Time Series Prediction with Variational Bayesian Nonlinear State-Space Models

Matti Tornio, Antti Honkela, and Juha Karhunen

Adaptive Informatics Research Centre, Helsinki University of Technology
P.O. Box 5400, FI-02015 TKK, Finland
{Matti.Tornio, Antti.Honkela, Juha.Karhunen}@tkk.fi
http://www.cis.hut.fi/projects/bayes/

Abstract. In this paper the variational Bayesian method for learning nonlinear state-space models introduced by Valpola and Karhunen in 2002 is applied to prediction in the ESTSP'07 time series prediction competition data set. The data set is pre-processed by approximately removing the periodic component of the data and the nonlinear state-space model is only learned on the residuals. The model uses multilayer perceptron (MLP) networks to model the nonlinearities of the system which allows the modelling of complex dynamical processes. The variational Bayesian learning approach is resistant to overfitting and allows comparison of different model structures using the derived lower bound on marginal log-likelihood. The desired predictions are evaluated as the mean of a Monte Carlo approximation of the predictive distribution.

1 Introduction

Traditionally, time series prediction is done using models based directly on the past observations of the time series. Perhaps the two most important classes of neural network based solutions used for nonlinear prediction are feedforward autoregressive neural networks and recurrent autoregressive moving average neural networks [10]. However, instead of modelling the system based on past observations, it is also possible to model the same information in a more compact form with a state-space model [3].

This paper uses the nonlinear state-space model (NSSM) introduced by Valpola and Karhunen in 2002 [11] to model a time series. The primary goal of the paper is to apply this publicly available[1] NSSM to the task of time-series prediction as a black box tool.

The nonlinearities of both the dynamics and the mapping from the states to observations are modelled with multilayer perceptron (MLP) networks. Training a nonlinear state-space model is a computationally challenging task and prone to overfitting. The NSSM in [11] uses variational Bayesian learning, which is both resistant against overfitting and computationally effective compared to e.g. sampling methods.

[1] http://www.cis.hut.fi/projects/bayes/software/

2 Nonlinear State-Space Models by Variational Bayesian Learning

2.1 The Model

The variational Bayesian nonlinear state-space model introduced by Valpola and Karhunen in [11] uses a general nonlinear state-space model for the observations $\mathbf{x}(t)$

$$\mathbf{s}(t) = \mathbf{g}(\mathbf{s}(t-1), \boldsymbol{\theta}_\mathbf{g}) + \mathbf{m}(t) \tag{1}$$
$$\mathbf{x}(t) = \mathbf{f}(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f}) + \mathbf{n}(t) \tag{2}$$

with states $\mathbf{s}(t)$, Gaussian innovation $\mathbf{m}$ and noise $\mathbf{n}$, and multi-layer perceptron (MLP) networks to model the nonlinearities $\mathbf{f}$ and $\mathbf{g}$. The functional form of the MLP networks is given by

$$\mathbf{g}(\mathbf{s}(t-1), \boldsymbol{\theta}_\mathbf{g}) = \mathbf{s}(t-1) + \mathbf{D}\tanh(\mathbf{C}\mathbf{s}(t-1) + \mathbf{c}) + \mathbf{d} \tag{3}$$
$$\mathbf{f}(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f}) = \mathbf{B}\tanh(\mathbf{A}\mathbf{s}(t) + \mathbf{a}) + \mathbf{b}, \tag{4}$$

where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ are the network weight matrices and $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{d}$ are the bias vectors. Inference and learning in the model can be made more reliable and efficient than in [11] by using the new linearisation described in [5].

