

# 电视广告分户推送与营运规划

## ——2019 同济大学数学建模比赛暨深圳杯 D 题

### 摘要

电视台通过向广告主销售广告时段获取销售利益. 因此, 卖方确定的合理广告出售底价以及买方确定的合理广告费支出对于买卖双方利益尤为重要. 除此以外, 实现广告与用户的匹配推送对于提升买方收益以及卖方业务利润均有显著作用. 本文着重对这一问题进行深入的分析以及建模.

对于问题一, 我们选择用户年龄段, 用户性别, 用户教育水平, 用户所在行业, 用户消费水平以及用户观看广告的投放时间等作为输入层的标签, 将广告中产品所属类型 (即广告类型), 广告中产品价格, 作为输出层的标签, 结合 BP 神经网络, 搭建多标签模型, 通过训练数据, 得出静态情况下的“广告-用户”分类匹配推送模型, 即输入一个用户所拥有的标签, 得出相对应的广告类型, 实现推送. 模型的准确率, 损失, 混淆矩阵等指标都有着良好的表现.

对于问题二, 我们结合实际选取与广告价格有关的数据量, 即节目类型, 广告播放时段, 直播关注度, 市占率, 时间等. 运用多元线性回归的方法, 确定分时段广告价格为被解释变量, 上述提到的变量为解释变量. 我们决定将回归过程分解为三阶段, 即根据节目类型以及时段预测直播关注度, 根据直播关注度以及节目类型预测市占率, 根据市占率, 节目类型, 广告位置以及时段预测解最终的合理估算底价. 经实证检验, 我们的回归模型显著. 对底价的预测能力较好.

对于问题三, 在静态模型的基础上, 将有时序性的历史收视记录作为输入, 搭建一维新的卷积神经网络, Flatten 之后, 将输出与第一问中的神经网络静态模型的输出进行融合, 搭建一个多输入单输出的神经网络并通过训练数据, 得出动态情况下的“产业趋势-用户行为”分类匹配推送更新模型, 即输入一个用户所拥有的标签, 即过去 24 小时的收视行为, 可以得出各广告产业的产品销售情况, 并可根据用户的后续行为变化更新产品推送情况及销售情况.

对于问题四, 我们首先建立静态的拍卖模型, 我们基于广告费用, 竞争对手数, 最近一个生产周期的销量, 构造买方的收入函数, 将求解合理广告费的问题转化为求解极大值点的问题. 当投入广告带来的效益大于不投入的效益与机会成本之和时, 该厂商参与竞拍. 由拍卖的性质, 最后成交价格为各厂商合理广告费的最大值. 然后建立动态的模型, 引入时间变量, 用  $t-1$  时的销量等参数推断  $t$  时的参与广告拍卖的厂家合理的广告投入, 形成一个不完全信息动态博弈过程. 将原本复杂的竞价交易模型简化为通俗易懂的数学模型. 实证表明, 我们的模型能客观的反映出优胜劣汰的交易市场规律.

**关键字:** BP 神经网络, 供求关系, 多元线性回归, 重复博弈, 不完全信息动态博弈

# 目录

一、 问题重述 .....	4
二、 建模分析 .....	4
三、 模型的假设 .....	5
四、 变量及符号说明 .....	6
五、 模型建立与求解 .....	7
5.1 问题一 .....	7
5.1.1 广告——用户分类匹配推送静态分类模型 .....	7
5.2 问题二 .....	8
5.2.1 静态合理底价估算模型 .....	8
5.3 问题三 .....	9
5.4 问题四 .....	10
5.4.1 买方利益最大化的广告出价静态模型 .....	11
5.4.2 卖方利益最大化的静态模型 .....	12
5.4.3 动态竞价交易模型 .....	12
5.5 问题五 .....	13
5.5.1 第一问算例 .....	13
5.5.2 问题二算例 .....	13
5.5.3 第三问算例 .....	17
5.5.4 第四问算例 .....	18
六、 模型检验 .....	20
七、 模型评价 .....	21
7.1 优点 .....	21
7.2 缺点 .....	21
附录 A 第一问代码 .....	23
附录 B 问题一完整结果 .....	28
附录 C 问题二实证分析 Eviews 多元线性回归报告 .....	29
附录 D 第三问代码 .....	30
附录 E 第三问完整测试数据 .....	33

附录 F 第四问 Matlab 代码 .....	33
--------------------------	----

## 一、问题重述

我们需要解决的问题如下:

1. 通过对于视频广告特征的分析, 选取合适的特征变量, 同时选取对应电视台用户画像当中适当的特征值, 建立数学模型, 研究二者之间的关系, 实现静态下的数据匹配.
2. 通过对分时段底价的影响因素分析, 选择合理的影响因素, 建立数学模型, 来估算卖方分时段竞卖的底价.
3. 研究电视频道用户的收视情况以及在播广告的产品销售情况, 基于问题一当中的静态模型, 实现电视台视频广告及用户的动态匹配推送模型, 实现广告推送业务的更新化, 与此同时, 结合模型, 预测出产品未来销售情况变化.
4. 在第三题的模型的基础上, 建立数学模型, 模拟竞价交易的过程, 同时实现卖方收益的最大化, 并尽可能提升收视率及买方的产品销售量以及利润.
5. 设计求解算法, 同时结合现有数据, 验证问题一至四中的模型.

## 二、建模分析

对于问题一, 我们通过对广告以及用户特征向量的筛选, 选择用户年龄段, 用户性别, 用户教育水平, 用户所在行业, 用户消费水平以及用户观看广告的投放时间等作为输入层的标签, 将广告中产品所属类型 (即广告类型), 广告中产品价格, 作为输出层的标签, 结合神经网络, 搭建多标签模型, 通过训练数据, 得出静态情况下的“广告-用户”分类匹配推送模型, 即输入一个用户所拥有的标签, 得出相对应的广告类型, 实现推送。

对于问题二, 我们结合实际选取与广告价格有关的数据量, 即节目类型, 广告播放时段, 直播关注度, 市占率, 时间等。运用多元线性回归的方法, 确定分时段广告价格为被解释变量, 上述提到的变量为解释变量。将预处理过的数据输入 Eviews, 测算变量之间相关性。经过 T 检验, 决定将回归过程分解为三阶段, 即根据节目类型及时段预测直播关注度, 根据直播关注度以及节目类型预测市占率, 根据市占率, 节目类型, 广告位置及时段预测解最终的合理估算底价。

对于问题三, 在静态模型的基础上, 将有时序性的历史收视记录作为输入, 搭建一维新的卷积神经网络, Flatten 之后, 将输出与第一问中的神经网络静态模型的输出进行融合, 搭建一个多输入单输出的神经网络并通过训练数据, 得出动态情况下的“产业趋势-用户行为”分类匹配推送更新模型, 即输入一个用户所拥有的标签, 即过去 24 小时的收视行为, 可以得出各广告产业的产品销售情况, 并可根据用户的后续行为变化更新产品推送情况及销售情况。

对于问题四, 我们首先建立静态的拍卖模型, 我们先基于广告费用, 竞争对手数,

最近一个生产周期的销量,构造买方的收入函数,将求解合理广告费的问题转化为求解极大化问题。当投入广告带来的效益大于不投入的效益与机会成本之和时,该厂商参与竞拍。由拍卖的性质,最后成交价格为各厂商合理广告费的最大值。然后建立动态的模型,引入时间变量,用  $t-1$  时的销量等参数推断  $t$  时的各个参数,形成一个不完全信息动态博弈过程,由博弈论理论知 [1], 这个过程的均衡一定存在。

对于问题五,我们对部分题目给出了算例。

### 三、模型的假设

- 各频道在一段时间内的节目安排不变,变化的只是它们之间的广告;
- 市场出清,即商品价格具有充分的灵活性,能使供给和需求迅速达到均衡.
- 所有人都是经济人,具有完全理性.
- 市场是完全竞争市场,厂商之间信息是不完全的.
- 各厂商的销售函数  $sell^i$ 、收入函数  $f^i$ 、机会成本函数  $\pi^i$  在一段时间内形式不变.
- 市场是有限的,因此收益函数取值不可能为  $+\infty$ .

#### 四、 变量及符号说明

符号	意义
$D_{1i}$	虚拟变量, 用于标记不同的节目类型
$D_{2i}$	虚拟变量, 用于标记广告播放时段
$T$	节目开始时间的相对值
$focus$	直播关注度
$rate$	市占率
$A_t^i$	时刻 $t$ 第 $i$ 个参与拍卖的厂商
$n_t$	时刻 $t$ 某电视台投放广告的同一行业厂商数
$N_t^*$	时刻 $t$ 市场上某行业总厂商数
$N_t$	时刻 $t$ 参与竞拍的厂商总数
$sell^i$	厂商 $A^i$ 的销售函数
$VC^i$	厂商 $A^i$ 的可变成本
$FC^i$	厂商 $A^i$ 的不变成本
$P_t^i$	厂商 $A^i$ 为使利益最大化而投入的广告费
$P_t^0$	时刻 $t$ 卖方的底价
$I(A)$	示性函数, $A$ 为 <b>true</b> 是 $I(A) = 1$ 否则为 0

