

Feature Learning

Alex(ander) Jung
Assistant Professor for Machine Learning
Department of Computer Science
Aalto University

Reading.

Ch. 9 of AJ, "Machine Learning: The Basics,"
Springer, Singapore, 2022.

preprint: <https://mlbook.cs.aalto.fi>



https://scikit-learn.org/stable/modules/feature_selection.html

Learning Goals

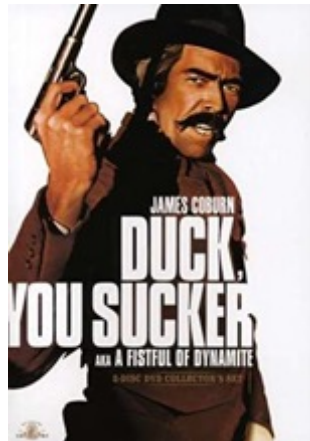
- understand challenges with long raw feature vectors
- basic idea of feature learning
- feature learning for visualization and privacy-protection
- principal component analysis
- random projections

Data Point = “Some Movie”

several **Gigabytes** of raw
feature bits!

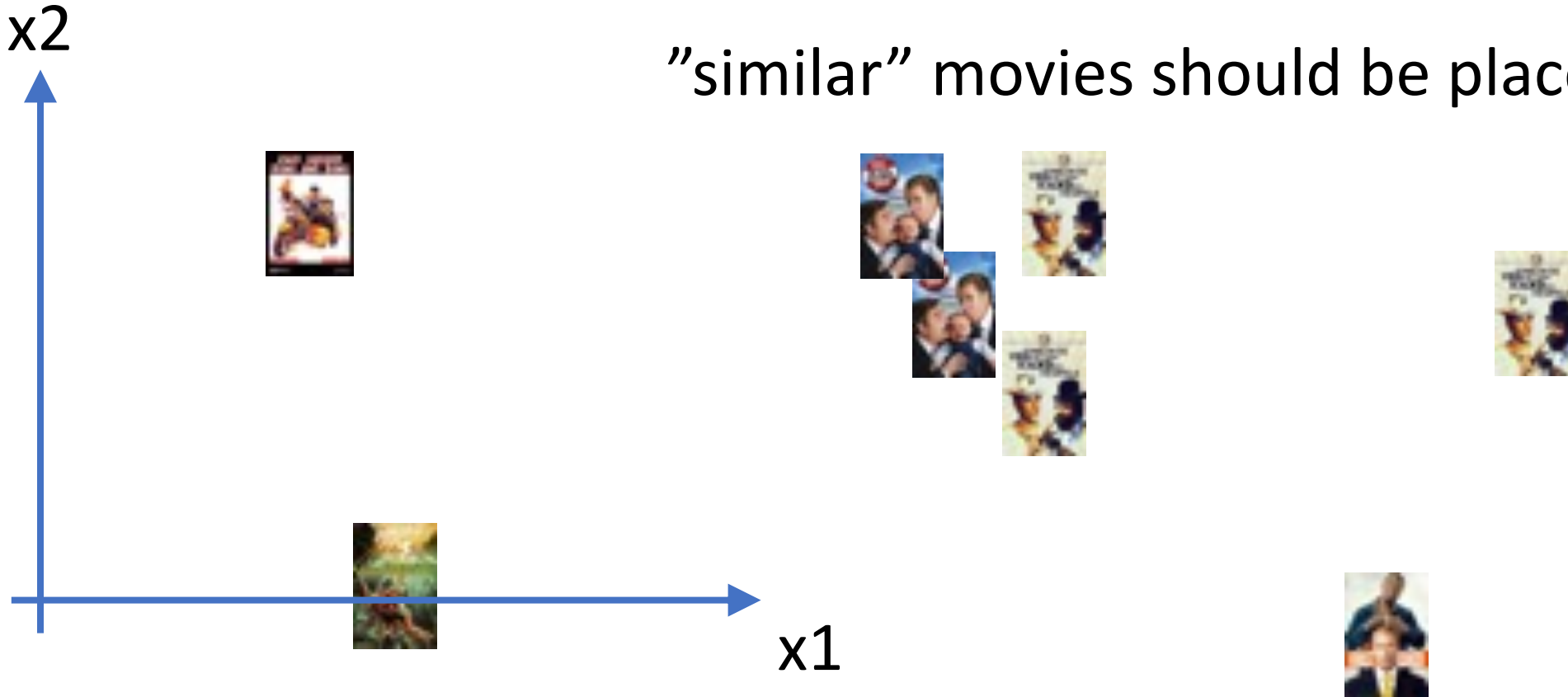


Dataset = "Bunch of Movies"



Scatterplot of Movies?

"similar" movies should be placed nearby!



movie represented by two features x_1, x_2 !

Curse of Big Data

Overwhelmed by Tons of Features!

- consider data point representing a person
- digital footprint can be used for constructing features
- health-records (including genetic fingerprint)
- credit-card transactions
- social media posts
- media collections
- travelling profile over last 20 years
-



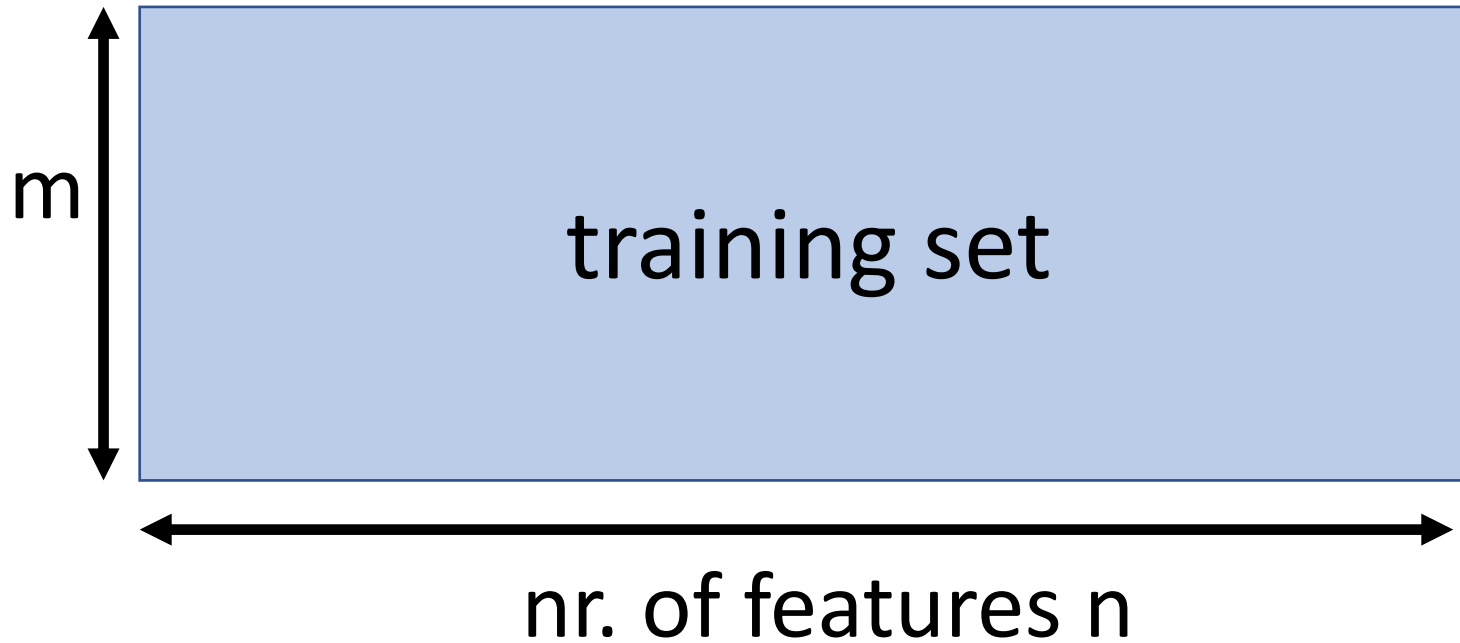
Statistical Challenge

- effective dimension d of hypothesis space
- d typically increases with larger nr. of features
- overfitting is likely when $d > \text{sample size } m$

