

# 五一数学建模竞赛

## 承 诺 书

我们仔细阅读了五一数学建模竞赛的竞赛规则。

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与本队以外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛规则的，如果引用别人的成果或其它公开的资料（包括网上查到的资料），必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺，严格遵守竞赛规则，以保证竞赛的公正、公平性。如有违反竞赛规则的行为，我们愿意承担由此引起的一切后果。

我们授权五一数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

参赛题号（从 A/B/C 中选择一项填写）：                     B                    

参赛队号：                     T34377523047                    

参赛组别（研究生、本科、专科、高中）：                     本科                    

所属学校（学校全称）：                     青海大学                    

参赛队员： 队员 1 姓名：                     李根                    

队员 2 姓名：                     彭杰辉                    

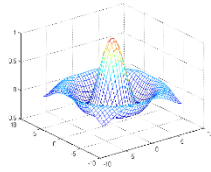
队员 3 姓名：                     姜雨秋                    

联系方式： Email:           1462890879@qq.com           联系电话：           13456015640          

日期：           2023           年           4           月           29           日

（除本页外不允许出现学校及个人信息）

# 五一数学建模竞赛



## 题 目： 关于站点城市的物流评估、预测与规划

关键词：随机森林-TOPSIS 模型 GRU-LSTM 模型 运输方案规划 需求分析

### 摘 要：

快递运输的物流水平与人们的日常生活以及物流企业的经济效益息息相关，如何判断和优化物流结构的各个环节将直接影响到生活水平和企业收益的提高。本文通过随机森林-TOPSIS 方法、GRU-LSTM 时序预测模型、最低成本运输方案规划模型、概率分布估计法等方法，建立了站点城市重要性评价方法，预测了站点城市发货可行性与发货数量，求解了最低运输成本的运输方案，并分析了快递的需求特征。

针对问题一，对附件 1 中的数据进行特征提取，综合考虑发货情况、收货情况、快递数量变化趋势、城市关系相关性、供需稳定性五大方面，共提取出 10 个站点城市评价指标，构建站点城市重要程度评价指标体系。利用随机森林-TOPSIS 评价模型进行样本训练，计算评价指标权重，识别出发货总量、总收货量和平均下游城市数量为关键评价指标，其权重分别达到 17.16%、14.57%和 11.46%，随后计算各站点城市的综合得分，判断出了五个重点城市 L、G、V、D、N。

针对问题二，搭建带有 GRU 的 LSTM 模型，并引入其他机器学习模型进行预测误差的比对，验证了 GRU-LSTM 模型的预测优良性，然后对确定日期各个城市对之间的货物运输量进行预测，并加总得到所有城市总快递运输数量的预测值。

针对问题三，对补充残缺日期数据后，将数据转化为是非变量，由连续性时序预测(数值预测)问题转变为离散型时序预测(事件预测)问题以提高预测准确性，通过对 GRU-LSTM 模型的训练得到城市对在确定日期发货可行性的预测值，并通过与可行性阈值 0.5 的比较，判断该城市对在确定日期能否正常发货。

针对问题四，本文以运输成本最小化为目标函数，以邻接站点、平衡条件、正向流量、有限步长为约束条件构建最小成本的非线性规划模型，通过 MATLAB 求解出确定日期的最低运输成本及最佳运输方案。

针对问题五，在对数据剔除异常值后按照年份、季节和收发货城市对进行交叉分组，通过文献研究构造出固定需求常数的经验计算公式，并采用 $3\sigma$ 准则来对固定需求常数进行准确性检验，检验结果表示固定需求常数的准确性较高，可以适用各季度数据。在对各组数据扣除固定需求常数后得到各组的非固定需求数据，然后采用最大似然估计的方法得到对非固定需求分布的均值和标准差的估计结果。

## 一、问题重述

随着互联网科技的发展，网络购物逐渐成为一种重要的消费方式，带动着快递服务需求飞速增长，为我国经济发展做出了重要贡献。准确地预测快递运输需求数量对于快递公司布局仓库站点、节约存储成本、规划运输线路等具有重要的意义，也有利于人民生活水平的提高。

因此，本文根据题目所提供的数据资料，建立数学模型以解决下面 5 个问题：

1.充分考虑收货量、发货量、数量趋势、相关性、供需稳定性等相关指标，建立站点城市重要程度评估模型，用以评价城市重要程度，筛选出排名前五的城市。

2.根据一定时期内“发货-收货”站点城市之间的总快递运输数量的历史数据，建立预测模型，对未来确定日期的“发货-收货”站点城市之间的总快递运输数量进行预测。

3.考虑突发因素导致快递无法发出的影响，建立时间序列模型，通过数据分析的结果对确定日期下城市对能否正常发货状况进行判断并预测各城市间快递运输量及所有城市的运输总量。

4.根据站点城市间的铁路运输网络及成本计算公式，在每个“发货-收货”站点城市对之间使用的路径数不超过 5 条的情况下，制定实现最低运输成本的运输方案，并给出确定日期的最低运输成本。

5.计算各城市快递数量固定需求常数，并在得到固定需求常数的基础上研究非固定需求的分布函数，并对其均值、标准差等特征值进行估计。

## 二、问题分析

### 2.1 问题一的分析

问题一要求对站点城市重要程度做出评估，并筛选出重要程度排名前 5 的站点城市。为了科学合理地进行综合评估，本问做法分为以下两步：

1.构建站点城市重要城市评价指标体系。根据附件 1 所提供的数据，综合考虑城市的物流强度和联系程度，对题目进行解读并对附件 1 数据进行分析，并结合文献分析法进行筛选，最终从发货情况、收货情况、快递数量变化趋势、城市关系相关性、供需稳定性五方面构建十项较为合理及准确的评估指标：各城市发货总量 $C_i$ 、发货持续性 $S_i$ 、各城市收货总量 $R_i$ 、收货持续性 $A_i$ 、各城市快递发货量长期变动趋势 $P_i$ 、各城市快递收货量长期变动趋势 $Q_i$ 、平均上游城市数量 $\bar{M}_i$ 、平均下游城市数量 $\bar{O}_i$ 、供给稳定性 $I_i$ 、需求稳定性 $U_i$ 。

2.站点城市重要程度评价。本文将随机森林与 TOPSIS 算法进行耦合，构建随机森林 TOPSIS 综合评价模型，对站点城市重要程度进行评价，首先进行训练样本集抽样，构建决策树随机森林，利用基尼指数来计算各类指标的信息贡献度，最终获得指标权重。随后将权重信息集成至 TOPSIS 算法中计算出所有指标的权重赋值，并计算各城市的综合得分，根据得分选出排名前五的重要城市。

## 2.2 问题二的分析

问题二要求对历史货运量数据进行分析，并在此基础上对未来两日的各“发货-收货”站点城市之间快递运输数量以及所有“发货-收货”站点城市之间的总快递运输数量进行预测。本题做法分为以下两步：

- 1.观察到数据随时间变动有较强的关联性，故本文选用 Cho 等人提出的带有 GRU 的 LSTM 模型，并引入多个机器学习模型与带有 GRU 的 LSTM 模型的预测优良性进行比对，以此来验证模型选择的准确性。

- 2.在带有 GRU 的 LSTM 模型下对附件 1 的数据进行训练，最终得出未来时期各“发货-收货”站点城市之间快递运输数量以及所有“发货-收货”站点城市之间的总快递运输数量的预测值。

## 2.3 问题三的分析

问题三要求对各城市对之间快递能否正常运输的情况进行判断，并预测可正常“发货-收货”的站点城市对之间的快递运输数量，本题做法分为以下两步：

- 1.由于带有 GRU 的 LSTM 模型对于时序预测的优良性已在问题二中展示，且其具有优秀的长短期记忆能力，故在本问中对缺失日期数据进行补全后继续沿用 GUR-LSTM 时序预测模型进行发货可行性预测。

- 2.为减小预测的误差，把补全后的数据转化为是非变量，将连续性时序预测(数值预测)问题转变为离散型时序预测(事件预测)问题以提高预测的准确性，训练模型以进行对发货可行性的预测，并设定阈值 0.5 作为判断标准，当可行性预测值大于 0.5 时即可正常发货，反之则不能。对于可正常发货城市的快递运输数量的预测继续沿用问题 2 的步骤并使用转化前的连续性数据进行 GUR-LSTM 模型的训练和预测。

## 2.4 问题四的分析

根据站点城市间的铁路运输网络及成本计算公式，在每个“发货-收货”站点城市对之间使用的路径数不超过 5 条的情况下，制定实现最低运输成本的运输方案，并给出确定日期的最低运输成本。本题做法如下所示：

基于图论中的费用最小化知识，对路线费用问题做出非线性规划，以成本最小化为目标函数，以邻接站点、平衡条件、正向流量、有限步长为约束条件构建最小成本的运输方案规划模型，并通过 MATLAB 进行求解出最低运输成本。

