SUPPORT VECTOR MACHINES & NEURAL NETWORKS

LECTURE 4 – SUPPORT VECTOR MACHINES PART # I

A. Bi-classification

History, LSVM, Approximate LSVM, Soft LSVM,

Kernel-based linear SVM, Nonlinear SVM

B. Multi-classification

OVO, OVA, Twin SVM

C. Prediction

Support vector regression (SVR)

Brief history of SVM

- Early years: (1960's) – taken from ***Measures of Complexity***, Springer, 2015. DOI 10.1007/978-3-319-21852-6

- Role1: Aleksander Yakovlevich Lerner (1913-2004), who was a faculty member associated with the Institute of Automation and Remote Control of the Soviet Academy of Sciences (later renamed as ICS -- the Institute of Control Sciences) and Moscow Institute of Physics and Technology (MIPT).
- Role 2: Alexey Yakovlevich Chervonenkis (1938-2014) who graduated from MIPT and worked in ICS, was given an opportunity to do "pattern recognition" as a PhD student in Lerner's laboratory.
- Role 3: Vladimir Vapnik (1936-present), another PhD student from Tashkent, joined the same laboratory. Vladimir graduated from Uzbek State University and started working in one of the research institutes in Tashkent. Aleksandr Lerner, while on his business trip to Tashkent, was persuaded to take the promising young researcher to Moscow for postgraduate study. The idea was that after receiving his Ph.D. the student would go back to Tashkent and enhance the local research community.
- One way or another, Alexey and Vladimir jointly worked on problems of pattern recognition in Professor Lerner's research group.

Brief history of SVM

- Early years: (1960's) – taken from **Measures of Complexity**, Springer, 2015. DOI 10.1007/978-3-319-21852-6

- In 1962–1964 they invented a new method of prediction and called it "Generalized Portrait." The algorithm constructed a hyperplane to separate classes of patterns. The method of Generalized Portrait could be reduced to work with scalar products of input vectors. At the time the Institute had no digital computers, only analog ones. This created a problem with inputting the initial data. They did this by calculating the scalar products by hand (or using calculators) and inputting them into the analog computers by adjusting corresponding resistors.
- In 1964, Vapnik finished his PhD thesis in Statistics from ICS under the supervision of Lerner.
- Later, starting from 1964, the Institute acquired digital computers, and the method of Generalized Portrait was implemented to solve many different recognition problems in geology, meteorology, and other fields.
- They also worked together to develop the "Vapnik- Chervonenkis (VC) theory of statistical learning".

Brief history of SVM

- Early years: (1960's) – taken from **Measures of Complexity**, Springer, 2015, DOI 10.1007/978-3-319-21852-6

References:

- Vapnik, V.N., Lerner, A. Ya., Recognition of patterns with help of generalized portraits. Avtomat. I Telemekh., 24(6), 774-780 (1963).

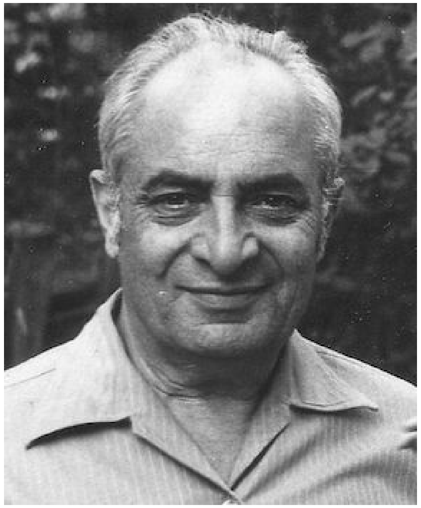
- Vapnik, V.N., Chervonenkis, A.Y.: Об одном классе алгоритмов обучения распознаванию образов (On a class of algorithms for pattern recognition learning, in Russian, English summary). Автоматика и телемеханика (Automation and Remote Control) 25(6), 937–945 (1964)

- Vapnik, V.N., Chervonenkis, A.Y.: On the uniform convergence of the frequencies of occurrence of events to their probabilities. Doklady Akademii Nauk SSSR 181, 781–783 (1968). Soviet Mathematics Doklady 9, 915–918

- Vapnik, V.N., Chervonenkis, A.Y.: Теория распознавания образов: Статистические проблемы обучения (Theory of Pattern Recognition: Statistical Problems of Learning: in Russian). Nauka, Moscow (1974). German translation: Theorie der Zeichenerkennung, transl. K.G. Stöckel and B. Schneider, ed. S. Unger and B. Fritzsch, Akademie Verlag, Berlin (1979)

Who is who

- Collaborations (Photos/information are taken from Wikipedia)



Live in New Jersey, USA died in Moscow, Russia died in Rehovot, Israe

(ATT, NEC, Facebook AI Research) (Royal Holloway, Univ of London) (Weizmann Insti of Science)

Brief history of SVM

- More recent years (1990’s)
- Vladimir Vapnik moved to the US in 1990 to join the Adaptive Systems Research Department at AT&T Bell Labs in Holmdel, New Jersey.

  He started collaborative research with many colleagues to advance the field of SVM such as
- Grace, Boser and Vapnik (1992) proposed soft SVM models with kernels for not-linearly separable applications.
- Vapnik and Cortes (1995) developed the statistical learning theory to officially introduce SVM.
- Showed in 1996 that an SVM classifier is comparable to the best optical character recognition system**.**
- A modified SVM model for regression analysis (SVR) was proposed in 1996.

References:

Cortes, C., Vapnik V.N., Support vector networks, Machine Learning, 20, 273-297 (1995).

Support vector machines (SVM)

- Support vector machines are mainly for pattern recognition in supervised machine learning.

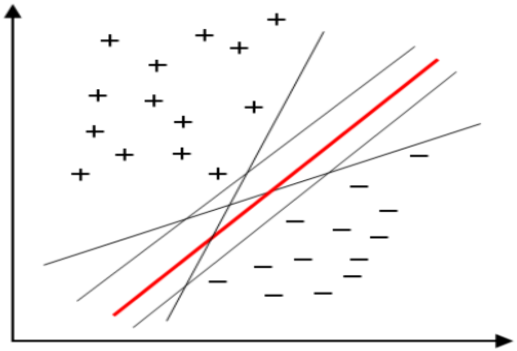
  - SVM is commonly used for classification (recognition, diagnosis, preference, prediction, etc.)
  - SVR means support vector regression
  - SVC means support vector clustering (unsupervised learning)

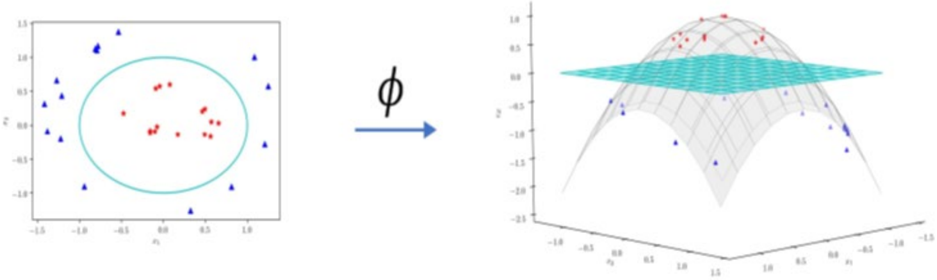
Bi-classification

- Problem facing:

  We have a set of N data points $\{\boldsymbol{x}^1, \boldsymbol{x}^2, \dots, \boldsymbol{x}^N\}$, $\boldsymbol{x}^i \in \mathbb{R}^n$, in two different classes labeled by $y_i \in \{-1, 1\}$, $i = 1, \dots, N$. Given a new data point $\bar{\boldsymbol{x}} \in \mathbb{R}^n$, should we label it with $\bar{y} = 1$ or $\bar{y} = -1$?

