H056

基于集成学习的 LRG 分级预测模型

摘要

新辅助放化疗结合全直肠系膜切除术是治疗直肠癌的重要途径之一。为了判断病人在放化疗后的情况，定义了淋巴结消退分级（LRG）和肿瘤消退分级（TRG）对淋巴结转移情况和原发灶的肿瘤情况进行评估。本文分析了影响 LRG 分级的重要指标，探索在手术前进行 LRG 分级预测的方法，并判断了 LRG 与 TRG 之间的关系。

对于问题一的探索 LRG 分级影响因素的要求，本研究首先对数据进行处理，根据基本假设去除了存在其他疾病的患者的数据和放疗与手术时间差过大的患者的数据，然后利用中位数和众数补全了部分缺失的数据，再根据得到的结果优化或删除偏差过大的异常数据，最后再次对缺失数据进行补全。利用优化之后的数据表格，本研究新定义了一种基于假设的“淋巴结”对患者总体情况进行概括的 LRG，再利用 SPSS，结合逐步回归分析的方法，分别对术前和术后的各项指标与 LRG 分级之间的关系系数进行求解。最终发现，对于术前的各项指标，年龄是影响 LRG 分级的因素；对于术后的各项指标，显著影响 LRG 的因素主要有获得最差评分的淋巴结的 LRG 值，所有淋巴结 LRG 值的和，淋巴结转移比率，肿瘤沉淀和阳性淋巴结数。

对于问题二，本研究通过直肠癌患者在进行 nCRT 后手术前的各项指标数据与在问题一中得到的 LRG 类别，建立了一个 LRG 分级预测模型。该模型使用直肠癌患者进行 nCRT 后手术前的各项数据作为输入，采用了集成学习的方法，将支持向量机回归模型、线性规划模型、随机森林回归模型组合起来，可以对患者的 LRG 分级进行预测。与其他模型相比，该集成学习模型准确率更高（可以达到 0.75）、稳定性更强（本研究进行了十余次测试，均稳定在 0.75）、受数据部分问题影响（例如类别不均衡）更小。

对于问题三的探索 LRG 与 TRG 之间关系的要求，本文对各种求解相关系数的方法进行比较，认为肯德尔和谐系数是最适合的求解 LRG 与 TRG 之间关系的方法。通过 SPSS 的计算，最终得到的 LRG 与 TRG 之间的关系系数为 0.179，显著性为 0.047。根据显著性的定义，当两个变量之间的显著性在 0.05 级别时两个变量之间相关性显著。由于 LRG 和 TRG 之间的关系系数相对较小，可以认为两者之间虽然存在关系，但只是弱相关关系，因此不能通过其中一个变量来反映另一个变量。

最后，报告对模型的优缺点进行了评价。

关键词 淋巴结消退分级 肿瘤消退分级 逐步回归分析 集成学习 肯德尔和谐系数 LRG 分级预测模型

目录

一、问题再述

直肠癌的发病与多种因素有关，而淋巴道转移是直肠癌细胞转移的途径之一[1]。通过术前新辅助放化疗（nCRT）结合全直肠系膜切除术（TME）可以对直肠癌进行有效治疗[2]，治疗的结果可以通过淋巴结消退分级（LRG）和肿瘤消退分级（TRG）以及其他一些参数评估定级。其中，LRG 的分级如表 1.1 所示：

LRG0	正常淋巴结，无退化或癌细胞迹象
LRG1	100%纤维化，无残留肿瘤细胞
LRG2	75－100%纤维化，0-25%肿瘤细胞
LRG3	50-75%纤维化，25-50%肿瘤细胞
LRG4	25-50% 纤维化，50-75%肿瘤细胞
LRG5	0-25%纤维化，75-100%肿瘤细胞

表 1.1 LRG 的分级和症状[2]

TRG 的分级如下表 1.2 所示：

Mandard TRG system	
TRG1	No viable cancer cells, complete response
TRG2	Single cells or small groups of cancer cells
TRG3	Residual cancer outgrown by fibrosis
TRG4	Significant fibrosis outgrown by cancer
TRG5	No fibrosis with extensive residual cancer

表 1.2 TRG 的分级和症状（Mandard 标准）

对于直肠癌的治疗，可以根据 nCRT 后 LRG 和 TRG 的值进行分析，根据分析的结果判断是否进行后续的 TME 手术以及手术中是否摘除淋巴结。

为了更好的对 nCRT 治疗后的效果进行分析，有必要结合治疗后检测到的各项指标，利用 LRG 和 TRG 对治疗效果进行分级。然而，由于 LRG 相关模型不够成熟，需要建立更加精准的模型对各项指标和 LRG 以及 TRG 之间的关系进行分析。

本研究根据解决问题过程中的实际需要，对问题改写为以下的描述：

问题 1：通过对表中各项指标及影响 LRG 的因素进行分析，确定影响 LRG 评级的主要因素；

问题 2：通过处理后的数据，寻找术前指标与 LRG 之间的关系，选择合适的算法，建立根据以上因素得到的 LRG 分级模型。

问题 3：利用相关性分析，从而分析 LRG 与 TRG 之间的关系。

二、基本假设与符号说明

2.1 基本假设

（1）假设所有病人都只患有直肠癌一种疾病

为了方便叙述，并且减小其他因素对直肠癌 LRG 分级的影响，在本文中，将假定所有符合条件的病人，在没有更多说明的情况下，都只患有直肠癌一种疾病，且由直肠癌导致的淋巴结转移是导致淋巴结检测为阳性的唯一原因。

（2）假设存在一个“淋巴结”，这个“淋巴结”的 LRG 情况可以概括该病人所有的被检测过

的淋巴结的 LRG 情况

每个病人身体中被检测的淋巴结数目不同，检测出阳性的淋巴结数目也不同，阳性淋巴结的分级也不同。为了能概括这个病人的大致情况，本研究假设存在一个“淋巴结”且这个“淋巴结”的 LRG 情况可以概括该病人各个淋巴结的 LRG 情况。本文假设通过 pT 和 pN 以及 LNR 能够得到一个“淋巴结”的 LRG，这个 LRG 将作为最笼统的参数对这个病人的 LRG 情况进行概括。具体的判断方法见表 3.5。

2.2 符号和术语说明

变量和术语	说明
nCRT	新辅助放化疗
TME	全直肠系膜切除术
LRG	淋巴结消退分级
TRG	肿瘤消退分级
cT	术前根据影像及肠镜等资料判断的 T 分期
pT	术后病理根据肿瘤浸润深度评估的 T 分期
cN	术前根据影像资料判断的 N 分期
pN	术后病理根据转移淋巴结的数目评估的 N 分期
LNR	淋巴结转移比率，即转移淋巴结的个数除以清扫淋巴结的总数
LRGmax	获得最差评分的淋巴结的 LRG 值
LRGsum	所有淋巴结 LRG 值的和
ECOG	从患者的体力来了解其一般健康状况和对治疗耐受能力的指标

三、问题一的数据处理和解答

3.1 问题分析与思路

问题 1 需要根据数据表 data.xls 的数据找到影响 LRG 的因素。首先需要分析已有的数据，已有数据主要包含了某医院在 2019 年 10 月至 2022 年 9 月之间的直肠癌患者接受 nCRT 后手术的临床收据，也分别给出了患者的性别、年龄、放疗时间、治疗时间等个人情况，以及距肛距离、术前术后的 N 分期、T 分期、阳性淋巴结数量、术后 TRG 和 LRG 评估值等与肿瘤情况有关的数值。

其次要对数据进行预处理，包括脏数据的清理、通过 SPSS 进行数据的缺失值补全和异常值处理，增强数据的可信度，帮助后续建模过程提高稳定性。然后，根据 LRG 的相关定义和假设以及手术后各项指标求出新的 LRG 并将其分级作为数据的最后一列。最后通过 SPSS 对求出的 LRG 与其他变量间的相关性进行分析，通过分析结果了解影响 LRG 的重要因素，并对术前影响 LRG 的因素进行降维。图 3.1 展示了基本的数据处理思路。

图 3.1 数据处理思路

3.2 数据分析和预处理

3.2.1 脏数据清理

在数据表 data.xls 的数据中，一共有 101 组病人术前术后各项指标的数据。根据以下的判断标准，本研究去除了某些脏数据：

A.个别病人本身患有其他疾病或接受了其他类型的手术（如横结肠造瘘），根据单一变量原则，这些数据应排除而减小其他因素对模型的影响。

B.个别病人的放化疗与手术时间间隔过长，最长的一组间隔时间达到 40 周（平均 11.11 周）。考虑到放化疗时间与手术时间间隔对手术效果的影响，这些数据也应该排除在模型建立的范围之外。同时，因为放疗与手术的时间差已经得出，具体的放化疗和手术的时间也可以去除。

