

Foundations of Machine Learning

learning note For reading translation

我真的不懂忧郁

Delft University of Technology

Foundations of Machine Learning

learning note For reading translation

by

我真的不懂忧郁

Student Name	Student Number
--------------	----------------

First Surname	1234567
---------------	---------

Instructor: I. Surname

Teaching Assistant: I. Surname

Project Duration: Month, Year - Month, Year

Faculty: Faculty of Aerospace Engineering, Delft

Cover: Canadarm 2 Robotic Arm Grapples SpaceX Dragon by NASA under
CC BY-NC 2.0 (Modified)

Style: TU Delft Report Style, with modifications by Daan Zwaneveld

Preface

A preface...

我真的不懂忧郁
Delft, September 2024

Summary

A summary...

目录

Preface	i
Summary	ii
Nomenclature	iv
1 Kernel Methods	1
1.1 Introduction	1
1.2 Positive definite symmetric kernel	1
1.3 Reproducing kernel Hilbert Space	3
1.4 Kernel-Based Algorithms	5
1.5 Negative definite symmetric kernels	6
1.6 Sequence Kernel	6
2 基于流形的学习	7
2.1 PCA 和 LDA	7
2.2 拓扑流形的概念	7
2.3 多尺度变换	7
2.4 局部线性嵌入	7
2.5 拉普拉斯特征映射	7
2.6 核函数与度量——NDS 核	7
2.7 理论成果	7
References	8
A Source Code Example	9
B Task Division Example	10

Nomenclature

If a nomenclature is required, a simple template can be found below for convenience. Feel free to use, adapt or completely remove.

Abbreviations

Abbreviation	Definition
ISA	International Standard Atmosphere
...	

Symbols

Symbol	Definition	Unit
V	Velocity	[m/s]
...		
ρ	Density	[kg/m ³]
...		

Chapter 1

Kernel Methods

1.1. Introduction

$K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ 称为 \mathcal{X} 上的 **Kernels**。

theorem 1.1.1: (Mercer's condition) 令 $\mathcal{X} \subset \mathbb{R}^N$ 是一个紧集^a, $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ 是一个对称连续函数, 则

$$K(x, x') = \sum_{n=0}^{\infty} \lambda_n \phi_n(x) \phi_n(x'), \quad \lambda_n > 0 \text{ is eigenvalue} \quad (1.1)$$

当且仅当 $\forall c \in L^2(\mathcal{X})$, 下面的条件成立

$$\int \int_{\mathcal{X} \times \mathcal{X}} c(x) c(x') K(x, x') dx dx' \geq 0 \quad (1.2)$$

^a \mathcal{X} 是紧集, 则存在有限个开覆盖

proof.

□

Mercer's condition 是核方法中的一个重要概念, 尤其在支持向量机 (SVM) 和核函数的理论中起着关键作用。它为一个函数能否作为合法的核函数提供了数学判据, 保证了凸性从而保证可以取到全局最小值。合法的核函数用于将数据从低维空间映射到高维空间, 在高维空间中可以更加容易地进行线性分割。

1.2. Positive definite symmetric kernel

$K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ 称为**正定核** (*positive definite symmetric, PDS*), 当对于任何 $\{x_1, \dots, x_m\} \subseteq \mathcal{X}$, 矩阵

$$\mathbf{K} = [K(x_i, x_j)]_{ij} \in \mathbb{R}^{m \times m} \quad (1.3)$$

是半正定对称矩阵, 即 $\forall \mathbf{c} = (c_1, \dots, c_m)^T \in \mathbb{R}^{m \times 1}$,

$$\mathbf{c}^T \mathbf{K} \mathbf{c} = \sum_{i,j=1}^n c_i c_j K(x_i, x_j) \geq 0 \quad (1.4)$$

example 1.2.1: (Polynomial Kernels) 对任意常数 $c > 0$, 一个 d 维多项式核定义为

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^N, \quad K(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}' + c)^d \quad (1.5)$$

多项式核将输入空间映射到更高维度的空间。作为一个例子, $N = 2$ 的输入空间, 二阶多项式多项式对应于下面的内积

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^2, \quad K(\mathbf{x}, \mathbf{x}') = (x_1 x'_1 + x_2 x'_2 + c)^2 = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2} x_1 x_2 \\ \sqrt{2} c x_1 \\ \sqrt{2} c x_2 \\ c \end{bmatrix}^T \begin{bmatrix} x'_1{}^2 \\ x'_2{}^2 \\ \sqrt{2} x'_1 x'_2 \\ \sqrt{2} c x'_1 \\ \sqrt{2} c x'_2 \\ c \end{bmatrix} \quad (1.6)$$

