

Comparative Evaluation of Deep Learning Architectures for Coastal Mean Sea Level

¹*Yeong Nain Chi, ²Catherine Ngo, ³Orson Chi

^{1,2}Department of Agriculture, Food and Resource Sciences, University of Maryland Eastern Shore, Princess Anne, Maryland.

³Department of Computer Science and Engineering Technology, University of Maryland Eastern Shore, Princess Anne, Maryland.

Abstract

Accurate forecasting of mean sea level (MSL) is critical for coastal risk management, infrastructure planning, and climate adaptation, particularly along the low-lying Mid-Atlantic coast of the United States. This study presents a comprehensive comparative evaluation of four data-driven deep learning and neural network architectures, including Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), one-dimensional Convolutional Neural Network (CNN), and a lagged feed-forward neural network (FFNN), applied to seasonally adjusted monthly MSL records at the NOAA tide gauge station in Ocean City, Maryland (Station 8570283), spanning August 2002 through February 2025. All models share a standardized 24-month lookback window (lags 1–13 for the FFNN), 80/20 train/test split, Adam optimization, and early stopping, enabling controlled comparison. Models are evaluated using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2) on held-out test data. Results indicate that the GRU achieves the best predictive performance among the neural architectures (RMSE = 0.0704 m, MAE = 0.0587 m, R^2 = 0.099), followed by the LSTM (RMSE = 0.0734 m, R^2 = 0.020). Even the GRU explained only about 10% of the test-period variance, and its RMSE was only about 3% lower than that of a persistence forecast. The CNN and FFNN both yield negative R^2 values (–0.219 and –0.907 respectively), which on this single test period placed the recurrent models ahead of the convolutional and feed-forward networks; this ranking was not stable across test periods (see below). A 24-month ensemble forecast is presented for all models through February 2027. The FFNN ensemble spread ($\pm 1\sigma$ across 20 independently initialized networks) is reported only as a measure of initialization variability; calibrated prediction intervals are derived from the ARIMA model and from rolling-origin forecast errors. When benchmarked under the identical protocol against persistence, seasonal-naïve, random walk with drift, linear trend, ARIMA, SARIMA and exponential smoothing (ETS) baselines, however, all three statistical time-series models outperformed every neural architecture: ARIMA(3,0,0) with a linear trend achieved the lowest test error (RMSE = 0.0642 m, R^2 = 0.250), and the GRU improved on a simple persistence forecast by only 3% in RMSE. These findings indicate that, for this univariate monthly record, the neural architectures tested did not add predictive value beyond well-specified linear time-series models; whether this holds at other stations has not yet been tested. An expanding-window rolling-origin evaluation over 49 forecast origins (December 2012 – December 2024), with every model re-trained or re-estimated at each origin, confirmed this ranking at forecast horizons of 1, 3, 6, 12 and 24 months: ARIMA had the lowest or near-lowest RMSE at every horizon (0.056–0.069 m), whereas the best neural model was 6–15% less accurate, and the single-split ranking of the neural architectures was not stable across test periods. All models predict the seasonally adjusted monthly MSL level in meters; training the networks on residuals from a training-period trend improved their accuracy and stabilised their forecasts but did not change their ranking relative to the statistical models. The NNAR model of R's `nnetar()`, added as a separate benchmark, was far more accurate than the FFNN on the single split (RMSE = 0.0708 m) but was unstable at multi-step horizons and did not outperform ARIMA. Backtesting of the recursive multi-step forecasts showed that the neural models increasingly under-forecast with lead time, and that ARIMA prediction intervals were close to nominal coverage, whereas the FFNN ensemble spread and the simulated `nnetar()` intervals were far too narrow to serve as prediction intervals.

Keywords: mean sea level; time series forecasting; deep learning; NNAR; LSTM; GRU; CNN; Ocean City; Mid-Atlantic.

Yeong Nain Chi

Department of Agriculture, Food and Resource Sciences University of
Maryland Eastern Shore Princess Anne, Maryland.
Email: ychi@umes.edu

Received: 10-Aug-2026

Revised: 11-Sep-2026

Accepted: 8-Oct-2026



© 2026 Copyright by the Authors.

Licensed as an open access article using a [CC BY 4.0 license](https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Global mean sea level (MSL) has risen by approximately 20 cm since the late nineteenth century, with the rate of rise accelerating over recent decades under the combined influence of ocean thermal expansion, land ice melting, and changes in terrestrial water storage (Church & White, 2011; Kopp et al., 2015). Along the US Atlantic coast, sea level rise is further amplified by regional factors including gravitational and rotational effects of ice mass loss, changes in the Atlantic Meridional Overturning Circulation (Yin et al., 2010), and ongoing glacial isostatic adjustment that causes land subsidence at many Mid-Atlantic tide gauge stations (Sweet et al., 2017; Boon, 2012). At Ocean City, Maryland, a major barrier-island resort community with significant economic and ecological vulnerability, the NOAA relative sea level trend exceeds the global mean, reflecting both eustatic rise and local land subsidence (Ezer, 2019; NOAA, 2025).

Accurate prediction of future monthly and seasonal MSL is essential for coastal flood risk assessment, infrastructure investment decisions, emergency management planning, and the design of nature-based and engineered coastal defenses. Traditional statistical approaches to sea level forecasting, including linear regression, autoregressive integrated moving average (ARIMA) models, and empirical mode decomposition, are well established but struggle to capture nonlinear interactions, long-range temporal dependencies, and abrupt episodic anomalies caused by storm surges, strong northeasterly winds, and shelf-wide oceanographic variability (Zhang et al., 1998). The last decade has witnessed a rapid proliferation of deep learning methods in geophysical time series forecasting. Recurrent neural networks (RNNs), and in particular the Long Short-Term Memory (LSTM) architecture introduced by (Hochreiter & Schmidhuber, 1997) and the Gated Recurrent Unit (GRU) proposed by (Cho et al., 2014), have demonstrated strong capacity to model sequential data with long-range dependencies. Convolutional Neural Networks (CNNs), originally developed for image recognition (LeCun et al., 1998), have been adapted to one-dimensional temporal sequences and offer efficient local pattern extraction. Neural Network Autoregression (NNAR), implemented in R's forecast package as `nnetar` (Hyndman & Athanasopoulos, 2018), applies a feedforward multilayer perceptron (MLP) to lagged observations, providing a transparent and computationally inexpensive autoregressive baseline that averages the forecasts of an ensemble of networks.

Despite the growing application of these methods to oceanographic and coastal problems, systematic comparative studies across multiple deep learning architectures applied to the same high-resolution tide gauge record are relatively rare. Most published work focuses on a single architecture or compares deep learning to classical statistical methods, leaving open the practical question of which neural network paradigm performs best for noisy, short monthly sea level records typical of operational NOAA tide gauge stations.

This study addresses that gap by implementing, training, and evaluating LSTM, GRU, CNN, and FFNN models on a single consistent dataset from Ocean City, Maryland, under a controlled experimental framework. The NNAR model of R's `nnetar()` (Hyndman & Athanasopoulos, 2018) is included as an additional benchmark. This study reports test-set performance metrics, provides 24-month ensemble forecasts, and discuss what the results imply for this station and study design.

The remainder of this paper is structured as follows. Section 2 reviews the relevant literature. Section 3 describes the study site and data. Section 4 introduces model configurations. Section 5 presents and discusses results. Section 6 provides conclusions and recommendations.

2. Literature Review

The US Atlantic coast, and particularly the roughly 1,000-km stretch north of Cape Hatteras identified as a "hotspot" of accelerated sea level rise (Ezer, 2019), experienced increases in the rate of sea level rise about three to four times the global average between 1950–1979 and 1980–2009 (Sallenger et al., 2012). (Ezer, 2019) demonstrated that weakening of the Gulf Stream contributes to regional sea level rise by allowing coastal waters to spread westward and elevate coastal sea levels. (Kopp et al., 2015) showed that gravitational fingerprinting of ice mass loss results in spatially heterogeneous sea level change, with the US East Coast experiencing amplification relative to global mean trends. (Boon, 2012) identified statistically significant acceleration in sea level rise at multiple Atlantic coast tide gauge stations, including those in the Chesapeake Bay and Mid-Atlantic region.

Ocean City, Maryland sits on Fenwick Island, a narrow barrier island characterized by high ecological sensitivity, tourism-dependent economy, and elevation generally within 1–2 m of mean sea level. The NOAA relative sea level trend at Station 8570283 reflects not only global and regional eustatic signals but also land subsidence related

to groundwater extraction, sediment compaction, and residual glacial isostatic adjustment. These compounding factors make Ocean City a high-priority site for sea level monitoring and prediction.

Recurrent neural networks have been applied to oceanographic forecasting with increasing frequency since the mid-2010s. LSTM networks, with their ability to learn long-range dependencies through cell state and gating mechanisms, have shown promise in modelling tidal and sub-seasonal sea level variability. LSTM networks have been reported to outperform ARIMA and simple feed-forward networks in some multi-step oceanographic prediction studies (Hochreiter & Schmidhuber, 1997; Kingma & Ba, 2015). GRUs, which simplifies the LSTM cell by merging the forget and input gates into a single update gate (Cho et al. 2014; Chung et al. 2014), have performed comparably to LSTMs in benchmark comparisons (Chung et al. 2014), with fewer parameters, which may reduce the risk of overfitting on short training records.

One-dimensional CNNs have been applied to hydrological and oceanographic time series through their ability to detect local temporal patterns using convolutional filters. While CNNs do not possess inherent sequential memory, they have been used effectively in hybrid architectures (e.g., CNN-LSTM) where local feature extraction precedes recurrent temporal modelling. Standalone 1D CNNs, however, may be less suited than recurrent architectures to univariate monthly series dominated by a long-term trend rather than by localized features (LeCun et al., 1998).

NNAR, as implemented in R's `nnetar()` function (Hyndman & Athanasopoulos (2018), applies a feedforward MLP to autoregressive lag inputs and is commonly used as a neural network baseline in forecasting competitions. Its ensemble approach, averaging predictions from multiple independently initialized networks, mirrors the recommendation for neural network forecasting under parameter uncertainty, although the spread of such an ensemble is not a calibrated prediction interval (Section 5.5). Despite widespread use in the forecasting literature, NNAR has received limited evaluation for sea level applications. The three-hidden-layer feed-forward model evaluated in this study (FFNN, Section 4.4) differs from `nnetar()`, so `nnetar()` is evaluated separately as a benchmark. The practical context of MSL forecasting is increasingly linked to coastal flood risk quantification. (Vousdoukas et al., 2018) demonstrated that global extreme sea level events are projected to intensify significantly under climate change scenarios, with high-latitude and tropical coastlines most

exposed but Mid-Atlantic barrier islands also substantially threatened. Sweet et al. (Sweet et al., 2017) developed US national sea level rise scenarios for infrastructure planning and flood frequency estimation, noting the critical role of local tide gauge records in downscaling global projections. Operational MSL forecasting, as distinct from seasonal climate prediction, aims to provide actionable outlooks on the 1–24 month horizon relevant to coastal management decisions.

3. Materials

3.1 Study Area

Ocean City is a town in Worcester County, Maryland, on the Atlantic coast of the United States (Figure 1). It occupies the southern part of Fenwick Island, a barrier island that extends north into Delaware and is separated from the mainland by Assawoman and Isle of Wight Bays. At the 2020 Census the town had 6,844 residents and a land area of 4.53 square miles (11.7 km²) (U.S. Census Bureau, 2026). Its southern end is the Ocean City Inlet, which was opened by a storm in August 1933 that separated Fenwick Island from Assateague Island and was subsequently fortified to keep it open (National Park Service, 2026). The NOAA tide gauge used in this study (Station 8570283, Ocean City Inlet) is located at the inlet, at 38.328° N, 75.092° W (NOAA CO-OPS, 2025).

3.2 Data Source and Extraction

The study uses monthly MSL time series from NOAA Tides & Currents, Station 8570283, located at Ocean City, Maryland (38.33°N, 75.09°W) (NOAA, 2025). The record spans August 2002 through February 2025, yielding 271 monthly observations. The variable is NOAA's monthly mean sea level product for this station: monthly means in meters relative to the most recent Mean Sea Level (MSL) datum established by NOAA CO-OPS, with the average seasonal cycle removed by NOAA (NOAA, 2025; NOAA CO-OPS, 2025). The NOAA record at Station 8570283 begins in 1975; this study uses the August 2002 – February 2025 segment. The modelling target of every model is this seasonally adjusted monthly MSL level itself (column `Monthly_MSL`, in meters). The series was not detrended or converted to anomalies before training; preprocessing is described in Section 4.5. Key dataset characteristics are summarized in Table 1.

The data were downloaded as the comma-separated file `8570283_meantrend.csv` from the station page of the NOAA Tides & Currents Sea Level Trends product (<https://>

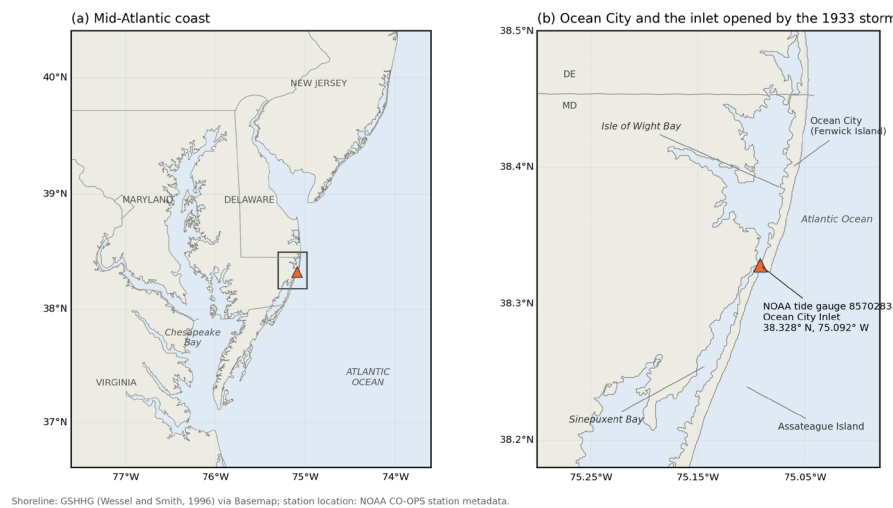


Figure 1. Study area. (a) Mid-Atlantic coast of the United States with the location of Ocean City, Maryland (box); (b) Ocean City, the Ocean City Inlet and NOAA tide gauge 8570283 (38.328° N, 75.092° W). Shoreline from the GSHHG database (Wessel & Smith, 1996); station location from NOAA CO-OPS (NOAA CO-OPS, 2025).

tidesandcurrents.noaa.gov/sltrends/sltrends_station.shtml?id=8570283) on 29 April 2025. The file contains the year, month, monthly mean sea level with the average seasonal cycle removed, NOAA’s linear trend, and its 95% confidence bounds. The NOAA record begins in June 1975 but contains no data from February 1999 to July 2002 (42 months); the analysis uses the continuous segment

that follows this gap, August 2002 – February 2025. This segment has no missing months, duplicate months or missing values, so no gap filling, interpolation or removal of observations was required. NOAA periodically re-estimates this product, and values in the version available in October 2026 differ from the analysed file by about 1–3 mm in the months checked.

Table 1. Dataset summary for NOAA Station 8570283, Ocean City, Maryland.