C.某些指标（如是否接受了 TME）全部为同样的数字，也可以去除。

D.某些指标（如 cN 和 cN+）得到的是对同一个结果的描述，因此可以去除其中一组或多组数据以减少数据容量。

E.删除备注。

进行了以上的数据处理之后，部分被清除的数据如下表 3.1 和 3.2 所示。

病案号	pT	pN	术后TRG	LRGmax	LNR分组	阳性淋巴结数	备注
387342	4	0	2	0	0	0	横结肠造瘘
394846	0	0	0	0	0	0	横结肠造瘘
405250	2	0	1	1	0	0	急诊横结肠造瘘

表 3.1 存在其他疾病的患者的数据

病案号	放疗与手术时间差（W）	pT	pN	术后TRG	LRGmax	LNR分组	阳性淋巴结数
360544	40.00	0	0	0	0	0	0

表 3.2 放疗与手术时间差过大的患者的数据

3.2.2 缺失值补全

由于部分数值的缺失（某些病例的肿瘤沉积、微血管浸润等指标缺失）会对本研究的建模产生一定的影响，因此需要对表中部分缺失数据进行补全。考虑到各个病例的个体性差异和共性，本研究决定采用平均数或者众数对数据进行补全。但由于计算出的平均数会得到小数，和现实情况不符，因此本研究最终选择利用众数对缺失数据进行补全。进行众数补全后的部分数据结果如下表 3.3 所示。

性别	年龄	mrTRG	与手术时	肿瘤分级	肿瘤沉积	微血管浸润	神经侵犯
1	73	2	11.71429	1	1	0	0
1	55	2	10	1	1	1	1
1	50	2	10.14286	1	1	0	0
1	39	2	11.71429	0	1	0	0
2	57	2	12.71429	1	0	0	0
2	51	2	10.57143	1	1	0	0
1	63	2	13	2	1	0	0
1	57	2	13	1	1	1	0
1	54	2	9.857143	1	1	0	0
1	53	2	13	1	1	0	0

表 3.3 部分数据补全后得到的各项指标

3.2.3 异常值处理

如果有某些值与正常的数据范围偏离较大的话，这些值会严重影响数据的判断和建模的

准确性。因此，在允许的范围内，本研究需要找出这些异常的数据并将他们剥离或填补上符合要求的数据。如在下图 3.2 和 3.3 中，本研究利用 3σ 检测和曲线波动检测对患者的周数差和距肛距离的数据进行图形绘制。

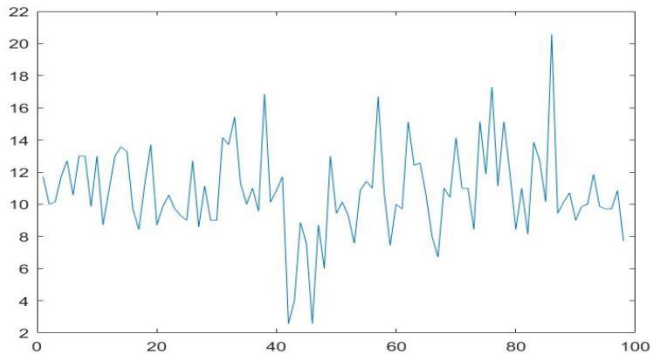


图 3.2 处理前的周数差的折线分布

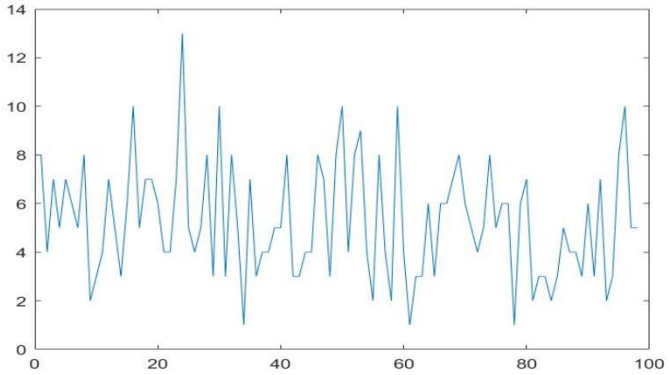


图 3.3 处理前的距肛距离的折线分布

从图中不难看出，某些数据明显偏离总体的情况，因此本研究需要判断是要将这些数据进行修正或是进行剥离。当然，在进行修正之前需要强调的是，某些客观数值无法或修正，只能通过数据剥离进行优化，这一过程或将减少可用数据的数量。考虑到数据表 data.xls 中的数据量十分有限，异常值的处理需要经过十分慎重仔细的考量。

3.2.4 去除异常值后补全缺失值

如上 3.2.3 中所叙述，将整条数据剥离或将影响可用数据的数量。在没有确定具体哪些因素会对 LRG 产生影响之前，盲目的去除部分因素缺少的病例数据是不可取的。因此，本研究参照 3.2.2 中提到的方法，利用中位数和众数，将缺失的部分数据进行补全。

通过以上的四个步骤，针对于问题一的数据处理基本完成。下图 3.4 展示了经过 3σ 检测和曲线波动检测和数据优化后的周数波动曲线。可以看到，根据 3σ 检测，一条偏差较大的数据被剥离，因此最大可能的保留了足够多的可以用于后续建模使用的数据。表 3.4 展示了被剥离数据的具体值。

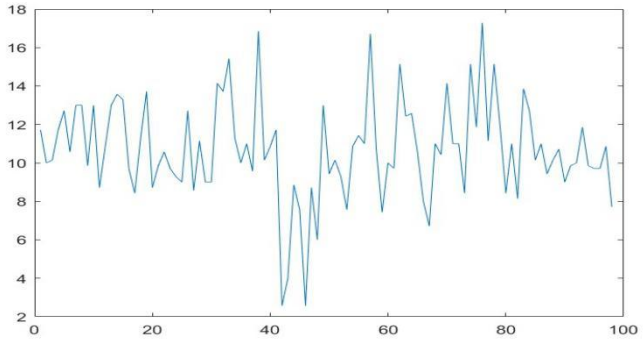


图 3.4 优化处理后的周数差波动曲线

性别	放疗与手术时间差	pT	pN	术后TRG	LRGmax
1	20.57	3	2	3	5

表 3.4 被剥离的异常数据

3.3 根据处理过的数据和 LRG 相关定义找出一个淋巴结，用它的 LRG 代表单个病人的情况

数据表 data.xls 中的表格里有两个数值，分别是 LRGmax 和 LRGsum。为了得到这两个值，需要对病人经过处理的淋巴结进行分级。每一个病人经过处理的淋巴结的数量不一，阳性的淋巴结数量也不同，每一个淋巴结的分级也不一致，因此，不论是 LRGmax 还是 LRGsum 的值都有很大的随机性。同时，由于数据的局限性，本研究并不知道每一个淋巴结具体的 LRG 分级。因此，需要一个更加笼统的比较参数对每一个病人的淋巴结 LRG 情况进行概括。

在这里，本研究假设通过 pT 和 pN 的值以及 LNR 能够得到一个“淋巴结”的 LRG，这个 LRG 将作为最笼统的参数对这个病人的 LRG 情况进行概括。具体的判断方式如下表 3.5：

<table>
<tr><th>pT</th><th>pN</th><th>LNR</th><th>LRG</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td rowspan="5">pT 和 pN 不同时为 0</td><td rowspan="5"></td><td>0</td><td>1</td></tr>
<tr><td>大于 0 小于 0.25</td><td>2</td></tr>
<tr><td>大于 0.25 小于 0.5</td><td>3</td></tr>
<tr><td>大于 0.5 小于 0.75</td><td>4</td></tr>
<tr><td>大于 0.75</td><td>5</td></tr>
<tr><td colspan="4">最后，在 LRGmax 和 LRG 中取最大值</td></tr>
</table>

表 3.5 “淋巴结”LRG 的判断方法

通过上表所示的判断方法，本研究可以得到一个全新的 LRG，部分病例 LRG 与 LRGmax 的对比如下表 3.6 所示。可以发现，新的 LRG 避免了某些特别大的 LRG 对整体判断的影响，将更多的因素纳入考虑。

性别	年龄	pT	pN	LRGmax	LNR	LRG类别
1	73	2	0	1	0	1
1	55	4	1	0	0	1
1	50	3	0	1	0	1
1	39	2	0	0	0	1
2	57	0	0	1	0	1

