

On the Application of Machine Learning Techniques to Regression Problems in Sea Level Studies

MAGNUS HIERONYMUS AND JENNY HIERONYMUS

Swedish Meteorological and Hydrological Institute, Norrköping, Sweden

FREDRIK HIERONYMUS

Gothenburg University, Gothenburg, Sweden

(Manuscript received 25 February 2019, in final form 3 July 2019)

ABSTRACT

Long sea level records with high temporal resolution are of paramount importance for future coastal protection and adaptation plans. Here we discuss the application of machine learning techniques to some regression problems commonly encountered when analyzing such time series. The performance of artificial neural networks is compared with that of multiple linear regression models on sea level data from the Swedish coast. The neural networks are found to be superior when local sea level forcing is used together with remote sea level forcing and meteorological forcing, whereas the linear models and the neural networks show similar performance when local sea level forcing is excluded. The overall performance of the machine learning algorithms is good, often surpassing that of the much more computationally costly numerical ocean models used at our institute.

1. Introduction

Global sea level rise is perhaps the most severe consequence of the ongoing climate change. The global mean sea level rise today is around 3 mm yr^{-1} , and it is projected to increase fourfold or more until the end of the century in the RCP8.5 scenario (Church et al. 2013; DeConto and Pollard 2016). The economic consequences for Europe in terms of flood damage from such a scenario were recently estimated by Vousdoukas et al. (2018). They estimated that, unless coastal defenses were upgraded, the expected annual cost of coastal flooding could increase by as much as a factor of 770. The stakes are thus high when it comes to adaptations for future sea level rise. Moreover, the regional variations in both sea level rise and sea level variability are large and have to be considered when planning for future sea level rise (Vousdoukas et al. 2017; Mitrovica et al. 2018; Melet et al. 2018). Such knowledge of local conditions can, however, be gained from the often extensive sea level records kept by local authorities around the world. The topic of this paper is the application of

machine learning techniques to solve some common regression problems that one may encounter when analyzing such data.

The sea level of the future is uncertain and depends crucially on how greenhouse gas emissions evolve over the coming decades. However, even if greenhouse gas emissions were to cease completely in the near future, the long equilibration time scales of the ocean and the ice sheets entail that we would still experience sea level rise for centuries to come from the warming that has already occurred (Levermann et al. 2013; Hieronymus 2019). Planning for coastal flooding is therefore an important objective for coastal communities around the world. The long and high-resolution observational time series of sea level, the treatment of which is the topic of this paper, is a key ingredient for making such flooding plans.

Machine learning techniques are today widely applied to solve, for example, classification, clustering, and regression problems across many different disciplines (van Maanen et al. 2010; Al-Jarrah et al. 2015; Leahy et al. 2018). In this paper, we will use historical sea level data from tide gauges situated at various locations around the Swedish coast. However, the method is universal and can just as easily be applied to data from other locations.

Corresponding author: Magnus Hieronymus, hieronymus.magnus@gmail.com

The Swedish sea level records contain many of the longest nearly continuous sea level series in the world (Ekman 1999) and are therefore invaluable resources for climate research. Moreover, several of the sites have data with 1-h resolution that stretch back more than 100 years in time, thus providing an excellent basis to study sea level extremes, coastal flooding, and other ocean dynamics that occur on short time scales (Hieronymus et al. 2017, 2018).

However, the great length of these records also poses some problems such as changes in measurement equipment and equipment placement, inconsistencies in the treatment of missing data, and errors in the digitalization of older data. Missing data exist on time scales from a few hours up to more than a year. The inconsistency in the missing data treatment occurs only on shorter time scales at which some of the older stations during certain time periods have reported values that are linearly interpolated in time, whereas others have simply reported “NaNs.” Linearly interpolating sea level in time is of course highly inaccurate, and a primary objective of this study is to quantify how well machine learning algorithms can be adapted to estimate those missing values. Besides the direct benefit of being able to reconstruct missing data, one can also use these regression methods to look for outliers in existing data that could correspond either to interesting oceanographic events or to, for example, digitalization errors. Another direct application of machine learning techniques that we hope to try out in the near future is the production of boundary forcing data for regional ocean models. The regional ocean model configurations most often used at our institute have two open boundaries in the North Sea, and wind-driven sea level variations at those boundaries are generally not known for future climate scenarios. When running such scenarios, one is therefore forced to choose between having no boundary data or having to first run a computationally costly downscaling experiment with a storm surge model to produce the required forcing. The latter option is often considered to be too costly, and this has placed some major restrictions on studies of sea level extremes from dynamically downscaled climate scenarios produced at our institute. However, statistical downscaling is cheap, and the required boundary data could be produced with the same regression techniques applied here to data from the Swedish tide gauge records. Another primary objective of this study is therefore to assess the viability of such an approach.

The machine learning methods that we have applied are three different artificial neural networks (ANNs) and multiple linear regression. All methods employ

supervised learning. That is, the chosen regression model is presented with input and corresponding output data from which it should “learn,” and the model is then tested on unseen data to assess how well it generalizes. A key difference between the ANNs and the multiple linear regression model is that ANNs can handle non-linearity intrinsically whereas the linear model has to be presented with transformed inputs to handle non-linearity. The performance difference between these methods can therefore be seen as a measure of the nonlinearity of the problem. A detailed introduction to the different ANNs used, as well as to the data, is found in the data and methods section, which is followed by a results section and a conclusions section.

2. Data and methods

a. Data

Figure 1 shows the location of the nine tide gauge stations used in this evaluation. Currently there are 23 such stations in operation around the Swedish coast, and our subset of nine stations is chosen on the basis of having nearly continuous data at hourly resolution for the years between 1961 and 2005. During these years, we also have ERA-40 data dynamically downscaled onto a European domain covering the latitudes between 27° and 72°N and longitudes between 22°W and 45°E (Uppala et al. 2005; Samuelsson et al. 2011; Berg et al. 2013). As forcing for our regression models, we use the winds and sea level pressures from the down-scaled ERA-40 data together with sea surface height (SSH) observations from the nine tide gauge stations. Moreover, to keep the forcing dataset at a reasonable size we make EOF decompositions (Hannachi et al. 2007; Monahan et al. 2009) of the forcing fields over their whole domains and use only the time series, that is, the principal components (PCs), as forcing. The PCs of the meteorological forcing are the same for all regression models, but the SSH PCs vary slightly from station to station since we remove the data from the station we intend to model when computing its SSH forcing EOFs. That is, the SSH PCs used to force the Stockholm station are computed using all stations other than Stockholm. However, the first five SSH PCs are very similar for all stations except Visby and Smögen, having an average correlation coefficient with the respective PCs computed using of all the stations of 0.98. For Visby, there is a similarly high consistency for the first three PCs and for Smögen for the first two. Table 1 shows the variance explained by the different EOFs.

The sea level dynamics are different inside and outside the Baltic Sea, particularly when it comes to tides,

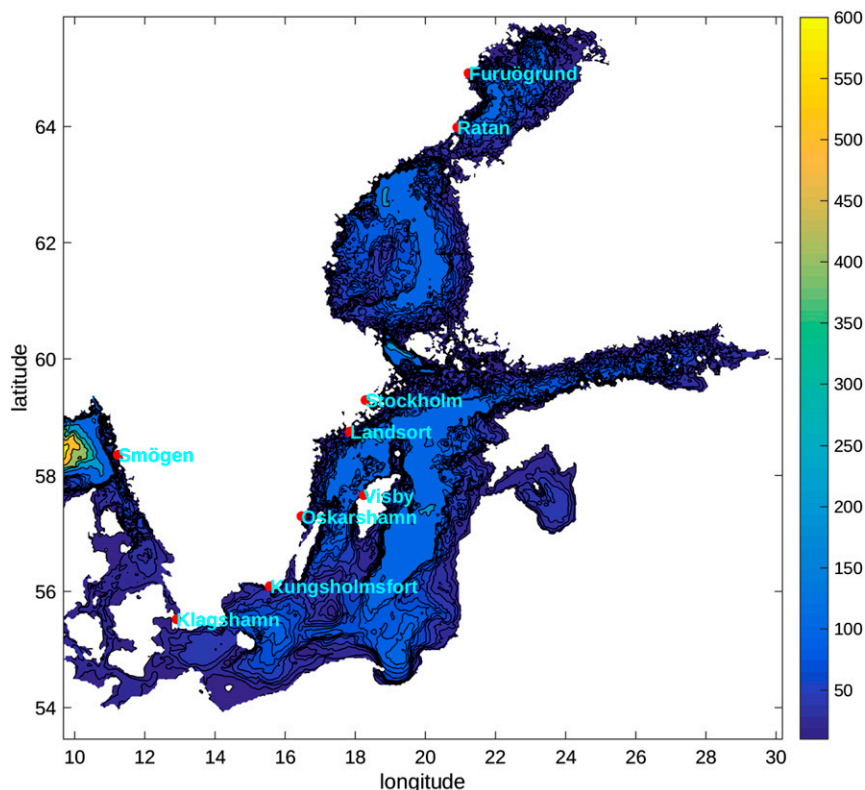


FIG. 1. Bathymetric chart of the study area, also showing the locations of the nine tide gauge stations. Depth contours are drawn every 10 m between 0 and 100 m and every 100 m between 100 and 600 m.

