Journal of Geodesy (2022) 96:71
https://doi.org/10.1007/s00190-022-01662-5

ORIGINAL ARTICLE

Modelling and prediction of GNSS time series using GBDT, LSTM and SVM machine learning approaches

Wenzong Gao[1] · Zhao Li[2] · Qusen Chen[2] · Weiping Jiang[2] · Yanming Feng[1]

Received: 28 February 2022 / Accepted: 27 August 2022 / Published online: 27 September 2022

Abstract

Global navigation satellite system (GNSS) site coordinate time series provides essential data for geodynamic and geophysical studies, realisation of a regional or global geodetic reference frames, and crustal deformation research. The coordinate time series has been conventionally modelled by least squares (LS) fitting with harmonic functions, alongside many other analysis methods. As a key limitation, the traditional modelling approaches simply use the functions of time variable, despite good knowledge of various underlying physical mechanisms responsible for the site displacements. This paper examines the use of machine learning (ML) models to reflect the effects or residential effects of physical variables related to Sun and the Moon ephemerides, polar motion, temperature, atmospheric pressure, and hydrology on the site displacements. To form the ML problem, these variables are constructed as the input vector of each ML training sample, while the vertical displacement of a GNSS site is regarded as the output value. In the evaluation experiments, three ML approaches, namely the gradient boosting decision tree (GBDT) approach, long short-term memory (LSTM) approach, and support vector machine (SVM) approach, are introduced and evaluated with the time series datasets collected from 9 GNSS sites over the period of 13 years. The results indicate that all three approaches achieve similar fitting precision in the range of 3–5 mm in the vertical displacement component, which is an improvement in over 30% with respect to the traditional LS fitting precision in the range of 4–7 mm. The prediction of the vertical time series with the three ML approaches shows the precision in the range of 4–7 mm over the future 24- month period. The results also indicate the relative importance of different physical features causing the displacements of each site. Overall, ML approaches demonstrate better performance and effectiveness in modelling and prediction of GNSS time series, thus impacting maintenance of geodetic reference frames, geodynamics, geophysics, and crustal deformation analysis.

Keywords GNSS time series · Modelling · Prediction · Machine learning · Gradient boosting decision tree · Long short-term memory · Support vector machine

1 Introduction

In the past three decades, tens of thousands of global navigation satellite system (GNSS) continuously operating reference stations (CORS) have been established worldwide. The coordinate time series derived from daily GNSS data of these CORSs have been used to study the tectonic motion, non-tectonic, inter-seismic strain, post-seismic deformation, slow slip events, surface mass-induced displacements, and other geodynamics-driven surface displacements. The coordinate time series from selected stable sites have also been used to establish and maintain a geodetic reference frame (Altamimi et al. 2011, 2016).

GNSS site motions are related to some physical variables, including the relative locations to Sun and the Moon,

✉ Wenzong Gao
wenzong.gao@hdr.qut.edu.au

Zhao Li
zhao.li@whu.edu.cn

Qusen Chen
chenqs@whu.edu.cn

Weiping Jiang
wpjiang@whu.edu.cn

Yanming Feng
y.feng@qut.edu.au

1 Faculty of Science, Queensland University of Technology, Brisbane 4000, Australia

2 GNSS Research Center, Wuhan University, Wuhan 430079, China

earth rotation parameters such as polar motion, temperature, atmospheric pressure, and hydrological parameters (Petit and Luzum 2010). Although these physical factors have somehow been dealt with in GNSS data processing for the GNSS daily coordinates, there are still residual effects and unconsidered factors. Dong et al. (2002) subtracted seasonal terms generated by known geophysical sources including the pole tide, ocean tide, atmospheric mass loading, non-tidal ocean mass loading, and snow and soil moisture mass loading. It was speculated that the residuals are caused by mismodelling of atmospheric delay, by bedrock thermal expansion, and by glacier surge and internal ice flow, depending on the locations of the sites and time. GNSS coordinate time series has been conventionally modelled by the least squares (LS) fitting method. In this way, the GNSS time series are fitted as a function of harmonic terms with constant annual and semi-annual amplitudes and phases, linear trend, jumps (offsets, steps, discontinuities), and other possible variables (Bock and Melgar 2016; Heflin et al. 2020). Related studies suggest that determining the time-varying seasonal signals may consider nonparametric annual signal (Tesmer et al. 2009; Freymueller 2009), or piecewise continuous linear polynomials (Davis et al. 2012), or singular spectrum analysis (Chen et al. 2013), or on a flexible semi-parametric model by Bennett (2008). However, there are still biases and large noises after removing the seasonal variation by LS fitting models. The LS fitting precision has remained between 4 and 6 mm in the vertical component for long time (Heflin et al. 2020; Davis et al. 2006; Chen et al. 2013). In addition, how well the LS model can predict the time series into future is rarely discussed.

In the traditional modelling of GNSS time series, the functional models are expressed as linear (or linearised) equations for unknown parameters and independent variables. For instance, in realisation of the international terrestrial reference frame (ITRF), the station position kinematic model contains parameters such as position offsets, velocity, amplitudes and phases of annul and semi-annul signals, and deformation breaks (Altamimi et al. 2018, 2021). The LS modelling of the time series is to determine the unknown parameters over a continuous time series period. The functional and statistical models are given and input data is processed to fit into the models and output the solutions. However, when we try to fit the GNSS coordinate time series with many other physical variables, regardless of independent and dependent variables, we cannot assume the linear relationship for the unknown parameters and potential impact factors or physical variables. With machine learning (ML), input data and expected solutions are trained to output the models, which are the relationships between the coordinate time series in three components and various physical variables. In other words, various physical factors or causes are taken into consideration without having the explicit mathematical equations. In addition, the traditional modelling approaches, such as LS and KF processing, data samples over a limited time period can effectively contribute to the modelling and prediction. In contrast, ML principles allow for making good use of historical data for modelling and prediction of GNSS time series. The challenge is how to properly apply ML approaches to modelling and prediction of the GNSS time series with different physical variables. We note some ML approaches have already been used to model and predict GNSS time series, with some success. For example, Alevizakou et al. (2018) precisely forecasted the position of 1000 GNSS stations in short and long terms using two types of artificial neural networks (ANNs), and Wang et al. (2021) achieved good prediction results for XJSS station using the long short-term memory (LSTM) approach, in which the prediction precision for 294 days is 3.2 mm. However, the essence of these studies is the same as that of traditional methods, that is, they only use the time variable to model the GNSS time series, without considering other physical variables related to site motions.

Some efforts have been made in using ML approaches to model the relationship between some physical factors and GNSS time series or residuals, for example, to correct environmental influences and improve the positioning accuracy. Mohammednour and Özdemir (2020) trained an ANN model by using surface meteorological parameters and number of observed satellites as inputs to eliminate troposphere delay. With application of this model, 3D position accuracy of the GNSS receiver is improved by about 20%. Ruttner et al. (2021) applied temporal convolutional neural network (TCN) to find connection between raw meteorological parameters and GNSS height residuals. They suggest that the trained TCN can achieve almost the same level with physical models on reduction of GNSS height residuals. Further, current research commonly uses meteorological parameters such as temperature, humidity, wind speed, and pressure to model GNSS time series or residuals. However, we argue that to precisely model GNSS time series, more comprehensive physical factors need to be used, as they can also have significant impact on the GNSS time series. Hydrologic surface loading which can produce displacements of several millimetres in GNSS time series (Herring et al. 2016) is one example of this. Therefore, to precisely model GNSS time series, more comprehensive physical factors are used in this research.

There have been some studies into the application of different ML approaches, such as boosting tree (BT), gradient boosting decision tree (GBDT), LSTM, and support vector machine (SVM), in addressing modelling problems in geodetic data analytics. Li et al. (2020) proposed using the BT algorithm to model the orbit prediction (OP) errors. This ML-derived OP error model was used to modify the future physics-based OP results. The errors of the physics-based OP results over the future 7 days achieve at least 50% accuracy

improvement with the application of the model. Furthermore, Li et al. (2021) applied the GBDT and convolutional neural networks (CNN) to model the underlying orbit error patterns. With the correction by model-predicted errors, the accuracy of OP over the future 14 days are improved by more than 75%, 90%, and 90% in the along-track, cross-track, and radial directions, respectively. Yang et al. (2019) proposed using a LSTM neural network-based dynamic model to predict landslide displacement. They claimed that the proposed LSTM-based model can effectively predict the displacement of stepwise landslides in the Three Gorges Reservoir Area. Dörterler and Faruk Bay (2018) applied the ML technique for the vehicular location prediction (VLP). They developed an ANN-based VLP model for cooperative active safety systems. Comparing the ANN-based model with other models which are based on KF and numerical integration methods (Caveney 2010), this ANN-based model shows better performance than others.

We hypothesise that ML principles can guide the modelling and prediction of GNSS time series well, due to the similar nature of error modelling problems as resolved in other applications. There are various well-developed ML algorithms over the past decades. In our experiments, the GBDT, LSTM, and SVM were chosen as they are considered to be suitable approaches, due to the proven performance in time series analysis. In using these ML approaches for GNSS time series modelling and prediction, the physical factors considered in this study include polar motion, the Sun and the Moon ephemerides, the temperature, the atmospheric pressure, and the hydrology. This contribution seeks to answer the key specific research questions in the workflow of machine learning: (1) how to define a ML problem with the input and output variables and how to prepare the datasets; (2) what modelling and prediction performance the ML models can achieve, and how much improvement would it be compared with the traditional LS fitting and prediction results.

Section 2 describes the GBDT, LSTM, and SVM principles and their evaluation methods. Section 3 prepares the input and output data for ML training and define the ML problem. In Sect. 4, the modelling and prediction results of the ML models and LS models are presented and evaluated. Finally, Sect. 5 concludes the paper.

2 Methodology

In a supervised ML problem, models are trained from labelled samples. These independent and identically distributed samples used for training are known as training data and can be represented as:

$$D_{\mathrm{Trn}} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\} \tag{1}$$

where N indicates the number of samples in the training dataset. And $x_n \in \mathbf{R}$, $n = 1, 2, \ldots, N$, represents the input vector. Each x_n consists of one or more features. In this paper, the features include a total of 12 variables: the time variable t, P_n^x and P_n^y variables representing polar motion, RA_n^S, DEC_n^S, and DEL_n^S variables representing astrometric right ascension, declination and apparent range of the Sun's centre with respect to the observing site in the international celestial reference frame (ICRF), RA_n^M, DEC_n^M, and DEL_n^M variables similarly representing astrometric right ascension, declination, and apparent range of the Moon's mass centre with respect to the observing site in the ICRF, TEM_n, AP_n, and HYD_n representing surface temperature, surface atmospheric pressure, and hydrology at the GNSS site. These variables are introduced in detail in Sect. 3.2. For non-ANN ML approaches such as GBDT and SVM, the input vector x_n can be represented as

$$\begin{aligned} x_n = [&t_n, P_n^x, P_n^y, \mathrm{RA}_n^S, \mathrm{DEC}_n^S, \mathrm{DEL}_n^S, \mathrm{RA}_n^M, \\ &\mathrm{DEC}_n^M, \mathrm{DEL}_n^M, \mathrm{TEM}_n, \mathrm{AP}_n, \mathrm{HYD}_n]^T. \end{aligned} \tag{2}$$

And the input matrix with two dimensions ($12 \times N$) can be expressed as $\boldsymbol{X} = (x_1, x_2, \ldots, x_N)$. It should be noticed that the input matrix for the LSTM approach has three dimensions as time-step is involved as another dimension. However, in this study, the time-step for the LSTM is set as 1. Therefore, the input matrix for the LSTM can also be regarded as a two-dimensional matrix in this case.

