|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **实验成绩** |  | | **批阅教师** |  | | **日 期** |  |   西南林业大学校徽11-01  大数据与智能工程学院  实 验（实习）报 告  **课程名称**  综合实习二  **专业班级**  计科2021  **学 号**  20211152020  **学生姓名**  陈志镭  **指导教师**  强振平    2024 年 06 月 24 日 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实习时间** | 2024年6月17日/6月18日/6月19日 | | **实习天数** | | 3天 |
| **实习地点** | 经管楼201/203 | | | | |
| **实习班级** | 计科2021 | | | | |
| **实习人数** | 88 | **分组情况** | | 每人1组 | |
| **实习目的** | 掌握机器学习和深度学习基础知识和运用 | | | | |
| **实习要求** | 1. 配置Python环境：Anaconda，Pycharm，jupter，Spyder等 2. 掌握Jupyter的使用方法 3. 掌握机器学习的构建方法 4. 掌握神经网络模型的训练方法 5. 掌握深度学习基础知识 | | | | |
| **实习内容** | 基础实验内容：  1. 应用举例一：  对某银行在降低贷款拖欠率的数据进行逻辑回归建模；  2. 应用举例二：  找出下列谁是学霸？  3. Titanic数据集分析  4. 餐饮客户价值分析  机器学习内容：（https://scikit-learn.org/stable/index.html）  1. 分类器比较  https://scikit-learn.org/stable/auto\_examples/classification/plot\_classifier\_comparison.html#sphx-glr-auto-examples-classification-plot-classifier-comparison-py  2. 聚类比较  https://scikit-learn.org/stable/auto\_examples/cluster/plot\_cluster\_comparison.html#sphx-glr-auto-examples-cluster-plot-cluster-comparison-py  3. 分解方法比较：  https://scikit-learn.org/stable/auto\_examples/cross\_decomposition/plot\_compare\_cross\_decomposition.html#sphx-glr-auto-examples-cross-decomposition-plot-compare-cross-decomposition-py  4. 人脸数据集分解（记得测试自己的人脸图像）  https://scikit-learn.org/stable/auto\_examples/decomposition/plot\_faces\_decomposition.html#sphx-glr-auto-examples-decomposition-plot-faces-decomposition-py | | | | |
|  | **（本栏须填写清楚实习的日期及该天实习的具体内容）**   * 2024年6月17日   完成Python环境的配置：Anaconda，Pycharm，jupter，Spyder等   * 2024年6月18日   完成机器学习课程的基础实验内容完。   * 2024年6月19日   完成深度学习框架搭建 1.tensorflow 2.pytorch。从两个框架中选择一个框架完成自选深度学习的项目，以及实验报告的撰写。  第一章 简单机器学习  1.对某银行在降低贷款拖欠率的数据进行逻辑回归建模；  1.1导入处理数据需要的包    图1  1.2对数据进行预处理    图2  1.3建立逻辑回归模型用于特征选择    图3  1.4使用筛选后的特征数据重新构造 X并输出    图4    图5  2. 找出下列谁是学霸？  2.1导入库使用K-means聚类算法来对数据进行聚类。    图6  2.2数据准备：创建了6个列表，每个列表包含4个数据点。    图7  2.3数据叠加：使用vstack函数将这些列表垂直叠加，生成一个2D数组data。  2.4 K-means 聚类：使用kmeans函数对数据进行K-means聚类，设置聚类中心的数量为2。kmeans函数返回聚类中心（centroids）和一个失配量。    图8  2.5量化（vq）：使用vq函数对数据进行量化，根据聚类中心对每个数据点进行分类。vq函数返回每个数据点的聚类结果（result）和一个失配量。  输出结果：打印聚类结果。    图9  3. Titanic数据集分析  3.1读入库导入数据显示数据    图10  3.2查看摘要数据计算总体生存率    图11  3.3按舱位计算生存率    图12  3.3.1绘制柱状图    图13  3.4按性别计算生存人数    图14  3.4.1绘制柱状图    图15  3.5按年龄组绘制生存率    图16  3.6按舱位和性别绘制生存率柱状图    图17  4. 