

中小微企业的信贷决策

摘要

中小微型企业由于缺少抵押资产，贷款风险较高，不能按期偿还和清付贷款利息的可能性较高。银行在给中小微企业发放贷款时，需要综合考虑多方面因素，以求风险最小化和利益最大化。因此，本文通过对企业的偿债能力、盈利能力和信用评级进行评估，同时结合银行不同贷款额度年利率及客户流失率，为银行制定合理的借贷策略，以保证银行在一定时间内既能有较高收益，也能维持客户稳定性。

针对问题一，对附件 1 中的 123 家企业信息进行分析处理，提取出影响信贷风险量化的五个主要因素，其中代表企业偿债能力的因素为销售增长率、销售利润率和销售总额，代表企业信用评级的因素为信誉等级和违约情况。采用基于专家打分（德尔菲法）的模糊层次分析法（FAHP）^[1]，得出五个因素的权重系数矩阵 $W = [w_1, w_2, w_3, w_4, w_5] = [0.155, 0.175, 0.21, 0.235, 0.235]$ ，建立了企业风险量化评估模型。再结合优先级放贷法、信贷额度分级法和名额分配抢占法建立银行信贷策略模型，制定出银行的信贷策略。在银行年度信贷总额固定时的信贷策略为，优先满足风险小的企业的贷款需求，采用信贷额度分级法对贷款额度进行分级，最后使用名额分配抢占法确定银行对 123 家企业的实际信贷策略。

针对问题二，在附件 2 中的 302 家企业缺少信誉等级和违约情况两个指标的情况下，根据附件 1 提供的数据，使用 MLP（多层感知器）神经网络对上述两个指标进行预测。将预测后的 302 家企业完整信息输入到问题一中的企业风险量化评估模型和银行信贷策略模型，得到在银行年度信贷总额为 1 亿元时，对于每家企业的放贷额度。考虑到不同贷款年利率会对客户流失率造成影响，为了使银行在一定年限内的收益最大化，使用动态规划算法，得到银行在 3 年内，贷款额度为 10 万~60 万的最优贷款年利率为 0.0665，贷款额度为 70 万~100 万的最优贷款年利率为 0.0705。最后对于信誉高的企业实行一定年利率优惠政策，保证银行优质客户的留存率。

针对问题三，由于突发因素对不同行业、不同类别的企业影响程度不同，因此在分析 302 家企业名称的组成因子后将企业划分为九大产业。本文以新冠疫情作为突发因素，从 Resset^[6]等数据库中下载到九个产业在新冠疫情影响下的财务数据，并以此作为依据对影响程度进行分析后将影响程度划分为五个等级。依据影响程度等级指标，对银行信贷策略做出以下调整：在企业贷款额度波动值为±20 万的原则下，调高受新冠疫情影响较小的企业信贷额度，降低受新冠疫情影响较大的企业信贷额度，以此重新规划每个贷款额度级别的可获贷款名额。最后使用动态规划算法，计算得出一年内银行对于贷款额度级别为 10 万~100 万的最优年利率均为 0.0745。

关键词：风险量化、信贷策略、模糊层次分析法、神经网络、动态规划

一、问题重述

1.1 问题背景

目前中小微企业发展迅速，为了保证企业发展的稳定性，大量企业需要向银行申请信贷。为了降低贷款风险，银行应尽量给发展稳定、偿债能力高、信誉等级高的企业发放贷款，因此需要对一个企业的信贷风险进行量化。同时为了提高银行收益，需要保证高质量企业的留存率。综上，需要制定出一套信贷风险低、收益高的信贷策略。

1.2 已知数据

附件 1：123 家有信贷记录企业的相关数据

附件 2：302 家无信贷记录企业的相关数据

附件 3：银行贷款年利率与客户流失率关系的统计数据

1.3 提出问题

根据实际和附件的数据信息，通过建立数学模型研究对中小微企业的信贷策略，解决以下问题：

问题一：对已知附件 1 的 123 家企业的信贷风险进行量化分析，给出该银行在年度信贷总额固定时对这些企业的信贷策略。

问题二：在问题一的基础上，对附件 2 中 302 家企业的信贷风险进行量化分析，并给出该银行在年度信贷总额为 1 亿元时对这些企业的信贷策略。

问题三：企业的生产经营和经济效益可能会受到一些突发因素影响，而且突发因素往往对不同行业、不同类别的企业会有不同的影响。综合考虑附件 2 中各企业的信贷风险和可能的突发因素（例如：新冠病毒疫情）对各企业的影响，给出该银行在年度信贷总额为 1 亿元时的信贷调整策略。

二、问题分析

2.1 问题一

问题一是在多指标影响下的决策问题。银行在做出放贷决策时需要考虑企业的偿债能力，同时要综合企业过往的信贷信誉记录，使得银行放贷后承担的风险最小。分析附件 1 提供的 123 家企业数据，我们通过有效的财务进项发票和销项发票流水，求取企业的销售增长率、销售利润率和销售总额，分别代表企业的成长能力、收益能力、规模大小。其次，我们将企业的信誉评级和违约记录数值化。按照银行规定剔除信誉等级为 D 的企业后，对提取出的五个指标，使用基于专家打分（德尔菲法）的模糊层次分析法^[1]，建立企业风险量化评估模型。在信贷策略制定上，采用优先级放贷法，考虑到银行年度信贷总额固定，并且不能满足所有客户的最大需求，所以我们提出信贷额度分级法和名额分配抢占法，结合以上三个方法得出银行信贷策略模型。

2.2 问题二

问题二主要研究在银行可贷额为 1 亿元的条件，银行对 302 家无信贷记录企业放贷的贷款策略，以达到银行收益最大化的目标。由于这些企业缺少信誉等级和违约情况两项指标，无法直接使用企业风险量化评估模型，因此需要根据附件一提供的 123 家企业进项销项信息及上述两项指标，建立神经网络预测模型，对附件 2 所有企业的这两项指标进行预测。之后，采用问题一中的企业风险量化评估模型和银行信贷策略模型，得出这些企业的贷款额度。为了达到利益最大化，需要考虑银行贷款年利率这一指标。由于年利率增高会导致客户流失率增高，从银行长期获益的角度看，需要在年利率和客户流失率之间找到平衡点，因此采用动态规划算法，找到使银行收益最大的信贷策略。

2.3 问题三

问题三是对问题二的进一步深化，考虑突发因素对企业生产和经营的影响，调整银行信贷策略。本文以新冠疫情作为突发因素进行分析。由于突发因素对于不同行业、不同类别的企业影响程度不同，因此先将 302 家企业按照名称划分为多个主要产业，我们从 Resset^[6]数据库中获取了多个行业在新冠病毒疫情前后的数据，包括投资额度、生产量等，通过计算以及作图分析来决定疫情对于各行业的影响程度，并且进行一个影响程度评级。然后，根据影响程度评级调整企业的可贷款额度和不同贷款额度的年利率，使得银行的收益最大化。