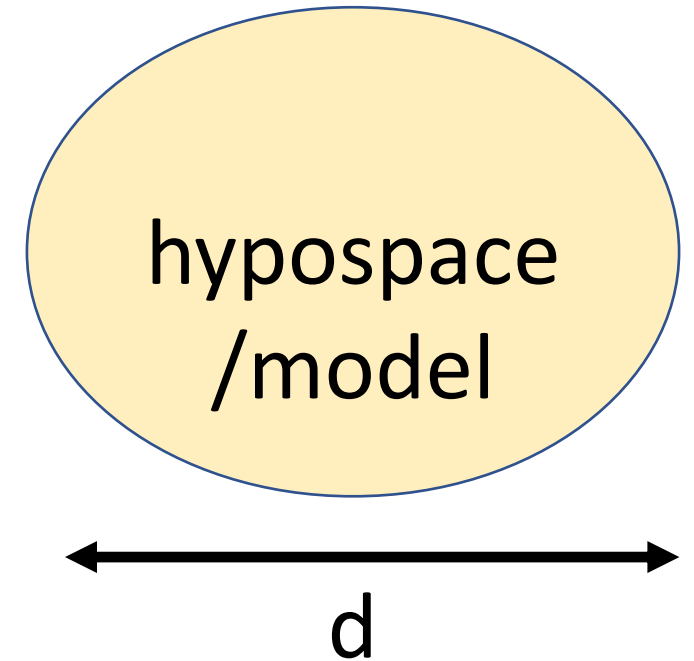
Linear Regression

- consider data points with n features
- have m labeled data points
- for $m < n$, plain linear regression will overfit
- n might be billions (e.g., HD-movies)
- would need billions of labeled data points

Data and Model Size



overfitting as soon as
 $d/m > 1$!



Computational Challenge

linear regression on data points with d features
amounts to inverting a matrix size $d \times d$

GDPR-Compliant Feature Selection

Data minimisation: The use of personal data has to be limited to what is necessary to fulfil the purpose it was collected for ...

Proportionality... The amount and nature of the data used has to be proportionate to the purpose and the least invasive for the data subject...

source: <https://www.auditingalgorithms.net/>

Feature Selection for Trustworthy AI

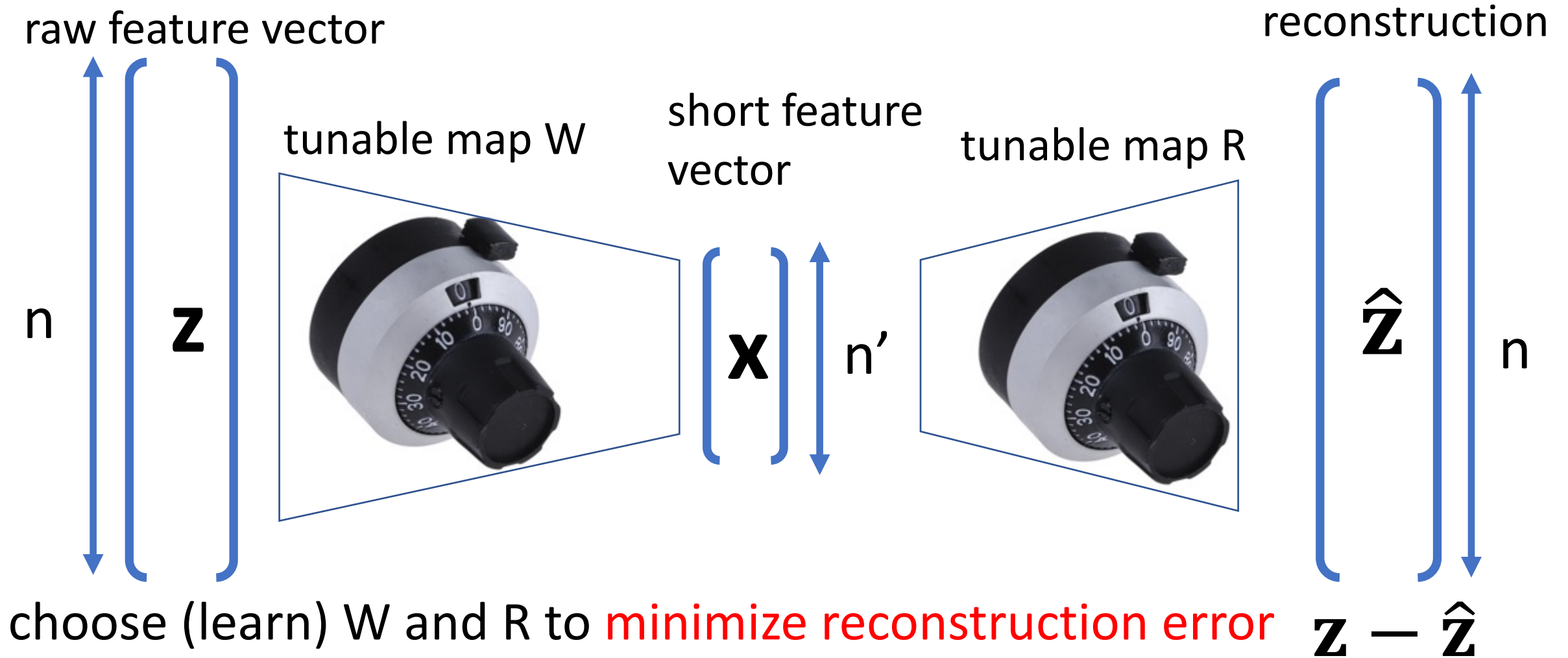
“**Privacy and data governance:** besides ensuring full respect for privacy and data protection, ...into account the quality, integrity ...and ensuring legitimised access to data...

Diversity, non-discrimination and fairness: Unfair bias must be avoided, as it could have multiple negative implications, from the marginalization of vulnerable groups, to the exacerbation of prejudice and discrimination....”

<https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai>

The Basic Idea of Feature Learning.

