Dimensionality Reduction

By Alex Cayco Gajic

neuromatch academy

Who is Alex Cayco Gajic?

- Junior Professor @ ENS (Paris)

- Motor control, cerebellar theory
- Population coding, statistical learning

Overview of tutorials

1. Geometric view of data
2. Principal component analysis

Toy data (2D)

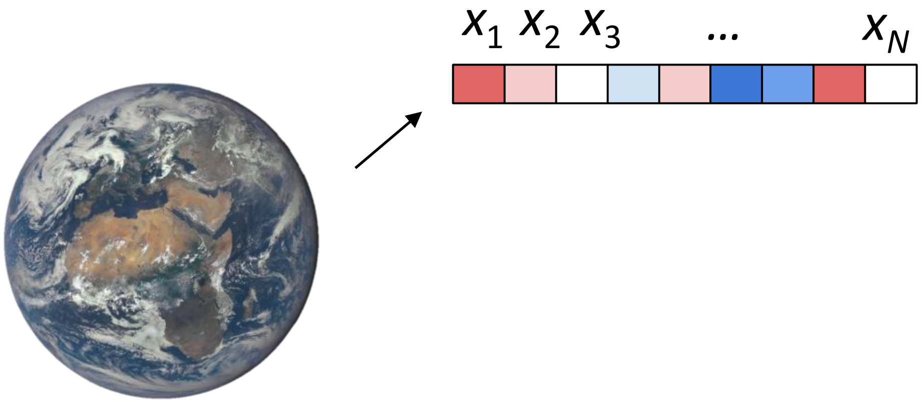
1. Dimensionality reduction and reconstruction
2. Nonlinear dimensionality reduction

Real data (high-D)

Geometric view of data

Tutorial 1

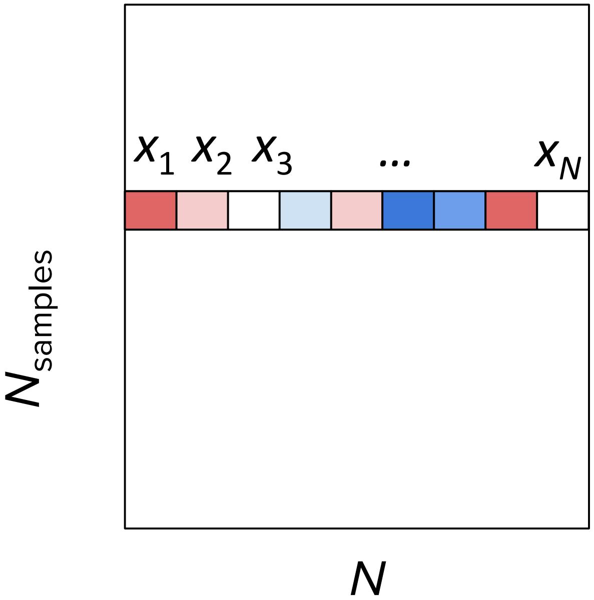
Multivariate data



Sample of *N* variables during a single observation.

How to represent multivariate data?

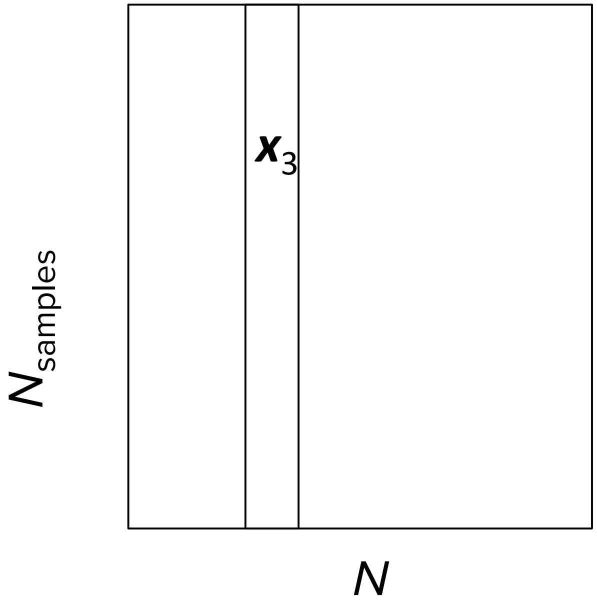
$\mathbf{X} =$



Single sample of all variables

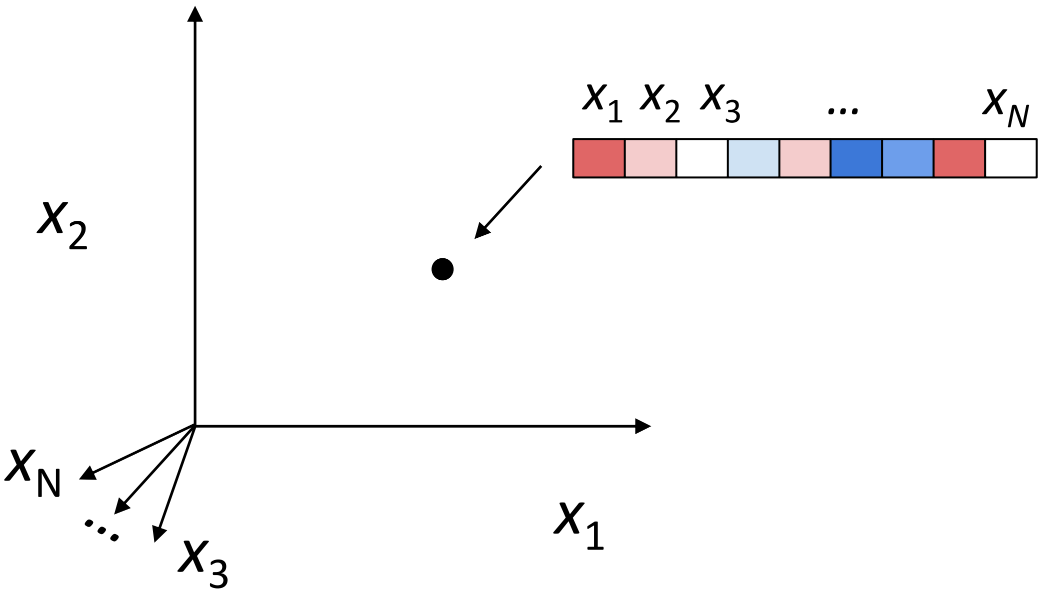
How to represent multivariate data?

$\mathbf{X} =$

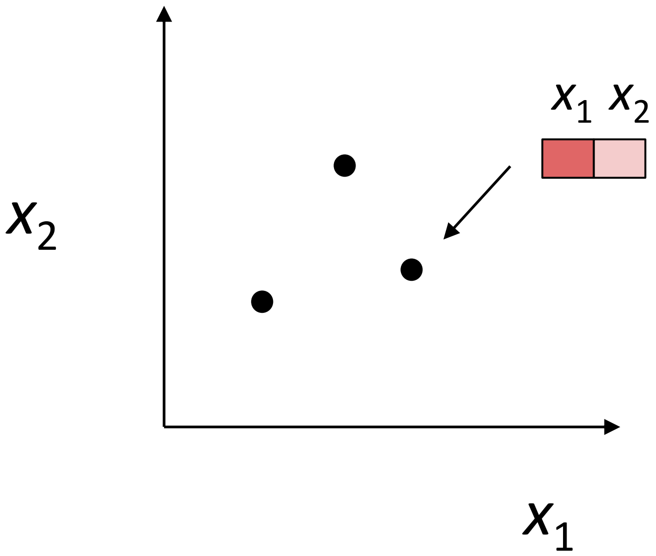


All samples of a **single** variable

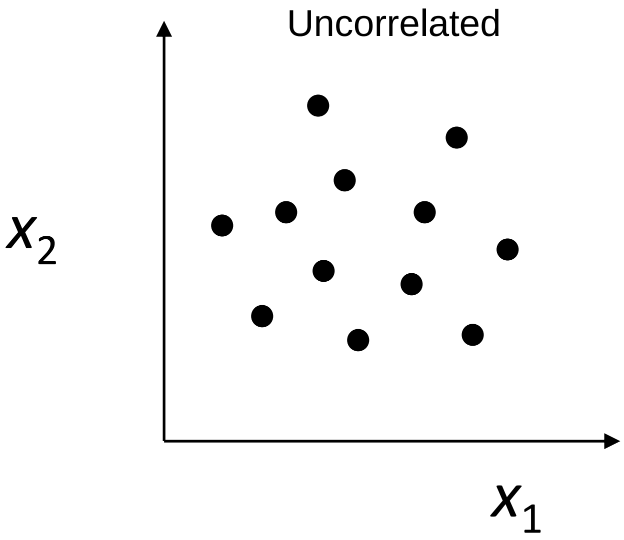
How to represent multivariate data?



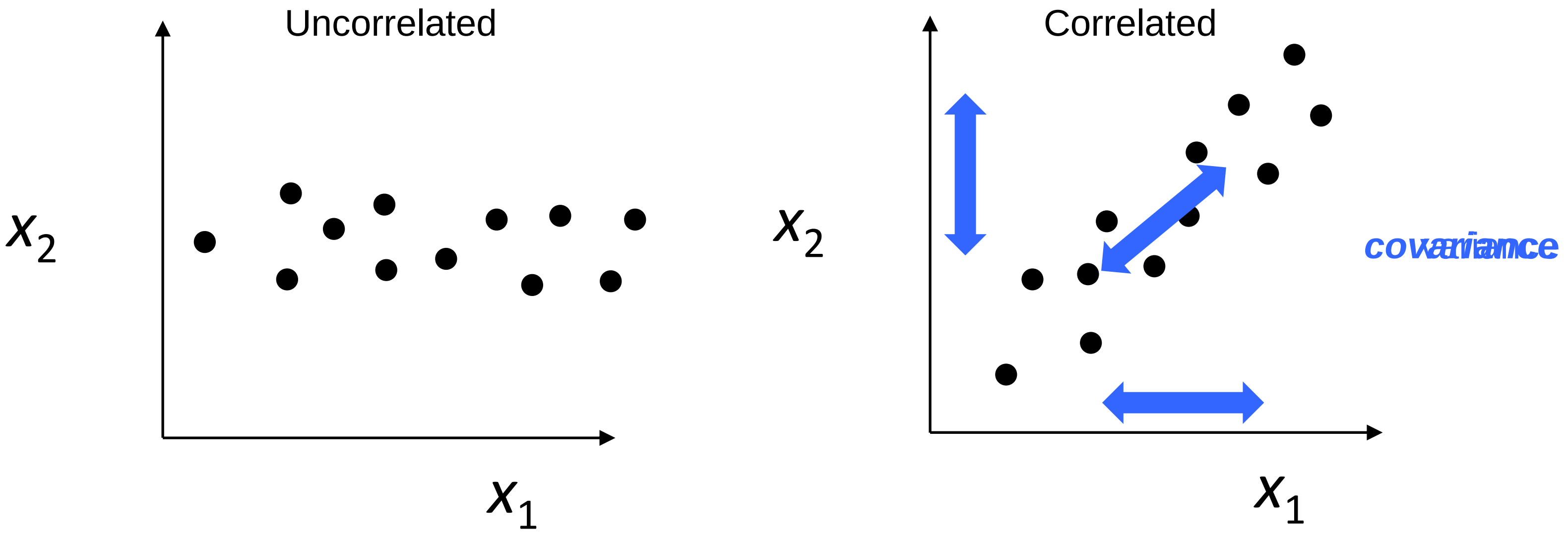
How to represent multivariate data?



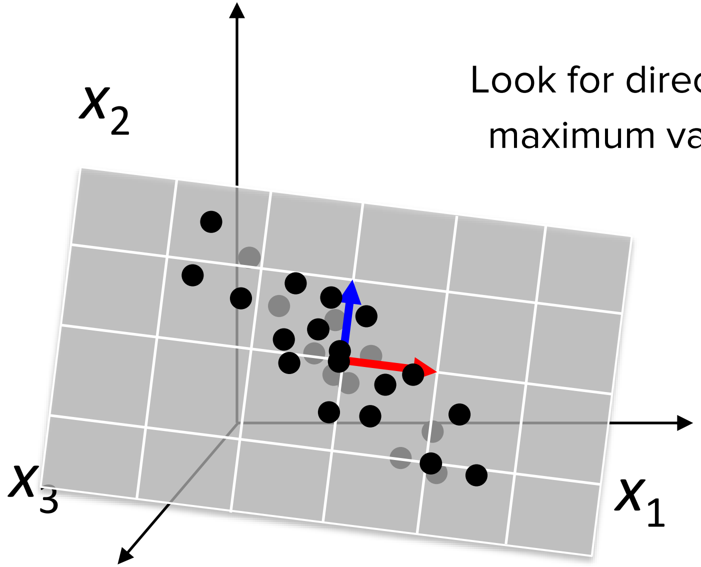
Multivariate data has variability



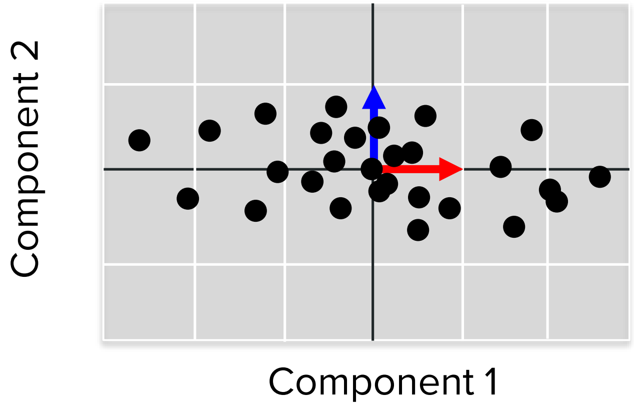
Multivariate data has variability



PCA: the big picture



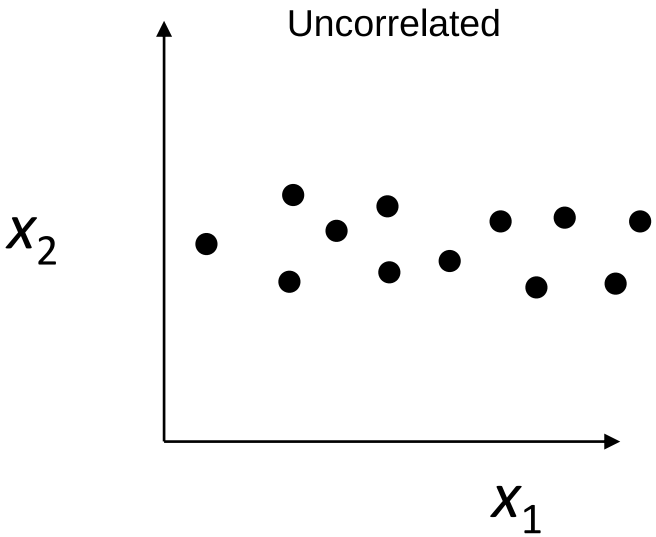
Look for directions of maximum variance.