三、模型假设

1. 相同贷款额度下的不同企业，贷款年利率相同。
2. 原则上贷款年利率与贷款额度成正相关。
3. 企业名称决定企业所处行业。
4. 企业理论最大可贷款额度为销售总利润的 50%。
5. 企业总是希望获得最高额度的贷款金额。

四、符号说明

符号	含义	单位
C_1	销售增长率	/
C_2	销售利润率	/
C_3	企业规模量化值	/
C_4	违约情况量化值	/
C_5	信誉等级量化值	/
R	企业信贷风险量化值	/
r_{ij}	判断矩阵重要关系	/
p	加性一致性检验指标	/
w_i	指标 <i>i</i> 的权重	/
W	权重矩阵	/
E	信贷额度	元
F	可放贷额度	元
G	实际可放贷额度	元
D_i	贷款额度级别为 <i>i</i> 的贷款额度	元

N_i	贷款额度级别为 <i>i</i> 的企业数量	/
X	所有企业的贷款总额度	元
ZT	银行信贷年度总额度	元
n_i	贷款额度级别为 <i>i</i> 的分配名额	/
n	放贷额度等级 $1 \leq n \leq 10$	/
m	年利率等级 $1 \leq m \leq 29$	/
$dp[n][m]$	放贷等级为 <i>n</i> ，年利率为 <i>m</i> 级时，1~ <i>n</i> 级的放贷利润总和	元

五、模型的建立与求解

5.1 数据预处理

对附件 1、附件 2 做以下处理：

1. 删除信誉评级为 D 的企业数据；
2. 删除进项销项数据中发票状态为作废发票的记录；
3. 利用 Excel 中的数据透视表计算出进项总金额、销项总金额、销项价税合计和 2016 年至 2020 年每个季度的销项金额；
4. 利用上述数据，计算每个季度销售增长率，求其平均销售增长率以及销售增长率方差，对平均增长率和方差进行激活函数 Sigmoid 归一化处理，两者加权得到销售增长率 C_1 ；

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

$$C_1 = \frac{\text{本季度销项总金额} - \text{上季度销项总金额}}{\text{上季度销项总金额}} \quad (2)$$

5. 利用上述数据，计算销售利润率 C_2 ；

$$C_2 = \frac{\text{本季度销项总金额} - \text{上季度销项总金额}}{\text{上季度销项总金额}} \quad (3)$$

6. 对销售总金额进行归一化处理，再使用 SPSS 软件进行系统聚类，聚类数为 5，得到 5 种销售总金额级别，以此来代表企业规模等级，再对规模等级进行量化；

表 1 企业规模量化标准

规模等级	1	2	3	4	5
量化值	1	0.8	0.6	0.4	0.2

7. 把违约情况映射为离散型变量；（附件 2 无此步处理）

表 2 企业违约情况量化标准

是否违约	否	是
离散值	1	0

8. 把信誉等级映射为离散型变量；（附件 2 无此步处理）

表 3 信誉等级量化标准

信誉等级	A	B	C	D
离散值	1	0.8	0.6	0.4

5.2 问题一建模与求解

问题一的求解过程分为两个部分，首先需要建立企业信贷风险评估模型，评估企业信贷风险，再建立银行信贷策略模型来确定银行在年度信贷总额固定时的信贷策略。

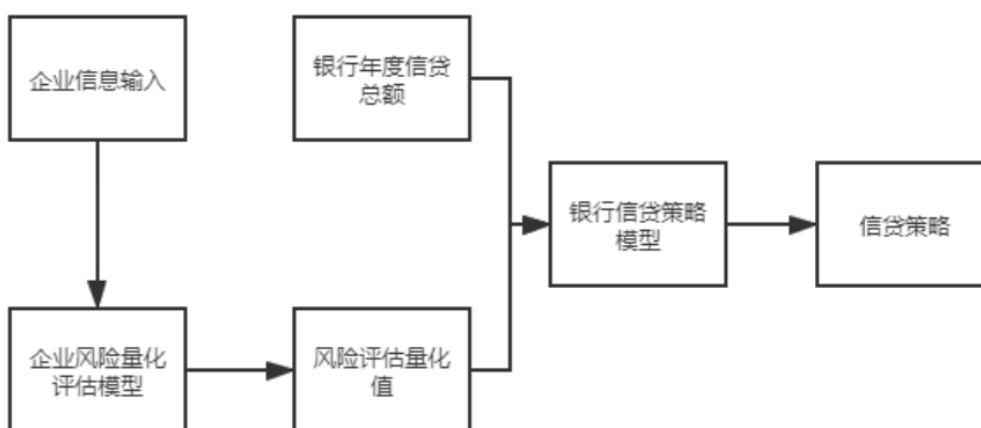


图 1 信贷策略流程图

银行首先根据企业的财务、信誉两方面数据，作为企业风险量化评估模型的输入，在企业风险量化评估模型中，采用模糊层次分析法对输入指标进行权重判断，再根据权重系数来计算企业的风险评估量化值。

在完成风险评估量化之后，将企业的风险评估量化值与银行年度信贷总额作为参数输入到银行信贷策略模型中，银行信贷策略模型中采用优先放贷法、信贷额度分级法和名额分配抢占法三种方法来决定最终银行对于企业的信贷策略。

5.2.1 企业风险量化评估模型的建立与求解

首先信贷风险评估涉及以下五个指标：销售增长率 C_1 、销售利润率 C_2 、企业规模量化 C_3 、信誉等级 C_4 、违约情况 C_5 。每个指标对于企业信贷风险 R 的影响程度不同，所以需要权重矩阵 $W = \{w_1, w_2, w_3, w_4, w_5\}$ ，因此可得到：

$$R = \sum_{i=1}^5 C_i * w_i \quad (4)$$

下面通过模糊层次分析法 (FAHP)^[1]来求解上述五个指标的权重矩阵。

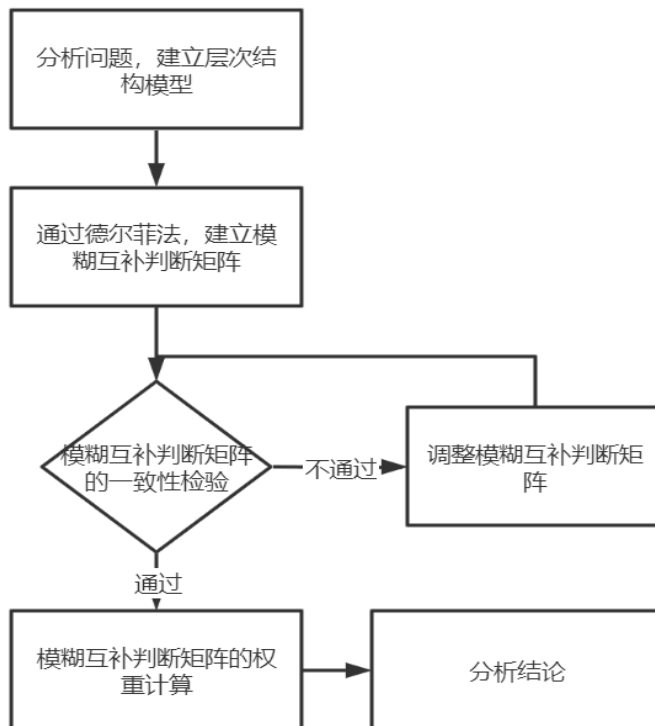


图 2 模糊层次分析法流程图 (FAHP)

