# 7

# *Modeling Uncertainty: From Point Estimates to Prediction Intervals*

In earlier chapters a framework was presented where a wavelet network can efficiently be constructed, initialized, and trained. In this chapter we discuss the reliability of estimates of wavelet networks, since forecasts are characterized by uncertainty due to (1) inaccuracy in the measurements of the training data, and (2) limitations of the model. More precisely, in this chapter the framework proposed is expanded by presenting two methods for estimating confidence and prediction intervals.

The output of the wavelet network is the approximation of the underlying function $f(\mathbf{x})$ obtained from the noisy data. In many applications, and especially in finance, risk managers may be more interested in predicting intervals for future movements of the underlying function $f(\mathbf{x})$ than simply point estimates. For example, financial analysts who want to forecast the future movements of a stock are interested not only in the prices predicted but also in the confidence and prediction intervals. For example, if the price of a stock moves outside the prediction intervals, a financial analyst will take a position in the stock. If the price of the stock is below the lower bound, the stock is traded lower that it should be and a long position must be taken. On the other hand, if the price stock is above the upper bound of the prediction interval, the stock is too expensive and a short position should be taken.

In real data sets the training patterns are usually inaccurate, since they contain noise or they are incomplete due to missing observations. Financial time series especially are dominated by these characteristics. As a result, the validity of the predictions of our model (as well as of any other model) is questioned. The uncertainty that results

*Wavelet Neural Networks: With Applications in Financial Engineering, Chaos, and Classification*, First Edition. Antonios K. Alexandridis and Achilleas D. Zapranis.

from the data, called the *data noise variance* $\sigma_\varepsilon^2$, contributes to the total variance of the prediction (Breiman, 1996; Carney et al., 1999; Heskes, 1997; Papadopoulos et al., 2000).

On the other hand, presenting to a trained network new data that were not introduced to the wavelet networks during the training phase, additional uncertainty is introduced to the predictions. Since the training set consists of a finite number of training pairs, the solution $\hat{\mathbf{w}}_n$ is likely not to be valid in regions not represented in the training sample (Papadopoulos et al., 2000). In addition, the iterative algorithm that is applied to train a wavelet network may converge to a local minimum of the loss function. This source of uncertainty, which arises from misspecifications in the model or in the parameter selection as well as from limitations of the training algorithm also contributes to the total variance of the prediction, is called the *model variance* $\sigma_m^2$ (Papadopoulos et al., 2000).

The model variance and the data noise variance are assumed to be independent. The total variance of the prediction is given by the sum of two variances:

$$\sigma_p^2 = \sigma_m^2 + \sigma_\varepsilon^2 \tag{7.1}$$

To apply wavelet networks in financial applications, a statistical measure of the confidence of the predictions must be derived.

If the total variance of a prediction can be estimated, it is possible to construct confidence and prediction intervals. In the first case we are interested in the difference between the network's output and the true underlying generating process; in the second case we are interested in the difference between the network's output and the value observed. We explore this in the remainder of the chapter.

## THE USUAL PRACTICE

In the framework of classical sigmoid neural networks, the methods proposed for constructing confidence and prediction intervals fall into three major categories: the analytical, the Bayesian, and the ensemble network methods. Analytical methods provide good prediction intervals only if the training set is very large (De Veaux et al., 1998). They are based on the assumptions that the noise in the data is independent and identically distributed with mean zero and constant standard deviation. In real problems that hypothesis usually does not hold. As a result, there will be intervals where the analytical method either over- or underestimates the total variance. Finally, with analytical methods the effective number of parameters must be identified, although pruning schemes such as the Irrelevant Connection Elimination scheme can be used to solve this problem. On the other hand, Bayesian methods are computationally expensive techniques that need to be tested further (Zapranis and Refenes, 1999; Ζαπράνης, 1999). Results from Papadopoulos et al. (2000) indicate that the use of Bayesian methods and the increase in the computational burden are not justified by their performance. Finally, analytical and Bayesian methods are computationally complex since the inverse of the Hessian matrix must be estimated, which under certain circumstances can be very unstable.

Finally, ensemble network methods create different versions of the initial network and then combine the outputs to provide constancy to the predictor by stabilizing the high variance of a wavelet network. In ensemble network methods the new versions of the network are generally created using bootstrapping. The only assumption needed is that the wavelet network provides an unbiased estimation of the true regression. Moreover, ensemble networks can handle nonconstant variance. Assuming a constant variance is a simplification of reality. In real data sets the variance changes over time as new data arrive. Similarly, in finance, the variance of daily data changes as new information arrives at the traders. Hence, we suppose that the total variance of the prediction is not constant and is given by

$$\sigma_p^2(\mathbf{x}) = \sigma_m^2(\mathbf{x}) + \sigma_\varepsilon^2(\mathbf{x}) \tag{7.2}$$

The methods most often cited are bagging (Breiman, 1996) and balancing (Carney et al., 1999; Heskes, 1997). In the following sections we adapt these two methods to construct confidence and prediction intervals under the framework of wavelet networks. A framework similar to the one presented by Carney et al. (1999) to estimate the total prediction variance $\sigma_p^2$ and to construct confidence and prediction intervals is adapted.