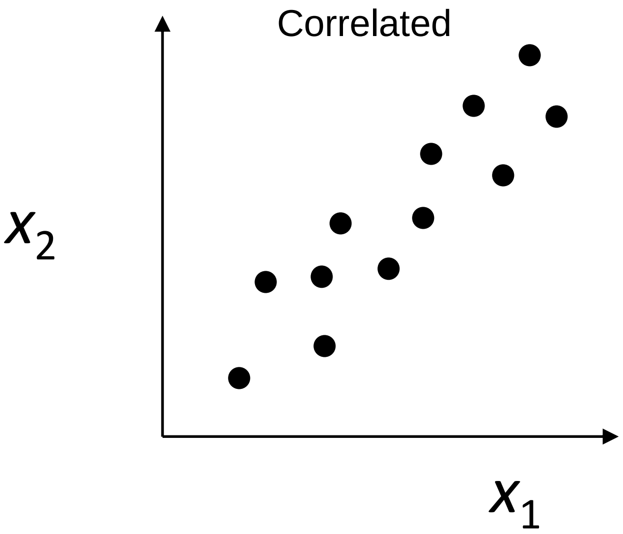


(break for tutorial exercise)

neuromatch academy

Multivariate data has variability





How to quantify variability

Variance $$\mathrm{var}(x_1) = E[x_1^2] - E[x_1]^2$$

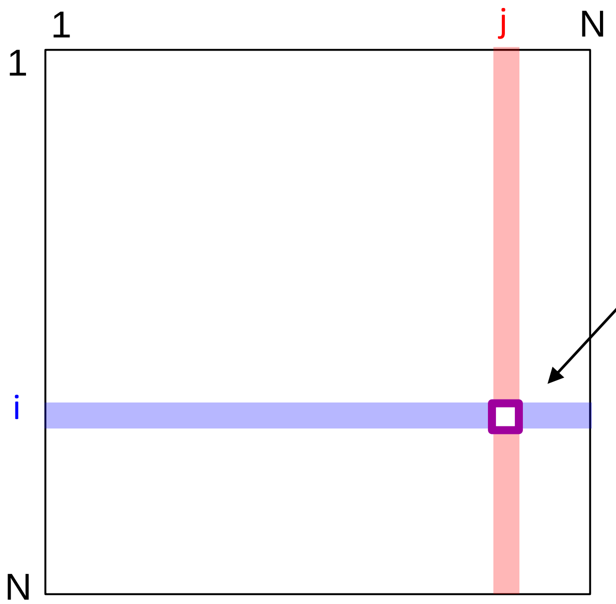
Covariance $$\mathrm{cov}(x_1, x_2) = E[x_1 x_2] - E[x_1]E[x_2]$$

Normalized to be within -1 to +1

Correlation $$\rho = \frac{\mathrm{cov}(x_1, x_2)}{\sqrt{\mathrm{var}(x_1)\mathrm{var}(x_2)}}$$

The covariance matrix **Σ**

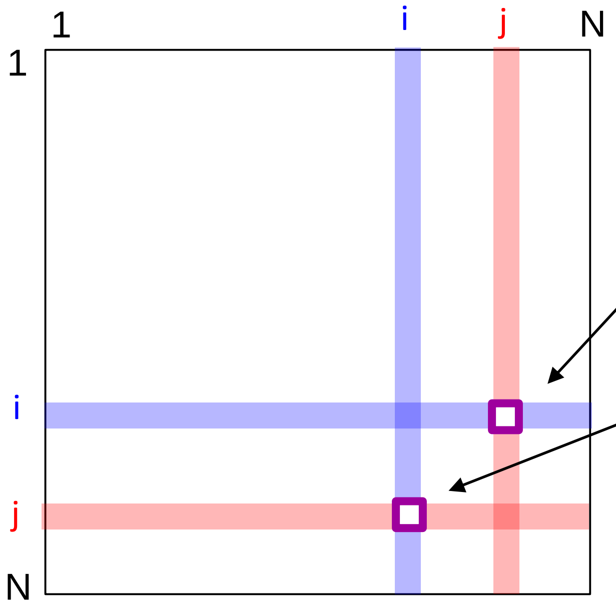
$$\Sigma =$$



$$\Sigma_{ij} = \mathrm{cov}(x_i, x_j)$$

The covariance matrix $\boldsymbol{\Sigma}$

$$\boldsymbol{\Sigma} =$$

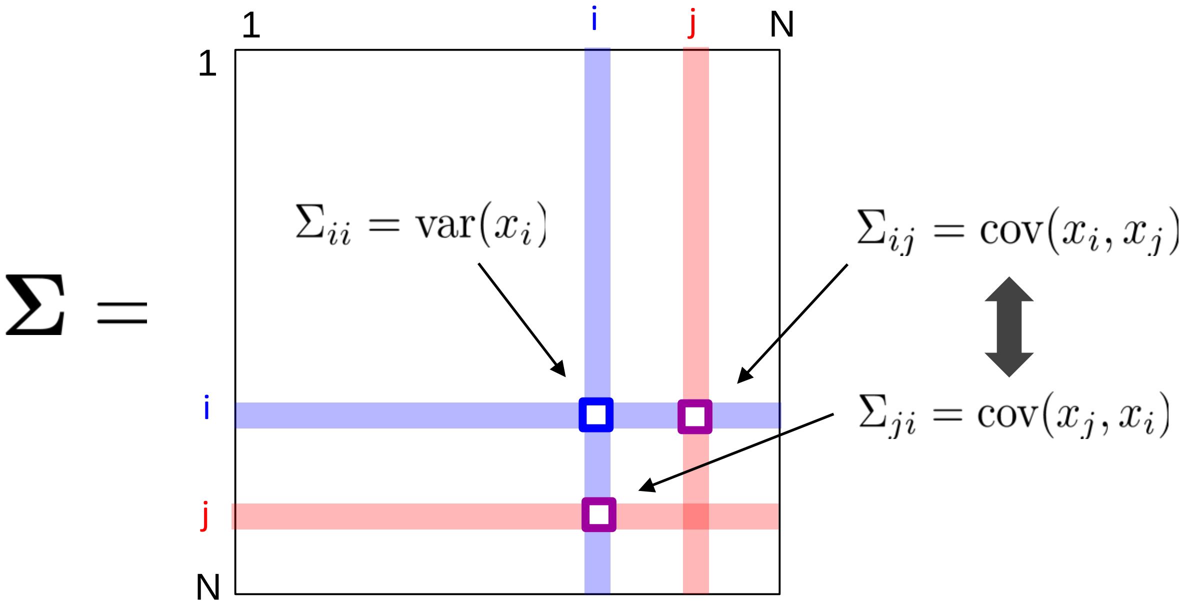


$$\Sigma_{ij} = \mathrm{cov}(x_i, x_j)$$

$$\Updownarrow$$

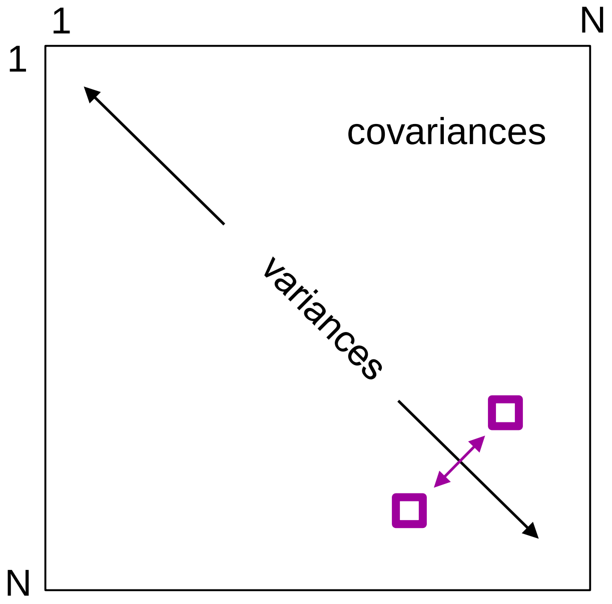
$$\Sigma_{ji} = \mathrm{cov}(x_j, x_i)$$

The covariance matrix **Σ**



The covariance matrix $\boldsymbol{\Sigma}$

$$\Sigma =$$



- Variances on the diagonal
- Covariances on the off-diagonal
- Symmetric matrix

$$\Sigma_{ij} = \Sigma_{ji}$$
$$\boldsymbol{\Sigma}^T = \boldsymbol{\Sigma}$$

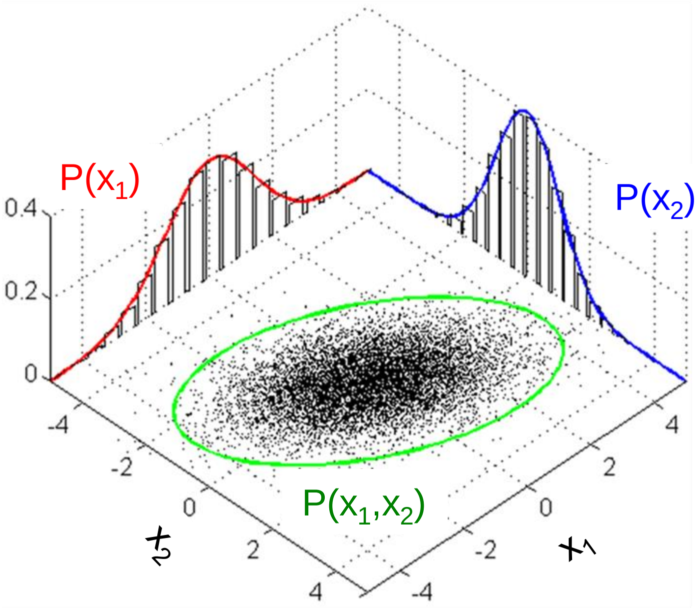
Multivariate normal distribution

Generalization of normal distribution to N dimensions

$$P(\mathbf{x}) \sim e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})}$$

- Parameters:
  $\boldsymbol{\mu}$ - mean of each variable
  $\boldsymbol{\Sigma}$ - covariance matrix

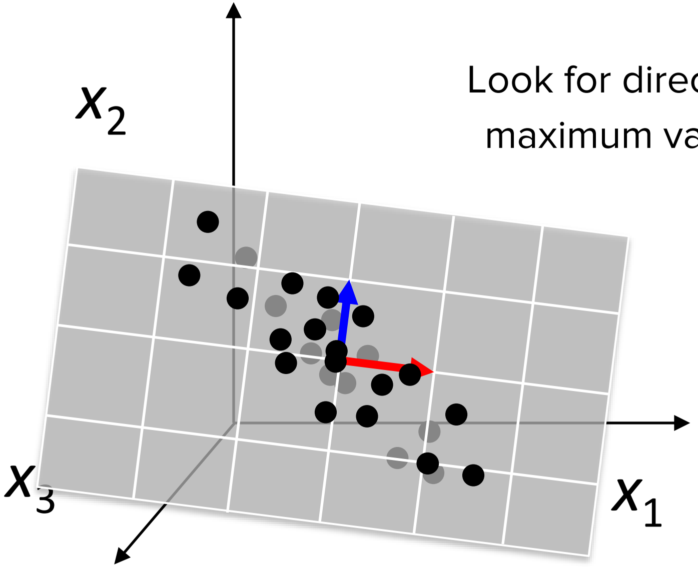
- Marginal distribution $P(x_i)$ is 1D Gaussian



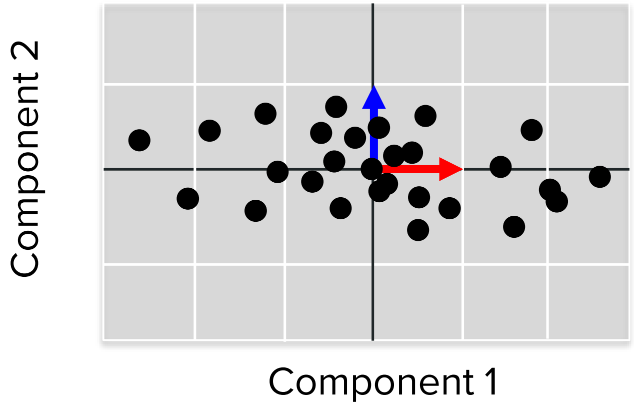
(break for tutorial exercise)

neuromatch academy

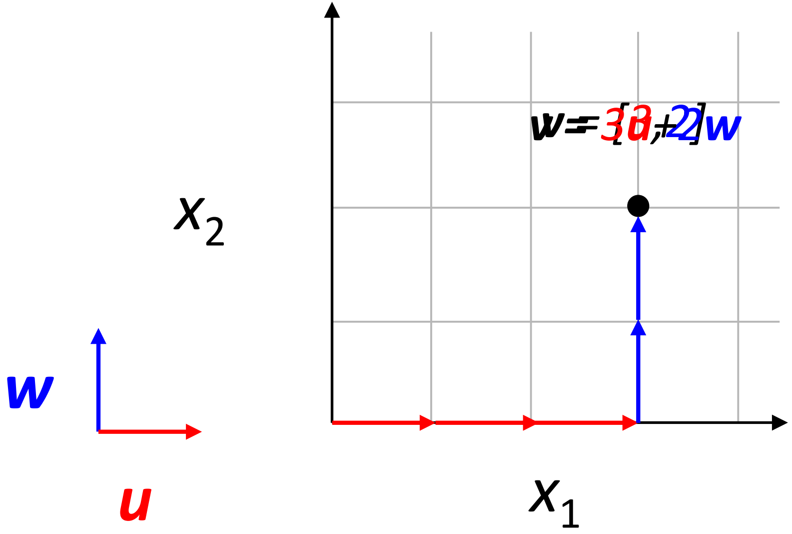
PCA: the big picture



Look for directions of maximum variance.



