

DOI: 10.20035/j.issn.1671-2668.2024.03.007

引用格式:李翠.基于基向量线性表达的多步交通流预测[J].公路与汽运,2024,40(3):33-36+52.

Citation: LI Cui. Multi-step traffic flow prediction based on linear expression of basis vectors[J]. Highways & Automotive Applications, 2024, 40(3): 33-36+52.

基于基向量线性表达的多步交通流预测*

李翠

(江西交通职业技术学院 信息工程学院, 江西 南昌 330013)

摘要: 为提高交通流多步预测精度,提出一种考虑基向量独立预测和线性表达的交通流多步预测方法。利用交通流的周期性,以天为周期单位将历史交通流数据进行分割和矩阵表达,再利用协方差矩阵的奇异值分解生成一组基向量,将每天的交通流时间序列表达为该组基向量的线性组合,将交通流的预测问题转化为基向量组合系数的独立预测问题;采用回声状态网络(Echo State Networks, ESN)分别建立各基向量组合系数的独立预测模型,并采用 Jaya 算法优化 ESN 模型参数。结果表明,采用 Jaya 算法可以快速找到 ESN 模型最佳参数,采用文中方法可以实现交通流多步预测,且预测精度比历史均值法和 Jaya-ESN 法更高。

关键词: 公路交通;高速公路;交通流预测;基向量独立预测;线性表达;回声状态网络(ESN)

中图分类号: U491.14

文献标志码: A

文章编号: 1671-2668(2024)03-0033-04

智能交通系统(Intelligent Traffic System, ITS)运用高科技手段解决交通运输问题,是人工智能系统在交通运输领域应用的新兴技术^[1-2]。作为 ITS 的重要内容,交通流预测为交通控制和交通诱导提供重要数据支撑,是近年来的研究热点^[3]。当前研究主要通过建立各种预测模型来挖掘交通流历史数据蕴含的时空规律,实现准确的交通流预测。温惠英等基于实际交通流量数据建立相似性度量方法,搭建了 3 层长短时记忆网络多断面交通流预测模型^[4]。Zhang W. B. 等通过筛选交通流数据的时空特征生成二维输入矩阵,利用卷积神经网络构建了交通流预测模型^[5]。为减少交通流随机噪声的干扰,一般采用多尺度分解技术进行降噪。杨飞等通过小波分解提取占有交通流绝大部分能量的混沌低频成分,采用回声状态网络(Echo State Network, ESN),通过变化时延进行交通流多步预测^[6]。万玉龙等利用小波包分解生成交通流子序列,先结合粒子群优化(Particle Swarm Optimization, PSO)算法和 ESN 算法分别对各子序列单独开展预测,再采用加权平均法融合各子序列的预测值得到模型的最终预测结果^[7]。马莹莹等将集合经验模态分解和小波分析相结合分解交通流时间序列,对比分析分量组

合预测和分量单独预测的效果,结果表明单独预测方法的效果更好^[8]。前述研究大多适用于单步预测,多步预测精度较低。李慧等利用交通流的周期性特征,结合主成分降维、PSO 算法和 ESN 算法建立交通流多步预测模型,应用结果表明该模型的预测效果较好但存在主成分维度难以确定的问题^[9]。为提高预测效果,本文提出一种考虑基向量独立预测和线性表达的多步交通流预测方法,通过预测各个基向量的组合系数实现交通流的多步预测。

1 相关理论的基本原理

1.1 交通流的基向量表达

以矩阵形式表达的前 d 天历史交通流数据为:

$$Q = \begin{bmatrix} q_{1,1} & \cdots & q_{1,d} \\ \vdots & & \vdots \\ q_{s,1} & \cdots & q_{s,d} \end{bmatrix} \quad (1)$$

式中: q_{ji} ($j=1,2,\dots,s; i=1,2,\dots,d$) 表示第 i 天第 j 个采样点的交通流量; s 表示每天的采样点数量; d 表示总的天数。

对 Q 的协方差进行奇异值分解,得:

$$SVD(QQ^T) = \Theta \Lambda \Theta^T \quad (2)$$

式中: Θ 为酉矩阵, $\Theta = [\theta_1, \theta_2, \dots, \theta_s]$, 各个列向

* 基金项目:江西省交通运输厅科技项目(2020X0012;2023H0039)

量(或特征向量)构成一组 s 维空间的基向量; \mathbf{A} 为对角矩阵, $\mathbf{A} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_s)$; λ_i 为特征值, 且 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_s \geq 0$ 。