2.2 Variational Bayes

Variational Bayesian learning [7, 2] is based on approximating the posterior distribution $p(\boldsymbol{\theta}, \boldsymbol{S}|\boldsymbol{X}, \mathcal{H})$ with a tractable approximation $q(\boldsymbol{\theta}, \boldsymbol{S}|\boldsymbol{\xi})$, where $\boldsymbol{X} = \{\mathbf{x}(t)|t = 1, \ldots, T\}$ is the data, $\boldsymbol{S} = \{\mathbf{s}(t)|t = 1, \ldots, T\}$ are the latent state values, $\boldsymbol{\theta}$ are the parameters of the model $\mathcal{H}$, and $\boldsymbol{\xi}$ are the (variational) parameters of the approximation. The approximation is fitted by maximising a lower bound on marginal log-likelihood

$$\mathcal{B} = \left\langle \log \frac{p(\boldsymbol{X}, \boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H})}{q(\boldsymbol{S}, \boldsymbol{\theta}|\boldsymbol{\xi})} \right\rangle = \log p(\boldsymbol{X}|\mathcal{H}) - D_{\mathrm{KL}}(q(\boldsymbol{S}, \boldsymbol{\theta}|\boldsymbol{\xi})||p(\boldsymbol{S}, \boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})), \tag{5}$$

where $\langle\cdot\rangle$ denotes expectation over q. This is equivalent to minimising the Kullback–Leibler divergence $D_{\mathrm{KL}}(q||p)$ between q and p [6, 2].

The posterior approximations for the network weights and biases, as well as all the other model parameters except latent states are modelled as Gaussian distributions with a diagonal covariance. The posterior approximation for the latent states is modelled as a Gaussian distribution with an almost diagonal covariance. The correlation between the corresponding components $s_j(t)$ and $s_j(t-1)$ of subsequent state vectors is modelled, however. This is a realistic minimal assumption for modelling the dynamical system and does not increase the computational cost significantly [11].

2.3 Learning

The nonlinear state-space model is learned by numerically maximising the bound (5). This optimisation requires evaluating the value of the bound and its gradient with respect to all the variational parameters $\boldsymbol{\xi}$. To speed up this optimisation, a conjugate

gradient method is used to update the variational parameters of the latent states and the MLP network weights and biases instead of the heuristic algorithm presented in [11]. The other model parameters are updated as described in [11].

At the beginning of the learning, the network weights and biases are initialised to random values drawn from a Gaussian distribution. The latent states are initialised to the first principal components of embedded data vectors [11]. To ensure that the learning does not get stuck in a local minimum early on, the latent states and the model hyperparameters are not updated until the network weights and biases have converged to reasonable values. It is also useful to use multiple different initialisations to avoid local minima.

Modelling noise as part of the state-space model means that the model can filter out most of the noise in the original data set. Dynamics of these smoothed-out observations are often easier to learn than the dynamics of the original data set. State-space based approach can also typically model the system in a more compact form than a neural network model based directly on the past observations.

The variational Bayesian approach also provides a straightforward way to perform model selection. The lower bound on marginal log-likelihood $\mathcal{B}$ can be used as a measure of model quality between models with different structure such as different number of hidden units or different dimensionality of the state-space. Even if there is not enough data for methods such as cross-validation, this lower bound can still be used to evaluate relative model quality [11].

3 Time series prediction

Given data $\boldsymbol{X}$ and background assumptions $\mathcal{H}$, the optimal way to make predictions of an unknown quantity y with respect to mean-squared error is to use the mean of the posterior predictive distribution $p(y|\boldsymbol{X}, \mathcal{H})$ as the point prediction [1].

The easiest way to compute predictions of future observations based on the NSSM is simply to iterate Equation (1) starting from the posterior mean of the latent states corresponding to the last observed data sample. In some cases it can be desirable to ignore the innovation process $\mathbf{m}(t)$ (process noise) while doing these computations, as long predictions can lead to very high variance and the mean values of the predictions thus converge to the long term mean over very long prediction windows.

Even though the same techniques that are used in learning can also be used to compute the predictions, sampling methods typically lead to more accurate inference. Using the same approximation to evaluate $\mathbf{g}(\mathbf{s}(t-1), \boldsymbol{\theta}_{\mathbf{g}})$ as in learning consecutively leads to severe underestimation of predictive variance because the parameters $\boldsymbol{\theta}_{\mathbf{g}}$ used in consecutive steps would be assumed to be two independent sets even though they are the same. This is not a problem in learning which only requires one-step prediction, but for accurate long-term prediction the sampling approach is necessary.

