模式识别

支持向量机SVM

吴建鑫
南京大学计算机系 & 人工智能学院，2020

统计学习方法的粗略分类

- Statistical learning methods
  - $p(y=i)$，$p(y=i|\boldsymbol{x})$，$p(\boldsymbol{x}|y=i)$，$p(\boldsymbol{x})$
  - 还记得其含义吗？
  - Generative (probabilistic) models: 估计$p(\boldsymbol{x}|y=i)$和$p(y)$
    - 然后用贝叶斯定理求$p(y=i|\boldsymbol{x})$
    - 生成模型（下一章）
  - Discriminative (probabilistic) models:直接估计$p(y=i|\boldsymbol{x})$
    - 判别模型（下一章）
  - Discriminant function：直接求一个把各类分开的边界
    - 不假设概率模型，如FLD（上一章），SVM（本章）
    - 更多阅读PRML1.5.4

目标

- ✓ 理解并掌握SVM中主要思想的含义、描述、数学表述
- ✓ 如何将一个好的idea形式化
- ✓ 能实际应用SVM

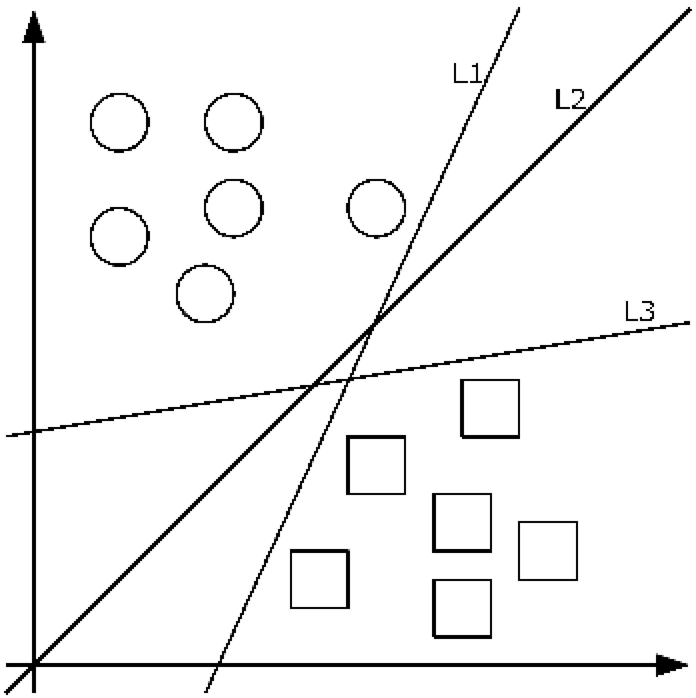
- ✓ 提高目标
  - 理解相关推导，能在有文献帮助下自主完成推导
  - 进一步能通过独立阅读、了解统计学习

SVM

Support vector machine支持向量机

注意SVM的形式化过程，和简化的思路

large margin（最大边际？）



用线性边界分开2类

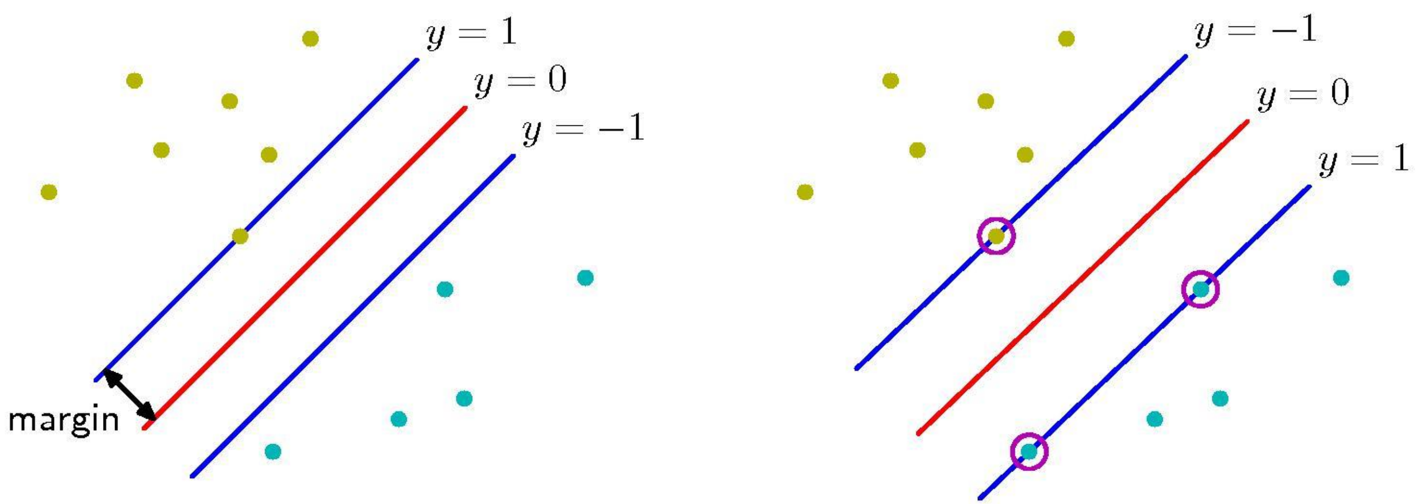
- 正类positive class，$y_i = 1$
- 负类negative class，$y_i = -1$

