

# 基于聚类与机器学习的零售商品销量预测<sup>①</sup>



周雨, 段永瑞

(同济大学 经济与管理学院, 上海 200092)  
通讯作者: 段永瑞, E-mail: yrduan@163.com

**摘要:** 本文提出一种基于 K-means 聚类与机器学习回归算法的预测模型以解决零售行业多个商品的销售预测问题, 首先通过聚类分析识别出具有相似销售模式的商品从而实现数据集的划分, 然后分别在每个子数据集上训练了支持向量回归、随机森林以及 XGBoost 模型, 通过构建数据池的方式增加了用于训练模型的数据量以及预测变量的选择范围. 在一家零售企业的真实销售数据集上对提出的模型进行了验证, 实验结果表明基于 K-means 和支持向量回归的预测模型表现最优, 且所提出的模型预测效果明显优于基准模型以及不使用聚类的机器学习模型.

**关键词:** 零售行业; 销售预测; 时间序列; 机器学习; 聚类

引用格式: 周雨, 段永瑞. 基于聚类与机器学习的零售商品销量预测. 计算机系统应用, 2021, 30(11): 188-194. <http://www.c-s-a.org.cn/1003-3254/8147.html>

## Retail Products Sales Forecast Based on Clustering and Machine Learning

ZHOU Yu, DUAN Yong-Rui

(School of Economics and Management, Tongji University, Shanghai 200092, China)

**Abstract:** In this study, we propose a forecasting model based on K-means clustering and a machine learning regression algorithm for the sales forecasting of multiple commodities in the retail industry. First, we utilize the clustering technique to identify commodities with similar sales patterns and then divide the whole dataset into different groups. Subsequently, three machine learning regression algorithms, i.e., support vector regression, random forest and XGBoost models, are trained on each sub-dataset. The data size for model training and the scope of forecasting variables are increased by the construction of a data pool. The proposed models are verified on a real sales dataset of a retail company. The experimental results show that the forecasting model based on K-means and support vector regression performs the best, and the forecasting performance of the proposed models is significantly better than that of the benchmark models and the machine learning models without using clustering.

**Key words:** retail industry; sales forecasting; time series; machine learning; clustering

## 1 引言

信息化浪潮以及大数据技术的蓬勃发展为各行各业带来了机遇与挑战, 通过数据驱动管理决策活动将可能为企业创造巨大的商业价值<sup>[1]</sup>. 零售企业通常储存着大量的销售明细数据, 这些数据为企业进行销售预测活动提供了可行性, 销售预测是零售企业进行多项

管理决策活动的基础, 因而对于企业具有重要意义, 例如精准的销售预测可以有效减少因库存积压或库存短缺造成的利益损失、帮助管理者更好地制定市场营销策略以及提升客户满意度水平, 从而使企业建立起长期的竞争优势<sup>[2]</sup>.

销售预测通常被当作时间序列预测问题来进行处

① 基金项目: 国家自然科学基金 (71771179, 71532015)

Foundation item: National Natural Science Foundation of China (71771179, 71532015)

收稿时间: 2021-01-21; 修改时间: 2021-02-23; 采用时间: 2021-03-03; csa 在线出版时间: 2021-10-22

理,为了满足企业在战略、战术及日常运营等不同层级上的决策需要,销售明细数据需要相应的在时间、供应链及产品等多个维度上进行不同程度的聚合.目前大多数研究主要集中于解决高聚合级别如总销售的预测问题,对于低聚合级别如商品SKU预测的相关研究还相对较少<sup>[3]</sup>,而后者相对更难获得准确的预测结果,其原因在于低层级往往包含较多的数据噪声,且影响销售波动的因素及其作用效果十分复杂,不同商品的销售情况可能存在较大的差异.近些年来,许多研究对销售的影响因素进行了讨论,除了历史销售表现之外,部分文献发现促销和价格等相关因素能够有效提升促销期的预测表现<sup>[4,5]</sup>.Huber等通过实证分析验证了日历因素如节假日对于销售预测的提升作用<sup>[6]</sup>.此外,一些文献还讨论了天气因素对于预测结果的影响<sup>[7,8]</sup>.

