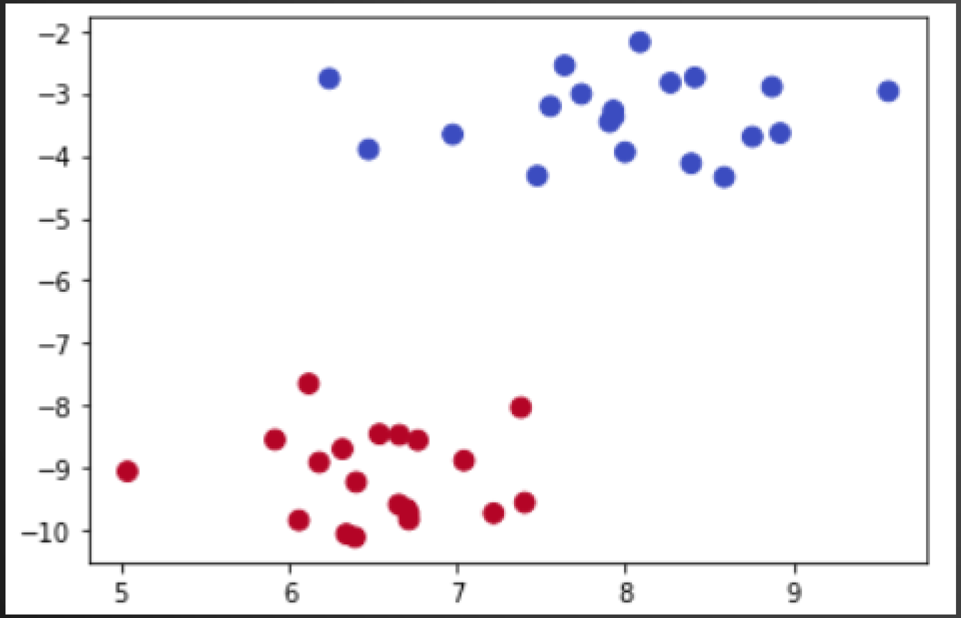
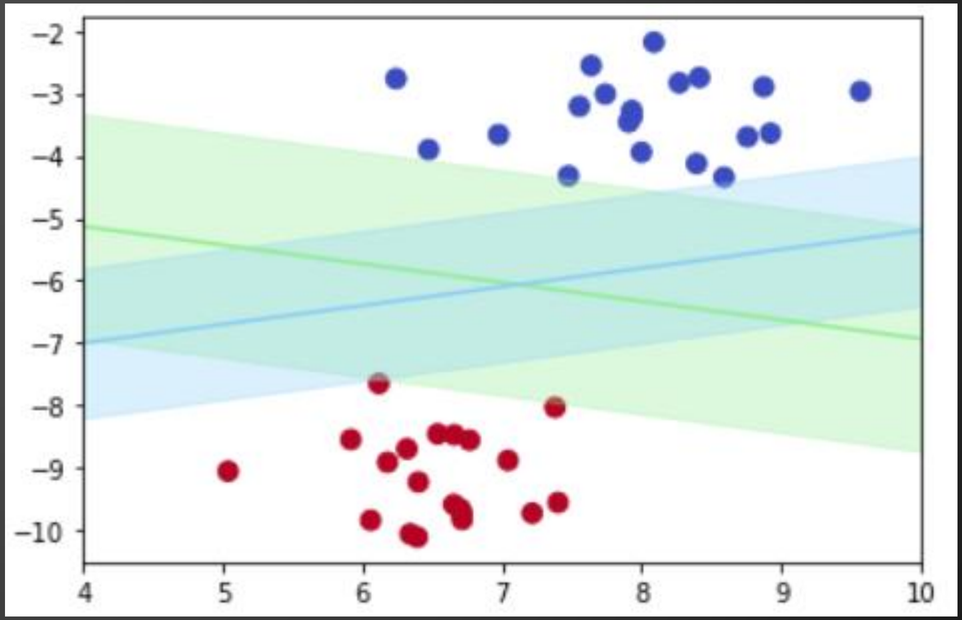
支持向量机（SVM）

