Principal component analysis

UE de Master 2, AOS1
Fall 2022

S. Rousseau

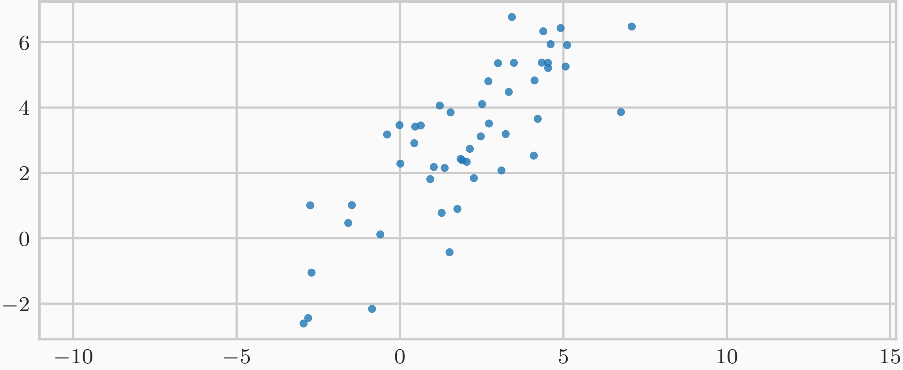
What is PCA?

Unsupervised multivariate technique for dimensionality reduction

- Developed by Pearson 1901
- Multipurpose technique:
    - Dimension reduction
    - Visualization
    - Decorrelation
    - Classification
    - Identifying underlying factors
    - Compression
    - Denoising

PCA in a nutshell

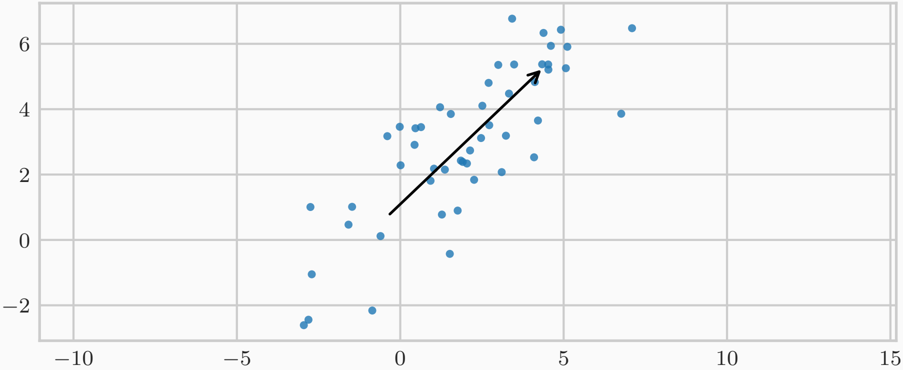
- Suppose we have a 2-dimensional dataset (design matrix X)



- Underlying data show a linear nature

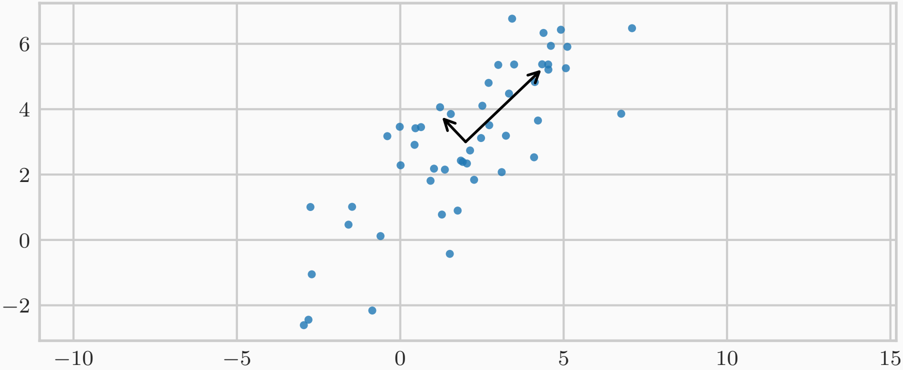
PCA in a nutshell

- PCA computes that linear nature (called first principal direction v_1)



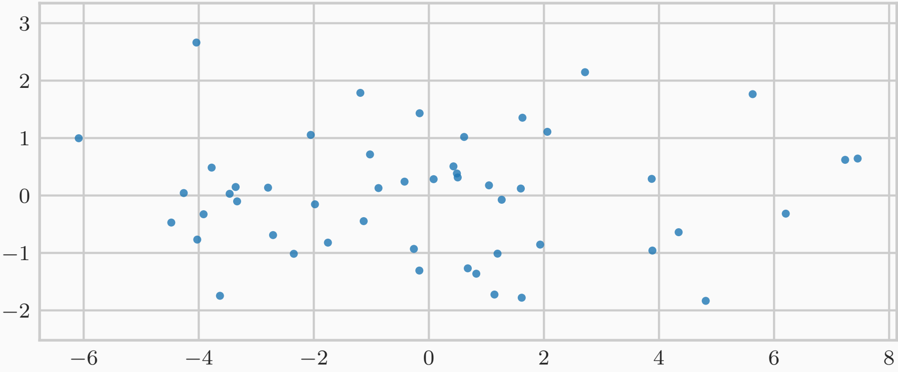
PCA in a nutshell

- Iterate on orthogonal space (second principal direction v_2)



