Research on express transport based on the ARIMA model

Yifan Lu1,*

1 Department of electronic Commerce, Nanjing University of Finance & Economics, Nanjing, China
* Corresponding Author Email: 2634080273@qq.com

Abstract. Based on the express transportation data of express companies, this paper makes an in-depth study of express transportation problems. First, the evaluation model to rank the importance of cities. Second, we predict the number of express deliveries between site cities with the time-series ARIMA model. Then, the random forest model of the CART decision tree was used to predict normal shipments and express delivery quantity. Next, the planning model is established based on the two parameters of railway transportation fixed cost and rated loading volume to plan the transportation scheme with the lowest cost of the express delivery company. Finally, the delivery requirements and model parameter estimates were estimated through the KDE model.

Keywords: Principal component analysis; Time-series ARIMA model; Random Forest model; Plan model; The KDE model for the probability density estimation.

1. Introduction

With the continuous development of China's economy and the continuous improvement of residents' living standards, online shopping has become an important way of consumption, and the logistics and transportation link has become an important link in the supply chain. Transportation cost accounts for more than half of the total cost of the logistics industry, which plays a vital role in the operation and development of the whole logistics chain, and can directly determine the total cost of logistics and the profits obtained. Therefore, clarifying the demand for express delivery, can improve the transportation efficiency of logistics, reduce logistics transportation costs, improve the quality of logistics transportation services, enhance the coverage of transportation network, integrate logistics transportation economic resources, and meet the higher requirements of the society for the transportation industry.

Therefore, based on the above background, it is of great significance to predict the quantity of express transportation demand, optimize the layout of warehouse stations of express companies, calculate transportation costs, plan transportation lines, and estimate express transportation demand. The analysis of express demand is the core of overcoming difficulties in the integration of economic resources of express transportation.

2. Building and solving the model of PCA

2.1. Section Headings

In order to measure the importance of the site city, we use the express transportation data between the site cities (shipping city-receiving city). We consider the receiving volume, delivery volume, express quantity growth or decrease trend, correlation, and get ten index factors, as shown in Table 1:
Table 1. Categorical interpretation of ten indicators

<table>
<thead>
<tr>
<th>Dimensionality</th>
<th>Index</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Receipts</td>
<td>Total amount received $X_1$</td>
<td>Each city can &quot;receive&quot; the total amount of express deliveries from all cities</td>
</tr>
<tr>
<td>Shipment volume</td>
<td>Total shipments $X_2$</td>
<td>Each city can &quot;ship&quot; the total amount of express delivery to all cities</td>
</tr>
<tr>
<td></td>
<td>The largest increase in shipments $X_3$</td>
<td>The maximum increase in daily shipments per city</td>
</tr>
<tr>
<td></td>
<td>Average increase in shipments $X_4$</td>
<td>The average increase in daily shipments for each city</td>
</tr>
<tr>
<td></td>
<td>Standard deviation of shipment volume increase $X_5$</td>
<td>The standard deviation of the increase in daily shipments per city</td>
</tr>
<tr>
<td>Trends in the volume of express deliveries</td>
<td>The largest increase in receipts $X_6$</td>
<td>The maximum increase in daily receipts per city</td>
</tr>
<tr>
<td></td>
<td>Average increase in receipts $X_7$</td>
<td>The average increase in daily receipts for each city</td>
</tr>
<tr>
<td></td>
<td>Standard deviation of the increase in receipts $X_8$</td>
<td>The standard deviation of the increase in daily receipts for each city</td>
</tr>
<tr>
<td></td>
<td>Total number of upstream shipping cities $X_9$</td>
<td>How many city goods can be &quot;received&quot; by each city</td>
</tr>
<tr>
<td>Correlation</td>
<td>Total number of downstream shipping cities $X_{10}$</td>
<td>How many cities can &quot;ship&quot; goods per city</td>
</tr>
</tbody>
</table>

The mathematical model we built through principal component analysis:

$$
\begin{align*}
F_1 &= a_{11}X_1 + a_{12}X_2 + \cdots + a_{110}X_{10} \\
\vdots
F_m &= a_{m1}X_1 + a_{m2}X_2 + \cdots + a_{m10}X_{10}
\end{align*}
$$  

(1)

Where $X_1, X_2, \ldots, X_{10}$ are the treated indicators of the original variables, and $F_1, \ldots, F_m (m < 6)$ are the new variables.