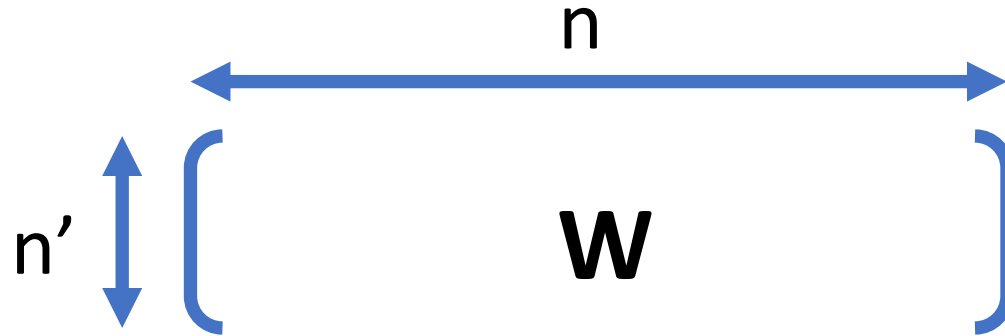
Basic Idea of Feature Learning



Linear Feature Learning

- use linear maps for compression and reconstruction

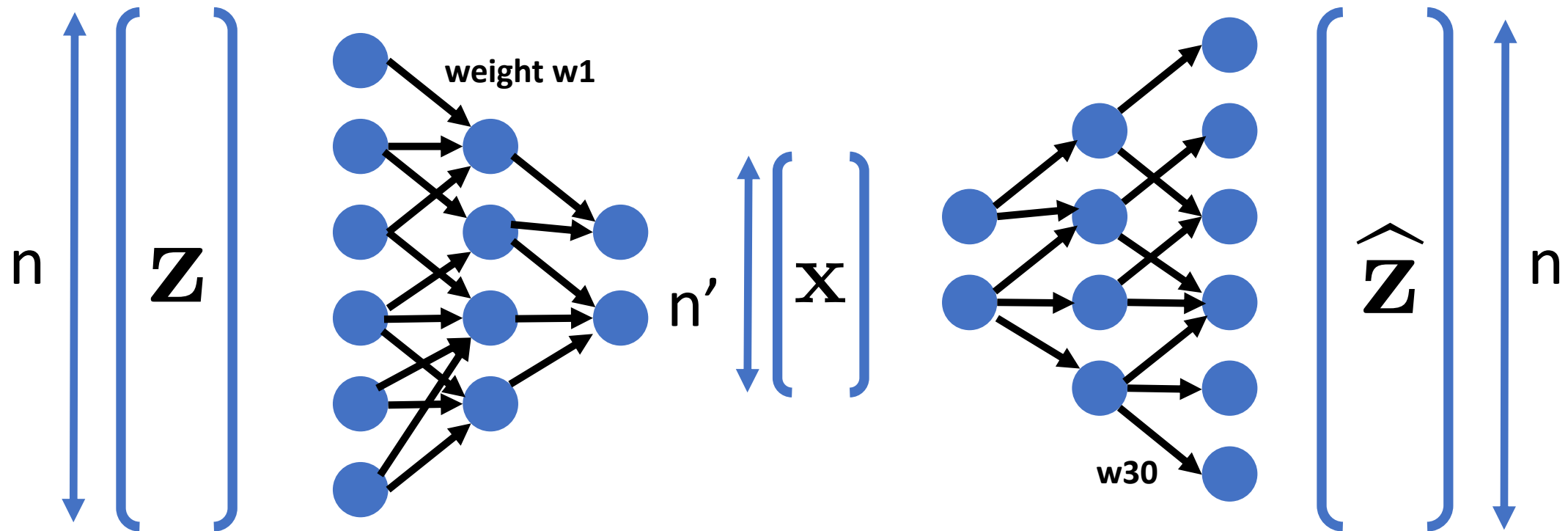
$$\mathbf{x} = \mathbf{W}\mathbf{z} \quad \hat{\mathbf{z}} = \mathbf{R}\mathbf{x}$$



choose matrices \mathbf{W} and \mathbf{R} to minimize $\mathbf{z} - \hat{\mathbf{z}} = (\mathbf{I} - \mathbf{R}\mathbf{W})\mathbf{z}$

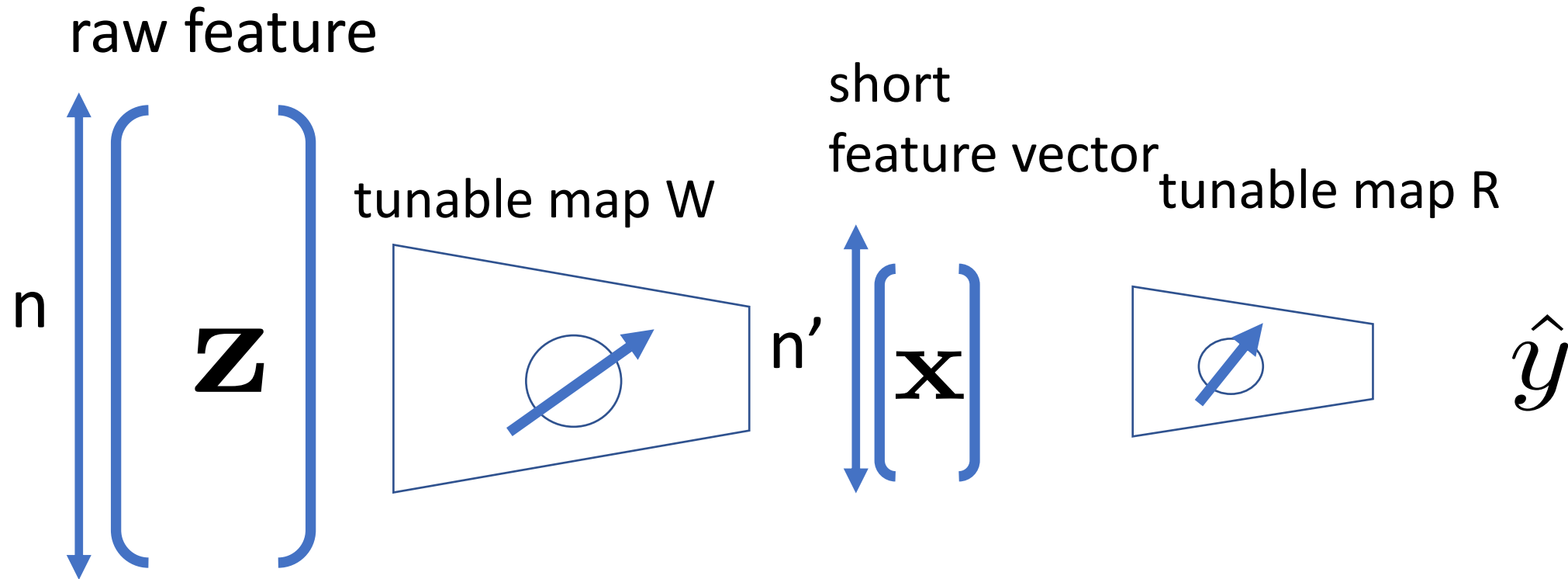
Non-Linear Feature Learning (“Autoencoder”)

use artificial neural networks for compression and reconstruction



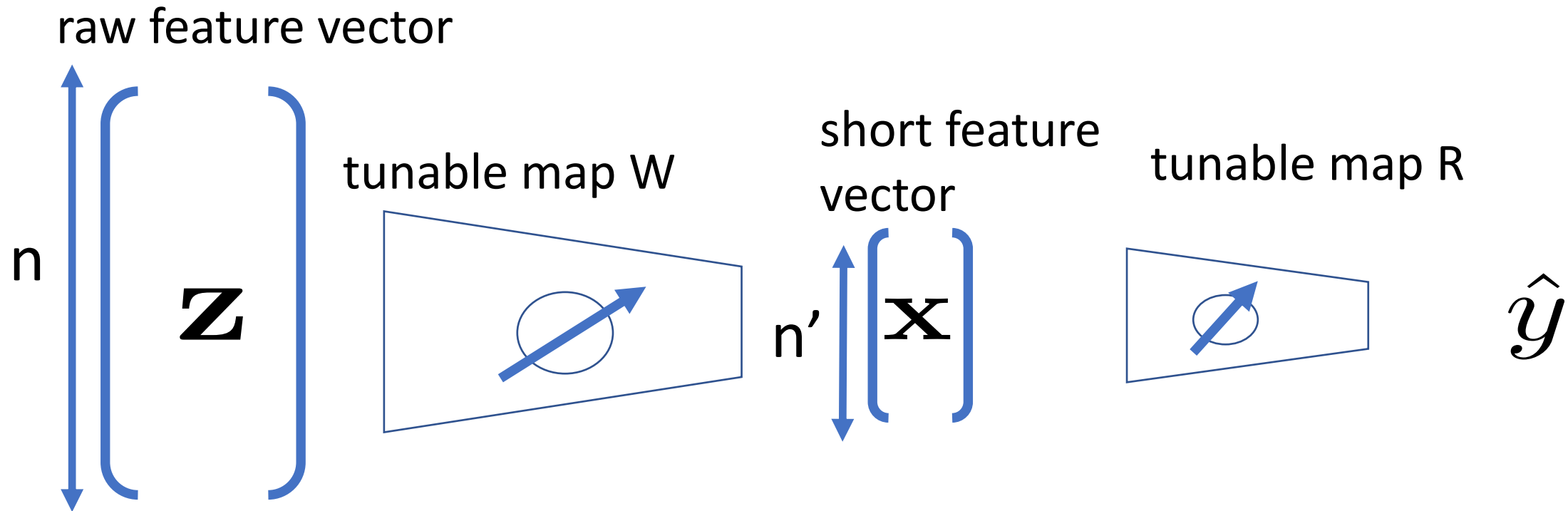
much like linear maps, ANNs are just **parametrized maps!**