可以看到这是维度为 6 的内积。

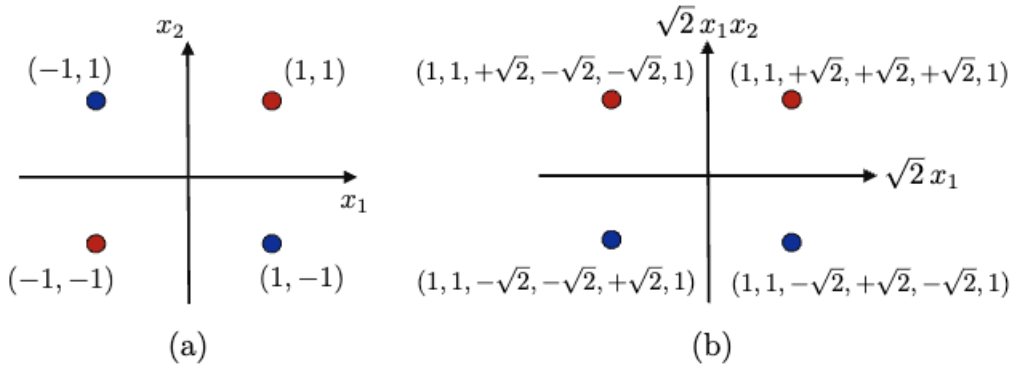


图 1.1: 异或问题

example 1.2.2: (Gaussian Kernels) 对于任意的常数 $\sigma > 0$, 高斯核 (Gaussian kernel) 或者称径向基函数 (radial basis function, RBF) 定义为

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^N, \quad K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}' - \mathbf{x}\|^2}{2\sigma^2}\right) \quad (1.7)$$

高斯核是应用中使用最为频繁的。我们将会证明高斯核是 PDS 核并且它能通过正规化的方法构造

$$K' : (\mathbf{x}, \mathbf{x}') \rightarrow \exp\left(-\frac{(\mathbf{x} \cdot \mathbf{x}')^n}{\sigma^2}\right) \quad (1.8)$$

example 1.2.3: (Sigmoid Kernels) 对于任意的实数 $a, b \geq 0$, 一个 Sigmoid kernel 定义为

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^N, \quad K(\mathbf{x}, \mathbf{x}') = \tanh(a(\mathbf{x} \cdot \mathbf{x}') + b) \quad (1.9)$$

1.3. Reproducing kernel Hilbert Space

对于度量空间 (X, d) ，如果存在完备度量空间任何 (\hat{X}, \hat{d}) ，它的某个稠密子空间 X_0 和 X 等距同构，则 (\hat{X}, \hat{d}) 是 (X, d) 的一个等距同构。度量空间都有完备化，且完备化在等距同构的意义下唯一。

$$d_1(x, y) = d_2(T(x), T(y)), \quad \forall x, y \in X \quad (1.10)$$

theorem 1.3.1: 任何度量空间 (X, d) 都存在一个完备度量空间 (\hat{X}, \hat{d}) ，使得 (X, d) 和 (\hat{X}, \hat{d}) 的一个稠子空间等距，且在等距的意义下，这样的空间 (\hat{X}, \hat{d}) 是唯一的，称 (\hat{X}, \hat{d}) 是 (X, d) 的完备化空间。

proof.

1. 首先构造空间 (\hat{X}, \hat{d}) ;

把 (X, d) 中的 *Cauchy* 列全体表示为 \hat{X} 。如果两个 *Cauchy* 列

2. 证明 (X, d) 与 (\hat{X}, \hat{d}) 中的一个稠子空间等距;
3. 证明 (\hat{X}, \hat{d}) 完备;
4. 在等距的意义下，证明完备化空间的唯一性;

□

lemma 1.3.2: (Cauchy-Schwarz inequality for PDS kernels) 令 K 为一个 PDS kernel，则对于任意的 $x, x' \in \mathcal{X}$

$$K(x, x') \leq K(x, x)K(x', x') \quad (1.11)$$

proof. 考虑矩阵

$$\mathbf{K} = \begin{pmatrix} K(x, x) & K(x, x') \\ K(x', x) & K(x', x') \end{pmatrix} \quad (1.12)$$

