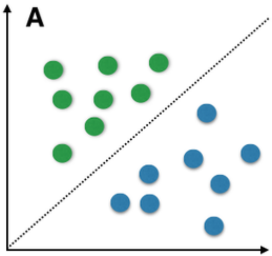
Mini-course on Kernel Techniques

Zoltán Szabó

Faculty of Engineering,
Free University of Bozen-Bolzano

Feb. 23, 2026

Idea of featurization: in classification



Decision surface:

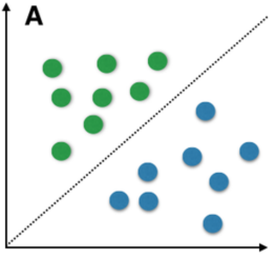
$$\{\mathbf{x} : \langle \boldsymbol{\beta}, \mathbf{x} \rangle = 0\} \Rightarrow$$

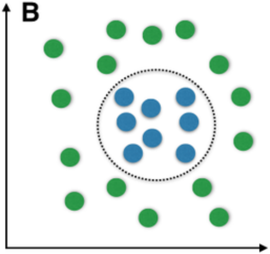
classes:

$$\{\mathbf{x} : \langle \boldsymbol{\beta}, \mathbf{x} \rangle \geq 0\} \qquad \{\mathbf{x} : \langle \boldsymbol{\beta}, \mathbf{x} \rangle < 0\} .$$

Idea of featurization: in classification

Idealized situation (Stylistic) real world





Decision surface (left):

$$\{\mathbf{x} : \langle \boldsymbol{\beta}, \mathbf{x} \rangle = 0\} \Rightarrow$$

classes:

$$\{\mathbf{x} : \langle \boldsymbol{\beta}, \mathbf{x} \rangle \geq 0\} \qquad \{\mathbf{x} : \langle \boldsymbol{\beta}, \mathbf{x} \rangle < 0\} \,.$$

Featurization – continued

On the ellipse

$$\left\{\mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} = 1\right\}$$

Featurization – continued

On the ellipse, outside

$$\left\{ \mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} = 1 \right\},$$

$$\left\{ \mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} > 1 \right\}$$

Featurization – continued

On the ellipse, outside, inside:

$$\left\{ \mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} = 1 \right\},$$

$$\left\{ \mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} > 1 \right\},$$

$$\left\{ \mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} < 1 \right\}.$$

Featurization – continued

On the ellipse, outside, inside:

$$\left\{\mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} = 1\right\},$$
$$\left\{\mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} > 1\right\},$$
$$\left\{\mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} < 1\right\}.$$

With polynomial feature: $\boxed{\varphi(\mathbf{x}) = (x_1^2, x_1, 1, x_2^2, x_2)}$:

- Decision surface: $\{\mathbf{x} : \langle \beta, \varphi(\mathbf{x}) \rangle = 0\}$.

Featurization – continued

On the ellipse, outside, inside:

$$\left\{\mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} = 1\right\},$$

$$\left\{\mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} > 1\right\},$$

$$\left\{\mathbf{x} : \frac{(x_1 - c_1)^2}{a^2} + \frac{(x_2 - c_2)^2}{b^2} < 1\right\}.$$

With polynomial feature: $\boxed{\varphi(\mathbf{x}) = (x_1^2, x_1, 1, x_2^2, x_2)}$:

- Decision surface: $\{\mathbf{x} : \langle \beta, \varphi(\mathbf{x})\rangle = 0\}$.
- Classes: $\{\mathbf{x} : \langle \beta, \varphi(\mathbf{x})\rangle > 0\}$, $\{\mathbf{x} : \langle \beta, \varphi(\mathbf{x})\rangle < 0\}$.

Quadratic & polynomial features

Still in $\mathbb{R}^2$:

$$\varphi(\mathbf{x}) = \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right),$$

$$\boxed{\langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle} = ?$$

Quadratic & polynomial features

Still in $\mathbb{R}^2$:

$$\varphi(\mathbf{x}) = \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right),$$

$$\boxed{\langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle} = \left\langle \begin{bmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{bmatrix}, \begin{bmatrix} (x_1')^2 \\ \sqrt{2}(x_1')(x_2') \\ (x_2')^2 \end{bmatrix} \right\rangle$$

Quadratic & polynomial features

Still in $\mathbb{R}^2$:

$$\varphi(\mathbf{x}) = \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right),$$

$$\boxed{\langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle} = \left\langle \begin{bmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{bmatrix}, \begin{bmatrix} (x_1')^2 \\ \sqrt{2}(x_1')(x_2') \\ (x_2')^2 \end{bmatrix} \right\rangle$$

$$= x_1^2(x_1')^2 + \underbrace{\sqrt{2}\sqrt{2}}_{2} x_1x_2(x_1')(x_2') + x_2^2(x_2')^2$$

Quadratic & polynomial features

Still in $\mathbb{R}^2$:

$$
\begin{aligned}
\varphi(\mathbf{x}) &= \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right), \\
\boxed{\langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle} &= \left\langle \begin{bmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{bmatrix}, \begin{bmatrix} (x_1')^2 \\ \sqrt{2}(x_1')(x_2') \\ (x_2')^2 \end{bmatrix} \right\rangle \\
&= x_1^2(x_1')^2 + \underbrace{\sqrt{2}\sqrt{2}}_{2}x_1x_2(x_1')(x_2') + x_2^2(x_2')^2 \\
&= \left(x_1x_1' + x_2x_2'\right)^2
\end{aligned}
$$

Quadratic & polynomial features

Still in $\mathbb{R}^2$:

$$\varphi(\mathbf{x}) = \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right),$$

$$\boxed{\langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle} = \left\langle \begin{bmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{bmatrix}, \begin{bmatrix} (x_1')^2 \\ \sqrt{2}(x_1')(x_2') \\ (x_2')^2 \end{bmatrix} \right\rangle$$

$$= x_1^2(x_1')^2 + \underbrace{\sqrt{2}\sqrt{2}}_{2} x_1x_2(x_1')(x_2') + x_2^2(x_2')^2$$

$$= \left(x_1x_1' + x_2x_2'\right)^2$$

$$= \left\langle \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \begin{bmatrix} x_1' \\ x_2' \end{bmatrix} \right\rangle^2 = \boxed{\langle \mathbf{x}, \mathbf{x}' \rangle^2} =: k(\mathbf{x}, \mathbf{x}').$$

Quadratic & polynomial features

Still in $\mathbb{R}^2$:

$$\begin{aligned}
\varphi(\mathbf{x}) &= \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right), \\
\boxed{\langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle} &= \left\langle \begin{bmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{bmatrix}, \begin{bmatrix} (x_1')^2 \\ \sqrt{2}(x_1')(x_2') \\ (x_2')^2 \end{bmatrix} \right\rangle \\
&= x_1^2(x_1')^2 + \underbrace{\sqrt{2}\sqrt{2}}_{2} x_1x_2(x_1')(x_2') + x_2^2(x_2')^2 \\
&= \left(x_1x_1' + x_2x_2'\right)^2 \\
&= \left\langle \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \begin{bmatrix} x_1' \\ x_2' \end{bmatrix} \right\rangle^2 = \boxed{\langle \mathbf{x}, \mathbf{x}' \rangle^2} =: k(\mathbf{x}, \mathbf{x}').
\end{aligned}$$

$\langle \mathbf{x}, \mathbf{x}' \rangle^d = \langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle$: $\varphi(\mathbf{x}) =$ d-order polynomial. $\Rightarrow$

Quadratic & polynomial features

Still in $\mathbb{R}^2$:

$$\varphi(\mathbf{x}) = \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right),$$

$$\boxed{\langle\varphi(\mathbf{x}), \varphi(\mathbf{x}')\rangle} = \left\langle \begin{bmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{bmatrix}, \begin{bmatrix} (x_1')^2 \\ \sqrt{2}(x_1')(x_2') \\ (x_2')^2 \end{bmatrix} \right\rangle$$

$$= x_1^2(x_1')^2 + \underbrace{\sqrt{2}\sqrt{2}}_{2} x_1x_2(x_1')(x_2') + x_2^2(x_2')^2$$

$$= \left(x_1x_1' + x_2x_2'\right)^2$$

$$= \left\langle \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \begin{bmatrix} x_1' \\ x_2' \end{bmatrix} \right\rangle^2 = \boxed{\langle\mathbf{x}, \mathbf{x}'\rangle^2} =: k(\mathbf{x}, \mathbf{x}').$$

$\langle\mathbf{x}, \mathbf{x}'\rangle^d = \langle\varphi(\mathbf{x}), \varphi(\mathbf{x}')\rangle$: $\varphi(\mathbf{x}) =$ d-order polynomial. $\Rightarrow$ Explicit computation would be computationally intense! $\boxed{\varphi = ?}$

Why kernels/RKHSs?

Key idea

$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}.$

1. flexibility & computational tractability

Why kernels/RKHSs?

Key idea

$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}.$

1. flexibility & computational tractability,
2. applicable on a wide variety of domains

Why kernels/RKHSs?

Key idea

$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}$.

1. flexibility & computational tractability,
2. applicable on a wide variety of domains,
3. RKHS $\mathcal{H}_k$: Hilbert structure $\Rightarrow$ statistical analysis

Why kernels/RKHSs?

Key idea

$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}.$

1. flexibility & computational tractability,
2. applicable on a wide variety of domains,
3. RKHS $\mathcal{H}_k$: Hilbert structure $\Rightarrow$ statistical analysis
4. supervised learning: classification, regression $\Rightarrow$ today

Why kernels/RKHSs?

Key idea

$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}$.

1. flexibility & computational tractability,
2. applicable on a wide variety of domains,
3. RKHS $\mathcal{H}_k$: Hilbert structure $\Rightarrow$ statistical analysis,
4. supervised learning: classification, regression $\Rightarrow$ today,
   - $\mathcal{H}_k \subset \mathbb{R}^{\mathcal{X}}$: extension to vector-valued setting $\exists$ ($\mathcal{H}_k \subset \mathcal{H}^{\mathcal{X}}$, $k(x, x') \in \mathcal{L}(\mathcal{H})$; dependency of output coordinates).

Why kernels/RKHSs?

Key idea

$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}$.

1. flexibility & computational tractability,
2. applicable on a wide variety of domains,
3. RKHS $\mathcal{H}_k$: Hilbert structure $\Rightarrow$ statistical analysis,
4. supervised learning: classification, regression $\Rightarrow$ today,
    - $\mathcal{H}_k \subset \mathbb{R}^{\mathcal{X}}$: extension to vector-valued setting $\exists$ ($\mathcal{H}_k \subset \mathcal{H}^{\mathcal{X}}$, $k(x, x') \in \mathcal{L}(\mathcal{H})$; dependency of output coordinates).
5. goes far beyond supervised learning (dimensionality reduction: KPCA, KCCA; information theoretical estimators: MMD, HSIC, KSD, ...) $\Rightarrow$ latter (KSD): Wednesday.

Content: supervised learning

- Classification (SVMC):
  1. linear separability $\Rightarrow$ hyperplane, margin

Content: supervised learning

- Classification (SVMC):
  1. linear separability $\Rightarrow$ hyperplane, margin,
  2. linear close-to separability $\Rightarrow$ slack variables (constraint violation), duality.

Content: supervised learning

- Classification (SVMC):
  1. linear separability $\Rightarrow$ hyperplane, margin,
  2. linear close-to separability $\Rightarrow$ slack variables (constraint violation), duality.
  3. non-linear close-to separability:
     - linearity after featurization $\Rightarrow$ kernel, RKHS, representer theorem.