## 2.5 问题五的分析

问题五要求按季度估计固定需求常数，并求解每一季度下各城市的非固定需求的概率分布，并估计其均值及标准差，本题做法分为以下两步：

- 1.对附件 2 数据剔除异常值(由于突发因素造成的非正常发货与零发货数据)，按照年份、季度以及“发货-收货”城市进行交叉分组，并求解每一组“发货-收货”站点城市的运输量平均值、运输量最小值、运输量标准差。在此基础上利用经验计算公式求解每一组数据的季度固定需求常数。

- 2.将附件 2 数据扣除每一组对应的固定需求常数，即可得到每个日期下各站点城市的非固定需求，根据中心极限定理的判断，非固定需求数据近似于正态分布，由于最大似然估计比其他估计方法更加简便易行，收敛性质更好并且预测结果也更准确，同时考虑到方法的可操作性，本文采用最大似然估计方法对非固定需求均值及标准差进行估计，最后加总得到本问结果。

### 三、问题假设

1. 无快递运输往来的站点城市之间不存在显著的相关性。
2. 站点城市之间正常发货或收货的与否与时间相关，可通过时间序列进行预测。
3. 不对快递重量和大小进行区分，令每件快递的重量为单位 1。仅考虑运输成本，而不考虑中转等其它成本。

### 四、符号说明

符号	说明	单位
$X_{ij}^d$	日期 d 内 i 城市向 j 城市的快递发货量	件
$Y_{ij}^d$	日期 d 内 i 城市从 j 城市取得的快递收货量	件
$m_{ij}^d$	日期 d 内收到城市 i 快递发货的城市总和	个
$n_{ij}^d$	日期 d 内城市 i 收到快递发货的城市总和	个
$\bar{X}_i$	城市 i 向其他城市的平均发货量	件
$\bar{Y}_i$	城市 i 取得各城市发货的平均收货量	件
$VMI_j$	权重得分	%
$b_{st}$	城市 s 与城市 t 之间运输的固定成本	/
$f_{st}^d$	日期 d 内城市 s 与城市 t 之间运输的货物量	件
$P_{st}$	各城市运输路径/路径数	条
$W_{ij}^d$	日期 d 从 i 城市向 j 城市快递能否正常发货	/
$t_{ij}$	每一季度下各城市与关联城市得快递运输数量的最小值	件

### 五、问题一模型的建立与求解

#### 5.1 数据特征提取

本部分根据附件一所提供的日期、发货城市、收货城市、货物运输量等数据，结合相关文献和题目信息，本文认为判断站点城市的重要程度要从发货情况、收货情况、快递数量变化趋势、城市关系相关性、供需稳定性五个方面<sup>[1]</sup>进行权重赋值和综合评估。本文就上述五个方面提取 10 个特征，用以衡量站点城市的重要程度。

### 5.1.1 发货情况

(1) 各城市发货总量 $C_i$ : 在过去一段时间内各个城市从本城市发往与其有相关快递往来的城市的快递量总和, 代表了各城市的快递发货量总和, 能够有效反映各城市的快递发货能力。

$$C_i = \sum_{d=1}^N \sum_{j \in \{A \dots Z\}} X_{ij}^d \quad (5-1)$$

其中,  $N$  表示过去时间的总天数 (273 天),  $d$  表示发货日期,  $i, j = \{A, B \dots, Y\}$ ,  $X_{ij}^d$  表示在日期  $d$ ,  $i$  城市向  $j$  城市的快递发货量。

(2) 发货持续性 $S_i$ : 在过去一段时间内各个城市快递发货量不为零的天数与过去时间天数总和的比值, 比值越大, 代表该城市的发货稳定性越强。

$$S_i = \frac{\sum_{d=1}^N \sum_{j \in \{A \dots Z\}} s_{ij}^d}{N} \quad (5-2)$$

$$s_{ij}^d = \begin{cases} 0, & \sum_{j \in \{A \dots Z\}} X_{ij}^d = 0 \\ 1, & \sum_{j \in \{A \dots Z\}} X_{ij}^d \neq 0 \end{cases} \quad (5-3)$$

其中,  $s_{ij}^d$  采用 (0, 1) 赋值法, 当在时间  $d$  内  $i$  城市向其他城市快递发货量总和不为零时赋值为 1; 发货量总和为零时赋值为 0

### 5.1.2 收货情况

(1) 各城市收货总量 $R_i$ : 在过去一段时间内各个城市在本城市收到与其有相关快递往来的城市发来的快递量总和, 代表了各城市的快递收货量总和, 能够有效反映各城市的快递收货容纳能力。

$$R_i = \sum_{d=1}^N \sum_{j \in \{A \dots Z\}} Y_{ij}^d \quad (5-4)$$

其中,  $d$  表示发货日期,  $Y_{ij}^d$  表示在日期  $d$ ,  $i$  城市从  $j$  城市取得的快递收货量。

(2) 收货持续性 $A_i$ : 在过去一段时间内各个城市快递收货量不为零的天数与过去时间天数总和的比值, 比值越大, 代表该城市的收货稳定性越强。

$$A_i = \frac{\sum_{d=1}^N \sum_{j \in \{A \dots Z\}} a_{ij}^d}{N} \quad (5-5)$$

$$a_{ij}^d = \begin{cases} 0, & \sum_{j \in \{A \dots Z\}} Y_{ij}^d = 0 \\ 1, & \sum_{j \in \{A \dots Z\}} Y_{ij}^d \neq 0 \end{cases} \quad (5-6)$$

其中,  $a_{ij}^d$  采用 (0,1) 赋值法, 当在时间  $d$  内  $i$  城市从其他城市取得的快递收货量总和不为零时赋值为 0; 收货量总和为零时赋值为 1。

### 5.1.3 快递数量变化趋势

(1) 各城市快递发货量长期变动趋势 $P_i$ : 通过 MATLAB 对相关数据进行最优拟合, 得到快递发货量的变化趋势图以及发货量长期变化趋势系数  $k_1$ , 以 D 城市的发货量变化趋势为例, 结果如图 1 所示:

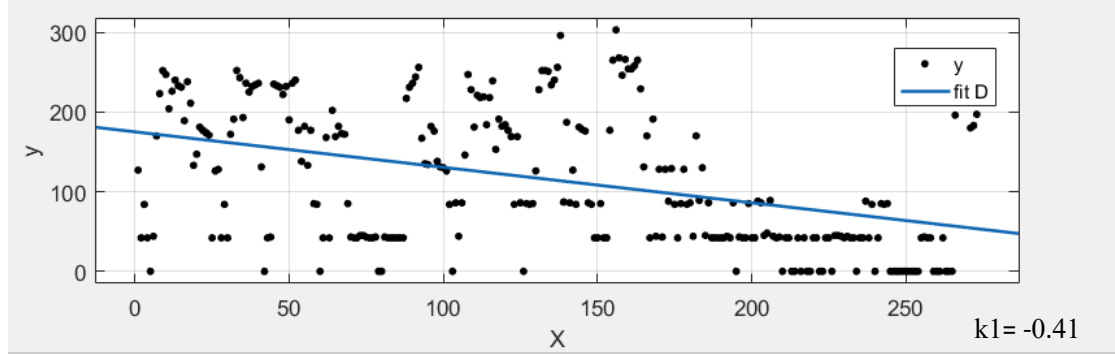


图 1 D 城市快递发货量长期变动趋势图

(2) 各城市快递收货量长期变动趋势 $Q_i$ : 通过 MATLAB 对相关数据进行最优拟合, 得到快递收货量的变化趋势图以及收货量长期变化趋势系数  $k_2$ , 以 D 城市的收货量变化趋势为例, 结果如图 2 所示:

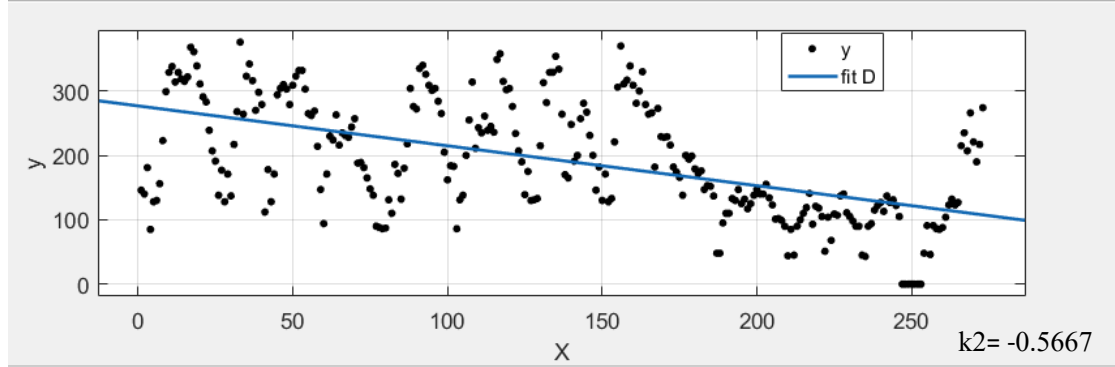


图 2 D 城市快递收货量长期变动趋势图

### 5.1.4 城市相关性

(1) 平均上游城市数量 $\bar{M}_i$ : 在过去一段时间内平均每一天有多少城市收到城市  $i$  的发货, 这一指标能够反映城市  $i$  作为上游基站与其他城市的物流联系强度。

$$\bar{M}_i = \frac{\sum_{d=1}^N \sum_{j \in \{A \dots Z\}} m_{ij}^d}{N} \quad (5-8)$$