- 可以有很多边界L1，L2，L3，…，在训练集上都100%正确（假设能完全分开）
- 哪个最好？

http://zh.wikipedia.org/wiki/File:Classifier.svg

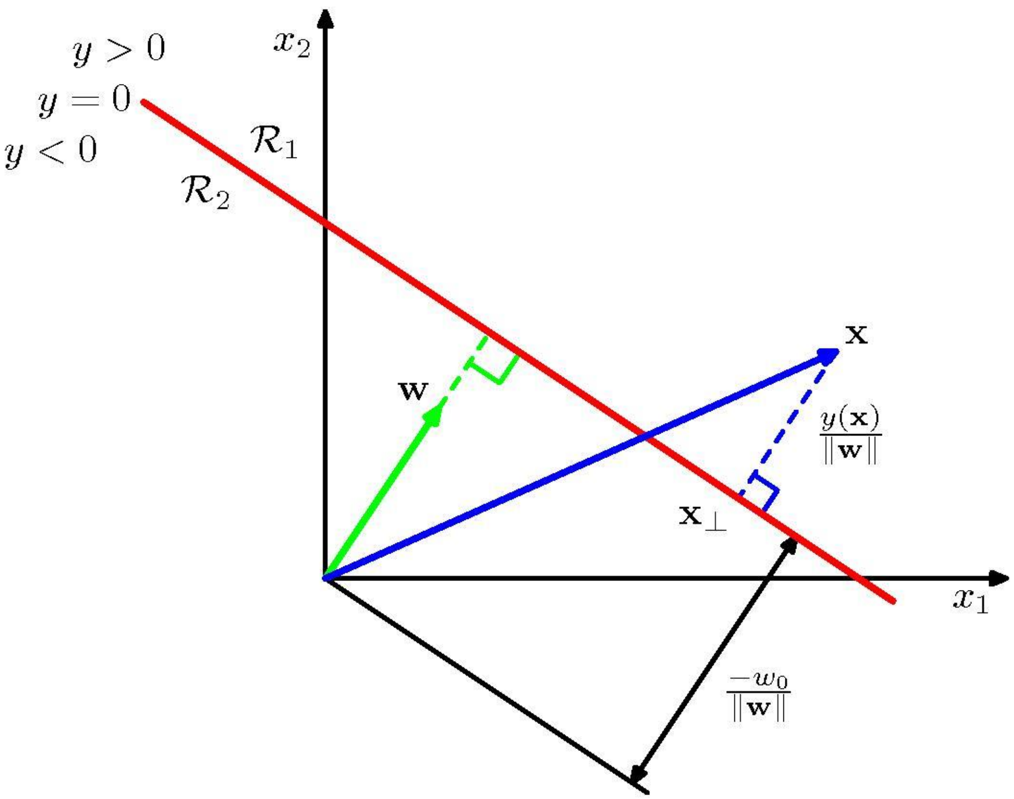
margin

图片来自PRML第7章



- 一个点（样例）的边际margin是其到分界超平面separating hyperplane的垂直距离
- SVM最大化（所有训练样本的）最小边际
- 有最小边际的点称为支持向量(support vectors)
  - 所以叫支持向量机support vector machine

几何geometry示意图



图片来自PRML第4章

- 分类超平面 $f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b = 0$
  - 红色
  - 绿色为其法向量 normal vector
- x为任一点/样例
  - 其到超平面的距离为?