表 3.6 LRGmax 与 LRG 的比较

3.4 利用 SPSS 对不同指标与 LRG 的相关性进行分析

3.4.1 不同指标之间的标准化

由于不同的指标之间单位、量级的差距，需要提前对各种指标进行标准化，以让它们可以在某一个共同的维度下进行比较。这里运用 SPSS 自带的标准化功能对不同的指标进行标准化，具体的标准化过程不再赘述。下表 3.7 展示了部分指标在标准化之前和标准化之后的数值（Z 代表标准化）。

肿瘤分级	Z肿瘤分级	肿瘤沉积	Z肿瘤沉积	微血管浸润	Z微血管浸润	神经侵犯	Z神经侵犯
1	-0.15946	1	0.43947	0	-0.33538	0	-0.55094
1	-0.15946	1	0.43947	1	2.95131	1	1.79655
1	-0.15946	1	0.43947	0	-0.33538	0	-0.55094
0	-1.89581	1	0.43947	0	-0.33538	0	-0.55094
1	-0.15946	0	-2.25227	0	-0.33538	0	-0.55094
1	-0.15946	1	0.43947	0	-0.33538	0	-0.55094
2	1.57689	1	0.43947	0	-0.33538	0	-0.55094
1	-0.15946	1	0.43947	1	2.95131	0	-0.55094
1	-0.15946	1	0.43947	0	-0.33538	0	-0.55094
1	-0.15946	1	0.43947	0	-0.33538	0	-0.55094

表 3.7 部分指标在标准化前后的数据

3.4.2 相关性比较

在经过标准化之后，本研究利用逐步回归分析的方法，利用 SPSS 软件对各种指标与能代表单个病人的 LRG 分级之间的关系进行判断。

逐步回归分析是在有两个或两个以上自变量时使用的一种回归分析方式。在实际生活中，大部分的因变量并不是受一个自变量控制的，因此引入逐步回归分析方法具有非常重要的实际意义。在问题一中，需要进行判断的各种自变量超过十个，运用普通的一元线性回归很难无法将所有的可能的影响因素囊括，同时也会大大增加计算的难度和时间。借助 SPSS 软件和逐步回归分析的方法，本研究得到了与 LRG 相关性最高的系数。

A.术前数据对 LRG 的影响

通过 SPSS 的计算和判断，本研究发现只有年龄一项是术前数据中对 LRG 的结果存在显著影响的。如下表 3.8 所示，年龄与 LRG 之间的相关系数约为 0.243，说明二者间存在联系。

模型摘要 [b]

模型	R	R 方	调整后 R 方	标准估算的错误	德宾-沃森
1	.243[a]	.059	.049	.97498983	2.021

a. 预测变量：(常量)，Zscore(年龄)

b. 因变量：Zscore(LRG 类别)

表 3.8 年龄与 LRG 间的相关系数

B.术后数据对 LRG 的影响

模型	Beta	显著性
(常量)		1.000
Zscore(肿瘤分级)	.025	.256
Zscore(术后 TRG)	.045	.328
Zscore(LRGmax)	.722	.000
Zscore(pN)	-.021	.613
Zscore(LRGsum)	-.440	.000
Zscore(pT)	.047	.310

Zscore(CRMR)	-.015	.491
Zscore(微血管浸润)	.010	.652
Zscore(神经侵犯)	.052	.058
Zscore(LNR)	.244	.023
Zscore(LNR 分组)	.124	.360
Zscore(总淋巴结数目分组)	-.022	.632
Zscore(肿瘤沉积)	.149	.001
Zscore(总淋巴结数目)	.071	.122
Zscore(阳性淋巴结数)	.291	.019

表 3.9 使用 SPSS 软件通过逐步回归分析得到的自变量显著性与 Beta 值

一般认为，当一个自变量的显著性低于 0.05 时，该自变量对因变量的影响无法忽略，且当 Beta 的绝对值越大，影响越强。在表 3.9 中可以看出，LRGmax，LRGsum，LNR，肿瘤沉积以及阳性淋巴结数（都以浅灰色标注）对于 LRG 的评级有着比较大的影响。

而通过对各种自变量与 LRG 分级之间的线性拟合，可以得到拟合度约为 0.916 的拟合曲线，如下表 3.10 中所示。其中，常数项即在代表因变量数轴上的截距，其他数字为该自变量所代表的斜率。

(常量)	3.249E-16
Zscore(LRGmax)	.773
Zscore(肿瘤沉积)	.161
Zscore(LRGsum)	-.507
Zscore(阳性淋巴结数)	.367
Zscore(pT)	.095
Zscore(总淋巴结数目)	.062
Zscore(LNR)	.309

表 3.10 拟合曲线的各项斜率与截距

可以看出，除了刚刚提到的五个自变量之外，pT 和总淋巴结数目也会对 LRG 产生影响，但它们对 LRG 的影响不及其他五个自变量。

3.5 总结

可以认为，术前影响 LRG 的因素为年龄，术后影响 LRG 的因素为 LRGmax，LRGsum，LNR，肿瘤沉淀以及阳性淋巴结数。

四、问题二的数据处理、建模和解答

4.1 数据处理

为了得到更加合适的数据，本研究仍然需要在第一问的数据处理的基础上，对剩下的数据进行进一步加工优化。

4.1.1 去除手术后的相关指标和数据

本问需要讨论的是通过手术前的数据来判断 LRG，并决定是否需要进行手术以及在手术中是否需要切除淋巴结。因此，对于本问题来说，手术后的相关数据无用，需要去除。

4.1.2 数据降维

数据降维是指去掉无关的因素。在这一步，需要去除的无关因素是 ECOG。由数据表 data.xls 可知，病人的 ECOG 指数基本都是 0，因此，此项数据也可以去除。

性别	年龄	术前CEA	距肛距离	cT	cN	mrTRG	LRG类别
1	73	0	8	4	0	2	1
1	55	0	4	3	2	2	1
1	50	0	7	4	0	2	1
1	39	0	5	3	2	2	1
2	57	0	7	4	2	2	1
2	51	1	6	3	2	2	0
1	63	0	5	3	2	2	1
1	57	0	8	4	1	2	1
1	54	0	2	3	0	2	1
1	53	0	3	4	0	2	1

表 4.1 去除了术后指标并降维后剩下的部分数据

4.2 模型选择

在这一部分，本研究将根据问题的需求，选择合适的模型，以下是选择模型的过程。对于每一个模型，本研究都会利用折线图（黄色代表实际情况，蓝色代表预测情况）和混淆矩阵（行代表真实值，列代表预测值）对预测的结果进行分析。

4.2.1 有序分类模型

问题二需要建立的 LRG 模型，实际上是一个有序分类模型。LRG 有且仅有 6 种类别（即 LRG0、LRG1、LRG2、LRG3、LRG4、LRG5），本研究需要做的是根据做过 nCRT 后患者身体的各项指标（输入），去判断该患者 LRG 的类别（输出），值得注意的是由于 LRG 的各个类别之间的并不是完全无关的，而是存在某种关系（例如 LRG0 与 LRG1 之间显然要比 LRG0 与 LRG5 之间更加远），因此这是一个有序分类模型。为了建立这一有序分类模型，本研究考虑了以下几种模型

（1）支持向量机有序分类模型

支持向量机（Support Vector Machine，SVM）是一种常用的分类模型，它基于统计学习理论中的结构风险最小化原则，通过寻找最优超平面来实现分类。在本次建模中，本研究使用的是一对多的方法：对于 k 个类别的多分类问题，SVM 训练 k(k-1)/2 个二分类模型，每个模型将两个类别区分开来。在预测时，将新样本送入这 k(k-1)/2 个模型中，每个模型投票一次，最终将其分类为得票最多的类别。

