Machine Learning for Time Series

Introduction

Laurent Oudre
laurent.oudre@ens-paris-saclay.fr

Master MVA
2024-2025

Contents

Organization of the course : lectures

- Six lectures on Thursday afternoons (except next week : course in the morning !) at ENS Paris Saclay
- Lectures will be in French but all material (slides, homeworks...) is in English.
- Attendance is mandatory
- Registration deadline for the mailing list : **October 15th, 2024**

  https://forms.gle/KECsUsdYU7ZFtLNT9

- ML for time series is part of the Modelling track !

Teaching material: http://www.laurentoudre.fr/ast.html

Organization of the course : tutorials

- For the tutorials there are two options :
  - Thursday morning : remote on Zoom
  - Thursday afternoon : onsite at ENS Paris Saclay
- Extra work for each tutorial : approximately 6 hours
- Attendance is mandatory
- Tutorial homeworks are **mandatory**

missing or late homeworks → fail the class

Teaching assistant: Charles Truong (ctruong@ens-paris-saclay.fr)

Validation

Validation: tutorials (25%) + mini-projects (25% report, 25% source code and 25% oral presentation)

- Projects can be done in groups of two, but no more than that
- Students are allowed to propose additional project : ask in advance !
- The mini project consists in reading a research paper, implement it in Python and launch experiments on real time series
- Report (PDF file, ≈ 5 pages) + source code (Jupyter Notebook). **Deadlines : December 18th (23:59) or January 9th (23:59)**
- A 10 min oral presentation is scheduled on **December, 19th and 20th and January, 9th and 10th**, which will finalize the course project
- **Due to the large number of students, *auditeurs libres* will not be able to validate the course (no grading for tutorials, no mini-project).**

Contents

What is a time series?

- A time series is a series of data points indexed in time order
- In practice, array of real numbers of size $D \times N$ where D is the number of dimensions and N the number of samples
  - Sample number n

n	0	1	2	3	4	5	6

  - Time series values $x[n]$

$x[n]$	0.7	0.2	0.8	0.9	0.3	0.2	0.7
	0.4	0.1	0.6	0.2	0.5	0.6	0.3

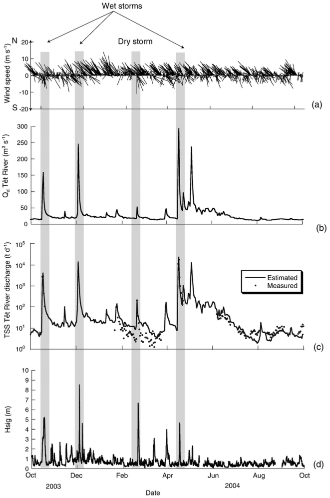
  - Timestamps $t[n]$

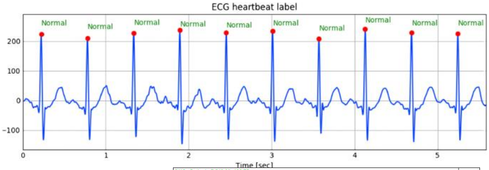
$t[n]$	16:30:01	16:30:23	16:31:43	16:32:38	16:33:06	16:33:16	16:33:56

An un-unified field

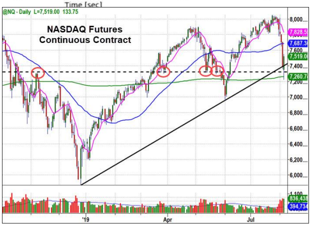
- Different scientific communities have given different names to the same mathematical object.
    - **Time series**: mathematics, statistics, economics, finance...
    - **Signals**: signal processing, physics, engineering, simulation...
    - **Sequences**: computer sciences, bioinformatics, data mining...
- In this course, we will use indifferently one of these terms.
- Typical definition: real-valued (or at least ordered) sequential data

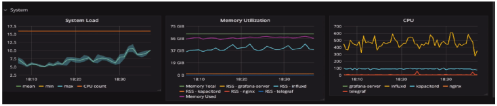
Time series are everywhere





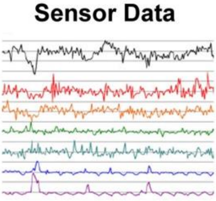
Meteorology,
Finance,
Healthcare,
Monitoring,
Epidemiology,
Sensor networks...

