PCA Review, Probabilistic PCA, GPLVMs

Week 6b - Statistical ML / Thang Bui / ANU / 2025 S1

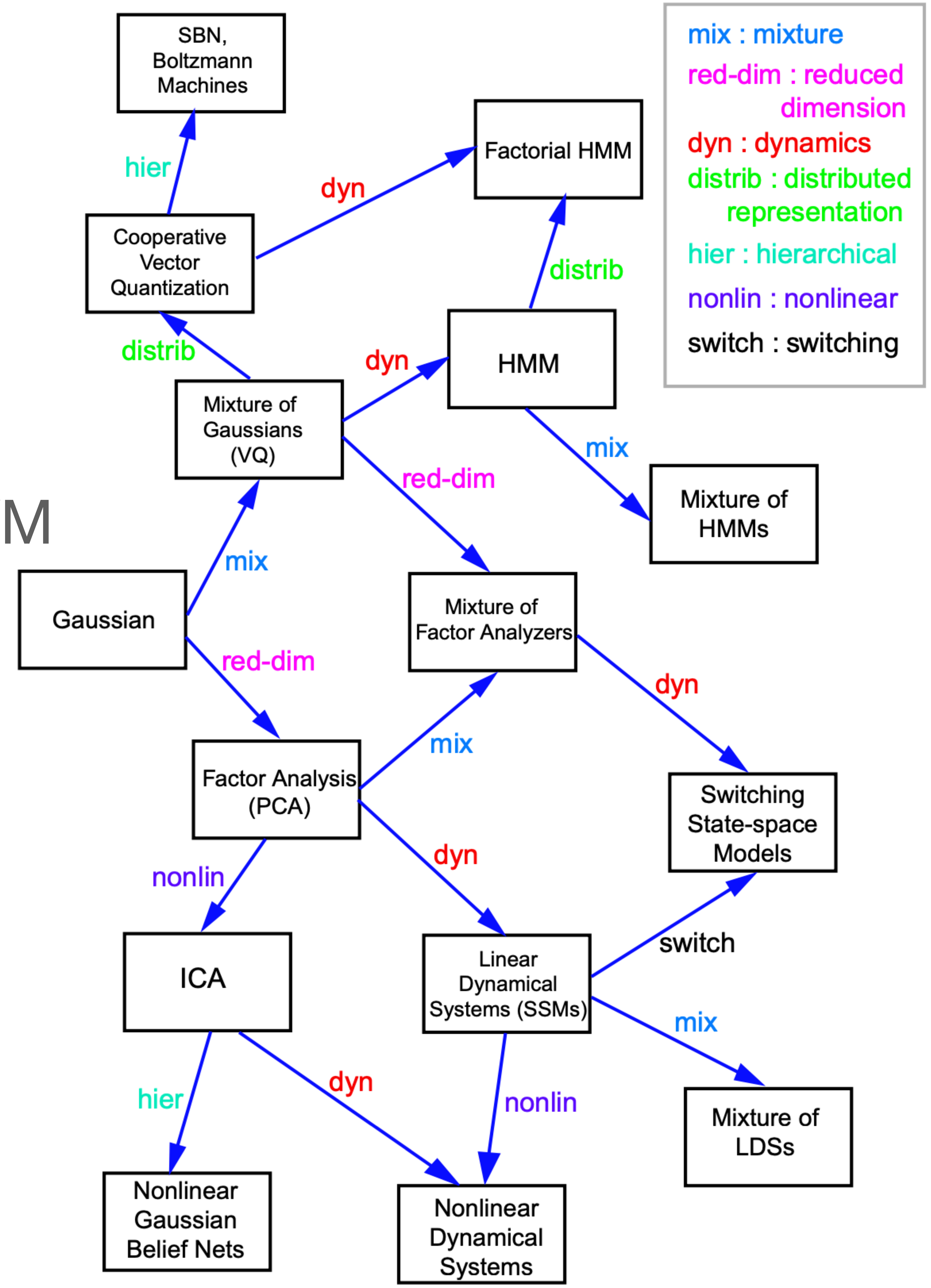
Housekeeping

+ Quiz 1 results are available on Wattle, 9.5/10 average!
+ Class survey is running. Also feedback [direct (emails) or indirect (class reps)] are appreciated.
+ Assignment 1 due this Friday [midnight]. 5% penalty for 5 mins - 24 hrs late, 100% penalty after.
+ I will be on leave the following two weeks so emails will be slow
+ Prof Jing Jiang will take over after the break.

Overview

1. Motivation
2. PCA review
3. Linear, Gaussian latent variable models and GPLVM

Reading: Bishop 12.1, 12.2, 12.4.2



Motivation

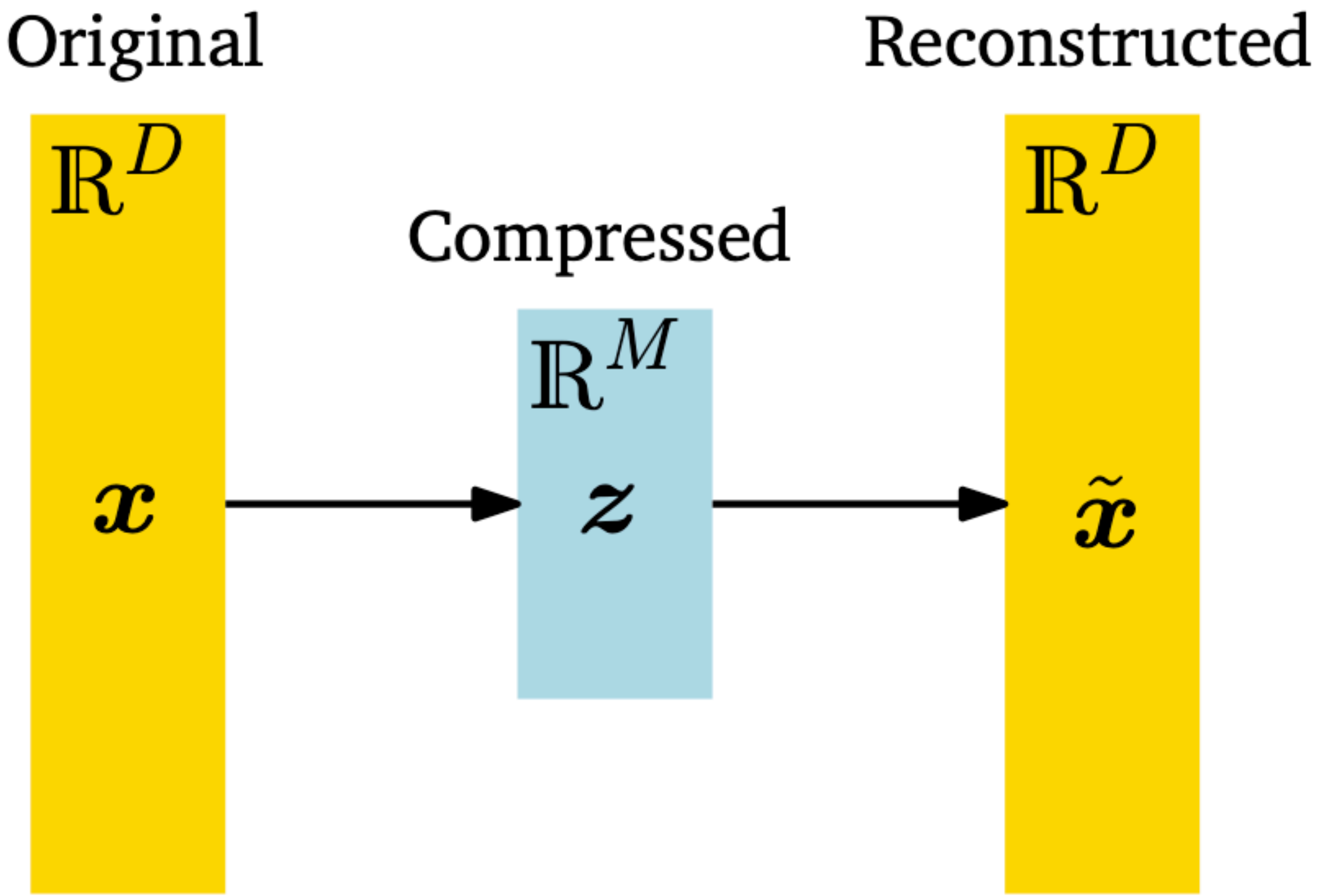
Motivation

Dimensionality reduction as data compression

Find lower-dimensional data without losing much information

M < D

z captures desirable variations in **x**
Reconstructed data is similar to **x**



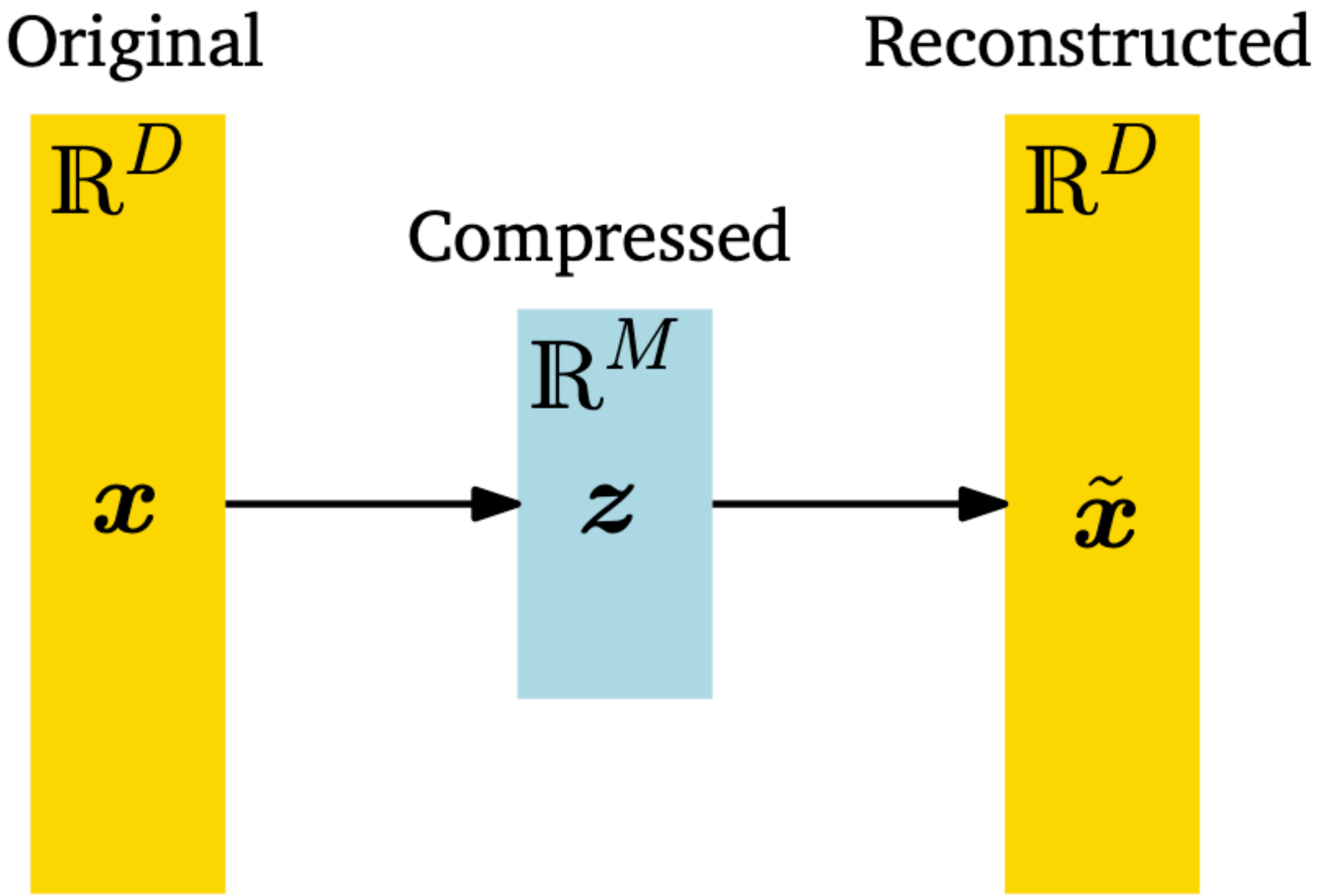
Motivation

Dimensionality reduction as data compression

Find lower-dimensional data without losing much information

M < D

z captures desirable variations in **x**
Reconstructed data is similar to **x**



Why?

