

队伍编号	MC2301588
题号	C

基于时间序列预测模型的物流网络优化分析

摘要

本题是典型的运输路线安排问题（*LRP*），运用最优化理论相关知识，基于时间序列预测 *LSTM* 模型，构建非线性双层规划模型，采用层次粒子群算法进行网络结构优化。

针对问题一，题目要求根据附件 1 预测 2023 年 1 月份每条线路每天的货量。对此先是采用 3 次样条插值法对部分缺失数据进行预处理，然后构建时间序列预测模型 *LSTM*，并使用 *MATLAB* 软件进行编程求解，得出每条线路每天的货量预测值。

针对问题二，题目要求在问题一的基础上，在 *DC5* 关停后，对物流线路网络结构优化，使得 *DC5* 关停前后货量发生变化的线路尽可能少、包裹日累积量尽可能少。首先筛选出与 *DC5* 有关的物流场地，然后剔除与 *DC5* 联系不紧密的物流场地，为了简化问题，我们以 2023 年 1 月份货量的均值代替每日的货量，只分析与 *DC5* 有紧密联系的物流场地。接着构建非线性双层规划模型，采用层次粒子群算法进行求解得出货量变化线路数、线路工作负荷和日累积包裹量。

针对问题三，题目要求在 *DC9* 关停后，可以通过增加线路对物流网络结构优化，使得 *DC9* 关停前后货量发生变化的线路尽可能少、包裹日累计量尽可能少。首先，在问题二的基础上，引入时间变量 t 和线路变化变量 x_{ikt} 。然后建立有时间变量的非线性双层规划模型，最后使用层次粒子群算法进行求解。

针对问题四，题目要求对此物流网络的物流场地及线路的重要性进行评价，并在问题一的基础上，研究如何通过新增物流场地及线路，优化网络的性能并探讨所建网络的鲁棒性。首先通过查阅资料获得物流场地及线路的评价指标并通过熵权法确定权重，然后使用 *TOPSIS* 分析方法，求得不同物流场地和线路的综合得分，综合得分越高者，重要性越高。接着规划新线路，以缓解重要性最高的物流场地和线路的压力，最后通过多种人为假设来探讨网络的鲁棒性。

关键词：3 次样条插值；*LSTM*；非线性双层规划；层次粒子群算法；*TOPSIS*；

目录

一、问题重述	1
1.1 问题背景	1
1.2 问题提出	1
二、问题的分析	2
2.1 问题的整体分析	2
2.2 问题一的分析	2
2.3 问题二的分析	3
2.4 问题三的分析	3
2.5 问题四的分析	3
三、模型的假设	3
四、符号说明	4
五、模型的建立与求解	4
5.1 问题一	4
5.1.1 数据预处理	4
5.1.2 问题一模型的建立	5
5.1.3 输入层和输出层	5
5.1.4 训练集和验证集	5
5.1.5 <i>LSTM</i> 神经网络的参数	6
5.1.6 <i>LSTM</i> 神经网络验证结果与误差分析	6
5.1.7 <i>LSTM</i> 神经网络预测结果	7
5.2 问题二	10
5.2.1 数据预处理	10
5.2.2 模型二的建立	11
5.2.3 模型的求解	11
5.3 问题三	12
5.3.1 问题三模型的建立与求解	12
5.3.2 问题三的结果分析	13
5.4 问题四	14
5.4.1 模型评价指标的构建	14
5.4.2 模型权重指标的确定	15
5.4.3 模型四的求解	15
六、模型的评价与改进	18
参考文献	20
附录 1：代码	21

一、问题重述

1.1 问题背景

我国作为世界上拥有最大电商物流网络的国家之一，电商物流网络的范围十分广泛，电商物流正在逐渐成为我国运输行业的主要范畴。然而，由于我国国土面积大、地形复杂、人口众多，电商物流还存在许多挑战与机遇。比如，我国的电商物流网络很容易受到一些自然因素的影响，如地震，这些因素在一定程度上会影响到各条线路运输的包裹数量，如果不能及时制定计划应对这些变化，很可能会带来不可估量的经济损失。因此预测各物流场地及线路的包裹数量，根据预测结果进行提前安排运输方案具有十分重要的意义。

1.2 问题提出

问题一：建立线路货量的预测模型，对 2023 - 01 - 01 至 2023 - 01 - 31 期间每条线路每天的货量进行预测，给出线路 $DC14 \rightarrow DC10$ 、 $DC20 \rightarrow DC35$ 、 $DC25 \rightarrow DC62$ 的预测结果。

问题二：在问题 1 的基础上，如果物流场地 $DC5$ 于 2023 - 01 - 01 开始关停，将 $DC5$ 相关线路的货量分配到其他线路使所有包裹尽可能正常流转且在保证各条线路的工作负荷尽可能均衡时使 $DC5$ 关停前后货量发生变化的线路尽可能少。如果存在部分日期部分货量没有正常流转，应使未能正常流转的包裹数量尽可能少。正常流转时，给出因 $DC5$ 关停导致货量发生变化的线路数及网络负荷情况；不能正常流转时，给出因 $DC5$ 关停导致货量发生变化的线路数、不能正常流转的货量及网络的负荷情况。

问题三：在问题 1 的基础上，如果被关停的物流场地为 $DC9$ ，同时允许对物流网络结构进行动态调整（每日均可调整），调整措施为关闭或新开线路，不包含新增物流场地，新开线路的运输能力的上限为已有线路运输能力的最大值。将 $DC9$ 相关线路的货量分配到其他线路，使所有包裹尽可能正常流转，并在各条线路得工作负荷尽可能均衡使得 $DC9$ 关停前后货量发生变化的线路尽可能少。如果存在部分日期没有满足要求的流转方案，应使未能正常流转的包裹日累计数量尽可能少。正常流转时，请给出因 $DC9$ 关停导致货量发生变化的线路数及网络负荷情况；不能正常流转时，请给出因 $DC9$ 关停导致货量发生变化的线路数、不能正常流转的货量及网络的负荷情况；同时给出每天的线路增减情况。

问题四：根据附件 1，请对该网络的不同物流场地及线路的重要性进行评价；为了改善网络性能，如果打算新增物流场地及线路，结合问题 1 的预测结果，分析新增物流场地应与哪几个已有物流场地之间新增线路，新增物流场地的处理能力及新增线路的运输能力应如何设置？考虑到预测结果的随机性，进一步探讨所建网络的鲁棒性。

二、问题的分析

2.1 问题的整体分析

以上问题是建立在预测模型的基础之上，首先通过时间序列预测模型预测出各条线路在 2023 年一月份每天的货量。由于附件 1 给出的数据有很多异常数据和缺失数据，故在进行预测时首先要进行数据处理。其次，如果 *DC5* 物流场地关停，在不能新增线路和物流场地的条件下我们要设计方案使货量尽可能地正常流转、受到影响的线路尽可能少。为了获得最优解，我们将此问题建模为非线性双层规划问题，建立约束条件采用层次粒子群算法进行求解。再次，如果 *DC9* 物流场地关停，我们可以对物流网络结构进行动态的调整，设计方案使得 *DC9* 关停后货量发生变化的线路数尽可能少，且使各物流场地日累计货量尽可能少。最后，根据附件 1 我们需要对此网络的物流场地及线路的重要性进行评价，我们将此问题建模为评价类模型，对于物流场地，以两年内接收的货物总量、与其他物流场地的关联度、两年内发送的货物总量和货量处理能力上限为评价指标。对于线路，以两年内货运总数、两年内的使用次数和货量运输能力上限为评价指标。

