**支持向量机**

**1. 简介**

支持向量机[1](Support Vector Machine, SVM)于1995年正式发表，主要作者是Vapnik。对于一个二分类问题，有一系列的样本点，它们从属于两个类别，如何找到一个“好”的超平面将其划分开。

即给定一组样本数据，这里的是向量，如果是二维，那就是平面上的一个个点，，表示两个不同的类别。

在二维平面中，一条直线的方程是，在三维空间是一个平面方程，甚至高维空间()中，这样的线性方程称为超平面，其中的未知数个数随之增加，通常写作向量形式

其中是维的向量，对应于每个未知量的待定系数，也是维的向量，是一个数。

两个平行的超平面和的距离计算公式如下:

而点到超平面的距离计算公式如下:

线性可分的定义:

存在一个超平面，将空间中所有的点分开，即对于任意的点，它满足下面的不等式:

这里的取值，只是为了区分两个类别，顺序和具体的值对公式的本质没有影响，例如将两种类别的交换顺序，这样对应的与原来的相差一个负号；同理，若修改为其它的值，所求的与原来的相差对应的倍数，表达的还是同一个平面。

合并(4)中的两个不等式，可以写作一个不等式:

如果这样的超平面存在，即式(5)成立，则称样本线性可分；否则线性不可分。若能够找到一个这样的超平面，则能够找到无数个，它们是一系列的平行超平面，只相差常数。

使得(5)式等号成立的这些向量，是超平面平行移动的边界，这些样本称为支持向量，由此可见，SVM寻找的超平面只与这些“支持向量有关”，因此SVM适用于数据量较小的分类问题。在边缘的两个超平面，其距离根据(2)式，有。但如何确定“最优”的一个超平面。Vapnik的想法是，使得最大，即尽可能把两类样本分隔开，让模型具备较好的泛化能力。一旦确定以后，选择这个间隔(margin)正中间的超平面，由此可以确定。

对应的优化问题可以写作:

进一步，可以将其化为:

式(7)是SVM的基本形式。

**2. 非线性可分处理**

但实际中，对于线性可分来说，有如下问题。

在低维空间中，线性可分的概率并不大，例如最经典的异或问题，在二维平面中，无法找到一条直线，将两类样本分开；但是在高维空间中，例如同样的点，在三维空间中，很容易可以找到一个平面将其分开，这暗含了一个思考，空间中的点，随着维度的升高，其被线性可分的概率越大。这其实就是Cover’s theorem[2]。

例如随机给定一些点，在二维平面中，这类似于一个异或问题，无法找到一条直线将其分开。如图1所示。

|  |  |
| --- | --- |
|  |  |
| (a). 二维空间的两类样本 | (b). SVM划分结果 |

图1 二维空间线性不可分情况

若定义一个映射，将二维坐标的点映射至三维空间点。此时在三维空间中，有较大的概率找到一个平面，将两个类别的样本分开。如图2所示。

|  |  |
| --- | --- |
|  |  |
| (a). 三维空间的两类样本 | (b). SVM划分结果 |

图2 三维空间SVM分类结果

将这个解在二维空间显示的结果如图3所示，此时二维空间看到的解不是“直线”。

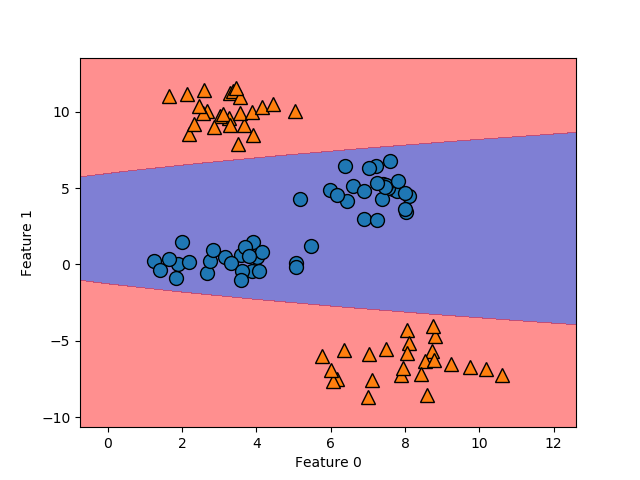


图3 二维空间看到的解

对于上述问题，将一个低维空间映射到高维空间，甚至是无限维度，因此需要引入映射；同时，为了使得样本尽量被线性可分，引入了松弛变量。满足(5)的样本，对应产生的间隔是“硬间隔”，即要求每一个样本点都要被划分正确，引入松弛变量的目的是，容忍模型中少量不满足(5)式的样本点，使得模型不必过拟合。

