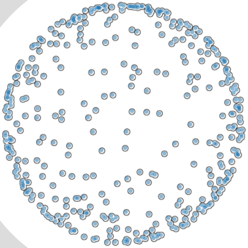
WiSe 2024/25

Machine Learning for Data Science

Lecture by G. Montavon



Lecture 3a

Principal Component Analysis

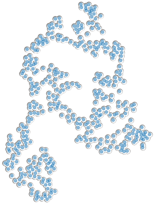
Recap Weeks 1-2

Data science

- Data science studies how one can systematically extract insights from real-world data.
- Data science addresses both the statistical and technical aspects of such process (e.g. robustness, scalability).

Visualizations:

- Visualizations can effectively convey large data distributions to the user.
- Low-dimensional embedding techniques such as MDS or T-SNE enable the visualization of large high-dimensional datasets.
- Sometimes, embeddings do not faithfully convey certain aspects of the data (e.g. distances not preserved in high dimensions, distortion of local geometry, spurious cluster structures, etc.).
- Visualizations do not give *quantifiable* insights.

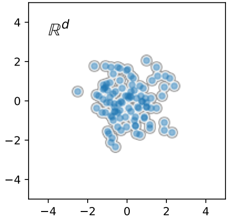


Today's Lecture

- Data Dispersion
- Principal Component Analysis (PCA)
  - Dispersion maximization view
  - Error minimization view
  - PCA as a constrained optimization problem
- Lagrange Multipliers
  - Framework for solving constrained optimization problems
  - Practical examples
- Application of Lagrange Multipliers to PCA
  - Reformulation of PCA as an eigenvalue problem
- PCA explains Data Dispersion

Measuring Data Dispersion

Dispersion is an important property of the data, and indicates how much variation there is in some dataset $\mathcal{D} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_N)$. There are various possible measures of dispersion.



Examples:

- Number of distinct data points.
- Radius of minimum enclosing sphere.
- Average square Euclidean distance from the dataset mean $\boldsymbol{m}$:

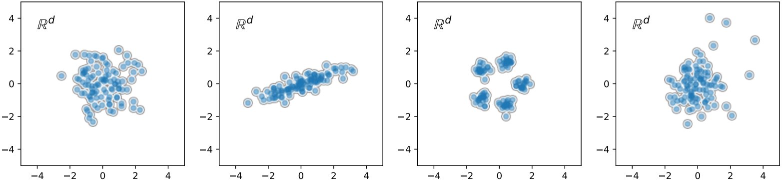
$$s(X) = \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \boldsymbol{m}\|^2$$

The measure $s(X)$ can be seen as a generalization of variance from univariate to multivariate data. Other generalizations are possible.

Explaining Data Dispersion

today's lecture

↓



Observation:

- All distributions above have the same dispersion $s(X)$. Yet, they are very different.
- Often, it is not only desirable to quantify the overall dispersion but also to explain the structure of dispersion.
- Today's lecture focuses on a specific type of dispersion which can be described in terms of directions in input space (aka. Principal Component Analysis).

Part 1

Principal Component Analysis

Hotelling. J Educational Psychology, 24:498–520, 1933.
Pearson. Phil. Mag. 2, 6, 1901.

Principal Component Analysis

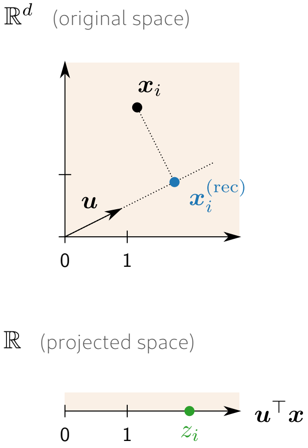
Preliminaries:

- Let $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N \in \mathbb{R}^d$ be a dataset, where d is the number of input features and N is the number of data points (or sample size).
- Let $\boldsymbol{u} \in \mathbb{R}^d$ be a vector of same dimensions, that represents some direction in input space, and that is constrained to be of norm 1, i.e. $\|\boldsymbol{u}\| = 1$.
- Data points can be projected onto this direction by computing the dot product.

$$\forall_{i=1}^N : \; z_i = \boldsymbol{u}^\top \boldsymbol{x}_i$$

- These projections can be backprojected on the input space, by multiplying again by the direction $\boldsymbol{u}$.