计算margin

- 投影点为$\boldsymbol{x}_\perp$， $\boldsymbol{x}-\boldsymbol{x}_\perp$为距离向量
  - 其方向与$\boldsymbol{w}$相同，为$\boldsymbol{w}/\|\boldsymbol{w}\|$
  - 其大小r可为0，或正，或负；margin为其大小的绝对值
- $\boldsymbol{x}=\boldsymbol{x}_\perp+r\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$，两边同乘以$\boldsymbol{w}^T$，然后加上$b$
  - $\boldsymbol{w}^T\boldsymbol{x}+b=\boldsymbol{w}^T\boldsymbol{x}_\perp+b+r\frac{\boldsymbol{w}^T\boldsymbol{w}}{\|\boldsymbol{w}\|}$
  - $f(\boldsymbol{x})=f(\boldsymbol{x}_\perp)+r\|\boldsymbol{w}\|$ 为什么？
  - $r=\frac{f(\boldsymbol{x})}{\|\boldsymbol{w}\|}$ 为什么?
  - $\boldsymbol{x}$的margin是$\frac{|f(\boldsymbol{x})|}{\|\boldsymbol{w}\|}=\frac{|\boldsymbol{w}^T\boldsymbol{x}+b|}{\|\boldsymbol{w}\|}$

分类、评价

✓ 怎么样分类？

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

✓ 对于任何一个样例，怎么知道预测的对错？

- $y_i f(\boldsymbol{x}_i)>0$ 正确 $y_i f(\boldsymbol{x}_i) < 0$ 错误
- 即，因为我们假设能完全分开，所以

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

SVM的形式化描述

- 那么，SVM问题是什么？

$$\operatorname*{argmax}_{\boldsymbol{w},b}\left(\min_i\left(\frac{|\boldsymbol{w}^T\boldsymbol{x}_i+b|}{\|\boldsymbol{w}\|}\right)\right)$$

$$\operatorname*{argmax}_{\boldsymbol{w},b}\left(\min_i\left(\frac{y_i(\boldsymbol{w}^T\boldsymbol{x}_i+b)}{\|\boldsymbol{w}\|}\right)\right)$$

$$\operatorname*{argmax}_{\boldsymbol{w},b}\left(\frac{1}{\|\boldsymbol{w}\|}\min_i\left(y_i(\boldsymbol{w}^T\boldsymbol{x}_i+b)\right)\right)$$

- 非常难以优化，怎么办？
  - 继续简化

换个角度看问题

- 到目前为止
  - 对$\boldsymbol{w}$没有限制，要求最大化最小的边际，难优化
- 判断对错：如果$yf(\boldsymbol{x})>0$ 即正确
  - 即$y(\boldsymbol{w}^T\boldsymbol{x}+b)>0$，只需要方向，完全不需要大小！
  - 如果$(\boldsymbol{w},b)$变为$(\lambda\boldsymbol{w},\lambda b)$，预测和边际会变吗？
- 那么我们可以限定$\min_i\left(y_i(\boldsymbol{w}^T\boldsymbol{x}_i+b)\right)$为1
  - 问题变为：在限制$\min_i\left(y_i(\boldsymbol{w}^T\boldsymbol{x}_i+b)\right)$为1时，最大化$\frac{1}{\|\boldsymbol{w}\|}$

$$\operatorname*{argmin}_{\boldsymbol{w},b} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}$$

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

拉格朗日乘子法, again

- ✓ $L(\boldsymbol{w}, b, \boldsymbol{a}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} - \sum_{i=1}^{n} a_i(y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) - 1)$
  - Subject to $a_i \geq 0$
- ✓ 作业：证明最优化的必要条件
- ✓ $\frac{\partial L}{\partial \boldsymbol{w}} = \boldsymbol{0}$ ➔ $\boldsymbol{w} = \sum_{i=1}^{n} a_i\, y_i \boldsymbol{x}_i$
- ✓ $\frac{\partial L}{\partial b} = 0$ ➔ $0 = \sum_{i=1}^{n} a_i\, y_i$
- ✓ 在此两条件下，将两个等式代入回L

