Advanced Time Series

Lecture 1:
Introduction to time series

Gleb Ivashkevich

The team

Gleb Ivashkevich

Anatoly Bardukov

What is ATS about?

Time series are **sequences.**

- all the sequential **deep learning** blocks can be applied (RNN, CNN, combinations, incl. transformers → adaptations)
- **various problems:** forecasting, classification, event prediction, representation learning
- **mental image** of time series problems - how to select parameters, architectures, etc.

What ATS is not about?

- AR, VAR, ARIMA, etc.
- specific domains
- econometrics
- financial time series

We will briefly **overview** the classical stuff. And will use it to build some **intuition**.

Course logistics

6 lectures, 3 hours each, Google classroom

- slides + notebook(-s)
- 4 homeworks (in teams)
- 2 paper reviews + 2 paper implementations
- manual grading
- Slack

Course structure

Intro + 4 topics:

- TS **forecasting**: RNNs$^{\text{(various)}}$, transformers
- TS **classification**: CNN$^{\text{(various)}}$ and combined$^{\text{(RNN+CNN)}}$ models
- **TTE prediction**: DL models for predictive maintenance, basics of survival analysis
- TS **representation learning**: VAMPnets, autoencoders$^{\text{(various)}}$

Course structure

Power consumption
Weather
Sales
Traffic

Forecasting

AR models (review only)
RNNs
Probabilistic forecasts

ATS

Activity data (inertial sensors)
Sound
Medical signals

Classification

Dilated/causal convolutions
CNNs and hybrid models
Attention mechanism
Segmentation

Any highly structured TS

Representation

VAMPnets
AEs

Equipment data
Earthquakes and other physical signals

TTE prediction

Survival models
Interpretability (very basics)

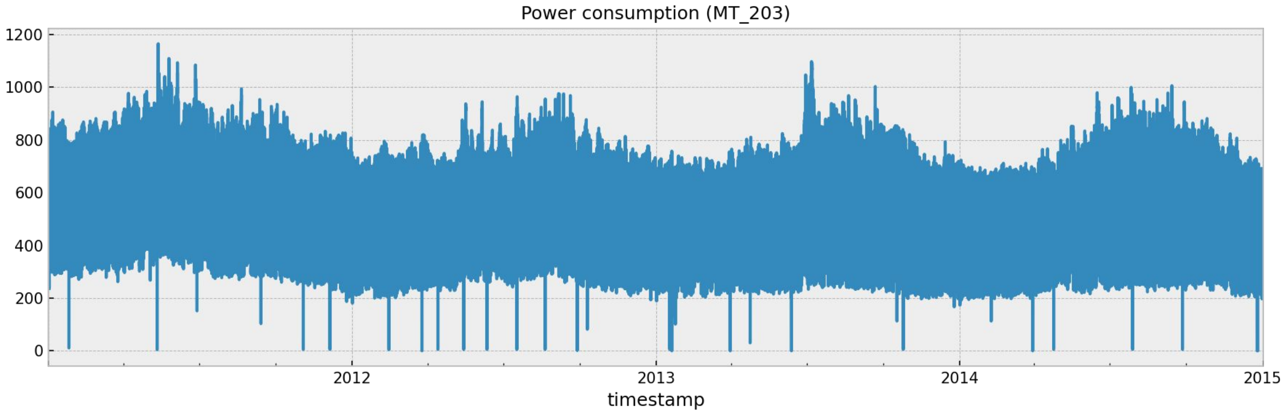
Time series basics:

temporal

Power consumption

Electricity load [dataset]

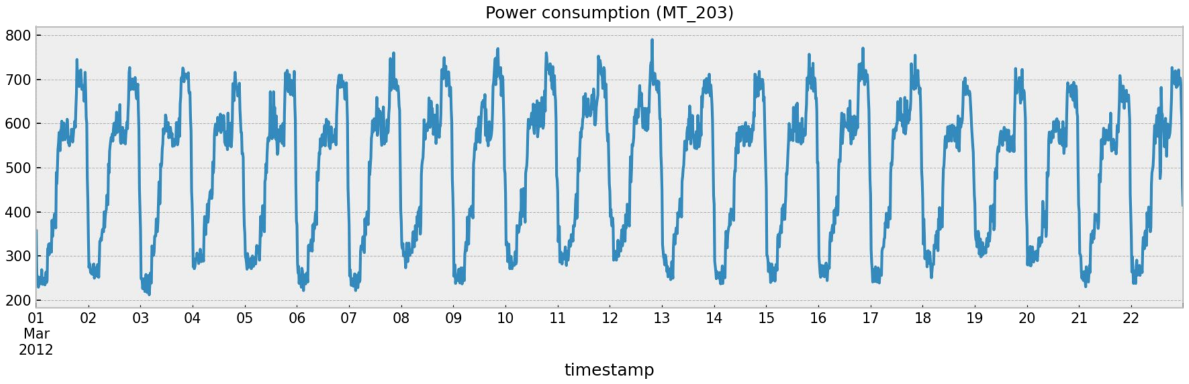
- 370 individual households
- 15 minutes sampling interval



Power consumption

Electricity load [dataset]

