上海交通大学安泰经济与管理学院

BUSS3620 人工智能导论

Project #5. 电商购物

刘佳璐 助理教授

电商购物

本项目将编写一个 AI 来预测电商平台顾客是否会完成购买。

```
$ python shopping.py shopping.csv
Correct: 4088
Incorrect: 844
True Positive Rate: 41.02%
True Negative Rate: 90.55%
```

背景介绍

并不是所有人浏览电商平台时都会购买商品，大多数顾客可能只是浏览商品而不会购买。对于电商平台来说，预测用户是否有意购买是一个非常重要的任务。电商平台可以根据用户购买意向向用户展示不同的内容，比如，如果平台认为用户不打算购买商品，它可以向用户显示折扣优惠增加用户购买欲望。那平台要如何确定用户的购买意向？这就是机器学习的用武之地。

你在本项目中需要构建一个近邻分类器来解决这个问题。给定用户的相关信息（他们访问了多少页面、他们是否在周末购物、他们使用哪种浏览器等），你的分类器将预测用户是否会购买商品。你的分类器不会完全准确（完美模拟人类行为是一项远远超出本门可范围的任务），但它应该比随机猜测要好。为了训练分类器，我将提供来自电商平台约 12000 个用户访问数据（网上其实有很多开源的电商数据，比如谷歌开源了自己的电商数据）。

你将使用衡量敏感度(sensitivity)(真阳性率(true positive rate))和特异性(specificity)(真阴性率(true negative rate))来评估这个分类器的准确性。敏感度是指正确识别的阳性样本的比例，也就是正确识别的购买用户的比例。特异性是指正确识别的阴性样本的比例，也就是正确识别的未购买用户的比例。我们的目标是构建一个在两个指标上都表现合理的分类器。

开始

- 从课程中心平台 Canvas 上下 `Week9 机器学习`单元中的 `week9_project.zip` 并且解压缩
- 当处于本项目文件所在的工作目录中时，在终端上运行 `pip3 install -r requirements.txt` 用来安装这次项目需要的 Python 包。

理解项目的相关文件

这个项目最主要的两个文件为：`shopping.py` 和 `shopping.csv`。

`shopping.csv`为电商平台用户数据集。你可以在 txt 文本编辑器中直接打开它，也可以用 Microsoft Excel、Apple Numbers 打开，会更容易直观地理解。这个数据包含了大约 12000 个用户访问数据。每一次用户访问数据(user session) 占一行。前六列表示在这次用户访问过程中对不同类型的页面的访问数据。`Administrative`、`Informational` 和 `ProductRelated` 列衡量用户访问了该类型页面多少次，它们对应的`_Duration` 列衡量用户在这些页面上花费的时间。`BounceRates`、`ExitRates` 和 `PageValues` 列包含一些用户访问的页面时网络分析的相关

数据，比如网页跳转率、网页流失率、网页价值。SpecialDay 是衡量用户访问日期与特殊节假日(如双 11)的接近程度的值。Month 是用户访问月份的缩写。OperatingSystems、Browser、Region 和 TrafficType 都是描述用户自身信息的整数。VisitorType 记录顾客是否是老顾客 Returning_Visitor 还是其他的值。Weekend 记录用户是否在周末访问。数据中最重要的一列是最后一列：Revenue 列。这一列表明用户最终是否购买了商品：TRUE 表示购买，FALSE 表示未购买，我们希望根据所有其他列的值(证据 evidence)来预测这一列(标签 label)。

接下来，打开 shopping.py。main 函数首先调用 load_data 函数从 csv 文件加载数据，然后将数据拆分为训练集和测试集，继而使用 train_model 在训练数据上训练机器学习模型，然后使用训练好的模型对测试数据集进行预测，最后 evaluate 函数输出模型的敏感性和特异性，将之打印到终端。

剩下的 load_data 函数、train_model 函数、evaluate 函数尚未实现，需要同学你来完成。

Week9_project.zip 中还包含 autograde 文件夹，里面存放测试代码的相关文件。

要求

---

load_data 函数

- 输入：csv 文件名
- 功能：读取数据，整理数据格式，返回一个元组(evidence, labels)。
  - evidence 是一个列表，其中每个值表示一个数据点数据点所有证据， labels 也是一个列表，每个值是每个数据点对应的标签
  - 由于电商数据的每一行都会有一个证据和一个标签，因此 evidence 列表的长度和 labels 列表的长度最终应和 csv 文件中的行数（不包括标题行）相等。此外，列表应按照用户在电子表格中出现的顺序排序。即 evidence[0]应该是第一个用户的证据，labels[0]应该是第一个用户的标签
  - evidence 列表中的每个元素本身都应该是一个列表，该列表的长度应为 17，即踢出 csv 文件最后一列（标签列）之外的列数。这个列表中每个证据的顺序也应与证据电子表格中出现的列的顺序相同
  - 请注意，要构建近邻分类器，我们所有的数据都需要是数字。请确保数据具有以下类型：
    - Administrative, Informational, ProductRelated, Month, OperatingSystems, Browser, Region, TrafficType, VisitorType, Weekend 都应该属于 int 类型
    - Administrative_Duration, Informational_Duration, ProductRelated_Duration, BounceRates, ExitRates, PageValues, SpecialDay 都应该是 float 类型
    - Month 中 Jan 应该对应成 0、Feb 对应成 1 ，以此类推，直到 Dec 对应 11
    - VisitorType 应该用 1 表示老顾客和 0 表示其他
    - Weekend 用 1 表示用户在周末访问， 0 表示不在周末访问
  - label 列表中的每个值，用 1 表示购买，0 表示未购买
  - 例第一个证据列表的值应该是[0, 0.0, 0, 0.0, 1, 0.0, 0.2, 0.2, 0.0, 0.0, 1, 1, 1, 1, 1, 1, 0]，第一个标签的值应该是 0
- 输出：一个元组(evidence, labels)

train_model 函数

- 输入：证据列表 evidence 和标签列表 label
- 功能：训练分类模型
  - 我已经导入所需的包 from sklearn.neighbors import KNeighborsClassifier。你需要在此函数中使用 KNeighborsClassifier
- 输出： scikit-learn 的 k 近邻分类器（k = 1）

`evaluate` 函数

- 输入：标签列表 `label`（测试集中用户的真实标签），预测列表 `prediction`（分类器预测的标签）
- 功能：评估模型效果
  - `sensitivity` 应该是一个 `0` 到 `1` 之间的浮点值，代表真实阳性率，即被准确识别的实际阳性标签的比例
  - `specificity` 应该是一个 `0` 到 `1` 之间的浮点值，代表真实阴性率，即被准确识别的实际阴性标签的比例
  - 标签中 `1` 代表阳性（购买商品的用户）或 `0` 代表阴性（未购买的用户）
  - 你可以假设标签中至少有一个阳性标签，以及一个阴性标签
- 输出：两个 `float` 类型的数值`(sensitivity, specificity)`

你不应该修改 `shopping.py` 中已经写好的其他部分。您可以使用 `numpy` 或 `pandas` 或任何其他 `scikit-learn` 的函数。你不应修改 `shopping.csv`。

测试代码

- 你可以使用代码 `pytest autograde/autograde.py --tb=no` 自行测试自己的代码是否满足要求。您需要安装 `requirements.txt` 中的 `pytest` 包。
- 请先确保你的程序能够成功运行并输出结果。请确保你的工作目录中包含 `shopping.py` 和 `shopping.csv`。