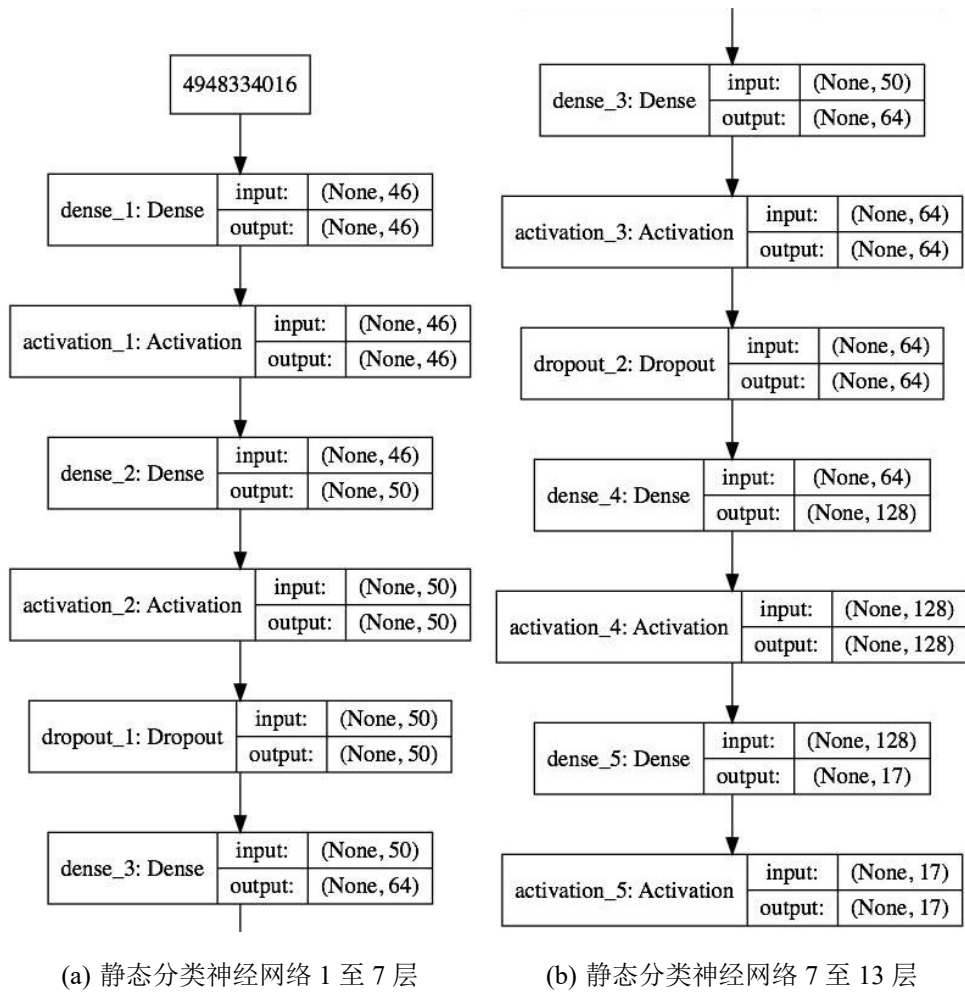
## 五、模型建立与求解

### 5.1 问题一

#### 5.1.1 广告——用户分类匹配推送静态分类模型

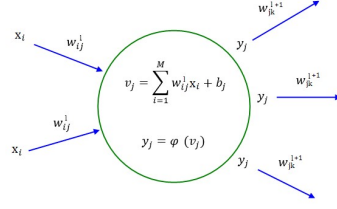
我们通过对广告特征以及用户画像的分析，分别选取出针对于广告特征以及用户画像的特征变量，确定广告方  $A^1, A^2, \dots$  作为输出层以及广告方  $A^i$  广告类型  $D_{adv}^i$  以及广告产品价格  $D_p^i$  作为输出层的标签，确定用户方  $u^1, u^2, \dots$  作为输入层以及用户  $u^i$  的年龄段  $D_{age}^i$ ，用户性别  $D_{sex}^i$ ，用户教育水平  $D_{edu}^i$ ，用户所在行业  $D_{job}^i$ ，用户消费水平  $D_{consume}^i$  以及用户观看广告的投放时间的相对值  $t$  作为输入层的标签。

首先, 用 python 的 selenium(浏览器框架测试库), 我们从酷云平台上爬取了含有用户画像和广告内容匹配信息的数据。并对其进行了数据清洗, 得到上述的特征. 进行了特征工程之后, 运用以 tensorflow 的 keras 深度学习库, 搭建了一个 BP 神经网络, 其层与层之间的抽象关系如下图所示:



有一个输入层 dense1, 一个输出层 activation\_5, 以及 10 个隐藏层. 其中的 activation 层, 起到激活函数的作用, 在神经网络中进行非线性变换. 其中的 dense 层, 与上一层所有

的 cell 相连, 对前一层的特征求加权和, 起到分类器的作用. 每一组 dense 层和 activation 层神经元的功能如下图所示:



从左向右看,  $x_i$  表示由上一个神经元传入的数值. 权重  $\omega$  中,  $i$  表示前一层神经元的序号,  $j$  表示指向的神经元的序号, 上标  $l$  为层次. 偏置  $b_j$  是属于这个神经元的,  $j$  表示这个神经元的序号. 经加权求和后得到  $v_j$ , 其中的  $M$  为前一层神经元总数. 再将结果代入激活函数  $\varphi$  中, 得到  $y_j$  作为此神经元的输出.

具体而言, 输入层函数为

$$v_j = \sum_{i=1}^M \omega_{ij} x_i + b_j$$

隐藏层所采用的激活函数为 relu 函数

$$\varphi(x) = \max\{0, x\}$$

输出层函数为

$$y_j = \varphi(v_j)$$

## 5.2 问题二

### 5.2.1 静态合理底价估算模型

对于问题二, 设某一频道的节目类型有  $n + 1$  种, 对应  $n$  个虚拟变量  $D_{1i}$ ,  $i = 1, 2, \dots, n$ , 对应第  $j$  种节目类型, 虚拟变量的取值为  $D_{1i} = \delta_{ij}$ ,  $\delta_{ij}$  为 Kronecker 符号. 广告播放时段分为节目前, 节目中插播, 节目结束后, 对应虚拟变量  $D_{21}$ ,  $D_{22}$ , 取之方法与上述同理. 设节目开始时间的相对值为  $T$ ,  $T \in [0, 1)$ , 比如晚上 7 点开始播放的新闻联播, 其  $T = \frac{19}{24}$ .

于是, 我们可以定义合理底价估算函数  $p(D_{11}, D_{12}, \dots, D_{1n}, D_{21}, D_{22}, T)$ , 这个函数可分解为以下函数:

- 直播关注度 *focus* 函数: 由于直播关注度和广告的位置无关, 故自变量选取应该去掉  $D_{21}$ ,  $D_{22}$ ; 得到如下函数形式:

$$focus = focus(D_{11}, D_{12}, \dots, D_{1n}, T)$$



- 市占率  $rate$  函数: 同样的市占率和广告的位置无关, 故自变量选取应该去掉  $D_{21}$ ,  $D_{22}$ ; 得到如下函数形式:

$$rate = rate(focus, D_{11}, D_{12}, \dots, D_{1n}, T)$$

- 合理底价估算函数:

$$p = p(rate, D_{11}, D_{12}, \dots, D_{1n}, D_{21}, D_{22}, T)$$

我们用多元线性回归模型来给出这三个函数的估计形式:

$$\begin{aligned} focus &= C + \sum_{i=1}^n \alpha_{1i} D_{1i} (\text{虚拟变量影响截距项}) \\ &\quad + f(T) + \sum_{i=1}^n \alpha_{2i} D_{1i} f(T) (\text{虚拟变量影响斜率}) \\ &\quad + \mu (\text{随机扰动项, 假设服从 } N(0, \sigma_1^2)) \\ rate &= C + \beta_1 focus + \sum_{i=1}^n \beta_{1i} D_{1i} + \sum_{i=1}^n \beta_{2i} D_{1i} focus + f(T) + \mu \\ p &= C + \gamma_1 rate + \sum_{i=1}^n \gamma_{1i} D_{1i} + \sum_{i=1}^n \gamma_{2i} D_{1i} rate + f(T) \\ &\quad + \gamma_{31} D_{21} + \gamma_{32} D_{22} + \gamma_{41} D_{21} rate + \gamma_{42} D_{22} rate + \mu \end{aligned}$$

这里  $f(T)$  需要我们根据具体的数据, 给出  $focus$  和  $T$  的合理关系, 比如线性. 这些函数的具体算例我们将在问题五中给出.

### 5.3 问题三

我们将从酷云下载的数据进行了特征工程, 将其中的品牌归为 17 个大类:

个人用品, 互联网, 交通, 农业, 化妆品/浴室用品, 商业/服务性行业, 娱乐及休闲, 家居用品, 家用电器, 房地产, 数码/电脑/办公用品, 清洁用品, 药品, 衣着, 酒类, 金融/投资行业, 食品及饮料.

并且, 将其销量情况用 0-16 的整形数据来抽象表示. 我们对第三问的数据及问题进行特征分析, 发现了我们所需的标签在单位时间的数据片段中的位置不具有高度相关性时, 1D CNN 是非常有效的, 因此我们为了挖掘出时序性的用户行为这一特征, 在第一问训练出的静态模型的基础上, 将有时序性的历史收视记录作为输入, 另搭建了新的一套 1D 卷积神经网络. 在数据预处理上, 我们拿到真实数据以后, 将广告所对应的行业归类到大的行业, 如食品, 将用户的所有的标签转化为虚拟变量  $dummy_i$ , 将第一问的所有变量加上了指标  $t$ , 设  $n_1, n_2, \dots, n_k$  为用户浏览的广告 [2], 按照时间顺序排列, 希望生成广告集合

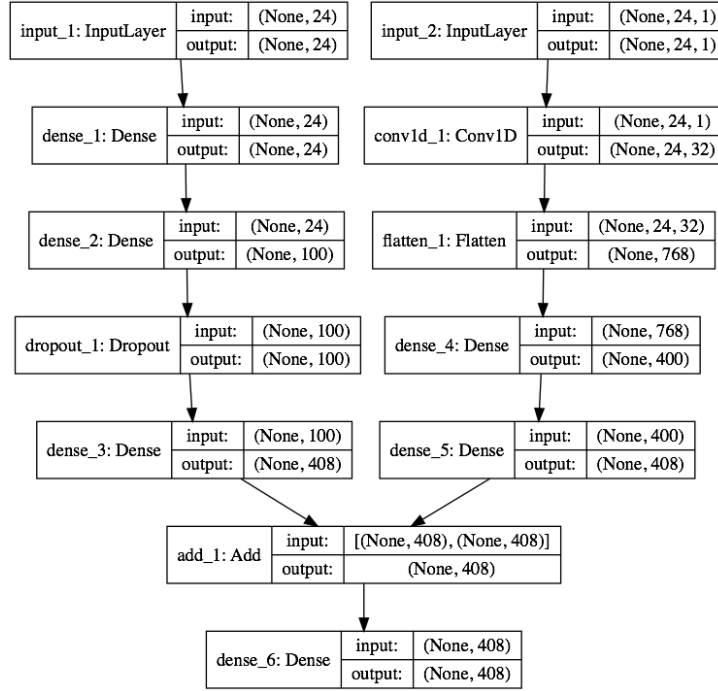
$$\mathcal{S} = \{n_{k_1}, n_{k_2}, \dots, n_{k_t}\}$$

s.t. 用户的偏好  $f$  满足

$$f(u, n_{k_i}) > f(u, n_{k_j}), \quad \text{if } i < j$$

$f$  用于计算某用户  $u$  对广告的评价. 然后我们将广告按照  $\mathcal{S}$  中给出的顺序推送给用户, 在推送的过程中, 又会继续收到用户的反馈, 我们可以继续调整  $\mathcal{S}$  中的广告元素.

然后搭建的网络如下:



图中的 **dropout** 可以防止模型的过拟合问题, **Flatten** 将其输出与第一问中的神经网络静态模型的输出进行融合, 搭建一个多输入单输出的神经网络并通过训练数据, 得出动态模型. 我们将用户一天的行为分为 24 段, 并在这个序列上使用 32 个长度为 3 的 1D 卷积核进行卷积运算,

$$y_i = \varphi\left(\sum_{n=1}^3 x_{i+n}\omega_n + \omega_b\right)$$

以获得时序性的用户行为特征. 最后将此 CNN 和 DFF 进行双模型融合, 后接全连接层 **dense\_6** 作为输出构成最终的模型, 即输入  $t - \Delta t$  时刻一个用户所拥有的标签, 这里即过去 24 小时的收视行为, 可以预测  $t$  时刻各广告产业的产品销售情况对广告投放做出更新, 并可根据用户的在  $t$  时刻的标签变化得出  $t$  时刻产品推送情况及销售情况的真实值。

## 5.4 问题四

由模型假设, 我们通过拍卖 [1], 使得电视台 (卖方) 与出价最高的销售商 (买方) 成交, 设成交价格为  $P$ , 有  $N$  个厂商参与本次拍卖, 记为  $A^1, A^2, \dots, A^N$ . 每个厂商心中的

底价为  $P^1, P^2, \dots, P^N$ , 则  $P = \max_{i=1,2,\dots,N} P^i$ .