Feature Learning for Labeled Data



choose W such that we can **predict** (using some map R) **the label y** from z with **maximum accuracy**

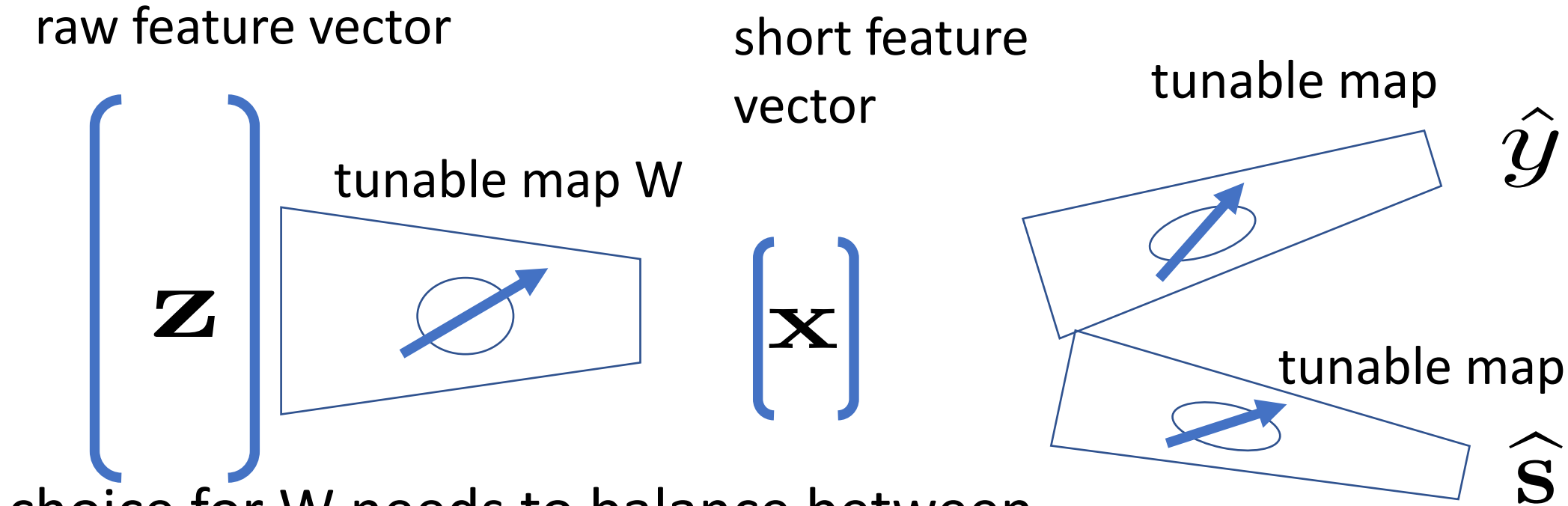
Feature Learning for Labeled Data



choice for W needs to **balance between**

- **compressing** raw feature vector as much as possible
- **keep parts** of raw features that are **relevant** for predicting y

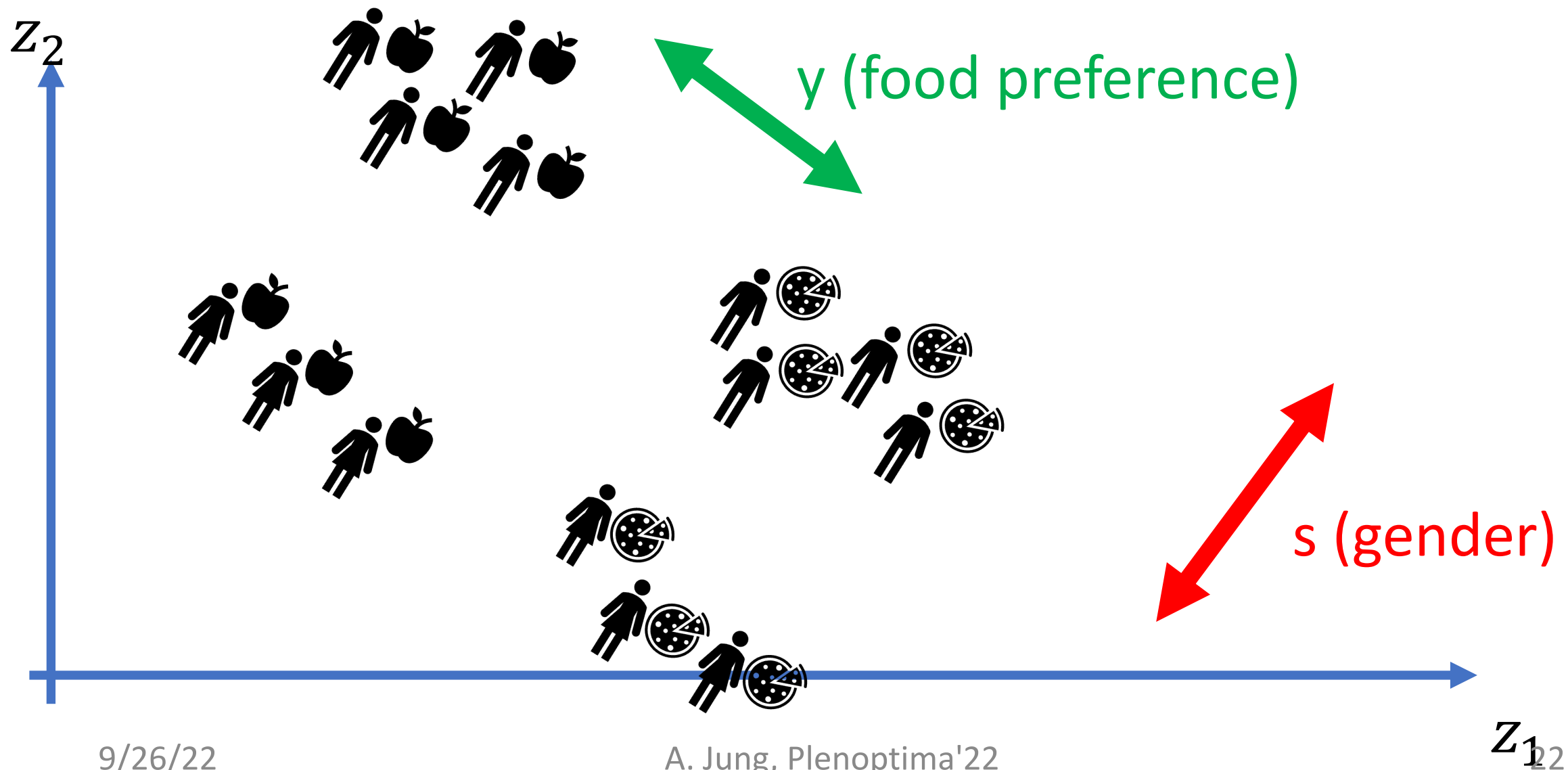
Privacy-Preserving Feature Learning



choice for W needs to balance between

- **compressing** raw feature vector as much as possible
- keep parts of raw features that are **relevant** for predicting y
- predicting **private variable** "s" is **not predictable** from x