从预测方法上看,常见的时序预测方法如指数平滑法<sup>[9]</sup>、ARIMA与状态空间模型<sup>[10]</sup>已经得到了较为广泛的讨论,然而传统时序方法的缺陷在于仅使用内部变量进行预测且无法有效捕捉非线性因素的影响.因此,许多文献提出基于数据驱动的机器学习方法将时序预测问题转化为监督学习的方式进行处理,其中较为常见的方法包括神经网络模型如极限学习机ELM<sup>[11]</sup>、深度学习网络DNN<sup>[12]</sup>以及回归算法如支持向量回归SVR<sup>[13,14]</sup>、基于回归树的集成算法如随机森林<sup>[15]</sup>、XGBoost<sup>[16]</sup>等.此外,部分文献认为结合聚类的预测模型可以有效提升预测效果,Venkatesh等对111个ATM机的每日现金需求量进行了预测,他们根据现金需求量的周季节性特征对ATM机进行了聚类,并在每个类别上分别训练了多种神经网络模型,实验结果表明基于聚类的预测模型总是优于不聚类的预测模型<sup>[17]</sup>.Lu等构建了基于独立成分分析、GHSOM聚类与SVR的月销售预测模型,根据38家PC销售分公司的历史销售数据对它们进行了类别的划分,并对每一个类别独立训练了SVR预测模型,实验结果表明该模型能有效提升预测效果<sup>[18]</sup>.这些研究所提出的预测模型在本质上是类似的,即在进行预测之前首先使用聚类算法将整个数据集划分为几个子数据集,通过提升数据集中的数据相似程度以达到减少数据噪声的目的,之后再通过每个子数据集中的数据构建出的数据池来进行预测模型的训练,这样所得到的预测模型的预测结果更加准确和稳定.不过需要注意的是,以上研究在预测变量的选择上仅考虑了历史滞后因素,因

此所提出的模型仍旧存在一定的改进空间.

总的来说,目前的相关研究仍主要集中于解决总销售预测或单个商品的销售预测问题,对于多个商品的销售预测场景的讨论还相对不足,因此本文将K-means聚类算法与SVR、随机森林、XGBoost等机器学习回归算法相结合以构建预测模型来解决零售行业中多个商品的日销量预测问题,通过构建数据池的方式扩充了用于模型训练的数据量,为了验证模型的预测效果,本文将所提出的模型与基准模型以及不使用聚类的预测模型进行了对比分析,所有模型的训练与评估都是在真实的零售销售数据集上完成的.

## 2 基于聚类与机器学习的预测模型

本文结合聚类与机器学习算法构建了商品级别的销售预测模型,模型的基本流程如图1所示.针对所要分析的问题场景,我们分别以 $Y = (y_1, y_2, \dots, y_t, \dots, y_n)$ 以及 $X = [x_{ij}], i = 1, 2, \dots, n, j = 1, 2, \dots, m$ 表示某件商品在 $n$ 个时间点上的销售量以及与销售量相关的外部变量的取值,则对于任意一个商品而言,预测目标即是基于已有数据预测该商品在下一个时间点的销售量 $y_{n+1}$ .

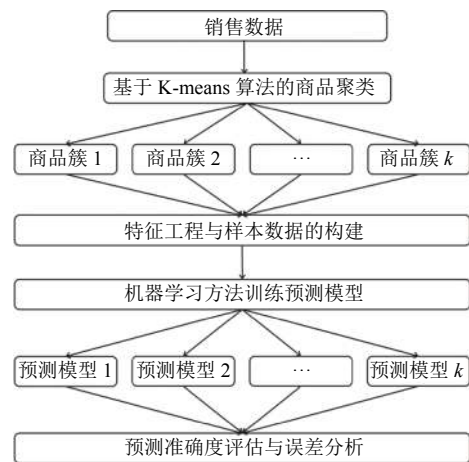


图1 基于聚类与机器学习的预测模型

### 2.1 基于 K-means 算法的商品聚类

零售企业往往需要对多种商品进行销量预测,考虑到不同商品销量的波动模式可能有较大的差异,若直接使用所有商品的数据训练模型则会引入大量的数据噪声,这可能会导致所训练出的模型有较大的预测偏差,因此在预测之前首先通过聚类分析将具有类似销售波动模式的商品进行划分,以此来达到降低数据

噪声的目的。

K-means 算法因其简单高效的特点在诸多领域得到了广泛应用<sup>[19]</sup>。该算法的核心思想为：首先从全部数据样本中随机选取  $K$  个初始聚类中心点  $\theta_i (1 \leq i \leq K)$  并计算其余样本点距离各聚类中心的欧式距离，并将该样本点划分到距离其最近的聚类中心所代表的簇中，之后计算每个簇中所有样本的均值以作为新的聚类中心并进行下一次迭代，算法的停止准则为达到最大的迭代次数或所有聚类中心都不再变化为止。

