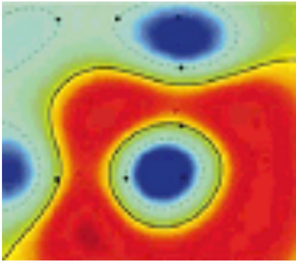
WiSe 2024/25

Machine Learning 1/1-X

Technische Universität Berlin



Lecture 5

Support Vector Machines

Support Vector Machines

Outline

- Lagrange Duality
- KKT optimality conditions
- Large margin classifiers
- Hard-margin SVM (Primal / Dual)
- Soft-margin SVM (Primal)
- Kernel SVM
- SVM and Hinge Loss
- SVM beyond Classification
- Applications

Lagrange Duality (1)

- We consider optimization problem in **canonical** form:

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f_0(\boldsymbol{x}) \\ \text{subject to} \quad & f_i(\boldsymbol{x}) \leqslant 0, \quad i = 1, \ldots, m \\ & h_i(\boldsymbol{x}) = 0, \quad i = 1, \ldots, p \end{aligned}$$

- The **Lagrange function** $\mathcal{L}\colon \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ is defined as a weighted sum of the objective and constraint functions:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i f_i(\boldsymbol{x}) + \sum_{i=1}^{p} \mu_i h_i(\boldsymbol{x}),$$

where $\boldsymbol{x}$ is called **primal** and $(\boldsymbol{\lambda}, \boldsymbol{\mu})$ the **dual** variables.

Lagrange Duality (2)

- The (Lagrange) **dual function** $g\colon \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ is defined as:

$$g(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \inf_{\boldsymbol{x} \in \text{domain}(f_0)} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}).$$

- The (convex!) optimization problem

$$\begin{aligned} \underset{(\boldsymbol{\lambda},\boldsymbol{\mu})}{\text{maximize}} \quad & g(\boldsymbol{\lambda}, \boldsymbol{\mu}) \\ \text{subject to} \quad & \boldsymbol{\lambda} \succeq \mathbf{0} \end{aligned}$$

  is called the (Lagrange) **dual problem**.

- The inequality $d^* \leqslant p^*$ always holds, where p^*, d^* are the optimal values of the primal and dual problem, respectively.

Lagrange Duality (3)

- The (Lagrange) **dual function** $g\colon \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ is defined as:

$$g(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \inf_{\boldsymbol{x} \in \text{domain}(f_0)} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}).$$

- The (convex!) optimization problem

$$\begin{aligned} \underset{(\boldsymbol{\lambda}, \boldsymbol{\mu})}{\text{maximize}} \quad & g(\boldsymbol{\lambda}, \boldsymbol{\mu}) \\ \text{subject to} \quad & \boldsymbol{\lambda} \succeq \mathbf{0} \end{aligned}$$

  is called the (Lagrange) **dual problem**.

- The inequality $d^* \leqslant p^*$ always holds, where p^*, d^* are the optimal values of the primal and dual problem, respectively. We refer to the difference $p^* - d^*$ as duality gap. In the case $p^* = d^*$ we talk about **strong duality**.

Karush–Kuhn–Tucker (KKT) Conditions

Theorem: Optimality Conditions

- For any optimization problem with differentiable objective and constraint functions for which strong duality obtains, any pair of primal and dual optimal $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ must satisfy KKT-conditions:

$$
\begin{aligned}
&\nabla_x \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = 0, && \text{(stationarity)}\\
&f_i(\boldsymbol{x}^*) \leqslant 0, && \text{(primal feasibility)}\\
&h_i(\boldsymbol{x}^*) = 0, && \text{(primal feasibility)}\\
&\lambda_i^* \geqslant 0, && \text{(dual feasibility)}\\
&\lambda_i^* \cdot f_i(\boldsymbol{x}^*) = 0 && \text{(complementary slackness)}
\end{aligned}
$$

- For any convex problem, the KKT-conditions are sufficient for $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ to be optimal with zero duality gap.

