改进模型指标含义:

R(Recency) 用户最近一次分期时间点与分析截止时间点间的间隔

F(Frequency) 观测时间内的消费次数与分期次数

M(Monetary) 观测时间内的消费金额与额度使用

N(Number) 还款金额为最低还款的次数与负债率

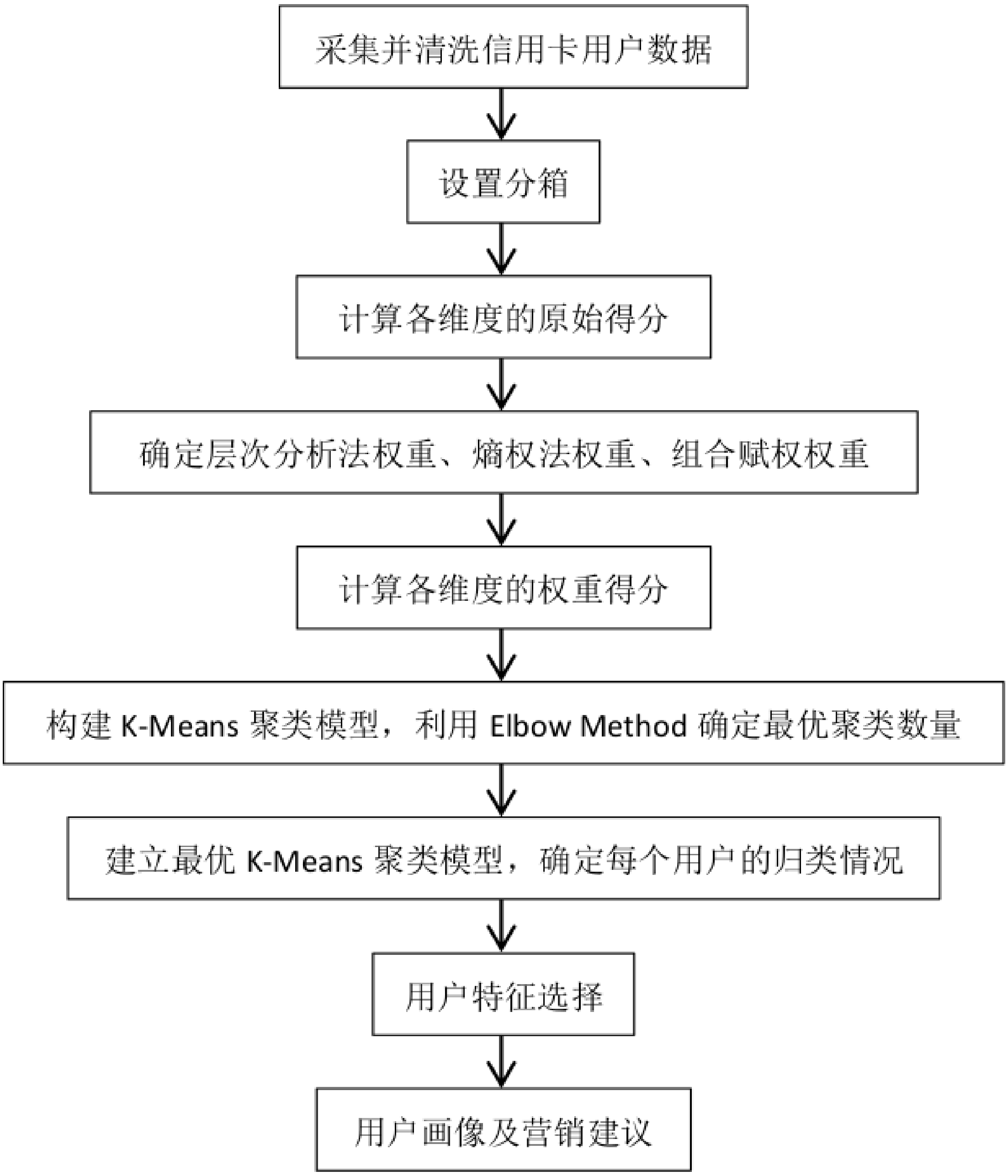
R：最近分期日期越接近分析截止日期，得分越高

F：观测时间内消费次数越多，得分越高，最高 5 分，最低 1 分；

M：观测时间内消费金额越高，得分越高，最高 5 分，最低 1 分；

N：观测时间内还款金额为最低还款的次数越少，得分越高，最高 5 分，最低 1 分；

分值	R	F	M	N
1	[-1.13, -1.00)	[-1.24, -0.84)	[-1.32, -1.09)	[-1.57, -0.78)
2	[-1.00, -0.79)	[-0.84, -0.66)	[-1.09, -0.84)	[-0.78, -0.48)
3	[-0.79, 0.47)	[-0.66, -0.41)	[-0.84, -0.38)	[-0.48, -0.07)
4	[0.47, 0.87)	[-0.41, 0.24)	[-0.38, 0.60)	[-0.07,0.63)
5	[0.87, 4.18]	[0.24, 36.92]	[0.60, 62.49]	[0.63,16.68]


采集并清洗信用卡用户数据
设置分箱
计算各维度的原始得分
确定层次分析法权重、熵权法权重、组合赋权权重
计算各维度的权重得分