其中,  $m_{ij}^d$  代表在日期  $d$  内收到城市  $i$  快递发货的城市总和。

(2) 平均下游城市数量 $\bar{O}_i$ : 在过去一段时间内平均每一天城市  $i$  收到多少城市的发货, 这一指标能够反映城市  $i$  作为下游收货点与其他城市的物流联系强度。

$$\bar{O}_i = \frac{\sum_{d=1}^N \sum_{j \in \{A \dots Z\}} n_{ij}^d}{N} \quad (5-9)$$

其中,  $n_{ij}^d$  代表在日期  $d$  内城市  $i$  收到快递发货的城市总和。

### 5.1.5 供需稳定性

(1) 供给稳定性 $I_i$ : 在过去一段时间内城市 $i$ 向其他城市发货量的标准差, 反映了城市 $i$ 与其他城市之间的快递供给稳定性。

$$I_i = \sqrt{\frac{1}{N} \sum_{i=d}^N \sum_{j \in \{A \cdots Z\}} (X_{ij}^d - \bar{X}_i)^2} \quad (5-10)$$

(2) 需求稳定性 $U_i$ : 在过去一段时间内城市 $i$ 取得收货量 (来自各城市) 的标准差, 反映了城市 $i$ 与其他城市之间的快递需求稳定性。

$$U_i = \sqrt{\frac{1}{N} \sum_{i=d}^N \sum_{j \in \{A \cdots Z\}} (Y_{ij}^d - \bar{Y}_i)^2} \quad (5-11)$$

综上所述, 本文总结出评估城市重要程度 5 个方面的 10 个指标, 5 个方面分别为发货情况、收货情况、快递数量变化趋势、城市关系相关性、供需稳定性, 10 个指标为发货总量 $C_i$ 、发货持续性 $S_i$ 、收货总量 $R_i$ 、收货持续性 $A_i$ 、快递发货量长期变动趋势 $P_i$ 、快递收货量长期变动趋势 $Q_i$ 、平均上游城市数量 $\bar{M}_i$ 、平均下游城市数量 $\bar{O}_i$ 、供给稳定性 $I_i$ 、需求稳定性 $U_i$ 。由此构建重要城市指标评价体系, 如图 3 所示。

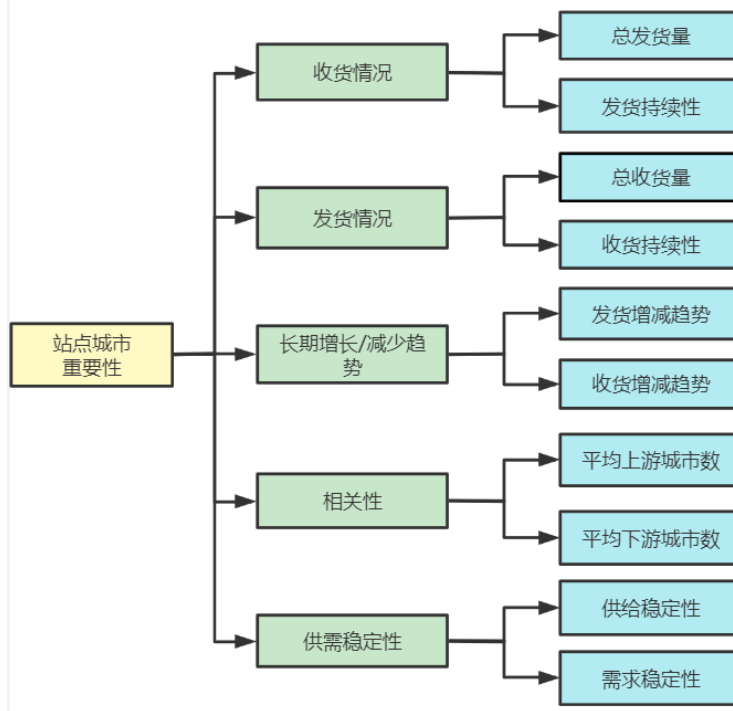


图 3 重要城市指标评价体系

### 5.2 站点城市重要程度评估模型的建立

本文采用随机森林-TOPSIS 模型构建站点城市重要程度评估模型, 并用于评价站点城市重要程度。随机森林-TOPSIS 模型是利用采样和构建树结构确定指标权重, 通过计算接近度确定重要度的方法, 该方法拥有充分利用原始数据、信息量损失小的优点, 模型采用无偏估计, 模型泛化能力强, 能够在多特征判断条件下中更好的评价数据。随机森林-TOPSIS 模型评价流程如图 4 所示。



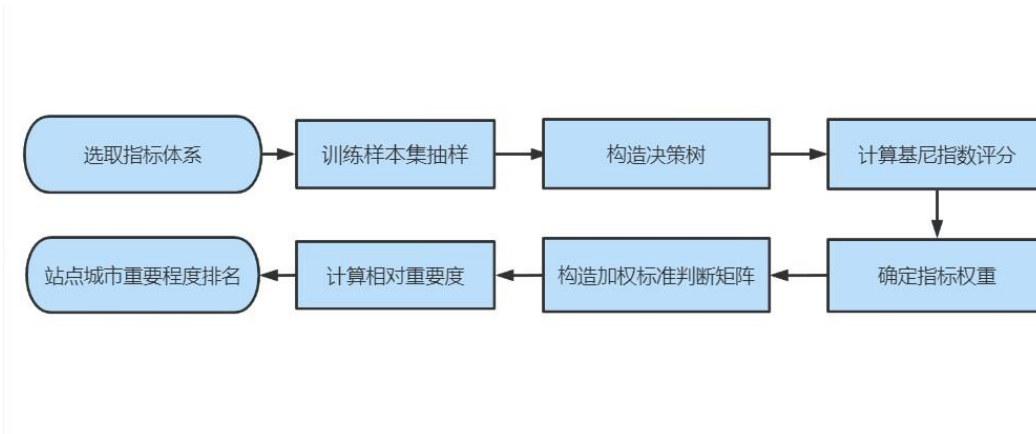


图 4 随机森林-TOPSIS 模型评价流程

### 5.2.1 随机森林算法简介

随机森林是一个用随机方式建立的，包含多个决策树的分类器。其输出的类别是由各个树输出的类别的众数而定<sup>[2]</sup>。它通过采取多个不同的训练样本子集来加大分类模型之间的相异性，从而能够提高该模型的泛化能力以及预测能力。随机森林将多个决策树结合在一起，每次数据集是随机有放回的选出，同时随机选出部分特征作为输入，所以该算法被称为随机森林算法。可以看到随机森林算法是以决策树为估计器的 Bagging 算法。

随机森林方法主要包括训练与分类两个过程，如图 5。随机森林训练过程通过 Bootstrap 抽样方法从训练样本中抽取个样本子集。将未选中的样本作为袋外数据；然后对于个样本子集采用 CART 方法构建对应的决策树。在单个决策树中的所有节点进行有放回的抽取个特征，通过计算每一个特征所包含的信息量来分裂生长<sup>[5]</sup>。决策分类过程是对每一棵决策树的决策结果使用简单多数投票法进行综合，得到最终的分类结果，如下式：

$$H(x) = \arg \max_Y \sum_{i=1}^K I(h_i(x) = Y) \quad (5-12)$$

其中， $H(x)$ 表示随机森林最终分类结果， $h_i(x)$ 表示单一决策树模型分类结果， $Y$ 为目标变量， $I(h_i(x) = Y)$ 为示性函数。随机森林原理如图 5 所示：