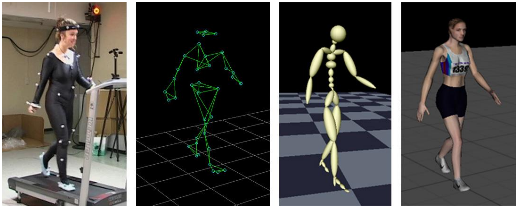


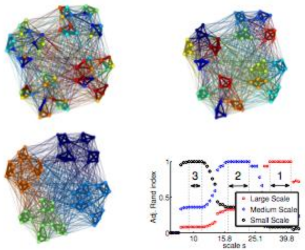


Univariate vs. multivariate

2D/3D trajectories,
Multivariate time series,
Multimodal data from
sensor networks,
Graph signals







Time series are complex

- Potentially massive data (e.g. sound : sampling frequency 44.1 kHz)
- Multivariate, multimodal, heterogeneous
- Noisy, missing data, trends, mixture of sources
- Often linked to an application context: data scientist is not trained to understand the data

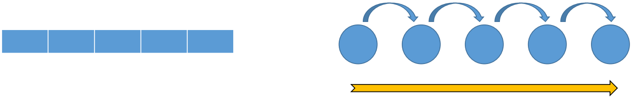
Annotations and ground truth

- Contrary to basic image processing tasks (e.g. classification of cats and dogs), annotating time series often require expertise
- Typical context:
  - Noisy and dirty data
  - A few annotated signals with blurry labels (confusing and hyper-specialized annotations that cannot be transformed into class labels)
  - An expert with several years in the business, but unable to translate it into ML-compatible annotations

How to use ML in this context?

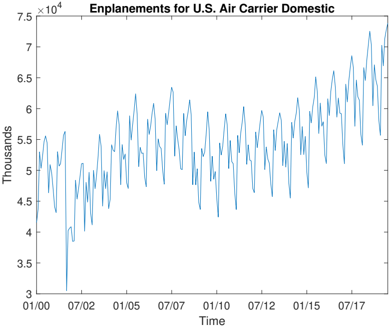
What about time?

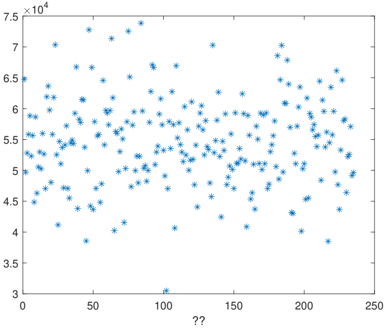
- What is the difference between regular data and time series ? Notion of sequence and chronology



- Each sample corresponds to the measurement of a phenomenon at a given time stamp.
- Time allows to study the evolution of the phenomenon and should be taken into account for processing the data

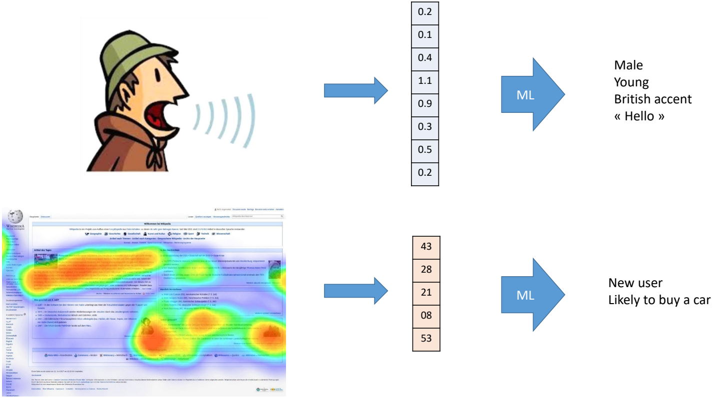
What about time?





Same time series... but mixed up times

World vs. Machine Learning



- Most ML algorithms do not care for time.
- How can we still use the time information to extract relevant features/patterns that can be used within a ML procedure ?

