LECTURE 9: CONSTRAINED NLP APPLICATIONS

Constrained optimization models for machine learning

1. Support vector machines for data classification
2. Support vector regression for data regression
3. Neural networks

Support vector machines (SVM)

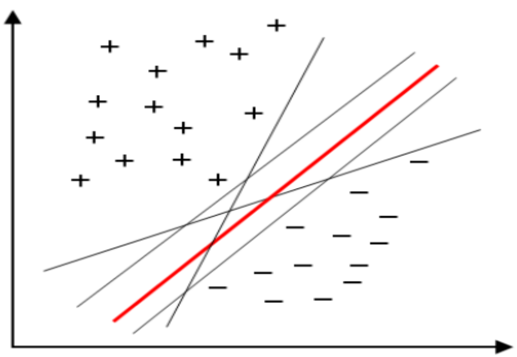
- Support vector machines are mainly for pattern recognition in supervised machine learning.
  - SVM is commonly used for classification (recognition, diagnosis, preference, prediction, etc.)
  - SVR means support vector regression
  - SVC means support vector clustering (unsupervised learning)

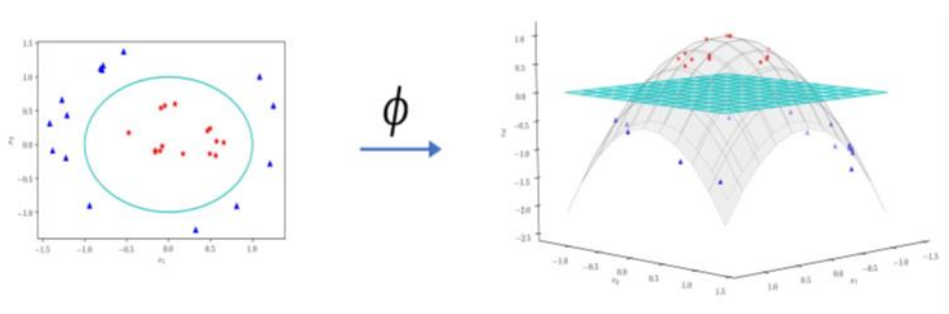
Bi-classification

- 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?

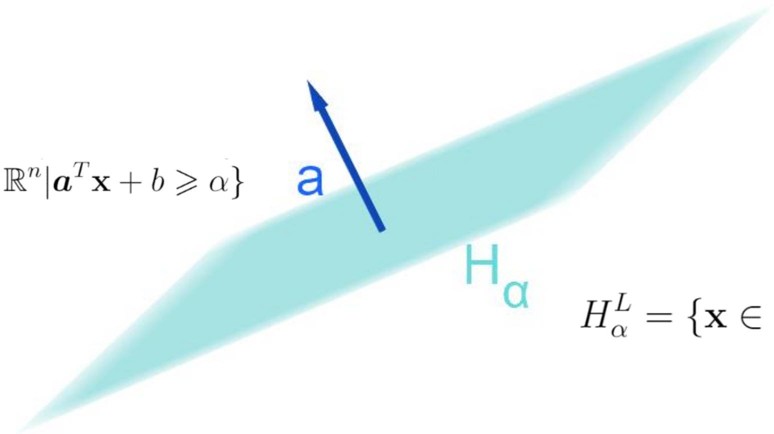




Contours of affine (linear) function

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

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

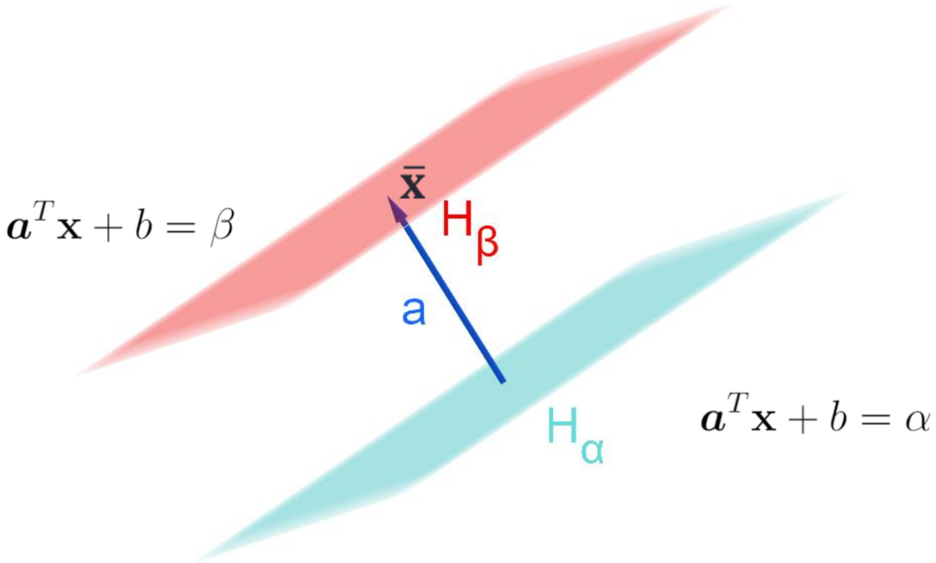


$H_\alpha^L = \{\mathbf{x} \in \mathbb{R}^n | \boldsymbol{a}^T\mathbf{x} + b \leqslant \alpha\}$

- 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

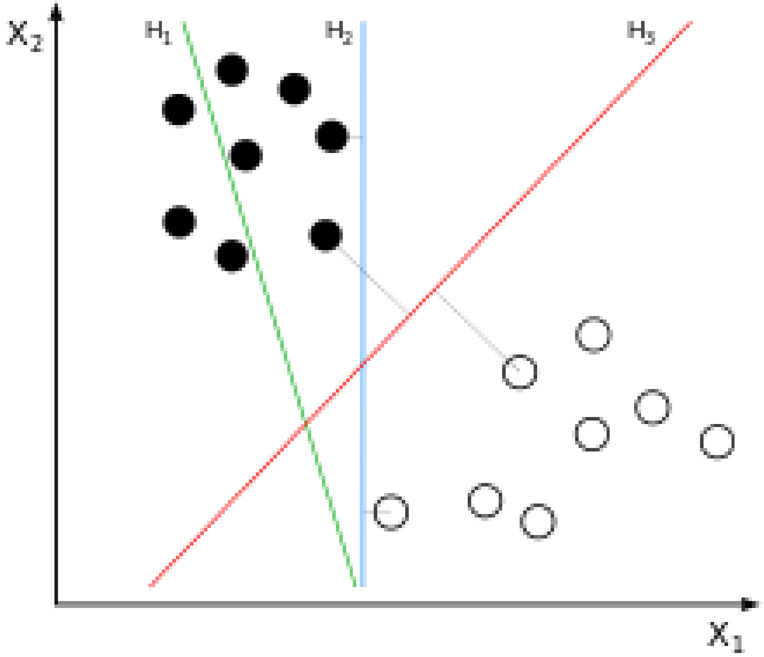
- Given $\bar{\mathbf{x}} \in \mathbb{R}^n$ and H_α, distance $(\bar{\mathbf{x}}, H_\alpha)$ = ?



- 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



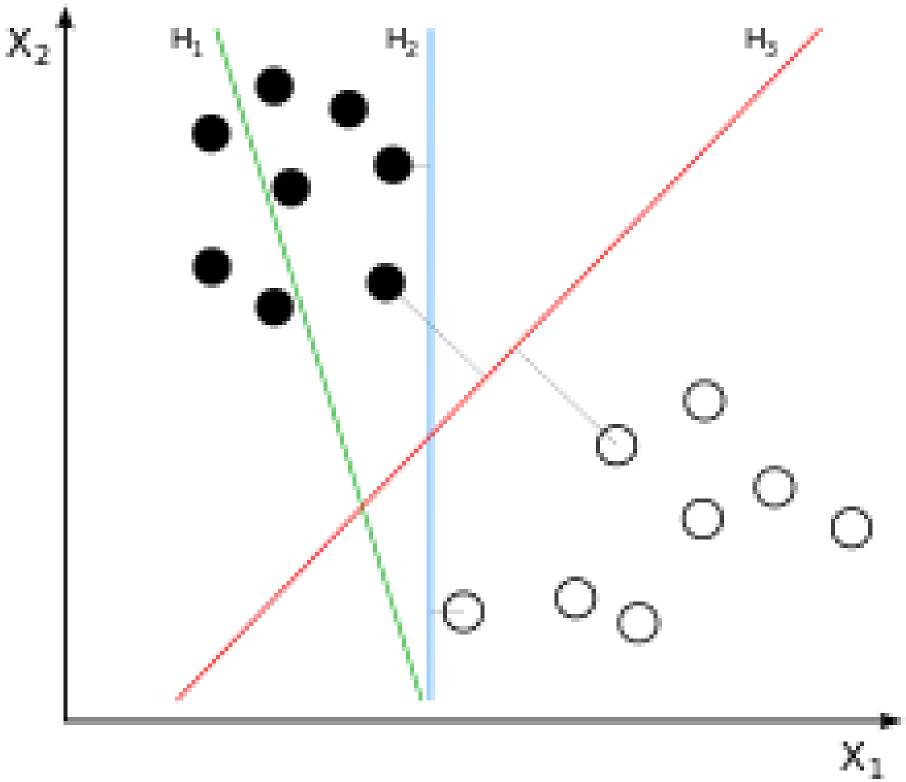
- Given a set of points $\{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$ with binary labels $y_i \in \{-1, 1\}$
- Find a hyperplane that strictly separates the two classes.