Attribute	Value	Notes
Source	NOAA Tides & Currents	Station 8570283
Location	Ocean City, MD (USA)	38.33°N, 75.09°W
Period	Aug 2002 – Feb 2025	271 monthly observations
Variable	Monthly Mean Sea Level (m)	Seasonal cycle removed by NOAA; relative to station MSL datum
Linear Trend	~0.057 → 0.175 m	NOAA fit to the 1975–present record (5.3 mm/yr); reference only, not used in modelling
Range (MSL)	–0.088 to +0.359 m	High variability $\sigma \approx 0.079$ m
Train / Test Split	80% / 20%	Aug 2002 – Dec 2020 / Jan 2021 – Feb 2025 (50 target months)

The linear trend column in the dataset represents the modelled long-term rise (ranging from approximately 0.057 m in August 2002 to 0.175 m in early 2025), while the Monthly MSL column represents the observed gauge reading including natural variability. The Linear_Trend column and its confidence bounds are NOAA’s least-squares fit to the full 1975–present station record (5.3 mm/yr). Because that fit uses observations from the test

period, it was used only for display (Figure 2) and for qualitative comparison, never for detrending, as a model input or in evaluation. Over the training months (August 2002 – December 2020), an OLS fit to Monthly_MSL gives a steeper local rate of 7.4 mm/yr. Figure 2 illustrates the full observed record alongside the NOAA linear trend and confidence bounds. The record exhibits pronounced interannual variability, episodic positive anomalies, which

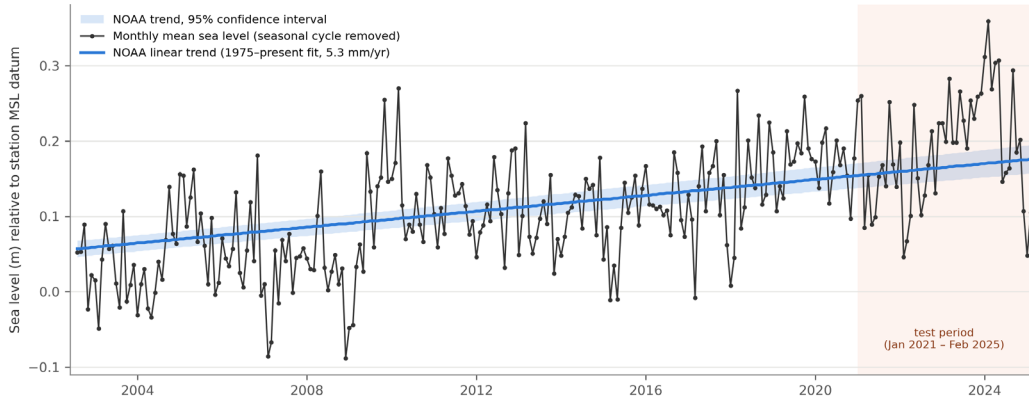


Figure 2. Monthly mean sea level at Ocean City, Maryland (NOAA Station 8570283; seasonal cycle removed), August 2002 – February 2025, with NOAA's linear trend and its 95% confidence interval (NOAA, 2025), (NOAA CO-OPS, 2025). The shaded interval is the test period of the single split.

may reflect storm surge and wind-driven coastal setup, and a clear positive trend consistent with regional sea level acceleration.

4. Methods

4.1 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a specialized recurrent neural network (RNN) architecture designed to capture both short-term and long-term temporal dependencies in sequential data while mitigating the vanishing gradient problem commonly encountered in conventional RNNs (Hochreiter & Schmidhuber, 1997). At each time step t , the LSTM unit processes the input vector x_t , the previous hidden state $h_{(t-1)}$, and the previous cell state $c_{(t-1)}$. The forget gate controls the retention of historical information:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (1)$$

The input gate determines how much new information is stored in memory:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

A candidate cell state is generated as:

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (3)$$

The cell state is then updated according to:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (4)$$

where $\odot$ denotes element-wise multiplication.

The output gate regulates information passed to the hidden state:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (5)$$

The hidden state is calculated as:

$$h_t = o_t \odot \tanh(c_t) \quad (6)$$

Finally, the forecast for the next time step is obtained using:

$$y_{t+1} = W_y h_t + b_y \quad (7)$$

This gating mechanism enables LSTM networks to effectively learn long-range temporal patterns in time-series forecasting applications.

4.2 Gated Recurrent Unit (GRU)

The Gated Recurrent Unit (GRU) is a simplified variant of LSTM that combines memory and hidden states into a single representation, thereby reducing computational complexity while retaining the ability to model long-term dependencies (Cho et al., 2014; Chung et al., 2014). Given input x_t and the previous hidden state $h_{(t-1)}$, the update gate is computed as:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (8)$$

The reset gate is defined as:

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (9)$$

The candidate hidden state is calculated by:

$$\tilde{h}_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h) \quad (10)$$

The hidden state is then updated according to:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \quad (11)$$

The forecast output is generated as:

$$y_{t+1} = W_y h_t + b_y \quad (12)$$

Compared with LSTM, GRU employs fewer parameters and often achieves comparable forecasting performance with reduced training time.

4.3 One-Dimensional Convolutional Neural Network (1D-CNN)

One-Dimensional Convolutional Neural Networks (1D-CNNs) extract local temporal features from sequential data through convolutional filters. CNN architectures have demonstrated strong performance in pattern recognition and time-series analysis because they efficiently learn hierarchical feature representations from raw inputs (LeCun et al., 1998). For a time-series input x and convolution kernel w of size k , the convolution operation at position t is defined as:

$$z_t = \sum_{i=1}^k w_i x_{t+i-1} + b \quad (13)$$

or, more compactly,

$$z_t = w * x_t + b \quad (14)$$

The resulting feature map is transformed using a nonlinear activation function, typically the Rectified Linear Unit (ReLU):

$$a_t = \max(0, z_t) \quad (15)$$

To reduce dimensionality and retain dominant features, max-pooling is applied:

$$m_j = \max_{a_{(j-1)p+1}, \dots, a_{jp}} \quad (16)$$

where p denotes the pooling window size.

The pooled features are flattened into a feature vector v , and the final prediction is produced using a dense layer:

$$y = W_d v + b_d \quad (17)$$

This architecture enables the model to learn important local patterns and temporal structures relevant to forecasting tasks.

4.4 Lagged Feed-Forward Neural Network (FFNN)

The lagged feed-forward neural network (FFNN) used in this study is a feed-forward network that uses lagged observations of a time series as predictor variables. The FFNN extends the traditional autoregressive (AR) framework by allowing nonlinear relationships between past observations and future values to be learned through hidden-layer transformations (Hyndman & Athanasopoulos, 2018). An NNAR(p, k) model consists of

p lagged inputs and k hidden neurons. For a univariate time series y_t , the input vector is:

$$X_t = [y_{t-1}, y_{t-2}, \dots, y_{t-p}]^T \quad (18)$$

The hidden layer is calculated as:

$$h_j = g(\sum_{i=1}^p w_{ji} x_i + b_j) \quad (19)$$

or, in vector notation,

$$h = g(WX_t + b) \quad (20)$$

where $g(\cdot)$ is typically a logistic sigmoid or hyperbolic tangent activation function.

The forecast output is generated by:

$$\hat{y}_t = \beta_0 + \sum_{j=1}^k \beta_j h_j \quad (21)$$

or equivalently,

$$\hat{y}_t = Vh + c \quad (22)$$

By incorporating nonlinear transformations of lagged observations, FFNN models can capture complex autoregressive relationships that are difficult to represent using linear time-series models. The FFNN used in this study generalises Eqs. (19)–(22) to three hidden layers (32, 16 and 8 sigmoid units) with dropout, min–max scaled inputs and weights estimated by Adam with early stopping (Section 4.5). It is therefore not the NNAR(p, P, k)_m model implemented by `nnetar()` in R's forecast package (Hyndman & Athanasopoulos, 2018; Hyndman & Khandakar, 2008), which has a single hidden layer, standardised inputs, weights estimated by BFGS through `nnet()`, and lags chosen by a data-driven rule. To avoid confusion, the model labelled NNAR in the original submission is referred to as the FFNN throughout this revision, and `nnetar()` is evaluated separately as an additional benchmark (Section 4.8).

4.5 Architectural hyperparameters

Modelling target and preprocessing. All models predict y_t , the seasonally adjusted monthly MSL in meters relative to the station MSL datum (Section 3.2). The neural networks are trained to predict the next month from lagged values, and multi-step forecasts are obtained recursively by feeding predictions back as inputs. No detrending, differencing or anomaly transformation was applied before training the neural networks, so the long-term rise is carried by the input values themselves and must be learned from the lagged levels. The only transformation

was min–max scaling, $x't = (y_t - y_{min}) / (y_{max} - y_{min})$, where y_{min} and y_{max} are the minimum and maximum over the training months only (re-computed at every origin in the rolling-origin evaluation). Network outputs were returned to meters by the inverse transformation $\hat{y}_t = \hat{x}'_t (y_{max} - y_{min}) + y_{min}$, and all metrics and forecasts are reported in meters on this level scale. Because test-period values above the training maximum (six of the 50 test months exceed 0.270 m) map to $x' > 1$, the networks must extrapolate beyond the scaled training range to follow the trend. For the statistical baselines (Section 4.6), the trend is estimated from the training data jointly with the other parameters: as the deterministic trend term in ARIMA and SARIMA, as the drift term of the random walk, or by OLS on the training months for the linear-trend baselines. Their forecasts are produced directly in meters. A sensitivity analysis in which the networks are instead trained on residuals from a training-period OLS trend is reported in Section 5.3.

All monthly MSL values were normalized to the $[0, 1]$ range using a MinMaxScaler fitted on the training months only. The scaler was subsequently used to inverse-transform all predictions back to physical units (meters) for evaluation and reporting. The dataset was split chronologically, with the first 80% of samples used for training and the most recent 20% reserved as a held-out test set. Because the split is applied to the input–target samples, 20% of the 247 windows used by the LSTM, GRU and CNN corresponds

to 50 target months (January 2021 – February 2025); for the FFNN (258 lag vectors) the test set covers approximately the last 52 months. The baseline models described in Section 4.6 were evaluated on the same 50 target months (January 2021 – February 2025) and estimated on August 2002 – December 2020. No future data were used to fit any scaler or model parameter. The exact month ranges of the training, validation and test segments of every model are listed in Table A1, and the order of splitting, scaling, window construction, validation and testing is shown in Figure A1 (Appendix A).

For the recurrent and CNN models (LSTM, GRU, CNN), input sequences of 24 months length were constructed by a sliding window of stride 1, yielding a total of 247 samples. For the FFNN, the input vector comprises the 13 most recent values, $y_{t-1}, \dots, y_{t-13}$. This set already contains the seasonal lag y_{t-12} , so no separate seasonal input is required. In the original implementation, y_{t-12} was nevertheless appended a second time, producing a 14-element vector with only 13 distinct inputs. The duplicate column is identical to an existing input and adds no information; re-training without it left the results unchanged (Section 5.4), and the reported FFNN results use the original implementation. With 13 lags, 258 input–target samples are available (targets from the 14th observation onward). Table 2 provides a detailed comparison of architectural hyperparameters.

Table 2. Architectural hyperparameters for all four models.

Parameter	LSTM	GRU	CNN	FFNN
Input Window	24 months	24 months	24 months	13 lags (14 inputs; lag 12 duplicated)
Architecture	2×LSTM (64→32)	2×GRU (64→32)	3×Conv1D (64→32→16)	MLP 3 layers
Activation	tanh / sigmoid	tanh / sigmoid	ReLU (conv), linear	Sigmoid / linear
Dropout	0.20	0.20	0.20	0.10
Optimizer	Adam	Adam	Adam	Adam
Loss	MSE	MSE	MSE	MSE
Batch Size	16	16	16	16
Max Epochs	200	200	200	200
Early Stop	Patience 20	Patience 20	Patience 20	Patience 20
Ensemble	1 model	1 model	1 model	20 models
Stopped Epoch	~200	24	21	~25 avg

The LSTM model consists of two stacked LSTM layers with 64 and 32 units respectively, each followed by a Dropout layer (rate = 0.2) [14], succeeded by a Dense layer with 16 ReLU units and a single linear output neuron. This architecture captures long-range temporal dependencies via the cell state and three-gate mechanism (input, forget, output) (Hochreiter & Schmidhuber, 1997).

The GRU model mirrors the LSTM architecture, replacing LSTM cells with GRU cells that combine the forget and input gates into a single update gate and merge the cell state with the hidden state (Cho et al., 2014). This reduces the number of trainable parameters by about one quarter (22,817 vs 29,857; Table A3).

The CNN model applies three successive one-dimensional convolutional layers with filter counts of 64, 32, and 16 and kernel size 3, followed by MaxPooling (pool size 2), Dropout (0.2), a Dense layer with 32 ReLU units, Dropout, and a linear output neuron. The CNN extracts hierarchical local temporal features from the input window but carries no persistent hidden state between windows (LeCun et al., 1998).

The FFNN model is a three-hidden-layer feedforward MLP with sigmoid activations in each hidden layer ($32 \rightarrow 16 \rightarrow 8$ nodes) and a linear output. Twenty independently seeded networks were trained and their predictions averaged, following the 20-network averaging used by R's `nnetar()`, although the network itself differs from `nnetar()` (Section 4.4) (Hyndman & Athanasopoulos, 2018). The spread of the ensemble is summarised as ± 1 standard deviation across members. This spread reflects only variability due to random initialization and is not a prediction interval (Section 4.9).

All models were compiled with the Adam optimizer (Kingma & Ba, 2015) (learning rate = 0.001, default $\beta_1 = 0.9$, $\beta_2 = 0.999$) and mean squared error (MSE) loss. Batch size was 16 for all models. Training was limited to 200 epochs with early stopping (patience = 20 epochs, monitoring validation loss, restoring best weights). For each model, 15% of the training data was withheld as a validation set for early stopping monitoring. No data augmentation or transfer learning was applied. TensorFlow and NumPy random seeds were fixed (seed = 42) for reproducibility, with the FFNN ensemble using seeds 0, 7, 14, ..., 133 across 20 members.

Model performance on the held-out test set was evaluated using three complementary metrics: Root Mean Squared Error: $RMSE = \sqrt{[(1/n) \sum (y_i - \hat{y}_i)^2]}$, sensitive to large prediction errors. Mean Absolute Error: $MAE = [(1/n) \sum |y_i$

$- \hat{y}_i|]$, providing an interpretable average absolute deviation in meters. Coefficient of Determination: $R^2 = 1 - SS_{res}/SS_{tot}$, measuring explained variance relative to a naïve mean forecast. Negative values indicate worse-than-mean performance. A 24-month rolling autoregressive forecast was generated for each trained model by iteratively feeding the most recent 24 predictions back as input, extending the series from March 2025 through February 2027. Because these forecasts are recursive, errors can accumulate with lead time; multi-step accuracy is therefore evaluated at every horizon from 1 to 24 months in the rolling-origin backtest (Sections 4.7 and 4.9).

4.6 Baseline forecasting methods

To establish whether the neural architectures add predictive value beyond common forecasting practice, seven families of baseline methods were evaluated under the same protocol as the neural models: the same chronological split, parameters estimated on training months only, and one-step-ahead forecasts over the test period in which each forecast uses only observations available before the target month (equivalent to the sliding-window evaluation of the LSTM, GRU and CNN). The baselines were: (i) persistence (naïve), $\hat{y}_t = y_{t-1}$; (ii) seasonal naïve, $\hat{y}_t = y_{t-12}$; (iii) random walk with drift, $\hat{y}_t = y_{t-1} + c$, where c is the mean monthly change over the training period; (iv) a linear trend fitted by ordinary least squares to the training months and extrapolated; (v) the same linear trend with 11 monthly indicator variables; (vi) ARIMA and seasonal ARIMA (SARIMA, period 12) models with a deterministic linear trend, with orders selected by minimum AIC over $p, q \in \{0, \dots, 3\}$ and $d \in \{0, 1\}$ (ARIMA) and over $p, q \in \{0, 1, 2\}$, $d \in \{0, 1\}$ and $P, Q \in \{0, 1\}$ (SARIMA), retaining only converged fits; and (vii) additive-error exponential smoothing (ETS) state-space models, with the form selected by minimum AICc among ETS(A,N,N), ETS(A,A,N), ETS(A,Ad,N), ETS(A,N,A) and ETS(A,Ad,A) [9].

