

H2O Time Series

Using machine learning for solving time series problems

Outline

- Introduction
- Time series modeling and processing
- Machine learning with time series
- Time series sources
- Conclusions

Introduction

- Motivation: many data come as time series. How can we use Machine Learning approaches to solve problems?
- Problems:
 - Analysis and understanding
 - Prediction/forecasting, classification, detection
 - Decision making and control
- Machine Learning and Time Series:
 - Different than standard machine learning problem
 - There are dynamic issues that need to be dealt with

Time series modeling and processing

- Definitions
- Operations with time series
- Basic modeling: linear time-invariant models (LTI)
- Advanced topics

Definitions

- What is a time series?

A measurement of quantity taken over time: in general, the output or state of a dynamic system.

- Sequential nature of time series is fundamental for its processing

- Mathematically:

If f is a function of time, then $\{f(t) \mid t \in T\}$ is a time series

- Characterization:

- Time continuous $f(t)$ or discrete $f(t_i) = f[i]$
- Numeric or Symbolic (categorical) according to f co-domain
- Single variable if $f(t)$ scalar, or multivariable if $f(t)$ is vector
- Symbolic sequences can be considered as time series (text, DNA)

Time series analysis and processing

Numerical time series analysis

- Converting from continuous to discrete time (Nyquist-Shannon sampling theorem):
A band-limited function $x(t)$ can be converted to a discrete function $x[i] = x(i/f_s)$ with no loss, provided that limit frequency $B < f_s/2$
- Analysis in time or transformed domains (for $x(t)$, $y(t)$)
- Transform operators:
 - Fourier transform and series (spectrum, frequency): $X(f) = \mathcal{F}(x(t))$
 - Laplace transform (complex domain, continuous time) $X(s) = \mathcal{L}(x(t))$
 - Z-transform (complex domain, discrete time) $X(z) = \mathcal{Z}(x[i])$
- Time domain operators:
 - Auto-correlation, Cross-correlation, Covariance: $r_{xx}(t)$, $r_{xy}(t)$
 - Convolution: $(x * y)(t)$

Time series analysis and processing

Linear time-invariant (LTI) systems

- Linear relation between input and output functions (time series)
- Invariant parameters
- Model (discrete-time):

$$y[i] = \sum_{k=0}^n b_k u[i - k] - \sum_{k=1}^n a_k y[i - k]$$

- Where
 - u is the input
 - y the output
 - n the order of the model
 - a_k are the *autoregressive or feedback* parameters
 - b_k are the *moving average or feedforward* parameters

Time series analysis and processing

Z-transform

- Converts function from discrete time domain into complex domain
- Definition

$$\mathcal{Z}\{x[i]\} = X(z) \stackrel{\text{def}}{=} \sum_{i=-\infty}^{\infty} x[i] z^{-i}$$

- Examples

Time domain	Z-transform
Unit pulse $\delta[i] = \begin{cases} 1, & i = 0 \\ 0, & i \neq 0 \end{cases}$	$\mathcal{Z}\{\delta[i]\} = 1$
Unit step $u[i] = \begin{cases} 0, & i < 0 \\ 1, & i \geq 0 \end{cases}$	$\mathcal{Z}\{u[i]\} = \frac{1}{1 - z^{-1}}$

Time series analysis and processing

Z-transform

Properties	Time domain	Z domain
Time expansion	$x_K[i] = \begin{cases} x[r], & i = Kr \\ 0, & i \notin KZ \end{cases}$	$X(z^K)$
Time shifting	$x[i - k]$	$z^{-k} X(z)$
Time reversal	$x[-i]$	$X(z^{-1})$
Scaling Z domain	$\alpha^{-i} x[i]$	$X(\alpha z)$
Differentiation	$i x[i]$	$-z \frac{dX(z)}{dz}$
Convolution	$x_1[i] * x_2[i]$	$X_1(z) X_2(z)$

Time series analysis and processing

Transfer Function

- Output of linear model can be found by mean of a Transfer function
- Building transfer function with Z-transform:

$$\mathcal{Z}\{y[i]\} = \mathcal{Z}\{\sum_{k=0}^n b_k u[i-k] - \sum_{k=1}^n a_k y[i-k]\}$$

$$Y(z) = U(z) \sum_{k=0}^n b_k z^{-k} - Y(z) \sum_{k=1}^n a_k z^{-k}$$

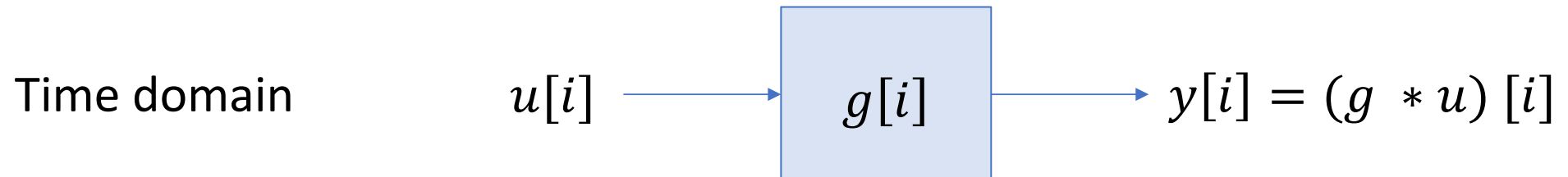
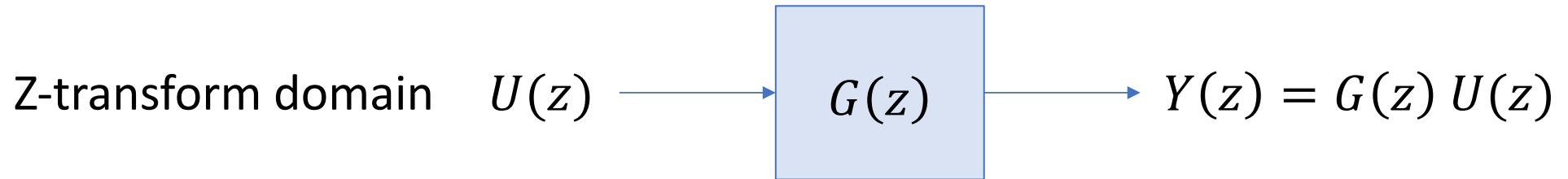
$$Y(z) \{1 + \sum_{k=1}^n a_k z^{-k}\} = U(z) \sum_{k=0}^n b_k z^{-k}$$