+ Data may have low *intrinsic* dimensionality [think about data living on a line in high dimensions]
+ visualisation / exploratory data analysis [e.g. compress 100-D data down to 2D to visualise patterns]
+ Using low dimensional data for learning [e.g. train a classifier using compressed data]

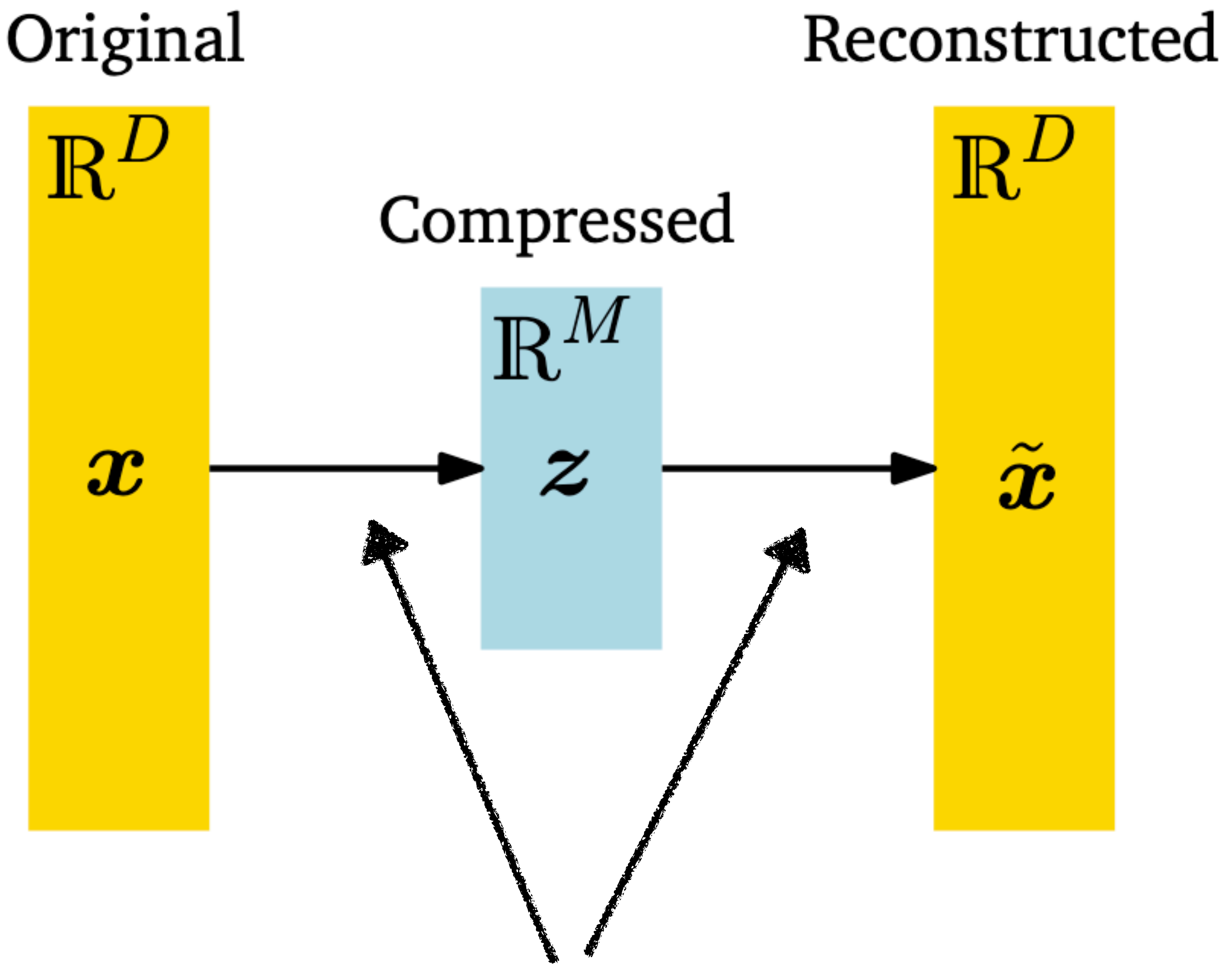
Motivation

Dimensionality reduction as data compression

Find lower-dimensional data without losing much information

M < D

z captures desirable variations in **x**
Reconstructed data is similar to **x**



Key question: how to construct these mappings?

Why?

+ Data may have low *intrinsic* dimensionality [think about data living on a line in high dimensions]

+ visualisation / exploratory data analysis [e.g. compress 100-D data down to 2D to visualise patterns]

+ Using low dimensional data for learning [e.g. train a classifier using compressed data]

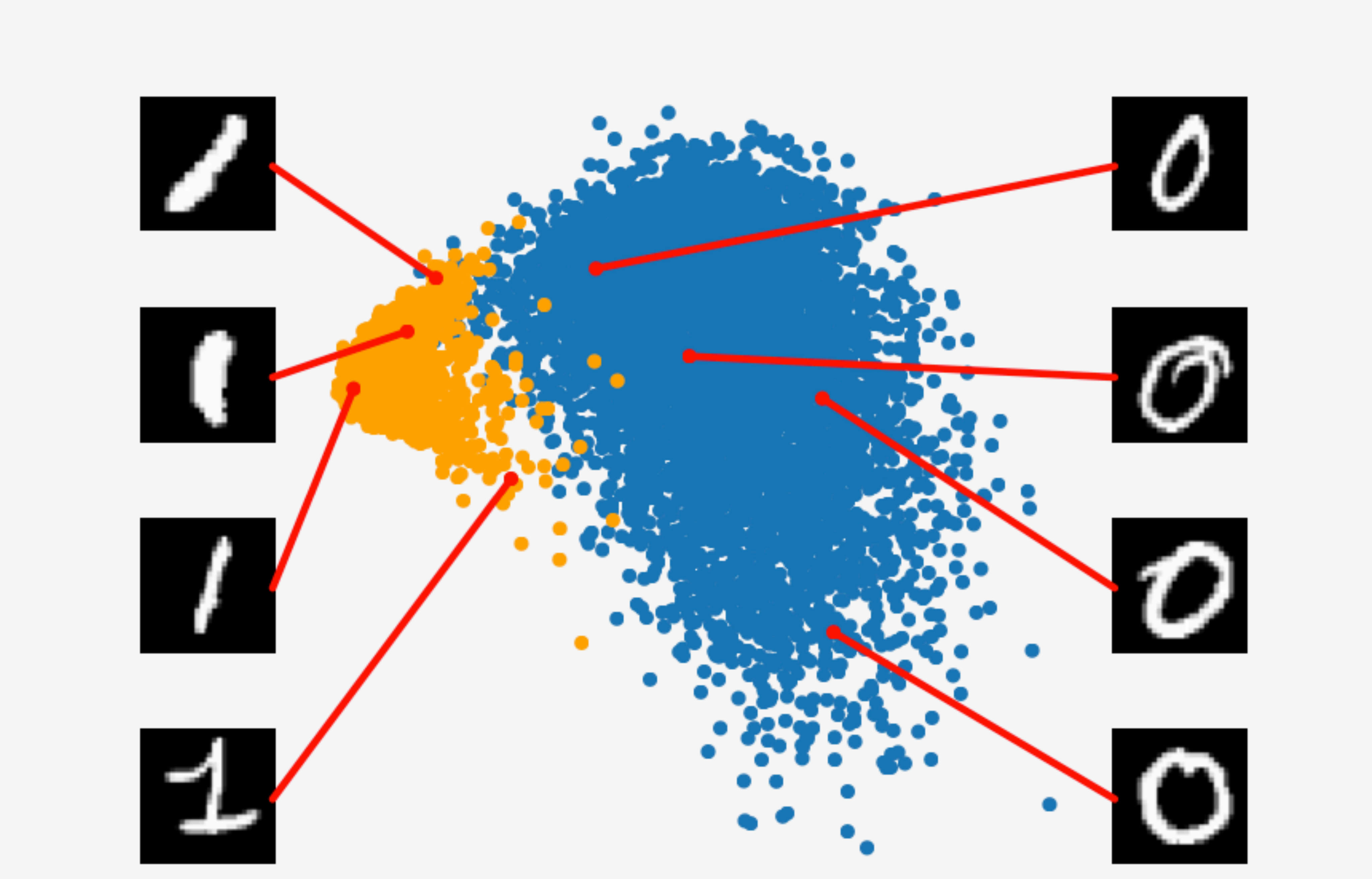
Motivation - example

Dimensionality reduction as data compression

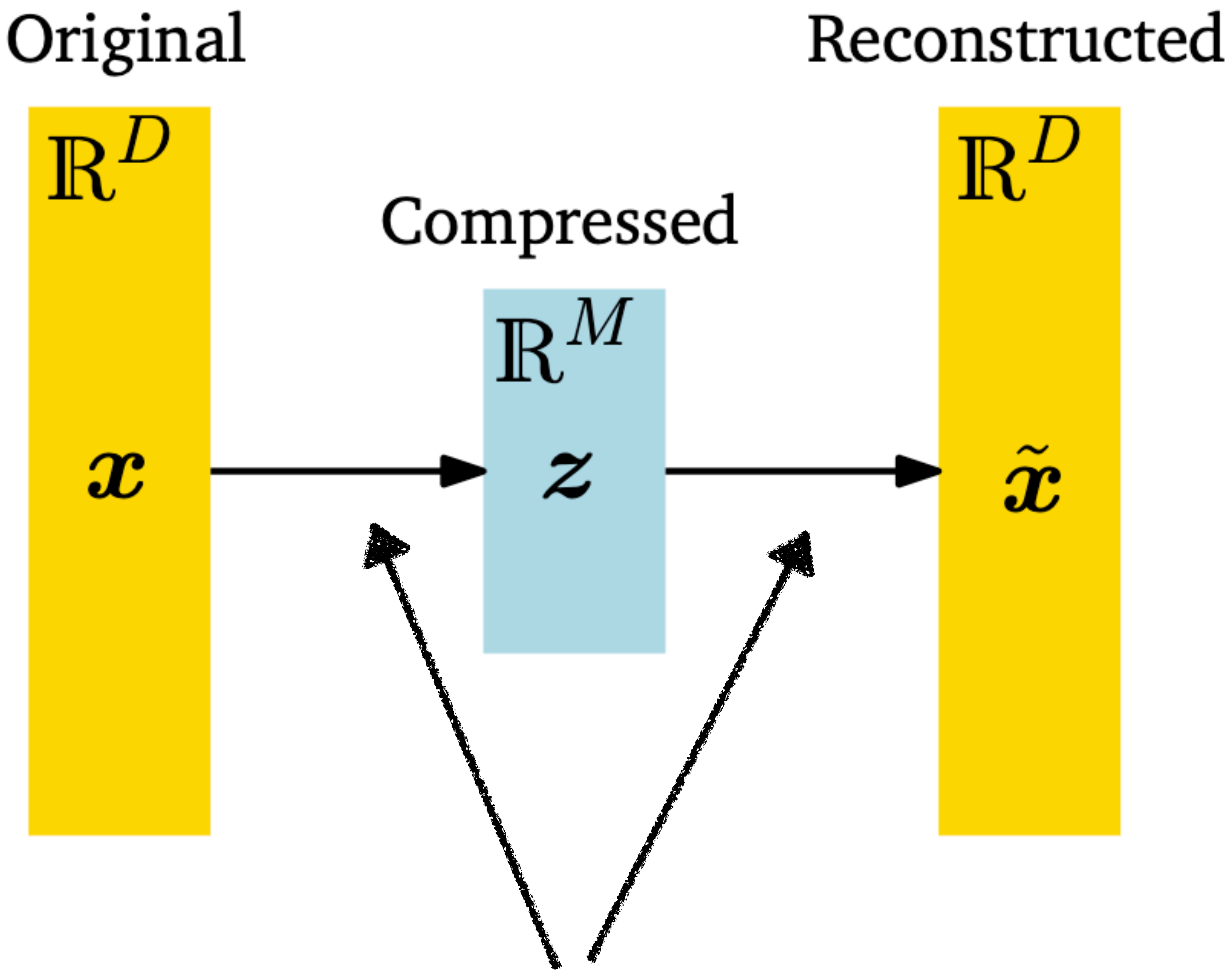
Find lower-dimensional data without losing much information

