

# Lecture 9: Learning max-sum classifier

Vojtěch Franc

April 27, 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
- ◆  $q_v: \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$  quality of label  $y_v$  given  $x_v$ ;  $\mathbf{q} = (q_v(x, y) \in \mathbb{R} \mid x \in \mathcal{X}, y \in \mathcal{Y}, v \in \mathcal{V})$
- ◆  $g_{vv'}: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$  quality a label pair  $(y_v, y_{v'})$ ;  
 $\mathbf{g} = (g_{vv'}(y, y') \in \mathbb{R} \mid (y, y') \in \mathcal{Y}^2, \{v, v'\} \in \mathcal{E})$

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

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

where

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

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

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

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

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

is the max-sum classifier

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

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



4/21

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

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

The max-sum problem  $\mathcal{P} = (\mathcal{E}, \mathcal{V}, \mathbf{q}, \mathbf{g}, \mathbf{x})$  associated with the classification  $h(\mathbf{x}; \mathbf{q}, \mathbf{g})$  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.  $\mathcal{P} = (\mathcal{V}, \mathcal{E}, \mathbf{q}, \mathbf{g}, \mathbf{x})$  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

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

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

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

subject to

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

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

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

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

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

### Questions:

1. Is the LP relaxation tight, i.e., does it hold that  $U(\mathbf{x}, \mathbf{q}^{\varphi^*}, \mathbf{g}^{\varphi^*}) = f(\mathbf{x}, \mathbf{y}^*, \mathbf{q}, \mathbf{g})$  ?
2. If yes how to get the labels  $\mathbf{y}^*$  ?

## 7.C: Interpretation of the dual of LP relaxation

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

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

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

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

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

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

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

## 7.C: Trivial max-sum problems

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

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

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

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

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

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

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

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

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

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

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  $\mathbf{y} \in \mathcal{C}_P$ :

- ◆ Checking that  $P^{\varphi^*}$  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  $\mathbf{y} \in \mathcal{C}_P$  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  $(\mathbf{q}, \mathbf{g})$  taking only values  $\{-\infty, 0\}$ .

## 7.C: Learning strictly trivial max-sum classifier

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

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

$$q_v^\varphi(y) = q_v(y) - \sum_{v' \in \mathcal{N}(v)} \varphi_{vv'}(y), \quad v \in \mathcal{V}, y \in \mathcal{Y}$$

$$g_{vv'}^\varphi(y, y') = g_{vv'}(y, y') + \varphi_{vv'}(y) + \varphi_{v'v}(y'), \quad \{v, v'\} \in \mathcal{E}, y, y' \in \mathcal{Y}$$

satisfies

$$q_v^\varphi(x_v, y_v^*) > \max_{y \in \mathcal{Y} \setminus \{y_v^*\}} q_v^\varphi(x_v, y), \quad v \in \mathcal{V}$$

$$g_{vv'}^\varphi(y_v^*, y_{v'}^*) > \max_{(y, y') \in \mathcal{Y}^2 \setminus \{(y_v^*, y_{v'}^*)\}} g_{vv'}^\varphi(y, y'), \quad \{v, v'\} \in \mathcal{E}$$

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

$$q_v^{\varphi^i}(x_v, y_v^i) > q_v^{\varphi^i}(x_v, y), \quad i \in \mathcal{I}, v \in \mathcal{V}, y \in \mathcal{Y} \setminus \{y_v^i\}$$

$$g_{vv'}^{\varphi^i}(y_v^i, y_{v'}^i) > g_{vv'}^{\varphi^i}(y, y'), \quad i \in \mathcal{I}, \{v, v'\} \in \mathcal{E}, (y, y') \in \mathcal{Y}^2 \setminus \{(y_v^i, y_{v'}^i)\}$$

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

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

$$q_v(x_v^i, y_v^i) - \sum_{v' \in \mathcal{N}(v)} \varphi_{vv'}^i(y_v^i) \leq q_v(x_v^i, y) - \sum_{v' \in \mathcal{N}(v)} \varphi_{vv'}^i(y)$$

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

$$\begin{aligned} \varphi_{vv'}^i(y_v^i) &\leftarrow \varphi_{vv'}^i(y_v^i) - 1, & \varphi_{vv'}^i(y) &\leftarrow \varphi_{vv'}^i(y) + 1, & v' \in \mathcal{N}(v) \\ q_v(x_v^i, y_v^i) &\leftarrow q_v(x_v^i, y_v^i) + 1, & q_v(x_v^i, y) &\leftarrow q_v(x_v^i, y) - 1 \end{aligned}$$

