Introduction to Algorithmic Trading Strategies
Lecture 2

Hidden Markov Trading Model

Haksun Li
haksun.li@numericalmethod.com
www.numericalmethod.com

Outline

- Carry trade
- Momentum
- Valuation
- CAPM
- Markov chain
- Hidden Markov model

References

- Algorithmic Trading: Hidden Markov Models on Foreign Exchange Data. Patrik Idvall, Conny Jonsson. University essay from Linköpings universitet/Matematiska institutionen; Linköpings universitet/Matematiska institutionen. 2008.
- A tutorial on hidden Markov models and selected applications in speech recognition. Rabiner, L.R. Proceedings of the IEEE, vol 77 Issue 2, Feb 1989.

FX Market

- FX is the largest and most liquid of all financial markets – multiple trillions a day.
- FX is an OTC market, no central exchanges.
- The major players are:
  - Central banks
  - Investment and commercial banks
  - Non-bank financial institutions
  - Commercial companies
  - Retails

Electronic Markets

- Reuters
- EBS (Electronic Broking Service)
- Currenex
- FXCM
- FXall
- Hotspot
- Lava FX

Fees

- Brokerage
- Transaction, e.g., bid-ask

Basic Strategies

- Carry trade
- Momentum
- Valuation

Carry Trade

- Capture the difference between the rates of two currencies.
- Borrow a currency with low interest rate.
- Buy another currency with higher interest rate.
- Take leverage, e.g., 10:1.
- Risk: FX exchange rate goes against the trade.
- Popular trades: JPY vs. USD, USD vs. AUD
- Worked until 2008.

Momentum

- FX tends to trend.
  - Long when it goes up.
  - Short when it goes down.
- Irrational traders
- Slow digestion of information among disparate participants

Purchasing Power Parity

- McDonald's hamburger as a currency.
- The price of a burger in the USA = the price of a burger in Europe
- E.g., USD1.25/burger = EUR1/burger
  - EURUSD = 1.25

FX Index

- Deutsche Bank Currency Return (DBCR) Index
- A combination of
  - Carry trade
  - Momentum
  - Valuation

CAPM

- Individual expected excess return is proportional to the market expected excess return.
- $E(r_i) - r_f = \beta_f\left[E(r_M) - r_f\right]$
  - r_i, r_M are geometric returns
  - r_f is an arithmetic return
- Sensitivity
  - $\beta_f = \frac{Cov(r_i, r_M)}{Var(r_M)}$

