Lecture 9: Learning max-sum classifier

Vojtěch Franc

April 23, 2015

7.C: Learning max-sum classifier from separable examples.

8.A: Learning two-class linear classifier from non-separable examples by SVM.

XEP33SML – Structured Model Learning, Summer 2015

7.C: Max-sum classifier

Setting:

- $(\mathcal{V}, \mathcal{E})$ is undirected graph; $\mathcal{V}$ are parts and $\mathcal{E} \subseteq \binom{|\mathcal{V}|}{2}$ pairs of related parts
- each part $v \in \mathcal{V}$ described by observation $x \in \mathcal{X}$ and label $y \in \mathcal{Y}$; $\mathcal{X}$ and $\mathcal{Y}$ are finite
- $q_v \colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ quality of label y_v given x_v; $\boldsymbol{q} = (g_v(x, y) \in \mathbb{R} \mid x \in \mathcal{X}, y \in \mathcal{Y}, v \in \mathcal{V})$
- $g_{vv'} \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ quality a label pair $(y_v, y_{v'})$;
  $\boldsymbol{g} = (g_{vv'}(y, y') \in \mathbb{R} \mid (y, y') \in \mathcal{Y}^2, \{v, v'\} \in \mathcal{E})$

Max-sum classifier: Given observations $\boldsymbol{x} = (x_v \in \mathcal{X} \mid v \in \mathcal{V}) \in \mathcal{X}^{\mathcal{V}}$, the max-sum classifier $h \colon \mathcal{X}^{\mathcal{V}} \to \mathcal{Y}^{\mathcal{V}}$ returns labeling $\boldsymbol{y} = (y_v \in \mathcal{Y} \mid v \in \mathcal{V}) \in \mathcal{Y}^{\mathcal{V}}$ with the maximal overall quality

$$h(\boldsymbol{x}; \boldsymbol{q}, \boldsymbol{g}) \in \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} f(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{q}, \boldsymbol{g})$$

where

$$f(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{q}, \boldsymbol{g}) = \sum_{v \in \mathcal{V}} q_v(x_v, y_v) + \sum_{\{v, v'\} \in \mathcal{E}} g_{vv'}(y_v, y_{v'})$$

The max-sum classifier is an instance of the linear classifier since $f(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{q}, \boldsymbol{g}) = \langle \boldsymbol{\Psi}(\boldsymbol{x}, \boldsymbol{y}), \boldsymbol{w} \rangle$ where $\boldsymbol{w} = (\boldsymbol{q}, \boldsymbol{g})$ and $\boldsymbol{\Psi} \colon \mathcal{X}^{\mathcal{Y}} \times \mathcal{Y}^{\mathcal{V}} \to \mathbb{R}^{|\mathcal{Y}||\mathcal{V}| + |\mathcal{E}||\mathcal{Y}|^2}$ is constructed appropriately.

7.C: Relation between Max-sum classifier and Gibbs distribution

- $(\mathcal{V}, \mathcal{E})$ is undirected graph
- $\{(X_v, Y_v) \mid v \in \mathcal{V}\}$ is a field of random variables taking values from $(x_v, y_v) \in X \times \mathcal{Y}, v \in \mathcal{V}$
- the random variables are distributed according to the Gibbs distribution

$$
\begin{aligned}
p_{\boldsymbol{q},\boldsymbol{g}}(\boldsymbol{x}, \boldsymbol{y}) &= \frac{1}{Z(\boldsymbol{q}, \boldsymbol{g})} \exp\left(\sum_{v \in \mathcal{V}} q_v(x_v, y_v) + \sum_{\{v,v'\} \in \mathcal{E}} g_{vv'}(y_v, y_{v'}) \right) \\
&= \frac{1}{Z(\boldsymbol{q}, \boldsymbol{g})} \exp f(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{q}, \boldsymbol{g})
\end{aligned}
$$

- The optimal (Bayes) classifier minimizing the expected risk under the 0/1-loss

$$
R(h) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y}) \sim p_{\boldsymbol{q},\boldsymbol{g}}} [\![\boldsymbol{y} \neq h(\boldsymbol{x})]\!]
$$

is the max-sum classifier

$$
h(\boldsymbol{x}; \boldsymbol{q}, \boldsymbol{g}) \in \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} f(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{q}, \boldsymbol{g})
$$

7.C: Learning max-sum classifier from linearly separable examples

Task: Given linearly separable training set $\mathcal{T} = \{(\boldsymbol{x}^i, \boldsymbol{y}^i) \in \mathcal{X}^{\mathcal{V}} \times \mathcal{Y}^{\mathcal{V}} \mid i \in \mathcal{I} = \{1, \ldots, m\}\}$ find quality functions $\boldsymbol{q}$, $\boldsymbol{g}$ of the max-sum classifier such that

$$\boldsymbol{y}^i = h(\boldsymbol{x}^i; \boldsymbol{q}, \boldsymbol{g}) = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} \left[\sum_{v \in \mathcal{V}} q_v(x_v^i, y_v) + \sum_{\{v,v'\} \in \mathcal{E}} g_{vv'}(y_v, y_{v'}) \right], \qquad i \in \mathcal{I}\,.$$

The max-sum problem $\mathcal{P} = (\mathcal{E}, \mathcal{V}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ associated with the classification $h(\boldsymbol{x}; \boldsymbol{q}, \boldsymbol{g})$ is tractable if:

1. $(\mathcal{V}, \mathcal{E})$ is acyclic graph
2. $\mathcal{Y}$ is fully ordered and $-g_{vv'}$, $\{v, v'\} \in \mathcal{E}$ are submodular w.r.t the ordering: for each $(y_v, y'_v, y_{v'}, y'_{v'}) \in \mathcal{Y}^4$ such that $y_v > y'_v$ and $y_{v'} > y'_{v'}$ it following inequality holds

$$g_{vv'}(y_v, y_{v'}) + g_{vv'}(y'_v, y'_{v'}) \leq g_{vv'}(y_v, y'_{v'}) + g_{vv'}(y'_v, y_{v'})$$

3. $\mathcal{P} = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ have a strictly trivial equivalent, that is, the LP relaxation is tight and the max-sum problem has unique solution

7.C: LP relaxation of max-sum problem (recall Lecture 3, section 3)

The max-sum problem

$$\boldsymbol{y}^* \in \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^\mathcal{V}} \left[\sum_{v\in\mathcal{V}} q_v(x_v, y_v) + \sum_{\{v,v'\}\in\mathcal{E}} g_{v,v'}(y_v, y_{v'}) \right]$$

The Schlesinger's LP relaxation of the max-sum problem reads

$$\boldsymbol{\mu}^* = \operatorname*{argmax}_{\boldsymbol{\mu}\in\mathbb{R}^{|\mathcal{V}||\mathcal{Y}|+|\mathcal{E}||\mathcal{Y}|^2}} \left[\sum_{v\in\mathcal{V}} \sum_{y\in\mathcal{Y}} \mu_v(y) q_v(x_v, y_v) + \sum_{\{v,v'\}\in\mathcal{E}} \sum_{(y,y')\in\mathcal{Y}^2} \mu_{vv'}(y,y') g_{vv'}(y,y') \right]$$

subject to

$$\sum_{y'\in\mathcal{Y}} \mu_{vv'}(y,y') = \mu_v(y), \{v,v'\}\in\mathcal{E}, y\in\mathcal{Y}, \qquad \sum_{y\in\mathcal{Y}} \mu_v(y) = 1, v\in\mathcal{V}, \qquad \boldsymbol{\mu}\geq\boldsymbol{0}$$

Note that adding a constraint $\boldsymbol{\mu}\in\{0,1\}$ makes the LP relaxation equivalent to the original max-sum problem.

