文本数据处理与KNN

第一周实验课　　冯禹豪

目录

中山大學 SUN YAT-SEN UNIVERSITY

中山大學
SUN YAT-SEN UNIVERSITY

1 实验课程要求

1 实验课程要求

实验课程内容：

- 由助教讲解实验内容
- 验收前一次的实验内容（包括公式推导、代码解释、现场运行代码产生结果）

实验课程要求：

- 实验需要一定的数学基础以及编程基础（公式的推导以及代码的实现）
- 禁止抄袭（代码和实验报告都禁止抄袭，若被发现后果严重）

中山大學
SUN YAT-SEN UNIVERSITY

2 文本数据处理

1. 分词。（针对中文有字级别，词级别，针对英文有字符级别，词级别，子词级别等）
2. 去停用词。（如‘i’, ‘me’, ‘my’, ‘there’, ‘when’, ‘where’等等）
3. 建立词表。如果词表过大，可以通过设定词语出现的频次的阈值来缩减词表大小。
4. 对文档中的词语进行编码。

文本数据处理—前三步

文档1：很不错的一个酒店，很舒服，服务态度很好。
文档2：酒店服务很热情，希望下次来还有这种服务。
文档3：苹果手机挺不错的。

分词

文档1：很/不错/的/一个/酒店，很/舒服，服务/态度/很好/。
文档2：酒店/服务/很/热情/，/希望/下次/来/还有/这种/服务/。
文档3：苹果/手机/挺/不错/的。

去停用词

文档1：不错 酒店 舒服 服务 态度 很好
文档2：酒店 服务 热情 希望 服务
文档3：苹果 手机 不错

建词表

词表：
不错 酒店 舒服 服务 态度 很好 热情 希望 苹果 手机

文本数据处理—编码

为什么需要对文本进行编码?

不像在计算机视觉领域中，图像其实就是多个像素点构成，像素值之间是可计算的。而一般文本很难进行直接计算的（也有可以计算的场景，如？），所以我们需要对文本中的词语进行编码。一般有如下几种编码方式：

- One-hot编码
- 词频表示
- 词频归一化后的概率表示（tf）
- 逆向文档频率（idf）
- TF-IDF表示

……

文本数据处理—One-hot编码

One-hot编码：文档中每一个词都是一个V维的向量（V是词表大小），其中向量中只有对应词表的位置是1，其余都是0。

文档1：不错 酒店 舒服 服务 态度 很好

词表：不错 酒店 舒服 服务 态度 很好 热情 希望 苹果 手机

文档1:

不错 [1, 0, 0, 0, 0, 0, 0, 0, 0, 0]

酒店 [0, 1, 0, 0, 0, 0, 0, 0, 0, 0]

舒服 [0, 0, 1, 0, 0, 0, 0, 0, 0, 0]

服务 [0, 0, 0, 1, 0, 0, 0, 0, 0, 0]

……

文本数据处理—词频表示

词频表示：每一个文档都是一个V维的向量，其中每一维的值对应词表的位置上该词语出现的次数。

文档1：不错 酒店 舒服 服务 态度 很好

文档2：酒店 服务 热情 希望 服务

文档3：苹果 手机 不错

	不错	酒店	舒服	服务	态度	很好	热情	希望	苹果	手机
文档1	1	1	1	1	1	1	0	0	0	0
文档2	0	1	0	2	0	0	1	1	0	0
文档3	1	0	0	0	0	0	0	0	1	1

文本数据处理—词频归一化后的概率表示

词频归一化后的概率表示：也叫term frequency，是指每个文档的词频归一化后的概率。

$$tf_{i,d} = \frac{n_{i,d}}{\sum_v n_{v,d}}$$

文档1：不错 酒店 舒服 服务 态度 很好

文档2：酒店 服务 热情 希望 服务

文档3：苹果 手机 不错

	不错	酒店	舒服	服务	态度	很好	热情	希望	苹果	手机
文档1	1/6	1/6	1/6	1/6	1/6	1/6	0	0	0	0
文档2	0	1/5	0	2/5	0	0	1/5	1/5	0	0
文档3	1/3	0	0	0	0	0	0	0	1/3	1/3

