CS725

Recall in k-means the i^{th} point is assigned to a single cluster that is closest to it w.r.t its cluster centroid (in terms of Euclidean distance). C^i was a 1-hot assignment.

What if we want a soft/probabilistic assignment?

$k=4$

$C^i = [0.8, 0.2, 0, 0] \longrightarrow$ meaning x_i is assigned to cluster 1 with probability 0.8 and cluster 2 with probability 0.2

"SOFT CLUSTERING"

Desired soft clustering algorithm

Initialize the cluster centroids

Repeat until convergence

Assignment step: $c_k^i = \dfrac{\exp\left(-\beta \|\mu_k - x_i\|^2\right)}{\sum_j \exp\left(-\beta \|\mu_j - x_i\|^2\right)}$

Update step: $\mu_k = \dfrac{\sum_i c_k^i x_i}{\sum_i c_k^i}$

How do we set β?

Expectation Maximization (EM) algorithm

EM is guaranteed to converge to a local optimum

Dimensionality Reduction

"Curse of dimensionality" → higher dimensions require an exponentially larger number of datapoints to cover the subspace in question

Dimensionality reduction (Motivation):

1. Better generalization (largely because data lies in low-dimensional subspaces)
2. Lossy compression
3. Allows for visualization (if d reduced to 2 or 3)

Popular dimensionality technique $\Rightarrow$ PCA (Principal Components Analysis)

PRINCIPAL COMPONENTS

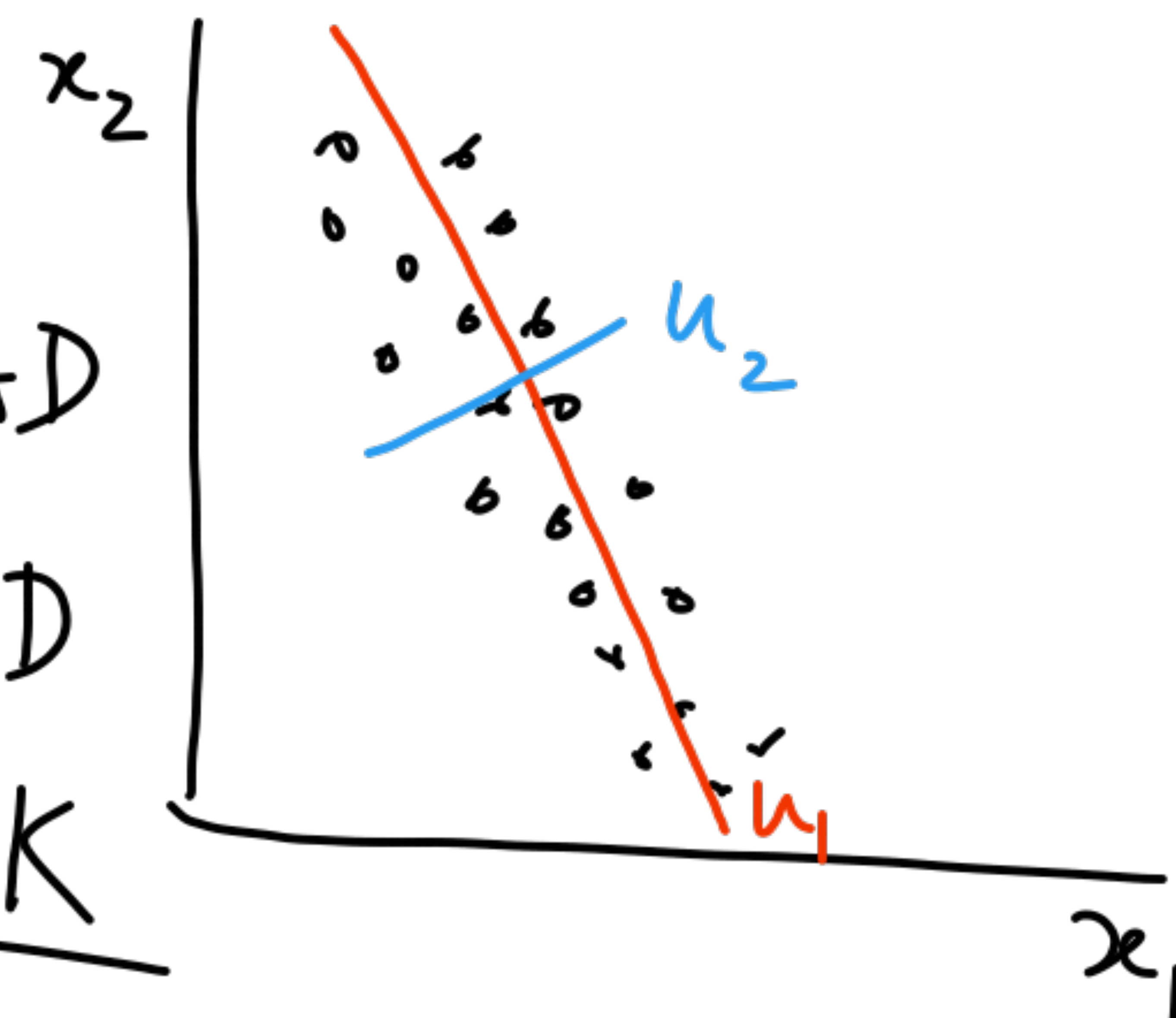
What does PCA do?