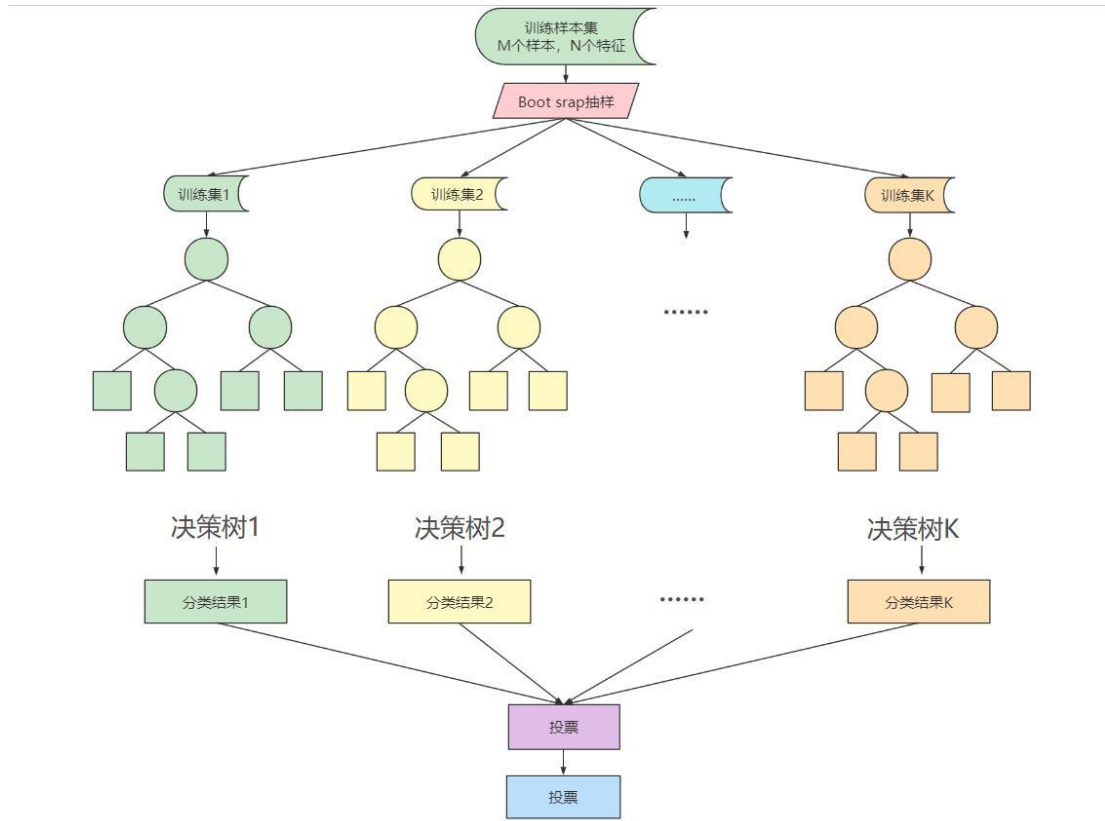


图5 随机森林原理示意图

### 5.2.2 随机森林-TOPSIS 模型的建立

随机森林-TOPSIS 模型设定一个最优值即正向最大值，一个负向最大值，通过排序确定评价对象的重要度。

**步骤 1:** 根据评价对象构造初始矩阵。设该矩阵有  $m$  个评价对象与  $n$  个评价指标， 设

$$X = (x_{ij})_{m \times n} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1n} & x_{2n} & \cdots & x_{nn} \end{bmatrix}, (i = 1, 2, \dots, m; j = 1, 2, \dots, n) \quad (5-13)$$

其中， $X$ 为初始矩阵， $x_{ij}$ 为第  $i$  个评价对象的第  $j$  个评价指标数据。

**步骤 2:** 数据标准化处理。

由于各指标的量纲，数量级均有差异，为消除量纲不同对评价结果的影响，需要对各指标进行标准化处理，本文采用极值法对数据进行处理。

当为正向指标时：

$$v_{ij} = \frac{x_{ij} - \min\{x_{1j}, x_{2j}, \dots, x_{nj}\}}{\max\{x_{1j}, x_{2j}, \dots, x_{nj}\} - \min\{x_{1j}, x_{2j}, \dots, x_{nj}\}} \quad (5-14)$$

当为负向指标时：

$$v_{ij} = \frac{\max\{x_{1j}, x_{2j}, \dots, x_{nj}\} - x_{ij}}{\max\{x_{1j}, x_{2j}, \dots, x_{nj}\} - \min\{x_{1j}, x_{2j}, \dots, x_{nj}\}} \quad (5-15)$$

其中， $\max\{x_{1j}, x_{2j}, \dots, x_{nj}\}$ 为各项指标的最大值， $\min\{x_{1j}, x_{2j}, \dots, x_{nj}\}$ 为各项指标的最小值。

**步骤 3：**计算每个评价指标的基尼指数评分 $VMI_j^{(Gini)[3]}$ ，即第  $j$  个评价指标在随机森林所有决策树中节点分裂不纯度的改变量均值。

基尼指数的计算公式为式：

$$GI_m = \sum_{k=1}^{|K|} \sum_{k' \neq k} p_{mk} p'_{mk} = 1 - \sum_{k=1}^{|K|} p_{mk}^2 \quad (5-16)$$

式中， $K$ 表示有 $K$ 各类别， $p_{mk}$ 表示节点  $m$  中类别  $k$  所占比例。则节点分支前后基尼指数变化量为：

$$VMI_j^{(Gini)} = GI_m - GI_l - GI_r \quad (5-17)$$

式中， $GI_l$ 和 $GI_r$ 分别表示分支前后两个新节点的基尼指数。

**步骤 4：**计算并确定权重。

设指标  $j$  在决策树  $i$  中出现的节点为集合  $M$ ，那么指标  $j$  在第  $i$  棵树的评分为：

$$VMI_{ij}^{(Gini)} = \sum_{m \in M} VMI_{jm}^{(Gini)} \quad (5-18)$$

假设随机森林中共有  $n$  棵树，则

$$VMI_j^{(Gini)} = \sum_{i=1}^n VMI_{ij}^{(Gini)} \quad (5-19)$$

最后将所有求得的权重评分归一化处理，得每个评价指标的权重 $VMI_j$ 。

$$VMI_j = \frac{VMI_j}{\sum_{i=1}^c VMI_i} \quad (5-20)$$

**步骤 5：**构建加权标准化判断矩阵。

$$Z = (VMI_j v_{ij})_{m \times n} = \begin{bmatrix} VMI_1 v_{11} & VMI_2 v_{12} & \cdots & VMI_n v_{1n} \\ VMI_1 v_{21} & VMI_2 v_{22} & \cdots & VMI_n v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ VMI_1 v_{m1} & VMI_2 v_{m2} & \cdots & VMI_n v_{mn} \end{bmatrix} \quad (5-21)$$

**步骤 6：**计算相对接近度。

设决策问题有 $m$ 个目标 $f_j(j = 1, 2, \dots, m)$ ， $n$ 个可行解，并设该问题的规范化加权目标的正负理想解分别为 $Z^+$ 、 $Z^-$ ，其中：

$$Z^+ = (\max z_{i1}, \max z_{i2}, \dots, \max z_{ij}) \quad (5-22)$$

$$Z^- = (\min z_{i1}, \min z_{i2}, \dots, \min z_{ij}) \quad (5-23)$$

$z_{ij}$ 为第 $i$ 个评价对象的第 $j$ 个评价指标的数据加权值,  $z_{ij} = VMI_j v_{ij}$ 。

那么用欧几里得范数作为距离的测度，则从任意可行解到的距离为

$$S_i^+ = \sqrt{\sum_{j=1}^m (Z_{ij} - Z_j^+)^2} \quad (5-24)$$

$$S_i^- = \sqrt{\sum_{j=1}^m (Z_{ij} - Z_j^-)^2} \quad (5-25)$$

式中， $S_i^+$ 为 $i$ 到正最大值的接近度， $S_i^-$ 为 $i$ 到负最大值的接近度。

**步骤 7：确定重要度.** 各评价对象的重要度为

$$C_i = \frac{S_i^-}{S_i^- + S_i^+} \quad (5-26)$$

式中： $C_i$ 为第 $i$ 个评价对象的重要度，以贴近 0 或 1 的距离判断重要度，越贴近 1，重要度越高，反之越贴近 0，重要度越低。

### 5.3 问题一的求解

利用 matlab 工具求解上述模型，计算出站点城市重要程度指标权重，如图 6 所示。根据站点城市重要程度指标权重，发现其中较为重要的指标（权重>10%）为发货总量、总收货量和平均下游城市数量，其权重分别达到 17.16%、14.57%和 11.46%。