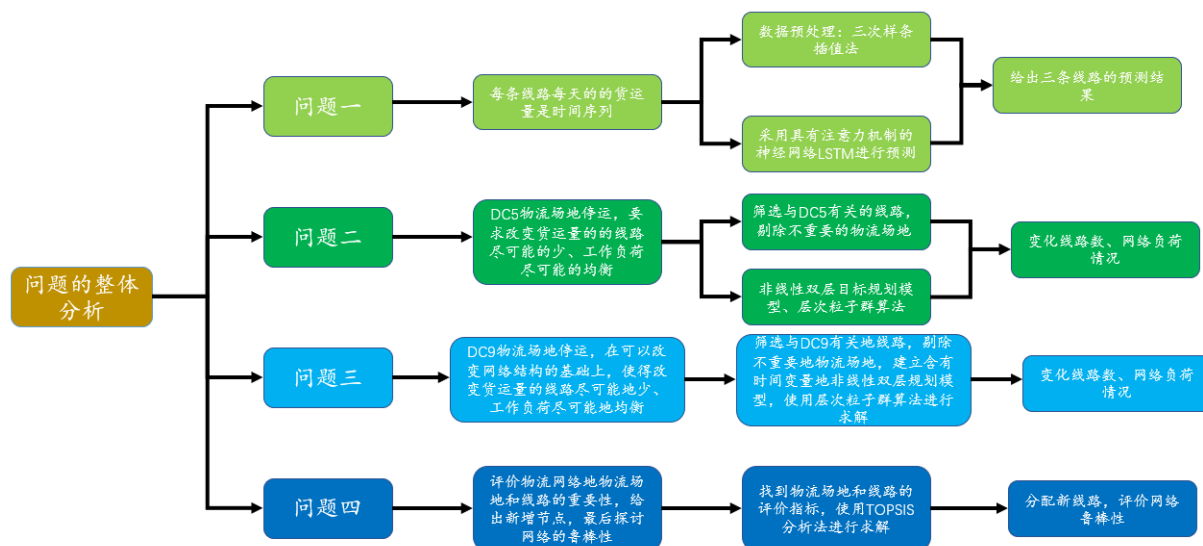


图 1 整体思路分析

2.2 问题一的分析

本题需要建立线路货量的预测模型，对每条线路在指定时间段进行货量预测。通过观察附件 1，可以发现对于多条线路，在 2021 - 01 - 01 至 2022 - 12 - 31 期间有很多缺值，并且含有很多异常数据，对此，我们对缺值和异常数据使用 3 次样条插值法^[1]进行插值。此外，我们建立了能够进行时间序列预测的 *LSTM* 神经网络模型^[2]，有大量的研究表明，*LSTM* 神经网络对时间序列进行预测有很好的效果。

2.3 问题二的分析

在问题 1 的基础上，研究 DC5 物流场地关停后对货量运输以及各物流场地处理包裹能力的影响。DC5 关停后，对本来运输到 DC5 的物流场地和运输线路有直接的影响，可能会造成网络不能正常流转，我们应尽量使原本运输到 DC5 的货量合理地分配到其它线路。我们的目标是让 DC5 关停前后货量发生变化的线路尽可能少，同时也保证未能正常流转的包裹日累计量尽可能少。为了解决此问题，在满足相应的约束条件下，我们采用非线性双层规划模型，上下层目标函数分别为使货量发生变化的线路数最少和货量日累计总量最少。使用层次粒子群算法^[4]进行求解。

2.4 问题三的分析

在问题 1 和问题 2 的基础上，DC5 变回正常流转，但 DC9 关停对货量运输以及各物流场地处理包裹能力的影响。对此，我们可以适当的改变网络结构来优化网络结构，并引入时间变量和线路变化变量建立非线性双层规划模型，最后使用层次粒子群算法进行求解。

2.5 问题四的分析

根据附件 1 我们需要对不同物流场地及线路的重要性进行评价，我们使用基于熵权 TOPSIS 模型^[7]对各物流场地和线路进行评价；对于线路，我们使用两年内的货运总数、两年内的使用次数和货量运输能力上限作为评价指标；对于物流场地，我们使用两年内接收的货物总量、与其他物流场地的关联度、两年内发送的货物总量和货量处理能力上限作为评价指标。结合问题 1，我们需要通过增加物流场地和线路来改善网络性能，最后对我们的物流网络进行鲁棒性评价。

三、模型的假设

1. 假设附件 1 的历史货量数据都是可靠的，可以反映未来货量的变化趋势；
2. 假设物流场地的关停对电商物流网络的影响主要是货量分配不均和部分货物无法正常流转；
3. 假设货量数据具有一定周期性、趋势性和随机性；
4. 假设货量分配方案总可以使得所有包裹尽可能得进行流转，且保持各线路得的工作负荷尽可能的均衡；
5. 假设各个物流场地的处理能力和各条线路的运输能力限制是可靠的；
6. 假设各个物流场地和各条线路在预测时间内能正常运作；
7. 不考虑各物流场地 2021 - 01 - 01 至 2022 - 12 - 31 的货物屯积量对 2023 年 1 月份的影响；

四、符号说明

符号	符号说明
P_{ij}	i 物流场地运送到 j 物流场地的货量预测值
Q_{ij}	因某物流场地关停后, 物流场地 j 接收来自 i 物流场地的货量
M_j	j 物流场地最大接收货量
i	发送物流场地
j	接收物流场地
e	信息熵值
d	信息效用值
D^+	正理想解距离
D^-	负理想解距离
C	综合度得分
a_{i1}	某线路两年内的货运总数
a_{i2}	某线路货量运输能力上限
a_{i3}	某线路两年内的使用次数
b_{i1}	某物流场地两年内接收的货物总量
b_{i2}	某物流场地与其他物流场地的关联度
b_{i3}	某物流场地两年内发送的货物总量
b_{i4}	某物流场地的货量处理能力上限

五、模型的建立与求解

5.1 问题一

5.1.1 数据预处理

根据附件 1 首先对每条线路的异常数据进行筛选和剔除, 剔除原则如下:

- ◀ 剔除每条线路货量变化幅度非常大的数据;
- ◀ 剔除每条线路非常离散的数据;

其次, 对处理过异常数据的数据进行插值, 我们采用在机器学习时间序列预测中效果很好的 3 样条插值法^[1] 进行插值。此方法的核心是将 $n+1$ 个数据点划分成 n 个区间, 每个区间构造多项式函数, 其形式如下式:

$$Y_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \quad i = 0, 1, \dots, n-1 \quad (1)$$

根据 3 次样条插值法每个分段函数之间的连接要求, 生成系数方程, 求解系数后即可进行插值处理。

5.1.2 问题一模型的建立

LSTM（*Long Short Term Memory*）长短期记忆神经网络是一种可以用于对时间序列进行分析和预测的神经网络，他很好的解决了一般递归神经网络的长期依赖问题。*LSTM* 长短期记忆神经网络具有十分强大的学习能力，十分适合用于解决间隔和延迟相对较长的时间序列^[3]。

5.1.3 输入层和输出层

LSTM 网络具有处理时间段序列的能力，我们使用经过插值处理和异常值剔除后的时间序列作为模型的输入，通过多个隐藏层，隐藏层中含有“遗忘门”，遗忘门具有可以决定什么样的信息应该被遗忘的能力。最后，此网络会输出 2023-01-01 到 2023-01-31 每条线路的货量预测数据。

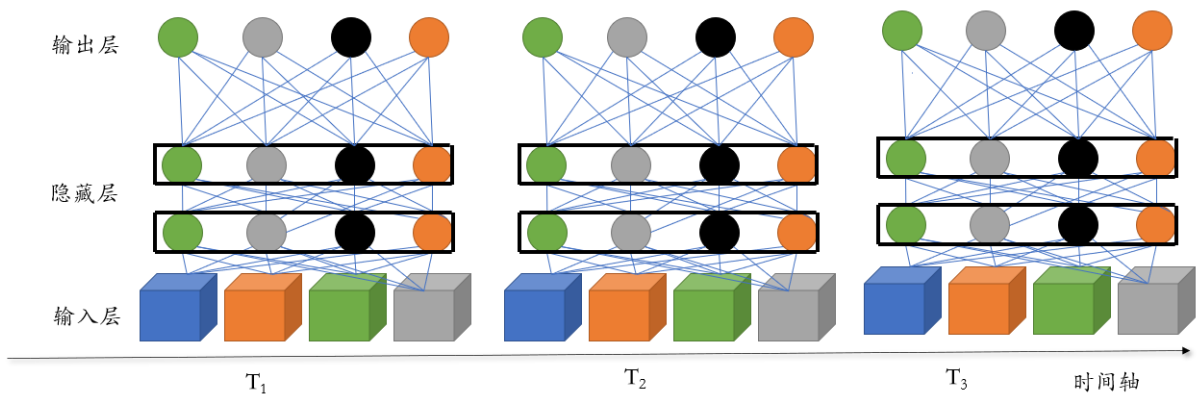


图 2 LSTM 网络结构

表 1 *LSTM* 神经网络时间序列预测模型中的参数

<i>MaxEpochs</i>	<i>GradientThreshold</i>	<i>InitialLearnRate</i>
1000	1	0.01
<i>LearnRateSchedule</i>	<i>LearnRateDropPeriod</i>	<i>LearnRateDropFactor</i>
0.2	400	0.15

