文章编号: 1001-0920 (2015) 03-0507-06 DOI: 10.13195/j.kzyjc.2013.1298

结构自适应的半监督自组织过程神经网络

王 兵[1], 许少华[1], 孟耀华[2], 王 辉[1], 李 娜[3]

(1. 东北石油大学 计算机与信息技术学院，黑龙江 大庆 163318；2. 哈尔滨工业大学 通信技术研究所，哈尔滨 150001；3. 大庆油田化工集团 东昊公司，黑龙江 大庆 163453)

摘 要: 针对时域空间中模式识别、聚类分析和未标记样本的有效利用问题, 提出一种基于半监督学习的网络结构自适应的二维自组织过程神经网络模型和算法. 通过构建可度量时变样本间相似性的广义 Fréchet 距离, 利用部分已标记动态样本的类别信息和过程特征, 采用奖励-惩罚更新规则, 根据网络学习目标函数, 对网络二维平面竞争层节点进行动态拆分或合并, 实现网络结构的自适应调整和样本的有效聚类. 仿真实验结果验证了模型和算法的有效性.

关键词: 自组织过程神经网络；半监督学习；结构自适应；模式识别

中图分类号: TP183 **文献标志码:** A

Semi-supervised self-organizing process neural network with self-adaptive structure

WANG Bing[1], *XU Shao-hua*[1], *MENG Yao-hua*[2], *WANG Hui*[1], *LI Na*[3]

(1. School of Computer and Information Technology，Northeast Petroleum University，Daqing 163318，China; 2. Communication Research Center，Harbin Institute of Technology，Harbin 150001，China；3. Donghao Company, Daqing Oilfield Chemical Company Limited，Daqing 163453，China. Correspondent：WANG Bing，E-mail: wangbing0812@sina.com)

Abstract: Aimed at the problems such as pattern recognition, cluster analysis, effective use of unlabeled samples, etc. in the time-varying space, a two-dimensional self-organizing process neural network with self-adaptive structure based on semi-supervised learning is proposed. By building the generalized Fréchet distance which is used to measure the similarity among time-varying function samples, using the class information and process features of partial labeled dynamic samples, adopting reward-punishment update rule, and according to the network learning objective function, the dynamic reconstruction of the network structure is realized with splitting and mergering competitive nodes in two-dimensional plane layer, and then effective clustering is implemented. Experimental results verify the effectiveness of the proposed model and algorithm.

Keywords: self-organizing process neural network；semi-supervised learning；self-adaptive structure；pattern recognition

0 引 言

自组织特征映射 (SOFM) 神经网络[1-2]是一种无教师自组织自学习网络, 具有抽取输入信号模式特征的能力, 在模式识别及模式完善等方面取得了较好的应用效果[3-5]. 然而, 在实际工程应用中, 许多系统的输入是依赖于时间变化的过程信号, 现有的 SOFM 模型的输入一般是与时间无关的常量, 在信息处理机制上无法直接反映时变输入信号的过程特征与变量之间的动态作用关系. 为此, 文献 [6] 建立了一种自组织过程神经网络模型, 该模型直接以时变过程信号为网络输入, 根据输入函数所隐含的过程式模式特征对其进行自组织, 并在竞争层将分类结果表现出来, 已在动态样本聚类和时变信号辨识中得到有效应用.

在实际工程中, 一些非线性动态系统往往难以获得足够多的有标记的动态过程采样数据, 但可以获得大量具有相同指标特征但无标记的时变样本数据. 因此, 近年来, 面向标记和未标记样本集合信息有效利用的半监督学习算法受到了广泛关注, 成为机器学习领域中一个新的前沿方向. 其中: 文献 [7] 利用负标记进行半监督学习, 文献 [8] 建立了基于局部线性回归的半监督数据聚类算法, 文献 [9] 开发了一种半监督多元回归树用于天气分类. 同时, 半监督学习算法与

收稿日期: 2013-09-18；**修回日期:** 2014-01-29.

基金项目: 黑龙江省教育厅基金项目(12511009)；黑龙江省教育厅科学技术研究项目(12521369).

作者简介: 王兵(1982−), 女, 讲师, 博士生, 从事神经网络、模式识别的研究；许少华(1962−), 男, 教授, 博士生导师, 从事智能信息处理、神经网络、过程控制等研究.

神经网络[10]、支持向量机[11-12]、多 Agent[13]等机器学习方法进行结合也取得了较好的应用效果.

