西南科技大学

Southwest University of Science and Technology

本科毕业设计（论文）

题目：基于数据复杂度的基因微阵列应用研究

学生姓名：孙蒙新

学生学号：20130353

专　　业：软件工程（卓越计划）

指导教师：杨春明

学院(部)：计算机科学与技术学院

教务处制表

基于数据复杂度的基因微阵列应用研究

摘要

基因微阵列数据是一种具有“高维高噪小样本”特点的多分类数据，主要用于癌症的研究和治疗领域。纠错输出编码矩阵算法（Error Correcting Output Code）是一种典型的多类分类算法，它将多类分类问题转化为一系列相似的二分类问题，综合多个二分类器的输出结果来实现多分类。文章针对基因微阵列数据的多分类问题，提出了一种基于数据复杂度的纠错输出编码矩阵算法（Error Correcting Output Code based on Data Complexity）。该算法首先将多个类别随机分成两组，然后通过数据复杂度对应的局部贪心算法不断交换组内类别，降低两组之间的数据耦合程度；当两组之间的耦合程度不再降低后，整个交换过程结束，形成线性分离最大的两类。本文选取六种基因微阵列癌症数据作为实验数据，运用 Roc、T-test 和 Wilcoxon 等三种特征选择方法筛选重要数据特征，利用 SVM 和 NativeBayes 作为基分类器。同时，为了验证算法的有效性，采用 Ordinal、ECOCONE、DECOC 和 Forest-ECOC 作为对比算法。实验结果表明，该算法在所有数据集上的平均正确率达到 93%，部分数据集正确率达到 100%，算法的平均 Fscore 分数达到 70%。当实验数据特征数量不同时，该算法的整体准确率变化较小，分类表现稳定。

关键词：基因微阵列数据，ECOC，数据复杂度，多分类

The Analysis of Microarray Datasets based on Data Complexity

ABSTRACT

With characteristic of high dimension, high noise and small samples, DNA microarray data is kind of typical multi-classification data and it is mainly used for cancer research and treatment. As a popular method to solve multi-classification problem, Error correcting output codes(ECOC) transforms multi-classification problem into a series of binary-classification problem and combines their outputs to achieve multi-classification. In this paper, a new ECOC(ECOC-DC) is proposed to complete DNA microarray classification problem by exploring data distributions based on data complexity theory. In our framework, multiple classes are randomly divided into two separate groups at first, and then algorithms are designed to reduce overlapping between groups by exchanging classes, leading to optimal ECOC coding matrices with high discriminative capability by minimizing feature related complexity measure. Experiments are carried out on six microarray datasets and three methods: ROC, T-test and Wilcoxon are used to select important features. In addition, we take advantage of SVM and NativeBayes as base learner and Ordinal-ECOC, ECOCONE, DECOC and Forest-ECOC as comparisons. Experimental results demonstrate average accuracy rate of all datasets with our technique is 93%, and some accuracy rates are 100%, and average Fscores reach to 70%. Meanwhile, ECOC-DC shows small accuracy fluctuation when features of datasets change.

Key words: DNA microarray data, ECOC, data complexity, multi-classification

目录

第一章 绪论

1.1 基因微阵列简介

1.1.1 基因微阵列数据概述

DNA 微阵列技术又称基因微阵列技术，是二十世纪九十年代发展起来的一项具有里程碑意义的重大生物研究技术。近年来随着基因微阵列技术的不断深入研究，生物学家在一次实验中可以观测到全部或部分人体内的基因表达数据，更能清晰地观察出成千上万的基因间的相互作用。基因表达时 DNA 序列首先转录为 mRNA 序列，mRNA 序列再进行定量杂交配种生成 cDNA 或寡核苷酸阵列，这些反映细胞组织中特定的 mRNA 平均分子数的数据，称为微阵列基因表达数据，简称为基因微阵列数据[1]。

微阵列数据包含细胞的生理状态、功能信息和调控机制等重要基因表达信息，所以微阵列数据常用来解决生物学和生物医学等问题。常见的研究包括代谢机制、预测未知功能基因、诊断患者疾病状态、探索正常细胞组织与疾病组织的差异化等研究。此外，基因微阵列数据与癌症研究密切相关。正常组织病变与基因表达值改变通常同时发生，基因表达值或基因突变通常会导致癌症的发生。基因微阵列数据分析存在多方面的好处。一方面研究基因表达数据可以帮助医学工作者了解癌症的起源、恶化和病变的过程，帮助医疗研究者研究与癌症肿瘤的发生、恶化和迁移转变等致癌基因群。通过查看癌症过程中基因表达发生的改变，医疗人员可以确定临床治疗中癌症细胞的影响及肿瘤预后的决定性因素。另一方面通过基因表达数据预测肿瘤样本的类别，为医学工作者提供诊断参考，加快肿瘤诊断，进而对癌症进行有针对性的临床治疗。基因微阵列数据对疾病诊断和治疗具有深远意义，相关课题已经成为机器学习和模式识别领域的重要研究课题[2-4]。

目前，基因微阵列数据分析方法仍然是科学人员的研究重点和难点，常用的研究方法可分为有监督学习法和无监督学习法两种。有监督学习法需要基因表达数据和预先已知的类别标签信息，在学习过程中，利用类别标签有目的地训练模型。无监督学习只需要基因微阵列数据本身。这种学习方法不需要使用任何先验知识和假设就能利用模型发现数据集的内在结构模式。无监督学习方法主要应用包括聚类法和降维法。有监督学习的主要应用则包括分类法和回归法。

1.1.2 基因微阵列数据特征

基因微阵列数据是形状为 M*N 的二维关系矩阵。矩阵的每一行代表一个基因在所有样本上的表达强度。矩阵的每一列代表一个样本所有基因的表达强度。由于基因微阵列技术实验环境特殊、实验步骤琐碎以及实验过程不稳定，基因微阵列原始数据存在大量影响因素，原始数据需要经过数据清理、关联和转化等过程才能被使用，基于以上过程，微阵列数据的特征如下：

（1）小样本高维度

即使基因微阵列技术发展迅速，至今为止，基因实验仍然面临操作复杂、费用昂贵的艰难情况，实验样本数量通常不超过 100，然而微阵列芯片中需要测量的基因数量十分巨大，通常在 5,000-15,000 数量级，由于基因数量远大于样本数量情况造成一个著名的问题-“维数灾难”。这种维度与样本之间 “高维度，小样本”的特点给科研人员带来极大的困扰。

（2）高噪音

微阵列实验环节复杂多变，实验环境标记效率、扫描属性、温湿度等实验环境的差别等不稳定因素都会影响实验数据的测量值。此外，经过芯片扫描得到的原始数据需经过预处理

才能使基因微阵列数据尽可能切实地反映被测样本在生物学上的差异。然而，因为预处理技术本身的不足，导致处理过后的数据部分基因表达值失真，数据带有极大的噪音成分。

总之，微阵列基因表达数据的特点是“高维高噪小样本”，微阵列基因表达数据分析研究面临着巨大的挑战。

1.2 研究背景

1.2.1 基于纠错输出编码的基因微阵列现状研究

基因微阵列数据的诊断与分类最大的挑战在于数据具有超多噪音、超高维度、样本量小的特点，这些特征使许多模式识别方法不能应用在微阵列数据中。除此之外，基因微阵列数据的各个维度之间通常存在着极为复杂的关系，每个维度都不是独立存在的，这样的情况在分析过程中可能引发三类问题：一是计算复杂度过大，二是有用的变量被隐藏，三是多类别识别分类器集成困难。目前，已经有大量学者展开对基因微阵列数据的研究。现今最为流行的方法是首先筛选数据特征以降低特征维度减轻实验困难，然后再通过机器学习的算法进行研究。纠错输出编码算法是一种典型的多分类器融合技术，中心思想是将一个多分类问题分解为若干个相似的二类分类问题，利用多个二分类器的融合模型解决多分类问题。纠错输出编码算法提出后吸引大量学者的关注，不少学者进行了相关的研究和探索。文献[5]的作者将不确定值引入纠错输出编码矩阵，拓展单分类器的混合输出。Crammer[6]等人则利用连续编码的概念优化编码矩阵，并将编码的设计问题转化为约束优化问题，并且证明找到一种最优的离散的编码设计问题是一个 NP 问题。此后，参考文献[7]的作者开始提出一种判别式 ECOC 改进编码算法，利用互信息作为类别空间分割标准，利用连续向前浮动搜索策略实现数据划分。目前，已有大量研究证明纠错输出编码算法在基因微阵列数据上取得了令人满意的效果[8-10]。其中文献[5]和[6]中的算法都属于数据独立型算法，主要采用随机方式构成编码矩阵，构建过程存在大量随机元素。这种方法存在两个缺点：一方面，随机元素导致矩阵多样，实验结果迥异，分类性能参差不齐，算法的稳定性令人堪忧。另一方面，随机方法忽略数据内部分布特性，其编码矩阵不能准确表达数据的分类情况，错误的分类会增加分类器的学习成本，降低分类器的判别能力。此外，文献[7]中的判别式 ECOC 改进编码算法属于一种依赖数据的算法。互信息是一种变量间相互依赖的量度，利用先验概率和后验概率准确计算出两个随机变量间的相关性。这种算法存在三个缺点：（1）算法需要大量的样本数据产生先验概率，基因微阵列数据样本数量稀少，此算法无法在基因微阵列数据上发挥最佳性能。（2）计算先验概率需要大量的时间。（3）概率计算只能保证尽最大努力实现最优分类，但是不能保证一定实现最优分配。

1.2.2 基于数据复杂度的基因微阵列数据现状研究

数据的特征直接影响分类器预测结果，特征鲜明的数据集可以使分类器充分学习类别特点，降低分类错误率。研究证明，复杂程度较高的数据集会干扰最近邻分类器 KNN 的性能，分类结果准确性大大降低。因为分类器预测结果的容错能力和正确分类能力强烈依赖于数据的复杂程度，所以出现大量关于数据特性的研究。文献[11]首次定义了二分类的复杂度测度，提出使用特征相关量度评估数据耦合程度。文献[12]不仅总结回顾了之前的复杂度测度还提出了两种新的测度：特征空间分离测度和领居分离测度，经过 UCI 数据实验证明，新的复杂度测度可以从特征和空间距离角度估计数据内部分布情况，帮助我们了解基因微阵列数据的重叠程度。数据复杂度从特征向量、空间向量和几何向量三个方面展示数据集的分布情况，数据特征对复杂度测量影响不大，所以研究人员开始将数据复杂度分析应用和基因微阵列数据结合起来，文献[14-16]分别展示了相关的研究成果。之前的工作中，研究人员只是将评估了数据集的复杂程度，但是没有对数据分类做出研究。此后，文献[13]中，作者首次利用复杂度构建决策树从而产生了一种新的二分类算法。构建二叉树时，在父节点分裂成两个子节点时，复杂度作为一种评估标准判断当前两个节点的分配是否是最优分配，从而得到两个

差异最大的两个子节点。这种利用数据复杂度作为分类标准的二分类算法比传统的二分类算法性能更优越，分类效果更好。受此方法启发，我们设想将数据复杂度应用到多分类问题中，利用多分类框架将问题化简为二分类问题，那么多分类问题可以得到解决。

1.3 论文研究内容与结构

论文的主要研究内容主要包括三个方面：

（1）特征选择方法。基因微阵列数据具有“高维高噪小样本”的特点，处理这类数据首要的方法是选用合适的特征选择方法筛选重要特征，降低数据维度，减小时间和性能消耗。

（2）基于数据复杂度的纠错编码矩阵算法。在纠错输出编码矩阵算法框架中，利用数据复杂度测度实现最优二分类分配，综合所有二分类分配方案形成新的多分类模型。

（3）基因微阵列数据的分类应用。基因微阵列数据是著名的多分类问题之一，研究基因微阵列数据分类可以攻克模式识别的难题，促进癌症研究进展。

本文的结构安排如下：

第一章主要通过绪论介绍了基因微阵列数据的来源和特征以及纠错输出编码矩阵和数据复杂度在基因微阵列数据上的一些研究成果，以及论文的主要研究内容和组织结构。

第二章详细介绍了三种重要的特征选择算法：T-test、ROC 和 Wilcoxon 并且分析特征选择算法在基因微阵列上的实验结果。

第三章主要介绍了纠错输出编码矩阵算法的编码、解码和二分类器的原理。

第四章首先介绍了本次改进算法中使用的两种类型的复杂度。然后介绍了本文提出的基于数据复杂度的纠错输出编码矩阵算法，详细叙述该算法的框架设计，之后又介绍了不同复杂度测度对应的局部贪心算法。

第五章介绍了实验中使用的基因微阵列数据、评价指标和基因微阵列实验结果，并针对结果展开详细的分析。

第六章总结论文的整体工作并提出未来研究的发展方向。

第二章 特征选择算法

在分类模型训练过程中，基因微阵列数据的高维度高噪音小样本的特点常使得模型出现过拟合问题，为此需要对基因微阵列数据进行特征选择和特征提取工作。特征选择主要选择有区分度的特征，移除样本的常数特征和随机特征等噪音特征，得到更简单的分类模型，提高模型的泛化能力。

2.1 特征选择算法原理

2.1.1 ROC 方法

ROC（receiver operating characteristic）是用描述分类器命中率与错误率之间的关系的一种统计决策理论[17]。Spackman 将此方法引入到机器学习的领域，并在模式识别领域具有广泛的应用。ROC 曲线利用命中率与错误率构成了一个二维空间，在分类的过程中应用非参数的检验方法，通过比较 ROC 曲线所含面积以和分类器斜率从而选择最佳的界限值，反映出某一特征对于分类的贡献度。

