

# 一种自组织混合模型在汇率波动性预测中的应用

倪 禾

(浙江工商大学 金融学院, 浙江 杭州 310018)

**摘要:** 汇率波动性的预测一直以来是研究金融市场者关注的焦点之一, 本文拓展了一种基于自组织神经网络技术的, 用于预测非平稳汇率波动性的自组织混合模型(SOMAR). SOMAR突破了传统模型对平稳性的假设, 变全局建模为局部建模, 使得全局非平稳数据变成局部平稳数据. 同时, 它也是一种基于神经网络技术的非参数回归模型, 结合传统回归模型的简易性和神经网络算法的灵活性, 拓展模型(ESOMAR)提高了对数据异构的适应性. 在对汇率波动性的预测实验中, ESOMAR体现出优于传统回归模型和一些基于其它神经网络模型的效果, 并证明了它在预测金融数据方面所具有的价值.

**关键词:** 自组织神经网络; 波动性; 汇率; 局部建模

**中图分类号:** TP183      **文献标识码:** A

## Exchange rate volatility prediction by an extended self-organizing mixture model

NI He

(School of Finance, Zhejiang Gongshang University, Hangzhou Zhejiang 310018, China)

**Abstract:** Exchange rate volatility prediction has long been the issue that most financial market researchers are concerned with. This article applies a self-organizing-map-based method to the self-organizing-mixture model(SOMAR) for predicting the non-stationary volatility of daily exchange rate. This extended SOMAR(ESOMAR) model is free from the constraint of stationarity which is required by most of the traditional regressive models; it also replaces the global modeling by the local modeling by splitting a non-stationary time series into piece-wise stationary time series episodes. Meanwhile, ESOMAR is a non-parametric neural network regressive model; it combines the simplicity of the traditional regressive model and the flexibility of neural networks, making it adaptive to the heterogeneous data. The prediction results of exchange rate volatility show that the ESOMAR outperforms many traditional regressive models as well as other neural-network-based approaches.

**Key words:** self-organizing neural network; volatility; exchange rate; local model

### 1 引言(Introduction)

国际分工日趋发达, 国际经贸合作高度紧密, 汇率成为了一个不管是国家经济的宏观决策者或是个体进出口商都密切关注的问题. 汇率交易不受交易地点和时间的限制, 是真正的不间断全球市场, 并且其交易数额巨大, 交易对象(货币)的流动性也优于一般金融资产. 对于汇率的研究和预测主要是基于对汇率的绝对数值(升降状况), 或者是基于对汇率的波动性(变化状况)两个方向. 对于意欲通过利用金融工具, 以及合理的资产调配达到规避汇率波动所造成的风险的投资者来说, 准确地预测汇率的波动性比预测汇率的数值更有价值. 尤其是在金融危机时期, 不管是1998年的亚洲金融危机, 还是2008年的世

界性金融危机, 剧烈的汇率变化都严重影响了国际贸易中进出口商品的价格, 和与之紧密联系的各国进出口商.

波动性(volatility)通常是指在给定时间段中, 人们所能够观测到的某一事物随时间的变化而出现的浮动或不定性. 在金融领域, 波动性被定义为在一时变模型中, 一个符合维纳过程的随机因子的即时标准差<sup>[1]</sup>. 因此波动性就是通常意义上所说的不确定性. 根据这样的定义, 波动性不是一个可以直接观测到的值, 而是需要计算得到. 计算波动性可以用多种方法, 比如常见的GARCH模型, Black-Scholes期权定价公式. 而不同的方法就有不同的波动性定义. 本文中的波动性定义为一段时间(数个交易日)内汇率

价格的标准差.

高频汇率数据的实证研究<sup>[2]</sup>表明汇率数据的分布大致符合对称性和厚尾性(leptokurtic). 对厚尾性的一种解释就是数据是独立的正态分布, 并且该分布的均值和方差是时变的<sup>[2,3]</sup>, 即造成汇率变化的市场信息是瞬息万变的. 如果把随机变量 $X_t$ 作为汇率数据, 那么 $X_t$ 的基于到时间 $t$ 的信息 $I_t$ 条件分布可以表示为,

$$X_t|I_t \sim N(\mu_t, f(I_t)), \quad (1)$$

其中:  $N$ 表示正态分布,  $\mu_t$ 表示一段 $X$ 的均值, 而 $f$ 表示一个任意函数或者映射. 正确地估计这个回归函数关系是本文做汇率波动性预测的目标. 在计量经济学中, 最常用的波动性回归方法—GARCH<sup>[4]</sup>一族模型被广泛应用到波动性回归的方法中. GARCH模型提出由一般回归方法(autoregressive moving average, ARMA)来做金融数据的回归分析, 之后得到误差, 虽然可以做到独立分布, 但是它的平方 $\sigma_t^2$ 具有自相关性.