  - Decision making: How? and Why?





Motivations and basic ideas

- Are the data points linearly separable?
  - How do we know?
  - If “Yes”, how to separate them apart?
  - If “No”, what to do?

Are the data points nonlinearly separable?

- How to separate them apart ?
- Will the “ARR” principle help?

Linear separability

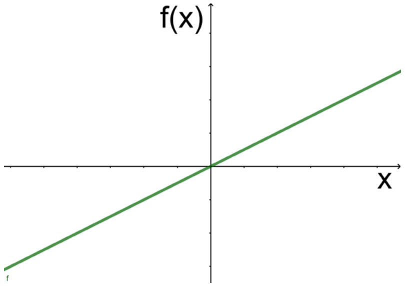
- Linear function: a function $f : \mathbb{R}^n \to \mathbb{R}$ is linear if

$$f(\alpha \mathbf{x} + \beta \mathbf{y}) = \alpha f(\mathbf{x}) + \beta f(\mathbf{y}) \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \quad \alpha, \beta \in \mathbb{R}.$$

Properties:

(i) f is linear if and only if $f(\mathbf{x}) = \boldsymbol{a}^T \mathbf{x}$ for some $\boldsymbol{a} \in \mathbb{R}^n$.

(ii) gra(f) = $\{(x, f(x))\}$ is a hyperplane passing the original point.



(iii) If f and g are linear and $\alpha \in \mathbb{R}$, then $f + g$, $f - g$, αf are linear.

Linear separability

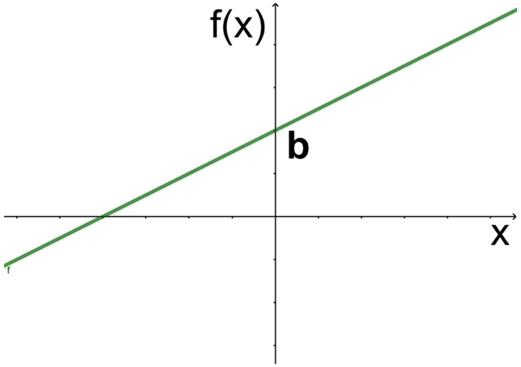
- Affine function: a function $f : \mathbb{R}^n \to \mathbb{R}$ is affine if

$$f(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) = \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y}) \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \quad \alpha \in \mathbb{R}.$$

Properties:

(i) f is affine if and only if $f(\mathbf{x}) = \boldsymbol{a}^T\mathbf{x} + b$ for some $\boldsymbol{a} \in \mathbb{R}^n$, $b \in \mathbb{R}$.

(ii) gra(f) is a hyperplane passing point **b.**

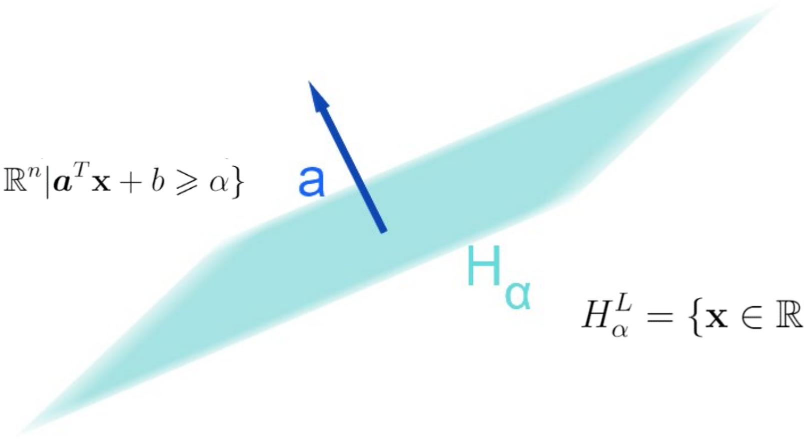


(iii) If f and g are affine and $\alpha \in \mathbb{R}$, then $f + g$, $f - g$, αf are affine.

Contours of affine (linear) function

- Define $H_\alpha = \{\mathbf{x} \in \mathbb{R}^n | \boldsymbol{a}^T \mathbf{x} + b = \alpha\}$

$H_\alpha^U = \{\mathbf{x} \in \mathbb{R}^n | \boldsymbol{a}^T \mathbf{x} + b \geqslant \alpha\}$

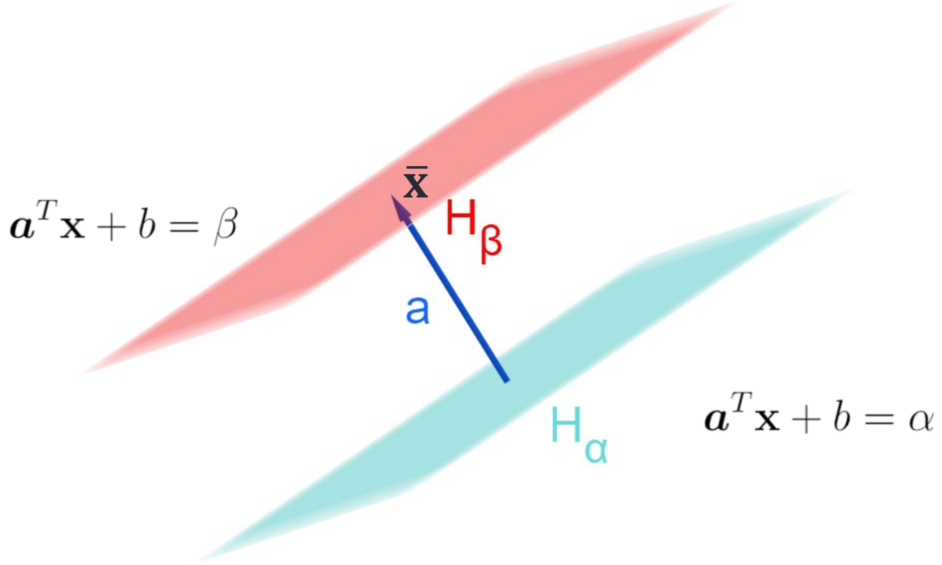


$H_\alpha^L = \{\mathbf{x} \in \mathbb{R}^n | \boldsymbol{a}^T \mathbf{x} + b \leqslant \alpha\}$

- A hyperplane in $\mathbb{R}^n$ with $\boldsymbol{a}$ being its normal vector.
- Moving along $\boldsymbol{a}$ will increase $f(\mathbf{x}) = \boldsymbol{a}^T \mathbf{x} + b, \quad x \rightarrow H_\alpha^U$

Contours of affine function

- Given $\bar{\mathbf{x}} \in \mathbb{R}^n$ and H_α, distance $(\bar{\mathbf{x}}, H_\alpha)$ = ?

