Kernels and Reproducing Kernel Hilbert Spaces

Krikamol Muandet
Max Planck Institute for Intelligent Systems
Tübingen, Germany

October 9, 2018

References

Some materials in this tutorial are based partly on the following references.

- **Support Vector Machines**. (Chapter 4)
  Steinwart and Christmann, Springer, 2008.
- **Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond**.
  Schölkopf and Smola, MIT Press, 2001.
- **Kernel Mean Embedding of Distributions: A Review and Beyond**. (Chapter 2)
  M, *Fukumizu, Sriperumbudur, and Schölkopf*. FnT ML, 2017.
- **Reproducing Kernel Hilbert Spaces in Probability and Statistics**.
  Berlinet and Thomas-Agnan. Springer, 2004.

Background Check

- Linear algebra
  - Linear equations
  - Matrix addition and multiplication
  - Eigenvalue, eigenvector and eigenspace

Background Check

- Linear algebra
  - Linear equations
  - Matrix addition and multiplication
  - Eigenvalue, eigenvector and eigenspace
- Functional analysis
  - Euclidean space and vector space
  - Inner product space and Hilbert space
  - Normed space and Banach space

Background Check

- Linear algebra
  - Linear equations
  - Matrix addition and multiplication
  - Eigenvalue, eigenvector and eigenspace
- Functional analysis
  - Euclidean space and vector space
  - Inner product space and Hilbert space
  - Normed space and Banach space
- Basics of machine learning
  - Classification and regression problems

Background Check

- Linear algebra
  - Linear equations
  - Matrix addition and multiplication
  - Eigenvalue, eigenvector and eigenspace
- Functional analysis
  - Euclidean space and vector space
  - Inner product space and Hilbert space
  - Normed space and Banach space
- Basics of machine learning
  - Classification and regression problems
- Kernels and reproducing kernel Hilbert spaces

Learning Problems

- Consider a simple **binary classification** problem.

Learning Problems

- Consider a simple **binary classification** problem.
- An input space $\mathcal{X} \subset \mathbb{R}^d$ and an output space $\mathcal{Y} = \{-1, +1\}$.

Learning Problems

- Consider a simple **binary classification** problem.
- An input space $\mathcal{X} \subset \mathbb{R}^d$ and an output space $\mathcal{Y} = \{-1, +1\}$.
- A training data $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n) \sim \mathbb{P}(X, Y)$.

Learning Problems

- Consider a simple **binary classification** problem.
- An input space $\mathcal{X} \subset \mathbb{R}^d$ and an output space $\mathcal{Y} = \{-1, +1\}$.
- A training data $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n) \sim \mathbb{P}(X, Y)$.
- Learn a predictor $f \in \mathcal{F}$ from the training data such that
  1. $f(\mathbf{x}_i) \approx y_i$ for $i = 1, \ldots, n$ and
  2. the predictor f **generalizes** well to previously unseen data.

Learning Problems

- Consider a simple **binary classification** problem.
- An input space $\mathcal{X} \subset \mathbb{R}^d$ and an output space $\mathcal{Y} = \{-1, +1\}$.
- A training data $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n) \sim \mathbb{P}(X, Y)$.
- Learn a predictor $f \in \mathcal{F}$ from the training data such that
  1. $f(\mathbf{x}_i) \approx y_i$ for $i = 1, \ldots, n$ and
  2. the predictor f **generalizes** well to previously unseen data.
- Throughout the tutorial, we consider

$$\mathcal{F} = \left\{ f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} \mid \mathbf{w} \in \mathbb{R}^d \right\}.$$

Learning Problems

- Consider a simple **binary classification** problem.
- An input space $\mathcal{X} \subset \mathbb{R}^d$ and an output space $\mathcal{Y} = \{-1, +1\}$.
- A training data $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n) \sim \mathbb{P}(X, Y)$.
- Learn a predictor $f \in \mathcal{F}$ from the training data such that
  1. $f(\mathbf{x}_i) \approx y_i$ for $i = 1, \ldots, n$ and
  2. the predictor f **generalizes** well to previously unseen data.
- Throughout the tutorial, we consider

$$\mathcal{F} = \left\{ f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} \mid \mathbf{w} \in \mathbb{R}^d \right\}.$$

- Given any $f \in \mathcal{F}$, our decision function is

$$g(\mathbf{x}) = \begin{cases} -1 & \text{if } f(\mathbf{x}) < 0 \\ +1 & \text{if } f(\mathbf{x}) \geq 0 \end{cases}.$$

Learning Problems

- Consider a simple **binary classification** problem.
- An input space $\mathcal{X} \subset \mathbb{R}^d$ and an output space $\mathcal{Y} = \{-1, +1\}$.
- A training data $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n) \sim \mathbb{P}(X, Y)$.
- Learn a predictor $f \in \mathcal{F}$ from the training data such that
  1. $f(\mathbf{x}_i) \approx y_i$ for $i = 1, \ldots, n$ and
  2. the predictor f **generalizes** well to previously unseen data.
- Throughout the tutorial, we consider

$$\mathcal{F} = \left\{ f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} \mid \mathbf{w} \in \mathbb{R}^d \right\}.$$

- Given any $f \in \mathcal{F}$, our decision function is

$$g(\mathbf{x}) = \begin{cases} -1 & \text{if } f(\mathbf{x}) < 0 \\ +1 & \text{if } f(\mathbf{x}) \geq 0 \end{cases}.$$

- We also write $g(\mathbf{x}) = \text{sgn}(f(\mathbf{x})) = \text{sgn}(\mathbf{w}^\top \mathbf{x})$.

Learning Algorithms

- We will consider a **Perceptron** learning rule
- Initialize $\mathbf{w}_0 = \mathbf{0}$. For $t = 1, \ldots, T$:
  1. $\hat{y}_t = \text{sgn}(\mathbf{w}_t^\top \mathbf{x}_t)$.
  2. If $\hat{y}_t = y_t$, do nothing.
  3. If $\hat{y}_t \neq y_t$, update $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + y_t \mathbf{x}_t$.

