REGIME DETECTION WITH STATE SPACE MODELS

5.6.2018

Jan Reichl

CONTENTS

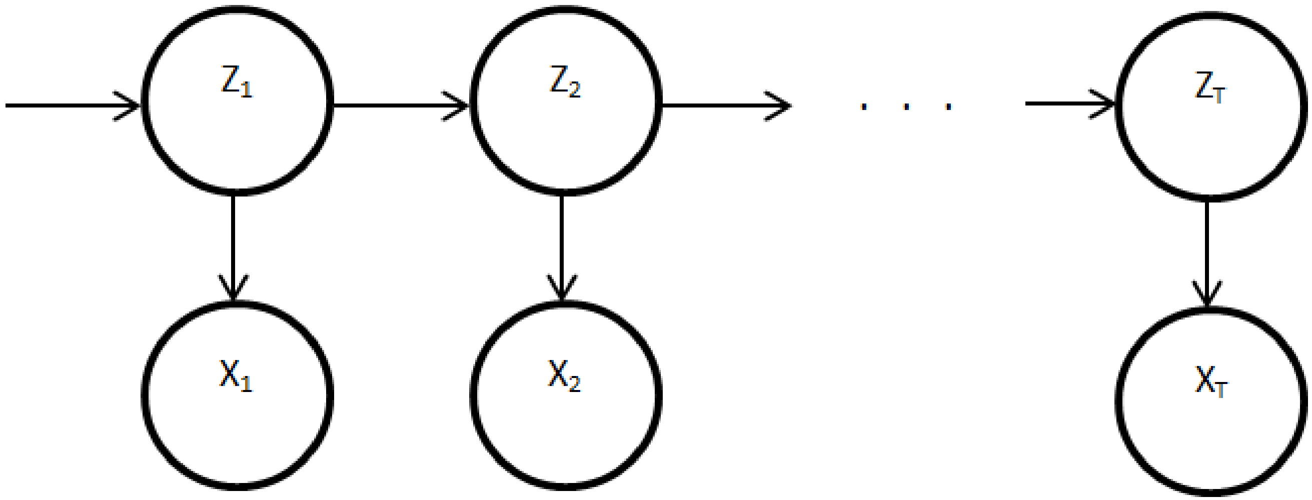
- Motivation
- Theoretical introduction
  - Model structure assumptions
  - Learning and inference
  - Drawbacks
- Applications
  - Simulated data example
  - Real world application

MOTIVATION

- Sequential data analysis
  - supervised learning – like any other classifier, not interesting for us,
  - *unsupervised learning* – clustering of sequential data, capturing changes in time series dynamics in probabilistic fashion,
- bootstrapping data while using estimated dynamics,
- regime change insights.

REPRESENTATION

- K hidden, unobservable states z_t with Markov transition matrix $A \in R^{K,K}$, given by $a_{ij} = P(z_t = j | z_{t-1} = i)$ and starting probability $\pi_i = P(z_1 = i)$,
- observations x_t with conditional distribution $p(x_t | z_t = k) = f_k(x_t | \Phi_k)$.
- Generally, model λ is described with an unknown set of parameters $\theta = \{\pi, A, \boldsymbol{\Phi}\}$.