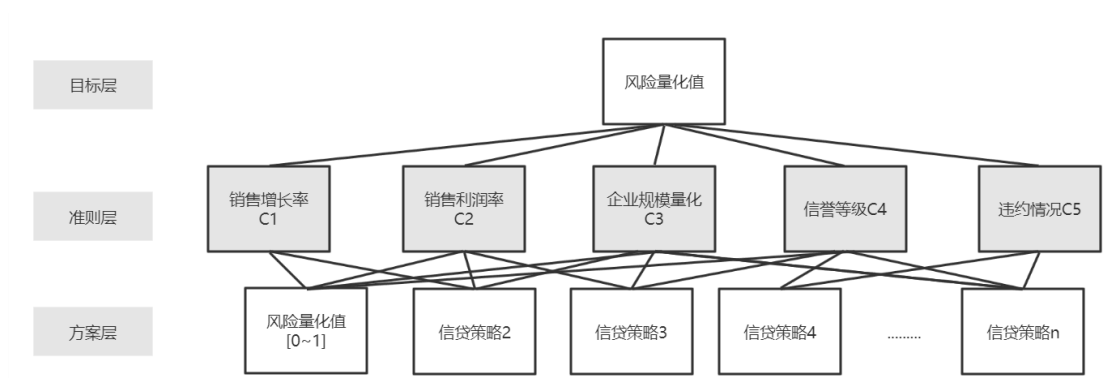


图 3 模糊层次分析法层次结构图 (FAHP)

1. 根据专家打分(德尔菲法)标准来对五个指标进行重要关系的判定，并且生成模糊互补判断矩阵，其中 r_{ij} 表示指标*i*对于指标*j*的重要程度。

表 4 德尔菲法标准

r_{ij}	说明
0.5	同样重要
0.6	稍微重要
0.7	一般重要
0.8	比较重要
0.9	极端重要

表 5 模糊互补判断矩阵

r_{ij}	销售增长率 C_1	销售利润率 C_2	企业规模量化 C_3	信誉等级 C_4	违约情况 C_5
销售增长率 C_1	0.5	0.4	0.3	0.2	0.2
销售利润率 C_2	0.6	0.5	0.3	0.3	0.3
企业规模量化 C_3	0.7	0.7	0.5	0.4	0.4
信誉等级 C_4	0.8	0.7	0.6	0.5	0.5
违约情况 C_5	0.8	0.7	0.6	0.5	0.5

2. 对模糊互补判断矩阵进行加性一致性检验，加性一致性检验指标 ρ 计算如下：

$$\rho = \frac{2}{n(n-1)(n-2)} \cdot \sum_{i=1}^{n-1} \sum_{j=i+1}^n \sum_{k=1, k \neq i, j}^n |r_{ij} - (r_{ik} - r_{jk} + 0.5)| \quad (5)$$

对于上方得到的模糊互补判断矩阵，使用式（5）计算得到：

$$\rho = 0.030000000000000002$$

查询相关资料^[2]，可得到 ρ 的阈值为 $\varepsilon = 0.15$ ，所以在本处 $\rho < \varepsilon$ ，即认为该模糊互补判断矩阵通过加性一致性检验。

3. 使用模糊互补判断矩阵对五项指标分别求得对应的权重 w_i 。

$$w_i = \frac{\sum_{j=1}^n r_{ij} + \frac{n}{2} - 1}{n(n-1)} \quad (6)$$

求得 $W = [w_1, w_2, w_3, w_4, w_5] = [0.155, 0.175, 0.21, 0.235, 0.235]$ 。

4. 采用式（4）即可计算出企业风险评估模型的量化值。

表 6 123 家企业部分风险量化结果

企业号	销售增长率	销售利润率	规模量化	信誉	违约	信贷风险量化结果(数值越高，风险越小)
E15	0.448783	0.946099	1	1	1	0.915128637
E16	0.342088	0.979958	1	1	1	0.904516243
E9	0.493559	0.799812	1	1	1	0.896468853
E7	0.453982	0.757711	1	1	1	0.882966624
E2	0.485477	0.706538	1	1	1	0.878893098
E10	0.384894	0.953294	1	0.8	1	0.85948499
E31	0.485928	0.792158	0.8	1	1	0.851946522
E8	0.485636	0.541365	1	1	1	0.850012364
E24	0.497892	0.451467	1	1	1	0.83617997

E13	0.374606	0.515681	1	1	1	0.828308235
E48	0.481631	0.657869	0.8	1	1	0.827779786
E22	0.487021	0.395606	1	1	1	0.824719266
E18	0.470401	0.317289	1	1	1	0.808437686
E28	0.377075	0.888005	0.8	0.8	1	0.804847466
E42	0.486878	0.999645	0.4	1	1	0.804403833
E3	0.472536	0.781551	1	0.6	1	0.796014548
E54	0.493349	0.684983	0.6	1	1	0.792341207
E17	0.474375	0.213161	1	1	1	0.790831393
E12	0.416933	0.525698	1	0.8	1	0.789621787

详细 123 家企业风险量化值请查看支撑材料：123 家企业风险量化.xlsx。

5.2.2 银行信贷策略模型的建立与求解

我们通过企业风险量化评估模型得出 123 家企业风险量化值，对风险量化值排序，银行优先考虑对风险量化值高（风险小）的企业实施贷款，因此引入优先放贷法。因为银行年度放贷金额有限，不能满足所有企业的贷款需求，因此需要使用信贷额度分级法以获得更多客户，在信贷额度分级法的基础上使用名额分配抢占法使得银行利益最大化和贷款风险较小化。因此银行信贷策略模型的建立需要优先放贷法，信贷额度分级法和名额分配抢占法三种方法。下面对这三种方法进行具体介绍。

1. 优先放贷法：

通过企业风险量化评估模型得到企业的风险量化值排序，对于银行来说，为了使得银行贷款风险最小化，因此优先给风险量化值高（风险小）的企业实施贷款。

2. 信贷额度分级法：

我们通过利润总额来反映企业的还款能力，综合考虑后以 50% 的利润总额作为企业理论信贷额度。由于银行可贷款额度为 10 万-100 万，因此企业理论信贷额度大于 100 万时以 100 万代替。统计后发现，大量企业的可贷额度均为 100 万，占据了银行年度信贷总额的大部分，会使得可贷额度稍低的企业无法获得贷款。故提出了信贷额度分级法，保证大部分企业能获得贷款。