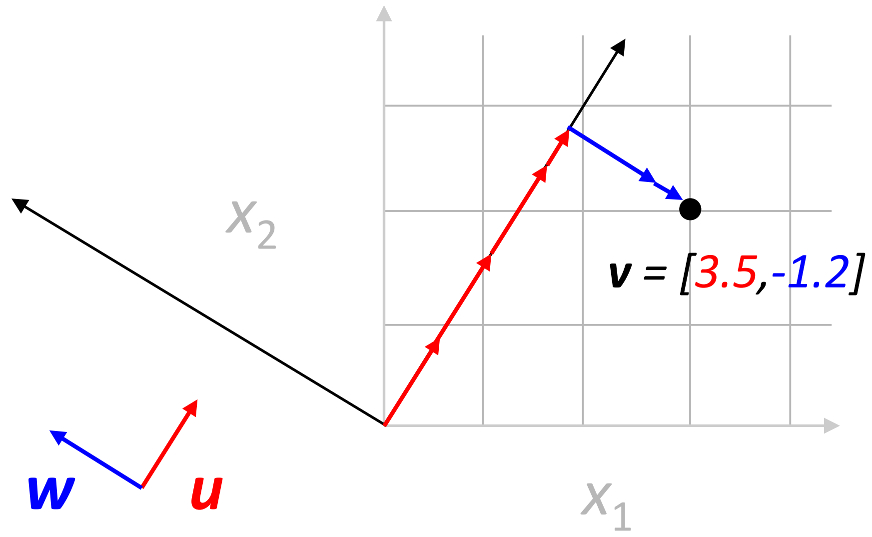
Many ways to represent multivariate data



Basis:

- Set of N vectors with which you can construct any point in the N-dimensional vector space.

Many ways to represent multivariate data



Basis:

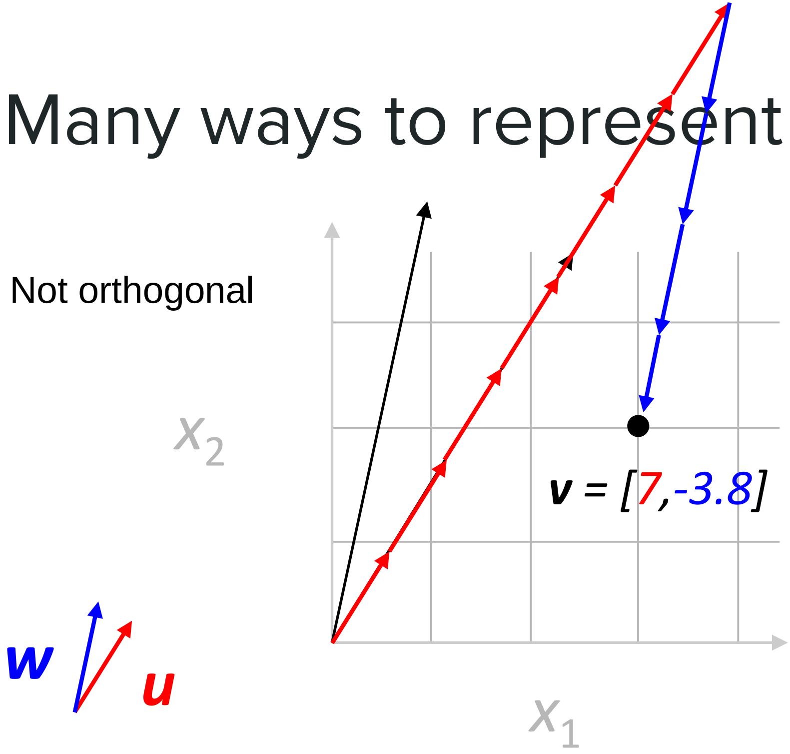
- Set of N vectors with which you can construct any point in the N-dimensional vector space.

Orthogonal basis

- All basis vectors are orthogonal.

Many ways to represent multivariate data

Not orthogonal



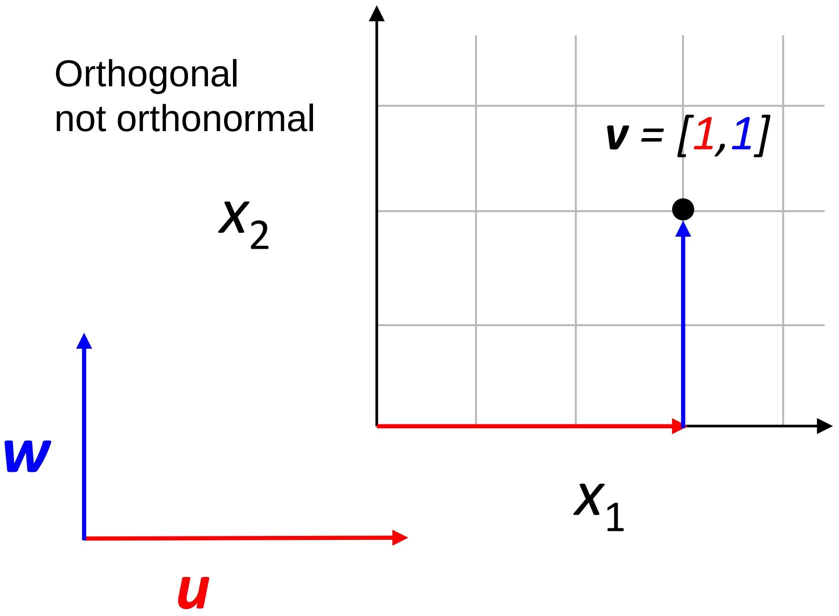
Basis:

- Set of N vectors with which you can construct any point in the N-dimensional vector space.

Orthogonal basis

- All basis vectors are orthogonal.

Many ways to represent multivariate data



Basis:

- Set of N vectors with which you can construct any point in the N-dimensional vector space.

Orthogonal basis

- All basis vectors are orthogonal.

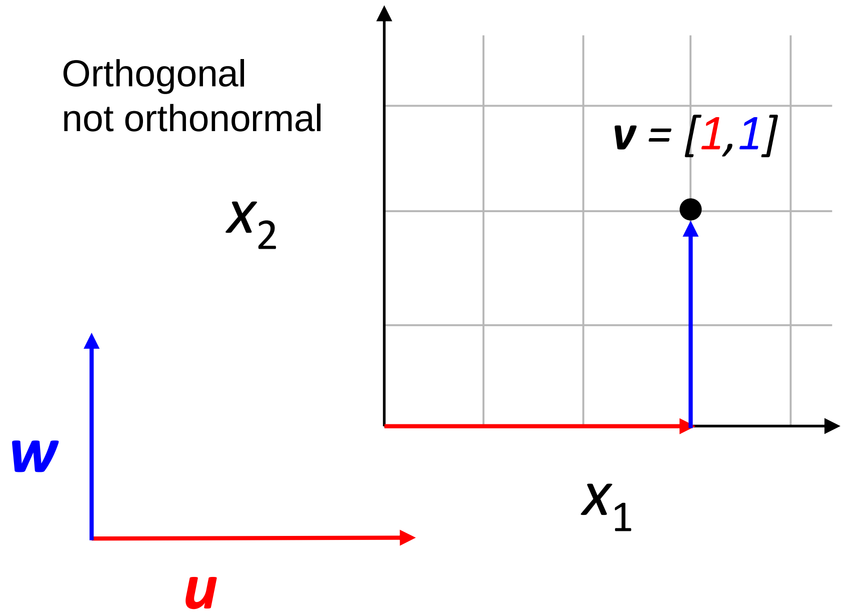
Orthonormal basis

- Orthogonal + all basis vectors have a length of 1.

$$\|\mathbf{u}\| = \|\mathbf{w}\| = 1$$

$$\|\mathbf{u}\| = \sqrt{u_1^2 + u_2^2}$$

Many ways to represent multivariate data



Basis:

- Set of N vectors with which you can construct any point in the N-dimensional vector space.

Orthogonal basis

- All basis vectors are orthogonal.

Orthonormal basis

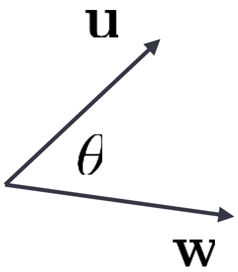
- Orthogonal + all basis vectors have a length of 1.

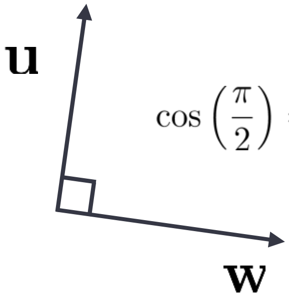
An orthogonal basis can easily be **normalized**:

$$\tilde{\mathbf{u}} = \frac{\mathbf{u}}{||\mathbf{u}||} \qquad \tilde{\mathbf{w}} = \frac{\mathbf{w}}{||\mathbf{w}||}$$

The dot product

$$\mathbf{u} \cdot \mathbf{w} = \|\mathbf{u}\| \, \|\mathbf{w}\| \cos(\theta)$$





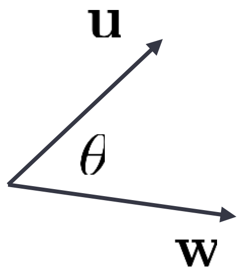
$$\cos\left(\frac{\pi}{2}\right) = 0$$

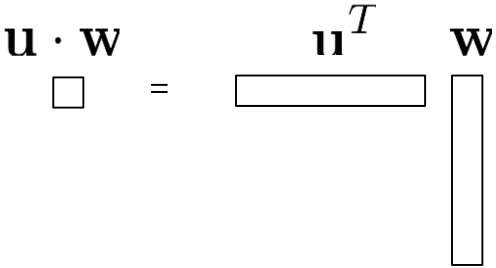
Orthogonal vectors ⟺ Dot product is zero

The dot product

$$\mathbf{u} \cdot \mathbf{w} = \|\mathbf{u}\| \, \|\mathbf{w}\| \cos(\theta)$$

$$\mathbf{u} \cdot \mathbf{w} = \sum_{i=1}^{N} u_i w_i$$





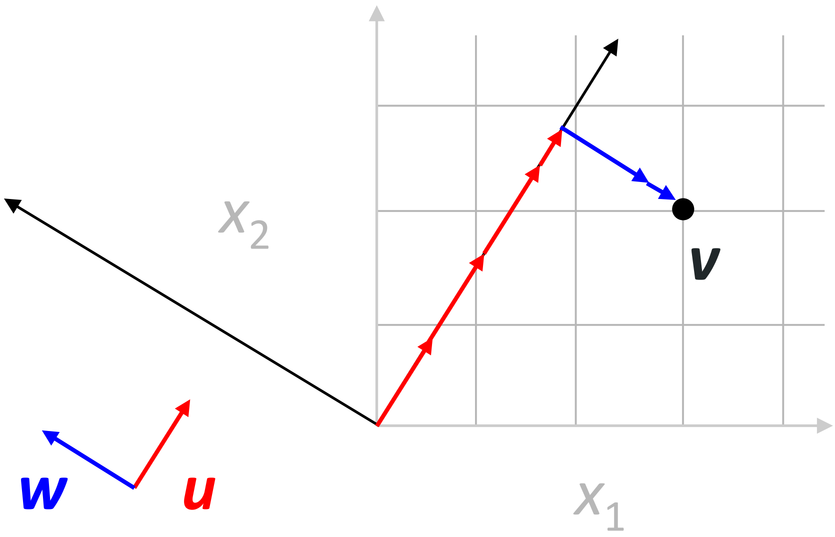
Orthogonal vectors ⟺ Dot product is zero

$$\mathbf{u}^T \mathbf{w} = 0$$

(break for tutorial exercise)

neuromatch academy

Change of basis



$v = ?$

$[3,2]$ → $[3.5,-1.2]$

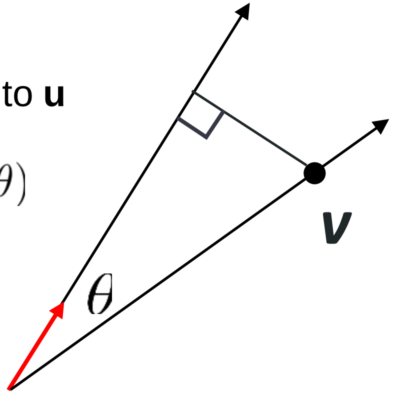
Standard basis   New basis

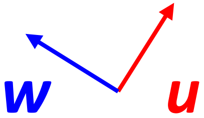
How do we transform coordinates to a new orthonormal basis?

