Autoencoders

An Introduction

Author:	Paul O'Leary and Anika Terbuch
Date:	6 Jule 2022
Document Number:	CoA2020-MLSS
Conference:	Summer School Presentation

Chair of Automation
Department Product Engineering, University of Leoben
Peter Tunner Straße 27, 8700 Leoben, Austria

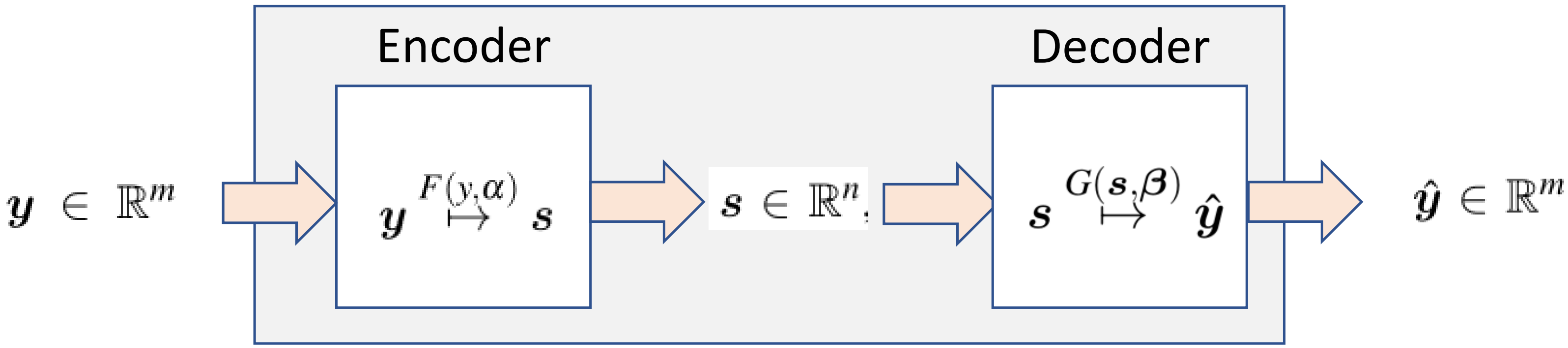
phone: +43/(0)3842/402-5301
fax: +43/(0)3842/402-5302
email: automation@unileoben.ac.at
web: automation.unileoben.ac.at

Context

1. The primary focus in this presentation is on time-series analysis.
2. Application to real-time machine data that for multi-variate time-series (MVTS).
3. Autoencoders will be considered as a means of identifying relevant information in data and with this to enable dimensionality reduction.
4. Autoencoders can be trained in an unsupervised manner, this alleviates the need for labelled data.
5. Final goal is hybrid-learning, i.e., combining: a) a-priori knowledge, b) analytical techniques and c) machine learning.

Elements of an Autoencoder

Autoencoder structure

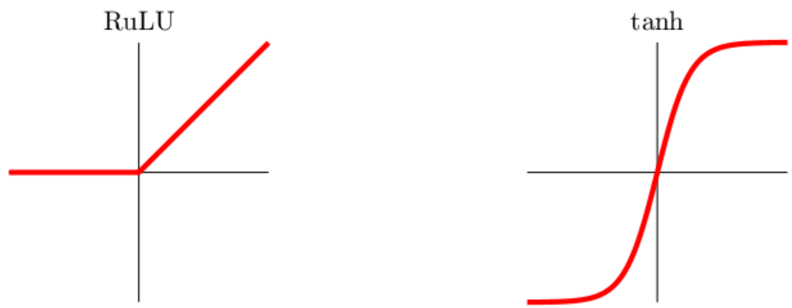


Single layer model

$$\boldsymbol{s} = f(\boldsymbol{W}_e\,\boldsymbol{y} + \boldsymbol{b}_e) \qquad \in \mathbb{R}^n$$

$$\hat{\boldsymbol{y}} = g(\boldsymbol{W}_d\,\boldsymbol{y} + \boldsymbol{b}_d) \qquad \in \mathbb{R}^m$$

Both f(x) and g(x) are activation functions, e.g.