For this purpose, the state values corresponding to the last observed data sample as well as all the network weights are sampled repeatedly from the variational posterior approximations, and the relevant predictions are evaluated using Eqs. (1) and (2) iteratively. This can be computationally much more demanding than using the same

procedure as in learning, but in most cases the time required for sampling is still insignificant compared to the training time of the original model.

In more complicated situations than direct prediction of future values, more advanced inference methods are needed to take into account the available future observations. This inference can be made more efficient by the method described in [9]. An example of this approach is given in [8], where the NSSM described in this paper is used to make predictions for a cart-pole system and at each time instant the new latent states are inferred from the latest observations and the future is predicted based on the model and the control signal.

4 Experiments: Prediction competition

The data set in the experiment was the prediction competition data set for ESTSP'07[2]. This is a one-dimensional time series with 875 samples. The data set appears strongly periodic with a period of approximately 52 samples. To make prediction of the time series easier, the data set is averaged over all the full periods (samples from 1 to 832) and this average is subtracted from the original data set.

After this preprocessing, a state-space with three dimensions was used to model the dynamics of the residual time series. A three-dimensional state-space was chosen because it resulted in the best value for the bound on marginal log-likelihood $\mathcal{B}$. Both the observation MLP network and the dynamical MLP network had 20 hidden units. During 400 first iterations of the learning, an embedded version of the data set was used as described in [11]. The embedded data vector was $\hat{\mathbf{x}}(t) = [\mathbf{x}^T(t)\ \mathbf{x}^T(t-1)\ \mathbf{x}^T(t-2)\ \mathbf{x}^T(t-4)\ \mathbf{x}^T(t-8)\ \mathbf{x}^T(t-16)]^T$. The latent states were initialised to the three first principal components of the embedded data vector. The learning of the model took about three hours on a 2.2 GHz AMD Opteron processor.

A short overview of the preprocessing and prediction algorithm for a periodic time series $\mathbf{x}$ with length T, an approximated period T_{per} and a number of full periods N_{per} can be seen in Table 1.

The predictions made using the sampling method with 1000 particles for the next 61 time steps can be seen in Fig. 1. The predictions were computed with the innovation process ignored. The prediction length of 61 time steps was chosen so that the data set with the predictions contains 18 full periods of 52 samples. The reconstruction of the residual data set based on the model can be seen in Fig. 2. The latent state-space can be seen in Fig. 3. As some of the state components appear clearly periodic, it is likely that the period of 52 samples used in preprocessing was slightly incorrect. The original data set may also have contained components with longer periods.

From the Figs. 2 and 3 it is clear that the model of the dynamics of the residual system has a large associated uncertainty. This is natural, as the residual data set seen in Fig. 2 is quite hard to predict as it appears to have very little structure and there is little data compared to the very broad prior over different models. This uncertainty can also be seen in the predictions of the states that are very close to the long-term mean, along with large error bars for the first two states. These large error bars do not affect

[2] Available at http://estsp2007.org/files/competition_data.txt

Table 1: Prediction algorithm for periodic data.

Learning:

1. Compute the periodic component $\mathbf{x}_{per}$ over the full periods. The periodic component is given by
   $\mathbf{x}_{per}(i) = \frac{1}{N_{per}} \sum_{j=1}^{N_{per}} \mathbf{x}(\mathrm{mod}(i, T_{per}) + (j-1) \cdot T_{per})$,
   where we define $\mathrm{mod}(a \cdot n, n) = n$
2. Subtract the periodic component from the original data set $\mathbf{x}$. The resulting residual data set for each $i = 1 \ldots T$ is given by
   $\mathbf{x}_{res}(i) = \mathbf{x}(i) - \mathbf{x}_{per}(i)$
3. Use the NSSM to learn a state-space representation for the residual data set $\mathbf{x}_{res}$

Prediction:

4. Sample the initial state for the prediction $\mathbf{s}(T)$ and the network parameters $\boldsymbol{\theta}_g$ and $\boldsymbol{\theta}_f$ from the model learned in step 3
5. Iterate Equations (1) and (2) using the values sampled at step 4
6. Add the periodic component back to the predicted samples to get the final predictions

the predictions of the output, as the contribution of the third state to it is roughly 1000 times larger than those of the first two.

