

支持向量机

高 阳，李文斌

<http://cs.nju.edu.cn/rl>, 2021.03.16

大纲

回顾

感知机

线性支持向量机

非线性支持向量机

多类支持向量机

从统计学的观点来看

- 机器学习的目的是得到映射： $x \mapsto y$
 - ✓ 类的先验概率： $p(y = i)$
 - ✓ 样本的先验概率： $p(x)$
 - ✓ 类条件概率（似然）： $p(x|y = i)$
 - ✓ 后验概率： $p(y = i|x)$

统计机器学习方法粗略分类

- 从概率框架的角度

- ✓ 生成式模型（Generative models）

- 估计 $p(\mathbf{x}|y = i)$ 和 $p(y = i)$ ，然后用贝叶斯定理求 $p(y = i|\mathbf{x})$

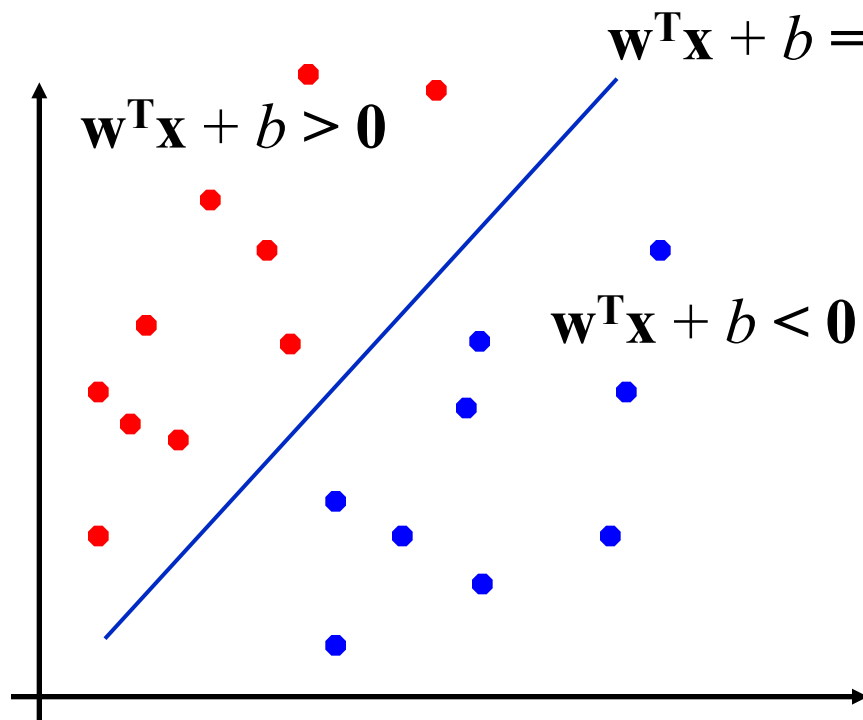
- ✓ 判别式模型（Discriminative models）

- 直接估计 $p(y = i|\mathbf{x})$
 - 判别函数（Discriminant function）：不假设概率模型，直接求一个把各类分开的边界

感知机：线性超平面

- 二分类

- ✓ 二分类问题可以看作是在特征空间上对类别进行划分的任务



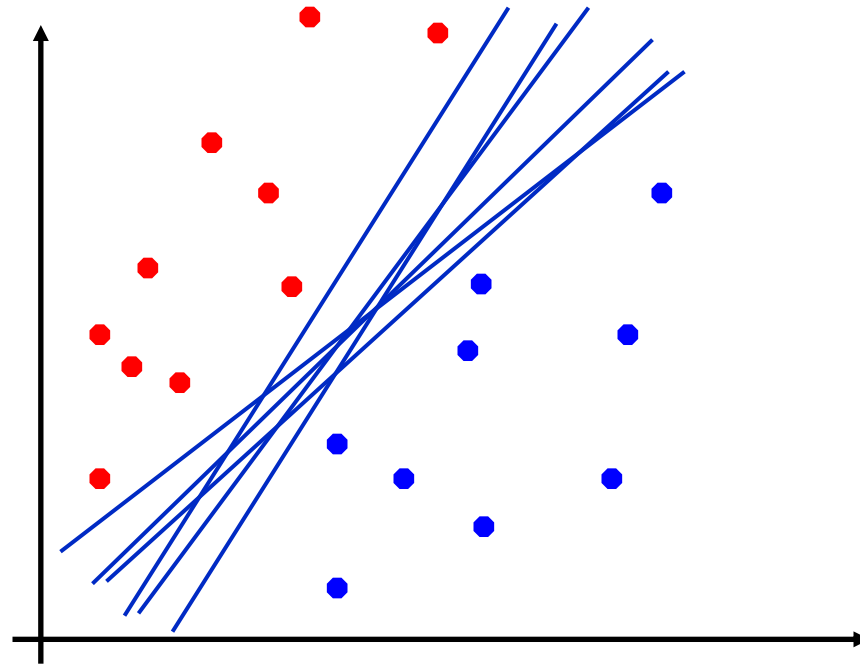
$\mathbf{w}^T \mathbf{x} + b = 0$ 划分超平面的线性方程