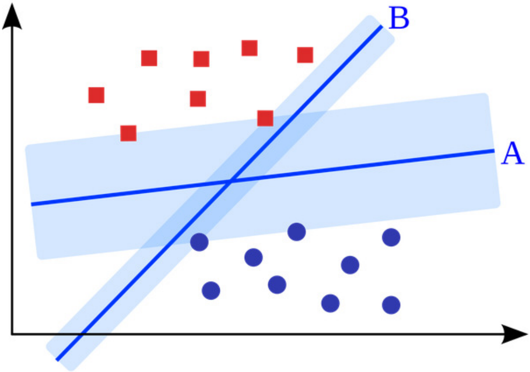
Content: supervised learning

- Classification (SVMC):
  1. linear separability $\Rightarrow$ hyperplane, margin,
  2. linear close-to separability $\Rightarrow$ slack variables (constraint violation), duality.
  3. non-linear close-to separability:
     - linearity after featurization $\Rightarrow$ kernel, RKHS, representer theorem.
- Regression (SVMR $\xrightarrow{\text{spec.}}$ kernel ridge regression):
  - quadratic cost + quadratic regularization,
  - an other application of the representer theorem.

(Towards) linear SVMC

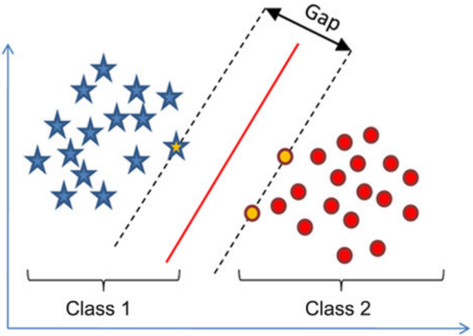
Idea of SVMC

- Task: binary classification.
- Given: training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^n \in \mathbb{R}^p \times \{-1, 1\}$, linearly separable.
- Question: Which separating line is the 'best'?



SVMC

Answer/intuition: the one with the largest margin.



Needed

hyperplane, margin

Hyperplanes

- Hyperplane going through the origin: with normal vector $\beta \in \mathbb{R}^p$

$$H_{\beta} := \{\mathbf{x} \in \mathbb{R}^p \ : \ \langle \beta, \mathbf{x} \rangle = 0\}.$$

Hyperplanes

- Hyperplane going through the origin: with normal vector $\boldsymbol{\beta} \in \mathbb{R}^p$

$$H_{\boldsymbol{\beta}} := \{\mathbf{x} \in \mathbb{R}^p \; : \; \langle \boldsymbol{\beta}, \mathbf{x} \rangle = 0\}.$$

- Hyperplane: with normal vector $\boldsymbol{\beta} \in \mathbb{R}^p$ and offset $\beta_0 \in \mathbb{R}$

$$H_{\boldsymbol{\beta},\beta_0} := \{\mathbf{x} \in \mathbb{R}^p \; : \; \langle \boldsymbol{\beta}, \mathbf{x} \rangle + \beta_0 = 0\}.$$

Hyperplanes

- Hyperplane going through the origin: with normal vector $\beta \in \mathbb{R}^p$

$$H_{\beta} := \{\mathbf{x} \in \mathbb{R}^p \ : \ \langle \beta, \mathbf{x} \rangle = 0\}.$$

- Hyperplane: with normal vector $\beta \in \mathbb{R}^p$ and offset $\beta_0 \in \mathbb{R}$

$$H_{\beta,\beta_0} := \{\mathbf{x} \in \mathbb{R}^p \ : \ \langle \beta, \mathbf{x} \rangle + \beta_0 = 0\}.$$

- The 2 sides of the hyperplane:

$$H^{+}_{\beta,\beta_0} := \{\mathbf{x} \in \mathbb{R}^p \ : \ \langle \beta, \mathbf{x} \rangle + \beta_0 > 0\},$$
$$H^{-}_{\beta,\beta_0} := \{\mathbf{x} \in \mathbb{R}^p \ : \ \langle \beta, \mathbf{x} \rangle + \beta_0 < 0\}.$$

We will use H_{β,β_0}

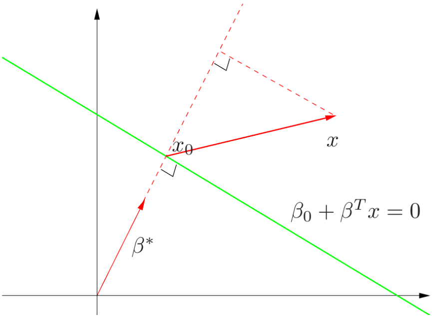
Hyperplane: properties

Recall:

$$H_{\boldsymbol{\beta},\beta_0} := \{\mathbf{x} \in \mathbb{R}^p \ : \ \langle \boldsymbol{\beta}, \mathbf{x} \rangle + \beta_0 = 0\}.$$

Properties:

1. For any point $\mathbf{x}_0 \in H_{\boldsymbol{\beta},\beta_0}$: $\langle \boldsymbol{\beta}, \mathbf{x}_0 \rangle = -\beta_0$ (by def).



Hyperplane: properties

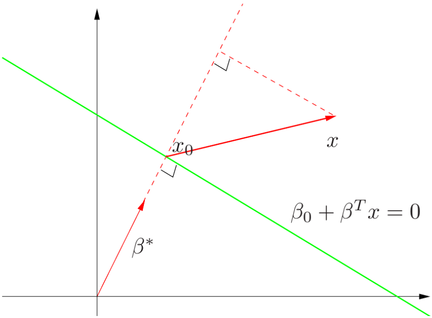
Recall:

$$H_{\beta,\beta_0} := \{\mathbf{x} \in \mathbb{R}^p \ : \ \langle \beta, \mathbf{x} \rangle + \beta_0 = 0\}.$$

Properties:

1. For any point $\mathbf{x}_0 \in H_{\beta,\beta_0}$: $\langle \beta, \mathbf{x}_0 \rangle = -\beta_0$ (by def).
2. Signed distance of any $\mathbf{x} \in \mathbb{R}^p$ from H_{β,β_0}: with $\beta^* = \frac{\beta}{\|\beta\|_2}$

$$\langle \beta^*, \mathbf{x} - \mathbf{x}_0 \rangle = \left\langle \frac{\beta}{\|\beta\|_2}, \mathbf{x} \right\rangle - \left\langle \frac{\beta}{\|\beta\|_2}, \mathbf{x}_0 \right\rangle$$



Hyperplane: properties

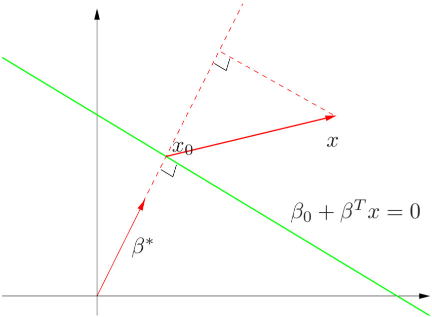
Recall:

$$H_{\beta,\beta_0} := \{\mathbf{x} \in \mathbb{R}^p \ : \ \langle \beta, \mathbf{x} \rangle + \beta_0 = 0\}.$$

Properties:

1. For any point $\mathbf{x}_0 \in H_{\beta,\beta_0}$: $\langle \beta, \mathbf{x}_0 \rangle = -\beta_0$ (by def).
2. Signed distance of any $\mathbf{x} \in \mathbb{R}^p$ from H_{β,β_0}: with $\beta^* = \frac{\beta}{\|\beta\|_2}$

$$\langle \beta^*, \mathbf{x} - \mathbf{x}_0 \rangle = \left\langle \frac{\beta}{\|\beta\|_2}, \mathbf{x} \right\rangle - \left\langle \frac{\beta}{\|\beta\|_2}, \mathbf{x}_0 \right\rangle \overset{①}{=} \frac{\langle \beta, \mathbf{x} \rangle + \beta_0}{\|\beta\|_2}.$$



Optimization problem for max-margin hyperplane

Wanted

$y_i = +1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 > 0$; $y_i = -1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 < 0$.

Optimization problem for max-margin hyperplane

Wanted

$y_i = +1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 > 0$; $y_i = -1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 < 0$.

$$\max_{\boldsymbol{\beta} \in \mathbb{R}^p, \beta_0 \in \mathbb{R}} M, \text{ s.t. } \underbrace{y_i \left(\frac{\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0}{\|\boldsymbol{\beta}\|_2} \right)}_{\text{classify correctly } (\mathbf{x}_i, y_i)} \geq \overbrace{M}^{\text{with } \geq M \text{ distance away from } H_{\boldsymbol{\beta}, \beta_0}}, \forall i \in [n].$$

Optimization problem for max-margin hyperplane

Wanted

$y_i = +1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 > 0$; $y_i = -1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 < 0$.

$$\max_{\boldsymbol{\beta} \in \mathbb{R}^p, \beta_0 \in \mathbb{R}} M, \text{ s.t. } \underbrace{y_i \left(\frac{\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0}{\|\boldsymbol{\beta}\|_2} \right)}_{\text{classify correctly } (\mathbf{x}_i, y_i)} \geq \overbrace{M}^{\text{with } \geq M \text{ distance away from } H_{\boldsymbol{\beta}, \beta_0}}, \forall i \in [n].$$

- $(\boldsymbol{\beta}, \beta_0)$: constraints $\checkmark \Rightarrow c(\boldsymbol{\beta}, \beta_0)$: constraints $\checkmark$ for any $c > 0 \Rightarrow$

Optimization problem for max-margin hyperplane

Wanted

$y_i = +1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 > 0;\ y_i = -1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 < 0.$

$$\max_{\boldsymbol{\beta}\in\mathbb{R}^p, \beta_0\in\mathbb{R}} M, \text{ s.t. } \underbrace{y_i \left(\frac{\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0}{\|\boldsymbol{\beta}\|_2} \right)}_{\text{classify correctly } (\mathbf{x}_i, y_i)} \geq \overbrace{M}^{\text{with } \geq M \text{ distance away from } H_{\boldsymbol{\beta},\beta_0}}, \forall i \in [n].$$

- $(\boldsymbol{\beta}, \beta_0)$: constraints $\checkmark \Rightarrow c(\boldsymbol{\beta}, \beta_0)$: constraints $\checkmark$ for any $c > 0 \Rightarrow$
- We set $\|\boldsymbol{\beta}\|_2 = \frac{1}{M}$ ($\Rightarrow M = \frac{1}{\|\boldsymbol{\beta}\|_2}$), and get

$$\max_{\boldsymbol{\beta}\in\mathbb{R}^p, \beta_0\in\mathbb{R}} \frac{1}{\|\boldsymbol{\beta}\|_2} \text{ s.t. } y_i \left(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 \right) \geq 1,\ \forall i \in [n] \Leftrightarrow$$

Optimization problem for max-margin hyperplane

Wanted

$y_i = +1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 > 0; \; y_i = -1 \Rightarrow \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 < 0.$

$$\max_{\boldsymbol{\beta}\in\mathbb{R}^p, \beta_0\in\mathbb{R}} M, \text{ s.t. } \underbrace{y_i \left(\frac{\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0}{\|\boldsymbol{\beta}\|_2} \right)}_{\text{classify correctly } (\mathbf{x}_i, y_i)} \geq \overbrace{M}^{\text{with } \geq M \text{ distance away from } H_{\boldsymbol{\beta},\beta_0}}, \forall i \in [n].$$