文本数据处理—逆向文档频率

逆向文档频率(inverse document frequency)，是一个词语普遍重要性的度量。假设总共有|C|篇文档，第i个词在|Ci|篇文档中出现：

$$idf_i = \log \frac{|C|}{|C_i|}$$

文档1：不错 酒店 舒服 服务 态度 很好

文档2：酒店 服务 热情 希望 服务

文档3：苹果 手机 不错

	不错	酒店	舒服	服务	态度	很好	热情	希望	苹果	手机
idf值	log(3/3)	log(3/3)	log(3/2)	log(3/3)	log(3/2)	log(3/2)	log(3/2)	log(3/2)	log(3/2)	log(3/2)

文本数据处理—TF-IDF

TF-IDF(term frequency - inverse document frequency)：TF*IDF，可以把IDF理解为TF的一个权重值。

$$tf\text{-}idf_{i,j} = \frac{n_{i,j}}{\sum_{k=1}^{V} n_{i,k}} \times \lg \frac{|D|}{1+|D_j|}$$

	不错	酒店	舒服	服务	态度	很好	热情	希望	苹果	手机
idf值	log(3/3)	log(3/3)	log(3/2)	log(3/3)	log(3/2)	log(3/2)	log(3/2)	log(3/2)	log(3/2)	log(3/2)

	不错	酒店	舒服	服务	态度	很好	热情	希望	苹果	手机
文档1	1/6	1/6	1/6	1/6	1/6	1/6	0	0	0	0
文档2	0	1/5	0	2/5	0	0	1/5	1/5	0	0
文档3	1/3	0	0	0	0	0	0	0	1/3	1/3

中山大學
SUN YAT-SEN UNIVERSITY

3 KNN算法

3 KNN算法

- 分类问题：预测离散值的问题，例如预测明天是否下雨。
- 回归问题：预测连续值的问题，例如预测明天气温是多少度。

在KNN算法中，针对分类问题，对相似度topK样本的标签，采用多数投票原则

Document number	The sentence words	emotion
train 1	I buy an apple phone	happy
train 2	I eat the big apple	happy
train 3	The apple products are too expensive	sadnesss
test 1	My friend has an apple	?

3 KNN算法

- 分类问题：预测离散值的问题，例如预测明天是否下雨。
- 回归问题：预测连续值的问题，例如预测明天气温是多少度。

在KNN算法中，针对回归问题，相似度topK的样本，根据相似度来进行加权。

Document number	The sentence words	the probability of happy
train 1	I buy an apple phone	0.8
train 2	I eat the big apple	0.6
train 3	The apple products are too expensive	0.1
test 1	My friend has an apple	?

用什么来衡量相似度？距离

Lp距离：

$$L_p(x_i, x_j) = \left(\sum_{l=1}^{n} |x_i^{(l)} - x_j^{(l)}|^p \right)^{\frac{1}{p}}$$

P=1为曼哈顿距离：

$$L_1(x_i, x_j) = \sum_{l=1}^{n} |x_i^{(l)} - x_j^{(l)}|$$

P=2为欧式距离：

$$L_2(x_i, x_j) = \left(\sum_{l=1}^{n} |x_i^{(l)} - x_j^{(l)}|^2 \right)^{\frac{1}{2}}$$

余弦相似度：

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}.$$

注意相似度和距离的一些区别：相似度值越大表示越相似，而距离的值越大表示越不相似。所以有余弦距离=1-余弦相似度

KNN分类步骤

原数据：

Document number	The sentence words	emotion
train 1	I buy an apple phone	happy
train 2	I eat the big apple	happy
train 3	The apple products are too expensive	sadnesss
test 1	My friend has an apple	?

编码：

Document number	I	buy	an	apple	...	friend	has	emotion
train 1	1	1	1	1	...	0	0	happy
train 2	1	0	0	1	...	0	0	happy
train 3	0	0	0	1	...	0	0	sadness
test 1	0	0	1	1	...	1	1	?

