机器学习 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号： | 姓名： | | 班级： |
| 实验题目： | | | |
| 实验学时：2 | | 实验日期： | |
| 实验目的：  将基本梯度下降法和牛顿法应用到表中的数据上，对二维数据给出判别，画出以迭代次数为准则函数的曲线；  估计两种方法的数学运算量；  画出收敛时间-学习率曲线，求出无法收敛的最小学习率。 | | | |
| 硬件环境：  Intel Core（TM） i5-4210U  RAM 8G  64位操作系统，基于x64处理器 | | | |
| 软件环境：  Win 10 64bit  MATLAB | | | |
| 实验步骤与内容：   1. 根据课本P184-185可知，寻找分离两类数据的权向量，最基本的问题是选取一个合适的准则函数（J（a）），当a为解向量时，J（a）最小。翻阅课本最终选择以5.6节的松弛算法作为准则函数（5.5.1节的函数实现更加简单但由于二阶导数不存在故不能够用于牛顿法中求算海森矩阵）。为了方便计算，忽视b的影响，得出追中用于编程实现的J（a）公式      1. 对J（a）求偏导数得到▽J（a），再对J（a）求二次偏导得到海森矩阵的计算方式        1. 查看数据的分布情况，简单判断其是否线性可分   points.jpg  通过此分布可以判断，除去某几个特殊点外，这些数据是线性可分的   1. 在得到公式后，通过编程实现。首先要对w1和w3的数据进行规范化。这里，将w1的y值标为1，将w3的值标为-1. 2. 松弛算法中进行计算的样本Y（a）表示被a错分的样本（P192），因此在拿到某一数据后首先检查它是否在满足a\*y<=b的样本集中 3. 在MATLAB中，通过函数sum（A）可得到矩阵A的各项和，通过norm（A，x）函数可以获得矩阵A的第x范数，这里取x=1，即A的最大奇异值      1. 在每次求和后更新a并记录J（a）值，同时检查η(k)\*▽J（a）的值（使用范数）是否已经进入误差允许范围，若是则跳出循环同时生成二维图如下   lab4-a.jpg  其中，绿线为梯度下降法，经过17次实现收敛，红线为牛顿法，在第二次就达成了收敛条件。可以看出，梯度下降法的收敛速度相比较牛顿法还是略慢了一些。然而与梯度下降法相比较，牛顿法存在一个求海森矩阵以及求海森矩阵的逆矩阵的过程，这个过程的时间复杂度为O（d3），本次的海森矩阵只有二维尚不能看出差别，而在维度较高时计算复杂度将非常高且耗时长。在实际应用过程中，多数都是使用牛顿法的改善版本。而在梯度下降法中，将η(k)设置为比较小的常数，虽然比每一步都是用最优的η(k)将需要更多次循环来校正，但通常总的时间开销却更少。   1. 在作出该图后，我们要找到无法收敛的最小学习率。通过设置一个有不同学习率值的循环，在每次循环执行结束后都会完成一次权向量的求解，并记录迭代次数作为收敛时间，画出收敛时间-学习率曲线如下   rate.jpg  通过放大查看最高点  highest.PNG  这里迭代次数过大，故将其断定为无法收敛，此时的学习率为0.35 | | | |
| 结论分析与体会：  这次实验相较之前几次难度有较大的增加，用了很长时间才勉强做出来。在本次实验中，基本梯度下降法和牛顿法都能求得最终的权向量，使得准则函数取得一个极小值。基本梯度下降法利用一阶导，而牛顿法利用二阶导，牛顿法每次都是使用基本梯度的最优情况，因此牛顿法下降步长大，较快收敛，但同时每次迭代的复杂度也更大。本次实验查阅了较多的资料，包括对基础知识包括偏导数在内的回顾和学习，以及对Logistic回归模型的查询与计算。最后仍然使用了松弛算法而不是Sigmoid函数，因此在代码中虽然有一部分Sigmoid作准则函数的尝试，但最后并未成功，也没有进行应用。 | | | |

**参考链接**

导数，偏导数，梯度

https://blog.csdn.net/walilk/article/details/50978864

偏导数

<https://zh.wikipedia.org/wiki/%E5%81%8F%E5%AF%BC%E6%95%B0>

复合函数求导

https://zhidao.baidu.com/question/168770776.html

损失函数收敛之处

https://developers.google.com/machine-learning/crash-course/reducing-loss/gradient-descent?hl=zh-cn

Logistic回归与梯度下降法

<https://blog.csdn.net/ligang_csdn/article/details/53838743>

norm函数

<https://blog.csdn.net/evan123mg/article/details/39347949>

范数

https://www.zhihu.com/question/20473040

<https://blog.csdn.net/susanzhang1231/article/details/52127011>

奇异值

https://baike.baidu.com/item/%E5%A5%87%E5%BC%82%E5%80%BC

海森矩阵（Hessian矩阵）（二阶偏导数矩阵）

https://zh.wikipedia.org/wiki/%E6%B5%B7%E6%A3%AE%E7%9F%A9%E9%98%B5

牛顿法 与 Hessian矩阵

https://www.cnblogs.com/jiahenhe2/p/8086857.html

https://blog.csdn.net/baimafujinji/article/details/51179381

松弛算法（Y(a) 是满足 a⊤y≤b 的全部 y 构成的集合）

http://www.cnblogs.com/Determined22/p/6507329.html

unit8

https://blog.csdn.net/bowen\_wu\_sysu/article/details/78878800