The $y_n \in \mathbf{R}$ is the corresponding output, or called response, which is the vertical coordinate of GNSS sites in this study. The purpose of ML is to find an underlying mapping relationship between $\boldsymbol{X} = (x_1, x_2, \ldots, x_N)$ and $Y = (y_1, y_2, \ldots, y_N)$, i.e. $\hat{Y} = \hat{f}(\boldsymbol{X})$, where $\hat{Y}$ is the predicted output, which usually differs slightly from Y because the ML model usually cannot represent the relationship with absolute accuracy. Once the relationship between $\boldsymbol{X}$ and Y is established, the prediction process gives the output $\hat{y}_{N+1}$ through $\hat{y}_{N+1} = \hat{f}(x_{N+1})$, where x_{N+1} is the corresponding new input vector.

In practice, the input vector x_n is usually normalised before training. Normalisation is a data preparation technique often applied for ML. The goal of normalisation is to change the feature values in the input dataset to a common scale, without distorting differences in the ranges of values (Jalal et al. 2020). There are two reasons that dataset should be normalised before ML training (Wang and Balog 2016): (1) objective functions of some ML algorithms will not work properly without normalisation if the input values vary widely; (2) gradient descent can converge faster with normalisation than without it. The standard score normalisation, also called z-score is applied in this study. If the mean and standard deviation (STD) of the input vector x_n

are known, the normalised input can be obtained via:

$$x_n' = \frac{x_n - \mu}{\sigma} \tag{3}$$

where μ and σ are the mean and STD of x_n, respectively.

A common challenge of ML modelling is to address the overfitting problem. Overfitting occurs when a ML model fits exactly against its training data but has poor fits for new datasets. This situation happens when training for too many epochs or when the model is too complex, leading to the "noise" or irrelevant information being learned by the ML model. To prevent overfitting, a part of the dataset is set aside as the "validation dataset" to check for overfitting. For convenience, training and validation datasets are referred to as the modelling dataset and denoted as D_{md}, as they are used together for training a model in this context. The mapping relationship $\hat{Y} = \hat{f}(\mathbf{X})$ can be different if different ML approaches are used. GBDT, LSTM and SVM approaches will be introduced in next subsections.

2.1 Gradient boosting decision tree

Boosting is an ensemble technique that converts weak learners (base learners) to strong ones in an iterative fashion. Gradient boosting is a competitive, robust, and interpretable boosting algorithm developed by Friedman (2001, 2002). It is typically used with the classification and regression tree (CART) (Breiman et al. 1984) of a fixed size as base learners, which is known as GBDT. GBDT has been one of the most commonly used techniques in Kaggle competitions and achieved excellent performances in many scientific applications, such as the orbit predictions of space debris (Li et al. 2020, 2021), real-time GNSS precipitable water vapour sensing (Zheng et al. 2022), and GPS signal reception classification (Sun et al. 2020). Therefore, we hypothesise that the GBDT could perform well in modelling and prediction of GNSS time series.

A boosting tree model can be represented as the additive model of decision trees

$$f_M(x) = \sum_{m=1}^{M} T(x; \Theta_m) \tag{4}$$

where $T(x; \Theta_m)$ is the decision tree with its parameters Θ_m and M indicates the number of trees.

The model can be constructed by using the boosting algorithm to fit the training data in a forward stage-wise manner:

$$f_0(x) = 0$$
$$f_m(x) = f_{m-1}(x) + T(x; \Theta_m), \quad m = 1, 2, \ldots, M$$

$$f_M(x) = \sum_{m=1}^{M} T(x; \Theta_m) \tag{5}$$

In the mth step of the algorithm, with the known current model $f_{m-1}(x)$, parameters of mth decision tree $\hat{\Theta}_m$ can be determined by minimising

$$\hat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} L(y_i, f_{m-1}(x_i) + T(x_i; \Theta_m)) \tag{6}$$

where $L(y, f(x))$ is the loss function which is usually the mean squared error (MSE):

$$\begin{aligned} L[y, f_m(x)] &= \frac{1}{2}[y - f_m(x)]^2 \\ &= \frac{1}{2}[r - T(x; \Theta_m)]^2 \end{aligned} \tag{7}$$

where $r = y - f_{m-1}(x)$ is the fitting residual of step $m-1$.

The GBDT algorithm solves Eq. 6 by iteratively creating a base learner $T(x; \Theta_m)$ that points in the negative gradient direction, which can be expressed as

$$\tilde{y} = -\left[\frac{\partial L[y, f(x)]}{\partial f(x)}\right]_{f(x)=f_{m-1}(x)} \tag{8}$$

$\tilde{y}$ is also called as the pseudo-residual because it is an approximate value of residual.

In GBDT method, the current base learner $T(x; \Theta_m)$ is applied to fit the pseudo-residuals of previous base learner $T(x; \Theta_{m-1})$, thus tackling the drawbacks of a weak learner. Finally, the desirable GBDT model can be built in the forward stage-wise manner.

2.2 Long short-term memory

The LSTM is an artificial recurrent neural network (RNN) architecture. It has been widely applied in time series anomaly detection and prediction (Malhotra et al. 2015), machine translation (Wu et al. 2016), speech recognition and synthesis (Fan et al. 2014), handwriting recognition (Carbune et al. 2020), for example, as well as many other applications. For a classic RNN with one hidden layer fed with the input dataset $\boldsymbol{X} = (x_1, x_2, \ldots, x_N)$, the hidden vector sequences $H = (h_1, h_2, \ldots, h_N)$, and the output sequence $\hat{Y} = (\hat{y_1}, \hat{y_2}, \ldots, \hat{y_N})$ are computed through iterating the following equations from $n = 1$ to N:

$$h_n = \mathcal{A}(W_{xh}x_n + W_{hh}h_{n-1} + b_h) \tag{9}$$
$$\hat{y}_n = \hat{f}_{\mathrm{LSTM}}(x_n) = W_{hy}h_n + b_y \tag{10}$$

where $\mathcal{A}$ is the hidden layer function, which is usually a sigmoid function in conventional RNNs; W_{xh}, W_{hh} and W_{hy}

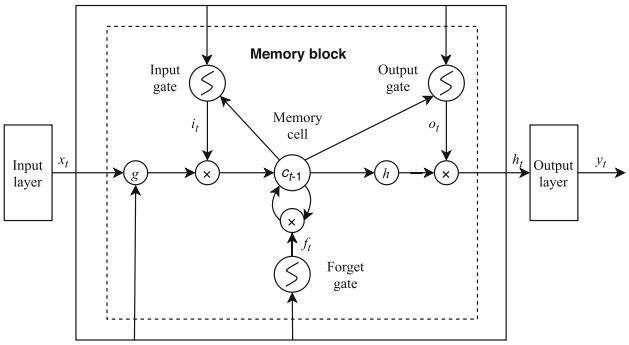

Fig. 1 LSTM neural network architecture (Ma et al. 2015)

indicate weight matrixes between input and hidden vectors, between different time steps of hidden vectors, and between hidden and output vectors, respectively; b_h and b_y are corresponding bias vectors of W_{hh} and W_{hy}, respectively.

Theoretically, a large enough RNN should be sufficient to generate arbitrarily complex sequences (Vincent et al. 2010). However, the application of RNN is limited by vanishing gradient or exploding gradient problems (Bengio et al. 1994). LSTM NN was therefore proposed by (Hochreiter and Schmidhuber 1997) to overcome these problems. Unlike the traditional ANN which typically consists of an input layer, one or more hidden layer(s), and an output layer, the basic unit of the LSTM NN hidden layer is the memory block (shown in Fig. 1) composed of a memory cell and three gates—input gate, forget gate, and output gate (Gers and Schmidhuber 2000). The cell remembers values in any time interval, and three gates regulate the flow of information in and out of the cell. In this way, useful information can be retained, and useless information can be eliminated. The hidden vector h_t of LSTM is different from the conventional RNN, which can be obtained as follows:

$$i_n = \sigma\left(W_{xi}x_n + W_{hi}h_{n-1} + W_{ci}c_{n-1} + b_i\right) \tag{11}$$

$$f_n = \sigma\left(W_{xf}x_n + W_{hf}h_{n-1} + W_{cf}c_{n-1} + b_f\right) \tag{12}$$

$$c_n = f_n c_{n-1} + i_n \tanh\left(W_{xc}x_n + W_{hc}h_{n-1} + b_c\right) \tag{13}$$

$$o_n = \sigma\left(W_{xo}x_n + W_{ho}h_{n-1} + W_{co}c_n + b_o\right) \tag{14}$$

$$h_n = o_n \tanh\left(c_n\right) \tag{15}$$

where σ is the logistic sigmoid function which is defined as $\sigma(x) = \frac{1}{1+e^{-x}}$; i_n, f_n, c_n, and o_n are the input gate, forget gate, cell activation vectors, and output gate, respectively; b_i , b_f , b_o , and b_c are their corresponding bias values; W denotes weight matrices and subscripts of it have the obvious meaning. For instance, W_{hi} indicates the hidden-input gate matrix, and W_xo indicates the input–output gate matrix, etc.

The output $\hat{y_n}$ then can be calculated from Eq. 10. And the $\hat{Y} = (\hat{y_1}, \hat{y_2}, \ldots, \hat{y_N})$ can be obtained by iterating Eqs. 10–15 from $n = 1$ to N.

