



# 基于 Boosting 梯度下降理论的时间序列建模方法

高云龙<sup>①②\*</sup>, 潘金艳<sup>③</sup>, 吉国力<sup>①</sup>, 高峰<sup>②</sup>

① 厦门大学自动化系, 厦门 361005;

② 西安交通大学系统工程研究所, 机械制造系统工程国家重点实验室, 西安 710049;

③ 集美大学信息工程学院, 厦门 361021

\* E-mail: gaoyl05@yahoo.com.cn

收稿日期: 2010-09-27; 接受日期: 2011-01-26

国家自然科学基金(批准号: 60974101)和教育部新世纪优秀人才计划(批准号: NCET-06-0828)资助项目

**摘要** 在预测问题中有很多实际问题具有时间序列特性, 如何分析并建立合理的时间序列预测模型具有重要的理论价值与广泛的应用前景. 随着对时间序列研究的逐步深入, 时间序列预测模型变得越来越复杂, 建模过程对设计技巧依赖性越来越强. 如何设计一种不依赖于操作者的能力和经验, 并且具有良好效果的建模方法呢? 本文在集成学习理论的基础上, 提出了基于 Boosting 梯度下降理论的自回归模型集成 (AR Ensemble Learning, AREL) 建模法, 从理论上分析并论证了该建模法的有效性. 为了避免 AREL 精确拟合每一个训练样本点, 在建模过程中引入了  $\epsilon$ -不敏感损失函, 从而降低随机噪声对时间序列预测模型的影响. 进而为了提高 AREL 对非平稳时间序列的处理能力, 提高算法的鲁棒性, 防止发生过拟合, 降低算法对模型阶次设置的敏感性, 提出了基于 AREL 的加权  $k$  近邻(weighted kNN)预报法. 通过实例测试, 并对结果进行了讨论, 验证了所提出的建模法与预报方法的有效性.

**关键词**

时间序列  
Boosting  
集成学习  
过拟合

对未来的预测、预报是人类智慧的集中体现, 通过对未来的预测、预报可以指导人类的行为, 对未来情况做出合理的判断, 从而提高人类对未来的把握与控制能力. 在预测问题中有很多实际问题具有时间序列特性, 如何分析并建立合理的时间序列预测模型, 从而提高对时间序列的预测精度, 具有重要的理论价值与广泛的应用前景.

最典型的预测问题是对电力系统负荷进行预测. 电力系统负荷预测是电网运行调度的基础. 近年来随着智能电网技术的发展, 电网调度逐渐向智能化、数字化过渡. 信息网络系统通过智能化控制实现电网的精确供能、对应供能、互助供能和互补供能, 将

能源利用效率和能源供应安全提高到全新的水平<sup>[1, 2]</sup>. 对具有自备电厂的大型用电企业进行自平衡调度是更好地实现精确供能、对应供能的一个有效途径<sup>[3, 4]</sup>. 比如对一个具有自备电厂的大型用电企业, 进行自平衡调度, 可以减少电能远距离输送的耗损, 提高企业利用余热、余压、高炉尾气的积极性, 从而降低污染与温室气体排放. 而且对企业进行自平衡调度也可以实现对应供能, 分区域供能, 避免部分电力系统故障对整个电力系统造成连锁反映, 从而提高电力系统的健壮性. 而实现自平衡调度的一个关键技术就是对企业用电负荷进行预测.

电力系统负荷预测技术经历了几十年的发展,

英文版发表信息: Gao Y L, Pan J Y, Ji G L, et al. A time-series modeling method based on the boosting gradient-descent theory. Sci China Tech Sci, 2011, 54: 1325-1337, doi: 10.1007/s11431-011-4340-1

先后产生了一系列科学、有效的方法, 如时间序列参数模型方法、神经网络方法、专家系统等, 并已在电力系统中得到了广泛的应用. 众所周知, 电力负荷数据绝大多数是时间序列数据, 时间序列同时蕴含着数据顺序和数据大小, 表现出客观世界的某一动态过程, 能反映出客观世界及其变化的信息. 因此在电力系统负荷预测中通常采用时间序列统计建模和统计推断进行分析研究.

由于时间序列在实际应用中非常重要, 众多的科研工作者展开了大量的研究, 得到许多重要的研究成果. 最简单的时间序列预测方法是持续法, 即把最近一点的观测值作为下一点的预测值<sup>[5]</sup>. 其他预测方法包括统计学方法, 和以神经网络, 支持向量机(SVM)为代表的人工智能方法<sup>[6]</sup>. 其中神经网络具有很好的非线性逼近能力, 但存在模型结构难以确定, 易于出现过度训练和训练不足, 陷入局部最小, 且对连接权初值敏感, 过度依赖设计技巧. 而统计学方法, 特别是自回归时间序列预测方法(AR 模型), 因为“自回归模型(AR 模型)的模型参数凝聚了系统状态的重要信息, 准确的 AR 模型能够深刻、集中地表达动态系统的客观规律, 而且大量研究表明, AR 的自回归参数对状态变化规律反映最为敏感, 具有对短序列建模的能力及能采用快速算法的特点, 满足及时性的要求”<sup>[7, 8]</sup>, 因而在当前时间序列预测中, 特别是对风电场风速与发电功率预测研究中逐渐受到了重视.

虽然 AR 模型对实际生活中的许多时间序列数据具有相当好的逼近性能, 但是在某些情况下简单的线性模型不能很准确地描述事物潜在的系统状态变化的客观规律, 表现出一定的局限性<sup>[9]</sup>, 而且 AR 模型要求时间序列具有平稳性、正态性、独立性, 也不适用于复杂时间序列. 当前对统计回归建模方法的研究主要集中在以下两个方面.

1) 如何消除时间序列的非平稳性. 如: 杨秀媛等人<sup>[6]</sup>采用差分算子进行很多次日差分和周差分运算以消除时间序列中的周期因素; Brown 等人<sup>[10]</sup>用 AR 模型拟合小时风速时, 采用了指数变换、规格化、对每月或季单独进行拟合等措施来消除日非平稳和季非平稳; 孙春顺等人<sup>[11]</sup>提出将单变量小时风速时间序列向量化, 以消除日周期非平稳;

2) 如何分析并建立合理的时间序列预测模型, 以适用于复杂时间序列. 这些方法包括: 周期自回归模型(Periodic AR, PAR)<sup>[12, 13]</sup>, 向量自回归模型

(Vector Autoregression, VAR)<sup>[14-17]</sup>, 门限自回归模型<sup>[9]</sup>, 子集门限自回归模型, 时变自回归模型<sup>[18]</sup>, 非线性自回归模型<sup>[19-22]</sup>等等. 如: Granger 和 Joyeux<sup>[23]</sup>推广 ARMA 为部分 ARMA 模型, Hannan 和 Deistler<sup>[24]</sup>提出向量 ARMA 模型, Chen 和 Tsay<sup>[25]</sup>提出非线性可加自回归模型, 此时所谓的线性模型已经包含有一定的非线性特征. 接着, 跳出线性的限制, 涌现出许多非线性模型, 如双线性模型等<sup>[9]</sup>.

虽然上面两个方面都有大量的研究成果出现, 并在各个领域都取得令人满意的结果. 但一般时间序列法要求采用差分算子进行很多次日差分和周差分运算以消除时间序列中的周期因素, 这不仅耗费时间, 而且使得模型的表达式十分复杂<sup>[26, 27]</sup>. 而通过建立复杂的回归模型以适用于复杂时间序列, 难点在于, 模型的建立变得越来越复杂, 模型诊断检验变得越来越困难. 而且复杂的回归模型待辨识的模型参量多且分属不同的类型, 计算复杂. 那么有没有一种不依赖于操作者的能力和经验, 对问题不需要过多先验知识, 而又能被普通的工程技术人员使用, 同时产生良好效果的方法呢? 近十几年来兴起的集成学习就是这样一种方法. 集成学习方法通过训练多个弱学习算法并将其结论进行合成, 可以显著地提高学习系统的泛化能力, Boosting 算法作为集成学习算法的主要代表, 得到了广泛的研究和应用, 其中, AdaBoost 算法更是在理论分析和实际应用中都获得了巨大成功<sup>[28-30]</sup>. 本文在集成学习的基础上, 主要研究 AR 模型通过集成是否可以表现出一定的非线性特征, 从而对实际生活中的许多时间序列都具有良好的逼近性能, 进而准确地描述事物潜在的系统状态变化的客观规律.

## 1 时间序列法基本原理

### 1.1 时间序列自回归模型

自回归(Autoregressive, AR)模型是现代时间序列分析与随机控制理论中广泛研究与应用的基本模型之一<sup>[31, 32]</sup>. 在著名的 Box-Jenkins 建模方法论中, 对于 AR 模型, 当前时刻的观测值由过去几个历史时刻的观测值和一个当前时刻的随机干扰来表示. 设时间序列为  $\dots, X_{-1}, X_0, X_1, \dots$  或  $\{X_t, t = \dots, -1, 0, 1, \dots\}$ . 若  $X_t$  是平稳时间序列, 则  $X_t$  可以下式所示的自回归

模型表示:

$$X_t = \sum_{i=1}^p a'_i X_{t-i} + \varepsilon(t), \quad (1)$$

其中  $\varepsilon(t)$  为 0 均值平稳白噪声过程,  $p$  称作自回归模型的阶次.  $a'_i, i=1, 2, \dots, p$  为模型的参数, 且  $a'_p \neq 0$ . 称(1)式为  $p$  阶自回归模型, 记作  $AR(p)$ .

