Adaptive learning of rational expectations using neural networks

Maik Heinemann*

University of Hannover, Institute of Economics, Königsworther Platz 1, D-30167 Hannover, Germany

Abstract

This paper investigates how adaptive learning of rational expectations may be modeled with the help of neural networks. Necessary conditions for the convergence of the learning process towards (approximate) rational expectations are derived using a simple nonlinear cobweb model. The results obtained are similar to results obtained within the framework of linear models using recursive least squares learning procedures. In general, however, convergence of a learning process based on a neural network may imply that the resulting expectations are not even local minimizers of the mean-squared prediction error. © 2000 Elsevier Science B.V. All rights reserved.

JEL classification: C 45; D 83

Keywords: Adaptive learning; Neural networks; Rational expectations

1. Introduction

Dynamic economic models usually require that agents within the model form expectations regarding future values of economic variables. Assumptions about how the agents form their expectations may then have far reaching consequences for conclusions derived from a particular model. The hypothesis of rational expectations states that the agents’ knowledge of their economic environment is such that the objective distributions of the relevant variables are
known to them. Thus, their expectations are based upon this objective distributions. In this extreme form, the hypothesis of rational expectations is naturally subject to criticism. As Sargent (1993, p. 3) remarks: ‘[…] rational expectations models impute much more knowledge to the agents within the model […] than is possessed by an econometrician, who faces estimation and inference problems that agents in the model have somehow solved’. Hence the question is, whether it is possible to establish rational expectations as a result of learning processes, without imposing such strong assumptions regarding the knowledge of the agents within the model.

One possible approach is to assume that the agents do not have any prior knowledge about the objective distributions of relevant variables, but instead are equipped with an auxiliary model describing the perceived relationship between these variables. In this case, agents are in a similar situation as the econometrician in the above quotation: They have to estimate the unknown parameters of their auxiliary model in order to form the relevant expectations on the basis of this model. This is but one possible notion of bounded rationality that is discussed in the economics literature. There exist many other approaches that try to capture the idea of limited knowledge and limited information processing capabilities of economic agents (cf. Conlisk, 1996). Nevertheless, in this paper I will refer to the above-described framework as the ‘boundedly rational learning approach’.

In the boundedly rational learning framework, the auxiliary model of the agents is correctly specified at best in the sense, that it correctly depicts the relationship between the relevant variables within the rational expectations equilibrium, but the model is misspecified during the learning process. While this implies that there may exist values for the parameters that result in rational expectations, the estimation problem faced by the agents is quite different from usual estimation problems, because the expectations of the agents itself affect the data underlying the estimation of parameters (‘forecast feedback’). This means that during the learning process the relationship between the variables observed by the agents, will change as long as the agents change their expectations scheme. The question, whether or not the estimated parameters will converge toward parameter values implying rational expectations is thus not a trivial one, because usual approaches to prove consistency of estimators are not applicable.

In case of linear models, i.e. models where the rational expectations equilibrium is a linear function of exogenous and lagged endogenous variables, recursive least squares can be used to estimate the parameters of the auxiliary model. Regarding this case, there are a number of contributions, where conditions for the converge of the learning process towards rational expectations are derived (cf. Bray, 1983; Bray and Savin, 1986; Fourgeaud et al., 1986; Marcet and Sargent, 1989).

In the nonlinear case, when the rational expectations equilibrium is not representable by a linear function of exogenous and lagged endogenous
variables, the analysis of adaptive learning procedures is more complicated. The problem is that the boundedly rational learning approach requires assumptions regarding the auxiliary model of the agents. It is obvious that the assumption of a correctly specified auxiliary model is a very strong one, because such an assumption presupposes extraordinary a priori knowledge of the agents regarding their economic environment.¹ But the removal of this assumption requires that the auxiliary model of the agents is flexible enough to represent various kinds of possible relationships between the relevant variables.

One way to achieve this flexibility is to assume that the auxiliary model of the agents is a neural network. Neural networks might be well suited for this task because they are, as will be shown below, able to approximate a wide class of functions at any desired degree of accuracy.² Thus, if the auxiliary model of the agents is given by a neural network, they may possibly be able to learn the formation of rational expectations, without the requirement of knowledge regarding the specific relationship of the relevant variables.

However, the present paper demonstrates that such a confidence in the power of neural networks might be too optimistic: First, if the agents in the model learn by estimating parameters of an auxiliary model, feedback problems will arise that might prevent the convergence of the learning process. As is shown below, it depends on the nature of the true model, i.e. upon how expectations affect actual outcomes, whether convergence and thus learning of rational expectations can occur. Second, there are still specification problems present, because the principle ability of neural networks to approximate a wide class of functions does not mean that a concrete specified network at hand is able to do so. A neural network that is misspecified in this respect will clearly not enable the agents to learn rational expectations. The best one can hope for is that convergence towards approximate rational expectations occurs, i.e. expectations that are merely local minimizers of the mean squared prediction error. The paper shows, that the convergence to such approximate rational expectations cannot be ruled out even if the neural network is correctly specified. Moreover, in general nonlinear models, neural network learning may result in expectations, that are not even local minimizers of the mean-squared prediction error.

