Università
degli Studi
di Padova

Numerical Linear Algebra and Learning from Data

(LM in Mathematics)

Fabio Marcuzzi

September 25, 2024

Università degli studi di Padova

Motivation: Numerical Linear Algebra

- The course is focused upon **numerical methods for linear algebra** and its application in linear and nonlinear problems, plus an intro in **Fourier and time-frequency analysis of data**. Linear algebra is fundamental in numerical methods for data analysis in general, from the age of Gauss up to the most recent approaches in data science (e.g. "data assimilation", "machine learning", "system identification" etc.).

- The numerical linear algebra for data analysis cannot be done on paper: we will do a **computer lab** where implement solution algorithms to assigned problems and do adequate numerical experiments.

Motivation: Numerical Linear Algebra

- The course is focused upon **numerical methods for linear algebra** and its application in linear and nonlinear problems, plus an intro in **Fourier and time-frequency analysis of data**. Linear algebra is fundamental in numerical methods for data analysis in general, from the age of Gauss up to the most recent approaches in data science (e.g. "data assimilation", "machine learning", "system identification" etc.).

- The numerical linear algebra for data analysis cannot be done on paper: we will do a **computer lab** where implement solution algorithms to assigned problems and do adequate numerical experiments.

Motivation: Learning from Data

- The mathematical way to data analysis implies not only a statistical description and the computation of synthesis indicators, but the effort at building **interpretable models**, which can be stochastic and/or deterministic, but surely not black-box: even with neural networks, we will follow the **physics-informed scientific machine learning** approach; even with classic machine learning methods, like PCA/SVD, we will **discover underlying principles from data**. This usually involves various fields of mathematics, like linear algebra, analysis, geometry, probability, mathematical physics and, last but not least, numerical analysis.
- $\rightarrow$ take-home message from NLALD: Learning from Data is mathematical modelling!

Motivation: Learning from Data

- The mathematical way to data analysis implies not only a statistical description and the computation of synthesis indicators, but the effort at building **interpretable models**, which can be stochastic and/or deterministic, but surely not black-box: even with neural networks, we will follow the **physics-informed scientific machine learning** approach; even with classic machine learning methods, like PCA/SVD, we will **discover underlying principles from data**. This usually involves various fields of mathematics, like linear algebra, analysis, geometry, probability, mathematical physics and, last but not least, numerical analysis.
- $\rightarrow$ take-home message from NLALD: Learning from Data is mathematical modelling!

Course organization

For a more detailed presentation see:
https://www.math.unipd.it/ marcuzzi/Didattica.htm

- Two lectures (+ one lab session) per week; 8+(1) lab sessions totally.

- 5-6 homeworks with individual correction.

- Exam: written, three questions
  - one about the lab sessions (15 pts)
  - one about the theory (10 pts)
  - one exercise, akin to homeworks (5 pts)

Course organization

For a more detailed presentation see:
https://www.math.unipd.it/ marcuzzi/Didattica.htm

- Two lectures (+ one lab session) per week; 8+(1) lab sessions totally.

- 5-6 homeworks with individual correction.

- Exam: written, three questions
    - one about the lab sessions (15 pts)
    - one about the theory (10 pts)
    - one exercise, akin to homeworks (5 pts)

Course organization

For a more detailed presentation see:
https://www.math.unipd.it/ marcuzzi/Didattica.htm

- Two lectures (+ one lab session) per week; 8+(1) lab sessions totally.
- 5-6 homeworks with individual correction.
- Exam: written, three questions
  - one about the lab sessions (15 pts)
  - one about the theory (10 pts)
  - one exercise, akin to homeworks (5 pts)

Course organization

For a more detailed presentation see:
https://www.math.unipd.it/ marcuzzi/Didattica.htm

- Two lectures (+ one lab session) per week; 8+(1) lab sessions totally.
- 5-6 homeworks with individual correction.
- Exam: written, three questions
  - one about the lab sessions (15 pts)
  - one about the theory (10 pts)
  - one exercise, akin to homeworks (5 pts)

Course organization

For a more detailed presentation see:
https://www.math.unipd.it/ marcuzzi/Didattica.htm

- Two lectures (+ one lab session) per week; 8+(1) lab sessions totally.

- 5-6 homeworks with individual correction.

- Exam: written, three questions
  - one about the lab sessions (15 pts)
  - one about the theory (10 pts)
  - one exercise, akin to homeworks (5 pts)

proposal of Seminar Activity (4 CFU): virtual exchange

- NB: is a substitute of the MOOC "Scientific computing with Python, that **will not be anymore accepted, after March 1st 2025, as Seminar Activity**".

