时间安排5周

A. 快速排序

Quick Sort，冒泡排序的改进，一趟排序分割两部分，一部分所有数据都比另一部分小，递归地对两部分进行排序，使整个数据变成有序数列。

1. 基准（pivot）：数据集中选择一个元素作为
2. 分区（partition）:所有小于“基准”的元素都移到基准左边，所有大于“基准”的数都移到基准右边
3. 对子集不断重复一二步，直至子集只剩一个元素

伪代码：

```
QOICKSORT(A, p, r)
 if p < r
  q = PARTITION(A,p,r)
  QUICKSORT(A,p,q-1)
  QUICKSORT(A,q+1,r)
```

如何选取基准pivot 时间复杂度（O(nlogn)到O(n^2)）

1. 选取第一个元素作为基准
2. 随机选取一个元素作为基准（避免对有序数排序列达到最坏复杂度）

B. 堆排序

二叉堆：完全二叉树，分为两类：

1. 最大堆：父节点值大于左右孩子值
2. 最小堆：...小于...
3. 插入节点，删除节点的自我调整：一般在堆顶，堆底调整，向上调整与父节点比较，交换；向下调整相反
4. 构建二叉堆：所有非叶子结点依次下沉；不是链式存储，而是顺序存储 child=parent*2+1/+2 (求父节点，(n-1)/2 向下取整)

C.

D.

E.

F.

第一周

本周问题：对于不同的特征该如何进行特征工程？模型评估中不同的指标用在什么场景中

第一章 特征方程

1.1.1 Garbage in, garbage out. 数据和特征往往决定了结果的上界，模型，算法的选择优化则是接近这个上限

特征工程：对原始数据进行一系列工程处理，提炼为特征，作为输入。是表示和展现数据的过程，去除原始数据中的杂志和冗余，设计更高效的特征

1.1.2 数据类型：结构优化数据，每列有清晰的定义，包含数值型，类别型

非结构化数据：文本、图像、音频、视频数据，无法用一个简单数值表示，没有清晰的类别定义，每条数据大小不同

1.1.3 特征归一化：消除量纲影响，使得不同指标之间具有可比性，若不归一化，结果可能倾向数值差别较大的特征。

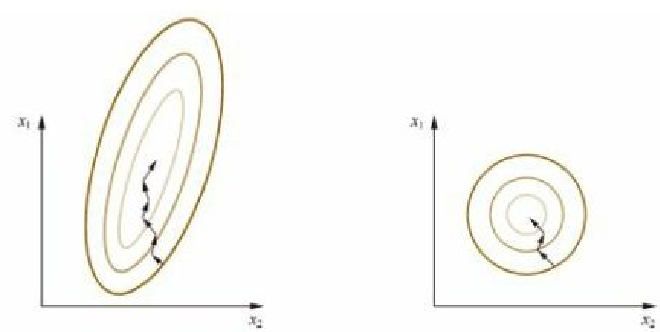
Min-Max Scaling，线性函数归一化，对原始数据进行线性变换，使结果映射到[0,1]的范围，实现等比缩放

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}},$$

Z-Score Normalization，将原始数据映射到均值为0，标准差为1的分布上

$$z = \frac{x - \mu}{\sigma}$$

E.g. 学习速率相同的情况下，当x1, x2更新速率一致时，容易同通过梯度下降找到最优解。



（a）未归一化数据的梯度下降过程　（b）归一化数据的梯度下降过程

Tip: 决策树依赖于信息增益比，与归一化无关，不改变样本在特征x上的信息增益

1.2 类别型特征：对于逻辑回归，支持向量机等模型来说，类别特征必须经过处理转化成数值特征才能正确工作

Ordinal Encoding 序号编码：

One-hot Encoding 独热编码： 处理不具有大小关系的特征，稀疏向量的形式。常配合特征选择来降低维度

Binary Encoding 二进制编码： 使用二进制对类别ID进行哈希映射，维度小

1.3.1 组合特征：把一阶离散特征组合，构成高阶特征
E.g. 推荐系统: 语言、类型 ->是否点击
语言 + 类型 ->是否点击

$$Y = \mathrm{sigmoid}(\sum_i \sum_j w_{ij} < x_i, x_j >)$$

组合特征维度 = 离散特征维度相乘
1.3.2对高维特征的组合，容易存在参数过多，过拟合的问题；并且不是所有的组合特征都是有意义的
降维： 将高维向量使用k维的低维向量表示
拓展：推荐系统矩阵分解
1.3.3寻找特征组合方法
基于决策树的特征组合寻找方法：采用梯度提升寻找决策树，在之前构建的决策树 残差上构建下一棵决策树
1.4 文本表示模型