5.1.4 训练集和验证集

对于每条线路的货量数据，经过异常值剔除和插值后将 2021-01-01 到 2022-11-30 的货量数据划分分训练集，2022-12-01 到 2022-12-31 的货量数据用于验证集。

5.1.5 *LSTM* 神经网络的参数

LSTM 神经网络时间序列预测模型中的参数主要包括最大训练回合数 *MaxEpochs*、训练梯度阈值 *GradientThreshold*、初始学习率 *InitialLearnRate*、学习率系数因子 *LearnRateSchedule*（用于降低学习率）、乘法之间的纪元数 *LearnRateDropPeriod* 本文设置的 *LSTM* 神经网络预测模型中的参数如表 1 所示：

5.1.6 *LSTM* 神经网络验证结果与误差分析

使用 *LSTM* 神经网络进行数据训练后，为了观察模型的准确度，我们使用预测值对比了 *DC14* \rightarrow *DC10*、*DC20* \rightarrow *DC35*、*DC25* \rightarrow *DC32* 这三条线路在 2022 年 12 月份的运货量，结果见图 3。表 2、表 3 和表 4 分别列出了三条线路的验证误差。

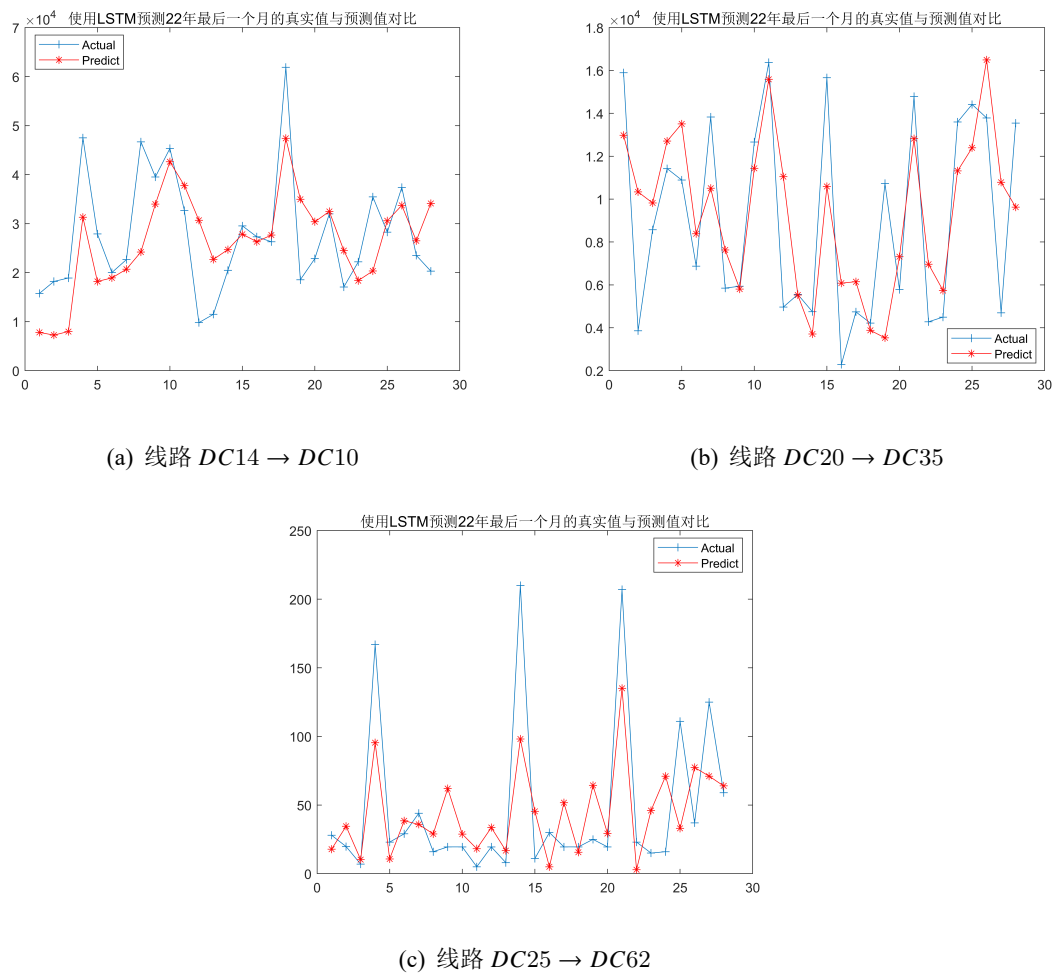


图 3 *LSTM* 网络最后一个月验证对比图

表 2 $DC14 \rightarrow DC10$ 预测对比

日期	实际货量	预测货量	误差	平均误差
2022-12-01	15762	10801	30.3 %	
2022-12-02	18204	16248	10.8 %	
...	
2022-12-14	20467	24700	19.8 %	15 %
...	
2022-12-30	23519	26503	12.5 %	

表 3 $DC25 \rightarrow DC62$ 预测对比

日期	实际货量	预测货量	误差	平均误差
2022-12-01	15893	12468	21 %	
2022-12-02	3589	2468	31.2 %	
...	
2022-12-13	5554	5756	3.6 %	25.2 %
...	
2022-12-30	13544	9950	22.1 %	

表 4 $DC20 \rightarrow DC35$ 预测对比

日期	实际货量	预测货量	误差	平均误差
2022-12-01	28	19	30. %	
2022-12-02	19.75	28	40 %	
...	
2022-12-13	44	35	18 %	19 %
...	
2022-12-30	59	63	8 %	

5.1.7 $LSTM$ 神经网络预测结果

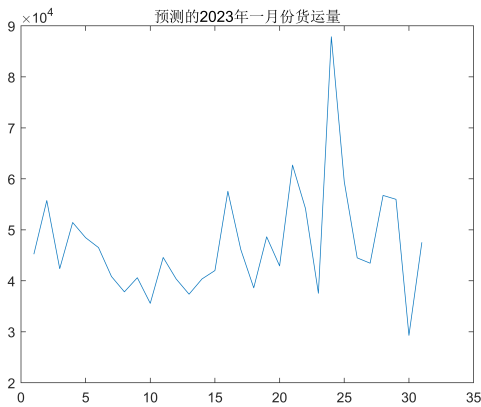
图 4 给出了题中所要求的三条线路在 2023 年 1 月份预测货量数据的折线图。表 5 和表 6 列举了题中所要求的三条线路在 2023 年 1 月份每天的货量预测值。

表 5 三条线路 2023 年 1 月份的预测结果

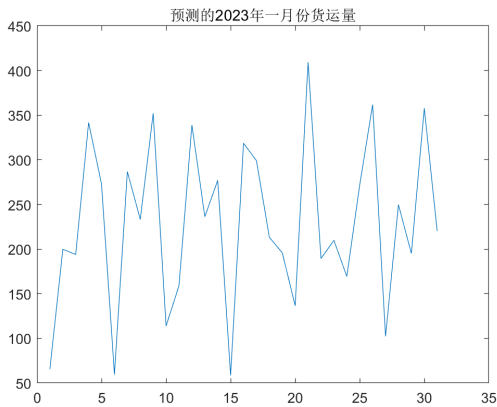
日期	$DC25 \rightarrow DC62$	$DC14 \rightarrow DC10$	$DC20 \rightarrow DC35$
01-01	12236	45218	66
01-02	11070	55746	200
01-03	11769	42380	194
01-04	6931	51440	342
01-05	6259	48461	273
01-06	10214	46513	60
01-07	8367	40851	287
01-08	9451	37835	233
01-09	18319	40607	352
01-10	10835	35580	114
01-11	12229	44583	159
01-12	10253	40370	339
01-13	7520	37374	236
01-14	6981	40364	277
01-15	11010	42010	59
01-16	7816	57560	319
01-17	14361	46130	299
01-18	15755	38617	213
01-19	6074	48625	196
01-20	3830	42921	137
01-21	6288	62698	409
01-22	10221	54185	189
01-23	10540	37553	210
01-24	11049	87867	169