Change of basis via projection

Project **v** onto **u**

$$\|\mathbf{v}\| \cos(\theta)$$





$$\begin{aligned} \mathbf{u} \cdot \mathbf{v} &= \|\mathbf{u}\| \|\mathbf{v}\| \cos(\theta) \\ &= \|\mathbf{v}\| \cos(\theta) \\ &= \mathbf{u}^T \mathbf{v} \quad = \mathbf{v}^T \mathbf{u} \end{aligned}$$

New coordinates from dot product!

Projection to orthonormal basis

For an orthonormal basis,
new coordinates are

old coordinates • new basis vector

$$\boxed{y_1} = \boxed{x_1 \ x_2}\,\boxed{\mathbf{u}}$$

$$\boxed{y_2} = \boxed{x_1 \ x_2}\,\boxed{\mathbf{w}}$$

Projection to orthonormal basis

For an orthonormal basis,
new coordinates are

old coordinates • new basis vector

$$\boxed{y_1\ y_2} = \boxed{x_1\ x_2}\ \boxed{\begin{matrix} \color{red}{u} & \color{blue}{w} \end{matrix}}$$

Projection to orthonormal basis

For an orthonormal basis,
new coordinates are

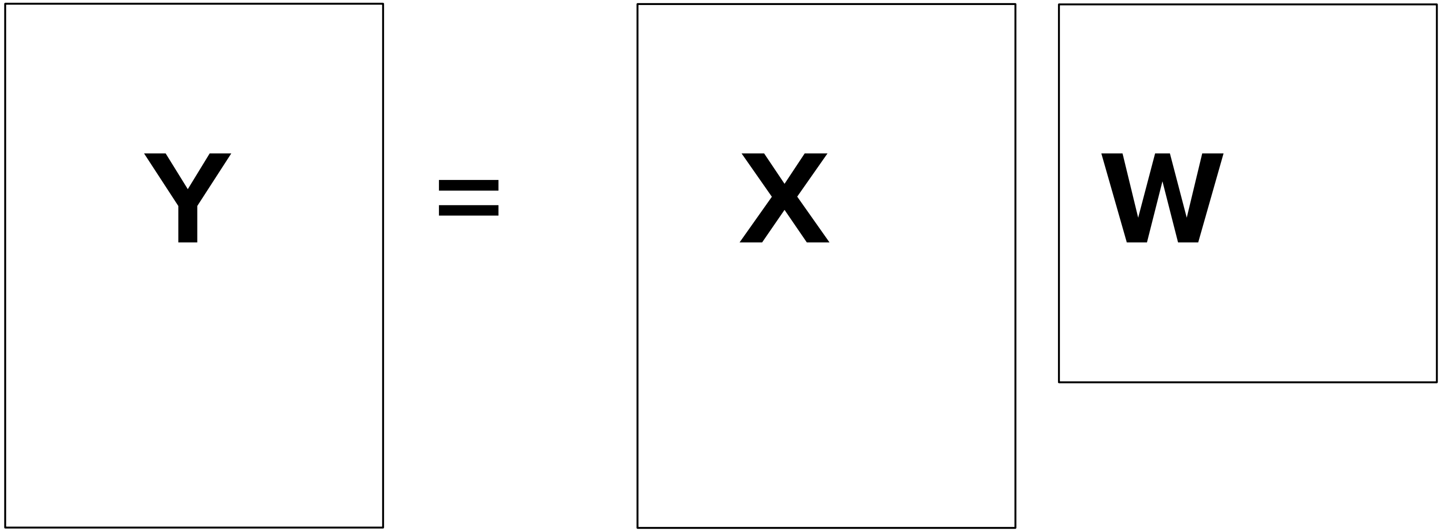
old coordinates • new basis vector

$$\mathbf{Y} = \mathbf{X} \quad \mathbf{W}$$

$$\begin{bmatrix} y_1 \; y_2 \\ \\ \\ \end{bmatrix} = \begin{bmatrix} x_1 \; x_2 \\ \\ \\ \end{bmatrix} \begin{bmatrix} \boldsymbol{u} & \boldsymbol{w} \end{bmatrix}$$

adding more samples…

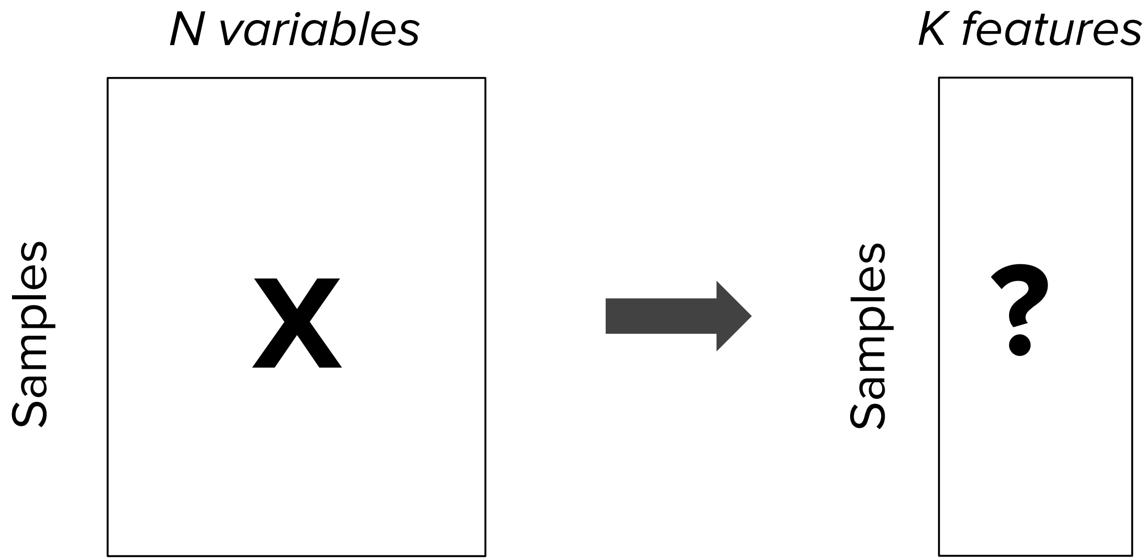
Projection to orthonormal basis



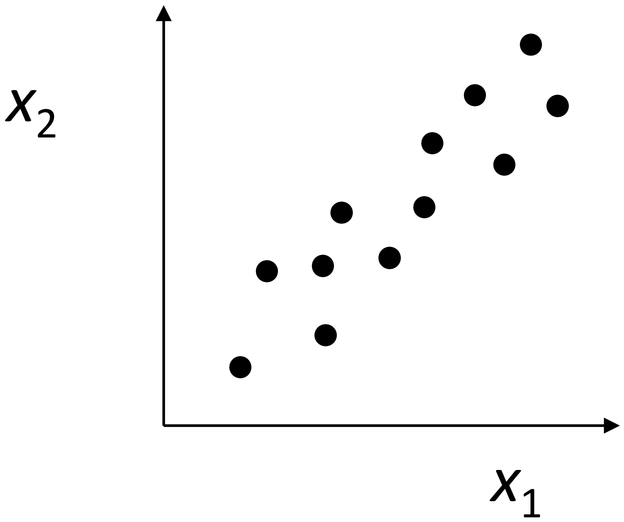
Principal components analysis

Tutorial 2

Goal of dimensionality reduction



Covariance reveals structure

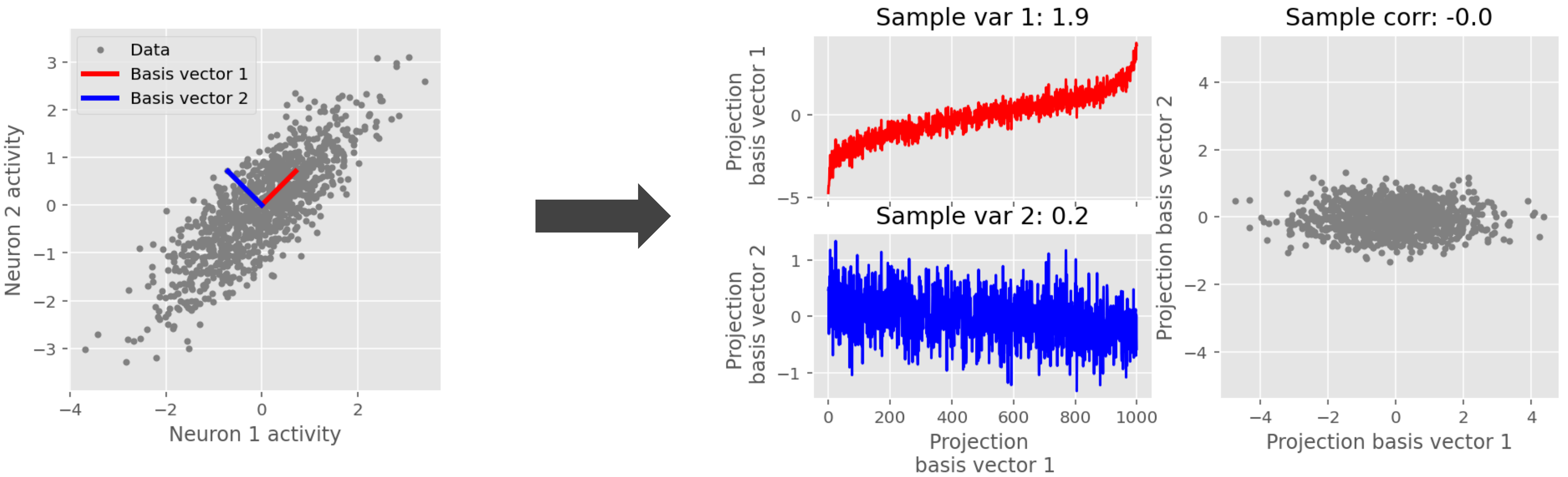


Expectation:

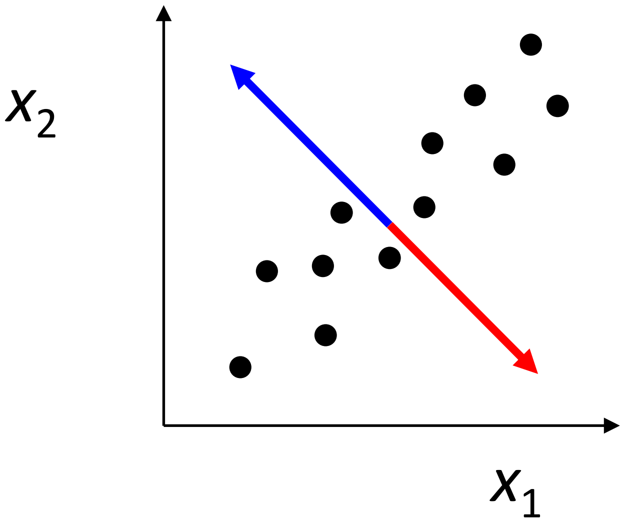
- Directions of large variation represent signal.
- Directions of small variation represent noise.

Goal of PCA: Find directions of maximum variance.

Projected variance is largest when basis is aligned with covariance direction



Covariance reveals structure



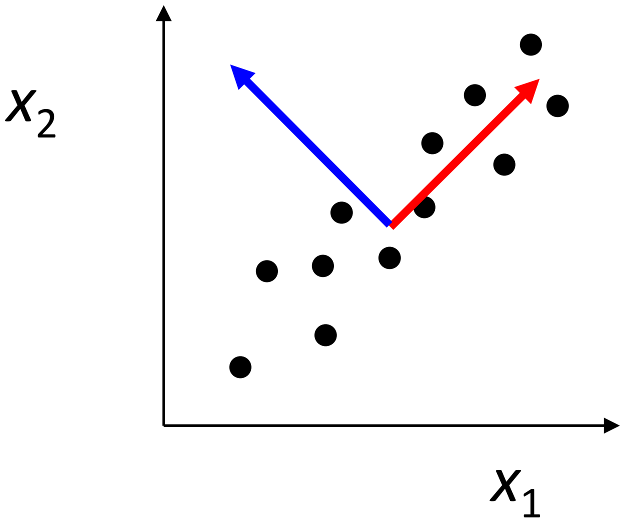
Expectation:

- Directions of large variation represent signal.
- Directions of small variation represent noise.

Goal of PCA: Find directions of maximum variance.

- $\mathbf{w}_1$ – vector that has highest projected variance
- $\mathbf{w}_2$ – vector that is orthogonal to $\mathbf{w}_1$ and has highest projected variance
- Etc.

Covariance reveals structure



Expectation:

- Directions of large variation represent signal.
- Directions of small variation represent noise.

Goal of PCA: Find directions of maximum variance.

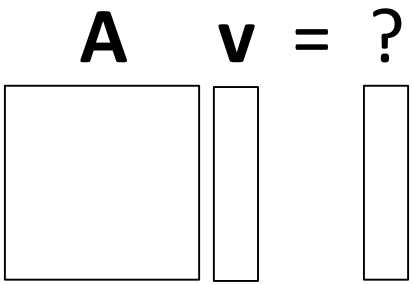
- $\mathbf{w}_1$ – vector that has highest projected variance
- $\mathbf{w}_2$ – vector that is orthogonal to $\mathbf{w}_1$ and has highest projected variance
- Etc.