#### 5.4.1 买方利益最大化的广告出价静态模型

本模型旨在给出在某一个时间点上, 什么样的条件下买方才会参与竞拍并且出多少的广告费以实现自身的利益最大化, 对此, 我们先给出买方的收入函数定义:

**定义 1 (买方的收入函数)** 设  $f(p, n, sell^i(p, N^*))$  为买方的收入函数,  $p$  代表广告费用,  $n$  代表在该时刻该电视台投放广告的同时行业厂商数,  $sell^i(p, N^*)$  代表最近的生产周期厂商的销售量, 它与广告投入及当前的从业行业数有关.

由其定义及微观经济学知识 [3] 我们不难知道  $f$  应具有下列性质:

- $f(0, n, sell^i(0))$  代表不投入广告带来的收入,  $f(0, n, sell^i(0))$  及  $sell^i(0, N^*)$  均应为正.
- $f$  与投入的广告费  $p$  正相关, 但是由于边际效益递减, 故每增加一单位的广告投入带来的收入增加越来越少. 即

$$\frac{\partial f}{\partial p} > 0 \quad \frac{\partial^2 f}{\partial p^2} < 0$$

- 由于市场是有限的,  $f$  在  $\mathbb{R}^+ \times \mathbb{N} \times \mathbb{R}^+$  上有上确界,  $sell^i(p, N^*)$  在  $\mathbb{R}^+$  上有上确界.
- $f$  与  $n$  负相关, 因为  $n$  增加, 则有更多的人抢占市场, 广告竞争更加的激烈, 收益变小, 即  $f$  变小. 反之亦然. 在本模型的讨论中,  $n$  外生给定. [4]
- $\sum_{i=1}^{N^*} sell^i(p^i) = D$ , 即一段时间内需求  $D$  不变, 也就是供给不变, 因此  $sell^i$  可以被理解为一种市场份额. 厂商  $A^i$  只能用上一阶段的广告费去估算自己的市场份额, 即销售量.

于是, 买方  $A^i$  在投入  $p$  元广告费的前提下, 其收益为:

$$F^i(p, n, sell^i(p, N^*)) = f^i(p, n, sell^i(p, N^*)) - p - VC^i(sell^i(p, N^*)) - FC^i$$

这里  $FC^i$  为其不变成本,  $VC^i(sell^i(p, N^*))$  为其可变成本, 是销量的函数.

为使利润最大化, 我们有  $\frac{\partial F^i(p, n, sell^i(p, N^*))}{\partial p} = 0$ , 即

$$f_1^i(p, n, sell^i(p, N^*)) + f_3^i(p, n, sell^i(p, N^*)) \frac{dsell^i}{dp} - 1 - \frac{dVC^i(sell^i(p, N^*))}{dsell^i} \frac{dsell^i(p, N^*)}{dp} = 0$$

**定理 1**  $\varphi^i(p) = F^i(p, n, sell^i(p, N^*))$  在  $[0, \infty)$  上必有最大值.

**证明 1** 由模型的假设, 当  $p$  很大的时候, 由于边际效益递减规律及市场的有限性,  $f$  有上确界, 故  $\exists p_0, s.t. \forall p > p_0, f^i(p, n, sell^i(p, N^*)) < p_0$ , 此时  $\phi(p) < -VC^i(sell^i(p, N^*)) - FC^i < 0$ . 由  $\varphi^i(p)$  在  $[0, p_0]$  的连续性,  $\varphi^i(p)$  在  $[0, p_0]$  上一定有最大值.

另一方面, 由于  $\frac{d\varphi^i(0)}{dp} > 0$ , 故  $\varphi^i$  的极大值不可能在  $p = 0$  时取得; 由上述分析,  $\varphi^i$  不可能在  $p = 0$  时取得; 故  $\exists P^i, s.t. \forall p \in [0, p_0], \varphi^i(P^i) > \varphi^i(p), \forall p > p_0, \varphi^i(P^i) > 0 > \varphi^i(p)$ . 故当  $p = P^i$  时,  $\varphi^i$  取得极大值, *Q.E.D.*

于是,为了使买方利益最大化,买方  $A^i$  在广告上的投入必为  $P^i$ ,作为买方,对于广告的效果是有所预期的.这里,我们假设为了投入  $P^i$  的广告费,而放弃的机会成本为  $\pi^i(p)$ .故若  $A^i$  参与竞拍,则必有

$$\varphi^i(P^i) > \varphi^i(0) + \pi^i(P^i)$$

该条件满足时,买方  $A^i$  才会参与竞拍.

#### 5.4.2 卖方利益最大化的静态模型

作为卖方,我们可以根据问题 5.2 的结果,根据某节目的类型,节目的播放时间,欲出售的广告所在位置(该节目播放前、中、后),来确定一个拍卖底价  $P^0$ . 拍卖结束后,与买方  $A^i$  以  $P^i$  的价格成交.

**定理 2** 卖方实现了利益最大化.

**证明 2** 若不然,则存在一个出价更高的买家  $A^j, j \neq i$ . 根据拍卖规则,卖方应与  $A^j$  成交,矛盾. *Q.E.D.*

#### 5.4.3 动态竞价交易模型

我们现在基于 5.4.1 及 5.4.2 提出的静态模型建立动态模型. 我们考虑从第  $t-1$  时刻到第  $t$  时刻的状态转移.

在  $t-1$  时刻,我们知道卖方的底价  $P_{t-1}^0$ ,某一行业在该电视台有  $n_{t-1}$  个广告在播,该行业的厂商总数为  $N_{t-1}^*$ ,参与竞标的厂商总数为  $N_{t-1}$ ,厂商  $A^i$  在最近一个生产周期的广告投入  $P_{t-1}^i$ .

则在  $t$  时刻,厂商预测在  $t$  时刻会有  $n_t$  个厂商的广告被播出,将这种预测作为真实值的猜测. 根据模型 5.4.1,可以计算出厂商  $A^i$  愿意投入的广告费  $P_t^i$ ; 根据模型 5.4.2,卖方的底价为  $P_t^0$ ; 设此时该行业的厂商为  $N_t^*$ ; 此时参与竞标的厂商数为

$$N_t = \sum_{i=1}^{N_t^*} I[F^i(P_t^i, n_{t-1}, \text{sell}^i(P_t^i, N_{t-1}^*)) > F^i(0, n_{t-1}, \text{sell}^i(P_t^i, N_{t-1}^*)) + \pi^i(P_t^i)]$$

这里  $I$  为示性函数,里面的表达式为真取值为 1, 否则为 0.

根据博弈论中不完全信息动态博弈 [1] 的结果,其他的厂商知道  $A_i$  会用上述方法计算自己的广告投入,于是其他的厂商在算  $A^i$  的市场份额时,将不会用  $P_{t-1}^i$ ,而会用通过上述算法得到的  $\hat{P}^i$ . 另一方面厂商  $A^i$  猜测其他厂商会这么做,于是  $A^i$  也会调整自己的价格,这样的过程一直下去,所有的厂商必然会收敛到一个均衡价格 [1], 我们这里继续沿用上面的符号  $P_t^i$ .

将这些厂商重新编号为  $A_t^i$ , 此时的成交价格为

$$\max_{i=1,2,\dots,N_t} P_t^i > P_t^0$$

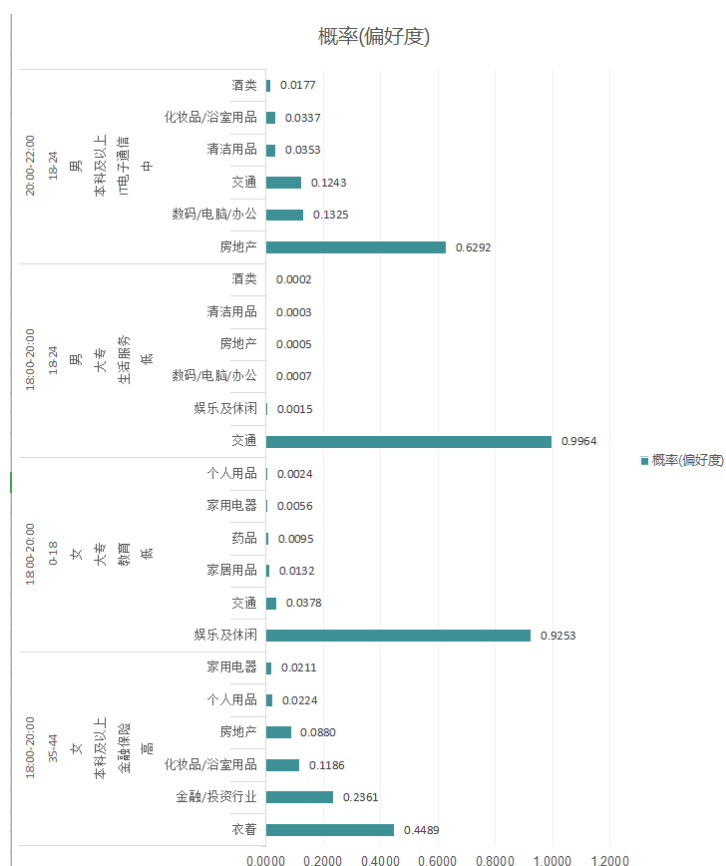
在所有的竞价均完成后, 我们可以得到某一行业在该电视台有  $n_t$  个广告在播, 这是真实值. 这样我们就完成了两个时间层的状态转移. 具体的博弈过程算例在问题五中给出.

## 5.5 问题五

我们将会结合数据给出算例:

### 5.5.1 第一问算例

我们结合模型, 进行了 10 组测试, 给出以下算例. 我们选取其中 4 个展示, 剩余的见附录.



输入为一个六维的特征向量, 输出是经过 `softmax` 函数后的匹配概率最大的六个广告类型, 即匹配概率代表用户对此广告类型的偏好度. 偏好度越高, 广告主应对其选择推送.

### 5.5.2 问题二算例

对于问题二的实证分析, 我们基于多元线性回归 [5], 以 CCTV1(综合类频道的代表), CCTV12(法制频道, 专题类代表), 湖南卫视(地方卫视的代表) 为例来讨论.