2.1.2 T-test 方法

t-test 方法是假设检验的一种经典方法，是一种用于样本含量较小，总体标准差未知的正态分布的一种检验方法[18]。将样本数据假设为 T 分布，利用公式推断差异发生的概率，利用概率的计算确定两样本均值是否存在显著差异。这种方法要求假设样本是符合正态分布，同时总体的均值已知，样本均值和方差均可得到的情况。T-test 的检验统计量的计算如下：

$$t = \frac{\bar{x} - u_0}{s/\sqrt{n}} \tag{2-1}$$

其中：$\bar{x}$表示样本均值，u_0表示假设的总体均值，s 表示样本标准差，n 表示样本数量。通常情况下，可以令 $u_0=0$，从而检验两样本均值是否有显著区别。

T-test 原本是用于假设检验的一种检验方法，在这里是用来比较两个不同样本的分布情况，进而筛选出具有明显差异的维度并进行进一步的维度筛选。

2.1.3 Wilcoxon 方法

Wilcoxon 测试方法结合 T-test 检验方法和误判阈值（TNOM）法的特点[19]，避免噪音对 T-test 检验方法产生影响，具体公式如下：

$$s(g) = \sum_{i \in N_0} \sum_{i \in N_1} I(x_j^g - x_i^g \leq 0) \tag{2-2}$$

其中：I(·)是一种判别函数，当且仅当括号内的逻辑表达式为真时，函数值为 1，反之函数取值为 0。x_j^g是样本 I 在基因 g 中的表达值，N_0和N_1分别表示不同类别中样本索引集合。此值可表示同一基因在两个类别中表达差异程度，该值越接近 0 或越接近最大值，表示对应基因对分类影响越大。判断基因重要性的依据如下式所示：

$$\mathrm{q(g)} = \max(\mathrm{s(g)}, n_0, n_1 - \mathrm{s(g)}) \tag{2-3}$$

2.2 特征选择结果

针对基因微阵列维度过多的问题，实验中采用本章列出的三种特征选择方法筛选数据重要特征。本小节主要展示部分筛选结果。

2.2.1 ROC 特征选择结果

我们在选取 ROC 方法在 Breast 和 Cancers 数据集的第一个样本上筛选结果作为展示结果，如表 2-1 和表 2-2 所示。

表2-1 ROC在Breast数据集的特征选择结果

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
1	0.16247	26	1.53137	51	0.07331	76	-1.06332
2	-2.15949	27	-0.90148	52	-1.31882	77	1.17997
3	-1.34054	28	0.16622	53	-1.43195	78	3.58284
4	-1.34729	29	-0.86776	54	-0.33204	79	1.32083
5	0.22691	30	-0.26835	55	1.62952	80	0.31008
6	-1.32331	31	-0.18968	56	-0.44443	81	1.67822
7	0.41347	32	-0.86252	57	-1.20643	82	0.57681
8	-0.82281	33	0.49289	58	-1.45968	83	2.33458
9	0.53635	34	-0.57930	59	0.63001	84	-2.07557
10	-1.70768	35	1.64750	60	-2.07932	85	-0.69918
11	-1.84555	36	0.67272	61	-1.67097	86	-1.32556
12	0.66897	37	1.92323	62	1.83032	87	1.40100
13	-1.03635	38	0.93945	63	1.66923	88	1.25714
14	0.87727	39	0.51462	64	0.12875	89	-1.15623
15	-1.56607	40	-0.29757	65	-0.78984	90	-0.18668
16	-2.29960	41	0.35054	66	1.89926	91	-0.53359
17	0.71992	42	-0.72390	67	-0.08404	92	2.18997
18	0.46218	43	2.07983	68	1.59880	93	5.52568
19	-0.10427	44	-0.07430	69	-1.17046	94	-0.24962
20	0.38650	45	-0.93370	70	1.37028	95	-1.46342
21	0.54310	46	2.04761	71	-2.03811	96	-1.53310
22	1.67223	47	-0.97191	72	-0.09527	97	-0.57930
23	3.47645	48	1.14400	73	0.41872	98	-1.18245
24	1.25040	49	0.68920	74	-0.97640	99	-0.10427
25	0.45094	50	-1.44544	75	0.76563	100	3.73494

表2-2 T-Test在Cancers数据集的特征选择结果

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
1	2.67415	26	-0.28654	51	-0.24992	76	-0.27016
2	1.69593	27	0.26473	52	-0.00705	77	0.54037
3	1.58027	28	-0.23065	53	-0.19017	78	2.96425
4	-0.26823	29	-0.31546	54	0.50182	79	-0.28269
5	-0.27594	30	-0.20173	55	0.31967	80	0.43050
6	-0.30582	31	-0.27402	56	3.62925	81	0.55001
7	0.05848	32	13.06357	57	-0.25667	82	0.55290
8	-0.27305	33	0.09800	58	-0.29233	83	2.05541
9	-0.16415	34	-0.31642	59	-0.07452	84	0.72059
10	-0.29425	35	-0.08223	60	0.16835	85	-0.10439

续表 2-2

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
11	-0.14969	36	0.32352	61	-0.19788	86	-0.28076
12	-0.30582	37	0.33990	62	-0.34726	87	-0.19402
13	-0.26052	38	0.17992	63	0.11053	88	0.47194
14	-0.29522	39	-0.27883	64	-0.18053	89	-0.29425
15	0.17317	40	-0.13138	65	-0.17764	90	-0.03500
16	0.40255	41	1.07044	66	0.08932	91	-0.17378
17	0.31196	42	0.06427	67	1.10128	92	0.51627
18	1.29404	43	-0.28462	68	-0.13138	93	1.29885
19	-0.29040	44	-0.17282	69	-0.24318	94	0.82083
20	-0.32317	45	-0.18053	70	-0.26920	95	2.15468
21	-0.31353	46	0.43532	71	-0.32124	96	0.31485
22	-0.23354	47	0.64735	72	-0.26920	97	-0.01958
23	-0.16415	48	-0.31642	73	-0.07452	98	0.72059
24	-0.29425	49	-0.08223	74	0.16835	99	-0.10439
25	-0.14969	50	0.32352	75	-0.19788	100	-0.28076

2.2.2 T-test 特征选择结果

我们在选取 T-test 方法在 Breast 和 Cancers 数据集的第一个样本上筛选结果作为展示结果，如表 2-3 和表 2-4 所示。

表2-3 T-test在Breast数据集的特征选择结果

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
1	2.59682	26	1.82508	51	3.73494	76	2.21919
2	2.53163	27	2.66050	52	0.57681	77	2.13303
3	6.56865	28	3.78140	53	-0.40547	78	1.15974
4	1.50739	29	2.59607	54	0.46218	79	2.02288
5	5.08511	30	6.81365	55	5.55939	80	1.89926
6	2.24691	31	2.85232	56	3.40827	81	1.44820
7	3.52740	32	1.12602	57	2.41100	82	1.12077
8	2.62679	33	3.51316	58	-0.07430	83	-0.29757
9	1.75540	34	1.68497	59	-2.29960	84	1.00089
10	1.35529	35	3.64878	60	3.13778	85	3.79114
11	3.33634	36	2.43198	61	1.53137	86	0.68920
12	2.85906	37	1.67223	62	-0.34403	87	4.22946
13	5.38631	38	3.11006	63	1.92323	88	0.32881
14	2.33907	39	2.04761	64	-0.82281	89	2.17948
15	2.86131	40	3.07185	65	0.22691	90	4.41452
16	2.92125	41	1.53661	66	3.74618	91	1.24365
17	3.58284	42	3.00367	67	1.67598	92	1.63552
18	3.77091	43	0.16622	68	1.44221	93	1.81759
19	3.83160	44	0.49289	69	1.25564	94	3.81811
20	2.74517	45	0.94919	70	3.08908	95	-0.35227
21	1.68047	46	2.75191	71	3.84059	96	-1.34729

续表2-3

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
22	2.65976	47	-0.26835	72	1.51189	97	-0.10427
23	4.57486	48	2.93623	73	1.64750	98	1.59880
24	2.47843	49	0.53635	74	-1.84555	99	1.84456
25	-2.15949	50	2.18997	75	2.09556	100	0.45094

表2-4 T-Test在Cancers数据集的特征选择结果

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
1	1.58027	26	1.13309	51	0.73505	76	0.32352
2	14.47453	27	1.77303	52	0.81793	77	-0.26920
3	1.18320	28	0.23293	53	-0.27787	78	0.13462
4	11.67093	29	-0.29811	54	1.73833	79	0.55001
5	10.44212	30	-0.28076	55	0.50856	80	3.80369
6	14.38586	31	1.73159	56	1.36054	81	0.44303
7	13.60714	32	1.12923	57	0.15486	82	0.09511
8	1.43089	33	1.87230	58	1.17260	83	1.07237
9	7.60672	34	0.31099	59	0.40255	84	0.55772
10	15.64936	35	0.43532	60	0.89504	85	13.06357
11	10.15878	36	-0.25763	61	-0.06584	86	0.77938
12	8.70445	37	3.15796	62	-0.34726	87	0.09800
13	1.07044	38	3.59166	63	0.50182	88	1.32102
14	0.86709	39	0.20112	64	-0.27883	89	-0.27594
15	9.53715	40	0.10378	65	0.09896	90	-0.30582
16	2.97099	41	0.21172	66	0.54037	91	-0.29907
17	3.02400	42	0.00740	67	0.17317	92	-0.21137
18	-0.28847	43	-0.27402	68	-0.28654	93	-0.18053
19	0.06427	44	0.92491	69	0.29364	94	-0.23354
20	2.38502	45	1.29404	70	0.95768	95	0.66180
21	-0.08030	46	-0.26823	71	-0.30582	96	1.39138
22	1.80772	47	-0.20173	72	-0.14969	97	0.04596
23	1.42704	48	2.05541	73	0.02572	98	-0.13041
24	3.87308	49	1.22079	74	1.76628	99	0.43628
25	-0.29715	50	0.44977	75	-0.08223	100	1.31813

2.2.3 Wilcoxon特征选择结果

我们在选取Wilcoxon方法在Breast和Cancers数据集的第一个样本上筛选结果作为展示结果，如表2-5和表2-6所示。

表2-5 Wilcoxon在Breast数据集的特征选择结果

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
1	-1.32331	26	-0.97640	51	-0.32155	76	-0.71566
2	-2.15949	27	-1.20643	52	-0.88125	77	-0.30507
3	-1.34729	28	-0.70442	53	-1.01087	78	0.22691

续表 2-5

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
4	-1.34054	29	-0.26835	54	1.64750	79	-0.73739
5	0.16247	30	-1.31882	55	-1.53310	80	-2.85855
6	-0.82281	31	-1.55109	56	1.53137	81	0.41347
7	-1.03635	32	-0.24962	57	-1.32556	82	-1.00787
8	-0.90148	33	-0.59054	58	0.66897	83	0.63001
9	-0.57930	34	-1.14499	59	-1.67097	84	-1.04309
10	-0.44443	35	-2.45469	60	0.87727	85	-1.25962
11	-0.18968	36	-2.29960	61	-1.35103	86	-0.94494
12	-1.70768	37	-1.15623	62	-1.43195	87	-0.49613
13	-0.33204	38	-1.00937	63	-0.92995	88	0.76563
14	-0.86776	39	-1.37052	64	-1.03485	89	-2.07932
15	-0.97191	40	-1.18245	65	-0.83554	90	-0.87525
16	0.16622	41	-1.48740	66	0.57681	91	-1.02361
17	-1.84555	42	-0.59653	67	-0.86252	92	-0.38149
18	-1.56607	43	-0.18668	68	-0.70517	93	-1.68296
19	-1.44544	44	-0.98090	69	0.67272	94	-0.07130
20	-0.69918	45	-0.56881	70	-0.91646	95	-0.52610
21	0.49289	46	-0.53359	71	-0.96591	96	1.33431
22	-0.93370	47	-2.03811	72	-1.66423	97	-1.08130
23	-0.07430	48	-1.13525	73	-0.68419	98	-1.10528
24	-0.57930	49	-1.80284	74	-2.55809	99	-1.11352
25	-0.29757	50	-1.21467	75	1.66923	100	-0.60477

表 2-6 Wilcoxon 在 Cancers 数据集的特征选择结果

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
1	2.67415	26	-0.23354	51	-0.17764	76	-0.29233
2	1.69593	27	-0.29233	52	-0.28462	77	-0.13041
3	-0.16415	28	-0.17282	53	-0.17571	78	-0.18053
4	-0.32317	29	0.17992	54	-0.19017	79	-0.12945
5	-0.27305	30	-0.30582	55	0.16835	80	-0.17860
6	-0.27594	31	0.11053	56	0.31967	81	-0.02633
7	-0.30582	32	-0.13812	57	-0.19788	82	-0.34726
8	-0.23065	33	-0.24992	58	13.06357	83	-0.23546
9	-0.29522	34	0.06427	59	0.55290	84	0.00259
10	0.05848	35	-0.26920	60	-0.00705	85	-0.32124
11	1.58027	36	-0.18053	61	-0.27980	86	-0.26920
12	-0.29425	37	-0.22968	62	0.43532	87	0.38520
13	-0.26823	38	-0.09090	63	-0.28269	88	0.31485
14	-0.31353	39	-0.13138	64	0.82083	89	-0.13138
15	0.31196	40	0.33990	65	0.95864	90	0.08932
16	0.64735	41	-0.24318	66	-0.17571	91	0.12306

