电子科技大学信息与软件工程学院

**实 验 报 告**

学 号 2019091604016

姓 名夏野

（实验） 课程名称 人工智能

实验时间 11月1日

实验教师 周帆

**电 子 科 技 大 学**

**实 验 报 告**

1. **实验名称：**

基于sklearn工具包进行线性回归和逻辑斯蒂回归

1. **实验学时：**

4学时

1. **实验目的：**

初步了解机器学习，学习如何使用python重要模块sklearn实现回归和分类任务。其中，线性回归是一类经典回归任务，逻辑斯蒂回归是一类经典分类任务。通过这个实验，能够掌握机器学习中简单的回归和分类任务的解决步骤，为后面复杂的深度学习打好基础。

1. **实验原理：**

回归分析：指一种预测性的建模技术，主要是研究自变量和因变量的关系。通常使用线/曲线来拟合数据点，然后研究如何使曲线到数据点的距离差异最小。线性回归是回归分析的一种。假设目标值（因变量）与特征值（自变量）之间线性相关（即满足一个多元一次方程，如：f(x)=w1x1+…+wnxn+b.）；然后构建损失函数；最后通过令损失函数最小来确定参数。

逻辑回归：属于机器学习里面的监督学习，它是以回归的思想来解决分类问题的一种非常经典的二分类分类器。由于其训练后的参数有较强的可解释性，在诸多领域中，逻辑回归通常用作 baseline 模型，以方便后期更好的挖掘业务相关信息或提升模型性能。

梯度下降：用来最小化我们的损失函数的。在直线方程中，导数就代表斜率；而在曲线方程中，导数代表的是该点处切线的斜率。因此，为了不断更新我们的参数，也就是w、b，我们要求损失函数的梯度，并沿着梯度的反方向就行更新，这样，就可以到达一个局部最低点。

Sklearn ：(全称 Scikit-Learn) 是基于 [Python](https://so.csdn.net/so/search?from=pc_blog_highlight&q=Python) 语言的机器学习工具。它建立在 NumPy, SciPy, Pandas 和 Matplotlib 之上，里面的 API 的设计非常好，所有对象的接口简单，很适合新手上路。在 Sklearn 里面有六大任务模块：分别是分类、回归、聚类、降维、模型选择和预处理

1. **实验内容：**
2. 线性回归：加载diabete数据集，划分数据集为训练集和测试集。搭建线性回归模型用于训练，得出预测结果。
3. 逻辑回归：加载iris数据集，使用某个或多个特征搭建二元或者多元逻辑斯蒂回归模型，判断样本分类。
4. **实验器材（设备、元器件）：**

1. 硬件：mac笔记本电脑

2. 软件：MAC OS，pycharm

1. **实验步骤：**
2. 线性回归：
   1. 加载数据集：使用sklearn包下的datasets模块，利用load\_diabetes()方法获得diabetes数据集，总共有442份数据，其中包括data和target。
   2. 切割数据集：利用python中的切片，将data和target分别切割成大约10:1的训练集和测试集。因此，获得400份训练数据和它对应的target，42份测试数据和它对应的target。由于在sklearn中封装的数据类型是numpy，因此生成的数据类型就是ndarray。我打印训练数据和测试数据的形状，一个(400, 10)，一个(42, 10)；
   3. 加载模型：使用sklearn包下的linear\_mode的类LinearRegression进行模型加载；
   4. 训练和测试：将训练数据和target作为参数传入fit方法中，进行模型的训练，然后利用predict方法预测测试数据；
   5. 评估：由于是回归任务，一般采用MAE或者MSE，但是考虑到MSE的量纲前后不一致，因此这里我才用MAE作为损失值。
3. 逻辑斯蒂回归：
   1. 加载数据集：使用sklearn包下的datasets模块，利用load\_iris()方法获得iris数据集，总共有150份数据，其中包括data和target。打印shape之后，形状是(150, 4)；
   2. 数据处理：由于从数据集中下载下来的数据它的数值分布不均匀，在进行分类任务的时候可能会造成模型训练的失败，因此为了提高最后的准确度，我选择使用标准化方法StandardScaler进行处理，利用fit\_transform对训练数据进行标准化；
   3. 加载模型：使用sklearn包下的linear\_mode的类LogisticRegression进行模型加载；
   4. 训练：将训练数据和target丢进fit()方法中进行训练；
   5. 评估：这里我自己设计方法获得acc，利用真实数据和预测结果进行比较求值，得到模型训练结果的准确率。
4. **实验结果与分析（含重要数据结果分析或核心代码流程分析）**
5. 线性回归：

