数据科学导论第 9 讲 —— 支持向量机

王小宁

中国传媒大学数据科学与智能媒体学院

2025 年 05 月 26 日

目录

分类

- 当因变量取离散值时，我们称之为分类。模仿回归的表达式，我们可以将分类问题写成

$$y = f(x)$$

- 其中，f 是关于 x 的函数，往往被称为分类器（classifier），比如 Logistic 模型、决策树、随机森林和支持向量机等都是经典的分类器。

支持向量机

- 支持向量机（support vector machine,SVM）是 90 年代中期发展起来的、基于统计学习理论的一种能够同时用于分类和回归的方法。支持向量机的理论出发点十分简单直观，在分类问题上，它就是通过寻求将特征空间一分为二的方法来进行分类的。
- 本讲以最常用的二分类问题为例，介绍支持向量机的原理。首先介绍基于超平面和间隔的最大间隔分类器（maximal margin classifier）。这种方法设计巧妙，原理简单，对大部分数据都容易应用。但是，由于超平面是由少数训练观测，即支持向量所确定，这就使得最大间隔分类器对样本的局部扰动反应灵敏。所以，进一步介绍了引入软间隔（soft margin）的支持向量分类器（support vector classifier）。

- 在实际问题中，不同类别观测之间常常是线性不可分的，面对这种情况，我们就需要使用支持向量机方法。支持向量机是将低维特征空间投影到高维中，从而在高维特征空间中实现线性可分，并且，在计算中使用了核函数技巧的一种方法。
- 最后将讨论支持向量机与 Logistic 回归的关系，以及支持向量回归问题。

问题的提出

例子 1

- 假设有 10 个观测，它们分属于两个类别，其中观测 1-5 个观测值是一类，观测值 6-10 是另一类，下表所示。
- 现在想建立一个分类器，使得对给定的任意一个新的观测，都能将它正确分类。那么，除了前面几章介绍的方法，例如 Logistic 回归、判别分析、树模型外，还有没有其他方法可以用于建立这样的分类器呢?

数据 1

Obs	V1	V2	V3	V4	V5	V6	V7	V8	V9	v10
x1	0.5	1	2.5	1	2.5	2.5	3	3	4	4
x2	3	2.5	3.5	2	3.8	1	2	1.5	3	1
x3	-1	-1	-1	-1	-1	1	1	1	1	1

最大间隔分类器

- 超平面（hyperplane）：一个 p 维空间的超平面就是它的 p-1 一个维的线性子空间。例如，二维空间的超平面是它的一维子空间，即一条直线，三维空间的超平面是它的二维子空间，即一个平面。如果用数学定义来表示，那么一个 p 维空间的超平面可定义为:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p = 0$$

- 所有满足以上式子的点 X 都会落到超平面上。

超平面

- $1 - 2x_1 - x_2 = 0$
- 对于任意给定的一个点 X，只需要将其代入上式的等号的左边项，就可以根据它的符号来判断 X 是位于超平面的哪一侧。