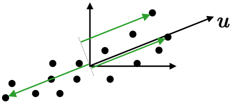
$$\forall_{i=1}^N : \; \boldsymbol{x}_i^{(\text{rec})} = \boldsymbol{u} \underbrace{\boldsymbol{u}^\top \boldsymbol{x}_i}_{z_i}$$



PCA: Two Formulations

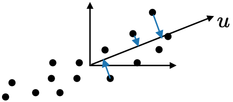
Dispersion Maximization:

- Find a projection $z = \boldsymbol{u}^\top \boldsymbol{x}$ of the data under which the dispersion (variance) is maximized.



Error Minimization:

- Find the direction that minimizes the reconstruction error (MSE) between the original data point $\boldsymbol{x}$ and its backprojection $\boldsymbol{x}^{(\text{rec})} = \boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}$.



The two views coincide **(Pearson 1901)**.

PCA as Dispersion Maximization

Objective:

- Find a direction $\boldsymbol{u}$ (with $\|\boldsymbol{u}\| = 1$) so that the data projected onto this direction (i.e. $z_1, \dots, z_N$) has **maximum variance**.
- This can be cast into an optimization problem:

$$\arg\max_{\boldsymbol{u}} \left[\frac{1}{N} \sum_{i=1}^{N} (\underbrace{\boldsymbol{u}^\top \boldsymbol{x}_i}_{z_i} - \widetilde{m})^2 \right]$$

  where $\widetilde{m} = \frac{1}{N} \sum_{i=1}^{N} z_i$ is the dataset mean in projected space.
- Making sure we first center the data, i.e. $\boldsymbol{m} = \boldsymbol{0}$, it implies that $\widetilde{m} = 0$, and the optimization problem simplifies to:

$$arg \max_{\boldsymbol{u}} \left[\frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{u}^\top \boldsymbol{x}_i)^2 \right]$$

PCA as Error Minimization

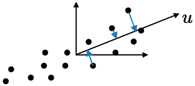
- Find a direction $\boldsymbol{u}$ (with $\|\boldsymbol{u}\| = 1$) so that the data projected on the corresponding subspace best reconstructs the original data, specifically, has **minimal squared distance** to the original data.
- Recall that the reconstruction model is given by:

$$\boldsymbol{x}^{(\text{rec})} = \boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}$$

- This can be cast into the optimization problem:

$$\arg\min_{\boldsymbol{u}} \left[\frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \underbrace{\boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}_i}_{\boldsymbol{x}_i^{(\text{rec})}} \|^2 \right]$$

Connecting the two Approaches



$$\arg\min_{\boldsymbol{u}} \left[\frac{1}{N}\sum_{i=1}^N \|\boldsymbol{x}_i - \boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}_i\|^2\right]$$

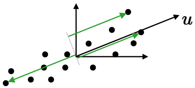
$$= \arg\min_{\boldsymbol{u}} \left[\frac{1}{N}\sum_{i=1}^N (\boldsymbol{x}_i - \boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}_i)^\top (\boldsymbol{x}_i - \boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}_i)\right]$$

$$= \arg\min_{\boldsymbol{u}} \left[\frac{1}{N}\sum_{i=1}^N \underbrace{\boldsymbol{x}_i^\top \boldsymbol{x}_i}_{\text{cst.}} - 2\boldsymbol{x}_i^\top \boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}_i + (\boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}_i)^\top (\boldsymbol{u}\boldsymbol{u}^\top \boldsymbol{x}_i)\right]$$

$$= \arg\min_{\boldsymbol{u}} \left[\frac{1}{N}\sum_{i=1}^N -2(\boldsymbol{x}_i^\top \boldsymbol{u})^2 + \boldsymbol{x}_i^\top \boldsymbol{u}\,\underbrace{\boldsymbol{u}^\top \boldsymbol{u}}_{=1}\,\boldsymbol{u}^\top \boldsymbol{x}_i\right]$$

$$= \arg\min_{\boldsymbol{u}} \left[\frac{1}{N}\sum_{i=1}^N -2(\boldsymbol{x}_i^\top \boldsymbol{u})^2 + (\boldsymbol{x}_i^\top \boldsymbol{u})^2\right]$$

$$= \arg\max_{\boldsymbol{u}} \left[\frac{1}{N}\sum_{i=1}^N (\boldsymbol{x}_i^\top \boldsymbol{u})^2\right]$$



Both approaches give the same result!

