



ELSEVIER

Information Sciences 144 (2002) 91–125

INFORMATION  
SCIENCES

AN INTERNATIONAL JOURNAL

www.elsevier.com/locate/ins

# Self-organizing feature maps predicting sea levels

Alfred Ultsch <sup>a,\*</sup>, Frank Röske <sup>b</sup>

<sup>a</sup> *Fachbereich Mathematik, Philipps-Universität Marburg, Hans-Meerwein-Straße, Lahnberge, 35032 Marburg, Germany*

<sup>b</sup> *Max-Planck-Institut für Meteorologie, Bundesstraße 55, 20146 Hamburg, Germany*

Received 18 May 2000; received in revised form 8 May 2001; accepted 15 August 2001

---

## Abstract

In this paper, a new method for predicting sea levels employing self-organizing feature maps is introduced. For that purpose the maps are transformed from an unsupervised learning procedure to a supervised one. Two concepts, originally developed to solve the problems of convergence of other network types, are proposed to be applied to Kohonen networks: a functional relationship between the number of neurons and the number of learning examples and a criterion to break off learning. The latter one can be shown to conform with the process of self-organization by using U-matrices for visualization of the learning procedure. The predictions made using these neural models are compared for accuracy with observations and with the prognoses prepared using six models: two hydrodynamic models, a statistical model, a nearest neighbor model, the persistence model, and the verbal forecasts that are broadcast and kept on record by the Sea Level Forecast Service of the Federal Maritime and Hydrography Agency (BSH) in Hamburg. Before training the maps, the meteorological and oceanographic situation has to be condensed as well as possible, and the weight and learning vectors have to be made as small as possible. The self-organizing feature maps predict sea levels better than all six models of comparison. © 2002 Elsevier Science Inc. All rights reserved.

---

\* Corresponding author.

E-mail addresses: [ultsch@informatik.uni-marburg.de](mailto:ultsch@informatik.uni-marburg.de) (A. Ultsch), [roeske@dkrz.de](mailto:roeske@dkrz.de) (F. Röske).

## 1. Introduction

Since decades sea level forecast is one of the official tasks of the German Hydrographic Institute in Hamburg, now since 1990 the Federal Maritime and Hydrographic Agency (Bundesamt für Seeschifffahrt und Hydrographie: BSH). It issues sea level predictions and warnings in case of storm surges or abnormally low water for the German coastal areas of the North and Baltic Sea, the adjoining rivers and ports. Every 6 hours a new sea level forecast is made for the coming 4 or 5 high and low tides at the North Sea coast and the estuaries of the rivers Ems, Jade, Weser, and Elbe. Vessel traffic centers and port authorities use these information for logistic purposes like control of traffic flow and optimal loading. State authorities and private people get all kinds of verbal and written information concerning sea level questions.

The sea level forecasts are of vital interest for people living in the coastal areas. The up and down of the water influences shipping, fishery, and particularly the traffic between the Frisian Islands and the main land over the tidal flats falling dry at low tide. To protect against abnormal high tides, a system of dikes, walls, barriers, and gates, which are closed in emergency, has been built. For long-term security the height of the dikes has to be raised. For an optimal resistance their angle of slope may not be too steep, and therefore an increase of height of the dike means also a considerable broadening of its base, for which limited space has to be made available. In front of this background reliable sea level prediction is increasingly important.

Furtheron, it is vital for today's shipping operation and port economy. Hamburg is one of the biggest ports of the world. To meet the increasing transport volume, larger and larger container vessels with correspondingly deep draft are calling at the German ports. It would be necessary, to deepen the whole fairway of the river Elbe from sea to Hamburg, to make the passage of these vessels independent of the height and time of tide. For ecological reasons only part of the fairway is deepened, and deep draft vessels sail the river according to tidal windows. To determine them in time, sea level predictions as accurate and reliable as possible along the fairway have to be made by the BSH.

The tidal predictions for the German North Sea ports, calculated by means of a quasi-harmonical method are precise within the limits of about 10 minutes and 10 cm. They are the basis of all sea level forecasts for the North Sea coast and make it possible, to determine, as long as wished in advance, the tide induced deviations from the mean high and mean low tides. The deviations caused by the wind – they are called anomalies in this paper – can be predicted good enough only as long in advance as reliable wind forecasts for the Southeastern part of the North Sea, i.e. the German Bight are available. Wind forecasts are provided by the Marine Weather Service (Seewetteramt: SWA) in Hamburg, which is part of the German Weather Service (Deutscher Wetterdienst: DWD) in Offenbach.

In the process of sea level forecast personal experience plays an important role. There is the desire, to use additional methods, in order to reduce possible inconsistencies of the prediction. New techniques shall work according to other principles than those already in use, to ensure maximal independence. Neural networks comply with this requirement. They have a certain relation with statistical methods, which they have to cope with, but because of the learning procedure neural networks are sufficiently independent of the models used by the BSH so far. Due to the similarity of artificial and biological neural networks the question can be put, if and how far the experience, which is in the brain of the experts, can be placed on an objective basis, and in this way used by inexperienced people. Till now successful applications of artificial neural networks have been developed and could improve predictions on different fields. Therefore, their use for sea level forecasts seems to be very promising.

The aim is, to develop an operational tool on the basis of neural networks, that is appropriate for the prediction of standard situations. The accuracy of the forecast in case of storm surges still depends on the expert's experience.

The models used by the BSH so far – empirical, statistical and numerical hydrodynamic methods – seek to explicitly model the underlying physical processes. The first of these employs a bulk formula only roughly approximating the processes, while the second one takes a hydrodynamic approach providing more details of the processes. The neural networks model implicitly, i.e. the way they work cannot be described by explicit formulae. In contrast to statistical methods, explicit formulae are not presupposed for neural networks. Is the implicit modelling of physical processes employed in the neural networks superior to the explicit modelling employed in the hydrodynamic and statistical methods? Is it possible to make predictions of natural processes using neural networks, when explicit knowledge of their non-linear interrelationships is not used, whereby their predictions are superior to the predictions of the other methods, which use this knowledge? Attempts will be made to answer these questions with regard to the use of the networks for predicting sea level.

The paper is organized as follows. In Section 2 the application will be described. Some definitions and a general description of the neural models will be given followed by the description of the data (see Table 1). In Section 3 the self-organizing feature maps will be presented and how they are transformed into supervised learning procedures. After that, the U-matrices used for visualizing the learning process will be described. Subsequently, it will be depicted how the data have to be prepared to become suitable for the training procedure. Some attention will be given to the wind data. Afterwards the learning procedure will be described including two concepts which have originally been developed for other network types and are proposed to be applied to the self-organizing feature maps. This is followed by a description of the experiments performed and with a specification of the neural models used for prediction. Section 3 is completed by a description of those models, the neural models will be

Table 1  
General description of the neural models

Abbr.	Model name	Principle number of variables	Temporal scanning of model predict.	Derived from model pred. using tidal predictions	Length of prognostic time window	Prognostic quantity
AR	Autoregression applied iteratively	Univariate	Hourly		1	Anomalies
CL	Classification	Univariate	Hourly	Times of high/low tides	>1	Anomalies, additional predictions
MR	Multiregression	Multivariate	Hourly, times of high/low tides		1	Anomalies
RW	Regression-plus-window	Multivariate	Hourly	Times of high/low tides	>1	Anomalies
MW	Multiwindow	Multivariate	Hourly	Times of high/low tides	>1	Anomalies
DC	Double classification	Univariate (vector)	Hourly	Times of high/low tides	>1	Additional predictions

compared with. In Section 4 the results are presented using the neural models. They will be compared with observations and with the models of comparison. In Section 5 the reasons will be given why the self-organizing feature maps have been chosen for this application and why those two concepts have been applied. Afterwards the results of the neural models will be discussed. The paper closes with some conclusions in Section 6.

## 2. Description of application

### 2.1. Definitions

Some terms used in the following are introduced and explained. The data set employed will later be divided into three subsets [31]: the training and validation sets consist of data determined during successive periods, with those of the validation set being recorded immediately after those of the training set. The prediction set consists of values predicted for the same period as that of the validation set. The prediction error is distinguished from the representation error, which is based on data actually recorded. For example, this would be the error if the statistical method called “Gesamtansatz” (see Section 3.7) were fed

with the meteorological data actually observed rather than with the meteorological forecasts. In this model, anomalies are a function of meteorological factors. Thus, to forecast anomalies, meteorological forecasts, referred to as additional predictions, are required. If the representation error is calculated from the training data set, it is called training error; if it is calculated from the validation data set, it is called validation error.

Anomalies are hindcast by neural networks and all other models used for comparison. This is equivalent to predicting them after they have already occurred. These hindcasts, however, are distinct from the forecasts for the still unknown future, which are prepared as the daily duty of the Sea Level Forecast Service of the BSH. In the case of the forecast, it is obvious that the future observations cannot be used for training the network. It is emphasized, that when making the hindcasts, the observations already made are also not used for training.

The factors to be predicted, such as sea level anomalies, are called prognostic quantities. Factors that influence the prognostic quantities, including wind and air pressure, are called indicating quantities. Time series of prognostic quantities are called prognostic time series; time series of indicating quantities are called indicating time series. Applying time windows to discrete hourly time series yields prognostic and indicating time windows, which consist of a number of time points separated by 1 hour. This number defines the lengths of the time windows. Multivariate methods, such as the “Gesamtansatz” or similar multiregression statistical methods [11], predict for one time point. Thus, the length of the prognostic time window equals one. They require additional predictions. Univariate methods, such as autoregressive linear and non-linear statistics [24], do not require additional predictions. Consequently, their prediction and representation errors are identical. The time windows of the univariate methods are divided into an indicating time window and a prognostic time window. If the length of the prognostic time window equals one, then the univariate method is referred to as autoregressive. If its length is greater than one, the method is referred to as classification.

Temporal patterns are discrete temporal structures that form the bases of statistical methods. In addition to multiregression and univariate temporal patterns, other combinations of these may be employed. Two classifying temporal patterns superimposed on one another yield a double classifying temporal pattern. Superimposing a multiregression temporal pattern on a classifying temporal pattern, as Boogaard proposed (priv. comm.), yields a regression-plus-window temporal pattern. If, however, a multiregression is combined with a classifying temporal pattern to create a kind of dyadic product, a multiwindow temporal pattern is produced, in which its number of time points is approximately equal to the product of the numbers of both basis temporal patterns. The set of all indicating time windows is called an indicating temporal pattern, while the set of all prognostic time windows is called a

Table 2

Quantities	Trends	Exponents of the non-linear transformation	z-Transformation
Sea levels	0.5 cm/y		Yes
Anomalies	0.38 cm/y	0.4	Yes
Wind speed	0.15 kn/y	0.5	Yes
Wind direction			Yes
Static air pressure	0.15 hPa/y	1.7	Yes
Three-hourly changes of pressure		1.3	Yes
Air temperature	0.3 K/y		Yes
Difference air minus water temperature	-0.1 K/y	1.6	Yes

prognostic temporal pattern. All of these temporal patterns can be combined with neural networks. These combinations are called neural models. Neural autoregression models can be applied iteratively several times for predictions at more than one time point [29].