为了符合真实的预测场景，我们将原始销售数据从时间点  $t_1$  切割为训练集和测试集两部分，并假设仅有训练集是已知的，接着以商品在训练期上每个时间点的销量作为输入变量来实现商品的聚类，由此，样本点与聚类中心间的欧式距离计算公式为：

$$d(s, \theta_i) = \sqrt{\sum_{j=1}^{t_1} (s_j - \theta_{ij})^2} \quad (1)$$

其中， $s$  即为某件商品在训练期的样本数据， $s_j, \theta_{ij}$  分别代表商品  $s$  与商品簇中心  $\theta_i$  的第  $j$  个变量的取值。通常情况下，我们以整体误差平方和来衡量聚类效果，

$$[X|Y] = \begin{bmatrix} y_1 & y_2 & \cdots & y_l & x_{l+1,1} & \cdots & x_{l+1,m} & y_{l+1} \\ y_2 & y_3 & \cdots & y_{l+1} & x_{l+2,1} & \cdots & x_{l+2,m} & y_{l+2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ y_{n-l} & y_{n-l+1} & \cdots & y_{n-1} & x_{n,1} & \cdots & x_{n,m} & y_n \end{bmatrix} \quad (4)$$

机器学习需要有充足的数据量为其提供支持，数据量不足将很可能导致过拟合情况的发生，使得所训练出的模型仅在特定的训练数据集上拥有良好的拟合效果而不具备对未来进行预测的可用性。一般情况下单个时间序列对象所包含的数据量较少，在进行训练集和测试集的划分后使得可用于模型训练的数据变得更加有限，导致在单个时间序列对象的预测问题上机器学习方法的表现甚至劣于更为简单的时间序列预测方法<sup>[20]</sup>。为了解决这一问题，本文利用不同商品的数据来构建数据池，并基于整个数据池的数据进行模型训练和测试：假设  $D_s$  表示由商品  $s$  的历史销售数据与相关的外部变量数据按照上述方式处理所得到的特征数据与目标变量数据，则所有商品都可以按照相同的方式进行处理，最终我们将所得到的全部数据进行合并： $D = \cup D_s, s = 1, 2, \dots, N$ 。

至此，我们可以基于数据池  $D$  进行训练集、验证集与测试集的划分。训练集数据用于模型的训练，使模型能够对特征值与目标变量之间的映射关系进行学习。验证集的主要作用体现在通过超参数调优以避免模型

计算公式如下：

$$SSE = \sum_{i=1}^k \sum_{s \in \theta_i} |d(s, \theta_i)|^2 \quad (2)$$

## 2.2 机器学习处理时间序列预测问题的基本形式

机器学习可以将时间序列预测转化为监督学习的方式来进行处理，为了构建机器学习模型训练所需要数据形式，我们首先需要在原始时序数据上使用滑动时间窗口对内部变量即历史滞后值进行提取：假设使用过去  $l$  个时间点上的历史数据作为预测变量，则在长度为  $n$  的原始数据  $Y = (y_1, y_2, \dots, y_l, \dots, y_n)$  上总共可以得到  $n-l$  个样本数据：

$$[X|Y] = \begin{bmatrix} y_1 & y_2 & \cdots & y_l & y_{l+1} \\ y_2 & y_3 & \cdots & y_{l+1} & y_{l+2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ y_{n-l} & y_{n-l+1} & \cdots & y_{n-1} & y_n \end{bmatrix} \quad (3)$$

其中， $X$  表示预测变量矩阵即特征矩阵， $Y$  为目标变量。进一步，我们可以将外部变量加入到当前的特征矩阵  $X$  中，从而得到最终的特征矩阵和目标变量矩阵：

出现过拟合与欠拟合的情况，最后，模型预测效果的评估在测试集数据上进行，本文使用  $MAE$  与  $SMape$  作为预测效果的评估指标：

$$MAE = \frac{1}{m} \sum_{i=1}^m |\tilde{y}_i - y_i| \quad (5)$$

$$SMape = \frac{100\%}{m} \sum_{i=1}^m \frac{|\tilde{y}_i - y_i|}{(|\tilde{y}_i| + |y_i|)/2} \quad (6)$$

