Technical note: Using ~~Deep Neural Network~~ Long Short-term Memory Models to Fill ~~Spatio-Temporal~~ Data Gaps in Hydrological Monitoring Networks

Huiying Ren[1], Erol Cromwell[2], Ben Kravitz[3,4], and Xingyuan Chen[4]

[1]Earth Systems Science Division, Pacific Northwest National Laboratory, WA, USA
[2]Advanced Computing, Mathematics, and Data Division, Pacific Northwest National Laboratory, WA, USA
[3]Department of Earth and Atmospheric Sciences, Indiana University, Bloomington, IN, USA
[4]Atmospheric Sciences and Global Change Division, Pacific Northwest National Laboratory, WA, USA

Correspondence: Xingyuan Chen (Xingyuan.Chen@pnnl.gov)

Abstract. The spatio-temporal dynamics in subsurface hydrological flows over a long time window ~~in subsurface hydrological flow~~ are usually quantified through a network of monitoring wells; however, such observations often are spatially sparse and temporal gaps exist due to poor quality or instrument failure. In this study, we explore the ability of ~~deep~~ recurrent neural networks to fill ~~in gaps in~~ gaps in a spatially distributed time-series ~~data, especially for the datasets with high-frequency dynamics. We selected a location~~ dataset from a well network that monitors the dynamic and heterogeneous hydrologic exchanges between the Columbia River and its adjacent groundwater aquifer at the U.S. Department of Energy~~'~~'s Hanford site~~to demonstrate and evaluate the new method, using a 10-year spatio-temporal hydrological dataset of~~. This 10-year-long dataset contains hourly temperature, specific conductance, and groundwater table elevation measurements from 42 wells ~~that monitor the dynamic and heterogeneous hydrologic exchanges between the Columbia River and its adjacent groundwater aquifer~~with various lengths of gaps. We employ a ~~deep nueral network (DNN) architecture that contains stacked~~ long short-term memory (LSTM) ~~, convolutional and dense layers to address both the spatial and~~ model to capture the temporal variations in the ~~property fields.~~ observed system behaviors for gap filling. The performance of the ~~DNN-based~~ LSTM-based gap filling method was evaluated against a traditional autoregressive integrated moving average (ARIMA) method in terms of both the error statistics and ~~capturing nonlinear, dynamic patterns in~~ how well they capture the temporal patterns in river corridor wells that exhibit various dynamics signatures. ~~Although~~ Our study demonstrates that the ARIMA models yield better average error statistics, ~~they fail to capture~~ yet they tend to have larger errors during time windows with abrupt changes or high-frequency (daily and subdaily) variations~~in system states that are typical characteristics of a complex dynamic system. The DNN-based models~~. The LSTM-based models are found to excel in capturing both the high-frequency and low-frequency (monthly and seasonal) dynamics~~that are present in time series at all wells, although they~~, although the inclusion of high-frequency fluctuations may also lead to overly dynamic predictions ~~as guided by the training data. The DNN is shown to improve the predictive ability by taking~~ in time windows that lacks such fluctuations. The LSTM is able to take advantage of the spatial information from neighboring wells ~~under highly challenging situations, such as multiple days of~~ to improve the gap filling accuracy, especially for long gaps in system states that vary at subdaily scales. ~~The DNN-based models afford the great advantage of accounting~~

~~for spatial and temporal correlations and nonlinearity in data without apriori assumptions. Although DNN~~ Despite the fact that LSTM models require substantial training data and ~~computational resources and~~ have limited extrapolation power beyond the conditions represented in the training data, they ~~showed promising potential for gap filling in highly dynamic~~ afford the great flexibity to account for the spatial and temporal correlations and nonlinearity in data without a priori assumptions. Thus, LSTMs provide effective alternatives to fill in data gaps in spatially distributed time-series observations characterized by multiple dominant ~~modes of variability. Capturing such dynamics is essential to generate the most valuable observations to advance~~ frequencies of variability, which are essential for advancing our understanding of dynamic complex systems.

1 Introduction

Long-term hydrological monitoring using distributed well networks is of critical importance for ~~many areas, including~~ understanding how ecosystems respond to chronic or extreme perturbations, as well as for informing policies and decisions related to natural resources and environmental issues (Wett et al., 2002; Taylor and Alley, 2002; Grant and Dietrich, 2017). One of the most common methods ~~of collecting hydrological~~ to collect hydrologic and chemistry data in groundwater is through wells (Güler and Thyne, 2004; Strobl and Robillard, 2008; Lin et al., 2012); however, ~~wells are necessarily sparse, leaving spatial gaps in the dataset. Moreover,~~ most well data ~~will also~~ have temporal gaps due to instrument failure or poor quality of measurements for numerous reasons. These data gaps degrade the quality of the dataset and increase the uncertainty in the spatial and temporal patterns that are derived from them. ~~Gap filling is necessary~~ While gap filling is essential for developing understanding of ~~the underlying system as well as~~ dynamic system behaviours and for use in creating continuous, internally consistent boundary conditions for numerical models~~. Many natural systems exhibit nonlinear or/and non-stationary behaviors due to evolving nonlinear dynamics, which makes it challenging to reproduce those complex patterns while filling in datagaps~~, one outstanding challenge is to capture the nonstationarity in data.

Various statistical methods have been developed to fill gaps in spatio-temporal datasets, with the most commonly used being the autoregressive integrated moving average (ARIMA) method (Han et al., 2010; Zhang, 2003). For any given spatial location, ARIMA uses temporal ~~autocorrelation~~ autocorrelations to predict unobserved data points in a time series. Spatio-temporal autocorrelations can be considered by using multivariate ARIMA and space-time autoregressive models (Kamarianakis and Prastacos, 2003; Wikle et al., 1998; Kamarianakis and Prastacos, 2005); however, ARIMA cannot capture nonlinear trends because it assumes a linear dependence between adjacent observations (Faruk, 2010; Valenzuela et al., 2008; Ho et al., 2002). In addition, all existing space-time ARIMA models assume fixed global autoregressive and moving average terms, which would fail to capture evolving dynamics in highly dynamic systems (Pfeifer and Deutrch, 1980; Griffith, 2010; Cheng et al., 2012, 2014). Spectral-based methods, such as singular spectrum analysis, maximum entropy method, and Lomb-Scargle periodogram, have been used to account for nonlinear trends while filling in gaps in spatio-temporal datasets (Ghil et al., 2002;

Hocke and Kämpfer, 2008; Kondrashov and Ghil, 2006). However, these methods use a few optimal spatial or temporal modes occurring at low frequencies to predict the missing values, with the other higher frequency components discarded as noise, which may lead to reduced accuracy of the statistical models in fitting the observations and in predicting missing values (Kondrashov et al., 2010; Wang et al., 2012). Kriging and maximum likelihood estimation used in spatial and spatio-temporal gap filling often face computational challenges ~~as they require~~ in computing the covariance matrix of the data vector, which can be quite large (Katzfuss and Cressie, 2012; Eidsvik et al., 2014). Other nonlinear methods have been explored with some success, including expectation-maximization or Bayesian probabilistic inference including hierarchical models, Gaussian process, and Markov chain Monte Carlo; the spatial and temporal correlations are most effectively captured by using models that build dependencies in different stages or hierarchies (Calculli et al., 2015; Banerjee et al., 2014; Datta et al., 2016; Finley et al., 2013; Stroud et al., 2017). In general, the expectation-maximization algorithm and Bayesian-based methods are sensitive to the choice of initial values and prior distributions in parameter space (Katzfuss and Cressie, 2011, 2012). Moreover, the prior distributions with all the associated parameters in both the spatial and temporal domains need to be specified, which becomes increasingly difficult in more complex systems. Empirical Orthogonal Functions (EOF) related interpolation methods, such as least squares EOF (LSEOF), data interpolation EOF (DINEOF), and recursively subtracted EOF (REEOF), are widely used to fill in missing data from geophysical fields such as clouds in sea surface temperature datasets or other satellite-based images with regular gridded domains (Beckers and Rixen, 2003; Beckers et al., 2006; Alvera-Azcárate et al., 2016). However, the requirement of gridded data by the EOF methods limits their use in filling data gaps in irregularly spaced monitoring networks.

Deep neural networks (DNNs) (Schmidhuber, 2015) are data-driven tools that, in principle, could provide a powerful way of extracting the nonlinear spatio-temporal patterns hidden in the distributed time-series data without knowing their explicit forms (Längkvist et al., 2014). They are increasingly ~~been~~ being used in geoscience domains to extract patterns and insights from the streams of geospatial data and to transform the understanding of complex systems (Reichstein et al., 2019; Shen, 2018; Sun, 2018; Sun et al., 2019; Gentine et al., 2018). The umbrella term of DNN contains numerous categories of architectures, depending on the problem at hand. For the analyses in this paper, which are focused on filling gaps in time-series data, a natural choice of the architecture is recurrent neural networks (RNNs) (JORDAN, 1986). ~~These networks~~ RNNs take sequences (e.g., time series) as input and output single values or sequences that follow. They are designed to use information about previous events to make predictions about future events, essentially by letting the model "remember." However, for longer sequences of data, RNNs have been shown to lose memory from previously trained data, i.e., they "forget" (Hochreiter et al., 2001). ~~This affects the performance of RNNs, particularly for data where the beginning of a sequence impacts the prediction, since this~~ The earlier information becomes exponentially less impactful for the prediction as the size of the sequence increases. Long short-term memory (LSTM) networks are variations of RNNs that are explicitly designed to avoid this problem by using memory cells to retain information about relevant past events (Hochreiter and Schmidhuber, 1997). RNNs and LSTMs have been successfully applied to text prediction (Graves, 2013), text translation (Wu et al., 2016), speech recognition (Graves et al., 2013), and image captioning (You et al., 2016)~~(Specifics on LSTM architectures are described in Section ??). There have been~~ . Recently, the applications of RNNs and LSTMs are also emerging in hydrology. For example, Kratzert et al. (2018) used LSTMs to predict watershed runoff from meteorological observations, Zhang et al. (2018) used LSTMs for predicting sewer overflow events from

rainfall intensity and sewer water level measurements, and Fang et al. (2017) used LSTMs to predict soil moisture with high fidelity. ~~Compared to a single RNN/LSTM layer, more complex LSTM architectures such as stacked and bidirectional LSTMs, CNN-LSTM or convolutional LSTM have the potential to capture extra features (Graves et al., 2013; Pascanu et al., 2013) as shown in various applications, including action recognition (Zhu et al., 2016) and vulnerable road users location predictions (Saleh et al., 2017). A bidirectional RNN/LSTM works by duplicating the recurrent network into two networks: one responsible for fitting the positive time direction (i.e. the forward states) and the other responsible for the negative time direction (i.e the backwards state) (Schuster and Paliwal, 1997). In general, the input sequence is fed as-is to the forward state and a reversed copy of the input sequence is fed to the backwards state. The bidirectional LSTM can be used in history matching problems.~~

Our study aims to evaluate the potential of using ~~LSTM layers within a DNN architecture to fill~~ the LSTM models for filling gaps in spatio-temporal ~~environmental time series .~~ time series data collected from a distributed network. The LSTM-based gap filling method is tested using datasets collected to understand the interactions between a regulated river and a contaminated groundwater aquifer. We treat the gap filling as a forecasting problem, i.e., we use the historical data as input to predict the missing values in the data gaps. ~~We demonstrate our method using a test case that focuses on understanding the interactions between a regulated river and contaminated groundwater aquifer. We adopt the stacked-LSTM combined with the convolutional layer as our DNN model to understand the interactions between a regulated river and contaminated groundwater aquifer. The DNN-based~~ The performance of the LSTM-based gap filling method is compared with traditional time series approaches (~~e.g.~~ i.e., ARIMA) to identify situations in which ~~DNNs outperform ARIMA as well as what the optimal configurations might be for this particular application~~LSTM models outperform or under-perform the ARIMA models.