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

$$\mathbf{g}_{vv'}(y_v^i, y_{v'}^i) + \varphi_{vv'}^i(y_v^i) + \varphi_{v'v}^i(y_{v'}^i) \leq \mathbf{g}_{vv'}(y, y') + \varphi_{vv'}^i(y) + \varphi_{v'v}^i(y')$$

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

$$\begin{aligned} \varphi_{vv'}^i(y_v^i) &\leftarrow \varphi_{vv'}^i(y_v^i) + 1, & \varphi_{v'v}^i(y_{v'}^i) &\leftarrow \varphi_{v'v}^i(y_{v'}^i) + 1, \\ \varphi_{vv'}^i(y) &\leftarrow \varphi_{vv'}^i(y) - 1, & \varphi_{v'v}^i(y') &\leftarrow \varphi_{v'v}^i(y') - 1, \\ \mathbf{g}_{vv'}(y_v^i, y_{v'}^i) &\leftarrow \mathbf{g}_{vv'}(y_v^i, y_{v'}^i) + 1, & \mathbf{g}_{vv'}(y, y') &\leftarrow \mathbf{g}_{vv'}(y, y') - 1 \end{aligned}$$

and go to step 2.

## 7.C: Generalization

**Theorem 1.** *Let  $P = (\mathcal{V}, \mathcal{E}, \mathbf{q}, \mathbf{g}, \mathbf{x})$  be a max-sum problem and let  $P$  have a **unique** solution. If  $(\mathcal{V}, \mathcal{E})$  is acyclic or quality functions  $-g$  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  $h(\mathbf{x}; \mathbf{w}) = \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}^{\mathcal{V}}} f(\mathbf{x}, \mathbf{w})$  with score

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

where  $\mathbf{w} \in \mathbb{R}^n$  are parameters to be learned while  $\Psi_v: \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^n$ ,  $v \in \mathcal{V}$  and  $\Psi_{vv'}: \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}^n$ ,  $\{v, v'\} \in \mathcal{E}$  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  $3 \times 3$  field contains numbers  $\{1, 2, \dots, 9\}$ .

## 7.C: Example: Sudoku solver

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

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

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

**Assignment for seminar:** learn the quality functions  $(q, g)$  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  $n$ -dimensionální vektor  $\mathbf{x} \in \mathcal{X} = \mathbb{R}^n$ .
- ◆ Hidden state (label) attains only two values  $y \in \mathcal{Y} = \{+1, -1\}$
- ◆ Linear classifier

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

A biased decision function can be obtained via transformation  $\mathbf{w}' = (\mathbf{w}; b)$  and  $\mathbf{x}' = (x; 1)$ .

- ◆ Let us assume 0/1-loss function  $\Delta(y, y') = [y \neq y']$ .
- ◆ We are going to discuss how to learn  $\mathbf{w}$  from examples  $\mathcal{T} = \{(\mathbf{x}^i, \mathbf{y}^i) \in \mathcal{X} \times \mathcal{Y} \mid i \in \mathcal{I}\}$ .



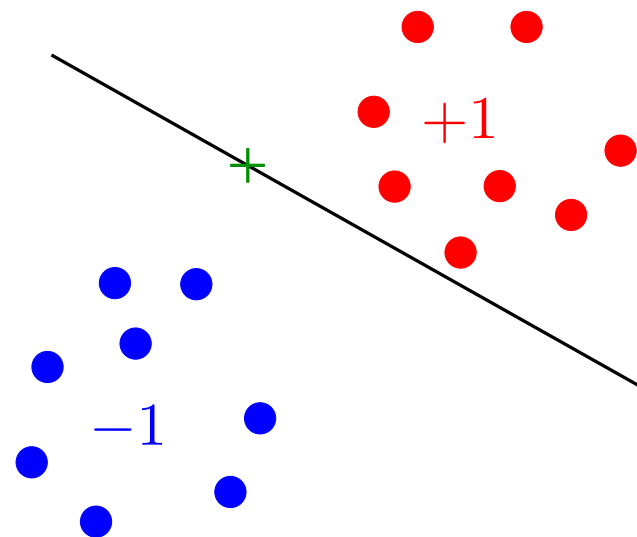
## 8.A: Two-class SVM: separable examples