Note both these activation functions have significant regions which are linear

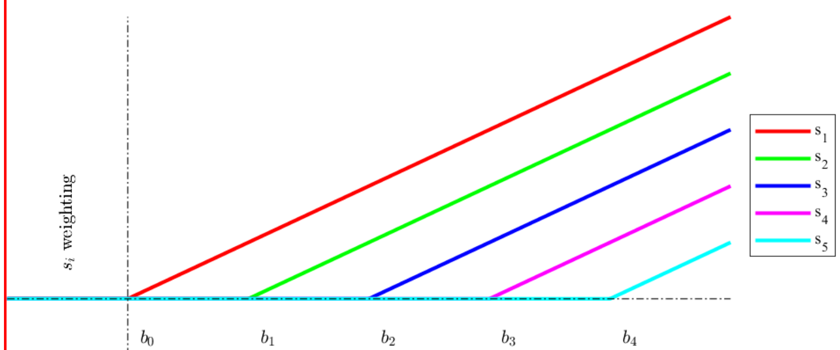
With the ReUL activation we have

$$z_i = \boldsymbol{w}_i + b_i$$

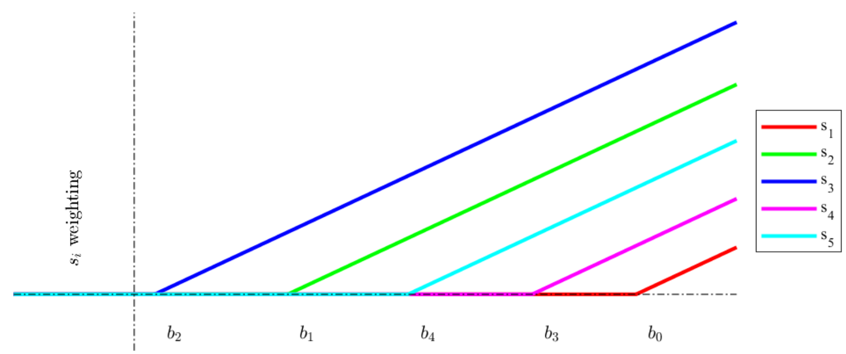
$$s_i = f(z_i)$$

$$\Rightarrow s_i = \boldsymbol{w}_i \boldsymbol{y} + b_i \quad \text{if } s_i > 0.$$

If the b’s were sorted



If the b’s were random



With the ReUL we always have a piece wise function of linear operations on y.

First use in the Context of ML

First definition in a learning context:

„ Learning Internal Representations by Error Propagation“ by: Rumelhart, **Hinton** and Williams, 1965.

TABLE 5

Input Patterns		Hidden Unit Patterns				Output Patterns
10000000	→	.5	0	0	→	10000000
01000000	→	0	1	0	→	01000000
00100000	→	1	1	0	→	00100000
00010000	→	1	1	1	→	00010000
00001000	→	0	1	1	→	00001000
00000100	→	.5	0	1	→	00000100
00000010	→	1	0	.5	→	00000010
00000001	→	0	0	.5	→	00000001

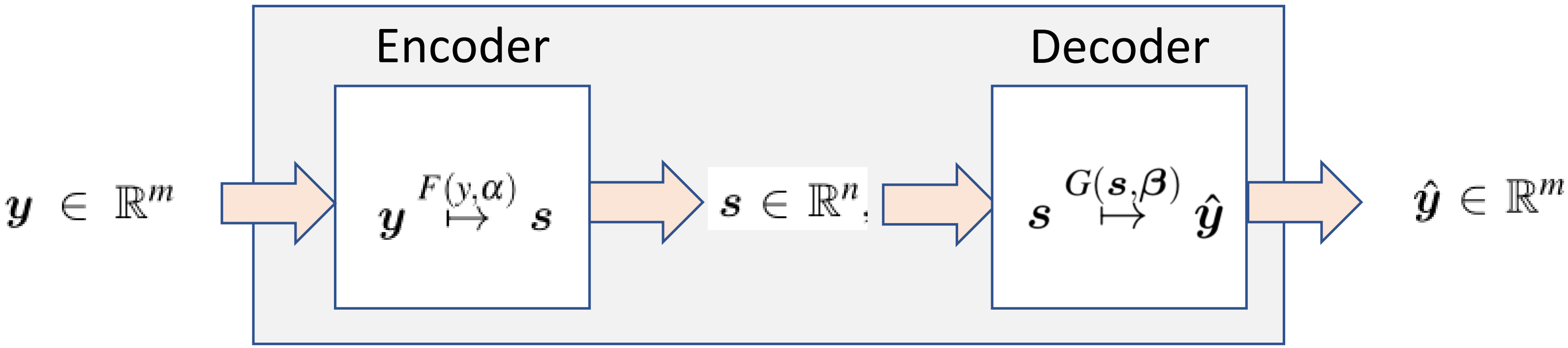
TABLE 8

Input Patterns		Hidden Unit Patterns		Output Patterns
00 + 00	→	111	→	000
00 + 01	→	110	→	001
00 + 10	→	011	→	010
00 + 11	→	010	→	011
01 + 00	→	110	→	001
01 + 01	→	010	→	010
01 + 10	→	010	→	011
01 + 11	→	000	→	100
10 + 00	→	011	→	010
10 + 01	→	010	→	011
10 + 10	→	001	→	100
10 + 11	→	000	→	101
11 + 00	→	010	→	011
11 + 01	→	000	→	100
11 + 10	→	000	→	101
11 + 11	→	000	→	110

