ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

Recurrent Restricted Kernel Machines for Time-series Forecasting

Arun Pandey, Hannes De Meulemeester, Henri De Plaen,
Bart De Moor and Johan A.K. Suykens *

ESAT-STADIUS, KU Leuven
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

Abstract. In this paper, we propose a novel method for time-series modeling and forecasting. It is based on the temporal formulation of Restricted Kernel Machines leading to a dynamical equation in the latent-variables. Forecasting involves finding the next latent variable and then solving a pre-image problem to predict a new-point in the input space. Further, we benchmark our model on several standard data sets against other well-known time-series models.

1 Introduction

In [1], Suykens proposed a new framework called Restricted Kernel Machines (RKM), which provides a representation of kernel methods with visible and latent variables. This representation has an objective function that is similar to the energy function of Restricted Boltzmann Machines (RBM), thus linking kernel methods with RBMs. Training and prediction requires characterizing the stationary points for the unknowns in the objective. This in turn provides the training and prediction schemes in the kernel methods setting. Restricted Kernel Machines have been previously extended for different tasks such as classification [1], generation [2, 3] and outlier detection [4]. We further extend the RKM framework to time series modeling by introducing a temporal correlation on the latent variables which provides powerful representation learning capabilities, and a novel forecasting method. The formulation draws connections with kernel autoregressive models [5] and Temporal Restricted Boltzmann Machines (TRBM) [6, 7], which are explored in the next sections.

*European Research Council under the European Union's Horizon 2020 research and innovation programme: ERC Advanced Grants agreements E-DUALITY(No 787960) and Back to the Roots (No 885682). This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information. Research Council KUL: Optimization frameworks for deep kernel machines C14/18/068; Research Fund (projects C16/15/059, C3/19/053, C24/18/022, C3/20/117, C3I-21-00316); Industrial Research Fund (Fellowships 13-0260, IOFm/16/004, IOFm/20/002) and several Leuven Research and Development bilateral industrial projects. Flemish Government Agencies: FWO: projects: GOA4917N (Deep Restricted Kernel Machines: Methods and Foundations), PhD/Postdoc grant, EOS Project no G0F6718N (SeLMA), SBO project S005319N, Infrastructure project I013218N, TBM Project T001919N, PhD Grant (SB/1SA1319N); EWI: the Flanders AI Research Program; VLAIO: CSBO (HBC.2021.0076) Baekeland PhD (HBC.20192204). This research received funding from the Flemish Government (AI Research Program). Other funding: Foundation 'Kom op tegen Kanker', CM (Christelijke Mutualiteit), Leuven.AI institute.

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

2 Recurrent Restricted Kernel Machines

2.1 Training

Our main objective is to capture the dynamics of a training data set $\mathcal{X}_T$ containing T time steps $\{\boldsymbol{x}_t\}_{t=1}^T \subset \mathscr{X}$. We define a feature map[1] $\phi : \mathscr{X} \to \mathscr{H}$ with $\mathscr{H}$ a (possibly infinite dimensional) Reproducing Kernel Hilbert Space (RKHS; see [8] for more details). Such a feature map could be constructed explicitly or implicitly via a kernel function $k(\boldsymbol{x},\boldsymbol{y}) : \mathscr{X}^2 \to \mathbb{R} : (\boldsymbol{x},\boldsymbol{y}) \mapsto \langle \phi(\boldsymbol{x}),\phi(\boldsymbol{y})\rangle_{\mathscr{H}}$. We also define a linear operator[2] $\boldsymbol{V} : \mathbb{R}^s \to \mathscr{H}$ with $s \leq T$ and it's adjoint $\boldsymbol{V}^\star$. Each datapoint $\boldsymbol{x}_t$ will be associated to a latent variable $\boldsymbol{h}_t \in \mathbb{R}^s$ through a pairing term $\langle \phi(\boldsymbol{x}_t), \boldsymbol{V}\boldsymbol{h}_t\rangle_{\mathscr{H}}$. To also capture time dependence, we only add one extra term compared to the original RKM framework [1]: time correlation in the latent space using a set of non-zero lag-dependent coefficients $\mathcal{A}_T = \{a_{t,l} | 1 \leq t \leq T \text{ and } 0 \leq l < p\}$ with $p \in \mathbb{Z}^+$ a lag parameter (other coefficients are assumed to be 0). Then consider the following objective function with diagonal matrix $\boldsymbol{\Lambda}$:

$$J_T(\boldsymbol{V},\mathcal{H}_T,\mathcal{X}_T) = \sum_{t=1}^{T} \Big[\overbrace{- \langle \phi(\boldsymbol{x}_t), \boldsymbol{V}\boldsymbol{h}_t\rangle_{\mathscr{H}}}^{\text{feature-space pairing}} \overbrace{- \sum_{l=0}^{p} a_{t,l} \boldsymbol{h}_{t-l}^\top \boldsymbol{h}_t}^{\text{temporal covariance}} + \underbrace{\frac{1}{2}\big(\boldsymbol{h}_t^\top \boldsymbol{\Lambda} \boldsymbol{h}_t + \|\phi(\boldsymbol{x}_t)\|_{\mathscr{H}}^2\big)\Big] + \frac{1}{2}\mathrm{Tr}(\boldsymbol{V}^*\boldsymbol{V})}_{\text{regularization}}. \tag{1}$$

Interpreting the objective function. The first two terms in (1) are similar to the TRBM's energy function [6] which is used (along with bias terms) to define a joint-probability distribution over some visible variables $\{\boldsymbol{x} \in \mathscr{X}\}$ and latent units $\{\boldsymbol{h} \in \{0,1\}^s\}$. It is trained with a maximum-likelihood approach where the gradients are approximated with contrastive divergence. In contrast, we propose to map the data into feature-space and center it to eliminate the need of a bias term. The first term in the objective maximizes the pairing between the visible variables in the feature-space $\{\phi(\boldsymbol{x}) : \boldsymbol{x} \in \mathscr{X}\}$ and latent variables $\{\boldsymbol{h} \in \mathbb{R}^s\}$. The second term maximizes the temporal covariance between current and past latent vectors. The regularization terms and constraints are meant to bound the objective.

Solving the objective. Given the visible variables, characterizing the stationary points of $J_T(\boldsymbol{V},\mathcal{H}_T|\mathcal{X}_T)$ in the latent variables and the pairing linear operator

[1] Throughout our discussion, we assume that the feature vectors are centered in the feature-space *i.e.* $\bar{\phi}(\boldsymbol{x}) = \phi(\boldsymbol{x}) - \mu_\phi$ with $\mu_\phi = \mathbb{E}_{\boldsymbol{\xi}\sim\mathcal{X}}[\phi(\boldsymbol{\xi})]$. Using an implicit formulation, it suffices to notice that $\langle \bar{\phi}(\boldsymbol{x}), \bar{\phi}(\boldsymbol{y})\rangle_{\mathscr{H}} = \langle \phi(\boldsymbol{x}) - \mu_\phi, \phi(\boldsymbol{y}) - \mu_\phi\rangle_{\mathscr{H}} = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{y})\rangle_{\mathscr{H}} - \langle \mu_\phi, \phi(\boldsymbol{y})\rangle_{\mathscr{H}} - \langle \phi(\boldsymbol{x}), \mu_\phi)\rangle_{\mathscr{H}} + \langle \mu_\phi, \mu_\phi\rangle_{\mathscr{H}} = k(\boldsymbol{x},\boldsymbol{y}) - \mathbb{E}_{\boldsymbol{\xi}\sim\mathcal{X}}[k(\boldsymbol{\xi},\boldsymbol{y})] - \mathbb{E}_{\boldsymbol{\zeta}\sim\mathcal{X}}[k(\boldsymbol{x},\boldsymbol{\zeta})] + \mathbb{E}_{\boldsymbol{\xi},\boldsymbol{\zeta}\sim\mathcal{X}}[k(\boldsymbol{\xi},\boldsymbol{\zeta})]$. In practice, we can compute statistics on $\mathcal{X}_T$.

[2] The linear operator $\boldsymbol{V}$ is often referred to as a *matrix* as it only exists explicitly in the case of finite dimensional Hilbert spaces $\mathscr{H}$. It then takes the form $\boldsymbol{V} \in \mathbb{R}^{\dim(\mathscr{H}) \times s}$ and $\boldsymbol{V}^* = \boldsymbol{V}^\top$.