运行结果：

Text

Description automatically generated

重要数据结果分析：

装载数据集得到数据集长度为442，类型为`numpy-ndarray`，进行数据划分、构建回归模型、训练后得到的最终训练集以及测试集的标准差如图所示。

核心代码流程分析：

1. 获取数据集

代码：

diabetes = datasets.load\_diabetes()

分析：

调用datasets.load\_diabetes()方法装载diabetes数据集，存入diabetes变量中。

1. 切分数据集

代码：

x\_train = diabetes.data[:-20]  
y\_train = diabetes.target[:-20]  
x\_test = diabetes.data[-20:]  
y\_test = diabetes.target[-20:]

分析：

调用diabetes.data[]分别将我们的数据集中的data和target两个数据切分，分为x，y两个数据，并且分为训练数据集合测试的数据集。

1. 构建线性回归模型，训练后并预测分析结果

代码：

linreg = linear\_model.LinearRegression()  
linreg.fit(x\_train, y\_train)  
linreg.predict(x\_test)

分析：

调用linear\_model.LinearRegression()函数构建我们的线性回归模型对象linreg。接着调用linreg中的fit方法，训练我们的对象模型。最后调用linreg的predict方法来进行预测。

4. 得出预测结果

std\_train = mean\_absolute\_error(y\_train, linreg.predict(x\_train))  
std\_test = mean\_absolute\_error(y\_test, linreg.predict(x\_test))

调用mean\_absolute\_error()函数分别得到测试集和训练集的标准差。

2. 线性回归：

运行结果：

Text

Description automatically generated

重要数据结果分析：

我们的逻辑回归模型经过训练后得到的对数据的准确度达到了97.33%

核心代码流程分析：

1. 获取数据集

代码：

iris = load\_iris()

分析：

调用iris = load\_iris()方法装载iris数据集，存入iris变量中。

1. 切分数据集

代码：

x = iris.data  
y = iris.target  
x = StandardScaler().fit\_transform(x)

分析：

调用iris.data, iris.target分别将我们的数据集中的data和target两个数据切分，分为x，y两个数据，并且分为训练数据集合测试的数据集。

1. 构建线性回归模型，训练后并预测分析结果

代码：

lr = LogisticRegression()  
lr.fit(x, y.ravel())

分析：

调用LogisticRegression()函数构建我们的逻辑回归模型对象lr。接着调用linreg中的fit方法，训练我们的对象模型。

4. 得出预测结果

y\_hat = lr.predict(x)   
result = y\_hat == y  
acc = np.mean(result)

调用lr中的predict()方法预测我们的数据集，并将结果存入到y\_hat中。接着调用np.mean()得到我们预测数据集的准确度。

1. **总结及心得体会：**

通过实验一，我初步了解机器学习，学习如何使用python重要模块sklearn实现回归和分类任务，同时也学习了机器学习中简单的分类和回归基本步骤。其中，线性回归是一类经典回归任务，逻辑斯蒂回归是一类经典分类任务。

机器学习一般步骤：

* + - 1. 装载数据集，进行模型训练时还需要一部分作为测试集，可以看到为训练的数据集来查看我们的训练成果如何。这样我们就可以人工调账我们的参数。一般来说，我们的训练集与测试集的比例为10：1。
      2. 构建训练模型，并使用fit放来进行训练
      3. 进行模型评估，得到我们训练模型的准确度。