7.C: Dual of LP relaxation

The Lagrange dual of the (primal) LP relaxation can be written as an unconstrained problem

$$\boldsymbol{\varphi}^* = \operatorname*{argmin}_{\boldsymbol{\varphi}} U(\boldsymbol{x}, \boldsymbol{q}^{\boldsymbol{\varphi}}, \boldsymbol{g}^{\boldsymbol{\varphi}}) = \operatorname*{argmin}_{\boldsymbol{\varphi}} \left[\sum_{v \in \mathcal{V}} \max_{y \in \mathcal{Y}} q_v^{\boldsymbol{\varphi}}(x_v, y) + \sum_{\{v,v'\} \in \mathcal{E}} \max_{(y,y') \in \mathcal{Y}^2} g_{vv'}^{\boldsymbol{\varphi}}(y, y') \right]$$

where $\boldsymbol{\varphi} \in \mathbb{R}^{2|\mathcal{E}||\mathcal{Y}|}$ is a vector of dual variables $\varphi_{vv'} \colon \mathcal{Y} \to \mathbb{R}$, $\varphi_{v'v} \colon \mathcal{Y} \to \mathbb{R}$, $\{v, v'\} \in \mathcal{E}$ and

$$\begin{aligned} g_{vv'}^{\boldsymbol{\varphi}}(y, y') &= g_{vv'}(y, y') + \varphi_{vv'}(y) + \varphi_{v'v}(y'), & \{v, v'\} \in \mathcal{E}, y, y' \in \mathcal{Y} \\ q_v^{\boldsymbol{\varphi}}(y) &= q_v(y) - \sum_{v' \in \mathcal{N}(v)} \varphi_{vv'}(y), & v \in \mathcal{V}, y \in \mathcal{Y} \end{aligned}$$

Questions:

1. Is the LP relaxation tight, i.e., does it hold that $U(\boldsymbol{x}, \boldsymbol{q}^{\boldsymbol{\varphi}^*}, \boldsymbol{g}^{\boldsymbol{\varphi}^*}) = f(\boldsymbol{x}, \boldsymbol{y}^*, \boldsymbol{q}, \boldsymbol{g})$?

2. If yes how to get the labels $\boldsymbol{y}^*$?

7.C: Interpretation of the dual of LP relaxation

Definition 1. *Problems* $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ *and* $P' = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}', \boldsymbol{g}', \boldsymbol{x})$ *are equivalent if* $f(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{q}, \boldsymbol{g}) = f(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{q}', \boldsymbol{g}')$ *for all* $\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}$.

Re-parametrization: Let $P^{\boldsymbol{\varphi}} = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}^{\boldsymbol{\varphi}}, \boldsymbol{g}^{\boldsymbol{\varphi}}, \boldsymbol{x})$ be the max-sum problem constructed from $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ by the re-reparametrization

$$
\begin{aligned}
g^{\boldsymbol{\varphi}}_{vv'}(y, y') &= g_{vv'}(y, y') + \varphi_{vv'}(y) + \varphi_{v'v}(y'), && \{v, v'\} \in \mathcal{E}, y, y' \in \mathcal{Y} \\
q^{\boldsymbol{\varphi}}_{v}(y) &= q_v(y) - \sum_{v' \in \mathcal{N}(v)} \varphi_{vv'}(y), && v \in \mathcal{V}, y \in \mathcal{Y}
\end{aligned}
\tag{R}
$$

Proposition 1. *Two max-sum problems* $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ *and* $P^{\boldsymbol{\varphi}} = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}^{\boldsymbol{\varphi}}, \boldsymbol{g}^{\boldsymbol{\varphi}}, \boldsymbol{x})$ *related by the re-parametrization* (R) *are equivalent.*

PROOF: It is seen from substituting (R) to $f(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{q}, \boldsymbol{g}) = f(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{q}^{\boldsymbol{\varphi}}, \boldsymbol{g}^{\boldsymbol{\varphi}})$. ∎

Interpretation of the dual of LP relaxation: In the class of equivalent problems $\{P^{\boldsymbol{\varphi}} \mid \boldsymbol{\varphi} \in \mathbb{R}^{2|\mathcal{E}||\mathcal{Y}|}\}$ find the one with minimal energy

$$
U(\boldsymbol{x}, \boldsymbol{q}^{\boldsymbol{\varphi}}, \boldsymbol{g}^{\boldsymbol{\varphi}}) = \sum_{v \in \mathcal{V}} \max_{y \in \mathcal{Y}} q^{\boldsymbol{\varphi}}_v(x_v, y) + \sum_{\{v, v'\} \in \mathcal{E}} \max_{(y, y') \in \mathcal{Y}^2} g^{\boldsymbol{\varphi}}_{vv'}(y, y')
$$

7.C: Trivial max-sum problems

Let us define a set $\mathcal{C}_P \subseteq \mathcal{Y}^{\mathcal{V}}$ which contains labelings $\boldsymbol{y} \in \mathcal{C}_P$ such that

$$\begin{aligned} q_v(x_v, y_v) &\geq \max_{y \in \mathcal{Y}} q_v(x_v, y)\,, & v \in \mathcal{V} \\ g_{vv'}(y_v, y_{v'}) &\geq \max_{(y,y') \in \mathcal{Y}^2} g_{vv'}(y, y')\,, & \{v, v'\} \in \mathcal{E} \end{aligned} \tag{Triv}$$

Definition 2. *The max-sum problem $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ is called trivial if $\mathcal{C}_P \neq \emptyset$.*

Definition 3. *The max-sum problem $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ is called strictly trivial if it is trivial and all the inequalitites (Triv) are satisfied strictly.*

Proposition 2. *For any max-sum problem $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ the inequality*

$$U(\boldsymbol{x}, \boldsymbol{q}, \boldsymbol{g}) \geq \max_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} f(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{q}, \boldsymbol{g})$$

holds true. The bound is tight if and only if P is trivial.