续表 2-6

特征编号	特征值	特征编号	特征值	特征编号	特征值	特征编号	特征值
17	1.29404	42	-0.31642	67	0.01511	92	-0.26438
18	0.32352	43	-0.08030	68	2.27033	93	-0.01573
19	0.09800	44	-0.14969	69	0.51627	94	1.10128
20	-0.26052	45	-0.28076	70	-0.28654	95	-0.35304
21	-0.08223	46	-0.27402	71	-0.21330	96	-0.16704
22	0.17317	47	-0.25763	72	-0.25667	97	0.04596
23	-0.31546	48	0.26473	73	-0.30196	98	-0.29425
24	0.40255	49	-0.27883	74	-0.20173	99	-0.24703
25	-0.29040	50	-0.05428	75	0.54037	100	3.62925

2.3 结果分析

从上述一系列表中可以看出，基因微阵列数据精确度非常高，常常达到小数点后 15 位，且每个基因数值非常小，数值维持在标准值 0，这为特征筛选和分类过程带来极大的困难。为了确认结果的相似程度，我们将各个特征选择算法结果进行两两比较，计算每对方法的相似特征个数，结果展示在表 2-7 中。观察表 2-7 发现，ROC 和 T-test，T-test 和 Wilcoxon 方法得到的相似特征个数为 0，这说明使用 T-test 方法和 ROC 及 Wilcoxon 方法得到的数据完全不同。基于 ROC 和 Wilcoxon 方法的结果中存在一个相似特征，说明 ROC 方法和 Wilcoxon 的选择结果具有一定的相似性。为了避免单个实验出现偶然情况影响实验判断，我们同时也将 Cancers 数据集的特征选择结果的相似特征的比较展示在表 2-8 中。观察表 2-8，发现 ROC 和 T-test、T-test 和 Wilcoxon 结果的相似特征各增加 1 个，而 ROC 和 Wilcoxon 相似特征个数为 2。综合所有结果来看，ROC 方法和 T-test 方法实现结果存在部分相似，Wilcoxon 的结果和其他两种方法的结果完全不同。

表2-7 Breast特征选择结果比较

	算法比较名称	相同特征个数
Breast	ROC--T-test	0
	ROC—Wilcoxon	1
	T-test-Wilcoxon	0

表2-8 Cancers特征选择结果比较

	算法比较名称	相同特征个数
Cancers	ROC--T-test	1
	ROC—Wilcoxon	2
	T-test-Wilcoxon	1

2.4 本章小结

本章主要介绍了 ROC，T-test 和 Wilcoxon 三种特征选择方法的原理和实验结果，通过实验结果分析，我们发现 ROC 和 T-test 特征选择结果存在一定的相似性，而 Wilcoxon 方法所得结果和前两种方法所得结果完全不同。

第三章 纠错输出编码矩阵算法

3.1 编码算法

ECOC 框架主要包含编码和解码两个过程。编码过程主要将多分类问题分解为一系列的二分类问题。不同的编码算法会产生不同的编码矩阵，编码算法主要分为数据无关的方法和数据相关的方法。数据无关的方法在构建编码矩阵的过程与数据集分布特征相互独立，而对于数据相关的算法，具有不同分布特征的数据集将得到不同的编码矩阵。

数据无关的 ECOC 算法是早期基于 ECOC 算法改进的一种流行趋势。这种算法意在产生一种通用算法，排除数据本身对算法产生的干扰。在众多的数据无关 ECOC 方法中：一对一(OVO)，一对余(OVA)，密集随机算法(DR)和稀疏随机算法(SR)是被普遍认可的四种算法。

样本具有 K 个类别时，一对一方法的中心思想是尽可能将问题分解成每对类的子问题。一对一方法训练时需要在任意两个类别的样本之间训练一个独立的二分类学习器，因此，对于 K 个类别的样本需要分类器个数为：K*(K-1)/2。这些分类器的输出投票结果即为预测类的预测结果。

对于 K 各类别的问题，一对余分类器中心思想是从每个类中学习训练一个分类器。一对多方法在训练时需要依次将某个类别的样本归为一类，其他剩余的样本归为另一类，两类之间进行比较并训练出一个二分类器。一对余方法共需要 K 个二分类器。

密集随机法和稀疏随机法都使用随机生成原理产生编码矩阵，然后运用矩阵判别原理验证编码矩阵是否合理，根据某个特定的准则选出一个最优的编码矩阵。区别在于：密集随机法生成的编码矩阵是二元编码，只包含{1，-1}，而稀疏随机法生成的编码是稀疏编码矩阵，为三元编码{-1,0,1}。因为编码包含元素个数不一致。密集随机法通常需要预先设定分类器个数，一般为 10*logK 个二分类器，而稀疏随机法通常需要 15*logK 个分类器，不同的标准会去掉不符合条件的分类器。这四种算法如图 2-1 所示。

在大量学者针对 ECOC 方法展开研究后发现，基于数据的 ECOC 改进策略真正考虑到了数据对于编码的行列分离性和基分类器性能的影响。基于数据的编码方式针对性强，对难分类数据表现出强大的适用能力。同时，数据的复杂程度影响二分类器的性能。耦合程度大的数据会使 KNN 等弱分类器的分类效果大打折扣，充分考虑数据对二分类器的影响才能真正提高 ECOC 算法的性能。基于数据相关的方法是 ECOC 算法研究的一个重点，目前比较经典的方法有：DECOC[7]和 ECOC-ONE[16]以及 ECOC-Forest[20]方法。

3.2 解码算法

经过训练集数据，根据已得到的相应的编码矩阵，对于测试集数据，只需将数据输入这个编码矩阵当中，每一个单独的分类器都能够得到一个分类标签，多个分类器生成一个类别向量。通过比较输出向量与编码矩阵中各个类别的标签向量，距离最小的码字所对应的类别标签即为分类样本的标签。解码算法一直是 ECOC 的研究热点之一，可以分为三类。第一类是基于输出编码和目标编码距离的解码策略。第二类是基于概率的解码策略。第三种是基于模式空间的解码策略。第一类是最常用的解码方式，汉明距离解码是最常见的此类解码方法之一，其解码规则简单，准确率高，且具有纠错能力。Dietterich 在首次提出 ECOC 方法时就利用该解码方法，但随着分类问题复杂程度的增加，汉明距离有时得不到理想的答案，出现解码正确率偏低甚至完全错误的情况，之后有作者在文献[21]中针对此问题进行了专门研究并指出汉明距离解码在面对复杂分类问题的弊病所在，同时引出了差异性度量作为解码的新准则。文献[22]提出了利用欧式距离代替汉明距离作为融合评价函数，提出了欧式距离解

码。因为欧式距离解码准确率更高，纠错能力更强，在实验中我们采用欧氏距离方法作为解码算法。

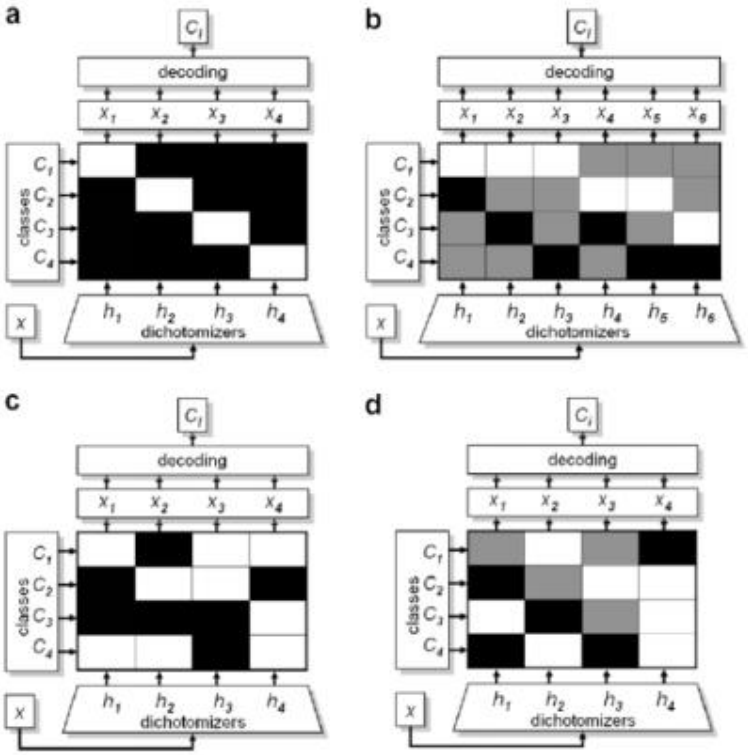


图 3-1 OVO、OVA、DR 和 SR 四种算法示意图

3.3 基分类器

ECOC 中的基分类器是决定分类实验效果的关键因素。一个二分类器对应编码矩阵的一列，一个二分类器需要输出自己对类别的判断，所有二分类器的输出共同组成最后的类别判断。二分类问题中，每个分类器只能把样本分为正负两类。常用的分类器有：决策树分类器、支持向量机分类器、最近邻分类器、朴素贝叶斯分类器、神经网络分类器等。其中，k-近邻分类器和神经网络分类器也可以解决多分类问题。本章主要对后面实验中使用到的贝叶斯算法和支持向量机算法进行详细解释。

3.3.1 贝叶斯分类器

贝叶斯分类器首先计算样本的先验概率，利用贝叶斯公式计算出其后验概率，即待测样本属于某一类的概率，该待测样本类别取决于具有最大后验概率的类别。贝叶斯分类器工作流程具体如图 3-2 所示。

如图所示，贝叶斯分类器训练和检测样本分为三个阶段：

（1） 第一阶段——准备阶段，这个阶段是贝叶斯分类重要的准备阶段，主要根据训练样本的特征属性确定重要的特征属性。同时还需要对每个特征属性进行划分，对其中一部分待分类项进行分类，形成训练样本空间。本阶段通过输入所有待分类数据直接输出特征属性和训练样本。

（2） 第二阶段——训练阶段，这个阶段主要是计算每个类别在训练样本中的出现频率及每个特征属性属于对每个类别的条件概率，并记录结果。需要输入特征属性和训练样本，输出分类器。

（3） 第三阶段——应用阶段。这个阶段需要使用分类器对分类项进行分类，对已经训练好的分类器输入待分类项，可直接输出待分类项与类别的映射关系。

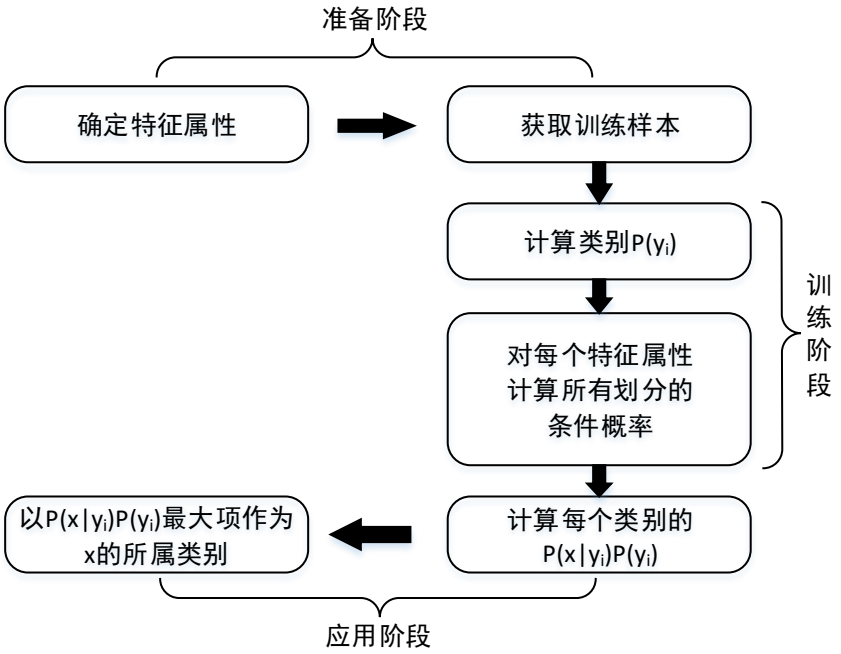


图 3-2 贝叶斯算法流程示意图

3.3.2 支持向量机分类器

支持向量机(Support Vector Machine)是 90 年代发展的一种基于统计学习理论的 VC 维和结构风险最小原理的二分类算法。算法通过寻求结构化风险最小提高特定训练样本的学习精度和无错误地识别任意样本的能力[23]，它主要解决小样本、非线性及高维模式识别问题。支持向量机算法通过寻找特征空间上间隔距离最大的线性分类器的方式解决小样本输入不足问题，算法具有超高学习能力，因而能在统计样本较少的情况下，依然能够获得良好的统计规律。算法主要分为两个步骤：