本文将半监督学习方法向时域空间推广, 结合自组织过程神经网络的信息处理机制和应用, 提出一种基于半监督学习的二维自组织过程神经网络 (STS-PNN) 模型和算法. 通过构建一种广义 Fréchet 测度泛数来度量时变函数空间中动态样本之间的距离, 基于该距离建立动态样本聚类算法, 实现利用标记及未标记动态样本的全部信息来进行网络模型的构建和训练. 同时, 针对无法事先预知样本类别数目的问题, 采用一种可动态重构二维输出平面竞争节点的方法进行网络训练, 有效提高了 STS-PNN 的自适应和泛化辨识能力. 实验结果验证了模型和算法的有效性.

1 二维自组织过程神经网络模型

二维自组织过程神经网络由两层构成, 第 1 层为动态模式输入层, 负责将输入函数向量的每一个分量函数按照一定的连接权函数传递到竞争层; 第 2 层为由自组织过程神经元组成的二维平面竞争输出层, 负责对输入层传递来的动态模式进行 "分析比较", 捕捉住各个输入函数所包含的过程模式特征, 并将其进行自组织, 在输出层形成能够反映样本模式类分布情况的有序特征图. 二维自组织过程神经网络的拓扑结构如图 1 所示.

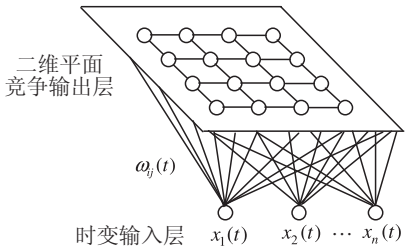


图 1 二维自组织过程神经网络模型

设网络输入空间为 $(C[0,T])^n$, 其中 $[0,T]$ 为时间过程采样区间. 输入层有 n 个过程式输入节点, 其输入函数向量为 $\boldsymbol{X}(t)=(x_1(t),x_2(t),\cdots,x_n(t))$, 二维平面竞争层共有 $m\times m$ 个自组织竞争输出节点, $\omega_{ij}(t)$ $(i=1,2,\cdots,n,\ j=1,2,\cdots,m\times m)$ 为输入层节点 i 与竞争层节点 j 的连接权函数.

2 动态时变样本间距离的测度

本文建立的 STS-PNN 模型的输入为一动态的时变函数向量, 为此, 构建可度量时变函数样本之间距离的测度泛数, 使其能够有效测量不同样本之间的性质差异. 现有的方法[6]一般是利用基函数的正交性, 对输入函数和网络权函数实施正交基展开后再进行计算. 但该方法中选用何种基函数以及如何确定基函数的展开项数还没有一个统一的指导标准. 此外, 由于实际应用中动态样本通常是由采样得到的离散点序列, 必需先将其拟合成解析函数再进行基展开, 不仅需要较大的工作量且容易产生误差. 为避免基函数展开方式带来的拟合误差和截断误差, 将时变函数看作是关于时间的一维曲线, 引入离散 Fréchet 距离来直接度量时变函数之间的差异, 然后将其推广到时变函数向量样本之间的度量.

2.1 离散 Fréchet 距离

离散 Fréchet 距离[14-15]是一种判断多边形曲线之间相似性距离的测度, 其定义如下:

1) 给定一个有 n 个至高点的多边形链 $P=\langle p_1,p_2,\cdots,p_n\rangle$, 一个沿着 P 的 k 步分割 P 的至高点成为 k 个不相交的非空子集 $\{P_i\}_{i=1,2,\cdots,k}$, 使得 $P_i=\langle p_{n_{i-1}+1},\cdots,p_{n_i}\rangle$ 和 $0=n_0<n_1<\cdots<n_k=n$.

2) 给定两个多边形链 $A=\langle a_1,a_2,\cdots,a_m\rangle$, $B=\langle b_1,b_2,\cdots,b_n\rangle$, 一个沿着 A 和 B 的组合步 (由一个沿着 A 的 k 步 $\{A_i\}_{i=1,2,\cdots,k}$ 和一个沿着 B 的 k 步 $\{B_i\}_{i=1,2,\cdots,k}$ 组成), 使得对于 $1\leqslant i\leqslant k$, 要么 $|A_i|=1$, 要么 $|B_i|=1$ (就是说 A_i, B_i 中有一个恰好包含一个至高点).