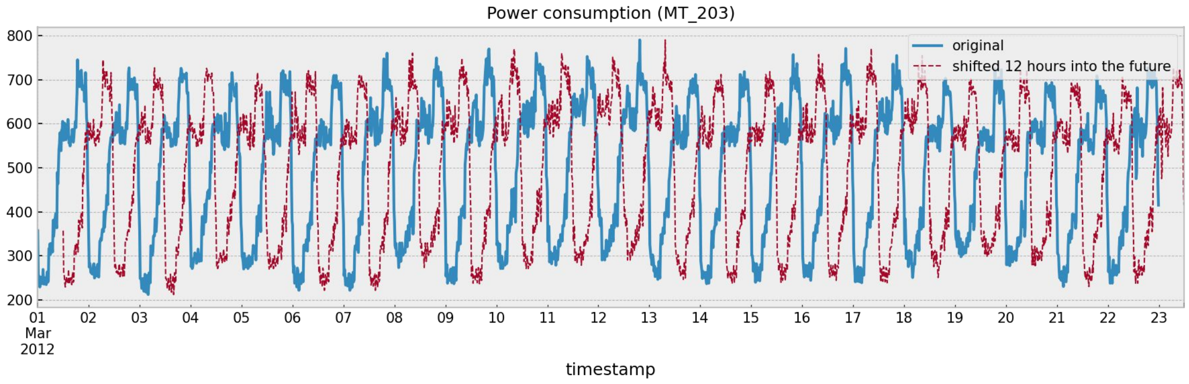
- 370 individual households
- 15 minutes sampling interval



Power consumption

Temporal structure?

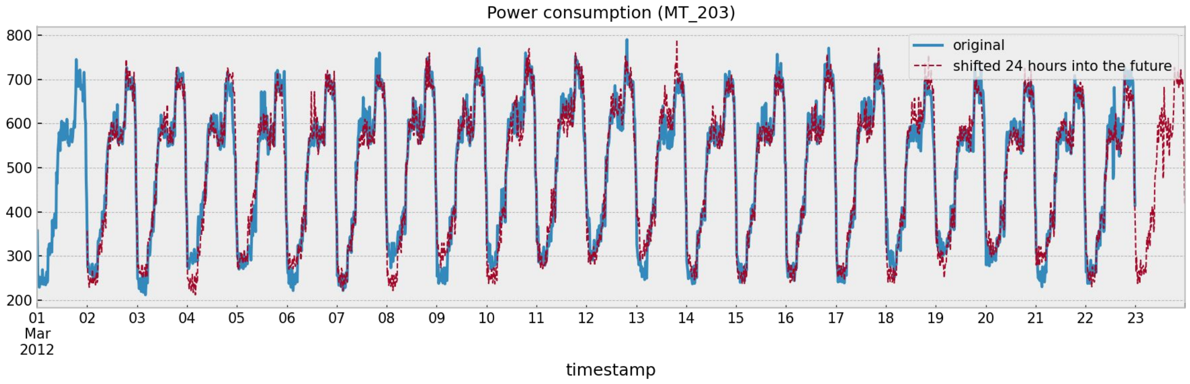
- shift the data: 12 hours



Power consumption

Temporal structure?

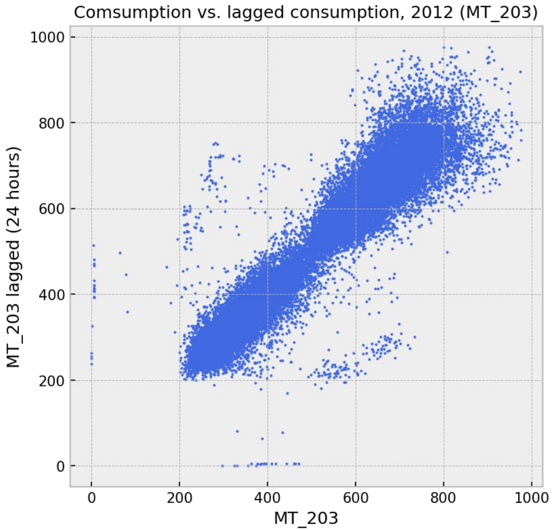
- shift the data: 24 hours

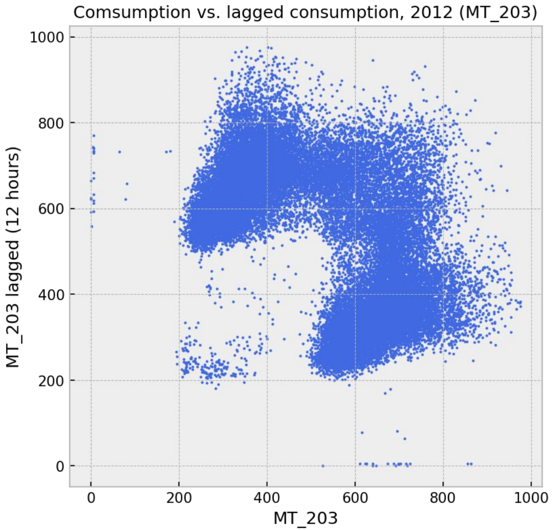


Power consumption

Temporal structure?

- shift the data: 12, 24 hours





Modeling

Modeling ideas:

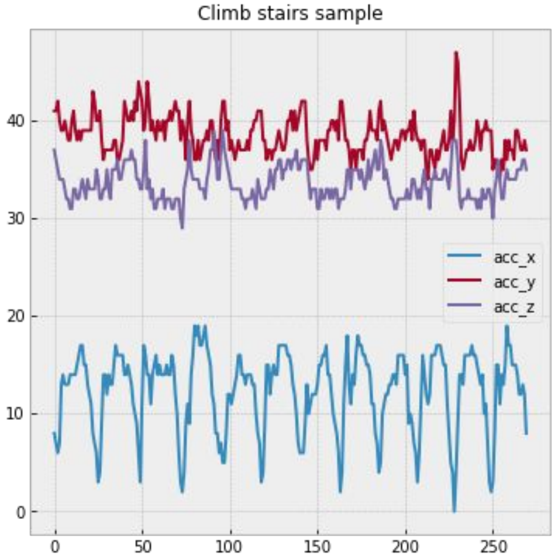
- use some AR-alike classical model
- use some RF, gradient boosting, provide lags explicitly
- **recurrent** model (modified?)

