Author response to reviews of manuscript

DELWAVE 1.0: Deep-learning surrogate model of surface wave climate in the Adriatic Basin

Peter Mlakar[1,6], Antonio Ricchi[2,3], Sandro Carniel[4], Davide Bonaldo[5,‡], and Matjaž Ličer[1,7,‡]

This review was submitted as a series of comments in the manuscript pdf document. We list our response below by stating roughly where the comment was placed and what the comment said. **Referee comments are in bold face blue**, while our response is in normal face and is marked as AR (authors response).

Response to Review #1

This is an interesting paper and the results are very promising. There are a few things that need improvement though, for instance parts of the manuscript should have more mathematical rigor instead of attempting to describe in words certain formulations. A more detailed description on computational times should also be included, since the focus is almost entirely on the comparison between numerical predictions of SWAN and DELWAVE. This is certainly important, but your initial argument was that DELWAVE was a mean to save computational time, and I did not find a convincing argument as to whether you have shown this in your paper. Please find in the attached document a complete list of minor and major points that I would like the authors to address before I can reconsider this manuscript for publication.

AR: We thank the reviewer for taking the time to review our paper and we address their specific remarks below.

Page 1:

• **the abstract is too long and should considerably be shortened just highlighting the major contributions of the paper and sparing the technical details for the body.**

AR: The abstract has been shortened and technical details were removed.

Page 2:

• On the statement 'Deep learning has shown a great potential to address these issues without hindering performance' focusing on 'without hindering performance'.

Referee comment: maybe this is a too strong statement put outside of context

AR: We thank the reviewer for pointing this out. We have toned down this statement and added further references to provide more context. This passage now reads:

> Deep learning has been shown to promise great potential to address these issues across multiple fields of science, including machine vision, natural language processing, and, more recently, in various subfields of meteorology (Janssens and Hulshoff, 2022; Beucler et al., 2021; Rasp et al., 2018) and oceanography (Rus et al., 2023; Sonnewald et al., 2021; Boehme and Rosso, 2021; Žust et al., 2021; Mallett et al., 2018). With particular reference to wave dynamics applications, James et al. (2018) proposed a machine learning system for predicting the steady-state response of the sea state in a coastal area to a given wind configuration, whereas Rodriguez-Delgado and Bergillos (2021) developed a framework for propagating onshore the open-sea information on incoming waves for renewable energy production purposes. In specific cases and for specific tasks deep

Page 3:

• On the statement 'In this paper we present a newly developed deep learning method, named DELWAVE, for emulating, at a computational price smaller by several orders of magnitude' focusing on 'at computational price smaller'.

Referee comment: compared to what?

AR: We thank the reviewer for pointing this out. The object of reference in this case is the SWAN model. We have made this explicit in this sentence.

> In this paper we present a newly developed deep learning method, named DELWAVE, for emulating non-stationary modelled surface sea states, such as those produced by SWAN albeit at a computational price smaller by several orders of magnitude, in response to given wind fields. The study site is the Adriatic Sea, a 200 km wide and 800 km long elongated epicontinental basin

Page 7:

• On the statement 'are given in terms of significant wave height (HS), mean wave direction (d), and energy period (Tm−1,0)'.

Referee comment: you should define these mathematically.

• Entire 3.1.1 subsection. **Referee comment: could this be written in math instead of in words?**

AR: We will define these quantities in our response, but since the background of wave modeling is beyond the scope of the paper, we refer the readers of the manuscript to the SWAN manual. For reviewer, we would like to clarify that the mentioned quantities are recalled here in their conventional definitions as $H_S = 4\sqrt{\int\int E(\omega,\theta)\, d\omega d\theta}$,

$d = \frac{180}{\pi} \arctan\left(\frac{\int \sin\theta E(\sigma,\theta)\, d\sigma d\theta}{\int \cos\theta E(\sigma,\theta)\, d\sigma d\theta} \right)$, and $T_{m-1,0} = 2\pi \frac{\int\int \omega^{-1} E(\omega,\theta)\, d\omega d\theta}{\int\int E(\omega,\theta)\, d\omega d\theta}$, where E is the wave directional spectrum, as reported in the SWAN user manual. As noted, in order to avoid overburdening the text, we did better clarify what we are referring to, but referring to the manual for the mathematical definitions. We hope this addresses the reviewer's question.

• Talking about location encoding in subsection 3.1.2.

Referee comment: What exactly do you mean with the word "encoding"? and Could you write this in math instead of in words?

AR: We have rewritten the entire 3.1 section in terms of equations describing the quantities in question. Most explicitly, section 3.1.2 now looks like this. We hope this answers reviewers' concerns.

3.1.2 Location encoding and grid encoding

We further complement the wind field input tensor $\mathbf{I}^t$ by a spatial encoding matrix. The purpose of this matrix is to provide the network with information about the specific location for which we wish to predict surface wave attributes. This approach allows us to easily add new locations into the training procedure by simply defining new spatial encoding matrices, without the need for any other modifications to the algorithm or model architecture.

