本次作业的主要目标是对航空公司的客户进行客户价值分析，建立相应的客户价值模型。根建立的客户价值模型，对航空公司的客户进行分类，根据商业上经典的五级客户分类模型，可以将客户分为潜在客户、目标客户、准客户、成交客户、忠诚客户。同时，根据美国数据库营销研究所Arthur Hughes 的研究，客户数据库中有三个神奇的要素，分别为最近一次消费（Recency）、消费频率（Frequency）、消费金额（Monetary）,这三个要素构成了经典的RFM模型，该模型是衡量客户价值和客户创利能力的重要工具与手段，在客户关系管理（CRM）中，占有重要的地位。

根据航空公司的数据属性值显示，数据集包含客户基本信息，乘机信息，积分信息等。在客户基本信息中，多数信息是无用的，例如会员卡号，性别，工作地城市等。依据RFM模型，我们基本可以排除客户基本信息中的全部信息。但是，考虑到客户成为航空公司会员时间的长短（入会时间到窗口观测时间）能够作为客户价值衡量的一种手段。因此，我们扩展了传统的RFM模型，增加了第四种衡量标准——客户关系长度（L）。同时，根据RFM模型，我们可以看出在乘机信息中，我们能够提取到观测窗口时间内的累积飞行里程（M），客户最近一次乘坐飞机距离观测窗口结束的月数（R），客户在观测窗口内乘坐公司飞机的次数（F）三个指标。在乘机信息中，平均折扣率这个属性，是反应客户价值的关键属性，当客户折扣率较高时，可能该客户的价值就偏低。因此，我们引入了AVG\_DISCOUNT(平均折扣率)作为模型建立的第五个衡量标准。在积分信息中，最重要的为总累计积分值，考虑到航空公司的积分值，一般与客户的累计飞行里程挂钩，因此这里则不引入该属性值作为本次数据模型建立的属性值。

因此，在这里我们在RFM模型的基础上，加入了入会时间长度（L），消费时间间隔（R），消费频率（F）、总飞行里程（M），折扣系数平均值（C）五种衡量指标。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 衡量指标 | L | R | F | M | C |
| 说明 | 会员入会时间到观测窗口的时间间隔（单位：月） | 客户最近一次乘坐公司飞机距离观测窗口结束月数的时间间隔（单位：月） | 客户在观测窗口内乘坐公司飞机的次数 | 客户在观测窗口内累计的飞行里程 | 客户在观测窗口内乘坐舱位所对应的折扣系数值 |

根据LRFMC模型，同时采用商业上经典的客户分类方法，将客户分为潜在客户、目标客户、准客户、成交客户、忠诚客户五种类型。根据已有分类数量的标准，分析各种分类算法的特点，我们采用了K-Means算法，这里，K=5,对应着我们对客户的五种分类模式。