2.3 Support vector machine

The SVM is another widely applied ML approach proposed by Cortes and Vapnik (1995). The principle of the SVM is to transform input data into a high-dimensional feature space through nonlinear transformation realised by the kernel function and then perform linear regression within the feature space (Smola and Schölkopf 2004). SVM regression is considered a nonparametric technique as it relies on kernel functions. SVM algorithms can be classified by usage of kernels, absence of local minima, or on number of support vectors (Ribeiro 2005). In this study, the linear epsilon-insensitive SVM (ε-SVM) regression is implemented. The goal of ε-SVM is to find a function $\hat{f}_{SVM}(x_n)$ which deviates from y_n by no more than ε, and at the same time it is as smooth as possible. The regression function can be represented as:

$$\hat{f}_{\mathrm{SVM}}(\boldsymbol{X}) = \omega \cdot \varphi(\boldsymbol{X}) + s \tag{16}$$

where ω is the weight vector; $\varphi(\boldsymbol{X})$ indicates a nonlinear mapping function that maps the input space $\boldsymbol{X}$ to high dimensional feature space; s is a scalar threshold. The SVM model performs linear regression in the high-dimensional feature space by ε-insensitive loss. The coefficients ω and s then can be estimated through minimising the regularised risk function:

$$\begin{aligned} &J(\omega) = \frac{\|\omega\|^2}{2} \\ &s.t.\ \forall n: \begin{cases} y_n - \varphi\left(\omega, x_n\right) - s \le \varepsilon \\ \varphi\left(\omega, x_n\right) + s - y_n \le \varepsilon \end{cases} \end{aligned} \tag{17}$$

It is likely that the function $f(\boldsymbol{X})$ that perfectly satisfies Eq. 17 for all points does not exist. Therefore, slack variables ζ_n and ζ_n^* are introduced for each point to deal with other infeasible constraints. These slack variables allow regression errors to exist up to the value of ζ_n and ζ_n^*, but still meet the required conditions. This objective function is described as (Vapnik 2013):

$$\begin{aligned} &J(\omega) = \frac{\|\omega\|^2}{2} + C\sum_{i=1}^{n}\left(\zeta_n + \zeta_n^*\right) \\ &s.t.\ \forall n: \begin{cases} y_n - \varphi\left(\omega, x_n\right) - s \le \varepsilon + \zeta_n \\ \varphi\left(\omega, x_n\right) + s - y_n \le \varepsilon + \zeta_n^* \\ \xi_n \ge 0 \\ \xi_n^* \ge 0 \end{cases} \end{aligned} \tag{18}$$

where the positive constant C is the box constraint, which indicates the penalty degree of the sample with an error that exceeds ε.

The optimisation method is used to maximise the function to solve the dual problem. The dual formula is constructed by

introducing non-negative Lagrangian multipliers α_n and α_n^* into the primal function for each observation x_n. We minimise the dual formula:

$$L(\alpha) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\alpha_i - \alpha_i^*\right)\left(\alpha_j - \alpha_j^*\right)G\left(x_i, x_j\right) + \varepsilon\sum_{i=1}^{N}\left(\alpha_i + \alpha_i^*\right) + \sum_{i=1}^{N} y_i\left(\alpha_i^* - \alpha_i\right) \quad (19)$$

$$s.t.\,\forall n: \begin{cases} \sum_{n=1}^{N}\left(\alpha_n - \alpha_n^*\right) = 0 \\ 0 \le \alpha_n \le C \\ 0 \le \alpha_n^* \le C \end{cases}$$

where the Gaussian kernel function

$$G\left(x_i, x_j\right) = \exp\left(-\left\|x_i - x_j\right\|^2\right) \quad (20)$$

is used as the kernel function of SVM. The SVM model obtained by minimising Eq. 19 is then given as:

$$\hat{f}(x) = \sum_{i=1}^{n}\left(a_i - a_i^*\right)G\left(x_n, x\right) + s \quad (21)$$

Then sequential minimal optimisation (SMO) (Platt 1998) approach is introduced to determine the appropriate parameters (e.g. C and ε) of SVM and the SVM model can be finally determined.

3 Defining the ML problem for GNSS time series modelling

In the workflow of machine learning, the input and output datasets must be clearly defined first: that is, what data are input for training and what data are to be predicted. For GNSS time series modelling and prediction, the output variable $\mathbf{Y}$ is the time series for one of the three coordinate components, that is, the height or vertical component in this context. The choice of the input variables is not straightforward and unique, because there are many potential features associated with the GNSS site motions. Theoretically, the trained model would be more accurate but more complex if more input features were included in the $\boldsymbol{X}$. However, if there is too much irrelevant information, overfitting may occur. In this sense, it is necessary to carefully consider which factors have impacts on the GNSS site motion. In the following, we will discuss how the input and output variables are collected.

3.1 Output data—GNSS height time series

The experimental study for this paper was implemented by using daily solutions of site coordinates provided by the Jet Propulsion Laboratory (JPL) (Heflin et al. 2020). The JPL uses GipsyX software (Bertiger et al. 2020) to process GNSS observations collected from over 2000 receivers in precise point positioning (PPP) mode (Zumberge et al. 1997) with JPL 3.0 final orbits (Dietrich et al. 2018), applying IERS (International Earth Rotation and Reference Systems Service) 2010 solid earth tides (Petit and Luzum 2010; Eanes 1983; Mathews et al. 2002), GPT2w (global pressure and temperature 2 wet) troposphere mapping functions and nominals (Böhm et al. 2015), GYM95 (GPS yaw attitude model-95) satellite yaw model (Bar-Sever 1996), and IGS (International GNSS Service) antenna phase centre maps (Rothacher and Mader 2002). The site coordinate time series provided by the JPL consist of coordinate displacements where the reference coordinates (a set of constant values) are subtracted. According to Heflin et al. (2020), coarse outliers are detected and removed from the time series, breaks are exhaustively detected using the "F test" method, and all parameters, including breaks and seasonal terms, are finally determined through LS fitting. New solutions are added every week and they are freely and openly available at https://sideshow.jpl.nasa.gov/post/series.html.

Nine GNSS sites, including six sites located in Australia namely MOBS, SYDN, PERT, DARW, KARR, and HOB2, two sites located in USA, namely P053 and CEDA, and one site located in Brazil namely POVE, were chosen as our study objects because of the long time span and low missing rate of the datasets collected by these sites. As shown in Fig. 2, the latitude span of these nine sites is from 48.7°N to 37.8°S. Among them, six Australian sites are close to the ocean and the other three sites are located inland. The vertical coordinates time series of these nine stations with a duration of 9 years (from 1 January 2008 to 31 December 2016) were used as output data of modelling datasets. In addition, the time series from 1 January 2017 to 1 January 2021, totalling 4 years of data, were used as test datasets to test prediction performance of derived models. The samples in some dates are missing due to instrument failure or other reasons. For example, the PERT station has the missing of 294 days of data in total, from 3 January 2012 to 24 October 2012 (see Fig. 6c). However, our adopted ML models are able to train these discontinuous datasets because the time series is not modelled with time variable only, but physical factors. This is also an advantage of the ML models compared with the traditional LS methods.

3.2 Input data—impact factors

Based on the IERS Conventions (2010) (Petit and Luzum 2010), the GNSS site motions are divided into three categories: (1) conventional displacements mainly related to tidal motions (mostly with near diurnal and semidiurnal frequencies) and other displacements with longer periods and

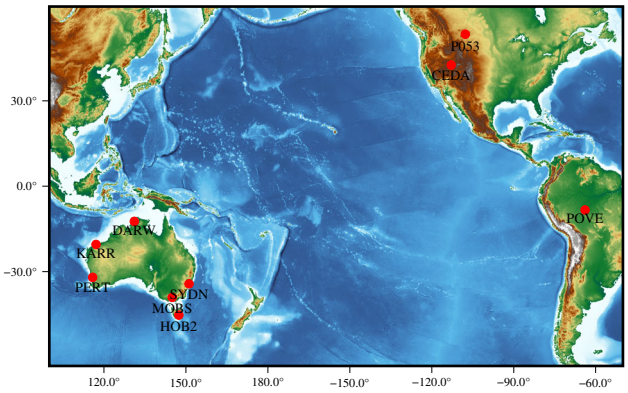

Fig. 2 Distribution of the applied GNSS sites

can be accurately modelled; (2) non-tidal motions associated with changing environmental loads; (3) displacements that affect the internal reference points within the observing instruments.

The GNSS site displacements are conventionally caused by the solid earth tides (SET), the ocean tidal loading (OTL), the atmospheric pressure loading (APL) arising from the attraction of the Sun and the Moon, and the pole tide and the ocean pole tide loading caused by changes in the centrifugal potential due to polar motion (Petit and Luzum 2010). These displacements can be modelled and corrected from the GNSS time series by means of corresponding models. However, displacements caused by some tidal loadings may not be considered and corrected in some GNSS observation processing cases. Furthermore, even with all corrections applied, these displacements still cannot be fully corrected because of models and estimation uncertainty. For example, after the correction of the SET-caused displacement, there still be residuals existing in the GNSS time series which can be up to 2.0 mm in vertical direction (Watson et al. 2006). For the GNSS time series used in this research, as described in Heflin et al. (2020), only the SET displacement is corrected by using the IERS 2010 SET model (Eanes 1983; Mathews et al. 2002). As a result, the SET displacement residuals, along with other displacements caused by the OTL, APL, pole tide and ocean pole tide, are still remained in these vertical coordinate time series. The six sites located in Australia are specifically affected by the OTL as they are close to the ocean. Studies show that time-varying deformations of the Earth produced by the OTL can reach 100 mm at some special coast regions (Li et al. 2014; Melachroinos et al. 2008). Atmospheric tides also have an impact on the site motions, especially on the vertical component. The amplitude of the vertical deformation caused by APL is equal to that of some OTL effects and should be considered when modelling the site motions (Petit and Luzum 2010). The root mean square (RMS) of average variations for the vertical component is 2.6 mm, but peak to peak variations can reach 40 mm (Capaldo et al. 2014). Dong et al. (2002) indicates that the impact of APL on oceanic islands and near coasts are smaller than within the continents. Their study shows that sites on the eastern Antarctica coast have strong semi-annual variations from APL with amplitudes of more than 1 mm. In addition, the pole tide could cause the variation of station coordinates of a couple of centimetres. And the deformation caused by the ocean pole tide, which is generated by the centrifugal effect of polar motion on the oceans, is typically no greater than 1.8, 0.5, 0.5 mm on radial, north, and east directions, respectively, but it can be larger occasionally (Petit and Luzum 2010).

Non-tidal motions are associated with changing environmental loads, e.g. from hydrology, atmosphere, and the ocean (Singh et al. 2021). Hydrology can cause GNSS site motions by means of water mass change. In continental interiors, water is stored in lakes, rivers, groundwater reservoirs, as well as in soil, vegetation and snowpack (Puskas et al. 2017). The mass of water varies seasonally (Fovell and Fovell 1993) and geographically (Miller 1994) and can cause the displacements of several millimetres in GNSS time series (Herring et al. 2016). The non-tidal atmospheric loading is the non-tidal part of the APL except for the conventional diurnal S_1 and semidiurnal S_2 components. The IERS 2010 conventions do not recommend involving non-tidal loading (NTL) deformations into the calculation of the reference point displacement because their modelling accuracy is rather low (Petit and Luzum 2010). However, in this study, these NTLs are expected to be figured into the GNSS site motions by introducing NTL-related variables, including hydrology, and surface atmospheric pressure data, into ML models.