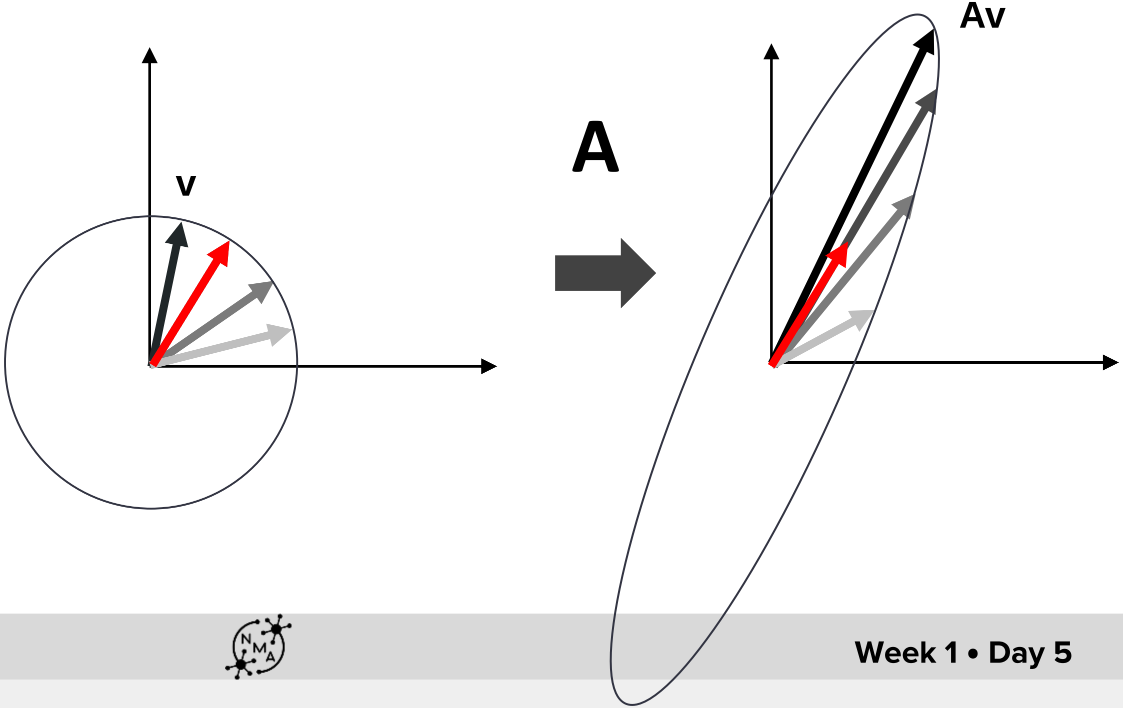
Mathematical solution:

- $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_N$ are the eigenvectors of $\mathbf{\Sigma}$.

Eigenvalue refresher

A matrix is a linear transformation



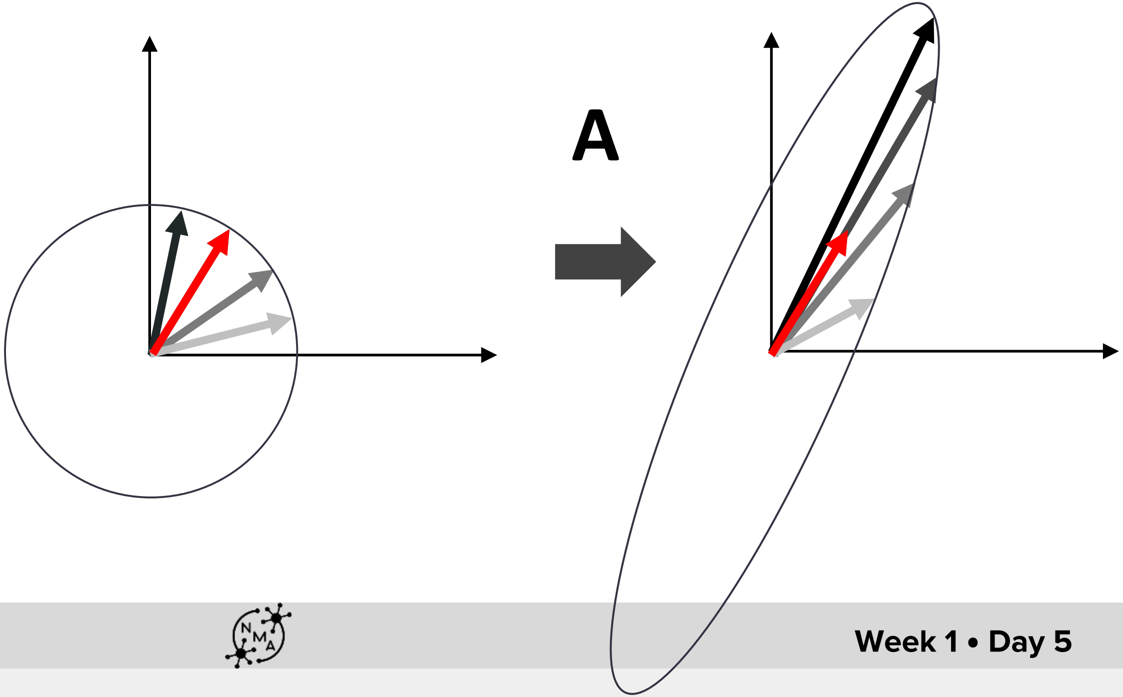


Eigenvalue refresher

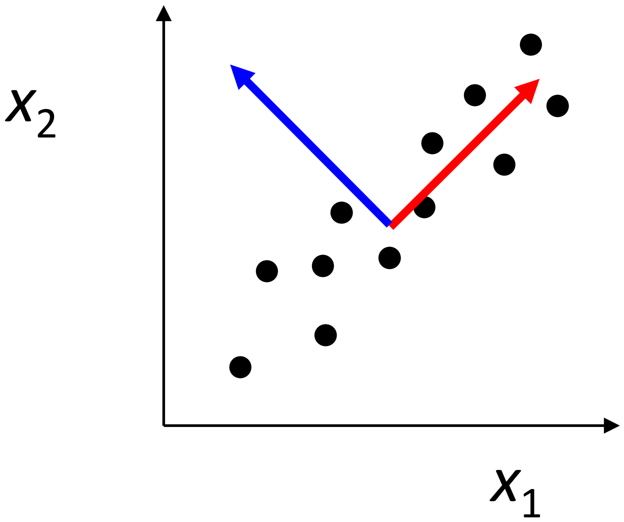
A matrix is a linear transformation

$$\mathbf{A}\ \mathbf{v} = \lambda\ \mathbf{v}$$

λ: eigenvalue
v: eigenvector



Covariance reveals structure



Expectation:

- Directions of large variation represent signal.
- Directions of small variation represent noise.

Goal of PCA: Find directions of maximum variance.

- $\mathbf{w}_1$ – vector that has highest projected variance
- $\mathbf{w}_2$ – vector that is orthogonal to $\mathbf{w}_1$ and has highest projected variance
- Etc.

Mathematical solution:

- $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_N$ are the eigenvectors of $\mathbf{\Sigma}$.
- Projected variance onto each $\mathbf{w}_i$ is given by its corresponding eigenvalue λ_i.

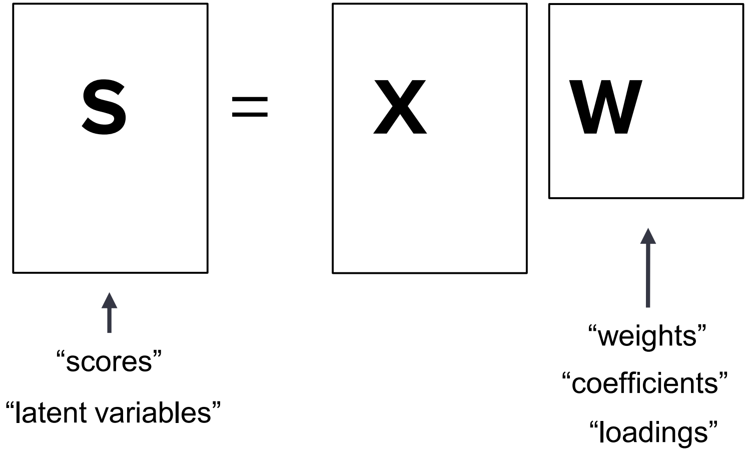
How to perform PCA

Basic algorithm

1. Subtract the mean
2. Calculate the eigenvectors $\mathbf{w}_i$ of the covariance matrix $\mathbf{\Sigma}$, ordered by their corresponding eigenvalue λ_i.
3. Project the data $\mathbf{X}$ onto the new basis $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_N$.

Key properties

- $\mathbf{w}_i$ are orthogonal
- $\mathbf{s}_i$ are uncorrelated
- Projected variance = λ_i



(break for tutorial exercise)

neuromatch academy

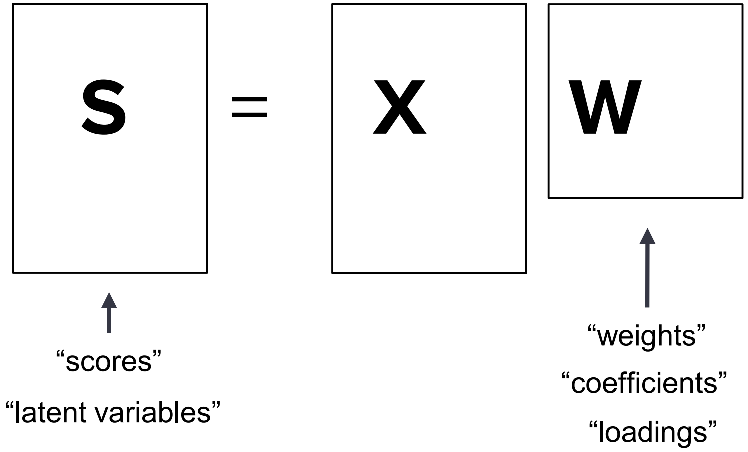
How to perform PCA

Basic algorithm

1. Subtract the mean
2. Calculate the eigenvectors $\mathbf{w}_i$ of the covariance matrix $\mathbf{\Sigma}$, ordered by their corresponding eigenvalue λ_i.
3. Project the data $\mathbf{X}$ onto the new basis $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N$.

Key properties

- $\mathbf{w}_i$ are orthogonal
- $\mathbf{s}_i$ are uncorrelated
- Projected variance = λ_i



$\mathbf{w}_i$ are orthogonal

What we know

- Since $\mathbf{w}_i$ eigenvectors of the covariance matrix

$$\hat{\mathbf{\Sigma}}\mathbf{w}_i = \lambda_i \mathbf{w}_i$$

- Covariance matrix is symmetric

$$\hat{\mathbf{\Sigma}} = \hat{\mathbf{\Sigma}}^T$$

What we want to show

$$\mathbf{w}_i \cdot \mathbf{w}_j = 0$$

$$\lambda_j \mathbf{w}_i^T \mathbf{w}_j$$

$$\Downarrow$$

$$(\lambda_j - \lambda_i)\mathbf{w}_i \cdot \mathbf{w}_j = 0$$

If the eigenvalues are different, then the eigenvectors must be orthogonal.

$\mathbf{s}_i$ are uncorrelated

What we know

- Since $\mathbf{w}_i$ eigenvectors of the covariance matrix
$$\hat{\mathbf{\Sigma}}\mathbf{w}_i = \lambda_i \mathbf{w}_i$$
- Scores $\mathbf{s}_i$ represent the projected data
$$\mathbf{s}_i = \mathbf{X}\mathbf{w}_i$$
- Since $\mathbf{X}$ is zero-mean, $\mathbf{s}_i$ is zero-mean
$$\bar{\mathbf{s}}_i = 0$$

What we want to show

$$\text{cov}(\mathbf{s}_i, \mathbf{s}_j) = 0$$

$$\text{cov}(\mathbf{s}_i, \mathbf{s}_j)$$

The scores (projected data) are uncorrelated.

λ_i describe projected variances

What we know

- Since $\mathbf{w}_i$ eigenvectors of the covariance matrix
$$\hat{\mathbf{\Sigma}}\mathbf{w}_i = \lambda_i \mathbf{w}_i$$
- Scores $\mathbf{s}_i$ represent the projected data
$$\mathbf{s}_i = \mathbf{X}\mathbf{w}_i$$
- Since $\mathbf{X}$ is zero-mean, $\mathbf{s}_i$ is zero-mean
$$\bar{\mathbf{s}}_i = 0$$

What we want to show

$$\mathrm{var}(\mathbf{s}_i) = \lambda_i$$

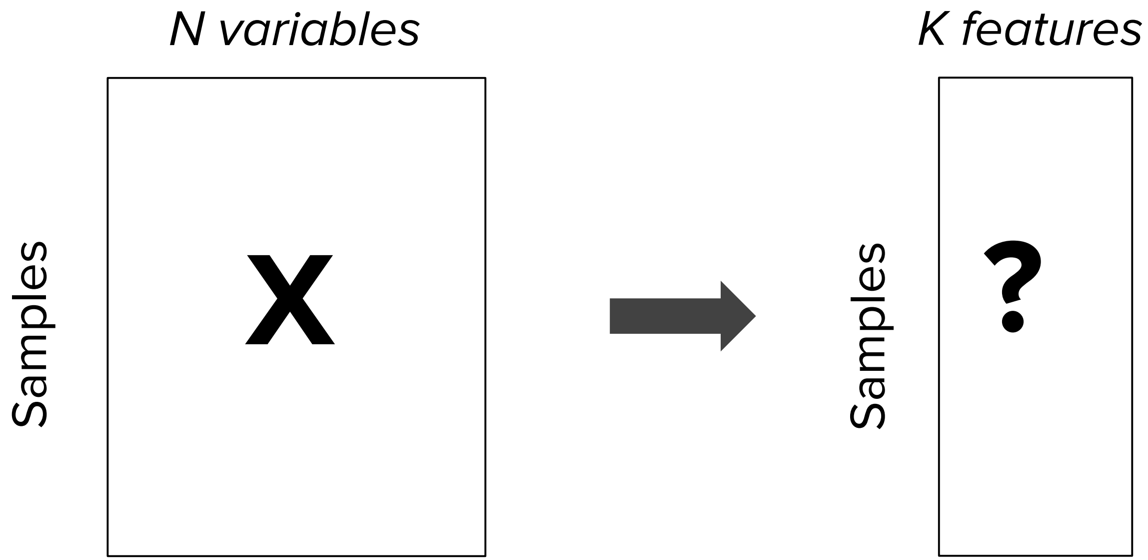
$$\begin{aligned}
\mathrm{var}(\mathbf{s}_i) &= \frac{1}{N_{\mathrm{samples}}}\mathbf{s}_i^T\mathbf{s}_i - \bar{\mathbf{s}}_i^2 \\
&= \frac{1}{N_{\mathrm{samples}}}\mathbf{s}_i^T\mathbf{s}_i \\
&= \frac{1}{N_{\mathrm{samples}}}(\mathbf{X}\mathbf{w}_i)^T\mathbf{X}\mathbf{w}_i \\
&= \frac{1}{N_{\mathrm{samples}}}\mathbf{w}_i^T\mathbf{X}^T\mathbf{X}\mathbf{w}_i \\
&= \mathbf{w}_i^T\hat{\mathbf{\Sigma}}\mathbf{w}_i \\
&= \lambda_i\mathbf{w}_i^T\mathbf{w}_i
\end{aligned}$$

The variance of the scores (projected data) is its corresponding eigenvalue.

