

# 基于时间序列模型的动态零售环境下 产品月销量预测

## ——基于多个模型的比较分析

徐浩淼

浙江财经大学数据科学学院, 浙江 杭州

收稿日期: 2023年11月10日; 录用日期: 2023年12月22日; 发布日期: 2023年12月29日

### 摘要

本研究旨在探讨如何在动态零售环境下有效地应用时间序列模型来预测产品的月销量。零售业的不断变化和不确定性使得销量预测变得至关重要。利用销售数据, 进行数据分析和特征工程。研究包括不同类型的模型, 如统计模型、机器学习模型和深度学习模型。实验结果以均方根任务(RMSE)为指标, 分析各模型的预测效果。本研究揭示了动态零售环境中的时间序列模型的潜力, 为员工提供更好的销量预测工具, 从而提高竞争力和盈利能力。这项研究还提供了时间序列模型在其他领域的潜在应用的意见。

### 关键词

时间序列模型, 动态零售, 月销量预测, 机器学习模型, 深度学习模型

# Prediction of Monthly Product Sales in a Dynamic Retail Environment Based on Time Series Models

## —Comparative Analysis Based on Multiple Models

Haomiao Xu

School of Data Sciences, Zhejiang University of Finance & Economics, Hangzhou Zhejiang

Received: Nov. 10<sup>th</sup>, 2023; accepted: Dec. 22<sup>nd</sup>, 2023; published: Dec. 29<sup>th</sup>, 2023

### Abstract

This study aims to explore how to effectively apply time series models in a dynamic retail envi-

ronment to predict monthly product sales. The continuous changes and uncertainties in the retail industry make sales forecasting crucial. Utilize sales data for data analysis and feature engineering. Research includes different types of models, such as statistical models, machine learning models, and deep learning models. The experimental results were analyzed using the root mean square task (RMSE) as an indicator to evaluate the predictive performance of each model. The discovery reveals the potential of time series models in dynamic retail environments, providing employees with better sales forecasting tools, thereby improving competitiveness and profitability. This study also provides insights into the potential applications of time series models in other fields.

## Keywords

Time Series Models, Dynamic Retail, Monthly Sales Forecast, Machine Learning Model, Deep Learning Model

Copyright © 2023 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. 引言

时间序列数据的预测在许多领域中具有重要意义，如金融、经济、交通和气象等。准确地预测时间序列的未来趋势和行为对于决策制定和规划具有关键作用。随着数据的积累和计算能力的提升，研究人员和数据科学家们探索了各种模型和算法来改进时间序列预测的准确性和效果。

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[1]。

在本研究中，我们关注时间序列数据的预测，并将重点放在不同模型对于时间序列预测效果的比较上。我们选择了一个包含日销售数据的真实数据集，其中每个数据点包括销售量、日期和商品等信息[2]。这个数据集具有一定的挑战性，因为随着时间的推移，商店和商品的列表会有所变化，需要构建一个稳健的模型来处理这种情况。

梁雯等(2016) [3]通过构建 VAR (Vector Autoregression)模型对安徽省碳排放与物流业之间的动态响应关系进行了实证研究。孙铭(2022) [4]分别采用 Random Forest、XGBoost (eXtreme Gradient Boosting)、LightGBM (Light Gradient Boosting Machine)对 2011~2016 年沃尔玛超市商品销售数据进行了全面透彻的分析及预测，发现 LightGBM 模型可以较好地用于处理海量数据、多特征背景下的大型超市商品销量预测问题，在大型连锁零售企业改善其经济效益方面有较大的意义。周雨等(2021) [5]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，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。周朱帆等(2020) [6]构建了基于差分自回归移动平均(ARIMA) (Autoregressive Integrated Moving Average)模型与随机森林(RF)、极端梯度提升树(XGBoost)的三种组合模型，将其应用于国内汽车市场批零量预测。张旭等(2021) [7]利用基于深度学习的 LSTM (Long Short Term Memory)模型对我国体育娱乐用品零售价格指数进行了预测。

本文研究的模型包括传统的统计模型、机器学习模型和深度学习模型。传统统计模型如 VAR 在时间序列预测中具有悠久的历史，并且在某些情况下仍然是强大的工具。机器学习模型如随机森林和支持向量回归在时间序列预测中提供了更多的灵活性和非线性建模能力。深度学习模型如长短期记忆网络 (LSTM) 则能够捕捉到更复杂的时间序列模式和长期依赖关系。

通过对比这些模型在数据集上的预测结果，我们可以评估它们在预测准确性、稳定性和计算效率方面的优劣。这样的比较有助于为时间序列预测问题提供洞见，并帮助研究人员和从业者选择最适合其数据和需求的模型。

本研究的目标是为时间序列预测领域提供实证性的比较分析，探索不同模型对于时间序列数据的预测效果，并为实际应用中的决策制定提供指导和建议。通过深入研究和理解不同模型的优缺点，我们可以进一步推动时间序列预测方法的发展和應用。

## 2. 数据介绍与分析

### 2.1. 数据介绍

本论文的研究数据集包含有关销售数据的信息，其中记录了从 2013 年 1 月至 2015 年 10 月的每日历史销售数据。数据集提供了商店、商品和销售相关信息，如商店编号、商品编号、销售日期、商品价格和销售数量等。

在论文中，通过对这个数据集进行详细的数据分析和特征工程，对销售数据进行了清洗和探索性分析，同时进行了特征提取和转换等处理。目标是理解数据的特征和趋势，为后续的时间序列预测建立良好的基础。

在分析和预测阶段，本文运用了多种预测模型，包括传统统计模型(如 VAR 模型)、机器学习模型(如随机森林、XGBoost 和 KNN (K-Nearest Neighbors))以及深度学习模型(如 LSTM 和 GRU (Gated Recurrent Unit))。我们对每个模型进行了详细介绍，并进行了参数调优和模型比较，评估它们在数据集上的预测准确度、稳定性和计算效率。