At this point it is worth to be emphasized that neural networks represent but one among many other procedures that can be used to approximate an unknown function. Nevertheless, neural networks have gained much popularity recently, especially in econometrics, and are sometimes treated as if they do not face any specification problems themselves. It should be noted, however, that

¹ A similar argument can be found in Salmon (1995).
² For a concise survey on neural networks see Sargent (1993); an introduction into this line of research is offered by Müller and Reinhardt (1990). Kuan and White (1994b) discuss neural networks from an econometric viewpoint.
many of the results that are derived below carry over to other approximation procedures as well. In the following, a model is formulated, where agents have to form expectations and use a neural network as their auxiliary model. In the next section, a simple reduced form equation will be specified, where the value taken by an endogenous variable depends on its expected value and the values taken by exogenous variables. After that, the neural network used as auxiliary model of the agents in the model will be described. Then the learning algorithm will be formulated and the question will be investigated, whether or not the agents are able to form (approximate) rational expectations using neural networks. Finally, a more general reduced form is considered in order to show that some of the results are due to the very special structure of the model analyzed so far.  

2. The model

2.1. A reduced form equation for the endogenous variable

Consider a model, where agents have to form expectations regarding an endogenous variable, whose value depends on observable exogenous variables and an unobservable error. The reduced form is given by:

\[ p_t = z p^e_t + g(x_t) + \varepsilon_t. \]

Here \( p^e_t \) denotes the agents’ expectation of the endogenous variable \( p \) in period \( t \) and \( p_t \) denotes the actual value the endogenous variable takes in period \( t \). \( x_t \) is a \( k \)-dimensional vector of exogenous variables, which can be observed before the expectation \( p^e_t \) is formed. It is assumed that \( x_t \) is for all \( t \) a vector of independent and identically distributed random variables. Additionally it is assumed that \( x_t \) is bounded for all \( t \), i.e. \( x_t \) takes only values in a set \( \Omega_x \subset \mathbb{R}^k \). \( \varepsilon_t \) is, like the elements of \( x_t \), for all \( t \) an independent and identically distributed random variable that satisfies \( \mathbb{E}[\varepsilon_t] = 0 \), \( \mathbb{E}[\varepsilon_t^2] = \sigma^2 \) and \( \mathbb{E}[\varepsilon_t | x_t] = 0 \). Like \( x_t \), \( \varepsilon_t \) is bounded for all \( t \) and takes values in a set \( \varepsilon_t \in \Omega_\varepsilon \subset \mathbb{R} \). Contrary to the elements of \( x_t \), \( \varepsilon_t \) is not observable by the agents, such that expectations regarding the

---

3 The work of Salmon (1995) and Kelly and Shorish (1994) deals at least partially with a similar question like this paper. Both papers take neural networks as the basis of learning processes. While the conclusions of Salmon (1995) exclusively rely upon simulation results, Kelly and Shorish (1994) use a learning rule, which makes their results not comparable to those obtained here.

4 Eq. (1) may be interpreted as the reduced form of a cobweb model if the endogenous variable \( p \) denotes the price of the relevant good: Given the supply function \( y'(p^e) = ap^e \) and the demand function \( y(p, x_t, \varepsilon_t) = b[p_t - g(x_t) + \varepsilon_t] \), market clearing implies that \( p_t = - (a/b)p^e_t + g(x_t) + \varepsilon_t \).
endogenous variable cannot be conditioned on this variable. Finally, the function \( g(x) \) is a continuous function for all \( x \in \Omega_x \).  

Reduced form (1) may be viewed as a special case of a more general class of nonlinear models, where the value of the endogenous variable \( p \) in period \( t \) is given by \( p_t = G(z_t^e, y_t, \varepsilon_t) \). Here \( z_t^e \) is a vector of expectations regarding future values of the endogenous variable, \( y_t \) is a vector of variables predetermined in period \( t \), \( \varepsilon_t \) is an unobservable error and \( G \) may be any continuous function. Note that \( y_t \) may contain exogenous as well as lagged endogenous variables. As shown by Kuan and White (1994a), the methodology used here to analyze learning processes based on the reduced form (1), can also be used to analyze this more general case. But contrary to the simple model (1), the required restrictions are quite abstract and it is not possible to derive conditions for convergence, which may be interpreted from an economic viewpoint. A more general reduced form than (1) will be considered below.

Given the reduced form (1) and because \( \text{E}[\varepsilon_t | x_t] = 0 \), rational expectations are given by:

\[
\bar{p}_t^e = \text{E}[p_t | x_t] = \text{E}\left[ \frac{g(x_t) + \varepsilon_t}{1 - \alpha} | x_t \right] = \frac{g(x_t)}{1 - \alpha} = \phi(x_t).
\]

As long as \( \alpha \neq 1 \), there exists a unique rational expectation of \( p_t \) whatever value is taken by the exogenous variables \( x_t \). In what follows, \( \phi(x_t) \) denotes the rational expectations function that gives this unique rational expectation of the endogenous variable for all \( x \in \Omega_x \).

It is obvious that the agents may not be able to form rational expectations if they do not know the reduced form of the model and especially the form of the function \( g(x_t) \). The question is, whether the agents can learn to form rational expectations, using observations of the variables \( p_{t-1}, p_{t-2}, \ldots \) and \( x_{t-1}, x_{t-2}, \ldots \). This means that the agents are at least aware of the relevant variables that determine the endogenous variable \( p_t \) in a rational expectations equilibrium. It is assumed that the agents have an auxiliary model at their disposal representing the relationship between the exogenous variables and the endogenous variable.

If \( g(x_t) \) is linear in \( x_t \), reduced form (1) becomes the linear model \( p_t = \alpha p_t^e + \beta'x_t + \varepsilon_t \), where \( \beta \) is a \( k \)-dimensional vector of parameters. This is the linear case mentioned in the introduction. Assuming that agents use the auxiliary model \( p = \delta'x \) to represent the relationship between \( x \) and \( p \) and that the

---

5 Zenner (1996) gives a survey of boundedly rational learning models. Following his classification, the model considered here is the simplest form of a static model, because the exogenous variables are assumed to be stationary and serially uncorrelated.
parameters $\delta$ are estimated using recursive least squares, the following can be shown (Bray and Savin, 1986; Marcet and Sargent, 1989):

(a) If the estimator $\hat{\delta}$ for $\delta$ converges, this results in rational expectations, i.e. $\hat{\delta} = (1 - \alpha)^{-1}\beta^*$.

