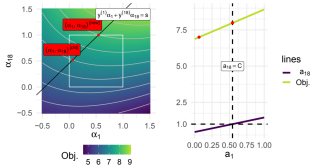


# Introduction to Machine Learning

## Support Vector Machine Training

### Learning goals

- Know that the SVM problem is not differentiable
- Know how to optimize the SVM problem in the primal via subgradient descent
- Know how to optimize SVM in the dual formulation via pairwise coordinate ascent



# SUPPORT VECTOR MACHINE TRAINING

- Until now, we have ignored the issue of solving the various convex optimization problems.
- The first question is whether we should solve the **primal** or the **dual problem**.
- In the literature SVMs are usually trained in the dual.
- However, SVMs can be trained both in the primal and the dual – each approach has its advantages and disadvantages.
- It is not easy to create an efficient SVM solver, and often specialized approaches have been developed, we only cover basic ideas here.

# TRAINING SVM IN THE PRIMAL

Unconstrained formulation of soft-margin SVM:

$$\min_{\boldsymbol{\theta}, \theta_0} \quad \frac{\lambda}{2} \|\boldsymbol{\theta}\|^2 + \sum_{i=1}^n L\left(y^{(i)}, f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)$$

where  $L(y, f) = \max(0, 1 - yf)$  and  $f(\mathbf{x} \mid \boldsymbol{\theta}) = \boldsymbol{\theta}^T \mathbf{x} + \theta_0$ .

(We inconsequentially changed the regularization constant.)

We cannot directly use GD, as the above is not differentiable.

## Solutions:

- 1 Use smoothed loss (squared hinge, huber), then do GD.  
NB: Will not create a sparse SVM if we do not add extra tricks.
- 2 Use **subgradient** methods.
- 3 Do stochastic subgradient descent.  
Pegasos: Primal Estimated sub-GrAdient SOLver for SVM.

# PEGASOS: SSGD IN THE PRIMAL

Approximate the risk by a stochastic 1-sample version:

$$\frac{\lambda}{2} \|\boldsymbol{\theta}\|^2 + L\left(y^{(i)}, f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)$$

With:  $f(\mathbf{x} \mid \boldsymbol{\theta}) = \boldsymbol{\theta}^T \mathbf{x} + \theta_0$  and  $L(y, f) = \max(0, 1 - yf)$

The subgradient for  $\boldsymbol{\theta}$  is  $\lambda \boldsymbol{\theta} - y^{(i)} \mathbf{x}^{(i)} \mathbb{I}_{yf < 1}$

---

Stochastic subgradient descent (without intercept  $\theta_0$ )

---

- 1: **for**  $t = 1, 2, \dots$  **do**
  - 2:     Pick step size  $\alpha$
  - 3:     Randomly pick an index  $i$
  - 4:     If  $y^{(i)} f(\mathbf{x}^{(i)}) < 1$  set  $\boldsymbol{\theta}^{[t+1]} = (1 - \lambda\alpha)\boldsymbol{\theta}^{[t]} + \alpha y^{(i)} \mathbf{x}^{(i)}$
  - 5:     If  $y^{(i)} f(\mathbf{x}^{(i)}) \geq 1$  set  $\boldsymbol{\theta}^{[t+1]} = (1 - \lambda\alpha)\boldsymbol{\theta}^{[t]}$
  - 6: **end for**
- 

Note the weight decay due to the L2-regularization.

# TRAINING SVM IN THE DUAL

The dual problem of the soft-margin SVM is

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y^{(i)} y^{(j)} \langle \mathbf{x}^{(i)}, \mathbf{x}^{(j)} \rangle \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq C \quad \sum_{i=1}^n \alpha_i y^{(i)} = 0 \end{aligned}$$

We could solve this problem using coordinate ascent. That means we optimize w.r.t.  $\alpha_1$ , for example, while holding  $\alpha_2, \dots, \alpha_n$  fixed.

But: We cannot make any progress since  $\alpha_1$  is determined by  $\sum_{i=1}^n \alpha_i y^{(i)} = 0$ !

# TRAINING SVM IN THE DUAL

**Solution:** Update two variables simultaneously

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y^{(i)} y^{(j)} \langle \mathbf{x}^{(i)}, \mathbf{x}^{(j)} \rangle \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq C \quad \sum_{i=1}^n \alpha_i y^{(i)} = 0 \end{aligned}$$

---

Pairwise coordinate ascent in the dual

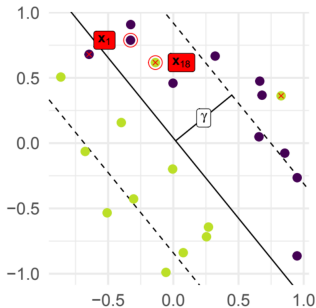
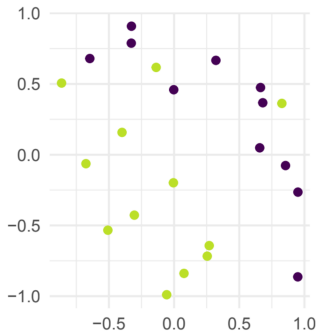
---

- 1: Initialize  $\alpha = 0$  (or more cleverly)
  - 2: **for**  $t = 1, 2, \dots$  **do**
  - 3:     Select some pair  $\alpha_i, \alpha_j$  to update next
  - 4:     Optimize dual w.r.t.  $\alpha_i, \alpha_j$ , while holding  $\alpha_k$  ( $k \neq i, j$ ) fixed
  - 5: **end for**
- 

The objective is quadratic in the pair, and  $s := y^{(i)}\alpha_i + y^{(j)}\alpha_j$  must stay constant. So both  $\alpha$  are changed by same (absolute) amount, the signs of the change depend on the labels.

# TRAINING SVM IN THE DUAL

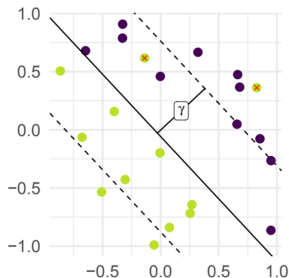
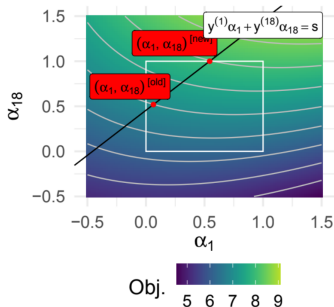
Assume we are in a valid state,  $0 \leq \alpha_i \leq C$ . Then we chose<sup>1</sup> two observations (encircled in red) for the next iteration. Note they have opposite labels so the sign of their change is equal.



# TRAINING SVM IN THE DUAL

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y^{(i)} y^{(j)} \langle \mathbf{x}^{(i)}, \mathbf{x}^{(j)} \rangle \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq C \quad \sum_{i=1}^n \alpha_i y^{(i)} = 0 \end{aligned}$$

We move on the linear constraint until the pair-optimum or the boundary (here:  $C = 1$ ).





# TRAINING SVM IN THE DUAL

Sequential Minimal Optimization (SMO) exploits the fact that effectively we only need to solve a one-dimensional quadratic problem, over in interval, for which an analytical solution exists.

