
Lecture notes on Scientific Computing 2

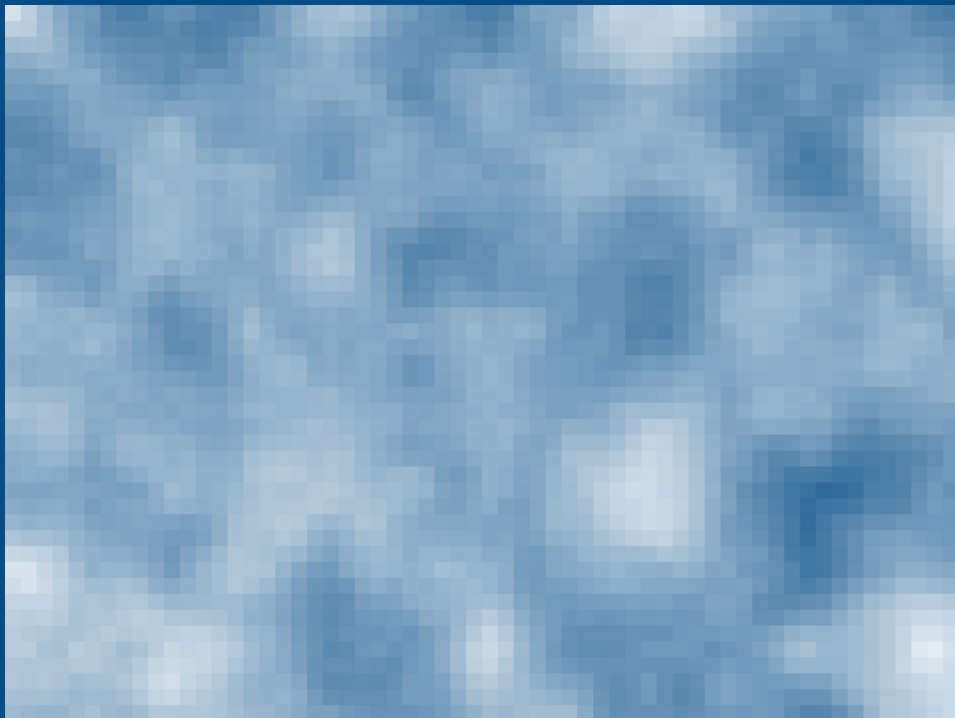
Written by
Manuel Hinz

mh@mssh.dev or s6mlhinz@uni-bonn.de

Lecturer

Prof. Dr. Jochen Garcke

garcke@ins.uni-bonn.de



Contents

Chapter 0 Manuel's notes and introduction	2
0.1 Overview	2
0.1.1 Function approximation / Interpolation	2
0.1.2 Dimensionality reduction	3
Chapter 1 Kernel based methods	4
1.1 Kernels	4
1.1.1 Examples	4
1.1.2 Kernels in machine learning	5
1.1.3 Mercer kernels	6
1.1.4 Properties of kernels	7
1.2 Reproducing Kernel Hilbert Space (RKHS)	10
1.2.1 Kernels for subspaces	13
1.3 Kernel Methods for prediction	23
1.4 Model selection	28
Chapter 2 Dimensionality reduction	34
2.1 Linear dimensionality reduction	34
2.1.1 Alternative derivations of PCA	35
2.1.2 (classical) multidimensional scaling (MDS)	37
2.1.3 Strange effects in high dimensions	38
2.1.4 Properties in dimensionality reduction approaches	39
2.2 nonlinear dimensionality reduction	39
2.2.1 Parallel transport unfolding	42
2.2.2 A view on Kernel MDS embedding	45
2.2.3 Maximum Variance Unfolding	45
2.3 Spectral clustering	48
2.4 Diffusion Maps	51
2.5 t-Stochastic Neighborhood Embedding (t-SNE)	54
2.6 Autoencoders	54
2.6.1 Variational Autoencoder	54
2.7 Generative Diffusion Models	57
2.7.1 Reversed diffusion process	58
Chapter 3 Misc	60
3.1 What we didn't cover	60
3.2 Review of the lecture notes	60
Appendix	62
Tutorials	62
List of Lectures	64

Chapter 0:

Manuel's notes and introduction

Warning

These are unofficial lecture notes written by a student. They are messy, will almost surely contain errors, typos and misunderstandings and may not be kept up to date! I do however try my best and use these notes to prepare for my exams. Feel free to email me any corrections to mh@mssh.dev or s6mlhinz@uni-bonn.de.
Happy learning!

General Information

- Ecampus: [Ecampus link](#)
- Basis: [Basis link](#)
- Website: <https://ins.uni-bonn.de/teachings/ss-2024-440-v3e2-wissenschaftlich/>
- Time slot(s): Tuesday 10-12 and Thursday 08-10
- Exams: Oral, unless more than 50 people take the exam
- Deadlines: tbd
- Two topics:
 - Kernel based methods for function approximation
 - Nonlinear dimensionality reduction / manifold learning / latent space embeddings
- Official lecture notes for most of the lectures
- Exercises are a mix of theory (proofs, (counter-)examples) and programming tasks

Start of lecture 01
(09.04.23)

0.1 Overview

We begin with a quick overview of the two parts of the lecture:

0.1.1 Function approximation / Interpolation

Consider $x_i \in \mathbb{R}^d$, $\hat{f}_i \in \mathbb{R}$:

$$\{(x_i, \hat{f}_i)\}_{i=1}^N.$$

Aim: Find a “good” function f such that

$$f(x_i) = \hat{f}_i, \quad i = 1, \dots, N$$

To compute f , we can make use of a discrete representation of f using **Ansatzfunctions** $\{b_j\}_{j=1}^N$:

$$f(x) = \sum_{j=1}^N c_j b_j(x).$$

Here we assume the same number of data and functions b_j .

For interpolation, we can solve this via:

$$BC = \hat{F}$$

Kernel functions that are centered at the locations x_j turn out to be a good choice:

$$b_j(x) = k(x_j, x)$$

which gives

$$f(x) = \sum_{j=1}^N c_j k(x_j, x).$$

We will also consider approximation instead of interpolation

$$f(x_i) \approx \hat{f}_i.$$

This is in particular relevant in machine learning, where one usually assumes, and actually has noise and measurement errors in the given data.

Example: Assess credit risk

Example: Chemistry / energy of molecules. This needs a kernel on graphs

Example: Time series. This needs a kernel on time series

Remark. We will also see that kernels relate to similarity measures and therefore to distances (dissimilarity).

Lagrangian Interpolation does not work great for a lot of points and higher dimensions

Careful not to discriminate, credit risk should be independent of neighbourhood for example!

Topics in part 1:

- What are kernels and their properties
- Reproducing Kernel Hilbert spaces as the function space in which we are working
- Function interpolation and their approximation properties
- Generalized interpolation for solving partial differential equations
- Kernel methods for prediction in machine learning, representer theorem and regularization
- Gaussian Process Regression and Support Vector Machines

0.1.2 Dimensionality reduction

Distances and similarities are a key aspect of the second topic of the course:

Dimensionality reduction for high-dimensional data

The key idea is to find a “good” low dimensional representation (called embedding), such that chosen properties in high dimensions are approximately preserved.

- Linear dimensionality reduction (numerical linear algebra)
- Nonlinear dimensionality reduction (numerical linear algebra with Non-euclidean geometry)
- Dimensionality reduction with neural networks and other nonlinear function representations

Chapter 1:

Kernel based methods

1.1 Kernels

Definition (Gaussian kernel). The **gaussian kernel** is a prime example of a kernel:

$$k(x, y) := \exp(-\alpha \|x - y\|_2^2) = \phi(\|x - y\|_2)$$

for all $x, y \in \mathbb{R}^d$ where α is a scaling parameter.

Definition 1.1. Let Ω be an arbitrary nonempty set. A function $k : \Omega \times \Omega \rightarrow \mathbb{R}$ is called **kernel** on Ω . We call k a **symmetric kernel** if

$$k(x, y) = k(y, x)$$

for all $x, y \in \Omega$.

Definition 1.2. A function $\Phi : \mathbb{R}^d \rightarrow \mathbb{R}$ is said to be **radial** if there exists a function $\phi : [0, \infty] \rightarrow \mathbb{R}$ such that

$$\Phi(x) = \phi(\|x\|_2)$$

for all $x \in \mathbb{R}^d$. Such a function is traditionally called a **radial basis function (rbf)**.

1.1.1 Examples

Example ((Inverse) multiquadratics). **Multiquadratics** are of the form

$$\phi(r) = (1 + \alpha r^2)^\beta$$

for positive β , while **inverse multiquadratics** have a $\beta < 0$.

Example (Polyharmonic kernels). **Polyharmonic kernels** are of the form

$$\phi(r) = r^\beta \log(|r|)$$

where $\beta \in 2\mathbb{Z}$.

The special case $\beta = 2$ is the so-called **thin-plate spline**. It relates to the partial differential equation that describes the bending of thin plates.

While the previous examples were monotone kernels (as a function of r), these are not!

Example (Wendland's kernels). **Wendland's kernels** are of the form

$$\phi_{a,1} := (1 - r)_+^{(a+1)} (1 + (a+1)r)$$

with the **cut-off function**

$$(x)_+ := \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases}$$

Remark. There are also non radial kernels:

Translation-invariant or **stationary** kernels are functions of differences:

$$k(x, y) = \Phi(x - y).$$

For periodic setups, we have the **Dirichlet kernel** as an example:

$$D(\phi) := \frac{1}{2} + \sum_{j=1}^N \cos(j\phi) = \frac{\sin\left(\left(n + \frac{1}{2}\right)\phi\right)}{2 \sin\left(\frac{\phi}{2}\right)}.$$

This is applied to differences $\phi = \alpha - \beta$ of angles or 2π -periodic arguments and is an important tool for Fourier series theory.

There are so called **zonal kernels**, for working on a sphere, where the kernel can be represented as a function of an angle. An example are functions of inner products, such as

$$k(x, y) = \exp(x^\perp y).$$

Remember, $x^\perp y$ is the (scaled) cosine of the angle between the two vectors.

Remark. We will see that a kernel k on Ω defines a function $k(x, \cdot)$ for all fixed $x \in \Omega$. The space

$$\mathcal{K}_0 := \text{span}\{k(x, \cdot) \mid x \in \Omega\}$$

can for example be used as a so called trial space in meshless methods for solving partial differential equations.

Remark. Kernels can always be restricted to subsets without losing essential properties. This easily allows kernels on embedded manifolds, e.g. the sphere.

Remark. Most of this works for complex kernels too.

1.1.2 Kernels in machine learning

In machine learning the data $x \in \Omega$ can be quite diverse and without (much) structure on first glance. For example consider images, text documents, customers, graphs, ...

Here, one views the kernel as a **similarity measure**, i.e.

$$k : \Omega \times \Omega \rightarrow \mathbb{R}$$

return a number $k(x, y)$ describing the similarity of two patterns x and y .

To work with general data, we first need to represent it in a Hilbert space \mathcal{F} , the so-called **feature space**. One considers the (application dependent) **feature map**

$$\Phi : \Omega \rightarrow \mathcal{F}.$$

The map describes each $x \in \Omega$ by a collection of **features** which are characteristic for a x and capture the essentials of elements of Ω . Since we are now in \mathcal{F} we can work with linear techniques. In particular we can use the scalar product in \mathcal{F} of two elements of Ω represented by their features:

$$\langle \Phi(x), \Phi(y) \rangle_{\mathcal{F}} =: k(x, y)$$

and define a kernel that way.

Remark. Given a kernel, neither the feature map nor the feature space are unique, as the following example shows:

Example. Let $\Omega = \mathbb{R}$, $k(x, y) = x \cdot y$. A feature map, with feature space $\mathcal{F} = \mathbb{R}$ is given by the identity map.

In \mathbb{R}^d , we can work with the standard scalar product

Reminder: A Hilbert space is a complete vector space with a scalar product

Such a construction can be made for any arbitrary kernel, therefore every kernel has many different feature spaces

But, the map $\Phi : \Omega \rightarrow \mathbb{R}^2$ defined by

$$\Phi(x) := (x/\sqrt{2}, x/\sqrt{2})$$

is also a feature map given the same k !

The following two examples show how one can handle non-euclidean origin spaces:

Example (Kernels on a set of documents). Consider a collection of documents. We represent each document as a **bag of words** and describe a bag as a vector in a space in which each dimension is associated with a term from the set of words, i.e. the dictionary. The feature map is

that is a set of frequencies of (chosen) words

$$\Phi(t) := (wf(w_1, t), wf(w_2, t), \dots, wf(w_d, t)) \in \mathbb{R}^d$$

where $wf(w_i, t)$ is the frequency of word w_i in document t .

A simple kernel is the vector space kernel

$$k(t_1, t_2) = \langle \Phi(t_1), \Phi(t_2) \rangle = \sum_{j=1}^d wf(w_j, t_1) wf(w_j, t_2).$$

Natural extensions to this kernel take e.g. word order, relevance or semantics into account, which can be achieved by using matrices in the scalar product:

$$k(t_1, t_2) = \langle S\Phi(t_1), S\Phi(t_2) \rangle = \Phi^\top(t_1) S^\top S \Phi(t_2).$$

Example (Graph kernels). Another non-euclidean data object are graphs, where the class of **random walk kernels** can be defined. These are based on the idea that given a pair of graphs, one performs random walks on both and counts the number of matching walks. With \tilde{A}_\times the adjacency matrix of the **direct product graph** of the two involved graphs, one defines:

$$k(G, H) := \sum_{j=1}^{N_G} \sum_{k=1}^{N_H} \sum_{l=1}^{\infty} \lambda_l [\tilde{A}_\times^l]_{j,k}.$$

More generally, one can define a **random walk graph kernel** k as

$$k(G, H) := \sum_{k=0}^{\infty} \lambda_k q_\times^T W_\times^k p_\times,$$

where W_\times is the **weight matrix** of the direct product graph, q_\times^T is the **stopping probability** on the direct product graph, and p_\times is the initial product distribution on the direct product graph.

1.1.3 Mercer kernels

More generally, one can consider kernels of the **Hilbert-Schmidt** or **Mercer** form

$$k(x, y) = \sum_{i \in I} \lambda_i \varphi_i(x) \varphi_i(y),$$

with certain functions $\varphi_i : \Omega \rightarrow \mathbb{R}$, certain positive **weights** λ_i and an index set I such that the following **summability condition** holds for all $x \in \Omega$:

$$k(x, x) = \sum_{i \in I} \lambda_i \varphi_i(x)^2 < \infty \quad (1)$$

Remark. Such kernels arise in machine learning if the functions φ_i each describe a feature of x and the feature space is the weighted l_2 -space of sequences with indices in I :

$$l_{2,I,\lambda} := \left\{ \{\xi_i\}_{i \in I} : \sum_{i \in I} \lambda_i \xi_i^2 < \infty \right\}.$$

This expansion also occurs when kernels generating positive operators are expanded into eigenfunctions on Ω . Such kernels can be views as arising from generalized convolutions. Generally kernels have three major application fields:

- Convolutions
- Trial spaces
- Covariances

We are mainly concerned with the last two.

Start of lecture 02
(11.04.24)

1.1.4 Properties of kernels

Consider an arbitrary set $X = \{x_1, \dots, x_N\}$ of N **distinct** elements of Ω and a symmetric Kernel K on $\Omega \times \Omega$.

N is the number of data points (always!)

$$f(x) = \sum_{j=1}^N a_j k(x_j, x), x \in \Omega$$

Remark. The set of $k(x_j, \cdot)$ might not be linear independent!

For X we construct the symmetric $N \times N$ Kernel matrix

$$K = K_{X,X} = (k(x_j, x_k))_{1 \leq j, k \leq N}$$

and obtain the interpolation problem

$$\hat{f}_k = f(x_k) = \sum_{j=1}^N a_j k(x_j, x_k)$$

in matrix form

$$K_{X,X} a = \hat{F}$$

Remark. With kernels, we will see that this is indeed solvable, because our matrix is symmetric and positive definite.

Definition 1.3. A Kernel on $\Omega \times \Omega$ is **symmetric and positive semidefinite**, if all Kernel matrices for all finite sets of distinct elements of Ω are symmetric and positive definite

semidefinite and definite have conflicting definitions in the literature!

Theorem 1.4. 1. Kernels arising from **feature maps** via

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle$$

are positive semidefinite.

2. **Hilbert-Schmidt** or **Mercer Kernels**

$$k(x, y) = \sum_{i \in I} \varphi_i(x) \varphi_i(y)$$

are positive semidefinite.

Proof. 1. K is a **Gram(-ian)** matrix

2.

$$\begin{aligned} a^\perp K a &= \sum_{j,k=1}^N a_j a_k k(x_j, x_k) = \sum_{j,k=1}^N a_j a_k \sum_{i \in I} \varphi_i(x_j) \varphi_i(x_k) \\ &= \sum_{i \in I} \lambda_i \sum_{j=1}^N a_j \varphi_i(x_j) \sum_{k=1}^N a_k \varphi_i(x_k) = \sum_{i \in I} \lambda_i \left(\sum_{j=1}^N a_j \varphi_i(x_j) \right)^2 \geq 0 \end{aligned}$$

A Gram matrix, is a matrix whose entries are given by inner products $K_{i,j} = \langle v_i, v_j \rangle$

□

Theorem 1.5. Let K be a symmetric positive semidefinite (spsd) Kernel on Ω . Then

1. $k(x, x) \geq 0$ for all $x \in \Omega$
2. $|k(x, y)|^2 \leq k(x, x)k(y, y)$ for all $x, y \in \Omega$
3. $2|k(x, y)|^2 \leq k(x, x)^2 + k(y, y)^2$ for all $x, y \in \Omega$
4. Any finite linear combination spsd Kernels with nonnegative coefficients gives a spsd Kernel. If any of these kernels is positive definite, and its coefficient is positive, then the combination of kernels is positive definite.
5. The product of two spsd kernels is spsd.
6. The product of two spd kernels is spd.

Proof. 1.: Use the set $\{x\}$ in Definition 1.3.

2.: Consider K of $\{x, y\}$. The determinant of such a positive semidefinite matrix is nonnegative, therefore

$$k(x, x)k(y, y) - k(x, y)^2 \geq 0$$

3.: $2ab \leq a^2 + b^2$ for all $a, b \in \mathbb{R}_0^+$. Therefore this follows from 2.

4.: Expand $a^\perp K a$ to see this.

5.: Follows from Lemma 1.6.

6.: Follows from Lemma 1.6 and a bit more linear algebra. \square

Lemma 1.6 (Schur's Lemma). For two matrices A, B , the matrix C with elements

$$C_{jk} = A_{jk}B_{jk}$$

is called the **Schur product** or **Hardarmard product**. The Schur product of two psd matrices is psd.

Proof. Decompose $A = S^\perp D S$ with S an orthogonal matrix and $D = \text{diag}(\lambda_1, \dots, \lambda_n)$ a diagonal matrix with $\lambda_i \geq 0$ the eigenvalues of A .

For all $q \in \mathbb{R}^N$ we look at

$$\begin{aligned} q^\perp C q &= \sum_{j,k} q_j q_k a_{jk} b_{jk} = \sum_{j,k=1}^N q_j q_k \sum_{m=1}^N \lambda_m S_{jm} S_{km} \\ &= \sum_{m=1}^N \lambda_m \sum_{j,k=1}^N \underbrace{q_j S_{jm}}_{P_{k,m}} \underbrace{q_k S_{km}}_{P_{km}} b_{jk} = \sum_{m=1}^N \sum_{j,k=1}^N \underbrace{P_{jm} P_{km} b_{jk}}_{\geq 0 \text{ since } B \text{ is psd}} \geq 0 \end{aligned}$$

\square

Remark. Note that we only considered symmetric matrices, the above also holds if one of the matrices is not symmetric, but positive definite instead.

Remark. Our overall aim is to go from kernels to a **Reproducing Kernel Hilbert space (RKHS)**. Therefore we define candidate spaces and a bilinear form in a way we would expect them.

Definition. For spsd K we define

$$H := \text{span}\{k(x, \cdot) \mid x \in \Omega\}.$$

In the same way

$$L := \text{span}\{\delta_x \mid x \in \Omega, \delta_x : H \rightarrow \mathbb{R}\}$$

the linear space of all finite linear combinations of pointevaluation functionals actions on functions of H , where

$$\delta_x(f) = f(x).$$

It is important that elements of L act on elements of H ! These two spaces are paired in some sense.

We can, by definition, write all Elements from L and H as

$$\lambda_{a,X} := \sum_{j=1}^N a_j \delta_{x_j}$$

$$f_{a,X} := \sum_{j=1}^N a_j k(x_j, x) = \lambda_{a,X}^{(y)} k(x, y)$$

with $a \in \mathbb{R}^n, X = \{x_1, \dots, x_N\} \subset \Omega$ any arbitrary finite subset of Ω .

Remark. From $f_{a,X} = 0$ or $\lambda_{a,X} = 0$ it does not follow that $a = 0$!

We now define a bilinear form on L

$$\langle \lambda_{a,X}, \lambda_{b,Y} \rangle_L := \sum_{j=1}^M \sum_{k=1}^N a_j b_k k(x_j, x_k) = \lambda_{a,X}^{(x)} \lambda_{b,Y}^{(y)} k(x, y) = \lambda_{a,X}(f_{b,Y})$$

Added remark. One has to be a bit careful here: $\lambda_{a,X}^{(x)} \lambda_{b,Y}^{(y)} k(x, y)$ does not mean point wise multiplication, but concatenation:

$$\lambda_{a,X}^{(x)} \lambda_{b,Y}^{(y)} k(x, y) = \lambda_{a,X}^{(x)} (\lambda_{b,Y}^{(y)} k(x, y))$$

This is well-defined, since it is based on the actions of the functional and not the specific representation.

We can observe that the bilinear form is psd, since the kernel matrices have this property.

$$|\lambda_{a,X}(f_{b,Y})| = |\langle \lambda_{a,X}, \lambda_{b,Y} \rangle_L| \leq \|\lambda_{a,X}\|_L \|\lambda_{b,Y}\|_L \quad (*)$$

Theorem 1.7. If K is spsd Kernel on Ω , the bilinear form $\langle \cdot, \cdot \rangle_L$ is positive definite in the space L of functionals defined on H . This L is a pre-Hilbert-space.

Proof. $0 = \langle \lambda_{a,X}, \lambda_{a,X} \rangle_L$ for $a \in \mathbb{R}^n, X = \{x_1, \dots, x_N\} \subset \Omega$.

Then by $(*)$ we have $\lambda_{a,X} = 0$ as a functional on H . □

Theorem 1.8. The mapping $R : \lambda_{a,X} \mapsto f_{a,X} = \lambda_{a,X}(k(\cdot, y))$ is linear and bijective from L onto H . Thus

$$\langle f_{a,X}, f_{b,Y} \rangle_H := \langle \lambda_{a,X}, \lambda_{b,Y} \rangle_L = \langle R(\lambda_{a,X}), R(\lambda_{b,Y}) \rangle_H$$

is an inner product on H . R acts as the Riesz map.

Proof. Linearity is obvious. If $f_{b,Y} = R(\lambda_{b,Y}) \in H$ vanishes, the definition of $\langle \cdot, \cdot \rangle_L$ implies that $\lambda_{b,Y}$ is orthogonal to all of L . Due to Theorem 1.7 it is zero. The Riesz property comes from the definition of $\langle \cdot, \cdot \rangle_L$:

$$\lambda_{a,X}(f_{b,Y}) = \langle \lambda_{a,X}, \lambda_{b,Y} \rangle_L = \langle f_{a,X}, f_{b,Y} \rangle_H = \langle R(\lambda_{a,X}), f_{b,Y} \rangle$$

□

Specializing to $\lambda_{1,x}$, i.e. to a point $x \in \Omega$, we get:

$$\begin{aligned} \langle \lambda_{1,x}, \lambda_{b,Y} \rangle_L &= \lambda_{1,x}(f_{b,Y}) = \delta_x(f_{b,Y}) = f_{b,Y}(x) \\ &= \langle R(\lambda_{1,x}), R(\lambda_{b,Y}) \rangle_H = \langle R(\lambda_{1,x}), f_{b,Y} \rangle_H = \langle k(x, \cdot), f_{b,Y} \rangle_H \end{aligned}$$

In other words, for all $f \in H, x \in \Omega$, we have

$$f(x) = \underline{\delta_x(f)} = \langle f, R(\delta_x) \rangle_H = \langle f, k(x, \cdot) \rangle_H$$

which is the so-called **reproduction equation** for values of functions from the inner product.

There might be different representations of elements in L, H . While the representation is not unique, the element is

Here we use that the functionals in L are restricted to functions in H

Added remark. In this lecture (\star) refers to the reproduction equation.