Corrolary: It is clear that if $U(\boldsymbol{x}, \boldsymbol{q}^{\boldsymbol{\varphi}}, \boldsymbol{g}^{\boldsymbol{\varphi}}) > \min_{\boldsymbol{\varphi}'} U(\boldsymbol{x}, \boldsymbol{q}^{\boldsymbol{\varphi}'}, \boldsymbol{g}^{\boldsymbol{\varphi}'})$ then $P^{\boldsymbol{\varphi}}$ is not trivial.

Definition 4. *The max-sum problem $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ has a (strictly) trivial equivalent iff there exist $\boldsymbol{\varphi}$ such $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}^{\boldsymbol{\varphi}}, \boldsymbol{g}^{\boldsymbol{\varphi}}, \boldsymbol{x})$ is (strictly) trivial.*

7.C: Solving trivial max-sum problems by the LP relaxation

We can try to solve the max-sum problem P by checking whether it has a trivial equivalent as follows:

1. Solve the dual of LP relaxation

$$\boldsymbol{\varphi}^* = \operatorname*{argmin}_{\boldsymbol{\varphi}} U(\boldsymbol{x}, \boldsymbol{q}^{\boldsymbol{\varphi}}, \boldsymbol{g}^{\boldsymbol{\varphi}})$$

   It is a convex problem which can be translated to linear program. However, of-the-shelf solvers are not applicable for large problems.

2. Check the tightness of the LP relaxation by try to find $\boldsymbol{y} \in \mathcal{C}_P$:

   - Checking that $P^{\boldsymbol{\varphi}^*}$ is strictly trivial, i.e. $|\mathcal{C}_P| = 1$, requires $\mathcal{O}(|\mathcal{V}||\mathcal{Y}| + |\mathcal{E}||\mathcal{Y}|^2)$ operations.
   - Finding the consistent labeling $\boldsymbol{y} \in \mathcal{C}_P$ can be expresses as a constraint satisfaction problem (CSP) which is NP-complete in general.
     CSP can be seen as an instance of max-sum problem with quality functions $(\boldsymbol{q}, \boldsymbol{g})$ taking only values $\{-\infty, 0\}$.

7.C: Learning strictly trivial max-sum classifier

Task: For a given training set $\{(\boldsymbol{x}^1, \boldsymbol{y}^1), \ldots, (\boldsymbol{x}^m, \boldsymbol{y}^m)\} \in (\mathcal{X}^\mathcal{V} \times \mathcal{Y}^\mathcal{V})^m$ find the quality functions $(\boldsymbol{q}, \boldsymbol{g})$ such that $\boldsymbol{y}^i = h(\boldsymbol{x}^i; \boldsymbol{q}, \boldsymbol{g})$, $i \in \mathcal{I}$, and the max-sum problems $P^i = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x}^i)$, $i \in \mathcal{I}$, have a strictly trivial equivalent.

If $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ has a strictly trivial equivalent and optimal solution is $\boldsymbol{y}^*$ then there must exist $\boldsymbol{\varphi}$ such that the re-parametrized quality functions

$$
\begin{aligned}
q_v^{\boldsymbol{\varphi}}(y) &= q_v(y) - \sum_{v' \in \mathcal{N}(v)} \varphi_{vv'}(y), && v \in \mathcal{V}, y \in \mathcal{Y} \\
g_{vv'}^{\boldsymbol{\varphi}}(y, y') &= g_{vv'}(y, y') + \varphi_{vv'}(y) + \varphi_{v'v}(y'), && \{v, v'\} \in \mathcal{E}, y, y' \in \mathcal{Y}
\end{aligned}
$$

satisfies

$$
\begin{aligned}
q_v^{\boldsymbol{\varphi}}(x_v, y_v^*) &> \max_{y \in \mathcal{Y} \setminus \{y_v^*\}} q_v^{\boldsymbol{\varphi}}(x_v, y), && v \in \mathcal{V} \\
g_{vv'}^{\boldsymbol{\varphi}}(y_v^*, y_{v'}^*) &> \max_{(y,y') \in \mathcal{Y}^2 \setminus \{(y_v^*, y_{v'}^*)\}} g_{vv'}^{\boldsymbol{\varphi}}(y, y'), && \{v, v'\} \in \mathcal{E}
\end{aligned}
$$

Hence, learning the max-sum problem with STE is equivalent to solving a set of $m(|\mathcal{V}|(|\mathcal{Y}| - 1) + |\mathcal{E}|(|\mathcal{Y}|^2 - 1))$ strict linear inequalities

$$
\begin{aligned}
q_v^{\boldsymbol{\varphi}^i}(x_v, y_v^i) &> q_v^{\boldsymbol{\varphi}^i}(x_v, y)\,, && i \in \mathcal{I}, v \in \mathcal{V}\,, y \in \mathcal{Y} \setminus \{y_v^i\} \\
g_{vv'}^{\boldsymbol{\varphi}}(y_v^i, y_{v'}^i) &> g_{vv'}^{\boldsymbol{\varphi}^i}(y, y')\,, && i \in \mathcal{I}, \{v, v'\} \in \mathcal{E}\,, (y, y') \in \mathcal{Y}^2 \setminus \{(y_v^i, y_{v'}^i)\}
\end{aligned}
$$

7.C: Perceptron learning strictly trivial max-sum classifier

1. Set $\boldsymbol{q} \leftarrow \mathbf{0}$, $\boldsymbol{g} \leftarrow \mathbf{0}$, $\boldsymbol{\varphi}^i \leftarrow \mathbf{0}$, $i \in \mathcal{I}$.
2. Find a triplet $i \in \mathcal{I}$, $v \in \mathcal{V}$, $y \in \mathcal{Y} \setminus \{y^i_v\}$ such that

$$q_v(x^i_v, y^i_v) - \sum_{v' \in \mathcal{N}(v)} \varphi^i_{vv'}(y^i_v) \leq q_v(x^i_v, y) - \sum_{v' \in \mathcal{N}(v)} \varphi^i_{vv'}(y)$$

3. If no such triplet (i, v, y) exists then go to Step 4. Otherwise update $\boldsymbol{q}$ and $\boldsymbol{\varphi}^i$ by

$$\begin{array}{rclrcl} \varphi^i_{vv'}(y^i_v) & \leftarrow & \varphi^i_{vv'}(y^i_v) - 1, & \varphi^i_{vv'}(y) & \leftarrow & \varphi^i_{vv'}(y) + 1, \quad v' \in \mathcal{N}(v) \\ q_v(x^i_v, y^i_v) & \leftarrow & q_v(x^i_v, y^i_v) + 1, & q_v(x^i_v, y) & \leftarrow & q_v(x^i_v, y) - 1 \end{array}$$

4. Find a five-tuple $i \in \mathcal{I}$, $\{v, v'\} \in \mathcal{E}$, $(y, y') \in \mathcal{Y}^2 \setminus \{(y^i_v, y^i_{v'})\}$ such that

$$\boldsymbol{g}_{vv'}(y^i_v, y^i_{v'}) + \varphi^i_{vv'}(y^i_v) + \varphi^i_{v'v}(y^i_{v'}) \leq g_{vv'}(y, y') + \varphi^i_{vv'}(y) + \varphi^i_{v'v}(y')$$

