Bayesci Yapay Öğrenme (I), Zaman Dizileri (II)

Yapay Öğrenme ve Bilgi İşlemede Yeni Teknikler,

Orta Doğu Teknik Üniversitesi

Instructor: A. Taylan Cemgil, Boğaziçi Universitesi

06.09.2016

Özet

- Giriş
  - Bayes Teoremi,
  - Basit bir Örnek
  - Olasılık Kuramı hatırlatma, olasılık tabloları
  - Bayesci Öğrenme
- Zaman Dizileri
  - Hesaplama Problemleri
  - Saklı Markov Modelleri
- Yaklaşık Çıkarım (Variational Bayes)

Bayes Kuralı

Thomas Bayes (1702-1761)

Bir $\boldsymbol{\lambda}$ parametresi hakkında, $\mathcal{D}$ verisini gördükten **sonraki** bilgimiz veriyi görmeden **önceki** bilgimiz ve verinin bize söylediği bilgnin birleşimidir.

$$p(\lambda|\mathcal{D}) = \frac{p(\mathcal{D}|\lambda)p(\lambda)}{p(\mathcal{D})}$$

$$\text{Sonsal Dağılım} = \frac{\text{Gözlem Modeli} \times \text{Önsel Dağılım}}{\text{Marjinal Olabilirlik}}$$

İki Zar: ‘Kaynak Ayrıştırma’

1. zar λ, 2. zar y

$$\mathcal{D} = \lambda + y$$

$\mathcal{D} = 9$ ise $\lambda =$?

İki Zar

$$\mathcal{D} = \lambda + y = 9$$

$\mathcal{D} = \lambda + y$	$y = 1$	$y = 2$	$y = 3$	$y = 4$	$y = 5$	$y = 6$
$\lambda = 1$	2	3	4	5	6	7
$\lambda = 2$	3	4	5	6	7	8
$\lambda = \mathbf{3}$	4	5	6	7	8	**9**
$\lambda = \mathbf{4}$	5	6	7	8	**9**	10
$\lambda = \mathbf{5}$	6	7	8	**9**	10	11
$\lambda = \mathbf{6}$	7	8	**9**	10	11	12

$$p(\lambda) \to p(\lambda|\mathcal{D}).$$

Gözlem modeli: $p(\mathcal{D}|\lambda)$

“Bürokratik” türetim

$$
\begin{aligned}
p(\lambda) &= \mathcal{C}(\lambda; [\ 1/6 \quad 1/6 \quad 1/6 \quad 1/6 \quad 1/6 \quad 1/6\ ]) \\
p(y) &= \mathcal{C}(y; [\ 1/6 \quad 1/6 \quad 1/6 \quad 1/6 \quad 1/6 \quad 1/6\ ]) \\
p(\mathcal{D}|\lambda, y) &= \delta(\mathcal{D} - (\lambda + y))
\end{aligned}
$$

$$
\begin{aligned}
p(\lambda, y|\mathcal{D}) &= \frac{1}{p(\mathcal{D})} \times p(\mathcal{D}|\lambda, y) \times p(y)p(\lambda) \\
\text{Sonsal} &= \frac{1}{\text{Kanıt}} \times \text{Olabilirlik} \times \text{Önsel}
\end{aligned}
$$

Kronecker delta $\delta(x) = \begin{cases} 1 & x = 0 \\ 0 & x \neq 0 \end{cases}$

Önsel Dağılım

$$p(y)p(\lambda)$$

$p(y) \times p(\lambda)$	$y = 1$	$y = 2$	$y = 3$	$y = 4$	$y = 5$	$y = 6$
$\lambda = 1$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$
$\lambda = 2$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$
$\lambda = 3$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$
$\lambda = 4$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$
$\lambda = 5$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$
$\lambda = 6$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$	$1/36$

- Olasılık $p(\lambda, y)$

Olabilirlik Fonksyonu - Gözlem modeli

$$p(\mathcal{D} = 9|\lambda, y)$$

$p(\mathcal{D} = 9\|\lambda, y)$	$y = 1$	$y = 2$	$y = 3$	$y = 4$	$y = 5$	$y = 6$
$\lambda = 1$	0	0	0	0	0	0
$\lambda = 2$	0	0	0	0	0	0
$\lambda = 3$	0	0	0	0	0	**1**
$\lambda = 4$	0	0	0	0	**1**	0
$\lambda = 5$	0	0	0	**1**	0	0
$\lambda = 6$	0	0	**1**	0	0	0

- Olabilirlik $\neq$ Olasılık. Sedece negatif olmayan bir fonksyon.

Olabilirlik $\times$ Önsel

$$\phi_{\mathcal{D}}(\lambda, y) = p(\mathcal{D} = 9|\lambda, y)p(\lambda)p(y)$$

$p(\mathcal{D} = 9\|\lambda, y)$	$y = 1$	$y = 2$	$y = 3$	$y = 4$	$y = 5$	$y = 6$
$\lambda = 1$	0	0	0	0	0	0
$\lambda = 2$	0	0	0	0	0	0
$\lambda = 3$	0	0	0	0	0	$\mathbf{1/36}$
$\lambda = 4$	0	0	0	0	$\mathbf{1/36}$	0
$\lambda = 5$	0	0	0	$\mathbf{1/36}$	0	0
$\lambda = 6$	0	0	$\mathbf{1/36}$	0	0	0

Marjinal Olabilirlik

$$
\begin{aligned}
p(\mathcal{D}=9) &= \sum_{\lambda,y} p(\mathcal{D}=9|\lambda,y)p(\lambda)p(y) \\
&= 0+0+\cdots+1/36+1/36+1/36+1/36+0+\cdots+0 \\
&= 1/9
\end{aligned}
$$

$p(\mathcal{D}=9\|\lambda,y)$	$y=1$	$y=2$	$y=3$	$y=4$	$y=5$	$y=6$
$\lambda=1$	0	0	0	0	0	0
$\lambda=2$	0	0	0	0	0	0
$\lambda=3$	0	0	0	0	0	$\mathbf{1/36}$
$\lambda=4$	0	0	0	0	$\mathbf{1/36}$	0
$\lambda=5$	0	0	0	$\mathbf{1/36}$	0	0
$\lambda=6$	0	0	$\mathbf{1/36}$	0	0	0

Sonsal Dağılım

$$p(\lambda, y|\mathcal{D}=9) \;\; = \;\; \frac{1}{p(\mathcal{D})} p(\mathcal{D}=9|\lambda, y)p(\lambda)p(y)$$

$p(\mathcal{D}=9\|\lambda, y)$	$y=1$	$y=2$	$y=3$	$y=4$	$y=5$	$y=6$
$\lambda=1$	0	0	0	0	0	0
$\lambda=2$	0	0	0	0	0	0
$\lambda=3$	0	0	0	0	0	$\mathbf{1/4}$
$\lambda=4$	0	0	0	0	$\mathbf{1/4}$	0
$\lambda=5$	0	0	0	$\mathbf{1/4}$	0	0
$\lambda=6$	0	0	$\mathbf{1/4}$	0	0	0

$$1/4 \;\; = \;\; (1/36)/(1/9)$$

Marjinal Sonsal Dağılım

$$p(\lambda|\mathcal{D}) \quad = \quad \sum_y \frac{1}{p(\mathcal{D})} p(\mathcal{D}|\lambda, y)p(\lambda)p(y)$$

	$p(\lambda\|\mathcal{D}=9)$	$y=1$	$y=2$	$y=3$	$y=4$	$y=5$	$y=6$
$\lambda=1$	0	0	0	0	0	0	0
$\lambda=2$	0	0	0	0	0	0	0
$\lambda=3$	$\mathbf{1/4}$	0	0	0	0	0	$1/4$
$\lambda=4$	$\mathbf{1/4}$	0	0	0	0	$1/4$	0
$\lambda=5$	$\mathbf{1/4}$	0	0	0	$1/4$	0	0
$\lambda=6$	$\mathbf{1/4}$	0	0	$1/4$	0	0	0

Orantılıdır $\propto$ notasyonu

$$p(\lambda|\mathcal{D}=9) \quad \propto \quad p(\lambda,\mathcal{D}=9) = \sum_y p(\mathcal{D}=9|\lambda,y)p(\lambda)p(y)$$

	$p(\lambda,\mathcal{D}=9)$	$y=1$	$y=2$	$y=3$	$y=4$	$y=5$	$y=6$
$\lambda=1$	0	0	0	0	0	0	0
$\lambda=2$	0	0	0	0	0	0	0
$\lambda=3$	$1/36$	0	0	0	0	0	$\mathbf{1/36}$
$\lambda=4$	$1/36$	0	0	0	0	$\mathbf{1/36}$	0
$\lambda=5$	$1/36$	0	0	0	$\mathbf{1/36}$	0	0
$\lambda=6$	$1/36$	0	0	$\mathbf{1/36}$	0	0	0

Model Seçim Örneği

Bilinmeyen sayıda zar atılıyor: $\lambda_1, \lambda_2, \ldots, \lambda_n,$

$$\mathcal{D} = \sum_{i=1}^{n} \lambda_i$$

$\mathcal{D} = 9$ ise kaç zar atıldı?