@incollection{rumelhart:errorpropnonote, address = {Cambridge, MA}, author = {Rumelhart, David E. and Hinton, Geoffrey E. and Williams, Ronald J.}, booktitle = {Parallel Distributed Processing: Explorations in the Microstructure of Cognition, {V}olume 1: {F}oundations}, editor = {Rumelhart, David E. and Mcclelland, James L.}, pages = {318--362}, year = 1985, publisher = {MIT Press}, title = {Learning Internal Representations by Error Propagation},}

Elements of an Autoencoder

Autoencoder structure (mappings)



Composite map

$$\hat{\boldsymbol{y}} = G(F(\boldsymbol{y}, \boldsymbol{\alpha}), \boldsymbol{\beta})$$

The goal

The goal is to generate a reproduction $\hat{\boldsymbol{y}}$ of the input $\boldsymbol{y}$ that achieves a high degree of dimensionality reduction, i.e., $n < m$; while, maintaining the *significant information* from the input data.

Expressing the goal as a cost function

$$E(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \|\boldsymbol{r}\|_2 + \lambda\, R(\boldsymbol{s}, N(0,1))$$

The Concept of Low Rank Identity

Can the identity matrix I be factored?

Of course, since $B^{-1} B = I$. With special case $A^T A = I$.

Do low rank factorizations exist?

Yes, if we can accept some finite error ϵ.

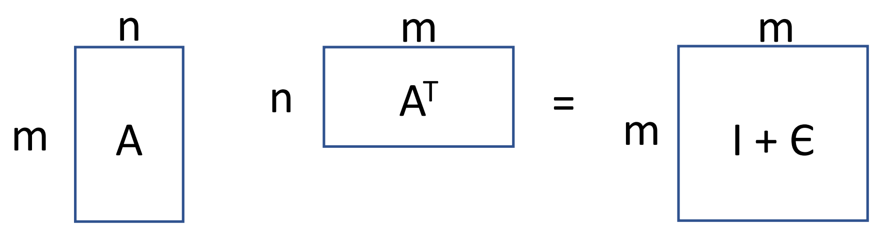


TABLE 5

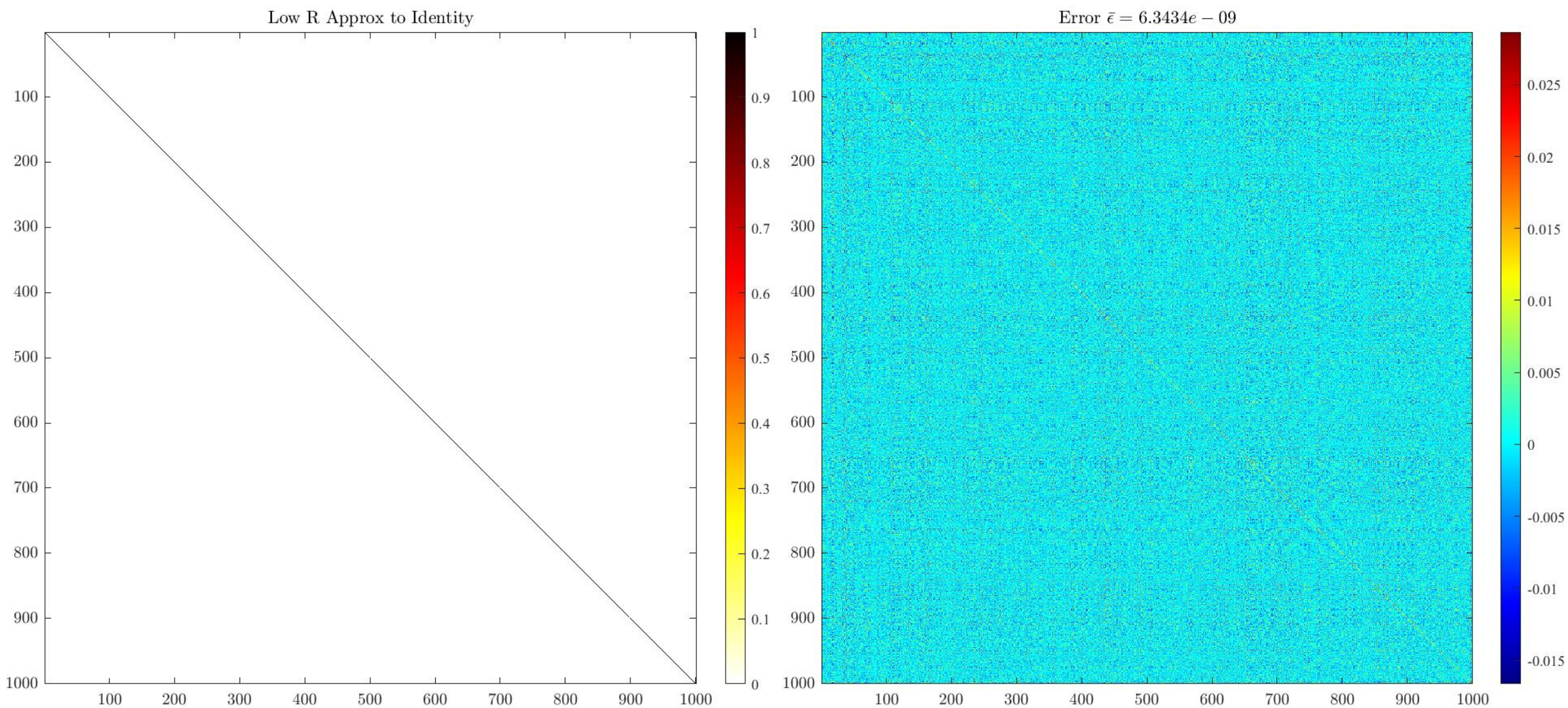
Input Patterns		Hidden Unit Patterns				Output Patterns
10000000	→	.5	0	0	→	10000000
01000000	→	0	1	0	→	01000000
00100000	→	1	1	0	→	00100000
00010000	→	1	1	1	→	00010000
00001000	→	0	1	1	→	00001000
00000100	→	.5	0	1	→	00000100
00000010	→	1	0	.5	→	00000010
00000001	→	0	0	.5	→	00000001