其中， $m$  为测试集的样本总数， $y_i$  为真实值， $\tilde{y}_i$  为预测值。

## 3 实证研究

### 3.1 数据

本文所使用的数据来源于国内某家零售企业所提供的真实销售数据集，我们选取了清洁用品品类下 160 件商品从 2011 年 4 月 1 日至 2011 年 11 月 8 日共 222 个时间点的日销售数据进行分析，对于少量缺失值使用了该时间点前后两周的平均销量进行了填充。其中 2011 年 4 月 1 日至 2011 年 9 月 27 日共 180 个时间点的的数据用于构建训练集和验证集，之后的 42 个时间点用于构建测试集，图 2 显示了随机抽取出的两件



商品的销售时序图,从图中我们可以看出商品呈现出明显的周季节性特点,但两种商品的销量趋势以及波动幅度都存在着明显的区别。这些规律在我们所分析的其他商品上是较为普遍存在的。

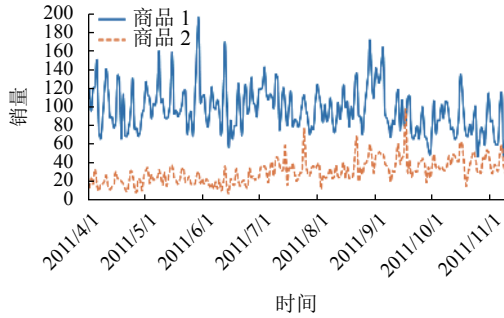


图2 商品销售时序图

### 3.2 预测变量

预测变量的选择往往依赖于特定领域的知识,我们参考了已有研究的相关结论并在数据探索性分析的基础上构建了本文的预测变量。

时间序列预测具备可行性的前提为假设过去出现过的数据模式在未来还会以类似的方式出现,因此自回归时间序列数据即历史滞后值是最常用也是最直接的预测变量。然而目前对于滞后值的选择范围还没有文献进行过系统的讨论,一般情况下对于短期预测而言使用近期的滞后值来进行预测会得到更为准确的预测效果,此外,基于滞后值计算的相关统计值也常被用来进行时间序列的预测。

外部变量的加入有助于提升机器学习模型的预测效果。除了用于划取历史滞后值之外,由时间维度衍生出的时不变的日历变量如月份、工作日与周末、节假日等在时间序列预测场景下很可能起到积极的作用。考虑到所分析的多数商品具有较为明显的周季节性特点,本文主要关注了商品销量在一周内的变化趋势以及在法定节假日期间的销售情况,并分别计算了相应的销售均值。从图3中我们可以看到,非节假日期间商品销量有着明显的周季节性特点,这与销售时序图所反映出现象是一致的,考虑到消费者在周末有更充裕的时间进行集中购物,商家也会在周末推出相应的促销优惠活动,因此周末的商品销量要明显地高于工作日。此外,节假日期间的销量普遍高于非节假日,特别是我们发现本应属于工作日的周一和周二的商品销量在节假日期间有了明显的增长。因此,本文在建立预测模型时纳入了相关的日历变量。最后,我们还考虑将

商品价格作为预测变量,由于价格本身也可以被看作是时间序列数据,因此对于价格变量同样使用了时间滑窗的方式进行处理,即我们考虑了短期价格波动对商品销量的影响,本文使用的所有预测变量见表1。

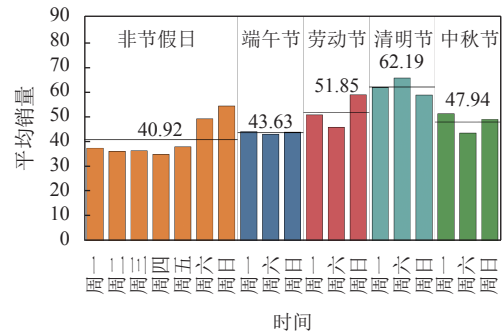


图3 周季节性与节假日对商品销量的影响

表1 预测变量

| 变量来源 | 变量描述                      |
|------|---------------------------|
| 交易数据 | 销量历史滞后值(Lag1-Lag7, Lag14) |
|      | 过去4周内每周的销量统计特征(平均值、标准差)   |
| 日历   | 周一至周日                     |
|      | 周末/工作日                    |
|      | 节假日/非节假日                  |
| 零售价  | 商品当天价格                    |
|      | 价格历史滞后值(Lag1-Lag7)        |
|      | 过去4周内每周的价格统计特征(平均值、标准差)   |