通过支持向量机有序分类模型，得到 LRG 预测类别后，与实际类别对比，得到折线图及混淆矩阵如下：

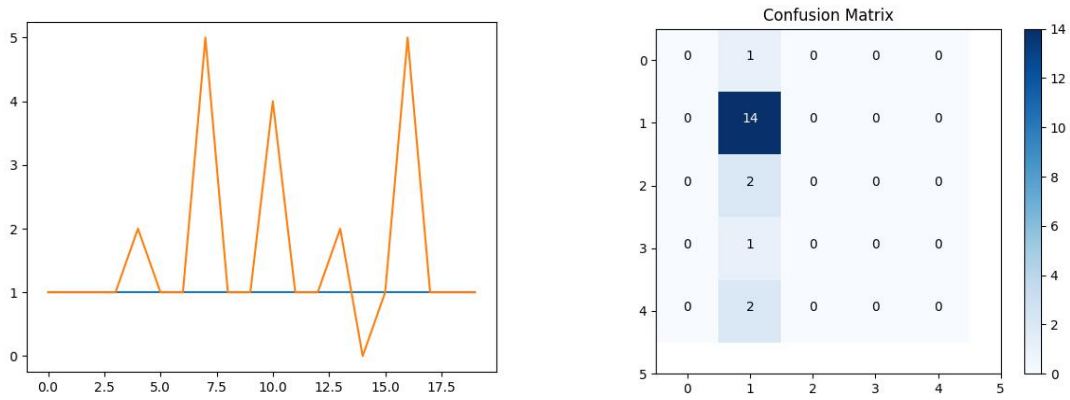


图 4.1 支持向量机有序分类模型的折线图和混淆矩阵

（2）决策树有序分类模型

决策树是一种常见的机器学习算法，基于树的结构进行决策，从根节点开始，沿着划分属性进行分支，直到叶节点。其中内部结点指根结点和中间结点，在这里进行针对某个属性进行判断；分支指该判断的可能结果，属性有多少个取值，就有多少个分支；叶节点：预测结果，即没有其他属性可以继续进行判断的节点。决策树的关键在于如何找合适的“划分属性。划分属性的依据一般基尼系数：

$$Gini(D) = -\sum_{k=1}^{|y|} p_k^2$$

即基尼系数越大，向根节点靠近。在进行决策树算法的过程中，可能需要剪枝：预剪枝（pre-pruning）：提前终止某些分支的生长；后剪枝（post-pruning）：生成一颗完整树，再回头从下往上“修剪”；剪枝是决策树对付“过拟合”的主要手段。

在使用直肠癌患者（nCRT 后，手术前）身体各项相关数据作为决策树的输入，预测 LRG 值的过程中，得到的决策树结构如下图所示：

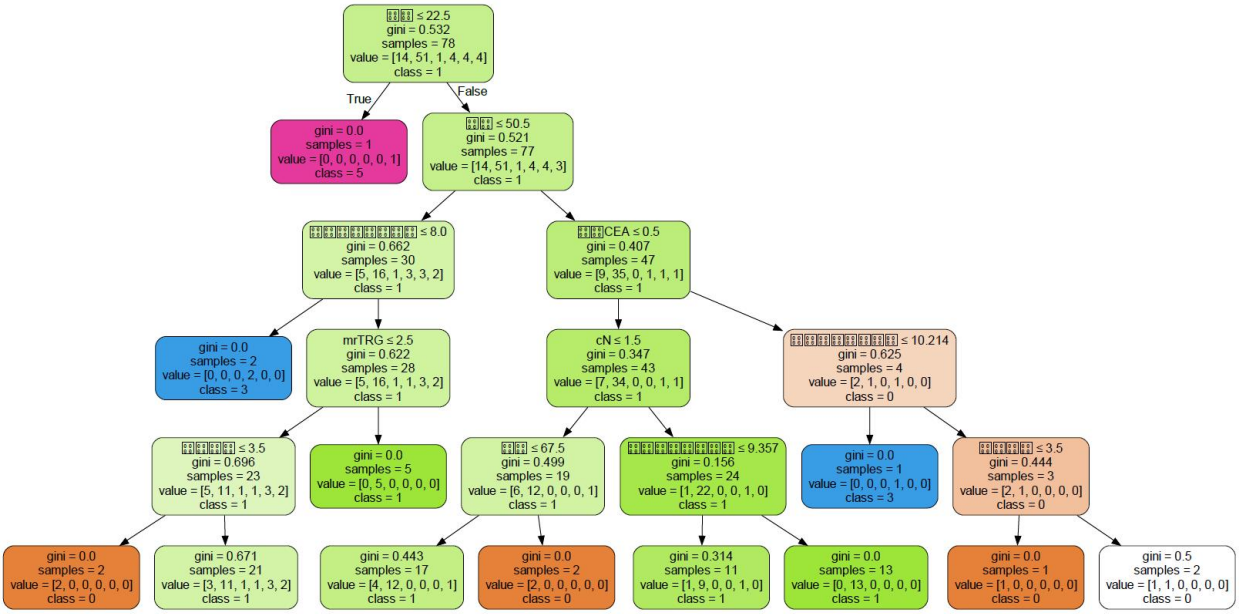


图 4.2 决策树有序分类模型结构

通过决策树，得到 LRG 预测类别后，与实际类别对比，得到折线图及混淆矩阵如下：

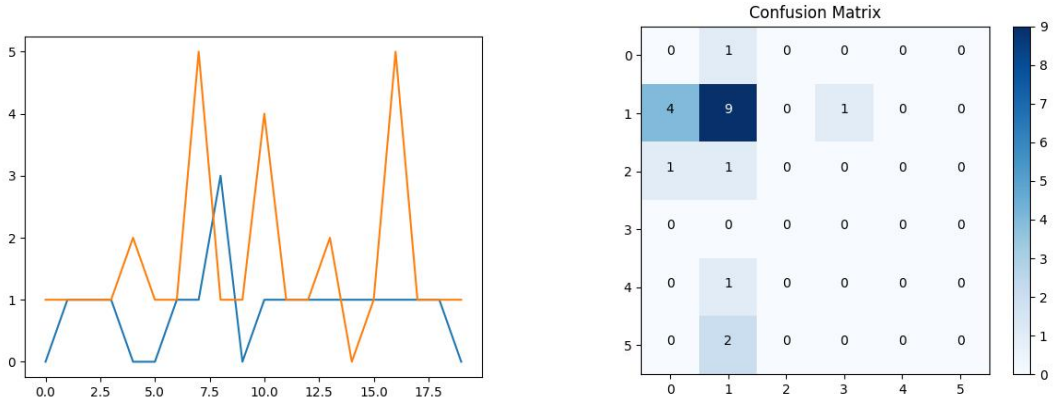


图 4.3 决策树有序分类模型的折线图和混淆矩阵

（3）随机森林有序分类模型

集成学习指将多个分类器组合，从而实现一个预测效果更好的集成分类器。随机森林是集成学习的一种。随机森林采用 Bagging 的思想，Bagging 就是：

- 每次有放回地从训练集中取出 n 个训练样本，组成新的训练集；
- 利用新的训练集，训练得到 M 个子模型；
- 对于分类问题，采用投票的方法，得票最多子模型的分类类别为最终的类别。

随机森林以决策树为基本单元，通过集成大量的决策树，就构成了随机森林。其构造过程如下：

1.构建单个决策树

第一步：T 中共有 N 个样本，有放回的随机选择 N 个样本。这选择好了的 N 个样本用来训练一个决策树，作为决策树根节点处的样本。

第二步：当每个样本有 M 个属性时，在决策树的每个节点需要分裂时，随机从这 M 个属性中选取出 m 个属性，满足条件 m << M。然后从这 m 个属性中采用某种策略（比如说信息增益）来选择一个属性作为该节点的分裂属性。

第三步：决策树形成过程中每个节点都要按照步骤 2 来分裂，一直到不能够再分裂为止。注意整个决策树形成过程中没有进行剪枝。

第四步：按照步骤 1~3 建立大量的决策树，这样就构成了随机森林了。

2.产生最终结果。

众多决策树构成了随机森林，每棵决策树都会有一个投票结果，最终投票结果最多的类别，就是最终的模型预测结果。

在使用直肠癌患者（nCRT 后，手术前）身体各项相关数据作为随机森林的输入，预测 LRG 值的过程中，得到的随机森林中决策树结构可见附件。通过随机森林，得到 LRG 预测类别后，与实际类别对比，得到折线图及混淆矩阵如下：