$$\sigma_t^2 = E[(X_t - \hat{\mu}_t)^2|I_t], \quad (2)$$

其中 $\hat{\mu}_t$ 表示 $X$ 的预测均值. 随着计算机性能的提高和计算方法的不断改良, 神经网络作为一种非参数法在回归预测方面逐渐凸显出它独到的优势. 所有基于参数法的回归方法, 都是基于数据平稳性的假设, 这意味着, 得出的定参数模型对于全部参与模型估计的数据都要有效. 这样的假设显然是一个强假设, 对很多现实中的数据, 尤其是被诸多因素影响下的金融数据来说, 是比较难以满足的. 而由此引出的结果, 也就可能不十分满意. 一个直观的解决办法是, 利用多个局部模型替代一个全局模型来分别做回归预测. 这样的好处是, 本来不平稳的整个数据, 可以变成一组子数据集, 而每个子数据集里面的数据相对整个数据集要平稳得多. 本文所提出的方法就是基于这样一个“divider-and-conquer”<sup>[5]</sup>的原则和自组织神经网络的思想, 组合几种神经网络算法的优点, 建立一个组合回归模型.

## 2 波动性估计(Volatility estimation)

关于波动性有很多定义方法<sup>[1]</sup>, 其中比较典型的有: 具有明确结构性, 并假设所有信息是可观测的, 用于离散数据的GARCH型波动性(GARCH volatility); 不假设所有信息都可以观测的, 主要用以估计连续数据的随机波动性(stochastic volatility); 其他还有累计波动性(integrated volatility); 根据B-S期权定价公式产生的隐含波动性(implied volatility), 等等.

本文基于处理离散数据的GARCH波动性估计方法, 来预测汇率数据的日收益 $r_t$ . 日收益可以表示为 $r_t = p_t - p_{t-1}$ , 其中 $p_t = \log(X_t)$ . 日收益可以分解为期望的条件平均收益和一个扰动项, 该扰动项可以表示为一个标准白噪音和条件波动率的乘积.

$$r_t = \mu_t + \sigma_t z_t, \quad (3)$$

其中 $z_t$ 代表一个零均值, 单位方差, 独立不相关的白噪音过程. 这种方法相对来说是最容易的, 它假设所有的相关信息都可观测得到, 并且GARCH模型采用的参数结构是正确、合理的. 在这样的假设下, 如果比较扰动项的平方(在GARCH波动模型中作为波动性的估计值)和真实的波动性, 可以发现由于 $z_t$ 具有单位方差, 所以对 $\sigma_t z_t$ 估计是无偏的,

$$(r_t - \mu_t)^2 = \sigma_t^2 z_t^2 = \sigma_t^2 + \sigma_t^2(z_t^2 - 1), \quad (4)$$

但是由于 $z_t^2$ 的影响, 这样的估计难免会引入一些误差. 采用随机波动性的估计方法, 可以改善这一情况<sup>[1]</sup>.

GARCH模型<sup>[4]</sup>的特点就是利用一个时间序列的历史方差(波动性)去解释描述它未来的方差(波动性). 两者的关系, 可由一个回归方程GARCH( $p, q$ )来描述,

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^q \alpha_i \epsilon_{t-i}^2 + \sum_{i=1}^p \beta_i \sigma_{t-i}^2. \quad (5)$$

其中 $\epsilon = r_t - \mu_t$ ,  $p, q$ 是根据对实验数据的先验知识预定的, 而 $\alpha, \beta$ 是根据最大似然估计(maximum likelihood estimation)的方法得到的. GARCH模型在预测金融数据的波动率上应用广泛, 如股市数据<sup>[6]</sup>, 汇率数据<sup>[7]</sup>, 等等.

虽然很多的研究结果表明GARCH模型在金融数据建模方面具有一般回归模型所不可比拟的功效, 但其模型本身仍然具有一定的局限性. 比如GARCH, 首先要求确定参数的结构以及误差的分布情况, 然后才能估测参数的值, 而正确确定参数的结构需要对数据有先验知识, 否则就只能作近似的猜测. 这是参数法(parametric method)模型共同的局限性, 解决的办法是把GARCH逐渐引入到非参数方法(nonparametric method)<sup>[10]</sup>. 另外, GARCH是一个全局模型, 会把所有的数据用于构建一个模型. 由此带来的问题是如果用于建模的数据非平稳, 那么得到的结果也就不可靠了. 解决方案除了常规的利用数据预处理来变非平稳数据为平稳数据外, 可以把非平稳数据切割开来, 成为分段平稳数据, 然后采取分而治之的方法做回归. 已经用于GARCH模型的有

代表性的方法见文献[11].

