自然语言处理

杨海钦

2025-2026-1学期

基于MMDS等课件更新

面试问题

1. 支持向量机(Support Vector Machine, SVM)的核心目标的是什么？请通俗解释 “最大间隔” 和 “支持向量” 的核心意义。
2. 核函数在 SVM 中的核心作用是什么？本质是解决什么问题？
3. 硬间隔 SVM 和软间隔 SVM 的核心区别是什么？软间隔引入的原因和作用是什么？

大纲

- 复习
- 支持向量机
  - 线性可分
  - 线性不可分
  - 非线性
    - 核函数
  - 如何求解参数
  - 多分类

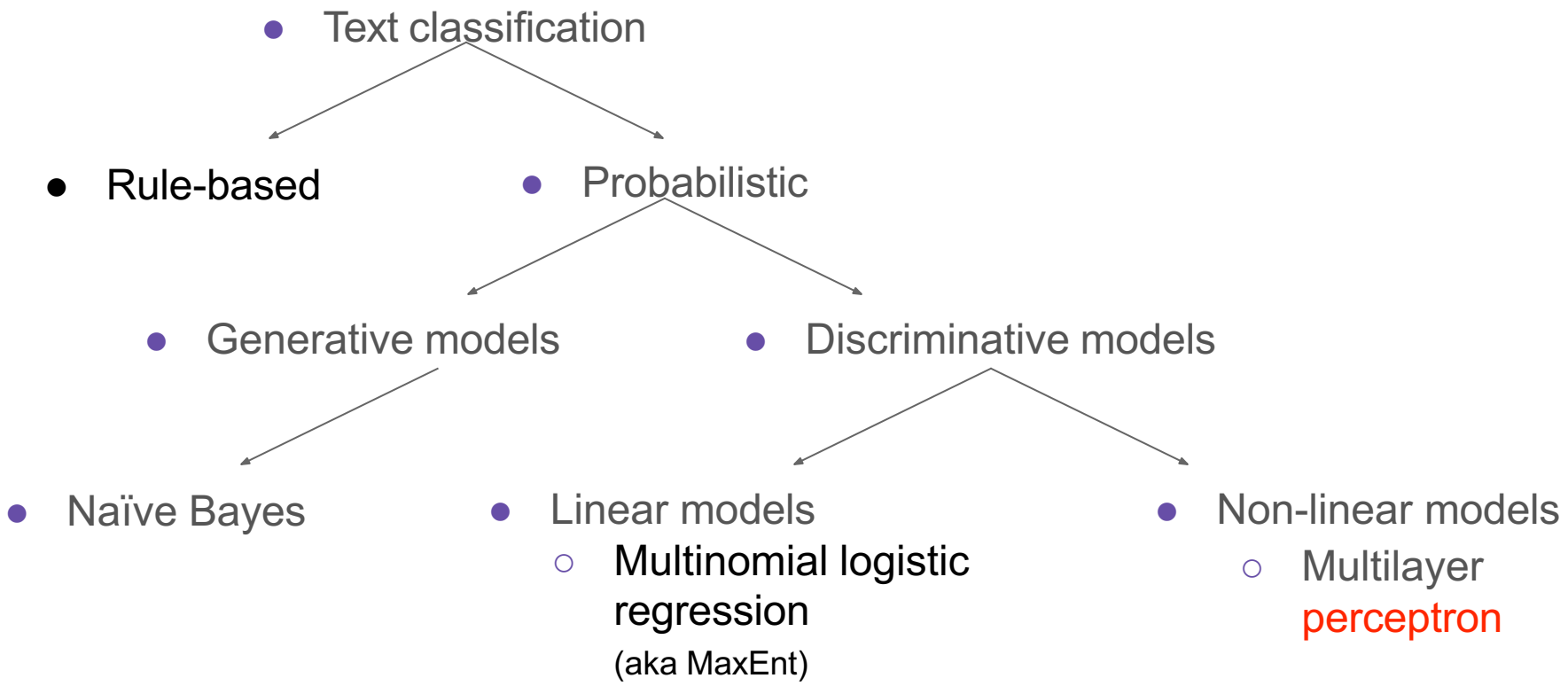
后续课程，我们将探讨

1. 如何将文本“吸收/digest”成函数可用的形式*F*?
   (关键词：特征、特征提取、特征选择、表征)
   feature, extraction/selection, representation

2. 我们可以用什么样的策略来创建决策函数*f*?
   (关键词：建模 models)

3. 如何评估决策函数*s*?
   (关键词：评估 evaluation)

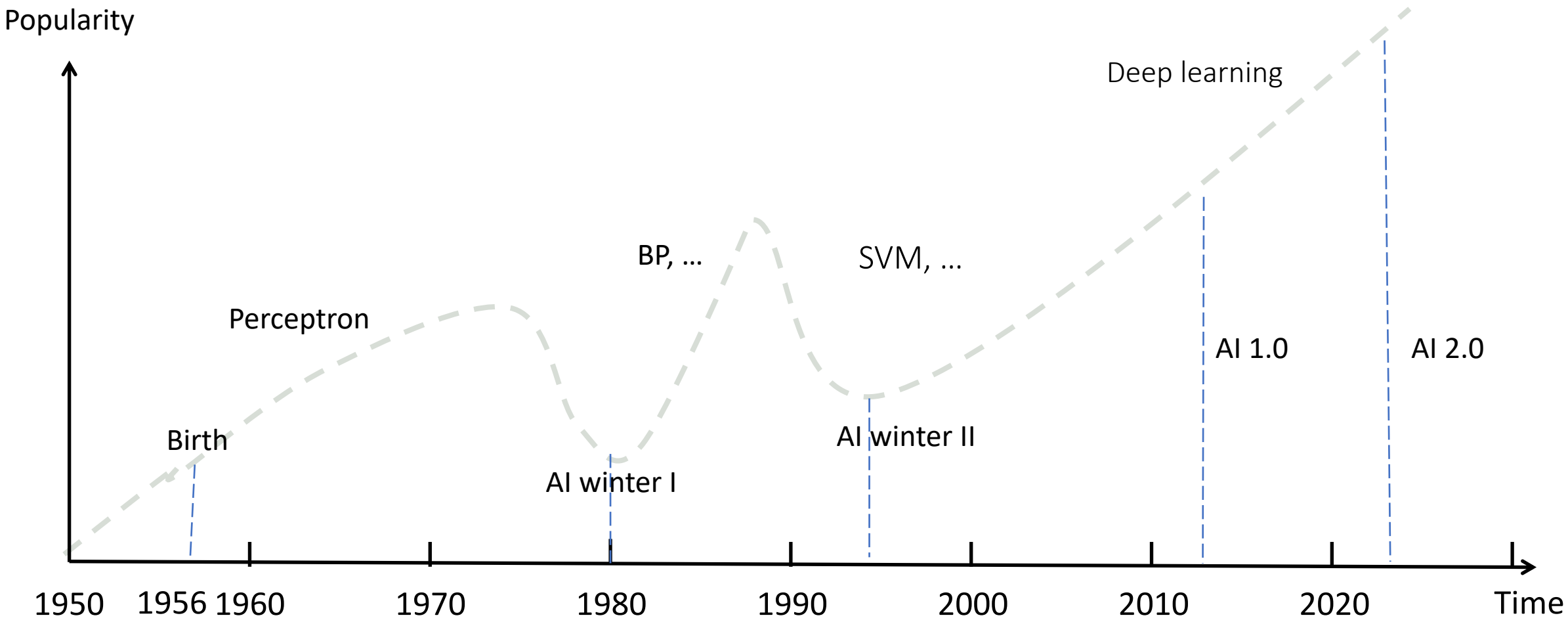
判定式线性模型: 感知器/Perceptron



概率机器学习分类器的组成部分

- 给定N个样本(输入/输出对): $\left(x^{(i)}, y^{(i)}\right)$
  1. 计算输入的特征表示: 对于每个输入观测值$x^{(i)}$ ，获得其对应的n维特征表示$\left[x_1^{(i)}, x_2^{(i)}, \cdots, x_n^{(i)}\right]$, 通常其特征$j$亦可记为$F_j(x)$
  2. 通过分类函数计算其预测值$\hat{y} = P(y|x)$, 分类函数类似于sigmoid或softmax函数
  3. 学习的目标函数，比如交叉熵损失
  4. 优化目标函数的算法：如随机梯度下降

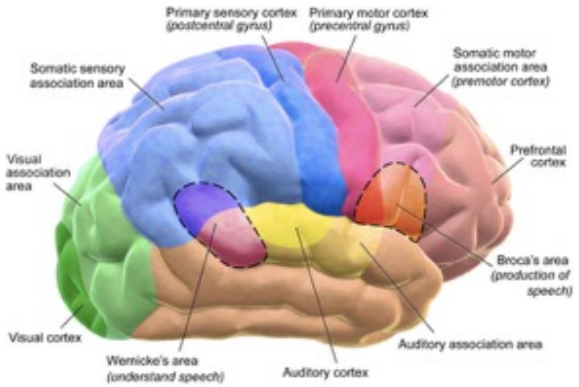
历史

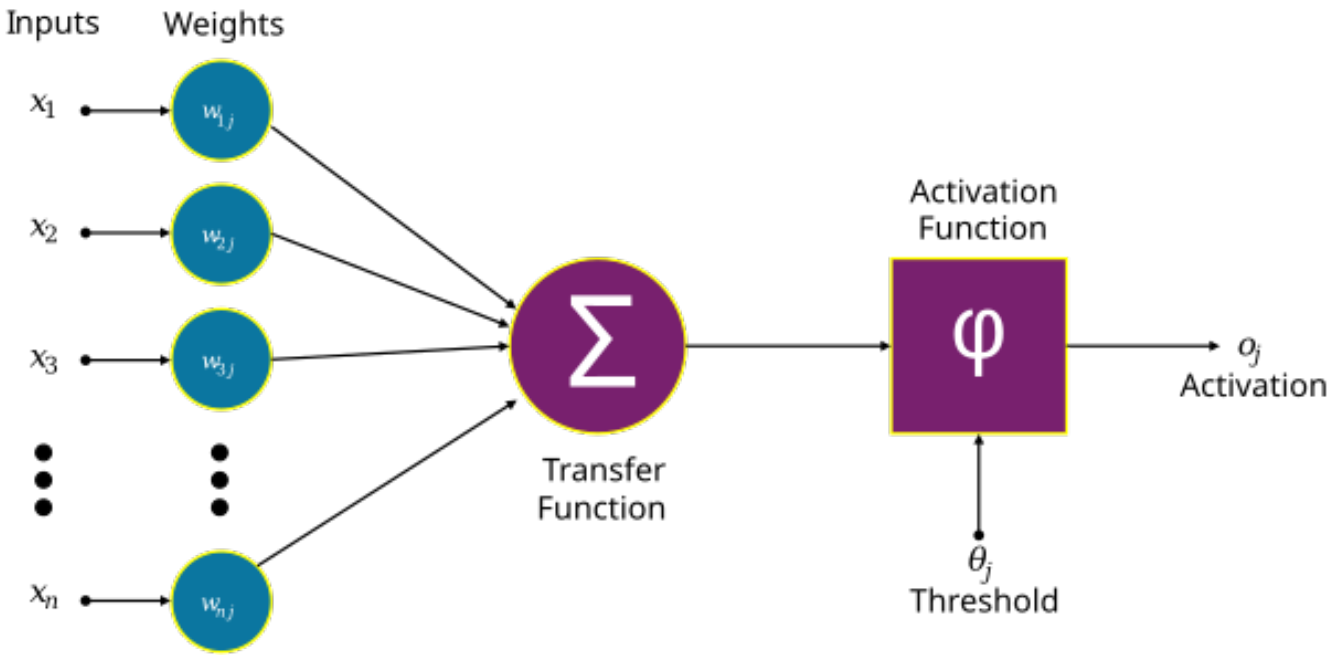

Popularity
Deep learning
BP, …
SVM, …
Perceptron
AI 1.0
AI 2.0
Birth
AI winter I
AI winter II
1950
1956
1960
1970
1980
1990
2000
2010
2020
Time

