人工智能基础算法 第二次作业

王怡丰 2024310877

1

对于

$$\min_{\beta}\frac{1}{2}\sum_{n=1}^{N}(y_n-\sum_{m=1}^{M}x_{nm}\beta_m)^2+\lambda\sum_{m=1}^{M}\beta_m^2$$

将其写作矩阵形式

$$\begin{aligned}&\min_{\beta}\frac{1}{2}||\mathbf{Y}-\mathbf{X}\beta||_{\mathbf{2}}^{\mathbf{2}}+\lambda||\beta||_{\mathbf{2}}^{\mathbf{2}}\\ =&\min_{\beta}\frac{1}{2}(\mathbf{Y^T}-\beta^{\mathbf{T}}\mathbf{X^T})\mathbf{(}\mathbf{Y}-\mathbf{X}\beta)+\lambda\beta^{\mathbf{T}}\beta\\ =&\min_{\beta}\frac{1}{2}\mathbf{Y^TY}-\mathbf{Y^TX}\beta+\frac{\mathbf{1}}{\mathbf{2}}\beta^{\mathbf{T}}\mathbf{X^TX}\beta+\lambda\beta^{\mathbf{T}}\beta\end{aligned}$$

对该式关于β求导，并令导数等于0

$$\begin{gathered}-\mathbf{X^TY}+\mathbf{X^TX}\beta+\mathbf{2}\lambda\beta=\mathbf{0}\\ \beta=\mathbf{(X^TX+2\lambda I)^{-1}X^TY}\end{gathered}$$

2

设计了逻辑回归程序，利用`sigmoid`函数$f(z)=\frac{1}{1+e^{-z}}$来实现线性回归到概率$(0,1)$的映射，利用`log_likelihood`函数计算模型的误差，利用`train`函数通过梯度下降法来对模型进行训练，得到线性回归的斜率和截距`weights`，最终利用`predict`函数预测测试样例结果为1的概率，并将概率大于50%的样例结果置为1，小于50%的置为0。

以A为训练集，以B为测试集进行了测试，结果正确率为88.7%。

3

设计了岭回归程序，程序架构与**2**中相似，区别在于改变了`log_likelihood`、`train`函数中的损失、梯度计算方式，添加了正则项。

首先在训练集A中进行10折交叉检验法，确定最佳的超参数λ。首先确定λ的量级

λ	**正确率**
1e-6	0.858
1e-5	0.873
1e-4	0.874
1e-3	0.870
1e-2	0.869
1e-1	0.868
1	0.743

对于$\lambda \in (1e-5, 1e-3)$，再取两个点精细化λ的取值

λ	**正确率**
5e-5	0.877
5e-4	0.869

于是在$\lambda \in (1e-5, 1e-4)$内确定取值

λ	**正确率**
1e-5	0.873
2e-5	0.874
3e-5	0.873
4e-5	0.876
5e-5	0.877
6e-5	0.878
7e-5	0.878
8e-5	0.879
9e-5	0.875
1e-4	0.874

因而确定$\lambda = 8e-5$，并以A为训练集，以B为测试集进行了测试，结果正确率为87.8%。

4

设计了lasso回归程序，程序架构与**3**中相似，区别在于改变了`log_likelihood`、`train`函数中正则项的计算方式。

首先在训练集A中进行10折交叉检验法，确定最佳的超参数γ。首先确定γ的量级

γ	**正确率**
1e-6	0.857
1e-5	0.873
1e-4	0.873
1e-3	0.866
1e-2	0.871
1e-1	0.871
1	0.869

于是在$\gamma \in (1e-5, 1e-4)$内确定取值

γ	**正确率**
1e-5	0.873
2e-5	0.875
3e-5	0.874
4e-5	0.876
5e-5	0.877
6e-5	0.880
7e-5	0.879
8e-5	0.878
9e-5	0.874
1e-4	0.873

因而确定$\gamma = 6e-5$，并以A为训练集，以B为测试集进行了测试，结果正确率为87.7%。

5

基于`sklearn`中已有的SVM实现了支撑向量机训练、判断程序。

首先在训练集A中进行10折交叉检验法，确定最佳的超参数C。首先确定C的量级

C	**正确率**
1e-10	0.528
1e-9	0.927
1e-8	0.987
1e-7	0.993
1e-6	0.993
1e-5	0.994
1e-4	0.994
1e-3	0.994
1e-2	0.994
1e-1	0.994
1	0.994
10	0.994
100	0.994

因而确定$C = 1e-4$，并以A为训练集，以B为测试集进行了测试，结果正确率为99.4%。

6

对以上四种方法的最终结果进行比较

方法	**正确率**
线性回归	88.7%
岭回归	87.8%
lasso回归	87.7%
支撑向量机	99.4%

对不同方法的结果进行比较，支撑向量机结果显著优于其他三种方法，线性回归略微优于岭回归和lasso回归，岭回归和lasso回归结果基本相同。这可能是因为相比于其他三种方法，支撑向量机对于对数据进行二分是一个更高效的方法，且本例直接调用成熟的库进行实现，能够实现很高的正确率；线性回归计算相对简单，且基本不涉及超参数的选择，而岭回归、lasso回归虽然通过进行正则化优化了训练拟合过程，但计算更为复杂，且需要选择超参数，受本例的计算能力、篇幅限制不能选出最佳的超参数，这导致正确率不及线性回归，若以更大的计算精度（如设置更多的梯度下降循环次数或设置收敛准则）对超参数进行选择，则可能可以提高这两种方法的正确率。