\mathbf{w} 为法向量，决定超平面的方向； b 为位移项，决定了超平面与原点之间的距离

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + b)$$

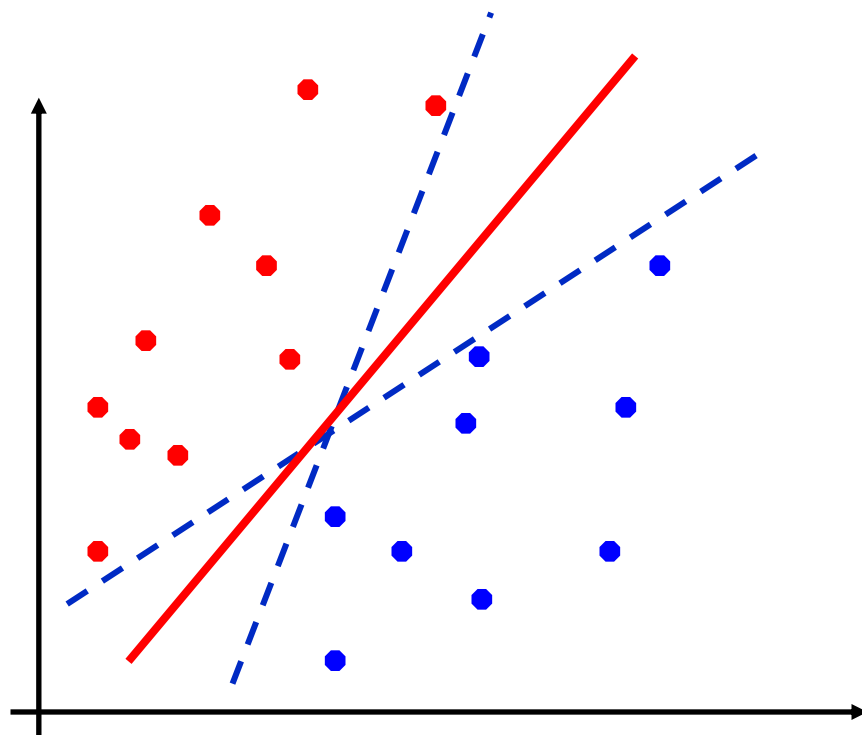
感知机：线性超平面

- 将训练样本分开的超平面很多，哪一个更好？



感知机：线性超平面

- 将训练样本分开的超平面很多，哪一个更好？

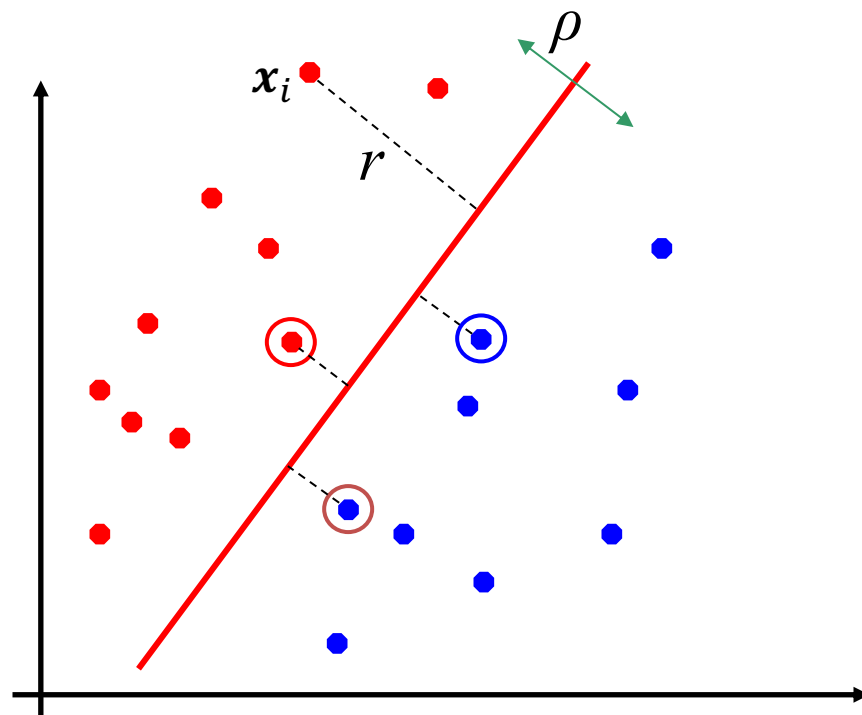


“正中间”的：鲁棒性最好，泛化能力最强

线性支持向量机

- 间隔与支持向量

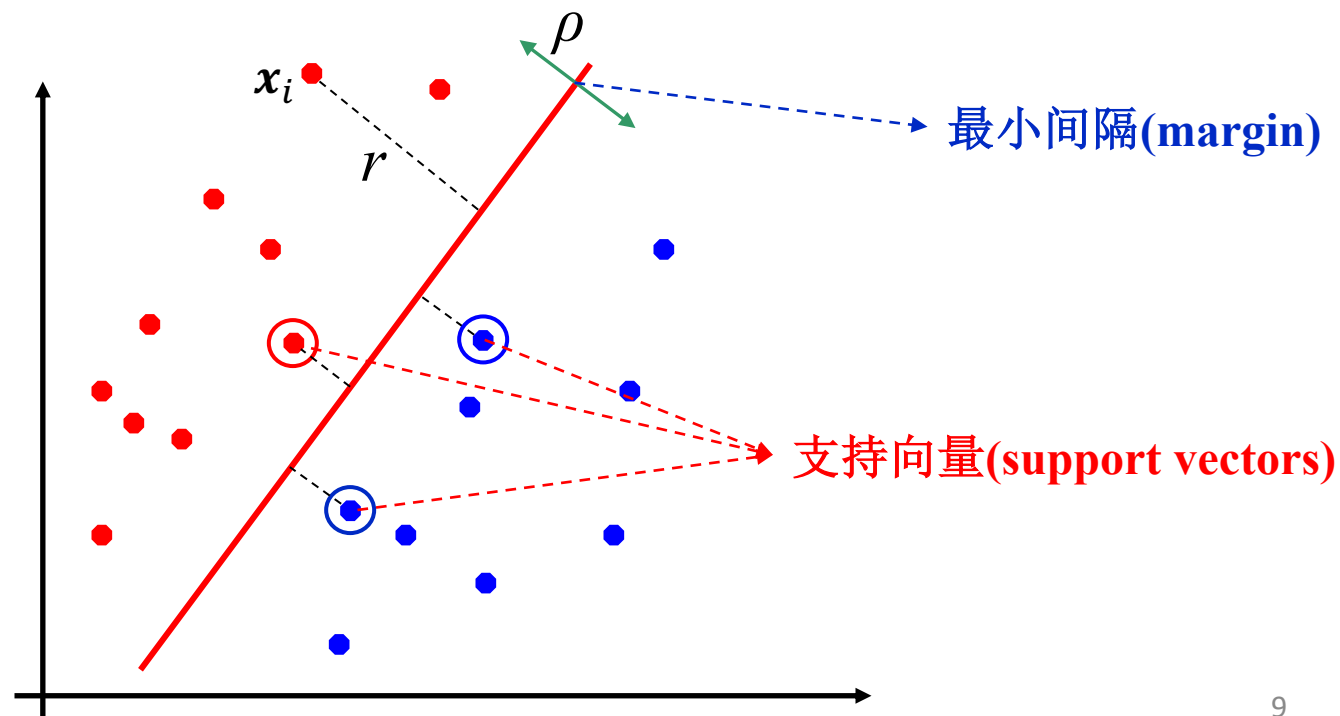
- ✓ 一个点（样例）对应的“间隔” margin 是其到分界超平面的垂直距离
- ✓ SVM最大化（所有训练样本的）最小间隔margin
- ✓ 有最小间隔的点称为支持向量(support vectors)
 - ❖ 所以叫支持向量机support vector machine



