上海对外经贸大学

2021 -2022 第一学期

《商务大数据案例分析》

案 例 报 告

报告名称：	基于在线零售数据的客户终身价值分析
学　　院：	统计与信息学院
专　　业：	数据科学与大数据技术
学　　号：	18076014
学生姓名：	陈郁欣
课程教师：	李睿
课程编号：	676.002.201

2021　年　11 月

基于在线零售数据的客户终身价值分析

摘要

随着客户关系管理的基础理论的不断完善，客户终身价值作为此理论的重要部分，也逐渐被企业所重视了起来，并且非合约关系下的价值度量一直是研究的难点。本文主要使用了真实商店在线零售数据，使用 BG/NBD 与 Gamma-Gamma 两大概率模型，对客户价值进行了预测，并且细分了群体类型，通过实例阐述客户终身价值分析模型以及细分模型的实现过程，验证了模型拟合有效性的同时，针对不同类型的客户提出了个性化的客户保持与营销策略。

关键词： 客户终身价值 客户群体细分 BG/NBD Gamma-Gamma

在经济发展飞速的现代社会，不同行业内的企业层出不穷，为了能够保持持续发展，避免被其他企业所打败，对于企业而言，最需要注重的问题就是对于不同种类的产品如何找出足以盈利的商业模式，产品种类决定了企业本身的定位以及专长，而找出盈利的商业模式则是运营主管以及营销主管要共同思考的问题。

当企业对商业模式进行决策、对营销预算进行估算的时候，顾客终身价值 Customer Lifetime Value（CLTV）是很关键的数据，企业可以根据此确定规划的营销预算是否会超出获利，也可以通过此寻找真正有价值的客户，在对顾客进行干预后，还可以根据客户生命周期价值的变化优化资源的投放。

一、理论基础

随着全球经济的快速发展，逐渐步入服务经济的新时代，客户演变为企业获取现金流的重要通道之一，同时自然也成为了企业在激烈的行业竞争中获取竞争优势的新来源。企业为了更好的发展，随着消费者市场的改变、行业竞争者的不断涌入，针对传统的市场营销方式也在逐渐转型。传统的营销主要以产品、价格、促销方式等方面以成本-收益的比较为导向，并没有针对客户进行不同的处理，这使得大多数企业在存量市场上陷入增长问题；随着新兴技术的发展，更多企业开始以客户为运营的重心。自1999年，Gartner Group Inc公司提出了CRM概念(Customer Relationship Management 客户关系管理)后，CRM就日益成为了学界和业界关注的重点。实施客户关系管理可以转变企业传统的经营理念，促进企业管理创新，同时也可以更好地定义企业的客户，企业就可以将有限的资源高效低投入到价值较高的客户群体中，从而取得收益的最大化；针对不同价值区间的客户，企业还可以根据各个客户群的特征偏好进行针对性的政策，提升客户满意度

和忠诚度，提高企业利润的同时，为企业带来显著的竞争优势，维系特定的老客户比吸引新客户更加有利可图。

客户价值首先由Roberts和Berger于1998年从企业的角度定义:“客户将来给企业的管理费用和利润所贡献的净现值”。接着，2002年，Cartwright R.首次提出了客户终身价值（Customer Lifetime Value，CLTV）的概念，是基于顾客寿命周期的价值，等于顾客所期望的终身收益减去终身成本。按照80/20法则，企业的少量客户（20%）创造了企业利润的绝大部分（80%），这意味着企业一旦失去了顶部的客户，就会丧失很大一部分利润。

并且，随着互联网、5G、人工智能等新兴技术的高速发展，企业使用大数据等技术记录客户使用及购买产品变得更加便捷，以此可以作为基础来进行深入的挖掘，这为企业进行一些必要的研究提供了有效的数据、效率等资源支撑。

由此，挖掘有高价值的客户，正确细分出客户群体，针对不同价值的客户制定有差别的营销策略是企业的生存之道，客户终身价值（CLTV）的测算和研究成为了客户关系管理的重心，为此提供了很好的解决思路。高级商业智能分析师Matthew Hull谈到该策略的重要性时表示：“该策略有助于营销人员有效地分配预算，以实现目标新客获取量，同时实现回报最大化。它还允许营销人员根据CLTV对客户进行分段和定位。例如，具有较低CLTV的客户需要更多定位，以增加购买频率或订单价值。此外，它可以帮助企业专注于定位可能产生最高CLTV的客户群，同时不太关注获取’一次购买’或’低LTV’客户。”CLTV包括了广义和狭义两个方面，广义上指的是客户在其整个生命周期内为企业带来的利润或损失的净现值，其中包括客户的历史贡献和客户的潜在贡献，而在企业进行客户关系管理的时候更侧重于客户未来的价值，也就是狭义的CLTV，即客户的潜在贡献。

二、研究内容

本文以“基于在线零售数据的客户终身价值分析”为题，以一定时间段内发生在英国的注册非商店在线零售的所有交易为研究对象，通过客户多维度的数据，结合 BG/NBD、Gamma-Gamma 模型以及狭义的 CLTV 模型来对客户价值进行量化分析，对所有交易过的客户进行相关数据挖掘并预测未来对该客户对企业的财务贡献，为后续企业的决策提供一定的理论支持，同时也便于企业细分客户群体，通过对不同客户实施不同类型的服务，提升客户的满意度、忠诚度，从而使得企业收益最大化。