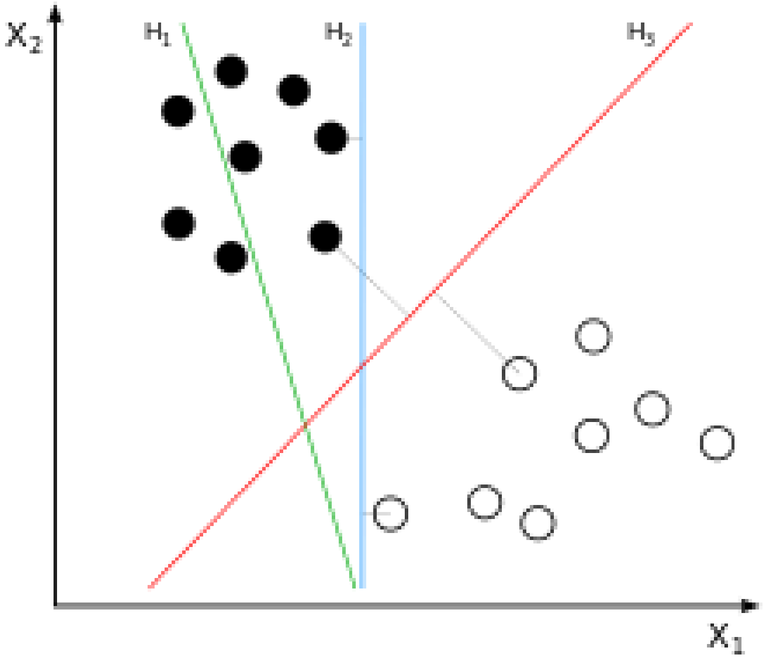


$\boldsymbol{a}^T\mathbf{x} + b = \beta$
$\bar{\mathbf{x}}$
H_β
a
H_α
$\boldsymbol{a}^T\mathbf{x} + b = \alpha$

- Distance between $\bar{\mathbf{x}}$ and H_α is $d(\bar{\mathbf{x}}, H_\alpha) = \dfrac{|\alpha - \beta|}{\|\boldsymbol{a}\|_2}$

Support vector machines – basic ideas

- Linearly separable



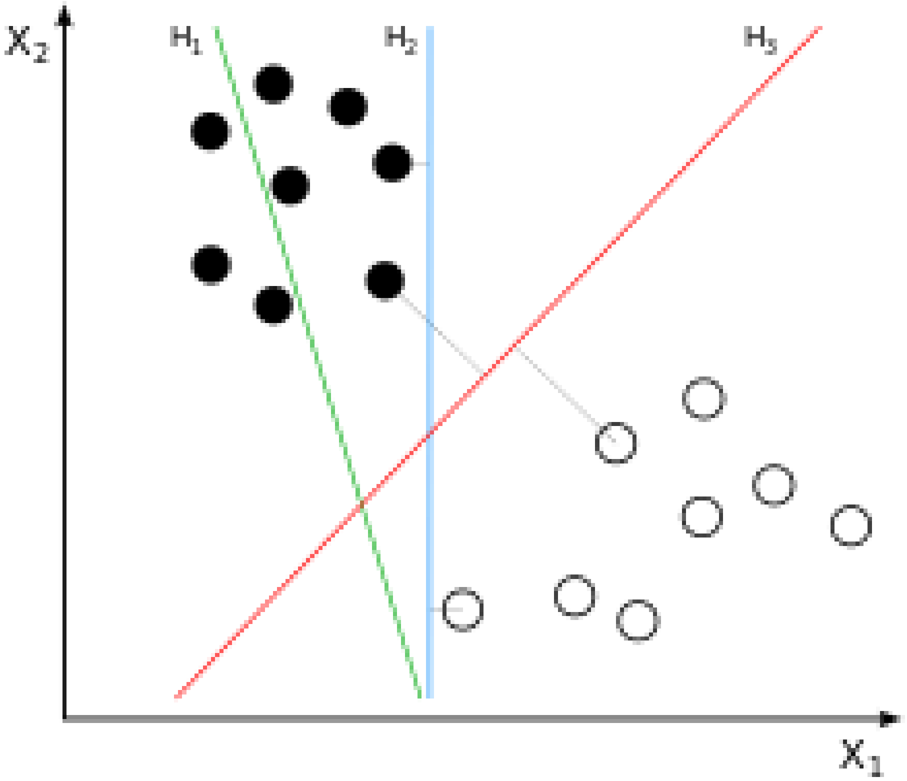
- Given a set of points $\{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$ with binary labels $y_i \in \{-1, 1\}$
- Find a hyperplane that strictly separates the two classes.

$$\begin{aligned} \boldsymbol{a}^T\mathbf{x}^i + b > 0 \quad \text{if } y_i = 1 \\ \boldsymbol{a}^T\mathbf{x}^i + b < 0 \quad \text{if } y_i = -1 \end{aligned}$$

$$y_i(\boldsymbol{a}^T\mathbf{x}^i + b) \geqslant 0, \quad i = 1, \ldots, N.$$

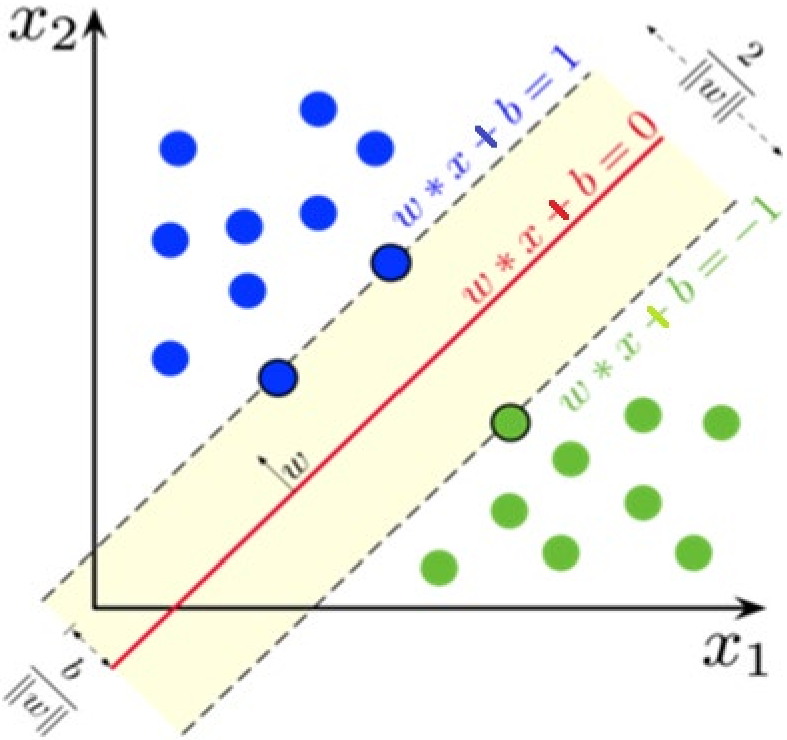
Support vector machines – basic ideas

- Which one to choose? (generalizability)



Linear support vector machine (LSVM) – basic model

- Linear separation with maximum margin (distance)



$$\begin{aligned} \max \quad & \frac{2}{\|\boldsymbol{w}\|_2} \\ s.t. \quad & y_i(\boldsymbol{w}^T \mathbf{x}^i + b) \geqslant 1 \\ & \forall i = 1, \ldots, N. \\ & \boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R}. \end{aligned}$$

equivalently,

$$\begin{aligned} \min \quad & \frac{\|\boldsymbol{w}\|_2}{2} \\ s.t. \quad & y_i(\boldsymbol{w}^T \mathbf{x}^i + b) \geqslant 1 \\ & \forall i = 1, \ldots, N. \\ & \boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R}. \end{aligned}$$

Linear SVM (hard margin) – LSVM model

- Primal LSVM

$$
\begin{aligned}
\min \quad & \frac{1}{2}\,\|\boldsymbol{w}\|_2^2 \\
\text{s.t.} \quad & y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right) \geq 1,\ \ i = 1,2,\dots,N \qquad \text{(LSVM)} \\
& \boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R}
\end{aligned}
$$

- It is a linearly constrained convex quadratic program with $n + 1$ variables and N inequality constraints.