2 Study Site and Data Description

A 10-year (2008–2018) spatio-temporal dataset was collected from a network of groundwater wells that monitor temperature (Water conductivity and temperature probe CS547A by Campbell Scientific), specific conductance (SpC) (Water conductivity and temperature ~~probeCS547A~~ probe CS547A by Campbell Scientific), and water-table elevation (stainless-steel pressure transducer CS451 by the Campbell Scientific) at the 300 Area of the U.S. Department of Energy Hanford site, located in southeastern Washington State. The groundwater well network was originally built to monitor the attenuation of legacy contaminants. The groundwater aquifer at our study site is composed of two distinct geologic formations: a highly permeable formation (Hanford formation, consisting of coarse gravelly sand and sandy gravel) underlain by a much less permeable formation (the Ringold Formation, consisting of silt and fine sand). The dominant hydrogeologic features of the aquifer are defined by the interface between the Hanford and Ringold formations and the heterogeneity within the Hanford formation (Chen et al., 2012, 2013).

The intrusion of river water into the adjacent groundwater aquifer causes mixing of two water bodies with distinct geochemistry and stimulates biogeochemical reactions at the interface. The river water has lower SpC (0.1–0~~.2~~ .12 mS/cm) than the groundwater (averaging ~ 0.4 mS/cm). Groundwater has a nearly constant temperature (16–17$^{\circ}C$) as opposed to seasonally varying river temperature (3–22$^{\circ}C$). The highly heterogeneous coarse-textured aquifer (Zachara et al., 2013) interacts with

dynamic river stages to create complex river intrusion and retreat pathways and dynamics. The time series of multi-year SpC and temperature observations at the selected set of wells in the network have demonstrated these complicated processes of river water intrusion into our study site (Figure 1). Wells near the river shoreline (e.g., wells 1-1, 1-10A, 2-2, and 2-3) tend to be strongly affected by river water intrusion in spring and summer. As such, the dynamic patterns of SpC and temperature correspond well with river stage fluctuations, specifically that SpC decreases and temperature increases with increasing river stage. Fluctuations of SpC in well 2-2 appear to be stronger and at higher frequency than in other wells, likely indicating its higher connectivity with the river. For wells that are farther inland (e.g., well 1-15), on the other hand, temperatures remain consistently within the groundwater temperature range and SpC has three noticeable dips (dropping from 0.5 to 0.4 mS/cm range), coinciding with the high river stages in years 2011, 2012, and 2017, which are featured with higher peak river stages than other years ~~so~~ such that the river water was able to intrude further into the groundwater aquifer. ~~Well 2-5 is~~ In wells located at an intermediate distance from the river~~compared to other wells shown in Figure 1, so~~ , such as Well 2-5, the intrusion of river water is evident in most of the years except in low-flow years such as 2009 and 2015, during which both SpC and temperature remain nearly unchanged.

The understanding we developed from earlier studies is that the physical heterogeneity contributes to the different response behaviors at different locations while the river stage dynamics lead to multi-frequency dynamics in those responses. The seasonal and annual variations are driven by natural climatic forcing (Amaranto et al., 2019, 2018), whereas the higher-frequency (i.e., daily and sub-daily) fluctuations are primarily induced by operations of the upstream hydroelectric dam operations to meet various demands of human society (Song et al., 2018). Our system is representative of many dam-regulated gravel-bed rivers across the world, where dam operations, as a typical anthropologenic activity, have significantly altered the hydrologic exchanges between river water and groundwater, as well as the associated thermal and biogeochemical processes (Song et al., 2018; Shuai et al., 2019; Zachara et al., 2020). Note that the multi-frequency variations in data ~~are characterizing~~ characterize the dynamic features of data, which could exist in both short-term and long-term time series data as a result of short-term or long-term monitoring effort.

To understand the multi-frequency variations of ~~all the variables~~ the river water and groundwater mixing in each well at the study site, we perform spectral analysis on multi-year SpC observations at each selected well using a discrete wavelet transform (DWT). The DWT is widely used for time–frequency analysis of time series and relies on a "mother wavelet", which is chosen to be the Morlet wavelet (Grossmann and Morlet, 1984) to deal with the time-varying frequency and amplitude in time-series data at this site (Stockwell et al., 1996; Grinsted et al., 2004). We illustrate the Wavelet Power Spectrum (WPS) in log scale and its normalized global power spectrum (average WPS over the time domain) for the multi-year SpC time series in the first two columns of Figure 2. Data gaps are shown as blank regions in Figure 2; examples include early year 2009 at well 1-1, the beginning of year 2011 at well 1-10A, and the later part of year 2012 at well 2-2. The amplitude of WPS represents the relative importance of variation at a given frequency compared to the variations at other frequencies across the spectrum. At wells 1-1, 1-10A, 2-3, 2-5, and 2-2, the strong intensities of SpC signals appear at the half-year and yearly frequencies; however, well 1-15 has a different pattern in that most of its high intensities are below the 256-hour frequency. The averaged WPS ~~more clearly~~ further shows the contrast in behaviors: wells 1-1, 1-10A, 2-3, and 2-5 have a dominant frequency at half a

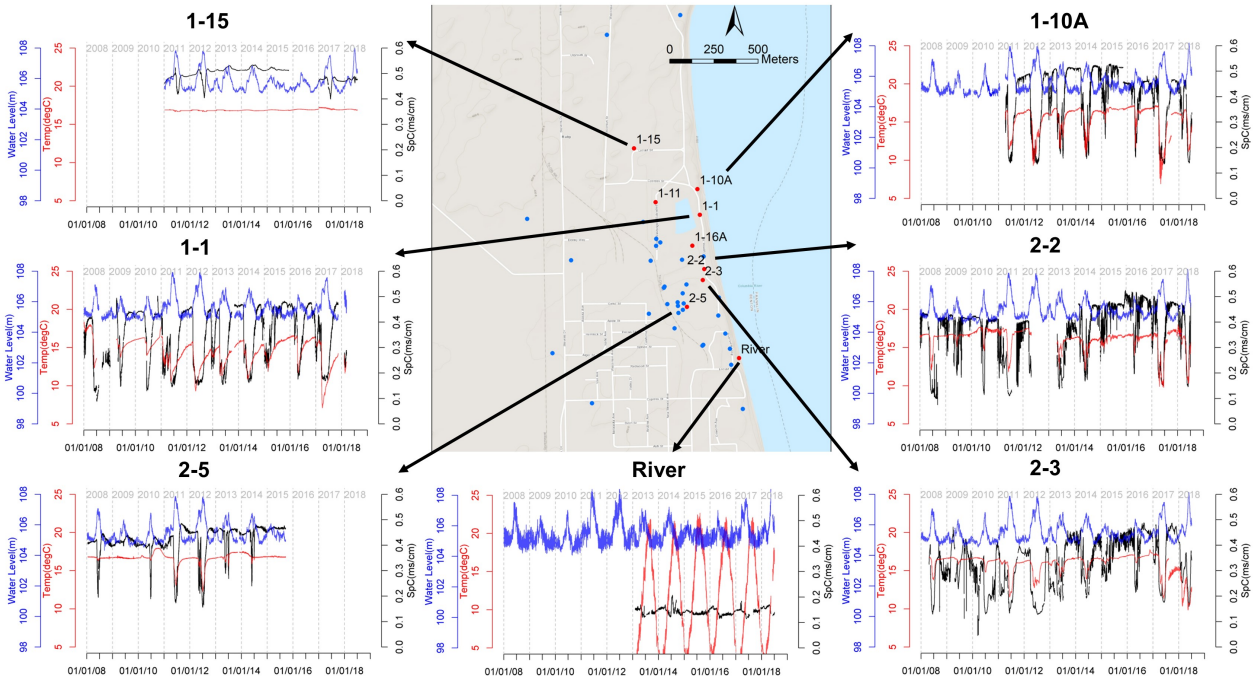

Figure 1. Groundwater monitoring well network at the 300 Area of the Hanford site and the monitoring data at select wells. Each well represented by a dot is instrumented to measure groundwater elevation, temperature, and SpC. The wells selected for this study are marked with red dots with well names. The three variables monitored in wells and in the river are shown in time-series plots with blue (water elevation), black (SpC), and red (temperature) lines. Base map @Google Maps.

year; well 2-2 has multiple dominant frequencies at daily, monthly, and seasonal scales; while well 1-15 has similar intensities at the half-year and hourly scales. Applying this information to the task at hand, we hypothesize that gap filling at well 2-2 could be more challenging due to ~~the multiple sources of its dynamic behaviors, manifested as significant powers at multiple frequencies.~~ such mixture of dynamics signatures.

~~WPS analysis of SpC at each well from 2008 to 2018. The first column is the spectrogram (in log10 scale) of SpC in each well; the second column is the averaged WPS; the third column is the coherence between SpC in each well and the river stage; and the fourth column is the averaged coherence with $p < 0.05$ values indicated in red.~~

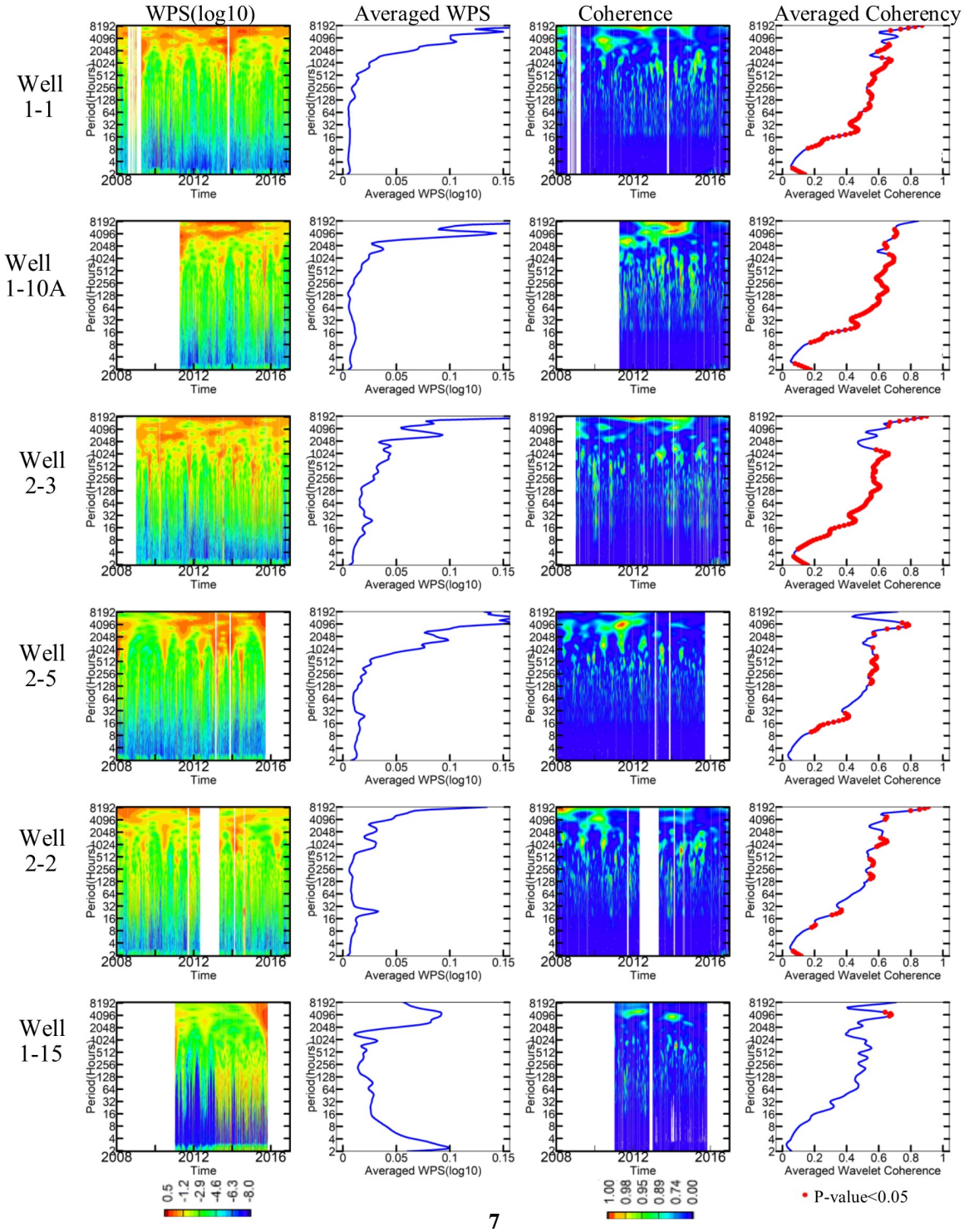

Figure 2. WPS analysis of SpC at each well from 2008 to 2018. The first column is the spectrogram (in log10 scale) of SpC in each well; the second column is the averaged WPS; the third column is the coherence between SpC in each well and the river stage; and the fourth column is the averaged coherence with $p < 0.05$ values indicated in red.