线性支持向量机

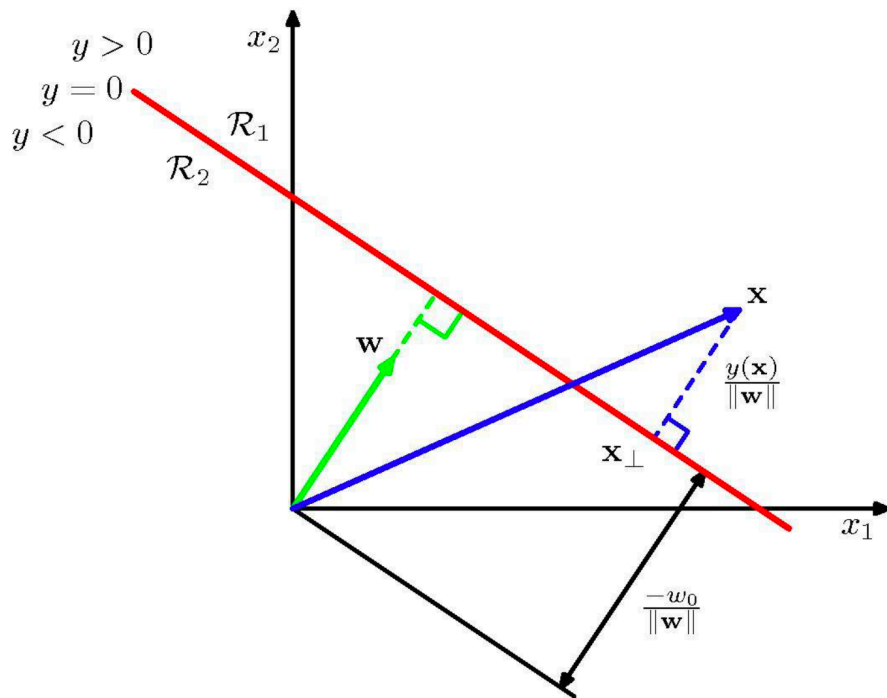
- 间隔与支持向量

- ✓ 一个点（样例）对应的“间隔” margin 是其到分界超平面的垂直距离
- ✓ SVM最大化（所有训练样本的）最小间隔margin
- ✓ 有最小间隔的点称为支持向量(support vectors)
 - ❖ 所以叫支持向量机support vector machine



线性支持向量机

- 几何示意图



- 分类超平面（红色）

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b = 0$$

- 绿色为其法向量(normal vector)

- \mathbf{x} 为任一点/样例，其到超平面的距离 r 为？

线性支持向量机

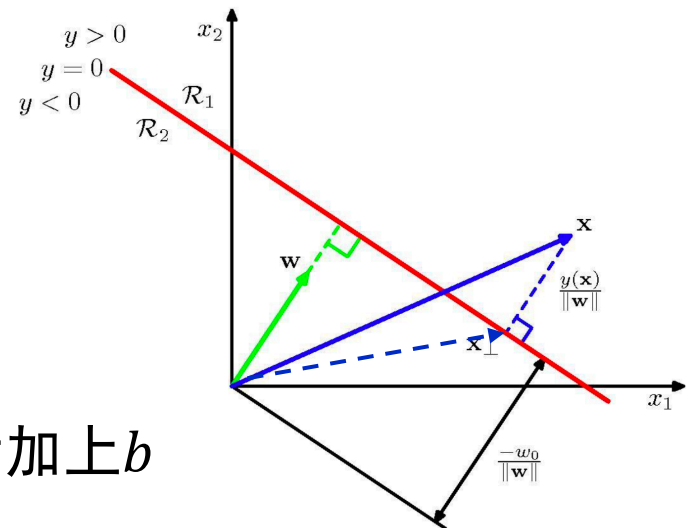
- 计算Margin

- ✓ x 的投影点为 x_{\perp} , $x - x_{\perp}$ 为距离向量

- 其方向与 w 相同, 为 $\frac{w}{\|w\|}$
- 其大小 r 可为0, 或正, 或负;
margin为其大小的绝对值

- ✓ $x = x_{\perp} + r \frac{w}{\|w\|}$, 两边同乘以 w^T , 然后加上 b

- $w^T x + b = w^T x_{\perp} + b + r \frac{w^T w}{\|w\|}$
- $f(x) = f(x_{\perp}) + r\|w\|$ 为什么?
- $r = \frac{f(x)}{\|w\|}$ 为什么?



$$f(x_{\perp}) = 0$$

线性支持向量机

- 计算Margin

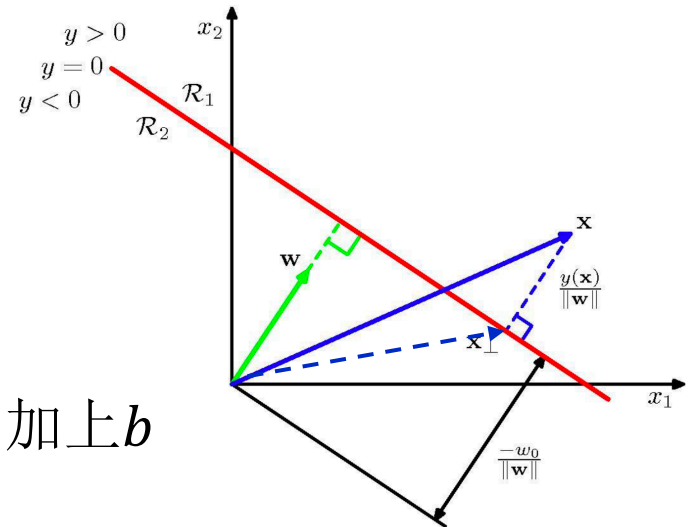
✓ \mathbf{x} 的投影点为 \mathbf{x}_\perp , $\mathbf{x} - \mathbf{x}_\perp$ 为距离向量