表 6 续表 5

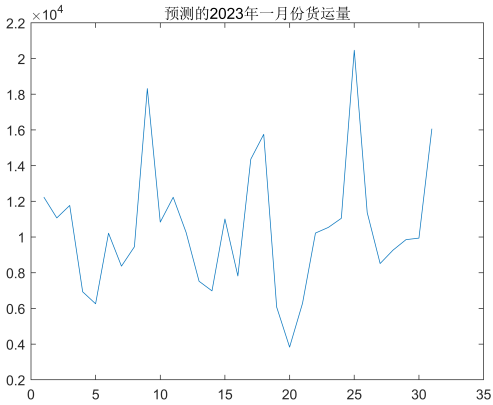
日期	$DC25 \rightarrow DC62$	$DC14 \rightarrow 10$	$DC20 \rightarrow DC35$
01-25	20466	59433	272
01-26	11374	44497	362
01-27	8514	43452	102
01-28	9267	56746	250
01-29	9853	55980	195
01-30	9938	29292	358
01-31	16068	47550	220



(a) 线路 $DC14 \rightarrow DC10$



(b) 线路 $DC20 \rightarrow DC35$



(c) 线路 $DC25 \rightarrow DC62$

图 4 三条线路 2023 年 1 月的预测值

5.2 问题二

为了精确地建立模型，在 $DC5$ 物流场地关停后，我们定义以下变量并做出适当的假设：

- 定义二值变量 x_{ij} 表示 $DC5$ 关停后从物流场地 i 到物流场地 j 运输的货量是否改变（1 表示改变；0 表示不变）；
- 定义 N 是与 $DC5$ 有联系的起始物流场地总数；
- 定义 X 是 $DC5$ 关停后货量可能发生变化的路线集合。
- 由于 2023 年 1 月份每天的线路不完全相同，相差极大，以某一天各线路的货量预测值来估计物流网络是否能正常流转会有较大偏差，不具有参考价值。于是我们以每条线路 2023 年一月份预测数据的平均值作为每条线路每天的预测值；通过分析每条线路月平均货量在 $DC5$ 关停后的变化来分析如何让物流网络尽可能地正常流转。
- 假设只通过调整与 $DC5$ 物流场地有联系的线路即可做到正常流转或未能正常流转时包裹日累计总量最少。

5.2.1 数据预处理

依据上文的假设我们首先找到在 $DC5$ 关停前 $DC5$ 作为目标物流地的线路、 $DC5$ 作为起始物流地的线路。其次，依据上文的假设我们可以对线路进行进一步筛选并对异常线路进行剔除，剔除原则如下：

- ▶ 剔除与 $DC5$ 物流场地无关的线路。
- ▶ 剔除月平均运输货量小于 50 的线路；
- ▶ 剔除 2023 年 1 月中超过 10 天预测值为 0 的线路。

经过数据处理，我们找到满足起始物流场地的线路有 5 条；满足目标物流场地的线路数有 73 条。

5.2.2 模型二的建立

为了解决此问题，我们使用非线性双层规划模型^[3]，以货量变化线路数最少和日累计货量最少为目标，双层目标函数和约束条件见下式：

$$\begin{cases} \text{Min} \sum_{ij \in X} x_{ij} \\ \text{s.t.} P_{i5} \leq \sum_{ij \in X} Q_{ij} x_{ij} \\ \text{Min} \sum_{i \in N} \left| P_{i5} - \sum_{ij \in X} Q_{ij} x_{ij} \right| \\ \text{s.t.} Q_{ij} \leq M_j \end{cases} \quad (2)$$

我们设 $\text{Min} \sum_{ij \in X} x_{ij}$ 为上层规划目标函数、 $\text{Min} \sum_{i \in N} \left| P_{i5} - \sum_{ij \in X} Q_{ij} x_{ij} \right|$ 为下从层规划目标函数。设 $\text{s.t.} P_{i5} \leq \sum_{ij \in X} Q_{ij} x_{ij}$ 为上层规划约束条件、 $\text{s.t.} Q_{ij} \leq m_j$ 为下层约束条件。非线性双层规划问题都有其自身的目标函数和约束条件，我们的目标是找到一个全局最优解使上下层目标函数达到相对平衡的最优。

5.2.3 模型的求解

为了求解上述非线性双层规划问题，我们采用层次粒子群算法^[5]，此算法的核心思想是可以通过不断的迭代求解双层规划中含有的两个优化问题，一方面在获得下层规划问题的最优解 y_d^* 求解上层规划问题的最优解 y_u^* ，另一方面则是在上层规划最优解 y_u^* 的基础上求解下层规划的最优解 y_d^{**} ，然后又再次将 y_d^{**} 传递到上层规划模型作为上层优化求解的基础，以此类推，获得双层规划问题的最优解，其求解过程中的传递动态方程见式（2）和式（3）

$$x_i(t+1) = x_i(t) + x_i(t)((AX(t))_i - X(t)^T AX(t)), i = 1, 2, \dots, n \quad (3)$$

$$y_i(t+1) = y_i(t) + y_i(t)((AY(t))_j - Y(t)^T AY(t)), j = 1, 2, \dots, m \quad (4)$$

各变量含义见表 7：结合上文中我们提出的模型，将层次粒子群算法用于求解，其

表 7 复制方程变量

$x_i(t)$	$X(t)$	$AX(t)_i$
上层总群 X 采用 i 的比例	上层种群混合策略状态向量	上层种群的收益矩阵
$y_i(t)$	$Y(t)$	$AY(t)_j$
下层种群 Y 采用策略 i 的比例	下层种群混合策略状态向量	上层种群的收益矩阵

具体求解步骤见图5。

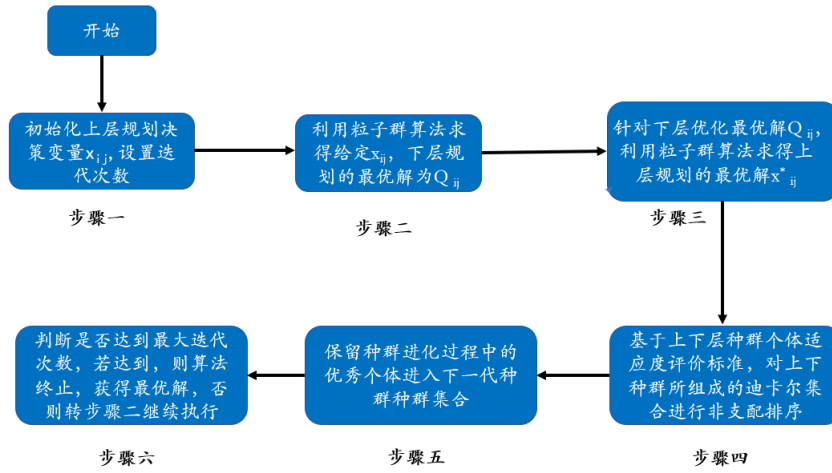


图 5 层次粒子群算法流程

经过 *MATLAB* 求解我们得出在 *DC5* 关停后, 物流网络可以正常流转, 且总改变线路数有 16 条, 具体线路规划和货量变化线路的工作负荷见表 8。且 *DC5* 关停后货量不发生变化的线路平均负荷为 8 %。整个物流网络的平均负荷为 19 %。

综上所述, *DC5* 关停后通过提前安排运输、改变线路运输可以使整个物流场地正常运转。

5.3 问题三

由附件 1 可知, 跟 *DC9* 物流场地相关的物流场地在 2021 到 2022 中, 从 2021-07-16 号开始出现数据异常, 因此决定将此日期以后的数据全部舍去, 只用此日期之前的数据进行预测。

在问题 2 的基础上, 为了精确地建模, 定义以下变量

- t 为 2023 年 1 月份中的第 t 天;
- Y 为第 t 天可能新增线路的集合;

