Hidden Markov Models with Stan

StanCon 2018, Asilomar, CA USA

Michael Weylandt

2018-01-10

Dept. of Statistics, Rice University

ldamiano0 Luis Damiano luisdamiano

HMMs in Stan? Absolutely!

Posted by Bob Carpenter on 7 February 2017, 2:20 pm

I was having a conversation with Andrew that went like this yesterday:

Andrew:

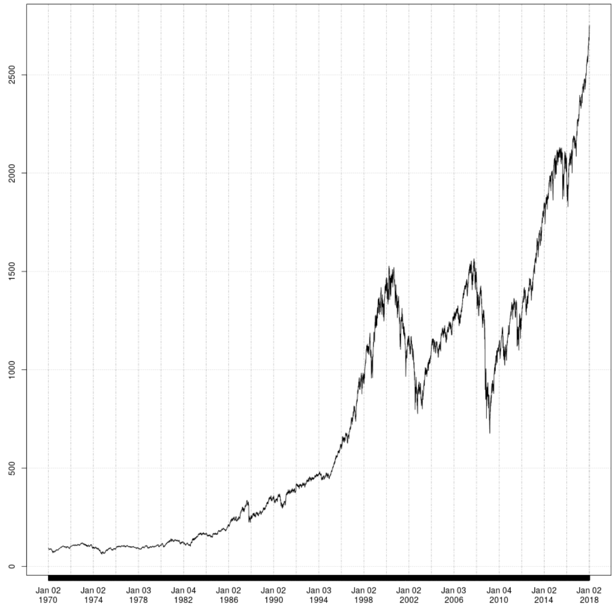
> Hey, someone's giving a talk today on HMMs (that someone was Yang Chen, who was giving a talk based on her *JASA* paper Analyzing single-molecule protein transportation experiments via hierarchical hidden Markov models).
>
> Maybe we should add some specialized discrete modules to Stan so we can fit HMMs. Word on the street is that Stan can't fit models with discrete parameters.

Me:

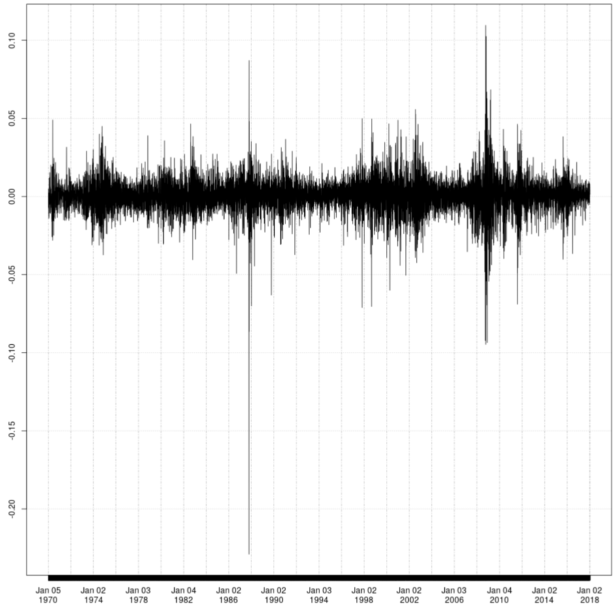
> Uh, we can already fit HMMs in Stan. There's a section in the manual that explains how (section 9.6, Hidden Markov Models).


PUBLIC SERVICE
ANNOUNCEMENT

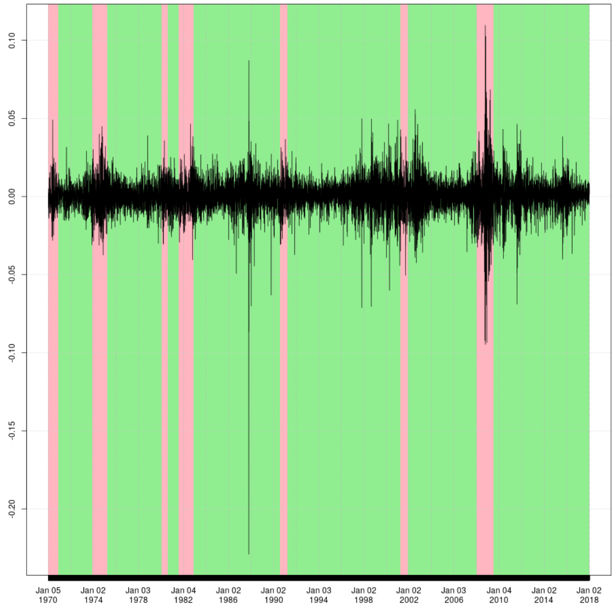
Hidden Markov Models



0
500
1000
1500
2000
2500
Jan 02 1970
Jan 02 1974
Jan 03 1978
Jan 04 1982
Jan 02 1986
Jan 02 1990
Jan 03 1994
Jan 02 1998
Jan 02 2002
Jan 03 2006
Jan 04 2010
Jan 02 2014
Jan 02 2018



0.10
0.05
0.00
-0.05
-0.10
-0.15
-0.20
Jan 05 1970
Jan 02 1974
Jan 03 1978
Jan 04 1982
Jan 02 1986
Jan 02 1990
Jan 03 1994
Jan 02 1998
Jan 02 2002
Jan 03 2006
Jan 04 2010
Jan 02 2014
Jan 02 2018

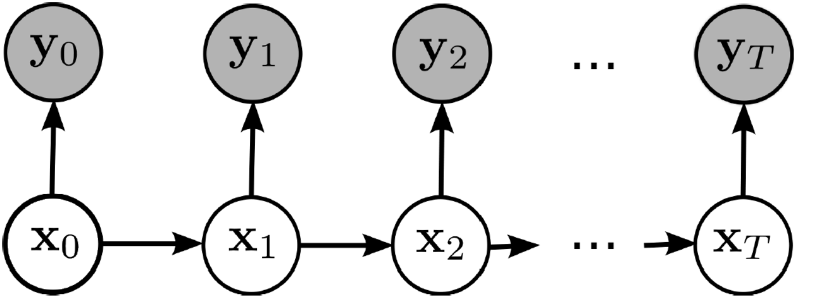

0.10
0.05
0.00
-0.05
-0.10
-0.15
-0.20
Jan 05 1970
Jan 02 1974
Jan 03 1978
Jan 04 1982
Jan 02 1986
Jan 02 1990
Jan 03 1994
Jan 02 1998
Jan 02 2002
Jan 03 2006
Jan 04 2010
Jan 02 2014
Jan 02 2018

Complex dynamics can be (mostly) captured by simpler underlying structure

Suppose we have a sequence of observations $\{Y_t\}$ with complex dynamics

Suppose we have a sequence of observations $\{Y_t\}$ with complex dynamics

A *Hidden Markov Model* assumes that $\{Y_t\}$ is driven by an unobserved (hidden) sequence $\{X_t\}$ which has Markovian dynamics



Formally, we have:

- Y_t depends only on X_t: $\mathrm{Law}[Y_t|Y_1, \ldots, Y_{t-1}, X_1, \ldots, X_t] = \mathrm{Law}[Y_t|X_t]$
- X_t is Markovian: $\mathrm{Law}[X_t|X_1, \ldots, X_{t-1}] = \mathrm{Law}[X_t|X_{t-1}]$

Formally, we have:

- Y_t depends only on X_t: $\text{Law}[Y_t | Y_1, \ldots, Y_{t-1}, X_1, \ldots, X_t] = \text{Law}[Y_t | X_t]$
- X_t is Markovian: $\text{Law}[X_t | X_1, \ldots, X_{t-1}] = \text{Law}[X_t | X_{t-1}]$

Sweet spot of tractability (underlying Markovian dynamics) and expressive power / interpretability

Important: $\{Y_t\}$ is **not** necessarily Markovian

Formally, we have:

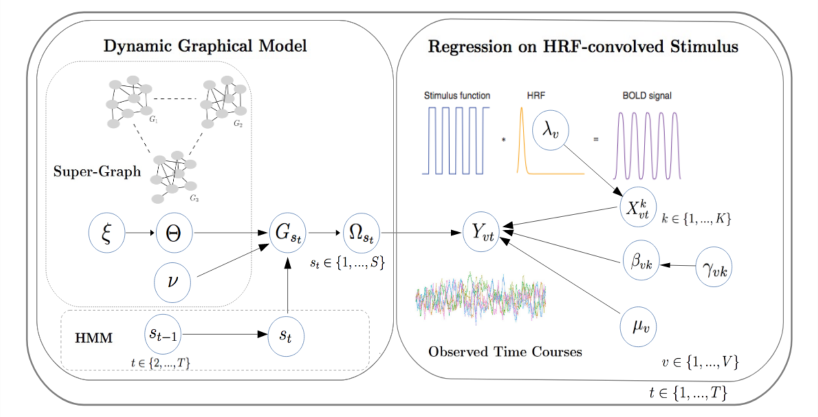
- Y_t depends only on X_t: $\mathrm{Law}[Y_t|Y_1,\dots,Y_{t-1},X_1,\dots,X_t] = \mathrm{Law}[Y_t|X_t]$
- X_t is Markovian: $\mathrm{Law}[X_t|X_1,\dots,X_{t-1}] = \mathrm{Law}[X_t|X_{t-1}]$

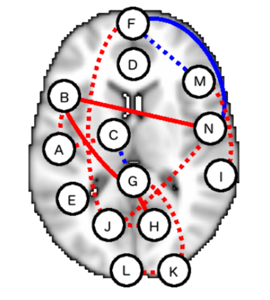
Sweet spot of tractability (underlying Markovian dynamics) and expressive power / interpretability

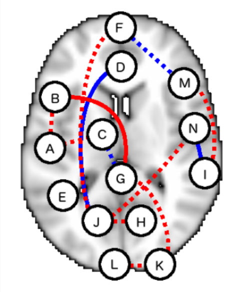
Important: $\{Y_t\}$ is **not** necessarily Markovian

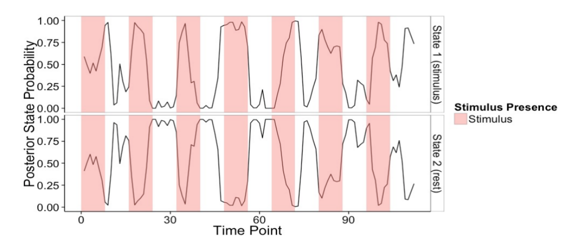
Can embed in larger models without loosing HMM dynamics:

- Direct or indirect observations of $\{X_t\}$
- $\{Y_t\}$ not observed directly


Dynamic Graphical Model
Super-Graph
G_1
G_2
G_3
ξ
Θ
ν
G_{s_t}
Ω_{s_t}
$s_t \in \{1, ..., S\}$
HMM
s_{t-1}
s_t
$t \in \{2, ..., T\}$
Regression on HRF-convolved Stimulus
Stimulus function
HRF
BOLD signal
*
=
λ_v
X^k_{vt}
$k \in \{1, ..., K\}$
Y_{vt}
β_{vk}
γ_{vk}
μ_v
Observed Time Courses
$v \in \{1, ..., V\}$
$t \in \{1, ..., T\}$


F
D
M
B
N
C
A
I
G
E
J
H
L
K


F
D
M
B
N
C
A
I
G
E
J
H
L
K


Posterior State Probability
1.00
0.75
0.50
0.25
0.00
State 1 (stimulus)
State 2 (rest)
0
30
60
90
Time Point
Stimulus Presence
Stimulus

Bayesian Inference in HMMs

Bayesian Inference

Bayesian inference is “easy:”

- Formalize priors and likelihood
- `Stan` happens
- Posterior inference

Bayesian Inference

Bayesian inference is "easy:"

- Formalize priors and likelihood
- `Stan` happens
- Posterior inference

Difficulty is in the likelihood:

- Many unobserved variables
- $L = \int_{\{x_t\} \in \mathcal{X}^T} \prod_{i=1}^{T} p(y_t | x_t)$

Bayesian Inference

Bayesian inference is "easy:"

- Formalize priors and likelihood
- `Stan` happens
- Posterior inference

Difficulty is in the likelihood:

- Many unobserved variables
- $L = \int_{\{x_t\} \in \mathcal{X}^T} \prod_{i=1}^{T} p(y_t | x_t)$

In general, essentially intractable; two commonly-used special cases:

- Everything is linear and Gaussian $\implies$ state-space models, Kalman Filtering
- $\mathcal{X}$ is finite – integral becomes a sum


A LONG TIME AGO IN A GALAXY FAR FAR AWAY...
STAR WARS
A NEW HOPE
TWENTIETH CENTURY FOX LUCASFILM LTD. STAR WARS EPISODE IV A NEW HOPE
MARK HAMILL HARRISON FORD CARRIE FISHER
PETER CUSHING ALEC GUINNESS
GEORGE LUCAS GARY KURTZ JOHN WILLIAMS
DOLBY DIGITAL
THX
www.STAR WARS.com

When $\mathcal{X}$ is finite, we can calculate the likelihood using the forward algorithm

A New Hope: The Forward Algorithm

When $\mathcal{X}$ is finite, we can calculate the likelihood using the forward algorithm

Dynamic programming algorithm which keeps computations tractable – linear instead of exponential in T

A New Hope: The Forward Algorithm

When $\mathcal{X}$ is finite, we can calculate the likelihood using the forward algorithm

Dynamic programming algorithm which keeps computations tractable – linear instead of exponential in T

Key idea: Markov property allows us to only consider previous state

A New Hope: The Forward Algorithm

When $\mathcal{X}$ is finite, we can calculate the likelihood using the forward algorithm

Dynamic programming algorithm which keeps computations tractable – linear instead of exponential in T

Key idea: Markov property allows us to only consider previous state

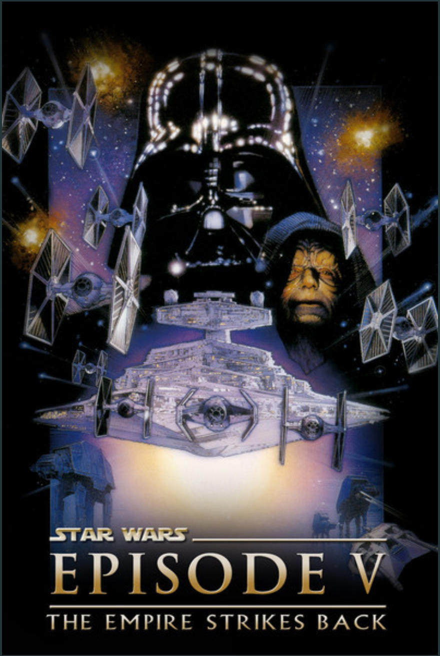
For each state $1, \ldots, K$, we sum probability of coming from each of $1, \ldots, K$ states at previous observation: $\mathcal{O}(K \times K)$

"Roll forward" through all T observations to get $\mathcal{O}(TK^2)$

Likelihood falls out of normalizability requirements

Summary of the hidden quantities and their corresponding inference algorithm.

Name	Hidden Quantity	Availability at	Algorithm	Complexity
Filtering	$p(z_t \mid \mathbf{y}_{1:t})$	t (online)	Forward	$O(K^2T)$ $O(KT)$ if left-to-right
Smoothing	$p(z_t \mid \mathbf{y}_{1:T})$	T (offline)	Forward-backward	$O(K^2T)$ $O(KT)$ if left-to-right
Fixed lag smoothing	$p(z_{t-\ell} \mid \mathbf{y}_{1:t}), \ell \geq 1$	$t + \ell$ (lagged)	Forward-backward	$O(K^2T)$ $O(KT)$ if left-to-right
State prediction	$p(z_{t+h} \mid \mathbf{y}_{1:t}), h \geq 1$	t		
Observation prediction	$p(y_{t+h} \mid \mathbf{y}_{1:t}), h \geq 1$	t		
MAP Estimation	$\text{argmax}_{\mathbf{z}_{1:T}}\, p(\mathbf{z}_{1:T} \mid \mathbf{y}_{1:T})$	T	Viterbi decoding	$O(K^2T)$
Log likelihood	$p(\mathbf{y}_{1:T})$	T	Forward	$O(K^2T)$ $O(KT)$ if left-to-right


STAR WARS
EPISODE V
THE EMPIRE STRIKES BACK

Posterior distribution of HMMs can be tricky to sample from:

The Posterior Strikes Back: Multimodality and Identifiability

Posterior distribution of HMMs can be tricky to sample from:

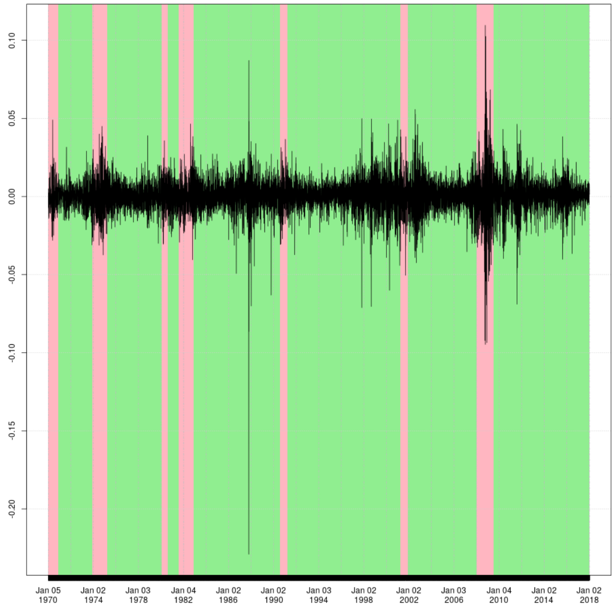
- Identifiability:
  - Strong identifiability (symmetry breaking) – use the `ordered` or `positive_ordered` type
  - Weak identifiability – use a strong prior and good initialization

The Posterior Strikes Back: Multimodality and Identifiability

Posterior distribution of HMMs can be tricky to sample from:

- Identifiability:
    - Strong identifiability (symmetry breaking) – use the `ordered` or `positive_ordered` type
    - Weak identifiability – use a strong prior and good initialization
- Multimodality:
    - Use strong priors or the semi-supervised strategy discussed in the Stan manual

A Regime Switching GARCH Model



0.10
0.05
0.00
-0.05
-0.10
-0.15
-0.20
Jan 05 1970
Jan 02 1974
Jan 03 1978
Jan 04 1982
Jan 02 1986
Jan 02 1990
Jan 03 1994
Jan 02 1998
Jan 02 2002
Jan 03 2006
Jan 04 2010
Jan 02 2014
Jan 02 2018

GARCH processes are widely used in finance to model volatility (standard deviation) of stock returns – volatility clustering and heavy(-ish) tails

Regime Switching GARCH

GARCH processes are widely used in finance to model volatility (standard deviation) of stock returns – volatility clustering and heavy(-ish) tails

"Regime Switching" GARCH: multiple GARCH processes with different parameters – return for each day is drawn from a GARCH processes selected by an underlying HMM (Haas *et. al*, J. Fin. Econ. 2004)

Regime Switching GARCH

GARCH processes are widely used in finance to model volatility (standard deviation) of stock returns – volatility clustering and heavy(-ish) tails

"Regime Switching" GARCH: multiple GARCH processes with different parameters – return for each day is drawn from a GARCH processes selected by an underlying HMM (Haas *et. al*, J. Fin. Econ. 2004)

$$Y_t | X_t = k \sim \text{GARCH}(\alpha_k, \beta_k) \tag{GARCH}$$

$$X_t | X_{t-1} = k \sim \text{Categorical}(\mathbf{p}_k) \tag{HMM}$$

Stan Implementation

luisdamiano/stancon2018/stan/hmm_garch.stan

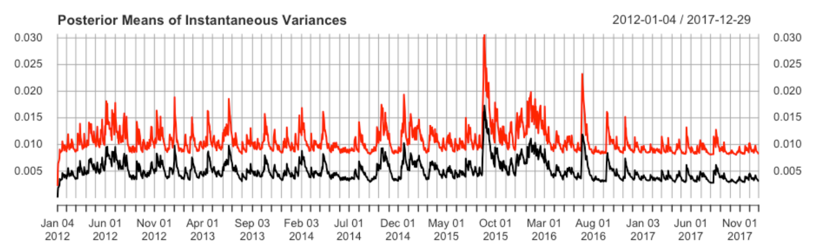
```
transformed parameters {
...

  // GARCH Component
  // ---------------

  // Initialize at unconditional variances
  sigma_t[1, 1] = alpha0[1] / (1 - alpha1[1] - beta1[1]); // Low-vol
  sigma_t[1, 2] = alpha0[2] / (1 - alpha1[2] - beta1[2]); // High-vol

  // GARCH dynamics rolling forward
  for(t in 2:T){
    for(i in 1:2){
      sigma_t[t, i] = sqrt(alpha0[i] +
                           alpha1[i] * pow(y[t-1], 2) +
                           beta1[i] * pow(sigma_t[t-1, i], 2));
    }
  }
}
```

Stan Implementation

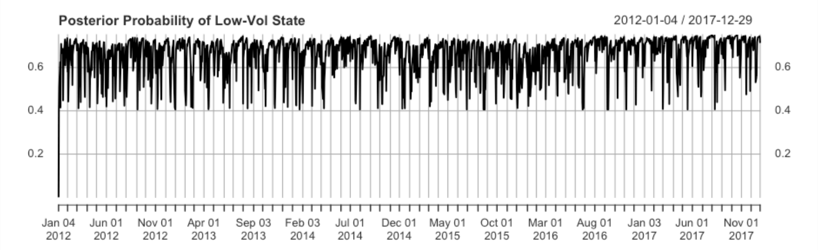
luisdamiano/stancon2018/stan/hmm_garch.stan

```
transformed parameters {
...
  // HMM Component
  // -------------
  // Calculate log p(state at t = j | history up to t) recursively
  // Markov property allows us to do one-step updates

  real accumulator[2];

  // Assume initial equal distribution among two states
  // Better model would be to weight by HMM stationary distribution
  log_alpha[1, 1] = log(0.5) + normal_lpdf(y[1] | 0, sigma_t[1, 1]);
  log_alpha[1, 2] = log(0.5) + normal_lpdf(y[1] | 0, sigma_t[1, 2]);

  for(t in 2:T){
    for(j in 1:2) { // Current state
      for(i in 1:2) { // Previous state
        accumulator[i] = log_alpha[t-1, i] + // Probability from previous obs
                         log(A[i, j]) + // Transition probability
                         // (Local) likelihood / evidence for given state
                         normal_lpdf(y[t] | 0, sigma_t[t-1, i]);
      }
      log_alpha[t, j] = log_sum_exp(accumulator);
    }
```

MSGARCH: Results



MSGARCH: Results



"Gain" from using the MLE instead of posterior inference

	Stocks			Indices			Exchange rates		
	QL 1%	QL 5%	wCRPS	QL 1%	QL 5%	wCRPS	QL 1%	QL 5%	wCRPS
Panel A: Markov–switching GARCH models									
GARCH $\mathcal{N}$	-3.65	-3.26	-2.24	-0.33	-0.58	-0.25	-1.33	0.99	-2.02
GARCH sk$\mathcal{N}$	-3.58	-2.93	-0.60	-1.56	-2.33	-1.04	-0.82	-1.24	-1.04
GARCH $\mathcal{S}$	-2.20	-5.78	-5.55	0.77	-0.17	-0.85	-0.78	0.29	0.35
GARCH sk$\mathcal{S}$	-5.04	-6.88	-7.04	1.13	-0.52	-0.58	-1.54	-1.64	-2.98
GJR $\mathcal{N}$	-1.91	-2.66	-3.22	-1.21	-2.95	-2.08	-1.09	-1.38	-3.61
GJR sk$\mathcal{N}$	-1.83	-3.12	-2.06	-1.11	-0.84	-1.40	0.06	-0.32	-1.17
GJR $\mathcal{S}$	-1.07	-3.11	-4.48	-1.29	-1.56	-4.11	-1.75	-2.40	-4.19
GJR sk$\mathcal{S}$	-3.10	-3.90	-5.28	-2.95	-2.02	-3.48	-1.59	-0.38	-2.50

Acknowledgements

10.6. Hidden Markov Models

A hidden Markov model (HMM) generates a sequence of T output variables y_t conditioned on a parallel sequence of latent categorical state variables $z_t \in \{1, \ldots, K\}$. These “hidden” state variables are assumed to form a Markov chain so that z_t is conditionally independent of other variables given z_{t-1}. This Markov chain is parameterized by a transition matrix θ where θ_k is a K-simplex for $k \in \{1, \ldots, K\}$. The probability of transitioning to state z_t from state z_{t-1} is

$$z_t \sim \mathsf{Categorical}(\theta_{z[t-1]}).$$

Thank You

Questions?