

# Lab02 实验报告

## 项子扬 PB16001768

### 1. 数据预处理

要求为：将 2 个数据集分别随机平均分为 5 份并保存，要求每个被试人员出现在每份样本集中的数量均匀。后续使用 5 折交叉验证。可以对数据进行删减、降维、组合等。

### 2.SVM

要求为：使用非线性 kernel 的软边界 SVM 对 2 个数据集的 valence\_arousal\_label 进行分类，分析结果。

实验内容：

语言：Python3.6.2 环境：Windows10

数据预处理：采用 sklearn 工具包中的 preprocessing 工具包，将数据平均划分成 5 份，做 5 折交叉检验。

测试方面：根据训练结果对测试结果进行比对，计算得出正确率。核函数为 rbf（高斯核函数）。将其他输入属性各自归一化再作为 SVM 的输入属性。

经过测试惩罚系数  $C=6$  时分类成功率较高：

DEAP数据集：

交叉验证正确率:[0.35510204 0.46938776 0.46090535 0.45867769 0.41908714]

平均正确率:0.432631993718496

MAHNOB-HCI数据集：

交叉验证正确率:[0.49541284 0.48148148 0.47169811 0.42857143 0.45714286]

平均正确率:0.46686134488800235

$C$  过高或过低都会导致成功率降低：

$C=0.01$ ：

DEAP数据集：

交叉验证正确率:[0.35510204 0.35510204 0.35802469 0.35950413 0.35684647]

平均正确率:0.35691587565022564

MAHNOB-HCI数据集：

交叉验证正确率:[0.31192661 0.31481481 0.31132075 0.31428571 0.31428571]

平均正确率:0.3133267207215623

$C=100$ ：

DEAP数据集：

交叉验证正确率:[0.27346939 0.26530612 0.38271605 0.35123967 0.36099585]

平均正确率:0.32674541592613837

MAHNOB-HCI数据集：

交叉验证正确率:[0.33944954 0.39814815 0.44339623 0.2952381 0.44761905]

平均正确率:0.3847702117409578

代码说明：

```
from sklearn.svm import SVC
from sklearn import preprocessing
from sklearn.model_selection import cross_val_score

此部分为导入的库函数，SVC 为支持向量机库函数，cross_val_score 用于交叉验证。

with open('data/DEAP/EEG_feature.txt', 'r') as fo:
    for line in fo:
        line = line.strip('\n') #除去行尾的换行符
        line = line.split('\t') #以制表符分割数据
        S.append([float(i) for i in line])
D['EEG_feature'] = S
```

此部分为从文件中读取数据并存入字典中，利用 strip 和 split 函数将文件中的元素分割开来以便存储。

```
for x in range(0, length):
    X.append(D['EEG_feature'][x] + D['subject_video'][x]) #将两个文件中的属性合并作为输入存入X表中

此句句用于将属性一并存入 X 表，用作库函数的输入。

scaler = preprocessing.MinMaxScaler()
X = scaler.fit_transform(X) #将X表中的属性归一化

此语句用于将输入属性归一化，用于提高分类成功率。

Y = [10 * x[0] + x[1] for x in D['valence_arousal_label']] #将待分类标签存入Y表中
此语句将待分类标签的两个分量组合成一个（取值范围为 11、12、21、22）。

model = SVC(C = punish, kernel = 'rbf') #调用库函数进行分类
res = cross_val_score(model, X, Y, cv = 5) #调用交叉验证得到正确率

此语句调用库函数进行分类，并使用分类器对测试集验证得到正确率。
```

## 4. Naïve Bayes（手动实现）

要求为：使用带拉普拉斯修正的 Naïve Bayes 对 2 个数据的被试人员(subject id)，对 MAHNOB-HCI 数据集的情感标签进行分类，分析结果。

实验内容：

语言：C 语言 环境：Windows10

数据预处理方面：经分析，在对被试人员进行分类时，视频编号标签没有太大意义，故删去该标签；在对情感标签进行分类时，被试人员和视频编号标签没有太大意义，故删去这两个标签。valence\_arousal\_label 标签由于只有 1 和 1、1 和 2、2 和 1、2 和 2 四种取法，故组合为一个标签，取值为离散的{11,12,21,22}；DEAP 数据集中共有 1216 个样本，MAHNOB-HCI 数据集中共有 533 个样本，为了方便分组以及读取，故舍弃若干样本，将剩余的 1215 个样本和 530 个样本按被试人员编号(subject id)均匀分配至 5 个文本文件中。

