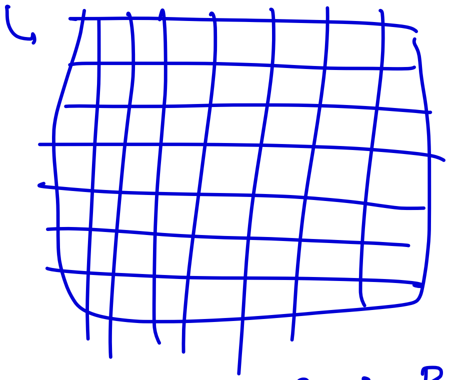
Centers that has the best loss.

Lecture 20 :

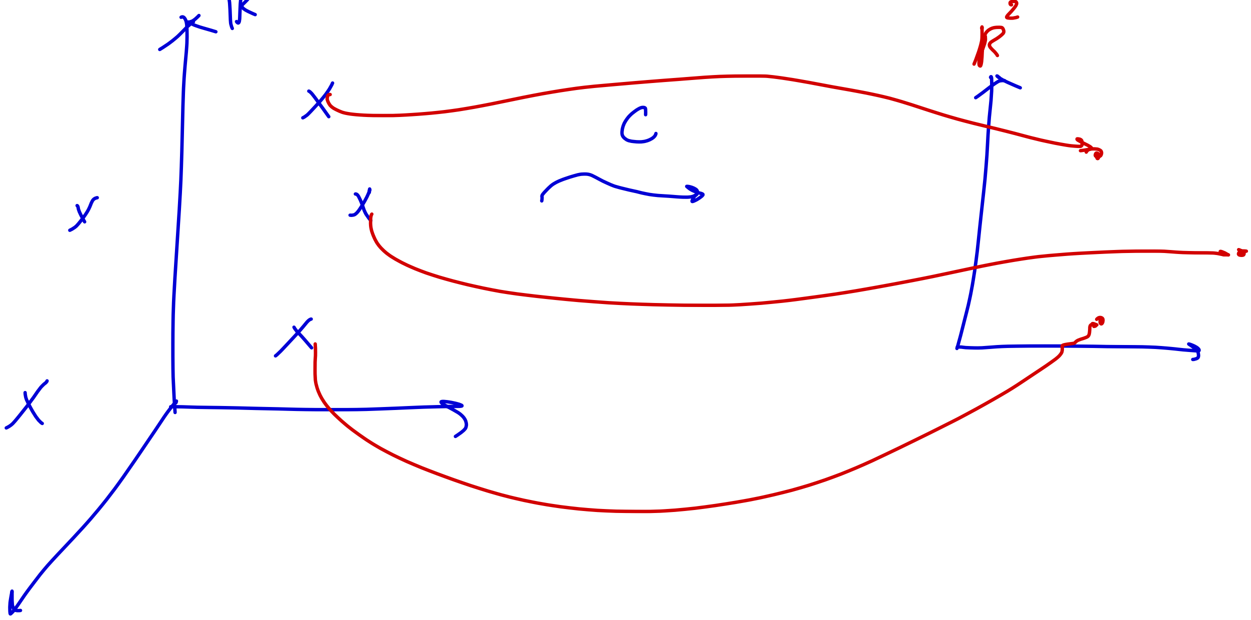
unsupervised learning
- clustering
- dimensionality reduction
  - PCA