Let $\mathbf{L}_l$ denote the location encoding sparse matrix for location l. Then

$$\mathbf{L}_l \in \mathbb{R}^{n_x \times n_y}, \quad \dim(\mathbf{L}_l) = [n_x, n_y]. \tag{3}$$

We now denote each ith row ($i = 1, \ldots, n_x$) and jth column ($j = 1, \ldots, n_y$) entry of $\mathbf{L}_l$ as $\mathbf{L}_{l,(i,j)}$ and compute the matrix entries as

$$\mathbf{L}_{l,(i,j)} = \frac{1}{\sqrt{2\pi}\varsigma} \exp\left[-\frac{1}{2} \frac{(l_i - i + 1)^2 + (l_j - j + 1)^2}{\varsigma^2} \right], \tag{4}$$

where we set the spatial variance to $\varsigma^2 = 20$. This variance corresponds to a standard deviation of $\sqrt{20} \sim 4 - 5$ grid cells or $0.45°$ in longitude and latitude, as shown on Figure 3. We determined the value of the spatial variance empirically, by testing multiple value configurations where we finally selected the spatial variance value which produced the best results. The variables l_i and l_j denote the corresponding l location's position in the spatial field, expressed in terms of row and column indices. We illustrate examples of multiple encodings for different locations in Figure 3.

Finally, we normalize the matrix entries to the range $[0, 1]$ by

$$\tilde{\mathbf{L}}_{l,(i,j)} = \frac{\mathbf{L}_{l,(i,j)} - \min(\mathbf{L}_l)}{\max(\mathbf{L}_l) - \min(\mathbf{L}_l)}. \tag{5}$$

We use this normalized location encoding matrix to augment the input wind field tensor to form the wind-location input tensor $\mathbf{I}_l^t$, where the tensor is now given for a specific location target and time. Here, the augmentation denotes the concatenation of the starting input tensor and the location encoding along the first dimension. This entails that $\dim(\mathbf{I}_l^t) = [3, n_x, n_y]$, where the increased size of the first dimension corresponds to this augmentation. To create training samples for all k locations, based on the same wind field, we use the following approach: we first randomly sample a wind field from the dataset. Then we augment the wind field with the k location matrices, where each individual augmentation produces its own $\mathbf{I}_l^t$ corresponding to a location l. This way, each training sample contains the wind field, together with a spatial encoding of a specific location. As we train the model, the training takes into account all different locations and all time steps during the same training process.

This input provides the necessary information for the model to distinguish between the different locations for which we require surface wave predictions. Without this encoding the model would most likely gravitate towards an average prediction at a specific time t for all locations, as it would not be able to distinguish between them. During DELWAVE training, we minimize the root mean squared differences loss function defined as

$$\mathcal{L} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}, \tag{6}$$

where y_i denotes the SWAN values for sample i, and $\hat{y}_i$ DELWAVE's predictions. If we were to omit the location encoding from the input tensor for time t, then each location would share the same input tensor at time t (the location encoding is what differentiates input tensors for each target location), however the wave field attributes of each location are not the same. Therefore, the average prediction of all target locations is the minimizer in this case.

The final transformation of the input tensor is the concatenation of the grid encoding. A building block of DELWAVE's architecture is the convolution operation, which is, by design, translation invariant. This implies that same signal at different spatial locations produces the same output response. Since the location of specific wind patterns with relation to the target location of interest is important (wind fetch), we have to go against this inherent invariability of the convolution operation to translations. We do this using the grid encoding which assigns a unique value to each spatial location inside the input field. This enables the network to learn wind features in specific regions of the input. We denote the grid encoding matrix as $\mathbf{C} \in \mathbb{R}^{n_x \times n_y}$. Individual entries of the matrix are computed as

$$\mathbf{C}_{(i,j)} = \frac{(i-1)n_y + j - 1}{n_x n_y}, \tag{7}$$

where i is the index to the row and j to the column. We augment the wind-location tensor with the above defined grid matrix to produce the final wind-location tensor (we do not explicitly denote the grid encoding presence inside the tensor) in the same way as we did in the case of the location encoding. We end up with a tensor containing 4 input fields (zonal wind, meridional wind, location encoding, grid encoding) of dimension $[n_x, n_y]$:

$$\dim(\mathbf{I}_l^t) = [4, n_x, n_y]. \tag{8}$$

Furthermore, we have created a new spatial encoding matrix Figure and added further description of spatial encoding to make this part of text clearer.

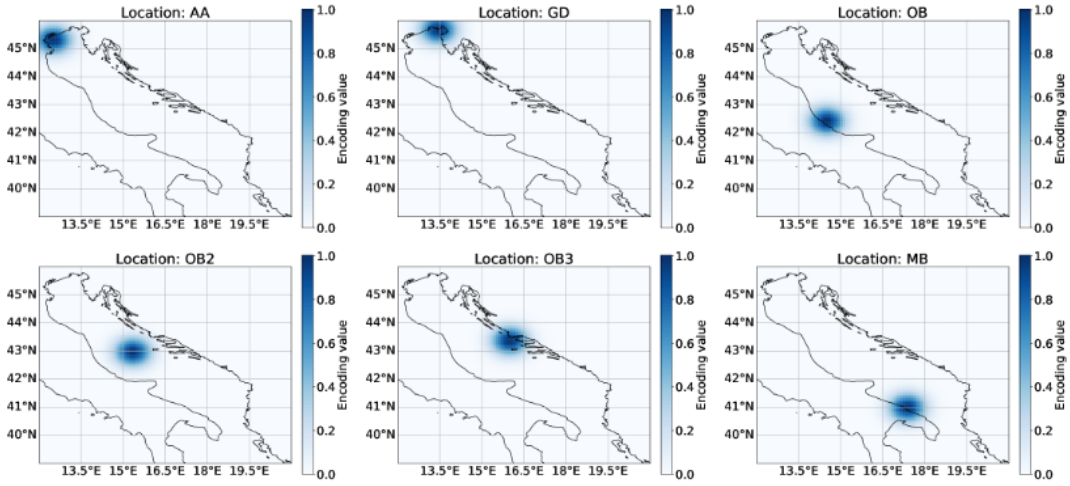


Figure 3. The visualization of the spatial encoding matrices for each location (the coastline is added for clarity). Each plot corresponds to one location encoding matrix which forms a part of the input sample tensor $\mathbf{I}_i$.