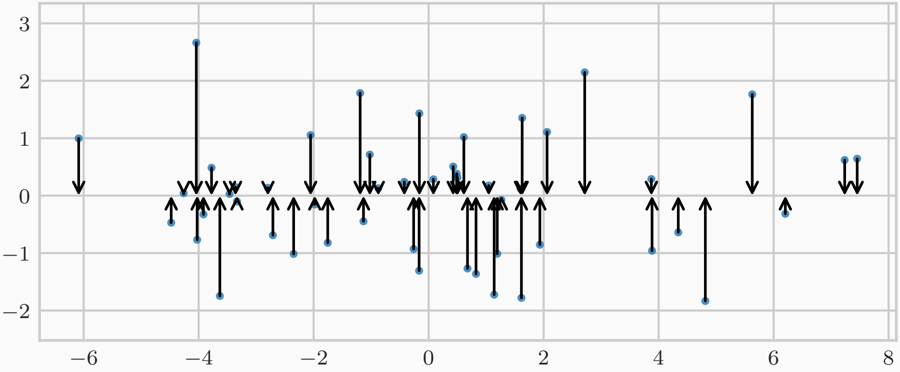
PCA in a nutshell

- Principal directions yields a new representation basis (new design matrix C)



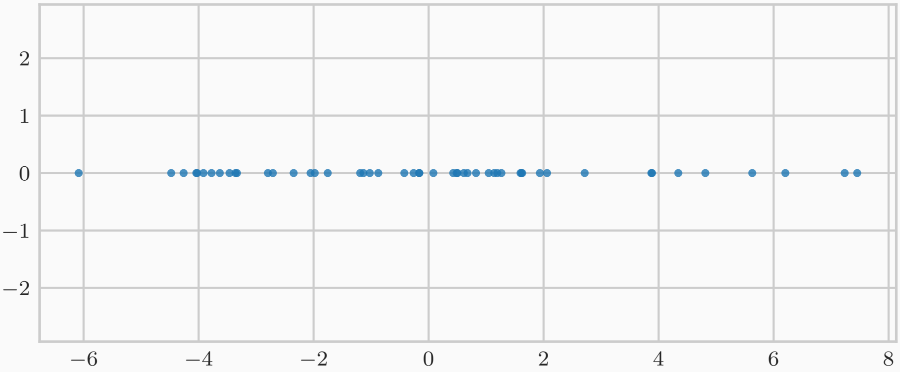
PCA in a nutshell

- Dimensionality reduction by orthogonal projection (selecting only first principal component $\boldsymbol{c}_1$)



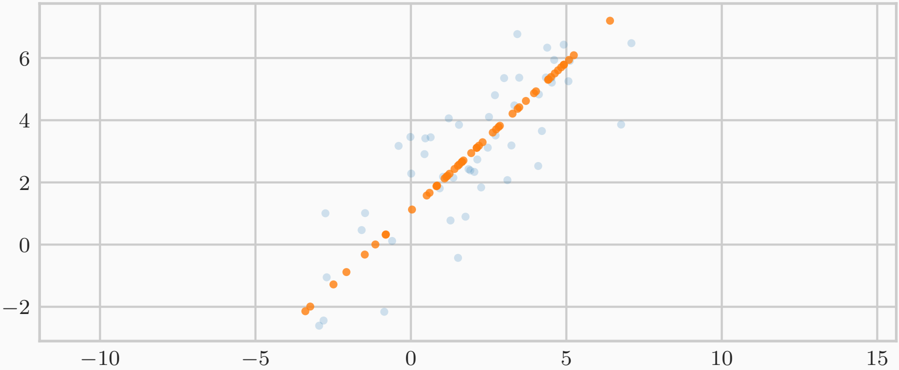
PCA in a nutshell

- Dimension reduction



PCA in a nutshell

- Reconstruction



Questions

- How do we compute the principal directions ?
    - Measure of spreadness
    - Maximization problem
- How many principal components ?
    - Explained variance
    - Scree plot
    - Task driven