- $(\boldsymbol{\beta}, \beta_0)$: constraints $\checkmark \Rightarrow c(\boldsymbol{\beta}, \beta_0)$: constraints $\checkmark$ for any $c > 0 \Rightarrow$
- We set $\|\boldsymbol{\beta}\|_2 = \frac{1}{M}$ ($\Rightarrow M = \frac{1}{\|\boldsymbol{\beta}\|_2}$), and get

$$\max_{\boldsymbol{\beta}\in\mathbb{R}^p, \beta_0\in\mathbb{R}} \frac{1}{\|\boldsymbol{\beta}\|_2} \text{ s.t. } y_i \left(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 \right) \geq 1, \; \forall i \in [n] \Leftrightarrow$$

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p, \beta_0\in\mathbb{R}} \frac{1}{2} \|\boldsymbol{\beta}\|_2^2 \text{ s.t. } y_i \left(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 \right) \geq 1, \; \forall i \in [n].$$

SVMC: hard vs soft

- Hard classification :
  - decision: $\hat{y}(\mathbf{x}) = \text{sign}(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + \beta_0)$.
  - objective:

$$\min_{\boldsymbol{\beta}, \beta_0} \frac{1}{2} \|\boldsymbol{\beta}\|_2^2 \ \text{ s.t. } \ y_i \left(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 \right) \geq 1, \forall i \in [n].$$

There might not be solution! (non-linearly separable case)

SVMC: hard vs soft

- Hard classification :
  - decision: $\hat{y}(\mathbf{x}) = \text{sign}(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + \beta_0)$.
  - objective:

$$\min_{\boldsymbol{\beta}, \beta_0} \frac{1}{2} \|\boldsymbol{\beta}\|_2^2 \;\text{ s.t. }\; y_i \left(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 \right) \geq 1, \forall i \in [n].$$

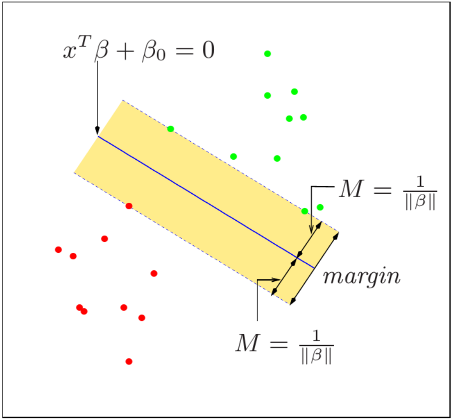
  There might not be solution! (non-linearly separable case)

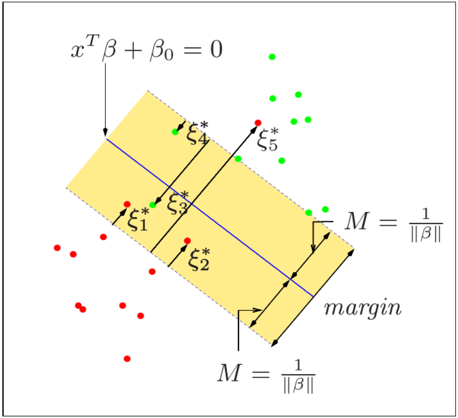
- Soft classification objective ($C > 0$):

$$\min_{\boldsymbol{\beta}, \beta_0, \boldsymbol{\xi}} \frac{1}{2} \|\boldsymbol{\beta}\|_2^2 + C \sum_{i=1}^{n} \xi_i \;\text{ s.t. }\; y_i \left(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 \right) \geq 1 - \overbrace{\xi_i ,}^{\text{slack variables}} \xi_i \geq 0, \; \forall i \in [n].$$

  Linear penalty on misclassification.

Hard vs soft SVMC: visual illustration ($\|\cdot\| := \|\cdot\|_2$)





Note on the objective of soft SVMC

$$\min_{\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^{n}\xi_i \ \text{ s.t. } \ y_i\left(\langle\boldsymbol{\beta},\mathbf{x}_i\rangle + \beta_0\right) \geq 1 - \xi_i,\ \xi_i \geq 0,\ \forall i \in [n]$$

Note on the objective of soft SVMC

$$\min_{\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^{n}\xi_i \text{ s.t. } y_i\left(\langle\boldsymbol{\beta},\mathbf{x}_i\rangle + \beta_0\right) \geq 1-\xi_i,\ \xi_i \geq 0,\ \forall i \in [n]. \Leftrightarrow$$

$$\min_{\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^{n}\max\left(1 - y_i\left(\langle\boldsymbol{\beta},\mathbf{x}_i\rangle + \beta_0\right), 0\right)$$

Note on the objective of soft SVMC

$$\min_{\beta,\beta_0,\xi} \frac{1}{2}\|\beta\|_2^2 + C\sum_{i=1}^{n}\xi_i \text{ s.t. } y_i\left(\langle\beta,\mathbf{x}_i\rangle+\beta_0\right) \geq 1-\xi_i,\ \xi_i \geq 0,\ \forall i \in [n]. \Leftrightarrow$$

$$\min_{\beta,\beta_0,\xi} \frac{1}{2}\|\beta\|_2^2 + C\sum_{i=1}^{n}\underbrace{\max\left(1-y_i(\underbrace{\langle\beta,\mathbf{x}_i\rangle+\beta_0}_{=f(\mathbf{x}_i)}),0\right)}_{=:h(y_i f(\mathbf{x}_i))},$$

where $h(u) = \max(1-u, 0)$ is the hinge loss.

Note on the objective of soft SVMC – continued

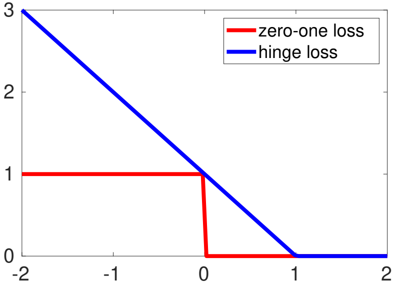
The hinge loss is the convex envelope of the zero-one loss:

$$z(u) = I_{\mathbb{R}_{<0}}(u), \qquad u = y_i f(x_i),$$
$$h(u) = \max(1 - u, 0).$$

Note on the objective of soft SVMC – continued

The hinge loss is the convex envelope of the zero-one loss:

$$z(u) = I_{\mathbb{R}_{<0}}(u), \qquad\qquad u = y_i f(x_i),$$
$$h(u) = \max(1-u, 0).$$



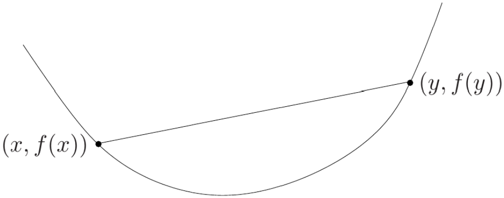
Computation

Convex optimization, duality.

Convex function

- A function $f : \mathbb{R}^d \to \mathbb{R}$ is called convex if
  1. $\mathrm{dom}(f) \coloneqq \{\mathbf{x} \in \mathbb{R}^d : f(\mathbf{x}) \text{ is defined}\}$ is convex, and
  2. for all $\mathbf{x}, \mathbf{y} \in \mathrm{dom}(f)$ and $\alpha \in [0, 1]$

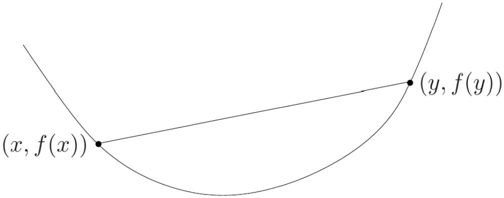
$$f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y}).$$



Convex and concave functions

- A function $f : \mathbb{R}^d \to \mathbb{R}$ is called convex if
  1. $\mathrm{dom}(f) := \{\mathbf{x} \in \mathbb{R}^d : f(\mathbf{x}) \text{ is defined}\}$ is convex, and
  2. for all $\mathbf{x}, \mathbf{y} \in \mathrm{dom}(f)$ and $\alpha \in [0, 1]$

$$f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y}).$$

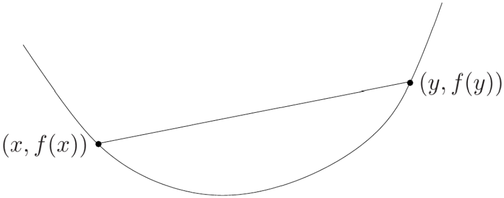


- f is called concave if $-f$ is convex.

Convex and concave functions, affine ones

- A function $f : \mathbb{R}^d \to \mathbb{R}$ is called convex if
  1. $\mathrm{dom}(f) \coloneqq \{\mathbf{x} \in \mathbb{R}^d : f(\mathbf{x}) \text{ is defined}\}$ is convex, and
  2. for all $\mathbf{x}, \mathbf{y} \in \mathrm{dom}(f)$ and $\alpha \in [0, 1]$

$$f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y}).$$



- f is called concave if $-f$ is convex.
- Affine functions ($\mathbf{x} \mapsto \langle \mathbf{w}, \mathbf{x} \rangle + b$) are both convex and concave.

Optimization

- Task:

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^d} \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_i(\mathbf{x}) \leq 0, \; i \in [m], \\ & h_j(\mathbf{x}) = 0, \; j \in [p]. \end{aligned}$$

$p^* :=$ optimal value of this problem. Assume: $\mathcal{D} := \left(\cap_{i=0}^m \operatorname{dom}(f_i)\right) \cap \left(\cap_{j \in [p]} \operatorname{dom}(h_j)\right) \neq \emptyset$.

Optimization

- Task:

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^d} \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_i(\mathbf{x}) \leq 0,\ i \in [m], \\ & h_j(\mathbf{x}) = 0,\ j \in [p]. \end{aligned}$$

  $p^* :=$ optimal value of this problem. Assume:
  $\mathcal{D} := \left(\cap_{i=0}^{m} \operatorname{dom}(f_i)\right) \cap \left(\cap_{j\in[p]} \operatorname{dom}(h_j)\right) \neq \emptyset$.

- Lagrangian function: $\operatorname{dom}(L) = \mathcal{D} \times \mathbb{R}^m \times \mathbb{R}^p$

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\mathbf{x}) + \sum_{i\in[m]} \lambda_i f_i(\mathbf{x}) + \sum_{j\in[p]} \nu_j h_j(\mathbf{x}).$$

Optimization

- Task:

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^d} \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_i(\mathbf{x}) \le 0,\ i \in [m], \\ & h_j(\mathbf{x}) = 0,\ j \in [p]. \end{aligned}$$

  $p^* :=$ optimal value of this problem. Assume:
  $\mathcal{D} := \left(\cap_{i=0}^m \mathrm{dom}(f_i)\right) \cap \left(\cap_{j\in[p]} \mathrm{dom}(h_j)\right) \neq \emptyset$.

- Lagrangian function: $\mathrm{dom}(L) = \mathcal{D} \times \mathbb{R}^m \times \mathbb{R}^p$

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\mathbf{x}) + \sum_{i\in[m]} \lambda_i f_i(\mathbf{x}) + \sum_{j\in[p]} \nu_j h_j(\mathbf{x}).$$

  dual variables: $\boldsymbol{\lambda} = (\lambda_i)_{i=1}^m \in \mathbb{R}^m$, $\boldsymbol{\nu} = (\nu_j)_{j=1}^p \in \mathbb{R}^p$.