5. If no such five-tuple (i, v, v', y, y') exists and no update was made in Step 3 then $(\boldsymbol{q}, \boldsymbol{g}, \boldsymbol{\varphi}^i, i \in \mathcal{I})$ solves the tasks. Otherwise update $\boldsymbol{g}$ and $\boldsymbol{\varphi}^i$ by

$$\begin{array}{rclrcl} \varphi^i_{vv'}(y^i_v) & \leftarrow & \varphi^i_{vv'}(y^i_v) + 1, & \varphi^i_{v'v}(y^i_{v'}) & \leftarrow & \varphi^i_{v'v}(y^i_{v'}) + 1, \\ \varphi^i_{vv'}(y) & \leftarrow & \varphi^i_{vv'}(y) - 1, & \varphi^i_{v'v}(y') & \leftarrow & \varphi^i_{v'v}(y') - 1, \\ g_{vv'}(y^i_v, y^i_{v'}) & \leftarrow & g_{vv'}(y^i_v, y^i_{v'}) + 1, & g_{vv'}(y, y') & \leftarrow & g_{vv'}(y, y') - 1 \end{array}$$

   and go to step 2.

7.C: Generalization

Theorem 1. *Let $P = (\mathcal{V}, \mathcal{E}, \boldsymbol{q}, \boldsymbol{g}, \boldsymbol{x})$ be a max-sum problem and let P have a unique solution. If $(\mathcal{V}, \mathcal{E})$ is acyclic or quality functions $-\boldsymbol{g}$ are sub-modular then P is equivalent to some strictly trivial problem.*

General form of quality functions: It is straightforward to extend the algorithm so that it learns a max-sum classifier $h(\boldsymbol{x}; \boldsymbol{w}) = \operatorname{argmax}_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} f(\boldsymbol{x}, \boldsymbol{w})$ with score

$$f(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}) = \langle \boldsymbol{w}, \boldsymbol{\Psi}(\boldsymbol{x}, \boldsymbol{y}) \rangle = \left\langle \boldsymbol{w}, \sum_{v \in \mathcal{V}} \boldsymbol{\Psi}_v(\boldsymbol{x}, y_v) + \sum_{\{v,v'\} \in \mathcal{E}} \boldsymbol{\Psi}_{v,v'}(\boldsymbol{x}, y_v, y_{v'}) \right\rangle$$

where $\boldsymbol{w} \in \mathbb{R}^n$ are parameters to be learned while $\boldsymbol{\Psi}_v \colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$, $v \in \mathcal{V}$ and $\boldsymbol{\Psi}_{vv'} \colon \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^n$, $\{v, v'\} \in \mathcal{E}$ are fixed.

7.C: Example: Sudoku solver

puzzle assignment

					8			
	1	9	5	6		2		
2	5			1		3	6	
9					2		8	1
	8	2	6		9			
5	7		1					2
	2	1		9			4	3
		5		7	6	8		
8	9		3					

solution

7	6	3	4	2	8	1	9	5
4	1	9	5	6	3	2	7	8
2	5	8	9	1	7	3	6	4
9	3	4	7	5	2	6	8	1
1	8	2	6	3	9	4	5	7
5	7	6	1	8	4	9	3	2
6	2	1	8	9	5	7	4	3
3	4	5	2	7	6	8	1	9
8	9	7	3	4	1	5	2	6

The task of Sudoku game is to fill empty fields such that each row, each column and each 3×3 field contains numbers $\{1, 2, \ldots, 9\}$.

7.C: Example: Sudoku solver

- We can solve Sudoku by an instance of max-sum classifier

$$\boldsymbol{y}^* = \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \Bigg(\underbrace{\sum_{v\in\mathcal{V}} q(x_v, y_v)}_{\text{copy given fields}} + \underbrace{\sum_{\{v,v'\}\in\mathcal{E}} g(y_v, y_{v'})}_{\text{neighbors must be different}} \Bigg)$$

- Play field $\mathcal{V} = \{(i,j) \in \mathbb{N}^2 \mid 1 \leq i \leq 9, 1 \leq j \leq 9\}$
- Assignment $\boldsymbol{x} = (x_v \in \{\Box, 1, \ldots, 9\} \mid v \in \mathcal{V}) \in \mathcal{X}^{\mathcal{V}}$; solution $\boldsymbol{y} = (y_v \in \{1, \ldots, 9\} \mid v \in \mathcal{V}) \in \mathcal{Y}^{\mathcal{V}}$
- Related fields $\mathcal{E} = \{\{(i,j),(i',j')\} \mid i = i' \vee j = j' \vee (\lceil i/3 \rceil = \lceil i'/3 \rceil \wedge \lceil j/3 \rceil = \lceil j'/3 \rceil)\}$
- $q\colon \{\Box, 1, \ldots, 9\} \times \{1, \ldots, 9\} \to \{0, -\infty\}$ such that
$$q(x,y) = \begin{cases} -\infty & \text{if} \quad x \neq \Box \wedge y \neq x \\ 0 & \text{otherwise} \end{cases}$$
- $g\colon \{1, \ldots, 9\}^2 \to \{0, -\infty\}$ such that $g(y, y') = \begin{cases} 0 & \text{if} \quad y \neq y' \\ -\infty & \text{if} \quad y = y' \end{cases}$

Assignment for seminar: learn the quality fanctions $(\boldsymbol{q}, \boldsymbol{g})$ from an example of Sudoku assignment and its correct solution.

7.C: Recap

So far we have been talking about:

7.A: Definition of structured classification task and its solution via generative and discriminative learning

7.B: Implementation of ERM learning using Perceptron algorithm

7.C: Learning of max-sum classifier

Next we show how to implement the ERM for non-separable examples and general linear classifier:

8.A: Learning two-class linear classifier from non-separable examples by SVM.

8.B: Structured output SVM.

8.C: Structured output SVM for learning max-sum classifiers.

8.A: Two-class linear classifier

- Observation is n-dimenzionální vektor $\boldsymbol{x} \in \mathcal{X} = \mathbb{R}^n$.
- Hidden state (label) attains only two values $y \in \mathcal{Y} = \{+1, -1\}$
- Linear classifier

$$h(\boldsymbol{x}; \boldsymbol{w}) = \operatorname*{argmax}_{y \in \mathcal{Y}} y \langle \boldsymbol{w}, \boldsymbol{x} \rangle = \begin{cases} +1 & \text{if} \quad \langle \boldsymbol{w}, \boldsymbol{x} \rangle \geq 0 \\ -1 & \text{if} \quad \langle \boldsymbol{w}, \boldsymbol{x} \rangle < 0 \end{cases}$$

  A biased decision function can be obtained via transformation $\boldsymbol{w}' = (\boldsymbol{w}; b)$ and $\boldsymbol{x}' = (x; 1)$.
- Let us assume $0/1$-loss function $\Delta(y, y') = [\![y \neq y']\!]$.
- We are going to discuss how to learn $\boldsymbol{w}$ from examples $\mathcal{T} = \{(\boldsymbol{x}^i, \boldsymbol{y}^i) \in \mathcal{X} \times \mathcal{Y} \mid i \in \mathcal{I}\}$.