## CONFIDENCE AND PREDICTION INTERVALS

Suppose that our set of observations is given by $D_n = (\mathbf{x}_i, y_i)$, $i = 1,\ldots,n$, which verifies the following nonlinear nonparametric wavelet network:

$$y_i = g_\lambda(\mathbf{x}_i; \mathbf{w}_0) + \varepsilon_i. \tag{7.3}$$

where $y_i$ is the output of the wavelet network $g_\lambda(\mathbf{x}_i; \mathbf{w}_0)$ and $\mathbf{w}_0$ represents the true vector of parameters for the specific unknown function $\varphi(\mathbf{x}_i)$, which is estimated by the network. This means that

$$g_\lambda(\mathbf{x}_i; \mathbf{w}_0) \approx \varphi(\mathbf{x}_i) \equiv \mathrm{E}[y_i \mid \mathbf{x}_i] \tag{7.4}$$

Initially, we assume that the error $\varepsilon_i$ is distributed independently and identically with zero mean and variance $\sigma_\varepsilon^2$.

The estimation of the vector $\mathbf{w}_0$ using least squares is given by the vector $\hat{\mathbf{w}}_n$. The vector $\hat{\mathbf{w}}_n$ is estimated by minimizing the sum of squares of the error:

$$\mathrm{SSE} = \sum_{i=1}^{n} [y_i - g_\lambda(\mathbf{x}_i; \mathbf{w})]^2 \tag{7.5}$$

For the input vector $\mathbf{x}_i$ and the weight vector of the network $\hat{\mathbf{w}}_n$, the output of the network is

$$\hat{y}_i = g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n) \tag{7.6}$$

Within this context, the concept of "confidence" has a dual meaning. In the first case we are interested in the accuracy of the estimation of the actual but unknown function $\varphi(\mathbf{x}_i)$. Namely, the distribution of the quantity

$$\varphi(\mathbf{x}_i) - g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n) \equiv \varphi(\mathbf{x}_i) - \hat{y}_i \tag{7.7}$$

referred to as the *confidence interval.* In the second case we are interested in the accuracy of the estimation regarding the network output observed. That relates to the distribution of the quantity

$$y_i - g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n) \equiv y_i - \hat{y}_i \tag{7.8}$$

and is referred to as the *prediction interval*.

Figures 7.1 and 7.2 present the relationship between the differences (7.7) and (7.8). In Figure 7.1 the prediction obtained by the wavelet network has a greater value than the value observed, while in Figure 7.2 the opposite is true. We observe that

$$\begin{aligned} & y_i - g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n) = \{\varphi(\mathbf{x}_i) - g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n)\} + \{y_i - \varphi(\mathbf{x}_i)\} \\ & \Leftrightarrow y_i - g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n) = \{\varphi(\mathbf{x}_i) - g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n)\} + \varepsilon_i \end{aligned} \tag{7.9}$$

or equivalently from relations (7.7) and (7.8),

$$y_i - \hat{y}_i = (\varphi(\mathbf{x}_i) - \hat{y}_i) + \varepsilon_i \tag{7.10}$$

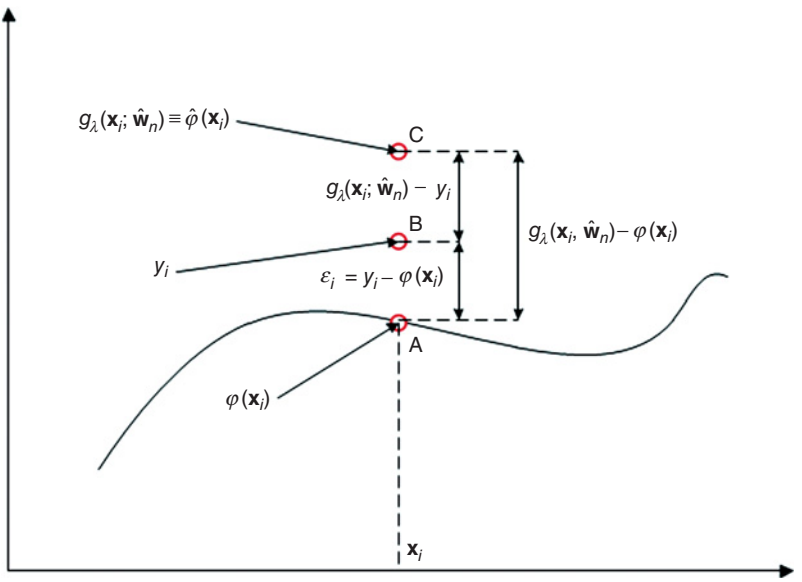


***Figure 7.1*** *Relationship between the wavelet network output $\hat{y}_i$, the observation $y_i$, and the underlying function $\varphi(\mathbf{x}_i)$ that created the observation by adding the stochastic term $\varepsilon_i$ in the case where the predicted value is greater than the observation ($\hat{y}_i > y_i$).*

