EE3-23: Machine Learning

Deniz Gündüz and and Krystian Mikolajczyk

Department of Electrical and Electronic Engineering
Imperial College London

Fall 2019

Today

- Support Vector Machines

Today

- Support Vector Machines
- Kernels

Support Vector Machines (SVM)

- One of the most successful classification algorithms (best until a few years ago)
- Maximizing the margin

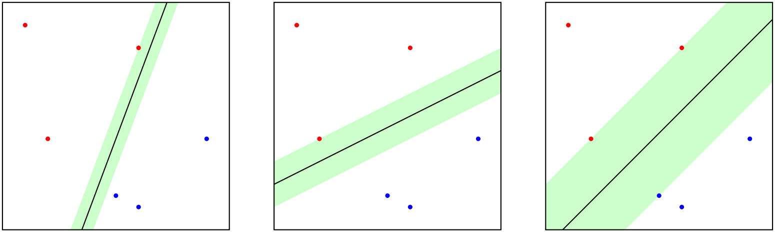
Support Vector Machines (SVM)

- One of the most successful classification algorithms (best until a few years ago)
- Maximizing the margin
  - Intuitively, in classification large margin is good
  - Disciplined explanation

Support Vector Machines (SVM)

- One of the most successful classification algorithms
  (best until a few years ago)

- Maximizing the margin
  - Intuitively, in classification large margin is good
  - Disciplined explanation



Distance to a hyperplane

Hyperplane $H(w,b) = \{x : w^\top x + b = 0\}$ with $\|w\| = 1$

Distance to a hyperplane

Hyperplane $H(w, b) = \{x : w^\top x + b = 0\}$ with $\|w\| = 1$

Distance of x from H is $|w^\top x + b|$

Distance to a hyperplane

Hyperplane $H(w, b) = \{x : w^\top x + b = 0\}$ with $\|w\| = 1$

Distance of x from H is $|w^\top x + b|$

- $p_x = x - (w^\top x + b)w \in H$:

  $$w^\top p_x + b = w^\top(x - (w^\top x + b)w) + b = w^\top x - \|w\|^2(w^\top x + b) + b = 0$$

Distance to a hyperplane

Hyperplane $H(w, b) = \{x : w^\top x + b = 0\}$ with $\|w\| = 1$

Distance of x from H is $|w^\top x + b|$

- $p_x = x - (w^\top x + b)w \in H$:

  $$w^\top p_x + b = w^\top (x - (w^\top x + b)w) + b = w^\top x - \|w\|^2 (w^\top x + b) + b = 0$$

- $x - p_x$ is parallel with $w \Rightarrow$ orthogonal to $H \Rightarrow p_x$ is the orthogonal projection

Distance to a hyperplane

Hyperplane $H(w, b) = \{x : w^\top x + b = 0\}$ with $\|w\| = 1$

Distance of x from H is $|w^\top x + b|$

- $p_x = x - (w^\top x + b)w \in H$:

  $$w^\top p_x + b = w^\top (x - (w^\top x + b)w) + b = w^\top x - \|w\|^2 (w^\top x + b) + b = 0$$

- $x - p_x$ is parallel with $w \Rightarrow$ orthogonal to $H \Rightarrow p_x$ is the orthogonal projection

Distance to a hyperplane

Hyperplane $H(w, b) = \{x : w^\top x + b = 0\}$ with $\|w\| = 1$

Distance of x from H is $|w^\top x + b|$

- $p_x = x - (w^\top x + b)w \in H$:

  $$w^\top p_x + b = w^\top (x - (w^\top x + b)w) + b = w^\top x - \|w\|^2 (w^\top x + b) + b = 0$$

- $x - p_x$ is parallel with $w \Rightarrow$ orthogonal to $H \Rightarrow p_x$ is the orthogonal projection
  More formally: any $u \in H$ is $p_x + v$ where $w^\top v = 0$ ($v \perp w$)

  $$\|x - u\|^2 = \|x - p_x + p_x - u\|^2 = \|\underbrace{x - p_x}_{const \cdot w} + v\|^2 = \|x - p_x\|^2 + \|v\|^2 \geq \|x - p_x\|^2$$

Hard-margin SVM

Points are separable: there exist w and b such that $y_i(w^\top x_i + b) > 0$

Hard-margin SVM

Points are separable: there exist w and b such that $y_i(w^\top x_i + b) > 0$

What is the minimum distance from separator for $\|w\| = 1$:
$\min_i |w^\top x_i + b| = \min_i y_i(w^\top x_i + b)$

Hard-margin SVM

Points are separable: there exist w and b such that $y_i(w^\top x_i + b) > 0$

What is the minimum distance from separator for $\|w\| = 1$:

$\min_i |w^\top x_i + b| = \min_i y_i(w^\top x_i + b)$

Max-margin separator

$(w_*, b_*) = \text{argmax}_{w,b:\|w\|=1} \min_i |w^\top x_i + b|$ s.t. $y_i(w^\top x_i + b) > 0$ for all i

Hard-margin SVM

Equivalently derived from:

Hard-margin SVM

$$\begin{array}{ll} \text{minimize} & \|w\|^2 \\ \text{subject to} & y_i(w^\top x_i + b) \geq 1 \text{ for all } i \end{array}$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i [1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i [1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

KKT (Karush-Kuhn-Tucker sufficient conditions):

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i[1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

KKT (Karush-Kuhn-Tucker sufficient conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i [1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

KKT (Karush-Kuhn-Tucker sufficient conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \qquad \Rightarrow \quad w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i [1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

KKT (Karush-Kuhn-Tucker sufficient conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \qquad \Rightarrow \qquad w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = -\sum_{i=1}^{n} \alpha_i y_i = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i [1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

KKT (Karush-Kuhn-Tucker sufficient conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \quad \Rightarrow \quad w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = -\sum_{i=1}^{n} \alpha_i y_i = 0 \quad \Rightarrow \quad \sum_{i=1}^{n} \alpha_i y_i = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i [1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

KKT (Karush-Kuhn-Tucker sufficient conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \qquad \Rightarrow \qquad w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = -\sum_{i=1}^{n} \alpha_i y_i = 0 \qquad \Rightarrow \qquad \sum_{i=1}^{n} \alpha_i y_i = 0$$

$$\alpha_i [1 - y_i(w^\top x_i + b)] = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i [1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

KKT (Karush-Kuhn-Tucker sufficient conditions):

$$\begin{aligned}
\nabla_w \mathcal{L} &= w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 && \Rightarrow \quad w = \sum_{i=1}^{n} \alpha_i y_i x_i \\
\nabla_b \mathcal{L} &= -\sum_{i=1}^{n} \alpha_i y_i = 0 && \Rightarrow \quad \sum_{i=1}^{n} \alpha_i y_i = 0 \\
\alpha_i [1 - y_i(w^\top x_i + b)] &= 0 && \Rightarrow \quad \alpha_i = 0 \text{ or } y_i(w^\top x_i + b) = 1
\end{aligned}$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i [1 - y_i(w^\top x_i + b)]$$

minimize in primal variables w, b, maximize in dual variables $\alpha_i \geq 0$

KKT (Karush-Kuhn-Tucker sufficient conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \quad \Rightarrow \quad w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = -\sum_{i=1}^{n} \alpha_i y_i = 0 \quad \Rightarrow \quad \sum_{i=1}^{n} \alpha_i y_i = 0$$

$$\alpha_i [1 - y_i(w^\top x_i + b)] = 0 \quad \Rightarrow \quad \alpha_i = 0 \text{ or } y_i(w^\top x_i + b) = 1$$

- Support vectors: x_i with $\alpha_i \neq 0$
- $y_i(w_*^\top x_i + b_*) = 1$ for support vectors
- w is a linear combination of the support vectors

Hard-SVM - Dual Formulation:

$$\max_{\alpha} \mathcal{L}(\alpha) \triangleq \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j x_i^{\top} x_j$$

subject to $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

- Quadratic program
- Once α_is are solved:
  - $w^* = \sum_{i:\alpha_i^* > 0} \alpha_i^* y_i x_i$

Hard-SVM - Dual Formulation:

$$\max_{\alpha} \mathcal{L}(\alpha) \triangleq \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j x_i^\top x_j$$

subject to $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

- Quadratic program
- Once α_is are solved:
  - $w^* = \sum_{i:\alpha_i^* > 0} \alpha_i^* y_i x_i$
  - $b^* = 1 - y_i (w^*)^\top x_i$ for any support vector x_i (equivalently: $b^* = -\frac{\max_{i:y_i=-1}(w^*)^\top x_i + \min_{i:y_i=+1}(w^*)^\top x_i}{2}$)

Prediction with Hard-SVM

Assume we fit our model to a training dataset, and wish to make a prediction for a new data sample x.

- Predict $y = 1$ if and only if $w^T x + b > 0$

We have

$$\begin{aligned} w^T x + b &= \left(\sum_{i=1}^{n} \alpha_i y_i x_i \right)^T x + b \\ &= \sum_{i=1:\alpha_i>0}^{n} \alpha_i y_i (x_i^T x) + b \end{aligned}$$

We only need the inner products with the support vectors!

SVM with feature vectors

Let z be a feature vector for x

Use z instead of x:

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j z_i^\top z_j$$

with constraints $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

What do we need?

