第三次作业

李子龙
上海交通大学
计算机科学与工程系
logcreative@outlook.com

1 SVM 与神经网络

1.1 数据集信息

对两个数据集进行测试，其规模如表 1 所示。其中 madelon 数据集特征维度多，训练集大小大于测试集大小；ijcnn1 数据集特征维度相对较少，但是数据集规模大，测试集大小大于训练集大小。

表 1: 数据集信息

数据集	训练集大小	测试集大小	特征维度
madelon	2000	600	500
ijcnn1	49990	91701	22

1.2 与 MLP 的比较

数据读取实现于 `src/utils.py`，特征将会被首先归一化再进行训练。SVM 实现于 `src/svm.py`，具体参数为

Listing 1: `src/svm.py`

```
12      model = svm.SVC(kernel=kernel, C=C)
```

MLP 实现于 `src/mlp.py`，具体参数为

Listing 2: `src/mlp.py`

```
12      model = MLPClassifier(solver='lbfgs', alpha=1e-5, hidden_layer_sizes=(int(
        feat_size*5)), random_state=1, max_iter=feat_size*30)
```

表 2 展示了默认参数下 SVM 与 MLP 的效果。从准确率来看，SVM 的准确率在 madelon 多特征维度数据集上略高于 MLP，在 ijcnn1 向本较多的数据集上低于 MLP。训练时间

上 SVM 收敛需要的时间也偏长，当然从后文可以看到这个时间可以通过调节参数的方式缩短。

表 2: SVM 与 MLP

数据集	准确率		训练时间 (s)	
	SVM	MLP	SVM	MLP
madelon	**0.585**	0.583	71	19
ijcnn1	0.919	**0.961**	130	91

1.3 不同的核函数

表 3 展示了使用不同核函数的结果。其中在多维度的 madelon 上 linear 核的表现最好，但是训练时间较长，使用其它核会略微降低一点准确率，但是时间可以减少一个数量级。在少一些维度的 ijcnn1 上，rbf 的准确率最高，可以超过表 2 的 MLP，此时的 poly 核是更好的性价比选择。

表 3: SVM 不同核函数，$C = 1$

数据集	准确率				训练时间 (s)			
	linear	poly	rbf	sigmoid	linear	poly	rbf	sigmoid
madelon	**0.585**	0.578	0.582	0.583	76	5	7	5
ijcnn1	0.919	0.948	**0.968**	0.867	150	70	141	127

1.4 不同的 C

表 4 展示了不同的 C 对 linear SVM 的影响，C 将控制对软间隔的容忍度。可见其对准确率不会有特别大的影响，但是在训练时间上会有差异。同等准确率的情况下，对 madelon 而言，$C = 0.1$ 最好；对 ijcnn1 而言，$C = 0.1$ 最好。

表 4: 不同的 C，linear

数据集	准确率				训练时间 (s)			
	0.01	0.1	0.5	1	0.01	0.1	0.5	1
madelon	0.568	**0.585**	0.57	**0.585**	5	9	29	71
ijcnn1	0.918	**0.919**	**0.919**	**0.919**	87	104	121	188