3) 一个沿着链 A 和 B 的组合步 $W=\{(A_i,B_i)\}$ 的花费为

$$d_F^W(A,B)=\max_i \max_{(a,b)\in A_i\times B_i} d(a,b), \tag{1}$$

则链 A 和 B 间的离散 Fréchet 距离为

$$d_F(A,B)=\min_W d_F^W(A,B), \tag{2}$$

这个组合步称为链 A 和 B 的 Fréchet 排列.

2.2 时变函数向量间的广义 Fréchet 距离

离散 Fréchet 距离可有效度量时变函数 (可看作是多边形链) 之间的差异, 而 STS-PNN 的输入为由时变函数组成的向量. 为此, 利用欧式距离可进行点目标匹配的性质, 将离散 Fréchet 距离与欧式距离相结合, 建立一种可度量时变函数向量样本间距离的广义 Fréchet 距离.

已知时变函数样本 $\boldsymbol{X}(t)=(x_1(t),x_2(t),\cdots,x_n(t))$ 和第 j 个二维平面竞争层节点的连接权函数向量 $\boldsymbol{\omega}_j(t)=(\omega_{1j}(t),\omega_{2j}(t),\cdots,\omega_{nj}(t))$, $j=1,2,\cdots,m\times m$, 则它们之间的广义离散 Fréchet 距离 $d(\boldsymbol{X}(t),\boldsymbol{\omega}_j(t))$ 定义如下:

$$d(\boldsymbol{X}(t),\boldsymbol{\omega}_j(t))=\mathrm{sqrt}\Big[\sum_{i=1}^{n} d_F(x_i(t),\omega_{ij}(t))^2\Big], \tag{3}$$

其中 $d_F(x_i(t),\omega_{ij}(t))$ 表示 $x_i(t)$ 与 $\omega_{ij}(t)$ 对应的离散采样点之间的离散 Fréchet 距离.

3 网络结构自适应的半监督学习算法

本文采用一种由网络目标函数驱动的动态重构二维平面竞争层节点的方法, 利用奖励-惩罚更新规则, 建立一种基于半监督学习的网络结构自适应的过

程神经元网络竞争学习算法, 从而提高网络的辨识性能和鲁棒性.

3.1 奖励-惩罚更新规则

自组织过程神经网络的学习和识别涉及时变模式样本间距离的度量和竞争更新两部分. 在学习更新过程中, 获胜神经元及其周围邻域内的神经元将进行不同强度的调整, 规则如下:

$$\boldsymbol{\omega}_r^{k+1}(t)=\boldsymbol{\omega}_r^k(t)+\eta(k)\Lambda(k,d_{r,j})(\boldsymbol{X}(t)-\boldsymbol{\omega}_r^k(t)). \tag{4}$$

其中: k 为当前迭代步数; $\boldsymbol{\omega}_r^k(t)$ 为优胜邻域内神经元 r 的连接权函数向量; $\eta(k)$ 为学习速率; $d_{r,j}$ 为当前待调神经元 r 和获胜神经元 j 在二维网格中的距离; $\Lambda(k,d_{r,j})$ 为一邻域函数, 它是迭代步数 k 和 $d_{r,j}$ 的函数, 并随着 $d_{r,j}$ 的增大和迭代的进行不断地减小, 可取高斯函数等.

在此更新策略下, 如果两个竞争层节点初始化时都比较接近某一分类, 但由于在更新过程中仅奖励了获胜的节点靠近该类而未惩罚失败的节点远离该类, 则终将使本应代表同一类的两个节点代表了两个不同的分类. 因此, 除了对获胜神经元及其邻域内神经元实施奖励使其靠近输入模式向量以外, 还应对在竞争中失败的神经元节点采用惩罚策略, 使其逐步远离学习样本, 这样也有助于加速网络的竞争聚类过程. 具体惩罚规则如下:

$$\boldsymbol{\omega}_o^{k+1}(t)=\boldsymbol{\omega}_o^k(t)-r(k)(\boldsymbol{X}(t)-\boldsymbol{\omega}_o^k(t)). \tag{5}$$

其中: k 为当前迭代步数, $\boldsymbol{\omega}_o^k(t)$ 为竞争失败神经元的连接权向量, $\gamma(k)$ 为学习速率, 一般可取 $0<\gamma<\eta<1$.