M-P神经元Neuron

[McCulloch–Pitts, 1943]

- 神经网络的基础





$$o_j = \varphi(\sum_{i=1}^{n} w_i x_i - \theta_j)$$

感知器算法/Perceptron

[F. Rosenblatt, 1958]

- 目标：找到一个误差小的线性分类器

1: Initialize $\mathbf{w}_0 = \mathbf{0}$
2: **for** $t = 1, 2, \ldots$ **do**
3: Observe $\mathbf{x}_t$ and predict $\text{sign}(\mathbf{w}_{t-1}^T \mathbf{x}_t)$
4: Update
- If $\mathbf{w}_{t-1}^T \mathbf{x}_t y_t \leq 0$, then $\mathbf{w}_t = \mathbf{w}_{t-1} + \mathbf{x}_t y_t$
- Otherwise $\mathbf{w}_t = \mathbf{w}_{t-1}$

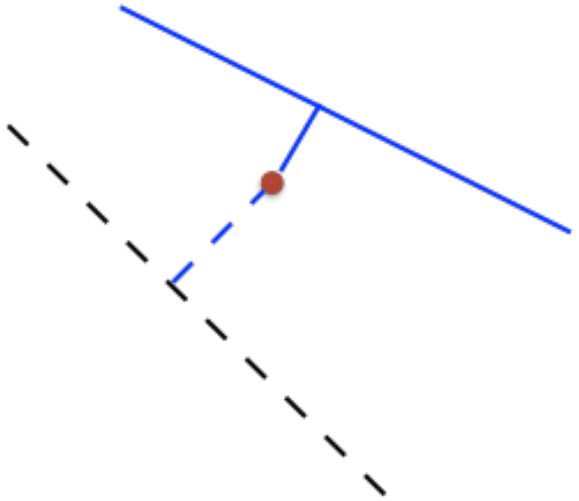
5: **end for**

- 如果没有误差，保持相同权重
- 否则，按加性规则更新

感知器: 直观解释

- 期望获得正向的间隔(margin)

$$\hat{y}_t \neq y_t \quad \text{iff} \quad \underbrace{y_t \mathbf{w}_{t-1}^T \mathbf{x}_t}_{\text{margin}} \leq 0$$



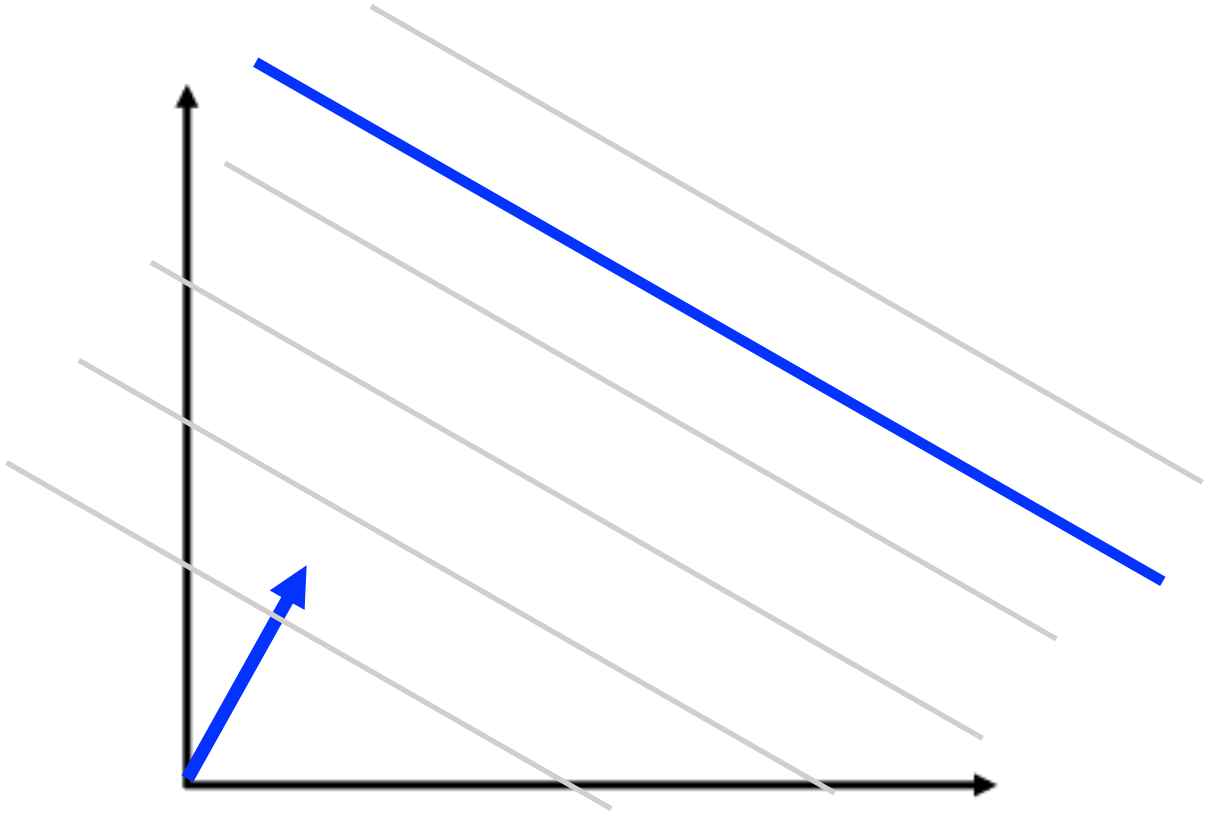
- 感知器更新间隔的影响

$$y_t \mathbf{w}_t^T \mathbf{x}_t = y_t (\mathbf{w}_{t-1} + y_t \mathbf{x}_t)^T \mathbf{x}_t = y_t \mathbf{w}_{t-1}^T \mathbf{x}_t + \|\mathbf{x}_t\|^2$$

间隔增加!

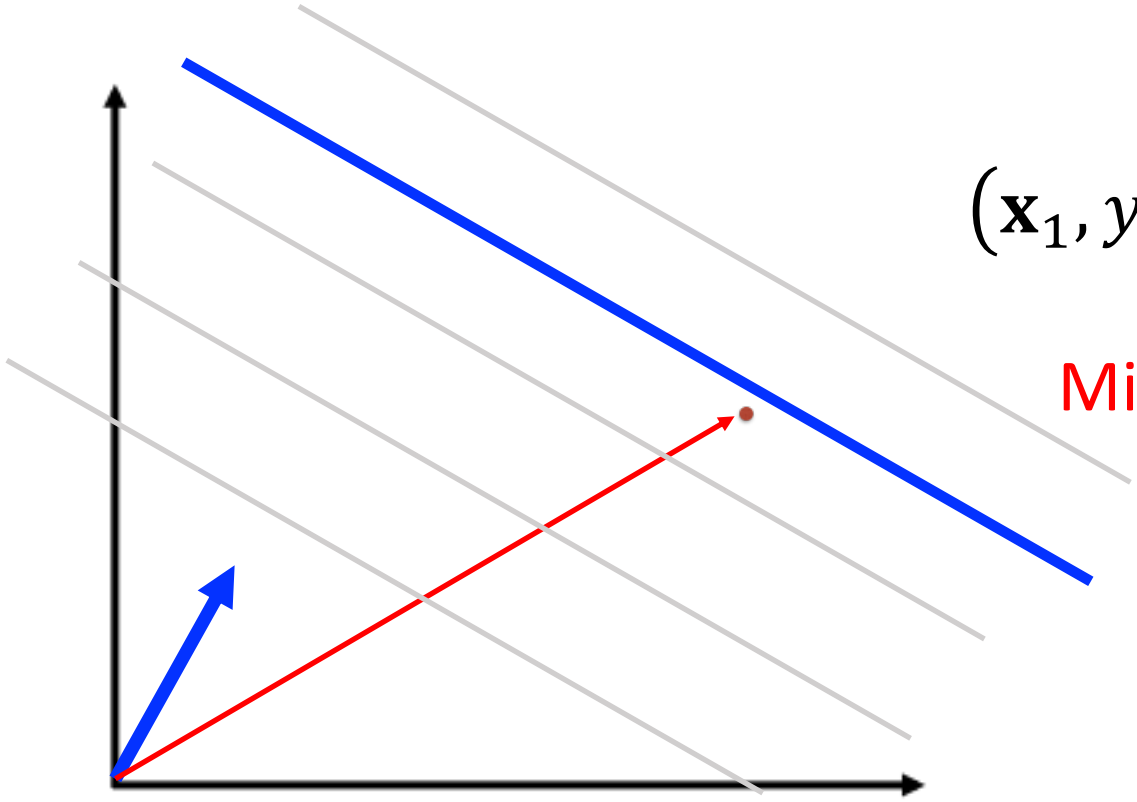
感知器: 几何解释

- 初始化



感知器: 几何解释

- $t = 1$

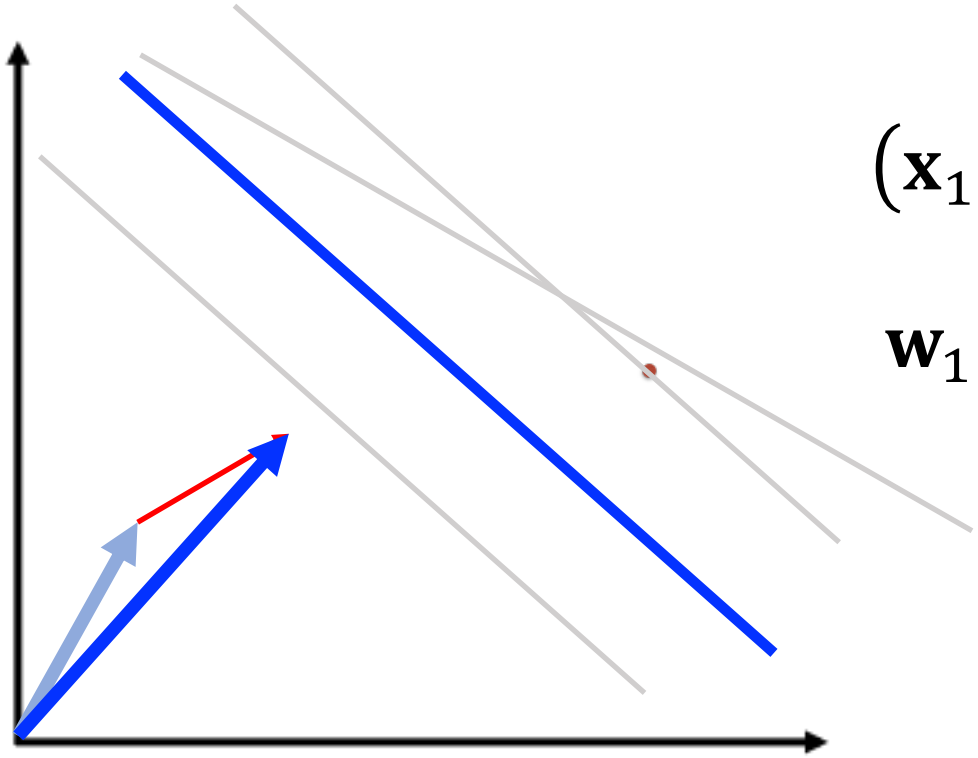


$(\mathbf{x}_1, y_1(= 1))$

Misclassification!