Through calculation, each column obtains the coefficient representing the principal components as the linear combination, and finally obtains an expression for the composite score

$$
F = 0.115X_1 + 0.113X_2 + 0.109X_3 + 0.106X_4 + 0.114X_5 + 0.107X_6 + 0.099X_7 + 0.116X_8 + 0.107X_9 + 0.098X_{10}
$$  

(2)

The data of each site city is brought into the above formula, and the comprehensive score of F value is calculated, and the comprehensive score and ranking of each site city can be obtained as shown in Table 2.

Table 2. Comprehensive score

<table>
<thead>
<tr>
<th>Ranking</th>
<th>City</th>
<th>Comprehensive score</th>
<th>Main component 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>L</td>
<td>2.658467</td>
<td>2.658467</td>
</tr>
<tr>
<td>2</td>
<td>G</td>
<td>2.417349</td>
<td>2.417349</td>
</tr>
<tr>
<td>3</td>
<td>V</td>
<td>1.557408</td>
<td>1.557408</td>
</tr>
<tr>
<td>4</td>
<td>X</td>
<td>0.637965</td>
<td>0.637965</td>
</tr>
<tr>
<td>5</td>
<td>J</td>
<td>0.422391</td>
<td>0.422391</td>
</tr>
<tr>
<td>6</td>
<td>R</td>
<td>0.364726</td>
<td>0.364726</td>
</tr>
<tr>
<td>7</td>
<td>W</td>
<td>0.35808</td>
<td>0.35808</td>
</tr>
<tr>
<td>8</td>
<td>O</td>
<td>0.324613</td>
<td>0.324613</td>
</tr>
<tr>
<td>9</td>
<td>Q</td>
<td>0.131501</td>
<td>0.131501</td>
</tr>
<tr>
<td>10</td>
<td>K</td>
<td>-0.06405</td>
<td>-0.06405</td>
</tr>
</tbody>
</table>
3. Model building based on the ARIMA algorithm

The ARIMA model is suitable for non-stationary time series data, where I represents the number of differences. The appropriate difference can make the original sequence become a stationary sequence before modeling the ARIMA model.

Step1: sequence judgment
To judge whether the model data we need to establish is a stationary sequence, if it is not a stationary sequence, we have to transform it to the stationary sequence, generally using the difference method. Then the smooth sequence is judged to be white noise sequence, if it is white noise sequence, the modeling is over, otherwise proceed to the next step.

Step2: Differential treatment
We treated the data differently and did not avoid the loss of information. Our number of differences is less than 2, which is less than the order of ARIMA \((p, d, q)\). AR represents an autoregressive process of order \(p\) and MA represents a moving average process of order \(q\).

\[
\begin{align*}
\varphi_1 z_{t-1} + \varphi_2 z_{t-2} + \cdots + \varphi_p z_{t-p} + \alpha_t - \theta_1 a_{t-1} - \cdots - \theta_q a_{t-q} \\
\varphi_p &= 1 - \varphi_1 B - \varphi_2 B^2 - \cdots - \varphi_p B^p \\
\theta_q &= 1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q
\end{align*}
\]

Step3: Fixed order
We use the BIC criterion method to determine the order of the model as order 2, and use the maximum likelihood method to estimate the parameter values, and the formula is:

\[
BIC(k) = \ln \hat{\sigma}_k^2 + \frac{2k \ln N}{N}
\]

Finally, the model, ARIMA \(p, d, q\) model are the expansion of ARMA \((p, q)\) model, where \(L\) is the lag operator, \(d \in \mathbb{Z}, d > 0\).

\[
(1 - \sum_{i=1}^p \varphi_i L^i)(1 - L)^d X_t = (1 + \sum_{i=1}^q \theta_i L^i) \varepsilon_t
\]

After establishing and solving the model, we get the number of express transportation between each station city and the total amount of express transportation between all "delivery-receiving" station cities as shown in Table 3:

<table>
<thead>
<tr>
<th>The number of express shipments between ship-to-receive cities</th>
<th>Total number of express shipments between all ship-to-receive cities</th>
</tr>
</thead>
<tbody>
<tr>
<td>M-U 103</td>
<td>12156</td>
</tr>
<tr>
<td>Q-V 43</td>
<td></td>
</tr>
<tr>
<td>K-L 59</td>
<td></td>
</tr>
<tr>
<td>G-V 573</td>
<td></td>
</tr>
<tr>
<td>V-G 463</td>
<td></td>
</tr>
<tr>
<td>A-Q 95</td>
<td>12693</td>
</tr>
<tr>
<td>D-A 43</td>
<td></td>
</tr>
<tr>
<td>L-K 78</td>
<td></td>
</tr>
</tbody>
</table>

4. Model building using the CART decision tree random forest algorithm

Random forest is an integrated learning method whose construction process includes the following steps and formulas:

Step 1: Bootstrap sampling
Bootstrap Sampling is a non-parametric statistical method. The basic idea forms multiple samples by sampling them back from the original data, and then uses these samples to estimate the distribution of statistics. The core steps are to:
N samples are extracted from the original data set, and the mean, standard deviation and other statistics are calculated based on these n samples to obtain the estimates of multiple statistics. These estimates were used to calculate confidence intervals, statistics, standard errors, etc.

Step 2: Gini coefficient $GiniIndex(D)$

The Gini coefficient is used to measure the probability of being misclassified into other categories when randomly sampling a sample, and the information gain is used to measure the purity improvement brought by a new node through feature segmentation:

$$Gini(D) = \sum_{i=1}^{n} p(x_i) \cdot (1 - p(x_i)) = 1 - \sum_{i=1}^{n} p(x_i)^2$$  \hspace{1cm} (6)

Where $p(x_i)$ is the probability of classified $x_i$ occurrence, and $n$ is the number of classifications. $Gini(D)$ Reflects the probability of inconsistent class labeling of randomly drawn two samples from dataset $D$. Therefore, the smaller the $Gini(D)$, the higher the purity of the dataset $D$.

For sample $D$, the number is $|D|$, according to whether the feature $A$ take a possible value $a$, the sample $D$ is divided into two parts $D_1$ and $D_2$.

$$\begin{cases} D_1 = (x, y) \in D|A(x) = a \\ D_2 = D - D_1 \end{cases}$$  \hspace{1cm} (7)

Under the condition of the attribute $A$, the Gini coefficient of the sample $D$ is defined as:

$$GiniIndex(D|A = a) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$  \hspace{1cm} (8)

Step 3: The CART decision tree algorithm

Calculate the Gini coefficient of each feature, and select the feature with the highest value as the splitting standard of the nodes, and select the best feature for segmentation.

For discrete variables, children are created according to the characteristic values; for continuous variables, the data set is divided into two parts according to the values.

Repeat steps a and b recursively until a stopping condition is reached, such as reaching a predetermined maximum depth or if the number of children is less than a certain threshold.

Where, at each node, the CART algorithm will compute the splitting criteria of all possible features and select the features with the largest purity gain of each child node to split.

Step 4: Characteristic subset

The feature subset can reduce the risk of overfitting and variance of a single decision tree and improve the generalization ability of the random forest whole. Each decision tree in the random forest adopts a different subset of features, which makes the independence of each decision tree more obvious and makes different predictions.

For each node, $m$ subsets of features were randomly selected from the original features and used to generate candidate split points. Feature subset selection formula:

$$m = \sqrt{p}$$  \hspace{1cm} (9)

Step 5: Optimize weak classifier to strong classifier with voting classifier

We trained several classifiers, each achieving over 80% accuracy. These classifiers are logistic regression classifier, SVM classifier, random forest classifier, KNN classifier, and so on, as shown in Figure 1:

![Fig. 1 Schematic diagram of training four classifiers](image_url)
The categories predicted by each classifier were aggregated and then the most voted category was selected. As shown in Figure 2, this majority vote classifier is called a hard vote classifier.

Such voting classifiers can often be more accurate than a single optimal classifier. And even if each classifier is a weak classifier, the result of the ensemble can still be a strong classifier.