- Finally the transfer function $G(z)$:

$$G(z) = \frac{Y(z)}{U(z)} = \frac{\sum_{k=0}^n b_k z^{-k}}{1 + \sum_{k=1}^n a_k z^{-k}}$$

Time series analysis and processing

Transfer Function



- The time-domain response of a linear model is the convolution between the input $u[i]$ and the unit pulse response $g[i]$

Time series analysis and processing

State space formulation

- In this formulation the internal state \mathbf{x} is accounted explicitly (reflecting the all the past story of the system):

$$\mathbf{y}[i] = \mathbf{C} \mathbf{x}[i] + \mathbf{D} \mathbf{u}[i]$$

$$\mathbf{x}[i + 1] = \mathbf{A} \mathbf{x}[i] + \mathbf{B} \mathbf{u}[i]$$

- Where \mathbf{A} is (square) state transition matrix, \mathbf{B} control matrix, \mathbf{C} output matrix and \mathbf{D} feedforward matrix
- State formulation may be more convenient for analysis, simulation and implementation (multivariable case)

Time series analysis and processing

Stochastic or random process (discrete time)

- Sequence of random variables realization (with common statistics)
- *White noise*: uncorrelated, flat power spectrum:

$$n_w(\mu = 0, \sigma)$$

- Correlated stochastic process: can produced by feeding white noise to a linear model. Examples:
 - *Random walk* or *Brownian motion* (red noise):
white noise + integrator (summation): $G(z) = \frac{1}{1-z^{-1}}$, $w[i] = \sum_{k=0}^i n[k]$
 - *Pink noise*, *grey noise*, etc.

Time series analysis and processing

Advanced Topics

- Multiple inputs/Multiple outputs (MIMO) models: using vectors and matrices $\mathbf{y}[i] = \sum_{k=0}^n A_k \mathbf{u}[i - k] - \sum_{k=1}^n B_k \mathbf{y}[i - k]$
- Linear time-variant systems: can be used to model non-linear systems: $y[i] = \sum_{k=0}^n b_k(t) u[i - k] - \sum_{k=1}^n a_k(t) y[i - k]$
- Non-linear models: $y[i] = f(u[i], \dots, u[i - k], y[i - 1], \dots, y[i - k])$
- Stochastic process with time variant parameters: $\sigma(t)$
- Symbolic sequence modeled by finite state automata

Machine learning with time series

Typical steps:

1. Conversion data to regular time series (if needed)
 - Sampling rate or period to use
 - Convert irregular time data (or events) into regular sampled
 - Resample (in time) and convert to numeric categorical/symbolic variables
2. Dynamic preprocessing (optional): convert dynamic data into static pattern
 - Tapped delay line (time shift), fixed
 - Apply dynamic filter (time convolution), adjustable
 - Use transformed domain
3. Apply machine learning
 - Static (standard) if dynamic processing is available
 - Dynamic models: recurrent neural networks and variants

Time series sources

- Medical and Biological
- Language: Natural Language, Speech and Music
- Nature and Environment
- Energy
- Industrial, Control, Machinery
- Financial and Economic
- Communication and Networks

Time series: Medical and Biological

- Nervous and muscular systems activity: Electrical activity from neurons (in brain or muscles)
 - electroencephalogram (EEG),
 - electrocardiogram (ESG)
 - electromyography (EMG)
 - polysomnography (PSG)
- Genomics: ADN sequences can be analyzed as time series

Time series: Natural Language, Music

- Speech processing: speech understanding (speech to text), speech synthesis, translation, compression
- Natural Language processing: understanding, translation, knowledge extraction, synthesis, summarizing
- Music: classification, synthesis, intelligent composition

Time series: Nature and Environment

- Meteorological variables: temperature, humidity, pressure, humidity, rainfall. Mapping, forecast
- Pollutants: source detection, modeling, forecasting
- Earthquakes: detection ground motion and waves, modeling, prediction(?). Water waves (tsunami)
- Astronomy and Astrophysics data

Time series: Energy

- Electric power consumption from large grid, micro grid or single consumer: modeling, forecast, control(?)

Time series: Industrial, Control, Machinery

- Signal detection and measurement (smart/soft sensors)
- System control: predictive, non-linear, multivariable, etc.
- Distributed sensors (internet of things)

Time series: Financial and Economic

- Financial markets values (stock, index, commodities, currencies):
- Econometric and macroeconomic series (PGB, CPI, rates, etc.)
- Consumer finance: credit risk, consumption, payments, fraud.

Time series sources: Communication and Networks

- Baseline signals: production, detection, errors,...
- Internet traffic, routing, information
- Local networks: optimization, detection, assignments
- Transportation networks: intelligent transportation

Conclusions

- There are many machine learning problems that are based on time series data (most?)
- Time series methods and models should be used when possible
- Machine learning solutions can be integrated seamlessly to existing production networks.
- Challenges: parallelize and on-line learning