**1. CCTV1 合理底价估算模型** 对于 CCTV1, 我们将其节目类型分为法制专题, 电视剧, 综艺, 新闻四种类型, 对应虚拟变量  $D_{11}, D_{12}$ , 将广告播放时段分为节目 (播出) 前, 节目中插播, 节目 (结束) 后, 对应虚拟变量  $D_{21}, D_{22}$ , 对应关系如下:

节目类型	$D_{11}$	$D_{12}$	$D_{13}$	插播时段	$D_{21}$	$D_{22}$
专题	1	0	0	节目前	1	0
电视剧	0	1	0	节目中	0	1
综艺	0	0	0	节目后	0	0
新闻	0	0	0			

利用我们已知的数据, 我们希望预测出合理的底价. 首先我们希望能根据节目的类型以及其播放时段、节目播出时间相对值  $t$ , 来预测其直播关注度  $focus$  建立如下的多元线性回归模型:

$$FOCUS = \beta_1 D_{12} |T - 0.5| + \beta_2 D_{12} |T - 0.8| + \beta_3 |T - 0.5| + \beta_4 D_{13} T + \beta_0$$

这里, 考虑到中午 12 点及晚上 19 点是两个收视高峰, 为提高模型的准确性, 我们分别用  $t - 0.5, |t - 0.8|$  来衡量一天中播出时间与中午 12 点及晚上 19 点之间的差距, 差距越大说明在同一天内, 播出时间分别离 12 点、19 点越远. 我们先对上述方程进行进行回归, 得到的回归方程为:

$$\widehat{focus} = 0.04662 D_{12} |T - 0.5| - 0.06139 D_{12} |T - 0.8| + 0.07177 D_{13} T - 0.058499 |T - 0.5| + 0.02583$$

然后我们建立市占率  $rate$  以及直播关注度  $focus$ , 虚拟变量  $D_{11}, D_{12}, D_{21}, D_{22}$  的关系, 剔除不显著变量以后, 得到的回归方程为:

$$\widehat{rate} = 0.3539 focus + 3.4877 D_{12} focus - 0.01007 (D_{11} + D_{12}) + 0.02759 D_{13} + 0.06727$$

最后, 我们通过已知数据中的市占率  $rate$  以及广告时长  $adv$ , 来推测价格的平均值, 我们在 Eviews 中回归, 结果如下:

$$\frac{\widehat{P}}{adv} = -528822.8227 D_{22} focus + 490186.7586 focus - 3746.5851 D_{11} - 4376.9636 D_{12} + 5829.23358572 D_{22} + 6885.64196555$$

将上述方程合并消去  $rate$ ,  $focus$ , 最后得到的方程如下:

$$\begin{aligned}\widehat{\frac{P}{adv}} = & 5829.2336D_{22} - 4376.9636D_{12} - 3746.585D_{11} - 28675.489|T - 0.5| \\ & - 30093.279D_{12}|T - 0.8| - 528822.8227D_{22}(0.07177D_{13}T - 0.06139D_{12}|T - 0.8| \\ & - 0.058499|T - 0.5| + 0.04662D_{12}|T - 0.5| + 0.02583) + 35181.485D_{13}T \\ & + 22852.9634D_{12}|T - 0.5| + 19547.0048\end{aligned}$$

**2. CCTV12 合理底价估算模型** 对于 CCTV12, 我们将其节目类型分为法制专题, 电视剧, 综艺三种类型, 对应虚拟变量  $D_{11}$ ,  $D_{12}$ , 将广告播放时段分为节目 (播出) 前, 节目中插播, 节目 (结束) 后, 对应虚拟变量  $D_{21}$ ,  $D_{22}$ , 对应关系如下:

节目类型	$D_{11}$	$D_{12}$	插播时段	$D_{21}$	$D_{22}$
法制专题	1	0	节目前	1	0
电视剧	0	1	节目中	0	1
综艺	0	0	节目后	0	0

利用我们已知的数据, 我们希望预测出合理的底价. 首先我们希望能根据节目的类型以及其播放时段、节目播出时间相对值  $t$ , 来预测其直播关注度  $focus$  建立如下的多元线性回归模型:

$$focus = \beta_0 + \beta_{11}D_{11} + \beta_{12}D_{12} + \beta_{21}D_{21} + \beta_{22}D_{22} + \beta_3|t - 0.8|$$

这里, 考虑到中午 12 点及晚上 18 点是两个收视高峰, 为提高模型的准确性, 我们用  $|t - 0.8|$  来衡量一天中播出时间与晚上 19 点之间的差距, 差距越大播出时间离 19 点越远. 我们先对上述方程进行进行回归, 得到的回归方程为:

$$\begin{aligned}\widehat{focus} = & -0.00269|T - 0.8| + 0.0001896D_{11} + 0.0003705D_{12} \\ & (0.0000) \quad (0.4336) \quad (0.1349) \\ & -0.0002305D_{21} - 6.8498 \times 10^{-5}D_{22} + 0.001775 \\ & (0.3551) \quad (0.7823) \quad (0.0000)\end{aligned}$$

$$R = 0.647745 \quad F = 72.81846 \quad P = 0.000000$$

我们看到,  $D_{11}$ ,  $D_{12}$ ,  $D_{21}$ ,  $D_{22}$  的系数是不显著的, 我们将其剔除, 得到以下回归方程

$$\begin{aligned}\widehat{focus} = & -0.0026588|T - 0.8| + 0.001868 \\ & (0.0000) \quad (0.0000)\end{aligned}$$

$$R = 0.627768 \quad F = 340.6727 \quad P = 0.000000$$

然后我们建立市占率  $rate$  以及直播关注度  $focus$ , 虚拟变量  $D_{11}, D_{12}, D_{21}, D_{22}$  的关系, 剔除不显著变量以后, 得到的回归方程为:

$$\begin{aligned}\widehat{rate} = & 2.9231 * focus + 0.005970 D_{12} - 0.009411 \\ & (0.0000) \quad (0.0000) \quad (0.0000) \\ R = & 0.508964 \quad F = 104.1692 \quad P = 0.000000\end{aligned}$$

最后, 我们通过已知数据中的市占率  $rate$  以及广告时长  $adv$ , 来推测价格的平均值, 我们在 Eviews 中回归, 结果如下:

$$\begin{aligned}\frac{\widehat{P}}{adv} = & 1935.91 * adv - 2481762 * rate - 29602.6146441 D_{12} + 15643.29 \\ & (0.0000) \quad (0.0001) \quad (0.0000) \quad (0.0934) \\ R = & 0.254679 \quad F = 22.78025 \quad P = 0.000000\end{aligned}$$

将上述方程合并消去  $rate, focus$ , 最后得到的方程如下:

$$\widehat{P} = 1935.9104adv - 14786.7542D_{12} - 19288.5831|t - 0.8| + 21266.232 \quad (1)$$

可见, 底价和广告时长, 是否为电视剧, 播放时间和 19 点之间的距离有关系.

**3. 湖南卫视合理底价估算模型** 对于湖南卫视, 我们将其节目类型分为电视剧, 综艺, 新闻三种类型, 对应虚拟变量  $D_{11}, D_{12}$ , 将广告播放时段同上分为节目 (播出) 前, 节目中插播, 节目 (结束) 后, 对应虚拟变量  $D_{21}, D_{22}$ , 对应关系如下:

节目类型	$D_{11}$	$D_{12}$	插播时段	$D_{21}$	$D_{22}$
电视剧	1	0	节目前	1	0
综艺	0	1	节目中	0	1
新闻	0	0	节目后	0	0

利用我们已知的数据, 我们希望预测出合理的底价. 首先我们希望能根据节目的类型以及其播放时段、节目播出时间相对值  $t$ , 来预测其直播关注度  $focus$ . 按照节目类型通过散点图观察  $rate$  和  $t$  之间的关系, 我们可以建立如下的多元线性回归模型:

$$\begin{aligned}focus = & \beta_0 + \beta_1 D_{11}T + \beta_2 |T - 0.3| + \beta_3 |T - 0.5|^2 + \beta_{41} D_{11} |T - 0.5| + \beta_{42} D_{12} |T - 0.5|^3 + \\ & \beta_5 |T - 0.9| + \beta_6 D_{11} |T - 0.9| + \beta_7 D_{12} |T - 0.85|^{0.5}\end{aligned}$$



这里, 考虑到中午 12 点及晚上 18 点是两个收视高峰, 为提高模型的准确性, 我们用  $|t - 0.8|$ , 来衡量一天中播出时间与晚上 19 点之间的差距, 差距越大播出时间离 19 点越远. 我们先对上述方程进行进行回归, 剔除不显著的变量, 得到以下回归方程:

$$\begin{aligned}\widehat{focus} = & 0.1056D_{11}T - 0.3761|T - 0.3| + 0.5766|T - 0.5|^2 - 0.2777D_{11}|T - 0.5| \\ & - 0.2785D_{12}|T - 0.5|^3 - 0.2191|T - 0.9| - 0.05596D_{11}|T - 0.9| + \\ & 0.01575D_{12}|T - 0.85|^{0.5} + 0.1661 \\ R = & 0.627768 \quad F = 340.6727 \quad P = 0.000000\end{aligned}$$

然后我们建立市占率  $rate$  以及直播关注度  $focus$ , 虚拟变量  $D_{11}, D_{12}, D_{21}, D_{22}$  的关系, 剔除不显著变量以后, 得到的回归方程为:

$$\begin{aligned}\widehat{rate} = & -6.811focus + 7.4645D_{11}focus + 9.467D_{12}focus \\ & - 0.03183D_{11} - 0.03177D_{12} + 0.06875 \\ R = & 0.432745 \quad F = 0.432745 \quad P = 0.000017\end{aligned}$$

最后, 我们通过已知数据中的市占率  $rate$  以及广告时长  $adv$ , 来推测价格的平均值, 我们在 Eviews 中回归, 结果如下:

$$\begin{aligned}\widehat{P} = & 3377.6842adv + 701623.8718 * rate + \\ & 35282.3786D_{12} + 25743.5631D_{21} - 40738.4427 \\ R = & 0.441840 \quad F = 10.29082 \quad P = 0.000003\end{aligned}$$

将上述方程合并消去  $rate, focus$ , 最后得到的方程如下:

$$\begin{aligned}\widehat{P} = & 3377.6842adv - 22333.3937D_{11} + 12991.4813D_{12} + 25743.5631D_{21} + 1797414.2932|t - 0.3| \\ & + 1046930.8284|t - 0.9| - 504898.6989D_{11}t + 1327354.22D_{11}|t - 0.5| \\ & + 267417.5809D_{11}|t - 0.9| - 5237295.04563D_{11}(0.3761|t - 0.3| + 0.219|t - 0.9| - 0.1056D_{11}t \\ & + 0.2777D_{11}|t - 0.5| + 0.05596D_{11}|t - 0.9| + 0.2785D_{12}|t - 0.5|^3 - 0.01575D_{12}|t - 0.85|^{0.5} \\ & - 0.57665|t - 0.5|^2 - 0.166) - 6642297.022D_{12}(0.3761|t - 0.3| + 0.219|t - 0.9| - 0.1056D_{11}t \\ & + 0.2777D_{11}|t - 0.5| + 0.05596D_{11}|t - 0.9| + 0.2785D_{12}|t - 0.5|^3 - 0.01575D_{12}|t - 0.85|^{0.5} \\ & - 0.57665|t - 0.5|^2 - 0.166) + 1330920.1989D_{12}|t - 0.5|^3 - 75266.7893D_{12}|t - 0.85|^{0.5} \\ & - 2755823.893|t - 0.5|^2 - 786003.684\end{aligned}$$