## 1.2 自回归模型的定阶

在平稳时间序列 AR 模型拟合的定阶分析中, 目前应用得最广泛的是 Akaike 信息准则(AIC):

$$\begin{cases} AIC(p) = \log \frac{RSS(p)}{N} + \frac{2p}{N}, \\ p_0 = \arg \min_p \{AIC(p)\}, \end{cases} \quad (2)$$

其中,  $N$  为观察数,  $RSS(p)$  为采用  $p$  阶 AR 模型的残差平方和,  $p$  属于自然数集合.  $p_0$  为最优拟合阶次. 增加 AR 模型的阶次可以提高模型拟合的优良性, AIC 鼓励数据拟合的优良性, 但是模型的阶次过高会出现过度拟合(Overfitting)的情况. 所以优先考虑的模型应是 AIC 值最小的那一个, 即选取使得  $AIC(p)$  达到极小值的正整数  $p_0$  为最优的拟合阶次. 通过使用 AIC 准则的方法可以寻找最好地解释数据但包含最少自由参数的模型.

## 1.3 自回归模型的参数估计

自回归模型的参数估计是时间序列分析中的基本问题. 基于不同情况, 很多自回归模型参数估计方法被提了出来, 如: 胡峰<sup>[32]</sup>给出了自回归模型系数与误差方差的递阶估计算法; 林正华等人<sup>[33]</sup>利用约束规划及计算数学理论, 讨论自回归模型参数的精确估计方法; 孟昭为<sup>[34]</sup>研究了多维向量自回归模型参数的极大似然估计与似然比检验. 尽管各种自回归模型参数估计方法被提了出来, 但最有效, 应用最广泛的方法为最小二乘法<sup>[35, 36]</sup>. 此处只考虑自回归模型参数的最小二乘估计方法.

对于平稳时间序列  $\{X_t, t = \dots, -1, 0, 1, \dots\}$  的一次观察得到一个长度为  $N$  的样本, 记作  $\{x_1, x_1, \dots, x_N\}$ . 对于(1)式给定的  $p$  阶自回归模型, 模型参数的最小二乘估计方法可以概括为求下式确定的无约束优化

问题.

$$\min_{a'_i, i=1, 2, \dots, p} \left[ \sum_{t=p+1}^N \left( X_t - \sum_{i=1}^p a'_i X_{t-i} \right)^2 \right]. \quad (3)$$

$$\text{记 } \tilde{x} = (x_{p+1}, \dots, x_N)^T, \quad \tilde{X} = \begin{pmatrix} x_p, x_{p+1}, \dots, x_{N-1} \\ x_{p-1}, x_p, \dots, x_{N-2} \\ \vdots \\ x_1, x_2, \dots, x_{N-p} \end{pmatrix},$$

采用多元函数求极值的办法可以推导得参数  $a'_i, i=1, 2, \dots, p$  的估计满足:

$$(a'_1, a'_2, \dots, a'_p)^T = (\tilde{X}\tilde{X}^T)^{-1} \tilde{X}\tilde{x}, \quad (4)$$

其中  $(a'_1, a'_2, \dots, a'_p)^T$  表示  $(a'_1, a'_2, \dots, a'_p)$  的转置.

## 2 Boosting 回归算法

Boosting 算法是基于统计学学习理论的一种重要的机器学习方法, 近年来在模式识别, 回归分析, 特征提取等方面得到了广泛的应用. 其基本思想来源于 PAC(Probably Approximately Correct)学习模型<sup>[37]</sup>. 在针对分类问题的 PAC 学习模型中, 若存在一个多项式时间内可以完成的学习算法, 并且该算法的识别正确率很高, 则该学习算法被称为强学习算法; 而如果学习算法识别一组概念时识别正确率仅比随机猜测略高, 则称这种学习算法为弱学习算法. Kearns 和 Valiant<sup>[38]</sup>首次提出并证明了 PAC 学习模型中弱学习算法和强学习算法具有等价性, 即当训练样本充分时, 能够将仅比随机猜测略好的弱学习算法提升为准确率足够高的强学习算法, 而不必直接去寻找通常情况下很难获得的强学习算法. 随后一些构造性的方法对该问题做了肯定的证明<sup>[39, 40]</sup>, 这就是最初的 Boosting 算法. Freund 和 Schapire<sup>[41]</sup>改进了 Boosting 算法, 提出了 AdaBoost (Adaptive Boosting) 算法, 该算法突破了早期 Boosting 算法的一些限制, 逐渐成为机器学习领域最为重要和常见的算法之一.

随着对 Boosting 算法研究的深入, 对 Boosting 分类问题的研究取得了长足的发展, 其中, AdaBoost 算法更是在理论分析和实际应用中都获得了巨大的成功. 因此, 一些研究者试图将分类方法引入到回归问题中. 在分类问题中训练样本被映射到离散空间, 因而应用到分类问题的 AdaBoost 算法仅需要通过改

变训练样本的权重而不需要改变训练样本本身就可以得到有效的组合算法. 相比较分类问题而言, 回归问题则是将样本映射到连续空间的情况, 因而回归问题更为复杂, 仅仅通过修改训练样本的权重而产生的基础学习器并不一定能够集成为一个有效的组合学习器. 因此在研究上也没有分类问题深入. 目前将 AdaBoost 算法应用到回归问题上大体有以下 3 种办法: 1) 通过改变样本权重的方式将 AdaBoost 算法应用到回归问题中, 如 AdaBoost.R 算法<sup>[41]</sup>; 2) 通过改变样本本身的方式将 AdaBoost 算法应用到回归问题中<sup>[42]</sup>; 3) 通过同时改变样本本身和样本权重的方式将 AdaBoost 算法应用到回归问题中.

本文通过改变样本本身的方式将 AdaBoost 算法应用到回归问题中, 基于 Boosting 梯度下降理论, 提出了自回归模型集成建模法(AREL)用于对时间序列分析、建模并进行预测.

## 2.1 Boosting 梯度下降理论

对于函数估计问题, 给定一个在  $(X, Y)$  空间中的训练样本集  $(x_i, y_i)_{i=1}^N$ , 目标是寻找一个从  $X$  到  $Y$  的映射  $F^*(X)$ , 使得对于所有的  $(X, Y)$ , 期望损失  $L(Y, F(X))$  最小, 即

$$F^*(X) = \arg \min_{F(X)} E_{x,y} L(Y, F(X)). \quad (5)$$

在 Boosting 回归算法中, 通常的做法是将  $F(X)$  表示为一个由基础函数组合而成的组合函数, 即  $F(X)$  可以表示为

$$F(X) = \sum_{t'} \alpha_{t'} h_{t'}(X, \beta_{t'}), \quad (6)$$

其中  $t' = 1, 2, \dots$  为基函数的个数,  $h_{t'}(X, \beta_{t'}) \in H$ ,  $H$  为一个确定的函数空间,  $\beta_{t'}$  为基函数的参数,  $\alpha_{t'}$  为基函数的组合系数.  $h_{t'}(X, \beta_{t'})$  通过不断迭代产生. 在 Boosting 回归算法中,  $h_{t'}(X, \beta_{t'})$  可以为不同类型的函数.

通过将  $F(X)$  作为目标函数可以将函数估计最优化问题变为一个在函数空间中求基函数的最优组合问题. 梯度法是最常用, 也是最简单的逼近极值点的数值计算方法. 在给定的训练样本  $(x_i, y_i)_{i=1}^N$ , 应用梯度法在函数空间  $H$  中求一个基函数的最优组合使

得期望损失  $L(Y, F(X))$  最小的过程为

1) 令  $F_{t'}(x_i) = \sum_{t'=0}^{t'} \alpha_{t'} h_{t'}(x_i)$ ,  $i = 1, 2, \dots, N$ , 其中  $t'$  可以为  $\{0, 1, 2, \dots, T'\}$  中任意一个值,  $T'$  为最大迭代次数, 当  $t'_1 = 0$  时,  $F_0(x_i) = \alpha_0 h_0(x_i)$  为初值. 此时损失函数对  $F(x_i)$ ,  $i = 1, 2, \dots, N$ , 的负梯度  $g_{t'_1}(x_i)$  为

$$g_{t'_1}(x_i) = - \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x_i)=F_{t'_1}(x_i)}. \quad (7)$$

2) 令  $\bar{g}_{t'_1} = (g_{t'_1}(x_1), g_{t'_1}(x_2), \dots, g_{t'_1}(x_N))$ ,  $\bar{h} = (h(x_1), h(x_2), \dots, h(x_N))$ , 其中  $h \in H$ . 在函数空间  $H$  中求中寻找一个  $h_{t'_1+1}$ , 使得  $\bar{h}_{t'_1+1}$  最接近平行于  $\bar{g}_{t'_1}$  的方向, 即

$$h_{t'_1+1} = \arg \max_{h \in H} \frac{\langle \bar{g}_{t'_1}, \bar{h} \rangle}{\|\bar{g}_{t'_1}\|_2 \cdot \|\bar{h}\|_2}, \quad (8)$$

其中  $\langle \bar{g}_{t'_1}, \bar{h} \rangle$  表示两个向量的内积.

3) 通过下式计算系数  $\alpha_{t'_1+1}$ .

$$\alpha_{t'_1+1} = \arg \min_{a \in R} \sum_{i=1}^N L(y_i, F_{t'_1}(x_i) + ah_{t'_1+1}(x_i)). \quad (9)$$

## 2.2 基于 Boosting 梯度下降理论的 AREL

为了描述方便, 此处定义一个时间序列向量化算子  $\nabla_i^p(W)$ , 通过时间序列向量化算子  $\nabla_i^p(W)$  将时间序列转变为向量化样本.