本文的研究主要涵盖以下四个方面：

（1） 首先完成客户终身价值模型的确认，找到适合的模型并构建。

（2） 对客户数据进行系统化的数据预处理工作，基于客户历史行为数据完

成未来的预测，并完成验证模型的拟合验证工作。

（3） 将量化后的客户预测数据整合，进行 CLTV 汇总，完成客户终身价值的量化。

（4） 将量化后的客户终身价值数据进行不同范围的客户群体细分及个性化的营销策略推荐。

基于以上分析，可以针对模型得出的客户终身价值，完成客户的分类和对应精细化手段，从而达到收益最大化。

三、模型原理

本文首先建立两个概率模型，依据客户交易行为，用概率分布来描述客户历史和未来的购买行为数据，分别建模构建客户阶段内购买频率和购买金额的概率分布，其中这两个指标互相独立。

3.1 BG/NBD 模型（贝塔几何/负二项模型）

对于客户终身价值，尤其是非合约关系下的价值计算一直是研究的难点，故本文从非契约客户关系情境下的重复购买行为（需要购买次数>1）出发，BG/NBD 模型，进行消费次数预测。

该模型的假设为：

1. 客户在活跃状态下，一个客户在特定交易时间间隔$t_j - t_{j-1}$内完成的交易量服从交易率为λ的泊松分布。

$$f\left(t_j \middle| t_{j-1}; \lambda\right) = \lambda e^{-\lambda(t_j - t_{j-1})}, t_j > t_{j-1} \geq 0$$

2. 客户的交易率λ服从形状参数为r，逆尺度参数为α的$gamma$分布。

$$f(\lambda | r, \alpha) = \frac{\alpha^r \lambda^{r-1} e^{-\lambda\alpha}}{\Gamma(r)}, \lambda > 0$$

3. 每个客户在交易j完成后流失的概率服从参数为p（流失率）的二项分布，其中假定客户在发生此次购买后立即退出，与实际购买行为有关。

$$P\left(\text{在第}j\text{次交易后立刻变得不活跃}\right) = p(1-p)^{j-1}, j = 1,2,3 \ldots$$

4. 客户的流失率p服从形状参数为a, b的$beta$分布。

$$f(p | a, b) = \frac{p^{a-1}(1-p)^{b-1}}{B(a,b)}, 0 \leq p \leq 1$$

其中$beta$函数$B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$

5. 每个客户的交易率λ和流失率p相互独立。

设在时间段T中客户的交易次数为x，T的第一个交易时间t_0为起点，t_x为最后交易时间。

该模型对于预测客户购买行为的关键表达式如下：

1. 在时长为t的时间内交易数为x的购买概率为$P[X(t)=x]$（整体）

$$P(X(t)=x|r,\alpha,a,b)=\frac{B(a,b+x)}{B(a,b)}\frac{\Gamma(r+x)}{\Gamma(r)x!}\left(\frac{\alpha}{\alpha+t}\right)^r\left(\frac{t}{\alpha+t}\right)^x+$$

$$\delta_{x>0}\frac{B(a+1,b+x-1)}{B(a,b)}[1-(\frac{\alpha}{\alpha+t})^r\left\{\sum_{j=0}^{x-1}\frac{\Gamma(r+j)}{\Gamma(r)\mathrm{j}!}(\frac{\mathrm{t}}{\alpha+\mathrm{t}})^j\right\}]$$

2. 在时长为t的时间内的期望交易次数为$E[X(t)]$（整体）

$$E(X(t)|r,\alpha,a,b)=\frac{a+b-1}{a-1}[1-\left(\frac{\alpha}{\alpha+t}\right)^r\ {}_2F_1\ (r,b;a+b-1;\frac{t}{\alpha+t})]$$

3. 在时长为t的时间内某个客户的期望交易次数$E(Y(t)|X=x,t,T)$

$$E(Y(t)|X=x,t_x,T,r,\alpha,a,b)$$

$$=\frac{\frac{a+b+x-1}{a-1}[1-\left(\frac{\alpha+T}{\alpha+T+t}\right)^{r+x}\ {}_2F_1\ (r+x,b+x;a+b+x-1;\frac{t}{\alpha+T+t})]}{1+\delta_{x>0}\frac{a}{b+x-1}(\frac{\alpha+T}{\alpha+t_x})^{r+x}}$$

通过以下似然函数对r,α,a,b四个参数进行求解：

$$L(r,\alpha,a,b|X=x,t_x,T)$$

$$=\frac{B(a,b+x)}{B(a,b)}\frac{\Gamma(r+x)\alpha^r}{\Gamma(r)(\alpha+T)^{r+x}}+\delta_{x>0}\frac{B(a+1,b+x-1)}{B(a,b)}\frac{\Gamma(r+x)\alpha^r}{\Gamma(r)(\alpha+t_x)^{r+x}}$$

根据客户历史数据，进行参数求解，即可预测客户未来某时间段内的交易次数。

3.2 Gamma-Gamma 模型

由于BG/NBD模型只对客户存续时间和交易次数进行建模，并没有涉及到客户未来交易所带来的现金价值，Fader等以及Colombo和Jiang提出使用Gamma分布代替正态分布拟合，预测在指定时间内的消费金额。