感知器: 几何解释

- 更新

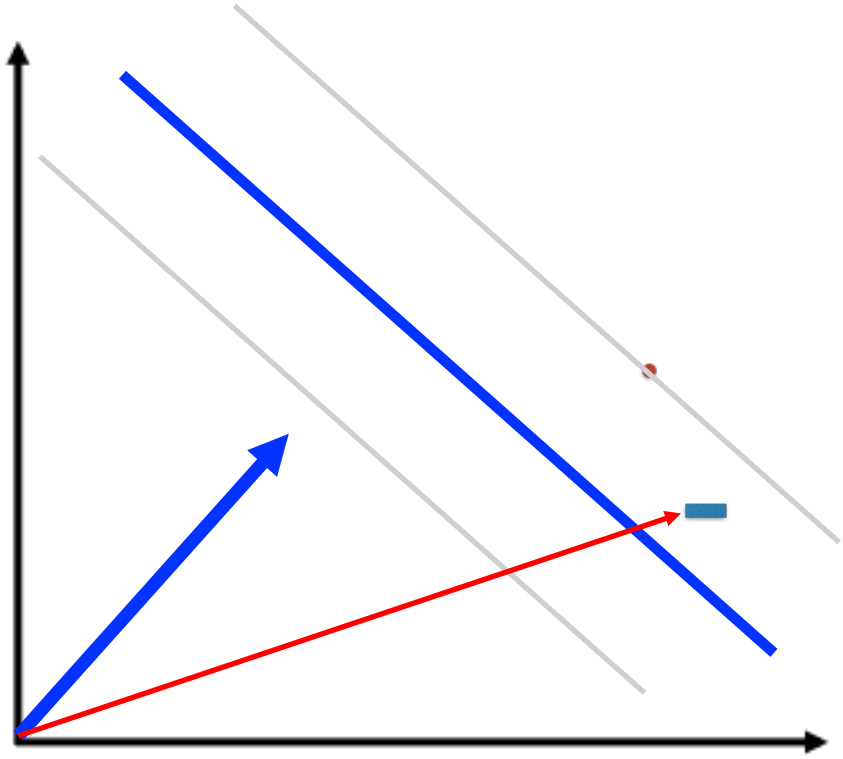


$$(\mathbf{x}_1, y_1(=1))$$

$$\mathbf{w}_1 = \mathbf{w}_0 + \eta \mathbf{x}_1 y_1$$

感知器: 几何解释

- $t = 2$



$(\mathbf{x}_1, y_1(= 1))$

$(\mathbf{x}_2, y_2(= -1))$

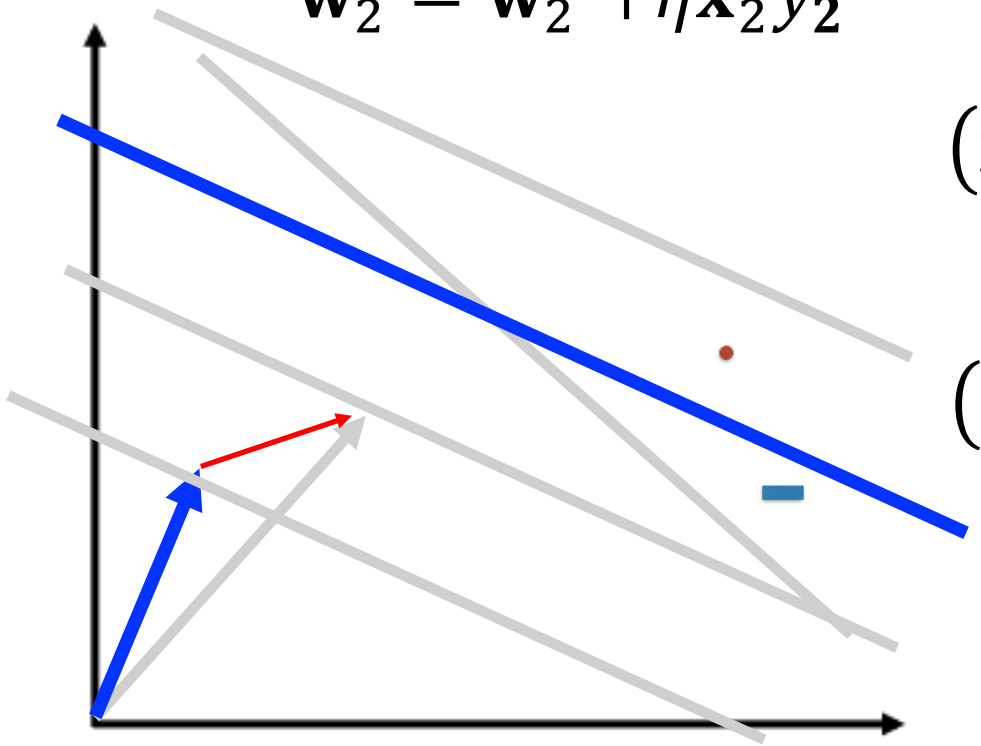
Misclassification!

感知器: 几何解释

- 更新

…

$$\mathbf{w}_2 = \mathbf{w}_2 + \eta \mathbf{x}_2 y_2$$



$(\mathbf{x}_1, y_1(=1))$

$(\mathbf{x}_2, y_2(=-1))$

感知器错误界

- 假设权值$\mathbf{w}_*$可以将数据$\mathbf{w}_*{}^T\mathbf{x}_i y_i$准确分开
- 定义间隔(margin)

$$\gamma = \frac{\min_i |\mathbf{w}_*^T \mathbf{x}_i|}{\|\mathbf{w}_*\|_2 \sup_i \|\mathbf{x}_i\|_2}$$

值越大，可信度越高

x的范数: 值越大, 错误界越大

- 感知器犯错误的次数最多为γ-2

支持向量机 Support Vector Machines

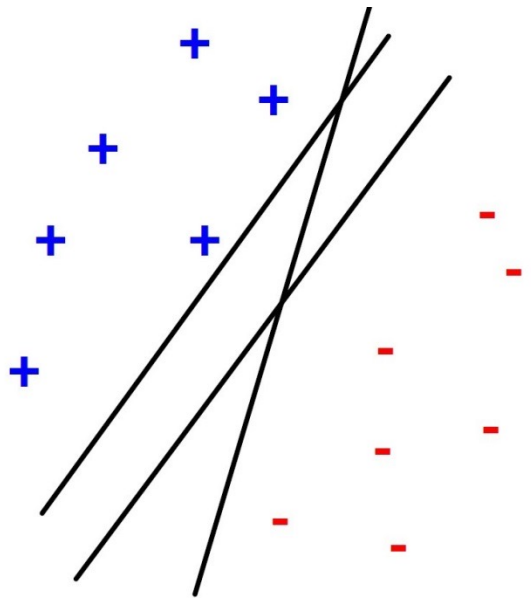
历史

- 支持向量机由Boser， Guyon & Vapnik于COLT-92提出
- 基于理论驱动的算法：60年代Vapnik & Chervonenkis发展了统计学习理论
- 性能表现不错：在许多领域(生物信息学，文本，图像识别，…)均表现良好

线性支持向量机/Support Vector Machine, SVM

- 训练集数据:
  - $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$
  - $\boldsymbol{x}_i \in X = Rd$
  - $y_i \in \mathcal{Y} = \{\pm 1\}$
- 目标:
  - 找到超平面$f(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b = 0$，可将二类样本分离开

- 哪个最好？

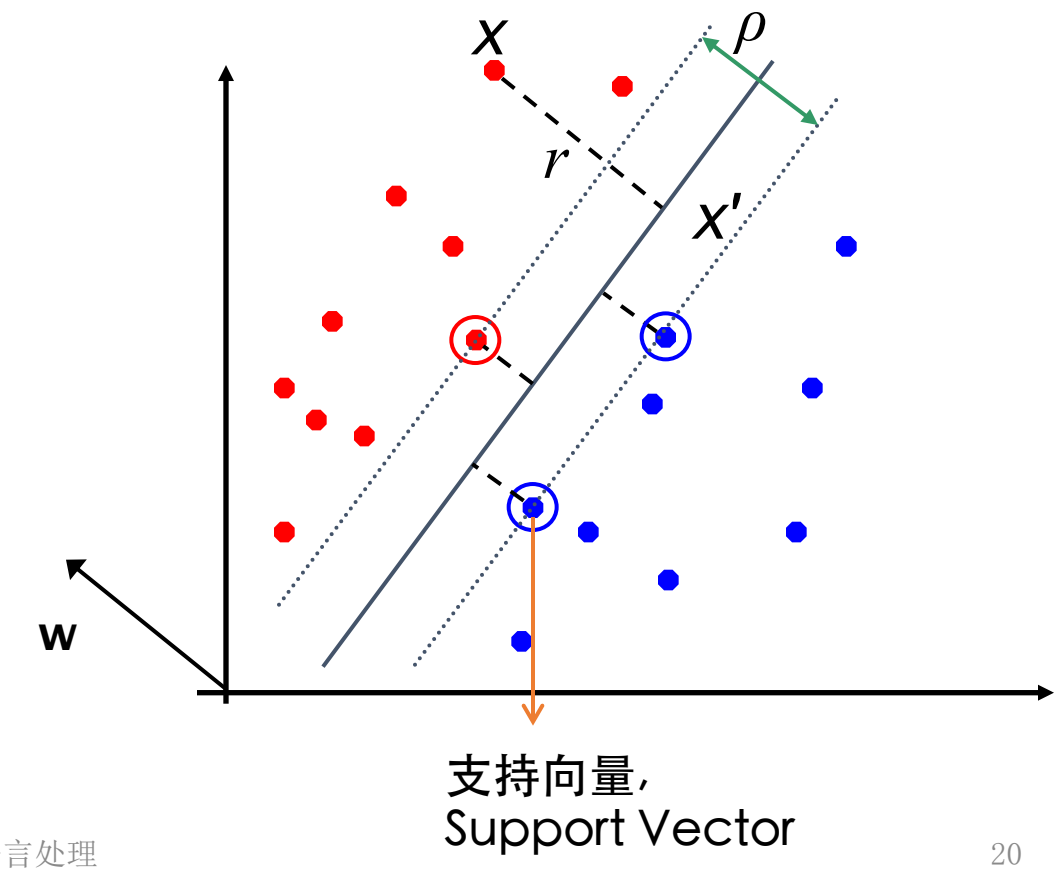


线性支持向量机

- 训练集数据:
  - $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$
  - $\boldsymbol{x}_i \in X = Rd$
  - $y_i \in \mathcal{Y} = \{\pm 1\}$

- 目标:
  - 找到分离超平面$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$，使得边界到正负样本的最小距离都最远