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1.
Available from http://www.i6doc.com/en/.

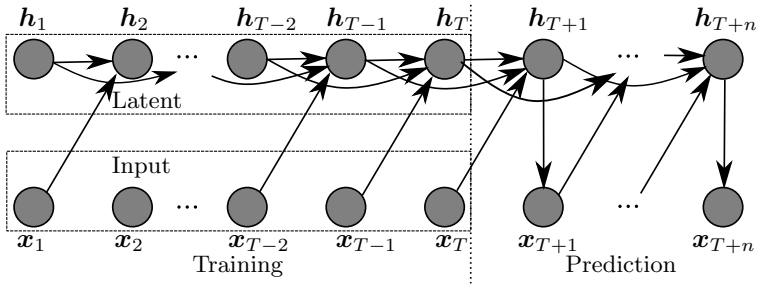

Fig. 1: Dependency graph of the Recurrent RKM model's *training* (3) and *prediction* (8) scheme for $a_{t,l}=1$ if $l=1$ and $a_{t,l}=0$ otherwise, and a linear kernel on $\mathscr{X}$.

leads to the following equations for $1\leq t\leq T$, where $\otimes$ is the outer product:

$$\begin{cases} \dfrac{\partial J_T}{\partial \boldsymbol{V}} = -\sum_{t=1}^{T} \phi(\boldsymbol{x}_t)\otimes \boldsymbol{h}_t + \boldsymbol{V} = 0 \quad \Longrightarrow \quad \boldsymbol{V} = \sum_{t=1}^{T} \phi(\boldsymbol{x}_t)\otimes \boldsymbol{h}_t, & (2) \\ \dfrac{\partial J_T}{\partial \boldsymbol{h}_t} = -\boldsymbol{V}^*\phi(\boldsymbol{x}_t) + \boldsymbol{\Lambda}\boldsymbol{h}_t - \left[\sum_{l=0}^{p} a_{t,l}\boldsymbol{h}_{t-l} + \sum_{l=1}^{p} a_{t+l,l}\boldsymbol{h}_{t+l}\right] = 0. & (3) \end{cases}$$

Eliminating $\boldsymbol{V}$ from (3) using (2) gives the following solution

$$[\boldsymbol{K}(\mathcal{X}_T) + \boldsymbol{A}]\boldsymbol{H}^\top = \boldsymbol{H}^\top \boldsymbol{\Lambda}, \tag{4}$$

where $\boldsymbol{H} = [\boldsymbol{h}_1,...,\boldsymbol{h}_T] \in \mathbb{R}^{s\times T}$, $\boldsymbol{A}_{i,j} = a_{i,i-j}$ for $i \geq j$ and $\boldsymbol{A}_{i,j} = a_{j-i,j}$ for $i < j$, and kernel matrix $\boldsymbol{K}(\mathcal{X}_T) = [k(\boldsymbol{x}_t, \boldsymbol{x}_{t'})]_{t,t'=1}^{T}$. We can see that any s eigenpairs of $\boldsymbol{K}(\mathcal{X}_T)+\boldsymbol{A}$ satisfies (4). The symmetry of $\boldsymbol{A}$ and of the kernel guarantees these eigenvalues to be real. If $\boldsymbol{A}$ is also positive semi-definite, then these eigenvalues are also guaranteed to be positive. An example of such a choice is $a_{t,l} = \exp\left(-l^2/2\sigma_t^2\right)$ for any bandwidth $\sigma_t \in \mathbb{R}^+$. Alternatively, $a_{t,l}$ can be a compactly supported function, for instance, an indicator $a_{t,l} = \mathbf{1}_{\{1,...,p\}}(l)$ (Fig. 1 is an example with $p=1$). Both these choices are however translational invariant, *i.e.* $a_{t,l} = a_l$ for any $a_{t,l} \in \mathcal{A}_T$. In other words, the local effect of time is the same at all time steps.