5 Discussion

The NSSM in [11] has been previously applied to several difficult prediction problems. One such example is the prediction of the dynamics of a complex system consisting of two Lorenz processes and a harmonic oscillator described in [11]. In [8], the model was used to predict the dynamics of a cart-pole system and the predictions were then used by a nonlinear model predictive controller. Even though the NSSM is better suited to modelling higher dimensional systems, it can also be used for modelling one-dimensional time series as in this paper.

The state-space model from [11] requires that the data set is evenly sampled. However, the recent extension of the model to continuous-time described in [4] allows the prediction of unevenly sampled time series as well. Continuous-time models also allow modelling both the short-term and long-term dynamics of the system more easily.

In theory the NSSM could have been used to predict the original data set without any preprocessing. However, with the limited amount of available data and a flexible prior over a large space of possible nonlinear models, there would have been significant posterior uncertainty on the dynamics and the global prediction would soon have converged to the long-term mean with large variance. In order to attain more meaningful predictions, more prior information such as the apparent periodicity of the signal have to be taken into consideration.

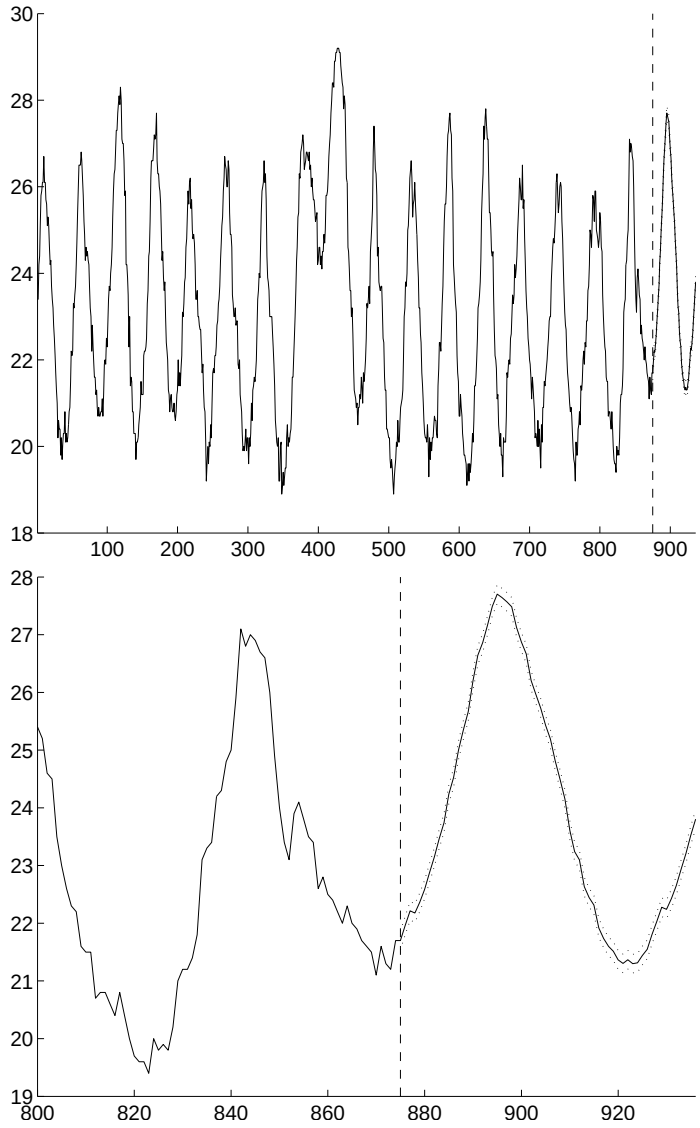

Fig. 1: Top: The original time series and the predicted 61 next time steps. Bottom: The original time series starting from time instant 800 and the predicted 61 next time steps. The dotted lines in both figures represent pseudo 95 % confidence intervals. Note that the intervals are smaller than in reality as the variance caused by the innovation is ignored.