Lagrange dual function

Consider for all $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\nu} \in \mathbb{R}^p$, the 'minimum' of the Lagrangian:

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) := \inf_{\mathbf{x} \in \mathcal{D}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x} \in \mathcal{D}} \left(f_0(\mathbf{x}) + \sum_{i \in [m]} \lambda_i f_i(\mathbf{x}) + \sum_{j \in [p]} \nu_j h_j(\mathbf{x}) \right).$$

Properties:

- g is concave [as a pointwise inf of affine functions of $(\boldsymbol{\lambda}, \boldsymbol{\nu})$]

Lagrange dual function

Consider for all $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\nu} \in \mathbb{R}^p$, the 'minimum' of the Lagrangian:

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) := \inf_{\mathbf{x} \in \mathcal{D}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x} \in \mathcal{D}} \left(f_0(\mathbf{x}) + \sum_{i \in [m]} \lambda_i f_i(\mathbf{x}) + \sum_{j \in [p]} \nu_j h_j(\mathbf{x}) \right).$$

Properties:

- g is concave [as a pointwise inf of affine functions of $(\boldsymbol{\lambda}, \boldsymbol{\nu})$],
- it gives a lower bound with $\boldsymbol{\lambda} \geq \mathbf{0}_m$ and any $\boldsymbol{\nu} \in \mathbb{R}^p$: $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \leq p^*$.

Lagrange dual function, Lagrange dual problem

Consider for all $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\nu} \in \mathbb{R}^p$, the 'minimum' of the Lagrangian:

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) := \inf_{\mathbf{x} \in \mathcal{D}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x} \in \mathcal{D}} \left(f_0(\mathbf{x}) + \sum_{i \in [m]} \lambda_i f_i(\mathbf{x}) + \sum_{j \in [p]} \nu_j h_j(\mathbf{x}) \right).$$

Properties:

- g is concave [as a pointwise inf of affine functions of $(\boldsymbol{\lambda}, \boldsymbol{\nu})$],
- it gives a lower bound with $\boldsymbol{\lambda} \geq \mathbf{0}_m$ and any $\boldsymbol{\nu} \in \mathbb{R}^p$: $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \leq p^*$.
- Lagrange dual problem: best lower bound with the Lagrange function

$$\begin{aligned} \max_{\substack{\boldsymbol{\lambda} \in \mathbb{R}^m, \\ \boldsymbol{\nu} \in \mathbb{R}^p}} \quad & g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \\ \text{s.t.} \quad & \boldsymbol{\lambda} \geq \mathbf{0}_m. \end{aligned}$$

Dual optimal: $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$; optimal value d^*. Above $\Rightarrow$ $d^* \leq p^*$.

KKT conditions for convex problems: optimality

- Assume: $(f_i)_{i=0}^m$ are convex and differentiable, and $(h_j)_{j=1}^p$ are affine.

KKT conditions for convex problems: optimality

- Assume: $(f_i)_{i=0}^m$ are convex and differentiable, and $(h_j)_{j=1}^p$ are affine.
- Let $\left(\tilde{\mathbf{x}}, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}}\right)$ satisfy the KKT conditions:

$$f_i(\tilde{\mathbf{x}}) \leq 0,\ \forall i \in [m], \tag{1}$$

$$h_j(\tilde{\mathbf{x}}) = 0,\ \forall j \in [p], \tag{2}$$

$$\tilde{\lambda}_i \geq 0,\ \forall i \in [m], \tag{3}$$

$$\tilde{\lambda}_i f_i(\tilde{\mathbf{x}}) = 0,\ \forall i \in [m], \tag{4}$$

$$\left.\frac{\partial L(\mathbf{x}, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\tilde{\mathbf{x}}} = \mathbf{0}, \tag{5}$$

i.e., (1)-(2): $\tilde{\mathbf{x}}$ is primal feasible; (3): dual feasibility; (4): complementary slackness; (5): $\tilde{\mathbf{x}}$ minimizes $L(\cdot, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}})$.

KKT conditions for convex problems: optimality

- Assume: $(f_i)_{i=0}^m$ are convex and differentiable, and $(h_j)_{j=1}^p$ are affine.
- Let $\left(\tilde{\mathbf{x}}, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}}\right)$ satisfy the KKT conditions:

$$
\begin{aligned}
f_i(\tilde{\mathbf{x}}) &\le 0, \ \forall i \in [m], && (1)\\
h_j(\tilde{\mathbf{x}}) &= 0, \ \forall j \in [p], && (2)\\
\tilde{\lambda}_i &\ge 0, \ \forall i \in [m], && (3)\\
\tilde{\lambda}_i f_i(\tilde{\mathbf{x}}) &= 0, \ \forall i \in [m], && (4)\\
\left.\frac{\partial L(\mathbf{x}, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\tilde{\mathbf{x}}} &= \mathbf{0}, && (5)
\end{aligned}
$$

  i.e., (1)-(2): $\tilde{\mathbf{x}}$ is primal feasible; (3): dual feasibility; (4): complementary slackness; (5): $\tilde{\mathbf{x}}$ minimizes $L(\cdot, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}})$.
- Then, $\tilde{\mathbf{x}}$ and $\left(\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}}\right)$ are primal and dual optimal, with zero duality gap.

KKT conditions for convex problems: optimality

- Assume: $(f_i)_{i=0}^m$ are convex and differentiable, and $(h_j)_{j=1}^p$ are affine.
- Let $\left(\tilde{\mathbf{x}}, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}}\right)$ satisfy the KKT conditions:

$$f_i(\tilde{\mathbf{x}}) \leq 0, \ \forall i \in [m], \tag{1}$$

$$h_j(\tilde{\mathbf{x}}) = 0, \ \forall j \in [p], \tag{2}$$

$$\tilde{\lambda}_i \geq 0, \ \forall i \in [m], \tag{3}$$

$$\tilde{\lambda}_i f_i(\tilde{\mathbf{x}}) = 0, \ \forall i \in [m], \tag{4}$$

$$\left.\frac{\partial L(\mathbf{x}, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\tilde{\mathbf{x}}} = \mathbf{0}, \tag{5}$$

  i.e., (1)-(2): $\tilde{\mathbf{x}}$ is primal feasible; (3): dual feasibility; (4): complementary slackness; (5): $\tilde{\mathbf{x}}$ minimizes $L(\cdot, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}})$.
- Then, $\tilde{\mathbf{x}}$ and $\left(\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}}\right)$ are primal and dual optimal, with zero duality gap.

Let us deploy this statement to our soft SVMC!

Soft SVMC: back to optimization

Objective:

$$\min_{\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^{n}\xi_i, \text{ s.t. } y_i\left(\langle\boldsymbol{\beta},\mathbf{x}_i\rangle + \beta_0\right) \geq 1-\xi_i,\ \xi_i \geq 0 \quad (\forall i) \Leftrightarrow$$

Soft SVMC: back to optimization

Objective:

$$\min_{\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^{n}\xi_i, \text{ s.t. } y_i\left(\langle\boldsymbol{\beta},\mathbf{x}_i\rangle+\beta_0\right)\geq 1-\xi_i,\ \xi_i\geq 0 \quad (\forall i) \Leftrightarrow$$

Standard convex optimization form:

$$\begin{aligned}
\min_{(\boldsymbol{\beta},\beta_0,\boldsymbol{\xi})\in\mathbb{R}^{p+1+n}} \quad & f_0(\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}) := \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^{n}\xi_i \\
\text{s.t.} \quad & f_i(\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}) := 1-\xi_i-y_i\left(\langle\boldsymbol{\beta},\mathbf{x}_i\rangle+\beta_0\right)\leq 0,\ i\in[n], \\
& f_i(\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}) := -\xi_i \leq 0,\ i\in\{n+1,\ldots,n+n\}.
\end{aligned}$$

Soft SVMC: back to optimization

Objective:

$$\min_{\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^n \xi_i, \text{ s.t. } y_i\left(\langle \boldsymbol{\beta}, \mathbf{x}_i\rangle + \beta_0\right) \geq 1 - \xi_i,\ \xi_i \geq 0 \quad (\forall i) \Leftrightarrow$$

Standard convex optimization form:

$$\begin{aligned}
\min_{(\boldsymbol{\beta},\beta_0,\boldsymbol{\xi})\in\mathbb{R}^{p+1+n}} \quad & f_0(\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}) := \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^n \xi_i \\
\text{s.t.} \quad & f_i(\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}) := 1 - \xi_i - y_i\left(\langle \boldsymbol{\beta}, \mathbf{x}_i\rangle + \beta_0\right) \leq 0,\ i \in [n], \\
& f_i(\boldsymbol{\beta},\beta_0,\boldsymbol{\xi}) := -\xi_i \leq 0,\ i \in \{n+1,\ldots,n+n\}.
\end{aligned}$$

KKT conditions kick in

$\Leftarrow$ f_i-s are convex and differentiable, no equality constraints.

Soft SVMC: continued

Lagrangian function: with $\alpha_i \geq 0, \mu_i \geq 0$ $(\forall i)$

$$L(\boldsymbol{\beta}, \beta_0, \boldsymbol{\xi}; \boldsymbol{\alpha}, \boldsymbol{\mu}) = \text{objective} + \text{Lagrangian multipliers} \times \text{conditions}$$
$$= \frac{1}{2} \|\boldsymbol{\beta}\|_2^2 + C \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i \left[y_i \left(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 \right) - 1 + \xi_i \right] - \sum_{i=1}^{n} \mu_i \xi_i .$$

Solving for $\frac{\partial L}{\partial \text{primal}} = 0$, we get ...

Soft SVMC: continued

$$L(\boldsymbol{\beta}, \beta_0, \boldsymbol{\xi}; \boldsymbol{\alpha}, \boldsymbol{\mu}) =$$
$$= \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + C\sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i \left[y_i\left(\langle \boldsymbol{\beta}, \mathbf{x}_i\rangle + \beta_0\right) - 1 + \xi_i\right] - \sum_{i=1}^n \mu_i \xi_i.$$

Optimality equations:

$$\mathbf{0} = \frac{\partial L}{\partial \boldsymbol{\beta}} = \boldsymbol{\beta} - \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i \quad (\boldsymbol{\beta} \leftrightarrow \boldsymbol{\alpha}),$$
$$0 = \frac{\partial L}{\partial \beta_0} = \sum_{i=1}^n \alpha_i y_i,$$
$$0 = \frac{\partial L}{\partial \xi_i} = C - \alpha_i - \mu_i, \ \forall i \in [n].$$

Plugging these equations back to L, we have ...

Soft SVMC: after a bit of calculation

- Lagrange dual problem (QP):

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^n} \underbrace{\sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j \boxed{\langle \mathbf{x}_i, \mathbf{x}_j\rangle}}_{\text{quadratic in } \boldsymbol{\alpha}}, \text{ s.t. } \underbrace{0 \le \alpha_i \le C\ (\forall i), \sum_{i=1}^{n}\alpha_i y_i = 0}_{\text{linear in } \boldsymbol{\alpha}}.$$

Soft SVMC: after a bit of calculation

- Lagrange dual problem (QP):

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^n} \underbrace{\sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i,j=1}^n \alpha_i\alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j\rangle}_{\text{quadratic in } \boldsymbol{\alpha}}, \text{ s.t. } \underbrace{0 \le \alpha_i \le C\ (\forall i), \sum_{i=1}^n \alpha_i y_i = 0}_{\text{linear in } \boldsymbol{\alpha}}.$$

- β_0 recovery: from complementary slackness, i.e. when $\alpha_i \neq 0$ (support vectors), $y_i\left(\langle \boldsymbol{\beta}, \mathbf{x}_i\rangle + \beta_0\right) - 1 = 0$ needs to hold.