Time series basics:
structural

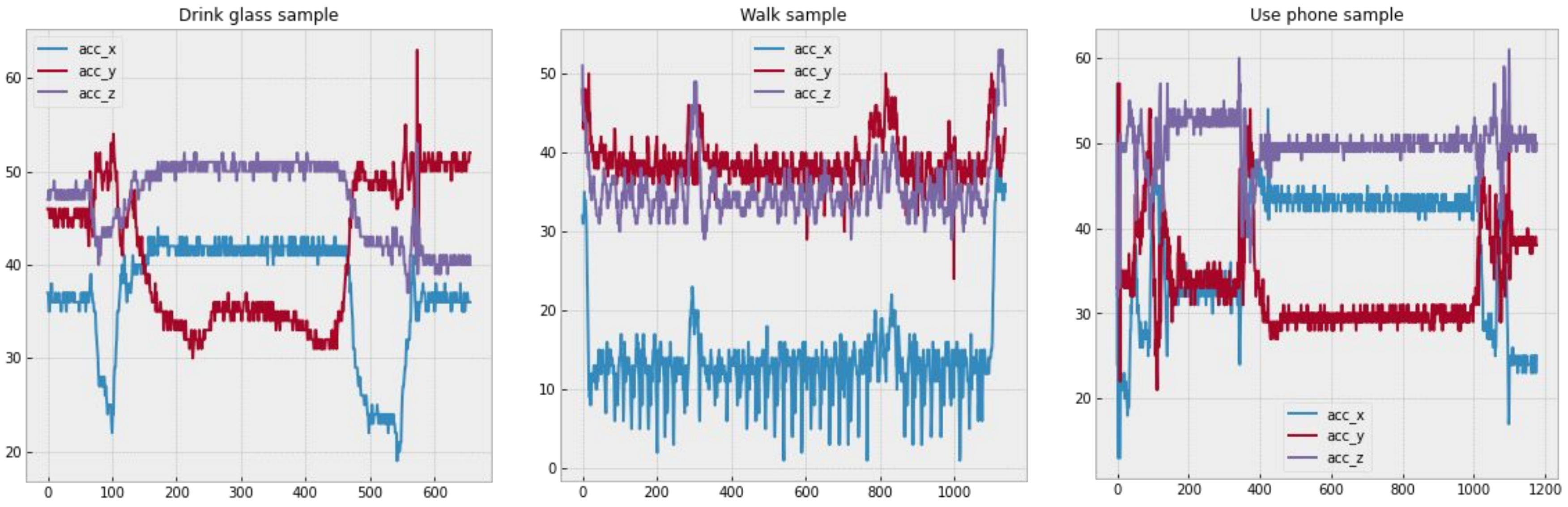
Activity recognition

ADL Recognition [dataset]

- multiple activities, short samples
- 50 Hz sampling rate



Activity recognition



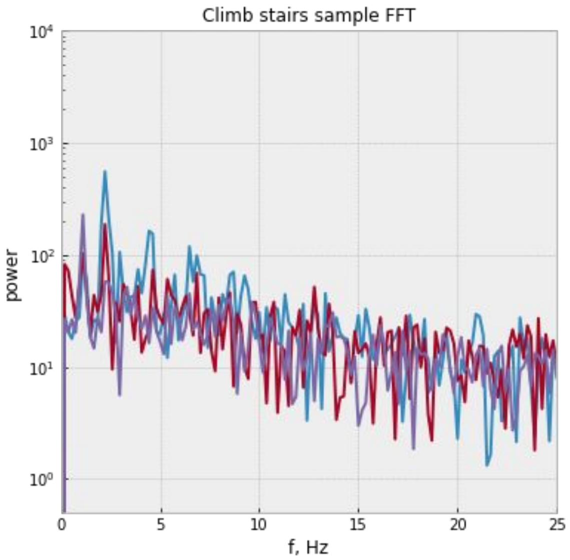
Activity recognition

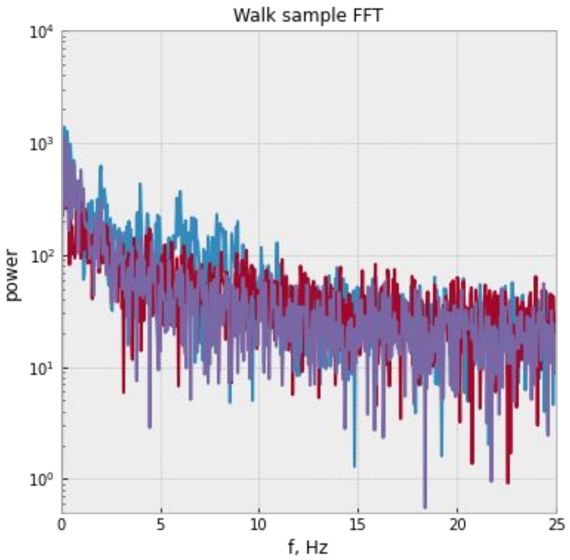
ACF? Non-informative: wrong “scale”

- FFT for the rescue

Activity recognition

Power spectrum:





Modeling

Modeling ideas:

- use manual features (`tsfresh`) and some classical model
- **CNN** model (modified?)

