



(12)发明专利申请

(10)申请公布号 CN 106295804 A

(43)申请公布日 2017. 01. 04

(21)申请号 201610671080.2

(22)申请日 2016.08.15

(71)申请人 清华大学

地址 100084 北京市海淀区北京市100084-82信箱

(72)发明人 胡坚明 裴欣 王星超 王振玉 张毅

(74)专利代理机构 北京众合诚成知识产权代理有限公司 11246

代理人 张文宝

(51)Int.Cl.

G06N 3/08(2006.01)

G06Q 10/04(2012.01)

权利要求书2页 说明书5页 附图2页

(54)发明名称

一种基于极端学习机网络的交通流预测方法

(57)摘要

本发明公开了一种基于极端学习机网络的交通流预测方法,所述极端学习机网络是前馈神经网络,所述极端学习机网络根据神经网络节点包括:输入层,隐含层,输出层;隐含层,所述隐含层是对输入层进行处理的层,通过所述输入层节点加权获得所述每一个隐含层节点;输出层,所述输出层为神经网络预测层,所述输出层用于验证神经网络的预测准确度。



1. 一种基于极端学习机网络的交通流预测方法,其特征在于,所述极端学习机网络是前馈神经网络,所述极端学习机网络根据神经网络节点包括:输入层,隐含层,输出层;

隐含层,所述隐含层是对输入层进行处理的层,通过所述输入层节点加权获得所述每一个隐含层节点;

输出层,所述输出层为神经网络预测层,所述输出层用于验证神经网络的预测准确度。

2. 如权利要求1所述的基于极端学习机网络的交通流预测方法,其特征在于:所述隐含层向所述输出层过渡的过程中需要激励函数 $g(x)$,所述激励函数 $g(x)$ 包括:

(1) Sigmoid函数

$$g(x) = \frac{1}{1 + \exp(-x)} \quad (4-1)$$

(2) 三角函数

$$g(x) = \sin(x) \quad (4-2)$$

(3) 阶跃函数

$$g(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (4-3)$$

(4) 高斯函数

$$g(x) = \exp(x^2) \quad (4-4)$$

(5) 曲面函数

$$g(x) = (x^2 + b^2)^{1/2} \quad (4-5)$$

(6) 小波函数

$$g(x) = a^{-1/2} \Psi\left(\frac{x-a}{b}\right) \quad (4-6)$$

3. 如权利要求2所述的基于极端学习机网络的交通流预测方法,其特征在于:对于任意 N 个随机样本 (x_i, t_i) , $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$, n 为所述输入层节点个数, $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$, 所述隐含层节点数为 m 个,所述隐含层激励函数为 $g(x)$,则数学模型如下所示:

$$\sum_{i=1}^m \beta_i g_i(x_j) = \sum_{i=1}^m \beta_i g(\omega_i \times x_j + b_i) = t_j \quad (4-7)$$

式中, $j=1, 2, \dots, N$, $\omega_i = [\omega_{i1}, \omega_{i2}, \dots, \omega_{in}]^T$ 为第 i 个隐含层节点与所述输入层节点之间的权值,第 i 个隐节点阈值为 b_i ,所述输出层节点与所述第 i 个隐含层节点的连接权值为 $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$,上述 N 个方程可以简写为

$$H\beta = T \quad (4-8)$$

其中

$$H = \begin{bmatrix} g(\omega_1 x_1 + b_1) & \dots & g(\omega_m x_1 + b_m) \\ \dots & \dots & \dots \\ g(\omega_1 x_N + b_1) & \dots & g(\omega_m x_N + b_m) \end{bmatrix}_{N \times m} \quad (4-9)$$

$$\beta = [\beta_1, \beta_2, \dots, \beta_k]_{k \times m}^T$$

$$T = [t_1, t_2, \dots, t_N]_{k \times N}^T$$

H 被称作隐含层输出矩阵, β 是输出层权值矩阵, T 是期望输出, k 是输出层节点个数;