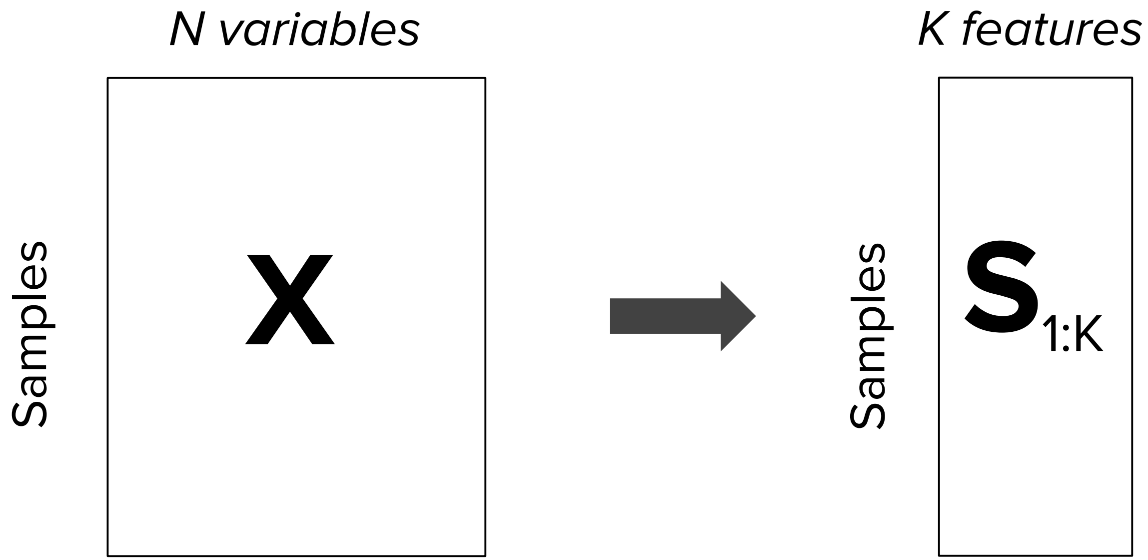
Dimensionality reduction and reconstruction

Tutorial 3

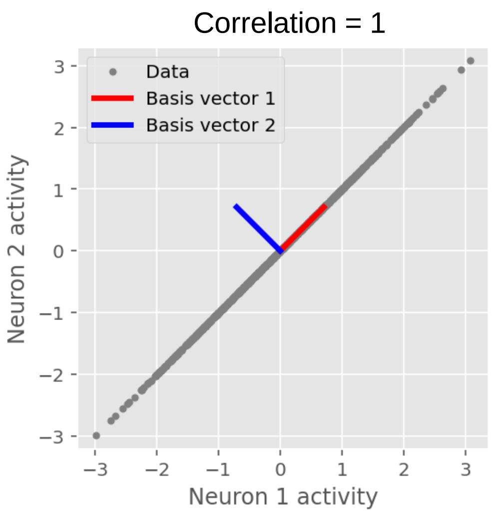
Goal of dimensionality reduction

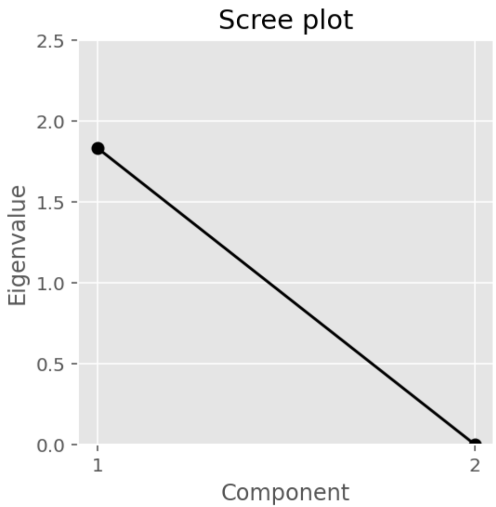


Dimensionality reduction via PCA



Intrinsic vs. extrinsic dimensionality

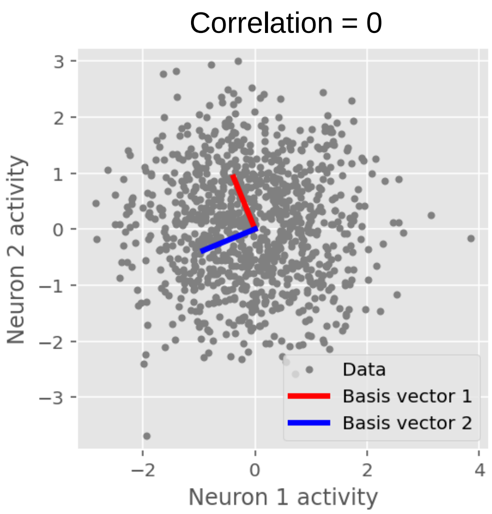


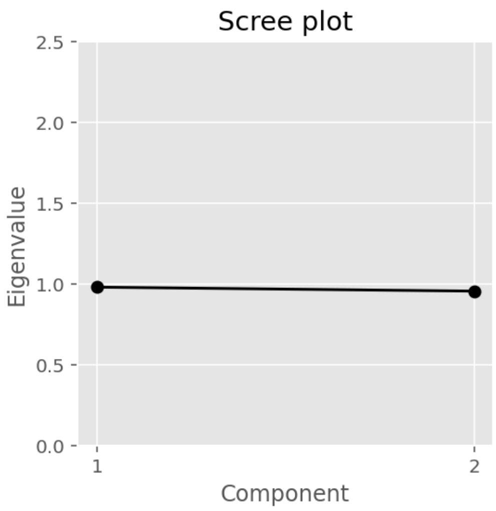


Extrinsic dimensionality: N = 2

Intrinsic dimensionality: K = 1

Intrinsic vs. extrinsic dimensionality

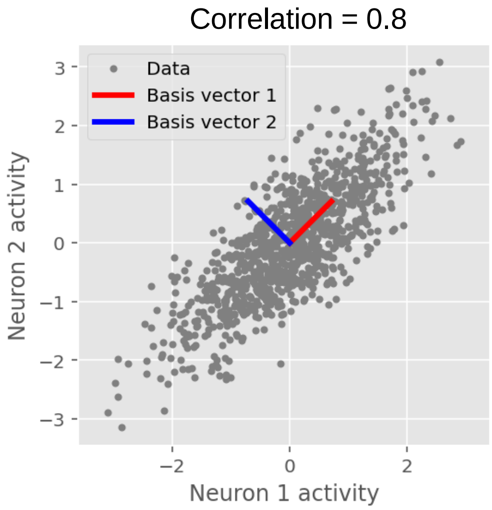


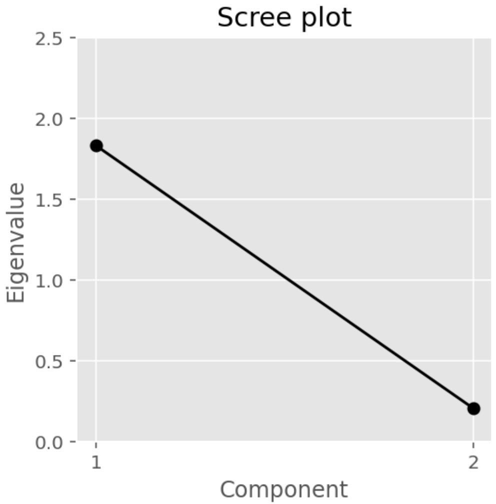


Extrinsic dimensionality: N = 2

Intrinsic dimensionality: K = 2

Intrinsic vs. extrinsic dimensionality

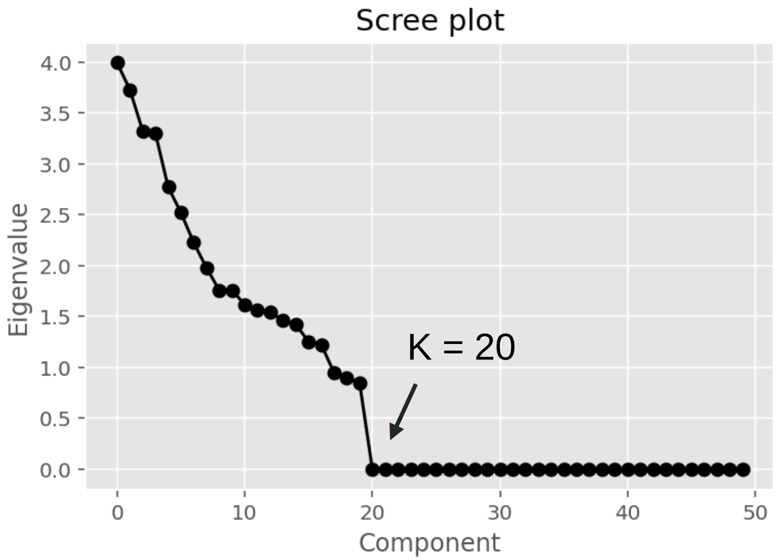




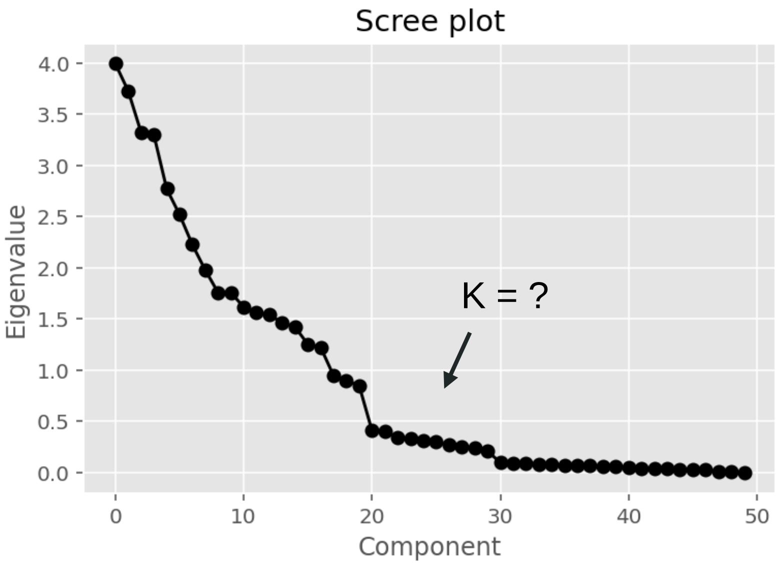
Extrinsic dimensionality: N = 2

Intrinsic dimensionality: K = ?

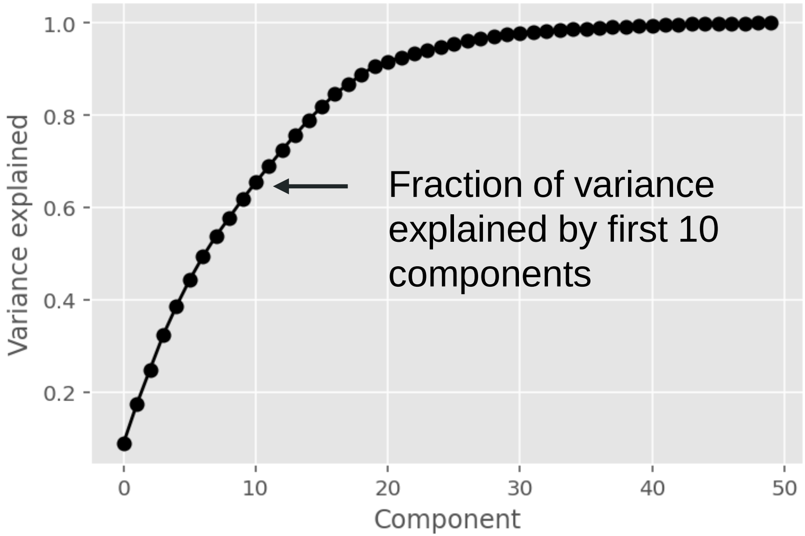
How to determine intrinsic dimensionality?



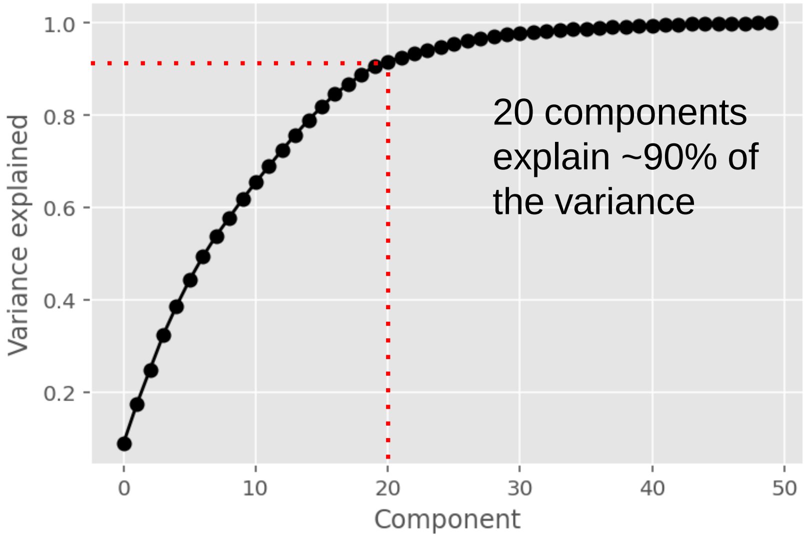
How to determine intrinsic dimensionality?