Slater's Theorem

We say that a convex optimization problem

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f_0(\boldsymbol{x}) \\ \text{subject to} \quad & f_i(\boldsymbol{x}) \leqslant 0, \quad i = 1, \dots, m \\ & h_i(\boldsymbol{x}) = 0, \quad i = 1, \dots, p \end{aligned}$$

satisfies the Slater's condition if there exists a strictly feasible point $\boldsymbol{x}$, i.e., $f_i(\boldsymbol{x}) < 0$ and $h_j(\boldsymbol{x}) = 0$ for all i, j.

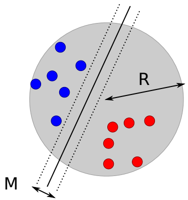
Slater's Theorem

For any convex problem for which Slater's condition holds, the KKT-conditions provide necessary and sufficient condition for $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ to be primal and dual optimal with zero duality gap.

Linear Classifiers with Margin

- Let $\phi(\mathbf{x})$ be some feature map mapping the data in $\mathbb{R}^d$ to some feature space $\mathbb{R}^D$.
- We consider functions of the type $f_\theta : \mathbb{R}^d \to \mathbb{R}$ with $f_\theta(\mathbf{x}) = \text{sign}(\mathbf{w}^\top \phi(\mathbf{x}) + b)$, where $\theta = (\mathbf{w}, b)$ denotes the parameters.
- We consider functions that classify with some margin:

$$\mathcal{F} = \Big\{ f_\theta \colon \theta \in \mathbb{R}^{d+1} \;, \quad \forall_{i=1}^n : \; |\mathbf{w}^\top \phi(\mathbf{x}_i) + b| \geq 1 \;, \; \frac{2}{\|\mathbf{w}\|} = M \Big\}$$

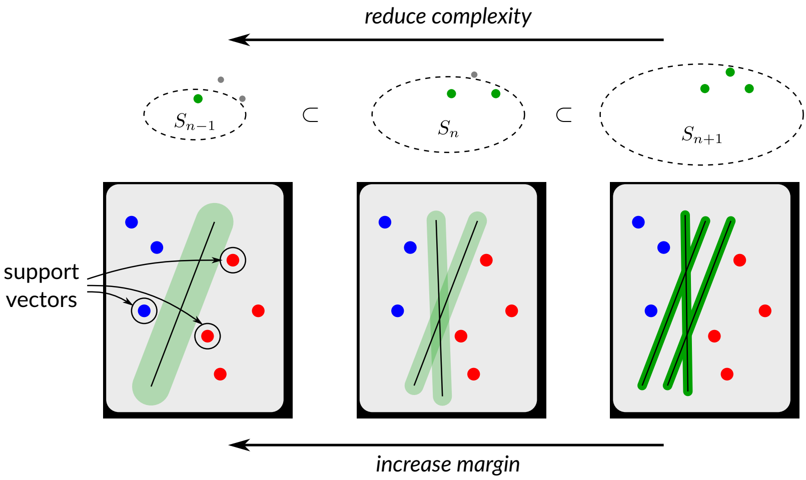


- The VC-dimension can be bounded as

$$h_\mathcal{F} \leq \min \Big\{ d+1 \,, \, \Big\lceil \frac{4R^2}{M^2} \Big\rceil + 1 \Big\}$$

(cf. Burges 1998).

Support Vector Machine (SVM)

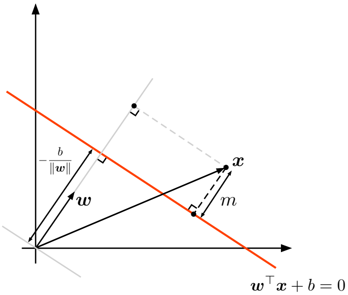


Hard-Margin SVM (Derivation)

- Given data $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$ with $y_i \in \{-1, 1\}$, we want to maximize the separation margin of the linear classifier $y(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$:

$$\underset{\boldsymbol{w},b}{\text{maximize}} \ \frac{1}{\|\boldsymbol{w}\|} \min_{i=1,\ldots,n} y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$$