### 2.2. 数据分析

#### 2.2.1. 销售数量分析

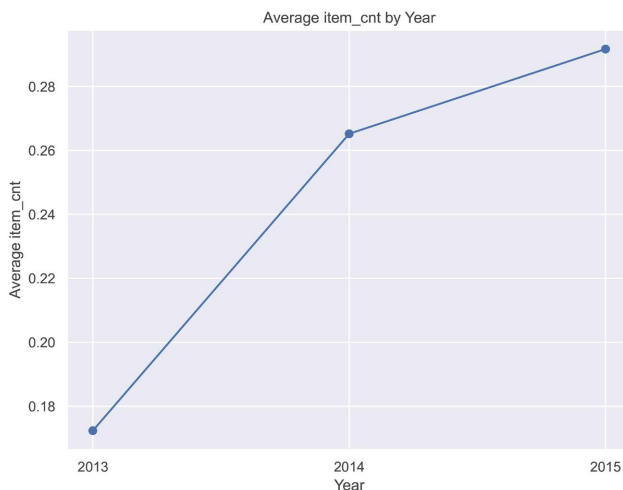


Figure 1. Average annual total sales volume  
图 1. 年销量均值

图 1 展示了每年销售数量的平均值，从图中可以看出平均销售数量在逐年上升。图 2 展示了每年销售数量的总和，从图中可以看出 2014 年的销售数量总量最高。

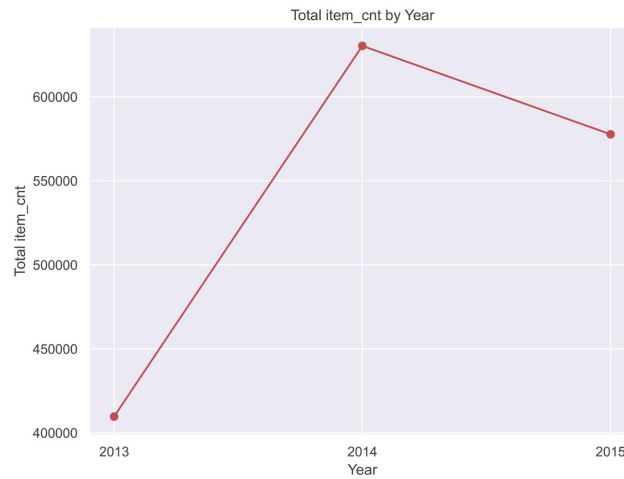


Figure 2. Annual total sales volume  
图 2. 年总销量

### 2.2.2. 商品种类分析

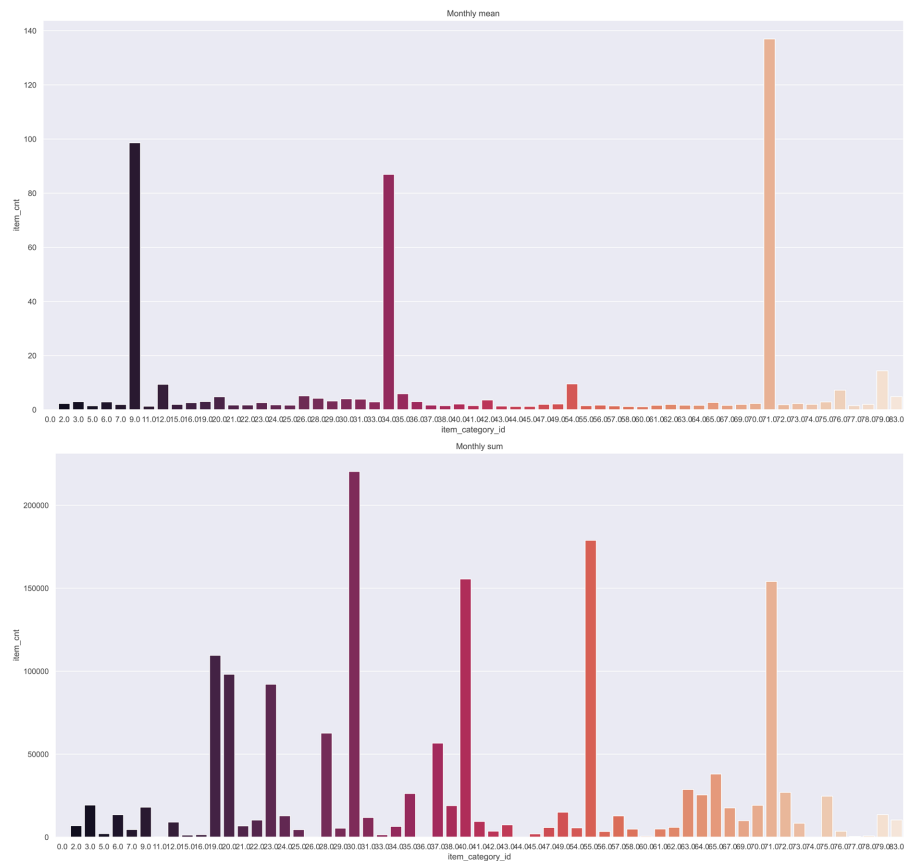
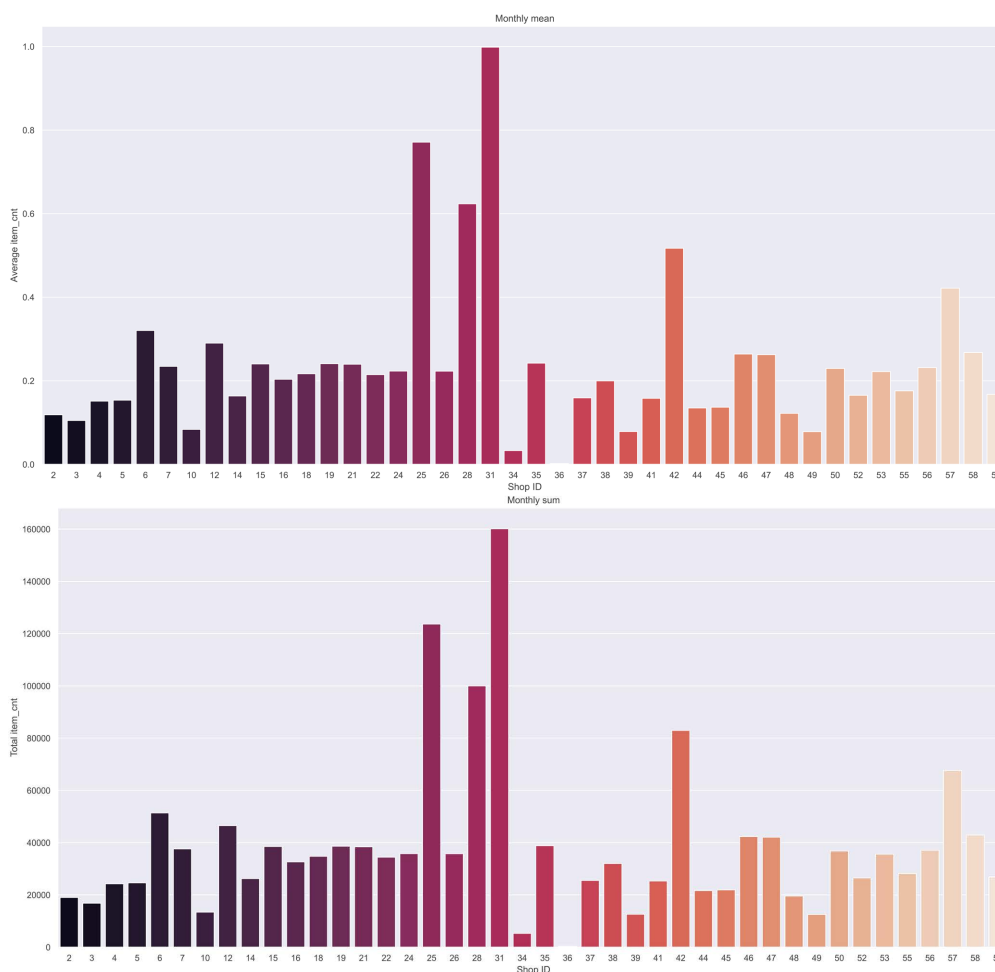


Figure 3. Monthly sales of different products  
图 3. 不同商品的月销量



**Figure 4.** Monthly sales of different stores  
**图 4.** 不同商店的月销量

图 3 通过两个柱状图展示了不同商品种类的销售情况，从图中可以看出月平均销量前三的商品编号为 9、34、71，月总量销量前三的商品编号为 30、55、71。

图 4 通过两个柱状图来展示不同商店的销售情况，从图中可以看出月平均销量和月总销量前三的商店编号都为 31、25、42。

### 2.3. 相关性分析

图 5 为各变量之间的热力图，它展示了各变量之间的相关性关系。“date\_block\_num”（日期块编号）与“year”（年份）和“month”（月份）之间的相关性非常强(0.94 和 0.22)。这是因为“date\_block\_num”是根据年份和月份生成的，因此它们之间存在明显的线性关系。