测试方面：根据训练结果对测试结果进行比对，记录正确和错误的结果数，计算得出正确率。

### 1.DEAP 数据集的被试人员(subject id)分类：

第一次：

```
分类器运算结果为：
1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 4 4 4 4 4 4 4 5 22 2 2 2 5 5 5 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 10 8 8 8 8
8 8 9 9 9 9 9 9 9 10 10 10 10 10 10 10 10 11 11 11 11 11 11 11 11 12 12 12 12 12 12 13 13 13 13 13 13 14 14 14 1
14 14 14 14 14 15 15 15 15 15 15 15 16 16 16 16 16 16 16 16 17 17 17 17 17 17 17 17 18 18 18 18 18 18 19 19 19 19
19 19 19 20 20 20 20 20 20 20 21 21 21 21 21 21 21 21 22 22 22 22 22 22 22 22 23 23 23 23 23 23 24 24 24 24 24
24 25 25 25 25 25 25 25 26 26 26 26 26 26 26 26 27 27 27 27 27 27 27 27 27 28 28 28 28 28 28 29 29 29 29 29 30
30 30 30 30 1 30 31 31 31 31 31 31 31 31 31 32 32 32 32 32 32 32
分类正确数：239
分类错误数：4
正确率为：0.983539
-----
Process exited after 1.099 seconds with return value 0
请按任意键继续. . .
```

第二次：

```
分类器运算结果为：
1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 3 3 3 3 3 3 3 4 4 4 4 4 4 5 5 5 5 5 5 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 8 8 8 8 8 8
8 9 9 9 9 9 9 9 10 10 10 10 10 10 10 10 10 11 11 11 11 11 11 11 11 12 12 12 12 12 12 13 13 13 13 13 13 14 14 14 1
4 14 14 14 15 15 15 8 15 15 15 16 16 16 16 16 16 16 16 17 17 17 17 17 17 17 17 18 18 18 18 18 18 18 8 19 19 19 19
19 20 20 8 20 20 20 20 21 21 21 21 21 21 21 21 21 22 22 22 22 22 22 22 22 8 23 23 23 23 23 23 23 23 24 24 24 24 24
25 25 25 25 25 26 26 26 26 26 26 26 26 26 26 27 27 27 27 27 27 27 27 27 28 28 28 28 28 28 28 28 29 29 29 29 29 30
30 30 30 31 31 31 31 31 31 31 31 31 31 32 32 32 32 32 32 32
分类正确数：236
分类错误数：7
正确率为：0.971193
-----
Process exited after 0.893 seconds with return value 0
请按任意键继续. . .
```

第三次：

```
分类器运算结果为：
1 1 1 1 1 1 1 8 2 2 2 2 2 2 2 3 3 3 3 3 3 3 4 4 4 4 4 4 4 5 5 5 5 5 5 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 8 8 8 8 8 8
8 9 9 9 9 9 9 9 10 10 10 10 10 10 10 10 10 11 11 11 11 11 11 11 11 11 12 12 12 12 12 12 13 13 13 13 13 13 14 14 14
14 14 14 15 15 15 15 15 15 15 15 15 16 16 16 16 16 16 16 16 17 17 17 17 17 17 17 17 18 18 18 18 18 18 18 19 19 19 19
19 20 20 20 20 20 20 20 20 21 21 21 21 21 21 21 21 21 22 22 22 22 22 22 22 22 23 23 23 23 23 23 23 24 24 24 24 24
25 25 25 25 25 25 25 26 26 26 26 26 26 26 26 26 27 27 27 27 27 27 27 27 23 27 28 28 28 28 28 28 28 28 29 29 29 29 29
30 30 30 30 30 31 31 31 31 31 31 31 31 31 31 31 32 32 32 32 32 32 32
分类正确数：241
分类错误数：2
正确率为：0.991770
-----
Process exited after 1.196 seconds with return value 0
请按任意键继续. . .
```