Design matrix X

Given of set of n points $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ in a p-dimensional space (usually $\mathbb{R}^p$), the **design matrix** gathers these points

$$X = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_n^T \end{bmatrix}$$

- Each row is a sample
- Each column is a feature

Preparing the dataset

- PCA **needs to have its data centered**. If it is not, replace each sample $\boldsymbol{x}_i$ by $\boldsymbol{x}_i - \overline{\boldsymbol{x}}$ where

$$\overline{\boldsymbol{x}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i$$

- From now on, the dataset is supposed to be centered
  - Point cloud is centered
  - The design matrix X is centered column-wise
- Most of the time, PCA require a feature rescaling: set standard deviation to 1
  - different order of magnitude

Toy example

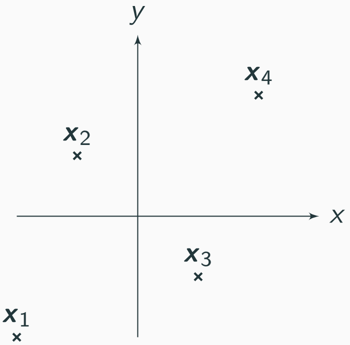
- 4 points in a 2-dimensional space ($n = 4$, $p = 2$)

$$\boldsymbol{x}_1 = \begin{pmatrix} -2 \\ -2 \end{pmatrix} \quad \boldsymbol{x}_2 = \begin{pmatrix} -1 \\ 1 \end{pmatrix} \quad \boldsymbol{x}_3 = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \boldsymbol{x}_4 = \begin{pmatrix} 2 \\ 2 \end{pmatrix}$$

- Design matrix is

$$X = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \boldsymbol{x}_3^T \\ \boldsymbol{x}_4^T \end{bmatrix} = \begin{pmatrix} -2 & -2 \\ -1 & 1 \\ 1 & -1 \\ 2 & 2 \end{pmatrix}$$

- Cloud look like this



Sample variance as measure of spreadness

- Sample variance is a good measure of spreadness

$$s^{\star 2} \triangleq \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2$$

- Inequality

$$\frac{1}{n}\left(\max_i x_i - min_i x_i\right)^2 \leqslant s^{\star 2} \leqslant \left(\max_i x_i - min_i x_i\right)^2$$

- Closed form formulation

Sample variance along an axis v

- For a vector $\mathrm{v} \in \mathbb{R}^p$ such that $\|\mathrm{v}\| = 1$
- Project (orthogonally) the $\boldsymbol{x}_i$'s on the line spanned by v
- New coordinate is: $\langle \boldsymbol{x}_i, \mathrm{v} \rangle$
- Sample variance of new coordinates along v is

$$\frac{1}{n} \sum_{i=1}^{n} \left(\langle \boldsymbol{x}_i, \mathrm{v} \rangle - \sum_{k=1}^{n} \langle \boldsymbol{x}_k, \mathrm{v} \rangle \right)^2$$

- Recall that X is **centered** ($\sum_{k=1}^{n} \boldsymbol{x}_k = 0$), sample variance reduces to

$$\frac{1}{n} \sum_{i=1}^{n} \langle \boldsymbol{x}_i, \mathrm{v} \rangle^2$$

- Which can be written in compact form $\frac{1}{n} \|X\mathrm{v}\|^2$

Toy example: variance along an axis

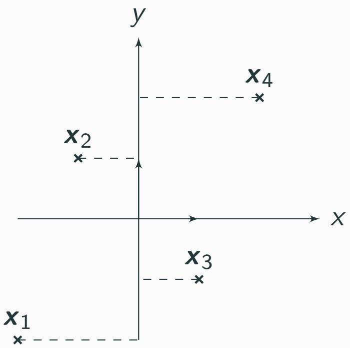
Sample variance along the axis $\mathrm{v} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

- Sample variance along y-axis

$$\frac{1}{4}\left(1^2 + 2^2 + (-1)^2 + (-2)^2\right) = \frac{5}{2}$$

- Compact form

$$\frac{1}{n}\|X\mathrm{v}\|^2 = \frac{1}{4}\left(1^2 + 2^2 + (-1)^2 + (-2)^2\right)$$



Maximizing sample variance along an axis

- Find a vector v that maximizes sample variance, which writes

$$\text{Maximize } \frac{1}{n}\left\|X\mathrm{v}\right\|^2 \text{ such that } \|\mathrm{v}\| = 1$$

- Maximization problem to find first principal direction

$$\boxed{\operatorname*{arg\,max}_{\mathrm{v}\in\mathbb{R}^p} \left\|X\mathrm{v}\right\|^2 \quad \text{s.t.} \quad \left\|\mathrm{v}\right\|^2 = 1}$$

Lagrangian formulation

- This is a constrained maximization problem

$$\arg\max_{\mathrm{v}\in\mathbb{R}^p} \|X\mathrm{v}\|^2 \quad \text{s.t.} \quad \|\mathrm{v}\|^2 = 1$$