设时域采样区间为 $[0,T]$, 采样时刻为 $0=t_1<t_2<\cdots<t_L=T$. 将 n 维时变函数样本 $\boldsymbol{X}(t)=(x_1(t),x_2(t),\cdots,x_n(t))$ 以及竞争层节点的连接权函数向量分别离散化为

$$\boldsymbol{X}(t_l)=(x_1(t_l),x_2(t_l),\cdots,x_n(t_l)),\ 1\leqslant l\leqslant L, \tag{6}$$

$$\boldsymbol{\omega}(t_l)=(\omega_1(t_l),\omega_2(t_l),\cdots,\omega_n(t_l)),\ 1\leqslant l\leqslant L, \tag{7}$$

则网络奖励-惩罚更新规则具体调整为

$$\begin{cases}\boldsymbol{\omega}_r^{k+1}(t_l)=\boldsymbol{\omega}_r^k(t_l)+\eta(k)\Lambda(k,d_{r,j})(\boldsymbol{X}(t_l)-\boldsymbol{\omega}_r^k(t_l)),\\ \qquad r\in\mathrm{Neiborhood}(j_{\mathrm{win}});\\ \boldsymbol{\omega}_o^{k+1}(t_l)=\boldsymbol{\omega}_o^k(t_l)-r(k)(\boldsymbol{X}(t_l)-\boldsymbol{\omega}_o^k(t_l)),\\ \qquad \mathrm{otherwise}.\end{cases} \tag{8}$$

3.2 网络学习目标函数

为了衡量网络的聚类辨识效果, 定义一个网络学习目标函数, 目的是最小化网络聚类的不纯度并使聚类数尽可能的少[16], 其定义如下:

$$E=\alpha\times\mathrm{Im}p+\beta\times\mathrm{Sct}, \tag{9}$$

其中 α 和 β 分别为不纯度 $\mathrm{Im}p$ 和分散度 Sct 的权系数, 并且 $\alpha+\beta=1$.

常见的不纯度度量有 3 种: 误分类不纯度, Gini 不纯度和熵不纯度. 为计算简便, 这里采用误分类率进行不纯度的计算, 将网络分类结果的不纯度定义为各个聚类结果不纯度的加权平均, 即

$$\mathrm{Im}p=\sum_{j=1}^{m\times m}|\boldsymbol{\omega}_j(t)|\times\mathrm{Mis}C(j)/S. \tag{10}$$

其中: $m\times m$ 为网络聚类数; S 为动态样本数; $|\boldsymbol{\omega}_j(t)|$ 为聚类 j 中包含的样本数; $\mathrm{Mis}C(j)$ 为聚类 j 的误分类率, 有

$$\mathrm{Mis}C(j)=\frac{|\boldsymbol{X}(t)\notin\mathrm{Dom}C(j),\boldsymbol{X}(t)\in j|}{|\boldsymbol{\omega}_j(t)|}, \tag{11}$$

$\mathrm{Dom}C(j)$ 为聚类 j 代表的优势类, 即在聚类 j 实际包含的多个分类中, 如果其中某一分类包含的样本数多于任何其他分类, 则该分类即为聚类 j 代表的优势类. 可以看出, 误分类率是聚类 j 中不属于优势类的样本所占的百分比.

分散度的度量采用如下形式:

$$\mathrm{Sct}=\sqrt{m\times m/S}. \tag{12}$$

网络聚类数越少, 分散度越小. 根据式 (9)～(12), 网络学习目标函数为

$$E=\alpha\times\Big(\sum_{j=1}^{m\times m}|\boldsymbol{X}(t)\notin\mathrm{Dom}C(j),\boldsymbol{X}(t)\in j|/S\Big)+\beta\times\sqrt{m\times m/S}. \tag{13}$$

3.3 网络结构自适应调整规则

一般来说, 自组织过程神经网络输出层节点的个数是事先给定的, 并且在学习过程中保持不变. 然而, 在实际应用中, 往往无法事先预知采样样本的分类情况. 因此, 需要一个变结构的网络学习规则来解决上述问题, 即网络输出层节点可以在某些规则的驱动下进行动态的拆分或合并, 以适应实际问题的需要.