第 i 天的历史交通流数据可被各个基向量线性表达, 即:

$$\mathbf{Q}_i = \boldsymbol{\theta} \boldsymbol{\beta}_i = \sum_{j=1}^s \beta_{ji} \boldsymbol{\theta}_j \quad (3)$$

式中: $\boldsymbol{\beta}_i$ 为对应 \mathbf{Q}_i 的组合系数向量, $\boldsymbol{\beta}_i = [\beta_{1i}, \beta_{2i}, \dots, \beta_{si}]^T$ 。

前 d 天组合系数构成的矩阵为:

$$\boldsymbol{\beta} = \boldsymbol{\theta}^{-1} \mathbf{Q} \quad (4)$$

式中: $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_d]$; 上标 -1 表示对矩阵取逆。

$\boldsymbol{\beta}$ 的第 i 列(即 β_i)对应 \mathbf{Q}_i , $\boldsymbol{\beta}$ 的第 j 行对应 $\boldsymbol{\theta}_j$ 。若基于前 d 天的 $\boldsymbol{\beta}$ 预测出第 $d+1$ 天的组合系数列向量 $\boldsymbol{\beta}_{d+1}$, 则可以根据式(3)计算第 $d+1$ 天的交通流预测向量为:

$$\mathbf{Q}_{d+1} = \boldsymbol{\theta} \boldsymbol{\beta}_{d+1} = \sum_{j=1}^s \beta_{j,d+1} \boldsymbol{\theta}_j \quad (5)$$

本文采用 ESN 模型和 Jaya 优化算法相结合的方式预测 $\boldsymbol{\beta}_{d+1}$ 。

1.2 ESN 模型

如图 1 所示, ESN 为简单的 3 层结构, 包括输入层、隐含层(又称为动态储备池)和输出层。隐含层包含大量稀疏连接的神经元, 且具有较好的短期记忆功能, 是 ESN 的核心组成部分。作为一种新型递归神经网络, ESN 具有训练速度快、预测精度高的优点。

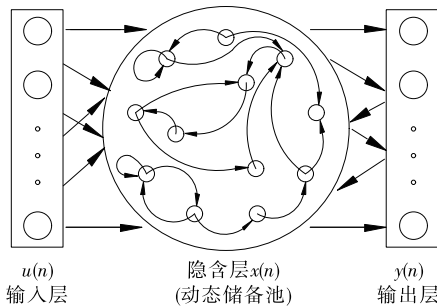


图 1 ESN 结构示意图

在采样时刻 n , ESN 的输入值、隐含层状态值和输出值为:

$$\begin{cases} \mathbf{u}(n) = [u_1(n), u_2(n), \dots, u_K(n)]^T \\ \mathbf{x}(n) = [x_1(n), x_2(n), \dots, x_N(n)]^T \\ \mathbf{y}(n) = [y_1(n), y_2(n), \dots, y_L(n)]^T \end{cases} \quad (6)$$

式中: K, N, L 分别为输入元、隐含层神经元和输出

元的数量。

状态值和输出值的计算公式分别为:

$$\mathbf{x}(n) = f_1[\mathbf{W}^{\text{in}} \mathbf{u}(n) + \mathbf{W} \mathbf{x}(n-1) + \mathbf{W}^b \mathbf{y}(n-1)] \quad (7)$$

$$\mathbf{y}(n) = \mathbf{W}^o[\mathbf{u}(n), \mathbf{x}(n), \mathbf{y}(n-1)] = \mathbf{W}^o \mathbf{X} \quad (8)$$

式中: f_1 采用 tanh 函数作为内部激活函数; $\mathbf{W}^{\text{in}} (N \times K)$ 、 $\mathbf{W} (N \times N)$ 、 $\mathbf{W}^b (N \times L)$ 、 $\mathbf{W}^o (L \times N)$ 分别代表输入、隐含层状态、反馈和输出权值矩阵。

为简化计算, \mathbf{W}^b 内各元素一般取为零, \mathbf{W}^{in} 和 \mathbf{W} 在初始化后保持不变, 这样, \mathbf{W}^o 成为 ESN 中唯一需要训练的权值矩阵。 \mathbf{W}^o 的计算公式为:

$$\mathbf{W}^o = \mathbf{Y} \mathbf{X}^T (\mathbf{X} \mathbf{X}^T)^{-1} \quad (9)$$

式中: $\mathbf{Y} (L \times M)$ 为输出值序列矩阵; $\mathbf{X} (N \times M)$ 为由输入值、隐含层状态值和输出值组成的序列矩阵; M 为序列矩阵 \mathbf{X} 的列数。

