

供应商和零售商之间定价关系的因果关系分析

马泽龙

东北大学, 辽宁 沈阳

Email: 1162604714@qq.com

收稿日期: 2021年3月23日; 录用日期: 2021年4月19日; 发布日期: 2021年4月27日

摘要

本文考虑使用两种机器学习算法来识别零售价格、供应商价格和销售量之间的因果关系, 目的是找到零售和供应商之间动态博弈下的定价权。本文使用最早期的PC算法和LiNGAM (线性非高斯无环模型)。所研究的数据集为某地区大型零售超市的食品产品的数据。PC算法不能在三个节点, 即零售价格、供应商价格和销售数量之间找到因果方向, 而基于非高斯分布误差数据下的LiNGAM算法能够找到三节点下的因果方向。

关键词

因果关系, PC算法, 定价权发现, 线性非高斯无环型模型

Causal Analysis of Pricing Relationship between Suppliers and Retailers

Zelong Ma

Northeastern University, Shenyang Liaoning

Email: 1162604714@qq.com

Received: Mar. 23rd, 2021; accepted: Apr. 19th, 2021; published: Apr. 27th, 2021

Abstract

In this paper, two machine learning algorithms are used to identify the causal relationship among retail price, supplier price and sales volume in order to find the pricing power under the dynamic game between retail and supplier. This paper uses the earliest PC algorithm and lingam (linear non-Gaussian acyclic model). The data set studied is the data of food products of a large retail supermarket in a certain area. PC algorithm cannot find the causal direction among three nodes, namely retail price, supplier price and sales quantity, while lingam algorithm based on non Gaus-

sian error data can find the causal direction under three nodes.

Keywords

Causality, PC Algorithm, Pricing Discovery, Linear Non-Gaussian Acyclic Model

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 前言

本文研究了以薯片为代表的零售食品行业的制造商和零售商之间的定价权。本文使用两种经典的算法：PC算法和LiNGAM(线性非高斯无环模型算法)。本文将两者都应用于推断某地区大型零售超市中收集的薯片零售价格、供应商价格和销售数量之间的因果关系。

本文所使用的两种机器学习算法，它们试图从观测数据中揭示变量之间的因果关系，而不是做结构的先验假设。两种算法在无环图结构上做搜索，得到的输出结果为有向无环图(DAG)。DAG被广泛用于表示非时间变量之间的因果关系[1]。PC算法是对该领域的早期算法，它对数据进行了高斯假设，使用条件独立性关系来发现因果关系。高斯数据的假设否定了高阶矩结构信息的需要[2]。然而，PC算法往往导致一组不可区分的因果模式(马尔科夫等价类)。例如，当 x 、 y 和 z 正态分布时，数据无法区分两个图： $x \leftarrow y \rightarrow z$ 和 $x \rightarrow y \rightarrow z$ 。这两个图都兼容相同的概率分布，因此是不可区分的和观测等效的。

LiNGAM的一个关键特征是假设变量(或扰动)的非高斯性。如果数据是非高斯的，本文可以应用高阶矩来识别因果模式。数据离正态性越远，LiNGAM确定的最终因果模式就越准确。本文阐明了PC和LiNGAM算法的异同。通过假设检验的方法，本文的数据拒绝正态原假设，因此可以使用LiNGAM识别变量中的因果结构。

2. PC算法

2.1. 条件独立性与因果图

PC算法是最早和广泛使用的机器学习算法之一。为了便于解释，先描述依赖模型的概念。设 X 、 Y 、 Z 表示三个互不相交的变量的子集。“ X 在给定 Z 的情况下独立于 Y ”可以用独立性声明 $I(X, Y | Z)$ 来表示。假设 M 是一个依赖模型，用来规定 $I(X, Y | Z)$ 是否为真。如果无向图 $G = (V, E)$ 中的 M 变量与 V 中的顶点集有直接对应关系，则 G 的拓扑结构反映了 M 模型中的一些性质。图 G 中 Z 的子集节点截断了所有的 X 节点到 Y 节点之间的路径，可以写作 $\langle X | Z | Y \rangle_G$ 。当 X 和 Y 的子节点集合通过集合 Z 连接时，控制 Z 可以理解为阻断以上路径[3]，并且引出了以下定义。