M < D

z captures desirable variations in **x**
Reconstructed data is similar to **x**



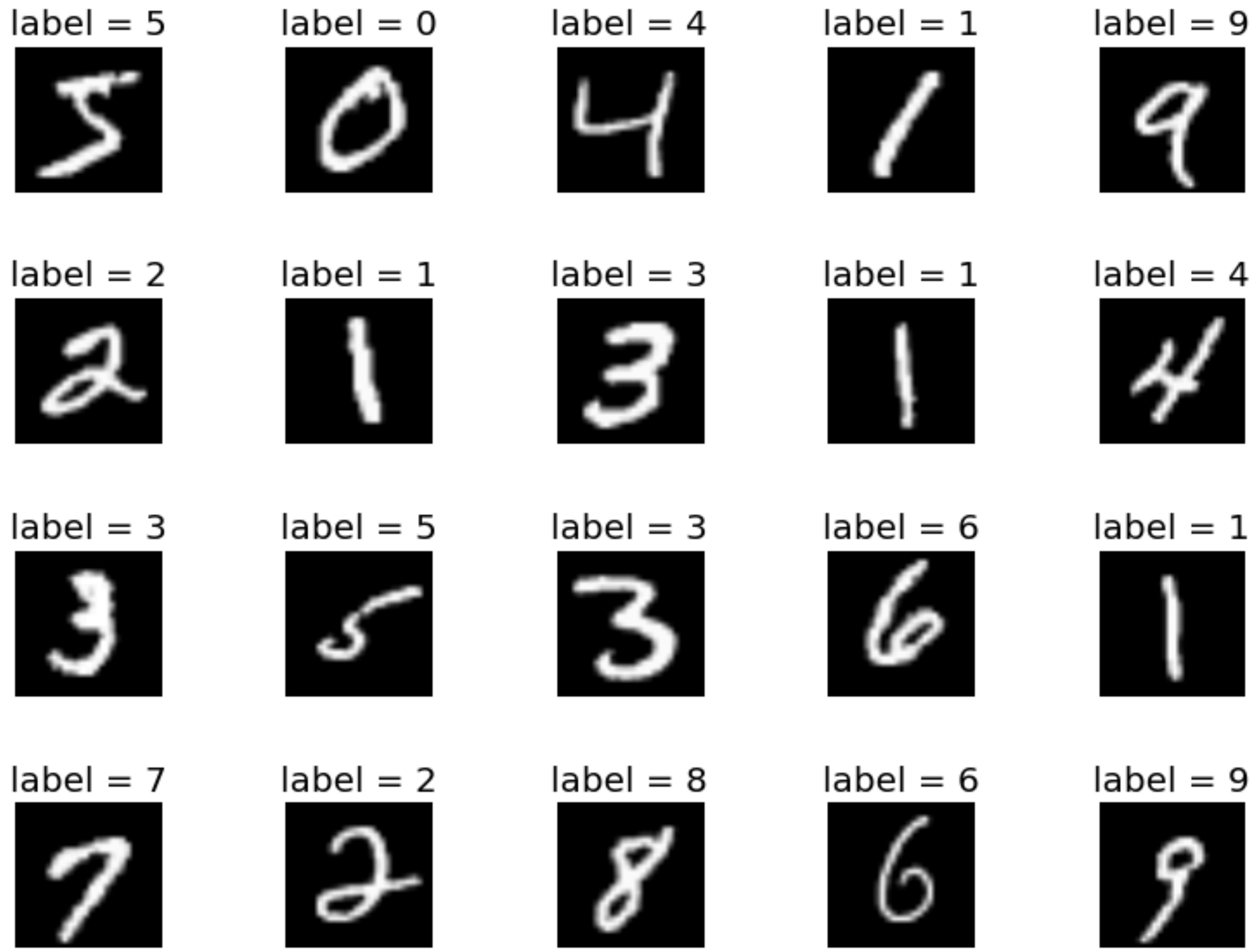
D = 784, M = 2



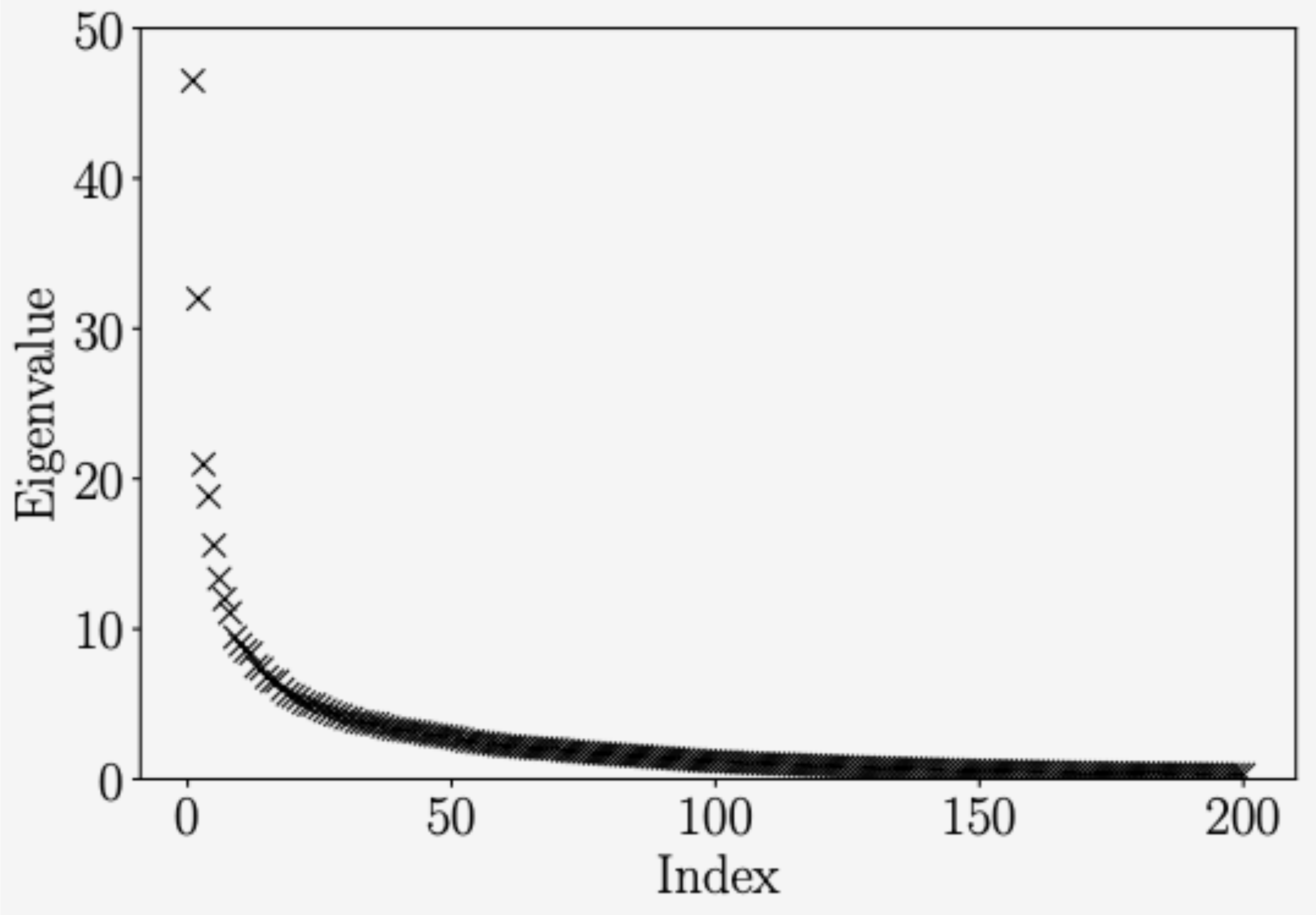
Key question: how to construct these mappings?

Example - dataset

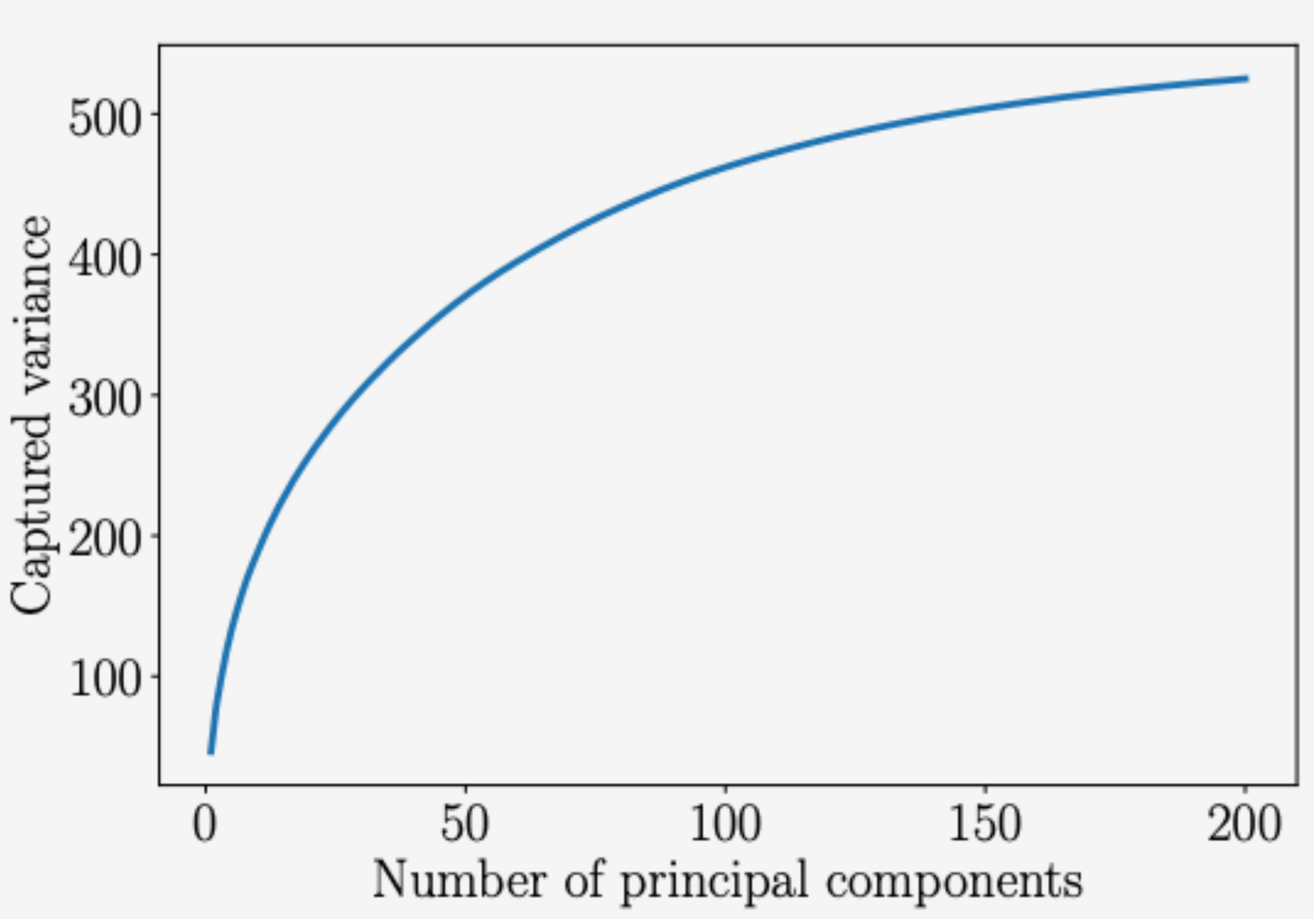
- 60,000 examples of handwritten digits 0 through 9.
- Each digit is a grayscale image of size 28×28, i.e., it contains 784 pixels.
- We can interpret every image in this dataset as a vector $x \in \mathbb{R}^{784}$



Example - PCA captured variance



(a) Top 200 largest eigenvalues



(b) Variance captured by the principal components.

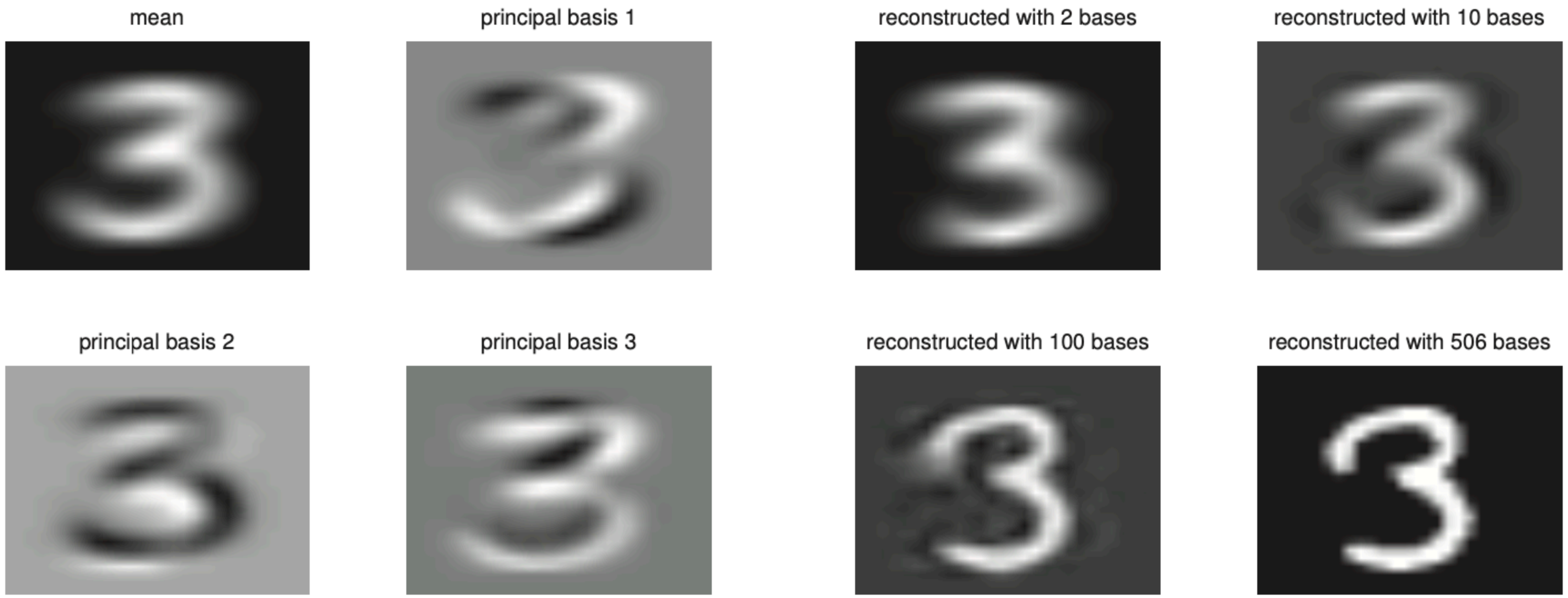
A 784-dim vector is used to represent an image

Taking all images of “3” in MNIST, we compute the eigenvalues of the data covariance matrix.