For $f = k(\cdot, y)$, we set $k(x, y) = \langle k(x, \cdot), k(y, \cdot) \rangle_H$. We furthermore can observe $\forall f \in H, x \in \Omega$:

$$|\delta_x(f)| = |f(x)| = |\langle f, k(x, \cdot) \rangle_H| \leq \|f\|_H \|k(x, \cdot)\|_H = \|f\|_H \sqrt{K(x, x)}$$

and

$$\langle \delta_x, \delta_y \rangle_L = \langle k(x, \cdot), k(y, \cdot) \rangle_H = k(x, y) \forall x, y \in \Omega$$

$$\|\delta_x - \delta_y\|_L^2 = \|\delta_x\|_L^2 - 2\langle \delta_x, \delta_y \rangle + \|\delta_y\|_L^2 = k(x, x) - 2\langle k(x, \cdot), k(y, \cdot) \rangle_H + k(y, y)$$

is a **distance** on Ω :

$$\text{dist}(x, y) := \|\delta_x - \delta_y\|_L = \sqrt{k(x, x) - 2\langle k(x, \cdot), k(y, \cdot) \rangle_H + k(y, y)}.$$

We see that for all $x, y \in \Omega$

$$|f(x)f(y)| \leq \|f\|_H \|\delta_x - \delta_y\|_L = \|f\|_H \text{dist}(x, y)$$

and therefore all functions in H are continuous with respect to this distance.

Theorem 1.9. Each symmetric positive definite kernel k on a set Ω is the **reproducing kernel** of a Hilbert space called the **native space** $\mathcal{H} = \mathcal{N}_k$ of the kernel. This Hilbert space is unique and it is a space of functions on Ω . The kernel k fulfills

$$\langle f, k(x, \cdot) \rangle_{\mathcal{H}} = f(x) \quad (2)$$

Proof. Citation: The existence of native spaces follows from standard Hilbert space arguments, see e.g. chapter 11 from the lecture notes of Schaback. \square

\mathcal{H} can be constructed as the closure of H

Added remark. The good ideas are from Schaback, the errors are from me, Prof. Garcke

The errors in this script are largely due to me :)

Proof of uniqueness:

If k is a reproducing kernel in a different Hilbert space T , we observe

$$\langle k(x, \cdot), k(y, \cdot) \rangle_{\mathcal{H}} = k(x, y) = \langle k(x, \cdot), k(y, \cdot) \rangle_T$$

which shows that the inner products coincide on H . Since T is a Hilbert space, it must contain the closure \mathcal{N}_k of H as a closed subspace. For T to be larger than \mathcal{H} non-zero element $f \in T$ must exist that is orthogonal to \mathcal{N}_k and in particular to H . We observe

$$f(x) = \langle f, k(x, \cdot) \rangle_T = 0 \quad \forall x \in \Omega.$$

which is a contradiction to $f \neq 0$, because of (2) for T .

Dual spaces:

$\delta_x : \mathcal{N}_k \rightarrow \mathbb{R}, f \mapsto f(x)$ for all $f \in \mathcal{N}_k, x \in \Omega$.

The dual space \mathcal{N}_k^* of \mathcal{N}_k is again a Hilbert space.

$$\begin{aligned} R : \mathcal{N}_k^* &\rightarrow \mathcal{N}_k \\ \lambda(f) &= \langle f, R(\lambda) \rangle_{\mathcal{N}_k} \forall f \in \mathcal{N}_k, \lambda \in \mathcal{N}_k^* \\ \langle \lambda, \mu \rangle_{\mathcal{N}_k^*} &= \langle R(\lambda), R(\mu) \rangle_{\mathcal{N}_k} \forall \lambda, \mu \in \mathcal{N}_k^* \end{aligned}$$

Also via the reproducing equation 2

$$\delta_x(f) = \langle f, k(x, \cdot) \rangle_{\mathcal{N}_k} \forall f \in \mathcal{N}_k, x \in \Omega.$$

So $k(x, \cdot)$ is the **Riesz representer** $R(\delta_x)$ of δ_x

$$\begin{aligned} \langle \delta_x, \delta_y \rangle_{\mathcal{N}_k^*} &= \langle R(\delta_x), R(\delta_y) \rangle_{\mathcal{N}_k} = k(x, y) & \forall x, y \in \Omega \\ \|\delta_x\|_{\mathcal{N}_k^*} &= \|k(x, \cdot)\|_{\mathcal{N}_k} = \sqrt{k(x, x)} & \forall x \in \Omega \\ \lambda(f) &= \langle f, \lambda^* k(x, \cdot) \rangle_{\mathcal{N}_k} & \forall f \in \mathcal{N}_k, \lambda \in \mathcal{N}_k^* \end{aligned}$$

so that $\lambda^* k(x, \cdot)$ is the Riesz representer of λ .

1.2 Reproducing Kernel Hilbert Space (RKHS)

Definition 1.10. A Hilbert space \mathcal{H} of functions on a set Ω with an inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ is called **RKHS** if there is a kernel function $k : \Omega \times \Omega \rightarrow \mathbb{R}$ with $k(x, \cdot) \in \mathcal{H}$ for all $x \in \Omega$ and the reproducing kernel property

$$f(x) = \langle f, k(x, \cdot) \rangle_{\mathcal{H}} \quad \forall x \in \Omega, f \in \mathcal{H}$$

This directly implies

$$k(y, x) = \langle k(y, \cdot), k(x, \cdot) \rangle_{\mathcal{H}} = \langle k(x, \cdot), k(y, \cdot) \rangle_{\mathcal{H}} = k(x, y).$$

For positive semi-definiteness one can use a Gram matrix argument or take any $X = \{x_1, \dots, x_N\} \subset \Omega$ and $a \in \mathbb{R}^n$

$$\begin{aligned} \sum_{j,k=1}^N a_j a_k k(x_j, x_k) &= \sum_{j,k=1}^N a_j a_k \langle k(x, \cdot), k(y, \cdot) \rangle_{\mathcal{H}} \\ &= \left\langle \sum_{j=1}^N a_j k(x_j, \cdot), \sum_{k=1}^N a_k k(x_k, \cdot) \right\rangle_{\mathcal{H}} \\ &= \left\| \sum_{j=1}^N a_j k(x, \cdot) \right\|_{\mathcal{H}}^2 \geq 0 \end{aligned}$$

Theorem 1.11. Each Hilbert space \mathcal{H} of real valued functions on some set Ω with point evaluation functionals

$$\delta_x : f \mapsto f(x) \quad \forall f \in \mathcal{H}$$

is a RKHS with a unique positive definite kernel k on Ω . The kernel is uniquely defined by providing the Riesz representers of the (continuous) point evaluation functionals. The space \mathcal{H} is the native space of the kernel.

Proof. Under the given hypothesis, there must be a Riesz representer of δ_x . By the definition of the Riesz map it takes the form $k(x, \cdot) \in \mathcal{H}$ satisfying the reproduction equation 2.

In other words, any such Hilbert space has a symmetric positive definite kernel.

The final statement follows from theorem 1.9, because the native space and \mathcal{H} are Hilbert spaces that contain all $k(x, y)$. \square

Theorem 1.12. If a Hilbert (sub-)space of functions on Ω has a finite orthogonal basis v_1, \dots, v_N the reproducing kernel is

$$k_N(x, \cdot) = \sum_{j=1}^N v_j(x) v_j(\cdot) \quad \forall x \in \Omega$$

In case of a subspace we have

$$\sum_{j=1}^N |v_j(x)|^2 = k_N(x, x) \leq k(x, x) \quad \forall x \in \Omega$$

Which in some sense means that larger dimensions of the subspace can't add too much to the norm

Proof. The kernel must have a representation in the ONB

$$k_N(x, \cdot) = \sum_{j=1}^N \langle k_N(x, \cdot), v_j \rangle_{\mathcal{H}} v_j(\cdot) \stackrel{(2)}{=} \sum_{j=1}^N v_j(x) v_j(\cdot)$$

For the subspace,

$$\begin{aligned} k_N(x, x) &= \langle k_N(x, \cdot), k_N(x, \cdot) \rangle_{\mathcal{H}} \\ &\stackrel{\text{Hilbert subspace}}{=} \langle k_N(x, \cdot), k(x, \cdot) \rangle_{\mathcal{H}} \\ &\leq \sqrt{k_N(x, x)} \sqrt{k(x, x)} \quad \forall x \in \Omega \end{aligned}$$

\square

Added remark. The subspace property does not hold for arbitrary Hilbert spaces, this tells us that a RKHS is really not the same as a normal Hilbert space!

Remember: Kernels of the Mercer form

Start of lecture 04
(18.04.24)

$$k(x, y) = \sum_{i \in I} \lambda_i \varphi_i(x) \varphi_i(y)$$

with the summability condition

$$k(x, x) = \sum_{i \in I} \lambda_i \varphi_i^2(x) < \infty.$$

Then observe

$$\begin{aligned} |f(x)| &= \left| \sum_{i \in I} \langle f, \varphi_i \rangle_{\mathcal{H}} \varphi_i(x) \right| \\ &\leq \sum_{i \in I} \left| \frac{\langle f, \varphi_i \rangle_{\mathcal{H}}}{\sqrt{\lambda_i}} \right| |\varphi_i(x)| \sqrt{\lambda_i} \\ &\leq \sqrt{\sum_{i \in I} \frac{\langle f, \varphi_i \rangle_{\mathcal{H}}^2}{\lambda_i}} \sqrt{\underbrace{\sum_{i \in I} \varphi_i^2(x) \lambda_i}_{< \infty}} \\ \mathcal{H} &:= \left\{ f \in \mathcal{H} : \|f\|_{\lambda}^2 = \sum_{i \in I} \frac{\langle f, \varphi_i \rangle_{\mathcal{H}}^2}{\lambda_i} \right\} \\ \langle f, g \rangle_{\lambda} &= \sum_{i \in I} \frac{\langle f, \varphi_i \rangle_{\mathcal{H}} \langle g, \varphi_i \rangle_{\mathcal{H}}}{\lambda_i} \quad \forall f, g \in \mathcal{H}_{\lambda} \end{aligned} \quad (3)$$

Using 3 as the kernel, we have to check if all $f_x := k(x, \cdot) \in \mathcal{H}_{\lambda}$.
Observe

$$\langle f_x, \varphi \rangle_{\mathcal{H}} = \lambda_i \varphi_i(x)$$

and

$$\sum_{i \in I} \frac{\langle f_x, \varphi_i \rangle_{\mathcal{H}}^2}{\lambda_i} = \sum_{i \in I} \lambda_i \varphi_i^2(x) < \infty$$

to see $f_x \in \mathcal{H}_{\lambda}$.

Check the reproduction equation

$$\begin{aligned} \langle f, k(x, \cdot) \rangle_{\lambda} &= \sum_{i \in I} \frac{\langle f, \varphi_i \rangle_{\mathcal{H}} \langle k(x, \cdot), \varphi_i \rangle_{\mathcal{H}}}{\lambda_i} \\ &= \sum_{i \in I} \frac{\langle f, \varphi_i \rangle_{\mathcal{H}} \lambda_i \varphi_i(x)}{\lambda_i} = f(x) \quad \forall x \in \Omega \end{aligned}$$

The kernel therefore reproduces on \mathcal{H}_{λ} . The proves theorem ??.

If a Hilbert space of functions on Ω has a countable ONB $\{\varphi_i\}_{i \in I}$, each summability condition (**) leads to a reproducing mercer kernel (*) for a suitable subspace of functions with continuous point evaluations.

Corollary 1.13. The spaces \mathcal{H}_{λ} defined above are the natives spaces of the corresponding Mercer kernels.

Example (Trigonometric polynomials). Consider the space of trigonometric polynomials $\frac{1}{\sqrt{2}}, \cos(nx), \sin(nx), n \in \mathbb{N}$ which are **orthonormal** in the inner product

$$\langle f, g \rangle = \frac{1}{\pi} \int_{-\pi}^{\pi} f(t)g(t)dt.$$

With $I = (0, 0) \cup (\mathbb{N}, 0) \cup (0, \mathbb{N})$

$$\varphi_i(x) = \begin{cases} \frac{1}{\sqrt{2}} & i = (0, 0) \\ \cos(nx) & i = (n, 0), n \geq 1 \\ \sin(nx) & i = (n, 0), n \geq 1 \end{cases}$$

So for $f \in \mathcal{H}$

$$f = \sum_{i \in I} \langle f, \varphi_i \rangle_{\mathcal{H}} \varphi_i.$$

All φ_i are uniformly bounded, so the summability condition does hold when the weights are summable.

Fixing some $m \geq 1$, we define

$$\lambda_i = \begin{cases} 1 & i = (0, 0) \\ n^{-2m} & \text{otherwise} \end{cases}$$

We set the Mercer kernel

$$k_{2m}(x, y) := \frac{1}{\sqrt{2}} + \sum_{n=1}^{\infty} n^{-2m} (\cos(nx) \cos(ny) + \sin(nx) \sin(ny))$$

One can see $K_{2m}'' = K_{2m-2}$, so K_{2m} piecewise polynomial of degree $2m$, which is $2m - 2$ times differentiable.

this can also be thought of as an **extension kernel**

This can be rewritten with the usual trigonometric rules

1.2.1 Kernels for subspaces

Let us fix a nonempty set $X \subset \Omega$ and look at the closed subspace

$$\mathcal{H}_X := \overline{\text{span}\{k(x, \cdot) | x \in X\}} \subseteq \mathcal{H}$$

Projector for \mathcal{H} to the closed subspace \mathcal{H}_0 : $\pi_0 : \mathcal{H} \rightarrow \mathcal{H}_0$ with properties

This is NOT $\{\mathcal{H}_0\}$, but a generic subspace

- $\pi_0^2 = \pi_0$
- π_0 gives unique best approximation in \mathcal{H}_0 , $u \mapsto u_0$
- $u_0 \perp u - u_0$
- $\text{Id} - \pi_0$ projects onto the orthogonal complement $\mathcal{H}_0^\perp = \{u \in \mathcal{H} \mid \langle u, v \rangle_{\mathcal{H}} = 0 \forall v \in \mathcal{H}_0\}$
- $\mathcal{H} = \mathcal{H}_0 + \mathcal{H}_0^\perp$

Theorem 1.14. Let \mathcal{H}_0 be a closed subspace of \mathcal{H} with reproducing kernel k_0 and let $\pi_0 : \mathcal{H} \rightarrow \mathcal{H}_0$ be the projection onto \mathcal{H}_0 .

The subspace kernel is

$$k_0(x, \cdot) = \pi_0 k(x, \cdot)$$

for all $x \in \Omega$. The reproducing kernel for the orthogonal complement \mathcal{H}_0^\perp is $k - k_0$.

Proof. $\text{Id} = \pi_0 + (\text{Id} - \pi_0) = \pi_0 + \pi_0^\perp$.

Thus $f(x) = (\pi_0 \circ f)(x) + (\pi_0^\perp \circ f)(x)$ inserted into the reproducing equation

$$\begin{aligned} f(x) &= \langle f, k(x, \cdot) \rangle_{\mathcal{H}} \\ &= \langle \pi_0 f + \pi_0^\perp f, \pi_0 k(x, \cdot) + \pi_0^\perp k(x, \cdot) \rangle_{\mathcal{H}} \\ &= \langle \pi_0 f, \pi_0 k(x, \cdot) \rangle + \langle \pi_0^\perp f, \pi_0^\perp k(x, \cdot) \rangle \end{aligned}$$

Using $f \in \mathcal{H}_0$ and $f \in \mathcal{H}_0^\perp$ eliminates on part of the sum each and the statements follow. \square

Remark. *Orthogonal space decompositions correspond to additive kernel decompositions using the appropriate projections.*

Theorem 1.15. *Let $X \subseteq \Omega$ be nonempty. For the closed subspace \mathcal{H}_X it holds*

$$\mathcal{H}_X^\perp = \{f \mid f \in \mathcal{H} : f(X) = \{0\}\}.$$

Proof. If $f(X) = \{0\}$, then $\langle f, v \rangle_{\mathcal{H}} = 0 \forall v \in \mathcal{H}_X$.

$$f(x) = \langle f, k(x, \cdot) \rangle_{\mathcal{H}}$$

since $f \in \mathcal{H}_X^\perp$ by the reproduction equation and conversely we set for $f \in \mathcal{H}_X^\perp$ that $f(X) = \{0\}$. \square

With π_X the projector from \mathcal{H} to \mathcal{H}_X we denote

$$f_X := \pi_X(f).$$

Standard results from Hilbert space theory gives us

Theorem 1.16. *Each function $f \in \mathcal{H}$ has an orthogonal decomposition*

$$f = f_X + f_{X^\perp}$$

with $f_X \in \mathcal{H}_X$ and $f_{X^\perp} \in \mathcal{H}_X^\perp$. In particular each $f \in \mathcal{H}$ has an interpolant $f_X \in \mathcal{H}_X$ recovering the values of f on X . Additionally

$$\|f - f_X\|_{\mathcal{H}} = \inf_{g \in \mathcal{H}_X} \|f - g\|_{\mathcal{H}}$$

and

$$\|f_X\|_{\mathcal{H}} = \inf_{\substack{g \in \mathcal{H}: \\ f(x)=g(x) \\ \forall x \in X}} \|g\|_{\mathcal{H}} = \inf_{v \in \mathcal{H}_{X^\perp}} \|f - v\|_{\mathcal{H}}$$

Corollary 1.17. *The interpolant $f_X \in \mathcal{H}_X$ to a function f on X is at the same time the best approximation to f from all functions in \mathcal{H}_X .*

This is just the previous theorem in words

Corollary 1.18. *The interpolant $f_X \in \mathcal{H}_X$ to a function f on X minimizes the norm under all interpolants from the full space \mathcal{H} .*

This property is usefull, if the norm encodes smoothness as well.

Corollary 1.19. *For all sets $X \subseteq Y \subseteq \Omega$ and $f \in \mathcal{H}$ we have*

$$\|f_X\|_{\mathcal{H}} \leq \|f_Y\|_{\mathcal{H}} \leq \|f\|_{\mathcal{H}}$$

Penalizing unnecessary changes of the function

and

$$\|f\|_{\mathcal{H}} \geq \|f - f_X\|_{\mathcal{H}} \geq \|f - f_Y\|_{\mathcal{H}}$$

where for completeness we define $f_\emptyset = 0, f_{\emptyset^\perp} = f$ and $\mathcal{H}_\emptyset = \{0\}$ with $\mathcal{H}_{\emptyset^\perp} = \mathcal{H}$.

Consider only $f(\cdot) = k(x, \cdot)$ for a fixed $f, x \in \Omega$.

Start of lecture 05
(23.04.24)

Definition 1.20. *The function*

$$P_X(x) := \|k(x, \cdot) - k_X(x, \cdot)\|_{\mathcal{H}} \quad x \in \Omega$$

is called **power function** w.r.t. the set X and the kernel k .

A different definition goes with the **error functional** $\epsilon_{X,x} \in \mathcal{H}^*$

$$\epsilon_{X,x} f \mapsto f(x) - (\Pi_X(f))(x).$$

The power function is then defined as $P_X(x) := \|\epsilon_{X,x}\|_{\mathcal{H}^*}$.

Theorem 1.21. *The two definitions for the power function are equivalent. P_X has the following properties*

1. $P_X(x) = 0 \ \forall x \in X$
2. $P_\emptyset(x)^2 = k(x, x) \ \forall x \in \Omega$
3. $P_\Omega(x) = 0 \ \forall x \in \Omega$
4. $0 = P_\Omega(x) \leq P_Y(x) \leq P_X(x) \leq P_\emptyset(x)$ for $X \subseteq Y \subseteq \Omega$
5. $P_X(x) = \inf_{g \in \mathcal{H}_X} \|k(x, \cdot) - g\|_{\mathcal{H}}$
6. $P_X(x) = \sup_{f \in \mathcal{H}, \|f\|_{\mathcal{H}} \leq 1, f(X) = \{0\}} f(x) \ \forall x \in \Omega$
7. $\forall x \in \Omega, f \in \mathcal{H}$

$$|f(x) - f_X(x)| = |f_X^\perp(x)| \leq P_X(x) \|f_X^\perp(x)\|_{\mathcal{H}} = P_X(x) \|f - f_X\|_{\mathcal{H}} \leq P_X(x) \|f\|_{\mathcal{H}}$$

Added remark. *General approximation goal: Split the approximation error into an error of the space and an error of the function (similarly to SC1).*

One aim is to generalize 7. to not rely on a specific point.

Proof. Due to $\langle \epsilon_{X,x}, \epsilon_{X,x} \rangle_{\mathcal{H}^*} = \langle R(\epsilon_{X,x}), R(\epsilon_{X,x}) \rangle_{\mathcal{H}}$ we have to show that the Riesz representer of $\delta_x \circ \Pi_X$ is $K_X(x, \cdot)$.

$$\begin{aligned} \langle f, R(\delta_x \circ \Pi_X) \rangle &= \delta_x \circ \Pi_X(f) = f_X(x) = \langle f_X, k(x, \cdot) \rangle_{\mathcal{H}} \\ &= \langle f_X, k_X(x, \cdot) + k_{X^\perp}(x, \cdot) \rangle_{\mathcal{H}} \\ &\stackrel{f_X \perp K_{X^\perp}}{=} \langle f_X, k_X(x, \cdot) \rangle_{\mathcal{H}} = \langle f - f_{X^\perp}, k_X(x, \cdot) \rangle_{\mathcal{H}} \\ &\stackrel{f_{X^\perp} \perp K_X}{=} \langle f, k_X(x, \cdot) \rangle_{\mathcal{H}} \end{aligned}$$

Proof of 7.:

$$\begin{aligned} f(x) - f_X(x) &= f_{X^\perp}(x) = \langle f_{X^\perp}, k(x, \cdot) \rangle_{\mathcal{H}} \\ &= \langle f_{X^\perp}, k(x, \cdot) - \underbrace{k_X(x, \cdot)}_{f_{X^\perp} \perp K_X} \rangle_{\mathcal{H}} \\ &\stackrel{\text{C.S.}}{\leq} \dots = \|f_{X^\perp}\|_{\mathcal{H}} P_X(x) \end{aligned}$$

Proof of 6.:

We see from the first inequality

$$P_X(x) \geq \sup_{\|f_{X^\perp}\|_{\mathcal{H}} \leq 1} |f_{X^\perp}(x)|$$

and equality must hold for the representer of $\epsilon_{X,x}$.

□

Notice the connection to operator norm approaches to 6.

Remark. *Consider the subspace $\mathcal{H}_X^* = \overline{\text{span}\{\delta_x \mid x \in X\}}$ of the dual space of \mathcal{H} . Then 5. can equivalently be given as*

$$P_X(x) = \inf_{\lambda \in \mathcal{H}_X^*} \|\delta_x\|_{\mathcal{H}^\perp} \quad (4)$$

Consider the interpolation of $f(x) = k(x, \cdot) \in \mathcal{H}$.

For $x \in \Omega$ we get for the interpolant in \mathcal{H}_X

$$k(x_k, x) = \sum_{j=1}^N u_j(x) k(x_j, x_k) \quad 1 \leq k \leq N \quad (5)$$

which has solution coefficients $u_j(x)$ as a function on Ω .

Added remark. If the kernel matrix is invertible, u_j is either 0 or 1. See Lagrange interpolation? In our setting it is enough to know that it exists, but might not be unique.

Theorem 1.22. If the kernel matrix is non-singular the u_j from 4 are $\in \mathcal{H}_X$ and there is a Lagrange basis $u_j(x_k) = \delta_{jk}, 1 \leq j, k \leq N$.

In general it still holds

This is sometimes called quasi-interpolation

$$f_X(x) = \sum_{j=1}^N u_j(x) f(x_j)$$

In the formula the influence of X and f are separated.

Proof. The first statement follows from 5. From the second:

$$\begin{aligned} f_X(x) &= \sum_{k=1}^N a_k k(x_k, x) \\ &= \sum_{k=1}^N a_k \sum_{j=1}^N u_j(x) k(x_j, x_k) \\ &= \sum_{j=1}^N a_j(x) \underbrace{\sum_{k=1}^N a_k k(x_j, x_k)}_{=f_{a,X}=f(x_j) \forall x_j \in X} = \sum_{j=1}^N u_j(x) f(x_j) \end{aligned} \quad \square$$

Theorem 1.23. The power function has the following explicit representation:

$$P_X(x) = k(x, x) - 2 \sum_{j=1}^N u_j(x) k(x_j, x) + \sum_{j=1}^N \sum_{k=1}^N u_j(x) u_k(x) k(x_j, x_k) = k(x, x) - k_X(x, x)$$

Proof. For $K_X \in \mathcal{H}_X$ $k_X(x, z) = \sum_{j=1}^N u_j(x) k(x_j, z)$

$$\begin{aligned} P_X^2(x) &= \langle k(x, \cdot) - k_X(x, \cdot), k(x, \cdot) - k_X(x, \cdot) \rangle_{\mathcal{H}} \\ &= k(x, x) - 2 \langle k(x, \cdot), \sum_{j=1}^N u_j(x) k(x_j, \cdot) \rangle_{\mathcal{H}} + \sum_{j=1}^N \sum_{k=1}^N u_j(x) \underbrace{u_k(x) k(x_j, x_k)}_{\text{with } a = k(x, x_j)} \\ &= k(x, x) - \underbrace{\sum_{j=1}^N u_j(x) k(x_j, x)}_{k_X(x, x)} \end{aligned} \quad \square$$

Consider $f_X(x) = \sum_{j=1}^N u_j(x) f(x_j)$ the interpolant on X .