信贷额度分级法执行步骤：

$$F = \frac{R_i}{\max(R)} \times E \quad (7)$$

其中 R_i 表示第 i 个企业的风险量化值， $\max(R)$ 表示所有企业中最大的风险量化值， F 表示可放贷额度， E 表示信贷额度。

将贷款额度分为 10 万，20 万……100 万 10 个放贷等级，在不超过企业偿还能力的前提下，将实际可放贷额度给出如下定义

$$G = F - F \bmod 100000 \quad (8)$$

其中 G 为实际可放贷额度。

以上操作实现了信贷额度分级制。

表 7 123 家企业部分信贷额度

企业号	信贷风险量化结果(数值越高, 风险越小)	利润总额	理论信贷额度	信贷额度	实际可放贷额度
E15	0.915129	221556932.3	110778466.1	1000000	1000000
E16	0.904516	233254176.1	116627088.1	1000000	900000
E9	0.896469	304583017.5	152291508.7	1000000	900000
E7	0.882967	463119627.5	231559813.8	1000000	900000
E2	0.878893	490837695.2	245418847.6	1000000	900000
E10	0.859485	362917578.8	181458789.4	1000000	900000
E31	0.851947	44730180.32	22365090.16	1000000	900000
E8	0.850012	232985051.7	116492525.9	1000000	900000
E24	0.83618	75704556.75	37852278.38	1000000	900000
E13	0.828308	142045850.7	71022925.35	1000000	900000
E48	0.82778	39951948.81	19975974.41	1000000	900000
E22	0.824719	55707469.22	27853734.61	1000000	900000
E18	0.808438	69002184.39	34501092.2	1000000	800000
E28	0.804847	52200638.26	26100319.13	1000000	800000
E42	0.804404	29321686.77	14660843.39	1000000	800000
E3	0.796015	533596137.3	266798068.7	1000000	800000
E54	0.792341	28407192.77	14203596.39	1000000	800000
E17	0.790831	42979478.39	21489739.2	1000000	800000
E12	0.789622	138565127.1	69282563.55	1000000	800000

详细 123 家企业风险量化值请查看支撑材料：123 家企业风险量化.xlsx。

3. 名额分配抢占法

通过观察发现, 贷款风险小, 信誉高的企业大多可放贷额度都较高, 在银行年度贷款金额一定的情况下, 会使得贷款额度等级只集中在高额区域。因此提出名额分配抢占法, 使得更多客户能够获得贷款的同时较小化银行信贷的风险。

① 名额分配原理:

$$X = \sum_{i=1}^{10} D_i \times N_i \quad (9)$$

其中 X 表示所有企业的信贷总额度, D_i 表示 i 这个额度级别的贷款额度, N_i 表示 i 这个额度级别的企业数量。

考虑到当银行信贷年度总额度 $ZT < X$ 时, 我们需要考虑如何让尽可能多的客户获得贷款, 所以采用下面的名额分配方法:

$$n_i = \frac{\frac{D_i \times N_i}{X} \times ZT}{D_i} \quad (10)$$

其中 n_i 为分配给贷款额度第 i 级的名额, 该处采用整数除法, 保证 n_i 为整数。

该方法根据每个贷款额度级别的总贷款额度在所有企业的信贷总额度 X 中的占比来分配目前银行拥有的年度总信贷额度 ZT , 使得当 ZT 较小时, 防止贷款额度级别高的企业直接用尽了年度总信贷额度 ZT , 同时贷款额度级别低的企业也能获得部分贷款。

②名额抢占原理：

由于名额分配后，每个贷款额度级别的名额有限，当信贷能力达到该贷款级别但是因为名额不足而失去贷款机会时，这些企业可以选择进行较低一级的贷款额度进行贷款，当这些企业来到较低贷款额度级别时，因为他们的偿还能力是最强的，所有他们会优先抢占这一级别的名额，这就是名额抢占原理。执行抢占名额的方法，能够使得银行尽量向偿还能力高的企业进行放贷，这就有效提高了信贷的安全性，降低银行信贷风险。

5.3 问题二建模与求解

问题二的求解主要包括三部分。首先需要建立神经网络，对附件一提供的123家企业信息进行训练，对附件二的302家的信誉等级和违约情况做出预测。然后通过问题一的企业风险量化评估模型和银行信贷策略模型，得出银行年度贷款金额为1亿元时这些企业的贷款额度。最后考虑银行年利率和企业流失率的关系，采用动态规划算法，找到使银行收益最大的信贷策略。

1. 神经网络预测信誉等级和违约情况

将123家和302家企业数据导入至SPSS中，使用SPSS的神经网络多层感知器预测模型，将信誉作为需要预测的因变量，增长率，利润率，规模等级作为特征的协变量。设置训练检验比为7:3，其他参数保持默认不动，选择输出ROC曲线，并且保存因变量的预测值及其类别。

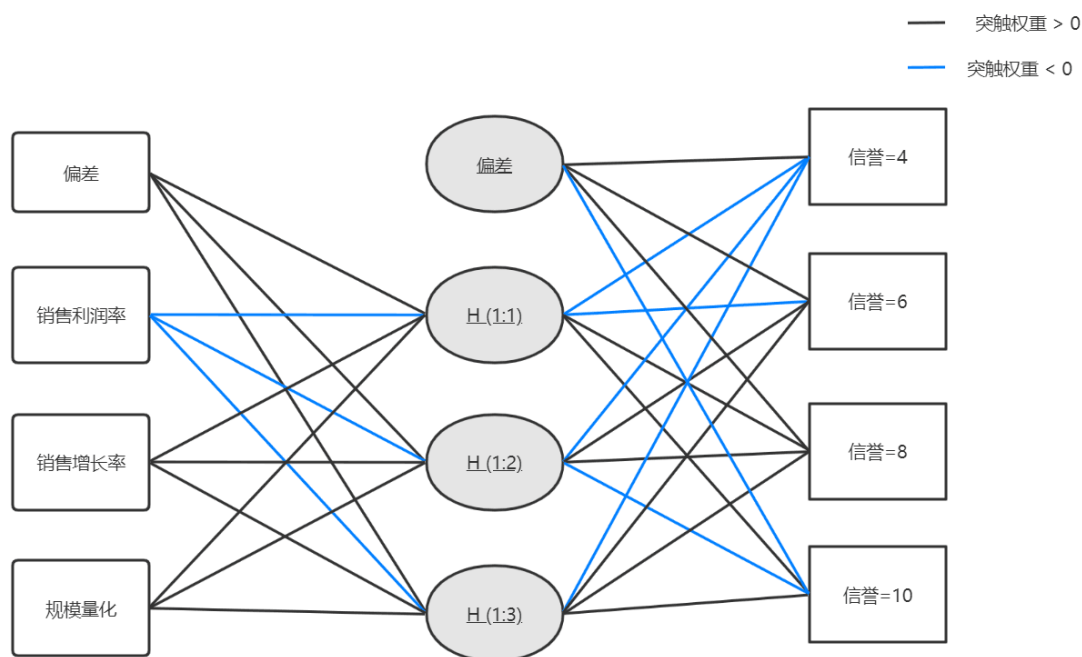
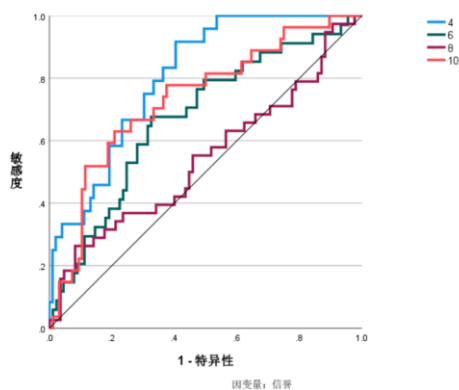