KNN分类步骤

计算test与每一个train的样本的距离，这里使用欧式距离，也可以使用别的距离：

$$d(train1, test1) = \sqrt{(1-0)^2 + (1-0)^2 + \cdots + (0-1)^2} = \sqrt{6};$$

$$d(train2, test1) = \sqrt{(1-0)^2 + (1-0)^2 + \cdots + (0-1)^2} = \sqrt{8};$$

$$d(train3, test1) = \sqrt{(0-0)^2 + (0-0)^2 + \cdots + (0-1)^2} = \sqrt{9};$$

若K=1，则test1的标签是train1的标签；若K=3，则多数投票可以得到test1的标签是happy。

KNN回归步骤

原数据：

Document number	The sentence words	the probability of happy
train 1	I buy an apple phone	0.8
train 2	I eat the big apple	0.6
train 3	The apple products are too expensive	0.1
test 1	My friend has an apple	?

编码：

Document number	I	buy	an	apple	...	friend	has	probability
train 1	1	1	1	1	...	0	0	0.8
train 2	1	0	0	1	...	0	0	0.6
train 3	0	0	0	1	...	0	0	0.1
test 1	0	0	1	1	...	1	1	?

KNN回归步骤

计算test与每一个train的样本的距离，然后每一种标签的概率就是由test样本与topK的样本的距离**倒数**作为权重，乘以topK样本该标签的概率：

$$P(test1\ is\ happy) = \frac{train1\ probability}{d(train1, test1)} + \frac{train2\ probability}{d(train2, test1)} + \frac{train3\ probability}{d(train3, test1)}$$

思考：为什么是倒数？如果要求得到的每一种标签的概率的和等于1，应该怎么处理？

回归问题指标—相关系数

相关系数是研究变量之间线性相关程度的量。例如[1,2,4]和[2,4,8]用相关系数可以求得相关程度是很高的。在回归问题的应用场景下用于计算实际概率向量以及预测概率向量之间的相似度。具体公式如下：

$$COR(X,Y)=\frac{cov(X,Y)}{\sigma_X\sigma_Y}=\frac{\sum_{i=1}^{n}(Xi-\overline{X})(Yi-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(Xi-\overline{X})^2\sum_{i=1}^{n}(Yi-\overline{Y})^2}}$$

在情感分类问题中，我们在测试集上有所有文档预测得到的概率值，也有真实的概率值。先分别计算六个维度上的真实概率值和预测概率值的相关系数，然后对六个维度取平均计算得到最终相关系数。

KNN参数设置

影响KNN的参数主要是**K的取值、距离度量方式、权重归一化**等。

- 针对K值，K值取的过大，表示学习的参考样本更多，会引入更多的噪音，所以可能存在欠拟合的情况；如果K值取的过小，参考样本少，容易出现过拟合的情况。一般K的取值可以考虑$\sqrt{N}$，其中N是训练样本的样本数。
- 不同距离度量方式产生的结果会不一样。（下一页会有例子）
- 权重归一化：

Name	Formula	Explain
Standard score	$X' = \frac{X-\mu}{\sigma}$	μ is the mean and σ is the standard deviation
Feature scaling	$X' = \frac{X-X_{min}}{X_{max}-X_{min}}$	X_{min} is the min value and X_{max} is the max value

不同距离度量方式的影响

例 3.1 已知二维空间的 3 个点 $x_1 = (1,1)^{\mathrm{T}}$, $x_2 = (5,1)^{\mathrm{T}}$, $x_3 = (4,4)^{\mathrm{T}}$，试求在 p 取不同值时，L_p 距离下 x_1 的最近邻点。

解 因为 x_1 和 x_2 只有第一维的值不同，所以 p 为任何值时，$L_p(x_1, x_2) = 4$。而

$$L_1(x_1, x_3) = 6, \quad L_2(x_1, x_3) = 4.24, \quad L_3(x_1, x_3) = 3.78, \quad L_4(x_1, x_3) = 3.57$$

于是得到：p 等于 1 或 2 时，x_2 是 x_1 的最近邻点；p 大于等于 3 时，x_3 是 x_1 的最近邻点。 ■

KNN算法的效率

假设训练集有N个样本，测试集有M个样本，每个样本是一个V维的向量。如果使用线性搜索的话，那么KNN的时间花销就是O(N*M*V)。

怎么加速呢?