第四次：

```
分类器运算结果为：
1 1 1 1 1 1 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 4 4 4 4 4 4 5 5 5 5 5 5 6 6 6 6 6 6 7 7 7 7 7 7 7 7 8 8 8 8 8 8
8 9 9 9 9 9 9 9 8 8 10 10 10 10 10 10 11 11 11 11 11 11 11 12 12 12 12 12 12 12 13 13 13 13 13 13 13 14 14 14 14 1
4 14 14 15 15 15 15 15 15 15 15 16 16 16 16 16 16 16 17 17 17 17 17 17 17 17 18 18 18 18 18 18 18 19 19 8 19 19 19
20 20 20 20 20 20 20 20 21 21 21 21 21 21 21 22 22 22 22 22 22 22 22 22 22 22 23 23 23 23 23 23 23 24 24 24 24 24 25
25 25 25 25 25 25 26 26 26 26 1 26 26 27 27 27 27 27 27 27 27 27 27 28 28 28 28 28 28 28 28 28 29 29 29 29 29 29 30 30 30
30 30 30 30 31 31 31 31 31 31 31 31 31 32 32 32 32 32 32 32
分类正确数：238
分类错误数：5
正确率为：0.979424
-----
Process exited after 0.7926 seconds with return value 0
请按任意键继续. . .
```

第五次：

```
分类器运算结果为：
1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 4 4 4 4 4 4 5 5 5 5 5 6 6 6 6 6 6 7 7 7 7 7 7 7 7 8 8 8 8 8 8
9 9 9 9 9 9 9 9 10 10 10 10 10 10 10 10 11 11 11 11 11 11 11 12 12 12 12 12 12 12 12 13 13 13 13 13 13 13 14 14 14 14 1
4 14 14 14 15 15 15 15 15 15 15 15 16 16 16 16 16 16 16 17 17 17 17 17 17 17 17 18 18 18 18 18 18 18 19 19 19 19 19 19
19 20 20 20 20 20 20 20 20 21 21 21 21 21 21 21 22 22 22 22 22 22 22 22 22 22 22 23 23 23 23 23 23 23 24 24 24 24 25
25 25 25 25 25 25 25 26 26 26 26 26 26 26 27 27 27 27 27 27 27 27 27 28 28 28 28 28 28 28 28 29 29 29 29 29 29 30 30 30
30 30 30 30 30 31 31 31 31 31 31 31 31 32 32 32 32 32 32 32
分类正确数：240
分类错误数：3
正确率为：0.987654
-----
Process exited after 1.027 seconds with return value 0
请按任意键继续. . .
```

5 次平均正确率为 98.27%。

2.MAHNOB-HCI 数据集的被试人员(subject id)分类：

第一次：

```
分类器运算结果为：
1 10 10 1 10 11 11 11 11 2 2 2 13 2 13 13 2 14 14 14 14 16 16 16 16 17 17 17 17 18 18 18 18 19 19 19 19 20 20 20 20 21
3 21 21 3 22 22 22 22 3 23 23 23 23 3 24 24 24 24 25 25 25 25 1 26 26 26 26 27 1 27 27 28 28 28 28 29 29 29 29 30 30
30 4 4 4 4 4 5 5 5 6 6 6 6 7 7 7 7 8 8 8
分类正确数：105
分类错误数：1
正确率为： 0.990566
-----
```

```
Process exited after 0.5329 seconds with return value 0
请按任意键继续. . . ■
```

第二次：

```
分类器运算结果为：
10 10 10 10 10 11 2 11 11 1 2 13 13 13 13 1 2 14 14 14 14 16 16 16 1 17 17 17 17 18 18 18 18 19 19 19 19 20 20 20 21
21 21 21 3 3 3 22 22 22 3 23 23 23 23 23 24 24 24 24 25 25 25 25 3 26 26 26 27 27 27 27 28 28 28 28 29 29 29 29 30 30
30 30 4 4 4 4 5 5 5 1 6 6 6 6 7 1 7 7 8 8 8
分类正确数：105
分类错误数：1
正确率为： 0.990566
-----
```

```
Process exited after 0.5262 seconds with return value 0
请按任意键继续. . . ■
```

第三次：

```
分类器运算结果为：
10 10 10 10 11 11 11 11 2 2 2 13 13 13 13 2 2 2 14 14 14 16 16 16 1 17 17 17 17 18 5 3 18 19 19 19 20 20 20 21 21
21 21 21 22 22 22 3 22 3 23 23 23 23 3 5 24 24 24 25 25 25 25 26 26 26 26 27 27 27 27 28 28 28 28 1 29 29 29 29 30 30 3
30 4 4 4 4 5 5 5 6 6 6 6 3 7 3 7 7 8 8 8
分类正确数：100
分类错误数：6
正确率为： 0.943396
-----
```

```
Process exited after 0.517 seconds with return value 0
请按任意键继续. . . ■
```