2.2 Prediction

The main idea is to generate $\{\boldsymbol{x}_{T+1},...,\boldsymbol{x}_{T+n}\}$ for some $n>0$. To do so, we now work in $\mathcal{X}_{T+n} = \mathcal{X}_T \cup \{\boldsymbol{x}_{T+1},...,\boldsymbol{x}_{T+n}\}$, $\mathcal{H}_{T+n} = \mathcal{H}_T \cup \{\boldsymbol{h}_{T+1},...,\boldsymbol{h}_{T+n}\}$ and consider $\mathcal{A}_{T+n}$. This gives the following objective

$$\begin{aligned} J_{T+n}(\boldsymbol{V}, \mathcal{H}_{T+n}, \mathcal{X}_{T+n}) = \sum_{t=1}^{T+n} \Big[&- \langle \phi(\boldsymbol{x}_t), \boldsymbol{V}\boldsymbol{h}_t \rangle_{\mathscr{H}} - \sum_{l=0}^{p} a_{t,l}\boldsymbol{h}_{t-l}^\top \boldsymbol{h}_t \\ &+ \frac{1}{2}\left(\boldsymbol{h}_t^\top \boldsymbol{\Lambda}\boldsymbol{h}_t + \|\phi(\boldsymbol{x}_t)\|_{\mathscr{H}}^2\right)\Big] + \frac{1}{2}\mathrm{Tr}(\boldsymbol{V}^*\boldsymbol{V}). \end{aligned} \tag{5}$$

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

Given the learned $\boldsymbol{V}$ from the training, characterizing the stationary points of $J_{T+n}(\mathcal{X}_{T+n},\mathcal{H}_{T+n}|\boldsymbol{V})$ in terms of visible and latent variables gives for $1\leq t\leq T+n$

$$\begin{cases} \dfrac{\partial J_{T+n}}{\partial \phi(\boldsymbol{x}_t)} = -\boldsymbol{V}\boldsymbol{h}_t + \phi(\boldsymbol{x}_t) = 0 \quad \Longrightarrow \quad \phi(\boldsymbol{x}_t) = \boldsymbol{V}\boldsymbol{h}_t, & (6) \\ \dfrac{\partial J_{T+n}}{\partial \boldsymbol{h}_t} = -\boldsymbol{V}^*\phi(\boldsymbol{x}_t) + \boldsymbol{\Lambda}\boldsymbol{h}_t - \left[\sum_{k=0}^{p} a_{t,l}\boldsymbol{h}_{t-l} + \sum_{l=1}^{p} a_{t+l,l}\boldsymbol{h}_{t+l}\right] = 0. & (7) \end{cases}$$

We first notice that $\phi(\boldsymbol{x}_t) = \boldsymbol{V}\boldsymbol{h}_t$ is true for all $1 \leq t \leq T$. Furthermore, we also have $\partial J_{T+n}/\partial \boldsymbol{h}_t = \partial J_T/\partial \boldsymbol{h}_t$ for all $1 \leq t \leq T-p$. Using $\partial J_{T+n}/\partial \boldsymbol{h}_t = 0$ (7) and the obtained $\boldsymbol{V}$ (2), with $t = T-p+1$, we can find an expression for $\boldsymbol{h}_{T+1}$. Iteratively, we can find an expression for $\boldsymbol{h}_{T+m}$ with $t = T-p+m$, until $m = n$:

$$\begin{aligned} a_{T+m,t}\boldsymbol{h}_{T+m} = &\Big[\boldsymbol{H}\boldsymbol{A}\boldsymbol{H}^\top - a_{T+m-p,0}\mathbb{I}_s\Big]\boldsymbol{h}_{T+m-p} \\ &- \Big[\sum_{l=1}^{p} a_{T+m-p,l}\boldsymbol{h}_{T+m-p-l} + \sum_{l=1}^{p-1} a_{T+m-p+l,l}\boldsymbol{h}_{T+m-p+l}\Big]. \end{aligned} \tag{8}$$

This can now be used in (6) to find $\phi(\boldsymbol{x}_{T+m})$, again with $t = T+m$:

$$\phi(\boldsymbol{x}_{T+m}) = \boldsymbol{V}\boldsymbol{h}_{T+m} = \sum_{t'=1}^{T} \phi(\boldsymbol{x}_{t'})\boldsymbol{h}_{t'}^\top \boldsymbol{h}_{T+m}. \tag{9}$$

Finally, to obtain new data points $\{\boldsymbol{x}_t\}_{t=T+1}^{T+n}$ in the input space, the *pre-image* problem on (9) needs to be solved.

