多分类-对数几率回归

171860659 吴紫航

算法思路

1.数据是一个 26 分类问题，用 OvR 转化为 26 个二分类器

2.每个二分类过程(采用 Logistic Regression)如下：

1) 设定初始参数$\boldsymbol{\beta^0}$: $(\omega_1;\omega_2;\dots;\omega_{16};b)$为全 0

2) 读取训练集，计算损失函数$l(\beta)$的梯度$\nabla l(\boldsymbol{\beta}) = -\sum_{i=1}^{14000}\hat{\boldsymbol{x}}_i(y_i - \frac{e^{\omega^T x+b}}{1+e^{\omega^T x+b}})$，其中$\hat{\boldsymbol{x}}_i = (\boldsymbol{x_i};1)$; $y_i = 1\ if$ 二分类器号和样本点类号一致，$else\ y_i = 0$

3) 设定超参数-迭代次数 t，用方程$\boldsymbol{\beta^{t+1}} = \boldsymbol{\beta^t} - \gamma\nabla l(\boldsymbol{\beta})$进行 t 次迭代，其中$\gamma$为超参数-学习速度，每次迭代需要重新计算$\nabla l(\boldsymbol{\beta})$

4) 对于测试集的 6000 个样本，都用$p = \frac{e^{\omega^T x+b}}{1+e^{\omega^T x+b}}$ 进行预测

3.所有二分类过程都完成后，对于每个样本，选取 26 个二分类器中，p 值最大的分类器结果作为最终预测结果

4.最后根据每个测试样本的预测结果和真实结果，计算 performances 表格

运行方式和输出文件

运行 LR_main.py，输出三个 csv 文件

输出文件	说明
result.csv	6000x26 的矩阵 对应 6000 个测试样本的 26 个二分类结果的 p 值
out.csv	6000x2 的矩阵 对应 6000 个测试样本的预测和真实类别

performance.csv	从上到下依次为 accuracy、microPrecision、microRecall、microF1、macroPrecision、macroRecall、macroF1

超参数

学习速度γ设定为 1e-6

迭代次数 t 设定为 500

性能分析

Performance Metric	Value (%)
accuracy	63.4667
micro Precision	63.4667
micro Recall	63.4667
micro F1	63.4667
macro Precision	63.4286
macro Recall	63.6776
macro F1	61.6450