The selected models were ARIMA(3,0,0) with linear trend, SARIMA(1,0,1)(0,0,1)₁₂ with linear trend, and ETS(A,N,N) (simple exponential smoothing); the seasonal ETS(A,Ad,A) is also reported for comparison. ARIMA and SARIMA parameters were estimated by maximum likelihood (Python statsmodels, SARIMAX) and held fixed over the test period, mirroring the fixed weights of the trained neural networks; as a robustness check they were also re-estimated every month on an expanding window. In

addition to RMSE, MAE and R^2 , the RMSE skill relative to persistence ($1 - \text{RMSE}_{\text{model}} / \text{RMSE}_{\text{persistence}}$) is reported, and differences in squared forecast error were tested with the Diebold–Mariano (DM) test using the Harvey–Leybourne–Newbold small-sample correction (Diebold & Mariano, 1995; Harvey et al., 1997). Twenty-four-month forecasts for March 2025 – February 2027 were produced by refitting each baseline to the full record.

4.7 Rolling-origin evaluation across forecast horizons

Because results from a single train/test split can depend strongly on the test period, all models were additionally evaluated with an expanding-window, rolling-origin design (Hyndman & Athanasopoulos, 2018; Tashman, 2000). Forecast origins were placed every three months from December 2012 to December 2024 (49 origins). At each origin T , every model was re-estimated using only observations up to and including month T . The neural networks were re-trained from scratch with the architectures and training settings of Table 2 (MinMax scaling fitted to the training window, 15% validation split and early stopping; a single seed of 42 for the LSTM, GRU and CNN and the 20-member ensemble for the FFNN), implemented in TensorFlow/Keras. The ARIMA and SARIMA orders and the ETS form were re-selected at every origin by AIC and AICc, respectively, over the candidate sets of Section 4.6, and the simple baselines were recomputed. The first origin leaves 125 months (August 2002 – December 2012) for initial training, or about 100 input windows for the recurrent and CNN models, and the training window grows by three months at each subsequent origin.

From each origin, forecasts were produced for horizons $h = 1$ –24 months and scored at $h = 1, 3, 6, 12$ and 24 months, the horizons relevant to the 24-month outlook of this study. Neural multi-step forecasts were generated recursively, feeding each prediction back as an input, as for the 24-month forecasts in Table 4; the statistical models produce the equivalent iterated forecasts. Because each target y_{T+h} must be observed, the number of scored forecasts declines from 49 at $h = 1$ to 41 at $h = 24$. RMSE, MAE and RMSE skill relative to persistence are reported at each horizon. Equal accuracy against persistence and against ARIMA was tested with the DM test, using a Newey–West variance estimate (Newey & West, 1987) with $\lfloor h/3 \rfloor - 1$ lags to allow for the overlap of multi-step forecast errors from successive origins. To assess sensitivity to the evaluation period, forecast errors were

also grouped by target year into three blocks (2013–2016, 2017–2020 and 2021–2025). As a check on the quarterly spacing of origins, the baseline models were also evaluated with monthly origins (146 origins).

4.8 NNAR benchmark (*nnetar*)

The cited NNAR method was implemented with *nnetar()* from the forecast package (version 8.21.1, R 4.3.3) (Hyndman & Khandakar, 2008; Hyndman et al., 2023) using its default settings, with the series declared as monthly (frequency 12). The *nnetar()* model *NNAR*(p, P, k)₁₂ uses as inputs the unique set of lags $\{1, \dots, p\} \cup \{12P\}$, with $P = 1$ seasonal lag by default; when $p \geq 12$ the seasonal lag is already one of the non-seasonal lags and no input is duplicated. The non-seasonal order p is the order of the linear autoregression selected by AIC for the seasonally adjusted series, and the network has a single hidden layer of $k = \text{round}((\text{number of inputs} + 1)/2)$ logistic units and a linear output. Inputs are centred and scaled, and 20 networks with random starting weights are fitted with *nnet()* and averaged; multi-step forecasts are recursive. *nnetar()* was evaluated in the same way as the other models: (i) on the single split, by fitting to the training months and applying the fitted networks without re-estimation to obtain one-step-ahead test forecasts; (ii) in the rolling-origin design of Section 4.7, by refitting at each of the 49 origins with p re-selected each time; and (iii) for the 24-month forecast, by fitting to the full record, with 95% prediction intervals from 1,000 simulated sample paths. Random seeds were fixed (42 for the single-split and full-record fits, and the origin index for the rolling-origin fits).

4.9 Multi-step evaluation and prediction intervals

Because the 24-month forecasts are produced recursively, forecast errors can accumulate with lead time. Multi-step accuracy was therefore evaluated at every horizon $h = 1$ –24 months in the rolling-origin backtest of Section 4.7, using RMSE and the mean error (observed minus forecast), which reveals systematic drift. Three kinds of uncertainty statement were then compared out of sample. (i) Model-based intervals: Gaussian prediction intervals from the ARIMA model re-estimated at each origin, and the simulated intervals of *nnetar()* (500 sample paths) (Hyndman & Khandakar, 2008). (ii) Empirical intervals for every model, formed by adding the $\alpha/2$ and $1 - \alpha/2$ quantiles of that model's own past h -step errors to its point forecast (Williams & Goodman, 1971). To avoid

look-ahead, the interval for origin T and horizon h used only errors from earlier origins whose targets had already been observed by month T , and it was computed only when at least 10 such errors were available. (iii) The FFNN ensemble spread ($\pm 1\sigma$ and $\pm 1.96\sigma$ across the 20 members), which reflects only variability due to random initialization, evaluated on the single split.

Intervals were assessed by their empirical coverage at nominal levels of 80% and 95%, their mean width, and the interval (Winkler) score, which adds to the interval width a penalty of $2/\alpha$ times the distance by which an observation falls outside the interval (Gneiting & Raftery, 2007); lower scores are better. Methods (i) and (ii) were evaluated on the same origin–horizon pairs (39 at $h = 1$, falling to 24 at $h = 24$, with targets from July 2015 onward). For the 24-month forecasts of Table 4, empirical 80% and 95% intervals were constructed from all rolling-origin errors at the corresponding horizon (41–49 errors). For the neural models these errors come from the re-implemented networks (Section 4.7), and applying them to the point forecasts of Table 4 is an approximation.

4.10 Climate-index covariates

To test whether the error patterns attributed to atmospheric forcing are associated with large-scale circulation, monthly North Atlantic Oscillation (NAO) and Arctic Oscillation (AO) indices were obtained from the NOAA Climate Prediction Center (NOAA Climate Prediction Center, 2026a, 2026b) (accessed 3 October 2026) and aligned with the MSL record. Three analyses were performed. (i) Correlation and least-squares regression of the detrended monthly MSL anomaly (Monthly_MSL minus an OLS trend fitted to the full record; descriptive use only) on each index, for all months and for winter (December–March), using the same month and the previous month; p -values use a heteroscedasticity- and autocorrelation-consistent (HAC) covariance estimate with six lags. (ii) Correlation of the 1-month rolling-origin forecast errors of ARIMA, GRU and persistence with the same-month indices (HAC, two lags). (iii) ARIMA(3,0,0) with trend extended with the indices as exogenous regressors (ARIMAX), evaluated on the single split and, for the AO, at the 49 rolling origins. Same-month indices are not known when a forecast is issued, so models using them are diagnostic and measure how much of the error is associated with concurrent circulation; models using previous-month indices are forecast-feasible. Local wind,

sea-level pressure, storm-surge and ocean-circulation data were not analysed.

5. Results and Discussion

Table 3 summarizes the test-set performance metrics for all four models. Among the four neural models, the GRU achieved the best performance on all three metrics (RMSE = 0.0704 m, MAE = 0.0587 m, $R^2 = 0.099$), followed by the LSTM (RMSE = 0.0734 m, MAE = 0.0627 m, $R^2 = 0.020$). Both recurrent models outperformed the CNN (RMSE = 0.0819 m, $R^2 = -0.219$) and the FFNN ensemble (RMSE = 0.1020 m, $R^2 = -0.907$) by substantial margins.

The universally low R^2 values across all models reflect the fundamental challenge of this dataset: monthly MSL at Ocean City exhibits high stochastic variability ($\sigma \approx 0.079$ m) relative to the long-term trend signal, which is likely influenced by meteorological forcing, shelf-scale ocean dynamics and episodic storm events; only the association with large-scale climate indices is tested here (Section 5.6). In such a regime, a model that perfectly captures the trend but fails to reproduce individual monthly anomalies will still exhibit low R^2 . The GRU's R^2 of 0.099 indicates that only about 10% of the test-period variance is explained, so roughly 90% of the month-to-month variation remains unexplained. Its RMSE (0.0704 m) is only 5% below the standard deviation of the test-period observations (0.0742 m) and 3% below the RMSE of a persistence forecast (0.0727 m; Table 5), and in the rolling-origin evaluation its improvement over persistence was not statistically significant at any horizon (Section 5.2). The GRU result should therefore be read as the best of the four neural models on one test period at one station, not as evidence of skilful monthly MSL prediction.

The GRU's advantage over the LSTM is consistent with the suggestion that GRUs, which have fewer parameters, may generalize better on small datasets (Chung et al., 2014). With only about 200 training windows, the LSTM's additional parameters may introduce marginal overfitting. Notably, early stopping ended GRU training at epoch 24, whereas the LSTM ran for the full 200 epochs, suggesting faster convergence. This difference should not be over-interpreted: across five random seeds, the re-implemented GRU and LSTM had overlapping RMSE ranges (0.0674–0.0779 m and 0.0733–0.0960 m; Table 8), and in the rolling-origin evaluation the LSTM was more accurate than the GRU at the 6-, 12- and 24-month horizons (Table

Table 3. Test-set performance metrics for all four models.

Metric	LSTM	GRU	CNN	FFNN
RMSE (m)	0.0734	0.0704	0.0819	0.1020
MAE (m)	0.0627	0.0587	0.0673	0.0838
R ²	0.020	0.099	-0.219	-0.907
Rank	2nd	1st	3rd	4th

6). The CNN's negative R² (-0.219) suggests that, on this test period, local temporal feature extraction without a recurrent hidden state was less effective than the recurrent models. The result is period-dependent, however: in the rolling-origin evaluation the CNN ranked first at the 1-month horizon in 2017–2020 (Table 7). The CNN applies fixed-length convolutional filters that detect patterns within the input window but cannot maintain context across

prediction steps. This limitation is most apparent in the test period (January 2021 to February 2025), which includes several above-normal sea level years (2020–2024) not well-represented in the earlier training period. The FFNN ensemble's strongly negative R² (-0.907) is somewhat surprising given the model's reasonable performance documented in the forecasting literature for other types of time series (Hyndman & Athanasopoulos, 2018; Zhang et al., 1998). The result likely reflects two

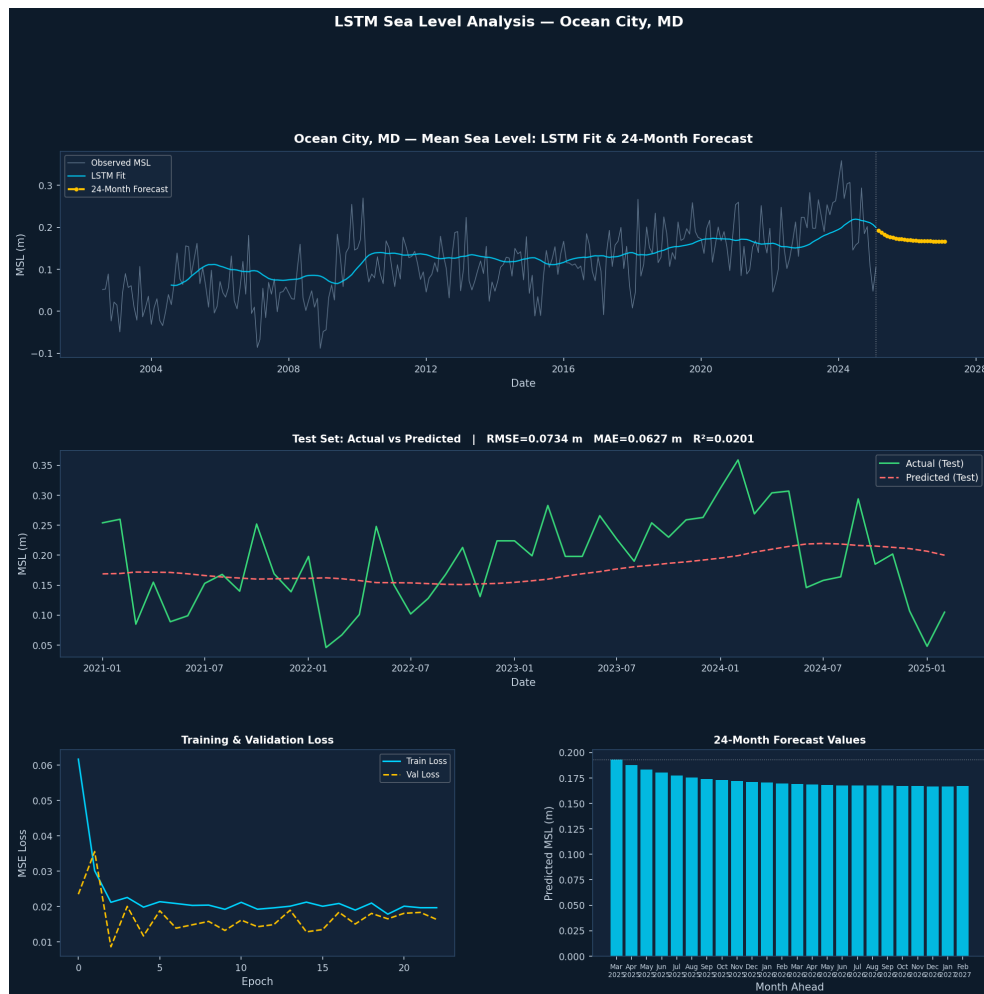


Figure 3. LSTM model results: (a) full MSL record with model fit and 24-month forecast; (b) test-set actual vs. predicted; (c) training/ validation loss; (d) 24-month forecast values. Early stopping did not trigger; training ran to epoch 200.

compounding factors: (1) the feedforward MLP cannot maintain a recurrent state across predictions, forcing reliance on a limited lag window; and (2) the high interannual variability of the series means that lagged observations from 1–13 months prior have limited predictive power for the current month's anomaly if, as seems likely but was not tested directly, much of the month-to-month variability is

driven by synoptic atmospheric forcing.

Figures 3–6 present the full diagnostic output for each model, comprising (a) the complete observed record with model fit and 24-month forecast, (b) test-set actual versus predicted values, (c) training and validation loss curves, and (d) the 24-month forecast bar chart.