Page 8:

- Talking about spatial encoding.

Referee comment: you are using this word a lot without having clearly explained what it means in your context.

AR: We provided a more thorough explanation of this concept in the previous response and corresponding explanation.

- Talking about section 3.1:

Referee comment: you are using the word 'added' but not in its mathematical sense. This part (3.1) should be revised, making it more mathematically sound and using math instead of writing everything in words.

AR: We agree with the reviewer that the term "added" is ambiguous in this context and does not properly reflect the actual operation performed. We have replaced this term with "concatenated" and have explained how this operation affects the input tensor. The remainder of section 3.1 was rewritten in mathematical notation.

• We mention how, if we do not include the Gaussian location encoding, the network has no information as for which location it is predicting the wave attributes. It would make sense then, that the prediction it forms should converge towards the average value for all stations.
Referee comment: do you have any explanation as to why this was happening?

AR: We thank the reviewer for pointing this out. We have provided a description of the intuition behind the reason for this convergence behavior. We have also softened the strength of this claim as we conducted only a handful of tests on this behavior. Indeed, the argument is more theoretical in the sense that, if there is no information (location encoding) to distinguish between target locations given an input at time t, then, minimizing the mean of squared differences, the minimizer should be the average prediction for all locations. We included further explanation in the text as follows:

> This input provides the necessary information for the model to distinguish between the different locations for which we require surface wave predictions. Without this encoding the model would most likely gravitate towards an average prediction
> 200 at a specific time t for all locations, as it would not be able to distinguish between them. During DELWAVE training, we minimize the loss function in the form of the mean of squared differences. If we were to omit the location encoding from the input tensor for time t, then each location would share the same input tensor at time t (the location encoding is what differentiates input tensors for each target location), however the wave attributes of each station are not the same. Therefore, the average prediction of all target locations is the minimizer in this case.

Page 10:
• **The referee would appreciate a more robust definition of 'shallow network', encoder being 'shared between timesteps', 'latent', and 'convolution filter with a kernel size of one', the later in a mathematical notation.**

AR: We have thoroughly rewritten relevant sections (3.1 and 3.2) to address reviewers concerns. We have added definitions for the terms "shallow network", "encoder", and the technique of the encoder being "shared between timesteps". Where appropriate, we expanded on the above terms using mathematical notation to render the concept clearer. In the context of the convolution with kernel size one we decided to augment this section with a reference to the exact implementation of this operation expressed in mathematical terms. Additionally, we corrected the temporal encoding figure in this section which we found contained some errors.
The term latent vector or latent encoding signifies a vector that is formed by passing an input in from an observable, interpretable space (for example spatial wind fields) through the transformation which maps this input onto a high dimensional manifold. Since the latter mapping is obtuse, described by the neural network transformation

(atmospheric encoder) we say that the resulting vector exists in latent (hidden) space. This notion is similar to that of an encoding and is frequently used in machine learning. Therefore, we have removed this term for the sake of clarity and replaced it with appropriate equivalents.

Pages 11 and page 12:

• **The referee would appreciate a more robust definition of dropout and a reason as to why target data in the training dataset is standardized.**

AR: We will answer both questions here extensively and we also include short explanations of referees questions in the manuscript.
Dropout is a regularization technique used in machine learning, particularly in the training of deep neural networks. The core idea behind dropout is to randomly "drop out" or deactivate a subset of neurons (i.e., units within the network) during each step or iteration of the training process, whilst scaling the remaining neurons proportionally to the dropout probability. Dropout helps in preventing the model from overfitting to the training data. Overfitting happens when a model learns not only the underlying patterns in the training data but also the noise and specific details, which can hinder its performance on unseen data. By dropping out random neurons, the network becomes less sensitive to specific weights as this operation prevents neuron coadaptation, leading to a more generalized model.

We have now included these reasons for dropout and modified the passage that the referee points to in the following fashion:

3.2.3 Regression block

Finally, the regression block, displayed in Figure 7, comprises of consecutive fully connected layers with skip connections. This block produces the final outputs: MWP, SWH, and MWD. To prevent overfitting and improve performance over unseen data in the cross-validation dataset, a dropout with a removal probability of 0.2 is applied between each fully connected layer, except the last two (the output layer and the penultimate layer).

Standardization or normalization of the training set in machine learning is a crucial preprocessing step, particularly for algorithms that are sensitive to the scale of the input features. Standardization involves rescaling the features so that they have a mean of 0 and a standard deviation of 1. This process is used often in classical statistics and has several important benefits. For example, it enables comparisons of impacts of different features, measured on different scales (e.g., one feature in m/s, another in degrees Celsius or millibars). Algorithms that compute distances between data points can be

biased towards features with larger scales. Standardization ensures that each feature contributes equally to these distance calculations. Standardization may also improve convergence in such neural networks as DELWAVE, which employ gradient descent algorithms.