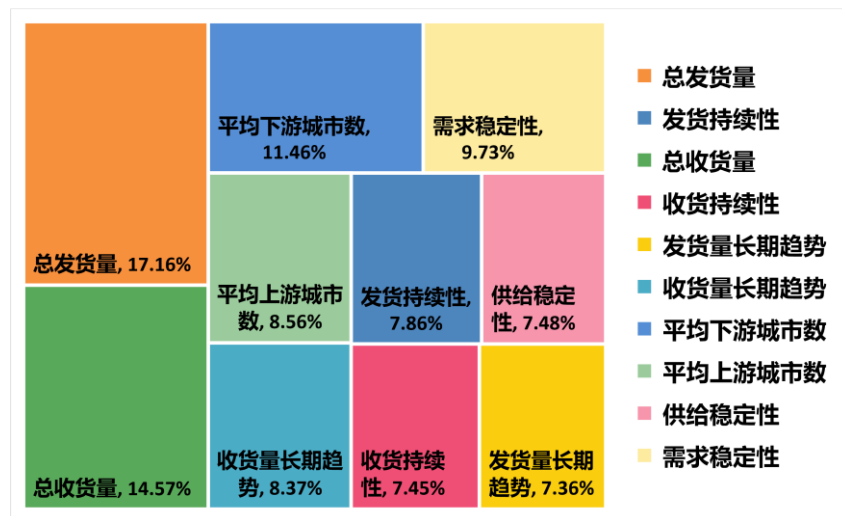


图 6 城市重要性指标权重树状图

在得到站点城市重要程度评价指标权重基础上计算得出各城市重要程度综合得分指数，通过比较得出站点城市的重要程度排名，如图 7 所示：

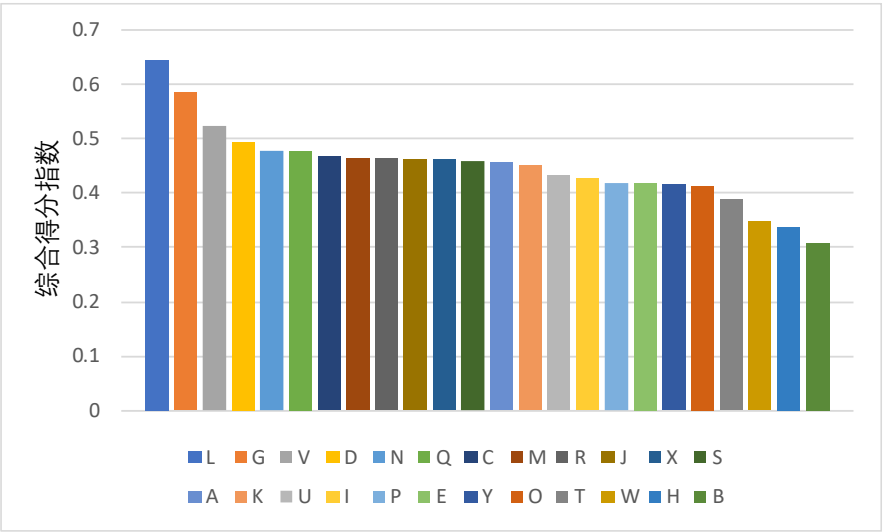


图 7 站点城市重要程度柱状图

综上，重要程度排名前五的站点城市分别为 L、G、V、D、N。

表 1 问题 1 结果

排序	1	2	3	4	5
城市名称	L	G	V	D	N

通过对数据的观察可以看出，L、G、V 城市排名较高主要是由于发货总量和收货总量明显高于其他城市，且有较多的上下游物流站点城，拥有更强的中心辐射性，而 D 和 N 城市的供需稳定性较高，因此重要程度综合计算排名也较高。

六、问题二模型的建立与求解

对于预测 2019 年 4 月 18 日和 2019 年 4 月 19 日各“发货-收货”站点城市之间快递运输数量，考虑到数据波动性较大、周期性较强，与时间有较强的关联性，我们选用 Cho 等人提出的带有 GRU 的 LSTM 模型，并通过比对其他机器学习模型(决策树、随机森林、LGBM、XGBOOST)验证带有 GRU 的 LSTM 模型的预测准确性，避免了单个模型预测可能存在的偶然性。

6.1 基于 GRU-LSTM 模型的站点城市快递运输数量预测模型

LSTM（Long short-term memory）是一种特殊的 RNN，解决了 RNN 长期记忆能力不足、梯度消失、梯度爆炸等问题。LSTM 适合对时间序列进行处理，原因是在时间序列中的重要事件之间可能存在未知持续时间的滞后，而对于间隙长度的相对不敏感性是 LSTM 的一大优势。LSTM 单元由输入门、遗忘门、输出门和单元状态组成，该单元记住任意时间间隔内的值，三个门控制进出单元的信息流。工作原理和流程如图 8、图 9，模型公式见（6-1）-（6-5）

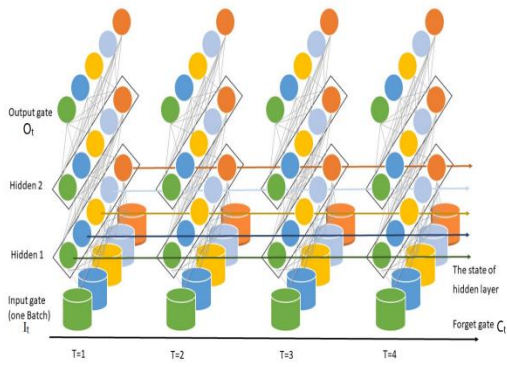


图 8 LSTM 模型工作原理

$$I_t = \sigma(X_t W_{xi} + H_{t-1} W_{hi} + b_i) \quad (6-1)$$

$$C_t = \sigma(X_t W_{xf} + H_{t-1} W_{hf} + b_f) \quad (6-3)$$

$$H_t = O_t \odot \tanh(C_t) \quad (6-5)$$

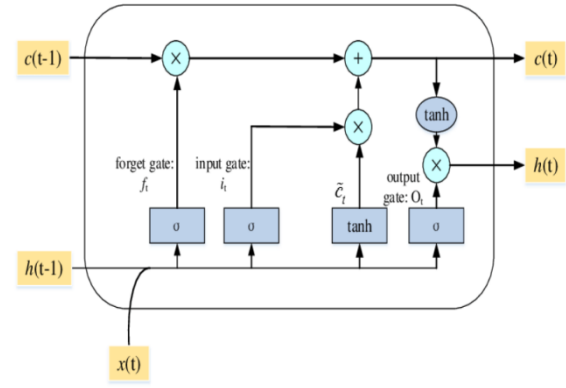


图 9 LSTM 流程

$$O_t = \sigma(X_t W_{xo} + H_{t-1} W_{ho} + b_o) \quad (6-2)$$

$$C_t \sim \tanh(X_t W_{xc} + H_{t-1} W_{hc} + b_c) \quad (6-4)$$

GRU 作为 LSTM 的一种变体，将忘记门和输入门合成了一个单一的更新门。同样还混合了细胞状态和隐藏状态，加诸其他一些改动<sup>[4]</sup>。最终的模型比标准的 LSTM 模型要简单，其流程图如图 10，模型公式见 (6-6) - (6-9)

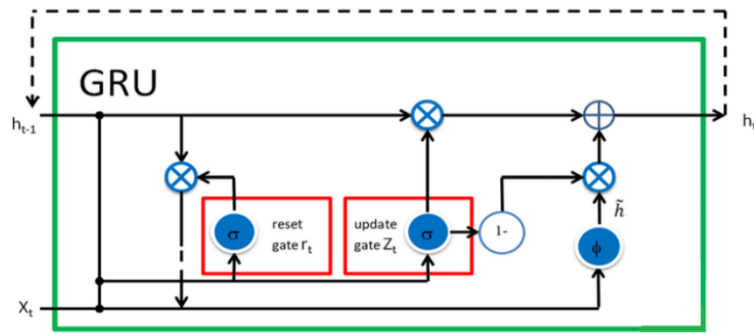


图 10 GRU 模型流程图

$$R_t = \sigma(X_t W_{xr} + H_{t-1} W_{hr} + b_r) \quad (6-6)$$

$$H_t \sim \tanh(X_t W_{xh} + R_t \odot H_{t-1} W_{hh} + b) \quad (6-8)$$

$$Z_t = \sigma(X_t W_{xz} + H_{t-1} W_{hz} + b_z) \quad (6-7)$$

$$H_t = Z_t \odot H_{t-1} + (1 - Z_t) \odot H_t \quad (6-9)$$

## 6.2 GRU-LSTM 模型预测准确性的评估

经 GRU-LSTM 模型预测，2023 年 3 月 1 日站点城市 M-U 之间快递运输数量为：157.23，模型误差为：0.0179（1.79%）

为充分验证或避免上述模型可能出现的错误，我们使用决策树、随机森林、LGBM、XGBoost 模型对 3 月 1 日站点城市 M-U 之间快递运输数量进行预测，相关结果如表 2：

表 2 各模型预测数据及误差

模型	预测值	误差
决策树	157.49	0.1052
随机森林	157.31	0.0372
LGBM	158.35	0.0203
XGBoost	156.67	0.1052
GRU-LSTM	157.23	0.0179