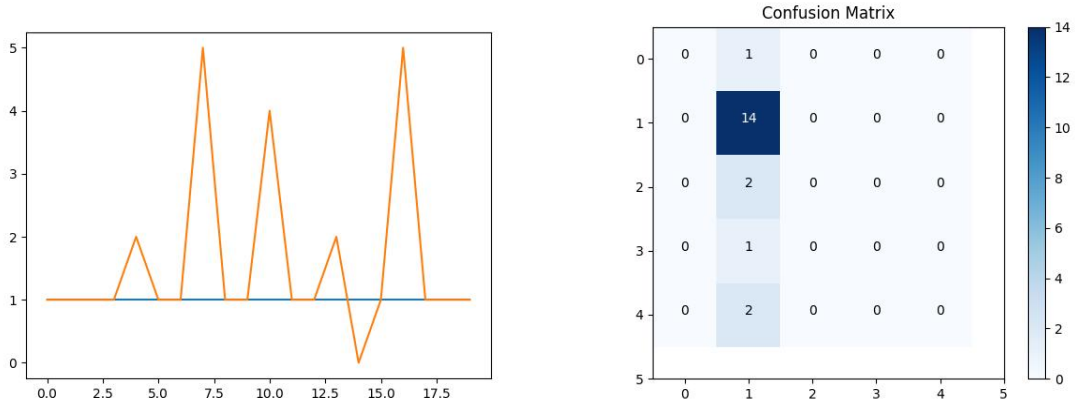


图 4.4 随机森林有序分类模型的折线图和混淆矩阵

（2）残差神经网络有序分类模型

神经网络有序分类模型基本原理是通过多层神经元之间的连接和权重，对输入数据进行特征提取和分类。神经网络分类模型通常包含以下几个组成部分：输入层（接受输入数据，并将其传递给下一层神经元）、隐藏层（对输入数据进行特征提取和转换，其中每个神经元都与上一层的所有神经元相连，并有一定的权重）、输出层（将隐藏层的输出映射到类别标签，并产生输出结果。

神经网络分类模型的训练过程通常包含以下几个步骤：初始化权重（将神经元之间的权重随机初始化）、前向传播（将输入数据传递到神经网络中，通过一系列的线性变换和非线性变换得到隐藏层和输出层的输出结果）计算损失函数（使用交叉熵等损失函数来衡量模型的预测结果与实际标签之间的差异）反向传播（根据损失函数的值，计算输出层和隐藏层中每个神经元的误差，并将误差反向传播回网络中，更新权重参数）、重复迭代（重复执行前向传播和反向传播过程，直到损失函数收敛或达到最大迭代次数）。

在对神经网络进行训练的过程中，出现了梯度消失的情况，因此本研究做出改进，加入了残差机制。

通过残差神经网络有序分类模型，得到 LRG 预测类别后，与实际类别对比，得到折线图及混淆矩阵如下：

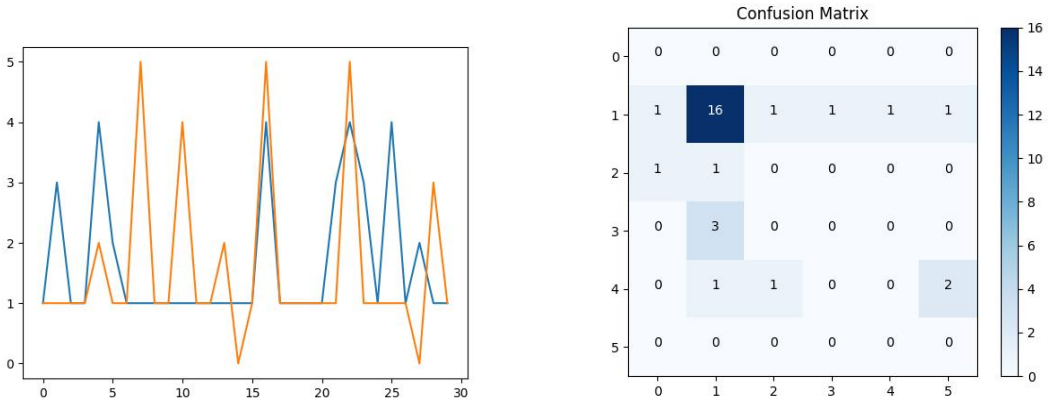


图 4.5 残差神经网络有序分类模型的折线图和混淆矩阵

4.2.2 回归模型

在有序分类模型存在一个很大的问题，类别不平衡。在数据集中有 60%标签均为 1，因此模型倾向于将所有输出都为 1，这样虽然能取得不错的准确率，但对于预测并无什么实际意义。本研究希望能够解决这样的问题，但因为数据量太少，且数据特殊，所以难以进行数据删减和增广操作。本研究决定将原来的有序分类模型改进为回归模型。

回归模型的输出是一个连续的数值，而不是一个离散的类别，但可以将连续的数值分成几个离散的类别，然后再使用分类模型进行处理。在本研究的建模过程中，默认当回归模型输出值大于 5 时，则类别为 5，小于 0 时，则类别为 0，在 0 与 5 之间时，则四舍五入到对应类别。

由于随机森林模型实质上等于多个决策树模型的集成，因此本研究并没有选择将决策树分类模型改进成对应的回归模型，而是另外选择了一种常见的回归模型：线性规划。

回归模型的原理与对应的有序分类模型基本相同，只需要改变一些细节（例如不使用有序分类模型中的独热码，选择其他损失函数等），因此不再一一赘述。以下是本研究在原来有序分类模型基础上改进或建立的几种回归模型：

（1）线性回归模型

线性回归是一种基本的统计学习方法，用于建立一个线性模型来描述自变量和因变量之间的关系。它假设自变量和因变量之间存在一个线性关系，并尝试找到最优的线性函数来描述这种关系。

在线性回归中，通常假设因变量 y 是由自变量 x 和噪声项 ε 的线性组合得到的，即：

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p + \varepsilon ,$$

其中，β0、β1、β2、...、βp 是模型的系数，x1、x2、...、xp 是自变量，ε 是一个随机噪声项。线性回归的目标是找到最优的系数 β0、β1、β2、...、βp，使得模型的预测值与真实值之间的误差最小。

在使用直肠癌患者（nCRT 后，手术前）身体各项相关数据作为自变量，LRG 作为因变量，使用线性回归进行预测，得到的 LRG 预测值与实际值进行对比，得到折线图与混淆矩阵如下：

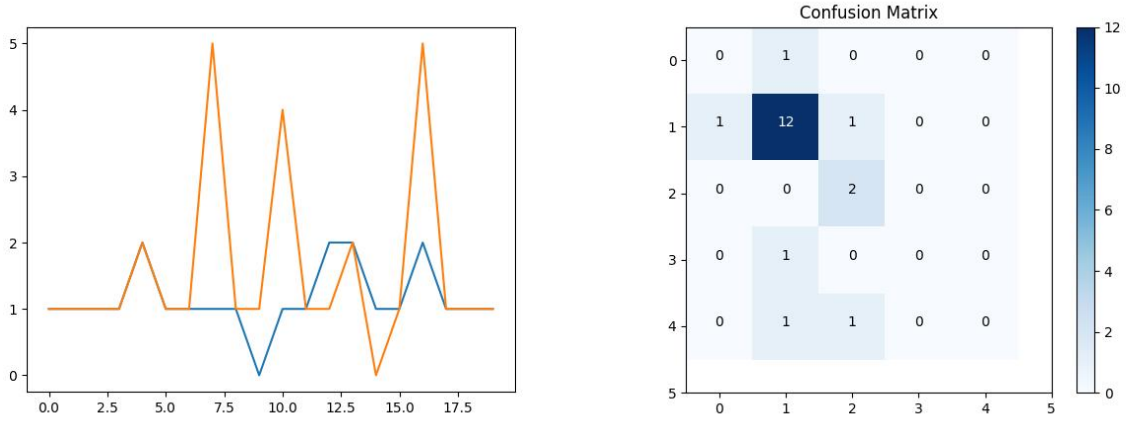


图 4.6 线性回归模型的折线图和混淆矩阵

（2）支持向量机回归模型

支持向量机回归模型目标是找到一个最优超平面，使得样本点到超平面的距离最小化，同时满足一定的容错率。与分类模型不同的是，回归模型中的目标变量是连续的，而不是离散的。使用支持向量机的回归模型得到的预测结果与实际值进行对比，得到折线图与混淆矩阵如下：