Let us also consider arbitrary estimation formulas

$$(x, f) \mapsto \sum_{j=1}^N v_j(x) f(x_j)$$

with no assumptions on the scalars v_j . For fixed x we get for the error functional

$$f \mapsto f(x) - \sum_{j=1}^N v_j(x) f(x_j) = \left(\delta_x - \sum_{j=1}^N v_j(x) \delta_{x_j} \right) (f).$$

Ad for optimal estimation for all $f \in \mathcal{H}$, we should choose v_j to minimize the following expression:

$$V_{X,v}(x) := \left\| \delta_x - \sum_{j=1}^N v_j(x) \delta_{x_j} \right\|_{\mathcal{H}^*}.$$

Remember the dual form of the fifth property 4:

$$P_X(x) = \inf_{\lambda \in \mathcal{H}^*} \|\delta_x - \lambda\|_{\mathcal{H}^*}$$

we also saw that the function u_j are the solution.

We also see that the optimal error, in the worst case sense, is described to be the power function.

Theorem 1.24. *In the above sense, kernel based approximation yields the best linear estimation of unknown function values $f(x)$ from known function values $f(x_j)$ at points x_j .*

$$k(s, t) = \text{cov}(X_s, X_t)$$

Start of lecture 06
(25.04.24)

The kernel comes from a covariance, where for every t in some Ω , we have a random variable with finite second moments.

Consider X_t with zero mean. In this case, what we did is called (simple) Kriging

connection to
geo-statistics!

$V_{X,v}^2(x)$ can be understood as the variance of the prediction error.

Now define the error of some general linear predictor at x

$$\mathcal{E}_{X,V,x} := X_x - \sum_{j=1}^n V_j(x) X_{x_j}$$

Statistics pov: This is an
unbiased estimator

$$\begin{aligned} \mathbb{E}(\mathcal{E}_{X,V,x}^2) &= \underbrace{\text{cov}}_{=k}(X_x, X_x) - 2 \sum_{j=1}^N V_j(x) \text{cov}(X_x, X_{x_j}) + \sum_{j=1}^N \sum_{k=1}^N V_j(x) V_k(x) \text{cov}(X_{x_j}, X_{x_k}) \\ &= \langle \delta_x, \delta_x \rangle_{\mathcal{H}^*} - 2 \sum_{j=1}^N V_j \langle \delta_x, \delta_{x_j} \rangle_{\mathcal{H}^*} + \sum_{j=1}^N \sum_{k=1}^N V_j(x) V_k(x) \langle \delta_{x_j}, \delta_{x_k} \rangle_{\mathcal{H}^*} \\ &= V_{X,v}^2(x) \end{aligned}$$

Short revision of the condition number:

$$\begin{aligned} \kappa(A) &= \frac{|\lambda_{\max}(A)|}{|\lambda_{\min}(A)|} \\ Ax = b \frac{\Delta x}{|x|} &\leq \kappa(A) \frac{|\Delta b|}{|b|} \end{aligned}$$

Power function and stability

There is an uncertainty principle (Schabach 1995):

It is impossible to make the power function and the condition number of the kernel matrix small at the same time.

To make this precise: enrich $X = \{x_1, \dots, x_n\}$ with another point $X_0 = x$ and define $u_0(\cdot) = -1$

$$A = k(x_j, x_k)_{0 \leq j, k \leq N}$$

We can somehow fix this
via regularisation, which
in this case is both usefull
numerically and
meaningful from a
statistics point of view

$$\begin{aligned} u^\perp A u &= \sum_{j=0}^N \sum_{k=0}^N u_j(x) u_k(x) k(x_j, x_k) \\ &\stackrel{u_0=-1}{=} k(x, x) - 2 \sum_{j=1}^N u_j(x) k(x, x_j) + 2 \sum_{j=1}^N \sum_{k=1}^N u_j(x) u_k(x) k(x_j, x_k) \\ &\stackrel{\text{thm. 1.23}}{=} P_X^2(x) \end{aligned}$$

A has $n+1$ non-negative real eigenvalues $\lambda_0 \geq \lambda_1 \geq \dots \geq \lambda_N \geq 0$.

$$\lambda_N \|u\|_2^2 \leq u^\perp A u \leq \lambda_0 \|u\|_2^2$$

gives

$$P_X^2(x) \geq \lambda_N \left(1 + \sum_{j=1}^N |u_j(x)|^2 \right) \geq \lambda_N$$

Elimination of the special value of the point x gives

Theorem 1.25. The kernel matrix for N points $\{x_1, \dots, x_n\} = X$ has a smallest eigenvalue λ bounded from above by

$$\lambda \leq \min_{1 \leq j \leq N} P_{X \setminus \{x_j\}}^2$$

This holds for every $x \in \Omega$

Back to approximation

$$|f(x) - f_X(x)| \leq |P_X(x)| \|f\|_{\mathcal{H}}$$

Assume that any directional derivative of both f and f_X is bounded by some C :

$$|f(x) - f_X(x)| \leq \underbrace{|f(x_j) - f_X(x_j)|}_{=0} + 2C\|x - x_j\|_2 \leq 2Ch_{X,\Omega}$$

if the connecting line between x and x_j is in Ω

Definition 1.26. The fill distance of a set of points $X \subseteq \Omega$ for a bounded Ω is defined to be

$$h_{X,\Omega} = \sup_{x \in \Omega} \min_{1 \leq j \leq N} \|x - x_j\|_2$$

How large are the holes in Ω such that we don't hit an $x \in X$. It describes the size of the largest data free domain

To get bounds on the optimal approximation, we can consider some functions v_0, \dots, v_N instead of the optimal u_j .

$$P_X^2(x) \leq k(x, x) - 2 \sum_{j=1}^N v_j(x) k(x_j, x) + \sum_{j=1}^N \sum_{k=1}^N v_j(x) v_k(x) k(x_j, x_k) \quad (6)$$

The simplest case uses nearest neighbor reconstruction.

Assume that for each $x \in \Omega$ we pick a $x_{nn(x)}$ and define

$$v_j(x) = \begin{cases} 1 & j = nn(x) \\ 0 & \text{else} \end{cases}$$

Then

$$P_X^2(x) \leq k(x, x) - 2k(x_{nn(x)}, x) + k(x_{nn(x)}, x_{nn(x)})$$

$$(7) \quad d_k((x, x_{nn(x)}))^2 = \text{dist}(x, x_{nn(x)})^2$$

Theorem 1.27. For k positive semi-definite the power function on non-empty sets X of interpolation points satisfies

$$p_X^2(x) \leq \min_{x_j \in X} d_k(x, x_j)$$

with d_k as in (7).

Assume that we can prove $P_X(x) \leq CE(x, h)$ for all data sets X with fill distance h .

This implies (by theorem 1.21)

$$|f(x) - \sum_{j=1}^N u_j(x) f(x_j)| \leq CE(x, h) \|f\|_{\mathcal{H}}$$

We now introduce the error operator

$$E_x^y(f(y)) := f(x) - \sum_{j=1}^N v_j(x) f(x_j)$$

This simplifies the notation, once one understands what is happening :)

to set

$$\begin{aligned} P_X^2(x) &\leq (6) = k(x, x) + \sum_{j=1}^N v_j(x) \left(\sum_{k=1}^N v_k(x) k(x_j, x_k) - k(x_j, x) \right) \\ &= E_x^z(k(z, x)) - \sum_{j=1}^N v_j(x) E_x^z(k(z, x_j)) \\ &= E_x^y E_x^z(k(z, y)) \end{aligned}$$

We are after a bound such as

$$|E_x^y(f(y))| = \left| f(x) - \sum_{j=1}^N v_j(x) f(x_j) \right| \leq \mathcal{E}_{X,k}(h) \|Lf\|$$

We then bound the power function by

$$P_X^2(x) |E_x^y E_x^z k(z, y)| \leq \mathcal{E}_{X,k}(h) \|L^y E_x^z k(z, y)\| \leq \mathcal{E}_{X,k}^2 \|L^y\| \|L^z k(y, z)\|$$

assume the final expression makes sense.

First, univariate case, $\Omega = [a, b]$, $X = \{x_1, \dots, x_n\} \subset \Omega$. For a given $x \in \Omega$

$$X_x = \{x_j \in X \mid j \in N(x) \subseteq \{1, \dots, N\}\}$$

$f \in C^k$, Taylor polynomial at x_0

$$p(x) = \sum_{j=0}^{k-1} \frac{f^{(j)}(x_0)}{j!} (x - x_0)^j$$

for $|x - x_0| \leq h$ we have the local approximation error

$$|f(x) - p(x)| = \frac{|f^{(k)}|}{k!} |x - x_0|^k \leq C(f) h^k$$

From formula for interpolation in Newton form

$$\begin{aligned} f(x) - p_X(x) &= [x, X_x] f \prod_{x_j \in X_x} (x - x_j) \forall x \in [a, b] \\ &\leq \frac{\|f^{(k)}\|_{\infty, [a, b]}}{k!} \underbrace{\prod_{x_j \in X_x} (x - x_j)}_{\leq Ch^k} \end{aligned}$$

We can now use this to bound

$$p_X^2(x) \leq (Ch^k)^2 \sup_{a \leq z \leq b} \sup_{a \leq y \leq b} \left| \frac{\partial^k}{\partial z^k} \frac{\partial^k}{\partial y^k} k(a, b) \right|$$

Theorem 1.28. Assume a psd kernel k on $[a, b]$ that is k times differentiable. Then for every point set $X \subset [a, b]$ of at least k points with fill distance at most h the power function can be bounded

$$P_X(x) \leq C_k h^k \quad (7)$$

with C_k depending on k and X .

Definition 1.29. A compact domain $\Omega \subset \mathbb{R}^d$ allows

uniformly stable local polynomial reproduction of order $l \geq 1$, if there are positive constants c_1, c_2, h_0 , s.t. for all finite sets of points $X := \{x_1, \dots, x_n\}$ with fill distance $h_{X, \Omega} \leq h_0$ there are scalar functions $u_1(x), \dots, u_N(x)$ s.t.

1.

$$\sum_{j=1}^N u_j(x) p(x_j) = p(x)$$

for all polynomials $p \in \mathcal{P}_l^d$ and $x \in \Omega$.

2. $\sum_{j=1}^N |u_j(x)| \leq c_1$

3. $u_j(x) = 0$ if $\|x - x_j\|_2 > C_s h_{X, \Omega}$.

Start of lecture 07
(30.04.24)

This approximation error
holds for approximation
which recovers
polynomials locally!

If we are careful, we can
control the constant C
above!

Added remark. We will only handle positive definite kernels, the results generalizes for psd kernels with additional conditions on the kernel.

We know focus on positive definite kernels.

Theorem 1.30. Let $\Omega \subset \mathbb{R}^d$ and let $k : \Omega \rightarrow \mathbb{R}$ be a p.d. kernel. Let X be a set of N distinct points of Ω and define the quadratic form $Q : \mathbb{R}^N \rightarrow \mathbb{R}$ for any $x \in \Omega$ (see 1.23)

$$Q(u) = k(x, x) - 2 \sum_{j=1}^N a_j(x, x_j) + \sum_{j=1}^N \sum_{k=1}^N u_j u_k k(x_j, x_k)$$

The min of $Q(u)$ is given for the vector from 2.3 denoted as u^* with $u_j^*(x_k) = \delta_{jk}$ and we have

$$Q(u^*(x)) \leq Q(u)$$

Proof. With $b = [k(x, \cdot), \dots, k(x_k, \cdot)]^\top$ and $A_{i,j} = k(x_i, x_j)$, $i, j = 1, \dots, N$ we have

$$Q(u) = k(x, x) - 2b^\top u + u^\top A u.$$

The min is the solution of $Au = b$

Remark. in the positive definite setup only!, which is fulfilled by $u = u^*(x)$. \square

Theorem 1.31. Assume $\Omega \subseteq \mathbb{R}^d$ bounded and satisfies an interior cone condition. Suppose $k \in C^{2k}(\Omega \times \Omega)$ is a symmetric positive definite kernel with native space \mathcal{H} . Let f_X be the interpolant to $f \in \mathcal{H}$ on the set $X = \{x_1, \dots, x_n\}$. Then there are positive constants h_0, C (independent of x, f, k) s.t.

$$|f(x) - f_X(x)| \leq C h_{X,\Omega}^k \sqrt{C_k(x)} \|f\|_{\mathcal{H}}$$

provided $h_{X,\Omega} \leq C h_0$. Here

$$C_k(x) = \max_{|\beta|=2k} \sup_{x, y \in \Omega \cap B(x, C_2 h_{X,\Omega})} |D_2^\beta(k(x, y))|$$

First some further notation:

For $\beta = (\beta_1, \dots, \beta_d) \in \mathbb{N}_0^d$ we define

$$D^\beta = \frac{\partial^{|\beta|}}{(\partial x_1)^{\beta_1} \dots (\partial x_d)^{\beta_d}}$$

with D_2^β we indicate that D^β is applied to $k(x, \cdot)$. Multivariate Taylor expansion of $k(x, \cdot)$ centered at x :

$$k(x, y) = \underbrace{\sum_{|\beta| < 2l} \frac{D_2^\beta k(x, x)}{\beta!} (y - x)^\beta}_{\sum_\beta T(x, \beta)} + R(x, y)$$

$$R(x, y) = \sum_{|\beta|=2l} \frac{D_2^\beta k(x, \xi_{x,y})}{\beta!} (y - x)^\beta$$

where $\xi_{x,y}$ lies on the line connecting x, y .

Proof. Theorem 1.21 $|f(x) - f_X(x)| \leq P_X(x) \|f\|_{\mathcal{H}}$. We know $P_X^2(x) = Q(u^*(x))$. Given the interior cone condition, we can obtain a $u(x)$ that has polynomial precision of degree $l \geq 2k - 1$. For those u we see

$$P_X^2(x) \leq Q(u) = k(x, x) - 2 \sum_{k=1}^N u_k(x, x_k) + \sum_{j=1}^N \sum_{k=1}^N u_j u_k k(x_j, x_k)$$

Apply Taylor expansion centered at x to $K(x, \cdot)$ and centered at x_j to $k(x_j, \cdot)$

$$Q(u) = k(x, x) - 2 \sum_{k=1}^N u_k \left[\sum T(x, \beta)(x_k - x)^\beta + R(x, x_k) \right] + \sum_{j=1}^N \sum_{k=1}^N u_j u_k \left[\sum T(x_j, \beta)(x_k - x_j)^\beta R(x_j, x_k) \right]$$

We identify $p(z) = (z - x)^\beta$, so that $p(x) = 0$ unless $\beta = 0$.
With the polynomial reproduction of u , this simplifies to

$$Q(u) = k(x, x) - \underbrace{2k(x, x)}_{\beta=0} - 2 \sum_{k=1}^N u_k R(x, x_k) + \sum_{j=1}^N u_j \sum T(x_j, \beta)(x - x_j)^\beta + \sum_{j=1}^N \sum_{k=1}^N u_j u_k R(x_j, x_k)$$

Look at Taylor expansion

$$\sum T(x_j, \beta)(x - x_j)^\beta = k(x_j, x) - R(x_j, x)$$

This gives

$$Q(u) = k(x, x) - \sum_k u_k \left[2R(x, x_k) - \sum_{j=1}^N u_j R(x_j, x_k) \right] + \sum_{j=1}^N u_j [k(x_j, x) - R(x_j, x)]$$

Once more Taylor: $k(x_j, x) = k(x, x_j) = \sum_\beta T(x, \beta)(x - x_j)^\beta + R(x, x_j)$

$$= 0 - \sum_k u_k \left[R(x, x_k) - \sum_{j=1}^N u_j R(x_j, x_k) + R(x_k, x) \right]$$

The polynomial reproduction gives

$$\sum_k |u_k| \leq C_1$$

uniform stability

For $u_j \neq 0$ (with 3.) we have $\|x - x_j\|_2 \leq C_2 h_{X, \Omega}$ and it holds $\|x_j - x_k\| \leq 2C_2 h_{X, \Omega}$. Thereby all three can be bounded by an expression such as $Ch_{X, \Omega}^{2k} C_k(x)$.

The interior cone condition shows that the ball remains inside, so that $C_k(x)$ is well defined.

Combining these bounds and taking the square root gives the bound for the power function. \square

Other interesting functionals:

Derivatives: $\lambda(f) = \frac{\partial f}{\partial x_j}(x)$

Integration: $\lambda(f) = \int_\Omega f(y) dy$

Consider $\Lambda \subseteq \mathcal{H}^*$ that generalizes the role of the point set X and the associated δ_{x_i} .

Start of lecture 08

(12.05.24) that we will focus

on for now

this yields quasi monte

carlo integration

$$\{(x_i, \lambda_i f)\}_{i=1}^N$$

more general data instead of $\{(x_i, f(x_i))\}$.

Dirichlet boundary value problem :

$$\begin{aligned} Lu &= f & \text{on } \Omega \subset \mathbb{R}^d \\ u &= g & \text{on } \partial\Omega \end{aligned} \quad (8)$$

with L a linear differential operator.

Collocation is a general approach that discretizes this by

The exact solution u^* of 8 solves ??.

Consider $U \subseteq \mathcal{H}$ of dimension at least N . Let $u \in U$.

Assume $\lambda_i \in \mathcal{H}^*$ of the form $\lambda_i = \delta_{x_i} \circ D^{\alpha(i)}$. Further assume that $\alpha(i) \neq \alpha(k)$ if $x_i = x_k$.

Then the λ_i are linearly independent on the native space of a p.d. kernel (Compare: Wendland).

We can proceed via **Hermite interpolation**. We assume $\{(x_i, \lambda_i f)\}_{i=1}^N, x_j \in \Omega, \lambda_j \in \Lambda$ with

$\Lambda = \{\lambda_1, \dots, \lambda_n\}$ linearly independent set of continuous linear functionals.

$$u(x) = \sum_{j=1}^N a_j \lambda_j^{(j)} k(x_j, x) \quad x \in \mathbb{R}^d$$

that satisfies

$$\lambda_j u = \lambda_j f \quad j = 1, \dots, N \quad f \in \mathcal{H}$$

$\lambda_i^{(1)}$ indicates that the functional acts on the first argument of k .

The LES has entries $A_{jk} \lambda_j^{(2)} \lambda_k^{(1)} k(x_k, x_j)$ for $j, k = 1, \dots, N$.

Example. Denote the centers of radial basis functions (RBFs) by ξ_j and denote the data location \underline{x}_j . Given

$$\{(\underline{x}_j, f(\underline{x}_j))\}_{i=1}^p$$

and

$$\{(\underline{x}_j, \frac{\partial f}{\partial x})(\underline{x}_j)_{j=p+1}^N\}$$

Thus

$$\lambda_j = \begin{cases} \delta_{x_j} & j = 1, \dots, p \\ \delta_{x_j} \frac{\partial}{\partial x} & j = p+1, \dots, N \end{cases}$$

with $k(\underline{x}_j, \underline{x}_k) = \varphi(\|\underline{x}_j - \underline{x}_k\|)$

$$\begin{aligned} u(\underline{x}) &= \sum_{j=1}^N a_j k(\cdot, \underline{x}) = \sum_{j=1}^p a_j k(\cdot, \underline{x}) + \sum_{j=p+1}^N a_j \frac{\partial}{\partial \xi_1} k(\xi, \underline{x}) \\ &= \sum_{j=1}^N a_j k(\xi_j, \underline{x}) - \sum_{j=p+1}^N a_j \frac{\partial}{\partial x} k(\xi_j, \underline{x}) \end{aligned}$$

since system matrix after ... u into $\lambda_j u = \lambda_j f$

$$\begin{bmatrix} K & K_\xi \\ K_x & K_{xx} \end{bmatrix}$$

with $K_{jk} = K(\xi_k, x_j) = \varphi(\|\xi_k - x_j\|)$ and

$$\begin{aligned} K_{\xi, jk} &= \frac{\partial \varphi}{\partial \xi} \varphi(\|\xi_k - x_j\|) = -\frac{\partial \varphi}{\partial x}(\|\xi_k - x_j\|) \\ K_{x, jk} &= \frac{\partial \varphi}{\partial x}(\|\xi_k, x_j\|) \\ K_{xx, jk} &= \frac{\partial^2 \varphi}{\partial x^2}(\|\xi_k, x_j\|) \end{aligned}$$

Added remark. We can do everything we did, but replacing point evaluations with point evaluations of (weak) derivatives to get a similar theory weak derivatives, without relying on point evaluations.

To measure errors we use generalized power functions

$$P_\Lambda(\mu) = \|\mu_\mu \circ \Pi_\Lambda\|_{\mathcal{H}^*}$$

where we use $\mu \in \mathcal{H}^*$ to measure the error instead of point evaluation functionals

$$\mu(f - f_\lambda) = (\mu - \mu \circ \Pi_\Lambda)f.$$

$$\begin{aligned} Lu &= f & \Omega &\subseteq \mathbb{R}^d \\ u &= g & \gamma &= \partial\Omega \end{aligned}$$

\underline{x}_j is the multiindex, x is the first entry: $\underline{x}_j = (x, y)$

CAREFUL: In this calculation a lot of stuff (everything with a non-constant index?) should have an underline? He added them inconsistently and often corrected himself, so I couldn't keep up.

with kernel based collocation method.

We use

$$u(x) = \underbrace{\sum_{j=1}^p a_j k(x_j, x)}_{\text{boundary } B} + \underbrace{\sum_{j=p+1}^N a_j L^{(1)} k(x_j, x)}_{\text{interior } I} \quad (9)$$

$X = B \cup I$.

We get a block matrix

$$A = \begin{bmatrix} K & L^{(1)} K \\ L^{(2)} K & L^{(2)} L^{(1)} K \end{bmatrix}$$

where $Au = \begin{bmatrix} g \\ f \end{bmatrix}$.

with

$$\left\{ \begin{array}{lll} K_{j,k} & = k(x_j, x_k) & x_j, x_k \in B \\ L^{(1)} K_{j,k} & = L^{(1)} K(\tilde{x}_k, x_j) & x_j \in B, \tilde{x}_k \in I \\ L^{(2)} K_{j,k} & = L^{(1)} K(x_k, \tilde{x}_j) & x_k \in B, \tilde{x}_j \in I \\ L^{(2)} L^{(1)} K_{j,k} & = L^{(2)} L^{(1)} K(\tilde{x}_k, \tilde{x}_j) & \tilde{x}_j, \tilde{x}_k \in U \end{array} \right.$$

Same structure as for Hermite interpolation: Non-singular if $\delta_{x_i} K, LK$ are linearly independent holds for suitable K .

Theorem 1.32. Let $\Omega \subseteq \mathbb{R}^d$ be a polygonal and open domain. Let L be a second order elliptic differential operator with coefficients in $C^{2(k-2)}(\bar{\Omega})$ that either vanishes on $\partial\Omega$ or have no zero here.

Suppose that $k \in C^{2k}(\mathbb{R}^d \times \mathbb{R}^d)$ is a positive definite kernel. Assume that $Lu = f$ in the system 8 has a unique solution $u \in \mathcal{N}_k(\Omega)$ for some $f \in C(\Omega, g \in C(\Gamma))$. Let \hat{u} be the approximation 10. Then

$$\|u - \hat{u}\|_{L_\infty(\Omega)} \leq Ch_{L,\Omega}^{k-2} \|u\|_{\mathcal{N}_k(\Omega)}$$

and

$$\|u - \hat{u}\|_{L_\infty(\partial\Omega)} \leq Ch_{B,\partial\Omega}^k \|u\|_{\mathcal{N}_k}$$

Sketch of the proof. For interior essentially as before for $L(u) - L\hat{u}$ and LLk is p.d. for p.d. k .

□

Start of lecture 09
(07.05.24)

1.3 Kernel Methods for prediction

Added remark. Statistics: finding structure in the data, while ML tries to use the same math to make predictions

Start of lecture 10
(14.05.24)

Definition 1.33. Let Ω, Σ be a measurable space and $Y \subseteq \mathbb{R}$ be a closed subset. Denote by $(x, y, f(x)) \in \Omega \times Y \times \mathbb{R}$ the triplet consisting of attributes (or features) x , an observation y and a prediction y .

A function $l : \Omega \times Y \times \mathbb{R} \rightarrow [0, \infty)$ is called a loss function if it is measurable and $l(x, y, y) = 0$ holds for all $x \in \Omega, y \in Y$.