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

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

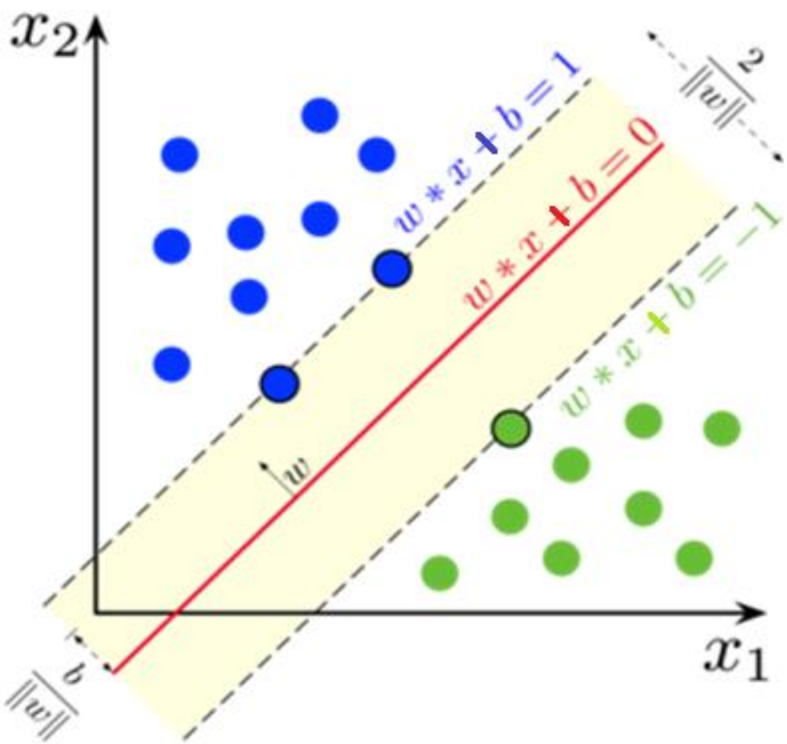
Support vector machines – basic ideas

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

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

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

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

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

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:

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

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

$\Rightarrow$ dual objective function

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

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

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

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

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

- 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

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

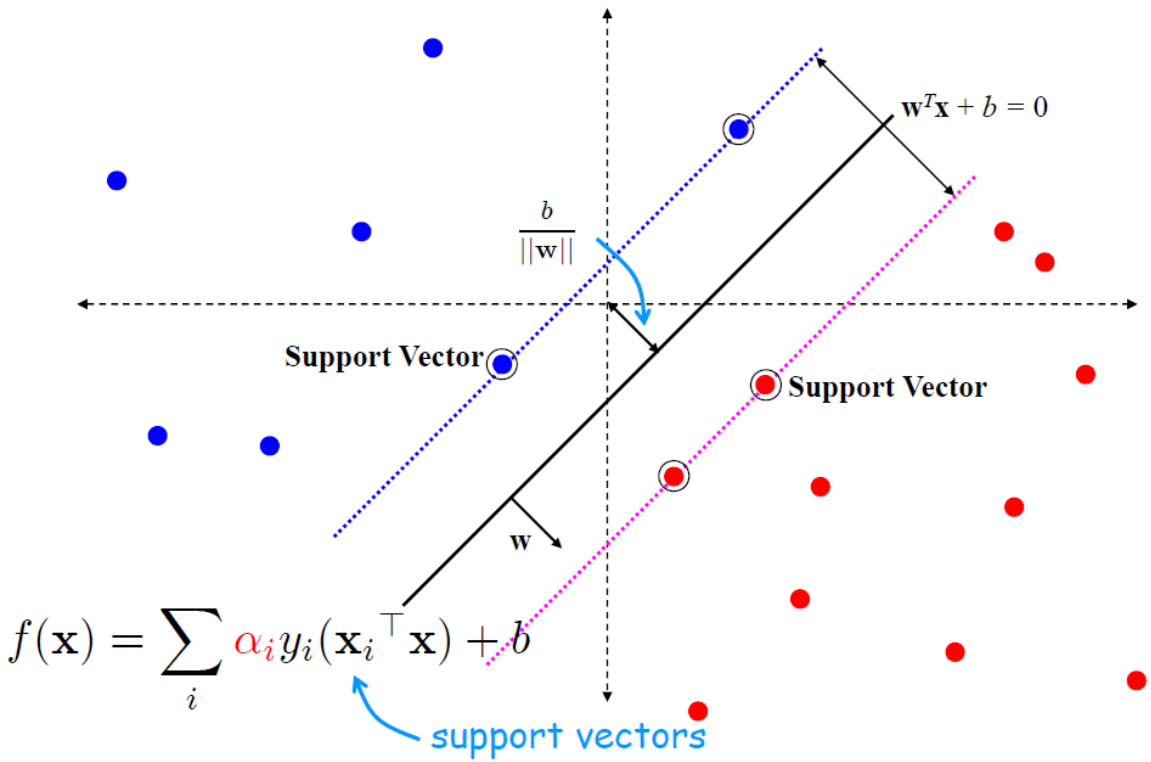
For a point $\boldsymbol{x}^i$ on the hyperplane, since $y_i^2 = 1$,

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

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

Supporting vectors

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



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 $\boldsymbol{x} \in \mathbb{R}^n$

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

where

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

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.

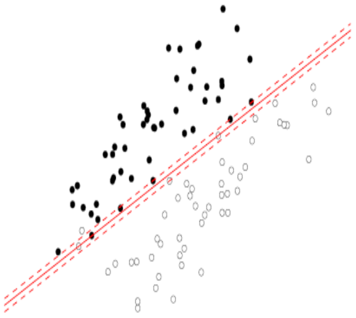
2. The set $S = \{\alpha_i \mid \alpha_i > 0\}$ is in general very sparse!

   -- easy to store and update

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 $C \to +\infty$, 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 $y_i(\boldsymbol{w}^T\boldsymbol{x}^i + b) < 1$ (a soft margin)

- Linear soft SVM

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

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

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

where $C > 0$ is a given parameter.

** 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

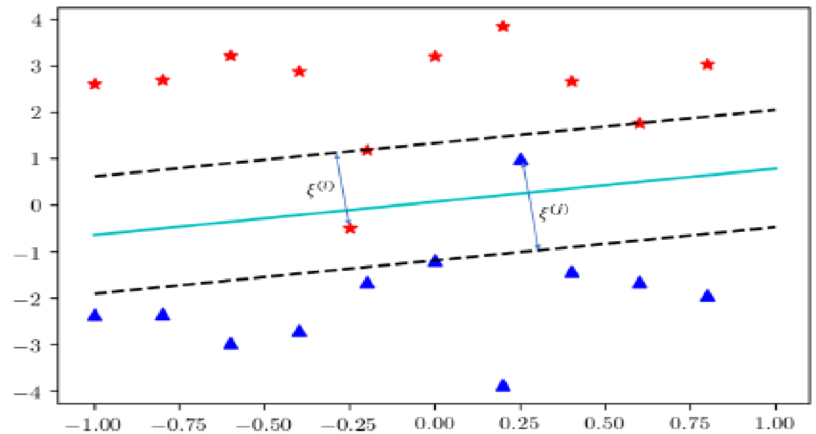
$$\min \ \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{N}\xi_i$$

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

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

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:

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

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

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

$$\Longleftrightarrow \alpha_i \leq C$$

$\Rightarrow$ 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

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

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

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

- 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)