（1）原始线性不可分特征数据映射到高维特征空间；

（2）在高维空间建立一个最大间隔超平面，用以将一类样本与其他样本分离。示例如图 3-3 所示。

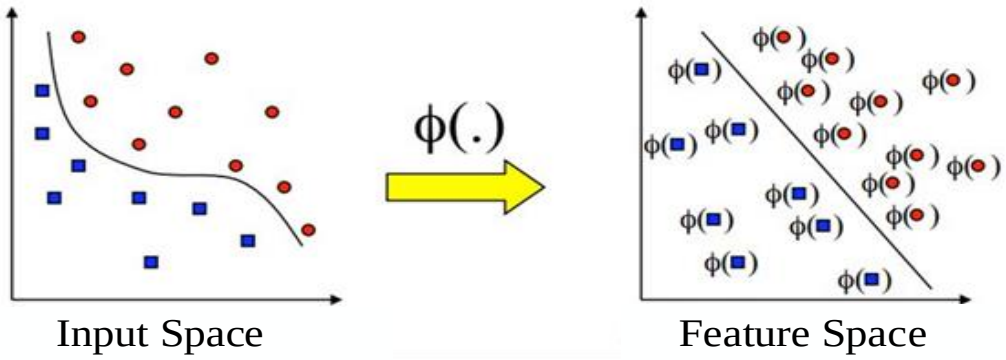


图 3-3 支持向量机空间转换示意图

支持向量机将输入空间通过某些非线性变换转换为高维特征空间，改变输入空间数据特征，以便找到数据之间的关联。在低维空间中，向量集通常是非常难划分的，如果将它们映射到高维空间则问题将可以得到解决。但同时计算复杂度随着维度的增加也随之增加。在高

维空间中关键点是确定非线性映射函数关系形式、函数参数、特征空间维度，同时“维数灾难”将成为计算高维特征空间运算最大的障碍。核函数正是解决计算复杂度的最佳方法。在选取核函数时，归纳误差最小的核函数就是最适合的核函数。图 3-4 是常用的线性核核函数和高斯核函数的对比。

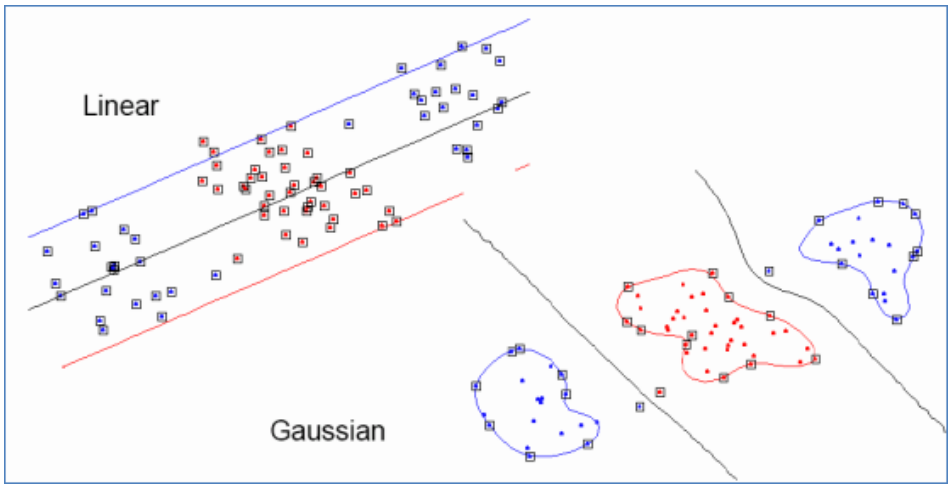


图 3-4 支持向量机中线性核函数和高斯核函数对比图

3.4 本章小结

本章主要详细介绍了纠错输出编码矩阵算法的原理，并分别介绍了算法中的编码和解码策略。本章在第一小节中仔细分析编码算法中的四种主要算法：OVO、OVA、Dense Random 和 Sparse Random 原理和需要的分类器数目，并附上流程图。在第二小节中介绍了不同的解码策略，说明汉明距离解码和欧式距离解码的优缺点。第三小节对纠错输出编码算法中的分类器进行了说明，并列出相应的公式和流程图。

第四章 基于数据复杂度的纠错输出编码算法设计

4.1 数据复杂度测度

数据复杂度主要描述数据内部规律性和非规律性特点，按照复杂度测度的几何和关联特性对其分为两类：类的特征重叠测度和类的分离度测度。接下来按照复杂度特性分别介绍这几种测度。

4.1.1. 特征重叠测度

这类复杂度主要研究单个特征维度的类别有效性及贡献度，其主要包含具体的三种测度：

（1）最大费希尔判别比（F1）

F1 通过单个特征值的均值和方差和衡量两个类之间的差异度，计算公式如下：

$$f_{i,j}^{k}=\frac{(\mu_i-\mu_i)^2}{\sigma_i^2-\sigma_j^2} \tag{4-1}$$

其中：μ_i，μ_j，σ_i和 σ_j 分别是类 i、j 的第 k 个 feature 的均值和方差。很明显$f_{i,j}^{k}$只和类的单个维度相关，所以对于多维数据，$f_{i,j}^{k}$的最大值被定义为 F1，其公式如下：

$$F1_{i,j}=MAX\left(f_{i,j}^{k}\right),i\in[1,\mathrm{d}] \tag{4-2}$$

F1 表示两类之间的紧密程度，F1 值越小，准确分类难度越大。

（2）重叠区域容量（F2）

对于一个二分类问题，F2 聚焦类的重叠区域和总长度，测量每个特征的重叠效率。原始测度定义为单个特征值的得分问题，但是对于多分类问题，F2 值往往趋近于 0，所以我们将每个特征值的 F2 和定义为新的 F2 测度。具体公式如下：

$$\mathrm{F2}=\sum_{k=1}^{d}\frac{\min\left(\max\left(f_i^k\right),\max\left(f_j^k\right)\right)-\max\left(\min\left(f_i^k\right),\min\left(f_j^k\right)\right)}{\max\left(\max\left(f_i^k\right),\max\left(f_j^k\right)\right)-\min\left(\min\left(f_i^k\right),\min\left(f_j^k\right)\right)} \tag{4-3}$$

其中：$\max\left(f_i^k\right)$和 $\min\left(f_i^k\right)$分别是类 i 的第 k 个 (k=1,⋯,d)特征的最大值和最小值。

F2 值越小说明两类之间的重合程度越小，两类之间存在鲜明的界限，其值越大说明类分类界限模糊。

（3）最大贡献特征（F3）

这个复杂度测度主要描述类分离过程中贡献最大的特征。每个特征都存在最大值和最小值，最大值和最小值分别构成当前特征的上边界和下边界。被单个特征分离的样本数量为特征的贡献度，对于多维度数据，F3 则被定义为最大贡献度，其定义如下所示：

$$F3_{i,j}=MAX\left(1-\frac{A_i^k-B_j^k}{T}\right) \tag{4-4}$$

其中：A_i^k 和B_j^k 分别是第 i 类和第 j 类在 k 维度上重合的样本数量，T 表示样本总数。

F3 值越小表示维度的重合的样本数量越多，分类难度越大。

4.1.2. 特征分离测度

这类测度主要通过类的边界判断类的分离程度。在这类测度中所有特征的有效性都被综合考虑从而避免了单个特征的影响。这类复杂度测度主要包含：

（1）类内最近邻距离和类外最近邻的距离比值（N2）

对于随机分布的数据来说，大多数样本会随机出现在类边界位置。而对于线性分离的类来说，相互分离的两个类之间会存在一个清晰的界限。类内最近邻距离和类外最近邻的距离的值为所有样本的类内最近邻居的间隔距离总和和类外最近邻居间隔距离总和的比值。

$$N2 = \frac{\sum_{i=1}^{n} intraDis(p_i)}{\sum_{i=1}^{n} interDis(p_i)} \quad (4\text{-}5)$$

其中：intraDis(p_i)是每个数据样本与其类内最近邻居之间的距离，interDis(p_i)是每个数据样本与其类外最近邻居之间的距离。n 表示数据集样本的总数量。

N2 测度的范围为$[0,+\infty)$，N2 值越小说明在同一特征空间内样本彼此疏远，整体分布稀疏。其值越大说明整体样本分布紧凑。

（2）基于最近邻分类器的错误率（N3）

最近邻分类器的错误率被定义为 N3 测度。这个数据复杂度主要描述数据样本之间的紧密程度。另外，考虑到训练样本分布不均衡问题，采用留一法保证训练结果的准确性和稳定性。

N3 的取值范围为[0,1]，N3 值越小，说明最近邻分类效果越好，数据不同类别之间存在明显的分类界限。

4.2 算法框架设计

实验采集数据存在人工误差和采集错误等问题，基因微阵列数据须经过数据清洗和缺省值填充等数据清洗步骤。针对“高维高噪小样本”的特点，我们采取数据特征选择方法降低基因微阵列的数据维度，以便选取重要特征数值。

ECOC 的性能关键是编码矩阵的优劣，编码矩阵的容错性和差异性越大，分类器越能学习样本特点，实验准确率越高。所以我们采用数据复杂度的方法形成行列分离程度大的编码矩阵。基于数据复杂度的 ECOC 编码算法是通过一种树形分布的方法形成编码矩阵。树的生成方式如下：首先从根节点开始，将包含多类的类别组随机划分成两部分，为了保证两类之间的平衡性，两部分尽可能包含相同数目的类别。随后利用数据复杂度测度评估两类之间的紧密程度，通过不断调整两类中包含的类别来减小两类之间的复杂度，当复杂程度达到局部最优时整个调整过程结束。保证每个二分类节点是当前最佳的分类方法后，将节点编码为 1 或-1（没有参与分类的节点编码为 0），继而形成编码矩阵的一列，全部类别划分结束后形成编码矩阵，之后利用基于数据复杂度的编码矩阵训练分类器并形成分类模型，然后利用该分类模型预测样本。在实验过程中，为了保证编码矩阵的生成过程不受上下文干扰，实验算法的每次分类过程都是独立于上个过程，具体的算法流程如下图：

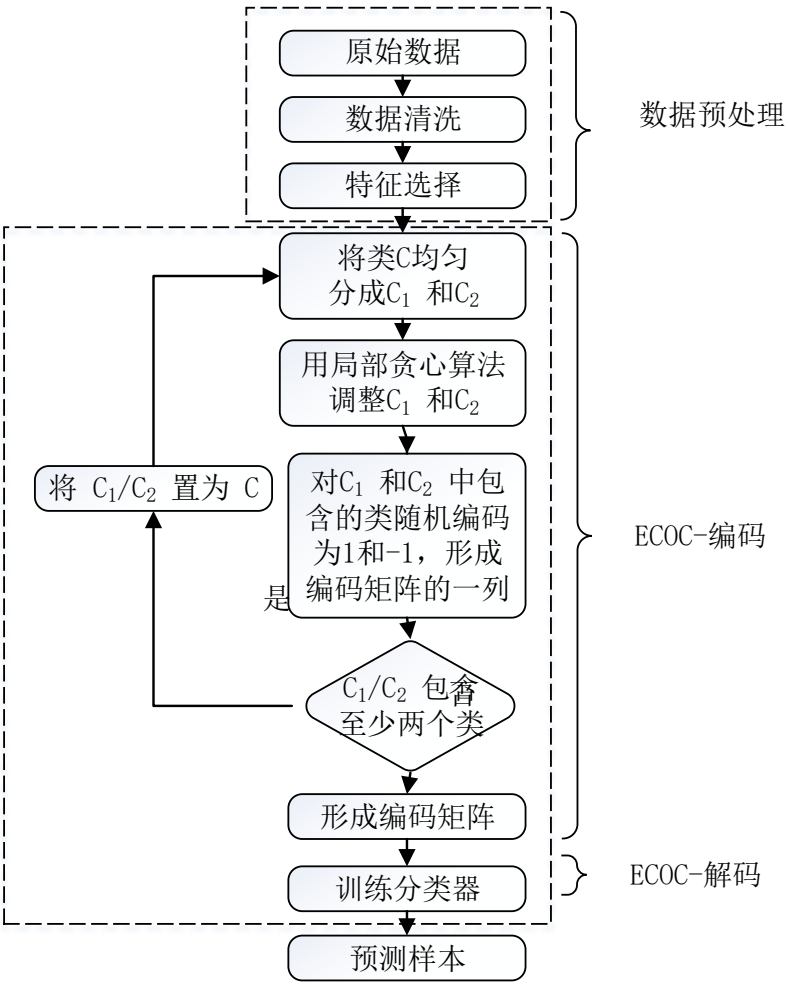


图 4-1 基于数据复杂度的纠错输出编码矩阵算法整体流程图

具体编码算法的伪代码如下：

Algorithm 1 Frame of minimized DC algorithm

Randomly divide classes in C into two groups and note as C_1 and C_2
DC ← data complexity score between C_1 and C_2 based on N2/N3
c_1/c_2 ←Call Algorithm2/ Algorithm3/ Algorithm4/ Algorithm5 with input (C_1, C_2)
C_1'/C_2' ← C_1/C_2 exchange c_1/c_2 for c_2/c_1
DC' ← data complexity score between C_1' and C_2' based on N2/N3
If DC' is less than DC **do**
  Update DC, C1 and C2 with DC', C1' and C2'
  GOTO step 2-4
Else If C1 or C2 contains one more classes **do**
  GOTO step 1
Else
  Return C1 and C2

Algorithm2-Algorithm5 是依照不同的数据复杂度测度原理设计的四种不同的局部调整算法。算法采用贪心策略，不断尝试搜索最不合适的类别重新进行分配，以达到最优分配。为避免穷举产生的时间和资源消耗，只要复杂度测度值不再变化则算法调整过程结束，当前分配被认定为最优分配。

4.3 局部贪心算法设计

两类分布问题是局部优化问题，单纯随机划分存在巨大的偶然性，只有不断尝试分配才能达到最优动态平衡，所以我们根据五种数据复杂度测度的原理分别设计了相应的局部贪心算法，在不考虑算法上下文的情况下，实现当前最好的类分配。