The most efficient low rank approximations are Sylvester equations of the form:

$$\boldsymbol{A}\,\boldsymbol{X} + \boldsymbol{X}\,\boldsymbol{B} = \boldsymbol{C}$$

Example Low Ran Approximation to Identity

Rank 10 deficint with $\in \approx 2.2e-9$



Low Rank Approximations are Nonunique

Original

Original

Structure (information) of the error is important, no only the magnitude.

Discrete bases functions can be used to define structure.

Both reconstructions are rank 10 deficient

Low rank approx (1)

Random bases

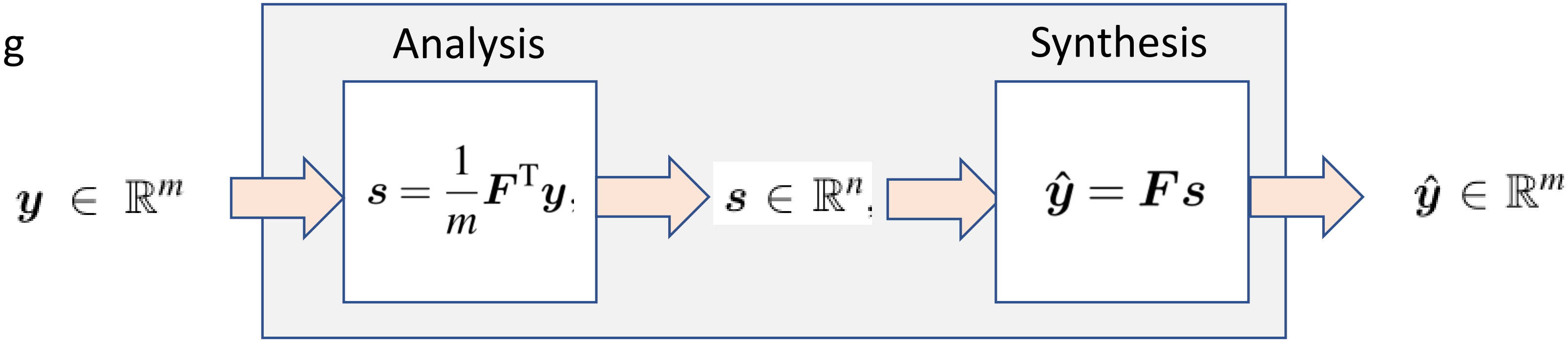
Low rank approx (2)

Polynomial bases

Paul.oleary@unileoben.ac.at

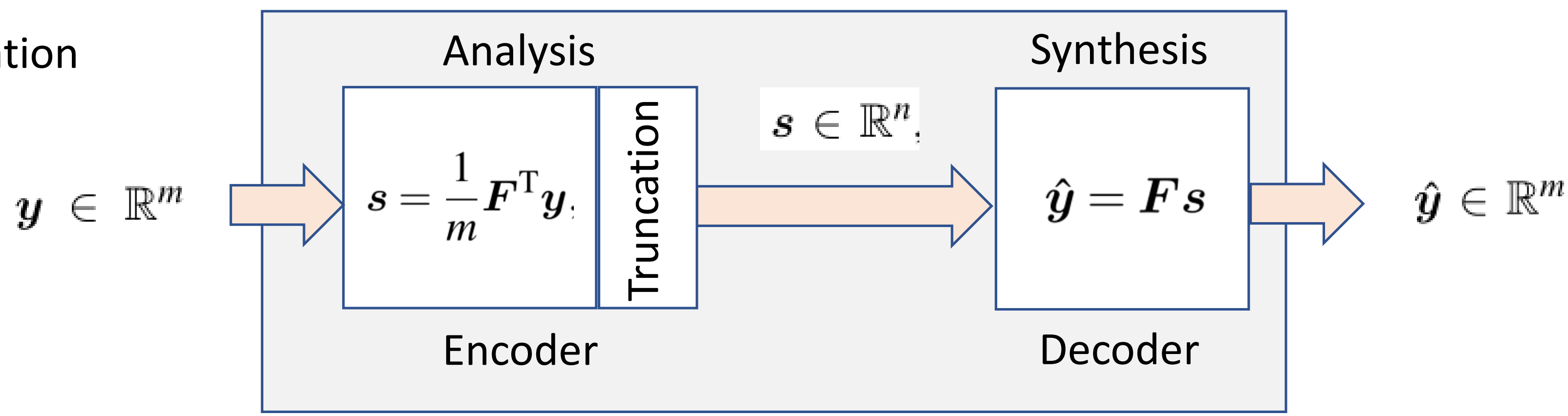
Fourier analysis in the context of autoencoders