在满足元素的所有子集 X 、 Y 和 Z 不相交的情况下，如果 M 的变量和 G 的节点 V 之间存在一对一的对应关系，无向图 G 是依赖模型 M 的依赖性映射(或 D -映射)。

$$I(X, Y | Z)_M \Rightarrow \langle X | Z | Y \rangle_G$$

类似地， G 是 M 的一个独立映射(或 I -映射)如果：

$$I(X, Y | Z)_M \Leftarrow \langle X | Z | Y \rangle_G$$

如果 G 同时是 D —映射和 I —映射, 则称 G 是 M 的完美映射(P-map), 即:

$$I(X, Y | Z)_M \Leftrightarrow \langle X | Z | Y \rangle_G$$

给定一个满足因果马尔可夫条件和稳定性条件的概率分布 P , 对于连续正态分布和离散多项式分布, DAG (或贝叶斯网络)是 P 的一个完美映射。

2.2. 因果马尔科夫条件

PC 算法中常用的分布假设是连续变量的高斯分布。从更广义的意义上说, 一个参数化的数据集是一对 G, Θ_G , 其中 G 是一个数据集, Θ_G 是一个自由参数集, 它把图 G 映射到一个概率分布上。为了给参数指定一个概率分布, 必须对参数施加一些限制, 例如标准差不能为负。DAG G 所代表任意的联合概率分布, 且可按照如下因子分解定理分解:

$$p(X_1 = x_1, \dots, X_n = x_n) = \prod_{i=1}^n p(X_i = x_i | Pa_i^G = pa_i^G, \Theta_G) \quad (1)$$

Pa_i^G 是 G 中 X_i 节点的父节点集合。根据 G 因子分解, 其等价于 DAG 中的每个变量 X 独立于所有的 (既不是 X 的父变量也不是 X 的后代节点)的变量, 而依赖于 G 中 X 的所有父变量。满足式(1)中的性质的概率分布, 称为 G 的因果马尔可夫条件[4]。Pearl 提出了 d 分离的概念来确定满足因果马尔可夫条件所包含的条件独立关系。

2.3. d 分离

阻断: 连个节点集合 X, Y 被 Z 阻断包含如下几个情况:

在链结构和叉结构中 $\rightarrow Z \rightarrow$, $\leftarrow Z \rightarrow$, 其中 Z 是被控制的变量节点;

在对撞结构中 $\rightarrow Z \leftarrow$, Z 不被控制, 并且 Z 的后代节点也不被控制。

d 分离: 两个节点集合 X, Y 被集合 Z 所 d 分离当且仅当 XY 节点之间全部路径被 Z 节点所阻断。

PC 发现算法伪代码如下所示。

PC 算法 PC algorithm

1. 在顶点集 $X = \{X_1, \dots, X_n\}$ 上形成一个完全无向图 G

2. 设置 $k = 0$ 并重复以下两步:

i. 对于 G 中所有的有序 l 邻接节点对 (X_i, X_j) 满足 $|\text{Adjacencies}(X_i) \setminus X_j| \geq k$

若存在子集合 $S \subset \text{Adjacencies}(X_i) \setminus X_j$ 其满足:

$$|S| = k \text{ and } (X_i \perp X_j) | S$$

则删除 (X_i, X_j) 代表的连边, 并且更新 S 为 sepsset

ii. 设置 $K = K + 1$ 并返回 i 步

3. 对于每一个有序三元组 (X_i, X_j, X_k) 其中 (X_i, X_j) (X_j, X_k) 是相邻的边, 但 (X_i, X_k) 不是相邻的边, 此时检查是否属于 sepsset (X_i, X_k)

如果是, 则给边定向

4. 使剩余的无向边尽可能多地被定向, 这样既不会产生新的 v 型结构, 也不会产生有向环。

3. 线性非高斯无环模型(LINGAM)

PC 算法通常是基于条件独立性来搜索因果模式, 而 LiNGAM 算法则是基于函数组成发现因果方向。LiNGAM 的识别依赖于独立成分分析(ICA)。对于高斯变量, ICA 不能找到正确的混合矩阵, 因为许多不同的混合矩阵产生完全相同的高斯联合密度。ICA 仅适用于非高斯数据。如果数据是非高斯的, 本文可