经对比发现，GRU-LSTM 模型相对其他模型误差较低，拟合程度较高，拟合效果如图 11：

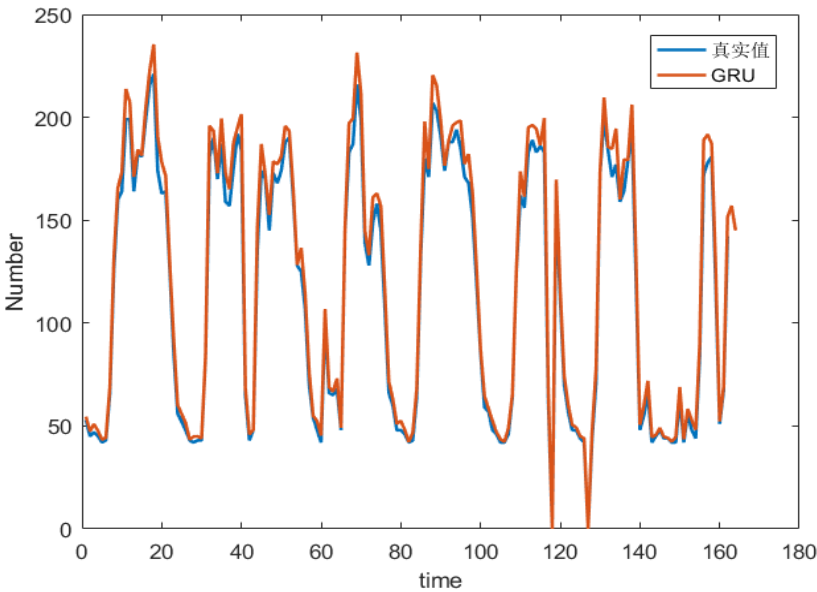


图 11 GRU 模型对 M-U 之间快递运输数量的拟合与预测

经过各模型对 Q-V、K-L 等站点城市对之间的预测结果的比对，同样得到 GRU 模型相对其他模型预测误差较低的结论。因此我们认为：使用 GRU-LSTM 模型预测得到的站点城市之间快递运输数量可信度较高。

6.3 问题二的求解

采用 GRU-LSTM 模型对 2019 年 4 月 18 日和 2019 年 4 月 19 日各“发货-收货”站点城市之间快递运输数量进行预测，并对全部“发货-收货”站点城市之间快递运输数量进行加总得到两天总快递运输数量。问题二结果(已根据实际意义四舍五入)如下表所示：

表 3 问题 2 结果

日期	“发货-收货”城市之间的快递运输数量		所有“发货-收货”城市之间的总快递运输数量
2019 年 4 月 18 日	M-U	157	10658
	Q-V	48	
	K-L	70	
	G-V	574	
2019 年 4 月 19 日	V-G	507	9430
	A-Q	129	
	D-A	45	
	L-K	293	

## 七、问题三模型的建立与求解

### 7.1 基于 GRU-LSTM 模型的离散型时序预测

考虑到 GRU-LSTM 模型对于时序数据的预测具有十足的优良性<sup>[5]</sup>，同时问题所给数据的信息较少，在本问继续采用问二的 GRU-LSTM 模型对能否正常发货进行离散值的时间序列预测。

由于站点城市之间无法正常发货或收货的情况包括无数据或运输数量为 0 的情况，首先通过 Excel 筛选出特定的“发货-收货”的站点城市对(发货城市-收货城市)，再对 2020 年 4 月 28 日—2023 年 4 月 27 日之间不存在的日期数据(6 月、11 月、12 月的剔除数据除外)进行补全(当日快递运输数量填补为 0)。

为减少预测误差，本部分通过公式 (7-1) 将连续性时序预测(数值预测)转化为离散型时序预测(事件预测)以提高预测的准确性，训练模型以进行对发货可行性的预测，并设定阈值 0.5 作为判断标准。若基于 GRU-LSTM 模型的发货可行性预测结果显示能正常发货（即预测结果大于 0.5），则同问题二步骤继续采用 GRU-LSTM 模型对转化前的数据进行连续性预测得到快递运输数量，否则不能正常发货，快递运输数量为 0。

$$w_{ij}^d = \begin{cases} 0, & x_{ij}^d = 0 \\ 1, & x_{ij}^d \neq 0 \end{cases} \quad (7-1)$$

其中， $w_{ij}^d$ 代表在 d 日期时 i 城市向 j 城市能否正常发货或收货，当站点城市对之间能正常收货或发货时，赋值为 1；当站点城市对之间无法正常发货或收货，赋值为 0。

以 S-Q 城市对为例，基于 GRU-LSTM 模型的 2023 年 4 月 28 日发货可行性预测结果为 0.4742 ( $<0.5$ )，故推断该日不能正常发货，快递运输数量为 0，预测结果如图 12 所示：

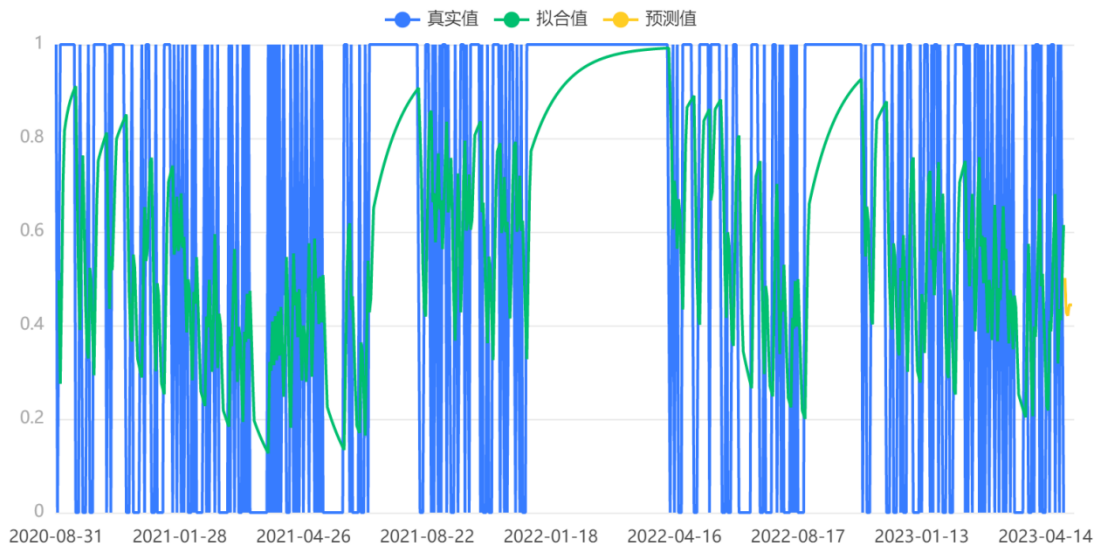


图 12 基于 GRU 模型的 S-Q 城市对发货可行性预测



7.2 对问题三的求解

采用 GRU-LSTM 模型对 2023 年 4 月 28 日和 2023 年 4 月 29 日各“发货-收货”站点城市队之间的发货可行性进行预测。最终问题三预测结果(已根据实际情况四舍五入)如下表所示：

表 4 问题 3 结果

日期	“发货-收货” 站点城市对	是否能正常发货 (填写“是”或“否”)	快递运输数量
2023 年 4 月 28 日	I-S	是	48
	M-G	是	49
	S-Q	否	0
	V-A	是	50
	Y-L	是	41
2023 年 4 月 29 日	D-R	是	55
	J-K	是	203
	Q-O	是	61
	U-O	是	46
	Y-W	是	169

八、问题四模型的建立与求解

8.1 运输方案非线性规划模型的建立

本文以铁路运输成本最低为目标，建立非线性规划模型。结合题目已知条件与假设，目标函数为：由发货城市  $s$  到收货城市  $t$  之间的快递运输成本最小。

$$\min \sum_{(v_i,v_j) \in A} b_{ij} \times \left[ 1 + \left( \frac{f_{ij}^d}{200} \right)^3 \right] \tag{8-1}$$

其中， $A$  表示由收发站点城市  $(s,t)$  之间运输铁路构成的有向网络， $V=\{v_A,v_B...,v_Y\}$  为  $A$  的顶点集， $b_{ij}$  表示城市  $i$  与城市  $j$  之间运输的固定成本， $f_{ij}^d$  表示日期  $d$  内城市  $i$  向  $j$  运输的货物量。

该规划模型需要满足以下约束：

(1) 邻接站点约束：将有向图转化为邻接矩阵  $D$  的形式以表示任意两个城市之间是否可直接进行铁路运输，则对于从发货城市  $s$  到收货城市  $t$  之间的每个站点城市  $(i,j)$  之间的运输必须满足  $u_{ij}=1$ 。

$$D = \begin{bmatrix} 0 & u_{AB} & \cdots & u_{AY} \\ u_{BA} & 0 & \cdots & u_{BY} \\ \vdots & \vdots & \ddots & \vdots \\ u_{YA} & u_{YB} & \cdots & 0 \end{bmatrix}$$

(8-2)

其中，当有向图中存在城市  $i$  指向城市  $j$  的箭头则  $u_{ij}=1$ ，否则  $u_{ij}=0$ 。

(2) 平衡条件约束：对于作为运输中转的城市，不能存在货物的停滞，即有流入量=流出量：

$$\sum_{j:(v_i, v_j) \in A} f_{ij}^d - \sum_{k:(v_i, v_k) \in A} f_{ki}^d = 0$$

(8-3)