$\{x_1, x_2, \dots, x_n\}$

image

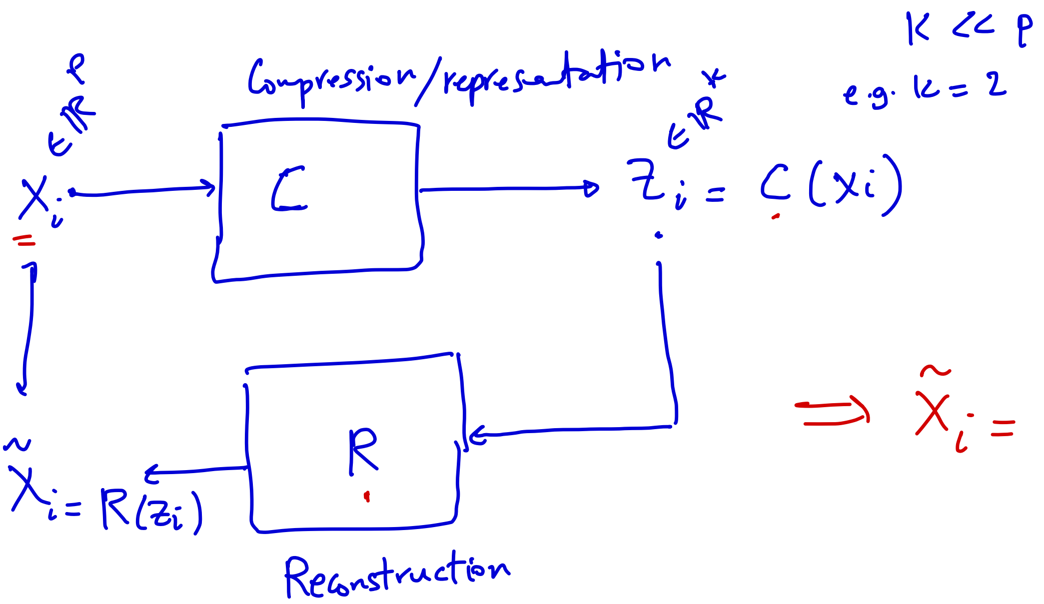


$\mathbb{R}^p \rightarrow P$: # pixels



- $x_1, x_2, \ldots, x_n \in \mathbb{R}^p$ (p is typically large)

- we'd like to represent the data in a low-dimensional space (e.g. $\mathbb{R}^2$)



$\Rightarrow \tilde{x}_i = R \circ C(x_i)$

$R(C(x_i))$

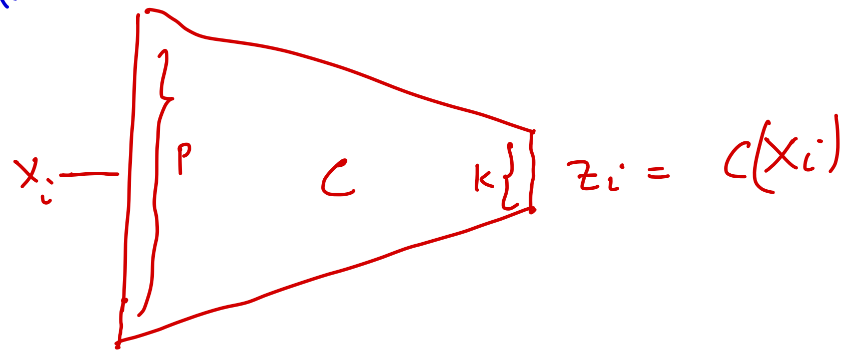
our goal is to design C, R such that x_i and $\tilde{x}_i$ are as close as possible.

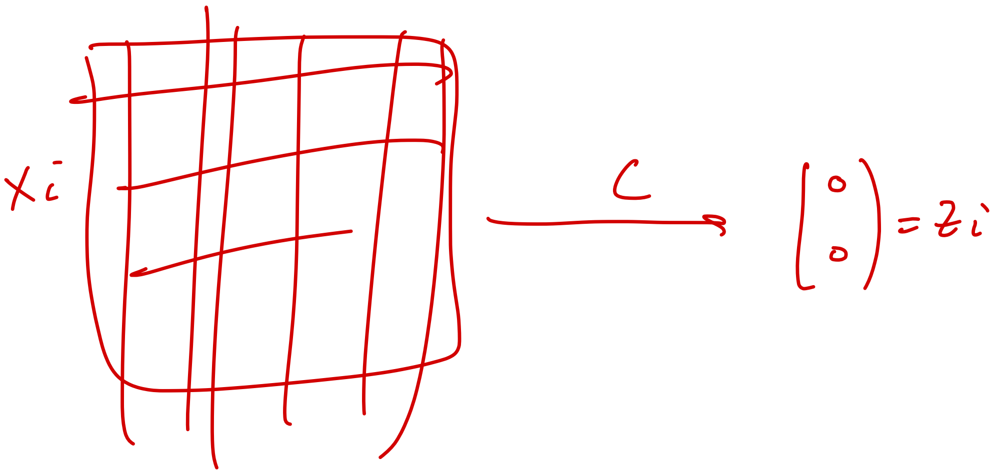
objective:

$$\underset{C,R}{\text{minimize}} \sum_{i=1}^{n} \|x_i - \tilde{x}_i\|_2^2$$

$$= \underset{C,R}{\text{minimize}} \sum_{i=1}^{n} \|x_i - R \circ C(x_i)\|_2^2$$

$$x_i = \begin{pmatrix} \vdots \\ \vdots \end{pmatrix} \in \mathbb{R}^p$$





As usual, let's start with the simplest class of mappings which are linear mappings.