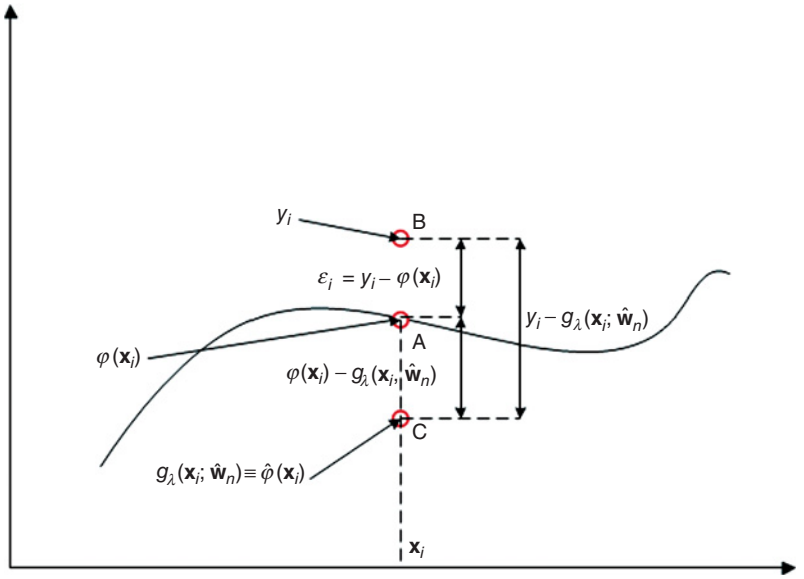


***Figure 7.2*** *Relationship between the wavelet network output $\hat{y}_i$, the observation $y_i$, and the underlying function $\varphi(\mathbf{x}_i)$ that created the observation by adding the stochastic term $\varepsilon_i$ in the case where the predicted value is less than the observation ($\hat{y}_i > y_i$).*

From (7.10) we can conclude that the confidence interval is included in the prediction interval. Since the difference $(\varphi(\mathbf{x}_i) - \hat{y}_i)$ and the error term $\varepsilon_i$ in (7.10) are statistically independent, it holds that

$$\begin{aligned} &E\left[(y_i - \hat{y}_i)^2\right] = E\left[(\varphi(\mathbf{x}_i) - \hat{y}_i)^2\right] + E\left[\varepsilon_i^2\right] \\ &\Leftrightarrow \text{var}\left[y_i - \hat{y}_i\right] = E\left[\varphi(\mathbf{x}_i) - \hat{y}_i\right] + \text{var}\left[\varepsilon_i^2\right] \end{aligned} \tag{7.11}$$

which results in

$$\sigma_p^2(\mathbf{x}) = \sigma_m^2(\mathbf{x}) + \sigma_\varepsilon^2 \tag{7.12}$$

The assumption that the variance of the term $\sigma_\varepsilon^2$ is constant, has in practice proved simplistic, particularly in the case of financial problems. Hence, equation (7.3) can be rewritten using the following more realistic form:

$$y_i = g_\lambda(\mathbf{x}_i; \mathbf{w}_0) + \varepsilon_i(\mathbf{x}_i) \tag{7.13}$$

In this case we have that

$$\begin{aligned} &y_i - g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n) = \{\varphi(\mathbf{x}_i) - g_\lambda(\mathbf{x}_i; \hat{\mathbf{w}}_n)\} + \varepsilon_i(\mathbf{x}_i) \\ &\Rightarrow \text{var}\left[y_i - \hat{y}_i\right] = E\left[\varphi(\mathbf{x}_i) - \hat{y}_i\right] + \text{var}\left[\varepsilon_i^2(\mathbf{x}_i)\right] \end{aligned} \tag{7.14}$$

which will result in

$$\sigma_p^2(\mathbf{x}) = \sigma_m^2(\mathbf{x}) + \sigma_\varepsilon^2(\mathbf{x}) \tag{7.15}$$

## CONSTRUCTING CONFIDENCE INTERVALS

To generate confidence intervals, distribution of the accuracy of the network prediction to the true underlying function is needed. In other words, the variance of the distribution of

$$f(\mathbf{x}) - \hat{y} \equiv f(\mathbf{x}) - g_\lambda(\mathbf{x}, \hat{\mathbf{w}}_n) \tag{7.16}$$

must be estimated. The model variance $\sigma_m^2$ will be estimated using two different bootstrapping methods: the bagging method, proposed by Breiman (1996), and the balancing method, proposed by Heskes (1997) and Carney et al. (1999). Both methods are variations of bootstrapping.