Solving the pre-image problem. An advantage of using a kernel function, $k(\boldsymbol{x},\boldsymbol{y}) = \langle\phi(\boldsymbol{x}),\phi(\boldsymbol{y})\rangle_{\mathscr{H}}$, is that all computations can be implicitly performed in feature space and the exact mapping $\phi: \mathscr{X} \to \mathscr{H}$ is not required. Working with an implicit feature map however gives rise to the pre-image problem. Given a point $\boldsymbol{\psi} \in \mathscr{H}$, find $\boldsymbol{x} \in \mathscr{X}$ such that $\boldsymbol{\psi} = \phi(\boldsymbol{x})$. This pre-image problem is known to be ill-posed as the exact pre-image might not exist [9]. Instead, an optimization problem is considered to find the approximate pre-image $\tilde{\boldsymbol{x}} = \operatorname{argmin}_{\tilde{\boldsymbol{x}}\in\mathscr{X}} \|\boldsymbol{\psi} - \phi(\tilde{\boldsymbol{x}})\|_{\mathscr{H}}^2$. We employ two different pre-image methods in this work to solve (9): kernel smoother [10] and kernel ridge regression [11].

Computational complexity. The eigendecomposition during training (see (4)) requires $\mathcal{O}(T^3)$ operations, the complexity of the predictions in latent space $\mathcal{O}(p)$ and in input space with the kernel-smoother $\mathcal{O}(T)$, whereas training the kernel-ridge regression $\mathcal{O}(T^3)$ since it involves solving a linear-system.

3 Experiments

We illustrate the representation learning capabilities by considering a simple sine wave as input to the RRKM model and exploring its latent space. Fig. 2 shows the latent space embedding of the learned sine wave and evolution of forecasted latent variables. Dynamics in the data are well represented in the latent space

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

and the forecasted latent variables continue to follow the training trajectory. In Fig. 3, we perform an ablation study on the Santa Fe data set to identify the effect of hyper-parameters on the forecasting performance. We vary bandwidths σ_x,σ_t and latent-space dimension s. The study shows that σ_t captures phase-shift, σ_x captures amplitude and s capture higher and lower frequencies.

The proposed model is compared to a recurrent neural network (RNN) and an ARMA model which are two of the most popular methods used in time series forecasting. On each data set [3], and method, hyperparameter tuning has been performed and the result of the best set of parameters, quantified as the mean squared error, is shown in Table 1. For all methods, the entire validation set is forecasted, in recursive fashion, starting from the end of the training set.

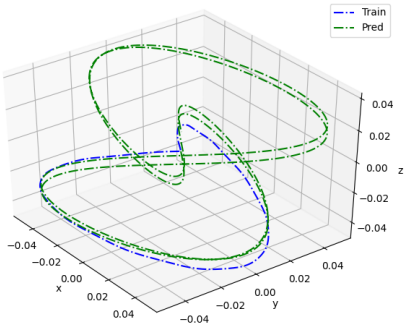

Fig. 2: Training and predicted latent variables of a sinusoidal data set.

When comparing to the baseline methods, RRKM is comparable or better. The RNN can have a better result, however, due to its stochastic nature, its performance has high variability while the RRKM is deterministic for the same parameters.

Table 1: Mean squared error on the forecasted data. Standard deviation for 10 iterations between brackets for the stochastic models.

Data	**RNN**	**ARMA**	**RRKM (Ours)**
Santa Fe	3075.06 ($\pm$794.10)	2224.55	**119.06**
Chickenpox	34329.95 ($\pm$9513.07)	23571.35	**20716.91**
Energy	16002.11 ($\pm$1809.89)	24797.40	**12764.097**
Turbine	2401.12 ($\pm$644.53)	1317.67	**1299.915**

4 Conclusion

In this work, we introduced the recurrent restricted kernel machine model. This framework provides new insights and ideas for time series modeling including latent space dynamics and a novel forecasting method. For future work, we believe a further exploration of the representation learning capabilities in latent space can provide new ways to interpret the data. Additionally, besides the topics mentioned in this work, the framework can be extended towards other tasks involving time series such as denoising, handling missing values and classification.