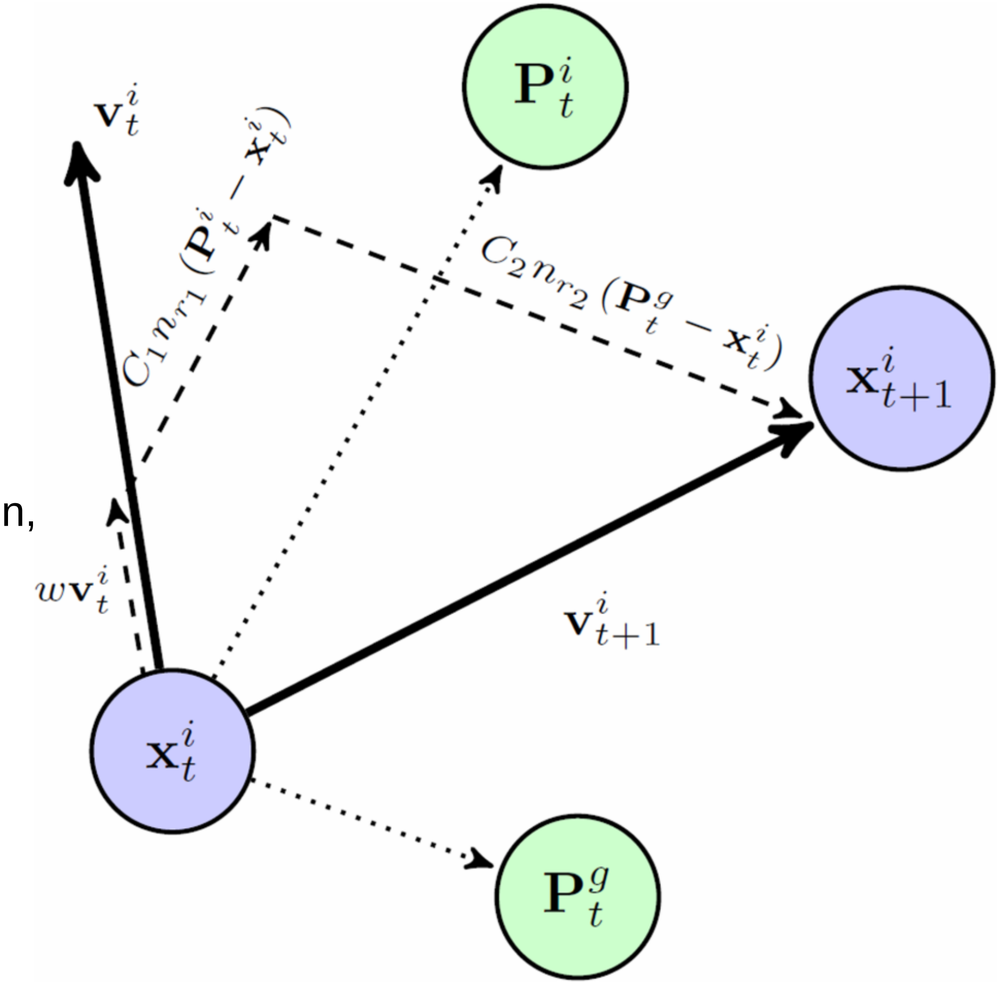
LEARNING

- Model $\lambda(\{\pi, A, \boldsymbol{\Phi}\})$ has to be fitted to the data $x_{1:T}$.
- Unsupervised learning the model is achieved via maximizing the likelihood function,
  - Baum-Welch algorithm – local search, multiple initializations, monotonic,
  - Particle swarm optimization – global search, costly computation, probabilistic constraints, implicit parameter regularization.
- Fit diagnostic with likelihood value and bootstrap for parameter correlation.

PARTICLE SWARM OPTIMIZATION

- Problem $\min_{x \in \Gamma} f(x)$
- objective function f,
- constrained space Γ.

- Initialize parameters ω, C_1, C_2, I,
- initialize swarm $\{x^i{}_t\}_{i \in I}$,
- each swarm particle $x^i{}_t$ is a solution,
- iteration at time t:
  - $n_{r1}, n_{r2} \sim U[0,1]$,
  - particle's best solution $P^i{}_t$,
  - swarm's best solution $P^g{}_t$.



Internal

INFERENCE

- Conditioned on the fitted model $\lambda(\hat{\pi}, \hat{A}, \widehat{\boldsymbol{\Phi}})$ and observations up to time T we can smooth, filter and predict, i.e., evaluate the posterior distribution
  - $p(z_t|x_{1:T})$ – past states probability,
  - $p(z_T|x_{1:T})$ – current states probability,
  - $p(z_{T+1}|x_{1:T})$ – future states probability.

Internal

DRAWBACKS

- Large number of parameters and static estimates,
- training sequence selection - overfitting to irrelevant data,
- complicated model identification,
- complicated model selection/comparison – likelihood ratio test, R-squared,
- state duration distribution, $P(z_t = k, \ldots, z_{t+\tau} = k, z_{t+\tau+1} \neq k) = (1 - a_{kk})(a_{kk})^{\tau - t}$, may decrease too fast.