- 其方向与 \mathbf{w} 相同, 为 $\frac{\mathbf{w}}{\|\mathbf{w}\|}$
- 其大小 r 可为 0, 或正, 或负;
margin 为其大小的绝对值

✓ $\mathbf{x} = \mathbf{x}_\perp + r \frac{\mathbf{w}}{\|\mathbf{w}\|}$, 两边同乘以 \mathbf{w}^T , 然后加上 b

- $\mathbf{w}^T \mathbf{x} + b = \mathbf{w}^T \mathbf{x}_\perp + b + r \frac{\mathbf{w}^T \mathbf{w}}{\|\mathbf{w}\|}$
- $f(\mathbf{x}) = f(\mathbf{x}_\perp) + r \|\mathbf{w}\|$ 为什么?
- $r = \frac{f(\mathbf{x})}{\|\mathbf{w}\|}$ 为什么?

✓ \mathbf{x} 的 margin 是 $\frac{|f(\mathbf{x})|}{\|\mathbf{w}\|} = \frac{|\mathbf{w}^T \mathbf{x} + b|}{\|\mathbf{w}\|}$



线性支持向量机

- 分类与评价

- ✓ 怎样分类？

- $f(\mathbf{x}) > 0$ ----分为正类； $f(\mathbf{x}) < 0$ ----分为负类
 - 那么 $f(\mathbf{x}) = 0$ 怎么办？

- ✓ 对于任何一个样例，判断预测的对错？（ $y_i = \{-1, 1\}$ ）

- $y_i f(\mathbf{x}_i) > 0$ ----正确； $y_i f(\mathbf{x}_i) < 0$ ----错误
 - 如果我们假设能完全分开，并且 $|y_i| = 1$ ，那么

$$y_i f(\mathbf{x}_i) = |f(\mathbf{x}_i)|$$

线性支持向量机

- SVM的形式化描述

- ✓ SVM的问题是什么？

$$\begin{aligned} \operatorname{argmax}_{\mathbf{w}, b} \left(\min_i \left(\frac{|\mathbf{w}^T \mathbf{x}_i + b|}{\|\mathbf{w}\|} \right) \right) & \dashrightarrow r = \frac{|f(\mathbf{x})|}{\|\mathbf{w}\|} = \frac{|\mathbf{w}^T \mathbf{x} + b|}{\|\mathbf{w}\|} \\ \operatorname{argmax}_{\mathbf{w}, b} \left(\min_i \left(\frac{y_i (\mathbf{w}^T \mathbf{x}_i + b)}{\|\mathbf{w}\|} \right) \right) & \dashrightarrow y_i f(\mathbf{x}_i) = |f(\mathbf{x}_i)| \\ \operatorname{argmax}_{\mathbf{w}, b} \left(\frac{1}{\|\mathbf{w}\|} \min_i (y_i (\mathbf{w}^T \mathbf{x}_i + b)) \right) & \end{aligned}$$

- ✓ 非常难以优化，怎么办？

- 继续简化

线性支持向量机

- 换个角度看问题

- ✓ 到目前为止

- 对 \mathbf{w} 没有限制，要求最大化最小的间隔，难优化

- ✓ 判断对错：如果 $yf(\mathbf{x}) > 0$ 即正确

- 即 $y(\mathbf{w}^T \mathbf{x} + b) > 0$ ，只需要方向，完全不需要大小！
 - 如果 (\mathbf{w}, b) 变为 $(c\mathbf{w}, cb)$ ，预测和间隔会变吗？

线性支持向量机

- 换个角度看问题

- ✓ 选择一个合适 $c > 0$, 使得 $\|w\| = 1$

$$\begin{aligned} \max_{w, b} \quad & \min_{1 \leq i \leq n} y_i f(\mathbf{x}_i) \\ \text{s.t.} \quad & y_i f(\mathbf{x}_i) > 0, \quad 1 \leq i \leq n \\ & \mathbf{w}^T \mathbf{w} = 1. \end{aligned}$$

仍然难求解!

- ✓ 假设某个最优解为 (\mathbf{w}^*, b^*) , 选择 c 为

$$c = \min_{1 \leq i \leq n} y_i ((\mathbf{w}^*)^T \mathbf{x}_i + b^*)$$

- ✓ 那么 $\frac{1}{c}(\mathbf{w}^*, b^*)$ 也是一个最优解

$$\min_{1 \leq i \leq n} y_i \left(\left(\frac{1}{c} \mathbf{w}^* \right)^T \mathbf{x}_i + \frac{1}{c} b^* \right) = 1 > 0$$

线性支持向量机

- 换个角度看问题

✓ 我们限定 $\min_i (y_i(\mathbf{w}^T \mathbf{x}_i + b))$ 为1, 不改变最优目标值

○ 问题变为: 在限制 $\min_i (y_i(\mathbf{w}^T \mathbf{x}_i + b))$ 为1时, 最大化 $\frac{1}{\|\mathbf{w}\|}$

$$\begin{aligned} & \underset{\mathbf{w}, b}{\operatorname{argmin}} && \frac{1}{2} \mathbf{w}^T \mathbf{w} \\ \text{s. t. } &&& y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \forall i \end{aligned}$$

线性支持向量机

- 拉格朗日乘子法

- ✓ $L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^n a_i (y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1)$

- Subject to $a_i \geq 0$

- ✓ 证明最优化的必要条件

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{0}$$

→

$$\mathbf{w} = \sum_{i=1}^n a_i y_i \mathbf{x}_i$$

$$\frac{\partial L}{\partial b} = 0$$

→

$$0 = \sum_{i=1}^n a_i y_i$$

- ✓ 在此两条件下，将两个等式代入回 L

$$\tilde{L}(\mathbf{a}) = \sum_{i=1}^n a_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n a_i a_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \quad \text{内积}$$

线性支持向量机

- Karush-Kuhn-Tucker条件, KKT

$$\mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i$$

$$\sum_{i=1}^n \alpha_i y_i = 0$$

$$\boxed{\alpha_i (y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1) = 0 \quad i = 1, 2, \dots, n} \quad \text{互补松弛性质}$$
$$\alpha_i \geq 0 \quad i = 1, 2, \dots, n$$