Privacy-Preserving Feature Learning

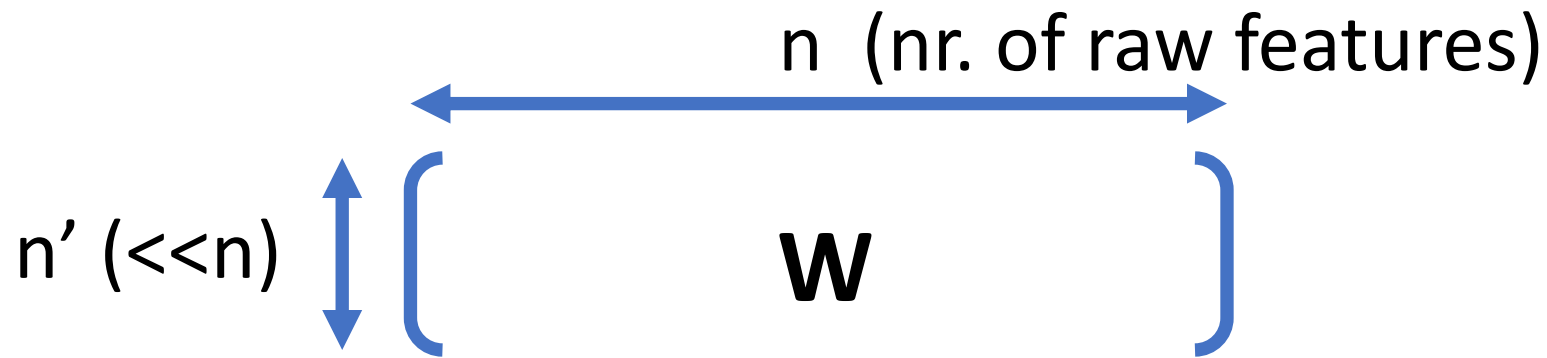


Principal Component Analysis (PCA)

Linear Feature Learning

- use linear maps for compression and reconstruction

$$\mathbf{x} = \mathbf{W}\mathbf{z} \quad \hat{\mathbf{z}} = \mathbf{R}\mathbf{x}$$



choose matrices \mathbf{W} and \mathbf{R} to minimize $\mathbf{z} - \hat{\mathbf{z}} = (\mathbf{I} - \mathbf{R}\mathbf{W})\mathbf{z}$

PCA as Risk Minimization

- m datapoints with raw feature vecs $\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}$

choose matrices \mathbf{W}, \mathbf{R} to minimize reconstruction error

$$L(\mathbf{W}, \mathbf{R}) = \frac{1}{m} \sum_{i=1}^m \left\| \mathbf{z}^{(i)} - \mathbf{R}\mathbf{W}\mathbf{z}^{(i)} \right\|^2$$

Principal Component Analysis (PCA)

- optimal **compression matrix** $\mathbf{W} = (\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(n')})$
- using “top” eigenvectors $\mathbf{u}^{(i)}$ **of sample covariance matrix**

$$\hat{\mathbf{C}} = (1/m) \sum_{i=1}^m \mathbf{z}^{(i)} (\mathbf{z}^{(i)})^T$$

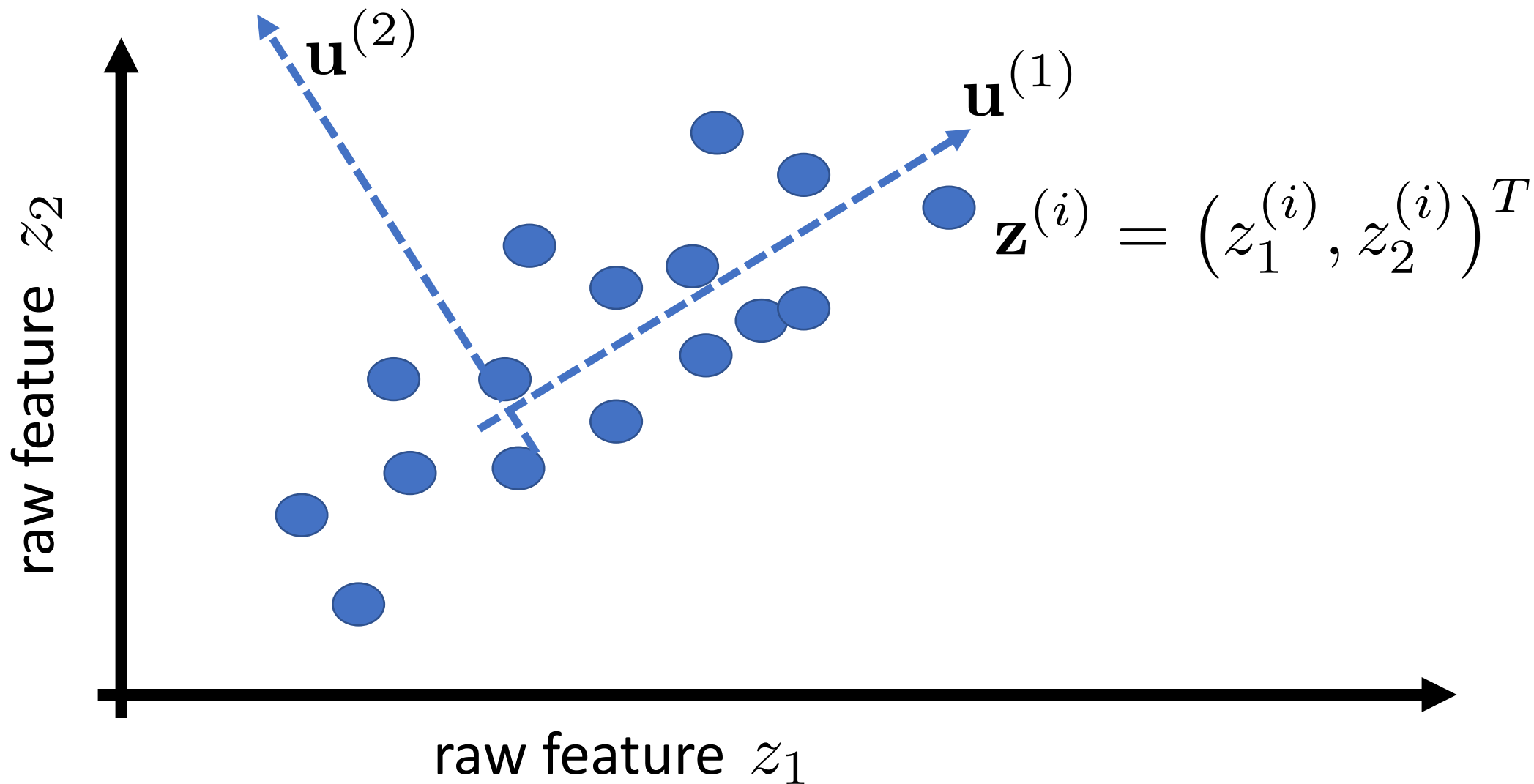
- **eigenvalue decomposition** of psd sample cov. matrix:

$$\hat{\mathbf{C}} = (\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(n)}) \text{diag}(\lambda_1, \dots, \lambda_n) (\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(n)})^T$$