5.3.1 问题三模型的建立与求解

为了解决此问题, 我们在问题 2 模型的基础上, 引入时间变量和线路变化变量, 建立新的双层目标函数和约束条件:

$$\left\{ \begin{array}{l} \text{Min} \sum_{i \in N} \sum_{j \in N} x_{ijt} + \sum_{ijt \in Y} Y_{ijt} \\ s.t. P_{i9t} \leq Q_{ijt} + \sum_{ijt \in Y} Q_{ijt} \\ \text{Min} \sum_{i \in N} \left| P_{i9t} - \sum_{j \in N} Q_{ijt} x_{ijt} - \sum_{j \in N} Q_{ijt} Y_{ijt} \right| \\ s.t. Q_{ijt} \leq M_j - p_{ijt} \end{array} \right. \quad (5)$$

表 8 网络组成流转线路规划

数量	变换线路	线路负荷量
1	$DC14 \rightarrow DC9$	100 %
2	$DC17 \rightarrow DC10$	100 %
3	$DC36 \rightarrow DC10$	96 %
4	$DC23 \rightarrow DC10$	95 %
5	$DC10 \rightarrow DC4$	91 %
6	$DC8 \rightarrow DC10$	90 %
7	$DC62 \rightarrow DC8$	85 %
8	$DC19 \rightarrow DC4$	84 %
9	$DC22 \rightarrow DC10$	53 %
10	$DC20 \rightarrow DC10$	46 %
11	$DC60 \rightarrow DC8$	43 %
12	$DC73 \rightarrow DC4$	35 %
13	$DC25 \rightarrow DC9$	24 %
14	$DC27 \rightarrow DC8$	20 %
15	$DC51 \rightarrow DC9$	18 %
16	$DC15 \rightarrow DC62$	15 %

使用与问题 2 相同的层次粒子群算法进行求解，经过 *MATLAB* 求解我们得出在 $DC9$ 关停后，我们可以通过调整网络结构，使上下层目标函数取得全局最优。其中增加的线路有 3 条，变换货量数的线路有 12 条，具体变化情况如表 9 所示。

5.3.2 问题三的结果分析

尽管在问题 3 中，可以每天新增线路和关闭原有线路，从而使目标函数 2 各物流场地的货物日累计量减少，但目标函数 1 变化线路总数会随之增大。对此我们采用层次粒子群算法对该双层目标函数进行求解，权衡两个目标函数在满足约束条件下相互影响的全局最优解，即 2023-01-01 至 2023-01-31 期间的每天线路均以表 9 的形式变化为满足条件的全局最优解。

表 9 网络组成流转线路规划

	货量变换线路	增加的线路
1	$DC14 \rightarrow DC8$	$DC56 \rightarrow DC14$
2	$DC14 \rightarrow DC9$	$DC15 \rightarrow DC26$
3	$DC51 \rightarrow DC8$	$DC4 \rightarrow DC14$
4	$DC25 \rightarrow DC10$	$DC16 \rightarrow DC36$
5	$DC15 \rightarrow DC62$	$DC66 \rightarrow DC8$
6	$DC44 \rightarrow DC10$	
7	$DC46 \rightarrow DC10$	
8	$DC47 \rightarrow DC35$	
9	$DC26 \rightarrow DC4$	
10	$DC48 \rightarrow DC10$	
11	$DC45 \rightarrow DC35$	
12	$DC58 \rightarrow DC9$	

5.4 问题四

为了解决此问题，我们建立基于熵权的 *TOPSIS* 模型^[6] 对各线路和物流场地进行评价。此模型通过计算接近度确定事物的重要度，此模型具有能够充分利用原始数据、信息量损失小的优点。

5.4.1 模型评价指标的构建

影响物流场地和线路的重要度的因素众多，体现在各个方面，经过调查研究物流场地和线路的评价指标如下表所示。

表 10 物流场地和线路的评价指标

	物流场地	线路
评价指标：	两年内接收的货物总量	两年内线路运输总货量
评价指标：	与其他物流场地的关联	线路最大运输货量
评价指标：	两年内发出货物总量	使用次数
评价指标：	两年内与其他物流场地的关联次数	

5.4.2 模型权重指标的确定

由于各评价指标的重要性不相同，故应该确定各评价指标的比例权重，具体步骤如下：

- (1) 根据线路和物流场地以及各自的评价指标构建初始矩阵，此矩阵有 2 个评价对象和 4 个评价指标；
- (2) 因指标的量纲不同，需要进行归一化得到规范化判断矩阵；
- (3) 权重的确定。根据熵的定义计算各指标的熵值 $E_j = \frac{1}{\ln m} \sum_{i=1}^m x_{ij} \ln r_{ij}$ (其中 r_{ij} 是 (2) 处理的结果)，定义一个指标冗余度 $F_j = 1 - E_j$ ；则各指标的熵权为：

$$W_j = \frac{F_j}{\sum_{j=1}^n F_j} \tag{6}$$

经过上述处理分别得到线路和物流产地各评价指标的权重，具体权重如下表所示。

表 11 物流场地各评价指标权重

评价指标	b_{i1}	b_{i2}	b_{i3}	b_{i4}
权重	48.45 %	7.738 %	27.266 %	16.146 %

表 12 线路各评价指标权重

评价指标	a_{i1}	a_{i2}	a_{i3}
权重	48.295 %	34.815 %	16.89 %

5.4.3 模型四的求解

- 定义 D_+ 和 D_- 值，此两值分别代表评价对象与最优解和最劣解的距离， D_+ 值越大，说明与最优解距离越远； D_i 值越大，说明与最劣解距离越远。
- 定义综合得分 C 值， $C = (D_i)/(D_+ + D_-)$ ， C 值越大说明研究对象越重要。

经过 *MATLAB* 求解我们得出了 1049 条线路和 81 个物流场地的 D_+ 、 D_- 和 C ，因篇幅有限，只列出部分数据，具体如下表所示。

由以上数据可知，对于所有线路， $DC14 \rightarrow DC8$ 得分最高，故最重要；对于所有物流场地， $DC10$ 和 $DC14$ 得分最高，故最重要。

结合问题 1，我们再次使用预测数据对 2023 年 1 月份各物流场地进行 *TOPSIS* 评价，得到各物流场地的综合得分如表 15 所示：

由下表数据可知，物流场地 $DC8$ 、 $DC14$ 、 $DC4$ 和 $DC10$ 在 2023 年 1 月份的重要性最高。综合考虑各种要求，我们决定在物流场地 $DC8$ 、 $DC14$ 、 $DC4$ 和 $DC10$ 这四个物流场地附近新增一个物流场地 $DC82$ 。

表 13 各线路综合得分

线路	D_+	D_-	综合得分指数
$DC14 \rightarrow DC8$	0.110	0.959	0.897
$DC14 \rightarrow DC9$	0.482	0.610	0.559
$DC36 \rightarrow DC10$	0.565	0.545	0.491
$DC36 \rightarrow DC4$	0.532	0.494	0.481
$DC17 \rightarrow DC10$	0.600	0.515	0.462
$DC30 \rightarrow DC14$	0.896	0.411	0.314
$DC14 \rightarrow DC4$	0.785	0.360	0.314
$DC50 \rightarrow DC14$	0.896	0.411	0.314
$DC43 \rightarrow DC14$	0.890	0.408	0.314
$DC49 \rightarrow DC10$	0.986	0.027	0.027
$DC56 \rightarrow DC4$	0.989	0.024	0.024
$DC13 \rightarrow DC9$	0.99	0.023	0.023
\vdots	\vdots	\vdots	\vdots

表 14 各物流场地综合得分

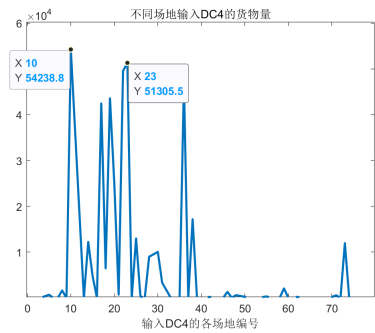
物流场地	D_+	D_-	综合得分指数
$DC10$	0.383	0.856	0.690
$DC14$	0.790	0.685	0.68
$DC4$	0.545	0.778	0.588
$DC8$	0.497	0.645	0.564
$DC62$	0.696	0.383	0.355
$DC5$	0.805	0.339	0.296
$DC9$	0.766	0.319	0.294
\vdots	\vdots	\vdots	\vdots