$K = X^T X$ 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

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

     (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

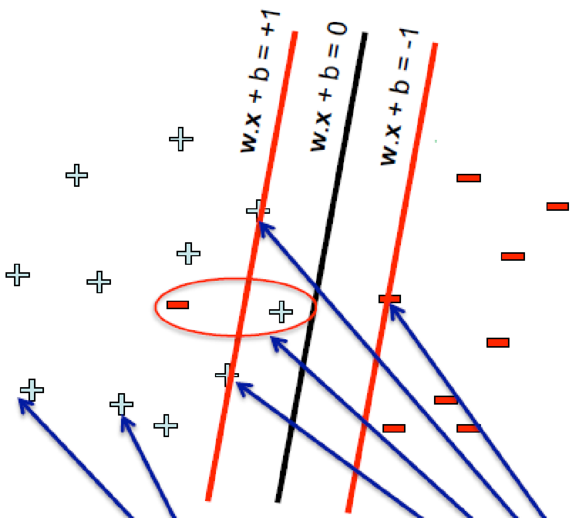
     $$\boldsymbol{w} = \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}^i$$

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

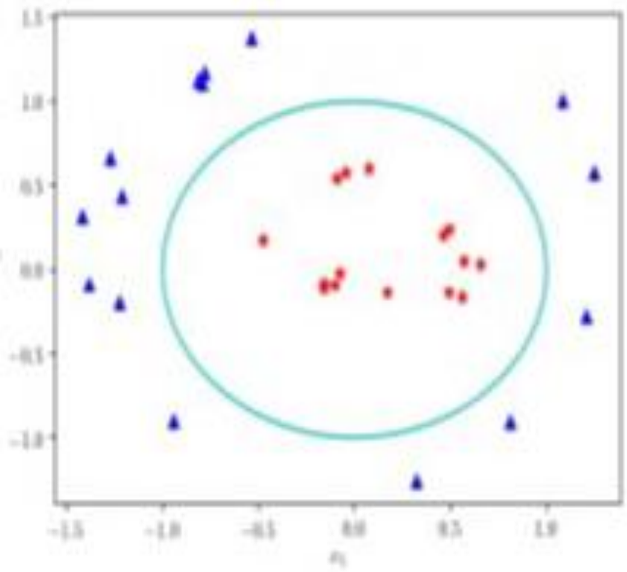
Comparisons and discussions

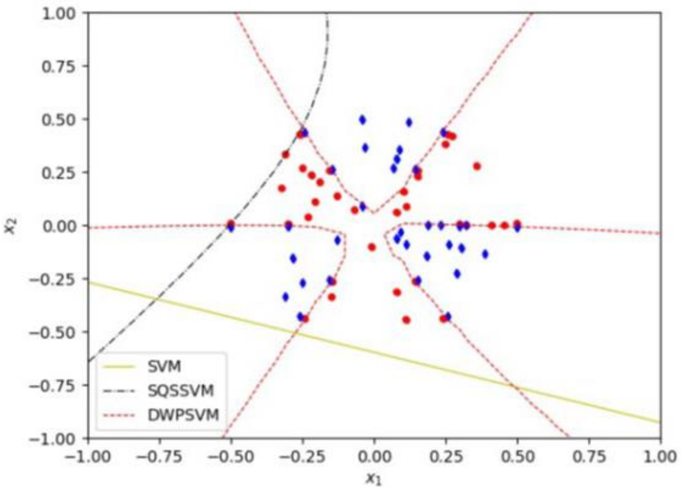
- LSVM vs. Approximate LSVM
  - applicability?
  - equivalency?
  - complexity?

- LSVM vs. LSSVM

- LSSVM vs. Approximate LSVM

SVM for not linearly separable data sets





- Will LSVM, Approximate LSVM, LSSVM work ?
- How well can they be?
- Any better SVM classifier?

SVM for not linearly separable data sets

- Basic ideas:

  1. Reformulate the problem in a higher dimensional space for linear separability
  (Kernel Method): LSVM with kernel functions

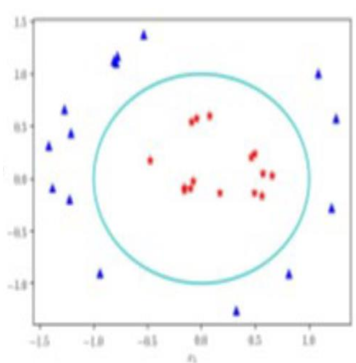
  2. Adopt nonlinear surface to separate data points apart in the original space
  - Quadratic surface SVM
  - Double-well potential function based SVM

Idea of kernel based SVM

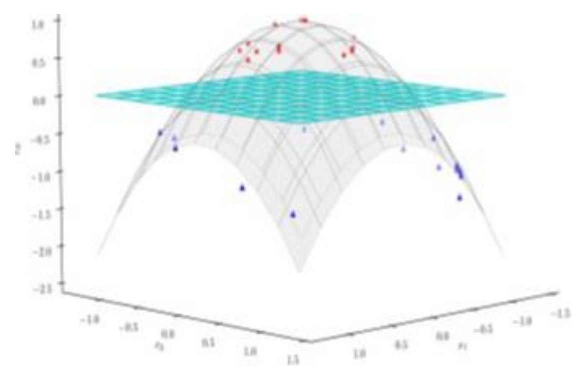
- Feature map: a function $\phi(\cdot)$: $\mathbb{R}^n \rightarrow \mathbb{R}^l$, with $l \geq n$, that maps all data points to a higher dimensional space for linear separation.

- Example 1: $\|x\|_2^2 < 1$, $\|\boldsymbol{x}\|_2^2 > 1$,

$$\phi_1(\boldsymbol{x}): \mathbb{R}^2 \rightarrow \mathbb{R}^3,\ \phi_1(\boldsymbol{x}) = \phi_1\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ 1-x_1^2-x_2^2 \end{pmatrix} = \begin{pmatrix} \boldsymbol{x} \\ 1-\|\boldsymbol{x}\|^2 \end{pmatrix}$$



$\phi \rightarrow$



Kernel-based soft SVM - KSSVM

- Using a *feature map* $\phi(\cdot):\ \mathbb{R}^n \rightarrow\ \mathbb{R}^l\ (l \geq n)$ to transform the problem to a higher dimensional space for linear separability.
- Build upon LSSVM
- Primal model

$$
\begin{aligned}
\min\quad & \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{N} \xi_i \\
s.t.\quad & y_i\left(\boldsymbol{w}^T\phi(\boldsymbol{x}^i) + b\right) \geq 1 - \xi_i, i = 1, \dots, N \qquad \text{(KSSVM)} \\
& \boldsymbol{w} \in \mathbb{R}^l,\ \ b \in \mathbb{R},\ \ \boldsymbol{\xi} \in \mathbb{R}_+^N
\end{aligned}
$$

where $C > 0$ is a given parameter.

** More variables involved than using LSSVM.

How difficult to solve DKSSVM?

- Lagrangian dual model

$$\max -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\, y_i\, K_{ij}\, y_j\alpha_j + \sum_i^N \alpha_i$$

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

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

where $K_{ij} = K(\boldsymbol{x}^i, \boldsymbol{x}^j) \triangleq \phi(\boldsymbol{x}^i)^T \phi(\boldsymbol{x}^j)$

- Given any feature map ϕ, corresponding K is psd and DKSSVM becomes a convex quadratic program with N bounded variables and only one linear equality constraint.
- In practice, we may use a kernel matrix $K = (K_{ij})$ without knowing the feature map $\phi(x)$.

Kernel-based soft SVM - DKSSVM

- SVM classifier

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

Dual version DKSSVM

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i y_i \, \phi(\boldsymbol{x}^i)^T \phi(\boldsymbol{x}) + b(\alpha_i)$$
$$= \sum_{i \in S} \alpha_i y_i \, K\left(\boldsymbol{x}^i, \boldsymbol{x}\right) + \bar{b}$$

Kernel matrix

- To make sure that $K_{ij} = K(\boldsymbol{x}^i, \boldsymbol{x}^j)$ is the inner product of $\phi(\boldsymbol{x}^i)$ and $\phi(\boldsymbol{x}^j)$ in the feature space, such that
  (1) DKSSVM is an easily solved convex QP,
  (2) there is a chance to solve KSSVM,
  we need K to be symmetric and positive semidefinite (Mercer's condition).