根据观察航空公司提供的数据集，我们发现数据中存在属性为空值，属性残缺等问题。因此，本次数据挖掘的主要流程如图1.1所示。

缺失值与异常值分析

航空公司数据集

数据清洗

数据规约

数据标准化

K-means客户分类，模型建立

模型分析，客户价值分析排名

根据所得模型，提出相应的促销策略

**图1.1 航空客户价值分析总体流程**

**第一步、如图1.1所示，因为航空公司数据集中，存在数据缺失，数据为空值等问题，因此首先对数据进行缺失值与异常值分析。**

**第二步、根据第一步的缺失值与异常值分析结果，我们进行相应的数据清洗，去除数据噪声，避免对模型的建立产生干扰。**

**第三步、清洗后的数据中包含了大量的冗余信息，根据我们采用的LRFMC模型，这里我们只需要数据集中的六种属性值，因此，这里我们通过数据规约，去除冗余值，得到相应的L、R、F、M、C的数值。**

**第四步、根据数据挖掘的基本步骤，同时避免数据值分散对模型建立的干扰，这里我们将进行数据的标准化，根据标准化的基本方法，这里我们采用了Z\_SCORE标准化方法。**

**第五步、将标准化好后数据进行K-means分类，得到相应的分类模型。**

**第六步、对模型进行分析，得出不同价值客户的属性特点，针对这些特点，提出相应的营销策略，制定相应的服务方案，提高航空公司的营业额。**

**下面将分布介绍每一步的具体实现流程与算法。**

**2.1 数据缺失值与异常值分析**

**2.2 数据预处理**

**2.2.1 数据清洗**

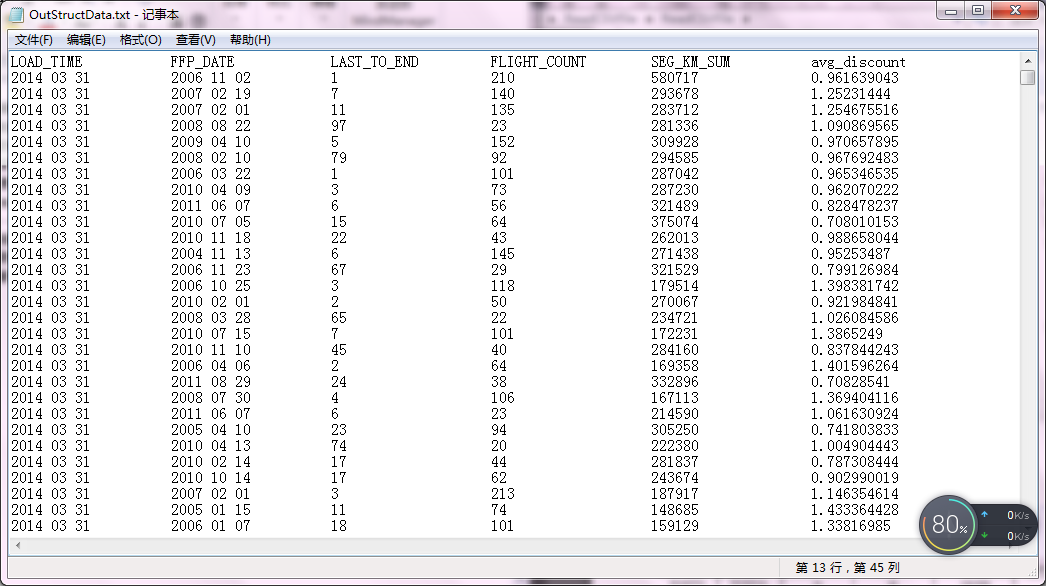
**2.2.2 属性规约**

**当数据清洗完成后，这里我们将进行相应的属性规约操作，根据LFRMC模型，在数据集中，我们需要提取出FFP\_DATE、LOAD\_TIME、FLIGHT\_COUNT、AVG\_DISCOUNT、SEG\_KM\_SUM、LAST\_TO\_END六种属性值，删除其余不相关的所有属性值。**

**算法实现：**

1. **根据数据集，结合CSV文件的特点，判断出相应属性值所在的位置；**
2. **根据属性值所在位置，读取数值，将其存放在一个LRFMC结构体数组中。**

**经过属性规约后的数据集如图1.2所示。**



**图1.2 属性规约后的数据集**

**2.2.3 LRFMC值计算**

**根据LRFMC模型，这里我们需要计算出L、R、F、M、C相应的属性值，根据各自的定义，我们可以得到：L = 观测结束时间到会员入会时间的间隔，R = 最后一次乘机时间到观测窗口结束的时间，F = 观测窗口内飞行的次数，M = 观测窗口内飞行的总公里数，C= 平均折扣率。**