Two visions: physics vs. statistics

- The notion of time have been used and modeled in physics since 18th century and before (eg. Fourier transform).
  First vision : a time series $x[1:N]$ is the result of the digitization of a physical phenomenon $x(t)$. Physical properties of this phenomenon can be retrieved and analyzed through the study of $x[1:N]$ (and vice/versa).
- Randomness can also play a part to model a wider class of signals.
  Second vision : a time series $x[1:N]$ is a realization of a stochastic process $X[1:N]$. Statistical properties of this phenomenon can be retrieved and analyzed through the study of $x[1:N]$ (and vice/versa).

In most cases, both approaches can be combined.

Deep learning: the optimal solution?

Deep learning achieves state-of-the-art results for several tasks **BUT**...

- Good performances $\neq$ good understanding of the data (cf next slide)
- DL is a black box that may not bring satisfaction to users on the field since they cannot interpret the results
- Although some networks are able to handle time (e.g. LSTM), they still only manage at most a few hundred time samples
- DL is bad in the context of scarce data and annotations

Deep learning: the optimal solution?

To validate our claim, we introduce a set of embarrassingly simple one-layer linear models named LTSF-Linear *for comparison. Experimental results on nine real-life datasets show that* LTSF-Linear *surprisingly outperforms existing sophisticated Transformer-based LTSF models in all cases, and often by a large margin. Moreover, we con-*

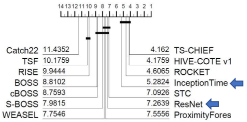
Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2022). Are Transformers Effective for Time Series Forecasting?. *arXiv preprint arXiv:2205.13504*.