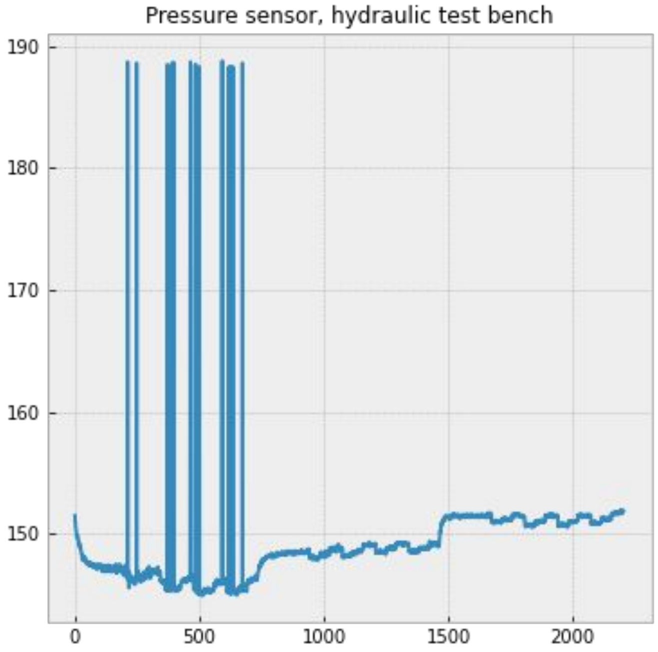
Time series basics: other examples

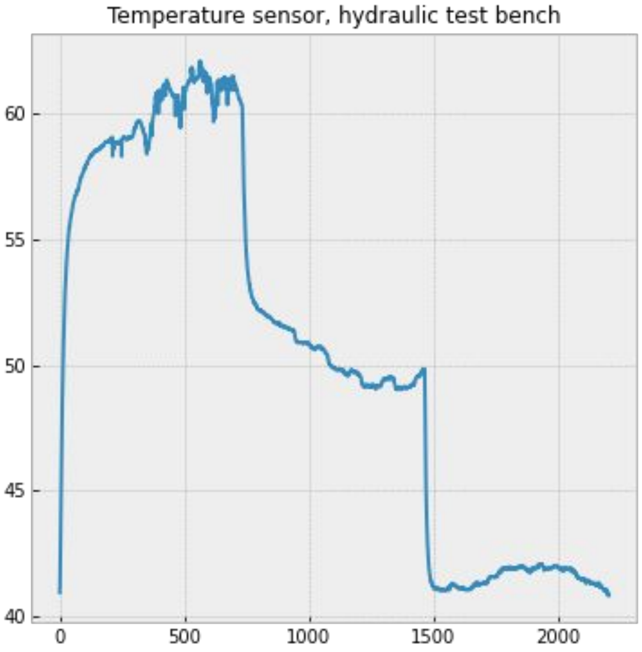
Industrial equipment

Condition monitoring of hydraulic systems [dataset]

- multiple sensors (temperature, pressure)
- 100 Hz sampling rate

Industrial equipment





Time series basics: tools

Pandas

Mostly for preprocessing:

- datetime operations
- resampling, rolling
- shifts

Cross-validation

Random split cannot be applied to time series

- use fixed split
- use rolling CV
- otherwise, autocorrelation will kill your model

Forecasting I

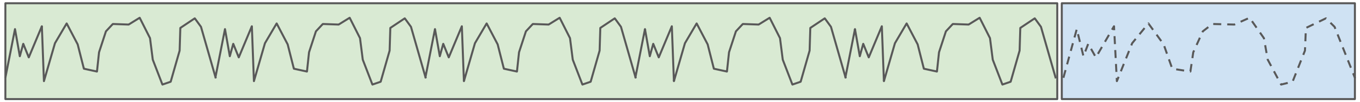
TS forecasting problem

Forecasting:

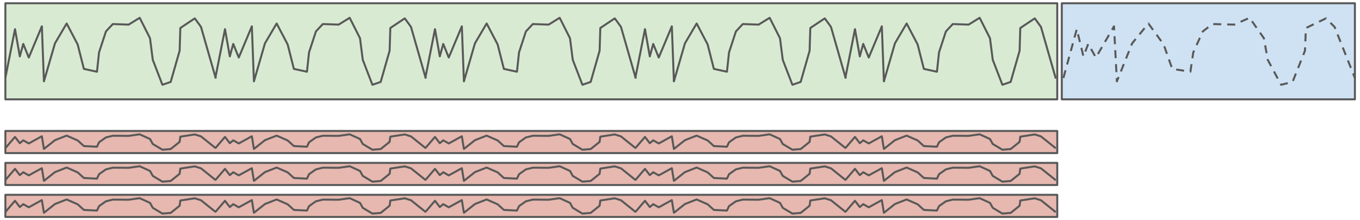
- estimate the **target time series in the future** using past data (endogenous)
- sometimes, you may know **something else besides target** (exogenous)
- depends strongly on **time scales** of relevant processes