### 3 基于自组织神经网络的拓展组合回归模型(Extended mixture regressive model based on SOM)

本文的方法是建立于上述方法之上的, 基于自组织神经网络(self-organizing neural network)的一个具有能保持拓扑结构的回归算法. 自组织神经网络<sup>[12]</sup>是一种具有自我学习能力的人工神经网络, 它能把输入空间中的高维训练数据映射到一个通常是低维、离散的空间, 并保持数据在原空间的拓扑结构. 由于它独特的邻域方程(neighborhood function), 自组织神经网络的训练过程还可自行规避陷入局部最小的情况出现. 自组织神经网络的用途非常广泛, 其中一个重要分支就是在时间序列的聚类分析上. 因为它具有保持拓扑结构的“学习能力”, 不但可以不受训练数据平稳性的影响, 还可以把原来不平稳的训练数据归类成几组平稳数据<sup>[5,13]</sup>.

自组织神经网络的这个特点可以用以解决预测汇率波动性中遇到的数据可能不够平稳的问题. 不同与GARCH模型利用一个回归方程, 本文采用的方法-自组织自回归模型, 是一个具有自组织神经网络学习特点的自回归模型. 此模型基于普通自回归模型参数的学习方法,

$$\mathbf{w}(t) = \mathbf{w}(t-1) + \eta e(t)\mathbf{x}(t), \quad (6)$$

其中 $e(t) = \mathbf{x}(t) - \mathbf{x}(t)^T \mathbf{w}(t-1)$ 是预测误差,  $\eta$ 是控制学习速度的常数步长. 这个学习方法是根据迭代最小二乘法(recursive least mean squares)推导出来的, 用于根据每步输入 $\mathbf{x}(t)$ 在线学习自回归模型权重 $\mathbf{w}$ . 自组织自回归模型变单个权重为 $N$ 个局部回归模型权重 $\mathbf{w}_i, i = 1, \dots, N$ , 在每一次模型得到输入数据时, 所有 $N$ 个局部回归模型的权重会根据下式更新,

$$\mathbf{w}_i(t) = \mathbf{w}_i(t-1) + h(r, i)e_i(t)\mathbf{x}(t), \quad (7)$$

式(7)和式(6)的最大区别是式(7)用一个邻域函数 $h(r, i)$ (其中 $r, i$ 表示该邻域函数的参数)替代式(6)中的常数步长 $\eta$ . 这个邻域函数来自于自组织神经网络的学习方程<sup>[12]</sup>, 邻域技术是自组织神经网络的技术核心之一, 在本文研究的汇率波动性预测上不仅能够使得多个局部模型同时进行训练, 还避免任何一个局部模型的训练陷入局部最小.  $h(r, i)$ 函数通常情况下是一个类高斯函数, 即具有单个函数峰值(最大值) $h^*(r^*, i^*)$ , 峰值边缘(邻域)其它值会根据其变量 $(r, i)$ 和峰值所对应变量 $(r^*, i^*)$ 的“距离”进行变

化, “距离”越大, 对应 $h(r, i)$ 的值越小. 当对应某个输入下 $h(r, i)$ 值趋于零, 则相当于局部回归模型 $i$ 在这个输入的情况下不进行学习. 同时由于这个邻域方程的影响, 使得当某一个局部模型 $i$ 的学习陷入一个局部最小时, 它的邻居 $\dots, i-1, i+1, \dots$ 通过邻域方程的影响, 把训练陷入局部最小的局部模型 $i$ 的参数拖出局部最小.

邻域方程的具体形式可以有多种, 原则是能使对象数据的训练能够高效收敛. 最常见的邻域函数就是高斯函数, 或者说在没有其他先验知识的情况下所建议选择的邻域函数

$$h_{r,i}(t) = \alpha(t) \exp\left(-\frac{f(r, i)}{2\sigma^2(t)}\right), \quad (8)$$

其中 $\alpha(t), \sigma(t)$ 是两个收缩函数, 随着时间的推移, 这两个函数的值会单调递减.  $f(r, i)$ 是确定待更新权重 $\mathbf{w}_i$ 对应的神经元 $i$ 到最合适神经元 $i^*$ 的距离. 通常为欧几里德距离 $f(r, i) = \|\mathbf{r}_i^* - \mathbf{r}_i\|^2$ . 在 $\mathbf{r}_i = \mathbf{r}_i^*$ 时取得此邻域函数的最大值. 邻域函数的效果在图1中简单演示. 图1(a)表示邻域函数分布在神经网络上的由于位置不同而造成的效果不同; 图1(b)表示一个典型的高斯邻域函数的图形, 其函数值随着和最合适神经元距离的增加而减少.