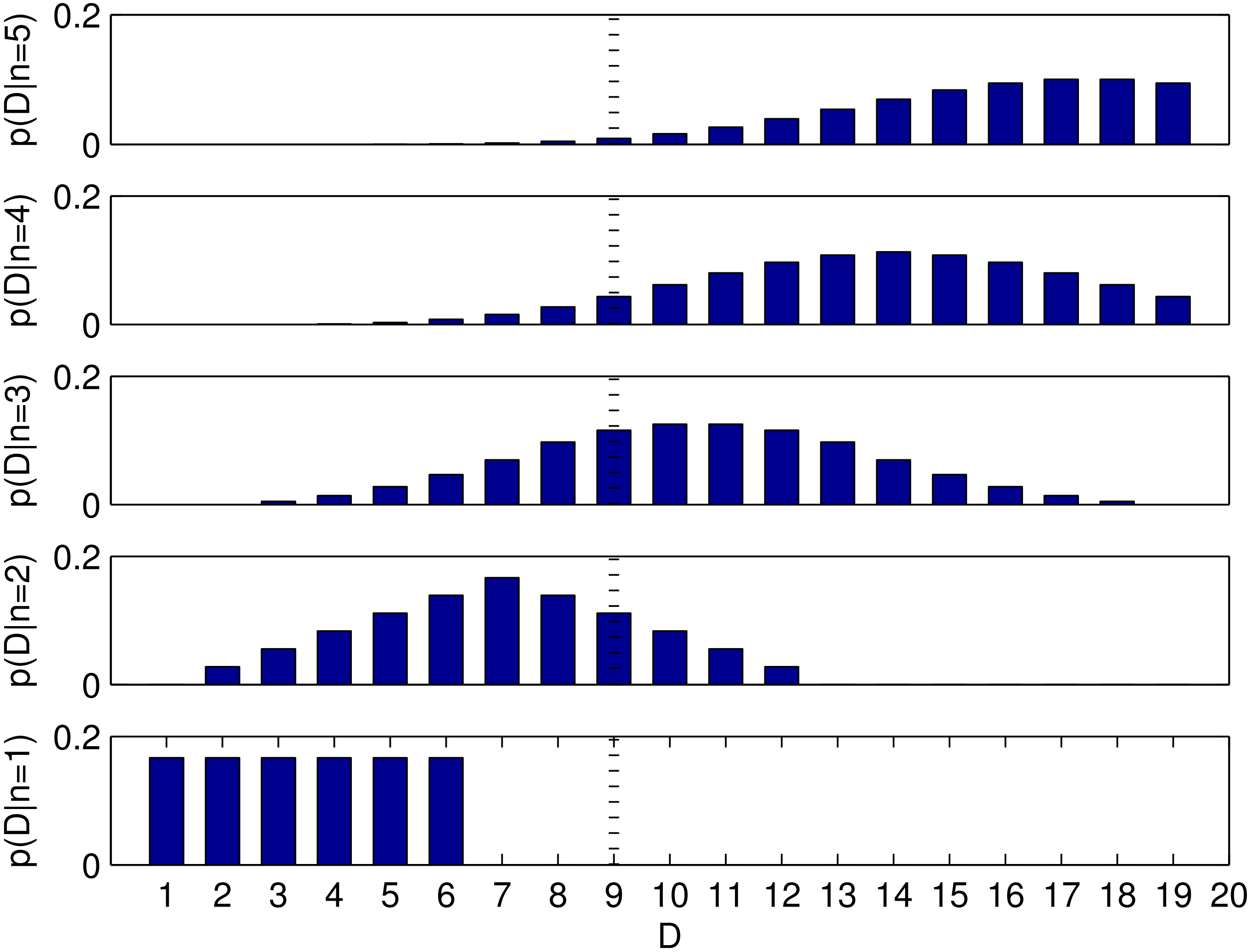
$$p(n) \propto 1$$

Model Seçimi

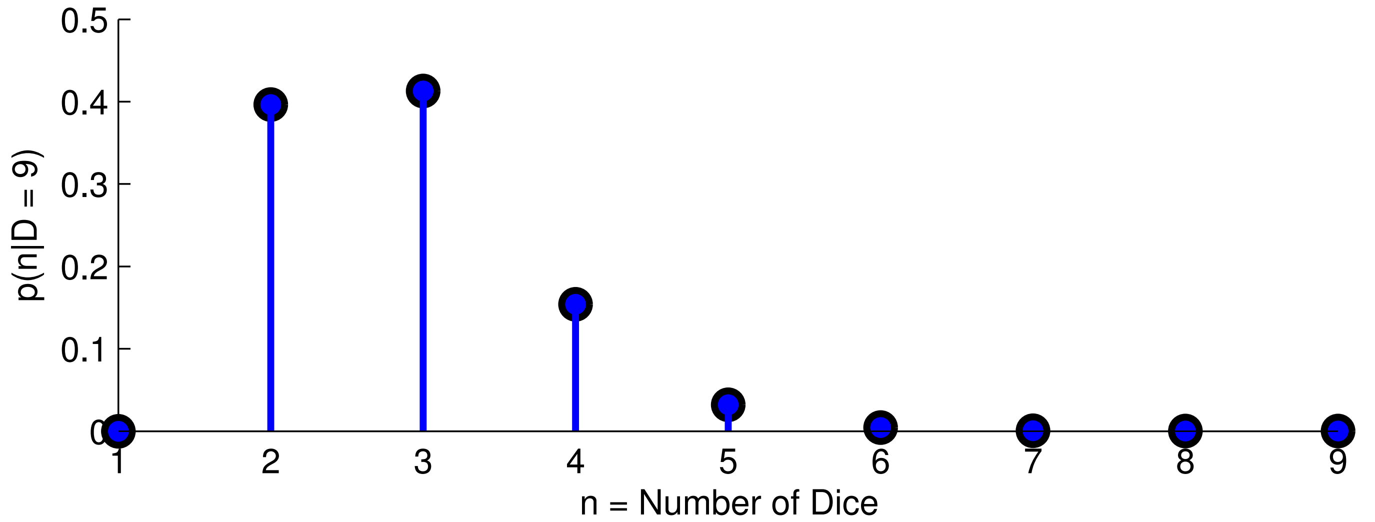
$$p(n|\mathcal{D}=9) = \frac{p(\mathcal{D}=9|n)p(n)}{p(\mathcal{D})} \propto p(\mathcal{D}=9|n)$$

$$\begin{aligned}
p(\mathcal{D}|n=1) &= \sum_{\lambda_1} p(\mathcal{D}|\lambda_1)p(\lambda_1) \\
p(\mathcal{D}|n=2) &= \sum_{\lambda_1}\sum_{\lambda_2} p(\mathcal{D}|\lambda_1,\lambda_2)p(\lambda_1)p(\lambda_2) \\
\dots \\
p(\mathcal{D}|n=n') &= \sum_{\lambda_1,\dots,\lambda_{n'}} p(\mathcal{D}|\lambda_1,\dots,\lambda_{n'}) \prod_{i=1}^{n'} p(\lambda_i)
\end{aligned}$$

$$p(\mathcal{D}|n) = \sum_{\boldsymbol{\lambda}} p(\mathcal{D}|\boldsymbol{\lambda}, n) p(\boldsymbol{\lambda}|n)$$



Model Seçimi



- Sezgi: Karmaşık modellerde olasılık daha büyük bir alana yayılır, gözlemlenen tek bir olayın olabilirliği düşer.

- Bayesci çıkarım “basit modelleri” tercih eder – Occam’s razor

- Bütün parametreler üzerinden toplam (tümlev) hesabı

Olasılıksal Yaklaşım

- Ne çözelim : Modelleme
  - Zanaat
- Nasıl çözelim : Çıkarım Algoritması
  - Mekanik-Otomatik (Teoride! Pratikte hep değil)
  - Genel

Olasılık Kuramı

- Pascal ve Fermat arasındaki mektuplaşma (Soylu ve kumarbaz bey de Meré)
- 1930'lar Aksyomatik gelişim (Reichenbach, Kolmogorov), Ölçüm (measure) Kuramı
- İstatistik: Ters olasılık – Olasılığın anlamı:
  - "Frequentist": Tekrarlanabilir deneylerdeki frekanslar
    * Bu ilaç etkili.
  - "Bayesian": Bilginin (inancın) derecesi
    * Yarın yüzde doksan yağmurlu.
- Brad Efron, *Modern science and the Bayesian-frequentist controversy* , 2005

  `http://www-stat.stanford.edu/~ckirby/brad/papers/2005NEWModernScience.pdf`
- Brad Efron, *Bayesians, frequentists, and scientists* , 2005

  `http://www-stat.stanford.edu/~ckirby/brad/papers/2005BayesFreqSci.pdf`

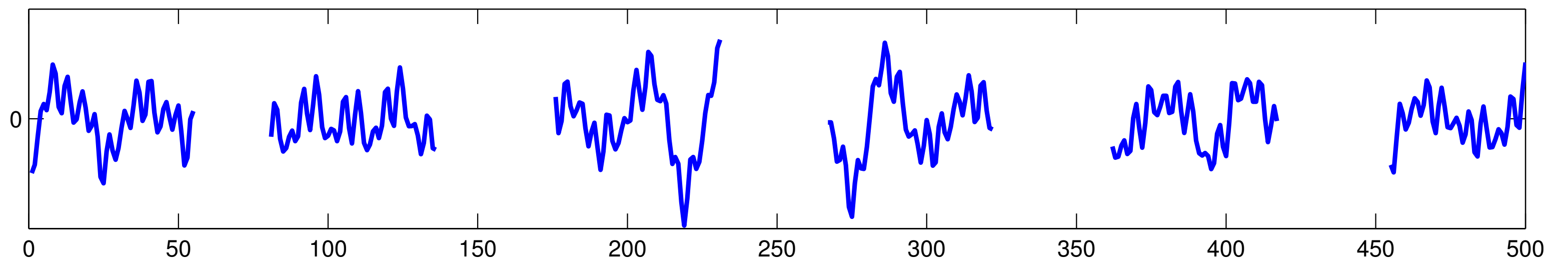
Tümdengelim (Deduction) ve Tümevarım (Inductive)

- a, b, and c tam sayılar olmak üzere

$$a^n + b^n = c^n$$

  denkleminin $n > 2$ için çözümü yoktur.

- Aşağıda verilen ses dalgası içerisindeki kayıp örnekleri bulunuz



Tümevarım'ın tehlikeleri

Borovik

$$\operatorname{snc}(x) \equiv \sin(x)/x$$

$$\begin{aligned}
\int_0^\infty \operatorname{snc}(x)dx &= \pi/2 \\
\int_0^\infty \operatorname{snc}(x)\operatorname{snc}(x/3)dx &= \pi/2 \\
\int_0^\infty \operatorname{snc}(x)\operatorname{snc}(x/3)\operatorname{snc}(x/5)dx &= \pi/2 \\
\int_0^\infty \operatorname{snc}(x)\operatorname{snc}(x/3)\operatorname{snc}(x/5)\operatorname{snc}(x/7)dx &= \pi/2 \\
\int_0^\infty \operatorname{snc}(x)\operatorname{snc}(x/3)\operatorname{snc}(x/5)\operatorname{snc}(x/7)\operatorname{snc}(x/9)dx &= \pi/2
\end{aligned}$$

$$
\begin{aligned}
\int_0^\infty \mathrm{snc}(x)\,\mathrm{snc}(x/3)\,\mathrm{snc}(x/5)\,\mathrm{snc}(x/7)\,\mathrm{snc}(x/9)\,\mathrm{snc}(x/11) dx &= \pi/2 \\
\int_0^\infty \mathrm{snc}(x)\,\mathrm{snc}(x/3)\,\mathrm{snc}(x/5)\,\mathrm{snc}(x/7)\,\mathrm{snc}(x/9)\,\mathrm{snc}(x/11)\,\mathrm{snc}(x/13) dx &= \pi/2 \\
\int_0^\infty \mathrm{snc}(x)\,\mathrm{snc}(x/3)\,\mathrm{snc}(x/5)\,\mathrm{snc}(x/7)\,\mathrm{snc}(x/9)\,\mathrm{snc}(x/11)\,\mathrm{snc}(x/13)\,\mathrm{snc}(x/15) dx &= \frac{467807924713440738696537864469}{935615849440640907310521750000} \cdot \pi
\end{aligned}
$$