其中模型做了如下假设：

1. 从客户角度来说，交易金额在每个客户的平均交易价值上随机波动。
2. 所观察到的交易价值均值是隐含价值均值$E(M)$的非完美计量。
3. 交易价值均值在客户中是变化的，即使这个值是稳定的。
4. 在客户中的平均交易价值的分布与交易过程无关，也就是说，现金价值与客户购买次数和客户存续时间可以分开建模。

令$z_1,z_2,\ldots,z_x$为客户历史上x次交易的每次交易价值序列，那么所观测到的历史交易价值均值为

$$\tilde{z} = \sum_{j=1}^{x} \frac{z_j}{x}$$

Gamma-Gamma模型认为：

z_j服从$gamma(p,v)$分布，根据$gamma$分布的可加性，在x个交易的总价值的分布服从$gamma(px,v)$，并且根据$gamma$分布的尺度特性，$\tilde{z}$服从$gamma(px,vx)$。（v服从$gamma(q,\gamma)$）。

根据模型参数分布属性，模型的条件期望为

$$E(M|p,q,\gamma,m_x,x) = \frac{(\gamma + m_x x)}{px + q - 1} = \left(\frac{q-1}{px+q-1}\right)\frac{\gamma p}{q-1} + \left(\frac{px}{px+q-1}\right)m_x$$

其中m_x是观测的交易价值均值。

使用最大似然函数，基于历史数据估算模型参数，即可以预测出平均交易价值。

3.3 DCF 模型

贴现现金流折现模型的核心是对净现金流进行折现，也就是计算出未来的现金流入量与流出量的差额，再选择合理的折现率，从而折现得出企业、项目或客户价值评估的模型。

现金流量折现公式来源于计算货币时间价值的现值公式以及复合回报

$$DCF = \frac{CF_1}{(1+r)^1} + \frac{CF_2}{(1+r)^2} + \cdots + \frac{CF_n}{(1+r)^n}$$

$$FV = DCF \cdot (1+r)^n$$

因此，贴现现值（对于一个未来期间的一项现金流量）表示为：

$$DPV = \frac{FV}{(1+r)^n}$$

其中DPV是未来现金流量的折现现值，FV为未来期间现金流量的名义价值，r是贴现率，反映占用资本的成本，n是未来现金流出现之前的年数。

对于从现在起未来数年内的任何时间段t的每个未来现金流量FV，在所有时间段内求和，将总和用作净现值数字。

$$DPV = \sum_{t=0}^{N} \frac{FV_t}{(1+r)^t}$$

通过对未来一定时间的估值，得到相应的客户价值。

四、模型求解

4.1 数据准备

本文使用的数据集是来自于伦敦大学学院的机器学习库

（https://archive.ics.uci.edu/ml/datasets/Online+Retail+II），包含了从2009年12月1日到2011年12月9日之间发生在英国的注册非商店在线零售的所有交易，此企业主要销售独特的全场礼品，并且企业的许多客户都为批发商，一次性购买数量较多。该数据集包含541910条销售数据，分别来自4338名客户。这里本文选择2010-2011年一年的数据进行对未来的预测。

表1 数据集样本

Invoice	StockCode	Description	Quantity	InvoiceDate	Price	Customer ID	Country
536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010/12/1 8:26	2.55	17850	United Kingdom
536365	71053	WHITE METAL LANTERN	6	2010/12/1 8:26	3.39	17850	United Kingdom
536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010/12/1 8:26	2.75	17850	United Kingdom
536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010/12/1 8:26	3.39	17850	United Kingdom

其中“Invoice”代表客户进行的每笔交易的发票编号，“StockCode”和“Description”分别表示购买产品/项目的代码和项目描述，“Quantity”和“Price”则为此笔订单购买的产品/项目总数量以及单价，需要对“Customer ID”中的所有符合标准的客户进行终身价值判断，以便留住对企业来说有利可图的客户。

本文模型部分使用Python的LIFETIMES程序包进行拟合、检验与预测的工作。

4.1.1 数据预处理

本数据集中的部分发票存在操作被中止的情况，需要对这些发票进行集中删除，避免影响对客户购买行为的评价。同时在检查“Quantity”和“Price”数据的时候，发现部分值<0，并且有些订单的“Customer ID”为空，这些非法数据都会对后续建模产生影响，需要进行非法值与缺失值的处理。