Finding Principal Components

- We first recall the PCA optimization problem of slide 8:

$$\arg\max_{\boldsymbol{u}} \Big[\frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{x}_i^\top \boldsymbol{u})^2\Big] \qquad \text{s.t.} \quad \|\boldsymbol{u}\| = 1$$

  and observe it can be rewritten as:

$$= \arg\max_{\boldsymbol{u}} \Big[\frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{u}^\top \boldsymbol{x}_i)(\boldsymbol{x}_i^\top \boldsymbol{u})\Big]$$

$$= \arg\max_{\boldsymbol{u}} \Big[\boldsymbol{u}^\top \underbrace{\Big(\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_i\boldsymbol{x}_i^\top\Big)}_{\Sigma}\boldsymbol{u}\Big] \qquad \text{s.t.} \quad \|\boldsymbol{u}\| = 1$$

  where Σ is the covariance matrix (remember that the data is assumed to be centered). The covariance matrix does not depend on $\boldsymbol{u}$ and can be precomputed.
- Optimization problems with equality constraints can be solved using the method of Lagrange Multipliers.

Part 2

Method of Lagrange Multipliers

Method of Lagrange Multipliers

The method of Lagrange multipliers is a general framework for finding solutions of constrained optimization problems of the type:

$$\arg\max_{\theta} f(\theta) \qquad \text{subject to} \qquad g(\theta) = 0$$

It consists of applying the following two steps:

- **Step 1:** Construct the 'Lagrangian':

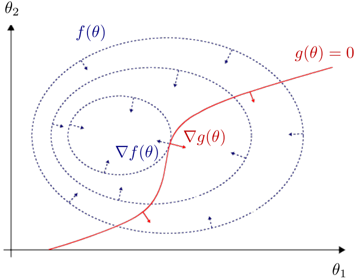
  $$\mathcal{L}(\theta, \lambda) = f(\theta) + \lambda \cdot g(\theta)$$

  where λ is called the Lagrange multiplier.

- **Step 2:** Solve the equation

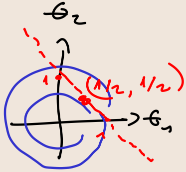
  $$\nabla \mathcal{L}(\theta, \lambda) = 0$$

  which is a necessary condition for the solution.



Intuition for step 2: the equation includes the equation $\nabla f(\theta) = -\lambda \nabla g(\theta)$, i.e. the gradient of objective and constraint are aligned, but point in opposite directions (cf. 2d plot).

Method of Lagrange Multipliers



Example 1: *Solve* the optimization problem:

$$\arg\max_{\theta} [1-(\theta_1^2+\theta_2^2)] \quad \text{s.t.} \quad \theta_1+\theta_2 = 1$$

$$\overbrace{\theta_1+\theta_2-1=0}^{g(x)}$$

$$\mathcal{L}(\theta,\lambda) = 1-\theta_1^2-\theta_2^2 + \lambda\cdot(\theta_1+\theta_2-1)$$

$$\frac{\partial \mathcal{L}}{\partial \theta_1} = -2\theta_1 + \lambda\cdot 1 = 0 \Rightarrow \underline{2\theta_1 = \lambda}$$

$$\frac{\partial \mathcal{L}}{\partial \theta_2} = -2\theta_2 + \lambda\cdot 1 = 0 \Rightarrow \underline{2\theta_2 = \lambda}$$