An overview of the neural models applied to the prediction of anomalies and to the predictions of other quantities which are used as additional predictions for the anomalies is given in Table 2. The multiregression neural model is the only one, which can be applied both for hourly anomalies and for anomalies at times of high/low tides.

## 2.2. Data description

To prepare time series of hourly anomalies, time series of hourly sea level data recorded from the tidal gauge near Cuxhaven by the BSH from 1985 to 1993 and hourly tidal predictions for the same location and for the same period were employed. To prepare time series of anomalies at times of high and low tides, the anomalies have been calculated from the hourly time series by interpolation.

Meteorological data were provided by the SWA. Hourly observations of the wind, static air pressure, three-hourly change in the air pressure and air temperature from 1985 to 1993, as well as the water temperature from 1988 to 1993 at Helgoland were used. In addition, the winds observed at light vessels “TW Ems” and “Deutsche Bucht” were also used. The winds of all three sites were vectorially averaged according to the rules proposed by Annutsch [21]. To prepare time series of meteorological data at times of high and low tides, the data have been derived from the hourly time series without interpolation, assuming that the time lag can vary between three and four hours.

There are geophysical arguments to include air pressure observations at weather stations from North-western Europe [21]. Hourly time series at

stations on the British Isles and in the Netherlands were used. Three-hourly time series at Scandinavian stations have been interpolated to hourly ones. Time series of values at times of high and low tides were derived by analogy with the data at Helgoland.

Using air pressure data is also an attempt to take external surges induced by air pressure into account. Another way to account for these surges is to include additional oceanographic data. There are geophysical arguments to use anomalies at Wick in Scotland and at West Terschelling in the Netherlands, as [21] suggests. From observed sea levels at these locations corresponding tidal predictions have been subtracted to yield anomalies.

### **3. Model description**

The idea behind neural networks is that if deterministic rules dictate a dynamic system, then, even if the behavior is chaotic, the future may to some extent be predictable from the behavior of the past states of the system that are similar to those of the present [8]. The experience in prediction gained by the staff of the BSH Sea Level Forecast Service indicates that learning from the past is possible. But this is only true for past events or situations which partially repeat. Situations which repeat themselves seldom or not at all, such as storm surges, are difficult to learn.

From the variety of neural network types the self-organizing feature map or also called Kohonen network after the Finnish physicist Teuvo Kohonen [14] has been chosen to learn to predict sea levels or anomalies. His neural network model can be motivated neuro-physiologically [20].

#### *3.1. Kohonen networks*

Kohonen networks or self-organizing feature maps are networks, which consist only of two layers, an input and an output layer. The output layer of Kohonen networks can be two-dimensional. The most important difference is that the neurons of the output layer are connected with each other. The arrangement of the output neurons plays an important role. Sensorial input signals, which are presented to the input layer, cause an excitation of the output neurons, which is restricted to a zone of limited extent somewhere in the layer. This excitation behavior comes from the back coupling of the neurons. It is essential to know how the interconnections of the neurons have to be organized in order to optimize the spatial distribution of their excitation behavior over the layer. Neurons with similar tasks can communicate over very short pathways. The optimization produces topographic maps of the input signals, in which the most important relationships of similarity between the input signals are converted into relationships among the neuron positions. This corresponds

to an abstracting capability which suppresses unimportant details and maps the most important features along the map dimension. Summarized, one can say that Kohonen networks seek to transpose the similarity of sensorial input signals to the neighborhood of neuron positions.

The input signals  $v_l$ ,  $l = 1, \dots, L$ , with  $L$  being the number of signals are transmitted by  $d$  entry fibers through a two-dimensional neuron layer,  $A$ . The signals excite or inhibit the neurons across synaptic connections. Neurons are identified by their location,  $\mathbf{r} = (x, y) \in A$ , which can be chosen as the positions on a discrete grid. Each neuron,  $\mathbf{r}$ , forms a weighted sum,  $\sum_l w_{r,l} v_l$  in its tree of dendrites, whereby  $v_l$  and  $w_{r,l}$  are the strengths of the synapses between axon  $l$  and neuron  $\mathbf{r}$ . For an excited neuron,  $w_{r,l}$  is positive, and for an inhibited one, it is negative.

The brain not only has to interpret sensorial input signals, it has also to control the muscular system. Therefore, the sensorial maps have to be expanded to motorial maps which are neural models capable of controlling, e.g. robots. These maps include an additional output weight at each neuron,  $\mathbf{r}$ . In addition to the input signals  $\mathbf{v}$ , the correct controlling action,  $u$ , must also be available. This situation corresponds to supervised learning. Just as the sensorial map is learned adaptively, the assignments of output values to grid points in the motorial map must be variable, as well. Renaming the weights  $w_r$  with  $w_r^{(\text{in})}$  and introducing the scalar weight  $w_r^{(\text{out})}$  according to  $u$ , then Kohonen's model of the motorial map can be written as follows:

0. Initialization: Random choice of  $(w_r^{(\text{in})}, w_r^{(\text{out})})$ .

1. Choice of stimulus: Choice of  $(\mathbf{v}, u)$ .

2. Response: The corresponding excitation center,  $\mathbf{r}'$ , referred to as bestmatch neuron, is only determined from  $\mathbf{v}$  according to the conditions defined by

$$\|\mathbf{v} - w_{\mathbf{r}'}^{(\text{in})}\| \leq \|\mathbf{v} - w_{\mathbf{r}}^{(\text{out})}\| \quad \forall \mathbf{r} \in A.$$

3. Step of adaptation: Change of synapse strengths,  $w_r^{(\text{in})}$  applying

$$w_r^{(\text{in,new})} = w_r^{(\text{in,old})} + \varepsilon \cdot h_{r\mathbf{r}'} \cdot (\mathbf{v} - w_r^{(\text{in,old})}).$$

4. Step of adaptation: Change of synapse strengths,  $w_r^{(\text{out})}$  applying

$$w_r^{(\text{in,new})} = w_r^{(\text{in,new})} + \varepsilon' \cdot h'_{r\mathbf{r}'} \cdot (u - w_r^{(\text{in,new})})$$

and continuing from step 1.

$\varepsilon$  is the size of a single adaptation step,  $0 < \varepsilon < 1$ , choosing high values initially so that the system will quickly and roughly learn the correct synapse strengths and then seeking values near zero so that the maps will asymptotically reach an equilibrium. The neighborhood function  $h_{r\mathbf{r}'}$  is approximated qualitatively by a unimodal function that reaches its maximum at  $\mathbf{r}'$  and depends only on distance  $\mathbf{r} - \mathbf{r}'$  and a radius,  $\sigma$ . Like  $\varepsilon$ ,  $\sigma$  also has to start at high values, allowing the coarse structure of the map to form. Then  $\sigma$  has to



decrease to permit the local fine structure to be formed.  $\|\mathbf{x}\|$  is the Euclidean vector norm,  $(\sum_l x_l^2)^{1/2}$  as [20] proposed.

Compared with sensorial maps, the algorithms of which are not shown here, step 4 is added to change to output values,  $w_r^{(\text{out})}$ , on the grid. This change takes place with complete analogy to the learning step for the synapse strength,  $w_r^{(\text{in})}$  but with its own learning step width,  $\varepsilon'$ , and neighborhood function,  $h'_{rr'}$ , with the corresponding radius,  $\sigma'$ . Thus, an adaptive table for mapping this system state on the controlling action is produced. The assignments of table entries to input values is not firmly predetermined but is rather developed during the learning phase when the table is filled. Table entries,  $(w_r^{(\text{in})}, w_r^{(\text{out})})^T$ , are distributed according to the density of the required controlling actions in the space of mapping  $w_r^{(\text{in})}$  on  $w_r^{(\text{out})}$ . Regions in this space, which often require controlling actions, receive more table entries, producing a higher resolution of the input–output relationships. Table entries, which are seldom or never used, were not assigned. Therefore, the memory available is used very economically.

Kohonen networks can be applied for prediction in two ways, which lead to the same result. In the first of these, the system state,  $\mathbf{v}$ , can be identified with the indicating temporal pattern, while the controlling action is identified with the prognostic temporal pattern. However, along with the development of motorial maps, Ultsch proposed making the sensorial map useful for predictions, in which not only the weight vectors but also the corresponding learning vectors are divided among the past state and prognostic descriptions. Only the past state description is used for determining the best match neuron by analogy with the motorial maps. Ultsch and Halmans [29] confirmed this concept and applied it to auto- and multiregressions and for classifying temporal patterns. The methods introduced here incorporate these ideas. Past state descriptions are identified with indicating temporal patterns, and prognostic description, with prognostic temporal patterns (Fig. 1).

Consequently, modified Kohonen networks can be regarded as adaptive tables to indicate and prognose temporal patterns. Nearest neighbor methods can also be regarded as tables, but they are not adaptive. The assignments of table entries to input values are strictly predetermined. This table structure permits a comparison of the Kohonen networks with the nearest neighbor procedure, as proposed by Ultsch. It is possible to evaluate how well the table entries have been distributed in the space of the temporal pattern and how good the kind of interpolation is, that is performed by the Kohonen network in the prognostic temporal pattern.

Kohonen networks are characterized by the idea of preservation of topology, which can be inspected visually. For two-dimensional Kohonen networks, a method has been developed to visualize the learning phase in three dimensions. Visualization is achieved using the unified distance matrix methods, referred to as U-matrix methods, developed by Ultsch [27]. Ultsch's original

2-dimensional neuron layer (output layer)

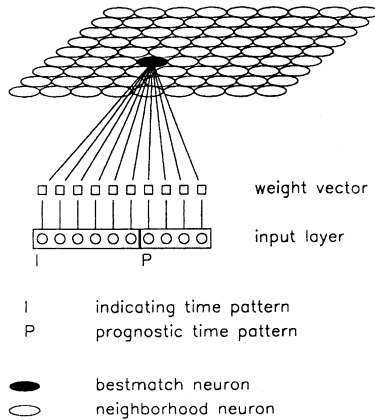


Fig. 1. Self-organizing feature map used for making predictions.