数据集也可能存在部分数据异常的情况，如过大或过小，所以在去除空缺值和负值后，绘制箱线图检查“Quantity”和“Price”数值型变量列是否存在异常值。

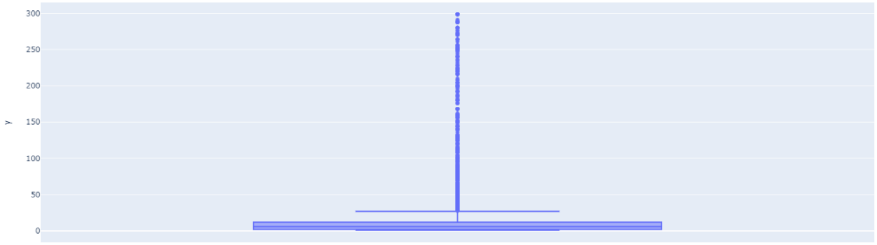

图1 Quantity箱线图

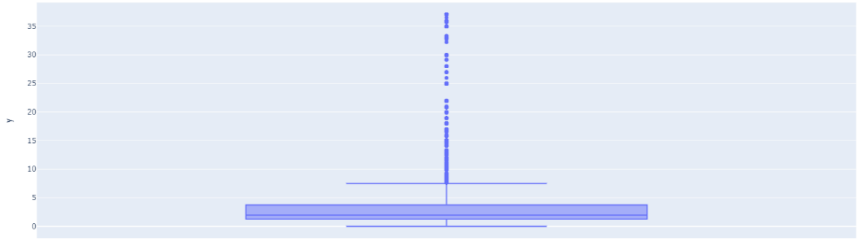

图2 Price箱线图

由上图可知，部分数据点为离群点，由于部分低价值的客户的行为有短时间内大量购买但是不回购的特点，本文需要考虑到可能在购买过程中确实存在部分批发商在某几次购买时大量采购的情况，所以本文对上界和下界的标准放宽。计算数量和价格数据的第1%分位数L和第99%分位U，上下插值IQR为

$$IQR = U - L$$

划定数据的上界为

$$U + 1.5IQR$$

下界为

$$L - 1.5IQR$$

将超出上下界的数值直接归为上界或者下界，以此对数值过于异常的值进行修改。

4.1.2 数据准备

在进行模型求解之前，需要对现有清洗好的数据集进行结构化，转变为模型需要的四列数据。

1. Recency：客户最新一次购买与第一次购买时间天数之差，所以如果只购买了一次的话，那么此指标为0。
2. T：客户在企业的持续周期（年龄），即客户从第一次购买到目前时间为止一共是多少天。
3. Frequency：客户重复购买次数，比总购买次数少1。
4. Monetary_value：客户每次购买的平均价值。

结构化后的部分数据示例如下：

表2 结构化数据示例

Customer ID	frequency	recency	T	monetary_value
12346	0.0	0.0	325.0	0.000000
12347	6.0	365.0	367.0	599.701667
12348	3.0	283.0	358.0	298.540000
12349	0.0	0.0	18.0	0.000000

绘制客户的重复购买次数分布图：

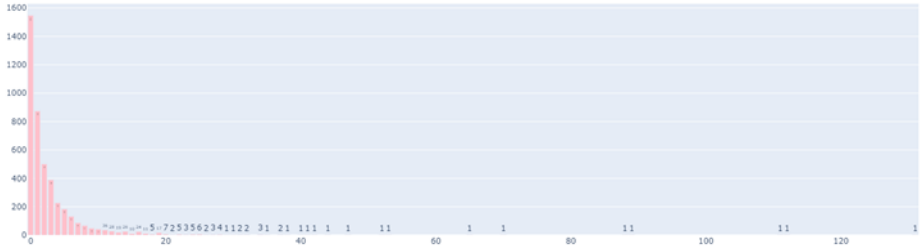

图3 客户重复购买次数分布图

可以观察到，大部分客户都是仅购买一次的客户，但是同时也有部分客户购买次数很多，这些购买次数多的客户可能就是潜在的终身价值较高的客户，需要进一步进行模型分析。

4.2 BG/NBD 模型求解

4.2.1 模型拟合

将生命周期导入BetaGeoFitter，得到模型拟合结果为：

$$a: 0.00, \alpha: 68.92, b: 2.96, r: 0.83$$

4.2.2 模型可视化

绘制频率矩阵图来对客户进行可视化预估。使用客户Recency数据和Frequency数据计算在下一个时间段内进行预期交易数量或进行重复交易的次数。

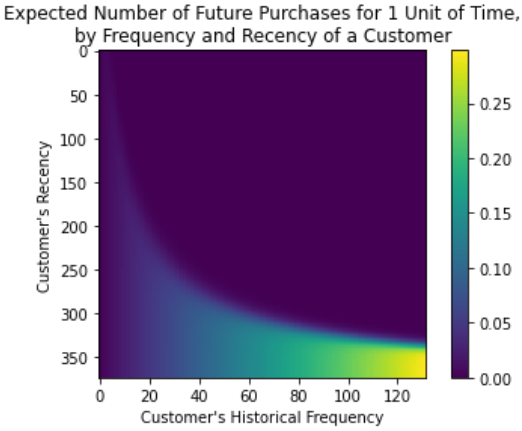

图4 交易次数频率图

从图中可以观察出，右下角的客户可以被认作企业最好的客户，例如某个客户购买了120次，最近一次购买是在客户距离第一次购买350天左右的时候，所以这个客户购买是频繁的；右上角的客户是最冷淡的客户，一次性可能购买很多产品，但是基本只出现了一次；同时，在图片下部还存在一个尾巴，代表不常购买的客户，可能近期对该企业进行过购买，所以不能确定已经流失还是还没有到下一次购买的时间。