Since the dynamics of the system are driven by the river stage, we perform magnitude-squared wavelet coherence analysis via the Morlet wavelet to reveal dynamic correlations between the SpC and river stage time series (Grinsted et al., 2004; Vacha and Barunik, 2012). Wavelet coherence in the time-frequency domain is plotted in the third column in Figure 2 and the average coherence is plotted in the fourth column; statistically significant values at the 95th percent confidence interval are indicated with red points. A larger coherence at a given frequency indicates a stronger correlation at that frequency between the SpC at a well and the river stage. We consider these two variables highly correlated when the coherence is larger than 0.7 (shown in green to red colors in Coherence plots). We found that such high correlations exist at multiple frequencies, from subdaily to daily to yearly, at all the wells close to the river (e.g., 1-1, 1-10A, 2-2, and 2-3), while the higher correlation regimes in wells farther from the river (e.g., 1-15 and 2-5) are shifted towards longer periods at semi-annual and annual frequencies and less persistent in time.

As can be seen in Figure 2, many of the wells have long data gaps, which have unknown effects on our ability to estimate dynamics from the wavelet spectra. As such, gap filling is needed to infer observations and guide modeling of the underlying system. Figure 3 provides a summary of gap lengths for the overall network of monitoring wells. The majority of the gap lengths of all the three monitored variables are less than 50 hours. Therefore, in our investigations we explore the ability of the methods in filling gaps of 1-, 6-, 12-, 24-, 48-, and 72-hour lengths using hourly data over an input window to capture the ~~high-frequency~~ multi-frequency fluctuations.

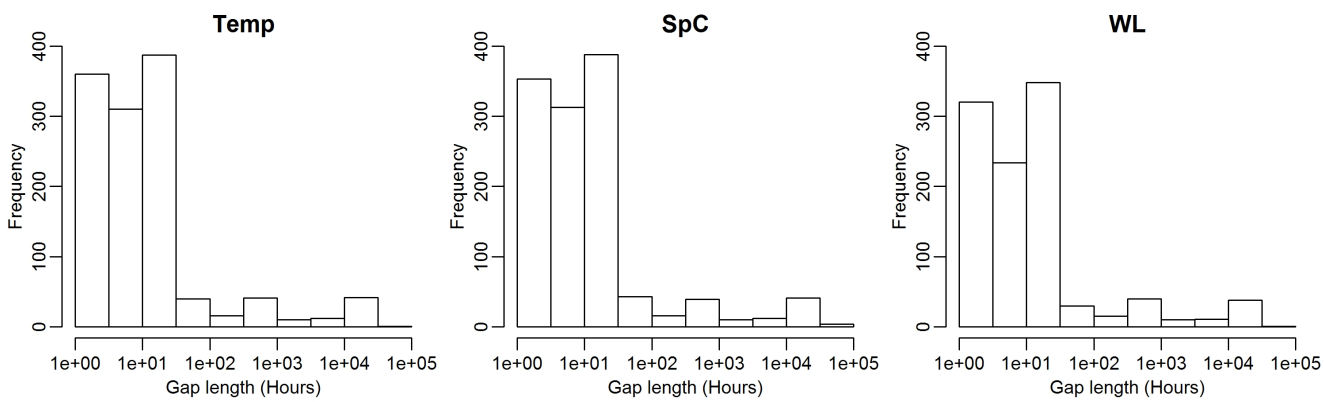

Figure 3. Histograms of gap lengths for each monitored variable, aggregated across all wells in the monitoring network during 2008-2018.

3 Gap-Filling Methods

In this section, we describe two methods we implemented to fill gaps of various lengths in SpC measurements at selected wells: ~~a DNN model using several LSTM network layers~~ an LSTM model and the traditional ARIMA model~~for comparison and assessing the strengths and limitations of the DNN model. In both methods, an~~ . While the LSTM model can be used to fill in gaps in all three variables, we focused our analyses on filling gaps in SpC because of its importance to reveal river water and groundwater mixing. Same set of analyses can be performed on water level and temperature. An input with M time steps (input window length) is provided to ~~predict outputs of N time steps (output window length) that follow~~ both models for estimating the missing SpC measurements. The LSTM model predicts the next time step that immediately follows the input window. ~~The performance of both methods were evaluated using testing datasets for the selected wells.~~

3.1 ~~DNN Models for Gap Filling~~

~~We designed a DNN architecture to train models of an input size~~ For gaps larger than one hour, this LSTM model is applied to fill in one missing value at a time. The entire gap is filled by sliding the input window forward hour by hour and treating the gap-filled values of the previous missing hours as observed values. The ARIMA model, on the other hand, fills the entire gap length using the input of M time steps ~~and an output size of N time steps to fill gaps of various lengths in groundwater well measurements. The input and predicted output at a time step contain the following three measurements~~ preceding the gap. The input window may contain multiple variables from a single ~~well~~ or multiple wells ~~: water level(m), temperature ($^{\circ}C$), and~~ that are relevant to the prediction. After experimenting with different sets of input variables (SpC only, SpC and water level, SpC plus water level and temperature), we found that including SpC and water level measurements in the input window yielded the most robust performance. Therefore, we used historic water level (m) and SpC (mS/cm) ~~, leaving the model to generalize nonlinear connections among them. Assuming the observations from W ($W \geq 1$) wells are used to fill in data gaps , the input size of the model is then $M \times 3W$. Similarly, the model output can be those three variables in the next N time steps for one or more wells. Using~~ observations to fill gaps in SpC time series of a single well. Using measurements from multiple wells as input ~~adds a spatial component to the model allows the DNN model~~ allows the models to account for both the temporal and spatial correlations in the data to ~~improve~~ impact gap-filling performance. Wells were selected based on adequate data availability and their distances from the ~~river. While the DNN model can be~~ target well. Assuming the observations from W ($W \geq 1$) wells are used to fill in ~~gaps in all three variables, we focused our analyses on filling gaps in SpC because of its importance to reveal river water and groundwater mixing. Same set of analyses can be performed on water level and temperature.~~ data gaps, the input size of the model is then $M \times 2W$.

~~We explored different DNN model architectures that contain a single or multiple LSTM layers for each desired combination of~~

3.1 LSTM Models for Gap Filling

We designed an LSTM architecture, as shown in Figure 4, to train models of an input size of M ~~and N at each well under various lengths of gaps with different amount of training data (2, 4 and 6 years).~~ time steps and an output size of one time step. The LSTM model contains a single LSTM layer followed by an output dense layer. The detailed structures of the LSTM layer is provided in the supplemental materials (Figures S1 and S2).

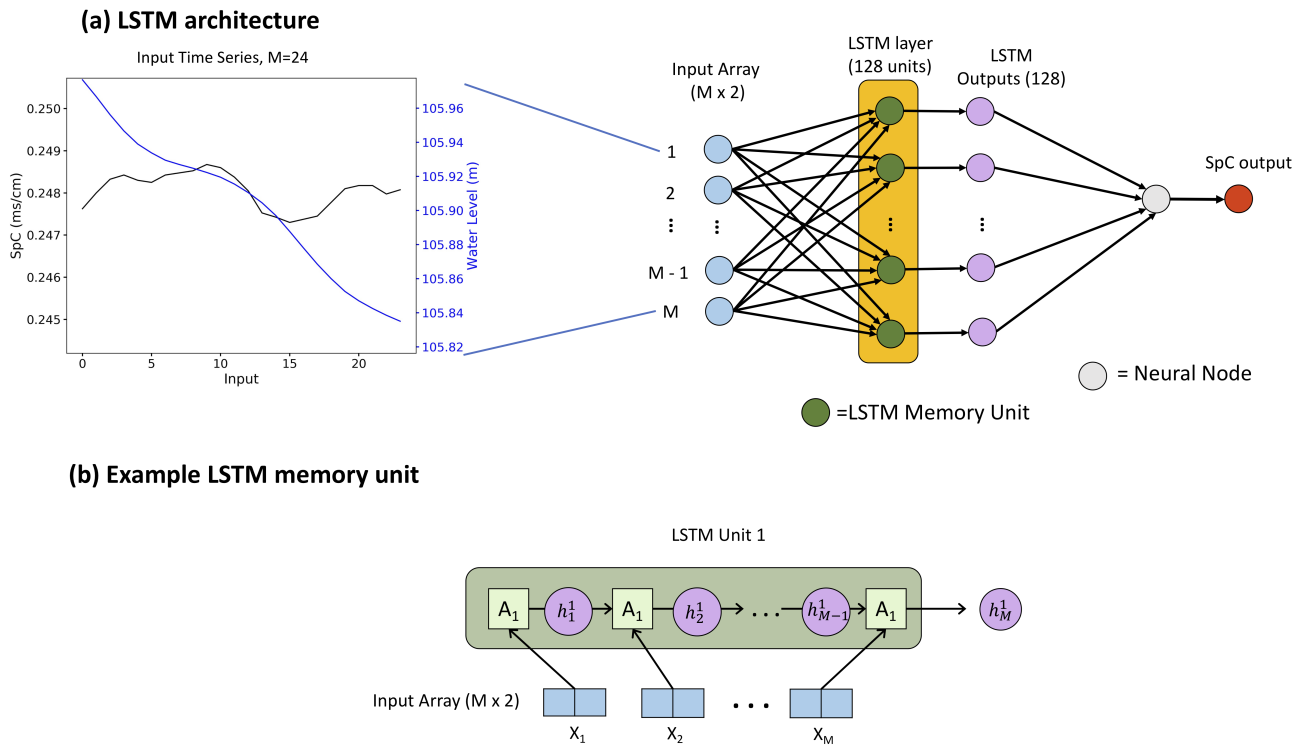

Figure 4. Illustration of LSTM models for gap filling. (a) Architecture of the LSTM models, where M is the input window size. Includes example input with $M = 24$ and example LSTM layer with 128 units; (b) Example of an LSTM unit, where A is the repeating module of the LSTM unit and h is the output.

Training data for the ~~DNN~~ LSTM models were created by finding data segments of ~~$M + N$~~ $M + 1$ hours that have no missing values, i.e., no gaps in the data, for ~~all three~~ both measurements over a specified monitoring window. The well data were then ~~preprocessed~~ pre-processed by normalizing all measurements ~~to fall between 0 and 1 using different scaling factors~~ via zero-mean and unit variance for each variable, as ~~temperature measurements are on a scale of 10^1,~~ SpC is on a scale of 10^{-1}, and water level is on a scale of 10^2. ~~After evaluating the gain in performance improvement by using increasingly more training data (details provided in the online supplemental materials), we concluded that 4 years of training data (2012-2015) was sufficient for all the models.~~ Validation datasets were used to select the best model hyperparameters ~~(??)~~ and the optimal ~~combination of~~ input window size M ~~and N~~ (3.1.1) for gap filling at each well. Another independent testing period was selected at each

well, depending on data availability, to compare the gap filling performance using the ~~DNN~~ LSTM and ARIMA methods. The complete set of alternatives we considered for each ~~DNN~~ LSTM model configuration is shown in Table 1. ~~Excluding combinations with $M < N$, 1080 unique models (180 models per well) were trained.~~ We used an Adam optimizer (Kingma and Ba, 2014) for training and the mean-squared error as the loss function. The models were trained for ~~30~~ 50 iterations (i.e., epochs) over the training data. ~~Each model configuration was trained using four different initialization seeds and error metrics were averaged to determine the best configuration.~~

~~In addition to the DNN models trained for the single-well setup, we also trained multi-well models that used observations from wells 1-1, 1-10A, and 1-16A to fill in data gaps for well 1-1. We explored the same set of configuration parameters shown in Table 1 for single-well models in multi-well models. We then compared the gap filling performance of the multi-well DNN with the single-well DNN model for well 1-1. The multi-well models were not explored for the other wells due to lack of neighboring wells in close proximity.~~

To evaluate the accuracy of the trained ~~DNN~~ LSTM models in filling SpC data gaps during the validation and testing processes, we assumed that synthetic gaps of various lengths (e.g., 1, 6, 12, 24, 48, and 72 hours, referred to as gap scenarios hereafter) exist in the validation or testing dataset of a well. ~~Then a DNN~~ We assume only the SpC measurements are missing while the water level measurements are available. Then an LSTM model configured with ~~input and output windows~~ an input of M ~~and N on a single or multiple wells is given the first~~ is given M hours of data from the time series preceding the occurrence of a gap (assuming no missing values in these M hours) to fill ~~in the first missing value in the gap by taking the first value of the predicted N hours. This gap-filled value is then treated as if it was observed when repeating this procedure to fill in the gap of the next hour . This sliding window moves forward hour by houruntil the entire gap of the data is filled~~the gap hour by hour. The accuracy of the gap-filling model is evaluated by calculating the mean absolute percentage error (MAPE; %) between the SpC values that are filled in (i.e., predicted) and that were observed:

$$MAPE = 100 \times \frac{1}{n} \sum_{t=1}^{n} \left| \frac{\text{Prediction} - \text{Observation}}{\text{Observation}} \right|, \tag{1}$$

where n is the ~~number of data points being missing~~total number of synthetic missing data points during the evaluation period.