### The Bagging Method

As mentioned earlier, ensemble network methods create different versions of the initial network and then combine the outputs to provide constancy to the predictor by stabilizing the high variance of the wavelet network. To do so, bootstrapping is used to create a new training sample from the initial data set. The algorithm of the bagging methods is described below.

In the first step, $B = 200$ new random samples with replacement are created from the original training sample. A new wavelet network is trained in each of the bootstrapped samples, $g_\lambda(\mathbf{x}^{(*i)}; \hat{\mathbf{w}}^{(*i)})$, where $(*i)$ indicates the $i$th bootstrapped sample and $\hat{\mathbf{w}}^{(*i)}$ is the solution of the $i$th bootstrapped sample. Each network is trained using the same topology as in the case of the original network (i.e., the same number of hidden units). Then each new network is evaluated to the original training sample, $\mathbf{x}$. In other words, we measure the forecasting accuracy of each bootstrapped network to the original data set.

The next step is to estimate the average output of the $B$ networks using the original training sample $\mathbf{x}$:

$$g_{\lambda,\text{avg}}(\mathbf{x}) = \frac{1}{B}\sum_{i=1}^{B} g_\lambda(\mathbf{x}; \hat{\mathbf{w}}^{(*i)}) \tag{7.17}$$

It is assumed that the wavelet network produces an unbiased estimate of the underlying function $f(\mathbf{x})$. This means that the distribution of $P(f(\mathbf{x}) \mid g_{\lambda,\text{avg}}(\mathbf{x}))$ is centered on the estimate $g_{\lambda,\text{avg}}(\mathbf{x})$ (Carney et al., 1999; Heskes, 1997; Zapranis and Livanis, 2005). Since the wavelet network is not an unbiased estimator (as any other model), it is assumed that the bias component arising from the wavelet network is negligible

compared to the variance component (Carney et al., 1999; Zapranis and Livanis, 2005). Finally, if we assume that the distribution of $P(f(\mathbf{x}) \mid g_{\lambda,\text{avg}}(\mathbf{x}))$ is normal, the model variance can be estimated by

$$\hat{\sigma}_m^2(\mathbf{x}) = \frac{1}{B-1} \sum_{i=1}^{B} \left[ g_\lambda(\mathbf{x}; \hat{\mathbf{w}}^{(*i)}) - g_{\lambda,\text{avg}}(\mathbf{x}) \right]^2 \tag{7.18}$$

To construct confidence intervals, the distribution of $P(g_{\lambda,\text{avg}}(\mathbf{x}) \mid f(\mathbf{x}))$ is needed. Since the distribution of $P(f(\mathbf{x}) \mid g_{\lambda,\text{avg}}(\mathbf{x}))$ is assumed to be normal, the "inverse" distribution $P(g_{\lambda,\text{avg}}(\mathbf{x}) \mid f(\mathbf{x}))$ is also normal. However, this distribution is unknown. Alternatively, it is estimated empirically by the distribution of $P(g_\lambda(\mathbf{x}) \mid g_{\lambda,\text{avg}}(\mathbf{x}))$ (Carney et al., 1999; Zapranis and Livanis, 2005). Then the confidence intervals are given by

$$g_{\lambda,\text{avg}}(\mathbf{x}) - t_{\alpha/2}\hat{\sigma}_m(\mathbf{x}) \leq f(\mathbf{x}) \leq g_{\lambda,\text{avg}}(\mathbf{x}) + t_{\alpha/2}\hat{\sigma}_m(\mathbf{x}) \tag{7.19}$$

where $t_{\alpha/2}$ can be found in a Student's $t$ table and $1 - a$ is the confidence level desired.

### The Balancing Method

However, the estimator of the model variance, $\hat{\sigma}_m^2$, given by (7.18) is known to be biased (Carney et al., 1999); as a result, wider confidence intervals will be produced. Carney et al. (1999) proposed a balancing method to improve the model variance estimator. The algorithm for the balancing method is described below.