Forecasting

Dataset	K-means [37]	UDFS [6]	NDFS [7]	RUFS [8]	RSFS [9]	KSC [22]	KDBA [20]	k-shape [5]	u-shapelet [24]	DTC [25]	USSL [29]	DEC [26]	IDEC [27]	DTCR
Arrow	0.6905	0.7254	0.7381	**0.7476**	0.7108	0.7254	0.7222	0.7254	0.6460	0.6692	0.7159	0.5817	0.6210	0.6868(0.0026)
Beef	0.6713	0.6759	0.7034	0.7149	0.6975	0.7057	0.6713	0.5402	0.6966	0.6345	0.6966	0.5954	0.6276	**0.8046**(0.0018)
BeetleFly	0.4789	0.4949	0.5579	0.6053	0.6516	0.6053	0.6052	0.6053	0.7314	0.5211	0.8105	0.4947	0.6053	**0.9000**(0.0001)
BirdChicken	0.4947	0.4947	0.7316	0.5579	0.6632	0.7316	0.6053	0.6632	0.5579	0.4947	**0.8105**	0.4737	0.4789	**0.8105**(0.0033)
Car	0.6345	0.6757	0.6260	0.6667	0.6708	0.6898	0.6254	0.7028	0.6418	0.6695	0.7345	0.6859	0.6870	**0.7501**(0.0022)
chlorineConcentration	0.5241	0.5282	0.5225	0.5330	0.5316	0.5256	0.5300	0.4111	0.5318	0.5353	0.4997	0.5348	0.5350	**0.5357**(0.0011)
coffee	0.7460	0.8624	**1.0000**	0.5476	**1.0000**	**1.0000**	0.4851	**1.0000**	**1.0000**	0.4841	**1.0000**	0.4921	0.5767	0.9286(0.0016)
diatomsizeReduction	0.9583	0.9583	0.9583	0.9333	0.9137	**1.0000**	0.9583	**1.0000**	0.7083	0.8792	**1.0000**	0.9294	0.7347	0.9682(0.0032)
dist.phal.outl.agegroup	0.6171	0.6531	0.6239	0.6252	0.6539	0.6535	0.6750	0.6020	0.6273	0.7812	0.6650	0.7785	0.7786	**0.7825**(0.0008)
dist.phal.outl.correct	0.5252	0.5362	0.5362	0.5252	0.5327	0.5235	0.5203	0.5252	0.5098	0.5010	0.5962	0.5029	0.5330	**0.6075**(0.0024)
ECG200	0.6315	0.6533	0.6315	0.7018	0.6916	0.6315	0.6018	0.7018	0.5758	0.6018	**0.7285**	0.6422	0.6233	0.6648(0.0034)
ECGFiveDays	0.4783	0.5020	0.5573	0.5020	0.5953	0.5257	0.5573	0.5020	0.5968	0.5016	0.8340	0.5103	0.5114	**0.9638**(0.0032)
GunPoint	0.4971	0.5029	0.5102	0.6498	0.4994	0.4971	0.5420	0.6278	0.6278	0.5400	**0.7257**	0.4981	0.4974	0.6398(0.0011)
Ham	0.5025	0.5219	0.5362	0.5107	0.5127	0.5362	0.5141	0.5311	0.5362	0.5648	**0.6393**	0.5963	0.4956	0.5362(0.0035)
Herring	0.4965	0.5099	0.5164	0.5238	0.5151	0.4940	0.5164	0.4965	0.5417	0.5045	**0.6190**	0.5099	0.5099	0.5759(0.0017)
Lighting2	0.4966	0.5119	0.5373	0.5729	0.5269	0.6263	0.5119	0.6548	0.5192	0.5770	**0.6955**	0.5311	0.5519	0.5913(0.0016)
Meat	0.6595	0.6483	0.6635	0.6578	0.6657	0.6723	0.6816	0.6575	0.6742	0.3220	0.7740	0.6475	0.6220	**0.9763**(0.0016)
Mid.phal.outl.agegroup	0.5351	0.5269	0.5350	0.5315	0.5473	0.5364	0.5513	0.5105	0.5396	0.5757	0.5807	0.7059	0.6800	**0.7982**(0.0028)
Mid.phal.outl.correct	0.5000	0.5431	0.5047	0.5114	0.5149	0.5014	0.5563	0.5114	0.5218	0.5272	**0.6635**	0.5423	0.5423	0.5617(0.0006)
Mid.phal.TW	0.0983	0.1225	0.1919	0.7920	0.8062	0.8187	0.8046	0.6213	0.7920	0.7115	0.7920	0.8590	0.8626	**0.8638**(0.0007)
MoteStrain	0.4947	0.5579	0.6053	0.5579	0.6168	0.6632	0.4789	0.6053	0.4789	0.5062	**0.8105**	0.7435	0.7324	0.7686(0.0036)
OSULeaf	0.5615	0.5372	0.5622	0.5497	0.5665	0.5714	0.5541	0.5538	0.5525	0.7329	0.6551	0.7484	0.7607	**0.7739**(0.0014)
Plane	0.9081	0.8949	0.8954	0.9220	0.9314	0.9603	0.9225	0.9901	**1.0000**	0.9040	**1.0000**	0.9447	0.9447	0.9549(0.0037)
Prox.phal.outl.ageGroup	0.5288	0.4997	0.5463	0.5780	0.5384	0.5305	0.5192	0.5617	0.5206	0.7430	0.7939	0.4263	**0.8091**	**0.8091**(0.0038)
Prox.phal.TW	0.4789	0.4947	0.6053	0.5579	0.5211	0.6053	0.5211	0.5211	0.4789	0.8380	0.7282	0.8189	**0.9030**	0.9023(0.0023)
SonyAIBORobotSurface	0.7721	0.7695	0.7721	0.7787	0.7928	0.7726	0.7988	0.8084	0.7639	0.5563	0.8105	0.5732	0.6900	**0.8769**(0.0033)
SonyAIBORobotSurfaceII	0.8697	0.8745	0.8865	0.8756	0.8948	**0.9039**	0.8684	0.5617	0.8770	0.7012	0.8575	0.6514	0.6572	0.8354(0.0016)
SwedishLeaf	0.4987	0.4923	0.5500	0.5192	0.5038	0.4923	0.5500	0.5333	0.6154	0.8871	0.8547	0.8837	0.8893	**0.9223**(0.0021)
Symbols	0.8810	0.8548	0.8562	0.8525	0.9060	0.8982	**0.9774**	0.8373	0.9603	0.9053	0.9200	0.8841	0.8857	0.9168(0.0022)
ToeSegmentation1	0.4873	0.4921	0.5873	0.5429	0.4968	0.5000	0.6143	0.6143	0.5873	0.5077	**0.6718**	0.4984	0.5017	0.5659(0.0006)
ToeSegmentation2	0.5257	0.5257	0.5968	0.5968	0.5826	0.5257	0.5573	0.5257	0.5020	0.5348	0.6778	0.4991	0.4991	**0.8286**(0.0028)
TwoPatterns	0.8529	0.8259	0.8530	0.8385	**0.8588**	0.8585	0.8446	0.8046	0.7757	0.6251	0.8318	0.6293	0.6338	0.6984(0.0025)
TwoLeadECG	0.5476	0.5495	0.6328	0.8246	0.5635	0.5464	0.5476	0.8246	0.5404	0.5116	**0.8628**	0.5007	0.5016	0.7114(0.0014)
wafer	0.4925	0.4925	0.5263	0.5263	0.4925	0.4925	0.4925	0.4925	0.4925	0.5324	**0.8246**	0.5679	0.5597	0.7338(0.0006)
Wine	0.4984	0.4987	0.5123	0.5021	0.5033	0.5006	0.5064	0.5001	0.5033	0.4906	**0.8985**	0.4913	0.5157	0.6271(0.0039)
WordsSynonyms	0.8775	0.8697	0.8760	0.8861	0.8817	0.8727	0.8159	0.7844	0.8230	0.8855	0.8540	0.8893	0.8947	**0.8984**(0.0003)
AVG Rank	10.6667	9.6806	7.2222	7.3889	6.8750	7.1389	7.9167	8.2361	8.2500	8.8194	3.5000	8.6528	7.5833	**3.0694**
AVG RI	0.5975	0.6077	0.6402	0.6478	0.6542	0.6582	0.6335	0.6419	0.6402	0.6238	0.7676	0.6351	0.6515	**0.7714**
Best	0	0	1	1	1	3	2	0	1	0	12	0	1	**17**
p-value	2.089E-6	4.8823E-6	3.4131E-5	5.7729E-5	4.1222E-5	1.3545E-4	1.2565E-5	1.4814E-4	3.4141E-5	3.0287E-7	9.7386E-1	8.7697E-07	3.2916E-7	-