- Commonly used kernels:
  1. Polynomial kernel of degree $d = 1, 2, \ldots$

  $K(\boldsymbol{x}^i, \boldsymbol{x}^j) = ((\boldsymbol{x}^i)^T \boldsymbol{x}^j + r)^d$ (homogeneous, if $r = 0$)

  (inhomogeneous, if $r > 0$)

  * *popular in image processing*

Polynomial kernels

- Example 1: (inhomogeneous degree 2)

For $\boldsymbol{x} \in \mathbb{R}^1, K(x^i, x^j) = (x^i x^j + 1)^2$ for $r = 1, d = 2,$

we have $\phi(x)^T = (1, \sqrt{2}x, x^2) \in \mathbb{R}^3$ such that

$$\phi(x^i)^T \phi(x^j) = 1 + 2x^i x^j + (x^i)^2 (x^j)^2 = (x^i x^j + 1)^2$$

Example 2: (homogeneous degree 2)

For $\boldsymbol{x} \in \mathbb{R}^2, K(\boldsymbol{x}^i, \boldsymbol{x}^j) = ((\boldsymbol{x}^i)^T \boldsymbol{x}^j)^2$ for $r = 0, d = 2,$

we have $\phi(\boldsymbol{x})^T = (x_1^2, \sqrt{2}x_1 x_2, x_2^2) \in \mathbb{R}^3$ such that

$$\phi(\boldsymbol{x}^i)^T \phi(\boldsymbol{x}^j) = (x_1^i)^2 (x_1^j)^2 + (x_2^i)^2 (x_2^j)^2 + 2(x_1^i x_2^i x_1^j x_2^j) = ((\boldsymbol{x}^i)^T \boldsymbol{x}^j)^2$$

**General form $\phi(x)$: contains all polynomial terms up to degree d.

Kernel matrix

Commonly used kernels:

2. Gaussian kernel with $\sigma \in \mathbb{R}\backslash\{0\}$

$$K(\boldsymbol{x}^i, \boldsymbol{x}^j) = exp\left(-\frac{\left\|\boldsymbol{x}^i - \boldsymbol{x}^j\right\|_2^2}{2\sigma^2}\right)$$

* no prior information, general purpose

***General form $\phi(x)$ in infinite dimensional feature space.*

3. Gaussian Radial basis function (RBF) kernel with $\gamma > 0$

$$K(\boldsymbol{x}^i, \boldsymbol{x}^j) = exp\left(-\gamma\left\|\boldsymbol{x}^i - \boldsymbol{x}^j\right\|_2^2\right)$$

* no prior information, general purpose

***General form $\phi(x)$* : see https://en.wikipedia.org/wiki/Radial_basis_function_kernel

Kernel matrix

Commonly used kernels:

4. Laplace RBF kernel with $\sigma > 0$

$K(\boldsymbol{x}^i, \boldsymbol{x}^j) = exp\ (\ -1/\sigma\ \|\boldsymbol{x}^i - \boldsymbol{x}^j\|_2\ )$

* no prior information, general purpose

5. Sigmoid kernel with $\beta > 0\ ,\ \theta \in \mathbb{R}$

$K(\boldsymbol{x}^i, \boldsymbol{x}^j) = tanh\left(\beta\left(\boldsymbol{x}^i\right)^T \boldsymbol{x}^j + \theta\right)$

* proxy for neural networks

Quality of kernel-based SVM

- Two major factors:

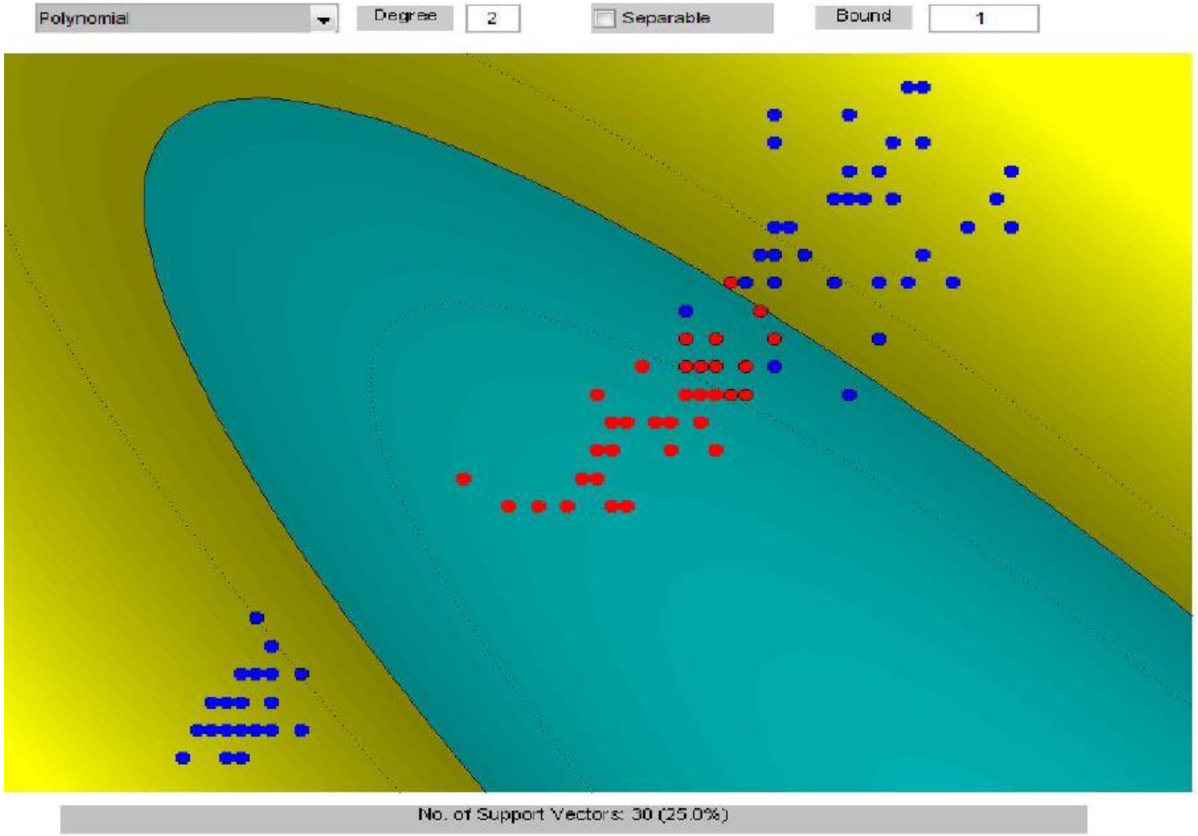
1. Like LSSVM, the parameter C plays a role.

2. The choice of an appropriate kernel matrix (and its parameters) is important.

Effect of kernel matrix

- Picture from Machine Learning 10-315, Aarti Singh, Oct 28, 2020,CMU

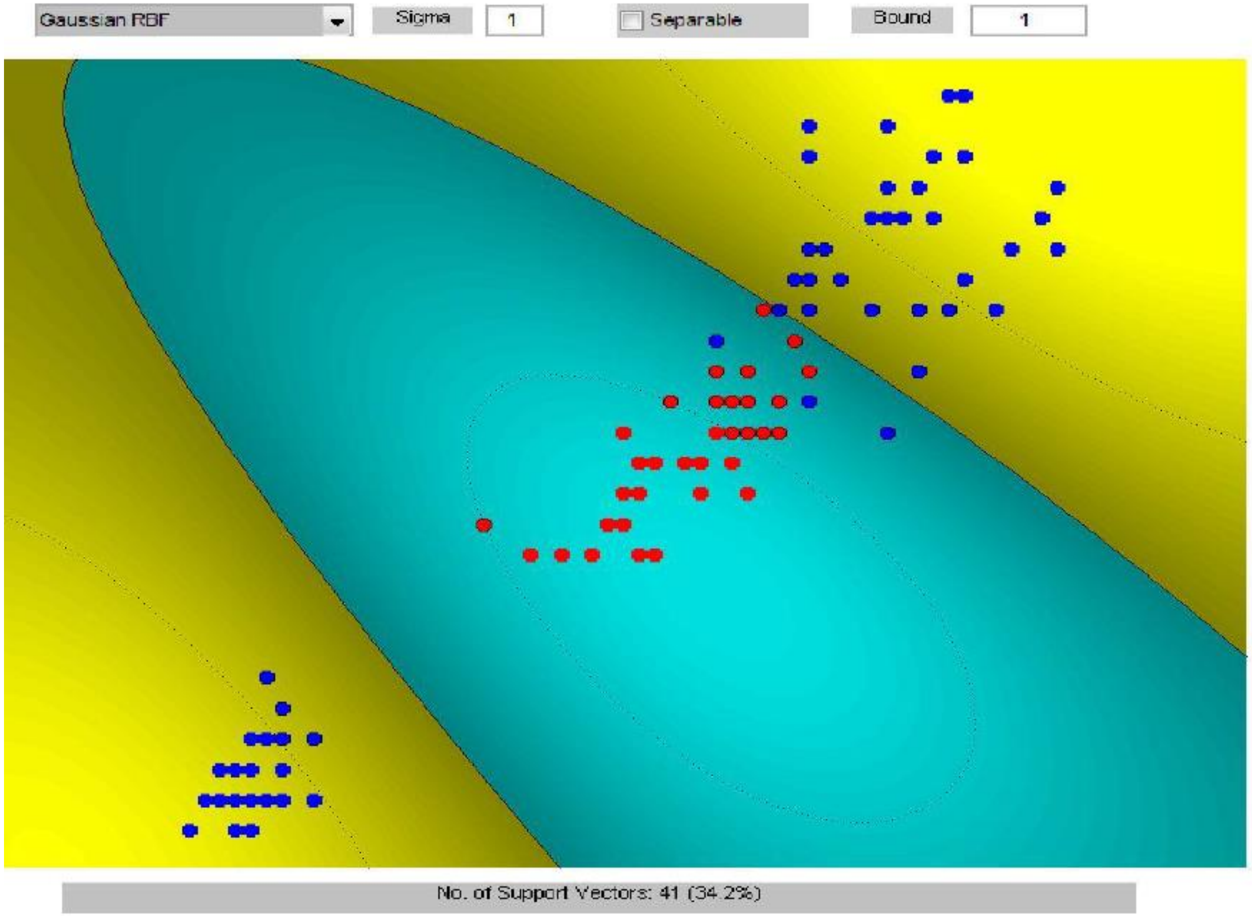
  Iris dataset, 1 vs 23, Polynomial Kernel degree 2 ($C = 1$)