- Compute/optimize $\mathcal{L}(\alpha)$: need $z_i^\top z_j$

SVM with feature vectors

Let z be a feature vector for x

Use z instead of x:

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j z_i^\top z_j$$

with constraints $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

What do we need?

- Compute/optimize $\mathcal{L}(\alpha)$: need $z_i^\top z_j$
- Classifier: $g(x) = \text{sign}(w^\top z + b) = \text{sign}\left(\sum_{z_i \text{ is SV}} \alpha_i y_i z_i^\top z + b \right)$

SVM with feature vectors

Let z be a feature vector for x

Use z instead of x:

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j z_i^\top z_j$$

with constraints $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

What do we need?

- Compute/optimize $\mathcal{L}(\alpha)$: need $z_i^\top z_j$
- Classifier: $g(x) = \text{sign}(w^\top z + b) = \text{sign}\left(\sum_{z_i \text{ is SV}} \alpha_i y_i z_i^\top z + b \right)$
- $b = 1 - y_i w^\top z_i$ (z_i support vector)

SVM with feature vectors

Let z be a feature vector for x

Use z instead of x:

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j z_i^\top z_j$$

with constraints $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

What do we need?

- Compute/optimize $\mathcal{L}(\alpha)$: need $z_i^\top z_j$
- Classifier: $g(x) = \text{sign}(w^\top z + b) = \text{sign}\left(\sum_{z_i \text{ is SV}} \alpha_i y_i z_i^\top z + b \right)$
- $b = 1 - y_i w^\top z_i$ (z_i support vector)
- Only need to compute $z_i^\top z_j$!

Generalized inner product

The kernel: $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}')$

- Example: $\mathbf{x} = (x_1, x_2)$

  $\mathbf{z} = \Phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2)$

  $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}') = 1 + x_1 x_1' + x_2 x_2' + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2$

Generalized inner product

The kernel: $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}')$

- Example: $\mathbf{x} = (x_1, x_2)$

  $\mathbf{z} = \Phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2)$

  $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}') = 1 + x_1 x_1' + x_2 x_2' + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2$

- Example: $K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x}^\top \mathbf{x}')^2 = (1 + x_1 x_1' + x_2 x_2')^2$

Generalized inner product

The kernel: $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}')$

- Example: $\mathbf{x} = (x_1, x_2)$
  $\mathbf{z} = \Phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2)$
  $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}') = 1 + x_1 x_1' + x_2 x_2' + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2$

- Example: $K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x}^\top \mathbf{x}')^2 = (1 + x_1 x_1' + x_2 x_2')^2$

Generalized inner product

The kernel: $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}')$

- Example: $\mathbf{x} = (x_1, x_2)$
  $\mathbf{z} = \Phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2)$
  $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}') = 1 + x_1 x_1' + x_2 x_2' + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2$

- Example: $K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x}^\top \mathbf{x}')^2 = (1 + x_1 x_1' + x_2 x_2')^2$
  $= 1 + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2 + 2x_1 x_1' + 2x_2 x_2' + 2x_1 x_2 x_1' x_2'$

Generalized inner product

The kernel: $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}')$

- Example: $\mathbf{x} = (x_1, x_2)$

  $\mathbf{z} = \Phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2)$

  $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}') = 1 + x_1 x_1' + x_2 x_2' + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2$

- Example: $$\begin{aligned} K(\mathbf{x}, \mathbf{x}') &= (1 + \mathbf{x}^\top \mathbf{x}')^2 = (1 + x_1 x_1' + x_2 x_2')^2 \\ &= 1 + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2 + 2x_1 x_1' + 2x_2 x_2' + 2x_1 x_2 x_1' x_2' \end{aligned}$$

  Inner product for

  $\mathbf{z} = (1, x_1^2, x_2^2, \sqrt{2}x_1, \sqrt{2}x_2, \sqrt{2}x_1 x_2)$

Generalized inner product

The kernel: $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}')$

- Example: $\mathbf{x} = (x_1, x_2)$

  $\mathbf{z} = \Phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2)$

  $\mathbf{z}^\top \mathbf{z}' = K(\mathbf{x}, \mathbf{x}') = 1 + x_1 x_1' + x_2 x_2' + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2$

- Example: $$K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x}^\top \mathbf{x}')^2 = (1 + x_1 x_1' + x_2 x_2')^2$$ $$= 1 + x_1^2 {x_1'}^2 + x_2^2 {x_2'}^2 + 2x_1 x_1' + 2x_2 x_2' + 2x_1 x_2 x_1' x_2'$$