Ma, Q., Zheng, J., Li, S., & Cottrell, G. W. (2019). Learning representations for time series clustering. Advances in neural information processing systems, 32.

Clustering



https://www.timeseriesclassification.com/results.php
(June 2020)

Classification

In line with related work [67], we found that deep learning approaches are not (yet) competitive despite their higher processing effort on training data. We could also confirm that *"simple methods yield performance almost as good as more sophisticated methods"* [56]. Still, no single algorithm clearly performs best. We highlighted sev-

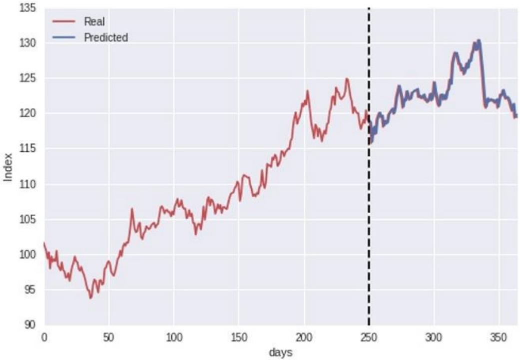
Schmidl, S., Wenig, P., & Papenbrock, T. (2022). Anomaly detection in time series: a comprehensive evaluation. *Proceedings of the VLDB Endowment*, *15*(9), 1779-1797.

Anomaly detection

= deep learning approach

How NOT to use DL for time series (1/4)

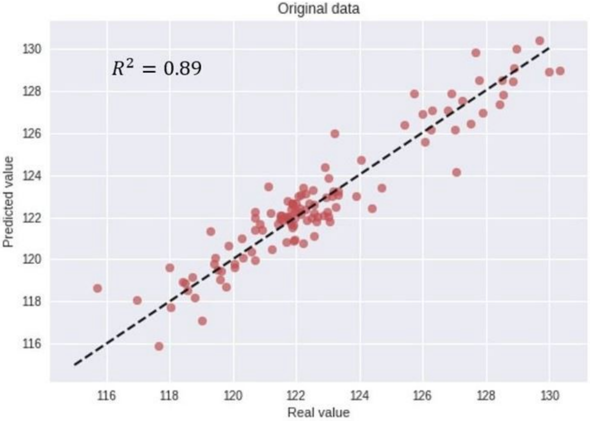
Blog post from V. Flovik
https://towardsdatascience.com/how-not-to-use-machine-learning-for-time-series-forecasting-avoiding-the-pitfalls-19f9d7adf424