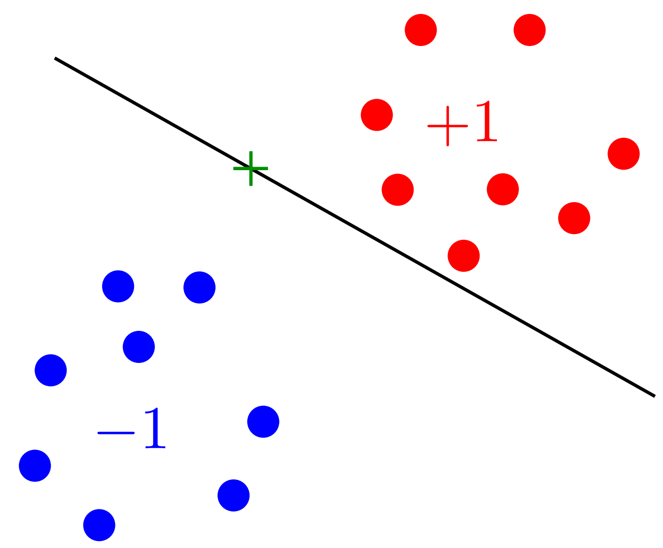
8.A: Two-class SVM: separable examples

- Linearly separable training examples $\mathcal{T} = \{(\boldsymbol{x}^1, y^1), \ldots, (\boldsymbol{x}^m, y^m)\} \in (\mathbb{R}^n \times \{+1, -1\})^m$ imply existence of $\boldsymbol{w} \in \mathbb{R}^n$ such that

$$R_{\mathcal{T}}(h(\cdot;\boldsymbol{w})) = \frac{1}{m}\sum_{i=1}^{m} [\![y^i \neq h(\boldsymbol{x}^i;\boldsymbol{w})]\!]$$

- Searching for $\boldsymbol{w}$ such that $R_{\mathcal{T}}(h(\cdot;\boldsymbol{w})) = 0$ lead to solving a set of linear inequalities:

$$y^i \langle \boldsymbol{w}, \boldsymbol{x}^i \rangle > 0\,, \quad i = 1, \ldots, m$$



8.A: Two-class classifier: optimal separating hyperplane

- Optimal separating hyperplane $\mathcal{H}^* = \{\boldsymbol{x} \in \mathbb{R}^n \mid \langle \boldsymbol{w}^*, \boldsymbol{x} \rangle = 0\}$ maximizes the geometrical margin to the training points:

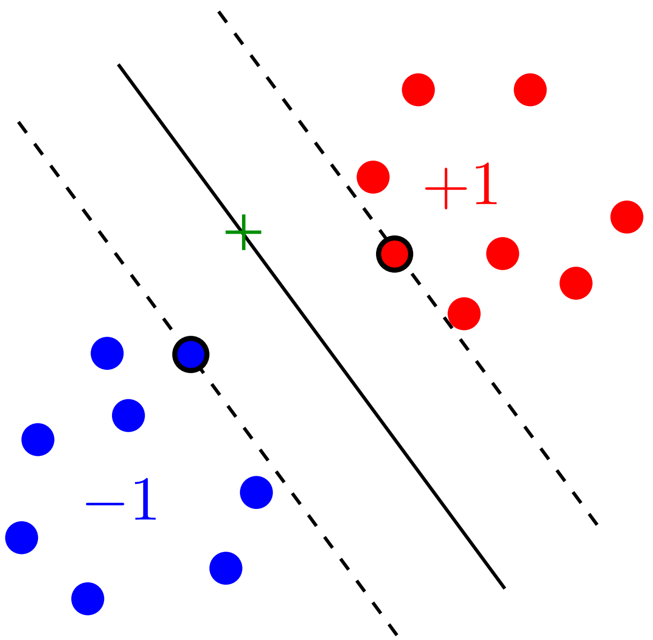
$$\boldsymbol{w}^* \in \operatorname*{argmax}_{\boldsymbol{w} \in \mathbb{R}^n} \min_{i=1,\ldots,m} \frac{y^i \langle \boldsymbol{w}, \boldsymbol{x}^i \rangle}{\|\boldsymbol{w}\|}$$

- Searching for the optimal hyperplane leads to quadratic programming

$$\boldsymbol{w}^* = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^n} \frac{1}{2} \|\boldsymbol{w}\|^2$$

subject to

$$y^i \langle \boldsymbol{w}, \boldsymbol{x}^i \rangle \geq 1\,, \quad i = 1, \ldots, m$$



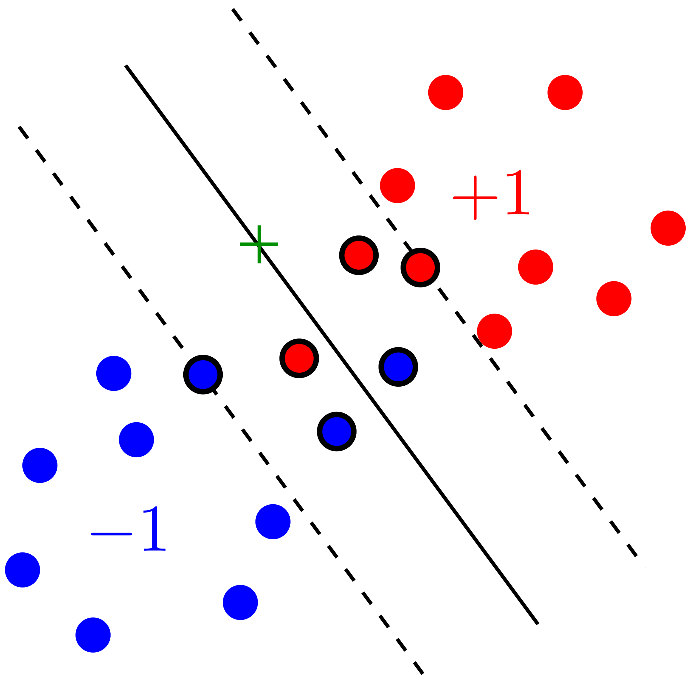
8.A: Two-class classifier: non-separable examples

$$g(\boldsymbol{w}^*, \boldsymbol{\xi}^*) = \operatorname*{argmin}_{\boldsymbol{w},\boldsymbol{\xi}} \left(\frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{m}\sum_{i=1}^{m} \xi_i \right)$$

subject to

$$\begin{array}{rcll} y^i\langle \boldsymbol{w}, \boldsymbol{x}^i \rangle & \geq & 1 - \xi_i \,, & i = 1, \ldots, m \\ \xi_i & \geq & 0 \,, & i = 1, \ldots, m \end{array}$$

where $\lambda > 0$ is a fixed regularization constant.