• Referencing the input data distributions for wave attributes, their skewed nature and our attempt at re-sampling.

Referee comment: but if that is how the input data looks like, why is it OK to change it?

AR: We thank the reviewer for pointing out this unclarity. We will attempt to clarify this and modify the manuscript accordingly. Resampling does not mean that we changed the input data in any way - what we did do is change the sampling strategy of the samples (from the unchanged input data) that DELWAVE was trained on. The reason for this is that neural networks often have problems at predicting extreme events in the tails of the distributions, because these events are, by definition, rare, and as such exert very limited influence during training. Therefore, if we want the network to model the tails of the distributions successfully, we need to help it get better acquainted with extreme events in these tails. We did this by oversampling extreme events in the training set: DELWAVE therefore encountered extreme events more often during training, and adjusted its response so that the predictions of extreme events got better. As can be seen from the histograms, this does not mean that DELWAVE performance dropped during normal conditions - oversampling merely enabled DELWAVE to achieve better performance during storms.

Related paragraph was reformulated to, hopefully, be clearer in this regard:

Neural networks often have difficulties predicting extreme events in the tails of the distributions, because these events are by definition rare and the network rarely encounters them in the training set. To better learn and model underrepresented values of the target variables we increased their presence in the training set by employing the so-called random importance-sampling. We illustrate random importance-sampling in Figure 8. If we observe the distributions of the three target variables in a randomly sampled batch we can see that these are skewed. For example, in Figure 8, the significant wave height is distributed similarly to a left slanted gamma-like distribution with a very long tail. Therefore, the model is not exposed to the tail of the distribution frequently which inhibits efficient training in that part of the distribution. This results in systematic errors, where the regression accuracy for significant wave height drops with increasing height. This is understandable as samples with wave heights above two meters constitute only a small fraction of the dataset.

Page 13:

• Possibly referencing the re-sampling of the input dataset and the two-part training procedure.

Referee comment: how long does it take to perform this 'adjustment' of the input sample for the training? I think the overall argument on the convenience of using deep learning should include a description of how long it takes to 'fix up' the input data set to have the model trained to an extent that makes it useful.

AR: We need to stress we do not manipulate the input data, we merely use a sampling strategy that overrepresents input samples for stormy conditions, which would otherwise be too rare for successful training. This importance resampling technique is quick to execute and is done on-the-fly as the model is training. Therefore, no additional preprocessing of the dataset is required. Importance resampling is done as each training batch is generated and usually takes around a second to complete.

• Talking about the temporal ablation study.

Referee comment: why are you not doing a study on the effect of the number of spatial grid points?

AR: We thank the reviewer for pointing out this potential ablation study. Indeed, it would be interesting to see how successful the neural network would be if the resolution of the input field would be reduced. However, due to the nature of our approach using a location encoding, this would be a more complicated task compared to the temporal ablation. Since the location encoding directly specifies the location for which we are predicting wave attributes, the meaning of this location encoding in a reduced resolution wind field becomes less clear. It is most likely that we would have to employ a different system altogether which would allow this. We believe that this is an interesting avenue for future research as it would allow us to empirically test spatial information richness of the wind field and possible redundancies.

• On the statement 'where the subscript denotes the number of used timesteps'.

Referee comment: is the number X here obtained as 1+ previous timesteps necessary? So in your argument before X was 11?

AR: Yes, this is exactly correct. For example, in the case of temporal ablation, DELWAVE_8 denotes 7 prior timesteps plus the current timestep. We have added a more clear explanation of this in the manuscript:

timesteps. Here, each of the five variants uses $n+1$ timestemps, where n denotes the number of previous timesteps (in case of DELWAVE$_8$ this means seven previous timesteps with the addition of the current one). Results of this study are presented in Table 1 and their validation loss during training in Figure 9.

Page 14:

• **Referee comment: What are the units for 'epochs' on the x axis in Figure 9?**

AR: In the context of the machine learning training process, an "epoch" refers to one complete training cycle through the entire training dataset. During a training epoch, the learning algorithm processes each sample in the dataset, using it to adjust the model's parameters (like weights and biases, see the response about the architecture) with the goal of minimizing the error between the predicted and actual outcomes. Therefore, the epoch itself is a unit of the learning process. Additionally, one epoch also contains a single pass through the entirety of the validation dataset, such that we can gauge the generalization of the neural network after the training dataset pass.
We hope this answers the referee's question.

Page 15:

• On statement 'DELWAVE predictions for HS , d and Tm−1,0 with respect to those'.
Referee comment: please define these mathematically

AR: we have addressed this remark already in relation to reviewer's remark on page 7.

• On the statement 'with respect to those'.
Referee comment: what does this mean exactly?

AR: We are stating a comparison between DELWAVE and SWAN. To make this more clear, we have replaced the sentence "with respect to those" with "compared to those".

• On the statement 'Finally, Figure 11'.
Referee comment: you mean Figure 12?

AR: Figure 11 contains the mean absolute error plots denoted with blue jagged lines. It is indeed this plot that we reference and we have made this more clear.

• On statement 'low amplitude, short wavelength, stochastic ocean surface behavior' focusing on 'ocean'.
Referee comment: why do you mention ocean if you are working on the Adriatic Sea?