### 3.3 实证模型

本文使用了多种机器学习回归算法构建预测模型,其中包括支持向量回归(Support Vector Regression, SVR)、集成算法随机森林(Random Forest, RF)<sup>[21]</sup>以及XGBoost(eXtreme Gradient Boosting)算法<sup>[22]</sup>。作为实验对照,本文选用了Naïve以及季节性指数平滑作为基准方法,Naïve将上一个时间点的真实值作为预测值来进行预测,故又被称作随机游动预测法,该方法因其简单且有效的特点而被广泛地用作基准方法<sup>[23]</sup>。Hyndman等对指数平滑家族做了完备的阐述<sup>[24]</sup>,结合本文的预测场景我们选择了季节性指数平滑来对每个商品进行单独预测,使用了过去七天的历史销量作为该方法的预测变量。

#### 3.3.1 支持向量回归

支持向量回归是在统计学习理论上发展起来的基于结构风险最小化原理的机器学习算法,该算法通过核函数将低维空间的非线性问题转化为高维空间的线性问题来进行求解,因而能够有效降低过度拟合

的可能性,具有较强的泛化能力<sup>[25]</sup>.本文选用径向基函数作为支持向量回归的核函数,使得该算法能够更好地处理非线性关系.

### 3.3.2 随机森林

随机森林是基于决策树的 Bagging 类集成学习算法,其核心思想是通过构建多个相互独立的决策树并采用平均值计算或多数表决原则对每个决策树输出的结果进行处理以得到最终的结果.为了保证随机性,在构建每个决策树时都采用随机有放回抽样的方式对训练集数据进行样本抽取,并且在每个节点都通过随机选取特征子集来决定分支结果.随机森林有诸多优点,由于对独立的决策树采用并行计算方式,使得其在大型数据集上也有较快的计算速度,此外随机森林能够处理高维数据、不同类型的特征以及缺省值数据,且不要求对数据进行规范化.

### 3.3.3 XGBoost

XGBoost 是基于梯度提升树算法 GBDT 的高效实现与改进算法,本质上属于 Boosting 类的集成学习算法.Boosting 通过逐步迭代的方式将多个弱学习器结合为强学习器,其中每一步迭代所构建的弱学习器都是根据已有弱学习器的预测残差进行进一步学习的.GBDT 在进行目标函数优化时仅使用到了一阶导数信息,而 XGBoost 对损失函数进行了二阶泰勒展开并增加正则项以避免过拟合情况的发生.XGBoost 具有高准确度、能够并行计算、可以分布式处理高维稀疏特征等优点.

## 3.4 预测结果分析

本文首先使用全部训练集数据对机器学习预测模型进行了训练,采用十折交叉验证以及网格搜索法进行了超参数选择,并在测试集数据上验证了模型的预测效果,表 2 对各模型的预测评估指标结果进行了汇总.从预测指标反映的结果来看,季节性指数平滑以及基于机器学习的预测模型结果都要优于基准方法 Naive,其中 3 种基于机器学习的预测模型的预测效果又要好于季节性指数平滑,由于季节性指数平滑是基于单个商品的销售数据来进行预测的,所考虑的预测信息有限,且不能很好地处理非线性关系,因此在多个商品的预测场景下通过构建数据池来扩充数据量和预测变量的选择范围更有助于发挥出机器学习模型的预测效果.最后,支持向量回归的预测结果优于随机森林以及 XGBoost,其相比于 Naive 在 SMAPE 及 MAE 指标上分别获得了 20.45% 以及 18.56% 的提升.

表 2 基准模型与机器学习模型的预测结果

| 模型        | SMAPE (%)      | MAE            |
|-----------|----------------|----------------|
| Naive     | <u>42.5071</u> | <u>12.0427</u> |
| 季节性指数平滑   | 41.3300        | 11.9307        |
| 支持向量回归SVR | <b>33.8158</b> | <b>9.8073</b>  |
| 随机森林RF    | 37.8864        | 10.8907        |
| XGBoost   | 35.1105        | 10.3353        |

注: 黑体加粗为每个指标下的最优结果,下划线为每个指标下的最差结果.