**定义(时间序列向量化).** 设时间序列的一次观察得到一个长度为  $N$  的样本  $W : \{x_1, x_1, \dots, x_N\}$ . 时间序列向量化算子  $\nabla_i^p(W)$  首先在时间序列  $\{x_1, x_1, \dots, x_N\}$  上定义一个长度为  $p+1$  的窗口, 其中  $x_i$  ( $i = 1, 2, \dots, N$ ) 为窗口内最后一个元素, 当  $1 \leq i < p+1$ , 在窗口内  $x_i$  的前面以 0 填充. 将窗口内前  $p$  个值作为新定义样本的属性值, 将窗口内最后一个值(第  $p+1$  个值)作为新定义样本的 label. 这样将一个窗口内的时间序列样本定义为一个带有 label 的, 属性个数为  $p$  的新样本. 通过滑动窗口(步长为 1, 即每次向后滑动一个观察值)可以将时间序列  $W : \{x_1, x_1, \dots, x_N\}$  定义成为  $N$

个新样本. 文中用  $\nabla_i^p(W)$  表示新定义样本的属性 ( $\nabla_i^p(W)$  表示为行向量), 其中  $p$  表示新样本属性个数,  $i$  表示新样本的 label 值在  $W: \{x_1, x_1, \dots, x_N\}$  中的序号.

在定义了时间序列向量化算子  $\nabla_i^p(W)$  之后, 文中基于 Boosting 梯度下降理论的 AREL 选用残差平方和函数作为损失函数, 且  $H$  为一个线性函数空间, 则基于 Boosting 梯度下降理论的 AREL 被描述如下.

**Gradient-boost AREL**

**输入:**  $W_0 = \{x_1, x_1, \dots, x_N\}$ ,  $F_0 = 0$ , 迭代次数  $T'$ ;

**For**  $t' = 1$  to  $T'$

1) 如果  $t' = 1$ , 置  $W_1(i) = W_0(i)$ , 否则置  $W_{t'}(i) = W_{t'-1}(i) - a_{t'-1} h_{t'-1}(\nabla_i^{p_{t'-1}}(W_{t'-1}))$ , 其中  $i = 1, 2, \dots, N$ . 以  $\{W_{t'}(i)\}_{i=1}^N$  作为新定义的时间序列.

2) 基于  $\{W_{t'}(i)\}_{i=1}^N$ , 定义  $p_{t'} = \arg \min_p (AIC(p))$ , 其中  $AIC(p) = \log \frac{RSS(p)}{N} + \frac{2p}{N}$ ,  $RSS$  为残差平方和,  $p$  属于自然数集合.

3) 令  $r_{t'} = (W_{t'}(1), W_{t'}(2), \dots, W_{t'}(N))$ ,  $\bar{h} = (W_{t'}(1), \dots, W_{t'}(p_{t'}), h(\nabla_{p_{t'}+1}^{p_{t'}}(W_{t'})), \dots, h(\nabla_N^{p_{t'}}(W_{t'})))$ , 由下式计算  $h_{t'}$ :

$$h_{t'} = \arg \max_{h \in H} \frac{\langle r_{t'}, \bar{h} \rangle}{\|r_{t'}\|_2 \cdot \|\bar{h}\|_2}. \tag{10}$$

4) 计算系数  $\alpha_{t'}$ , 其中  $\bar{h}_{t'} = (W_{t'}(1), \dots, W_{t'}(p_{t'}), h_{t'}(\nabla_{p_{t'}+1}^{p_{t'}}(W_{t'})), \dots, h_{t'}(\nabla_N^{p_{t'}}(W_{t'})))$ .

$$\alpha_{t'} = \frac{\langle r_{t'}, \bar{h}_{t'} \rangle}{\|\bar{h}_{t'}\|_2^2}. \tag{11}$$

5) 当  $i < p_{t'} + 1$ , 置  $a_{t'} h_{t'}(\nabla_i^{p_{t'}}(W_{t'})) = r_{t'}(i)$ . 更新  $F_{t'-1}$  为  $F_{t'}$ .

$$F_{t'} = F_{t'-1} + a_{t'} h_{t'}(\nabla_i^{p_{t'}}(W_{t'})). \tag{12}$$

**End**

**输出最终预测模型:**

$$F_{T'} = \sum_{t'=1}^{T'} a_{t'} h_{t'}(\nabla_i^{p_{t'}}(W_{t'})).$$

**定理 1.** 在 Gradient-boost AREL 中

(i) (10)式及(11)式确定的弱学习器  $h_{t'}$  及其步长  $\alpha_{t'}$  可以在算法的每次迭代中最大程度地降低损失.

(ii) 因为  $H$  为一个线性函数空间, 则第  $t'(1 \leq t' \leq T')$  次迭代学习得到的弱学习器  $a_{t'} h_{t'}(\nabla_i^{p_{t'}}(W_{t'}))$  为一个线性函数, 则该函数可表示为  $(a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}})(\nabla_i^{p_{t'}}(W_{t'}))^T$ , 其中  $a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}}$  可以在时间序列  $\{W_{t'}(i)\}_{i=1}^N$  上通过 (4) 式计算.  $(\nabla_i^{p_{t'}}(W_{t'}))^T$  表示对向量  $\nabla_i^{p_{t'}}(W_{t'})$  求转置.

**证明.** (i) 经过  $t'-1$  次迭代后损失函数为  $L_{t'-1} = \sum_{i=1}^N \left( W_0(i) - \sum_{j=1}^{t'-1} a_j h_j(\nabla_i^{p_j}(W_j)) \right)^2$ , 则此时损失函数对  $F(x_i)$ ,  $i = 1, 2, \dots, N$ , 的负梯度为

$$\begin{aligned} & - \frac{\partial L}{\partial F(x_i)} \Big|_{F(x_i) = \sum_{j=1}^{t'-1} a_j h_j(\nabla_i^{p_j}(W_j))} \\ & = 2 \left( W_0(i) - \sum_{j=1}^{t'-1} a_j h_j(\nabla_i^{p_j}(W_j)) \right). \end{aligned} \tag{13}$$

经过  $t'-1$  次迭代后, 由 Gradient-boost AREL 可知损失函数在当前点的负梯度满足

$$2 \left( W_0(i) - \sum_{j=1}^{t'-1} a_j h_j(\nabla_i^{p_j}(W_j)) \right) = 2r_{t'}(i), \tag{14}$$

其中,  $i = 1, 2, \dots, N$ . 第  $t'$  次迭代中, 如果  $h_{t'}$  满足

$$\frac{\bar{h}_{t'}}{\|\bar{h}_{t'}\|_2} = \frac{r_{t'}}{\|r_{t'}\|_2}, \tag{15}$$

则  $h_{t'}$  为当前点损失函数的最速下降方向. 满足(15)式的  $h_{t'}$  不一定属于  $H$ , 因此第  $t'$  次迭代中,  $h_{t'}$  应从  $H$  中挑选一个与  $r_{t'}$  最接近于平行的方向, 即

$$h_{t'} = \arg \max_{h \in H} \frac{\langle r_{t'}, \bar{h} \rangle}{\|r_{t'}\|_2 \cdot \|\bar{h}\|_2}, \tag{16}$$

满足(16)式的  $h_{t'}$  为当前损失函数的最速下降方向.

在  $h_{t'}$  确定的条件下, 损失函数变为

$$L_{t'}(i) = \left( W_0(i) - \sum_{j=1}^{t'-1} a_j h_j \left( \nabla_i^{p_j} (W_j) \right) - a_{t'} \bar{h}_{t'} \left( \nabla_i^{p_{t'}} (W_{t'}) \right) \right)^2,$$

由 Gradient-boost AREL 可知  $r_{t'}(i) = W_0(i) - \sum_{j=1}^{t'-1} a_j h_j \left( \nabla_i^{p_j} (W_j) \right)$ , 则步长  $\alpha_{t'}$  可以由下式得到

$$\alpha_{t'} = \arg \min_a (r_{t'} - a_{t'} \bar{h}_{t'}) \cdot (r_{t'} - a_{t'} \bar{h}_{t'})^T, \quad (17)$$

其中  $(r_{t'} - a_{t'} \bar{h}_{t'})^T$  表示对向量求转置,  $r_{t'}$ ,  $\bar{h}_{t'}$  在 Gradient-boost AREL 算法中所定义. 式  $\arg \min_a (r_{t'} - a_{t'} \bar{h}_{t'}) \cdot (r_{t'} - a_{t'} \bar{h}_{t'})^T$  的最优解为使得损失函数沿  $h_{t'}$  方向下降的最优步长. 因为

$$\begin{aligned} \frac{\partial (r_{t'} - a_{t'} \bar{h}_{t'}) \cdot (r_{t'} - a_{t'} \bar{h}_{t'})^T}{\partial a_{t'}} &= 0 \\ \Rightarrow a_{t'} \bar{h}_{t'} \cdot (\bar{h}_{t'})^T - r_{t'} \cdot (\bar{h}_{t'})^T & \\ = 0 \Rightarrow a_{t'} &= \frac{\langle r_{t'}, \bar{h}_{t'} \rangle}{\|\bar{h}_{t'}\|_2^2}, \end{aligned} \quad (18)$$

则  $a_{t'} = \frac{\langle r_{t'}, \bar{h}_{t'} \rangle}{\|\bar{h}_{t'}\|_2^2}$  为  $\arg \min_a (r_{t'} - a_{t'} \bar{h}_{t'}) \cdot (r_{t'} - a_{t'} \bar{h}_{t'})^T$  的最优解.