由于二维自组织过程神经网络在初始学习阶段, 权向量处在无序状态, 如果此时对节点进行拆分或合并, 只能徒劳增加网络计算的工作量. 当经过若干次的迭代, 权向量分布趋于稳定, 分类模式也初步确定, 网络从无序走向有序后, 再对网络进行变结构拆分或合并, 可以在细调网络权值的同时, 对代表模式过多、利用过剩的节点进行拆分, 对欠利用的相似节点进行合并, 使网络结构趋于合理化. 这里采用网络学习目标函数的变化增量来衡量网络是否已从无序走向有序, 即

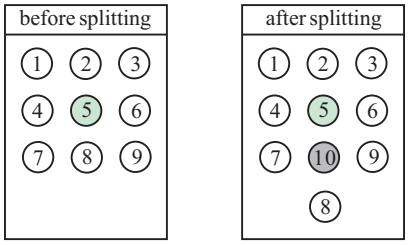
$$\Delta E=E(k+1)-E(k)<\zeta. \tag{14}$$

其中: k 为迭代次数; ΔE 为两次迭代前后网络学习目标函数的差值; ζ 为一给定的有序阈值常数, 当两次迭代目标函数差值 ΔE 小于此给定常数 ζ 时, 可以认为网络学习趋于稳定, 即排序阶段已经结束, 从而进行

节点拆分和合并.

3.3.1 拆分规则

误分类率 Mis$C(j)$ 表示聚类 j 中不属于优势类的样本所占的百分比. 可以看出, 误分类率越大, 该输出节点聚类效果越差, 因此可选择误分类率最大的节点作为预拆分的对象. 此时权向量已经经过排序阶段, 相似节点在二维输出平面上的位置也比较接近, 因此新拆分出的节点应该放置在被拆分节点的附近. 具体拆分规则如图 2 所示.



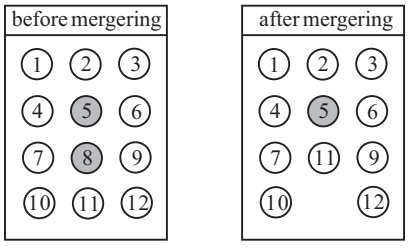
(a) 拆分前 (b) 拆分后

图 2 拆分前后节点的分布

首先, 选择具有最大误分类率的神经元节点作为预拆分对象, 例如 5 号节点; 然后在与 5 号节点相邻的 4 个节点中找出与其权值差距最大即最不相似的节点, 例如 8 号节点; 在 5 号节点和 8 号节点之间插入新拆分出的 10 号节点, 并且 8 号及其下面的所有节点均沿 5 号到 8 号连线方向向下移动一个网格. 拆分后 5 号节点和 10 号节点的权向量分别取拆分前 5 号节点优势类和次优势类成员样本的平均值. 是不是只要网络学习不停, 就要找出一个具有最大误分类率的节点进行拆分呢? 即使此时该节点的误分类率已经很小. 显然不是的, 这样就需要设置一个拆分的停止准则. 为此, 可以用网络学习目标函数作为度量标准, 先对网络进行预拆分, 如果拆分前后网络学习目标函数值减小, 则执行此拆分, 否则不进行拆分.

3.3.2 合并规则

如果两个输出层节点的权向量具有相同的优势类, 并且在所有权向量中它们之间的距离最小, 即最为相似, 则可将这两个权向量作为候选的合并对象. 是否最终要执行合并操作, 同上述拆分停止准则相同, 取决于合并后的网络学习目标函数值是否减小. 合并后新节点的权向量取合并前两个节点权向量的平均值. 分别计算新节点与合并前两节点周围 4 个相邻节点权向量的平均差值, 选择差值较小的节点位置作为合并后新节点的存放位置. 空出节点的位置由该节点后面的节点沿该节点到新节点连线方向自动补全, 如图 3 所示.



(a) 合并前 (b) 合并后

图 3 合并前后节点的分布

3.4 算法步骤描述

Step 1: 初始化. 初始化竞争层节点数 $m \times m$, 权函数向量 $\boldsymbol{\omega}_j(t)$, $j = 1, 2, \cdots, m \times m$, 学习速率 η 和 γ, 邻域函数 Λ, 不纯度以及分散度系数 α 和 β, 学习目标阈值 ξ, 有序阈值 ζ, 最大迭代次数 K. 置当前迭代次数 $k = 0$, 置有序标志 Flag = false.

Step 2: 标记类. 将样本集输入网络进行聚类, 标识各个聚类结果的优势类 Dom$C(j)$, $j = 1, 2, \cdots, m \times m$. 若某一聚类包含的样本均为无标记样本, 则该聚类的优势类标记为“未知类”.

Step 3: 学习更新. 对于每一个样本 $\boldsymbol{X}(t)$, 随机输入网络进行学习, 其更新过程分以下两种情况:

1) 若 $\boldsymbol{X}(t)$ 为已标记样本, 则与 $\boldsymbol{X}(t)$ 属于相同类的聚类以及未知类进行竞争学习, 根据式 (4), 获胜的神经元及其邻域内神经元对应的权向量按照奖励规则进行更新, 竞争失败的权向量接受惩罚, 与 $\boldsymbol{X}(t)$ 不属于同一类的聚类的权向量在此次竞争中保持不变.