Effect of kernel matrix

- Picture from Machine Learning 10-315, Aarti Singh, Oct 28, 2020,CMU

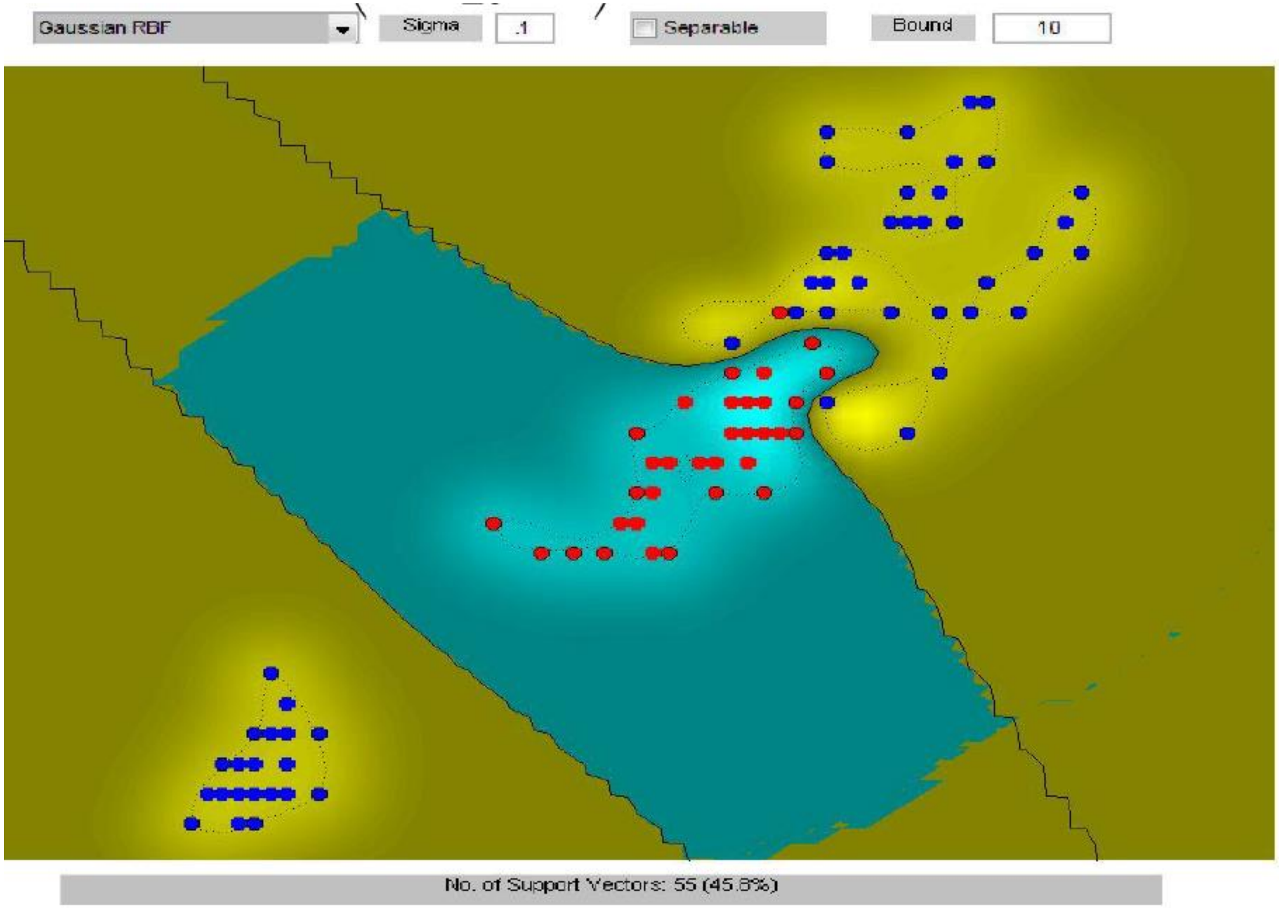
Iris dataset, 1 vs 23, Gaussian RBF kernel $(C = 1, \sigma = 1)$



Effect of kernel matrix

- Picture from Machine Learning 10-315, Aarti Singh, Oct 28, 2020,CMU

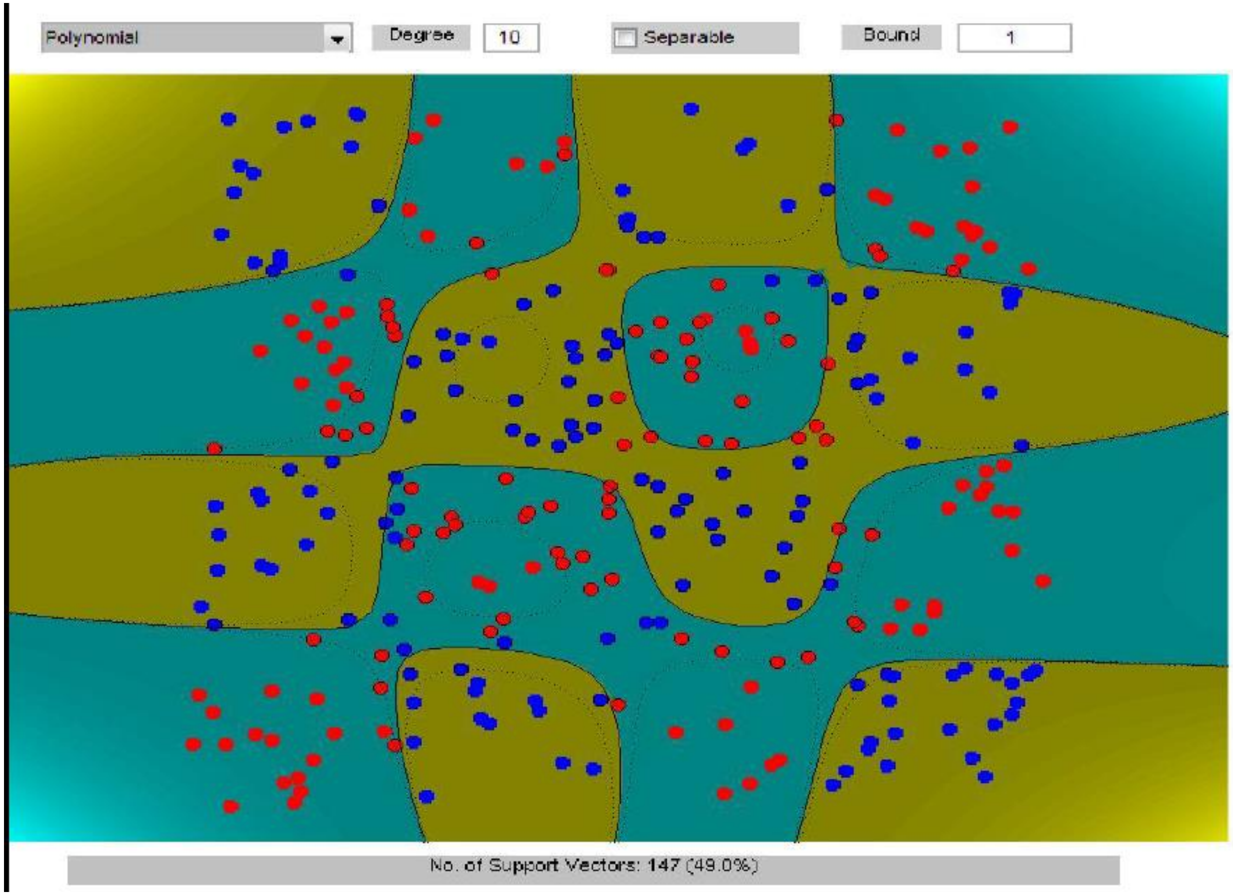
Iris dataset, 1 vs 23, Gaussian RBF kernel ($C = 10, \sigma = 1$)