### 5.5.3 第三问算例

输入单个用户 24h 内的收视行为, 输出 17 个行业大类在 24h 内的产品销售情况趋势, 部分结果如下图所示, 完整的结果见附录:

播放时间	性别	教育水平	所在行业	消费水平	广告类型
0:00-2:00	男	高中及以下	生活服务	低	交通
2:00-4:00	男	高中及以下	生活服务	低	交通
4:00-6:00	男	高中及以下	生活服务	低	家用电器
6:00-8:00	男	高中及以下	生活服务	低	无
8:00-10:00	男	高中及以下	生活服务	低	娱乐及休闲
10:00-12:00	男	高中及以下	生活服务	低	交通
12:00-14:00	男	高中及以下	生活服务	低	化妆品/浴室用品
14:00-16:00	男	高中及以下	生活服务	低	无
16:00-18:00	男	高中及以下	生活服务	低	无
18:00-20:00	男	高中及以下	生活服务	低	无
20:00-22:00	男	高中及以下	生活服务	低	娱乐及休闲
22:00-0:00	男	高中及以下	生活服务	低	商业/服务性行业

(c) 输入

period	商业/服务性行业	娱乐及休闲	家居用品	家用电器	房地产
1	0.0286	-0.0134	-0.0497	0.0096	-0.0239
2	0.0375	-0.0337	-0.0499	-0.0473	0.0318
3	0.0172	-0.0163	3.9999	0.0096	0.0007
4	0.0084	0.0262	3.0807	-0.0282	0.0113
5	0.0012	0.0118	2.0733	0.0194	-0.0762
6	-0.0537	-0.0147	0.9960	0.0010	-0.0163
7	-0.1007	-0.0586	3.0215	0.0406	-0.0019
8	0.0489	-0.0516	-0.0343	0.0538	0.0191
9	-0.0112	-0.0111	1.0599	0.0180	0.0607
10	-0.0129	0.0210	1.0280	0.0019	0.0867
11	-0.0167	-0.0355	3.0173	-0.0198	0.0156
12	-0.0403	0.0380	1.9817	0.0523	0.0116
sum	-0.0937	-0.1380	20.1244	0.1110	0.1191

(d) 输出

#### 5.5.4 第四问算例

我们讨论市场上只有一家广告供给方 ( 卖方), 两家企业美的, 格力. 初始状态下, 他们都选择不打广告, 即  $p_0^1 = p_0^2 = 0$ , 此时他们的市场份额相同, 即销售量相同. 由于格力做空调的技术更加成熟, 因此同样的产品, 其成本相对于美的更低. 设市场的总需求  $D = 100000$ , 空调价格  $P = 1000$ , 格力公司成本  $C^1 = 100$ , 美的公司成本  $C^2 = 300$ , 他们的不变成本  $FC = 5000000$ , 则不打广告时的销量为  $Q = 50000$ . 设格力的出价为  $p^1$ , 美的的出价为  $p^2$ , 则格力公司的销量为  $sell^1 = D \frac{p^1 + Q}{p^1 + p^2 + 2Q}$ , 美的公司的销量为  $sell^2 = D \frac{p^2 + Q}{p^1 + p^2 + 2Q}$ . 格力公司的利润为  $profit^1 = 900sell^1 - FC - 1.03p^1$ , 美的公司的利润  $profit^2 = 700sell^2 - FC - 1.03p^2$ . 这里  $1.03p^i$  是若将  $p^i$  元存入银行, 获得的本利和, 作为机会成本.

我们这里描述在一个时间点上的博弈过程:

**Step 1** 先求出两家公司第一轮博弈的出价  $p_1^1, p_1^2$ . 格力公司预测美的公司不会加大广告投入, 即已知  $p_1^2 = p_0^2 = 0$  的前提下最大化自身的利润, 解得

$$p_1^1 = \sqrt{\frac{900 \times 10^5 \times (50000 + p_1^2)}{1.03}} - p_1^2 - 100000 = 1990199.04$$

**Step 2** 同理对于美的公司, 美的预测格力的出价为  $p_1^1$ , 用同样的方法去求出  $p_1^2 = 9684597.172$

**Step 3** 对于格力, 重新预测美的的出价  $p_1^2$ , 更新自己的出价  $p_2^1 = 19380558$

依次类推, 最后的出价一定收敛 [1], 且收敛于  $(p^{1*}, p^{2*})$ , 其中  $p^{1*}, p^{2*}$  必为方程组

$$\begin{cases} \frac{\partial profit^1(p^{1*}, p^{2*})}{\partial p^1} = 0 \\ \frac{\partial profit^2(p^{1*}, p^{2*})}{\partial p^2} = 0 \end{cases}$$

的解, 这里的收敛过程为:

迭代次数 $i$	$p_i^1$	$p_i^2$
1	1990199.042875	9684957.172000
2	19380558.865072	16858447.973511
3	21478997.331412	16671960.054904
4	21452928.966776	16674863.338537
5	21453343.861087	16674817.240893
6	21453337.275745	16674817.972598
7	21453337.380274	16674817.960984
8	21453337.378615	16674817.961168
9	21453337.378641	16674817.961165
10	21453337.378641	16674817.961165

此时双方的利润分别为

$$profit^1 = 900 \times 10^5 \frac{Q + p^{1*}}{2Q + p^{1*} + p^{2*}} - FC = 45625000 > 45000000$$

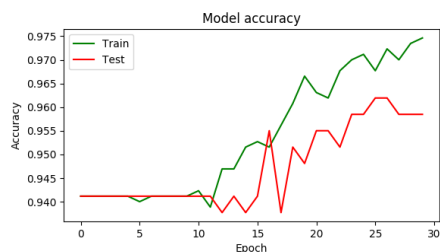
$$profit^2 = 700 \times 10^5 \frac{Q + p^{1*}}{2Q + p^{1*} + p^{2*}} - FC = 36375000 < 45000000$$

说明美的公司不会进入广告市场. 即此时真正的均衡广告出价应为  $(1990199.05, 0)$ , 双方真正的利润分别为  $(82847094.99, -2847094.99)$ , 虽然美的公司利润为负, 但是由于亏损额小于不变成本  $FC$ , 因此美的公司会继续经营空调业务.

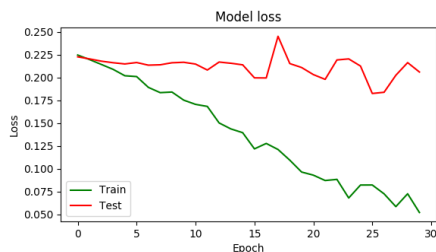
将上述过程循环执行, 最后能得出若美的公司不提高技术降低成本, 则由于连续的亏损, 其终究退出市场的结论.

## 六、模型检验

第一问, 神经网络的 Accuracy 及 loss 如下图所示:



(e) Model Accuracy

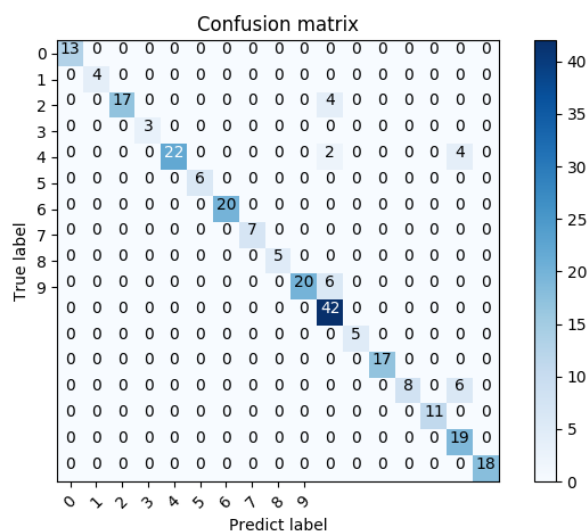


(f) Model Loss

这里我们选择的 loss 函数为 binary-crossentropy 交叉熵损失函数, 即

$$loss = - \sum_{i=1}^n \hat{y}_i \log y_i + (1 - \hat{y}_i) \log(1 - \hat{y}_i)$$

混淆矩阵如下:

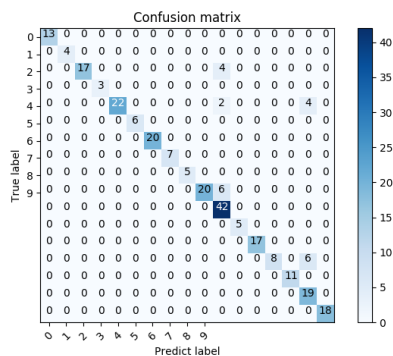


我们看出混淆矩阵中非 0 元素主要集中于主对角线, 这说明我们的模型命中率较好, 真实度较高.

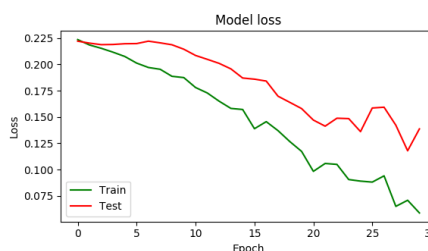
第二问, 我们从附录 A 的检验报告可看出, 所有的系数均显著, 且经济意义正确.

第三问, 和第一问类似, 其 loss 和混淆矩阵如下图

可见其真实程度较好.



(g) Model Accuracy



(h) Model Loss

## 七、模型评价

### 7.1 优点

- 神经网络模型准确率较高.
- relu 函数相比于 sigmoid 函数而言, 计算代价小, 速度快.
- 第二问第四问模型直观易懂, 经济学意义明显, 较好的反映了广告的供求关系.
- 第四问借助博弈论的理论, 将复杂的竞价交易过程简化成经济学模型.

### 7.2 缺点

- BP 神经网络训练能力同预测能力有时会存在矛盾, 也可能出现过拟合现象.
- 第二问采用多元线性回归, 检验其异方差性, 内生性等等会更完善.
- 某些线性回归的可决系数  $R^2$  只有 50% 左右, 仍有提高的空间.
- 我们假设市场的需求不变, 其实可以尝试引入时间变量, 将需求设置为价格和时间的函数, 会更加的贴近实际.

