SUPPORT VECTOR MACHINE OPTIMIZATION

Pei-Yuan Wu
Electrical Engineering Department
National Taiwan University

1

OUT-LINE

- Loss function for SVM
- Gradient Descent Algorithm

OPTIMIZATION

- Learning an SVM has been formulated as a constrained optimization problem over w and ξ

$$\min_{\boldsymbol{w}\in\mathbb{R}^d, b\in\mathbb{R}, \xi_1,\dots,\xi_N\geq 0} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i$$

subject to $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \xi_i$, for $i = 1,\dots,N$

- The constraint $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \xi_i$, can be written more concisely as

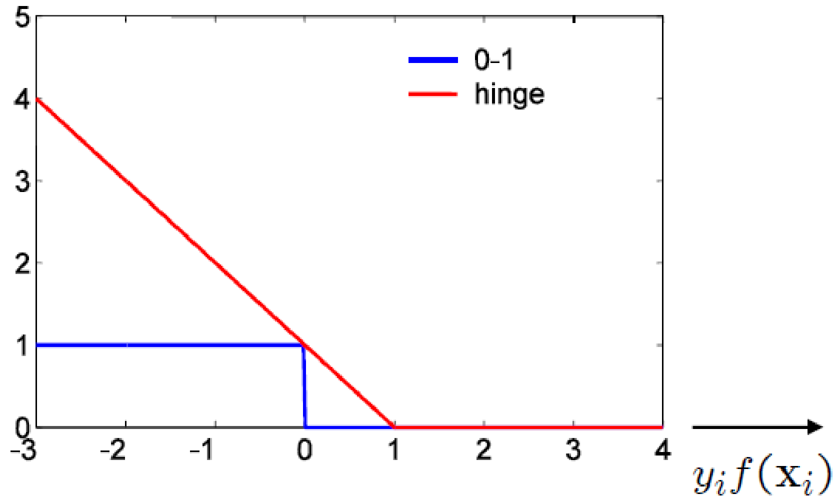
$$y_i f(\boldsymbol{x}_i) \geq 1 - \xi_i$$

which, together with $\xi_i \geq 0$, is equivalent to

$$\xi_i \geq \max(0, 1 - y_i f(\boldsymbol{x}_i))$$

- Hence the learning problem is equivalent to the unconstrained optimization problem over w

$$\min_{\boldsymbol{w}\in\mathbb{R}^d, b\in\mathbb{R}} \underbrace{\frac{1}{2}\|\boldsymbol{w}\|^2}_{\text{regularization}} + C\sum_{i=1}^{N}\underbrace{\max(0, 1 - y_i f(\boldsymbol{x}_i))}_{\text{hinge loss}}$$