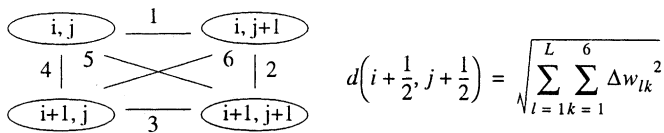


Fig. 2. Illustration of the modified U-matrix method. All six possible differences of weights of four neighboring neurons are used. Their root mean square is assigned to the center of the neurons.

method has been modified to improve its clarity and uniformity by the following procedures:

Position vector  $r = (x, y) \in N^2$  is renamed by numbering  $i, j \in 1, \dots, n$ , assuming a quadratic grid of neurons with  $n$  neurons in each dimension. This grid can be interpreted as a quadratic matrix of weight vectors with  $L$  denoting the length of the weight vectors. On the basis of four neighboring neurons distance  $d$  can be calculated as illustrated in Fig. 2.

This yields the matrix,  $U \in R^{(n-1) \times (n-1)}$ , having each dimension reduced to one less than those of the matrix of weight vectors.

### 3.2. Data preparation

Before the Kohonen networks can be trained, special demands have to be fulfilled. These networks treat input signals as independent random variables. According to the central limit theorem, such variables must be normally distributed. Data not so distributed can be converted to data following a normal distribution [23].

This distribution is most important with Kohonen's motorial map, in which metric Euclidean distances are used. This metric value treats differences in two vectors as quadratic. Therefore, Kohonen networks react very sensitively to extreme values in the time series. Therefore, extreme values must be evaluated to determine whether they are real, such as from storm tides, or artificial. If they have been recognized as errors, they must be removed from the time series, leaving gaps. Gaps also arise due to missing observations.

These gaps have partially been filled in to obtain a set of temporal pattern vectors as complete as possible. Neural networks can complete data sets [28]. However, this capability is already utilized when the networks are applied for making predictions. Incomplete learning vectors consisting of only an indicating temporal pattern are completed using network predictions, regarded as a prognostic temporal pattern. Therefore, gaps were filled in by other methods.

After this procedure has been completed, a trend analysis of the filled time series must be performed. There are primarily two reasons for eliminating large-scale trends: 1. The network will assume that a trend is important information and will attempt to learn to use that information for prediction. That would be fine if it could do so without compromising itself. However, by eliminating trends the network is freed to concentrate on finer details. 2. Neural networks are inherently non-linear. If large variation is superimposed on more important subtle information, the learning process will tend to scale down overall information, thus depriving the smaller variation information of the benefits of non-linearity [16]. The coefficients of the linear trends are shown in Table 3. Tidal predictions which are subtracted from sea levels prior to the trend elimination can also be regarded as trends.

After the trends have been eliminated, the following procedure has to be completed. First, the time series must be pre-processed. Afterwards, the time series are organized as a set of learning vectors. Then, a subset of learning vectors is selected from this set, and is used for training the Kohonen networks. Finally, the networks are used for making predictions, which have also to be processed. This step is called post-processing.

For pre-processing, several statistical parameters have to be derived. These are also used for the post-processing. To avoid failures during this step, the parameters must be representative of each component of the learning vectors. Therefore, they are calculated not from the subset but from the set of temporal pattern vectors, i.e., consequently, from the entire time series.

Pre-processing is required, because the vector components have to be made comparable. To achieve this, correlations must be eliminated, distributions assimilated, and the values have to be mapped into a comparable range [27]. If two of the quantities that influence the anomalies locally were correlated, only one of them would influence the anomalies. Therefore, for neural multiregression models, using the same input quantities as the "Gesamtansatz", the first prerequisite is fulfilled.

Table 3  
Specified neural models

Prognostic quantity	Neural models	Length of prognostic time window	Length of indicating time window(s) ( $N$ to be varied)	Time lag in hours relative to the anomalies at Cuxhaven
Anomalies at Cuxhaven	AR	1	1	
Anomalies at Cuxhaven	CL	18	$N$	
Anomalies at Cuxhaven	MR	1	1	
Anomalies at Cuxhaven	RW	18	1	
Anomalies at Cuxhaven	MW	18	18 (21 for air pressure)	
Wind speed, wind direction	CL	15	$N$	3
Wind vector	DC	$2 \times 15$	$2 \times N$	$2 \times 3$
Air pressure	CL	15 or 18	$N$	3 or 0
Change of air pressure	CL	18	$N$	0
Air temperature	CL	15	$N$	3
Difference of air minus water temperature	CL	15	$N$	3
Anomalies at West Terschelling	CL	14	$N$	4
Anomalies at Wick	Prediction not necessary due to the time lag			15

The second and third prerequisites are related to the motorial map. The sensitivity of the Euclidean distances to extreme values strongly affects the pre-processing. Data distributions can generally be regarded as skewed normal distributions. If distributions of the input quantities differ in skewness, the distributions may not be comparable, and the learning process might be disturbed by undesirable extreme values. Therefore, skewness of the data distributions must be eliminated.

This can be done by non-linear transformation of the data involving raising every datum to an exponential power. Exponents where  $p < 1$  reduce right skewness, and those where  $p > 1$  reduce left skewness. The value  $p = 0$  is replaced by a logarithm [23]. The optimal  $p$  per time series was found by minimizing the sum of the differences between the data organized as quantiles and the quantiles of the theoretical normal distribution [10]. In order to permit the data to be raised to exponential powers, the minima for the time series had to be found and subtracted from each datum.

The third prerequisite meets the requirement that all the quantities in the temporal pattern must be similarly weighted in the Euclidean distances. None of the values of the quantities may be of magnitudes greater than the others. Therefore, the time series that have been transformed non-linearly but still have to be mapped into a range of values that can be compared. In addition, their ranges must be comparable with the range of random numbers needed for initializing the networks. It has been shown that the best results can be achieved using a normalization with the mean and standard deviation, the so-called  $z$ -transformation [9,27]. In order to be consistent with the non-linear transformation, the momentums based on the rank, i.e. median,  $\mu$ , and the empirical standard deviation,  $\sigma$ , were used instead [10]. Thus, the non-linear transformation and the  $z$ -transformation can be combined according to the following formula, that is called the pre-processing ( $i = 1, \dots, n$ ):

$$x_i^{\text{new}} = \begin{cases} \frac{(x_i^{\text{old}} - c)^p - \mu}{\sigma} & \text{for } p \neq 0, \\ \frac{(\ln(x_i^{\text{old}} - c) - \mu)}{\sigma} & \text{for } p = 0, \end{cases} \quad i = 1, \dots, n,$$

where  $c = \min_j(x_j^{\text{old}}) - 1$  ( $j = 1, \dots, n$ ). The  $z$ -transformation standardizes the first and second momentum, i.e. the median of the transformed time series equals zero and the empirical standard deviation equals zero. The non-linear transformation standardizes the third momentum of distributions. Then, the pre-processing can be extended to the global pre-processing, i.e. including more than one geographical point, as proposed by Latif (priv. comm.). Parameters calculated locally cannot be averaged globally because of the non-linearity of the pre-processing. Therefore, the following method was chosen. Parameters  $p$ ,  $c$ ,  $\mu$ , and  $\sigma$  are not related to the temporal order of the data. They can therefore be calculated from a data set created by joining all time series of a predetermined factor, e.g. air pressure, at all weather stations chosen. Parameters determined in this way can then be applied locally to take local effects into consideration.

During the learning process, the magnitude of the randomly initialized weights in the Kohonen networks does not change significantly. In order to make the predictions from the network relevant to the actual conditions, they must be processed again. This post-processing includes all steps used in pre-processing but in a reversed order, demonstrated by the following formula:

$$y_k^{\text{new}} = \begin{cases} (y_k^{\text{old}} \cdot \sigma + \mu)^{1/p} + c & \text{for } p \neq 0, \\ e^{y_k^{\text{old}} \cdot \sigma + \mu} + c & \text{for } p = 0, \end{cases} \quad k \geq 1,$$

where  $k$  is dependent upon the temporal pattern chosen,  $k = 1$  in the temporal patterns of the auto- and multiregression, and  $k > 1$  for all other temporal patterns,  $k$  will be defined later in the description of experiments.

In Table 2 the pre-processing together with the trend analysis are roughly summarized. Sea levels are included, because they have also been used for training Kohonen networks. Air pressure observations at weather stations from North-western Europe were pre-processed by analogy with the air pressure at Helgoland and anomalies at Wick and West Terschelling were pre-processed by analogy with the anomalies at Cuxhaven, but these data are not represented in Table 3.

### 3.3. Wind treatment

Multivariate neural network models require additional predicted inputs, such as meteorological factors. Kohonen networks can also be used to predict these factors. However, they are predicted only by means of their own past, i.e. univariately. Predicting the wind direction entails a problem related to the range of values. Wind directions are recorded on a continuous scale with  $360^\circ$  being identical to  $0^\circ$ . Using wind directions in the training of Kohonen networks, this range is adapted to a linear scale open at both ends. Therefore, the data show jumps when the wind changes from northwest to northeast or vice versa. Generally, when these jumps occur can be learned by the Kohonen networks. However, those that occur in the network predictions are no longer at the same time as those actually observed. Therefore, the prediction errors are unnecessarily high.

These jumps can be removed by transposing the wind data, which are on polar coordinates, on Cartesian coordinates. Physically, both coordinate systems are equivalent, i.e. the winds in these systems are identical. But the distributions of the wind vector components change, if the coordinate system is changed. The Cartesian wind components are normal distributed. The distribution of the wind speed is similar to a Poisson- or Maxwell-distribution. The distribution of wind direction is in the range of  $0^\circ$  and  $360^\circ$ . Therefore, this distribution does not include undesirable extreme values and is thus better suited for training the Kohonen networks than the Cartesian wind components. Therefore, the wind data have not been transposed on Cartesian coordinates.

Jumps have been removed in another way instead. In a learning vector, a downward jump is defined as follows: if  $d_i - d_{i+1} > 180^\circ$ . The following is a definition of an upward jump: if  $d_i - d_{i+1} < -180^\circ$ . All wind directions between a downward and the next upward jump were increased by  $360^\circ$ . In only one downward jump were taken, the value of the direction was increased until to the end or the beginning of the vector.

### 3.4. Data selection

There are reasons that not the whole set of temporal pattern vectors is used for training the Kohonen networks. These reasons will be described later. Only

a subset, the so-called learning vectors, will be used instead. To select such a subset, the time series must be organized as this set of temporal pattern vectors, which are also called  $m$ -histories [1]. First, one of the temporal patterns is chosen, which are defined in the introduction. Then, this pattern is moved from the first recorded value to the next until to the last through all time series needed by that pattern. In this way, a set of vectors is produced that can overlap if the temporal pattern chosen has at least one time window with more than one time point.