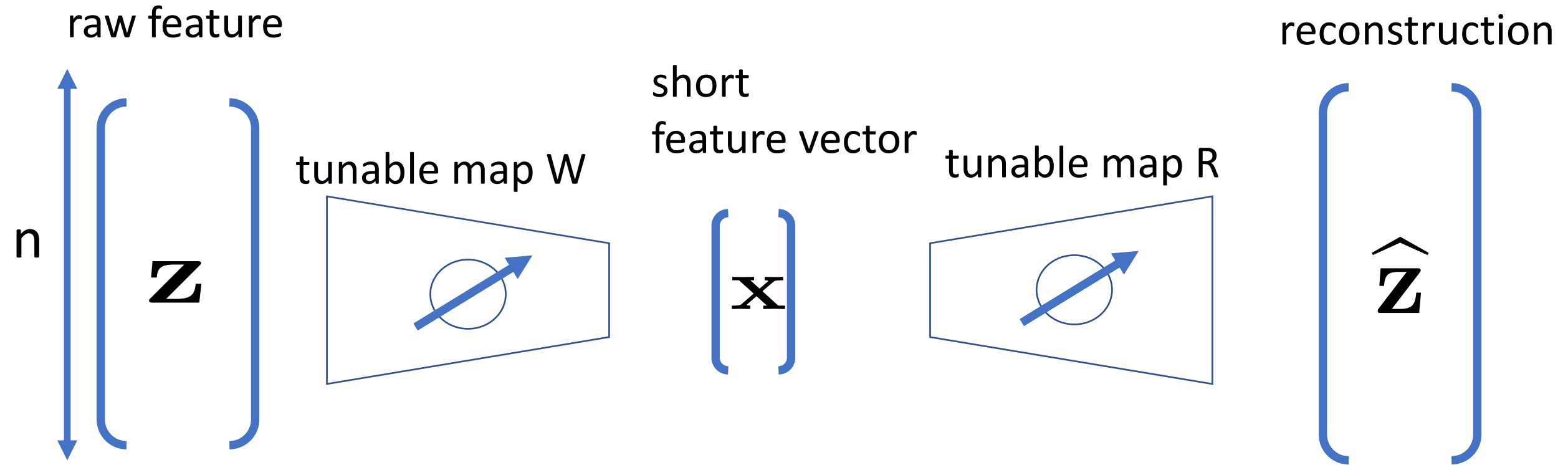
non-negative eigenvalues

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \geq 0.$$

Geometry of PCA



Computational Complexity



choose (learn) W and R to **minimize reconstruction error** $\mathbf{Z} - \hat{\mathbf{Z}}$

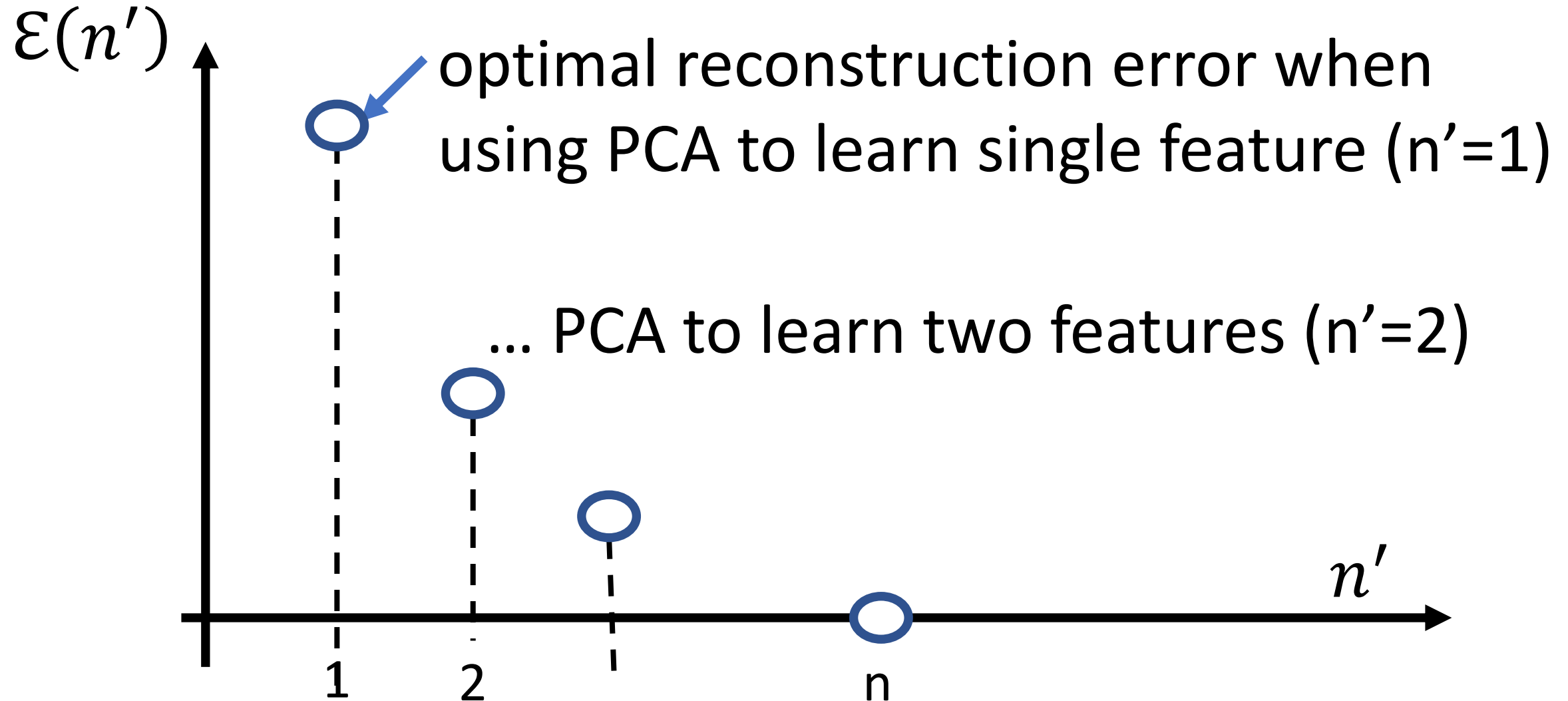
PCA requires eigenvalue decomposition of “ $n \times n$ ” matrix!

How to choose n' ?

- PCA requires number n' of learned features as input
- $n'=2$ for visualization (scatter plot)
- chose n' to balance compression with optimal reconstruction error

$$L(n') = \min_{\substack{\mathbf{W} \in \mathbb{R}^{n' \times n} \\ \mathbf{R} \in \mathbb{R}^{n \times n'}}} L(\mathbf{W}, \mathbf{R})$$