  Inner product for

  $\mathbf{z} = (1, x_1^2, x_2^2, \sqrt{2}x_1, \sqrt{2}x_2, \sqrt{2}x_1 x_2)$

Computing $\mathbf{z}$ is not needed! – Kernel trick

Kernel trick

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j z_i^\top z_j$$

with constraints $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

Classifier:

$$g(x) = \text{sign}(w^\top z + b) = \text{sign}\left(\sum_{z_i \text{ is SV}} \alpha_i y_i z_i^\top z + b \right)$$

$$b = 1 - y_i w^\top z_i = 1 - y_i \sum_{z_j \text{ is SV}} \alpha_j y_j z_i^\top z_j$$

Kernel trick

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j K(x_i, x_j)$$

with constraints $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

Classifier:

$$g(x) = \mathrm{sign}(w^\top z + b) = \mathrm{sign}\left(\sum_{z_i \text{ is SV}} \alpha_i y_i K(x_i, x) + b \right)$$

$$b = 1 - y_i w^\top z_i = 1 - y_i \sum_{z_j \text{ is SV}} \alpha_j y_j K(x_i, x_j)$$

Kernel trick

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j K(x_i, x_j)$$

with constraints $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i y_i = 0$.

Classifier:

$$g(x) = \operatorname{sign}(w^\top z + b) = \operatorname{sign}\left(\sum_{z_i \text{ is SV}} \alpha_i y_i K(x_i, x) + b \right)$$

$$b = 1 - y_i w^\top z_i = 1 - y_i \sum_{z_j \text{ is SV}} \alpha_j y_j K(x_i, x_j)$$

Indeed, no need to transform the features as long as we can compute K!

Polynomial kernel

$$K(\mathbf{x}, \mathbf{x}') = (c + \mathbf{x}^\top \mathbf{x}')^q = \left(c + \sum_{j=1}^{d} x_i x_i'\right)^q$$

d^q terms if expanded! $\Rightarrow$ Computational benefits

Gaussian (Radial Basis Function - RBF) kernel

Assume the original instance space is R, and consider feature map

$$\Phi(x)_n = \frac{1}{\sqrt{n!}} \exp -x^2/2x^n$$

Then

$$\begin{aligned}
\Phi(x)^T \Phi(x') &= \sum_{n=0}^{\infty} \left(\frac{1}{\sqrt{n!}} e^{-x^2/2} x^n \right) \left(\frac{1}{\sqrt{n!}} e^{-(x')^2/2} (x')^n \right) \\
&= e^{-\frac{x^2+(x')^2}{2}} \sum_{n=0}^{\infty} \frac{(x \cdot x')^n}{n!} \\
&= e^{-\frac{\|x-x'\|^2}{2}}
\end{aligned}$$

Other kernels?

- Requirement about the kernel K: it computes inner products in the $\mathcal{Z}$ space: $K(x, x') = z^\top z'$

Other kernels?

- Requirement about the kernel K: it computes inner products in the $\mathcal{Z}$ space: $K(x, x') = z^\top z'$
  - Consequences: K is symmetric and positive semidefinite: for any $x_1, \ldots, x_n$,

$$K_{x_1,\ldots,x_n} = \begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) & \ldots & K(x_1,x_n) \\ K(x_2,x_1) & K(x_2,x_2) & \ldots & K(x_2,x_n) \\ \vdots & \ldots & \ddots & \vdots \\ K(x_n,x_1) & K(x_n,x_2) & \ldots & K(x_n,x_n) \end{bmatrix}$$

is symmetric and positive semidefinite (Mercer condition).

Other kernels?

- Requirement about the kernel K: it computes inner products in the $\mathcal{Z}$ space: $K(x, x') = z^\top z'$
  - Consequences: K is symmetric and positive semidefinite: for any $x_1, \dots, x_n$,

$$K_{x_1,\dots,x_n} = \begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) & \dots & K(x_1,x_n) \\ K(x_2,x_1) & K(x_2,x_2) & \dots & K(x_2,x_n) \\ \vdots & \dots & \ddots & \vdots \\ K(x_n,x_1) & K(x_n,x_2) & \dots & K(x_n,x_n) \end{bmatrix}$$

    is symmetric and positive semidefinite (Mercer condition).
  - Indeed, if $Z^\top = (z_1, \dots, z_n)$ then $K_{x_1,\dots,x_n} = ZZ^\top$ and

$$u^\top K_{x_1,\dots,x_n} u = u^\top ZZ^\top u = (Z^\top u)^\top Z^\top u = \|Z^\top u\|^2 \geq 0$$

Other kernels?