AR: We only use “ocean” as a figure of speech. Similarly, we tend to call numerical ocean models for regional basins “ocean” models even though they are used for modeling of marginal seas. We also refer to “ocean currents” and “ocean waves” even when talking about currents and waves in the Mediterranean or Aegean or Adriatic…We have nevertheless replaced the word “ocean” in the manuscript with the word “sea”.

Page 21:

• When we mention that DELWAVE is fast enough to train and conduct inference. **Referee comment: you should show computational time comparisons between offline training + run of DELWAVE vs run of SWAN**

AR: We have added an estimate for the model’s training time and we have provided an expected inference time when using the trained model. While the model took about two days and a half to train, it can process more than 100 input fields per second on a personal desktop computer with a low end graphics cards.

Furthermore, the DELWAVE GitHub repository contains instructions pertaining to installing libraries required by DELWAVE, the project structure setup, model training, and usage of the already trained model, which can be found on Zenodo, in conjunction with the training and test data.

Page 22:

• On statement ’We have thoroughly analyzed which architecture’ focusing on ’architecture’

Referee comment: what do you mean? this word is usually used in computer science to describe hardware

AR: In machine learning, "network architecture" refers to the structural design of a neural network. This encompasses the arrangement and connections of the nodes in the network, as well as how these nodes are organized into layers. Key aspects of network architecture include:

> Layer Types: These can include input layers, hidden layers (such as convolutional layers in CNNs, such as DELWAVE,), and output layers. Each layer type serves a specific function in processing the data.

Number of Layers: This refers to the depth of the network. Deep learning involves networks with many layers.
Number of Nodes in Each Layer: This determines the width of each layer.
Connections Between Nodes: This includes the way nodes in different layers are connected to each other, such as in fully connected, convolutional, or recurrent structures.
Activation Functions: These are functions applied to the nodes to introduce non-linear properties to the network, enabling it to learn more complex patterns. In DELWAVE, SiLU activations are used.
Weight and Bias Parameters: These are part of the learning aspect of the network, adjusted during the training process.

The design of the network architecture is crucial as it influences the model's ability to learn from data, its efficiency, and its performance on specific tasks such as image recognition, natural language processing, or predictive modeling. Different architectures are suited for different types of tasks; for example, DELWAVE is based on Convolutional Neural Networks (CNNs) which are often used for image processing. And we technically provide the input data to DELWAVE as a sequence of images.

It is not at all obvious which network architecture will perform best. Therefore DELWAVE architecture was changed and tested extensively before we arrived at the final version of the architecture that is presented in the manuscript. We hope this answers the reviewer's question.

On Figures:

• Figure 9: **Referee comment: What are the units for 'epochs' on the x axis in Figure 9?**

AR: In the context of the machine learning training process, an "epoch" refers to one complete cycle through the entire training dataset. During an epoch, the learning algorithm processes each sample in the dataset, using them to adjust the model's parameters (like weights and biases, see the response about the architecture) with the goal of minimizing the error between the predicted and actual outcomes. Therefore, the epoch itself is a unit of the learning process. We hope this answers the referee's question.

• Figure 10: **Referee comment: what is displayed on the x axis? mean wave direction? I do not understand how to read this picture.**

AR: Figure 10 depicts a scatter plot of DELWAVE forecasts (y-axis) compared to their SWAN targets (x-axis) for mean wave period, significant wave height and mean wave direction. If DELWAVE predicted each value perfectly and precisely matched the corresponding value from the SWAN model, all points would lie solely on the diagonal of the plot. In such plots, departures from diagonal are a measure of imperfection. The closer to the diagonal the scattered set lies, the better the forecast. We have now included more data into the Figure caption and we hope this clarifies the Figure:

Figure 10. A scatter plot of DELWAVE forecasts (y-axis) compared to their SWAN targets (x-axis) for mean wave period [s] (**1st column**), significant wave height [m] (**2nd column**) and mean wave direction [°] (**3rd column**) at locations AA (**top row**), OB (**middle row**) and MB (**bottom row**). Mean wave directions are listed in nautical notation (0° = North, 90° = East, etc.). Dashed diagonal line in each plot indicates a perfect forecast.

• Figure 12: **Referee comment: what are the white parts in the plots?**
AR: The white parts in the plot refer to combinations of direction and variable for which no occurrence was found in the data. This has been clarified in the caption in the revised version.

• Figure 14: On dotted lines,
referee comment: it's hard to see they are dotted
AR: The figure has been adjusted to improve readability.

• Figure 15: **Referee comment: It would be more informative to scale the y-axis to better show the quantitative difference between the red and blue line. For instance, the bottom right corner plot should have a y-axis ranging from 1.9 to 4. The part from 0 to 2 is useless. You should also maybe introduce a curve on a different axis that displays the pointwise error. Another thing is, why are the red and blue curves different compared to the black one? Can you expand on that??**
AR: We thank the reviewer for this comment. We should have stated more clearly that the aim of Figure 15 was precisely to demonstrate that the red and blue line, depicting 99-percentile wave field conditions at the end of 21st century, were closer to one another than they are to the black line, which is depicting the control period 1970-1998. This image aims to show that the errors, introduced by the DELWAVE model, are substantially smaller than the difference between scenario (2070-2100) and control periods (1970-1997). We believe that replotting the images would not make this clearer and we hope the reviewer forgives us for not replotting the image. We have amended the manuscript to include this explanation and hopefully make the paper more readable.