which are virtually absent in the Baltic Sea (Zijl et al. 2013; Kulikov and Medvedev 2013; Hieronymus et al. 2017). The semienclosed nature of the Baltic Sea also makes it more sheltered from Atlantic Ocean storm surges. The sea level variability in the Baltic Sea is consequently mostly driven locally by the wind and is governed by standing waves (Samuelsson and Stigebrandt 1996). The northern part of the Swedish west coast, here embodied solely by the Smögen station, is in contrast to the Baltic Sea affected both by tides and Atlantic storm surges. Sea level regression at Smögen is thus expected to be harder than at the other stations for two different reasons. The first being the richer sea level dynamics, which imply a more complicated forcing–response relationship than in the Baltic. The second is the distribution of the stations, where we have a reasonable spatial resolution in the Baltic Sea while Smögen is almost alone on the west coast. The latter point does not affect the meteorological forcing, which is well resolved everywhere. However, the useful information for sea level predictions at Smögen deductible from Baltic sea levels is likely restricted to crude information about the large-scale state of the atmosphere through a strong connection between Baltic

sea levels and the North Atlantic Oscillation (Andersson 2002) and about the phase of the tides.

The sea level time series are linearly detrended to remove the joint effect of postglacial rebound and mean sea level rise. Apart from this, there are no corrections applied to any of the forcing sets.

b. Methods

Three different ANNs from the MATLAB software “Deep Learning Toolbox” are used in this investigation:

TABLE 1. Variance explained by the forcing EOFs; p is sea level pressure, u is zonal wind, v is meridional wind, and SSH is sea level. Note that the sea level EOFs here are computed using all of the stations. These modes are thus not identical to those used as forcing where one station is always removed from the computation.

	p	u	v	SSH
Mode 1	26.1%	18.4%	19.2%	65.3%
Mode 2	19.2%	15.9%	14.8%	18.2%
Mode 3	16.1%	8.2%	8.6%	7.5%
Mode 4	8.0%	5.1%	5.9%	4.9%
Mode 5	6.6%	4.5%	4.5%	2.7%
Mode 6	4.4%	3.6%	4.0%	0.56%
Mode 7	3.4%	3.2%	3.5%	0.34%
Mode 8	2.5%	2.9%	2.9%	0.23%

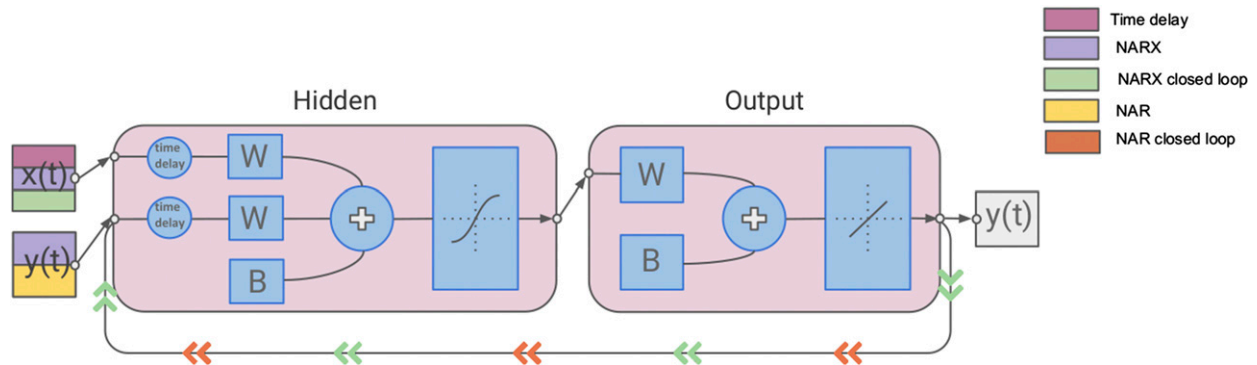


FIG. 2. Schematic of the architecture of the different network types used in this paper; W are the weights, and B are the biases. A hidden node is shown on the left, and an output node is shown on the right. The different networks use different inputs, and the demanded inputs of the different networks are color coded in the left boxes showing the input signals $x(t)$ and $y(t)$. The closed loop mode in which the predicted values are fed back into the network exists for the NAR and NARX network and is illustrated with the line on the bottom connecting the output and hidden nodes. The amount of hidden nodes and the length of the time delays are different in our different applications, and those are reported in their respective figure captions. The number of output nodes is always equal to the number of feedback inputs $[y(t)]$.

the time delay ANN, the nonlinear autoregressive with external input (NARX) ANN, and the nonlinear autoregressive (NAR) ANN. The networks differ by their requirements of input data and are therefore suitable for slightly different problems. A schematic of these networks and the closed and open loop form of the two networks for which it is applicable is shown in Fig. 2. The basic structure of all networks is the same and consists of an input layer, a hidden layer, and an output layer. The input layer is shown farthest to the left in Fig. 2 and shows the external inputs $x(t)$ and feedback inputs $y(t)$ used by the respective network. Both $x(t)$ and $y(t)$ can be more than one time series. The feedback inputs are past values of our desired output series. That is, in this study it is always time series of SSH. The external inputs are time series of parameters other than our desired output such as, for example, surface winds, sea level pressure, or SSH from a neighboring station.

The schematic for the hidden layer shows the time delay farthest to the left. This is the number of past values of external and feedback inputs used to calculate the output $y(t)$. Here we use time delays between 12 and 48 h. The hidden layer contains a number of hidden nodes. Only one is depicted, but we use between 8 and 40 for the different applications. The sigmoid function is used as activation function throughout this paper. The output from each hidden node is essentially a linear combination of the inputs that is transformed by the sigmoid function according to

$$h_i = f \left(B_i + \sum_{j=1}^n \sum_{k=1}^m W_{jk} x_{jk} + \sum_{j=n+1}^p \sum_{k=1}^m W_{jk} y_{jk} \right), \quad (1)$$

where f is the sigmoid function, W_{jk} are the input weights, x_{jk} are the external inputs, y_{jk} are the feedback inputs, and B_i is the bias. Here we have n external input time series with a time delay of m and $p - n$ feedback time series also with a time delay of m . The total number of weights as well as inputs is thus $p \times m$.

The output nodes have the same basic structure as the hidden nodes. However, since they operate on data transferred from the hidden nodes, the total number of weights as well as inputs is equal to the number of hidden nodes. The number of output nodes meanwhile is equal to the number of feedback outputs. The activation function for the output nodes is linear. The open and closed loop configurations only exist for the networks where $y(t)$ is both an input and output variable. The networks can only forecast one time step ahead in open loop configuration, but in the closed loop configuration the forecast $y(t)$ value is fed back into the network so that arbitrarily long forecasts can be done. This is illustrated with the line connecting the input and output nodes in Fig. 2.