结合问题 1，我们可以分别得到物流场地 $DC8$ 、 $DC14$ 、 $DC4$ 和 $DC10$ 在 1 月份的日平均输入货量和输出货量。具体数据如图 6 所示。

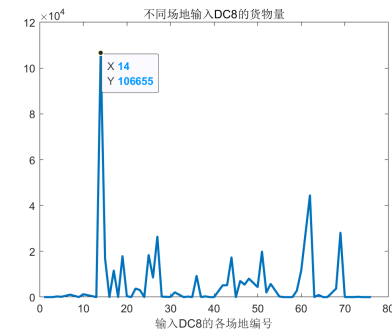
从上图可以看出，对于 4 个重要性最高的物流场地有几个输入或输出货量最大的线

表 15 2023 年 1 月份物流场地综合得分

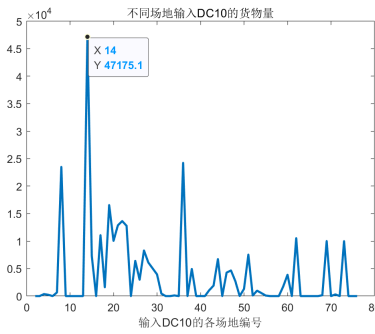
物流场地	综合得分指数
DC8	0.662
DC14	0.660
DC4	0.613
DC10	0.613
DC62	0.54
DC9	0.50
DC5	0.39
⋮	⋮



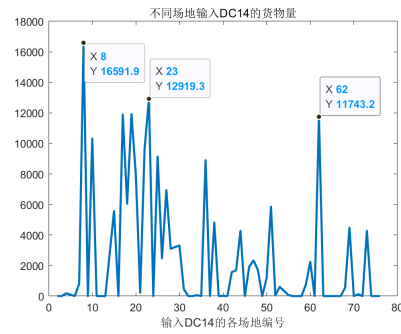
(a) DC4 日平均输入货量



(b) DC8 日平均输入货量



(c) DC10 日平均输入货量



(d) DC14 日平均输入货量

图 6 4 个物流场地日平均输入货量

路，于是我们将这部分线路中的货量以 50 % 的配比分给新的线路，新开的线路如表 16 所示。

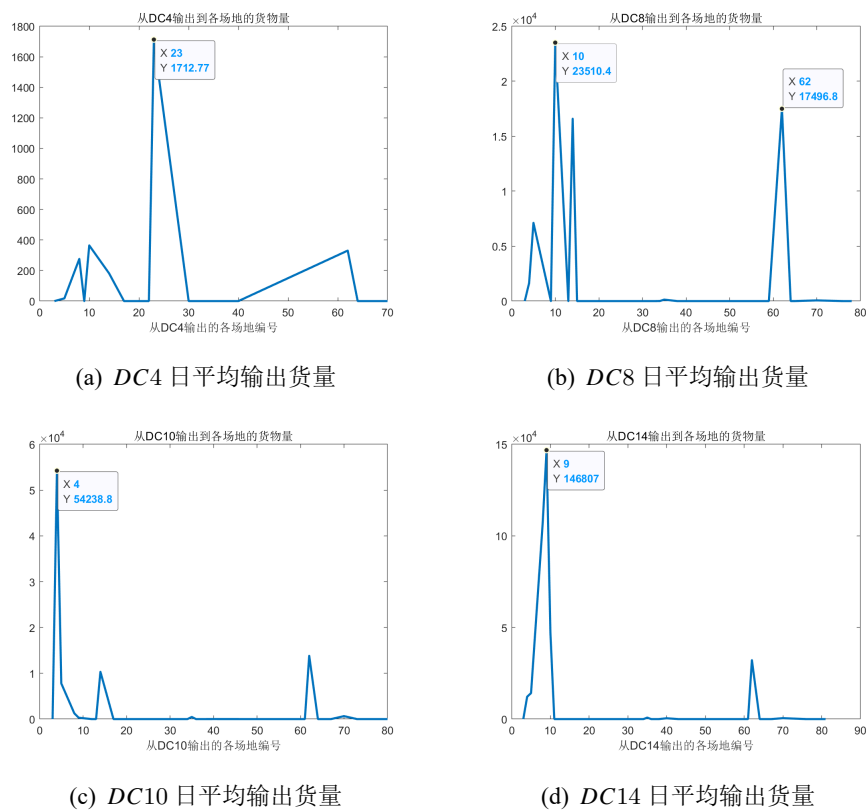


图 7 4 个物流场地日平均出货量

表 16 新增线路

$DC14 \rightarrow DC82$	$DC23 \rightarrow DC82$
$DC62 \rightarrow DC82$	$DC10 \rightarrow DC82$
$DC23 \rightarrow DC82$	$DC82 \rightarrow DC10$
$DC82 \rightarrow DC14$	$DC82 \rightarrow DC62$
$DC82 \rightarrow DC4$	$DC82 \rightarrow DC9$

六、模型的评价与改进

模型的优点：

- 为了更加准确地预测时间序列，我们对所有的数据进行了预处理，剔除了部分的异常数据并进行了数据填充。
- 问题一中使用 *LSTM* 神经网络具有强大的时间序列预测的能力，该神经网络通常比传统的时间序列预测算法效果更好。
- 问题二和问题三中使用层次粒子群算法可以很好地解决非线性双层目标规划。

- 问题四使用的 *TOPSIS* 模型能够充分利用原始数据信息，结果能精确地反映各评价方案之间的差距且计算简单易行。

模型的缺点：

- 假设条件较多，可能会对模型的普遍性造成影响。
- 模型求解时间过长，消耗计算资源过多。
- 问题二中为了简化模型的建立，我们仅对各线路日均流转量进行讨论与分析，并以此给出需要调整的路线数和网络工作负荷。
- 由于数据量过多，处理十分复杂，不同的模型所要求的数据不同，故我们未将建立的模型与其他模型进行对比。

模型的改进：

- 问题一中的 *LSTM* 时间序列预测模型可与其他预测模型进行结合和优化。最后的预测值应与其他模型的预测值进行加权平均，从而提高预测的准确性。
- 在问题二和问题三中减少假设，从而提高模型的普遍性。

参考文献

- [1] 王丽娟, 所辉. 时间序列预测中插值法的使用与分析 [J]. 电脑编程技巧与维护, 2022, No.449(11):13-15. DOI:10.16184/j.cnki.comprg.2022.11.046.
- [2] 常昊. 基于 LSTM 神经网络的地铁短时客流量预测研究 [D]. 西京学院, 2022. DOI:10.27831/d.cnki.gxjxy.2022.000059.
- [3] 程肇兰, 张小强, 梁越. 基于 LSTM 网络的铁路货运量预测 [J]. 铁道学报, 2020, 42(11):15-21.
- [4] 李昌兵, 杜茂康, 付德强. 基于层次粒子群算法的非线性双层规划问题求解策略 [J]. 系统工程理论与实践, 2013, 33(09):2292-2298.
- [5] 张晓涵. 基于层次粒子群算法的地下物流多级节点选择规划研究 [J]. 中国储运, 2021(05):162-164. DOI:10.16301/j.cnki.cn12-1204/f.2021.05.063.
- [6] 尹传忠, 张祥栋, 曲思源, 赵明, 刘永东. 基于熵权 TOPSIS 模型的长三角地区规划铁路评价探讨 [J]. 铁道运输与经济, 2021, 43(01):77-82+101. DOI:10.16668/j.cnki.issn.1003-1421.2021.01.13.
- [7] 刘玘玘. 基于熵权 TOPSIS-耦合协调度-灰色关联度的长江经济带水资源承载力综合评价 [J/OL]. 武汉大学学报 (工学版):1-15[2023-04-16]. <https://kns-cnki-net.wvvpn.ncu.edu.cn/kcms/detail/42.1675.T.20230410.1432.004.html>