On the basis of the above data classification using the random forest algorithm, the data is classified according to whether it can be delivered normally. According to the data of the time of the normal shipment, we analyzed the ARIMA model. $ARIMA(p, d, q)$ model:

$$
(1 - \sum_{i=1}^{p} \phi_i L^i)(1 - L)^d X_t = (1 + \sum_{i=1}^{d} \theta_i L^i)\epsilon_t
$$

Where, $L$ is the lag operator, $\phi$ represents the prediction coefficient of the autoregressive model, $\theta$ represents the coefficient of the moving average model, $p, q$ is the order, $d \in \mathbb{Z}$, $d > 0$.

After classification prediction of random forest model and quantitative prediction of ARIMA model, we obtained the number of express transport between each station city and the total amount of express transport between all station cities as shown in Table 4:

<table>
<thead>
<tr>
<th>Ship-to-receive site city</th>
<th>Whether it can be shipped normally (Fill in Yes or No)</th>
<th>Number of express shipments</th>
</tr>
</thead>
<tbody>
<tr>
<td>I-S</td>
<td>No</td>
<td>-</td>
</tr>
<tr>
<td>M-G</td>
<td>Yes</td>
<td>44</td>
</tr>
<tr>
<td>S-Q</td>
<td>Yes</td>
<td>100</td>
</tr>
<tr>
<td>V-A</td>
<td>Yes</td>
<td>122</td>
</tr>
<tr>
<td>Y-L</td>
<td>Yes</td>
<td>34</td>
</tr>
<tr>
<td>D-R</td>
<td>No</td>
<td>-</td>
</tr>
<tr>
<td>J-K</td>
<td>Yes</td>
<td>183</td>
</tr>
<tr>
<td>Q-O</td>
<td>No</td>
<td>-</td>
</tr>
<tr>
<td>U-O</td>
<td>No</td>
<td>-</td>
</tr>
<tr>
<td>Y-W</td>
<td>No</td>
<td>-</td>
</tr>
</tbody>
</table>

5. *Model building based on the planning model*

In order to accurately state the corresponding constraints and the derivation of the value of the objective function, the following modeling takes the A-E route as an example to generalize to the whole route network, planning the optimal path with the least cost.

Step 1: Mark the adjacent cities and build a data matrix

All the transportation processes can only be carried out between the adjacent cities of a given route. In order to mark whether the two cities are adjacent, we use the 0-1 marker and solve it through the matrix:

$$
city(A) = \begin{pmatrix}
city(x_{1,1}) & \cdots & city(x_{1,25}) \\
\vdots & \ddots & \vdots \\
city(x_{25,1}) & \cdots & city(x_{25,25})
\end{pmatrix}
$$
If adjacent, then, \( city(x_{ij}) = 1 \), and if not adjacent, then \( city(x_{ij}) = 0 \).

**Step 2: Analysis of single routes between city pairs**

Here first set the policy variable as a route (example A-E route), here is the 0-1 matrix:

\[
route^{1}_{AE} = \begin{pmatrix}
    city(x_{1,1}) & \cdots & city(x_{1,25}) \\
    \vdots & \ddots & \vdots \\
    city(x_{25,1}) & \cdots & city(x_{25,25})
\end{pmatrix}
\]  

(12)

\[
transport^{1}_{AE} = \begin{pmatrix}
    y_{1,1} & \cdots & y_{1,25} \\
    \vdots & \ddots & \vdots \\
    y_{25,1} & \cdots & y_{25,25}
\end{pmatrix}
\]  

(13)

At the same time, because our transportation path can only choose each step, to go to the adjacent city, constraints:

\[
\text{city}(y_{ij}) \leq \text{city}(x_{ij})
\]  

(14)

This constraint can reduce the time complexity of the operation, reducing the possible 25 * 25 routes to about 30 routes.

**Step 3: Analysis of the transportation process of a single route**

First of all, for the middle passing city, the delivery volume must be equal to the shipment volume:

\[
\Sigma_i y_{ij} = \Sigma_j y_{ji}
\]  

(15)

Secondly, for the same route, you can only enter and exit in the corresponding city once:

\[
\Sigma_i \text{city}(y_{ij}) = \Sigma_j \text{city}(y_{ji}) = 1
\]  

(16)

For the starting city (for example, city A), there is no path to arrive, and there is only one path to start:

\[
\Sigma_i y_{1j} = 0, \Sigma_j y_{1j} = 1
\]  

(17)

Similarly, for the destination city (example city E), there is only one path to arrive and no path to start:

\[
\Sigma_i y_{5j} = 1, \Sigma_j y_{5j} = 0
\]  