Training the networks is always done in open loop mode since closed loop training did not improve performance. For training we use 184 465 hourly data points from 1982 to 2004, and the last 10 000 data points in the set corresponding to 2004–05 data are saved for testing. The years between 1961 and 1982 were at first also intended to be used as training data. However, memory requirements made it hard to use such a large training set, and some of these data were instead used as additional testing data to see whether results were consistent for different periods. The training dataset is further split into 75% training data, 15% validation data, and 15% test data. Training is

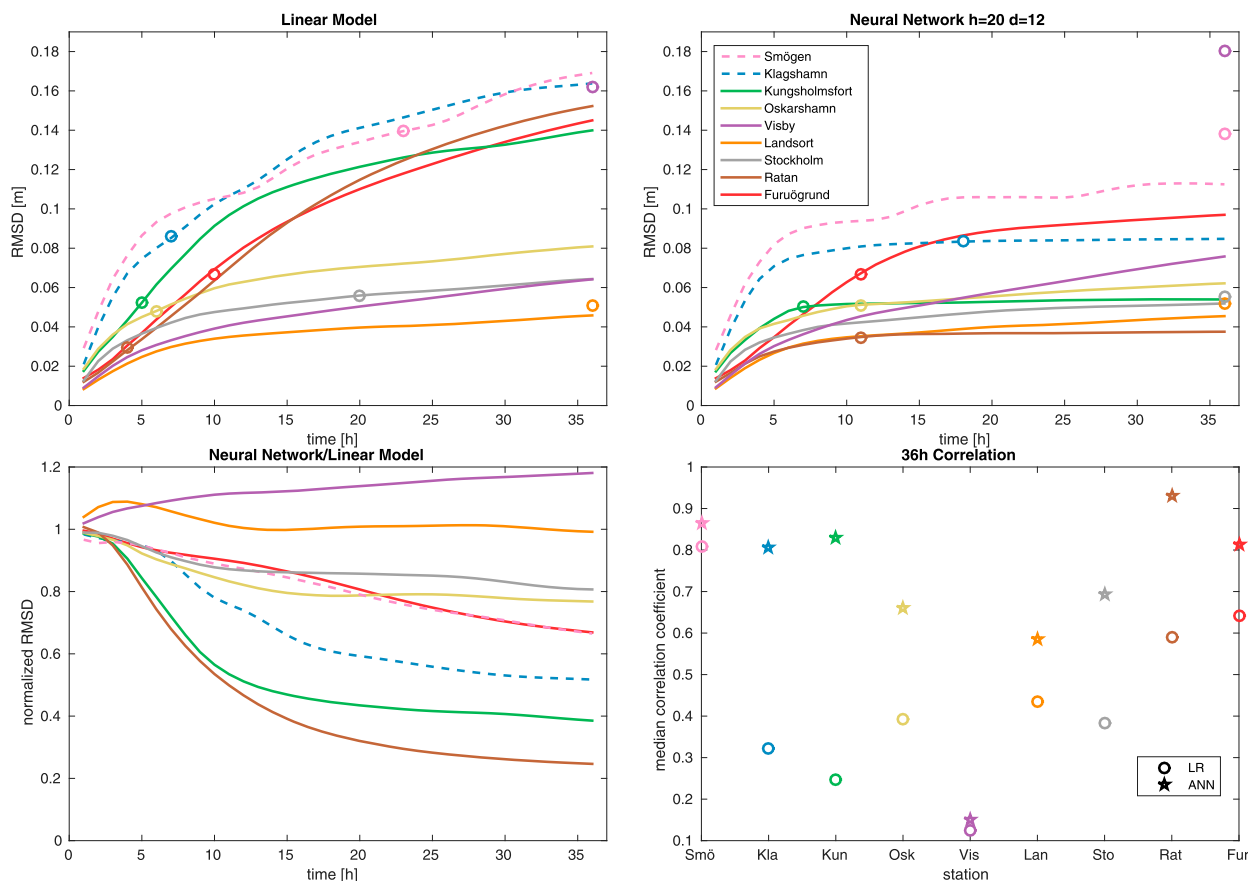


FIG. 3. Mean RMSD, median correlation coefficients and the ratio of the RMSD over multiple 36-h forecasts for the linear regression model and the NARX neural network. Open circles in the neural network panel show the RMSD from the time delay network. Those are plotted at the x value at which the error from the time delay network is closest to that from the NARX ANN (i.e., the NARX ANN is superior to the time delay ANN for projections shorter than the x value where the circle is plotted). The open circles in the linear model panel are done in the same way, but instead of using the time delay network we use a linear model forced with the same input as the time delay network. Solid lines are stations in the Baltic, and dashed lines are outside the Baltic. The d and h in the neural network panel title indicate the delay used and the number of hidden nodes, respectively.

done using the Levenberg–Marquardt backpropagation algorithm, which is the standard MATLAB choice for this network. However, we also tried a few other algorithms, but this had no discernible impact on the quality of the forecasts.

To compare the performance of the ANNs with a different kind of model, we have also set up multiple linear regression models using the MATLAB standard “fitlm” routine. The fitting of these linear models is done using ordinary least squares. In each comparison of a linear model and an ANN, we use exactly the same feedback inputs, external inputs, and delays for both models.

3. Results

The mean root-mean-square deviation (RMSD) and median correlation coefficient from 36-h forecasts started every 1 h over the last 10 000 h in the dataset (i.e.,

over the unused test set) for the linear regression model and the NARX neural network are shown in Fig. 3. Solid lines indicate stations in the Baltic Sea, and dashed lines indicate stations outside or, in the case of Klagshamn, on the boundary of the Baltic.

The NARX network is superior, and in fact is vastly superior, to the linear model for most stations with the exception of Visby. The Visby station forecasts, for both the linear and NARX models, are very poorly correlated with the observed values and are thus not useful. The RMSD at Visby appears to be decent, but this is the result of low variability rather than good forecasting skill, which will be obvious later on when nondimensional RMSD is presented. Both models used here are forced with exactly the same data and use the same 12-h time delay. The feedback input consists initially of 12 prior hourly sea level observations, but since the models operate in the closed loop

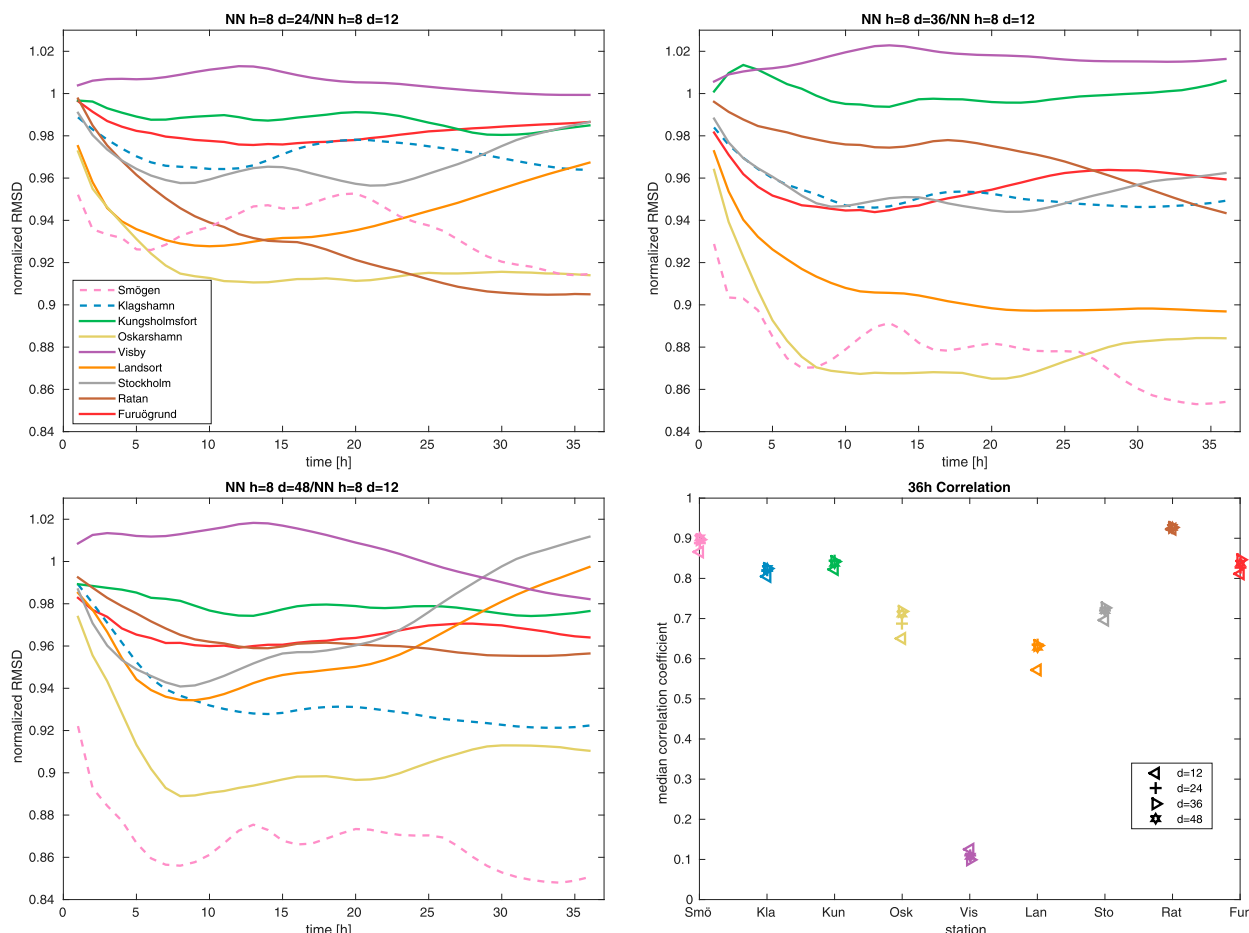


FIG. 4. Normalized RMSD in NARX projection for different delays and 36-h correlation coefficients from the various experiments. The RMSDs are normalized with those from the NARX network with a 12-h delay. Solid lines are stations in the Baltic, and dashed lines are outside the Baltic. The d and h in the panel titles indicate the delay used and the number of hidden nodes, respectively.

mode these are gradually replaced with predicted values. The external input data consist of the first three PCs for meridional wind, zonal wind, sea level pressure, and SSH from the tide gauge stations. The reason for the failure of the Visby forecast is not known to us, and we will see later on that the predictions are much better for other testing datasets, suggesting that the failure is at least in part related to the forcing. A plausible explanation is that the sea level at Visby is more strongly governed by local dynamics than at other stations and thus is harder to predict.