TS forecasting setup

endogenous only weather time series, power consumption, sales

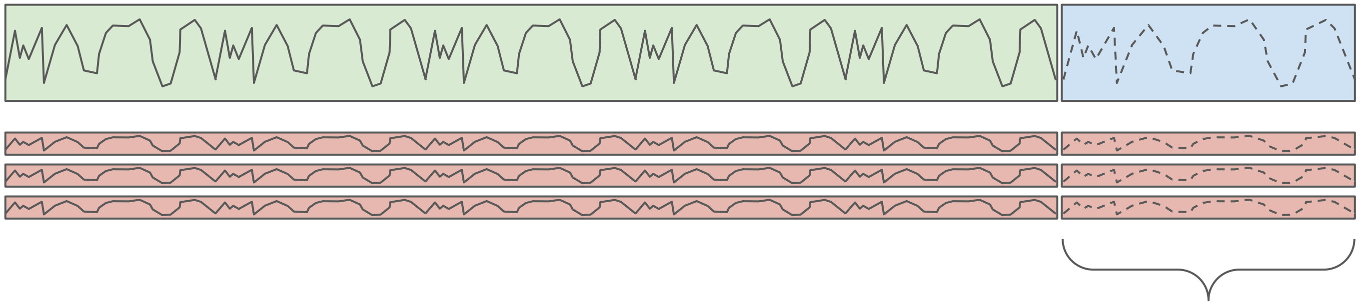


endogenous + past exogenous manufacturing



TS forecasting setup

endogenous + past and future exogenous power consumption, sales



may be a forecast

TS forecasting: past and future

- future values may depend on past information
- they can depend on the future information as well
- you cannot forecast if you do not have information
- **no free lunch**

TS forecasting: past and future

Power consumption **tomorrow** depends on:

- consumption **today, yesterday**, etc.
- weather **tomorrow**,
- traffic **tomorrow**

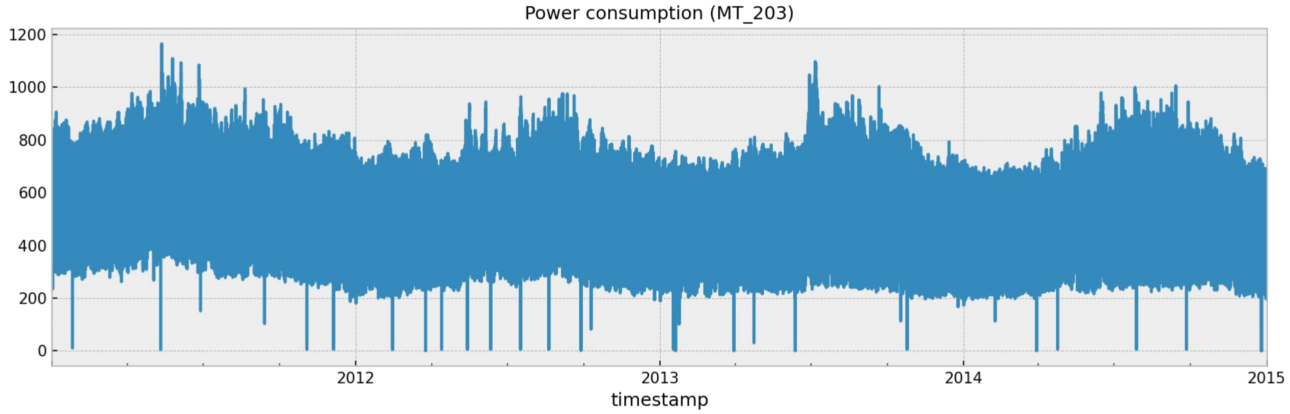
Time series: concepts

- stationarity: $X(t), X(t+1), \ldots \rightarrow X(t+h), X(t+1+h), \ldots$
- seasonality: season variations $\rightarrow$ calendar
- trend
- autocorrelation

Power consumption

Electricity load dataset

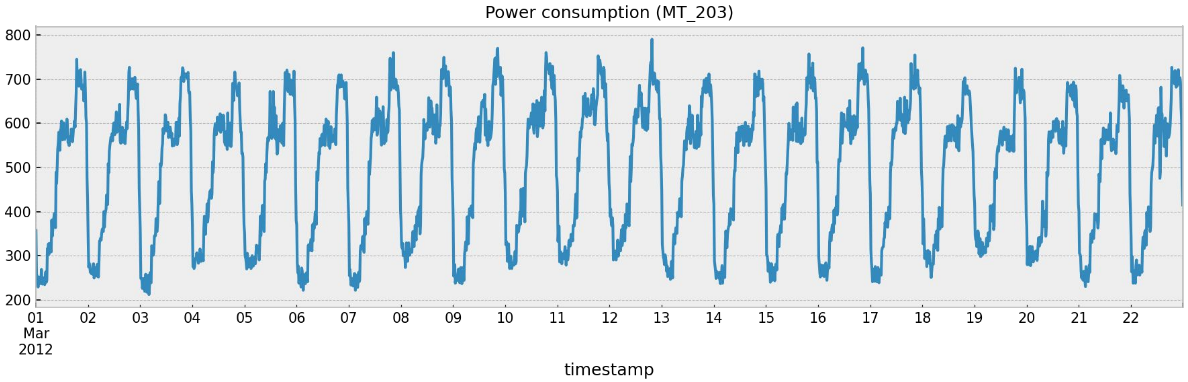
- 370 individual households
- 15 minutes sampling interval



Power consumption

Electricity load [dataset]