Soft SVMC: after a bit of calculation

- Lagrange dual problem (QP):

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^n} \underbrace{\sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j \boxed{\langle \mathbf{x}_i, \mathbf{x}_j\rangle}}_{\text{quadratic in } \boldsymbol{\alpha}}, \text{ s.t. } \underbrace{0 \le \alpha_i \le C \ (\forall i), \sum_{i=1}^{n}\alpha_i y_i = 0}_{\text{linear in } \boldsymbol{\alpha}}.$$

- β_0 recovery: from complementary slackness, i.e. when $\alpha_i \neq 0$ (support vectors), $y_i\left(\langle \boldsymbol{\beta}, \mathbf{x}_i\rangle + \beta_0\right) - 1 = 0$ needs to hold.
- Decision:

$$\hat{y}(\mathbf{x}) = \operatorname{sign}(\langle \boldsymbol{\beta}, \mathbf{x}\rangle + \beta_0) = \operatorname{sign}\left(\left\langle \sum_{i\in[n]}\alpha_i y_i \mathbf{x}_i, \mathbf{x}\right\rangle + \beta_0\right)$$

Soft SVMC: after a bit of calculation

- Lagrange dual problem (QP):

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^n} \underbrace{\sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i,j=1}^n \alpha_i\alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle}_{\text{quadratic in } \boldsymbol{\alpha}}, \quad \text{s.t.} \quad \underbrace{0 \le \alpha_i \le C \ (\forall i), \sum_{i=1}^n \alpha_i y_i = 0}_{\text{linear in } \boldsymbol{\alpha}}.$$

- β_0 recovery: from complementary slackness, i.e. when $\alpha_i \neq 0$ (support vectors), $y_i \left(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + \beta_0 \right) - 1 = 0$ needs to hold.
- Decision:

$$\hat{y}(\mathbf{x}) = \operatorname{sign}(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + \beta_0) = \operatorname{sign}\left(\left\langle \sum_{i\in[n]} \alpha_i y_i \mathbf{x}_i, \mathbf{x} \right\rangle + \beta_0 \right)$$

$$= \operatorname{sign}\left(\sum_{i\in[n]} \alpha_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + \beta_0 \right).$$

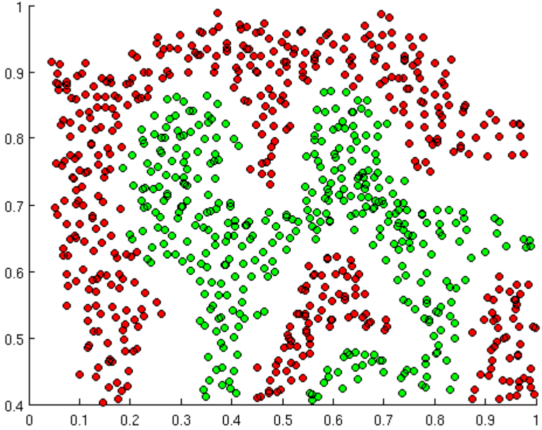
(Towards) nonlinear SVMC

If linear separability does not hold

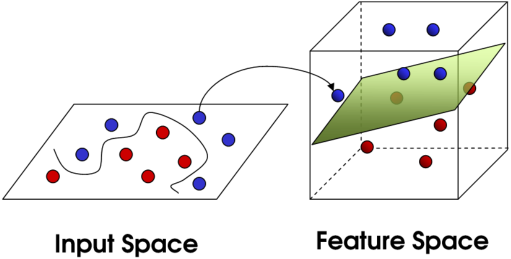
- Until this point:
  - (almost) linearly separable case.

If linear separability does not hold

- Until this point:
  - (almost) linearly separable case.
- Now:



If linear separability does not hold: kernel trick



Nonlinear SVMC

- Linear SVMC (dual):

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle, \text{ s.t. } \sum_{i=1}^{n} \alpha_i y_i = 0,\ 0 \le \alpha_i \le C\ (\forall i).$$

Nonlinear SVMC

- Linear SVMC (dual):

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle, \text{ s.t. } \sum_{i=1}^{n} \alpha_i y_i = 0,\ 0 \le \alpha_i \le C\ (\forall i).$$

- Nonlinear SVMC (dual):

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j k(x_i, x_j), \text{ s.t. } \sum_{i=1}^{n} \alpha_i y_i = 0,\ 0 \le \alpha_i \le C\ (\forall i).$$

Nonlinear SVMC

- Linear SVMC (dual):

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle, \text{ s.t. } \sum_{i=1}^{n} \alpha_i y_i = 0,\ 0 \le \alpha_i \le C\ (\forall i).$$

- Nonlinear SVMC (dual):

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j k(x_i, x_j), \text{ s.t. } \sum_{i=1}^{n} \alpha_i y_i = 0,\ 0 \le \alpha_i \le C\ (\forall i).$$

- Nonlinear SVMC (primal):

$$\min_{f\in\mathcal{H}_k, \boldsymbol{\xi}} \frac{1}{2}\|f\|_{\mathcal{H}_k}^2 + C\sum_{i=1}^{n} \xi_i, \text{ s.t. } y_i f(x_i) \ge 1 - \xi_i,\ \xi_i \ge 0,\ \forall i.$$

What is $\mathcal{H}_k$? Note: '$+\beta_0$' also works.

Kernel examples on $\mathbb{R}^d$ ($c \geq 0$, $p \in \mathbb{Z}^+$)

$$k_p(\mathbf{x}, \mathbf{y}) = (\langle \mathbf{x}, \mathbf{y} \rangle + c)^p$$

Kernel examples on $\mathbb{R}^d$ ($\gamma > 0$, $c \geq 0$, $p \in \mathbb{Z}^+$)

$$k_p(\mathbf{x},\mathbf{y}) = (\langle \mathbf{x},\mathbf{y}\rangle + c)^p, \qquad k_G(\mathbf{x},\mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_2^2},$$

$$k_e(\mathbf{x},\mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_2}, \qquad k_L(\mathbf{x},\mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_1},$$

$$k_C(\mathbf{x},\mathbf{y}) = \frac{1}{1+\gamma\left\|\mathbf{x}-\mathbf{y}\right\|_2^2}, \qquad k_{\tilde{e}}(\mathbf{x},\mathbf{y}) = e^{\gamma\langle \mathbf{x},\mathbf{y}\rangle}.$$

Kernel examples on $\mathbb{R}^d$ ($\gamma, \sigma, v > 0$, $c \geq 0$, $p \in \mathbb{Z}^+$)

$$k_p(\mathbf{x},\mathbf{y}) = (\langle \mathbf{x},\mathbf{y}\rangle + c)^p, \qquad k_G(\mathbf{x},\mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_2^2},$$

$$k_e(\mathbf{x},\mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_2}, \qquad k_L(\mathbf{x},\mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_1},$$

$$k_C(\mathbf{x},\mathbf{y}) = \frac{1}{1+\gamma\|\mathbf{x}-\mathbf{y}\|_2^2}, \qquad k_{\tilde{e}}(\mathbf{x},\mathbf{y}) = e^{\gamma\langle \mathbf{x},\mathbf{y}\rangle}.$$

Or the flexible Matérn family:

$$k_M(\mathbf{x},\mathbf{y}) = \frac{2^{1-v}}{\Gamma(v)}\left(\frac{\sqrt{2v}\,\|\mathbf{x}-\mathbf{y}\|_2}{\sigma}\right)^{v} K_v\left(\frac{\sqrt{2v}\,\|\mathbf{x}-\mathbf{y}\|_2}{\sigma}\right),$$

where

- K_v: modified Bessel function of the second kind of order v,
- Specific cases: For $v=\frac{1}{2}$ one gets $k(\mathbf{x},\mathbf{y}) = e^{-\frac{\|\mathbf{x}-\mathbf{y}\|_2}{\sigma}}$. Gaussian kernel: $v \to \infty$.

Some kernel-enriched domains: $(\mathcal{X}, k)$

- Strings [Watkins, 1999, Lodhi et al., 2002, Leslie et al., 2002, Kuang et al., 2004, Leslie and Kuang, 2004, Saigo et al., 2004, Cuturi and Vert, 2005],
- time series [Rüping, 2001, Cuturi et al., 2007, Cuturi, 2011, Király and Oberhauser, 2019],
- trees [Collins and Duffy, 2001, Kashima and Koyanagi, 2002],
- groups and specifically rankings [Cuturi et al., 2005, Jiao and Vert, 2016],
- sets [Haussler, 1999, Gärtner et al., 2002, Balanca and Herbin, 2012, Fellmann et al., 2024], probability distributions [Berlinet and Thomas-Agnan, 2004, Hein and Bousquet, 2005, Smola et al., 2007, Sriperumbudur et al., 2010],
- various generative models [Jaakkola and Haussler, 1999, Tsuda et al., 2002, Seeger, 2002, Jebara et al., 2004],
- fuzzy domains [Guevara et al., 2017], or
- graphs [Kondor and Lafferty, 2002, Gärtner et al., 2003, Kashima et al., 2003, Borgwardt and Kriegel, 2005, Shervashidze et al., 2009, Vishwanathan et al., 2010, Kondor and Pan, 2016, Bai et al., 2020, Borgwardt et al., 2020, Schulz et al., 2022, Nikolentzos and Vazirgiannis, 2023].

Kernel (generalization of $\mathbf{a}^\top \mathbf{b}$), RKHS

Our assumption

input space: $\mathcal{X}$ is a set.

Def-1 (feature): $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called **kernel** if $\exists$ feature map $\varphi : \mathcal{X} \to \mathcal{H}$(ilbert) s.t.

$$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}, \quad x, x' \in \mathcal{X}.$$

Def-2 (reproducing kernel):

- $\mathcal{H} \subset \mathbb{R}^{\mathcal{X}}$: Hilbert space.
- $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a r.k. of $\mathcal{H}$ if for $\forall x \in \mathcal{X}, f \in \mathcal{H}$:
  1. $k(\cdot, x) \in \mathcal{H}$: 'generators',
  2. $f(x) = \langle f, k(\cdot, x)\rangle_{\mathcal{H}}$: reproducing property.

Kernel, RKHS – continued

Def-2 (reproducing kernel):

- $\mathcal{H} \subset \mathbb{R}^{\mathcal{X}}$: Hilbert space.
- $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a r.k. of $\mathcal{H}$ if for $\forall x \in \mathcal{X}, f \in \mathcal{H}$:
  1. $k(\cdot, x) \in \mathcal{H}$: 'generators',
  2. $f(x) = \langle f, k(\cdot, x) \rangle_{\mathcal{H}}$: reproducing property.

Remarks

1. Specifically, $k(x, y) = \langle k(\cdot, x), k(\cdot, y) \rangle_{\mathcal{H}}$;
   note: $k(\cdot, x) =:$ canonical feature map.
2. Reproducing property $\Rightarrow$ computational tractability.

Kernel, RKHS – continued

- Def-3 (Gram matrix):
  - $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ symmetric function.
  - k is called positive definite if for $\forall n \in \mathbb{Z}^+, (x_i)_{i=1}^n \in \mathcal{X}^n$

$$\mathbb{R}^{n\times n} \ni \mathbf{G} = [k(x_i, x_j)]_{i,j=1}^n \succeq \mathbf{0}_{n\times n}.$$

Kernel, RKHS – continued

- Def-3 (Gram matrix):
  - $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ symmetric function.
  - k is called **positive definite** if for $\forall n \in \mathbb{Z}^+, (x_i)_{i=1}^n \in \mathcal{X}^n$

$$\mathbb{R}^{n\times n} \ni \mathbf{G} = [k(x_i, x_j)]_{i,j=1}^n \succeq \mathbf{0}_{n\times n}.$$

Benefits

1. Optimization advantage: $\exists(\mathbf{G} + \lambda\mathbf{I})^{-1}$.
2. Computation: reduces to linear algebra.

Kernel, RKHS – continued

Def-4 (evaluation):

- $\mathcal{H} \subset \mathbb{R}^{\mathcal{X}}$: Hilbert space.
- Let $\delta_x : f \in \mathcal{H} \mapsto f(x) \in \mathbb{R}$, with $x \in \mathcal{X}$.
- $\mathcal{H}$ is called RKHS if for $\forall x \in \mathcal{X}$, δ_x is continuous.