4.3.1. 基于类的特征重叠测度局部贪心算法

影响基于最大费希尔判别比的编码矩阵算法的最大因素为特征的均值和平方差的比值，因为数据整体耦合程度较大，方差值偏小，所以平方差的影响可以被忽略，那么只需要最大化两类均值差的平方和就可以达到增大 F1 值的目的。对于平方和问题，我们考虑一种，另一种只需要交换就可以实现相同的目的。假设两类存在数据整体上的大小关系，对数值大的一类来说，用特征值大的子类换掉特征值小的子类，对数值较小的一类来说，用特征值小的子类换掉特征值大的子类，可扩大类间的距离。基于这一思想，我们设计第一种局部贪心算法，算法具体如下：

Algorithm 2 (Based on F1)

1. Calculate DC for C1/C2 using formulas(1) and (2)
2. cmin=argmin(di,1),di,1 D1,cmax=argmax(dj,2), dj,2 D2
3. Exchange cmin/cmax to form C1’/C2’
4. Calculate DC’ for C1’/C2’ using formulas(1) and (2)
5. cmax=argmax(di,1),di,1 D1,cmin=argmin(dj,2), dj,2 D2
6. Exchange cmax/cmin to form C1"/C2"
7. Calculate DC" for C1”/C2” using formulas(1) and (2)
8. **IF** max(DC’, DC")>DC
9. **IF** DC’>DC"
10. replace C1/C2 with C1’/C2’, DC=DC’
11. **ELSE**
12. replace C1/C2 with C1”/C2”, DC=DC”
13. **GoTo** step 2
14. **ELSE**
15. **Stop** the Algorithm2

对于重叠区域容量和最大贡献特征测度，首先通过比较平均特征值确定类的整体数值的大小，然后按照类似第一种局部贪心算法思想调整类别分布。

Algorithm 3 (Based on F2/F3)

1. Calculate DC value under formula(3)/formula(4)
2. Get the mean value m1/m2 of D1/D2
3. L=argmax(m1,m2), S=argmin(m1,m2)
4. cmax = argmax(di,L),di DL (DL is train data of CL)
5. cmin = argmin(dj,S),dj Ds (Ds is train data of Cs)
6. Exchange cmax/cmin to form C1’/C2’
7. Calculate DC’ value
8. **IF** DC’<DC
9. replace C1/C2 with C1’/C2’, DC=DC’
10. **GoTo** step 2
11. **ELSE**
12. **Stop** the Algorithm3

4.3.2. 基于类的分离度测度的局部贪心算法

根据 N2 复杂度测度的原理计算每个类别与类内的最近邻距离总和和类外最近邻距离总和的比值确定两类中距离最远的类别为需要调整的类别。为了简化计算最近邻的计算方式，用类别中心值表示类别位置。所有距离计算依靠类别中心值完成。

Algorithm 4 (based on N2)

1. Set C_1 and C_2 as C and CA
2. X/XA is set of average Features of samples belonging to c in C/CA
3. $R(i) \leftarrow \frac{\sum_{j=1}^{length(C)} \sqrt{(X(i)-X(j))^2}}{\sum_{k=1}^{length(A)} \sqrt{(X(i)-XA(k))^2}}$(i=1, … length(C))
4. $c_1 \leftarrow \underset{i}{\text{argmax}}(R(i))$
5. Set C_1 and C_2 as C and CA, and **GOTO** step2-5, $c_2 \leftarrow \underset{i}{\text{argmax}}(R(i))$
6. **Stop** the Algorithm4

类似 N2 思想，N3 采用样本之间的距离和作为判别标准。与其他类距离最远的类成为需要重新分配的类别。

Algorithm5 (based on N3)

1. Set C_1 as C
2. X is set of average Feature of samples belonging to c in C
3. $DSUM(i) \leftarrow \sum_{j=1}^{length(C)} \sqrt{(X(i)-X(j))^2}$(i=1, … length(C))
4. $c_1 \leftarrow \underset{i}{\text{argmax}}(DSUM(i))$
5. set C_2 as C and GOTO step2-5, $c_2 \leftarrow \underset{i}{\text{argmax}}(DSUM(i))$
6. **Stop** the Algorithm5

4.4 本章小结

本章首先介绍基于数据复杂度的纠错编码矩阵算法中的使用到的五种复杂度测度，描述了原始测度和本文测度的创新部分，以创新测度为基础，有针对性引出本章后续小节提出的基于数据复杂度的纠错输出编码矩阵算法。之后详细介绍了基于数据复杂度的 ECOC 算法原理并给出了详细的伪代码，使读者能够清楚了解算法原理，在了解的基础上能全面重现算法。之后本章介绍了五种基于数据复杂度的局部贪心调整分配算法，仔细说明算法设计思路，并附上相应的代码和流程图。

第五章 实验设计与结果分析

5.1 基因微阵列数据集

实验数据都是典型的基因微阵列数据，为了有效针对多分类问题进行研究，所有实验数据包含多个类别。所有数据的类别数量、特征数量、训练样本数量和测试样本数量如下表所示：

表5-1 实验数据

#	#数据集	#类别数量	#特征数量	#训练样本数量	#测试样本数量
1	Breast	5	9216	54	30
2	Cancers	9	12,533	86	74
3	DLBCL	6	4026	58	30
4	Leukaemia2	3	12582	57	15
5	Leukaemia3	7	12582	215	112
6	Lung1	3	7129	64	32

为了充分验证我们算法的有效性，我们选取了类别数目不同的数据集。数据集的类别数目为 3-9 之间。需要注意的是，我们研究的是多分类问题，所有数据集的类别数量至少是三个。其中，我们选取了典型分类困难的 Cancers 数据集，此数据集拥有上万个特征维度，但只有 86 个训练样本。过多的特征为特征选择算法和分类算法都带来了巨大的压力。在另一方面，为避免算法倾向于分类困难的数据集，我们还选取了相对较为简单的 Breast、DLBCL 等数据集，测试算法的普遍适用性。

5.2 实验评价指标

基因微阵列数据存在大量噪声数据和易混淆数据，单维度的准确度无法准确衡量算法分类效果，采用多种方式从多个角度观测分类结果才能达到良好的效果。精确率（precision）、特效性（specificity）、召回率（Recall）和 Fscore 是机器学习、自然语言处理、模式识别和信息检索等领域中常用集中检测评估指标。其中精确率又称“精度”、“正确率”，召回率又称“查全率”，Fscore 是精确率和召回率的总和指标。一般来说，精确率和召回率反映样本遗漏和样本等多个方面，全面检测算法有效性和健壮性。评估指标的具体计算公式如下：

$$\text{Accuracy} = \text{avg}\left(\sum_{i=1}^{n}\frac{TP_i+TN_i}{P+N}\right) \quad (5\text{-}1)$$

$$\text{precision} = \text{avg}\left(\sum_{i=1}^{n}\frac{TP_i}{TP_i+FP_i}\right) \quad (5\text{-}2)$$

$$\text{recall} = \text{avg}\left(\sum_{i=1}^{n}\frac{TP_i}{P_i}\right) \quad (5\text{-}3)$$

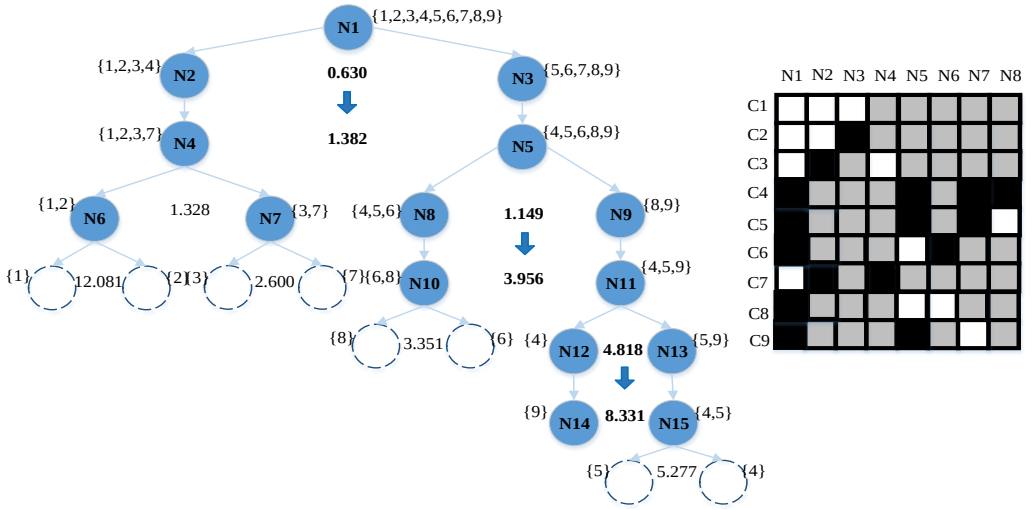
$$\text{specifity} = \text{avg}\left(\sum_{i=1}^{n}\frac{TN_i}{N_i}\right) \quad (5\text{-}4)$$

$$\text{Fscore} = \text{avg}\left(\sum_{i=1}^{n}\frac{(\beta^2+1)*precision_i*recall_i}{\beta^2+precision_i+recall_i}\right) \quad (5\text{-}5)$$

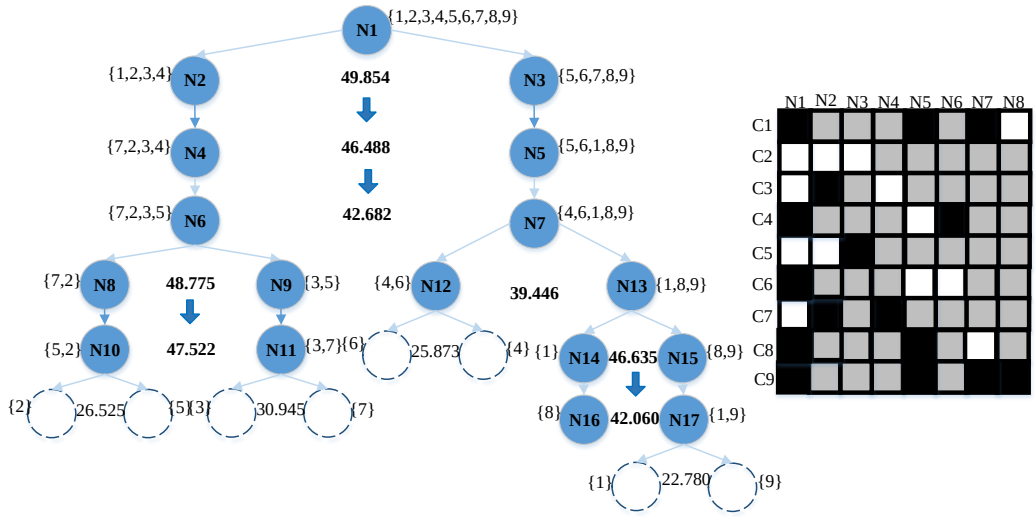
其中：n 表示多分类问题中的类别数目，实验中 β 值为 1。

5.3 编码矩阵结果

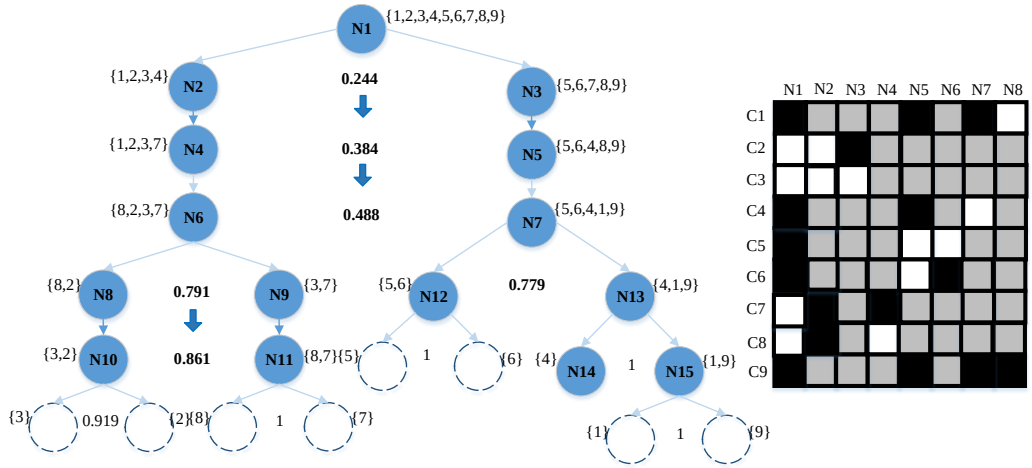
本节主要说明基于数据复杂度的 ECOC 算法的编码的具体过程，我们选用 Cancers 作为数据集，首先利用 ROC 算法过滤原始数据，随机选取前 100 个特征值作为训练和测试数据。数据集包含 9 个类别分别用 1-9 个数字随机表示。