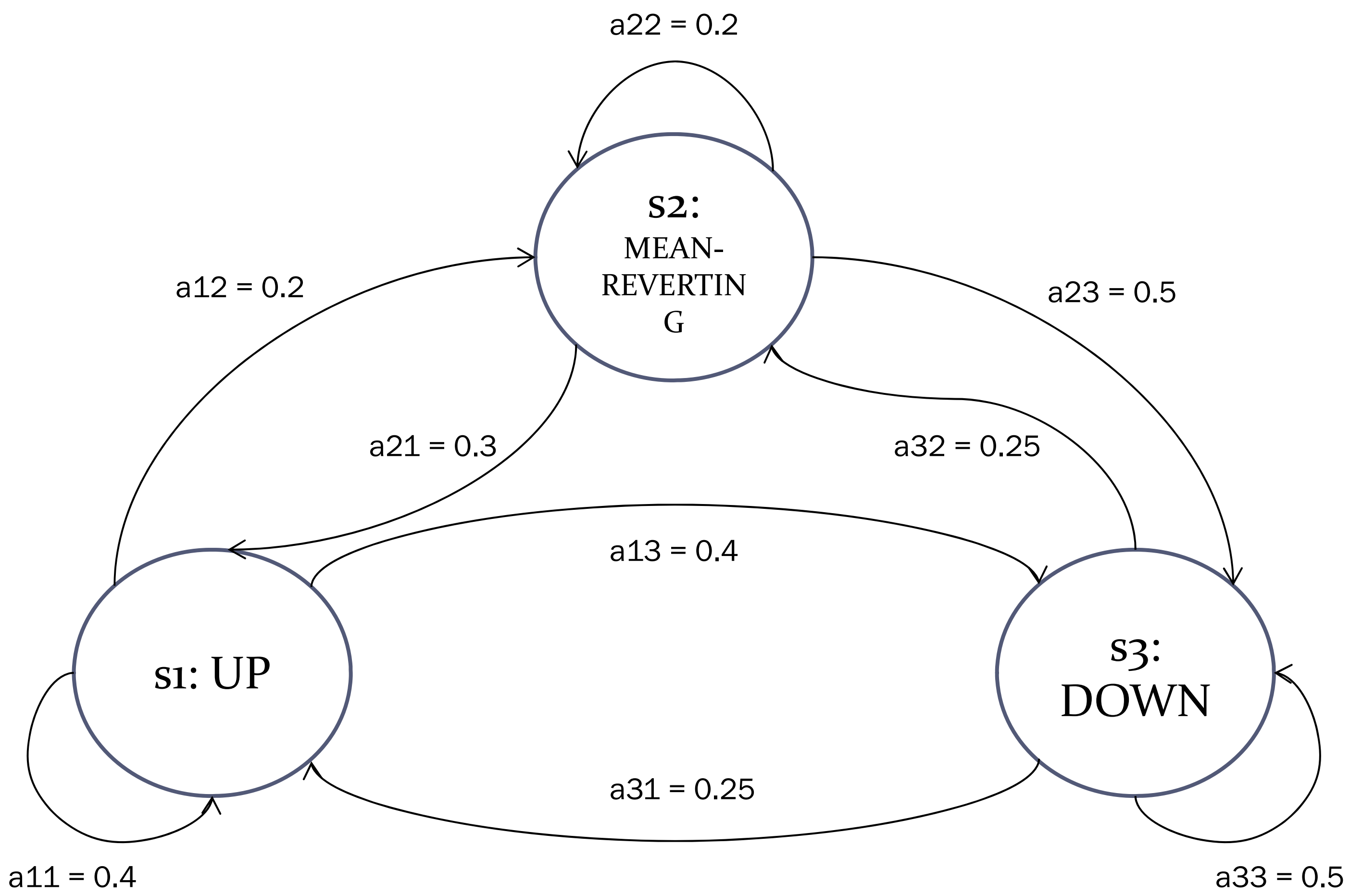
Alpha

- Alpha is the excess return above the expected excess return.
- $\alpha = r - E(r_i)$
- For FX, we usually assume $r_f = 0$.

Bayes Theorem

- Bayes theorem computes the posterior probability of a hypothesis H after evidence E is observed in terms of
  - the prior probability, $P(H)$
  - the prior probability of E, $P(E)$
  - the conditional probability of $P(E|H)$
- $P(H|E) = \frac{P(E|H)P(H)}{P(E)}$

Markov Chain



Example: State Probability

- What is the probability of observing
  - $\Omega = \{s_3, s_1, s_1, s_1\}$
- $P(\Omega|\text{Model}) = \text{P}(s_3, s_1, s_1, s_1|\text{Model})$
- $= \text{P}(s_3|\text{Model}) \times \text{P}(s_1|s_3,\text{Model}) \times \text{P}(s_1|s_1,\text{Model}) \times \text{P}(s_1|s_1,\text{Model})$
- $= 1 \times 0.25 \times 0.4 \times 0.4$
- $= 0.04$