As for the pointwise error, we would like to point out that these DELWAVE errors are already discussed explicitly in Section 5.1., and are depicted in Figures 10 and 11. We hope that this explanation adequately addresses the reviewer's concerns.

Referee comment: various typos and stylistic/technical remarks regarding the contents of the manuscript.

AR: We thank the reviewer for the technical and stylistic comments. All corrections have been incorporated into the manuscript.

Response to Review #2

As before, **referee comments are in bold face blue**, while our response is in normal face and is marked as AR (authors response).

This study presents an emulator for wind-waves in the Adriatic Sea that is claimed to be capable to produce "numerically cheap large-ensemble predictions over synoptic to climate time scales". While the procedure following which the emulator was built, trained and tested is well-described and seems sound (this should be evaluated by an expert in machine learning techniques), the setup and evaluation of the SWAN wave model is lacking. Further, no discussion about the choices made to build the emulator is present. Consequently, although the scientific significance and quality of this manuscript for the understanding of future wind-wave climate in the Adriatic Sea has the potential to be high, it is, in the current stage, not convincing. I list below some major comments to support the authors with the resubmission of their manuscript.

AR: We thank the reviewer for taking the time to review our paper and we address their specific remarks below.

The main issue with the article is the relatively low resolution used in the COSMO-CLM and SWAN models. In particular, the SWAN model horizontal resolutions reach at best 2 km for the AA station but up to 8-9 km for the MB station and about 6 km for the remaining stations. In the southern Adriatic the resolutions are close to the Med-CORDEX regional climate models covering the full Mediterranean Sea at about 12 km of resolution. Further, from Bonaldo et al. (2020), it seems that "the minimum water depth in the model grid equals approximately 8 m". In my opinion this defy the purpose of using an emulator for the wind-waves at only 6 locations of interest. Such

an approach should, in fact, allow to reach a resolution of few (maybe hundred) meters at locations of interest (where bathymetry should be updated with observations; e.g., multi-beam or LiDAR) in order to properly resolve refraction, diffraction, shoaling, reflection, etc. of the waves along the coastline.

AR: In terms of resolution and processes, the setup and resolution is generally compatible with an accurate description of refraction, as can be seen from a simple application of Snell's law, keeping the rotation of the wave propagation within the directional resolution in most of the conditions, thus preserving causality and stability without invoking the refraction limitators implemented in SWAN (see SWAN technical documentation for details). In the same conditions, there is no major hindrance to energy conservation and therefore to shoaling description. With reference to the state of the art of publicly available ocean climate regional modeling, to our best knowledge Med-CORDEX does not provide wave information, and in any case those models have serious problems in reproducing other atmosphere-driven processes (Dunić et al., 2019). Furthermore, pushing the analysis at a very high resolution on the nearshore is out of the scope of the present work, as it was in the case of the work by Bonaldo et al. (2020), for several reasons. The most relevant ones in the context of this discussion are probably the lack of homogeneous high-resolution morpho-bathymetric data along the whole Adriatic coast, of extensive long-term wave observations for model validation in the very nearshore zone, and the impossibility of reproducing, again at the scale of the whole Adriatic Sea, the feedback between morphodynamic processes and nearshore wave dynamics (of course updating the bathymetry with observations is challenging at this scale for the past and simply impossible in future conditions). In this direction, keeping the analysis independent on morphodynamics removes an important element of uncertainty. Nonetheless, for very nearshore applications in which morphodynamics is crucial, DELWAVE can be used to assess climate and inter-model variability at the regional scale thus guiding the setup of the relevant nearshore model experiments.
Finally, it is important to notice (and we make it clearer in the revised version, as we probably did not succeed completely in conveying this message) that DELWAVE has been designed to be applicable to a broad set of conditions and geographical settings, and the choice of focusing on a limited number of test locations is only aimed at demonstrating its skills under different geomorphological (open coast, sheltered bay, etc.) and meteo-marine regimes (exposure to swell/wind seas, bimodal wind regimes, etc.), as better detailed in the following response.

References

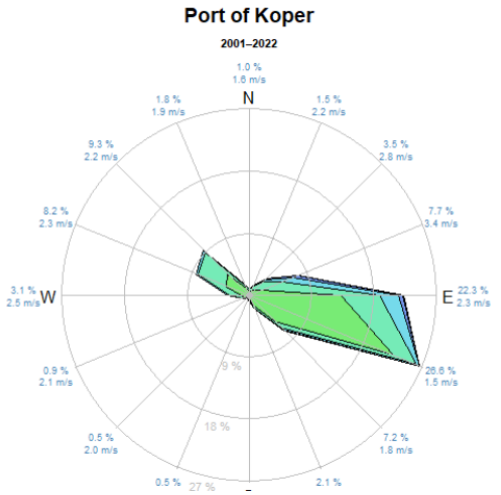
Dunić, N., Vilibić, I., Šepić, J., Mihanović, H., Sevault, F., Somot, S., Waldman, R., Nabat, P., Arsouze, T., Pennel, R., Jordà, G., & Precali, R. (2019). Performance of multi-decadal ocean simulations in the Adriatic Sea. *Ocean Modelling*, *134*(May 2018), 84–109. https://doi.org/10.1016/j.ocemod.2019.01.006

The choice of the locations where the emulators were built is also a bit puzzling and unexplained. Why the stations OB2 and OB3 are of any interest? Why the emulator results are only presented for AA, OB and MB and not at Grado, for example? Why emulators were not built for all the major coastal cities along the Adriatic coast and/or the Adriatic LNG terminal, the major commercial harbors like Koper, etc.? I understand the choice to include the wave buoy stations but not to limit the emulators to it.