- Observation: rescaling $\boldsymbol{w} \mapsto k\boldsymbol{w}$ and $\boldsymbol{b} \mapsto k\boldsymbol{b}$, $k \neq 0$ results in the same objective value. We can use this fact to set $\min_{i=1,\ldots,n} y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = 1$.



$$\boxed{m = \frac{\boldsymbol{w}^\top \boldsymbol{x}}{\|\boldsymbol{w}\|} - \left(-\frac{b}{\|\boldsymbol{w}\|}\right) = \frac{\boldsymbol{w}^\top \boldsymbol{x} + b}{\|\boldsymbol{w}\|}}$$

Hard-Margin SVM (Derivation)

- This gives the following optimization problem

$$\underset{\boldsymbol{w},b}{\text{maximize}} \ \frac{1}{\|\boldsymbol{w}\|} \quad \text{subject to} \quad \min_{i=1,\ldots,n} y_i(\boldsymbol{w}^\top \boldsymbol{x_i} + b) = 1$$

  or equivalently

$$\underset{\boldsymbol{w},b}{\text{minimize}} \ \frac{1}{2}\|\boldsymbol{w}\|^2 \quad \text{subject to} \quad y_i(\boldsymbol{w}^\top \boldsymbol{x_i} + b) \geqslant 1, i = 1, \ldots, n$$

- Replacing $\boldsymbol{x}$ by (non-linear) features $\phi(\boldsymbol{x})$ gives the (primal) **hard-margin** formulation of the Support Vector Machine:

$$\underset{\boldsymbol{w},b}{\text{minimize}} \ \frac{1}{2}\|\boldsymbol{w}\|^2 \quad \text{subject to} \quad y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b) \geqslant 1, i = 1, \ldots, n$$

Hard-Margin SVM (Primal Problem)

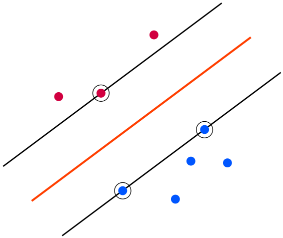
1. The classifier with largest margin between the positive and negative data points $\{(\boldsymbol{x}_i, y_i)\}_{i=1,\dots,n}$ can be obtained by solving a convex optimization problem:

$$\boxed{\begin{array}{ll} \underset{\boldsymbol{w},b}{\text{minimize}} & \frac{1}{2}\|\boldsymbol{w}\|^2 \\ \text{subject to} & y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b) \geqslant 1, \quad i = 1, \dots, n \end{array}}$$

2. The decision function $f : \mathbb{R}^d \to \{1, -1\}$ is given by

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^\top \phi(\mathbf{x}) + b)$$

and it can be used to classify new data.



3. Data points $(\boldsymbol{x}_i, y_i)$ where a corresponding constraint is active, i. e., $y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b) = 1$ are called **support vectors**.

Deriving the Dual of the Hard-Margin SVM

- Consider the hard-margin formulation of SVM

$$\underset{\boldsymbol{w},b}{\text{minimize}} \;\; \frac{1}{2}\|\boldsymbol{w}\|^2 \quad \text{subject to} \quad y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b) \geqslant 1, \quad i = 1, \dots, n$$

- Write the Lagrangian

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

- Compute the dual function $g(\boldsymbol{\alpha}) = \inf_{\boldsymbol{w},b} \mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha})$:

$$g(\boldsymbol{\alpha}) = \begin{cases} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle, & \text{if } \sum_{i=1}^{n} \alpha_i y_i = 0 \\ -\infty & \text{else} \end{cases}$$

  where we used the fact that $\mathcal{L}$ is strictly convex and

$$\nabla_{\boldsymbol{w}} \mathcal{L} = \mathbf{0} \;\Rightarrow\; \boldsymbol{w} = \sum_{i=1}^{n} \alpha_i y_i \phi(\boldsymbol{x}_i) \quad \text{and} \quad \frac{\partial \mathcal{L}}{\partial b} = 0 \;\Rightarrow\; \sum_{i=1}^{n} \alpha_i y_i = 0$$