3.1.1 ~~Hyperparameter search for the optimal DNN architecture~~

~~We performed a hyperparameter search to explore different model architecture configurations, i.e., the number of LSTM layers, number of units per LSTM layer, number (and size of) dense layers, and activation functions. The search was performed on~~ In addition to the LSTM models trained for the single-well setup, we also trained multi-well models that used observations from wells 1-1, 1-10A, and 1-16A to fill in data gaps for well 1-1~~only due to computational cost. We chose the optimal DNN architecture using model performance on validation data set of well 1-1 (see Table 1) using MAPE defined in Eq. (1).~~

~~The final DNN architecture,~~ . We explored the same set of configuration parameters for single-well models as shown in ~~Figure ??, contains three LSTM layers, followed by two dense layers with dropout, a convolutional layer, and a final output dense layer. Stacking three layers of LSTM was found to yield better performance than a one- or two-layer architecture. Each of the three LSTM layers has 128 units because this configuration outperformed others with more or fewer number of units.~~

Table 1. Parameters used in training single-well ~~DNN~~ LSTM models[a].

Parameter	Values
Training wells	1-1, 1-10A, 1-15, 2-2, 2-3, 2-5
Synthetic gap length (hours)	1, 6, 12, 24, 48, 72
Model input window (M hours)	24, 48, 72, 96, 120, 144, 168
~~(N hours~~**LSTM Units (U units)**	~~1, 6, 12, 24, 48, 72, 120, 144, 168~~ 32, 64, 128
Learning Rate (L)	1e-3, 1e-4, 1e-5
Training period	2012-2015
Validation period[a]	2011
Testing Period[b]	2008 for well 2-5; 2017 for well 1-15; 2016 for all other wells

[a] ~~no models were trained for combinations with $M < N$.~~ used to select the best ~~DNN~~ LSTM model configurations and hyperparameters.

[b] used to evaluate performance of ~~DNN~~ LSTM vs ARIMA.

~~The output from the last LSTM layer of size M x 128 is fed into two consecutive dense layers where every input neuron is connected to every output neuron with a weight matrix and bias vector. The output from the second dense layer of size M x 64 is fed into a convolutional layer with 24 filters of size M-N+1, reducing the output size to N x 24. Finally, a dense layer is applied to yield a model output of our desired size, $N \times 3$ for a single well or multiple of that when the model is designed to fill in data gaps in multiple wells . The detailed structures of the LSTM layers, dense layer, and convolutional layer are provided in the supplemental material. Dropout, i.e., randomly disables a selected fraction of neurons, was used in the dense layers as the regularization technique to enhance robust model performance and prevent overfitting (Hinton et al., 2012). We adopted a dropout rate of 0.3 after testing a set of alternatives (0, 0.1, 0.2, 0.3 and 0.4) .~~ Table 1 in multi-well LSTM models. We then compared the gap filling performance of the multi-well LSTM with the single-well LSTM model for well 1-1. The multi-well models were not explored for the other wells due to lack of neighboring wells in close proximity.

~~Architecture of the DNN models, where M is the input window size and N is the output window size. Includes example input and output data with $M = 24$ and $N = 12$. For a more detailed diagram of the LSTM layers, dense layer, and convolutional layer, see Figures S1, S2, and S3, respectively, in the Supplemental Online Material.~~

~~Each memory unit~~

3.1.1 Optimizing LSTM model configuration

We performed a hyperparameter search to explore different LSTM model configurations, including the input time window size M, the number of units (U) in the LSTM layer~~is further illustrated in Figure **??**. The top panel shows generic representations of an RNN in a looped (left) or chained (right) form, which allows information to be passed to the next successor and persist. While all RNNs have the form of a chain of repeating modules of neural network (i.e., boxes labeled as A in Figure **??**) , the module being repeated can take different structural design to control the information flow, leading to different variants of RNN. Standard LSTMs use three gates, as shown in the bottom panel of Figure **??**, to control the flow of information from one state to another and capture long-term dependencies. Each gate is composed of a linear layer with a sigmoid activation function. A forget gate ($\boldsymbol{f_t}$) decides what information to throw away from the previous memory state by using a sigmoid function that outputs a value between 0~~ , and the learning rate (L) at each well. We chose the optimal LSTM configuration using model performance on the validation data set (see Table 1) based on the MAPE defined in Eq. (1~~, where 0 represents completely forget the information and 1 represents completely keep the information. An input gate ($\boldsymbol{i_t}$) decides which values from the new input to be used for updating the memory state. The input gate is combined with a vector of new candidate input values out of a tanh layer (generates values between -1 and 1)through pointwise multiplication to yield information to be added to the current state. Finally, an output gate ($\boldsymbol{o_t}$) decides what to output based on the input and the previous memory state. The previous hidden state and the current input are passed to a sigmoid layer of the output gate, while the tanh layer scales the current memory state. Then, pointwise multiplication of the outputs from the tanh and sigmoid layers leads to the output of this repeating module. For a more detailed description of the components of the LSTM unit, the reader is referred to Hochreiter and Schmidhuber (1997) .~~

~~A diagram for network representing an LSTM unit. The top panel shows the looped and chain versions of a generic RNN, where $\boldsymbol{x_t}$ is the input, $\boldsymbol{h_t}$ is the output, and A is the repeating module of the LSTM unit. The bottom panel shows a diagram of the LSTM unit with the three main information gates: a forget gate ($\boldsymbol{f_t}$), an input gate ($\boldsymbol{i_t}$), and an output gate ($\boldsymbol{o_t}$). Images adapted from Olah (2015).~~

3.1.2 ~~Optimizing M and N for gap filling~~

~~Using the optimal DNN architecture, we further analyzed model MAPE metrics during the validation period with various combinations of M and N under each gap scenario for each well. The~~ . The combination that yielded the lowest SpC MAPE were selected as the best configuration for a given gap length at each well. The best model configurations were then used to evaluate the LSTM-based ~~DNN~~ gap filling method against the ARIMA-based method (3.2) using relative errors (similar to MAPE by setting n=1 in Eq. (1) and ~~remove~~ removing the absolute value operation) calculated for each data point in the testing period ~~as listed in table 1. The test period~~ (table 1), which varied among the wells due to availability of continuous data required for testing.

3.2 ARIMA Models for Gap Filling

ARIMA is one of the most general classes of models for extrapolating time series to produce forecasts and we used it as a baseline to compare and assess the ~~DNN~~ LSTM gap-filling method. ARIMA ~~is applicable to nonstationary processes in that~~

~~the dataset can be made stationary by differencingif necessary. Differencing~~can be applied to nonstationary time series data using a combination of differencing, autogressive, and moving average components~~make up a~~ . A nonseasonal ARIMA(p,d,q) model is given by:

$$Y_t = c + \phi_1 Y_{t-1}^d + \phi_p Y_{t-p}^d + ... + \theta_1 e_{t-1} + \theta_q e_{t-q} + e_t, \tag{2}$$

where ϕs and θs are polynomials of orders p and q, respectively, each containing no roots inside the unit circle. es are the error terms, $Y_\cdot^d$ is $Y_\cdot$ differenced d times, and c is a constant. Note that only non-seasonal terms (p,d,q) are included in ~~Equ~~Eq. (2). Seasonal structure can be added with parameters $(P,D,Q)_m$ to the base ARIMA model to become ARIMA$(p,d,q)(P,D,Q)_m$, ~~including~~ with a periodic component containing m periods. $c \neq 0$ implies a polynomial of order $d+D$ in the forecast function. The detailed mathematical equations for ~~seasonal ARIMA have been added in~~ the seasonal ARIMA are provided in the online supplemental materials.

The main task in ARIMA-based forecasting is to select appropriate model orders, i.e., the values of p,q,d,P,Q,D. If d and D are known, we can select the orders p,q,P,Q via an information criterion such as the Akaike Information Criterion (AIC):

$$AIC = -2\log(L) + 2(p+q+P+Q+k), \tag{3}$$

where $k=1$ if $c \neq 0$ and 0 otherwise, and L is the maximized likelihood of the model fitted to the differenced data. The best fitted parameters of the ARIMA model can be determined by minimizing the AIC.

~~Similar to the DNN-based gap filling, an ARIMA model is built for each combination of input and output window sizes for each well~~ The ARIMA models were built using the `auto.arima` function from the R package ~~called~~ `forecast` (Hyndman et al., 2007). ~~The length of output N in ARIMA corresponds to the gap lengths. Each trained ARIMA model was tested on the wellthat is used for training the model, as was done in the DNN approach. Accuracy of the ARIMA-based gap filling was evaluated~~

Similar to the LSTM-based gap filling, we explored various input window sizes for the ARIMA model at each well. An optimal input window size is chosen for each gap length using the same MAPE metric ~~shown in~~ (Eq. (1)~~during the testing period listed in Table 1 and compared with the DNN-based methods~~) on the validation dataset.

4 Results and Discussion

4.1 Performance of single-well ~~DNN~~ LSTM models

We ~~evaluated the accuracy of DNN models in filling gaps of various lengths of the SpC measurements in the year 2011 following the steps described in section 3.1.1. MAPEs~~ selected the best combination of LSTM units (U) and learning rate (L) for each input time window (M) under each gap length at each well using the MAPE metric. The validation MAPEs of those selected models were summarized in boxplots ~~for different DNN model configuration parameters~~under different grouping, as shown in Figure 5. Each MAPE boxplot was drawn from a group of models with one parameter (corresponding to each x-axis) fixed at the given value while all the other parameters, including the training wells and gap scenarios, cycle through

their possible combinations. ~~Although we attempted to train models with output window sizes greater than 24 hours, these models performed noticeably worse than those with output windows less than or equal to 24 hours (results shown in the online supplemental materials). Thus, our analyses here focus on models with output window less than or equal to 24 hours.~~

As shown in Figure 5 (a), model performance deteriorates as the gap length increases, indicating that the ~~DNN-based~~ LSTM-based method tends to lose ground truth information from its input to inform prediction. In comparing MAPEs across various input window sizes shown in Figure 5 (b), we observe that models with all input windows have comparable ~~median MAPEs~~MAPE summary statistics, with those ~~of 24, 72, 144 and 168 hours~~larger input windows ($>$ 96 hours) leading to slightly smaller ~~median MAPEs. The 144- and 168-hour~~ MAPE quartiles. The larger input windows also yield ~~lower third quartile of MAPE and~~ fewer outliers on the larger MAPE end, indicating that the memory units in the LSTM layers are capturing important daily to weekly signatures (evident in WPS plots in Figure 2 for all wells except for Well 1-15) for some wells. ~~As shown in 5 (c), daily and subdaily output windows yield comparable median MAPEs, with the 24-hour output window leading to smaller third quartile and fewer large MAPE outliers than its 1-, 6-, and 12-hour counterparts. Overall, an input window of 144 hours and an output windows of 24 hours appear to be a robust model configuration for all wells and gap lengths considered.~~

The performance of single-well ~~DNN~~ LSTM models varied among the wells as shown in Figure 5 (~~d~~c). The ~~DNN~~ LSTM models for well 1-15 lead the performance with the smallest MAPEs, while those for well 2-2 yield the worst performance. The ~~DNN~~ LSTM models for wells 1-1,1-10A, 2-3 and 2-5 performed comparably overall, with slightly more large MAPE outliers for well ~~1-10A~~2-3.