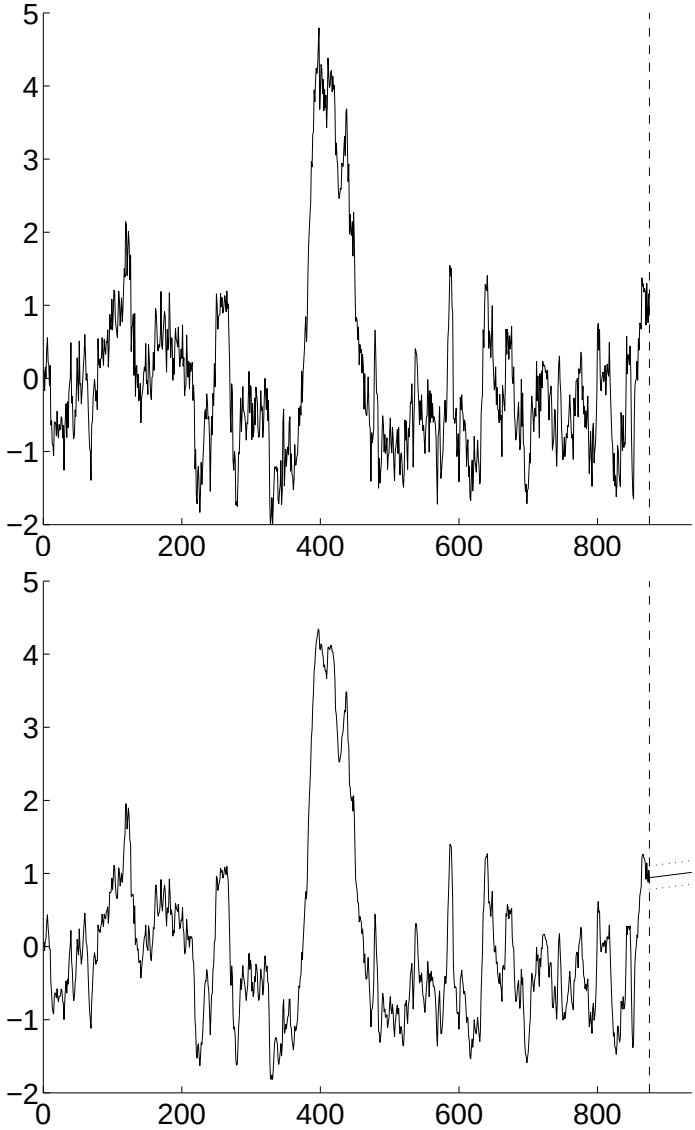

Fig. 2: Top: The original residual data set. Bottom: The mean of the reconstruction of the residual data set based on the model and its predictions. The reconstructed data set is the original data set with the observation noise filtered out. The dotted lines represent pseudo 95 % confidence intervals. The intervals are again smaller than in reality as the variance caused by the innovation is ignored.