(b) The estimator for $\delta$ will converge towards $(1 - \alpha)^{-1}\beta^*$ if and only if $\alpha < 1$.

As the function $g(x_t)$ from (1) does not need to be linear in $x_t$, the rational expectations function $/\phi(x_t)$ needs not to be a linear function too. Thus, if one assumes that agents use parametrically specified auxiliary models and have no prior knowledge regarding the functional form of $/\phi(x_t)$, it would be advisable to take an auxiliary model that is flexible enough to approximate various possible functional forms at least sufficiently well. The next subsection establishes that neural networks may be auxiliary models having the desired property.

### 2.2. Hidden layer neural networks

In the following a special form of neural networks is considered. These networks are called hidden layer neural networks and a typical structure is depicted in Fig. 1. In this type of networks, information proceeds only in one direction and there exists only one layer of hidden units between the input units and the output units.

Each of the hidden units $i = 1, \ldots, m$ receives a signal that is the weighted sum of all inputs $x_j$, $j = 1, \ldots, k$. This implies that $\vec{h}_i = \sum_{j=1}^{k} w_{i,j} x_j$ is the signal received by the hidden unit $i$, where $w_{i,j}$ for $i = 1, \ldots, m$ represents a threshold value. In each hidden unit the signal received is transformed by an activation function $S$, such that $h_i = S(\vec{h}_i)$ is the output signal of the hidden unit $i$. Finally, the output unit receives the weighted sum of all these output signals, such that $y = \sum_{i=1}^{m} h_i + q_0$ is the output of the neural network, where $q_0$ represents a threshold value. The so-defined mapping from inputs $x_j$ to the output $y$ given by the so-specified network may be written as follows:

$$y = q_0 + \sum_{i=1}^{m} q_i S \left( w_{i,0} + \sum_{j=1}^{k} w_{i,j} x_j \right)$$

$$= f(x, \theta),$$

where $x = (x_1, \ldots, x_k)$ and $\theta = (q_0, q_1, w_{1,0}, \ldots, w_{1,k}, q_2, \ldots, w_{m,k})$. If the neural network contains $k$ input units and $m$ hidden units, $\theta$ is a $q$-dimensional vector, where $q = 1 + m (k + 2)$.

A usual choice for the activation function is a continuous, monotone function, mapping the real numbers into the interval $[0,1]$. A function that exhibits this properties is

$$S(z) = \frac{1}{1 + \exp(-z)}, \quad z \in \mathbb{R}.$$  

(4)
What makes this kind of neural networks interesting with respect to the problem of learning is their ability to approximate nearly any function.\textsuperscript{6}

\textit{Theorem 2.1} (Hornik et al., 1989, Theorem 2.4). \textit{The neural network (3) with one layer of hidden units and activation functions given by (4) is able to approximate any continuous function defined over some compact set to any desired accuracy if the network contains a sufficient number of hidden units.}

Theorem 2.1 guarantees the existence of a neural network such that a suitably specified distance between a given continuous function and this network becomes arbitrarily small. Thus, in what follows ‘perfect approximation’ means that we have such a neural network and a vector of parameters $\theta^*$ such that $\phi(x_t) = f(x_t, \theta^*)$ almost surely. Unfortunately, the theorem makes no statement regarding the number of hidden units required in order to obtain such a perfect approximation. This means that given a neural network (3) with $m$ hidden units, it is by no means guaranteed that a perfect approximation of the rational expectations function $\phi(x_t)$ is possible, i.e. the misspecification of a neural network cannot be ruled out a priori.

\subsection{The objectives of learning}

If it is assumed that the auxiliary model used by the agents in the model is a neural network of form (3), the expectation of $p$ given a vector of parameters

\textsuperscript{6}There exist several versions of this theorem in the neural networks literature. See e.g. Cybenko (1989) and Funahashi (1989).
and observations $x$ of the exogenous variables is given by $p^e = f(x, \theta)$. In general, this expectation turns out to be incorrect and the agents may wish to change the values of the parameters in order to enhance the predictive power of their model. This is what we call learning. Before this learning process is analyzed in more detail, it useful to be more precise regarding the objectives of learning.

It seems plausible to use the mean-squared error of the expectations as a measure for the success of learning. This mean-squared error is defined as the expected value of the squared deviation of the agents' expectation about the endogenous variable $p^e = f(x, \theta)$ from its actual value, which is given by $p = x f(x, \theta) + g(x) + \varepsilon$. Denoting this mean-squared error as $\lambda(\theta)$ we get

$$\lambda(\theta) = E[(x f(x, \theta) + g(x) + \varepsilon - f(x, \theta))^2] = (1 - x)^2 E\left[\frac{g(x) + \varepsilon}{1 - x} - f(x, \theta)\right]^2 = (1 - x)^2 E\left[\phi(x) + \frac{\varepsilon}{1 - x} - f(x, \theta)\right]^2.$$ (5)

A plausible learning objective is to search for a vector $\theta$ that minimizes $\lambda(\theta)$. Such an optimal vector of parameters $\theta^*$ satisfies

$$\theta^* = \arg\min_{\theta \in \Theta^e} \lambda(\theta)$$ (6)

and the necessary condition for the associated optimization problem is given by

$$\nabla_\theta \lambda(\theta) = 2(1 - x)^2 E\left\{\nabla_\theta f(x, \theta) \left[\phi(x) + \frac{\varepsilon}{1 - x} - f(x, \theta)\right]\right\} = 0.$$ (7)