同时还可以绘制是否活着概率图来通过购买次数多少和购买时间近远来判断客户是否流失。

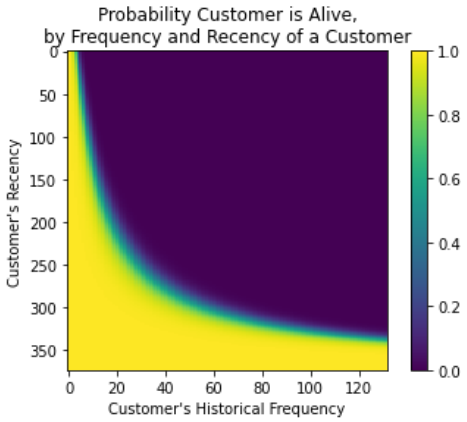

图5 客户是否已流失概率图

4.2.3 模型效果评估

为了检验拟合出来的模型是否正确，本文使用两种方法对模型进行评估。

1. 比较数据与使用拟合模型参数模拟的人工数据。

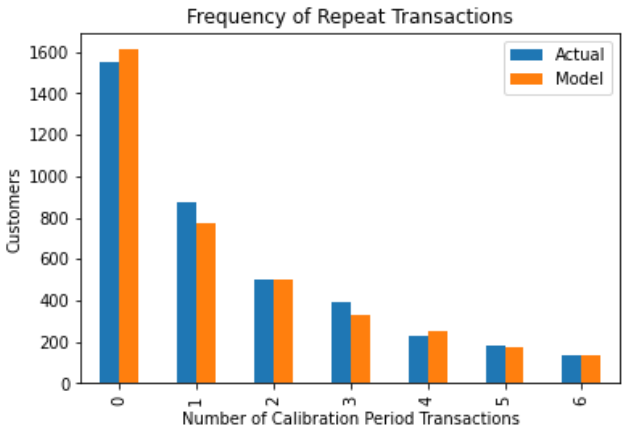

图6 模型拟合效果直方图

由实际数据和模型数据的对比直方图可以看出，在部分频数的数量上存在比较小的差距，可见模型的拟合效果良好。

2. 同时，可以将数据集按照时间划分为校准期数据集和保持期数据集，以此观察模型在尚未见过的数据上的表现，类似于机器学习中的交叉验证。划分后拟合对比曲线如下：

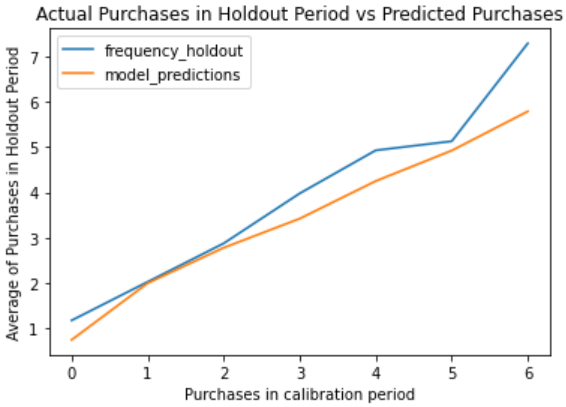

图7 模型拟合效果折线图

校准期间从开始到2011-06-01；保持期从2011-06-01到2011-12-09，该图通过重复购买次数对校准期内所有客户进行分组，然后再保持期中对其重复购买进行平均。橙线和蓝线分别代表模型预测和实际结果，由图可以看出，模型可以比较准确地预测出样本中客户群的行为，在第一次后较有低估。

4.2.4 预测

根据客户历史，我们可以设定需要预测的时间范围，预测每个客户未来的购买情况，这里以90天的预测购买次数部分数据为例。

表3 BG/NBD模型90天预测结果

Customer ID	expected
14911	27.281727
12748	23.346817
17841	23.139717
15311	18.583505

4.3 Gamma-Gamma 模型求解

4.3.1 模型拟合

Gamma-Gamma依赖于一个重要的假设，其假定货币价值和购买频率之间关系较小或没有关系，所以在进行模型拟合之前，需要使用Pearson检查两个向量之间的相关性，检查得出向量之间只存在弱相关性，可以进行下一步拟合。

表4 Pearson相关系数

	monetary_value	frequency
monetary_value	1.000000	0.313821
frequency	0.313821	1.000000

将数据导入GammaGammaFitter，这里仅仅估算至少有一次重复购买的客户，得到模型拟合结果为：

$$p: 3.76, q: 0.34, v: 3.65$$

4.3.2 估算

对每个客户的平均交易价值（平均利润的条件期望）估算部分数据如下

表5 Gamma-Gamma模型部分估算数据

Customer ID	frequency	recency	T	monetary_value	expected	expected_average_profit
18102	25.0	367.0	367.0	8671.891800	5.403680	8733.558156
12415	15.0	313.0	337.0	7556.773333	3.552344	7646.791728
14646	44.0	353.0	354.0	6047.558295	9.687533	6071.943299
15749	1.0	97.0	332.0	4370.040000	0.396023	5308.542943

4.4 CLTV 汇总

最后在前面两个模型的基础上，导入DCF方法计算总CLTV，假定贴现率为0.1，对未来12个月的客户价值进行分析，得到结果如下。

表6 部分客户未来12月的CLTV值

Customer ID	CLV
14646	220682.624509
18102	177054.934677
14096	118069.228846
14911	108354.243778

4.4.1 客户群体细分

首先对预测出的客户12个月的价值进行0-1标准化，便于制定范围进行群体划分。此时的群体划分可以根据企业自身情况，结合经济政策、销售产品特性等进行调整。这里本文划定CLTV值范围为0.8~1的客户为企业的高价值客户，0.6~0.8为较高价值客户，0.3~0.6为较低价值客户，0~0.3为低价值客户。

部分客户的类别划分情况如下：

表7 部分客户群体划分情况

Customer ID	clv	scaled_clv	segment
14646	220682.624509	1.000000	高价值
18102	177054.934677	0.802323	高价值
14096	118069.228846	0.535059	较低价值
14911	108354.243778	0.491040	较低价值
17450	106136.613341	0.480992	较低价值