根据上述方程的最小范数二乘解的求解方法,可以得到:

$$\beta = H^+T \quad (4-10)$$

其中, H^+ 是H的Moore-Penrose(MP)广义逆矩阵;

Moore-Penrose(MP)广义逆矩阵有如下性质:

$$H H^+ H = H$$

$$H^+ H H^+ = H^+ \quad (4-11)$$

$H H^+$ 和 $H^+ H$ 均为对称矩阵

若H为行满秩或者列满秩,则可以直接用下面的公式来求解Moore-Penrose(MP)广义逆矩阵 H^+ :

$$H^+ = H' (H H')^{-1} \text{或}$$

$$H^+ = (H H')^{-1} H' \quad (4-12)$$

其中, H' 是H的转置矩阵。

一种基于极端学习机网络的交通流预测方法

技术领域

[0001] 本发明涉及道路交通监测的技术领域,特别是涉及基于极端学习机网络的交通流预测方法。

背景技术

[0002] 交通流预测是一项复杂的工作,它是软硬件结合的。目前,大部分探测器装在道路两侧。

[0003] 感应线圈是现今比较常用的交通流探测器,当有车辆经过时,感应线圈就会记录经过车辆的车牌号,速度,经过时间等详细信息,每个线圈一天中会记录所有经过车辆的详细信息。这些信息会被加以处理,一般是对每个时间段经过感应线圈的车辆进行计数,处理过的数据会变成一个数组。而在某个区域内的车辆数量一般满足以下公式:

$$[0004] \quad N_t = N_{t-1} + n_t^{\text{in}} - n_t^{\text{out}} \quad (2-1)$$

[0005] 其中, N_t 表示在第 t 个时间点该路段的车辆总数, N_{t-1} 表示在 $t-1$ 个时间点该路段的车辆总数, n_t^{in} 表示在 t 个时间段内进入该路段的车辆总数, n_t^{out} 表示在 t 个时间段离开该路段的车辆总数。

[0006] 均匀采样是目前主要的交通流预测方式,而非均匀采样用于以下应用场景:

[0007] (1)数据丢失。有些整理好的数据本来是按照均匀时间段来采样的,但是可能在探测,处理,存储的过程中出现了数据丢失,此时的数据就无法用上述理论进行预测。然而以上数据还是很有价值的,不能因为丢失了一些数据就将它全部丢弃。

[0008] (2)数据原始的组织方式并非采用均匀采样。真正的原始数据,是线圈记录的数据,是摄像头记录的数据,是记录每辆车到达这个地点的时间,速度等数据。这些数据可以有新的组织方式,甚至可以用原始数据来对交通流进行预测。

[0009] 目前的BP神经网络算法需要学习迭代的过程,速度非常慢,并且精度也不是很理想。而且目前的预测算法只针对均匀采样的交通流数据,而对非均匀采样数据和有丢失采样数据无法处理。

[0010] 前馈神经网络是一种神经网络,且前馈神经网络是现今应用十分广泛的一种交通预测模型。前馈神经网络的原理是基于模式识别的,需要建立起历史数值和未来数值之间的函数关系。

$$[0011] \quad y_{\text{FFNN}}(t) = \sum_{j=1}^N c_j \psi \left(\sum_{i=1}^k w_{i,j} y(t-i) + b_j \right) + b_0 \quad (2-2)$$

[0012] 其中, N 表示隐层神经元数目, ψ 为基函数, $w_{i,j}$ 是第 i 个输入神经元和第 j 个隐层神经元之间的系数, c_j 是第 j 个隐层神经元与输出神经元间的系数, b_i 是常数偏置。

[0013] 因此希望有一种基于极端学习机网络的交通流预测方法可以克服或至少减轻现有技术上述缺陷。

发明内容

[0014] 本发明的目的在于提供一种基于极端学习机网络的交通流预测方法来克服现有技术中存在的上述问题。

[0015] 为实现上述目的,本发明提供一种基于极端学习机网络的交通流预测方法包括:

[0016] 所述极端学习机网络是前馈神经网络,所述极端学习机网络根据神经网络节点包括:输入层,隐含层,输出层;

[0017] 隐含层,所述隐含层是对输入层进行处理的层,通过所述输入层节点加权获得所述每一个隐含层节点;

[0018] 输出层,所述输出层为神经网络预测层,所述输出层用于验证神经网络的预测准确度。

[0019] 优选地,所述隐含层向所述输出层过渡的过程中需要激励函数 $g(x)$,所述激励函数 $g(x)$ 包括:

[0020] (1)Sigmoid函数

$$[0021] \quad g(x) = \frac{1}{1+\exp(-x)} \quad (4-1)$$

[0022] (2)三角函数

$$[0023] \quad g(x) = \sin(x) \quad (4-2)$$

[0024] (3)阶跃函数

$$[0025] \quad g(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (4-3)$$

[0026] (4)高斯函数

$$[0027] \quad g(x) = \exp(-x^2) \quad (4-4)$$

[0028] (5)曲面函数

$$[0029] \quad g(x) = (x^2+b^2)^{1/2} \quad (4-5)$$

[0030] (6)小波函数

$$[0031] \quad g(x) = a^{-\frac{1}{2}} \Psi\left(\frac{x-a}{b}\right) \quad (4-6)。$$

[0032] 优选地,对于任意 N 个随机样本 (x_i, t_i) , $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$, n 为所述输入层节点个数, $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$, 所述隐含层节点数为 m 个,所述隐含层激励函数为 $g(x)$,则数学模型如下所示:

$$[0033] \quad \sum_{i=1}^m \beta_i g_i(x_j) = \sum_{i=1}^m \beta_i g(\omega_i \times x_j + b_i) = t_j \quad (4-7)$$

[0034] 式中, $j=1, 2, \dots, N$, $\omega_i = [\omega_{i1}, \omega_{i2}, \dots, \omega_{in}]^T$ 为第 i 个隐含层节点与所述输入层节点之间的权值,第 i 个隐节点阈值为 b_i ,所述输出层节点与所述第 i 个隐含层节点的连接权值为 $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$,上述 N 个方程可以简写为

$$[0035] \quad H\beta = T \quad (4-8)$$

[0036] 其中

$$[0037] \quad H = \begin{bmatrix} g(\omega_1 x_1 + b_1) & \dots & g(\omega_m x_1 + b_m) \\ \dots & \dots & \dots \\ g(\omega_1 x_N + b_1) & \dots & g(\omega_m x_N + b_m) \end{bmatrix}_{N \times m} \quad (4-9)$$

$$[0038] \quad \beta = [\beta_1, \beta_2, \dots, \beta_k]_{k \times m}^T$$

$$[0039] \quad T = [t_1, t_2, \dots, t_N]_{k \times N}^T$$

[0040] H被称作隐含层输出矩阵,β是输出层权值矩阵,T是期望输出,k是输出层节点个数。

[0041] 根据上述方程的最小范数二乘解的求解方法,可以得到:

$$[0042] \quad \hat{\beta} = H^+ T \quad (4-10)$$

[0043] 其中, H^+ 是H的Moore-Penrose(MP)广义逆矩阵。

[0044] Moore-Penrose(MP)广义逆矩阵有如下性质:

$$[0045] \quad H H^+ H = H$$

$$[0046] \quad H^+ H H^+ = H^+ \quad (4-11)$$

[0047] $H H^+$ 和 $H^+ H$ 均为对称矩阵

[0048] 若H为行满秩或者列满秩,则可以直接用下面的公式来求解Moore-Penrose(MP)广义逆矩阵 H^+ :