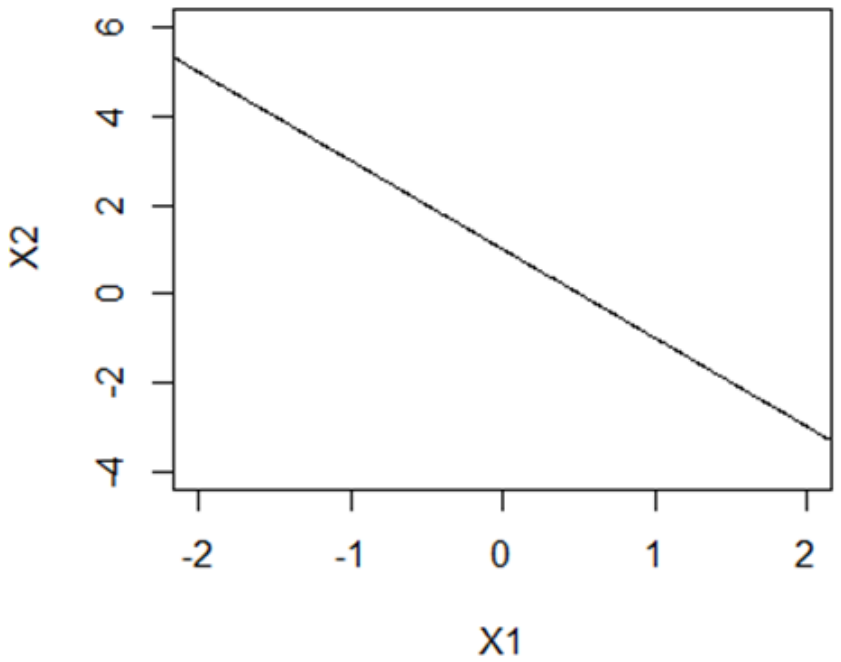


图 1: 超平面

构造分类器

- 假设我们有 n 个 p 维训练样本数据:$x_i = (x_{i1}, x_{i2}, \cdots, x_{ip})^T, i = 1, 2, \cdots, n$。
- 他们分属于两个类别，即 $y_i \in \{-1, 1\}$
- 在支持向量机的问题中，我们一般都用 “-1” 和 “1” 来表示两个类别。我们的目标是根据这些训练数据建立一个分类器，使得对于新的测试数据我们能准确地识别它们属于哪一类。
- 假设我们可以构造一个超平面把上述不同类别的观测完全分割开来:

$$\{x : f(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p = \beta_0 + \beta^T x = 0\}$$

- 那么这个超平面应该满足:

$$y_i = 1, f(x_i) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} = \beta_0 + \beta^T x > 0$$

$$y_i = -1, f(x_i) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} = \beta_0 + \beta^T x < 0$$