四种类型的客户存在比例为：

表8 客户群体类型比例

客户群体类型	比例

高价值	0.4610
较高价值	0
较低价值	0.0014
低价值	0.7732
无法判断	0.2250

其中由于模型只能对重复购买次数一次以上的客户进行分析，所以存在部分只进行过一次购买的客户无法判断其价值高低。在全部客户中，低价值的客户占了77.3%很大一部分比重，并且从具体数值可以观察出情况较为分散，企业可以针对此类客户再次进行细分，分别给出不同的管理策略以保证客户价值最大化：

1. 对于价值较低的客户，企业可以在前期的营销环节，加强网页、微信、抖音等新兴且成本较低的渠道的推广，并且可以降低客户门槛，吸引客户进行消费，争取将其逐步发展为高价值的客户。

2. 对于价值非常低的客户，当前价值和潜在价值得分可能都几乎为零，他们的流失可能还会减轻企业维护客户的一部分负担，所以这些客户企业可以考虑放弃。

3. 企业的少数优质客户是该企业当前与未来利润的主要挖掘点，在客户生命周期理论中，他们属于稳定期客户，当前价值与潜在价值都很高，所以应该极力维持与客户之间的关系，可以在一定时间间隔内进行礼品馈赠、询问改善建议等，提供针对性的产品服务，邀请客户参加客户服务节，提升客户忠诚度。

4. 对粘性不够的客户做流失预警监控，制定相应的挽回机制，防止客户被竞争企业抢夺走。

五、结论

本文从客户终身价值的角度研究了英国零售商品的客户情况，在当前竞争越来越激烈的商品销售市场，研究客户价值能够有效提升企业的核心竞争力，加强企业客户关系管理能力，本文围绕批发零售业企业个人客户终身价值，进行了数据分析等工作，并且得到了相应客户的未来12个月价值，对不同类型的客户群体进行了初步划分并给出了应对策略，可以降低企业的经营和运行成本，也可以提高企业经济效益。

但是本文建立的模型存在过多假设需要满足的情况，真实世界的数据存在部分误差可能会对结果造成影响，并且客户群体划分范围通过主观判断进行确定，结果可能与实际情况存在些许偏差，具体范围还需要根据企业自身情况加以修改。

同时对于客户终身价值的问题，针对不同客户，数据方面可以尽可能多地获取客户的其他特征属性，以便更好地更有针对性地提供服务，也可以加以机器学

习的方法，如分类树、广义可加模型、支持向量机等建立客户流失预测模型，改善传统的概率模型的效果，从而为企业客户关系管理提供更多量化依据。

六、参考文献

[1] 连漪,杨硕. (2016). 基于忠诚度的客户价值细分模型构建及其应用. 商业经济研究(14), 4.

[2] 李毅彩. (2019). 基于实现客户终身价值的客户服务策略研究—— 以助听器验配为例. 中国商论(12), 4.

[3] 成栋, 孙莹璐, & 薛薇. (2021). 非合约型客户终身价值的稳健性度量:经典方法与机器学习算法的综合测算研究. (2019-4), 83-98.

[4] 王海力. (2018). F 寿险公司个人客户终身价值测评研究. (Doctoral dissertation, 湖南大学).

[5] 张春莲. (2007). 客户购买行为的 BG/NBD 预测模型及其应用研究. (Doctoral dissertation, 哈尔滨工业大学).

[6] 施雯茜.(2021).基于自由现金流折现模型的沃华医药估值研究(Doctoral dissertation,东华大学).

[7] 胡家瑞.(2020).基于推荐效应的客户终身价值模型优化及实证研究(Doctoral dissertation,北京邮电大学).

[8] 后流量时代企业的掘金之道：如何定义客户终身价值（CLTV）？_营销. (2021). Retrieved 28 June 2019, from https://www.sohu.com/a/323535253_613637

[9] 用户增长 - BG/NBD 概率模型预测用户生命周期 LTV（二）_素质云笔记/Recorder...-CSDN 博客. (2021). Retrieved 22 November 2021, from https://mattzheng.blog.csdn.net/article/details/115909704?spm=1001.2101.3001.6650.1&utm_medium=distribute.pc_relevant.none-task-blog-2~default~CTRLIST~default-1.no_search_link&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2~default~CTRLIST~default-1.no_search_link

[10] 使用 BG/NBD 模型与 Gamma-Gamma 模型预测客户的生命周期价值 CLV/LTV_maiyida123 的博客-CSDN 博客. (2021). Retrieved 22 November 2021, from https://blog.csdn.net/maiyida123/article/details/119832582?spm=1001.2101.3001.6650.7&utm_medium=distribute.pc_relevant.none-task-blog-2~default~BlogCommendFromBaidu~default-7.no_search_link&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2~default~BlogCommendFromBaidu~default-7.no_search_link

[11] 如何预测（计算）用户价值—BG/NBD 模型_afk110 的博客-CSDN 博客. (2021). Retrieved 22 November 2021, from https://blog.csdn.net/afk110/article/details/110931341

[12] 用户存续期价值评估 CLV(三) Gamma-Gamma 模型 Python 模拟_Magic Ktwc37 的博客-CSDN 博客_gamma 模型. (2021). Retrieved 22 November 2021, from https://blog.csdn.net/weixin_43171270/article/details/106267086?spm=1001.2014.3001.5502