支持向量机原理

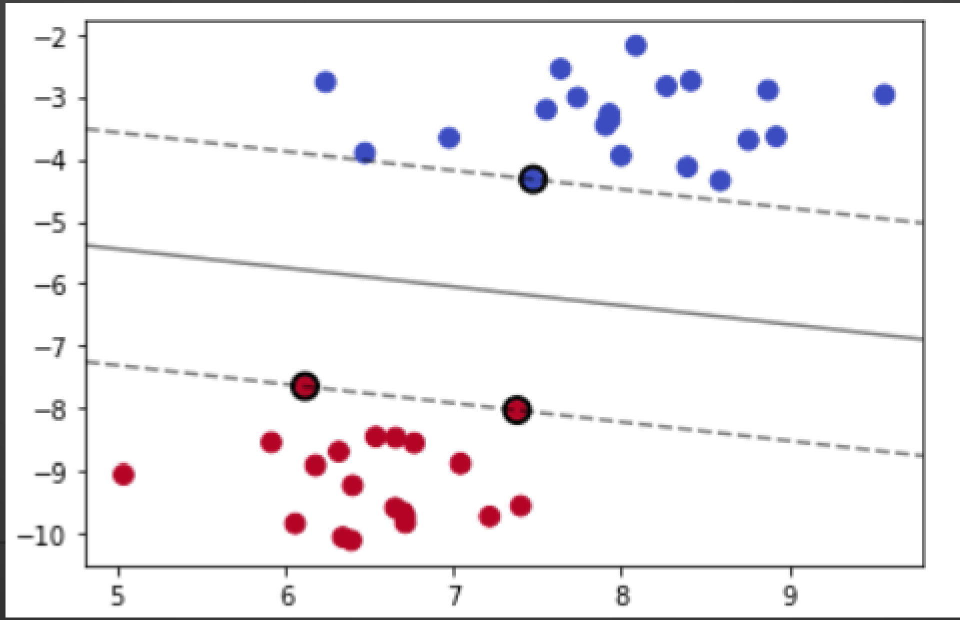
- 支持向量机：support vector machine，SVM
- 目标：对于以下数据，找到一条最优直线将两组数据分开，最优的意思是希望间隔尽可能大。





SVM中的几个关键概念

- 分隔超平面（hyperplane）：分隔的超平面位于间距的中间。
- 间隔（margin）：两根虚线间的距离
- 支持向量（support vector）：在间距边界上的点。
- 目标：寻找最大间隔，也就是要求这个超平面方程。

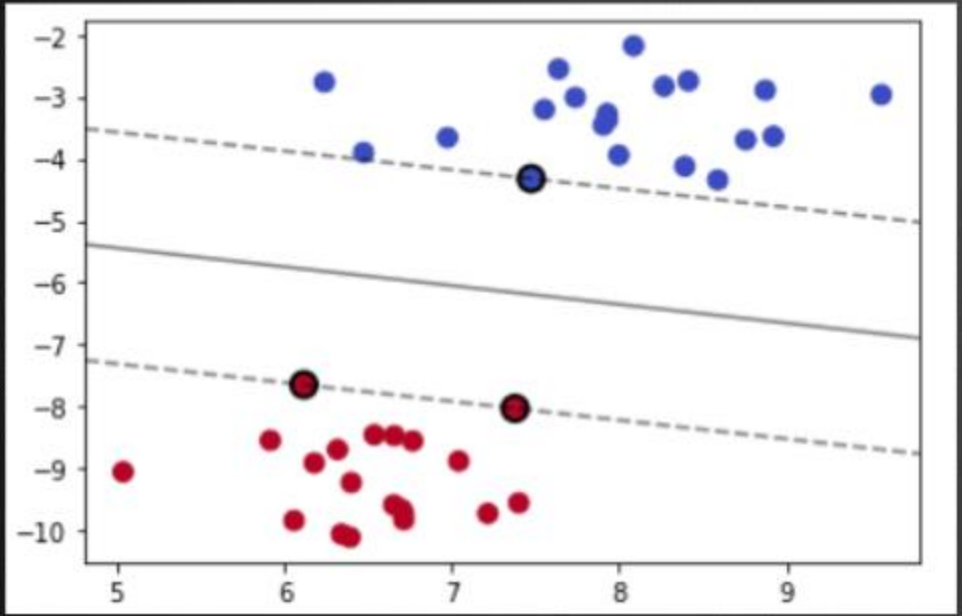


SVM问题：由简至繁三类问题

1. 当训练样本线性可分时，通过硬间隔最大化，学习一个线性可分支持向量机；
2. 当训练样本近似线性可分时，通过软间隔最大化，学习一个线性支持向量机；
3. 当训练样本线性不可分时，通过核技巧和软间隔最大化，学习一个非线性支持向量机。

SVM原理：线性可分

假设有m个样本点$(\vec{x}_1, y_1), (\vec{x}_2, y_2), \cdots, (\vec{x}_m, y_m)$，$y_i \in \{-1, +1\}$，每一个样本有$n$个特征。



任意分隔超平面可以用以下这个线性方程来描述：

$$w^T x + b = 0$$

接下来，从两个角度来看SVM问题：**几何角度**和**函数角度**。

从几何角度看SVM

根据几何知识，任意样本点$x=(x_1,x_2,...,x_n)$到直线$w^Tx+b=0$的距离公式为：

$$\frac{|w^Tx+b|}{||w||}$$

其中，$||w||=\sqrt{w_1^2+w_2^2...+w_n^2}$

假设间隔为d，我们的目标是最大化这个距离d，这相当于**目标函数**，用数学式子表达为：

$$\max_{w,b}\ d$$

而对于其他点到平面的距离则要求大于等于$\frac{d}{2}$，样本边界的点（支持向量）使等号成立，这相当于**约束条件。**

对于类别+1，即$y_i=+1$，要求$\frac{w^Tx_i+b}{||w||}\geq\frac{d}{2}$；

对于类别-1，即$y_i=-1$，要求$\frac{w^Tx_i+b}{||w||}\leq-\frac{d}{2}$

即

$$\begin{cases} \frac{w^T x_i + b}{||w||} \geq +\frac{d}{2} & y_i = +1 \\ \frac{w^T x_i + b}{||w||} \leq -\frac{d}{2} & y_i = -1 \end{cases}$$