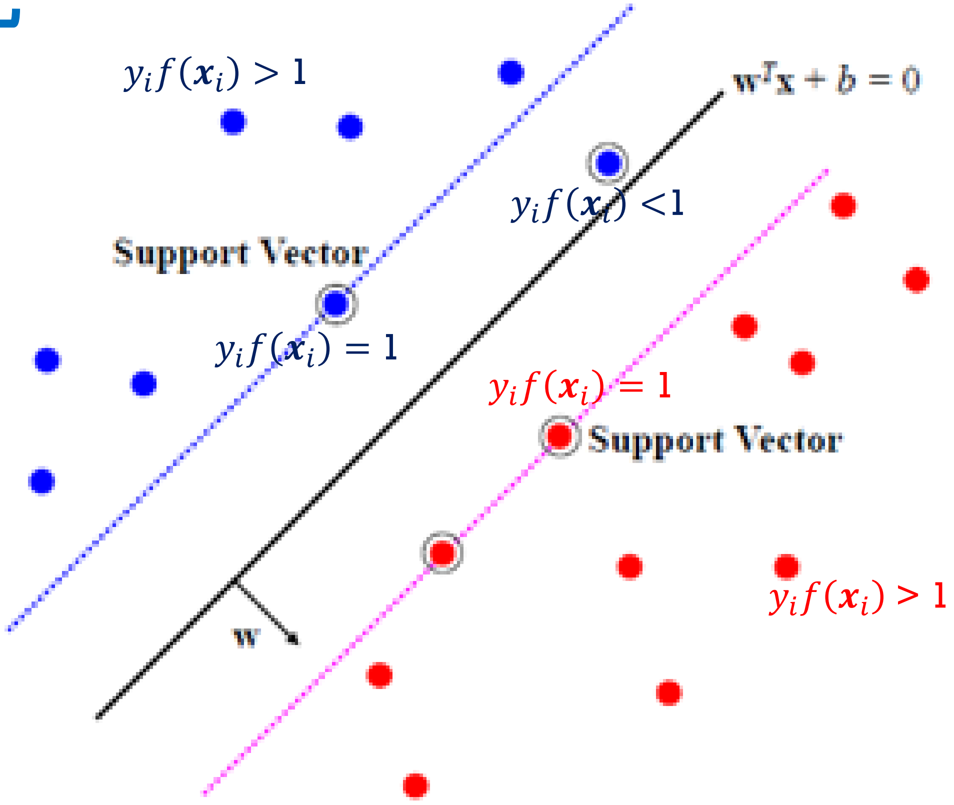


LOSS FUNCTION

$$\min_{\boldsymbol{w}\in\mathbb{R}^d} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N} \underbrace{\max(0, 1 - y_i f(\boldsymbol{x}_i))}_{\text{hinge loss}}$$

Points are in three categories:

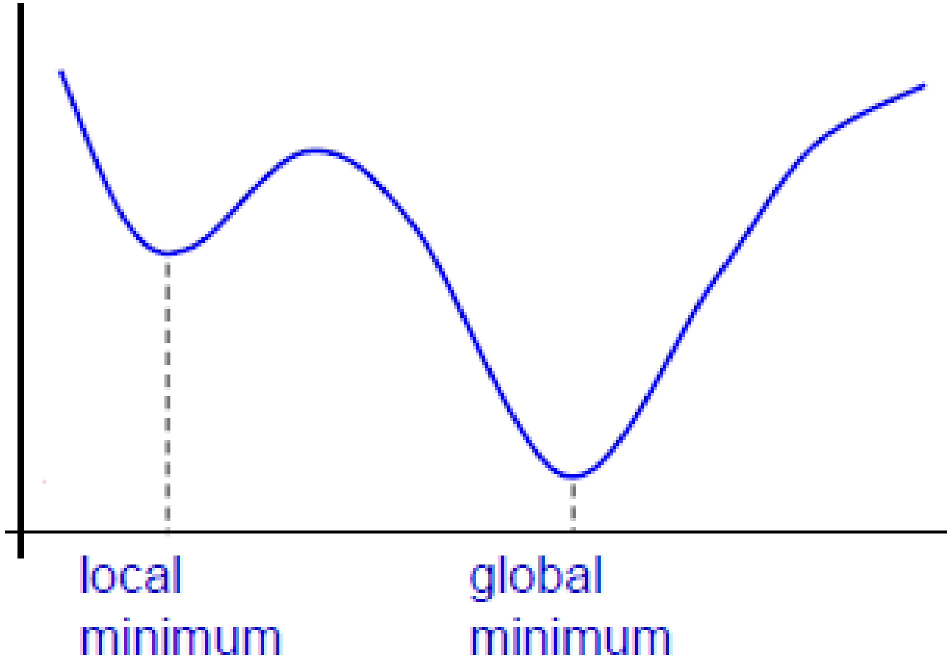
- $y_i f(\boldsymbol{x}_i) > 1$
  - ➢ Point is outside margin.
  - ➢ No contribution to loss
- $y_i f(\boldsymbol{x}_i) = 1$
  - ➢ Point is on margin.
  - ➢ No contribution to loss.
- $y_i f(\boldsymbol{x}_i) < 1$
  - ➢ Point violates margin constraint.
  - ➢ Contributes to loss