Example. • l_2 loss, which relates to the mean

• l_1 loss, which relates to the median

• $l_H = \begin{cases} \frac{1}{2}\xi^2 & |\xi| \leq \sigma \\ \sigma|\xi| - \frac{1}{2}\sigma^2 & \text{otherwise} \end{cases}$ hubert loss

- $l_\epsilon(\xi) = \max(|\xi| - \epsilon, 0) =: |\xi|_\epsilon$ ϵ -sensitive loss

Weighting loss functions might be useful to emphasize important data points! For Classification:

- $l(x, y, f(x)) = \begin{cases} 0 & y = f(x) \\ 1 & \text{otherwise} \end{cases}$
- $l(x, y, f(x)) = \begin{cases} 0 & y = \text{sgn}(f(x)) \\ 1 & \text{otherwise} \end{cases}$
- $l(x, y, f(x)) = l_1(1 + \exp(-yf(x)))$ logistic loss, relates to probability
- soft margin / hinge loss $\max(1 - yf(x), 0)$, important for SVMs

much of the past research efforts were focused on SVMs

Added remark. l_1, l_2 losses penalize overestimation in classification problems, therefore the other losses might be a better idea.

Both hinge loss and logistic loss functions give a way to rank the data.

Definition 1.34. Let $l : \Omega \times Y \times \mathbb{R} \rightarrow [0, \infty)$ be a loss function and \mathbb{P} be a probability measure on $\Omega \times Y$. Then, for a measurable function $f : \Omega \rightarrow \mathbb{R}$, the **expected l-risk** is defined by

$$R_{l,p}(f) := \int_{\Omega \times Y} l(x, y, f(x)) d\mathbb{P}(x, y) = \int_{\Omega} \int_Y l(x, y, f(x)) d\mathbb{P}(y|x) d\mathbb{P}_{\Omega}(x).$$

For a given data set $D := \{x_j, y_j\}_{j=1}^N$ with $x_j \in \Omega, y_j \in Y$ we can define the empirical measure

$$P_{\text{imp}}(x, y) = \frac{1}{N} \delta_{x_j, y_j}$$

Definition 1.35. The **empirical l-risk** of a function $f_{\Omega} \rightarrow \mathbb{R}$ is defined as

$$R_{l,\text{emp}}(f) = \int_{\Omega \times Y} l(x, y, f(x)) dP_{\text{imp}}(x, y) = \frac{1}{N} \sum_{j=1}^N l(x_j, y_j, f(x_j))$$

Added remark. Regularization via penalty terms or by enforcing some sparsity condition on the α_j .

We will assume that $R_{l,\text{emp}}(f)$ is continuous on f .

Operator inversion lemma

$$R_{l,\text{reg}}(f) = R_{l,\text{emp}}(f) + \lambda s(f)$$

Here, the smoothness or sparsity is enforced by the regularization term s .

Often s is convex. The regularization parameter λ balances the empirical error and the regularization.

Theorem 1.36 (Representer theorem). Let $s : [0, \infty) \rightarrow \mathbb{R}$ be a strictly monotone increasing function, Ω be a set, \mathcal{H} a RKHS over Ω and let $l : \Omega \times Y \times \mathbb{R}$ be a continuous loss function. Then, for given data $D = \{(x_j, y_j)\}_{j=1}^N, x_j \in \Omega, y_j \in Y$ and $\lambda > 0$, each minimizer $f \in \mathcal{H}$ of the regularized empirical risk

$$R_{l,\text{reg}}(f) = \frac{1}{N} \sum_{j=1}^N l(x_j, y_j, f(x_j)) + \lambda s(\|f\|_{\mathcal{H}})$$

admits a representation $f(x) = \sum_{j=1}^N \alpha_j k(x_j, x)$, that is $f \in \mathcal{H}_X, X = \{x_1, \dots, x_N\}$.

Proof. w.l.o.g. we assume $s(\|f\|_{\mathcal{H}}) = \bar{s}(\|f\|_{\mathcal{H}})$.

We decompose any $f \in \mathcal{H}$ into $f_X \in \mathcal{H}_X$ and $f_{X^\perp} \in \mathcal{H}_{X^\perp}$ (Theorem 1.16)

$$f = f_X + f_{X^\perp} = \sum_{j=1}^N \alpha_j k(x_j, x) + f_{X^\perp}$$

We know

$$\langle f_{X^\perp}, k(x_k, \cdot) \rangle_{\mathcal{H}} = 0$$

with the reproduction equation we write

$$f(x_i) = \langle f, k(x_i, \cdot) \rangle_{\mathcal{H}} = \sum_{j=1}^N \alpha_j k(x_i, x_j) + \langle f_{X^\perp}, k(x_i) \rangle_{\mathcal{H}} = \sum_{j=1}^N \alpha_j k(x_i, x_j)$$

The loss term part does not depend on f_{X^\perp} . Further, for all f_{X^\perp} it holds

$$s(\|f\|_{\mathcal{H}}) = \bar{s}(\|f\|_{\mathcal{H}}^2 + \|f_{X^\perp}\|^2) \geq \bar{s}(\|f_X\|_c H^2)$$

Therefore for any fixed $\alpha \in \mathbb{R}^N$ the objective is minimal if $f_{X^\perp} = 0$. This has to hold for any minimizer f . □

Remark. $f + q$, $f \in \mathcal{H}$, $q \in \text{span}\{\varphi_p\}$ For this setup a corresponding representer theorem does hold.

Remark. If both loss function and s are convex, one has a single minimum.

Consider regularized least squares regression,

$$R_{l_2, \text{reg}}(f) = \frac{1}{N} \sum_{j=1}^N (f(x_j) - y_j)^2 + \frac{\lambda}{2} \|f\|_{\mathcal{H}}^2$$

where $f(x) = \sum_{j=1}^N \alpha_j k(x_j, x)$.

$$\frac{1}{N} \sum_{j=1}^N \left(\sum_{k=1}^N \alpha_k k(x_k, x_j) - y_j \right)^2 + \frac{\lambda}{2} \sum_{j,k=1}^N \alpha_j \alpha_k k(x_j, x_k)$$

derivation w.r.t. α_k yields:

$$\frac{2}{N} \sum_{j=1}^N \left(\sum_{k=1}^N \alpha_k k(x_k, x_j) - y_j \right) k(x_k, x_j) + \frac{\lambda}{2} \sum_{j=1}^N \alpha_j (x_j, x_k)$$

All together for all a_k this holds and gives

$$\begin{aligned} 0 &= \frac{2}{N} K(K\alpha - Y) + \frac{\lambda}{2} K\alpha \\ \implies K(K + \lambda NI)\alpha &= KY \\ \implies (K + \lambda NI)\alpha &= Y \end{aligned}$$

In $L^2(\Omega)$ we have $\langle f, g \rangle = \int_{\Omega} f g dx$. We aim to write $\langle f, g \rangle_{\mathcal{H}} = \langle Sf, Sg \rangle_{L^2(\Omega)} = \int_{\Omega} Sf(x) \cdot Sg(x) dx$, where S is called a **regularization operator**.

Start of lecture 11
(16.05.24)

Definition 1.37. A **regularization operator** S is defined as a linear map from the space of functions

$$\{f \mid f : \Omega \rightarrow \mathbb{R}\}$$

into a space D equipped with a scalar product. The regularization term $s(f)$ takes the form

$$s(f) := \langle S(f), S(f) \rangle_D.$$

sometimes we multiply $\frac{1}{2}$
to s

Remark. Since we can always define $\tilde{S} = (S^* S)^{\frac{1}{2}}$ and

$$\langle f, S^* S f \rangle_D = \langle Sf, Sf \rangle_D$$

we can assume S is a positive semidefinite (regularization) operator.

Definition 1.38. Let $k : \Omega \times \Omega \rightarrow \mathbb{R}$ be continuous, Ω be a compact domain, ν be a Borel measure and $L_2^\nu(\Omega)$ be the Hilbert space of square integrable functions on Ω . We define the **integral operator** $T_k : L_2^\nu(\Omega) \rightarrow L_2^\nu(\Omega)$ by

$$T_k(f)(\cdot) = \int_{\Omega} k(x, \cdot) f(x) d\nu$$

and we call k the kernel of T_k .

Mercer kernels

$$K(x, y) = \sum_{i=1}^{\infty} \lambda_i \phi_i(x) \phi_i(y)$$

with eigenvalues λ_j and eigenfunctions ϕ_j w.r.t. the eigenproblem

$$\langle k(x, \cdot), \phi \rangle_{\sigma} = \int_{\Omega} k(x, y) \phi(y) \sigma(y) dy = \lambda \phi(x) \iff (T_k \phi)(x) = \lambda \phi(x)$$

Fredholm integration operator of the second kind

Definition 1.39. Given a linear ... or partial differential operator \mathcal{L} on $\Omega \subseteq \mathbb{R}^d$, the **Green's kernel** g of \mathcal{L} is defined as the solution of

$$\mathcal{L}(g)(x, z) = \delta(x - z)$$

$$\int f(z) \delta(x - z) dz = f(x).$$

The Green kernel is not uniquely defined this way, so one adds homogenous boundary conditions e.g. $g(x, z) = 0$ for $x \in \partial\Omega$, $\lim_{\|x\| \rightarrow \infty} g(x, z) = 0$.

Solutions of $\mathcal{L}u = f$ with appropriate boundary conditions can now be given as

$$u(x) = \int_{\Omega} g(x, z) f(z) dz$$

Check

$$\begin{aligned} \mathcal{L}u(x) &= \mathcal{L} \int_{\Omega} g(x, z) f(z) dz \\ &= \int_{\Omega} \mathcal{L}g(x, z) f(z) dz \\ &= \int_{\Omega} \delta(x - z) f(z) dz = f(x) \end{aligned}$$

We can ...

$$Gf(x) = \int_{\Omega} g(x, z) f(z) dz$$

as the “inverse” of the differential operator \mathcal{L} , i.e.

$$\mathcal{L}u = f \iff u = Gf$$

Example (Brownian Bridge kernel). $\Omega = [0, 1]$, consider bvp (boundary value problem)

$$-u''(x) = f(x), u(0) = 0 = u(1)$$

The corresponding Green's kernel is

$$g(x, z) = \min(x, z) - xz = \begin{cases} x(1 - z) & x \leq z \\ z(1 - x) & x \geq z \end{cases}$$

We observe that for g it must hold

$$\mathcal{L}g(x, z) = 0$$

for $x \neq z$, z fixed.

- $g(0, z) = g(1, z) = 0$
- g is continuous along the diagonal $x = z$
- for fixed $z \in (0, 1)$ one observes for $\frac{dg}{dx}$ a jump disc. at $x = z$ of the form

$$\lim_{x \rightarrow z^-} \frac{dg}{dx}(x, z) = 1 + \lim_{x \rightarrow z^+} \frac{dg}{dx}(x, z)$$

There is a further connection to the Brownian bridge of stochastic analysis? Not just the end points

Remark. Whenever \mathcal{L} is a self adjoint differential operator, the corresponding Green's kernel is symmetric and the integral operator G is self adjoint.

Theorem 1.40. For every RKHS \mathcal{H} with reproducing kernel k there exists a corresponding regularization operator $S : \mathcal{H} \rightarrow D$ s.t. for all $f \in \mathcal{H}$

$$f(x) = \langle Sk(x, \cdot), Sf(\cdot) \rangle_D \quad (10)$$

The second statement is much more interesting, the first follows from $S = Id$

In particular

$$\langle Sk(x, \cdot), Sk(y, \cdot) \rangle_D = k(x, y)$$

likewise, for every regularization operator $S : \mathcal{F} \rightarrow D$, where \mathcal{F} is some function space equipped with a scalar product and with corresponding Green's kernel f on S^*S , there exists a corresponding RKHS \mathcal{H} with reproducing kernel K , s.t. both equations are fulfilled.

Proof. First direction: $S = Id, D = \mathcal{H}$.

Second direction:

$$f(x) = \langle f, \delta(x - z) \rangle_{\mathcal{F}} = \langle f, \mathcal{L}g_x \rangle_{\mathcal{F}} = \langle f, S^*Sg_x \rangle_{\mathcal{F}} = \langle Sf, Sg_x \rangle_D$$

for all $f \in S^*S\mathcal{F}$, where g_x is the Green's kernel for S^*S and natural boundary conditions. Further we have with $f = g_z$

$$g_z(x) = \langle Sg_x, Sg_z \rangle = \langle Sg_z, Sg_x \rangle = g_x(z)$$

In this sense g is symmetric and we write

$$k(x, z) = g_z(x).$$

We observe that $x \rightarrow Sg_x$ is actually a feature map, i.e. $\langle Sg_x, Sg_z \rangle_D$. Since kernels arising from feature maps result in Gram matrices for the kernel matrix we set that K is a p.s.d..

It can be seen that the corresponding RKHS is the closure of

$$\{f \in S^*S\mathcal{F} \mid \|Sf\|_D^2 < \infty\}$$

□

To simplify, we consider the full space kernel, i.e. without boundary / decay conditions. For $g_z(x) = k(x, z)$, g for the differential operator \mathcal{L} , we observe

$$\begin{aligned} \mathcal{L} \int_{\Omega} k(x, z) \sigma(x) dx &= \mathcal{L} \lambda \phi(z) \\ \iff \underbrace{\int_{\Omega} \delta(x - z) \phi(x) \sigma(x) dx}_{\phi(z) \sigma(z)} &= \lambda \mathcal{L} \phi(z) \\ \implies \dots \implies L\phi(z) &= \frac{1}{\lambda} \phi(z) \sigma(z) \end{aligned}$$

For simplicity we assume that \mathcal{L} has no Eigenvalue 0.

Example. We have $\int_{\Omega} k(x, z) \phi(x) \sigma(x) dx = \lambda \phi(z)$ with $\sigma = 1, k(x, z) = \min(x, z) - xz$ on $\Omega = [0, 1]$.

This gives

$$\int_0^z x \phi(x) dx + \int_z^1 z \phi(x) dx - \int_0^1 xz \phi(x) dx = \lambda \phi(z)$$

Now apply $\mathcal{L} = -\frac{d^2}{dz^2}$ to the equation:

$$\begin{aligned} \frac{d}{dz} \left(z \phi(z) - \int_1^z \phi(x) dx - z \phi(z) - \int_0^1 x \phi(x) dx \right) &= \lambda \phi''(z) \\ \iff \frac{1}{\lambda} \phi(z) &= \phi''(z) \end{aligned}$$

Theorem 1.41. Given a regularization operator S with an expansion of S^*S into a discrete normalized eigendecomposition with eigenvalues and eigenfunctions γ_i, ϕ_i , we define a kernel with

$$k(x, y) = \sum_{i, \gamma_i \neq 0} \frac{d_i}{\gamma_i} \phi_i(x) \phi_i(z)$$

where $d \in \{0, 1\}$ for all i and $\sum_{i=1}^{\infty} \frac{d_i}{\gamma_i}$ is ... Then k satisfies theorem 1.40

$$\langle Sk(x, \cdot), Sk(z, \cdot) \rangle_D = k(x, z) = \langle k(x, \cdot), k(z, \cdot) \rangle_{\mathcal{H}}$$

where the RKHS is given by $\text{span}\{\phi_i | d_i = 1\}$

Proof.

$$\begin{aligned} \langle Sk(x, \cdot), Sk(z, \cdot) \rangle &= \langle k(x, \cdot), S^* Sk(z, \cdot) \rangle \\ &= \left\langle \sum_i \frac{d_i}{\gamma_i} \phi_i(x) \phi_j(\cdot), S^* S \sum_i \frac{d_i}{\gamma_i} \phi_i(x) \phi_j(\cdot) \right\rangle \\ &= \sum_{j, k} \frac{d_j}{\gamma_j} \frac{d_k}{\gamma_k} \phi_j(x) \phi_k(z) \langle \phi_j(\cdot), \underbrace{S^* S \phi_k(\cdot)}_{\gamma_k \phi_k} \rangle \\ &\stackrel{\text{ONB}}{=} \sum_j \frac{d_j}{\gamma_j^2} \gamma_j \phi_j(x) \phi_j(z) = k(x, z) \end{aligned}$$

From the construction of k follows the statement of the span. □

Start of lecture 12
(28.05.24)

1.4 Model selection

To estimate the predictive performance of a learned model, we use train- and test/ validation data. Where D_t, D_V are the training and the validation data sets, respectively. The **empirical risk** on D_V is used to measure the generalization performance.

In case of hyperparameters, a simple strategy is to use a selection of hyperparameters, train models, and pick the hyperparameters with the best empirical risk.

Added remark. For time series data a random split might be bad (if there is a drift), therefore we might prefer taking years up to year x !

Added remark. If we use hyperparameter search, we also have to split the data into 3 parts. One for training, 1 for hyperparameter search and one to estimate the predictive power.

If the number of data is small, the measured performance on the validation data can have a large uncertainty. A strategy to use is k -fold cross validation.

We use k disjoint equally sized subsets, use each one once for D_V and the $k - 1$ other ones as training data D_t . The average performance over the k runs is then the performance measure.

80/20, 70/30 are common splits, this does depend on the size of the dataset! Might not be the best idea in practice, where training may take weeks! Notice how the run time scales exponentially in the number of hyperparameters

Added remark. In the deep learning setting run the same training with different seeds, then take the average.

For $k = N$ is known as **leave-one-out** cross validation.

Definition 1.42. There is a exact definition in the script (this was not done in the lecture).

Definition 1.43. A **Gaussian process** is a collection of random variables, any finite number of which have a joint Gaussian distribution.

Mean $m(x) = \mathbb{E}(f(X))$ and covariance $k(x, z) = \mathbb{E}((f(x) - m(x))(f(z) - m(z)))$

$$f(x) = GP(m(x), k(x, z))$$

Key example: Gaussian kernel:

$$\text{cov}(f(x), f(z)) = k(x, z) = \exp(-\frac{1}{2\sigma^2} \|x - z\|^2)$$

Added remark. We can then sample from this space of functions! The better fitted the space is to our data, the better our predictions.

Choose $X_* := \{x_1, \dots, x_N\}$ and generate a random process with a covariance matrix

$$f_* = \mathcal{N}(0, k(X_*, X_*))$$

Consider noisy data with additive gaussian noise ϵ , with variance σ_N^2 ;

$$y = f(x) + \epsilon$$

The prior on the noisy observations becomes

$$\text{cov}(y_p, y_q) = k(x_p, x_q) + \sigma_N^2 \delta_{pq}.$$

or

$$\text{cov}(Y) = k(X, X) + \sigma_N^2 I$$

Joint distribution of observed target and f at test locations

$$\begin{bmatrix} y \\ f_* \end{bmatrix} \sim \mathcal{N} \left(0, \begin{bmatrix} k(x, x) + \sigma_N^2 I & k(x, x_*) \\ k(x_*, x) & k(x_*, x_*) \end{bmatrix} \right)$$

We need to restrict this joint distribution to contain only those functions that “agree” with the observed data.

$$f_* \mid X, Y, x_* \sim \mathcal{N} \left(k(x_*, X) [k(X, X) + \sigma_N^2 I]^{-1} Y, k(x_*, x_*) - k(x_*, X) [k(X, X) + \sigma_N^2 I]^{-1} k(X, x_*) \right)$$

Boil down to one evaluation x_l we have

$$f(x_l) = k(X, x_l)^\top (k(X, X) + \sigma_N^2 I)^{-1} Y = \sum_{i=1}^N \alpha_i k(x_i, x_l)$$

$$V(f) = k(x_l, x_l) - k(X, x_l)^\top [k(X, X) + \sigma_N^2 I]^{-1} k(X, x_l)$$

Added remark. What is a length scale? Variance in data points?

Consider the marginal likelihood or evidence

$$p(Y \mid X) = \int \underbrace{p(Y \mid f, X)}_{\text{likelihood}} \underbrace{p(f \mid X)}_{\text{prior}} df$$

One can obtain for the log marginal likelihood

$$\log(p(Y \mid X)) = -\frac{1}{2} Y^\top (K + \sigma_N^2 I)^{-1} Y - \frac{1}{2} \log(K + \sigma_N^2 I) - \frac{N}{2} \log(2\pi)$$

θ hyperparameters of K :

$$\log(p(Y | X, \theta)) = -\frac{1}{2} \underbrace{Y^\top (K_\theta + \sigma_N^2 I)^{-1} Y}_{\text{data fit}} - \underbrace{\frac{1}{2} \log(K_\theta + \sigma_N^2 I)}_{\text{complexity penalty}} - \underbrace{\frac{N}{2} \log(2\pi)}_{\text{Normalization parameter}}$$

One aims to optimize the evidence of the data, in this Bayesian optimization one can also include the hyperparameters.

Added remark. *Bayesian optimization. Optimize which data point would minimize unexplained variance?*

$k(x, y) = \sum_{i=1}^{\infty} \gamma_i \phi_i(x) \phi_i(y)$ a mercel kernel. For $f \in \mathcal{H}_K$

$$f(x) = \sum_{i=1}^{\infty} c_i \phi_i(x)$$

$$\langle f, f \rangle_{\mathcal{H}_K} = \langle Sf, Sf \rangle_{L^2} = \langle f, Sf \rangle_{L^2} = \sum_{i=1}^{\infty} \frac{c_i^2}{\gamma_i} < \infty$$

The rough terms ϕ_i have smaller prior variance and are therefore more strongly penalized by the regularization.

Start of lecture 13
(04.06.24)

Added remark. • *Motivation: Finding a classifier, specifically by separating data by a hyperplane!*

- *Idea for SVMs: find a separating hyperplane while maximizing the margin between the hyperplane and the data points.*
- *Outliers or not linearly separable data is a problem!*
- *For SVMs slack variables can be a good idea to counteract some of those problems (where C is another regularization parameter)*
- *We can also use feature maps to embed our data into a higher dimensional space, where it might be linearly separable*
- *Not a unique way to choose such a feature map*
- *Sadly the number of support vectors is linear in the number of data points*

$$\begin{aligned} \xi_i &\geq 1 - y_i(\langle w_i, \Phi(x_i) \rangle + b), \quad \xi_i \geq 0 \\ \xi_i &\geq \max(0, y_i(\langle w_i, \Phi(x_i) \rangle + b)) \\ &= l_h(y_i, \langle w_i, \Phi(x_i) \rangle + b) \end{aligned}$$

It is therefore related to the hinge loss!
 (w, b) :

$$\begin{aligned} f_{(w,b)}(x) &= \langle w, \Phi(x) \rangle + b \\ \min_{(w,b)} \underbrace{\langle w, w \rangle}_{\|f\|_{\mathcal{H}}} + C \sum_{i=1}^N l_h(y_i, \langle w_i, \Phi(x_i) \rangle + b) \end{aligned}$$

This gives the RKHS view on SVMs.
This, once more, motivates

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle$$

$$\begin{aligned}\|\Phi(x) - \Phi(y)\|^2 &= \langle \Phi(x), \Phi(x) \rangle - 2\langle \Phi(x), \Phi(y) \rangle + \langle \Phi(y), \Phi(y) \rangle \\ &= k(x, x) - 2k(x, y) + k(y, y)\end{aligned}$$

Shifting data: $x \mapsto x - t$, $\|x - y\|^2$ is invariant under the shift, but $\langle x, y \rangle$ is not!

There was a mistake here in the lecture!

$$\begin{aligned}\langle x - t, y - t \rangle &= \langle x, y \rangle - \langle x, t \rangle - \langle t, y \rangle + \langle t, t \rangle \\ &= \frac{1}{2} (\langle x, x \rangle - 2\langle x, t \rangle + \langle t, t \rangle + \langle y, y \rangle - 2\langle y, t \rangle + \langle t, t \rangle + 2\langle x, y \rangle - \langle x, x \rangle - \langle y, y \rangle) \\ &= \frac{1}{2} (\|x - t\|^2 + \|y - t\|^2 - \|x - y\|^2) \\ &\sim \tilde{k}(x, y)\end{aligned}$$

This $\tilde{k}(x, y) = \langle x - t, y - t \rangle$ is still psd.

$$\sum_{j,k} a_j a_k \langle x_j - t, x_k - t \rangle = \left\| \sum_j c_j (x_k - t) \right\|^2$$

So, for any choice of $t \in \mathbb{R}$ we get a similarity measure $\tilde{K}(x, y)$ associated with the dissimilarity measure $\|x - y\|$.
Now replace $\langle \cdot, \cdot \rangle$ by k .

Lemma 1.44. Let $t \in \Omega$ and K be a symmetric kernel on $\Omega \times \Omega$. Then $\tilde{k}(x, y) = \frac{1}{2}(k(x, y) - k(x, t) - k(t, y) + k(t, t))$ is positive semidefinite if and only if K is cond. psd. of order 1. If $k(t, t) \leq 0$, then

$$\hat{K}(x, y) := \frac{1}{2}(k(x, y) - k(x, t) - k(t, y))$$

is psd iff K is cpsd. of order 1.