- Implications?

LSVM Classifier

- LSVM provides $(\bar{\boldsymbol{w}}, \bar{b})$ to form a classifier for bi-classification:

- Given an input data point $\boldsymbol{x} \in \mathbb{R}^n$

$$class_{LSVM}\ (\boldsymbol{x}) = sign\ (\bar{\boldsymbol{w}}^T \boldsymbol{x} + \bar{b})$$

where

$$sign\ (\mathrm{y}) = \begin{cases} +1, & \text{if } \ y > 0 \\ -1, & \text{if } \ y < 0 \end{cases}$$

Linear SVM (hard margin) – LSVM model

- What else can be say about LSVM?

  - Dual LSVM

  - Optimality conditions

  - Solution methods

Lagrangian dual approach

- Primal LSVM

$$\begin{aligned} \min \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right) \geq 1,\ i = 1, 2, \ldots, N \quad \text{(LSVM)} \\ & \boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R} \end{aligned}$$

- *Lagrangian multiplier method*:

  - associating the i^{th}constraint, assign a multiplier $\alpha_i \geq 0$ to construct the Lagrangian function

$$L(\boldsymbol{w}, b, \alpha) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \Sigma_{i=1}^N \alpha_i(1 - y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right))$$

* α_i indicates the influence of the data point $(\boldsymbol{x}^i, y_i)$

Lagrangian dual approach

- Stationary point of the Lagrangian function

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \Sigma_{i=1}^{N}\alpha_i(1 - y_i(\boldsymbol{w}^T\boldsymbol{x}^i + b))$$

Lagrangian dual function

$$h(\boldsymbol{\alpha}) \triangleq min_{\boldsymbol{w}\in\mathbb{R}^n,\, b\in\mathbb{R}}\, L(\boldsymbol{w}, b, \alpha)$$

- Optimality conditions:

$$\nabla_{\boldsymbol{w}} L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = 0 \Longrightarrow \boldsymbol{w} = \Sigma_{i=1}^{N}\alpha_i y_i \boldsymbol{x}^i$$

$$\nabla_b L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = 0 \Longrightarrow \Sigma_{i=1}^{N}\alpha_i y_i = 0$$

⇒ dual objective function

$$h(\boldsymbol{\alpha}) = -\frac{1}{2}\,(\Sigma_{i=1}^{N}\alpha_i y_i \boldsymbol{x}^i)^T\, \Sigma_{i=1}^{N}\alpha_i y_i \boldsymbol{x}^i + \Sigma_{i=1}^{N}\alpha_i$$

Lagrangian dual approach

KKT conditions for LSVM:

- Stationarity

$$\boldsymbol{w} = \Sigma_{i=1}^{N} \alpha_i y_i \boldsymbol{x}^i \quad \text{and} \quad \Sigma_{i=1}^{N} \alpha_i y_i = 0$$

- Primal feasibility

$$y_i\left(\boldsymbol{w}^T \boldsymbol{x}^i + b\right) \geq 1, \qquad i = 1, 2, \dots, N$$

- Dual feasibility

$$\alpha_i \geq 0, \qquad i = 1, 2, \dots, N$$

- Complementary slackness

$$\alpha_i(1 - y_i\left(\boldsymbol{w}^T \boldsymbol{x}^i + b\right)) = 0$$

Dual linear SVM (DLSVM)

- Lagrangian dual model

$$\begin{aligned}
\max \quad & -\tfrac{1}{2}\textstyle\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i y_i (\boldsymbol{x}^i)^T \boldsymbol{x}^j y_j \alpha_j + \sum_{i=1}^{N}\alpha_i \\
s.t. \quad & \textstyle\sum_{i=1}^{N}\alpha_i y_i = 0 \qquad\qquad\qquad\qquad \text{(DLSVM)} \\
& \alpha_i \geq 0, i = 1, \dots, N
\end{aligned}$$

- The Hessian of the dual objective function

$h(\boldsymbol{\alpha}) = -\frac{1}{2}\boldsymbol{\alpha}^T H \boldsymbol{\alpha} + \sum_{i=1}^{N}\alpha_i$ is

$$H = Diag(y)X^T X Diag(y) \succcurlyeq 0$$

- DLSVM is a convex quadratic program with N nonnegative variables and 1 linear equality constraint.

LSVM or DLSVM ?

- Which one to solve? Why?

  - LSVM or DLSM?

  - how about $n \gg N$ and $N \gg n$?

- How are they related?

  - *primal* – *dual* relation

Relations of LSVM and DLSVM

Key relations:

1. Convex QP pair means there is no duality gap!

2. Complementary slackness says that

$$\alpha_i\big(y_i(\boldsymbol{w}^T\boldsymbol{x}^i + b)\ - 1\big) = 0,\ \forall i = 1, 2, \dots, N$$

(a) $\alpha_i = 0$ holds for data point $\boldsymbol{x}^i$ not on separation hyperplane

(inactive constraint means $\boldsymbol{x}^i$ plays no role)

(b) $\alpha_i > 0$ means the point $\boldsymbol{x}^i$ lies on separation hyperplane

(active constraint means $\boldsymbol{x}^i$ is a supporting vector)