**因此，根据属性规约的值，我们得到如下计算方法：**

**L = (LOAD\_TIME – FFP\_DATE) / 30;**

**R = LAST\_TO\_END/30;**

**F = FLIGHT\_COUNT;**

**M = SEG\_KM\_SUM;**

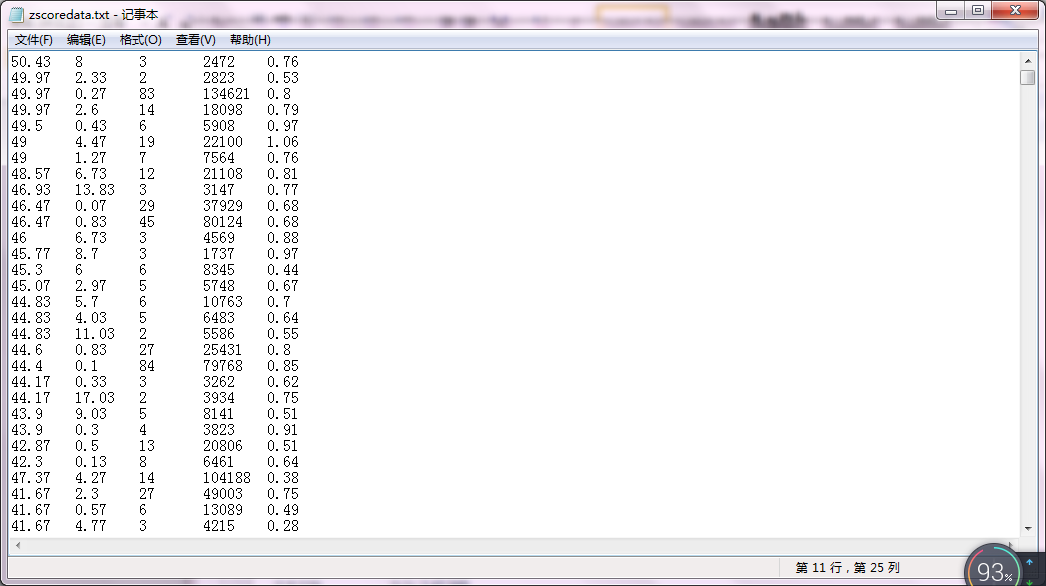
**C = AVG\_DISCOUNT;**

**上式中，L的计算值以月为单位，因此，首先计算间隔天数，然后除以30天，作为月数的间隔。R中同样除以30天作为月数的计算基本单位。**

**算法实现：**

1. **首先，将字符串2014/3/31,以/作为分隔符，分成年月日，存放在Date结构体数组中；**
2. **利用L = （LOAD\_TIME – FFP\_DATE）/30 ,首先计算出日期间隔天数，后面除以30，计算出相应的L值。**
3. **R、F、M、C通过相应的赋值语句得到。**