Prediction of stock index with LSTM network: use the first 250 days as training data. Prediction seems great ‼

How NOT to use DL for time series (2/4)

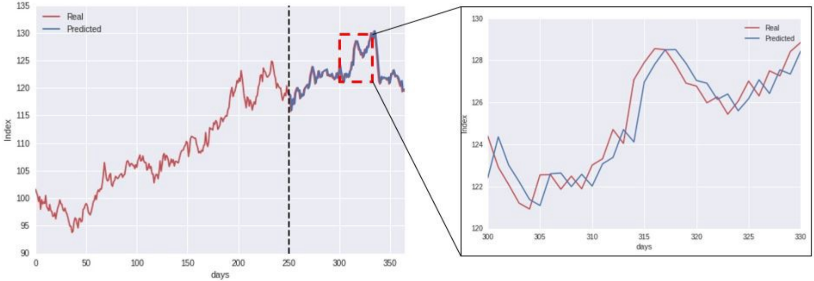
Blog post from V. Flovik
https://towardsdatascience.com/how-not-to-use-machine-learning-for-time-series-forecasting-avoiding-the-pitfalls-19f9d7adf424



Great performances too! Accurate prediction and RMSE !

How NOT to use DL for time series (3/4)

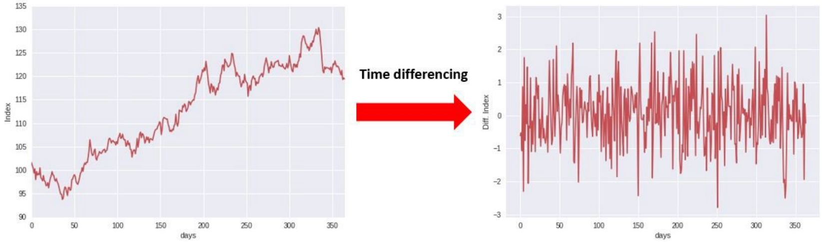
Blog post from V. Flovik
https://towardsdatascience.com/how-not-to-use-machine-learning-for-time-series-forecasting-avoiding-the-pitfalls-19f9d7adf424



In fact, LSTM was just repeating the previous sample...

How NOT to use DL for time series (4/4)

Blog post from V. Flovik
https://towardsdatascience.com/how-not-to-use-machine-learning-for-time-series-forecasting-avoiding-the-pitfalls-19f9d7adf424



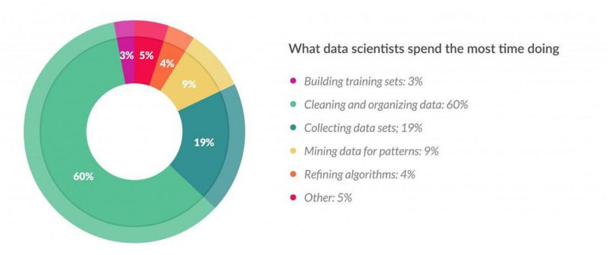
In fact, the data was a random walk : impossible to predict. This could have been detected by a careful pre-investigation...

Contents

Data science

- Data science is not (or at least should not) attempting to obtain the best performances by launching DL packages in Python
- Data science also aims at understanding the data, interacting with experts, bring human intelligence and expertise and improve knowledge
- Artificial intelligence cannot be intelligent if the data scientist is not
- Applying complex DL methods does not prevent from a thorough preliminary phase... **and ML can also help for this!**

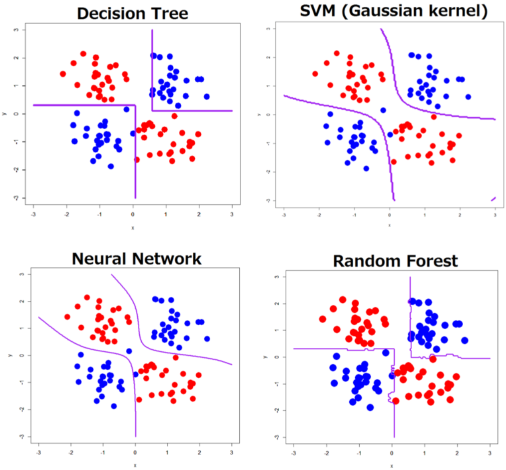
Understanding data: complex and time-consuming task