Uygulamalar

- Ön bilgi ve gözlemlenen verinin birleştirilmesi için doğal bir çerçeve $\Rightarrow$ Öğrenme
  - Tıbbi tanı (Semptom/Hastalık)
  - Konuşma Tanıma (İşaret/Hece)
  - Bilgisayarla Görme (Görüntü/Nesne)
  - Robotik, Hedef Takibi (Algılayıcı/Pozisyon)
  - Finans (Geçmiş fiyatlar, Piyasa haberleri/Gelecek fiyat)

Olasılık Tabloları

$p(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$
$x_1 = 1$	0.3	0.3
$x_1 = 2$	0.1	0.3

- Marjinal: $p(x_1)$, $p(x_2)$
- Şartlı: $p(x_1|x_2)$, $p(x_2|x_1)$
- Sonsal: $p(x_1, x_2 = 2)$, $p(x_1|x_2 = 2)$
- Marjinal olabilirlik: $p(x_2 = 2)$
- En büyük: $p(x_1^*) = \max_{x_1} p(x_1|x_2 = 1)$
- Mod: $x_1^* = \arg\max_{x_1} p(x_1|x_2 = 1)$
- Max-marginal: $\max_{x_1} p(x_1, x_2)$

Cevaplar

$p(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$
$x_1 = 1$	0.3	0.3
$x_1 = 2$	0.1	0.3

- Marginals:

$p(x_1)$	
$x_1 = 1$	0.6
$x_1 = 2$	0.4

$p(x_2)$	$x_2 = 1$	$x_2 = 2$
	0.4	0.6

- Conditionals:

$p(x_1 \| x_2)$	$x_2 = 1$	$x_2 = 2$
$x_1 = 1$	0.75	0.5
$x_1 = 2$	0.25	0.5

$p(x_2 \| x_1)$	$x_2 = 1$	$x_2 = 2$
$x_1 = 1$	0.5	0.5
$x_1 = 2$	0.25	0.75

Answers

$p(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$
$x_1 = 1$	0.3	0.3
$x_1 = 2$	0.1	0.3

- Posterior:

$p(x_1, x_2 = 2)$	$x_2 = 2$
$x_1 = 1$	0.3
$x_1 = 2$	0.3

$p(x_1 \vert x_2 = 2)$	$x_2 = 2$
$x_1 = 1$	0.5
$x_1 = 2$	0.5

- Evidence:

$$p(x_2 = 2) = \sum_{x_1} p(x_1, x_2 = 2) = 0.6$$

Answers

$p(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$
$x_1 = 1$	0.3	0.3
$x_1 = 2$	0.1	0.3

- Max: (get the value)

$$\max_{x_1} p(x_1 | x_2 = 1) = 0.75$$

- Mode: (get the index)

$$\operatorname*{argmax}_{x_1} p(x_1 | x_2 = 1) = 1$$

- Max-marginal: (get the “skyline”) $\max_{x_1} p(x_1, x_2)$

$\max_{x_1} p(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$
	0.3	0.3

Learning

- Maximum Likelihood,
- Penalised Likelihood,
- Bayesian Learning

Inference and Learning

- Data set

$$\mathcal{D} = \{x_1, \dots x_N\}$$

- Model with parameter λ

$$p(\mathcal{D}|\lambda)$$

- Maximum Likelihood (ML)

$$\lambda^{\text{ML}} = \arg\max_{\lambda} \log p(\mathcal{D}|\lambda)$$

- Predictive distribution

$$p(x_{N+1}|\mathcal{D}) \approx p(x_{N+1}|\lambda^{\text{ML}})$$

Regularisation

- Prior

$$p(\lambda)$$

- Maximum a-posteriori (MAP) : Regularised Maximum Likelihood

$$\lambda^{\text{MAP}} = \arg\max_{\lambda} \log p(\mathcal{D}|\lambda)p(\lambda)$$

- Predictive distribution

$$p(x_{N+1}|\mathcal{D}) \approx p(x_{N+1}|\lambda^{\text{MAP}})$$

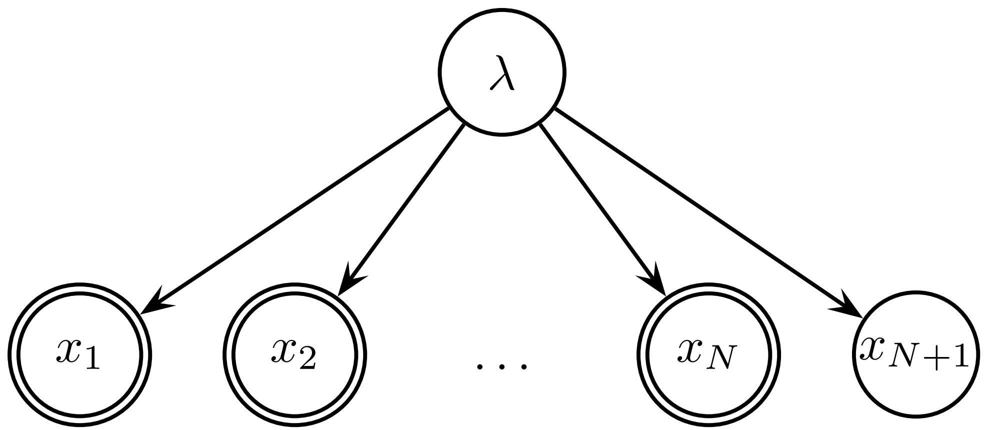
Bayesian Learning

- We treat parameters on the same footing as all other variables
- We integrate over unknown parameters rather than using point estimates (remember the many-dice example)
  - Self-regularisation, avoids overfitting
  - Natural setup for online adaptation
  - Model selection

Bayesian Learning

- Predictive distribution

$$p(x_{N+1}|\mathcal{D}) = \int d\lambda \;\; p(x_{N+1}|\lambda)p(\lambda|\mathcal{D})$$



- Bayesian learning is just inference ...

Probabilistic Modelling

Probability Distributions

- Following distributions are used often as elementary building blocks:
  - Discrete
    - Categorical, Bernoulli, Binomial, Multinomial, Poisson
  - Continuous
    - Gaussian,
    - Beta, Dirichlet
    - Gamma, Inverse Gamma, Exponential, Chi-square, Wishart
    - Student-t, von-Mises

Exponential Family

- Many of those distributions can be written as

$$p(x|\theta) = h(x)\exp\{\theta^\top \psi(x) - A(\theta)\}$$

$$A(\theta) = \log \int_{\mathcal{X}^n} dx\, h(x)\exp(\theta^\top \psi(x))$$

$A(\theta)$	log-partition function
θ	canonical parameters
$\psi(x)$	sufficient statistics
$h(x)$	weighting function

Bernoulli Distribution. $\mathcal{BE}(c; w)$

Binary (Bernoulli) random variable $c = \{0, 1\}$ with probability of success w

$$p(c = 1|w) = w \qquad p(c = 0|w) = 1 - w$$

We write

$$\begin{aligned}
p(c|w) &= w^c (1-w)^{1-c} \\
&= \exp\left(c \log w + (1-c)\log(1-w)\right) \\
&= \exp\left(\log(\frac{w}{1-w})c + \log(1-w)\right) \\
&\equiv \mathcal{BE}(c; w)
\end{aligned}$$

Is Bernoulli an Exponential Family ?

$$\mathcal{BE}(c; w) \quad = \quad \exp\left(\log(\frac{w}{1-w})c + \log(1-w)\right)$$

$$p(c|\theta) \quad = \quad h(c)\exp\{\theta^\top \psi(c) - A(\theta)\}$$

$\theta = \log(\frac{w}{1-w})$	canonical parameters
$A(\theta) = -\log(1+e^\theta)$	log-partition function
$\psi(c) = c$	sufficient statistics
$h(c) = 1$	weighting function

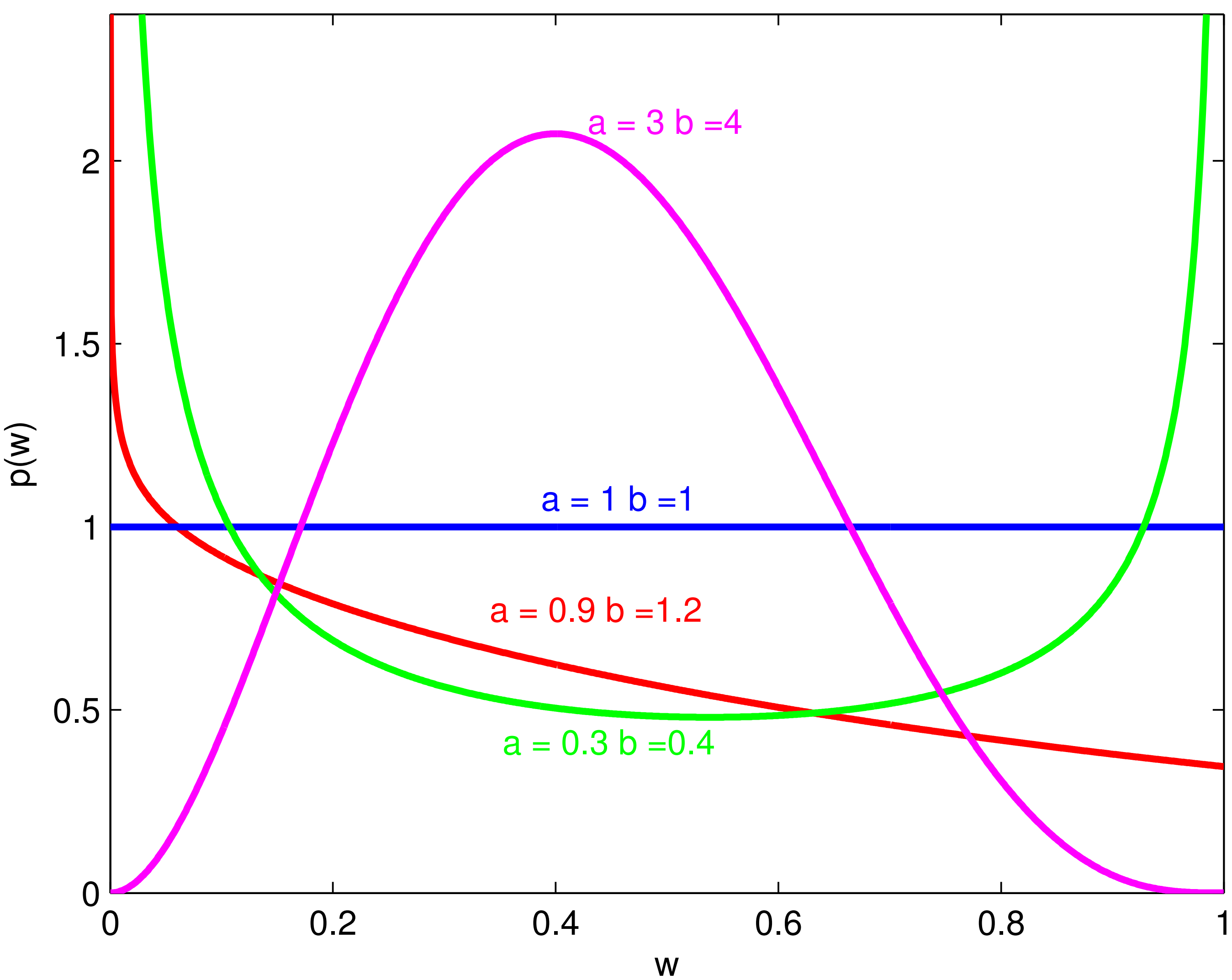
Beta Distribution. $\mathcal{B}(w; a, b)$

$$
\begin{aligned}
\mathcal{B}(w; a, b) &\equiv \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} w^{a-1}(1-w)^{b-1} \\
&= \exp\left((a-1)\log w + (b-1)\log(1-w) - A(a,b)\right) \\
&= \exp\left(\begin{pmatrix} a-1 & b-1 \end{pmatrix} \begin{pmatrix} \log w \\ \log(1-w) \end{pmatrix} - A(a,b)\right) \\
A(a,b) &= \log\Gamma(a) + \log\Gamma(b) - \log\Gamma(a+b)
\end{aligned}
$$