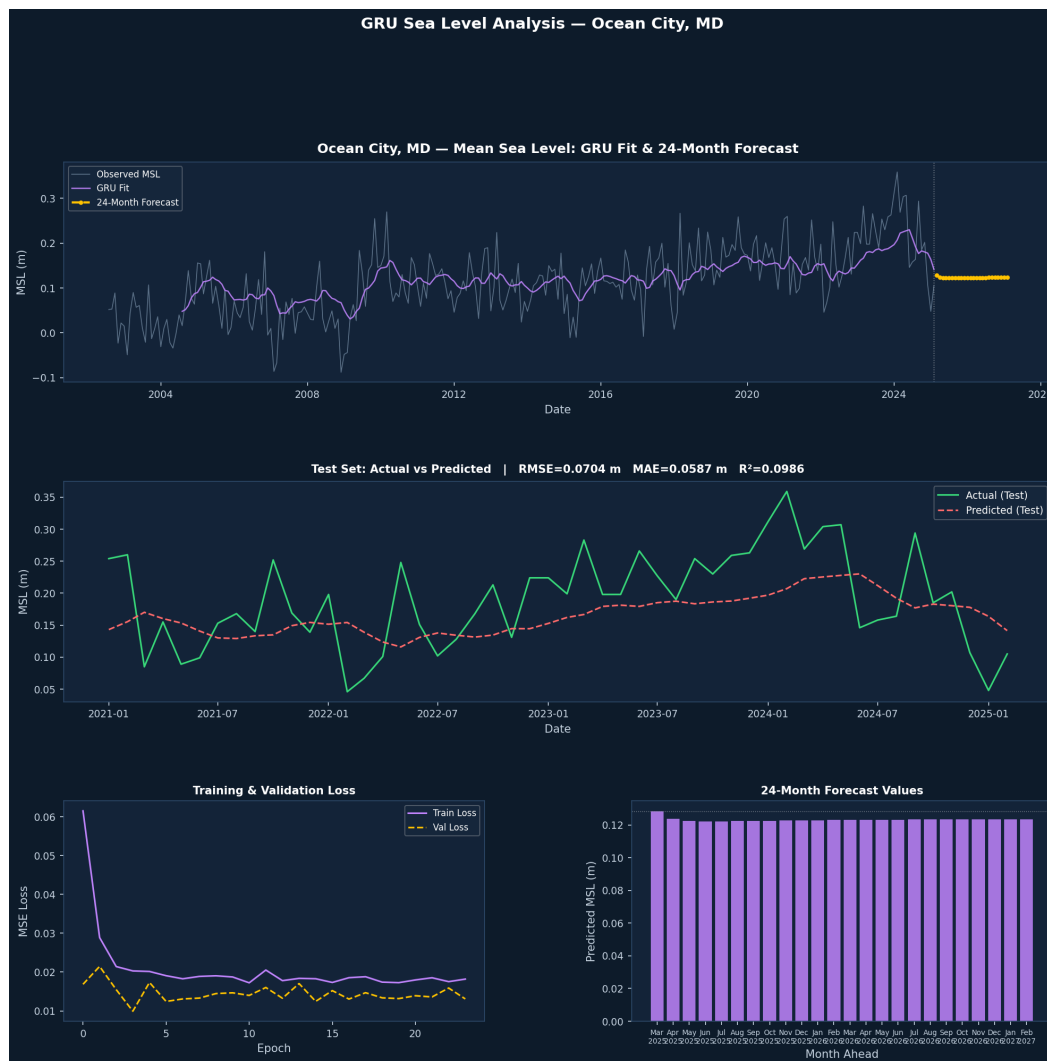


Figure 4. GRU model results: (a) full MSL record with model fit and 24-month forecast; (b) test-set actual vs. predicted; (c) training/validation loss (converged at epoch 24); (d) 24-month forecast values. The GRU achieved the best test-set performance across all metrics.

The training loss curves (panel c in each figure) are instructive. The GRU and CNN both exhibit clean convergence with validation loss tracking training loss closely, indicating adequate regularization. The LSTM's validation loss is slightly more erratic but ultimately stabilizes. The FFNN ensemble loss curves, averaged across 20 members, show rapid initial descent followed by

early stopping, consistent with the sigmoid-activated MLP saturating quickly on a small, normalized dataset.

The test-set actual versus predicted plots (panel b) reveal that all models capture the general level of MSL during the test period but struggle with individual anomalous months, particularly the high-MSL months of January–February 2021 (October 2019, another high month, lies in

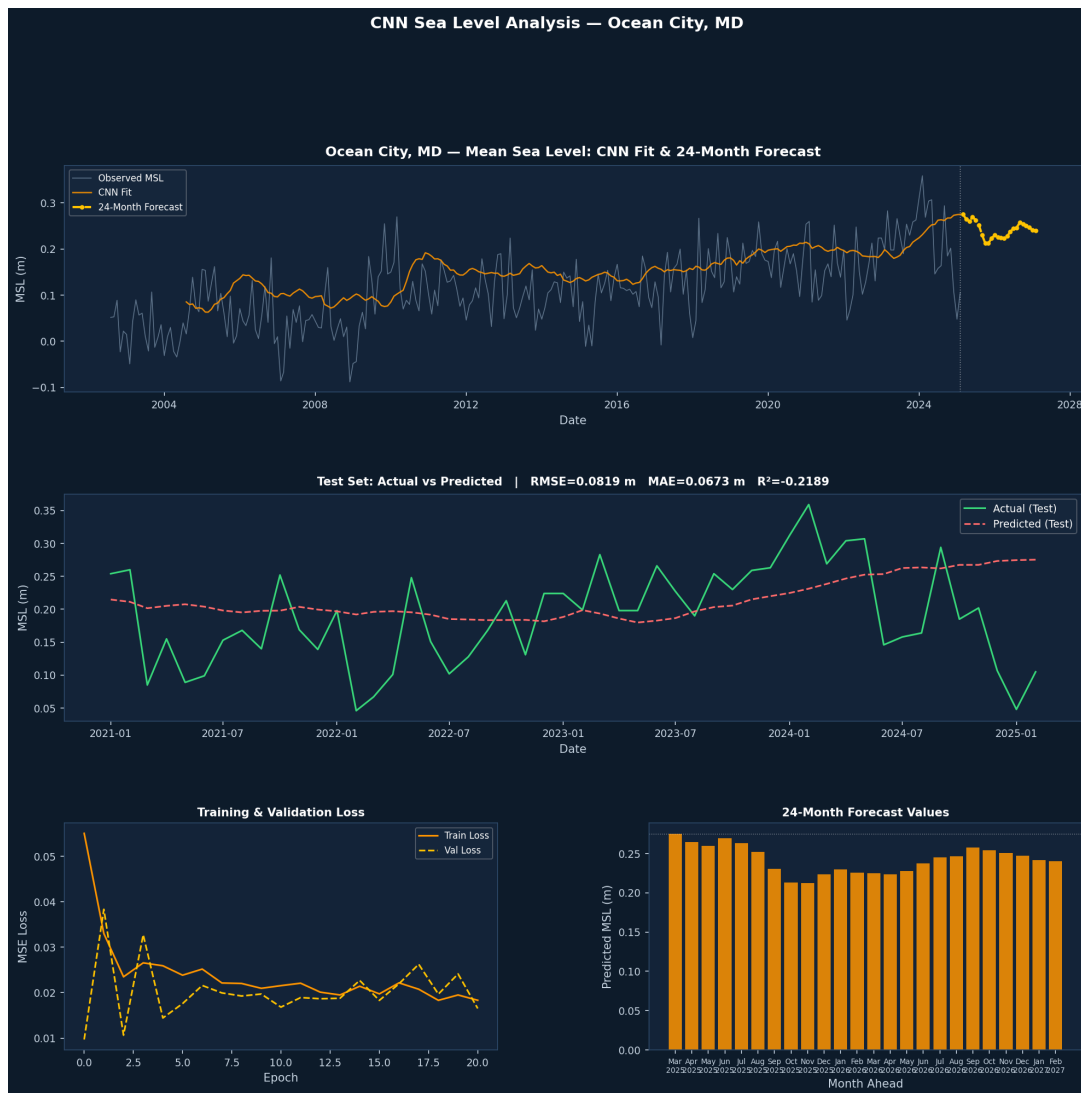


Figure 5. CNN model results: (a) full MSL record with model fit and 24-month forecast; (b) test-set actual vs. predicted; (c) training/ validation loss; (d) 24-month forecast values. The oscillatory forecast pattern reflects overfitting to short-term cyclical features.

the training period). January 2021 coincided with strongly negative phases of the Arctic Oscillation ($AO = -2.48$) and North Atlantic Oscillation ($NAO = -1.11$), and October 2019 with a negative NAO (-1.41). This is consistent with wind-driven coastal setup and storm activity, which are known to force sea level along this coast (Andres et al., 2013), but local wind, pressure and storm-surge data were not analysed, so these mechanisms remain possible explanations rather than demonstrated causes. Section 5.6 tests the broader association between forecast errors and climate indices. Anomalies of this kind are unlikely to be predictable from the MSL series alone, which is a limitation of all purely univariate approaches, statistical as

well as neural.

Table 4 summarizes the 24-month forecast outputs for all models, grouped by six-month periods. The LSTM projects a gradual but stable MSL plateau between approximately 0.167 and 0.183 m, consistent with the recent observed trend. The GRU projects a lower plateau of approximately 0.122–0.124 m. The CNN forecast displays more oscillatory behavior (0.211–0.275 m), overshooting the recent observed values, likely because of the CNN’s tendency to extrapolate short-term patterns from its receptive field. The FFNN ensemble projects the lowest values (approximately 0.108–0.113 m) with a $\pm 1\sigma$ uncertainty range of roughly ± 0.026 m, reflecting ensemble spread.

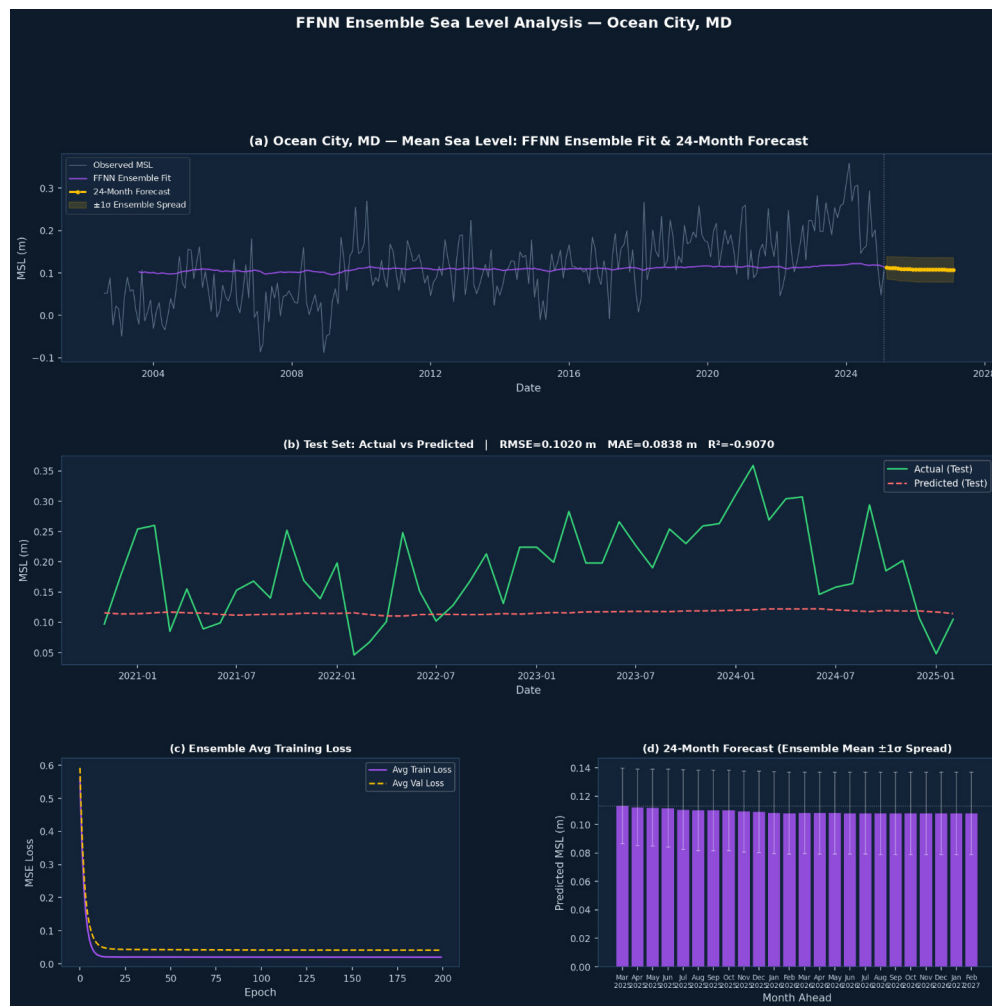


Figure 6. FFNN ensemble ($n = 20$; lags 1–13; 32–16–8 hidden units) results: (a) full MSL record with ensemble fit, 24-month forecast, and $\pm 1\sigma$ ensemble spread (variation across 20 initializations; not a prediction interval); (b) test-set actual vs. predicted; (c) ensemble-averaged training/validation loss; (d) 24-month forecast with $\pm 1\sigma$ ensemble-spread bars.

Table 4. Summary of 24-month forecast mean sea level (m) by six-month periods, March 2025 – February 2027.

Period	LSTM (m)	GRU (m)	CNN (m)	FFNN (m)
Mar–Aug 2025	0.183–0.177	0.128–0.122	0.275–0.252	0.113–0.110
Sep 2025–Feb 2026	0.174–0.170	0.122–0.123	0.231–0.225	0.110–0.108
Mar–Aug 2026	0.169–0.168	0.123–0.123	0.225–0.246	0.108–0.108
Sep 2026–Feb 2027	0.168–0.167	0.123–0.124	0.257–0.240	0.108–0.108
24-Month Mean	0.171	0.123	0.242	0.109

The divergence in forecast level across models (ranging from ~ 0.110 m for FFNN to ~ 0.242 m for CNN) underscores the sensitivity of data-driven forecasts to architecture choice. All forecasts are levels in meters relative to the station MSL datum. For comparison, the NOAA trend line reaches 0.176 m in February 2025 and,

extended at its 5.3 mm/yr rate, about 0.187 m by February 2027. The LSTM forecast (0.167–0.183 m) is closest to a continuation of this trend, whereas the GRU (0.122–0.124 m) and FFNN (0.108–0.113 m) forecasts lie well below it. CNN overpredicts near-term MSL, while the FFNN and GRU forecasts appear to revert toward a historical mean

rather than extrapolating the trend, which may reflect the models' limited effective memory horizon.

Observations published by NOAA since the original analysis allow a first check of these forecasts. For March 2025 – August 2026 (18 months), the RMSE of the ARIMA forecast was 0.075 m, and 16 of the 18 observations fell within its 95% prediction interval, compared with 1 of 18 for the simulated `nnetar()` interval. For the six months in this period whose values are listed in Table 4, the RMSE was 0.085 m for the CNN, 0.086 m for the LSTM, 0.119 m for the GRU and 0.130 m for the FFNN, compared with 0.074 m for ARIMA (Supplementary Table S1). Six months are too few for firm conclusions, and NOAA's current values differ from the archived data by a few millimetres.

For context, the GRU forecast for 2025–2027 projects MSL values approximately 0.12 m above the station MSL datum (about 0.07 m above the August 2002 value of the NOAA trend line, 0.057 m). The LSTM forecast is broadly consistent with this assessment. Both models' forecasts should be interpreted as data-driven statistical extrapolations rather than physically based projections and ideally would be combined with NOAA official sea level rise scenarios for risk management applications.