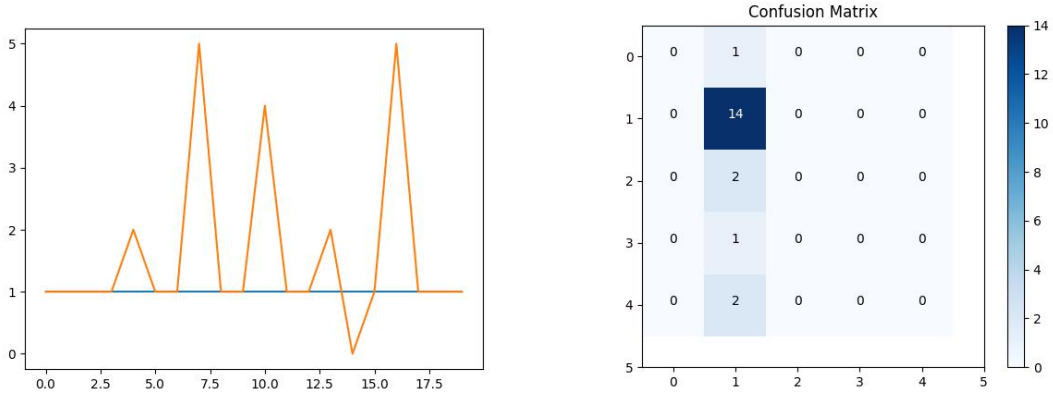


图 4.7 支持向量机回归分类模型的折线图和混淆矩阵

（3）随机森林回归模型

将随机森林分类模型改进成回归模型，需要进行以下几个方面的调整：目标变量类型（随机森林回归模型输出为连续型目标变量），决策树节点评价标准（决策树节点的评价标准为均方误差（MSE）而非基尼系数），叶子节点输出值（叶子节点的输出值为该节点内所有样本的目标变量的平均值），预测结果（预测结果由所有决策树的平均输出值决定）使用随机森林的回归模型得到的预测结果与实际值进行对比，得到折线图与混淆矩阵如下：

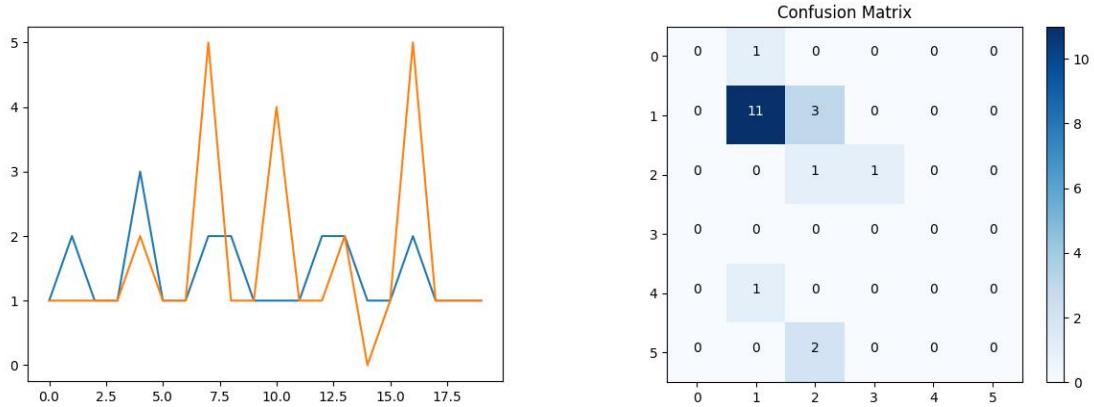


图 4.8 随机森林回归模型的折线图和混淆矩阵

（4）残差神经网络回归模型

神经网络回归模型中，输出层通常只包含一个神经元，其输出结果是一个连续的实数值；且采用 MSE 作为损失函数，衡量预测结果与实际结果之间的差异。通过残差神经网络回归模型，得到 LRG 预测类别后，与实际类别对比，得到折线图及混淆矩阵如下：

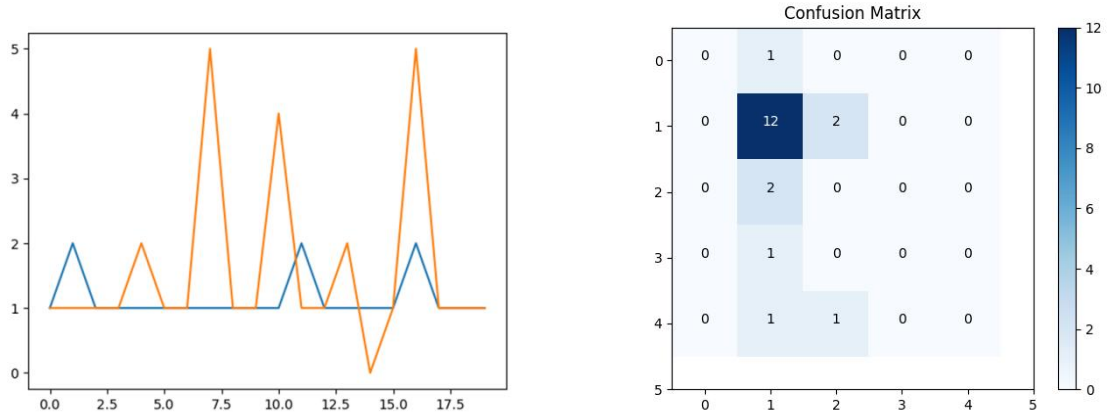


图 4.9 残差神经网络回归模型的折线图和混淆矩阵

4.2.3 基于支持向量机、随机森林和线性规划的集成学习模型

根据对 4.2.1 有序分类模型与 4.2.2 回归模型预测结果的分析，得到以下结论：有序分类模型无法很好地处理类别不均衡的问题（除了决策树有序分类模型外，几乎都出现了对部分数据过拟合的现象，且决策树模型准确率较低），因此本研究希望能够使用回归模型。但 4.2.2 的回归模型中，由于数据太少，残差神经网络模型很难训练。由于 LRG 模型用于医疗方面，这显然是一个严重的缺陷。而其他回归模型中，支持向量机模型仍存在较严重的受类别不均衡影响的情况，而随机森林回归模型与线性规划模型的准确率又偏低，本研究希望能够将它们的优点整合出来，得到一个既可以有效解决类别不均衡影响，又有较高准确率的模型：基于支持向量机、随机森林和线性规划的集成学习模型

4.3 模型构成及预测结果

本研究的目标是学习出一个稳定的且在各个方面表现都较好的模型，但实际情况往往不这么理想，本研究只能得到多个有偏好的模型（弱监督模型，在某些方面表现的比较好，例如支持向量机模型仍存在较严重的受类别不均衡影响的情况，而随机森林回归模型与线性回归模型的准确率又偏低）。集成学习就是组合这里的多个弱监督模型以期得到一个更好更全面的强监督模型，集成学习潜在的思想是即便某一个弱分类器得到了错误的预测，其他的弱分类器也可以将错误纠正回来。单个学习器本研究称为弱学习器，相对的集成学习则是强学习器。

在此次建模中，本研究选择了线性回归、支持向量机、随机森林三个回归器作为集成学习模型的基础模型，同时训练这三个基础模型，之后得到的预测结果，取加权平均值作为最后结果。模型结构如下图 4.10 所示：

