社会计算代码使用说明

鉴于大家对Python 不太熟悉，对文本分类的流程也不太清晰，我们给出了一些示例供大家参考，基本还是结合今天的PPT来的。

Python语法的学习我们这里就不细讲了，大家可以自行查找资料学习。习惯视频学习的同学推荐车万翔（NLP大牛）老师的公开课，《Python程序设计》，共37节，时长总计5个小时左右，在b站和网易云课堂都有，一样的，大家边看视频边实践快速入门。

B站Python车万翔（<https://www.bilibili.com/video/av48465889>）

网易云Python车万翔(<https://mooc.study.163.com/course/1000002017#/info>)

机器学习方面，如果时间宽裕想深入了解的话，建议大家阅读李航老师《统计学习方法》，或者周志华老师《机器学习》，看前面几个章节即可，打好基础是十分关键的。如果没时间，多看看博客并动手实践也可以快速掌握内容，完成作业需求。不需要乱买书，没有营养只会大段复制代码的杂书没什么意义。

关于代码大家可以按照1-5的顺序来学习。基于Python3,建议使用Python3.6和Python3.7。以下1-4都可以直接运行。可能需要pip（conda） install几个包，numpy, sklearn和jieba

1\_get\_text.py是读取文件的代码，可以用来学习读取文件并按行拆开，拆成一个一个的句子

2\_split\_text.py 是分词的代码，即将完整的句子分成一个一个的词语

3\_text\_to\_vector是将文本转换成向量的代码，实现了onehot的过程，在PPT中也有对应的展示，这里添加了单词频率的过滤，频率过低的词语将不会出现在向量中，样例的min\_freq设为1，在实际中应该大一点，例如min\_freq=5。注意：在实际操作中，词表应是训练集，测试集和验证集中的单词的集合，即这三个数据集应当一起计算词表，计算one hot向量。这里只实现了简单的one-hot过程，鼓励大家实现tf-idf。

4\_LogisticsRegression.py是利用得到的向量进行训练和预测的代码，还实现了将结果写入文件的部分。为了方便起见，我们这里没有采用真正的文本的向量进行预测，只使用了随机生成的向量，但是从文本得到向量后，方法和这个是一样的

5\_main.py是完整的代码框架，大家可以往里面填写相关项，实现完整的读文件，分句，分词，向量化，训练，预测和写入结果的过程，只有函数接口，内容留空，大部分内容在前面已经实现，大家往里面结合具体任务写入具体内容即可。也可以不用相关的代码自己独立实现完整的流程。可以调包，也鼓励动手造轮子。

前三个py文件是前后一贯相承的，第四个是单独的LogisticRegression代码。第五个是总体框架。

Data文件夹下是相应的样例数据，每个只有4个句子左右，供调试练习。关于样例代码、作业、数据等等有问题可以在群里提问，也可私聊，或通过课代表转告。

祝大家学习愉快。