说明：样本边界的点（支持向量）使等号成立。

接下来，将这两个类别的约束条件合起来，写成一个式子，即

$$y_i \times \frac{w^T x_i + b}{||w||} \geq \frac{d}{2}$$

仔细看上面这个式子，无论$y_i = +1$还是$y_i = -1$，都满足上面给出的约束条件。
对于样本边界的点（支持向量），有以下式子成立。

$$y_i \times \frac{w^T x_i + b}{||w||} = \frac{d}{2}$$

由于上式是从几何角度得出的，不放将其称为**几何支持向量等式。**

从函数角度看SVM

以上是从几何的角度来考虑SVM问题，接下来从函数的角度来考虑SVM问题。

已知分隔超平面的方程为：

$$w^T x + b = 0$$

考虑函数$f(x) = w^T x + b$

若是超平面能够正确地分类样本，则对于类别1，即当$y_i = 1$时，$w^T x_i + b > 0$，对于类别-1，即当$y_i = -1$时，$w^T x_i + b < 0$。

为了方便接下来的讨论，在保持分隔超平面方程$w^T x + b = 0$不变的情况下，可以对函数$w^T x + b$进行缩放，使其满足当$y_i = 1$时，$w^T x_i + b \geq 1$，当$y_i = -1$时，$w^T x_i + b \leq -1$，样本边界的点（支持向量）使等号成立。

所以，对于样本边界的点（支持向量），有以下式子成立。

$$y_i \times (w^T x_i + b) = 1$$

上一页得到$y_i \times (w^T x_i + b) = 1$，结合之前得出的“几何支持向量等式”

$$y_i \times \frac{w^T x_i + b}{||w||} = \frac{d}{2}$$

有

$$d = \frac{2}{||w||}$$

所以，目标函数变为：

$$\max \frac{2}{||w||}$$

约束条件变为：

$$y_i \times (w^T x_i + b) \geq 1 \, , i = 1, 2, \cdots, m.$$

SVM问题的损失函数

但是，对于机器学习问题来说，一般都是求损失函数（目标函数）的最小值。

因为最大化$\frac{2}{||w||}$和最小化$\frac{||w||}{2}$等价，为了方便计算，加上一个平方去除根号，即得$\frac{1}{2}||w||^2$。

所以将以上目标函数变为$\min \frac{1}{2}||w||^2$。

于是，得到下面这个优化问题：

$$\min \frac{1}{2}||w||^2$$
$$s.t. \quad y_i(w^T x_i + b) \geq 1 \,, i = 1, 2, \cdots \,, m.$$

其中，$||w|| = \sqrt{w_1^2 + w_2^2 ... + w_n^2}$

说明：s.t.是subject to （such that）的缩写，受约束的意思。

SVM问题的求解

接下来，求解这个SVM优化问题，本质上是一个不等式约束优化问题。

已知SVM优化问题为：

$$\min \frac{1}{2}||w||^2$$
$$s.t. \quad y_i(w^T x_i + b) \geq 1\,, i = 1, 2, \cdots, m.$$

用拉格朗日乘子法求解，大致步骤为：

- 第1步：构造拉格朗日函数，得到其对偶问题
- 第2步：分别对参数w和b求偏导并令其等于0，得到只含参数λ的二次规划问题
- 第3步：利用SMO算法求解这个二次规划问题，得到参数λ
- 第4步：求解参数w和b，得到超平面方程，即最大分隔超平面。

第1步：构造拉格朗日函数，得到其对偶问题

根据拉格朗日乘子法，构造拉格朗日函数：

$$L(w,b,\lambda)=\frac{1}{2}||w||^2+\sum_{i=1}^{m}\lambda_i[1-y_i(w^Tx_i+b)] \quad s.t.\ \lambda_i\geq 0$$

假设找到了目标函数的最小值p，即$\frac{1}{2}||w||^2=p$。在上面的式子中，右边第二项$\sum_{i=1}^{n}\lambda_i[1-y_i(w^Tx_i+b)]\leq 0$，因为$\lambda\geq 0$。所以有$L(w,b,\lambda)\leq p$，为了找到最优的参数$\lambda$，使得$L(w,b,\lambda)$接近$p$，所以问题转换为$\max\limits_{\lambda} L(w,b,\lambda)$。

于是，之前的优化问题可以转换为：

$$\min_{w,b}\ \max_{\lambda} L(w,b,\lambda) \quad s.t.\ \lambda_i\geq 0$$

利用强对偶性，将以下目标函数转化为：

$$\max_{\lambda}\min_{w,b} L(w,b,\lambda)$$

第2步：分别对参数w和b求偏导并令其等于0，得到只含参数λ的二次规划问题

求解参数w和b，对参数求偏导，得

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n} \lambda_i x_i y_i = 0$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{n} \lambda_i y_i = 0$$