- Learning leads to a convex quadratic programming.
- Two-class linear Support Vector Machines (SVM) algorithm.

8.A: Minimization of the regularized empirical risk: two-class classifier

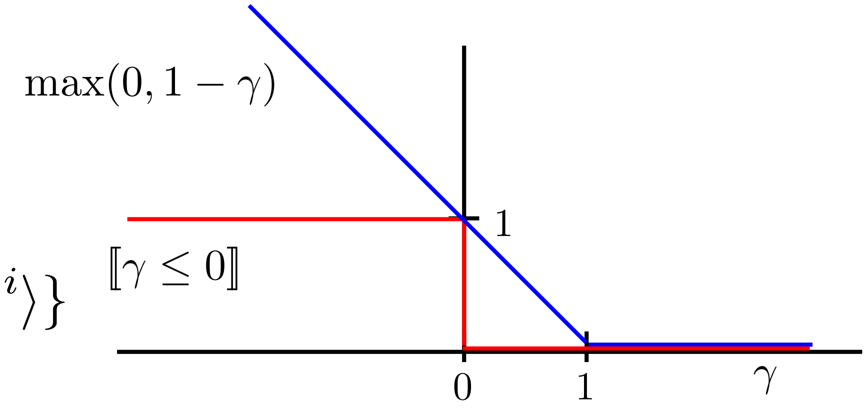
- Learning of the SVM classifier can be seen as an unconstrained problem

$$\boldsymbol{w}^* = \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\lambda \underbrace{\Omega(\boldsymbol{w})}_{\text{regularizer}} + \underbrace{\hat{R}_{\mathcal{T}}(\boldsymbol{w})}_{\text{surrogate of empirical risk}} \right)$$

- The regularizer $\Omega\colon \mathbb{R}^n \to \mathbb{R}$ is a convex function: $\Omega(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2$ or $\Omega(\boldsymbol{w}) = \|\boldsymbol{w}\|_1$.
- The surrogate risk is a convex upper bound of the empirical risk

$$\hat{R}_{\mathcal{T}}(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m} \max\left\{0, 1 - y^i\langle \boldsymbol{w}, \boldsymbol{x}^i\rangle\right\}$$

$$\begin{aligned} [\![y^i \neq h(\boldsymbol{x}^i;\boldsymbol{w})]\!] &= [\![y^i\langle \boldsymbol{w}, \boldsymbol{x}^i\rangle \leq 0]\!] \\ &\leq \max\{0, 1 - y^i\langle \boldsymbol{w}, \boldsymbol{x}^i\rangle\} \end{aligned}$$



8.A: Minimization of the regularized empirical risk: structured classifier

- Given training examples $\mathcal{T} = \{(\boldsymbol{x}^i, \boldsymbol{y}^i) \in \mathcal{X} \times \mathcal{Y} \mid i \in \mathcal{I}\}$, the goal is to learn parameters $\boldsymbol{w} \in \mathbb{R}^n$ of a general linear classifier

$$h(\boldsymbol{x}; \boldsymbol{w}) = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}} \langle \boldsymbol{w}, \boldsymbol{\Psi}(\boldsymbol{x}, \boldsymbol{y}) \rangle$$

where $\boldsymbol{\Psi} \colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$ is fixed feature map.

- Regularized empirical risk minimization based learning leads to solving

$$\boldsymbol{w}^* = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^n} \Big(\lambda \underbrace{\Omega(\boldsymbol{w})}_{\text{regularizer}} + \underbrace{\hat{R}_{\mathcal{T}}(\boldsymbol{w})}_{\substack{\text{surrogate of} \\ \text{empirical risk}}} \Big)$$

where $\Omega \colon \mathbb{R}^n \to \mathbb{R}$ is a (convex) regularizer and $\hat{R}_{\mathcal{T}} \colon \mathbb{R}^n \to \mathbb{R}$ is a surrogate of the empirical risk

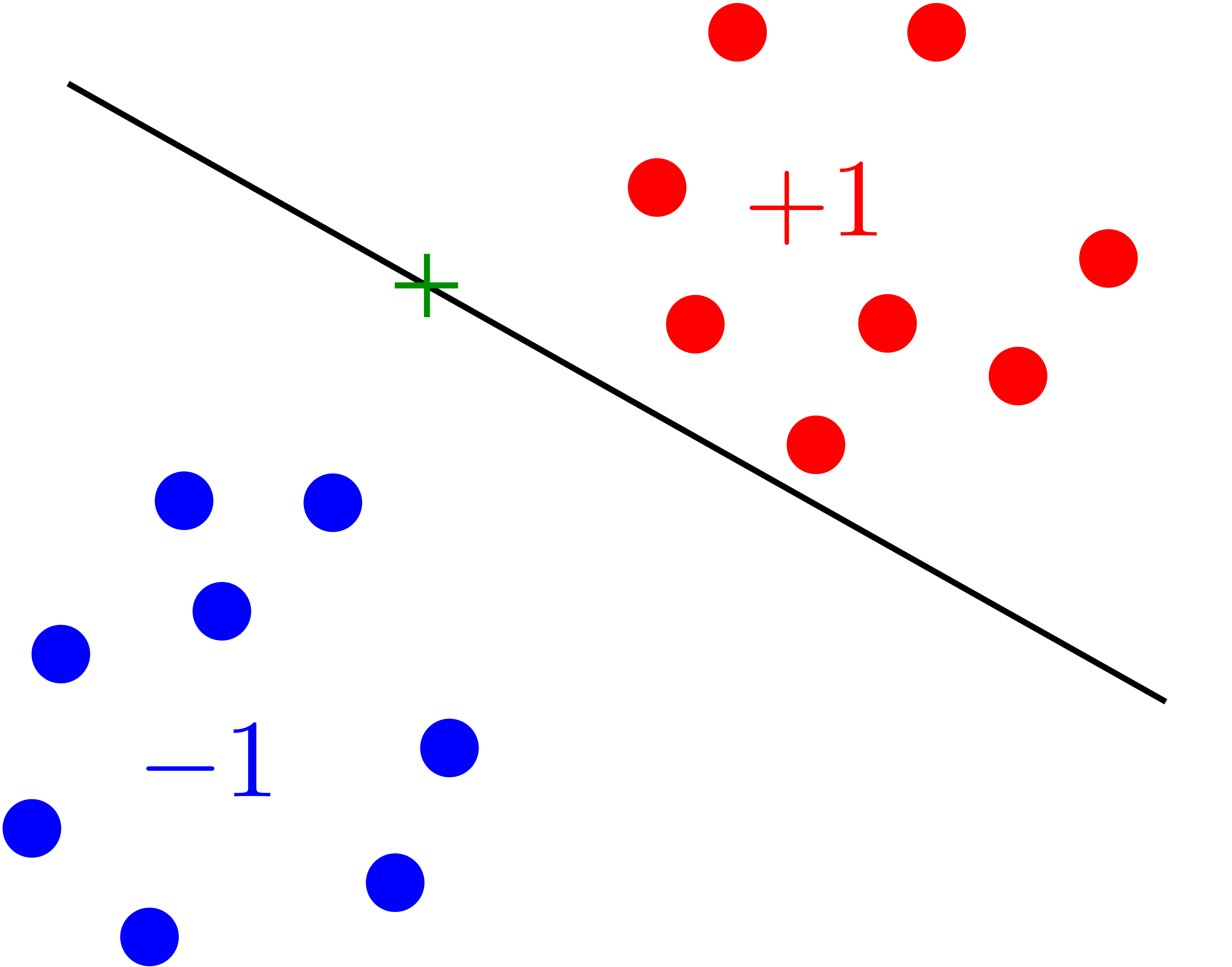
$$R_{\mathcal{T}}(h(\cdot; \boldsymbol{w}) = \frac{1}{m} \sum_{i=1}^{m} \ell(\boldsymbol{y}^i, h(\boldsymbol{x}^i; \boldsymbol{w}))$$