- First normalize the constraints

$$\arg\max_{\mathrm{v}\in\mathbb{R}^p} \|X\mathrm{v}\|^2 \quad \text{s.t.} \quad 1 - \|\mathrm{v}\|^2 = 0$$

- Use the Lagrangian formulation

$$\arg\max_{\mathrm{v}\in\mathbb{R}^p} \|X\mathrm{v}\|^2 + \mu\Big(1 - \|\mathrm{v}\|^2\Big)$$

    - now unconstrained maximization problem
    - μ is a Lagrange multiplier

Differentiating matrix expression

- $\|Xv\|^2 = v^T X^T X v$
- For a tiny h

$$\begin{aligned} \|X(v+h)\|^2 &= (v+h)^T X^T X(v+h) \\ &= v^T X^T X v + h^T X^T X v + v^T X^T X h + h^T X^T X h \\ &= \|Xv\|^2 + 2h^T X^T X v + \mathcal{O}\left(\|h\|^2\right) \\ &= \|Xv\|^2 + \left\langle 2X^T X v, h \right\rangle + \mathcal{O}\left(\|h\|^2\right) \end{aligned}$$

- Extract the expression that is linear in h

$$\nabla_v \|Xv\|^2 = 2X^T X v$$

Differentiating the Lagrangian

- Differentiating $\mathcal{L}(\mathrm{v}, \mu) = \|X\mathrm{v}\|^2 + \mu\left(1 - \|\mathrm{v}\|^2\right)$ w.r.t. v yields

$$\nabla_{\mathrm{v}}\mathcal{L} = 2X^TX\mathrm{v} - 2\mu\mathrm{v}$$

- Setting the gradient to zero yields

$$X^TX\mathrm{v} = \mu\mathrm{v} \qquad \Longleftrightarrow \qquad \frac{1}{n}X^TX\mathrm{v} = \frac{\mu}{n}\mathrm{v}$$

- First principal direction v is an **eigenvector** of the **sample covariance matrix** $V = \frac{1}{n}X^TX$
- In that case the sample variance along v is the corresponding eigenvalue

$$\frac{1}{n}\|X\mathrm{v}\|^2 = \frac{1}{n}\mathrm{v}^TX^TX\mathrm{v} = \frac{\mu}{n}\mathrm{v}^T\mathrm{v} = \frac{\mu}{n}$$

Solution to the maximization problem

- Use the **sample covariance matrix**

$$V = \frac{1}{n} X^T X$$

- Find the (unit) eigenvector v_1 with respect to **greatest eigenvalue** of the sample covariance matrix $V = \frac{1}{n} X^T X$
- Variance along v_1 is given by the eigenvalue

$$\frac{1}{n} \left\| X \mathrm{v}_1 \right\|^2 = \lambda_1$$

Toy example: sample covariance matrix

Computing the sample covariance matrix

- (Centered) Design matrix is

$$X = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \boldsymbol{x}_3^T \\ \boldsymbol{x}_4^T \end{bmatrix} = \begin{pmatrix} -2 & -2 \\ -1 & 1 \\ 1 & -1 \\ 2 & 2 \end{pmatrix}$$

- Sample covariance is

$$V = \frac{1}{4} X^T X = \begin{pmatrix} 5/2 & 3/2 \\ 3/2 & 5/2 \end{pmatrix}$$

Toy example: diagonalization

Diagonalizing the sample covariance matrix

- Computing eigenvalues by solving

$$\det \begin{pmatrix} \lambda - 5/2 & -3/2 \\ -3/2 & \lambda - 5/2 \end{pmatrix} = 0$$

  yields $\lambda_1 = 4$ or $\lambda_2 = 1$

- Computing (unit) eigenvector corresponding to highest eigenvalue $\lambda_1 = 4$

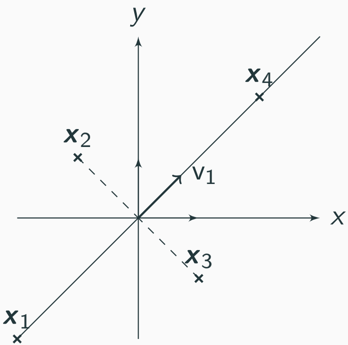
$$V \mathrm{v}_1 = 4\mathrm{v}_1 \quad \text{yields} \quad \mathrm{v}_1 = \begin{pmatrix} \sqrt{2}/2 \\ \sqrt{2}/2 \end{pmatrix}$$