构建 K-Means 聚类模型，利用 Elbow Method 确定最优聚类数量
建立最优 K-Means 聚类模型，确定每个用户的归类情况
用户特征选择
用户画像及营销建议

（1）层次分析法下的维度权重计算

使用层次分析法获取三个维度各自的权重，首先需要按照层次分析法的原则，将相关因素分解成三层，得到基于 RFMN 的层次结构。

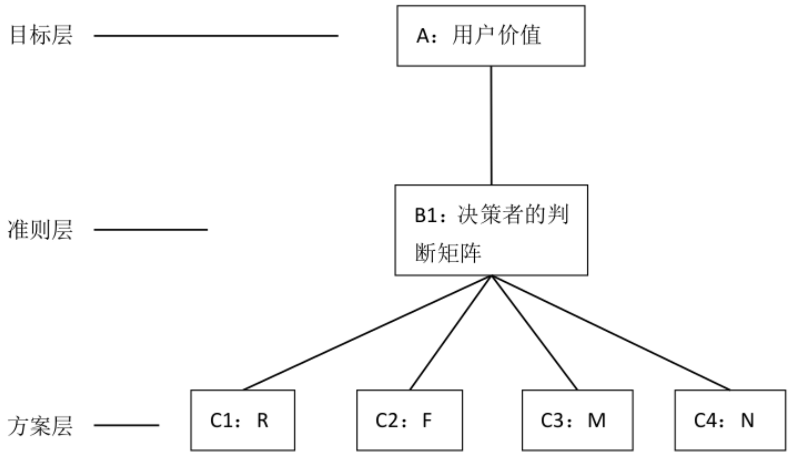


图 3.2 基于 RFMN 的层次结构

其次，根据文献[34]中提到的 1-9 尺度来确定各维度的权重。将专家给出的判断矩阵作为权重的数据来源根据，再通过 SPSSAU 软件进行层次分析，得到各维度的权重 $W_j^{'}$。

（2）熵权法下的维度权重计算

记 RFMN 模型所有维度的原始得分数据集为 $OS=(os_{ij})_{m\times n}$，其中 m 为用户总数，n 为维度个数，$os_{ij}(i=1,2,\dots,m; j=1,2,\dots,n;\text{本文中}n=4)$ 为第 i 个用户第 j 个维度原始得分。

计算在第 j 个维度下，第 i 个用户的维度原始得分所占比重 p_{ij}：

$$p_{ij}=os_{ij}/\sum_{i=1}^{m}os_{ij}. \tag{3.1}$$

记第 j 个维度的信息熵为 E_j：

$$E_j=-\alpha\sum_{i=1}^{m}p_{ij}\ln p_{ij},\text{其中}\alpha=\frac{1}{\ln m}. \tag{3.2}$$