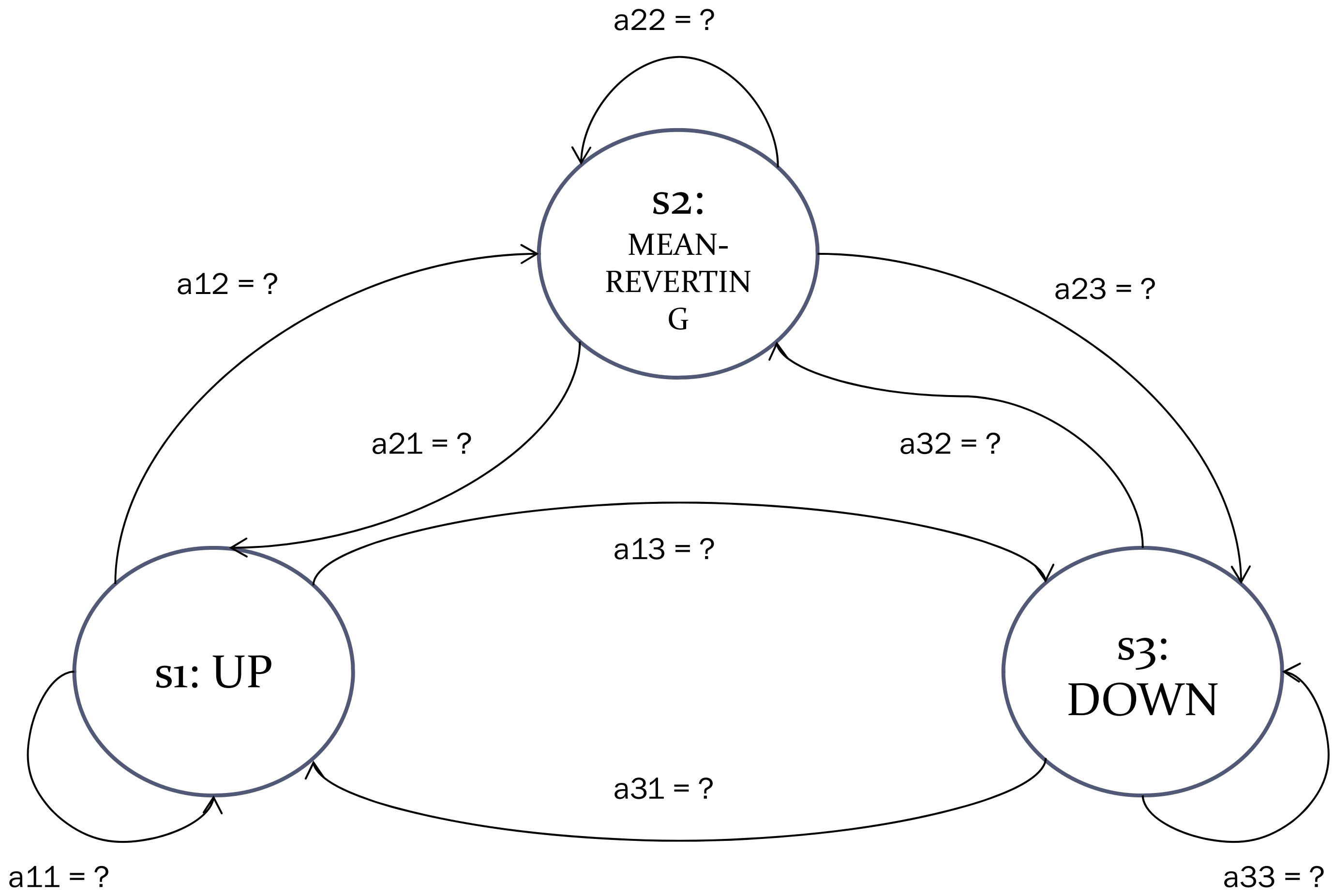
Markov Property

- Given the current information available at time $(t-1)$, the history, e.g., path, is irrelevant.
- $P(q_t|q_{t-1},\cdots,q_1) = P(q_t|q_{t-1})$
- Consistent with the weak form of the efficient market hypothesis.