- 370 individual households
- 15 minutes sampling interval

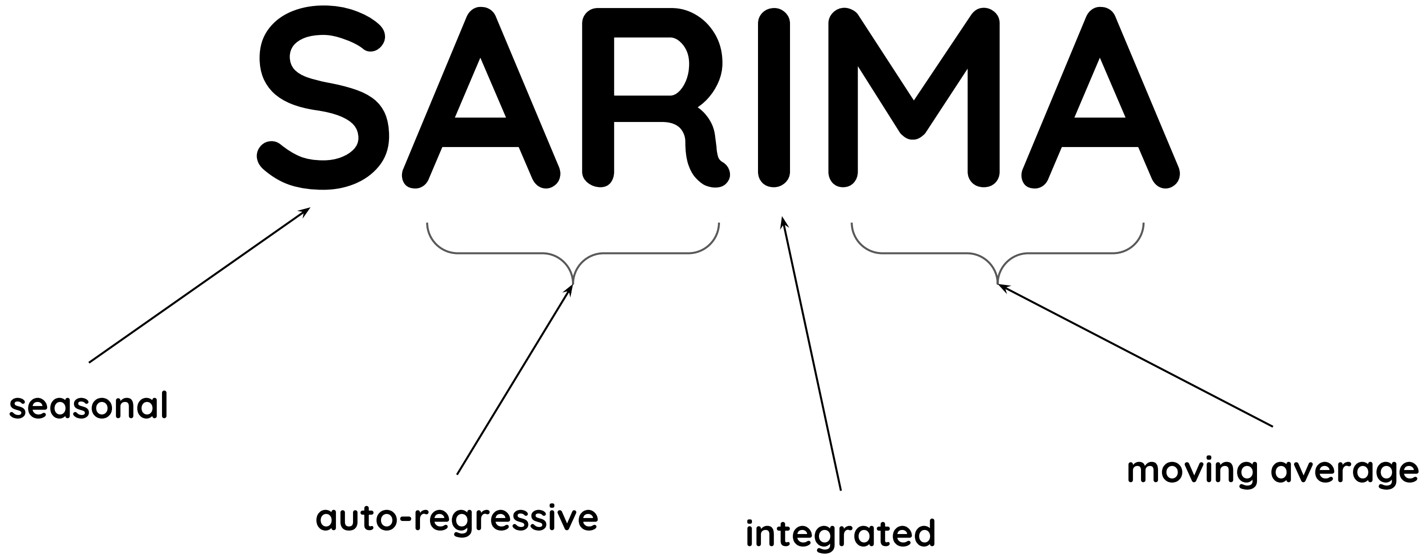


Classical models

- AR: autoregressive
- MA: moving average
- ARIMA: AR integrated moving average

More: Time Series: Autoregressive models AR, MA, ARMA, ARIMA

SARIMA


SARIMA
seasonal
auto-regressive
integrated
moving average

SARIMA: tools

`statstools`:

- a lot of time series functionality
- a lot of classical time series models
- convenient plotting

Classical models limitations

- linearity
- multiple seasonalities
- stationarity
- somewhat tricky
- **good baseline**

Data generation process

- all the underlying processes, which result in the observed data
- may be multilayered and non-linear
- not everything is known at inference time

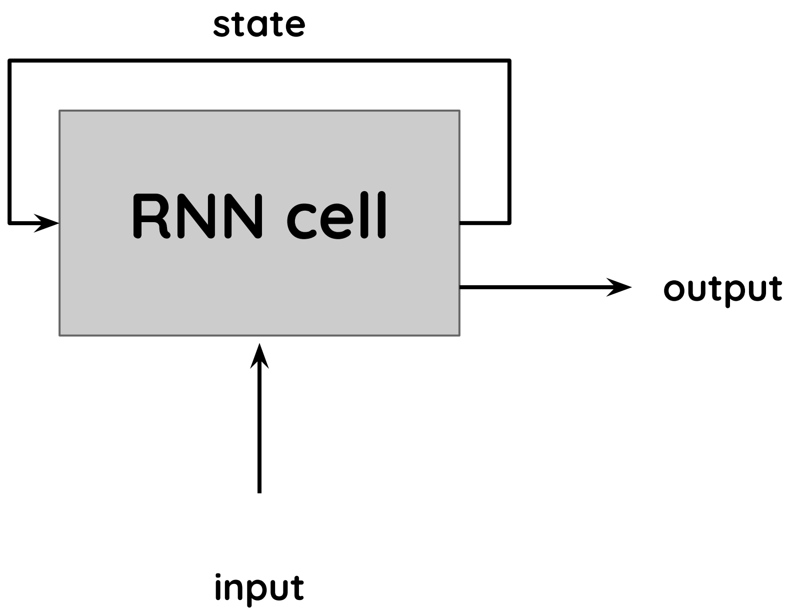
Recurrent models

Power consumption

Power consumption [data in US grid](#):

- multiple years
- hourly
- generally clean
- seasonality patterns on multiple scales

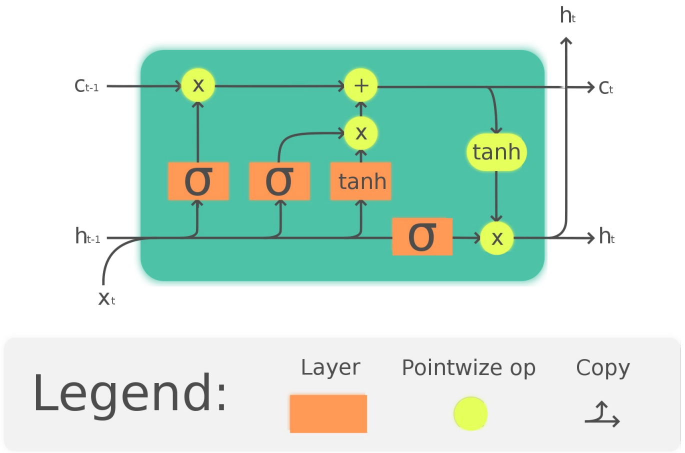
Recurrent neural network



Recurrent neural network

- **sequential** data
- have their own problems
- can be combined with other blocks (CNN)