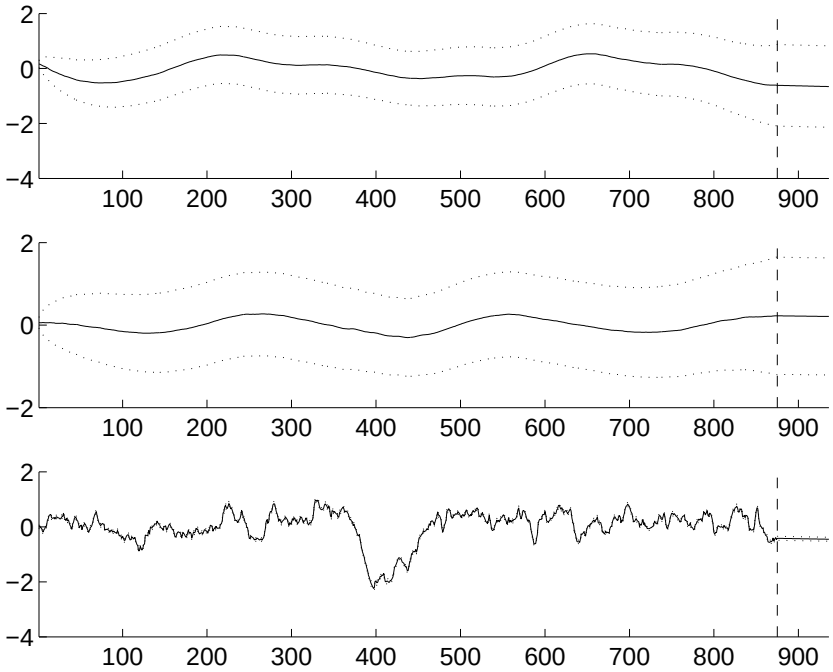

Fig. 3: The three dimensional latent state-space. Each of the three components of the state vector and their predictions are shown in its own figure. The dotted lines represent pseudo 95 % confidence intervals. The intervals are again smaller than in reality as the variance caused by the innovation is ignored.

6 Conclusion

In this paper we have applied the variational Bayesian NSSM of Valpola and Karhunen [11] to time series prediction. The prediction results with the ESTSP'07 prediction competition data set are presented.

Using state-space models for time series prediction has several benefits. The use of latent states allows easy handling of noisy data as the noise can be filtered out of the latent states. The state-space also allows creating models for partially observed systems, where some of the observations are not available. Finally, state-space models can usually represent the dynamics of the model in a more compact form than a model based directly on the past observations. Using variational Bayesian methods for learning these NSSMs is both resistant against overfitting and provides a cost function which can be used for model comparison.

Acknowledgments

The authors would like to thank Tapani Raiko for fruitful discussions. This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

References

[1] J. M. Bernardo and A. F. M. Smith. *Bayesian Theory*. J. Wiley, 2000.

[2] C. Bishop. *Pattern Recognition and Machince Learning*. Springer, Cambridge, 2006.

[3] S. Haykin. *Neural Networks – A Comprehensive Foundation, 2nd ed.* Prentice-Hall, 1999.

[4] A. Honkela, M. Tornio, and T. Raiko. Variational Bayes for continuous-time nonlinear state-space models. In *NIPS*2006 Workshop on Dynamical Systems, Stochastic Processes and Bayesian Inference*, Whistler, B.C., Canada, 2006.

[5] A. Honkela and H. Valpola. Unsupervised variational Bayesian learning of nonlinear models. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 593–600. MIT Press, Cambridge, MA, USA, 2005.

[6] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. In M. Jordan, editor, *Learning in Graphical Models*, pages 105–161. The MIT Press, Cambridge, MA, USA, 1999.

[7] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[8] T. Raiko and M. Tornio. Learning nonlinear state-space models for control. In *Proc. Int. Joint Conf. on Neural Networks (IJCNN'05)*, pages 815–820, Montreal, Canada, 2005.

[9] T. Raiko, M. Tornio, A. Honkela, and J. Karhunen. State inference in variational Bayesian nonlinear state-space models. In *Proceedings of the 6th International Conference on Independent Component Analysis and Blind Source Separation (ICA 2006)*, pages 222–229, Charleston, South Carolina, USA, March 2006.

[10] A. Trapletti. *On Neural Networks as Statistical Time Series Models*. PhD thesis, Technische Universität Wien, 2000.

[11] H. Valpola and J. Karhunen. An unsupervised ensemble learning method for nonlinear dynamic state-space models. *Neural Computation*, 14(11):2647–2692, 2002.