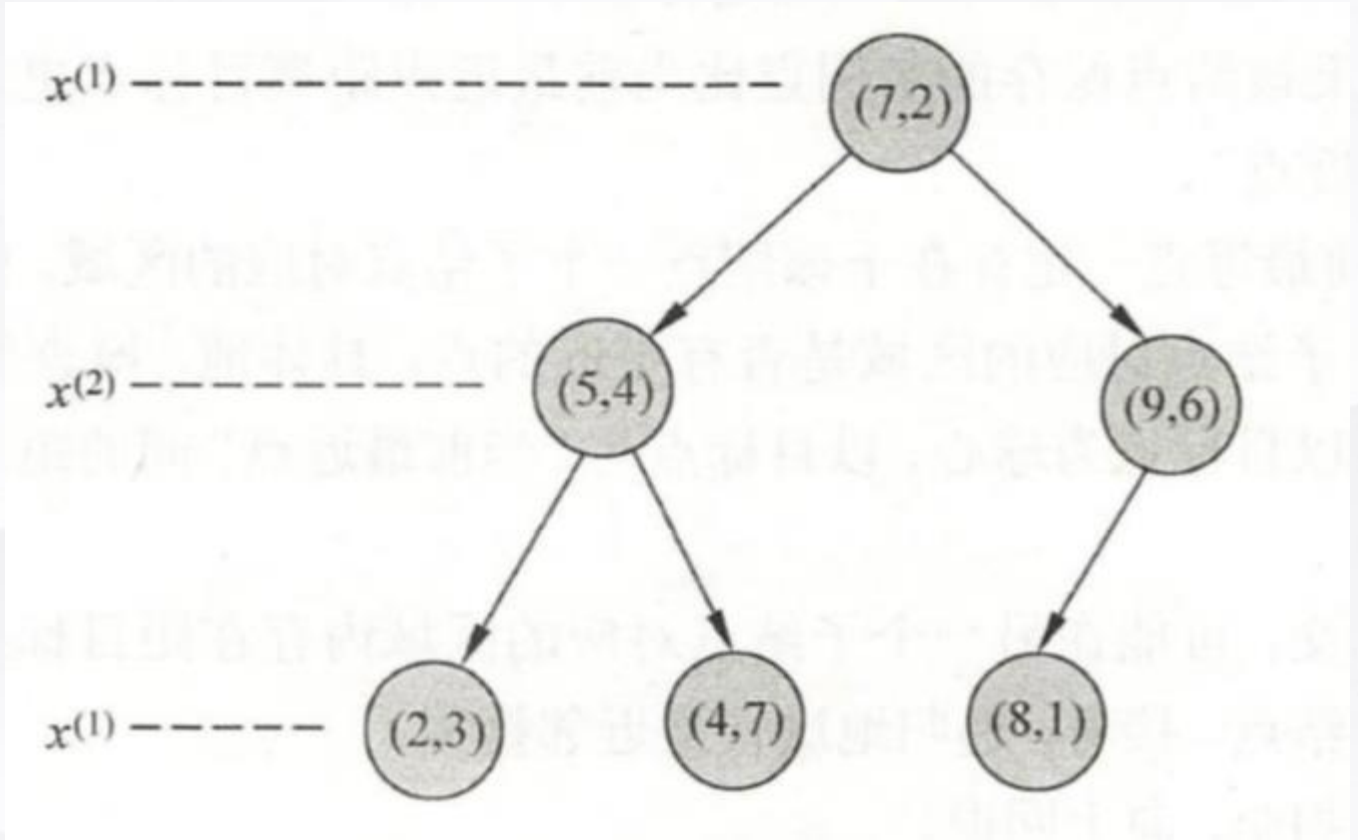
KNN算法的加速—kd树的构建

kd(k-dimensional)树是一种对k维空间中的实例点进行二叉树状的存储以便进行快速检索的树形结构。建造kd树的流程如下：