2) 若 $\boldsymbol{X}(t)$ 为无标记样本, 则全部聚类参与竞争, 获胜则奖励, 失败则接受惩罚.

Step 4: 根据式 (13) 计算网络学习目标函数 E, 若 $E < \xi$, 则停止运算, 输出结果, 否则继续.

Step 5: 标记类. 将样本集输入网络进行聚类, 重新辨识各个聚类结果的优势类.

Step 6: 如果 Flag = false, 则根据式 (14) 计算网络学习目标函数增量 ΔE. 若 $\Delta E > \zeta$, 则转 Step 3 执行, 否则 Flag = true.

Step 7: 网络重构. 按照节点拆分-合并规则对输出层竞争节点进行拆分或合并 (按迭代交替进行, 每次最多执行一次拆分/合并).

Step 8: 标记类. 将样本集输入网络进行聚类, 重新辨识各个聚类结果的优势类.

Step 9: 根据式 (13) 计算网络学习目标函数 E, 若 $E < \xi$ 或 $k > K$, 则停止运算, 输出结果; 否则, 更新迭代次数 $k = k + 1$, 转 Step 3 执行.

关于有序标志 Flag, 若没有设置 Flag, 网络经过排序阶段进入重构后出现了前后两次迭代误差增量 ΔE 大于有序阈值 ζ 的情况, 则网络将再次进入排序阶段直到满足 $\Delta E < \zeta$ 才会重新进行节点重构. 显然, 即使在重构阶段, 误差增量偶尔超出设定阈值的情况应该也是允许的, 因此设定了有序标志 ζ, 一旦网络经过排序阶段便不再考虑误差增量因素, 直接进行

细调重构直到学习结束. 在半监督学习过程中, 并不是所有输出神经元节点都有权利参与竞争, 只有优势类与输入模式相同的或未知类节点才可以参与竞争, 不过一旦竞争获胜, 获胜神经元及其邻域内的所有神经元都会受到奖励, 而竞争失败的神经元则根据惩罚规则进行更新. 可以看出, 如果网络输入样本均为无标记样本, 则所有权函数向量也会成为未知类, 上述学习过程将变成通常意义的无监督自组织学习. 此外, 网络可以在训练过程中根据已定义的目标函数对竞争节点进行动态调节, 通过拆分或合并实现网络重构. 因此, 网络对初始竞争层节点数的设置是不敏感的, 提高了其在实际应用中的适应性.

4　仿真实验

本次实验采用 UCI 数据集中的合成控制图表时间序列数据集进行仿真. 该数据集共包含 600 个 6 种不同类的控制图表数据, 每类含有 100 个样本, 每个样本由 60 个系数指标特征进行描述, 经常被用来检测时间序列时变样本的聚类、分类能力. 6 种控制图表类别分别为: 正常 (Normal), 循环 (Cyclic), 增长趋势 (Increasing trend), 减少趋势 (Decreasing trend), 向上偏移 (Upward shift) 和向下偏移 (Downward shift). 每类样本典型示例如图 4 所示.

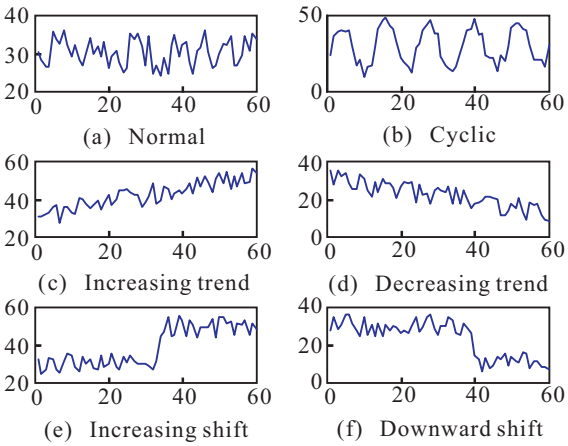


图 4　样本时间序列图

由图 4 可看出, 该时间序列信号的分类依据主要来源于信号的发展趋势和周期性特征, 过多的细节变化信息将会增加网络聚类的复杂性和难度. 因此, 采用多分辨小波分析的方法对原始信号进行分解, 利用分解后的低频系数对原始信号进行重构作为自组织过程神经网络的输入信号. 这是因为信号的发展趋势信息往往隐藏在信号的低频成份中[17], 当然信号的周期性特征也是需要特别保留的. 通过多次实验对比分析, 这里采用 dB3 小波进行分解, 并应用第 2 层低频分解系数进行单层重构, 重构后的信号如图 5 所示.