Learning Algorithms

- We will consider a **Perceptron** learning rule
- Initialize $\mathbf{w}_0 = \mathbf{0}$. For $t = 1, \ldots, T$:
  1. $\hat{y}_t = \text{sgn}(\mathbf{w}_t^\top \mathbf{x}_t)$.
  2. If $\hat{y}_t = y_t$, do nothing.
  3. If $\hat{y}_t \neq y_t$, update $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + y_t \mathbf{x}_t$.
- Any solution can be expressed as

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i.$$

Learning Algorithms

- We will consider a **Perceptron** learning rule
- Initialize $\mathbf{w}_0 = \mathbf{0}$. For $t = 1, \ldots, T$:
  1. $\hat{y}_t = \text{sgn}(\mathbf{w}_t^\top \mathbf{x}_t)$.
  2. If $\hat{y}_t = y_t$, do nothing.
  3. If $\hat{y}_t \neq y_t$, update $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + y_t \mathbf{x}_t$.
- Any solution can be expressed as

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i.$$

- The **dual perceptron algorithm** uses the prediction rule

$$\hat{y} = \text{sgn}(\mathbf{w}^\top \mathbf{x}) = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i\right)^\top \mathbf{x} = \text{sgn} \sum_{i=1}^{n} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{x}).$$

Learning Algorithms

- We will consider a **Perceptron** learning rule
- Initialize $\mathbf{w}_0 = \mathbf{0}$. For $t = 1, \ldots, T$:
  1. $\hat{y}_t = \text{sgn}(\mathbf{w}_t^\top \mathbf{x}_t)$.
  2. If $\hat{y}_t = y_t$, do nothing.
  3. If $\hat{y}_t \neq y_t$, update $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + y_t \mathbf{x}_t$.
- Any solution can be expressed as

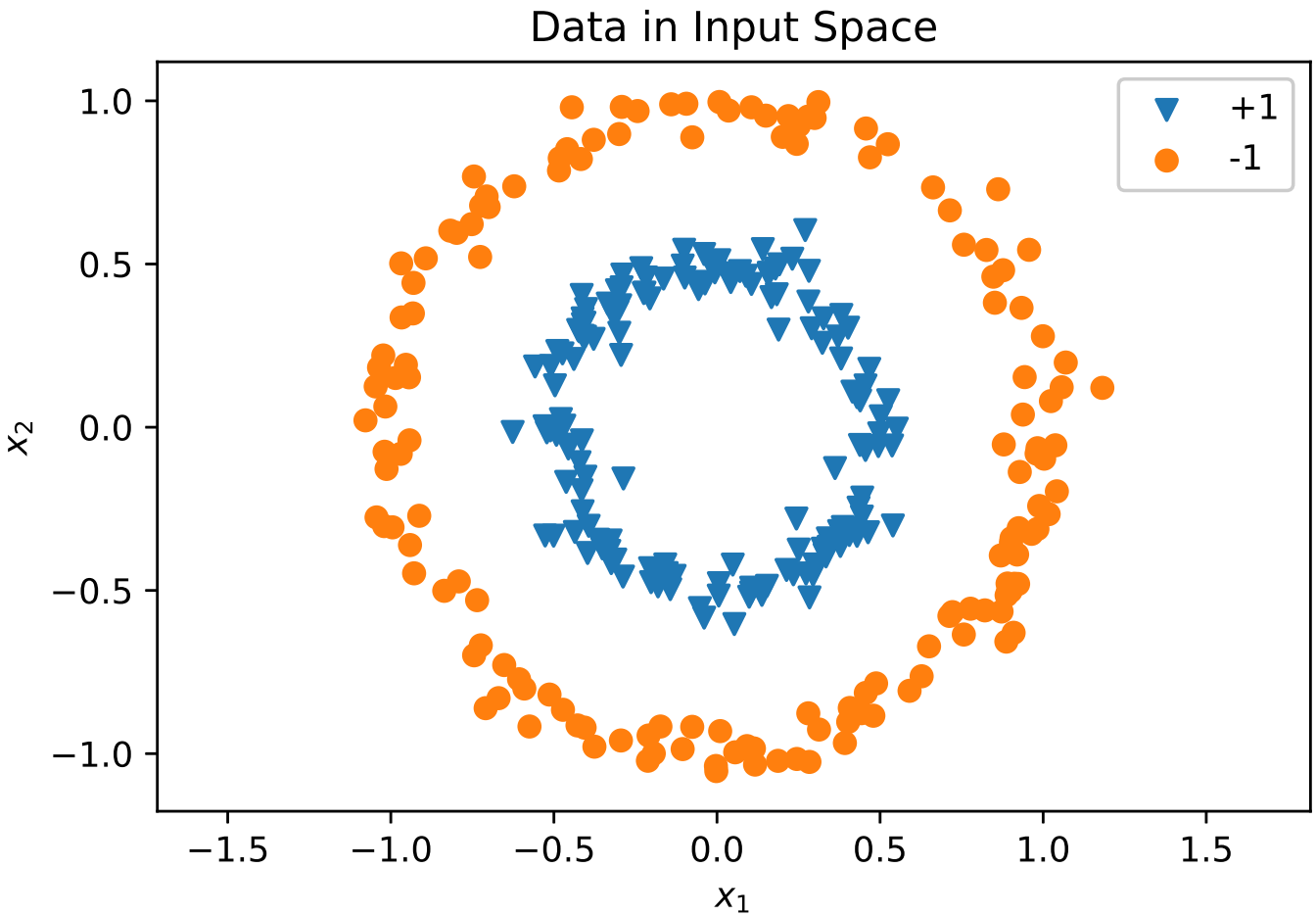
$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i.$$