Training



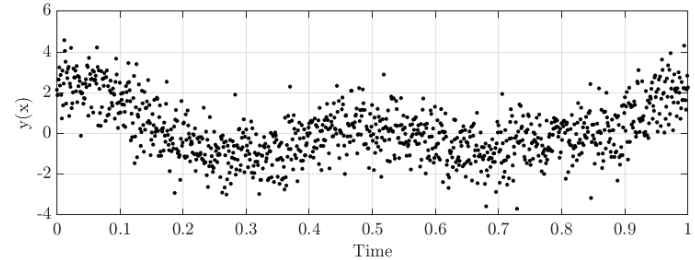
Fourier filtering using truncation (Dimensionality reduction)

Evaluation



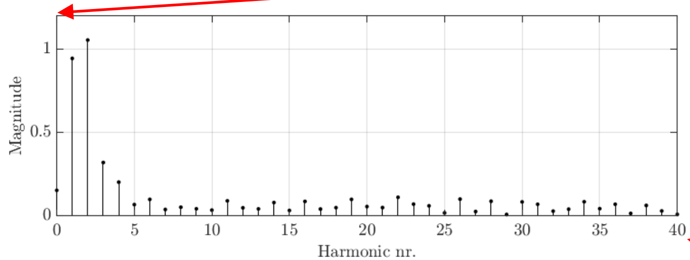
Example of Truncated Fourier Spectrum

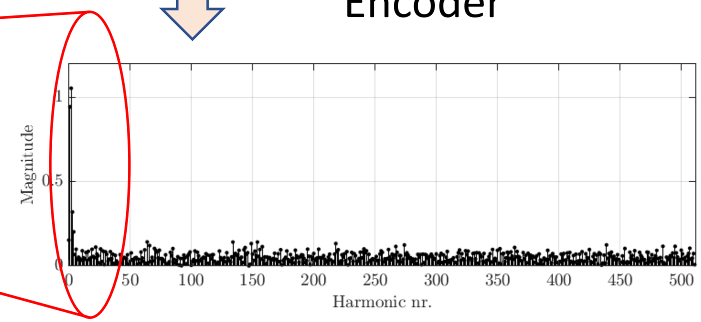
Raw data



Encoder

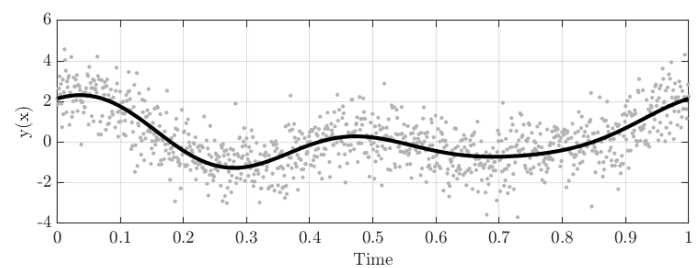
Spectrum





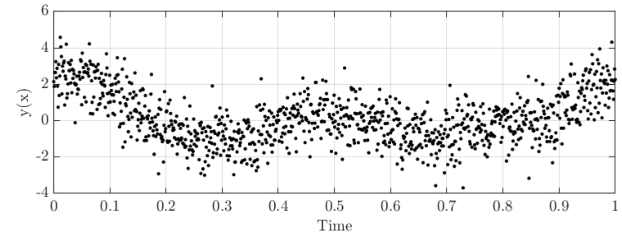
Decoder

Generation (truncated spectrum)