The open circles in the panel showing the RMSD of the neural network show the RMSD from the time delay neural network evaluated over the whole test set. The same is done in the linear case, but with a linear model using the same data as the time delay network. These open circles thus indicate the quality of predictions done using the same external inputs as when computing the lines but without the use of any data from the station

that we are predicting (i.e., the same $x(t)$ but no $y(t)$ is used as input). Furthermore, they are plotted at the x value where their RMSDs are closest to the corresponding RMSD from the NARX network or the linear model using the same data. That is, the position that a given open circle has on the x axis indicates the time frame over which the prior values at the station can be used to improve the prediction. There is a strong station dependence on this time frame. For the Stockholm, Landsort, Smögen, and Visby stations the prior $y(t)$ values improve the prediction for the whole 36-h period in the case of the NARX network, whereas those values only improve the predictions shorter than about 7 h at Kungsholmsfort.

Figure 4 shows the effect of increasing the time delay while keeping the same forcing as in Fig. 3. That is, more prior data from the input series are used in the 36-h prediction. Improvements of up to 15% can be seen at Smögen, which is the station with the largest tidal

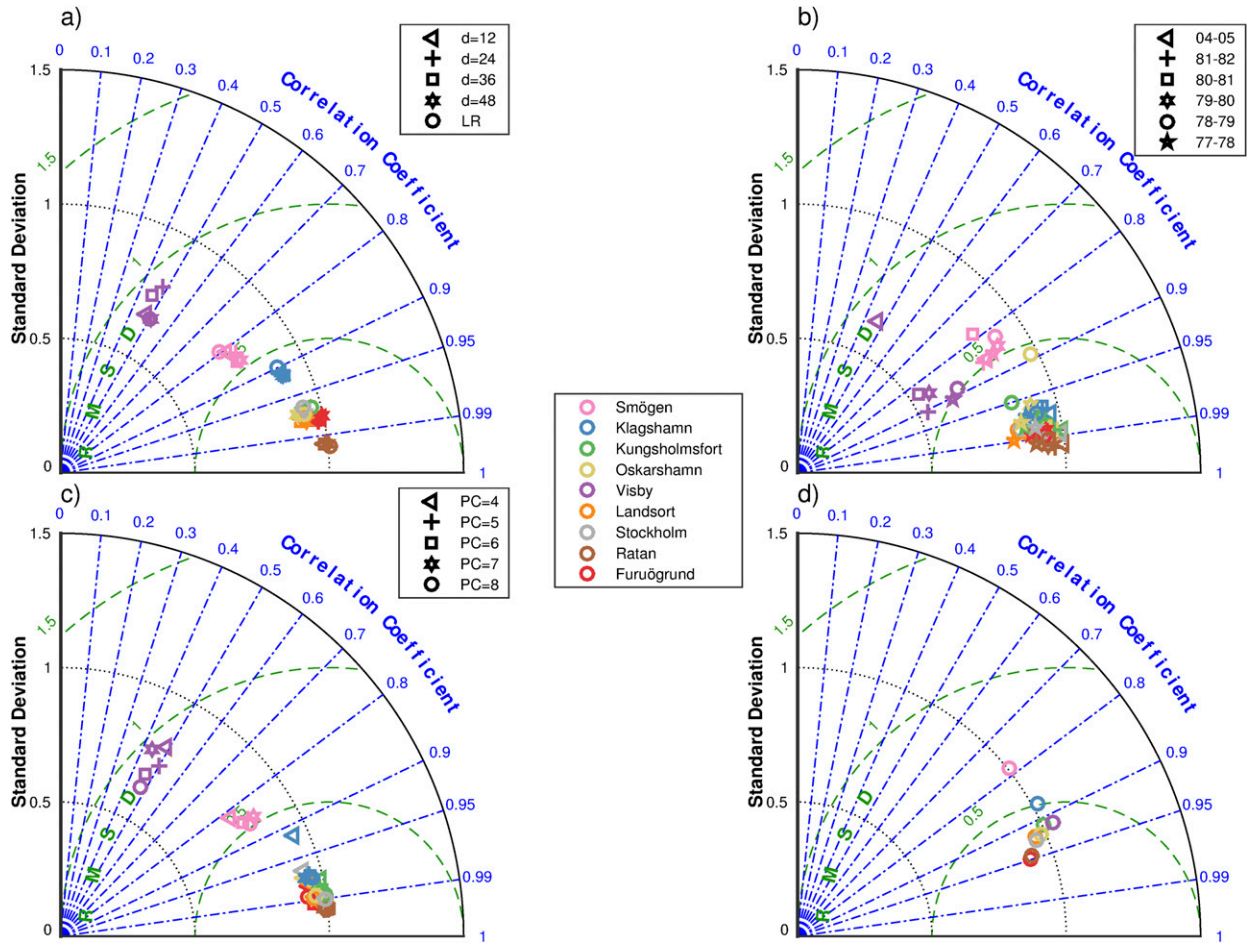


FIG. 5. Nondimensional Taylor diagrams showing the performance of the time delay network with eight hidden nodes, the linear model, and the barotropic version of NEMO-Nordic: (a) the time delay network at various delays and the linear model with a 12-h delay, (b) the time delay network at various time periods with a 12-h delay and eight PCs, (c) the time delay network with a 12-h delay and different numbers of PCs used, and (d) the barotropic version of NEMO-Nordic forced with the full ERA-40 forcing. Normalization is achieved through dividing both the RMSD and standard deviation of the models by those from the observations. The legend in (b) shows which years the data in the different test sets are from.

amplitudes. Successful tidal prediction using neural networks has also been demonstrated by Lee (2004) and Pashova and Popova (2011), and a longer time delay can be advantageous to better estimate the phase of the tide. However, little is gained with respect to the RMSD at any station by having a delay larger than 36 h, or with respect to correlations for delays larger than 24 h. These periods correspond well to the longest seiche and tidal periods of importance in the basins (Samuelsson and Stigebrandt 1996; Jönsson et al. 2008).

The performance of the time delay neural network with various delays and the corresponding linear model over the whole test period is shown in a nondimensional Taylor diagram in Fig. 5a. These models use the same external inputs as those in Fig. 4 but no feedback input.

That is, no sea level data from the local station are used as input. What we see is thus the performance of a neural network and a linear regression model in predicting a little more than a year of missing data from the respective stations. The performance difference between the two models is small, whereas the performance difference between stations is large. The performance for the Baltic stations with the exception of Visby, however, is extremely good. In fact, the machine learning algorithms outperform the numerical model NEMO-Nordic used by the Swedish Meteorological and Hydrological Institute at several stations on these metrics (Hieronymus et al. 2017, 2018; Hordoir et al. 2019).