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

SVM的对偶形式

- 在原来的空间（输入空间，input space）中
  - 变量是$\boldsymbol{x}_i$，称为SVM的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$\boldsymbol{a}$，可以得到原始空间中的最优解$\boldsymbol{w}$
- SVM的对偶形式dual form

$$\underset{\boldsymbol{a}}{\arg\max} \sum_{i=1}^{n} a_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j$$

$$s.t. \quad a_i \geq 0$$

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

剩下的问题

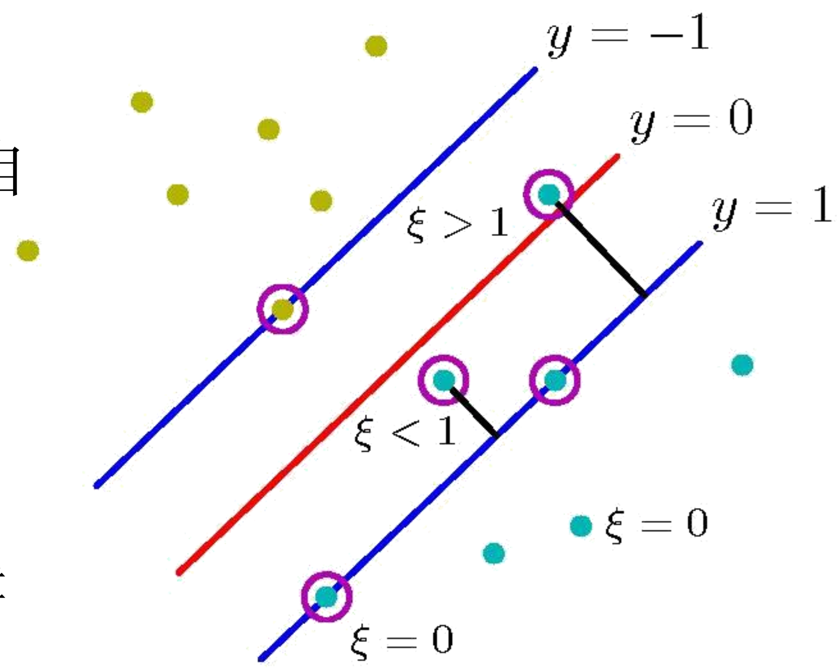
- ✓ 如何最优化？
  - 对偶空间中
  - 原始空间中
- ✓ 如果能允许少数点$y_i f(\boldsymbol{x}_i) < 1$
  - 如果允许一个点$y_i f(\boldsymbol{x}_i) < 1$，但是大幅度增加margin呢？
- ✓ 如果不是线性可分的linearly separable，但是可以用非线性的边界分开non-linearly separable?
- ✓ 如果不是两个类，而是多个呢？

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？
- $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1$ ➔ $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \xi_i$
- ξ_i：松弛变量slack variable，即允许犯的错误
- $\xi_i \geq 0$

- =0，（0　1），=1，>1各自代表什么？



图片来自PRML第7章

如何惩罚？

- ✓ Primal space

$$\underset{\boldsymbol{w},b,\xi}{\operatorname{argmin}} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{n}\xi_i$$

$$s.t. \quad y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0$$

- ✓ $C > 0$：正则化参数regularization parameter
  - ξ_i—代价， 我们要最小化代价函数（总代价）
  - $\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}$—正则项regularization term，对分类器进行限制，使复杂度不至于太高（另一个角度，还是最大化边际）
  - 那么，怎么确定C的值？

Soft margin的对偶形式

- ✓ 自主阅读PRML

$$\underset{\boldsymbol{a}}{\operatorname{argmax}} \sum_{i=1}^{n} a_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j$$

$$s.t. \qquad C \geq a_i \geq 0$$

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

- ✓ 对偶形式仅依赖于内积！

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
- $\boldsymbol{x} = (x_1, x_2), \boldsymbol{z} = (z_1, z_2)$
- $K(\boldsymbol{x}, \boldsymbol{z}) = (1 + \boldsymbol{x}^T \boldsymbol{z})^2 = (1 + x_1 z_1 + x_2 z_2)^2$

$$= 1 + 2x_1 z_1 + 2x_2 z_2 + x_1^2 z_1^2 + x_2^2 z_2^2 + 2x_1 z_1 x_2 z_2$$

$$= \begin{pmatrix} 1 \\ \sqrt{2}x_1 \\ \sqrt{2}x_2 \\ x_1^2 \\ x_2^2 \\ \sqrt{2}x_1 x_2 \end{pmatrix}^T \begin{pmatrix} 1 \\ \sqrt{2}z_1 \\ \sqrt{2}z_2 \\ z_1^2 \\ z_2^2 \\ \sqrt{2}z_1 z_2 \end{pmatrix}$$

