Homework 2

张思源　　*21110850018*

November 19, 2021

1 Ex1

在导入 mnist 数据集后, 首先对数据集进行可视化, 只需要对 DataFrame 格式的图片矩阵化, 然后进行 reshape 即可，具体结果如下图所示:

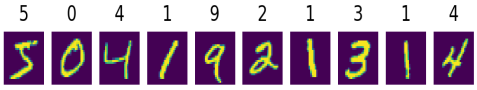

Figure 1: Some examples of mnist

之后, 对各个标签 (即数字 0 到 9) 对应的向量化图像的数目进行统计, 以判断是否存在数据差异过大的情况, 因为在数据差异过大的情况下直接对数据进行分类操作易出现问题. 经过统计, **各个标签对应的向量化图像的数目差距不大, 且由于数据维数较高 (784 维), 数据的凸包几乎不可能不相交, 故数据是线性不可分的, 这是 mnist 数据集最重要的特点**. 具体如下图所示:

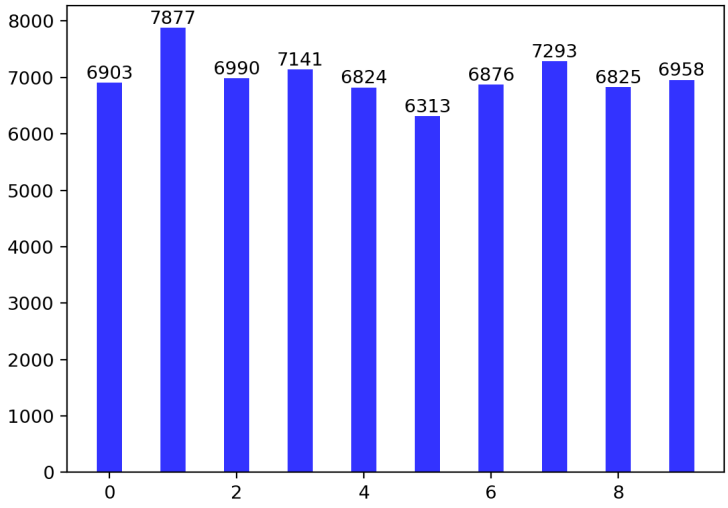

Figure 2: Number of data corresponding to different labels

为了更好说明这一步工作的重要性, 这里引入一个特殊的例子:

e.g. 1 *考虑同一个数据集即 mnisit 数据集, 这时任务考虑为二分类任务, 即区分对于一个手写数字, 是 0 还是非 0. 不妨假设 0 到 9 各个数字对应的图像数目是相等的, 此时我们训练分类器的方法为无论标签为哪个数字, 对应的输出结果均为非 0, 这样我们得到的结果准确率可达 90%, 但是召回率会非常低, 这说明当不同类规模差距较大时, 我们需要对其进行预处理.*

在此基础上, 利用留出法, 划出 70000 个数据中的后 10000 个数据为测试集, 其余数据为训练集, 选取 kernel 为 rbf, 即高斯核函数, 去训练模型, 并记录模型的训练与测试时间, 在 2.6 GHz 六核 Intel Core i7 的 cpu,python3.8 下训练时间和测试时间如下图所示:

```
训练时间为 375 s
在测试集上的精确率为 0.9792
测试时间为 156 s
```

Figure 3: Results of SVM

最终模型在测试集上的准确率为 0.9792, 可以看出模型较好的对 10 种手写数字图像进行了分类. 同时模型的混淆矩阵如下图, 可以看出模型同时有着较高的召回率和精确率.

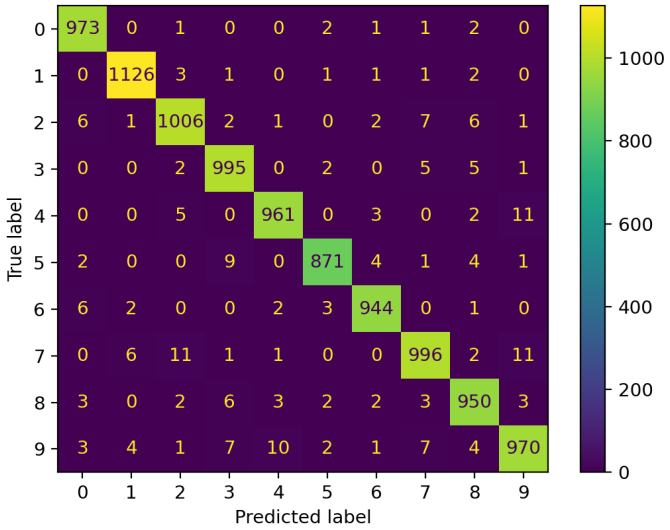

Figure 4: Confusion matrix

而对于全连接网络, 这里使用了 pytorch 构建全连接网络, 网络结构为 5 层, 具体结构如下图所示:

```
DistributedDataParallel(
  (module): fcn(
    (linear1): Linear(in_features=784, out_features=512, bias=True)
    (linear2): Linear(in_features=512, out_features=256, bias=True)
    (linear3): Linear(in_features=256, out_features=128, bias=True)
    (linear4): Linear(in_features=128, out_features=64, bias=True)
    (linear5): Linear(in_features=64, out_features=10, bias=True)
  )
)
```

Figure 5: Fully connected network

进而执行训练与测试, 可以得到训练时间与测试精确率如下图所示:

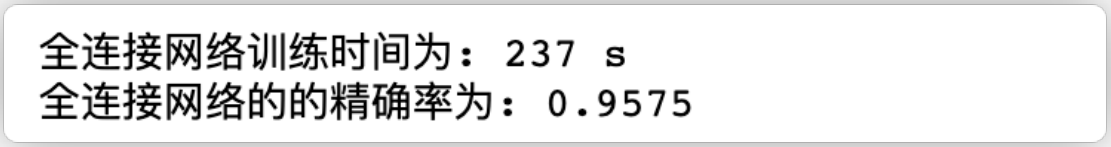

Figure 6: Results of fully connected network

可以看出,SVM 方法与全连接网络方法的精确率差距不大, 但是由于全连接网络可以调用服务器计算, 这里调用的类脑集群服务器 GPU15 节点 GTX1080Ti*3, 可以大大提高计算效率.

References

[1] 李航. 统计学习方法 [M]. 清华大学出版社, 2012.

[2] Goodfellow, Ian, et al. Deep Learning[M]. MIT Press, 2016.

[3] Peter Harrington, 李锐…[等. 机器学习实战 [M]. 人民邮电出版社, 2013.