NEMO-Nordic comes in two flavors: a full 3D version (Hordoir et al. 2019) and a barotropic version (Hieronymus et al. 2017, 2018). Figure 5c shows the performance of the

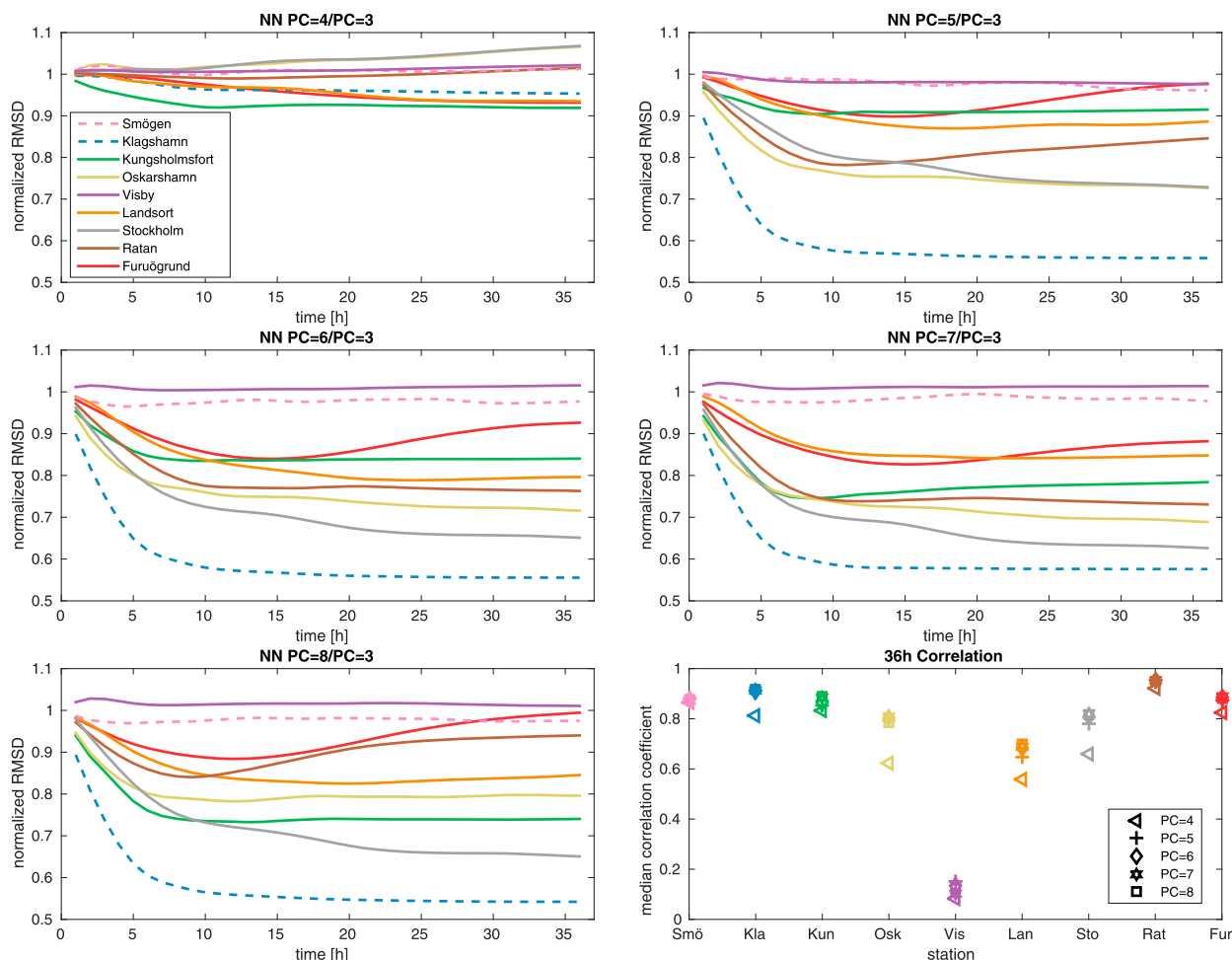


FIG. 6. Normalized RMSD and 36-h correlation coefficients for NARX projection with different numbers of PCs used as inputs. The RMSDs are normalized with those from the NARX network, with 3 PCs used as input. All networks have eight hidden nodes and a 12-h delay. Solid lines are stations in the Baltic, and dashed lines are outside the Baltic.

time delay neural network with different numbers of PCs used as forcing. This can be compared with the performance of the barotropic NEMO-Nordic over the whole test period and using the same ERA-40 forcing, which is shown in Fig. 5d. Note, however, that NEMO-Nordic is here forced with the full atmospheric forcing fields from ERA-40 and not just the PCs we use to force the ANNs. The SSH forcing from the other stations is, however, only used with the ANNs, because there is no data assimilation in this NEMO-Nordic version. The performance for the Baltic stations, with the notable exception of Visby, is excellent and, in fact, is superior to the result we get from the numerical ocean model. Another interesting point about the Baltic stations is that the regressions are, in general, performing better in the storm surge-prone northern Baltic than in its more calm central parts. Low-mode EOFs tend to have antinodes where

the variability is the strongest; our truncated EOF representations might therefore give better representations of the variability in more energetic areas and thus favor the northern over the central Baltic Sea stations. However, Ratan, which is situated south of Furuögrund and has weaker variability than Furuögrund, is the best-performing station. Performance is thus not simply related to the latitude of the station or the range of variability. The performance at the west coast station of Smögen is significantly worse than that at the Baltic stations both for NEMO-Nordic and the ANN. Here, the ANN has the upper hand when it comes to correlations, whereas NEMO-Nordic has a more realistic standard deviation. Overall, for this regression application the ANNs are clearly superior to the numerical model. They are also much cheaper to run and much more easy to improve on. The latter point stems from the fact that there are many more

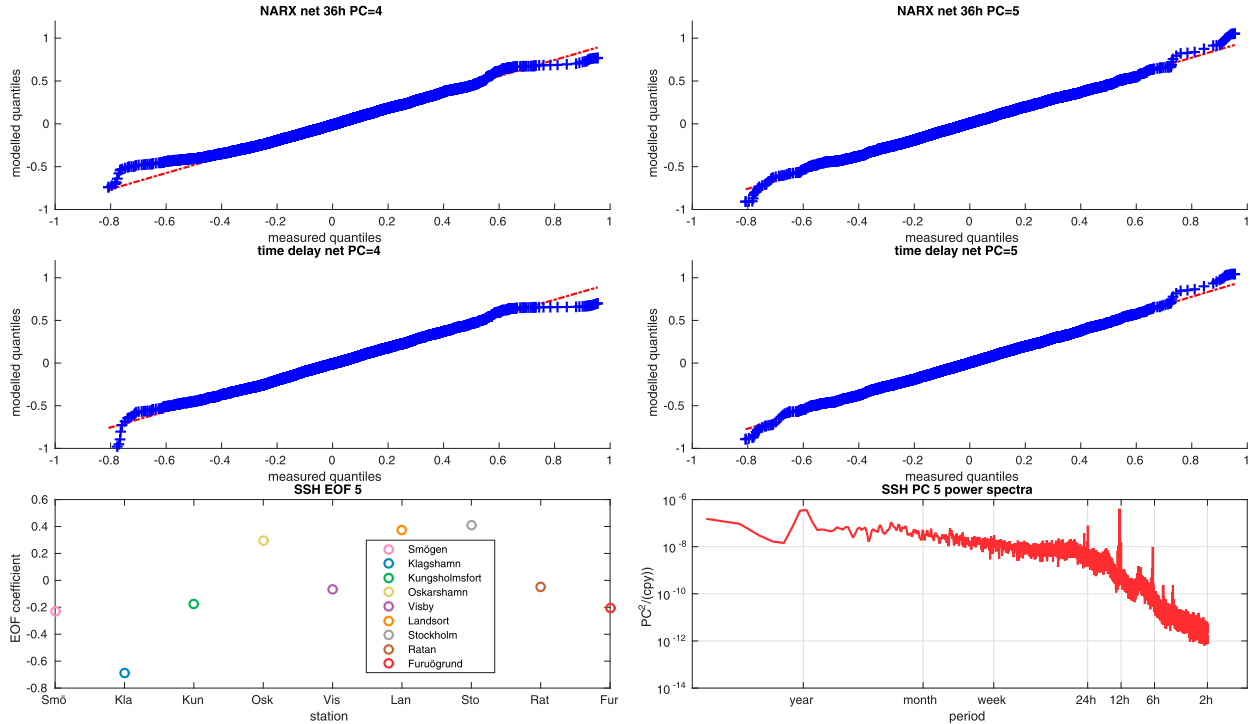


FIG. 7. Empirical quantile plots at the Klagshamn station, with four and five PCs used, for the NARX and time delay network, and the fifth SSH EOF pattern and the power spectrum of the fifth SSH PC. The power spectrum is computed using Welch's method. The quantiles shown are between 0 and 100 with a step of 0.005.

possible forcings that could be utilized than those that we have already included. The ANN performance on the west coast would, for example, likely be much closer to that in the Baltic Sea if we had used also sea level stations from Norway and Denmark.