The FFNN ensemble provides a spread ($\pm 1\sigma$ across 20 independently initialized networks), which represents only variability due to random initialization. The ensemble spread of approximately ± 0.026 m across the 24-month horizon is relatively narrow compared to the observed monthly variability ($\sigma \approx 0.079$ m), indicating that ensemble spread underestimates total predictive uncertainty. Full predictive intervals would require additional treatment of observation noise, structural model uncertainty, and climate variability, ideally through Bayesian neural network approaches or conformal prediction methods. Calibrated prediction intervals are evaluated in Section 5.5.

5.1 Comparison with forecasting baselines

Table 5 and Figure 7 compare the neural models with the baselines on the identical 50-month test period. Three findings stand out. First, all three statistical time-series models outperform every neural architecture. ARIMA(3,0,0) with trend achieved the lowest error of any model (RMSE = 0.0642 m, MAE = 0.0528 m, $R^2 = 0.250$), followed by SARIMA (RMSE = 0.0657 m, $R^2 = 0.217$) and ETS(A,N,N) (RMSE = 0.0662 m, $R^2 = 0.204$). The ARIMA RMSE is 8.8% lower than that of the GRU (0.0704 m), and its R^2 is about 2.5 times larger. Re-estimating the ARIMA and SARIMA parameters every month changed

their RMSE by less than 0.001 m, so this result does not depend on the fixed-parameter design.

Second, persistence is a demanding benchmark for this series. The naïve forecast (RMSE = 0.0727 m) is practically indistinguishable from random walk with drift (0.0727 m) and the extrapolated linear trend (0.0732 m), and the GRU improves on it by only 3.2% in RMSE. The LSTM (0.0734 m) is marginally worse than persistence, and the CNN (0.0819 m; skill = -0.13) and FFNN (0.1020 m; skill = -0.40) are substantially worse; the FFNN error is close to that of the seasonal-naïve forecast (0.1041 m), the weakest baseline. The seasonal-naïve result also shows that the mean annual cycle at this station is weak relative to interannual and month-to-month variability: adding monthly indicators to the linear trend, or a seasonal component to ETS, slightly reduced accuracy.

Third, the margins among the better-performing models are small relative to the sampling variability of a 50-month test set. The DM test did not reject equal accuracy between ARIMA and persistence ($p = 0.143$), ETS(A,N,N) ($p = 0.534$) or SARIMA ($p = 0.355$); only the seasonal-naïve forecast was significantly less accurate than both persistence ($p = 0.016$) and ARIMA ($p = 0.001$).

For the neural models, DM tests were computed from the test-set predictions of the re-implemented GRU and LSTM trained with the original seed (42; Section 5.3), whose RMSEs (0.0674 and 0.0743 m) are close to those in Table 3 (0.0704 and 0.0734 m). Neither model differed significantly from persistence (GRU $p = 0.44$; LSTM $p = 0.85$). The GRU was not significantly less accurate than ARIMA ($p = 0.23$), whereas the LSTM was ($p = 0.018$). Across five random seeds, the GRU never differed significantly from persistence ($p \geq 0.44$) and was significantly less accurate than ARIMA for only one seed ($p = 0.008$; otherwise, 0.10–0.25), while the LSTM was significantly less accurate than ARIMA for all five seeds ($p = 0.0005$ –0.033).

The baselines also put the 24-month forecasts in Table 4 in context. Refitted to the full record, the trend-bearing models (ARIMA, SARIMA and the OLS trend) project a 24-month mean MSL of 0.199–0.210 m, rising to about 0.213–0.217 m by February 2027, consistent with continuation of the station's linear trend. In contrast, level-only methods remain close to the most recent observations (persistence 0.105 m; ETS(A,N,N) 0.128 m). The GRU (0.123 m) and FFNN (0.109 m) forecasts therefore behave like level-only forecasts anchored to the low values of late

Table 5. Test-set performance (January 2021 – February 2025, $n = 50$, one-step-ahead) of the baseline methods and the four neural models, ordered by RMSE.

Model	Type	RMSE (m)	MAE (m)	R ²	Skill vs persistence	DM p vs persistence	DM p vs ARIMA
ARIMA(3,0,0) + trend	Statistical	0.0642	0.0528	0.250	0.116	0.143	—
ARIMA(3,0,0) + trend, re-estimated monthly	Statistical	0.0649	0.0530	0.235	0.108	0.153	0.334
SARIMA(1,0,1)(0,0,1) ₁₂ + trend	Statistical	0.0657	0.0536	0.217	0.097	0.254	0.355
SARIMA(1,0,1)(0,0,1) ₁₂ + trend, re-estimated monthly	Statistical	0.0658	0.0544	0.213	0.095	0.247	0.241
ETS(A,N,N) (AICc-selected)	Statistical	0.0662	0.0528	0.204	0.090	0.221	0.534
ETS(A,Ad,A)	Statistical	0.0682	0.0548	0.154	0.061	0.451	0.314
GRU	Neural	0.0704	0.0587	0.099	0.032	†	†
Persistence (naïve)	Simple	0.0727	0.0580	0.039	0.000	—	0.143
Random walk with drift	Simple	0.0727	0.0579	0.039	0.000	0.869	0.145
Linear trend (OLS)	Simple	0.0732	0.0606	0.026	−0.007	0.953	0.050
LSTM	Neural	0.0734	0.0627	0.020	−0.010	†	†
Linear trend + monthly dummies	Simple	0.0747	0.0622	−0.014	−0.028	0.839	0.086
CNN	Neural	0.0819	0.0673	−0.219	−0.126	†	†
FFNN	Neural	0.1020	0.0838	−0.907	−0.403	†	†
Seasonal naïve (lag 12)	Simple	0.1041	0.0846	−0.969	−0.432	0.016	0.001

Note. Skill = $1 - \text{RMSE}/\text{RMSE}_{\text{persistence}}$. DM p = two-sided Diebold–Mariano p-value (squared-error loss, Harvey–Leybourne–Newbold correction). Neural-model metrics are those reported in Table 3. Best value in bold. † DM tests for the neural models require their test-set predictions.

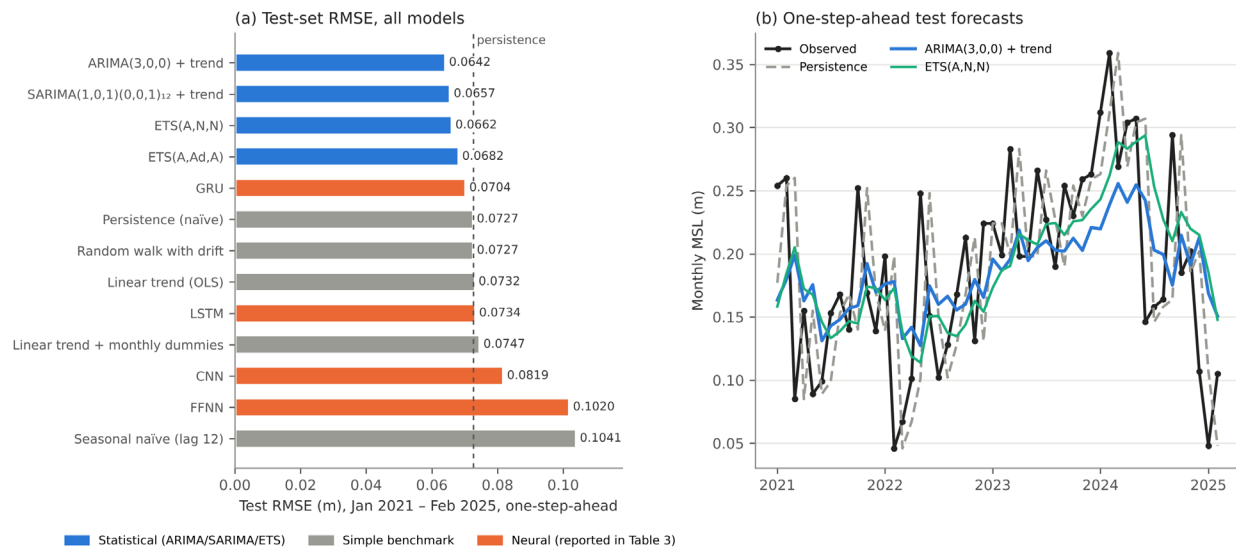


Figure 7. Comparison of neural and baseline forecasts on the common test period (January 2021 – February 2025): (a) one-step-ahead test RMSE for all models, with the persistence benchmark shown as a dashed line (neural values from Table 3); (b) observed monthly MSL with persistence, ARIMA(3,0,0) + trend and ETS(A,N,N) one-step-ahead forecasts.

2024 and early 2025, not like extrapolations of the long-term trend, while the LSTM (0.171 m) lies between the two groups. The ARIMA 95% prediction interval widens from 0.040–0.256 m in March 2025 to 0.096–0.334 m in February 2027 and contains all four neural forecasts at the 24-month horizon, which shows that the differences among the models' point forecasts are small compared with the intrinsic predictive uncertainty of monthly MSL.

Taken together, these results indicate that on this univariate monthly record the neural architectures do not add predictive value beyond well-specified linear time-series models. ARIMA(3,0,0) with trend represents the two components the data support, a deterministic secular rise and short-memory autocorrelation (AR coefficients 0.28, 0.05 and 0.17), with six parameters, whereas the LSTM and GRU estimate thousands of weights from fewer than 200 training windows. The GRU therefore ranks first among the neural architectures in Table 3, but it is not the most accurate model overall.

5.2 Rolling-origin performance by forecast horizon

Table 6 and Figure 8 summarise the rolling-origin results. Pooled over all origins, ARIMA had the lowest RMSE at $h = 1, 3, 6$ and 24 months (0.0558, 0.0639, 0.0646 and 0.0687 m), and at $h = 12$ it was effectively tied with the linear trend (0.0667 vs 0.0665 m). Its RMSE skill relative to persistence ranged from 0.19 to 0.30 and increased with horizon, because persistence, unlike the

trend-bearing models, cannot follow the secular rise; the improvement over persistence was significant at $h = 1, 3, 12$ and 24 months ($p \leq 0.038$) but not at $h = 6$ ($p = 0.075$). The ETS(A,N,N) model, selected at every origin, was competitive at $h = 1$ but deteriorated with horizon because it extrapolates a constant level. The seasonal-naïve forecast and random walk with drift were no better than persistence at any horizon.

None of the re-trained neural models outperformed ARIMA at any horizon. The best neural model was the GRU at $h = 1$ and 3 months (0.0590 and 0.0686 m), the CNN at $h = 6$ months (0.0739 m) and the LSTM at $h = 12$ and 24 months (0.0769 and 0.0777 m), corresponding to RMSE values 6–15% higher than those of ARIMA. These differences were statistically significant for the LSTM at $h = 3, 6$ and 12 ($p \leq 0.037$), for the CNN at $h = 12$ and 24 ($p \leq 0.013$) and for the FFNN at every horizon ($p \leq 0.028$), but not for the GRU ($p = 0.065$ –0.43). The GRU improved on persistence at every horizon (skill 0.04–0.20), but none of these improvements was significant ($p \geq 0.055$), and the FFNN was no better than persistence at any horizon (skill –0.15 to 0.01). The recursive CNN forecasts degraded most at long lead times (RMSE 0.0935 m at $h = 24$). For the 24-month horizon emphasised in Table 4, ARIMA was 0.009–0.029 m more accurate in RMSE than the four neural models.

Table 6. Expanding-window rolling-origin RMSE (m) by forecast horizon h (months), with RMSE skill relative to persistence in parentheses.

Model	$h = 1$	$h = 3$	$h = 6$	$h = 12$	$h = 24$
Scored forecasts (n)	49	48	47	45	41
Persistence (naïve)	0.0701†	0.0796†	0.0794	0.0919†	0.0978†
Seasonal naïve	0.0772 (–0.10)†	0.0928 (–0.17)†	0.0917 (–0.15)†	0.0919 (+0.00)†	0.0978 (+0.00)†
Random walk with drift	0.0704 (–0.00)†	0.0803 (–0.01)†	0.0809 (–0.02)	0.0948 (–0.03)†	0.1019 (–0.04)†
Linear trend (OLS)	0.0582 (+0.17)	0.0657 (+0.17)	0.0652 (+0.18)	0.0665 (+0.28)*	0.0689 (+0.30)*
ARIMA + trend (AIC)	0.0558 (+0.20)*	0.0639 (+0.20)*	0.0646 (+0.19)	0.0667 (+0.27)*	0.0687 (+0.30)*
SARIMA + trend (AIC)	0.0566 (+0.19)	0.0644 (+0.19)*	0.0675 (+0.15)	0.0708 (+0.23)*	0.0692 (+0.29)*
ETS (AICc)	0.0566 (+0.19)	0.0663 (+0.17)	0.0700 (+0.12)	0.0762 (+0.17)*†	0.0797 (+0.19)*
GRU	0.0590 (+0.16)	0.0686 (+0.14)	0.0762 (+0.04)	0.0777 (+0.15)	0.0784 (+0.20)
LSTM	0.0600 (+0.14)	0.0733 (+0.08)†	0.0750 (+0.06)†	0.0769 (+0.16)*†	0.0777 (+0.21)*
CNN	0.0624 (+0.11)	0.0718 (+0.10)	0.0739 (+0.07)	0.0809 (+0.12)†	0.0935 (+0.04)†
FFNN (20-member)	0.0808 (–0.15)†	0.0883 (–0.11)†	0.0876 (–0.10)†	0.0915 (+0.00)†	0.0977 (+0.00)†

*Note. Origins every three months, December 2012 – December 2024; all models re-estimated or re-trained at every origin. Best value per horizon in bold. * significantly more accurate than persistence; † significantly less accurate than ARIMA (DM test, $p < 0.05$).*

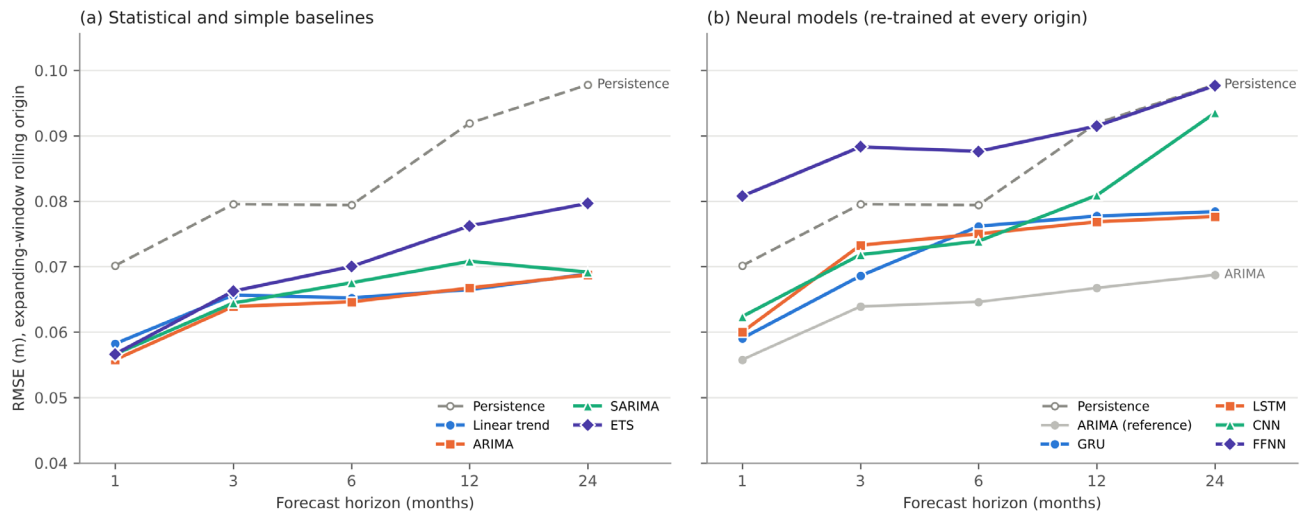


Figure 8. Expanding-window rolling-origin RMSE by forecast horizon for (a) the statistical and simple baselines and (b) the neural models re-trained at every origin; persistence (dashed) and ARIMA (grey, panel b) are shown for reference. Seasonal naïve and random walk with drift are omitted for clarity (see Table 6).