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

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

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

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



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

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

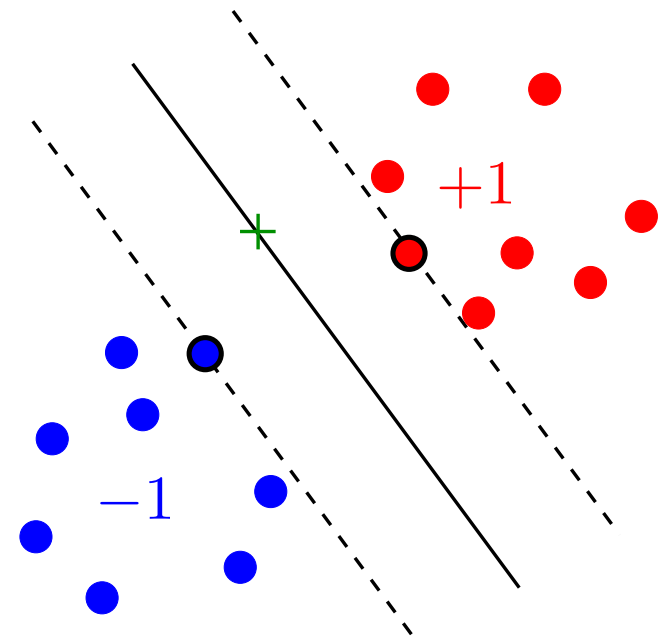
$$w^* \in \operatorname{argmax}_{w \in \mathbb{R}^n} \min_{i=1, \dots, m} \frac{y^i \langle w, x^i \rangle}{\|w\|}$$

- Searching for the optimal hyperplane leads to quadratic programming

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

subject to

$$y^i \langle w, x^i \rangle \geq 1, \quad i = 1, \dots, m$$



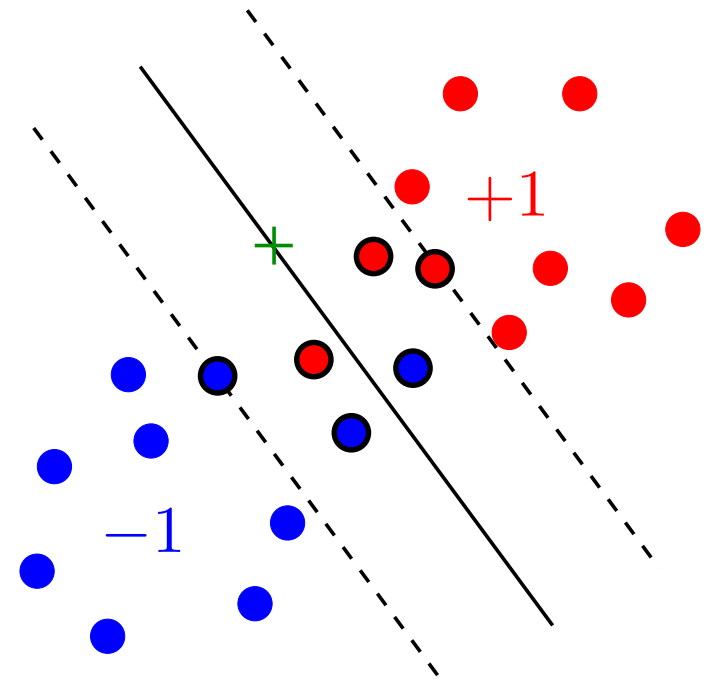
## 8.A: Two-class classifier: non-separable examples

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

subject to

$$\begin{aligned} y^i \langle \mathbf{w}, \mathbf{x}^i \rangle &\geq 1 - \xi_i, & i = 1, \dots, m \\ \xi_i &\geq 0, & i = 1, \dots, m \end{aligned}$$

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

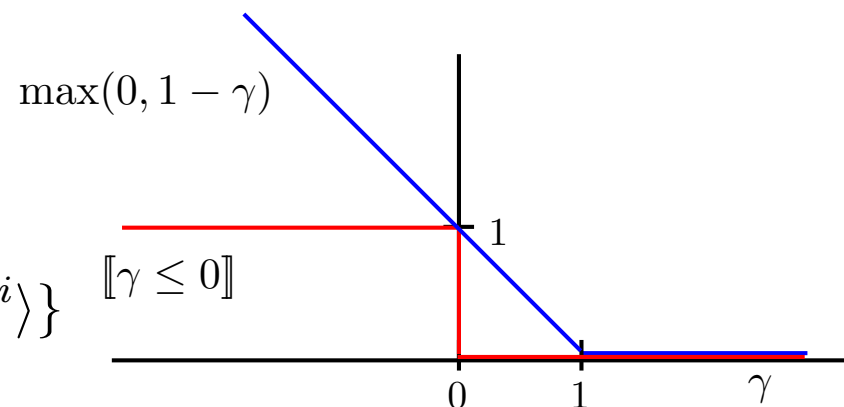
- ◆ Learning of the SVM classifier can be seen as an unconstrained problem

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

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

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

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



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

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

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

where  $\Psi: \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^n$  is fixed feature map.

- Regularized empirical risk minimization based learning leads to solving

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

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

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

and  $\ell: \mathcal{Y} \times \mathcal{Y} \rightarrow [0, \infty)$  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