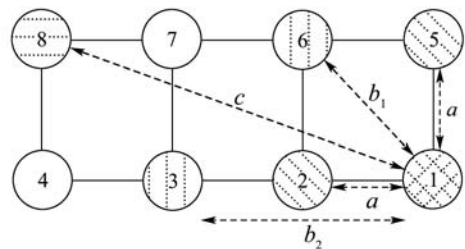


图 1(a) 邻域函数在神经网络上的不同效果  
Fig. 1(a) The various values of neighborhood function due to the locations of different neurons

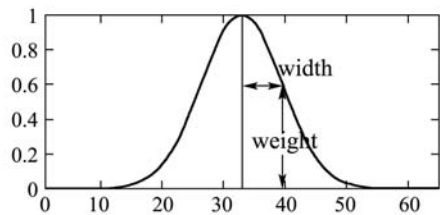


图 1(b) 一个典型的高斯邻域函数图形  
Fig. 1(b) Illustration of a typical Gaussian neighborhood function

另一个重要问题是确定 $i^*$ , 即最合适模型(“the best fit unit” 或者“winner”). 找到合适的 $i^*$ 对整个学习过程有非常重要的作用. 学者们提出了很多方法来确定 $i^*$ , 其中有Temporal SOM<sup>[15]</sup>, Recurrent

SOM<sup>[16]</sup>, Recursive SOM<sup>[17]</sup>, Merge SOM<sup>[18]</sup>, 等等. 这些方法大多是源于意图把时间序列里面前后数据中所隐含的顺序信息包含到模型训练过程当中去. 简而言之, 由于数据是连续的, 所以本次输入数据和它前一次的输入数据应该有联系, 或者说在单位时间间隔内不应该发生随机的突变. 因此有理由让上次的最适合神经元 $i^*(t-1)$ 比其它神经元更有机会成为本次的最适合神经元 $i^*(t)$ . 处理方法是在计算新的 $i^*(t)$ 时, 给 $i^*(t-1)$ 加上一个比其他 $i$ 大的权重. 设 $f(r, i)$ 是每个神经元 $i$ 到最适合神经元的“距离”,  $\hat{f}(r, i)$ 是通过计算得到的“距离”, 则有

$$f(r, i)_t = (1 - \beta)f(r, i)_{t-1} + \beta\hat{f}(r, i)_t, \quad (9)$$

其中,  $\beta$ 是一个小于1的非负数, 用于调节加权值的大小,  $\beta$ 越小, 相应加权越大; 当 $\beta = 1$ 时, 相当于没有加权, 和普通自组织网络找最合适神经元一样. 经过这样的迭加, 使得“前最合适神经元”对未来训练的“影响力”随着时间 $t$ 增加逐渐减小, 因此系数 $\beta$ 也被称为“leaky integrate”<sup>[16]</sup>.

关于计算 $f(r, i)$ , 一般用以求出自回归模型参数的迭代最小二乘法是利用误差 $e(t) = x(t) - \mathbf{x}(t)^T \mathbf{w}(t-1)$ 求得的, 这个方法不适用于计算 $f(r, i)$ . 因为那样求出的 $e(t)$ 的绝对值随机递减(stochastic gradient descend, SGD), 而不是单调递减. 现在所需要的是在每一次得到输入时, 都可以找到唯一的 $i^*$ , 也就是 $f(r, i)$ 唯一的极值. 因此用误差 $e(t)$ 接近白噪声(white noise, WN)的程度作为 $f(r, i)$ 的结果<sup>[13]</sup>. 具体的方法通过计算一组相邻误差 $e(t)$ 的自相关系数的绝对值的和(sum of the autocorrelation coefficients, SAC), 最小的即为最接近于白噪声, 也就是相对应的 $i^*$ .

$$\hat{f}(r, i)_t = \sum_{k=0}^r \left\| \frac{1}{(r-k)\sigma_i^2} \sum_{t=1}^{r-k} (e_i(t) - \mu_i) \times (e_i(t+k) - \mu_i) \right\|, \quad (10)$$

其中:  $r$ 是一组相邻误差的个数,  $\mu_i, \sigma_i$ 分别是那组误差的均值和标准差,  $k$ 是求自相关系数时的延长度. 这样计算出来的最小值即对应最适合神经元 $\hat{f}(r, i^*) = \min \hat{f}(r, i)$ . 用这样的“距离”可以在训练过程中对于每一次输入都找到唯一的最合适神经元 $i^*$ . 如果采用均方差(mean-squared-error, MSE)的方法, 则无法得到唯一解, 具体来说, 对于同一个(组)输入时, 如果利用均方差, 会有很多组回归模型的参数对应局部最小值. 在图2中展示了在同一组输入下, 分别由SAC方法和均方差方法求出的

最合适神经元(图中最小值的位置所对应的位置). 其中, 上图表示了由SAC方法找到了一个全局最小值, 该值对应的位置即为用SAC方法找到的最合适神经元, 这样找到的神经元所代表的参数非常接近于 $[0.2, -0.2]$ ; 下图则是由MSE来寻找最合适神经元. MSE找到的是非常多的局部最小值, 完全掩埋了真正的全局最小值, 以及它对应的真正最合适神经元.