From the set of pre-processed temporal pattern vectors, a subset of learning vectors can be selected by the help of two methods. The first is a method of classical cluster analysis [4,26], which has been slightly modified. For predicting standard situations, vectors must be selected which are typical or representative of the entire vector set. An analogy can be seen between the set of temporal pattern vectors of length  $L$  and a vector space of dimension  $L$ . Searching for data representative of this space can be regarded as a multicenter problem as Bandelt proposed.  $L$ -dimensional spheres containing all vectors are sought in a way that minimizes the sphere radius. Vectors at the midpoints of the spheres are the representative data.

In order to get an idea of the performance of the algorithm, the number of vectors should be optional. If an increase of the number of learning vectors was accompanied by a decrease in the prediction errors of the Kohonen networks, there would be a possibility that the sea level forecast using neural networks could be improved by using computers of greater capacity.

For this purpose, disjunct methods that presuppose a known number of clusters are most appropriate. Although these clusters are artificial, they can be useful if they have certain features. The minimal distance (MD) method was chosen, in which every element is at the shortest distance from the center of its class, thereby fulfilling the minimal variance criterion, whereby the data are clustered as a minimal distance partition [26]. In that algorithm, the center vectors are determined by averaging the vectors in each cluster. In order to find representative vectors those with the shortest distance from the average center vectors were sought. Each component of these representative vectors conform roughly to the normal distribution.

Before the minimal distance method was used for data selection, an own method was developed. Some details of this method were also suggested by Kleine (priv. comm.). However, it became apparent that this method selected extreme vectors instead of typical or representative ones. Therefore, the vectors were selected mainly by the minimal distance method. However, it also became apparent that the new method can support the minimal distance method during the selection procedure. The subset of learning vectors can be composed by some vectors selected by the minimal distance method and by some selected by the new method. For a certain temporal pattern, it was possible to reduce the prediction errors of the Kohonen networks that were trained by this subset.

The new method will be described in detail now. The number of vectors selected should also be optional in this case. From the set of temporal pattern vectors, pairs can be chosen, and their Euclidean distances calculated to produce a matrix of these distances. Generally, the set of temporal pattern vectors will be reduced by reducing this matrix. Vectors are eliminated successively until the number desired remains. The matrix of a large set of vectors requires much time for computation of the elimination process. To avoid this, many small matrices of the same size are established to approximate this big one matrix. The size of the small matrices is optional for flexible adjustment to the computer capacity and corresponds to the number of vectors used for one matrix.

At the start, all temporal pattern vectors are arranged in a circle. According to the number chosen some vectors are selected, which lie one after another in the circle. A group is defined including these vectors. The matrix of this group is calculated with only one vector eliminated. The next group is defined including the remaining vectors and the succeeding one. Thus, the group progresses through the circle, in which number of vectors decreases by one as it passes each other group. The number of vectors chosen per group is called the group size.

With  $N$  as the total number of vectors for any temporal pattern,  $n$  is the number of vectors desired where  $n \ll N$  and vectors  $\mathbf{x}_i$ ,  $i = 1, \dots, N$ , of length  $L$ . Choose group size  $G$  with  $G \leq N$ . Set the circulation parameter to  $t = N$  and  $p = 0$ . The process continues with the following steps:

1. Choose a group  $\mathbf{x}'_j$  from  $\mathbf{x}_i$ , with  $\mathbf{x}'_j = \mathbf{x}_{p+j}$ ,  $i = 1, \dots, t$ ,  $j = 1, \dots, G$ .
- 2a. Calculate the distance matrix  $d_{jk} = \|\mathbf{x}_j - \mathbf{x}_k\|$ , where  $d_{jk} = d_{kj}$ ,  $d_{jj} = 0$  and

$$\|\mathbf{x}_j - \mathbf{x}_k\| = \sqrt{\sum_{l=1}^L (x_{lj} - x_{lk})^2}, \quad j, k = 1, \dots, G.$$

- 2b. Determine  $j'$  and  $k'$  from

$$d_{j'k'} = \min_{j,k} d_{jk}, \quad j, k = 1, \dots, G.$$

- 2c. Determine  $i'$  from

$$\sum_m d_{i'm} = \min \left( \sum_m d_{j'm}, \sum_m d_{k'm} \right), \quad m = 1, \dots, G.$$

3. Eliminate  $\mathbf{x}_{i'}$  from  $\mathbf{x}_i$  by renumbering:  $\mathbf{x}_{i+1} \rightarrow \mathbf{x}_i$

$$\forall i = \begin{cases} i' + p - t \cdots t - 1 & \text{if } (p > t - G) \text{ and } (i' \geq t - p) \\ i' + p \cdots t - 1 & \text{else} \end{cases}$$



4. Control of the circulation:

- if  $p < t - G$ , increase  $p$  by  $(G - 1)$
- if  $p = t - G$ , set  $p = 0$
- if  $p > t - G$ , subtract  $(t - G)$  from  $p$   
 and if then  $i' < p$ , subtract 1 from  $p$ .

Subtract 1 from  $t$ .

If  $t = n$ , break off, else go to step 1.

Learning vectors  $x_i, i = 1, \dots, n$ , are obtained with their mean Euclidean distance set to be

$$\bar{d} = \frac{2}{n(n-1)} \sum_{i,j(j>i)}^n d_{ij}.$$

The equation has been set to a maximum value dependent upon  $G$ . The greater the value of  $G$  chosen, the larger is  $d$ , and the better is the approximation. If  $G = N$ , the largest group would be obtained, and the approximation would be perfect. This method is called method of circular group reduction (CGR).

Both methods, MD and CGR, are related to the Kohonen algorithm through their quadratic distance metrics. Vectors selected by MD or by both MD and CGR are used for training the Kohonen networks. In order to guarantee random behavior and to avoid the possibility that the order of vectors would be temporal, the vectors were presented randomly for each learning epoch (see the following section) and with different permutations from epoch to epoch.

3.5. Learning procedure

The learning procedures, during which the learning vectors are presented to the input layer, can be divided into learning epochs. During one learning epoch, all learning vectors are used. Thus, during a learning phase, which consists of several learning epochs, the vectors are repeatedly presented to the input layer [9].

Several parameters have to be determined before the networks can be trained. A suitable decrease in the width of the learning step,  $\varepsilon$ , and the radius,  $\sigma$ , in the neighborhood function  $h_{rr'}$  is important. Following the method of Ritter et al. [20], an exponential law of decrease is used:

$$\varepsilon(i) = \varepsilon_s \left( \frac{\varepsilon_e}{\varepsilon_s} \right) \left[ \left( \frac{i}{n_{LV} n_{L,max}} \right)^2 \right]$$

and

$$\sigma(i) = \sigma_s \left( \frac{\sigma_e}{\sigma_s} \right) \left( \frac{i}{n_{LV} n_{L,max}} \right) \quad \text{with } i = 1, \dots, n_{LV} n_{L,max},$$

where  $n_{LV}$  is the number of learning vectors, and  $n_{L,max}$ , the maximum number of learning epochs,  $\varepsilon_s = 1$ ,  $\varepsilon_e = 0.1$ ,  $\sigma_s = n_N$ , and  $\sigma_e = 1$ . The term  $n_N$  represents the total number of neurons in a quadratic grid or the total number of weight vectors arranged in a quadratic matrix. Therefore,  $(n_N)^{1/2}$  is the number of rows or columns in this matrix. It is suggested that the  $\varepsilon$  be permitted to decrease exponentially with the exponent being quadratic instead of linear to achieve better convergence. The form of  $h_{rr'}$  is not important, as Ultsch reported. Thus, a form can be chosen that does not require much time for computing. This is based on the Manhattan-distance:

$$h_{rr'} = 1 - \frac{|\mathbf{r} - \mathbf{r}'|}{\sigma},$$

$$|\mathbf{r} - \mathbf{r}'| = |x - x'| + |y - y'|.$$

For simplification,  $\varepsilon' = \varepsilon$ ,  $h'_{rr'} = h_{rr'}$ , and  $\sigma' = \sigma$  has been chosen.

Both,  $n_N$  and  $n_{L,max}$  have been chosen as a function of  $n_{LV}$ . From several experiments using Kohonen networks, the following formulae were derived:

$$n_N = 10n_{LV} \quad \text{and} \quad n_{L,max} = 10\sqrt{n_{LV}},$$

where  $n_N$  is a natural quadratic number. It is less important that  $n_{L,max}$  be dependent variable than  $n_N$ . Thus, the square root of  $n_{LV}$  is sufficient in this case. These formulae were developed together with the methods for selecting learning vectors and the criterion for breaking off learning.

It can be observed that during the learning procedure the validation error decrease at first but later remain constant or even increase, marking the loss of the capability for generalization and to recall associatively and the beginning of adaptation to training data noise. The training error, on the other hand, continues to decrease, even to zero. The network has then learned by heart, i.e. so-called temporal overfitting or overlearning occurred. To avoid this, a break-off is proposed to be applied to the learning procedure.

Discontinuing learning seems to be incompatible with the process of self-organization. However, it can be demonstrated that during the learning phase, a point does exist when Kohonen networks begin rote learning and loose their capability for generalization. At this point, Kohonen networks start to change from an associative memory to a conventional memory without having relationships among the weights. This point can be found by such methods as visual inspection of U-matrices. However, in order to find this point automatically, a criterion has been developed based for validation and training errors calculated after each learning epoch. This criterion can be described as follows.

Both errors are traced independently of each other. At every temporal local minimum in the validation error, the current network weights are stored. The learning phase is broken off not later than the following condition is fulfilled:

$$mrmse_T < \frac{1}{2}mrmse_V \quad \text{or} \quad mae_T < \frac{1}{2}mae_V.$$

This is called the cut-off condition. The errors being compared are chosen to be *mrmse* or *mae* according to the temporal pattern. The time when the cut-off condition is fulfilled is called the cut-off time. In the first place, the learning phase should be broken off, when a minimum in the validation error occurs a prior to the cut-off time. But in some cases such a minimum does not occur up to this time. Then, to avoid rote learning, the “emergency brake” has to be drawn, which is the cut-off condition. In this case, the break-off time would coincide with the cut-off time and the validation error at this time is defined to be the minimum.

This is called the temporal global minimum to distinguish it from the global minimum of the error surface. If the validation error at the cut-off time is not the temporal global minimum, then the validation error at this time is greater than that at an earlier time, which is then considered the break-off time. Whether this time coincides with the cut-off time or not, the stored network state corresponding to the break-off time is regarded as the final state of the Kohonen network. This means that the Kohonen network has finished learning. The criterion used is called the combined minimum cut-off learning break-off criterion, which is illustrated in Fig. 3. This criterion can be applied only then, if the training errors reach zero, before the number of learning epochs has reached its maximum,  $n_{L,max}$ . This is ensured for every case by the choice of  $n_N$  and  $n_{L,max}$ .