Kernel, RKHS – continued

Def-4 (evaluation):

- $\mathcal{H} \subset \mathbb{R}^{\mathcal{X}}$: Hilbert space.
- Let $\delta_x : f \in \mathcal{H} \mapsto f(x) \in \mathbb{R}$, with $x \in \mathcal{X}$.
- $\mathcal{H}$ is called RKHS if for $\forall x \in \mathcal{X}$, δ_x is continuous.

Advantage

Strong notion of convergence: $f_n \to f \Rightarrow$ for $\forall x \in \mathcal{X}$, $f_n(x) \to f(x)$.

Kernel, RKHS – continued

Def-4 (evaluation):

- $\mathcal{H} \subset \mathbb{R}^{\mathcal{X}}$: Hilbert space.
- Let $\delta_x : f \in \mathcal{H} \mapsto f(x) \in \mathbb{R}$, with $x \in \mathcal{X}$.
- $\mathcal{H}$ is called RKHS if for $\forall x \in \mathcal{X}$, δ_x is continuous.

Advantage

Strong notion of convergence: $f_n \to f \Rightarrow$ for $\forall x \in \mathcal{X}$, $f_n(x) \to f(x)$.

Note:

- $k \overset{1:1}{\leftrightarrow} \mathcal{H}_k = \overline{\text{Span}}(k(\cdot, x) : x \in \mathcal{X})$: Fourier, polynomials, splines, ...

Kernel, RKHS – continued

Def-4 (evaluation):

- $\mathcal{H} \subset \mathbb{R}^{\mathcal{X}}$: Hilbert space.
- Let $\delta_x : f \in \mathcal{H} \mapsto f(x) \in \mathbb{R}$, with $x \in \mathcal{X}$.
- $\mathcal{H}$ is called RKHS if for $\forall x \in \mathcal{X}$, δ_x is continuous.

Advantage

Strong notion of convergence: $f_n \to f \Rightarrow$ for $\forall x \in \mathcal{X}$, $f_n(x) \to f(x)$.

Note:

- $k \overset{1:1}{\leftrightarrow} \mathcal{H}_k = \overline{\text{Span}}(k(\cdot, x) : x \in \mathcal{X})$: Fourier, polynomials, splines, ...

How do we construct kernels?

Kernel factory

- We know: $k(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{y}$ is a kernel.

Kernel factory

- We know: $k(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{y}$ is a kernel.
- A few useful rules:
  1. Non-negative shift. k: kernel $\Rightarrow k + \gamma$: kernel ($\gamma \in \mathbb{R}^{\geq 0}$). Why?

Kernel factory

- We know: $k(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{y}$ is a kernel.
- A few useful rules:
  1. Non-negative shift. k: kernel $\Rightarrow k + \gamma$: kernel ($\gamma \in \mathbb{R}^{\geq 0}$). Why? $\Leftarrow$ Gram.

Kernel factory

- We know: $k(\mathbf{x},\mathbf{y}) = \mathbf{x}^\top \mathbf{y}$ is a kernel.
- A few useful rules:
  1. Non-negative shift. k: kernel $\Rightarrow k+\gamma$: kernel ($\gamma \in \mathbb{R}^{\geq 0}$). Why? $\Leftarrow$ Gram.
  2. Cone. If $k_m : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ kernel, $\alpha_m \geq 0$ ($m = 1, \ldots, M$), then

$$\sum_{m=1}^{M} \alpha_m k_m(x,y) = \ ?$$

Kernel factory

- We know: $k(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{y}$ is a kernel.
- A few useful rules:
  1. Non-negative shift. k: kernel $\Rightarrow k + \gamma$: kernel ($\gamma \in \mathbb{R}^{\geq 0}$). Why? $\Leftarrow$ Gram.
  2. Cone. If $k_m : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ kernel, $\alpha_m \geq 0$ ($m = 1, \ldots, M$), then

$$\sum_{m=1}^{M} \alpha_m k_m(x, y) = \sum_m \alpha_m \langle \varphi_m(x), \varphi_m(y) \rangle_{\mathcal{H}_m}$$

Kernel factory

- We know: $k(\mathbf{x},\mathbf{y}) = \mathbf{x}^\top \mathbf{y}$ is a kernel.
- A few useful rules:
  1. Non-negative shift. k: kernel $\Rightarrow k+\gamma$: kernel ($\gamma \in \mathbb{R}^{\geq 0}$). Why? $\Leftarrow$ Gram.
  2. Cone. If $k_m : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ kernel, $\alpha_m \geq 0$ ($m = 1,\ldots,M$), then

$$\sum_{m=1}^{M} \alpha_m k_m(x,y) = \sum_m \alpha_m \langle \varphi_m(x), \varphi_m(y)\rangle_{\mathcal{H}_m}$$
$$= \langle \varphi(x), \varphi(y)\rangle_{\mathcal{H}},$$
$$\varphi(x) = (\sqrt{\alpha_1}\varphi_1(x), \ldots, \sqrt{\alpha_M}\varphi_M(x)) \in \mathcal{H} := \oplus_{m=1}^{M}\mathcal{H}_m.$$

Kernel factory

- We know: $k(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{y}$ is a kernel.
- A few useful rules:
  1. Non-negative shift. k: kernel $\Rightarrow k + \gamma$: kernel ($\gamma \in \mathbb{R}^{\geq 0}$). Why? $\Leftarrow$ Gram.
  2. Cone. If $k_m : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ kernel, $\alpha_m \geq 0$ ($m = 1, \ldots, M$), then

$$\begin{aligned}
\sum_{m=1}^{M} \alpha_m k_m(x, y) &= \sum_{m} \alpha_m \langle \varphi_m(x), \varphi_m(y) \rangle_{\mathcal{H}_m} \\
&= \langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}}, \\
\varphi(x) &= (\sqrt{\alpha_1}\varphi_1(x), \ldots, \sqrt{\alpha_M}\varphi_M(x)) \in \mathcal{H} := \oplus_{m=1}^{M} \mathcal{H}_m.
\end{aligned}$$

Example: $\oplus_{m=1}^{M} \mathbb{R} = \mathbb{R}^M$.

Kernel factory – continued

4. **Product.** If $(k_m)_{m=1}^M$ are kernels on $(\mathcal{X}_m)_{m=1}^M$, then

$$\left(\otimes_{m=1}^M k_m\right)\left(\left(x_1, \ldots, x_M\right), \left(x_1', \ldots, x_M'\right)\right) = \prod_{m=1}^M k_m\left(x_m, x_m'\right).$$

Kernel factory – continued

4. Product. If $(k_m)_{m=1}^M$ are kernels on $(\mathcal{X}_m)_{m=1}^M$, then

$$\left(\otimes_{m=1}^M k_m\right)\left((x_1,\ldots,x_M),(x_1',\ldots,x_M')\right)=\prod_{m=1}^M k_m\left(x_m,x_m'\right).$$

- Thus, $(k_m)_{m=1}^M : \mathcal{X}\times\mathcal{X}\to\mathbb{R}$ kernels $\Rightarrow \prod_{m=1}^M k_m(x,x')$: kernel on $\mathcal{X}$.

Kernel factory – continued

4. Product. If $(k_m)_{m=1}^M$ are kernels on $(\mathcal{X}_m)_{m=1}^M$, then

$$\left(\otimes_{m=1}^M k_m\right)\left((x_1,\ldots,x_M),(x_1',\ldots,x_M')\right)=\prod_{m=1}^M k_m\left(x_m,x_m'\right).$$

- Thus, $(k_m)_{m=1}^M:\mathcal{X}\times\mathcal{X}\to\mathbb{R}$ kernels $\Rightarrow\prod_{m=1}^M k_m(x,x')$: kernel on $\mathcal{X}$.
- Consequence ($\gamma\geq 0$, $p\in\mathbb{Z}^+$):

$$k(\mathbf{x},\mathbf{y})=\left(\langle\mathbf{x},\mathbf{y}\rangle_2+\gamma\right)^p$$

is a kernel.

Kernel factory

6. Limit. If $(k_n)_{n\in\mathbb{N}}$ are kernels on $\mathcal{X}$, then

$$k(x,x') := \lim_{n\to\infty} k_n(x,x')$$

is a kernel. Why?

Kernel factory

6. Limit. If $(k_n)_{n\in\mathbb{N}}$ are kernels on $\mathcal{X}$, then

$$k(x,x') := \lim_{n\to\infty} k_n(x,x')$$

is a kernel. Why? $\Leftarrow$ Gram.

Kernel factory

6. Limit. If $(k_n)_{n\in\mathbb{N}}$ are kernels on $\mathcal{X}$, then

$$k(x,x') := \lim_{n\to\infty} k_n(x,x')$$

is a kernel. Why? $\Leftarrow$ Gram.
Example ($\gamma > 0$):

$$k(\mathbf{x},\mathbf{y}) = e^{\gamma\langle\mathbf{x},\mathbf{y}\rangle_2} = \sum_{n\in\mathbb{N}} \frac{\left(\gamma\langle\mathbf{x},\mathbf{y}\rangle_2\right)^n}{n!}$$

Kernel factory

6. Limit. If $(k_n)_{n\in\mathbb{N}}$ are kernels on $\mathcal{X}$, then

$$k(x,x') := \lim_{n\to\infty} k_n(x,x')$$

is a kernel. Why? $\Leftarrow$ Gram.
Example ($\gamma > 0$):

$$k(\mathbf{x},\mathbf{y}) = e^{\gamma\langle\mathbf{x},\mathbf{y}\rangle_2} = \sum_{n\in\mathbb{N}} \frac{\left(\gamma\langle\mathbf{x},\mathbf{y}\rangle_2\right)^n}{n!}$$

Reason: polynomial kernel & limit rule.

Kernel factory – continued

7. Pre-post multiplication. k kernel on $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$, then

$$\tilde{k}(x, y) = f(x)k(x, y)f(y)$$

is a kernel. Check (feature view):

$$\tilde{k}(x, y) = f(x)k(x, y)f(y)$$

Kernel factory – continued

7. Pre-post multiplication. k kernel on $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$, then

$$\tilde{k}(x, y) = f(x)k(x, y)f(y)$$

is a kernel. Check (feature view):

$$\tilde{k}(x, y) = f(x)k(x, y)f(y) = f(x)\langle\varphi(x), \varphi(y)\rangle_{\mathcal{H}}f(y)$$

Kernel factory – continued

7. Pre-post multiplication. k kernel on $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$, then

$$\tilde{k}(x,y) = f(x)k(x,y)f(y)$$

is a kernel. Check (feature view):

$$\begin{aligned}\tilde{k}(x,y) &= f(x)k(x,y)f(y) = f(x)\langle \varphi(x), \varphi(y)\rangle_{\mathcal{H}} f(y)\\ &= \Big\langle \underbrace{f(x)\varphi(x)}_{=:\tilde{\varphi}(x)}, f(y)\varphi(y)\Big\rangle_{\mathcal{H}}.\end{aligned}$$

Kernel factory – continued

7. Pre-post multiplication. k kernel on $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$, then

$$\tilde{k}(x, y) = f(x)k(x, y)f(y)$$

is a kernel. Check (feature view):

$$\begin{aligned}\tilde{k}(x, y) &= f(x)k(x, y)f(y) = f(x)\langle\varphi(x), \varphi(y)\rangle_{\mathcal{H}}f(y)\\ &= \Big\langle \underbrace{f(x)\varphi(x)}_{=:\tilde{\varphi}(x)}, f(y)\varphi(y)\Big\rangle_{\mathcal{H}}.\end{aligned}$$

Example (Gaussian kernel, $\gamma > 0$): previous example & new rule

$$k(\mathbf{x}, \mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_2^2}$$

by using $\|\mathbf{x} - \mathbf{y}\|_2^2 = \|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2 - 2\langle\mathbf{x}, \mathbf{y}\rangle$.