Toy example: variance along v_1

- Variance along v_1 is:

$$\frac{\left(-2\sqrt{2}\right)^2 + 0 + \left(-2\sqrt{2}\right)^2 + 0}{4} = 4$$

- It is also the eigenvalue $\lambda_1 = 4$



Finding v_2

- New maximization problem
  - Same objective
  - Restricting to directions orthogonal to v_1

$$\underset{\mathrm{v}\in\mathbb{R}^p}{\arg\max} \|X\mathrm{v}\|^2 \quad \text{s.t.} \quad \|\mathrm{v}\|^2 = 1 \text{ and } \langle \mathrm{v}, \mathrm{v}_1 \rangle = 0$$

- Lagrangian formulation

$$\mathcal{L}(\mathrm{v}, \mu_1, \mu_2) = \|X\mathrm{v}\|^2 + \mu_1 \left(1 - \|\mathrm{v}\|^2\right) + \mu_2 \langle \mathrm{v}, \mathrm{v}_1 \rangle$$

- Unconstrained maximization problem

$$\underset{\mathrm{v}\in\mathbb{R}^p}{\arg\max} \|X\mathrm{v}\|^2 + \mu_1 \left(1 - \|\mathrm{v}\|^2\right) + \mu_2 \langle \mathrm{v}, \mathrm{v}_1 \rangle$$

  - Two Lagrange multipliers μ_1 and μ_2

Finding v_2

- Setting the gradient to zero

$$\nabla_v \mathcal{L}(v, \mu_1, \mu_2) = 2X^T X v - 2\mu_1 v + \mu_2 v_1 = 0$$

- Taking the inner product with v_1 and using $\langle v, v_1 \rangle = 0$ and $\frac{1}{n} X^T X v_1 = \lambda_1 v_1$

$$\langle \nabla_v \mathcal{L}(v, \mu_1, \mu_2), v_1 \rangle = 0 \text{ yields } \mu_2 = 0$$

- Same as before we get

$$X^T X v = \mu_1 v$$

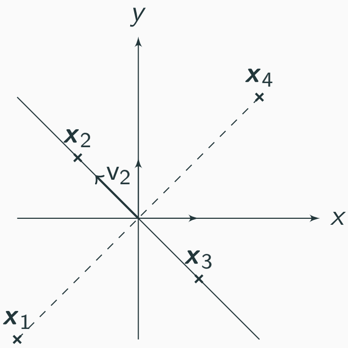
- Find (unit) eigenvector v_2 of **sample covariance matrix** with respect to second greatest eigenvalue λ_2

Toy example: variance along v_2

- Variance along v_2 is:

$$\frac{0 + \left(\sqrt{2}\right)^2 + \left(-\sqrt{2}\right)^2 + 0}{4} = 1$$

- It is also the eigenvalue $\lambda_2 = 1$



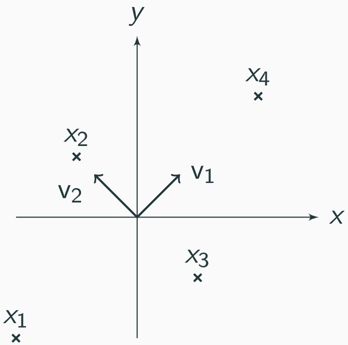
Summary

To compute the PCA of X:

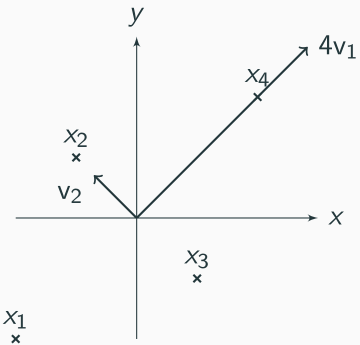
- First center X and possibly rescale features
- Compute the eigen vectors $\mathrm{v}_1, \ldots, \mathrm{v}_p$ corresponding to eigenvalues $\lambda_1 \geqslant \ldots \geqslant \lambda_p$ of V
- $\mathrm{v}_1, \ldots, \mathrm{v}_p$ is a new (orthonormal) representation basis
- Variance along v_i is λ_i

Toy example: principal directions

Principal directions



Principal directions scaled by eigenvalues



Principal component

$$\boxed{\text{Principal components}} = \boxed{\begin{array}{l}\text{New features in the new} \\ \text{representation basis}\end{array}}$$

- The principal directions $(\mathrm{v}_1, \dots, \mathrm{v}_p)$ form a new basis of representation
- The coordinate of all the $\boldsymbol{x}_i$'s w.r.t. v_k is the k-th principal component
- Formally $\boldsymbol{c}_k = X\mathrm{v}_k$
- Formally $C_k = [\boldsymbol{c}_1, \dots, \boldsymbol{c}_k] = XV_k$ where $V_k = [\mathrm{v}_1, \dots, \mathrm{v}_k]$

Principal component properties

- Principal components are also centered
- Principal components are **decorrelated**: $\langle \boldsymbol{c}_k, \boldsymbol{c}_l \rangle = \delta_{kl}$
- Sample variance of principal component $\boldsymbol{c}_k$ is equal to corresponding eigenvalue λ_k of sample variance–covariance matrix