“shop\_id”（商店 ID）和“item\_category\_id”（商品类别 ID）之间的相关性较弱，但仍然存在一定程度的相关性(0.01 和 0.09)。这可能是某些商店更倾向于销售特定类别的商品。

“item\_price”（商品价格）与“mean\_item\_price”（平均商品价格）之间存在较强的正相关性(0.67)。这意味着商品价格和平均商品价格之间存在一定的线性关系。

“item\_cnt”（商品销售量）与“mean\_item\_cnt”（平均商品销售量）和“transactions”（交易次数）之间存在较强的正相关性(0.45 和 0.56)。这表明商品销售量与平均销售量和交易次数之间存在一定的线性关系。

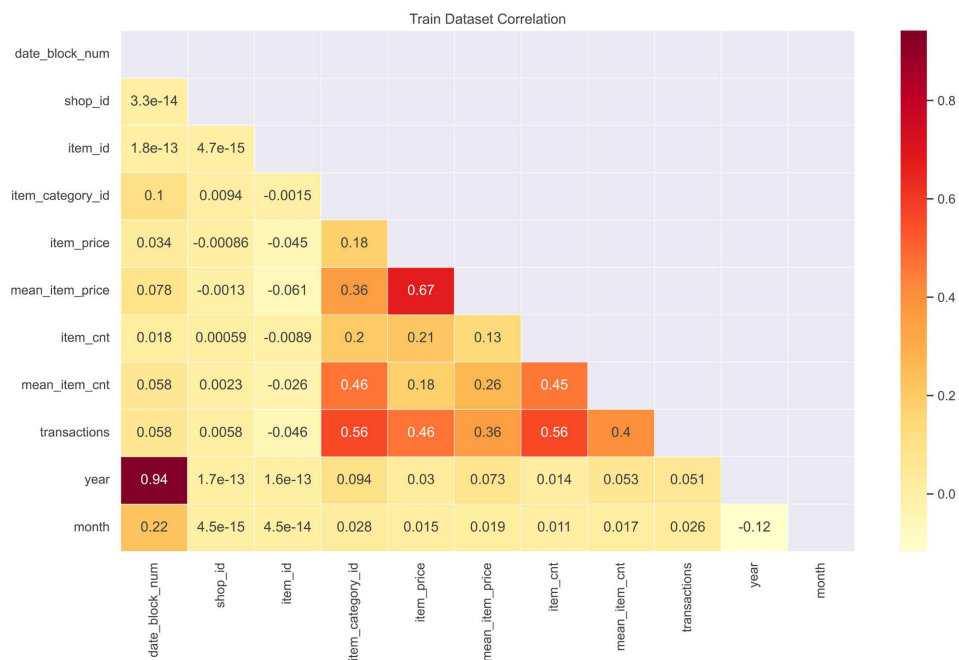


Figure 5. Correlation analysis of thermal maps  
图 5. 热力图相关性分析

## 2.4. 特征工程

### 2.4.1. 机器学习特征工程

本研究的特征工程主要包括以下步骤：

1) 数据过滤：根据特定条件过滤数据，保留符合要求的样本。具体条件包括 `item_cnt` (每个商品的销售数量)在 0 到 20 之间，以及 `item_price` (每个商品的价格)小于 400,000。

2) 特征提取与特征选择：`item_cnt_month` 特征：根据时间顺序对 `item_cnt` 进行滞后操作，得到下一个月的销售数量。`item_price_unit` 特征：计算每个商品的单位价格，即商品价格除以销售数量。`hist_min_item_price` 和 `hist_max_item_price` 特征：计算每个商品的历史最低价格和历史最高价格。`price_increase` 和 `price_decrease` 特征：计算每个商品的价格增加和价格降低量。滚动窗口统计特征：对 `item_cnt` 进行滚动窗口计算，包括最小值、最大值、平均值和标准差，针对不同的组合(`shop_id`、`item_category_id`、`item_id`)进行计算。`item_cnt_shifted` 特征：根据时间顺序对 `item_cnt` 进行滞后操作，得到前几个月的销售数量。`item_trend` 特征：根据前几个月的销售数量计算每个商品的销售趋势。并且本文只使用其中的部分特征来训练。

3) 数据集划分：将整体数据集划分为训练集、验证集和测试集，用于模型的训练、评估和预测。

4) 均值编码特征：基于训练集数据，计算不同特征(如 `shop_id`、`item_id`、`year` 和 `month`)的平均销售数量，并将其作为新的特征加入到训练集和验证集中。

### 2.4.2. 深度学习特征工程

首先将原始销售数据按照日期、店铺 ID 和商品 ID 进行聚合，生成了一个月度销售数据集。然后，通过移动窗口的方式构建了特征序列，其中包括了历史销售数量作为特征，并将下个月的销售数量作为标签。最后，将数据集划分为训练集和验证集，为深度学习模型的训练和验证做准备。并且每一行代表一个样本，每一列代表一个月份。



### 3. 模型介绍

#### 3.1. 统计模型

VAR 模型是一种多变量时间序列模型，用于描述多个变量之间的相互依赖关系和时间序列的动态演化。VAR 模型基于过去时间点的值来预测当前时间点的值，并假设变量之间存在线性关系。核心思想是将多个时间序列变量作为模型的输入，并利用这些变量的过去观测值来预测未来观测值。具体而言，VAR 模型将多个时间序列变量组成一个向量，通过引入滞后项(lag)来捕捉变量之间的动态关系。模型的阶数(order)表示引入的滞后项数量，用于控制模型的复杂度和灵活性。

VAR 模型是一种多变量时间序列模型，适用于描述多个变量之间的相互依赖关系和预测未来观测值。它是分析时间序列数据和预测未来趋势的有用工具。

LightGBM 是一个开源、快速、高效的基于决策树算法的提升框架，支持高效的并行训练。梯度提升(Gradient Boosting)思想是：一次性迭代变量，迭代过程中，逐一增加子模型，并保证损失函数不断减小。

#### 3.2. 机器学习模型

##### 3.2.1. LightGBM 模型

LightGBM 是一个开源、快速、高效的基于决策树算法的提升框架，支持高效的并行训练。梯度提升(Gradient Boosting)思想是：一次性迭代变量，迭代过程中，逐一增加子模型，并保证损失函数不断减小。

梯度提升决策树 GBDT 拥有梯度提升(GB)和决策树(DT)的功能特性，具有训练效果好、不易过拟合等优点。LightGBM 模型是 GBDT 的一种，用于处理海量数据问题。

##### 3.2.2. XGBoost 模型

XGBoost 是一种基于梯度提升树(Gradient Boosting Tree)算法的集成学习模型。它通过串行训练弱分类器(决策树)，并将它们组合成强分类器，以提高预测性能。

XGBoost 模型的核心思想是迭代地训练决策树模型，并在每次迭代中优化损失函数。它使用梯度提升算法，通过拟合前一轮模型的残差来逐步改进预测结果。每个新的决策树模型都被设计为修正前面模型的错误，使整体模型逐步逼近真实标签。