In addition to the tidal and non-tidal loadings, the GNSS site position is also affected by bedrock thermal expansion caused by surface temperature changes. Surface temperature changes cause internal temperature change in the surface cement piers where GNSS antennas are installed on GNSS sites, and contribute to temperature change in the bedrock through heat conduction, thereby causing change in the vertical displacement of GNSS stations. According to the research of Yan et al. (2010), the impact of temperature change on GNSS reference stations can reach 2.8 mm. Therefore, temperature change is a non-negligible factor that causes annual changes in the vertical displacement of GNSS sites.

The temperature not only directly causes the site deformation through the bedrock thermal expansion mechanism, but also indirectly affects site position by generating changes in the hydrology and atmospheric pressure. For example, comparing the temperature time series and hydrology time series which are shown in Fig. 4, it is found that within a 1-year time span, the hydrological variable consistently records at the minimum value when the temperature is the highest of the year.

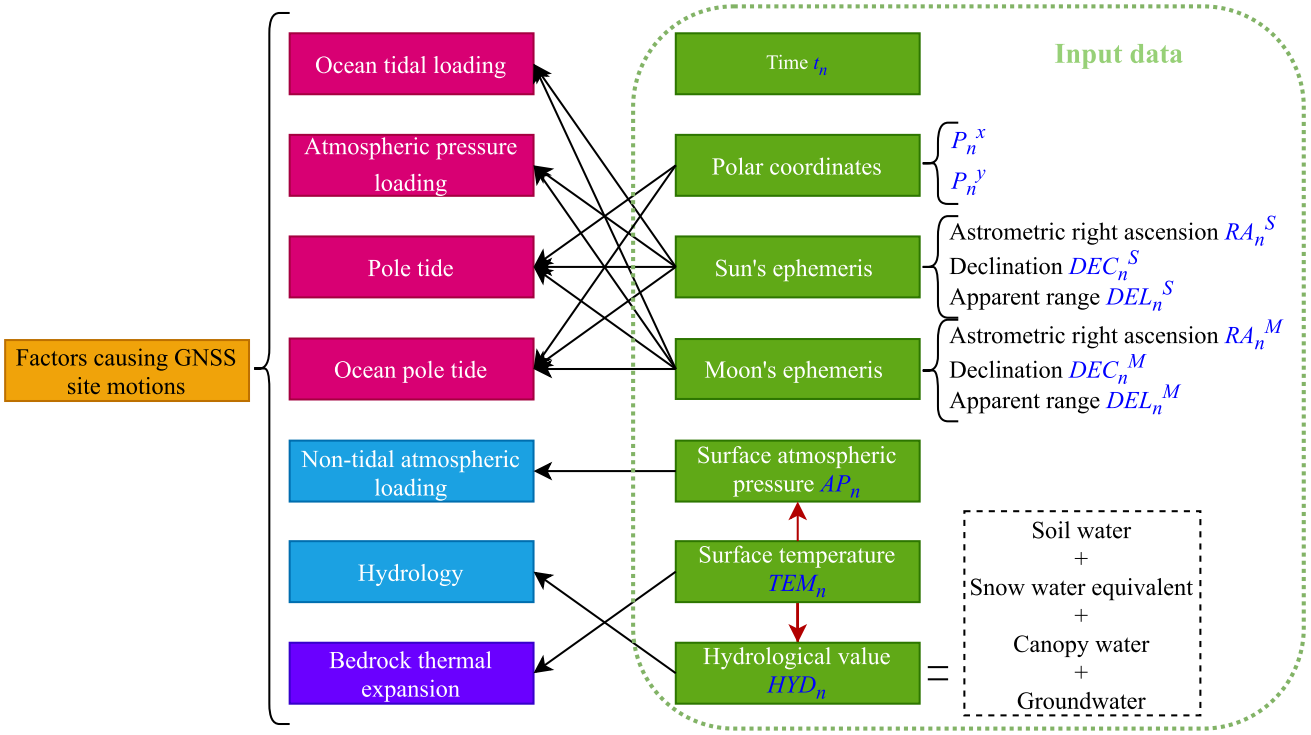

Fig. 3 Relationships between impact factors and input data sets

Based on the above knowledge of the factors causing the GNSS site motions, it is necessary to represent these factors using several datasets which can be used as input data in the ML training. As noted in Sect. 2.1 the GNSS site motions can be represented by 12 variables in a ML model. Figure 3 illustrates the relationships between these variables and physical factors. The backward arrows indicate that each variable can represent the influence of one or more physical factors. However, it must be noted that the input variable time t is the variable of all the factors. The site motions caused by the OTL and the APL are represented by the ephemerides of the Sun and Moon, which are calculated by the JPL's HORIZONS system, including their right ascension, declination, and distance with respect the studied GNSS site in ICRF. The coordinates of the celestial ephemeris pole relative to the to the IRP, the IERS Reference Pole, are provided by the IERS and used to represent the site motions caused by pole tide and ocean pole tide. The surface temperature and atmospheric pressure data are obtained from the National Centres for Environmental Prediction (NCEP) reanalysis products. The spatial resolution of these products is $2.5° \times 2.5°$, and the temporal resolution is 1 day. The hydrological variable is represented by the terrestrial water storage consisting of soil water, snow water equivalent, canopy water, and groundwater, obtained from the Global Land Data Assimilation System (GLDAS) products provided by the National Aeronautics and Space Administration (NASA). These products have a spatial resolution of $0.25° \times 0.25°$, and a temporal resolution of 1 day.

All these impact variables have the same temporal resolution with the GNSS time series which is 1 day. Variables for the MOBS site are illustrated in Fig. 4. It can be seen from the figure that while the three variables related to the lunar ephemeris have a 1-month cycle, other variables have seasonal variations. It also can be observed that there are missing values in variable time series. These values are removed from the datasets and their epochs are corresponding to that of GNSS time series' missing values. This enables every input vector consisting of 12 input data on each epoch has a corresponding output value.

4 Experimental results and numerical analysis

The workflow of the data analysis is shown in Fig. 5. The modelling datasets for each site are normalised and denoted as D_m^s where s indicates the site name, for example, if $s =$ MOBS, the D_m^s is the modelling dataset for MOBS site. 80% timespan of D_m^s is used as training data and 20% is used as validation data. Then D_m^s are trained through GBDT, LSTM NN, and SVM, respectively. The modelling and prediction performances of GBDT-, LSTM-, and SVM-models are evaluated on modelling datasets and test datasets, respectively.

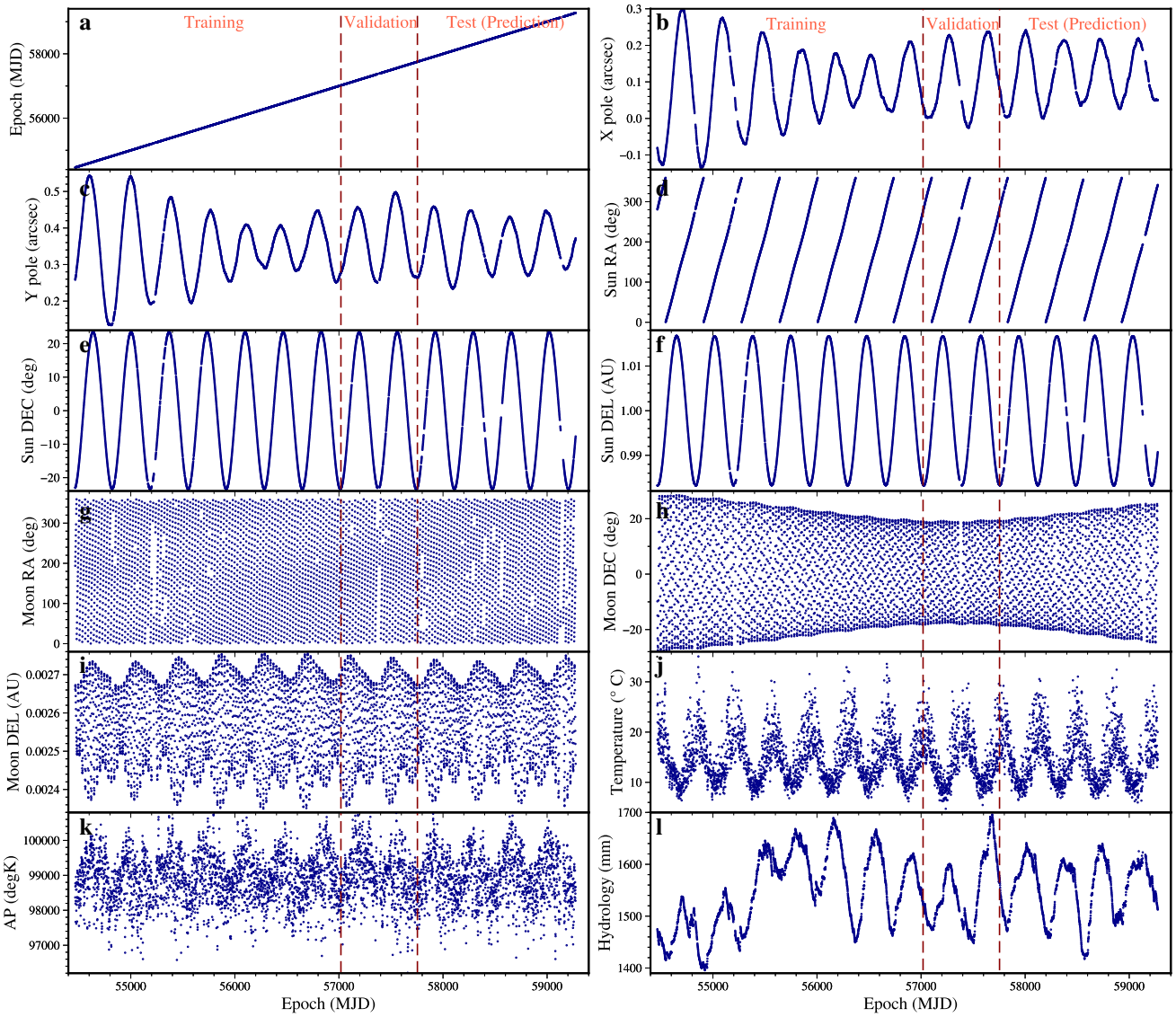

Fig. 4 Impact factor time series (input datasets) for MOBS site

4.1 Training models

The GBDT algorithm is implemented through the scikit-learn which is a machine learning library (Pedregosa et al. 2011). Parameters are tuned by a grid search processing, and the result shows that the GBDT model achieves its best performance when employing 100 base learners, with their maximum depth of 3 and learning rate of 0.1. The LSTM NN is constructed by the Keras package (Chollet 2015) using TensorFlow (Abadi et al. 2016) as the backend. This LSTM NN has three layers: the first two layers are LSTM layers with 64 and 32 neurons, respectively, and the last one is a dense layer. LSTM-models for each site are obtained through training the D_m^s on the LSTM NN, with time step set to 1 and the number of training epochs set to 150, as the performance of the model no longer improves on the validation dataset after 150 epochs of training. The modelling dataset D_m^s is also trained through the SVM algorithm implemented by the Machine Learning Toolbox provided by the MATLAB. Compared with the LSTM algorithm, SVM has less hyperparameters to be tuned. As mentioned in Sect. 2.3, the Gaussian kernel function is used as the kernel function, and the SMO is chosen as the sequential minimal optimisation of the applied SVM.