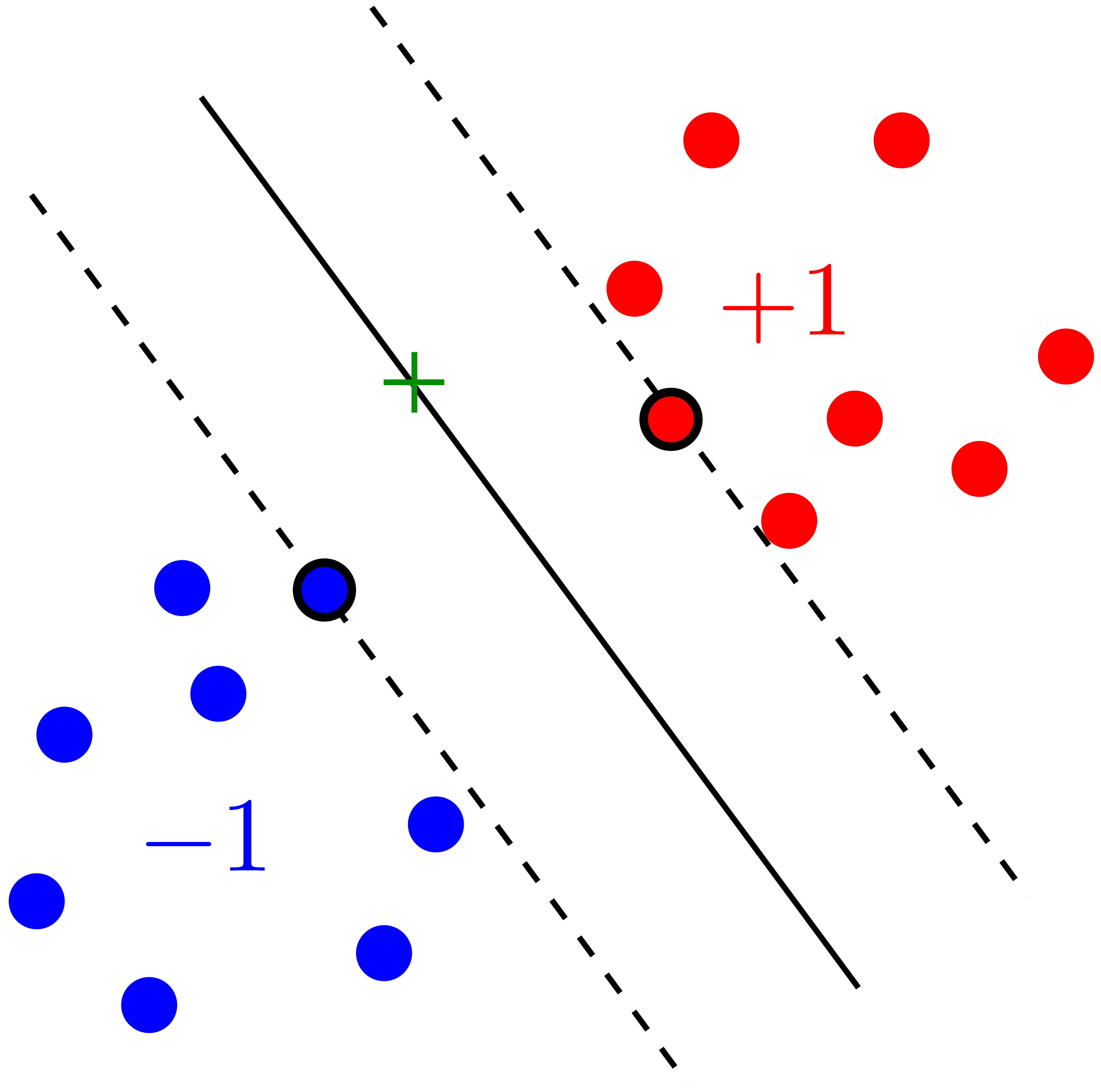
and $\ell \colon \mathcal{Y} \times \mathcal{Y} \to [0, \infty)$ is an application specific loss.

Question: How to construct the surrogate $\hat{R}_{\mathcal{T}}$ for a generic linear classifier and loss ?