Low Rank Approximations with Bases

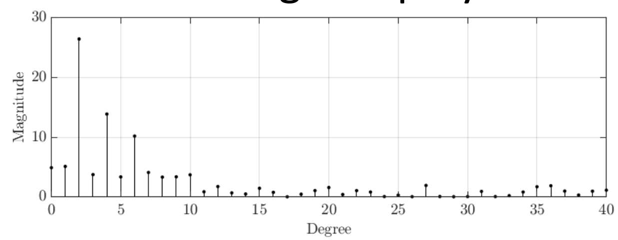
Raw data

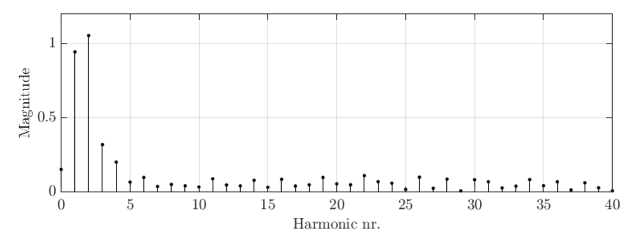


Discrete orthogonal polynomials

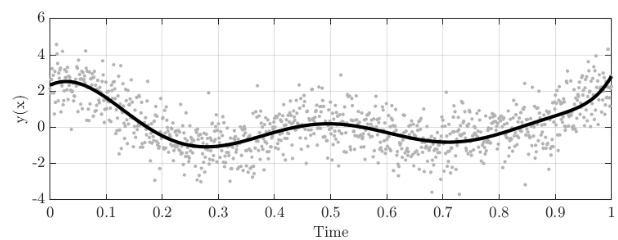
Fourier bases (DFT)