Mean :

$$
\langle w \rangle_{\mathcal{B}} = a/(a+b)
$$

Beta Distribution. $\mathcal{B}(w; a, b)$



Univariate Gaussian. $\mathcal{N}(x; m, S)$

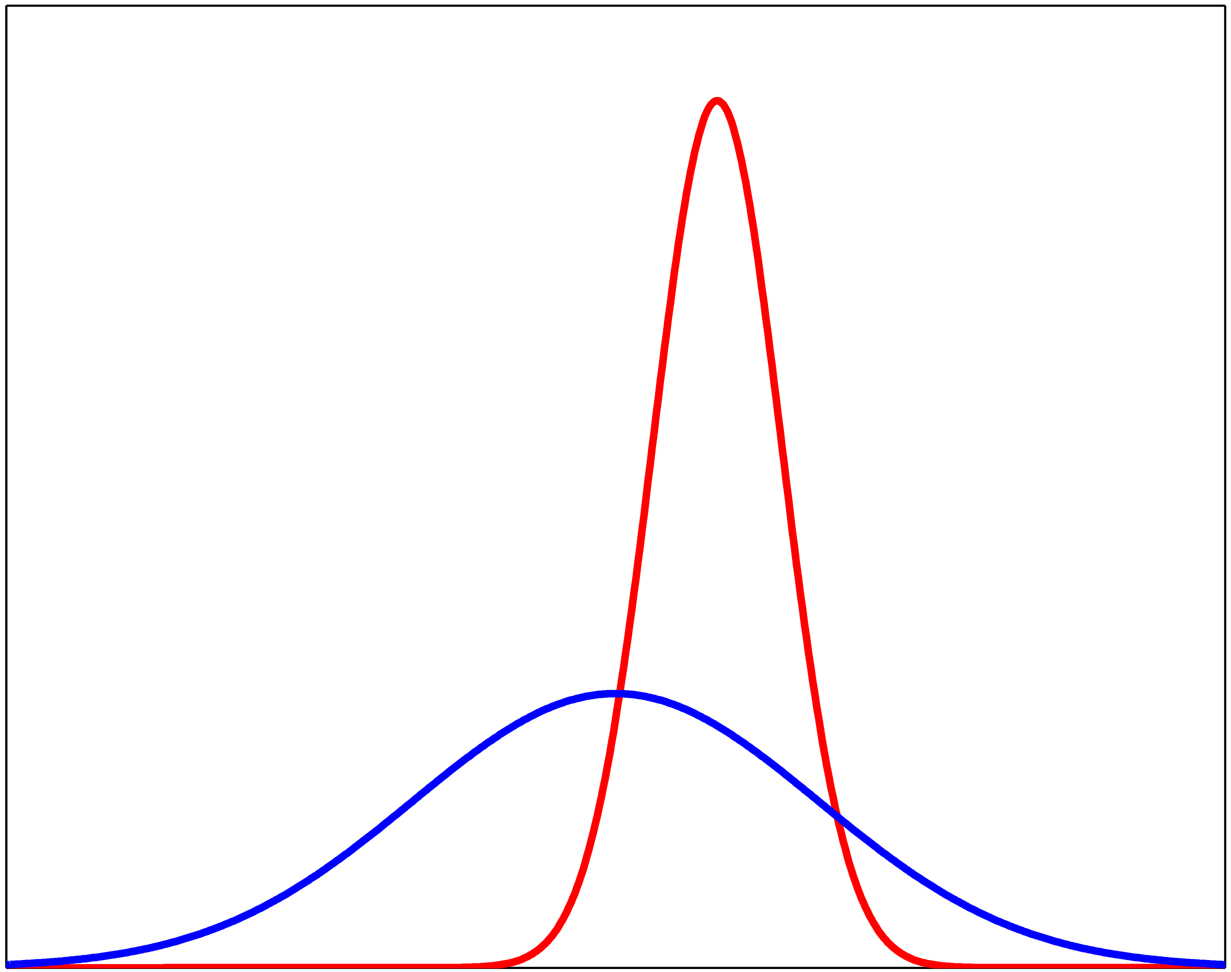
The Gaussian distribution with mean m and covariance S has the form

$$
\begin{aligned}
\mathcal{N}(x; m, S) &= (2\pi S)^{-1/2} \exp\{-\frac{1}{2}(x-m)^2/S\} \\
&= \exp\{-\frac{1}{2}(x^2 + m^2 - 2xm)/S - \frac{1}{2}\log(2\pi S)\} \\
&= \exp\left\{\frac{m}{S}x - \frac{1}{2S}x^2 - \left(\frac{1}{2}\log(2\pi S) + \frac{1}{2S}m^2\right)\right\} \\
&= \exp\{\underbrace{\begin{pmatrix} m/S \\ -\frac{1}{2}/S \end{pmatrix}^\top}_{\theta} \underbrace{\begin{pmatrix} x \\ x^2 \end{pmatrix}}_{\psi(x)} - A(\theta)\}
\end{aligned}
$$

Hence by matching coefficients we have

$$
\exp\left\{-\frac{1}{2}Kx^2 + hx + g\right\} \Leftrightarrow S = K^{-1} \quad m = K^{-1}h
$$

Gaussian.



Inverse Gamma Distribution. $\mathcal{IG}(r; a, b)$

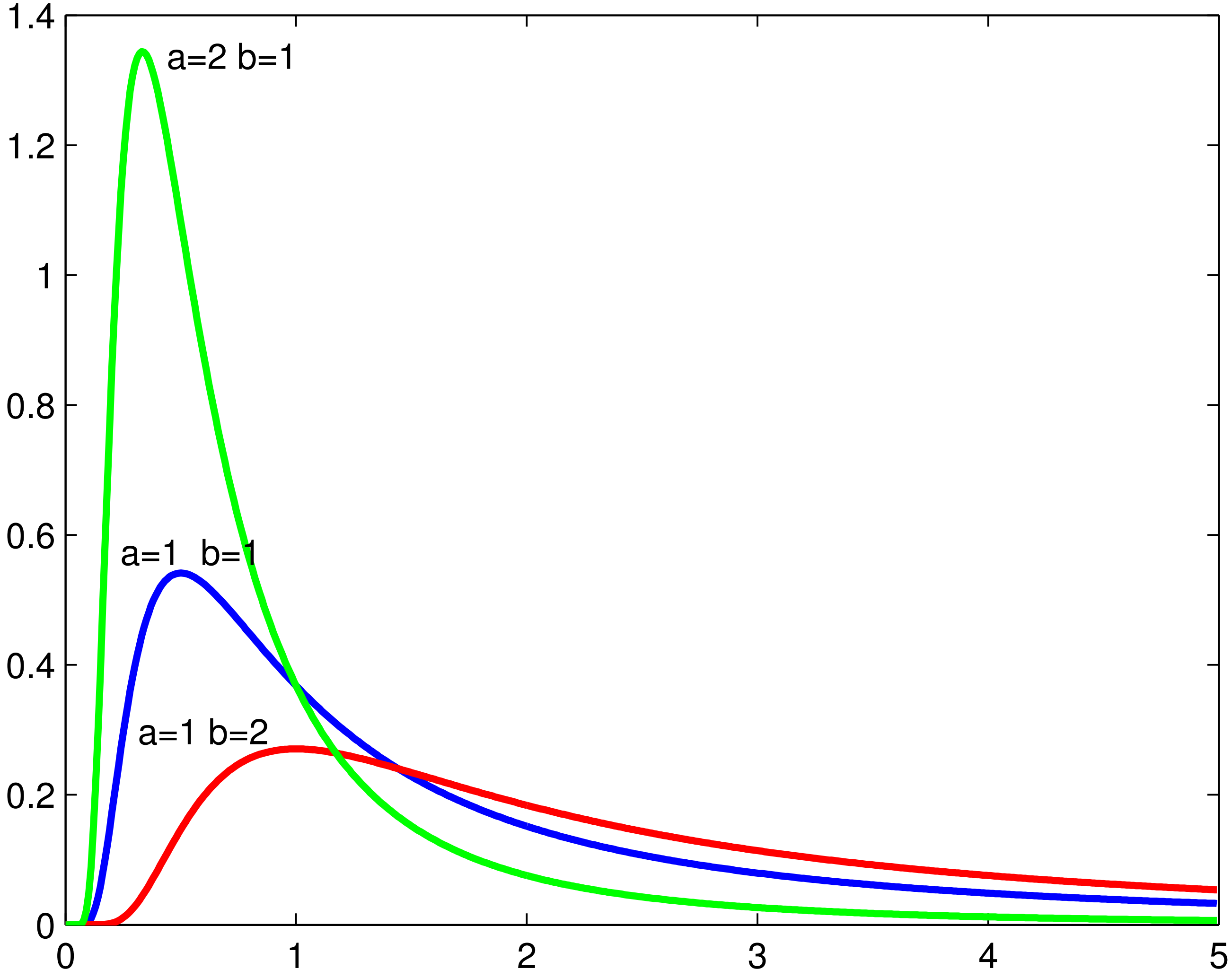
The inverse Gamma distribution with shape a and scale b

$$
\begin{aligned}
\mathcal{IG}(r; a, b) &= \frac{1}{\Gamma(a)} \frac{r^{-(a+1)}}{b^{-a}} \exp(-\frac{b}{r}) \\
&= \exp\left(-(a+1)\log r - \frac{b}{r} - \log\Gamma(a) + a\log b \right) \\
&= \exp\left(\left(\begin{array}{c} -(a+1) \\ -b \end{array} \right)^{\top} \left(\begin{array}{c} \log r \\ 1/r \end{array} \right) - \log\Gamma(a) + a\log b \right)
\end{aligned}
$$