- 这个最好

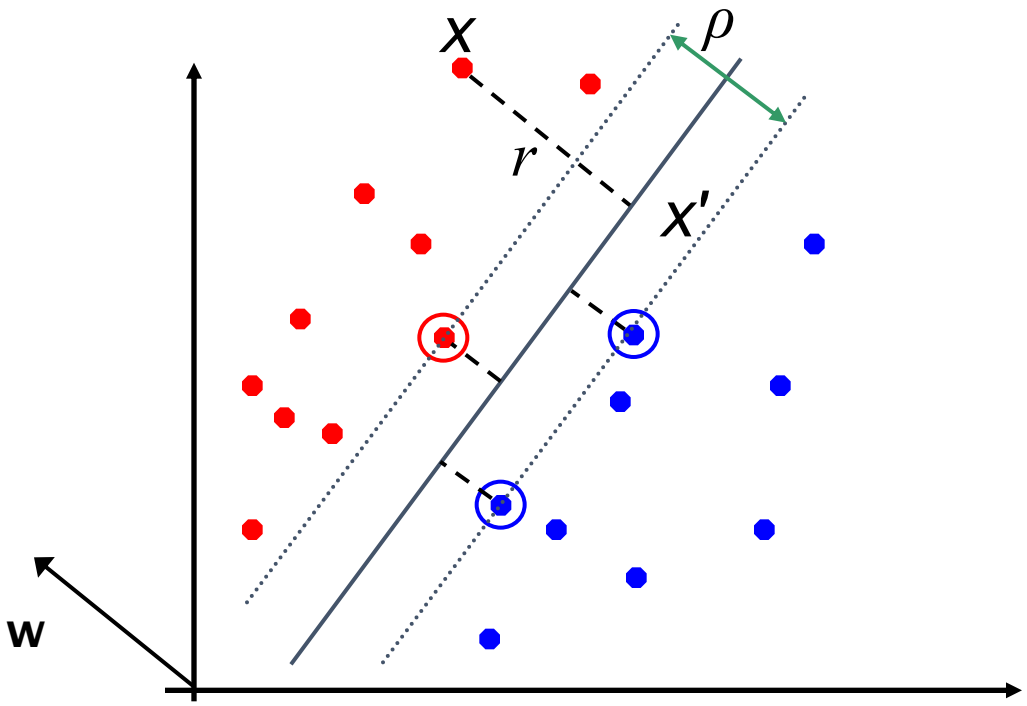


线性支持向量机

- **几何间隔**：样本点(x_i, y_i)到超平面的距离

$$\gamma_i = y_i \frac{\boldsymbol{w} \cdot \boldsymbol{x}_i + b}{\|\boldsymbol{w}\|}$$

- 证明:
  - $\boldsymbol{x}' - \boldsymbol{x} // \boldsymbol{w}$, 其中$\boldsymbol{w}/\|\boldsymbol{w}\|$是单位向量
    - 对应线段为 $r\boldsymbol{w}/\|\boldsymbol{w}\|$
    - $\boldsymbol{x}' = \boldsymbol{x} - yr\boldsymbol{w}/\|\boldsymbol{w}\|$
  - 由于 $\boldsymbol{x}'$在判定平面上， $\boldsymbol{w} \cdot \boldsymbol{x}' + b = 0$
    - 于是 $\boldsymbol{w} \cdot (\boldsymbol{x} - yr\boldsymbol{w}/\|\boldsymbol{w}\|) + b = 0$
  - 因为 $\boldsymbol{w} \cdot \boldsymbol{w} = \|\boldsymbol{w}\|^2$, $\boldsymbol{w} \cdot \boldsymbol{x} - yr\|\boldsymbol{w}\| + b = 0$
  - 因此，$r = y\frac{\boldsymbol{w} \cdot \boldsymbol{x} + b}{\|\boldsymbol{w}\|}$



线性支持向量机

- 目标: 最大化所有样本点的最小几何间隔

$$\max_{\boldsymbol{w},b} \gamma \left(:= \min_{i=1,\cdots,N} \gamma_i\right)$$

$$\text{s.t.}\, y_i \frac{\boldsymbol{w}\cdot\boldsymbol{x}_i + b}{\|\boldsymbol{w}\|} \geq \gamma,$$

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

- 定义函数间隔为：

$$\hat{\gamma} = \gamma\|\boldsymbol{w}\|$$

- 取函数间隔为1，得

$$\gamma = \frac{1}{\|\boldsymbol{w}\|}$$

- 代入获得目标函数

$$\max_{\boldsymbol{w},b} \frac{1}{\|\boldsymbol{w}\|}$$

$$\text{s.t.}\, y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq 1,$$

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

线性支持向量机

- 最大化间隔
  - 符合直觉、理论支持(VC维理论)、实际效果不错
- 线性支持向量机的目标函数:

$$\max_{\boldsymbol{w},b} \rho = \frac{2}{\|\boldsymbol{w}\|}$$

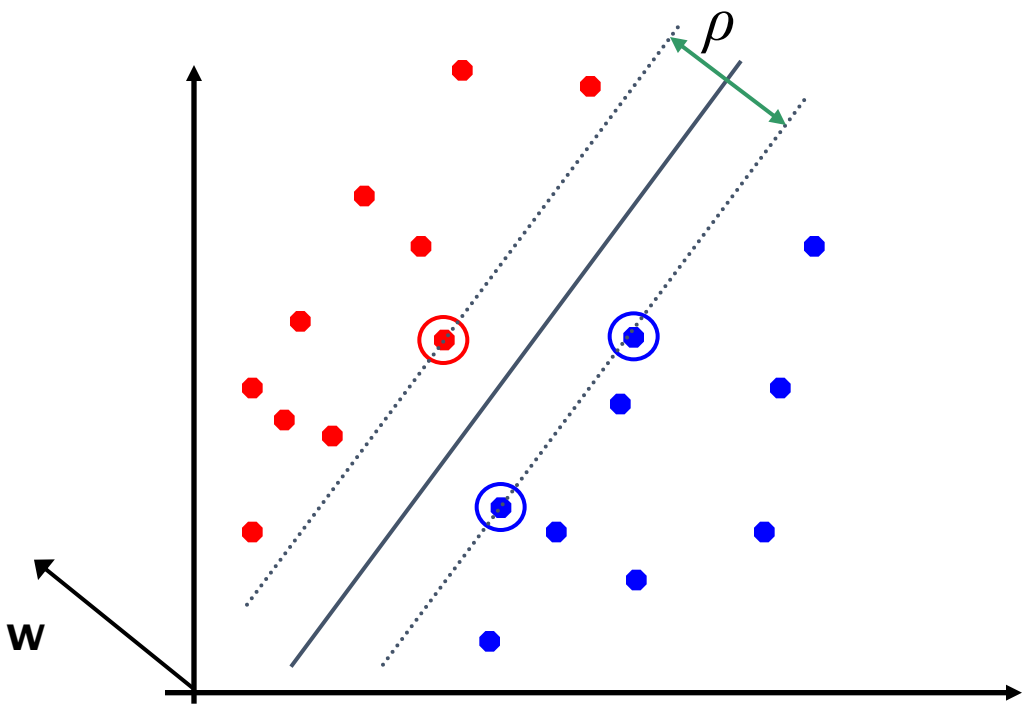
$$\text{s.t.}\, y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq 1, i = 1,2,\cdots,N$$

- 等价形式

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2$$

$$\text{s.t.}\, y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq 1, i = 1,2,\cdots,N$$

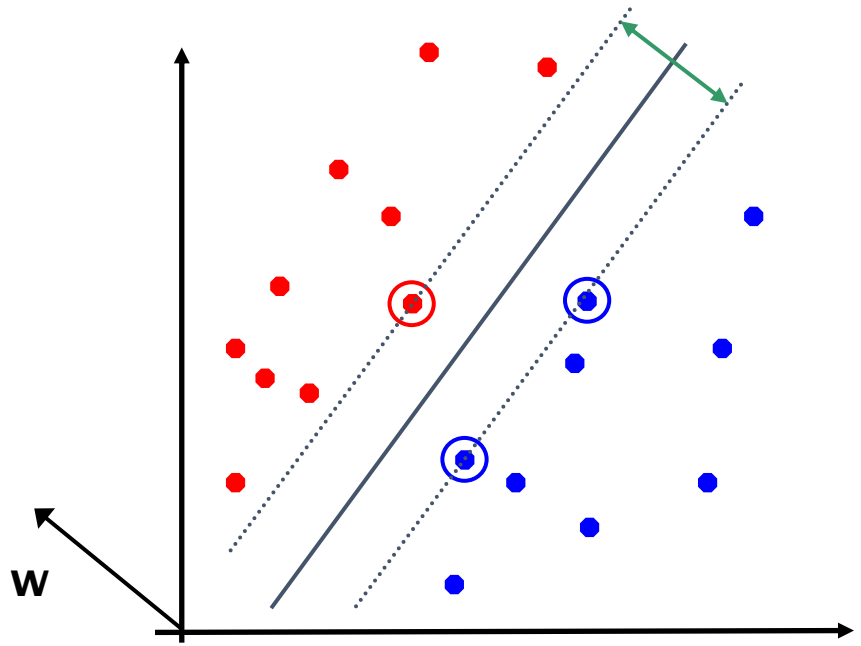
- 线性可分



线性支持向量机: Primal vs. Dual

- 原始/Primal

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2$$
$$\text{s.t.}\, y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq 1, i = 1,2,\cdots,N$$



- 对偶/Dual

$$\max_{\boldsymbol{\alpha}} -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y_i\alpha_i{\boldsymbol{x}_i}^T\boldsymbol{x}_i\alpha_j y_j + \sum_{i=1}^{N}\alpha_i,$$
$$\text{s.t.} \sum_{i=1}^{N} y_i\alpha_i = 0$$

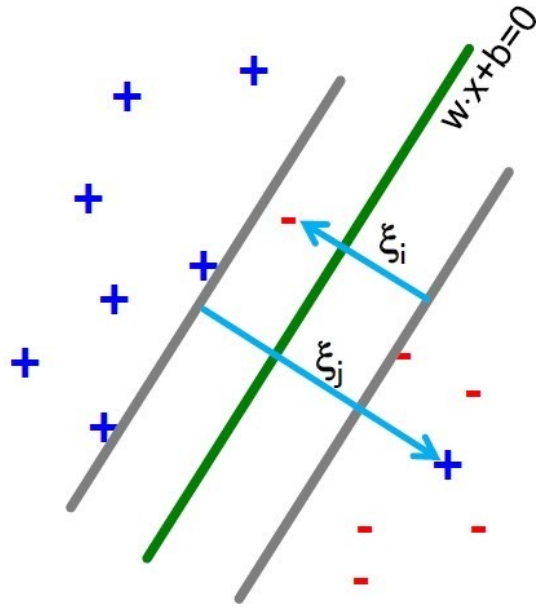
- KKT条件
  - $\nabla_{w^*}L = \boldsymbol{w}^* - \sum_{i=1}^{N}\alpha_i^* y_i\boldsymbol{x}_i = \boldsymbol{0} \Rightarrow \boldsymbol{w}^* = \boldsymbol{\alpha}^{*T}\widetilde{\boldsymbol{X}}$
  - $\nabla_b L = 0 \Rightarrow \boldsymbol{\alpha}^{*T}\mathbf{Y} = 0$
  - $\alpha_i^*[y_i(\boldsymbol{w}^*\cdot\boldsymbol{x}_i + b^*) - 1] = 0, \forall\, i \in [1,N]$
- 支持向量:
  - $\alpha_i^* \neq 0$

软间隔分类器/Soft-margin Classifier

- 带噪声的情形

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i$$
$$\text{s.t. } y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq 1 - \xi_i,$$
$$\xi_i \geq 0,$$
$$i = 1,2,\cdots,N$$