Project points to x_1? BAD

" " " x_2? BAD

" " " u_1? OK

" " " u_2? BAD



PCA: finds the dimensions of maximal variance and projects points onto these dimensions

Goal of dimensionality reduction:

Given a point $x \in \mathbb{R}^d$, find a projection matrix $U \in \mathbb{R}^{d \times k}$, where $k < d$, to project the d-dimensional x to a k-dimensional

$$z = U^T x.$$

How do we find U?

PCA finds U via two objectives (which are equivalent!)

Objective 1:

$$z = U^T x \quad (\text{ENCODE step})$$

$$\hat{x} = UU^T x \quad (\text{DECODE step})$$

Objective of PCA is to minimize the reconstruction error between $\hat{x}$ and x

$$\min_{U,\, U^T U = I} \frac{1}{n} \sum_i \| x_i - \hat{x}_i \|^2 = \min_{U,\, U^T U = I} \sum_i \| x_i - UU^T x_i \|^2 \rightarrow A$$

Objective 2

find U s.t. the projected points have maximal variance

$$\max_{U,\, U^T U = I} E\left[\|U^T x\|^2\right] - \left(E\left[U^T x\right]\right)^2$$

Consider x's to be mean-centered i.e., $E[x] = 0$

$$\max_{U,\, U^T U = I} E\left[\|U^T x\|^2\right]$$

B

Both objectives (A) and (B) are equivalent!

Proof:

$$x = UU^T x + (I - UU^T)x$$

$$\|x\|^2 = \|UU^T x + (x - UU^T x)\|^2$$

$$= \|UU^T x\|^2 + \|(x - UU^T x)\|^2$$

$$\cancel{+2(UU^T x)^T(x - UU^T x)} \quad \because U^T U = I$$

$$E\left[\|x\|^2\right] = E\left[\|UU^T x\|^2\right] + E\left[\|(x - UU^T x)\|^2\right]$$

$$E\left[\|x\|^2\right] = E\left[\|UU^Tx\|^2\right] + E\left[\|x - UU^Tx\|^2\right]$$

$$\underbrace{E\left[\|x\|^2\right]}_{\text{VARIANCE of the data}} = \underbrace{E\left[\|U^Tx\|^2\right]}_{\text{Variance of the projected data}} + \underbrace{E\left[\|x - UU^Tx\|^2\right]}_{\text{reconstruction error}}$$

$\because$ rotation by U doesn't change the norm

Given that the variance of the data is fixed (given the data), minimizing reconstruction error is equivalent to maximizing the projected data's variance

How do we find U?

Consider $k=1$; the optimization problem for PCA becomes

$$\max_{\|U\|=1} \frac{1}{n}\sum_i {U^T x_i}^2 = \max_{U^T U=1} \frac{1}{n} \|X^T u\|^2 \text{ where}$$

$$X = \begin{bmatrix} \uparrow & & \uparrow \\ x_1 & \cdots & x_n \\ \downarrow & & \downarrow \end{bmatrix}_{d\times n}$$

$$= \max_{U^T U=1} \frac{1}{n} u^T X X^T u$$

$$= \max_{U^T U=1} u^T \left(\frac{1}{n} X X^T\right) u$$

X is mean-centred

$$= \boxed{\max_{U^T U=1} u^T S u}$$ where S is the covariance matrix

$\longrightarrow$ C

Using convex optimization to solve (c), we get the optimal u that maximizes $u^T S u$ is the eigenvector of S with the largest eigenvalue

This is the first principal component.