matrix $\quad C = [\quad]_{k\times}$

$$z_i = C(x_i) = C x_i$$

e.g. $k=2 \longrightarrow C = \begin{bmatrix} \text{———} \\ \text{———} \end{bmatrix}_{2\times p}$

$$= \begin{bmatrix} u_1^T \\ \hline u_2^T \end{bmatrix}$$

$$\Rightarrow \quad z_i = C x_i = \begin{bmatrix} U_1^T x_i \\ U_2^T x_i \end{bmatrix}$$

Also, R = linear mapping

$$\tilde{x}_i = R z_i = R \cdot C \cdot x_i$$

R: $[\quad]_{p \times k}$, $R \in \mathbb{R}^{p \times k}$, $C \in \mathbb{R}^{k \times p}$

objective: $$\min_{R, C} \sum_{i=1}^{n} \| x_i - R.C.x_i \|_2^2 \qquad \text{(PCA)}$$

$R \in \mathbb{R}^{p \times k}$, $C \in \mathbb{R}^{k \times p}$

$C : \mathbb{R}^p \longrightarrow \mathbb{R}^k$

$C = (\text{matrix})_{k \times p}$

This problem is called the Principal Component Analysis (PCA)

In order to solve this problem, we need to review an important tool in linear algebra:

(this tool is important for many other branches of Data science)

The Singular Value Decomposition:

Theorem: Every symmetric matrix, $A_{p\times p}$, can be written as

$$A_{p\times p} = U^T_{p\times p} \Lambda_{p\times p} U_{p\times p}$$

- where $UU^T = I_{p\times p}$ identity matrix

$$\Lambda = \text{diagonal}: \begin{pmatrix} \lambda_1 & & 0 \\ & \lambda_2 & \\ 0 & & \ddots & \\ & & & \lambda_n \end{pmatrix}$$

- $A = U^T \cdot \begin{pmatrix} \lambda_1 & & 0 \\ & \lambda_2 & \\ 0 & & \ddots & \\ & & & \lambda_n \end{pmatrix} U$

- $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_n$

- $\lambda_i =$ eigenvalues of A.

- Each row of U is an "eigenvector" of the matrix A.

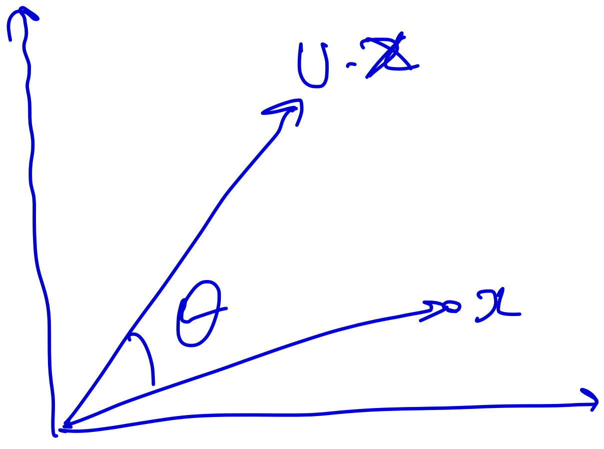
Example:

Assume A is 2×2 matrix.