In the first strep, $B = 200$ new random samples with replacement are created from the original training sample. As in the bagging method, a new wavelet network is trained in each of the bootstrapped samples, $g_\lambda(\mathbf{x}^{(*i)}; \hat{\mathbf{w}}^{(*i)})$, where $(^*i)$ indicates the $i$th bootstrapped sample and $\hat{\mathbf{w}}^{(*i)}$ is the solution of the $i$th bootstrapped sample. Each network is trained using the same topology as in the case of the original network. Then each new network is evaluated to the original training sample $\mathbf{x}$. In other words, we measure the forecasting accuracy of each bootstrapped network for the original data set.

Then the $B$ bootstrapped samples are divided into $M$ groups. More precisely, in our case the 200 ensemble samples are divided into $M = 8$ groups of 25 samples each. Next, the average output of each group is estimated:

$$\zeta = \left\{ g_{\lambda,\text{avg}}^{(i)}(\mathbf{x}) \right\}_{i=1}^{M} \tag{7.20}$$

The model variance is not estimated just by the $M$ ensemble output since this estimation will be highly volatile (Carney et al., 1999). To overcome this, a set of $P = 1000$ bootstraps of the values of $\zeta$ is created:

$$Y = \left\{ \zeta_j^* \right\}_{j=1}^{P} \tag{7.21}$$

where

$$\zeta_j^* = \left\{ g_{\lambda,\text{avg}}^{(*j1)}(\mathbf{x}), g_{\lambda,\text{avg}}^{(*j2)}(\mathbf{x}),\ldots, g_{\lambda,\text{avg}}^{(*jM)}(\mathbf{x}) \right\} \tag{7.22}$$

is a bootstrapped sample of $\zeta$. Then the model variance is estimated for each one of these sets by

$$\hat{\sigma}_j^{2*}(\mathbf{x}) = \frac{1}{M} \sum_{k=1}^{M} \left[ g_{\lambda,\text{avg}}^{(*jk)}(\mathbf{x}) - g_{\lambda,\text{avg}}^{j}(\mathbf{x}) \right]^2 \tag{7.23}$$

where

$$g_{\lambda,\text{avg}}^{j}(\mathbf{x}) = \frac{1}{M} \sum_{k=1}^{M} g_{\lambda,\text{avg}}^{(*jk)}(\mathbf{x}) \tag{7.24}$$

Finally, the average model variance is estimated by taking the average of all $\hat{\sigma}_j^{2*}(\mathbf{x})$:

$$\hat{\sigma}_m^2(\mathbf{x}) = \frac{1}{P} \sum_{j=1}^{P} \hat{\sigma}_j^{2*}(\mathbf{x}) \tag{7.25}$$

This procedure is not computationally expensive since there is no need to train new networks. Hence, the complexity of both methods is similar and depends on the number $B$ of the wavelet networks that must be trained.

Following the same assumptions as in the bagging method, confidence intervals can be constructed. Since a good estimator of the model variance is obtained, the improved confidence intervals using the balancing methods are given by

$$g_{\lambda,\text{avg}}(\mathbf{x}) - z_{\alpha/2}\hat{\sigma}_m(\mathbf{x}) \leq f(\mathbf{x}) \leq g_{\lambda,\text{avg}}(\mathbf{x}) + z_{\alpha/2}\hat{\sigma}_m(\mathbf{x}) \tag{7.26}$$

where $z_{\alpha/2}$ can be found in a standard Gaussian distribution table and $1 - a$ is the confidence level desired.

## CONSTRUCTING PREDICTION INTERVALS

To generate prediction intervals, the distribution of the accuracy of the network prediction to target values is needed. In other words, the variance of the distribution of

$$y - \hat{y} \equiv y - g_\lambda\left(\mathbf{x}, \hat{\mathbf{w}}_n\right) \tag{7.27}$$

must be estimated. To construct prediction intervals, the total variance of the prediction, $\sigma_p^2$, must be estimated. As presented earlier, the total variance of the prediction is the sum of the model variance and the data noise variance. In the preceding section

a method for estimating the model variance was presented. Here we emphasize a method for estimating the data noise variance.

### The Bagging Method

First, the algorithm for creating predicting intervals using the bagging method is presented. To estimate the noise variance $\sigma_\varepsilon^2$, maximum likelihood methods are used. More precisely, a wavelet network will be trained on the residuals. An analytical description of the algorithm follows.

First, the initial wavelet network $g_\lambda(\mathbf{x}; \hat{\mathbf{w}}_n)$ is estimated and the solution $\hat{\mathbf{w}}_n$ of the loss function is found. Since it is assumed that the estimated wavelet network is a good approximation of the unknown underlying function, the vector $\hat{\mathbf{w}}_n$ is expected to be very close to the true vector $\mathbf{w}_0$ that minimizes the loss function.