$$[0049] \quad H^+ = H' (H H')^{-1} \text{或}$$

$$[0050] \quad H^+ = (H H')^{-1} H' \quad (4-12)$$

[0051] 其中, H' 是H的转置矩阵。

[0052] 本发明提出一种基于极端学习机网络的交通流预测方法,通过本发明的极端学习机算法进行交通流的预测,实现对非均匀采样和有丢失采样数据的预测。

附图说明

[0053] 图1是本发明基于极端学习机网络的交通流预测方法的均匀数据预测结果。

[0054] 图2是本发明基于极端学习机网络的交通流预测方法的非均匀数据预测结果。

[0055] 图3是本发明基于极端学习机网络的交通流预测方法的有丢失数据预测结果。

[0056] 图4是本发明基于极端学习机网络的交通流预测方法的均匀和非均匀数据预测流程图。

[0057] 图5是本发明基于极端学习机网络的交通流预测方法的有丢失数据预测流程图。

具体实施方式

[0058] 为使本发明实施的目的、技术方案和优点更加清楚,下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行更加详细的描述。在附图中,自始至终相同或类似的标号表示相同或类似的元件或具有相同或类似功能的元件。所描述的实施例是本发明一部分实施例,而不是全部的实施例。下面通过参考附图描述的实施例是示例性的,旨在用于解释本发明,而不能理解为对本发明的限制。基于本发明中的实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。下面结合附图对本发明的实施例进行详细说明。

[0059] 在本发明一宽泛实施例中:基于极端学习机网络的交通流预测方法包括:

[0060] 所述极端学习机网络是前馈神经网络,所述极端学习机网络根据神经网络节点包括:输入层,隐含层,输出层;

[0061] 隐含层,所述隐含层是对输入层进行处理的层,通过所述输入层节点加权获得所

述每一个隐含层节点；

[0062] 输出层,所述输出层为神经网络预测层,所述输出层用于验证神经网络的预测准确度。

[0063] 所述隐含层向所述输出层过渡的过程中需要激励函数 $g(x)$,所述激励函数 $g(x)$ 包括:

[0064] (1)Sigmoid函数

$$[0065] \quad g(x) = \frac{1}{1+\exp(-x)} \quad (4-1)$$

[0066] (2)三角函数

$$[0067] \quad g(x) = \sin(x) \quad (4-2)$$

[0068] (3)阶跃函数

$$[0069] \quad g(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (4-3)$$

[0070] (4)高斯函数

$$[0071] \quad g(x) = \exp(-x^2) \quad (4-4)$$

[0072] (5)曲面函数

$$[0073] \quad g(x) = (x^2+b^2)^{1/2} \quad (4-5)$$

[0074] (6)小波函数

$$[0075] \quad g(x) = a^{-\frac{1}{2}} \Psi\left(\frac{x-a}{b}\right) \quad (4-6)。$$

[0076] 对于任意 N 个随机样本 (x_i, t_i) , $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$, n 为所述输入层节点个数, $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$, 所述隐含层节点数为 m 个, 所述隐含层激励函数为 $g(x)$, 则数学模型如下所示:

$$[0077] \quad \sum_{i=1}^m \beta_i g_i(x_j) = \sum_{i=1}^m \beta_i g(\omega_i \times x_j + b_i) = t_j \quad (4-7)$$

[0078] 式中, $j=1, 2, \dots, N$, $\omega_i = [\omega_{i1}, \omega_{i2}, \dots, \omega_{in}]^T$ 为第 i 个隐含层节点与所述输入层节点之间的权值, 第 i 个隐节点阈值为 b_i , 所述输出层节点与所述第 i 个隐含层节点的连接权值为 $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$, 上述 N 个方程可以简写为

$$[0079] \quad H\beta = T \quad (4-8)$$

[0080] 其中

$$[0081] \quad H = \begin{bmatrix} g(\omega_1 x_1 + b_1) & \dots & g(\omega_m x_1 + b_m) \\ \dots & \dots & \dots \\ g(\omega_1 x_N + b_1) & \dots & g(\omega_m x_N + b_m) \end{bmatrix}_{N \times m} \quad (4-9)$$

$$[0082] \quad \beta = [\beta_1, \beta_2, \dots, \beta_k]_{k \times m}^T$$

$$[0083] \quad T = [t_1, t_2, \dots, t_N]_{k \times N}^T$$

[0084] H 被称作隐含层输出矩阵, β 是输出层权值矩阵, T 是期望输出, k 是输出层节点个数。

[0085] 根据上述方程的最小范数二乘解的求解方法, 可以得到:

$$[0086] \quad \hat{\beta} = H^+ T \quad (4-10)$$

[0087] 其中, H^+ 是 H 的 Moore-Penrose (MP) 广义逆矩阵。

[0088] Moore-Penrose (MP) 广义逆矩阵有如下性质:

[0089] $H H^+ H = H$

[0090] $H^+ H H^+ = H^+$ (4-11)

[0091] $H H^+$ 和 $H^+ H$ 均为对称矩阵

[0092] 若 H 为行满秩或者列满秩, 则可以直接用下面的公式来求解 Moore-Penrose (MP) 广义逆矩阵 H^+ :

[0093] $H^+ = H' (H H')^{-1}$ 或

[0094] $H^+ = (H' H)^{-1} H'$ (4-12)

[0095] 其中, H' 是 H 的转置矩阵

[0096] 如图1所示, 均匀采样数据: 合并数为2, 之前的原始数据的采样间隔为5分钟, 合并为最终数据时的采样间隔为10分钟, 最终预测误差: 8.75%。

[0097] 如图2所示, 非均匀采样数据: 在均匀采样数据的基础上(采样间隔为10分钟), 最大合并数为2, 最终的预测误差为: 12.56%。

[0098] 如图3所示, 有丢失采样数据: 在均匀采样数据的基础上(采样间隔为10分钟), 丢失率为10%, 最终的预测误差为: 11.03%。

[0099] 如图4所示, 对于均匀和非均匀采样数据的预测流程包括: 1. 确定隐含层节点数; 2. 对每次预测寻找训练集; 3. 对训练集整体进行学习; 4. 得到预测参数; 5. 带入测试输入进行预测; 6. 记录时间和误差等数据。

[0100] 如图5所示, 对于有丢失采样数据的预测图包括: 1. 将15天的数据分开; 2. 判断每天的数据是否丢失; 3. 如果数据丢失则舍弃, 如果数据未丢失则保留; 4. 将未丢失数据组成矩阵; 5. 对矩阵整体进行学习。

[0101] 最后需要指出的是: 以上实施例仅用以说明本发明的技术方案, 而非对其限制。尽管参照前述实施例对本发明进行了详细的说明, 本领域的普通技术人员应当理解: 其依然可以对前述各实施例所记载的技术方案进行修改, 或者对其中部分技术特征进行等同替换; 而这些修改或者替换, 并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