$$A = \begin{pmatrix} a & c \\ c & b \end{pmatrix}$$

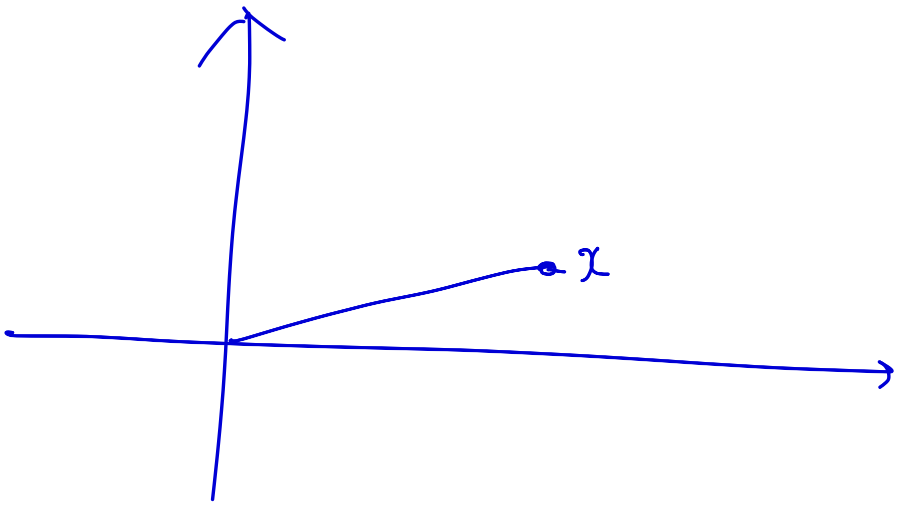
$$A = \underbrace{U}_{\text{Rotation}(\theta)} \cdot \overbrace{\underbrace{\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}}}^{\text{rescaling}} \cdot \underbrace{U^T}_{\text{Rotation}(-\theta)}$$

$$U \cdot U^T = I \longrightarrow U = \underbrace{\begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}}$$



$$\Lambda = \begin{pmatrix} 4 = \lambda_1 \geq 0 & 0 \\ 0 & 1 = \lambda_2 \geq 0 \end{pmatrix}$$ rescaling

Λx



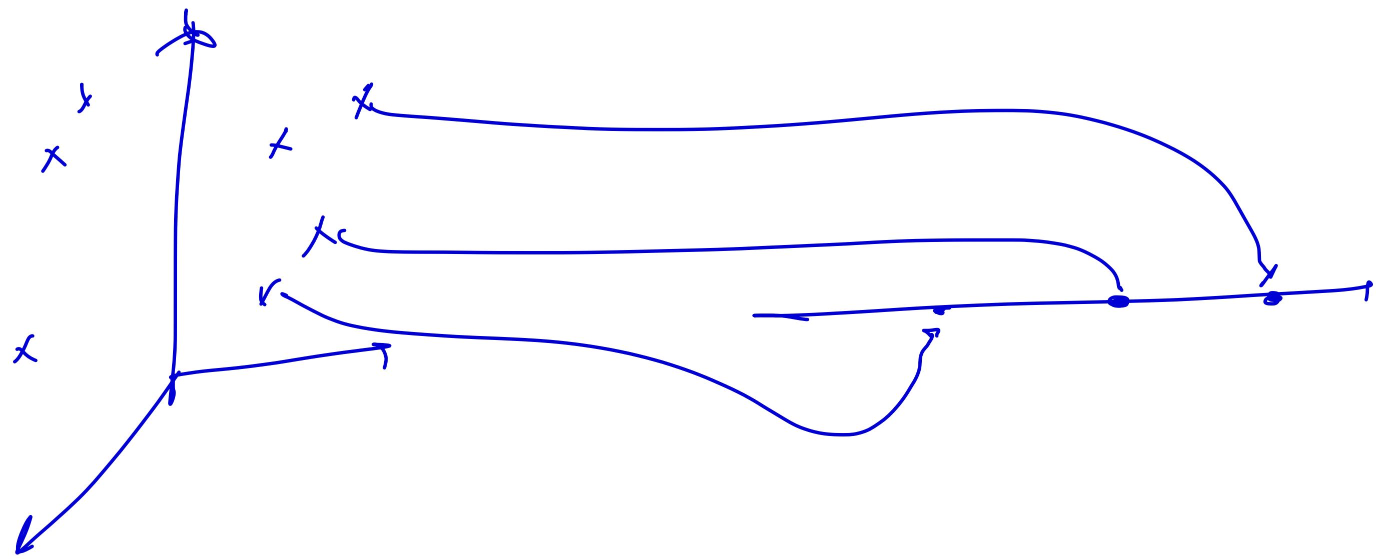
Λx

$$A = \text{Rotation}(-\theta) \cdot \text{Rescaling} \cdot \text{Rotation}(\theta)$$

---

let's go back to our problem.

let's assume for simplicity that $k=1$; we're looking for the best 1-dimensional representation of the data.