Hidden Markov Chain

- Only observations are observable (duh).
- World states may not be known (hidden).
  - We want to model the hidden states as a Markov Chain.
- Two assumptions:
  - Markov property
  - $P(\omega_t | q_{t-1}, \cdots, q_1, \omega_{t-1}, \cdots, \omega_1) = P(\omega_t | q_t)$

Markov Chain



Problems

- Likelihood
  - Given the parameters, λ, and an observation sequence, Ω, compute $P(\Omega|\lambda)$.
- Decoding
  - Given the parameters, λ, and an observation sequence, Ω, determine the best hidden sequence Q.
- Learning
  - Given an observation sequence, Ω, and HMM structure, learn λ.

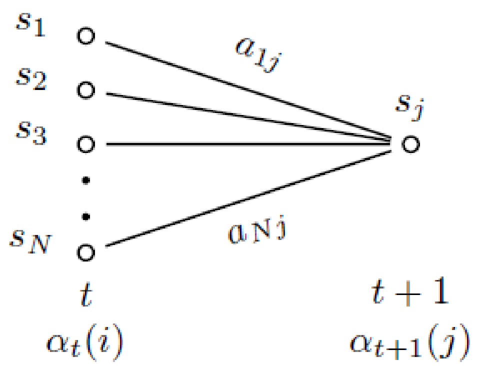
Likelihood Solutions

Likelihood By Enumeration

- $P(\Omega|\lambda) = \sum_{\{q\}'s} P(\Omega, Q|\lambda)$
- $= \sum_{\{q\}'s} P(\Omega|Q, \lambda) \times P(Q|\lambda)$
- $P(\Omega|Q, \lambda) = \prod_{t=1}^{T} P(\omega_t|q_t, \lambda)$
- $P(Q|\lambda) = \pi_{q_1} \times a_{q_1 q_2} \times a_{q_2 q_3} \times \cdots \times a_{q_{T-1} q_T}$
- But… this is not computationally feasible.

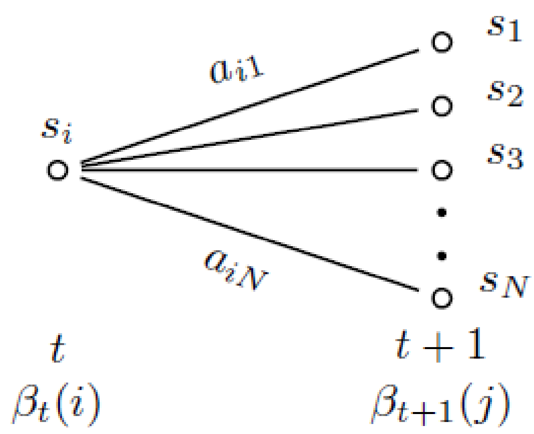
Forward Procedure

- $\alpha_t(i) = P(\omega_1, \omega_2, \cdots, \omega_t, q_t = s_i|\lambda)$
  - the probability of the partial observation sequence until time t and the system in state s_i at time t.
- Initialization
  - $\alpha_1(i) = \pi_i b_i(\omega_1)$
    - b_i: the conditional distribution of ω
- Induction
  - $\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(\omega_{t+1})$
- Termination
  - $P(\Omega|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$, the likelihood



Backward Procedure

- $\beta_t(i) = P(\omega_{t+1}, \omega_{t+2}, \cdots, \omega_T | q_t = s_i, \lambda)$
  - the probability of the system in state s_i at time t, and the partial observations from then onward till time t
- Initialization
  - $\beta_T(i) = 1$
- Induction
  - $\beta_t(i) = \sum_{j=1}^{N} a_{ij}\, b_j(\omega_{t+1})\beta_{t+1}(j)$



Decoding Solutions

Decoding Solutions

- Given the observations and model, the probability of the system in state s_i is:
- $\gamma_t(i) = P(q_t = s_i|\Omega, \lambda)$
- $= \frac{P(q_t=s_i,\Omega|\lambda)}{P(\Omega|\lambda)}$
- $= \frac{\alpha_t(i)\beta_t(i)}{P(\Omega|\lambda)}$
- $= \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)}$

Maximizing The Expected Number Of States

- $q_t = \text{argmax}_{1 \le i \le N}[\gamma_t(i)]$
- This determines the most likely state at every instant, t, without regard to the probability of occurrence of sequences of states.