Source: https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/

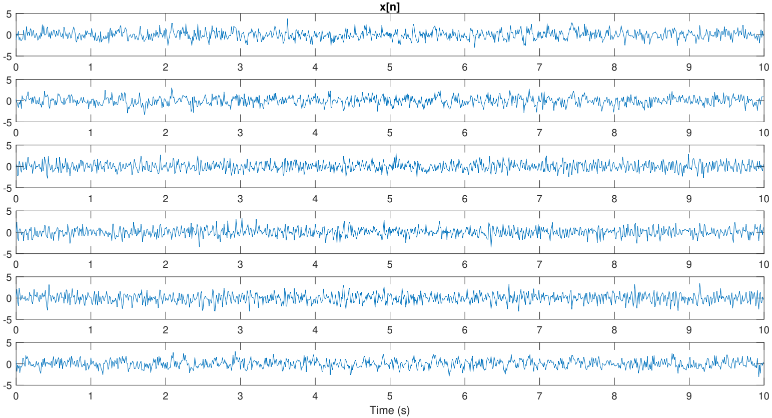
- Understanding the data for extracting the relevant information
- Understand what you do, why you do it and how you do it: interpretability

Representation vs. complexity



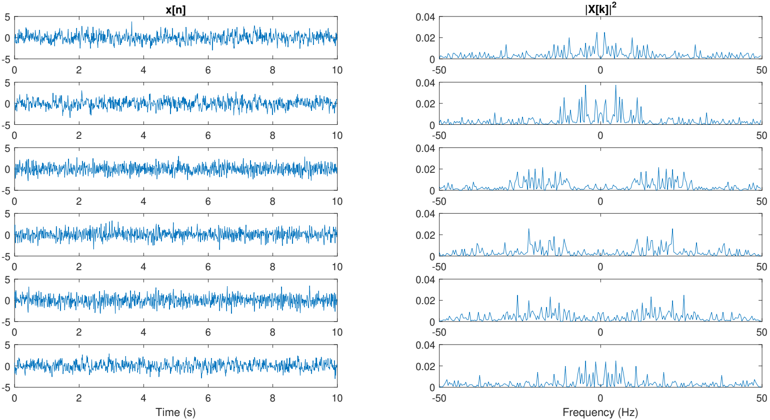
Data complexity often translates into algorithm and model complexity

Importance of representation



Two classes of signals?

Importance of representation



Trivial in the frequency domain

Main ML tasks for time series

- **Prediction:** Predict the future values a time series
- **Completion/interpolation:** Recover missing/lost samples in a time series
- **Classification:** Assign a class label to a time series or to a subsequence
- **Clustering:** Form several groups of time series with the same properties
- **Query by content/indexation:** Given an input time series, retrieve the closest time series in a large database up to a given measure of fit
- **Segmentation/change-point detection:** Find significant abrupt changes in the time series
- **Anomaly detection:** Find abnormal events in a time series
- **Pattern extraction:** Find repetitive events in a time series

Hidden ML tasks for time series

- **Understand the data:** know where they come from, how they were acquired, what are their characteristics, interact with domain-experts and understand their problems
- **Improve the data:** find accurate representation spaces where the events of interest can be seen, consolidate the data (denoising, detrending, detection/removal of outliers)
- **Model the data:** physical/statistical or expert-based models, simple, adaptive and interpretable models
- **Extract information from the data:** find repetitive patterns, features of interest, change-points, anomalies

Contents

Aim of the course

Machine Learning for (understanding) Time Series

- Focus on the *hidden tasks*: understand, improve, model and extract information
- Interpretable and reproducible ML algorithms: white boxes (no Deep Learning)
- Unsupervised and semi-supervised ML approaches
- Methodology can be applied for prediction, classification, clustering etc...

Outline of the course

- Lecture 1: Pattern Recognition and Detection
- Lecture 2: Feature Extraction and Selection
- Lecture 3: Models and Representation Learning
- Lecture 4: Data Enhancement and Preprocessings
- Lecture 5: Change-Point and Anomaly Detection
- Lecture 6: Multivariate Time Series