(ii) 当第  $t'$  ( $1 \leq t' \leq T'$ ) 次迭代学习到的弱学习器表示为:  $(a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}}) \left( \nabla_i^{p_{t'}} (W_{t'}) \right)^T$ , 则上述算法中第  $t'$  次迭代的损失函数可以表示为

$$\begin{aligned} &(r_{t'} - a_{t'} \bar{h}_{t'}) \cdot (r_{t'} - a_{t'} \bar{h}_{t'})^T \\ &= \sum_{i=1}^N \left( r_{t'}(i) - (a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}}) \left( \nabla_i^{p_{t'}} (W_{t'}) \right)^T \right)^2. \end{aligned} \quad (19)$$

当第  $t'$  次迭代得到的弱学习器  $a_{t'} h_{t'} \left( \nabla_i^{p_{t'}} (W_{t'}) \right)$  表示为  $(a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}}) \left( \nabla_i^{p_{t'}} (W_{t'}) \right)^T$ , 由上述 Gradient-boost AREL 算法可知: 当  $i < p_{t'} + 1$ ,  $(a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}}) \left( \nabla_i^{p_{t'}} (W_{t'}) \right)^T = r_{t'}(i)$ . 此时(19)式可变为

$$\begin{aligned} &(r_{t'} - a_{t'} \bar{h}_{t'}) \cdot (r_{t'} - a_{t'} \bar{h}_{t'})^T \\ &= \sum_{i=p_{t'}+1}^N \left( r_{t'}(i) - (a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}}) \left( \nabla_i^{p_{t'}} (W_{t'}) \right)^T \right)^2 \\ &= \sum_{i=p_{t'}+1}^N \left( r_{t'}(i) - \sum_{j=1}^{p_{t'}} a'_{t',j} r_{t'}(i-j) \right)^2 \\ &= \sum_{i=p_{t'}+1}^N \left( r_{t'}(i) - \sum_{j=1}^{p_{t'}} a'_{t',j} r_{t'}(i-j) \right)^2, \end{aligned} \quad (20)$$

此时模型参数的最小二乘估计方法为求解式  $\min_{a'_{t',j}, j=1,2,\dots,p_{t'}} \sum_{i=p_{t'}+1}^N \left( r_{t'}(i) - \sum_{j=1}^{p_{t'}} a'_{t',j} r_{t'}(i-j) \right)^2$  确定的无约束优化问题. 分别对  $a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}}$  求偏导数, 并置偏导数等于零, 可得

$$\begin{cases} \frac{\partial \sum_{i=p_{t'}+1}^N \left( r_{t'}(i) - \sum_{j=1}^{p_{t'}} a'_{t',j} r_{t'}(i-j) \right)^2}{\partial a'_{t',1}} = 0, \\ \vdots \\ \frac{\partial \sum_{i=p_{t'}+1}^N \left( r_{t'}(i) - \sum_{j=1}^{p_{t'}} a'_{t',j} r_{t'}(i-j) \right)^2}{\partial a'_{t',p_{t'}}} = 0. \end{cases} \quad (21)$$

通过求解式(21)即可得到  $a'_{t',1}, a'_{t',2}, \dots, a'_{t',p_{t'}}$  的取值. 由 1.3 节给出的自回归模型参数的最小二乘估计方法可以得到  $a'_{t',1}, \dots, a'_{t',p_{t'}}$  的解析表达形式, 即(4)式所示.

### 2.3 基于 Boosting 梯度下降理论的 AREL 特性分析

**定理 2.** 对于上面给出的 Gradient-boost AREL 中, 第  $t'$  ( $1 \leq t' \leq T'$ ) 次迭代后的损失为  $L_{t'} = \sum_{i=1}^N \left( W_0(i) - \sum_{j=1}^{t'} a_j h_j \left( \nabla_i^{p_j} (W_j) \right) \right)^2$ . 定义  $\xi_{t'} = \frac{\langle r_{t'}, \bar{h}_{t'} \rangle}{\|r_{t'}\|_2 \cdot \|\bar{h}_{t'}\|_2}$ , 其中  $r_{t'}$ ,  $\bar{h}_{t'}$  在 Gradient-boost AREL 中所定义, 则第  $t'$  次迭代后损失函数  $L_{t'}$  为上次迭代后损失  $L_{t'-1}$  的  $1 - \xi_{t'}^2$  倍.

**证明.** 令  $\bar{W}_0 = (W_0(1), \dots, W_0(N))$ ,  $\bar{F}_{t'} =$

$\left( \sum_{j=1}^{t'} a_j h_j \left( \nabla_1^{p_j} (W_j) \right), \dots, \sum_{j=1}^{t'} a_j h_j \left( \nabla_N^{p_j} (W_j) \right) \right)$ , 则  $L_{t'}$  可以表示为

$$L_{t'} = (\bar{W}_0 - \bar{F}_{t'}) \cdot (\bar{W}_0 - \bar{F}_{t'})^T, \quad (22)$$

其中  $(\bar{W}_0 - \bar{F}_{t'})^T$  表示向量  $(\bar{W}_0 - \bar{F}_{t'})$  的转置.

因为,  $L_{t'} = \sum_{i=1}^N \left( W_0(i) - \sum_{j=1}^{t'} a_j h_j \left( \nabla_i^{p_j} (W_j) \right) \right)^2$ , 且

$a_{t'} = \frac{\langle r_{t'}, \bar{h}_{t'} \rangle}{\|\bar{h}_{t'}\|_2^2}$ , 则:

$$\begin{aligned} L_{t'} &= (\bar{W}_0 - \bar{F}_{t'}) \cdot (\bar{W}_0 - \bar{F}_{t'})^T \\ &= L_{t'-1} - 2a_{t'} (\bar{W}_0 - \bar{F}_{t'-1}) \cdot (\bar{h}_{t'})^T + a_{t'}^2 (\bar{h}_{t'}) \cdot (\bar{h}_{t'})^T \\ &= L_{t'-1} - \frac{\left( (\bar{W}_0 - \bar{F}_{t'-1}) \cdot (\bar{h}_{t'})^T \right)^2}{\|\bar{h}_{t'}\|_2^2} \\ &= L_{t'-1} \cdot (1 - \xi_{t'}^2). \end{aligned} \quad (23)$$

**定理 3.** 设  $W_0(i) \in \left[ -\frac{B}{2}, \frac{B}{2} \right]$ , 如果第  $t'$  次迭代后损失函数  $L_{t'}$  至少为上次迭代后损失  $L_{t'-1}$  的  $1 - \xi_{\min}^2$

倍, 其中  $\xi_{\min} = \min_{1 \leq t' \leq T'} \xi_{t'}$ , 且  $\xi_{t'} = \frac{\langle r, \bar{h}_{t'} \rangle}{\|r\|_2 \cdot \|\bar{h}_{t'}\|_2}$ . 则对于

任意  $\rho > 0$ , Gradient-boost AREL 在  $T' = \frac{\ln\left(\frac{4\rho}{B^2}\right)}{\ln(1 - \xi_{\min}^2)}$

次迭代后, 组合函数  $F_{T'}$  的损失函数满足:  $L_{T'} \leq N\rho$ .

**证明.** 因为  $L_0 = \sum_{i=1}^N (W_0(i) - F_0)^2$ , 且  $-\frac{B}{2} \leq W_0(i) \leq \frac{B}{2}$ . 由 Gradient-boost AREL 的初始化知  $F_0 = 0$ , 则:

$$L_0 = \sum_{i=1}^N (W_0(i) - F_0)^2 = \sum_{i=1}^N (W_0(i))^2 \leq N \frac{B^2}{4}. \quad (24)$$

因为第  $t'$  次迭代后损失函数  $L_{t'}$  至少为上次迭代后损失  $L_{t'-1}$  的  $1 - \xi_{\min}^2$  倍, 则经过  $T'$  次迭代后, 有

$$L_{T'} = L_0 \cdot (1 - \xi_{\min}^2)^{T'} \leq \frac{NB^2 \cdot (1 - \xi_{\min}^2)^{T'}}{4}. \quad (25)$$

如果经过  $T'$  次迭代后损失函数  $L_{T'} \leq N\rho$ , 则有

$$\frac{NB^2 \cdot (1 - \xi_{\min}^2)^{T'}}{4} \leq N\rho \Rightarrow T' \ln(1 - \xi_{\min}^2) \leq \ln\left(\frac{4\rho}{B^2}\right) \Rightarrow T' \leq \frac{\ln\left(\frac{4\rho}{B^2}\right)}{\ln(1 - \xi_{\min}^2)}, \quad (26)$$

即 Gradient-boost AREL 在  $T' = \frac{\ln\left(\frac{4\rho}{B^2}\right)}{\ln(1 - \xi_{\min}^2)}$  次迭代后,

组合函数  $F_{T'}(W_0)$  的损失函数  $L_{T'} \leq N\rho$ .

## 2.4 使用 $\varepsilon$ -不敏感损失函数的 AREL

通过上节分析, 可以看出 AREL 是一种有效地提升 AR 模型的学习算法, 且定理 2 表示, 随着迭代次数的增加, 模型的训练误差指数级单调下降, 这表明 AREL 是一种高效的学习算法, 但同时也表明 AREL 是一种贪婪学习算法, 因而随着迭代次数的增加, AREL 容易发生过度拟合. 如果训练样本中包含异常样本, 那么如何提高 AREL 的鲁棒性, 防止过拟合是 AREL 的一个关键问题.

任何从训练样本中学习并构建模型的学习算法, 都无法回避噪声样本的问题. 如针对著名的 AdaBoost 算法, soft margin AdaBoost, regularization Boosting, weighting decay method, BrownBoost 等众多的防止过拟合的方法被提了出来<sup>[43]</sup>. 对 SVM 算法也有很多新颖的防止过拟合手段<sup>[44]</sup>. 对时间序列上的任意一个时刻, 该时刻的值实际上是对服从某种分布的随机变量的一次观察, 因而时间序列上任意一个时刻的观察值都包含一定的随机因素. AREL 精确拟合每一个观察值, 增加了 AREL 对随机因素的拟合程度, 从而降低了 AREL 的泛化能力.

Vapnik 在定义了  $\varepsilon$ -不敏感损失函数的基础上提出了  $\varepsilon$  支持向量回归算法. 其中参数  $\varepsilon$  表示系统对回归函数在样本点上的误差要求.  $\varepsilon$  越小, 回归函数在该样本点上的误差要求也越小, 函数估计的经验风险也就越小. 本文受该思路的启发, 在分析了时间序列随机性特征的基础上提出了使用  $\varepsilon$ -不敏感损失函

数的 AREL 算法.