Toy example: principal components

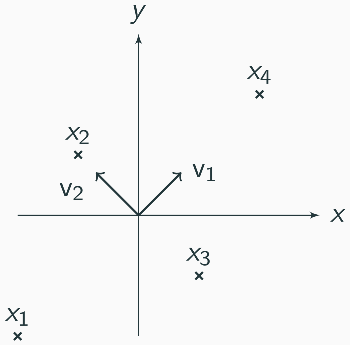
Before PCA

$$X = \begin{pmatrix} -2 & -2 \\ -1 & 1 \\ 1 & -1 \\ 2 & 2 \end{pmatrix}$$

After PCA

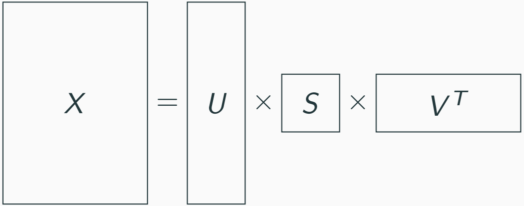
$$C = \begin{array}{c} \begin{pmatrix} -2\sqrt{2} & 0 \\ 0 & \sqrt{2} \\ 0 & -\sqrt{2} \\ 2\sqrt{2} & 0 \end{pmatrix} \\ \begin{array}{cc} \boldsymbol{c}_1 & \quad \boldsymbol{c}_2 \end{array} \end{array}$$

Principal directions



Singular value decomposition (SVD)

- X is a random matrix (non-necessarily square)
- The decomposition



- Columns of U and V are orthonormal ($U^T U = V^T V = I_k$)
- S is diagonal > 0 (singular values)
- S is unique if singular values are ordered (U and V are not unique)
- Nonzero eigenvalues of $X^T X$ (or XX^T) are squared singular values of X.

PCA by SVD

How a SVD can help in computing a PCA?

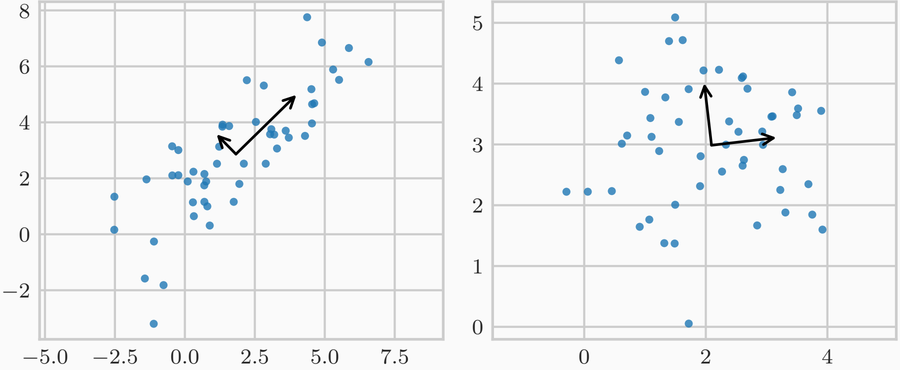
- Suppose that $X = USV^T$ is the SVD of X
- The sample variance-covariance matrix is then: $\frac{1}{n}X^TX = \frac{1}{n}VS^2V^T$
- $\frac{1}{n}X^TX = \frac{1}{n}VS^2V^T$ is a (partial) diagonalization of X
- V gathers the eigenvectors (for nonzero eigenvalues)
- $\frac{\sigma_1^2}{n}, \ldots, \frac{\sigma_k^2}{n}$ are the (nonzero) eigenvalues of $\frac{1}{n}X^TX$
- US gathers the principal components

Choosing the number of principal components

- The scree plot and the elbow empirical law
- Explained variance
- Task driven by cross–validation

Choosing the number of principal components

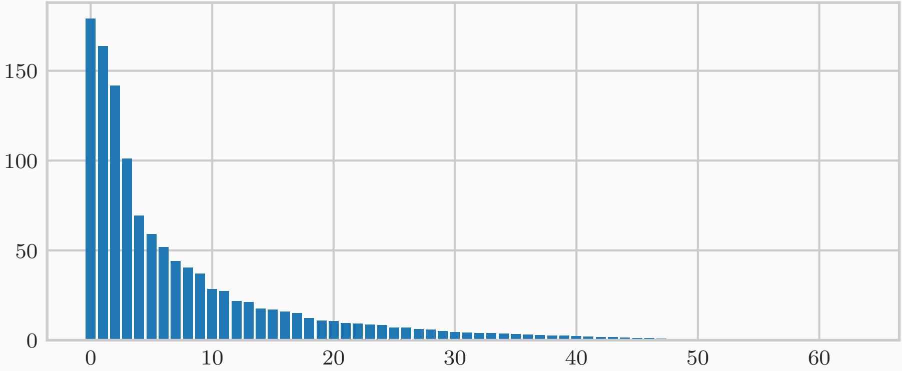
- Compare the two 2-dimensional datasets ($\|\text{arrows}\| = \sqrt{\lambda_i}$)