图4 网络结构图

结构图中左边为输入层，中间是隐藏层，右边是输出层。该网络采用三个隐藏神经元，三个特征输入，预测输出为一个四分类结果。优化算法使用梯度下降法，经过多次训练比较，发现初始学习率为0.0001，动能为0.001时训练效果较好。

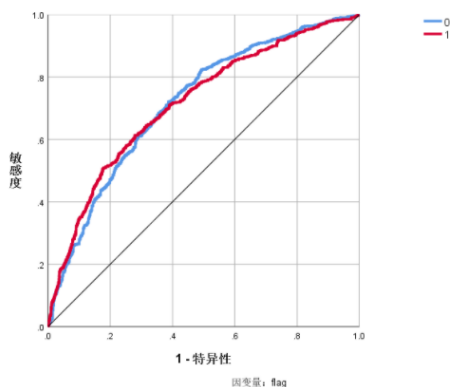


曲线下方的区域

区域		
信誉	4	.806
	6	.673
	8	.544
	10	.734

图 5 信誉等级 ROC 曲线图及下方区域面积

通过获得信誉等级 ROC 曲线下方区域可知预测准确率，可知预测效果较为不错，达到 69%。



曲线下方的区域

区域		
flag	0	.716
	1	.716

图 6 是否违约 ROC 曲线图及下方区域面积

通过获得是否违约 ROC 曲线下方区域可知预测准确率，可知预测效果较为不错，达到 71.6%。

将所得结果导出到 Excel，从而获得 302 家企业的信誉等级和违约情况。

2. 企业风险量化评估模型和银行信贷策略模型获得 302 家企业贷款额度

①通过问题一模糊层次分析法 (FAHP) 计算的五个指标的权重，可计算出企业风险评估模型的量化值。

表 8 302 家企业部分风险量化结果

企业号	销售增长率	销售利润率	规模量化	信誉	违约	信贷风险量化结果(数值越高，风险越小)
E127	0.477968	0.968443	1	1	1	0.923562611
E139	0.481972	0.94196	1	1	1	0.919548755
E185	0.460239	0.943077	1	1	1	0.916375425
E152	0.514703	0.894199	1	1	1	0.916263745
E190	0.432016	0.962311	1	1	1	0.915366995
E134	0.469833	0.914252	1	1	1	0.912818112
E153	0.424674	0.945942	1	1	1	0.911364309
E156	0.411799	0.952294	1	1	1	0.910480328
E149	0.422545	0.938371	1	1	1	0.909709411
E128	0.434008	0.927216	1	1	1	0.909533982
E140	0.413133	0.937196	1	1	1	0.908044781

E161	0.386747	0.955649	1	1	1	0.907184364
E176	0.486077	0.826262	1	1	1	0.89993783
E144	0.380564	0.877864	1	1	1	0.892613726
E126	0.46865	0.78656	1	1	1	0.890288796
E151	0.393292	0.839513	1	1	1	0.887875141
E141	0.48541	0.744625	1	1	1	0.885547924
E193	0.487169	0.742736	1	1	1	0.885489956
E135	0.394737	0.79054	1	1	1	0.879528786

详细 302 家企业风险量化值请查看支撑材料：302 家企业风险量化.xlsx。

②通过对风险量化值排序，使用问题一银行信贷策略模型，算出 302 家企业实际贷款额度以及所处放贷等级。

表 9 302 家企业部分实际放贷结果

企业号	信贷风险量化结果(数值越高，风险越小)	利润总额	理论信贷额度	信贷额度	银行可放贷额度	实际放贷额度
E127	0.923563	695097054.8	347548527.4	1000000	1000000	1000000
E139	0.919549	203220153.9	101610076.9	1000000	900000	900000
E185	0.916375	98308609.99	49154305	1000000	900000	900000
E152	0.916264	115332883.4	57666441.72	1000000	900000	900000
E190	0.915367	137474039.8	68737019.92	1000000	900000	900000
E134	0.912818	127742757.6	63871378.77	1000000	900000	900000
E153	0.911364	303686434.9	151843217.5	1000000	900000	900000
E156	0.91048	179335882.8	89667941.38	1000000	900000	900000
E149	0.909709	107581309.3	53790654.63	1000000	900000	900000
E128	0.909534	245380621.3	122690310.6	1000000	900000	900000
E140	0.908045	217717674.7	108858837.3	1000000	900000	900000
E161	0.907184	184411975.1	92205987.57	1000000	900000	900000
E176	0.899938	90486337.87	45243168.94	1000000	900000	900000
E144	0.892614	105334054.1	52667027.04	1000000	900000	900000
E126	0.890289	511507166.7	255753583.4	1000000	900000	900000
E151	0.887875	96710036.99	48355018.5	1000000	900000	900000
E141	0.885548	127065690.5	63532845.25	1000000	900000	900000
E193	0.88549	112980951.5	56490475.75	1000000	900000	900000
E135	0.879529	131228430	65614215	1000000	900000	900000