Note that the solution of (7) does not need to be unique. If there exists a vector of parameters satisfying the condition $\nabla_\theta \lambda(\theta) = 0$, it results in a (local) minimum of the mean-squared error if the Jacobian matrix

$$J_\lambda(\theta) = \nabla_\theta^2 \lambda(\theta) = -E\left\{\nabla_\theta^2 f(x, \theta) \left[\phi(x) + \frac{\varepsilon}{1 - x} - f(x, \theta)\right]\right\} + E\{\nabla_\theta f(x, \theta) \nabla_\theta f(x, \theta)^T\}$$ (8)

is positive semidefinite. A (local) minimum at $\theta^*$ is (locally) identified if $J_\lambda(\theta^*)$ is positive definite. Otherwise, at least one eigenvalue of $J_\lambda(\theta^*)$ equals zero, such that the minimum is not (locally) identified. This means that there exists a neighborhood of $\theta^*$ with vectors $\theta$ that imply an identical value for the mean-squared error $\lambda(\theta^*)$.

---

7 There exist economic models where this measure is not appropriate, because this mse will not be smallest in the rational expectations equilibrium. However, in the model considered here this is not an issue.
Now define the set
\[
\Theta^L = \{ \theta \in \mathbb{R}^q \mid \nabla \phi \lambda (\theta) = 0, \quad J_\lambda (\theta) \text{ is positive semidefinite} \},
\]
i.e. \( \Theta^L \) is the set of all vectors of parameters for the neural network implying a – possibly not identified – (local) minimum of the mean-squared error \( \lambda (\theta) \).

If the neural network is able to give a perfect approximation of the unknown rational expectations function \( \phi (x) \), there exist vectors of parameters implying \( \lambda (\theta) = \sigma^2 \). Note that in the neural network specified here, all hidden units use identical activation functions, such that there will be no unique vector having this property. Let \( \Theta^G = \{ \theta \in \mathbb{R}^q \mid \lambda (\theta) = \sigma^2 \} \) denote the set of all these vectors of parameters. Obviously, \( \Theta^G \) is a subset of \( \Theta^L \). \( \Theta^G \) will be empty if the number of hidden units in the neural network is not sufficient to give a perfect approximation of \( \phi (x) \).

Any \( \theta \in \Theta^G \) implies that the expectations formed on the basis of the neural network coincide with rational expectations according to (2). This is not true for the remaining vectors of parameters \( \theta \in \Theta^L \setminus \Theta^G \). All these vectors of parameters result in (local) minima of the mean squared error \( \lambda (\theta) \), but they do not imply \( \phi (x) = f (\theta, x) \) almost surely. These vectors of parameters merely result in more or less accurate approximations of the unknown rational expectations function. Thus, following Sargent (1993), the resulting equilibria will be labeled as approximate rational expectations equilibria.\(^8\)

3. Learning of (approximate) rational expectations

As mentioned above, learning in the model considered here means that agents estimate the parameters of their auxiliary model on the basis of observations of exogenous and endogenous variables. The question, whether the agents are able to learn to form rational expectations is thus equivalent to the question, whether their estimation procedure yields asymptotically correct parameter values. This is in turn equivalent to estimated parameter vectors that converge to a \( \theta \in \Theta^G \). If the neural network used by the agents is not able to give a perfect approximation of the unknown function \( \phi (x) \), the question is, whether there will at least result approximate rational expectations, i.e. whether the estimated parameter vectors converge to a \( \theta \in \Theta^L \).

Given a vector of parameters \( \theta_t \), the expectation function in period \( t \) is given as \( p_t = f (x_t, \theta_t) \). Thus, from reduced form (1) the actual value of the endogenous variable in \( t \) is obtained as
\[
p_t = z f (x_t, \theta_t) + g(x_t) + \varepsilon_t.
\]

\(^8\) Properly speaking, as noted by Kuan and White (1994a), the label ‘incorrect belief equilibria’ would be more appropriate.
If \( f(x_t, \theta_t) \neq \phi(x_t) \), the agents' expectation turns out to be incorrect and the value taken by the endogenous variable in \( t \) diverges from its value in the rational expectations equilibrium.

In what follows it is assumed that the learning algorithm used by the agents is the so called 'backpropagation algorithm', a widely used algorithm in the neural network literature.\(^9\) This algorithm modifies the vector of parameters \( \theta_t \) according to the product of the actual expectational error \( p_t - p_t^\varepsilon = p_t - f(x_t, \theta_t) \) and the gradient of the neural network with respect to \( \theta_t \):

\[
\theta_{t+1} = \theta_t + \gamma_{t+1} \left[ \nabla_{\theta} f(x_t, \theta_t)(p_t - f(x_t, \theta_t)) \right].
\]

(9)

Here \( \gamma_t \) is a learning rate that satisfies \( \gamma_t = t^{-\kappa}, 0 < \kappa \leq 1 \). The declining learning rate implies that the modifications of \( \theta_t \) according to (9) become smaller over time. This is required in order to derive conclusions regarding the convergence of the vector of parameters.