Hard-Margin SVM (Dual Problem)

- The dual problem has the following form:

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}\succeq 0}{\text{maximize}} & \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle \\ \text{subject to} & \sum_{i=1}^{n} \alpha_i y_i = 0 \end{array}$$

- Due to the relationship $\boldsymbol{w} = \sum \alpha_i y_i \phi(\boldsymbol{x}_i)$ the decision function is given as

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i y_i \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x})\rangle + b$$

- How do we find the bias b? Note that for each support vector $\boldsymbol{x}_i \in S$, it holds $y_i \cdot f(\boldsymbol{x}_i) = 1$, where S denotes the set of support vectors. Here, it is enough to use one arbitrary support vector to compute b. However, the following provides numerically more stable solution:

$$b = \frac{1}{|S|}\sum_{i\in S}(y_i - \sum_{j\in S}\alpha_j y_j \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j)\rangle)$$

Hard-Margin SVM (Dual Problem)

- On the previous slide we saw how to compute bias in the dual formulation:

$$b = \frac{1}{|S|} \sum_{i \in S} (y_i - \sum_{j \in S} \alpha_j y_j \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle)$$

  The remaining question here is how to find S.

- Based on the complementary slackness in the KKT-conditions

$$\alpha_i \cdot (1 - y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b)) = 0$$

  we conclude

$$\boldsymbol{x}_i \text{ is a support vector } \Leftrightarrow \alpha_i > 0.$$

Soft-Margin SVM (Primal Problem)

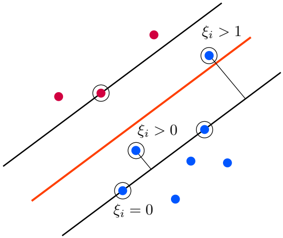
ξ_i（松弛变量）：用于度量样本 x_i 违反间隔的程度，确保数据点可以在某种程度上被错误分类。

C（正则化参数）：权衡间隔大小和误分类代价。**C 越大，模型对误分类的容忍度越低（趋向 Hard Margin）；C 越小，模型允许更多误分类（更趋向 Soft Margin）。**

1. If the data $\{(\boldsymbol{x}_i, y_i)\}_{i=1,\dots,n}$ is not separable (e.g. due to noise), we introduce *slack variables* $(\xi_i)_i$ that allows for data points to violate the margin constraints at the cost of additional penalty. We refer to this formulation as **soft-margin** SVM:

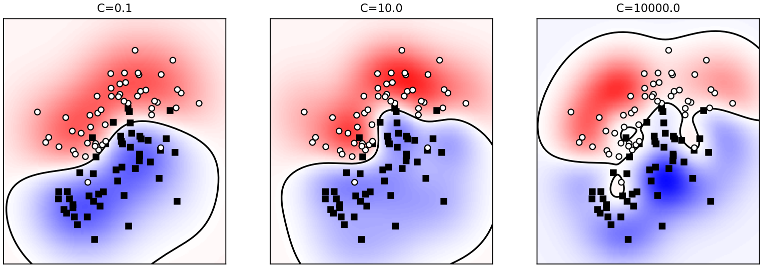
$$\begin{aligned} \underset{\boldsymbol{w},b,\boldsymbol{\xi}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{subject to} \quad & y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b) \geqslant 1 - \xi_i, \quad && i = 1, \dots, n \\ & \xi_i \geqslant 0, && i = 1, \dots, n \end{aligned}$$

2. Here, $C \in (0, \infty)$ is a regularization constant controlling the trade-off between the margin size and the constraint violation. For $C \to \infty$ we recover the hard-margin formulation.
3. Data points $(\boldsymbol{x}_i, y_i)$ for which either $\xi_i > 0$ or $y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b) = 1$ holds are called **support vectors**.