- From May 2025 there will be a new "virtual exchange" online course: **"Computation for Data Driven Discovery and Physics-Aware Soft-Sensors"**
- instructors:
  Assad Oberai (University of Southern California - USC)
  Fabio Marcuzzi (Universitá degli Studi di Padova - UniPD)
- 16 hrs lectures + collaborative project (online). Exam: oral poster presentation.
- Topics:
  - Physics-Informed Deep Operator Networks
  - Physics-based probabilistic inverse problems
  - Physics-Aware Soft-Sensors

proposal of Seminar Activity (4 CFU): virtual exchange

- NB: is a substitute of the MOOC "Scientific computing with Python, that **will not be anymore accepted, after March 1st 2025, as Seminar Activity**".

- From May 2025 there will be a new "virtual exchange" online course: **"Computation for Data Driven Discovery and Physics-Aware Soft-Sensors"**
- instructors:
  Assad Oberai (University of Southern California - USC)
  Fabio Marcuzzi (Universitá degli Studi di Padova - UniPD)
- 16 hrs lectures + collaborative project (online). Exam: oral poster presentation.
- Topics:
  - Physics-Informed Deep Operator Networks
  - Physics-based probabilistic inverse problems
  - Physics-Aware Soft-Sensors

proposal of Seminar Activity (4 CFU): virtual exchange

- NB: is a substitute of the MOOC "Scientific computing with Python, that **will not be anymore accepted, after March 1st 2025, as Seminar Activity**".

- From May 2025 there will be a new "virtual exchange" online course:
  "Computation for Data Driven Discovery and Physics-Aware Soft-Sensors"
- instructors:
  Assad Oberai (University of Southern California - USC)
  Fabio Marcuzzi (Universitá degli Studi di Padova - UniPD)
- 16 hrs lectures + collaborative project (online). Exam: oral poster presentation.
- Topics:
  - Physics-Informed Deep Operator Networks
  - Physics-based probabilistic inverse problems
  - Physics-Aware Soft-Sensors

proposal of Seminar Activity (4 CFU): virtual exchange

- NB: is a substitute of the MOOC "Scientific computing with Python, that **will not be anymore accepted, after March 1st 2025, as Seminar Activity**".

- From May 2025 there will be a new "virtual exchange" online course:
  "Computation for Data Driven Discovery and Physics-Aware Soft-Sensors"
- instructors:
  Assad Oberai (University of Southern California - USC)
  Fabio Marcuzzi (Universitá degli Studi di Padova - UniPD)
- 16 hrs lectures + collaborative project (online). Exam: oral poster presentation.
- Topics:
  - Physics-Informed Deep Operator Networks
  - Physics-based probabilistic inverse problems
  - Physics-Aware Soft-Sensors

proposal of Seminar Activity (4 CFU): virtual exchange

- NB: is a substitute of the MOOC "Scientific computing with Python, that **will not be anymore accepted, after March 1st 2025, as Seminar Activity**".

- From May 2025 there will be a new "virtual exchange" online course:
  "Computation for Data Driven Discovery and Physics-Aware Soft-Sensors"
- instructors:
  Assad Oberai (University of Southern California - USC)
  Fabio Marcuzzi (Universitá degli Studi di Padova - UniPD)
- 16 hrs lectures + collaborative project (online). Exam: oral poster presentation.
- Topics:
  - Physics-Informed Deep Operator Networks
  - Physics-based probabilistic inverse problems
  - Physics-Aware Soft-Sensors

Topics

Matrix factorizations

- Data are usually organized in matrices, or even tensors (i.e. multi-dimensional arrays).
- Identifying the underlying spatiotemporal coherent structures of a data set and extracting meaningful information is a key problem in data analysis; the results are usually obtained as low-rank matrices.
- Matrix factorizations are the fundamental mathematical tool. There are *exact* and *approximate* matrix factorizations:
  - exact: QR, SVD, NMF, etc
  - approximate, i.e. Low-Rank Matrix Approximations (LRMA): TSVD, NMF

Matrix factorizations

- Data are usually organized in matrices, or even tensors (i.e. multi-dimensional arrays).
- Identifying the underlying spatiotemporal coherent structures of a data set and extracting meaningful information is a key problem in data analysis; the results are usually obtained as low-rank matrices.
- Matrix factorizations are the fundamental mathematical tool. There are *exact* and *approximate* matrix factorizations:
  - exact: QR, SVD, NMF, etc
  - approximate, i.e. Low-Rank Matrix Approximations (LRMA): TSVD, NMF