对于发点  $v_s$  和收点  $v_t$ ，有

$$\sum_{j:(v_s, v_j) \in A} f_{sj}^d = \sum_{j:(v_k, v_t) \in A} f_{kt}^d = X_{st}^d$$

(8-4)

(3) 正向流量约束：根据实际意义，各个城市之间货物的流量必然为正值或零：

$$0 \leq f_{ij}^d$$

(4) 有限步长约束：每个“发货-收货”站点城市对之间使用的路径数不超过 5 条：

$$P_{st} \leq 5$$

其中  $P_{st}$  表示的从发货城市  $s$  到收货城市  $t$  所经过的路径数（若  $s$  可直接运输到  $t$  则路径数为 1，若需经一个城市中转则路径数为 2）。

综合以上讨论，发货城市  $s$  到收货城市  $t$  之间的最低成本运输方案非线性规划模型为：

$$\begin{aligned} & \min \sum_{(v_i, v_j) \in A} b_{ij} \times \left[ 1 + \left( \frac{f_{ij}^d}{200} \right)^3 \right] \\ & \text{s.t.} \begin{cases} u_{ij} = 1, & (v_i, v_j) \in A \\ \sum_{j:(v_s, v_t) \in A} f_{st}^d = X_{st}^d \\ \sum_{j:(v_i, v_j) \in A} f_{ij}^d - \sum_{k:(v_i, v_k) \in A} f_{ki}^d = 0 \\ \sum_{m:(v_k, v_t) \in A} f_{kt}^d = v_{st} \\ 0 \leq f_{ij}^d \\ P_{st} \leq 5 \end{cases} \end{aligned}$$

(8-3)

则对于满足以上条件的路径解即为能实现在确定日期使得发货城市  $s$  到收货城市  $t$  之间的货物运输成本达到最低的最佳运输方案，将当日所有收发城市对之间的最低运输成本相加，即可得到当日总的最低运输成本。

8.2 问题四的求解

对以上最低成本运输方案非线性规划模型使用 MATLAB 进行求解，得到问题四结果如下：

表 5 问题 4 结果

日期	最低运输成本
2023 年 4 月 23 日	6634
2023 年 4 月 24 日	7279
2023 年 4 月 25 日	6576
2023 年 4 月 26 日	6752
2023 年 4 月 27 日	6418

九、问题五模型的建立与求解

9.1 固定需求常数分析

9.1.1 数据处理与常数指标计算

对附件 2 数据进行预处理，筛除由于突发因素造成的非正常发货与零发货数据。将预处理数据按照年份、季度和收发城市对进行交叉分组，并在此基础上判别平均值、标准差及最小值 $t_{ij}$ 等各常数指标，最终得出 1100 组季度组数据。

9.1.2 固定需求常数的计算

由于题目中要求固定需求常数一般不能简单的认定为快递需求历史数据的最小值，且通常小于需求的最小值，故本文根据文献资料<sup>[6]</sup>构造经验计算公式，充分考虑到各组数据平均值、标准差及最小值对固定需求常数 $H_i^s$ 的影响，一般认为固定需求常数 $H_i^s$ 为平均值减去两倍的标准差，并以此进行估算：

$$H_i^s = \overline{t_{ij}^s} - 2\sigma_{ij}^s \tag{9-1}$$

9.1.3 固定需求常数的准确性检验

根据中心极限定理，我们可以认为固定需求常数服从于正态分布 $X \sim (\mu, \sigma^2)$ ，为了对其准确性进行检验，本文采用正态分布检验中的 $3\sigma$ 原则，观察各季度固定需求常数的分布落点，若每一年各季度落点集中且邻近均值，则可以认为固定需求常数的准确性较高，经过检验，结论成立。

9.2 非固定需求分析

9.2.1 非固定需求数据的获取

根据上一步中得出的固定需求常数，以附件二预处理过后的数据扣除各季度城市组固定需求常数得出各组的非固定需求数量  $G$ ：

$$G_i^s = X_{ij}^d - H_i^s \quad (9-2)$$

其中， $X_{ij}^d$ 表示在日期  $d$  内城市  $i$  向城市  $j$  的快递运输总量。

### 9.2.2 非固定需求分布函数常数指标的计算

根据中心极限定理，当数据量  $n \rightarrow \infty$  时，数据分布越接近于正态分布，由于附件 2 中每个季度每个“发货-收货”站点城市对的非固定需求数据量足够大，这里可以将非固定需求分布近似作为正态分布进行求解。

为了估计非固定需求的概率分布，我们可以使用最大似然估计（MLE）方法，估计概率分布的均值和标准差。

### 9.3.3 正态分布下最大似然估计的原理

单变量正态分布的形式为：

$$\rho(x | \theta) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (9-3)$$

其中均值  $\mu$  和方差  $\sigma^2$  为未知参数，即待估计的参数

$$\theta = [\theta_1, \theta_2]^T = [\mu, \sigma^2]^T \quad (9-4)$$

用于估计的样本仍然是

$$X = \{x_1, x_2, \dots, x_n\} \quad (9-5)$$

根据如下式子：

$$\nabla \theta H(\theta) = \sum_{i=1}^N \nabla \theta \ln \rho(x_i | \theta) = 0 \quad (9-6)$$

从正态分布式(9-3)可以得到：

$$\ln \rho(x_k | \theta) = -\frac{1}{2} \ln 2\pi - \frac{1}{2\theta_2} (x_k - \theta_1)^2 \quad (9-7)$$

分别对两个未知参数求偏导，得到：

$$\nabla_{\theta} \ln \rho(x_k | \theta) = \begin{bmatrix} \frac{1}{\theta^2} (x_k - \hat{\theta}_1) \\ -\frac{1}{2\theta_2} + \frac{1}{2\theta_2^2} (x_k - \hat{\theta}_1)^2 \end{bmatrix} \quad (9-8)$$

因此，最大似然估计应该是以下方程组的解：

$$\begin{cases} \sum_{k=1}^N \frac{1}{\theta^2} (x_k - \hat{\theta}_1)^2 = 0 \\ -\sum_{k=1}^N \frac{1}{\theta^2} + \sum_{k=1}^N \frac{1}{\theta^2} (x_k - \hat{\theta}_1)^2 = 0 \end{cases} \quad (9-9)$$

可以得到：

$$\hat{\mu} = \hat{\theta}_1 = \frac{1}{N} \sum_{k=1}^N x_k \quad (9-10)$$

$$\widehat{\sigma^2} = \widehat{\theta}_2 = \frac{1}{N} \sum_{k=1}^N (x_k - \hat{\mu})^2$$

(9-11)

综上，可以得到各组数据的非固定需求均值及非固定需求标准差，在此基础上进行求和即可求解非固定需求均值总和及非固定需求标准差总和。

表 6 问题 5 结果

季度	2022 年第三季度 (7—9 月)		2023 年第一季度 (1—3 月)	
“发货-收货” 站点城市对	V-N	V-Q	J-I	O-G
固定需求常数	17.6428173	25.7961041	87.3582469	33.2476698
非固定需求均值	91.7926278	83.1852627	73.5680327	79.0088648
非固定需求标准差	67.8235143	56.7059387	90.7022508	41.3535272
固定需求常数总和	571.145		437.291	
非固定需求均值总和	672.751		848.775	
非固定需求标准差总和	596.478		421.620	

十、模型的评价、改进和推广

10.1 模型的评价

10.1.1 模型的优点

- 1.在建立城市重要程度评价模型中，在题目所提供数据信息的基础上，通过充分论证选取科学、合理的指标，评价指标多、涉及范围广，可以对站点城市进行全面且准确的评估。
- 2.构建随机森林-TOPSIS 供应商重要性综合评价模型，通过随机森林指标权重计算，通过机器学习得到的重要性评价指标权重可信度高，利用 TOPSIS 评价站点城市重要性时对数据利用程度高，能够有效解决实际问题。
- 3.采用带有 GRU 的 LSTM 模型，并通过比对其他机器学习模型(决策树、随机森林、LGBM、XGBOOST)，多角度验证及分析，避免了单个模型预测可能存在的偶然性，并提高了数据精度。
- 4.采用最大似然估计比其他估计方法更加简单可行，并且数据无偏或者渐近无偏，当样本数目增加时，收敛性质更好。

10.1.2 模型的缺点

- 1.本文数据异常值和缺失值过多，数据量太大，导致数据运行结果精确度下降、误差比例提高。

- 2.本文中通过阈值分层, 阈值的主观性较强, 近似于经验, 而非高精度标准。
- 3.本文中采用的模型部分简化, 导致部分功能的缺失, 但影响不大。

## 10.2 模型的改进

- 1.通过阅读文献和查找资料选取高精度阈值, 以更加准确地判断问题三中快递正常发货情况。
- 2.在计算概率分布特征值时, 可以引入核密度函数进行比较提高结果精确性。
- 3.在进行数据预测时, 可以引入时间序列分析、回归分析、插值与拟合等模型进行优化。

## 10.3 模型的推广

- 1.本文中所构建的各类规划模型及图论模型可以为物流以及其他产业提供最低价格的优选方案。
- 2.本文中所提出的权重比例综合得分评价模型可以用于解决生活中各类评价性问题。

# 十一、参考文献

- [1]Chow G, Heaver T D, Henriksson L E. Logistics performance: definition and measurement[J]. International journal of physical distribution & logistics management, 1994, 24(1): 17-28.
- [2]Breiman L. Random forests[J]. Machine learning, 2001, 45: 5-32.
- [3]周璇, 王晓佩, 梁列全等. 基于随机森林算法的制冷剂充注量故障诊断[J]. 华南理工大学学报(自然科学版), 2020, (2):16-24.
- [4]Chung J, Gulcehre C, Cho K, et al. Gated feedback recurrent neural networks[C]//International conference on machine learning. PMLR, 2015: 2067-2075.
- [5]Chung J, Gulcehre C, Cho K H, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling[J]. arXiv preprint arXiv:1412.3555, 2014.
- [6]李小俊.快递需求量组合预测模型构建及实证研究【D】.西南交通大学,2017(03).