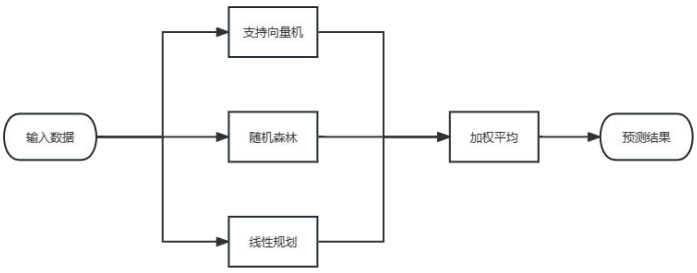


图 4.10 模型结构示意图

通过基于支持向量机、随机森林和线性回归的集成学习模型，得到预测结果与实际结果折线图、混淆矩阵如下图 4.11 所示：

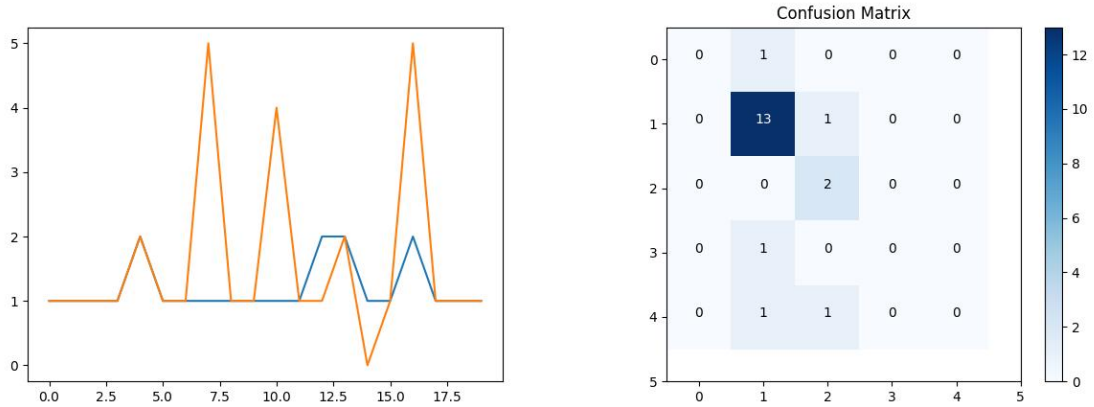
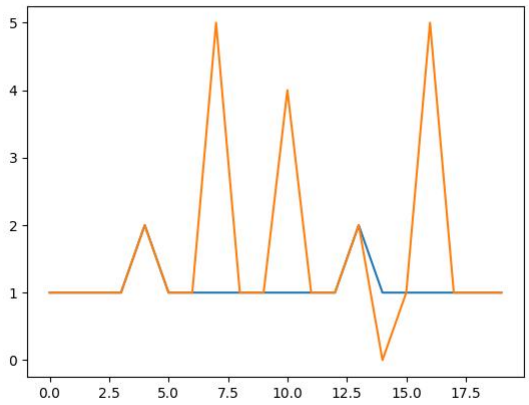
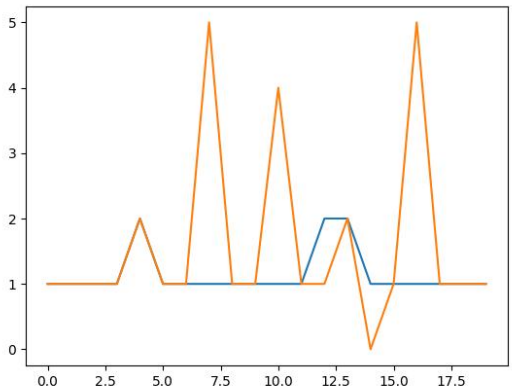


图 4.11 基于支持向量机、随机森林和线性回归的集成学习模型的折线图

除此之外，本研究还进行了对基于支持向量机和随机森林的集成学习模型的分析，得到



其准确度为 7.5（2 种）和 8.0（1 种）时预测结果与实际结果的折线图：



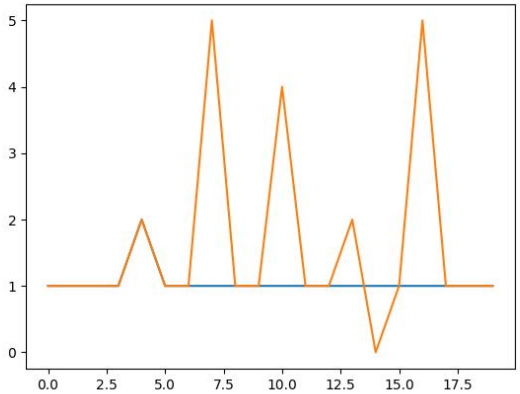


图 4.13 准确度为 8.0 时的折线图

基于支持向量机、随机森林和线性回归的集成学习模型与其他模型的准确率对比如下表：

	准确率		
	最优	最多	最差
SVM	0.7	0.7	0.7
决策树	0.45	0.45	0.45
随机森林	0.7	0.7	0.7
神经网络	0.7	0.65-0.7	0.53
SVR	0.7	0.7	0.7
线性回归	0.7	0.7	0.7
随机森林	0.6	0.6	0.6
神经网络	0.65	0.6或0.65	0.5
SVR+随机森林进行预测	0.8	0.75	0.7
SVR+随机森林+线性规划进行预测	0.75	0.75	0.75

表 4.2 基于支持向量机、随机森林和线性回归的集成学习模型与其他模型的准确率

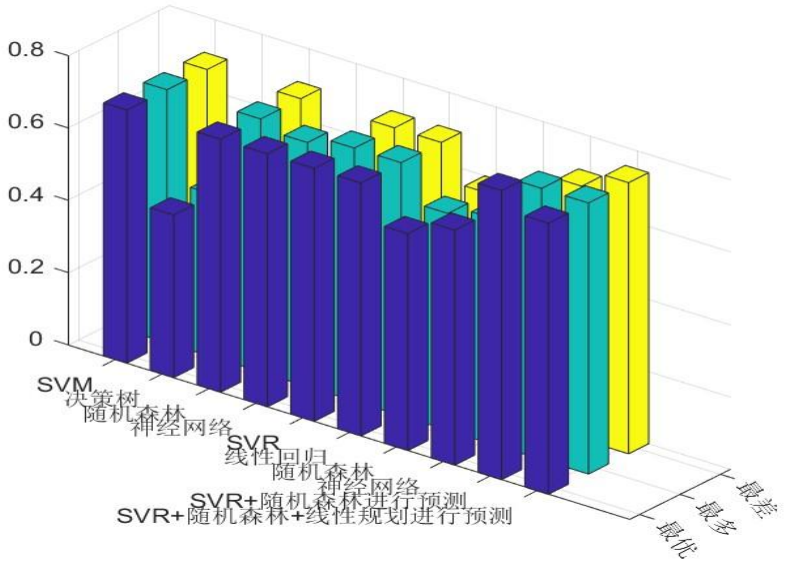


图 4.14 各类模型的准确度示意图

4.4 模型的优缺点

（1）优点：受类别不均衡的影响较小，且模型预测结果相对问题，不会出现训练不起来导致准确率极低或波动较大的情况；

（2）缺点：对于预测 LRG 这类医疗相关对准确率要求极高的问题，0.75 的正确率可能无法很好地满足需求，需要进一步得到更多数据，进行分析与优化模型，以提高模型的准确率。

五、问题三的处理与解答

5.1 不同变量间相关性大小的判断方法

为了了解不同变量之间的相关程度，本研究需要探索不同变量之间的相关性的大小。一般来说，探索不同变量间相关性大小的方法主要有计算变量间的皮尔逊相关系数，斯皮尔曼相关系数以及肯德尔和谐系数。

皮尔逊相关系数主要被用来计算两个变量之间的线性关系程度如何，得到的相关性系数的绝对值介于 0 与 1 之间。当相关性系数越接近于 1 时，说明两个变量之间的关系越紧密，越趋近于正相关。当相关性系数越接近于-1 时，说明两个变量之间的关系越紧密，越趋近于负相关。当相关性系数越接近于 0，说明两个变量之间的关系越小。但需要注意的是，皮尔逊相关系数非常容易收到异常值的影响，也不能测量非线性关系下两个变量间的关系系数。

斯皮尔曼相关系数则是用于衡量两个变量的依赖性的非参数指标[3]，当两个变量完全单调相关时，相关系数的值则为 1 或-1。当两个变量之间完全没有关系时，相关系数的值则为 0。需要注意的是，斯皮尔曼相关系数适用于判断两个非正态分布或有不能剔除的异常值的连续变量之间的相关关系。然而，斯皮尔曼相关系数强调两个变量各自需要有一定的单调性，而不同个体间 LRG 的取值和 TRG 的取值是无关的，因此无法使用斯皮尔曼相关系数进行计算。

肯德尔和谐系数则可以计算多个等级变量的相关程度。与前两者相同的是，肯德尔和谐系数也是处于-1 至 1 之间，且绝对值越大关系越强，绝对值越小关系越小。因为肯德尔和谐系数并不要求变量间必须呈现线性关系或变量内部呈现某种单调性，因此，肯德尔和谐系数是更加适合本题所研究内容的相关系计算系数。

5.2 再述 LRG 与 TRG

很明显，由于评价标准的不同，LRG 和 TRG 都是单独的得到的。LRG 的评级与对应的淋巴

结中纤维化细胞和癌细胞所占的比例有关，而 TRG 则与肿瘤本身的癌细胞剩余量有关[3]。考虑到癌症的原发灶与淋巴结转移之间的关系，讨论 LRG 与 TRG 之间的关系是非常有必要的。同时，如前文所述，nCRT 后直肠癌病人的某些指标可以大致推测 LRG 的分级，pT 则是评估原发灶情况的指标之一，这说明 LRG 与 TRG 之间或许存在某种关系。因此，如果能够通过 LRG 判断 TRG 的情况，将为医生的诊断和评估带来极大的方便。

5.3 利用肯德尔和谐系数得到的结果

利用 SPSS 计算得到的肯德尔系数如下表 5.1 所示。

相关性

			Zscore(术后 TRG)	Zscore(LRGmax)
肯德尔 tau_b	Zscore(术后 TRG)	相关系数	1.000	.179*
		Sig.（双尾）	.	.047
		N	98	98
	Zscore(LRGmax)	相关系数	.179*	1.000
		Sig.（双尾）	.047	.
		N	98	98

*. 在 0.05 级别（双尾），相关性显著。

表 5.1 利用 SPSS 计算得到的肯德尔系数

可以发现，表中得到的 LRG 和 TRG 之间的相关系数约为 0.179，显著性为 0.047。一般认为，显著性在 0.05 级别说明两个变量间存在关系。结合得到的数值，可以认为虽然 LRG 和 TRG 之间存在关系，但是这个关系非常弱。因此，对于同一个体，无法通过 LRG 或者 TRG 推测出另一个值的情况。在涉及到更加精确地判断时，仍然需要同时对原发灶肿瘤和淋巴结的情况进行细致的病理生理分析。

六、模型评估

6.1 模型优点

（1）受类别不均衡的影响较小；

（2）模型预测结果相对稳定，不会出现训练不起来导致准确率极低或波动较大的情况。

6.2 模型缺点

对于预测 LRG 这类医疗相关对准确率要求极高的问题，0.75 的正确率可能无法很好地满足需求，需要进一步得到更多数据，进行分析与优化模型，以提高模型的准确率。

参考文献

[1]医学百科，直肠癌，www.yixue.com/直肠癌#.E6.89.A9.E6.95.A3.E9.80.94.E5.BE.84，访问时间：2023 年 5 月 19 日 12:20

[2]L. He et al., Lymph node regression grading of locally advanced rectal cancer treated with neoadjuvant chemoradiotherapy, World Journal of Gastrointestinal Oncology , 14(8): pp. 1429-1445, 2022.