Kernel factory – continued

7. Pre-post multiplication. k kernel on $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$, then

$$\tilde{k}(x,y) = f(x)k(x,y)f(y)$$

is a kernel. Check (feature view):

$$\tilde{k}(x,y) = f(x)k(x,y)f(y) = f(x)\langle \varphi(x), \varphi(y)\rangle_{\mathcal{H}} f(y)$$
$$= \Big\langle \underbrace{f(x)\varphi(x)}_{=:\tilde{\varphi}(x)}, f(y)\varphi(y) \Big\rangle_{\mathcal{H}}.$$

Example (Gaussian kernel, $\gamma > 0$): previous example & new rule

$$k(\mathbf{x},\mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_2^2} = e^{-\gamma\|\mathbf{x}\|_2^2} e^{2\gamma\langle \mathbf{x},\mathbf{y}\rangle_2} e^{-\gamma\|\mathbf{y}\|_2^2}$$

by using $\|\mathbf{x}-\mathbf{y}\|_2^2 = \|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2 - 2\langle \mathbf{x},\mathbf{y}\rangle$.

Properties of k control that of $\mathcal{H}_k$

- k: bounded $[\sup_{x,y\in\mathcal{X}} k(x,y) \le C] \Rightarrow \forall f \in \mathcal{H}_k$ is bounded

Properties of k control that of $\mathcal{H}_k$

- k: bounded $[\sup_{x,y\in\mathcal{X}} k(x,y) \le C] \Rightarrow \forall f \in \mathcal{H}_k$ is bounded:

$$|f(x)| \overset{\text{repr}}{=} \left|\langle f, k(\cdot,x)\rangle_{\mathcal{H}_k}\right| \overset{\text{CBS}}{\le} \|f\|_{\mathcal{H}_k} \underbrace{\|k(\cdot,x)\|_{\mathcal{H}_k}}_{\sqrt{k(x,x)}}.$$

Properties of k control that of $\mathcal{H}_k$

- k: bounded $[\sup_{x,y\in\mathcal{X}} k(x,y) \le C] \Rightarrow \forall f \in \mathcal{H}_k$ is bounded:

$$|f(x)| \overset{\text{repr}}{=} \left|\langle f, k(\cdot,x)\rangle_{\mathcal{H}_k}\right| \overset{\text{CBS}}{\le} \|f\|_{\mathcal{H}_k} \underbrace{\|k(\cdot,x)\|_{\mathcal{H}_k}}_{\sqrt{k(x,x)}}.$$

- k: continuous $\Rightarrow \mathcal{H}_k$: separable $\left[\ell^2(\mathbb{N})\right]$.

Properties of k control that of $\mathcal{H}_k$

- k: bounded $[\sup_{x,y\in\mathcal{X}} k(x,y) \le C] \Rightarrow \forall f \in \mathcal{H}_k$ is bounded:

$$|f(x)| \overset{\text{repr}}{=} \left|\langle f, k(\cdot,x)\rangle_{\mathcal{H}_k}\right| \overset{\text{CBS}}{\le} \|f\|_{\mathcal{H}_k} \underbrace{\|k(\cdot,x)\|_{\mathcal{H}_k}}_{\sqrt{k(x,x)}}.$$

- k: continuous $\Rightarrow \mathcal{H}_k$: separable $\left[\ell^2(\mathbb{N})\right]$.
- k: bounded and continuous $\Rightarrow \forall f \in \mathcal{H}_k$ is bounded & continuous.

Properties of k control that of $\mathcal{H}_k$

- k: bounded $[\sup_{x,y\in\mathcal{X}} k(x,y) \le C] \Rightarrow \forall f \in \mathcal{H}_k$ is bounded:

$$|f(x)| \overset{\text{repr}}{=} \left|\langle f, k(\cdot,x)\rangle_{\mathcal{H}_k}\right| \overset{\text{CBS}}{\le} \|f\|_{\mathcal{H}_k} \underbrace{\|k(\cdot,x)\|_{\mathcal{H}_k}}_{\sqrt{k(x,x)}}.$$

- k: continuous $\Rightarrow \mathcal{H}_k$: separable $\left[\ell^2(\mathbb{N})\right]$.
- k: bounded and continuous $\Rightarrow \forall f \in \mathcal{H}_k$ is bounded & continuous.
- $k \in C^m \Rightarrow \forall f \in \mathcal{H}_k$ is m-times continuously differentiable.

Properties of k control that of $\mathcal{H}_k$

- k: bounded $[\sup_{x,y\in\mathcal{X}} k(x,y) \le C] \Rightarrow \forall f \in \mathcal{H}_k$ is bounded:

$$|f(x)| \overset{\text{repr}}{=} \left|\langle f, k(\cdot,x)\rangle_{\mathcal{H}_k}\right| \overset{\text{CBS}}{\le} \|f\|_{\mathcal{H}_k} \underbrace{\|k(\cdot,x)\|_{\mathcal{H}_k}}_{\sqrt{k(x,x)}}.$$

- k: continuous $\Rightarrow \mathcal{H}_k$: separable $\left[\ell^2(\mathbb{N})\right]$.
- k: bounded and continuous $\Rightarrow \forall f \in \mathcal{H}_k$ is bounded & continuous.
- $k \in C^m \Rightarrow \forall f \in \mathcal{H}_k$ is m-times continuously differentiable.
- k: analytic $\Rightarrow \forall f \in \mathcal{H}_k$ is analytic.

Representer theorem ⇒ finiteD parameterization!

- Given: $\{(x_i, y_i)\}_{i=1}^n$, say classification/regression.
- Goal:

$$J(f) = V(x_1, y_1, f(x_1), \ldots, x_n, y_n, f(x_n)) + r\left(\|f\|^2_{\mathcal{H}_k}\right) \to \min_{f \in \mathcal{H}_k},$$

r : monotonically increasing.

Representer theorem ⇒ finiteD parameterization!

- Given: $\{(x_i, y_i)\}_{i=1}^n$, say classification/regression.
- Goal:

$$J(f) = V(x_1, y_1, f(x_1), \ldots, x_n, y_n, f(x_n)) + r\left(\|f\|^2_{\mathcal{H}_k}\right) \to \min_{f \in \mathcal{H}_k},$$

r : monotonically increasing.

- Example:

$$V(\ldots) = \frac{1}{n}\sum_{i\in[n]} \max(1 - y_i f(x_i), 0) \quad \text{(soft classification)},$$

$$V(\ldots) = \frac{1}{n}\sum_{i\in[n]} [f(x_i) - y_i]^2 \quad \text{(regression-1)},$$

$$V(\ldots) = \frac{1}{n}\sum_{i\in[n]} |f(x_i) - y_i|_\epsilon \quad \text{(regression-2)},$$

with $|z|_\epsilon := \max(0, |z| - \epsilon)$, where $z \in \mathbb{R}$.

Representer theorem – continued

...then

- $\exists$ solution in the form:

$$\boxed{f^* = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i)}, \quad \alpha_i \in \mathbb{R}.$$

- r: strictly increasing $\Rightarrow$ $\forall$ solution is of this form.
- Example: $r(z) = \lambda z$, $\lambda > 0$.

Representer theorem – proof

Objective

$$J(f) = V(x_1, y_1, f(x_1), \ldots, x_n, y_n, f(x_n)) + r\left(\|f\|^2_{\mathcal{H}_k}\right) \to \min_{f \in \mathcal{H}_k}.$$

Decompose & Pythagorean theorem:

$$\begin{aligned} S &= \operatorname{Span}\left(k(\cdot, x_i) \,:\, i \in [n]\right), \\ f &= f_S + f_\perp, \\ \|f\|^2_{\mathcal{H}_k} &= \|f_S\|^2_{\mathcal{H}_k} + \underbrace{\|f_\perp\|^2_{\mathcal{H}_k}}_{\geq 0} \geq \|f_S\|^2_{\mathcal{H}_k}. \end{aligned}$$

Representer theorem – proof

Objective

$$J(f) = V(x_1, y_1, f(x_1), \ldots, x_n, y_n, f(x_n)) + r\left(\|f\|^2_{\mathcal{H}_k}\right) \to \min_{f \in \mathcal{H}_k}.$$

Decompose & Pythagorean theorem:

$$\begin{aligned} S &= \operatorname{Span}\left(k(\cdot, x_i) : i \in [n]\right), \\ f &= f_S + f_\perp, \\ \|f\|^2_{\mathcal{H}_k} &= \|f_S\|^2_{\mathcal{H}_k} + \underbrace{\|f_\perp\|^2_{\mathcal{H}_k}}_{\geq 0} \geq \|f_S\|^2_{\mathcal{H}_k}. \end{aligned}$$

In J

- 1st term: depends on f_S only,
  $f(x_i) = \langle f, k(\cdot, x_i)\rangle_{\mathcal{H}_k} = \langle f_S + f_\perp, k(\cdot, x_i)\rangle_{\mathcal{H}_k} = \langle f_S, k(\cdot, x_i)\rangle_{\mathcal{H}_k}.$

Representer theorem – proof

Objective

$$J(f) = V(x_1, y_1, f(x_1), \ldots, x_n, y_n, f(x_n)) + r\left(\|f\|^2_{\mathcal{H}_k}\right) \to \min_{f \in \mathcal{H}_k}.$$

Decompose & Pythagorean theorem:

$$\begin{aligned} S &= \operatorname{Span}\left(k(\cdot, x_i) \,:\, i \in [n]\right), \\ f &= f_S + f_\perp, \\ \|f\|^2_{\mathcal{H}_k} &= \|f_S\|^2_{\mathcal{H}_k} + \underbrace{\|f_\perp\|^2_{\mathcal{H}_k}}_{\geq 0} \geq \|f_S\|^2_{\mathcal{H}_k}. \end{aligned}$$

In J

- 1st term: depends on f_S only,
  $f(x_i) = \langle f, k(\cdot, x_i)\rangle_{\mathcal{H}_k} = \langle f_S + f_\perp, k(\cdot, x_i)\rangle_{\mathcal{H}_k} = \langle f_S, k(\cdot, x_i)\rangle_{\mathcal{H}_k}$.
- 2nd term: can only decrease by neglecting $f_\perp$ ($r \nearrow$).

Regression: kernel ridge regression

Kernel ridge regression

- Given: $\{(x_i, y_i)\}_{i=1}^n$, $\mathcal{H} := \mathcal{H}_k$, $y_i \in \mathbb{R}$.
- Task ($\lambda > 0$):

$$J(f) = \frac{1}{n}\sum_{i=1}^{n}[y_i - f(x_i)]^2 + \lambda \|f\|_{\mathcal{H}}^2 \to \min_{f \in \mathcal{H}}.$$

Kernel ridge regression

- Given: $\{(x_i, y_i)\}_{i=1}^n$, $\mathcal{H} := \mathcal{H}_k$, $y_i \in \mathbb{R}$.
- Task ($\lambda > 0$):

$$J(f) = \frac{1}{n}\sum_{i=1}^{n}[y_i - f(x_i)]^2 + \lambda \|f\|_{\mathcal{H}}^2 \to \min_{f\in\mathcal{H}}.$$

- Analytical solution:

$$f(x) = [k(x_1, x), \ldots, k(x_n, x)]\,(\mathbf{G} + \lambda n \mathbf{I}_n)^{-1}[y_1; \ldots; y_n],$$
$$\mathbf{G} = [k(x_i, x_j)]_{i,j=1}^n.$$

Kernel ridge regression

- Given: $\{(x_i, y_i)\}_{i=1}^n$, $\mathcal{H} := \mathcal{H}_k$, $y_i \in \mathbb{R}$.
- Task ($\lambda > 0$):

$$J(f) = \frac{1}{n}\sum_{i=1}^n [y_i - f(x_i)]^2 + \lambda \|f\|_{\mathcal{H}}^2 \to \min_{f \in \mathcal{H}}.$$

- Analytical solution:

$$f(x) = [k(x_1, x), \ldots, k(x_n, x)] (\mathbf{G} + \lambda n \mathbf{I}_n)^{-1} [y_1; \ldots; y_n],$$
$$\mathbf{G} = [k(x_i, x_j)]_{i,j=1}^n.$$

Question

How do we get this solution?