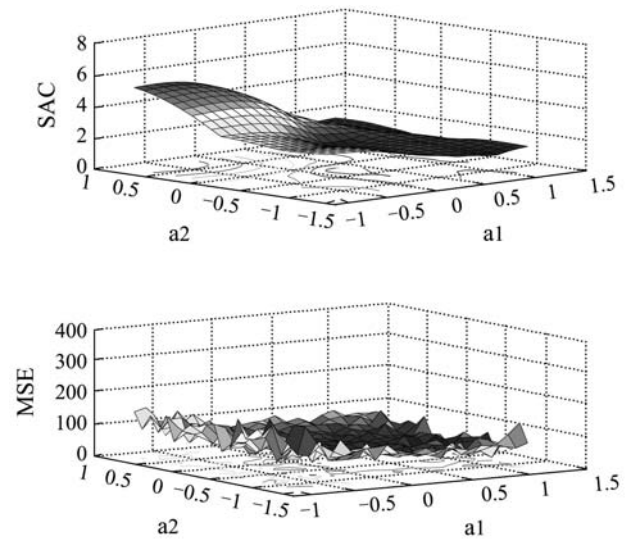


图2 在同样人工产生数据(由回归参数 $[0.2, -0.2]$ 产生)输入下, 用SAC和MSE分别寻找最合适神经元  
Fig. 2 The best fit neurons picked by SAC and MSE with the same artificial data (generated by a simple AR model parameter  $[0.2, -0.2]$ )

同时还要考虑用于做回归预测的一组局部模型是否是同构的(homogenous), 也就是说它们的结构, 如参数的个数是不是都相同, 贸然地假设具有很大的风险, 在同类文章没有能够找到明确的证据的情况下, 本文假设那些局部模型有可能不是同构的. 相应地解决非同构的办法是使用两层网络, 第一层网络的作用是根据数据的变化(速度、形状等等)把数据分成几类, 每一类数据由一个局部模型来处理, 每一个局部模型对应于第一层网络中每一个神经元. 第二层网络中有 $N$ 个子网, 对应于第一层网络中的 $N$ 个神经元, 每一个子网由若干个神经元构成, 而子网中每一个神经元代表一种局部模型的结构. 第二层网络的作用就是通过累积学习来最后确定每一个子网当中最优秀(最合适)的一个神经元, 也就对应于最合适的局部模型的结构. 用这样的方法, 可以避免对是否同构做不确定性的假设, 而把所有可能的结构都作为备选放到第二层网络中的每一个子网当中去. 图3展示了上文提到的两层网络的构成形式. 第一层网络中, 神经元是处于一个一维网络中; 在第

二层网络中, 每一个子网用一个虚线框表示, 表征神经气网络没有拓扑限制. 而第一层网络中每一个神经元都一一对应第二层网络里面的每一个子网.

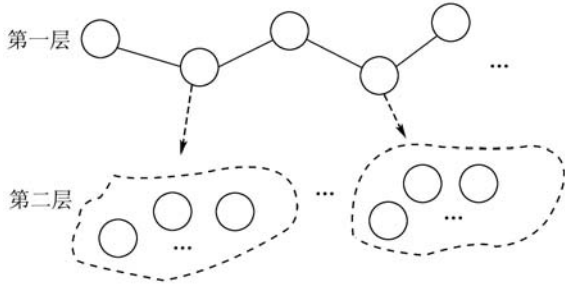


图3 两层网络的构成示意图

Fig. 3 Two layers' network structure

为了能够进行无偏学习, 本文采用神经气(neural gas, NG)<sup>[19]</sup>模型来对第二层网络中子网神经元进行训练. 神经气是一种和自组织神经网络同族的神经网络算法. 相比于有相对固定网络结构的自组织网络来说具有无拓扑限制的优势. 神经气网络有类似于自组织神经网络的邻域函数, 但是其邻域是通过每次的排序得到的, 而不是按照任何事先限定的网络结构决定的. 因此, 第二层网络中每个神经元所代表的局部回归模型参数的更新函数为:

$$\mathbf{w}_{i,j}(t) = \mathbf{w}_{i,j}(t-1) + g(\lambda, k_{i,j}) h_{r,i}(t) e_i(t) \mathbf{x}(t), \quad (11)$$