- Requirement about the kernel K: it computes inner products in the $\mathcal{Z}$ space: $K(x, x') = z^\top z'$
  - Consequences: K is symmetric and positive semidefinite: for any $x_1, \dots, x_n$,

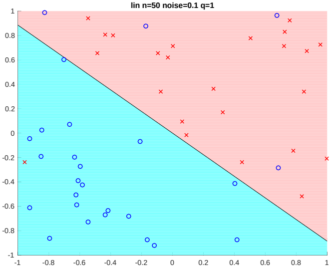
$$K_{x_1,\dots,x_n} = \begin{bmatrix} K(x_1, x_1) & K(x_1, x_2) & \dots & K(x_1, x_n) \\ K(x_2, x_1) & K(x_2, x_2) & \dots & K(x_2, x_n) \\ \vdots & \dots & \ddots & \vdots \\ K(x_n, x_1) & K(x_n, x_2) & \dots & K(x_n, x_n) \end{bmatrix}$$

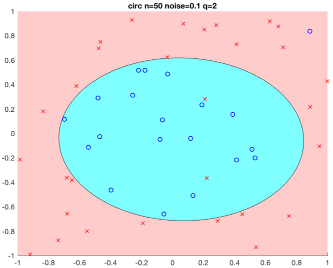
    is symmetric and positive semidefinite (Mercer condition).
  - Indeed, if $Z^\top = (z_1, \dots, z_n)$ then $K_{x_1,\dots,x_n} = ZZ^\top$ and

$$u^\top K_{x_1,\dots,x_n} u = u^\top ZZ^\top u = (Z^\top u)^\top Z^\top u = \|Z^\top u\|^2 \geq 0$$

- This is sufficient! $\mathcal{Z}$ exists as long as the Mercer conditions are satisfied.

Two non-separable cases



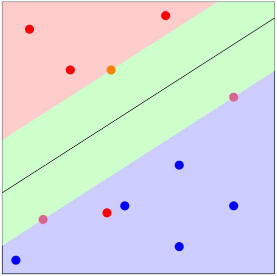


Soft-margin SVM

Non-separable case:

- Cannot guarantee

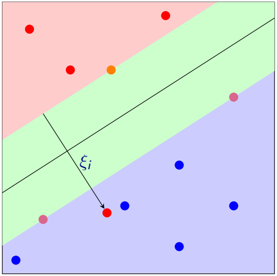
  $y_i(w^\top x_i + b) \geq 1$ for all i



Soft-margin SVM

Non-separable case:

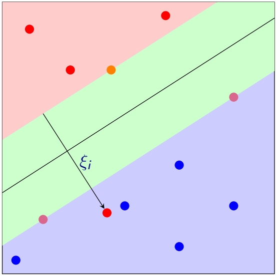
- Cannot guarantee
  $y_i(w^\top x_i + b) \geq 1$ for all i
- Relax condition: $y_i(w^\top x_i + b) \geq 1 - \xi_i$
  $\xi_i \geq 0$ is a "slack" variable



Soft-margin SVM

Non-separable case:

- Cannot guarantee
  $y_i(w^\top x_i + b) \geq 1$ for all i
- Relax condition: $y_i(w^\top x_i + b) \geq 1 - \xi_i$
  $\xi_i \geq 0$ is a "slack" variable
- Total margin violation: $\sum_{i=1}^n \xi_i$



Soft-margin SVM

Non-separable case:

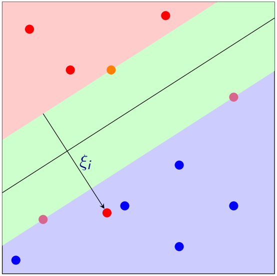
- Cannot guarantee

  $y_i(w^\top x_i + b) \geq 1$ for all i

- Relax condition: $y_i(w^\top x_i + b) \geq 1 - \xi_i$

  $\xi_i \geq 0$ is a "slack" variable

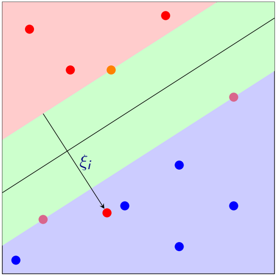
- Total margin violation: $\sum_{i=1}^n \xi_i$



Soft-margin SVM

Non-separable case:

- Cannot guarantee
  $y_i(w^\top x_i + b) \geq 1$ for all i
- Relax condition: $y_i(w^\top x_i + b) \geq 1 - \xi_i$
  $\xi_i \geq 0$ is a "slack" variable
- Total margin violation: $\sum_{i=1}^n \xi_i$



$$
\begin{array}{ll}
\text{minimize} & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^n \xi_i \\
\text{subject to} & y_i(w^\top x_i + b) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \text{ for all } i
\end{array}
$$