Effect of kernel matrix

- Picture from Machine Learning 10-315, Aarti Singh, Oct 28, 2020,CMU

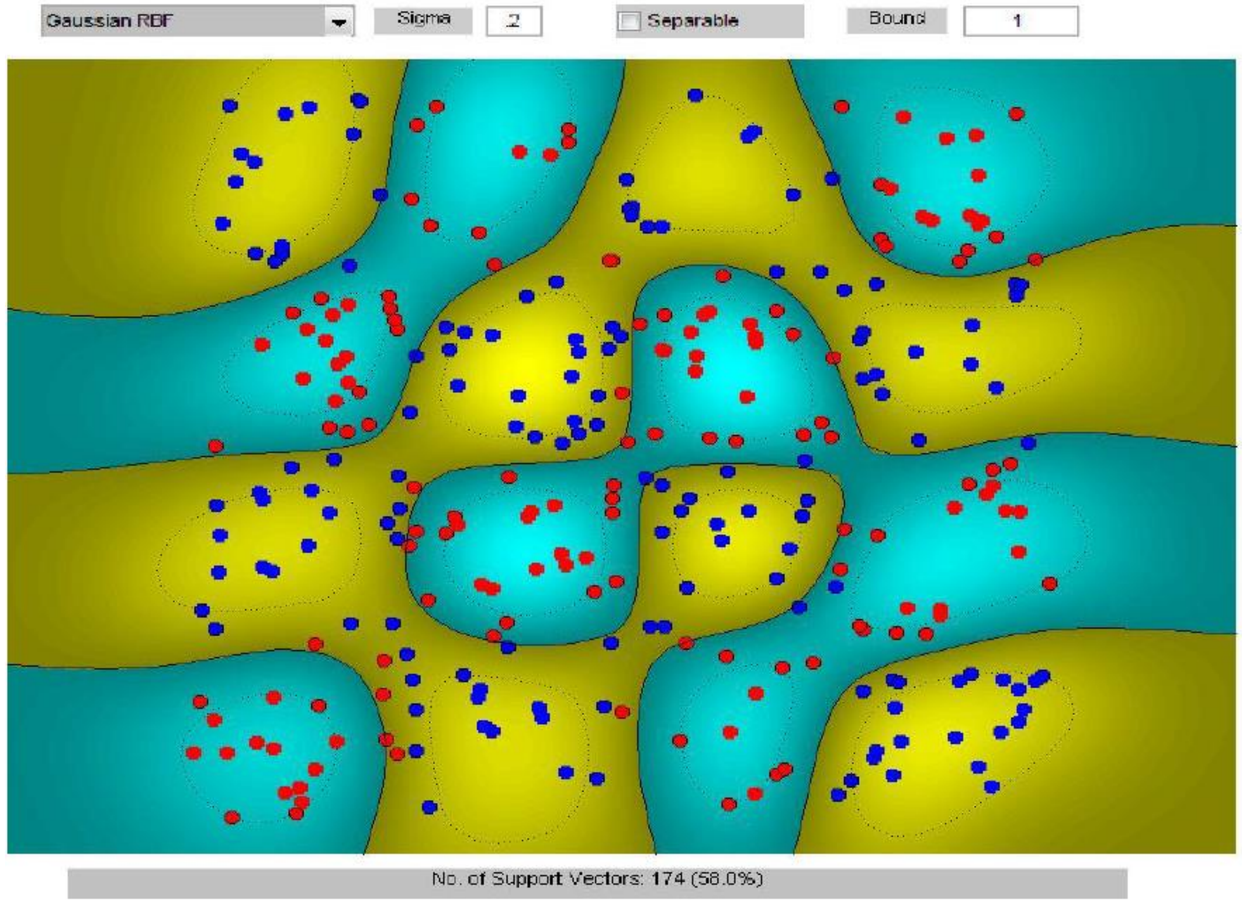
  Chessboard dataset, Polynomial kernel ($d = 10, C = 1$)



Effect of kernel matrix

- Picture from Machine Learning 10-315, Aarti Singh, Oct 28, 2020,CMU

  Chessboard dataset, Gaussian RBF kernel $(C = 1, \sigma = 2)$



Quality of kernel-based SVM

- Two major factors:

1. Like LSSVM, the parameter C plays a role.

2. The choice of an appropriate kernel matrix
 (and its parameters) is important.

Question: How to choose/design right ones?
- theoretical analysis?
- computational experiments !

Ideas of choosing parameters

- Example: choosing parameter C

1. Define an error or score measure:

   for example, MSE (mean squares error),

   MAPE (mean absolute percentage error),

   $1/\|\boldsymbol{w}\|_2^2$, or $\sum_{i=1}^{N} y_i\left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right)$, …

2. Conduct computational experiments with different value of C :

   - statistically meaningful

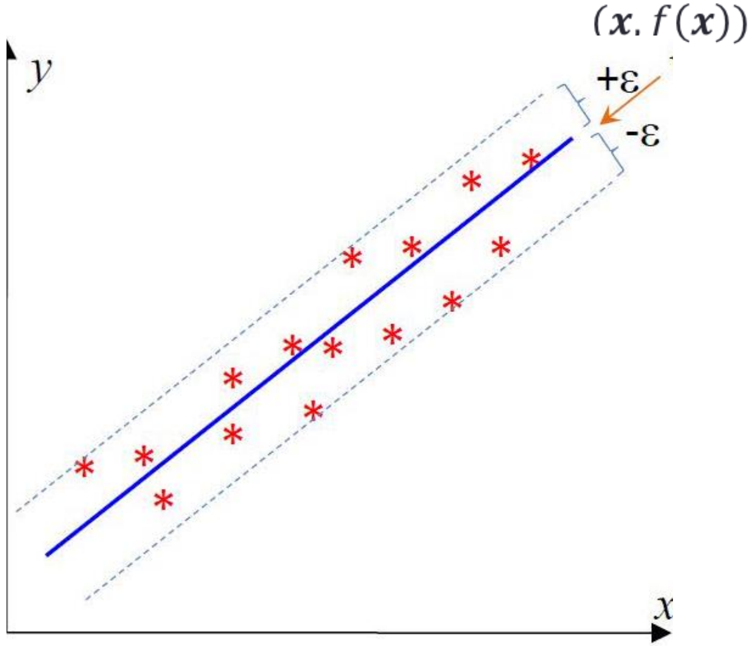
3. Plot resulting error measures against C.
4. Find the elbow/ turning point value of C.

