1 绪论

1.1 研究背景、意义及价值

时间序列是按照时间顺序记录的一系列数据点,是一种非常常见的数据形式。理想的时间序列数据都存在着一定的内在规律，并表现出一定的特性。然而，实际的时间序列数据常常都会受到外部因素的影响，使得在分析和寻找其内在规律和特性时存在一定的困难。在现实中有很多问题的分析都可以归结为时间序列的分析处理问题，如金融数据，能源数据，气象数据，GDP 数据，电力数据，医疗数据等。通过对时间序列的分析，我们能够更好的认识时间序列所代表的意义并解释其内在的变动规律，从中找寻有价值的信息从而指导我们做出正确的决策。

近年来，国内外诸多学者致力于研究时间序列分析模型，并且不断地把新的思想和方法融入到新的模型结构中。同时各个领域的学者也在不断地把新方法运用到各个领域里，因此时间序列分析及应用得以在各个领域迅速发展。如生物信息学，遗传学，多媒体，社会学，经济学，金融学等。近十年来，数据挖掘领域积累了大量的时间序列数据，各个平台开源数据和技术，都为时间序列数据的研究和发展打下了坚实的基础。当今时间序列的研究主要集中于序列匹配，模式识别，异常检验，主题发现，索引，聚类，分类，可视化，分割，趋势分析，相似检验，自动文摘和长短预测[57]。作为时间序列分析的主要用途之一，时间序列预测是统计学、经济学和管理学等研究中的热点和难点[58]。时间序列预测是通过对被预测事物历史信息的分析和研究，发现其内在规律并建立时间序列预测模型对未来时刻的信息进行预测。本文的主要研究内容之一就是有关金融时间序列中较常见的股票市场指数预测的问题。

金融市场在现代社会经济中扮演着极其重要的角色，经济活动影响着世界各国的经济发展。预测金融资产的未来价格一直是金融市场上的一个经久不衰的话题。因为准确的预测金融市场的未来的表现情况，无论对个人投资者、机构投资者还是政府金融机构都具有显著的意义。他们可以通过提前预估未来市场的走势而提前拟定相关策略并及时调整资产配置降低决策风险。股票市场指数的预测是投资者和研究人员最常用、最重要并且最具有挑战性的金融时间序列预测问题之一。尤其是近来量化投资的火热又把金融时间序列的处理和预测分析推向了一个新的高潮，再加上机器学习的快速发展，使得越来越多的学者投身于机器学习在金融领域的研究之中。

早期的金融时间序列主要通过统计学手段分析与研究。早期预测模型主要有

随机游走模型，指数平滑（ESM）、移动平均法（MA）、自回归（AR）、平滑过渡自回归模型[54]（STAR）、自回归差分移动平滑模型（ARIMA）、向量自回归（VAR）模型、自回归条件异方差模型（ARCH），广义自回归条件异方差模型（GARCH）。大量的研究证明，GARCH 模型在对价格波动率的预测和分析时可以取得良好的效果。目前金融时间序列的分析和预测仍然是以 GARCH 模型为基础的[59]。

机器学习和计算机技术的快速发展，使得机器学习和深度学习算法在金融时间序列的分析和预测的应用中变得越来越广泛。如树（TREE）模型，集成树（Ensemble Tree）模型，支持向量机（SVM）模型，人工神经网络（ANN）模型以及基于上述模型的一些发展和改进模型（RNN、LSTM、GAN 等）。研究表明，机器学习算法在金融时间序列的分析和预测中具有良好的表现性能。

本文主要基于 EMD（经验模态分解）方法把原始时间序列分解成不同的时间序列，这些序列代表着原序列不同时域和频域的特征，然后对分解得到的序列进行建模，充分挖掘序列隐藏的信息。对于一个趋势性较强的时间序列往往在不同的时间内表现出不同的趋势，很难捕捉其内在的变化规律，利用经验模态分解方法可以把序列分解成不同的特征序列，来达到简化序列的目的。本文的研究方法与其他学者略有不同，其他学者关于 EMD 与预测模型的结合是把所有分解出来的子序列都进行建模，然后再把模型集成。本文使用方法的特殊之处在于文中对于分解出来序列的特征仅仅构建一个模型，大大简化了模型的复杂度，提高了效率。而且本文为了避免在分解序列时使用到未来的信息，近而建立了基于 EMD 分解的自适应模型，来研究在不涉及未来信息的情况下经验模态分解是否对预测有帮助。

本文改进提出了一个基于 EMD 的新的预测框架，可以明显提高预测效率，而且在一定程度上也能提高预测的精度。使用 EMD 进行序列分解把复杂的序列分解成若干简单的、具有不同特征的序列，然后依据不同序列的特征进行建模，仅仅建立一个单一的模型，简化传统的基于 EMD 的预测模型框架，提高模型的预测效率及模型预测精度，为基于 EMD 的时间序列处理问题提供新的思路。而且，本文也着重研究了经验模态分解中“前瞻性偏差”（Look-ahead bias）对模型预测结果的影响，并改进提出了一种消除前视性偏差的预测模型，使得模型预测结果更加具有现实意义。

1.2 文献综述

1.2.1 股票指数预测研究文献综述

股票市场指数的预测是投资者和研究人员最重要也是最具有挑战性的金融

时间序列预测问题之一[19]。关于股票市场是否具有可预测性的问题由来已久，早在 1900 年 Bachelier 第一次用一个随机游走方式来表征股票价格的变动。后来也有一部分学者凭借自己的经验去测试价格变化的随机游走特征[42,43,44,45]。1970 年 Malkiel 和 Fama[46]基于有效市场假说对股票市场进行了一些研究。有效市场假说表明所有新信息都会立刻反应到资产价格中，因此未来资产价格变动于过去和现在的信息无关。然而，很多研究者通过实验反驳了市场有效假说，经验证据表明股票市场在某种程度上是可以预测的。如文献[47-53]中等研究者和学者的实证实验都表明股票市场在一定程度上是可预测的，并且不断涌现的基本面分析方法、技术分析方法以及大量研究股票市场走势、回报率以及波动率的文章，也间接表明了股票市场在一定程度上是可以预测的。

传统的时间序列预测方法主要是一些计量经济学模型如指数平滑（ESM）、AR、MA、加权移动平均法、ARMA、ARIMA、STAR[54]、VAR、ARCH、GARCH。这些模型的基本思想是一致的，都是基于一定的假设，或对数据进行变换，然后建立线性模型。例如 ARIMA 就是对数据差分后的 ARMA 模型，GARCH 是对收益或者收益残差和波动率进行平方然后对取平方得到的数据进行回归预测分析。

Ariyo, Adewumi and Ayo(2014)[55,13]用 ARIMA 模型来预测纽约证券交易所指数（NYSE）和尼日利亚证券交易所指数（NSE）,他们认为 ARIMA 模型在股票指数的短期预测中具有特殊的潜力，并且后来他们又对比研究了 ARIMA 和人工神经网络（ANN）来预测 NYSE 指数，认为 ANN 模型更加具有适应性。E.Chong, C.Han and F.C.Park(2017)[7]等比较了 AR 模型与应用主成分分析（PCA）、受限的玻尔兹曼机（RBM）和自编码（AE）的降维方法的 ANN 和 DNN 对一些股票价格进行预测分析，他们相信没有理由认为机器学习得到的结果一定比自回归得到的结果更加具有优越性。M.Kumar and M.Thenmozhi (2009)[35] 收益率变换对数据处理，然后使用 ARMA 模型、ANN、以及 ANN 与 ARMA 的混合模型来对 NIFTY 指数进行预测。

Ju Jie Wang 等(2012)[27]利用指数平滑模型（ESM）、ARIMA、ANN 以及用遗传算法（GA）优化的混合模型来对道琼斯指数（DJIAI）和深圳综合指数（SZII）进行预测研究，他们认为混合模型性能表型会更好。Ha Young Kim 等(2018)[6]研究 GARCH 模型、LSTM 模型以及他们的混合模型在股票价格波动率的预测中的应用，他们认为将 LSTM 模型的与多个 GARCH 相结合可以提高预测效果。传统的统计模型 ARMA 和 ARCH 一直在金融时间序列中起着及其重要的作用，绝大多是计量经济学模型都与他们有关。而且 ARCH 模型作为获得 2003 年诺贝尔经济学奖的计量经济学成果之一，足以表明其在时间序列处理中的地位。而且，目前许多有关计量经济学的分析和预测研究仍然是以 GARCH 为基础。

然而，金融时间序列本质上是复杂的、嘈杂的、动态的、非平稳的、非线性的、非参数的和混沌的[1,16,19,,56]。大量的基于非线性处理模型的出现也为金融时间序列的分析和预测带来了新的处理方法。特别是由于计算效率的提高一些传统机器学习模型和深度学习模型在金融时间序列中得到了非常广泛的应用。

Y.Baek and H.Y.Kim(2018)[1]利用数据增强提出了有两个 LSTM 模型组成的组合模型对标准普尔 500（S&P500）指数和 KOSPI200 指数进行预测研究，结果表明他们提出的有两个 LSTM 的组合模型优于一般的 DNN、RNN 和单独的 LSTM 模型。D.M.Q.Nelson, A.C.M.Pereira(2017)[10]利用 LSTM 模型预测巴西圣保罗 IBovespa 指数的走势，并且与随机森林（RF）、多层感知机（MLP）和随机模型相比，LSTM 的准确率更高。Y. Chen and Y. Hao(2017)[19]利用加权的支持向量机（SVM）和加权的最近邻（KNN）模型预测了短期、中期和长期上证综合指数和深圳综合指数以及指数走势，他们认为经过加权的 SVM 模型预测效果会好一点。

还有些学者利用机器学习模型来分析国际市场间的相互影响对预测的作用如：L. S. Malagrino, N. T. Roman 和 A. M. Monteiro(2018)[20]利用朴素贝叶斯模型结合全球市场其它地区主要的股票市场指数的走势预测了巴西圣保罗 IBovespa 指数的走势，虽然预测结果一般但是朴素贝叶斯模型利用解释和分析市场间的联系。同样 M. Thenmozhi 和 G. Sarath Chand(2016)[4]利用支持向量机（SVM）回归分析研究了全球六个市场美国（道琼斯，标普 500），英国（FTSE-100），印度（NSE），新加坡（SGX），香港（恒生）和中国（上证）股票市场指数之间的价格信息传递，实证结果表明，加入了全球市场因子的预测结果优于仅仅只有滞后价格信息的预测结果。

还有一些学者在使用历史数据的前提下增加了技术指标或者是市场新闻情绪、投资者情绪等指标。如 X.Zhang(2018)[2]加入了在雪球网提取的情感指数利用 SVM 来预测上证指数的走势，他们的结果表明，情感指数的加入，有利于市场走势的预测。B.Weng(2018)[15]使用在线数据源预测短期股票价格，结合历史股票价格、几项有名的技术指标、特定股票已发布新闻的数量和情绪分数、谷歌搜索给定股票的搜索趋势、维基百科特定页面访问量。使用神经网络（ANN）回归集成，支持向量机（SVM）回归集成，提升树（Boosting Tree）和随机深林（Random Forest）回归来预测股票的价格，实证表明提升树模型对股票价格预测效果更好。在机器学习快速发展的今天，大量的机器学习和深度学习算法已经渗入到金融时间序列的预测中如人工神经网络 ANN（E.Guresen(2011)[3], E.Chong, C.Han and F.C.Park(2017)[7], M.Qiu, Y.Song(2018)[11], B.Weng(2018)[15], X. Zhong(2017)[16]）、支持向量机 SVM（X.Zhang(2018)[2], M.Thenmozhi(2016)[4], B. Weng(2018)[15],

Q.Xu(2019)[18], Y.Chen(2017)[19]），深度神经网络 DNN（RNN，LSTM）（Y.Baek and H.Y.Kim(2018)[1], T.Fischer and C.Krauss(2018)[5], H.Y.Kim and C.H.Won(2018)[6], H.M(2018)[9], D.M.Q.Nelson[10]）等。

1.2.2 经验模态分解应用研究文献综述

经验模态分解（Empirical Mode Decomposition , EMD）是 1998 年由美籍华人专家 Huang 在瞬时频率概念的基础上提出的一种新型自适应信号时频分析技术，是从时序的角度进行时间序列分析的工具，主要用于信号分解和去噪。经过十几年的发展，基于 EMD 的时频分析方法逐步形成了一套完备的理论体系，受到国内外诸多学者的广泛关注和研究[58]。经验模态分解因为其前提假设少，适用性广，提出之后很快就被用于许多理工科领域[59]。

现有文献中结合 EMD 进行预测的模型，按照对子序列（IMF）是否使用相同的模型来说可以主要分为两类。

第一类是对不同的子序列使用不同的模型。Y.Xiang(2018)[61] 等利用 EMD 对降雨时间序列进行分解，然后对第一个 IMF 使用 SVR 进行预测，其他分量使用 ANN 进行预测，把所有的预测结果加总得到最后结果，他们认为分解后的时间序列能够得到更好的预测效果。田大中（2015）[63] 利用 EMD 对网络流量序列分解，然后结合 ARIMA 和 SVM 分别对各分量的 IMF 和余项进行预测，然后将预测结果利用 ANN 非线性叠加作为最终的预测结果。他认为经过组合的模型优于单个 ARIMA 和 SVM 模型。张文风（2018）[59]对使用经验模态分解后的子序列分别使用不同的模型进行预测，然后把预测结果利用 RBF（径向基）神经网络再次进行建模得到最终的预测值。

第二类也是最流行的一种方式，对每个子序列使用相同的模型进行预测。S.Huang(2014)[67]等对基于 EMD 分解的个子序列分别利用 SVM 进行建模预测，然后把预测的值直接进行相加。他们认为经过混合的 EMD—SVM 模型的预测效果优于单独的 ANN 的 SVM 模型。Z.Qu(2019)[68]等对风速时间序列进行 EMD 分解，并对高频的 IMF 进行二次分解，然后对每一个序列使用 Flower-pollination 算法优化的 BPNN 进行预测然后把预测结果相加。陈渝（2019）[60]等利用 EMD 分解门诊量序列，然后对每个分量 IMF 序列建立 LSTM 模型去预测门诊量，最后将结果直接相加。他们的结果表明用 EMD 进行分解数据时得到的组合模型的预测精度比不进行序列分解时的单一 LSTM 和单一 SVM 的预测精度更好。H. Zhou(2019)[62]等利用 EMD 对股票指数走势进行预测研究，他们把分解的 IMF 利用因子分解机（Factorization machine）结合神经网络来预测上证综合指数、标普 500 以及纳斯达克综合指数的走势。E. Meng(2019)[64]等利用 EMD 分解水流量序

列并结合 SVM 对水流量进行预测，他认为各分量的组合模型所得到的结果优于单个序列的 ANN 模型和 SVM 模型。N. Sun(2018)[65]提出了一种自适应的两层分解序列方法-结合 EMD 和 VMD(变分模态分解, Variational Mode Decomposition, VMD)两种序列分解技术来分解序列，然后利用 PCA 降维技术分别选取各分量结合超限学习机（extreme learning machine, ELM）对风速进行预测，他们认为两阶段分解后的组合超限学习机模型优于序列单次分解的模型和序列不分解所构建的模型。Q. Tan(2108)[66]等提出了自适应 EEMD 与 ANN 的混合方法对水流量进行预测，他们认为在雨季的时候新提出的模型可以更准确的预测未来的水流量，但是在旱季的时候新提出的模型预测结果不如季节自回归（SAR）的预测结果。

1.3 本文主要内容与创新

1.3.1 本文的主要内容

本文主要改进了以往与 EMD 算法混合的预测模型，提出了一个基于 EMD 的新的预测框架，可以明显提高预测效率，而且在一定程度上也能提高预测的精度。其次本文也研究了结合 EMD 算法进行预测的模型中存在的前瞻性偏差问题，针对这个偏差，本文探讨了剔除该偏差后，结合 EMD 的预测模型是否还具有优越性。最后本文为消除前瞻性偏差的预测模型进行了改进，使得模型的预测结果更加具有现实意义。

文章内容进行如下安排：

第 1 章为绪论。本章阐述了本文的研究背景及意义。并且针对股票指数预测以及经验模态分解的应用分别进行了综述。

第 2 章为 EMD 算法的概述。本章中主要对 EMD 算法进行了概述，并且阐述了 EMD 算法中存在的问题，介绍了几种改进的 EMD 算法。并且表明了文章对端点问题和停止准则问题的处理方法。

第 3 章为基于 EMD 与神经网络的混合预测模型。本章为论文的主要内容之一，本章主要针对 EMD 与 ANN 混合预测模型提出了改进意见，并进行实证研究和对比研究，并依据实证研究得出改进模型的优越性。

第 4 章为基于 EMD 与神经网络的自适应预测模型。文章为论文主要内容的另一部分，本章主要针对 EMD 算法中存在的前瞻性偏差进行研究，并探讨了前瞻性偏差对混合模型预测结果的影响，实证研究表明剔除前瞻性偏差后结合 EMD 算法的预测模型的预测结果并没有得到改善。据此，本章中对剔除前瞻性偏差的自适应模型提出了一点改进。实证结果表明，改进后的模型对预测结果有明显的提升作用。