进一步,本文进行了实验以验证聚类是否能够进一步提升机器学习预测模型的预测效果.聚类数对最终的预测效果具有直接影响,过少的聚类数意味着不能很好地识别具有相似销售模式的商品,即不能很好地实现降低数据噪声的作用,而过多的聚类数会导致子数据集中包含较少的数据量,这将会影响到预测模型的最终预测效果.基于以上考虑,我们分别取聚类数  $K=2, 3, 4$  并实验了 3 种机器学习模型在不同聚类数下的预测结果,为了方便对比,我们使用  $K=1$  来表示不使用聚类的情況.

表 3 汇总了不同聚类数下各预测模型的预测结果,从中我们可以看到结合聚类的机器学习模型的预测效果在原有基础上都有了进一步的提升,其中 K-SVR 模型在聚类数  $K=2$  时得到了最优预测结果,而 K-RF 以及 K-XGBoost 模型则在聚类数  $K=3$  时表现最佳,当聚类数为 4 时预测效果的提升程度则最小,因此我们认为将商品划分为 2 或 3 个商品簇是相对合适的.此外,无论在哪种情况下支持向量回归的预测结果总是优于两种集成算法,因此相较于聚类而言,算法的选择对最终的预测结果有着更大的影响.

为了更加直观的对比不同模型的预测结果,绘制了不同预测模型的预测值与真实值的散点图,具体情况见图 4.其中样本分布越集中于对角线说明预测误差越小.

## 4 结论

销售预测能够为零售企业安排生产计划、控制库存水平以及制定营销策略等多项管理决策活动提供支持.目前的大部分研究是针对单个时间序列对象如总销售或某个特定商品的销售情况进行分析的,而在真实预测场景中企业往往需要对多个商品进行预测,本文认为基于聚类分析与机器学习方法的预测模型能够很好地应用于此类预测场景,其原因主

要在此类预测场景下可以通过构建数据池的方式来增加用于训练机器学习模型的数据量以及预测变量的选择范围. 此外, 考虑到数据池构建的同时会引入数据噪声, 因此在训练模型之前可以通过聚类分析将具有相似销售模式的商品进行划分. 本文基于 K-means 聚类与支持向量回归、随机森林、XGBoost 构建了 3 种不同的预测模型, 并使用真实的零售企业销售数据对模型进行了验证. 结果表明 3 种预测模型的预测效果要明显好于基准预测方法, 其中表现最优的预测模型 K-SVR ( $K=2$ ) 在 *SMAPE* 及 *MAE* 两项预测评估指标上相较于 Naïve 方法分别提升了 21.23% 和 20.46%.

表 3 基于聚类与机器学习的预测模型的预测结果

| 模型        | 聚类数 $K$ | <i>SMAPE</i> (%) | <i>MAE</i>     |
|-----------|---------|------------------|----------------|
| K-SVR     | 1       | <u>33.8158</u>   | <u>9.8073</u>  |
|           | 2       | <b>33.4839</b>   | <b>9.5788</b>  |
|           | 3       | 33.6669          | 9.6310         |
|           | 4       | 33.6754          | 9.7041         |
| K-RF      | 1       | <u>37.8864</u>   | <u>10.8907</u> |
|           | 2       | 35.1959          | 10.1981        |
|           | 3       | <b>34.5631</b>   | <b>10.1213</b> |
|           | 4       | 35.6776          | 10.3626        |
| K-XGBoost | 1       | <u>35.1105</u>   | <u>10.3353</u> |
|           | 2       | 34.5187          | 10.1203        |
|           | 3       | <b>34.2659</b>   | <b>10.0412</b> |
|           | 4       | 34.9527          | 10.2064        |

注: 黑体加粗为同一模型在不同聚类数下的最优结果, 下划线为同一模型在不同聚类数下的最劣结果.

