无监督学习

1. 聚类

   纵向结构,根据数据的特征,将样本分为几类

2. 特征学习

   横向结构,根据样本的特征,将样本的特征维度映射到另一个空间

3. 密度估计

   就是找出数据从哪个分布中采样的,找出那个分布.

监督学习就是建立一种映射关系:$x \rightarrow y$

无监督学习就是从无标签的数据中学习出一些有用的模式.

举例:监督学习就是将物品分类,如这个水果是苹果吗? 答案会是Yes/No.而无监督学习就是将水果根据其形状等特征归类成几类,比如弄成一摞,一摞里全是苹果.

聚类

将样本集合中相似的样本分配到相同的类/簇(cluster),不相似的样本分配到不同的类/族,使得类内样本间距较小而类间间距较大.

核心概念

样本间距离/相似性

1. L_1, L_2 距离
2. 余弦距离
3. 相关系数
4. 汉明距离

常见的聚类任务

1. 图像分割
2. 文本聚类
3. 社交网络分析

类/簇

类没有一个严格的定义,可以理解为一组相似的样本.

类内间距

🔴样本间平均距离:

$$avg(C) = \frac{2}{|C|(|C|-1)} \sum_{1 \le i \le j \le |C|} d_{ij} \tag{1}$$

$|C|$为类里面的样本数

🔴样本间最大距离(直径,diamete r)

$$dia(C) = \max_{1 \le i \le j \le |C|} d_{ij} \tag{2}$$

类间间距

🔴样本间最短距离

$$D_{pq} = \min\{d_{ij} | x^{(i)} \in C_p, x^{(j)} \in C_q\} \tag{3}$$

🔴样本均值间距离

$$D_{pq} = d_{\mu_p \mu_q} \tag{4}$$

聚类方法

常见聚类方法:

- K均值聚类
- 层次聚类
- 密度聚类
- 谱聚类

聚类效果评估

📕外部指标:就是有y的,但是没给.

- Jaccard指数

$$JC = \frac{TP}{TP + FN + FP} \tag{5}$$

- FM指数
- Rand指数

构建新型的混淆矩阵,该混淆矩阵的数据点为数据对$(x^{(i)}, x^{(j)})$共有$\frac{N(N-1)}{2}$对,.

混淆矩阵		预测结果	
		同类	不同类
真实结果	同类	#TP	#FN
	不同类	#FP	#TN

数据集中每组数据$(x^{(i)}, x^{(j)})$
在预测/参考中结果

内部指标:没有外部参考

1. DM指数

$$DBI = \frac{1}{k}\sum_{i=1}^{k}\max_{j\neq i}\left(\frac{avg(C_i)+avg(C_j)}{d_{\mu_i\mu_j}}\right) \tag{6}$$

分子两个是类内间距,分母是类间间距. 这个指数越小越好.

2. Dunn指数

K-means

1. 确定k,k是我们类别的数量.
2. 随机生成k个类中心.
3. 将空间中的点计算到k个中心的距离.然后落在不同区域的点就属于不同的类.
4. 于是我们就对数据进行了一个划分.有了这些划分之后,重新计算每个类的类中心.
5. 再根据新的类中心对这个空间进行划分.
6. 不断迭代,直至收敛(每个点的类别不在发生变化.)

目标函数

$$E = \sum_{i=1}^{k}\sum_{x\in C_i}||x-\mu_i||^2 \tag{7}$$

其中μ_i是第i个簇C_i的均值向量.

E值刻画了簇内样本围绕簇均值向量的紧密程度.

收敛性

NP Hard问题，迭代优化：

- 固定均值向量，优化分布
- 固定划分，优化均值向量

k的选择

只要目标函数的降低不开始剧烈抖动时候的k就可以了.

类中心初始化

- 大于最小间距的随机点/样本点
- K个相互距离最远的样本点
- K个等距的网格点

优点

- 实现简单
- 时间复杂度低 $O(N)$

缺点

- K值选择
- 主要适合凸集
- 初始值影响较大.

层次聚类

层次聚类通过计算不同类别数据点间的相似度来创建一棵有层次的嵌套聚类树.

- 聚合:自底向上

- 分裂:自顶向下.

聚合

1. 将每个样本分到单独的类
2. 不断迭代下面的过程,直至满足终止条件
    - 计算两两类簇之间的距离,找到距离最小的两个类簇 c_1 和 c_2
    - 合并类簇 c_1 和 c_2 为一个类簇.

分裂

1. 将所有样本分到同一类
2. 不断迭代下面过程,直至满足终止条件
    - 在同一个类簇(记为c)中计算两两样本之间的距离,找出距离最远的两个样本a,b.
    - 将样本a,b分配到不同的类簇 c_1 和 c_2 中
    - 计算原类簇(c)中剩余的其他样本点和a,b的距离,若是dis(a)<dis(b),则将样本点归到 c_1 中,否则归为 c_2 中.