七、附录

```
#导入包
# installlation required
!pip install Lifetimes #python 自带计算 cltv
!pip install plotly

# libraries
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from lifetimes import BetaGeoFitter #Basic Frequency/Recency analysis using the BG/NBD model
from lifetimes import GammaGammaFitter #Gamma-Gamma submodel and predict the conditional, expected average lifetime value of our customers.
from lifetimes.utils import summary_data_from_transaction_data
from lifetimes.plotting import plot_period_transactions
from sklearn.preprocessing import MinMaxScaler#Transform features by scaling each feature to a given range.
import warnings
warnings.filterwarnings("ignore")
import plotly.express as px
import numpy as np

import datetime
import plotly.graph_objects as go

from lifetimes.plotting import plot_frequency_recency_matrix#可视化频率矩阵，上次购买的时刻和频率（重复交易的数量）
from lifetimes.plotting import plot_probability_alive_matrix#客户是否还活着
from lifetimes.plotting import plot_period_transactions#模型评估效果检查
from lifetimes.utils import calibration_and_holdout_data#交叉验证
from lifetimes.plotting import plot_calibration_purchases_vs_holdout_purchases#绘制模型交叉验证图
```

```
#加载和检查数据
df = pd.read_csv('/content/drive/MyDrive/Online Retail II Data Set from ML Repository/Year 2010-2011.csv',encoding='unicode_escape')
df.head()

#数据预处理
##非法值处理
#去除含有C的表示中止操作的发票 和为空的发票(字符串的模糊筛选)
data= df[~df['Invoice'].str.contains('C',na=False)]
#买的量和价钱不能为小于0 所以去除掉
print((data.Quantity < 0).value_counts())
print((data.Price < 0).value_counts())
data.query('Quantity > 0 & Price > 0',inplace=True)

##缺失值处理
data.isnull().sum()
data.dropna(inplace=True)#不返回东西，只修改data里的值
data.isnull().sum()
data.describe()
#这里我们看到ID也被当成数值型数据了 后面要进行处理一下

##异常值处理
#观察两个数值型变量的列是否有异常值
fig = px.box(y=data['Quantity'], notched=True)
fig.show()
fig = px.box(y=data['Price'], notched=True)
fig.show()
#对异常值进行修改 超过上界or下界就直接把数值归为上界or下界
def two_bound(df,var):
  res = np.percentile(df[var],(1,99),interpolation='midpoint')
  #上四分位数
  quartile1 = res[0]
  #下四分位数
  quartile3 = res[1]
```

```
  interquantile_range = quartile3 - quartile1
  up_limit = quartile3 + 1.5 * interquantile_range
  low_limit = quartile1 - 1.5 * interquantile_range
  return low_limit, up_limit

def replace_bound(df, var):
    low_limit, up_limit = two_bound(df, var)
    df.loc[(df[var] < low_limit), var] = low_limit
    df.loc[(df[var] > up_limit), var] = up_limit

replace_bound(data,'Quantity')
replace_bound(data,'Price')

##类别转换
#把 Customer ID 转换为类别型变量
print(data['Customer ID'].dtypes)
data['Customer ID']=data['Customer ID'].astype(int)
data['Customer ID']=data['Customer ID'].astype(str)
print(data['Customer ID'].dtypes)
data = data.reset_index()
del data['index']
#把时间的那个字段转化成时间戳的形式
datatime = []
for i in range(0,len(list(data['InvoiceDate']))):
  temp = data['InvoiceDate'][i]
  dateTime_p = datetime.datetime.strptime(temp,'%m/%d/%Y %H:%M')
  datatime.append(dateTime_p)

data['InvoiceDate']=datatime

#探索性数据分析
##类别变量
cat_cols = [col for col in data.columns if data[col].dtypes =="O"]
cat_cols
#先算出现的频数
```

```
for col_name in cat_cols:
  print(pd.DataFrame({col_name:data[col_name].value_counts(),
            '比率':100 * data[col_name].value_counts()/len(data)}))
fig, ax = plt.subplots(figsize=(15,8))
sns.countplot(x=data['Country'], data=data)
plt.xticks(rotation = 45, ha = 'right')
plt.title('Country Distribution')
plt.show()
#每个描述对应的销售量 也就是可以理解为一种类型物品的销售量
sale_product = data.groupby('Description').agg({'Quantity':'sum'})
sale_product.reset_index(inplace=True)
sale_product
#销售订单数 一种类型物品的销售量
sale_num = data.groupby('Description').agg({'Quantity':'count'})
sale_num.reset_index(inplace=True)
sale_num
#销售量 top 20
top20 = sale_product.sort_values(by='Quantity',ascending = False).head(20)
fig   =   go.Figure(data=[go.Bar(x=top20['Description'],   y=top20['Quantity'],
text=top20['Quantity'],          textposition='auto',marker_color=['aliceblue',
'antiquewhite', 'aqua', 'aquamarine', 'azure',
            'beige', 'bisque', 'black', 'blanchedalmond', 'blue',
            'blueviolet', 'brown', 'burlywood', 'cadetblue',
            'chartreuse', 'chocolate', 'coral', 'cornflowerblue',
            'cornsilk', 'crimson'])])
fig.show()

##数值变量
num_cols = [col for col in data.columns if data[col].dtypes != 'O']
num_cols