Effect of the parameter C

The larger the parameter C the more the decision boundary is forced to correctly classify every data point. For $C \to \infty$ we recover the hard-margin formulation. For $C \to 0$ the robustness of "correctly" classified points increases.



Kernel Functions

Definition (Kernel function)

A kernel is a function $\kappa \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ that for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ satisfies

$$\kappa(\boldsymbol{x}, \boldsymbol{y}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{y}) \rangle$$

where $\phi \colon \mathcal{X} \to \mathcal{F}$ is a mapping from some $\mathcal{X}$ to a Hilbert space $(\mathcal{F}, \langle \cdot, \cdot \rangle)$.

Definition (Finitely positive semi-definite functions)

A function $\kappa \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ satisfies the finitely positive semi-definite property if it is symmetric and for which the matrices formed by restriction to any finite subset of the space $\mathcal{X}$ are positive semi-definite.

Theorem (Kernel matrices)

The kernel functions satisfy the finitely positive semi-definite property. That is, the corresponding kernel matrices are positive semi-definite.

Examples of Kernels

Observation: In the SVM dual form, we never need to access the feature map $\phi(\cdot)$ explicitly. Instead, we can always express computations in terms of the kernel function.

Examples of commonly used kernels satisfying the Mercer property are:

Linear	$k(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle$	
Polynomial	$k(\mathbf{x}, \mathbf{x}') = (\langle \mathbf{x}, \mathbf{x}' \rangle + \beta)^\gamma$	$\beta \in \mathbb{R}_{\geq 0}, \gamma \in \mathbb{N}$
Gaussian	$k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2)$	$\gamma \in \mathbb{R}_{>0}$

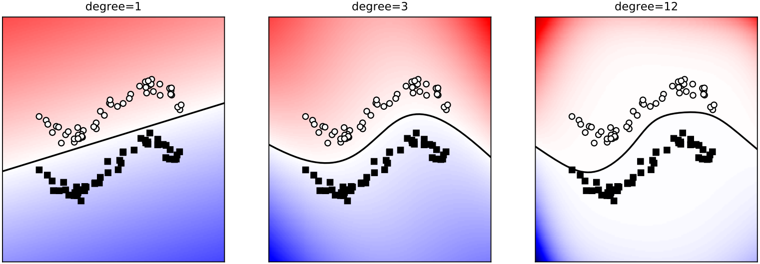
Note: The feature map associated to the Gaussian kernel is infinite-dimensional. However, in the dual form, we never need to access it for training and prediction, and we only need the kernel function.

SVM with Polynomial Kernel

The SVM decision function

$$g(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y_i \underbrace{(\langle \mathbf{x}_i, \mathbf{x} \rangle + a)^\gamma}_{k(\mathbf{x}_i, \mathbf{x})} + b$$

is polynomial of degree γ.



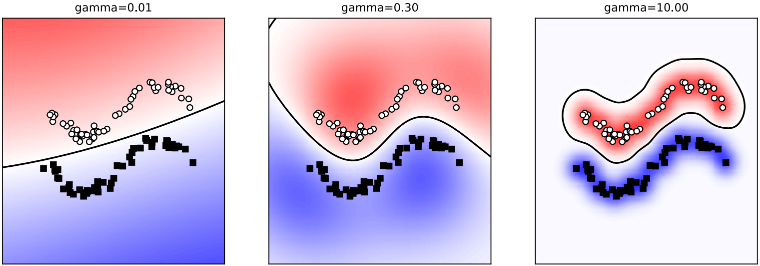
Observation: The polynomial decision function has very high value in the extrapolation regime.

SVM with Gaussian Kernel

The SVM decision function

$$g(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y_i \underbrace{\exp(-\gamma \|\mathbf{x}_i - \mathbf{x}\|^2)}_{k(\mathbf{x}_i, \mathbf{x})} + b$$

is a superposition of ‘bump’ functions. Intermediate values of γ usually give the best results.