ML models are trained on a computer with an Intel Core I7 processor with 2.60 GHz processing speed. On this platform, the GBDT, LSTM, and SVM algorithms spend 0.7 s, 29.3 s, and 0.5 s on average to train a model, respectively. Obviously, the GBDT and SVM are time-efficient algorithms in the GNSS time series modelling case. In contrast, the LSTM algorithm, as an ANN approach, is time-consuming.

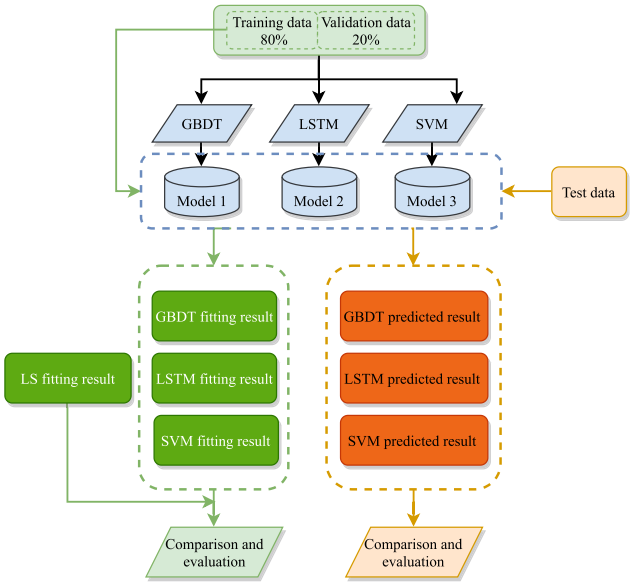

Fig. 5 Workflow of study case

As mentioned in Sect. 2, these models could describe the underlying relationship $\hat{Y} = \hat{f}(\boldsymbol{X})$ between input data $\boldsymbol{X}$ (physical variables) and output data Y (GNSS site vertical displacement or motion). For a given input vector x, the estimated output value $\hat{y}$ can be obtained through the model: $\hat{y} = \hat{f}(x)$.

The GBDT-fitting results $\hat{Y}^s_{GBDT,m}$ and GBDT-predicted results $\hat{Y}^s_{GBDT,t}$ are obtained through $\hat{Y}^s_{GBDT,m} = \hat{f}^s_{GBDT}(\boldsymbol{X}^s_m)$ and $\hat{Y}^s_{GBDT,t} = \hat{f}^s_{GBDT}(\boldsymbol{X}^s_t)$, respectively, where $\boldsymbol{X}_m$ and $\boldsymbol{X}_t$ indicate input data from the modelling dataset and test dataset, respectively. The fitting and prediction results of LSTM and SVM are obtained in the same way.

The fitting performance of GBDT, LSTM, and SVM models is compared with a LS model applied by Heflin et al. (2020). The LS-fitting results are denoted as $\hat{Y}^s_{LS,m}$.

4.2 Fitting performance evaluation

To verify the performance of the GBDT, LSTM, SVM, and LS models, the root mean square error (RMSE) is adopted as the evaluation indicator and can be calculated through

$$\mathrm{RMSE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(y_i - \hat{y}_i\right)^2} \tag{22}$$

where y_i is the measured vertical coordinate on epoch i; $\hat{y}_i$ is the corresponding model-derived value; M is the number of epochs.

The fitting and predicting results derived from three ML methods are illustrated in Fig. 6, which shows all having good consistence with the measured values. The LS curves are also plotted over the fitting period, but prediction curves are unavailable from the datasets collected. The statistics of fitting precision (FPs) for GBDT, LSTM, SVM, and LS methods (Fig. 7) shows that all the three ML methods achieve consistently better fitting performance than the LS method for each studied site. Among them, the GBDT achieved the slightly better fitting precision. The fitting RMSE of GBDT models for nine studied sites range from 2.7 mm (MOBS and P053 sites) to 4.3 mm (DARW site), with an average of 3.4 mm, which is an improvement by 36%, compared with the 5.3 mm achieved by the LS method. The fitting results from the LSTM and SVM methods also show similar improvements. The average FPs of LSTM and SVM are 3.5 and 3.7 mm, improved by 34% and 30% as compared to that of LS, respectively.

In addition to higher FPs achieved by ML models relative to the LS models, the ML models have two advantages when modelling time series: (1) there are no needs to detect breaks or discontinuity, and (2) there are no needs to bridge missing data. As a result, the discontinuous datasets can all be used in ML processing. When applying the LS models, it is necessary to detect the breaks in the time series as these breaks will significantly affect the LS fitting performance. For instance, according to our experiments with these nine sites, the LS FP can be decreased by 14% if without break detection. However, if the ML models only use previous positions as input data as presented in (Alevizakou et al. 2018; Wang et al. 2021), the continuity of the time series is still required. In contrast, the adopted ML models can still work well when there are missing values in modelling dataset, as these models use physical variables as inputs. Actually, when training GBDT models using the scikit-learn library, modelling samples are randomly split into training and validation samples (the illustration of data division shown in Fig. 4 is inaccurate in this case). Therefore, missing some samples barely have impact on training models. For example, though PERT and KARR sites lost 315 and 172 days of data in the training data respectively, the FP of the GBDT models for these two sites can still reach 3.1 and 3.2 mm. These results indicate no signs that the discontinuity of modelling dataset has impacted the adopted ML models. It is to be noted that, the disadvantages of the proposed ML approaches are also obvious: (1) it takes more time on training model, and (2) data collection of various physical variables is required. The physical variables were collected from different institutions and stored in different formats with different spatial or temporal resolutions. Therefore, the proposed ML models require a lot of preparatory work compared traditional methods such as LS method.

4.3 Prediction performance evaluation

A model's ability to generalise is critical to its success. Prediction performance refers to the model's generalisation

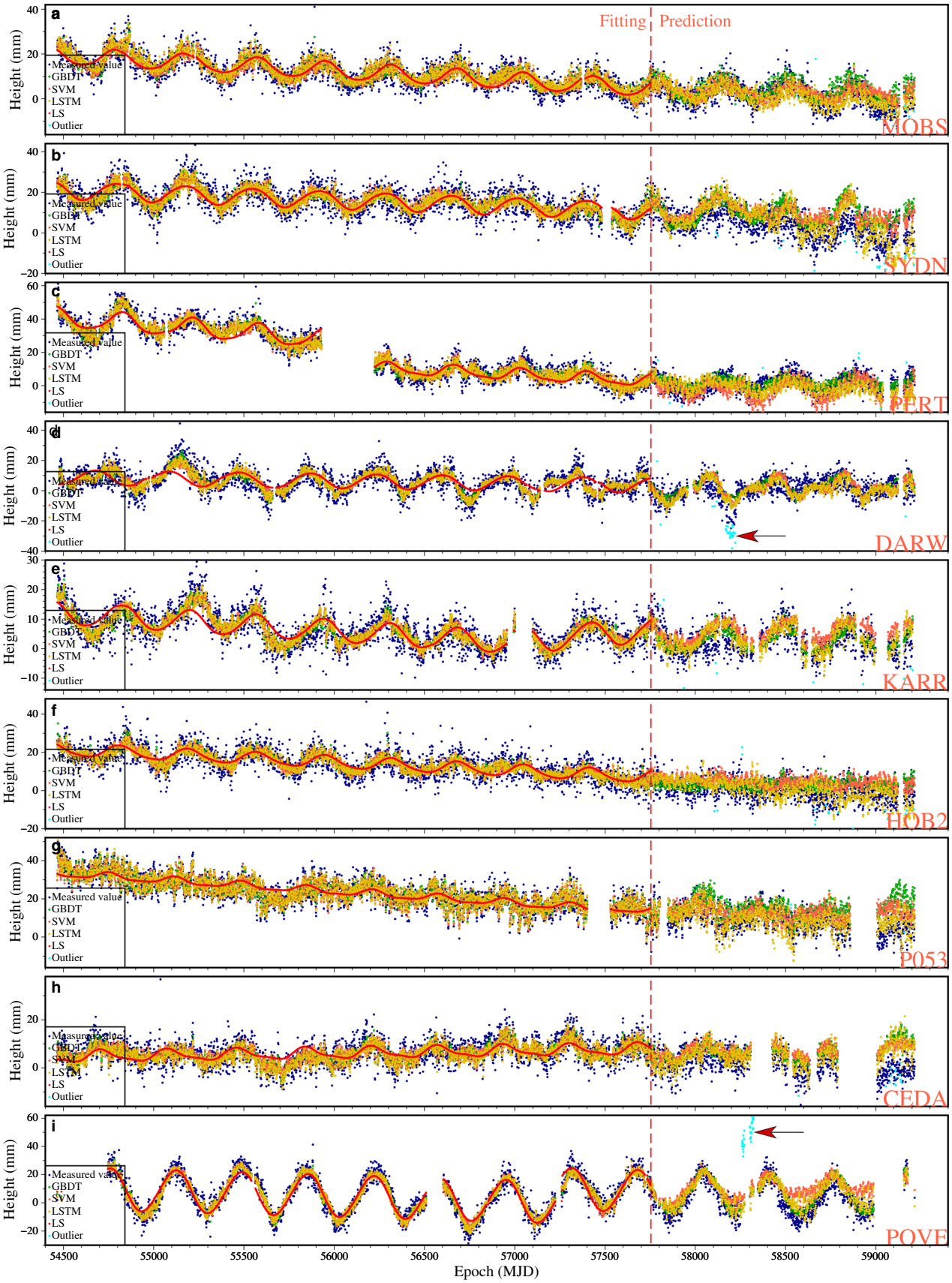

Fig. 6 Fitting and prediction results derived from GBDT, LSTM, SVM, and LS models from 9 GNSS sites

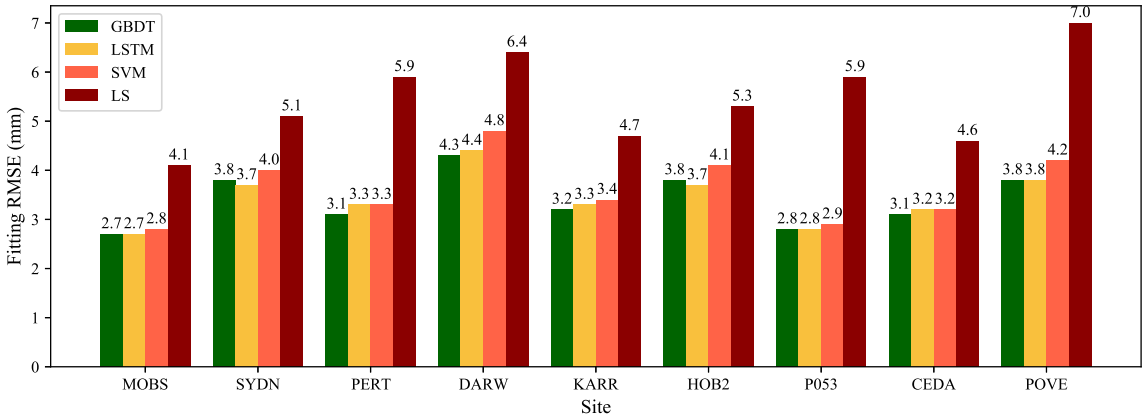

Fig. 7 Comparison of fitting precisions of GBDT, LSTM, SVM, and LS models

ability to adapt properly to unseen data in the time series. The right-hand side part of Fig. 6 illustrates the prediction results of three ML models for the future 48 months against the measured data. Despite not being plotted, our computation shows that the prediction errors of LS models can grow to several centimetres within 24 months, indicating a significant limitation of the LS models for predicting the GNSS site motions. However, the good consistence between the predicted and measured data samples illustrates that ML models have the ability to predict vertical displacements with established physical variables.