In the next step the residuals between the network output and the target values are estimated. The noise variance can be approximated by a second wavelet network, $f_v(\mathbf{x}; \hat{\mathbf{u}}_n)$, where the squared residuals of the initial wavelet network are used as target values (Satchwell, 1994). In the second wavelet network, $f_v(\mathbf{x}; \hat{\mathbf{u}}_n)$, $v$ is the number of hidden units and $\hat{\mathbf{u}}_n$ is the estimated vector of parameters that minimizes the loss function of the second wavelet network. Since it is assumed that the estimated wavelet network is a good approximation of the unknown underlying function, the vector $\hat{\mathbf{u}}_n$ is expected to be very close to the true vector $\mathbf{u}_0$ that minimizes the loss function. The following cost function is minimized in the second network:

$$\sum_{i=1}^{n} \left\{ [g_\lambda(\mathbf{x}_i; \mathbf{w}_0) - y_i]^2 - f_v(\mathbf{x}_i; \mathbf{u}_0) \right\}^2 \tag{7.28}$$

and for a new set of observations, $\mathbf{x}^*$, that were not used in the training, we have that

$$\hat{\sigma}_\varepsilon^2(\mathbf{x}^*) \approx f_v(\mathbf{x}^*; \mathbf{u}_0) \tag{7.29}$$

This technique assumes that the residual errors are caused by variance alone (Carney et al., 1999). To estimate the noise variance, data that were not used in the training of the bootstrapped sample should be used. One way to do this is to divide the data set in a training and a validation set. However, leaving out these test patterns is a waste of data (Heskes, 1997). Alternatively, an unbiased estimation of the output of the wavelet network, $\hat{y}_{\text{ub}}(\mathbf{x})$, can be approximated by

$$\hat{y}_{\text{ub}}(\mathbf{x}) = \frac{\sum_{i=i}^{B} q_i^m \hat{y}_i(\mathbf{x})}{\sum_{i=i}^{B} q_i^m} \tag{7.30}$$

where $q_i^m$ is zero if the pattern $m$ appears on the $i$th bootstrap sample and 1 otherwise. Constructing the new network $f_v(\mathbf{x}; \mathbf{u})$, we face the problem of model selection again. Using the methodology described in the preceding section, the correct number of $v$ hidden units is selected. Usually, 1 or 2 hidden units are sufficient to model the residuals.

Using the estimation of the noise variance, the total variance can be calculated. The prediction intervals can be constructed using the following relationship:

$$g_{\lambda,\text{avg}}(\mathbf{x}^*) - t_{\alpha/2}\hat{\sigma}_p(\mathbf{x}^*) \leq f(\mathbf{x}^*) \leq g_{\lambda,\text{avg}}(\mathbf{x}^*) + t_{\alpha/2}\hat{\sigma}_p(\mathbf{x}^*) \tag{7.31}$$

where $t_{\alpha/2}$ can be found in a Student's $t$ distribution table and $1 - a$ is the confidence level desired.

### The Balancing Method

As in the case of confidence intervals, the balancing method can be used to improve the accuracy of the intervals. The algorithm for estimating the noise variance is the same as in the case of the bagging method. However, the difference lies in the estimation of the model variance. Hence, if the balancing method is used, the prediction intervals are given by

$$g_{\lambda,\text{avg}}(\mathbf{x}^*) - z_{\alpha/2}\hat{\sigma}_p(\mathbf{x}^*) \leq f(\mathbf{x}^*) \leq g_{\lambda,\text{avg}}(\mathbf{x}^*) + z_{\alpha/2}\hat{\sigma}_p(\mathbf{x}^*) \tag{7.32}$$

where $z_{\alpha/2}$ can be found in a standard Gaussian distribution table and $1 - a$ is the confidence level desired.

## EVALUATING THE METHODS FOR CONSTRUCTING CONFIDENCE AND PREDICTION INTERVALS

In this section the bagging and balancing methods are evaluated in constructing confidence and prediction intervals. The two methods will be tested in the two functions $f(x)$ and $g(x)$, given by (3.15) and (3.17), respectively.

The confidence intervals are presented for the first function in Figure 7.3. The first part of the figure presents the confidence intervals using the bagging method, and the