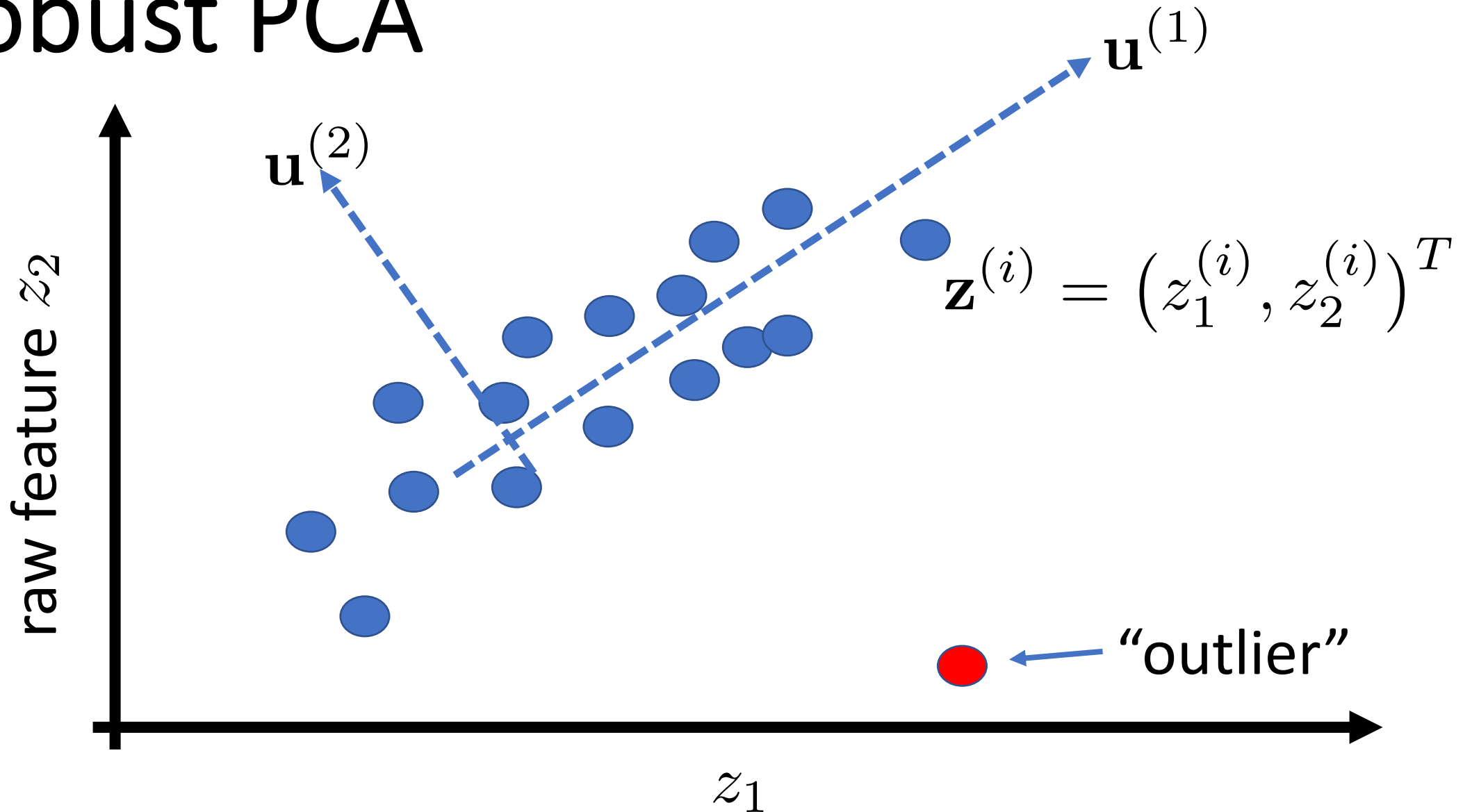
Elbow Method



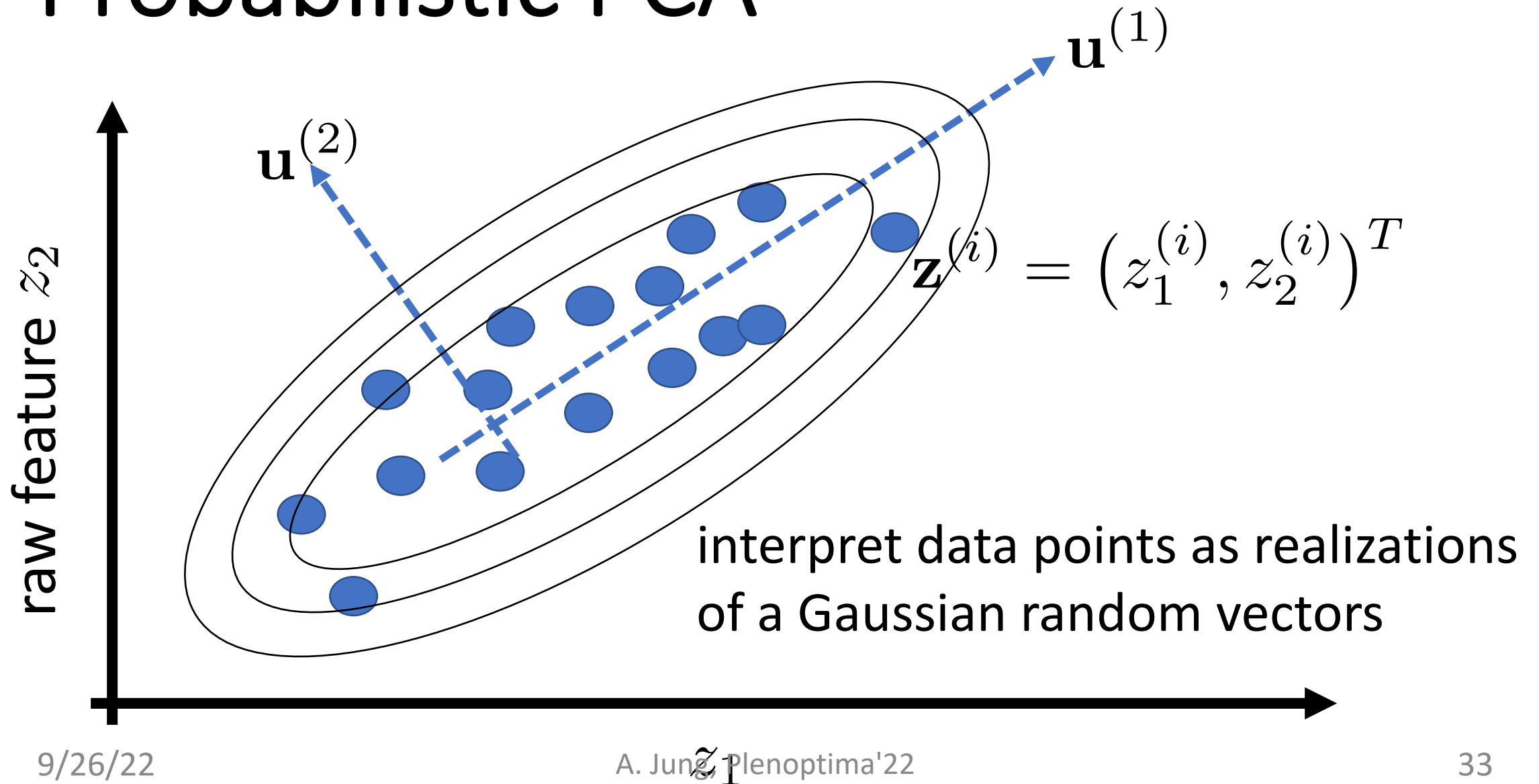
Variations of PCA

- **robust PCA:** uses a different measure of reconstruction error
- **probabilistic PCA:** uses a statistical model for data points
- **sparse PCA:** new features depend on few raw features