(18)

In this way, we complete the decision variable for the first path from A to E. Similarly, A-E can have at most five paths, and we also need to define the first \( k \) (\( k = 1,2,3,4,5 \)) route in the same way.

\[
route^{k}_{AE} = \begin{pmatrix}
    city(y^{k}_{1,1}) & \cdots & city(y^{k}_{1,25}) \\
    \vdots & \ddots & \vdots \\
    city(y^{k}_{25,1}) & \cdots & city(y^{k}_{25,25})
\end{pmatrix}
\]  

(19)

\[
transport^{k}_{AE} = \begin{pmatrix}
    y^{k}_{1,1} & \cdots & y^{k}_{1,25} \\
    \vdots & \ddots & \vdots \\
    y^{k}_{25,1} & \cdots & y^{k}_{25,25}
\end{pmatrix}
\]  

(20)

**Step 4: Analysis of the total amount of express transportation related to the routes**

During the transportation journey, we assume that the quantity of express delivery is \( quantity_{AE} \). Since the amount of goods received must be equal to the quantity of goods sent, the total amount will not change in the middle:

For starting cities (for example, City A):

\[
\Sigma_k \Sigma_j y^{k}_{1j} = quantity_{AE}
\]  

(21)
For the destination city (for example, City E):

$$\sum_k \sum_i y_{ki}^j = \text{quantity}_{AE}$$  \hspace{1cm} (22)

Step 5: Costing

Transportation cost calculation formula for each route:

$$\text{Cost} = \text{Fixed Cost} \times \left[ 1 + \left( \frac{\text{Actual loading volume}}{\text{Rated loading capacity}} \right)^3 \right]$$  \hspace{1cm} (23)

Among them, the fixed cost and the rated loading volume are both parameters, and the actual loading volume of route $k$ is $\sum_k y_{ij}^k$. Then, the actual loading volume from the $i$th city to the $j$th city is:

$$\left( \sum_k y_{ij}^k \right)_{AB} + \left( \sum_k y_{ij}^k \right)_{BC} + \cdots + \left( \sum_k y_{ij}^k \right)_{XY}$$  \hspace{1cm} (24)

To bring the actual loading volume into the above cost calculation formula, it is necessary to sum up all the adjacent roads to obtain our objective function:

$$\sum_{\text{route}} \left[ \left( \sum_k y_{ij}^k \right)_{AB} + \cdots + \left( \sum_k y_{ij}^k \right)_{XY} \right] \cdot \text{fixed cost}(x_{ij})$$  \hspace{1cm} (25)

After the establishment and adjustment of the optimization model, we solved the equation of the objective function under the above constraints by using lingo software, planned the route with the lowest cost, and calculated the lowest transportation cost as shown in Table 5:

<table>
<thead>
<tr>
<th>Date</th>
<th>Lowest shipping costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Day 1</td>
<td>16432.06</td>
</tr>
<tr>
<td>Day 2</td>
<td>22964.94</td>
</tr>
<tr>
<td>Day 3</td>
<td>18677.11</td>
</tr>
<tr>
<td>Day 4</td>
<td>14567.27</td>
</tr>
<tr>
<td>Day 5</td>
<td>13786.90</td>
</tr>
</tbody>
</table>

6. Model building, test and solution based on the KDE algorithm

Step 1: Calculate the kernel function, and the kernel function selects the Gaussian curve $K(u)$:

$$K(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}$$  \hspace{1cm} (26)

N samples were selected for each city to establish the probability density function $Pr(x)$:

$$Pr(x_t) = \frac{1}{N} \sum_{i=1}^{N} \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi} \sigma_j} e^{-\frac{(x_t-x_i)^2}{2 \sigma_j^2}}$$  \hspace{1cm} (27)

Step 2: Further process the former scenic spots detected in the first stage to reduce the false detection rate

Calculate the maximum probability that the pixel value belongs to the neighborhood distribution:

$$P_N(x_t) = \max Pr(x_t | B_y)$$  \hspace{1cm} (28)

We have made nuclear density estimates for fixed demand and non-fixed demand in each city:

$$\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^{n} K \left( \frac{x-x_i}{h} \right)$$  \hspace{1cm} (29)