由以上方程可得

$$w = \sum_{i=1}^{n} \lambda_i x_i y_i, \quad \sum_{i=1}^{n} \lambda_i y_i = 0$$

将以上结果代入第一步构造的拉格朗日函数，可得：

$$
\begin{aligned}
L(w,b,\lambda) &= \frac{1}{2}||w||^2 + \sum_{i=1}^{n}\lambda_i[1 - y_i(w^T x_i + b)] \\
&= \frac{1}{2}||\sum_{i=1}^{n}\lambda_i x_i y_i||^2 + \sum_{i=1}^{n}\lambda_i - \sum_{i=1}^{n}\lambda_i y_i(w^T x_i + b) \\
&= \frac{1}{2}||\sum_{i=1}^{n}\lambda_i x_i y_i||^2 + \sum_{i=1}^{n}\lambda_i - \sum_{i=1}^{n}\lambda_i y_i w^T x_i + \sum_{i=1}^{n}\lambda_i y_i b \\
&= \frac{1}{2}||\sum_{i=1}^{n}\lambda_i x_i y_i||^2 + \sum_{i=1}^{n}\lambda_i - \sum_{i=1}^{n}\lambda_i y_i(\lambda_i x_i y_i) x_i + \sum_{i=1}^{n}\lambda_i y_i b
\end{aligned}
$$

好了，观察一下上面的式子，第一项和第四项都为零，于是得到一个仅含参数λ的式子

$$
L(\lambda) = \sum_{i=1}^{n}\lambda_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j y_i y_j (x_i x_j)
$$

第3步：利用SMO算法求解这个二次规划问题，得到参数λ

于是，得到这样二次规划问题：

$$\max L(\lambda) = \max[\sum_{i=1}^{n} \lambda_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j y_i y_j (x_i x_j)]$$

$$s.t. \quad \sum_{i=1}^{n} \lambda_i y_i = 0, \lambda_i \geq 0$$

这种二次规划问题一般采用SMO(Sequential Minimal Optimization)算法求解。

SMO算法，中文名称为序列最小优化算法，核心思想：每次优化一个参数，其他参数固定。

对于以上这个优化问题，无法每次只优化一个参数，因为有一个约束条件$\sum_{i=1}^{n} \lambda_i y_i = 0$，如果每次优化一个λ，其他λ固定，则这个待优化的λ将不再是变量，因为它可以有其他固定的λ推出，所以对于这个问题，每次选取两个λ优化。

优化的具体步骤为：

1、选择两个需要优化的参数λ_i和λ_j，其他参数固定。

此时，约束条件变成：$\lambda_i y_i + \lambda_j y_j = c, \lambda_i \geq 0, \lambda_j \geq 0$，其中，$c = -\sum_{k \neq i,j} \lambda_k y_k$，由此可以得出$\lambda_j = \frac{c - \lambda_i y_i}{y_i}$，也就是说，其实我们可以用$\lambda_i$表达$\lambda_j$，这样就把这个优化问题变成了仅有一个约束条件的优化问题，这个唯一的约束条件是$\lambda_i \geq 0$。