第 5 章为结论与展望。本章主要对论文的主要结论和成果进行总结，并针对论文的不足，规划下一步的研究方向。

1.3.2 本文的创新点

本文的创新点主要有以下两点：

首先，针对基于 EMD 分解的组合预测模型中模型数量较多、规模复杂、误差累积和计算量大等问题，本文提出了一种基于 EMD 算法的单一模型预测结构。本文第三章的“后视实验”研究以及第四章的“预测实验”研究都证明了本文提出的单一预测模型 S-EMD-ANN、S-AEMD-ANN 和 S-AEMD-ANN^{a} 优于文献中传统意义上的组合模型 EMD-ANN[66,68,71,72]，AEMD-ANN[65、66]和 AEMD-ANN^{a}。基于 EMD 的单一预测模型简化了网络结构，从而减少了模型个数并且简化了参数调节，使得模型的训练时间得到大幅度改善。并且从预测结果的各个评价指标来看，单一预测模型比传统的组合预测模型在预测精度上也有一定的提升。

其次，针对基于经验模态分解过程中存在的前瞻性偏差问题，本文结合文献 Q. Tan(2018)[66]、X. Zhang(2015)[69]、Dennis(2017)[70]中的自适应混合模型的概念提出了一种改进的自适应预测模型。第四章的主要内容是基于经验模态分解数据中端点失真问题的改进，由于自适应预测模型会使得端点效应在测试数据中极大的暴露出来，所以靠近端点的数据分解会严重失真，而在训练数据中却不会存在这样的问题，这样就会使得使用训练数据训练出来的模型可能在测试数据上表现很差，甚至并不一定优于单独的预测模型。经过 4.2.2 小节的分析，本文提出了一种改进的自适应预测模型，在模型训练和测试时删除一些可能存在严重失真的变量，尽量用真实的数据去建模预测。4.3 节大量实证实验结果表明，经过变量处理的 AEMD-ANN^{a} 预测模型优于不经过变量处理的 AEMD-ANN 预测模型，经过变量处理的 S-AEMD-ANN^{a} 预测模型优于不经过变量处理的 S-AEMD-ANN 预测模型。并且，与文献[70]中结论不同的是，本文结论认为即使剔除前瞻性偏差，如果依据本文的改进模型能对单独的预测模型起到改善作用。本文提出的改进模型 S-AEMD-ANN^{a} 在文中的各种实证结果上都要优于单独的 ANN 模型，这也说明了本文提出改进方案的有效性。

2 EMD 算法概述

经验模态分解是一种处理非线性、非平稳性信号的方法, 其本质是对时间序列进行平稳化处理。与小波分解不同的是 EMD 是由数据驱动的一种序列分解方式，其根据原序列的极值特征直接对原序列进行分解而不依赖于基函数。

2.1 EMD 的概念和基本算法

2.1.1 EMD 算法的基本概述

首先 Huang 在其文章中定义了本征模态函数(Intrinsic mode functions)。当函数满足以下两个条件时，我们视其为一个本征模态函数：

（1）在整个数据集中，极值点的个数和过零点的个数必须相等或最多相差 1 个；

（2）在任意时间点，由局部极大值定义的包络线和由局部极小值定义的包络线的均值为零。

EMD 分解是一种直观的、后验的、自适应的分解，完全由数据驱动。该方法的实质是对数据中固有振荡模态的特征时间尺度进行经验识别，然后对数据进行相应的分解。EMD 分解基于以下假设：

（1）信号序列至少有两个极值，一个极大值一个极小值；

（2）时间尺度的特征由极值之间的时间间隔来定义；

（3）如果数据没有极值只有拐点，则可以通过对数据差分来获取极值点。

该分解过程从本质上讲是一个筛选过程，筛选出频率由高到低的本征模态函数，并剩余一个单调残差序列，也称残差序列为趋势项。则根据其定义可知，EMD 可以自适应地把一个时间序列分解成有限个数的本征模函数（IMF）和一个残差项。其算法描述如下：

Step1：输入原始序列信号 $x(t)$；

Step2：获取原始序列 $x(t)$ 的所有极大值点和极小值点，并分别对极大值点和极小值点用 2 条 3 样条插值曲线拟合，得到一条上包络线（upper envelope）$\mathrm{e}_u(t)$ 和下包络线（lower envelope）$\mathrm{e}_l(t)$。

Step3：计算包络线的均值；

$$m(t)=[e_u(t)+e_l(t)]/2 \qquad (2\text{-}1)$$

Step4：计算 $x(t)$ 和 $m(t)$ 的差，得到 $h(t)$；

$$h(t)=x(t)-m(t) \tag{2-2}$$

Step5：检查$h(t)$是否满足上述的两个 IMF 的条件，如果$h(t)$是一个 IMF 或满足停止准则，则转到 Step6；否则令$x(t)=h(t)$，并重复 Step1-Step4 直到$h(t)$是一个 IMF 或满足停止准则。

Step6：计算残差项（residue）$r(t)=x(t)-h(t)$，如果$r(t)$是一个单调函数或者只有一个极值点则分解完成；否则令$x(t)=r(t)$，重复 Step1-5。

至此，$x(t)$被分解为若干个本征模态函数和一个趋势项，用公式（2-3）表示：

$$x(t)=\sum_{i=1}^{m}h_i(t)+r(t) \tag{2-3}$$

在上述的分解过程中，分解得到的本征模函数很容易满足第一个条件，即极值点的个数与过零点的个数相等或最多相差 1 个，然而第二个条件即局部上包络线和下包络线的均值为 0 却不一定能通过变换满足。而且当第二个条件发挥到极致时，可能会消除物理意义上的振幅波动，可能使得到的 IMF 成为恒定的纯调频信号。为了确保 IMF 各组成部分保持足够的振幅和频率调制的物理意义，我们必须确定一个停止筛选过程的标准[73]。

2.1.2 EMD 算法的停止准则

根据 2.1.1 中 IMF 的定义，严格 IMF 应满足局部上包络函数和下包络函数的均值恒为 0，实际上很难做到这一点只能接近于 0。针对这种情况，一些学者通过实验给出了一些相对较为理想的停止准则也被称为局部包络条件。

EMD 的提出者 Huang(1998)[73]在其论文中给出了标准偏差(SD)准则，通过限制 SD 的值来判断是否停止筛选。SD 是由两个连续的筛选结果计算得到的，用公式（2-4）表示[73]：

$$SD=\sum_{t=0}^{T}\left[\frac{\left|h_{i(k-1)}(t)-h_{i(k)}(t)\right|^2}{h_{i(k-1)}^2(t)}\right] \tag{2-4}$$

其中$h_{i(k)}(t)$为第i个 IMF 第k次迭代的值，$h(t)$计算公式见（2-2）。当 SD 小于研究者给定的某个ε值时，就认为该$h_{i(k)}(t)$为第i个本征模态函数。Huang 在其文章中通过大量实验给出ε取值在 0.2 和 03 之间时分解效果较好。后来 Huang 等人又给出了另一个相似的停止准则用公式（2-5）表示：

$$SD=\sum_{t=0}^{T}\left[\frac{\left|m_{i(k)}(t)\right|^2}{h_{i(k-1)}^2(t)}\right] \tag{2-5}$$

其中 $m_{i(k)}(t)$ 为第 i 个 IMF 第 k 次迭代的上下包络线的均值，其计算公式见（2-1）。然而，由于以下原因 SD 标准很难实现:首先，SD 小到什么程度就足够小，这其实并没什么标准；其次，这个准则不依赖于 IMF 的定义，因为平方差很小但不能保证函数会有相同数量的过零点和极值。为了弥补这些缺点，Huang (1999,2003)等人提出了第二种类型的标准，称为 S 停止准则。使用这种类型的停止准则，筛选过程只有在零点个数和极值个数相等或最多相差 1 并且保持连续 S 次相同时才停止。Huang(2003)等人的大量试验表明，S 的最佳取值范围应该在 3 到 8 之间，但更倾向于较低的数字。显然，任何选择都是有特定的使用范围，需要一个严格的理由[74]。

法国学者 Gabriel Rilling(2003)[75]等基于 Huang 等的研究提出了一个较为常用评价函数 $\sigma(t)$ 来控制筛分过程的停止

$$\sigma(t)=\left|\frac{m(t)}{a(t)}\right|=\left|\frac{e_u(t)+e_l(t)}{e_u(t)-e_l(t)}\right| \qquad (2\text{-}6)$$

其中 $m(t)=[e_u(t)+e_l(t)]/2$ 为上下包络线的均值，$a(t)=[e_u(t)-e_l(t)]/2$。并设置三个阈值 θ_1、θ_2、α，当满足以下两个条件：

（1）当 $\sigma(t)$ 里面小于 θ_1 的比率达到 α；

（2）$\sigma(t)$ 不存在大于 θ_2 的值。

则认为分解结束，中止筛选，一般取 $\alpha\approx0.05$，$\theta_1\approx0.05$ 和 $\theta_2\approx10*\theta_1\approx0.5$ [75]。这也是本文中所采用的停止准则。

2.1.3 EMD 算法的端点问题

EMD 相关算法的应用过程中，端点效应的处理是一个重要的步骤。EMD 对信号进行分解时，需要去一步步“筛选”出本征模函数。如果数据在端点处不是极值，那么在对极值点三样条插值拟合构造上下包络线时，将会出现误差从而产生端点效应。在端点附近时，三样条插值将出现发散现象，随着本征模函数的一步一步提取，端点效应向内扩散最终导致污染数据[59]。Huang 在文章中指出在信号的两端，根据端点信号的振幅和频率，分别加两个特征波，可以有效抑制端点效应，但没有给出具体的做法。

处理端点问题最简单的方法是在迭代过程中不断地剔除非极值点的部分，但是对于预测问题来讲抛弃右端点的值也就意味着抛弃了近期的数据，然而近期的数据往往是包含信息量最大并且最重要的数据。所以如果是对时间序列做简单的

分析分解的话可以使用此方法，但是对于预测问题，往往不适合此方法。本文主要研究的是对股价指数的预测问题，不适合使用此方法。

另外一种解决端点问题的思想就是人为的在端点处添加一些数据，也称为延拓法。随着 EMD 算法的广泛应用，一些学者陆续的提出了一些较为经典的延拓方法，这些延拓方法大致可以概括为两种：直接延拓法和预测延拓法[59]。

直接延拓法就是通过一定的技术手段将原有数据向外延拓。目前主要的延拓方法有，镜像延拓[76]、偶延拓、Coughlin 法（Coughlin and Tung 2004）[79]、周期延拓[80]、极值延拓、斜率法(Datig and Schlurmaim 2004)[77]、平行线延拓、加窗口延拓[81]和 Rato 法(Rato 2008)[78]等。

预测延拓法是指根据预测模型对未来一些点进行外推预测的方法。常见的预测延拓方法有时间序列预测延拓，神经网络预测延拓，以及一些利用机器学习方法进行端点预测的延拓方法。与直接延拓相比预测延拓往往需要更大的计算量，而且要去根据具体情况选择合适的预测方法，这样给延拓带来一些不确定性。

本文主要研究的是模型的简化和优化问题，并不关注这些延拓方法的区别。所以本文取最为常用并且简单的延拓方法—镜像延拓法来处理端点问题。镜像延拓法通过对数据边界的极值点添加镜像,并以对称的方式增加 n 个极值点的方式来抑制分解过程中的端点效应。以图 2-1 增加一个极值点为例，镜像延拓法过程如下：

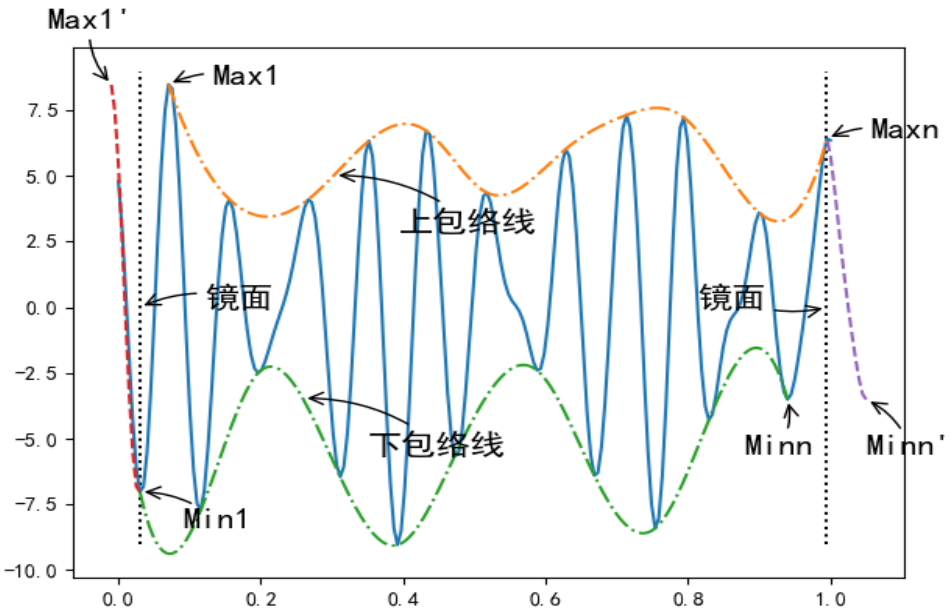


图 2-1 镜像延拓法过程

Step1：找到原始序列最左端的极值点 Min1（第一个极小值点），以及最右端的极值点 Maxn （最后一个极大值），如图 2-1 中所示；

Step2：分别以 Min1 和 Maxn 为镜面得到 Max1 的镜面对称点 Max1′ 和 Minn 的镜面对称点 Minn′，如图 2.1 所示；

Step3：最后根据数据原有的极大值点和极小值点加上有镜面对称得到的点，分别对运用三样条插值构建上包络线和下包络线。这样可以在一定程度上抑制端点的发散（图 2.1 给的上、下包络线为未进行延拓下的包络线）。

2.2 EMD 算法的改进研究

因为停止条件的限制，在经验模态分解过程中可能会出现模态混叠，使得不同特征的序列分解成一个子序列。针对此类问题一些学者提出了一系列的改进算法。其中在处理信号序列中最常用的是以下介绍的 EEMD 算法和 CEEMD 算法。在其它非信号序列数据处理中 EMD 算法和 EEMD 算法较为常用。本文研究的是有序列分解和简单神经网络混合模型的预测框架的优化问题，并不注重于不同的分解算法对结果的影响。所以本文选择最简单而且效率最高的 EMD 算法。

2.2.1 集成经验模态分解

针对经验模态分解存在的模态混叠现象，Huang(2009)[82] and Wu 等人提出集成经验模态分解(Ensemble Empirical Mode Decomposition, EEMD)算法，使得经验模态分解的应用得到了进一步的发展。

集成经验模态分解（EEMD）算法的主要思想是：先向原始序列数据添加一个白噪声，再对混合序列进行分解，重复 N 次。最后，将得到的 N 组 IMF 和趋势项子序列取平均作为最后的分结果。集成经验模态分解被认为是经验模态分解的一个极其重要的改进。经过 N 次集成平均后添加的白噪声可以认为基本相互抵消，能够更好的实现信号序列的分解，克服了 EMD 中出现的模态混叠现象。在 2.1.1 章节 EMD 算法的基础上，EEMD 算法步骤如下：

Step1：在原始信号中 $x(t)$ 添加一个白噪声信号 $\varepsilon_m(t)$，得到下列混合信号：

$$x_m(t) = x(t) + \varepsilon_m(t) \tag{2-7}$$

其中，m 表示第 m 次信号混合，$m = 1,2,\cdots,N$，N 为信号混合的总次数，是一个人为设定参数，通常取 $N = 100$[82]。

Step2：混合后的信号序列 $x_m(t)$ 进行经验模态分解，可得第 m 次分解的子序列 h_{mi} 和 r_m 其中 $i = 1,2,\cdots,n$ 为分解的 n 个 IMF（为了确保分解信号个数相同一般要求添加的白噪声为低信噪比的白噪声序列）。重复该步骤，直至 $m = N$。

Step3：每个 IMF 和余项 r 取平均，作为最终分解结果，得到第 i 项 IMF 和余项：

$$\bar{h}_i = \frac{1}{N}\sum_{m=1}^{N} h_{mi} \tag{2-8}$$

$$\bar{r} = \frac{1}{N}\sum_{m=1}^{N} r_m \tag{2-9}$$

Step4：得到 $x(t)$ 最终的分解结果：

$$x(t)=\sum_{i=1}^{n} \bar{h}_i + \bar{r} \tag{2-10}$$

其中， n 为分解的到的本征模函数的个数。

2.2.2 完整集成经验模态分解

根据 2.2.1 中的介绍，我们知道集成经验模态分解算法是通过添加噪声项来解决模态混叠问题。引入噪声项后我们假设经过多次添加噪声项，噪声之间可以相互抵消，但实际引入的噪声项并不能完全抵消。我们可以在应用中通过增加集成的次数来减弱残留噪声项的相对大小，但这样做会大大的增加计算量。为解决经验模态分解过程中的模态混叠问题，同时消除集成分解过程中添加的白噪声对信号重构的影响, Yeh J R and Shieh J S(2010)[83]提出了完整集成经验模态分解(Complete Ensemble Empirical Mode Decomposition, CEEMD)。该算法的主要思想是成对的添加噪声，并且使得所添加的白噪声和为零。基于 EMD 和 EEMD 算法，CEEMD 算法分解过程如下。

Step1：将白噪声成对的添加到原始信号 $x(t)$ 中，得到一组待分解信号，并且该组信号的白噪声序列和为零：

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x(t) \\ \varepsilon(t) \end{bmatrix} \tag{2-11}$$

其中， $x(t)$ 表示原始信号序列， $\varepsilon(t)$ 表示加入的白噪声， $x_1(t)$ 表示与正的白噪声叠加后的信号序列， $x_2(t)$ 表示与负的白噪声叠加后的信号序列。

Step2：分别对 $x_1(t)$ 、 $x_2(t)$ 进行 EMD 分解，重复 N 次。得到分解后的 IMF 子序列和余项：

$$\bar{h}_i = \frac{1}{2N}\sum_{m=1}^{N} (h_{mi}^1 + h_{mi}^2) \tag{2-12}$$

$$\bar{r} = \frac{1}{2N}\sum_{m=1}^{N} (r_m^1 + r_m^2) \tag{2-13}$$

其中， h_{mi}^1 、 h_{mi}^2 和 r_m^1 、 r_m^2 分别表示第 m 次混合信号 $x_1(t)$ 、 $x_2(t)$ 分解得到的子序列

和余项。

Step3：最终，由 CEEMD 算法将原始信号分解为

$$x(t)=\sum_{i=1}^{n}\overline{h_i}+\overline{r} \tag{2-14}$$

其中，n 为分解的到的本征模函数的个数。

2.2.3 具有自适应噪声的完整集成经验模态分解

EEMD 和 CEEMD 都是通过对原序列添加噪声来处理模态混叠问题。有些学者考虑在分解过程中添加噪声来处理模态混叠问题，提出了具有自适应噪声的完整集成经验模态分解(Complete Ensemble Empirical Mode Decomposition with Adaptive Noise, CEEMDAN)算法，主要思想是：在分解过程的每个阶段都添加自适应白噪声，通过计算唯一的余量信号获得 IMF 子序列，在较少的集成次数后就能够使重构误差几乎为零[84]。CEEMDAN 算法过程如下（Humeau-Heurtier(2105)[85]）：

Step1：在原始信号序列 $x(t)$ 中添加一些自适应白噪声：

$$x^i(t)=x(t)+\omega_0\varepsilon^i(t), i=1,2,\cdots,N \tag{2-15}$$

其中，$x^i(t)$ 表示第 i 次加入白噪声后的信号，ω_0 表示噪声系数（噪声标准差），$\varepsilon^i(t)$ 表示第 i 次添加的白噪声，N 表示集成次数。

Step2: 对每个信号 $x^i(t)$，分别采用 EMD 算法分解获得第一个 IMF 分量，然后对这 N 个分量取均值：

$$h_1(t)=\frac{1}{N}\sum_{i=1}^{N}h_1^i \tag{2-16}$$

其中，h_1^i 表示 $x^i(t)$ 分解得到的第一个 IMF 信号。则余项信号为：

$$r_1(t)=x(t)-h_1(t) \tag{2-17}$$

Step3：定义 $E_j(\bullet)$ 为使用 EMD 算法分离出第 j 个本征模函数的过程，分解 $r_1(t)+\omega_1E_1(\varepsilon^i(t))$，$i=1,2,\cdots,N$，直到分解出一个 IMF。$\omega_k$ ($k=1$ 在此步)在每一步允许选择的信噪比（SNR）。然后，得到第二个 IMF：

$$h_2(t)=\frac{1}{N}\sum_{i=1}^{N}E_1(r_1(t)+\omega_1 E_1(\varepsilon^i(t))) \quad (2\text{-}18)$$

Step4: $k=2,3,\cdots$ 计算第 k 个余项：

$$r_k(t)=r_{k-1}(t)-h_k \quad (2\text{-}19)$$

使用 EMD 算法分解 $r_k(t)+\omega_k E_k(\varepsilon^i(t))$，$i=1,2,\cdots,N$ 直到分解出一个 IMF，定义第 $k+1$ IMF：

$$h_{k+1}(t)=\frac{1}{N}\sum_{i=1}^{N}E_1(r_k(t)+\omega_k E_k(\varepsilon^i(t))) \quad (2\text{-}20)$$

Step5：回到 Step4，并令 $k=k+1$，直到余项不能再被分解。得到最终的余项：

$$r_n(t)=x(t)-\sum_{i=1}^{n}h_i \quad (2\text{-}21)$$

其中 n 为分解得到的 IMF 个数。最终原信号序列 $x(t)$ 被写作：

$$x(t)=\sum_{i=1}^{n}h_i+r_n(t) \quad (2\text{-}22)$$

2.3 本章小结

本章对经验模态分解算法进行了详细的介绍。介绍了经验模态分解算法的思想、基本原理以及算法实现过程。而且本章也对经验模态分解存在的问题以及其改进算法做了一些简单的介绍。本文并没有深入探讨不同的停止准则、不同的端点处理手段、不同的经验模态分解改进算法对结果的影响。本文主要的研究内容是经验模态分解与简单神经网络混合预测模型结构的改进。所以，在此说明本文所采用的停止准则为 2.1.2 中提到的法国学者 Gabriel Rilling(2003)[75]提出的停止准则，端点问题的处理依据 2.1.3 中详细介绍的而且较为常用的镜像延拓法，分解算法采用最简单并且效率最高的 EMD 算法分解序列。

3 基于 EMD 与神经网络的混合预测模型

3.1 相关算法介绍

3.1.1 ANN 算法介绍

人工神经网络(ANN)是一种模拟生物神经网络结构和功能的数学模型。因为其在变量间非线性关系强大的识别能力，在过去的几十年里得到了广泛的应用。在众多不同类型的人工神经网络中，反向传播人工神经网络(BP-ANN)由于其结构简单、易于实现，已被证明是解决许多问题的有力工具[66]。本文中使用三层神经网络预测股票指数，网络结构如图 3-1 所示。

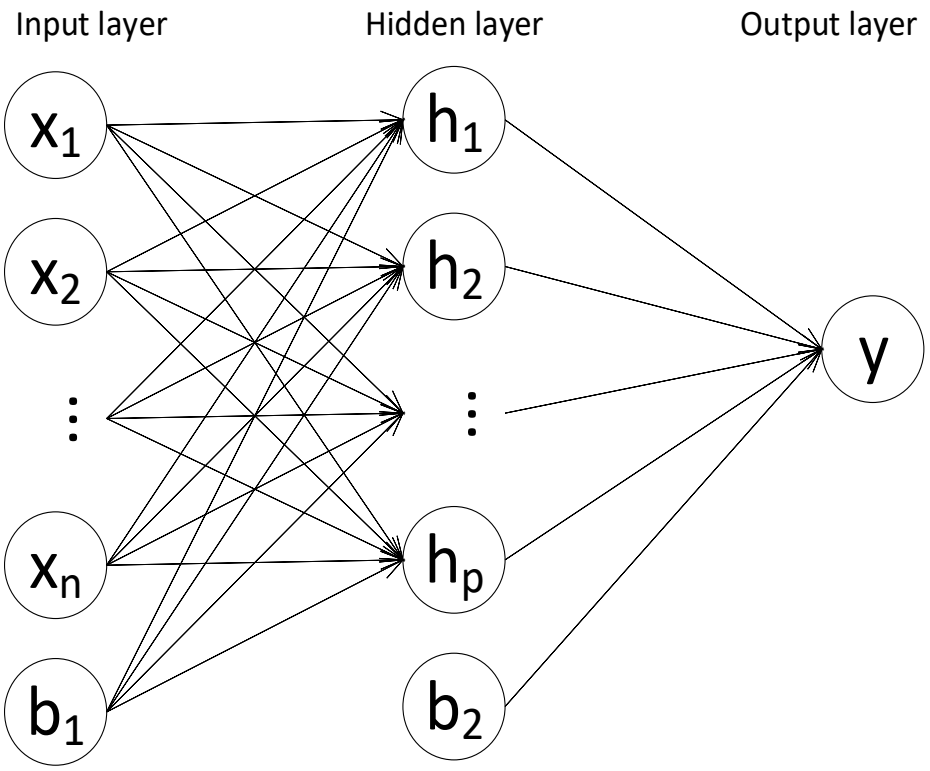


图 3-1 简单神经网络节点图

假设输入层有 n 个节点，隐藏层有 p 个节点，输出层有一个节点。则 b_1 是 p 维向量，b_2 是 1 维向量。对于隐藏层的节点 h_i 来说

$$h_i = \phi(\sum_{j=1}^{n} w_{ij} \bullet x_j + b_{1i}) \qquad (3\text{-}1)$$

其中 $i=1,2,\cdots,p$，h_i 为个隐藏层的第 i 个节点，w_{ij} 为输入变量 x_j 的权重值，b_{1j} 为第 j 个隐藏层节点的偏置，$\phi(\bullet)$ 为激活函数。

$$y_k=\psi(\sum_{j=1}^{p}w_{kj}\bullet h_j+b_{2k}) \qquad (3\text{-}2)$$

其中 $k=1$(图 3-1 只有一个输出，可以有多个输出)，y_k 为个输出层的第 k 个节点，w_{kj} 为隐藏变量 h_j 的权重值，b_{2k} 为第 k 个输出节点的偏置，$\psi(\bullet)$ 为激活函数。

3.2 EMD 与 ANN 混合预测模型及改进

自 1998 年经验模态分解法提出以来，经验模态分解算法已经取得了较大的发展，并且理论知识日趋完善。经验模态分解因为其前提假设少，适用性广，提出之后很快就被用于许多理工科领域[59]。正如第一章节文献综述所介绍的，使用 EMD 分解算法，然后对其子序列分别进行预测，最后在把预测结果相加这一预测模型框架在许多领域的研究都表明能取得较好的表现性能。然而这一预测模型框架，往往存在模型过多、模型复杂、计算量过大、误差累积、参数难以调优等一系列问题。特别是在 EMD 算法与机器学习相关算法结合时，例如 EMD-ANN[59,61,62,63,66,68,71,72,84,86]、EMD-SVM[58,61,63,64,67,]、EMD-LATM[60]等都会出现模型过多所带来的计算量大、超参数过多难以对模型调优等问题。本文针对这些问题提出了一种改进的预测模型结构，并结合实证实验研究模型的表现性能。人工神经网络模型由于具有非线性、非局限性的优良性质，在进行组合预测模型中是最常用的基本模型之一。本章针对 EMD 与 ANN 的混合预测模型改进提出一种基于 EMD 与 ANN 混合的单一预测模型。

3.2.1 基于 EMD 与 ANN 的预测模型

现实中许多时间序列都具有高度复杂的特征，其高度非线性非平稳性性等一系列性质，使得直接对序列分析变得比较困难，也难以用具体模型去捕捉时间序列的特性，进而难以建立精确的预测模型。近年来对时间序列的研究越来越广泛，一系列对时间序列的处理方法相继被提出，如单尺度分析、多尺度分析、DFA 趋势分析、MF-DFA 趋势分析[57]、先分解后组合[84]分析等一些方法。本文所研究的就是基于“先分解后组合”思想的一种方法。先分解后组合的模型框架即首先将复杂的时间序列使用一些分解算法分解成若干子序列，然后再对其子序列进行建模分析预测。已有文献所给出的预测模型都是多个预测模型的组合模型。其基本结构与混合的 EMD-ANN 模型相似，如图 3-2 所示，只是基础预测模型和分解算法略有不同。

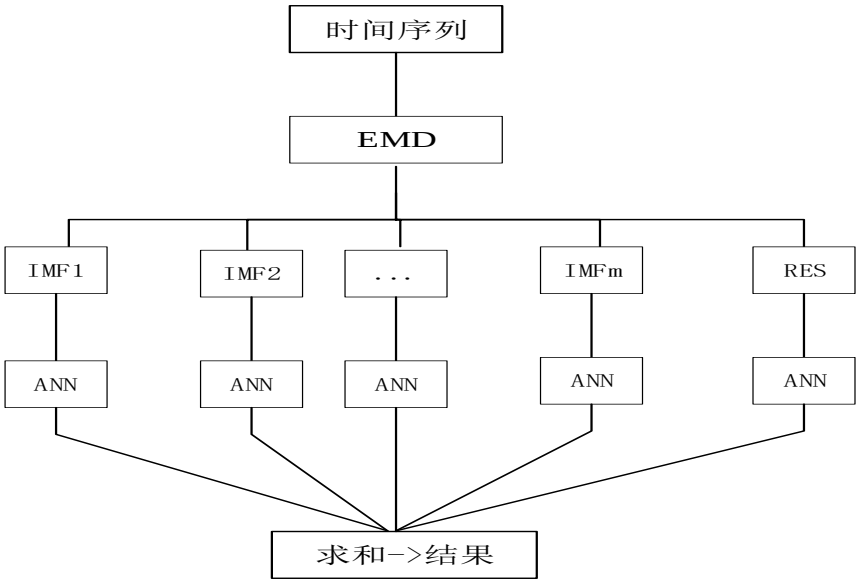


图 3-2 EMD 与 ANN 混合预测模型结构图

EMD-ANN 的预测框架结构如图 3-2 所示，其主要算法步骤如下：

Step1：时间序列数据使用 EMD 算法进行分解，得到子序列 IMF1，IMF2,……，IMFm 和一个余项 RES。

Step2：所有的子序列分别建立 ANN 模型，使用前l个序列值，预测第$l+1$项的值，具体变量预测方式如图 3-3 所示。

Step3：对所有子序列的预测结果进行叠加得到最终预测值。

$$\begin{bmatrix} \text{original:} & o_1 & o_2 & o_3 & \cdots & o_{l+1} & o_{l+2} & \cdots & o_n \\ IMF1: & i_{1,1} & i_{1,2} & i_{1,3} & \cdots & i_{1,l+1} & i_{1,l+2} & \cdots & i_{1,n} \\ IMF2: & i_{2,1} & i_{2,2} & i_{2,3} & \cdots & i_{2,l+1} & i_{2,l+2} & \cdots & i_{2,n} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ IMFm: & i_{m,1} & i_{m,2} & i_{m,3} & \cdots & i_{m,l+1} & i_{m,l+2} & \cdots & i_{m,n} \\ residue: & r_1 & r_2 & r_3 & \cdots & r_{l+1} & r_{l+2} & \cdots & r_n \end{bmatrix}$$

图 3-3 EMD-ANN 模型变量预测示意图

如图 3-3 所示，绝大多数学者都是对每一个 IMF 和余项都进行建模，最后把每个模型得到的结果叠加作为最后的预测值。如图 3-3 虚线标识所示，对每一个子序列（IMF 和 residue）分别用前l项序列值，预测第$l+1$项，并使用滑动窗口

法预测余下各项。

3.2.2 基于 EMD 与 ANN 的单一预测模型

传统的基于 EMD 的预测模型框架都是对于每一个分解出来的 IMF 进行建模预测，然而对于每个 IMF 进行建模代价往往要面临极大的计算量这一问题。特别是在使用机器学习模型对每个 IMF 进行预测时往往需要很大的计算量，特别是现在使用支持向量机、人工神经网络等算法在构建模型的时候，往往有一定量的超参数要进行试验调优，所以所组合的模型个数越多也就会有越多的超参数需要去确定。而且在对每个 IMF 结果进行叠加的时候往往会造成大量的累积误差，如果有 m 个模型叠加作为最后结果就会有 m 项的模型误差累积。再者有学者研究（H. Zhou(2019)[62]、张文风（2017）[59]）表明经过 EMD 分解的各项 IMF 往往具有一定的短期相关性，各 IMF 的特征之间具有一定的“交互作用[62]”，所以如果只是对各个子序列进行建模往往会忽略各子序列之间的相互影响。考虑到这些问题，本文提出了一种基于 EMD 算法的单一模型预测框架并基于此框架建立 S(single)-EMD-ANN 模型。模型框架如图 3-4 所示。

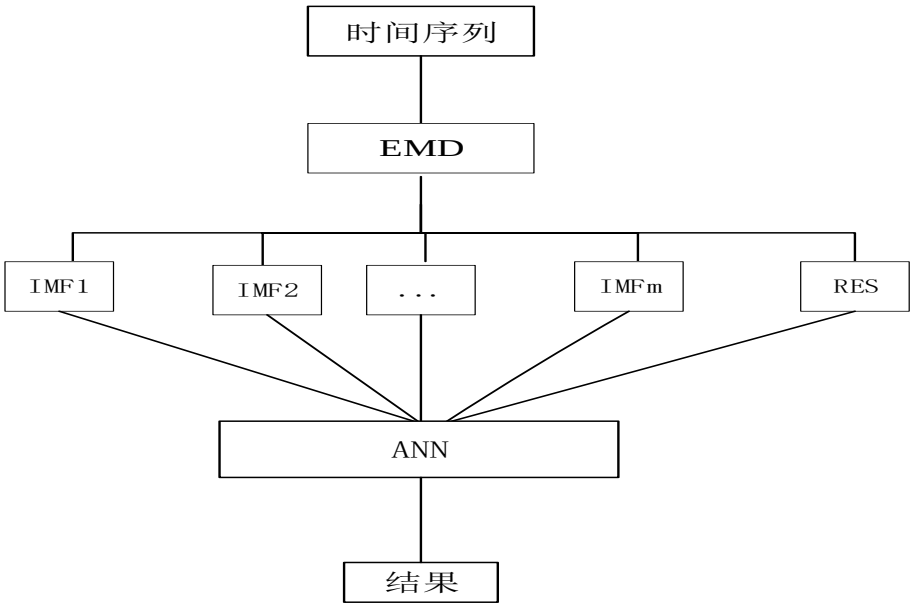


图 3-4 EMD 与 ANN 混合的单一预测模型结构图

S-EMD-ANN 的预测框架结构如图 3-4 所示，其主要算法步骤如下：

Step1：时间序列进行使用 EMD 算法进行分解，得到子序列 IMF1，IMF2，……，IMFm 和一个余项 RES。

Step2：所有的子序列建立一个 ANN 模型，用每一个子序列的前 l 个序列值，

直接预测第$l+1$项的值，具体变量预测方式如图 3-5 所示。

Step3：直接预测最终的结果值。

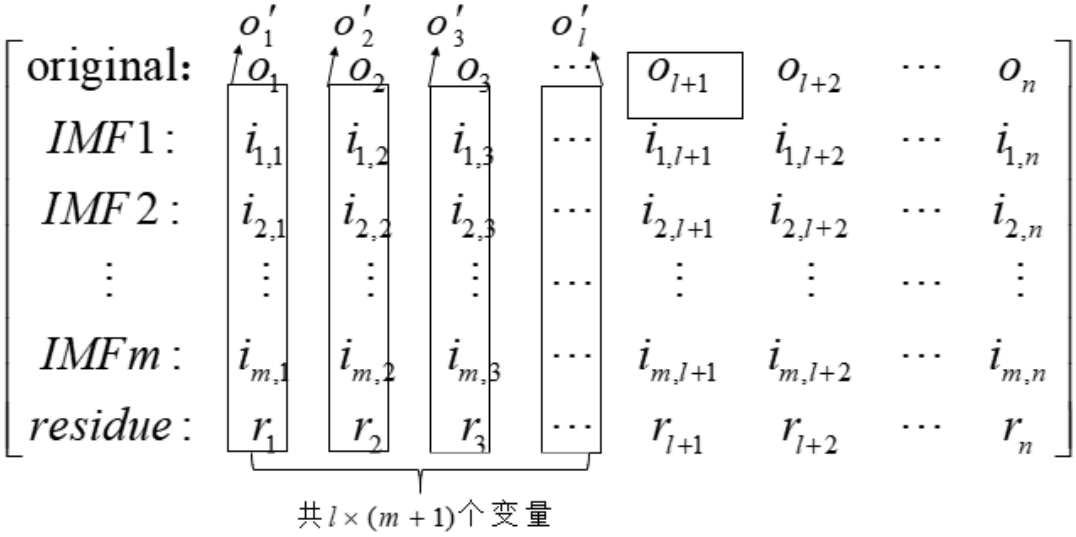


图 3-5 S-EMD-ANN 变量预测示意图

如图 3-5 所示，S-EMD-ANN 算法设计思想是把经过 EMD 分解出来的子序列当作是不同时域和频域的特征子序列，然后在每个子序列上选择一些滞后变量作为子序列的特征直接预测最后的结果。如图 3-5 所示把每个o_i时间点的所有分解值作为o_i的特征向量，最后用这些特征向量组成变量集$[o_1' \quad o_2' \quad \cdots \quad o_l']$，直接去预测最终值$o_{l+1}$。也就是用每个子序列前$l$个序列值，共$l\times(m+1)$个的变量去预测第$l+1$项，并使用滑动窗口法预测余下各项。

3.3 实证分析研究

3.3.1 数据介绍

本文实证研究数据主要选择上海证券综合指数简称上证指数和标准普尔 500 指数。上证指数选择从 1990 年 12 月 19 日（基准日）-2019 年 4 月 2 日的数据共 6916 个时间点序列数据。为了数据的一般性，对于标准普尔 500 指数我们也选择相同日期的数据即 1990 年 12 月 19 日-2019 年 4 月 2 日的数据共 7123 个时间点序列数据。考虑到要与线性模型结果作对比研究，线性模型不适合做长期的预测，并且在实际中对于股票指数的预测研究都是根据新的数据来更新参数，所以大部分研究都做的是短时间内的预测。如：Zhou(2019)[62]对几个不同地区的股指预测选择 250 个左右的作为测试集，Carnelossi Furlaneto(2017)[70]选择 250 个（大约一年的交易日数）值作为测试集。本文中每一个数据集我们都选择留下最

后 300 个数据点即一年多的股票指数作为测试数据。数据描述见表 3-1。

表 3-1 数据集描述

数据名称	训练数据	测试数据	均值	方差	数据日期范围
上证指数	6616	300	1924.18	1071.92	1990.12.19-2019.04.02
标普 500	6823	300	1253.42	601.77	1990.12.19-2019.04.02

3.3.2 基于 EMD 的数据分解

本小节主要是依据数据差分变换规则，对数据进行预处理然后再进行分解。EMD 分解算法最初提出是解决复杂的信号问题，信号序列一般都是高频、趋势性不强。然而，如图 3-6 和图 3-7 虚线所示，本文处理的股票指数时间序列具有较强的局部趋势性并且频率较低。所以本文对数据进行一步差分预处理。文献[87,88]认为差分运算可以放大高频信息，显示高频信息引起的极值点，从而为提高频率分辨率提供了可能[88]。而且差分数据可以对数据进行缩放，在学习类算法中对数据缩放是加快学习速度的一个重要方法。

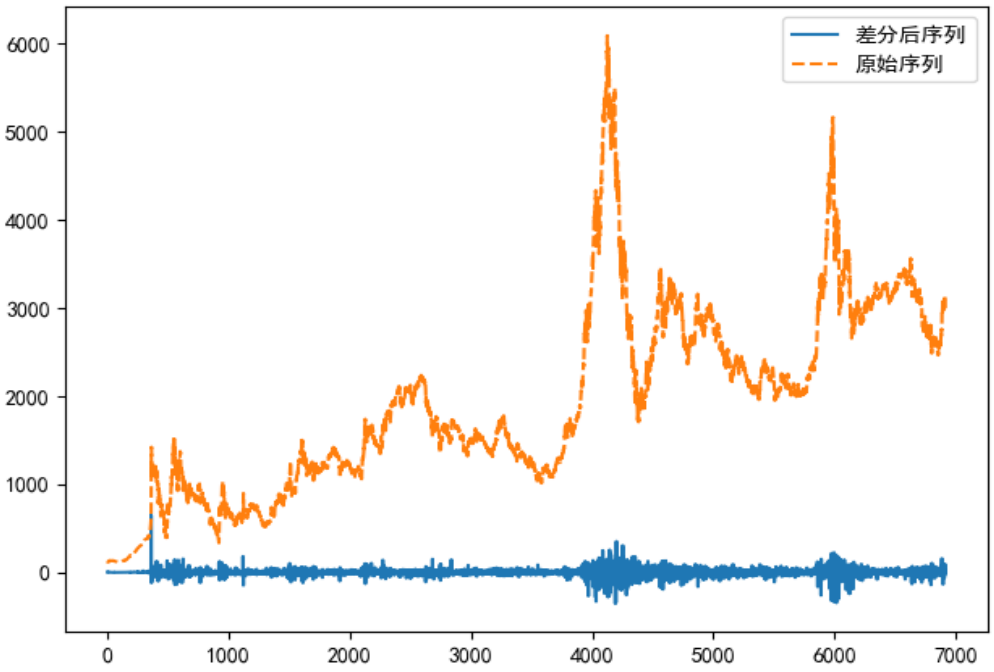


图 3-6 上证指数原时间序列数据和差分后的时间序列数据

此外，根据第二章 EMD 分解原理内容我们知道经验模态分解是由数据驱动的序列分解方式，再进一步讲经验模态分解是由序列的极值点驱动的序列分解方式，因为它主要是依据由极值点拟合的上下包络线分解的一种分解算法。所以我们可以知道当一个序列的极值点越多的话则上下包络线越接近真实的序列的局

部极值。表 3-2 中我们比较了股票指数的原序列和其差分后序列的极值点个数，可知本文中的两个股票指数序列差分后会有更多的极值点，可以使得上下包络线更加接近原始序列，也使得更多的点在包络线上。从图 3-6 和图 3-7 也可以看出差分后的序列的极值点比原序列极值点要多。同时，增加极值点个数减弱端点效应的影响这一点对于第四章的研究内容有重要的意义。

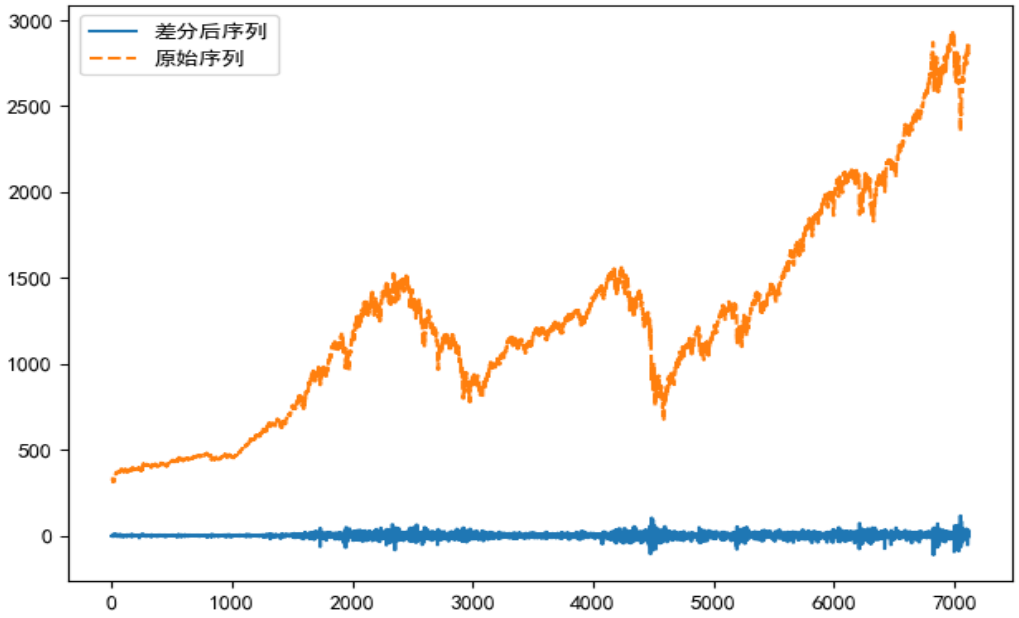


图 3-7 标准普尔 500 指数原时间序列和差分后的时间序列数据

表 3-2 数据集的极值点情况

数据名称		极大值点个数	极小值点个数	极值点个数
上证指数	原序列	1666	1666	3332
	差分后序列	2265	2264	4529
标普 500 指数	原序列	1827	1828	3655
	差分后序列	2380	2379	4759

上证指数差分后的时间序列进行前文所述的检验模态分解，得到如图 3-8 的分解结果。如图 3-8 所示，上证指数分解得到 12 个 IMF 和一个余项。

标准普尔 500 指数差分后的时间序列进行前文所述的检验模态分解，得到如图 3-9 的分解结果。如图 3-9 所示，标准普尔 500 指数分解得到 11 个 IMF 和一个余项。

图 3-8 和图 3-9 中我们看出，当一个特征子序列局部波动较大时，其它的子序列也会在一定的程度上表现出较大的波动性，这说明特征子序列之间存在着局

部的“交互作用”。

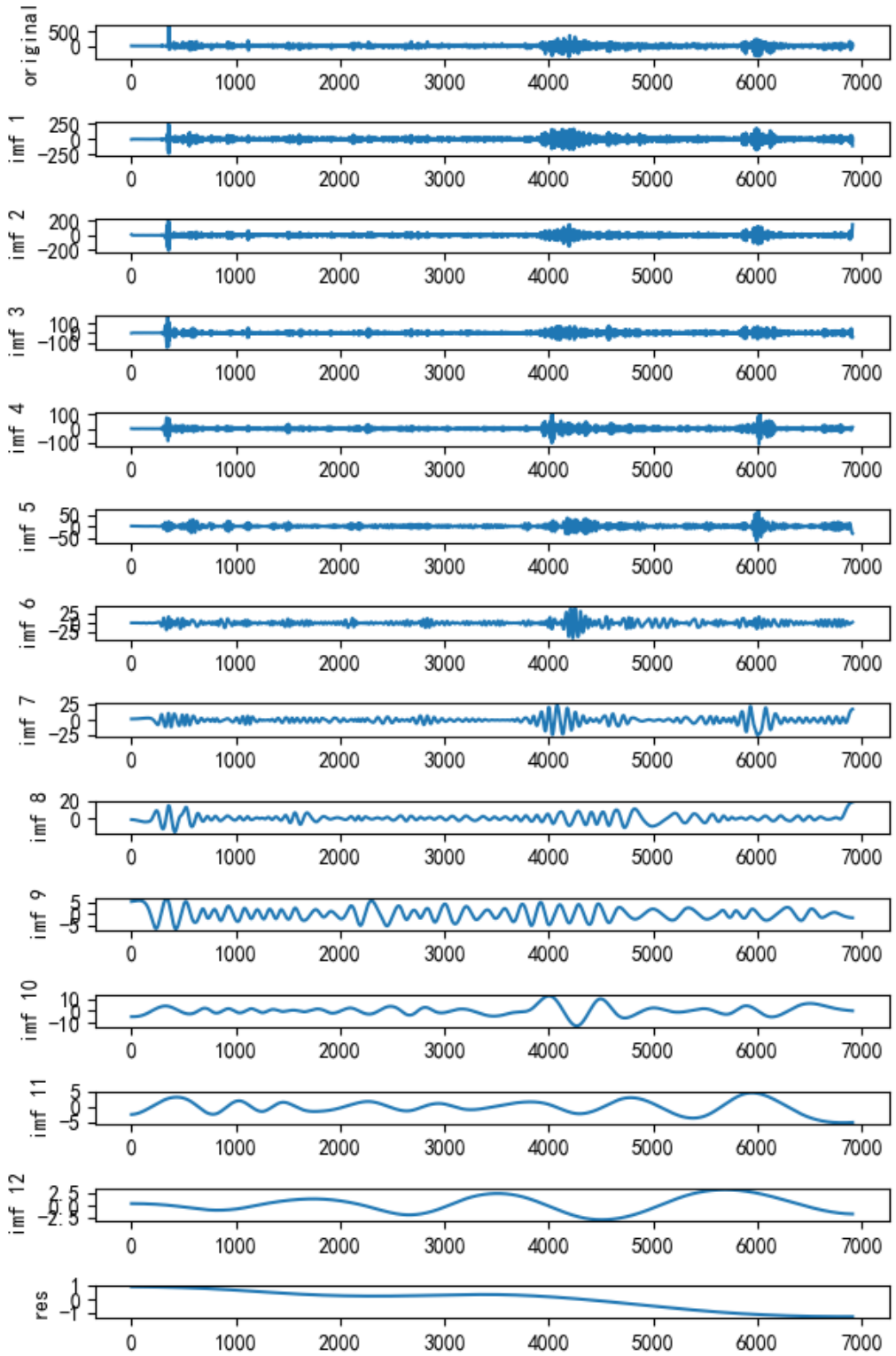


图 3-8 上证指数差分后的序列分解结果

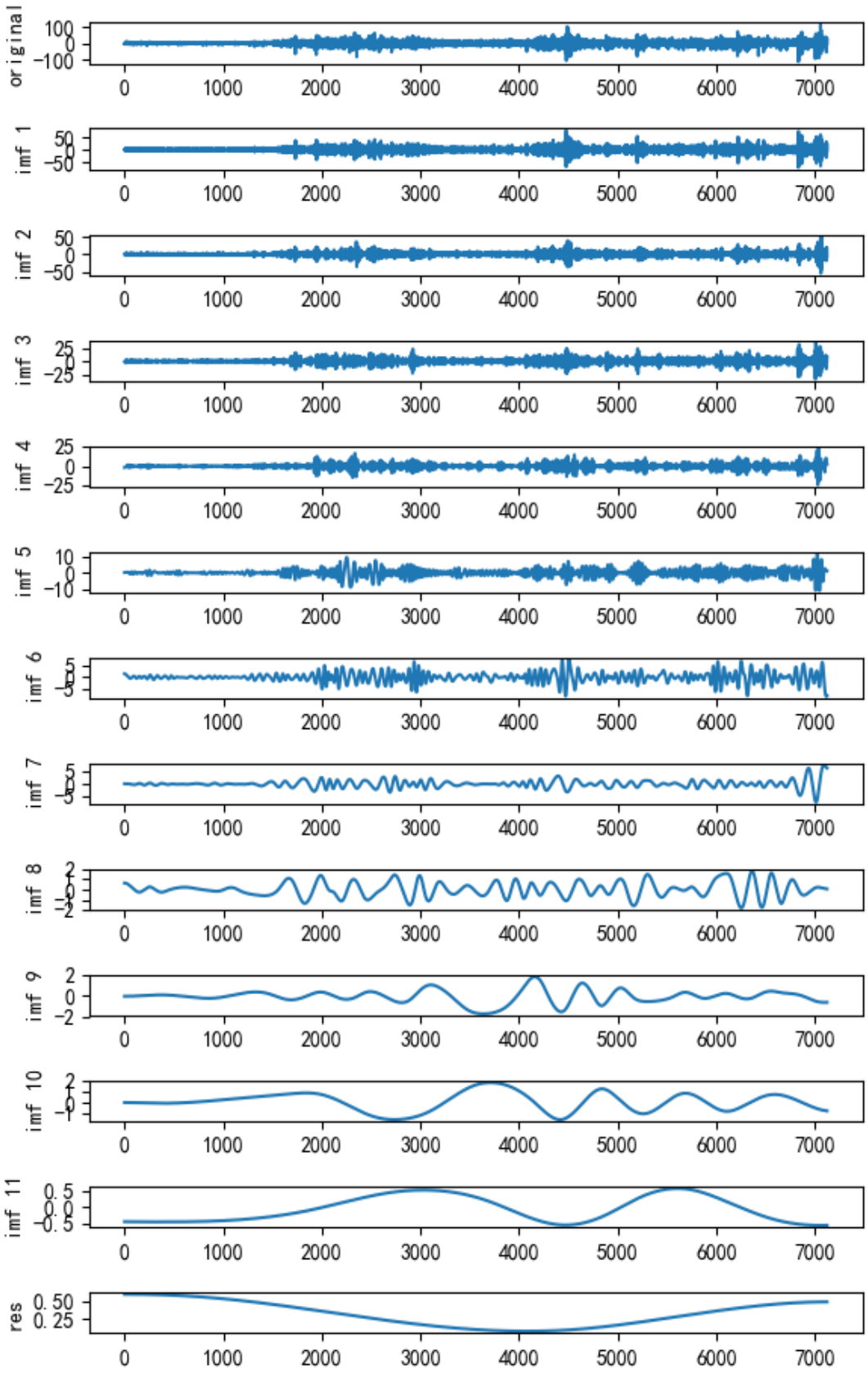


图 3-9 标准普尔 500 指数差分后的序列分解结果

3.3.3 模型性能评价指标

本小节实证结果从模型的总体预测精度以及模型预测结果之间差异性显著检验准则对预测效果进行综合评价。

3.3.3.1 整体预测准确度评估指标

文章中采用平均绝对误差(MAE)、平均绝对百分误差(MAPE)和均方根误差(RMSE)三个常用的总体误差评价指标对模型预测准确度进行评价。评价指标计算公式如下：

$$RMSE=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i-\hat{y}_i)^2} \tag{3-3}$$

$$MAPE=\frac{1}{N}\sum_{i=1}^{N}\left|(y_i-\hat{y}_i)/y_i\right| \tag{3-4}$$

$$MAE=\frac{1}{N}\sum_{i=1}^{N}\left|y_i-\hat{y}_i\right| \tag{3-5}$$

其中 y_i 为观测值， $\hat{y}_i$ 为预测值， N 为样本数量。

为了定量评价不同模型的性能，定义指标 RMSE、MAE 和 MAPE 的改进百分比。

$$p_{index}=(index_1-index_2)/index_1 \tag{3-6}$$

其中 $index$ 为 RMSE、MAE 或 MAPE， $index_1$ 和 $index_2$ 分别是模型 1 和模型 2 的评价指标值。

3.3.3.2 显著性检验指标

本文还采用 Diebold Mariano (DM)检验[90,91]和 Wilcoxon Signed rank (WS)检验来验证两种模型的测试数据的预测值之间是否存在统计学意义上的显著性差异。

首先，DM 检验是一种常用的非参数检验方法。假设一对预测值的误差 $(g(y_t,\hat{y}_{1t}),g(y_t,\hat{y}_{2t}))$，$t=1,2,\cdots,n$，$n$ 为测试样本数。模型预测结果的质量依据一些指定的损失函数 $g(\bullet)$ 来判断，常用的损失函数 $g_{it}=\left|y_t-\hat{y}_{it}\right|$， $g_{it}=(y_t-\hat{y}_{it})^2$，$g_{it}=\left|(y_t-\hat{y}_{it})/y_t\right|$ 等。假如两个模型预测效果相同，则两个模型的 $g(\bullet)$ 的差的期望为 0。即：

$$E[g_{1t}-g_{2t}]=0 \tag{3-7}$$

并令

$$d_t=g_{1t}-g_{2t} \tag{3-8}$$

很自然的根据观察到的样本均值进行检验：

$$\bar{d} = \sum_{t=1}^{n} d_t \tag{3-9}$$

其中 y_t 为样本 t 的观测值，$\hat{y}_{it}$ 为第 i 个模型的第 t 项样本的预测值。

假设向前 h 步（本文为向前一步预测也就是提前一天预测）预测。

$$V(\bar{d}) \approx n^{-1}\left[\gamma_0 + 2\sum_{k=1}^{h-1}\gamma_k\right] \tag{3-10}$$

$$\hat{\gamma}_k = n^{-1}\sum_{t=k+1}^{n}(d_t - \bar{d})(d_{t-k} - \bar{d}) \tag{3-11}$$

则得到 DM（Diebold-Mariano）统计量：

$$S_1 = DM = \left[\hat{V}(\bar{d})\right]^{-1/2}\bar{d} \tag{3-12}$$

其中 $\hat{V}(\bar{d})$ 有公式（3-10）和公式（3-11）得到。在零假设下，DM 统计量服从渐进标准正态分布（证明见[91]）。也即：

$$DM \sim N(0,1) \tag{3-13}$$

学者 Harvey, Leybourne 和 Newbold[92]给出了更严谨的修正统计量，即 DM 检验的修正检验，也称为 HLN 检验。

$$S_1^* = HLN = \left[\frac{n+1-2h+n^{-1}h(h-1)}{n}\right]^{1/2} S_1 \tag{3-14}$$

并得出 HLN 服从自由度为 $(n-1)$ 的 T 分布（证明见[92]）。即

$$S_1^* = HLN \sim t(n-1) \tag{3-15}$$

现在常用的 DM 检验一般指 DM 检验的修正，即 HLM 检验。本文 DM 统计量的检验也是使用 HLM 检验值。

WS（Wilcoxon signed-rank test，威尔科克森符号秩检验）检验，也是一种非参数假设检验方法，它成对的检查 2 个数据集中的数据（即 paired difference test），判断 2 个数据集是不是来自相同的分布总体。如果可以得到精确的有限样本检验，则可以根据零中值损失差的零假设检验，检验观察到的损失差，即有 d_t 的中位数为零。如果采样的损失差异具有对称的分布，则零假设与上述 DM 检验一致，表示相同的准确度[1]。WS 统计检验值定义如下：

$$WS = \min\left(\sum_{t=1}^{n} I_+(d_t)rank(|d_t|),\ \sum_{t=1}^{n} I_-(d_t)rank(|d_t|)\right) \tag{3-16}$$

$$I_{+}(d_t)=\begin{cases}1，d_t>0\\0，其他\end{cases} \quad (3\text{-}17)$$

$$I_{-}(d_t)=\begin{cases}1，d_t<0\\0，其他\end{cases} \quad (3\text{-}18)$$

其中 d_t 如公式 3-8 所示，n 为样本数量

3.3.4 实证结果分析

实证研究数据是 3.3.1,3.3.2 小节给出的数据，测试数据预测评价指标结果如表 3-3 与表 3-4 所示。关于模型自变量的选择，文献[62]分别选择了窗口大小为 3,4,5 的滞后变量数据作为自变量进行建模分析。本文为了研究模型之间的差异，分别选择了窗口大小为 3-9 的滞后变量（表格中 Lag 代表滞后变量的个数）。模型结构是一个输入层、一个隐藏层和一个输出层。输出层节点为 1 即向前一步预测，在 ANN 和 EMD-ANN 模型中网络隐藏层节点数我们选择经验值 128。在 S-EMD-ANN 中由于其输入变量远多于 ANN 和 EMD-ANN，所以我们选择经验节点数 256。并且对于所有的模型都在输入层和隐藏层加入 Dropout 层，并根据实验法得出当值为 0.4 模型表现更好（对比了 0.3、0.4 和 0.5）。输入层节点对于 ANN 模型和 EMD-ANN 模型来说为滞后变量（Lag）的个数，对于 S-EMD-ANN 模型来说输入节点个数为滞后变量与子序列的乘积（Lag*m）。为了增加预测结果的直观性，实验结果以箱线图的形式对比给出，如图 3-10 和图 3-11。

表 3-3 和表 3-4 中的模型和数据说明：ARIMA 模型的平稳性和白噪声检验结果，以及对应的自回归项数 p 和移动平均项数 q 见附录 A，其中 p 、 q 值的确定依据 AIC 信息准则选取。ANN、EMD-ANN 和 S-EMD-ANN 中每个窗口大小下（自变量 Lag 的值）对应的 MAPE(%)、MAE、RMSE、T 评估指标值为 20 次实验的平均值。Mean 列表示所有选取的窗口下模型表现评估指标的平均值，即 140（20*7）次实验的预测结果评估指标的均值。表中 E-A 表示 EMD-ANN 模型，S-E-A 表示 S-EMD-ANN 模型，（N，T）表示模型个数和训练模型所需时间（分钟）。

从表 3-3、表 3-4 和图 3-10、图 3-11 我们可以得出以下结论：

结论 1，对于上证指数和标准普尔 500 指数时间序列数据集来说，ANN、EMD-ANN、S-EMD-ANN 模型的预测结果要好于 ARIMA 模型，以预测精度评估指标 MAPE 为例，在上证指数数据集上分别改进 55.8%、77.9%、78.3%，在标准普尔 500 指数数据集上分别改进 22.2%、60.5%，61.2%。所以可以知道非线性模型 ANN、EMD-ANN、S-EMD-ANN 模型优于线性模型 ARIMA 模型。这一点和大多数学者的研究结果基本相同。

表 3-3 不同模型下上证指数的预测结果评价

Model		Lag							
		3	4	5	6	7	8	9	Mean
ARIMA	MAPE				3.120				
	MAE				89.90				
	RMSE				117.2				
ANN	MAPE	1.381	1.377	1.378	1.377	1.378	1.381	1.378	1.379
	MAE	39.58	39.45	39.49	39.44	39.49	39.57	39.47	39.50
	RMSE	52.82	52.65	52.70	52.62	52.68	52.81	52.65	52.70
	(1, T)	0.071	0.072	0.074	0.076	0.071	0.074	0.074	0.073
E-A	MAPE	0.687	0.673	0.690	0.688	0.695	0.696	0.696	0.689
	MAE	19.77	19.39	19.88	19.83	20.06	20.09	20.07	19.87
	RMSE	26.04	25.95	26.45	26.30	26.44	26.53	26.42	26.30
	(13, T)	0.943	0.932	0.963	1.002	0.968	0.956	0.954	0.96
S-E-A	MAPE	0.673	0.684	0.683	0.678	0.676	0.671	0.673	0.677
	MAE	19.39	19.72	19.68	19.53	19.50	19.40	19.46	19.53
	RMSE	25.29	25.59	25.52	25.91	26.06	26.01	26.27	25.81
	(1, T)	0.082	0.085	0.089	0.094	0.095	0.096	0.097	0.091

表 3-4 不同模型下标准普尔 500 指数的预测表现结果

Model		Lag							
		3	4	5	6	7	8	9	Mean
ARIMA	MAPE				1.334				
	MAE				35.98				
	RMSE				48.96				
ANN	MAPE	1.039	1.036	1.039	1.038	1.035	1.038	1.037	1.038
	MAE	28.06	27.98	28.05	28.02	27.96	28.04	28.02	28.02
	RMSE	39.86	39.75	39.81	39.79	39.68	39.78	39.76	39.78
	(1, T)	0.072	0.073	0.072	0.072	0.072	0.072	0.072	0.072
E-A	MAPE	0.519	0.528	0.529	0.525	0.527	0.528	0.530	0.527
	MAE	14.02	14.27	14.28	14.19	14.24	14.27	14.32	14.23
	RMSE	19.38	19.88	19.81	19.79	19.81	19.88	19.89	19.78
	(12, T)	0.868	0.869	0.873	0.868	0.869	0.870	0.876	0.879
S-E-A	MAPE	0.511	0.514	0.516	0.521	0.523	0.518	0.524	0.518
	MAE	13.84	13.93	13.96	14.11	14.16	14.04	14.19	14.03
	RMSE	19.21	19.10	19.12	19.28	19.45	19.40	19.64	19.31
	(1, T)	0.083	0.086	0.087	0.089	0.091	0.094	0.095	0.089

结论 2，正如大多数文献研究的结论一样，在本文的数据集上 EMD-ANN 模型预测结果优于单独的 ANN 模型。以 MAPE 评估指标为例，在上证指数数据集上改进 50.0%（(1.379-0.689)/1.379），在标准普尔 500 指数数据集上改进 49.2%。本文所提出的 S-EMD-ANN 也同样优于单独的 ANN 模型，表现评估指标以

MAPE 为例，在上证指数数据集上改进 50.9%，在标准普尔 500 指数数据集上改进 50.1%，同时我们也发现单从结果来看本文所提出的 EMD-ANN 的改进模型 S-EMD-ANN 的表现结果略优于 EMD-ANN 的表现结果。

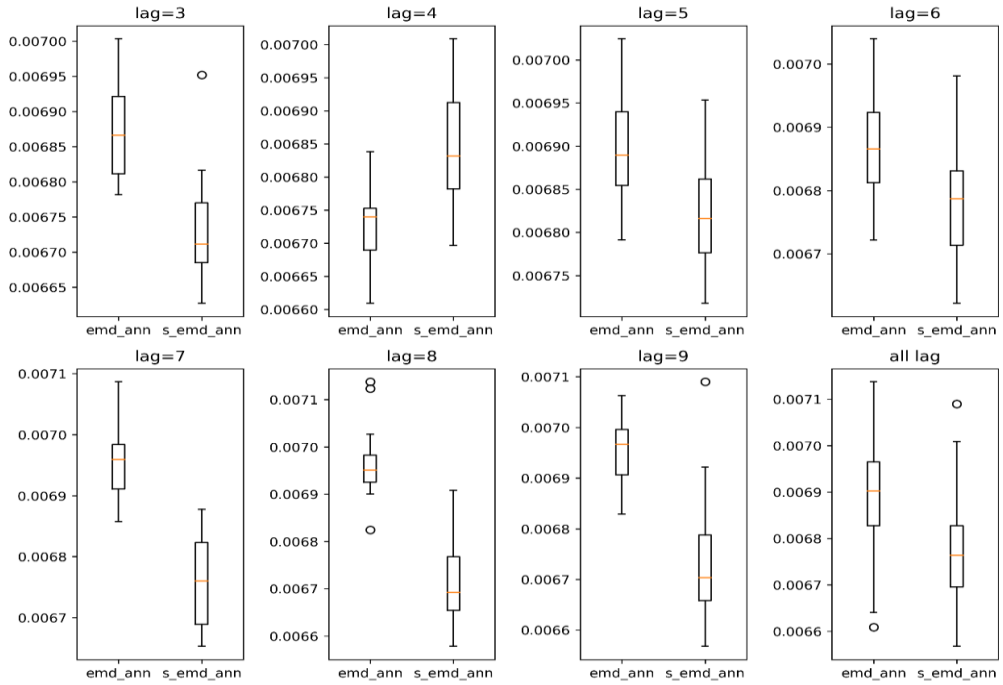


图 3-10 上证指数数据中 EMD-ANN 与 S-EMD-ANN 模型的 MAPE 指标值

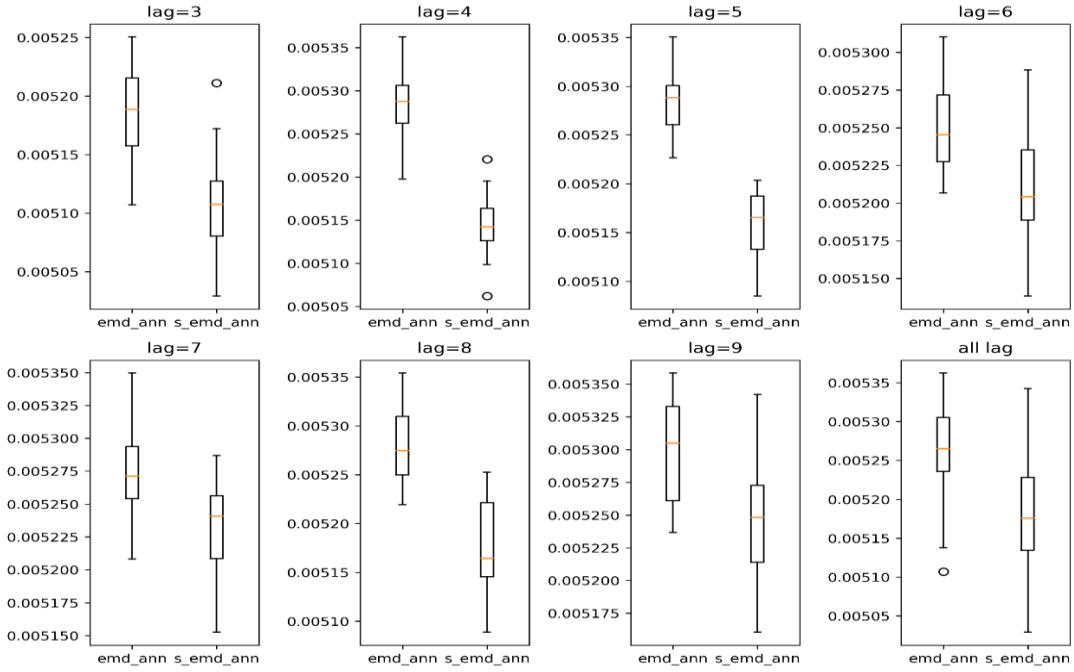


图 3-11 标准普尔 500 指数数据中 EMD-ANN 与 S-EMD-ANN 模型的 MAPE 指标值

表 3-5 上证指数模型差异性检验 DM 与 WX 检验 p 值

S-E-A vs E-A		Lag							
		3	4	5	6	7	8	9	Mean
APE	DM	0.377	0.713	0.633	0.480	0.186	0.099	0.103	0.282
	WX	0.554	0.390	0.708	0.824	0.344	0.166	0.349	0.511
AE	DM	0.416	0.705	0.626	0.469	0.191	0.111	0.118	0.296
	WX	0.623	0.394	0.707	0.846	0.338	0.180	0.358	0.552
SE	DM	0.225	0.560	0.189	0.507	0.426	0.328	0.520	0.299
	WX	0.421	0.520	0.885	0.964	0.526	0.305	0.761	0.403

附注：S-E-A 指 S-EMD-ANN 模型，E-A 指 EMD-ANN 模型。

表 3-6 标准普尔 500 指数模型差异性检验 DM 与 WX 检验 p 值

S-E-A vs E-A		Lag							
		3	4	5	6	7	8	9	Mean
APE	DM	0.435	0.247	0.244	0.630	0.527	0.307	0.435	0.281
	WX	0.943	0.804	0.819	0.706	0.893	0.688	0.727	0.835
AE	DM	0.476	0.281	0.267	0.674	0.580	0.339	0.477	0.313
	WX	0.940	0.790	0.819	0.696	0.885	0.699	0.732	0.837
SE	DM	0.626	0.135	0.161	0.274	0.343	0.275	0.482	0.220
	WX	0.791	0.970	0.993	0.538	0.704	0.976	0.995	0.986

结论 3，本章节主要的研究内容。从图 3-10 和图 3-11 我们可以看出除了在上证指数数据上自变量数量（Lag=4）等于 4 时 S-EMD-ANN 模型的 MAPE 值大于 EMD-ANN 模型的 MAPE 值。在其他选取的窗口下 S-EMD-ANN 模型的 MAPE 值都要小于 EMD-ANN 模型的 MAPE 值，从附录 B 中的 MAE 和 RMSE 指标值也可以得出类似的结果。整体预测结果，在 MAPE 评价指标上 S-EMD-ANN 比 EMD-ANN 在上证指数和标准普尔 500 指数分别改进了 1.74%，1.71%，在 MAE 评价指标上分别改进了 1.71%，1.41%，在 RMSE 评价指标数上分别改进了 1.86%、2.38%。虽然本文提出的改进模型在预测精度上提升比较小，但是在效率上却有很大的提高。从表 3-3 和表 3-4 中我们看出，在上证指数数据上本文所提出的 S-EMD-ANN 只需要 1 个模型而一般的 EMD-ANN 模型需要 13（与 EMD 分解得到的子序列数相等）模型,而且模型训练时间减少了 90.5%，在标准普尔 500 指数数据上 S-EMD-ANN 只需要 1 个模型而一般的 EMD-ANN 模型需要 12 个模型，模型训练时间减少了 89.9%。单个模型还有一点优势在于像 ANN 等一些学习类模型往往需要一些超参数需要去搜索确定，模型数量的减少也大大的提高了超参数择优的效率。为了更进一步说明两个模型结果的差异，我们对两个模型的预测结果进行差异性检验，利用 DM 检验和 WX 检验检验两个模型的结果差异性是否显著。假设检验结果如表 3-5 与表 3-6 所示，从表中数据可以看出无论对于绝对百分比误差(APE)、绝对误差(AE)、还是方差(SE)，S-EMD-ANN 和 EMD-ANN

两种模型在任何窗口下的预测结果都没有显著的差异，假设检验的 p 值都远大于 0.05，所以我们可以认为 S-EMD-ANN 模型的预测结果和 EMD-ANN 模型的预测结果没有显著性的差异。从前面的表述也可以看出 S-EMD-ANN 模型的预测结果相较于 EMD-ANN 模型的预测结果在三种评估指标上都仅仅改进了 1.5%左右。但从模型的训练时间和模型数量上看 S-EMD-ANN 模型远优于 EMD-ANN 模型。故本文认为，该章节提出的改进模型在预测精度上有所提高但提高效果不明显，但在模型的结构和运算效率上有极大的改进。

3.3.5 同类研究对比

为了进一步研究本文所提出模型的预测能力，本小节选取了一些文献的研究内容，并利用文献中的数据使用本文的预测模型得出预测结果，并与文献结果进行对比研究。本小节的模型参数同 3.3.4 小节相同，模型预测结果为 20 次实验的均值。

表 3-7 上证指数同类预测模型研究结果

上证指数	Zhou (2019) [62] EMD-FNN			本文模型结果（S-EMD-ANN）		
Lag	MAPE	MAE	RMSE	MAPE	MAE	RMSE
3	*0.0123*	*37.1778*	*61.5138*	0.0074	21.83	31.45
4	*0.0143*	*43.5513*	*66.7576*	0.0076	22.42	32.11
5	*0.0152*	*54.8499*	*89.9065*	0.0075	22.35	32.74

附注：斜体部分数据选自对应的参考文献。

表 3-8 标准普尔 500 指数同类预测模型研究结果

标普 500	Zhou (2019) [62]EMD-FNN			本文模型结果（S-EMD-ANN）		
Lag	MAPE	MAE	RMSE	MAPE	MAE	RMSE
3	*0.0105*	*13.0396*	*17.6591*	0.00812	10.09	13.67
4	*0.0122*	*15.1386*	*20.3978*	0.00831	10.33	13.91
5	*0.0130*	*16.1700*	*22.1277*	0.00843	10.47	14.09

表 3-7 和表 3-8 中斜体数据选自 Zhou(2019)[62]的研究结果其中表 3-7 为上证指数研究结果，数据为 2012 年 1 月 4 日到 2016 年 12 月 30 日，共 1214 个数据点，表中数据为最后 242 个数据点的预测结果评价指标值。从表中的评价指标来看，本文所提出的模型的预测效果要远远优于文献中的预测结果。当自变量个数为 3 时（文献中选择的最优自变量数量），可以看出，MAPE 指标改进了 39.84%，MAE 指标改进了 41.28%，RMSE 指标改进了 48.87%。表 3-8 为标准普尔 500 指数研究结果，数据为 2007 年 1 月 3 日到 2011 年 12 月 30 日，共 1260 个数据点，

最后 252 个数据点的预测结果评价指标值。从表中的评价指标来看，本文所提出的模型的预测效果要优于文献中的预测结果，当自变量的个数为 3 时（文献[62]中选择的最优窗口），可以得到 MAPE 指标改进了 22.67%，MAE 指标改进了 22.62%，RMSE 指标改进了 22.59%。同时从本文模型结果中，我们也可以进一步证明文献[62]所选的自变量窗口大小时，确实是当自变量个数等于 3 时模型的表现结果较好。

表 3-9 沪深 300 股指期货指数同类研究结果

预测模型	史(2015)[86]				本文模型
	GARCH	ARIMA	RBF(ANN)	EMD-RBF(ANN)	S-EMD-ANN
MAE	*22.8945*	*20.1335*	*18.6743*	*15.8276*	11.09
RMSE	*30.2314*	*25.4589*	*24.7865*	*21.3992*	13.75

附注：斜体部分数据选自对应的参考文献。

表 3-9 中斜体数据为史(2015)[86]的研究结果，其研究的是对沪深 300 股指期货的预测。数据为 2012 年 5 月 22 日到 2014 年 9 月 9 日，表中数据为后 61 个数据点的预测结果评价指标。文献[86]选择的窗口大小也为 3 但是没给出理由，为了进行对比研究表中本文的模型也选择窗口大小为 3 时的结果。从表中的结果评价指标来看本文所提出的模型的预测效果要优于文献中的预测结果，其中 MAE 指标改进了 29.93%，RMSE 指标改进了 35.74%。

与其他学者研究的结果对比发现，本文提出的改进模型预测效果远高于预期结果。分析其原因，可能是由于 3.3.2 小节中对数据进行了差分处理。3.3.2 小节只给出了一些初步的研究，未来可能是研究方向之一。

3.4 本章小结

本章主要提出了一种基于 EMD-ANN 预测模型改进得到的 S-EMD-ANN 预测模型，通过大量的实证研究并结合对比其它文献研究的研究结果我们可以得到以下结论。首先，本文所提出的 S-EMD-ANN 模型在模型的数量以及复杂程度是上都远小于 EMD-ANN 模型，并且 S-EMD-ANN 模型在训练是需要的时间远小于 EMD-ANN 训练需要的时间，所以 S-EMD-ANN 的效率更高。其次，从两种模型的结果差异性检验上我们可以看出两种预测模型的结果没有显著性差异。但是在各个评价指标上我们可以看出 S-EMD-ANN 预测模型的结果要略优于 EMD-ANN 预测模型的结果。所以我们可以得出结论 S-EMD-ANN 预测模型的预测结果要比 EMD-ANN 预测结果要好，但结果改进不是很大，从 3.3.4 小节我们知道在各个指标上的改进大约在 1.5%左右。再者，在 3.3.5 小节中在与其它学者的研

究成果进行对比时，发现 S-EMD-ANN 模型的预测结果比其它学者的研究结果改进了 20%-50%之间，说明本文所提出的模型在其他数据上也能有良好的表现结果。

4 基于 EMD 与神经网络的自适应预测模型

4.1 EMD 算法中的前瞻性偏差分析

“前视性”偏差（Look-ahead bias[70]）也有称为“前瞻性”偏差，可以定义为对直到稍后的日期才可用信息的无意使用。换句话说，就是无意中用到了未来的信息预测未来。Mahfoud,Mani(1996)[93]指出，商业和公开的金融数据可能从一开始就含有前瞻性偏见。例如，与政府经济指标相关的数据在审查过程，可能因为未来的某项指标而修改过去的数据[70]。

除了人为的向数据本身添加的前瞻性偏差之外，使用一些技术手段对数据进行预处理时也有可能引入前瞻性偏差。例如，数据标准化时会引入前瞻性偏差，数据标准化是非常流行的数据预处理方式，同时也是处理时间序列的常用方式。Min-max 和 Z-score 标准化，可以说是两种最流行的标准化技术。他们都是利用一些数据的统计信息对数据进行变换处理。然而，对于一个数据集来说最大值，最小值以及标准差都是整个数据的统计信息，特别是大部分研究在计算这些统计信息时是用了个数据集来计算的。那么，对于我们要用来测试的数据来说，我们在预测之前就已经知道了它的最大值、最小值或者标准差的情况，这显然是引入了前瞻性偏差。该偏差可能对具有一定统计特征的数据影响不大，但在对一些复杂的时间序列来说可能影响就会偏大，特别是一些具有一定周期性和波动性的数据。在经验模态分解中也存在这样的前瞻性偏差，而且这种前瞻性偏差影响比较大。然而不幸的是，在诸多文献的研究工作中，这种性质的偏差经常被忽视。而且由于 EMD 算法的固有特性，在使用经验模态分解算法时会使得混合模型中含有大量的前瞻性偏差，这种偏差对模型的评价会产生很大的影响。在有关 EMD 混合模型的文献中包含前瞻性偏差的测试结果似乎很普遍[70]，这也使得诸多学者的研究结果都看起来很“漂亮”。

正如 N. Sun (2018)[65] 、Q. Tan(2018)[66] 和 X. Zhang(2015)[69]提出的问题，现有的大部分文献[57-64]对于数据集划分都存在一定的瑕疵，现有的文献在处理数据是都会忽略掉“前瞻性”偏差。现在有大部分文献在划分数据时，都是将整个时间序列进行经验模态分解，然后将分解后的序列划分为训练集和测试集。然而，这样的序列分解方式使用了未来时间点的信息，所以不是真实的预测。如图 4-1 所示，对 T1 点的值分解使用了极值点 T2 和 T3 的值，而 T2 和 T3 对于 T1 来说属于未来的信息。所以如果我们要真实预测 T2 时刻的值那么对序列分解时只能用 T2 时刻之前的历史值。在文献[66]中学者 Q. Tan 将涉及未来信息的实证实验

称为“后视实验”（Hindcast experiment[66]），把不涉及未来信息的预测称为“预测实验”（Forecast experiment[66]）。在基于 EMD 的一系列混合模型的研究中后视实验是一种及其理想化的实验，是一种不考虑未来极值点对结果影响下的实验。然而，绝大多数的研究者都停留在理想化的研究中，显然本文中第 3 章的实验也是理想化的实验是包含前瞻性偏差的。

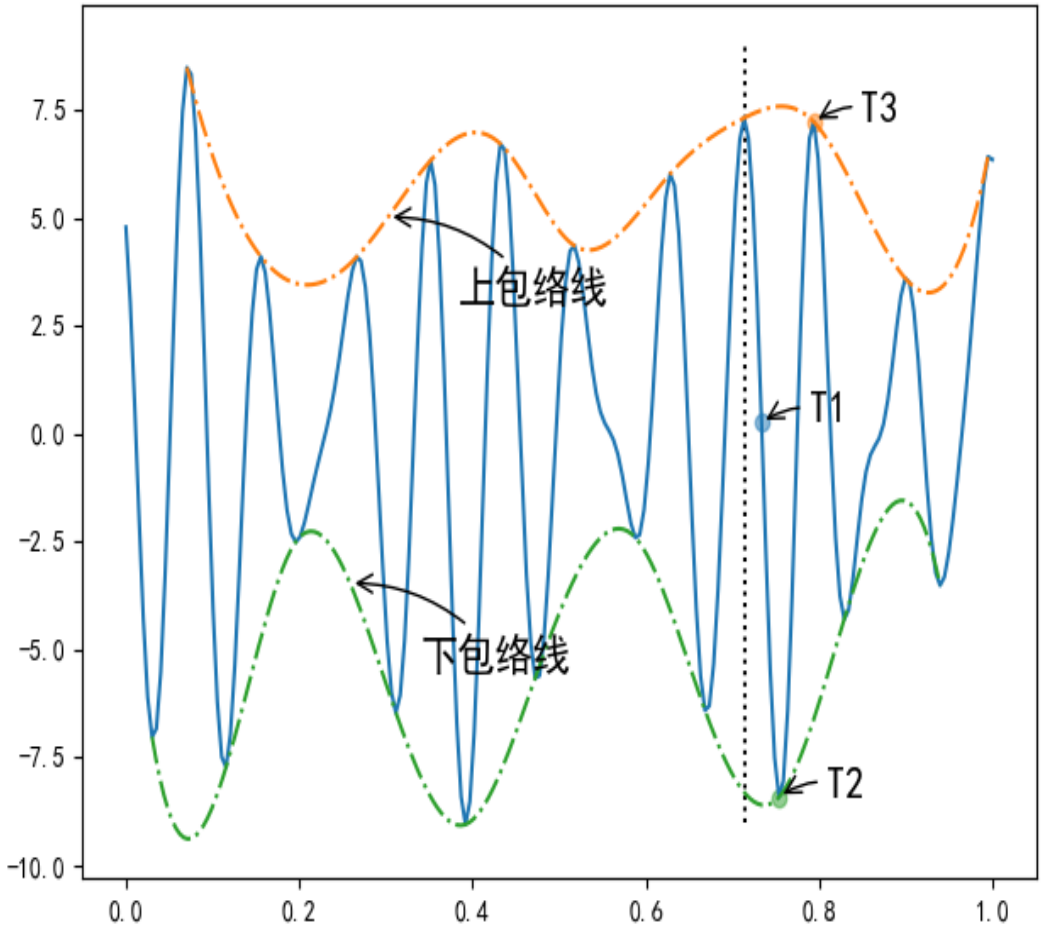


图 4-1 EMD 分解示意图

4.2 EMD 与 ANN 混合的自适应预测模型

4.2.1 基于 EMD 与 ANN 的自适应预测模型

为了解决基于 EMD 混合模型存在前瞻性偏差这一问题，N. Sun (2018)[65] 、Q. Tan(2018)[66]、Dennis(2017)[70]和 X. Zhang(2015)[69]学者在工程领域中都使用了一种自适应方法来构建基于 EMD 的一系列预测模型。预测模型基本流程如图 4-2 所示。并用 AEMD-ANN 表示该模型。

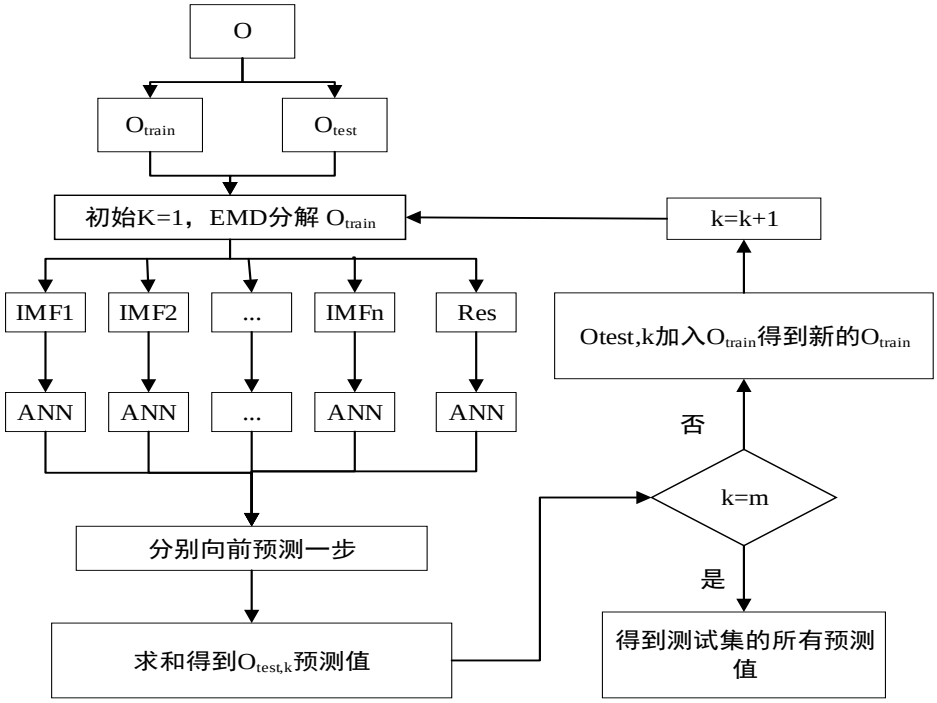


图 4-2 AEMD-ANN 预测模型流程图

图 4-2 所示 AEMD-ANN 算法的核心思想是：对于测试数据来说，当预测 T+1 时刻的值时，在分解时只使用 T+1 时刻之前的数据，避免使用到未来的极值信息。再把 T+1 时刻的值添加到训练数据中重新分解数据，重新训练模型，然后再预测 T+2 时刻的值，依次类推直到所有的测试数据被预测出来。自适应的意思就是每增加一个值，都要重新训练模型的参数，使得模型能够学习到最新数据的特征。其算法流程如下：

Step1：原始数据序列O分解为训练集O_{train}和测试集O_{test}，并初始化 k=1；

Step2：使用 EMD 算法把原序列分解成 n 个 IMF 和一个余项；

Step3：对每一子序列建立 ANN 模型，用子序列的前l项的值，预测第$l+1$项的值，具体变量预测方式如图 3-3 所示。

Step4：每个模型向前预测一步，预测结果求和得到测试数据集的第 k 项的值$O_{test,k}$的预测值；

Step5：如果 k=m，则输出所有测试数据的预测值，否则，将$O_{test,k}$加入O_{train}得到新的训练数据，并令 k=k+1 重复 Step2-Step5。

4.2.2 基于 EMD 与 ANN 自适应预测模型的改进

根据 3.2 章节的分析，我们知道传统的 EMD 与 ANN 混合模型，存在模型结构复杂，模型数量大，模型训练效率低等问题，这些问题在自适应模型中同样存在并且更加严重。因为，对于测试数据的每个数据点都要重新分解数据和训练模型，所以可想而知传统的自适应混合模型的计算量是非常庞大的，在后面的实

证实验结果中这个问题也充分的显现了出来。基于这些问题我们同样给出 EMD 与 ANN 混合的单一模型预测框架，如图 4-3 所示。并用 S-AEMD-ANN 表示该模型。

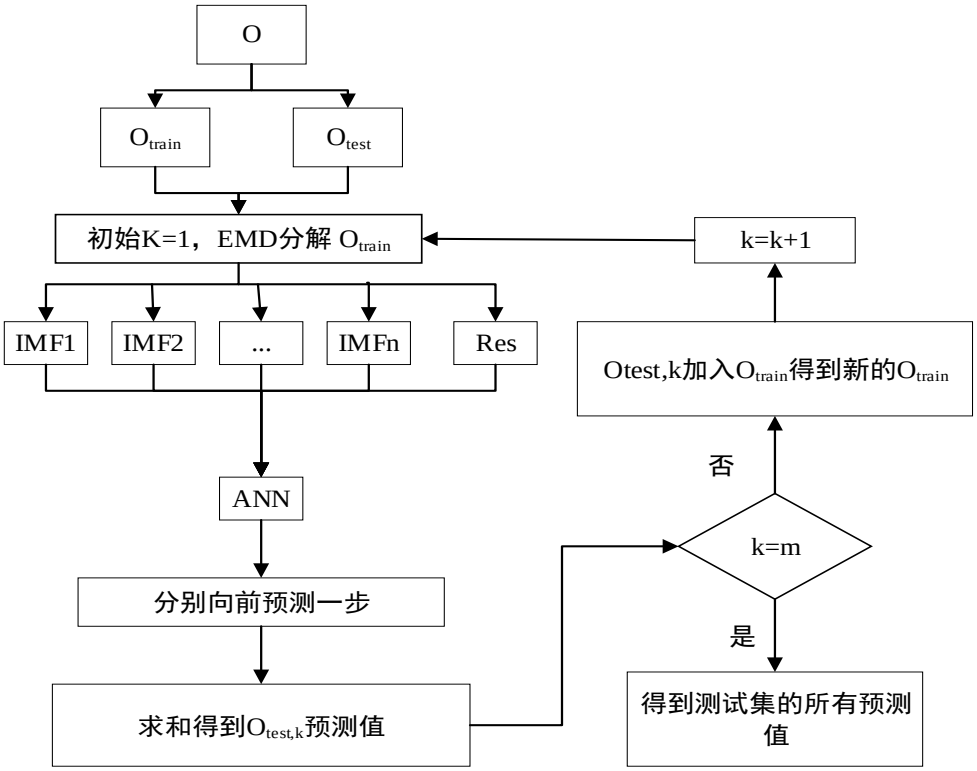


图 4-3 S-AEMD-ANN 预测模型流程图

S-AEMD-ANN 的算法和 AEMD-ANN 算法只有下面两步的差异：

Step3：对所有的子序列建立一个 ANN 模型，用每一个子序列的前 l 项序列值直接预测原始序列数据的第 $l+1$ 项的值，具体变量预测方式如图 3-5 所示。

Step4:向前预测一步得到测试数据集的第 k 项的值 $O_{test,k}$ 的预测值。

根据 4.3 的结论 3 我们知道，如果只是简单地使用自适应 EMD 分解去改善 ANN 的预测结果，虽然单一模型的预测结果优于多个模型的预测结果，但是和单独的 ANN 预测模型相比结果比较不理想。根据检验模态分解算法的原理我们可以知道以下因素是使得预测结果不理想的原因之一：对于训练数据来说每一个 t 时刻的数据点，用来预测 t 时刻数据的自变量（$t-1, t-2, \cdots, t-Lag$ 时刻的数据）都是真实的，也就是说这些自变量都是根据真实的数据分解出来的数据，是存在前瞻性误差的。然而，对于测试数据来说，当我们在测试数据集中预测 t 时刻的数据时，我们用来分解时间序列的数据点只能使用到 t-1 时刻，那么可想而知我们不知道 t-1 时刻的数据是不是极值点也不知道下一个极值点在哪一时刻，所以只能依据延拓等手段来预估未来的极值点以完成数据的分解。那么在这种情况下

就会得到一些虚假的分解值，而且时间点越靠近 t 时刻的数据失真越严重，当极值点的个数越少越稀疏，失真数据点就越多（这也是 3.3.2 小结对数据差分来增加极值点的一个重要原因）。因此，我们使用真实分解数据训练得到的参数在应用于测试数据的失真分解数据时有可能会给测试的预测结果带来巨大的误差。由于测试数据集中存在虚假的分解数据，因此本文对 AEMD-ANN 和 S-AEMD-ANN 模型进行简单的改进，算法改进如下：

AEMD-ANN 算法的改进，并将改进后的模型记为 AEMD-ANN^a。

Step3：对每一子序列建立 ANN 模型，用$l-k$(删除k项，距离预测值较近的滞后变量)项序列值，预测第$l+1$项的值，具体变量预测方式如图 4-4 所示。

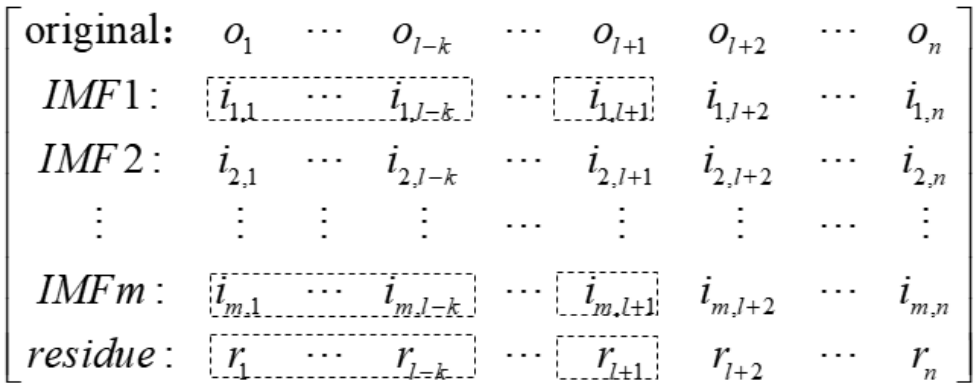


图 4-4 AEMD-ANN^a 模型变量预测示意图

S-AEMD-ANN 算法的改进，并将改进后的模型记为 S-AEMD-ANN^a。

Step3：对所有的子序列建立一个 ANN 模型，用$l-k$(删除k项，距离预测值较近的滞后变量)项序列值直接预测第$l+1$项的值，具体变量预测方式如图 4-5 所示。

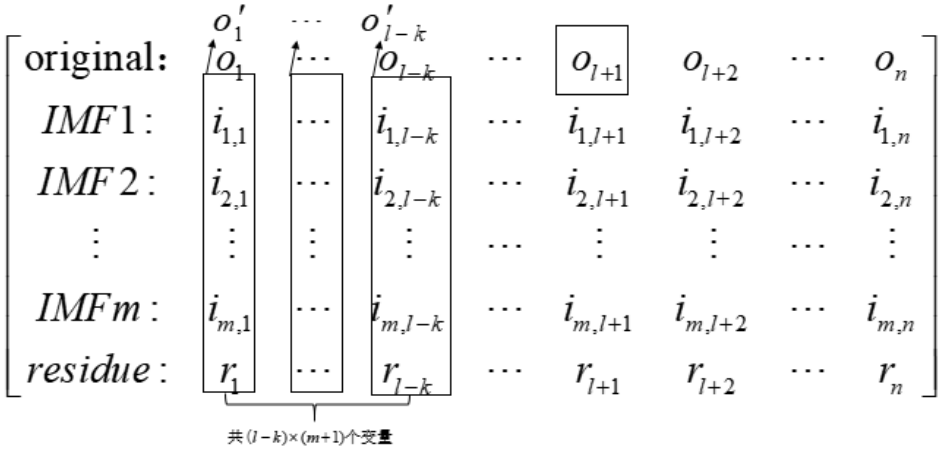


图 4-5 S-AEMD-ANN^a 模型变量预测示意图

改进的 AEMD-ANN[a] 和 S-AEMD-ANN[a] 模型主要依据以下两个假设：

首先，对于一些失真的数据点我们选择对其进行直接删除处理，尽量选择失真较小的数据点作为自变量，从而减少失真数据点对预测结果的影响。这显然是合理的。

其次，对于所要预测的时间点的数据来说，我们假设它与多个滞后时间点的数据相关，在一定程度上可以舍弃一些可能严重失真的滞后变量，来减少失真数据对预测结果的影响。我们从表 4-1 可以看出这种假设是完全合理的。从表 4-1 中我们看出，上证指数原数据和标准普尔 500 指数原数据 T 时刻的数据值与滞后 9 个变量的数据值相关系数相差不大，都与 T 时刻的数据值具有很大的相关性。对于本文使用的差分后的序列来说，相关性也不仅仅是依据距离 T 时刻远近来决定的，例如在上证指数中 T 时刻与 T-1、T-2 时刻的相关系数分别为 0.036、0.037，而与 T-3、T-4 时刻数据的相关系数分别为 0.045、0.072。所以在一定程度上牺牲一些失真的数据点来换取预测结果的准确性是可取的。而且，从 4.3 的结论 4 我们也可以看出，这种做法对于改进模型来说是可取的。

表 4-1 数据自相关系数

数据	T	T-1	T-2	T-3	T-4	T-5	T-6	T-7	T-8	T-9
SZ-A	1	0.999	0.999	0.998	0.997	0.996	0.995	0.994	0.993	0.993
SZ-B	1	0.036	0.037	0.045	0.072	0.004	0.052	0.021	0.007	0.009
SP-A	1	0.999	0.999	0.999	0.999	0.999	0.999	0.998	0.998	0.998
SP-B	1	0.048	0.037	0.007	0.017	0.038	0.004	0.008	0.017	0.014

附注：SZ-A 表示上证指数序列的原序列，SZ-B 表示上证指数序列差分后的序列。SP-A 表示标准普尔 500 指数序列的原序列，SP-B 表示标准普尔 500 指数差分后的序列。

4.3 实证分析

实证研究数据依据 3.3.1,3.3.2 小节给出的数据。模型自变量的选择，本文分别选择了窗口大小为 5-11 的滞后变量（表中 Lag=5,…,11）。模型结构和参数设置与第三章相同。测试数据预测表现结果如表 4-2 和表 4-3 所示。并且，为了更直观的呈现所有实验的结果，以箱线图来对比研究所有实验预测结果的 MAPE 值，如图 4-6 和图 4-7 所示。

表 4-2 和表 4-3 中的数据说明：ARIMA 模型的时间序列数据平稳性和白噪声检验结果以及模型对应的自回归项数 p 和移动平均项数 q 见附录 A，其中 p、q 值的确定依据 AIC 信息准则选取。ANN、AEMD-ANN(AE-A)、S-AEMD-ANN(S-AE-A)、AEMD-ANN[a](AE-A[a])、S-AEMD-ANN[a](S-AE-A[a])预测模型中，每个窗口下（自变量 Lag 的值）对应的 MAPE(%)、MAE、RMSE、T（时间）评估指标值

是 20 次实验的平均值。此外，对于改进模型 AEMD-ANN[a]、S-AEMD-ANN[a]，实验过程中对比了 k=1 和 k=2 的预测结果，当 k 的值取 2 时模型预测精度较好，所以，本文选择删除两个自变量即令 k=2。Mean 列表示的是所有选取的窗口下的模型表现评估指标的平均值即 140（20*7）次实验的预测结果评估指标的均值。由于计算量的限制，本章节的测试数据选择了大约两个月的交易数据共 50 个数据点作为测试数据，也即是留下最后 50 个数据点为测试数据集。

表 4-2 不同自适应模型下上证指数的预测表现结果

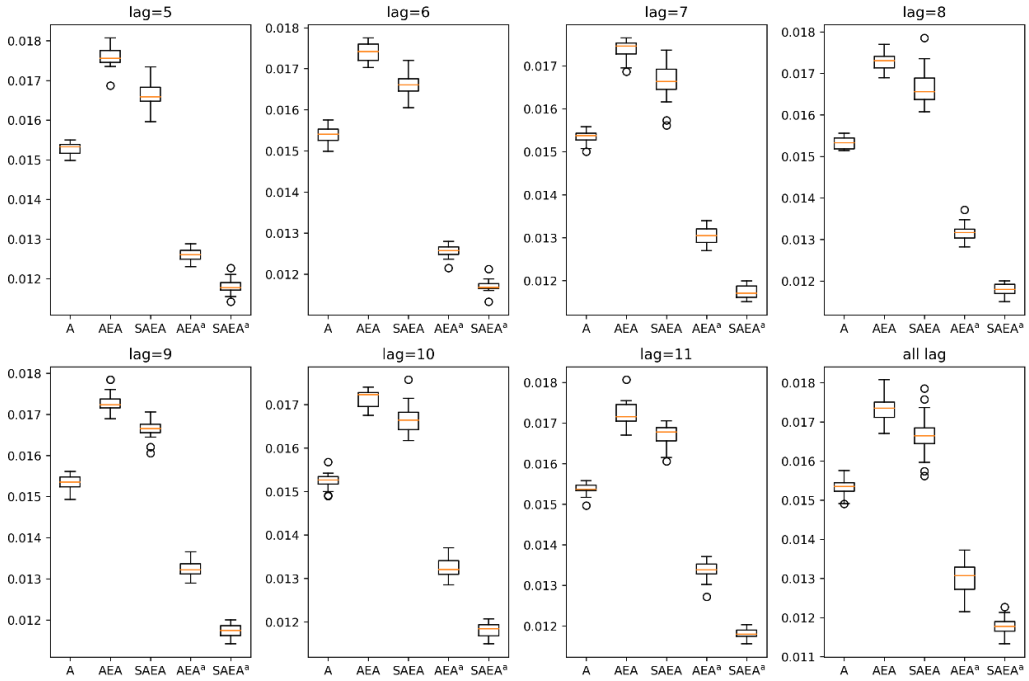
Model		Lag							
		5	6	7	8	9	10	11	Mean
ARIMA	MAPE				3.323				
	MAE				96.99				
	RMSE				130.81				
ANN	MAPE	1.529	1.540	1.535	1.532	1.534	1.525	1.537	1.533
	MAE	44.48	44.79	44.64	44.57	44.62	44.35	44.70	44.59
	RMSE	61.97	63.36	62.13	62.07	62.15	61.81	62.24	62.10
	(1, T)	0.072	0.072	0.072	0.072	0.072	0.078	0.075	0.073
AE-A	MAPE	1.759	1.740	1.737	1.728	1.727	1.713	1.724	1.733
	MAE	51.37	50.82	50.77	50.49	50.45	50.04	50.37	50.62
	RMSE	70.98	70.11	70.39	69.80	69.42	69.33	69.83	69.98
	(13*, T)	51.45	52.10	51.56	51.68	52.00	51.71	52.16	51.81
S-AE-A	MAPE	1.664	1.662	1.664	1.666	1.663	1.669	1.670	1.665
	MAE	48.55	48.48	48.56	48.61	48.48	48.69	48.70	48.58
	RMSE	65.17	65.22	65.30	65.61	65.20	65.54	65.05	65.30
	(1, T)	5.601	5.797	5.617	5.824	5.619	5.599	5.787	5.692
AE-A[a]	MAPE	1.262	1.254	1.305	1.316	1.323	1.323	1.337	1.303
	MAE	36.67	36.41	37.87	38.21	38.47	38.44	38.87	37.85
	RMSE	49.96	52.06	53.67	54.23	54.19	54.67	55.43	53.46
	(13*, T)	49.81	47.05	48.61	48.24	48.24	48.48	47.65	48.33
S-AE-A[a]	MAPE	1.181	1.172	1.173	1.180	1.174	1.181	1.181	1.177
	MAE	34.26	33.97	34.02	34.23	34.06	34.26	34.26	34.15
	RMSE	49.03	48.72	48.96	48.87	48.74	49.21	49.10	48.94
	(1, T)	5.344	4.983	5.057	5.136	5.097	5.104	5.106	5.118

附注：*表示的模型个数是第一次进行经验模态分解的值，因为每次加入新变量都要从新分解，故只给出大约的模型个数。

从表 4-2、表 4-3 以及图 4-6、图 4-7 我们可以得出以下结论：

结论 1，从表 4-2 和图 4-6 我们可以看出，上证指数数据集上 ANN，AEMD-ANN，S-AEMD-ANN，AEMD-ANN[a] 和 S-AEMD-ANN[a] 模型的预测结果都优于 ARIMA 模型的预测结果。然而，从表 4-3 和图 4-7 中我们可以看出，标准普尔

500 数据集上只有 ANN 模型和 S-AEMD-ANN^a 的预测结果优于 ARIMA 模型的预测结果。所以我们认为只有 ANN 和 S-AEMD-ANN^a 模型的预测结果是优于线性模型 ARIMA 模型的预测结果，而其他几种模型在剔除前瞻性偏差之后，没有理由认为模型表现性能比线性模型 ARIMA 模型好。



附注：ARIMA 模型对应的 MAPE 值过大，为了增加其他模型的对比效果故没在图中标示

图 4-6 上证指数数据中剔除前瞻性偏差后模型的 MAPE 指标值

结论 2，对比表 4-2，表 4-3 和图 4-6，图 4-7 的 AEMD-ANN 和 S-AEMD-ANN 以及 AEMD-ANN^a 和 S-AEMD-ANN^a 模型的预测结果，再次验证了 3.3.4 小节结论 3 的正确性，即本文所提出的单一混合预测模型要优于传统的多个模型混合预测模型。以 MAPE 评价指标为例，S-AEMD-ANN 模型比 AEMD-ANN 在上证指数数据集上平均改进了 3.92%，在标准普尔 500 数据集上平均改进了 11.02%，S-AEMD-ANN^a 模型比 AEMD-ANN^a 在上证指数数据集上平均改进了 9.67%，在标准普尔 500 指数数据集上平均改进了 20.77%。在模型数量上和模型训练效率上也有显著提高，在训练时间上，S-AEMD-ANN 模型比 AEMD-ANN 在上证指数数据集上平均减少了 89.01%，在标准普尔 500 指数数据集上平均减少了 89.71%，S-AEMD-ANN^a 模型比 AEMD-ANN^a 在上证指数数据集上平均减少了 89.41%，在标准普尔 500 指数数据集上平均减少了 90.20%。

结论 3，本章主要结论。对比表 4-2，表 4-3 和图 4-6，图 4-7 的 ANN 模型预测结果评价指标与 AEMD-ANN、S-AEMD-ANN 模型预测结果评价指标，我

们可以知道，如果剔除前瞻性偏差对结果的影响，那么 EMD 分解算法对 ANN 模型的预测结果不仅没有起到改善作用，反而使得预测结果比单独的 ANN 模型还要差。而且，结合附录 C 我们可以知道不管从 MAPE、MAE、RMSE 中任何一个评价指标来衡量模型结果的好坏，AEMD-ANN、S-AEMD-ANN 这两种模型预测结果都要差于单独的 ANN 模型预测结果。所以，如果剔除前瞻性偏差对预测结果的影响，那么我们没有理由认为在传统的混合方式下 EMD 分解算法对 ANN 预测模型有提升作用。并且，Dennis(2017)[70]在利用 EMD 和 SVM 以及 EMD 和 RF(Random Forest)预测走势的时候认为如果剔除前瞻性偏差 EMD-SVM 模型和 EMD-RF 模型的测结果不一定优于单独的 SVM 和 RF 模型。这些都说明，经验模态分解的前瞻性偏差在应用的时候对预测模型的影响是不能忽略的。

表 4-3 不同自适应模型下标普 500 指数的预测表现结果

Model		Lag							
		5	6	7	8	9	10	11	Mean
ARIMA	MAPE				0.936				
	MAE				25.91				
	RMSE				32.31				
ANN	MAPE	0.822	0.819	0.823	0.820	0.820	0.819	0.819	0.820
	MAE	22.64	22.57	22.66	22.60	22.61	22.57	22.58	22.60
	RMSE	28.47	28.39	28.50	28.41	28.42	28.37	28.40	28.42
	(1, T)	0.071	0.067	0.067	0.069	0.069	0.068	0.069	0.069
AE-A	MAPE	1.056	1.064	1.069	1.075	1.078	1.081	1.077	1.071
	MAE	28.91	29.12	29.26	29.39	29.47	29.55	29.44	29.31
	RMSE	37.02	37.34	37.53	37.78	38.19	38.05	38.11	37.72
	(12*, T)	41.81	41.85	42.04	42.54	42.17	42.28	42.83	42.22
S-AE-A	MAPE	0.960	0.952	0.954	0.941	0.959	0.953	0.954	0.953
	MAE	26.36	26.13	26.19	25.84	26.33	26.16	26.21	26.17
	RMSE	33.51	33.13	33.16	32.91	33.33	33.12	33.24	33.20
	(1, T)	4.332	4.308	4.319	4.445	4.316	4.323	4.356	4.343
AE-A[a]	MAPE	0.874	0.859	0.898	0.913	0.924	0.932	0.938	0.905
	MAE	23.97	23.60	24.63	25.03	25.34	25.55	25.73	24.84
	RMSE	32.07	31.60	33.63	34.28	34.74	35.02	35.13	33.78
	(12*, T)	42.21	42.50	42.38	42.65	42.53	42.72	44.06	42.72
S-AE-A[a]	MAPE	0.717	0.720	0.713	0.716	0.720	0.719	0.717	0.717
	MAE	19.77	19.84	19.65	19.74	19.83	19.82	19.76	19.77
	RMSE	24.69	24.75	24.54	24.63	24.71	24.77	24.67	24.68
	(1, T)	4.278	4.164	4.159	4.178	4.164	4.169	4.183	4.185

结论 4，本章的主要结论。从表 4-2、表 4-3 和图 4-6、图 4-7 中的 AEMD-ANN、S-AEMD-ANN 和 AEMD-ANN[a]、S-AEMD-ANN[a] 模型的表现结果看，本

文在 4.2.2 章节提出的模型改进方式有利于提高文献[65、66、69、70]中自适应模型的预测精度。以 MAPE 结果评价指标来看，AEMD-ANN^a 模型比 AEMD-ANN 模型在上证指数数据上和标准普尔 500 指数数据上分别改进了 24.81%（(1.733-1.303)/1.733）和 15.50%，S-AEMD-ANN^a 模型比 S-AEMD-ANN 模型在上证指数数据上和标准普尔 500 指数数据上分别改进了 29.31%和 24.76%。而且，图 4-6 和图 4-7 中可以看出，本文所提出的改进模型 S-AEMD-ANN^a 在两种数据集上预测表现性能都是最好的，并且最重要的是都优于单独的 ANN 模型，这说明经验模态分解算法如果依据本文提出的改进模型，即使剔除前瞻性偏差后也可以改进 ANN 预测模型的表现性能。

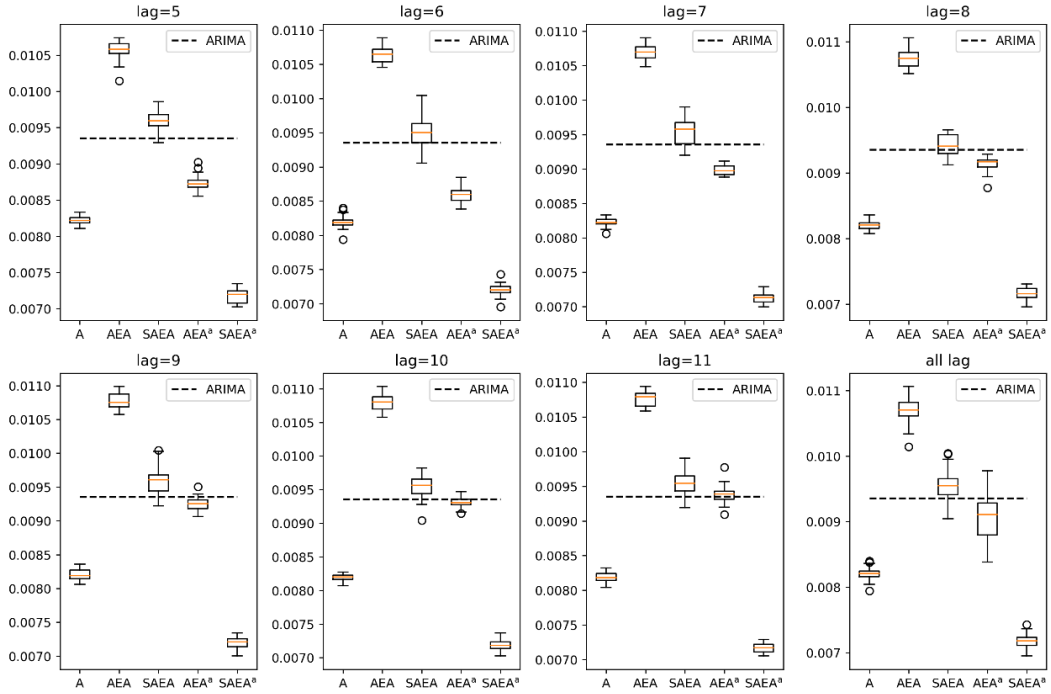


图 4-7 标准普尔 500 指数数据中剔除前瞻性偏差后模型的 MAPE 指标值

4.4 本章小结

本章在 4.1 小节中分析前视性误差的来源及成因，并且说明了经验模态分解算法中存在较大的前瞻性偏差。在 4.2 小结中，本文结合相关文献提出了一种改进的自适应预测模型。在 4.2 小节的介绍中可以知道自适应模型可以剔除前瞻性偏差，但有研究表明[70]剔除前瞻性偏差后，基于 EMD 的混合预测模型表现似乎不一定会优于单独的预测模型，4.3 小节的结论 3 我们也证实了这一点。基于此本文提出了一种改进的混合模型 S-AEMD-ANN^a。结合实证研究分析可知，本文提出的混合模型不论在效率上还是在预测精度上都优于文献中所提出的模型。而

且，最重要的是本文所提出的混合预测模型 S-AEMD-ANN[a] 优于单独的 ANN 预测模型，这说明如果使用本文改进的模型，即使剔除前瞻性偏差检验模态分解算法也是可以对预测模型起到改善作用。另外，本章的研究内容也再次证明了第三章本文所提出的基于 EMD 分解算法的单一混合预测模型的优越性，基于 EMD 的单一混合模型（图 3-4）比文献中的组合模型（图 3-2）在效率上有很大的提高，模型也比较简单，并且预测精度也有一定程度的改善。

5 结论与展望

5.1 本文结论

本文主要的研究内容是基于经验模态分解的金融时间序列预测方法的优化与改进。首先，针对目前基于检验模态分解算法的组合预测模型存在的问题，如预测模型复杂、组合模型规模过大、计算量大、误差累积等问题，文中第三章提出了一种基于预测结构的优化方式。其次，针对现有的基于经验模态分解的混合模型存在前瞻性偏差的问题，文中第四章提出了一种改进的自适应预测模型。并且，通过一些真实的金融时间序列进行实证研究，证明了模型改进的有效性。本文主要内容可以分为以下两点。

首先，针对基于 EMD 分解的组合预测模型中模型数量较多、规模复杂、误差累积和计算量大等问题，本文提出了一种基于 EMD 算法的单一模型预测结构。本文提出的预测模型不同于传统的预测模型。传统的预测模型是对 EMD 算法分解后的每一个子序列都进行建模预测，然后把预测结果相加得到最终预测结果。本文的预测模型是对 EMD 算法分解后的序列建立一个预测模型，避免叠加模型过多而使得预测结构复杂、计算量大、参数过多等问题，同时模型数量的减少也在一定程度避免了模型叠加时所造成的累积误差问题。本文第三章的“后视实验”研究以及第四章的“预测实验”研究都证明了本文提出的单一预测模型 S-EMD-ANN、S-AEMD-ANN 和 S-AEMD-ANN[a] 优于文献中传统意义上的组合模型 EMD-ANN[66,68,71,72]，AEMD-ANN[65、66]和 AEMD-ANN[a]。基于 EMD 的单一预测模型有效缩减了网络结构，从而有效减少了模型个数和简化了参数调节，使得模型的训练时间得到大幅度改善。并且从预测结果的各个评价指标来看，单一预测模型比传统的组合预测模型在预测精度上也有一定的提升。

其次，针对基于 EMD 分解过程中存在的前瞻性差问题，本文结合文献 Q. Tan(2108)[66]、X. Zhang(2015)[69]、Dennis(2017)[70]中的自适应混合模型的概念提出了一样改进的自适应预测模型。本文提出的改进模型主要从两个方面对文献中的自适应混合预测模型进行改善，第一个方面，主要基于第一部分的研究从模型结构上进行改善，将文献中组合自适应预测模型改为单一自适应预测模型，从 4.3 小结的实证实验研究中我们知道此改进对模型的运算效率有很大的提升，极大缩减了模型结构和训练时间同时也提升了模型的预测精度。第二方面，主要基于经验模态分解数据中端点失真问题的改进，由于自适应预测模型会使得端点效应在测试数据中极大的暴露出来所以靠近端点的数据分解会严重失真，而在训练数据

中却不会存在这样的问题，这样就会使得使用训练数据训练出来的模型可能在测试数据上表现很差，甚至并不一定优于单独的预测模型，X. Zhang(2015)[69]、Dennis(2017)[70]和本文 4.3 小结的结论 3 说明，如果剔除前视性偏差基于 EMD 分解的 SVM、RF、ANN 等一些混合预测模型都不一定优于单独的 SVM，RF 和 ANN 等预测模型。经过 4.2.2 小节的分析，本文提出了一种改进的自适应预测模型，在模型训练和测试时删除一些可能存在严重失真的变量，尽量用真实的数据去建模预测。4.3 节大量实证实验结果表明，经过变量处理的 AEMD-ANN^a 预测模型优于不经过变量处理的 AEMD-ANN 预测模型，经过变量处理的 S-AEMD-ANN^a 预测模型优于不经过变量处理的 S-AEMD-ANN 预测模型。并且，与文献中结论不同的是，本文结论认为即使剔除前瞻性偏差，如果依据本文的改进模型能对单独的预测模型起到改善作用。本文提出的改进模型 S-AEMD-ANN^a 在文中的各种实证结果上都要优于单独的 ANN 模型，这也说明了本文提出改进方案的有效性。

5.2 研究展望

本文主要研究内容有以下两个方面，第一个方面是基于经验模态分解的混合预测模型结构的优化研究；第二个方面是基于经验模态分解的混合模型中存在的前瞻性偏差问题的研究。文中并没有特别注重神经网络模型参数的调优，后续如果进行参数调优的话可能有更好的预测结果。而且，在文章的撰写和文献的阅读过程中发现文献未来可能有以下几个研究方向。

针对检验模态分解中高频子序列分解不稳定且无序的问题，有学者提出了二次分解技术，也就是高频无序的第一个 IMF 分量进行二次分解，再对二次分解出来的序列进行建模预测。如 Qu (2019)[68]在对风速预测中对无序的第一个本征模函数（IMF）进行二次经验小波分解（EWT empirical wavelet transform），然后对各分解的子序列进行预测。Sun (2018)在对短期风速预测时对无序的第一个本征模函数进行二次自适应变分模态分解（AVMD adaptive variational mode decomposition），然后对各子序列预测。在他们的研究结果中二次分解可以提高预测模型的精度。在未来的研究中，可以把二次分解技术运用到本文所提出的单一模型预测结构，预计也能提高预测的精度。

本文主要以最常用的 ANN 为基础预测模型结合 EMD 分解构建单一预测模型。因为文章中有大量的重复试验，ANN 在各种神经网络中计算量相对较小适合大量的重复试验。在未来可能会研究更加适合处理时间序列的循环神经网络（RNN）模型，如 LSTM 等模型结合本文的方法进行研究。在少量的实证研究表明，以 LSTM 为基础模型结合经验模态分解的单一自适应预测模型的预测结

果会更加精确，因为未进行大量的重复试验，该部分结果在附录 D 中给出。

第三章中的对比结果，发现本文所进行的实验结果 S-EMD-ANN 预测模型比 EMD-ANN 预测结果只改进了 1.5%左右，这是符合预期的。但在与同类研究对比时，发现结果改进在 20%-50%之间。对比研究发现，可能是在本文数据处理时进行了差分，造成了差异。可能对于低频时间序列差分后的数据在进行经验模态分解时能更好的提取时间序列的特征，这是一个猜想只在 3.2.2 小节进行了初步的研究，将来可能是研究方向之一。