根据定义， K 是 PDS 核，则 \mathbf{K} 是 SPSD 对于所有的 $x, x' \in \mathcal{X}$ 。 \mathbf{K} 的特征值的积 $\det(\mathbf{K})$ 必须是非负的，因此 $K(x', x) = K(x, x')$ ，我们有

$$\det(\mathbf{K}) = K(x, x)K(x', x') - K(x, x')^2 \geq 0 \quad (1.13)$$

□

theorem 1.3.3: 令 $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ 是一个 PDS 核，则存在一个 Hilbert Space \mathbb{H} 以及 $\Phi : \mathcal{X} \rightarrow \mathbb{H}$ ，使得

$$\forall x, x' \in \mathcal{X}, \quad K(x, x') = \langle \Phi(x), \Phi(x') \rangle \quad (1.14)$$

\mathbb{H} 有如下名为再生 (Reproducing) 的性质

$$\forall h \in \mathbb{H}, \forall x \in \mathcal{X}, \quad h(x) = \langle h, K(x, \cdot) \rangle \quad (1.15)$$

\mathbb{H} 称为再生核希尔伯特空间 (reproducing kernel Hilbert Space, RKHS)。

proof. 对于任意的 $x \in \mathcal{X}$, 定义 $\Phi : \mathcal{X} \rightarrow \mathbb{R}^{\mathcal{X}}$

$$\forall x, x' \in \mathcal{X}, \quad K(x, x') = \langle \Phi(x), \Phi(x') \rangle \quad (1.16)$$

定义 \mathbb{H}_0 为 $\Phi(x)$ 的线性组合的集合

$$\mathbb{H}_0 := \left\{ \sum_{i \in I} a_i \Phi(x_i) : a_i \in \mathbb{R}, x_i \in \mathcal{X}, |I| < \infty \right\} \quad (1.17)$$

在 $\mathbb{H}_0 \times \mathbb{H}_0$ 中引入一个运算 $\langle \cdot, \cdot \rangle$, 对于所有的 $f, g \in \mathbb{H}_0$, 其中 $f = \sum_{i \in I} a_i \Phi(x_i)$, $g = \sum_{j \in J} b_j \Phi(x'_j)$

$$\langle f, g \rangle = \sum_{i \in I, j \in J} a_i b_j K(x_i, x'_j) = \sum_{j \in J} b_j f(x'_j) = \sum_{i \in I} a_j g(x_i) \quad (1.18)$$

根据定义自然 $\langle \cdot, \cdot \rangle$ 是一个对称算子。最后两个等号展示 $\langle f, g \rangle$ 并不依赖于 f 和 g 的形式, 以及显示 $\langle \cdot, \cdot \rangle$ 的双线性。对于任意 $f = \sum_{i \in I} a_i \Phi(x_i) \in \mathbb{H}_0$, 由于 K 是一个 *PDS* 核, 我们有

$$\langle f, f \rangle = \sum_{i, j} a_i a_j K(x_i, x_j) \geq 0 \quad (1.19)$$

因此, $\langle \cdot, \cdot \rangle$ 是一个双线性半正定型。更一般地, 对于任意的 $f_1, \dots, f_m \in \mathbb{H}_0$, $c_1, c_2, \dots, c_m \in \mathbb{R}$,

$$\sum_{i, j=1}^m c_i c_j \langle f_i, f_j \rangle = \left\langle \sum_{i=1}^m c_i f_i, \sum_{j=1}^m c_j f_j \right\rangle \geq 0 \quad (1.20)$$

因此, $\langle \cdot, \cdot \rangle$ 是 \mathbb{H}_0 上的 *PDS* 核, 因此, 对于任意的 $f \in \mathbb{H}_0$ 和任意的 $x \in \mathcal{X}$, 根据 *Cauchy-Schwarz inequality*, 我们写为

$$\langle f, \Phi(x) \rangle^2 \leq \langle f, f \rangle \langle \Phi(x), \Phi(x) \rangle \quad (1.21)$$

根据再生性: 对于任意的 $f = \sum_{i \in I} a_i \Phi(x_i) \in \mathbb{H}_0$, 根据 $\langle \cdot, \cdot \rangle$ 的定义

$$\forall x \in \mathcal{X}, \quad f(x) = \sum_{i \in I} a_i K(x_i, x) = \langle f, \Phi(x) \rangle \quad (1.22)$$