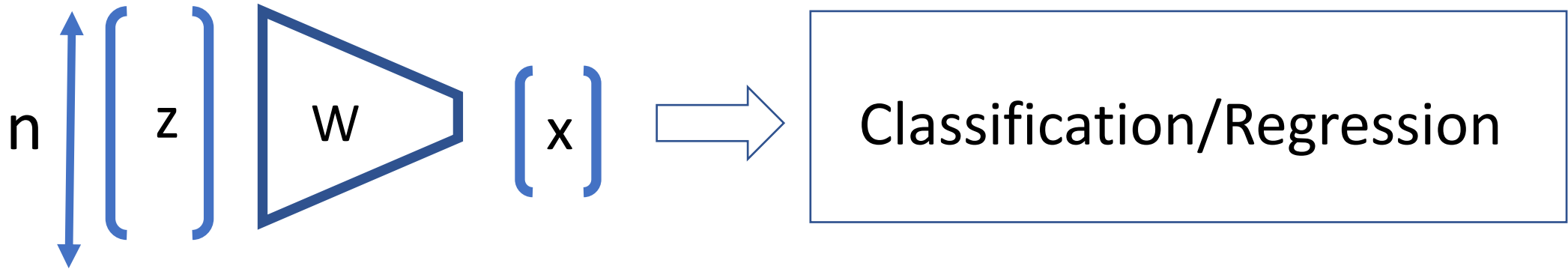
Robust PCA



Probabilistic PCA

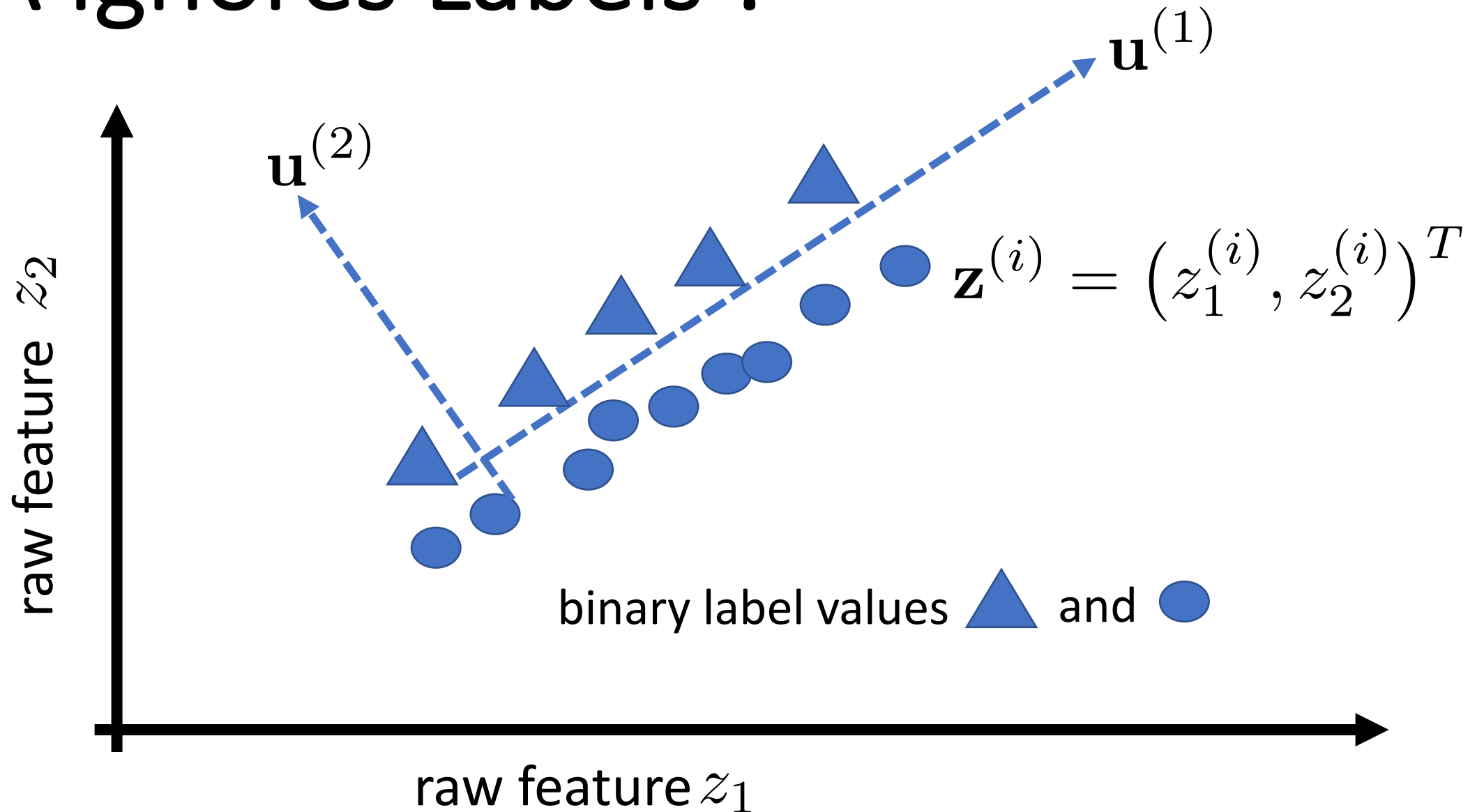


PCA as Pre-Processing

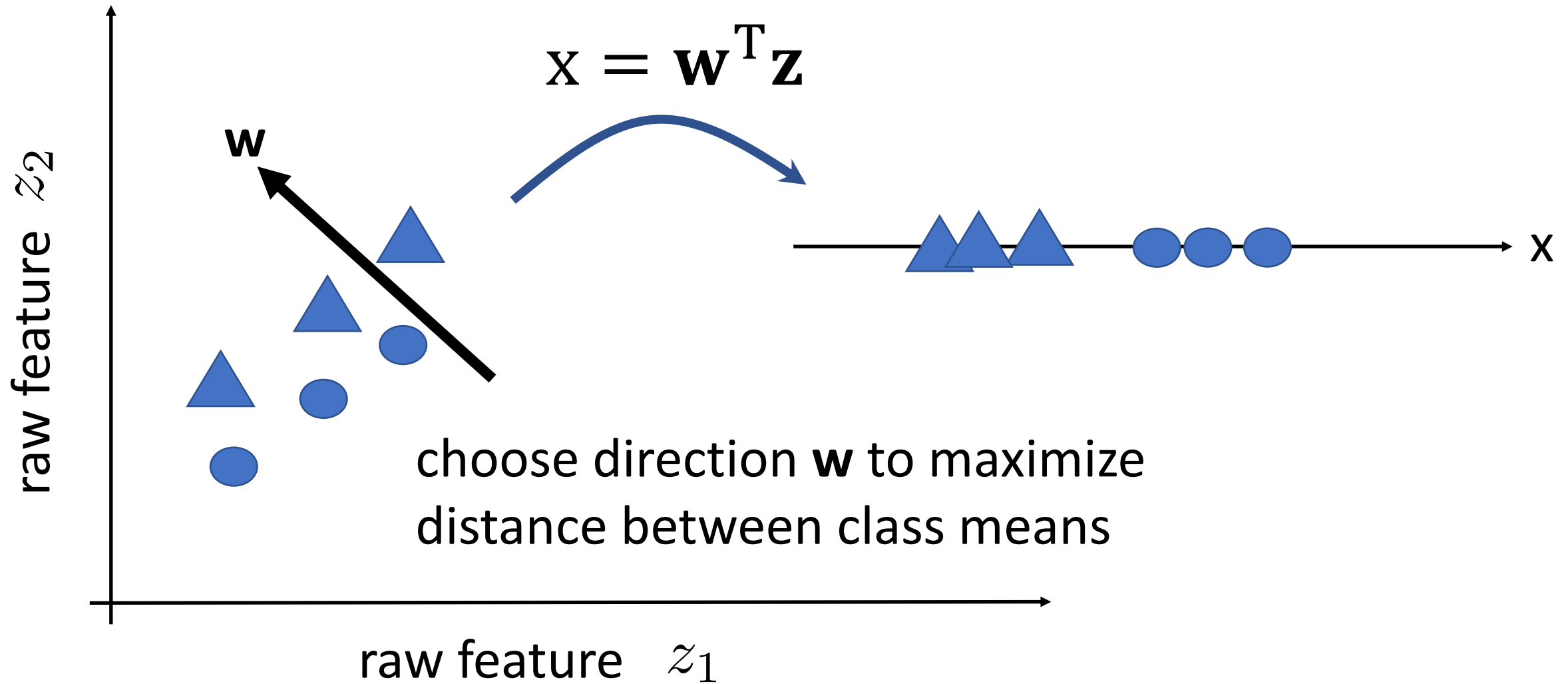


- PCA delivers a compression matrix W
- replace (long) raw features z with shorter features $x = Wz$
- apply regression/classification methods to new features x
- CAUTION: PCA ignores label information!

PCA Ignores Labels !



Fisher's Linear Discriminant



Random Projections

- consider **random projection** $\mathbf{x} = \mathbf{W} \mathbf{z}$
- entries of matrix \mathbf{W} are **randomly chosen**
- **no learning/tuning** of \mathbf{W} required!
- in many settings, **works surprisingly well**
- known as **compressed sensing**

So What?

- feature learning methods determine **relevant features**
- learning **two features** allows to **scatter plot** !
- optimal **linear** feature learning = **PCA**
- PCA ignores label information!
- **random projections** as computationally light alternative

Thank You !