其中

$$g(\lambda, k_{i,j}) = \eta(t) \exp(-k_{i,j}/\lambda(t)), \quad (12)$$

$$\lambda(t) = \lambda_0 (\lambda_f / \lambda_0)^{t/t_{\max}} \quad (13)$$

是神经气模型所采用的典型邻域函数,  $\lambda_i, \lambda_f$  两个参数主要用于控制邻域的边缘.  $\eta(t)$  是一个学习率, 一个随着时间增加而递减的参数. 在更新方程中, 每一个局部回归模型的参数为  $\mathbf{w}_{i,j}(t)$ ,  $i$  表示对应该局部模型在第一层网络中的位置,  $j$  表示它在第二层网络中的位置. 每一个参数  $\mathbf{w}_{i,j}(t)$  和两个邻域函数  $g(\lambda, k_{i,j})$  以及  $h_{r,i}$  都有关. 在每一次输入数据来临时, 首先根据局部回归模型最可能阶数(通过BIC来选择)选取合适长度的输入矢量, 并根据SAC测量方法找到在第一层网络中的最优神经元, 继而确定邻域函数  $h_{r,i}$ . 然后进入到第二层网络, 在每个子网中根据局部回归模型的不同阶数调整输入矢量的维数, 并再次利用SAC方法为该子网中的几个局部回归模型排序  $k_{i,j}$ . 最后根据式(11)逐个更新所有子网中的每一个局部回归模型的参数, 记录每个子网中在每次更新时的最合适单位  $j^*(t)$ . 整个学习过程结束之后, 在每一个子网中挑选出一个局部回归模型, 这个模型就是在训练当中, 被选择为最合适模型  $j^*(t)$  次数最多的一个. 这个局部回归模型就成为

该子网的代表, 或者说最终的学习结果. 相对应第一层中  $N$  个神经元, 结果中就有  $N$  个局部回归模型, 它们的阶数有可能是不同的. 当应用于预测时, 预测数据从这  $N$  个局部回归模型中挑选最合适的进行回归预测. 挑选的方法可以有多种, 本文实验中利用的是把原有训练数据放到学习所得的  $N$  个局部回归模型中, 挑选出每个模型的最适用典型数据. 每组预测数据和  $N$  组最适用典型数据比较, 找到最适合使用的局部回归模型来进行下一步预测.

综上所述, 此方法是作者提出的SOMAR模型<sup>[13,14]</sup>的拓展, 结合反馈神经网络的特点, 引入一级反馈, 加强了相邻学习数据之间所包含的时间信息在学习环节中的应用. 并且利用两层神经网络的方法, 确定每一个局部回归模型的最优阶数, 突破了原来所有局部回归模型都必须使用默认的阶数, 而不能根据数据的变化相应的进行动态调整的局限性; 增加学习调整期(fine tuning, FT), 通过重复试验找到合适的学习调整期的开始时间, 合适的SAC值用以滤去不合适的输入. 因为计算SAC的需要, 每次的输入需要一组输入, 而不是一个输入. 一组相邻的输入一般来说是由一个回归模型主要产生, 也可能是由多个回归模型联合产生. 在第2种情况下, 显然用这些数据去选择一个相对的最合适回归模型就不太恰当, 因此在学习调整期应该忽略那些输入组.

#### 4 对比实验(Comparative experiment)

在对比实验中, 主要是检验基于自组织神经网络的回归方法在预测汇率波动性方面, 和GARCH, ARMA, MLP(BP算法训练的多层感知器), 以及自组织神经网络方法比较, 有多大的效果提升. 实验所用的数据是国际主要的几种货币对, 美元对英镑, 美元对欧元, 美元对港币各十年左右(2500个连续交易日)的日兑换价格(数据来源: PACIFIC汇率数据库sauder school of bussiness, UBC, <http://fx.sauder.ubc.ca/>). 人民币的汇率主要是因为在大部分的时间都是和美元挂钩的, 所以和其它采用自由兑换政策的货币不具有足够的可比性.

实验的简要步骤是, 首先应用每个货币对的所有历史收益(利用日兑换价格的自然对数差进行兑换价格到收益的计算)数据, 根据BIC找出最优回归模型阶数. 然后利用GARCH( $p, q$ )模型进行波动性的估计预测. 再根据GARCH所得收益回归模型参数, 计算出收益误差的方差. 用一个可变长度移动窗口(rolling window)把一维的方差序列转变成一组按照时间序列排列的矢量. 最后利用本文提出的基于自组织神经网络的回归模型去预测

波动性. 并和GARCH模型直接得出的预测值, 及用ARMA, MLP, 自组织神经网络方法间接得到的预测值(意在比较本文提出的ESOMAR模型和其他常用回归模型)比较. 其中ARMA和MLP系采用全局模型, 对用GARCH模型(AR)部分产生出来的误差的方差进行回归得出波动率. 自组织神经网络本身也可以用回归模型处理时间序列数据, 对用GARCH模型(AR)部分产生出来的误差的方差采用聚类法进行回归分析, 进而得到预测波动率.