The question, whether agents will asymptotically learn to form (approximate) rational expectations is now equivalent to the question, whether \( \theta_t \) according to (9) will converge to a \( \theta \in \Theta^L \). However, Eq. (9), describing the evolution of the vector of parameters \( \theta \), represents a nonlinear, nonautonomous and stochastic difference equation. The analysis of such a difference equation is a quite complex task. But as shown by Ljung (1977), the asymptotic properties of \( \theta_t \) may be approximated with the help of the differential equation:\(^{10}\)

\[
\dot{\theta}(\tau) = \bar{Q}(\theta(\tau)),
\]

(10)

where\(^{11}\)

\[
\bar{Q}(\theta) = E\{ \nabla_{\theta} f(x, \theta) [p - f(x, \theta)] \}
= E\{ \nabla_{\theta} f(x, \theta) [g(x) + \varepsilon - (1 - z)f(x, \theta)] \}.
\]

Ljung (1977) specifies assumptions necessary for this approximation to be valid. Since the assumptions regarding reduced form (1) of the model considered here, imply that \( x_t \) as well as \( e_t \) are independent and identically distributed random variables, these assumptions are satisfied here.\(^{12}\)

\(^{9}\) See on this White (1989). Sometimes, this algorithm is labeled as 'generalized delta rule'. Note that (9) is nothing more than a stochastic gradient algorithm.

\(^{10}\) Detailed descriptions of this methodology can be found in Marcet and Sargent (1989) and Sargent (1993).

\(^{11}\) Assume for instance the linear model from Section 2, where \( g(x) = x'\beta \) and assume further that the agents use a linear auxiliary model, such that \( p_t^\varepsilon = x_t'\theta_t \). In this case we have \( \nabla_{\theta} f(x, \theta) = x \). With \( M_x \) denoting the \((k \times k)\) matrix of moments \( E[xx'] \), the \( k\)-dimensional differential equation is given by \( \dot{\theta} = M_x[\beta - (1 - z)\theta] \).

\(^{12}\) This is shown in detail by White (1989). As noted above, Kuan and White (1994a) analyze more general models and derive conditions for this approximation to be valid in such a case too.
Note that (10) is a deterministic differential equation. This means that all results derived with the help of (10), regarding the stochastic difference equation (9) are valid only in a probabilistic sense. The respective theorems are derived by Ljung (1977). They are consequences of the fact that the time path of $\theta_t$ according to (9) is asymptotically equivalent to the trajectories of $\theta$ resulting from the differential equation (10). This implies that for $t \to \infty$, $\theta_t$ from (9) will – if ever – converge only to stationary points of (10). Moreover, this implies that the probability for such a convergence to occur is positive only if this stationary point is (locally) stable.

However, a remark is necessary, because the convergence results derived with the help of the associated differential equation are not valid if the respective stationary points belong to an unbounded continuum, i.e. some of the results stated by Ljung (1977) may be not valid if the stationary points are not identified. Although it is in general the case that the differential equation will exhibit continua of stationary points, there will always exist identified fixed points, unless the neural network is overparameterized, i.e. unless there are more hidden units than needed to ensure a perfect approximation of the unknown function $\phi(x)$.

But irrespective of this problem, at least the following statement about the learning algorithm (9) with the associated differential equation (10) is possible: the algorithm will only converge to parameter vectors that are stationary points of the associated differential equation and it will not converge to stationary points that are unstable.

Thus, if we want to analyze the asymptotic properties of the learning algorithm (9), the set of all stationary points of (10) is of special interest. Because $\theta$ is a constant, it follows that

$$
\bar{Q}(\theta) = E\{\nabla_\theta f(x, \theta)[g(x) + e - (1 - z)f(x, \theta)]\}
= (1 - z)E\left\{\nabla_\theta f(x, \theta) \left[\frac{g(x) + e}{1 - z} - f(x, \theta)\right]\right\}
= (1 - z)E\left\{\nabla_\theta f(x, \theta) \left[\phi(x) + \frac{e}{1 - z} - f(x, \theta)\right]\right\}
= -\frac{1}{2(1 - z)} \nabla_\theta \phi(x). \tag{11}
$$

13 Let $f_m(x, \theta)$ denote a neural network (3) with $m$ hidden units and assume that there exists a $\theta^* = (q_0^m, q_T^m, w^m_{1T}, w^m_{10})$ such that $\phi(x) = f_1(x, \theta^*)$ (a.s.), where $m = 1$ is the smallest number of hidden units that enables such a perfect approximation. Given the activation function (4), there exists a vector $\theta^{m*} = (q_0^m + q_T^m, -q_T^m, -w^m_{10}, -w^m_{11})$ such that $f_2(x, \theta^{m*}) = f_1(x, \theta^{m*})$ (a.s.), i.e. the stationary point $\theta^{m*}$ is not unique. Now consider the overparameterized neural network $f_2(x, \theta)$, where $m = 2$. For this network a vector $\theta^{*'} = (q_0^m, a, w^m_{1T}, w^m_{10}, q_T^m - a, w^m_{10}, w^m_{11})$, where $a$ can be any real number, implies $f_2(x, \theta^{*'}) = f_2(x, \theta^{*'})$ (a.s.) (it is possible to construct more such vectors, this is just an example). Thus, we get continua of stationary points. The same line of argument can be used to show that if for a network with $m$ hidden there exists a vector $\theta$ such that $\nabla_\theta \phi(x) = 0$, we get continua of stationary points in a neural network with $m + 1$ hidden units.
It is apparent from (11) that differential equation (10) is a gradient system, the potential of which is proportional to the mean-squared error $\hat{\lambda}(\theta)$ from (5).\textsuperscript{14} Hence:

**Proposition 3.1.** Any $\theta$ implying that the mean-squared error $\hat{\lambda}(\theta)$ from (5) takes an extreme value, is a stationary point of differential equation (10).

This proposition gives useful information regarding the set of all $\theta$ that might be convergence points for $\theta_t$ according to (9).\textsuperscript{15} We know that any $\theta \in \Theta^L$ represents a vector of parameters that is a possible result of the agents’ learning process.\textsuperscript{16} But by now, it is neither guaranteed that vectors of parameters contained in the set $\Theta^L$ are (locally) stable, nor excluded that the evolution of $\theta_t$ might be such that a (local) maximum of the mean-squared error is attained.