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \quad i = 1, 2, \dots, n$$

- ✓ 一般而言, KKT条件不是充分必要的; 但在原始SVM问题里, 这些条件对于确定最优解即是充分又必要的

线性支持向量机

- SVM的对偶形式

- ✓ 在原来的空间（即输入空间）中

- 变量是 \mathbf{x}_i ，称为SVM的原始形式（primal form）

- ✓ 现在的问题里面

- 变量是 a_i ，即拉格朗日乘子，称为对偶空间（dual space）

- 对偶空间完成优化后，得到最优的 \mathbf{a} ，可以得到原始空间中的最优解 \mathbf{w}

- ✓ SVM的对偶形式（dual form）

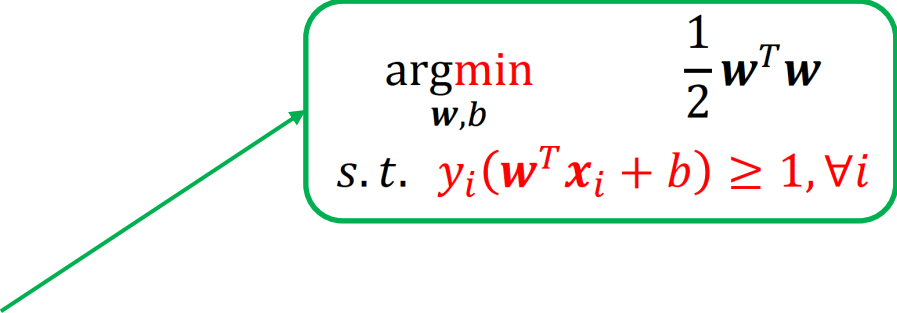
$$\begin{aligned} \arg\max_{\mathbf{a}} \quad & \sum_{i=1}^n a_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n a_i a_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{s. t.} \quad & a_i \geq 0 \\ & \sum_{i=1}^n a_i y_i = 0 \longrightarrow \frac{\partial L}{\partial b} = 0 \rightarrow 0 = \sum_{i=1}^n a_i y_i \end{aligned}$$

线性支持向量机

- 剩下的问题

- ✓ 如何最优化？

- 对偶空间中
- 原始空间中


$$\begin{aligned} \underset{\mathbf{w}, b}{\operatorname{argmin}} \quad & \frac{1}{2} \mathbf{w}^T \mathbf{w} \\ \text{s.t.} \quad & y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \forall i \end{aligned}$$

- ✓ 如果能允许少数点 $y_i f(\mathbf{x}_i) < 1$

- 如果允许一个点 $y_i f(\mathbf{x}_i) < 1$ ，但是大幅度增加margin呢？

- ✓ 如果不是线性可分的（linearly separable），但是可以用非线性的（non-linearly separable）边界分开？

- ✓ 如果不是两个类，而是多个呢？

线性支持向量机

- Soft margin

- ✓ 可以允许少数点margin比1小

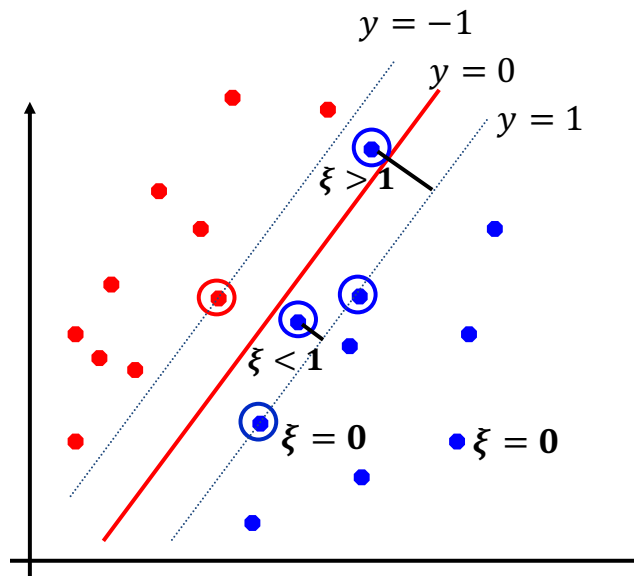
- 但是犯错误是有惩罚的，否则？

- $y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \rightarrow y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i$

- ξ_i : 松弛变量 (slack variable)，即允许犯的错误

- $\xi_i \geq 0$

- 右图 $\xi = 0$, > 1 各自
代表什么？



线性支持向量机

- 如何惩罚？

- ✓ 原始空间 (Primal space)

$$\begin{aligned} \operatorname{argmin}_{\mathbf{w}, b, \xi} \quad & \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^n \xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \end{aligned}$$

- ✓ $C > 0$: 正则化参数 (regularization parameter)

- ξ_i ---- 代价，我们要最小化代价函数（总代价）
- $\frac{1}{2} \mathbf{w}^T \mathbf{w}$ ---- 正则项（regularization term），对分类器进行限制，使得模型复杂度不至于太高（另一个角度，还是最大化间隔）
- 那么，如何确定 C 的值？

线性支持向量机

- Soft margin的对偶形式

$$\begin{aligned} \operatorname{argmax}_a \quad & \sum_{i=1}^n a_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n a_i a_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{s.t.} \quad & \mathbf{C} \geq a_i \geq 0 \\ & \sum_{i=1}^n a_i y_i = 0 \end{aligned}$$

✓ 对偶形式仅依赖于内积！

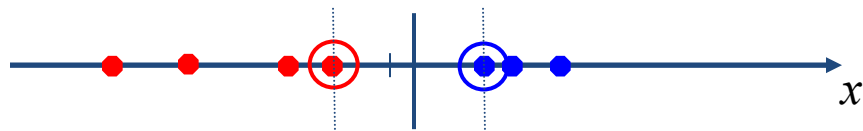
线性支持向量机

- 总结

- ✓ 分类器是一个分离超平面（separating hyperplane）
- ✓ 最重要的训练样本是“支持向量”，它们定义了超平面，而其他训练样本被忽略了。二次优化算法可以识别哪些样本是具有非零朗格朗日乘子 a_i 的支持向量
- ✓ 对偶问题中，训练样本只以内积形式出现。

