**支持向量机理论**

**线性支持向量机**

给定一个训练样本其中

。为训练样本中第个特征向量。是的类标记代表正类，代表负类，假设训练数据集是线性可分的，线性支持向量机学习目标是寻找一个分类间隔最大的分离超平面，将特征空间划分为两部分，一部分为正类，一部分为负类。分类间隔指的是距分离超平面最近的样本点到平面的距离。

分离超平面为，计算出分类间隔为，所以线性可分支持向量机的原始最优化问题为





其中，最大化和最小化是等价的。约束条件表示任一训练样本点到分离超平面的距离要大于等于分类间隔

原始最优化问题是一个凸二次规划问题。为求解原问题。利用Wolfe对偶定理，能够得到等价对偶问题，





通过求解对偶问题得到原始问题的最优解。

如果训练样本存在少量噪声数据，即训练样本不能完全线分，可以为每个样本引入松弛变量,使训练样本点到分离超平面的距离加上松弛变量大于等于分类间隔，所以线性不可分的线性支持向量机学习问题变为

 



其中，为惩罚参数，值大时，对误分类的惩罚增大，值小时，对误分类的惩罚减小。最优化目标函数包含两层含义，一使分类间隔尽量大，同时使误分类点的个数尽量少。

相应的对偶问题为





**非线性支持向量机**

对于线性不可分的情况，可以用一个非线性变换，将原特征空间映射到新的特征空间。



将在原来特征空间线性不可分的问题，转换为新特征空间中的线性可分的问题。

定义核函数，



其中都是样本点的特征向量

相应的非线性支持向量的对偶优化问题为





在求解最优化的问题的时候，只需要考虑核函数，而不需要显式地定义特征空间和非线性变换函数，这样的技巧称为核技巧。在实际的应用中往往需要应用领域知识直接选择核函数，常用的核函数主要有类

多项式和函数，所得到的是阶多项式分类器

高斯核函数，所得到的分类器与传统的径向基神经网络分类器最大区别在于每个基函数中心对应一个支持向量，它们与输出权值都是由算法自动确定

核函数，这时支持向量机实现是包含一个隐含层的多层感知机，隐含层的节点个数由算法自动确定，而且不存在困扰神经网络的局部极小点问题。

**支持向量机训练算法**

**块算法**

块算法是由Boser**[1-10]**等人提出来的，出发点是删除矩阵中对应拉格朗日乘子为零的行和式不会对最终的结果产生影响。其目标就是通过某种迭代方式逐步排除非支持向量，块算法将矩阵的规模从样本数的平方减少到具有非零拉格朗日乘子的样本数的平方。块算法减少了矩阵的规模，所以在很大程度上降低了训练过程中对存储容量的要求。但是块算法的目标是找出所有的支持向量，所以当支持向量的数量很大的时候，块算法会变得非常复杂。

**分解算法**

分解算法**[1-11]**是由Osuna等人提出来的,是解决大规模问题的有效方法。分解算法是将原二次规划问题分解成一系列规模较小的二次规划问题，进行迭代求解，在每次迭代的过程中，选择拉格朗日乘子分量的一个子集作为工作集，在工作集上，利用传统的优化算法，求解子问题。

序列最小最优化**[1-13]**（sequential minimal optimization）算法,是在1998年，由Platt提出。序列最小最优化算法是分解算法的一个特例。其工作集个数为2，SMO将原二次规划问题分解为只包含两个变量的子问题。其中两个子问题可以解析求解，工作集的选择是基于启发式的。首先从外层循环中选取出违背KKT条件最严重的点，并将其对于的变量之后作为第一个变量，再在内层循环中，选择第二个变量。SMO算法的工作集为2，虽然可能会增加迭代次数，但是子问题规模很小，所需的计算量小，所以SMO算法通常表现出整体快速收敛的性质.另外SMO算法不需要存储核矩阵，没有矩阵运算，容易实现。

**增量算法**

前面介绍的块算法，分解算法都不支持在线训练。而增量算法支持在线训练。

增量学习是机器学习系统在处理新增样本时，能够只对原学习结果中与新样本有关的部分进行增加修改或删除操作，与之无关的部分则不被触及。增量训练算法的一个突出特点是支持向量机的学习不是一次离线进行的，而是一个数据逐一加入反复优化的过程。

**支持向量机的应用**

**结束语**