详细 302 家企业实际放贷请查看支撑材料：302 家企业风险量化.xlsx。

③动态规划使银行收益最大化

最后考虑银行年利率和企业流失率的关系，采用动态规划算法，找到使银行收益最大的信贷策略。

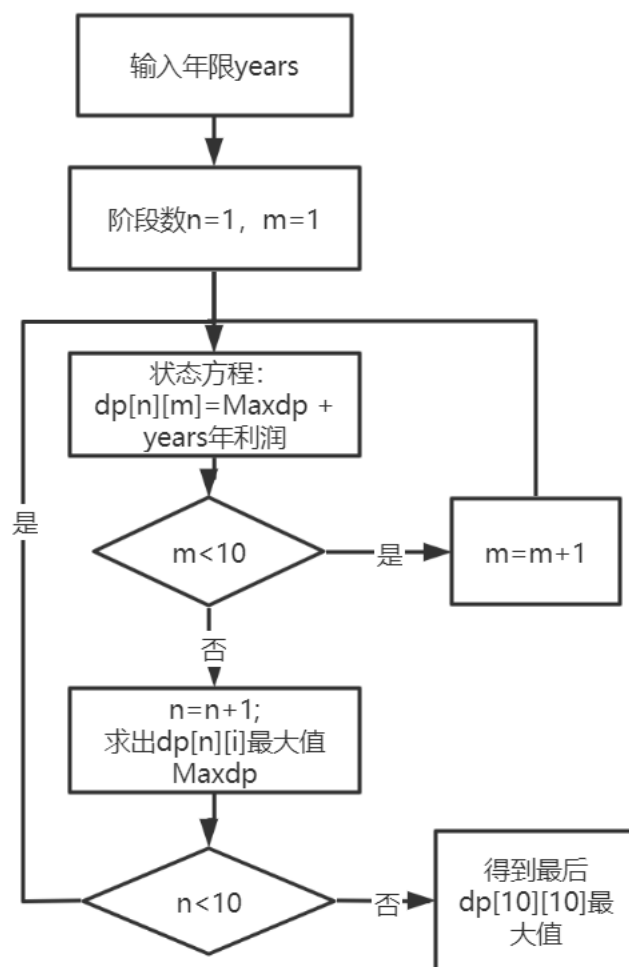


图 7 动态规划流程图

通过前两个步骤，我们获得了 302 家企业最后实际放贷金额，通过名额分配抢占法的计算方法得出银行年度贷款金额为 1 亿元时这些企业的贷款额度。进行统计后得出每一个放贷等级下企业的总个数和 A, B, C 三个信誉等级的个数。

表 10 各贷款额度级别人数信誉级别分布

贷款额度	信誉 A 级人数	信誉 B 级人数	信誉 C 级人数	累计人数
100 万	1	0	0	1
90 万	21	0	0	21
80 万	27	0	0	27
70 万	32	8	0	40
60 万	0	32	0	32
50 万	0	7	0	7
40 万	0	4	3	7
30 万	0	3	4	7
20 万	0	3	1	4
10 万	0	6	15	21

由动态规划法可知，优化目标递推方程为：

$$dp[n][m] = \max(dp[n-1][i]) + get(n, m, years) \quad (11)$$

其中 n 为放贷额度等级， m 为年利率等级， $dp[n][m]$ 表示放贷等级为 n ， n 级放贷等级对应年率为 m 级时， $1 \sim n$ 级的放贷利润总和。 $get(n, m, years)$ 为放贷等级为 n ，年利率等级为 m 的企业，在贷款 $years$ 年内银行的总收益。

动态规划算法源代码见支持材料 P2dynamic.java

因为贷款年数不同对最终结果有所影响，并且对客户流失率高于 60% 的年利率不给予考虑。因此我们将 $years$ 假设为 3，即计算银行三年内最大收益且年利率不大于 0.0785。通过运行程序，在放贷等级 1~6（借款为 10 万~60 万）的企业年利率定为 0.0665，放贷等级 7~10（借款为 70 万~100 万）的企业年利率定为 0.0705。

④优惠政策

银行使用上一步动态规划所得的年利率策略的前提下，可以对信誉高、信贷风险小的企业给予利率优惠。依然采用优先级的方法进行优惠。银行对风险量化排名前百分之十的企业给予年利率优惠，在给此企业贷款年利率的基础上降低一个级别的年利率；对风险量化排名前百分之五的企业给予年利率优惠，在给此企业贷款年利率的基础上降低两个级别的年利率。使用此方法可以使得银行贷款年利率大致趋势不变的情况下对少有的信誉高的企业优惠，提高优秀企业的留存率。

5.4 问题三建模与求解

问题三的模型是对问题一和问题二模型的进一步沿用，考虑在突发因素（本文为新冠疫情）的作用下，银行通过调整放贷额度和年利率，维持收益最大化。

1. 在分析企业名称的组成因子后将 302 家企业划分为九大主要产业。

2. 查阅网络数据库^[6]中各个产业的财务报表，可知新冠疫情爆发初期对所有产业呈负面影响，但随着防控政策的完善和力度的加大，部分产业恢复。部分产业季度利润变化如图。

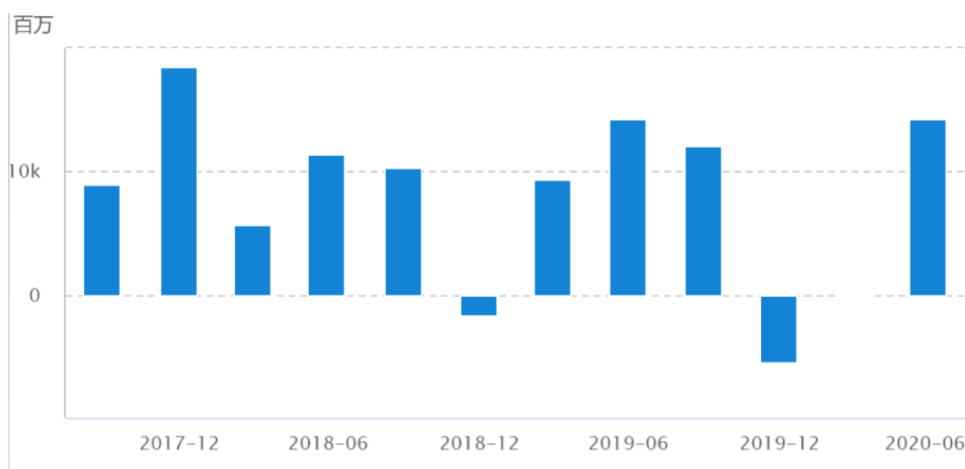


图 8 科技行业季度利润变化图

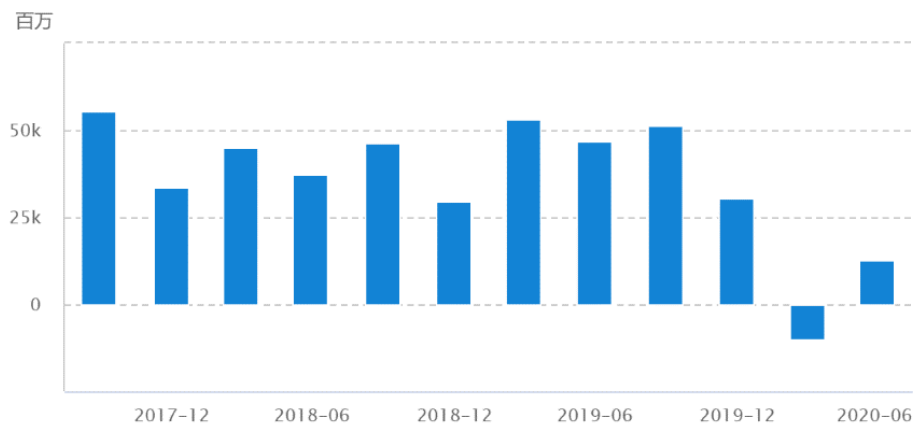


图 9 交通运输行业季度利润变化图

可以将新冠疫情对不同产业的影响划分为 5 个等级。

表 11 受新冠疫情影响程度等级划分

影响程度	基本无影响	影响较小	一般影响	影响较大	影响极大
量化值	1	0.5	0	-0.5	-1

3. 影响程度评级可以反映一个企业还贷能力的波动，因此我们在原有的可贷款额度上，根据影响程度的评级进行额度调整。由于为了使银行收益最大化，应将年度信贷总额尽可能全部贷出，因此制定出如下调整策略：