- The **dual perceptron algorithm** uses the prediction rule

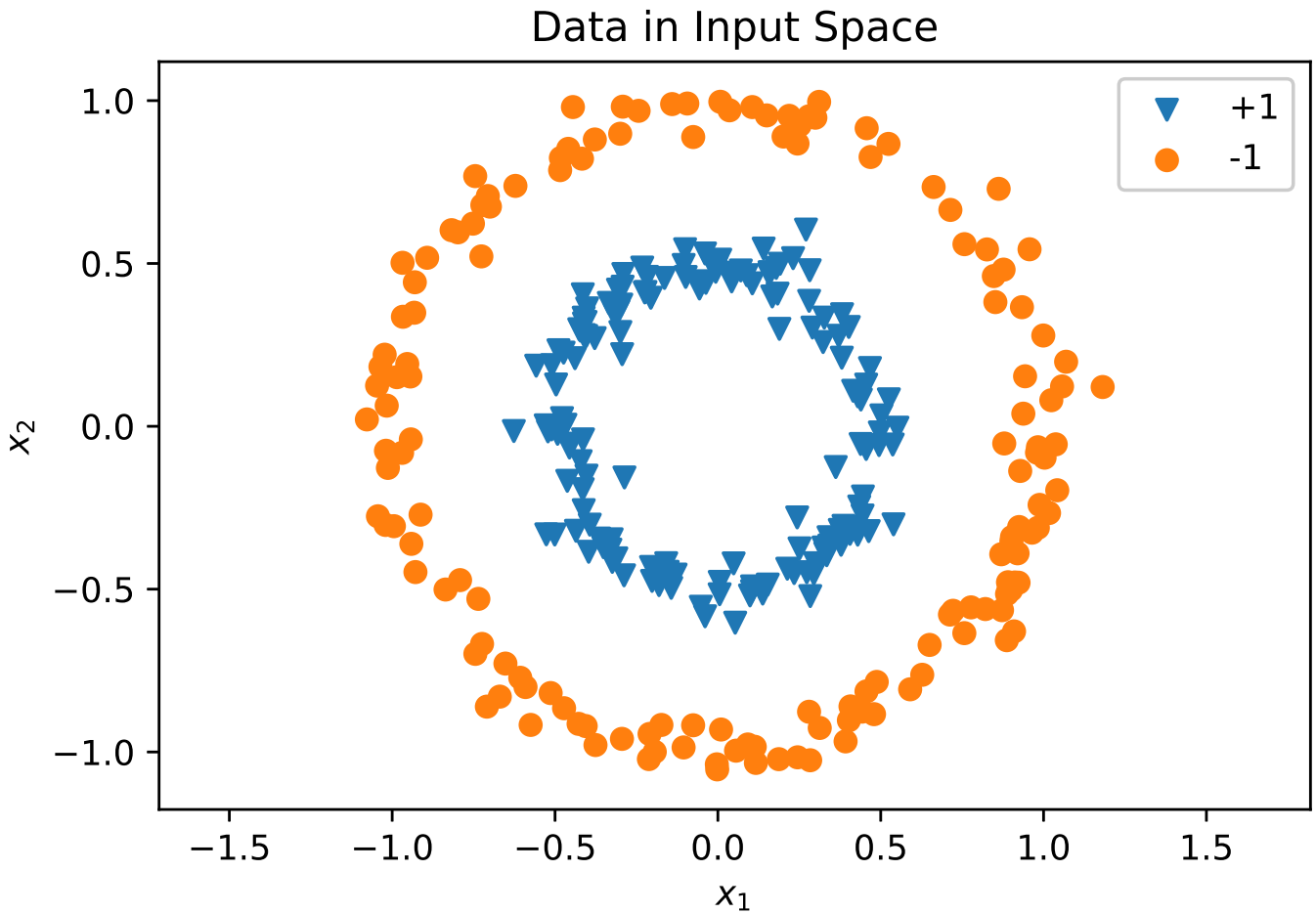
$$\hat{y} = \text{sgn}(\mathbf{w}^\top \mathbf{x}) = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i\right)^\top \mathbf{x} = \text{sgn} \sum_{i=1}^{n} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{x}).$$

- **Step 3:** If $\hat{y}_t \neq y_i$, update $\alpha_i \leftarrow \alpha_i + 1$.

Classification Problem



Classification Problem



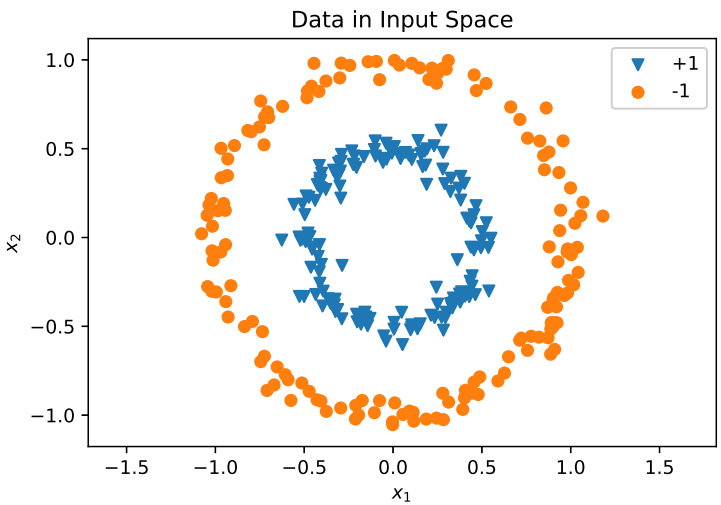
Question: How would you solve the problem?

Feature Map

$$\phi \ : \ (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$$

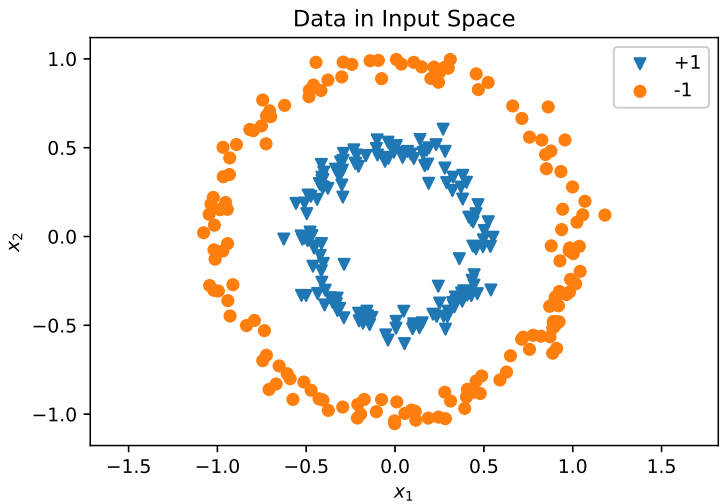
Feature Map

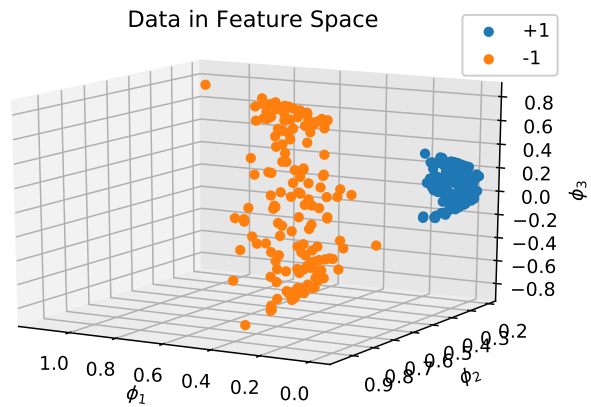
$$\phi \ : \ (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$$