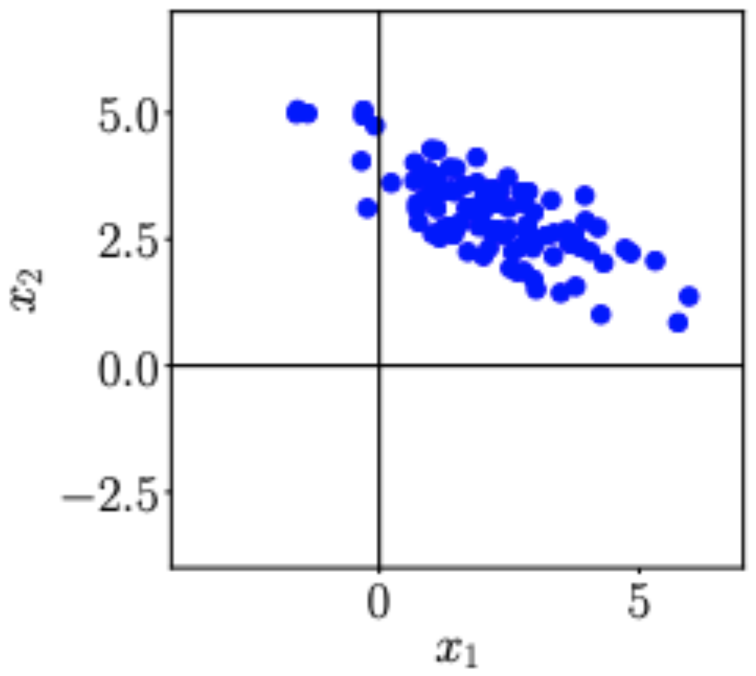
We see that only a few of them have a value that differs significantly from 0.

Most of the variance, when projecting data onto the subspace spanned by the corresponding eigenvectors, is captured by only a few principal components

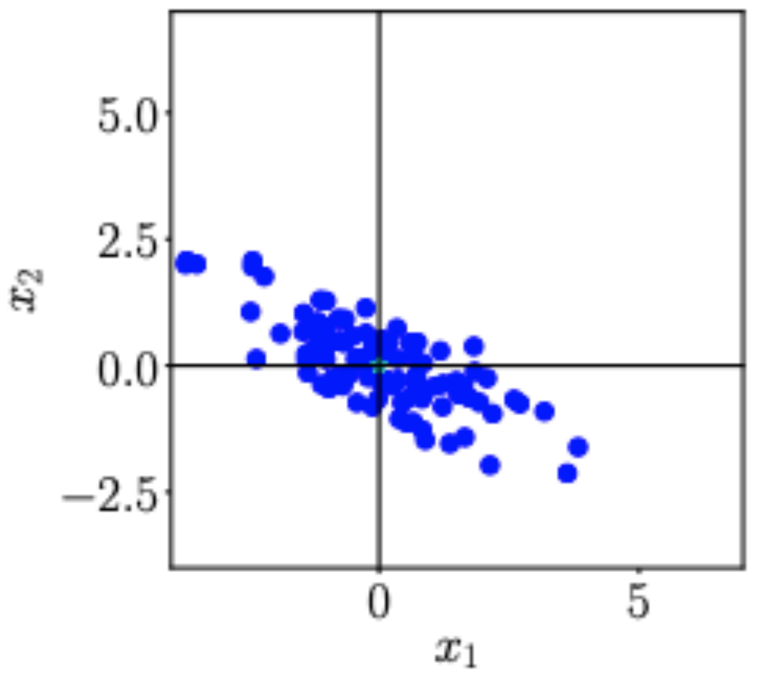
Example - PCA reconstruction



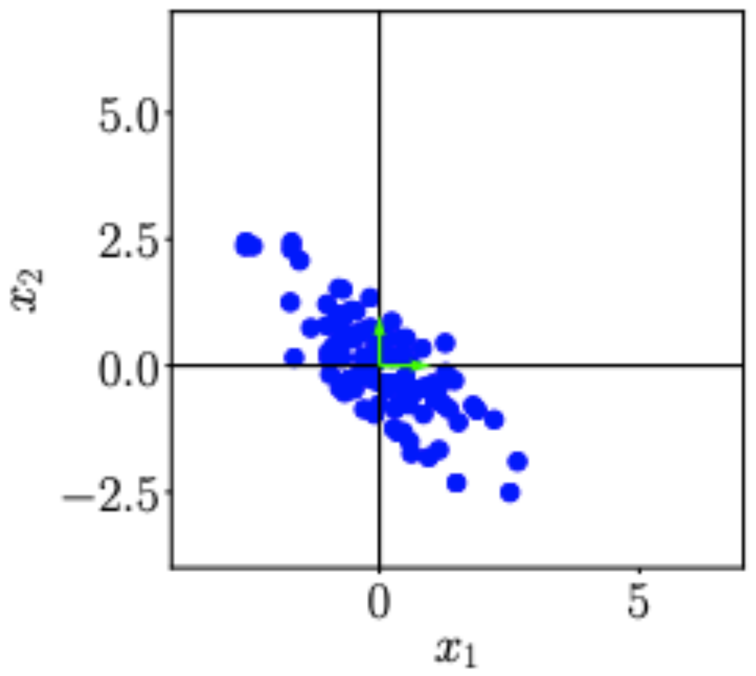
PCA in practice



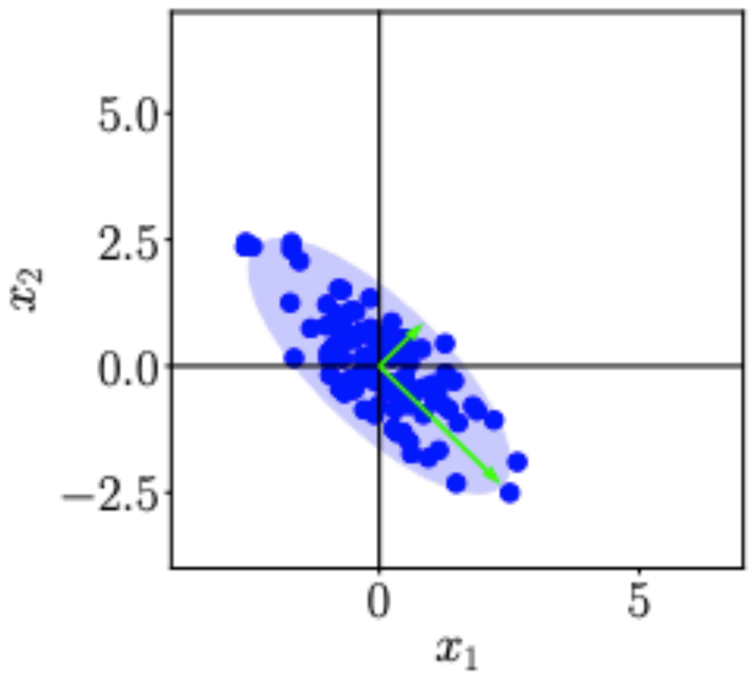
(a) Original dataset.



(b) Step 1: Centering by subtracting the mean from each data point.

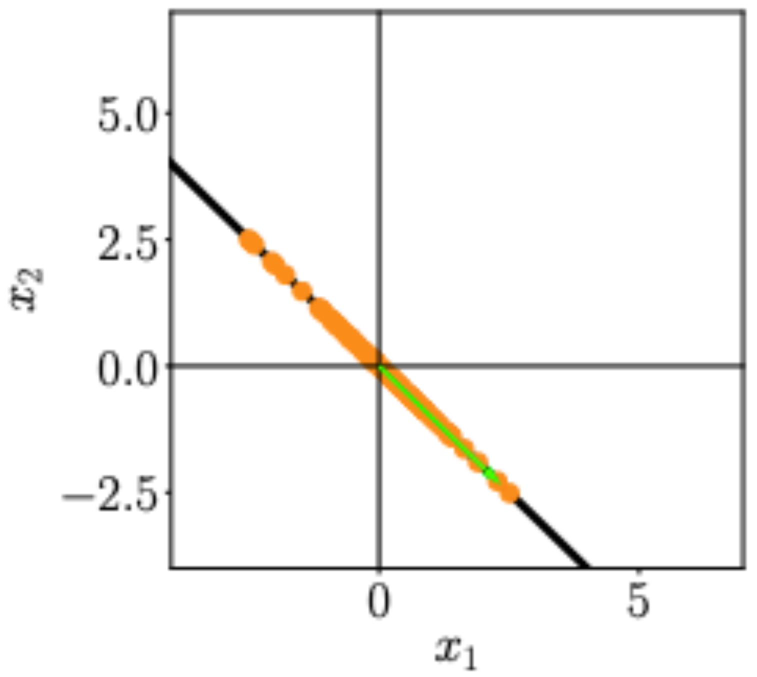


(c) Step 2: Dividing by the standard deviation to make the data unit free. Data has variance 1 along each axis.

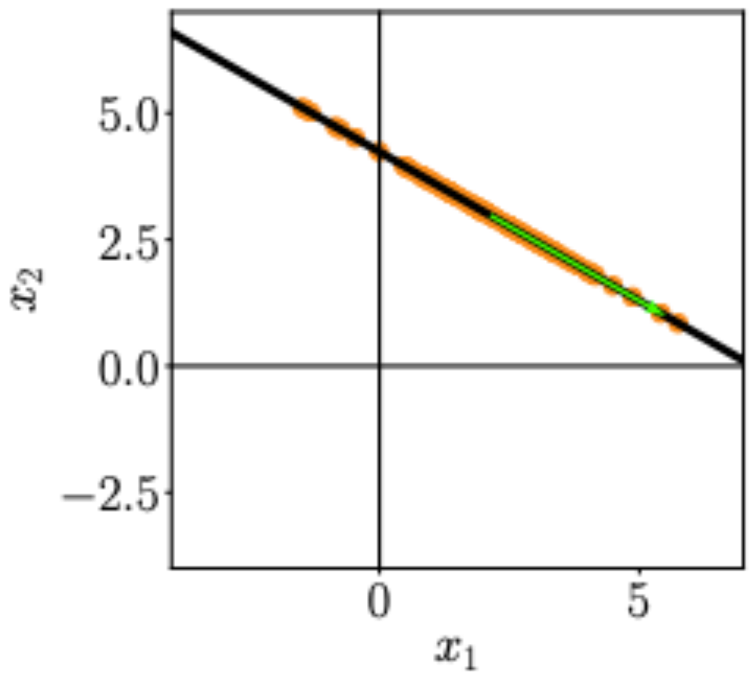


(d) Step 3: Compute eigenvalues and eigenvectors (arrows) of the data covariance matrix (ellipse).

eigendecomposition



(e) Step 4: Project data onto the principal subspace.