在问题二的数据基础上，调高受新冠疫情影响较小的企业信贷额度，降低受新冠疫情影响较大的企业信贷额度，从而得到图 13 的数据，并以此重新规划每个贷款额度级别的可获贷款名额。

表 11 影响程度等级对应贷款额度调整标准

影响程度	基本无影响	影响较小	一般影响	影响较大	影响极大
量化值	1	0.5	0	-0.5	-1
额度调整值	20 万	10 万	0	-10 万	-20 万

表 12 302 家企业贷款额度调整部分结果

企业号	信贷风险量化结果(数值越高，风险越小)	银行可放贷额度	突发因素评级	银行放贷额度（受突发因素影响后）
E127	0.923562611	1000000	-1	800000
E139	0.919548755	900000	-1	700000
E185	0.916375425	900000	-0.5	800000
E152	0.916263745	900000	0.5	1000000
E190	0.915366995	900000	-1	700000
E134	0.912818112	900000	-1	700000
E153	0.911364309	900000	-1	700000
E156	0.910480328	900000	-1	700000
E149	0.909709411	900000	-1	700000
E128	0.909533982	900000	-1	700000

E140	0.908044781	900000	-1	700000
E161	0.907184364	900000	-1	700000
E176	0.89993783	900000	1	1000000
E144	0.892613726	900000	-1	700000
E126	0.890288796	900000	-1	700000
E151	0.887875141	900000	-1	700000
E141	0.885547924	900000	0	900000
E193	0.885489956	900000	-1	700000
E135	0.879528786	900000	-1	700000

详细 302 家企业实际放贷请查看支撑材料：302 家企业风险量化-突发因素.xlsx。

4. 最后考虑银行年利率和企业流失率的关系，依然采用问题二使用的动态规划算法，找到使银行收益最大的信贷策略。

通过前面步骤，我们获得了 302 家企业在新冠疫情影响下，银行最后实际放贷金额，依然通过名额分配抢占法的计算方法得出银行年度贷款金额为 1 亿元时这些企业的贷款额度。进行统计后得出每一个放贷等级下企业的 A, B, C 三个信誉等级的个数。

表 13 各贷款额度级别人数信誉级别分布-新冠疫情

贷款额度	信誉 A 级人数	信誉 B 级人数	信誉 C 级人数	累计人数
100 万	5	0	0	5
90 万	7	0	0	7
80 万	13	0	0	13
70 万	25	0	0	25
60 万	31	8	0	39
50 万	1	40	0	41
40 万	0	12	11	23
30 万	0	1	6	7
20 万	0	5	8	13
10 万	0	15	15	30

因为贷款年数不同对最终结果有所影响，并且对客户流失率高于 60% 的年利率不给予考虑。因此我们将 years 假设为 1，即新冠疫情的影响时长为一年。通过运行程序，在放贷等级 1~10（借款为 10 万~100 万）的企业年利率定为 0.0745。

5. 优惠政策

银行对于信誉高的企业同样实施问题二提出的优惠政策，来保证优质客户的留存率

六、模型的评价

6.1 模型的优点

（1）企业风险量化评估模型，是在德尔菲法和模糊层次分析法的基础上建立起来的，获得的五个因素的权重系数矩阵具有较高可靠性和稳定性，因此能够评判一个企业的贷款风险值，对于不同企业数据有较好的适应性和可比较性。

(2) 银行信贷策略模型, 通过优先级放贷、信贷额度分级和名额抢占式三种方法来建立的模型, 能够在降低放贷风险的基础上满足更多客户需求, 在现实应用中具有良好的可行性和较高说服力。

(3) 使用动态规划法求银行利润最优解, 综合考虑到贷款年利率和客户流失率的关系, 找到一个较佳平衡点, 并且能够根据银行考虑年限的不同制定不同的策略, 具有较高灵活性和可操作性。

(4) 在考虑突发因素对放贷金额的影响条件下, 使用的数据客观真实, 能够反映出新冠疫情对不同企业的影响, 结果准确客观, 对于现实具有很好的参考价值。

6.2 模型的缺点

(1) 问题二中使用附件一的 123 家企业作为训练集, 使用神经网络来预测 302 家企业的信誉等级和违约情况。因为条件限制, 训练集的数量不够多, 对于后面的预测准确性有待提高。

(2) 问题三中企业划分因为受可用数据的限制, 只分为 9 个大类, 并且只能以公司名称中关键词作为判断, 严谨性方面有待提高

6.3 模型的改进

(1) 对附件二中 302 家企业信誉等级和违约情况的预测, 后期需要去更多数据库中查询有用资料, 增大训练集个数, 最好将训练集与测试集的比例提高到 3: 1, 使得预测模型准确性较高

(2) 对问题三中企业类别划分标准, 后续应该去参考更多可用文献和更详细划分标准, 查找有用的数据集资源, 使得企业的划分依据有更加细致和严谨。

七、参考文献

- [1] 黄静. 基于模糊层次分析法的银行信贷风险量化研究[D]. 华中科技大学, 2007.
- [2] 徐华, 邱彤, 赵劲松. 模糊判断矩阵构造和矩阵一致性检验的同步方法[J]. 清华大学学报(自然科学版), 2010, 50(06): 913-916. DOI:10.16511/j.cnki.qhdxxb.2010.06.011
- [3] 徐改丽. 模糊互补判断矩阵的一致性及其群体集结方法研究[D]. 广西大学, 2007.
- [4] 姜启源, 谢金星, 叶俊. 数学模型(第四版). 北京: 高等教育出版社, 2011.
- [5] 石少俭, 张弘, 石峥. 动态规划算法的研究[J]. 电脑知识与技术, 2020, 16(18): 48-49. DOI:10.14004/j.cnki.ckt.2020.1856
- [6] <http://www.resset.cn/>

附录

支撑文件列表

文件（夹）名	说明	文件存放路径
123 家企业风险量化	问题一 123 家企业风险量化结果及信贷策略	/123 家企业风险量化.xlsx
302 家企业风险量化	问题二 302 家企业风险量化结果及信贷策略	/302 家企业风险量化.xlsx
302 家企业风险量化-突发因素	问题三 302 家企业风险量化结果及信贷策略	/302 家企业风险量化-突发因素.xlsx
P1judge	问题一中模糊判断矩阵加性一致性检验算法	/源程序/P1judge.py
P2dynamic	问题二及问题三中动态规划算法	/源程序 /P2dynamic.java
SPSSMLPnetwork	问题二中 SPSS 的 MLP 神经网络交互指令	/源程序 /SPSSMLPnetwork.txt
行业数据	问题三中各行业新冠疫情前后营销数据	/行业数据/