Hence by matching coefficients, we have

$$
\exp\left\{ \alpha \log r + \beta\frac{1}{r} + c \right\} \Leftrightarrow a = -\alpha - 1 \qquad b = -\beta
$$

Inverse Gamma



Gamma Distribution. $\mathcal{G}(\lambda; a, b)$

The Gamma distribution with shape a and **inverse scale** b

$$
\begin{aligned}
\mathcal{G}(\lambda; a, b) &= \frac{1}{\Gamma(a)} b^a \lambda^{(a-1)} \exp(-b\lambda) \\
&= \exp\left((a-1)\log\lambda - b\lambda - \log\Gamma(a) + a\log b\right) \\
&= \exp\left(\left(\begin{array}{c} (a-1) \\ -b \end{array}\right)^\top \left(\begin{array}{c} \log\lambda \\ \lambda \end{array}\right) - \log\Gamma(a) + a\log b\right)
\end{aligned}
$$

Hence by matching coefficients, we have

$$
\exp\left\{\alpha \log r + \beta \frac{1}{r} + c\right\} \Leftrightarrow a = \alpha + 1 \qquad b = -\beta
$$

Random number generation

- Bernoulli: $\mathcal{BE}(x; p)$

```
x = double(rand<p);
```

- Binomial: $\mathcal{BI}(x; p, N)$

```
x = sum(double(rand(N,1)<p));
```

  Not efficient for large N

- Poisson: $\mathcal{PO}(x; \lambda)$

```
x = poissrnd(lambda);
```

- Beta: $\mathcal{B}(x; a, b)$

```
x = betarnd(a, b);
```

- Gaussian: $\mathcal{N}(x; \mu, S)$

```
x = sqrt(S).*randn(size(S)) + mu;
```

- Gamma: $x \sim \mathcal{G}(x; a, b)$

```
x = gamrnd(a, 1./b);
```

or more securely

```
x = gamrnd(a, 1)./b;
```

which is also

```
x = gamrnd(a)./b;
```

- Inverse Gamma $x \sim \mathcal{IG}(x; a, b)$

```
x = b./gamrnd(a);
```

Conjugate priors: Posterior is in the same family as the prior.

Example: posterior inference for the probability of success w of a binary (Bernoulli) random variable c

$$
\begin{aligned}
p(c|w) &= \mathcal{BE}(c; w) = \exp\left(c \log w + (1-c)\log(1-w)\right) \\
p(w) &= \mathcal{B}(w; a, b)
\end{aligned}
$$

$$
\begin{aligned}
p(w|c) &\propto p(c|w)p(w) \\
&\propto \exp\left(c \log w + (1-c)\log(1-w)\right) \\
&\quad \times \exp\left((a-1)\log w + (b-1)\log(1-w)\right) \\
&\propto \mathcal{B}(w; a+c, b+(1-c))
\end{aligned}
$$

$$
p(w|c) = \begin{cases} \mathcal{B}(w; a+1, b) & c = 1 \\ \mathcal{B}(w; a, b+1) & c = 0 \end{cases}
$$

Conjugate priors: Posterior is in the same family as the prior.

Example: posterior inference for the variance R of a zero mean Gaussian.

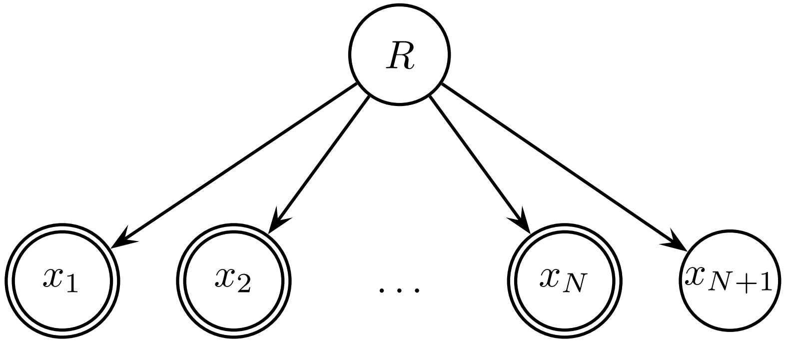
$$
\begin{aligned}
p(x|R) &= \mathcal{N}(x; 0, R) \\
p(R) &= \mathcal{IG}(R; a, b)
\end{aligned}
$$

$$
\begin{aligned}
p(R|x) &\propto p(R)p(x|R) \\
&\propto \exp\left(-(a+1)\log R - b\frac{1}{R}\right)\exp\left(-(x^2/2)\frac{1}{R} - \frac{1}{2}\log R\right) \\
&= \exp\left(\left(\begin{array}{c} -(a+1+\frac{1}{2}) \\ -(b+x^2/2) \end{array}\right)^\top \left(\begin{array}{c} \log R \\ 1/R \end{array}\right)\right) \\
&\propto \mathcal{IG}(R; a+\frac{1}{2}, b+x^2/2)
\end{aligned}
$$

Like the prior, this is an inverse-Gamma distribution.

Conjugate priors: Posterior is in the same family as the prior.

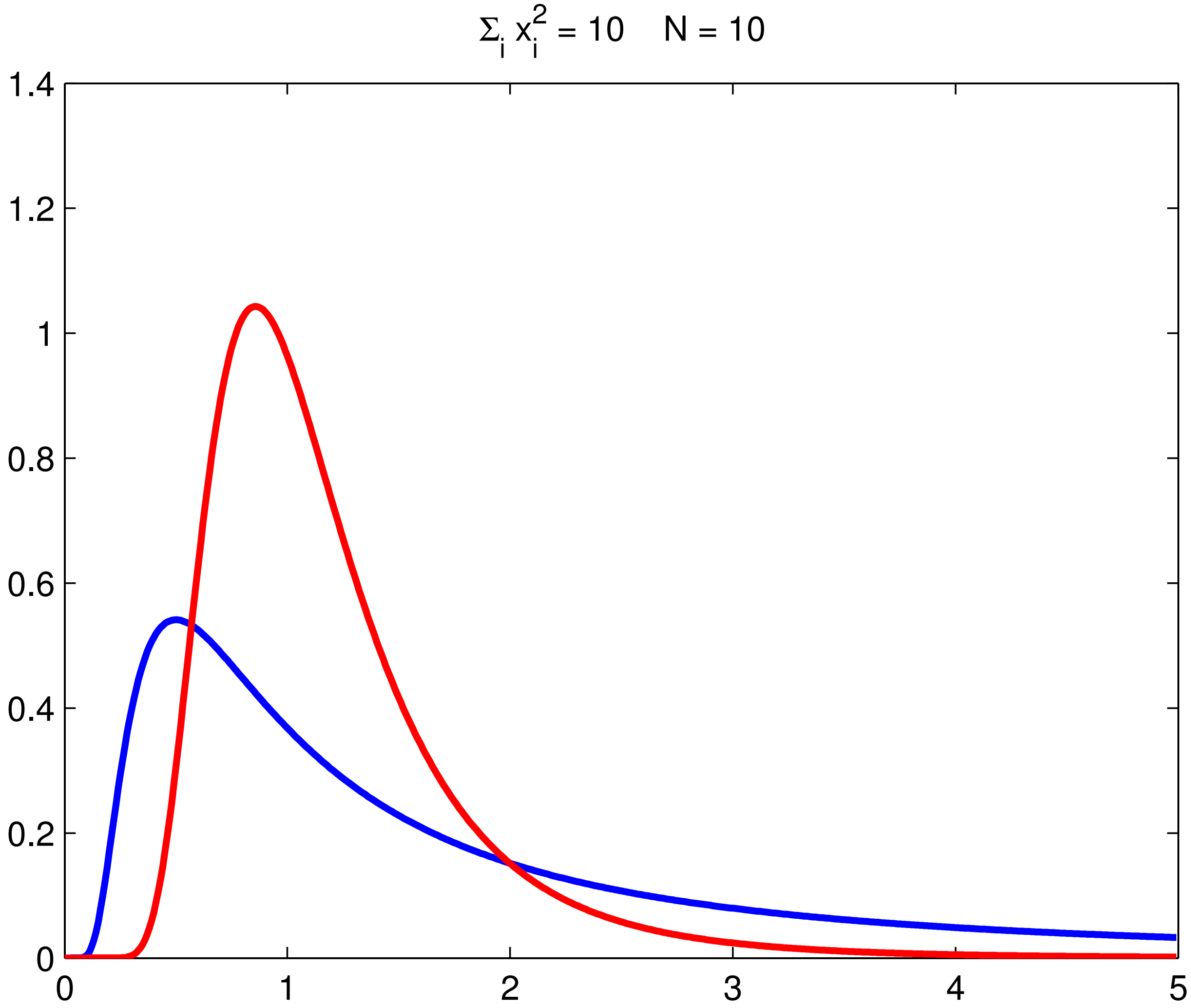
Example: posterior inference of variance R from $x_1, \dots, x_N$.