LSTM: idea

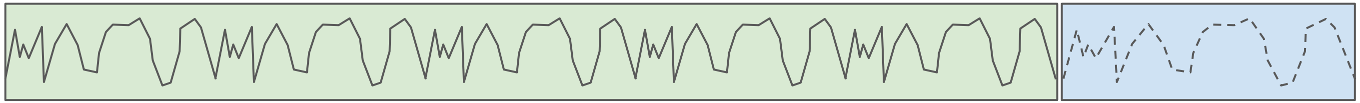


$$
\begin{aligned}
f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f)\\
i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i)\\
o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o)\\
\tilde{c}_t &= \sigma_c(W_c x_t + U_c h_{t-1} + b_c)\\
c_t &= f_t \circ c_{t-1} + i_t \circ \tilde{c}_t\\
h_t &= o_t \circ \sigma_h(c_t)
\end{aligned}
$$

Encoder-decoder architecture

Encoder-decoder setup

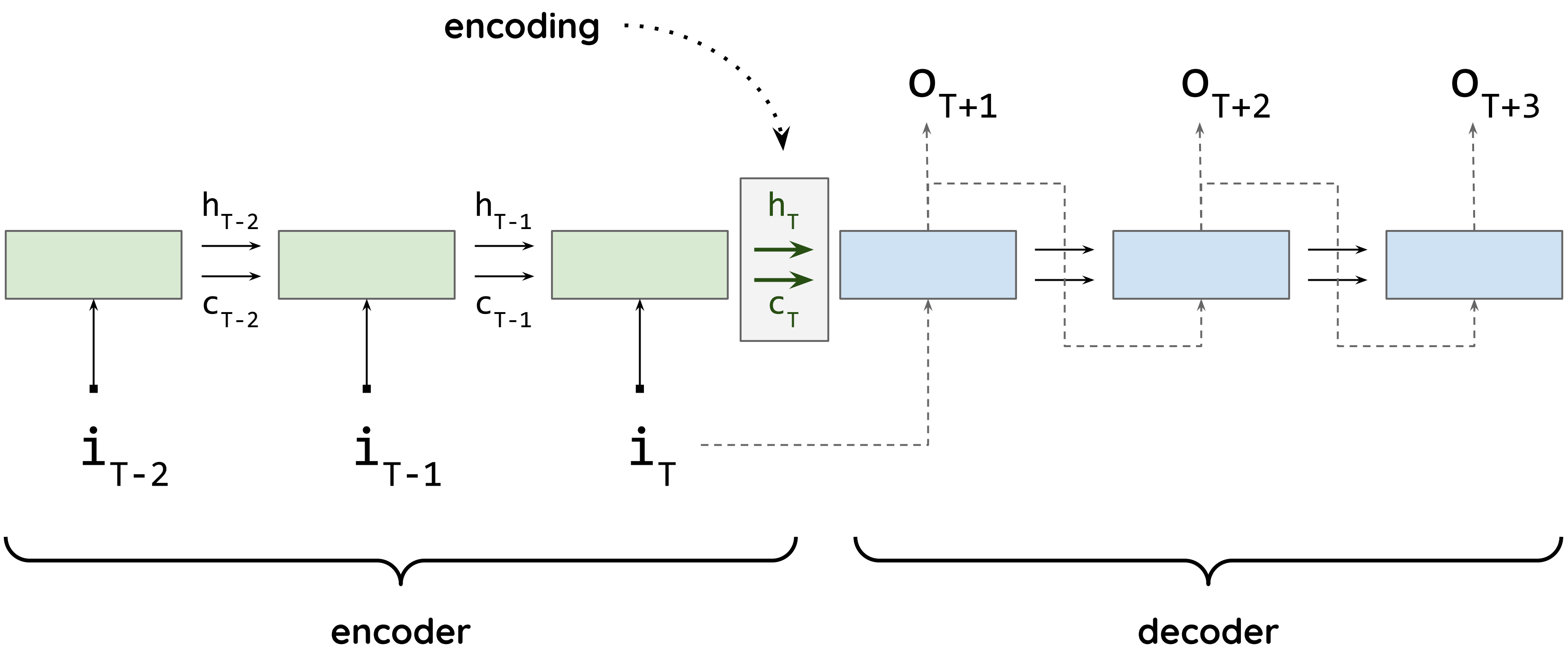
endogenous only weather time series, power consumption, sales