- Look at the **decreasing rate** of the λ_i

Scree plot

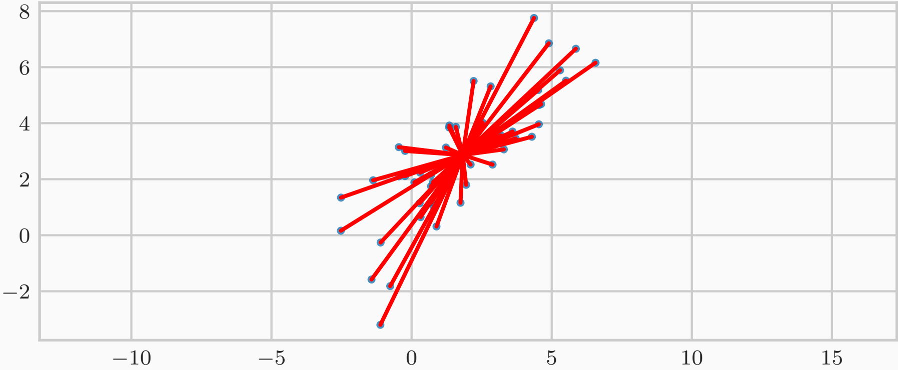
- Barplot of the λ_i's in decreasing order



- Study the decreasing rate of the λ_i's and cut at the elbow

Total variance

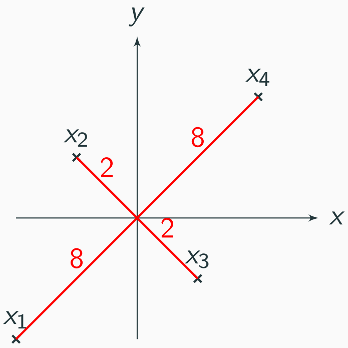
- Total "energy" of the point cloud



- Formally: trace V or $\sum_{i=1}^{p} \lambda_i$

Toy example: total variance

- In our running example



- Total variance is: $\frac{8+8+2+2}{4} = 5$
- Sum of eigenvalues is: $4 + 1 = 5$ (or trace $V = 5$)

Explained variance

- Definition

$$\boxed{\text{Explained variance}} = \boxed{\begin{array}{l}\text{Total variance once we have}\\ \text{projected data onto a chosen}\\ \text{space}\end{array}}$$

- In particular for spaces spanned by $v_1, \ldots, v_k$

$$\boxed{\begin{array}{l}\text{Explained variance of space}\\ \text{spanned by } (v_1, \ldots, v_k)\end{array}} = \lambda_1 + \cdots + \lambda_k$$

Explained variance of $\mathrm{Span}(\mathrm{v}_1, \ldots, \mathrm{v}_k)$

- $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ original dataset
- $\left(V_k^T \boldsymbol{x}_1, \ldots, V_k^T \boldsymbol{x}_n\right)$ projected on $\mathbb{R}^k$
- Explained variance of the $\left(V_k^T \boldsymbol{x}_1, \ldots, V_k^T \boldsymbol{x}_n\right)$

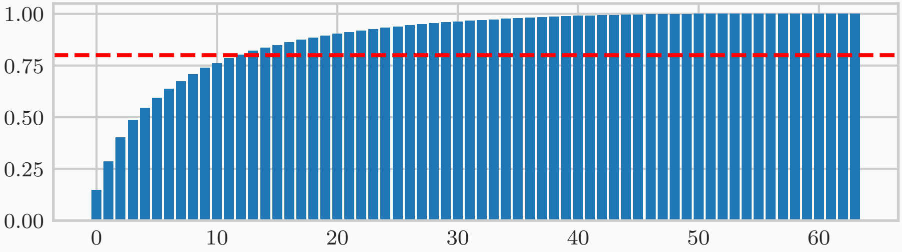
$$
\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n}\left\| V_k^T \boldsymbol{x}_i - \frac{1}{n}\sum_{j=1}^{n} V_k^T \boldsymbol{x}_j \right\|^2 &= \frac{1}{n}\sum_{i=1}^{n}\left\| V_k^T \boldsymbol{x}_i \right\|^2 = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i^T V_k V_k^T \boldsymbol{x}_i && \text{(centered)} \\
&= \frac{1}{n}\operatorname{trace}\left(X V_k V_k^T X^T\right) \\
&= \frac{1}{n}\operatorname{trace}\left(X^T X V_k V_k^T\right) && \text{(shifting property of trace)} \\
&= \operatorname{trace}\left(V V_k V_k^T\right) \\
&= \operatorname{trace}\left(V_k \operatorname{diag}(\lambda_1, \ldots, \lambda_k) V_k^T\right) && \text{(eigenvectors of } V\text{)} \\
&= \operatorname{trace}\left(V_k^T V_k \operatorname{diag}(\lambda_1, \ldots, \lambda_k)\right) && \text{(shifting property again)} \\
&= \sum_{i=1}^{k} \lambda_i
\end{aligned}
$$