Viterbi Algorithm

- The maximal probability of the system travelling these states and generating these observations:
- $\delta_t = \max[P(q_1, q_2, \cdots, q_t = s_i, \omega_0, \cdots, \omega_t | \lambda)]$

Viterbi Algorithm

- Initialization
  - $\delta_1(i) = \pi_i b_i(\omega_1)$
- Recursion
  - $\delta_t(j) = \max[\delta_{t-1}(i)a_{ij}]b_j(\omega_t)$
    - the probability of the most probable state sequence for the first t observations, ending in state j
  - $\psi_t(j)= \text{argmax}[\delta_{t-1}(i)a_{ij}]$
    - the state chosen at t
- Termination
  - $P^* = \max[\delta_T(i)]$
  - $q^*= \text{argmax}[\delta_T(i)]$

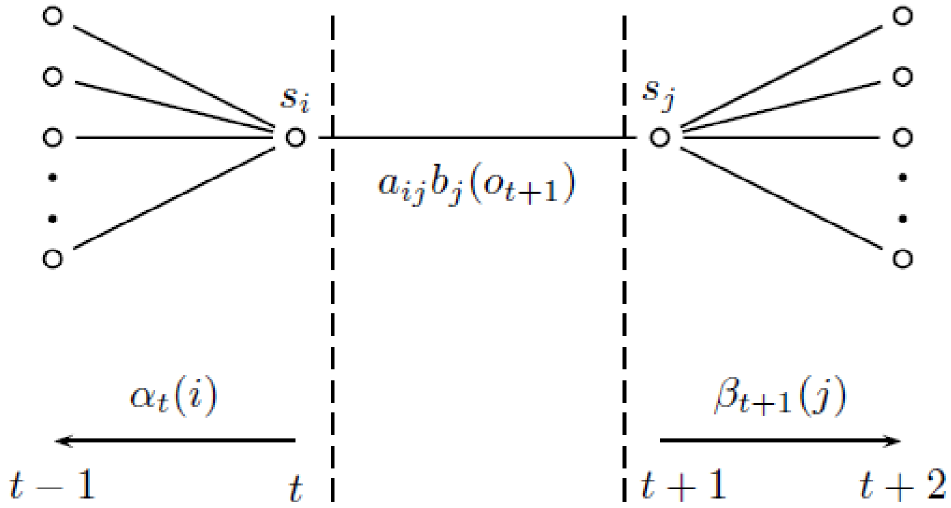
Learning Solutions

As A Maximization Problem

- Our objective is to find λ that maximizes $P(\Omega|\lambda)$.
- For any given λ, we can compute $P(\Omega|\lambda)$.
- Then solve a maximization problem.
- Algorithm: Nelder-Mead.

Baum-Welch

- the probability of being in state s_i at time t, and state s_j at time $(t+1)$, given the model and the observation sequence
- $\xi_t(i,j) = P\left(q_t = s_i, q_{t+1} = s_j | \Omega, \lambda\right)$



Xi

- $\xi_t(i,j) = P\left(q_t = s_i, q_{t+1} = s_j | \Omega, \lambda\right)$
- $= \frac{P\left(q_t = s_i, q_{t+1} = s_j, \Omega | \lambda\right)}{P(\ \Omega|\lambda)}$
- $= \frac{\alpha_t(i) a_{ij} b_j(\omega_{t+1}) \beta_{t+1}(j)}{P(\ \Omega|\lambda)}$
- $\gamma_t(i) = P(q_t = s_i\ |\Omega, \lambda)$
- $= \sum_{j=1}^{N} \xi_t(i,j)$