因此, 对于所有的 $x \in \mathcal{X}$, $|f(x)|^2 \leq \langle f, f \rangle K(x, x)$ 。这意味着 $\langle \cdot, \cdot \rangle$ 定义了一个 \mathbb{H}_0 上的内积, 因此 \mathbb{H}_0 称为了一个 *pre-Hilbert space*。下面只要对 \mathbb{H}_0 能完备化就能形成 \mathbb{H} 。根据 *Cauchy-Schwarz* 不等式, 对 $\forall x \in \mathcal{X}$, $f \mapsto \langle f, \Phi(x) \rangle$ 是 *Lipschitz* 的, 因此是连续的, 因此 \mathbb{H}_0 在 \mathbb{H} 中稠密。

□

Normlized PDS Kernels

lemma 1.3.4: 令 K 是一个 *PDS kernel*, 则 K 的规范核 K' 也是 *PDS kernel*.

PDS Kernels Closure Properies

theorem 1.3.5: *PDS kernel* 在和、积、张量积、逐点极限下是闭集，且可以展开成幂级数

$$\sum_{n=0}^{\infty} a_n x^n, a_n \geq 0 \text{ for } \forall n \in \mathbb{N} \quad (1.23)$$

1.4. Kernel-Based Algorithms

SVMs with PDS kernels

由于一个 *PDS* 核暗示着定义一个内积，我们能延伸 *SVM*

$$\begin{aligned} \max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j K(x_i, x_j) \\ \text{subject to : } 0 \leq \alpha_i \leq C \wedge \sum_{i=1}^m \alpha_i y_i = 0, i \in [m] \end{aligned} \quad (1.24)$$

假设的解 h 能写成

$$h(x) = \text{sgn} \left(\sum_{i=1}^m \alpha_i y_i K(x_i, x) + b \right) \quad (1.25)$$

其中 $b = y_i - \sum_{j=1}^m \alpha_j y_j K(x_j, x_i), \forall x_i, 0 < \alpha_i < C$ 。我们能将优化问题重新写成向量的形式，通过 K 的核矩阵 \mathbf{K}

$$\begin{aligned} \max_{\alpha} 2 \mathbf{1}^T \alpha - (\alpha \circ \mathbf{y})^T \mathbf{K} (\alpha \circ \mathbf{y}) \\ \text{subject to : } \mathbf{0} \leq \alpha \leq \mathbf{C} \wedge \alpha^T \mathbf{y} = 0 \end{aligned} \quad (1.26)$$

Representer theorem

theorem 1.4.1: 令 $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ 是一个 *PDS* 核， \mathbb{H} 对应 *RKHS*，则，对于任意的不减函数 $G : \mathbb{R} \rightarrow \mathbb{R}$ 和任意的损失函数 $L : \mathbb{R}^m \rightarrow \mathbb{R} \cup \{+\infty\}$ ，优化问题

$$\arg \min_{h \in \mathbb{H}} F(h) = \arg \min_{h \in \mathbb{H}} G(\|h\|_{\mathbb{H}}) + L(h(x_1), \dots, h(x_m)) \quad (1.27)$$

存在一个解的形式为

$$h^* = \sum_{i=1}^m \alpha_i K(x_i, \cdot) \quad (1.28)$$

如果 G 在先前假设是增函数名，则任何解都是这样的形式。

proof.

□

Learning guarantees

theorem 1.4.2: (Rademacher complexity of kernel-based hypotheses) 令 $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ 是一个 *PDS* 核， $\Phi : \mathcal{X} \rightarrow \mathbb{R}$ 是一个关于 K 的特征映射。令 $S \subseteq \{x : K(x, x) \leq r^2\}$ 是一个尺寸

为 m 的例子, 令 $\mathcal{H} = \{x \mapsto \langle \mathbf{w}, \Phi(x) \rangle : \|\mathbf{w}\|_{\mathbb{H}} \leq \Lambda, \exists \Lambda > 0\}$, 则

$$\hat{\mathcal{R}}_S(\mathcal{H}) \leq \frac{\Lambda \sqrt{\text{Tr}[\mathbf{K}]}}{m} \leq \sqrt{\frac{r^2 \Lambda^2}{m}} \quad (1.29)$$

proof.