3. Dual to primal conversion says that

$$\boldsymbol{w} = \ \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}^i$$

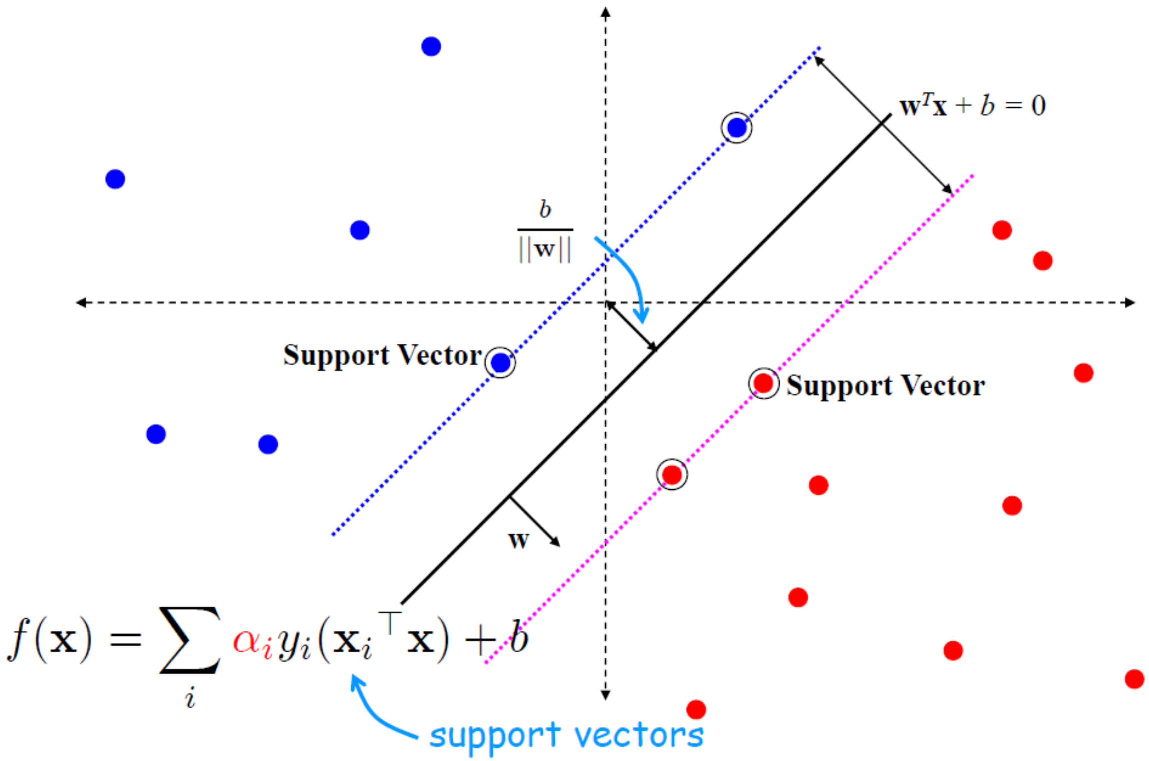
For a point $\boldsymbol{x}^i$ on the hyperplane, since $y_i^2 = 1$,

$$y_i\big(\boldsymbol{w}^T\boldsymbol{x}^i + b\big) = 1 \ \Leftrightarrow \boldsymbol{w}^T\boldsymbol{x}^i + b = y_i$$

$$\Leftrightarrow b = y_i - \boldsymbol{w}^T\boldsymbol{x}^i$$

Supporting vectors

- Picture from “C19 Machine Learning Hilary 2015 A. Zisserman”


$\mathbf{w}^T\mathbf{x} + b = 0$
$\frac{b}{||\mathbf{w}||}$
Support Vector
Support Vector
$\mathbf{w}$
$f(\mathbf{x}) = \sum_i \alpha_i y_i (\mathbf{x}_i{}^\top \mathbf{x}) + b$
support vectors

Dual LSVM Classifier

- DLSVM provides $\bar{\boldsymbol{\alpha}} \in \mathbb{R}_{+}^{N}$ to form a classifier of bi-classification by taking $S = \{\, i \mid \bar{\alpha}_i > 0,\ \ i = 1, \dots, N\}$ and $\bar{b} = y_k \ - (\sum_{i \in S} \bar{\alpha}_i y_i \boldsymbol{x}^i)^T \boldsymbol{x}^k$ for any particular $k \in S$.

- Given an input data point $\boldsymbol{x} \ \in \mathbb{R}^n$

$$class_{DLSVM}\ (\boldsymbol{x}) = sign(\sum_{i \in S} \bar{\alpha}_i\ y_i (\boldsymbol{x}^i)^T \boldsymbol{x}\ + \bar{b})$$

where

$$sign\ (\mathrm{y}) = \begin{cases} +1, & \text{if } \ y > 0 \\ -1, & \text{if } \ y < 0 \end{cases}$$

Primal LSVM vs. Dual LSVM

- SVM classifier

$$class_{SVM}(\boldsymbol{x}) = sign(f(\boldsymbol{x}))$$

- Primal version (LSVM)

$f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x} + b$: *learning from data the normal vector and intercept*

- Dual version (DLSVM)

$f(\boldsymbol{x}) = \sum_{i \in S} \alpha_i y_i (\boldsymbol{x}^i)^T \boldsymbol{x} + \bar{b}$

: *learning from data the role of each data point*

Primal LSVM vs. Dual LSVM

- Primal version (LSVM)

$$f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$$

- Dual version (DLSVM)

$$f(\boldsymbol{x}) = \sum_{i\in S} \alpha_i y_i (\boldsymbol{x}^i)^T\boldsymbol{x} + \bar{b}$$

Potentials of DLSVM:

1. Its dimensionality is fixed !

    -- N variables and one linear equality constraint

    -- solely determined by the number of data points N

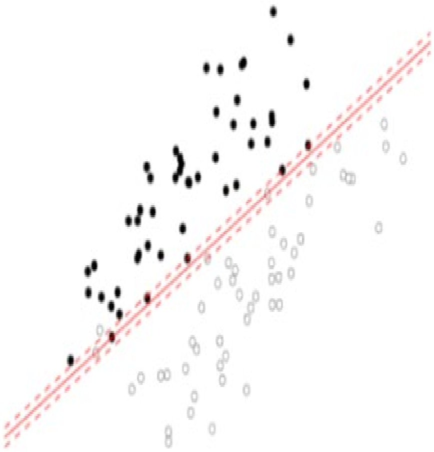
    -- independent of the size of each data point n.

2. The set $S = \{\alpha_i \mid \alpha_i > 0\}$ is in general very sparse!

    -- easy to store and update

Linearly Separable ?

- How do we know a given dataset is linearly separable?
- How difficult to check the linear separability?



$$\min \frac{1}{2} \|\boldsymbol{w}\|_2^2$$

$$\text{s.t. } y_i\left(\boldsymbol{w}^T \boldsymbol{x}^i + b\right) \geq 1,$$

$$i = 1, 2, \dots, N$$

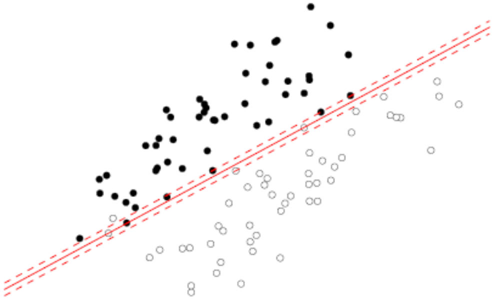
$$\boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R}$$

Approximate LSVM – basic idea

- Not necessarily linearly separable

  - approximate linear separation by imposing penalty

$$\min \quad \sum_{i=1}^{N} \max\{0, 1 - y_i(\boldsymbol{a}^T \mathbf{x}^i + b)\}$$



- penalty $1 - y_i(\boldsymbol{a}^T \mathbf{x}^i + b)$ for misclassifying point $\mathbf{x}^i$.
- can be interpreted as a heuristic for minimizing # of misclassified points.
- a piecewise-linear minimization problem with variables $\boldsymbol{a}, b$.

Approximate LSVM

- Any implications?

  - checking linear separability (1st run)

  - generalizability (re-run LSVM?)

- How to develop an approximate LSVM?

  - Is this a difficult problem, why?