层次聚类的优缺点

- 简单,容易理解
- 合并点/分裂点选择不太容易
- 合并/分类的操作不能进行撤销
- 大数据集不太合适
- 执行效率较低 $O(TN^2)$,T为迭代次数,N为样本点个数

特征学习

目的

- 降维数据可视化
- 稀疏编码

主成分分析 **PCA**

数据的原始特征可能存在的问题:

- 高维 $\rightarrow$ 维度灾难,过拟合
- 冗余性 $\rightarrow$ 学习效果差
- 解决方法:降维

一种最常用的数据降维方法

- 线性投影

$$z = W^T x \tag{8}$$

-
- 满足

$$W^T W = I \tag{9}$$

优化准则

最大投影方差:使得在转换后的空间中的数据的方差最大,尽可能多的保留原数据的信息

最小重构误差

具体做法

- 样本点 $x^{(n)}$ 投影之后的表示为

$$z^{(n)} = w^T x^{(n)} \tag{10}$$

- 所有的样本投影后的方差为:

$$\begin{aligned} \sigma(X;w) &= \frac{1}{N}\sum_{n=1}^{N}(w^T x^{(n)} - w^T \bar{x})^2 \\ &= \frac{1}{N}(w^T X - w^T \bar{X})(w^T X - w\bar{X})^T \\ &= w^T \Sigma w \end{aligned} \tag{11}$$

- 目标函数

$$\max_w w^T \Sigma w + \lambda(1 - w^T w) \tag{12}$$

- 对目标函数求导并令导数等于0,可得

$$\Sigma w = \lambda w \tag{13}$$

编码

给定一组基向量$A = [a_1, a_2, \cdots, a_M]$,将输入样本$x$表示为这些基向量的线性组合

$$\begin{aligned} x &= \sum_{m=1}^{M} z_m a_m \\ &= AZ \end{aligned} \tag{14}$$

两个参数都是要学习的,最终是用z表示x.A是对所有样本学习,z是对于单个样本学习.

稀疏编码

过完备

如果M个基向量刚好可以支撑M维的欧式空间,则这M个基向量是完备的.如果M个基向量可以支撑D维的欧式空间,$M > D$,则这M个基向量是过完备的、冗余的.

“过完备”基向量是指基向量个数远远大于其支撑空间维度,因此这些基向量一般不具备独立,正交等性质

稀疏编码

找到一组“过完备”的基向量(即M>D)来进行编码.

具体做法

给定一组N个输入向量$x^{(1)},\cdots,x^{(N)}$,其稀疏编码的目标函数定义为

$$\mathcal{L}(A,Z)=\sum_{n=1}^{N}(||x^{(n)}-Az^{(n)}||^2+\eta\rho(z^{(n)})) \tag{15}$$

其中$\rho(\cdot)$是一个稀疏性衡量函数,η是一个超参数,用来控制稀疏性的强度.

稀疏性衡量函数	说明
$\rho(z)=\sum_{i=1}^{p}I(\|z_i\|>0)$	不是连续可导
$\rho(z)=\sum_{i=1}^{p}\|z_i\|$	常用
$\rho(z)=\sum_{i=1}^{p}\log(1+z_i^2)$	常用
$\rho(z)=\sum_{i=1}^{p}-\exp(-z_i^2)$	常用

交替优化

1. 固定基向量 A ,对于每个输入 $x^{(n)}$,计算其对应的最优编码

$$\min_{z^{(n)}}||x^{(n)}-Az^{(n)}||^2+\eta\rho(z^{(n)}),\forall n\in[1,N] \tag{16}$$

2. 固定上一步得到的编码 $\{z^{(n)}\}_{n=1}^{N}$,计算最优的基向量

$$\min_{A}\sum_{n=1}^{N}(||x^{(n)}-Az^{(n)}||^2)_\lambda\frac{1}{2}||A||^2 \tag{17}$$

稀疏编码的优点

1. 降低后续计算
2. 可解释性强
3. 便于特征选择

自编码器

目标函数:重构误差

$$\begin{aligned}\mathcal{L} &= \sum_{i=1}^{N} ||x^{(n)} - g(f(x^{(n)}))||^2 \\ &= \sum_{i=1}^{N} ||x^{(n)} - f \circ g(x^{n})||^2\end{aligned} \tag{18}$$

稀疏自编码器

通过给自编码器中隐藏层单元z加上稀疏性限制,自编码器可以学习到数据中一些有用的结构.

目标函数:

$$\mathcal{L} = \sum_{i=1}^{N} ||x^{(n)} - x'^{(n)}||^2 + \eta\rho(Z) + \lambda||W||^2 \tag{19}$$

W表示自编码器中的参数.

