: singular value decomposition (SVD)

Find  $\mathbf{X} = \mathbf{U}_{d \times d} \Sigma_{d \times n} \mathbf{V}_{n \times n}^\top$

where  $\mathbf{U}^\top \mathbf{U} = I_{d \times d}$ ,  $\mathbf{V}^\top \mathbf{V} = I_{n \times n}$ ,  $\Sigma$  is diagonal

# Computing Principal Components

Method 1: eigendecomposition

$\mathbf{U}$  are eigenvectors of covariance matrix  $C = \frac{1}{n}\mathbf{X}\mathbf{X}^\top$

Computing  $C$  already takes  $O(nd^2)$  time (very expensive)

Method 2: singular value decomposition (SVD)

Find  $\mathbf{X} = \mathbf{U}_{d \times d} \Sigma_{d \times n} \mathbf{V}_{n \times n}^\top$

where  $\mathbf{U}^\top \mathbf{U} = I_{d \times d}$ ,  $\mathbf{V}^\top \mathbf{V} = I_{n \times n}$ ,  $\Sigma$  is diagonal

Computing top  $k$  singular vectors takes only  $O(ndk)$

# Computing Principal Components

Method 1: eigendecomposition

$\mathbf{U}$  are eigenvectors of covariance matrix  $C = \frac{1}{n}\mathbf{X}\mathbf{X}^\top$

Computing  $C$  already takes  $O(nd^2)$  time (very expensive)

Method 2: singular value decomposition (SVD)

Find  $\mathbf{X} = \mathbf{U}_{d \times d} \Sigma_{d \times n} \mathbf{V}_{n \times n}^\top$

where  $\mathbf{U}^\top \mathbf{U} = I_{d \times d}$ ,  $\mathbf{V}^\top \mathbf{V} = I_{n \times n}$ ,  $\Sigma$  is diagonal

Computing top  $k$  singular vectors takes only  $O(ndk)$

Relationship between eigendecomposition and SVD:

Left singular vectors are principal components



# Probabilistic Interpretation

Generative Model [Tipping and Bishop, 1999]:

For each data point  $i = 1, \dots, n$ :

Draw the latent vector:  $\mathbf{z}_i \sim \mathcal{N}(0, I_{k \times k})$

Create the data point:  $\mathbf{x}_i \sim \mathcal{N}(\mathbf{U}\mathbf{z}_i, \sigma^2 I_{d \times d})$

PCA finds the  $\mathbf{U}$  that maximizes the likelihood of the data

$$\max_{\mathbf{U}} p(\mathbf{X} \mid \mathbf{U})$$