2、对这个优化问题，对参数λ_i求导，可以得到λ_i的一个值，进而还可以得到λ_j的一个值。

3、重复步骤1,2，即多次迭代，直至函数收敛。

第4步：求解参数w和b，得到超平面方程，即最大分隔超平面。

在利用强对偶性转换中，通过求偏导，得到

$$w=\sum_{i=1}^{n}\lambda_i x_i y_i,\quad \sum_{i=1}^{n}\lambda_i y_i=0$$

第一个式子，$w=\sum_{i=1}^{n}\lambda_i x_i y_i$，通过这个式子可以得到$w$。

接着，求参数b。随便找一个支持向量（边界点）(x_0,y_0)代入方程$y_0(w^T x_0+b)=1$，解出b即可。

$b=y_0-w^T x_0$

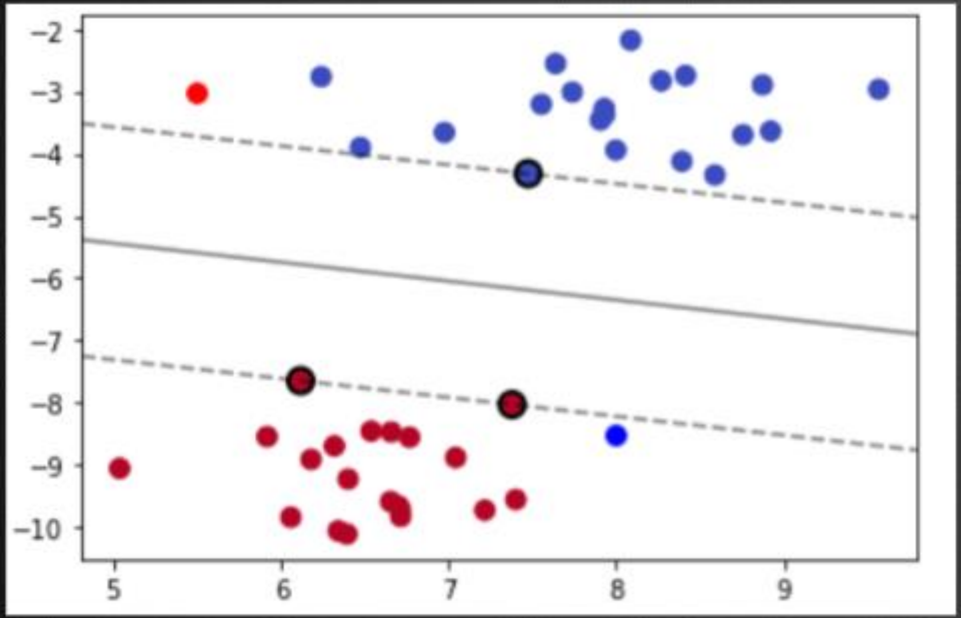
为了使模型更加具有鲁棒性，可以求得支持向量的均值。

$$b=\frac{1}{n}\sum_{i=1}^{n}(y_i-w^T x_i)$$

至此，参数w和b都求出来了，于是得到分隔超平面的方程。

软间隔

前面我们的推导都基于线性可分的情况，但是在实际应用中，完全线性可分的样本是很少的，总有一些异常点，如下图所示。



为了解决这个问题，为每个样本引入一个松弛变量ξ_i，令$\xi_i \geq 0$，且$1 - y_i(w^T x_i + b) - \xi_i \leq 0$。

这里需要说明：异常点里间隔边界越远，其松弛变量ξ_i的值就越大。对于正常点来说，其松弛变量ξ_i的值等于0，即满足$y(w^T x + b) \geq 1$

软间隔SVM问题求解

增加软间隔后，优化目标变成了：

$$\min \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i \quad s.t.\ g_i(w) = 1 - y_i(w^T x_i + b) - \xi_i \leq 0, \xi_i \geq 0, i = 1, 2, \ldots, n$$

其中C是一个大于0的参数，C也是sklearn中svm库的一个参数，可以理解为对错误样本的惩罚程度。

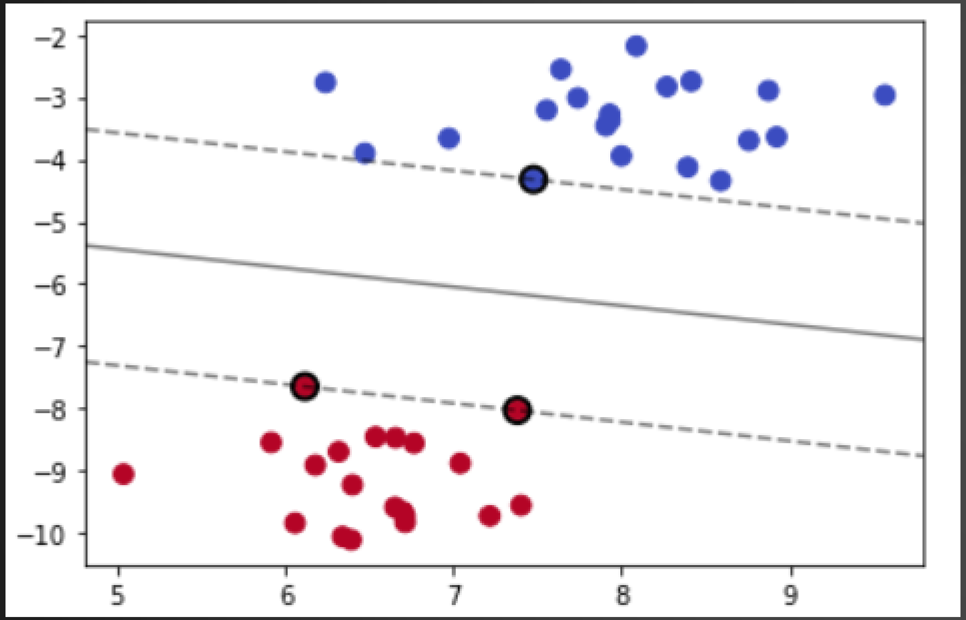
- C 越大， ξ_i就越小，说明有较少的异常点跨过间隔边界，此时模型也就越复杂。
- C越小，ξ_i就越大，说明有较多的异常点跨过间隔边界，此时模型也就越简单。