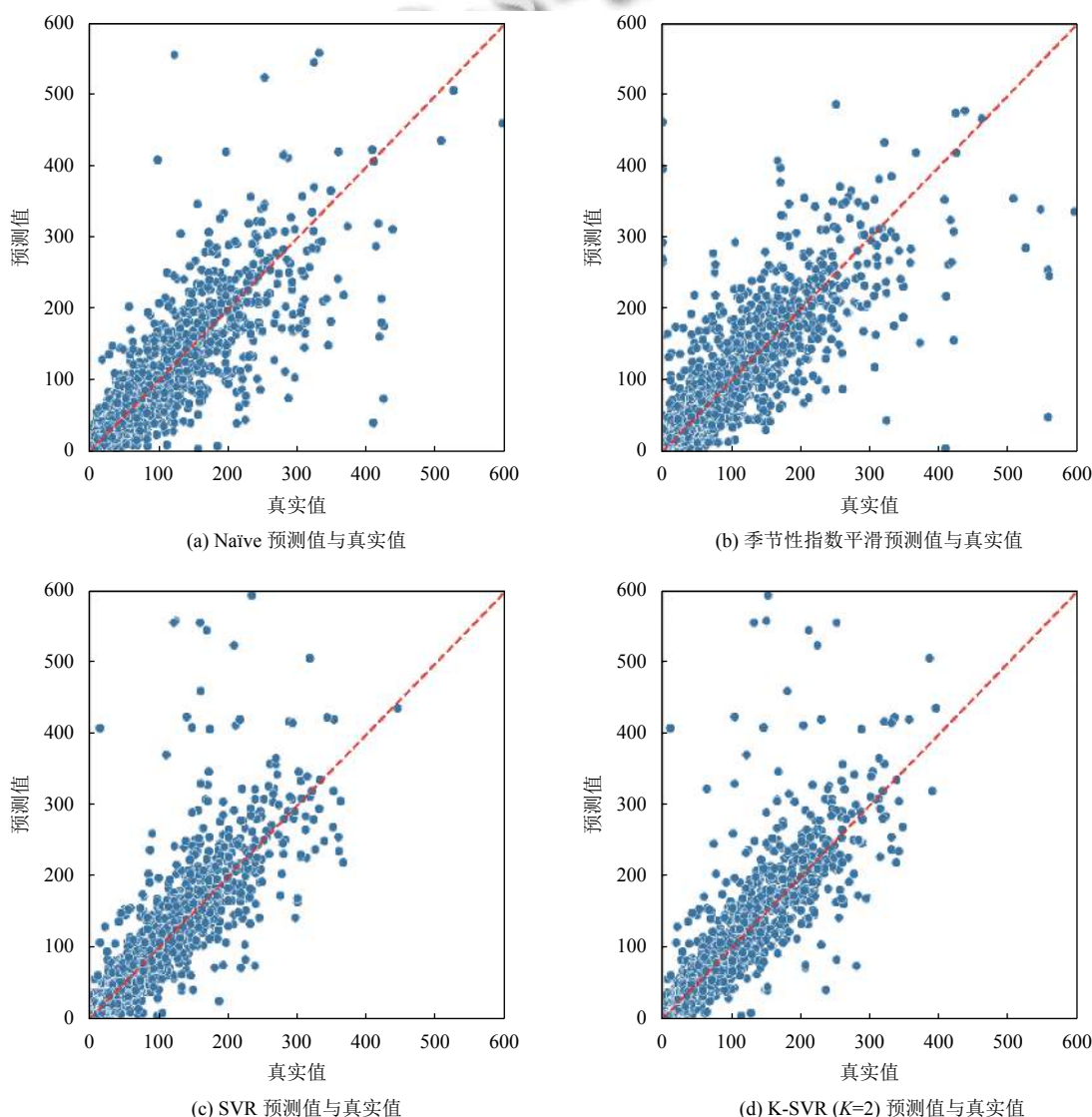


图 4 不同模型预测值与真实值关系散点图



本文所提出的预测思路不仅对零售行业的销量预测问题具有参考意义,其还可以应用到类似的时间序列预测场景中.对于未来的研究方向我们认为可以尝试将更多性能优异的机器学习算法用于此类预测场景.此外由于K-means聚类算法的缺陷之一为需要手动设定聚类数,因此也可以尝试使用其他类型的聚类算法如层次聚类法.