在 3.2 节中给出的 Gradient-boost AREL 循环的第一步中定义:

$$W_{t'}'(i) = 0, \text{ if } \left| W_0(i) - \sum_{j=1}^{t'-1} a_j h_j \left( \nabla_i^{p_j} (W_j) \right) \right| \leq \varepsilon, \quad (27a)$$

$$W_{t'}'(i) = W_0(i) - \sum_{j=1}^{t'-1} a_j h_j \left( \nabla_i^{p_j} (W_j) \right),$$

$$\text{if } \left| W_0(i) - \sum_{j=1}^{t'-1} a_j h_j \left( \nabla_i^{p_j} (W_j) \right) \right| > \varepsilon, \quad (27b)$$

以  $\{W_{t'}'(i)\}_{i=1}^N$  代替  $\{W_{t'}(i)\}_{i=1}^N$  进入下一步迭代.

在本文算法中通过引入  $\varepsilon$ -不敏感损失函, 避免了 AREL 精确拟合每一个训练样本点, 从而降低了随机因素对预测模型的影响, 提高 AREL 的泛化能力, 降低过度拟合的风险.

### 2.5 基于 AREL 的 weighted kNN 预报方法

在前几节详细介绍了 AREL, 并分析了 AREL 的特点, 指出 AREL 是一种高效的学习算法. 在 2.4 节进一步分析了时间序列的特点, 提出了使用  $\varepsilon$ -不敏感损失函数的 AREL. 虽然这些分析和第三节的实例测试与结果分析都验证了 AREL 和使用  $\varepsilon$ -不敏感损失函数的 AREL 可以有效地处理时间序列, 并且提升单个 AR 模型的性能, 但是一个不可忽略的事实是: 尽管引入  $\varepsilon$ -不敏感损失函数可以降低 AREL 发生过拟合的风险, 但是引入  $\varepsilon$ -不敏感损失函数也带来一个问题, 那就是在 AREL 的每次迭代中需要的 AR 模型阶次较大, 否则随着迭代的进行 AREL 的损失函数并不下降. 例如: 当 AREL 经过  $t'-1$  次迭代后对原始时间序列的  $N$  个训练样本拟合效果非常好, 即经过  $t'-1$  次迭代后, 在很多训练样本上满足误差要求, 亦即

$$\left| W_0(i) - \sum_{j=1}^{t'-1} a_j h_j \left( \nabla_i^{p_j} (W_j) \right) \right| \leq \varepsilon. \text{ 此时根据(27)式计算}$$

得到下一次迭代的时间序列中很多样本取值为零, 只有极少数的非零点. 如果在该次迭代中 AR 模型阶次过小, 则根据(4)式计算 AR 模型的参数时, 得到的模型参数接近于零, 根本无法形成对训练样本的有效拟合, 此时 AREL 继续迭代不会降低损失函数. 而如果 AR 模型阶次过大, 则根据(4)式计算 AR 模型的

参数时, 矩阵  $\tilde{X}\tilde{X}^T$  有时会出现奇异, 这为模型的参数的计算带来困难, 同时模型阶次较大也增加了过度拟合的风险. 因而在 AREL 中, 模型阶次的选择对模型的预测效果影响比较大, 即 AREL 对模型阶次的设置比较敏感. 针对不同问题的时间序列数据在 AREL 中需要确定模型的阶次范围, 这样得到的模型不仅自适应性较差, 而且对模型阶次范围的估计也增加了工作量.

事实上 AR 模型的模型参数凝聚了系统状态的重要信息, 准确的 AR 模型能够深刻、集中地表达动态系统的客观规律. 定理 2 与定理 3 显示, 经过有限次迭代后, AREL 学习得到的组合模型对原始时间序列的  $N$  个训练样本具有优良的拟合效果, 因而该组合模型的参数很好地表达动态系统的客观规律. 基于该结论, 本文首先利用 AREL 学习得到的模型参数作为权重, 选择与待预测点附近具有相同或相似状态变化规律的  $k$  个时间序列片段, 以这  $k$  个时间序列片段对待预测点作出预测. 具体算法描述如下.

#### 基于 AREL 的 weighted kNN 预报方法.

**输入:** 待预测时间序列的长度  $J$ , 近邻个数  $k$ ;

**输入:**  $W_0 = \{x_1, x_1, \dots, x_N\}$ , 置  $F_0 = 0$ . 迭代次数  $T'$ ; 令第  $j(1 \leq j \leq J)$  个预测值的输出表示为  $\tilde{W}(j)$ , 置  $\tilde{W}(j) = 0, 1 \leq j \leq J$ .

**调用:** 使用  $\varepsilon$ -不敏感损失函数的 AREL 生成  $T'$  个  $\{W_{t'}'(i)\}_{i=1}^N$ , 与  $T'$  个  $AR_{t'}(p_{t'})$  自回归模型.

**For**  $j=1$  to  $J$

**For**  $t'=1$  to  $T'$

1) 以  $AR_{t'}(p_{t'})$  自回归系数作为权重, 以加权的欧拉距离为标准选择  $\nabla_{N+j}^{p_{t'}}(W_{t'}')$  的  $k$  个近邻. 其中加权的欧拉距离定义为(29)式.

2) 不妨设  $\nabla_{N+j}^{p_{t'}}(W_{t'}')$  的  $k$  个近邻近邻为  $\nabla_{l_1}^{p_{t'}}(W_{t'}'), \nabla_{l_2}^{p_{t'}}(W_{t'}'), \dots, \nabla_{l_k}^{p_{t'}}(W_{t'}')$ . 置:

$$\tilde{W}(j) = \tilde{W}(j) + \frac{W_{t'}'(l_1) + \dots + W_{t'}'(l_k)}{k}. \quad (28)$$

3) 置  $W_{t'}' = \left\{ \{W_{t'}'(i)\}_{i=1}^{N+j-1}, \frac{W_{t'}'(l_1) + \dots + W_{t'}'(l_k)}{k} \right\}$ .

**End**



End

上述算法中加权的欧拉距离定义为

$$d(\nabla_{N+i}^{P'}(W_{i'}'), \nabla_{N+j}^{P'}(W_{j'}')) = \sqrt{\left( \nabla_{N+i}^{P'}(W_{i'}') M M^T (\nabla_{N+j}^{P'}(W_{j'}'))^T \right)}, \quad (29)$$

其中  $M$  为一个对角矩阵, 对角线元素  $M_{ii}$  表示对第  $i$  个属性所加的权重.

这种算法具有两个非常重要的特点.

1) 通过 AREL, 学习并得到了一个组合的 AR 模型, 以该模型的参数作为权重选择与待预测点附近具有相同变化规律的时间序列片段. 其中模型的阶次决定了近邻法中选择近邻的时间序列片段的长度. 因为对整个时间序列在时间轴上进行了重新的组织, 通过选择与待预测点具有相同变化规律的时间片段对待预测点进行预测, 所以该算法具有较高的预测精度, 并且适用于非平稳时间序列.

2) 基于 AREL 的 weighted kNN 预报方法, 通过选择与待预测点附近具有相同变化规律的时间序列片段, 这学习并注意到了时间序列的局部变化性, 而这些时间片段通过投票的方式对待预测点进行预报, 可以降低噪声样本的影响, 从而提高了这种预报方法的鲁棒性.

### 3 算例及其结果分析

为了对本文所提出的算法进行验证, 在本节的第 3.1 和 3.2 小节, 采用西北电网陕西省用电负荷数据为例, 对本文所提出的算法进行了实例测试. 在算例中使用了 8 d 的用电负荷数据共 768 个数据点, 每 15 min 一个采样点. 其中前 6 d 的用电负荷数据用作训练数据, 然后使用本文提出的算法对后两天的用电负荷进行预测. 为了进一步验证基于 AREL 的 weighted kNN 预报法对非平稳时间序列的处理能力, 在 3.3 小节, 使用我国某特大型钢铁企业共 11 d 的实际用电数据对基于 AREL 的 weighted kNN 预报法进行了测试. 大型钢铁企业用电负荷数据为每 5 min 一个采样点, 算例中取其中前 10 d 共 2880 个用电负荷数据点用来建立模型, 然后对第 11 d 的用电负荷进行预测. 本文算例中模型阶次在 10~100 范围内采用 Akaike 信息准则进行选择.

### 3.1 采用 $\varepsilon$ -不敏感损失函数的 AREL 对 AR 模型的提升作用

本算例中  $\varepsilon$  置为训练数据中最大用电负荷的 0.01%, 迭代次数选择为 50 次. 实验结果中损失函数定义为

$$L_{T'} = \sum_{i=p+1}^N \left( W_{\text{train}}(i) - \sum_{t'=1}^{T'} a_{t'} h_{t'}(\nabla_i^{P'}(W_{t'}')) \right)^2, \quad (30)$$

其中,  $W_{\text{train}}$  表示  $N$  个原始训练数据序列;  $F_{T'} = \sum_{t'=1}^{T'} a_{t'} h_{t'}(\nabla_i^{P'}(W_{t'}'))$  为 AREL 经过  $T'$  次迭代后得到的组合模型(其中 AREL 使用(27)式定义的  $\varepsilon$ -不敏感损失函数).  $p$  为  $t'(1 \leq t' \leq T')$  次迭代中回归模型最大的模型阶次, 之所以从第  $p+1$  个样本开始计算, 是因为算法中的参数采用近似最小二乘估计, 损失了  $p$  个样本信息.

随着迭代的进行 AREL 的损失  $L_{T'}$  变化如图 1 所示. 图 1 给出了 AREL 损失随着迭代的进行而变化的过程, 图 1 中显示 AREL 每一步迭代都在上一步的基础上减小损失, 从实例上验证了定理 2 给出的结论.