附 录

附录 1：代码

问题一的代码：

```
%lstm_test_3.m文件
clc;
clear;
datatest1= readmatrix('D:\lstm_pred\test_14-10.xlsx');
data=datatest1(:,4)';
n=length(data);
train_data = data(1:n-29); %用前23个月作为训练集
Validate_data = data(n-30:n); %最后一个月的数据用来作为验证集
%对数据进行标准化
mean_value = mean(train_data);
std_value = std(train_data);
standardized_train_data = (train_data - mean_value) / std_value;
train_x = standardized_train_data(1:end-1); %输入的每个时间步，LSTM网络学习预测下一个时间步
train_y = standardized_train_data(2:end);
%创建LSTM回归网络，并设置较优参数
features_num = 1;
response_num = 1;
hidden_unit_num = 500;
def_layers();
def_options();%定义LSTM函数的参数
net = trainNetwork(train_x,train_y,layers,options);
%对神经网络初始化
net = predictAndUpdateState(net,train_x); %将新的train_x数据用在网络上进行初始化网络状态
[net,predict_y] = predictAndUpdateState(net,train_y(end));
%用训练的数据预测2022年最后一个月的数据以及23年一月的数据
for i = 2:62
    [net,predict_y(:,i)] = predictAndUpdateState(net,predict_y(:,i-1),'ExecutionEnvironment','cpu');
end
predict_y = std_value*predict_y + mean_value;
%计算RMSE判断训练效果
rmse = sqrt(mean((predict_y(1:31)-Validate_data).^2)) ;
%显示预测的2023-01-01到2023-01-31的货运量
figure(1)
January=1:31;
plot(January,predict_y(32:62))
```

```

title('预测的2023年一月份货运量');
figure(2)
plot(1:n,data,'b',January+n,predict_y(32:62))
xlswrite('DC20_DC35_predict.xlsx',[January; predict_y(32:62)]');

```

%def_options.m文件

```

options = trainingOptions('adam', ...
    'MaxEpochs',1000, ...
    'GradientThreshold',1, ...
    'InitialLearnRate',0.01, ...
    'LearnRateSchedule','piecewise', ...
    'LearnRateDropPeriod',400, ...
    'LearnRateDropFactor',0.15, ...
    'Verbose',0, ...
    'Plots','training-progress');

```

%def_layers.m文件

```

layers = [ ...
    sequenceInputLayer(features_num) %输入层
    lstmLayer(hidden_unit_num) % lstm层
    fullyConnectedLayer(response_num) %为全连接层,是输出的维数。
    regressionLayer];

```

问题二的代码:

%Question2.m文件

```

clc;
clear;
two_years=xlsread("D:\MATLAB\special_for_Matlab\2023_test\所有路线流量数据处理结果.
    xlsx");
all_data=xlsread("D:\MATLAB\special_for_Matlab\2023_test\predict_data.xlsx");
all_end5=xlsread("D:\MATLAB\special_for_Matlab\2023_test\clear_end5.xlsx");
%各路线的阈值
max_produce=two_years(:,3);
%用一个月的平均值代替每天的
for i=1:size(all_end5,1)%平均运到5的货量
    all_end5(i,2)=sum(all_end5(i,3:end))/31;
end
for i=1:size(all_data,1)
    all_data(i,34)=sum(all_data(i,3:end))/31;
end

```



```

%第34列换成他还能承载多少货物
all_data(:,34)=max_produce;
%将与5有关联的结点按照平均值降序排序
all_end5=sortrows(all_end5,-2);
cumulative_quantity=0;%货物累计量
change_num=0;%线路变化量
%%使用粒子群算法求解
n = 500;
narvs =size(all_data(:,1));
a1= 2.05; % 个体学习因子
a2 = 2.05; % 社会学习因子
C = a1+a2;
factor = 2/abs((2-C-sqrt(C^2-4*C)));
w_max = 0.9;
w_min = 0.4;
max_iterations = 500; % 迭代的次数
v_max = 15*ones(1,narvs); %最大速度
x_lower = -20*ones(1,narvs); %下界
x_upper = 20*ones(1,narvs); %上界

% 初始化粒子的位置和速度
x = zeros(n,narvs);
for i = 1: narvs
    x(:,i) = x_lower(i) + (x_upper(i)-x_lower(i))*rand(n,1);
end
v = -v_max + 2*v_max .* rand(n,narvs);
%计算粒子的适应度
fitness = zeros(n,1); % 初始化这n个粒子的适应度全为0
for i = 1:n
    fitness(i) = fun_1(x(i,:)); % 调用fun_1函数来计算适应度
end
best_x = x;
ind = find(fitness == min(fitness), 1);
best_g = x(ind,:);

for d = 1:max_iterations
    tem = best_g;
    for i = 1:n %对离子的速度和适应度进行更新
        f_i = fitness(i);
        f_avg = sum(fitness)/n;
        f_min = min(fitness);
        if f_i <= f_avg
            w = w_min + (w_max - w_min)*(f_i - f_min)/(f_avg - f_min);
        else

```

```

        w = w_max;
    end
    v(i,:) = factor * (w*v(i,:) + a1*rand(1)*(best_x(i,:) - x(i,:)) + a2*rand
        (1)*(best_g - x(i,:))); % 更新第i个粒子的速度
    x(i,:) = x(i,:) + v(i,:); %对第i个粒子的位置进行更新
    for j = 1: narvs %对那些超出范围的粒子进行更新
        if x(i,j) < x_lower(j)
            x(i,j) = x_lower(j);
        elseif x(i,j) > x_upper(j)
            x(i,j) = x_upper(j);
        end
    end
    fitness(i) = fun_1(x(i,:));
    if fitness(i) < fun_1(best_x(i,:))%判断是否需要更新粒子的最佳位置
        best_x(i,:) = x(i,:);
    end
    if fitness(i) < fun_1(best_g)
        best_g = best_x(i,:);
    end
end
best_fitness(d) = fun_1(best_g);
delta_fitness = abs(fun_1(best_g) - fun_1(tem)); % 计算相邻两次迭代适应度的变化
量
end
plot(best_fitness) % 绘制出每次迭代最佳适应度的变化图
xlabel('迭代次数');

%data_oprate.m文件

clc;clear;
data=xlsread("D:\MATLAB\special_for_Matlab\2023_test\predict_data.xlsx");
start5_index=find(data(:,1)==9);%找出所有从5出发的数据行号
end5_index=find(data(:,2)==9);%找出所有终点为5的数据行号
data_start5=round(data(start5_index,:));
data_end5=round(data(end5_index,:));
xlswrite('D:\MATLAB\special_for_Matlab\2023_test\data_start9.xlsx', data_start5);
xlswrite('D:\MATLAB\special_for_Matlab\2023_test\data_end9.xlsx', data_end5);
%数据处理,将30天内累计包裹量小于5的剔除
clear_start5=[];
for i =1:size(data_start5,1)
    if sum(data_start5(i,3:end))>5
        clear_start5=[clear_start5;data_start5(i,:)];
    end
end
end

```

```

clear_end5=[];
for i =1:size(data_end5,1)
    if sum(data_end5(i,3:end))>5
        clear_end5=[clear_end5;data_end5(i,:)];
    end
end
xlswrite('D:\MATLAB\special_for_Matlab\2023_test\clear_start9.xlsx', round(
    clear_start5));
xlswrite('D:\MATLAB\special_for_Matlab\2023_test\clear_end9.xlsx', round(
    clear_end5));

```

问题三的代码:

%Question3.m文件

```

clc;
clear;
two_years=xlsread("D:\MATLAB\special_for_Matlab\2023_test\所有路线流量数据处理结果.
    xlsx");
all_data=xlsread("D:\MATLAB\special_for_Matlab\2023_test\predict_data.xlsx");
all_end9=xlsread("D:\MATLAB\special_for_Matlab\2023_test\data_end9.xlsx");
%各路线的阈值
max_produce=two_years(:,3);
%用一个月的平均值代替每天的
for i=1:size(all_end9,1)%平均运到5的货量
    all_end9(i,2)=sum(all_end9(i,3:end))/31;
end
for i=1:size(all_data,1)
    all_data(i,34)=sum(all_data(i,3:end))/31;
end
%第34列换成他还能承载多少货物
all_data(:,34)=max_produce;
%将与5有关联的结点按照平均值降序排序
all_end9=sortrows(all_end9,-2);
cumulative_quantity=0;%货物累计量
change_num=0;%线路变化量
% 初始化粒子的位置和速度
x = zeros(n,narvs);
for i = 1: narvs
    x(:,i) = x_lower(i) + (x_upper(i)-x_lower(i))*rand(n,1);
end
v = -v_max + 2*v_max .* rand(n,narvs);
%计算粒子的适应度