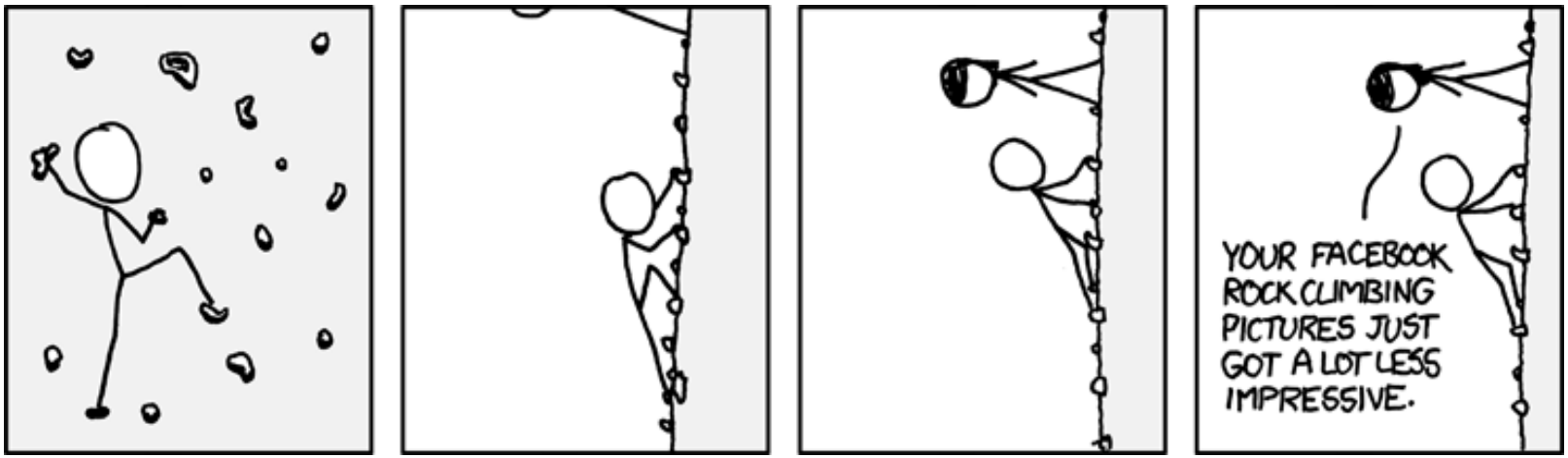


Feature Map

$$\phi \ : \ (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$$







Our recipe:

1. construct a non-linear feature map $\phi : \mathcal{X} \to \mathcal{H}$.
2. evaluate $D_\phi = \{\phi(x_1), \phi(x_2), \ldots, \phi(x_n)\}$.
3. learn in $\mathcal{H}$ using D_ϕ.

Dual Perceptron Revisited

- Recall our feature map $\phi : (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$

Dual Perceptron Revisited

- Recall our feature map $\phi : (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$
- The prediction rule of **dual perceptron algorithm** becomes

$$\hat{y} = \text{sgn}(\mathbf{w}^\top \phi(\mathbf{x})) = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle\right).$$

Dual Perceptron Revisited

- Recall our feature map $\phi : (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$
- The prediction rule of **dual perceptron algorithm** becomes

$$\hat{y} = \text{sgn}(\mathbf{w}^\top \phi(\mathbf{x})) = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle\right).$$

- An inner product between $\phi(\mathbf{x})$ and $\phi(\mathbf{z})$ in $\mathbb{R}^3$

$$\begin{aligned} \langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle_{\mathbb{R}^3} &= (x_1^2, x_2^2, \sqrt{2}x_1x_2)^\top (z_1^2, z_2^2, \sqrt{2}z_1z_2) \\ &= x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2 \end{aligned}$$

Dual Perceptron Revisited

- Recall our feature map $\phi : (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$
- The prediction rule of **dual perceptron algorithm** becomes

$$\hat{y} = \text{sgn}(\mathbf{w}^\top \phi(\mathbf{x})) = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle\right).$$

- An inner product between $\phi(\mathbf{x})$ and $\phi(\mathbf{z})$ in $\mathbb{R}^3$

$$\begin{aligned}
\langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle_{\mathbb{R}^3} &= (x_1^2, x_2^2, \sqrt{2}x_1x_2)^\top (z_1^2, z_2^2, \sqrt{2}z_1z_2) \\
&= x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2 \\
&= (x_1z_1)^2 + (x_2z_2)^2 + 2(x_1z_1)(x_2z_2) \\
&= (x_1z_1 + x_2z_2)^2
\end{aligned}$$

Dual Perceptron Revisited

- Recall our feature map $\phi : (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$
- The prediction rule of **dual perceptron algorithm** becomes

$$\hat{y} = \text{sgn}(\mathbf{w}^\top \phi(\mathbf{x})) = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle\right).$$

- An inner product between $\phi(\mathbf{x})$ and $\phi(\mathbf{z})$ in $\mathbb{R}^3$

$$\begin{aligned}
\langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle_{\mathbb{R}^3} &= (x_1^2, x_2^2, \sqrt{2}x_1x_2)^\top (z_1^2, z_2^2, \sqrt{2}z_1z_2) \\
&= x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2 \\
&= (x_1z_1)^2 + (x_2z_2)^2 + 2(x_1z_1)(x_2z_2) \\
&= (x_1z_1 + x_2z_2)^2 \\
&= (\mathbf{x} \cdot \mathbf{z})^2.
\end{aligned}$$

Dual Perceptron Revisited

- Recall our feature map $\phi : (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$
- The prediction rule of **dual perceptron algorithm** becomes

$$\hat{y} = \text{sgn}(\mathbf{w}^\top \phi(\mathbf{x})) = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle\right).$$

- An inner product between $\phi(\mathbf{x})$ and $\phi(\mathbf{z})$ in $\mathbb{R}^3$

$$\begin{aligned}
\langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle_{\mathbb{R}^3} &= (x_1^2, x_2^2, \sqrt{2}x_1x_2)^\top (z_1^2, z_2^2, \sqrt{2}z_1z_2) \\
&= x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2 \\
&= (x_1z_1)^2 + (x_2z_2)^2 + 2(x_1z_1)(x_2z_2) \\
&= (x_1z_1 + x_2z_2)^2 \\
&= (\mathbf{x} \cdot \mathbf{z})^2.
\end{aligned}$$

- For $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, define a **kernel** function $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}')^2$.

Dual Perceptron Revisited

- Recall our feature map $\phi : (x_1, x_2) \longmapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$
- The prediction rule of **dual perceptron algorithm** becomes

$$\hat{y} = \text{sgn}(\mathbf{w}^\top \phi(\mathbf{x})) = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle\right).$$

- An inner product between $\phi(\mathbf{x})$ and $\phi(\mathbf{z})$ in $\mathbb{R}^3$

$$\begin{aligned}
\langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle_{\mathbb{R}^3} &= (x_1^2, x_2^2, \sqrt{2}x_1x_2)^\top (z_1^2, z_2^2, \sqrt{2}z_1z_2) \\
&= x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2 \\
&= (x_1z_1)^2 + (x_2z_2)^2 + 2(x_1z_1)(x_2z_2) \\
&= (x_1z_1 + x_2z_2)^2 \\
&= (\mathbf{x} \cdot \mathbf{z})^2.
\end{aligned}$$

- For $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, define a **kernel** function $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}')^2$.
- Hence, $\hat{y} = \text{sgn}\left(\sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x})\right)$.

Kernels

Definition

A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a **kernel** on $\mathcal{X}$ if there exists a Hilbert space $\mathcal{H}$ and a map $\phi : \mathcal{X} \to \mathcal{H}$ such that for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ we have

$$k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{H}}. \tag{1}$$

We call ϕ a **feature map** and $\mathcal{H}$ a **feature space** of k.

Kernels

Definition

A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a **kernel** on $\mathcal{X}$ if there exists a Hilbert space $\mathcal{H}$ and a map $\phi : \mathcal{X} \to \mathcal{H}$ such that for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ we have

$$k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{H}}. \tag{1}$$

We call ϕ a **feature map** and $\mathcal{H}$ a **feature space** of k.