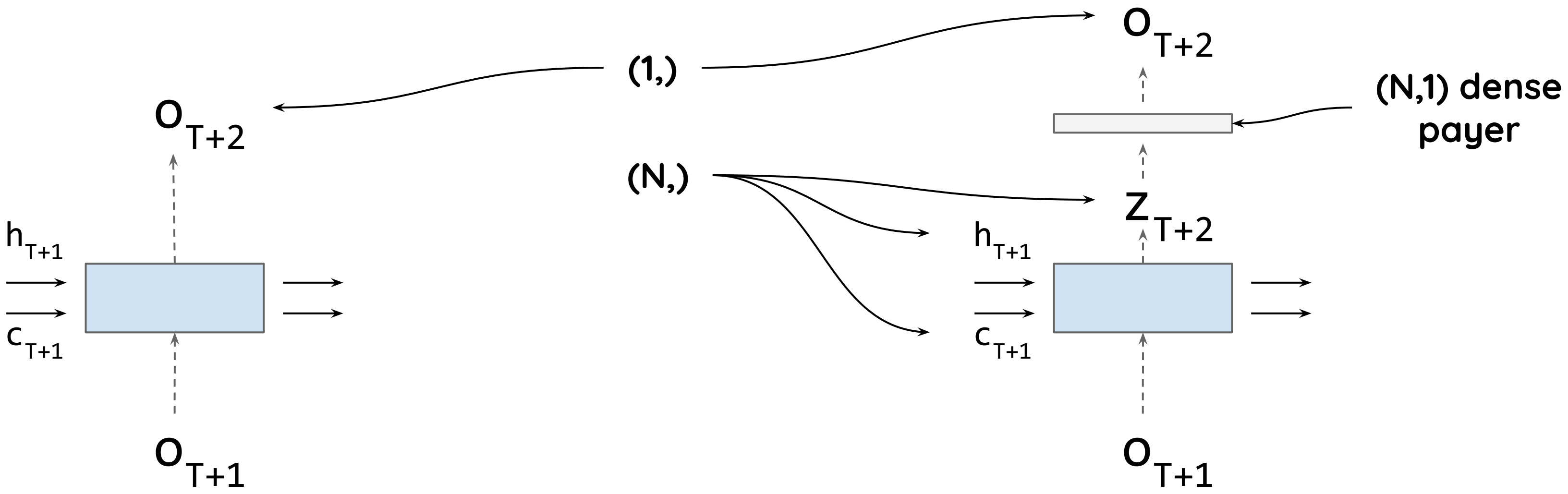
The most basic setup:

- **only target** itself is used
- **calendar** information may be added

Encoder-decoder arch



Encoder-decoder: implementation details



*1D target, N hidden units

Encoder-decoder arch

Variations:

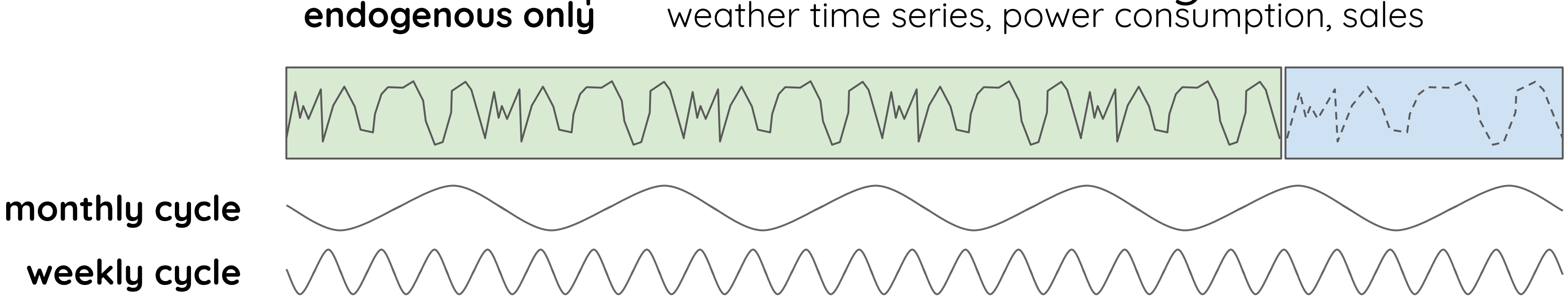
- encoder and decoder may **share weights**
- encoder and decoder may have **different architectures**
- **calendar** information, **exogeneous** variables may be added
- **categoricals** may be added (one-hot or embeddings)

Calendar information

Two main options:

- **one-hot encoded** (month, day of week, weekend/weekday, holidays, sale)

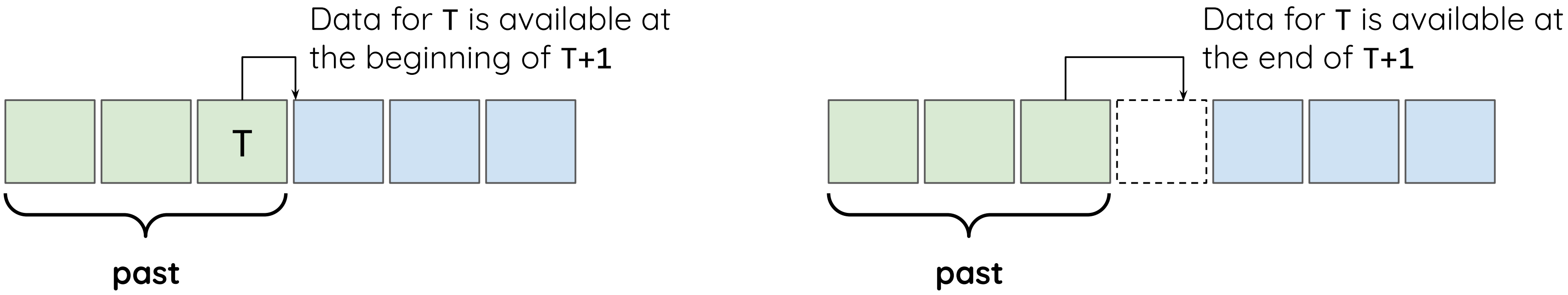
- **Fourier features:** explicit multi-seasonality



Production considerations

Forecasting windows:

- it is usually desirable to forecast on **regular intervals**
- based on data availability, you may need to **skip a window**



questions?