OPTIMIZATION CONTINUED

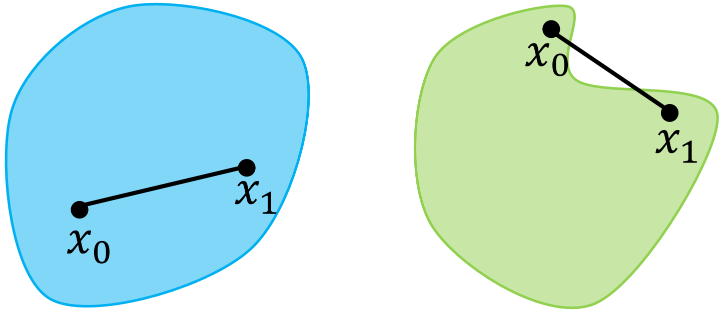
$$\min_{\boldsymbol{w}\in\mathbb{R}^{\boldsymbol{d}},b\in\mathbb{R}} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\max(0, 1 - y_i f(\boldsymbol{x}_i))$$

- Does this loss function have a unique minimal solution?
- Does the solution depend on the starting point of an iterative optimization algorithm (such as gradient descent)?



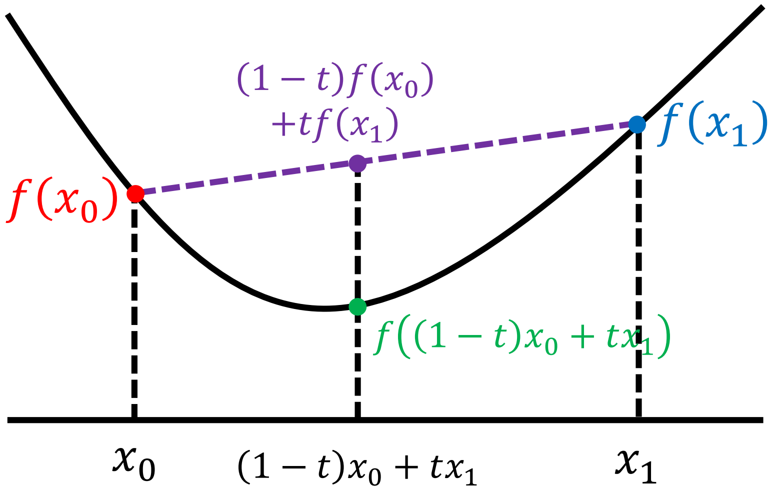
CONVEX FUNCTIONS

We say $\Omega \subset \mathbb{R}^d$ is convex if every $x_0, x_1 \in \Omega$ and $0 \le t \le 1$ satisfy $(1-t)x_0 + tx_1 \in \Omega$



Let Ω be a convex set in $\mathbb{R}^d$. A function $f: \Omega \to \mathbb{R}$ is convex if every $x_0, x_1 \in \Omega$ and $0 \le t \le 1$ satisfy

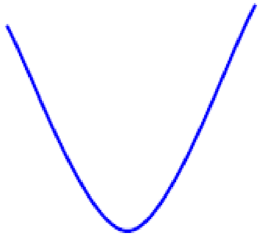
$$f\big((1-t)x_0 + tx_1\big) \le (1-t)f(x_0) + tf(x_1)$$



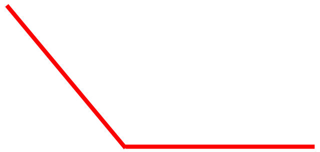
Line joining $(x_0, f(x_0))$ and $(x_1, f(x_1))$ lies above the function graph

CONVEX FUNCTION PROPERTIES

SVM's loss function: $\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\max(0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b))$



+



is convex

Lemma: Let Ω be a convex set in $\mathbb{R}^d$, f, g be convex functions on Ω, $\alpha, \beta \geq 0$, then $\alpha f + \beta g$ is convex.