In order to prove that this criterion is compatible with the process of self-organization, the corresponding sequence of U-matrices which can be calculated from the current state of the network after each learning epoch is also shown (Fig. 4). During Epochs 1–4, a very simple coarse structure starts to form. The structure changes considerably from epoch to epoch. During Epochs 32–35, these changes get weaker. During Epochs 46–50, the changes have almost ceased and the structure has stabilized. This time period includes the time the minimum is reached, Epoch 47, and the time the conditions for cut-off prevail, Epoch 49. During Epochs 59–62, a new kind of behavior begins, as reported by Ultsch and Halmans [29]. The structure begins to elevate to an extreme degree, increasing its steepness. The structure which previously displayed only moderate waves starts to become more crude and becomes steepest at Epoch 100. This new behavior indicates that rote learning has begun. Thus, compatibility of the combined learning break-off criterion and the process of self-organization could be demonstrated. Therefore, the criterion provides a good point of generalization. Because this criterion can be applied in every case, conformity also exists for all other cases.

In Figs. 3 and 4, the results using a Kohonen network which has been trained from learning vectors based on a classifying temporal pattern were shown. A total of 100 learning vectors were used to train a map of  $32 \times 32 = 1024$  neurons with a maximum of 100 learning epochs yielding totally 10 000 training steps.

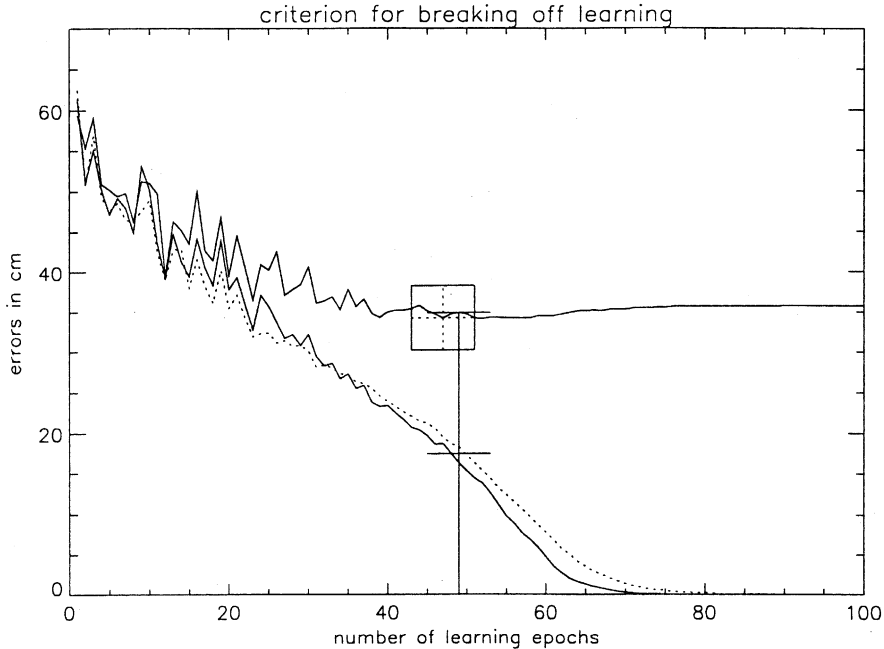


Fig. 3. Validation errors (upper solid line) and training errors (lower solid line) were calculated after each learning epoch. Some of the training errors were calculated according to the quadratic exponential law of decrease for  $\varepsilon$ , while the other training errors (lower dashed line) were determined according to a linear exponential law of decrease for  $\varepsilon$ . The square in the figure surrounds the temporal global minimum for the validation errors, and the long vertical line indicates the time point from which the cut-off condition is fulfilled. A univariate temporal pattern is used. Therefore, validation and prediction errors are identical.

Instead of anomalies,  $z$ -transformed sea levels were used for training. Although these data bear a strong harmonic component and are not distributed randomly, Kohonen networks can be trained by them. The corresponding errors are high (Fig. 3), but the structure that evolves during the learning process can be explained very simply.

The neurons in the map can be regarded as been composed of edges of quadratic frames interlocked symmetrically at the center of the map (Fig. 5). The network assumes that the important information is in the harmonic component. The weights have learned this component in such manner, that portions of tide-like curves can be detected for each weight vector. In each edge neighboring vectors have roughly the same phase shift. In the outermost edge, the phase shift is minimal. In the innermost frame, which consists of only four vectors, the phase shift is maximal. Thus, the distances of the U-matrices increase from the outermost edge of the map to its center, producing the central mountains illustrated in Fig. 4.

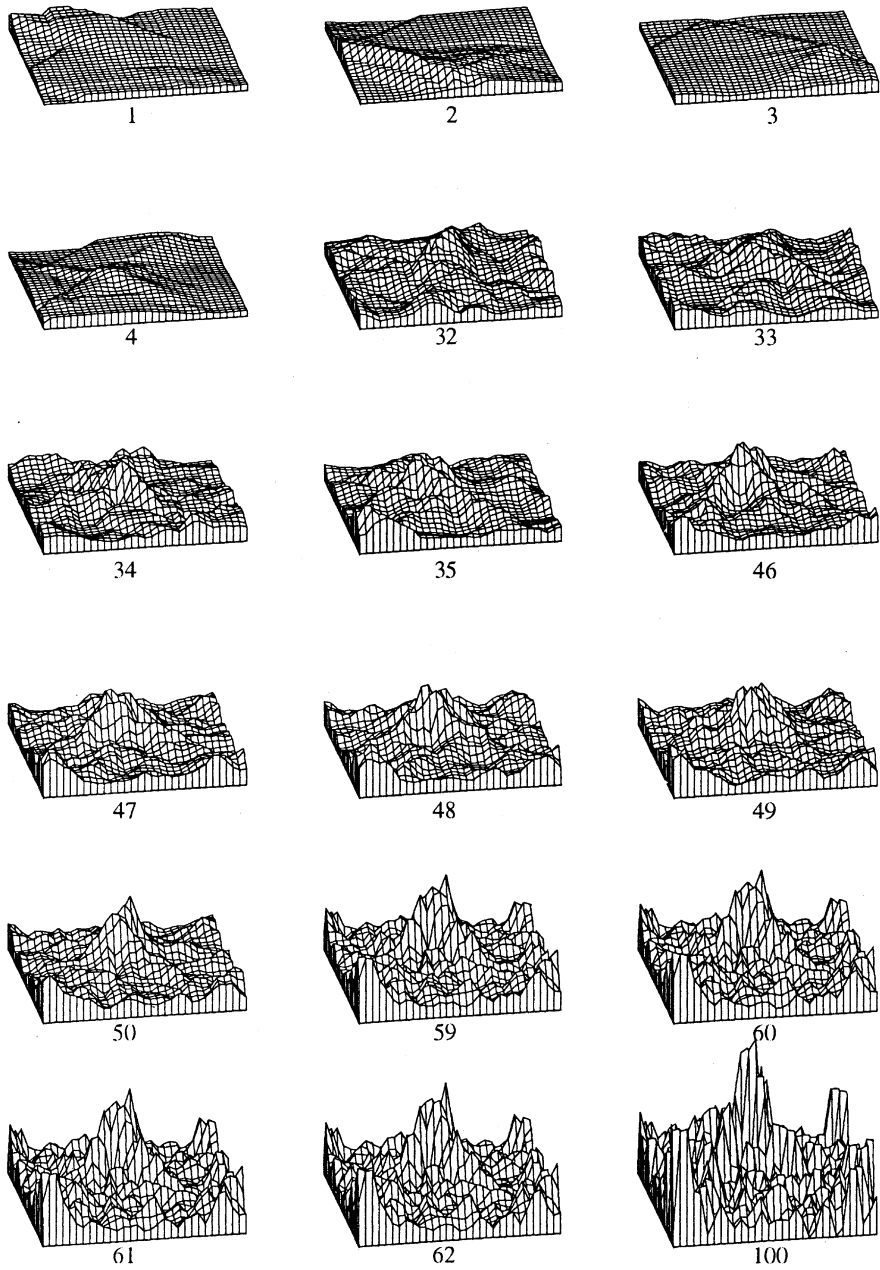


Fig. 4. Sequence of the U-matrices corresponding to the errors in Fig. 3. Network states are shown after selected numbers of the learning epochs. The state of initialization is omitted. The effects of the break-off criterion are: the minimum is after 47, and the moment the cut-off occurs is after 49 epochs.

## Graphical representation of the weights of a Kohonen network

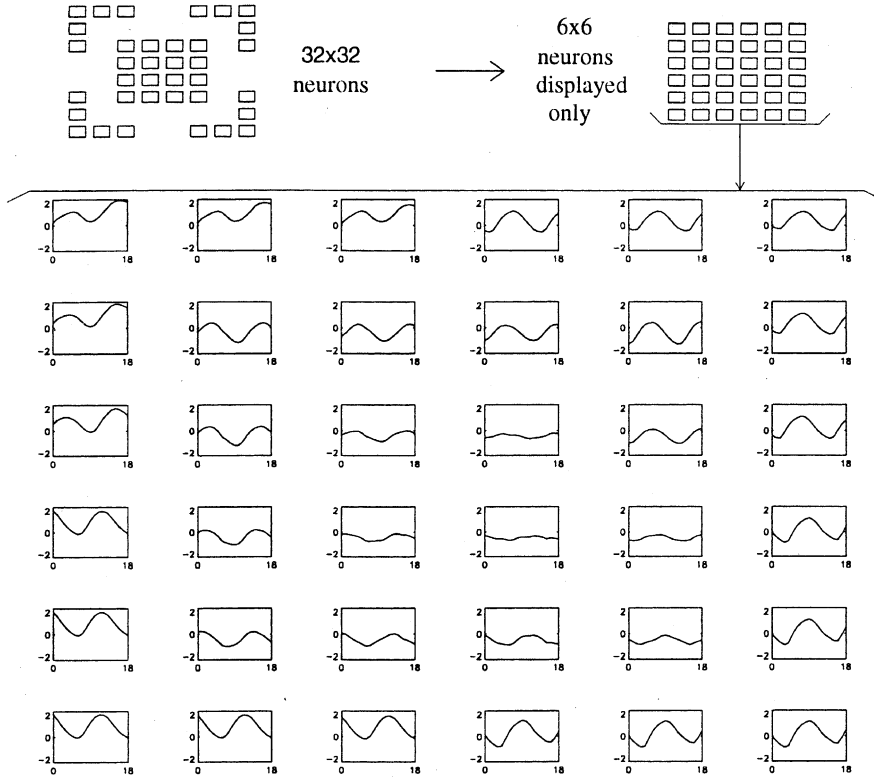


Fig. 5. Weights vectors of a Kohonen network which has finished learning according to Figs. 3 and 4. The upper figure explains the lower figure. Weight vectors of the outermost corners and the innermost kernel of the Kohonen network have been merged together to illustrate the principle how the network has self-organized in this case. The temporal pattern is univariate. Therefore, the weights can be represented as time series.