Matrix factorizations

- Data are usually organized in matrices, or even tensors (i.e. multi-dimensional arrays).
- Identifying the underlying spatiotemporal coherent structures of a data set and extracting meaningful information is a key problem in data analysis; the results are usually obtained as low-rank matrices.
- Matrix factorizations are the fundamental mathematical tool. There are *exact* and *approximate* matrix factorizations:
  - exact: QR, SVD, NMF, etc
  - approximate, i.e. Low-Rank Matrix Approximations (LRMA): TSVD, NMF

Matrix factorizations

- Data are usually organized in matrices, or even tensors (i.e. multi-dimensional arrays).
- Identifying the underlying spatiotemporal coherent structures of a data set and extracting meaningful information is a key problem in data analysis; the results are usually obtained as low-rank matrices.
- Matrix factorizations are the fundamental mathematical tool. There are *exact* and *approximate* matrix factorizations:
  - exact: QR, SVD, NMF, etc
  - approximate, i.e. Low-Rank Matrix Approximations (LRMA): TSVD, NMF

Matrix factorizations

- Data are usually organized in matrices, or even tensors (i.e. multi-dimensional arrays).
- Identifying the underlying spatiotemporal coherent structures of a data set and extracting meaningful information is a key problem in data analysis; the results are usually obtained as low-rank matrices.
- Matrix factorizations are the fundamental mathematical tool. There are *exact* and *approximate* matrix factorizations:
  - exact: QR, SVD, NMF, etc
  - approximate, i.e. Low-Rank Matrix Approximations (LRMA): TSVD, NMF

Exact matrix factorizations: QR and SVD

We will recall and see the numerical analysis of QR and SVD **(LAB)**

- recall: Gram-Schmidt orthogonalization, Householder and Givens transformations;
- updating/downdating a QR factorization;
- QR with pivoting for flat matrices and rank-revealing QR;

Exact matrix factorizations: CMR, Exact NMF

We will see the CMR and NMF matrix factorizations and applications **(LAB)**

Columns-M-Rows (CMR) factorizations: $A = CMR$, where C are the r independent columns of the r-rank matrix A, R are r rows of A and M is a recombination matrix; they use columns and rows of the original matrix, to the advantage of interpretation.

Nonnegative Matrix Factorizations (NMF): $A = WH$, where W has r columns and H has r rows, require the factors of the low-rank approximation to be componentwise nonnegative:

- this makes it possible to interpret them meaningfully, e.g. when they correspond to nonnegative physical/abstract quantities;
- *nonnegative rank*: the smallest r such that an Exact NMF exists.

Approximate matrix factorizations: TSVD

- Eckart-Young theorem: the sum of the first k singular components of A, that are rank-1 matrices, gives the best approximation of A using rank-k matrices.

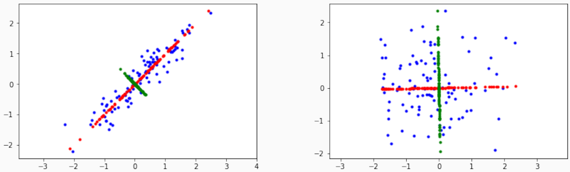
Approximate matrix factorizations: Approximate NMF

The NMF with additional constraints, regularizations and different objective functions, creates a variety of different Approximate NMF models.

- uniqueness (identifiability) and computational complexity are big issues.

Linear Dimensionality Reduction: PCA

Linear Dimensionality Reduction (LDR) techniques are mainly Low-Rank Matrix Approximations (LRMA). A remarkable example is Principal Component Analysis (PCA):



- very often, it is implemented as a TSVD.

Linear Dimensionality Reduction: Approximate NMF

Linear Dimensionality Reduction (LDR) techniques represent each data point as a linear combination of a small number of basis elements. With an Approximate NMF, each column of the matrix W is a basis element. Therefore, the dimensionality reduction is tied with the choice of r.

- approximations are often made here to reduce the computational complexity and to get uniqueness;
- Lee and Seung (Nature, 1999) popularized NMF in a variety of applications of learning from data.

Linear Dimensionality Reduction: Subspace System Identification

Linear Dimensionality Reduction (LDR) within the DLTI model class. Discrete-time Linear Time-Invariant (DLTI) dynamical systems play a fundamental role in data-related applications.

Three representations for the same class of models: **discrete convolution**, **finite differences (ARMA)**, **state-space**.