为了直观显示 AREL 的预测效果, 图 2 给出了 AREL 的预测结果与实际用电负荷的比较. 从图 2 中可以看出, 随着预测长度的增加, 算法的预测效果有所下降. 但总体而言对于长达 48 h 的用电负荷预测, AREL 的预测效果良好, 基本上展现出电力负荷的周期性趋势性的变化规律.

平均绝对百分误差(MAPE)是对模型进行检验而惯常使用的一个直接指标, 通常情况下可以使用平均绝对百分误差来表示模型的拟合优良程度, 也可

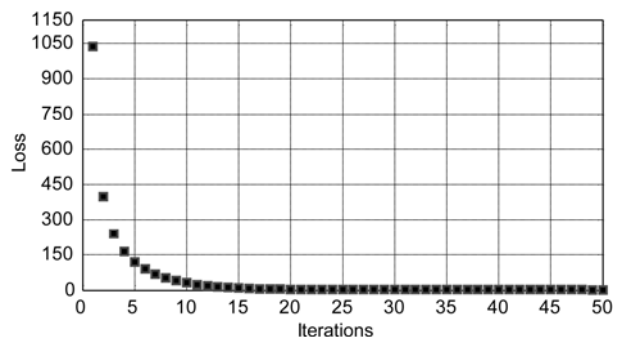


图 1 AREL 算法中  $L_{T'}$  变化的过程

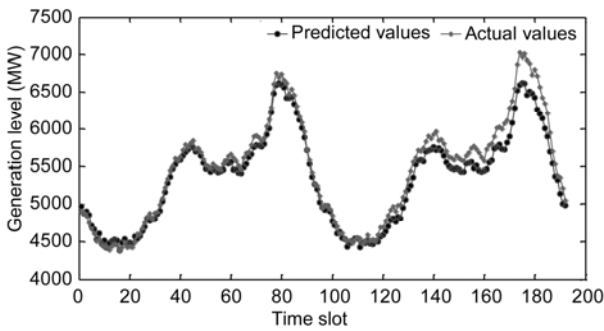


图2 AREL 的预测结果与实际用电负荷的比较

以使用平均绝对百分误差对模型预测结果的优劣进行判断. 如果用  $W_{\text{test}}(i)$  表示第  $i$  个数据点预测结果的真实数据,  $\sum_{t'=1}^{T'} a_{t'} h_{t'}(\nabla_{i}^{P_{t'}}(W_{t'}))$  为 AREL 得到的预测模型对第  $i$  个数据点预测的结果, 则对长度为  $N'$  的时间序列预测结果的平均绝对百分误差定义为

$$\text{MAPE}_i = \frac{100}{N'} \sum_{i=1}^{N'} \left| \frac{W_{\text{test}}(i) - \sum_{t'=1}^{T'} a_{t'} h_{t'}(\nabla_{i}^{P_{t'}}(W_{t'}))}{W_{\text{test}}(i)} \right| \quad (31)$$

此处使用平均绝对百分误差表示时间序列预测模型预测结果的优劣. 下图 3 给出了 AREL 预测结果的平均绝对百分误差随着迭代的进行而变化的过程.

图 3 显示, 随着迭代的进行,  $F_{T'}$  对未来的预测效果也进一步得到了提高, 通过迭代, AREL 对单个 AR 模型的预测效果的提升非常明显. 结合图 1 中显示的结果: 随着迭代的进行, AREL 对训练数据的损失  $L_{T'}$  逐渐减小, AREL 得到的组合模型  $F_{T'}$  对训练数据的拟合效果进一步提高, 可以说明通过 Boosting 技术可以对 AR 模型进行有效地提升.

### 3.2 基于 AREL 的 weighted kNN 预报法预测结果

通常预测不可能 100% 准确. 因此, 对预测系统的使用者而言, 误差来源及误差范围是与预测值同等重要的信息, 它可以使用者能够估计出信赖预测结果的风险<sup>[45, 46]</sup>. 同理预测系统或者预测算法对系统或者算法的参数设置具有较高的稳定性, 也可以使用者以较高的置信度信赖该系统或者算法. 尽管 3.1 节的测试结果表明, 采用 Boosting 技术可以提升 AR 模型的性能, 使用  $\epsilon$ -不敏感损失函数的

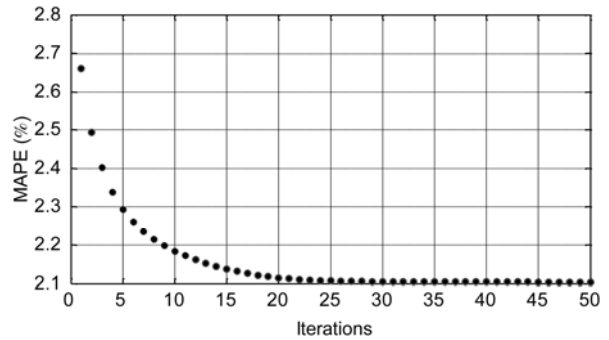


图3 AREL 预测结果平均绝对百分误差随着迭代的进行而变化的过程

AREL 是一种有效的时间序列分析方法. 但在测试中也发现, 当模型阶次选择不合适, 使用  $\epsilon$ -不敏感损失函数的 AREL 预测效果不太理想, AREL 对模型阶次的选择较为敏感. 为了降低 AREL 对模型阶次选择的敏感性, 本文设计了基于 AREL 的 weighted kNN 预报方法. 此处给出基于 AREL 的 weighted kNN 预报方法对实际数据的一些测试结果. 基于 AREL 的 weighted kNN 预报法中近邻个数  $k$  置为 2.

图 4 给出了基于 AREL 的 weighted kNN 预报法  $L_{T'}$  随着迭代的进行而发生变化的过程. 图 4 显示基于 AREL 的 weighted kNN 预报法每一步迭代都在上一步的基础上损失有所减小. 与图 1 显示结果比较, 图 4 显示基于 AREL 的 weighted kNN 预报法的  $L_{T'}$  随着迭代的进行下降速度相对较慢.

图 5 显示了基于 AREL 的 weighted kNN 预报法的预测结果与实际用电负荷的比较. 从图 5 中可以看出, 基于 AREL 的 weighted kNN 预报法的预测结果也可以很好的跟踪电力负荷的周期性与趋势性的变化规律, 但是与 AREL 的预测结果相比较而言基于

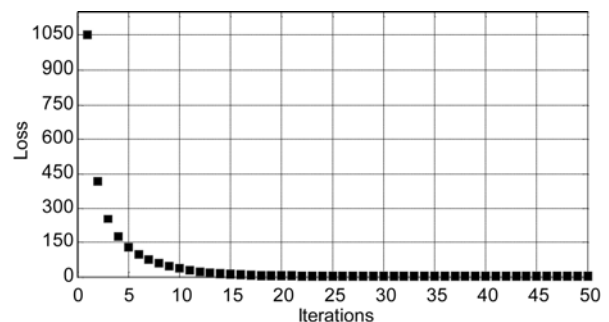


图4 基于 AREL 的 weighted kNN 预报法  $L_{T'}$  变化的过程

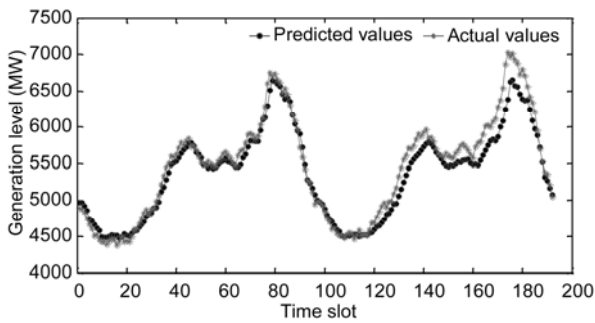


图5 基于 AREL 的 weighted kNN 预报法的预测结果与实际用电负荷的比较

AREL 的 weighted kNN 预报法预测结果更加平稳, 预测曲线也比较光滑.

图6显示了基于 AREL 的 weighted kNN 预报法预测结果平均绝对百分误差随着迭代的进行而变化的过程. 从图6中可以看出, 基于 AREL 的 weighted kNN 预报法的预测结果随着迭代的进行, 平均绝对百分误差逐渐减小. 因为随着迭代的进行 AREL 得到的组合模型  $F_T$  对训练数据的拟合效果进一步提高, 因而该模型的参数可以更好地反映系统状态动态变化的客观规律, 通过以该模型参数作为权重可以更加精确地选择与待预测点附近具有相同变化规律的时间序列片段, 因此随着迭代的进行, 基于 AREL 的 weighted kNN 预报法的预测结果平均绝对百分误差逐渐减小.

在本文算例中的参数设置下, 虽然基于 AREL 的 weighted kNN 预报法也取得了较好的预测结果, 但是与 AREL 预测结果相比较而言, 预测效果稍差一点. 如基于 AREL 的 weighted kNN 预报法经过 50 次迭代最小平均绝对百分误差为 2.19%, 而 AREL 最小

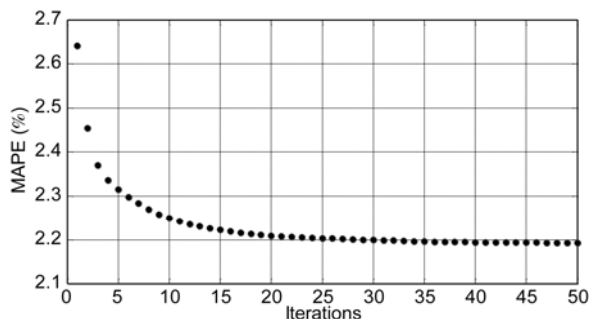


图6 基于 AREL 的 weighted kNN 预报法平均绝对百分误差随着迭代的进行而变化的过程

平均绝对百分误差达到 2.10%.