Table 7. RMSE (m) and rank among the 11 models (in parentheses) by target-year block at $h = 1, 6$ and 12 months.

Model	h=1: 2013– 2016	h=1: 2017– 2020	h=1: 2021– 2025	h=6: 2013– 2016	h=6: 2017– 2020	h=6: 2021– 2025	h=12: 2013– 2016	h=12: 2017– 2020	h=12: 2021– 2025
Persistence (naïve)	0.069 (10)	0.079 (10)	0.061 (3)	0.081 (9)	0.088 (8)	0.068 (1)	0.082 (9)	0.098 (8)	0.094 (6)
Seasonal naïve	0.049 (4)	0.079 (9)	0.095 (10)	0.082 (10)	0.098 (11)	0.094 (9)	0.082 (9)	0.098 (8)	0.094 (6)
Random walk with drift	0.070 (11)	0.080 (11)	0.061 (4)	0.083 (11)	0.089 (9)	0.069 (2)	0.087 (11)	0.100 (11)	0.096 (9)
Linear trend (OLS)	0.048 (2)	0.051 (3)	0.072 (7)	0.051 (3)	0.069 (1)	0.073 (4)	0.052 (2)	0.069 (1)	0.074 (2)
ARIMA + trend (AIC)	0.053 (6)	0.055 (6)	0.059 (1)	0.052 (4)	0.069 (2)	0.071 (3)	0.052 (3)	0.070 (2)	0.074 (1)
SARIMA + trend (AIC)	0.053 (7)	0.055 (7)	0.061 (2)	0.051 (1)	0.071 (6)	0.077 (6)	0.051 (1)	0.073 (3)	0.082 (3)
ETS (AICc)	0.055 (9)	0.050 (2)	0.064 (5)	0.063 (8)	0.070 (3)	0.076 (5)	0.060 (8)	0.075 (5)	0.088 (4)
GRU	0.049 (5)	0.054 (5)	0.071 (6)	0.055 (6)	0.070 (4)	0.097 (10)	0.054 (5)	0.075 (4)	0.095 (8)
LSTM	0.047 (1)	0.052 (4)	0.075 (8)	0.056 (7)	0.075 (7)	0.089 (7)	0.053 (4)	0.079 (7)	0.090 (5)
CNN	0.054 (8)	0.044 (1)	0.082 (9)	0.054 (5)	0.071 (5)	0.091 (8)	0.055 (7)	0.077 (6)	0.100 (10)
FFNN (20-member)	0.048 (3)	0.075 (8)	0.107 (11)	0.051 (2)	0.096 (10)	0.104 (11)	0.054 (6)	0.098 (10)	0.107 (11)

Note. Each block contains 13–17 scored forecasts. Shaded cells mark the top three models in each block; rank 1 in bold.

Table 7 shows that the ranking of models depends strongly on the evaluation period, as the reviewer anticipated. At $h = 1$ month the top-ranked model differed in each block: the LSTM in 2013–2016 (0.047 m), the CNN in 2017–2020 (0.044 m) and ARIMA in 2021–2025 (0.059 m). In the most recent block, which approximately coincides with the test period of the single 80/20 split, all four neural models ranked sixth or lower, and at $h = 6$ even persistence ranked first, reflecting the sharp fall of the series after its early-2024 peak. Neural rankings were the least stable: at $h = 1$ the CNN moved from first (2017–2020) to ninth (2021–2025) and the FFNN from third (2013–2016) to last (2021–2025). In contrast, at the 12-month horizon the three trend-bearing statistical models (linear trend, ARIMA and SARIMA) occupied the top three ranks in every block. A single split could therefore support different conclusions about the neural architectures depending on where it is placed, whereas pooled rolling-origin results consistently favour the trend-bearing statistical models. As a consistency check on the re-implementation, the re-trained GRU had an RMSE of 0.071 m at $h = 1$ in the 2021–2025 block, close to the 0.0704 m reported for the single split (Table 3).

Evaluating the baselines with monthly rather than quarterly origins (146 origins) changed their RMSE by at most 0.010 m and left their ordering unchanged apart from a near-tie between ARIMA and the linear trend at $h = 24$ (0.0654 vs 0.0652 m), indicating that the quarterly spacing of origins does not bias the comparison. Across all origins, ARIMA(3,0,0) was the most frequently selected non-seasonal order (26 of 49 origins).

5.3 Sensitivity to the modelling target

To determine whether the choice of modelling target affects the conclusions, the four networks were re-trained on the single split (training through December 2020, test January 2021 – February 2025) with two targets: (i) the seasonally adjusted MSL level, as in the main analysis; and (ii) residuals from an OLS linear trend fitted to the training months only, $z_t = y_t - (\beta_0 + \beta_1 t)$, with the same training-only min–max scaling applied to z_t . For target (ii), the trend extrapolated with the training coefficients was added back to the network output, $\hat{y}_t = \hat{z}_t + \beta_0 + \beta_1 t$, so that predictions, metrics and forecasts remain in meters on the level scale and no test-period information enters the trend. The training-period trend was 7.4 mm/yr. The LSTM, GRU and CNN were each trained with five random seeds (42, 1,

2, 3 and 4), and the FFNN as the 20-member ensemble, using the re-implementation described in Section 4.7. For the 24-month forecasts, each model (and, for target (ii), the trend) was refitted to the full record.

Level-trained networks closely reproduced the single-split results in Table 3 (GRU mean RMSE 0.0701 m vs 0.0704 m reported). However, their accuracy depended strongly on the random seed (RMSE 0.0674–0.0779 m for the GRU and 0.0733–0.0960 m for the LSTM), and their 24-month mean forecasts ranged from 0.117 to 0.243 m across models and seeds (Table 8). Detrending before training improved every architecture. Mean test RMSE fell to 0.0668 m for the GRU, 0.0712 m for the LSTM, 0.0751 m for the CNN and 0.0726 m for the FFNN (from 0.1119 m). Seed-to-seed variability fell by a factor of about 3–11, and the 24-month mean forecasts converged on 0.198–0.227 m, in line with the trend-bearing statistical models (0.199–0.210 m; Section 5.1). Detrending therefore removes the tendency of the level-trained GRU and FFNN forecasts to fall back toward earlier levels, which arises because those networks must extrapolate beyond the scaled training range to follow the rise. Even with detrending, however, no network outperformed ARIMA with trend (0.0642 m), SARIMA (0.0657 m) or ETS (0.0662 m) on the same test period, so the conclusions of Sections 5.1 and 5.2 are unchanged. The main analysis retains the level target to remain consistent with the original experiments; removing a training-period trend before training may be preferable in future applications of these architectures, although this was tested only at this station.

5.4 Comparison with the *nnetar()* NNAR benchmark

On the single split, *nnetar()* selected NNAR(16,1,8)12: lags 1–16 (which include lag 12), eight hidden units and 145 weights per network. Its test RMSE was 0.0708 m (MAE 0.0585 m; $R^2 = 0.089$), much lower than that of the three-hidden-layer FFNN (0.1020 m) and essentially equal to that of the GRU (0.0704 m). It did not, however, outperform persistence (DM $p = 0.81$) or ARIMA (0.0642 m; $p = 0.17$). Its in-sample RMSE (0.0077 m) was an order of magnitude smaller than its test RMSE, which indicates strong overfitting of 145-weight networks to about 200 training months.

In the rolling-origin evaluation (Table 9), *nnetar()* had the lowest RMSE of all models at $h = 1$ (0.0550 m vs 0.0558 m for ARIMA; DM $p = 0.86$) but deteriorated rapidly with horizon: 0.0718 m at $h = 3$, 0.0785 m at $h = 6$ and 0.1054 m at $h = 12$, worse than persistence (0.0919 m), before

Table 8. Single-split test performance (January 2021 – February 2025, $n = 50$, one-step-ahead) and 24-month mean forecasts (March 2025 – February 2027) of the re-implemented neural models trained on the MSL level or on residuals from a training-period OLS trend.

Model	Training target	Test RMSE (m), mean $\pm$ SD	RMSE range (m)	R ² (mean)	24-month mean forecast (m)	Reported in Table 3 (RMSE)
GRU	Level (as in main analysis)	0.0701 $\pm$ 0.0044	0.0674–0.0779	0.104	0.117–0.185	0.0704
GRU	Residual from training-period OLS trend	0.0668 $\pm$ 0.0004	0.0663–0.0673	0.188	0.210–0.227	
LSTM	Level (as in main analysis)	0.0804 $\pm$ 0.0092	0.0733–0.0960	–0.186	0.165–0.243	0.0734
LSTM	Residual from training-period OLS trend	0.0712 $\pm$ 0.0016	0.0693–0.0732	0.079	0.198–0.208	
CNN	Level (as in main analysis)	0.0804 $\pm$ 0.0031	0.0768–0.0847	–0.175	0.173–0.230	0.0819
CNN	Residual from training-period OLS trend	0.0751 $\pm$ 0.0011	0.0739–0.0765	–0.025	0.198–0.221	
FFNN	Level (as in main analysis)	0.1119 (ensemble)	—	–1.274	0.126	0.1020
FFNN	Residual from training-period OLS trend	0.0726 (ensemble)	—	0.043	0.211	
ARIMA	Level, deterministic trend term	0.0642	—	0.250	0.199	

Note. All values are in meters on the level scale. LSTM, GRU and CNN: five seeds; FFNN: 20-member ensemble. ARIMA from Table 5 for reference.

Table 9. Comparison of the `nnetar()` NNAR benchmark with the FFNN, the GRU and two baselines.

Model	Single split RMSE (m)	Single split R ²	Rolling RMSE, h = 1	Rolling RMSE, h = 3	Rolling RMSE, h = 6	Rolling RMSE, h = 12	Rolling RMSE, h = 24
Persistence (naïve)	0.0727	0.039	0.0701	0.0796	0.0794	0.0919	0.0978
ARIMA + trend	0.0642	0.250	0.0558	0.0639	0.0646	0.0667	0.0687
GRU	0.0704	0.099	0.0590	0.0686	0.0762	0.0777	0.0784
FFNN, original (14 inputs)	0.1020	–0.907	0.0808	0.0883	0.0876	0.0915	0.0977
<code>nnetar()</code> NNAR	0.0708	0.089	0.0550	0.0718	0.0785	0.1054	0.0779

Note. Single split: test period January 2021 – February 2025 ($n = 50$), one-step-ahead; values from Tables 3 and 5. Rolling origin: expanding window with 49 origins (Section 4.7), RMSE in meters by horizon h (months). Best value per column in bold.

0.0779 m at $h = 24$. It was significantly less accurate than ARIMA at $h = 6$ ($p = 0.027$). The automatically selected order was unstable across origins: NNAR(3,1,2)12 at 17 origins, NNAR(4,1,3)12 at 7, NNAR(14,1,8)12 or NNAR(16,1,8)12 at 20 and NNAR(24,1,12)12 at 5, so the hidden layer ranged from 2 to 12 units, and recursive multi-step forecasts from the larger networks were erratic. Fitted to the full record, `nnetar()` selected NNAR(24,1,12)12

and projected a 24-month mean MSL of 0.175 m, rising from 0.166 m in March 2025 to 0.223 m in February 2027. Its simulated 95% prediction interval was only 0.162–0.170 m in March 2025 and 0.216–0.230 m in February 2027. This is far narrower than the observed month-to-month variability ($\sigma \approx 0.079$ m) and the ARIMA interval (0.040–0.256 m in March 2025), because the simulation resamples the overfitted in-sample residuals; these intervals should

therefore not be used as a measure of forecast uncertainty. Removing the duplicate lag-12 input from the FFNN (13 distinct lags instead of 14 inputs) left its single-split performance unchanged in the re-implementation (RMSE 0.1119 m in both configurations; 24-month mean forecast 0.126 m in both). The poor accuracy of the FFNN therefore stems from its architecture and training rather than from the duplicated lag, and the contrast with `nnetar()` shows that the FFNN should not be taken as representative of the NNAR method. Neither neural autoregression outperformed ARIMA, so the conclusions of Sections 5.1–5.3 are unchanged.

5.5 Multi-step error growth and prediction intervals

Figure 9a–b shows how forecast errors develop with lead time in the rolling-origin backtest. ARIMA errors grew slowly, from an RMSE of 0.056 m at $h = 1$ to 0.069 m at $h = 24$, and its mean error stayed within ± 0.01 m at every horizon. The recursive neural forecasts degraded faster: RMSE rose from 0.059 to 0.078 m for the GRU, from 0.060 to 0.078 m for the LSTM and from 0.062 to 0.094 m for the CNN. All three accumulated a growing positive mean error (observed minus forecast), that is, they increasingly under-forecast; at $h = 24$ the bias was 0.036 m (GRU), 0.038 m (LSTM) and 0.046 m (CNN), compared with 0.009 m for ARIMA. The FFNN under-forecast at every horizon (by 0.041–0.067 m) and `nnetar()` by up to 0.031 m. This drift is the accumulation of error that the recursive scheme allows: the level-trained networks gradually fall back toward earlier values instead of following the rise (Section 5.3). The saw-tooth pattern in Figure 9a–b is an artefact of the quarterly spacing of origins, because horizons that are multiples of three always target the same four calendar months.

Table 10 and Figure 9c report the out-of-sample calibration of the prediction intervals. The ARIMA Gaussian intervals were closest to nominal: their 95% intervals covered 91–100% of observations across horizons, although the 80% intervals covered only 59–82%, and they had the lowest interval score at every horizon and at both levels. Empirical intervals built from each model's past errors covered 69–96% of observations at the 95% level and 47–88% at the 80% level, somewhat below nominal. This shortfall reflects the small calibration sets (as few as 10 errors) and the larger errors of the most recent years, which are not represented in the earlier errors. The simulated `nnetar()` intervals were far too narrow (95% coverage 31–46%; mean width 0.08–0.11 m vs 0.21–0.23 m for ARIMA), because they

resample overfitted in-sample residuals (Section 5.4).

The FFNN ensemble spread is not a prediction interval. On the single split, the mean $\pm 1\sigma$ band of the 20 members was about ± 0.028 m wide and contained only 18% of the one-step-ahead test observations; widened to $\pm 1.96\sigma$ it contained 34%. For a 24-month recursive forecast from December 2020, the corresponding coverages of January 2021 – December 2022 were 29% and 54%. Fitted to the full record, the spread remained about ± 0.03 m at all lead times (± 0.033 m at $h = 1$ and ± 0.034 m at $h = 24$). It therefore does not grow with horizon and represents initialization variability only. The $\pm 1\sigma$ band in Figure 6 is accordingly labelled as ensemble spread.

Table 11 gives 80% and 95% intervals for the 24-month forecasts. For the neural models, the empirical intervals are asymmetric about the point forecasts, with more room above than below, because the backtest errors of these models are biased toward under-forecasting; for example, the GRU forecast of 0.124 m for February 2027 has a 95% interval of 0.026–0.291 m. The 95% intervals at February 2027 are 0.23–0.28 m wide for all models, larger than the spread of the point forecasts (0.108–0.240 m). The differences between the models' 24-month forecasts are therefore small compared with the uncertainty of any one of them.