Auto-regressive moving-average (ARMA) models:

- general solution
- Hankel matrices: realizability theorem
- time-series analysis **(LAB)**

Algebraic properties of DLTI systems in state-space form:

- free and forced response
- Reachability and Observability

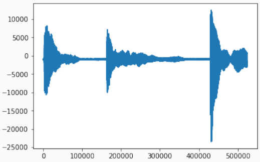
The “”minimal realization problem“ and its connection with linear dimensionality reduction techniques mentioned above”:

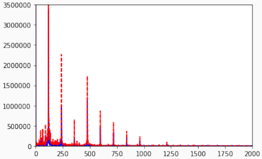
- Kalman’s Theorem on minimal realizations
- algorithms: Ho-Kalman, ERA, subspace methods

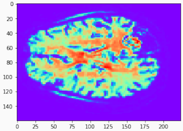
Special Matrices (Linear Transforms)

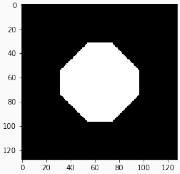
Fourier analysis of data plays a central role in applications. **(LAB)**

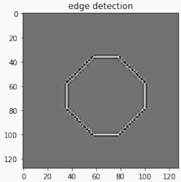
- Discrete Fourier Transform (DFT)
- The Frequency Response of a DLTI dynamical system
- algorithm: the Fast Fourier Transform (FFT), da $O(n^2)$ a $nlog(n)$ operazioni !
- Short-Time Fourier Transform (STFT)
- Wavelets Transforms











Linear parameter estimation

- ordinary least-squares, recursive least-squares **(LAB)**
- total least-squares (TLS)
- non-negative least-squares: Lawson-Hanson algorithm

Linear parameter estimation

- ordinary least-squares, recursive least-squares **(LAB)**
- total least-squares (TLS)
- non-negative least-squares: Lawson-Hanson algorithm

Linear parameter estimation

- ordinary least-squares, recursive least-squares **(LAB)**
- total least-squares (TLS)
- non-negative least-squares: Lawson-Hanson algorithm

Inverse problems, ill-posedness, regularization

- l_2-norm regularization: TSVD and Tichonov
- l_1-norm regularization: basis pursuit; "sparse recovery" for underdetermined linear systems, mixed l_1-l_2-norm regularization.

Inverse problems, ill-posedness, regularization

- l_2-norm regularization: TSVD and Tichonov

- l_1-norm regularization: basis pursuit; "sparse recovery" for underdetermined linear systems, mixed l_1-l_2-norm regularization.

Nonlinear parameter estimation

From linearity to **convexity**: Nonlinear least squares:

- Gauss-Newton
- Levenberg-Marquardt

Computation of the gradient:

- finite differences,
- adjoint method,
- stochastic gradient.

Nonlinear parameter estimation

From linearity to **convexity**: Nonlinear least squares:

- Gauss-Newton
- Levenberg-Marquardt

Computation of the gradient:

- finite differences,
- adjoint method,
- stochastic gradient.

Nonlinear parameter estimation

From linearity to **convexity**: Nonlinear least squares:

- Gauss-Newton
- Levenberg-Marquardt

Computation of the gradient:

- finite differences,
- adjoint method,
- stochastic gradient.

Nonlinear parameter estimation

From linearity to **convexity**: Nonlinear least squares:

- Gauss-Newton
- Levenberg-Marquardt

Computation of the gradient:

- finite differences,
- adjoint method,
- stochastic gradient.

Nonlinear parameter estimation

From linearity to **convexity**: Nonlinear least squares:

- Gauss-Newton
- Levenberg-Marquardt

Computation of the gradient:

- finite differences,
- adjoint method,
- stochastic gradient.

Nonlinear parameter estimation

From linearity to **convexity**: Nonlinear least squares:

- Gauss-Newton
- Levenberg-Marquardt

Computation of the gradient:

- finite differences,
- adjoint method,
- stochastic gradient.

Neural Networks Learning

We will study the numerical algorithms for Deep Learning **(LAB)**

- Back-propagation
- Momentum
- Stochastic Gradient
- avoiding saddle-points
- convolutional layers
- regularization

$\Rightarrow$ with a focus on *scientific machine learning*!

State estimation

- Kalman filtering
- Variational Data Assimilation (VDA)

Computational inverse problems

We see applications from systems describable with a mathematical-physical model governed by

ordinary differential equations (ODEs)

- mass-spring-damper system (nonlinear parameters estimation problem) **(LAB)**

partial differential equations (PDEs)

- hidden corrosion detection (inverse heat problem) **(LAB)**
- contact force reconstruction
- anomaly detection in mechanical vibrations

$\Rightarrow$ with a focus on *physics-aware soft sensors*!