## 十二、附录

### 附录一

介绍：该 NOTEBOOK 代码用于求站点城市重要程度评价指标权重

```
import numpy
import pandas
from spsspro.algorithm import supervised_learning
#生成案例数据
data_x = pandas.DataFrame({
    "A": numpy.random.random(size=100),
    "B": numpy.random.random(size=100)
})
data_y = pandas.Series(data=numpy.random.choice([1, 2], size=100), name="C")
#随机森林分类
result = supervised_learning.random_forest_classifier(data_x=data_x,
data_y=data_y)
print(result)
```

### 附录二

介绍：该 MATLAB 代码用于求站点城市重要程度评价指标权重

`x=xlsread('第一问指标.xlsx); % 读入数据, x 为原始数据矩阵, 每列对应一个指标`  
`[s,w]=shang(x) % s 返回各行得分, w 返回各列权重`

```
function [s,w]=shang(x)
[n,m]=size(x);
%% 数据的归一化处理
[X,ps]=mapminmax(x');
ps.ymin=0.002; % 归一化后的最小值
ps.ymax=0.996; % 归一化后的最大值
ps.yrange=ps.ymax-ps.ymin; % 归一化后的极差,若不调整该值,则逆运算会出错
X=mapminmax(x',ps);
% mapminmax('reverse',xx,ps); % 反归一化,回到原数据
X=X'; % X 为归一化后的数据, 5 列(指标)
%% 计算第 j 个指标下,第 i 个记录占该指标的比重 p(i,j)
for i=1:n
    for j=1:m
        p(i,j)=X(i,j)/sum(X(:,j));
    end
end
%% 计算第 j 个指标的熵值 e(j)
k=1/log(n);
for j=1:m
```

```

e(j)=-k*sum(p(:,j).*log(p(:,j)));
end
d=ones(1,m)-e; % 计算信息熵冗余度
w=d./sum(d); % 求权值 w
s=w*p'; % 求综合得分[\code]

```

### 附录三

介绍：该 python 代码用于计算站点城市的重要性

```

# -*- coding: utf-8 -*-
"""
Created on Tue Aug 3 08:09:56 2021
@author: Amor
"""
import pandas as pd
import numpy as np
url = r'C:\Users\DELL\Desktop\转化.xlsx'
df = pd.read_excel(url, header = None)
# An highlighted block
df = df.astype('int64')
# An highlighted block
df.info()
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
x, y = df.iloc[:, 0:].values, df.iloc[:, 0].values
x_train, x_test, y_train, y_test = train_test_split(x, y,
test_size = 0.3, random_state = 0)
feat_labels = df.columns[0:]
forest = RandomForestClassifier(n_estimators=10000,
random_state=0, n_jobs=-1)
forest.fit(x_train, y_train)
importances = forest.feature_importances_
indices = np.argsort(importances)[::-1]
for f in range(x_train.shape[1]):
    print("%2d) %-*s %f" % (f + 1, 30, feat_labels[indices[f]],
importances[indices[f]]))

```

### 附录四

介绍：该 MATLAB 代码用于进行时序预测



```

clc;clear;close all;
load('data_load.mat')
mu = mean(data_get);
sig = std(data_get);
data_norm = (data_get - mu) ./ sig;
%% 决定网络的输入、输出、总数
num_features=96*7; %输入时间特征
N_predict=96; %输出时间特征,输出需要预测的时间点
num_train=length(data_norm)-num_features;
for i=1:num_train-N_predict
    goodsdata= data_norm(i:i+num_features);
    net_input(i,:)=goodsdata(:)'; %输入一天货物信息
end

for i=1:num_train-N_predict
    timelabel= data_norm(i+num_features+1:i+num_features+N_predict);
    net_output(i,:)=timelabel(:)'; %输入一天货物信息
end
train_num=round(size(net_input,1)*0.8);
%%
train_net_input=net_input(1:train_num,:); %输入训练数据
test_net_input=net_input(train_num+1:end,:); %输入测试数据

train_net_output=net_output(1:train_num,:); %输入训练数据
test_net_output=net_output(train_num+1:end,:); %输入测试数据

XTrain = net_input';
YTrain = net_output';
XTest=test_net_input';
%% 训练神经网络参数设定
layers = [
    sequenceInputLayer(size(XTrain,1),'Name','sequence')
    lstmLayer(128,'Name','lstm')
    dropoutLayer(0.1,'Name','dropout')
    fullyConnectedLayer(size(YTrain,1),'Name','fc')
    regressionLayer('Name','regressionoutput')];
options = trainingOptions('adam', %指定训练选项, 求解器设置为 adam, 1000 轮训练。
    'MaxEpochs',300, %最大训练周期为 150
    'GradientThreshold',1, %梯度阈值设置为 1
    'InitialLearnRate',0.01, %指定初始学习率 0.01
    'LearnRateSchedule','piecewise', %每当经过一定数量的时期时, 学习率就会
    乘以一个系数。
    'LearnRateDropPeriod', 50,

```

```

        'LearnRateDropFactor',0.5, ... %在 50 轮训练后通过乘以因子 0.5 来降低学
习率。
        'Verbose',0, ... %如果将其设置为 true，则有关训练进度的信息将被打印到命
令窗口中,0 即是不打印。
        'Plots','training-progress'); %构建曲线图
net = trainNetwork(XTrain,YTrain,layers,options); %训练神经网络
save('GRU_net', 'net'); %将 net 保存为 GRU_net
%% 多步预测
load('GRU_net.mat')
net=predictAndUpdateState(net, XTrain);
numTimeStepsTest=numel(XTest(1,:));
YPred=[];
%%
%得到预测结果
[net,YPred]= predictAndUpdateState(net, XTest);
YPred1=YPred.*sig'+mu';
test_net_output1=test_net_output.*sig'+mu';
YPred1=YPred1';
N=100; %选取一部分看结果
plot(test_net_output1(N,:))
hold on
plot(YPred1(N,:))
hold on
xlabel('Time point')
ylabel('power')
legend('True','predict')
MAPE=sum(sum(abs(YPred1-test_net_output1)./YPred1))/(size(YPred1,1)*size(YP
red1,2));
disp('整体预测误差')
disp(MAPE)
%% 最终预测结果
XTest1=data_norm(end-num_features:end);
[net,YPred2]= predictAndUpdateState(net, XTest1');
YPred_last=YPred2.*sig'+mu'; %输出的最终结果

```

## 附录五

介绍：该 MATLAB 代码用于求发货城市 vs 到收货城市 vt 的运输成本

```

clc,clear
x=xlsread('第三问矩阵.xlsx'); % 读入有向图数据
fix=xlsread('固定成本.xlsx'); % 读入固定成本
M=xlsread('货物运输量.xlsx'); % 货物运输量数据
load data1

NN=cellstr(strcat('v',int2str([2:5]'))); %构造中间节点
NN={'vs',NN{:},'vt'}; %添加发点和收点
G=digraph;G=addnode(G,NN);
G1=addedge(G,L(:,1),L(:,2),cell2mat(L(:,3)));

c=full(adjacency(G1,'weighted')); %导出容量矩阵
f=optimvar('f',25,25,'LowerBound',0);
prob=optimproblem;prob.Objective=sum(sum(fix.*(1.+(f./200).^3)));
con1=[sum(f(1,:))==M
      sum(f(:,[2:end-1]))'==sum(f([2:end-1],:),2)
      sum(f(:,end))==M];
prob.Constraints.con1=con1;
prob.Constraints.con2=p<=5;
x0.x=zeros(25,25);
[sol,fval,flag,out]=solve(prob,x0)
ff=sol.f %显示最小费用对应的矩阵

```