1.3 Jaya 优化算法

Jaya 优化算法由 Venkata Rao R. 于 2016 年首次提出^[10]。它是一种计算效率很高的元启发式优化算法, 需要调节的参数很少, 具有编程方便、计算快速等优点。该算法可以引导每个候选解朝着最佳解不断靠近, 同时逐渐远离最差解, 从而使更新后的候选解质量越来越高。Jaya 算法的候选解按下式更新:

$$X_{i,j,k+1} = X_{i,j,k} + r_1 (X_{i,\text{best},k} - |X_{i,j,k}|) - r_2 (X_{i,\text{worst},k} - |X_{i,j,k}|) \quad (10)$$

式中: $X_{i,j,k}$ 表示经过 k 次迭代后, 第 j 个候选解的第 i 维变量值; r_1, r_2 为 $0 \sim 1$ 的随机数; $X_{i,\text{best},k}$ 、 $X_{i,\text{worst},k}$ 分别表示迭代 k 次后的最佳解和最差解; $r_1 (X_{i,\text{best},k} - |X_{i,j,k}|)$ 表示候选解朝着最佳解不断靠拢的趋势程度; $r_2 (X_{i,\text{worst},k} - |X_{i,j,k}|)$ 表示候选解避开最差解的程度。

2 交通流预测流程

记第 j 个基向量组合系数时间序列为 $\boldsymbol{\beta}^j$ (即 $\boldsymbol{\beta}$ 的第 j 行), $\boldsymbol{\beta}^j = [\beta_1^j, \beta_2^j, \dots, \beta_d^j]$ ($j = 1, 2, \dots, s$), 基于 $\boldsymbol{\beta}^j$ 构建 ESN 预测模型的输入矩阵 \mathbf{H} :

$$\mathbf{H} = \begin{bmatrix} \beta_1^1 & \beta_2^1 & \cdots & \beta_{d-K}^1 & \beta_{d-K+1}^1 \\ \beta_1^2 & \beta_2^2 & \cdots & \beta_{d-K+1}^2 & \beta_{d-K+2}^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \beta_{K-1}^j & \beta_K^j & \cdots & \beta_{d-2}^j & \beta_{d-1}^j \\ \beta_K^j & \beta_{K+1}^j & \cdots & \beta_{d-1}^j & \beta_d^j \end{bmatrix} \quad (11)$$

式中: \mathbf{H} 中第 n 列对应式(6)中 n 时刻的输入值, 即

$$\mathbf{u}(n) = (\beta_n^j, \beta_{n+1}^j, \dots, \beta_{n+K-1}^j)^T.$$

考虑 $L=1$, 则输出值序列矩阵为:

$$\mathbf{Y} = [\beta_{K+1}^j, \beta_{K+2}^j, \dots, \beta_d^j] \quad (12)$$

根据上述分析, 针对 β^j 的结合 ESN 算法和 Jaya 算法的基向量组合系数预测流程为: 1) 对 β^j 进行归一化处理, 使各元素的值均位于 $-1.0 \sim 1.0$ 。2) 将 ESN 预测值与实际值的平均绝对误差作为适应度函数, 以输入元数量、储备池规模和储备池内部连接权谱半径作为优化变量。3) 确定组合系数时间序列训练样本长度、预测步长、输入数据和输出数据的维度、储备池输入单元尺度和稀疏度。4) 初始化种群。5) 针对不同种子训练 ESN 预测模型, 并计算适应度值, 不断更新最优种子。6) 如果适应度值小于指定阈值, 则停止优化, 得到最优权值矩阵并进入下一步; 否则, 按式(10)更新候选解, 并返回第5步继续该优化过程。7) 令 $n=d-K+1$, 设定输入值, 根据训练好的权值矩阵计算相应输出值, 该输出值即为所需组合系数预测值 β_{d+1}^j 。

按照上述流程依次针对各个 β^j ($j=1, 2, \dots, s$) 进行基向量组合系数预测, 得到相应的组合系数预测值 β_{d+1}^j , 再根据式(5)计算第 $d+1$ 天的交通流时间序列预测值。该方法利用基向量组合系数的单步预测实现交通流的多步预测, 基于前 d 天历史数据, 一次性给出第 $d+1$ 天所有 s 个时刻的交通流量预测值。