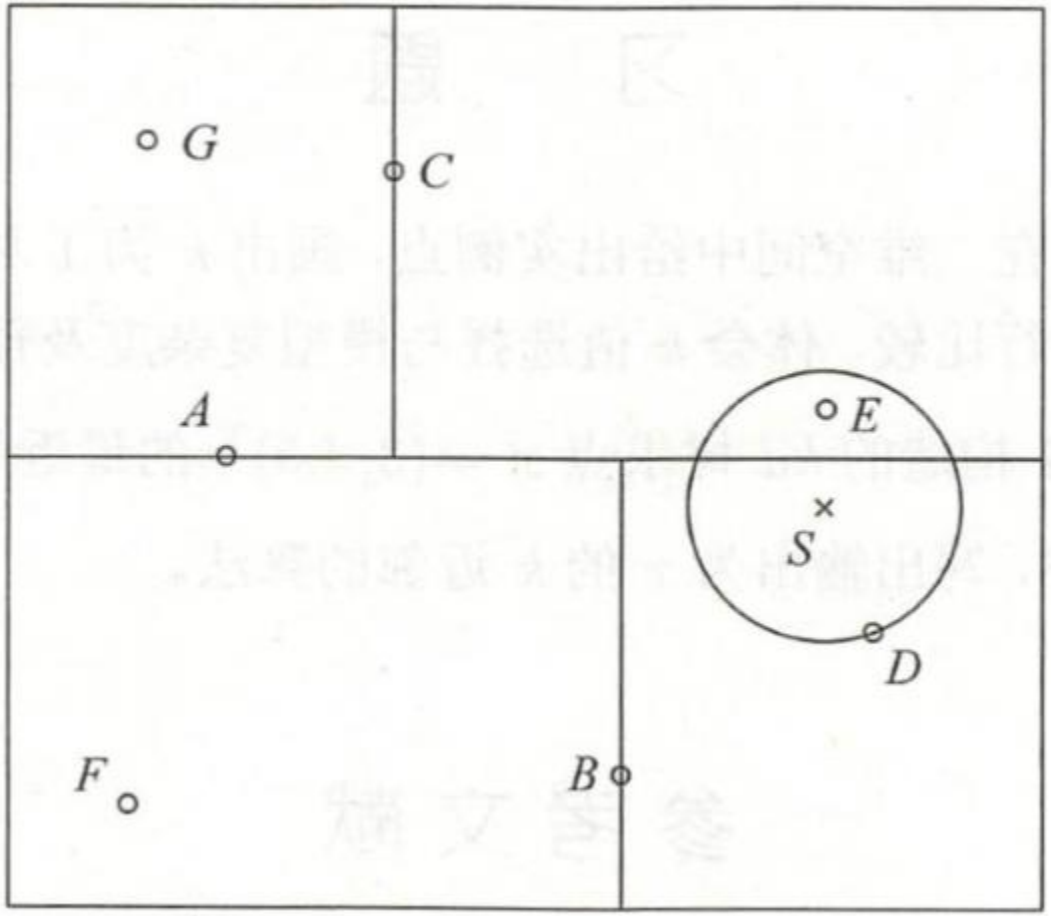
1. 选择第L维(L=j(mod k) + 1，其中j是树深度，k是向量维度)的中位数对应的样本作为当前根节点，第L维小于等于中位数的样本构成左子树，大于中位数构成右子树。
2. 重复1的步骤，一直到子树为空。