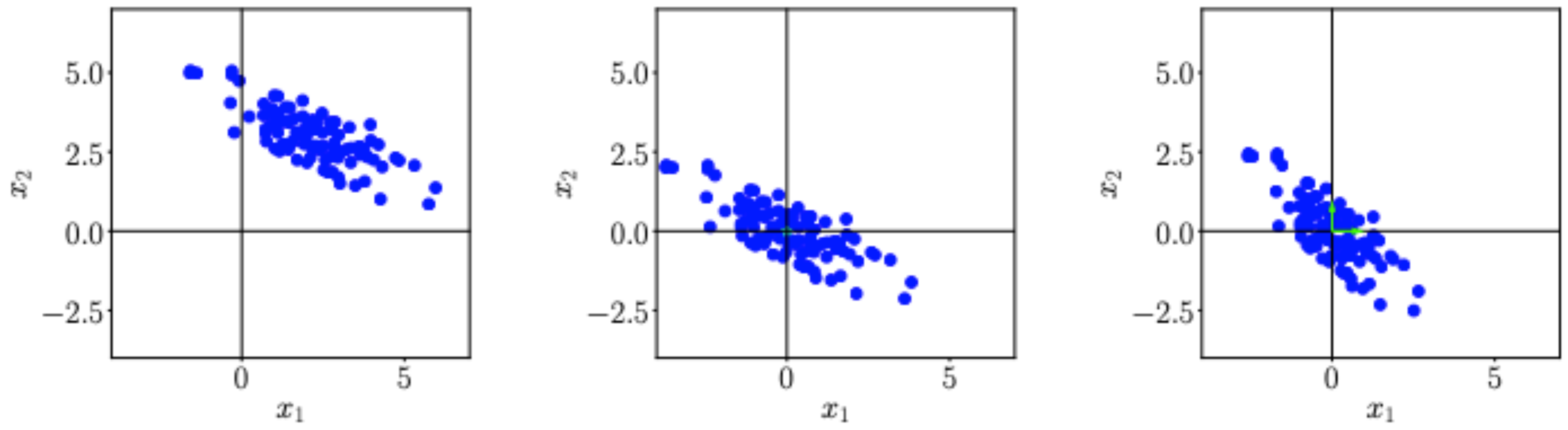
(f) Undo the standardization and move projected data back into the original data space from (a).

Step 1. Mean subtraction

We center the data by computing the mean μ of the dataset and subtracting it from every single data point. This ensures that the dataset has mean 0.

Step 2. Standardisation

Divide the data points by the standard deviation σ_d of the dataset for every dimension. Now the data has variance 1 along each axis.



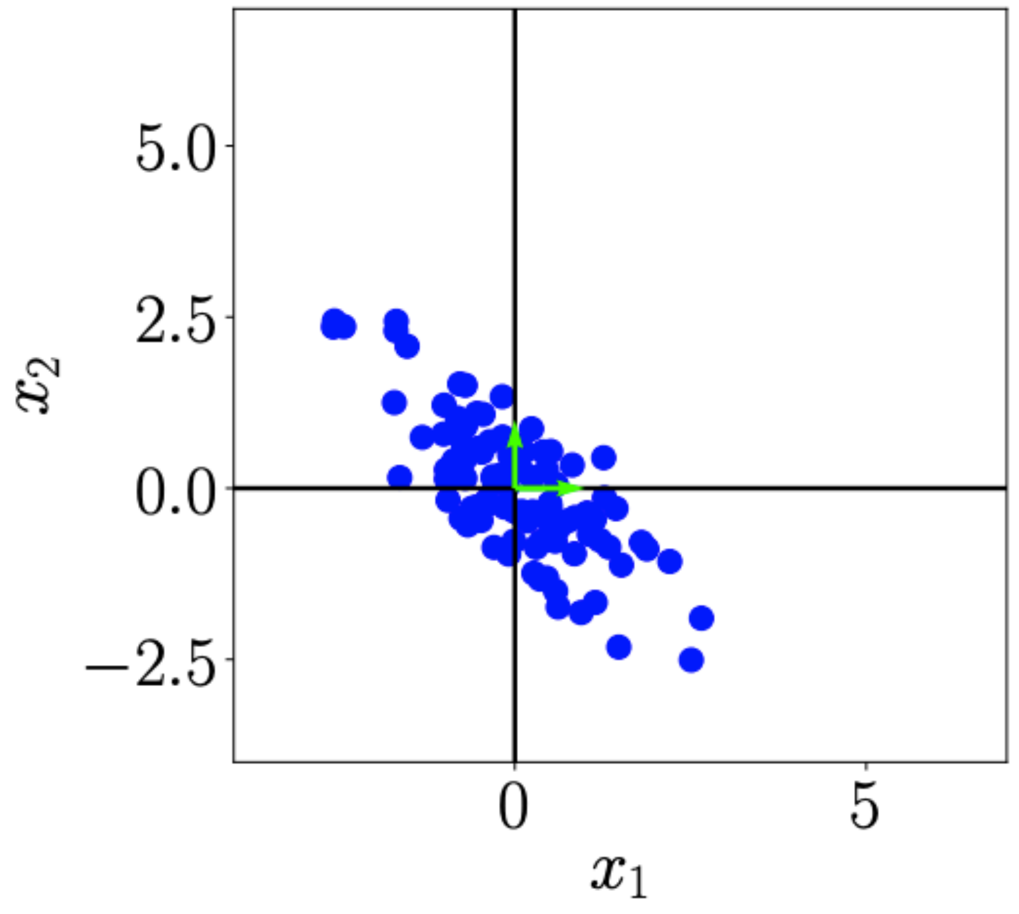
(a) Original dataset.

(b) Step 1: Centering by subtracting the mean from each data point.

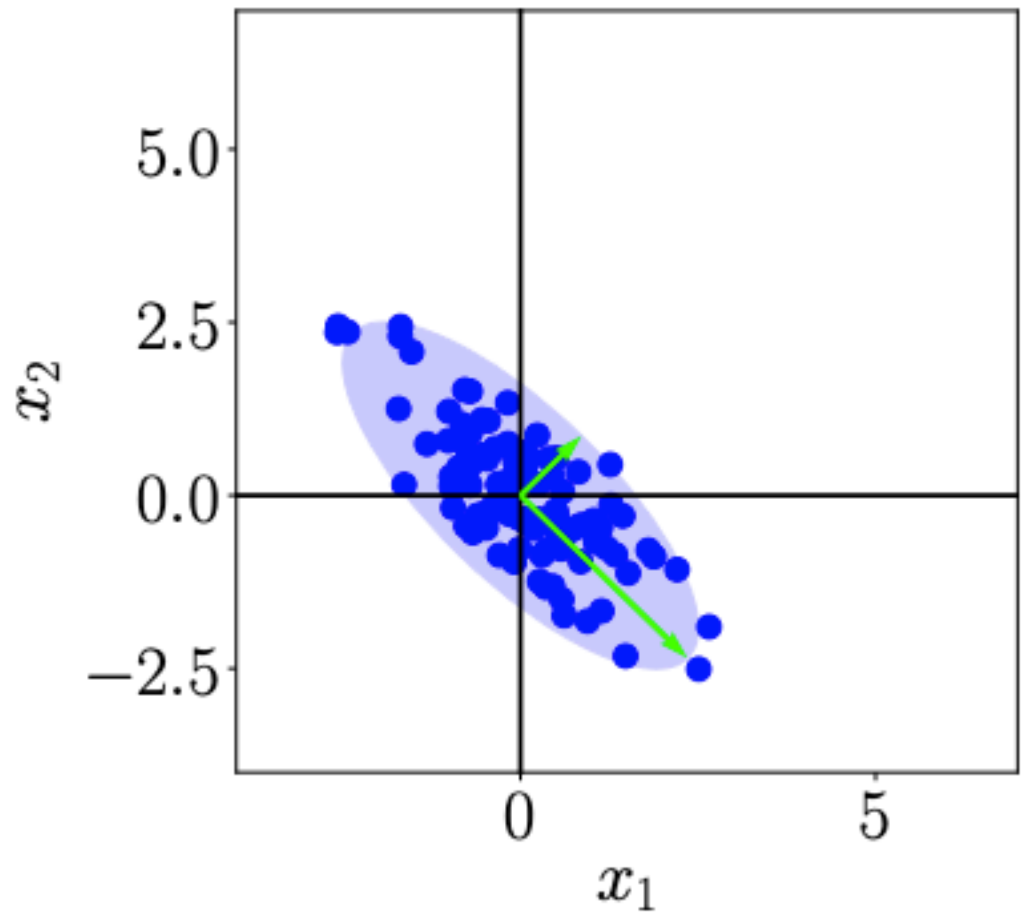
(c) Step 2: Dividing by the standard deviation to make the data unit free. Data has variance 1 along each axis.

Step 3. Eigendecomposition of the covariance matrix

Compute the data covariance matrix and its eigenvalues and corresponding eigenvectors. The longer vector (larger eigenvalue) spans the principal subspace U



(c) Step 2: Dividing by the standard deviation to make the data unit free. Data has variance 1 along each axis.



(d) Step 3: Compute eigenvalues and eigenvectors (arrows) of the data covariance matrix (ellipse).

4. **Projection**

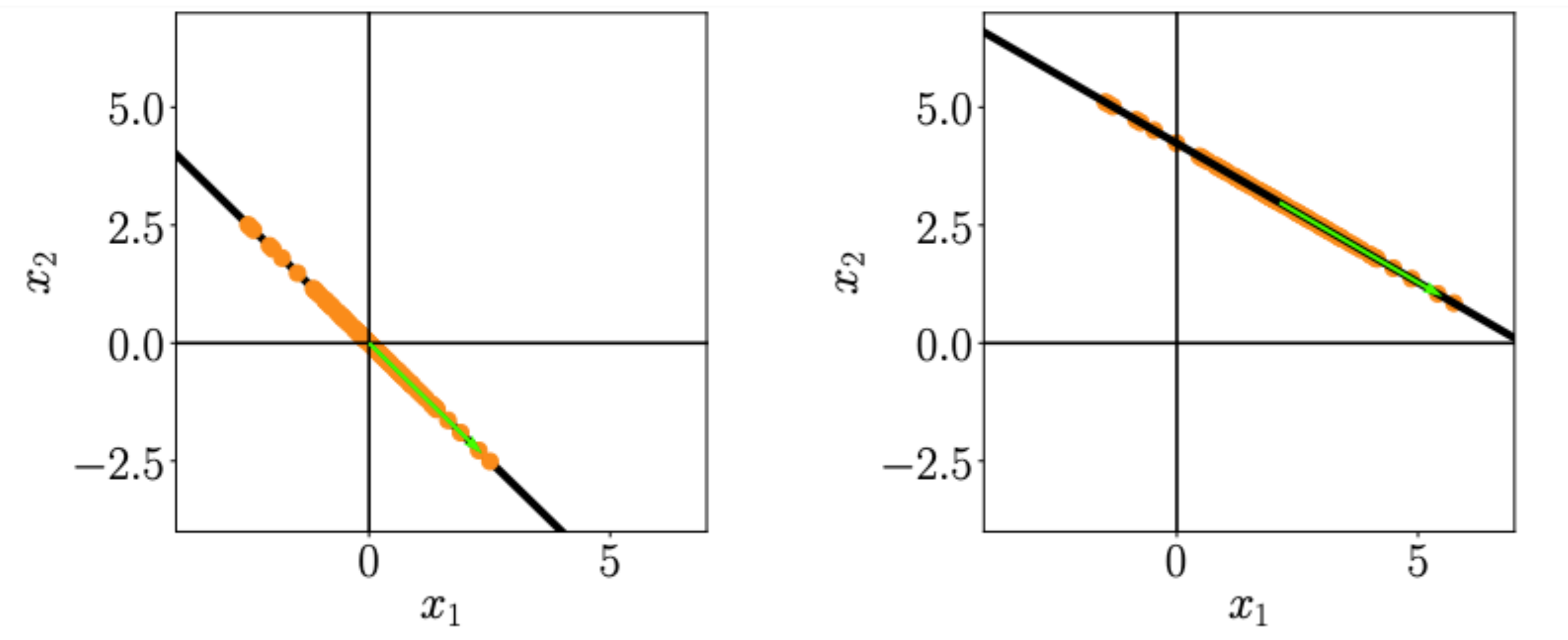
We can project any data point $\boldsymbol{x}_* \in \mathbb{R}^D$ onto the principal subspace.

projection as $\tilde{\boldsymbol{x}}_* = \boldsymbol{B}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{x}_*$

coordinates $\boldsymbol{z}_* = \boldsymbol{B}^{\mathrm{T}}\boldsymbol{x}_*$ with respect to the basis of the principal subspace. Here, $\boldsymbol{B}$ is the matrix that contains the eigenvectors that are associated with the largest eigenvalues of the data covariance matrix as columns.

5. Rescaling data

To obtain our projection in the original data space (i.e., before standardization), we need to undo the standardization: multiply by the standard deviation before adding the mean.



(e) Step 4: Project data onto the principal subspace.

(f) Undo the standardization and move projected data back into the original data space from (a).