**计算后的数据集结果图1.3所示**

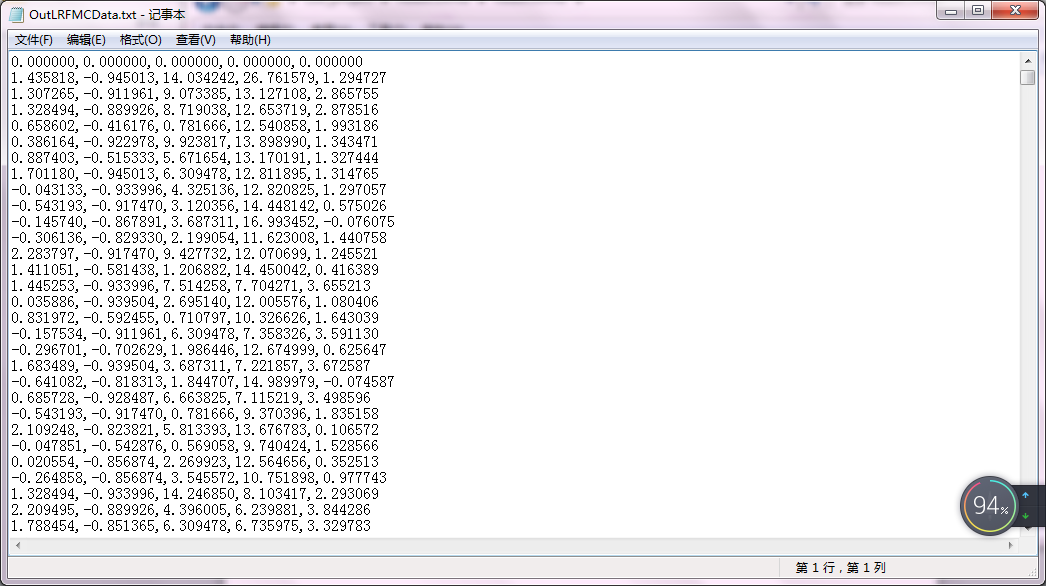


**图 3.1 LRFMC数值计算结果**

**2.2.4 数值标准化**

**这里从众多标准化的方法中，选择了Z\_score标准化。计算公式为：**

**根据上式，我们分别将L、R、F、M、C数值进行了标准化，标准化后的结果如图1.4所示。**



**图1.4 数值标准化后结果**

**3.1 LRFMC客户模型生成**

**根据上面提到的五种客户类型，这里我们采用了K-means算法，将K值赋值为5，然后进行客户分类。**

**3.1.2 客户聚类**

**利用K-means++算法对数据进行聚类，与K-means随机化选择初始化点不同，K-means++算法使得初始的聚类中心之间的相互距离要尽可能的远。**

**算法步骤如下：**

1. **先从我们的数据库随机挑个随机点当“种子点”**
2. **对于每个点，我们都计算其和最近的一个“种子点”的距离D(x)并保存在一个数组里，然后把这些距离加起来得到Sum(D(x))。**
3. **然后，再取一个随机值，用权重的方式来取计算下一个“种子点”。这个算法的实现是，先取一个能落在Sum(D(x))中的随机值Random，然后用Random -= D(x)，直到其<=0，此时的点就是下一个“种子点”。**
4. **重复2和3直到k个聚类中心被选出来**
5. **利用这k个初始的聚类中心来运行标准的k-means算法。**

**代码如下图：**

// use k-means ++ method

void InitializeCorKMeans(){

for (int i = 0; i < sampleNum; i ++) {

sample[i].Clas = -1;

}

int tempIndex = (int)((sampleNum - 1) \* rand() / (RAND\_MAX+0.1)); //产生 0 - (sampleNum-1)的随机数，使其作为seed

for (int i = 0; i < dimension; i ++) {

coreNode[0].address[i] = sample[tempIndex].address[i];

}

double len[sampleNum];

memset(len, 0, sizeof(len));

//计算另外k-1个初始中心

for (int i = 1; i < k; i ++) {

double sumLen = 0;

//更新每个sample距离D(x)

for (int j = 0; j < sampleNum; j ++) {

double minlen = 99999;

for (int m = 0; m < i; m ++) {

if(minlen > distance(coreNode[m], sample[j])){

minlen = distance(coreNode[m], sample[j]);

}

}

len[j] = minlen;

sumLen += len[j];

}

double random = (sumLen) \* rand() / (RAND\_MAX+0.1); // 产生0-sum(D(x))的随机值random

for (int j = 0; j < sampleNum; j ++) {

if(random < 0){

for (int m = 0; m < dimension; m ++) {

coreNode[i].address[m] = sample[j].address[m];

}

break;

}

random -= len[j];

}

}

}

**用K-means++算法进行分类后，结果如下表：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **客 户** | **类 别** | **中心点** | | | | |
| **类 别** | **个 数** | **L** | **R** | **M** | **F** | **C** |
| **类别0** | **24899** | **-0.699468** | **-0.412737** | **-0.163175** | **-0.163749** | **-0.239461** |
| **类别1** | **3822** | **0.098363** | **-0.012033** | **-0.200326** | **-0.198902** | **2.301111** |
| **类别2** | **15807** | **1.158505** | **-0.375845** | **-0.089081** | **-0.096903** | **-0.152706** |
| **类别3** | **12195** | **-0.314916** | **1.682692** | **-0.573924** | **-0.537001** | **-0.166581** |
| **类别4** | **5325** | **0.482255** | **-0.799382** | **2.485566** | **2.425889** | **0.302865** |

**做折线图如下：**

**利用K-means算法分类后的结果如图1.5所示；**

**图 1.5 客户聚类结果及相应中心点**

**3.2 客户价值分析**

**由图1.5所示，**