The prediction results of ML models as shown in Fig. 6 also clearly indicate grouped sample outliers of 20–60 mm found in DARW and POVE sites. These outliers could be caused by data processing, antenna height changes, or site motion due to physical reasons such as earthquakes. The real causes should be directed to further investigation or reference to results from different GNSS processing systems. For instance, the DARW time series results from Geoscience Australia database (https://gnss.ga.gov.au/network) did not present similar grouped outliers over the period of 31 January 2018 to 16 April 2018.

To evaluate generalisation capability of these ML models in prediction, the test data period of 48 months was equally divided into eight 6-month data parts in the chronological order. The prediction errors of ML models over each data part are grouped together to obtain RMSE values as prediction precision (PP). The grouped sample outliers in DARW and POVE sites are removed from the RMSE statistics. As shown in Fig. 8, the RMSE values for GBDT, LSTM and SVM methods show the growth of the prediction errors from 4 to 6 mm within the first 12 months, to 5–8 mm in the second 12 months, and to 6–15 mm in the last 24 months. It is observed that three ML models have roughly similar prediction performance over the first 24 months. Over the next 24 months, however, prediction precision varies with ML models and sites. For instance, the LSTM models show slightly better prediction capability on most of the tested sites, while the GBDT models show superior prediction performance on PERT and POVE sites. Unlike traditional LS models based only on the time variable for the input data (Bock and Melgar 2016; Heflin et al. 2020), the prediction performance of the same ML models could vary because of different impacts of the physical factors depending on site locations and local environment.

While generalisation capability is the essential assessment of ML models, the prediction performance of the GNSS coordinate time series is critical for deformation analysis and maintenance of a regional or global geodetic reference frame (Altamimi et al. 2011, 2016). As shown above, the comparison between the predicted and measured values can detect and identify slow or sudden changes in the new samples, which could be the displacements caused by site deformation, or outliers caused by data treatments and antenna problems. A geodetic reference frame is realised by a large member of regionally or globally distributed geodetic reference stations with respect to a particular time epoch. For instance, the ITRF 2020 is referred to the epoch 2015.0 (Altamimi et al. 2018) and provides with the station position kinematic model to compute the station position on other dates. The kinematic model, perhaps being updated regularly with new time series data, includes terms for linear displacements by velocity, post-seismic deformation and the annul and semi-annul signals (Altamimi et al. 2016). The fitted and predicted coordinate time series from ML models can alternatively be used to compute the current station coordinates of the reference stations with high precision.

4.4 Feature importance analysis

Some ML algorithms can derive the parameters of the feature importance (FI) for each physical variable, which indicates the degree that the variable affects the GNSS site displacements. For tree-based ensemble methods, the importance of

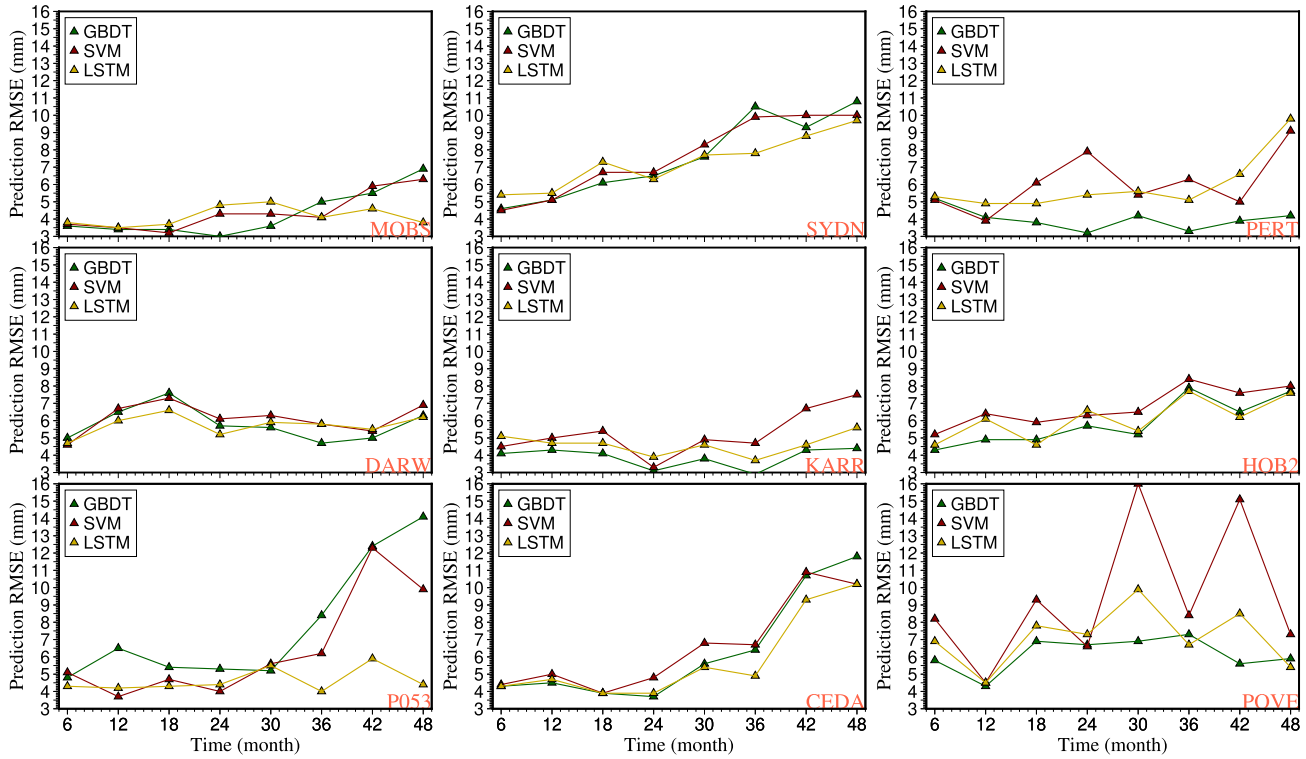

Fig. 8 Illustration of prediction precision of GBDT, LSTM and SVM models versus predicted periods

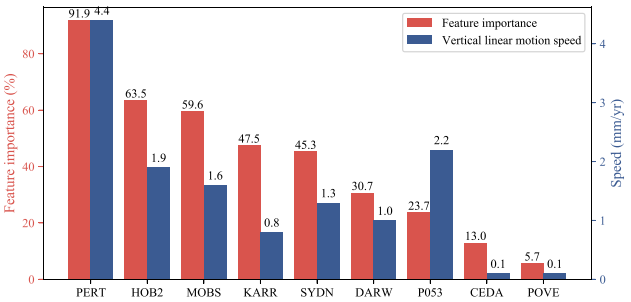

Fig. 9 Relationship between feature importance of time and vertical motion speed

a feature is computed as the (normalised) total reduction of the criterion brought by that feature. It is also known as the Gini importance or mean decrease in impurity (MDI) (Louppe et al. 2013). As the GBDT approach achieves better modelling and good short-term prediction performance compared with other two ML approaches, FIs derived from GBDT models are used to study the relationship between physical variables and GNSS site motions.

FIs for 12 physical variables (Table 1) reveals that time and hydrological variables are two dominant factors or variables affecting GNSS site displacements. It can be found from Fig. 6 and Table 1 that the FI of the time variable (FIOT) is higher when the site vertical linear motion is faster (e.g. PERT site). The vertical linear motion speeds of nine sites are calculated by least squares estimates (Heflin et al. 2020) and are shown in Fig. 9 along with FIOT. Figure 9 roughly illustrates the positive correlation between FIOT and site motion speed. For instance, FIOT and vertical motion speeds reached the highest values at PERT site and lowest values at CEDA and POVE sites. However, the site motion speed could be more dominantly caused by other factors, such as the hydrological factor as shown in P053.

The feature importance parameters indicate that the hydrologic variable could be another dominant factor. Hydrologic variable shows big feature importance on DARW, P053, CEDA, and POVE sites, especially on the POVE site where the feature importance of hydrologic variable (FIOH) is over 70%. The high FIOHs of these four sites correspond with relevant features of their locations and terrains. The satellite images around the four sites extracted from Google Maps as shown in Fig. 10 shows that they all are closed to water bodies (reservoirs, lakes, rivers, etc.). Water mass changes can cause elastic deformation at the Earth's surface (Puskas et al. 2017). In addition, groundwater pumping can drive poroelastic deformation in agricultural regions (Argus et al. 2014). The DARW site is located in the south of Darwin, Australia. There are two reservoirs near this site—the Manton Dam with a volume of 13.3 gigalitres (GL) is less than 1 km from the site, and the Darwin River Dam which can hold up to 259.0 GL of water is in the 10 km range from the site. The CEDA site located in Utah state of the USA is about 20 km from the Great Salt Lake, one of the largest saltwater lakes

Table 1 Feature importance (%) for 12 input variables

Site	t	P^x	P^y	RA^S	DEC^S	DEL^S	RA^M	DEC^M	DEL^M	TEM	AP	HYD
MOBS	59.6	1.1	2.1	3.3	4.7	2.3	0.3	0.3	0.3	24.4	0.5	1.1
SYDN	45.3	2.7	9.5	3.1	18.7	2.7	0.9	0.5	0.8	7.4	4.6	3.8
PERT	91.9	0.4	0.9	1.3	1.1	1.1	0.1	0.1	0.1	1.5	1.3	0.2
DARW	30.7	6.9	2.2	10.3	4.3	3.6	0.9	0.7	0.8	1.5	1.1	37.1
KARR	47.5	2.2	1.2	5.2	3.2	12.3	0.5	0.7	0.7	6.7	10.2	9.6
HOB2	63.5	3.0	1.9	3.6	2.9	12.9	0.5	1.3	1.0	4.9	2.1	2.4
P053	23.7	0.5	0.7	1.7	1.0	1.2	0.3	0.3	0.3	7.6	14.8	47.8
CEDA	13.0	3.2	2.5	2.1	3.7	5.6	1.0	1.3	0.9	15.2	3.6	47.9
POVE	5.7	1.4	2.0	9.2	8.5	1.8	0.2	0.1	0.2	0.1	0.3	70.3