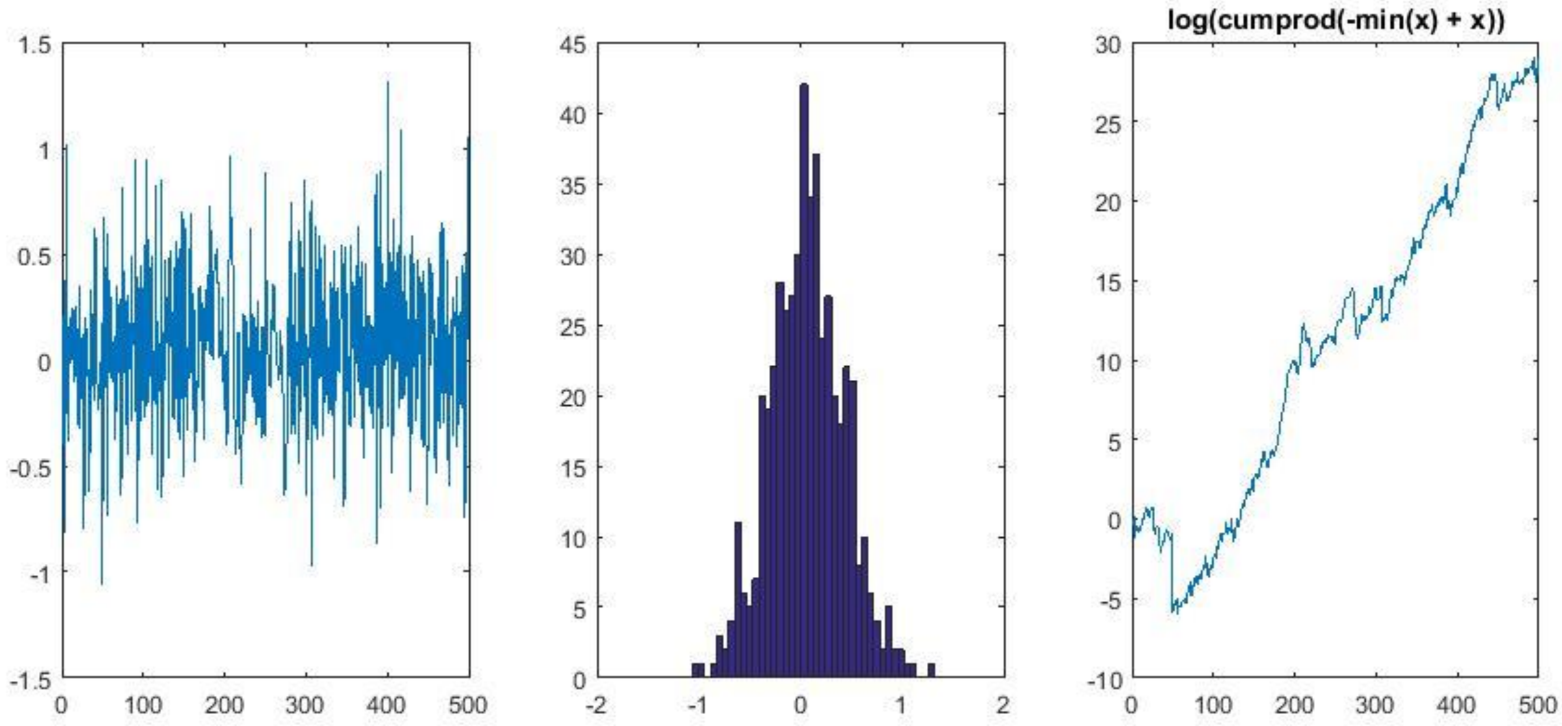
Internal

SIMULATED DATA EXAMPLE

- Simulated 500 points from a 2-state auto regressive HMM with known parameters
  - $\pi = [0.1, 0.9]$
  - $A = \begin{matrix} 0.95 & 0.05 \\ 0.02 & 0.98 \end{matrix}$
  - $p_k(x_t | \Phi_k) = \mu_k + b_k x_{t-1} + N(0, {\sigma_k}^2), k \in \{1, 2\},$
  - $\Phi_1 = \{-0.01, 0.7, 0.1\}, \Phi_2 = \{0.1, -0.3, 0.1\}.$