##### 3.2.3. 随机森林模型

随机森林是一种集成学习方法，用于解决分类和回归问题。它由多个决策树构成，每棵决策树都是基于随机选择的特征子集进行训练。

随机森林的基本思想是通过构建多个决策树，每棵树对输入样本进行独立的预测，并通过投票或平均来确定最终的预测结果[8]。这种集成的方式使得随机森林能够减少过拟合，并具有较好的泛化能力。

##### 3.2.4. 线性模型

线性模型是一类常见的机器学习算法，其基本思想是通过线性组合特征来建立输入变量与输出变量之间的关系。线性模型的预测结果是输入特征与其对应权重的线性组合。线性模型的优点包括简单、易于解释和快速训练。它们可以应用于各种不同类型的问题，包括分类和回归。常见的线性模型包括线性回归、逻辑回归和支持向量机(Support Vector Machine, SVM) [9]。

线性模型的局限性在于其假设输入特征和输出之间存在线性关系。对于复杂的非线性问题，线性模型可能无法提供准确的预测。然而，线性模型在解释性和计算效率方面具有优势，在许多实际应用中仍

然非常有用。

### 3.2.5. KNN 模型

K 最近邻(KNN)是一种简单的机器学习算法。它根据最相似的训练样本来预测未知样本的类别或值。具体地说, KNN 找到离未知样本最近的 K 个邻居, 并根据它们的标签进行预测。KNN 适用于分类和回归任务, 并且对各种数据类型和问题类型都适用。它的优点是易于实现, 能够处理非线性问题和多类别分类。然而, KNN 的缺点是计算复杂度较高, 对特征缩放和选择敏感。尽管如此, KNN 仍然是一个常用且有用的机器学习算法。

## 3.3. 深度学习模型

### 3.3.1. LSTM 模型

LSTM 模型由一系列的 LSTM 单元组成, 每个 LSTM 单元包含一个单元状态(cell state)和三个门: 输入门(input gate)、遗忘门(forget gate)和输出门(output gate)。这些门控制着信息的输入、输出和遗忘, 以便有效地处理序列数据。

LSTM 模型通过时间步(time step)逐个输入序列数据, 并根据当前输入和前一时间步的状态来更新单元状态。在每个时间步, LSTM 单元根据输入数据和当前状态计算输出, 同时更新单元状态和门的状态。这样, LSTM 模型能够根据序列数据的上下文信息学习并预测未来的值。

### 3.3.2. Autoencoder 模型

Autoencoder (自编码器)是一种无监督学习模型, 用于数据的降维、特征提取和生成。它由编码器和解码器两部分组成, 通过将输入数据压缩到低维编码表示, 再从编码表示中重建出原始输入数据。Autoencoder 的目标是最小化重建误差, 即使得重构的数据尽可能接近原始输入。

### 3.3.3. GRU 模型

GRU (门控循环单元)是一种常用的循环神经网络(Recurrent Neural Network, RNN)模型, 用于处理序列数据和建模时序关系。与传统的 RNN 模型相比, GRU 模型通过引入门控机制来更好地捕捉长期依赖性, 并且在训练过程中减轻了梯度消失的问题。

GRU 模型的关键是门控单元, 它由更新门(update gate)和重置门(reset gate)组成。更新门控制着前一时间步的信息在当前时间步的传递程度, 从而重置门的信息。这种门控机制允许 GRU 模型更好地决定如何使用和遗忘过去的信息, 从而有效地建模序列数据中的时序关系。

### 3.3.4. DeepAR 模型

DeepAR (Deep Autoregressive Model)深度自回归模型是一种用于时间序列预测的神经网络模型, 特别适用于具有季节性和长期依赖性的数据。DeepAR 模型基于循环神经网络(RNN)的架构, 并引入了注意力机制和编码器 - 解码器结构, 能够对时间序列数据进行建模预测。

DeepAR 模型的核心思想是将时间序列数据转化为条件生成问题, 即给定一段历史观测值, 预测未来的值。模型通过编码器将历史序列转换为隐藏表示, 并在解码器中使用这些隐藏表示生成未来的预测序列。模型通过自回归方式生成每个时间步的预测值, 将其作为输入序列的一部分, 逐步生成整个预测序列。

## 4. 结果描述与分析

### 4.1. 测度指标

RMSE (Root Mean Square Error, 均方根误差)是一种用于评估预测模型性能的指标。它测量了预测值



与实际观测值之间的平均差异程度，即模型预测值与真实值之间的平均偏差。

## 4.2. 统计模型预测

传统统计模型在时间序列预测领域具有悠久的历史。本文选择了向量自回归模型(Vector Autoregression, VAR)作为代表。VAR 模型能够捕捉时间序列之间的相互依赖关系，对于多变量时间序列数据具有较好的建模能力。为了检验数据集中各列的平稳性，我们对每一列应用了单位根检验(Augmented Dickey-Fuller test)。ADF 统计量用于评估时间序列是否具有单位根，即是否是非平稳的。每一列的 ADF 统计量分别为：-148.95、-450.39、-140.97、-317.91、-317.80、-225.93、-448.87、-448.80、-448.59、-449.20，这些 ADF 统计量的值远小于对应的临界值，进一步支持了这些列是平稳的结论。因此，可以直接使用 VAR 模型，VAR 模型所计算出来的均方根误差为 1.28。

## 4.3. 机器学习模型预测

### 4.3.1. LightGBM 与 XGBoost 预测

本文使用了 Optuna 优化框架来自动调整 LightGBM 和 XGBoost 模型的参数，并提高其性能。

Optuna 会随机生成一组参数配置，并将其传递给 LightGBM 与 XGBoost 模型进行训练。模型设置了早停机制，以在模型没有显著改善时提前停止训练，以节省时间和资源。

通过运行 100 次迭代的优化过程，找到了最佳的参数配置。使用这个最佳配置，训练了最终的 LightGBM 与 XGBoost 模型，以得到在完整训练数据集上训练的最佳模型。

LightGBM 的训练均方根误差与验证均方根误差分别为：0.71、0.78。图 6 展示了 LightGBM 模型的预测结果与真实标签之间的关系，可以看出 LightGBM 模型的拟合效果一般。