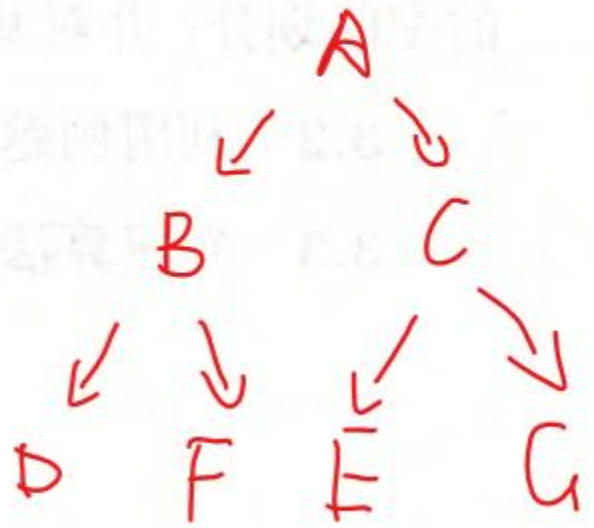
KNN算法的加速—kd树的构建

$$T = \{(2,3)^{\mathrm{T}}, (5,4)^{\mathrm{T}}, (9,6)^{\mathrm{T}}, (4,7)^{\mathrm{T}}, (8,1)^{\mathrm{T}}, (7,2)^{\mathrm{T}}\}$$



KNN算法的加速—kd树的搜索





4 作业要求

4 实验作业要求

实验任务1:

将数据集“semeval.txt”的数据表示成TF-IDF矩阵，并保存为“学号_姓名拼音_TFIDF.txt”文件。

其中词表的顺序要做一个排序。数据具体长这样：

```
1   all:148 anger:22 disgust:2 fear:60 joy:0 sad:64 surprise:0  mortar assault leav at least dead
2   all:131 anger:0 disgust:0 fear:0 joy:93 sad:0 surprise:38   goal delight for sheva
```

每一行是一个样本，一共有三列，每一列都是用tab隔开。其中第一列是文本编号；第二列是总情感权重，各情感权重，各项之间用空格隔开；第三列是文本内容，单词之间以空格隔开。

4 实验作业要求

实验任务2:

使用KNN进行分类任务。数据文件为classification_dataset，其中train_set用于训练。validation_set是验证集，通过调节K值、不同距离度量等参数来筛选准确率最好的一组参数。在测试集test上应用该参数做预测，输出结果保存为"学号_姓名拼音_KNN_classification.csv"

```
Words (split by space),label
europe retain trophy with big win,joy
senate votes to revoke pensions,sad
```

数据一共有两列，其中每一列用英文逗号隔开。第一列为文档，词之间用空格隔开；第二列是标签。

4 实验作业要求

实验任务3:

使用KNN进行回归任务。数据文件为regression_dataset，其中train_set用于训练。validation_set是验证集，通过调节K值、不同距离度量等参数来筛选相关系数最好的一组参数。在测试集test上应用该参数做预测，输出结果保存为"学号_姓名拼音_KNN_regression.csv"。注意：6种概率相加要等于1。

```
Words (split by space),anger,disgust,fear,joy,sad,surprise
europe retain trophy with big win,0,0,0,0.8721,0,0.1279
senate votes to revoke pensions,0.1625,0,0.225,0,0.4375,0.175
```

数据一共有七列，其中每一列用英文逗号隔开。第一列为文档，词之间用空格隔开；第二到七列是标签对应的概率。

实验提交

- 提交到作业FTP： 用户/密码：
- 提交格式：
  - 总文件命名为"学号_姓名拼音_lab1.zip"，注意要提交zip压缩文件。
  - 压缩文件下包含两部分：code文件夹、result文件夹和一个report.pdf。code文件夹下存放实验代码，result文件夹下存放上述提到的结果文件，（不是每次实验都要求交结果，如果没有的话就不用这个文件夹）。
- 注意事项：报告是pdf格式；如果提交的代码是python的话，只需要提交.py文件；如果需要更新提交的版本，则在后面加_v1，_v2。如第一版是"学号_

实验验收

1. 验收日期：下一次实验课
2. 验收形式：在每个时段上课前会上传一个小数据集到群上，提前下载好然后课上验收时当场跑程序，TA会根据结果判断算法是否正确。验收结束可以离开教室。

Thanks