非线性支持向量机

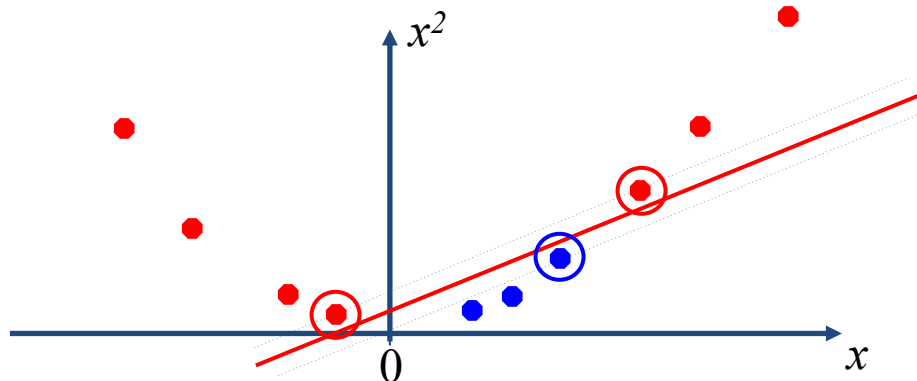
- ✓ 数据集是线性可分，效果会很好



- ✓ 数据集太困难了，怎么办？



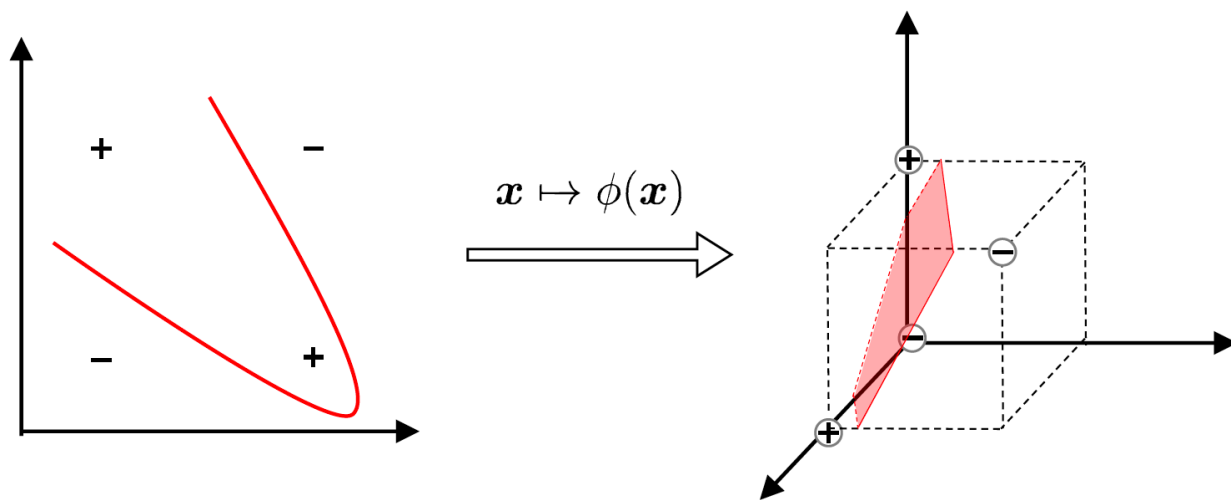
- ✓ 如果把数据映射到高维空间里？



非线性支持向量机

- 特征空间映射

- ✓ 将样本从原始空间映射到一个更高维的特征空间，使样本在这个特征空间内线性可分



- ✓ 如果原始空间是有限维(属性数有限)，那么一定存在一个高维特征空间使样本可分

非线性支持向量机

- 内积：线性和非线性的联系
 - ✓ 线性和非线性有时候紧密联系在一起----通过内积
 - ✓ $\mathbf{x} = (x_1, x_2), \mathbf{z} = (z_1, z_2)$

$$\begin{aligned} K(\mathbf{x}, \mathbf{z}) &= (1 + \mathbf{x}^T \mathbf{z})^2 = (1 + x_1 z_1 + x_2 z_2)^2 \\ &= 1 + 2x_1 z_1 + 2x_2 z_2 + x_1^2 z_1^2 + x_2^2 z_2^2 + 2x_1 z_1 x_2 z_2 \\ &= \begin{pmatrix} 1 \\ \sqrt{2}x_1 \\ \sqrt{2}x_2 \\ x_1^2 \\ x_2^2 \\ \sqrt{2}x_1 x_2 \end{pmatrix}^T \begin{pmatrix} 1 \\ \sqrt{2}z_1 \\ \sqrt{2}z_2 \\ z_1^2 \\ z_2^2 \\ \sqrt{2}z_1 z_2 \end{pmatrix} \end{aligned}$$

非线性支持向量机

- Kernel trick

- ✓ 两个向量 $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ ，一个非线性函数 $K(\mathbf{x}, \mathbf{y})$

- ✓ 对于满足某些条件的函数 K ，一定存在一个映射 (mapping)
 $\phi: \mathbb{R}^d \rightarrow \Phi$ ，使得对任意 \mathbf{x}, \mathbf{y}

$$K(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y})$$

- 非线性函数 K 表示两个向量的相似程度
 - 其等价于 Φ 里面的内积

- ✓ Φ : 特征空间 (feature space)

- 可以是有限维的空间，但也可以是无穷维的空间 (infinite dimensional Hilbert space)

非线性支持向量机

- 什么样的限制条件

- ✓ 必须存在特征映射（feature mapping），才可以将非线性函数表示为特征空间中的内积

- ✓ **Mercer's condition**（Mercer条件，是充分必要的）：

- 对任何满足 $\int g^2(\mathbf{u})d\mathbf{u} < \infty$ 的非零函数（平方可积函数），对称函数 K 满足条件：

$$\iint K(\mathbf{u}, \mathbf{v})g(\mathbf{u})g(\mathbf{v}) d\mathbf{u}d\mathbf{v} \geq 0$$

- ✓ 另一种等价形式：

- 对任何一个样本集合 $\{x_1, \dots, x_n\}, x_i \in \mathbb{R}^d$ ，如果矩阵 $K = [K_{ij}]_{ij}$ （矩阵的第 i 行、第 j 列元素 $K_{ij} = K(x_i, x_j)$ ）总是半正定的，那么函数 K 满足 Mercer 条件