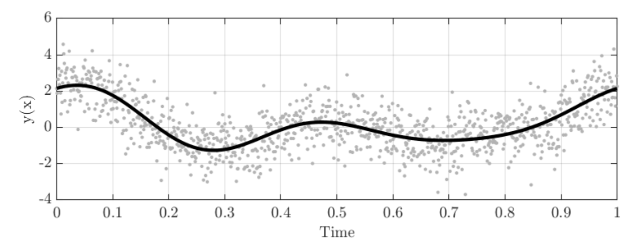
Spectrum wrt Bases



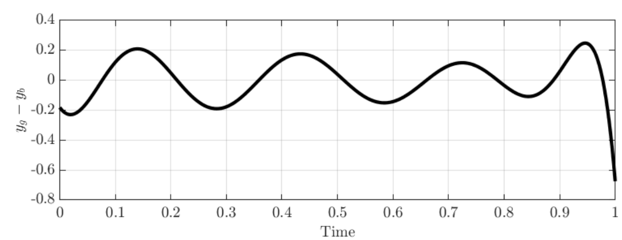


Low rank approximations



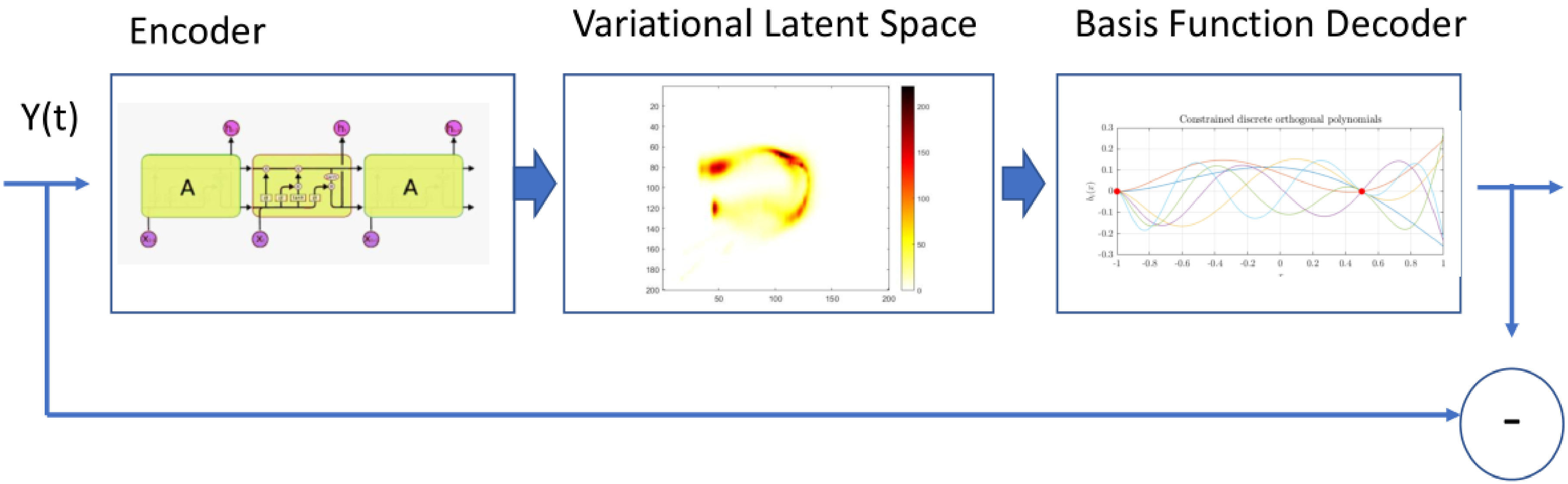


Model differences



Paul.oleary@unileoben.ac.at

New Proposed Architecture



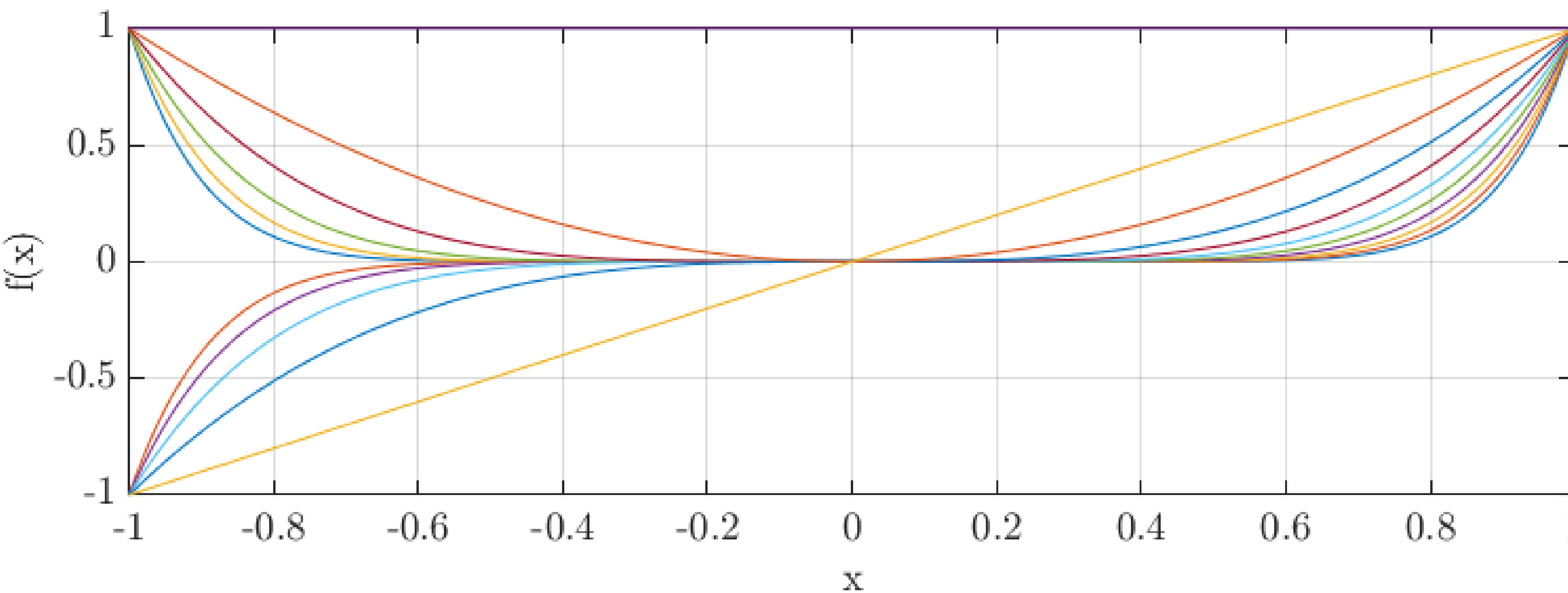
But also Y(x)

Error prorogation

$$\boldsymbol{y}_m = \boldsymbol{B}_c \, \boldsymbol{\alpha}, \tag{20}$$

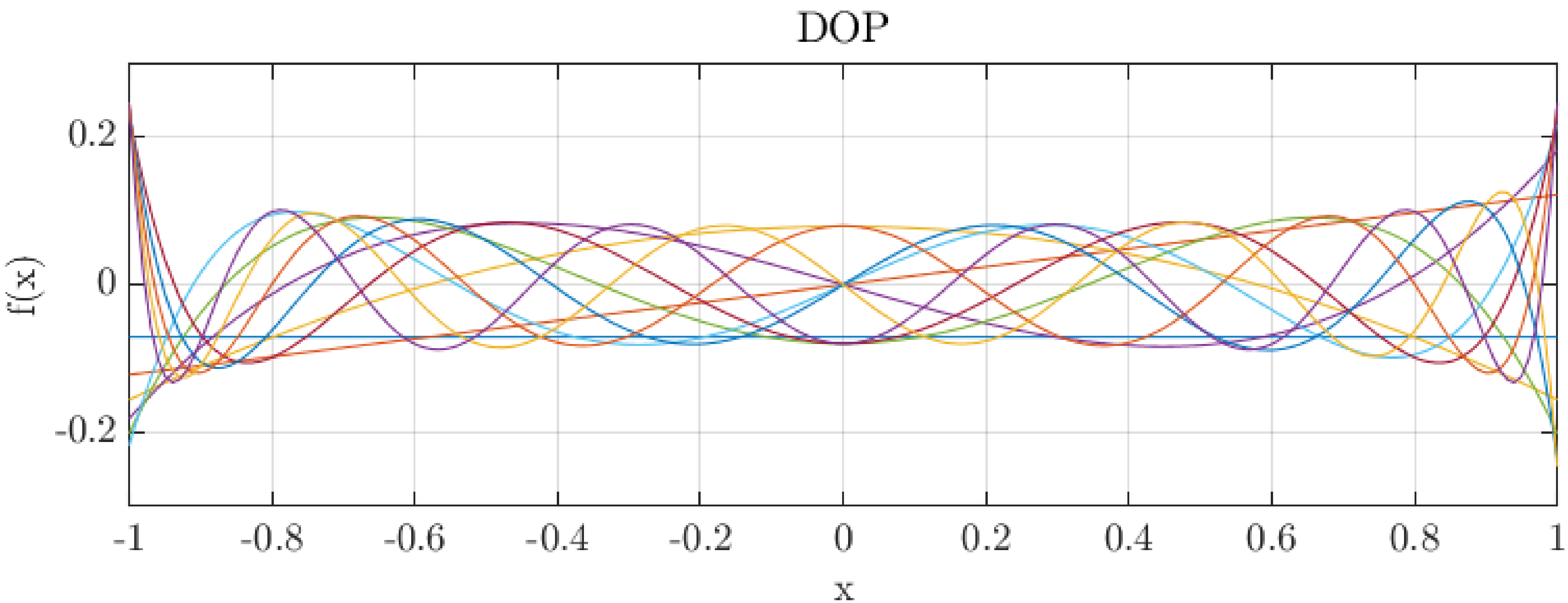
If $\boldsymbol{\Lambda}_\alpha$ is the covariance of $\boldsymbol{\alpha}$ (Variance of the latent space as approximation), then,

