数据科学与工程数学基础

作业提交规范及第 1 次作业

教师：黄定江
助教：陈诺、刘文辉

2022 年 5 月 8 日

作业提交规范

1. 作业提交形式：**练习本或笔记本**（建议统一使用一般的**练习本**即可，不接收以纸张的方式书写的作业）。另外，若作业包含代码部分，**请将代码文件压缩后**上传到**第 1 次作业代码传送门**。代码压缩文件命名格式：“**hw1_ 代码 _ 学号 _ 姓名**”，命名示例：hw1_ 代码 _52215903014_ 刘文辉。其中，“hw1_ 代码”表示第 1 次作业代码。
2. 作业书写说明：
   (a) 可以讨论，禁止抄袭！
   (b) 练习本封面至少包含两方面信息：**姓名**和**学号**
   (c) 每一次的作业**请另起一页**，并在**第一行标明第几次作业**。例如“第 1 次作业”；
   (d) 每一题请**标注题号**，无需抄题，直接解答；
   (e) 题与题之间**请空一行**；
   (f) 不要求字好，但要求书写整体清晰易读。
3. 作业提交途径：纸质作业交给**学习委员**，由学习委员**按学号顺序**收齐后统一在截止日期前交到**助教实验室**。**单数周**布置的作业交到助教刘文辉处**数学馆西 109**；**双数周**布置的作业交到助教陈诺处**地理馆 353**。
4. 作业评分说明：正常提交作业的按照实际评分记录；逾期补交作业的根据逾期情况在实际评分基础上酌情扣分；未交作业的当次作业记为 0 分。

第 1 次作业

提交截至时间：**2022/03/04 下周五 20:00（晚上）**

理论部分

习题 1. 现有一组图片数据集，任务目标是将这些图片分类。其中图片中包含的类别有：猫、狗、鹦鹉、人。试试用 *one-hot* 向量将类别表示为向量。

解. 猫、狗、鹦鹉、人可分别表示为

$$\begin{pmatrix}1\\0\\0\\0\end{pmatrix},\begin{pmatrix}0\\1\\0\\0\end{pmatrix},\begin{pmatrix}0\\0\\1\\0\end{pmatrix},\begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

习题 2. 现有文本集（一行为一个文本）：

```
I know.
You know.
I know that you know.
I know that you know that I know.
```

试计算，该文本集中各个单词的 *TF-IDF* 值。

解. *TF* 值：

TF	*I*	*You(you)*	*know*	*that*
文档 *1*	*0.5*		*0.5*	
文档 *2*		*0.5*	*0.5*	
文档 *3*	*0.2*	*0.2*	*0.4*	*0.2*
文档 *4*	*0.25*	*0.125*	*0.325*	*0.25*

IDF 值：

	I	*You(you)*	*know*	*that*
IDF	$\ln(4/3)$	$\ln(4/3)$	*0*	$\ln(2)$

TF-IDF 值：

TF 列，行	*I*	*You(you)*	*know*	*that*
文档 *1*	$0.5\ln(4/3)$		*0*	
文档 *2*		$0.5\ln(4/3)$	*0*	
文档 *3*	$0.2\ln(4/3)$	$0.2\ln(4/3)$	*0*	$0.2\ln(2)$
文档 *4*	$0.25\ln(4/3)$	$0.125\ln(4/3)$	*0*	$0.25\ln(2)$

实操部分

习题 3. 利用 *python* 统计 *IMDB* 影评数据集 *data.txt* 文件中，每一行表示一篇影评文档。请计算每篇影评中各单词的 *tf*、*idf* 以及 *tf-idf*。提交时需要分别提交三个 *csv* 文件，分别含有各单词的 *tf*、*idf* 以及 *tf-idf* 以及实现的代码。如下给出部分参考代码（仅供参考）：

```
num_of_doc=10
tf_words=[]
num_of_words=[]
#读入数据并计算tf
with open('data.txt', 'r',encoding='UTF-8') as f:
    for i in range(num_of_doc):
        words = f.readline().split()
        #num_of_words.append(# todo #)#
        tf_words.append({})
        for word in words:
            if word in tf_words[i]:
                # todo #
            else:
                # todo #

#计算idf
idf_words={}
for i in range(num_of_doc):
    for item in tf_words[i].items():
        # todo #
```

```
21
22 #计算tf-idf
23 tf_idf_words=[]
24 for i in range(num_of_doc):
25     tf_idf_words.append({})
26     for item in tf_words[i].items():
27         # todo #
28
29
30 #导出csv
31 import pandas as pd
32
33 tf=pd.DataFrame(tf_words).fillna(0)
34 idf=pd.DataFrame([idf_words]).fillna(0)
35 tf_idf=pd.DataFrame(tf_idf_words).fillna(0)
36
37 tf.to_csv('tf.csv')
38 idf.to_csv('idf.csv')
39 tf_idf.to_csv('tf_idf.csv')
```

解. 略