Proof. 1. \tilde{k} psd $\implies K$ cpsd:

Let $a \in \mathbb{R}^n$ s.t. $\sum a_i = 0$ and let $(x_i)_{i=1}^N$, then

$$0 \leq \sum_{i,j=1}^N a_i a_j \tilde{k}(x_i, x_j) = \sum_{i,j} \sum_{i,j=1}^N a_i a_j k(x_i, x_j)$$

Since

$$\sum_j a_j \underbrace{\sum_i a_i k(x_i, t)}_{=c} = 0$$

2. K cpsd. $\implies \tilde{k}$ psd.:

Let $a \in \mathbb{R}^n$, $(x_i)_{i=1}^N$, let $x_0 = t$, $a_0 = -\sum_{i=1}^N a_i$. Then

$$0 \leq \sum_{i,j=1}^N a_i a_j k(x_i, x_j) = \sum_{i,j=1}^N a_i a_j \tilde{k}(x_i, x_j)$$

Thus \tilde{k} is psd. □

If we take negative square distance, here $\sum_i a_i = 0$ implies

$$-\sum_{i,j} a_j a_k \|x_j - x_i\|^2 = -\sum_j a_j \sum_k \|x_k\|^2 - \sum_k a_k \sum_j \|x_j\|^2 + \sum_j a_j \sum_k \langle x_j, x_k \rangle$$

so $-\|x - y\|^2$ is cpsd.

Therefore kernels that are cpsd. of order 1 are sometimes called negative definite kernels for $-K$. From Mercers theorem we have that \tilde{k} (psd) has a feature map $\phi : \Omega \rightarrow \mathcal{F}$, $\tilde{k}(x, y) = \langle \Phi(x), \Phi(y) \rangle$. Therefore $\|\Phi(x) - \Phi(y)\|^2 = \tilde{k}(x, x) + \tilde{k}(y, y) - 2\tilde{k}(x, y)$. For fixed $t \in \mathbb{R}$ we use the definition of \tilde{k} . $t \in \mathcal{F}$ prob. ???

$$\begin{aligned}\|\Phi(x) - \Phi(y)\|^2 &= \frac{1}{2}(k(x, x) - k(x, t) - k(t, x) + k(t, t) + k(y, y) - k(y, t) - k(t, y) + k(t, t) - 2(k(x, y) - k(x, t) - k(t, y) + k(t, t))) \\ &= -k(x, y) + \frac{1}{2}(k(x, x) + k(y, y))\end{aligned}$$

This shows

Theorem 1.45. *Let k be a cspd kernel of order 1 on Ω . Then there exists a Hilbert space \mathcal{F} and a mapping $\Phi : \Omega \rightarrow \mathcal{F}$ s.t.*

$$-k(x, y) + \frac{1}{2}(k(x, x) + k(y, y)) = \|\Phi(x) - \Phi(y)\|_{\mathcal{F}}^2$$

If $k(x, x) = 0$ for all $x \in \Omega$, we have

$$k(x, y) = -\|\Phi(x) - \Phi(y)\|_{\mathcal{F}}^2$$

and $\sqrt{-k(x, y)}$ is a semi-metric and a metric if $k(x, y) \neq 0$ for $x \neq y$.

Consider the

$$F[f](\omega) = \frac{1}{(2\pi)^{-d/2}} \int_{\mathbb{R}^d} f(x) \exp(-i\langle x, \omega \rangle) dx$$

and the [inverse Fourier transform](#)

$$F^{-1}[f](x) = \frac{1}{(2\pi)^{-d/2}} \int_{\mathbb{R}^d} f(\omega) \exp(i\langle x, \omega \rangle) d\omega$$

where

$$f(x) = F^{-1}[F[f]](x)$$

Consider S , where S^* is diagonalizable on the Fourier basis.

Denote $v(\omega)$ a nonnegative, symmetric function on \mathbb{R}^d :

$$v(\omega) = v(-\omega) \geq 0$$

further assume $v(\omega) \rightarrow 0$ for $\|\omega\| \rightarrow \infty$ and denote by Ω the support of v .

$$\langle Sf, Sg \rangle_D = (2\pi)^{\frac{d}{2}} \int_{\Omega} \frac{\overline{F[f](\omega)} F[g](\omega)}{v(\omega)} d\omega$$

Small values of $v(\omega)$ correspond to strong damping of the corresponding frequencies. This is desirable for large ω , i.e. high frequency components that correspond to rapid changes in f .

Now look at $g(x, z) = (2\pi)^{d/2} \int_{\Omega} e^{i\omega(x-z)} v(\omega) d\omega$. Let f have the support of its Fourier transform contained in Ω . We see

$$\begin{aligned}\langle Sg(x, \cdot), Sf \rangle &= (2\pi)^{d/2} \int_{\Omega} \frac{\overline{F[g(x, \cdot)](\omega)} F[f](\omega)}{v(\omega)} d\omega \\ &= (2\pi)^{-d/2} \int_{\Omega} \frac{\overline{v(\omega)} \exp(i\langle x, \omega \rangle) F[f](\omega)}{v(\omega)} d\omega \\ &= (2\pi)^{-d/2} \int_{\Omega} \exp(i\langle x, \omega \rangle) F[f](\omega) d\omega = f(x)\end{aligned}$$

According to theorem 1.40 g is the kernel corresponding to S .

This is a special case of

Theorem (Bocher's theorem). *For given $\kappa : \mathbb{R}^d \rightarrow \mathbb{C}$ the expression*

$$k(x, y) = \kappa(x - y)$$

defines a kernel if and only if there exists a unique finite Borel measure μ on \mathbb{R}^d s.t.

$$\kappa(x) = \int_{\mathbb{R}^d} \underbrace{\exp(i\langle x, y \rangle)}_{\cos(\langle x, y \rangle) \text{ for the real case}} d\mu(y)$$

Added remark. *random kitchen sink paper ... 20007*

Chapter 2:

Dimensionality reduction

Added remark. *Idea: High dimensional data $Y \rightarrow$ find low dimensional data X which represents Y to a good degree.*

For example: Represent vectors w.r.t. to a subset of the basis!

Set Y of $y_i \in \mathbb{R}^d$, $i = 1, \dots, N$ goal is to find

$$x_i \in \mathbb{R}^p, p \ll d$$

d is called the **extrinsic** dimension, while p is the **intrinsic** dimension.

We are now in the unsupervised setting

2.1 Linear dimensionality reduction

$\{y_i\}_{i=1}^N$ are samples of a random variable $Y(\omega) \in \mathbb{R}^d$. We assume Y stems from p unknown latent variables $X(\omega) \in \mathbb{R}^p$ by a linear transformation W .

$$Y = WX$$

which can be read as a statement about vectors (a single realization of the random variable) or in terms of matrices (the sampled realizations as a whole)-

We assume that Y, X are mean centered, i.e. $\mathbb{E}(Y) = 0$. We further assume W to be an axis change, i.e. the columns w_i of W are orthogonal to each other and unit norm:

$$W^\top W = I_p$$

We write the data in matrix form

$$Y = \begin{bmatrix} | & & | \\ y_1 & \dots & y_N \\ | & & | \end{bmatrix}$$

For any W , consider the pseudo inverse

$$W^\dagger = (W^\top W)^{-1} W^\top = W^\top$$

and therefore

$$X_i = W^\top y_i.$$

We aim for a good reconstruction of Y by X :

$$\begin{aligned} & \mathbb{E}(\|y - W(W^\top y)\|_2^2) \\ &= \mathbb{E}(y^\top y - 2y^\top W W^\top y + y^\top W W^\top W W^\top y) \\ &= \mathbb{E}(y^\top y - y^\top W W^\top y) \end{aligned}$$

He writes $Y \in \mathbb{R}^d$, which seems to be the wrong space for a random variable ...

with slightly abusive notation, as the matrix Y consists of N samples of the random variable Y

where

$$\begin{aligned}\mathbb{E}(Y^\top W W^\top \underbrace{Y}_{\in \Omega \times \mathbb{R}^d}) &\approx \frac{1}{N} \sum_{i=1}^N y_i^\top W W^\top y_i \\ &= \frac{1}{N} \text{tr}(Y^\top W W^\top \underbrace{Y}_{\in \mathbb{R}^{d \times N}}) \\ &= \frac{1}{N} \text{tr}(W^\top Y Y^\top W)\end{aligned}$$

Adding the constraint, we set the lagrangian

$$\mathcal{L} = \text{tr}(W^\top Y Y^\top W) + \text{tr}((I_p - W^\top W)\Lambda)$$

where $\Lambda = \Lambda^\top \in \mathbb{R}^{p \times p}$.

Conditions for an extrema:

$$(\star) : Y Y^\top W = W \Lambda \implies \Lambda = W^\top Y Y^\top W$$

The objective reduces to $\text{tr}(\Lambda)$. We can rotate W and have the same [reconstruction error](#), i.e. we can use $W' = WR$ giving $\Lambda' = R \Lambda R^\top$ for some rotation matrix R . $\implies \Lambda = \Lambda^\top$ is diagonalizable with orthogonal matrices, so we choose R s.t. Λ' is diagonal, w.l.o.g. Λ is diagonal. From (\star) follows that the columns of W must be eigenvectors of $Y Y^\top$ with corresponding eigenvalues as the diagonal of Λ . Since we maximize $\text{tr}(\Lambda)$, we take the p largest eigenvalues of $Y Y^\top$ and the corresponding eigenvectors.

SVD of Y : $U \Sigma V^\top$, we take the first p columns of U for W , i.e. $W = U I_{d \times p}$. Furthermore, as U is orthogonal,

$$X = W^\top Y = W^\top U \Sigma V^\top = I_{p \times d} \Sigma V^\top.$$

Using $\Lambda = W^\top U \Sigma^2 U W$, we get

$$\text{tr}(\Sigma^2) - \text{tr}(I_{p \times d} \Sigma^2) = \sum_{i=p+1}^d \sigma_i^2 = \sum_{i=p+1}^d \lambda_i$$

Theorem 2.1. Let $Y = [y_1, \dots, y_N] \in \mathbb{R}^{d \times N}$ be a matrix of zero mean data points. Denote the SVD of Y by $Y = U \Sigma V^\top$. Then for given $p < d$, the minimizer W for

$$\min_{\substack{W \\ W^\top W = I_p}} \sum_{i=1}^N \|y_i - W W^\top y_i\|_2^2$$

is given using $W = [u_1, \dots, u_p]$. The lower dimensional embedding is given by $X = I_{p \times d} \Sigma V^\top = I_{p \times d} U^\top Y$ and the [reconstruction error](#) is

$$\sum_{i=p+1}^d \sigma_i^2$$

To get compactness: Take [ball of functions](#): $\|f\| \leq r$ and use Arzelà-Ascoli.

Start of lecture 15
(11.06.24)

2.1.1 Alternative derivations of PCA

- Projection: We aim for $y = \sum_{i=1}^p x_i w_i$ with $y_i, w_i \in \mathbb{R}^d, w_i^\top, w_j = \delta_{ij}$. Then $x_i = \langle y_i, w_i \rangle$.
- Approximation with rank constraints:

$$\min_A \|Y - A\|_F^2$$

s.t. $\text{rank } A = p$

- From the statistical perspective:

Definition 2.2. Given a zero mean multivariate random variable $Y \in \Omega \times \mathbb{R}^d$. The *p* principle components of Y are defined as the p uncorrelated linear components of Y :

$$x_i = w_i^\top y \in \mathbb{R}, \quad w_i \in \mathbb{R}^d, \quad i = 1, \dots, p$$

s.t. the variance of x_i maximized subject to $w_i^\top w_i = 1$ and $\text{Var}(x_1) \geq \text{Var}(x_2) \geq \dots \geq \text{Var}(x_p)$

Theorem 2.3. Assume that the rank of the covariance matrix $\mathbb{E}(YY^\top)$ is larger than p . Then the first p principle components of a zero mean, multivariate random variable Y , denoted by $x_i, i = 1, \dots, p$ are given by

$$x_i = w_i^\top Y$$

where $\{w_i\}_{i=1}^p$ are the p orthonormal eigenvectors of $\mathbb{E}(YY^\top)$ associated its p largest eigenvalues $\{\lambda_i\}_{i=1}^p$. Moreover $\lambda_i = \text{Var}(X_i)$.

In the exercise we show that the greedy approach works in this specific setting?

Added remark. Y , which has N columns and d rows ...

Added remark. In general we might want to work with tensors (for example if we have a time dependent structure). This is possible, but can be generalized in multiple, non-equivalent, ways. The sum of vectors representation is not so nice, there are counter examples ...

We aim for embeddings that approximately preserve distances.

$$d^d(y_1, y_2) \approx d^p(x_1, x_2)$$

Here, we always talk about euclidean distances!

Definition 2.4. A $N \times N$ symmetric matrix D is called Euclidean distance matrix (EDM), if there exists an integer $d > 0$ and a vector set $Y = \{y_1, \dots, y_N\}, y_i \in \mathbb{R}^d$, s.t. $D_{i,j} = d_E^2(y_i, y_j)$, where d_E is the euclidean distance. The vector set Y is called the configuration of D . We write $D \in \text{EDM}$.

Careful! Some authors use the euclidean distance matrix with a square, and some without the square!

Definition 2.5. A $N \times N$ symmetric matrix D with non-negative entries $d_{i,j}$ is called distance matrix, if $d_{ii} = 0$ for all i and

$$\sqrt{d_{ij}} \leq \sqrt{d_{ik}} + \sqrt{d_{kj}}$$

for all i, j, k . We write $D \in \text{DM}$.

Obviously $\text{EDM} \subset \text{DM}$.

$$(\star) \quad d_E^2(y_i, y_j) = \underbrace{\langle y_i, y_i \rangle}_{G_{ii}} - 2 \underbrace{\langle y_i, y_j \rangle}_{G_{ij}} + \underbrace{\langle y_j, y_j \rangle}_{G_{jj}}$$

where $G_{ij} = \langle y_i, y_j \rangle = (Y^\top Y)_{ij}$.

As for PCA, we aim for mean centered data, where we use the centering matrix

$$H = I - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^\top,$$

where $\mathbf{1}_W = \mathbf{1}_N \cdot \mathbf{1}_N^\top$ is the matrix of all ones, where $\mathbf{1}_N$ is the vector of all ones. with

$$Y^c = Y - \left(\frac{1}{N} Y \mathbf{1}_N\right) \mathbf{1}_N^\top = YH.$$

We set the centered data. The centered Gram matrix

$$G^c = (Y^c)^\top Y^c = H^\top Y^\top Y H = H^\top G H$$

Theorem 2.6. For the Euclidean distance matrix D and the centered Gram matrix G^c of a data set Y , it holds

$$G^c = -\frac{1}{2} H D H$$

Proof. Straight forward calculation from (\star) . □

Lemma 2.7. Assume that the matrix $D \in \text{EDM}$ and let $G^c = -\frac{1}{2}HDH$. If the rank of G^c is r , then there is a centered r dimensional configuration $Y = \{y_1, \dots, y_N\} \in \mathbb{R}^r$, s.t. $d_E(y_i, y_j) = d_{ij}$.

Proof. Since $D \in \text{EDM}$, there exists a set $Z = \{z_1, \dots, z_N\} \in \mathbb{R}^d$, s.t. $d_{ij} = d_E(z_i, z_j)$, G^c is the Gram matrix of Z and therefore psd. The rank is r , so we have $G^c = Y^\top Y$ with a centered data matrix (via EDM). The centered data satisfies $d_{i,j} = d_E(y_i, y_j)$. We call r the **intrinsic configuration dimension** and Y is the **exact configuration** of D . \square

2.1.2 (classical) multidimensional scaling (MDS)

Instead of exact configuration of dimension r , we seek a lower dimensional configuration $Y \subset \mathbb{R}^p$, $p < r$;

$$X = \underset{X \subset \mathbb{R}^{p \times N}}{\operatorname{argmin}} \sum_{i,j}^N |d_{ij}^2 - d_E^2(x_i, x_j)| \quad (\star\star)$$

s.t. $X = T(Y)$, with T an orthogonal projection from \mathbb{R}^r to a p -dimensional space $S_p \subset \mathbb{R}^r$ and Y is an exact configuration.

Lemma 2.8. Let $Z \subset \mathbb{R}^r$ be a given data set with $D_Z = [d_E^2(z_i, z_j)]_{i,j=1}^N$ and let $G_Z^c = -\frac{1}{2}HD_ZH$. Then

$$\operatorname{tr}(G_Z^2) = \frac{1}{2N} \sum_{i,j=1}^N d_E^2(z_i, z_j)$$

Proof. Write out G_Z^2 , look at diagonal, straight forward calculation. \square

Lemma 2.9. Let D_Z as before. $ZH =: \tilde{Z} = [\tilde{z}_1, \dots, \tilde{z}_N]$. Then $\|\hat{Z}\|_F = \frac{1}{\sqrt{2N}}\|D_Z\|_F$.

Proof. With $\|D_Z\|_F^2 = \sum d_E^2(z_i, z_j)$ and $\|\hat{Z}\|_F^2 = \operatorname{tr}(\hat{Z}^\top \hat{Z}) = \operatorname{tr}(\hat{G}_Z^2)$. The result follows from lemma 2.8. \square

Theorem 2.10. Let $Y \subset \mathbb{R}^r$ be an exact configuration of $D \in \text{EDM}$. The SVD of Y is given by $U\Sigma V^\top$. For a given $p \leq r$, let $U_p = [u_1, \dots, u_p]$. Then

$$X = U_p^\top Y$$

is a solution of the MDS minimization problem $(\star\star)$ with an error of

$$2N \sum_{i=p+1}^r \sigma_i^2.$$

Proof. Let S_p be a p -dim subspace of \mathbb{R}^r . Let B be a $r \times p$ orthogonal matrix, whose columns form an orthonormal basis of S_p . We have $T(y) = BB^\top y$ and observe

$$d_E(B^\top y_i, B^\top y_j) = \|B^\top(y_i - y_j)\| = \|T(y_i - y_j)\| = d_E(Ty_i, Ty_j).$$

With $\|y_i - y_j\| \geq \|T(y_i - y_j)\|$. We get for the objective function

$$\begin{aligned} \sum_{i,j=1}^N d_E^2(y_i, y_j) - d_E^2(B^\top y_i, B^\top y_j) &= \sum_{i,j=1}^N \langle y_i - y_j, y_i - y_j \rangle - \langle B^\top(y_i - y_j), B^\top(y_i - y_j) \rangle \\ &= \langle B^\top y, B^\top y \rangle = \langle B^\top B y, B^\top y \rangle \dots = 2 \langle BB^\top(y_i - y_j), (y_i - y_j) \rangle + \langle BB^\top(y_i - y_j), (y_i - y_j) \rangle \\ &= \sum_{i,j=1}^N |(I - BB^\top)(y_i - y_j)|^2 = \|D_Z\|_F^2 \end{aligned}$$

with $Z = (I - BB^\top)Y$. Using Lemma 2.9, we get $\|D_Z\|_F^2 = 2N\|ZH\|_F^2 = 2N\|Z\|_F^2$, since Y is centered Z is also centered. Therefore we have to solve

$$\underset{\substack{B \in \mathbb{R}^{r \times p} \\ B^\top B = I_p}}{\operatorname{argmin}} \|Y - BB^\top Y\|_F.$$

Again we use Schmidt-Eckart-Yanns for the SVD. The matrix $U_p \Sigma_p V_p^\top$??? So getting $B = U_p$ gives

$$U_p U_p^\top U \Sigma V^\top = U_p \Sigma_p V_p^\top$$

and $X = U_p^\top Y$. The error estimate follows from ??? SVD and Lemma 2.9. \square

It is important to understand the flip: You can choose the minimum of the number of dimensions and the number of data points.

Recap:

$$G^c = Y^\top Y = (W^\top X | W X) = X^\top W^\top W X = X^\top X$$

EVD of G^c :

$$\begin{aligned} G^c &= V \Lambda V^\top = (V \Lambda^{\frac{1}{2}})(\Lambda^{\frac{1}{2}} V^\top) \\ &= (\Lambda^{\frac{1}{2}} V^\top)^\top (\Lambda^{\frac{1}{2}} V^\top) \end{aligned}$$

Taking the top p eigenvalues gives

$$X_{\text{MDS}} = I_{p \times N} \Lambda^{\frac{1}{2}} V^\top$$

Also $Y = U \Sigma V^\top$ for PCA:

$$\begin{aligned} X_{\text{PCA}} &= I_{p \times d} U^\top Y = I_{p \times d} \Sigma V^\top \\ &= I_{p \times d} (\Sigma^\top \Sigma)^{\frac{1}{2}} V^\top = I_{p \times d} \Lambda^{\frac{1}{2}} V^\top \end{aligned}$$

We can use the following approaches:

- SVD of Y : ($d \times N$): Reconstruct Y
- EVD of $Y Y^\top$ ($d \times d$): maximal variance
- EVD of $Y^\top Y$ ($N \times N$): preserving similarity

Start of lecture 16
(13.06.24)
EVD= Eigen value
decomposition

The SVD is most
commonly used and ist
most stable. If you have
extreme differences in d
and N , it might be worth
to use on of the other two
approaches

Algorithm 1 MDS

Input: EDM D, P

Output: embedding X in p dimensions

$$G = -\frac{1}{2} H D H$$

$$[V_p, \Lambda_p] = \text{EVD}(G, p)$$

$$\text{Return } \Lambda_p^{\frac{1}{2}} V^\top$$

Added remark. *There are multiple generalizations!*

2.1.3 Strange effects in high dimensions

Angles

First we look at angles between vectors. In 2d the angle between the diagonal and an axis is $\frac{\pi}{4}$.
In general:

$$\varphi = \arccos \frac{\langle x, y \rangle}{\|x\| \|y\|}$$

Then $\alpha = \frac{1}{\sqrt{d}}$.

Volume

The volume of the hypersphere in d dimensions:

$$\frac{r^d \pi^{\frac{d}{2}}}{\Gamma(1 + \frac{d}{2})}$$

The Volume of the hypercube is $(2r)^d$.

Therefore in higher dimensions the sizes of a unit hypersphere and its bounding hypercube are not close, their ratio goes to 0!

Concentration of measure

If we increase the radius of the sphere by just some small ϵ , we can see that most of the mass lies in the outer shell of hyperspheres!

$$\frac{V_r}{V_{r(1-\epsilon)}} = \frac{1}{(1-\epsilon)^d}$$

Curse of dimensionality

The higher the dimension, the more points I need to approximate functions of the same complexity as the dimension increases. This scaling is exponential! Therefore we need lots of data to recover functions in higher dimensions.

If we do datascience in higher dimensions, we always implicitly assume that there is more structure, otherwise most problems would not be feasible

ANOVA decomposition

However the dimensions might be highly correlated! We can then use the ANOVA decomposition:

$$f(x) = \sum_{\tilde{d}_i=1, \dots, D} f_{\tilde{d}_i}(x_{\tilde{1}}, \dots, x_{\tilde{p}})$$

2.1.4 Properties in dimensionality reduction approaches

1. Estimate the intrinsic dimensionality
2. What kind of properties of the original data does one (implicitly or explicitly) assume to hold and to approximately preserve
3. One often aims for latent variable separation (statistical independence, orthogonality)¹

For PCA/MDS, we can consider the rank of the matrix as the intrinsic dimensionality. We can use criteria such as

$$\frac{\sum_{i=1}^p \lambda_i}{\sum_{i=1}^d \lambda_i} \geq 0.95$$

or

$$\lambda_{p+1} \leq 0.01 \sum_{i=1}^d \lambda_i$$

This assumes no noise, otherwise we can be reasonably sure that the intrinsic dimensionality is smaller than the rank

Another procedure: The L -curve: Linearly approximate the eigenvalues as a function of the index from both the left and the right and take the intersection as the cutoff point. In the latter approaches we assume a fast decline in eigenvalues.

2.2 nonlinear dimensionality reduction

We use manifolds instead of linear subspaces. Ideally we use geodesic distances $d_M(x, y)$ and preserve d_M in the embedding. But both M and d_M are unknown, only $D \subset M$ is given.

Approach: Build an undirected neighborhood graph $[Y, E]$ and use the graph distances d_G instead of d_M and aim to preserve it.

Definition 2.11. Given a graph $[Y, E]$ for a data set $Y \subset \mathbb{R}^d$, s.t. $(y_i, y_j) \in E$ if and only if y_i, y_j are adjacent^a. We define the **graph distance d_G** between two points $y_i, y_j \in Y$ by

1. If $(y_i, y_j) \in E$ then $d_G(y_i, y_j) = d_E(y_i, y_j)$
2. If $(y_i, y_j) \notin E$ then let $\Gamma\{\gamma \mid \gamma = (\gamma_0, \dots, \gamma_s), \gamma_i \in Y, \gamma_0 = y_i, \gamma_s = y_j\}$ and define

$$d_G(y_i, y_j) = \min_{\gamma \in \Gamma} \sum_{i=0, \dots, s-1} d_E(\gamma_i, \gamma_{i+1})$$

^ain some geometric senses

¹this is also useful for interpretability!