$$\boldsymbol{\Lambda}_{\boldsymbol{y}_m} = \boldsymbol{B}_c \, \boldsymbol{\Lambda}_\alpha \, \boldsymbol{B}_c^{\mathrm{T}}. \tag{21}$$

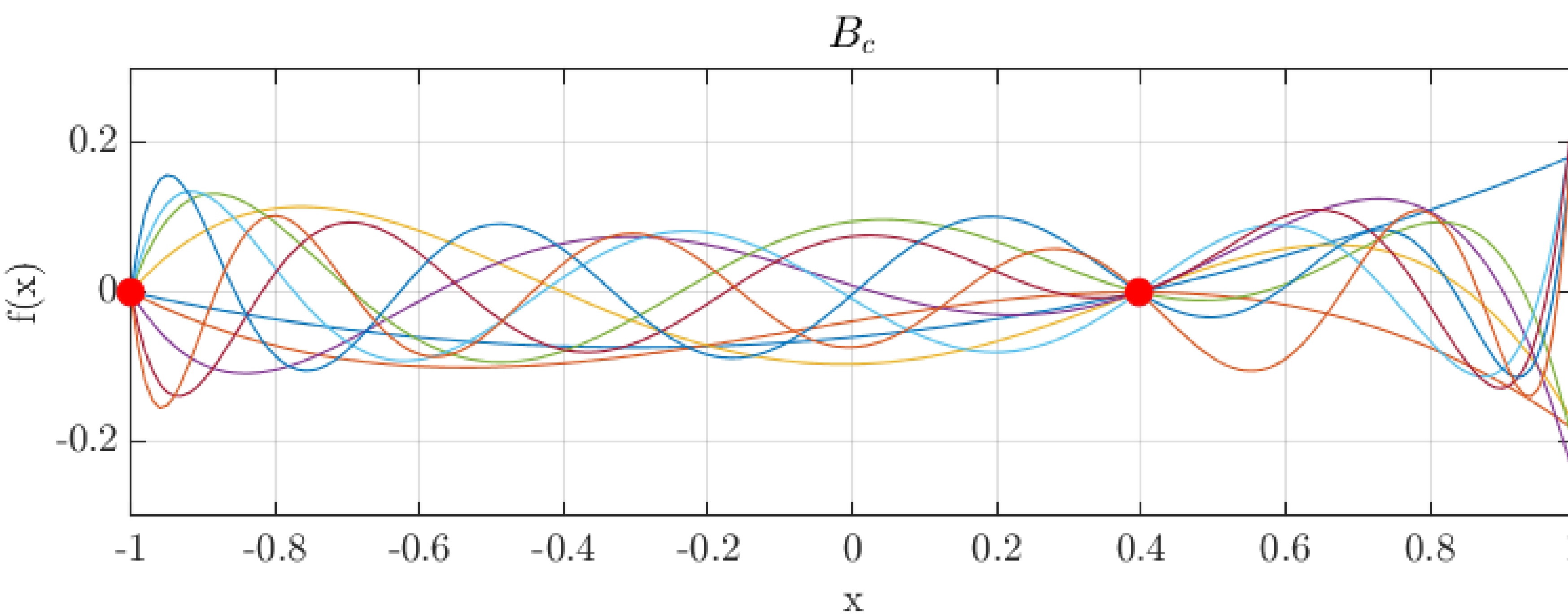


Paul.oleary@unileoben.ac.at

Discrete Orthogonal Polynomials



Constrained Discrete Orthogonal Polynomials



Example 2