3 评价指标

采用均方根误差 E_{RMSE} 、平均绝对误差 E_{MAE} 和平均百分比误差 E_{MAPE} 评价上述方法的预测效果, 表达式如下:

$$E_{\text{RMSE}} = \sqrt{\frac{1}{n} \sum_{j=1}^n (q_j^{\text{pre}} - q_j^*)^2} \quad (13)$$

$$E_{\text{MAE}} = \frac{1}{n} \sum_{j=1}^n |q_j^* - q_j^{\text{pre}}| \quad (14)$$

$$E_{\text{MAPE}} = \frac{1}{n} \sum_{j=1}^n \left| \frac{q_j^* - q_j^{\text{pre}}}{q_j^*} \right| \quad (15)$$

式中: n 为预测数据的长度; q_j^{pre} 、 q_j^* 分别为第 j 个预测值和真实值。

E_{RMSE} 、 E_{MAE} 、 E_{MAPE} 越小, 预测效果越好。

4 试验分析

4.1 数据集

试验数据为某快速路在 2018 年 10 月 1—31 日

采集的工作日交通流数据。采样间隔为 15 min, 共 2 208 个采样点。以前 2 122 个采样点作为训练样本, 后 96 个采样点作为测试样本。原始交通流时间序列见图 2。由图 2 可知: 每个工作日的交通流具有一定的相似性和周期性, 但它们之间存在较大差别。这种差别是由交通流的随机性引起的。

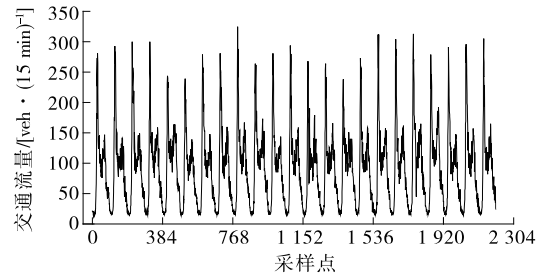


图2 原始交通流时间序列

4.2 预测结果

共有 96 个基向量, 每个基向量对应一个独立的 ESN 预测模型。Jaya 优化算法的种群规模取 20 个, 最大迭代次数取 50 次。

前 5 个基向量的寻优过程迭代曲线见图 3。这些迭代曲线在最初几步快速下降, 随后变得越来越平缓而趋近于收敛。总体而言, Jaya 优化算法的寻优速度较快。在第 50 次迭代结束后, 得到对应于各个基向量的最优输入元数量、储备池规模和储备池内部连接权谱半径, 据此得到各个基向量的组合系数预测值(见图 4)。由于基向量 $j=24 \sim 96$ 的组合系数预测值与真实值均小于 10^{-10} , 未在图 4 中绘出。由图 4 可知: 基向量 1 的组合系数远大于其他

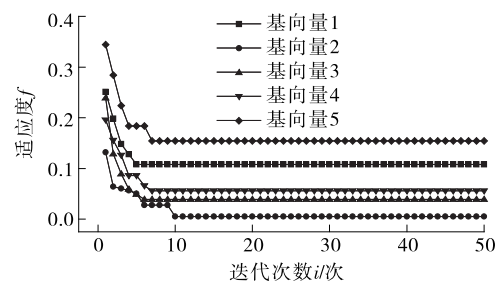


图3 Jaya 算法的优化迭代曲线

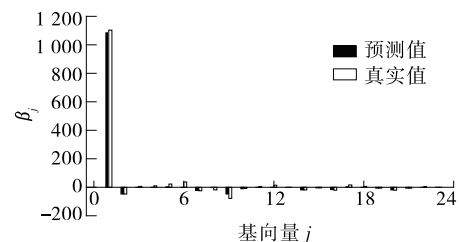


图4 各个基向量的组合系数预测值与真实值的对比

基向量。这是由于基向量 1 代表一天内交通流的总体变化趋势[见图 5(a)],其他具有较小组合系数的基向量(如基向量 2)实际上代表一天内交通流的某种局部变化或随机成分[见图 5(b)]。为了丰富交通流变化细节,在按式(5)计算交通流预测值时,这些具有较小组合系数的基向量应予以保留。

图 6 为历史均值法、ESN 法和 Jaya-ESN 法和本文方法的预测交通流量对比。历史均值法通过对若干历史前值取平均实现多步预测;ESN 法和 Jaya-ESN 法均为基于交通流时间序列的单步预测,区别在于前者未对 ESN 模型参数进行优化,后者则进行了优化。由图 6 可知:4 种方法的预测结果均能大致反映短时交通流的变化趋势。但在第 30 个采样点后,历史均值法的预测结果过于平缓,ESN 法的预测结果波动过大,Jaya-ESN 法的预测结果与真实值的波动规律更接近但波峰大多错位较大,本