### 3.6. Description of experiments

For neural network models predicting only hourly anomalies (Table 2), a system was required to simulate the sea level forecast procedure of the BSH. In general, the BSH predicts the second and third extremum to be expected in the future. They lie between the 6th and the 18th hour after the time points the predictions have to be made. Therefore, prognostic time windows per day were defined, which contain 18 hourly, time points each ( $k = 18$ ). These time windows mutually overlap at six time points. Only the final 12 hourly, time points in each time window are used for assessing the forecast skill.

Almost all indicating time windows of the multiwindow neural model were also defined to contain 18 hourly, time points each except that one for air pressure (Table 4). Considering the time lags of all indicating quantities relative to the prognostic quantity of the anomalies at Cuxhaven, the lengths of the prognostic time windows of the models to produce additional predictions can also be specified (Table 4). The first five rows of Table 4 are relating to the neural models which are used to predict anomalies for Cuxhaven. The lower rows are referring to the indicating quantities of these models. These quantities have to be predicted, if the time lag is much lower than 18 hours. It will be attempted to predict these quantities by univariate neural models and to compare them with the forecasts by the SWA and the DWD.

To get best results for the CL model its length  $N$  of the indication time window will be varied. In order to ensure that the indicating and prognostic time window were equally treated regardless of their lengths, the metric Euclidean distances were slightly modified. First, the distances were calculated for every time window. Then, both results were averaged. However, the modified metrics value have no effect because only the indicating time window is used for determining the best match neuron. Nevertheless, it has an effect on the selection of learning vectors.

Only for the MR neural model applied to anomalies at times of high/low tides the anomalies at Wick and West Terschelling will be used as indicating quantities.

Table 4  
General description of the models of comparison

Models	Principle number of variables	Temporal scanning of predictions	Number of values predicted simultaneously	Quantity predicted
Statistical model (“Gesamtansatz”)	Multivariate	Times of high/low tides	1	Anomalies
Hydrodynamic sea level model		Times of high/low tides		Sea levels
Hydrodynamic anomaly model		Times of high/low tides		Anomalies
Nearest neighbor	Univariate, multivariate	Hourly, times of high/low tides	>1	Anomalies
Persistence	Univariate	Times of high/low tides	1	Anomalies
Verbal predictions of the Sea Level Forecast Service		Hourly Times of high tides	>1 1	Anomalies Deviation from the mean high tide

### 3.7. *Models of comparison*

The predictions of the neural networks will be compared with the predictions of six models. The first three are the statistical method called “Gesamtansatz”, a sea level hydrodynamic model, and an anomaly hydrodynamic model.

Multiregression analysis and other methods, such as power series, have been used in the past to identify the most important meteorological factors influencing the local anomalies [3]. The most important of these factors are the speed and direction of the wind. Following the wind in importance are the static air pressure, air pressure changes at three hour intervals, air temperature, and the difference between air and water temperature [2].

The statistically determined relationships between the individual meteorological factors and their effects on the anomalies indicated that interactions between tides and the anomalies occur. Therefore, during the 1970s, a mathematical and physical approach, the so-called “Gesamtansatz” was developed but not documented. It permits a combined analysis of both tidal and meteorological effects on the sea level in a single step [3]. The “Gesamtansatz” revealed that there is a non-linear multiregression relationship between the magnitudes of the meteorological factors and the anomalies [21].

Empirical evidence indicated that most meteorological quantities have their greatest influences on anomalies after a delay of about 3 hours. Therefore, this time lag has been built into the formula. While the local effect of air pressure has also been built into the formula, external surges induced by rapid air pressure variations and travelling around the North Sea have not. This empirical, statistical method “Gesamtansatz” has been re-examined recently by Müller-Navarra and Giese [17]. The coefficients of the model have been found not to differ significantly from those obtained in the 1970s.

During the 1980s, an operational, three-dimensional, numerical hydrodynamic model has been developed by the BSH for the whole North and Baltic Sea employing so-called primitive equations (Krauss, 1973) [6,7]. The model extends also over the Northeastern Atlantic to cover surges coming in from there. Based on meteorological forecasts prepared with the atmosphere model for Europe, the ‘Europamodell’ of the DWD [15], sea levels and other quantities are predicted.

A second hydrodynamical model has been used operationally since the end of 1996, which distinguishes between tides and anomalies due to the wind [13]. It calculates first the tides separately, then tide and wind induced sea level changes together, and subtracts both results from one another in a certain way.

In addition, the nearest neighbor method [18], which is not used by the BSH, was utilized as the fourth model. For an optimal analogy with neural networks, a quadratic measure of similarity, the Euclidean distance, was chosen. Moreover, the simplest model of all, the persistence model, was used as the fifth model. In



general the sea levels and not the anomalies are compared. Consequently, the tidal predictions have to be added to the models predicting anomalies.

The sixth model used to round off the standard of comparison are the verbal predictions that are broadcast and kept on record by the BSH Sea Level Forecast Service. They are based on the results of the first two models, which have been assessed by the experience of the forecasting experts. Therefore, these data are not independent of the other comparison models. Nevertheless, these predicted values have been included because they are the most precise and therefore can be used as a standard for evaluating the neural networks. The Service predicts mainly the extrema of tides, i.e. the high and low tide levels; hourly levels are less important. An overview of all models of comparison is given in Table 4.

#### 4. Results

Using all neural models, hindcasts were made for the year 1993. The results were compared with the predictions of all six comparison models for the same year and with the data actually observed. The training, validation and prediction errors are calculated as the differences between predicted and observed values. These differences are called residues. For predicting hourly anomalies, residues  $\varepsilon_i$ ,  $i = 1, \dots, 8760$ , are defined, according to  $N = 730$  time windows with 12 relevant hourly values each. For predicting anomalies at times of high and low tides, residues  $\delta_n$ ,  $n = 1, \dots, N$ , are defined with  $N = 705$  high tides and  $N = 706$  low tides. The value of  $\varepsilon_i$  is averaged as the mean root mean square error (*mrmse*), and  $\delta_n$  is averaged as the mean absolute error (*mae*) and the standard deviation of error (*sde*) as follows:

$$mrmse = \frac{1}{N} \sum_{j=1}^N \sqrt{\frac{1}{12} \sum_{k=1}^{12} \varepsilon_{k+12(j-1)}^2},$$

$$sde = \sqrt{\frac{1}{N-1} \sum_{n=1}^N \left( \delta_n - \frac{1}{N} \sum_{n=1}^N \delta_n \right)^2},$$

$$mae = \frac{1}{N} \sum_{n=1}^N |\delta_n|$$

( $\varepsilon_i$ ,  $\delta_n$  and  $N$  see text).

First, the prediction of the hourly anomalies determined by univariate neural models is presented. The CL model provided a prediction error 26% lower than that of the nearest neighbor method (Table 5). In addition, the

Table 5  
Hourly and derived hindcasts

Prognostic quantity	Neural models	<i>mrmse</i> l (derived: <i>mae</i> )	Models of comparison	<i>mrmse</i>	<i>mae</i>
Anomalies at Cuxhaven	AR	24 cm	Univariate nearest neighbor (like CL)	30 cm	
Anomalies at Cuxhaven	CL	22 cm/(20 cm)	Persistence	23 cm	
Anomalies at Cuxhaven	MW, RW, MR	$\geq 24$ cm/ ( $\geq 21$ cm)			
Wind speed	CL	4.6 kn/(4.7 kn)	Persistence	4.8 kn	
Wind speed			SWA		4.0 kn
Wind speed			DWD		3.9 kn
Wind direction	CL	41°/(41°)	Persistence	38°	
Wind direction			SWA		24°
Wind direction			DWD		27°
Wind vector	DC	4.6 kn and 41°			
Air pressure	CL	2.4–3.0 hPa	Persistence	2.5–3.7 hPa	
Change of air pressure	CL	1.0–1.4 hPa	Persistence	1.1–1.6 hPa	
Air temperature	CL	1.2 K	Persistence	1.3 K	
Difference of air minus water temperature	CL	1.1 K	Persistence	1.2 K	
Anomalies at West Terschelling	CL	14 cm	Persistence	14 cm	

errors in the predictions made using all neural models (Table 2) are lower than those of the nearest neighbor method (not shown).

While varying the length of the indicating time window of the CL model, it was observed that the prediction errors are minimal when the lengths are minimal ( $N = 1$ ). To avoid that the learning is broken off too close at  $n_{L,max}$ , a length of two time points rather than one was chosen for the indicating time window ( $N = 2$ ).

Taking this minimal length into account and using the modified metric for the selection, a prediction error (*mrmse*) of 22 cm was determined for the CL model (Table 5). This is 1 cm better than the persistence model and 2 cm better than the AR model applied iteratively to make predictions for 18 hours (Table 5). In addition to this larger error, this model required 10 times as much time to compute as the CL model. Furthermore, its break-off criterion for learning did not function because the training error did not decrease fast enough.

Second, results are derived from the prediction of hourly anomalies using multivariate models. The results can be summarized tersely with the statement

that all multivariate models, including the MR model used to make hourly predictions, make greater prediction errors (*mrmse*) than the CL model (Table 5). While searching for evidence that computers with more capacity should improve of sea level forecasts, all neural network models including the MR applied for prediction of anomalies at times of high and low tides were allowed to vary in network size, as defined by the number of learning vectors. When this number was increased from 100 to a maximum number limited by the computing time, 700 for MR models, prediction errors averaged for all temporal patterns decreased by 1 cm.

Third, supplemental findings in addition to these results are reported that are relevant to the predictions of the meteorological variables. These hindcasts can be used as additional predictions for the MW and MR models. Wind speed and direction can be predicted simultaneously employing a DC model, or each component can be predicted separately using CL models. In the DC model, the errors in the predictions of wind direction can be reduced by 14°, or 25%, by employing the jumping treatment, as described above. The prediction errors in the CL models are equal to those in the DC model (Table 5). In each case, the CL models were better than the persistence models, except with regard to wind direction (Table 5).