因此，第 j 个维度的权重为 $W_j^{''}$：

$$W_j^{''}=(1-E_j)/\sum_{j=1}^{n}(1-E_j). \tag{3.3}$$

2.2.3 组合赋权法

根据上述理论可知，主观赋权法侧重决策者的意图（即是决策者对不同指标的重视程度），主观性较强；而客观赋权法侧重客观，可能会出现权重和实际相反的情况。而将两者相结合的组合赋权法平衡了两者的优劣。组合赋权法可以弥补单一赋权带来的主观性/客观性过强的不足，同时又保留了主观随机性和客观公正性，使得权重能够实现主客观统一，评价真实公正。

将由主观赋权法得到的属性权重记为 $W^{'}=(W_1^{'},W_2^{'},\dots,W_n^{'})^T$，而由客观赋权法得到的属性权重记为 $W^{*}=(W_1^{*},W_2^{*},\dots,W_n^{*})^T$。两组权重向量分别满足

$$0\le W_j^{'}\le 1,\sum_{j=1}^{n}W_j^{'}=1;0\le W_j^{*}\le 1,\sum_{j=1}^{n}W_j^{*}=1. \tag{2.2}$$

则组合赋权权重为：

$$W=TW^{'}+UW^{*} \tag{2.3}$$

其中，T、U 分别为 $W^{'}$、W^{*} 的重要程度。

为了使得 $W=TW^{'}+UW^{*}$ 中的权重满足

$$0\le W_j\le 1,\sum_{j=1}^{n}W_j=1 \tag{2.4}$$

对 T、U 进行归一化处理。根据文献[33]对 $\overline{T}$、$\overline{U}$ 进行定义：

$$\overline{T}=\sum_{i=1}^{m}\sum_{j=1}^{n}b_{ij}W_j^{'}/\sum_{i=1}^{m}\sum_{j=1}^{n}b_{ij}(W_j^{'}+W_j^{*}) \tag{2.5}$$

$$\overline{U}=\sum_{i=1}^{m}\sum_{j=1}^{n}b_{ij}W_j^{*}/\sum_{i=1}^{m}\sum_{j=1}^{n}b_{ij}(W_j^{'}+W_j^{*}) \tag{2.6}$$

其中，b_{ij} 为规范化后的第 i 位用户第 j 个指标值，m 为用户总数。

2.1.2 K-Means 聚类算法

K-Means 聚类算法是经典的基于划分的聚类算法之一。算法根据簇类数目和初始聚类中心点，基于某种分配原则进行划分，然后重复执行更新中心点和分配对象集的操作，直到每个对象被划分到最合适的簇类中去。最终要达到的目的是使得类中的点尽可能集中，而类间的距离尽可能大。即是准则函数 F 达到最小，此时得到的聚类结果能够实现上述目的，使得得到的聚类结果达到最好。判断聚类结果优劣的准则函数 F：

$$\min F=\sum_{j=1}^{k}\sum_{i=1}^{n_j}\left\|x_i-c_j\right\|^2 \tag{2.1}$$

其中：k 为聚类数目，n_j 为 j 中的数据点总数，x_j 为类 j 中的数据点，c_j 为类 j 的聚类中心。

假设数据集 X 中有 k 个聚类子集 $C_1,C_2,\ldots,C_j$，每个聚类子集的聚类中心分别为 $c_1,c_2,\ldots,c_j$，K-Means 聚类算法的具体步骤如下：

①确定聚类数目 k；

②确定初始聚类中心 $c_1,c_2,\ldots,c_j$；

③依次计算对象 x_i 与每个聚类中心的欧式距离 d，以距离最小为划分标准进行划分；

④计算各新生成类的均值，作为新的聚类中心；

⑤重复③-⑤，直到聚类中心不再改变，终止迭代。

在上述步骤中有以下几个概念需要进行说明。

（1）对象 x_i 与 x_j 间的欧式距离为 $d(x_i,x_j)$，$d(x_i,x_j)=\sqrt{(x_i-x_j)^T(x_i-x_j)}$；

（2）聚类中心 $c_j=\frac{1}{n_j}\sum_{i=1}^{n_j}x_i$。