**使用人工神经网络和回归树进行LOL游戏时长的预测**

**PB20000156 徐亦昶**

1. 人工神经网络

本次实验采用包含两个隐含层的神经网络。尝试过的特征组合如下：6组布尔型数据+20组整型数据直接送到输入层；将20组整型数据分成5组，每组数据的和以及6组布尔型数据共10组数据送到输入层；将布尔型数据进行主成分分析压缩成两组，和之前求和得到的4组数据共6组数据送到输入层；仍然进行主成分分析，这次直接压缩成一组。测试结果显示，倒数第二一种方法可以保证较高的精确度，同时训练速度较快，可以在较短的时间内训练完所有的数据。

每层节点的选择：仅对第二层节点进行调整。实验流程是每次将第二层节点数调整后，使用训练：测试=4：1的比例划分数据集，并测试模型在测试集上的mean-square。测试结果如下：

|  |  |
| --- | --- |
| 节点数 | Mean-square |
| 5 | 35258.973 |
| 8 | 33336.777 |
| 10 | 34379.316 |
| 12 | 33587.03 |
| 15 | 32847.03 |
| 16 | 33505.152 |
| 17 | 33894.35 |

可见15隐含层15个节点最合适。此外，尝试过3个隐含层，发现泛化性能不如2个隐含层强。

编写过程中遇到的一个很令人困扰的问题：发现每次对输入特性进行predict的时候预测结果都是一样的，不管输入特征是什么。就连训练仿佛也是找**一个**中间值，使得它和各标签的mean-square最小。原因：没有将输入向量归一化，导致了梯度消失。

最终使用全部的60000个含标签数据训练，经过30000次迭代（用时约12分钟）后，模型在训练集上的mean-square为14664.758。

1. 回归树

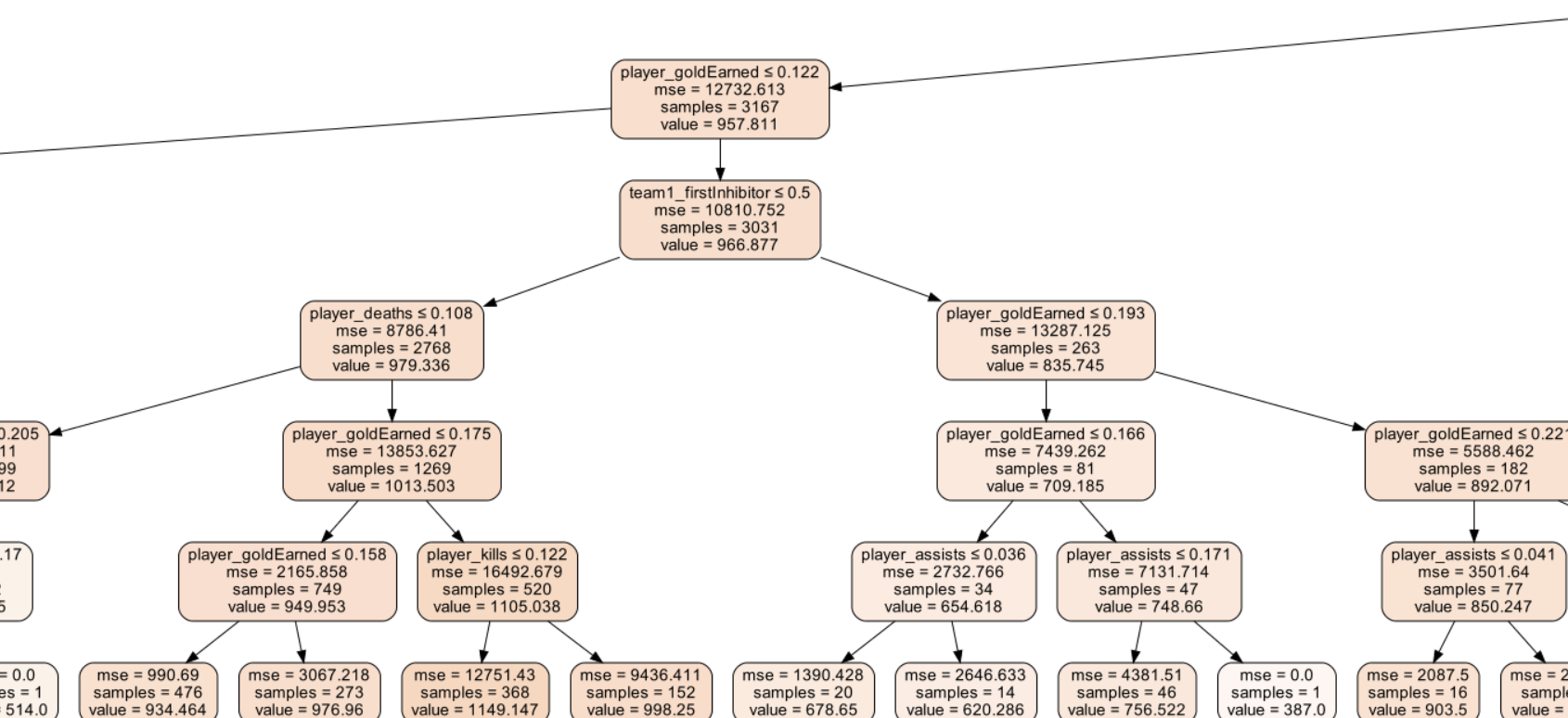
本次使用sklearn的回归树，特征选择为：将20组整型数据分成5组，每组数据的和以及6组布尔型数据共10组数据作为输入，模型可在10秒之内完成训练。

参数的调节：对于回归树，本次实验调节的是树的最大深度，具体步骤类似与人工神经网络的测试。测试结果如下：

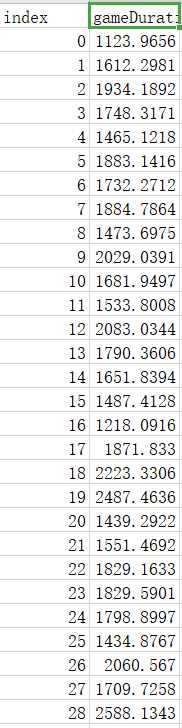
|  |  |
| --- | --- |
| Max-depth | Mean-square |
| 8 | 40723 |
| 9 | 39782 |
| 10 | 40356 |
| 15 | 46304 |

因此最佳最大深度为9。以为最大深度对所有含标签数据进行训练，得到测试数据集上的mean-square为16547.849275097902，略低于人工神经网络，但是泛化性能可能更好。

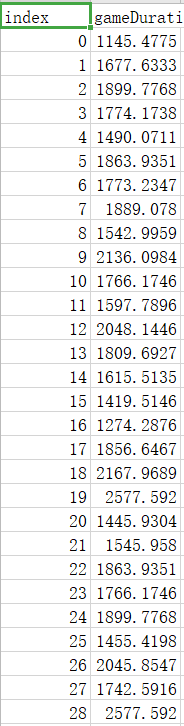
最后，使用graphviz对生成的回归树进行可视化，结果如下（由于原图太长，这里仅展示一部分）：



对于不含标签的数据进行预测后，得到两种方案所得结果的对比：



（人工神经网络）



（回归树）

可见两者数据非常接近。

注意：回归树代码直接运行可能可视化部分不通过，原因是graphviz需要从网上专门下载，并添加到环境变量。