To find k dimensions of maximal variance, pick the top-k eigenvectors with the highest eigenvalues

PCA Algorithm

1. Mean-centre the data to get $X = \begin{bmatrix} \uparrow & & \uparrow \\ x_1 & \cdots & x_n \\ \downarrow & & \downarrow \end{bmatrix}_{d \times n}$

2. Compute the covariance matrix $S = \frac{1}{n} X X^T$

3. Find the top-k eigenvectors of S, in descending order of eigenvalues.

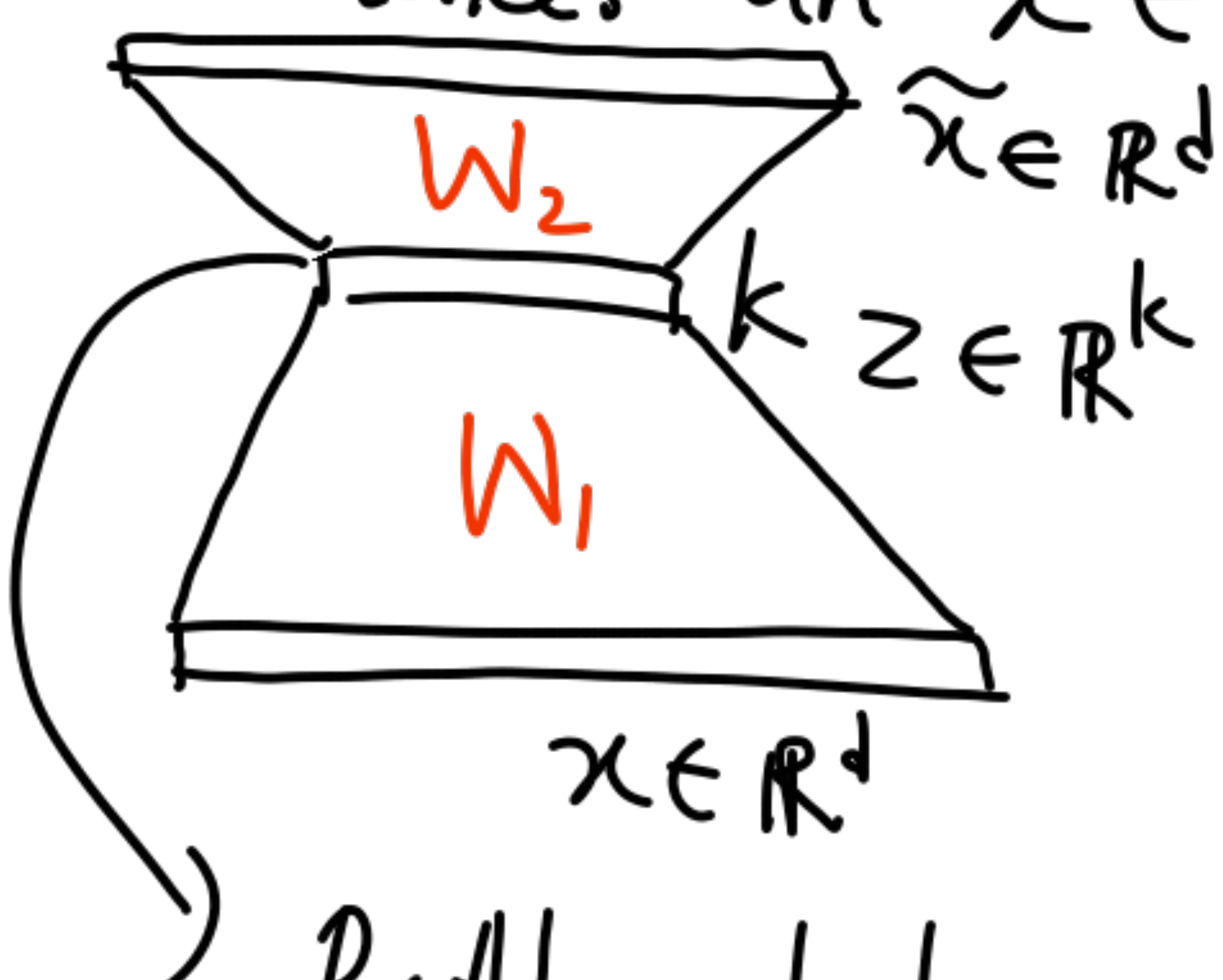
   chosen

   Say these are $u_1, \ldots, u_k$

4. Projection matrix $U = \begin{bmatrix} \uparrow & & \uparrow \\ u_1 & \cdots & u_k \\ \downarrow & & \downarrow \end{bmatrix}_{d \times k}$

Connection between PCA and neural networks (with linear activations)

An Autoencoder is a type of feedforward NN that takes an $x \in \mathbb{R}^d$ as input and aims to reconstruct x



Bottleneck layer

Autoencoder with a single hidden layer of dimensionality k and linear activation functions.

$\tilde{x} = W_2 W_1 x$ // optimal weights for W_1 would be the top-k principal components

if the goal is to minimize a mean-squared loss function