Parameter C provides a balance between minimizing $\|w\|^2$ (large margin) and ensuring that most samples have functional margin at least 1 (minimum number of misclassified samples)

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0 \qquad \Rightarrow \qquad w = \sum_{i=1}^{n}\alpha_i y_i x_i$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w\mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0 \qquad \Rightarrow \qquad w = \sum_{i=1}^{n}\alpha_i y_i x_i$$

$$\nabla_b\mathcal{L} = -\sum_{i=1}^{n}\alpha_i y_i = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0 \qquad \Rightarrow \qquad w = \sum_{i=1}^{n}\alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = -\sum_{i=1}^{n}\alpha_i y_i = 0 \qquad \Rightarrow \qquad \sum_{i=1}^{n}\alpha_i y_i = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w\mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0 \qquad \Rightarrow \qquad w = \sum_{i=1}^{n}\alpha_i y_i x_i$$

$$\nabla_b\mathcal{L} = -\sum_{i=1}^{n}\alpha_i y_i = 0 \qquad \Rightarrow \qquad \sum_{i=1}^{n}\alpha_i y_i = 0$$

$$\nabla_{\xi_i}\mathcal{L} = C - \alpha_i - \beta_i = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i + \sum_{i=1}^{n} \alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n} \beta_i \xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \qquad \Rightarrow \qquad w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = -\sum_{i=1}^{n} \alpha_i y_i = 0 \qquad \Rightarrow \qquad \sum_{i=1}^{n} \alpha_i y_i = 0$$

$$\nabla_{\xi_i} \mathcal{L} = C - \alpha_i - \beta_i = 0 \qquad \Rightarrow \qquad \alpha_i + \beta_i = C$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0 \qquad \Rightarrow \qquad w = \sum_{i=1}^{n}\alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = -\sum_{i=1}^{n}\alpha_i y_i = 0 \qquad \Rightarrow \qquad \sum_{i=1}^{n}\alpha_i y_i = 0$$

$$\nabla_{\xi_i} \mathcal{L} = C - \alpha_i - \beta_i = 0 \qquad \Rightarrow \qquad \alpha_i + \beta_i = C$$

$$\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w\mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0 \quad \Rightarrow \quad w = \sum_{i=1}^{n}\alpha_i y_i x_i$$

$$\nabla_b\mathcal{L} = -\sum_{i=1}^{n}\alpha_i y_i = 0 \quad \Rightarrow \quad \sum_{i=1}^{n}\alpha_i y_i = 0$$

$$\nabla_{\xi_i}\mathcal{L} = C - \alpha_i - \beta_i = 0 \quad \Rightarrow \quad \alpha_i + \beta_i = C$$

$$\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] = 0 \quad \Rightarrow \quad \alpha_i = 0 \text{ or } y_i(w^\top x_i + b) = 1 - \xi_i$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w \mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0 \quad \Rightarrow \quad w = \sum_{i=1}^{n}\alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = -\sum_{i=1}^{n}\alpha_i y_i = 0 \quad \Rightarrow \quad \sum_{i=1}^{n}\alpha_i y_i = 0$$

$$\nabla_{\xi_i} \mathcal{L} = C - \alpha_i - \beta_i = 0 \quad \Rightarrow \quad \alpha_i + \beta_i = C$$

$$\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] = 0 \quad \Rightarrow \quad \alpha_i = 0 \text{ or } y_i(w^\top x_i + b) = 1 - \xi_i$$

$$\beta_i\xi_i = 0$$

Lagrangian formulation:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i$$

minimize in primal variables w, b, ξ, maximize in dual variables $\alpha_i, \beta_i \geq 0$

KKT (Karush-Kuhn-Tucker conditions):

$$\nabla_w\mathcal{L} = w - \sum_{i=1}^{n}\alpha_i y_i x_i = 0 \quad \Rightarrow \quad w = \sum_{i=1}^{n}\alpha_i y_i x_i$$

$$\nabla_b\mathcal{L} = -\sum_{i=1}^{n}\alpha_i y_i = 0 \quad \Rightarrow \quad \sum_{i=1}^{n}\alpha_i y_i = 0$$

$$\nabla_{\xi_i}\mathcal{L} = C - \alpha_i - \beta_i = 0 \quad \Rightarrow \quad \alpha_i + \beta_i = C$$

$$\alpha_i[1 - \xi_i - y_i(w^\top x_i + b)] = 0 \quad \Rightarrow \quad \alpha_i = 0 \text{ or } y_i(w^\top x_i + b) = 1 - \xi_i$$

$$\beta_i\xi_i = 0 \quad \Rightarrow \quad \beta_i = 0 \text{ or } \xi_i = 0$$

Soft-margin SVM – dual

Minimize

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j x_i^\top x_j$$

subject to $0 \leq \alpha_i \leq C$, $\sum_{i=1}^{n} \alpha_i y_i = 0$

$b = y_i - \sum_{i=j}^{n} \alpha_j y_j x_j^\top x_i$
when $0 < \alpha_i < C$.