这里不能过大，否则会使求出的超平面过于发散，因此在目标函数中引入了对它的约束，这里是超参数，称为正则项。是映射。

**3. 核函数**

如果是一个无限维度的映射，这会给求解问题带来困难，因此有了一定的转换。由此引入了核函数(kernel function)。核函数的定义是，对于变换，有

这里本质是一个数，而等式的右边是两个的内积。即，我们无需知道具体的变换是什么，使用核函数便能够算出。

并不是所有的函数都可以作为核函数，这里有了Mercer定理[3]的支撑。核函数将线性的机器学习变为了非线性。常见的核函数有以下几种，如表1。

表1 常见的内核

|  |  |  |
| --- | --- | --- |
| 名称 | 表达式 | 参数 |
| 线性核 |  | 本质没有变换 |
| 多项式核 |  |  |
| 高斯核 |  |  |
| Sigmoid核 |  | 和均为超参数 |

**4. 原问题和对偶问题**

在式(8)中，仅有，并没有核函数的影子，所以还需要对优化问题变形，即原问题和对偶问题。

原问题的定义:

这是一个很朴实的定义，囊括了所有可能的情况，例如，若实际优化问题是求取最大值，可以变换为求的形式；同理，若约束中有大于等于的情况，也可以通过移项变为标准形式。事实上，等式约束也可以用两个不等式等价表示，函数其实也可以不用。

其对偶问题的定义如下:

这里和都是向量，目标是求这个函数的最大值，中的部分，是固定和遍历，求取函数最小值。这里表示每一个中的分量，其值都大于0。

假设原问题的解为，对偶文件的解为，则定义原问题和对偶问题的间距(Duality Gap)，令

由于对其中的式子求取下确界(是自变量)，则

根据约束条件，，，结合式(13)，式(12)可以进一步缩放为:

即

对于一些特定的问题，这个对偶间距。

这里有强对偶定理:

若是凸函数，且，，则。

而SVM中的优化问题满足强对偶问题。进一步可以推出，即对于，或。这个条件称为K.K.T条件。

**5. 优化问题推到过程**

实际的SVM的理论推到过程如下:

将式(8)化为标准格式，注意这里和都是待求的参数，将其化为对偶问题，得式(16)。

这里原问题中没有等式的约束，即是不存在的，但是不等式约束有两个，因此分别用和对其进行限定。

令。分别求，和。

这里是向量，对其求偏导，等同于对其中每一个分量求偏导，有:

令，和，有:

将式(18)，带入，化简可得:

此时，最后求解的问题变为如下形式:

这里由于，将消去，减少了一个求解的变量。优化问题的标准求解算法是SMO，这里假设求出了最优解。

由于满足，即对偶问题的最优解，对应的原问题也是最优解。即

但无限维度的映射中，可能并不清晰，因此的计算相对困难。事实上，若存在一个样本需要预测，并不需要知道，只需要计算出它属于哪一个类别即可，即对于一个预测样本，其预测类别为，将式(21)带入，

这里再次化为了核函数形式!对于的求解，需要用到K.K.T条件。结合式(16)，可以推出:

同时注意到式子依旧成立，因此若，则，则；同时成立，因此可以解出

由于，因此，式(24)进一步化为:

最后将求出的多个进行一个平均，作为实际的。

至此，二分类情况下的SVM理论大致如此。

**6. 多分类问题**

而对于多分类的问题，SVM是也可以通过改造达到目的。最原始的方法是修改目标函数，然后找到最优解，但计算复杂度相对较高；更为常用的方式是后面两种:

(1) 一类对其它类(one-v.s.-rest, ovr)

依次把某个类别归为一类，其余类别归属于另外一类，若总供有类，一共需要训练个模型。

(2). 一类对另一类(one-v.s.-one, ovo)

对于所有的类别，两两之间训练一个判别模型，若总供有类，一共需要训练个模型。

当然还有一些其它的组合方式，例如用层次的树状结构进行划分。

**Reference**

1. Cortes C, Vapnik V. Support-vector networks[J]. Machine learning, 1995, 20(3): 273-297.
2. https://en.wikipedia.org/wiki/Cover%27s\_theorem
3. https://en.wikipedia.org/wiki/Mercer%27s\_theorem
4. 周志华. 机器学习
5. 胡浩基 机器学习(课程)
6. Boyd S, Boyd S P, Vandenberghe L. Convex optimization[M]. Cambridge university press, 2004.