More Advanced Kernels

More advanced kernels can be used to incorporate prior knowledge into the classifier, and to let the algorithm operate on non-vector data.

Examples:

- Bag-of-words kernels (text classification, image recognition)
- Tree kernels (text classification)
- Weighted degree kernel (bioinformatics)
- Graph kernels

Some of these kernels will be studied in Machine Learning 2.

Computational Requirements of SVMs

Computation required by the **primal**

- Optimization of a vector $(\mathbf{w}, b)$ in $\mathbb{R}^{d+1}$ subject to n constraints.
- Primal is infeasible when d is very large (e.g. infinite-dimensional).

Computation required by the **dual**

- Computation of a kernel matrix K of size $n \times n$ with $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$
- Dual is infeasible when n is large (e.g. 1 million data points).

What if both n and d are large?

- *Random features* (approximates infinite-dimensional feature maps through randomization).
- *Neural networks* (learns an efficient feature map of size $\ll D$).

SVM and Hinge Loss

Recall that our decision boundary is defined by $g(\mathbf{x}) = \mathbf{w}^\top \phi(\mathbf{x}_i) + b$, and therefore, we can write the constraint as:

$$\xi_i \geq 1 - y_i \cdot g(\mathbf{x})$$

Together with $\xi_i \geqslant 0$ this implies

$$\xi_i = \max(0, 1 - y_i \cdot g(\mathbf{x}))$$

By injecting it directly in the objective we get an unconstrained optimization problem

$$\min_{\mathrm{w}, b, \boldsymbol{\xi}} \quad \underbrace{\frac{1}{2}\|\mathbf{w}\|^2}_{\mathcal{E}_{\text{reg}}} + \underbrace{C \sum_{i=1}^{N} \overbrace{\max\left(0, 1 - y_i \cdot g(\mathbf{x})\right)}^{\ell_{\text{hinge}}(y, g(\mathbf{x}))}}_{\mathcal{E}_{\text{emp}}}$$

i.e. we optimize the sum of a fitting term and a regularization term (as we do e.g. for logistic regression but with a different loss function).

SVM Beyond Classification

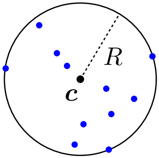
Support vector data description (SVDD):

Consider an *unlabeled* dataset $\mathcal{D} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$. We would like to find the smallest enclosing sphere of the data, i.e.

$$\min_{R,\mathbf{c}} \ R^2$$

subject to the constraints

$$\forall_{i=1}^n : \ \|\phi(\mathbf{x}_i) - \mathbf{c}\|^2 \leq R^2.$$



We consider a point to be anomalous if the point exceeds a certain distance from the center $\mathbf{c}$, i.e.

$$f(\mathbf{x}) = \mathrm{sign}(\|\phi(\mathbf{x}) - \mathbf{c}\| - \tau)$$

Note: Like for the SVM, the SVDD algorithm also has a dual form, it can expressed in terms of kernels, and we can add slack variables to it.

SVM Beyond Classification

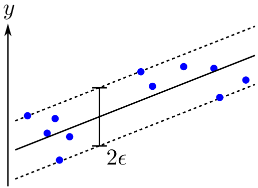
Support vector regression (SVR):

Consider a *labeled* regression dataset $\mathcal{D} = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ with $y_i \in \mathbb{R}$. We would like to build the regression model

$$\min_{\mathrm{w},b} \|\mathbf{w}\|^2$$

subject to the constraints

$$\forall_{i=1}^{n} : \ |w^\top \phi(\mathbf{x}_i) + b - y_i| \leq \epsilon$$



Note: Like for the SVM, support vector regression has a dual form, it can be kernelized, and we can add slack variables to it. With the slack variables, the constraints above can also be seen as applying an ϵ-sensitive loss.