Proof: For every $x_0, x_1 \in \Omega$ and $0 \leq t \leq 1$, one has

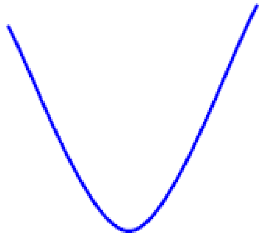
$$f\big((1-t)x_0 + tx_1\big) \leq (1-t)f(x_0) + tf(x_1)$$
$$g\big((1-t)x_0 + tx_1\big) \leq (1-t)g(x_0) + tg(x_1)$$

Hence $(\alpha f + \beta g)\big((1-t)x_0 + tx_1\big) \leq (1-t)(\alpha f + \beta g)(x_0) + t(\alpha f + \beta g)(x_1)$

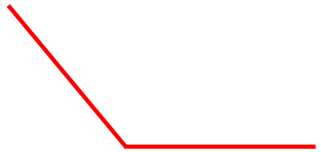
<u>non-negative sum of convex functions is convex</u>

CONVEX FUNCTION PROPERTIES

SVM's loss function: $\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N} \max(0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b))$



+



is convex

➔local optimal is global optimal

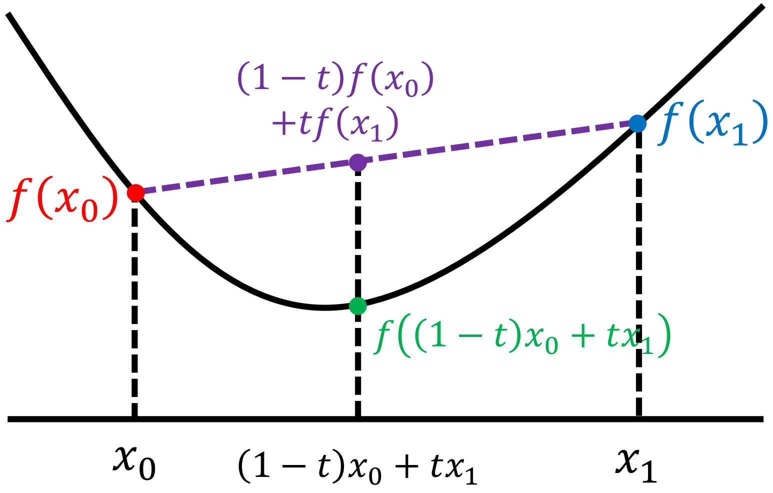
➔Gradient descent always leads to global optimal regardless of initialization.

Lemma: Let Ω be a convex set in $\mathbb{R}^d$, f be a convex function on Ω. If $x_0, x_1 \in \Omega$ are local minimal points of f, then $f(x_0) = f(x_1)$.

Proof: WLOG assume $f(x_0) < f(x_1)$. For all $0 \le t < 1$, one has

$$f\big((1-t)x_0 + tx_1\big) \le (1-t)f(x_0) + tf(x_1) < f(x_1)$$

That is, $f(x) < f(x_1)$ for all $\mathrm{x} \in \{(1-t)x_0 + tx_1 : 0 \le t < 1\}$. Hence x_1 cannot be local minimal, leading to contraction.



If the loss function is convex, then a locally optimal point is globally optimal

GRADIENT DESCENT ALGORITHM FOR SVM

- To minimize a loss function $L(\boldsymbol{w}_t)$ we use the iterative update

$$\boldsymbol{w}_{t+1} \leftarrow \boldsymbol{w}_t - \eta_t \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t)$$

  where η_t is the learning rate (at time t).

- First, rewrite the optimization problem as an average

$$L(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{N}\sum_{i=1}^{N} \max\big(0, 1 - y_i f(\boldsymbol{x}_i)\big)$$

  where $\lambda = 1/(NC)$ and $f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x} + \boldsymbol{b}$

SUB-GRADIENT DESCENT ALGORITHM FOR SVM

$$L(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{n}\sum_{i=1}^{N}\mathcal{L}(\boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{w})$$

- The iterative update is

$$\boldsymbol{w}_{t+1} \leftarrow \boldsymbol{w}_t - \eta_t \nabla_w L(\boldsymbol{w}_t)$$

$$\leftarrow \boldsymbol{w}_t - \eta_t\left(\lambda \boldsymbol{w}_t + \frac{1}{n}\sum_{i=1}^{n}\nabla_w \mathcal{L}(\boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{w}_t)\right)$$

  where η_t is the learning rate.

- In the Pegasos algorithm the learning rate is set at $\eta_t = \frac{1}{\lambda t}$
- Alternative: Stochastic gradient decent.