第四次：

```
分类器运算结果为：
10 10 10 10 11 11 11 11 2 2 13 13 13 13 14 14 14 14 16 16 16 17 17 17 17 1 18 18 5 18 19 19 19 19 20 20 20 20 2
1 21 21 21 3 22 22 22 22 22 3 23 23 23 23 1 1 24 24 24 25 25 25 25 26 26 26 26 27 27 27 27 28 28 28 28 29 29 29 29 30 30
30 30 30 1 4 4 4 5 5 5 5 6 6 6 6 7 7 7 7 5 8 8 8
分类正确数：103
分类错误数：3
正确率为：0.971698
-----
Process exited after 0.6163 seconds with return value 0
请按任意键继续. . .
```

第五次：

```
分类器运算结果为：
10 10 10 10 11 11 11 11 2 2 2 13 13 13 13 29 14 14 14 14 1 16 16 16 17 17 17 17 18 18 18 18 19 19 19 19 1 20 20 20 20 21
21 21 21 3 22 22 22 22 3 3 23 23 23 23 24 24 24 1 24 25 25 25 25 26 26 26 26 27 27 27 27 28 28 28 28 29 29 29 29 30 30 3
0 30 4 1 4 4 4 3 5 5 5 6 6 6 6 7 7 7 7 8 8 8 8
分类正确数：103
分类错误数：3
正确率为：0.971698
-----
Process exited after 0.7342 seconds with return value 0
请按任意键继续. . .
```

5 次平均正确率为 97.36%。

3.MAHNOB-HCI 数据集的 valence\_arousal\_label 标签分类：

第一次：

```
分类器运算结果为：
6 11 11 6 11 1 1 1 1 2 2 5 5 5 5 5 5 4 1 4 4 2 2 0 2 6 6 6 6 6 6 6 6 6 6 2 2 2 2 6 2 6 5 5 4 2 2 2 2 12 3 11 3 6 12
0 0 0 0 2 2 2 5 6 6 6 4 6 5 4 5 0 0 2 1 2 2 0 2 5 4 4 0 5 4 5 0 0 5 4 6 1 4 4 4 4 4 3 3 3 3 4 6 5
分类正确数：16
分类错误数：90
正确率为：0.150943
=====
```

```
Process exited after 0.5633 seconds with return value 0
请按任意键继续. . .
```

第二次：

```
分类器运算结果为：
6 4 4 4 4 3 0 3 3 6 2 5 0 0 0 6 2 4 4 4 5 11 11 11 6 6 6 6 6 6 6 6 6 6 6 6 2 2 2 11 2 3 4 0 6 6 12 11 3 3 12 5 4 5 6 4
4 1 0 2 11 2 2 3 6 6 6 6 5 5 0 0 5 1 0 1 5 2 2 2 2 4 4 2 2 4 2 0 6 6 6 6 6 3 3 3 3 2 6 3 3 4 6 6
分类正确数：10
分类错误数：96
正确率为：0.094340
=====
```

```
Process exited after 0.5854 seconds with return value 0
请按任意键继续. . .
```

第三次：

```
分类器运算结果为：
11 11 11 0 4 1 4 1 1 1 1 0 0 0 0 1 1 1 3 3 6 12 0 0 6 6 6 6 6 6 6 6 6 6 6 6 11 11 0 2 3 1 3 0 0 0 0 0 12 0 3 3 6 3 11
6 6 0 0 0 11 11 11 0 6 6 6 6 1 1 1 1 1 1 1 6 2 0 2 11 5 4 12 0 1 11 0 0 6 3 6 6 4 4 4 4 3 3 3 3 3 6 4 6
分类正确数：13
分类错误数：93
正确率为：0.122642
=====
```

```
Process exited after 0.4962 seconds with return value 0
请按任意键继续. . .
```

第四次:

```
分类器运算结果为:
1 1 1 1 3 3 3 3 3 1 1 5 5 5 5 1 1 1 3 4 0 11 0 6 6 6 6 12 12 12 1 12 6 12 6 6 11 11 11 2 3 3 0 1 12 3 3 5 5 2 6 5 11 3
11 2 4 1 1 1 11 11 11 1 12 12 4 12 1 5 5 5 5 12 0 2 2 2 2 1 1 1 4 1 12 1 1 1 6 4 6 6 3 3 3 3 3 3 3 6 6 3 1
分类正确数: 10
分类错误数: 96
正确率为: 0.094340
-----
Process exited after 0.4909 seconds with return value 0
请按任意键继续. . .
```