Example

1. $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}')^2$ for $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^2$
   - $\phi(\mathbf{x}) = (x_1^2, x_2^2, \sqrt{2} x_1 x_2)$.
   - $\mathcal{H} = \mathbb{R}^3$.

Examples of Kernels

- **Linear kernel:** $k(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'$
  - $\phi(\mathbf{x}) = \mathbf{x}$ and $\mathcal{H} = \mathbb{R}$
  - $\phi(\mathbf{x}) = (\mathbf{x}/\sqrt{2}, \mathbf{x}/\sqrt{2})$ and $\mathcal{H} = \mathbb{R}^2$

Examples of Kernels

- **Linear kernel:** $k(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'$
  - $\phi(\mathbf{x}) = \mathbf{x}$ and $\mathcal{H} = \mathbb{R}$
  - $\phi(\mathbf{x}) = (\mathbf{x}/\sqrt{2}, \mathbf{x}/\sqrt{2})$ and $\mathcal{H} = \mathbb{R}^2$
- **Polynomial kernel:** $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}' + c)^m$ for $c \geq 0$
  - $\phi(\mathbf{x}) = \left(\sqrt{\binom{m}{n_1, \ldots, n_{d+1}}} x_1^{n_1} \cdots x_d^{n_d} \cdot c^{n_{d+1}/2} \right)_{n_i \geq 0, \sum_{i=1}^{d+1} n_i = m}$
  - $\dim(\mathcal{H}) = \binom{d+m}{m}$

Examples of Kernels

- **Linear kernel:** $k(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'$
  - $\phi(\mathbf{x}) = \mathbf{x}$ and $\mathcal{H} = \mathbb{R}$
  - $\phi(\mathbf{x}) = (\mathbf{x}/\sqrt{2}, \mathbf{x}/\sqrt{2})$ and $\mathcal{H} = \mathbb{R}^2$
- **Polynomial kernel:** $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}' + c)^m$ for $c \geq 0$
  - $\phi(\mathbf{x}) = \left(\sqrt{\binom{m}{n_1, \ldots, n_{d+1}}} x_1^{n_1} \cdots x_d^{n_d} \cdot c^{n_{d+1}/2} \right)_{n_i \geq 0, \sum_{i=1}^{d+1} n_i = m}$
  - $\dim(\mathcal{H}) = \binom{d+m}{m}$
- **Exponential kernel:** $k(\mathbf{x}, \mathbf{x}') = \exp(\langle \mathbf{x}, \mathbf{x}' \rangle)$
  - Assume $\mathbf{x} \in \mathbb{R}$ and use Taylor series expansion of $e^{\mathbf{x}}$,

$$\phi(x) = \left[1, x, \sqrt{\frac{1}{2!}} x^2, \sqrt{\frac{1}{3!}} x^3, \ldots \right], \quad \mathcal{H} = \mathbb{R}^\infty$$

Examples of Kernels

- **Linear kernel:** $k(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'$
  - $\phi(\mathbf{x}) = \mathbf{x}$ and $\mathcal{H} = \mathbb{R}$
  - $\phi(\mathbf{x}) = (\mathbf{x}/\sqrt{2}, \mathbf{x}/\sqrt{2})$ and $\mathcal{H} = \mathbb{R}^2$
- **Polynomial kernel:** $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}' + c)^m$ for $c \geq 0$
  - $\phi(\mathbf{x}) = \left(\sqrt{\binom{m}{n_1, \ldots, n_{d+1}}} x_1^{n_1} \cdots x_d^{n_d} \cdot c^{n_{d+1}/2}\right)_{n_i \geq 0, \sum_{i=1}^{d+1} n_i = m}$
  - $\dim(\mathcal{H}) = \binom{d+m}{m}$
- **Exponential kernel:** $k(\mathbf{x}, \mathbf{x}') = \exp(\langle \mathbf{x}, \mathbf{x}' \rangle)$
  - Assume $\mathbf{x} \in \mathbb{R}$ and use Taylor series expansion of $e^{\mathbf{x}}$,

$$\phi(x) = \left[1, x, \sqrt{\frac{1}{2!}} x^2, \sqrt{\frac{1}{3!}} x^3, \ldots\right], \quad \mathcal{H} = \mathbb{R}^\infty$$

- **Gaussian RBF kernel:** $k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|_2^2)$
  - Assume $\mathbf{x} \in \mathbb{R}$ and use Taylor series expansion of $e^{\mathbf{x}}$,

$$\phi(x) = \exp(-\gamma x^2) \left[1, \sqrt{\frac{2\gamma}{1!}} x, \sqrt{\frac{(2\gamma)^2}{2!}} x^2, \sqrt{\frac{(2\gamma)^3}{3!}} x^3, \ldots\right], \quad \mathcal{H} = \mathbb{R}^\infty$$

Building More Complicated Kernels

Let k_1, k_2 be kernels on $\mathcal{X}$. Then, the following are valid kernels.

- $k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}')$.
- $k(\mathbf{x}, \mathbf{x}') = \alpha k_1(\mathbf{x}, \mathbf{x}')$ for $\alpha > 0$.
- $k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') k_2(\mathbf{x}, \mathbf{x}')$.
- $k(\mathbf{x}, \mathbf{x}') = k_1(A(\mathbf{x}), A(\mathbf{x}'))$ for a map $A : \tilde{\mathcal{X}} \to \mathcal{X}$.
- $k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x}) f(\mathbf{x}')$ for a real-valued function f on $\mathcal{X}$.

Building More Complicated Kernels

Let k_1, k_2 be kernels on $\mathcal{X}$. Then, the following are valid kernels.

- $k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}')$.
- $k(\mathbf{x}, \mathbf{x}') = \alpha k_1(\mathbf{x}, \mathbf{x}')$ for $\alpha > 0$.
- $k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}')k_2(\mathbf{x}, \mathbf{x}')$.
- $k(\mathbf{x}, \mathbf{x}') = k_1(A(\mathbf{x}), A(\mathbf{x}'))$ for a map $A : \tilde{\mathcal{X}} \to \mathcal{X}$.
- $k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x})f(\mathbf{x}')$ for a real-valued function f on $\mathcal{X}$.

Question: What property do all kernels have in common?

Positive Definite Kernels

Definition (Positive definiteness)

A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called **positive definite** if, for all $n \in \mathbb{N}$, $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ and all $x_1, \ldots, x_n \in \mathcal{X}$, we have

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_j, x_i) \geq 0. \tag{2}$$

Equivalently, we have that a **Gram** matrix $\mathbf{K}$ is positive definite.

Positive Definite Kernels

Definition (Positive definiteness)

A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called **positive definite** if, for all $n \in \mathbb{N}$, $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ and all $x_1, \ldots, x_n \in \mathcal{X}$, we have

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_j, x_i) \geq 0. \tag{2}$$

Equivalently, we have that a **Gram** matrix $\mathbf{K}$ is positive definite.