(a). Class decomposition based on F1 measure



(b). Class decomposition based on F2 measure

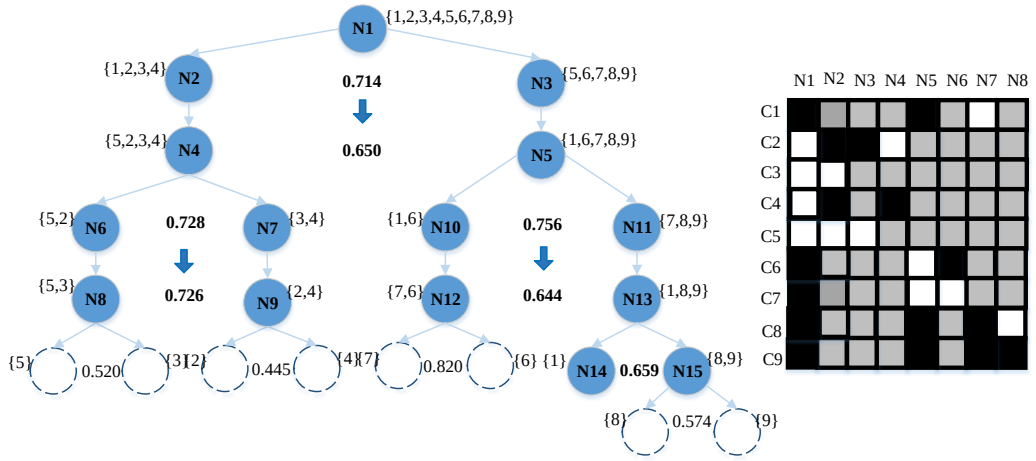


(c). Class decomposition based on F3 measure

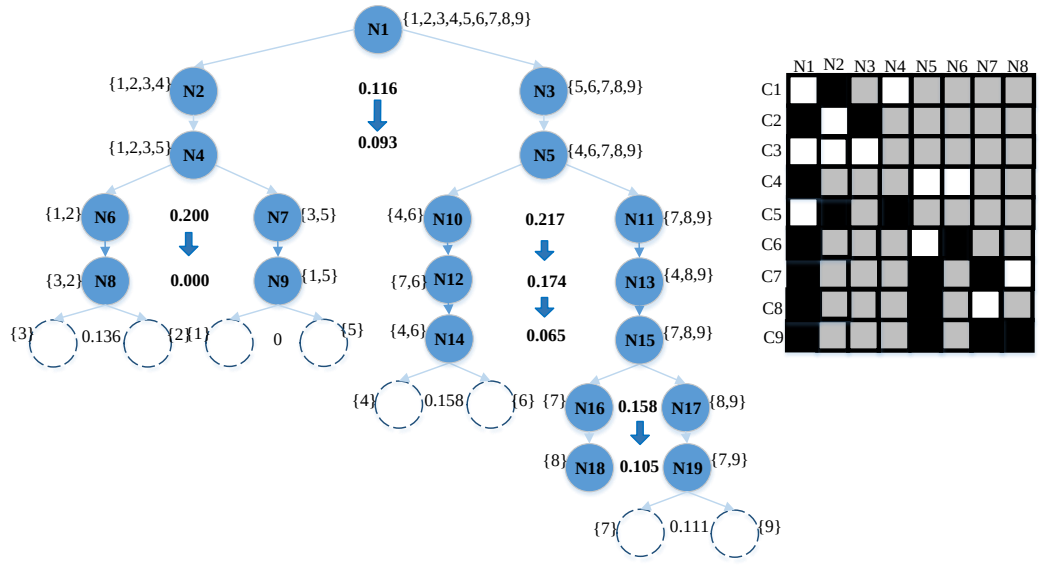
图 5-1 F1，F2 和 F3 的矩阵编码过程和编码矩阵图

从图 5-1 中可以看出，在每个节点的局部优化过程中存在多次局部类交换行为。其中，F1 测度出现两次交换，而 F2 和 F3 测度则经过三次交换后达到最优状态。从图中可以看出，经过交换后的复杂度数值明显增大，说明数据分布更稳定，类间数据的重合区域减小，类内数据分布更紧密。同时，对比三种复杂度产生的编码矩阵，可以发现即使编码矩阵的列数相同，但是矩阵整体差异较大，类别划分结果迥然不同。

上述的几幅图表明不同数据复杂度产生不同的编码矩阵，并且几种编码矩阵差异非常大，所以我们将这几种矩阵融合一种新的编码矩阵-基于特征的编码矩阵。因为融合过程中去掉了几种编码矩阵中相同的部分，所以基于特征的编码矩阵存在较好的行列分离性。



(a). Class decomposition based on N2 measure



(b). Class decomposition based on N3 measure

图 5-2 N2 和 N3 的矩阵编码过程和编码矩阵图

从图 5-2 中可以看出，基于 N2 和 N3 的数据复杂度变化和基于 F1，F2 和 F3 的复杂度变化过程类似。每个树节点的局部优化过程中存在多次局部类交换行为，N2 测度出现两次交换，而 N3 测度则经过三次交换后达到稳定状态。N3 的变化次数多于 N2 的变化次数，说明从 N3 的角度衡量数据，数据更复杂，数据可降低空间更大。从图中可以看出，经过交换后

的复杂度数值明显降低，说明数据分布更稳定，耦合区域减小，分布更紧密。N 系列复杂度生成的编码矩阵排列差异较大。编码矩阵的差异度越大，训练分类器效果越好。

5.4 数据复杂度变化结果

构建二叉树的过程中，每个节点都存在多次贪心类别变换过程，每次类别变换都会引起复杂度变换过程。为了进一步说明局部贪心算法的有效性，下图绘制了 6 种实验数据在某个节点的局部数据复杂度变化过程图。

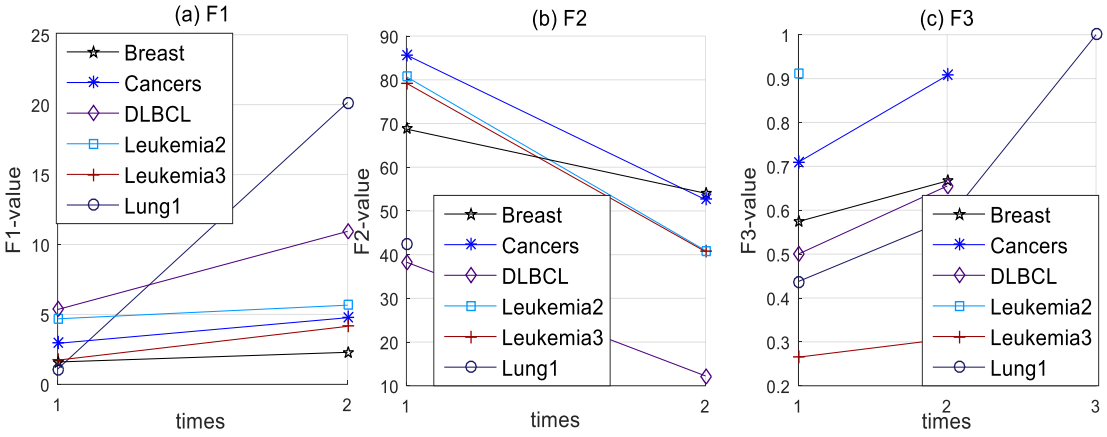


图 5-3 F1，F2 和 F3 复杂度测度在所有数据集上的变化过程图

图 5-3 主要描述了 F 系列（F1、F2 和 F3）数据复杂度在六种不同的数据集中的局部变化过程。从图 5-2 的(c)图可以看出，Lung1 数据刚开始的复杂程度测度值为 0.45 左右，经过两次不断调整最后复杂度测度值为 1。F3 的值变化范围是[0,1]，此时 Lung1 的结果已经达到最优，说明此时数据集的两个类别之间界限明显，线性分离程度高。从三幅图中可以看出，对于复杂程度一般的基因微阵列数据，三种复杂度测度都完成两次类别交换过程，而对于一些典型分类困难数据集如 Cancer 和 Lung1 等，类别交换行为进行了三次，且每次复杂度变化明显，每次类别交换后数据的复杂程度明显降低。对于 Lung1 等单一数据集，F3 测度的变换次数明显高于 F1 和 F2 测度变换过程的次数，说明 F3 测度对复杂数据更敏感，更能充分捕捉数据的潜在信息，并针对数据产生多次变化，通过调换节点中不同的类别不断调整复杂度值进而逐渐降低数据的复杂程度。

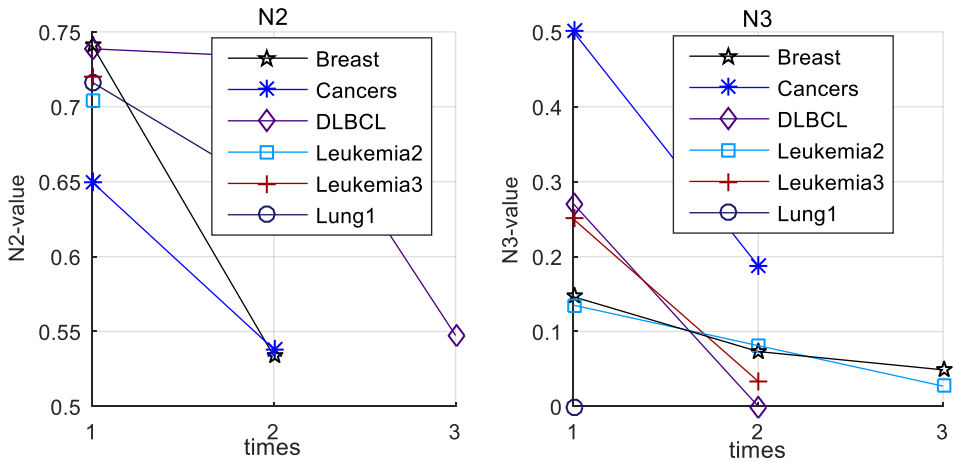


图 5-4 N2，N3 在所有数据集的变化过程图

图 5-4 图表示 N2 和 N3 复杂度测度的局部变化过程。从图中可以看出 N2 和 N3 复杂度变化较小，甚至有些数据集的复杂度测度没有发生变化。

对于这系列的测度来说，对于一般的基因微阵列数据，两种复杂度测度也实现了两次类别交换，而对于Lung1、Leukemia2和Breast等基因微阵列数据集类别交换次数增多到三次。通过对两种不同系列的复杂度变化进行比较，两种不同原理的复杂度测度都对Lung1数据集表现出多次变化，说明Lung1数据集属于典型分类困难数据集。对于分类困难点的数据集，基于类的特征重叠测度和类的分离测度的局部贪心算法能够有效应对。对于Lung1数据集，N3测度的变换次数较少，说明N3测度对复杂数据不敏感。

5.5 特征数量唯一的分类预测结果

表5-2到表5-5表示特征数量唯一时，基于特征重叠测度的改进算法和其他算法的实验分类结果。本次实验中，我们采用ROC特征选择算法筛选数据，特征维度设为100。利用支持向量机和贝叶斯作为基础二分类器。

从表5-2中可以看出，我们的算法在所有的测试数据中都取得了令人满意的效果，其他的ECOC分类算法的实验结果超过远远不如我们算法实验结果。我们算法中，基于特征重叠的ECOC算法的实验结果非常相似。三种测度都可以实现某种数据集的最好结果，识别准确率在Leukaemia2数据集上达到100%。相比其他算法的95%的正确率，我们算法的准确率提高了5%。同时，多数数据集的最优结果由基于F3测度的ECOC改进算法完成的，这说明F3测度的分类性能强于其他两种测度。在表5-2中，CF-ECOC几乎实现所有最优分类结果。

另外，对比表5-2和表5-3发现，基于三种测度的CF编码矩阵能够很好地分类全部数据集。分类结果的准确率达到93%。良好的分类结果表明三种测度产生的编码矩阵差异几乎涵盖数据所有表征，因而编码矩阵聚合获得更健壮的实验结果。

从表5-4中我们可以发现，我们算法的Fscore得分至少为0.5，Leukemia2和Breast数据的Fscore分数至少为0.9，这对于基因微阵列数据来说是非常不错的成绩。这证明我们的算法不仅可以实现正确分类还可以实现分类全面，算法具有良好的稳定性。

表5-2 基于ROC特征选择方法和SVM分类器的准确率结果

Dataset	F1	F2	F3	CF	Ordinal	DECOC	ECOCONE	Forest
Breast	0.960	**0.973**	0.960	0.933	0.947	0.933	0.933	0.960
Cancers	0.964	**0.970**	**0.970**	0.967	0.952	0.964	0.964	0.967
DLBCL	**0.989**	**0.978**	**0.989**	**0.989**	0.967	**0.989**	0.967	0.967
Leukaemia2	**1.000**	1.000	**1.000**	**1.000**	1.000	**0.956**	1.000	0.956
Leukaemia3	**0.878**	**0.849**	**0.916**	**0.895**	**0.842**	0.865	**0.885**	0.878
Lung1	0.875	0.875	**0.875**	0.875	0.875	0.896	0.875	0.875
Average	0.944	0.941	0.952	0.943	0.930	**0.934**	0.937	0.934

表5-3 基于ROC特征选择方法和NativeBayes分类器的准确率结果

Dataset	F1	F2	F3	CF	Ordinal	DECOC	ECOCONE	Forest
Breast	0.907	0.893	0.907	**0.933**	**0.933**	0.907	0.907	**0.933**
Cancers	0.940	0.967	0.949	**0.976**	0.940	0.952	0.958	0.949
DLBCL	0.911	0.922	0.911	0.922	0.956	0.922	**0.967**	0.933
Leukaemia2	**1.000**	**1.000**	**1.000**	**1.000**	0.956	0.956	0.956	0.956
Leukaemia3	0.875	0.870	0.883	0.857	0.816	0.862	0.865	**0.885**
Lung1	**0.917**	0.854	**0.917**	**0.917**	0.854	0.854	0.875	0.854
Average	0.925	0.918	0.928	**0.934**	0.909	0.909	0.921	0.918