第五次:

```
分类器运算结果为:
0 0 6 6 1 1 1 1 0 12 11 5 5 5 5 11 0 6 4 6 6 0 0 11 6 6 6 6 6 6 6 6 6 1 6 6 12 11 11 11 4 0 4 0 6 12 12 12 12 6 6 6 6
6 5 0 0 6 1 0 11 11 11 11 6 6 6 6 0 0 5 1 12 1 1 1 12 12 12 2 0 0 0 0 4 6 1 4 0 4 6 6 6 4 3 3 3 3 3 3 3 6 6 6 6
分类正确数: 17
分类错误数: 89
正确率为: 0.160377
-----
Process exited after 0.6311 seconds with return value 0
请按任意键继续. . .
```

5 次平均正确率为 12.45%

结果分析:

在对两个数据集的被试人员进行分类时, 正确率较高 (97%), 而对 MAHNOB-HCI 数据集的情感标签进行分类时, 正确率较低 (12%), 可能是由于不同情绪在本实验中各属性上的影响有些类似, 导致分类失误; 也有可能是删去的被试人员和视频编号对实验有较大影响。

代码说明:

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
#define pi 3.14159265
#define M 424//训练样本数
#define N 106//测试样本数
#define C 13
```



此部分为宏定义，M 是训练集的大小，N 是测试集的大小，C 是待分类标签的取值范围大小。

```
struct SAM {  
    double feature[160]; //160 维脑电波特征数据  
    int id; //被试者编号  
    int label; //愉悦度标签，只能取11、12、21、22  
    int emo; //情感标签  
};
```

此为存储样本各个属性的结构体，feature 是脑电波特征，id 是被试人员编号，label 是经过合并的愉悦度-唤醒度标签，取值为{11,12,21,22}，emo 是情绪标签（只会在 MAHNOB-HCI 数据中出现）。

```
FILE *fpr,*fpw; //读写文件指针  
int i,j,k,cnt[C],right,res;  
int count1[C][4],count2[C][13];  
double pc[C],mean[C][160],var[C][160],sum[C][160],pt[C],max;  
double pxc1[C][4]; //愉悦唤醒度标签的条件密度  
double pxc2[C][13]; //情感标签的条件密度  
double correct;  
int h[N]; //测试结果  
struct SAM sample[M];  
struct SAM test[N];
```

此为程序变量定义。fpr、fpw 为读写文件指针，cnt 用于统计待分类标签的数量，count1、count2 用于统计离散标签的数量；pc 为先验概率，mean 为平均值，var 为方差，sum 用于计算平均值，pt 为测试样例计算的概率，pxc1、px2 为离散的条件密度，sample、test 为训练集数组、测试集数组。

```
//计算先验概率  
for(i=0;i<C;i++) cnt[i]=0;  
for(i=0;i<M;i++){  
    cnt[sample[i].emo]++;  
}  
for(i=0;i<C;i++){ //拉普拉斯修正  
    pc[i]=(double)(cnt[i]+1)/(M+C);  
}
```

此部分为先验概率的计算，cnt 自增时，若对 id 分类，下标为 sample[i].id-1，若对 emo 分类，下标为 sample[i].emo；由于是离散型的统计，需要进行拉普拉斯修正。

```

//计算条件概率
for(i=0;i<C;i++){//初始化
    for(j=0;j<160;j++) sum[i][j]=0.0;
}
for(i=0;i<M;i++){//遍历训练集
    for(j=0;j<160;j++){
        sum[sample[i].emo][j]+=sample[i].feature[j];
    }
}
for(i=0;i<C;i++){//平均值
    for(j=0;j<160;j++){
        mean[i][j]=(double)sum[i][j]/cnt[i];
    }
}
for(i=0;i<C;i++){//初始化
    for(j=0;j<160;j++) sum[i][j]=0.0;
}
for(i=0;i<M;i++){//遍历训练集
    for(j=0;j<160;j++){
        sum[sample[i].emo][j]+=pow(sample[i].feature[j]-mean[sample[i].emo][j],2);
    }
}
for(i=0;i<C;i++){//方差
    for(j=0;j<160;j++){
        var[i][j]=(double)sum[i][j]/cnt[i];
    }
}

```

此部分计算的是 160 维连续变量的平均值和方差，用于之后计算条件概率。由于是连续变量，不需要进行拉普拉斯修正。

