

1. SVM 原理

SVM 是一种二类分类模型。它的基本模型是在特征空间中寻找间隔最大化的分离超平面的线性分类器。

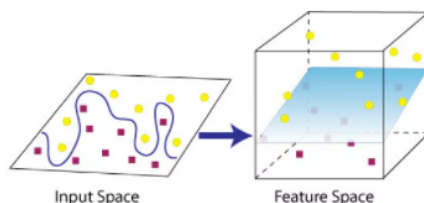
- 当训练样本线性可分时，通过硬间隔最大化，学习一个线性分类器，即线性可分支持向量机；
- 当训练数据近似线性可分时，引入松弛变量，通过软间隔最大化，学习一个线性分类器，即线性支持向量机；
- 当训练数据线性不可分时，通过使用核技巧及软间隔最大化，学习非线性支持向量机。

以上各种情况下的数学推到应当掌握，硬间隔最大化（几何间隔）、学习的对偶问题、软间隔最大化（引入松弛变量）、非线性支持向量机（核技巧）。

2. SVM 为什么采用间隔最大化

当训练数据线性可分时，存在无穷个分离超平面可以将两类数据正确分开。感知机利用误分类最小策略，求得分离超平面，不过此时的解有无穷多个。线性可分支持向量机利用间隔最大化求得最优分离超平面，这时，解是唯一的。另一方面，此时的分离超平面所产生的分类结果是最鲁棒的，对未知实例的泛化能力最强。可以借此机会阐述一下几何间隔以及函数间隔的关系。

4. 为什么 SVM 要引入核函数



当样本在原始空间线性不可分时，可将样本从原始空间映射到一个更高维的特征空间，使得样本在这个特征空间内线性可分。而引入这样的映射后，所要求解的对偶问题的求解中，无需求解真正的映射函数，而只需要知道其核函数。核函数的定义： $K(x,y)=\langle \phi(x),\phi(y) \rangle$ ，即在特征空间的内积等于它们在原始样本空间中通过核函数 K 计算的结果。一方面数据变成了高维空间中线性可分的数据，另一方面不需要求解具体的映射函数，只需要给定具体的核函数即可，这样使得求解的难度大大降低。

6. SVM 核函数之间的区别

一般选择线性核和高斯核，也就是线性核与 RBF 核。线性核：主要用于线性可分的情形，参数少，速度快，对于一般数据，分类效果已经很理想了。RBF 核：主要用于线性不可分的情形，参数多，分类结果非常依赖于参数。有很多人是通过训练数据的交叉验证来寻找合适的参数，不过这个过程比较耗时。如果 Feature 的数量很大，跟样本数量差不多，这时候选用线性核的 SVM。如果 Feature 的数量比较小，样本数量一般，不算大也不算小，选用高斯核的 SVM。

5. 为什么SVM对缺失数据敏感

这里说的缺失数据是指缺失某些特征数据，向量数据不完整。SVM 没有处理缺失值的策略。而 SVM 希望样本在特征空间中线性可分，所以特征空间的好坏对SVM的性能很重要。缺失特征数据将影响训练结果的好坏。

附加题:

CV:

1.常见的模型加速方法

≡ 模型加速通常是指模型训练完成后，对模型推理速度得提高。

常用思路有：

- 网络剪枝系列思路 | 模型量化系列思路

常用工具：

- tensorTR
- Tensor Comprehension | Distiller

NLP:

Self-attention对比RNN和CNN在处理NLP任务时分别有哪些优势

1 层计算复杂度

大多数任务中， n 往往小于 d ，所以在大多数情况下，self-attention的计算复杂度更低，对于句长较大的任务，可以使用restricted self-attention(r -centered)来降低计算复杂度

卷积层的复杂度一般高于RNN，大概是乘以 k

2 串行操作最小步数

RNN是在一个序列上的处理是不可以并行的

3 连接两个元素之间的最远路径（长距离依赖的路径长度）

$O(1)$ (self-attention)更好地学习到长距离依赖;

值得注意的是，针对连续卷积，需要 $O(n/k)$ 层卷积，对于空洞卷积需要 $O(\log_k(n))$ 层

同理restricted self-attention需要 $O(n/r)$ 层,降低了计算复杂度的同时，牺牲了长距离依赖的学习

总而言之，对于一般的翻译任务，self-attention的层计算复杂度低，单层内每个操作可并行，并且具有良好的长距离依赖学习能力

BI:

1.DNN与DeepFM之间的区别

DNN是DeepFM中的一个部分，DeepFM多一次特征，多一个FM层的二次交叉特征