尽管在本文算例的参数设置下, 基于 AREL 的 weighted kNN 预报法与 AREL 预测结果相比较而言, 预测效果稍差. 但是与 AREL 相比较而言基于 AREL 的 weighted kNN 预报法对算法的参数设置不敏感, 基于 AREL 的 weighted kNN 预报法具有更高的稳定性. 图7和8分别给出 AREL 与基于 AREL 的 weighted kNN 预报法的模型阶次从 1~100 变化时, 经过 30 次迭代后两种算法预测结果的平均绝对百分误差随最大模型阶次的变化而发生变化的过程(图中当平均绝对百分误差大于等于 1 时, 置平均绝对百分误差为 1).

从图7与8的结果比较而言, 基于 AREL 的 weighted kNN 预报法随着最大模型阶次的变化预测结果的平均绝对百分误差比较稳定, 基于 AREL 的 weighted kNN 预报法对模型阶次的设置不敏感, 因而对于任意一个时间序列而言, 基于 AREL 的 weighted kNN 预报法的预测结果也具有更高的可信度.

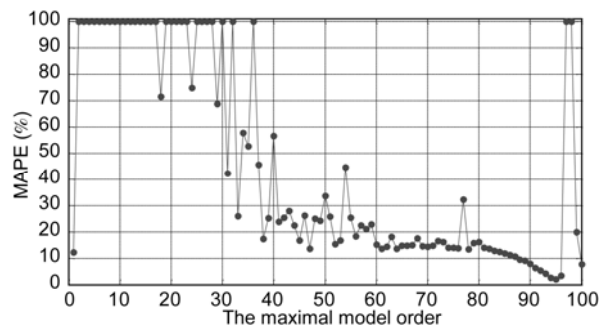


图7 AREL 预测结果的平均绝对百分误差随最大模型阶次的变化而变化的过程

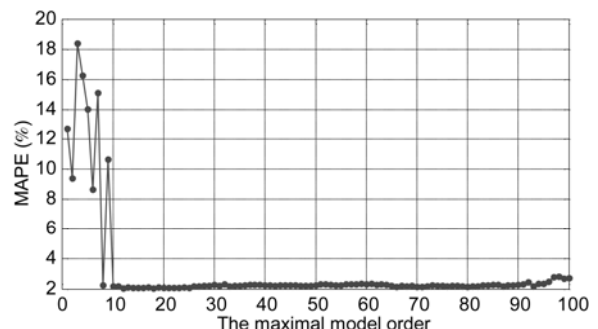


图8 基于 AREL 的 weighted kNN 预报法预测结果的平均绝对百分误差随最大模型阶次的变化而变化的过程

### 3.3 对钢铁企业用电负荷的预测

随着电力市场的发展和参与者对于本身用电管理要求的提高, 对于大型用电企业自身的用电进行预测的需求变得越来越迫切. 对于大型用电企业而言, 电能耗费是企业生产成本的重要组成部分, 为了对企业生产进行更科学地管理, 对企业效益进行预估, 需要对于大型用电企业未来的用电量进行预测, 以便核算电能成本; 同时, 由于大型用电企业用电的特殊性, 随着电力市场改革的深入和推广, 大型用电企业所在电网出于更加严格的安全性考虑, 需要提前得到企业的用电量信息, 以便进行电能管理, 提高电网运行的安全稳定性.

对钢铁企业而言, 企业用电与生产过程相关. 企业生产设备众多, 且生产环节之间相互关联, 既有离散生产过程, 也有连续生产过程, 从而造成企业用电表现出 irregularity. 另外大型用电设备的启停易造成企业用电的大幅度负荷波动. 同时对企业众多的大型用电设备的检修行为依赖对设备运行状况的经验判断, 这也为企业用电带来不确定性与随机性. 如下图 9 给出某大型钢铁企业某一天的用电负荷数据, 其中每 5 min 一个数据点, 每天共 288 个数据点.

图 9 显示出该大型钢铁企业用电具有大幅度冲击性及其不稳定性特征, 这一天的平均用电负荷为 1076.3 MW, 最大用电负荷为 1153.9 MW, 最小用电负荷为 968.5 MW, 负荷波动幅度达到 185.3 MW, 负荷波动幅度占平均负荷的 17.22%. 大型钢铁企业用电负荷中可利用预测信息不稳定, 规律性差, 这造成对大型钢铁企业进行电力负荷预测非常困难.

以该大型钢铁企业共 11 d 的实际用电数据为原始数据, 该数据序列为每 5 min 一个采样点, 取其中的前 10 d 共 2880 点数据作为训练数据. 该数据前 20 个自相关函数数值如图 10 所示.

从图 10 中可看出, 自相关函数不能快速衰减到零, 该大型钢铁企业实际用电负荷时间序列非平稳.

下面基于 AREL 的 weighted kNN 预报法对该钢铁企业用电负荷进行预测. 算例中  $\varepsilon$  置为训练数据中最大用电负荷的 0.5%, 迭代次数选择为 15 次. 下图 11 给出了基于 AREL 的 weighted kNN 预报法的损失 ( $L_{T'}$ ) 与预测结果平均绝对百分误差随着迭代的进行而变化的过程.

从图 11 可以看出, 随着迭代的进行  $L_{T'}$  逐渐减

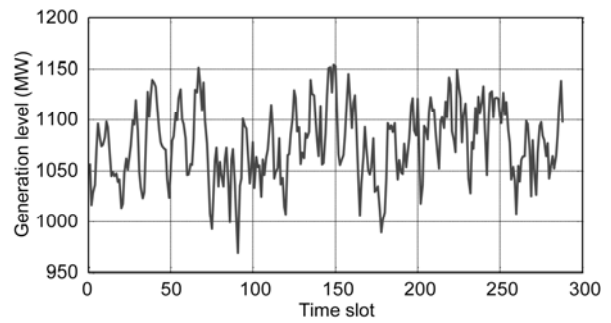


图 9 大型钢铁企业用电负荷数据

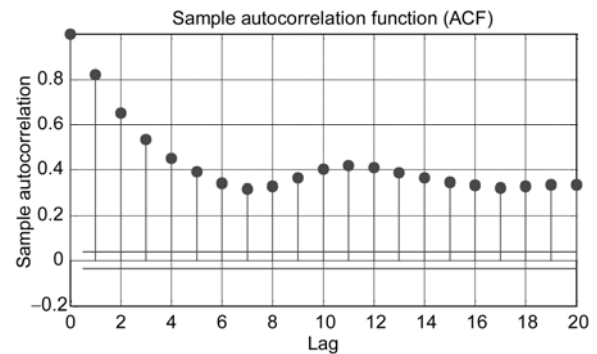


图 10 大型钢铁企业用电负荷数据自相关函数

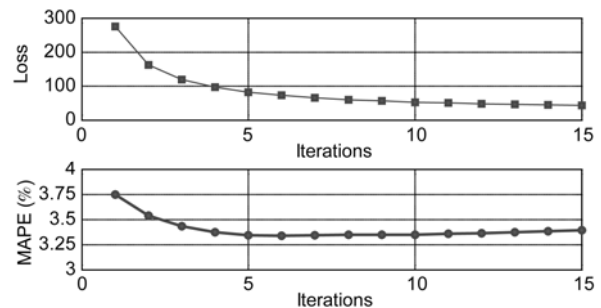


图 11 基于 AREL 的 weighted kNN 预报法  $L_{T'}(W_{train}, F_{T'})$  与预测结果平均绝对百分误差

小, 满足文中定理 2 给出的结论. 训练误差以指数速度下降, 说明基于 AREL 的 weighted kNN 预报法是一种有效的学习方法. 随着迭代的进行, 对一天 288 个点的预测结果平均绝对百分误差从 3.75% 降低到 3.39%, 基于 AREL 的 weighted kNN 预报法随着迭代的进行提升效果明显, 该算法完全可以用作对非平稳时间序列的分析.

另外一个很明显的现象是随着迭代的进行基于

AREL 的 weighted kNN 预报法在第 5 次迭代平均绝对百分误差取得最小(3.34%), 然后随着迭代的进行平均绝对百分误差又逐渐升高达到(3.39%)。这说明下述两个问题。

1) 基于 AREL 的 weighted kNN 预报法的方法虽然可以通过迭代对 AR 模型达到提升的效果, 但是当时间序列训练样本受到较大的随机噪声影响时, 基于 AREL 的 weighted kNN 预报法与传统的 Boosting 算法一样也容易发生过拟合。虽然在基于 AREL 的 weighted kNN 预报法中通过引入  $\epsilon$ -不敏感损失函数降低了模型对随机因素的过拟合程度, 另外采取 weighted kNN 预报法也进一步降低了随机因素的影响。但是如果训练样本受到较大的噪声影响, 数据不规律, 可预测性差的条件下任何算法都无法回避过度拟合这一问题。

2) 与问题 1) 相对应, 图 11 也说明大型钢铁企业用电不规律, 可利用预测信息不稳定, 因而对大型钢铁企业进行负荷预测非常困难。这一现象也可以从预测结果中看出, 如下图 12 给出了基于 AREL 的 weighted kNN 预报法对钢铁企业用电负荷 1 d 共 288 个点的用电负荷预测的结果与实际用电负荷的比较。

图 12 显示基于 AREL 的 weighted kNN 预报法可以很好地跟踪钢铁企业负荷的变化趋势, 对绝大部分数据点上(208 个数据点)的预测绝对误差小于 50 MW。但是钢铁企业用电负荷的随机突变形式, 造成在突变点上预测精度不高, 约有 80 个数据点上预测绝对误差大于 50 MW。

图 12 结果说明了基于 AREL 的 weighted kNN 预报法是一种有效的钢铁企业负荷预测方法, 但是钢铁企业的随机的、大幅度冲击性用电负荷特征造成基于 AREL 的 weighted kNN 预报法预测精度无法进一步提升甚至发生过拟合。提高数据质量是提升预测精度的另一种有效途径。但要提升数据质量涉及到钢铁企业的具体生产流程, 及其整个企业的生产计划与内部各个大型用电设备的检修等一系列具体问题, 本文对此不做进一步阐述。