以利用变量的高阶统计量来获得更强的识别结果。

3.1. 基于独立成分分析(ICA)的 LINGAM

中心极限定理(CLT)指出,任何独立源信号的混合通常具有比任何构成的原始变量更接近正态分布的分布[5]。假设本文观察混合变量的独立信号 $S = (S_1, \dots, S_n)$, 本文有

$$X = As \quad (2)$$

其中 s 是相互独立的分量。根据 CLT, 任意一个 s 都比混合变量 x 更加的非高斯, 本文可以把独立分量写成混合变量的逆矩阵形式。ICA 的目标是找到“分离矩阵” W , 其中 W 使统计独立分量的非高斯性之和最大化。

$$\tilde{S} = \tilde{W}X \text{ and } \tilde{W} = A^{-1} \quad (3)$$

3.2. 线性非高斯无环模型(LiNGAM)

Shimizu, Hoyer 等人开发了 LiNGAM, 基于独立分布非高斯扰动的假设, 实现了对非高斯分布变量 $X = (X_1, X_2, \dots, X_n)$ 的搜索策略。假设向量之间存在因果关系, 可以用如下结构方程模型表示:

$$x_i = \sum_{k(j)<k(i)} b_{ij}x_j + e_i \quad (4)$$

其中 $k(i)$ 表示 x_i 的因果顺序, 并且 X_i 是 X_j 的直接原因。扰动项 e 是相互独立的非高斯分布, 且方差非零。如果每个变量都有一个零均值, 本文就得到以下方程组:

$$e = (I - B)X = \tilde{B}X \quad (5)$$

$$\tilde{B} = \tilde{I} - \tilde{W} \quad (6)$$

$$(I - B)X = e \Rightarrow X = (I - B)^{-1}e = Ae \quad (7)$$

PC 发现算法伪代码如下所示。

LINGAM 算法 Lingam algorithm

1. 给定一个 $m \times n$ 的数据矩阵 x , 其中每列包含一个样本向量 x , 对 x 的每一行中减去均值, 应用 ICA 算法得到一个分解公式 $X = As$, 计算 $W = A^{-1}$ 。
 2. 找出一个 W 的行排列, 得到一个主对角线上没有零的矩阵 \tilde{W} 。
 3. 将 W 的每一行除以相应的对角元素, 得到一个新的矩阵 \tilde{W}' , 所有 1 元素都在其对角线上。
 4. 估计 B 的估计值 \tilde{B} by $X = BX + e$
 5. 对 \tilde{B} 进行排列, 直到它是严格下三角, 最后因果顺序。
-

4. 结构向量自回归模型(SVAR)

上述讨论没有提到变量之间的时间顺序及其可能的复杂性。一般来说, 这类问题是通过使用向量自回归来解决的。由于 Sims (1980) 提出的向量自回归模型(VAR 模型)不能提供足够的信息来研究同一时期经济变量之间的因果影响。所以本文使用改良的结构向量自回归模型(SVAR)。

假设 t 时刻有 n 个相关变量, 定义 SVAR 模型为

$$X_t = B_0X_t + B_1X_{t-1} + \dots + B_pX_{t-p} + e_t \quad (8)$$

其中 p 为使用滞后时间的个数, B_0 为瞬时效应, 反映变量在同一时间的因果顺序。由 SVAR 模型导出简化后的 VAR 模型为:

$$X_t = \sum_{i=1}^p (I - B_0)^{-1} B_i X_{t-i} + (I - B_0)^{-1} e_t = \sum_{i=1}^p M_i X_{t-i} + (I - B_0)^{-1} e_t \quad (9)$$

$$\hat{u}_t = (I - B_0)^{-1} e_t \quad (10)$$

$$(I - B_0)\hat{u}_t = e_t \Rightarrow \hat{u}_t = B_0\hat{u}_t + e_t \quad (11)$$

本文使用图模型在 SVAR 识别中的寻找 B_0 (10)中的矩阵。在使用 PC 算法时, 本文从测试 \hat{u} 之间的条件独立性关系开始。如果误差项是非正态分布的, 则用 LiNGAM 算法求出矩阵。

5. 数据与试验结果

5.1. 数据描述

本文使用中国某大型零售超市的商品扫描数据 Data.A, 该超市是该地区最大的连锁超市之一。数据来自调研的五年期间的 88 个商店。Data.A 将产品按 16 个区域分为 4 个价格等级进行定价: 促销价格、便宜、中等、高价位。Data.A 数据库是一个不平衡的面板数据, 有些商店的数据存在大量缺失值。其中 1 号、15 号店、21 号店。有本文所调查产品上好佳薯片和乐事薯片的完整销售数据, 因此使用这两个店的数据作为实证数据。