#Date 这个部分就不做另外分析了 很明显可以看出是从 2010-12 到 2011-12
for col_name in num_cols:
  if col_name != 'InvoiceDate':
    print(data[col_name].describe())
    data[col_name].hist(bins=20)
```

```
    plt.xlabel(col_name)
    plt.title(col_name)
    plt.show()
#每笔发票的总金额
data["TotalPrice"] = data["Price"] * data["Quantity"]

#数据准备
###recency：客户上一次购买和第一次购买的差异
###T：客户的时间(购买年龄)
###frequency：重复购买的总次数
###monetary_value：每次购买的平均收益
cltv_df = summary_data_from_transaction_data(data,'Customer ID','InvoiceDate',
                        monetary_value_col='TotalPrice',
                        observation_period_end='2011-12-9',
                        freq='D')
cltv_df.head()
temp_list=list(set(list(cltv_df['frequency'])))
temp_count=[0]*len(temp_list)
for i in range(0,len(list(cltv_df['frequency']))):
  index=temp_list.index(list(cltv_df['frequency'])[i])
  temp_count[index]+=1
#频率分布
fig                                                                                =
go.Figure(data=[go.Bar(x=temp_list,y=temp_count,text=temp_count,textposition='a
uto',marker_color='pink')])
fig.show()

#BG-NBD 模型
#penalizer_coef 通过 L2 范数来控制参数的大小
bgf = BetaGeoFitter(penalizer_coef=0.000001)
bgf.fit(cltv_df['frequency'], cltv_df['recency'], cltv_df['T'])
print(bgf)
bgf.summary
#可视化频率矩阵
plot_frequency_recency_matrix(bgf)
```

```
#仍然活着的概率
plot_probability_alive_matrix(bgf)
#评估模型的拟合效果 1.首先是将数据与模拟的人工数据与安装模型的参数进行比较。
plot_period_transactions(bgf)
print(min(data['InvoiceDate']))
print(max(data['InvoiceDate']))
#交叉验证
#划分验证期
val_data = calibration_and_holdout_data(data,'Customer ID','InvoiceDate',
                        calibration_period_end='2011-06-01',
                        observation_period_end='2011-12-09')
val_data.head()
bgf.fit(val_data['frequency_cal'],val_data['recency_cal'],val_data['T_cal'])
plot_calibration_purchases_vs_holdout_purchases(bgf,val_data)

##预测
#预测时间 客户未来 t 天的购买次数
t=90
cltv_df["expected"]                                                    =
bgf.predict(t,cltv_df['frequency'],cltv_df['recency'],cltv_df['T'])
cltv_df.sort_values("expected", ascending=False).head(10)

#Gamma-Gamma Model
cltv_df[['monetary_value', 'frequency']].corr()
returing_id = cltv_df[cltv_df['frequency'] > 0]
ggf = GammaGammaFitter(penalizer_coef = 0.01)
ggf.fit(returing_id['frequency'],returing_id['monetary_value'])
print(ggf)
#估算每个客户的平均交易价值(平均利润的条件期望)
returing_id["expected_average_profit"]                                 =
ggf.conditional_expected_average_profit(returing_id['frequency'],returing_id['m
onetary_value'])
returing_id.sort_values("expected_average_profit", ascending=False).head(20)

#Customer lifetime value
```

```
cltv = ggf.customer_lifetime_value(bgf,#the model to use to predict the number
of future transactions
                    cltv_df['frequency'],
                    cltv_df['recency'],
                    cltv_df['T'],
                    cltv_df['monetary_value'],
                    freq='D')
#默认算客户预期寿命为 12 个月的，freq 是 T 的计算单位，贴现率 0.01
#重设索引
cltv = cltv.reset_index()
cltv_final = cltv_df.merge(cltv,on='Customer ID',how='left')
cltv_final.sort_values(by='clv',ascending=False).head(10)
#1 Month CLTV:
#discount_rate 月度调整贴现率
cltv_1 = ggf.customer_lifetime_value(bgf,
                      cltv_df['frequency'],
                      cltv_df['recency'],
                      cltv_df['T'],
                      cltv_df['monetary_value'],
                      time=1,
                      freq='D',
                      discount_rate=0.01)
cltv_1.head()
cltv_1=cltv_1.reset_index()
cltv_1=cltv_df.merge(cltv_1,on='Customer ID',how='left')
cltv_1.sort_values(by='clv',ascending=False).head(10)

#CLTV 预测的细分
#对 clv 的值进行 0-1 标准化
scaler = MinMaxScaler(feature_range=(0, 1))
scaler.fit(cltv_final[["clv"]])
cltv_final["scaled_clv"] = scaler.transform(cltv_final[["clv"]])

cltv_final.sort_values(by="scaled_clv", ascending=False).head()
value=list(cltv_final['scaled_clv'])
```

```
segment=[]
for i in value:
  if i>=0 and i<0.3:
    segment.append('低价值')
  elif i>=0.3 and i<0.6:
    segment.append('较低价值')
  elif i>=0.6 and i<0.8:
    segment.append('较高价值')
  elif i>=0.8:
    segment.append('高价值')
  else:
    segment.append('nan')

cltv_final['segment']=segment
#进行排序
cltv_final=cltv_final.sort_values(by="scaled_clv", ascending=False)
cltv_final.head()
```