Example (Any kernel is positive definite)

Let k be a kernel with feature map $\phi : \mathcal{X} \to \mathcal{H}$, then we have

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_j, x_i) = \left\langle \sum_{i=1}^{n} \alpha_i \phi(x_i), \sum_{j=1}^{n} \alpha_j \phi(x_j) \right\rangle_{\mathcal{H}} \geq 0. \tag{3}$$

Positive definiteness is a **necessary** condition.

Positive Definite Kernels

Positive definiteness is also a **sufficient** condition.

Theorem

A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a kernel if and only if it is symmetric and positive definite.

Positive Definite Kernels

Positive definiteness is also a **sufficient** condition.

Theorem

A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a kernel if and only if it is symmetric and positive definite.

Proof Sketch

1. $\mathcal{H}_{\text{pre}} := \left\{ \sum_{i=1}^{n} \alpha_i k(\cdot, x_i) : n \in \mathbb{N}, \alpha_1, \ldots, \alpha_n \in \mathbb{R}, \mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{X} \right\}$
2. Let $f := \sum_{i=1}^{n} \alpha_i k(\cdot, \mathbf{x}_i)$ and $g = \sum_{j=1}^{m} \beta_j k(\cdot, \mathbf{x}'_j)$
3. Define $\langle f, g \rangle := \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j k(\mathbf{x}'_j, \mathbf{x}_i)$
4. Show that $\langle \cdot, \cdot \rangle$ is an inner product on $\mathcal{H}_{\text{pre}}$.
5. Let $\mathcal{H}$ be a completion of $\mathcal{H}_{\text{pre}}$ and $\mathcal{I} : \mathcal{H}_{\text{pre}} \to \mathcal{H}$ be the corresponding isometric embedding.
6. Then, we have $\langle \mathcal{I}k(\cdot, \mathbf{x}'), \mathcal{I}k(\cdot, \mathbf{x}) \rangle_{\mathcal{H}} = \langle k(\cdot, \mathbf{x}'), k(\cdot, \mathbf{x}) \rangle_{\mathcal{H}_{\text{pre}}} = k(\mathbf{x}, \mathbf{x}')$
7. Hence, $\mathbf{x} \mapsto \mathcal{I}k(\cdot, \mathbf{x})$ defines a feature map of k.

Applications of Kernels

- Kernelized learning algorithms
  - Perceptron
  - Linear discriminant analysis (LDA)
  - Support vector machine (SVM)
  - Gaussian process (GP)
- Non-linear unsupervised learning
  - K-means
  - Principal component analysis (PCA)
  - Independent component analysis (ICA)
  - Canonical correlation analysis (CCA)
- Non-linear statistical methods
  - Maximum mean discrepancy (MMD)
  - Kernel two-sample test
  - Kernel (conditional) independence test
  - Kernel for deep generative models, e.g., MMD-GAN

Reproducing Kernel Hilbert Spaces

Hilbert space

A *Hilbert space* is an **inner product** space that is also a **complete metric space** w.r.t. the norm defined by the inner product.

Reproducing Kernel Hilbert Spaces

Hilbert space

A *Hilbert space* is an **inner product** space that is also a **complete metric space** w.r.t. the norm defined by the inner product.

Reproducing Kernel Hilbert Spaces

Hilbert space

A *Hilbert space* is an **inner product** space that is also a **complete metric space** w.r.t. the norm defined by the inner product.

Example

- Euclidean space $\mathbb{R}^d$ with the usual inner product.
- Sequence space ℓ^2 of $\mathbf{z} := (z_1, z_2, \ldots)$ such that the series $\sum_{n=1}^{\infty} |z_n|^2$ converges.
- The space of square-integrable functions $L_2[a, b]$.

Reproducing Kernel Hilbert Spaces

Definition

Let $\mathcal{H}$ be a Hilbert space of functions mapping from $\mathcal{X}$ into $\mathbb{R}$.

Reproducing Kernel Hilbert Spaces

Definition

Let $\mathcal{H}$ be a Hilbert space of functions mapping from $\mathcal{X}$ into $\mathbb{R}$.

1. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a **reproducing kernel** of $\mathcal{H}$ if we have $k(\cdot, x) \in \mathcal{H}$ for all $x \in \mathcal{X}$ and the **reproducing property**

$$f(x) = \langle f, k(\cdot, x) \rangle \tag{4}$$

holds for all $f \in \mathcal{H}$ and all $x \in \mathcal{X}$.

Reproducing Kernel Hilbert Spaces

Definition

Let $\mathcal{H}$ be a Hilbert space of functions mapping from $\mathcal{X}$ into $\mathbb{R}$.

1. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a **reproducing kernel** of $\mathcal{H}$ if we have $k(\cdot, x) \in \mathcal{H}$ for all $x \in \mathcal{X}$ and the **reproducing property**

$$f(x) = \langle f, k(\cdot, x) \rangle \tag{4}$$

   holds for all $f \in \mathcal{H}$ and all $x \in \mathcal{X}$.

2. The space $\mathcal{H}$ is called a **reproducing kernel Hilbert space (RKHS)** over $\mathcal{X}$ if for all $x \in \mathcal{X}$ the Dirac functional $\delta_x : \mathcal{H} \to \mathbb{R}$ defined by

$$\delta_x(f) := f(x), \qquad f \in \mathcal{H}, \tag{5}$$

   is continuous.

Reproducing Kernel Hilbert Spaces

Definition

Let $\mathcal{H}$ be a Hilbert space of functions mapping from $\mathcal{X}$ into $\mathbb{R}$.

1. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a **reproducing kernel** of $\mathcal{H}$ if we have $k(\cdot, x) \in \mathcal{H}$ for all $x \in \mathcal{X}$ and the **reproducing property**

$$f(x) = \langle f, k(\cdot, x) \rangle \tag{4}$$

   holds for all $f \in \mathcal{H}$ and all $x \in \mathcal{X}$.

2. The space $\mathcal{H}$ is called a **reproducing kernel Hilbert space (RKHS)** over $\mathcal{X}$ if for all $x \in \mathcal{X}$ the Dirac functional $\delta_x : \mathcal{H} \to \mathbb{R}$ defined by

$$\delta_x(f) := f(x), \qquad f \in \mathcal{H}, \tag{5}$$

   is continuous.

Remark: If $\|f_n - f\|_{\mathcal{H}} \to 0$ for $n \to \infty$, then for all $x \in \mathcal{X}$, we have

$$\lim_{n \to \infty} f_n(x) = f(x)$$

.

Reproducing Kernels

Lemma (Reproducing kernels are kernels)

Let $\mathcal{H}$ be a Hilbert space over $\mathcal{X}$ with a reproducing kernel k. Then $\mathcal{H}$ is an RKHS and is also a feature space of k, where the feature map $\phi : \mathcal{X} \to \mathcal{H}$ is given by

$$\phi(x) = k(\cdot, x), \qquad x \in \mathcal{X}. \tag{6}$$

We call ϕ the **canonical feature map**.