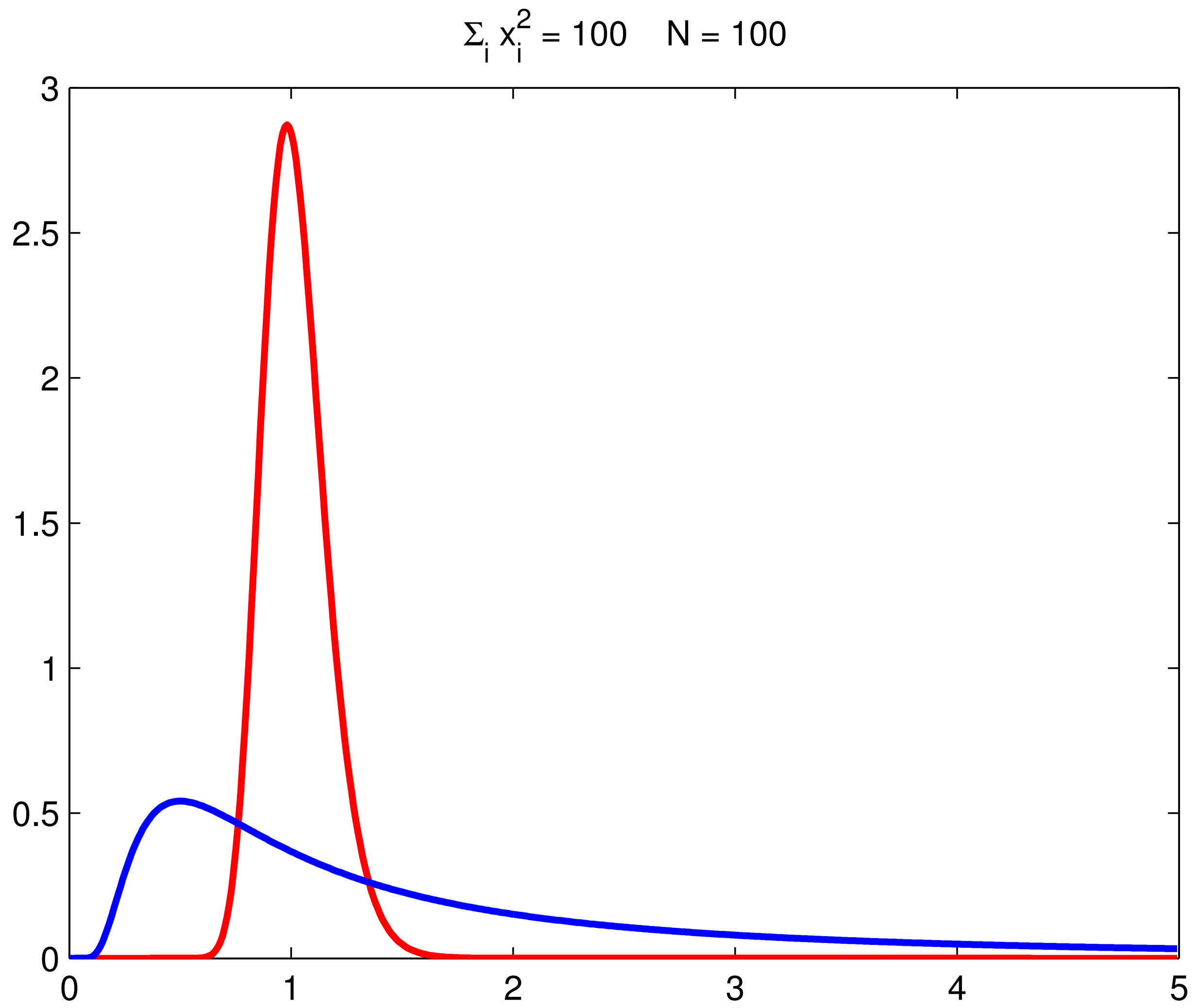
$$
\begin{aligned}
p(R|x) &\propto p(R)\prod_{i=1}^{N} p(x_i|R) \\
&\propto \exp\left(-(a+1)\log R - b\frac{1}{R}\right)\exp\left(-\left(\frac{1}{2}\sum_i x_i^2\right)\frac{1}{R} - \frac{N}{2}\log R\right) \\
&= \exp\left(\begin{pmatrix} -(a+1+\frac{N}{2}) \\ -(b+\frac{1}{2}\sum_i x_i^2) \end{pmatrix}^\top \begin{pmatrix} \log R \\ 1/R \end{pmatrix}\right) \propto \mathcal{IG}(R; a+\frac{N}{2}, b+\frac{1}{2}\sum_i x_i^2)
\end{aligned}
$$

Sufficient statistics are **additive**

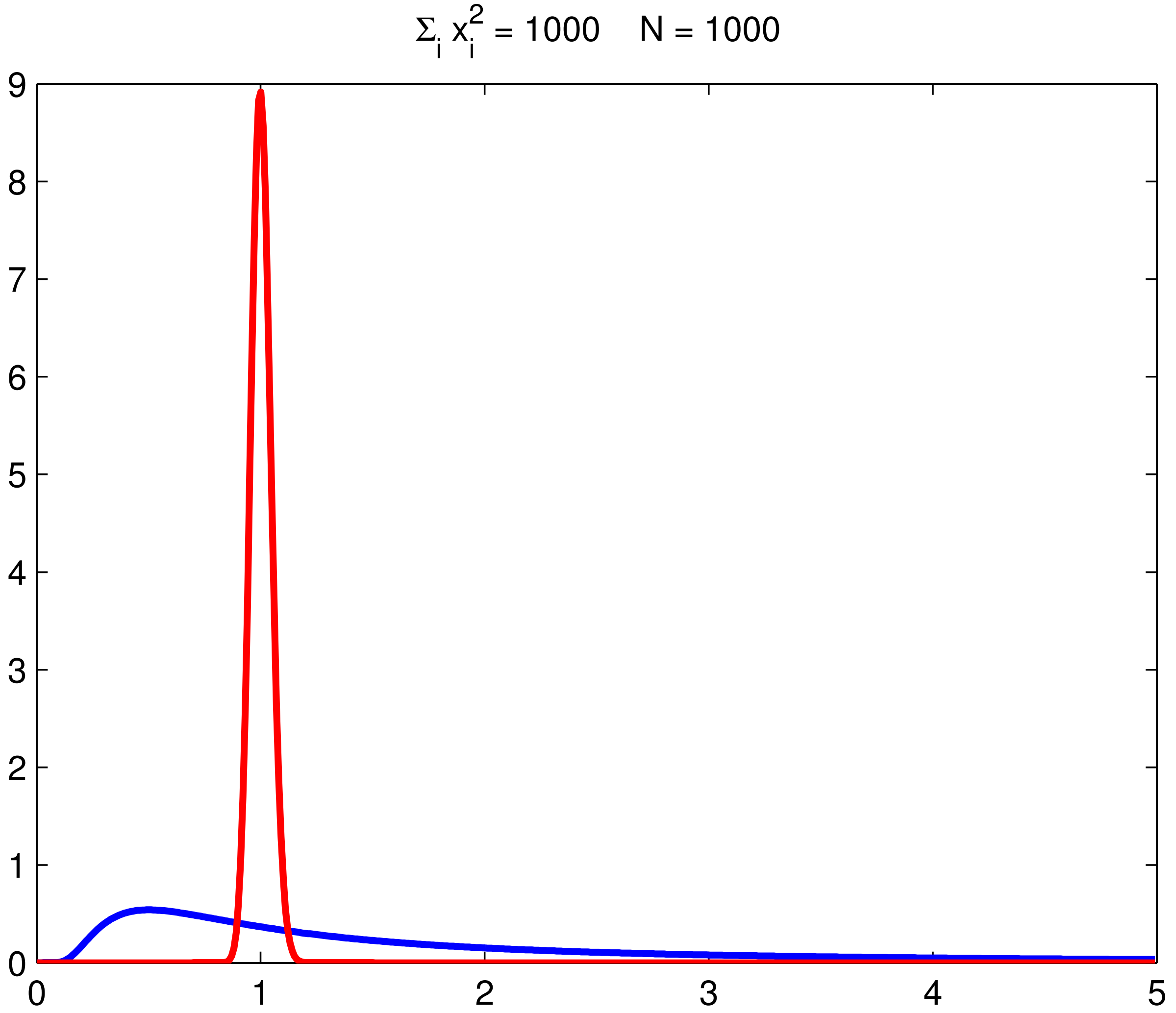
Inverse Gamma, $\sum_i x_i^2 = 10 \quad N = 10$



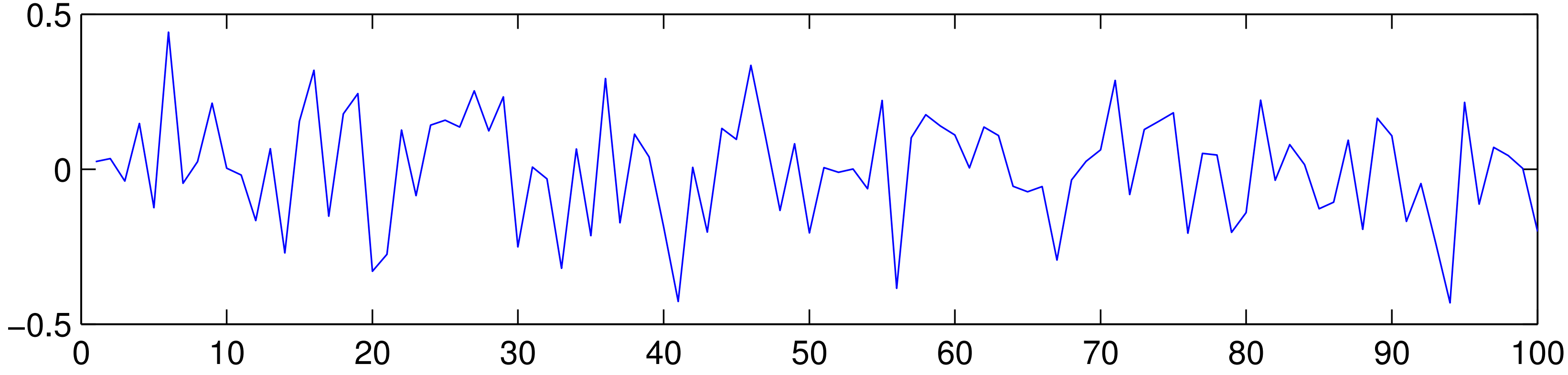
Inverse Gamma, $\sum_i x_i^2 = 100 \quad N = 100$



Inverse Gamma, $\sum_i x_i^2 = 1000 \quad N = 1000$



Example: AR(1) model



$$x_k = Ax_{k-1} + \epsilon_k \qquad\qquad k = 1 \dots K$$

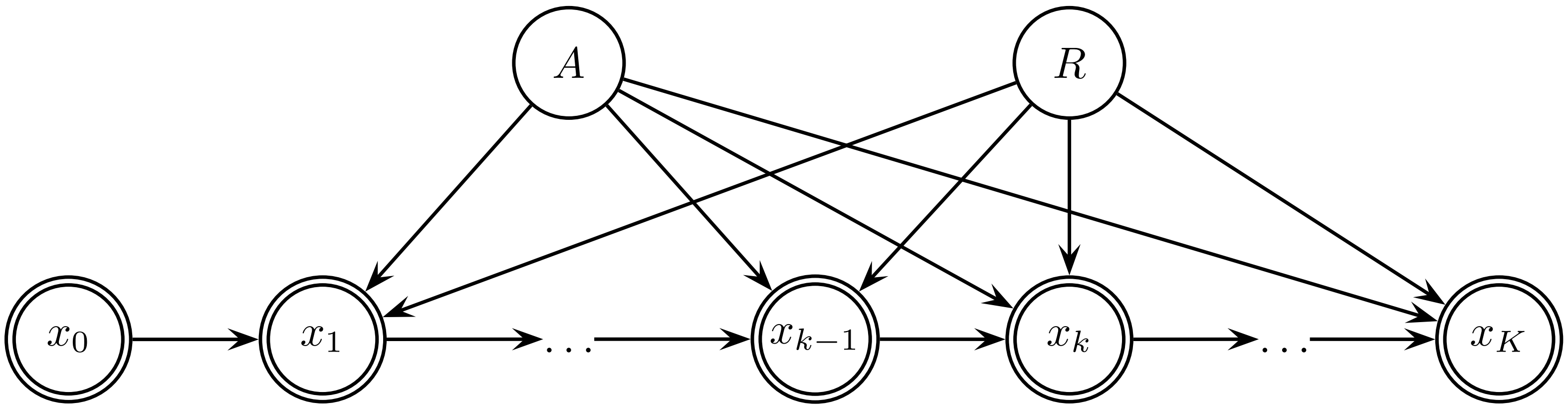
ϵ_k is i.i.d., zero mean and normal with variance R.