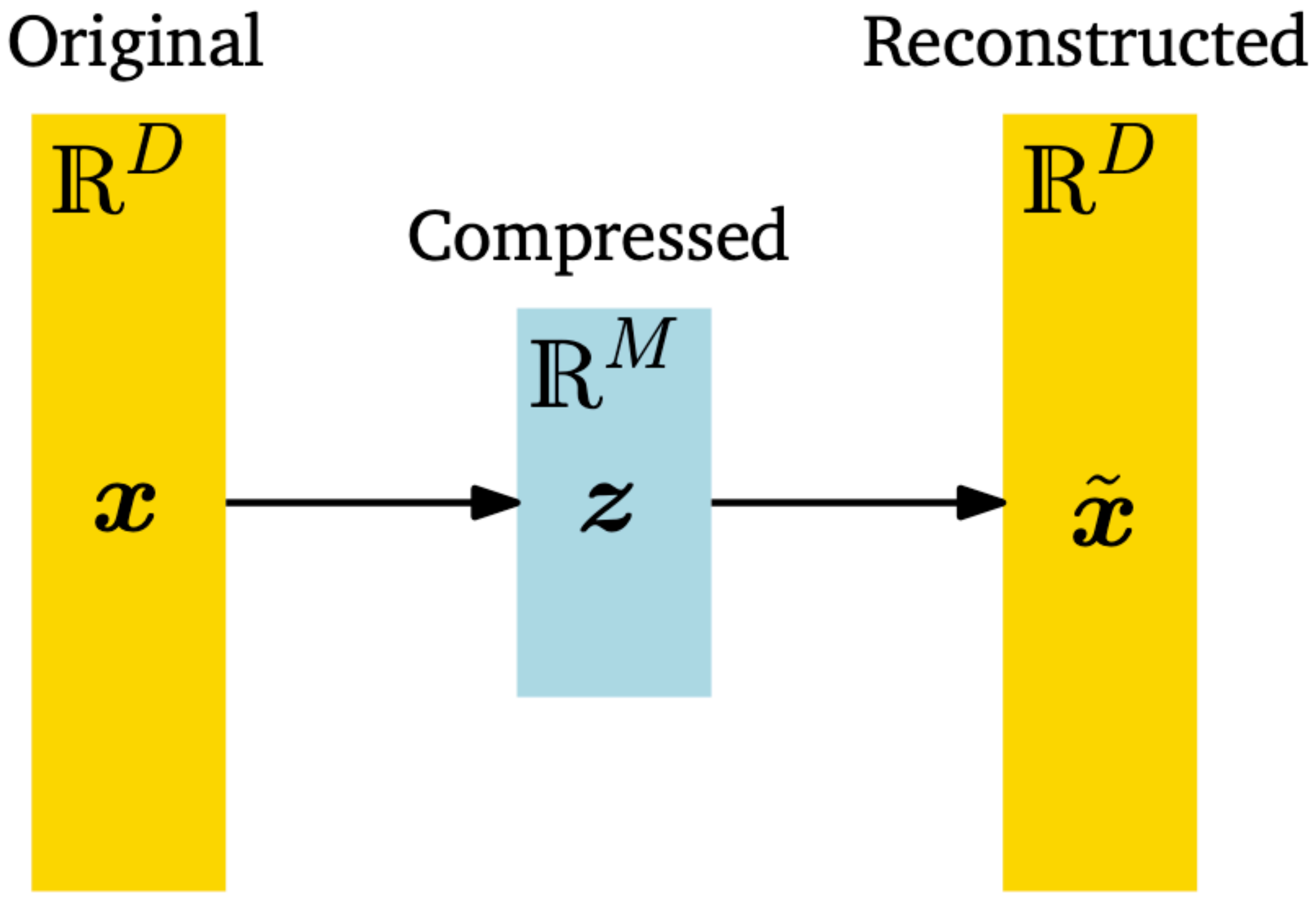
Exercise: Show that PCA is rotationally invariant

Overview

1. Motivation
2. **PCA review**
3. Linear Gaussian latent variable models and GPLVM

Reading: Bishop 12.1, 12.2, 12.4.2

Problem setup



Problem setup

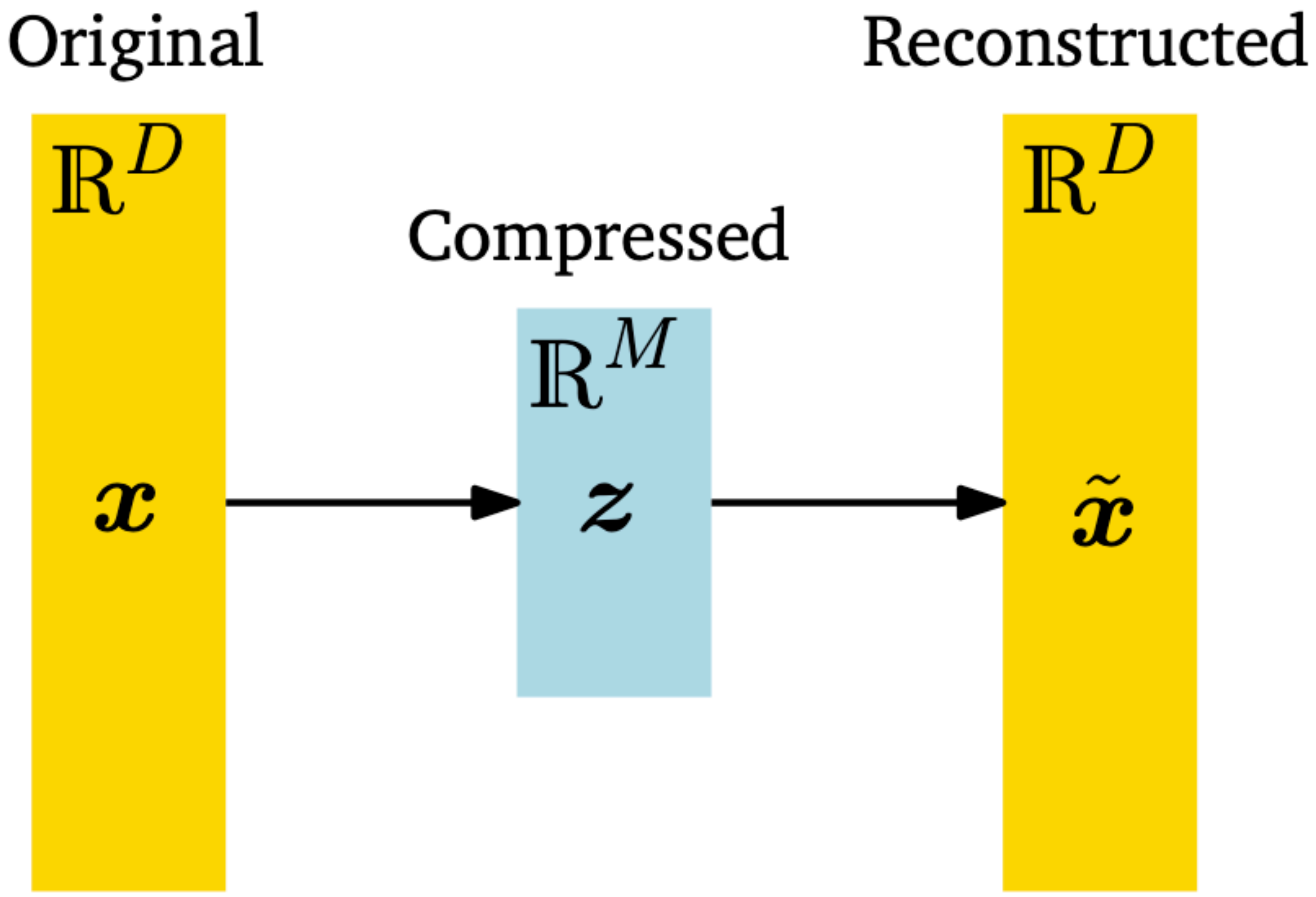
We consider an i.i.d. dataset $X = \{x_1, x_2, \ldots, x_N\}, x_n \in \mathbb{R}^D$,

with mean $\mathbf{0}$ and covariance matrix $S = \dfrac{1}{N}\sum_{n=1}^{N} x_n x_n^\intercal$

We assume there exists a *low-dimensional* compressed representation (code): $z_n = B^\intercal x_n$, $z_n \in \mathbb{R}^M$, $M < D$.

The projection matrix: $B = \left[b_1, b_2, \ldots, b_M\right] \in \mathbb{R}^{D\times M}$, columns are orthonormal.

Reconstruction using B: $\tilde{x}_n = Bz_n$



Problem setup

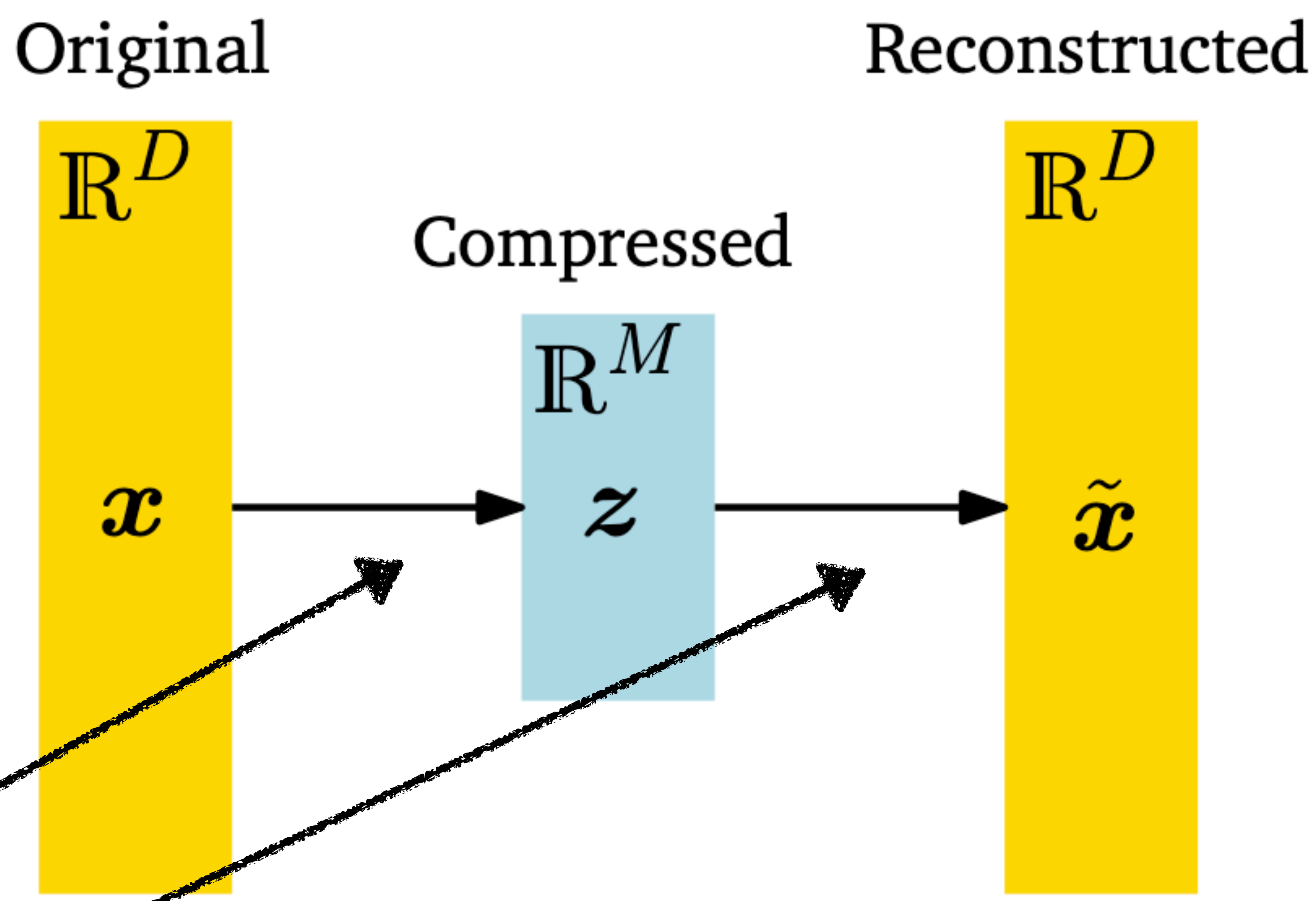
We consider an i.i.d. dataset $X = \{x_1, x_2, \ldots, x_N\}$, $x_n \in \mathbb{R}^D$,

with mean $\mathbf{0}$ and covariance matrix $S = \dfrac{1}{N}\sum_{n=1}^{N} x_n x_n^\intercal$

We assume there exists a *low-dimensional* compressed representation (code): $z_n = B^\intercal x_n$, $z_n \in \mathbb{R}^M$, $M < D$.

The projection matrix: $B = [b_1, b_2, \ldots, b_M] \in \mathbb{R}^{D \times M}$, columns are orthonormal.