Reproducing Kernels

Lemma (Reproducing kernels are kernels)

Let $\mathcal{H}$ be a Hilbert space over $\mathcal{X}$ with a reproducing kernel k. Then $\mathcal{H}$ is an RKHS and is also a feature space of k, where the feature map $\phi : \mathcal{X} \to \mathcal{H}$ is given by

$$\phi(x) = k(\cdot, x), \qquad x \in \mathcal{X}. \tag{6}$$

We call ϕ the **canonical feature map**.

Proof

We fix an $\mathbf{x}' \in \mathcal{X}$ and write $f := k(\cdot, \mathbf{x}')$. Then, for $\mathbf{x} \in \mathcal{X}$, the reproducing property yields

$$\langle \phi(\mathbf{x}'), \phi(\mathbf{x}) \rangle = \langle k(\cdot, \mathbf{x}'), k(\cdot, \mathbf{x}) \rangle = \langle f, k(\cdot, \mathbf{x}) \rangle = f(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}'). \tag{7}$$

Kernels and RKHSs

Theorem (Every RKHS has a unique reproducing kernel)

Let $\mathcal{H}$ be an RKHS over $\mathcal{X}$. Then $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined by

$$k(\mathbf{x}, \mathbf{x}') = \langle \delta_{\mathbf{x}}, \delta_{\mathbf{x}'} \rangle_{\mathcal{H}}, \quad \mathbf{x}, \mathbf{x}' \in \mathcal{X} \tag{8}$$

is the only reproducing kernel of $\mathcal{H}$. Furthermore, if $(e_i)_{i \in I}$ is an orthonormal basis of $\mathcal{H}$, then for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ we have

$$k(\mathbf{x}, \mathbf{x}') = \sum_{i \in I} e_i(\mathbf{x}) \overline{e_i(\mathbf{x}')}. \tag{9}$$

Kernels and RKHSs

Theorem (Every RKHS has a unique reproducing kernel)

Let $\mathcal{H}$ be an RKHS over $\mathcal{X}$. Then $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined by

$$k(\mathbf{x}, \mathbf{x}') = \langle \delta_{\mathbf{x}}, \delta_{\mathbf{x}'} \rangle_{\mathcal{H}}, \quad \mathbf{x}, \mathbf{x}' \in \mathcal{X} \tag{8}$$

is the only reproducing kernel of $\mathcal{H}$. Furthermore, if $(e_i)_{i \in I}$ is an orthonormal basis of $\mathcal{H}$, then for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ we have

$$k(\mathbf{x}, \mathbf{x}') = \sum_{i \in I} e_i(\mathbf{x}) \overline{e_i(\mathbf{x}')}. \tag{9}$$

Conversely, every kernel also has a unique RKHS; see Thm. 4.21 in Steinwart and Christmann (2008).

Universal Kernels

Definition

A continuous kernel k on a compact metric space $\mathcal{X}$ is called **universal** if the RKHS $\mathcal{H}$ of k is dense in $C(\mathcal{X})$, i.e., for every function $g \in C(\mathcal{X})$ and all $\varepsilon > 0$ there exist an $f \in \mathcal{H}$ such that

$$\|f - g\|_\infty \leq \varepsilon.$$

Universal Kernels

Definition

A continuous kernel k on a compact metric space $\mathcal{X}$ is called **universal** if the RKHS $\mathcal{H}$ of k is dense in $C(\mathcal{X})$, i.e., for every function $g \in C(\mathcal{X})$ and all $\varepsilon > 0$ there exist an $f \in \mathcal{H}$ such that

$$\|f - g\|_\infty \leq \varepsilon.$$

Example

Let $\mathcal{X}$ be a compact subset of $\mathbb{R}^d$. Then, the following kernels are universal:

$$\begin{aligned}
\text{Exponential kernel:} \quad & k(\mathbf{x}, \mathbf{x}') := \exp(\langle \mathbf{x}, \mathbf{x}' \rangle) \\
\text{Gaussian RBF kernel:} \quad & k(\mathbf{x}, \mathbf{x}') := \exp(-\gamma^{-2} \|\mathbf{x} - \mathbf{x}'\|_2^2) \\
\text{Binomial kernel:} \quad & k(\mathbf{x}, \mathbf{x}') := (1 - \langle \mathbf{x}, \mathbf{x}' \rangle)^{-\alpha}
\end{aligned}$$

Hilbert Space Embedding of Distributions

Definition

Let $\mathscr{P}$ be a space of all probability measures on a measurable space $(\mathcal{X}, \Sigma)$ and $\mathcal{H}$ an RKHS endowed with a reproducing kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Then, a **kernel mean embedding** is defined by a mapping

$$\mu : \mathscr{P} \to \mathcal{H}, \quad \mathbb{P} \mapsto \int k(\cdot, \mathbf{x}) \, \mathrm{d}\mathbb{P}(\mathbf{x}). \tag{10}$$

Hilbert Space Embedding of Distributions

Definition

Let $\mathscr{P}$ be a space of all probability measures on a measurable space $(\mathcal{X}, \Sigma)$ and $\mathcal{H}$ an RKHS endowed with a reproducing kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Then, a **kernel mean embedding** is defined by a mapping

$$\mu : \mathscr{P} \to \mathcal{H}, \quad \mathbb{P} \mapsto \int k(\cdot, \mathbf{x}) \, \mathrm{d}\mathbb{P}(\mathbf{x}). \tag{10}$$

Remark: For a Dirac measure $\delta_{\mathbf{x}}$, $\delta_{\mathbf{x}} \mapsto \mu[\delta_{\mathbf{x}}] \equiv \mathbf{x} \mapsto k(\cdot, \mathbf{x})$.

Hilbert Space Embedding of Distributions

Definition

Let $\mathscr{P}$ be a space of all probability measures on a measurable space $(\mathcal{X}, \Sigma)$ and $\mathcal{H}$ an RKHS endowed with a reproducing kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Then, a **kernel mean embedding** is defined by a mapping

$$\mu : \mathscr{P} \to \mathcal{H}, \quad \mathbb{P} \mapsto \int k(\cdot, \mathbf{x}) \, \mathrm{d}\mathbb{P}(\mathbf{x}). \tag{10}$$

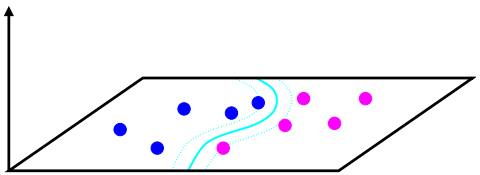
Remark: For a Dirac measure $\delta_{\mathbf{x}}$, $\delta_{\mathbf{x}} \mapsto \mu[\delta_{\mathbf{x}}] \equiv \mathbf{x} \mapsto k(\cdot, \mathbf{x})$.