4.2 Single-well ~~DNN~~ LSTM and ARIMA comparisons

The single-well ~~DNN-based~~ LSTM gap filling approach was compared to the ARIMA approach using relative errors calculated for each data point that was assumed to be missing in the testing data by setting n=1 in Eq. (1) for MAPE. ~~Best~~ Relative errors were used to show overestimations or underestimations by both approaches. Their respective best model configurations determined on the validation dataset (i.e., data from year 2011) ~~, as described in sections 3.1.1 and 3.2,~~ were used in comparing the two approaches. Figure 6 illustrates the ~~input and output windows selected as the best model configurations for DNN~~ optimal input windows for the LSTM and ARIMA methods. ~~The output window N of an ARIMA model is the same as the length of the gap it is built to fill.~~ We observe that ~~DNN~~ LSTM models require less or equal input information than ~~that required by~~ the ARIMA method ~~for the wells tested. None of the optimal output window sizes exceeds 24 hours for the DNN models. We only compared two methods for gap lengths of 24, 48 and 72 hours because both methods were highly accurate in filling small gaps such as one hour.~~ under all gap lengths for all the wells except well 2-5.

Figure 7 shows the ~~interquartile ranges~~ boxplots of relative errors under different gap lengths for all ~~individual wells, each bounded by its 25th to 75th percentiles. Relative errors were used to show overestimations or underestimations by both approaches. The horizontal dotted lines represent the $\pm$5% relative error range that are typical measurement errors of the SpC sensors deployed at the site~~testing wells. For both approaches, the relative errors increase as the gap length increases as expected. The ARIMA models tend to perform better than the ~~DNN~~ LSTM models in terms of ~~error statistics. For both approaches, the relative errors increase as the gap length increases as expected, especially so for well 1-1 when the gap lengths~~

~~are~~ interquartile range. However, the ARIMA models, in general, produce more outliers of large positive or negative relative errors than the LSTM models, especially for larger gap lengths (48 and 72 hours~~. While all the relative errors yielded from the ARIMA method are within the ±5% measurement error range, there are a few cases using DNN leading to relative errors outside the typical observational error range with longer gaps for wells 1-1, 1-10A and 2-2. The relative errors in the ARIMA models tend to distribute symmetrically on both sides of 0%, whereas errors in the DNN models appear to skew toward the negative side for all wells except for well 2-5.~~). For well 1-15, the relative errors ~~for all three gap lengthsare very close to 0. Wells 1-1 and 2-2 have larger relative errors over the testing period using both approaches~~of both approaches are small for all gap lengths. Both approaches appear to have larger error outliers at well 2-3.

In addition to the error statistics, it is also important to examine how well a gap-filling method captures the desired ~~dynamic~~ dynamics patterns in the gap-filled time series. Therefore, the SpC time series reproduced by the gap-filling methods during the testing period (2016 for wells 1-1, 1-10A, 2-2, 2-3; 2017 for well 1-15; 2008 for well 2-5) with 24-hour synthetic gaps are evaluated against the real time series. Model configurations are the same as those used in error statistics comparison (Figure 6). A gap length of 24 hours is selected as an example because we consider it as a reasonably challenging case to fill gaps in time series data exhibiting significant nonstationarity, such as the SpC data at well 2-3. Moreover, the relative performance between the two approaches are similar at other gap lengths with varying error magnitudes.

As shown in the first ~~and second columns~~ column of Figure 8, ~~the~~ both approaches capture the general dynamic patterns in the data fairly well. The time series of relative errors for both methods are provided in Figure S3 in the online supplemental materials for more details. The ARIMA approach (blue lines in column 1) ~~can capture the smooth changes in the observations but not~~ missed some abrupt changes that occur over a short time window (i.e., at higher frequency)~~; these occur~~ , leading to more error spikes in all wells~~except 1-15. The spikes in errors during those rapid changes were not captured in Figure 7as they are outside of the interquartile ranges of the relative errors~~, consistent with relative error outliers in Figure 7. This is an indication that ~~ARIMA fails to capture~~ the ARIMA models lack mechanisms to represent such high-frequency ~~(daily and subdaily) dynamics and nonlinear trends despite having smaller error quantiles. The DNN approach~~ changes. The LSTM approach (red lines in column 1 plots), on the other hand, is able to better capture such dynamics in ~~some wells(e. g., wells 1-15, 2-3 and 2-5).~~ all the wells. However, the ~~DNN approach appears to overestimate the~~ inclusion of such high-frequency ~~(daily and subdaily) fluctuations in some wells near the river (i.e., wells 1-1, 1-10A, and 2-2),~~ fluctuations may also lead to overly dynamic predictions in time windows dominated by lower-frequency fluctuations, which contributed to less desirable relative errors distributed between the first and third quartiles ~~(7)~~in some wells (i.e., wells 1-1, 2-3, 2-2, and 1-15), as shown in Figure S4 in the supplemental materials. This is likely caused by the variability in dynamics signatures among the training, validation and ~~test periods. For well 1-15, which exhibits less dynamic behavior in SpC, both gap filling methods perform well in terms of both the relative errors and capturing the dynamic patterns.~~ testing periods, as well as the selection of training loss functions and validation metric that balance between the occurrence of small vs large errors to achieve optimal solutions.

To further investigate how the relative performance of the two gap-filling methods depends on the inherent dynamics in each time series, spectral analyses for the testing SpC datasets were performed using the same wavelet decomposition method for the multi-year analyses (shown earlier in Figure 2). As shown in Figure 8, the time windows of high relative errors are found

to approximately co-locate with the time when the high-frequency (daily and subdaily) signals are gaining more power. The difference between the ~~DNN~~ LSTM and ARIMA models tend to be amplified during those time windows. Wells 1-1, 1-10A, and 2-2 share similar seasonal patterns in WPS, with the highest intensity bin above 1024 hours across February to July. Their average WPSs all show peaks around daily and subdaily frequencies. Well 2-3 has its greatest energy between 16 to 256 hours from January to July. Well 2-5 has low intensities of variability at daily and subdaily frequencies with the low-frequency variations (monthly and seasonal) dominating the Jan to March time frame. For well 1-15, one of its strongest intensities is above 2048 hours across the entire year, and the other strong intensities are narrow bands between 16 to 256 hours. In general, both ~~DNN~~ LSTM and ARIMA are effective at capturing low-frequency variability (monthly and seasonal). Although ~~DNN~~ LSTM is more effective at capturing high-frequency (daily and subdaily) fluctuations and nonlinearities in the datasets, it may also lead to overly dynamic predictions ~~when the training data contain more significant~~ in time windows with no considerable high-frequency ~~signatures than the system behavior to be predicted~~fluctuations. However, the errors during these time windows are small and can be improved by smoothing if such fluctuations are not desirable.

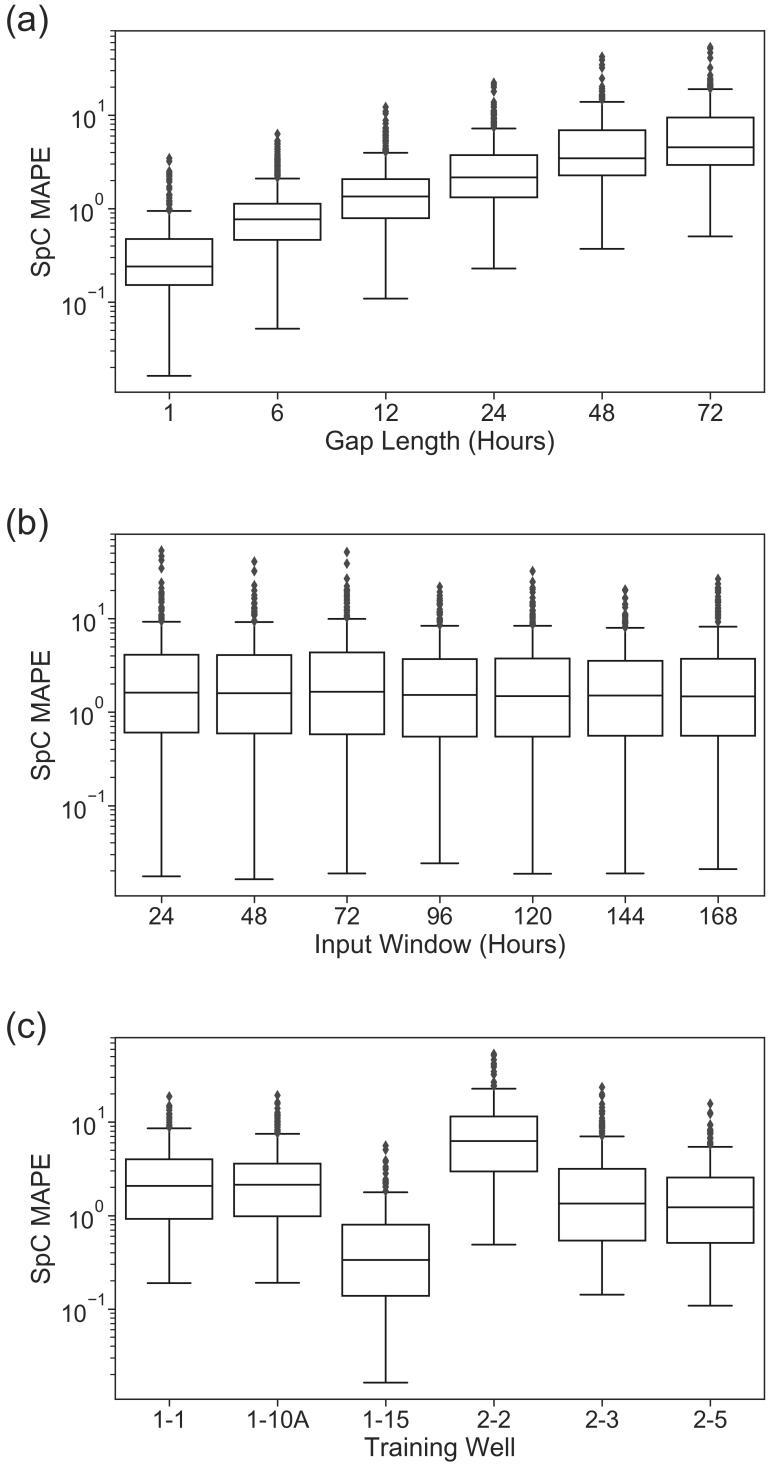

Figure 5. Gap filling performance for SpC evaluated against the validation datasets~~under multiple model configuration parameters (a-c) or~~ , grouped by ~~training wells (d). (a) distribution of SpC MAPE vs. tested~~ gap lengths ~~;~~ (~~b~~a)~~distribution of SpC MAPE vs.~~ , model input window size M ~~;~~ (~~e~~b)~~distribution of SpC MAPE vs. model output window size~~ N~~;~~ , and training wells (~~d~~c)~~distribution of SpC MAPE aggregated by wells~~.