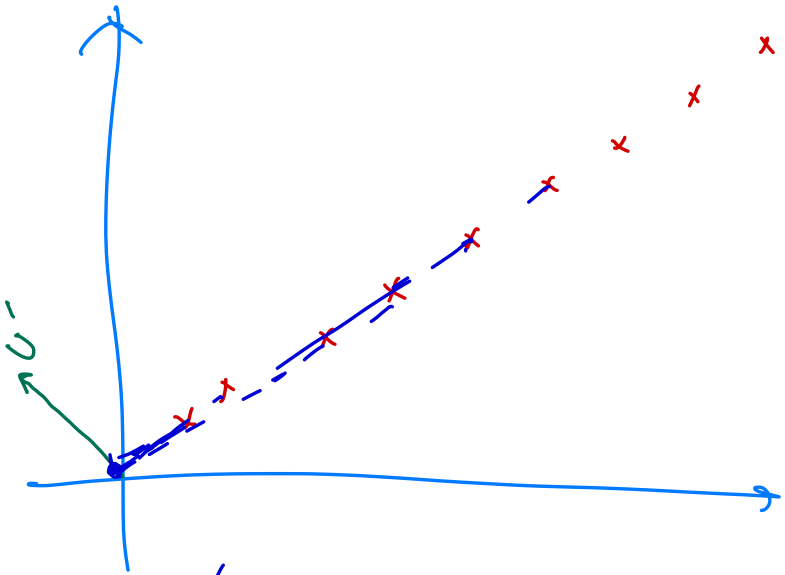


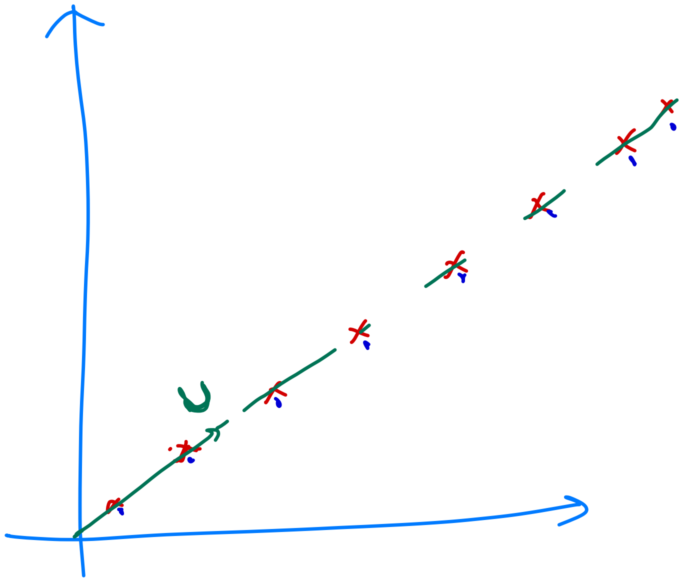
$$C = [\quad\quad]_{1\times p} \longrightarrow C = U^T$$

$$C \cdot X_i = \underset{\substack{= \\ \downarrow \\ \text{vector}}}{U^T} X_i$$

When $k=1$, our goal is to find a single vector U^T such that the representation $z_i = U^T X_i$ is "the most informative" representation.

Example:



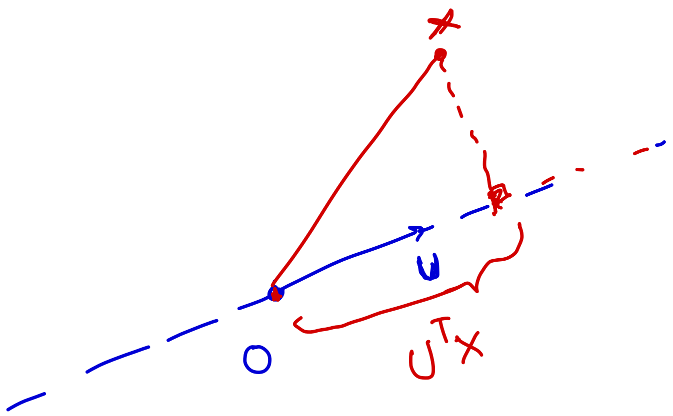


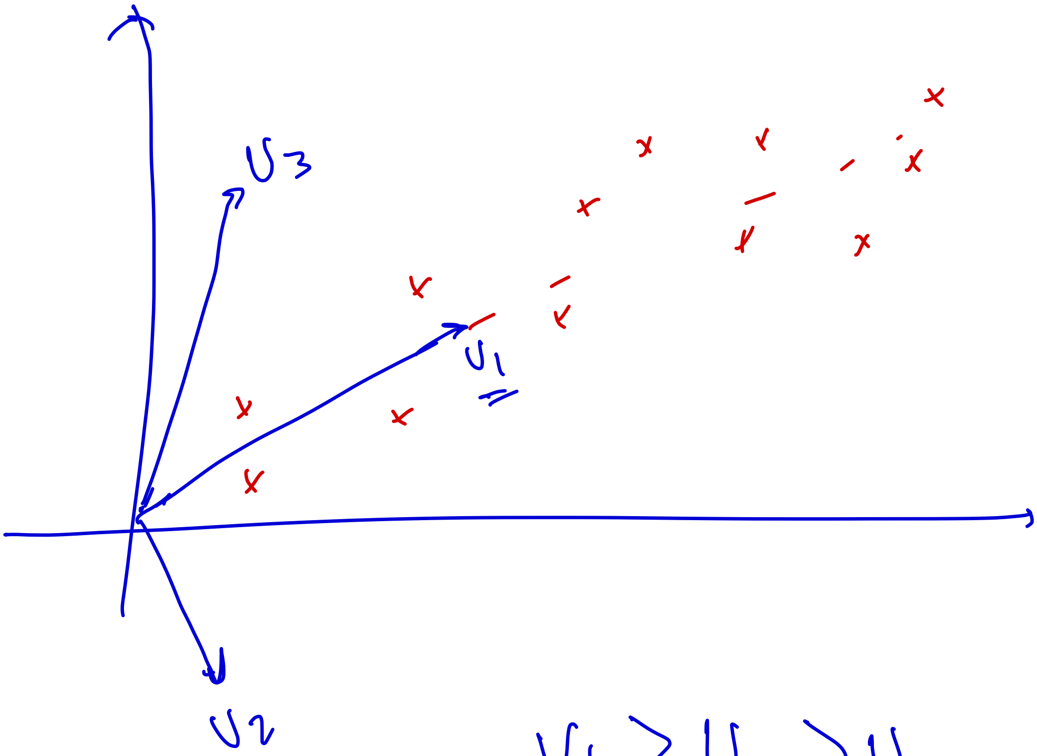
$u'^T x_i \approx 0$

$u^T x_i$

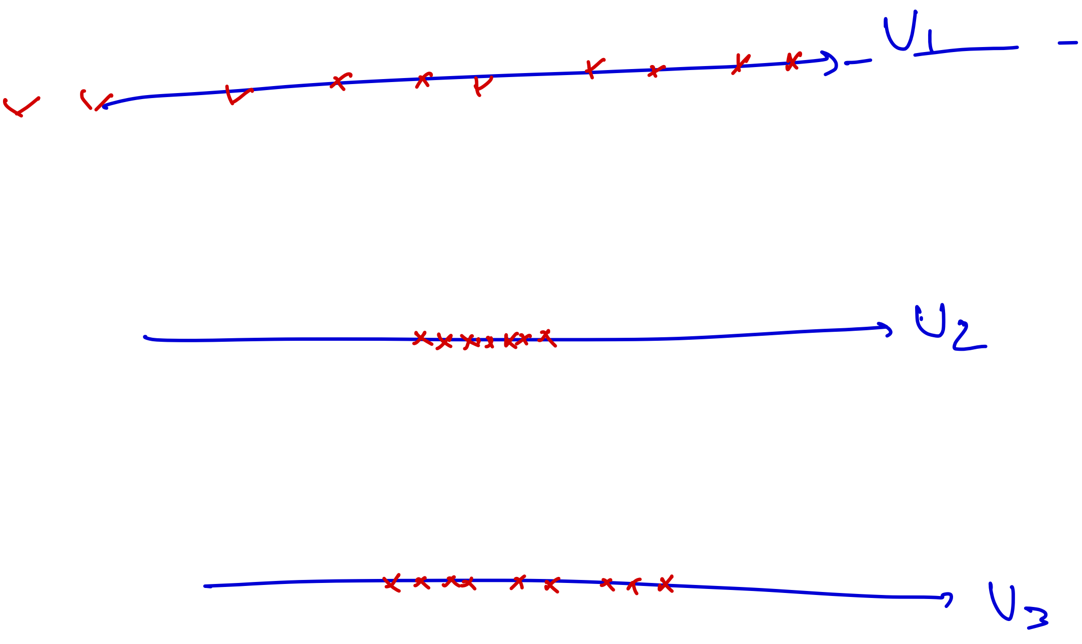
u is a better direction than u' in terms of capture more information about the data

Assume $\|u\|_2 = 1$





$$u_1 > u_3 > u_2$$



We're looking for a direction u along which data has the most "variation".