问题一源程序：P1judge.py

```
t = [
    [0.5, 0.4, 0.2, 0.2, 0.3],
    [0.6, 0.5, 0.3, 0.3, 0.3],
    [0.8, 0.7, 0.5, 0.5, 0.6],
    [0.8, 0.7, 0.5, 0.5, 0.6],
    [0.7, 0.7, 0.4, 0.4, 0.5]
]
p = 0.0
for i in range(4):
    for j in range(i+1, 5):
        for k in range(5):
            p += abs(t[i][j]-(t[i][k]-t[j][k]+0.5))

p *= 2/(5*4*3)
print(p)
```

问题二源程序：SPSS 交互命令

```
EXECUTE.
DATASET NAME 数据集 1 WINDOW=FRONT.
*Multilayer Perceptron Network.
MLP 信誉 (MLEVEL=N) BY 规模量化 WITH 销售增长率 销售利润率
/RESCALE COVARIATE=STANDARDIZED
/PARTITION TRAINING=7 TESTING=3 HOLDOUT=0
/ARCHITECTURE AUTOMATIC=YES (MINUNITS=1 MAXUNITS=50)
```

```

/CRITERIA TRAINING=BATCH OPTIMIZATION=SCALEDCONJUGATE
LAMBDAINITIAL=0.0000005
SIGMAINITIAL=0.00005 INTERVALCENTER=0 INTERVALOFFSET=0.5
MEMSIZE=1000
/PRINT CPS NETWORKINFO SUMMARY CLASSIFICATION
/PLOT NETWORK ROC GAIN LIFT PREDICTED
/SAVE PREDVAL PSEUDOPROB
/STOPPINGRULES ERRORSTEPS= 1 (DATA=AUTO) TRAININGTIMER=ON
(MAXTIME=15) MAXEPOCHS=AUTO
ERRORCHANGE=1.0E-4 ERRORRATIO=0.001
/MISSING USERMISSING=EXCLUDE .

```

问题二、三源程序：P2dynamic.java

```

import org.junit.Test;

public class test {

    /**
     * 利率（只考虑前 14，后面流失率过高，不考虑
     */
    private static final double rate[]={
        0.04,
        0.0425,
        0.0465,
        0.0505,
        0.0545,
        0.0585,
        0.0625,
        0.0665,
        0.0705,
        0.0745,
        0.0785,
        0.0825,
        0.0865,
    };

    /**
     * A 信用流失率
     */
    private static final double lossA[]={
        0,
        0.094574126,
        0.135727183,
        0.224603354,
        0.302038102,
    };
}

```

```

        0.347315668,
        0.41347177,
        0.447890973,
        0.497634453,
        0.511096612,
        0.573393087,
        0.609492115,
        0.652944774
    };
/**
 * B 信用流失率
 */
private static final double lossB[]={
    0,
    0.066799583,
    0.13505206,
    0.20658008,
    0.276812293,
    0.302883401,
    0.370215852,
    0.406296668,
    0.458295295,
    0.508718692,
    0.544408837,
    0.548493958,
    0.588765696,
};
/**
 * C 信用流失率
 */
private static final double lossC[]={
    0,
    0.068725306,
    0.122099029,
    0.181252146,
    0.263302863,
    0.290189098,
    0.34971559,
    0.390771683,
    0.45723807,
    0.492660433,
    0.513660239,
    0.530248706,
    0.587762408,
};

```

```

};
private static final int numPeople[][]= {
//      {1, 0, 0},
//      {21, 0, 0},
//      {27, 0, 0},
//      {32, 8, 0},
//      {
//          0, 32, 0
//      },
//      {
//          0, 7, 0
//      },
//      {
//          0, 4, 3
//      },
//      {
//          0, 3, 4
//      },
//      {
//          0, 3, 1
//      },
//      {
//          0, 6, 15
//      }
//      {5, 0, 0},
//      {7, 0, 0},
//      {13, 0, 0},
//      {25, 0, 0},
//      {
//          31, 8, 0
//      },
//      {
//          1, 40, 0
//      },
//      {
//          0, 12, 11
//      },
//      {
//          0, 5, 8
//      },
//      {
//          0, 5, 8
//      },
//      {

```

```

                                0, 15, 15
                                }
};

/**
 * 测试函数
 */
@Test
public void runTest()
{
    System.out.println( method2(9, 10, 2));
}

public double [][]dp= new double[10][14];

/**
 * 动态规划自底向上
 * @param n : 额度 n 级
 * @param m : 年利率是 m 级
 */
public double method(int n,int m,int years)
{
    double fanalRate =0;
    double tempRate =0;
    for(int i=m;i>0;i--) {
        double temp=0;
        if(n-1==0)
        {
            temp = getM(0, m, years);
        }else {
            temp = method(n - 1, i, years) + getM(n, m, years);
            tempRate = rate[m];
        }
        if(dp[n][m]<temp) {
            dp[n][m] =temp;
            fanalRate = tempRate;
        }
    }
    System.out.println("第"+n+"层客户");
    System.out.println("使用利率"+fanalRate);
    return dp[n][m];
}

```

```

/**
 * 自顶向下
 * @param n
 * @param m
 * @param years
 * @return
 */
public double method2(int n,int m,int years){
    int i;
    double rating=0;
    double temp=0;
    for(i=0;i<n;i++) {
        for (int j = 0; j < m; j++) {
            if(i == 0)
            {
                dp[0][j] = getM(0, j, years);
                System.out.println(0+"层"+j+"层"+dp[0][j]);
            }else {
                double dpMax=0;
                for(int k=0;k<m;k++){
                    if(dp[i-1][k]>dpMax)
                    {
                        dpMax=dp[i-1][k];
                        rating = rate[k];
                    }
                }
                System.out.println(i);
                System.out.println("层 rating"+rating);
                dp[i][j] = dpMax +getM(i, j, years);
            }
        }
    }

    return dp[n-1][m-1];
}

```

```

/**
 * years 年银行对 n 级的获利, 使用 m 层的为利润率
 * @param n
 * @param years
 * @return
 */
public double getM(int n,int m,int years)

```

```

{
    /**
    * 10W * n
    */
    int money = 10 * (n+1);
    double Tn = 0;
    /**
    * x 为 m 层利率时候平均流失率
    */
    double x = 0;
    int people = numPeople[9-n][0] + numPeople[9-n][1] +
numPeople[9-n][2];
    System.out.println("people"+people);
    x = (lossA[m] * numPeople[9-n][0] + lossB[m] *
numPeople[9-n][1] + lossC[m] * numPeople[9-n][2]) / people;
    for (int i = 0; i < years; i++) {
        Tn += money * (Math.pow((1 - x), i)) * rate[m];
    }
    return Tn;
}
}

```