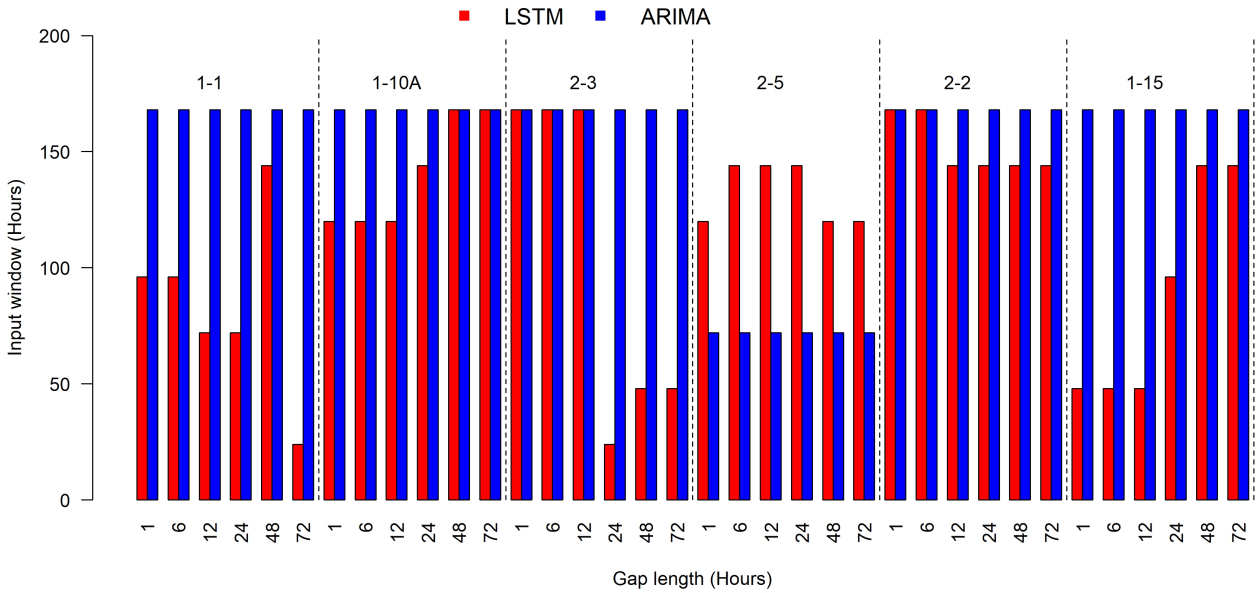

Figure 6. ~~Best~~ Optimal input ~~and output~~ windows for ~~DNN~~ the LSTM and ARIMA models ~~for filling~~ to fill gaps of various lengths at each well.

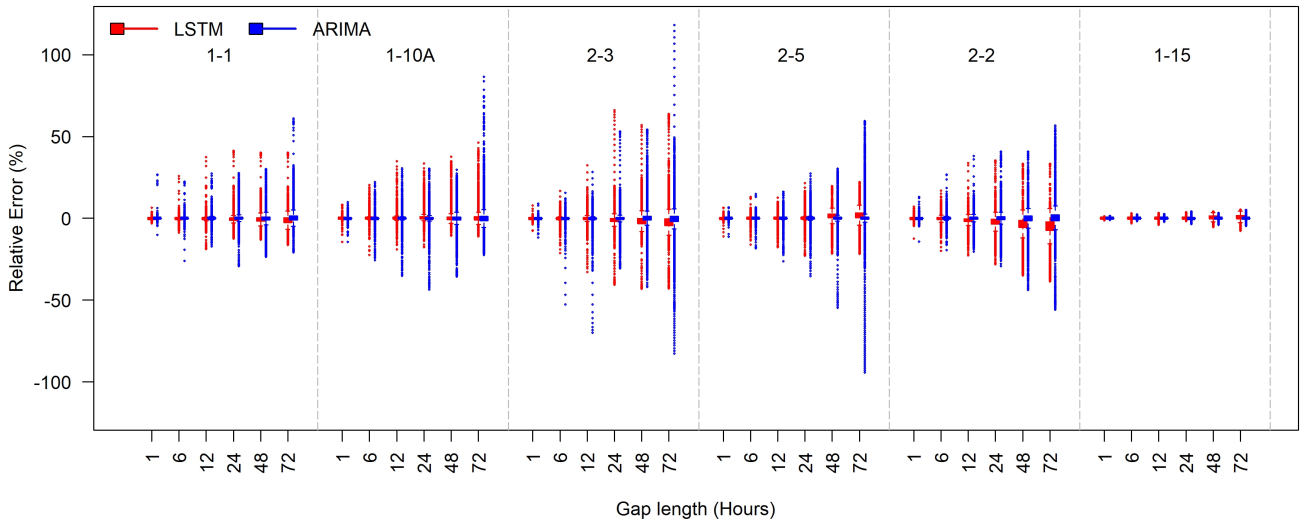

Figure 7. ~~Summary~~ Boxplots of ~~SpC~~ relative errors for filling SpC gaps of various lengths (~~i.e.~~ 1, 6, 12, 24, 48, and 72 hours) ~~for~~ at each well during the test periods. The best ~~DNN~~ LSTM and ARIMA models ~~tested~~ were used for ~~each well~~evaluation. The LSTM and ARIMA models are represented by red bars and blue bars, respectively.

~~Columns 1 and 2 show time series of model predictions (in red) from ARIMA and DNN methods, respectively, assuming 24-hour synthetic gap in the SpC data, compared with observations (in black) and the relative errors (in blue). The best model configurations were used for all models. The testing data come from year 2016 for wells 1-1, 1-10A, 2-2, and 2-3, from year 2017 for well 1-15 and from 2008 for well 2-5. Column 3 is the spectrogram of each well and column 4 is the WPS averaged over for the corresponding year.~~

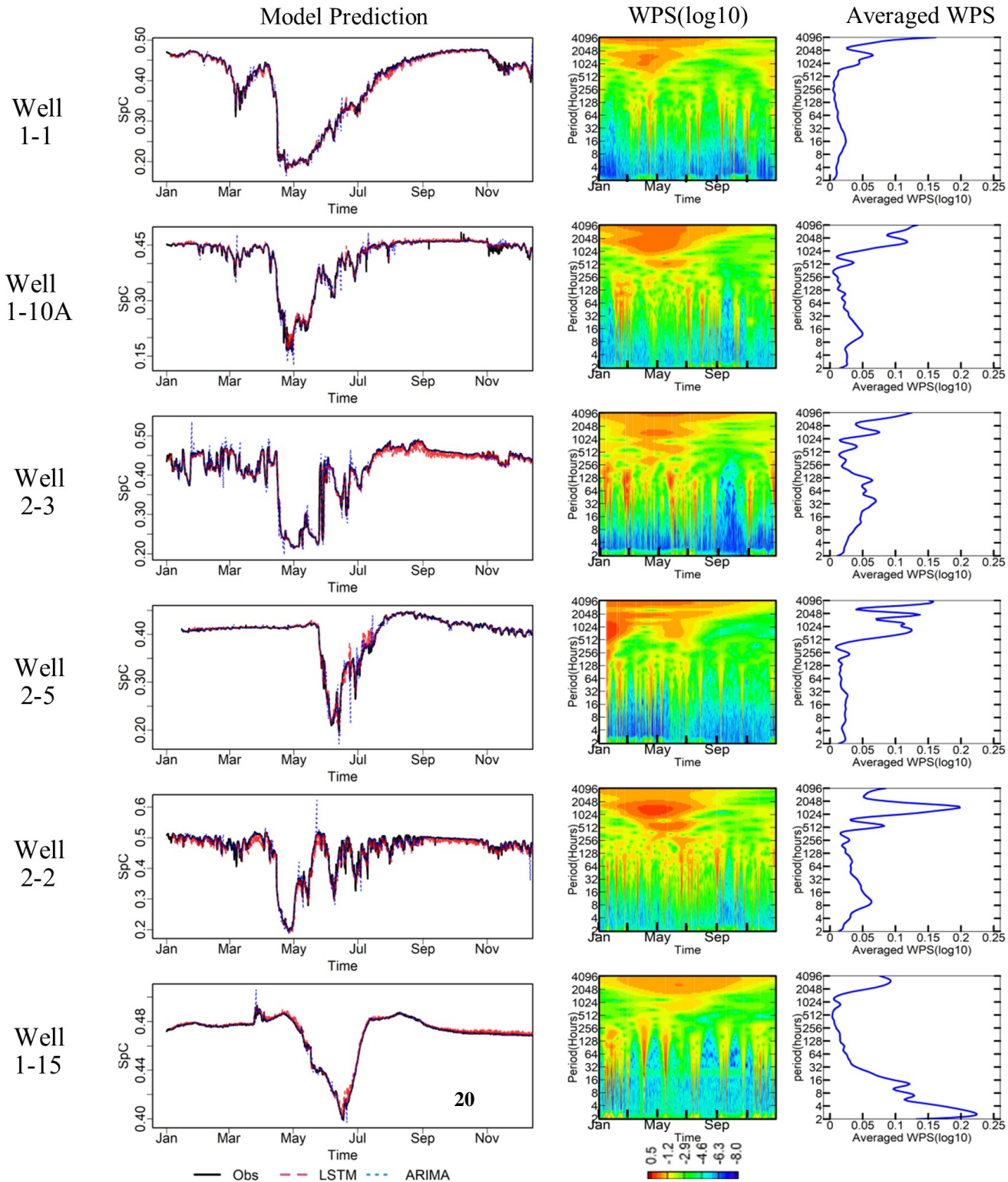

Figure 8. Columns 1 shows time series of model predictions from the LSTM (in red) and ARIMA (in blue) methods, respectively, assuming 24-hour synthetic gap in the SpC data, compared with observations (in black). The best model configurations were used for all models. The testing data come from year 2016 for wells 1-1, 1-10A, 2-2, and 2-3, from year 2017 for well 1-15 and from 2008 for well 2-5. Column 2 is the spectrogram of each well and column 3 is the WPS averaged over for the corresponding year.

There is also significant difference in computational cost between the ~~DNN~~ LSTM and ARIMA methods for gap filling. ARIMA requires very little computational resources: the `auto.arima` function in R requires approximately 40 seconds for fit and validate a model ~~using one year of data~~ for each prediction segments on a personal computer with a 3.00 GHz CPU. Conversely, training and validating a single ~~DNN~~ LSTM model takes approximately 20-30 minutes on dual NVIDIA P100 12GB PCI-e based GPUs.

4.3 Performance of multi-well ~~DNN~~ models

We evaluated the predictive ability of the multi-well ~~DNN models~~ models using both approaches in filling gaps of various lengths in the SpC data at well 1-1 by comparing the performance against their single-well counterparts. Well 1-1 was chosen because of data availability in nearby wells (wells 1-10A and 1-16A). ~~Moreover, both the ARIMA and single-well DNN methods had difficulty in capturing its dynamic patterns as discussed in Section ??.~~ Similar to the single-well ~~DNN~~ ARIMA and LSTM model for well 1-1, the multi-well ~~DNN~~ models also predict the ~~three variables for the well 1-1 only~~SpC measurement using water level and SpC from three wells. We adopted the same ~~DNN~~ LSTM architecture from the single-well ~~models~~ LSTM model and trained the same set of alternatives considering input window sizes~~and output window sizes~~, LSTM units, and learning rates for various gap lengths as listed in Table 1. ~~Only output window sizes smaller than 24 hours were considered as learnt from the single-well models.~~ The same training and validation periods were adopted to select the optimal combination of M~~and N. Results were summarized in Figure 9, where each boxplot was generated in the same manner as in Figure 6.~~, U, and L. For the multi-well ARIMA models, we included additional variables as regression terms when building and fitting models using the `auto.arima` function.

~~Compared to~~ The boxplots of relative errors yielded from the single-well ~~DNN models , the multi-well DNN models significantly improve the gap filling accuracy at well 1-1 with longer gaps (48~~ and ~~72 hours) while perform comparably with smaller gaps (i.e.~~multi-well models using both approaches are provided in Figure 9 for comparison. Additional spatial information helps the ARIMA models to reduce the large errors when the gaps are small (e.g., 1 and ~~24~~ 6 hours), ~~as shown in Figure 9 (a). The multi-well DNN models reduce the fraction of larger MAPEs under all the input window sizes (figure 9 (b)) and all output windows (figure 9 (c)). Further one to one comparisons between different M and N combinations are provided in figure ?? under different gap scenarios. Natural log scales were on both axes for better separations in data points. All the points below the 1:1 line represent cases where a multi-well DNN outperforms a single-well DNN. The percentage of points below the 1:1 line increases with the gap lengths: 17.9%, 35.7%, 64.3%, and 82.1% for gaps of 1, 24, 48, and 72 hours, respectively~~while it exacerbates the errors for gaps larger than 6 hours. The LSTM approach, on the contrary, benefits from the information carried by the neighboring wells to fill in those larger gaps, while the performance for small gaps stay unchanged,

indicating that the information from a single well is sufficient to fill in those small gaps. Therefore, including spatial information from neighbouring wells in the LSTM models could potentially increase the chance of successes in filling data gaps under more challenging circumstances, such as capturing more complex dynamic patterns ~~and~~ with longer data gaps.