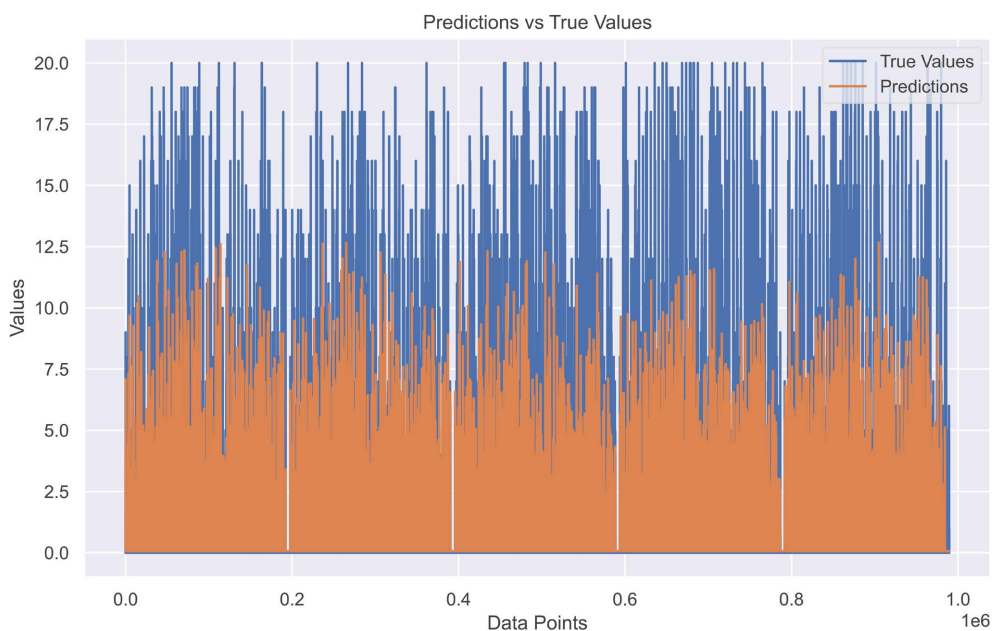
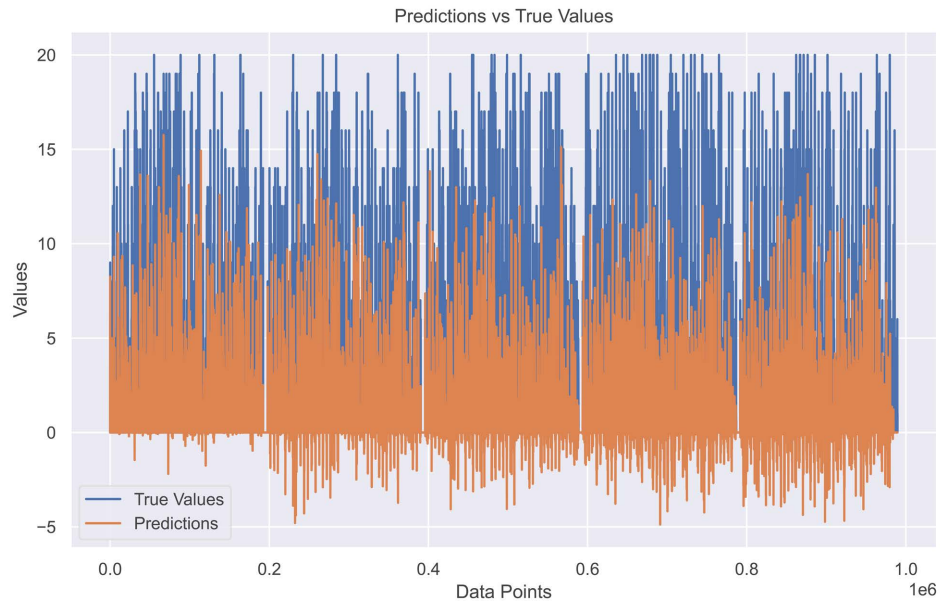


Figure 6. Comparison of LightGBM prediction results

图 6. LightGBM 预测结果比较

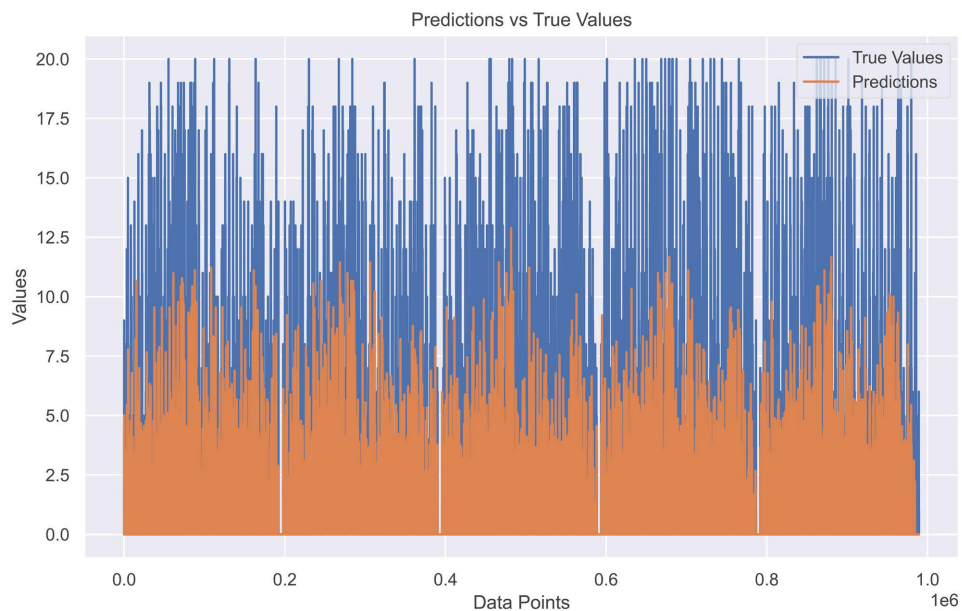
XGBoost 的训练均方根误差与验证均方根误差分别为：0.68、0.85。图 7 展示了 XGBoost 模型的预测结果与真实标签之间的关系，从图中可以看出 XGBoost 模型的拟合效果一般。



**Figure 7.** Comparison of XGBoost prediction results  
**图 7.** XGBoost 预测结果比较

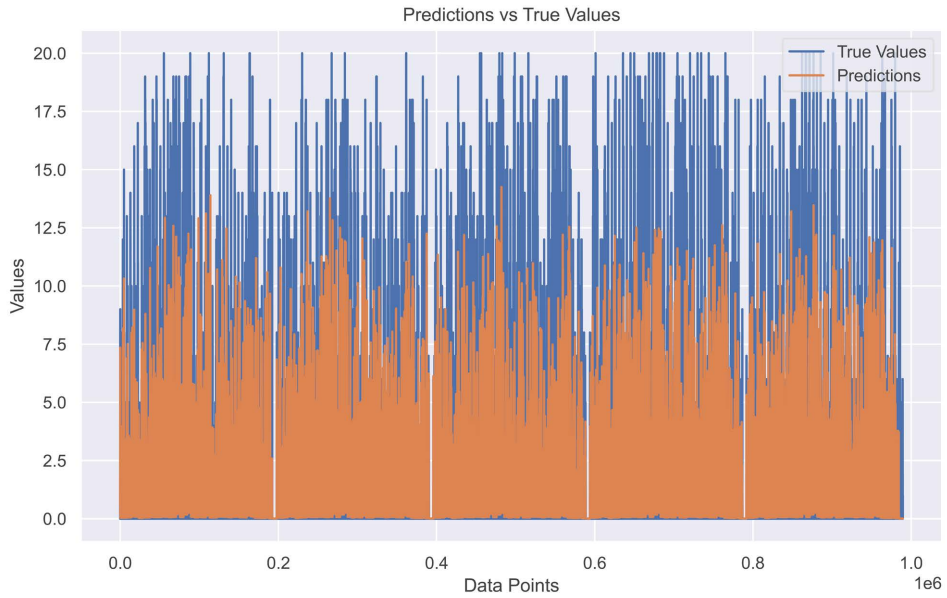
#### 4.3.2. KNN、线性模型和随机森林预测

由于数据太多，KNN 只取了 10 w 个数据来拟合模型。KNN 的训练均方根误差与验证均方根误差分别为：0.50、0.80。图 8 展示了 KNN 模型的预测结果与真实标签之间的关系，从图 8 中可以看出 KNN 的拟合效果较差。