5.6 Association with climate indices

Monthly MSL anomalies at Ocean City were significantly and negatively associated with the same-month NAO ($r = -0.34$, $p < 0.001$) and AO ($r = -0.30$, $p < 0.001$), with anomalies about 1.7–1.9 cm higher per unit decrease of either index (Table 12, Figure 10a). Higher sea level in negative NAO/AO months is consistent with the weaker westerlies and more frequent coastal storms of those phases and with the known role of along-shelf wind stress in this region (Andres et al., 2013), but the indices explained only 9–11% of the anomaly variance. The association weakened sharply at a one-month lag ($r = -0.17$ for the AO and -0.12 for the NAO), except for the AO in winter ($r = -0.26$, $p = 0.011$).

The 1-month backtest errors of ARIMA and of the GRU were also correlated with the same-month NAO ($r = -0.42$ and -0.39 , $p \leq 0.003$; Figure 10b): both models tend to under-forecast in negative-NAO months and over-forecast in positive-NAO months. The largest single-split error, an over-forecast of 0.13 m in February 2022, occurred with strongly positive indices (AO = 1.54, NAO = 1.68), and the errors in March 2021, September 2024 and December

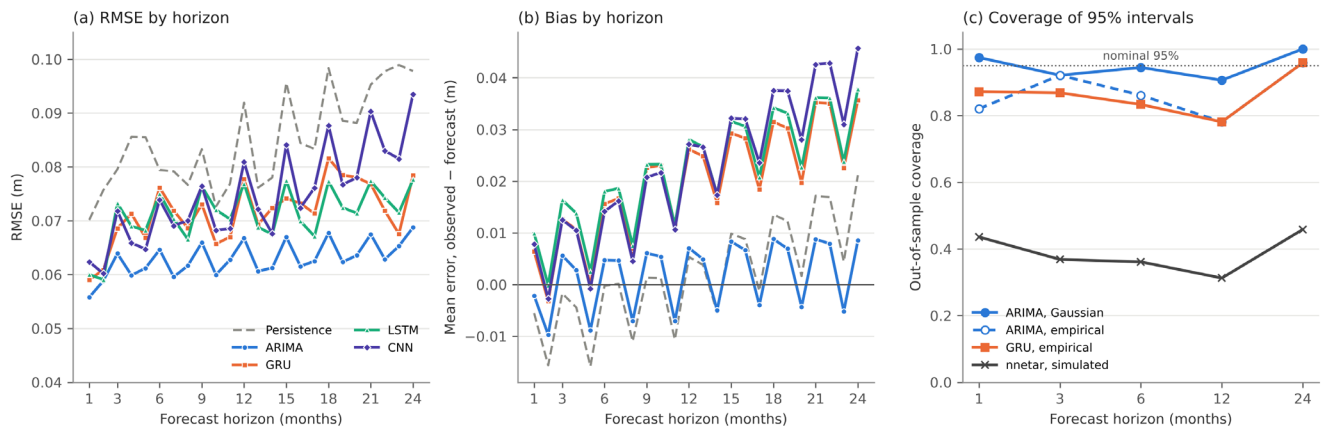


Figure 9. Multi-step backtest results (expanding-window rolling origin, 49 origins): (a) RMSE and (b) mean error (observed – forecast) by forecast horizon for persistence, ARIMA and three neural models; (c) out-of-sample coverage of nominal 95% prediction intervals (ARIMA Gaussian and empirical, GRU empirical, nnetar() simulated). The saw-tooth pattern in (a) and (b) reflects the quarterly spacing of forecast origins.

Table 10. Out-of-sample coverage of nominal 80% and 95% prediction intervals by forecast horizon h (months), and mean 95% interval (Winkler) score over the five horizons.

Interval method	$h = 1$: 80% / 95%	$h = 3$: 80% / 95%	$h = 6$: 80% / 95%	$h = 12$: 80% / 95%	$h = 24$: 80% / 95%	Mean 95% Winkler score (m)
Evaluation pairs (n)	39	38	36	32	24	
ARIMA, Gaussian (model-based)	0.82 / 0.97	0.71 / 0.92	0.67 / 0.94	0.59 / 0.91	0.67 / 1.00	0.258
ARIMA, empirical	0.77 / 0.82	0.66 / 0.92	0.64 / 0.86	0.50 / 0.78	0.67 / 0.96	0.339
Persistence, empirical	0.77 / 0.92	0.76 / 0.87	0.78 / 0.94	0.72 / 0.81	0.58 / 0.79	0.542
GRU, empirical	0.67 / 0.87	0.74 / 0.87	0.67 / 0.83	0.56 / 0.78	0.88 / 0.96	0.415
LSTM, empirical	0.67 / 0.79	0.68 / 0.84	0.69 / 0.81	0.47 / 0.81	0.79 / 0.96	0.375
CNN, empirical	0.69 / 0.85	0.66 / 0.84	0.64 / 0.78	0.50 / 0.69	0.54 / 0.88	0.425
FFNN, empirical	0.54 / 0.72	0.61 / 0.92	0.53 / 0.81	0.53 / 0.69	0.71 / 0.79	0.419
nnetar(), empirical	0.72 / 0.92	0.61 / 0.87	0.56 / 0.81	0.50 / 0.75	0.83 / 0.92	0.461
nnetar(), simulated (model-based)	0.36 / 0.44	0.29 / 0.37	0.25 / 0.36	0.19 / 0.31	0.21 / 0.46	1.379

Note. (lower is better; best in bold). Empirical intervals use each model's own past rolling-origin errors available at the forecast origin.

2024 also had the sign expected from the indices. However, four of the eight largest test errors (including January 2025 and June 2024) occurred with index values close to zero, so the indices do not account for all large errors.

Adding the indices to ARIMA confirmed their limited value for forecasting. With same-month indices, which would not be available in real time, the test RMSE fell from 0.0642 m to 0.0617 m (about 4%; DM $p = 0.21$), and in the rolling-origin evaluation the 1-month RMSE fell from 0.0539 to 0.0517 m ($p = 0.29$). With previous-month

indices, which could be used operationally, accuracy was unchanged (0.0642 m). These results provide some support for an atmospheric contribution to the error pattern, but they do not establish wind events, storm surge or shelf-scale dynamics as its cause; testing those mechanisms requires local wind, pressure and ocean data.

All four models share several common limitations. First, they are purely univariate, using only the MSL time series itself. Incorporating relevant physical predictors,

Table 11. Point forecasts (m) with 80% and 95% prediction intervals for March 2025, February 2026 and February 2027.

Model (inter- val type)	Mar 2025 point	80% interval	95% interval	Feb 2026 point	80% interval	95% interval	Feb 2027 point	80% interval	95% interval
ARIMA (Gaussian)	0.148	0.078– 0.219	0.040– 0.256	0.206	0.128– 0.284	0.087– 0.325	0.215	0.137– 0.293	0.096– 0.334
ARIMA (em- pirical)	0.148	0.052– 0.205	0.045– 0.243	0.206	0.134– 0.295	0.072– 0.319	0.215	0.142– 0.309	0.076– 0.325
GRU (empir- ical)	0.128	0.056– 0.213	0.001– 0.227	0.123	0.056– 0.258	0.018– 0.278	0.124	0.087– 0.256	0.026– 0.291
LSTM (empir- ical)	0.183	0.115– 0.253	0.070– 0.290	0.170	0.100– 0.286	0.065– 0.320	0.167	0.127– 0.297	0.083– 0.314
CNN (empir- ical)	0.275	0.218– 0.351	0.154– 0.384	0.225	0.129– 0.338	0.114– 0.380	0.240	0.168– 0.379	0.134– 0.403
FFNN (empir- ical)	0.113	0.076– 0.233	0.040– 0.287	0.108	0.092– 0.264	0.018– 0.287	0.108	0.093– 0.270	0.012– 0.288

Note. Neural point forecasts are those of Table 4; their empirical intervals use the rolling-origin errors of the re-implemented networks at the same horizon. All values in meters relative to the station MSL datum.

Table 12. Association of monthly MSL anomalies and forecast errors with the NAO and AO indices (upper part) and test performance of ARIMA with index covariates (lower part).

Analysis	r	p (HAC)	R ²	Slope (cm per index unit)
Anomaly vs NAO, same month, all months (n = 271)	−0.34	<0.001	0.114	−1.9
Anomaly vs NAO, same month, Dec–Mar (n = 91)	−0.31	0.003	0.096	−2.4
Anomaly vs NAO, previous month, all months (n = 270)	−0.12	0.080	0.013	−0.6
Anomaly vs NAO, previous month, Dec–Mar (n = 91)	−0.16	0.214	0.026	−1.3
Anomaly vs AO, same month, all months (n = 271)	−0.30	<0.001	0.090	−1.7
Anomaly vs AO, same month, Dec–Mar (n = 91)	−0.34	<0.001	0.116	−1.7
Anomaly vs AO, previous month, all months (n = 270)	−0.17	0.037	0.029	−1.0
Anomaly vs AO, previous month, Dec–Mar (n = 91)	−0.26	0.011	0.070	−1.4
ARIMA 1-month backtest error vs NAO, same month (n = 49)	−0.42	<0.001	0.175	−2.0
ARIMA 1-month backtest error vs AO, same month (n = 49)	−0.18	0.151	0.033	−1.2
GRU 1-month backtest error vs NAO, same month (n = 49)	−0.39	0.003	0.149	−1.9
GRU 1-month backtest error vs AO, same month (n = 49)	−0.20	0.098	0.040	−1.3
Persistence 1-month backtest error vs NAO, same month (n = 49)	−0.29	0.080	0.086	−1.8
Persistence 1-month backtest error vs AO, same month (n = 49)	−0.08	0.470	0.007	−0.7

Model (single split, test January 2021 – February 2025)	RMSE (m)	MAE (m)	R ²	DM p vs ARIMA
ARIMA(3,0,0)+trend	0.0642	0.0528	0.250	—
+ AO, same month (diagnostic: index known)	0.0632	0.0525	0.274	0.52
+ NAO, same month (diagnostic)	0.0618	0.0507	0.306	0.24
+ AO and NAO, same month (diagnostic)	0.0617	0.0507	0.309	0.21
+ AO, previous month (forecast-feasible)	0.0642	0.0527	0.251	0.75
+ NAO, previous month (forecast-feasible)	0.0642	0.0530	0.250	0.99
Rolling origin, 1-month (49 origins): ARI-MA(3,0,0)+trend / + AO same month / + AO previous month	0.0539 / 0.0517 / 0.0550			0.29 / 0.29

Note. Models with same-month indices are diagnostic, because the index is not known when the forecast is made.

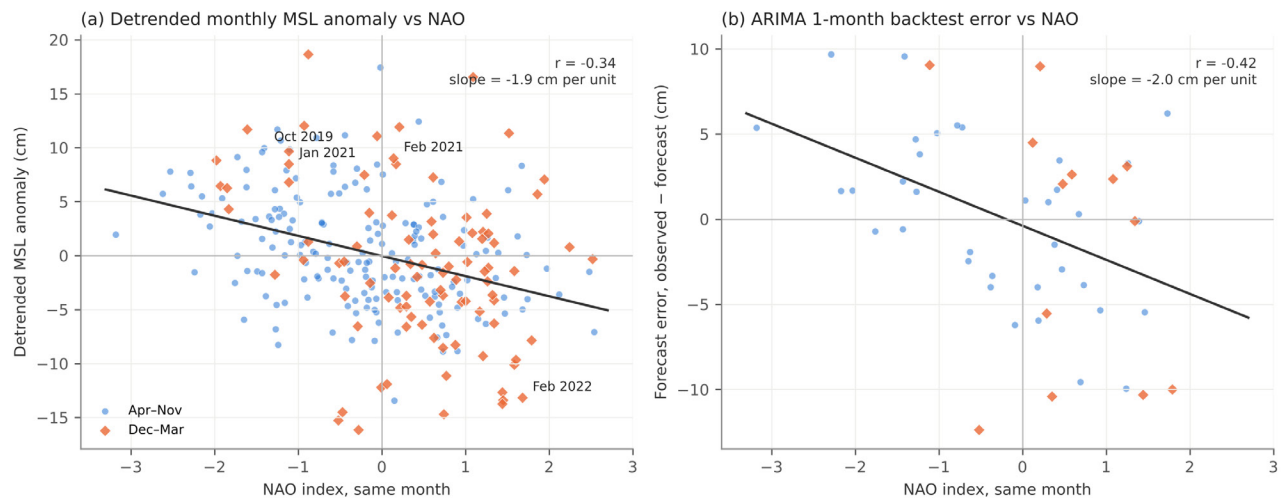


Figure 10. (a) Detrended monthly MSL anomaly and (b) ARIMA 1-month rolling-origin forecast error against the same-month NAO index, with least-squares lines; winter months (December–March) are shown as diamonds.

including sea surface temperature anomalies, North Atlantic Oscillation index, coastal wind stress, and precipitation, may improve performance, particularly for the episodic anomalies that currently drive high errors. However, in Section 5.6 same-month NAO and AO indices reduced the ARIMA test RMSE by only about 4%, and previous-month indices not at all, so local wind, pressure and shelf-circulation covariates are probably needed. Second, the training record of approximately 22 years (271 months) is modest by deep learning standards; longer records from nearby stations with longer operational histories (e.g., Baltimore, Lewes) could augment training data. Third, all models are static and do not update in real-time as new observations arrive; operational deployment would require rolling retraining or recursive update schemes. Fourth, all results come from a single tide gauge with 22 years

of seasonally adjusted monthly data and from one set of architectures and hyperparameters, and the neural results rely on a single random seed per model (per origin in the rolling-origin evaluation); the GRU's advantage over the other neural models on the single split is therefore a limited result for this station and study design. Whether the rankings reported here hold at other stations, for other record lengths or under other variability regimes has not been tested.

6. Conclusion

This study conducted a systematic comparative evaluation of four neural network architectures, including LSTM, GRU, CNN, and FFNN, for MSL forecasting at Ocean City, Maryland, using 271 months of NOAA tide gauge observations (August 2002 – February 2025). On the

single 80/20 split, the GRU emerged as the best-performing neural model, achieving the lowest RMSE (0.0704 m), lowest MAE (0.0587 m), and highest R^2 (0.099) on the held-out test set. This R^2 nevertheless means that about 90% of the test-period variance was unexplained. The LSTM ranked second across all metrics, while the CNN and FFNN both yielded negative R^2 values, indicating they performed worse than a naïve mean forecast on this dataset. When the same protocol was applied to seven forecasting baselines, however, ARIMA(3,0,0) with a linear trend (RMSE = 0.0642 m, R^2 = 0.250), SARIMA and simple exponential smoothing all outperformed the GRU, and the GRU improved on a persistence forecast by only 3% in RMSE. A rolling-origin evaluation over 49 expanding-window origins confirmed these findings at horizons of 1–24 months and showed that the single-split ranking among the neural architectures depends on the choice of test period.

