**实验二 基于SVM的数据分类**

**一、实验目的**

通过编程调试实现SVM算法并在给定数据集上进行测试，实现数据分类。

**二、实验设备和环境**

**win10 64bit**

**gcc 4.9.2**

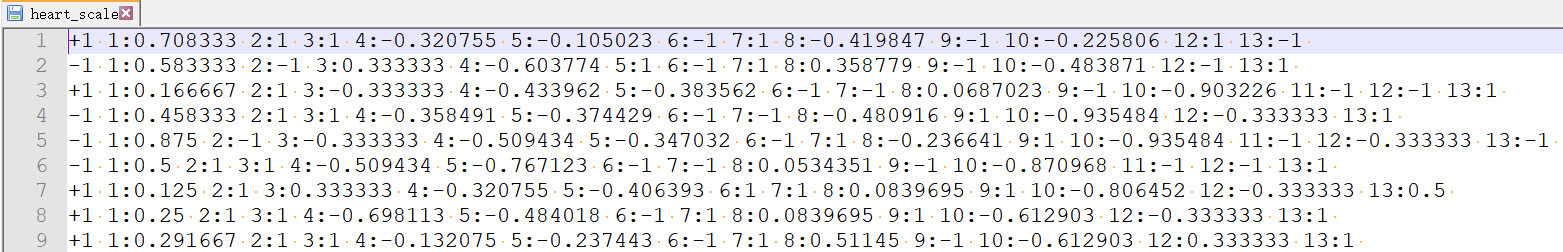
**三、实验内容**

自己编写或从网上下载SVM的源代码，调试运行SVM算法，研究其参数设定，并在给定数据集上完成测试，实现数据分类，并计算准确率（Precision）和查全率（Recall）。

**四、实验过程、结果及分析**

1.下载libsvm-3.24，发现windows子目录中有编译好的svm-train.exe和svm-predict.exe文件，可以直接使用。

libsvm-3.24中带有一个示例数据文件heart-scale，查看其格式，发现与本实验数据格式不符，因此需要调整数据格式。



调整数据格式的代码：

preprocess.cpp

1. #include<cstdio>
2. #include<cstring>
3. #include<vector>
4. **using** **namespace** std;
5. **void** preprocess(**char**\* data\_name,**char**\* label\_name,**char**\* output){
6. freopen(label\_name,"r",stdin);
7. vector<**int**>label;label.clear();
8. **int** tmp;
9. **while**(scanf("%d",&tmp)!=EOF)
10. label.push\_back(tmp);
11. **int** n=label.size();
12. **static** **char** buf[1<<16];
13. freopen(data\_name,"r",stdin);
14. freopen(output,"w",stdout);
15. **for**(**int** i=0;i<n;i++){
16. gets(buf);
17. **int** len=strlen(buf),cnt=0;
18. **if**(label[i])printf("+1 ");
19. **else** printf("-1 ");
20. **for**(**int** j=0;j<len;j++){
21. **if**(buf[j]!=' '){
22. printf("%d:",++cnt);
23. **int** k=j;
24. **while**(k<len&&buf[k]!=' '){
25. printf("%c",buf[k]);
26. k++;
27. }
28. printf(" ");
29. j=k-1;
30. }
31. }
32. printf("\n");
33. }
34. }
35. **int** main(){
36. **char** test\_data\_name[]="test\_sample.txt";
37. **char** test\_label\_name[]="test\_sample\_label.txt";
38. **char** test\_output[]="test\_sample\_data.txt";
39. preprocess(test\_data\_name,test\_label\_name,test\_output);
41. **char** train\_data\_name[]="train\_sample.txt";
42. **char** train\_label\_name[]="train\_sample\_label.txt";
43. **char** train\_output[]="train\_sample\_data.txt";
44. preprocess(train\_data\_name,train\_label\_name,train\_output);
45. **return** 0;
46. }

编译运行后生成train\_sample\_data.txt和test\_sample\_data.txt作为svm输入数据。

参照libsvm-3.24文档，编写命令训练并预测。

`svm-train' Usage

=================

Usage: svm-train [options] training\_set\_file [model\_file]