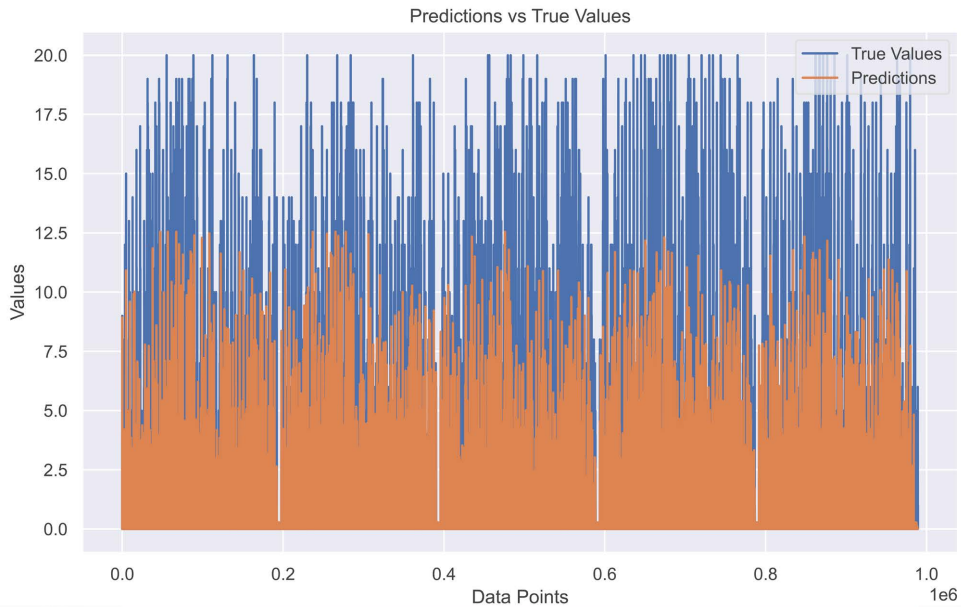
**Figure 8.** Comparison of KNN prediction results  
**图 8.** KNN 预测结果比较

线性模型的训练均方根误差与验证均方根误差分别为：0.73、0.78。图 9 展示了线性模型的预测结果与真实标签之间的关系，从图中可以看出线性模型的拟合效果一般。



**Figure 9.** Comparison of linear model prediction results  
**图 9.** 线性模型预测结果比较

随机森林的训练均方根误差与验证均方根误差分别为：0.70、0.78。图 10 展示了随机森林模型的预测结果与真实标签之间的关系，从图中可以看出随机森林模型的拟合效果一般。



**Figure 10.** Comparison of random forest prediction results  
**图 10.** 随机森林预测结果比较

#### 4.4. 深度学习模型预测

##### 4.4.1. LSTM 模型预测

本文使用的 LSTM 模型是一个具有三个 LSTM 层和三个全连接层的序列预测模型[10]。它的目标是根据输入的时间序列数据进行预测。模型使用均方误差作为损失函数，并使用 Adam 优化器进行模型训

练。表 1 是 LSTM 模型的结构。

**Table 1.** LSTM model structure  
**表 1.** LSTM 模型结构

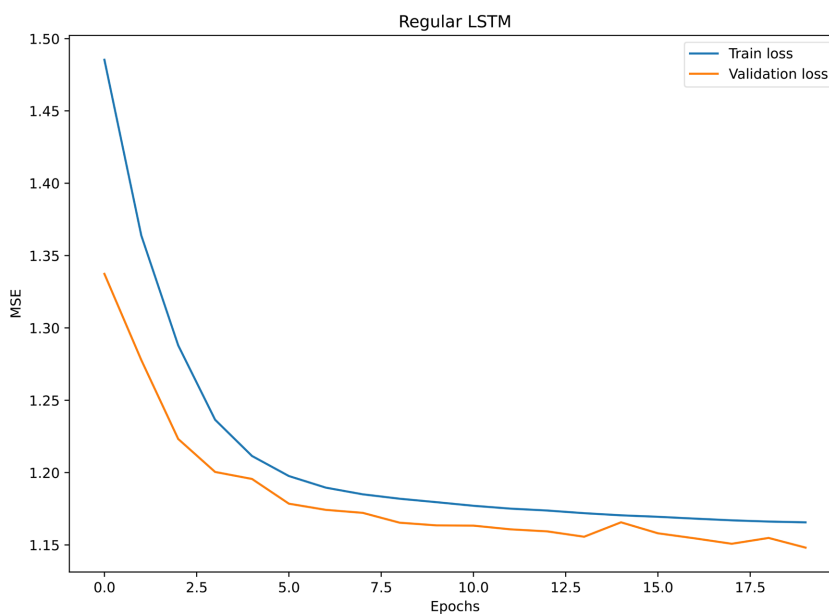
Layer (type)	Output Shape	Param
lstm (LSTM)	(None, 12, 10)	480
lstm_1 (LSTM)	(None, 12, 6)	408
lstm_2 (LSTM)	(None, 1)	32
dense (Dense)	(None, 10)	20
dense_1 (Dense)	(None, 10)	110
dense_2 (Dense)	(None, 1)	11

Total params: 1,061。

Trainable params: 1,061。

Non-trainable params: 0。

模型每个训练步骤使用了 128 个样本，学习率为 0.0001，总共训练了 20 次。得到了训练均方根误差与验证均方根误差分别为：1.08、1.07。图 11 展示了 LSTM 模型训练 20 次的训练误差与验证误差的曲线。



**Figure 11.** LSTM training error and validation error curves

**图 11.** LSTM 训练误差与验证误差曲线图

#### 4.4.2. Autoencoder 模型预测

模型由编码器(Encoder)和解码器(Decoder)组成，用于实现无监督的特征学习和数据重构。编码器部分包含了多个 LSTM 层，用于将输入的时间序列数据压缩为低维表示。每个 LSTM 层的输出作为下一层的输入，通过这种逐层压缩的方式，最终将时间序列数据编码为具有较低维度的表示[11]。