To test the robustness of our regressions to different weather conditions, we used the unused training data (see section 2b) from the years before 1982 to create five additional test sets of length 10 000 h. The performance over these periods is shown in Fig. 5b. Overall, the performance is similar for different time periods. The largest exception is Visby, where the correlation is much better in these earlier test sets. The regressions also appear to have a slight tendency to underestimate the standard deviations over these earlier periods. We think this might be the result of having worse meteorological forcing because far fewer satellites were in operation in those days and as a consequence weather reanalyses were not as well constrained as they are today.

Figure 6 shows the effects of increasing the number of PCs used as external inputs by the NARX neural network while keeping the time delay fixed, as is done in Fig. 5c for the time delay network. The number of PCs used is the same for all forcing datasets. That is, we use the same number of PCs for meridional wind, zonal wind, surface pressure, and tide gauge data. This seems

to be a reasonable choice for a proof-of-concept study like this one. However, if we were instead interested in getting the very best result at a given station there are likely better options available. The p values reported later in this section for the different coefficients from the linear regression model could prove helpful for making an educated guess as to what data streams are most important also for the ANNs. However, some trial-and-error experiments would undoubtedly also be required given the fundamental differences between the linear model and the ANNs. Here we find a significant improvement of the result when the fifth PCs are included, whereas the improvements from including the subsequent PCs are more modest.

The good improvement from including the fifth PCs is particularly clear at the Klagshamn station both for the time delay and the NARX network. A more detailed view of the impact of adding the fifth PCs is given in Fig. 7, where empirical quantile plots are shown for the two networks using four or five PCs. In both cases we can see that when four PCs are used we tend to underestimate extremes on both the high and low end and that this bias is largely rectified when the fifth PCs are included in the forcing. It is, however, not possible to conclude from these experiments whether all of the fifth PCs are needed to get this improvement or whether, for

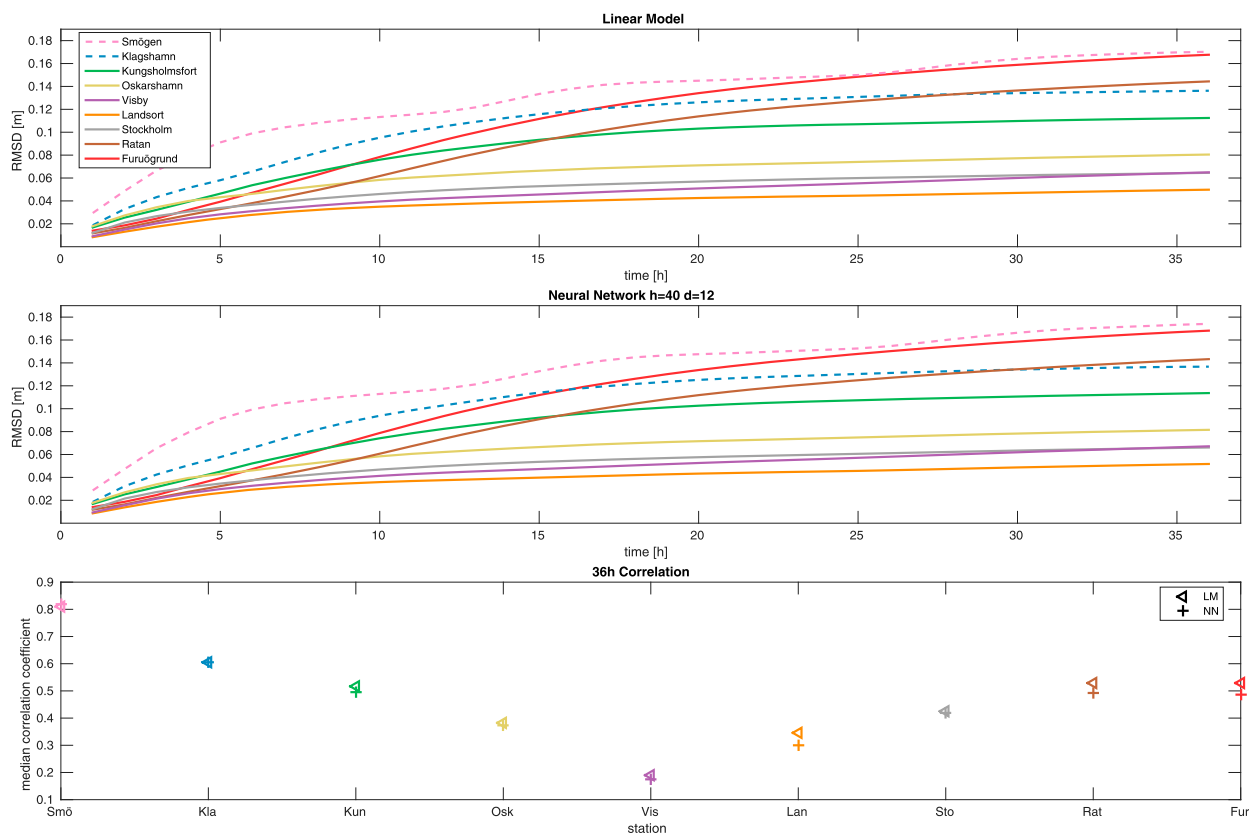


FIG. 8. Mean RMSD and median correlations over multiple 36-h forecasts using only past SSH data for the linear regression model and the NAR neural network. Solid lines are stations in the Baltic, and dashed lines are outside the Baltic. The d and h in the middle panel title indicate the delay used and the number of hidden nodes, respectively.

example, adding the fifth SSH PC is enough. However, the spatial structure of the fifth SSH PC also shown in Fig. 7 suggests that it, at least, plays an important part in this improvement. This EOF has antinodes in Klagshams, Oskarshamn, Stockholm, and Landsort, and all of these stations except Landsort are significantly improved when it is included. The Ratan and Furuögrund stations, which correspond to nodes in the fifth SSH EOF, are also significantly improved by the inclusion of the fifth PCs. This suggests that the fifth PCs from the meteorological forcing are likely to be important for these stations. The power spectrum of the fifth SSH PC is also shown in Fig. 7. Here we find peaks at the key tidal periods and at a period of 1 year. This EOF mode is thus unlikely to correspond to a single physical process, and its effect on sea level extremes is consequently unlikely to be directly related to a single missing process. In physical terms the mode represents water sloshing back and forth between the south and central Baltic Sea, and at least at Klagshamn it seems that this sloshing motion is in phase with the sea level extremes at the station.

To evaluate the effects of different parts of the forcing we have also done experiments using only tide gauge

forcing or only meteorological forcing. Figure 8 shows the mean RMSDs and median correlation coefficients from 36-h forecasts with a 12-h time delay like in Fig. 3 but only using the tide gauge data as forcing. The ANN used here is thus the NAR neural network, and the linear model is constructed so that it uses the same data as the ANN. The performance of the linear model and the ANN is very similar here, which indicates that the generalizable information in the sea level forcing dataset is mostly linear in nature. The quality of this forecast, in which only the tide gauge data are used, is very similar to that of the linear model in Fig. 3 in which also the meteorological forcing is used. The large quality difference between the ANN and the linear model shown in Fig. 3 therefore suggests that the ANN's superiority in that experiment is primarily owing to a better utilization of meteorological forcing. Figure 9 shows which coefficients in the linear model in Fig. 3 are significantly different from 0 at the 0.05 level and corroborates the picture that the linear model does not make great use of meteorological forcing. Here we find significant coefficients at nearly all time delays for the sea level at the forecast station and typically a fair amount of significant