Kernel ridge regression

By the representer theorem

$$f = \sum_{i=1}^{n} a_i k(\cdot, x_i), \quad (a_i \in \mathbb{R}).$$

Kernel ridge regression

By the representer theorem

$$f = \sum_{i=1}^{n} a_i k(\cdot, x_i), \quad (a_i \in \mathbb{R}).$$

Multiplying the objective by n, using the reproducing property:

$$\begin{aligned}
\tilde{J}(f) &= \sum_{j=1}^{n} \left[y_j - \langle f, k(\cdot, x_j) \rangle_{\mathcal{H}} \right]^2 + \lambda n \|f\|_{\mathcal{H}}^2 \\
&= \|\mathbf{y} - \mathbf{G}\mathbf{a}\|_2^2 + (\lambda n)\mathbf{a}^\top \mathbf{G}\mathbf{a} \\
&= \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{G}\mathbf{a} + \mathbf{a}^\top \left[\mathbf{G}^2 + (\lambda n)\mathbf{G} \right] \mathbf{a}.
\end{aligned}$$

Kernel ridge regression

By the representer theorem

$$f = \sum_{i=1}^{n} a_i k(\cdot, x_i), \quad (a_i \in \mathbb{R}).$$

Multiplying the objective by n, using the reproducing property:

$$\begin{aligned}\tilde{J}(f) &= \sum_{j=1}^{n} \left[y_j - \langle f, k(\cdot, x_j)\rangle_{\mathcal{H}}\right]^2 + \lambda n \|f\|_{\mathcal{H}}^2 \\ &= \|\mathbf{y} - \mathbf{G}\mathbf{a}\|_2^2 + (\lambda n)\mathbf{a}^\top \mathbf{G}\mathbf{a} \\ &= \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{G}\mathbf{a} + \mathbf{a}^\top \left[\mathbf{G}^2 + (\lambda n)\mathbf{G}\right]\mathbf{a}.\end{aligned}$$

Solving $\mathbf{0} = \frac{\partial \tilde{J}}{\partial \mathbf{a}}$, one gets $\mathbf{a}^* = (\mathbf{G} + \lambda n \mathbf{I}_n)^{-1}\mathbf{y}$

Kernel ridge regression

By the representer theorem

$$f = \sum_{i=1}^{n} a_i k(\cdot, x_i), \quad (a_i \in \mathbb{R}).$$

Multiplying the objective by n, using the reproducing property:

$$\begin{aligned}\tilde{J}(f) &= \sum_{j=1}^{n} \left[y_j - \langle f, k(\cdot, x_j)\rangle_{\mathcal{H}}\right]^2 + \lambda n \|f\|_{\mathcal{H}}^2 \\ &= \|\mathbf{y} - \mathbf{G}\mathbf{a}\|_2^2 + (\lambda n)\mathbf{a}^\top \mathbf{G}\mathbf{a} \\ &= \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{G}\mathbf{a} + \mathbf{a}^\top \left[\mathbf{G}^2 + (\lambda n)\mathbf{G}\right]\mathbf{a}.\end{aligned}$$

Solving $\mathbf{0} = \frac{\partial \tilde{J}}{\partial \mathbf{a}}$, one gets $\mathbf{a}^* = (\mathbf{G} + \lambda n \mathbf{I}_n)^{-1}\mathbf{y}$ by

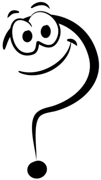
$$\frac{\partial \mathbf{a}^\top \mathbf{B}\mathbf{a}}{\partial \mathbf{a}} = \left(\mathbf{B} + \mathbf{B}^\top\right)\mathbf{a}, \quad \frac{\partial \mathbf{c}^\top \mathbf{a}}{\partial \mathbf{a}} = \mathbf{c}.$$

Summary

1. featurization idea.
2. SVMC:
    - max-margin principle,
    - optimization: duality, KKT conditions,
    - linear (hard/soft), non-linear,
    - kernels, RKHS ('any' data type).
3. representer theorem: finiteD parameterization.
4. kernel ridge regression.

Summary

1. featurization idea.
2. SVMC:
    - max-margin principle,
    - optimization: duality, KKT conditions,
    - linear (hard/soft), non-linear,
    - kernels, RKHS ('any' data type).
3. representer theorem: finiteD parameterization.
4. kernel ridge regression.



Bai, L., Cui, L., Rossi, L., Xu, L., Bai, X., and Hancock, E. (2020).
Local-global nested graph kernels using nested complexity traces.
Pattern Recognition Letters, 134:87–95.

Balanca, P. and Herbin, E. (2012).
A set-indexed Ornstein-Uhlenbeck process.
Electronic Communications in Probability, 17:1–14.

Berlinet, A. and Thomas-Agnan, C. (2004).
Reproducing Kernel Hilbert Spaces in Probability and Statistics.
Kluwer.

Borgwardt, K., Ghisu, E., Llinares-López, F., O'Bray, L., and Riec, B. (2020).
Graph kernels: State-of-the-art and future challenges.
Foundations and Trends in Machine Learning, 13(5-6):531–712.

Borgwardt, K. M. and Kriegel, H.-P. (2005).
Shortest-path kernels on graphs.
In *International Conference on Data Mining (ICDM)*, pages 74–81.

Collins, M. and Duffy, N. (2001).
Convolution kernels for natural language.
In *Advances in Neural Information Processing Systems (NIPS)*, pages 625–632.

Cuturi, M. (2011).
Fast global alignment kernels.
In *International Conference on Machine Learning (ICML)*, pages 929–936.

Cuturi, M., Fukumizu, K., and Vert, J.-P. (2005).
Semigroup kernels on measures.
Journal of Machine Learning Research, 6:1169–1198.

Cuturi, M. and Vert, J.-P. (2005).
The context-tree kernel for strings.

Neural Networks, 18(8):1111–1123.

Cuturi, M., Vert, J.-P., Birkenes, O., and Matsui, T. (2007).
A kernel for time series based on global alignments.
In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 413–416.

Fellmann, N., Blanchet-Scalliet, C., Helbert, C., Spagnol, A., and Sinoquet, D. (2024).
Kernel-based sensivity analysis for (excursion) sets.
Technometrics, 66(4):575–587.

Gärtner, T., Flach, P., Kowalczyk, A., and Smola, A. (2002).
Multi-instance kernels.
In *International Conference on Machine Learning (ICML)*, pages 179–186.

Gärtner, T., Flach, P., and Wrobel, S. (2003).
On graph kernels: Hardness results and efficient alternatives.
Learning Theory and Kernel Machines, pages 129–143.

Guevara, J., Hirata, R., and Canu, S. (2017).
Cross product kernels for fuzzy set similarity.
In *International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6.

Haussler, D. (1999).
Convolution kernels on discrete structures.
Technical report, University of California at Santa Cruz.
(http://cbse.soe.ucsc.edu/sites/default/files/convolutions.pdf).

Hein, M. and Bousquet, O. (2005).
Hilbertian metrics and positive definite kernels on probability measures.
In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 136–143.

Jaakkola, T. S. and Haussler, D. (1999).
Exploiting generative models in discriminative classifiers.

In *Advances in Neural Information Processing Systems (NIPS)*, pages 487–493.

Jebara, T., Kondor, R., and Howard, A. (2004).
Probability product kernels.
Journal of Machine Learning Research, 5:819–844.

Jiao, Y. and Vert, J.-P. (2016).
The Kendall and Mallows kernels for permutations.
In *International Conference on Machine Learning (ICML)*, volume 37, pages 2982–2990.

Kashima, H. and Koyanagi, T. (2002).
Kernels for semi-structured data.
In *International Conference on Machine Learning (ICML)*, pages 291–298.

Kashima, H., Tsuda, K., and Inokuchi, A. (2003).
Marginalized kernels between labeled graphs.
In *International Conference on Machine Learning (ICML)*, pages 321–328.

Király, F. J. and Oberhauser, H. (2019).
Kernels for sequentially ordered data.
Journal of Machine Learning Research, 20:1–45.

Kondor, R. and Pan, H. (2016).
The multiscale Laplacian graph kernel.
In *Advances in Neural Information Processing Systems (NIPS)*, pages 2982–2990.

Kondor, R. I. and Lafferty, J. (2002).
Diffusion kernels on graphs and other discrete input.
In *International Conference on Machine Learning (ICML)*, pages 315–322.

Kuang, R., Ie, E., Wang, K., Wang, K., Siddiqi, M., Freund, Y., and Leslie, C. (2004).
Profile-based string kernels for remote homology detection and motif extraction.
Journal of Bioinformatics and Computational Biology, 13(4):527–550.

Leslie, C., Eskin, E., and Noble, W. S. (2002).
The spectrum kernel: A string kernel for SVM protein classification.
Biocomputing, pages 564–575.

Leslie, C. and Kuang, R. (2004).
Fast string kernels using inexact matching for protein sequences.
Journal of Machine Learning Research, 5:1435–1455.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., and Watkins, C. (2002).
Text classification using string kernels.
Journal of Machine Learning Research, 2:419–444.

Nikolentzos, G. and Vazirgiannis, M. (2023).
Graph alignment kernels using Weisfeiler and Leman hierarchies.
In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2019–2034.

Rüping, S. (2001).
SVM kernels for time series analysis.
Technical report, University of Dortmund.
(http://www.stefan-rueping.de/publications/rueping-2001-a.pdf).

Saigo, H., Vert, J.-P., Ueda, N., and Akutsu, T. (2004).
Protein homology detection using string alignment kernels.
Bioinformatics, 20(11):1682–1689.

Schulz, T. H., Welke, P., and Wrobel, S. (2022).
Graph filtration kernels.
In *AAAI Conference on Artifical Intelligence (AAAI)*, pages 8196–8203.

Seeger, M. (2002).
Covariance kernels from Bayesian generative models.
In *Advances in Neural Information Processing Systems (NIPS)*, pages 905–912.

Shervashidze, N., Vishwanathan, S. V. N., Petri, T., Mehlhorn, K., and Borgwardt, K. M. (2009).
Efficient graphlet kernels for large graph comparison.
In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 488–495.

Smola, A., Gretton, A., Song, L., and Schölkopf, B. (2007).
A Hilbert space embedding for distributions.
In *Algorithmic Learning Theory (ALT)*, pages 13–31.

Sriperumbudur, B., Gretton, A., Fukumizu, K., Schölkopf, B., and Lanckriet, G. (2010).
Hilbert space embeddings and metrics on probability measures.
Journal of Machine Learning Research, 11:1517–1561.

Tsuda, K., Kin, T., and Asai, K. (2002).
Marginalized kernels for biological sequences.
Bioinformatics, 18:268–275.

Vishwanathan, S. N., Schraudolph, N., Kondor, R., and Borgwardt, K. (2010).
Graph kernels.
Journal of Machine Learning Research, 11:1201–1242.

Watkins, C. (1999).
Dynamic alignment kernels.
In *Advances in Neural Information Processing Systems (NIPS)*, pages 39–50.