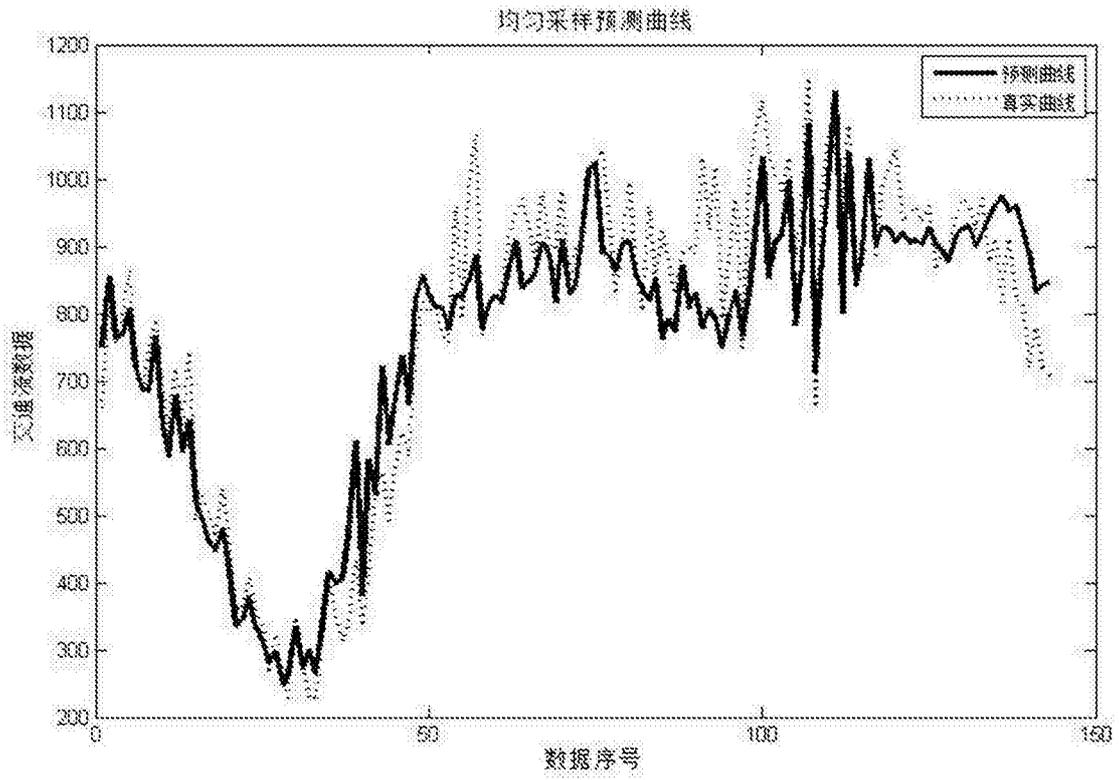


图1

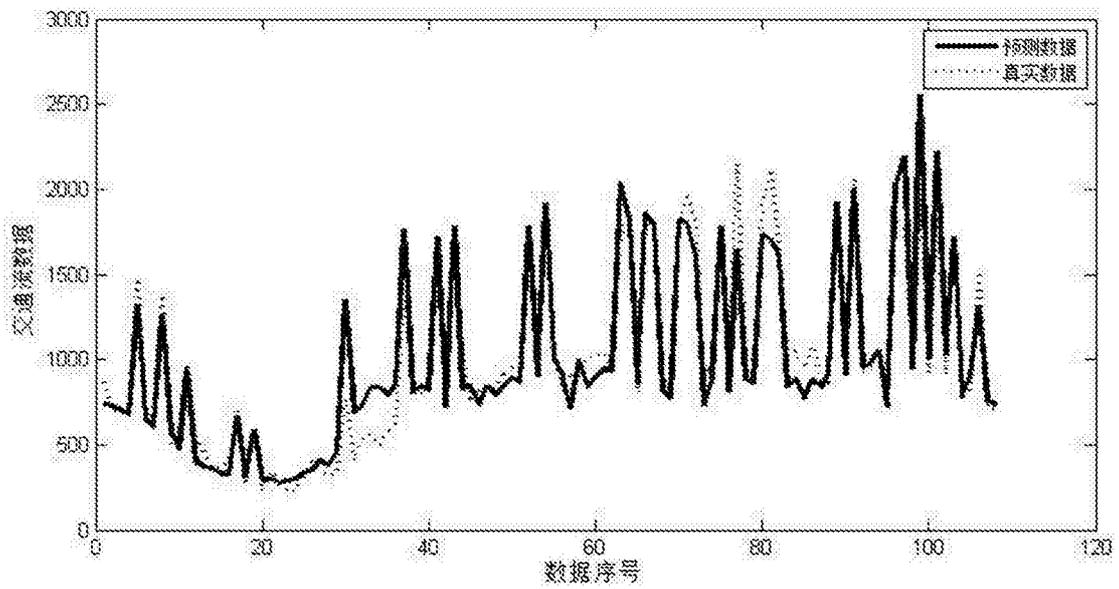


图2