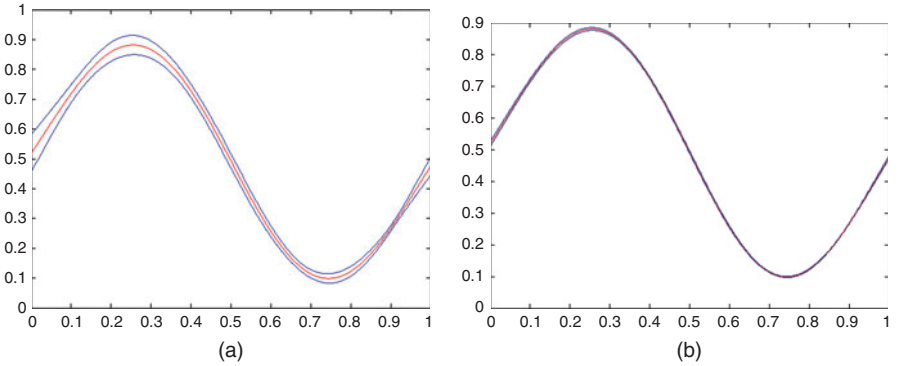


***Figure 7.3*** *Confidence intervals for the first case using the bagging (a) and balancing (b) methods.*

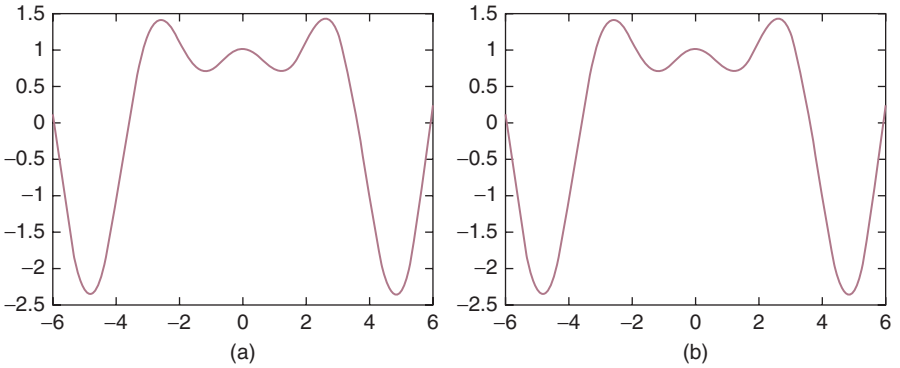


***Figure 7.4*** *Confidence intervals for the second case using the bagging (a) and balancing (b) methods.*

second part presents the confidence intervals using the balancing method. Similarly, Figure 7.4 presents the confidence intervals for the second function, where the first part refers to the bagging method, and the second part refers to the balancing method. It is clear that the confidence intervals using the balancing method are significantly narrower. This is due to the biased model variance estimator of the bagging method, which results in overestimation of the confidence intervals (Carney et al., 1999).

The 95% prediction intervals of the first function, $f(x)$, are presented in Figure 7.5. Again, the first part refers to the bagging method, and the second part refers to the balancing method. It is clear that both methods were able to capture the change in the variance of the noise. In both cases a wavelet network with 2 hidden units was used to approximate function $f(x)$, and a wavelet network with 1 hidden unit was used to approximate the residuals in order to estimate the noise variance. To compare the two methods, the prediction interval correct percentage (PICP) is used. PICP is the percentage of data points contained in the prediction intervals. Since the 95%

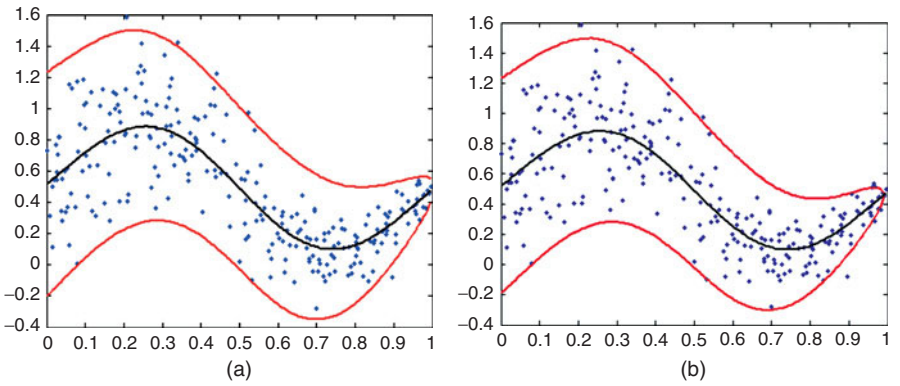


***Figure 7.5*** *Prediction intervals for the first case using the (a) bagging (PICP = 98%) and (b) balancing (PICP = 95%) methods.*