The time series of air pressure data recorded at selected weather stations in Northwestern Europe were used by the MR models to predict anomalies at times of high and low tides. In order to make additional predictions for these models, air pressure at these stations was also predicted using CL models. Generally, the results are comparable to those determined at Helgoland. The prediction errors of the CL models are not higher than those of the persistence models. In Table 5 these errors are not shown in detail for each station but only summarized.

Wind predictions at times of high tides minus a time lag of 3–4 hours can be calculated from hourly predictions and be compared with the wind forecast by the DWD and SWA (Table 5). A total of 640 predictions were made for 1993. The SWA produces the best forecasts for wind direction. The DWD forecasts are direct model outputs, but the SWA forecasts are produced by meteorologists, who have used the atmosphere model for Europe since 1991 and the atmosphere model for Germany since 1993 [15].

The fourth results concern the prediction of anomalies at times of high and low tides. The main focus was given the anomalies at times at high tides. These can be calculated from the predicted hourly anomalies. However, because these prediction errors, *mae*, are of the same order of magnitude as those of the hourly predictions, *mrmse*, and therefore very high, this method of extracting values at times of high and low tides from the hourly predictions is not further pursued (Table 5). Instead, only the MR model is used, which can be applied to directly predict anomalies at times of high and low tides without producing hourly predictions as an intermediate step.

To predict anomalies at Cuxhaven at times of high tides an MR model trained using vectorially averaged winds and the anomalies at West Terschelling is regarded as the optimal with respect to the choice of oceanographic and meteorological input factors. This optimal model has been found by comparing prediction errors of MR models using several combinations of these and other indicating quantities [21]. After having identified anomalies at times at high tides, it has been assumed to be optimal for predicting anomalies at times of low tides, as well.

Additional predictions for the optimal MR models include wind from the SWA. A CL neural model has been used to predict anomalies at West Terschelling. This model has been compared with the persistence model. Both have a prediction error, *mrmse*, of 14 cm (Table 5). Therefore, a neural network model need not to be used for predicting anomalies at West Terschelling. Predicting these anomalies using the persistence model allows current anomalies to be regarded as additional predictions for the optimal MR models.

Thus far, learning vectors have been selected only by the MD method. In order to reduce the prediction error even more, the method of CGR has also been used. The ratio MD:CGR of the numbers of learning vectors has been optimized with respect to the prediction errors. The optimal ratio at high tide is 50:50, and that at low tide is 25:75. Thus, prediction errors of the optimal low tide anomaly MR model could be decreased by 2–17 cm, and those of the optimal high tide anomaly MR model could be decreased by 1–14 cm, that is 1 cm less than that of the Sea Level Forecast Service (Table 6).

The high tide anomaly MR model encompasses 600 learning vectors,  $78 \times 78$  neurons, a maximum of 250 learning epochs, and in total a maximum of 150 000 learning steps. The low tide anomaly MR model encompasses 300 learning vectors,  $55 \times 55$  neurons, a maximum of 180 learning epochs, and in total 54 000 learning steps. A total of 668 predictions were made for 1993.

Concerning the low tide, the optimal MR neural model seems to be the very best. But the impression deceives. If the standard deviation, *sde*, of the residues is calculated for each model instead of the *mae*, the result is the same for the MR neural model and for the hydrodynamic anomaly model. The last model

Table 6  
Direct hindcasts at times of high/low tides

Neural models	Models of comparison	High Tide ( <i>mae/sde</i> in cm)	Low Tide ( <i>mae/sde</i> in cm)
CL		20	22
Optimal MR		14/22	17/27
	Statistical (“Gesamtansatz”)	22	30
	Hydrodynamic sea level	20	32
	Hydrodynamic anomaly	17/24	40/27
	Sea Level Forecast Service	15/23	

presents a strong bias or systematic error, which raises the *mae*. In hydrodynamic models such a bias is a preserving quantity, and therefore can be considered during the operational forecast if known. Because the standard deviations are the same, an *F*-test should be performed to achieve statistical significance on equality. But unfortunately, the residues of the hydrodynamic anomaly model were not available in detail. Thus, the *F*-test could not be performed.

Concerning the high tide, an *F*-test could be performed in order to evaluate the significance of 1 cm difference between the optimal MR neural model and the verbal forecasts of the Sea Level Forecast Service, because the residues were available. The *F*-test is valid regardless of the distribution because the sample size is sufficiently large [25]. The null hypothesis was tested, to see whether both standard deviations were equal. The null hypothesis can be discarded at the 10% level but not at the 5% level ( $f = 1.12$ ,  $t = 1.4$ ). However, the high tide predictions of the Sea Level Forecast Service are in decimeters. The residues of these predictions were considered to be zero by the Service, if they were between plus and minus one decimeter. Thus, 27% of the residues in 1993 are zero. But the predictions of the neural models are in centimeters. Its residues

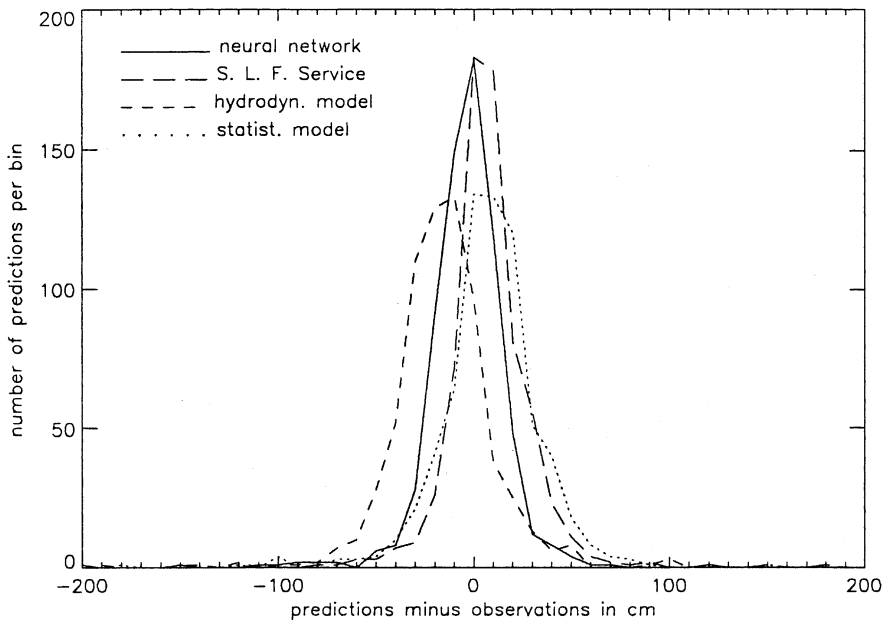


Fig. 6. Distributions of residues for predicting anomalies during high tide at Cuxhaven using four models: (1) the optimal MR model; (2) the Sea Level Forecast Service; (3) the hydrodynamic sea level model; and (4) the statistical model of the “Gesamtansatz”. The bin size is 10 cm.

will very likely not be zero. This makes the predictions of the Service seem better in the comparisons.

Distributions of the residues of four models for predicting anomalies at times of high tides are displayed in Fig. 6. The models are even more superior when they are higher and narrower and better centered around zero in their distribution.

## 5. Discussion

Kohonen networks are characterized by two basic ideas, that of self-organization and that of preservation of topology. These ideas can be utilized for forecasting problems by transforming the unsupervised feature map into a supervised one. There are neural networks which may be the first choice for forecasting problems, the multilayer feed forward networks or perceptrons using the error backpropagation algorithm [22] or so-called back-propagation networks. But these networks pose a number of problems. The most serious is that of convergence, i.e. the failure of finding the global minimum in the error surface [27]. The convergence is influenced by the initialization of the networks, which is usually a random process, by the training data and by the network architecture, pointing to the problem of over-fitting [19].

There are two other types of neural networks, the time delay and the recurrent neural networks, which are related to the back-propagation networks. The time delay neural network is functionally equivalent to the finite-duration impulse response (FIR) multilayer perceptron [30]. The synaptic FIR-filter has been developed, to allow for dynamic presentation. An FIR multilayer perceptron is trained by the temporal back-propagation algorithm. However, when it has finished learning, all weights are fixed again [12]. Thus, representation is static again. The recurrent neural network uses a normal threshold model for the neurons, but information of output neurons is routed back to the input neurons [33]. During this dynamic presentation weights are changed continuously. Recurrent neural networks need a lot of computing time [5]. They are particularly suitable for statistically variant signals like speech. However, concerning medium-range time scales, the data required for the sea level forecast are statistically rather invariant.

Just as the back-propagation networks have problems of convergence, the Kohonen networks have also such problems, but they are less pronounced. To encounter such problems of back-propagation networks several concepts have been developed. Two of them may also help to reduce such problems of the Kohonen networks.

One of these concepts attempts to solve the problem of overfitting. There are statistical arguments which suggest that the number of training patterns required to fully determine the weights in a network is approximately

proportional to the number of weights in the network [31,32]. In order to avoid overloading the networks and preventing them from forgetting what they have learned, the number of neurons has been chosen as a function of the number of learning vectors. This functional relationship is proposed to be applied to Kohonen networks, which has never yet been attempted before, as far as the authors know. The larger the amount of material to be learned by increasing number of learning vectors, the more time is needed for learning. In addition, when more material must be learned, there are more relationships within the material to be learned, and disproportionately more time is required for learning. Therefore, the number of the maximum number of learning epochs has also been chosen as a function of the number of learning vectors.

Another concept to encounter the problems of convergence is to break off learning. This can avoid the network to stay too long inside the plateau of the learning curve, which is characteristic for back-propagation [9]. Breaking off learning also deals with overfitting [31]. For [1] the choice of a suitable break-off criterion is imperative, because the back-propagation training process is carrying out its gradient descent *ad in-finitum*. He proposes a criterion, which also considers the problem of overfitting. This criterion does not only use the validation errors but also the training errors in linear combination with the first ones. He also proposes to store the network weights when a new temporal minimum of these combined errors is reached. However, breaking off the learning eliminates only the symptoms but does not cure the cause of the problem, as Masters [16] argues. He recommends, when using back-propagation networks, architecture and initialization should be improved instead of breaking off learning.

With regard to the architecture, however, Kohonen networks show a reverse behavior as compared with the back-propagation networks. The prediction errors of the former ones decrease instead of increasing as the number of neurons increases. With regard to the initialization, Kohonen networks do not vary so much with different initializations. When different sets of random numbers are used, the representation errors of the back-propagation networks can vary from each other by about 10% [9]. When combined with the MD method for selection, which must also be initialized with random numbers, the prediction errors of the Kohonen networks vary only 2–3% using different sets of random numbers. Therefore, no arguments remain to reject breaking off learning. An argument is gained even for applying it, by the vanishing of the training errors which indicates rote learning or temporal overfitting.