非线性支持向量机

- 核支持向量机Kernel SVM

- ✓ 核函数 (kernel function) : K

- ✓ 对偶形式:

$$\operatorname{argmax}_a \sum_{i=1}^n a_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n a_i a_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

- ✓ 分类边界: $\mathbf{w} = \sum_{i=1}^n a_i y_i \phi(\mathbf{x}_i)$

- ✓ 怎样预测:

$$\mathbf{w}^T \phi(\mathbf{x}) = \phi(\mathbf{x})^T \left(\sum_{i=1}^n a_i y_i \phi(\mathbf{x}_i) \right) = \sum_{i=1}^n a_i y_i K(\mathbf{x}, \mathbf{x}_i)$$

- 线性: $\mathbf{w} = \sum_{i=1}^n a_i y_i \mathbf{x}_i$, $\mathbf{w}^T \mathbf{x}$ 的计算量为 $O(d)$
- 非线性 (核) 方法测试所需时间为?
- 假设计算 K 的时间为 $O(d)$, 是 $O(nd)$ 吗?

非线性支持向量机

- 非线性核

- ✓ 线性核 (linear kernel) , dot-product kernel:

$$K(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{y}$$

- ✓ 非线性核 (non-linear kernel)

- RBF (radial basis function)/高斯 (Gaussian) 核

$$K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma \|\mathbf{x} - \mathbf{y}\|^2)$$

- 多项式核

$$K(\mathbf{x}, \mathbf{y}) = (\gamma \mathbf{x}^T \mathbf{y} + c)^d$$

-

非线性支持向量机

- 非线性核

✓ $\mathbf{x} = (x_1, x_2), \mathbf{z} = (z_1, z_2)$

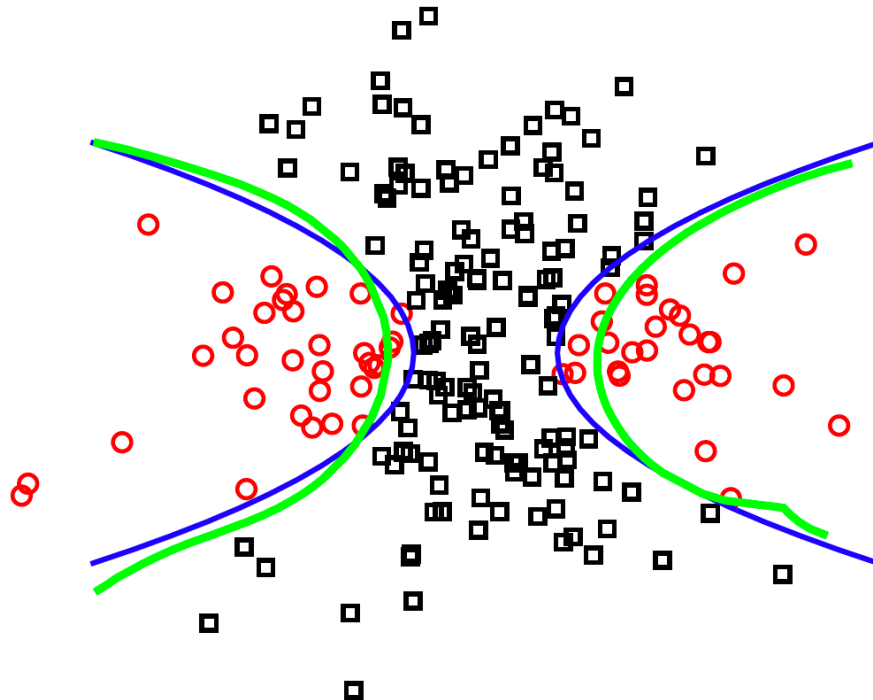
$\gamma = 1, c = 1, d = 2$
多项式kernel的一个特例

$$\begin{aligned} K(\mathbf{x}, \mathbf{z}) &= (1 + \mathbf{x}^T \mathbf{z})^2 = (1 + x_1 z_1 + x_2 z_2)^2 \\ &= 1 + 2x_1 z_1 + 2x_2 z_2 + x_1^2 z_1^2 + x_2^2 z_2^2 + 2x_1 z_1 x_2 z_2 \\ &= \begin{pmatrix} 1 \\ \sqrt{2}x_1 \\ \sqrt{2}x_2 \\ x_1^2 \\ x_2^2 \\ \sqrt{2}x_1 x_2 \end{pmatrix}^T \begin{pmatrix} 1 \\ \sqrt{2}z_1 \\ \sqrt{2}z_2 \\ z_1^2 \\ z_2^2 \\ \sqrt{2}z_1 z_2 \end{pmatrix} \end{aligned}$$

2维空间被映射到6维空间！

非线性支持向量机

- 非线性核的例子（RBF）
 - Ground Truth boundary 真实分类边界
 - 使用RBF核的SVM得到的分类边界



非线性支持向量机

- 超参数

- ✓ 如何决定 C 、 γ 、...

- 必须给定这些参数的值，才能进行SVM学习，SVM本身不能学习这些参数
 - 称为超参数（hyper-parameter）
 - 对SVM的结果有极大的影响

- ✓ 用交叉验证在训练集上学习

- 在训练集上得到不同参数的交叉验证准确率
 - 选择准确率最高的超参数的数值

多类支持向量机

- 多类 (Multiclass)

- ✓ 思路：转化为2类问题

- ✓ 1 - vs. -1 (one versus one) :

- C 个类 $\{1, 2, \dots, C\}$

- 设计 $\binom{C}{2}$ 个分类器，用 i 和 j 两类 ($i > j$) 的训练数据进行学习

- 一共 $C(C-1)/2$ 个，其中每个类出现 $C-1$ 次

- 对测试样本 \mathbf{x} ，一共会得到 $C(C-1)/2$ 个结果，然后投票

- 对每个分类器 f_i 采用其二值输出，即 $\text{sign}(f_i(\mathbf{x}))$

| | 1 | 2 | 3 |
|---|---|---|---|
| 1 | | | |
| 2 | 1 | | |
| 3 | 1 | 3 | |

多类支持向量机

- 多类 (Multiclass)