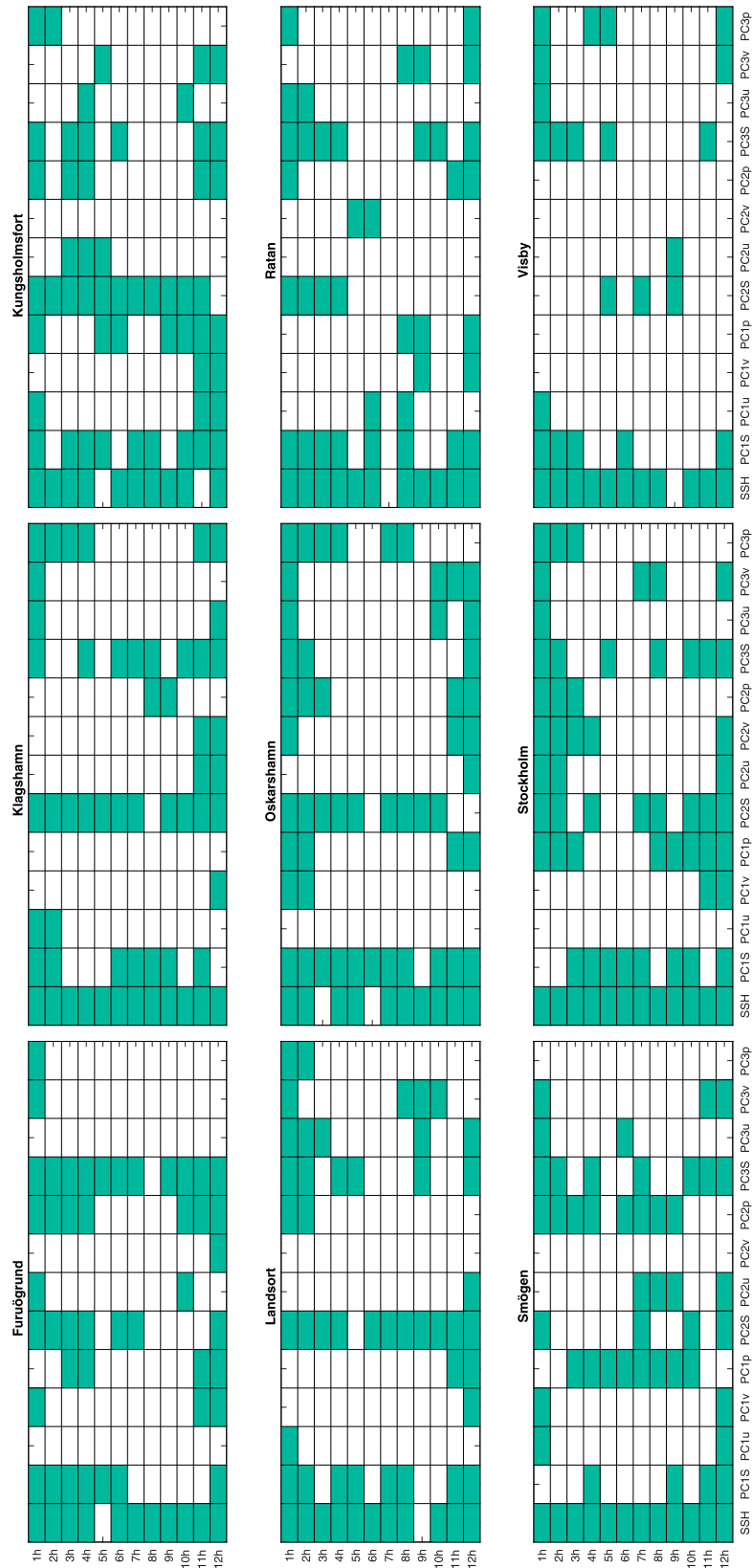


FIG. 9. Significance test for the coefficients of the linear model used in Fig. 3. Coefficients marked as green are significant at the 0.05 level using the null hypothesis that the coefficient is equal to zero. The y axis shows the time delay, and the x axis is the datatype. Here, SSH is the SSH data from the same station, PC1S is the first PC of SSH, PC1u is the first PC of zonal wind, PC1v is the first PC of meridional wind, and PC1p is the first PC of sea surface pressure. PC2 and PC3 are the second and third PCs.

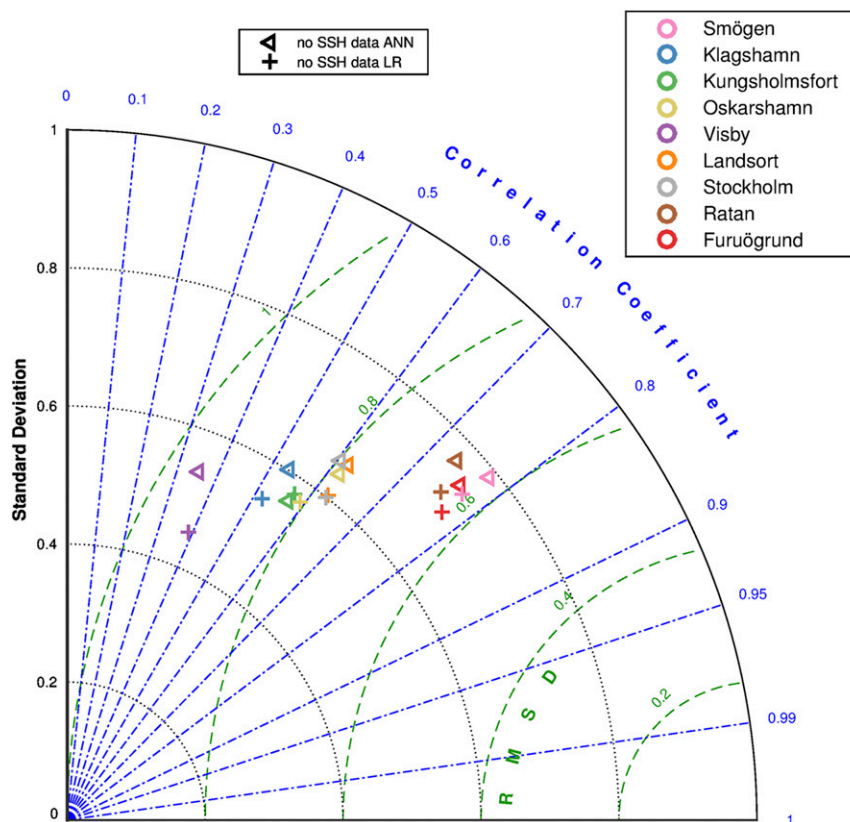


FIG. 10. Nondimensional Taylor diagram showing the performance of a time delay network with eight hidden nodes forced with just atmospheric forcing. Normalization is the same as in Fig. 5.

coefficients for the first three sea level PCs, whereas significant coefficients are more sparse for the meteorological forcing.

Figure 10 shows a nondimensional Taylor diagram of the performance of a linear model and the time delay neural network over the whole test period. Both models are forced with the first five PCs, but only from the atmospheric forcing set. The forecasts here are thus done without any knowledge of the state of the ocean and are therefore indicative of the performance we can get for boundary data to use for future dynamical downscalings of ocean data forced with already downscaled atmospheric climate scenarios. The performance is, of course, much worse than when ocean data are also present, but many of the stations have a decent correlation with observations, and if comparable quality can be achieved at the open boundary of our numerical ocean model it would very likely be a superior boundary forcing to today's practice, where either zeros or monthly means are used as boundary forcing in climate scenario integrations. A performance difference between the linear model and the ANN is mostly seen in the standard deviation, which is better for the ANN. Moreover, it is

interesting to note that the linear model is nearly as good as the neural network when using only SSH forcing, when using only atmospheric forcing, and when using remote SSH forcing and atmospheric forcing. The main nonlinearity in this regression problem, or at least the nonlinearity that can be used by the ANNs to improve the predictions, is thus introduced when the local SSH forcing is combined with the other forcings.

4. Conclusions

Artificial neural networks and multiple linear regression models were tested on several different regression problems one can encounter when using sea level data from tide gauge stations. The regression problems differed in availability of data, and the performance difference between the ANNs and the linear models was very problem dependent. The ANN proved vastly superior to the linear model when local SSH data, remote SSH data, and meteorological forcing were all used (i.e., when the NARX network was used), whereas the performance differences were much smaller in the other cases. Our interpretation of this result is that the

main nonlinearity that the ANNs were able to use to improve the predictions was introduced when local sea level forcing was combined with the other forcings.

The sensitivity to the length of the time delay and number of PCs used as external input was also evaluated. For the time delay length we found little improvement from increasing the delay to longer than 36 h; 36 h is slightly longer than the most important seiche and tidal periods in the North and Baltic Seas, so this is a number that makes sense from a physical perspective. An alternative approach to determine the best time delay would be to look at the autocorrelation function for SSH. However, the autocorrelation is very different for different stations. In particular, it is very strong on long lag times for the central Baltic stations, which makes it difficult to find a single cutoff to use for all stations. For the number of PCs used, we found a large improvement when the fifth PC was included and more modest improvements for higher PCs. For stations in the south and central Baltic, we found the improvement to be related to the fifth SSH PC, whereas the meteorological PCs appeared to be more important for the northern Baltic stations.

The overall performance of the machine learning algorithms on sea level prediction was very impressive. For the metrics discussed here, the performance of these algorithms rivals that of the much more computationally expensive numerical ocean models on the Swedish west coast and exceeds it in the Baltic Sea. Moreover, the large data availability from more tide gauges and better meteorological forcing suggests that further improvements may be achieved without much difficulty. The performance at the troublesome Swedish west coast, for example, would likely improve a lot if more tide gauge station data were used, including such from Norway and Denmark.