    -- nonlinear and non-differentiable objective function

    -- $n + 1$ real variables

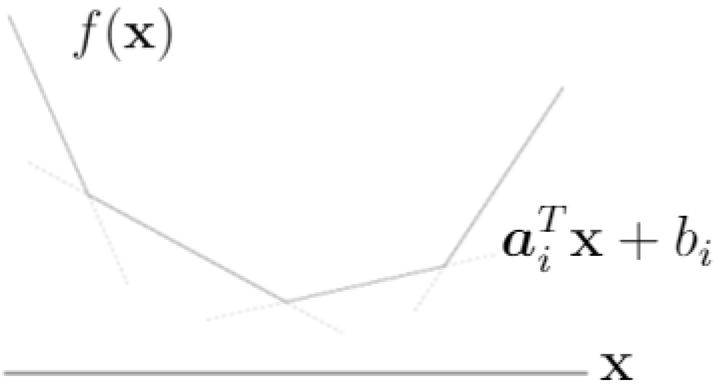
  - How to handle the problem?

Piecewise linear (affine) function

- Piecewise linear (affine) function:

$f : \mathbb{R}^n \to \mathbb{R}$ is (convex) **piecewise-linear** if it can be expressed as

$$f(\mathbf{x}) = \max_{i=1,\ldots,m} (\boldsymbol{a}_i^T \mathbf{x} + b_i)$$



Piecewise linear (affine) function

Properties:

(i) A piecewise linear function is a convex function.

(ii) If f and g are piecewise linear, then $f + g$ is piecewise linear.

$$f(\mathbf{x}) + g(\mathbf{x}) = \max_{i=1,\ldots,m} (\boldsymbol{a}_i^T \mathbf{x} + b_i) + \max_{i=1,\ldots,p} (\boldsymbol{c}_i^T \mathbf{x} + d_i)$$

$$= \max_{\substack{i=1,\ldots,m \\ j=1,\ldots,p}} ((\boldsymbol{a}_i + \boldsymbol{c}_j)^T \mathbf{x} + (b_i + d_j))$$

Commonly seen:

** Absolute value function $|x| = \max\{-x, x\}$ is piecewise linear.

** 1-norm $\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$ is piecewise linear.

** max-norm $\|\mathbf{x}\|_\infty = \max\{|x_1|, \ldots, |x_n|\}$ is piecewise linear.

Minimizing a piecewise linear function

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x}) = \max_{i=1,\ldots,m} (\boldsymbol{a}_i^T\mathbf{x} + b_i)$$

- **equivalent LP** (with varables $\mathbf{x}$ and auxiliary scalar variable t)

$$\begin{array}{ll} \min & t \\ s.t. & \boldsymbol{a}_i^T\mathbf{x} + b_i \leqslant t, \quad i = 1, \ldots, m. \\ & \mathbf{x} \in \mathbb{R}^n, t \in \mathbb{R}. \end{array}$$

Is our problem difficult?

- Minimizing a piecewise linear function

$$\min \Sigma_{i=1}^{N} \max\{0, 1 - y_i(\boldsymbol{w}^T \boldsymbol{x} + b)\}$$

- Equivalent LP $(n + 1 \text{ free variables}, N \text{ nongegative variables},$
$N \text{ inequality constraints})$

$$\begin{aligned} \min \quad & \Sigma_{i=1}^{N} t_i \\ s.t. \quad & 1 \text{ - } y_i(\boldsymbol{w}^T x^i + b) \leq t_i, i = 1, \dots, N \\ & t_i \geq 0, i = 1, 2, \dots, N \qquad \text{(Approximate LSVM)} \\ & \boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R} \end{aligned}$$

or

$$\begin{aligned} \min \quad & \Sigma_{i=1}^{N} \xi_i \\ s.t. \quad & y_i\left(\boldsymbol{w}^T x^i + b\right) \geq 1 - \xi_i, i = 1, \dots, N \\ & \boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R}, \boldsymbol{\xi} \in \mathbb{R}_+^N \end{aligned}$$

Minimizing a piecewise linear function over a linear system

$$\begin{aligned} \min \quad & f(\mathbf{x}) = \max_{i=1,\ldots,m} (\boldsymbol{a}_i^T \mathbf{x} + b_i) \\ s.t. \quad & M\mathbf{x} = \boldsymbol{d} \\ & \mathbf{x} \in \mathbb{R}^n \end{aligned}$$

- **equivalent LP** (with varables $\mathbf{x}$ and auxiliary scalar variable t)

$$\begin{aligned} \min \quad & t \\ s.t. \quad & \boldsymbol{a}_i^T \mathbf{x} + b_i \leqslant t, \quad i = 1, \ldots, m. \\ & M\mathbf{x} = \boldsymbol{d} \\ & \mathbf{x} \in \mathbb{R}^n, t \in \mathbb{R}. \end{aligned}$$

Minimizing a sum of piecewise linear functions

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x}) + g(\mathbf{x}) = \max_{i=1,\ldots,m} (\boldsymbol{a}_i^T\mathbf{x} + b_i) + \max_{i=1,\ldots,p} (\boldsymbol{c}_i^T\mathbf{x} + d_i)$$

$$= \max_{\substack{i=1,\ldots,m \\ j=1,\ldots,p}} ((\boldsymbol{a}_i + \boldsymbol{c}_j)^T\mathbf{x} + (b_i + d_j))$$

- **equivalent LP** with $m + p$ inequalities

$$\begin{aligned} \min \quad & t_1 + t_2 \\ s.t. \quad & \boldsymbol{a}_i^T\mathbf{x} + b_i \leqslant t_1, \quad i = 1, \ldots, m \\ & \boldsymbol{c}_i^T\mathbf{x} + d_i \leqslant t_2, \quad i = 1, \ldots, p \\ & \mathbf{x} \in \mathbb{R}^n, t_1 \in \mathbb{R}, t_2 \in \mathbb{R}. \end{aligned}$$

Minimizing a sum of piecewise linear functions over a linear system

$$
\begin{aligned}
\min \quad & f(\mathbf{x}) + g(\mathbf{x}) = \max_{i=1,\ldots,m} (\boldsymbol{a}_i^T \mathbf{x} + b_i) + \max_{i=1,\ldots,p} (\mathbf{c}_i^T \mathbf{x} + d_i) \\
s.t. \quad & M\mathbf{x} = \boldsymbol{d} \\
& \mathbf{x} \in \mathbb{R}^n.
\end{aligned}
$$

- **equivalent LP** with $m + p$ inequalities and a linear system.

$$
\begin{aligned}
\min \quad & t_1 + t_2 \\
s.t. \quad & \boldsymbol{a}_i^T \mathbf{x} + b_i \leqslant t_1, \quad i = 1, \ldots, m \\
& \mathbf{c}_i^T \mathbf{x} + d_i \leqslant t_2, \quad i = 1, \ldots, p \\
& M\mathbf{x} = \boldsymbol{d} \\
& \mathbf{x} \in \mathbb{R}^n, t_1 \in \mathbb{R}, t_2 \in \mathbb{R}.
\end{aligned}
$$

Minimizing 1-norm function

$$\min \quad \|A\mathbf{x} - \boldsymbol{b}\|_1$$

- ℓ_1**-norm** of m-vector $\mathbf{y}$ is

$$\|\mathbf{y}\|_1 = \sum_{i=1}^{m} |y_i| = \sum_{i=1}^{m} \max\{y_i, -y_i\}.$$

- **equivalent LP** (with variable $\mathbf{x}$ and auxiliary vector variable $\boldsymbol{u} = [u_1, \ldots, u_m]^T$.