表5-4 基于ROC特征选择方法和SVM分类器的Fscore结果

Dataset	F1	F2	F3	CF	Ordinal	DECOC	ECOCONE	Forest
Breast	0.829	**0.918**	0.829	0.789	0.727	0.754	0.683	0.845
Cancers	0.569	0.589	**0.595**	0.580	0.542	0.582	0.570	0.575
DLBCL	**0.976**	0.932	**0.976**	0.971	0.908	0.961	0.903	0.903
Leukaemia2	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	0.905	**1.000**	0.905
Leukaemia3	0.445	0.373	**0.672**	0.513	0.408	0.403	0.538	0.471
Lung1	0.790	0.790	0.790	0.790	0.790	**0.835**	0.790	0.790
Average	0.768	0.767	**0.810**	0.774	0.729	0.740	0.747	0.748

表5-5 基于ROC特征选择方法和NativeBayes分类器的Fscore结果

Dataset	F1	F2	F3	CF	Ordinal	DECOC	ECOCONE	Forest
Breast	0.583	0.555	0.583	0.696	**0.774**	0.646	0.587	0.767
Cancers	0.510	**0.635**	0.535	0.603	0.493	0.527	0.546	0.541
DLBCL	0.451	0.552	0.451	0.517	**0.772**	0.646	0.772	0.652
Leukaemia2	**1.000**	**1.000**	**1.000**	**1.000**	0.914	0.896	0.896	0.896
Leukaemia3	0.516	0.442	**0.537**	0.494	0.332	0.430	0.453	0.533
Lung1	**0.860**	0.673	**0.860**	**0.860**	0.738	0.673	0.790	0.673
Average	0.653	0.643	0.661	**0.695**	0.670	0.636	0.674	0.677

表 5-7 到表 5—8 表示特征数量唯一时，基于 N 系列的 ECOC 改进算法和其他算法的实验分类结果，本次实验结果采用的特征选择算法是 Wilcoxon 方法，特征维度为 80。

从表中可以看出，我们的算法在所有的测试数据中都取得了令人满意的效果，并且实验结果超过其他的 ECOC 分类算法。基于 N2 和 N3 的算法实验结果差别很大，明显看出基于 N2 复杂度测度的算法优势明显，几乎在所有数据集上都取得了良好的结果。

从表 5-6 中可以看出，当使用 NativeBayes 作为基分类器时，基于 N2 测度的算法在三种数据集上胜出，而基于 N3 的算法结果相对差一些，只在两种数据集上胜出。同时，基于 N2 的算法取得平均结果的最优结果，说明基于 N2 的算法整体效果较好。从 Fscores 得分上来看，我们算法整体 Fscores 得分数较高，对于 Leukemia3 和 Lung1 两种数据集来说，尽管其他算法的 Fscores 分数较高，但是分数依然维持在 0.4，我们的算法的得分为 0.35，结果相差 0.5。

从表 5-7 中可以看出，当使用 SVM 作为基分类器时，基于 N2 复杂度测度的 ECOC 改进算法优势明显，在所有 ECOC 算法中表现出色。无论是从单个数据集的表现还是整体数据集的均值结果表现来看，基于 N2 的 ECOC 算法都能实现良好的分类结果，可以有效解决多类分类问题。

表 5-6 基于 Wilcoxon 特征选择方法和 NativeBayes 分类器的 Accuracy 和 Fscore 结果

	datasets	N2	N3	Ordinal	DECOC	ECOCONE	Forest
	Breast	**0.99**	0.89	0.92	0.95	0.93	0.88
	Cancers	**0.99**	0.95	0.96	0.94	0.97	0.93
	DLBCL	0.98	0.92	0.97	0.97	0.97	0.92
Accuracies	Leukemia2	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	Leukemia3	0.84	0.78	0.84	**0.85**	0.79	0.82
	Lung1	0.90	**0.94**	0.90	**0.94**	0.92	0.90
	Average	**0.95**	0.91	0.93	0.94	0.93	0.91

续表 5-6

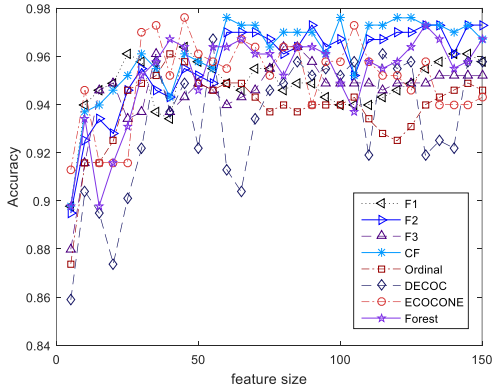
	datasets	N2	N3	Ordinal	DECOC	ECOCONE	Forest
Fscores	Breast	**0.97**	0.63	0.67	0.83	0.72	0.43
	Cancers	**0.74**	0.53	0.55	0.50	0.69	0.48
	DLBCL	0.90	0.58	0.79	0.86	0.79	0.55
	Leukemia2	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	Leukemia3	0.35	0.25	0.34	**0.41**	0.22	0.32
	Lung1	0.80	0.88	0.80	**0.91**	0.85	0.80
	Average	**0.79**	0.64	0.69	0.75	0.71	0.60

表 5-7 基于 Wilcoxon 特征选择方法和 SVM 分类器的 Accuracy 和 Fscore 结果

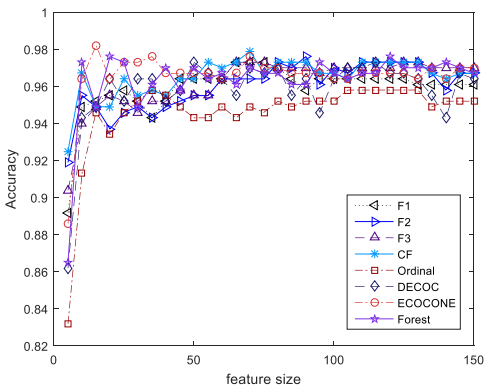
	datasets	N2	N3	Ordinal	DECOC	ECOCONE	Forest
Accuracies	Breast	**0.987**	0.907	0.920	0.933	0.907	0.920
	Cancers	**0.991**	0.949	0.967	0.949	0.985	0.961
	DLBCL	0.944	0.922	**0.967**	0.956	**0.967**	0.922
	Leukemia2	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
	Leukemia3	**0.885**	0.824	0.834	0.867	0.819	0.857
	Lung1	**0.917**	0.896	0.896	0.896	0.896	**0.917**
	Average	**0.954**	0.916	0.931	0.934	0.929	0.929
Fscores	Breast	**0.970**	0.685	0.674	0.827	0.704	0.820
	Cancers	0.657	0.528	0.573	0.513	**0.700**	0.572
	DLBCL	0.628	0.552	**0.794**	0.772	**0.794**	0.552
	Leukemia2	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
	Leukemia3	**0.500**	0.350	0.327	0.476	0.366	0.413
	Lung1	**0.860**	0.835	0.835	0.835	0.835	**0.860**
	Average	**0.769**	0.658	0.701	0.737	0.733	0.703

5.6 特征数量变化的分类预测结果

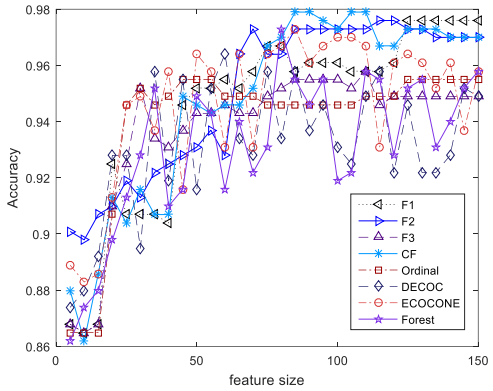
为避免单一特征数据数量造成实验结果偶然性误差大，我们测试了正确率在特征数量从 5 变化到 150 时结果的变换情况。



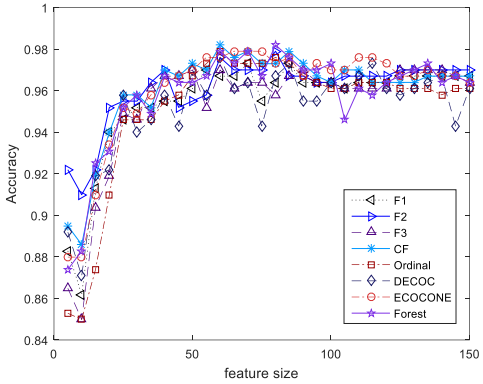
(a) ROC-NativeBayes



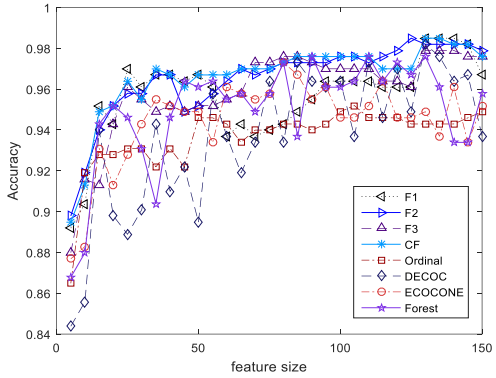
(b) ROC-SVM



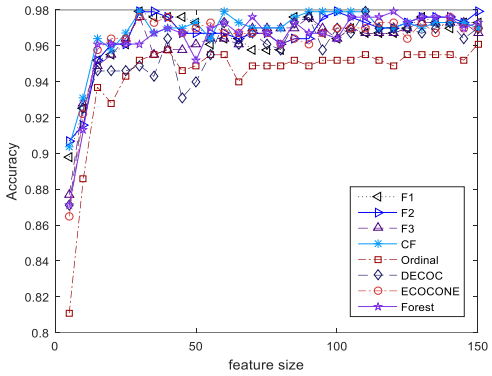
(c) T-test-NativeBayes



(d) T-test-SVM



(e) Wilcoxon- NativeBayes



(f) Wilcoxon-SVM

图5-5基于F系列的算法正确和特征数量关系图

图 5-5 表示基于类的特征重叠的算法正确率结果随特征数量变化关系图。从上图看出，无论是采用哪种特征选择算法，特征数量 30 是一个分界点，30 之前所有算法的实验并不理想，30 之后算法结果准确率明显提高，并各类 ECOC 算法开始逐渐稳定在某个区间范围内。观察图形左面的三幅图发现，我们的算法在所有 ECOC 算法中准确率一直保持领先状态，且每个准确率随特征数量变化不大，三种测度实验结果的浮动范围为[0,0.03]，相反的，DECOC，ECOCONE 和 ECOC-Forest 变化较大，说明我们的算法结果受特征数量影响较小，在不同特征数量的情况下，我们的算法依然保持较高的准确率。从图中右面的三幅图中可以看出，使用 SVM 作为基分类器的所有 ECOC 算法结果整体较为稳定，各种 ECOC 算法的实验结果都趋近于 0.98，对于基因微阵列数据来说，分类结果令人十分满意。同时也说明二分类器性能的好坏直接影响实验分类结果，性能优越的分类器可以实现更高的预测准确率。

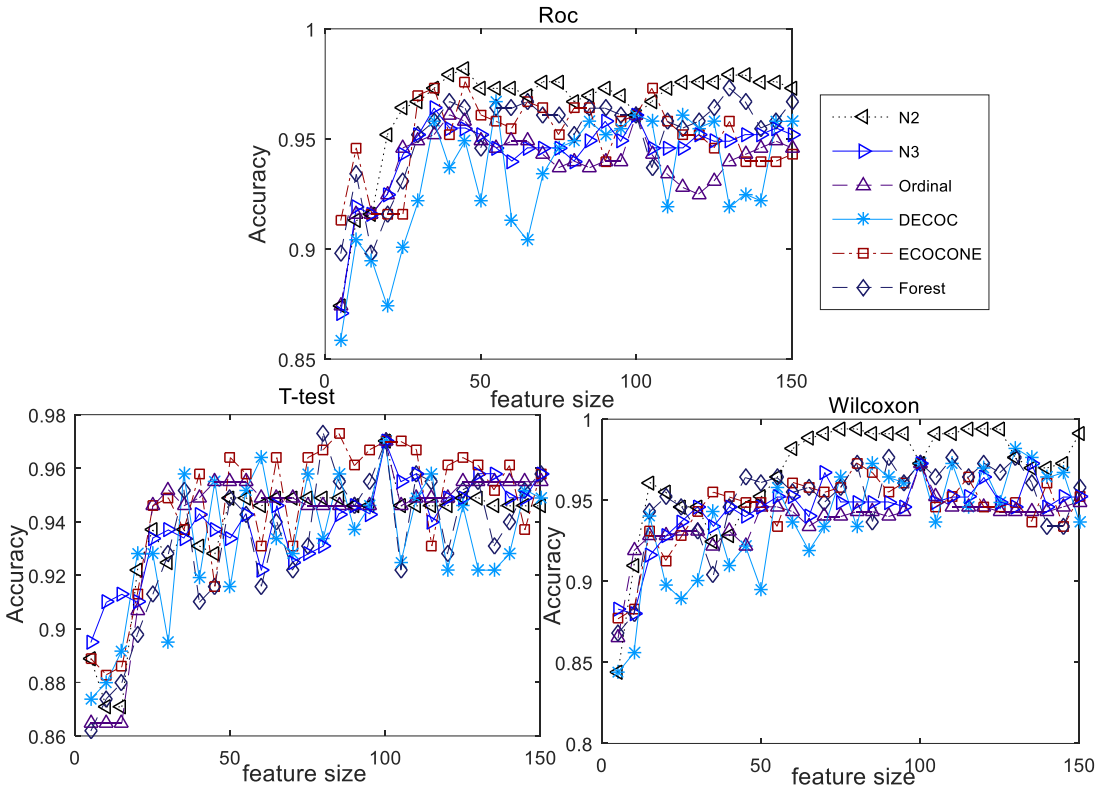


图 5-6 基于 N 系列的算法正确和特征数量关系图

图 5-6 表示基于类的分离测度算法的正确率结果随特征数量变化关系图。从上图看出，和基于类的特征重叠复杂度算法结果相似， 30 个特征数量依然是分界点，30 之前所有算法的实验并不理想，30 之后所有算法结果准确率明显提高，并各类 ECOC 算法开始逐渐稳定在[0.96,1]之间。上图中，DECOC 算法结果变化较大，其性能受特征数量影响较大，在不同特征数量的情况下，我们的算法依然保持较高的准确率。同时我们的算法在所有 ECOC 算法中准确率一直保持领先状态。使用 Roc 和 Wilcoxon 方法选择特征时，我们的算法都取得了良好的结果，几种基于数据复杂度的 ECOC 算法的实验结果都趋近于 1。