从每类样本中随机抽取 60 个共 360 个样本组成本次实验用样本集. 为体现本文提出的自组织过程神经网络学习算法的优越性, 分别将其与普通自组织过程神经网络 (CS-PNN) 和 K-均值算法 (K-Mean) 进行对比. STS-PNN 和 CS-PNN 的网络竞争输出层节点数均设置为 3×3, K-Mean 的初始竞争层节点数设置为 8. STS-PNN 算法的不纯度因子和分散度因子设为 $\alpha = 0.8$, $\beta = 0.2$. 经过多次实验对比分析, 3 种算法的性能对比结果如图 6 所示.

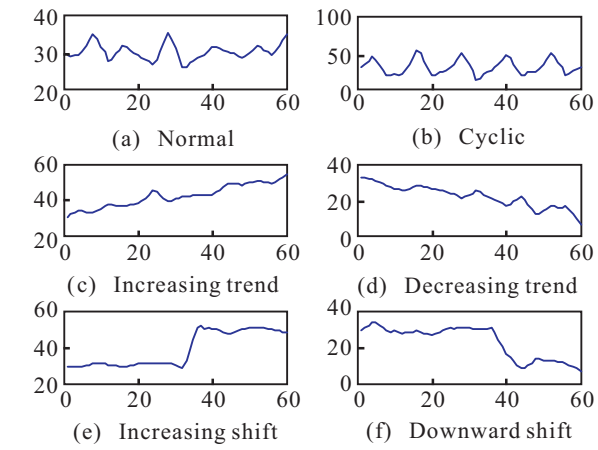


图 5　小波分解重构后的样本时间序列图

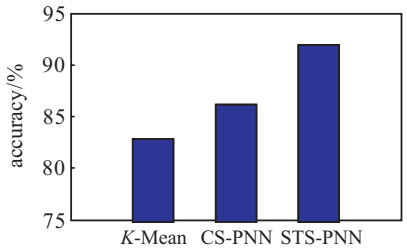


图 6　3 种方法辨识结果对比

实验结果表明, STS-PNN 算法的聚类分类能力比 CS-PNN 和 K-Mean 均有所提高, 其中 K-Mean 算法聚类能力最差. 对此结果可作如下分析: 1) STS-PNN 和 CS-PNN 均利用了输入样本的时间过程特征, 因此聚类能力优于 K-Mean 算法; 2) STS-PNN 采用了奖励-惩罚更新规则, 对权参数初始化设置不敏感; 3) STS-PNN 利用了样本的标记信息进行约束更新, 对网络训练具有较好的指导作用. 因此, STS-PNN 算法的聚类能力优于 CS-PNN. 因为只有 STS-PNN 具有动态调整网络结构的能力, 所以 STS-PNN 最终的聚类数为实际样本类别数 6, 而其他两类算法均为初始设定的数目.

为了测试样本集中标记样本所占比例对 STS-PNN 算法性能的影响, 随机抽取样本集中的 0%, 3%, 60%, 80% 样本作为有标记样本进行聚类分析, 实验结果如图 7 所示. 可以看出, 样本的识别率会随着有标记样本比例的增加而提高. 当样本标记比例达到 80% 以上时, 识别率达到最高.

为了测试不同初始聚类数对 STS-PNN 算法性能的影响, 分别对初始化竞争层节点数为 2×2, 2×3,

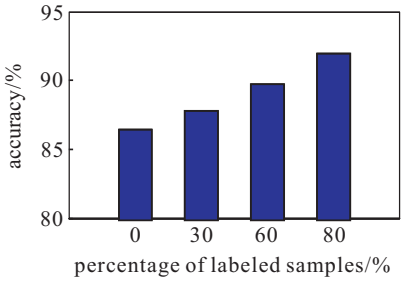


图 7 不同比例标记样本下的性能对比

3×3进行实验分析. 实验结果如表1所示, 算法最终均聚为6类, 初始聚类数目越接近实际样本类别数目, 聚类过程越快. 其中, 初始竞争层节点数为3 × 3时, 聚类数随着迭代的进行其变化过程如图8所示.