The results support the following key conclusions. First, on the single split at this station, the recurrent models (LSTM, GRU) were more accurate than the feed-forward (FFNN) or local convolutional (CNN) architectures, but this ranking was not stable across test periods or horizons (Section 5.2) and should not be generalised to other stations without further testing. Second, the GRU converged faster than the LSTM and was slightly more accurate on the single split; this is consistent with, but does not demonstrate, an advantage of fewer parameters for short records, and the difference was within seed-to-seed variability (Table 8). Third, all models struggle to predict individual monthly anomalies; about 10% of the anomaly variance covaried with the same-month NAO and AO indices, consistent with a contribution from atmospheric forcing, but the indices had almost no forecast value at a one-month lead. Multivariate models with local wind, pressure and ocean covariates are a necessary next step. Fourth, the FFNN ensemble spread (about ± 0.026 m) is much narrower than the observed monthly variability and does not provide calibrated prediction intervals; interval estimates from a model with an explicit error distribution, such as ARIMA, are preferable. Fifth, and most importantly for practice, none of the neural architectures outperformed the ARIMA, SARIMA or ETS baselines, and only the GRU improved, marginally, on persistence; the advantages of recurrent gating described above are therefore relative to other neural designs, not to established statistical methods. Sixth, removing a training-period trend before training improved every neural architecture and stabilised its forecasts, but

did not close the gap to the statistical models. Seventh, the three-hidden-layer FFNN should not be equated with the NNAR method: the `nnetar()` implementation was much more accurate at short horizons, although it did not outperform ARIMA either, and its multi-step forecasts and simulated prediction intervals were unreliable. Eighth, the recursive neural forecasts accumulated a growing under-forecasting bias with lead time (0.036–0.046 m at 24 months for the GRU, LSTM and CNN), and among the uncertainty statements tested only the ARIMA model-based intervals were close to nominal coverage in backtesting; the 24-month forecasts should be read together with these intervals (Table 11).

These findings are specific to Ocean City and to this study design, and they do not support a general recommendation for the operational deployment of any of the neural models. At this station, ARIMA and ETS models with a deterministic trend were the most reliable of the methods tested, and any neural forecaster for similar records should at a minimum be compared with such baselines over multiple forecast origins; whether the rankings found here hold at other tide gauge stations remains untested. The GRU's 24-month forecast for Ocean City (0.122–0.124 m) sits close to the most recent observed level, whereas trend-bearing baselines project approximately 0.20–0.22 m by early 2027, so the GRU forecast should not be read as an extrapolation of the station's long-term trend. Future work should explore (1) multivariate LSTM/GRU architectures incorporating atmospheric forcing indices; (2) transfer learning from longer neighboring station records; (3) Bayesian and conformal prediction intervals for operational uncertainty quantification; and (4) hybrid CNN-LSTM architectures that combine local pattern extraction with sequential memory for multi-step coastal sea level forecasting. Before any of these models is recommended for operational use, the comparison with simple and statistical baselines should be repeated over multiple forecast origins at multiple tide gauge stations. Candidate covariates include local wind stress and sea-level pressure from NOAA CO-OPS stations or reanalyses, to separate the inverse-barometer and wind-setup contributions, together with indices of Gulf Stream transport and regional sea surface temperature.

Data and Code Availability

The monthly mean sea level (MSL) time-series data analyzed in this study were obtained from the

publicly available National Oceanic and Atmospheric Administration (NOAA) Tides and Currents repository for station 8570283: NOAA Sea Level Trends.

The code used for data acquisition and processing, model development, forecasting, statistical analysis, and figure generation is available from the corresponding author upon

reasonable request. Processed datasets, model outputs, and other materials required to reproduce the findings of this study are also available upon reasonable request.

Appendix A. Reproducibility Details

Table A1. Data source, exact segment dates and random seeds.

Item	Specification
Data file	NOAA CO-OPS Sea Level Trends, Station 8570283 (Ocean City Inlet, MD), file 8570283_meanrend.csv; monthly mean sea level, seasonal cycle removed, meters relative to the station MSL datum; accessed 29 April 2025. archived copy SHA-256 7ef31a2058738a05e5d995bdbb-90366da26df9f3fd5a88fad84c010952d17662
Analysis period	August 2002 – February 2025, 271 consecutive months; no missing months (NOAA gap February 1999 – July 2002 excluded by starting in August 2002)
LSTM, GRU, CNN (single split)	Inputs: previous 24 months. Scaler fitted on August 2002 – December 2020. Training targets August 2004 – June 2018 (167 windows); validation for early stopping July 2018 – December 2020 (30 windows, last 15% of training windows); test January 2021 – February 2025 (50).
FFNN (single split)	Inputs: lags 1–13. Scaler fitted on the training months (through October 2020). Training targets September 2003 – March 2018 (175); validation April 2018 – October 2020 (31); test November 2020 – February 2025 (52).
Statistical baselines and nnetar() (single split)	Estimation August 2002 – December 2020 (221 months); test January 2021 – February 2025 (50), one-step-ahead.
Rolling origin	49 origins every three months, December 2012 – December 2024; expanding training window from August 2002 to the origin; models re-fitted (and scaler re-fitted) at each origin; for the neural networks the last 15% of training windows used for early stopping; forecasts $h = 1$ –24 months, targets January 2013 – February 2025.
24-month outlook	March 2025 – February 2027, recursive.
Random seeds	TensorFlow/NumPy seed 42 (LSTM, GRU, CNN); FFNN ensemble seeds 0, 7, ..., 133; seeds 42, 1, 2, 3 and 4 in the sensitivity analysis; R set.seed(42) for nnetar() single split and full record, set.seed(origin index) in the rolling origin.

Table A2. Software used for the analyses in this revision.

Software	Version
Python	3.13.16
TensorFlow (CPU) / Keras	2.21.0 / 3.15.1
NumPy / pandas / SciPy	2.5.3 / 3.0.5 / 1.18.1
statsmodels / scikit-learn / matplotlib	0.15.0 / 1.9.1 / 3.11.2
R / forecast / nnet	4.3.3 / 8.21.1 / 7.3.19
Hardware	Linux x86_64, CPU only (2 cores); about 2–3 h to retrain all neural models over 49 origins

Table A3. Trainable parameters and training samples of each model.

Model	Configuration	Trainable parameters	Training samples (single split)
LSTM	$2 \times \text{LSTM} (64, 32) + \text{Dense} (16) + \text{Dense} (1)$; input 24×1	29,857	167 (+30 validation)
GRU	$2 \times \text{GRU} (64, 32) + \text{Dense} (16) + \text{Dense} (1)$; input 24×1	22,817	167 (+30 validation)
CNN	$3 \times \text{Conv1D} (64, 32, 16; \text{kernel } 3, \text{valid padding}) + \text{MaxPooling} (2) + \text{Dense} (32) + \text{Dense} (1)$	12,657	167 (+30 validation)
FFNN	14 inputs (lags 1–13, lag 12 duplicated) $\rightarrow 32 \rightarrow 16 \rightarrow 8 \rightarrow 1$; 20 networks	1,153 per network (1,121 with 13 inputs)	175 (+31 validation)
nnetar() NNAR(16,1,8) ₁₂	16 inputs $\rightarrow 8 \rightarrow 1$; 20 networks (single split)	145 per network	205
nnetar() NNAR(24,1,12) ₁₂	24 inputs $\rightarrow 12 \rightarrow 1$; 20 networks (full record)	313 per network	247
ARIMA(3,0,0) + trend	Constant, trend, 3 AR coefficients, innovation variance	6	221
SARIMA(1,0,1) (0,0,1) ₁₂ + trend	Constant, trend, AR, MA, seasonal MA, innovation variance	6	221
ETS(A,N,N)	Smoothing parameter, initial level, error variance	3	221
Linear trend / drift / persistence / seasonal naïve	—	2 / 1 / 0 / 0	221

Note. Neural-network counts are from Keras count_params() and agree with closed-form formulas; nnetar() counts are the fitted nnet weights; statistical-model counts are the fitted statsmodels parameters. The GRU count assumes the Keras default reset_after = True (TensorFlow 2.0 and later; 22,529 with reset_after = False), and the CNN count assumes the default valid padding (14,193 with same padding).

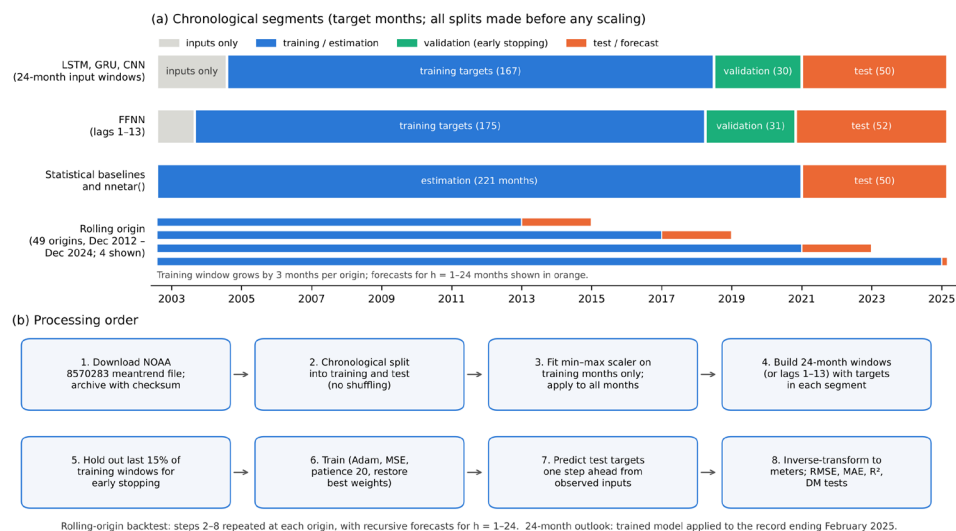


Figure A1. Order of splitting, scaling, window construction, validation and testing. (a) Target months of each segment for the single split and for four of the 49 rolling origins; (b) processing steps. The scaler is fitted only on training months, and validation windows for early stopping are the last 15% of the training windows, so no test-period information enters training or scaling.

References

- Andres, M., Gawarkiewicz, G. G., & Toole, J. M. (2013). Interannual sea level variability in the western North Atlantic: Regional forcing and remote response. *Geophysical Research Letters*, 40, 5915-5919. <https://doi.org/10.1002/2013GL058013>
<https://doi.org/10.1002/2013GL058013>
- Boon, J. D. (2012). Evidence of sea level acceleration at US and Canadian tide stations, Atlantic Coast, North America. *Journal of Coastal Research*, 28(6), 1437-1445.
<https://doi.org/10.2112/JCOASTRES-D-12-00102.1>
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv*. <https://arxiv.org/abs/1406.1078>
<https://doi.org/10.3115/v1/D14-1179>
- Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv*. <https://arxiv.org/abs/1412.3555>
- Church, J. A., & White, N. J. (2011). Sea-level rise from the late 19th to the early 21st century. *Surveys in Geophysics*, 32(4-5), 585-602.
<https://doi.org/10.1007/s10712-011-9119-1>
- Diebold, F. X., & Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 13(3), 253-263.
<https://doi.org/10.1080/07350015.1995.10524599>
- Ezer, T. (2019). Regional differences in sea level rise between the Mid-Atlantic Bight and the South Atlantic Bight: Is the Gulf Stream to blame? *Earth's Future*, 7(7), 771-783.
<https://doi.org/10.1029/2019EF001174>
- Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477), 359-378.
<https://doi.org/10.1198/016214506000001437>
- Harvey, D., Leybourne, S., & Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, 13(2), 281-291.
[https://doi.org/10.1016/S0169-2070\(96\)00719-4](https://doi.org/10.1016/S0169-2070(96)00719-4)
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.
<https://doi.org/10.1162/neco.1997.9.8.1735>
- Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and practice* (2nd ed.). OTexts.
<https://doi.org/10.32614/CRAN.package.fpp2>
- Hyndman, R. J., Athanasopoulos, G., Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., Petropoulos, F., Razbash, S., Wang, E., & Yasmien, F. (2023). *Forecast: Forecasting functions for time series and linear models* (Version 8.21.1) [R package]. <https://pkg.robjhyndman.com/forecast/>
- Hyndman, R. J., & Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software*, 27(3), 1-22.
<https://doi.org/10.18637/jss.v027.i03>
- Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. *International Conference on Learning Representations*. <https://arxiv.org/abs/1412.6980>
- Kopp, R. E., Hay, C. C., Little, C. M., & Mitrovica, J. X. (2015). Geographic variability of sea-level change. *Current Climate Change Reports*, 1(3), 192-204.
<https://doi.org/10.1007/s40641-015-0015-5>
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.
<https://doi.org/10.1109/5.726791>
- National Park Service. (2026). Assateague Island National Seashore: History and culture. <https://www.nps.gov/asis/learn/historyculture/history-and-culture.htm>
- Newey, W. K., & West, K. D. (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica*, 55(3), 703-708.
<https://doi.org/10.2307/1913610>

NOAA Climate Prediction Center. (2026a). Monthly mean Arctic Oscillation index since January 1950. https://www.cpc.ncep.noaa.gov/products/precip/CWlink/daily_ao_index/monthly.ao.index.b50.current.ascii.table

NOAA Climate Prediction Center. (2026b). Monthly mean North Atlantic Oscillation index since January 1950. <https://www.cpc.ncep.noaa.gov/products/precip/CWlink/pna/norm.nao.monthly.b5001.current.ascii.table>

NOAA CO-OPS. (2025a). Relative sea level trend data, Station 8570283 Ocean City Inlet, MD: Monthly mean sea level with the average seasonal cycle removed, relative to the most recent MSL datum. https://tidesandcurrents.noaa.gov/sltrends/data/8570283_meantrend.txt

NOAA CO-OPS. (2025b). Station 8570283, Ocean City Inlet, MD: Station information. <https://tidesandcurrents.noaa.gov/stationhome.html?id=8570283>

NOAA. (2025). Tides and currents - Relative sea level trend: Ocean City, MD. National Oceanic and Atmospheric Administration. https://tidesandcurrents.noaa.gov/sltrends/sltrends_station.shtml?id=8570283

Sallenger, A. H., Doran, K. S., & Howd, P. A. (2012). Hotspot of accelerated sea-level rise on the Atlantic coast of North America. *Nature Climate Change*, 2(12), 884-888. <https://doi.org/10.1038/nclimate1597>
<https://doi.org/10.1038/nclimate1597>

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1), 1929-1958.

Sweet, W. V., Kopp, R. E., Weaver, C. P., Obeysekera, J., Horton, R. M., Thieler, E. R., & Zervas, C. (2017). Global and regional sea level rise scenarios for the United States (NOAA Technical Report NOS CO-OPS 083). National Oceanic and Atmospheric Administration. https://tidesandcurrents.noaa.gov/publications/techrpt83_Global_and_Regional_SLR_Scenarios_for_the_US_final.pdf

Tashman, L. J. (2000). Out-of-sample tests of forecasting accuracy: An analysis and review. *International Journal of Forecasting*, 16(4), 437-450. [https://doi.org/10.1016/S0169-2070\(00\)00065-0](https://doi.org/10.1016/S0169-2070(00)00065-0)

U.S. Census Bureau. (2026). QuickFacts: Ocean City town, Maryland (2020 Census population and land area). <https://www.census.gov/quickfacts/oceancitytownmaryland>

Vousdoukas, M. I., Mentaschi, L., Voukouvalas, E., Verlaan, M., Jevrejeva, S., Jackson, L. P., & Feyen, L. (2018). Global probabilistic projections of extreme sea levels show intensification of coastal flood hazard. *Nature Communications*, 9(1), 2360. <https://doi.org/10.1038/s41467-018-04692-w>

Wessel, P., & Smith, W. H. F. (1996). A global, self-consistent, hierarchical, high-resolution shoreline database. *Journal of Geophysical Research*, 101(B4), 8741-8743. <https://doi.org/10.1029/96JB00104>

Williams, W. H., & Goodman, M. L. (1971). A simple method for the construction of empirical confidence limits for economic forecasts. *Journal of the American Statistical Association*, 66(336), 752-754. <https://doi.org/10.1080/01621459.1971.10482340>

Yin, J., Griffies, S. M., & Stouffer, R. J. (2010). Spatial variability of sea level rise in twenty-first century projections. *Journal of Climate*, 23(17), 4585-4607. <https://doi.org/10.1175/2010JCLI3533.1>

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14(1), 35-62. [https://doi.org/10.1016/S0169-2070\(97\)00044-7](https://doi.org/10.1016/S0169-2070(97)00044-7)