本文将本文的案例视为两个制造商(上好佳食品、乐事公司)与一个零售商的交互, 因为这些商店属于同一零售链, 通过对比两种商品能够更好的说明定价权的问题。本文假定两家制造商为每家商店预先设定了批发价。为了找到定价关系, 本文使用其中最核心的三个边变量: 零售价格(Pr)、制造商价格(Pm)和销量(Q)。

5.2. 实证结果

LiNGAM 算法的剪枝策略采用 bootstrapping 的一个简化版本。首先, 将数据分成 m 个大小相等的组, 然后在每组上运行 LiNGAM 算法, 为每组生成一个 B 估计矩阵, 然后求的平均值, 并计算其标准差。

本文首先估计了一个简化的 VAR 模型(8), 然后分析了方程上的估计残差。本文使用 Schwarz 信息准则(SIC)来选择最优时滞。本文直接使用 1 号 15 号 21 号店的原始数据, 而不是使用残差数据, 因为这些数据没有滞后效应。在实证检验上, 几乎所有的变量序列都使用增强 Dickey-Fuller 检验拒绝平稳性的原假设。本文在 0.05 显著性水平下验证了两个店(销售数据)在平稳漂移的情况下, 拒绝了增强 Dickey-Fuller 检验。相关的假设检验见表 1。

Table 1. Enhanced Dickey fuller test and P value of original data structure residuals

表 1. 原始数据结构残差的增强 Dickey-Fuller 检验及 P 值

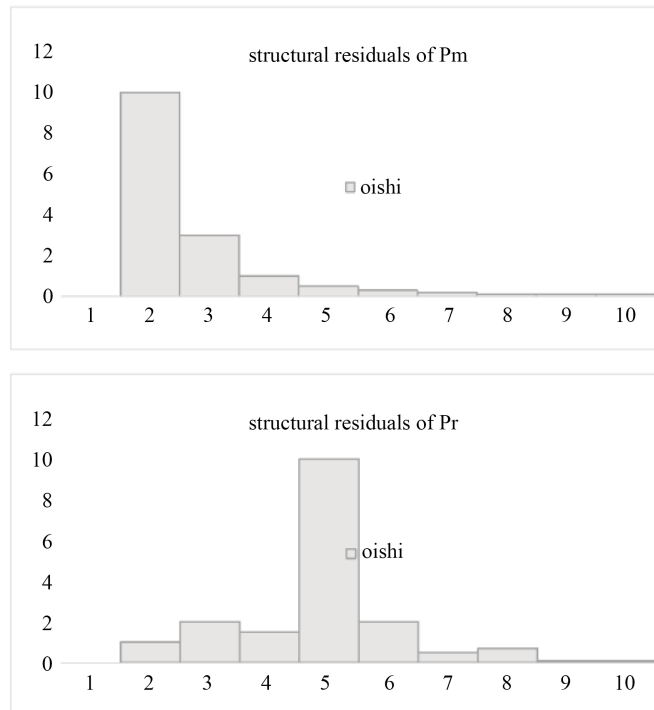
产品	商店	Q	Pr	Pm
Oishi	1	-15.808 (0.001)	-16.801 (0.001)	-18.231 (0.001)
Lay's	1	-15.817 (0.001)	-15.547 (0.001)	-18.645 (0.001)
Oishi	15	-11.909 (0.001)	-1.631 (0.097)	-2.658 (0.008)
Lay's	15	-15.434 (0.001)	-15.396 (0.001)	-17.662 (0.001)
Oishi	21	-12.116 (0.001)	-1.808 (0.067)	-2.699 (0.007)
Lay's	21	-15.796 (0.001)	-15.619 (0.001)	-17.140 (0.001)

此外，本文在 0.05 显著性水平下使用 Jarque-Bera 检验非高斯性的结构残差，见表 2。Jarque-Bera 测试结果发现商店 1 的 Oishi 的 Pr 数据的结构残差不拒绝零假设的正态分布。“在只有一个高斯分量的情况下，本文可以估计模型，因为单个高斯分量没有任何其他可以与之混合的高斯分量”。本文可以可靠地发现唯一正确的 LiNGAM 结果，当且仅当最多有一个误差项是高斯的。因此，本文认为对 1 号店的该数据应用 LiNGAM 算法是合适的。

Table 2. Enhanced Dickey fuller test and P-value of original data structure residuals
表 2. 原始数据结构残差的增强 Dickey-Fuller 检验及 P 值