The conditions for local stability of a fixed point are usually stated with respect to the Jacobian matrix of $Q(\theta)$ evaluated at this fixed point. Thus, with respect to the learning algorithm (9), we get:

**Proposition 3.2.** Let $\theta^\ast$ be a stationary point of differential equation (10). The probability that for $t \to \infty$, $\theta_t$ according to (9), will converge to $\theta^\ast$ is positive only if the real parts of all eigenvalues of the Jacobian matrix

$$J(\theta^\ast) = \left. \frac{\partial Q(\theta)}{\partial \theta'} \right|_{\theta^\ast}$$

are nonpositive.

Because differential equation (10) is a gradient system, with potential $F(\theta) = 2(\beta - 1)^{-1} \nabla_{\theta} \hat{\lambda}(\theta)$, we get together with (8)

$$J(\theta) = (\beta - 1) J_{\hat{\lambda}}(\theta). \quad (12)$$

Let $\theta^\ast$ be any element from $\Theta^L$, i.e. $\theta^\ast$ implies that $\hat{\lambda}(\theta)$ attains a (local) minimum. In this case, $J_{\hat{\lambda}}(\theta^\ast)$ is positive semidefinite and thus all eigenvalues of

\textsuperscript{14} A dynamic system $\dot{\theta} = Q(\theta)$ is called a gradient system if there exists a function $F:\Theta \to \mathbb{R}$ such that $-\nabla_{\theta} F(\theta) = Q(\theta)$ (Hirsch and Smale, 1974). Here we have $F(\theta) = [2(1 - \beta)]^{-1} \hat{\lambda}(\theta)$.

\textsuperscript{15} Note that the set of all these vectors of parameters will in general contain only a subset of all stationary points of (10). So, for instance any saddle point of $\hat{\lambda}(\theta)$ is a stationary point of (10).

\textsuperscript{16} If the neural network is not able to give a perfect approximation of the unknown function $\phi(x_t)$, this means that the auxiliary model of the agents is misspecified. As shown by Kuan and White (1994a) and demonstrated in the next section, such misspecification will usually result in approximate rational expectations that are not minima of $\hat{\lambda}(\theta)$. In the simple model considered in this section, this case cannot occur.
$J_{\theta}(\theta^*)$ are nonnegative.\textsuperscript{17} From (12) follows that $J(\theta^*)$ will possess eigenvalues which are exclusively non positive, only if $\alpha - 1 < 0$. Otherwise if $\alpha - 1 > 0$, all eigenvalues of $J(\theta^*)$ are nonnegative.

Conversely, for all $\tilde{\theta}$ that imply a (local) maximum of $\lambda(\theta)$, $J_{\tilde{\theta}}(\tilde{\theta})$ will be negative semidefinite. Thus, all eigenvalues of $J(\tilde{\theta})$ are nonnegative only if $\alpha - 1 < 0$ and all eigenvalues are nonpositive if $\alpha - 1 > 0$.

Summarizing, we get:

**Proposition 3.3.** Let $\theta^*$ be any element of the set $\Theta^L$, i.e. $\theta^*$ implies a local minimum of the mean-squared error $\lambda(\theta)$. The probability that $\theta_t$ from (9) converges to $\theta^*$ asymptotically is positive only if $\alpha - 1 < 0$.

Note that $\Theta^L$ contains the rational expectations equilibrium if the neural network is able to give a perfect approximation of the unknown rational expectations function. Therefore, Proposition 3.3 states that this rational expectations function is ‘learnable’ with the help of a neural network only if $\alpha - 1 < 0$. Thus, we get a condition for convergence of the learning process toward (approximate) rational expectations which are equivalent to that derived in the context of linear models.

It is interesting to note that contrary to the results obtained in case of linear models described in Section 5 it is possible for the learning process to converge to vectors of parameters that will not imply rational expectations, even if the auxiliary model is correctly specified. Because $\alpha < 1$ implies that all (local) minima of the mean squared error are possible convergence points of the learning algorithm (9), learning may merely result in (approximate) rational expectations although the neural network is able to give a perfect approximation of the unknown rational expectations function.\textsuperscript{18}

The stability condition $\alpha < 1$ may be interpreted as follows: Assume that given a vector of parameters $\theta$, and a realization of the exogenous variables $x$, the expectation of the endogenous variable $p^e = f(x, \theta)$ (under) overestimates the actual value $p$. In this case, the learning algorithm (9) modifies the vector of parameters in a way that given $x$, there results a lower (higher) expectation. Whether the repetition of this procedure will result in the correct rational expectation given $x$, depends on the value of the parameter $\alpha$.

With $x$ given, the two cases can be distinguished: In case of Fig. 2a, with $\alpha < 1$, the nature of the learning algorithm described above, implies that the

\textsuperscript{17} As $J(\theta)$ is a real symmetric matrix, all eigenvalues of $J(\theta)$ are real.

\textsuperscript{18} The possible convergence of algorithms towards local rather than global optima in nonlinear environments is a well-known problem, that may be resolved by the method of simulated annealing (cf. White, 1989). Unfortunately, in case of feedback there are no convergence results available for such algorithms.
Fig. 2. Learnability of correct expectations.

Expectation error $p_t - p_t^e$ becomes smaller, whereas in case the of Fig. 2, with $\alpha < 1$, it implies that this expectation error becomes larger. In the first case, an algorithm based on this procedure may converge, while in the second case such an algorithm would never converge.

Proposition 3.3 formulates a necessary condition for learning process (9) to reach its objective. Reversing the underlying argument, we get:
Proposition 3.4. If $\alpha - 1 > 0$, only (local) maxima of $\lambda(\theta)$ are stable stationary points of differential equation (10) and thus possible convergence points for learning algorithm (9).