Kernel trick

- ✓ 两个向量$\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$，一个非线性函数$K(\boldsymbol{x}, \boldsymbol{y})$
- ✓ 对于满足某些条件的函数K，一定存在一个映射（mapping）$\phi: \mathbb{R}^d \mapsto \Phi$，使得对任意的$\boldsymbol{x}, \boldsymbol{y}$

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

  - 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

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$\int g^2(\boldsymbol{u})d\boldsymbol{u} < \infty$的非零函数，对称函数$K$满足条件： $\iint g(\boldsymbol{u})K(\boldsymbol{u},\boldsymbol{v})g(\boldsymbol{v})d\boldsymbol{u}d\boldsymbol{v} \geq 0$
- ✓ 看上去眼熟？另一种等价形式：对**任何**一个样本集合$\{\boldsymbol{x}_1,\dots,\boldsymbol{x}_n\}, \boldsymbol{x}_i \in \mathbb{R}^d$，如果矩阵$K = \left[K_{ij}\right]_{i,j}$（矩阵的第$i$行、第$j$列元素$K_{ij} = K(\boldsymbol{x}_i,\boldsymbol{x}_j)$）总是半正定的，那么函数$K$满足Mercer条件
- ✓ 如何判定是否满足？有几种方法？

核支持向量机Kernel SVM

- 核函数kernel function: K
- 对偶形式：
  $$\underset{\boldsymbol{a}}{\operatorname{argmax}} \sum_{i=1}^{n} a_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j)$$
- 分类边界： $\boldsymbol{w} = \sum_{i=1}^{n} a_i\, y_i \phi(\boldsymbol{x}_i)$
- 怎样预测：$\boldsymbol{w}^T\phi(\boldsymbol{x}) = \phi(\boldsymbol{x})^T\left(\sum_{i=1}^{n} a_i\, y_i \phi(\boldsymbol{x}_i)\right) = \sum_{i=1}^{n} a_i\, y_i K(\boldsymbol{x}, \boldsymbol{x}_i)$
  - 线性： $\boldsymbol{w} = \sum_{i=1}^{n} a_i\, y_i \boldsymbol{x}_i$， $\boldsymbol{w}^T\boldsymbol{x}$ 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？