分割超平面

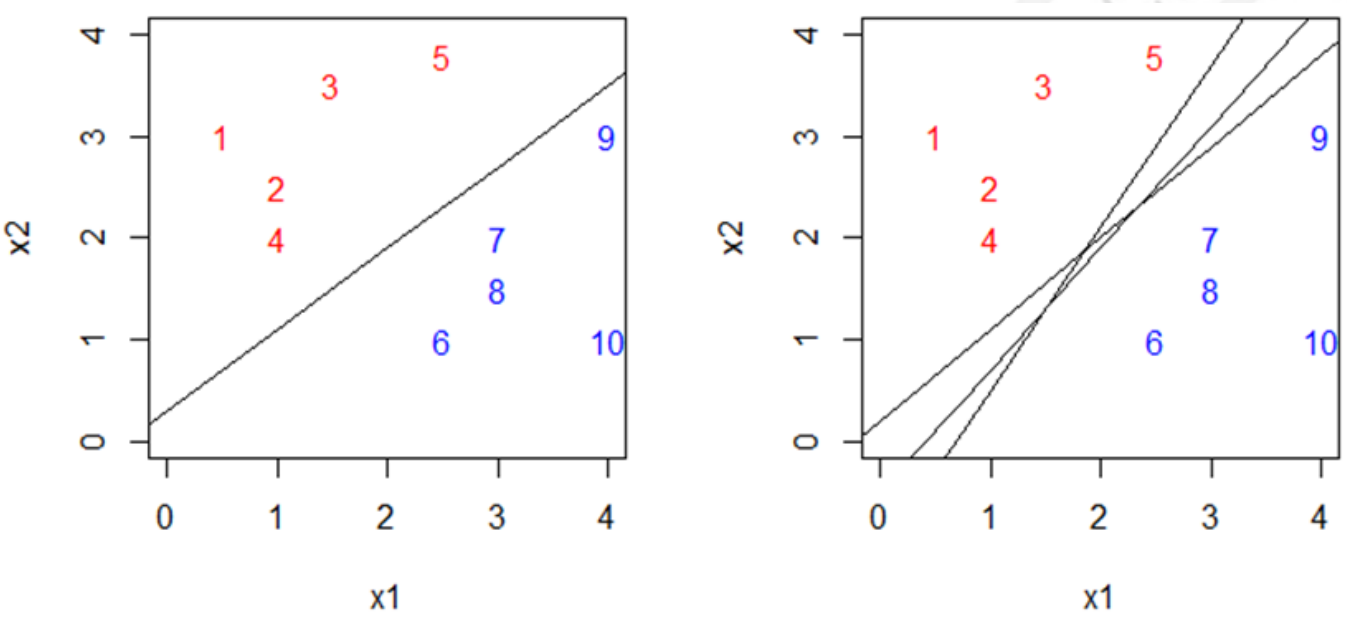


图 2: 分割超平面

- 另外，对于落在超平面两侧的观测点，我们还可以根据它们到超平面的距离来定义这种分类的准确性。由几何知识可知，点 x_i 到超平面

$$H = \{X : f(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p = \beta_0 + \beta^T x = 0\}$$

- 的距离为：

$$D(h, x_i) = |\beta^{*T}(x_i - x_0)| = \frac{|\beta^T(x_i - x_0)|}{||\beta||} = \frac{|\beta^T x_i + \beta_0)|}{||\beta||} = y_i \frac{f(x_i)}{||\beta||}$$

- 如果观测点距离超平面很远，那么我们就能肯定对该观测点的分类判断，但如果观测点离超平面很近，那么我们就不大确定对该观测点的判断是否正确。

构建最大间隔分类器

- 一般来说，若对于给定的不同类别的观测，可以构造某个超平面将他们分割开来，将这个超平面稍微地上移或下移或旋转，只要不碰到原有的那些观测，就能得到另外的超平面。
- 为了合理地构造分类器，有必要选择一个“最合适”的超平面。那么哪一个超平面才是“最合适”的呢?
- 最大间隔超平面（maximal margin hyperplane）：把位于超平面两侧的所有训练观测到超平面的距离的最小值称作观测与超平面的间隔（margin），若观测点离超平面距离越远，则对于该观测点的判断会更加有信心，这也表明了间隔实际上是代表了误差的上限。
- 基于此，就应该选择与这些观测具有最大间隔的超平面作为分类器, 称它为最大间隔分类器（maximal margin classifier）。

支持向量

- 观测的最大间隔超平面，即图中的黑色实线。两条虚线称为边界，它们到最大间隔超平面的距离是一样的，所以任意一条虚线到黑色实线的距离就是观测与超平面的间隔。

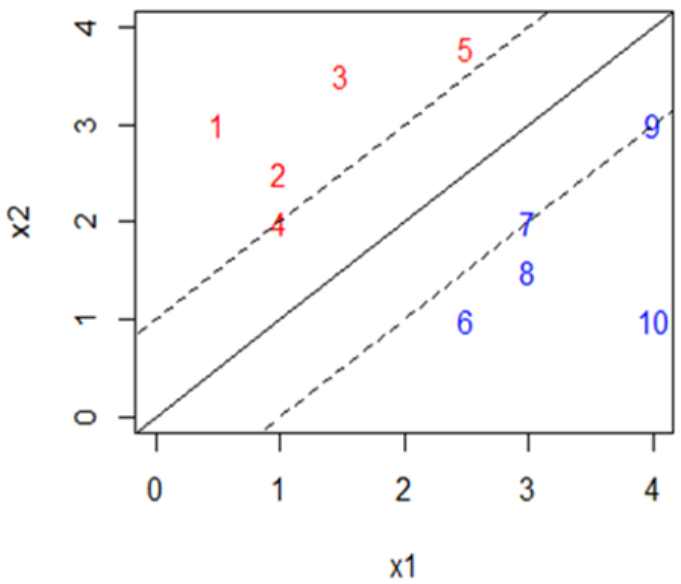


图 3: 最大间隔超平面

- 从图中可以看出，处于虚线上的观测 4、7、9 确定了最大间隔超平面，观测 4、7、9 中任意一个观测点只要靠实线稍移动了都会导致最大间隔超平面发生变化，而其余观测不管怎么移动，只要不越过各自的边界，就不会对超平面造成影响。
- 把 4、7、9 这样的观测称为支持向量（support vector），相当于超平面由这些点支撑（持），而每个点都是自变量空间中的一个向量。