This proposition states that the goal of the learning process will be completely missed if $\alpha - 1 > 0$, because in this case the learning process steers towards (local) maxima of the mean squared error $\lambda(\theta)$. But as is easily shown, this implies that for $t \to \infty$ we have $\theta_t \to \infty$, because for any parameter space $\Theta \subset \mathbb{R}^q$, any maximum of $\lambda(\theta)$ will be on the closure of this set. In order to verify this statement it must be shown that within any such set, there exists no $\theta^*$ satisfying the sufficient conditions for a maximum: a maximum of $\lambda(\theta)$ implies $E[g(x) + \varepsilon - (1 - \alpha)f(x, \theta^*)] = 0$, because one element of $\theta^*$ represents a constant term. Assuming that this element is the first element of $\theta^*$, $\lambda(\theta^*)$ is a maximum only if we have for any sufficiently small $\mu$ and $\theta_0 = \theta^* + (\mu, 0, \ldots, 0)$ that $\lambda(\theta^*) \geq \lambda(\theta_0)$. But because $E[g(x) + \varepsilon - (1 - \alpha)f(x, \theta^*)] = 0$ we get $\lambda(\theta_0) = \lambda(\theta^*) + [(1 - \alpha)\mu]^2 > \lambda(\theta^*)$, such that $\lambda(\theta^*)$ can be no maximum.

Propositions 3.2 and 3.3 give necessary and sufficient conditions for a parameter vector $\theta^* \in \Theta^L$ to be a locally stable fixed point of differential equation (10). These conditions are thus necessary and sufficient for the probability that $\theta_t$ converges to an element of $\Theta^L$ for $t \to \infty$ to be nonzero. But this does not imply that $\theta_t$ will converge almost surely to an element of $\Theta^L$. This will be the case only if a $\theta^* \in \Theta^L$ is a locally stable stationary point of differential equation (10) and if additionally it is guaranteed that $\theta_t$ will almost surely stay within the domain of attraction of that stationary point infinitely often. Corresponding sufficient conditions for almost sure convergence of $\theta_t$ towards an identified $\theta^*$ may be formulated following Ljung (1977) by augmenting algorithm (9) with a so-called projection facility. However, in general nonlinear models it might be an extremely difficult task to formulate a projection facility ensuring almost sure convergence. Moreover, it is for the agents in the model to formulate a suitable projection facility. Thus, the use of such a projection facility in order to achieve almost sure convergence has been rightly criticized, e.g. by Grandmont and Laroque (1991).

4. A more general model

In the preceding section it was demonstrated that certain conditions have to be satisfied to ensure that a learning process based upon a neural network may converge towards – possibly approximate – rational expectations. It should be
noted, however, that approximate rational expectations are merely locally optimal, because they are local minimizers of the mean-squared prediction error. This means, that approximate rational expectations learned with the help of a misspecified neural network may differ substantially from correct rational expectations – the label ‘approximate’ says nothing about the actual distance between approximate and correct rational expectations. Thus, one should refrain from interpreting the above-derived convergence results too much in favor of neural networks.

In more general nonlinear models, a similar problem arises, but here this problem has a new quality with serious consequences. In order to describe this problem in more detail, let us now consider the more general reduced form

$$p_t = G(p_t^e, x_t) + \varepsilon_t$$  \hspace{1cm} (13)

and suppose that all other assumptions regarding the model are still valid. Thus, the only difference is that reduced form (13) is not additively separable in the expectation $p_t^e$ of the endogenous variable and a function of the exogenous variables $x_t$. Although this model is still a quite simple one, because the reduced form (13) does not contain lagged endogenous variables, the analysis of learning becomes now a more complicated one.\footnote{Since the function $G()$ is nonlinear, the presence of lagged endogenous variables would imply that the variables that enter the estimation procedure follow nonlinear stochastic processes. See Kuan and White (1994b).}

The first problem regards possible rational expectations equilibria. Given reduced form (13), it is by no means guaranteed, that there exists a unique function $\phi(x_t)$ which is a rational expectations function, i.e. that satisfies

$$\phi(x_t) = E[G(\phi(x_t), x_t) + \varepsilon_t | x_t]$$

almost surely.

Thus, it might be the case that there exist multiple equilibria. However, the possible existence of multiple rational expectations equilibria is not a specific feature of nonlinear models.

Another problem is that contrary to the model based on reduced form (1), the model based on (13) implies that the rest points of the differential equation associated with learning algorithm (9) will not coincide with local minima of the mean squared error $\lambda(\theta)$ (cf. Kuan and White, 1994b).

To see this, assume as before that $p^e = f(x, \theta)$. The mean-squared error $\lambda(\theta)$ is then given by

$$\lambda(\theta) = E\{G(f(x, \theta), x) + \varepsilon - f(x, \theta)\}^2$$

and the necessary condition for a minimum of $\lambda(\theta)$ results as

$$\nabla_\theta \lambda(\theta) = 2 E\{\nabla_\theta f(x, \theta)[G(f(x, \theta), x)$$

$$+ \varepsilon - f(x, \theta)][G_p(f(x, \theta), x) - 1]\} = 0,$$

\hspace{1cm} (14)
where $G_p$ denotes the partial derivative of $G(p,x)$ with respect to $p$. But, because $p_t - f(x_t, \theta_t) = G(f(x_t, \theta_t), x) + \varepsilon - f(x, \theta)$, the differential equation associated with the learning algorithm (9) is given by

$$
\dot{\theta} = E\{\nabla_p f(x, \theta)[G(f(x, \theta), x) + \varepsilon - f(x, \theta)]\}. \tag{15}
$$