文方法的预测曲线最贴近真实曲线。图 7 所示预测结果的绝对误差也表明了这一点。

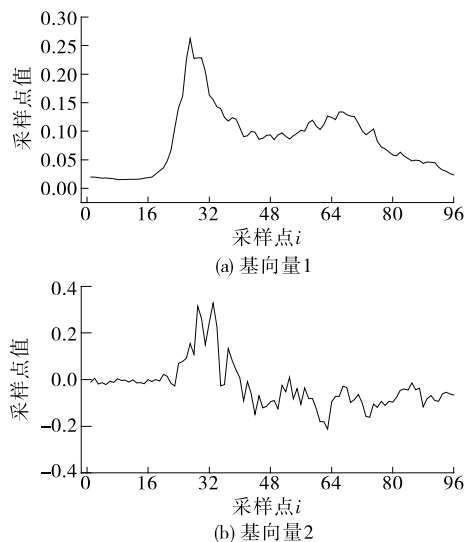


图 5 基向量分布

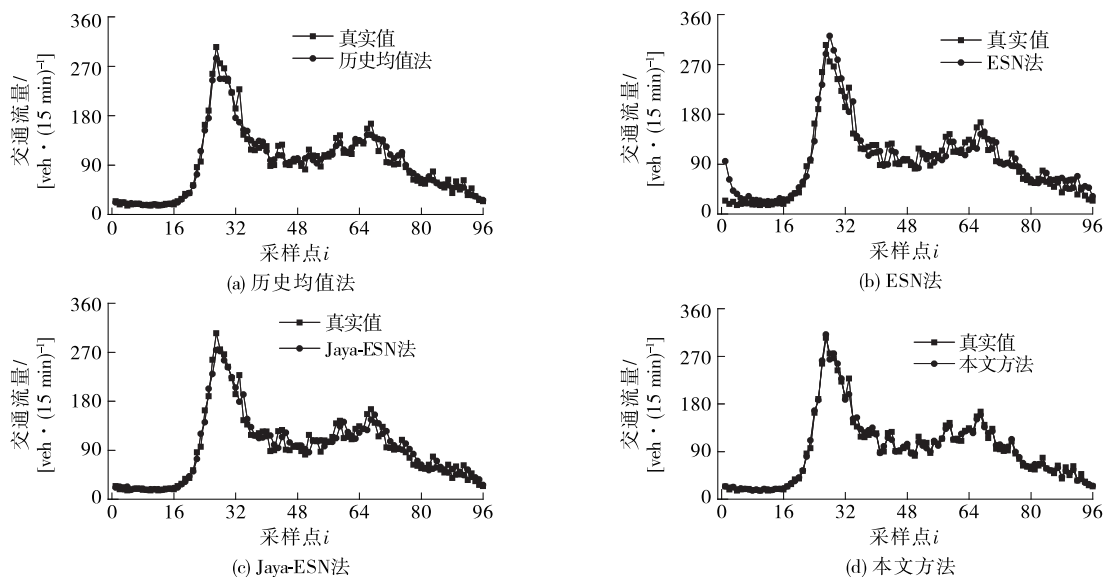


图 6 4 种方法的交通流量预测结果对比

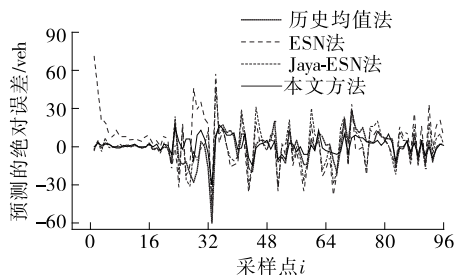


图 7 4 种方法的交通流量预测结果的绝对误差对比

4 种方法的交通流量预测结果的评价指标见表 1。由表 1 可知:本文方法的评价指标最佳,其次为历史均值法,ESN 法最差。由图 6(b)、图 6(c)可

知 ESN 法和 Jaya-ESN 法的预测值均具有一定的延迟效应(即波峰错位),这是导致其评价指标较差的根本原因。本文方法在 Jaya-ESN 法的基础上采

表 1 4 种方法的交通流量预测结果的评价指标

预测方法	E_{RMSE}	E_{MAE}	E_{MAPE}
历史均值法	14.601 7	10.822 9	0.096 2
ESN 法	20.017 4	15.177 1	0.270 2
Jaya-ESN 法	16.413 3	11.770 8	0.141 0
本文方法	6.649 2	4.768 8	0.055 0

(下转第 52 页)