最大间隔分类器

$$\max_{\beta_0,\beta} M$$

$$s.t.\ \|\beta\|=\sum_{j=1}^{p}\beta_j^2=1$$

$$y_i f(x_i)=y_i(\beta_0+x_i^T\beta)\geq M,\ \ \forall i$$

- 对于这个最优化问题，第一个约束条件是令参数的 2-范数为 1，这个条件实际上是保证了求解上述最优化问题时能得到参数的唯一解。
- 转化为求解它的对偶问题会更加容易些。这里，为了使求解得到的参数的唯一性，采用另一种约束 $M=\frac{1}{||\beta||}$，可以得到上式的对偶问题：

$$\min_{\beta_0,\beta}\|\beta\|$$

$$s.t.\ y_i f(x_i)=y_i(\beta_0+x_i^T\beta)\geq 1,\quad \forall i$$

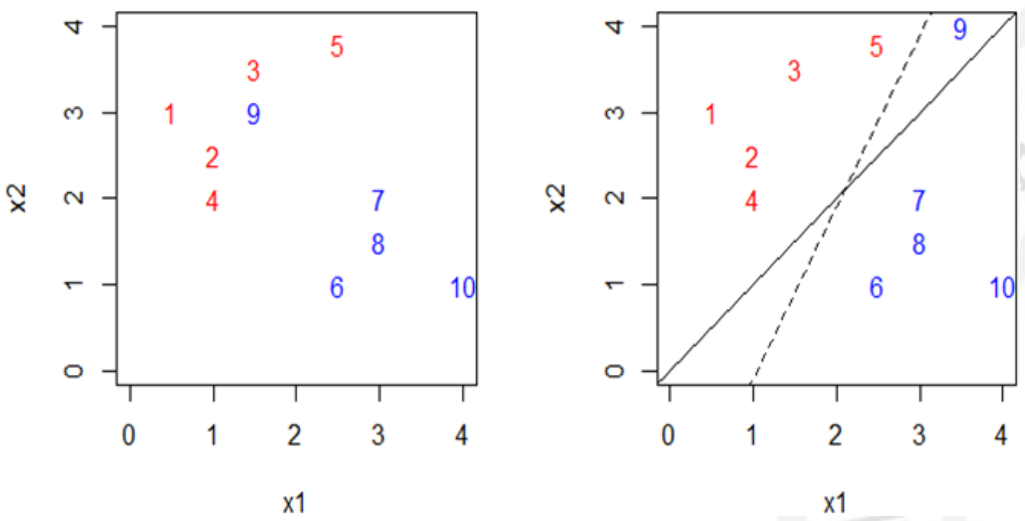


图 4: 不存在最大间隔超平面

- 找不到任何一个超平面可以完美地将不同类别的观测分割开来。
- 需要对超平面的概念进行扩展，即只要求超平面能将大部分不同类别的观测区别开来就好，称这样的超平面为软间隔（soft margin），相应的将由软间隔建立的分类器称为支持向量分类器（support vector classifier）。

支持向量分类器

支持向量分类器

- 软间隔分类: 为了提高分类器的稳定性以及对测试数据分类的效果，有必要对超平面的概念进行扩展，即只要求超平面能将大部分不同类别的观测区别开来就好。
- 软间隔的这个定义又包含两种情况，一是允许部分观测穿过边界，但此时对观测数据的分类仍然是正确的；二是允许部分观测数据分类错误

支持向量分类器

- 由软间隔建立的分类器就是支持向量分类器，它同样是通过建立超平面将训练观测分为两侧，以测试观测落入哪一侧来判断归属于哪一类，不同的只是这时的超平面是允许部分观测穿过边界，或者部分观测分类错误的。