AR: We thank the reviewer for this comment. We revised the manuscript to reflect more clearly that DELWAVE is trained on all listed points at once. There is only one emulator, which produces predictions for all points at once. We can in principle add an arbitrary number of training points to DELWAVE. In the present manuscript the training points were chosen to cover most of wave climate variability (coastal and offshore) in the Adriatic basin. For this we need coastal points like OB and MB, but also offshore points like OB2. AA was included as a location of particular importance due to the long-running marine observational infrastructure at that location.
We have included Grado location in the training set and also in the manuscript at the reviewer's request. The quality of forecasts at Grado location is completely in line with other locations from the initial version of the manuscript. We hope this satisfies the reviewer.

As for Koper, we chose not to include it in the training set since substantial waves are rarely, if ever, an issue in Koper - the strongest wind in Koper is Bora, which blows offshore and causes waves at AA location, which is included. The other dominant Adriatic wind, the southeasterly Scirocco, is occluded in Koper by the surrounding topography, but it is present at other locations in the paper. Only westerly episodes have the potential to cause waves but they are rare, as can be seen from the windrose of the port of Koper:

Koper, port



Back to the list of sites

This image is available at the Slovenian environment agency website at the following web location: https://meteo.arso.gov.si/met/en/climate/diagrams/wind/koper-luka/ (last accessed: 22 Nov 2023).

This leads to another important point. The evaluation of the SWAN model against observations is not presented. The COSMO-CLM model has been evaluated for the (EURO-?) CORDEX domain by forcing its boundaries with reanalysis (i.e., ERA-Interim). The evaluation of the SWAN model should thus be performed during the period of this control run for extreme events (and not as a climatology like done in Bonaldo et al., 2020) and compare with the available observations in order to assess the capacity of the COSMO-CLM and SWAN models to reproduce bora/sirocco winds and wave parameters, respectively. Without such an evaluation for extreme events, the skills of the COSMO-CLM and SWAN models during sirocco/bora events, and, hence, of the emulator, cannot be thoroughly assessed and no conclusion about the quality of the results presented in the manuscript can be reached (i.e., an emulator can only be as good as the geoscientific models it is built with).

AR: In the first version of the manuscript we probably did not emphasize enough that the focus of the work is on the developed methodology and its potential in the emulation of computationally demanding wave modelling tasks, with particular reference to ensemble approaches over climate change time scales. In this, the application to the Adriatic Sea should not be considered as an attempt to actually improve the model projections (this is not the scope of this tool), nor of course as a

forecast of sea states to be compared to synchronous observations, but instead as a demonstrator for the applicability of DELWAVE. In this sense, the quality of the results should not be intended in terms of matching with observations, but instead in terms of matching with the model. With particular reference to the SWAN implementation from Bonaldo et al. (2020), we acknowledge that the evaluation run suggested by the reviewer can in principle be useful for assessing the capability of a model to reproduce the observed regional dynamics in response to a good (in theory the best) available approximation of reality in terms of forcing and boundary conditions. In the case mentioned by the Reviewer, this approximation was given by ERA-Interim (worth noting in this context, as far as we know the fields from the COSMO-CLM evaluation run are not publicly accessible at present), as described by Bucchignani et al. (2016). Nonetheless, since the skill assessment of a wave climate model is generally aimed at evaluating whether it can recreate the past wave climate to a sufficient degree of accuracy, it is quite common in this kind of studies to focus on the comparison of climate model results under "historical" conditions (that is, driven by a climate model in the recent past) under observations or reanalyses. Of course, since "historical" runs are not synchronised with the observed variability, their results can only be considered in aggregated terms. Some examples along this line can be found for instance in Benetazzo et al. (2012), Fan et al. (2014), Tiron et al. (2015), and De Leo et al. (2021). Although in aggregated terms, the validation provided by Bonaldo et al. (2020) shows that Bora and Sirocco storm regimes and wave parameters are well captured by the model. Furthermore, assessing the skills of DELWAVE against SWAN rather than against the observations (clearly, only in aggregated terms and with reference to the past climate) allows to draw a conclusion on the quality of the results of DELWAVE independently on the performance of the model used for the training, providing a much more general information in terms of replicability and transferability of this method. In the revised version we try to better clarify these aspects.

References:

Benetazzo, A., Fedele, F., Carniel, S., Ricchi, A., Bucchignani, E., & Sclavo, M. (2012). Wave climate of the Adriatic Sea: A future scenario simulation. *Natural Hazards and Earth System Science*, *12*(6), 2065–2076. https://doi.org/10.5194/nhess-12-2065-2012;

De Leo, F., Besio, G., & Mentaschi, L. (2021). Trends and variability of ocean waves under RCP8.5 emission scenario in the Mediterranean Sea. *Ocean Dynamics*, *71*(1), 97–117. https://doi.org/10.1007/s10236-020-01419-8;

Tiron, R., Gallagher, S., Gleeson, E., Dias, F., & McGrath, R. (2015). The future wave climate of Ireland: From averages to extremes. *Procedia IUTAM*, *17*(2013), 40–46. https://doi.org/10.1016/j.piutam.2015.06.007

Fan, Y., Lin, S. J., Griffies, S. M., & Hemer, M. A. (2014). Simulated global swell and wind-sea climate and their responses to anthropogenic climate change at the end of the twenty-first century. *Journal of Climate*, *27*(10), 3516–3536. https://doi.org/10.1175/JCLI-D-13-00198.1

In terms of the technical implementation of the emulator, I would recommend the article to be reviewed by an expert in machine learning. The article is presenting a lot of details about the way the emulator was built that only such a specialist can accurately review. Another important question not discussed in this paper is whether or not the emulator can be used with other regional atmospheric forcing than COSMO-CLM. In terms of producing robust ensembles to cover the climate uncertainty, it is crucial to use as many different atmospheric climate models as possible.