Choosing number of principal components

- Proportion of explained variance by k principal components is

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i}$$

- We want k such that $\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i} > 80\%$ (for example)
- Normalized cumulative sum and percent threshold



Task driven

- PCA is often a preprocessing step
- Number of retained principal components k is a parameter to learn
- Consider k as a hyperparameter of the model
- Compute it by cross-validation

Projecting new samples

Suppose we have learned a PCA transformation and we want to transform unseen samples.

- First don't forget to remove to sample mean and maybe rescale the new data
- New k features for a sample $\boldsymbol{x}_{n+1}$ are $V_k^T \boldsymbol{x}_{n+1}$
- New k features for an array of samples Y are YV_k

Reconstructing

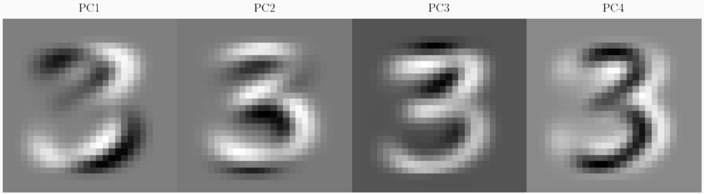
- A sample can be projected on the k-dimensional space spanned by $\mathrm{v}_1, \ldots, \mathrm{v}_k$: $V_k^T \boldsymbol{x} \ldots$
- $\ldots$ and reconstructed to the original n-dimensional space: $V_k V_k^T \boldsymbol{x}$
- $V_k V_k^T$ is an orthogonal projector onto the space spanned by $\mathrm{v}_1, \ldots, \mathrm{v}_k$ because

$$V_k V_k^T \mathrm{v}_l = \begin{cases} \mathrm{v}_l & \text{if } l \leqslant k \\ 0 & \text{else} \end{cases}$$

- Exact reconstruction if $k = n$ (because $V_n = U$ is orthogonal thus $V_n V_n^T = I_n$)

MNIST digits

- MNIST dataset: 7131 samples of the digits "3", $784 = 28 \times 28$ features
- Learn PCA on those digits. Here are the first principal components



Reconstructing digits: denoising property

- Learn PCA on those digits, select k so as to have 95% of explained variance
- Reconstruct noisy unseen digits with k features

Denoising property!

- Interpretation: variations along last principal components are mostly noise

Image compression

- Image of size: $507 \times 676 \times 3$
- Consider each band as a design matrix, X_r, X_g, X_b
  - There is 507 samples and 676 features for each band
- Image reconstruction at different compression rate

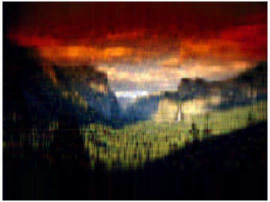
(a) Original image

(b) Rate 90%, 28 PCs

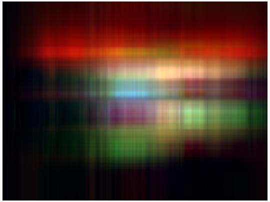
Image compression

(a) Original image

(b) Rate 60%, 115 PCs



(c) Rate 95%, 14 PCs



(d) Rate 99%, 2 PCs

PCA in Python and Scikit–Learn

- Import the PCA module

```python
from sklearn.decomposition import PCA
```

- Instantiate a PCA object and specify number of principal components to retain or percentage of explained variance

```python
pca = PCA(n_components=10)
pca = PCA(n_components=0.95)
```

- Standardize the dataset (if applicable)

```python
from sklearn.preprocessing import StandardScaler
X_std = StandardScaler().fit_transform(X)
```

- Fitting the model with a dataset (design matrix)

```python
pca.fit(X)
```

PCA in Python and Scikit–Learn

- Available information in `pca` object
  - `pca.explained_variance_`: Array of the λ_i's
  - `pca.mean_`: Sample mean of the design matrix
  - `pca.components_`: Matrix V_k^T with k equal to `n_components`
- Available methods (functions) in `pca` object
  - `pca.transform(X_new)`: Projection of new data
  - `res = pca.fit_transform(X)`: Fit and return new features

[1] Karl Pearson. "LIII. On Lines and Planes of Closest Fit to Systems of Points in Space". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (11 1901), pp. 559–572.