Reconstruction using B: $\tilde{x}_n = B z_n$



PCA: linear mappings

Problem setup

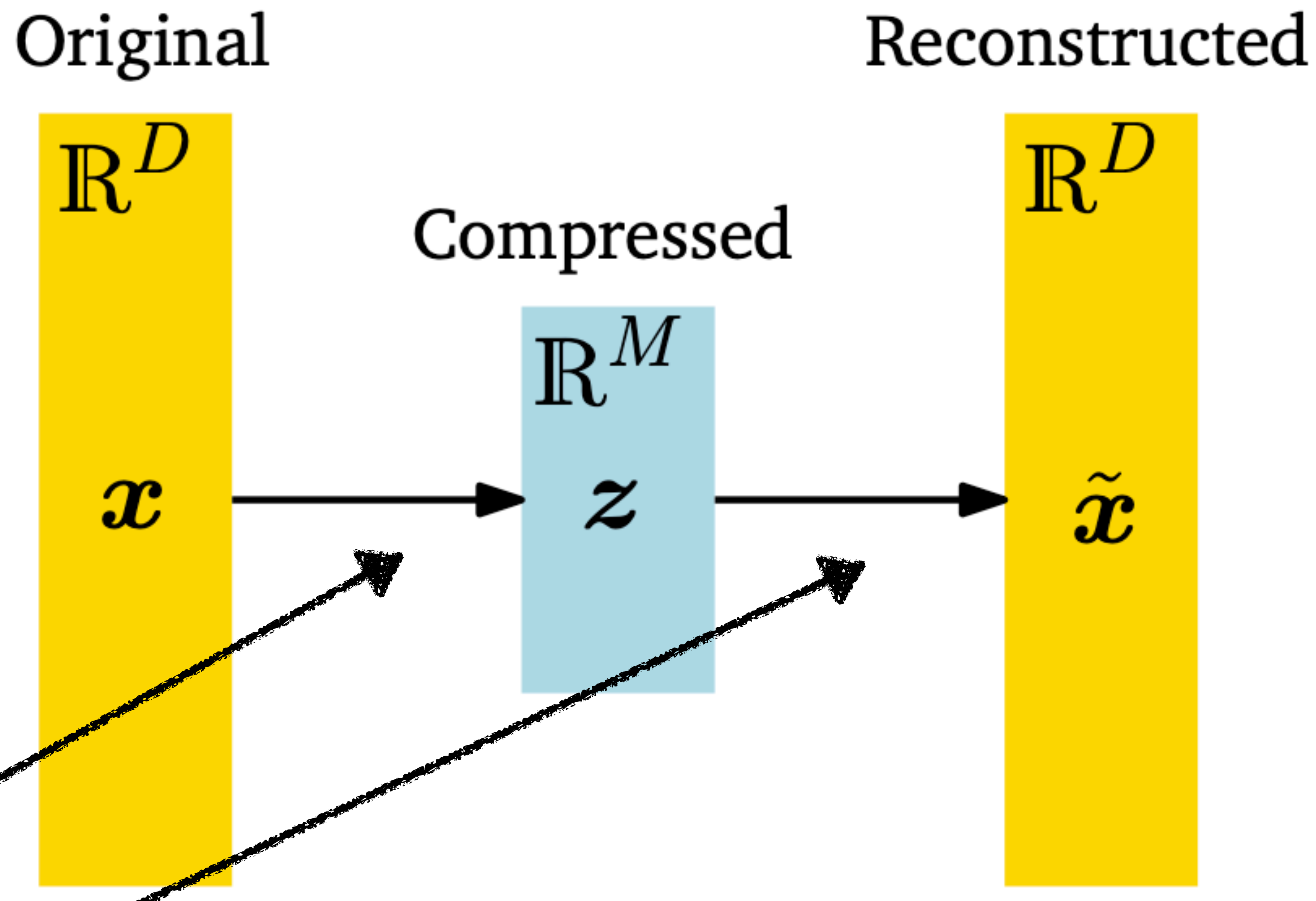
We consider an i.i.d. dataset $X = \{x_1, x_2, \ldots, x_N\}$, $x_n \in \mathbb{R}^D$,

with mean $\mathbf{0}$ and covariance matrix $S = \dfrac{1}{N}\sum_{n=1}^{N} x_n x_n^\intercal$

We assume there exists a *low-dimensional* compressed representation (code): $z_n = B^\intercal x_n$, $z_n \in \mathbb{R}^M$, $M < D$.

The projection matrix: $B = [b_1, b_2, \ldots, b_M] \in \mathbb{R}^{D\times M}$, columns are orthonormal.

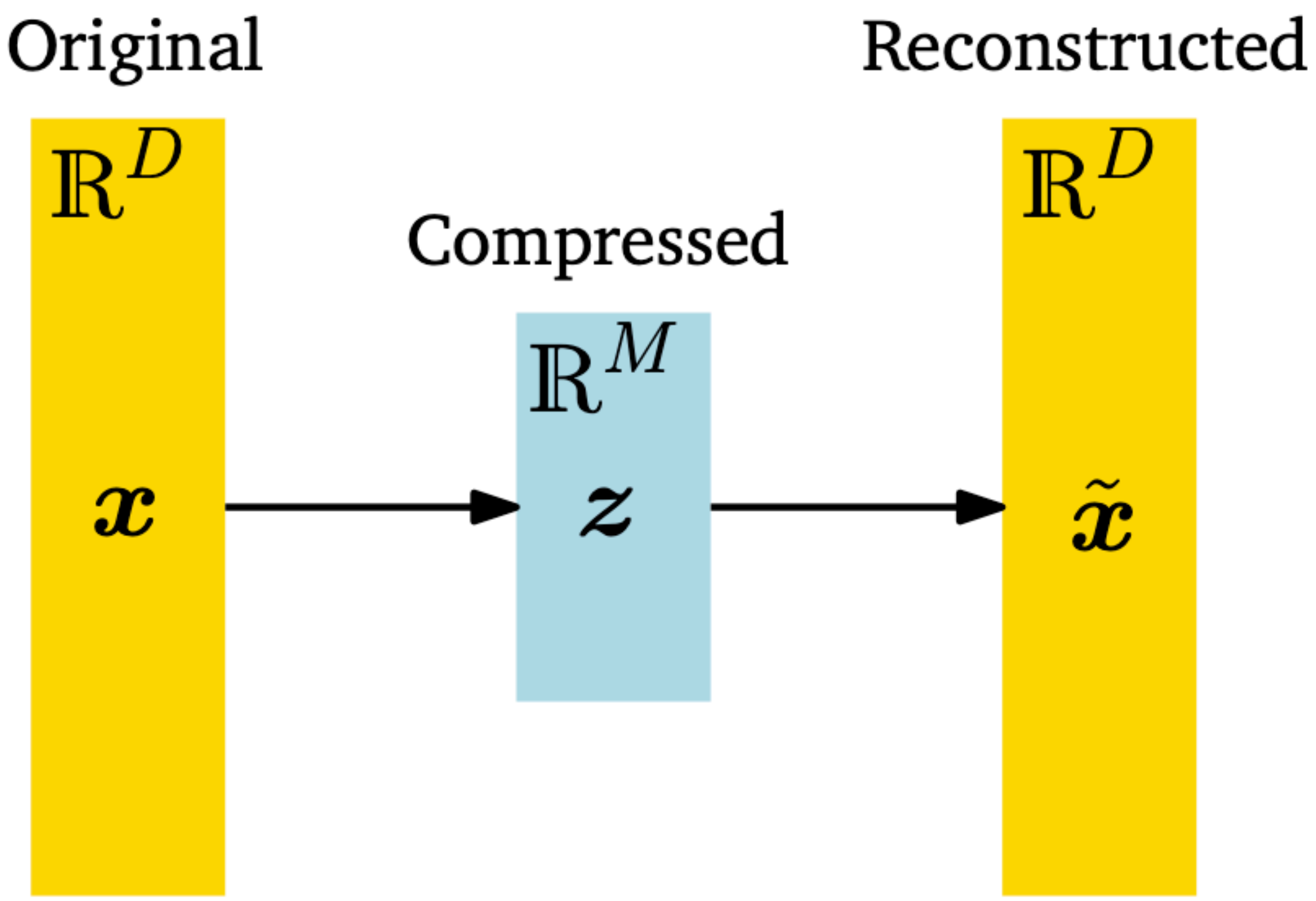
Reconstruction using B: $\tilde{x}_n = B z_n$



PCA: linear mappings

Goal: find z_n and the *basis vectors* $b_1, b_2, \ldots, b_M$ so that the reconstructed data are *similar* to the original data, and the compressed data retain most of the *variation* in the original data

PCA - two perspectives



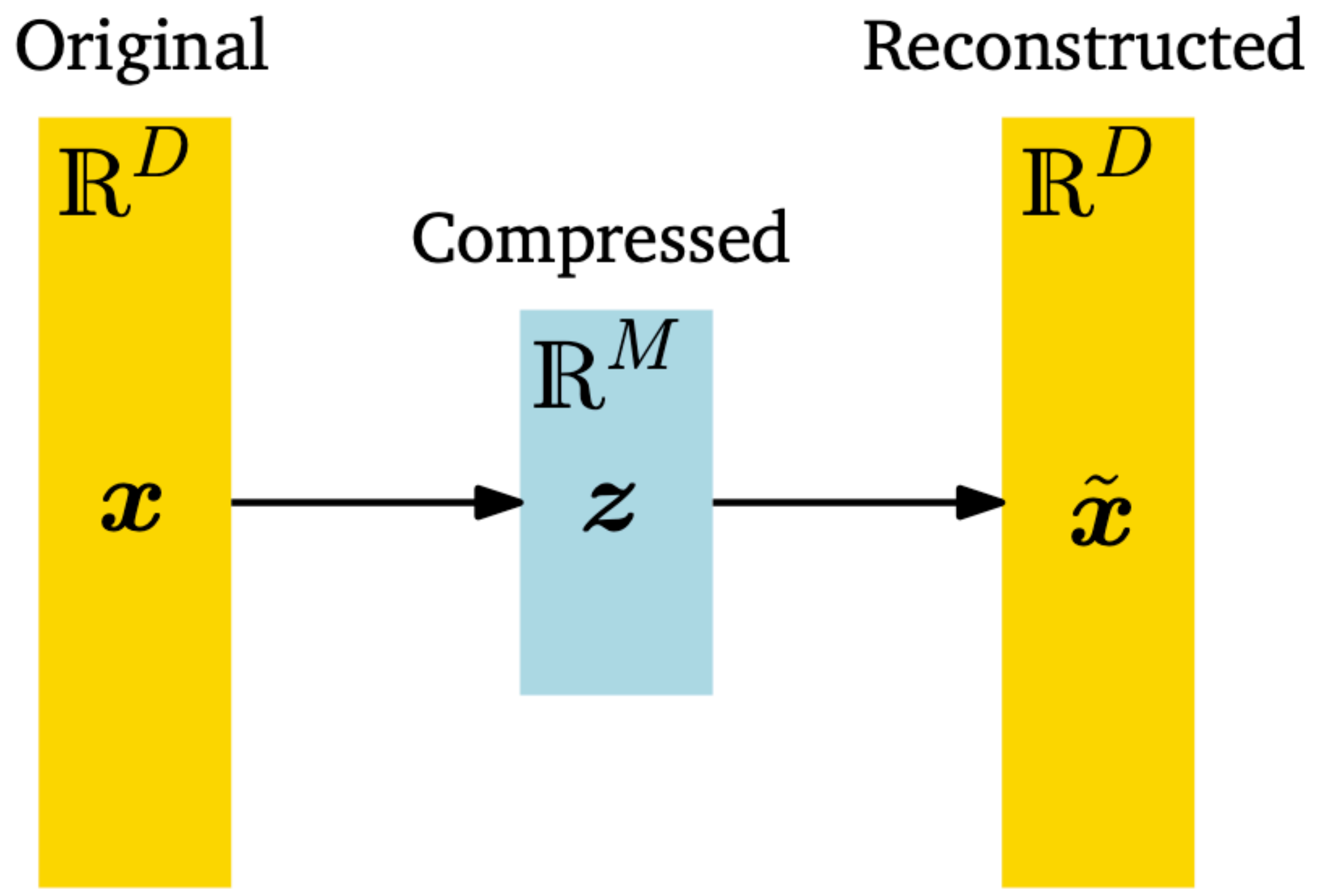
PCA: linear mappings

$z_n = B^\intercal x_n,\ z_n \in \mathbb{R}^M,\ M < D$

$\tilde{x}_n = Bz_n$

PCA - two perspectives

Goal: find z_n and the *basis vectors* $b_1, b_2, \dots, b_M$ so that the reconstructed data are *similar* to the original data, and the compressed data retain most of the *variation* in the original data.