[3]赵权权 等, 直肠癌原发肿瘤消退分级与淋巴结消退分级关系的研究, 中外胃肠外科杂志, vol 20, no.9, pp.1050-1054, 2017.

非技术性报告

近年来，随着人们生活水平的不断提高，饮食习惯和饮食结构的改变以及人口老龄化，我国直肠癌的发病率与死亡率均保持上升趋势，现已成为我国继肺癌后的第二大癌症。术前新辅助放化疗（nCRT）后接受全直肠系膜筋膜切除术（TME）是局部晚期直肠癌的标准治疗模式，但在接受 nCRT 后是否需要进行 TME 尚难以判断，有学者定义了淋巴结消退分级(LRG），作为接受了 nCRT 的局部晚期直肠癌患者淋巴结治疗反应情况的指标。

为了研究影响 LRG 的关键因素，我们首先对部分临床数据做了预处理。由于单一变量原则，排除了本身患有其他疾病或接受了其他类型手术的患者的数据，考虑到放疗与手术时间差对手术效果的影响，排除放疗与手术时间差过大的患者的数据。同时，按照已有数据的固有规律，我们补全了缺失数据，并对原有数据中的某些异常值做了修改。对经过上述处理得到的最终数据，利用 SPSS 软件对它们做了逐步回归分析分析，得到了不同因素与 LRG 系数之间的相关性。分析比较得到，影响 LRG 的关键因素为接受了 TME 手术后患者的 LRGmax，LRGsum，LNR，肿瘤沉淀数以及阳性淋巴结数。

找出影响 LRG 的关键因素后，更进一步地，我们建立了预测 LRG 的模型：即基于线性回归、支持向量机回归、随机森林回归的集成学习模型。该模型输出通过三个基础模型输出的加权平均得到，不仅可以取得不错且稳定的准确率（我们测试了十余次，准确率均稳定在 0.75），且可以很好地解决类别不均衡（即 LRG 的类别大多数情况下都为 1）的问题。因此，我们得到的模型可以在一定程度上预测 LRG，从而更加精准判断是否需要进行 TME，或在 TME 中是否需要切除淋巴结。

而目前临床医生在 nCRT 后的 6-8 周左右会用肿瘤消退分级（TRG）指标来评估原发肿瘤对治疗的反应，以判断患者是否需要继续接受手术治疗或选择何种手术方式。由此可见，TRG 和 LRG 对患者的疗效与预后的评估同样具有重要临床意义。于是我们对 TRG 和 LRG 之间的关系进行了探讨，通过对 TME 术后部分患者的 TRG 和 LRG 进行相关性分析，得到 TRG 和 LRG 存在较弱的正相关关系的结论。该结论表示 TRG 和 LRG 有关，但是 TRG 不能用来反应 LRG，因此，在直肠癌新辅助放化疗反应的时候需对 TRG 和 LRG 分别进行评估，以更好地对患者的疗效及预后进行评估。

附件说明

附件 1 是数据部分（包括 data.xlsx：某医院自 2019 年 10 月至 2022 年 9 月的直肠癌 nCRT 后手术的临床数据；spss_data.xlsx:去除受其他因素影响数据、补全缺少值后的数据；spss_data_版本 2.xlsx:去除异常值且再次进行补全缺失值后的数据；final_data.xlsx:求得 LRG 类别之后的数据；predict_data.xlsx:在问题 2 用于预测的数据 ；

附件 2 是代码部分（regression_model 中是回归模型的代码（包括集成学习代码），classify_model 中为有序分类模型的代码，CalculatingLRG.py 为问题 1 计算 LRG 类别的代码）；

附件 3 是各模型预测结果的准确率。

附录一 问题一中计算 LRG 的程序

```
import pandas as pd

# 创建 DataFrame 来存储数据
df = pd.read_excel('spss_data_版本 2.xlsx')
df = df.astype({'mrTRG': int, '肿瘤分级': int,'微血管浸润':int,'神经侵犯':int})
# 计算 LRG 的类别
def calculate_lrg(df):
        y_rate = df['阳性淋巴结数']/df['总淋巴结数目']
        y_lrg = 0
        if df['pT'] == 0 and df['pN'] == 0 and y_rate == 0:
                y_lrg = 0
        elif y_rate == 0 and ~(df['pT'] == 0 and df['pN'] == 0):
                y_lrg = 1
        elif y_rate > 0 and y_rate <= 0.25:
                y_lrg = 2
        elif y_rate > 0.25 and y_rate <= 0.5:
                y_lrg = 3
        elif y_rate > 0.5 and y_rate <= 0.75:
                y_lrg = 4
        else:
                y_lrg = 5
        if y_lrg >= df['LRGmax']:
                return y_lrg
        else:
                return df['LRGmax']

df['LRG 类别'] = df.apply(calculate_lrg, axis=1)
print(df[['淋巴结初始报告总计', 'LRG 类别']])
df.to_excel('./final_data.xlsx', index=False)
```

附录二 基于支持向量机、随机森林、线性回归的集成学习回归模型程序

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix
import itertools
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import VotingRegressor
from sklearn.svm import SVR
from sklearn.ensemble import RandomForestRegressor
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
import numpy as np
data = pd.read_excel('predict_data.xlsx', header=None)

X = data.iloc[1:, :-1]
y = data.iloc[1:, -1]


# 对特征进行标准化处理
scaler = StandardScaler()
X = scaler.fit_transform(X)



# 将数据集拆分为训练集和测试集
X_train, X_test, y_train, y_test = train_test_split(X, y, test
_size=0.2, random_state=42)



# 定义三个基模型：支持向量机、随机森林和线性回归
svm = SVR(kernel='linear')
rf = RandomForestRegressor(n_estimators=100, random_state=42)
lr = LinearRegression()
# 定义集成模型，并将三个基模型加入到集成模型中
ensemble = VotingRegressor(estimators=[('svm', svm), ('rf', rf),
 ('lr', lr)])
# 训练集成模型
ensemble.fit(X_train, y_train)



# 使用测试集来评估模型性能
y_pred = ensemble.predict(X_test)
```

```
#  对预测结果进行处理，使其满足题目要求
y_pred  =  np.clip(y_pred,  0,  5)
y_pred  =  np.round(y_pred)
#  计算预测结果与实际结果相同的比例
accuracy  =  np.mean(y_pred  ==  y_test)
print('Accuracy:',  accuracy)

plt.plot(list(range(len(y_test.to_numpy()))),y_pred,y_test.to_numpy())
plt.show()




cm  =  confusion_matrix(list(y_test),  y_pred)
print(cm)
#  绘制混淆矩阵图像
plt.imshow(cm,  cmap=plt.cm.Blues)

#  添加颜色条
plt.colorbar()

#  设置坐标轴标签
classes  =  list(range(6))
tick_marks  =  np.arange(len(classes))
plt.xticks(tick_marks,  classes)
plt.yticks(tick_marks,  classes)

#  设置坐标轴刻度
thresh  =  cm.max()  /  2.
for  i,  j  in  itertools.product(range(cm.shape[0]),  range(cm.shape
[1])):
        plt.text(j,  i,  format(cm[i,  j],  'd'),
                         horizontalalignment="center",
                         color="white"  if  cm[i,  j]  >  thresh  el
se  "black")

#  添加标题
plt.title("Confusion  Matrix")

#  显示图像
plt.show()
```