```

for(i=0;i<C;i++){//初始化
    for(j=0;j<4;j++) count1[i][j]=0;
}
for(i=0;i<M;i++){
    switch(sample[i].label){//统计label标签
        case 11:{
            count1[sample[i].emo][0]++;
            break;
        }
        case 12:{
            count1[sample[i].emo][1]++;
            break;
        }
        case 21:{
            count1[sample[i].emo][2]++;
            break;
        }
        case 22:{
            count1[sample[i].emo][3]++;
            break;
        }
        default:printf("error!\n");
    }
}
for(i=0;i<C;i++){
    for(j=0;j<4;j++){//拉普拉斯修正
        pxc1[i][j]=(double)(count1[i][j]+1)/(M+4);
    }
}

```

此部分计算离散变量的条件概率，若对 id 进行分类，需要计算 label 和 emo，若对 emo 进行分类，需要计算 label。需要进行拉普拉斯修正。

```

for(i=0;i<N;i++){//计算概率估计
    for(j=0;j<C;j++){
        pt[j]=pc[j];//乘上离散条件概率
        for(k=0;k<160;k++){//乘上连续条件概率
            pt[j]*=exp(-pow(test[i].feature[k]-mean[j][k],2)/(2*var[j][k]))/sqrt(2*pi*var[j][k]);
        }
        switch(test[i].label){//根据测试样例的label选择相应的条件概率
            case 11:{
                pt[j]*=pxc1[j][0];
                break;
            }
            case 12:{
                pt[j]*=pxc1[j][1];
                break;
            }
            case 21:{
                pt[j]*=pxc1[j][2];
                break;
            }
            case 22:{
                pt[j]*=pxc1[j][3];
                break;
            }
        }
        //pt[j]*=pxc2[j][test[i].emo];//乘上情感标签条件概率
    }
}

```

此处对测试样例的各个分类的概率进行计算，在连续变量的计算时采用的是正态分布概率密度分布函数。

```
max=-1;
for(j=0;j<C;j++){
    if(pt[j]>max){
        max=pt[j];
        res=j;
    }
}
h[i]=res;//得出分类结果
}
```

此处是选出概率最大者，作为函数的分类结果。之后经过比对计算正确率。

## 5.LR

此部分实验的内容要求为：使用逻辑回归对 2 个数据集的 valence\_arousal\_label 进行分类，分析结果。

实验内容：

语言：Python3.6.2 环境：Windows10

数据预处理：采用 sklearn 工具包中的 preprocessing 工具包，将数据平均划分成 5 份，做 5 折交叉检验。

测试方面：根据训练结果对测试结果进行比对，计算得出正确率。将其他输入属性各自归一化再作为 LR 的输入属性。

结果：

DEAP数据集：

交叉验证正确率:[0.26938776 0.33469388 0.37860082 0.40909091 0.39419087]

平均正确率:0.35719284723170647

MAHNOB-HCI数据集：

交叉验证正确率:[0.36697248 0.42592593 0.45283019 0.32380952 0.46666667]

平均正确率:0.40724095642911634

代码说明：

```
from sklearn.svm import SVC
from sklearn import preprocessing
from sklearn.model_selection import cross_val_score
```

此部分为导入的库函数，SVC 为支持向量机库函数，cross\_val\_score 用于交叉验证。

```
with open('data/DEAP/EEG_feature.txt', 'r') as fo:
    for line in fo:
        line = line.strip('\n') #除去行尾的换行符
        line = line.split('\t') #以制表符分割数据
        S.append([float(i) for i in line])
D['EEG_feature'] = S
```

此部分为从文件中读取数据并存入字典中，利用 strip 和 split 函数将文件中的元素分割开来以便存储。

```
for x in range(0, length):
    X.append(D['EEG_feature'][x] + D['subject_video'][x]) #将两个文件中的属性合并作为输入存入X表中
```

此语句用于将属性一并存入 X 表，用作库函数的输入。

```
scaler = preprocessing.MinMaxScaler()  
X = scaler.fit_transform(X) #将X表中的属性归一化
```

此语句用于将输入属性归一化，用于提高分类成功率。

```
Y = [10 * x[0] + x[1] for x in D['valence_arousal_label']] #将待分类标签存入Y表中
```

此语句将待分类标签的两个分量组合成一个（取值范围为 11、12、21、22）。

```
model = LogisticRegression() #调用库函数进行分类  
res = cross_val_score(model, X, Y, cv = 5) #调用交叉验证得到正确率
```

此语句调用库函数进行分类，并使用分类器对测试集验证得到正确率。