AR: we would like to thank the reviewer for pointing this out. DELWAVE can certainly be used with any kind of regional atmospheric and wave models, provided their results span a large enough time window to make learning meaningful.

We fully agree with the reviewer's appeal to use DELWAVE with as many atmospheric and wave models as possible. While this extends beyond the scope of this initial manuscript, we are dedicated to the idea and have done our best to provide a self-contained code DELWAVE repository with training examples in order to facilitate the widespread training and use of DELWAVE emulator. The following has been added to the Introduction section of the manuscript.

> While DELWAVE model, presented in this manuscript, has been trained and tested on the outputs of COSMO-CLM and SWAN models, the model can be used with any regional atmospheric and wave modeling setup, or their ensembles, provided that available model results span a large enough time window to make DELWAVE training meaningful.

As nevertheless the COSMO-CLM model is corrected with the ERA5 reanalysis, one can even ask, why not using directly the CMIP6 100-km resolution ensemble of atmospheric models (obviously corrected with ERA5 in order to catch extreme events) to implement, test and train the emulator. As it seems that the authors limit their study to wave buoy locations, it could even be envisioned to directly use the wave observations to build such an emulator (and maybe even skip the SWAN modelling). I am not suggesting that it is a better method but I am just highlighting that, over all, there is a lack of discussion concerning both the choices made to build the emulator and the practical use of the emulator for climate studies.

Actually, it is important to point out that COSMO-CLM has been used, in another work (Benetazzo et al., 2022), to correct ERA5 fields precisely because this reanalysis struggles in the reproduction of wind fields on the Adriatic basin: working the other way around, if this is what the reviewer is suggesting, may not prove really beneficial. Concerning the possibility of using observed wave data instead of numerical models, we recall that, as previously remarked, DELWAVE is designed for general applicability and we are focusing on climatologies. Therefore, with respect to our approach, the main limitations with working on wave observations are that the latter strategy is actually bound to be site-specific, and above all it does not permit projections until the end of the century, because of course we don't have the future data that we would need for predictions it the model was trained on observations. We hope that the modifications made throughout the paper in the revised version can better clarify these aspects.

Again, we haven't made it clear enough that we are not comparing against observations but are working with climatologies. We cannot train on observations because then we cannot make projections until the end of 21-century, because we don't have data from the future that we would need for predictions it the model was trained on observations. I think this comment is related to the fact that the reviewer is not an expert in machine learning.

Finally, the last major point is that the authors did not prove that their emulator was cheaper than, for example, look-up tables that are commonly used for wind-waves. In combination with the relatively low resolution of the SWAN model and the lack of evaluation of the COSMO-CLM and SWAN models during storm events, the manuscript thus fails to prove the added value of the DELWAVE model.

Recalling the previous points concerning the validation of SWAN in the previous work and the discussion on the benefits of focusing, in this work, on the agreement between emulator and model (generality, transferability, etc.), and the general applicability of DELWAVE on any number of locations within the domain of the wind and wave models, we see a clear advantage from using DELWAVE instead of lookup tables. Unless the reviewer is referring to some specific kind we are not aware of, lookup tables typically require a large number of data for being "calibrated" (the parallel of "training" in DELWAVE) and they are definitely site-specific, but are way less flexible in explicitly capturing behaviours and processes that instead are embedded, though in different ways, both in wave models and in DELWAVE (e.g. spatial and temporal variability of wind fields, the relationship with basin-scale sea states). Furthermore, if we understand lookup-tables correctly, they merely reflect the past and do not allow for ensembles of basin-scale wave-field projections on climate timescales in a changing wave climate -

which is an additional DELWAVE benefit. We hope this appropriately addresses the reviewer's concerns.

Overall, even if the mathematical exercise of setting up an emulator for wind-waves in the Adriatic Sea is interesting as such, I am not convinced that, the DELWAVE emulator can really be used for the intended purposes stated in the introduction: study "morphodynamic processes", the "safety and durability of human infrastructures, along the coast and offshore" and assess "the feasibility and improving the design of wave energy converter facilities".

The goal of our paper is to present a newly developed model emulator and demonstrate its capability to reproduce the results of a wave model under different conditions and geometries, with the overall goal of providing a generally applicable tool for supporting studies in which wave climate is particularly relevant. Exactly like numerical models need some dedicated calibration and validation effort when applied to a new context or geographical setting, it is expected that new applications of DELWAVE can require some specific fine tuning and/or validation. With this caveat, DELWAVE can be applied as a surrogate of a "traditional" model to the extent to which the validation proves it capable of emulating the model prediction. We add some text along these lines in the conclusions.