产品	商店	Q	Pr	Pm
Oishi	1	6470.851 (0.001)	2.545 (0.234)	258.760 (0.001)
Lay's	1	1043.772 (0.001)	253.373 (0.001)	350.698 (0.001)
Oishi	15	539.005 (0.001)	35.447 (0.001)	398.893 (0.001)
Lay's	15	279.538 (0.001)	65.192 (0.001)	0.204 (0.001)
Oishi	21	1138.068 (0.001)	19.452 (0.002)	103.201 (0.001)
Lay's	21	1306.867 (0.001)	90.927 (0.001)	424.459 (0.001)

图 1 显示了原始数据经验分布的直方图，从上到下依次对应 Pm、Pr、Q。在各个商店的数据中，可以看到在各个变量间，至多只有一个类高斯分布。



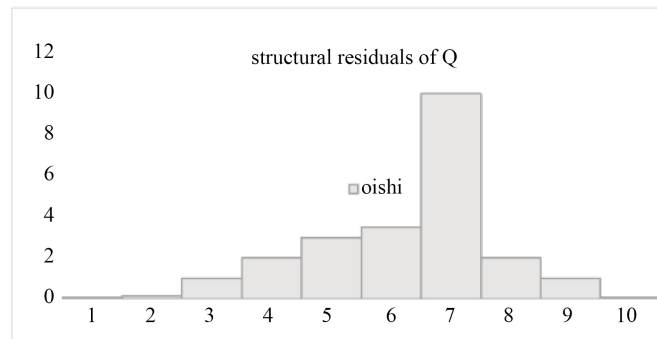


Figure 1. Distribution of structural residuals
图 1. 结构残差的直方图

从图 2 可以看出，PC 算法在 0.1 的显著性水平下返回无向边，说明该算法没办法有效发现三个节点之间的因果关系。另一方面，VAR-LiNGAM 的结果很好地表明了定价模式，即 $p_r \rightarrow p_m$ 。换句话说，该零售商店有能力影响这两个著名的薯片制造商的价格这意味着零售商有更大的定价权。

	VAR-PC	VAR-LiNGAM
1		
15		
21		

Figure 2. Causality diagram of VAR-PC and VAR lingam of Oishi data
图 2. Oishi 数据的 VAR-PC 和 VAR-LiNGAM 的因果图

6. 结论

本文探讨了两种机器学习算法，PC 算法和 LiNGAM 算法，试图揭示变量之间的因果关系，包括零售价格，制造商价格，和销售数量，以确定更大的定价权的一方。本文的 VAR-LiNGAM 图表清楚地表明，零售价格影响了两家主要的薯片制造商的出厂价格，并且具有非常明显的结构因果效应。通过额外的市场研究结果表明，零售商超的巨大定价权可能来自其在超市行业的市场份额，其自有品牌产品的销售，或来自两个薯片制造商之间的激烈竞争。然而，由于缺乏数据来估计制造商间的竞争，本文无法验证后一种猜想。而且该商超数据库缺乏关于制造商利润的信息，如果通过利润分析可能会从额外的角度去验证二者之间的定价关系。验证发现零售商相对于酸奶制造商具有更高的议价能力，这一发现与本文的 VAR-LiNGAM 结果一致。但是值得注意的是，对于通用的因果问题发现仍然是误差项是否足够非正态，以保证 LiNGAM 能够生效，即数据的质量足以诱导正确的估计。

参考文献

- [1] Pearl, J. (2009) Causality: Models, Reasoning, and Inference. 2nd edition, Cambridge University Press, Los Angeles.
<https://doi.org/10.1017/CBO9780511803161>
- [2] Shimizu, S., Hyvärinen, A., Kano, Y. and Hoyer, P.O. (2005) Discovery of Non-Gaussian Linear Causal Models Using ICA. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence (UAI-2005), Quebec. non-Gaussianity.*
- [3] Hyvärinen, A., Zhang, K., Shimizu, S. and Hoyer, P.O. (2010). Estimation of a Structural Vector Autoregression Model Using Non-Gaussianity. *Journal of Machine Learning Research*, **11**, 1709-1731.
- [4] Spirtes, P. (2010) Introduction to Causal Inference. *The Journal of Machine Learning Research*, **99**, 1643-1662.
- [5] Stone, J.V. (2004) Independent Component Analysis: A Tutorial Introduction. Cambridge, MA: MIT Press.
<https://doi.org/10.7551/mitpress/3717.001.0001>