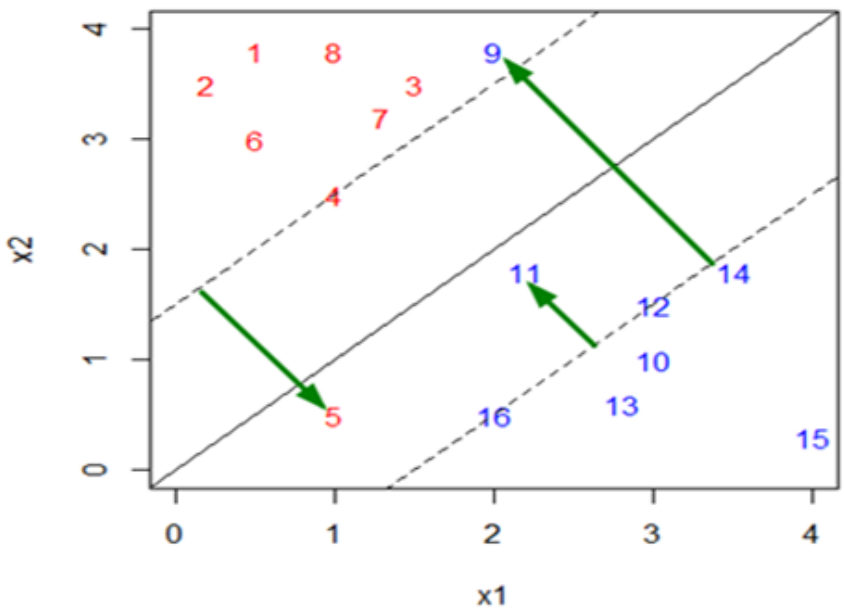
$$\max_{\beta_0,\beta,\varepsilon} M$$

$$s.t.\ \|\beta\| = \sum_{j=1}^{p} \beta_j^2 = 1$$

$$y_i f(x_i) = y_i(\beta_0 + x_i^T \beta) \geq M(1-\varepsilon_i),\ \ \forall i$$

$$\sum_{i=1}^{n} \varepsilon_i \leq C,\quad \varepsilon_i \geq 0$$

- 松弛变量的作用在于允许训练观测中有小部分观测可以穿过边界，甚至是穿过超平面。



- C 是所有松弛变量和的上界，是能容忍观测穿过边界的数量或者说程度。C 越大，能容忍观测点穿过边界的程度增大，间隔越宽。
- C 的选择涉及了偏差-方差的权衡问题。当 C 越大时，能容忍观测穿过边界的程度增大，间隔越宽，则此时能够降低方差，但却可能因拟合不足而产生较大的偏差；相反，C 越小，间隔越窄，分类器很有可能会过度拟合数据，即虽然降低了偏差，但可能产生较大的方差。在实际问题中，一般也是通过交叉验证的方法来确定 C 。

求解

- 同样的，转化为对偶问题的求解会更容易些，这里依然采用约束 $M = \frac{1}{||\beta||}$，来控制求解得到的参数的唯一性，可以得到上式的对偶问题：

$$\min_{\beta_0,\beta} \|\beta\|$$
$$s.t.\ y_i f(x_i) = y_i(\beta_0 + x_i^T\beta) \geq 1 - \varepsilon_i,\ \forall i$$
$$\sum_{i=1}^{n} \varepsilon_i \leq C,\quad \varepsilon_i \geq 0$$

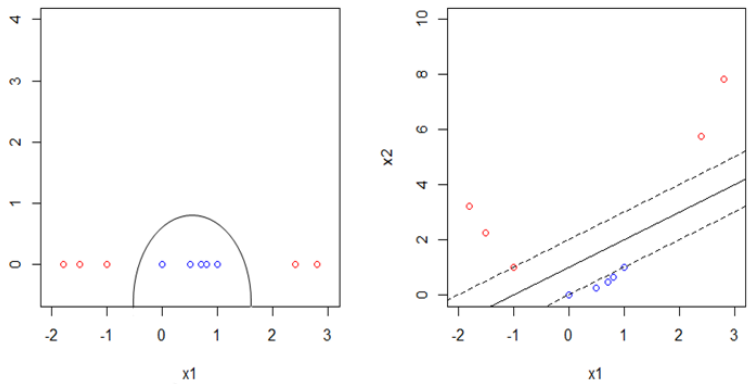
- 或者

$$\min_{\beta_0,\beta} \frac{1}{2}\|\beta\|^2 + \lambda\sum_{i=1}^{n} \varepsilon_i$$
$$s.t.\ y_i f(x_i) = y_i(\beta_0 + x_i^T\beta) \geq 1 - \varepsilon_i,\ \forall i$$
$$\varepsilon_i \geq 0$$