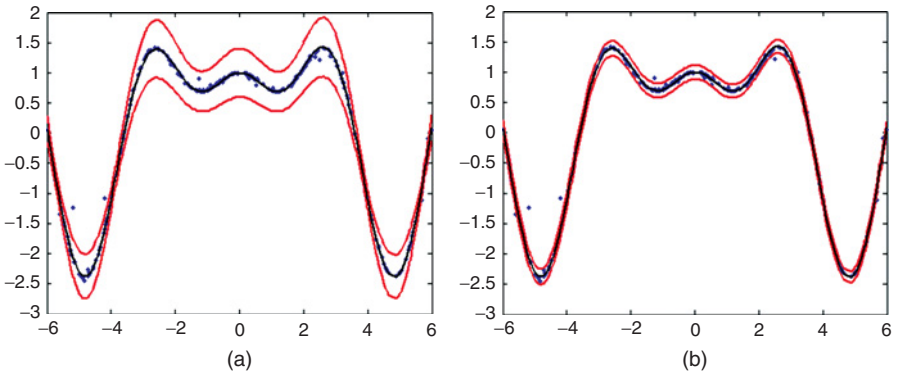


***Figure 7.6*** *Prediction intervals for the second case using the (a) bagging (PICP = 98.33%) and (b) balancing (PICP = 97.33%) methods.*

prediction intervals were estimated, a value of PICP close to 95 is expected. The bagging prediction intervals contain 98% of the data points (PICP), while in the case of the balancing method the PICP = 95% and is equal to the nominal value of 95%.

The same analysis is then repeated for the second function, $g(x)$. The 95% prediction intervals of $g(x)$ are presented in Figure 7.6. The first part refers to the bagging method and the second part refers to the balancing method. In both cases a wavelet network with 8 hidden units was used to approximate function $g(x)$, and a wavelet network with 2 hidden units was used to approximate the residuals in order to estimate the noise variance. As in the preceding case, the two methods are compared using the PICP. For the bagging method the PICP = 98.33%, while for the balancing method the PICP = 97.33%.

It is clear that the balancing method produced an improved estimator of the model variance. Our results are consistent with those of Breiman (1996), Carney et al. (1999), Heskes (1997), Papadopoulos et al. (2000), Zapranis and Livanis (2005), and Ζαπράνης (1999). In all cases the intervals produced by the balancing method were significantly smaller, while the PICP was considerable improved and closer to its nominal value.

## CONCLUSIONS

In this chapter we described the main methodologies mentioned in the literature to estimate confidence intervals and prediction intervals for nonlinear nonparametric wavelet neural networks. In real applications, researchers and practitioners are generally more interested in prediction intervals. This is due to the fact that the prediction intervals are associated with the accuracy with which we can predict future prices and not just limited to the assessment of the correctness of the estimation of the actual function.

The model variance and the data noise variance are assumed to be independent. The total variance of the prediction is given by the sum of two variances. We assumed that the variance is not constant but changes over time.

Although the maximum likelihood method is considered biased, it can be used to estimate the variance, which depends on the network's input vector. Methods of random repetitive sampling, such as bootstrapping, make no assumptions about the nature of the noise; they can only estimate the uncertainty of the model variance and thus cannot be used to construct predictive intervals but only confidence intervals. For the construction of prediction intervals, an estimation of noise variance is needed. To approximate the noise variance a second wavelet network is trained, with the squared residuals of the initial wavelet network used as the target values.

We described two methods for constructing prediction and confidence intervals: bagging and balancing. Our results indicate that the balancing methods provide narrower confidence intervals and prediction intervals that produce more accurate PICP. The disadvantage of the iterative methodologies is that they are computationally expensive. In any case, analysis of the various approaches for estimating confidence and prediction intervals is an extremely interesting field of research.

## REFERENCES

Breiman, I. (1996). "Bagging predictors." *Machine Learning*, 24, 123–140.

Carney, J. G., Cunningham, P., and Bhagwan, U. (1999). "Confidence and prediction intervals for neural network ensembles." *IJCNN'99*, Washington, DC.

De Veaux, D. R., Schumi, J., Schweinsberg, J., and Ungar, H. L. (1998). "Prediction intervals for neural networks via nonlinear regression." *Technometrics*, 40(4), 273–282.

Heskes, T. (1997). "Practical confidence and prediction intervals." *Advances in Neural Information Processing Systems*, 9, 176–182.

Papadopoulos, G., Edwards, J. P., and Murray, F. A. (2000). "Confidence estimation methods for neural networks: a practical comparison." *ESANN*, Bruges, Belgium, 75–80.

Satchwell, C. (1994). "Neural networks for stochastic problems: more than one outcome for the input space." *Neural Computing Applications Forum Conference*, Aston University, Birmingham, UK.

Zapranis, A. D., and Livanis, E. (2005). "Prediction intervals for neural network models." *Proceedings of the 9th WSEAS International Conference on Computers*, World Scientific and Engineering Academy and Society, Athens, Greece, 1–7.

Zapranis, A., and Refenes, A. P. (1999). *Principles of Neural Model Indentification, Selection and Adequacy: With Applications to Financial Econometrics*. Springer-Verlag, New York.

Ζαπράνης, Α. (2005). *Χρηματοιοκονομική και Νευρωνικά Συστήματα*, Κλειδάριθμος, Αθήνα.