- 线性不可分



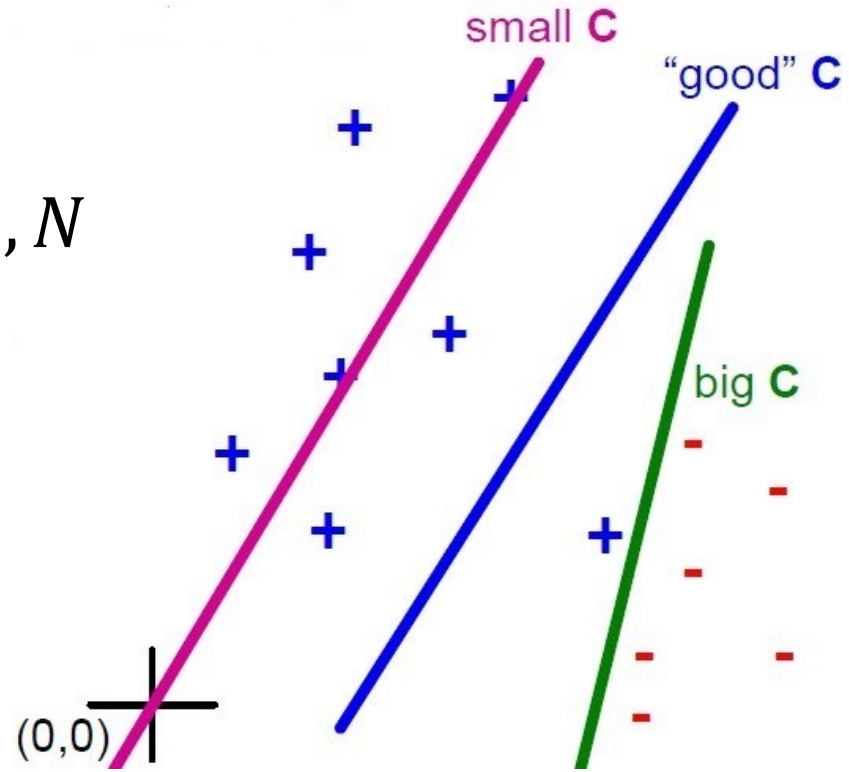
- 间隔 ≥ 1，误差为0
- 间隔<1，线性惩罚
- ξ_i：松弛变量 /Slack variables

松弛惩罚/Slack Penalty C

- 目标函数

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i$$
$$\text{s.t. } y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1,2,\cdots,N$$

- C的影响
  - $C = 0$: ξ_i可设任意值
    - $\boldsymbol{w} = \boldsymbol{0}$: 忽略数据
  - $C = \infty$: $\xi_i = 0$
    - 依靠$\boldsymbol{w}, b$ 分离数据



软间隔分类器

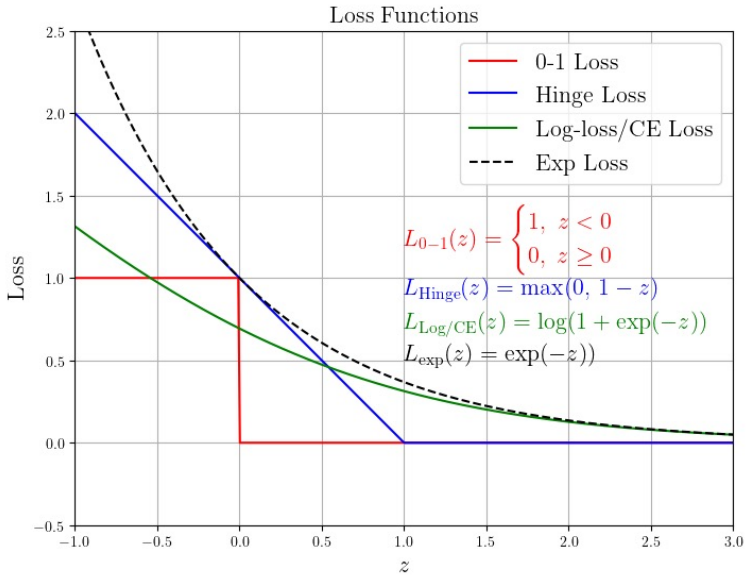
- 目标函数
  - $\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N} \max\{0, 1 - y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b)\}$

Margin /间隔

Regularization Parameter/正则参数

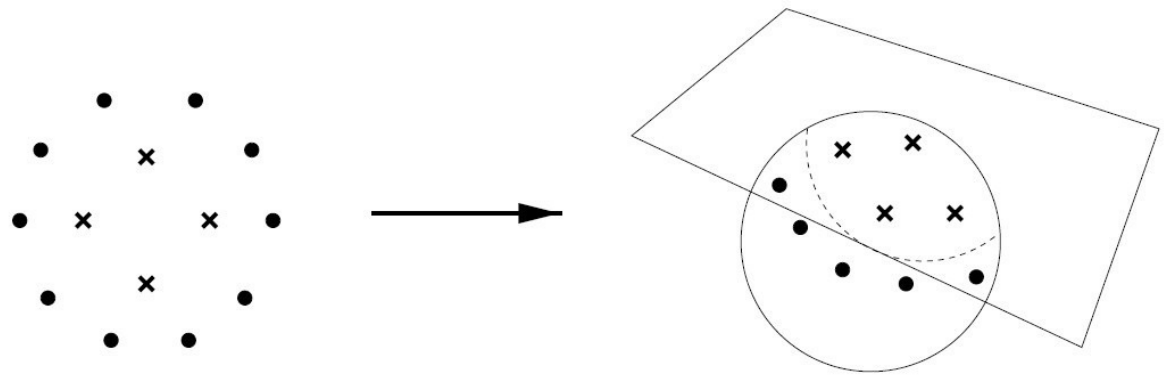
Empirical loss L/经验损失

  - Hinge loss
    - $L_{\text{Hinge}}(z) = \max\{0, 1 - z\}$



非线性可分的支持向量机

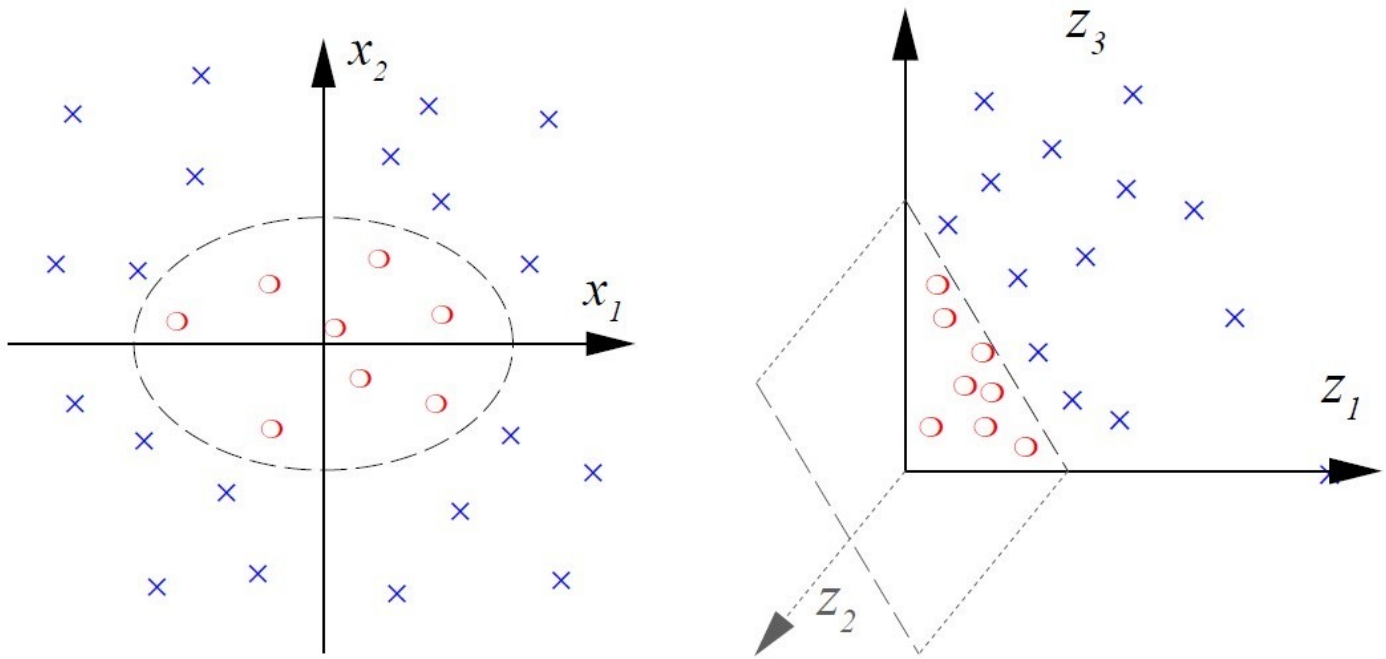
- 线性分类器不够复杂
  - 将数据映射到更丰富的特征空间，包括非线性特征
  - 在此空间中构造一个超平面，使其核心方程仍保持一致
- 数据映射
  - $\boldsymbol{x} \mapsto \Phi(\boldsymbol{x}): R^d \to R^\phi$
  - ϕ可能是无穷维
- 判定函数
  - $f(\boldsymbol{x}) = \boldsymbol{w} \cdot \Phi(\boldsymbol{x}) + b$



例子: 多项式映射

$$\Phi : R^2 \rightarrow R^3$$

$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{(2)} x_1 x_2, x_2^2)$$



SVM: 核技巧/Kernel Tricks

- SVM的对偶目标函数是二次凸规划

$$\max_{\boldsymbol{\alpha}} -\frac{1}{2}\boldsymbol{\alpha}^T\mathbf{Q}\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\mathbf{1}_N, \text{s.t. } \mathbf{0} \le \boldsymbol{\alpha} \le C\mathbf{1}_N, \boldsymbol{\alpha}^T\mathbf{Y} = 0$$

  - $Q_{ij} = y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $\mathbf{1}_N = (1, 1, \cdots, 1)^T$, $\mathbf{Y} = (y_1, y_2, \cdots, y_N)^T$
- 判定函数:
  - $f(\boldsymbol{x}) = \boldsymbol{w} \cdot \Phi(\boldsymbol{x}) + b = \sum_{i=1}^{N} \alpha_i\, y_i \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}) + b = \sum_{i=1}^{N} \alpha_i\, y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b$
    - $\boldsymbol{w} = \sum_{i=1}^{N} \alpha_i y_i\, \Phi(\boldsymbol{x}_i)$
      - 表示理论(representer theorem, Kimeldorf & Wahba, 1971))
      - 支持向量: $\alpha_i \neq 0$
    - b使得$\boldsymbol{\alpha}^T\mathbf{Y} = 0$
- 核函数: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$

核函数

- 为什么使用核函数？
  - 将不可分的分类问题变得可分
  - 将数据映射到更好的表示空间
- 常用核函数
  - 线性
    - $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^T \boldsymbol{x}_j$
  - 多项式/Polynomial
    - $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \boldsymbol{x}_i^T \boldsymbol{x}_j)^d$
  - 径向基函数 (Radial basis function)
    - $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = e^{-\gamma \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2} = e^{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 / 2\sigma^2}$

In-class Practice

SVM: 如何获得$\boldsymbol{w}, b$

- 以软间隔分类器为例:

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

- 常用方法: 工具包
  - LIBSVM: http://www.csie.ntu.edu.tw/~cjlin/libsvm/

- 其他方法: 梯度下降

SVM: 如何获得$\boldsymbol{w}$

- 最小化下述目标函数 $G(\boldsymbol{w}, b)$：

$$G(\boldsymbol{w}, b) = \frac{1}{2}\sum_{j=1}^{d}\left(w^{(j)}\right)^2 + C\sum_{i=1}^{N}\max\left\{0, 1 - y_i\left(\sum_{j=1}^{d} w^{(j)}\, x_i^{(j)} + b\right)\right\}$$

Empirical loss L/经验损失

- 计算梯度

$$\nabla(j) = \frac{\partial G(\boldsymbol{w}, b)}{\partial w^{(j)}} = w^{(j)} + C\sum_{i=1}^{N}\frac{\partial L(\boldsymbol{x}_i, y_j)}{\partial w^{(j)}}$$

$$\frac{\partial L(\boldsymbol{x}_i, y_j)}{\partial w^{(j)}} = \begin{cases} 0 & \text{if } y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq 1 \\ -y_i x_i^{(j)} & \text{otherwise} \end{cases}$$

SVM: 如何获得$\boldsymbol{w}$

- 梯度下降: 迭代直到收敛

For $j = 1, \cdots, d$

- 计算: $\nabla(j) = \frac{\partial G(\boldsymbol{w},b)}{\partial w^{(j)}} = w^{(j)} + C \sum_{i=1}^{N} \frac{\partial L(\boldsymbol{x}_i, y_j)}{\partial w^{(j)}}$
- 更新: $w^{(j)} = w^{(j)} - \eta \nabla(j)$

- 问题
  - 梯度计算$\nabla(j)$耗时$O(N)$
    - N: 训练样本数

- 随机梯度下降 (SGD)
  - 对每个单独的训练样本进行评估
    - $\nabla(j,i) = \frac{\partial G(\boldsymbol{w},b)}{\partial w^{(j)}} = w^{(j)} + C \frac{\partial L(\boldsymbol{x}_i, y_j)}{\partial w^{(j)}}$

- 算法: 迭代直到收敛

For $i = 1, \cdots, N$

For $j = 1, \cdots, d$

- 计算: $\nabla(j,i)$
- 更新: $w^{(j)} = w^{(j)} - \eta \nabla(j,i)$

例子: 文本分类

- Leon Bottou的论文给出的例子：
  - 路透社RCV1文档语料库
    - 预测文档的类别
      - 1 vs. 其余分类
  - N = 781,000个训练样本(文档)
  - 23,000个测试样本
  - d = 5万个特征 (词袋模型)
    - 每个单词一个特征
    - 删除停用词
    - 删除低频词

Léon Bottou and Olivier Bousquet: The Tradeoffs of Large Scale Learning, Optimization for Machine Learning, 351-368, 2011

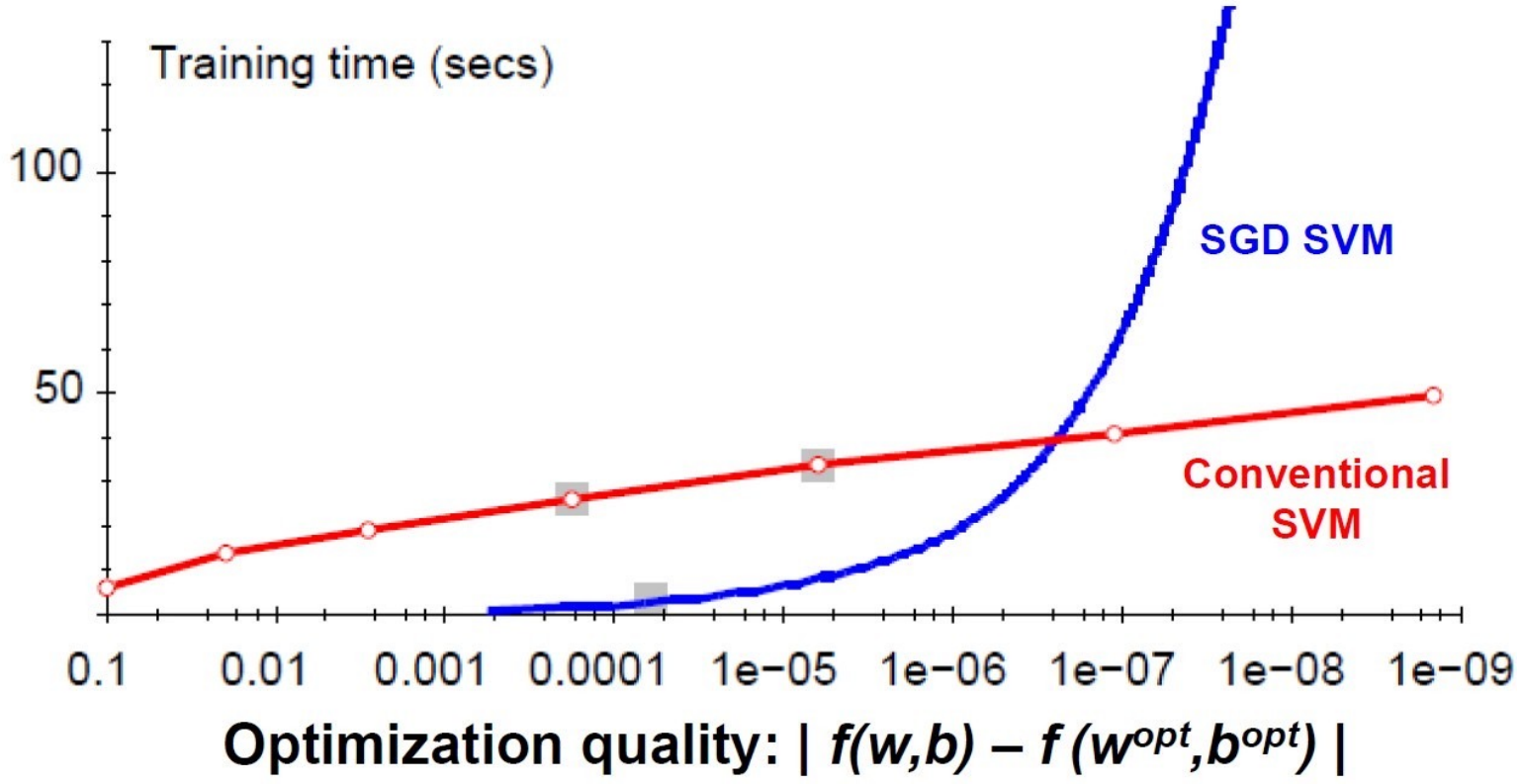
例子: 文本分类

- 问题:
  - SGD能否找到$f(\boldsymbol{w}, b)$ 的最小值？
  - SGD找到$f(\boldsymbol{w}, b)$的最小值有多快？
  - 测试集的误差是多少？

	Training time	*Value of f(w,b)*	*Test error*
Standard SVM	23,642 secs	*0.2275*	*6.02%*
"Fast SVM"	66 secs	0.2278	6.03%
SGD SVM	1.4 secs	*0.2275*	*6.02%*

- 结论
  - SGD-SVM成功地找到$f(\boldsymbol{w}, b)$ 的最小值
  - SGD-SVM非常快
  - SGD-SVM测试集误差相当

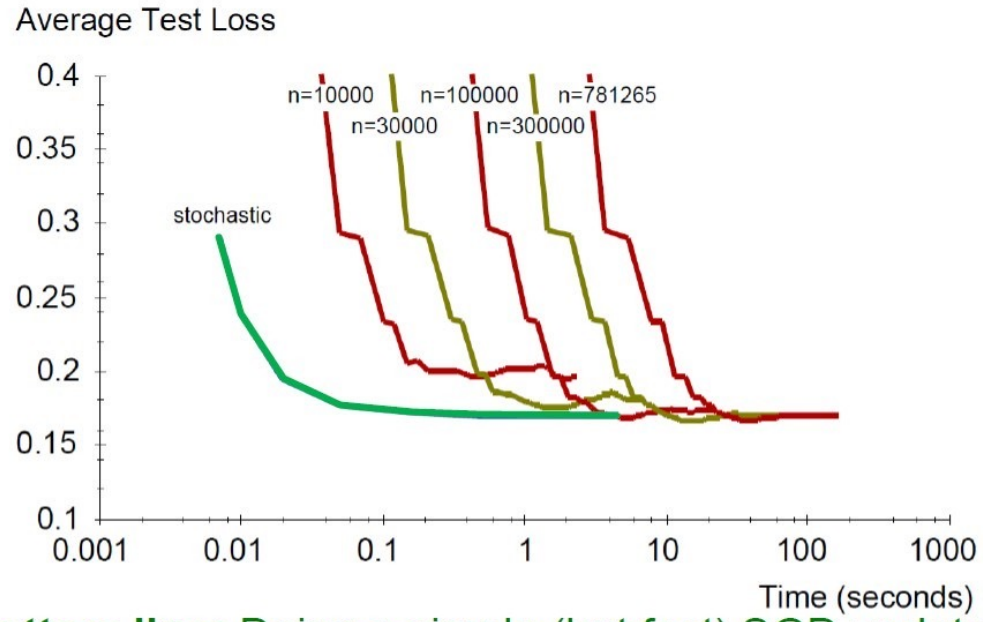
优化“准确性”



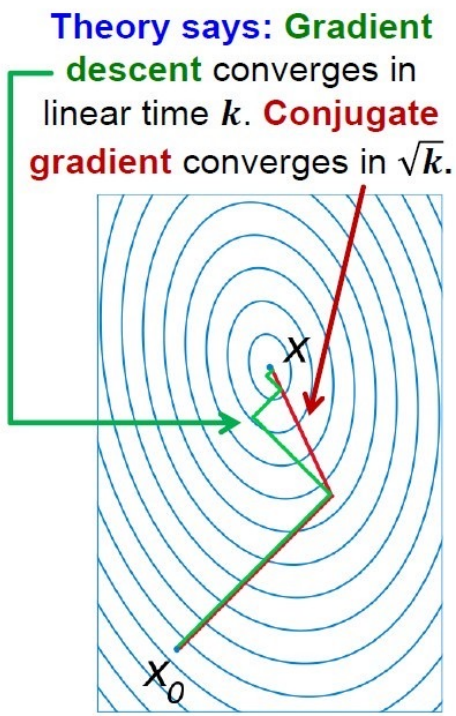
For optimizing *f(w,b) within reasonable* quality
SGD-SVM is super fast

SGD vs. Batch Conjugate Gradient

- SGD on full dataset vs. Batch Conjugate
  - Gradient on a sample of *N* training examples



Bottom line: Doing a simple (but fast) SGD update many times is better than doing a complicated (but slow) BCG update a few times



k… condition number

实际考量

- 需要选择学习速率η, t_0

$$w_{t+1} \leftarrow w_t - \frac{\eta_t}{t + t_0}\left(w_t + C\frac{\partial L(x_i, y_i)}{\partial w}\right)$$

- Leon建议:
  - 选择t_0使预期的初始更新与预计权重的量级相当
  - 选择η:
    - 选择一个小的子样本
    - 尝试不同的学习率η (如: 10, 1, 0.1, 0.01, …)
    - 选择一个最能损失函数的值
    - 使用η在全数据集下迭代100k次

实际考量

- 稀疏线性SVM：
  - 特征向量$\boldsymbol{x}_i$是稀疏的（即包含多个零）
  - 如何表示? 如 $x_i = [0,0,0,1,0,0,0,0,5,0,0,0,0,0,0,\ldots]$
    - $\boldsymbol{x}_i = [(4,1),(9,5)]$
- 如何高效更新梯度?