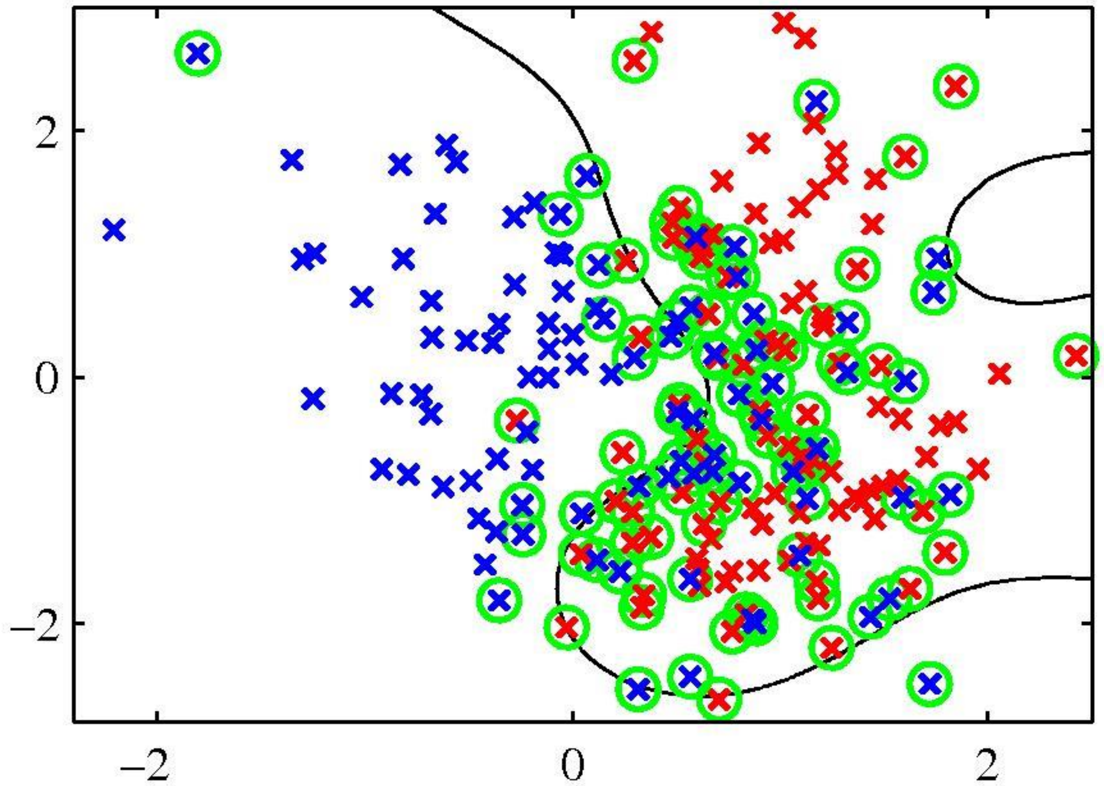
Complementary Slackness

- ✓ 对所有i，KKT条件包括$(C - a_i)\xi_i = 0$
  - 情况1：$C > a_i > 0$，$\xi_i = 0$，在特征空间中边际为1的两个超平面上
  - 情况2：$a_i = C$，对ξ_i没有限制
    - 可以在超平面上、介于两个超平面之间、或以外（即分类错误）
  - 情况3：$a_i = 0$，在预测时不需要计算
- ✓ 这代表什么？
  - 复杂度由$a_i > 0$的个数，而非样本的总数目来决定
  - $\boldsymbol{a}$是稀疏的
  - 在soft margin SVM中，称$a_i > 0$对应的$\boldsymbol{x}_i$为支持向量

非线性核

- ✓ 线性核linear kernel, dot-product kernel:
  $K(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{x}^T\boldsymbol{y}$
- ✓ 非线性核non-linear kernel
  - RBF(radial basis function)、高斯(Gaussian)核:
    $K(\boldsymbol{x},\boldsymbol{y}) = \exp(-\gamma\|\boldsymbol{x}-\boldsymbol{y}\|^2)$
  - 多项式核: $K(\boldsymbol{x},\boldsymbol{y}) = (\gamma\boldsymbol{x}^T\boldsymbol{y}+c)^d$
  - …
  - 进一步阅读:更多核函数
    http://www.zhizhihu.com/html/y2010/2292.html

非线性核的例子(RBF)



图片来自PRML第7章

超参数

- ✓ 如何决定C、γ、$\cdots$
  - 必须给定这些参数parameter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来学习
  - 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Multiclass(1)

- 思路：转化为2类问题
- 1-vs-1 （one versus one）：C个类$\{1,2,\ldots,C\}$
  - 设计$\binom{C}{2}$个分类器：用i和$j\,(i>j)$两类的训练数据学习
  - 一共$C(C-1)/2$个，其中每个类出现C次
  - 对测试样本$\boldsymbol{x}$，一共会得到$C(C-1)/2$个结果，然后投票vote
  - 每个分类器f_i采用其二值输出，即$\mathrm{sign}(f_i(\boldsymbol{x}))$

	1	2	3
1			
2	1		
3	1	3	

多类Multiclass(2)

- 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采用其实数值输出，即$f_i(\boldsymbol{x})$
  - $f_i(\boldsymbol{x})$的实数输出可以看成是其“信心”confidence
  - 最终选择信心最高的那个类为输出

$$\underset{i}{\operatorname{argmax}}\, f_i(\boldsymbol{x})$$

多类Multiclass(3)

- 直接解决多类问题(进一步阅读)
  - Crammer-Singer方法
  - http://jmlr.org/papers/v2/crammer01a.html

- DAGSVM(进一步阅读)
  - http://research.microsoft.com/apps/pubs/?id=68541

- ECOC(进一步阅读)
  - http://www.jair.org/papers/paper105.html

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
   - 或者实践（已有线性分类器的缺点，如感知机perceptron）
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

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$\min_i \left(y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)\right)$为1
  - 也可以对原问题做不重要的修改以使简化成为可能
  - 如（进一步阅读）LIBLINEAR假设不使用b

进一步的阅读

- 如果对本章的内容感兴趣，可以参考如下文献
  - 凸函数、拉格朗日乘子法、KKT条件：
    - Convex Optimization第一、二、五章
  - SVM和统计学习
    - http://research.microsoft.com/pubs/67119/svmtutorial.pdf
    - 最新会议论文集：ICML、NIPS、AISTATS、COLT、…
  - SMO: http://en.wikipedia.org/wiki/Sequential_minimal_optimization
    - LIBSVM, SVMLight
  - Pegasos: http://www.cs.huji.ac.il/~shais/code/
  - DCD/LIBLINEAR: http://www.csie.ntu.edu.tw/~cjlin/liblinear/
  - 加性核：我的主页→publications页面→[W5]