### 参考文献

- 1 徐宗本,冯芷艳,郭迅华,等.大数据驱动的管理与决策前沿课题.管理世界,2014,(11):158-163.
- 2 Fildes R, Ma SH, Kolassa S. Retail forecasting: Research and practice. International Journal of Forecasting, 2019. [doi: 10.1016/j.ijforecast.2019.06.004]
- 3 许姗姗.基于机器学习的商品销售预测的研究.统计与管理,2019,(4):49-52.
- 4 Ali ÖG, Sayın S, Van Woensel T, et al. SKU demand forecasting in the presence of promotions. Expert Systems with Applications, 2009, 36(10): 12340-12348. [doi: 10.1016/j.eswa.2009.04.052]
- 5 Huang T, Fildes R, Soopramanien D. Forecasting retailer product sales in the presence of structural change. European Journal of Operational Research, 2019, 279(2): 459-470. [doi: 10.1016/j.ejor.2019.06.011]
- 6 Huber J, Stuckenschmidt H. Daily retail demand forecasting using machine learning with emphasis on calendric special days. International Journal of Forecasting, 2020, 36(4): 1420-1438. [doi: 10.1016/j.ijforecast.2020.02.005]
- 7 Steinker S, Hoberg K, Thonemann UW. The value of weather information for E-commerce operations. Production and Operations Management, 2017, 26(10): 1854-1874. [doi: 10.1111/poms.12721]
- 8 Badorf F, Hoberg K. The impact of daily weather on retail sales: An empirical study in brick-and-mortar stores. Journal of Retailing and Consumer Services, 2020, 52: 101921. [doi: 10.1016/j.jretconser.2019.101921]
- 9 De Livera AM, Hyndman RJ, Snyder RD. Forecasting time series with complex seasonal patterns using exponential smoothing. Journal of the American Statistical Association, 2011, 106(496): 1513-1527. [doi: 10.1198/jasa.2011.tm09771]
- 10 Ramos P, Santos N, Rebelo R. Performance of state space and ARIMA models for consumer retail sales forecasting. Robotics and Computer-Integrated Manufacturing, 2015, 34: 151-163. [doi: 10.1016/j.rcim.2014.12.015]
- 11 Sun ZL, Choi TM, Au KF, et al. Sales forecasting using extreme learning machine with applications in fashion retailing. Decision Support Systems, 2008, 46(1): 411-419. [doi: 10.1016/j.dss.2008.07.009]
- 12 Loureiro ALD, Miguéis VL, da Silva LFM. Exploring the use of deep neural networks for sales forecasting in fashion retail. Decision Support Systems, 2018, 114: 81-93. [doi: 10.1016/j.dss.2018.08.010]
- 13 Lu CJ. Sales forecasting of computer products based on variable selection scheme and support vector regression. Neurocomputing, 2014, 128: 491-499. [doi: 10.1016/j.neucom.2013.08.012]
- 14 Yu XD, Qi ZQ, Zhao YM. Support vector regression for newspaper/magazine sales forecasting. Procedia Computer Science, 2013, 17: 1055-1062. [doi: 10.1016/j.procs.2013.05.134]
- 15 Pavlyshenko BM. Machine-learning models for sales time series forecasting. Data, 2019, 4(1): 15. [doi: 10.3390/data4010015]
- 16 Ji SW, Wang XJ, Zhao WP, et al. An application of a three-stage XGBoost-based model to sales forecasting of a cross-border E-commerce enterprise. Mathematical Problems in Engineering, 2019, 2019: 8503252.
- 17 Venkatesh K, Ravi V, Prinzie A, et al. Cash demand forecasting in ATMs by clustering and neural networks. European Journal of Operational Research, 2014, 232(2): 383-392. [doi: 10.1016/j.ejor.2013.07.027]
- 18 Lu CJ, Wang YW. Combining independent component analysis and growing hierarchical self-organizing maps with support vector regression in product demand forecasting. International Journal of Production Economics, 2010, 128(2): 603-613. [doi: 10.1016/j.ijpe.2010.07.004]
- 19 Jain AK. Data clustering: 50 years beyond K-means. Pattern Recognition Letters, 2010, 31(8): 651-666. [doi: 10.1016/j.patrec.2009.09.011]
- 20 Makridakis S, Spiliotis E, Assimakopoulos V. Statistical and Machine Learning forecasting methods: Concerns and ways forward. PLoS One, 2018, 13(3): e0194889. [doi: 10.1371/journal.pone.0194889]
- 21 Breiman L. Random forests. Machine Learning, 2001, 45(1): 5-32. [doi: 10.1023/A:1010933404324]
- 22 Chen TQ, Guestrin C. Xgboost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco: ACM, 2016. 785-794.
- 23 Hyndman RJ, Athanasopoulos G. Forecasting: Principles and Practice. 2nd ed. Melbourne: OTexts, 2018.
- 24 Hyndman RJ, Koehler AB, Ord JK, et al. Forecasting with exponential smoothing: The state space approach. Berlin: Springer, 2008.
- 25 张文雅,范雨强,韩华,等.基于交叉验证网格寻优支持向量机的产品销售预测.计算机系统应用,2019,28(5):1-9. [doi: 10.15888/j.cnki.csa.006905]