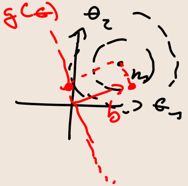
$$\overbrace{\lambda/2}^{\theta_1} + \overbrace{\lambda/2}^{\theta_2} = 1 \Rightarrow \lambda = 1$$

$$\theta_1 = \lambda/2 = 1/2 \qquad \theta_2 = \lambda/2 = 1/2$$

Method of Lagrange Multipliers

Example 2: Let $\boldsymbol{\theta}, \boldsymbol{m}, \boldsymbol{b} \in \mathbb{R}^d$ and $||\boldsymbol{b}|| = 1$. *Solve* the optimization problem:

$$\underset{\boldsymbol{\theta}}{\arg\min}\, \underbrace{||\boldsymbol{\theta} - \boldsymbol{m}||}_{f(\theta)} \quad \text{s.t.} \quad \underbrace{\boldsymbol{\theta}^\top \boldsymbol{b}}_{g(\theta)} = 0$$



$$=$$

$$\mathcal{L}(\theta, \lambda) = \frac{1}{2} \| \theta - m \|^2 + \lambda\, \theta^\top b$$

$$\frac{\partial \mathcal{L}}{\partial \theta} = (\theta - m) + \lambda b = 0$$

$$\left(\begin{array}{l} b^\top \theta - b^\top m + \lambda b^\top b = 0 \\ 0 - b^\top m + \lambda 1 = 0 \Rightarrow \lambda = b^\top m \end{array} \right)$$

$$(\theta - m) + b\, b^\top m = 0$$

$$\theta = m - b\, b^\top m$$

Part 3

Back to PCA

The Solution of PCA

Recall that our PCA optimization problem has the form:

$$\arg\max_{\boldsymbol{u}} [\boldsymbol{u}^\top \Sigma \boldsymbol{u}] \quad \text{s.t.} \quad \|\boldsymbol{u}\| = 1$$

We rewrite the constraint as $\|\boldsymbol{u}\|^2 = 1$ and look for a solution by applying the method of Lagrange multipliers.

- **Step 1:** Build the Lagrangian

$$\mathcal{L}(\boldsymbol{u}, \lambda) = \boldsymbol{u}^\top \Sigma \boldsymbol{u} + \lambda \cdot (1 - \|\boldsymbol{u}\|^2)$$

- **Step 2:** Set gradient of Lagrangian to zero:

$$\begin{aligned} \nabla_{\boldsymbol{u}} \mathcal{L}(\boldsymbol{u}, \lambda) &= \boldsymbol{0} \quad &\Rightarrow& \quad & \Sigma \boldsymbol{u} &= \lambda \boldsymbol{u} \\ \nabla_{\lambda} \mathcal{L}(\boldsymbol{u}, \lambda) &= 0 \quad &\Rightarrow& \quad & \|\boldsymbol{u}\|^2 &= 1 \end{aligned}$$

! PCA solution is an eigenvector of Σ

- Note that the eigenvector is determined up to a sign flip (i.e. in practice, our interpretation of the PCA result should not depend on that sign flip).

Which Eigenvector?

- Start with the eigenvalue problem

$$\Sigma \boldsymbol{u} = \lambda \boldsymbol{u}$$

- Multiply the eigenvalue problem by $\boldsymbol{u}^\top$ on both sides, and identify the terms of the resulting equation:

$$\underbrace{\boldsymbol{u}^\top \Sigma \boldsymbol{u}}_{\text{objective}} = \lambda \underbrace{\boldsymbol{u}^\top \boldsymbol{u}}_{1}$$

- In other words, for the objective to be maximized, we should choose the eigenvector in a way that the corresponding eigenvalue λ is maximum. In other words, one should choose the leading eigenvector.

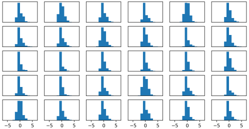
! PCA solution is the leading eigenvector of Σ

Application: Breast Tumor Analysis

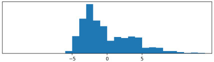
Breast Cancer Wisconsin (Diagnostic)

- 569 instances
- 30 features
- meta-data (benign/ malignant)

statistics of each features



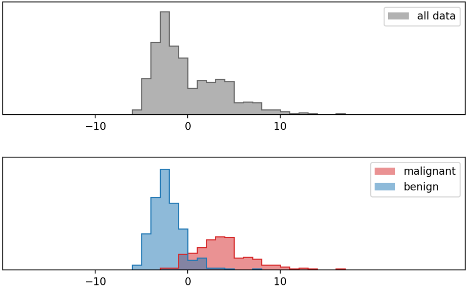
statistics in PCA space



▶ PCA conveys in a more concise way the main variations in a dataset. Instead of looking at as many histograms as there are input features, one only needs to look at a single one.