非结构化数据 -> 表示文本数据
1.4.1 词袋模型 & N-gram模型：将每篇文章看成一袋子词，并忽略每个词出现的顺序，每篇文章可以表示为一个长向量，每一维代表一个单词，权重（常用TF-IDF计算权重）表示这个词在原文中的重要成度

$$\text{TF-IDF}(t,d)=\text{TF}(t,d)\times\text{IDF}(t),$$

$$\text{IDF}(t)=\log\frac{\text{文章总数}}{\text{包含单词}t\text{的文章总数}+1}.$$

IDF(t)为逆文章频率: 如果一个词在非常多的文章中出现，那么它是一个比较通用的词汇
单词好的级别划分有时候并不是一种好的方法，e.g. N, L, P和NLP。所以实际应用中一般可以对单词采用词干抽取（Word Stemming）处理
1.4.2 主题模型
1.4.3 词嵌入：每个词都映射成低维空间（通常500-300维）上的稠密向量 （向量大部分值非0）
1.4.4 深度学习模型：相当于自动地进行特征方程

。。。。缺失一小部分。。。。
1.5 图像数据不足是的处理方法

训练数据不足时，就说明模型从原始数据中获取的信息比较少，这种时候需要更多的先验信息。
作用于模型上：模型采用特定的内在结构、条件假设或者添加约束条件。
作用于数据集上：根据特定的先验假设去调整，变换和拓展训练集，让其展现出更多的等有用的信息。

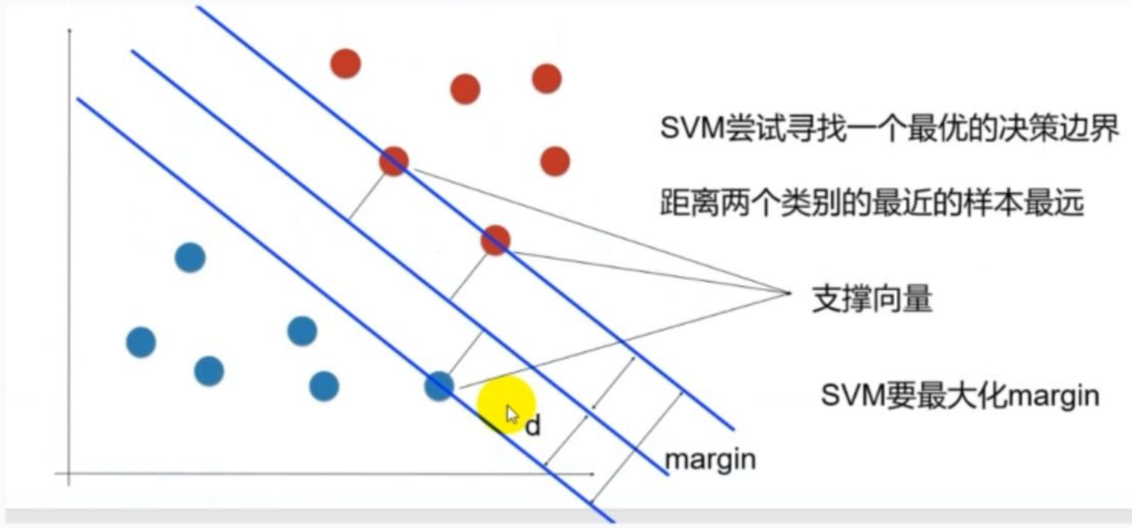
1.5.1 具体到图像分类任务上，训练数据不足带来的问题主要在过拟合方面，在测试集上泛化效果不佳。处理方法主要目的是较少过拟合风险：简化模型（将非线性模型简化为线性）、添加约束项以减少假设空间（L1/L2正则）、集成学习、DropOut超参数等