PCA: linear mappings

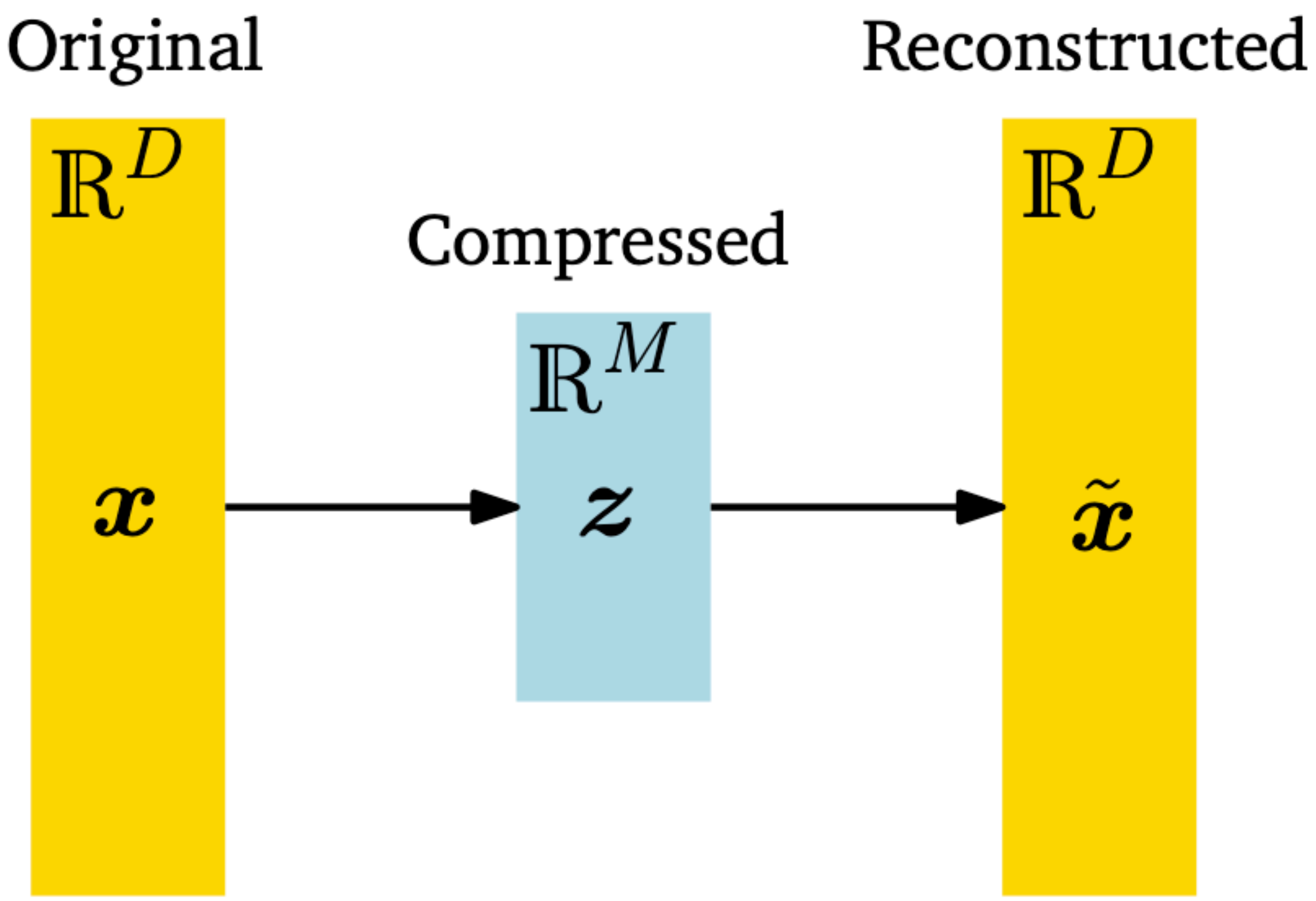
$z_n = B^\intercal x_n,\ z_n \in \mathbb{R}^M,\ M < D$

$\tilde{x}_n = B z_n$

PCA - two perspectives

Goal: find z_n and the *basis vectors* $b_1, b_2, \ldots, b_M$ so that the reconstructed data are *similar* to the original data, and the compressed data retain most of the *variation* in the original data.

Question: Next steps? Ideas?



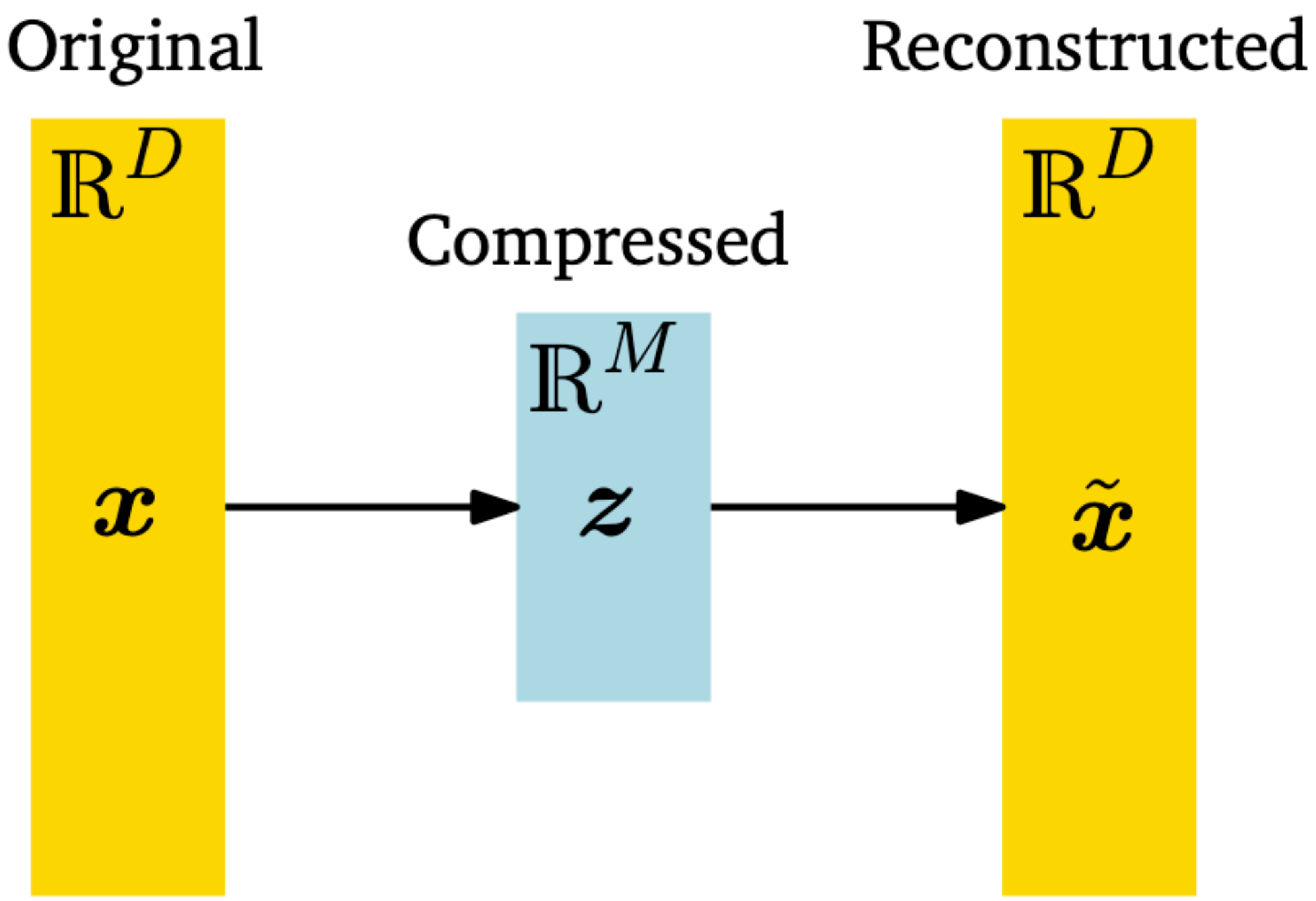
PCA: linear mappings

$z_n = B^\intercal x_n,\ z_n \in \mathbb{R}^M,\ M < D$

$\tilde{x}_n = Bz_n$

PCA - two perspectives

Goal: find z_n and the *basis vectors* $b_1, b_2, \ldots, b_M$ so that the reconstructed data are *similar* to the original data, and the compressed data retain most of the *variation* in the original data.



Question: Next steps? Ideas?

Answer: Two approaches

+ Search for B that **maximises** the **variance** of the low-dimensional representations [analysis/max var perspective]
+ Search for B and z that minimises the reconstruction loss [synthesis/projection perspective]

Both give *identical* solutions! **Why?**

PCA: linear mappings

$z_n = B^\intercal x_n,\ z_n \in \mathbb{R}^M,\ M < D$

$\tilde{x}_n = Bz_n$

Overview

1. Motivation
2. PCA review
3. **Linear, Gaussian latent variable models and GPLVM [whiteboard]**

Reading: Bishop 12.1, 12.2, 12.4.2

Probabilistic PCA

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \mathrm{I})$$

$$p(\mathbf{x} \mid \mathbf{z}) = \mathcal{N}(\mathbf{x}; W\mathbf{z} + \mu, \sigma^2 \mathrm{I})$$

Benefits: a proper probabilistic density model

+ EM algorithm for computational efficiency
+ Can be extended to handle binary/categorical data
+ Can be extended to handle discrete latent variables
+ Generate samples

Notes: find the same principal subspace as PCA

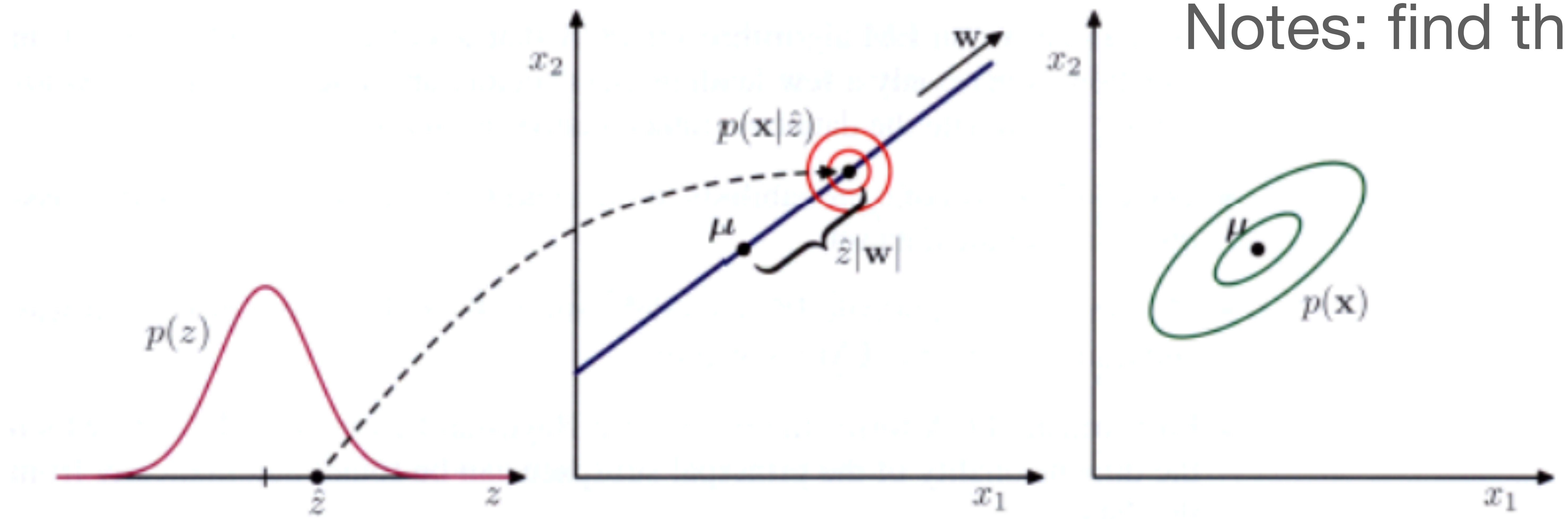


Figure 12.9 An illustration of the generative view of the probabilistic PCA model for a two-dimensional data space and a one-dimensional latent space. An observed data point $\mathbf{x}$ is generated by first drawing a value $\hat{z}$ for the latent variable from its prior distribution $p(z)$ and then drawing a value for $\mathbf{x}$ from an isotropic Gaussian distribution (illustrated by the red circles) having mean $\mathbf{w}\hat{z} + \boldsymbol{\mu}$ and covariance $\sigma^2\mathbf{I}$. The green ellipses show the density contours for the marginal distribution $p(\mathbf{x})$.

Overview

1. Motivation
2. PCA review
3. Probabilistic PCA: linear, Gaussian latent variable models and GPLVM

Reading: Bishop 12.1, 12.2, 12.4.2

Enjoy the break and see you in W12!