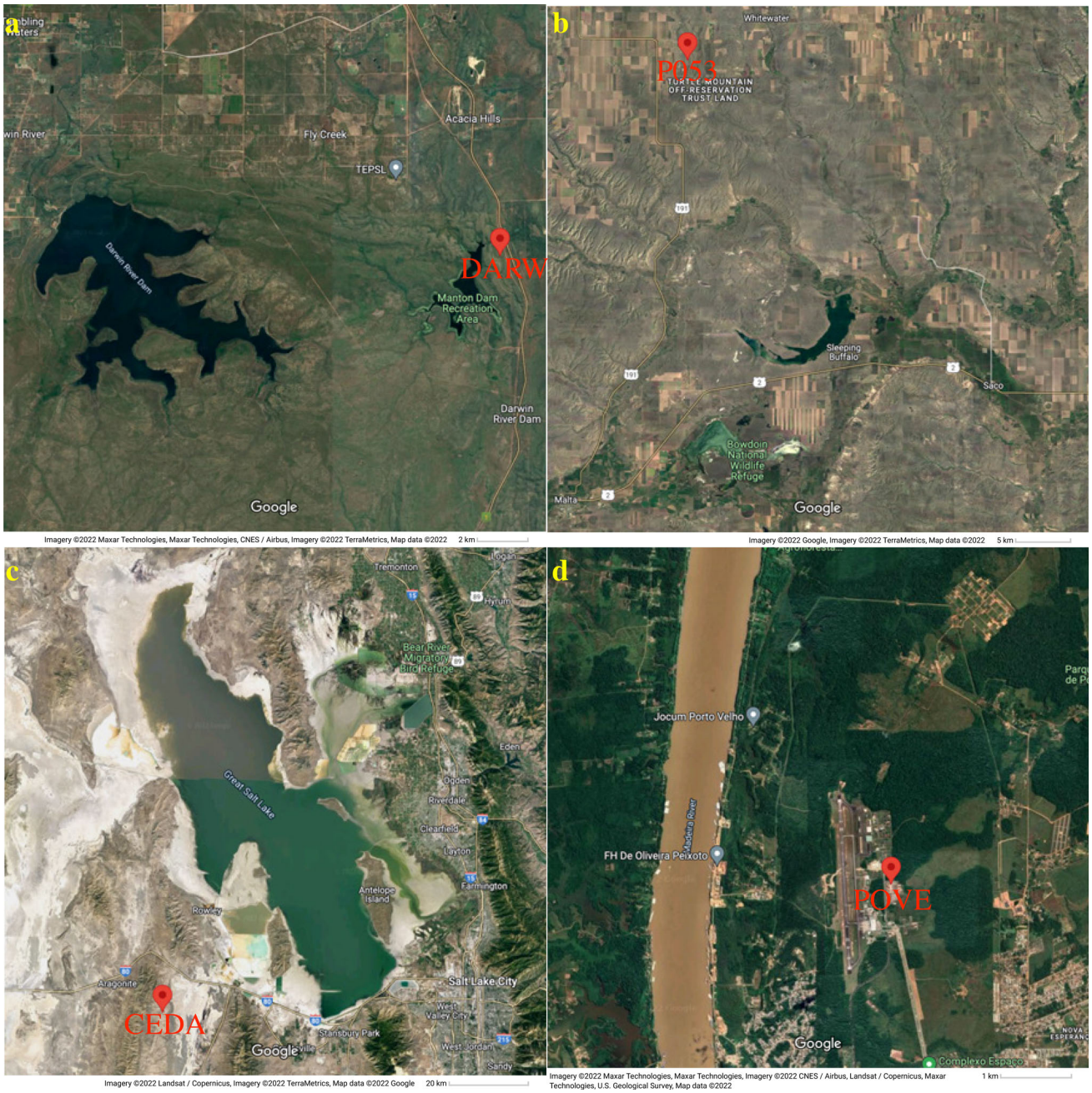

Fig. 10 Satellite images around DARW, P053, CEDA, and POVE sites (these images are thankfully from Google Maps)

in the world (Ghosal et al. 2021). The P053 site is located in Whitewater, Montana, USA. There are two water bodies about 30 km south of this site—the Nelson reservoir and the Lake Bowdoin. In addition, the site is about 100 km away from the Missouri River in Northern America. Site displacements could also be affected by the poroelastic deformation caused by groundwater pumping as this site is located in the agricultural regions of Montana state. A large volume of groundwater extraction for agricultural irrigation could have a significant impact on site motion. The POVE site is located in Porto Velho, Rondônia, Brazil, where lies on the east bank of the Madeira River and is a part of the Amazon Rainforest, featuring the tropical monsoon climate. In addition, this site is also in the agriculture region. For all these reasons, the hydrologic variable achieves the highest feature importance on this site.

In addition, Table 1 indicates that the FI of sun's position (RA^S, DEC^S and DEL^S) (FIOS) and the FI of moon's position (RA^M, DEC^M and DEL^M) (FIOM) are all relatively small and varying from site to site. This result is not necessarily inconsistent with the high influence Sun's and Moon's tide-generating force on vertical displacement (Thurman 1994). We should notice that the SET, which is the most influential tidal loading and can up to 0.4 m (Lambeck 1988), is corrected from GNSS time series by means of the IERS 2010 SET model when processing GNSS observations (Petit and Luzum 2010; Eanes 1983; Mathews et al. 2002). The small FIOS and FIOM results only reflect the residual effects of these variables.

5 Conclusion

Long-term GNSS time series from widely distributed reference stations has played a fundamental role in geophysics and geodynamics studies, deformation monitoring, and maintenance of a regional or global geodetic reference frame. GNSS coordinate time series have been conventionally fitted by the least square (LS) model as a function of time variable over the past tens of years. This paper has proposed to model the underlying pattern between the GNSS vertical time series and physical variables through the GBDT, LSTM, and SVM machine learning approaches. To form the machine learning problem, we turn the major impact factors into the input variables and the vertical time series into the outputs. There are 12 identified variables related to the vertical site motion, including time variable, polar motion coordinates, Sun and Moon coordinates, temperature, atmospheric, and hydrologic parameters. The experimental results derived from nine GNSS sites have demonstrated that all the trained GBDT, LSTM, and SVM models are capable of capturing most features of the underlying site motion pattern. Compared with the LS model proposed by (Heflin et al. 2020), ML models have shown evidently higher fitting precision on studied sites. The average fitting precision of GBDT, LSTM, and SVM models are 3.4, 3.5, and 3.7 mm, improvements by 36%, 34%, and 30%, respectively, with respect to the LS fitting results. In addition, all the ML approaches show consistent performance for modelling GNSS time series.

The GBDT, LSTM and SVM approaches have also demonstrated good generalisation capability that the trained ML models are able to predict the vertical displacements using the established physical variables in the predicted period. The prediction precision of ML models is a little lower than fitting precision, but slowly ranging from about 4 to 15 mm, as the prediction time terms span from 6 to 48 months. High prediction performance can enhance the detection of large displacements samples and prediction of the current station coordinates of the reference stations with high precision.

Overall, the ML models have shown the evident improvement in modelling of GNSS time series with respect to the conventional LS model. In addition, this paper has shown that the ML models can be used to predict the vertical motion to long future periods, thus enhancing and extending the applications of GNSS time series analysis in geodesy and geodynamics. Future research may extend the adoption of different ML models to fit and predict GNSS time series in three coordinate components and other types of geodetic time series. This may enable the inclusion of more and different impact factors or features for individual sites or data types and generate more complete knowledge about using ML approaches for geodetic data analytics and applications.

Acknowledgements The first author acknowledges the scholarship support received from Queensland University of Technology and China Scholarship Council. The third author acknowledges the support by the National Natural Science Foundation of China (Grant No. 42004017). The QUT authors also acknowledge the partial support by the project "Automated Monitoring and Analytics for Geotechnical Structural Performance Using the Internet of GNSS Things" (2019- 2022). The project is co-funded by the Department of Industry, Science, Energy and Resources (Innovative Manufacturing CRC Ltd), Mthing Pty Ltd and Monitum Pty Ltd (IMCRC/MON/24042019).

Author Contributions The first author prepared the data, created the processing codes, performed the analysis and draft the paper. The second, third, and forth authors reviewed the manuscript and provided inputs for analysis of data and results. The fifth author initiated the topic, conceived and designed the analysis, and improved the paper. All authors read and approved the final manuscript.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

Data availability The polar motion dataset is available in the International Earth Rotation and Reference Systems Service at https://www.iers.org/IERS/EN/DataProducts/EarthOrientationData/eop.html. The near surface temperature and surface atmospheric pressure datasets are available in the National Centers for Environmental Prediction at https://psl.noaa.gov/data/gridded/data.ncep.reanalysis.surface.html. The hydrological data is available in the Global Land Data Assimila-

tion System at https://disc.gsfc.nasa.gov/datasets?keywords=GLDAS. The GNSS coordinate time series are provided by the Jet Propulsion Laboratory at https://sideshow.jpl.nasa.gov/post/series.html as cited in Heflin et al. (2020).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M et al (2016) Tensorflow: a system for large-scale machine learning. In: 12th USENIX symposium on operating systems design and implementation (OSDI 16), pp 265–283

Alevizakou EG, Siolas G, Pantazis G (2018) Short-term and long-term forecasting for the 3d point position changing by using artificial neural networks. ISPRS Int J Geo Inf 7(3):86. https://doi.org/10.3390/ijgi7030086

Altamimi Z, Collilieux X, Métivier L (2011) Itrf 2008: an improved solution of the international terrestrial reference frame. J Geod 85(8):457–473

Altamimi Z, Rebischung P, Métivier L, Collilieux X (2016) Itrf 2014: a new release of the international terrestrial reference frame modeling nonlinear station motions. J Geophys Res Solid Earth 121(8):6109–6131. https://doi.org/10.1002/2016jb013098

Altamimi Z, Rebischung P, Collilieux X, Metivier L, Chanard K (2018) Roadmap toward itrf2020. AGU Fall Meeting Abstracts 2018:G42A-08

Altamimi Z, Rebischung P, Metivier L, Collilieux X, Chanard K, Teyssendier-de-la Serve M (2021) Preparatory analysis and development for the itrf2020. In: EGU general assembly conference abstracts, pp EGU21–2056