We assume $Y \subset M \subset \mathbb{R}^d$ and that an isometric mapping

$$f : M \rightarrow \mathbb{R}^p$$

exists $f(y) = x$ for $y \in M$ and

$$d_E(f(y_i), f(y_j)) = d_M(y_i, y_j)$$

for all $y_i, y_j \in M$.

Assume Y is sampled densely enough from M , we expect that

$$d_G(y_i, y_j) \approx d_M(y_i, y_j).$$

$D_G = [d_G^2(y_i, y_j)]_{i,j=1}^N$, we aim for $D \in \text{EDM}$, with a configuration $X \in \mathbb{R}^p$ s.t.
 $D_G \approx D = [d_E^2(x_i, x_j)]_{i,j=1}^N$.

Algorithm 2 Isomap

Input: Dataset Y , p

Output: data set embedding X in p dimensions

Build a neighborhood graph $[Y, E]$

$D_{ij} = [d_E^2(x_i, x_j)]$ using Dijkstra's algorithm

$G = -\frac{1}{2}H D H$

$[V_p, \Lambda_p] = \text{EVD}(G, p)$

Return $\Lambda_p^{\frac{1}{2}} V^T$

Assume $D_G \in \text{EDM}$ we can invoke the MDS theorem 2.10. We get

$$\sum_{i,j=1}^N |d_G^2(y_i, y_j) - d_E^2(x_i, x_j)| \leq 2N \sum_{l=p+1}^N \lambda_l.$$

For the analysis we will use r -ball graphs:

$$(y_i, y_j) \in E \iff \|y_i - y_j\|_E \leq r.$$

Furthermore we use the Hausdorff distance between Y and M

$$\epsilon = H(M | Y) = \sup_{y \in M} \min_{y_i \in Y} \|y - y_i\|$$

Theorem 2.12. Consider $M \subset \mathbb{R}^d$ compact and a sample $Y = \{y_1, \dots, y_N\} \subset M$ and let $\epsilon = H(M | Y)$. For $r > 0$ form the corresponding r -ball graph. When $\epsilon \leq \frac{r}{4}$, we have for any $x, z \in Y$

$$d_G(x, z) \leq \left(1 + \frac{4\epsilon}{r}\right) d_M(x, z)$$

In practice we often choose k -nearest neighbors instead of r -ball neighborhoods!

Start of lecture 17
(18.06.24)

Added remark. CMDS is the same as MDS in our lecture. This is relevant for the sheets.

Proof. For $d_E(x, z) \leq r$ we have that $(x, z) \in E$ in the r -ball graph and so

$$d_G(x, z) = d_E(x, z) \leq d_M(x, z).$$

Now consider $d_E(x, z) > r$ and let $a = d_M(x, z)$ and let $\gamma : [0, a] \rightarrow M$ be parametrized by arc length s.t. $\gamma(0) = x = y_{i_0}$ and $\gamma(a) = z = y_{i_s}$. Let $\hat{y}_{i_j} = \gamma\left(\frac{ja}{s}\right)$ for $j = 0, \dots, s$ where $s = \lceil 2a/r \rceil \geq 2$, where $\hat{y}_0 = a, \hat{y}_s = z$. Let $y_j = \arg\min_{y \in Y} d_E(y, \hat{y}_j)$. Clearly $\max_s d_E(y_{i_j}, \hat{y}_j) \leq \epsilon$ for any $j = 0, \dots, s-1$.

$$\begin{aligned} d_E(y_{i_j}, y_{i_{j+1}}) &\leq d_E(y_{i_j}, \hat{y}_j) + d_E(\hat{y}_j, \hat{y}_{j+1}) + d_E(\hat{y}_{j+1}, y_{i_{j+1}}) \\ &\leq \epsilon + d_M(\hat{y}_j, \hat{y}_{j+1}) + \epsilon \\ &= a/s + 2\epsilon \leq r/2 + 2\epsilon < r \end{aligned}$$

Therefore $(y_{i_0}, \dots, y_{i_s})$ forms a path in the r -ball graph.

This might not be the shortest path in the graph

$$\begin{aligned}
 d_G(x, z) &\leq \sum_{i=j}^{s-1} d_E(y_j, y_{j+1}) \\
 &\leq d_M(\underbrace{\hat{y}_0}_{=x}, \hat{y}_1) + \epsilon + \sum_{j=1}^{s-2} d_M(\hat{y}_j, \hat{y}_{j+1}) + 2\epsilon + d_M(\hat{y}_{s-1}, \underbrace{\hat{y}_s}_{=z}) + \epsilon \\
 &= d_M(x, z) + 2(s-1)\epsilon \\
 &\leq \left(1 + \frac{4\epsilon}{r}\right) d_M(x, z)
 \end{aligned}$$

we use that $s-1 \leq \frac{2a}{r}$ with $a = d_M(x, z)$. \square

Added remark. Here we use that the subset of a shortest path is still a shortest path for the specific start and end points.

Lemma 2.13. Let $\gamma : [0, a] \rightarrow \mathbb{R}^d$ be a unit speed curve with curvature bounded by κ . Then $d_E(\gamma(s), \gamma(t)) \geq \frac{2}{\kappa} \sin\left(\frac{\kappa|t-s|}{2}\right)$

Proof of a weaker statement. Let c denote the unit-speed parametrization of a circle of radius $\frac{1}{\kappa}$. From Darbins (1957) we obtain $|t-s| \leq \frac{\pi}{\kappa}$:

$$\langle \gamma(s), \gamma(t) \rangle \geq \langle c(s), c(a) \rangle$$

This leads to

$$\begin{aligned}
 \|\gamma(t) - \gamma(s)\| \|\dot{\gamma}(s)\| &\stackrel{\text{C.S.}}{\geq} \langle \dot{\gamma}(s), \gamma(t) - \gamma(s) \rangle \\
 &= \int_s^t \langle \dot{\gamma}(s), \gamma(u) \rangle du \\
 &\geq \int_s^t \langle c(s), c(u) \rangle du \\
 &= \langle c(s), c(t) - c(s) \rangle \\
 &= \frac{1}{\kappa} \sin(\kappa \cdot (|t-s|))
 \end{aligned}$$

when $0 \leq t-s \leq \frac{\pi}{\kappa}$.

We now assume $M \subset \mathbb{R}^d$ is compact and connected C^2 -manifold with empty or C^2 -boundary. In particular, shortest path on M have curvature bounded by some κ . \square

Lemma 2.14. Suppose $M \subset \mathbb{R}^d$ has the above properties. Then for any $x, z \in M$ s.t. $d_M(x, z) \leq \frac{\pi}{\kappa}$

1. $d_M(x, z) \max\left(\frac{2}{\pi}, 1 - \frac{\kappa^2}{24} d_M(x, z)^2\right) \leq d_E(x, z) \leq d_M(x, z)$. Moreover there is $\tau > 0$ depending on M s.t. for all $x, z \in M$ with $d_E(x, z) \leq \tau$ it holds
2. $d_E(x, z) \leq d_M(x, z) \leq d_E(x, z) \min\left(\frac{\pi}{2} + C_0 \kappa^2 d_E(x, z)^2\right)$ where c_0 is a constant that can be taken to be $c_0 = \frac{\pi^2}{50}$.

Proof. For 1.:

Take $x, z \in M$ and let $a = d_M(x, z) \leq \frac{\pi}{\kappa}$. Let $\gamma : [0, a] \rightarrow M$ be a unit-speed shortest path on M s.t. $\gamma(0) = x, \gamma(a) = z$. By assumption γ has curvature bounded by κ . Applying lemma 2.13 gives

$$d_E(x, z) = d_E(\gamma(0), \gamma(a)) \geq \frac{2}{\kappa} \sin\left(\kappa \frac{a}{2}\right) \geq \max\left(a - \frac{\kappa^2}{24} a^3, \frac{2}{\pi} a\right)$$

and $\sin(t) \geq t - \frac{t^3}{6}, \sin(t) \geq \frac{2t}{\pi} \forall t \leq \frac{\pi}{2}$.

For 2.: by 1. $d_M(x, z) \leq \frac{\pi}{\kappa}$, then $d_M(x, z) \leq \frac{\pi}{2} d_E(x, z)$ and

$$\begin{aligned} d_M(x, z) &\leq \frac{d_E(x, z)}{\left(1 - \frac{\kappa^2}{24} d_M(x, z)^2\right)} \\ &\leq \frac{d_E(x, z)}{\left(1 - \frac{\kappa^2}{24} \left(\frac{\pi}{2} d_E(x, z)\right)^2\right)} \\ &\leq d_E(x, z) (1 + C_0 \kappa^2 d_E(x, z)^2) \end{aligned}$$

where the last inequality holds if $\kappa d_E(x, z)$ is small enough. Therefore, it is enough to show that there is $\tau > 0$ s.t.

$$d_M(x, z) \leq \frac{\pi}{\kappa}$$

when $d_E(x, z) \leq \tau$. This is true, because the smoothness of M guarantees that d_M be continuous as a function of the compact set $M \times M$. \square

Theorem 2.15. Suppose $M \subset \mathbb{R}^d$ has the above properties. Consider a sample $Y = \{y_1, \dots, y_m\} \subset M$. For $r > 0$ from the corresponding r -ball graph. Let c_0 and τ be defined per lemma 2.14 and $\kappa r \leq \frac{1}{3}$. We have

$$d_M(x, z) \leq (1 + C_0 \kappa^2 r^2) d_G(x, z) \quad \forall x, y \in Y$$

Proof. Fix $x, z \in Y$. Let $x = y_{i_0}, \dots, z = y_{i_s}$ define a shortest path in the graph joining x, z , so that $d_G(x, z) = \sum_{j=0}^{s-1} \Delta_j$, where $\Delta_j = d_E(y_{i_j}, y_{i_{j+1}})$. Define $a = d_M(x, z)$ and $a_j = d_M(y_{i_j}, y_{i_{j+1}})$. Since $\Delta_j \leq r \leq \tau$ by lemma 2.14 we see

$$\Delta_j \min\left(\frac{\pi}{2} c_0 \kappa^2 \Delta_j^2\right) \geq a_j.$$

By assumption, $\kappa r \leq \frac{1}{3}$, and this can be seen to imply $1 + C_0 \kappa^2 r^2 \leq \frac{\pi}{2}$, which then implies that $a_j \leq \Delta_j + C_0 \kappa^2 \Delta_j^3$, we thus have

$$a \leq \sum_{j=1}^{s-1} a_j \leq \sum_{j=0}^{s-1} (\Delta_j + C_0 \kappa^2 \Delta_j^3) \leq \sum_{j=0}^{s-1} \Delta_j (1 + C_0 \kappa^2 r^2) = (1 + C_0 \kappa^2 r^2) d_G(x, z) \quad \square$$

Added remark. This is the justification for using Isomap. One should consider, that the assumptions are NOT always met in practice, which might result our algorithm to fail. In particular we don't know M in practice, which makes our assumptions hard to check ... Luckily the assumptions are often met “naturally”.

2.2.1 Parallel transport unfolding

Underlying idea: Systematically overestimating the distance (by doing unnecessary turns on the graph). We then think about the tangent spaces and try to connect the local views of the tangent spaces. Main idea we find a better path (closer to the geodesic).

We will now consider the computational ingredients using discrete parallel transport. We are using a l -nearest neighbor graph.

1. Approximate tangent spaces: Take q -NN in the graph, $q \geq k$, denoted

$$\{\tilde{y}_l^i\}_{l=1}^q, \quad \hat{y}_l^i = \tilde{y}_l^i - y_i \quad \hat{Y}^i = [\hat{y}_1^i, \dots, \hat{y}_q^i]$$

The p left singular vectors of \hat{Y}^i to the p largest singular values give an orthonormal basis for T_i , spanning the approximate tangent space

$$T_i = [t_1^i, \dots, t_p^i] \in \mathbb{R}^{d \times p}$$

2. Now given y_o and y_j connected in G

$$R_{j,i} = \operatorname{argmin}_{R \in O(p)} \|T_i - T_j R\|_F$$

$R_{j,i}$ is the discrete metric connection between y_i, y_j , it best aligns T_i, T_j . By definition $R_{i,j} = R_{j,i}^{-1} = R_{j,i}^T$. With $T_i^T T_j = U \Sigma V^T$ we can show $R_{j,i} = V U^T$.

Start of lecture 18
(20.06.24)

separate k, q to avoid unwanted connection while still getting good approximations
The space spanned by the columns of T_i

$O(p)$ are the $p \times p$ orthogonal matrices

3. Consider (y_i, y_j, y_k) and start w.l.o.g. y_i . The first edge

$$l_i = y_j - y_i$$

is projected onto T_i : $v_i = T_i^\top e_i$. We set $z_i = 0$ and $z_j = z_i + v_i$ defines the first modified edge. Now, $e_j = y_k - y_i$ and $T_j^\top e_j$ is parallel transported into the tangent space at y_i

Projected in the L^2 sense

$$v_j = R_{i,j}[T_j^\top e_j]$$

and $z_k = z_j + v_j$ Under some assumptions

$$\|v_i + v_j\|_2 = d_E(z_i, z_k) \approx d_M(y_i, y_k)$$

4. For longer paths y_{i_1}, \dots, y_{i_m} we iteratively project the edges $l_{i_s} = y_{i_{s+1}} - y_{i_s}$ onto T_{i_j} before parallel transporting back to T_{i_1}

$$V_{i_s} = \left(\prod_{j=1}^{s-1} R_{i_j, i_{j+1}} \right) [T_{i_s}^\top l_{i_s}]$$

with $V = \sum_{j=1}^M v_{I_S}$ WE GET $\|v\|_2 \approx d_M(y_1, y_{i_m})$

$$(\star) \quad 1 - \xi \frac{d_G(y_i, y_j)}{d_M(y_i, y_j)} \leq 1 + \xi$$

We saw in PTU

$$\min_{R \in O(p)} \|Y^\top - X^\top R\|_F.$$

Generally aligning two point sets using an orthogonal transformation is called **procrustes problem**.

Pointwise:

$$\min_{R \in O(p)} \sum_{i=1}^N \|y_i - Rx_i\|_2^2.$$

Consider two configurations X, Y , given

$$\|Y^\top Y - X^\top X\|_F = \epsilon^2$$

and

$$\|X^\dagger\|_2 \epsilon \leq \frac{1}{\sqrt{2}}$$

it holds

$$\min_{R \in O(d)} \|Y^\top X^\top R\|_F \leq (1 + \sqrt{2}) \|X^\dagger\|_2 \epsilon^2$$

For $D, \tilde{D} \in \text{EDM}$ it holds using $\epsilon^2 = \frac{1}{2} \|H(\tilde{D} - D)H\|_F$ a corresponding result. Using (\star) for Isomap and X the exact embedding using d_M into \mathbb{R}^p one has

$$\min_{R \in O(p)} \left(\frac{1}{N} \sum_{i=1}^N \|z_i - Rx_i\|^2 \right)^{\frac{1}{2}} \leq \frac{36\sqrt{p}\rho^3}{w^2} \xi$$

for Z the approximate embedding and ρ, w are singular values of our datamatrix:

$$\rho = \rho(X) = \frac{\sigma_1(X)}{\sqrt{N}}$$

radius of X largest deviations along any direction in space.

$$w = w(X) = \frac{\sigma_p(w)}{\sqrt{N}}$$

half width of X , smallest deviations along any direction in space.

Using kernels again ...

Nonlinear PCA

Reminder: $C = \frac{1}{N} Y \cdot Y^\top = \frac{1}{N} \sum_{i=1}^N y_i \cdot y_i^\top$ we consider some **feature map** $\phi : \mathbb{R}^d \rightarrow \mathbb{R}^r \rightarrow \mathcal{F}$ For simplicity, we assume to be mean-centered on the feature space:

$$\sum_{i=1}^N \phi(y_i) = 0.$$

Now, do PCA in \mathcal{F} , $C_\phi = \frac{1}{N} \sum_{i=1}^N \phi(y_i) \phi(y_i)^\top = \frac{1}{N} \phi(Y) \phi(Y)^\top = \frac{1}{N} \Phi \cdot \Phi^\top$.

EVD gives $C_\phi u_i = \lambda_i u_i$ and by theorem 2.3 the nonlinear principal components are

$$X_i = u_i^\top \Phi.$$

IGNORE:

$$\Phi \Phi^\top \rightarrow \Phi^\top \Phi$$

$$U^\top = V^\top \Phi^\top$$

$$X = V^\top \Phi^\top \Phi = \Lambda^{-\frac{1}{2}} \Phi^\top \Phi = \Lambda^{-\frac{1}{2}} V^\top V \Lambda V^\top = \Lambda^{\frac{1}{2}} V^\top$$

STOP IGNORING

$$x_I = V^\top \Phi^\top \Phi(y_i)$$

$$K_{i,j} = \langle \Phi(y_i), \Phi(y_j) \rangle_{\mathcal{F}} =: K(y_i, y_j)$$

A projection of a point y with image $\phi(y)$ gives

$$\begin{aligned} x &= V^\top \Phi^\top \phi(y) \\ &= \sum_{i=1}^N v_i \langle \phi(y_i), \phi(y) \rangle_{\mathcal{F}} \\ &= \sum_{i=1}^N v_i K(y_i, y) \end{aligned}$$

This kernelized form we call **Kernel MDS**, but it is usually called **Kernel PCA** in the literature.

To consider non-centered data, we need to center it:

$$\tilde{\phi}(y_i) := \phi(y_i) - \frac{1}{N} \sum_{j=1}^N \phi(y_j)$$

with associated kernel \tilde{K} .

$$\tilde{\Phi}(Y) = (\Phi^c(Y) - \frac{1}{N} \Phi(Y) \mathbf{1}) \mathbf{1}^\top = \Phi(Y) H$$

Action of H goes through the SP

$$\tilde{K} = K^c = H K H$$

Algorithm 3 Nonlinear PCA

Input: nonlinear data set $Y \phi : \mathbb{R}^d \rightarrow \mathbb{R}^r, p$

Output: configuration X in $p \leq \min(r, N)$ dimensions

$$K = -H \Phi^\top(Y) \Phi H$$

$$[V_p, \Lambda_p] = \text{EVD}(K, p)$$

$$\text{Return } \Lambda_p^{\frac{1}{2}} V_p^\top$$

The interpretation step of PCA gets a lot harder. Especially, since the feature maps are not unique ...

Algorithm 4 Nonlinear MDS**Input:** nonlinear data set Y , Kernel $K : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ **Output:** configuration X in $p \leq N$ dimensions

$$K_{i,j} = K(y_i, y_j)$$

$$\hat{K} = H K H$$

$$[V_p, \Lambda_p] = \text{EVD}(\hat{K}, p)$$

$$\text{Return } \Lambda_p^{\frac{1}{2}} V_p^\top$$

Connection of PCA and MDS (motivated by the kernelized versions):

$$X_{\text{PCA}} = I_{p \times d} U^\top Y = I_{p \times N} \Lambda^{\frac{1}{2}} V^\top$$

Start of lecture 19
(25.06.24)**2.2.2 A view on Kernel MDS embedding**

$$K = V \Lambda V^\top \iff V^\top K V = \Lambda$$

$$\begin{aligned} \lambda_i \delta_{i,j} &= V_i^\top L V_j = \sum_{l_1, l_2}^N V_{l_1, i} k(y_{l_1}, y_{l_2}) V_{l_2, j} \\ &= \left\langle \sum_{l_1=1}^N V_{l_1, i} k(y_{l_1}, \cdot), \sum_{l_2=1}^N V_{l_2, j} k(y_{l_2}, \cdot) \right\rangle \end{aligned}$$

This tells us that the

$$f_i(\cdot) = \frac{1}{\sqrt{\lambda_i}} \sum_{l=1}^N V_{l, i} k(y_l, \cdot)$$

is a orthonormal basis of \mathcal{H}_Y if $\text{rank} K = N$.The first p $f_i(y)$ give an embedding into \mathbb{R}^p of y .

Remark. Let $\tilde{K} = H K H$. It can be seen that \tilde{K} is psd iff K is cpsd. The centering removes the dependence on the origin in the feature space. Reminder: $G^c = -\frac{1}{2} H D H$, for comparison. We also considered negative distance as a cpsd kernel.

Remark. Just using any kernel can actually increase the embedding dimension. Take the Gaussian kernel, it will give a matrix with full rank N . Looking at the decay of EV of the kernel matrix shows a weak decay!

2.2.3 Maximum Variance Unfolding**Added remark.** he uses $|||$ for matrix norms, while I still use $||$!

1. Euclidean embedding space Gram matrix K is to be computed, so a constraint is $K \in \text{SPSD}$, i.e. the symmetric positive semi-definite matrices.
2. We use Kernel MDS and aim for a **centered** K .
3. We want to **locally preserve distances**: Locally euclidean distances are ok

$$d_E(y_i, y_j) = d_E(\phi(y_i), \phi(y_j))$$

4. We **maximize the variance** by maximizing the pairwise distance in the feature space for y_i, y_j with $(y_i, y_j) \notin E$:

$$\frac{1}{2N} \sum_{i,j=1}^N d_E^2(\phi(y_i), \phi(y_j)) \stackrel{\text{Lemma 2.8}}{=} \text{tr}(G_\phi^c)$$

Definition 2.16. Given a data set Y and a neighborhood graph $[Y, E]$. The solution of

$$\max_{K \in \text{SPSD}} \text{tr}(K) \text{ s.t. } \sum_{i,j}^N K_{ij} = 0 \quad (1)$$

and with $(y_i, y_j) \in E$: $d_E^2(y_i, y_j) = K_{ii} - 2K_{ij} + K_{jj}$ as a second condition, where $\sum_{i,j}^N K_{ij} = 0$ is the centering condition

Algorithm 5 MVU

Input: data set Y , its pairwise distances, p

Output: Embedding in their p dimensions

Build their neighborhood graph $[V, E]$

Solve (1) to get K

$\hat{K} = HKH$

$[V_p, \Lambda_p] = \text{EVD}(K, p)$

Return $\Lambda_p^{\frac{1}{2}} V_p^T$

We can write the MVU problem also as

$$\max_{D \in \text{EDM}} \sum_{i,j=1}^N D_{ij} \text{ s.t. } D_{ij} = d_E^2(y_i, y_j) \text{ for } (y_i, y_j) \in E \quad (2)$$

Theorem 2.17. Let $C \subset \text{DM}$ and $[Y, E]$ a data graph with weights d_W . If the graph is connected, the following two constraint optimization problems are equivalent:

1. (2)

$$\max_{D \in \text{EDM}} \sum_{i,j=1}^N D_{ij} \text{ s.t. } D_{ij} = d_E^2(y_i, y_j) \text{ for } (y_i, y_j) \in E$$

2.

$$\min_{D \in C} \|D - D^G\|_1 \min_{D \in C} \sum_{i,j=1}^N |D_{ij} - D_{ij}^G| \quad (3)$$

s.t. $D_{ij} = d_W(y_i, y_j)$ if $(y_i, y_j) \in E$ where $D_{ij}^G = d_G^2(y_i, y_j)$ is the (squared) graph distance matrix for the edge weights d_W .

Proof. Let $D \in C$. Then for all $1 \leq i, j \leq N$ and paths $\gamma \in [Y, E]$ connecting $y_i = \gamma_0, y_j = \gamma_{s+1}$ the triangle inequality implies

$$\sqrt{D_{ij}} \leq \sum_{k=0}^s \sqrt{D_{\gamma_k \gamma_{k+1}}} = \sum_{k=0}^s \sqrt{d_W^2(\gamma_k \gamma_{k+1})} = \|\gamma\| = l$$

In particular this holds for the shortest paths between y_i, y_j , $D_{ij} \leq d_G^2(y_i, y_j)$ for all $1 \leq i, j \leq N$. This gives

$$\begin{aligned} \sum_{i,j=1}^N D_{ij} - \underbrace{\sum_{i,j=1}^N D_{ij}^G}_{\text{const}} &= \sum_{i,j=1}^N (D_{ij} - D_{ij}^G) \\ &= - \sum_{i,j=1}^N |D_{ij} - D_{ij}^G| = -\|D - D^G\|_1 \quad \square \end{aligned}$$

Corollary 2.18. Let $[Y, E]$ be a connected graph with weights d_W . Then D^G is the unique solution

Note that we are optimizing over distance matrices, not euclidean distance matrices!

of

$$\max_{D \in EDM} \sum_{i,j=1}^N D_{ij}$$

s.t. $D_{ij} = d_W^2(y_i, y_j)$ if $(y_i, y_j) \in E$.

Corollary 2.19. Let $[Y, E]$ be a connected graph with weights d_W . Then

1. If $D^G \in EDM$, then D^G is the unique solution of the MVU problem (1)
2. The problem from definition 2.16 or (1) is equivalent to

$$\min_{D \in EDM} \|D - D^G\|_1$$

s.t. $D_{ij} = d_E^2(y_i, y_j)$ if $(y_i, y_j) \in E$.