支持向量机

非线性决策分类

- 一般情况下，如果两个类别的观测之间存在线性边界，那么建立一个支持向量分类器可以得到不错的效果。但是有些情况下，如果边界是非线性的，那么支持向量分类器往往效果很差。



- 在一个一维空间中有几个观测点，它们分属于两个类别，分别用红色和蓝色表示不同的类别。在这种情况下，无法用一个点（即一维空间的超平面）来将不同类别的观测分开，而只有用一条复杂的曲线才能将它们分开。边界不是线性的，支持向量分类器是无效的。

把这个问题扩展到p维空间中，可以类似处理，还可以考虑使用观测x_i的不同多项式，如二次、三次甚至是更高阶多项式，或者是不同观测的交互项来扩大特征空间，进而在这个扩大的特征空间中构造一个线性超平面。例如对于p维观测 $x_i = (x_{i1}, x_{i2}, \cdots, x_{ip})^T$，使用$2p$个特征$x_{i1}, x_{i1}^2, x_{i2}, x_{i1}^2, \cdots x_{in}^2$,来构造支持向量分类器，此时的最优化问题就变成：

$$\max_{\beta_0, \beta_1, \beta_2, \varepsilon} M$$

$$s.t. \sum_{k=1}^{2} \sum_{j=1}^{p} \beta_{kj}^2 = 1$$

$$y_i f(x_i) = y_i(\beta_0 + x_i^T \beta_1 + (x_i^2)^T \beta_2) \geq M(1 - \varepsilon_i), \ \forall i$$

$$\sum_{i=1}^{n} \varepsilon_i \leq C, \quad \varepsilon_i \geq 0$$

- 能得到一个线性的边界，这个边界在原始特征空间中是一个二次多项式，通常它的解是非线性的。

构建支持向量机

- 支持向量机是支持向量分类器的一个扩展，它的基本思想通过将特征空间进行扩展，在扩展后的特征空间中求解线性超平面。但这里，支持向量机是通过核函数（kernel）来扩展特征空间的，这种扩展方式使得在新的特征空间中能有效求解得到线性的超平面。
- 在介绍核函数之前，对内积的概念进行介绍。两个观测 x_i 和 x_k 的内积定义为：$< x_i, x_k >= \sum_{j=1}^{p} x_{ij} x_{kj}$
- 可以证明，支持向量分类器的解可以描述为内积的形式:

$$f(x) = \beta_0 + \sum_{i=1}^{n} \alpha_i < x, x_i >$$

- 为了计算 $f(x)$ 的值，需要计算的是新的观测点 x 与每个训练观测 x_i 的内积。但事实证明，有且仅有支持向量对应的 α_i 是非零的，所以，若用 S 表示支持向量观测点的指标的集合，上式可写为:

$$f(x) = \beta_0 + \sum_{i \in S} \alpha_i < x, x_i >$$

- 为了估计系数进而得到线性分类器 $f(x)$，所需的仅仅是内积。现在，用一种一般化的形式 $K(x_i, x_k)$ 来代替内积，此时公式变成:

$$f(x) = \beta_0 + \sum_{i \in S} \alpha_i K(x, x_i)$$

核函数

- 核函数其实是一类用来衡量观测之间的相似性的函数。
- 常用的核函数

1. 线性核函数:$K(x_i, x_k) = \sum_{j=1}^{p} x_{ij}x_{kj}$
2. 多项式核函数:$K(x_i, x_k) = (1 + \sum_{j=1}^{p} x_{ij}x_{kj})^d$
3. 径向核 (高斯核) 函数:$K(x_i, x_k) = exp(-\gamma \sum_{j=1}^{p}(x_{ij} - x_{kj})^2))$