To avoid this, a break-off is proposed to be applied to the learning procedure of the Kohonen networks, which has never yet been attempted before, as far as the authors know. The combined minimum cut-off learning break-off criterion is suggested that allows both, the validation and training errors, to be traced independently from each other in contrast to the criterion of [1] in order to account separately for the loss of the capability for generalization and for

the rote learning. This criterion can only work if the network weights are stored as [1] proposes. But in this case the combined errors are not used but only the validation errors instead. This break-off criterion could be shown to be conform with the process of self-organizing by using U-matrices for visualization of the learning procedure. Breaking off the learning procedure can also help finding the global minimum in the error surface which can also be defined for Kohonen networks.

Sea levels or anomalies have been predicted univariately. Other designations for the univariate predictions employing back-propagation networks have been used in the literature on neural networks, including single-step prediction for autoregression and multistep prediction for classification [31]. It has been found that multistep predictions of anomalies using Kohonen networks outperform single-step predictions. In addition, it has been found, that the number of indicating values in the univariate temporal pattern has to be as small as possible. This contradicts the mathematical hypothesis that mapping between two spaces is best if both spaces have equal dimensions. However, this finding does not invalidate the method Ultsch and Halmans [29] used for predicting hail storms, and can be used to confirm the quality of these predictions.

The Sea Level Forecast Service found that data on the wind are absolutely necessary to predict the anomalies well, that means a multivariate approach is imperative. Therefore, it was expected that additional consideration of the wind data in the neural networks would make it possible to make hourly predictions with less error than that of the classifying model. The reason that this expectation was not fulfilled is that Kohonen networks with longer vectors are less suitable for filtering out the important information during the learning phase. It seems that the error-reducing effect of the wind data is counteracted by the greater length of the vectors. Therefore, to lower the prediction errors, vectors have to be shortened, and, at the same time, the wind data may not be omitted. It is not possible to fulfill both requirements when predicting hourly anomalies, but it is possible when predicting anomalies at times of high and low tides using multiregression neural models.

Regarding the prediction of meteorological quantities to obtain additional predictions for the multivariate forecast of the anomalies, it has been found, that the Kohonen networks produced a prediction of the wind direction inferior to that of the persistence model. In contrast to wind speed, wind direction remains a vectorial quantity, but the spatial structure is not considered in the classifying neural model. Apparently, this model learns a false relationship in this way.

The observation that the error-reducing effect of additional data is counteracted by the longer vectors as mentioned above could also be observed when a multiregression neural model was trained with air pressure data from weather stations in North-western Europe in addition to the wind data. Continuing on this way of searching for a Kohonen network which is minimal with respect to the length of weight and learning vectors and is the best with respect to the



prediction errors, a multiregression neural model has been found whose weight vectors consist of only four components. The model predictions have been compared with observations and with all models of comparison. After an examination of the residues including an  $F$ -test applied to them, it can be reported with little reservation that the Kohonen networks are superior not only to all existing models but also to the verbal forecasts of the Sea Level Forecast Service.

## 6. Conclusion

With this paper we could show that Kohonen networks can predict sea levels without using explicit knowledge of the non-linear interrelationships of the meteorological and oceanographic processes even better than the other methods which use such knowledge. Therefore, the networks can model these processes implicitly better than the hydrodynamic and statistical methods can explicitly. It was shown that learning from the past is possible for standard situations. The results show that a limited automation of the forecast procedure has become possible. Therefore, the experience of prediction experts could be placed on an objective basis to a certain degree. It was observed that the predictions are the better, the smaller the weight and learning vectors are and the more the situation is condensed before training. It could be shown, that an improvement of the forecast skill can be expected by employing computers with greater capacities. We took a batch learning algorithm, because the statistical properties do not change very much over medium time scale. However, the question arise, what will happen to these properties over the long term. Therefore, it is recommended to continue learning taking expanded time series into consideration.

## Acknowledgements

The authors would like to thank the staff of the Federal Maritime and Hydrography Agency (BSH) in Hamburg for providing the basic objectives of developing a forecasting system using neural networks and for financial assistance to carry out the research. The authors are grateful to the staffs of the other institutes that provided data sets: the Marine Weather Service in Hamburg, the British Oceanographic Data Centre in Merseyside, Great Britain, and the Directoraat-Generaal Rijkswaterstaat, Rijksinstituut voor Kust en Zee in The Hague, The Netherlands.

## References

- [1] U. Anders, *Multivariate Zeitreihenanalyse mit neuronalen Netzwerken*, Diplomarbeit, Universität Karlsruhe, Karlsruhe, 1993.

- [2] R. Annutsch, Wasserstandsvorhersage und Sturmflutwarnung, *Der Seewart* 38 (5) (1977) 185–204.
- [3] R. Annutsch, Entwicklung des Gezeiten- und Windstau-Dienstes. Sonderdruck aus: *Meeresforschung in Hamburg*, Hrsg. v. Gerd Wegner, Deutsche Hydrographische Zeitschrift, Ergänzungsheft, Reihe B, Nr. 25 (1993).
- [4] H.-H. Bock, Automatische Klassifikation, Theoretische und praktische Methoden zur Gruppierung und Strukturierung von Daten (Cluster-Analyse), Vandenhoeck & Ruprecht, Göttingen, 1974.
- [5] M. Caudill, Putting Time in a Bottle, *AI EXPERT*, 1993.
- [6] S. Dick, K.C. Soetje, Ein operationelles Ausbreitungsmodell für die Deutsche Bucht, *Deutsche Hydrographische Zeitschrift, Ergänzungsheft Reihe A*, Nr. 16 (1990).
- [7] Deutsche Meteorologische Gesellschaft, Das Operationelle Modellsystem des BSH. *Mitteilungen DMG Nr. 3*, Traben-Trarbach, 1993.
- [8] J.B. Elsner, A.A. Tsonis, Nonlinear prediction, chaos, and noise, *Bulletin American Meteorological Society* 73 (1) (1992).
- [9] C. de Groot, Nonlinear Time Series Analysis with Connectionist Networks, IPS Research Report No. 93-03, Diss. ETH No. 10038, Zürich, 1993.
- [10] J. Hartung, B. Elpelt, K.-H. Klösener, *Statistik. Lehr- und Handbuch der angewandten Statistik. 5.*, durchges. Aufl., Oldenbourg, München, 1986.
- [11] J. Hartung, B. Elpelt, *Multivariate Statistik 4.*, durchges. Aufl., Oldenbourg, München, 1992.
- [12] S. Haykin, *Neural Networks. A Comprehensive Foundation*, Macmillan College Publishing Company, 1994.
- [13] F. Janssen, Auswirkung unterschiedlicher Windschubspannungsansätze auf die Qualität von Wasserstandsvorhersagen mit einem hydrodynamisch-numerischen Nordseemodell, Diplomarbeit am Bundesamt für Seeschifffahrt und Hydrographie, Hamburg, 1996.
- [14] T. Kohonen, Self-Organization and Associative Memory, in: *Springer Series in Information Sciences*, vol. 8(3), Springer, Berlin, 1989.
- [15] D. Majewski, Short Description of the Europa-Modell (EM) and Deutschland-Modell (DM) of the Deutscher Wetterdienst (DWD) as at July 1993. Internal Publication of Research Department of DWD, Offenbach, 1993.
- [16] T. Masters, *Practical Neural Network Recipes in C++*, Academic Press/Harcourt Brace Jovanovich, Boston, 1993.
- [17] S.H. Müller-Navarra, H. Giese, Improvements of an Empirical Model to forecast wind surge in the German Bight, *Deutsche Hydrographische Zeitschrift*, Hamburg, 2000 (submitted).
- [18] G. Nakhaeizadeh, Learning Prediction of Time Series. A Theoretical and Empirical Comparison of CBR with some other Approaches, in: M.M. Richter (Ed.), *Proceedings-Band des Work-shop EWCBR-93*, Universität Kaiserslautern, 1993 (to appear).
- [19] H. Rehkugler, T. Poddig, Anwendungsperspektiven und Anwendungsprobleme von Künstlichen Neuronalen Netzwerken, *Information Management* 2/92, 1992.
- [20] H. Ritter, T. Martinetz, K. Schulten, *Neuronale Netze, Eine Einführung in die Neuroinformatik selbstorganisierender Netzwerke*, Addison-Wesley, Reading, MA, 1992.
- [21] F. Röske, Sea level forecasts using neural networks, *Deutsche Hydrographische Zeitschrift* 49 (1) (1997) 71–99.
- [22] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, *Nature* 323 (1986) 533–536.
- [23] R.B. Schlittgen, *Einführung in die Statistik, Analyse und Modellierung von Daten. 3.*, durchges. Aufl., Oldenbourg, München, 1991.
- [24] R.B. Schlittgen, H.J. Streitberg, *Zeitreihenanalyse. 5.*, völlig überarb. und erw. Aufl., Oldenbourg, München, 1994.
- [25] C.-D. Schönwiese, *Praktische Statistik für Meteorologen und Geowissenschaftler*, Borntraeger, Berlin, 1992.

- [26] H. Späth, Cluster-Formation und-Analyse, Oldenbourg, München, 1983.
- [27] A. Ultsch, Konnektionistische Modelle und ihre Integration mit wissensbasierten Systemen, Forschungsbericht Nr. 396, Universität Dortmund, 1991a.
- [28] A. Ultsch, G. Halmans, R. Mantyk, CONKAT: A Connectionist Knowledge Acquisition Tool, Department of Computer Science, University of Dortmund, 1991b.
- [29] A. Ultsch, G. Halmans, Self Organizing Neural Networks and hailstorm prediction, in: Proc. Annual Meeting Ges. f. Klassifikation, Oldenburg, März, 1994.
- [30] E.A. Wan, Time Series Prediction by Using a Connectionist Network with Internal Delay Lines, In: Weigend and Gershenfeld (1993), 1993.
- [31] A.S. Weigend, B.A. Hubermann, D.E. Rumelhart, Predicting the future: a connectionist approach, International Journal of Neural Systems 1 (3) (1990) 193–209.
- [32] A.S. Weigend, N.A. Gershenfeld, in: A.S. Weigend, N.A. Gershenfeld (Eds.), Proceedings of the NATO Advanced Research Workshop on Comparative Time Series Analysis, vol. XV, Santa Fe Institute, Addison-Wesley, Reading, MA, 1993.
- [33] R.J. Williams, D. Zipser, A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. Communicated by Fernando Pineda, Neural Computation 1, 1989, pp. 270–280.