$$w \leftarrow w - \eta \left(w + C \frac{\partial L(x_i, y_i)}{\partial w} \right)$$

- 两阶段近似：

$$w \leftarrow w - \eta C \frac{\partial L(x_i, y_i)}{\partial w}$$

$$w \leftarrow w(1 - \eta)$$

有效: 如果w稀疏，仅部分坐标j被更新
昂贵: 如果w是稠密的，所有坐标都需要更新

实际考量

- 方案1: $\boldsymbol{w} = s\boldsymbol{v}$
  - 向量w表示成标量s和向量$\boldsymbol{v}$的乘积
  - 更新过程为：
    1. $v = v - \eta C \frac{\partial L(x_i, y_i)}{\partial w}$
    2. $s = s(1 - \eta)$
- 方案2
  - 对每个训练样例只执行步骤1
  - 以较低的频率执行步骤2)更新较大的学习率η

- 两阶段近似

$$w \leftarrow w - \eta C \frac{\partial L(x_i, y_i)}{\partial w}$$

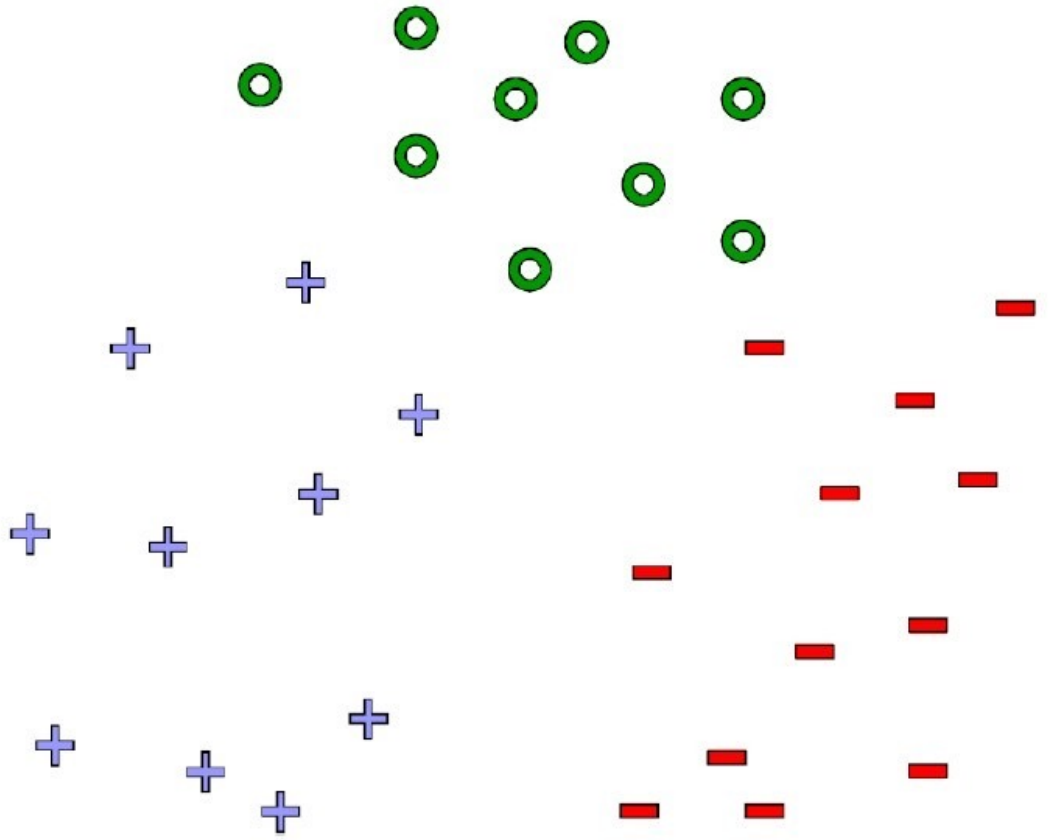
$$w \leftarrow w(1 - \eta)$$

实际考量

- 停止条件: SGD需要迭代多少次？
  - 用交叉验证的方式提前停止
    - 创建验证集
    - 监视验证集上的损失函数
    - 当损失停止减少时停止
- 步骤
  - 提取训练数据的两个不相交子样本A和B
  - 在A上训练，在B上验证停止
  - epoch的个数是k的估计值
  - 在全数据集上运行k次(epochs)

SVM: 多分类/Multiple Classes

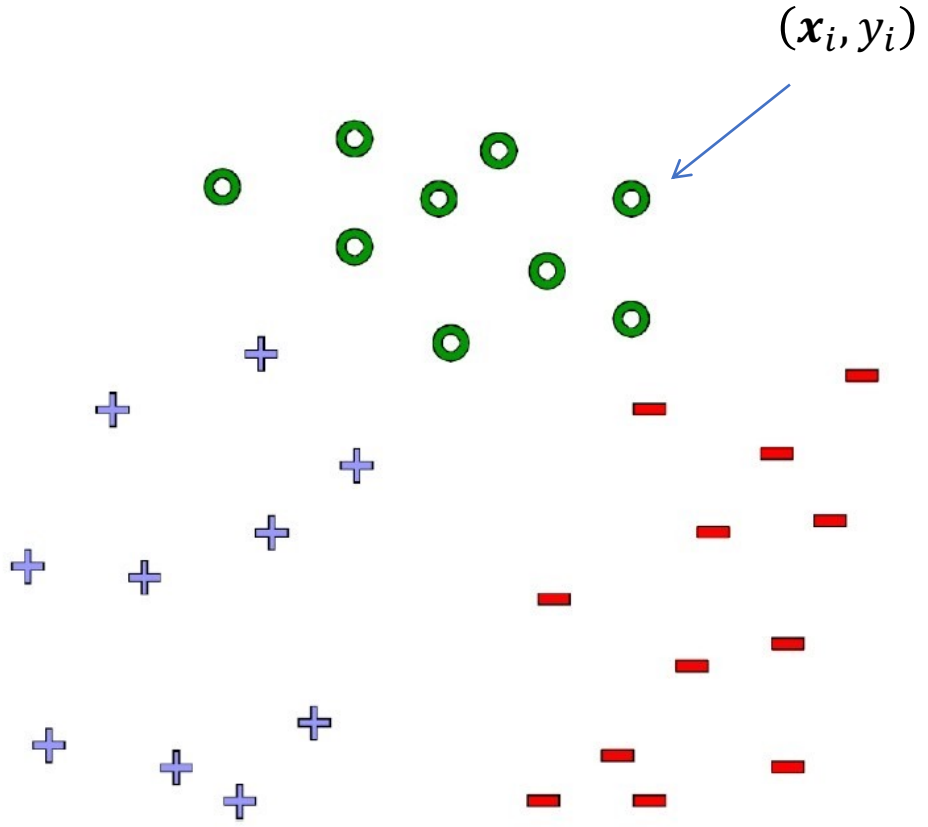
- 方案1: One vs. rest
  - 学习三个分类器
    - + vs. {0, –}
    - – vs. {0, +}
    - 0 vs. {+, –}
  - 获得参数:
    - $\boldsymbol{w}_+, b_+; \boldsymbol{w}_-, b_-; \boldsymbol{w}_o, b_o$
  - 决策:
    - $\underset{c}{\operatorname{argmax}}\, \boldsymbol{w}_c \boldsymbol{x} + b_c$



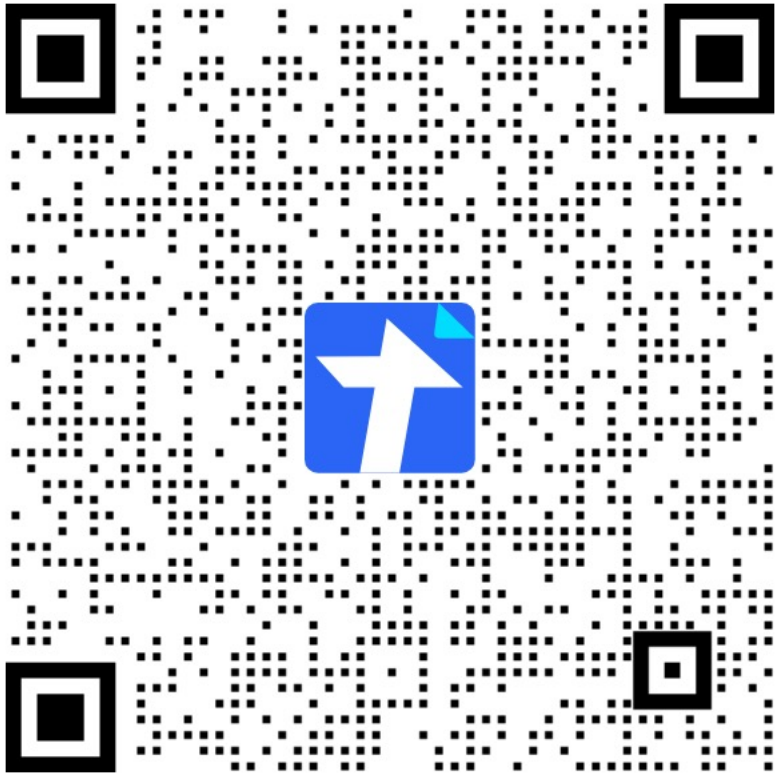
SVM: 多分类/Multiple Classes

- 方案2
  - 同时学习三组分类器参数
  - 正确的类有最大的间隔

$$\boldsymbol{w}_{y_i}\boldsymbol{x}_i + b_{y_i} \geq 1 + \boldsymbol{w}_c\boldsymbol{x}_i + b_c,$$
$$\forall c \neq y_i, \forall i$$



一句话总结



额外阅读

- http://i.stanford.edu/~ullman/pub/ch12.pdf
- http://www.csie.ntu.edu.tw/~cjlin/libsvm/
- http://www.cs.columbia.edu/~kathy/cs4701/documents/jason_svm_tutorial.pdf

Appendix

1.线性支持向量机推导

2. 非线性软间隔支持向量机推导

线性支持向量机的推导(拉格朗日法/Lagrangian)

- 目标函数
  - $\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2$, s. t. $y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b)\geq 1, i=1,2,\cdots,N$
- 引入拉格朗日算子/multipliers
  - $L(\boldsymbol{w},b)=\frac{1}{2}\|\boldsymbol{w}\|^2-\sum_{i=1}^{N}\alpha_i(y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b)-1)$
- 最小值条件

$$\frac{\partial L}{\partial \boldsymbol{w}}=\boldsymbol{w}-\sum_{i=1}^{N}\alpha_i y_i\boldsymbol{x}_i=\boldsymbol{0}\Rightarrow \boldsymbol{w}=\boldsymbol{\alpha}^T\widetilde{\boldsymbol{X}}$$

$$\frac{\partial L}{\partial b}=\sum_{i=1}^{N}\alpha_i y_i=0\Rightarrow \boldsymbol{\alpha}^T\boldsymbol{Y}=0$$

线性支持向量机的推导(拉格朗日法)