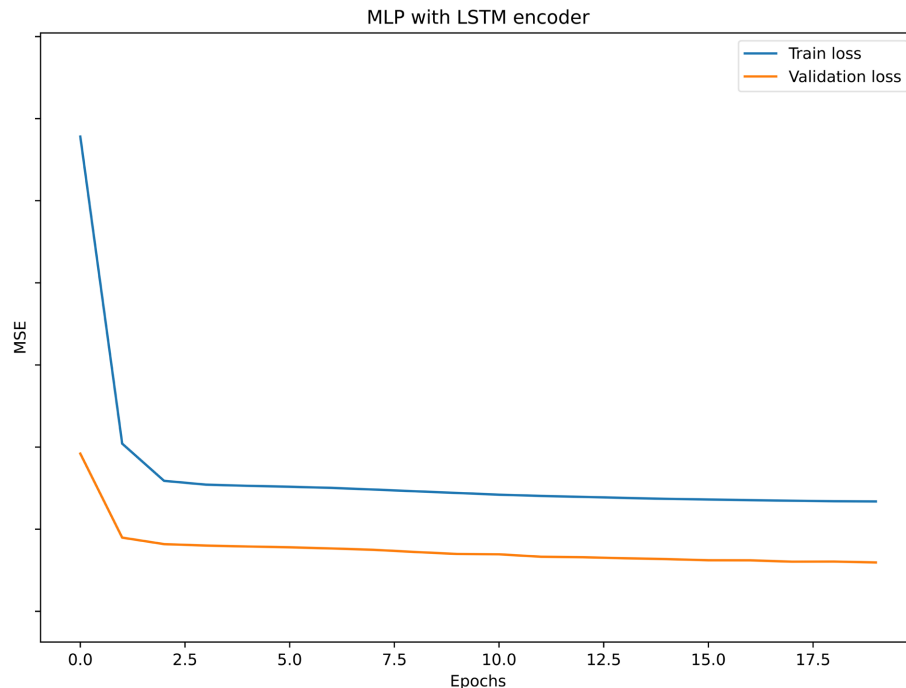


Figure 12. Autoencoder training error and validation error curves

图 12. Autoencoder 训练误差与验证误差曲线图

解码器部分同样包含了多个 LSTM 层，用于将编码后的低维表示解码为重构的时间序列数据。编码器的最后一个时间步的输出经过 RepeatVector 层进行复制，然后再通过一系列 LSTM 层进行解码和重构。

模型的输出是通过 TimeDistributed 层应用在 LSTM 层上的 Dense 层产生的，用于对每个时间步进行输出。在训练过程中，模型使用均方误差(MSE)作为损失函数，并使用 Adam 优化器进行参数优化。这个 Autoencoder 模型的目标是通过压缩和解压缩的过程，学习数据的重要特征表示，并实现对时间序列数据的重构和恢复。

使用编码器模型(encoder)对训练集和验证集的时间序列数据进行编码，得到了相应的编码后的时间序列数据。将编码后的数据存储在 train\_encoded 和 validation\_encoded 变量中。再将编码后的时间序列数据(train\_encoded 和 validation\_encoded)添加到原始的训练集和验证集中作为新的特征列。使用 train['encoded']和 valid['encoded']将编码后的数据添加到对应的数据集中。将原始的标签列从训练集和验证集中移除，得到了经过编码的特征数据集(X\_train\_encoded 和 X\_valid\_encoded)和对应的标签集(Y\_train\_encoded 和 Y\_valid\_encoded)。

最后创建了一个多层感知机(MLP)模型，该模型包含了几个 Dense 层。这个模型用于使用编码后的特征数据进行监督学习，目标是预测标签值。在训练过程中，使用均方误差(MSE)作为损失函数，并使用 Adam 优化器进行参数优化。得到了平均训练均方根误差与平均验证均方根误差分别为：1.10、1.09。

图 12 展示了 Autoencoder 模型训练 20 次的训练误差与验证误差的曲线。

#### 4.4.3. GRU 模型预测

本文使用的 GRU 模型由多个 GRU 层和全连接层组成。GRU 层用于捕捉时间序列数据中的长期依赖关系，通过记忆单元和门控机制来处理输入数据[12]。这些层的激活函数使用了 ReLU (线性整流单元)，用于引入非线性特征。最后的全连接层用于输出预测结果。表 2 是 GRU 模型的结构。

**Table 2.** GRU model structure  
**表 2.** GRU 模型结构

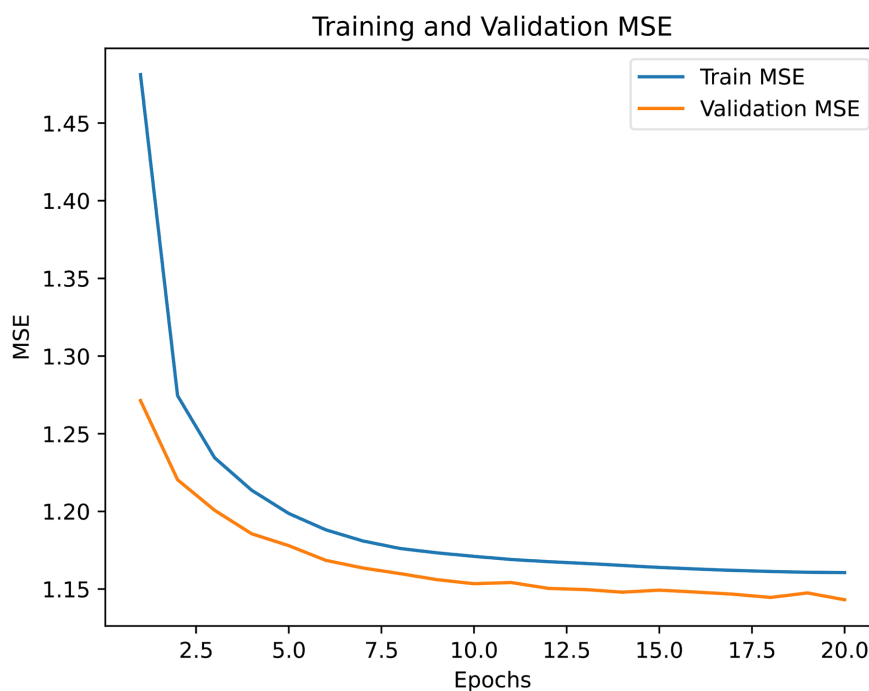
Layer (type)	Output Shape	Param
gru (GRU)	(None, 12, 10)	390
gru_1 (GRU)	(None, 12, 6)	324
gru_2 (GRU)	(None, 1)	27
dense_7 (Dense)	(None, 10)	20
dense_8 (Dense)	(None, 10)	110
dense_9 (Dense)	(None, 1)	11

Total params: 882。

Trainable params: 882。

Non-trainable params: 0。

模型每个训练步骤使用了 128 个样本，学习率为 0.0001，总共训练了 20 次。得到了训练均方根误差与验证均方根误差分别为：1.07、1.07。图 13 展示了 GRU 模型训练 20 次训练误差与验证误差的曲线。



**Figure 13.** GRU training error and validation error curves

**图 13.** GRU 训练误差与验证误差曲线图

#### 4.4.4. DeepAR 模型预测

DeepAR 模型是一个基于神经网络的序列预测模型，用于时间序列数据的预测和生成。模型的核心部分是一个多层的长短期记忆网络(LSTM)，用于学习时间序列数据中的时序模式和依赖关系[13]。

模型的输入维度为 `input_dim`，表示输入数据的特征维度。本文设定为 1，即每个时间步只有一个特征。模型的隐藏状态维度为 `hidden_dim`，表示 LSTM 层的隐藏状态的维度大小。本文设定为 64。



模型的输出维度为 `output_dim`，表示模型的预测输出的维度。本文设定为 1，即模型预测的是单个数值。

模型由 `num_layers` 层 LSTM 组成，每层 LSTM 具有相同的隐藏状态维度和输入维度。在 LSTM 层之后，通过全连接层(`nn.Linear`)将 LSTM 的输出映射到最终的预测输出维度。

在模型的前向传播中，输入数据经过 LSTM 层得到隐藏状态表示，然后通过全连接层进行映射得到预测输出。

模型的损失函数使用均方误差(MSE)损失函数，优化器采用 Adam 优化器。在训练过程中，使用批量梯度下降(batch gradient descent)进行模型参数的更新。得到了训练均方根误差与验证均方根误差分别为：1.25，1.21。图 14 展示了 DeepAR 模型训练 20 次训练误差与验证误差的曲线。