- Mathematically, we're looking for a direction u such the "variance" of $U^T \cdot X$ is the most.

- Assume that the data come from a distribution $X \sim P(X)$. Also assume for simplicity that $E[X] = 0$.

$$\max_{\substack{u^T \in \mathbb{R}^{1 \times p} \\ \|u\|_2 = 1}} \operatorname{Var}(u^T \cdot X)$$

$$= \max_{u : \|u\|_2 = 1} E\left[(u^T X)^2 \right]$$

$$= \max_{: \|U\|_2 = 1} E\left[u^T X^T X u \right]$$

$$= \max_{: \|U\|_2 = 1} u^T \underbrace{E[X X^T]}_{\text{Covariance matrix of the data}} u$$

$$\left(\begin{array}{l} X \in \mathbb{R}^p \\ E[XX^T] \\ = \Sigma_{p \times p} \end{array} \right)$$

Let $\Sigma \triangleq E[XX^T]$

$$= \max_{\substack{u: \\ \|u\|_2 = 1}} u^T \Sigma u$$

The solution of the problem above can be found from the singular value decomposition of Σ.

$$\Lambda = \begin{pmatrix} \lambda_1 & & 0 \\ & \lambda_2 & \\ 0 & & \ddots \ \lambda_p \end{pmatrix}$$

$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$

SVD $\longrightarrow \Sigma = U^T \Lambda U$

It can be shown the the first row of the matrix U, is the maximizer of the following problem:

$$\max_{u: \|u\|=} u^T \Sigma u$$

$$U = \begin{pmatrix} u_1^T \\ \hline u_2^T \\ \hline \vdots \\ \hline u_p^T \end{pmatrix}_{p \times p}$$

- u_1^T is the solution to the problem above.
- u_1^T is the principal component of the data

- v_1^T is the vector along which the data has the most "variation".

$$\Sigma = E[XX^T]$$

- We know that the data

$$x_1, x_2, \ldots, x_n \overset{iid}{\sim} \text{distribution}$$

$$X \sim p(x)$$

in practice:

- We cannot compute the matrix Σ because we do not know the distribution of the data

$$\Sigma \approx \underbrace{\frac{1}{n} \sum_{i=1}^{n} x_i x_i^T}_{\text{unbiased estimator}}$$

Algorithm (PCA) : (k=1)

- Input: $X_1, X_2, \ldots, X_n \in \mathbb{R}^p$

- Compute the empirical covariance matrix of data: $\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} X_i X_i^T$

- Compute the singular value decomposition of $\hat{\Sigma}$:

$$\hat{\Sigma} = U^T \Lambda U$$

- Use the first row of U, u_1, as the principal component of the data:

$$z_i = u_1^T X_i .$$

---

* let's now find the best 2-dimensional representation of the data.

$$\longrightarrow \max_{\substack{u_1, u_2 \\ \|u_1\|, \|u_2\| = 1 \\ \text{(orthogonal) } u_1^T u_2 = 0}} u_1^T \Sigma u_1 + u_2^T \Sigma u_2$$

It can be shown that u_1, u_2 are the first two rows of the matrix U.

* In general if we want to find a matrix $C = \begin{bmatrix} u_1^T \\ \hline u_2^T \\ \vdots \\ \hline u_k^T \end{bmatrix}_{k \times n}$ then

we need to solve the following optimization problem:

Cumulative variance of the data along the directions $u_1, \dots, u_k$ ←

$$\sum_{i=1}^{k} u_i^T \Sigma u_i$$

$$\text{s.t.} \quad \|u_i\| = 1$$

$$\text{and} \quad u_i^T u_j = 0 \quad \text{for } i \neq j$$

* it can be shown that the solution to the above optimization problem is the first k rows of the matrix U.

Algorithm PCA for general k:

- Input : $x_1, \dots, x_n \in \mathbb{R}^p$

- Compute: $$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^T$$

- Compute: (SVD) $$\hat{\Sigma} = U^T \Lambda U$$

- $$C = \begin{bmatrix} u_1^T \\ \hline u_2^T \\ \vdots \\ \hline u_k^T \end{bmatrix}$$ where $u_1, \dots, u_k$ are the first k rows of the matrix U.

It can be proven that PCA gives the optimal linear representation of the data (it solves equation (PCA) exactly).