- ✓ 1 - vs. -all (或 1 - vs. -rest) :

- 设计 C 个分类器, 第 i 个分类器用类 i 做正类, 把其他所有 $C - 1$ 个类别的数据合并在一起做负类

- ❖ 和交叉验证的步骤有些类似

- ❖ 每个新的分类器 f_i 采用其实值输出, 即 $f_i(\mathbf{x})$

- $f_i(\mathbf{x})$ 的实数输出可以看成是属于第 i 类的 “信心” (confidence)
 - 最终选择信心最高的那个类为输出

$$\operatorname{argmax}_i f_i(\mathbf{x})$$

多类支持向量机

- 多类（Multiclass）
 - ✓ 直接解决多类问题
 - Crammer-Singer方法
 - <http://jmlr.org/papers/v2/crammer01a.html>
 - ✓ DAGSVM
 - <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/dagsvm.pdf>
 - ✓ ECOC
 - <https://arxiv.org/pdf/cs/9501101.pdf>

支持向量机的实现

- ✓ SVM Website: <http://www.kernel-machines.org/>
- ✓ 代表性实现
 - **LIBSVM**: 非常有效且出名的SVM实现，实现了multi-class classifications, nu-SVM, one-class SVM等，而且有很多java、python等接口
 - **SVM-light**: 简单但是性能不如LIBSVM，仅支持二分类，并且只用C语言实现
 - **SVM-torch**: C语言实现

支持向量机的启发

- 从SVM的介绍学到的思想？

1. 确定问题，对问题有充分的认识（实践、理论）

2. 好的思路、想法idea（如margin）

- 从理论（概率、统计？）中来

- 从实践（已有线性分类器的缺点，如感知机）中来

3. 形式化

- 用精确的数学形式表达出来

- 如果不能精确描述，或说明你的idea有问题

- 简化，开始时避免复杂、模糊的想法：限制条件（如，线性可分），从较小范围开始（如，2类）

4. 数学基础和研究

- 用到的几何、凸优化、拉格朗日乘子法、Hilbert空间。。。

- 经典的相关数学背景要熟悉：至少知道到哪里查

支持向量机的启发

- 简化：一种可靠的思路
 - ✓ 问题（特别是数学问题）难以解决时，尽量简化
 - 问题的表述，如果难以形式化，可以将问题简化
 - 简化后的问题可以去除很多复杂的考虑，但是原问题的核心要保持
 - 如SVM从二类、线性、可分的情况开始
 - ✓ 有时可以通过换思路的方法等价简化
 - 如SVM限定 $\min_i (y_i(\mathbf{w}^T \mathbf{x}_i + b))$ 为1
 - 也可以对原问题做不重复的修改以使简化成为可能

支持向量机的延伸

- 深度学习时代的SVM?

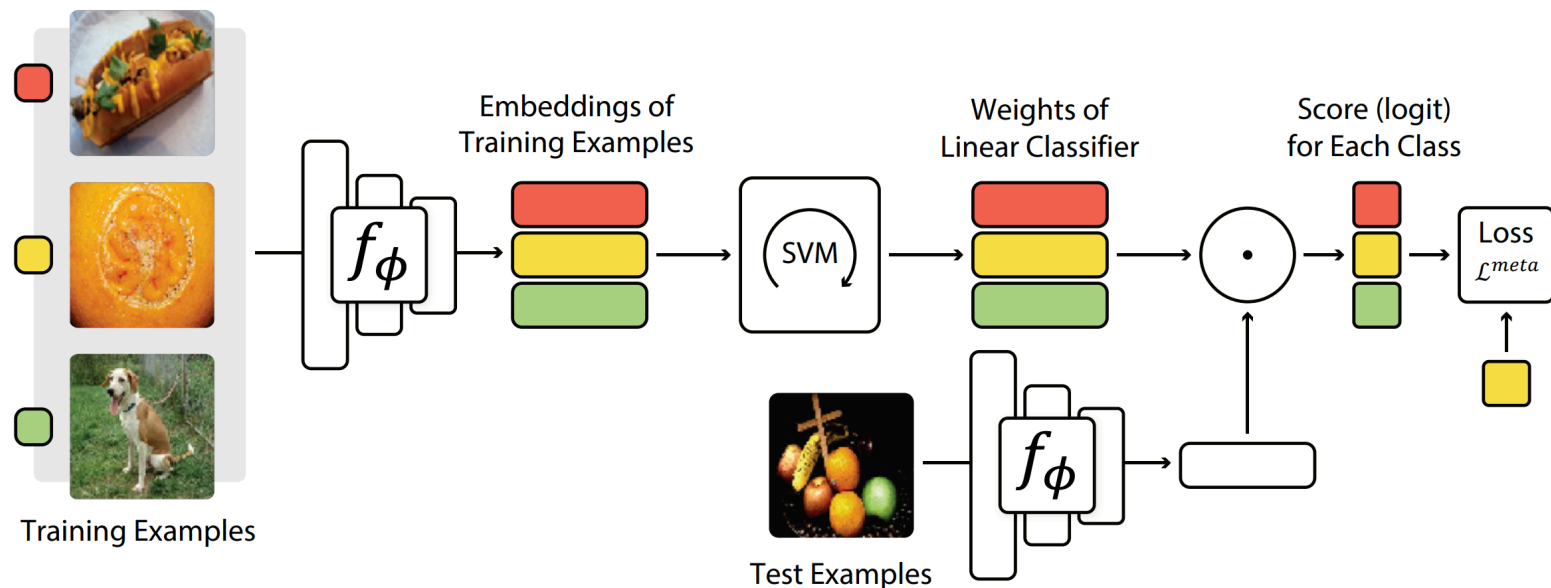


Figure 1. **Overview of our approach.** Schematic illustration of our method MetaOptNet on an 1-shot 3-way classification task. The meta-training objective is to learn the parameters ϕ of a feature embedding model f_ϕ that generalizes well across tasks when used with regularized linear classifiers (*e.g.*, SVMs). A task is a tuple of a few-shot training set and a test set (see Section 3 for details).

Lee K, Maji S, Ravichandran A, et al. Meta-learning with differentiable convex optimization. CVPR 2019: 10657-10665.

谢谢！