Estimation Equation

- By summing up over time,
- $\gamma_t(i)$ ~ the number of times s_i is visited
- $\xi_t(i,j)$ ~ the number of times the system goes from state s_i to state s_j
- Thus, the parameters λ are:
  - $\overline{\pi_i} = \gamma_1(i)$, initial state probabilities
  - $\overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$, transition probabilities
  - $\overline{b_j(v_k)} = \frac{\sum_{t=1,\omega_t=v_k}^{T-1} \gamma_t(j)}{\sum_{t=1}^{T-1} \gamma_t(j)}$, conditional probabilities

Estimation Procedure

- Guess what λ is.
- Compute $\overline{\lambda_i}$ using the estimation equations.
- Practically, we can estimate the initial λ by Nelder-Mead to get "closer" to the solution.

Conditional Probabilities

- Our formulation so far assumes discrete conditional probabilities.
- The formulations that take other probability density functions are similar.
  - But the computations are more complicated, and the solutions may not even be analytical, e.g., t-distribution.

Heavy Tail Distributions

- t-distribution
- Gaussian Mixture Model
  - a weighted sum of Normal distributions

Trading Ideas

- Compute the next state.
- Compute the expected return.
- Long (short) when expected return > (<) 0.
- Long (short) when expected return > (<) c.
  - c = the transaction costs
- Any other ideas?

Experiment Setup

- EURUSD daily prices from 2003 to 2006.
- 6 unknown factors.
- Λ is estimated on a rolling basis.
- Evaluations:
  - Hypothesis testing
  - Sharpe ratio
  - VaR
  - Max drawdown
  - alpha

Best Discrete Case

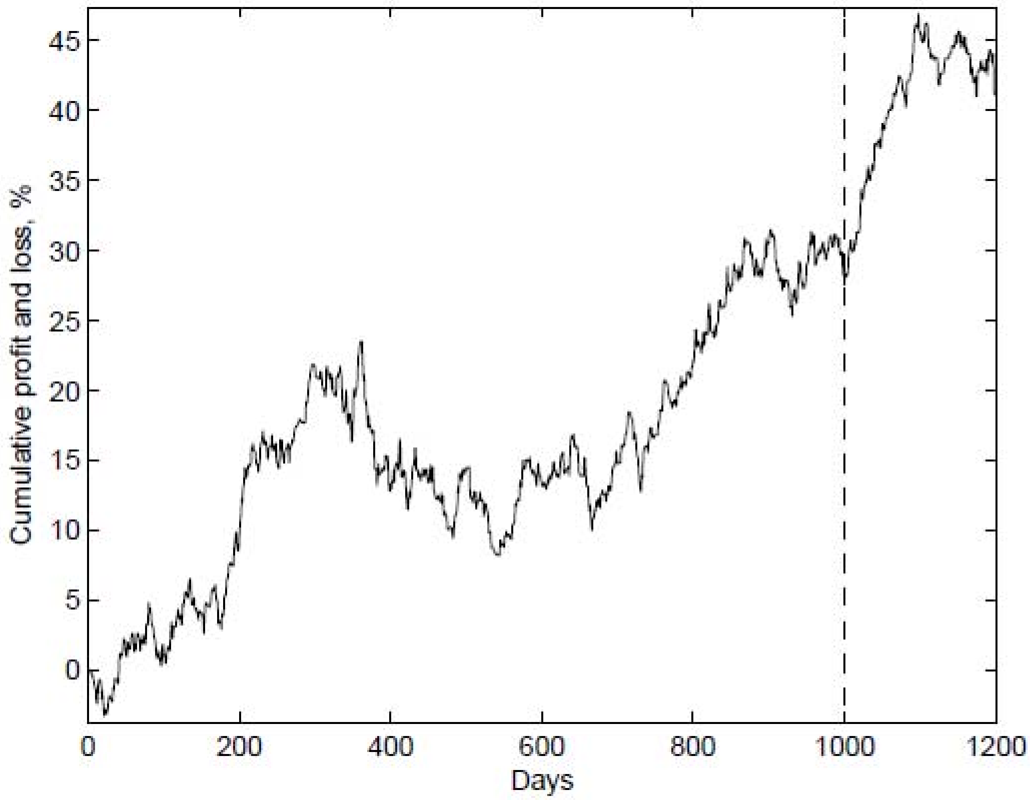


Figure 4.2: Using only the cross as input data with a 30 days window and 3 states.

Best Continuous Case

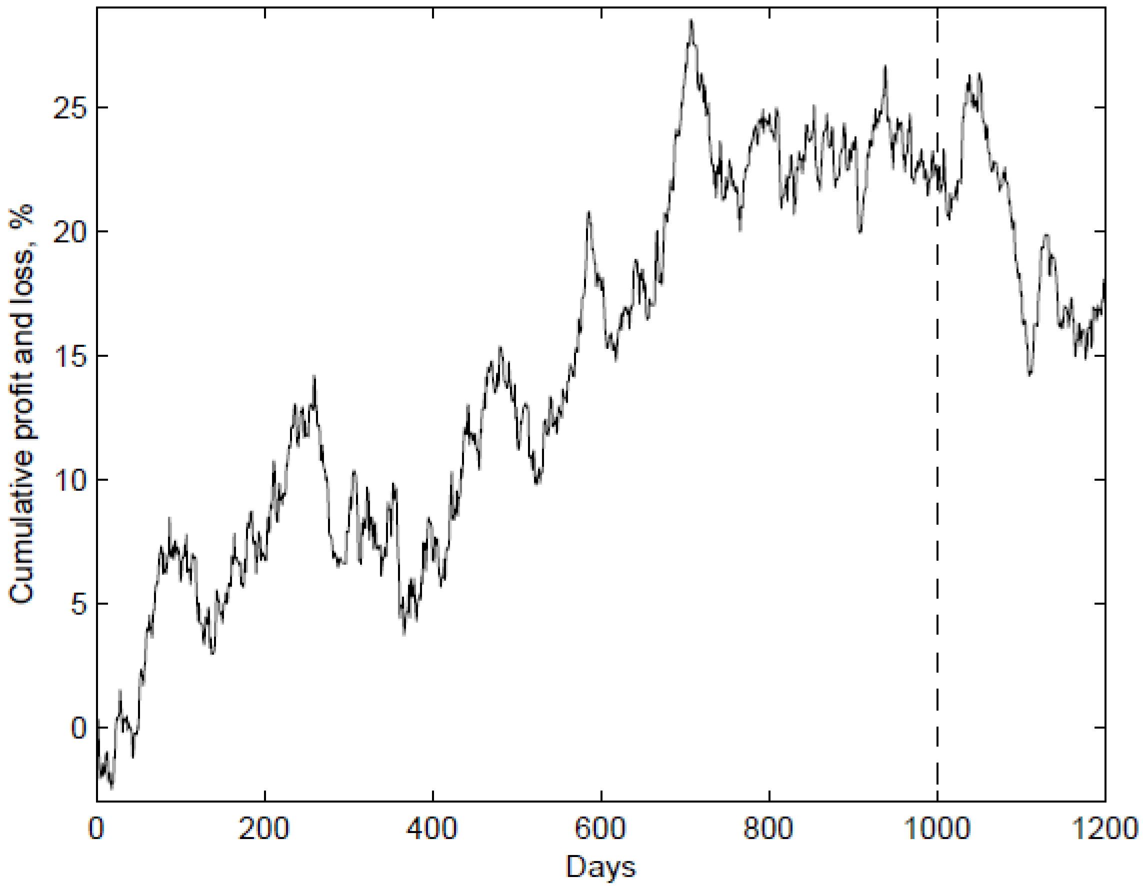


Figure 4.8: Using only the cross as input data with a 20 days window, 4 states and 2 mixture components.

Results

- More data (the 6 factors) do not always help (esp. for the discrete case).
- Parameters unstable.

TODOs

- How can we improve the HMM model(s)? Ideas?