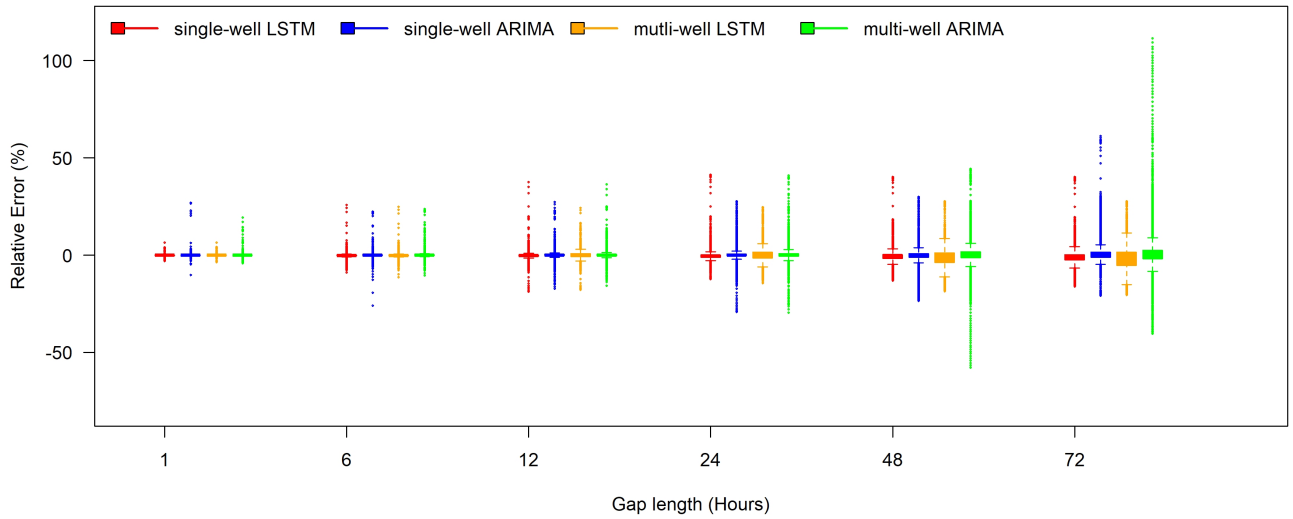

Figure 9. Comparing relative error performance between the best single-well LSTM models (well 1-1, red)~~and~~ , multi-well ~~DNN~~ LSTM models (wells 1-1, 1-10A and 1-16A, yellow), single-well ARIMA model (blue), and multi-well ARIMA model (green) for filling in various SpC ~~data gaps~~ gap lengths for well 1-1 during the ~~testing~~ test period (year ~~2011~~2016).~~(a) distribution of SpC MAPE vs. tested gap lengths; (b) distribution of SpC MAPE vs. model input window size M; (c) distribution of SpC MAPE vs. model output window size N; (d) distribution of SpC MAPE aggregated by wells used to train the models~~

~~Comparing performance between single-well (well 1-1) and multi-well DNN models (wells 1-1, 1-10A and 1-16A) for filling SpC data gaps of various lengths for well 1-1 during the testing period (year 2011). The subplots (a)-(d) correspond to a gap length of 1, 24, 48 and 72 hours, respectively. Each data point represents a unique model configuration (size of input window, size of output window). The x-axis and y-axis are the natural log of SpC MAPE for single-well and multi-well DNN models, respectively. The dashed black line in each plot is the 1:1 line.~~

5 Conclusion

In this study, we implemented ~~a DNN-based~~ an LSTM-based gap filling method to account for spatio-temporal correlations in monitoring data. We extensively evaluated the ~~new method in~~ method on filling data gaps in groundwater SpC measurements that are often used to indicate groundwater and river water interactions along river corridors. We optimized ~~a DNN architecture that contains stacked LSTM , convolutional, and dense layers~~ an LSTM architecture to take advantage of a 10-year spatially distributed multi-variable time series dataset collected by a groundwater monitoring well network for filling SpC data

gaps. A primary advantage of using ~~DNN~~ LSTM is the ability to incorporate spatio-temporal correlations and nonlinearity in model states without assuming a priori an explicit form of correlations or nonlinear functions in advancing system states~~as a priori~~. We compared the performance of single-well ~~DNN-based~~ LSTM-based gap-filling method with a traditional gap-filling method, ARIMA, to evaluate how well ~~a DNN~~ an LSTM model can capture multi-frequency dynamics. We also trained ~~DNN~~ LSTM and ARIMA models that take input from multiple wells to predict responses at one well. The multi-well ~~DNN~~ models were compared with single-well models to assess the improvement in gap filling performance by including additional spatial correlation from neighboring wells.

In general, both ~~DNN and ARIMA~~ LSTM and ARIMA methods were highly accurate in filling ~~small~~ smaller data gaps (i.e., 1 ~~hour~~and 6 hours). They were reasonably effective at filling in ~~gaps of 24, 48,~~ medium gaps between 12 and ~~72 hours. The relative errors were mostly within the range of instrument measurement error.~~ 48 hours, while more work is needed for gaps larger than 48 hours. Both models captured the long-term trends in data (i.e., low-frequency variations at the monthly or seasonal time scales)~~, except during some time windows with highly dynamic fluctuations~~. The ARIMA method was found to ~~be~~ have difficulty in capturing abrupt changes. Thus, it is more suitable for time series with less dynamic behavior. ~~DNNs~~ Compared with the ARIMA models, the LSTM models excel in dealing with high-frequency dynamics (daily and subdaily) ~~or~~ and nonlinearities, although they require more training data and computational resources. ~~The DNN approach also appeared to overestimate the~~ As a side effect of including high-frequency (daily and subdaily) fluctuations in ~~some wells near the river (i.e., wells 1-1, 1-10A, and 2-2), which was likely caused by the variability in dynamics signatures among the training, validation and test periods. Availability~~ the model, the LSTM approach may produce overly dynamic predictions in time windows that lacks such dynamics. Thus, availability of sufficient training data that cover a wide range of conditions is critical for the success of ~~DNN-based~~ LSTM methods, as is with any ~~DNN-based~~ deep learning method. Extrapolating the ~~DNN~~ LSTM models to conditions beyond those in the training data remains as a major challenge.

Wavelet analysis could provide useful insights to the dynamic signatures of the data and the change in composition of their important frequencies over time, which can serve as a prior basis for selecting an appropriate gap-filling method. For example, the ARIMA method would work well if the dynamics are dominated by seasonal cycles, while more sophisticated approaches like ~~DNN-based~~ LSTM-based methods could work better if there is evidence of weekly, daily and subdaily fluctuations. Depending on the mixture of high- and low-frequency variability inherent in the time series, different ~~DNN~~ LSTM architecture and configurations can be explored and evaluated through hyperparameter ~~searches~~ search with respect to LSTM layers, dense layers and activation functions to achieve better performance in capturing more complex dynamics. The optimal LSTM model configuration and performance that could be achieved would vary case by case.

We also demonstrated that incorporating spatial information from neighboring stations in ~~DNN~~ LSTM models could contribute to performance improvement under challenging scenarios with dynamic system behaviours with reasonably longer data gaps ~~(multiple days). The optimal DNN model configuration and performance that could be achieved would vary case by case~~up to 2 days. However, other alternatives need to be explored for gaps beyond 2 days. The bidirectional ~~LSTM can be explored to formulate the gap filling as a history matching problem and evaluate the value of observed data in~~ and convolutional LSTMs are two promising methods to leverage information from the future time window ~~relative to missing data for filling the data~~

~~gaps~~and from spatially distributed networks. While we introduced a new method that can be broadly applied to fill in gaps in irregularly spaced network for monitoring groundwater and surface water interactions, the transferrability of this method to other monitoring systems could be evaluated more extensively by community participation. Capturing spatio-temporal dynamics in system states is essential for generating the most valuable insights to advance our understanding of dynamic complex systems.

Code and data availability. The well observations have been made accessible at https://sbrsfa.velo.pnnl.gov/datasets/?UUID=14febd81-05b6-47fb-be52 -439c4382decd

Author contributions. HR and EC developed scripts and performed the analyses. BK contributed on interpretation of the results. XC conceived and designed the study. All authors contributed to writing the manuscript.

Competing interests. The authors declare that they have no conflicts of interest.

Acknowledgements. This research was supported by the U.S. Department of Energy (DOE), Office of Biological and Environmental Research (BER), as part of BER's Subsurface Biogeochemical Research Program (SBR). A portion of methodology development was supported by the Laboratory Directed Research and Development Program at Pacific Northwest National Laboratory, a multiprogram national laboratory operated by Battelle for DOE under contract DE-AC05-76RL01830. This research was performed using PNNL Institutional Computing at Pacific Northwest National Laboratory. This research was also supported in part by the Indiana University Environmental Resilience Institute and the *Prepared for Environmental Change* grand challenge initiative. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

References

Alvera-Azcárate, A., Barth, A., Parard, G., and Beckers, J.-M.: Analysis of SMOS sea surface salinity data using DINEOF, Remote sensing of environment, 180, 137–145, 2016.

Amaranto, A., Munoz-Arriola, F., Corzo, G., Solomatine, D. P., and Meyer, G.: Semi-seasonal groundwater forecast using multiple data-driven models in an irrigated cropland, Journal of Hydroinformatics, 20, 1227–1246, 2018.

Amaranto, A., Munoz-Arriola, F., Solomatine, D., and Corzo, G.: A spatially enhanced data-driven multimodel to improve semiseasonal groundwater forecasts in the High Plains aquifer, USA, Water Resources Research, 55, 5941–5961, 2019.

Banerjee, S., Carlin, B. P., and Gelfand, A. E.: Hierarchical modeling and analysis for spatial data, CRC press, 2014.

Beckers, J.-M. and Rixen, M.: EOF calculations and data filling from incomplete oceanographic datasets, Journal of Atmospheric and oceanic technology, 20, 1839–1856, 2003.

Beckers, J.-M., Barth, A., and Alvera-Azcárate, A.: DINEOF reconstruction of clouded images including error maps? application to the Sea-Surface Temperature around Corsican Island, 2006.

Calculli, C., Fassò, A., Finazzi, F., Pollice, A., and Turnone, A.: Maximum likelihood estimation of the multivariate hidden dynamic geostatistical model with application to air quality in Apulia, Italy, Environmetrics, 26, 406–417, 2015.

Chen, X., Murakami, H., Hahn, M. S., Hammond, G. E., Rockhold, M. L., Zachara, J. M., and Rubin, Y.: Three-dimensional Bayesian geostatistical aquifer characterization at the Hanford 300 Area using tracer test data, Water Resources Research, 48, https://doi.org/10.1029/2011WR010675, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2011WR010675, 2012.

Chen, X., Hammond, G. E., Murray, C. J., Rockhold, M. L., Vermeul, V. R., and Zachara, J. M.: Application of ensemble-based data assimilation techniques for aquifer characterization using tracer data at Hanford 300 area, Water Resources Research, 49, 7064–7076, https://doi.org/10.1002/2012WR013285, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2012WR013285, 2013.

Cheng, T., Haworth, J., and Wang, J.: Spatio-temporal autocorrelation of road network data, Journal of Geographical Systems, 14, 389–413, https://doi.org/10.1007/s10109-011-0149-5, https://doi.org/10.1007/s10109-011-0149-5, 2012.

Cheng, T., Haworth, J., Anbaroglu, B., Tanaksaranond, G., and Wang, J.: Spatiotemporal data mining, in: Handbook of regional science, pp. 1173–1193, Springer, 2014.

Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E.: Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets, Journal of the American Statistical Association, 111, 800–812, 2016.

Eidsvik, J., Shaby, B. A., Reich, B. J., Wheeler, M., and Niemi, J.: Estimation and prediction in spatial models with block composite likelihoods, Journal of Computational and Graphical Statistics, 23, 295–315, 2014.

Fang, K., Shen, C., Kifer, D., and Yang, X.: Prolongation of SMAP to Spatiotemporally Seamless Coverage of Continental U.S. Using a Deep Learning Neural Network, Geophysical Research Letters, 44, 11,030–11,039, https://doi.org/10.1002/2017GL075619, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2017GL075619, 2017.

Faruk, D. Ö.: A hybrid neural network and ARIMA model for water quality time series prediction, Engineering Applications of Artificial Intelligence, 23, 586–594, 2010.

