



回归分析：Logistic回归分析教程

一、Logistic回归简介

1.1.1 什么是Logistic回归

Logistic回归是一种统计学方法，用于分析一个或多个自变量与一个二分类因变量之间的关系。与线性回归不同，Logistic回归预测的是事件发生的概率，其输出值被限制在0和1之间，表示事件发生的可能性。Logistic回归的核心是Sigmoid函数，数学表达式为：

[P =]

其中， $(z = b_0 + b_1X_1 + b_2X_2 + \dots + b_nX_n)$ ， (b_0) 是截距， (b_1, b_2, \dots, b_n) 是回归系数， (X_1, X_2, \dots, X_n) 是自变量。Sigmoid函数将线性组合的值转换为概率。

2.1.2 Logistic回归的应用场景

Logistic回归广泛应用于各种领域，包括但不限于：

- 医学研究：预测患者是否患有某种疾病，如基于年龄、性别、体重等特征预测糖尿病。
- 金融行业：信用评分模型，预测贷款人是否违约。
- 市场营销：预测用户是否会点击广告或购买产品。
- 社会科学：分析社会现象，如预测某人群是否会支持某个政策。

3.1.3 Logistic回归与线性回归的区别

Logistic回归与线性回归的主要区别在于：

- 因变量类型：线性回归适用于连续型因变量，而Logistic回归适用于分类因变量，尤其是二分类。
- 预测目标：线性回归预测的是因变量的具体数值，Logistic回归预测的是事件发生的概率。
- 模型输出：线性回归的输出可以是任何实数，Logistic回归的输出被限制在0和1之间。
- 损失函数：线性回归通常使用均方误差作为损失函数，Logistic回归使用对数损失（log loss）或交叉熵损失。

3.1 示例：使用Python进行Logistic回归分析

假设我们有一组数据，包含年龄和是否购买保险的信息，我们将使用Logistic回归来预测一个人是否会购买保险。

```
import pandas as pd
import numpy as np
```

```

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix

# 创建示例数据
data = {
    'Age': [25, 30, 35, 40, 45, 50, 55, 60, 65, 70],
    'Bought_Insurance': [0, 0, 0, 1, 1, 1, 1, 1, 1, 1]
}
df = pd.DataFrame(data)

# 划分数据集
X = df['Age'].values.reshape(-1,1)
y = df['Bought_Insurance'].values
X_train, X_test, y_train, y_test = train_test_split(X, y,
    test_size=0.2, random_state=42)

# 创建Logistic回归模型
model = LogisticRegression()
model.fit(X_train, y_train)

# 预测
y_pred = model.predict(X_test)

# 评估模型
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))

```

在这个例子中，我们首先导入了必要的库，然后创建了一个简单的数据集，其中包含年龄和是否购买保险的信息。我们使用`train_test_split`函数将数据集划分为训练集和测试集，然后创建了一个**Logistic**回归模型并用训练数据拟合它。最后，我们用测试集评估模型的性能，通过混淆矩阵和分类报告来查看预测的准确性。

3.2 代码解释

1. 数据导入和预处理：使用pandas库创建**DataFrame**，然后将数据划分为特征（年龄）和目标变量（是否购买保险），并使用`train_test_split`函数将数据集分为训练集和测试集。
2. 模型创建和训练：使用sklearn.linear_model中的**LogisticRegression**类创建模型，并用训练数据拟合模型。
3. 预测和评估：使用模型对测试集进行预测，然后通过sklearn.metrics中的**confusion_matrix**和**classification_report**函数评估模型的预测性能。

通过这个例子，我们可以看到**Logistic**回归在处理分类问题时的基本流程和方法。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/257000162026006133>