

# 工科创 IV-J 平时作业 2

叶梓淳 520030910302

2023/3/6

## 1 被试依存条件

被试依存条件下，需要为 3 个 session 的每名被试单独训练模型，训练数据是本次 session 当前被试的 16 段视频，测试数据是剩余的 8 段视频，计算每个模型的正确率，最终得到 45 个模型的平均正确率。要求使用一对其他 (one-versus-rest) 法，即对每类样本都要训练一个模型，测试集分别在 4 类样本的模型上测试，声明为本类的概率最高的模型作为测试集类别。

调用 sklearn 中 svm.SVC 函数进行分类，分别使用核函数为线性核函数以及高斯核函数观察最后的平均准确率，

- 计算测试集准确率

---

```
1 def calc_accuracy(train_data, train_label, test_data, test_label):
2     #为每一类分别创建好train_label
3     train_label_sub = [[] for _ in range(4)]
4     for i in range(0,4):
5         for j in range (len(train_label)):
6             if(train_label[j] == i):
7                 train_label_sub[i].append(1)
8             else:
9                 train_label_sub[i].append(0)
10        train_label_sub[i] = np.array(train_label_sub[i])
11    train_label_sub = np.array(train_label_sub)
12
13
14    #创建好每一类的分类器
15    proba = []
16    for i in range(0,4):
```

```

17     predictor = svm.SVC(gamma='scale', C=1.0, kernel='rbf',
18                          probability=True)
19     #predictor = svm.SVC(gamma='scale', C=1.0, kernel='linear',
20                          max_iter=100000, probability=True)
21     predictor.fit(train_data, train_label_sub[i])
22     proba.append(predictor.predict_proba(test_data)[: ,1])
23     proba = np.array(proba)
24
25     #概率最大的类别作为预测类别
26     accuracy = 0.
27     for k in range(len(test_label)):
28         max_proba = 0
29         max_index = 0
30         for i in range(0,4):
31             if(max_proba <= proba[i][k]):
32                 max_proba = proba[i][k]
33                 max_index = i
34             if(max_index == test_label[k]):
35                 accuracy +=1.0
36
37     return accuracy/ len(test_label)

```

---

最终观察到线性核函数下分类器平均准确率为 0.595，高斯核函数下分类器平均准确率为 0.604。

对于测试集中不同类的样本，分别统计其精准率 (Precision) 与召回率 (Recall) 并取平均。如果某个模型  $TP + FP = 0$  或者  $TP + FN = 0$ ，则不计入平均精准率与召回率。得到如下表格：

label	参与的模型数量	精准率	召回率
0	39	0.820	0.715
1	24	0.674	0.817
2	40	0.643	0.726
3	36	0.725	0.688

可以看出 label 为 1 的测试数据较少，精准率较低，label 为 2 测试数

据较多，精准率较低。

## 2 被试独立条件

被试独立条件下，将每个 session 下每名被试的训练集数据与测试集数据合并，由于训练时间的限制，我将三次实验分开运行，取平均准确率。代码逻辑与上问相似，只是数据处理方式不同。得到三次 session 准确率分别是 0.391, 0.429, 0.413。平均为 0.411。

- 划分测试集和训练集，分三次训练

---

```
36
37 def get_test_train(data_g, label_g, index):
38     test_data = data_g[index]
39     test_label = label_g[index]
40
41     train_data = np.array([])
42     train_label = np.array([])
43
44     if(index == 0):
45         train_data = data_g[1]
46         train_label = label_g[1]
47         for i in range(2,15):
48             train_data = np.append(train_data, data_g[i], axis=0)
49             train_label = np.append(train_label, label_g[i], axis=0)
50     else:
51         train_data = data_g[0]
52         train_label = label_g[0]
53         for i in range(1,15):
54             if(i != index):
55                 train_data = np.append(train_data, data_g[i], axis=0)
56                 train_label = np.append(train_label, label_g[i],
57                                         axis=0)
57
58     return train_data, train_label, test_data, test_label
```

---

可以看出被试独立条件下准确率不如被试依存条件，这是因为不同 session 进行的时间不同，即使是同一个人得到的实验数据差别也较大，因此正确率

比被试依存条件更低。