Step 3: On this basis, we conducted a chi-square test. This statistic is defined as:
Step 4: Statistical analysis

Set the random variable \( \eta_j = \sum_{i=1}^{100000} \xi_{ij}, j = 1, 2, \ldots, 5 \). \( \eta_j \) represents the total number of \( j \) out of 100,000 express goods, since \( \xi_{ij}, j = 1, 2, \ldots, 100000 \) is relatively independent, that is, the unfixed demand constant between cities is independent of each other, \( \eta_j \) follows a binomial distribution, namely:

\[
P\{\eta_j = k\} = \binom{100000}{k} \cdot (p_j)^k \cdot (1 - p_j)^{100000-k}, \quad j = 1, 2, \ldots, 5
\]  

(31)

You can also get the information that:

\[
E(\eta_j) = 100000p_{j}, j = 1, 2, \ldots, 5
\]  

(32)

\[
D(\eta_j) = 100000p_{j} \cdot (1 - p_{j}), j = 1, 2, \ldots, 5
\]  

(33)

We take its mathematical expectation as the non-fixed demand mean:

\[
E(\frac{50%}{\eta_j}) = \frac{1}{2} E(\eta_j) = 50000p_{j}, j = 1, 2, \ldots, 5
\]  

(34)

Based on the above model algorithm, we derive the fixed demand constant for the specified quarter, the specified "ship-receive" site cities, and the sum of the fixed demand constant for all the "ship-receive" city pairs in the quarter, and derive the data related to the non-fixed demand, as shown in Table 6.

<table>
<thead>
<tr>
<th>Quarter</th>
<th>Q3 2022 (July-September)</th>
<th>Q1 2023 (January-March)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ship-to-receive site city pair</td>
<td>V-N</td>
<td>V-Q</td>
</tr>
<tr>
<td>Fixed demand constant</td>
<td>21</td>
<td>24</td>
</tr>
<tr>
<td>Non-fixed demand mean</td>
<td>78.48</td>
<td>86.18</td>
</tr>
<tr>
<td>Non-fixed demand standard deviation</td>
<td>36.75</td>
<td>38.43</td>
</tr>
<tr>
<td>The sum of fixed demand constants</td>
<td>1840.32</td>
<td>2678.58</td>
</tr>
<tr>
<td>The sum of the non-fixed mean requirements</td>
<td>5836.86</td>
<td>7656.64</td>
</tr>
<tr>
<td>The sum of the standard deviations of the non-fixed demand</td>
<td>2699.18</td>
<td>3634.33</td>
</tr>
</tbody>
</table>

7. Evaluation of the model

7.1. Advantage

Principal component analysis can greatly reduce the number of features in the data set, reduce the computational complexity and improve the efficiency of machine learning algorithm, and find the principle component of maximizing variance to explain the maximum variability of data, and improve sampling efficiency.

The ARIMA algorithm model is simple and requires only endogenous variables without the aid of other exogenous variables.

Random forest algorithm can handle high dimensional features, suitable for large data sets, and has good generalization ability; random forest can automatically handle correlation and interaction, so it is more accurate and stable than a single decision tree; when the data set contains a lot of noise or missing values, random forest can maintain good performance; output feature importance score, convenient to understand the importance of each feature for classification or regression.

The advantages of KDE algorithm are wide applicability for various types of data distribution, and can adjust the prediction results; high accuracy, more detailed modeling and description of raw data,
so the prediction results have good accuracy; using cross-validation technology to solve the problems of missing data, and outliers in the data set.

7.2. Weakness

PCA is not conducive to the specific understanding and interpretation of data; it may lose some information or cause some errors; and choose a more suitable dimension reduction method for nonlinear data sets.

The ARIMA algorithm requires that temporal data is stable or after poor differentiation; it essentially only captures linear relationships, not nonlinear relationships.

The random forest algorithm has a long training time when processing large-scale datasets; because the randomness model may overfit some data sets, we need to set parameters reasonably to overcome this problem; for high-dimensional data, the performance of random forest becomes worse with the number of features, resulting in "dimensional disaster".

KDE algorithm has high calculation cost, which requires the kernel function and the data distribution when the data dimension increases; the analysis results have certain randomness, and different parameter adjustment and kernel function selection may get different results.

References