Total variance explained



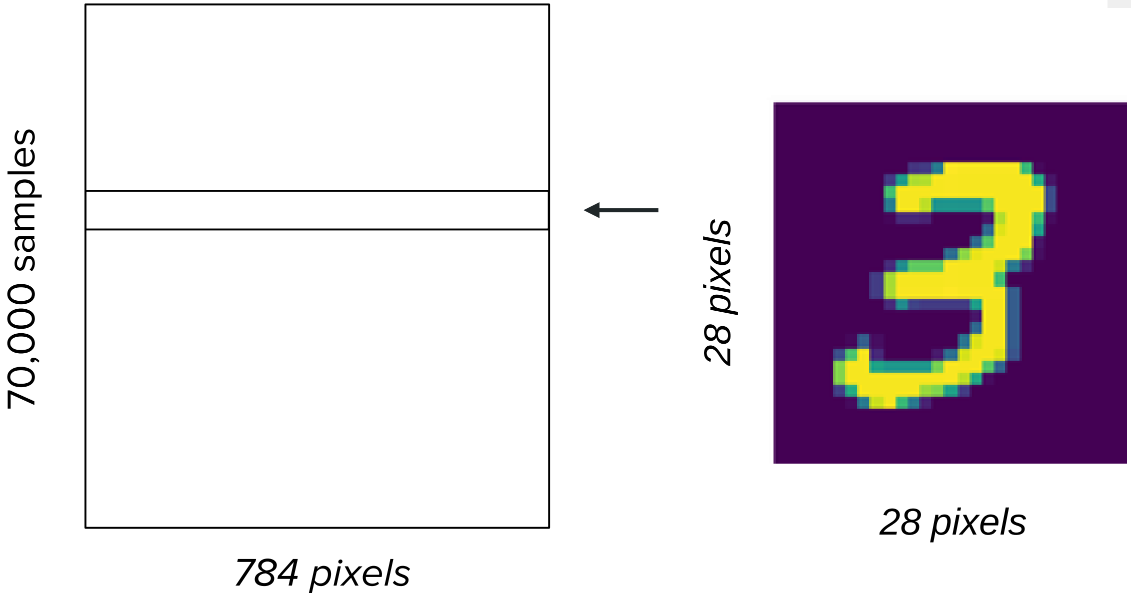
Total variance explained



The MNIST dataset

The MNIST dataset

$\mathbf{X} =$



(break for tutorial exercise)

neuromatch academy

Reconstruction from PCA

Once we have **S** and **W** how do we reconstruct **X** ?

$$\mathbf{S} = \mathbf{X} \; \mathbf{W}$$

Algorithm for PCA

1. Subtract the mean
2. Calculate the eigenvectors $\mathbf{w}_i$ of the covariance matrix $\boldsymbol{\Sigma}$, ordered by their corresponding eigenvalue λ_i.
3. Project the data **X** onto the new basis $\mathbf{w}_{1,}$ $\mathbf{w}_2$, …, $\mathbf{w}_N$.

Reconstruction from PCA

Once we have **S** and **W** how do we reconstruct **X** ?

$$\mathbf{S}\,\mathbf{W}^T = \mathbf{X}\,\underbrace{\mathbf{W}\,\mathbf{W}^T}_{= \text{Identity matrix}}$$

because $\mathbf{w}_i$ are orthonormal basis

Reconstruction from PCA

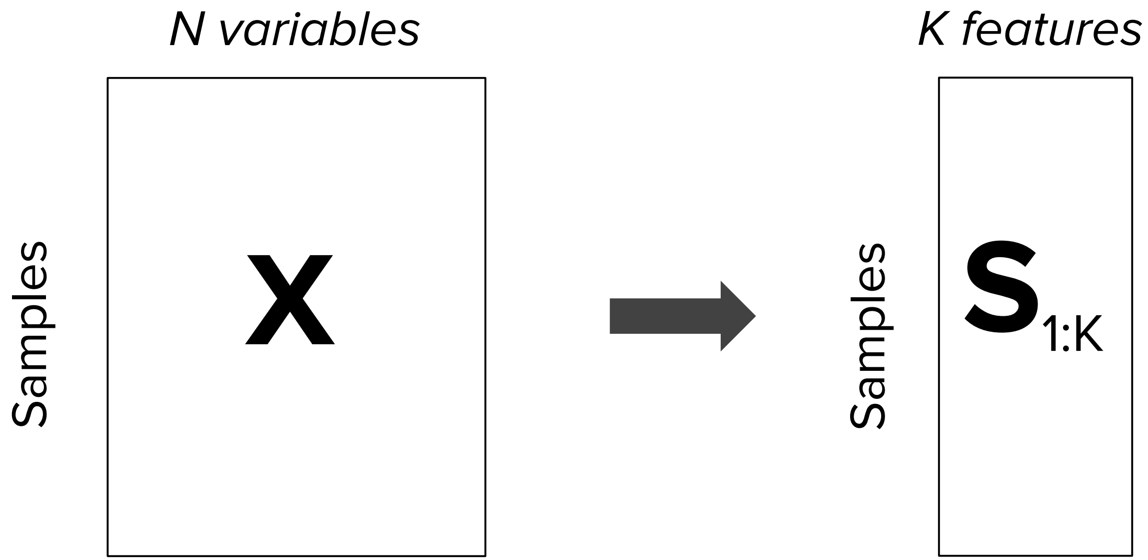
Once we have **S** and **W** how do we reconstruct **X** ?

$$\mathbf{S} \; \mathbf{W}^{T} = \mathbf{X}$$

Algorithm for reconstruction from PCA

1. Multiply scores by transpose of the weight matrix
2. Add the mean

Dimensionality reduction via PCA



Reconstruction from PCA

Once we have **S** and **W** how do we reconstruct **X** ?

$$\mathbf{S}_{1:K} \, (\mathbf{W}_{1:K})^T = \hat{\mathbf{X}}$$

Algorithm for reconstruction from PCA

1. Truncate scores and weight matrix after top K components
2. Multiply scores by transpose of the weight matrix
3. Add the mean

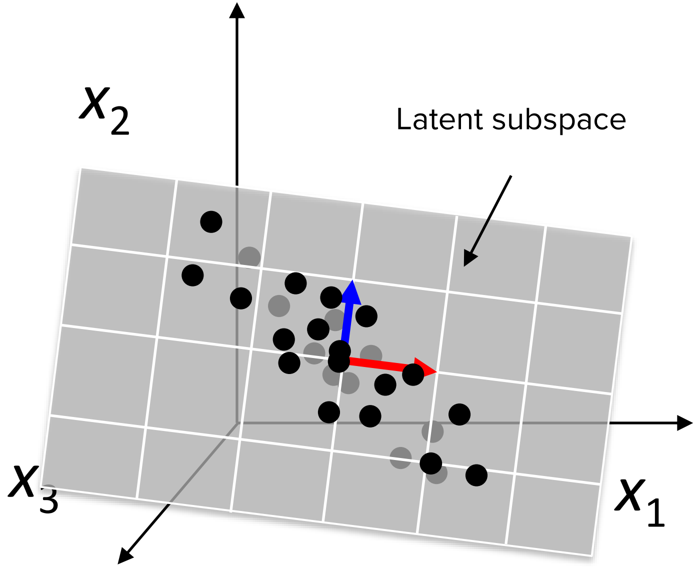
Goal of PCA: Find K-dimensional basis that minimizes the reconstruction error.

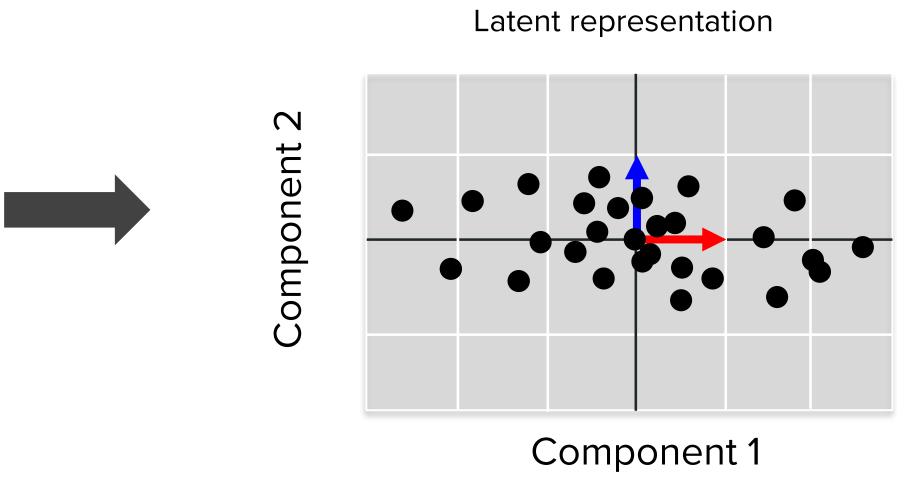
$$\|\mathbf{X} - \hat{\mathbf{X}}\|^2$$

Nonlinear dimensionality reduction

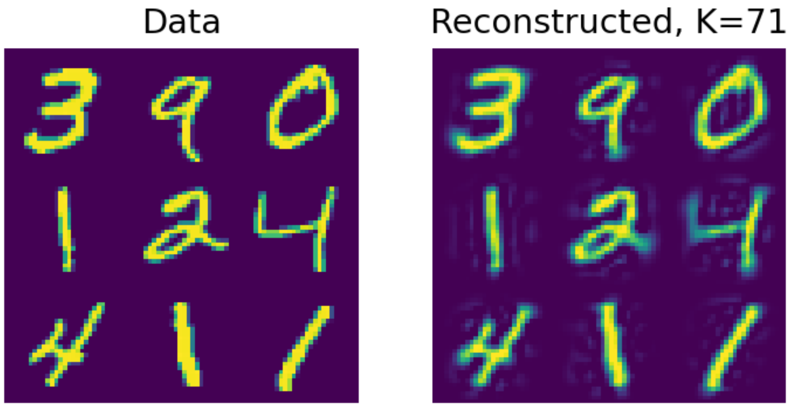
Tutorial 4

PCA: the big picture



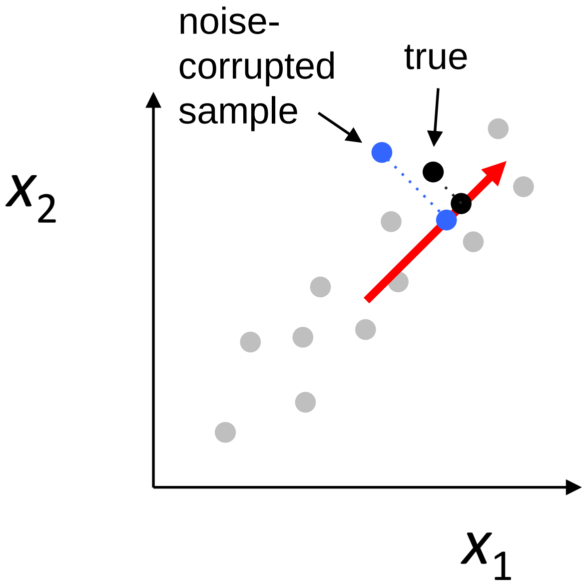


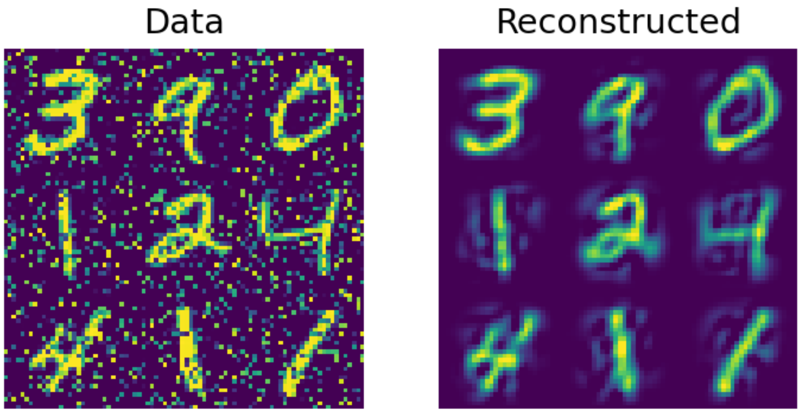
PCA for compression



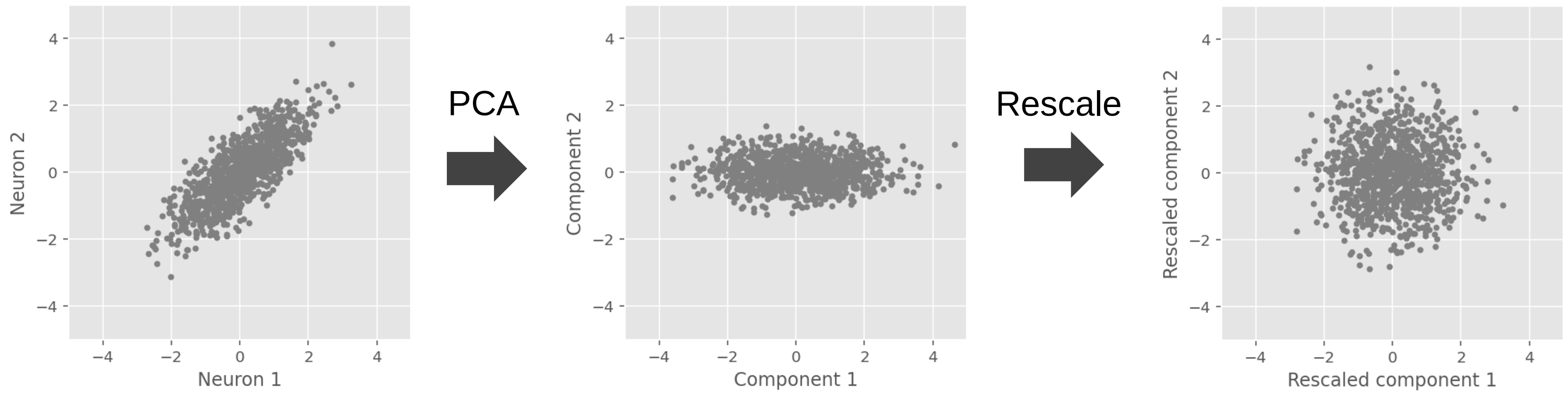
$$\mathbf{S}_{1:K} (\mathbf{W}_{1:K})^T \approx \mathbf{X}$$