Argus DF, Fu Y, Landerer FW (2014) Seasonal variation in total water storage in California inferred from GPS observations of vertical land motion. Geophys Res Lett 41(6):1971–1980

Bar-Sever YE (1996) A new model for GPS yaw attitude. J Geod 70(11):714–723

Bengio Y, Simard P, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Netw 5(2):157–166

Bennett RA (2008) Instantaneous deformation from continuous GPS: contributions from quasi-periodic loads. Geophys J Int 174(3):1052–1064

Bertiger W, Bar-Sever Y, Dorsey A, Haines B, Harvey N, Hemberger D, Heflin M, Lu W, Miller M, Moore AW et al (2020) GipsyX/RTGx, a new tool set for space geodetic operations and research. Adv Space Res 66(3):469–489

Bock Y, Melgar D (2016) Physical applications of GPS geodesy: a review. Rep Prog Phys 79(10):106801

Böhm J, Möller G, Schindelegger M, Pain G, Weber R (2015) Development of an improved empirical model for slant delays in the troposphere (GPT2w). GPS Solut 19(3):433–441

Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and regression trees. Routledge, London

Capaldo P, Fratarcangeli F, Nascetti A, Mazzoni A, Porfiri M, Crespi M (2014) Centimeter range measurement using amplitude data of terrasar-x imagery. Int Arch Photogrammetry, Remote Sens Spat Inf Sci XL 7:55–61

Carbune V, Gonnet P, Deselaers T, Rowley HA, Daryin A, Calvo M, Wang LL, Keysers D, Feuz S, Gervais P (2020) Fast multi-language LSTM-based online handwriting recognition. Int J Docum Anal Recogn (IJDAR) 23(2):89–102

Caveney D (2010) Cooperative vehicular safety applications. IEEE Control Syst Mag 30(4):38–53

Chen Q, van Dam T, Sneeuw N, Collilieux X, Weigelt M, Rebischung P (2013) Singular spectrum analysis for modeling seasonal signals from GPS time series. J Geodyn 72:25–35

Chollet F (2015) Keras. https://github.com/fchollet/keras

Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297

Davis JL, Wernicke BP, Bisnath S, Niemi NA, Elósegui P (2006) Subcontinental-scale crustal velocity changes along the Pacific-North America plate boundary. Nature 441(7097):1131–1134

Davis JL, Wernicke BP, Tamisiea ME (2012) On seasonal signals in geodetic time series, J Geophys Res 117:B01403

Dietrich A, Ries P, Sibois AE, Sibthorpe A, Hemberger D, Heflin MB, David MW (2018) Reprocessing of GPS products in the IGS14 frame. AGU Fall Meeting Abstracts 2018:G33C-0690

Dong D, Fang P, Bock Y, Cheng M, Miyazaki S (2002) Anatomy of apparent seasonal variations from GPS-derived site position time series. J Geophys Res Solid Earth 107(B4):ETG-9

Dörterler M, Faruk Bay Ö (2018) Neural network based vehicular location prediction model for cooperative active safety systems. Promet-Traffic Transp 30(2):205–215

Eanes R (1983) Earth and ocean tide effects on Lageos and Starlette. In: Proceedings of the ninth international symposium on Earth tides, E. Sckweizerbart'sche Verlagabuchhandlung

Fan Y, Qian Y, Xie FL, Soong FK (2014) TTS synthesis with bidirectional LSTM based recurrent neural networks. In: Fifteenth annual conference of the international speech communication association

Fovell RG, Fovell MYC (1993) Climate zones of the conterminous United States defined using cluster analysis. J Clim 6(11):2103–2135

Freymueller J (2009) Seasonal position variations and regional reference frame realization. In: Geodetic reference frames, Springer, pp 191–196

Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29:1189–1232

Friedman JH (2002) Stochastic gradient boosting. Comput Stat Data Anal 38(4):367–378

Gers FA, Schmidhuber J (2000) Recurrent nets that time and count. In: Proceedings of the IEEE-INNS-ENNS international joint conference on neural networks. IJCNN 2000. Neural computing: new challenges and perspectives for the new millennium, vol 3. IEEE, pp 189–194

Ghosal S, Karmakar A, Sahay P, Das U (2021) Analysis of lakes over the period of time through image processing. In: Mandal JK, Mukherjee I, Bakshi S, Chatterji S, Sa PK (eds) Computational Intelligence

and Machine Learning, Springer Singapore, Singapore, pp 173–184

Heflin M, Donnellan A, Parker J, Lyzenga G, Moore A, Ludwig LG, Rundle J, Wang J, Pierce M (2020) Automated estimation and tools to extract positions, velocities, breaks, and seasonal terms from daily GNSS measurements: illuminating nonlinear salton trough deformation. Earth Sp Sci 7(7):e2019EA000644

Herring TA, Melbourne TI, Murray MH, Floyd MA, Szeliga WM, King RW, Phillips DA, Puskas CM, Santillan M, Wang L (2016) Plate boundary observatory and related networks: GPS data analysis methods and geodetic products. Rev Geophys 54(4):759–808. https://doi.org/10.1002/2016rg000529

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

Jalal MM, Tasnim Z, Islam MN (2020) Exploring the machine learning algorithms to find the best features for predicting the risk of cardiovascular diseases. In: International conference on intelligent computing & optimization. Springer, pp 559–569

Lambeck K (1988) Geophysical geodesy. Clarendon, Oxford

Li B, Huang J, Feng Y, Wang F, Sang J (2020) A machine learning-based approach for improved orbit predictions of LEO space debris with sparse tracking data from a single station. IEEE Trans Aerosp Electron Syst 56(6):4253–4268. https://doi.org/10.1109/TAES.2020.2989067

Li B, Zhang Y, Huang J, Sang J (2021) Improved orbit predictions using two-line elements through error pattern mining and transferring. Acta Astronaut 188:405–415. https://doi.org/10.1016/j.actaastro.2021.08.002

Li Z, Jiang W, Ding W, Deng L, Peng L (2014) Estimates of minor ocean tide loading displacement and its impact on continuous GPS coordinate time series. Sensors 14(3):5552–5572

Louppe G, Wehenkel L, Sutera A, Geurts P (2013) Understanding variable importances in forests of randomized trees. In: Burges C, Bottou L, Welling M, Ghahramani Z, Weinberger K (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc, vol 26, pp 1–9

Ma X, Tao Z, Wang Y, Yu H, Wang Y (2015) Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. Transp Res C Emerg Technol 54:187–197

Malhotra P, Vig L, Shroff G, Agarwal P et al (2015) Long short term memory networks for anomaly detection in time series. In: Proceedings, vol 89, pp 89–94

Mathews PM, Herring TA, Buffett BA (2002) Modeling of nutation and precession: new nutation series for nonrigid earth and insights into the Earth's interior. J Geophys Res Solid Earth 107(B4):ETG-3

Melachroinos SA, Biancale R, Llubes M, Perosanz F, Lyard F, Vergnolle M, Bouin MN, Masson F, Nicolas J, Morel L et al (2008) Ocean tide loading (OTL) displacements from global and local grids: comparisons to GPS estimates over the shelf of Brittany, France. J Geod 82(6):357–371

Miller JA (1994) Ground water atlas of the United States. Appl Hydrogeol 2(4):59–62

Mohammednour AB, Özdemir AT (2020) GNSS positioning accuracy improvement based on surface meteorological parameters using artificial neural networks. Int J Commun Syst 33(9):e4373. https://doi.org/10.1002/dac.4373

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V et al (2011) Scikit-learn: machine learning in python. J Mach Learn Res 12:2825–2830

Petit G, Luzum B (2010) IERS conventions. Technical report, Bureau International des Poids et mesures sevres (France)

Platt J (1998) Sequential minimal optimization: a fast algorithm for training support vector machines. Tech Rep MSR-TR-98-14

Puskas CM, Meertens CM, Phillips D (2017) Hydrologic loading model displacements from the national and global data assimilation systems (NLDAS and GLDAS). UNAVCO Geodetic Data Service Group

Ribeiro B (2005) Support vector machines for quality monitoring in a plastic injection molding process. IEEE Trans Syst Man Cybern C (Appl Rev) 35(3):401–410

Rothacher M, Mader G (2002) Receiver and satellite antenna phase center offsets and variations. In: Position Paper of the "Antenna Session"

Ruttner P, Hohensinn R, D'Aronco S, Wegner JD, Soja B (2021) Modeling of residual GNSS station motions through meteorological data in a machine learning approach. Remote Sens 14(1):17. https://doi.org/10.3390/rs14010017

Singh VV, Biskupek L, Müller J, Zhang M (2021) Impact of non-tidal station loading in LLR. Adv Space Res 67(12):3925–3941

Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Stat Comput 14(3):199–222

Sun R, Wang G, Zhang W, Hsu LT, Ochieng WY (2020) A gradient boosting decision tree based GPS signal reception classification algorithm. Appl Soft Comput 86:105942

Tesmer V, Steigenberger P, Rothacher M, Boehm J, Meisel B (2009) Annual deformation signals from homogeneously reprocessed VLBI and GPS height time series. J Geod 83(10):973–988

Thurman H (1994) Introductory oceanography. Macmillan, New York

Vapnik V (2013) The nature of statistical learning theory. Springer, Berlin

Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA, Bottou L (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 11(12):3371–3408

Wang J, Jiang W, Li Z, Lu Y (2021) A new multi-scale sliding window LSTM framework (MSSW-LSTM): a case study for GNSS time-series prediction. Remote Sens 13(16):3328. https://doi.org/10.3390/rs13163328

Wang Z, Balog RS (2016) Arc fault and flash detection in photovoltaic systems using wavelet transform and support vector machines. In: 2016 IEEE 43rd photovoltaic specialists conference (PVSC). IEEE, pp 3275–3280

Watson C, Tregoning P, Coleman R (2006) Impact of solid Earth tide-models on GPS coordinate and tropospheric time series. Geophys Res Lett 33:L08306

Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, Cao Y, Gao Q, Macherey K et al (2016) Google's neural machine translation system: bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144

Yan HM, Chen W, Zhu YZ, Zhang WM, Zhong M, Liu GY (2010) Thermal effects on vertical displacement of GPS stations in China. Chin J Geophys 53(2):252–260

Yang B, Yin K, Lacasse S, Liu Z (2019) Time series analysis and long short-term memory neural network to predict landslide displacement. Landslides 16(4):677–694. https://doi.org/10.1007/s10346-018-01127-x

Zheng Y, Lu C, Wu Z, Liao J, Zhang Y, Wang Q (2022) Machine learning-based model for real-time GNSS precipitable water vapor sensing. Geophys Res Lett 49(3):e2021GL096408

Zumberge J, Heflin M, Jefferson D, Watkins M, Webb F (1997) Precise point positioning for the efficient and robust analysis of GPS data from large networks. J Geophys Res Solid Earth 102(B3):5005–5017