Lemma

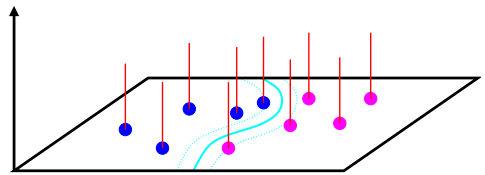
If $\mathbb{E}_{X \sim \mathbb{P}}[\sqrt{k(X, X)}] < \infty$, *then* $\mu_{\mathbb{P}} \in \mathcal{H}$ *and*

$$\mathbb{E}_{X \sim \mathbb{P}}[f(X)] = \langle f, \mu_{\mathbb{P}} \rangle, \quad f \in \mathcal{H}. \tag{11}$$

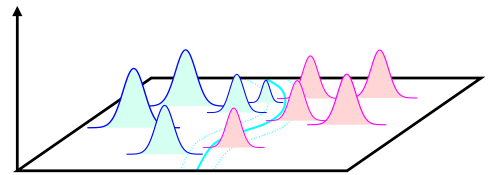
Support Measure Machine (SMM)



$x \mapsto k(\cdot, x)$

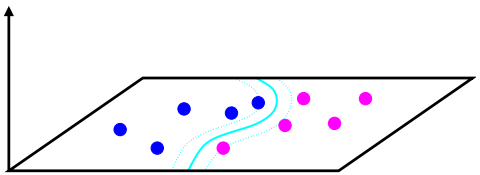
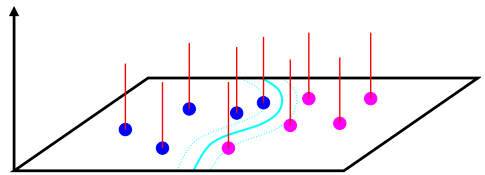
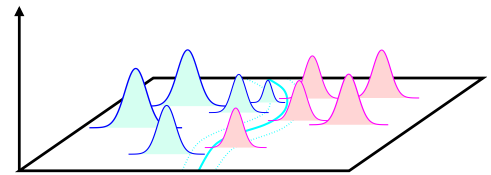


$\delta_x \mapsto \int k(\cdot, z) d\delta_x(z)$



$\mathbb{P} \mapsto \int k(\cdot, z) d\mathbb{P}(z)$

Support Measure Machine (SMM)

$x \mapsto k(\cdot, x)$ $\quad$ $\delta_x \mapsto \int k(\cdot, z) d\delta_x(z)$ $\quad$ $\mathbb{P} \mapsto \int k(\cdot, z) d\mathbb{P}(z)$

Theorem

Under technical assumptions on $\Omega : [0, +\infty) \to \mathbb{R}$*, and a loss function* $\ell : (\mathcal{P} \times \mathbb{R}^2)^m \to \mathbb{R} \cup \{+\infty\}$*, any* $f \in \mathcal{H}$ *minimizing*

$$\ell\left(\mathbb{P}_1, y_1, \mathbb{E}_{\mathbb{P}_1}[f], \ldots, \mathbb{P}_m, y_m, \mathbb{E}_{\mathbb{P}_m}[f]\right) + \Omega\left(\|f\|_{\mathcal{H}}\right)$$

admits a representation of the form

$$f = \sum_{i=1}^{m} \alpha_i \mathbb{E}_{x \sim \mathbb{P}_i}[k(x, \cdot)] = \sum_{i=1}^{m} \alpha_i \mu_{\mathbb{P}_i}.$$

Kernel Discrepancy Measure for Distributions

- Maximum mean discrepancy (MMD)

$$\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}, \mathcal{H}) := \sup_{h \in \mathcal{H}, \|h\| \leq 1} \left| \int h(x) \, \mathrm{d}\mathbb{P}(x) - \int h(x) \, \mathrm{d}\mathbb{Q}(x) \right| \quad (12)$$

Kernel Discrepancy Measure for Distributions

- Maximum mean discrepancy (MMD)

$$\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}, \mathcal{H}) := \sup_{h \in \mathcal{H}, \|h\| \leq 1} \left| \int h(x) \, \mathrm{d}\mathbb{P}(x) - \int h(x) \, \mathrm{d}\mathbb{Q}(x) \right| \quad (12)$$

- MMD is an **integral probability metric (IPM)** and corresponds to the RKHS distance between mean embeddings.

$$\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}, \mathcal{H}) = \|\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}\|_{\mathcal{H}}^2. \quad (13)$$

Kernel Discrepancy Measure for Distributions

- Maximum mean discrepancy (MMD)

$$\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}, \mathcal{H}) := \sup_{h \in \mathcal{H}, \|h\| \leq 1} \left| \int h(x) \, \mathrm{d}\mathbb{P}(x) - \int h(x) \, \mathrm{d}\mathbb{Q}(x) \right| \quad (12)$$

- MMD is an **integral probability metric (IPM)** and corresponds to the RKHS distance between mean embeddings.

$$\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}, \mathcal{H}) = \|\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}\|_{\mathcal{H}}^2. \quad (13)$$

- If k is **universal**, then $\|\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}\|_{\mathcal{H}} = 0$ if and only if $\mathbb{P} = \mathbb{Q}$.

Kernel Discrepancy Measure for Distributions

- Maximum mean discrepancy (MMD)

$$\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}, \mathcal{H}) := \sup_{h \in \mathcal{H}, \|h\| \leq 1} \left| \int h(x) \, \mathrm{d}\mathbb{P}(x) - \int h(x) \, \mathrm{d}\mathbb{Q}(x) \right| \quad (12)$$

- MMD is an **integral probability metric (IPM)** and corresponds to the RKHS distance between mean embeddings.

$$\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}, \mathcal{H}) = \|\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}\|_{\mathcal{H}}^2. \quad (13)$$

- If k is **universal**, then $\|\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}\|_{\mathcal{H}} = 0$ if and only if $\mathbb{P} = \mathbb{Q}$.
- Given $\{\mathbf{x}_i\}_{i=1}^n \sim \mathbb{P}$ and $\{\mathbf{y}_j\}_{j=1}^m \sim \mathbb{Q}$, the empirical MMD is

$$\widehat{\mathrm{MMD}}_u^2(\mathbb{P}, \mathbb{Q}, \mathcal{H}) = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j \neq i}^n k(\mathbf{x}_i, \mathbf{x}_j) + \frac{1}{m(m-1)} \sum_{i=1}^m \sum_{j \neq i}^m k(\mathbf{y}_i, \mathbf{y}_j) - \frac{2}{nm} \sum_{i=1}^n \sum_{j=1}^m k(\mathbf{x}_i, \mathbf{y}_j).$$

Future Directions

- Kernel choice problem
- Kernel methods in high-dimensional spaces
- Kernel-based statistical methods
- Kernels and deep learning
- Scalable kernel machines
- Kernel methods in causality

Q & A