** check many other “cross-validation” methods.

Linear support vector regression

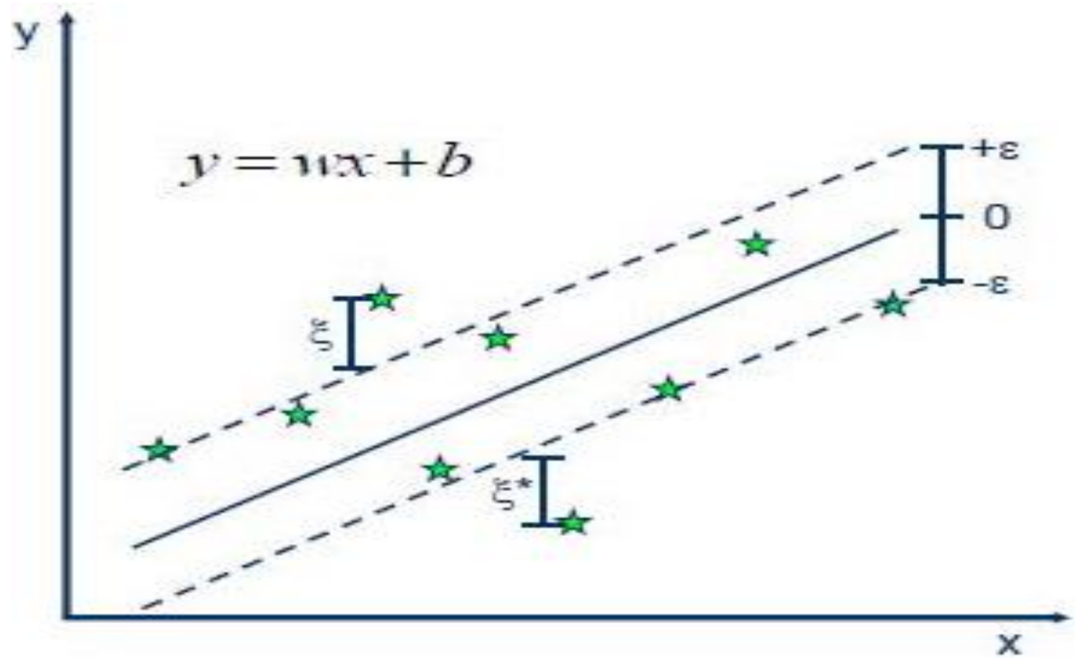
- Problem settings:
  - Dataset $\{\,(\boldsymbol{x}^i, y_i) \in \mathbb{R}^n \times \mathbb{R} \mid i = 1,2.,,,,N\}$ of N data points
  - tube tolerance $\varepsilon > 0$

- Aim: to find affine map $f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$ with wide margin such that $|y_i - f(\boldsymbol{x}^i)| < \varepsilon,\, i = 1, \ldots, N$



Observation

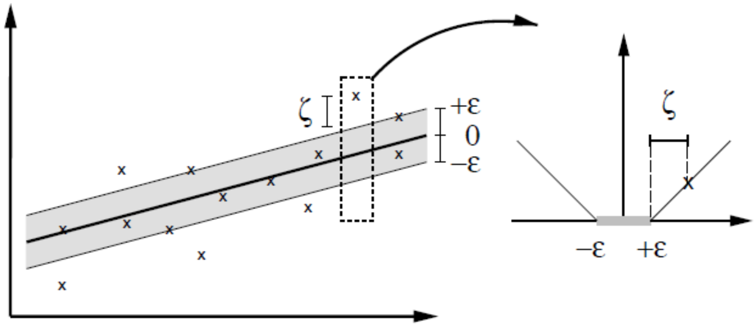
- Question: How big the box tolerance ε should be?
  - When $\varepsilon\ (> 0)$ is too small, we may not be able to box all data-points in the tube.



Linear soft support vector regression

- Primal model: (For a given $C > 0$)

$$
\begin{aligned}
Min \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{N}\xi_i \\
\text{s.t.} \quad & y_i - \boldsymbol{w}^T\boldsymbol{x}^i - b \le \varepsilon + \xi_i,\ i = 1,\dots,N \quad \text{(LSSVR)} \\
& y_i - \boldsymbol{w}^T\boldsymbol{x}^i - b \ge -\varepsilon - \xi_i,\ i = 1,\dots,N \\
& \boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R}, \boldsymbol{\xi} \in \mathbb{R}_+^N
\end{aligned}
$$



soft margin with $\varepsilon - insensitive$ loss function

Linear soft SVR - LSSVR

1. (LSSVR) is a convex quadratic program with $n + 1$ free variables, N non-negative variables, and $2N$ linear inequality constraints.

2. (LSSVR) is always feasible.

3. Who are supporting vectors?

4. Any dual information?

Dual LSSVR - DLSSVR

- Lagragian

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

- KKT conditions

  \- Primal & dual feasibility

  (i) $\alpha_i, \alpha_i^*, \eta_i \geq 0, i = 1, \dots, N;$

  (ii) $\varepsilon + \xi_i - y_i + \boldsymbol{w}^T\boldsymbol{x}^i + b \geq 0; \varepsilon + \xi_i + y_i - \boldsymbol{w}^T\boldsymbol{x}^i - b \geq 0;$

Dual LSSVR - DLSSVR

- Lagragian

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

- KKT conditions

  Stationarity

  (iii) $\nabla_{\boldsymbol{w}} L = \boldsymbol{w} - \sum_{i=1}^N (\alpha_i - \alpha_i^*)\boldsymbol{x}^i = 0;$

  (iv) $\nabla_b L = \sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0;$

  (v) $\nabla_{\xi_i} L = C - \eta_i - (\alpha_i + \alpha_i^*) = 0;$

  $\Longrightarrow \eta_i = C - (\alpha_i + \alpha_i^*) \geq 0$ and $0 \leq \alpha_i + \alpha_i^* \leq C$

Dual soft support vector regression -DLSSVR

- Dual model:

$$\begin{aligned} Max \quad & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*) < \boldsymbol{x}^i, \boldsymbol{x}^j > (\alpha_j - \alpha_j^*) \\ & -\varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i\,(\alpha_i - \alpha_i^*) \\ \text{s.t.} \quad & \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \qquad \text{(DLSSVR)} \\ & 0 \le \alpha_i + \alpha_i^* \le C, \alpha_i \ge 0, \alpha_i^* \ge 0, i = 1, \dots, N \end{aligned}$$

*Depending on $y_i > \boldsymbol{w}^T\boldsymbol{x}^i + b$, or $y_i < \boldsymbol{w}^T\boldsymbol{x}^i + b$, at least one of α_i or α_i^* = 0. So we have

$$\begin{aligned} Max \quad & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*) < \boldsymbol{x}^i, \boldsymbol{x}^j > (\alpha_j - \alpha_j^*) \\ & -\varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i\,(\alpha_i - \alpha_i^*) \\ \text{s.t.} \quad & \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \qquad \text{(DLSSVR)} \\ & 0 \le \alpha_i \le C,\; 0 \le \alpha_i^* \le C, i = 1, \dots, N \end{aligned}$$

Dual soft support vector regression -DLSSVR

- Observations:

1. (DLSSVR) is a convex quadratic program with $2N$ bounded variables and 1 linear equality constraint.

2. (DLSSVR) is independent of the size of n, which is absolved in the inner product of $(\boldsymbol{x}^i)^T\boldsymbol{x}^j = <\boldsymbol{x}^i, \boldsymbol{x}^j>$.

DLSSVR

- Dual-to-primal conversion:
- KKT (iii) say that

$$\nabla_{\boldsymbol{w}} L = \boldsymbol{w} - \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) \boldsymbol{x}^i = 0.$$

Hence,

$$\boldsymbol{w} = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) \boldsymbol{x}^i \quad \text{and}$$

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) < \boldsymbol{x}^i, \boldsymbol{x} > + b$$

* This is called a "support vector expansion" of $f(\boldsymbol{x})$.

* What is b ?

DLSSVR

- KKT conditions: Complementary slackness:

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

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

  (viii) $\eta_i\xi_i = (C - (\alpha_i + \alpha_i^*))\ \xi_i = 0$

Observations:

1. Depend on $y_i > \boldsymbol{w}^T\boldsymbol{x}^i + b$, or $y_i < \boldsymbol{w}^T\boldsymbol{x}^i + b$,

   at least one of α_i or $\alpha_i^* = 0$.