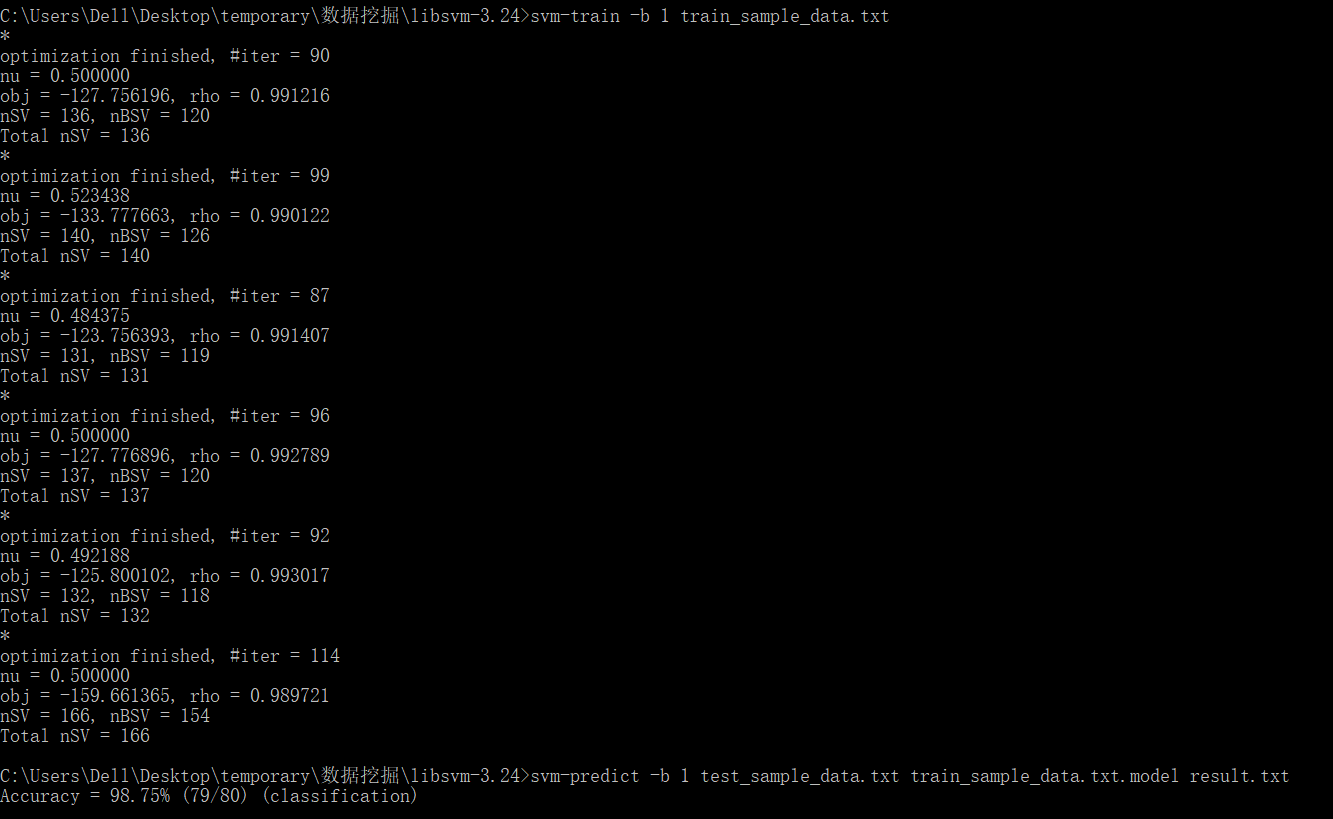
`svm-predict' Usage

===================

Usage: svm-predict [options] test\_file model\_file output\_file

options:

-b probability\_estimates: whether to predict probability estimates, 0 or 1 (default 0); for one-class SVM only 0 is supported



准确率Accuracy=98.75%（79/80）

观察result.txt，发现预测错误的是将第67个样本误判为鸟，实际非鸟。

鸟为正样本，非鸟为负样本。

|  |  |  |
| --- | --- | --- |
|  | **相关(Relevant),正类** | **无关(NonRelevant),负类** |
| **被检索到(Retrieved)** | true positives(TP)  20 | false positives(FP)  1 |
| **未被检索到(Not Retrieved)** | false negatives(FN)  0 | true negatives(TN)  59 |

精确率Precision:

查全率/召回率Recall: