A Bias-Variance-Complexity Trade-Off Framework for Complex System Modeling

Lean Yu[1,2], Kin Keung Lai[2,3], Shouyang Wang[1,3], and Wei Huang[4]

[1] Institute of Systems Science, Academy of Mathematics and Systems Science,
Chinese Academy of Sciences, Beijing 100080, China
{yulean, sywang}@amss.ac.cn
[2] Department of Management Sciences, City University of Hong Kong,
Tat Chee Avenue, Kowloon, Hong Kong
{msyulean, mskklai}@cityu.edu.hk
[3] College of Business Administration, Hunan University, Changsha 410082, China
[4] School of Management, Huazhong University of Science and Technology,
1037 Luoyu Road, Wuhan 430074, China

Abstract. This study proposes a new complex system modeling approach by extending a bias-variance trade-off into a bias-variance-complexity trade-off framework. In the framework, the computational complexity is introduced for system modeling. For testing purposes, complex financial system data are used for modeling. Empirical results obtained reveal that this novel approach performs well in complex system modeling and can improve the performance of complex systems by way of model ensemble within the framework.

1 Introduction

In the last few decades, system modeling and optimization – an important aspect of complex systems – has proved to be one of the hardest tasks in studying complex systems. The topic has, as a result, received increased attention, especially due to its difficulties and wide applications. Two key problems are (i) how to select an appropriate model class from various model classes (i.e., modeling) and (ii) how to make the final model closer to specific complex systems; in other words how to improve final model performance based on the given data (i.e., optimization or improvement).

In order to solve these problems, a bias-variance-complexity trade-off framework is proposed. The theoretical background of our framework is provided by the bias-variance-noise decomposition of the generalization error and introduction of complexity (see below). We argue that the introduction of complexity into the framework can lead to an appropriate model class selection, and an ensemble of the selected model class can lead to performance improvement of the final complex systems model under the proposed framework. Our procedures and methods are described in Section 3. In Section 4 an example from the financial complex system domain is presented for further explanation. Some concluding remarks are drawn in Section 5.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3980, pp. 518 – 527, 2006.

2 The Bias-Variance-Complexity Trade-Off Framework

This section mainly describes the theoretical background of the proposed framework. It has two parts: bias-variance-noise decomposition and bias-variance-complexity trade-off.

2.1 Bias-Variance-Noise Decomposition

Assume that there is a true function $y=f(x)+\varepsilon$, where ε is normally distributed with zero mean and standard deviation σ. Given a set of training sets D: {(xi, yi)}, we fit the unknown function h(x) = w·x + ξ to the data by minimizing the squared error $\sum_i [y_i - h(x_i)]^2$. Now, given a new data point x* with the observed value $y^* = f(x^*) + \varepsilon$, we would like to understand the expected error $E[(y^* - h(x^*))^2]$. We then decompose this formula into "bias", "variance" and "noise" in the following:

$$\begin{aligned}
&E[(h(x^*) - y^*)^2] = E[(h(x^*))^2 - 2h(x^*)y^* + (y^*)^2] \\
&= E[(h(x^*))^2] - 2E[h(x^*)]E(y^*) + E[(y^*)^2] \qquad (\because E(Z - \overline{Z})^2 = E(Z^2) - \overline{Z}^2) \\
&= E[(h(x^*) - \overline{h}(x^*))^2] + (\overline{h}(x^*))^2 - 2\overline{h}(x^*) f(x^*) + E[(y^* - f(x^*))^2] + (f(x^*))^2 \\
&= E[(h(x^*) - \overline{h}(x^*))^2] + E[(y^* - f(x^*))^2] + (\overline{h}(x^*) - f(x^*))^2 \\
&= Var(h(x^*)) + E(\varepsilon^2) + Bias^2(h(x^*)) \\
&= Bias^2(h(x^*)) + Var(h(x^*)) + \sigma^2
\end{aligned} \tag{1}$$

From Equation (1), the expected error consists of three components: bias, variance and noise. The loss of the bias is from the difference between average prediction and optimal prediction, and is mainly caused by the learning algorithm. The variance originates from the difference between any prediction and the average prediction, and is often caused by using different training sets. The noise is very small and comes from the difference between optimal prediction and true function. Usually, the noise is hard to reduce, as in practice the inherent noise is often unknown. Thus the expected error is roughly equal to the sum of the squared bias and variance, as seen in Equation (2) below.

$$\text{Expected error } E[(h(x^*) - y^*)^2] = \text{bias}^2 (h(x^*)) + \text{variance } (h(x^*)). \tag{2}$$

2.2 The Bias-Variance-Complexity Trade-Off Framework

According to [1], bias decreases as model complexity (i.e., the number of parameters) increases, whereas variance increases with model complexity, i.e., the more complex the model, the higher the variance. On the other hand, if the model is too simple, the bias will increase. There is a close relation among bias, variance and complexity, as is illustrated in Fig 1.

From Fig. 1, we can see that there is a trade-off relationship among bias, variance and complexity. Through this trade-off, the optimal complexity can be found. Also, bias and variance are optimal because the sum of two parts can attain the minimum in the total error curve.

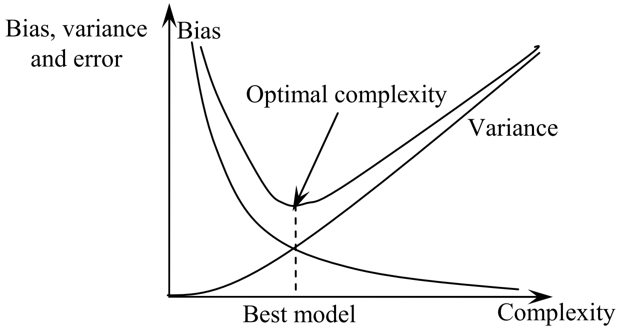

Fig. 1. The relationship among bias, variance and complexity

The bias-variance-complexity trade-off provides a conceptual framework for modeling complex systems, i.e., for determining a good model for complex systems. The detailed discussions are presented in the next section.

3 Complex System Modeling Under the Proposed Framework

In this section, we mainly discusses the two important modeling problems for complex systems – model selection and model improvement and optimization – under the proposed framework.

3.1 The Model Selection Under the Proposed Framework

As previously stated, the bias-variance-complexity trade-off provides a conceptual framework for determining a good model for complex systems. However, we need to obtain a practical criterion for determining a good model by optimizing the three components under the proposed framework. It should be noted that the ultimate goal of model selection under the proposed framework is to choose a good model which will perform the best on future testing data, i.e., a good generalization.

As earlier revealed, if the model is too complex for the amount of training data, it learns (or memorizes) parts of noise as well as problems in the underlying structure, resulting in “overfitting” or high variance as well as low bias. The selected model will perform badly in the testing data (or have a weak generalization). Inversely, if the model is not complex enough, it cannot capture the underlying structure in the data, no matter how much data it is given; this leads to “underfitting” or high bias. The selected model will also perform very badly (or have a poor generalization). In addition, the parsimony principle or Okham’s razor [2] shows that “from all models describing the process with the same level of accuracy, the simplest is the best”. This implies that we may find a good model selection criterion through a rational trade-off among bias, variance and complexity.

Based on the above descriptions and the proposed framework, it is possible to find a good model by minimizing the following selection criterion:

$$\text{Model selection criterion} = f(\text{bias, variance, complexity}). \tag{3}$$

From Equation (3), we see that the model selection criterion is actually a multi-objective optimization problem, i.e., minimizing the bias and variance for a given or appropriate complexity.

The aim of selection is for the model to perform good generalization of new observations or unknown data. However, as [3] pointed out, model generalization is often defined as the prediction risk. With the help of this concept and Equation (3), we suppose that $x \in \boldsymbol{R}^m$ is random sampled according to a distribution $p(x)$, the expected output for x is $y = f(x) + \varepsilon$ and the prediction output is $h(x)$. We can then formulate a concrete model selection criterion in the following:

$$\text{Model selection criterion} = \varphi(d/n)\int p(x)[f(x) - h(x)]^2 dx \tag{4}$$

where φ is a monotonically increasing function of the ratio of model complexity (i.e., the number of parameters or degrees of freedom) d and the training sample size n [4]. The function φ is often called the penalization factor because it inflates the average residual sum of squares for increasingly complex models. Several forms of φ have been proposed in the statistical literature, such as final prediction error (FPE) [5] and Schwartz' criterion (SC) [6]:

$$\text{FPE: } \varphi(q) = (1+q)(1-q)^{-1} \tag{5}$$

$$\text{SC: } \varphi(q,n) = 1 + 0.5 \cdot (\log n) \cdot q \cdot (1-q)^{-1} \tag{6}$$

where q denotes the ratio of model complexity and training sample size. In this study we used FPE as the penalization factor of complexity.

In addition, in view of the results of [3], the model selection criterion can be approximated by the expected performance on a finite test set.

$$\text{Model selection criterion} = \varphi(d/n)E[(f(x) - h(x))^2]\,. \tag{7}$$

With Equations (1), (2) and (5), the final model selection criterion can be written as

$$\text{Selection criterion} = \left[(n+d)/(n-d)\right] \cdot \left[(\bar{h}(x) - f(x))^2 + E[(h(x) - \bar{h}(x))^2]\right]. \tag{8}$$

As can be seen from Equation (8), the bias, variance and complexity are all taken into account in the model selection process. Through the trade-off of bias, variance and complexity, as shown in Fig. 1, we can find an appropriate model class from various model classes for specific complex systems. In practice, the generic judgment rule is known as "the smaller the selection criterion value the better the model".

3.2 The Model Improvement Under the Proposed Framework

Although a good model can be selected using the model selection criterion described above, the model does not necessarily give a good generalization because of the difficulties of complex system modeling. In order for a complex system to perform well, it is necessary to improve and optimize the selected model from the previous phase. In this study, model ensemble is used.

Model ensemble is a subject of active research. It makes possible an increase in generalization performance by combining several individual models trained on the same tasks. The ensemble approach has been justified both theoretically [7] and empirically [8]. Generally, the creation of an ensemble is divided into two steps, the first being the judicious creation of the individual ensemble members and the second their appropriate combination to produce the ensemble output. The widely-used ensemble model includes bagging [9] and boosting [10]. In this study, we propose a new approach to building an ensemble model for complex systems based on the proposed bias-variance-complex trade-off framework.

Our proposed approach is based on the observation that the generalization error of an ensemble model can be improved if the predictors on which averaging is done disagree and if their fluctuations are uncorrelated [11]. We now consider the case of an ensemble model $\hat{h}(\mathbf{x})$ consisting of M individual models, $\hat{h}_1(\mathbf{x})$,..., $\hat{h}_M(\mathbf{x})$; the ensemble model is represented as

$$\hat{h}(\mathbf{x}) = \sum_{i=1}^{M} w_i \hat{h}_i(\mathbf{x}) \tag{9}$$

where the weights may sum to one, i.e., $\sum_{i=1}^{M} w_i = 1$. Given the testing data D_{test} = $\{(x_1,y_1),\ldots,(x_N,y_N)\}$, then and the ensemble mean squared error is defined as:

$$\text{Ensemble mean squared error} = (1/NM)\sum_{i=1}^{N}\sum_{j=1}^{M}(y_i - \hat{h}_j(\mathbf{x}_i))^2 \tag{10}$$

By introducing the average model $\bar{h}(\mathbf{x}_i) = (1/M)\sum_{j=1}^{M} h_j(\mathbf{x}_i)$ the mean squared error can be decomposed into bias and variance in terms of Equations (1) and (2):

$$\text{Bias}^2 = (1/N)\sum_{i=1}^{N}(y_i - \bar{h}(\mathbf{x}_i))^2 \tag{11}$$

$$\text{Variance} = (1/NM)\sum_{i=1}^{N}\sum_{j=1}^{M}(\bar{h}(\mathbf{x}_i) - \hat{h}_j(\mathbf{x}_i))^2 . \tag{12}$$

We can examine the effects of bias and variance from Equations (11) and (12). The bias terms depends on the target distribution (y), while the variance term does not. A more elaborate formulation would further decompose the bias term into true bias and noise (see Equation (1)), but as in practice the inherent noise is often unknown, the current definition is used here. The variance terms of the ensemble could be decomposed in the following way:

$$\begin{aligned} Var(\hat{h}(\mathbf{x})) &= E[(\hat{h}(\mathbf{x}) - \bar{h}(\mathbf{x}))^2] = E[(\hat{h}(\mathbf{x}) - E(\hat{h}(\mathbf{x})))^2] \\ &= E[(\sum_{i=1}^{M} w_i h_i(\mathbf{x}))^2] - (E[\sum_{i=1}^{M} w_i h_i(x)])^2 \\ &= \sum_{i=1}^{M} w_i^2 (E[h_i^2(\mathbf{x})] - E^2[h_i(\mathbf{x})]) + 2\sum_{(i<j)} w_i w_j (E[h_i(x)\cdot h_j(x)] - E[h_i(x)]E[h_j(x)]) \end{aligned} \tag{13}$$

where the expectation is taken with respect to D. The first sum in Equation (13) marks the lower limit of the ensemble variance and is the weighted mean of the variance of ensemble members. The second sum contains the cross terms of the ensemble members and disappears if the models are completely uncorrelated [11]. Thus, we focus on the second part so as to lower the variance.

Through observing Equations (11) and (13), we can find several ways to reduce the expected error by ensemble under the bias-variance-complexity trade-off framework: (i) increase the number of individual models with the given data as much as possible to lower bias; (ii) build some independent models for lowering variance; and (iii) keep an appropriate computational complexity for ensemble. The final ensemble model can improve the performance of complex systems by rational trade-off processing based on the proposed framework.

3.3 The New Modeling Approach for Complex Systems Under This Framework

Based on the previous two subsections, a novel modeling approach for complex systems is proposed. This approach consists of two phases, model selection and model improvement, or five steps. The procedure is as follows.

A. Model selection phase

1) Given a data set of complex system D, a disjoint training set D_{train} and D_{test} is first created. The former is used to build a model and the latter to test the model.

2) Because we do not know the patterns of complex systems due to their difficulties, different types of models (or model classes) are used to try to capture the characteristics of complex systems.

3) With the D_{train}, different model classes can be built. In terms of the D_{test} and corresponding model selection criterion mentioned in the Section 3.1, an appropriate model class can be selected. Generally, the selected model class can capture more useful patterns than can other candidate model classes, and so will have low bias in a sense.

B. Model improvement phase

4) As Section 3.2 showed, model ensemble can improve the model performance of a complex system. From Equation (11), we know that the improvement in the bias is very limited. Inversely, the improvement space in the variance is large (Equations (12) and (13)) only if the individual models are independent or completely uncorrelated [11]. The simplest method for creating diverse ensemble members is to train each model using randomly initialized conditions, such as different model architectures and

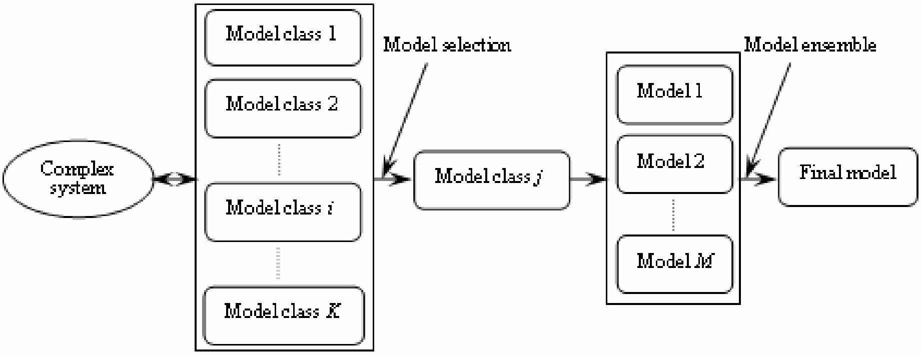

Fig. 2. The basic process of the proposed approach

different training subsets [12]. So in this step, we create a large number of ensemble members with different initialized conditions.

5) The different ensemble members obtained from the previous step are synthesized into the final model for a complex system.

The basic process is illustrated in Figure 2. For further interpretation, an example from the financial domain is presented in the following section.

4 The Empirical Study

In this section, foreign exchange rate modeling and forecasting is used as an illustrative example of our proposed approach. Two widely traded exchange rates the US dollar/euro (USD/EUR) and the US dollar/Japanese yen (USD/JPY) are chosen. We take daily data (source: DataStream) from 1 January 2000 until 31 October 2004 as entire data sets (partial data sets excluding holidays). For space reasons, the original data are not listed in this study but can be viewed on the website. For convenience, we take daily data from 1 January 2000 to 31 October 2003 as the training data set (D_{train}), and daily data from 1 November 2003 to 31 October 2004 as the testing set (D_{test}); these are used to evaluate the model performance. In this study, four model classes, linear polynomial model (LPM), K-Nearest-Neighbor (KNN) model, logit regression model (LRM), and feed-forward neural network (FNN) model, are selected as candidate model classes. We generated 100 training sets $\{D_i \ (i = 1,2,\ldots,100)\}$ of fixed size N with the use of D_{train} to train different model classes. We let $\bar{h}(x) = (1/100)\sum_{i=1}^{100} h(\mathbf{x}, D_i)$ denote the average of these model classes. We then verify our approach with the following procedures.

A. Model class selection phase

Based on the above descriptions and model selection criterion, the results of four model classes are reported below.

As can be seen from Table 1, we know that the best model class is the feed-forward neural network model in terms of the proposed model selection criterion for two exchange rates. With regard to complexity (here referring to model parameters), the effect of penalty has been taken into consideration by the selection criterion value.

Table 1. Simulation results of model selection for two exchange rates *

Exchange	Classes	Bias^2	Variance	Complexity	Criterion value
USD/EUR	LPM	0.065413	0.053815	10	0.129164
	KNN	0.058445	0.081252	14	0.156271
	LRM	0.075854	0.078639	12	0.171241
	FNN	0.032587	0.054014	30	**0.110219**
USD/JPY	LPM	0.184577	0.084758	14	0.301290
	KNN	0.223143	0.105472	13	0.364666
	LRM	0.254876	0.085347	8	0.351490
	FNN	0.128114	0.094756	37	**0.300299**

* Criterion value = (Bias^2 + Variance) × penalization factor of complexity.

In addition, we also see that the bias of the FNN model is the lowest of the four model classes, as earlier revealed. According to the proposed model selection criterion, we select the FNN model as an agent for exchange-rate modeling. However, we also note that the variance of the FNN model is relatively large in the four model classes, implying that there is room for improvement in the FNN model. In the sequel, model improvement is performed.

B. Model improvement phase

In this phase, we use model ensemble technique to improve the performance of the model selected in the previous phase. As Equation (13) shows, the variance will be reduced if the models are uncorrelated. That is, model diversity can reduce model error variance. In the case of neural network ensembles, the networks can have different architecture, different training algorithms or different training subsets, and different initialized weights or random weights [12–13]. In our study, we use these diverse methods to create different ensemble members. In order to have fair competition, the estimation of bias and variance is calculated for every ensemble with different complexity (here referring to the number of ensemble members). Simulation results are presented in the Table 2.

Table 2. Simulation results of different ensemble models for two exchange rates *

Exchange	Type	Complexity	Bias^2	Variance	Expected error
USD/EUR	Benchmark	1	0.032587	0.054014	0.086601
	Ensemble1	50	0.032225	0.050257	0.082482
	Ensemble2	100	0.032126	0.048253	**0.080379**
	Ensemble3	150	0.032158	0.048878	0.081036
	Ensemble4	200	0.032254	0.048854	0.081108
	Ensemble5	250	0.032545	0.048832	0.081377
USD/JPY	Benchmark	1	0.128114	0.094756	0.222870
	Ensemble1	50	0.127453	0.090015	0.217468
	Ensemble2	100	0.126587	0.089547	0.216134
	Ensemble3	150	0.127098	0.084854	**0.211952**
	Ensemble4	200	0.127805	0.087055	0.214860
	Ensemble5	250	0.127987	0.089811	0.217798

* Expected error = (Bias^2 + Variance).

From Table 2, we see that (a) the ensemble model with 100 members performs the best for USD/EUR, while for USD/JPY, the ensemble model with 150 members performs the best. The main reason is that the fluctuation of the Japanese yen is more complex than that of the euro; (b) compared to the benchmark model (i.e., single model), all the ensemble models lower the bias and variance, but bias reduction is less than variance reduction, implying that the ensemble can effectively reduce the variance; (c) of all the ensemble models, the most complex one does not necessarily give the best performance, as revealed by experiments, implying that an ensemble model should have an appropriate complexity (be neither too complex nor too simple); (d) all the ensemble models perform better by observing the expected error, implying that the ensemble technique is an effective complex system modeling technique for improving modeling performance.

5 Conclusions

In this study we propose a novel complex system modeling approach based on the bias-variance-complexity trade-off framework. This approach consists of two phases: model selection and model improvement. In the first phase, we select an appropriate model class as modeling agent in terms of bias-variance-complexity trade-off. In the second phase, we improve complex system model performance by ensemble, based on the framework. Experimental results demonstrate that the proposed approach is effective.

Acknowledgements

This work is partially supported by National Natural Science Foundation of China (NSFC No. 70221001); Chinese Academy of Sciences; Key Research Institute of Humanities and Social Sciences in Hubei Province-Research Center of Modern Information Management and Strategic Research Grant of City University of Hong Kong (SRG No. 7001677).

References

1. Geman, S., Bienenstock, E., Doursat, R.: Neural networks and the bias/variance dilemma. Neural Computation 4 (1992) 1–58
2. Myung, I. J., Pitt, M. A.: Applying Okham's razor in modeling cognition: A Bayesian approach. Psychonomic Bulletin & Review 4 (1996) 79–95
3. Moody, J.: Prediction risk and architecture selection for neural networks. In: Cherkassky, V., Friedman, J. H., Wechsler, H. (eds.): From Statistics to Neural Networks: Theory and Pattern Recognition Applications. NATO ASI Series F, Springer-Verlag, 1994
4. Hardle, W., Hall, P., Marron, J. S.: How far are automatically chosen regression smoothing parameters from their optimum? Journal of the American Statistical Association 83 (1988) 86–95
5. Akaike, H.: Statistical predictor information. Annals of the Institute of Statistical Mathematics 22 (1970) 203–217
6. Shwartz, G.: Estimating the dimension of a model. Annals of Statistics 6 (1978) 461–464
7. Hansen, L. K., Salamon, P.: Neural network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (1990) 993–1001
8. Opitz, D, Maclin, R.: Popular ensemble methods: An empirical study. Journal of Artificial Intelligence Research 11 (1999) 169–198
9. Breiman, L.: Bagging predictors. Machine Learning 24 (1996) 123–140
10. Freund, Y., Schapire, R. E.: Experiments with a new boosting algorithm. Machine Learning: Proceedings of the 13th International Conference (1996) 148–156
11. Krogh, A., Sollich, P.: Statistical mechanics of ensemble learning. Physical Review E 55 (1997) 811–825

12. Perrone, M. P., Cooper, L. N.: When neural networks disagree: Ensemble methods for hybrid neural networks. In: Mammone, R. J. (ed.): Neural Networks for Speech and Image Processing. Chapman-Hall (1993) 126–142
13. Krogh, A., Vedelsby, J.: Neural network ensembles, cross validation, and active learning. In: Tesauro, G., Touretzky, D., Leen, D. (eds.): Advances in Neural Information Processing Systems. MIT Press (1995) 231–238