说明

- 关于核函数的选择一直以来都是支持向量机研究的热点，但是学者们通过大量的研究并没有形成定论，即没有最优核函数。通常情况下，径向核函数是使用最多的。
- 采用核函数而不是直接扩展特征空间的方式的优势在于，使用核函数，仅需要计算 C_n^2 个成对组合的 $K(x_i, x_k)$ 而若采取直接扩展特征空间的方式，是没有明确的计算量的；
- 对于某些核函数，例如径向核函数来说，它的特征空间是不确定的，并且可以扩展到无限维，所以是无法对这样的特征空间进行计算的。

与 Logistic 回归的关系

关系

- 为了建立支持向量分类器 $f(x)$, 可将优化问题转化为:
$\min_{\beta_0,\beta_1,\cdots,\beta_p} \{\sum_{i=1}^{n} max[0, 1 - y_i f(x_i)] + \gamma \sum_{j=1}^{p} \beta_j^2\}$
- γ 为调节参数，$\gamma \sum_{j=1}^{p} \beta_j^2$ 为岭回归的惩罚项，即损失 + 罚形式:

$$\min_{\beta_0,\beta_1,\cdots,\beta_p} \{L(X, Y, \beta) + \gamma P(\beta)\}$$

- $L(X, Y, \beta) = \sum_{i=1}^{n} max[0, 1 - y_i f(x_i)]$ 损失函数称为铰链损失 (Hinge loss)。
- 特点：对于边界外完全判对的观测点，损失为零；对于边界上的观测点以及判错的观测点，损失是线性的。

。

区别

- 铰链损失和在 Logistic 回归中使用的损失函数是非常接近的，只不过 Logistic 回归的损失函数在任何时候都是非零的，所以通常来说，SVM 和 Logistic 回归的结果也是非常接近的。
- 对于一个给定的问题，该如何选择是使用 SVM 还是 Logistic 回归呢?

1. 当类别的区分度较高时，选择 SVM 会更加合适，当然此时 LDA 也是适用的;
2. 如果想要得到估计的概率，那么就得选择 Logistic 回归;
3. 最后，对于决策边界是非线性的情况，使用了核函数的 SVM 方法是应用得更加广泛的。

支持向量回归

支持向量回归

- 它与分类问题的思想是类似的，不同的地方在于，现在的目的是要寻找一个超平面，在距离超平面 ε 的范围内尽可能的包含最多的观测点。
- 数学公式来定义，考虑如下回归模型：

$$f(x) = \beta_0 + \sum_{m=1}^{M} \beta_m h_m(x)$$

- 则支持向量回归所选的超平面是如下最优化问题的解：

$$\min_{\beta_0, \beta_2 \cdots \beta_M} \{ \sum_{i=1}^{n} V_\varepsilon(y_i - f(x_i)) + \lambda/2 \sum_{m=1}^{M} \beta_m^2$$

- 同样采用了" 损失函数 + 惩罚" 的形式，其中

$$V_\varepsilon = \begin{cases} 0, if & |\gamma| < \varepsilon \\ |\gamma| - \varepsilon, others \end{cases}$$

- 称为 ε 不敏感损失，它将与超平面的距离小于 ε 的观测的损失定义为 0，而距离大于等于 ε 的观测的损失定义为线性的形式。
- 只有距离大于等于 ε 的观测才会影响超平面的确定，与分类问题类似，我们称这些观测为支持向量。

- 上式的解可以表示为：

$$\hat{f}(x) = \sum_{i=1}^{n} \hat{\alpha}_i K(x, x_i)$$

- 支持向量回归是对普通线性回归的一种非线性推广，相比较于处理线性回归的最小二乘法而言，它的好处在于，它避开了对回归中因变量分布的假设，也不再局限于线性模型了。

本周推荐

1. 一本书：《传染-塑造消费、心智、决策的隐秘力量》，乔纳·伯杰，电子工业出版社，2017.8《疯传》姊妹篇
2. 一部电影：《统计的乐趣（The joy of stats）》，BBC
3. 作业：《R 语言实战（第 2 版）》，第 17 章代码实现

谢　谢!