Estimation problem:

Given $x_0, \dots, x_K$, determine coefficient A and variance R (both scalars).

AR(1) model, Generative Model notation

$$
\begin{aligned}
A &\sim \mathcal{N}(A; 0, P) \\
R &\sim \mathcal{IG}(R; \nu, \beta/\nu) \\
x_k | x_{k-1}, A, R &\sim \mathcal{N}(x_k; A x_{k-1}, R) \qquad\qquad x_0 = \hat{x}_0
\end{aligned}
$$



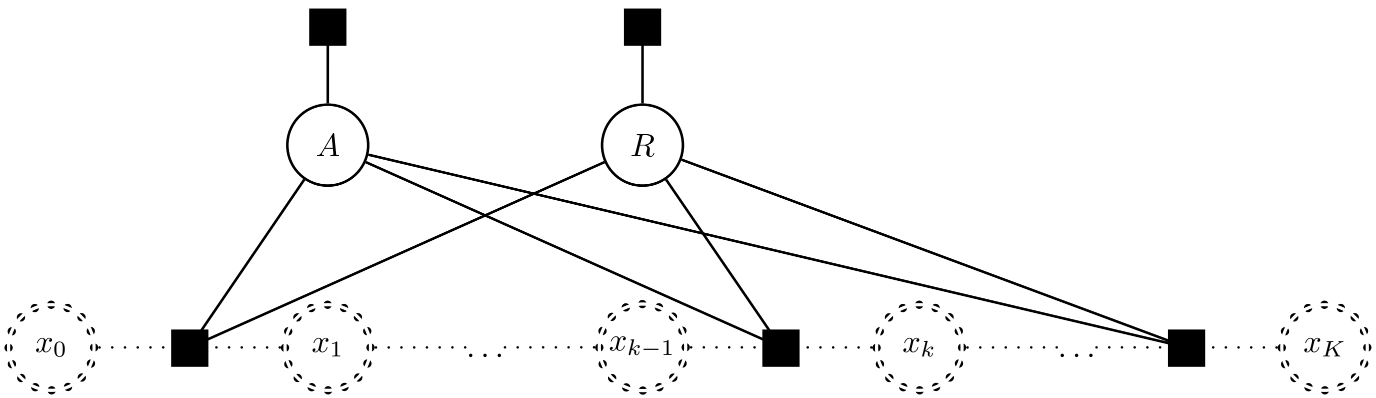
Observed variables are shown with double circles

AR(1) Model. Bayesian Posterior Inference

$$
\begin{aligned}
p(A, R|x_0, x_1, \dots, x_K) &\propto p(x_1, \dots, x_K|x_0, A, R)p(A, R) \\
\text{Posterior} &\propto \text{Likelihood} \times \text{Prior}
\end{aligned}
$$

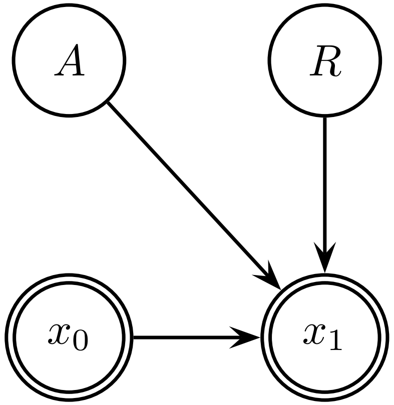
Using the Markovian (conditional independence) structure we have

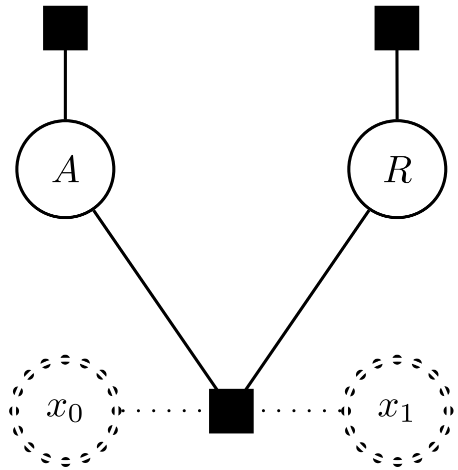
$$
p(A, R|x_0, x_1, \dots, x_K) \propto \left(\prod_{k=1}^{K} p(x_k|x_{k-1}, A, R) \right) p(A)p(R)
$$



Numerical Example

Suppose $K = 1$,





By Bayes' Theorem and the structure of AR(1) model

$$
\begin{aligned}
p(A, R|x_0, x_1) &\propto p(x_1|x_0, A, R)p(A)p(R) \\
&= \mathcal{N}(x_1; Ax_0, R)\mathcal{N}(A; 0, P)\mathcal{IG}(R; \nu, \beta/\nu)
\end{aligned}
$$

Numerical Example

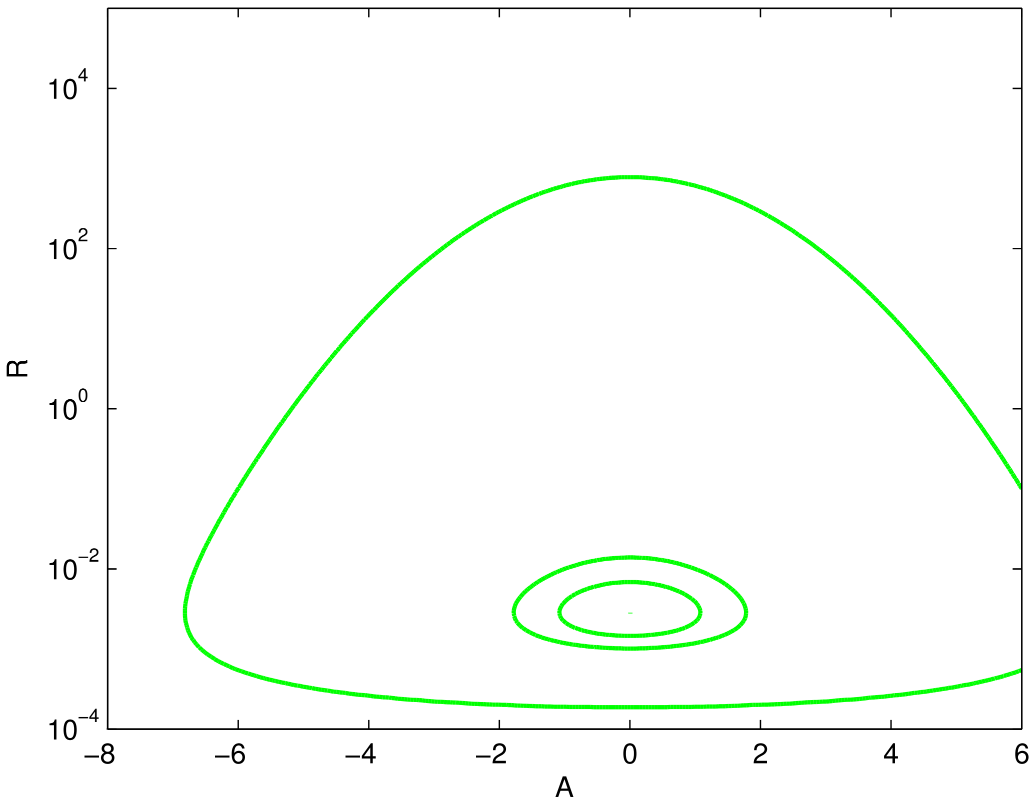
$$
\begin{aligned}
p(A, R|x_0, x_1) &\propto p(x_1|x_0, A, R)p(A)p(R) \\
&= \mathcal{N}(x_1; Ax_0, R)\mathcal{N}(A; 0, P)\mathcal{IG}(R; \nu, \beta/\nu) \\
&\propto \exp\left(-\frac{1}{2}\frac{x_1^2}{R} + x_0 x_1 \frac{A}{R} - \frac{1}{2}\frac{x_0^2 A^2}{R} - \frac{1}{2}\log 2\pi R\right) \\
&\quad \exp\left(-\frac{1}{2}\frac{A^2}{P}\right)\exp\left(-(\nu+1)\log R - \frac{\nu}{\beta}\frac{1}{R}\right)
\end{aligned}
$$

This posterior has a nonstandard form

$$
\exp\left(\alpha_1 \frac{1}{R} + \alpha_2 \frac{A}{R} + \alpha_3 \frac{A^2}{R} + \alpha_4 \log R + \alpha_5 A^2\right)
$$

Numerical Example, the prior $p(A, R)$

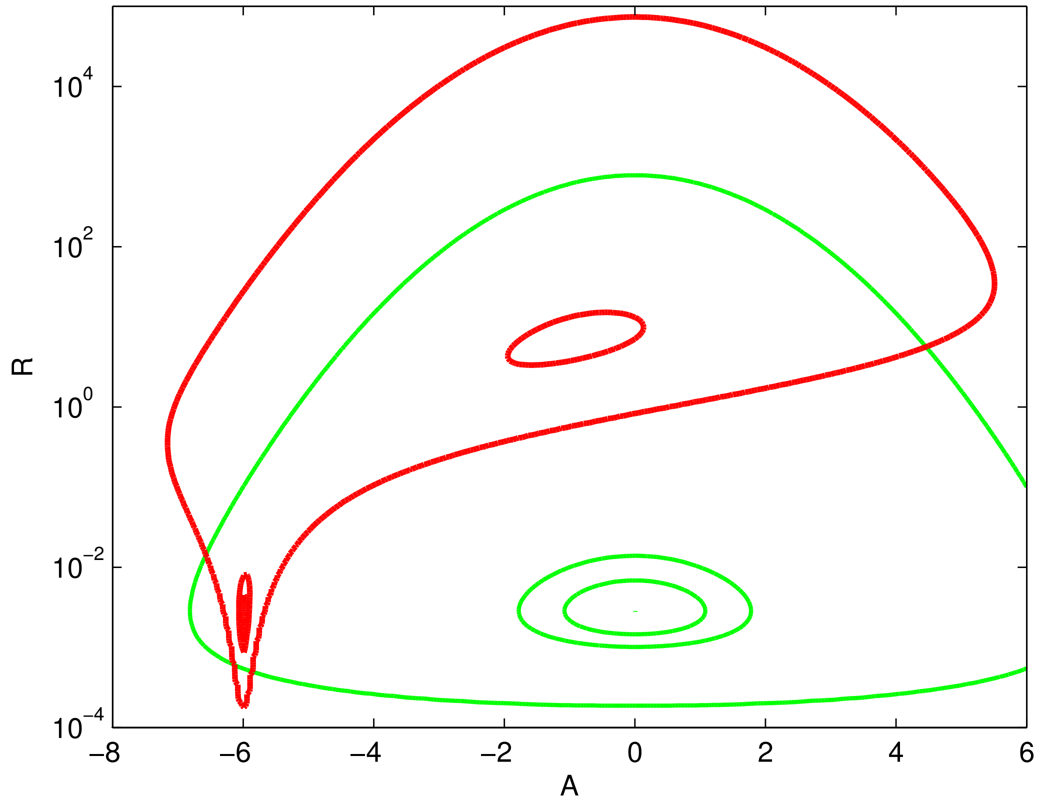
Equiprobability contour of $p(A)p(R)$



$$A \sim \mathcal{N}(A; 0, 1.2) \qquad R \sim \mathcal{IG}(R; 0.4, 250)$$

Suppose: $x_0 = 1 \qquad x_1 = -6 \qquad x_1 \sim \mathcal{N}(x_1; Ax_0, R)$

Numerical Example, the posterior $p(A, R|x)$



Note the bimodal posterior with $x_0 = 1, x_1 = -6$

- $A \approx -6 \Leftrightarrow$ low noise variance R.
- $A \approx 0 \Leftrightarrow$ high noise variance R.