以美元对英镑为例, 用前2000期数据进行学习训练, 后200期数据做预测实验. 首先把兑换价格变为收益, 用BIC准则找出最佳回归模型阶数AR= 4. 以经验数值 $p, q$ 为参数, 用GARCH( $p, q$ )做出收益数据的误差, 再根据GARCH模型对误差平方 $\epsilon^2$ 和波动性 $\sigma$ 的关系 $\epsilon_t \sim N(0, \sigma_t)$ 以及经验 $q$ 值, 计算出误差所对应的即时波动性. 如图4所示.

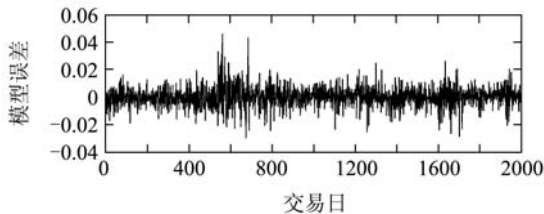


图 4(a) 美元兑英镑收益的AR模型误差

Fig. 4(a) The AR modeling errors of USD to GBP return

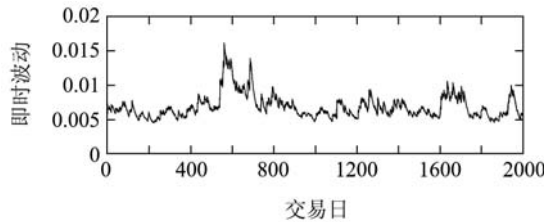


图 4(b) 表示的是该误差所对应的波动性.

Fig. 4(b) The corresponding volatilities

利用这些训练数据, 可以找出本文提出的回归模型的各项参数.

为了更好地量化ESOMAR模型的预测结果, 在表1, 2, 3中罗列了预测值和真值之间的均方差MSE, 和预测值的方差(variance)两个指标. 使用方差的原因主要是这里的预测值为几次独立实验的均值. 较小的方差就对应着较大的置信度. 表中NG表示神经网络, MSE值应 $\times 10^{-3}$ , Var值应 $\times 10^{-6}$ .

表 1 美元兑英镑

Table 1 USD vs GBP

	MSE	Var		MSE	Var
ESOMAR	4.231	2.142	SOMAR	4.342	2.201
ARMA	4.823	2.793	MLP	4.710	2.534
GARCH	4.532	2.238	NG	4.602	2.348

表 2 美元兑欧元

Table 2 USD vs EURO

	MSE	Var		MSE	Var
ESOMAR	3.223	2.013	SOMAR	3.514	2.203
ARMA	4.037	2.672	MLP	4.001	2.641
GARCH	3.871	2.216	NG	3.724	2.41

表 3 美元兑港币

Table 3 USD vs HKD

	MSE	Var		MSE	Var
ESOMAR	5.176	2.562	SOMAR	5.310	2.612
ARMA	5.993	2.982	MLP	5.815	2.986
GARCH	5.551	2.863	NG	5.761	2.699

实验结果显示ESOMAR模型是一个可靠的回归模型, 在汇率波动性预测上有可以超越传统GARCH模型和其它一些常用回归模型的表现. 它的优势主要表现为对数据非线性和非平稳性具有良好适应能力, 另外作为一种非参数方法, 它的结构是由它的训练数据, 而不是模型本身决定的, 甚至在训练时不需要很多对训练数据的先验知识.

## 5 总结和结论(Summary and conclusion)

本文拓展了一个基于自组织神经网络思想的回归模型(ESOMAR). 该模型利用了自组织神经网络的构架, 构筑了一组局部回归模型. 每一个局部模型都能根据实际训练数据的需要进行结构和参数的调整, 模型之间还能保持有一定的拓扑关系, 该关系保证了模型之间的连续性以及模型的训练不会进入局部最小. 因为一段非平稳时间序列有可能被分成若干个平稳时间序列, 原本的复杂非线性情况也会因为局部模型的分解作用而变得相对容易, 所以使用多个局部模型替代单个全局模型, 具有能突破对数据非平稳性的限制. 当然ESOMAR方法还远远不是一个完美的回归预测模型, 比如说目前对某一点未来的预测还是基于对该点以及该点附近一段数据的一个最佳局部模型来完成; 同时如果加入使用几个次佳的局部模型就有可能达到更好的效果.