## 参考文献

- [1] 马洪宽. 博弈论. 同济大学出版社, 2015.
- [2] 艾鹏强. 基于时序行为和标签关系的个性化新闻推荐系统研究. Master's thesis, 天津理工大学, 2016.
- [3] 高鸿业 et al. 微观经济学. 北京: 中国人民大学出版社, 2007.
- [4] 曼昆. 宏观经济学. 中国人民大学出版社, 2016.
- [5] 李子奈, 潘文卿, et al. 计量经济学, volume 7. 高等教育出版社, 2000.

## 附录 A 第一问代码

```
#import warnings
#warnings.filterwarnings("ignore")

import matplotlib as mpl
mpl.use('TkAgg')
import matplotlib.pyplot as plt

import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
import pandas as pd
from keras.models import Sequential
from keras.layers import Dense,Activation,Dropout
from keras.optimizers import SGD
from sklearn.preprocessing import MultiLabelBinarizer
import pickle
import itertools
from keras.utils import plot_model
import os
from keras.utils.vis_utils import plot_model
import tensorflow as tf

data = pd.read_csv("T1_Artificial_Data.csv")
Y = data.values[:,7:8]
X = data.values[:,1:7]

X_mlb = MultiLabelBinarizer()
X_mlb.fit(X)
X = X_mlb.transform(X)
print(X_mlb.classes_)
#print(X[:5,:])
f = open("./model/T1_X_mlb", "wb")
f.write(pickle.dumps(X_mlb))
f.close()
```

```

Y_mlb = MultiLabelBinarizer()
Y_mlb.fit(Y)
Y = Y_mlb.transform(Y)
print(Y_mlb.classes_)
#print(Y[:5,:])
f = open("./model/T1_Y_mlb", "wb")
f.write(pickle.dumps(Y_mlb))
f.close()

X_df = pd.DataFrame(X)
Y_df = pd.DataFrame(Y)

'''
#X_df = pd.get_dummies(X_df)
#Y_df = pd.get_dummies(Y_df)
'''

print(X_df.head(5))
print(Y_df.head(5))

X_train, X_test, Y_train, Y_test = train_test_split(X_df, Y_df, test_size =
    0.33, random_state=42)
print("X_train_shape:", X_train.shape)
print("Y_train_shape", Y_train.shape)
print("X_test_shape", X_test.shape)
print("Y_test_shape", Y_test.shape)

model=Sequential([Dense(X_train.shape[1], input_dim=X_train.shape[1]),
    Activation('relu'),
    Dense(50),
    Activation('relu'),
    Dropout(0.2),
    Dense(64),
    Activation('relu'),
    Dropout(0.2),
    Dense(128),

```



```

    Activation('relu'),
    Dense(Y_train.shape[1]),
    Activation('softmax')])

model.summary()

sgd = SGD(lr=0.1, decay=1e-6, momentum=0.9, nesterov=True)
model.compile(loss="binary_crossentropy", optimizer=sgd, metrics=["accuracy"])
history =
    model.fit(X_train,Y_train,epochs=30,batch_size=5,validation_split=0.25)

loss, accuracy = model.evaluate(X_test,Y_test)

pred_y = model.predict(X_train)
pred_label = np.argsort(-pred_y)[:,:1]
true_label = np.argsort(-Y_train)[:][0]
print(Y_test.head(5))
print(true_label.head(5))

print("test loss:",loss)
print("test accuracy:", accuracy)
prob = model.predict(X_test.head(10))
#print(prob[:10,:])
idxs = np.argsort(-prob)[:,:3]

X_test = np.array(X_test)
#print(X_test.shape[1])
#print(len(X_mlb.classes_))
np.set_printoptions(suppress=True)
for i in range(idx.shape[0]):
    print(X_mlb.inverse_transform(X_test)[i])
for j in range(idx.shape[1]):
    print(Y_mlb.classes_[idx[i][j]],
          ":",np.around(prob[i][idx[i][j]]*100,4),end="% ")
print("\n")

plot_model(model, to_file='./Figure/model.png')

```

```

model.save("./model/Task_1.hdf5")

# 绘制训练 & 验证的准确率值
plt.plot(history.history['acc'], "g")
plt.plot(history.history['val_acc'], "r")
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()

# 绘制训练 & 验证的损失值
plt.plot(history.history['loss'], "g")
plt.plot(history.history['val_loss'], "r")
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='lower left')
plt.show()

confusion_mat = confusion_matrix(true_label, pred_label)
def plot_confusion_matrix(cm, classes, normalize=False, title='Confusion
    matrix', cmap=plt.cm.Blues):
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)
    if normalize:
        cm = cm.astype('float')/cm.sum(axis=1)[:, np.newaxis]
    thresh = cm.max()/2.0
    for i,j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j,i,cm[i,j], horizontalalignment='center', color='white' if cm[i,j] >
            thresh else 'black')
    plt.tight_layout()

```

```
plt.ylabel('True label')  
plt.xlabel('Predict label')  
plt.show()  
  
plot_confusion_matrix(confusion_mat, classes = range(10))
```

## 附录 B 问题一完整结果

test loss: 0.1598327404413468

test accuracy: 0.9562594218131824

**Input** ('18:00-20:00', '35-44', '女', '本科及以上', '金融保险', '高')

**Output** 衣着: 44.886% 金融/投资行业: 23.61% 化妆品/浴室用品: 11.862% 房地产: 8.7963% 个人用品: 2.2387% 家用电器: 2.1112%

**Input** ('0-18', '18:00-20:00', '低', '大专', '女', '教育')

**Output** 娱乐及休闲: 92.5324% 交通: 3.7829% 家居用品: 1.3213% 药品: 0.9536% 家用电器: 0.5559% 个人用品: 0.2446%

**Input** ('18-24', '18:00-20:00', '低', '大专', '生活服务', '男')

**Output** 交通: 99.6358% 娱乐及休闲: 0.146% 数码/电脑/办公用品: 0.0734% 房地产: 0.0548% 清洁用品: 0.0343% 酒类: 0.0177%

**Input** ('18-24', '20:00-22:00', 'IT 电子通信', '中', '本科及以上', '男')

**Output** 房地产: 62.9205% 数码/电脑/办公用品: 13.2506% 交通: 12.433% 清洁用品: 3.5259% 化妆品/浴室用品: 3.372% 酒类: 1.7673%

**Input** ('4:00-6:00', '65+', '低', '大专', '女', '无')

**Output** 个人用品: 97.4525% 药品: 1.1369% 家居用品: 0.6533% 食品及饮料: 0.207% 化妆品/浴室用品: 0.1453% 衣着: 0.1231%

**Input** ('18-24', '22:00-0:00', '低', '大专', '女', '生活服务')

**Output** 娱乐及休闲: 84.0095% 交通: 7.6668% 药品: 3.8912% 个人用品: 1.404% 家居用品: 1.3285% 清洁用品: 0.3355%

**Input** ('16:00-18:00', '18-24', '中', '本科及以上', '男', '金融保险')

**Output** 酒类: 89.113% 食品及饮料: 3.0111% 交通: 2.8059% 化妆品/浴室用品: 1.2169% 房地产: 1.1838% 数码/电脑/办公用品: 0.7427%

**Input** ('10:00-12:00', '25-34', '中', '女', '文化体育娱乐', '本科及以上')

**Output** 化妆品/浴室用品: 69.6046% 房地产: 9.5812% 金融/投资行业: 5.7738% 衣着: 5.4633% 清洁用品: 3.0178% 数码/电脑/办公用品: 1.6367%

**Input** ('10:00-12:00', '18-24', 'IT 电子通信', '中', '女', '本科及以上')

**Output** 清洁用品: 25.1317% 化妆品/浴室用品: 24.1783% 数码/电脑/办公用品: 23.2346% 房地产: 14.2259% 交通: 4.5675% 金融/投资行业: 2.8246%

**Input** ('10:00-12:00', '18-24', '低', '女', '本科及以上', '餐饮')

**Output** 清洁用品: 50.2956% 化妆品/浴室用品: 28.4322% 个人用品: 4.6298% 数码/电脑/办公用品: 3.4407% 食品及饮料: 3.3067% 金融/投资行业: 2.8819%

# 附录 C 问题二实证分析 Eviews 多元线性回归报告

CCTV1 的三个回归方程的 Eviews 报告如下图所示:

Dependent Variable: FOCUS Method: Least Squares Date: 04/30/19 Time: 17:03 Sample: 1 100 Included observations: 100					Dependent Variable: RATE Method: Least Squares Date: 04/30/19 Time: 17:07 Sample: 1 100 Included observations: 100					Dependent Variable: P/ADV Method: Least Squares Date: 04/30/19 Time: 17:14 Sample: 1 100 Included observations: 100				
Variable	Coefficient	Std. Error	t-Statistic	Prob.	Variable	Coefficient	Std. Error	t-Statistic	Prob.	Variable	Coefficient	Std. Error	t-Statistic	Prob.
D12*ABS(T-0.5)	0.046621	0.020454	2.279286	0.0249	FOCUS	0.353949	0.085975	4.116857	0.0001	D22*FOCUS	-528822.8	102129.3	-5.177975	0.0000
D12*ABS(T-0.8)	-0.061391	0.022861	-2.685462	0.0085	D12*FOCUS	3.487672	0.305317	11.42312	0.0000	FOCUS	490186.8	99284.52	4.937192	0.0000
ABS(T-0.5)	-0.058499	0.017446	-3.353074	0.0011	D11+D12	-0.010072	0.004520	-2.228100	0.0282	D11	-3746.585	1245.048	-3.009190	0.0034
D1*T	0.071772	0.016617	4.319196	0.0000	D13	0.027592	0.008462	3.260761	0.0015	D12	-4376.964	1325.968	-3.300963	0.0014
C	0.025830	0.005460	4.730821	0.0000	C	0.067265	0.003585	18.76150	0.0000	D22	5829.234	1328.288	4.386530	0.0000
										C	6885.642	1103.839	6.237903	0.0000
R-squared	0.367332	Mean dependent var	0.014265		R-squared	0.695854	Mean dependent var	0.079107		R-squared	0.286515	Mean dependent var	8675.276	
Adjusted R-squared	0.340893	S.D. dependent var	0.027125		Adjusted R-squared	0.683048	S.D. dependent var	0.033367		Adjusted R-squared	0.248564	S.D. dependent var	5583.139	
S.E. of regression	0.022025	Akaike info criterion	-4.744601		S.E. of regression	0.018785	Akaike info criterion	-5.062817		S.E. of regression	4839.767	Akaike info criterion	19.86525	
Sum squared resid	0.046083	Schwarz criterion	-4.614342		Sum squared resid	0.033523	Schwarz criterion	-4.932559		Sum squared resid	2.20E+09	Schwarz criterion	20.02156	
Log likelihood	242.2300	Hannan-Quinn criter.	-4.691883		Log likelihood	258.1409	Hannan-Quinn criter.	-5.010099		Log likelihood	-987.2623	Hannan-Quinn criter.	19.92851	
F-statistic	13.78943	Durbin-Watson stat	1.995974		F-statistic	54.33760	Durbin-Watson stat	0.867129		F-statistic	7.549548	Durbin-Watson stat	1.140179	
Prob(F-statistic)	0.000000				Prob(F-statistic)	0.000000				Prob(F-statistic)	0.000005			