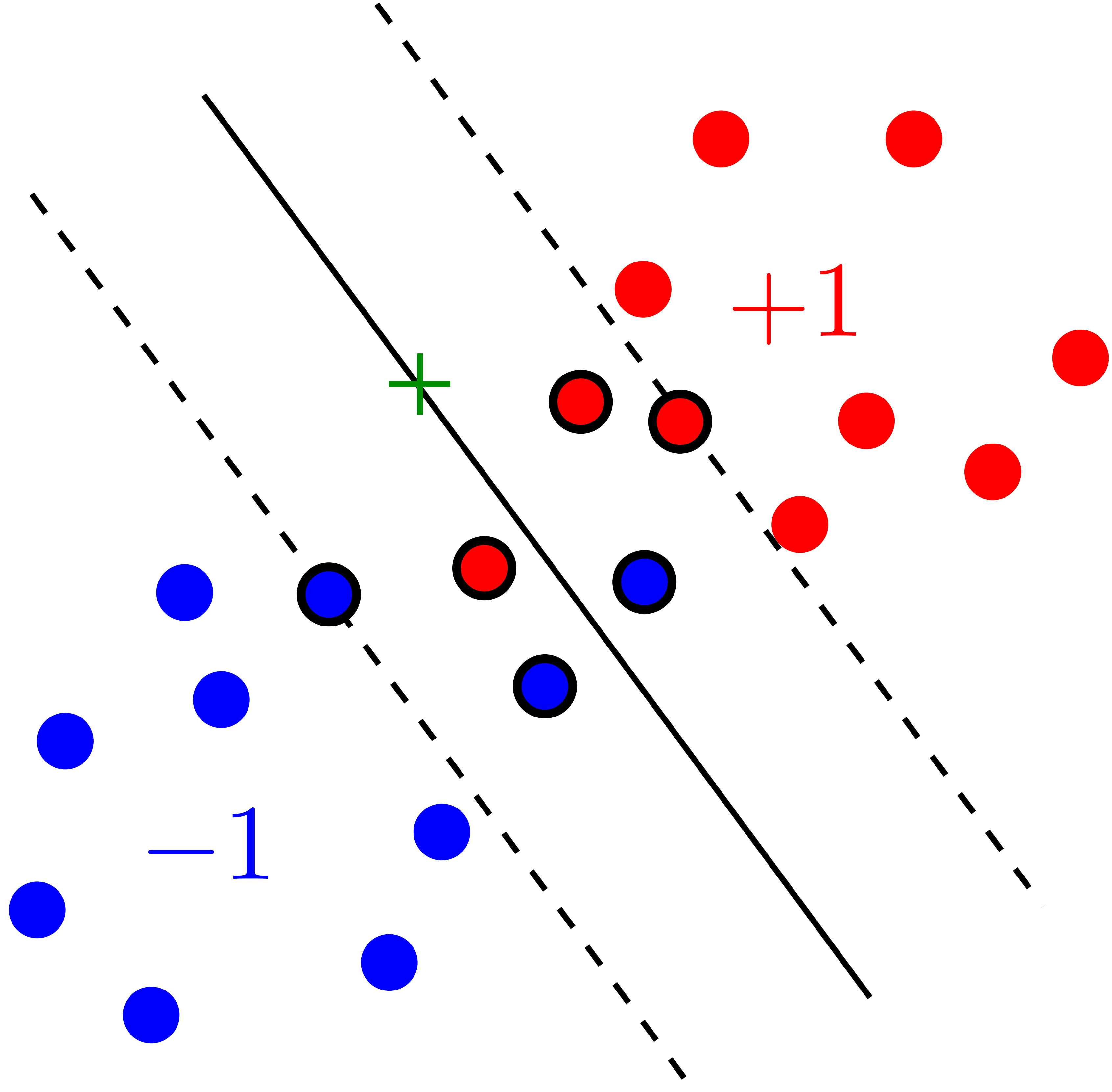
					8			
	1	9	5	6		2		
2	5			1		3	6	
9					2		8	1
	8	2	6		9			
5	7		1					2
	2	1		9			4	3
		5		7	6	8		
8	9		3					

7	6	3	4	2	8	1	9	5
4	1	9	5	6	3	2	7	8
2	5	8	9	1	7	3	6	4
9	3	4	7	5	2	6	8	1
1	8	2	6	3	9	4	5	7
5	7	6	1	8	4	9	3	2
6	2	1	8	9	5	7	4	3
3	4	5	2	7	6	8	1	9
8	9	7	3	4	1	5	2	6

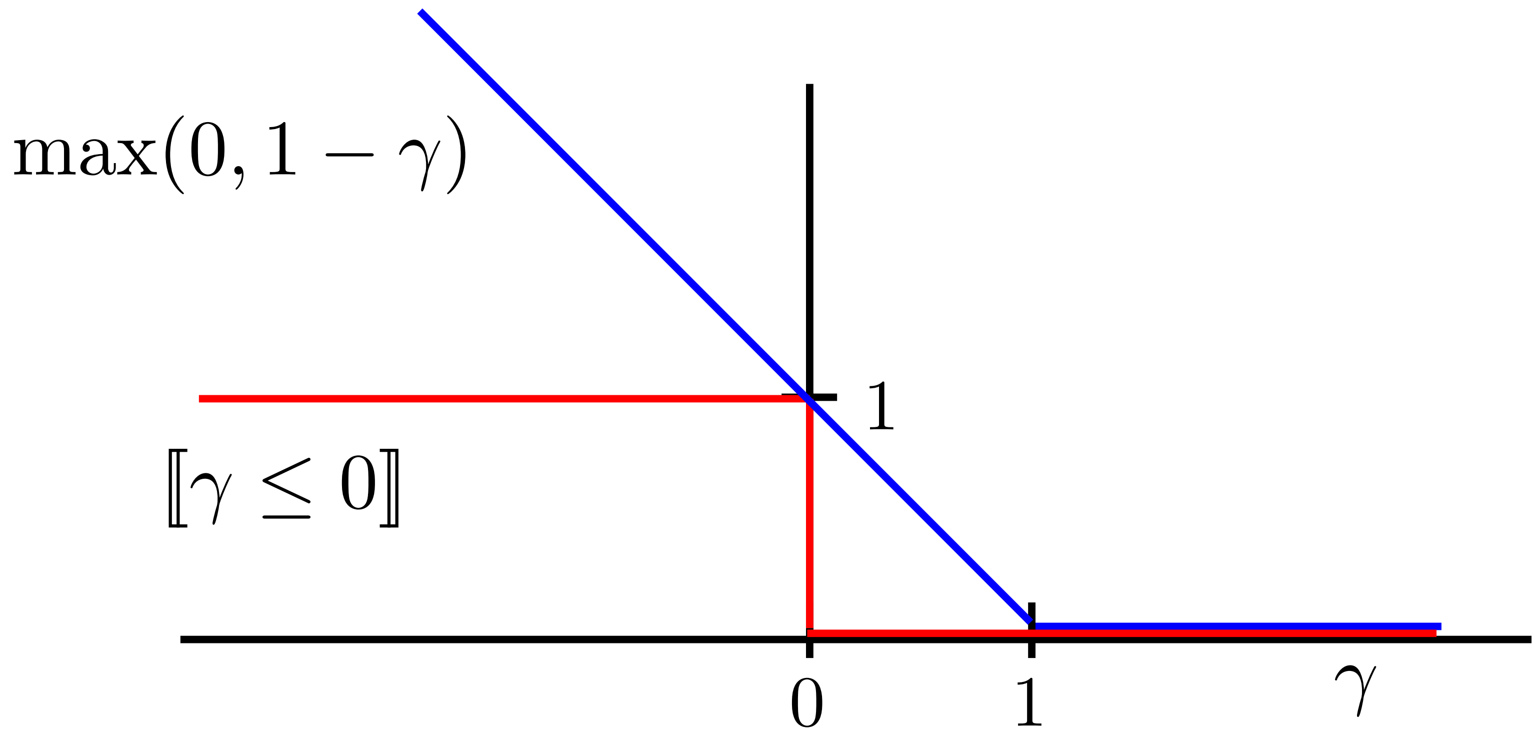

$+1$
-1



+1
−1



+1
−1



$\max(0, 1-\gamma)$
$[\![\gamma \leq 0]\!]$
1
0
1
γ