PCA for denoising

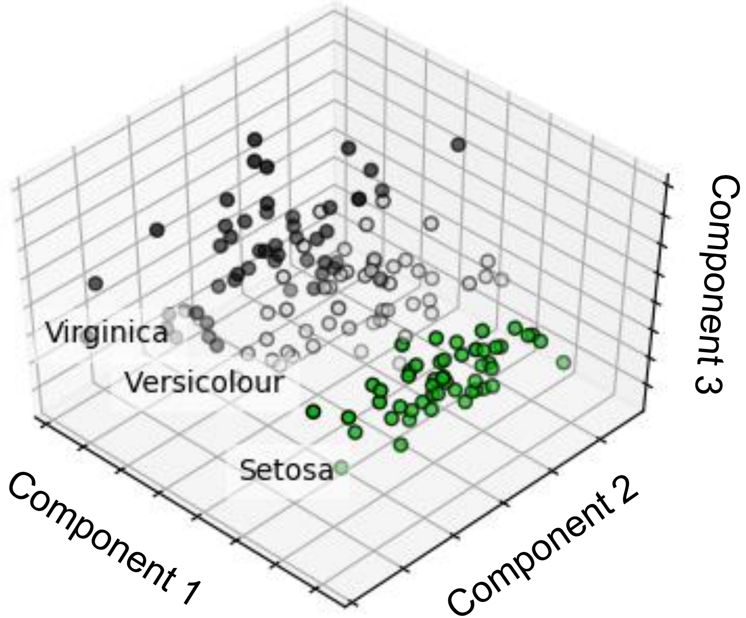




PCA for whitening

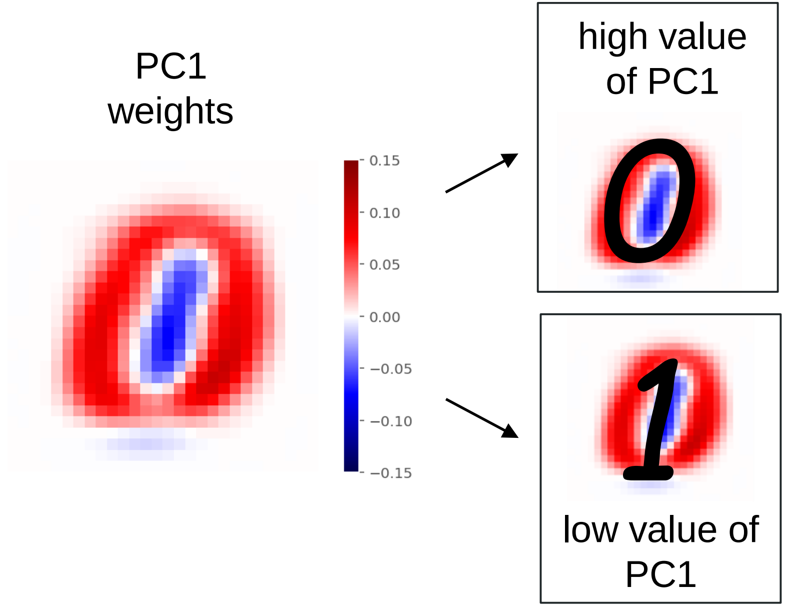


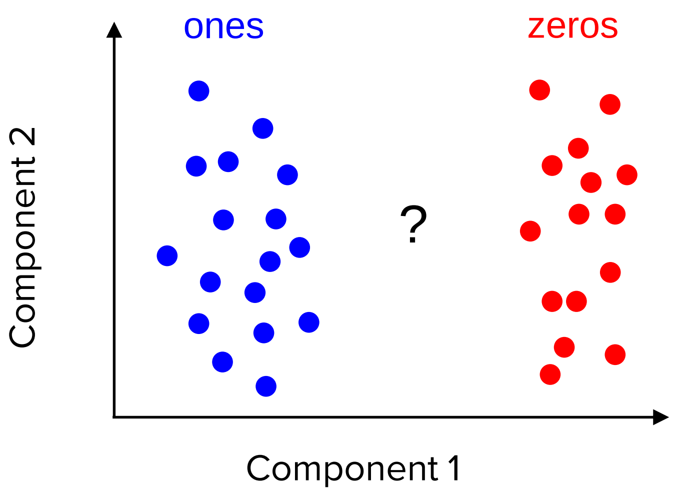
PCA for visualization



Source: scikit-learn.org

Visualizing MNIST





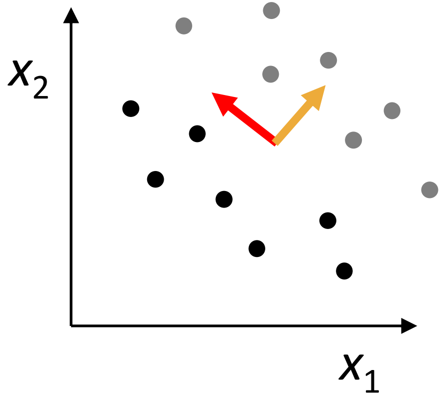
(break for tutorial exercise)

neuromatch academy

Linear dimensionality reduction

Linear transformation to lower dimensional representation **Y**

$$\mathbf{Y} = \mathbf{X}\ \mathbf{W}$$



Probabilistic PCA (PPCA):

- Explicit noise model $\mathbf{x}|\mathbf{y} \sim \mathcal{N}(\mathbf{y}\mathbf{W}, \sigma_\epsilon^2 \mathbf{I})$

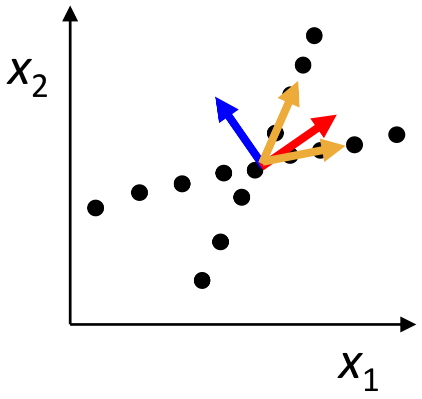
Factor Analysis (FA):

- Non-isotropic noise $\mathbf{x}|\mathbf{y} \sim \mathcal{N}(\mathbf{y}\mathbf{W}, \mathbf{D})$

Linear discriminant analysis (LDA) :

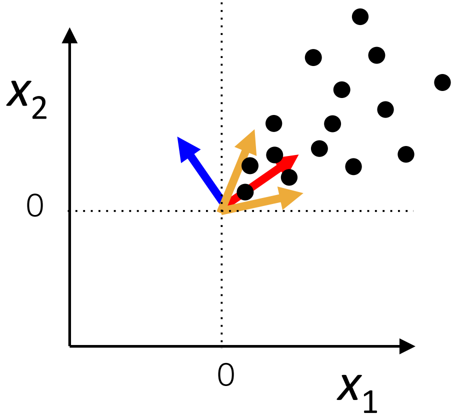
- Preserve class discriminatory information
- Example of **supervised** dimensionality reduction

Blind source separation



Independent Components Analysis (ICA) :

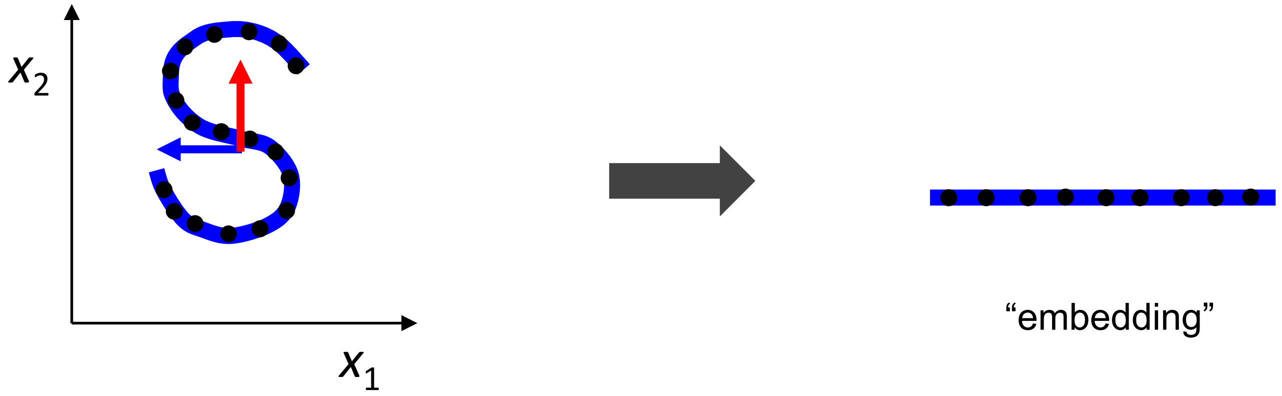
- Stronger condition than uncorrelated
- Basis vectors not necessarily orthogonal
- Components not ordered by importance



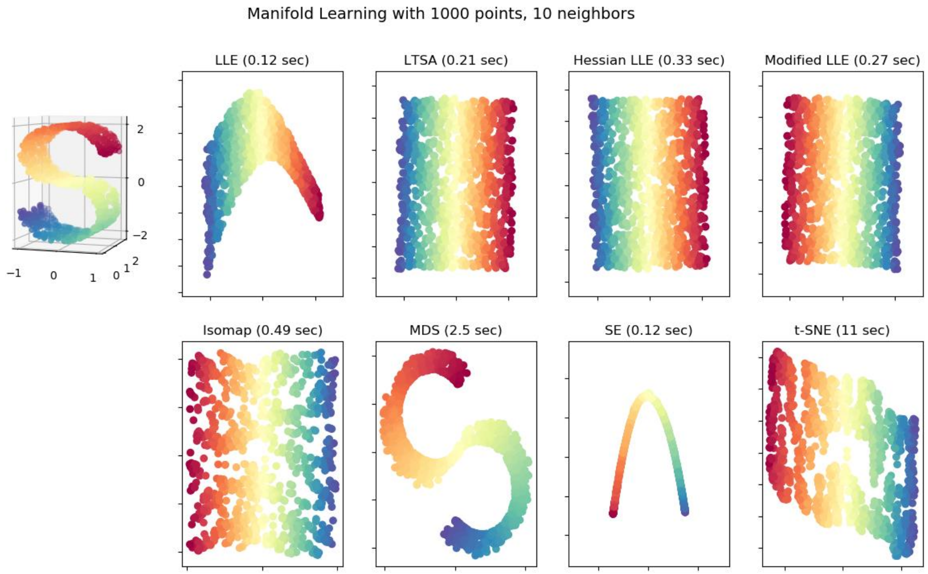
Nonnegative Matrix Factorization (NMF) :

- Weights and components positive
- Basis vectors not necessarily orthogonal
- No linear mapping to low-D space

When is linear not enough?



Nonlinear dimensionality reduction



Source: scikit-learn.org

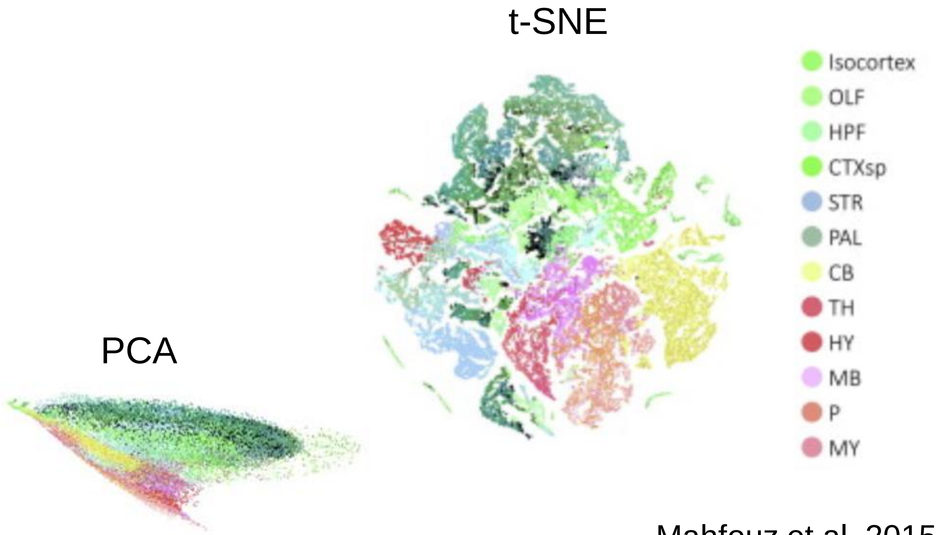
t-distributed stochastic neighbor embedding

t-SNE

- Visualization in 2D or 3D
- Define similarity between samples **X**
- Find a mapping to low-dimensional **Y** that preserves similarities as much as possible

Differences from PCA

- Nonlinear
- Stochastic
- No reconstruction
- Free parameter: *perplexity*



Mahfouz et al. 2015

(break for tutorial exercise)

neuromatch academy