Therefore MVU can be considered as a regularized shortest path problem.

Remark. Isomap can be seen to use the best EDM Approximation to D^G in the sense of

$$\min_{D \in EDM} \|H(D - D^G)H\|_F.$$

We saw in the exercises that this relates to cutting of eigenvalues!

We denote for a set S by \mathbb{R}^S the set of real valued functions on S . The restriction of a function $f \in \mathbb{R}^S$ to a subset \tilde{S} of S is denoted by $f|_{\tilde{S}}$.

Lemma 2.20. Let S be a set, $\tilde{S} \subset S, C \subseteq \mathbb{R}^{\tilde{S}}, f \in \mathbb{R}^S$ and $\tilde{f} \in \mathbb{R}^{\tilde{S}}$. Let $\|\cdot\|$ be a norm on $\mathbb{R}^{\tilde{S}}$ and $c, \epsilon \geq 0$. If

$$\|\tilde{f} - f|_{\tilde{S}}\| \leq c\epsilon \quad (4)$$

and

$$(1 - \epsilon)f|_{\tilde{S}} \in C \quad (5)$$

, then

$$\|\hat{f} - f|_{\tilde{S}}\| \leq (2c + \|\tilde{f}\|)\epsilon$$

for all $f \in \arg\min_{\tilde{f} \in C} \|\tilde{f} - f|_{\tilde{S}}\|$

Proof.

$$\begin{aligned} \|\hat{f} - f|_{\tilde{S}}\| &\leq \|\hat{f} - \tilde{f}\| + \|\tilde{f} - f|_{\tilde{S}}\| \\ &\stackrel{(4)}{=} \dots \end{aligned}$$

□

Theorem 2.21. Let $[Y, E]$ be a connected graph, $Y \subset M$, where M is a convex and compact manifold. For $D_G = [d_G^2(y_i, y_j)]_{i,j=1}^N$. We assume it holds that it holds for some $\epsilon > 0$ that

$$(1 - \epsilon)d_M^2(y_i, y_j) \leq d_G^2(y_i, y_j) \leq (1 + \epsilon)d_M^2(y_i, y_j)$$

Then the solution D of the MVU problem from definition 2.16 satisfies

$$\|D - D_M\|_1 \leq 3\|D_M\|_1 \epsilon \leq 3(N \cdot \text{diam}(M))^2 \epsilon$$

where $\text{diam}(M) = \sup_{x,y \in M} d_M(x, y)$.

Proof. Let $S = M \times M$, $\tilde{S} = Y \times Y$, $C = \{\hat{f} \in \mathbb{R}^{\tilde{S}} \mid [\hat{f}(y_i, y_j)]_{i,j=1}^N \in EDM\}$ $f(y, z) = d_M^2(y, z)$, $\tilde{f}(y_i, y_j) = d_G^2(y_i, y_j)$, $\|x\| = \|x(y_i, y_j)_{i,j=1}^N\|_1$, $c = \|D^M\|_1$.

Condition 1 of 2.20

$$\begin{aligned}\|\tilde{f} - f|_{\tilde{S}}\| &= \|D^G - D^M\|_1 \\ &\leq \sum_{i,j=1}^N \epsilon d_M^2(y_i, y_j) \\ &= \epsilon \|D_M\|_1 = \epsilon c\end{aligned}$$

Condition 2: We observe $d_M \in \text{EDM}$. Since EDM is a cone, it follows that $(1 - \epsilon)D_M \in \text{EDM}$, so (5) holds in lemma 2.20. This implies

$$\|\tilde{f} - f|_{\tilde{S}}\| \leq (2c + \|f\|_{\tilde{S}})\epsilon = 3\|D_M\|_1\epsilon$$

Therefore $\|DD_M\|_1 \leq 3\|D_M\|_1\epsilon$.

Additionally $\|D_M\|_1 = \sum_{i,j=1}^N d_M^2(y_i, y_j) \leq N^2 \max_{1 \leq i,j \leq N} d_M^2(i, y_j) \leq (N \text{diam } M)^2$ □

Start of lecture 20
(27.06.24)

2.3 Spectral clustering

CLustering \triangleq automatically finding groups in unlabeled data.

Let $[Y, E]$ be an undirected graph with **weight adjacency matrix** $w = [w_{jk}]_{j,k=1}^N$. In particular: if $(y_i, y_j) \in E \iff w_{ij} > 0$. The **degree of a node** $y_j \in Y$ is $d_j = \sum_{k=1}^N w_{jk}$ and the degree matrix is given by $D = \text{diag}(d_1, \dots, d_N)$. Often we consider

$$w_{jk} = \exp\left(\frac{-\|y_j - y_k\|_2^2}{2\sigma^2}\right)$$

If w is just the adjacency matrix, this coincides with the original definition of the degree of a node: the number of adjacent nodes

With that we define the **graph Laplacian**

$$L = D - W$$

Theorem 2.22. *The matrix $L = D - W$ satisfies the following properties*

1. For any $f \in \mathbb{R}^n$ it holds

$$f^T L f = \frac{1}{2} \sum_{j,k=1}^N w_{jk} (f_j - f_k)^2$$

2. L is spsd

3. The smallest eigenvalue of L is 0, the corresponding eigenvector is the vector of ones $\mathbf{1}$.

4. L has N nonnegative, real-valued eigenvalues $0 \leq \lambda_1 \leq \dots \leq \lambda_N$.

Proof. 1.:

$$\begin{aligned}f^T L f &= f^T D f - f^T W f = \sum_{j=1}^N d_j f_j^2 - \sum_{j,k=1}^N f_j f_k w_{jk} \\ &= \frac{1}{2} \left(\sum_{k=1}^N \underbrace{d_j}_{=\sum_k w_{jk}} f_j^2 - 2 \sum_{j,k=1}^N f_j f_k w_{jk} + \sum_{j=1}^N d_k f_k^2 \right) \\ &= \frac{1}{2} \sum_{j,k=1}^N w_{jk} (f_j - f_k)^2\end{aligned}$$

2.: D, W are symmetric, so is L . psd follows from 1., since $w \geq 0$

3.: psd \implies nonnegative eigenvalues. The fact that 0 is an eigenvalue follows from the definition of D via W

4.: follows from 1. - 3. □

Theorem 2.23. Let $[Y, E]$ be an undirected graph with nonnegative weights. Then the multiplicity k of the eigenvalue 0 of L gives the number of connected components A_1, \dots, A_k in the graph. The eigenspace of the eigenvalue 0 is spanned by the indicator vector $1_{A_1}, \dots, 1_{A_k}$ of these components.

Proof. For $k = 1$, assume f is eigenvector for $\lambda = 0$. So

$$0 = f^T L f = \sum_{j,k=1}^N \underbrace{w_{jk}}_{\geq 0} \underbrace{(f_j - f_k)^2}_{\geq 0} \underbrace{\quad}_{=0}$$

by $w_{jk} > 0 \implies f_j = f_k \implies f$ is constant for all nodes can be connected by a path in $[Y, E]$. Since all nodes of a connected component can be connected by a path, f is constant on the component. So for one connected component 1_{A_1} is an eigenvector for $\lambda = 0$, which is the indicator function of A_1 .

$k > 1$:

$$L = \begin{bmatrix} L_1 & & & \\ & L_2 & & \\ & & \ddots & \\ & & & L_N \end{bmatrix}$$

Wlog the nodes are ordered according to the component. So L is of block-diagonal form.

For a block diagonal matrix, the spectrum is given by the union of the spectra of the blocks. The eigenvectors of L are those of the blocks filled with zeros. Each L_i is a graph laplacian (GL), so the argument for $k = 1$ carries over. Together, L has eigenvalue 0 with multiplicity k and the indicator functions are eigenvectors.

This makes sense, since eigenfunctions for the eigenvalue 0 of the Laplace-Beltrami operator are constant functions, in the sense of constant on each connected component

Here we only used the distinction $w_{ij} > 0$, which means for this purpose it would be fine to just use the adjacency matrix!

Algorithm 6 Spectral clustering

Input: L, k

Output: Clusters A_1, \dots, A_k

Compute $U_k = EVD(L, k)$

for $do i = 1 \dots N$

x_j j th component of U in \mathbb{R}^k

end for

Cluster the points $\{x_i\}_i$ with k -means into clusters c_1, \dots, c_k

return $(A_i)_{i=1}^k$ with $A_i = \{j \mid x_j \in c_i\}$

We can determine k , but if we can also set k , if our graph is connected

Algorithm 7 Kmeans

Input: X, k

Output: k clusters that segment the data

Pick randomly k points x_j as cluster centers

while not converged **do**

for each x_j determine the closest cluster center x_j and assign it to it

for each cluster, update the cluster center to be the mean of the cluster

end while

Cluster the points $\{x_i\}_i$ with k -means into clusters c_1, \dots, c_k

return $(A_i)_{i=1}^k$ with $A_i = \{j \mid x_j \in c_i\}$

while cluster assignments do choose???

Consider now **random walks on graphs**, stochastic process that randomly jumps from node to node.

Transition probability p_{jk} pf jumping in one step from y_j to y_k

$$p_{jk} = \frac{w_{jk}}{d_j}$$

The **transition matrix** $P = [p_{jk}]_{j,k=1}^N$ of the random walk is

$$P = D^{-1}W.$$

Let $[Y, E]$ be connected and non-bipartite.

The corresponding random walk always processes a unique stationary distribution $\pi = [\pi_1, \dots, \pi_N]^\top$, since $[Y, E]$ is connected.

$$\pi_i = \frac{d_i}{\text{Vol}(y)}$$

where $\text{Vol}(y) = \sum_{i=1}^N d_i$. We introduce the **normalized graph laplacian** or **random Walk laplacian**

$$L_{\text{RW}} = D^{-1}L = I - D^{-1}W = I - P$$

and observe

$$L_{\text{RW}}u = \lambda u \iff Pu = (1 - \lambda)u \iff Lu = \lambda Pu.$$

We can also consider $L_{\text{sym}} = D^{-\frac{1}{2}}LD^{-\frac{1}{2}} = I - D^{-\frac{1}{2}}WD^{-\frac{1}{2}}$, the

symmetric normalized graph Laplacian.

We now use the **normalized cut (Ncut)** as a criteria to partition the graph, s.t. edges between partitions have very low weight and edges inside partitions have large weights.

Definition 2.24. Let $G = [Y, E, W]$ be a connected, undirected graph with symmetric, nonnegative edge weight matrix. The **normalized cut partition** of G into k **partitions** is given by

$$\min_{\substack{A_1, \dots, A_k \subset V: \cup A_j = V \\ A_j \cap A_k = \emptyset}} \text{Ncut}(A_1, \dots, A_k)$$

where $\text{Ncut}(A_1, \dots, A_k) = \frac{1}{2} \sum_{i=1}^k \frac{W(A_i \bar{A}_i)}{\text{vol}(A_i)}$ and $W(A, B) = \sum_{j \in A, k \in B} w_{j,k}$, $\bar{A}_i = V \setminus A_i$.

Theorem 2.25. Let $[Y, E, W]$ be a connected and non-bipartite graph. Assume that we run the random walk $(X_t)_{t \in \mathbb{N}}$ starting with x_0 in the stationary distribution π , i.e. $x_0 \sim \pi$. For disjoint subsets $A, B \subset V$, we denote

$$P(B | A) = P(X_1 \in B | X_0 \in A).$$

Then $\text{Ncut}(A, \bar{A}) = \frac{1}{2} (P(\bar{A} | A) + P(A | \bar{A}))$

This tells us that when minimizing Ncut , we look for a cut, through the graph s.t. a random walk rarely transitions from A to \bar{A} and vice versa.

A graph is bipartite if $\exists U \cup V = Y, U \cap V = \emptyset$ with $(x, y) \in E \implies x \in U, y \in V$ or vice versa

stationary means $P\pi = \pi$

Theorem 2.23 also works for L_{RW}

Review: Markov Chain / Random Walk

Family $(X_t)_{t \in \mathbb{N}}$ of random variables on a countable set $X = \{x_1, \dots, x_N\} \subseteq \Omega \subseteq \mathbb{R}^d$ is a **Markov chain**, if

$$\mathbb{P}(X_t = x_{i_t} | X_{t-1} = x_{i_{t-1}}, \dots, X_1 = x_{i_1}) = \mathbb{P}(X_t = x_{i_t} | X_{t-1} = x_{i_{t-1}}) \quad \forall t \in \mathbb{N}, i_j \in \{1, \dots, N\}$$

Now if $\mathbb{P}(X_t = x_i | X_{t-1} = x_j) = p_{ji}$ is the same for all times t , we have a

homogen Markov Chain and call it a **random walk** on the data.

Further, for such a random walk, $\pi = [\pi_1, \dots, \pi_N]^\top$ is the stationary distribution, if

1. $\pi \geq 0$ and $|\pi|_1 = 1$
2. $\pi = P^\top \pi$, i.e. $\pi_i = \sum_{j=1}^N P_{ji} \pi_j$

As said, we collect the p_{ji} in the **probability transition matrix**

$$P := [p_{ji}]_{j,i=1}^N$$

Start of lecture 21
(02.07.24)

This is not assumed to be symmetric

Proof of theorem 2.25.

$$\begin{aligned} P(x_0 \in A \mid x_1 \in B) &= \sum_{j \in A, k \in B} P(X_0 = j, X_1 = k) = \sum_{j \in A, k \in B} \pi_j p_{jk} \\ &= \sum_{j \in A, k \in B} \frac{d_j}{\text{Vol}(Y)} \frac{w_{jk}}{d_j} = \sum_{j \in A, k \in B} w_{jk} \end{aligned}$$

The stationary distribution of the random walk can be seen to be

$$\pi_i = \pi(x_i) = \frac{d(x_i)}{\text{Vol}(Y)} = \frac{d(x_i)}{\sum_{i=1}^N d(X_i)}$$

With this, we obtain

$$\begin{aligned} \mathbb{P}(X_1 \in B \mid X_0 \in A) &= \frac{\mathbb{P}(X_0 \in A, X_1 \in B)}{\mathbb{P}(X_0 \in A)} \\ &= \left(\frac{1}{\text{Vol}(Y)} \sum_{j \in A, k \in B} w_{jk} \right) \left(\frac{\text{Vol}(A)}{\text{Vol}(Y)} \right)^{-1} \\ &= \sum_{j \in A, k \in B} \frac{w_{jk}}{\text{Vol}(A)} = \frac{W(A, B)}{\text{Vol}(A)} \end{aligned}$$

The result follows from the definition of Ncut. \square

2.4 Diffusion Maps

Idea: consider longer paths, apply P a couple of times. probability of transition from y_i to y_j in t steps is $[P^t]_{ij}$.

Further the weights in P^t specify local geometry, we can capture with P^t geometric features at different scales. In other words, the markov chain shows fast and slow directions of propagation.

Definition 2.26. Let P be the transition matrix of a random walk on a graph. The diffusion distance is defined as

$$dd_t^2(y_i, y_j) = \sum_{k=1}^N |[P^t]_{ik} - [P^t]_{jk}|^2 \frac{1}{\pi(y_k)}$$

One interpretation of dd_t is as the weighted L_2 distance between the two probability distributions $([P^t]_{ik})_{k=1}^N$ and $([P^t]_{jk})_{k=1}^N$.

For increasing t the distributions become wider and flatter, therefore more overlap and the points become closer in the dd_t sense, with

Multiplying dd_t gives:

$$\begin{aligned} dd_t^2(y_i, y_j) &= \sum_{k=1}^N [P^t]_{ik}^2 \frac{1}{\pi_k} - 2 \sum_{k=1}^N ([P^t]_{ik} [P^t]_{jk}) \frac{1}{\pi_k} + \sum_{k=1}^N [P^t]_{jk}^2 \frac{1}{\pi_k} \\ &= \langle y_i, y_i \rangle_{\mathbb{P}^t \pi} - 2 \langle y_i, y_j \rangle_{P^t \pi} + \langle y_j, y_j \rangle_{P^t \pi} \\ &= \sum_{k=1}^N ([P^t]_{ik} - [P^t]_{jk})^2 \frac{1}{\pi_k} \end{aligned}$$

This is a weighted scalar product of a feature map that involves probabilities of sums from a point y_i to any other y_k in t steps.

Consider the weighted graph $[Y, E]$, where the weights are given by a kernel $k(x, y)$ that is non-negative.

We define $d(y_j) := \sum_{k=1}^N k(y_j, y_k)$ and observe

$$p(y_j, y_k) = \frac{k(y_j, y_k)}{d(y_i)}.$$

While P is not always symmetric, dd is!

The original choice is the gaussian kernel!

Then $p(y_j, y_k) \geq 0$, P is not symmetric, but $\sum_{k=1}^N p(y_j, y_k) = 1$. Now

$$(\star\star) \tilde{p}(y_j, y_k) = p(y_j, y_k) \sqrt{\frac{d(y_j)}{d(y_k)}} = \frac{k(y_j, y_k)}{\sqrt{d(y_j)} \sqrt{d(y_k)}}$$

\tilde{p} is symmetric and we can obtain a spectral decomposition of it:

$$\tilde{p}(y_j, y_k) = \sum_{i=1}^n \lambda_i \phi_i(y_j) \phi_i(y_k)$$

and from spectral graph theory one can obtain

$$\lambda_0 = 1 \geq \lambda_1 \geq \lambda_{N_1}.$$

Furthermore $\phi_0 = \sqrt{\pi}$, where $\pi(y_i) = \frac{d(y_i)}{\sum_{j=1}^N d(y_j)}$ is the stationary distribution of the associated random walk.

With $(\star\star)$ we get:

$$p(y_j, y_k) = \sum_{i=1}^N \lambda_i \psi_i(y_j) \chi_i(y_k)$$

where

$$\phi_i(y_j) = \frac{\phi_i(y_j)}{\pi(y_j)}$$

and

$$\chi_i(y_k) = \phi_i(y_k) \sqrt{\pi(y_k)}.$$

In particular $\psi_0 = 1$. For repeated applications of p , we obtain, due to the orthogonality of the eigendecomposition:

$$(\star) P_t(y_j, y_k) = \sum_{i=0}^{N-1} \lambda_i^t \underbrace{\psi_i(y_j)}_{\text{Coefficient}} \chi_i(y_k)$$

Definition 2.27. We define the family of diffusion maps $\{\Psi_t^s\}_{t \in \mathbb{N}, 1 \leq s \leq N_1}$ by

$$\Psi_t^s(y_k) = \begin{bmatrix} \lambda_1^t \psi_1(y_k) \\ \lambda_2^t \psi_2(y_k) \\ \vdots \\ \lambda_s^t \psi_s(y_k) \end{bmatrix}$$

where λ_i, ψ_i are from (\star) . We also set $\Psi_t = \Psi_t^{N-1}$. Each component $\lambda_i^t \psi_i$ is called a diffusion coordinate.

Added remark. For higher t we need less eigenfunctions for a good representation, since $\lambda_i \leq 1$! There is a coarsening effect for larger t .

$$P_t(y_k, y_k) = \sum_{i=0}^{N-1} \lambda_i^t \psi_i(y_j) \chi_i(y_k) \quad (6)$$

Start of lecture 22
(04.07.24)

Definition. Let P be the transition map of a random walk on a graph $[V, E]$ and $\{\psi_i\}_{i=0}^{N-1}$ be the eigenvector of P with eigenvalues $1 = \lambda_0 \geq \lambda_1 \geq \dots \geq \lambda_{N-1}$. The ψ are orthonormal w.r.t.

Here again ψ_i are vectors, $\psi_i(y_k) = (\psi_i)_k$ is the k th entry

$$\langle \psi_i, \psi_j \rangle_\pi = \sum_{k=1}^N \psi_i(y_k) \psi_j(y_k) \pi(y_k)$$

...

Theorem 2.28. *The diffusion distance dd_t is equal to the euclidean distance in the diffusion map space:*

$$\begin{aligned} dd_t^2(y_j, y_l) &= \|\Psi_t(y_j) - \Psi_t(y_l)\|^2 \\ &= \sum_{i=1}^{N-1} \lambda_i^{2t} (\psi_i(y_j) - \psi_i(y_l))^2 \end{aligned}$$

Proof. Inserting 6 into dd_t gives

$$dd_t^2(y_j, y_l) = \sum_{k=1}^N \underbrace{\left(\sum_{i=1}^{N-1} \lambda_i^t (\psi_i(y_j) - \psi_i(y_l)) \chi_i(y_k) \right)^2}_{\text{independent of } k} \frac{1}{\pi(y_k)}$$

Multiplying out and observe that

$$\langle \chi_i, \chi_k \rangle_{\frac{1}{\pi}} = \delta_{jk}$$

gives

$$dd_t^2(y_j, y_l) = \sum_{k=1}^N \lambda_i^{2t} (\psi_i(y_j) - \psi_i(y_l))^2 \quad \square$$

Besides the underlying manifold view, one can see the data as samples from the [equilibrium distribution](#) of stochastic dynamical systems.

1. Manifold view → recover the manifold structure regardless of the data distribution
2. Stochastic dynamical system view → density of points is the quantity of interest

To generalize diffusion maps one can add a density normalization step: 1.: with

$$q_k(y_i) = \sum_{l=1}^N k(y_i, y_l)$$

leads to the density normalization

$$K^{(\alpha)}(x, y) = \frac{K(x, y)}{q_k^\alpha(x) q_k^\alpha(y)}.$$

2.: set $d^{(\alpha)}(y_i) = \sum_{l=1}^N k^{(\alpha)}(y_i, y_l)$

$$P^{(\alpha)}(y_i, y_j) = \frac{K^{(\alpha)}(x, y)}{d^{(\alpha)}(y_i)}$$

Remark. *One can show that α corresponds to a specific type of flow field on the submanifold the data lies on:*

- $\alpha = 0$: flow field follows the normalized graph laplacian for gaussian kernel
- $\alpha = 1$: leads to a finite sample approximation of Brownian motion random process, flow fields follows the Laplace (-Beltrami) operator on the manifold
- $\alpha = \frac{1}{2}$: any α between 0 and 1 introduces a drift term in addition to the Brownian motion.
 $\alpha = \frac{1}{2}$ reflects the Folker-Plank dynamics of the random walk

$\alpha = 1$ is a reasonable choice, it captures the structure of the manifold

Added remark. *A bit more detail and information can be found in his script, since we are running low on time!*

2.5 t-Stochastic Neighborhood Embedding (t-SNE)

- two d curved manifold
- ten d curved manifold

It is problematic to use pairwise distances in two dimensions to approximate pairwise distances in ten dimensions!

- in ten dimensions it is possible to have 11 points that are mutually equidistant
- the volume of a sphere centered on a point y scales as r^d

t-SNE minimizes the (KL) divergence between two distributions. One measures pairwise similarities in the input objects, the other in the embedding of the corresponding points. Given $\{y_i\}_{i=1}^N$ and a distance function $d(y_i, y_j)$, commonly euclidean distance. We use

$$p(y_j | y_i) = \frac{\exp(-\frac{d(y_i, y_j)^2}{2\sigma_i^2})}{\sum_{k \neq i} \exp(-d(y_i, y_k)^2 / (2\sigma_i^2))}$$

y_j would get picked as a neighbor of y_i in proportion to a Gaussian probability density with variance σ_i

Further, $p(y_i | y_i) = 0$. To obtain symmetry we use define $p_{ij} = \frac{p(y_j | y_i) + p(y_i | y_j)}{2N}$.

The probability in the embedding space is modelled by a Student-t distribution (which will be a Cauchy distribution).

$$q_{ij} = \frac{(1 + \|x_i - x_j\|^2)^{-1}}{\sum_k \sum_{l \neq i} (1 + \|x_i - x_l\|^2)^{-1}}$$

t-SNE computes $\{x_1, \dots, x_n\}$ that minimize the [Kullback-Leibner-divergence](#)² between distributions p for Y and q for X .

$$C = \text{KL}(P | Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

the gradient of the KL-divergence between p and q is

$$\frac{\partial C}{\partial x_i} = 4 \sum_j (p_{ij} - q_{ij})(x_i - x_j)(1 + \|x_i - x_j\|^2)^{-1}$$

Added remark. *Interpreting t-SNE is hard, distances between clusters might not be meaningful! We will use quadrees to organize the data. We also use the Barnes-Hut approximation.*

Added remark. *U-map and t-SNE are both used in practice!*

I was ill, no notes. Content: Finishing t-SNE and start of

Start of lecture 23
(09.07.24)

2.6 Autoencoders

2.6.1 Variational Autoencoder

In [VAE](#) we use a probabilistic model $p(Y | X = x)$. and aim to approximate it. We generate a generative decoder by

- modelling the latent space distribution
- approximating the conditionals $p(Y | X)$ and $p(X | Y)$

Start of lecture 24
(11.07.24)
 X, Y are random variables, \bar{X}, \bar{Y} are in the dataset

²or [relative entropy](#)

In the latent space we assume

$$P(X) = \mathcal{N}(0, I).$$

The decoder density is modelled as

$$P(Y | X = x) = \mathcal{N}(f(x), \sigma^2 I)$$

He does not really care about X vs $x \dots$

where f is a [deep neural network](#).

$$f : \mathbb{R}^q \rightarrow \mathbb{R}^d$$

As the optimization criteria we use maximum likelihood of obtaining $\bar{Y} = \{y_1, \dots, y_N\}$

$$\begin{aligned} \log(P(\bar{Y})) &= \log(P(Y = \bar{Y})) \\ &\stackrel{y_i \text{ independent realizations of } Y}{=} \log\left(\prod_{i=1}^N p(Y = y_i)\right) \\ &= \sum_{i=1}^N \log(p(Y = y_i)) \\ &= \sum_{i=1}^N \log\left(\underbrace{\int_{\mathbb{R}^q} p(y_i | X = x)p(x)dx}_{= \mathbb{E}_p(P(y|X=x))}\right) \end{aligned}$$

To compute the integral we could use a Monte Carlo estimator, but that is slow to converge.