数据扩充：对原始数据进行适当变化达到扩充数据集的效果
（1）一定程度内的随机旋转、 平移、 缩放、 裁剪、 填充、 左右翻转等
这些变换对应着同一个目标在不同角度的观察结果。
（2） 对图像中的像素添加噪声扰动， 比如椒盐噪声、 高斯白噪声等。
（3） 颜色变换。 例如， 在图像的RGB颜色空间上进行主成分分析， 得到3个主成分的特征向量p_1,p_2,p_3及其对应的特征值 $\lambda_1,\lambda_2,\lambda_3$， 然后在每个像素的RGB值上添加增量$[p_1,p_2,p_3]\cdot[\alpha_1\lambda_1,\alpha_2\lambda_2,\alpha_3\lambda_3]^T$， 其中 $\alpha_1,\alpha_2,\alpha_3$是均值为0、 方差较小的高斯分布随机数

借助已有的其他模型或数据进行迁移学习，借助一个在大数据集上预训练好的通用模型，在小数据集（目标任务)上进行微调

第二章

第二周
在算法面试中，手推SVM已经越来越像快速排序一样，成为了一道必点菜。而SVM的原理的推导实际上需要一定的数学基础，比如拉格朗日乘子法、核函数等。而越往后学习越能发现，这些数学的基础知识在很多地方都会用到，故而学习SVM的原理以及不仅仅是对SVM这个算法本身的学习了，它更像是一种基础。我们建议每位同学在学完本章后，都应当能够在一张白纸上推导出SVM的算法全过程。在本章我们也提供了一些SVM推导的小视频，以供大家学习。
2.1.1 线性可分：两类点被一条直线完全分开
2.2.1 SVM名词： 分类面， 核映射
2.2.2 超平面：最大间隔把两类样本分开，就是最佳超平面
最大间隔超平面（最佳超平面）：两类样本分割在超平面两侧。两侧距离超平面最近的样本点 到超平面距离最大化



在蓝色样本中存在一些距离我们的**超平面**最近的一些点，这些点叫做支撑向量。

2.1.3 支持向量：SVM尝试寻找最优决策边界，距离两个类别的最近样本最远

2.1.2 SVM 最优化问题推导

N维上，点x到直线距离

$$\frac{|w^T x + b|}{\|w\|}$$

根据SVM定义

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

稍作转化可以得到：

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

然后能够得到：

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

此处w,b为原w,b 乘以方程左边的底；y意义为标签值，取-1， 1方便计算

对于上面这个方程我们还能简写一下：

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

所以我们可以得到上下两个超平面的方程为：

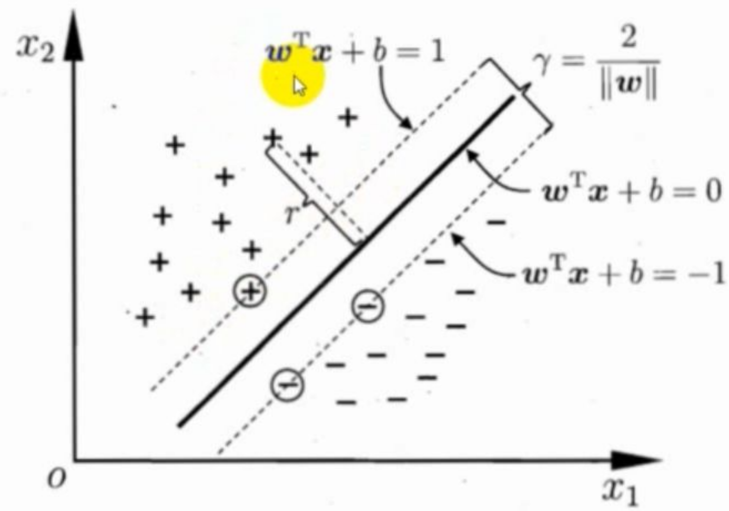


图 6.2　支持向量与间隔

并且每个支撑向量到超平面的距离可以写为：

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

我们要最大化这个距离也就是

$$max \frac{|w^T x_i + b|}{||w||}$$

在样本点确定以后，$|w^T x_i + b|$是一个常数，所以这个式子就变成了：

$$max \frac{1}{||w||}$$

也就是：

$$min||w||$$

为了计算方便，我们取：

$$min \frac{1}{2} ||w||^2$$

所以得到最后的优化问题是：

$$min \frac{1}{2} ||w||^2$$

$$s.t.\ \ y_i(w^T x_i + b) \geq 1$$