The reason for the performance difference between the west coast and the Baltic Sea is twofold, as we discussed in section 2a, owing both to differences in the ocean environments and to differences in the density of available tide gauge stations. Sea level regressions in different ocean environments will generally require tide gauge networks with different densities to perform well. It is thus not possible to formulate a general recommendation for the density needed to achieve a given performance in some ocean environment on the basis of the findings in this article. However, an ocean model could relatively easily be set up to test different placements to find an, in some sense, optimal observations network that could be installed at a given location.

The exceptionally good performance found when local sea level data were used with the NARX network opens up the possibility to use the network both for

interpolations of missing data and as a quality check for the data in the records. The great performance of the time delay network, especially in the Baltic, shows that this network can be used to stretch records from existing stations back in time with an accuracy exceeding that of our numerical models. The performance of the time delay network when only the atmospheric forcing is used, was very good for some stations and less good for others. An avenue of future development, which will be reported on elsewhere, is our intended application of this network to create boundary data for dynamic downscalings of ocean data from already downscaled atmospheric data.

Acknowledgments. The sea level data used in this paper are available online (<https://www.smhi.se/en/services/open-data/oceanographic-observations-1.33356>). We also thank two anonymous referees for their helpful comments.

REFERENCES

- Al-Jarrah, O. Y., P. D. Yoo, S. Muhaidat, G. K. Karagiannidis, and K. Taha, 2015: Efficient machine learning for big data: A review. *Big Data Res.*, **2**, 87–93, <https://doi.org/10.1016/j.bdr.2015.04.001>.
- Andersson, H. C., 2002: Influence of long-term regional and large-scale atmospheric circulation on the Baltic sea level. *Tellus*, **54A**, 76–88, <https://doi.org/10.1034/j.1600-0870.2002.00288.x>.
- Berg, P., R. Döschner, and T. Koenigk, 2013: Impacts of using spectral nudging on regional climate model RCA4 simulations of the Arctic. *Geosci. Model Dev.*, **6**, 849–859, <https://doi.org/10.5194/gmd-6-849-2013>.
- Church, J., and Coauthors, 2013: Sea level change. *Climate Change 2013: The Physical Science Basis*, T. F. Stocker et al., Eds., Cambridge University Press, 1137–1141.
- DeConto, R. M., and D. Pollard, 2016: Contribution of Antarctica to past and future sea-level rise. *Nature*, **531**, 591–597, <https://doi.org/10.1038/nature17145>.
- Ekman, M., 1999: Climate changes detected through the world's longest sea level series. *Global Planet. Change*, **21**, 215–224, [https://doi.org/10.1016/S0921-8181\(99\)00045-4](https://doi.org/10.1016/S0921-8181(99)00045-4).
- Hannachi, A., I. T. Jolliffe, and D. B. Stephenson, 2007: Empirical orthogonal functions and related techniques in atmospheric science: A review. *Int. J. Climatol.*, **27**, 1119–1152, <https://doi.org/10.1002/joc.1499>.
- Hieronymus, M., 2019: An update on the thermosteric sea level rise commitment to global warming. *Environ. Res. Lett.*, **14**, 054018, <https://doi.org/10.1088/1748-9326/ab1c31>.
- , J. Hieronymus, and L. Arneborg, 2017: Sea level modelling in the Baltic and the North Sea: The respective role of different parts of the forcing. *Ocean Modell.*, **118**, 59–72, <https://doi.org/10.1016/j.ocemod.2017.08.007>.
- , C. Dieterich, H. Andersson, and R. Hordoir, 2018: The effects of mean sea level rise and strengthened winds on extreme sea levels in the Baltic Sea. *Theor. Appl. Mech. Lett.*, **8**, 366–371, <https://doi.org/10.1016/j.taml.2018.06.008>.
- Hordoir, R., and Coauthors, 2019: NEMO-Nordic 1.0: A NEMO based ocean model for Baltic and North Seas—Research and operational applications. *Geosci. Model Dev.*, **12**, 363–389, <https://doi.org/10.5194/gmd-12-363-2019>.

- Jönsson, B., K. Döös, J. Nycander, and P. Lundberg, 2008: Standing waves in the Gulf of Finland and their relationship to the basin-wide Baltic seiches. *J. Geophys. Res.*, **113**, C03004, <https://doi.org/10.1029/2008JG000688>.
- Kulikov, E. A., and I. P. Medvedev, 2013: Variability of the Baltic sea level and floods in the Gulf of Finland. *Oceanology*, **53**, 145–151, <https://doi.org/10.1134/S0001437013020094>.
- Leahy, T. P., F. P. Llopi, M. D. Palmer, and N. H. Robinson, 2018: Using neural networks to correct historical climate observations. *J. Atmos. Oceanic Technol.*, **35**, 2053–2059, <https://doi.org/10.1175/JTECH-D-18-0012.1>.
- Lee, T.-L., 2004: Back-propagation neural network for long-term tidal predictions. *Ocean Eng.*, **31**, 225–238, [https://doi.org/10.1016/S0029-8018\(03\)00115-X](https://doi.org/10.1016/S0029-8018(03)00115-X).
- Levermann, A., P. U. Clark, B. Marzeion, G. A. Milne, D. Pollard, V. Radic, and A. Robinson, 2013: The multimillennial sea-level commitment of global warming. *Proc. Natl. Acad. Sci. USA*, **110**, 13 745–13 750, <https://doi.org/10.1073/pnas.1219414110>.
- Melet, A., B. Meyssignac, R. Almar, and G. L. Cozannet, 2018: Under-estimated wave contribution to coastal sea-level rise. *Nat. Climate Change*, **8**, 234–239, <https://doi.org/10.1038/s41558-018-0088-y>.
- Mitrovica, J. X., C. C. Hay, R. E. Kopp, C. Harig, and K. Latychev, 2018: Quantifying the sensitivity of sea level change in coastal localities to the geometry of polar ice mass flux. *J. Climate*, **31**, 3701–3708, <https://doi.org/10.1175/JCLI-D-17-0465.1>.
- Monahan, A. H., J. C. Fyfe, M. H. P. Ambaum, D. B. Stephenson, and G. R. North, 2009: Empirical orthogonal functions: The medium is the message. *J. Climate*, **22**, 6501–6514, <https://doi.org/10.1175/2009JCLI3062.1>.
- Pashova, L., and S. Popova, 2011: Daily sea level forecast at tide gauge Burgas, Bulgaria using artificial neural networks. *J. Sea Res.*, **66**, 154–161, <https://doi.org/10.1016/j.seares.2011.05.012>.
- Samuelsson, M., and A. Stigebrandt, 1996: Main characteristics of the long-term sea level variability in the Baltic Sea. *Tellus*, **48A**, 672–683, <https://doi.org/10.3402/tellusa.v48i5.12165>.
- Samuelsson, P., and Coauthors, 2011: The Rossby Centre regional climate model RCA3: Model description and performance. *Tellus*, **63A**, 4–23, <https://doi.org/10.1111/j.1600-0870.2010.00478.x>.
- Uppala, S. M., and Coauthors, 2005: The ERA-40 Re-Analysis. *Quart. J. Roy. Meteor. Soc.*, **131**, 2961–3012, <https://doi.org/10.1256/qj.04.176>.
- van Maanen, B., G. Coco, K. R. Bryan, and B. G. Ruessink, 2010: The use of artificial neural networks to analyze and predict alongshore sediment transport. *Nonlinear Processes Geophys.*, **17**, 395–404, <https://doi.org/10.5194/npg-17-395-2010>.
- Vousdoukas, M. I., L. Mentaschi, E. Voukouvalas, M. Verlaan, and L. Feyen, 2017: Extreme sea levels on the rise along Europe's coasts. *Earth's Future*, **5**, 304–323, <https://doi.org/10.1002/2016EF000505>.
- , —, —, A. Bianchi, F. Dottori, and L. Feyen, 2018: Climatic and socioeconomic controls of future coastal flood risk in Europe. *Nat. Climate Change*, **8**, 776–780, <https://doi.org/10.1038/s41558-018-0260-4>.
- Zijl, F., M. Verlaan, and H. Gerritsen, 2013: Improved water-level forecasting for the northwest European shelf and North Sea through direct modelling of tide, surge and non-linear interaction. *Ocean Dyn.*, **63**, 823–847, <https://doi.org/10.1007/s10236-013-0624-2>.