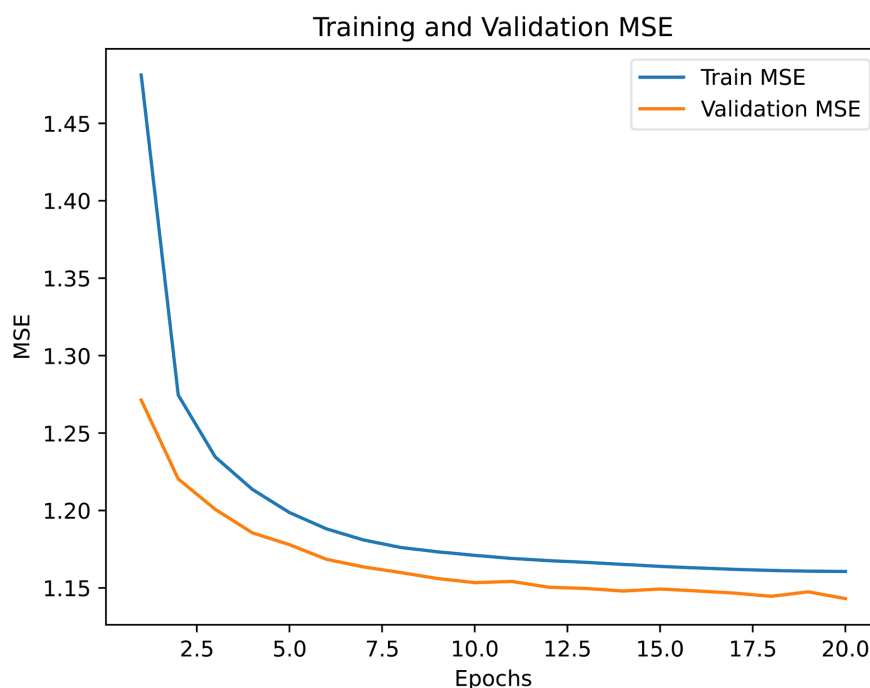


Figure 14. DeepAR training error and validation error curves

图 14. DeepAR 训练误差与验证误差曲线图

## 5. 总结

图 15 记录了各个模型的训练集 RMSE 和验证集 RMSE，得出以下结论：

1) LightGBM 表现最好：LightGBM 模型在验证集上达到了最低的 RMSE 值，表明其在预测产品月销量方面具有较高的准确性。它可能通过利用特征交互和树结构来捕捉时间序列数据中的非线性关系，从而提高了预测性能。

2) XGBoost 和随机森林表现相近：XGBoost 和随机森林模型的训练集和验证集 RMSE 值接近，说明它们在预测产品月销量方面具有类似的表现。这些模型都是基于集成学习的方法，通过组合多个弱模型来提高整体的预测能力。

3) KNN 模型表现较好：KNN 模型在训练集上取得了较低的 RMSE 值，但在验证集上略有增加。这可能表示 KNN 模型对训练集的拟合效果较好，但在未见过的数据上可能出现一定的过拟合。

4) 线性回归模型表现一般：线性回归模型在训练集和验证集上的 RMSE 值相对较高，可能是因为线

性模型难以捕捉时间序列数据中的非线性关系，它可能需要更复杂的模型来提高预测准确性。

5) LSTM、Autoencoder 和 GRU 模型表现相似：LSTM、Autoencoder 和 GRU 模型在训练集和验证集上的 RMSE 值接近，显示出相似的预测性能。这些模型都是基于深度学习的方法，能够处理时间序列数据中的长期依赖关系，但可能需要更多的训练数据和调优来提高性能。

6) VAR 和 DeepAR 模型表现较差：VAR 和 DeepAR 模型在训练集和验证集上的 RMSE 值较高，可能是因为它们在处理时间序列数据的特定问题上存在一定的限制或模型参数设置不够优化，这可能需要进一步的改进和调整来提高预测准确性。

模型	训练集RMSE	验证集RMSE
VAR	1.28	1.28
LignrGBM	0.71	0.78
XGBoost	0.68	0.85
随机森林	0.7	0.78
KNN	0.5	0.8
线性回归	0.73	0.78
LSTM	1.08	1.07
Autoencoder	1.1	1.09
GRU	1.07	1.07
deepar	1.25	1.21

Figure 15. RMSE predicted by different models  
图 15. 不同模型预测的 RMSE

## 参考文献

- [1] Fildes, R., Ma, S.H. and Kolassa, S. (2019) Retail Forecasting: Research and Practice. *International Journal of Forecasting*, **38**, 1283-1318. <https://doi.org/10.1016/j.ijforecast.2019.06.004>
- [2] 许姗姗. 基于机器学习的商品销售预测的研究[J]. 统计与管理, 2019(4): 49-52.
- [3] 梁雯, 秦胜杰, 张伟. 基于 VAR 模型的物流业与碳排放动态关系研究——以安徽省为例[J]. 资源开发与市场, 2016, 32(9): 1042-1046.
- [4] 孙铭. 基于 LightGBM 的超市商品销量预测[D]: [硕士学位论文]. 大连: 大连理工大学, 2022. <https://doi.org/10.26991/d.cnki.gdllu.2021.003534>
- [5] 周雨, 段永瑞. 基于聚类与机器学习的零售商品销量预测[J]. 计算机系统应用, 2021, 30(11): 188-194. <https://doi.org/10.15888/j.cnki.csa.008147>
- [6] 朱周帆, 郝鸿, 张立文. 基于机器学习与时间序列组合模型的中国汽车市场预测[J]. 统计与决策, 2020, 36(8): 177-180.
- [7] 张旭, 谌莉. 体育娱乐用品零售价格指数预测研究——基于深度学习算法的分析[J]. 价格理论与实践, 2021(9): 116-119+202. <https://doi.org/10.19851/j.cnki.CN11-1010/F.2021.09.286>
- [8] Ho, T.K. (2020) The Random Subspace Method for Constructing Decision Forests. *IEEE Transactions on Pattern Analy-*

- sis and Machine Intelligence*, **20**, 832-844. <https://doi.org/10.1109/34.709601>
- [9] Cortes, C. and Vapnik, V. (1995) Support-Vector Networks. *Machine Learning*, **20**, 273-279. <https://doi.org/10.1007/BF00994018>
- [10] 庞传军, 张波, 余建明. 基于 LSTM 循环神经网络的短期电力负荷预测[J]. 电力工程技术, 2021, 40(1): 175-180+194.
- [11] 张星宇, 陈卓, 黄印, 等. 基于多模态栈式混合自编码器的药物靶标相互作用预测[J/OL]. 计算机工程与应用, 2023: 1-9. <http://kns.cnki.net/kcms/detail/11.2127.TP.20231108.0843.002.html>
- [12] 耿阳, 王海龙, 张楠, 等. 基于误差补偿 LSTM-GRU 的综合能源系统多元负荷预测[J/OL]. 电气工程学报, 2023: 1-11. <https://kns.cnki.net/kcms/detail/10.1289.TM.20231103.1112.008.html>
- [13] 朱刚, 李文, 杜守国, 等. 基于深度学习模型 DeepAR 的时间序列预测及应用实例[J]. 电子商务, 2020(7): 83-86.