接下来求解这个新的优化问题。

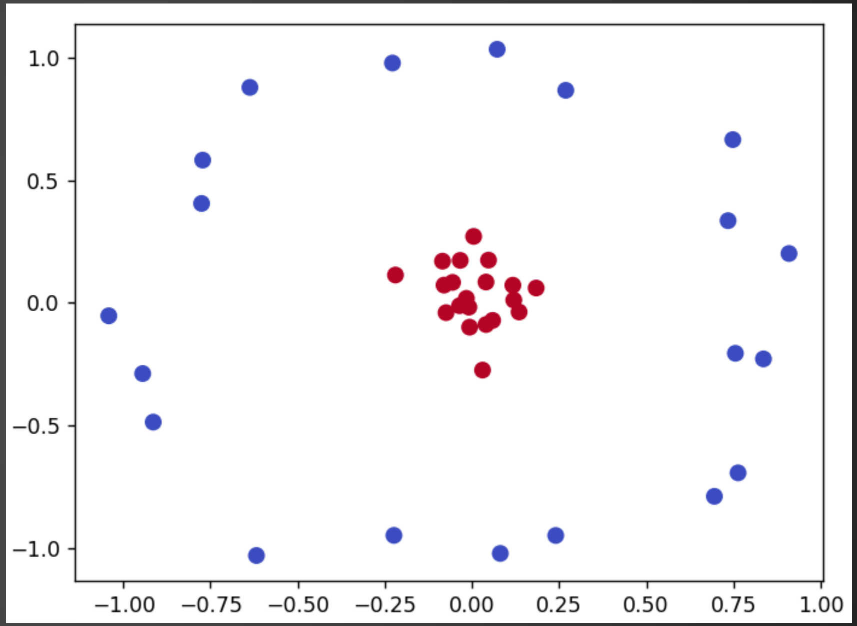
还是用之前的方法：拉格朗日乘子法。

详细求解步骤参考课件资料：一文搞懂支持向量机。

从线性可分到不可分



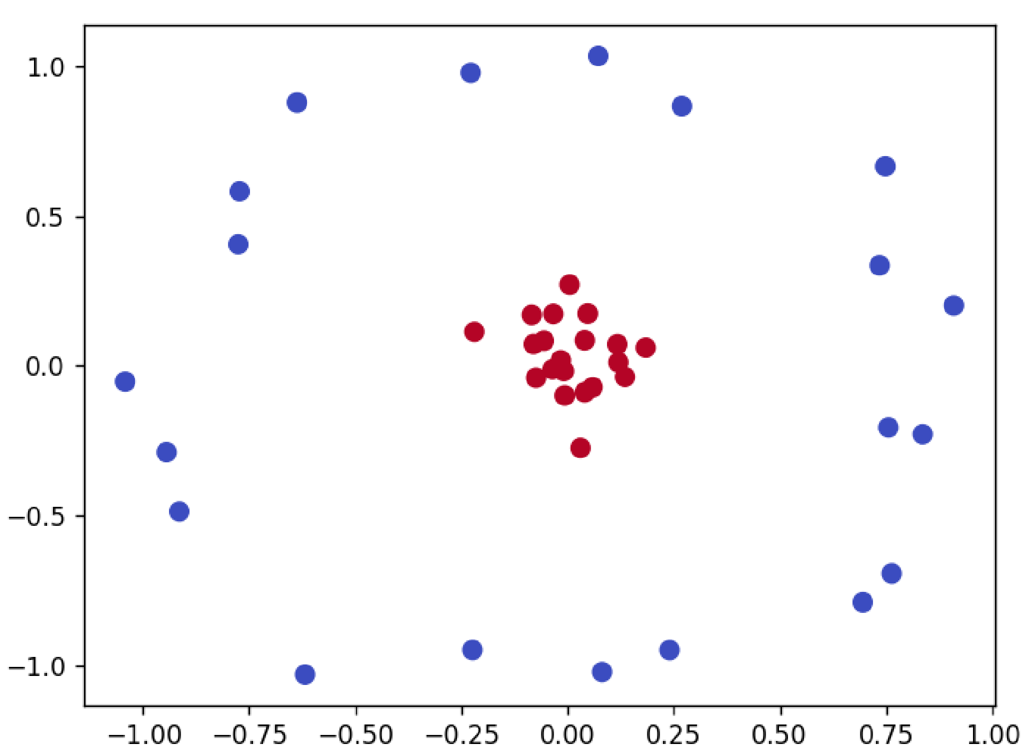
线性可分的情况

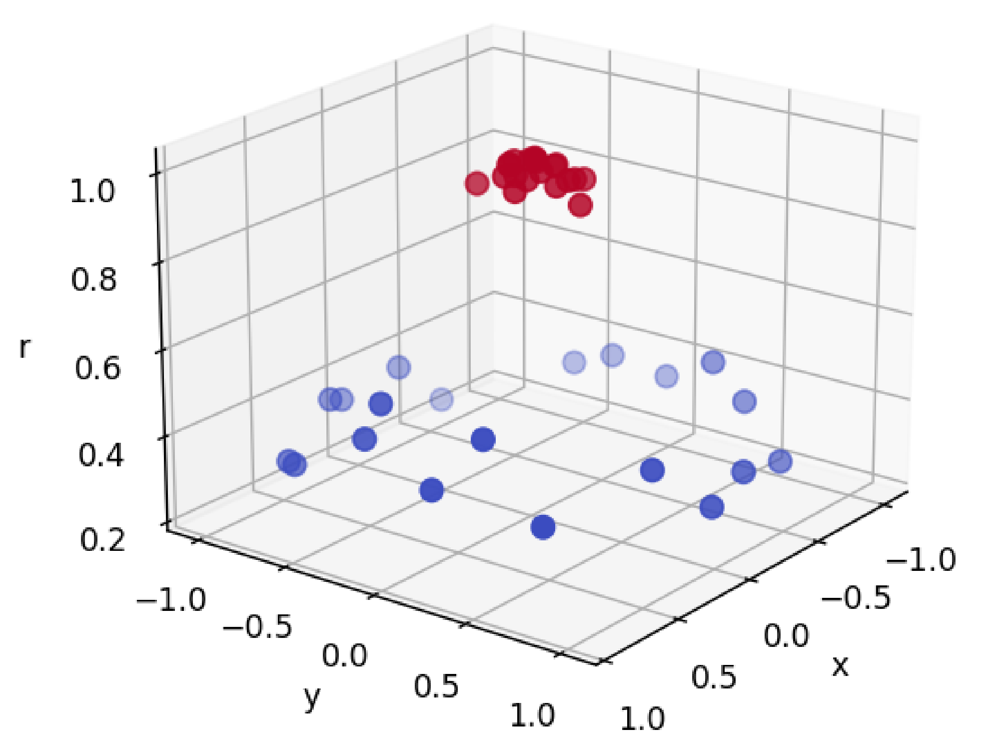


线性不可分的情况

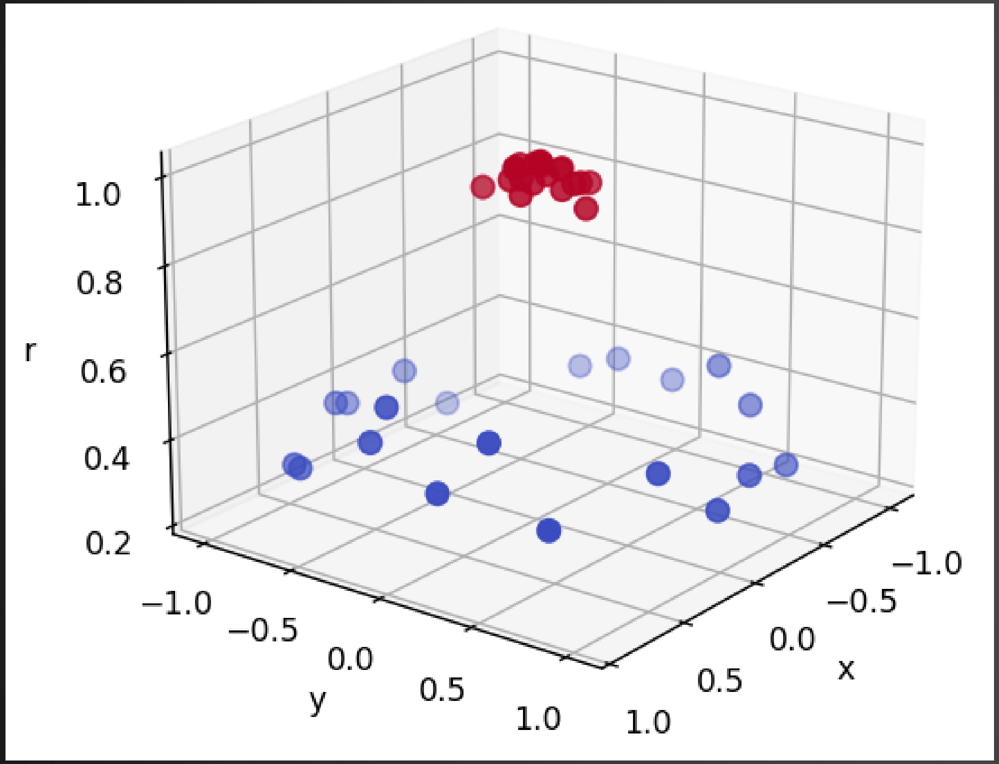
线性不可分？映射到高维空间去！

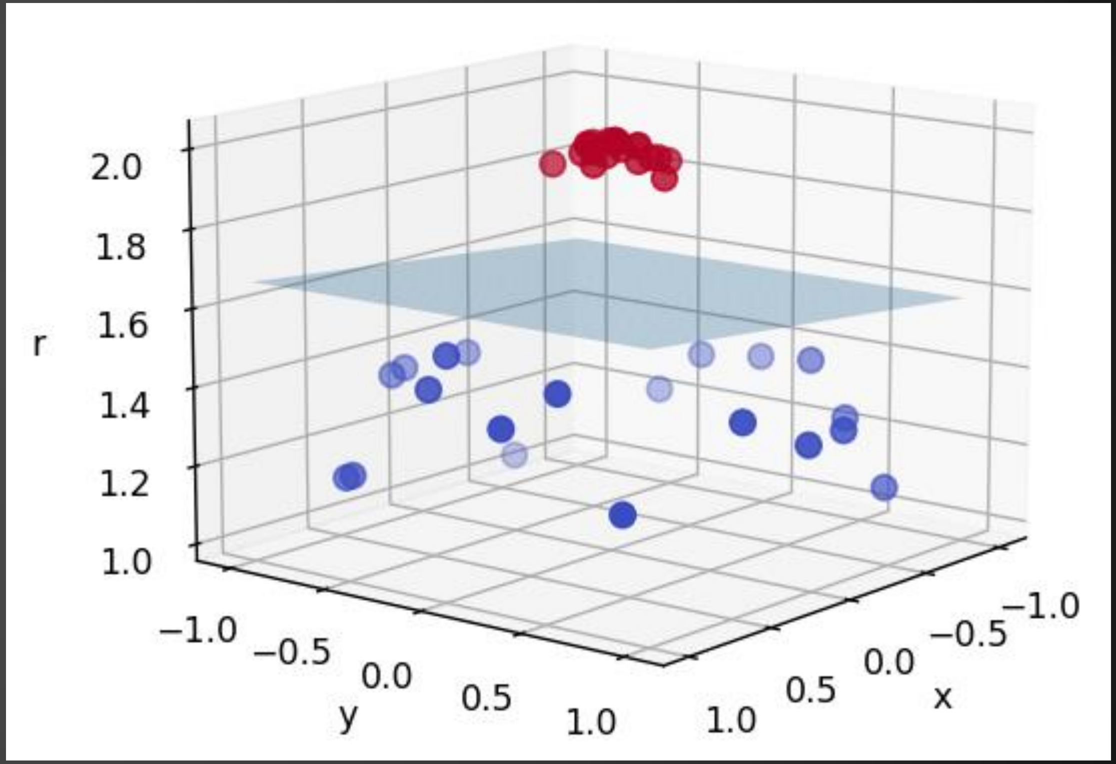
- 通过给数据增加一个特征（维度），将二维空间映射到三维空间





在三维空间中，可以找到一个平面，将这些样本点分开。





从线性可分到不可分

假设找到一个新的函数$\phi(x)$，它能够将原来的样本点x映射到新的高维空间，那么超平面的方程为：
$f(x) = w\phi(x) + b$

此时，非线性SVM的对偶问题为：

$$\max_{\lambda}[\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j(\phi(x_i)\cdot\phi(x_j)) - \sum_{j=1}^{n}\lambda_i]$$

$$s.t.\quad \sum_{i=1}^{n}\lambda_i y_i = 0, \lambda_i \geqq 0, C - \lambda_i - \mu_i = 0$$