1. Compute $x_j \sim P(x)$, $\frac{1}{N} \sum_{j=1}^N \sum_{i=1}^N \log p(y_i | X = X_j)$

We can be seen to use [important sampling](#), where the x_i are drawn, where $P(y_i | X = X_j)$ is large. We construct $\nu(X | Y = y)$ to be able to sample those x that are likely to produce y . So instead we compute a sampling estimate of

I think it should be $P(y_j | X = x_j)$ here \dots

$$\int_{\mathbb{R}^q} P(Y | X = x)\nu(x | Y = y)dx = \mathbb{E}_\nu(P(Y | X = x))$$

Using the Kullback-Leibner divergence between ν and the true probability distribution $p(x | Y = y)$

$$K_{\nu, P} = \text{KL}(\nu(x | Y = y) \| P(x | Y = y)) = \mathbb{E}_\nu(\log(\nu(x | Y = y)) - \log(P(x | Y = y)))$$

Using Bayes theorem we see

$$P(x | Y = y) = \frac{P(y | X = x)P(X = x)}{P(Y = y)}$$

$$\begin{aligned} K_{\nu, P} &= \mathbb{E}_\nu\left(\log(\nu(x | Y = y)) - \log\left(\frac{P(y | X = x)P(X = x)}{P(Y = y)}\right)\right) \\ &= \mathbb{E}_\nu(\log(\nu(x | Y = y)) - \log(P(y | X = x)P(X = x))) + \underbrace{\log(P(Y = y))}_{\text{Constant}} \end{aligned}$$

We can rearrange this:

$$\underbrace{\log(P(Y = y))}_{\text{log-evidence of model}} - \underbrace{K_{\nu, P}}_{\geq 0} = \underbrace{\mathbb{E}_\nu(\log(P(y | X = x))) - \text{KL}(\nu(x | Y = y) \| p(X = x))}_{\text{Evidence lower bound (ElBo)}}$$

This was the goal to optimize.

If $\nu(x | Y = y)$ is a good approximation for $p(x | Y = y)$, $K_{\nu, P}$ will be small.

If we are maximizing the sum of the ElBos for each y_i we are maximizing $\log(P(\bar{Y}))$.

Maximizing ElBo achieves

- maximizing the log-likelihood of \bar{Y} and therefore the generative model $p(Y | X)$ becomes better
- minimizing the KL-divergence between ν and the posterior P and therefore the so called **inference model** ν becomes better

$$\nu(x | Y = y) \sim \mathcal{N}(g_1(y), \exp(g_2(y))^T I)$$

with $g_1, g_2 : \mathbb{R}^d \rightarrow \mathbb{R}^q$ two DNNs encoding the mean and variance of ν , where $\exp(g_2(y))$ is meant componentwise (on the diagonal). Usually g_1, g_2 have the same network architecture.

Here $\exp(g_2(y))$ is meant componentwise which than is taken as the diagonal?

$$(\tilde{f}, \tilde{g}_1, \tilde{g}_2) = \operatorname{argmin}_{f, g_1, g_2} \sum_{i=1}^N \underbrace{\mathbb{E}_{\nu} [\log(P(y_i | X = x))] - \text{KL}(\nu(x | Y = y_i) \| P(X = x))}_{L_i(f, g_1, g_2)}$$

Remark. The σ from $P(Y | X = x) \sim \mathcal{N}(f(x), \sigma^2 I)$ can be interpreted as a regularization parameter. It changes $P(y | X = x)$ without affecting the $KL(\nu \| p(X))$, it balances the two ElBo terms. σ reflects how accurately we expect the model to reconstruct Y .

One can see

$$\text{KL}(\nu(x | Y = y) \| p(X = x)) = \frac{1}{2} \left(- \sum_{i=1}^q (g_2(y))_i - q + \sum_{i=1}^q \exp(g_2(y))_i + \sum_{i=1}^q (g_1(y))_i^2 \right)$$

The derivative can be computed by automatic differentiation. For the first term we could perform a MC-estimator, with slow convergence. Instead we draw a sample $X_i \sim \nu(X | Y = y_i)$ for each y_i in the minibatch and use the **one-shot estimator** $\log(P(y_i | X = x))$ as an unbiased estimator. We then just average over all samples in the minibatch to evaluate the loss.

Essentially $\nabla_{g_1, g_2} \mathbb{E}_{\nu}(\dots) \neq \mathbb{E}_{\nu}(\nabla_{g_1, g_2} \dots)$.

We can use the **reparametrization trick**:

We draw $x(Y)$ according to $\nu(X | Y = y)$ and parametrize via

$$x(Y) = g_1(y) + \sqrt{\exp(g_2(y))} IZ$$

with $Z \sim \mathcal{N}(0, 1)$. Then

$$\begin{aligned} \nabla_{g_1, g_2} \mathbb{E}_Z(\log(P(y | X = x))) &= \nabla_{g_1, g_2} \mathbb{E}_{\nu}(\log(y | X = g_1(y) + \sqrt{\exp(g_2(y))} IZ)) \\ &= \mathbb{E}_Z(\nabla_{g_1, g_2} \log(P(y | X = x))) \\ &= \nabla_{g_1, g_2} \mathbb{E}_Z(\log(y | X = g_1(y) + \sqrt{\exp(g_2(y))} IZ)) \end{aligned}$$

Here we decouple our drawing process from taking the derivative

We define for $B \subseteq \{1, \dots, N\}$ the **minibatch loss** on B

Start of lecture 25
(16.07.24)

$$L_B(f, g_1, g_2) = \frac{1}{|B|} \sum_{i=1}^{|B|} L_{B_i}(f, g_1, g_2)$$

Algorithm 8 Stochastic minibatch gradient descent**Input:** Y , learning rate $\eta > 0$, minibatch size $\kappa \leq N$, number of epochs $S \in \mathbb{N}$ **Output:** Updated weights and biasesInitialize all weights W and biases b **for** $s = 1, \dots, S$ **do**subdivide $\{1, \dots, N\}$ into $\lceil \frac{N}{\kappa} \rceil$ disjoint subsets B_i **for** $B_1, \dots, B_{\lceil \frac{N}{\kappa} \rceil}$ **do**calculate f, g_1, g_2 for all $i \in B$ calculate $\nabla_{W,b} L_B(f, g_1, g_2)$ update each weight $w_{i,j}^{(l)} \leftarrow w_{i,j}^{(l)} - \eta \frac{\partial}{\partial w_{i,j}^{(l)}} L_B(f, g_1, g_2)$ update each bias $b_j^{(l)} \leftarrow b_j^{(l)} - \eta \frac{\partial}{\partial b_j^{(l)}} L_B(f, g_1, g_2)$ **end for****end for**forward propagation
w.r.t. all weights and
biases

One can also

- use momentum or Nestorov update
- adapt step size / learning rate
- use ADAM adaptive moment estimation

Added remark. *Definition, theorems, connections! Also kernels in both topics, there is a connection! Can you tell me something about? 30min +x**Bring student ID and government ID. Questions are not always precise. Don't talk too long about one topic. How quick do we come to the point. He won't ask questions on the exercises, but they can help.*

2.7 Generative Diffusion Models

Definition 2.29. A time-discrete **forward diffusion process** starting at a data point $y^{(i)} \in \mathbb{R}^d$, which has been drawn according to some unknown distribution p , is defined by

$$y^{(i+1)} = \sqrt{1 - \beta_{i+1}} y^{(i)} + \sqrt{\beta_{i+1}} \epsilon \quad (\star)$$

for each $i \in W$. Here $0 < \beta_i < 1$ is called the noise level and $\epsilon \sim \mathcal{N}(0, I)$, $\epsilon \in \mathbb{R}^d$.For $i \rightarrow \infty$, the y^{i+1} become completely random, normally distributed vectors. Consider the stochastic differential equations (SDE) in the form of

$$\frac{dY(t)}{dt} = \mu(y(t), t) + \sigma(y(t), t) \frac{dW(t)}{dt} \quad (\star \star \star)$$

with $Y, W : [0, \infty) \rightarrow \mathbb{R}^d$ are stochastic processes. The drift coefficient $\mu : \mathbb{R}^d \times [0, \infty) \rightarrow \mathbb{R}^d$ and the diffusion coefficient $\sigma : \mathbb{R}^d \times [0, \infty) \rightarrow \mathbb{R}$. W is modelled as a multidimensional Brownian Motion / Wiener process, where $W(\tilde{t}) - W(t)$ are independent and modelled as $\mathcal{N}(0, (\tilde{t} - t)I)$.

In particular,

$$\frac{dY(t)}{dt} = -\frac{1}{2}\beta(t)Y(t) + \sqrt{\beta(t)} \frac{dW(t)}{dt} \quad (\star \star \star \star)$$

with $\beta(t) : [0, \infty) \rightarrow \mathbb{R}$ noise variance.We use for discretization the Euler-Maruyama-scheme with step size δt :

$$Y(t + \delta t) = Y(t) - \frac{1}{2}\beta(t)Y(t)\delta t + \sqrt{\beta(t)}\sqrt{\delta t}\epsilon \quad (\star \star)$$

Consider (\star) with $y^{(i)} = Y(i\delta t)$ and $\beta_i = \beta((i+1)\delta t)\delta t$

$$Y((i+1)\delta t) = \sqrt{1 - \beta(i\delta t)}Y(i\delta t) + \sqrt{\beta(i\delta t)}\epsilon$$

Use Taylor expansion on the drift part around $\delta t = 0$:

$$\sqrt{1 - \beta(i\delta t)\delta t} = 1 - \frac{\beta(i\delta t)}{2}\delta t + O((\delta t)^2)$$

So up to a second order term in δt , $(\star\star)$ resembles \star .

2.7.1 Reversed diffusion process

To draw according to the data distribution, we need the reverse diffusion process \bar{Y} . It transforms $Y(t)$ for t very large, back.

Generally, it can be shown that a forward SDE $(\star\star\star)$ can be reversed in time to get the reverse time SDE. In case of $(\star\star\star)$ one gets

$$\frac{d\bar{Y}}{dt} = \frac{1}{2}\beta(t)(\bar{Y}(t) + \underbrace{g(\bar{Y}(t), t)}_{\text{Score function}}) + \sqrt{\beta(t)}\frac{d\bar{W}(t)}{dt}$$

for some $g : \mathbb{R}^d \times [0, \infty) \rightarrow \mathbb{R}^d$. Euler-Maruyama gives

$$\bar{Y}(t - \delta t) = \bar{Y}(t) - \frac{1}{2}\beta(t)(\bar{Y}(t) + g(\bar{Y}(t), t)) + \sqrt{\beta(t)}\delta t\epsilon$$

Observations: So in both directions we compute terms depending on the current iterate and a Gaussian increment.

$$P^\star(y^{(i)} | y^{(i-1)}) = \mathcal{N}(\sqrt{1 - \beta_i}y^{(i-1)}, \beta_i I)$$

$$P^\star(y^{(i-1)} | y^{(i)}, y^{(0)}) = \mathcal{N}(\tilde{m}_i, \tilde{\sigma}_i I)$$

where $\tilde{m}_i, \tilde{\sigma}_i$ only depend on $y^{(i)}, y^{(0)}$ and $\beta_j, 1 \leq j \leq i$.

But $y^{(0)}$ is not known when reversing the process for new images, so we aim to approximate this by

$$P(y^{(i-1)} | y^{(i)}) = \mathcal{N}(m_i, \sigma_i I)$$

with m_i, σ_i can be learned or set. The noise level is typically fixed $\sigma_i = \beta_i$ and m_i is the output of a neural network given $y^{(i)}$, learned from training data Y_1, \dots, Y_N .

As for VAE

$$\mathbb{E}_{P^\star(Y(0))}[P(Y(0))]$$

Start of lecture 26
(18.07.24)

This cannot directly be computed, so we use the ElBo

$$\log P(Y(0)) = \log \left(\int P^\star(y(1:T) | y(0)) \frac{P(y(0:T))}{P^\star(y(1:T) | y(0))} dy(1:T) \right)$$

$$y(1:T) = (y(1), \dots, y(T))$$

Now with $p(y(0:T)) = \underbrace{P^\star(y(T))}_{\mathcal{N}(0, I)} \prod_{i=1}^T P(Y(i-1) | Y(i))$ and T the number of steps of the

forward process, T large enough. One can see

$$\log(Y(0)) \geq \int P^\star(Y(1:T) | Y(0)) \log(P^\star(Y(T))) \prod_{i=1}^T \frac{P(Y(i-1) | Y(i))}{P^\star(Y(i) | Y(i-1))}$$

$$= \mathbb{E}_{P^\star} \left[\log P^\star(Y(T)) + \sum_{i=1}^T \log \frac{P(Y(i-1) | Y(i))}{P^\star(Y(i) | Y(i-1))} \right]$$

Given training data, we can approximate this empirically

$$\approx \frac{1}{N} \sum_{i=1}^N \left(\log(P^\star(Y_j(T))) + \sum \log \frac{P(Y_j(i-1) | Y_j(i))}{P^\star(Y_j(i) | Y_j(i-1))} \right)$$

where $P^*(Y_j(i) | Y_j(i-1))$ is known and we can obtain

$$y_j(i)$$

by sampling from $P^*(Y_j(i) | y_j(i-1))$. Therefore we can evaluate this expression for each $y_j = y_j(0)$.

We assume that T is large enough, s.t. $P^*(Y_j(T)) \sim \mathcal{N}(0, I)$. Together, we can evaluate the expression and minimize it with a SDG-type algorithm w.r.t. weights and biases of the network for m_i .

With the learned w_i , we then propagate a random sample $Y(T) \sim \mathcal{N}(0, I)$ by computing the m_i and following the backwards process for $i = T_1, \dots, 1$ and get the new data $y(0)$ approximately from the image distribution.

Chapter 3:

Misc

3.1 What we didn't cover

For dimensionality reduction: There is further probabilistic views on that! We can also use an embedding, which more strongly reflects our data classes. We might want to include this information in our dim. reduction.

From the stochastic view (think PCA, we assume linear map). But we can just think of random matrices, there is the Johnson-Lindenstrauss lemma.

In view of NNs we can also talk about restricted Boltzmann machines.

For the kernel stuff: We didn't do sampling inequalities! Idea of oversampling. We can also do Wendler-Kernels?

Compactly supported
kernels \rightarrow sparse matrices

3.2 Review of the lecture notes

Kernels

- Kernels, applications
- Feature map / mercer kernel view , kernels as generalizations of scalar products
- Collection of properties, how do we do computations with kernels
- Riesz-Rep / Functional analysis view
- Kernels give a way to represent functions / reproducing kernel hilber spaces
- Theorem 1.9 semi is missing
- Subspaces and connections to mercer kernels
- Function approximation through projection onto subspaces
- Best approximation / minimizing norm (Hilbert projection theorem)
- Power function and error bounds
- Power function in terms of the kernel
- Power function and condition of kernel matrix (eigenvalues, smallest eigenvalue and power functions), relation to fill distance (balancing act)
- Polynomial reproduction / cone condition, a bit technical
- Seite 36: Beweisidee Taylorexpanssion wichtig
- Hermite interpolation
- Connection to PDEs

- cpsd \rightarrow later kernel PCA (for order 1)
- Loss functions (as a relaxation of interpolation)
- Some loss functions
- Expected and empirical risk
- Regularization (also relates to power functions and fill distances)
- representer theorem and regularization operators
- Greens kernel (vllt. nicht so wichtig?)
- eigendecomposition and kernels \rightarrow mercer kernel
- Gaussian process
- SVMs (key algorithm) (connection to hinge loss?)
- 1.55 distances and cpsd of order 1

dimensionality reduction

- Distances, kernels connections
- Work with data PCA / MDS / SVD
- Several ways to view / derive PDA
- Key thing: MDS is the same as PCA just slightly different. Similarities ...
- Theorem 2.6 important to connect scalar products to distances
- 3 different ways to compute the embedding (page 90)
- Problems in higher dimensions
- Graph distances, Iso map ...
- Under certain assumptions Iso map ... is reasonable
- Parallel transport unfolding (not as important? Differential geometry)
- Perturbation analysis
- Nonlinear PCA and Kernel MDS
- Maximum variance unfolding, find a kernel ..., similar to iso map, also works with graph distances
- Spectral clustering, diffusion processes and random walks on graphs, cuts ...
- K-means
- diffusion maps, non symmetric matrix \rightarrow construct symmetric matrix
- diffusion distance
- density aware diffusion (only sketched)
- Two steps in the diffusion map algorithm
- t-SNE(why can we compute this? quadrees!)
- Autoencoder, VAE (ElBo)
- Diffusion process and SDEs

Appendix

Tutorials

All norms and scalar products w.r.t. H if not specified.

Start of tutorial 02
(23-04-24)

a)

Proof.

$$\begin{aligned} |h_n(t) - h_m(t)| &\stackrel{\text{Reproduction equality}}{\leq} |\langle h_n - h_m, k(t, \cdot) \rangle| \\ &\stackrel{\text{C.S.}}{\leq} \|h_n - h_m\|_H \|k(t, \cdot)\|_H \rightarrow 0 \end{aligned}$$

Therefore $h_n(t)$ is Cauchy $\implies h_n(t)$ converges. □

b)

Proof. Show definition is well defined:

1.: $\langle f, g \rangle \in \mathbb{R}$ for all $f, g \in \mathcal{N}_k$.

f_n, g_n Cauchy sequences, $f_n \rightarrow f, g_n \rightarrow g$ pointwise.

1.1.: Show that $\lim_{n \rightarrow \infty} \|f_n\|_H$ exists in \mathbb{R} .

$$|\|f_n\|_H - \|f_m\|_H| \leq \|f_n - f_m\|_H \implies \|f_n\|_H \text{ is Cauchy}$$

and therefore bounded.

1.2.: Show $\langle f, g \rangle_H = \lim_{n \rightarrow \infty} \langle f_n, g_n \rangle_H$ exists in \mathbb{R} .

$$\begin{aligned} |\langle f_n, g_n \rangle_H - \langle f_m, g_m \rangle_H| &= |\langle f_n - f_m, g_n \rangle_H + \langle f_m, g_n - g_m \rangle_H| \\ &\leq \|f_n - f_m\|_H \|g_n\|_H + \|f_m\|_H \|g_n - g_m\|_H \end{aligned}$$

Let f_n, f'_n, g_n Cauchy, $f_n \rightarrow f, f'_n \rightarrow f, g_n \rightarrow g$ pointwise.

We need:

$$\begin{aligned} \lim_{n \rightarrow \infty} \langle f_n, g_n \rangle_H &= \lim_{n \rightarrow \infty} \langle f'_n, g_n \rangle_H \\ &\iff \lim_{n \rightarrow \infty} \langle f_n - f'_n, g_n \rangle_H = 0 \\ &\iff f_n \rightarrow 0 \text{ pointwise} \implies \langle f_n, g_n \rangle \rightarrow 0 \\ &\iff \forall f_n \rightarrow 0 \text{ pointwise } \langle f_n, f_n \rangle \rightarrow 0 \\ &\iff f = 0 \implies \langle f, f \rangle_H = 0 \end{aligned}$$

$h_n = f_n - f'_n$.

1.: f_n is Cauchy in H

$$\|(f_n - f'_n) - (f_m - f'_m)\|_H \leq \|f_n - f_m\|_H + \|f'_n - f'_m\|_H$$

2.: h_n bounded by M (follows from Cauchy)

3.: $h_n \rightarrow 0$ in H

Fix $l \in \mathbb{N}$ and let $h_l = \sum_{i=1}^m a_i k(x, \cdot)$

$$\langle h_n, h_l \rangle = \lambda_{A^{(l)}, x^{(l)}}(h_n) = \sum_{i=1}^m a_i h_m(x_i) \rightarrow 0$$

$$\implies (\star) : \forall \epsilon > 0 : \exists n_0 \in \mathbb{N} : \forall n \geq n_0 |\langle h_n, h_l \rangle| < \epsilon$$

h_n is Cauchy $\implies \forall \delta > 0 \exists l \in \mathbb{N} \forall n \geq l$:

$$\|h_n - h_l\| < \delta/M.$$

Choose n_0 s.t. (\star) holds with δ

$$\begin{aligned} \|h_n\|_H^2 &= \langle h_n, h_n \rangle_H = \langle h_n, h_l \rangle + \langle h_n, h_n - h_l \rangle \\ &\leq \delta + \|h_n\|_H \|h_n - h_l\| < \delta + M \frac{\delta}{M} \end{aligned}$$

Choose $\delta = \frac{\epsilon}{2}$.

Still have to show $\langle f, f \rangle = 0$:

$$\lim_{n \rightarrow \infty} f_n(t) = \lim_{n \rightarrow \infty} \langle f_n, k(t, \cdot) \rangle_H \leq \lim_{n \rightarrow \infty} \|f_n\| \underbrace{\|k(t, \cdot)\|}_{\text{bounded}}$$

□

c)

Proof. f_n Cauchy:

$f_n \rightarrow f$ in \mathcal{N}_k and $f \in \mathcal{N}_k$.

Construct Cauchy sequence in H :

$\lim_{n \rightarrow \infty} f_n(k) = f_n$ and $f_n^{(k)} \in H$.

$\forall n \in \mathbb{N}$ choose $k(n)$ such that

$$\|f_n - f_n^{(k(n))}\| < \frac{1}{n}$$

$g_n = f_n^{(k(n))}$. Let $\epsilon > 0$, $n_0 = \max\{\frac{3}{\epsilon}, m\}$

$\forall k, l \geq m : \|f_k - f_l\| < \epsilon/3$

$$\begin{aligned} \|g_n - g_m\| &\leq \|f_n^{(k(n))} - f_n\| + \|f_n - f_m\| + \|f_m - f_m^{(k(n))}\| \\ &\leq \frac{\epsilon}{3} + \frac{\epsilon}{3} + \frac{\epsilon}{3} = \epsilon \end{aligned}$$

for all $n, m \geq n_0$.

Set $f \in \mathcal{N}_k$ as the limit of g_n (because it is a Cauchy sequence in H).

2.: f_n converges to f in \mathcal{N}_k .

$$\begin{aligned} \|f_n - f\| &\leq \|f_n - f_{n+1}\| + \|f_{n+1} - f_{n+1}^{(k(n+1))}\| + \|g_{n+1} - f\| \\ &< \frac{\epsilon}{3} + \frac{1}{n+1} + \frac{\epsilon}{3} = \epsilon \end{aligned}$$

□

$$\begin{aligned}\exp(-(x-y)^2) &= \exp(-(x^2 - 2xy + y^2)) \\ &= \exp(-x^2) \exp(+2xy) \exp(-y^2) \\ &= \sum_{k=0}^{\infty} \frac{(2xy)^k}{k!} \exp(-x^2) \exp(-y^2) \\ &= \sum_{k=0}^{\infty} \underbrace{\frac{2^k}{k!}}_{=\lambda_k} \underbrace{\frac{x^k}{\exp(x^2)}}_{\varphi_k(x)} \underbrace{\frac{y^k}{\exp(y^2)}}_{\varphi_k(y)}\end{aligned}$$

$$D_n(x, y) = \sum_{k=-n}^n \overline{\exp(-ikx)} \exp(-iky)$$

List of Lectures

- [Lecture 01](#)
- [Lecture 02](#)
- [Lecture 03](#)
- [Lecture 04](#)
- [Lecture 05](#)
- [Lecture 06](#)
- [Lecture 07](#)
- [Lecture 08](#)
- [Lecture 09](#)
- [Lecture 10](#)
- [Lecture 11](#)
- [Lecture 12](#)
- [Lecture 13](#)
- [Lecture 14](#)
- [Lecture 15](#)
- [Lecture 16](#)
- [Lecture 17](#)
- [Lecture 18](#)
- [Lecture 19](#)
- [Lecture 20](#)
- [Lecture 21](#)
- [Lecture 22](#)
- [Lecture 23](#)
- [Lecture 24](#)
- [Lecture 25](#)
- [Lecture 26](#)