

机器学习实验---人工神经网络MLP基于sklearn的实现

原创

卷心菜菜 于 2017-10-23 23:25:13 发布 4351 收藏 6

分类专栏: [机器学习](#) 文章标签: [神经网络](#) [机器学习](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/qq5q13638/article/details/78324697>

版权



[机器学习](#) 专栏收录该内容

6 篇文章 0 订阅

订阅专栏

导语

上次写了随机森林的基于sklearn库的大致实现步骤, 这次我们来看看人工神经网络的实现步骤吧~

MLP就是多层感知器的意思, 我们这里也是通过引入MLPClassifier来实现的。

步骤详解:

首先从sklearn中导入MLPC

导入待训练的数据集

通过调用MLPC训练器对训练集进行训练

利用训练集对测试数据进行测试

返回错误率

```
from sklearn.neural_network import MLPClassifier
x = [[0,0]
y = [0,1]

clf = MLPClassifier(solver='lbfgs',alpha=1e-5,hidden_layer_sizes=(5,2),random_state=1)

clf.fit(X,y)

X1 = [[2,2],[-1,2]]
preY1 = clf.predict(X1)
```

以上代码从输入训练集, 和输入测试集, 最后给出了对测试集的预测。

那我们接下来解析一下MLPClassifier分类器的属性设定吧~

```
solver='lbfgs'
```

solver给的是神经网络的求解方法, lbfgs是其中一种求解方法, 在数据比较小的时候表现较好。

adam这种方法则鲁棒性较强,

sgd则在参数调整较优的时候会有最佳表现。

本文代码来自于官网, 思路是参考的各路大神的博客~