□

corollary 1.4.3: (Margin bounds for kernel-based hypotheses) 令 $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ 是一个 PDS 核, 其中 $r^2 = \sup_{x \in \mathcal{X}} K(x, x)$, 令 $\Phi : \mathcal{X} \rightarrow \mathbb{H}$ 是一个关于 K 的特征映射, 令 $\mathcal{H} = \{\mathbf{x} \mapsto \mathbf{w} \cdot \Phi(x) : \|\mathbf{w}\|_{\mathbb{H}} \leq \Lambda, \exists \Lambda \geq 0\}$. 固定 $\rho > 0$, 则对于任意的 $\delta > 0$

$$R(h) \leq \hat{R}_{S,p}(h) + 2\sqrt{\frac{r^2 \Lambda^2 / \rho^2}{m}} + \sqrt{\frac{\log \frac{1}{\delta}}{2m}} \quad (1.30)$$

$$R(h) \leq \hat{R}_{S,p}(h) + 2\sqrt{\frac{\text{Tr}[\mathbf{K}] \Lambda^2 / \rho^2}{m}} + 3\sqrt{\frac{\log \frac{2}{\delta}}{2m}} \quad (1.31)$$

1.5. Negative definite symmetric kernels

definition 1.5.1: (Negative definite symmetric kernels, NDS) 一个核 $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ 称为负定对称 (Negative-definite symmetric, NDS), 如果这是一个对称核并且 $\forall (x_1, \dots, x_m) \subseteq \mathcal{X}$ 以及 $\mathbf{c} \in \mathbb{R}^{m \times 1}$, 满足 $\mathbf{1}^T \mathbf{c} = 0$ 下面的关系成立

$$\mathbf{c}^T \mathbf{K} \mathbf{c} \leq 0 \quad (1.32)$$

明显地, 如果 K 是 PDS , 则 $-K$ 是 NDS , 但反过来一般来说并不成立。

example 1.5.1: (Squared distance——NDS kernel)

1.6. Sequence Kernel

Weighted transducers

Rational kernel

Chapter 2

基于流形的学习

2.1. PCA 和 LDA

2.2. 拓扑流形的概念

2.3. 多尺度变换

保持度量不变

2.4. 局部线性嵌入

保持线性结构不变

2.5. 拉普拉斯特征映射

近邻图，拉普拉斯矩阵

2.6. 核函数与度量——NDS 核

2.7. 理论成果

References

- [1] I. Surname, I. Surname, and I. Surname. “The Title of the Article”. In: *The Title of the Journal* 1.2 (2000), pp. 123–456.

Chapter A

Source Code Example

Adding source code to your report/thesis is supported with the package listings. An example can be found below. Files can be added using `\lstinputlisting[language=<language>]{<filename>}`.

```
1 """
2 ISA Calculator: import the function, specify the height and it will return a
3 list in the following format: [Temperature,Density,Pressure,Speed of Sound].
4 Note that there is no check to see if the maximum altitude is reached.
5 """
6
7 import math
8 g0 = 9.80665
9 R = 287.0
10 layer1 = [0, 288.15, 101325.0]
11 alt = [0,11000,20000,32000,47000,51000,71000,86000]
12 a = [-.0065,0,.0010,.0028,0,-.0028,-.0020]
13
14 def atmosphere(h):
15     for i in range(0,len(alt)-1):
16         if h >= alt[i]:
17             layer0 = layer1[:]
18             layer1[0] = min(h,alt[i+1])
19             if a[i] != 0:
20                 layer1[1] = layer0[1] + a[i]*(layer1[0]-layer0[0])
21                 layer1[2] = layer0[2] * (layer1[1]/layer0[1])**(-g0/(a[i]*R))
22             else:
23                 layer1[2] = layer0[2]*math.exp((-g0/(R*layer1[1]))*(layer1[0]-layer0[0]))
24     return [layer1[1],layer1[2]/(R*layer1[1]),layer1[2],math.sqrt(1.4*R*layer1[1])]
```

Chapter B

Task Division Example

If a task division is required, a simple template can be found below for convenience. Feel free to use, adapt or completely remove.

表 B.1: Distribution of the workload

Task	Student Name(s)
Summary	
Chapter 1 Introduction	
Chapter 2	
Chapter 3	
Chapter *	
Chapter * Conclusion	
Editors	
CAD and Figures	
Document Design and Layout	