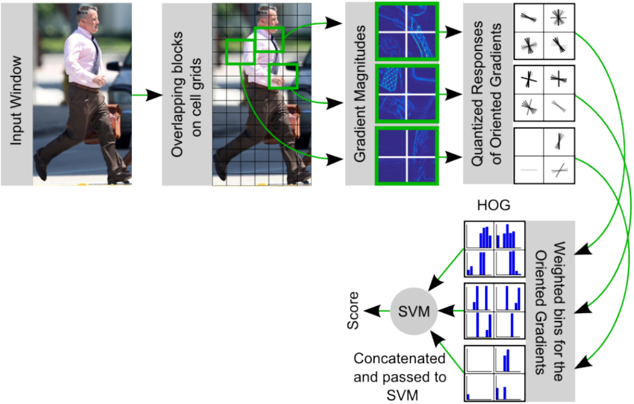
SVM for Text Categorization

Precision/recall-breakeven point on the ten most frequent Reuters categories and microaveraged performance over all Reuters categories.

	Bayes	Rocchio	C4.5	k-NN	SVM (poly) degree $d =$ 1	2	3	4	5	SVM (rbf) width $\gamma =$ 0.6	0.8	1.0	1.2
earn	95.9	96.1	96.1	97.3	98.2	98.4	**98.5**	98.4	98.3	**98.5**	98.5	98.4	98.3
acq	91.5	92.1	85.3	92.0	92.6	94.6	**95.2**	95.2	95.3	95.0	95.3	95.3	**95.4**
money-fx	62.9	67.6	69.4	78.2	66.9	72.5	75.4	74.9	**76.2**	74.0	75.4	**76.3**	75.9
grain	72.5	79.5	89.1	82.2	91.3	93.1	**92.4**	91.3	89.9	**93.1**	91.9	91.9	90.6
crude	81.0	81.5	75.5	85.7	86.0	87.3	88.6	**88.9**	87.8	**88.9**	89.0	88.9	88.2
trade	50.0	77.4	59.2	77.4	69.2	75.5	76.6	77.3	**77.1**	76.9	78.0	**77.8**	76.8
interest	58.0	72.5	49.1	74.0	69.8	63.3	67.9	73.1	**76.2**	74.4	75.0	**76.2**	76.1
ship	78.7	83.1	80.9	79.2	82.0	85.4	86.0	**86.5**	86.0	**85.4**	86.5	87.6	87.1
wheat	60.6	79.4	85.5	76.6	83.1	84.5	85.2	**85.9**	83.8	**85.2**	85.9	85.9	85.9
corn	47.3	62.2	87.7	77.9	86.0	86.5	85.3	**85.7**	83.9	**85.1**	85.7	85.7	84.5
microavg.	**72.0**	**79.9**	**79.4**	**82.3**	84.2	85.1	85.9	86.2	85.9	86.4	86.5	86.3	86.2
					combined: **86.0**					combined: **86.4**			

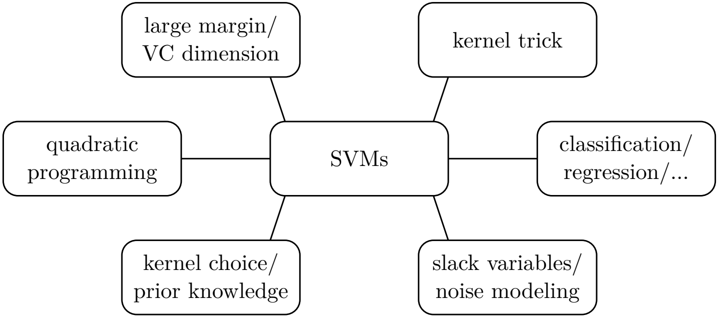
Source: Joachims et al. 1998, Text categorization with Support Vector Machines: Learning with many relevant features

SVM/HoG for Pedestrian Detection



Source: Dutta 2015, Pedestrian Detection using HOG and SVM in Automotives

Visual Summary



SVM for ECG-based arrhythmia Detection

Wang et al. (2014), Hardware Specialization in Low-power Sensing Applications to Address Energy and Resilience