2. When data-point $\left(\boldsymbol{x}^i, y_i\right)$ is in the tube

   $$\left|y_i - \left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right)\right| < \varepsilon \Rightarrow \alpha_i = 0 \text{ and } \alpha_i^* = 0.$$

DLSSVR

- KKT conditions: Complementary slackness:

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

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

  (viii) $\eta_i\xi_i = (C - (\alpha_i + \alpha_i^*))\,\xi_i = 0$

Observations:

3. When data-point $\left(\boldsymbol{x}^i, y_i\right)$ is outside of the tube,

$$\left|y_i - \left(\boldsymbol{w}^T\boldsymbol{x}^i + b\right)\right| > \varepsilon \Rightarrow \xi_i > 0 \Rightarrow \alpha_i = C \text{ or } \alpha_i^* = C.$$

4. $\alpha_i \in (0, C)$ or $\alpha_i^* \in (0, C)$ happens only when $\left(\boldsymbol{x}^i, y_i\right)$ lies on the tube

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

$\Rightarrow$ either $y_i - (\boldsymbol{w}^T\boldsymbol{x}^i + b) = \varepsilon \Rightarrow b = \varepsilon - y_i + \boldsymbol{w}^T\boldsymbol{x}^i$, when $\alpha_i \in (0, C)$

or $y_i - (\boldsymbol{w}^T\boldsymbol{x}^i + b) = -\varepsilon \Rightarrow b = -\varepsilon - y_i + \boldsymbol{w}^T\boldsymbol{x}^i$, when $\alpha_i^* \in (0, C)$

5. Supporting vectors are indeed sparse!

DLSSVR

- Dual-to-primal conversion:
- KKT (iii) say that

$$\nabla_{\boldsymbol{w}} L = \boldsymbol{w} - \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\boldsymbol{x}^i = 0.$$

Hence,

$$\boldsymbol{w} = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\boldsymbol{x}^i$$

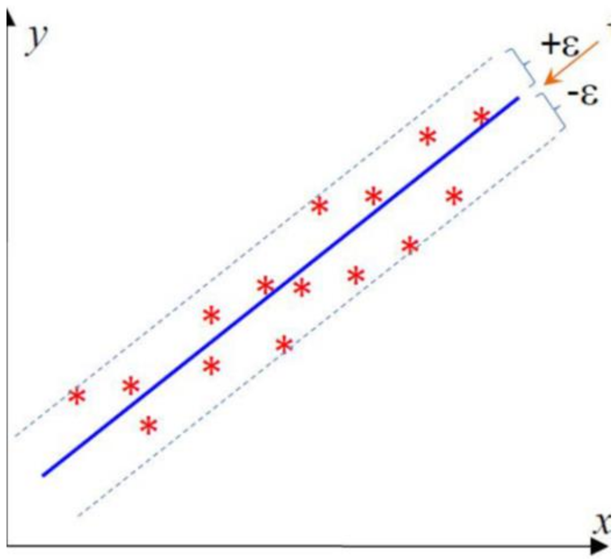
$$b = \begin{pmatrix} \varepsilon - y_i + \boldsymbol{w}^T\boldsymbol{x}^i, & \text{if } \alpha_i \in (0, C) \\ -\varepsilon - y_i + \boldsymbol{w}^T\boldsymbol{x}^i, & \text{if } \alpha_i^* \in (0, C) \end{pmatrix}$$

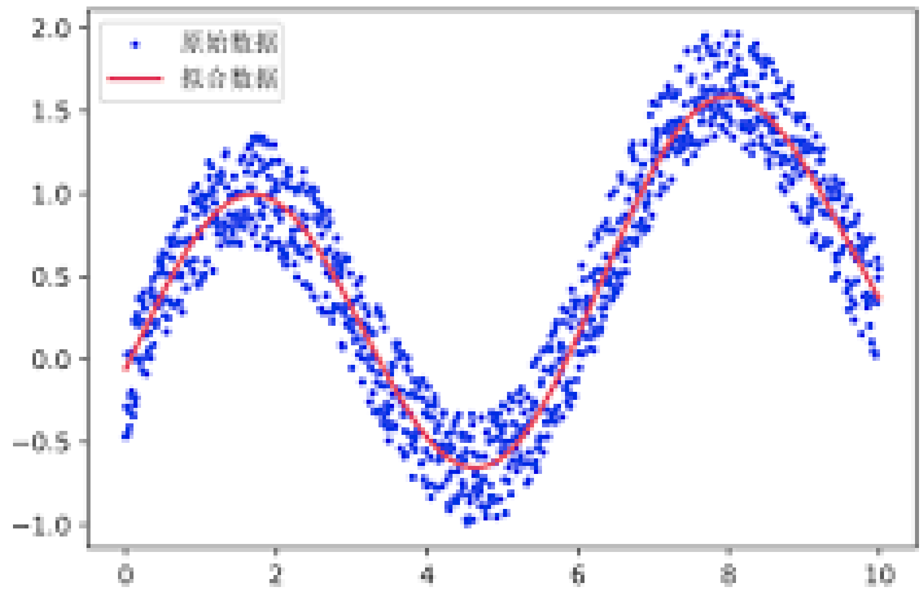
and DLSSVR prediction is

$$f(\boldsymbol{x}) = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) < \boldsymbol{x}^i, \boldsymbol{x} > + b$$

SVM-based nonlinear regression

- From linear to nonlinear regression





Kernel-based linear soft SVR

- Use a *feature map* $\phi(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^l \ (l \geq n)$ to transform the problem to a higher dimensional space for linear separability.

- Primal model: (For a given $C > 0$)

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

$$\text{s.t.} \quad y_i - \boldsymbol{w}^T\phi(\boldsymbol{x}^i) - b \leq \varepsilon + \xi_i, \ i = 1, \dots, N \ \text{(KLSSVR)}$$

$$y_i - \boldsymbol{w}^T\phi(\boldsymbol{x}^i) - b \geq -\varepsilon - \xi_i, \ i = 1, \dots, N$$

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

* Dimensionality changes from n to l.

Dual kernel-based linear soft support vector regression

- Dual model:

$$
\begin{aligned}
Max \quad & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*) < \phi(\boldsymbol{x}^i), \phi(\boldsymbol{x}^j) > (\alpha_j - \alpha_j^*) \\
& -\varepsilon \sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i (\alpha_i - \alpha_i^*) \\
\text{s.t.} \quad & \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \qquad \text{(DKLSSVR)} \\
& 0 \le \alpha_i \le C, 0 \le \alpha_i^* \le C, i = 1, \dots, N
\end{aligned}
$$

*(DKLSSVR) is a convex quadratic program with $2N$ bounded variables and 1 linear equality constraint.

*(DKLSSVR) is independent of the size of n, which is absolved in the inner product of $\phi(\boldsymbol{x}^i)^T\phi(\boldsymbol{x}^j) =< \phi(\boldsymbol{x}^i), \phi(\boldsymbol{x}^j) >$.

Kernel-based linear soft SVR

- Knowing an admissible kernel (Mercer's condition)

  $K = (k(x, x'))$ with $k(x, x') = \phi(x)^T \phi(x')$ rather than the feature mapping $\phi(x)$ explicitly, we have a kernel-based LSSVR for nonlinear regression:

$$
\begin{aligned}
Max \quad & -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\alpha_i - \alpha_i^*) k(\boldsymbol{x}^i, \boldsymbol{x}^j)\, (\alpha_j - \alpha_j^*) \\
& -\varepsilon \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i\, (\alpha_i - \alpha_i^*) \\
\text{s.t.} \quad & \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0 \qquad \text{(DKLSSVR)} \\
& 0 \le \alpha_i \le C, 0 \le \alpha_i^* \le C, i = 1, \dots, N
\end{aligned}
$$

DLSSVR vs. DKLSSVR

- Same structure, same complexity:

$$
\begin{aligned}
Max \quad & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*)k(\boldsymbol{x}^i, \boldsymbol{x}^j)\,(\alpha_j - \alpha_j^*) \\
& -\varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i\,(\alpha_i - \alpha_i^*) \\
\text{s.t.} \quad & \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \qquad\qquad \text{(DKLSSVR)} \\
& 0 \le \alpha_i \le C, 0 \le \alpha_i^* \le C, i = 1, \dots, N
\end{aligned}
$$

$$
\begin{aligned}
Max \quad & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*) < \boldsymbol{x}^i, \boldsymbol{x}^j > (\alpha_j - \alpha_j^*) \\
& -\varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i\,(\alpha_i - \alpha_i^*) \\
\text{s.t.} \quad & \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \qquad\qquad \text{(DLSSVR)} \\
& 0 \le \alpha_i \le C, 0 \le \alpha_i^* \le C, i = 1, \dots, N
\end{aligned}
$$

Support vector expansion of KLSSVR

- For KLSSVR

$$\boldsymbol{w} = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\phi(\boldsymbol{x}^i)$$

$$b = \begin{cases} \varepsilon - y_i + \boldsymbol{w}^T\boldsymbol{x}^i, \text{ if } \alpha_i \in (0, C) \\ -\varepsilon - y_i + \boldsymbol{w}^T\boldsymbol{x}^i, \text{if } \alpha_i^* \in (0, C) \end{cases}$$

KLSSVR Prediction:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\phi(\boldsymbol{x}^i)^{\boldsymbol{T}}\phi(\boldsymbol{x}) + b$$

or

$$f(\boldsymbol{x}) = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)k(\boldsymbol{x}^i, \boldsymbol{x}) + b$$