Application: Breast Tumor Analysis

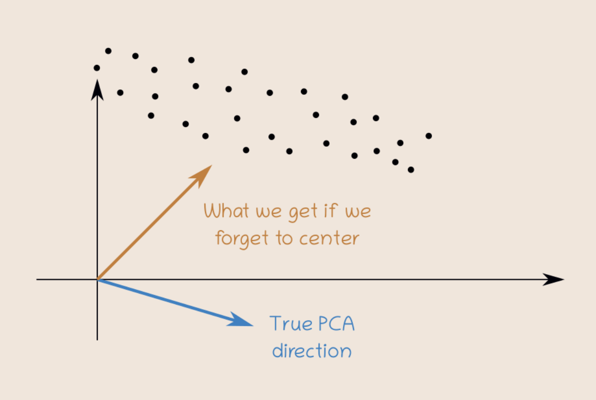
+ color-coding with
meta-data



- PCA analysis can be color-coded based on meta-data. Here, for example, one can see that the two tumor types are well-differentiated with the principal component, i.e. features collectively represent well variations associated to tumor malignancy.

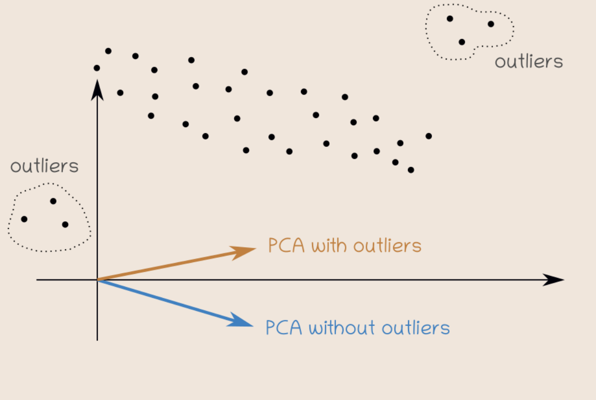
Further Remarks about PCA

Further Remarks



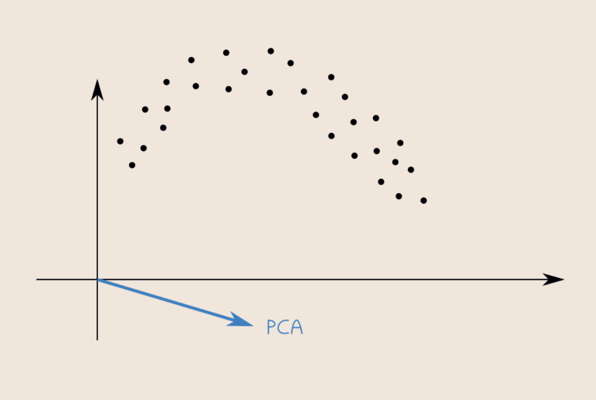
- Don't forget to center the data before applying PCA.

Further Remarks



- PCA is not very robust to outliers. For PCA to be meaningful, data needs to be first cleaned from outliers. Alternatively, robust variants of PCA need to be used.

Further Remarks



- PCA does not describe well the data when it is strongly non-Gaussian. It fails to account for the fact that data may vary locally in different directions.

Summary

Summary

- **Principal Component Analysis** is a dimensionality reduction technique that implements [Pearson 1901]'s principle of minimizing *noise* and maximizing *signal*. (It does both simultaneously**!**).
- PCA reduces high-dimensional data to one dimension (the principal component). The latter represents the **main trend** in the data, which often captures interesting information such as class membership.
- PCA is a well-known method, available in most ML libraries. No hyperparameters to select. It is furthermore is an **exact method**. It always finds the optimum of the objective (defined up to a sign flip).