```

```

fitness = zeros(n,1); % 初始化这n个粒子的适应度全为0
for i = 1:n
    fitness(i) = fun_1(x(i,:)); % 调用fun_1函数来计算适应度
end
best_x = x;
ind = find(fitness == min(fitness), 1);
best_g = x(ind,:);
for i=1:31
    for d = 1:max_iterations
        tem = best_g;
        for i = 1:n %对离子的速度和适应度进行更新
            f_i = fitness(i);
            f_avg = sum(fitness)/n;
            f_min = min(fitness);
            if f_i <= f_avg
                w = w_min + (w_max - w_min)*(f_i - f_min)/(f_avg - f_min);
            else
                w = w_max;
            end
            v(i,:) = factor * (w*v(i,:) + a1*rand(1)*(best_x(i,:) - x(i,:)) + a2*
                rand(1)*(best_g - x(i,:))); % 更新第i个粒子的速度
            x(i,:) = x(i,:) + v(i,:); %对第i个粒子的位置进行更新
            for j = 1: narvs %对那些超出范围的粒子进行更新
                if x(i,j) < x_lower(j)
                    x(i,j) = x_lower(j);
                elseif x(i,j) > x_upper(j)
                    x(i,j) = x_upper(j);
                end
            end
            fitness(i) = fun_1(x(i,:));
            if fitness(i) < fun_1(best_x(i,:))%判断是否需要更新粒子的最佳位置
                best_x(i,:) = x(i,:);
            end
            if fitness(i) < fun_1(best_g)
                best_g = best_x(i,:);
            end
        end
        best_fitness(d) = fun_1(best_g);
        delta_fitness = abs(fun_1(best_g) - fun_1(tem)); % 计算相邻两次迭代适应度的
            变化量
    end
end
end

```

问题四的代码:

```

%Question4_1.m文件

clc;
clear;
two_years=xlsread("D:\MATLAB\special_for_Matlab\2023_test\所有路线流量数据处理结果.
    xlsx");
path_data=zeros(1049,3);
for i=1:size(path_data,1)
    %计算两年内线路使用的次数
    index1=find(two_years(i,3:end)>0);
    path_data(i,1)=size(index1,2);
    %计算两年内线路运送货物数
    path_data(i,2)=sum(two_years(i,3:end));
    %计算线路最大货物量
    path_data(i,3)=max(two_years(i,3:end));
end
% xlswrite('D:\MATLAB\special_for_Matlab\2023_test\线路评价指标.xls',path_data)
%%物流场地的重要性：两年内接收货物总量、两年内发送货物总量、两年内与其他场地关联次数，
    与多少个场地关联
point_data=zeros(81,4);
for i=1:81
    %两年内接收的货物总量
    index2=find(two_years(:,2)==i);
    point_data(i,1)=sum(sum(two_years(index2,3:end)));
    %两年内发送货物总量
    index3=find(two_years(:,1)==i);
    point_data(i,2)=sum(sum(two_years(index3,3:end)));
    %两年内与其他场地的关联次数
    non_zero_num=nnz(two_years(index2,3:end))+nnz(two_years(index3,3:end));
    point_data(i,3)=non_zero_num;
    %与多少个场地关联
    temp1=two_years(index2,1);
    temp2=two_years(index3,2);
    point_data(i,4)=length(unique([temp1' temp2']));
end
% xlswrite('D:\MATLAB\special_for_Matlab\2023_test\物流场地评价指标.xls',
    point_data)
%基于熵权法的TOPSIS评价
%对路线进行评价
[x_num,y_num] = size(path_data);
%对数据进行标准化
stanD_path_data = path_data ./ repmat(sum(path_data.*path_data) .^ 0.5, x_num, 1);
%判断之前标准化后是否还存在负数，如果存在则再次标准化

```

```

if sum(sum(stand_path_data<0)) >0
    for i = 1:x_num
        for j = 1:y_num
            stand_path_data(i,j) = [path_data(i,j) - min(path_data(:,j))]/ [max(
                path_data(:,j)) - min(path_data(:,j))];
        end
    end
end
entropy_weight_path = entropy_weight_method(stand_path_data);
%计算D+、D-以及最后的得分
D_p = sum([(stand_path_data - repmat(max(stand_path_data),x_num,1)) .^ 2] .*
    repmat(entropy_weight_path,x_num,1),2) .^ 0.5;
D_n = sum([(stand_path_data - repmat(min(stand_path_data),x_num,1)) .^ 2] .*
    repmat(entropy_weight_path,x_num,1),2) .^ 0.5;
path_score = D_n ./ (D_p+D_n) % 未归一化的得分
stand_score = path_score / sum(path_score)
[path_sorted_score,index] = sort(stand_score , 'descend');

%对物流场地进行评价
[x_num,y_num] = size(point_data);
%对数据进行标准化
stand_point_data = point_data ./ repmat(sum(point_data.*point_data) .^ 0.5, x_num,
    1);
%判断之前标准化后是否还存在负数，如果存在则再次标准化
if sum(sum(stand_point_data<0)) >0
    for i = 1:x_num
        for j = 1:y_num
            stand_point_data(i,j) = [point_data(i,j) - min(point_data(:,j))]/ [max
                (point_data(:,j)) - min(point_data(:,j))];
        end
    end
end
entropy_weight_point = entropy_weight_method(stand_point_data);
%计算D+、D-以及最后的得分
D_p = sum([(stand_point_data - repmat(max(stand_point_data),x_num,1)) .^ 2] .*
    repmat(entropy_weight_point,x_num,1),2) .^ 0.5;
D_n = sum([(stand_point_data - repmat(min(stand_point_data),x_num,1)) .^ 2] .*
    repmat(entropy_weight_point,x_num,1),2) .^ 0.5;
point_score = D_n ./ (D_p+D_n);
stand_score = point_score / sum(point_score);%对得分进行归一化
[point_sorted_score,index] = sort(stand_score , 'descend');

%Question4_2.m文件

```

```

clc;
clear;
predict_January=xlsread("D:\MATLAB\special_for_Matlab\2023_test\预测一月份数据.
    xlsx");
%%物流场地的重要性：一个月内接收货物总量、一个月内发送货物总量、一个月内与其他场地关联
    次数,与多少个场地关联
point_data=zeros(81,4);
for i=1:81
    %一个月内接收的货物总量
    index2=find(predict_January(:,2)==i);
    point_data(i,1)=sum(sum(predict_January(index2,3:end)));
    %一个月内发送货物总量
    index3=find(predict_January(:,1)==i);
    point_data(i,2)=sum(sum(predict_January(index3,3:end)));
    %一个月内与其他场地的关联次数
    non_zero_num=nnz(predict_January(index2,3:end))+nnz(predict_January(index3,3:
        end));
    point_data(i,3)=non_zero_num;
    %与多少个场地关联
    temp1=predict_January(index2,1);
    temp2=predict_January(index3,2);
    point_data(i,4)=length(unique([temp1' temp2']));
end
xlswrite('D:\MATLAB\special_for_Matlab\2023_test\一月份预测指标.xlsx',point_data

%entropy_weight_method.m文件

function [W] = entropy_weight_method(A)
    [n,m] = size(A);
    D = zeros(1,m);
    for i = 1:m
        x_i = A(:,i);
        p = x_i / sum(x_i);
        Entropy = -sum(p .* def_log(p)) / log(n); % 计算信息熵
        D(i) = 1- Entropy;
    end
    W = D ./ sum(D); %进行归一化得到熵增法确定的系数
end

%def_log.m文件

function [Y] = def_log(X) %定义一个def_log函数，当输入的X中元素为0时，返回0
n = length(X);

```

```
Y = zeros(n,1);  
for i = 1:n  
    if X(i) == 0  
        Y(i) = 0;  
    else  
        Y(i) = log(X(i));  
    end  
end  
end
```