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} u_i \\ s.t. \quad & -\boldsymbol{u} \leqslant A\mathbf{x} - \boldsymbol{b} \leqslant \boldsymbol{u} \\ & \mathbf{x} \in \mathbb{R}^n, \boldsymbol{u} \in \mathbb{R}^m. \end{aligned}$$

Minimizing 1-norm over a linear system

- Sparse signal recovery via 1-norm minimization
  - **Estimation by l_1-norm minimization**: compute estimate by solving

$$\begin{aligned} \min \quad & \|\mathbf{x}\|_1 \\ s.t. \quad & A\mathbf{x} = \boldsymbol{b}. \end{aligned}$$

Estimate is signal with smallest l_1-norm, consistent with measurements.

  - **Equivalent LP** (variables $\mathbf{x}, \boldsymbol{u} \in \mathbb{R}^n$)

$$\begin{aligned} \min \quad & \mathbf{1}^T \boldsymbol{u} \\ s.t. \quad & -\boldsymbol{u} \leqslant \mathbf{x} \leqslant \boldsymbol{u} \\ & A\mathbf{x} = \boldsymbol{b} \\ & \mathbf{x} \in \mathbb{R}^n, \boldsymbol{u} \in \mathbb{R}^n. \end{aligned}$$

Minimizing max-norm function

$$\min \quad \|A\mathbf{x} - \boldsymbol{b}\|_\infty$$

- ℓ_∞**-norm** of m-vector $\mathbf{y}$ is

$$\|\mathbf{y}\|_\infty = \max_{i=1,\ldots,m} |y_i| = \max_{i=1,\ldots,m} \max\{y_i, -y_i\}.$$

- **equivalent LP** (with variable $\mathbf{x}$ and auxiliary scalar variable t.

$$\begin{aligned} \min \quad & t \\ s.t. \quad & -t\mathbf{1} \leqslant A\mathbf{x} - \boldsymbol{b} \leqslant t\mathbf{1} \\ & \mathbf{x} \in \mathbb{R}^n, t \in \mathbb{R}. \end{aligned}$$

Minimizing max-norm over a linear system

$$\begin{aligned} \min \quad & \|A\mathbf{x} - \boldsymbol{b}\|_\infty \\ s.t. \quad & M\mathbf{x} = \boldsymbol{d} \\ & \mathbf{x} \in \mathbb{R}^n. \end{aligned}$$

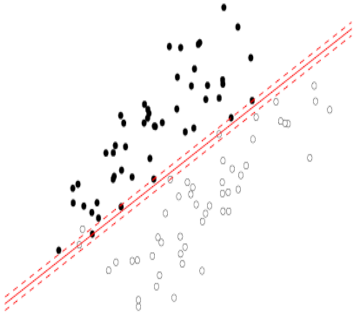
- **equivalent LP** (with variable $\mathbf{x}$ and auxiliary scalar variable t.

$$\begin{aligned} \min \quad & t \\ s.t. \quad & -t\mathbf{1} \leqslant A\mathbf{x} - \boldsymbol{b} \leqslant t\mathbf{1} \\ & M\mathbf{x} = \boldsymbol{d} \\ & \mathbf{x} \in \mathbb{R}^n, t \in \mathbb{R}. \end{aligned}$$

Approximate LSVM considering generalizability

- Basic Idea: Open the margin to allow violation with penalized tolerance.
- Original model

$$\min \quad \sum_{i=1}^{N} \max\{0, 1 - y_i(\boldsymbol{a}^T \mathbf{x}^i + b)$$



- New model

$$\min \ \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{N} max\{0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}^i + b)\}$$

where $C > 0$ is a given parameter.

** C is an indicator emphasizing possible violations.

When $C \rightarrow +\infty$, new model returns to the original model.

Linear SVM with soft margin

- Reformulate the new model

$$\min \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{N} max\{0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}^i + b)\}$$

by allowing violations $y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right) < 1$ (a soft margin)

- Linear soft SVM

$$\min \ \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{N}\xi_i$$

s.t. $y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right) \geq 1 - \xi_i, i = 1, \dots, N$ (LSSVM)

$$\boldsymbol{w} \in \mathbb{R}^n, \ b \in \mathbb{R}, \ \boldsymbol{\xi} \in \mathbb{R}_+^N$$

where $C > 0$ is a given parameter.

** When $C \to +\infty$, $\boldsymbol{\xi} \to \mathbf{0}$ and LSSVM becomes LSVM, but it may fail.

Linear soft SVM (LSSVM)

- Geometric meaning and complexity

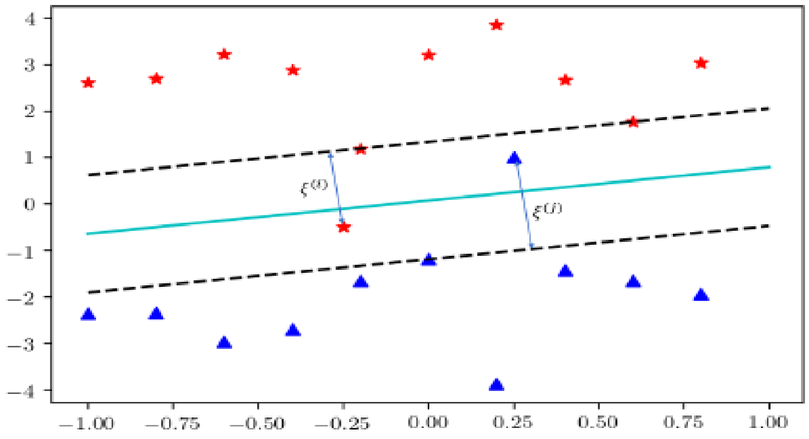
$$\min \ \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{N}\xi_i$$

$$s.t. \ \ y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right) \geq 1 - \xi_i, i = 1, \ldots, N \quad \text{(LSSVM)}$$

$$\boldsymbol{w} \in \mathbb{R}^n, \ \ b \in \mathbb{R}, \ \ \boldsymbol{\xi} \in \mathbb{R}_+^N$$

where $C > 0$ is a given parameter.

- Linearly constrained convex quadratic program with $n + 1 + N$ variables and N inequality constraints.



LSVM vs. LSSVM

- LSVM works only for those linearly separable datasets.

  -- Why?
- LSSVM is always feasible even a dataset is not linearly separable.

  -- Why?
- For a linearly separable dataset, will LSVM and LSSVM produce the same separation hyperplane?

  -- Why?
- LSSVM has N more nonnegative variables than LSVM. What can we expect to meet for the dual LSSVM?

  -- N more constraints?

Lagrangian dual approach

- Stationary point of the Lagrangian function

$$L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\,\Sigma_{i=1}^N \xi_i + \Sigma_{i=1}^N \alpha_i \left(1 - \xi_i - y_i\left(\boldsymbol{w}^T \boldsymbol{x}^i + b\right)\right) - \Sigma_{i=1}^N \theta_i \xi_i$$

where $\alpha_i \geq 0$ and $\theta_i \geq 0$.