[3] `https://rdrr.io/cran/TSPred/man/SantaFe.A.html`, `https://archive.ics.uci.edu/ml/datasets/Hungarian+Chickenpox+Cases`, `https://archive.ics.uci.edu/ml/datasets/Appliances+energy+prediction`, `https://archive.ics.uci.edu/ml/datasets/Gas+Turbine+CO+and+NOx+Emission+Data+Set`.

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1.
Available from http://www.i6doc.com/en/.

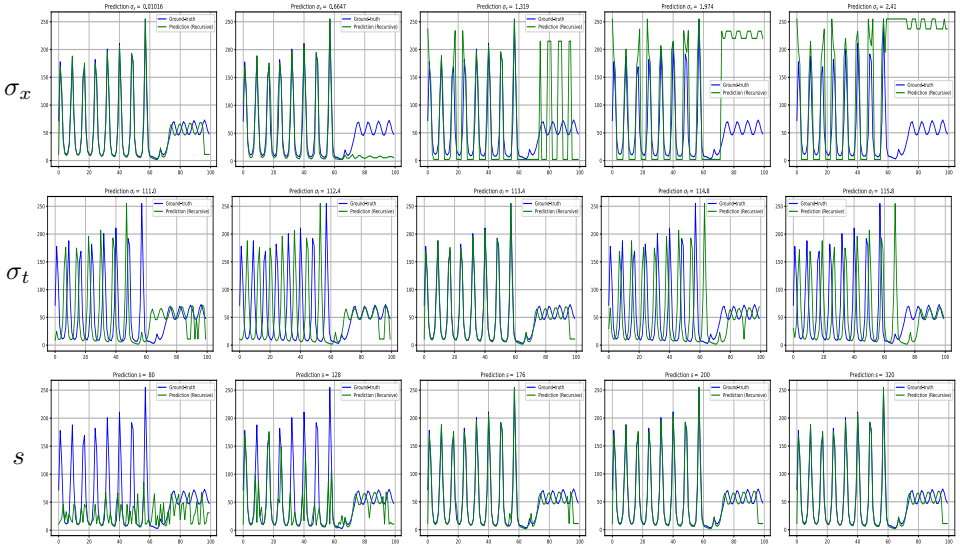

Fig. 3: Ablation study on the Santa Fe laser data set.

References

[1] Johan A. K. Suykens. Deep restricted kernel machines using conjugate feature duality. *Neural Computation*, 29(8):2123–2163, 2017.

[2] Arun Pandey, Joachim Schreurs, and Johan A.K. Suykens. Generative restricted kernel machines: A framework for multi-view generation and disentangled feature learning. *Neural Networks*, 135:177–191, 2021.

[3] Arun Pandey, Michaël Fanuel, Joachim Schreurs, and Johan A. K. Suykens. Disentangled representation learning and generation with manifold optimization. *To appear in Neural Computation. CoRR*, abs/2006.07046, 2020.

[4] Francesco Tonin, Panagiotis Patrinos, and Johan A.K. Suykens. Unsupervised learning of disentangled representations in deep restricted kernel machines with orthogonality constraints. *Neural Networks*, 142:661–679, 2021.

[5] Maya Kallas, Paul Honeine, Clovis Francis, and Hassan Amoud. Kernel autoregressive models using Yule-Walker equations. *Signal Processing*, 93(11):3053–3061, 2013.

[6] Ilya Sutskever, Geoffrey E. Hinton, and Graham W. Taylor. The recurrent temporal restricted boltzmann machine. In *Advances in Neural Information Processing Systems*, volume 21, 2008.

[7] Takayuki Osogami. Boltzmann machines for time-series. *CoRR*, abs/1708.06004, 2019.

[8] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In *International conference on artificial neural networks*, pages 583–588. Springer, 1997.

[9] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.

[10] Joachim Schreurs and Johan A. K. Suykens. Generative kernel PCA. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning* , pages 129–134, 2018.

[11] Jason Weston, Bernhard Schölkopf, and Gökhan Bakir. Learning to find pre-images. In *Advances in Neural Information Processing Systems*, volume 16, 2003.