表 1 不同初始聚类数下的性能对比

初始聚类数目	最终聚类数目	迭代次数
2 × 2	6	2 007
2 × 3	6	1 813
3 × 3	6	2 115

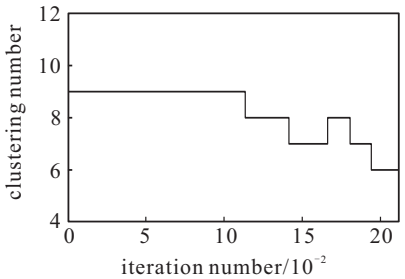


图 8 聚类数目变化图

5 结 论

针对动态样本模式识别及未标记样本信息的有效利用问题, 本文提出了一种基于半监督学习的网络结构自适应的二维自组织过程神经网络模型和训练算法. 在已标记样本的约束指导下, 通过构建可度量时变函数样本间相似性的广义离散Fréchet距离和基于该距离的动态样本聚类算法, 实现了网络结构的自适应调整和有效辨识. 文中建立的模型和方法也可推广到其他非线性动态系统中, 为时变信号的模式识别、聚类分析等问题提供一种新的解决方法.

参考文献(References)

[1] Teuvo Kohonen. Automatic formation of topological maps of patterns in a self-organizing system[C]. Proc of the 2nd Scandinavian Conf on Image Analysis. Espoo, 1981: 214-220.

[2] Teuvo Kohonen. Self-organized formation of topologically correct feature maps[J]. Biological Cybernetics, 1982, 43(1): 59-69.

[3] Teuvo Kohonen. Physiological interpretation of the self-organizing map algorithm[J]. Neural Networks, 1993, 6(7): 895-905.

[4] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position[J]. Biological Cybernetics, 1980, 36(4): 193-202.

[5] 易荣庆, 李文辉, 王铎. 基于自组织神经网络的特征识别[J]. 吉林大学学报: 工学版, 2009, 39(1): 148-153.
(Yi R Q, Li W H, Wang D. Feature recognition based on self-organized neural network[J]. J of Jilin University: Engineering and Technology Edition, 2009, 39(1): 148-153.)

[6] 许少华, 何新贵, 李盼池. 自组织过程神经网络及其应用研究[J]. 计算机研究与发展, 2003, 40(11): 1612-1615.
(Xu S H, He X G, Li P C. Research and applications of self-organization process neural networks[J]. J of Computer Research and Development, 2003, 40(11): 1612-1615.)

[7] Hou C P, Nie F P, Wang F, et al. Semisupervised learning using negative labels[J]. IEEE Trans on Neural Networks, 2011, 22(3): 420-432.

[8] Zhang H, Yu J, Wang M, et al. Semi-supervised distance metric learning based on local linear regression for data clustering[J]. Neurocomputing, 2012, 93: 100-105.

[9] Alex J Cannon. Semi-supervised multivariate regression trees: Putting the 'circulation' back into a 'circulation-to-environment' synoptic classifier[J]. Int J of Climatology, 2012, 32(14): 2251-2254.

[10] John Zhong Lei, Ali A Ghorbani. Improved competitive learning neural networks for network intrusion and fraud detection[J]. Neurocomputing, 2012, 75(1): 135-145.

[11] Bai Y Q, Niu B L, Chen Y. New SDP models for protein homology detection with semi-supervised SVM[J]. Optimization, 2013, 62(4): 561-572.

[12] Tian Xilan, Gilles Gasso, Stéphane Canu. A multiple kernel framework for inductive semi-supervised SVM learning[J]. Neurocomputing, 2012, 90: 46-58.

[13] Herrera M, Izquierdo J, Pérez-García R, et al. Multi-agent adaptive boosting on semi-supervised water supply clusters[J]. Advances in Engineering Software, 2012, 50: 131-136.

[14] Alt H, Godau M. Measuring the resemblance of polygonal curves[C]. Proc of the 8th Annual Symposium on Computational Geometry. Berlin, 1992: 102-109.

[15] Eiter T, Mannila H. Computing discrete fréchet distance[R]. Viena: Information Systems Department, Technical University of Viena, 1994.

[16] John Zhong Lei, Ali A Ghorbani. Improved competitive learning neural networks for network intrusion and fraud detection[J]. Neurocomputing , 2012, 75(1): 135-145.

[17] Hernadez E, Weiss G. A first course on wavelets[M]. New York: CRC Press, 1996: 6-52.

(责任编辑: 孙艺红)