汇率数据是一种运用非常广泛的重要的金融时间序列, 但同时也是一种具有非平稳、非线性的不容易进行建模和预测的时间序列. 通过对汇率波动性的实验结果表明, ESOMAR模型对处理、预测复杂的金融时间序列比传统方式具有独到的算法优势. ESOMAR, 辅以相应的变化, 同样可以应用到其它的金融时间序列的建模中去, 为负责金融数据分析的专家的发掘工作提供有效的、客观的数字信息. 同时也佐证了它所代表的神经网络算法在金融经济方面的应用, 虽然尚处于起步阶段, 但有理由相信能取得累累硕果.

## 参考文献(References):

- [1] ANDERSEN T G, BOLLERSLEV T, CHRISTOFFERSEN P F, et al. *Volatility and Correlation Forecasting*[M]. North Holland: Handbook of Economic Forecasting, 2005.
- [2] DIEBOLD F X. *Empirical Modeling of Exchange Rate Dynamics*[M]. New York: Springer-Verlag, 1988.
- [3] FRIEDMAN D, VANDERSTEEL S. Short-run fluctuation in foreign exchange rates[J]. *Journal of International Economics*, 1982, 13(1): 171 – 186.
- [4] BOLLERSLEV T. Generalized autoregressive conditional heteroskedasticity[J]. *Journal of Econometrics*, 1986, 31(3): 307 – 327.
- [5] CAO L J. Support vector machines experts for time series forecasting[J]. *Neurocomputing*, 2002, 51(19): 321 – 339.
- [6] YU J. Forecasting volatility in the New Zealand stock market[J]. *Applied Financial Economics*, 2002, 12(3): 193 – 202.
- [7] KEARNEY C, PATTON A J. Multivariate GARCH modeling of exchange rate volatility transmission in the European monetary system[J]. *Financial Review*, 2005, 35(1): 29 – 48.
- [8] 苏岩, 杨振海. GARCH(1,1)模型及其在汇率条件波动预测中的应用[J]. *数理统计与管理*, 2007, 26(4): 57 – 62.  
(SU Yan, YANG Zhenghai. GARCH(1,1) model and its application in forecasting conditional volatility of exchange rate[J]. *Application of Statistics and Management*, 2007, 26(4): 57 – 62.)
- [9] 伍戈, 姜波克, 唐建伟. 外汇市场信息与汇率波动性研究[J]. *财经研究*, 2002, 28(7): 15 – 19.  
(WU Ge, JIANG Boke, TANG Jianwei. A study on information of the foreign-exchange market and volatility of foreign-exchange rate[J]. *The Study of Finance and Economics*, 2002, 28(7): 15 – 19.)
- [10] BUHLMANN P, MCNEIL A J. An algorithm for nonparametric GARCH modelling[J]. *Computational Statistics & Data Analysis*, 2002, 40(4): 665 – 683.
- [11] FORNARI F, MELE A. Signal and volatility switching ARCH model[J]. *Journal of Applied Econometrics*, 1997, 12(1): 49 – 65.
- [12] KOHONENE T. *Self-organising Maps*[M]. Berlin: Springer, 2001.
- [13] NI H, YIN H. Time-series prediction using self-organising mixture autoregressive network[C] // *Proceedings of the Intelligent Data Engineering and Automated Learning*, Berlin: Springer-Verlag, 2007, 4881: 1000 – 1009.
- [14] NI H, YIN H. Self-organising mixture autoregressive model for non-stationary time series modelling[J]. *International Journal of Neural Systems*, 2008, 18(6): 1 – 12.
- [15] CHAPPELL G, TAYLOR J. The temporal Kohonen map[J]. *Neural Networks*, 1993, 6(3): 441 – 445.
- [16] KOSKELA T. Time series prediction using recurrent SOM with local linear models[J]. *International Journal of Knowledge-Based Intelligent Engineering Systems*, 1997, 2(2): 60 – 68.
- [17] VOEGTLIN T, DOMINEY P F. Recursive self-organising maps[J]. *Neural Networks*, 2002, 15(8): 979 – 991.
- [18] STRICKERT M, HAMMER B. Merge SOM for temporal data[J]. *Neurocomputing*, 2005, 64: 39 – 72.
- [19] MARTINETZ T, SCHULTEN K. A “neural-gas” network learns topologies[J]. *Artificial Neural Networks*, 1991, 1: 397 – 402.

## 作者简介:

倪禾 (1978—), 男, 2008年于英国曼彻斯特大学电子工程系取得博士学位, 现为浙江工商大学金融学院讲师, 研究方向为神经网络模型及模式识别方法在金融时间序列建模方面的应用, 如应用自组织神经网络、支持向量机、遗传算法等用于汇率预测, E-mail: nihe@mail.zjgsu.edu.cn.