Internal

SIMULATED DATA EXAMPLE CONT'D



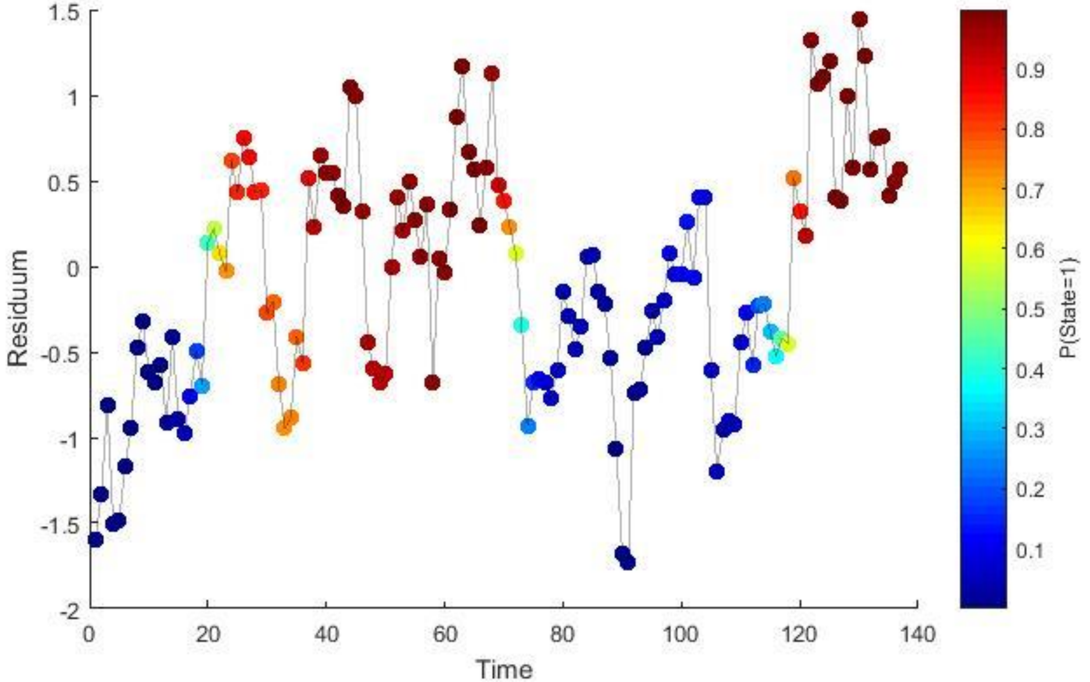
Internal

SIMULATED DATA EXAMPLE CONT'D

- Particle swarm trained 2-state HMM with AR(1) emissions well recovers the parameters with the highest likelihood estimates
  - $\hat{\pi} = [1e - 5,\ 1]$,
  - $\hat{A} = \begin{matrix} 0.9567 & 0.0433 \\ 0.0211 & 0.9789 \end{matrix}$
  - $\widehat{\Phi}_1 = \{-0.0057, 0.7595, 0.0908\}$, $\widehat{\Phi}_2 = \{0.0896, -0.3941, 0.1166\}$.
- Likelihood comparison
  - $logL(\theta|x) = -188$ vs. $logL(\hat{\theta}|x) = -180$
- MAP state classification
  - accuracy ~ 94%, with state1/state2 ratio 186/314.

REAL DATA APPLICATION

- Synthetic time series, where stationarity and mean reversion is assumed and tested, e.g., residuals of cointegrated instruments
  - belief, that the remaining variance is random unexplainable noise, but still might contain a certain structure to exploit.
  - Fit 2-state t-HMM,
  - $\mu_1 = .3, \mu_2 = -.5,$
  - based on HMM yet,
    we don’t short but wait.



Thank you for your attention.

Internal

REFERENCES

- Slides with hidden Markov model theory closely follow C.M.Bishop book *Pattern Recognition and Machine Learning*, 2006
- Figure on slide 5 is taken from J.T.Bryson, Xin Jin and S.K.Agrawal paper *Optimal Design of Cable-Driven Manipulators Using Particle Swarm Optimization*, 2015