(i) focus 函数

(j) rate 函数

(k) P/adv 函数

图 1 CCTV1

CCTV12 的三个回归方程的 Eviews 报告如下图所示:

<div>Dependent Variable: FOCUS Method: Least Squares Date: 04/29/19 Time: 22:59 Sample: 1 204 Included observations: 204</div> <table><tr><th>Variable</th><th>Coefficient</th><th>Std. Error</th><th>t-Statistic</th><th>Prob.</th></tr><tr><td>ABS(T-0.8)</td><td>-0.002659</td><td>0.000144</td><td>-18.45732</td><td>0.0000</td></tr><tr><td>C</td><td>0.001868</td><td>5.65E-05</td><td>33.03991</td><td>0.0000</td></tr><tr><td>R-squared</td><td>0.627768</td><td>Mean dependent var</td><td>0.001027</td><td></td></tr><tr><td>Adjusted R-squared</td><td>0.625926</td><td>S.D. dependent var</td><td>0.000782</td><td></td></tr><tr><td>S.E. of regression</td><td>0.000479</td><td>Akaike info criterion</td><td>-12.44201</td><td></td></tr><tr><td>Sum squared resid</td><td>4.63E-05</td><td>Schwarz criterion</td><td>-12.40948</td><td></td></tr><tr><td>Log likelihood</td><td>1271.085</td><td>Hannan-Quinn criter.</td><td>-12.42885</td><td></td></tr><tr><td>F-statistic</td><td>340.6727</td><td>Durbin-Watson stat</td><td>0.824684</td><td></td></tr><tr><td>Prob(F-statistic)</td><td>0.000000</td><td></td><td></td><td></td></tr></table>					Variable	Coefficient	Std. Error	t-Statistic	Prob.	ABS(T-0.8)	-0.002659	0.000144	-18.45732	0.0000	C	0.001868	5.65E-05	33.03991	0.0000	R-squared	0.627768	Mean dependent var	0.001027		Adjusted R-squared	0.625926	S.D. dependent var	0.000782		S.E. of regression	0.000479	Akaike info criterion	-12.44201		Sum squared resid	4.63E-05	Schwarz criterion	-12.40948		Log likelihood	1271.085	Hannan-Quinn criter.	-12.42885		F-statistic	340.6727	Durbin-Watson stat	0.824684		Prob(F-statistic)	0.000000				<div>Dependent Variable: RATE Method: Least Squares Date: 04/29/19 Time: 23:00 Sample: 1 204 Included observations: 204</div> <table><tr><th>Variable</th><th>Coefficient</th><th>Std. Error</th><th>t-Statistic</th><th>Prob.</th></tr><tr><td>FOCUS</td><td>2.923146</td><td>0.331534</td><td>8.817044</td><td>0.0000</td></tr><tr><td>D12</td><td>0.005970</td><td>0.000558</td><td>10.70563</td><td>0.0000</td></tr><tr><td>C</td><td>0.009411</td><td>0.000452</td><td>20.84198</td><td>0.0000</td></tr><tr><td>R-squared</td><td>0.508964</td><td>Mean dependent var</td><td>0.014288</td><td></td></tr><tr><td>Adjusted R-squared</td><td>0.504078</td><td>S.D. dependent var</td><td>0.005232</td><td></td></tr><tr><td>S.E. of regression</td><td>0.003684</td><td>Akaike info criterion</td><td>-8.354811</td><td></td></tr><tr><td>Sum squared resid</td><td>0.002729</td><td>Schwarz criterion</td><td>-8.30916</td><td></td></tr><tr><td>Log likelihood</td><td>855.1908</td><td>Hannan-Quinn criter.</td><td>-8.335073</td><td></td></tr><tr><td>F-statistic</td><td>104.1692</td><td>Durbin-Watson stat</td><td>0.845802</td><td></td></tr><tr><td>Prob(F-statistic)</td><td>0.000000</td><td></td><td></td><td></td></tr></table>					Variable	Coefficient	Std. Error	t-Statistic	Prob.	FOCUS	2.923146	0.331534	8.817044	0.0000	D12	0.005970	0.000558	10.70563	0.0000	C	0.009411	0.000452	20.84198	0.0000	R-squared	0.508964	Mean dependent var	0.014288		Adjusted R-squared	0.504078	S.D. dependent var	0.005232		S.E. of regression	0.003684	Akaike info criterion	-8.354811		Sum squared resid	0.002729	Schwarz criterion	-8.30916		Log likelihood	855.1908	Hannan-Quinn criter.	-8.335073		F-statistic	104.1692	Durbin-Watson stat	0.845802		Prob(F-statistic)	0.000000				<div>Dependent Variable: P Method: Least Squares Date: 04/29/19 Time: 23:01 Sample: 1 204 Included observations: 204</div> <table><tr><th>Variable</th><th>Coefficient</th><th>Std. Error</th><th>t-Statistic</th><th>Prob.</th></tr><tr><td>ADV</td><td>1935.910</td><td>284.3162</td><td>6.809005</td><td>0.0000</td></tr><tr><td>RATE</td><td>2481762.</td><td>617521.2</td><td>4.018910</td><td>0.0001</td></tr><tr><td>D12</td><td>-29602.61</td><td>6945.814</td><td>-4.261936</td><td>0.0000</td></tr><tr><td>C</td><td>-15643.29</td><td>9280.832</td><td>-1.685548</td><td>0.0934</td></tr><tr><td>R-squared</td><td>0.254679</td><td>Mean dependent var</td><td>39566.67</td><td></td></tr><tr><td>Adjusted R-squared</td><td>0.243499</td><td>S.D. dependent var</td><td>43673.30</td><td></td></tr><tr><td>S.E. of regression</td><td>37985.75</td><td>Akaike info criterion</td><td>23.94722</td><td></td></tr><tr><td>Sum squared resid</td><td>2.89E+11</td><td>Schwarz criterion</td><td>24.01228</td><td></td></tr><tr><td>Log likelihood</td><td>-2438.617</td><td>Hannan-Quinn criter.</td><td>23.97354</td><td></td></tr><tr><td>F-statistic</td><td>22.78025</td><td>Durbin-Watson stat</td><td>0.229483</td><td></td></tr><tr><td>Prob(F-statistic)</td><td>0.000000</td><td></td><td></td><td></td></tr></table>					Variable	Coefficient	Std. Error	t-Statistic	Prob.	ADV	1935.910	284.3162	6.809005	0.0000	RATE	2481762.	617521.2	4.018910	0.0001	D12	-29602.61	6945.814	-4.261936	0.0000	C	-15643.29	9280.832	-1.685548	0.0934	R-squared	0.254679	Mean dependent var	39566.67		Adjusted R-squared	0.243499	S.D. dependent var	43673.30		S.E. of regression	37985.75	Akaike info criterion	23.94722		Sum squared resid	2.89E+11	Schwarz criterion	24.01228		Log likelihood	-2438.617	Hannan-Quinn criter.	23.97354		F-statistic	22.78025	Durbin-Watson stat	0.229483		Prob(F-statistic)	0.000000			
Variable	Coefficient	Std. Error	t-Statistic	Prob.																																																																																																																																																																															
ABS(T-0.8)	-0.002659	0.000144	-18.45732	0.0000																																																																																																																																																																															
C	0.001868	5.65E-05	33.03991	0.0000																																																																																																																																																																															
R-squared	0.627768	Mean dependent var	0.001027																																																																																																																																																																																
Adjusted R-squared	0.625926	S.D. dependent var	0.000782																																																																																																																																																																																
S.E. of regression	0.000479	Akaike info criterion	-12.44201																																																																																																																																																																																
Sum squared resid	4.63E-05	Schwarz criterion	-12.40948																																																																																																																																																																																
Log likelihood	1271.085	Hannan-Quinn criter.	-12.42885																																																																																																																																																																																
F-statistic	340.6727	Durbin-Watson stat	0.824684																																																																																																																																																																																
Prob(F-statistic)	0.000000																																																																																																																																																																																		
Variable	Coefficient	Std. Error	t-Statistic	Prob.																																																																																																																																																																															
FOCUS	2.923146	0.331534	8.817044	0.0000																																																																																																																																																																															
D12	0.005970	0.000558	10.70563	0.0000																																																																																																																																																																															
C	0.009411	0.000452	20.84198	0.0000																																																																																																																																																																															
R-squared	0.508964	Mean dependent var	0.014288																																																																																																																																																																																
Adjusted R-squared	0.504078	S.D. dependent var	0.005232																																																																																																																																																																																
S.E. of regression	0.003684	Akaike info criterion	-8.354811																																																																																																																																																																																
Sum squared resid	0.002729	Schwarz criterion	-8.30916																																																																																																																																																																																
Log likelihood	855.1908	Hannan-Quinn criter.	-8.335073																																																																																																																																																																																
F-statistic	104.1692	Durbin-Watson stat	0.845802																																																																																																																																																																																
Prob(F-statistic)	0.000000																																																																																																																																																																																		
Variable	Coefficient	Std. Error	t-Statistic	Prob.																																																																																																																																																																															
ADV	1935.910	284.3162	6.809005	0.0000																																																																																																																																																																															
RATE	2481762.	617521.2	4.018910	0.0001																																																																																																																																																																															
D12	-29602.61	6945.814	-4.261936	0.0000																																																																																																																																																																															
C	-15643.29	9280.832	-1.685548	0.0934																																																																																																																																																																															
R-squared	0.254679	Mean dependent var	39566.67																																																																																																																																																																																
Adjusted R-squared	0.243499	S.D. dependent var	43673.30																																																																																																																																																																																
S.E. of regression	37985.75	Akaike info criterion	23.94722																																																																																																																																																																																
Sum squared resid	2.89E+11	Schwarz criterion	24.01228																																																																																																																																																																																
Log likelihood	-2438.617	Hannan-Quinn criter.	23.97354																																																																																																																																																																																
F-statistic	22.78025	Durbin-Watson stat	0.229483																																																																																																																																																																																
Prob(F-statistic)	0.000000																																																																																																																																																																																		

(a) focus 函数

(b) rate 函数

(c) P 函数

图 2 CCTV12

湖南卫视的三个回归方程的 Eviews 报告如下图所示:

Dependent Variable: FOCUS Method: Least Squares Date: 04/30/19 Time: 09:30 Sample: 1 57 Included observations: 57					Dependent Variable: RATE Method: Least Squares Date: 04/30/19 Time: 10:02 Sample: 1 57 Included observations: 57					Dependent Variable: P Method: Least Squares Date: 04/30/19 Time: 13:29 Sample: 1 57 Included observations: 57				
Variable	Coefficient	Std. Error	t-Statistic	Prob.	Variable	Coefficient	Std. Error	t-Statistic	Prob.	Variable	Coefficient	Std. Error	t-Statistic	Prob.
D11*T	0.105649	0.015248	6.928571	0.0000	FOCUS	-6.811388	2.607962	-2.611767	0.0118	ADV	3377.684	1009.586	3.345613	0.0015
ABS(T-0.3)	-0.376104	0.067736	-5.52460	0.0000	D11	-0.031831	0.008000	-3.978883	0.0002	RATE	701623.9	300526.6	2.334648	0.0235
ABS(T-0.5)^2	0.576649	0.101645	5.673193	0.0000	D12	-0.031770	0.010416	-3.050246	0.0036	D12	35262.38	10084.68	3.498813	0.0010
D11*ABS(T-0.5)	-0.277745	0.036742	-7.559253	0.0000	D11*FOCUS	7.464534	2.617316	2.851980	0.0063	D21	25743.56	12293.78	2.094031	0.0411
D12*ABS(T-0.5)^3	-0.278492	0.088967	-3.130277	0.0030	D12*FOCUS	9.467034	2.766589	3.421915	0.0012	C	-40738.44	17929.03	-2.272206	0.0272
ABS(T-0.9)	-0.219088	0.037691	-5.812236	0.0000	C	0.068751	0.007536	9.123084	0.0000					
D11*ABS(T-0.9)	-0.055956	0.015377	-3.639045	0.0007										
D12*ABS(T-0.85)^0.5	0.015749	0.007657	2.056945	0.0451										
C	0.166038	0.028838	5.757638	0.0000										
R-squared	0.612786	Mean dependent var	0.005730		Adjusted R-squared	0.432745	Mean dependent var	0.047060		R-squared	0.441840	Mean dependent var	39406.28	
Adjusted R-squared	0.548250	S.D. dependent var	0.007977		Adjusted R-squared	0.377131	S.D. dependent var	0.015844		Adjusted R-squared	0.398905	S.D. dependent var	40135.94	
S.E. of regression	0.005362	Akaike info criterion	-7.475179		S.E. of regression	0.012505	Akaike info criterion	-5.826156		S.E. of regression	31117.53	Akaike info criterion	23.61256	
Sum squared resid	0.001380	Schwarz criterion	-7.152592		Sum squared resid	0.007974	Schwarz criterion	-5.611098		Sum squared resid	5.04E+10	Schwarz criterion	23.79178	
Log likelihood	222.0426	Hannan-Quinn criter.	-7.349810		Log likelihood	172.0454	Hannan-Quinn criter.	-5.742577		Log likelihood	-667.9580	Hannan-Quinn criter.	23.68221	
F-statistic	9.495289	Durbin-Watson stat	2.391375		F-statistic	7.781322	Durbin-Watson stat	1.339509		F-statistic	10.29082	Durbin-Watson stat	1.303219	
Prob(F-statistic)	0.000000				Prob(F-statistic)	0.000017				Prob(F-statistic)	0.000003			

(a) focus 函数

(b) rate 函数

(c) P 函数

图 3 湖南卫视

## 附录 D 第三问代码

```
#import warnings
#warnings.filterwarnings("ignore")

import matplotlib as mpl
mpl.use('TkAgg')
import matplotlib.pyplot as plt

import keras
from sklearn.model_selection import train_test_split
import pandas as pd
import numpy as np
from keras.models import Sequential
from keras.layers import Dense,Activation,Dropout
from keras.layers import Add,Conv2D
from keras.optimizers import SGD
from sklearn.preprocessing import MultiLabelBinarizer
import pickle
from keras.utils import plot_model
import os
from keras.utils.vis_utils import plot_model

data = pd.read_csv("T3_Artificial_Data.csv")
X = data.values[1:,1:8]
Y_ = data.values[1:,8:25].astype(int)
Y_ = Y_.flatten()
X1_ = X[:,[0,2,3,4,5]]
X2_ = X[:,6]
X1 = [X1_[x,:] for x in range(0,len(X1_),24)]
X2 = [X2_[x:x+24] for x in range(0,len(X2_),24)]
Y = [Y_[x:x+24*17] for x in range(0,len(Y_),24*17)]

X1_mlb = MultiLabelBinarizer()
X1_mlb.fit(X1)
X1 = X1_mlb.transform(X1)
print("length of X1_mlb.classes:",len(X1_mlb.classes_),end="\n\n")
```

```

X1_df = pd.DataFrame(X1)
X2_df = pd.DataFrame(X2).replace(['无', '个人用品', '互联网', '交通', '农
业', '化妆品浴室用品/', '商业服务性行业/', '娱乐及休闲', '家居用品', '家用电
器', '房地产', '数码电脑办公用品//', '清洁用品', '药品', '衣着', '酒类', '金融投资行
业/', '食品及饮料'],
[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17])
Y_df = pd.DataFrame(Y)

print(X1_df.head(5))
print(X2_df.head(5))
print(Y_df.head(5))

X1_train, X1_test, X2_train, X2_test, Y_train, Y_test =
    train_test_split(X1_df, X2_df, Y_df, test_size = 0.33, random_state=42)
print("X1_train_shape:", X1_train.shape)
print("X2_train_shape:", X2_train.shape)
print("Y_train_shape:", Y_train.shape)
print("X1_test_shape:", X1_test.shape)
print("X2_test_shape:", X2_test.shape)
print("Y_test_shape:", Y_test.shape)

input1 = keras.layers.Input(shape=(X1_train.shape[1],))
x1 = keras.layers.Dense(X1_train.shape[1], activation='relu')(input1)
x1 = keras.layers.Dense(100, activation='relu')(x1)
x1 = keras.layers.Dropout(0.2)(x1)
x1 = keras.layers.Dense(Y_train.shape[1], activation='relu')(x1)

input2 = keras.layers.Input(shape=(X1_train.shape[1],1))
x2 = keras.layers.Conv1D(32, 3, padding='Same', activation='relu')(input2)
x2 = keras.layers.core.Flatten()(x2)
x2 = keras.layers.Dense(400, activation='relu')(x2)
x2 = keras.layers.Dense(Y_train.shape[1], activation='relu')(x2)

added = keras.layers.Add()([x1, x2])
out = keras.layers.Dense(Y_train.shape[1])(added)
model = keras.models.Model(inputs=[input1, input2], outputs=out)

```

```

model.summary()

model.compile(loss='mean_squared_error', optimizer='adam')
model.summary()
X2_train_r = np.zeros((10, 24, 1))
X2_train_r[:, :, 0] = X2_train.values[:, :]
X2_test_r = np.zeros((5, 24, 1))
X2_test_r[:, :, 0] = X2_test.values[:, :]

print(X2_train)
history = model.fit([X1_train, X2_train_r], Y_train, batch_size=2,
                    nb_epoch=60, validation_data=([X1_test, X2_test_r], Y_test))

plot_model(model, to_file='./Figure/model3.png', show_shapes=True,
            show_layer_names=True)
model.save("./model/Task_3.hdf5")

# 绘制训练 & 验证的损失值
plt.plot(history.history['loss'], "g")
plt.plot(history.history['val_loss'], "r")
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='lower left')
plt.show()

```



附录 E 第三问完整测试数据

period	个人用品	互联网	交通	农业	化妆品/浴室用品	商业/服务业	娱乐及休闲	家居用品	家用电器	房地产	数码/电脑办公	清洁用品	药品	衣着	酒类	金融/投资行业	食品及饮料
1	-0.0370	-0.0007	-0.0609	0.0159	-0.0396	0.0286	-0.0134	-0.0497	0.0096	-0.0239	0.0092	-0.0050	0.0523	0.0053	0.0233	-0.0385	-0.0354
2	-0.0101	-0.0165	0.0504	0.0418	0.0046	0.0375	-0.0337	-0.0499	-0.0473	0.0318	0.0369	0.0057	-0.0303	0.0347	0.0260	0.0141	0.0022
3	0.0066	-0.0851	-0.0619	0.0466	-0.0023	0.0172	-0.0163	3.9999	0.0096	0.0007	-0.0444	-0.0785	0.0038	-0.0023	-0.0023	0.0077	-0.0480
4	-0.0126	-0.0343	0.0129	0.0145	0.0391	0.0084	0.0262	3.0807	-0.0262	0.0113	-0.0128	-0.0082	0.0037	0.0123	-0.0182	-0.0180	-0.0775
5	0.0673	-0.0373	0.0194	-0.0088	-0.0065	0.0012	0.0118	2.0733	0.0194	-0.0762	0.0236	-0.0032	-0.0131	0.0368	0.0345	-0.0146	0.0276
6	-0.0297	0.0310	0.0754	0.0009	-0.0151	-0.0537	-0.0147	0.9960	0.0010	-0.0163	-0.0118	-0.0199	0.0209	-0.0635	0.0044	0.0211	0.0382
7	-0.0045	-0.0248	-0.0305	-0.0496	0.0329	-0.1007	-0.0586	3.0215	0.0406	-0.0019	0.0293	0.0349	0.0094	0.0325	0.0183	0.0434	-0.0545
8	-0.0413	-0.0552	-0.0254	0.0231	0.0133	0.0489	-0.0516	-0.0343	0.0538	0.0191	-0.0987	0.0011	0.0422	-0.0045	-0.0409	-0.0120	0.0194
9	-0.0497	0.0590	-0.0195	-0.0427	-0.0129	-0.0112	-0.0111	1.0599	0.0180	0.0607	0.0090	-0.0015	0.0012	-0.0443	0.0439	0.0066	0.0314
10	0.0551	0.0397	-0.0438	0.0002	-0.0271	-0.0129	0.0210	1.0280	0.0019	0.0867	0.0206	0.0111	-0.0214	0.0004	0.0159	0.0131	0.0005
11	-0.0256	-0.0347	0.0031	0.0198	-0.0424	-0.0167	-0.0355	3.0173	-0.0198	0.0156	-0.0016	0.0736	0.0019	0.0247	-0.0193	0.0119	-0.0146
12	0.0334	0.0127	-0.0074	-0.0106	0.0273	-0.0403	0.0380	1.9817	0.0523	0.0116	0.0178	-0.0100	-0.0209	0.0145	0.0699	0.0436	0.0431
sum	-0.0481	-0.1463	-0.0881	0.0511	-0.0286	-0.0937	-0.1380	20.1244	0.1110	0.1191	-0.0228	0.0001	0.0496	0.0466	-0.1554	0.0783	-0.0675

## 附录 F 第四问 Matlab 代码

```
clear
clc
p1=0;
p2=0;
for i=1:10
    p1=sqrt(900*10^5*(50000+p2)/1.03)-p2-100000;
    p2=sqrt(700*10^5*(50000+p1)/1.03)-p1-100000;
    fprintf('%d & %f & %f\\\\\\ \\n',i,p1,p2)
end
p1=1990199.042875;
p2=0;
vpa(900*10^5*(p1+50000)/(p1+p2+100000)-5000000)
vpa(900*10^5*(p2+50000)/(p1+p2+100000)-5000000)
```