

## 1. SVM优缺点

优点：

- 1、可以有效解决高维特征的分类和回归问题
- 2、无需依赖全体样本，只依赖支持向量
- 3、有大量的核技巧可以使用，从而可以应对线性不可分
- 4、样本量中等偏小照样有较好的效果

缺点：

- 1、如果特征维度远大于样本个数，SVM表现一般
- 2、SVM在样本巨大且使用核函数时计算量很大
- 3、非线性数据的核函数选择依旧没有标准
- 4、SVM对缺失数据敏感
- 5、特征的多样性导致很少使用svm，因为 svm 本质上是属于一个几何模型，这个模型需要去定义 instance 之间的 kernel 或者 similarity（线性svm中的内积），而我们无法预先设定一个很好的similarity。这样的数学模型使得 svm 更适合去处理 “同性质” 的特征

## SVM参数C的选择

作为当今最为流行的分类算法之一，SVM 已经拥有了不少优秀的实现库，如 `libsvm` 等，因此，我们不再需要自己手动实现 SVM（要知道，一个能用于生产环境的 SVM 模型并非课程中介绍的那么简单）。

在使用这些库时，我们通常需要声明 SVM 需要的两个关键部分：

1. 参数  $C$
2. 核函数 ( Kernel )

由于  $C$  可以看做与正规化参数  $\lambda$  作用相反，则对于  $C$  的调节：

- 低偏差，高方差，即遇到了**过拟合**时：减小  $C$  值。
- 高偏差，低方差，即遇到了**欠拟合**时：增大  $C$  值。

## SVM核函数的选择

- 当特征维度  $n$  较高，而样本规模  $m$  较小时，**不宜使用核函数**，否则容易引起过拟合。
- 当特征维度  $n$  较低，而样本规模  $m$  足够大时，考虑使用**高斯核函数**。不过在使用高斯核函数前，需要进行特征缩放 ( feature scaling )。另外，当核函数的参数  $\delta$  较大时，特征  $f_i$  较为平缓，即各个样本的特征差异变小，此时会造成**欠拟合（高偏差，低方差）**：

简述SVM硬间隔推导过程

简述SVM软间隔推导过程

具体推导请看：

[https://blog.csdn.net/github\\_31101389/article/details/106480409](https://blog.csdn.net/github_31101389/article/details/106480409)

附加题：

BI:

### 3.deepfm的embedding初始化有什么值得注意的地方吗

I 常规的是Xavier，输出和输出可以保持正态分布且方差相近：

- ≡ `np.random.rand(layer[n-1],layer[n])*np.sqrt(1/layer[n-1])`
  - relu的情况下通常是HE，保证半数神经元失活的情况下对输出方差影响最小：  
`np.random.rand(layer[n-1],layer[n])*np.sqrt(2/layer[n-1])`
  - 文本项目上也可以用预训练好的特征

NLP:

CNN 中的 1\*1 卷积有什么作用

[https://blog.csdn.net/sscc\\_learning/article/details/79863922](https://blog.csdn.net/sscc_learning/article/details/79863922)

CV:

### 目标检测里如何有效解决常见的前景少背景多的问题

常用解决样本数据不平衡思路都可。

- b. 硬数据挖掘
- a. 类平衡思路改进过得损失函数