#### 4 结论

本文首先在集成学习理论的基础上, 提出了基于 Boosting 梯度下降理论的 AREL。这种建模方法不依赖于操作者的经验和能力, 对问题不需要过多的

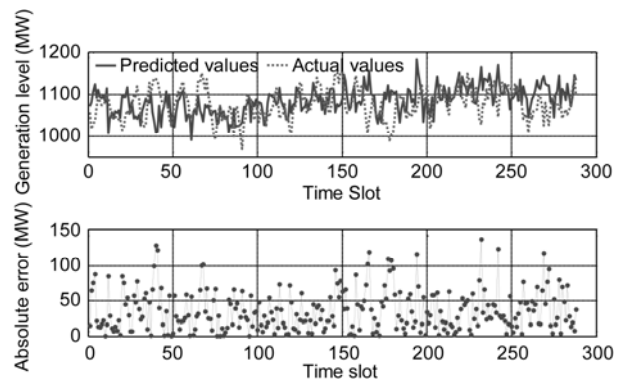


图 12 基于 AREL 的 weighted kNN 预报法预测结果

先验知识, 而又能被普通的工程技术人员使用。理论分析和实例测试均表明这种建模方法性能良好。为了进一步提高这种建模方法的预测精度, 避免了 AREL 在迭代过程中精确拟合每一个训练样本点, 从而降低对随机过程进行一次试验观察中随机因素对预测模型的影响, 引入了不敏感损失函数。进而为了提高 AREL 对非平稳时间序列处理能力, 提出了基于 AREL 的 weighted kNN 预报方法, 提高 AREL 对噪声样本的鲁棒性, 防止 AREL 发生过拟合, 降低 AREL 对模型阶次设置的敏感性。

通过对实际数据的测试, 结果显示随着迭代的进行 AREL 对时间序列预测效果的提升比较明显, 进而说明通过 Boosting 技术可以有效提升 AR 模型的性能。与 AREL 相比较而言基于 AREL 的 weighted kNN 预报法对算法的参数设置不敏感, 基于 AREL 的 weighted kNN 预报法具有更高的稳定性, 并且基于 AREL 的 weighted kNN 预报法可以有效地处理非平稳时间序列。

基于 Boosting 梯度下降理论的时间序列建模方法可以分析并建立合理的时间序列预测模型, 从而提高对时间序列的预测精度。另外本文基于 Boosting 梯度下降理论的时间序列建模方法不仅仅局限与对 AR 模型进行 ensemble learning, 其他的时间序列模型完全适用本文提出的建模法。在分析并建立了合理的单个时间序列预测模型(如周期自回归模型, 向量自回归模型, 门限自回归模型, 子集门限自回归模型, 时变自回归模型, 非线性自回归模型等等), 通过本文提出的建模方法可以进一步提高对时间序列的预测精度, 预测模型的泛化能力。这种建模方法具有重要的理论价值与广泛的应用前景。

## 参考文献

- 1 陈树勇, 宋书芳, 李兰欣, 等. 智能电网技术综述. 电网技术, 2009, 33(8): 1-7
- 2 管晓宏, 翟桥柱, 冯泳翰, 等. 一类含积分约束的生产制造系统优化调度. 中国科学: 技术科学, 2010, 40(1): 41-51
- 3 高云龙, 高峰, 翟桥柱, 等. 大型用电企业关口平衡问题研究. 中国电机工程学报, 2009, 28(19): 71-77
- 4 高云龙, 高峰, 潘金艳, 等. 高耗能企业关口平衡优化调度及其输出功率控制方式. 中国电机工程学报, 2010, 30(19): 76-83
- 5 Alexiadis M C, Dokopoulos P S, Sahsamanoglou H S. Short term forecasting of wind speed and related electrical power. Sol Energy, 1998, 63(1): 61-68
- 6 杨秀媛, 肖洋, 陈树勇. 风电场风速和发电功率预测研究. 中国电机工程学报, 2005, 25(11): 1-5
- 7 张龙, 熊国良, 柳和生, 等. 基于时变自回归模型与支持向量机的旋转机械故障诊断方法. 中国电机工程学报, 2007, 27(9): 99-103
- 8 徐峰, 王志芳, 王宝圣. AR 模型应用于振动信号趋势预测的研究. 清华大学学报, 1999, 29(4): 57-59
- 9 王耀青, 刘维奇. 带正则变化尾误差的函数系数自回归模型的概率性质. 山西大学学报(自然科学版), 2008, 31(3): 318-322
- 10 Brown B G, Katz R W, Murphy A H. Time series models to simulate and forecast wind speed and wind power. J Appl Meteorol, 1984, 23(8): 1184-1195
- 11 孙春顺, 王耀南, 李欣然. 小时风速的向量自回归模型及应用. 中国电机工程学报, 2008, 28(14): 112-117
- 12 Franses P H, Paap R. Model selection in periodic autoregressions. Oxford Bull Econom Statist, 1994, 56(4): 421-439
- 13 赵宏伟, 任震, 黄雯莹. 基于周期自回归模型的短期负荷预测. 中国电机工程学报, 1997, 17(5): 348-351
- 14 Lütkepohl H. New Introduction to Multiple Time Series Analysis. Berlin: Springer, 2005
- 15 Allen P G, Morzuch B J. Twenty-five years of progress, problems, and conflicting evidence in econometric forecasting. What about the next 25 years?. Int J Forecast, 2006, 22(3): 475-492
- 16 Greene W H. Econometric Analysis. New Jersey: Prentice Hall, 2003
- 17 Ewing B T, Kruse J B, Schroeder J L, et al. Time series analysis of wind speed using VAR and the generalized impulse response technique. J Wind Eng Ind Aerodyn, 2007, 95(3): 209-219
- 18 费万春, 白伦. 自协方差非平稳时间序列的时变参数自回归模型. 中国科学 A 辑: 数学, 2009, 39(1): 71-78
- 19 金阳, 安鸿志. 带有重尾扰动项的非线性自回归模型. 中国科学 A 辑: 数学, 2005, 35(1): 39-45
- 20 吴奇峰, 赵延孟, 李元, 等. 非参数自回归模型异方差的小波检验. 应用数论学报, 2009, 32(4): 595-607
- 21 杜秀丽, 汪凤泉. 非平稳环境激励下模态参数识别的连续时间 AR 方法. 中国科学 E 辑: 技术科学, 2009, 39(10): 1736-1742
- 22 Du X L, Wang F Q. Modal identification of system driven by lévy random excitation based on continuous time AR model. Sci China Ser E-Tech Sci, 2009, 52(12): 3649-3653
- 23 Granger C W J, Joyeux R. An introduction to long-memory time series models and fractional differencing. J Time Ser Anal, 1980, 1(1): 15-29
- 24 Hannan E J, Deistler M. The Statistical Theory of Linear Systems. New York: Springer, 1988
- 25 Chen R, Tsay R S. Nonlinear additive ARX models. J Am Stat Assoc, 1993, 88(423): 955-967
- 26 刘晨辉. 电力负荷预报理论和方法. 哈尔滨: 哈尔滨工业大学出版社, 1987
- 27 赵宏伟, 任震, 黄雯莹. 考虑周周期性的短期负荷预测. 中国电机工程学报, 1997, 17(3): 211-216
- 28 Schapire R E, Singer Y. Improved Boosting algorithms using confidence-rated predictions. Mach Learn, 1999, 37(3): 297-336
- 29 Drucker H, Cortes C, Jackl L D, et al. Boosting and other ensemble methods. Neural Comput, 1994, 6(6): 1289-1301
- 30 Schapire R E, Freund Y, Bartlett P L, et al. Boosting the margi: A new explanation for the effectiveness of voting methods. Ann Stat, 1998, 26(5): 1651-1686
- 31 汪荣鑫. 随机过程. 西安: 西安交通大学出版社, 1995
- 32 胡峰. 自回归模型参数的递阶辨识. 自动化学报, 1994, 20(4): 464-469
- 33 林正华, 冯仁忠. 自回归模型参数的最小二乘估计. 吉林大学自然科学学报, 2001, 39(2): 1-4
- 34 孟昭为. 向量自回归模型参数的极大似然估计. 山东工程学院学报, 2001, 15(2): 25-28
- 35 顾岚. 时间序列分析在经济中的应用. 北京: 中国统计出版社, 1994
- 36 Koutsoyiannis A. Theory of Econometrics, an Introductory Exposition of Econometric Method. London and Basingstoke: The Macmillan Press LTD, 1977
- 37 Valiant L G. A theory of the learnable. Commun ACM, 1984, 27(11): 1134-1142
- 38 Kearns M, Valiant L G. Cryptographic limitations on learning Boolean formulae and finite automata. J ACM, 1994, 41(1): 67-95
- 39 Schapire R E. The strength of weak learnability. Mach Learn, 1990, 5(2): 197-227
- 40 Freund Y. Boosting a weak algorithm by majority. Infn Comput, 1995, 121(2): 256-285

- 41 Freund Y, Schapire R E. A decision theoretic generation of online learning and an application to boosting. *J Comput Syst Sci*, 1997, 55(1): 119–139
- 42 Friedman J H. Greedy function approximation: A gradient boosting machine. *Annals Stat*, 2001, 29(5): 1189–1232
- 43 Gao Y L, Gao F. Edited AdaBoost by weighted kNN. *Neurocomputing*, 2010, 73(16-18): 3079–3088
- 44 Vapnic V N. *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995
- 45 Lange M. On the uncertainty of wind power predictions—Analysis of the forecast accuracy and statistical distribution of errors. *J Solar Energy Eng*, 2005, 127(2): 177–184
- 46 Möhrle C. *Uncertainty in Wind Energy Forecasting*. Cork: National University of Ireland, 2004