Parameter tuning

- How to select kernels?

Parameter tuning

- How to select kernels?
  - Kernel learning:

Parameter tuning

- How to select kernels?
  - Kernel learning:
    - ★ $K_j(x, x') = \phi_j^\top(x)\phi_j(x')$, and $K(x, x') = \sum_{j=1}^{J} \gamma_j K_j(x, x')$

Parameter tuning

- How to select kernels?
  - Kernel learning:
    - ★ $K_j(x, x') = \phi_j^\top(x)\phi_j(x')$, and $K(x, x') = \sum_{j=1}^{J} \gamma_j K_j(x, x')$
    - ★ Equivalent to having feature vector $z^\top = (\phi_1^\top, \dots, \phi_J^\top)$ and weight vector $w = (w_1, \dots, w_J)$

Parameter tuning

- How to select kernels?
  - Kernel learning:
    - ★ $K_j(x, x') = \phi_j^\top(x)\phi_j(x')$, and $K(x, x') = \sum_{j=1}^{J} \gamma_j K_j(x, x')$
    - ★ Equivalent to having feature vector $z^\top = (\phi_1^\top, \ldots, \phi_J^\top)$ and weight vector $w = (w_1, \ldots, w_J)$
    - ★ Penalize to limit the number of kernels used: instead of minimizing $\|w\|^2$, minimize $\left(\sum_{j=1}^{J} \|w_j\|^p\right)^{2/p}$ – mixed L_1-L_2 penalty for $p = 1$.

Parameter tuning

- How to select kernels?
  - Kernel learning:
    - ★ $K_j(x, x') = \phi_j^\top(x)\phi_j(x')$, and $K(x, x') = \sum_{j=1}^{J} \gamma_j K_j(x, x')$
    - ★ Equivalent to having feature vector $z^\top = (\phi_1^\top, \ldots, \phi_J^\top)$ and weight vector $w = (w_1, \ldots, w_J)$
    - ★ Penalize to limit the number of kernels used: instead of minimizing $\|w\|^2$, minimize $\left(\sum_{j=1}^{J} \|w_j\|^p\right)^{2/p}$ – mixed L_1-L_2 penalty for $p = 1$.
- Training time with QP typically $\Theta(n^3)$–can be much faster with GD/SGD with an approximate solution

Gaussian–RBF kernels

Gaussian RBF (radial basis function) kernel:

$$K(x, x') = \exp\Big(-\gamma \underbrace{\|x - x'\|^2}_{\textit{radial}} \Big)$$

Gaussian–RBF kernels

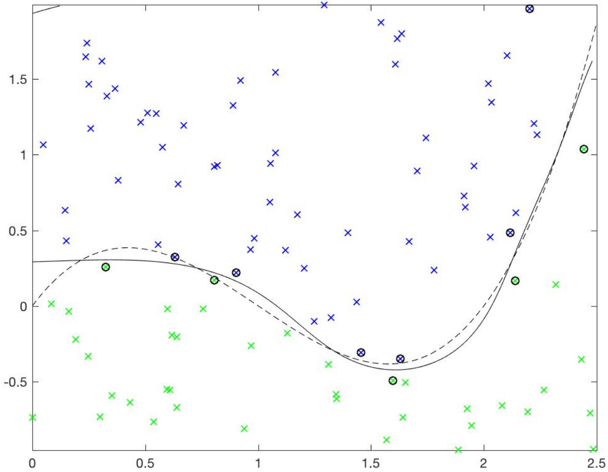
Gaussian RBF (radial basis function) kernel:

$$K(x, x') = \exp\Big(-\gamma \underbrace{\|x - x'\|^2}_{radial} \Big)$$

SVM predictor

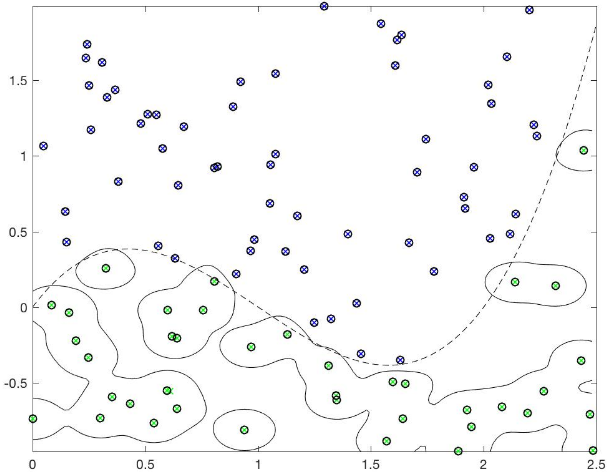
$$g(x) = \text{sign}\left(\sum_{x_i \text{ is SV}} \alpha_i y_i K(x_i, x) + b \right) = \text{sign}\left(\sum_{x_i \text{ is SV}} \alpha_i y_i e^{-\gamma \|x - x_i\|^2} + b \right)$$