和稀疏编码一样,稀疏自编码器的优点是有很高的可解释性,并同时进行了隐式的特征选择.

降噪自编码器

通过引入噪声来增加编码鲁棒性的自编码器.

1. 对于一个向量 x ,我们首先根据一个比例 μ 随机将 x 的一些维度的值设置为0,得到一个被损失的向量 $\widetilde{x}$.
2. 然后将被损坏的向量 $\widetilde{x}$ 输入给自编码器得到编码 z ,并重构出原始的无损输入x.

自监督学习

不再只是以输入重构作为目标,在无标签样本自身寻找更多样的“目标”

🍎图形任务中的自监督学习

旋转角度预测

人为的构造旋转角度,构建y,进行预测建模.

🍎文本任务中的自监督学习

掩码语言模型

就是对一句话中的某个词进行掩码,然后用其他的词来预测这个位置的词.

概率密度估计

- 参数密度估计

根据先验知识假设随机变量服从某种分布,然后通过训练样本来估计分布的参数据

估计方法:最大似然估计

$$\log p(\mathcal{D};\theta) = \sum_{n=1}^{N} \log p(x^{(n)};\theta) \tag{20}$$

- 非参数密度估计

不假设数据服从某种分布,通过将样本空间划分为不同的区域并估计每个区域的概率来近似数据的概率密度函数.

📖参数密度估计存在的问题

1. 模型选择问题(如何确定数据分布)
2. 不可观测变量问题(很难准确估计数据的真实分布)
3. 维度灾难问题(随着维度增加,估计参数所需要的样本数量指数级增长)

📖非参数密度估计

- 对于高维空间中的一个随机向量 x ,假设其服从一个未知分布 $p(x)$,则 x 落入空间中的小区域 $\mathcal{R}$ 的概率为

$$P = \int_{\mathcal{R}} p(x)dx \tag{21}$$

- 给定 N 个训练样本 $D = \{x^{(n)}\}_{i=1}^{N}$,落入区域 R 的样本数量 K 服从二项分布

$$P_K = \binom{N}{K} P^K (1-P)^{1-K} \tag{22}$$

当N非常大时,我们可以近似认为

$$P \approx \frac{K}{N} \tag{23}$$

假设区域R足够小,其内部的概率密度是相同的,则有

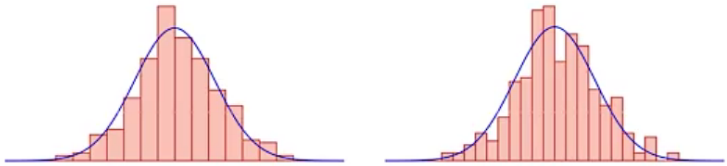
$$P \approx p(x)V \tag{24}$$

- 综合上述两个公式

$$p(x) \approx \frac{K}{NV} \tag{25}$$

直方图方法

一种非常直观的估计连续变量密度函数的方法,可以表示为一种柱状图.



(a) 10个区间（bin）　　(b) 30个区间（bin）

以一维随机变量为例，首先将其取值范围分成M个连续的、不重叠的区间（bin），每个区间的宽度为Δ_m. 给定N个训练样本$\mathcal{D}=\{x^{(n)}\}_{n=1}^{N}$，我们统计这些样本落入每个区间的数量$K_m$，然后将它们归一化为密度函数.

核密度估计

核密度估计是一种直方图方法的改进(也叫Parzen窗方法)

- 假设$\mathcal{R}$是d维空间中的一个以点x为中心的“超立方体”,并定义 核函数 来表示一个样本z是否落入该超立方体中

$$\phi(\frac{z-x}{H})=\begin{cases}1, & if|z_i-x_i|<\frac{H}{2},1\leq i\leq D\\ 0, & else\end{cases} \tag{26}$$

- 点x的密度估计为

$$p(x)=\frac{K}{NH^D}=\frac{1}{NH^D}\sum_{n=1}^{N}\phi(\frac{x^{(n)}-x}{H}) \tag{27}$$

K近邻方法

要估计点x的密度,首先找到一个以x为中心的球体,使得落入球体的样本数量为K,就可以计算出点x的密度.

非参数密度估计的缺点

- 非参数密度估计需要保留整个训练集
- 而参数密度估计不需要保留整个训练集,因此在存储和计算上更加高效.

半监督学习

半监督学习

- 自训练(self-training)

一部分有标签的数据,一部分没标签的数据进行训练.

Step1: 先用有标签的数据进行训练,得出一个模型

Step2: 用训练的模型对没标签的数据进行预测.

Step3:将预测的数据和原先训练的数据组成一个新的训练集

Step4:用新的数据集在训练.

Step5:不断迭代.