可以看到，这个对偶问题的公式跟之前的唯一不同在于：$(x_i \cdot x_j)$变成了$(\phi(x_i)\cdot\phi(x_j))$

说明：对偶问题公式参考前面的对偶推导。

虽然看起来这只是一点点的不同，但是当将样本点映射到高维空间后，计算量会变得很大。

核函数

为了解决计算量的问题，引入核函数。

设想有一个函数$k(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$，这样就将计算$\phi(x_i) \cdot \phi(x_j)$转化为计算函数$k(x_i, x_j)$的值，而不必计算高维空间中$\phi(x_i)$的内积。

这个函数称之为核函数。当然，可以通过数学证明核函数一定存在。

于是，非线性SVM的对偶问题就转化为：

$$\max_{\lambda}[\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j k(x_i, x_j) - \sum_{j=1}^{n}\lambda_i]$$

$$s.t. \quad \sum_{i=1}^{n}\lambda_i y_i = 0, \lambda_i \geqq 0, C - \lambda_i - \mu_i = 0$$

这样，就能够以较小的计算量求解这个非线性SVM优化问题。

高斯核函数

高斯核函数也叫rbf核，其全称为径向基函数（Radius Basic Function），表达式如下。

$$
\begin{aligned}
k(x_i, x_j) &= \exp(-\frac{||x_i - x_j||^2}{2\sigma^2}) \\
&= \exp(-\gamma||x_i - x_j||^2)
\end{aligned}
$$

rbf核表达式中涉及两个样本点的距离计算，距离可以理解为两个样本点的相似程度。

γ是sklearn中的svm库中的另外一个参数，对于参数γ可以这样理解。

- γ越大，意味着两个样本点比较接近时才会被判定为相似，这样决策边界会变得较为扭曲，容易发生过拟合。
- γ越小，意味着两个样本点容易被判定为相似，此时模型较为简单，容易发生欠拟合。

SVM模型中的两个重要参数：参数C和gamma

参数C，控制每个点的重要性

- C越大，说明越不能容忍出现误差，容易过拟合
- C越小，容易欠拟合，C过大或过小，泛化能力变差

默认情况，C=1

参数gamma，控制高斯核宽度的参数

- gamma越大，支持向量越少，模型越复杂
- gamma越小，模型越简单

默认情况，gamma=1 / (n_features * X.var())

总结：由简至繁三类问题

1. 当训练样本线性可分时，通过硬间隔最大化，学习一个线性可分支持向量机；
2. 当训练样本近似线性可分时，通过软间隔最大化，学习一个线性支持向量机；
3. 当训练样本线性不可分时，通过核技巧和软间隔最大化，学习一个非线性支持向量机。

案例：企业员工离职预测

1. 案例介绍
2. 数据读取、认识数据
3. 数据探索及预处理
4. SVM建模
5. 数据缩放
6. 调参：网格搜索

关于案例介绍、数据读取、数据探索及预处理请参考逻辑回归中的讲解。