RBF-kernel width



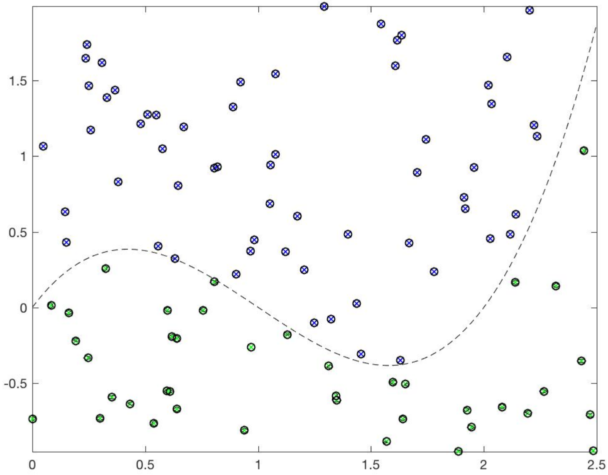
$$\gamma = 1$$

RBF-kernel width



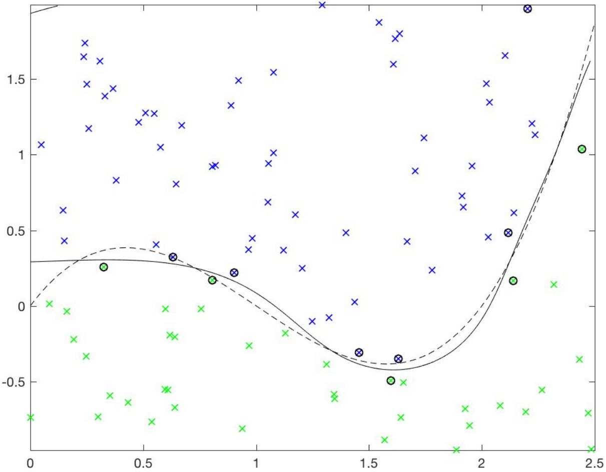
$\gamma = 10$

RBF-kernel width



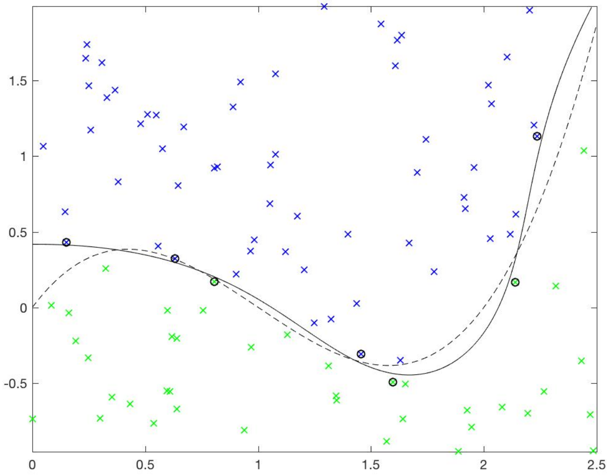
$$\gamma = 100$$

RBF-kernel width



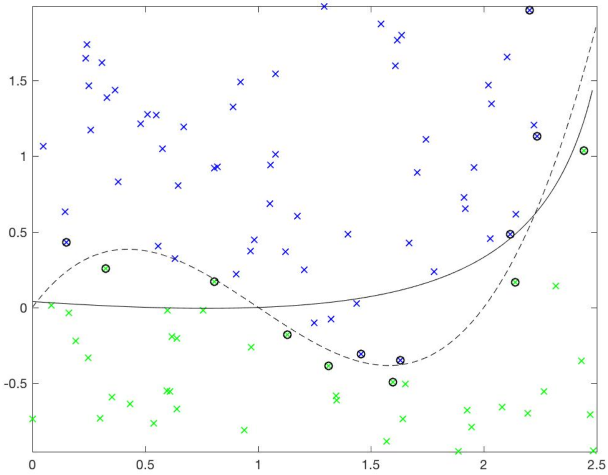
$\gamma = 1$

RBF-kernel width



$$\gamma = 0.1$$

RBF-kernel width



$$\gamma = 0.01$$