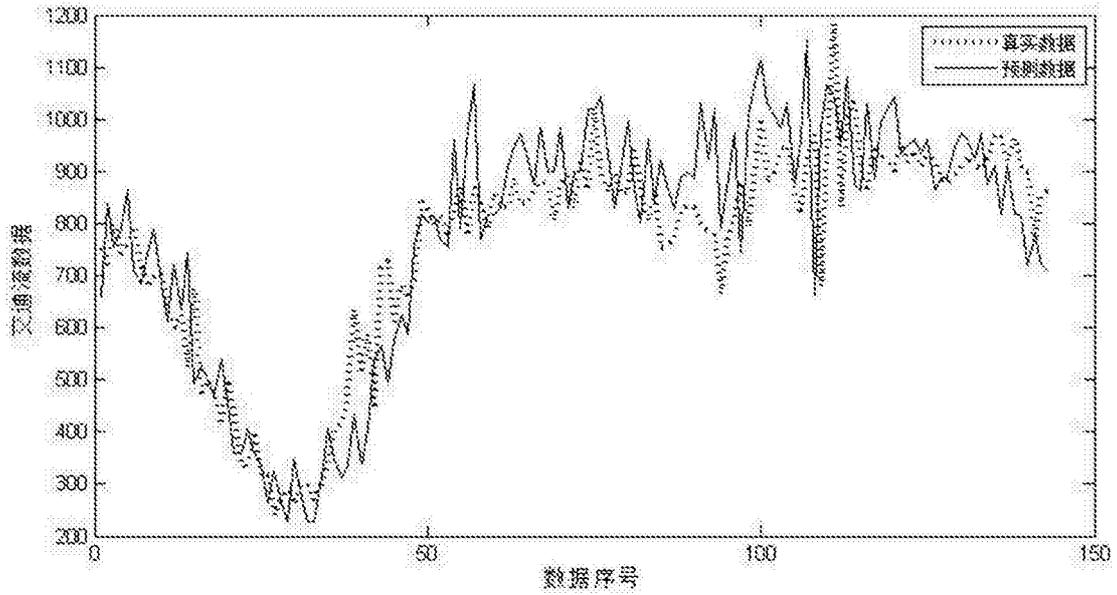


图3

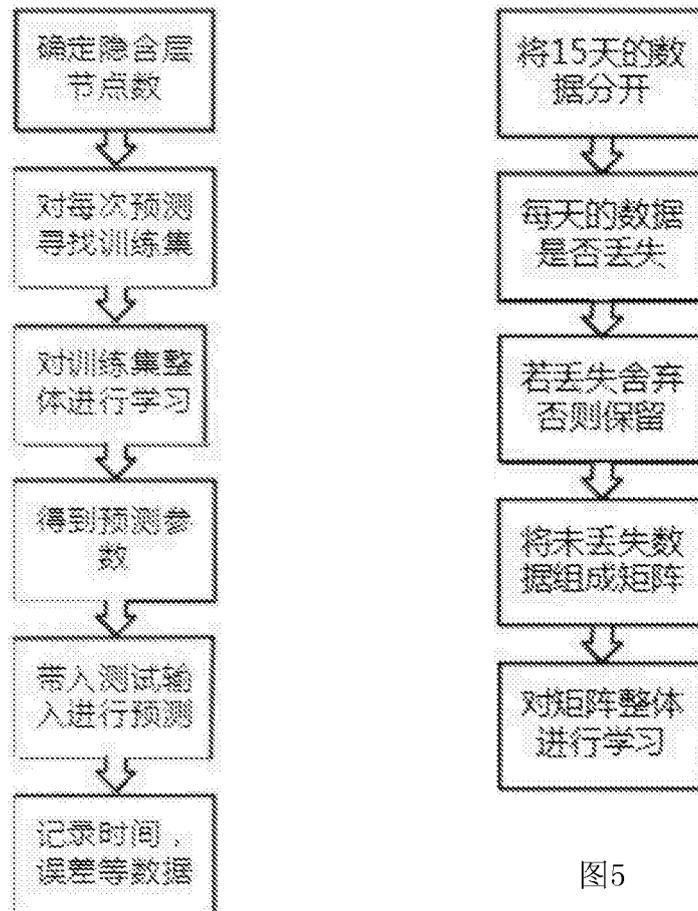


图5

图4