餐饮客户价值分析  4.1导入所需包和导入数据进行标准化    图18  4.2 设置k=3，进行聚类数据标准化处理    图19  4.3 获取聚类结果，保存到excel    图20  4.4 绘制密度图并将结果保存    图21    图22  第二章 机器学习内容  1. 分类器比较  1.1导入所需库    图23  1.2定义分类器名称和实例：    图24  1.3生成数据集    图25  1.4绘图    图26    图27  2. 聚类比较  2.1导入必需的库    图28  2.2设置数据集    图29  2.3设置聚类参数    图30  2.4绘制图形    图31  2.5 图像聚类比较    图32  3.分解方法比较  3.1导入所需的库    图33  3.2典型相关分析 (CCA)：    图34  3.3可视化典型相关分析的结果：    图35    图36  3.4 PLS回归 (PLS2)：    图37  3.5 PLS回归 (PLS1)：    图38  3.6**典型相关分析 (CCA)**：    图39  4．人脸数据集分解（记得测试自己的人脸图像）  4.1导入必需包和数据集    图40  4.2定义一个脸部轮廓函数    图41  4.3分辨    图42  4.4非负成分    图43  4.5独立成分    图44  4.6稀疏成分    图45  4.7字典学习    图46  4.8因子分析成分    图47  4.9图片效果    图48  4.10导入自己的图形    图49  4.11转变图片数据类型    图50  4.11根据训练好的模型测试自己的头像    图51  第三章 基于PyTorch的开发  1.定义网络模型  1.1导入必须库    1.2定义了一个简单的神经网络模型 Net，包括两个卷积层 (Conv2d)、两个dropout层 (Dropout2d) 和两个全连接层 (Linear)。这个模型用于处理28x28的单通道图像，并输出一个包含10个类别预测概率的张量。      1.4 random\_data 是一个随机生成的28x28的单通道图像数据，通过模型 my\_nn 进行前向传播，得到了输出 result，它包含了对10个类别的预测概率。    2. 模型保存加载的基础：state\_dict  2.1导入必需包    2.2定义了一个简单的卷积神经网络模型 Net，用于处理彩色图像分类    2.3使用了ReLU作为激活函数，并在每个卷积层后面跟随最大池化操作。  初始化了一个SGD优化器，用于优化网络参数。打印了模型和优化器的状态字典，展示了各自的参数及其形状。    3. 图像中目标检查  3.1  3.2    4. 视觉迁移学习  4.1导入必需的包    4.2数据预处理：使用torchvision.transforms进行数据增强和标准化。    4.3显示数据图像    4.4可视化：定义visualize\_model函数来可视化模型的预测结果    4.5模型定义    4.6可视化图形    5. 对抗样本生成  5.1导入必需库和数据集    5.2模型定义    5.3FGSN攻击函数    5.4测试函数    5.5运行测试函数    5.6文字识别可视化      6. 文本：名称分类  6.1导入必需库和数据集    6.2构建分类字典    6.3定义RNN模型    6.4训练数据集    6.5模型评估    6.6训练循环    6.7绘制损失矩阵    6.8预测 | | | | |
| **实习总结** | 在本次实习过程中，我通过实现一个基于循环神经网络（RNN）的文本分类模型，系统地掌握了从数据预处理到模型训练和评估的全过程。这个项目使我对深度学习和自然语言处理有了更深刻的理解，并让我对PyTorch框架的使用更加熟练。  数据预处理是项目的第一步，也是非常关键的一步。我学会了如何读取和处理包含非ASCII字符的文本数据，并将这些数据转换为适合神经网络输入的张量（Tensor）。通过编写数据转换和处理函数，我能够有效地将原始数据转换为模型可以处理的格式。这不仅提高了我的编程技能，还加深了我对数据预处理重要性的认识。  在模型构建方面，我使用PyTorch定义了一个简单的RNN模型，并了解了其结构和工作原理。通过实验不同的超参数设置，我学会了如何优化模型性能。在训练过程中，我使用了负对数似然损失函数（NLLLoss）和随机梯度下降（SGD）优化器。这些知识和技能将为我未来进行更复杂的模型设计和优化打下坚实基础。  为了评估模型的性能，我绘制了损失曲线和混淆矩阵。通过这些可视化手段，我能够直观地看到模型的训练效果和分类性能。这不仅帮助我发现和改进模型中的不足之处，还提高了我在数据分析和可视化方面的能力。  在项目过程中，我遇到了许多困难和挑战，例如数据处理中的字符转换问题和模型训练中的过拟合现象。通过查阅资料网上集思广益和反复试验，我逐步解决了这些问题。这一过程锻炼了我的问题解决能力和独立思考能力。  本次实习不仅让我掌握了基于PyTorch的文本分类技术，还提升了我的编程能力和项目管理能力。通过这个项目，我更加坚定了在深度学习和自然语言处理领域继续学习和研究的决心。未来，我希望能够在更复杂和多样化的项目中应用所学知识，进一步提高自己的技能水平。 | | | | |