Finley, A. O., Banerjee, S., and Gelfand, A. E.: spBayes for large univariate and multivariate point-referenced spatio-temporal data models, arXiv preprint arXiv:1310.8192, 2013.

Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., and Yacalis, G.: Could Machine Learning Break the Convection Parameterization Deadlock?, Geophysical Research Letters, 45, 5742–5751, https://doi.org/10.1029/2018GL078202, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018GL078202, 2018.

Ghil, M., Allen, M., Dettinger, M., Ide, K., Kondrashov, D., Mann, M., Robertson, A. W., Saunders, A., Tian, Y., Varadi, F., et al.: Advanced spectral methods for climatic time series, Reviews of geophysics, 40, 3–1, 2002.

Grant, G. E. and Dietrich, W. E.: The frontier beneath our feet, Water Resources Research, 53, 2605–2609, 2017.

Graves, A.: Generating sequences with recurrent neural networks, arXiv preprint arXiv:1308.0850, 2013.

Graves, A., Mohamed, A., and Hinton, G.: Speech recognition with deep recurrent neural networks, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6645–6649, https://doi.org/10.1109/ICASSP.2013.6638947, 2013.

Graves, A., Mohamed, A.-r., and Hinton, G.: Speech recognition with deep recurrent neural networks, in: Acoustics, speech and signal processing (icassp), 2013 ieee international conference on, pp. 6645–6649, IEEE, 2013.

Griffith, D. A.: Modeling spatio-temporal relationships: retrospect and prospect, Journal of Geographical Systems, 12, 111–123, 2010.

Grinsted, A., Moore, J. C., and Jevrejeva, S.: Application of the cross wavelet transform and wavelet coherence to geophysical time series, Nonlinear processes in geophysics, 11, 561–566, 2004.

Grossmann, A. and Morlet, J.: Decomposition of Hardy functions into square integrable wavelets of constant shape, SIAM journal on mathematical analysis, 15, 723–736, 1984.

Güler, C. and Thyne, G. D.: Hydrologic and geologic factors controlling surface and groundwater chemistry in Indian Wells-Owens Valley area, southeastern California, USA, Journal of Hydrology, 285, 177–198, 2004.

Han, P., Wang, P. X., Zhang, S. Y., and Zhu, D. H.: Drought forecasting based on the remote sensing data using ARIMA models, Mathematical and Computer Modelling, 51, 1398–1403, 2010.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R.: Improving neural networks by preventing co-adaptation of feature detectors, arXiv preprint arXiv:1207.0580, 2012.

Ho, S., Xie, M., and Goh, T.: A comparative study of neural network and Box-Jenkins ARIMA modeling in time series prediction, Computers & Industrial Engineering, 42, 371–375, 2002.

Hochreiter, S. and Schmidhuber, J.: Long Short-Term Memory, Neural Comput., 9, 1735–1780, https://doi.org/10.1162/neco.1997.9.8.1735, https://doi.org/10.1162/neco.1997.9.8.1735, 1997.

Hochreiter, S., Bengio, Y., Frasconi, P., Schmidhuber, J., et al.: Gradient flow in recurrent nets: the difficulty of learning long-term dependencies, 2001.

Hocke, K. and Kämpfer, N.: Gap filling and noise reduction of unevenly sampled data by means of the Lomb-Scargle periodogram, Atmospheric Chemistry and Physics Discussions, 8, 4603–4623, 2008.

Hyndman, R. J., Khandakar, Y., et al.: Automatic time series for forecasting: the forecast package for R, 6/07, Monash University, Department of Econometrics and Business Statistics, 2007.

JORDAN, M.: Attractor dynamics and parallelism in a connectionist sequential machine, Proc. of the Eighth Annual Conference of the Cognitive Science Society (Erlbaum, Hillsdale, NJ), 1986, https://ci.nii.ac.jp/naid/10018634949/en/, 1986.

Kamarianakis, Y. and Prastacos, P.: Forecasting traffic flow conditions in an urban network: Comparison of multivariate and univariate approaches, Transportation Research Record: Journal of the Transportation Research Board, pp. 74–84, 2003.

Kamarianakis, Y. and Prastacos, P.: Space–time modeling of traffic flow, Computers & Geosciences, 31, 119–133, 2005.

Katzfuss, M. and Cressie, N.: Spatio-temporal smoothing and EM estimation for massive remote-sensing data sets, Journal of Time Series Analysis, 32, 430–446, 2011.

Katzfuss, M. and Cressie, N.: Bayesian hierarchical spatio-temporal smoothing for very large datasets, Environmetrics, 23, 94–107, 2012.

Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, CoRR, abs/1412.6980, http://arxiv.org/abs/1412.6980, 2014.

Kondrashov, D. and Ghil, M.: Spatio-temporal filling of missing points in geophysical data sets, Nonlinear Processes in Geophysics, 13, 151–159, 2006.

Kondrashov, D., Shprits, Y., and Ghil, M.: Gap filling of solar wind data by singular spectrum analysis, Geophysical research letters, 37, 2010.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K., and Herrnegger, M.: Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks, Hydrology and Earth System Sciences, 22, 6005–6022, https://doi.org/10.5194/hess-22-6005-2018, https://www.hydrol-earth-syst-sci.net/22/6005/2018/, 2018.

Lin, C. Y., Abdullah, M. H., Praveena, S. M., Yahaya, A. H. B., and Musta, B.: Delineation of temporal variability and governing factors influencing the spatial variability of shallow groundwater chemistry in a tropical sedimentary island, Journal of hydrology, 432, 26–42, 2012.

Längkvist, M., Karlsson, L., and Loutfi, A.: A review of unsupervised feature learning and deep learning for time-series modeling, Pattern Recognition Letters, 42, 11 – 24, https://doi.org/https://doi.org/10.1016/j.patrec.2014.01.008, http://www.sciencedirect.com/science/article/pii/S0167865514000221, 2014.

Olah, C.: Understanding LSTM Networks, https://colah.github.io/posts/2015-08-Understanding-LSTMs/, 2015.

Pascanu, R., Gulcehre, C., Cho, K., and Bengio, Y.: How to Construct Deep Recurrent Neural Networks, 2013.

Pfeifer, P. E. and Deutrch, S. J.: A three-stage iterative procedure for space-time modeling phillip, Technometrics, 22, 35–47, 1980.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., and Prabhat: Deep learning and process understanding for data-driven Earth system science, Nature, 566, 195–204, https://doi.org/10.1038/s41586-019-0912-1, http://www.nature.com/articles/s41586-019-0912-1, 2019.

Saleh, K., Hossny, M., and Nahavandi, S.: Intent prediction of vulnerable road users from motion trajectories using stacked LSTM network, in: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 327–332, https://doi.org/10.1109/ITSC.2017.8317941, 2017.

Schmidhuber, J.: Deep learning in neural networks: An overview, Neural Networks, 61, 85 – 117, https://doi.org/https://doi.org/10.1016/j.neunet.2014.09.003, http://www.sciencedirect.com/science/article/pii/S0893608014002135, 2015.

Schuster, M. and Paliwal, K. K.: Bidirectional recurrent neural networks, IEEE Transactions on Signal Processing, 45, 2673–2681, 1997.

Shen, C.: A Transdisciplinary Review of Deep Learning Research and Its Relevance for Water Resources Scientists, Water Resources Research, 54, 8558–8593, https://doi.org/10.1029/2018WR022643, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018WR022643, 2018.

Shuai, P., Chen, X., Song, X., Hammond, G. E., Zachara, J., Royer, P., Ren, H., Perkins, W. A., Richmond, M. C., and Huang, M.: Dam Operations and Subsurface Hydrogeology Control Dynamics of Hydrologic Exchange Flows in a Regulated River Reach, Water Resources Research, 55, 2593–2612, https://doi.org/10.1029/2018WR024193, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018WR024193, 2019.

Song, X., Chen, X., Stegen, J., Hammond, G., Song, H.-S., Dai, H., Graham, E., and Zachara, J. M.: Drought Conditions Maximize the Impact of High-Frequency Flow Variations on Thermal Regimes and Biogeochemical Function in the Hyporheic Zone, Water Resources Research, 2018.

Stockwell, R. G., Mansinha, L., and Lowe, R.: Localization of the complex spectrum: the S transform, IEEE transactions on signal processing, 44, 998–1001, 1996.

Strobl, R. O. and Robillard, P. D.: Network design for water quality monitoring of surface freshwaters: A review, Journal of environmental management, 87, 639–648, 2008.

Stroud, J. R., Stein, M. L., and Lysen, S.: Bayesian and maximum likelihood estimation for Gaussian processes on an incomplete lattice, Journal of computational and Graphical Statistics, 26, 108–120, 2017.

Sun, A. Y.: Discovering State-Parameter Mappings in Subsurface Models Using Generative Adversarial Networks, Geophysical Research Letters, 45, 11,137–11,146, https://doi.org/10.1029/2018GL080404, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018GL080404, 2018.

Sun, A. Y., Scanlon, B. R., Zhang, Z., Walling, D., Bhanja, S. N., Mukherjee, A., and Zhong, Z.: Combining Physically Based Modeling and Deep Learning for Fusing GRACE Satellite Data: Can We Learn From Mismatch?, Water Resources Research, 55, 1179–1195, https://doi.org/10.1029/2018WR023333, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018WR023333, 2019.

Taylor, C. J. and Alley, W. M.: Ground-water-level monitoring and the importance of long-term water-level data, 1217-2002, US Geological Survey, 2002.

Vacha, L. and Barunik, J.: Co-movement of energy commodities revisited: Evidence from wavelet coherence analysis, Energy Economics, 34, 241–247, 2012.

Valenzuela, O., Rojas, I., Rojas, F., Pomares, H., Herrera, L. J., Guillén, A., Marquez, L., and Pasadas, M.: Hybridization of intelligent techniques and ARIMA models for time series prediction, Fuzzy sets and systems, 159, 821–845, 2008.

Wang, G., Garcia, D., Liu, Y., De Jeu, R., and Dolman, A. J.: A three-dimensional gap filling method for large geophysical datasets: Application to global satellite soil moisture observations, Environmental Modelling & Software, 30, 139–142, 2012.

Wett, B., Jarosch, H., and Ingerle, K.: Flood induced infiltration affecting a bank filtrate well at the River Enns, Austria, Journal of Hydrology, 266, 222–234, 2002.

Wikle, C. K., Berliner, L. M., and Cressie, N.: Hierarchical Bayesian space-time models, Environmental and Ecological Statistics, 5, 117–154, 1998.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Łukasz Kaiser, Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., and Dean, J.: Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, CoRR, abs/1609.08144, http://arxiv.org/abs/1609.08144, 2016.

You, Q., Jin, H., Wang, Z., Fang, C., and Luo, J.: Image captioning with semantic attention, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4651–4659, 2016.

Zachara, J. M., Long, P. E., Bargar, J., Davis, J. A., Fox, P., Fredrickson, J. K., Freshley, M. D., Konopka, A. E., Liu, C., McKinley, J. P., et al.: Persistence of uranium groundwater plumes: contrasting mechanisms at two DOE sites in the groundwater–river interaction zone, Journal of contaminant hydrology, 147, 45–72, 2013.

Zachara, J. M., Chen, X., Song, X., Shuai, P., Murray, C., and Resch, C. T.: Kilometer-scale hydrologic exchange flows in a gravel-bed river corridor and their implications to solute migration, Water Resources Research, n/a, e2019WR025 258,

https://doi.org/10.1029/2019WR025258, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2019WR025258, e2019WR025258 2019WR025258, 2020.

Zhang, D., Lindholm, G., and Ratnaweera, H.: Use long short-term memory to enhance Internet of Things for combined sewer overflow monitoring, Journal of Hydrology, 556, 409 – 418, https://doi.org/https://doi.org/10.1016/j.jhydrol.2017.11.018, http://www.sciencedirect.com/science/article/pii/S0022169417307722, 2018.

Zhang, G. P.: Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing, 50, 159–175, 2003.

Zhu, W., Lan, C., Xing, J., Zeng, W., Li, Y., Shen, L., and Xie, X.: Co-Occurrence Feature Learning for Skeleton Based Action Recognition Using Regularized Deep LSTM Networks, https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11989/12149, 2016.