Lagrangian dual function

$$h(\boldsymbol{\alpha}, \boldsymbol{\theta}) \triangleq min_{\boldsymbol{w} \in \mathbb{R}^n,\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}_+^N} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\theta})$$

Optimality conditions:

$$\nabla_{\boldsymbol{w}} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\theta}) = 0 \Longrightarrow \boldsymbol{w} = \Sigma_{i=1}^N \alpha_i y_i \boldsymbol{x}^i$$

$$\nabla_b L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\theta}) = 0 \Longrightarrow \Sigma_{i=1}^N \alpha_i y_i = 0$$

$$\nabla_{\xi} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\theta}) = 0 \Longrightarrow C - \alpha_i = \theta_i \geq 0$$

$$\Leftrightarrow \alpha_i \leq C$$

⇒ dual objective function

$$h(\boldsymbol{\alpha}) = -\frac{1}{2}\left(\Sigma_{i=1}^N \alpha_i y_i \boldsymbol{x}^i\right)^T \Sigma_{j=1}^N \alpha_j y_j \boldsymbol{x}^j + \Sigma_{i=1}^N \alpha_i$$

Dual linear soft SVM (DLSSVM)

- Lagrangian dual model

$$\max \quad -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i y_i((\boldsymbol{x}^i)^T\boldsymbol{x}^j)y_j\alpha_j + \sum_{i}^{N}\alpha_i$$

$$s.t. \quad \sum_{i=1}^{N}\alpha_i y_i = 0 \qquad \text{(DLSSVM)}$$

$$0 \le \alpha_i \le C, \quad i = 1, 2, \dots, N$$

- The Hessian of the objective function in $\boldsymbol{\alpha}$ is

$$H = Diag(y)X^TXDiag(y) \succcurlyeq 0$$

- DLSSM is convex quadratic program with N bounded variables and 1 linear equality constraint.
- The quadratic term is determined by an $N \times N$ (kernel) matrix (in terms of the # of data points)

$K = X^TX$ with $K_{ij} = (\boldsymbol{x}^i)^T\boldsymbol{x}^j$ (regardless the dimensionality of each data point $\boldsymbol{x}^i$).

Relations of LSSVM and DLSSVM

Key relations:

1. Convex QP pair means there is no duality gap!
2. Complementary slackness says that

$$\alpha_i\left(y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i+b\right)-1+\xi_i\right)=0,\ \forall i=1,2,\dots,N$$

(a) $\alpha_i=0$ holds for data point $\boldsymbol{x}^i$ with $y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i+b\right)>1-\xi_i$

(inactive constraint means such $\boldsymbol{x}^i$ plays no role)

(b) $C>\alpha_i>0$ means the point $\boldsymbol{x}^i$ with $y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i+b\right)=1-\xi_i\leq 1$

(active constraint means $\boldsymbol{x}^i$ is a supporting vector)

(c) support vectors are $\boldsymbol{x}^i$s with $y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i+b\right)\leq 1$ including those corresponding to $C>\alpha_i>0$.

3. Dual to primal conversion says that

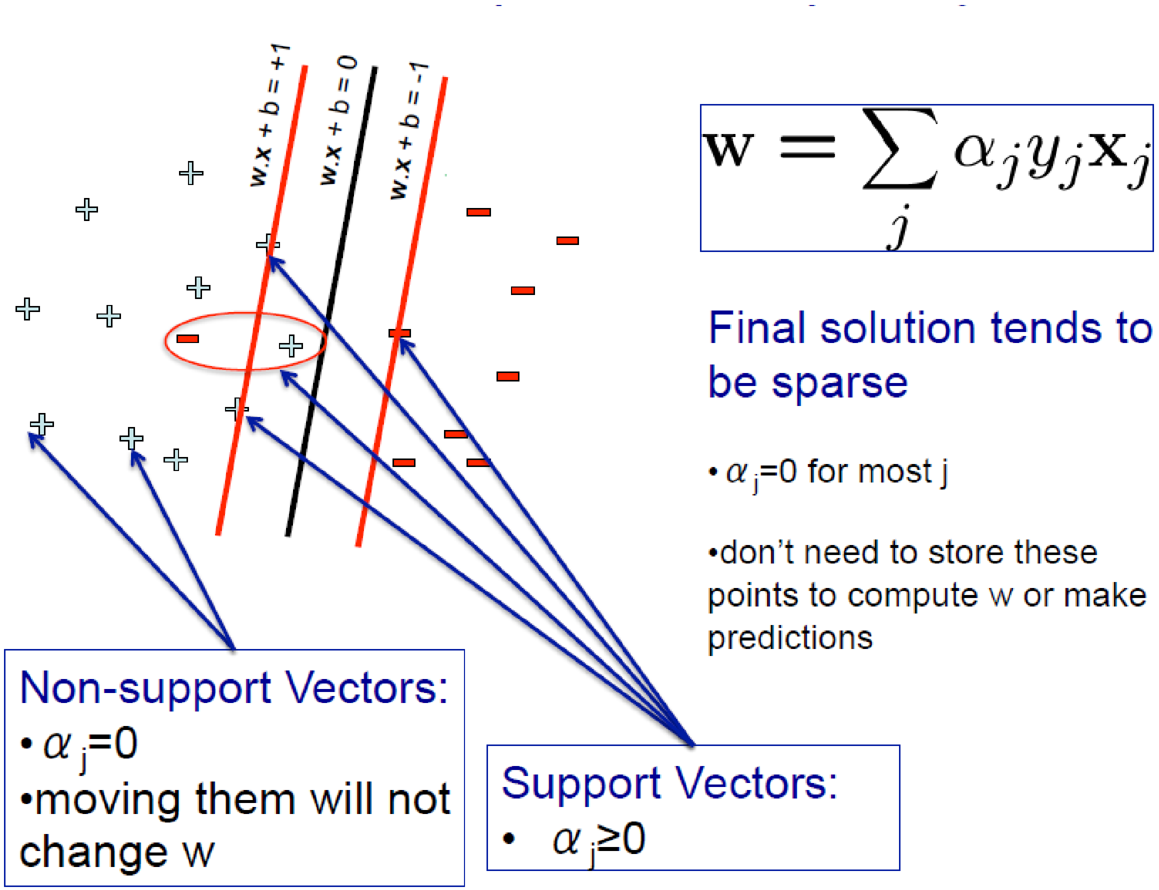
$$\boldsymbol{w}=\sum_{i=1}^{N}\alpha_i y_i\boldsymbol{x}^i$$

For a point $\boldsymbol{x}^i$ on the hyperplane $H_1\ or\ H_{-1}$, since $y_i^2=1$,

$$y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i+b\right)=1\ \Leftrightarrow \boldsymbol{w}^T\boldsymbol{x}^i+b=y_i\ \Leftrightarrow b=y_i-\boldsymbol{w}^T\boldsymbol{x}^i$$

Dual LSSVM

- Picture taken from David Sontag, SVM & Kernels Lecture 6.



$$\mathbf{w} = \sum_j \alpha_j y_j \mathbf{x}_j$$

Final solution tends to be sparse

- α_j=0 for most j
- don’t need to store these points to compute w or make predictions

Non-support Vectors:

- α_j=0
- moving them will not change w

Support Vectors:

- $\alpha_j \geq 0$

LSSVM vs. DLSSVM ?

- Which one to solve? Why?

  - LSSVM or DLSSM?

  - how about $n \gg N$ and $N \gg n$?

- What's the effect of choosing different parameter value of C?

- Classifier?

  - $class_{LSSVM}(\boldsymbol{x}) = ?$

  - $class_{DLSSVM}(\boldsymbol{x}) = ?$

Comparisons and discussions

- LSVM vs. Approximate LSVM
  - applicability?
  - equivalency?
  - complexity?

- LSVM vs. LSSVM

- LSSVM vs. Approximate LSVM