A closer look at (14) and (15) reveals that the zeros of these two equations will in general not coincide, because the term $G_p(f(x, \theta), x)$ is neither deterministic nor independent from the other terms in (14). The zeros will coincide, if we have a $\theta^*$ such that $f(x, \theta^*) = \phi(x)$ almost surely, because we then also have $G(f(x, \theta^*), x) + \varepsilon - f(x, \theta^*) = 0$ almost surely. Thus, if there exists a $\theta^*$ implying $\lambda(\theta^*) = \sigma^2$, i.e. if the neural network is correctly specified and is able to give a perfect approximation of the unknown function $\phi(x)$, this $\theta^*$ is a rest point of differential equation (15). This is not true, however, for any $\theta$ that results only in a local minimum of $\lambda(\theta)$, because in such a case we have $G(f(x, \theta), x) + \varepsilon - f(x, \theta) \neq 0$ almost surely and a zero of (15) will not correspond to a zero of (14), i.e. not correspond to a local minimum of the mean-squared prediction error. Only if the model has the property that the expectation regarding the endogenous variable enters the reduced form in a linear way such that $G_p(f(x, \theta), x) - 1$ is a constant, any local minimum of $\lambda(\theta)$ will coincide with a rest point of (15).

Heinemann and Lange (1997) analyze neural network learning in a model with a reduced form given by (13). Assuming a correctly specified neural network, i.e. assuming that there exists a $\theta^*$ implying $f(x, \theta^*) = G(f(x, \theta^*), x)$ almost surely, they are able to derive a stability condition for the learning process. The respective necessary condition is $E[G_p(f(x, \theta^*), x) - 1] < 0$, which is a generalization of the condition $\alpha - 1 < 0$ derived from the simpler model considered above. But even if the neural network is correctly specified, there may exist other stable rest points of differential equation (15). As has been shown above, contrary to the simple model based on reduced form (1), these rest points will not imply that the mean-squared error $\lambda(\theta)$ is locally minimized. A related problem is that differential equation (15) is not a gradient system. Therefore, the analysis of the local dynamics of the differential equation associated with neural network learning becomes much more complicated if more general reduced forms are considered.

Thus, misspecification remains an issue even if neural networks are in principle able to approximate nearly any unknown function. A learning process that is based upon a misspecified neural network will in general lead towards expectations that are not even local minimizers of the mean-squared prediction error. There is no simple solution for this problem, because the number of hidden units that is necessary to achieve a perfect
approximation of the unknown rational expectations function cannot be determined a priori.\footnote{White (1990) considers neural network learning without feedback, where the number of hidden units is permitted to grow over time. The asymptotic analysis of the learning process is demanding even for the case without feedback and – at least to my knowledge – there are no results available concerning the feedback case.}

5. Conclusions

In this paper it was demonstrated, how neural networks can be used to model adaptive learning of – possibly rational – expectations. The analysis was based upon a simple Cobweb model, where the rational expectations function is a nonlinear function of exogenous variables. Given such a true model, the assumption of a correctly specified auxiliary model used by the agents to form the relevant expectations would imply an extraordinary a priori knowledge on the side of the agents. Clearly, the need of such a priori knowledge weakens any results concerning the potential learning of rational price expectations.

Because a neural network is able to approximate a wide class of functions to any desired degree of precision, it seems to be a suitable auxiliary model if the specific relation between exogenous variables and the resulting endogenous variable is unknown. It was shown that the possible convergence of a learning process based on a neural network towards rational expectations is governed by the properties of the underlying true model. In the simple case considered here, where the reduced form of the model is additively separable in the price expectation and a function of exogenous variables, the resulting stability condition is equal to the condition known from linear models and this stability condition holds irrespectively of the neural network to be correctly specified.

But even if the neural network used by the agents is correctly specified, i.e. if it contains enough hidden units to give a perfect approximation of the rational expectations function, learning does not need to converge to rational expectations: if the mean-squared error of the price expectations is taken as a measure of the success of the learning procedure, it can be shown that agents might merely learn to form approximate rational expectations, implying a (local) minimum of this mean-squared error. Apart from the local optimality of approximate rational expectations, nothing can be said regarding the actual distance between these expectations and rational expectations. This means that the performance of approximate rational expectations might be quite poor. Thus, the convergence results achieved above should be read with caution – learning of rational expectations is not a simple task even with the help of neural networks.

It must be emphasized that the model considered in Section 3 is a very simple static model with an additively separable reduced form containing independent
and identically distributed exogenous variables. This simple structure implies that the resulting associated differential equation takes a form which can be analyzed quite easily and enables the derivation of a definite stability condition for the learning algorithm.

As shown by Kuan and White (1994a), it is in principle possible to analyze more general models using the same methodology, but with increased analytical expense. Although it is true that a correctly specified neural network might be able to learn rational expectations even in more general models, the analysis of the reduced form considered in the previous section demonstrated that a misspecification of the neural network will have serious consequences: in general, the learning procedure yields parameter estimates implying expectations that are not even local minimizers of the mean-squared error.

Picking up a point already mentioned in the introduction, it becomes clear, that this objection can be raised against any procedure that is used in the learning process to approximate the unknown rational expectations function. Thus, the problem of misspecification is neither solved by neural networks nor exclusively related to them, because it concerns all parametric auxiliary models upon which a learning process may be based. The only way to overcome the problem of misspecification would be the use of nonparametric models as advocated by Chen and White (1994).

**Acknowledgements**

The author would like to thank Christiane Clemens, Mikko Packalen and anonymous referees of this journal for helpful comments and suggestions regarding an earlier version of this paper.

**References**