5.7 编码矩阵长度比较结果

ECOC 作为一种多分类算法框架，编码矩阵的规模影响二分类器的使用数量和算法运行时间，所以我们比较了基于数据复杂度和其他 ECOC 算法生成的编码长度，具体长度如下表所示：

表5-8 ECOC编码矩阵长度表

Dataset	F1	F2	F3	N2	N3	CF	Ordinal	DECOC	ECOCONE	Forest
Breast	**4**	**4**	**4**	**4**	**4**	12	**4**	**4**	7	12
Cancers	**8**	**8**	**8**	**8**	**8**	20	**8**	**8**	11	24
DLBCL	**5**	**5**	**5**	**5**	**5**	13	**5**	**5**	7	15
Leukaemia2	**2**	**2**	**2**	**2**	**2**	3	**2**	**2**	4	6
Leukaemia3	**6**	**6**	**6**	**6**	**6**	17	**6**	**6**	8	18
Lung	**2**	**2**	**2**	**2**	**2**	4	**2**	**2**	8	6

从表 5-8 中可以发现，我们的五种基于复杂度测度的 ECOC 算法和 Ordinal 以及 DECOC 产生 n-1 列编码矩阵，而由 CF、ECOCONE 和 Forest-ECOC 等算法生成的编码矩阵则规模较大。

长度越短的纠错编码矩阵将使用越少的二分类器，分类器数目越少算法训练时间和预测时间越短。综上所述，基于五种复杂度测度的算法时间、分类器数目消耗最小但是实验效果最好，我们的算法从时间、分类器数目和实验结果等各方面来说都实现了不错的分类效果。

5.8 本章小结

本章首先介绍了实验中使用的六种基因微阵列数据，然后介绍了评价算法结果的五种指标。之后介绍并分析五种基于数据复杂度的纠错输出编码矩阵算法和 Ordinal、DECOC、ECOCONE 和 Forest-ECOC 的正确率和 Fscore 结果、复杂度变化情况，还比较了不同的 ECOC 生成的编码矩阵的大小。

实现结果表明，在不同的结果中，本文提出的算法都占据优势地位。为了进一步说明算法具有良好的健壮性，可以应对不同特征数量对算法的影响，根据特征维度数量确定和特征维度数量变化两种情况下的算法表现，从结果来看，本文提出的算法不依赖特征数量，在任意特征数量下，该算法都能取得较好的结果并且算法时间消耗最少。

第六章 结论

6.1 本文主要工作

本文的主要工作是：

（1）分析不同的特征选择算法的原理，针对实验中的基因阵列数据集，选用三种不同的特征选择方法降低基因微阵列数据维度。

（2）提出一种基于数据复杂度的纠错输出编码算法（ECOC-DC）。我们运用 F1、F2、F3、N2 和 N3 等复杂度测度生成不同的编码矩阵，并将这种编码矩阵运用到 ECOC 算法中。这种方法是属于数据驱动的一种 ECOC 算法，相比于传统的一对一，一对多、DEOC 和 Forest-ECOC 等改进 ECOC 算法，我们算法的稳定性和容错能力和学习能力更强。微阵列数据的分类准确率在我们的算法上得到非常大的提高。我们算法在所有数据集上的平均准确率达到 93%，Fscore 分数达到 70%，部分数据集准确率达到 100%。另外，对于典型分类困难的 Cancers 数据集，ECOC-DC 的准确率达到 95%。相比其他算法的结果，准确率提高了 2%。

（3）提出四种局部贪心调整算法。针对不同复杂度测度制定相应的局部贪心策略进一步调整二分类结果，使每一步的复杂度分配达到最优，这种二分类的分配方式充分利用数据本身的特点，有效利用基因微阵列数据高维度的数据特性，尽可能开发潜在最优特征的价值。

（4）提出一种基于特征的融合编码矩阵。本文将基于 F1，F2 和 F3 复杂度测度生成编码矩阵融合，生成基于特征的融合编码矩阵-CF 矩阵，并在实验过程测试矩阵效果，实验结果证明 CF 编码矩阵行列具有非常大的分离性，基于 CF 编码的 ECOC 算法具有良好的分类效果。对比基于 F1，F2 和 F3 的算法结果，新的 ECOC 算法将结果的准确率提高 1%，Fscore 提高 2%。

（5）将创新算法应用到基因微阵列数据上。本文通过数据复杂度的算法解决基因微阵列数据上多分类问题。为了验证算法的有效性，我们将传统的 ECOC、DECOC、ECOCONE 和 ECOC-Forest 作为比较算法。实验结果表明 ECOC-DC 实现更好的分类结果和更健壮的稳定性。基于单一特征数量和基于变化特征数量两个方面的稳定结果进一步证明本文提出的算法受特征数量影响较小，该算法能在不同的数据上都取得良好的结果。

6.2 研究展望

基因微阵列数据在生物学领域的研究有着广泛的应用，由于数据具有“高维高噪小样本”的特点，本文提出了一种基于数据复杂度的多分类学习模型。该学习模型可以从以下几个方面进行延伸：

(1)本文主要解决基因微阵列数据多分类问题，本文在生成编码算法的过程中选用五种基于数据复杂度测度生成编码矩阵，不断调整类的分配以降低类别简的复杂度，但是这五种复杂度主要针对数据特征和类别分界线，会出现过分依赖特征的问题，不能全面准确的判断类的复杂程度，所以在今后的研究过程中将选用更多不同的复杂度测度从多个角度衡量数据内部分布情况，从而进一步提高算法结果。

(2)在实验中我们选用贝叶斯和 SVM 两种分类器对编码矩阵进行训练，但是单一分类器的实验结果对于不同的分类产生不同的影响，对于分类器，可以进一步调整各个分类器的权重，使得一些更加重要的分类器具有更高的权重。从而进一步改进分类器的准确率。

(3)神经网络分类器是近年来一个值得关注的研究领域，神经网络方法能够对于多个维度进行深度挖掘，找出各个维度之间的相互关系。虽然神经网络方法的隐含层的训练过程是黑箱的，缺乏理论方面的指导，但其对于数据分类的高准确率却吸引了众多研究人员在这个领域的探索。所以我们的算法中运用神经网络分类器也同样是一个值得关注的研究方向。

参考文献

[1] 陈东，癌症基因微阵列分类方法的研究[d]. 2012，湖南大学.

[2] Ritchie M E, Phipson B, Wu D, et al. limma powers differential expression analyses for RNA-sequencing and microarray studies[J]. Nucleic Acids Research, 2015, 43(7):e47.

[3] Yasrebi H. Comparative study of joint analysis of microarray gene expression data in survival prediction and risk assessment of breast cancer patients[J]. Briefings in Bioinformatics, 2016, 17(5):771.

[4] N-Canedo V, nchez-Maro, O, N, et al. A review of microarray datasets and applied feature selection methods[J]. Information Sciences An International Journal, 2014, 282(5):111-135.

[5] Allwein E L, Schapire R E, Singer Y. Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers[C]// Seventeenth International Conference on Machine Learning. Morgan Kaufmann Publishers Inc. 2000:9-16.

[6] Crammer K, Singer Y. On the Learnability and Design of Output Codes for Multiclass Problems[C]// Thirteenth Conference on Computational Learning Theory. Morgan Kaufmann Publishers Inc. 2000:35-46.

[7] Pujol O, Radeva P, Vitria J. Discriminant ECOC: A Heuristic Method for Application Dependent Design of Error Correcting Output Codes[M]. IEEE Computer Society, 2006.

[8] Liu K H, Zeng Z H, Ng V T Y. A Hierarchical Ensemble of ECOC for cancer classification based on multi-class microarray data[J]. Information Sciences, 2016, 349-350:102-118.

[9] Tapia E, Serra E, González J C. Recursive ECOC for Microarray Data Classification[C]// International Conference on Multiple Classifier Systems. Springer-Verlag, 2005:108-117.

[10] Ho T K, Basu M. Complexity Measures of Supervised Classification Problems[J]. Pattern Analysis & Machine Intelligence IEEE Transactions on, 2002, 24(3):289-300.

[11] Singh S. Multi-Resolution Estimates of Classification Complexity[J]. Pattern Analysis & Machine Intelligence IEEE Transactions on, 2003, 25(12):1534-1539.

[12] Li Y, Dong M, Kothari R. Classifiability-based omnivariate decision trees[J]. IEEE Transactions on Neural Networks, 2005, 16(6):1547-60.

[13] Tan J, Chua K S, Zhang L. Algorithmic and Complexity Issues of Three Clustering Methods in Microarray Data Analysis[M]// Computing and Combinatorics. Springer Berlin Heidelberg, 2005:74-83.

[14] Bolan-Canedo, V., L. Moran-Fernandez, and A. Alonso-Betanzos. *An insight on complexity measures and classification in microarray data*. in *International Joint Conference on Neural Networks*. 2015.

[15] Bolan-Canedo V, Moran-Fernandez L, Alonso-Betanzos A. An insight on complexity measures and classification in microarray data[C]// International Joint Conference on Neural Networks. IEEE, 2015:1-8.

[16] Escalera S, Pujol O. ECOC-ONE: A Novel Coding and Decoding Strategy[C]// International Conference on Pattern Recognition. IEEE Computer Society, 2006:578-581.

[17] Fawcett T. An introduction to ROC analysis[J]. Pattern Recognition Letters, 2006, 27(8):861-874.

[18] Wang D, Zhang H, Liu R, et al. t-Test feature selection approach based on term frequency for text categorization ☆[J]. Pattern Recognition Letters, 2014, 45(1):1-10.

[19] 曾艳, 李桂花, 庄刘. 完全随机设计两样本的 Wilcoxon 检验与 K-S 检验功效比较[J]. 中国卫生统计, 2011, 28(4):372-374.

[20] Escalera S, Pujol O, Radeva P. Boosted Landmarks of Contextual Descriptors and Forest-ECOC: A novel framework to detect and classify objects in cluttered scenes[J]. Pattern Recognition Letters, 2007, 28(13):1759-1768.

[21] Kuncheva L I. Using diversity measures for generating error-correcting output codes in classifier ensembles[J]. Pattern Recognition Letters, 2005, 26(1):83-90.

[22] Crammer K, Singer Y. On the Learnability and Design of Output Codes for Multiclass Problems[J]. Machine Learning, 2002, 47(2):201-233.

[23] Cortes C, Vapnik V. Support-Vector Networks[M]. Kluwer Academic Publishers, 1995.

谢辞

毕业设计从 2017 年 3 月至今已有近 3 个月的时间了，在这段时间内，我努力学习算法研究基本知识，在本校杨春明老师和厦门大学的刘老师的帮助下完成了本人的毕业设计，在算法实现和论文书写的过程中我逐渐掌握了论文写作方法、论文书写格式要求以及基本学术研究方法。本次论文从选题到完成，每一步我都遇到了许许多多的困难，但是每一步的过程中导师都给予我细心的帮助和指导。所有的学术成果都凝聚了导师大量的心血。在此，谨向导师表示真挚的敬意和衷心的感谢！

韶华飞逝，转瞬既是毕业时节，春梦秋云，聚散不易，且行且珍惜。回想大学四年，期间无数的老师都曾给予我很多无私的指导和帮助，让我在学业上取得了一定的成绩，也在生活上变得更加独立自信。在此我所有教过我的老师们表示衷心的谢谢，非常感谢你们四年的辛劳，感谢四年里面你们辛勤不断的教诲，感谢你们的付出！

四年的学习过程，我不仅仅收获了大量的专业知识，思维方式、学习能力和阅历取得一定的进步。非常庆幸这四年来我遇到了志同道合的朋友，一直在工作上、学习上、生活上都给予了我无微不至的帮助和照顾，让我能幸福快乐地度过四年的大学生活。非常感谢身边同学四年来对我的爱护、包容和帮助，谢谢你们。

最后要感谢的是我的父母，感谢你们不仅充分信任我，让我自己选择自己喜欢的专业，更全力支持我的学业。未来我会更加努力地学习和工作，不辜负父母对我的谆谆教导和殷殷期望！

未来不管我在哪里，身边有谁，我都会记得曾有无数的老师和同学在我身边陪我走过生命中重要的四年，未来不论我走向何方，我会记得将友善和付出继续传递下去，希望有更多的人可以和我一样受益。