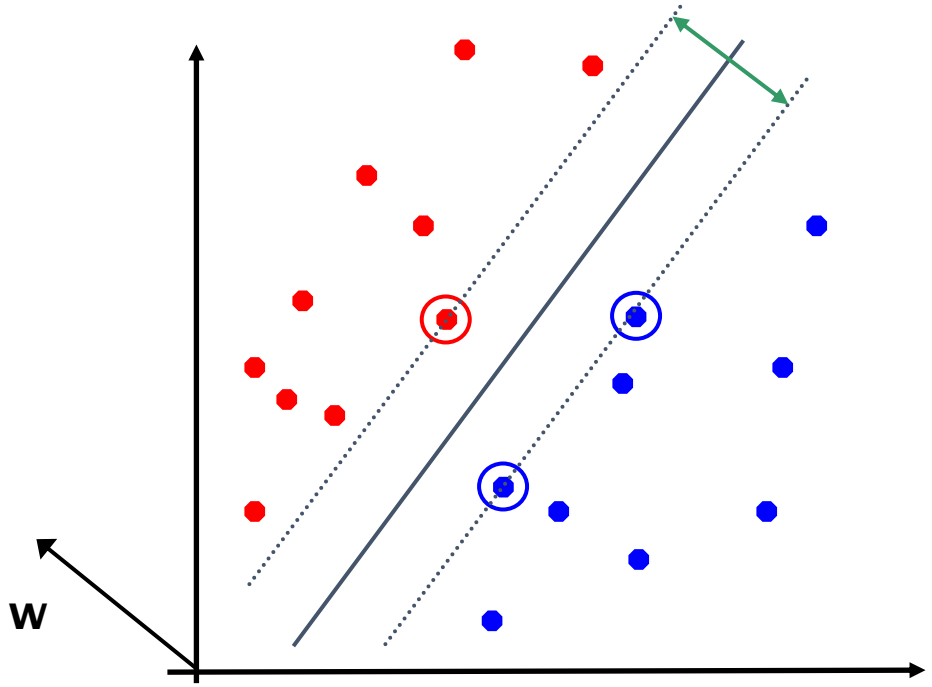
- 将 $\boldsymbol{w} = \boldsymbol{\alpha}^T \widetilde{\boldsymbol{X}}, \boldsymbol{\alpha}^T \boldsymbol{Y} = 0$, 代入
  - $L(\boldsymbol{w}, b) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{N} \alpha_i (y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1)$
- 得到

$$\max_{\boldsymbol{\alpha}} -\frac{1}{2}\boldsymbol{\alpha}^T \widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{X}}^T \boldsymbol{\alpha} + \boldsymbol{\alpha}^T \mathbf{1}_N, \text{s.t. } \boldsymbol{\alpha}^T \boldsymbol{Y} = 0$$

  - 其中
    - $\boldsymbol{\alpha}^T \widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{X}}^T \boldsymbol{\alpha} = \sum_{i=1}^{N}\sum_{j=1}^{N} y_i \alpha_i \boldsymbol{x}_i{}^T \boldsymbol{x}_i \alpha_j y_j = \sum_{i=1}^{N}\sum_{j=1}^{N} y_i \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_j)\, \alpha_j\, y_j$

线性支持向量机: Karush-Kuhn-Tucker (KKT)条件

- 最优值时:
  - 梯度条件
    - $\nabla_{\boldsymbol{w}^*} L = \boldsymbol{w}^* - \sum_{i=1}^{N} \alpha_i^* y_i \boldsymbol{x}_i = \boldsymbol{0} \Rightarrow \boldsymbol{w}^* = \boldsymbol{\alpha}^{*T} \widetilde{\boldsymbol{X}}$
    - $\nabla_b L = 0 \Rightarrow \boldsymbol{\alpha}^{*T} \boldsymbol{Y} = 0$
  - 原始条件
    - $\alpha_i^* \geq 0, y_i(\boldsymbol{w}^* \cdot \boldsymbol{x}_i + b^*) - 1 \geq 0, \forall\, i \in [1, N]$
  - 对偶条件
    - $\alpha_i^* \geq 0, \forall\, i \in [1, N]$
  - 互补松弛条件
    - $\alpha_i^* [y_i(\boldsymbol{w}^* \cdot \boldsymbol{x}_i + b^*) - 1] = 0, \forall\, i \in [1, N]$
- 支持向量:$\alpha_i^* \neq 0$



非线性软间隔支持向量机的推导(拉格朗日法/ Lagrangian)

- 目标函数
  - $\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i$, s. t. $y_i(\boldsymbol{w}\cdot\Phi(\boldsymbol{x}_i)+b) \geq 1-\xi_i, \xi_i \geq 0, i = 1,2,\cdots,N$
- 引入拉格朗日算子
  - $L(\boldsymbol{w},b) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i(y_i(\boldsymbol{w}\cdot\Phi(\boldsymbol{x}_i)+b)-1+\xi_i) - \sum_{i=1}^{N}\beta_i\xi_i$
- 最小值条件

$$\frac{\partial L}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_{i=1}^{N}\alpha_i y_i \Phi(\boldsymbol{x}_i) = \boldsymbol{0} \Rightarrow \boldsymbol{w} = \boldsymbol{\alpha}^T\widetilde{\boldsymbol{X}}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N}\alpha_i y_i = 0 \Rightarrow \boldsymbol{\alpha}^T\mathbf{Y} = 0$$

$$\left.\begin{array}{l}\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \beta_i = 0 \\ \alpha_i \geq 0, i = 1,2,\cdots,N\end{array}\right\} \Rightarrow \boldsymbol{0} \leq \boldsymbol{\alpha} \leq C\boldsymbol{1}_N$$

非线性软间隔支持向量机的推导(拉格朗日法)

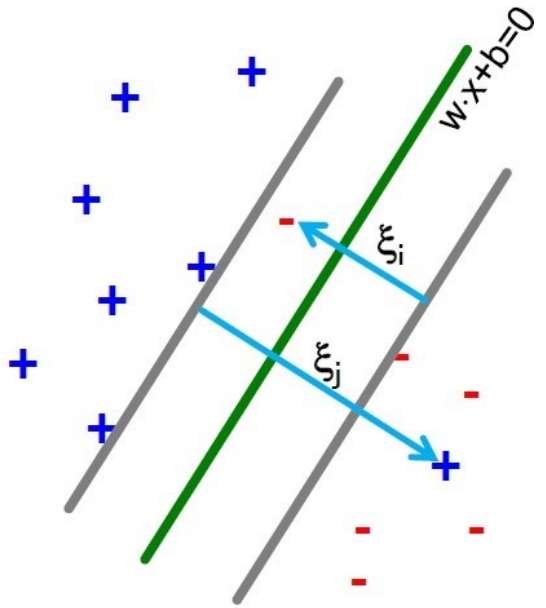
- 将 $\boldsymbol{w} = \boldsymbol{\alpha}^T \widetilde{\boldsymbol{X}}, \boldsymbol{\alpha}^T \boldsymbol{Y} = 0, C = \alpha_i + \beta_i$代入
  - $L(\boldsymbol{w}, b) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i(y_i(\boldsymbol{w}\cdot\Phi(\boldsymbol{x}_i) + b) - 1 + \xi_i) - \sum_{i=1}^{N}\beta_i\xi_i$
- 得到

$$\max_{\boldsymbol{\alpha}} -\frac{1}{2}\boldsymbol{\alpha}^T\widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{X}}^T\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\mathbf{1}_N, \text{s.t. } \mathbf{0} \le \boldsymbol{\alpha} \le C\mathbf{1}_N$$

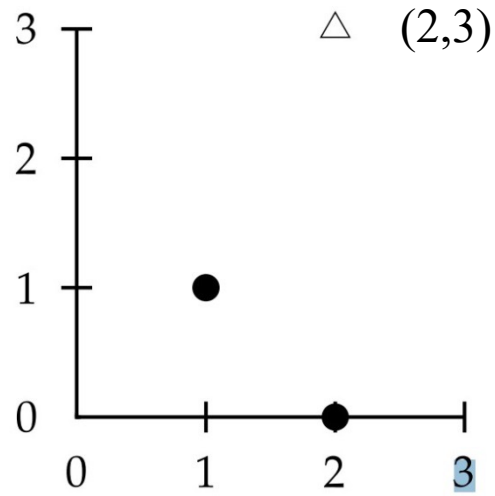
  - 其中
    - $\boldsymbol{\alpha}^T\widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{X}}^T\boldsymbol{\alpha} = \sum_{i=1}^{N}\sum_{j=1}^{N} y_i\alpha_i\Phi(\boldsymbol{x}_i)^T\Phi(\boldsymbol{x}_j)\alpha_j y_j = \sum_{i=1}^{N}\sum_{j=1}^{N} y_i\alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_j)\,\alpha_j\, y_j$

非线性软间隔支持向量机：Karush-Kuhn-Tucker (KKT)条件

- 最优值时:
  - 梯度条件
    - $\nabla_{\boldsymbol{w}^*} L = \boldsymbol{w}^* - \sum_{i=1}^{N} \alpha_i^* y_i \Phi(\boldsymbol{x}_i) = \boldsymbol{0} \Rightarrow \boldsymbol{w}^* = \boldsymbol{\alpha}^{*T} \widetilde{\boldsymbol{X}}$
    - $\nabla_b L = 0 \Rightarrow \boldsymbol{\alpha}^{*T} \boldsymbol{Y} = 0$
    - $\nabla_{\xi_i^*} L = 0 \Rightarrow C - \alpha_i^* - \beta_i^* = 0, \forall\, i \in [1, N]$
  - 原始条件
    - $\alpha_i^* \geq 0, y_i(\boldsymbol{w}^* \cdot \Phi(\boldsymbol{x}_i) + b^*) - 1 + \xi_i^* \geq 0, \forall\, i \in [1, N]$
  - 对偶条件
    - $\alpha_i^* \geq 0, \beta_i^* \geq 0$
  - 互补松弛条件
    - $\alpha_i^* [y_i(\boldsymbol{w}^* \cdot \Phi(\boldsymbol{x}_i) + b^*) - 1 + \xi_i^*] = 0$
    - $\beta_i^* \xi_i^* = 0$
- 支持向量:$\alpha_i^* \neq 0$

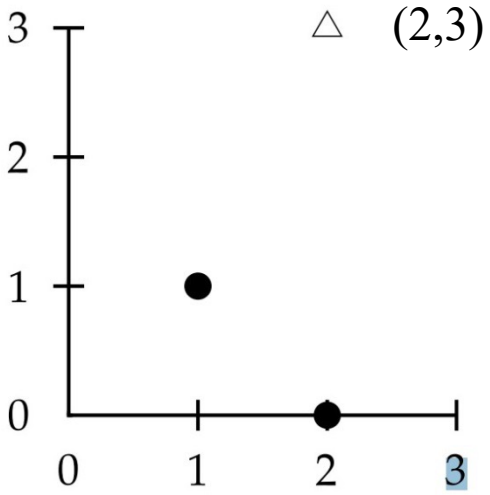


In-class Practice



- 根据上图所示的数据集，构建支持向量机/SVM，计算其决策边界

Ans. to In-class Practice



- 我们需要最小化$|\boldsymbol{w}|$，同时满足$y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1$
- 有两个支持向量，$(1,1), (2, 3)$，满足上述约束
- 于是: 最大边距权重向量将平行于两个类点之间的最短连线，即$(1,1)$和$(2, 3)$之间的连线
  - $\boldsymbol{w} = (a, 2a)^T$
- 代入约束
  - $a + 2a + b = -1$
  - $2a + 6a + b = 1$
- 解得: $a = \frac{2}{5}, b = -\frac{11}{5}$
- 决策边界: $f(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b = \frac{2}{5}x_1 + \frac{4}{5}x_2 - \frac{11}{5}$