Remarks

- The point estimates such as ML or MAP are not always representative about the solution
- (Unfortunately), exact posterior inference is only possible for few special cases
- Even very simple models can lead easily to complicated posterior distributions
- Ambiguous data usually leads to a multimodal posterior, each mode corresponding to one possible explanation

Remarks

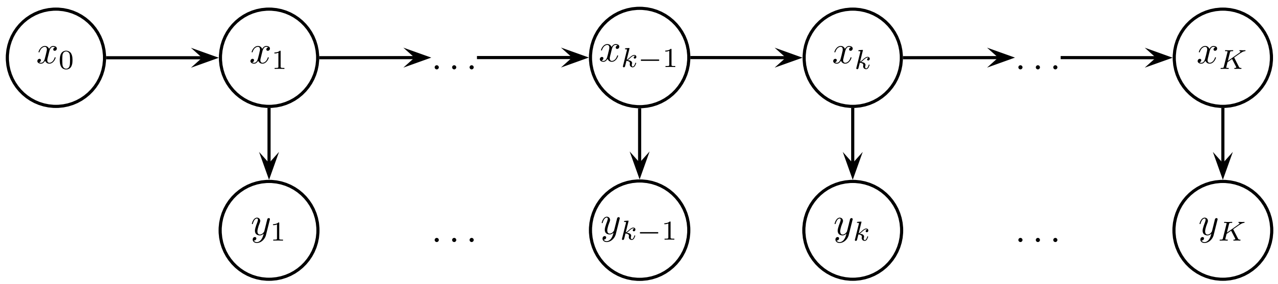
- *A-priori* independent variables often become dependent *a-posteriori* ("Explaining away")
- The difficulty of an inference problem depends, among others, upon the particular "parameter regime" and observed data sequence

Lecture Outline

- Sequential data, Terminology
- Hidden Markov Models
- Implementation of the Forward-Backward algorithm
- Finding the MAP trajectory: the Viterbi algorithm

Sequential Data: Models, Inference, Terminology

In signal processing, machine learning, robotics, statistics many phenomena are modelled by dynamical models

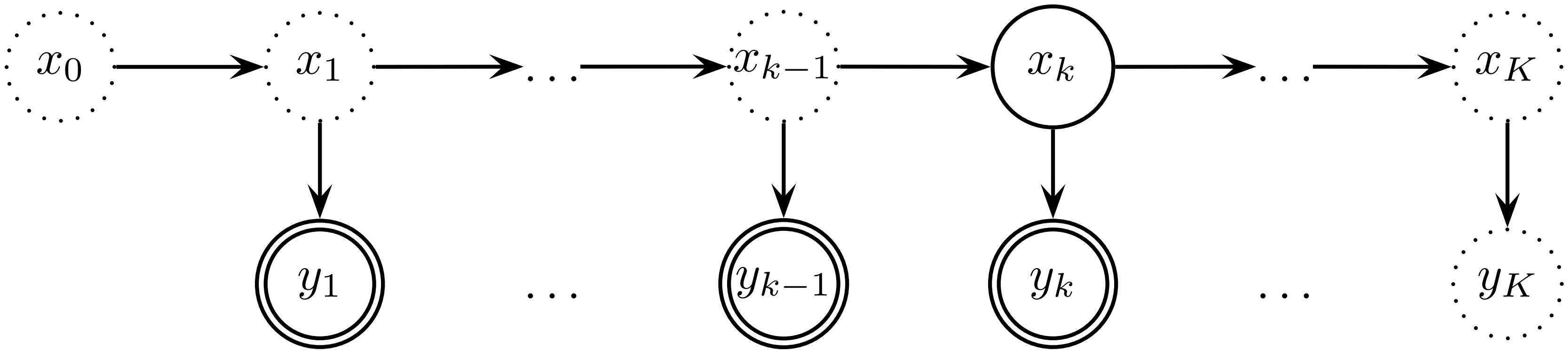


$$
\begin{aligned}
x_k &\sim p(x_k|x_{k-1}) && \text{Transition Model} \\
y_k &\sim p(y_k|x_k) && \text{Observation Model}
\end{aligned}
$$

- x is the latent state (tempo, pitch, velocity, attitude, class label, ...)
- y are observations (samples, onsets, sensor reading, pixels, features, ...)
- In a full Bayesian setting, x includes unknown model parameters

Online Inference, Terminology

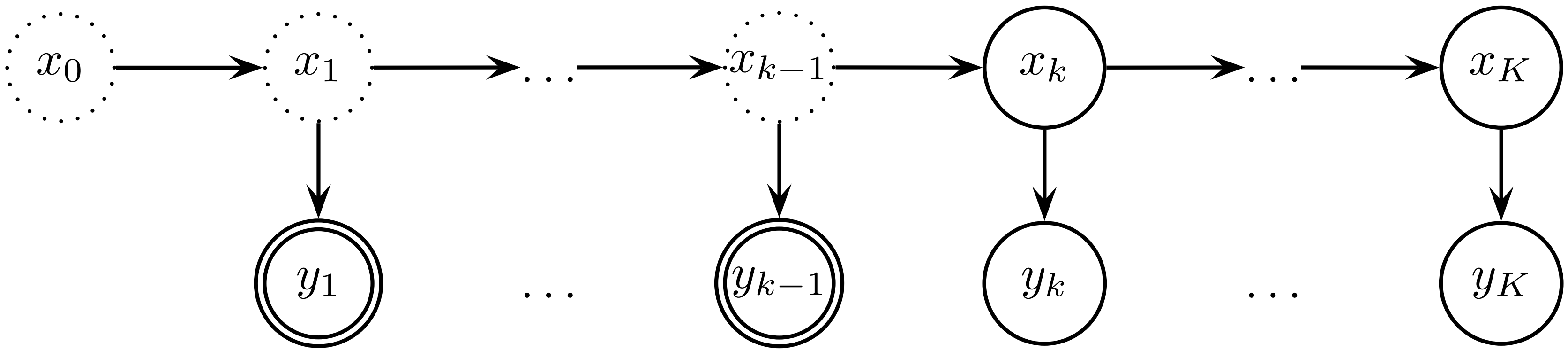
- **Filtering:** $p(x_k|y_{1:k})$
  - Distribution of current state given all past information
  - Realtime/Online/Sequential Processing



- Potentially confusing misnomer:
  - More general than “digital filtering” (convolution) in DSP – but algoritmically related for some models (KFM)

Online Inference, Terminology

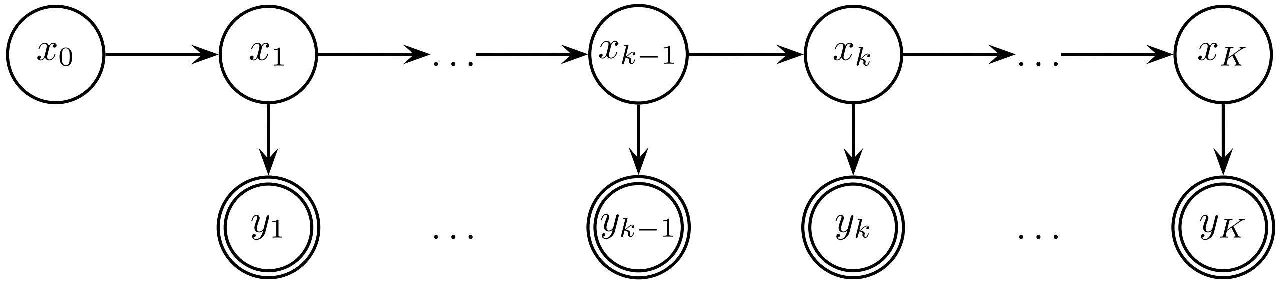
- **Prediction** $p(y_{k:K}, x_{k:K}|y_{1:k-1})$
  - evaluation of possible future outcomes; like filtering without observations



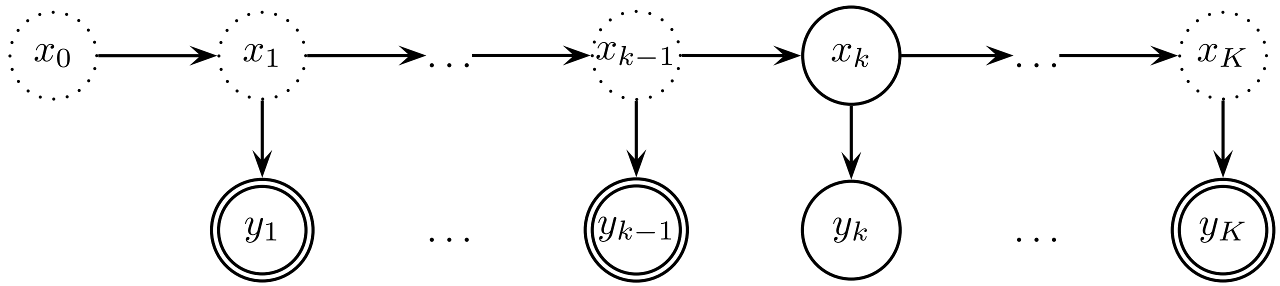
- Accompaniment, Tracking, Restoration

Offline Inference, Terminology

- **Smoothing** $p(x_{0:K}|y_{1:K})$,
  Most likely trajectory – Viterbi path $\arg\max_{x_{0:K}} p(x_{0:K}|y_{1:K})$
  better estimate of past states, essential for learning

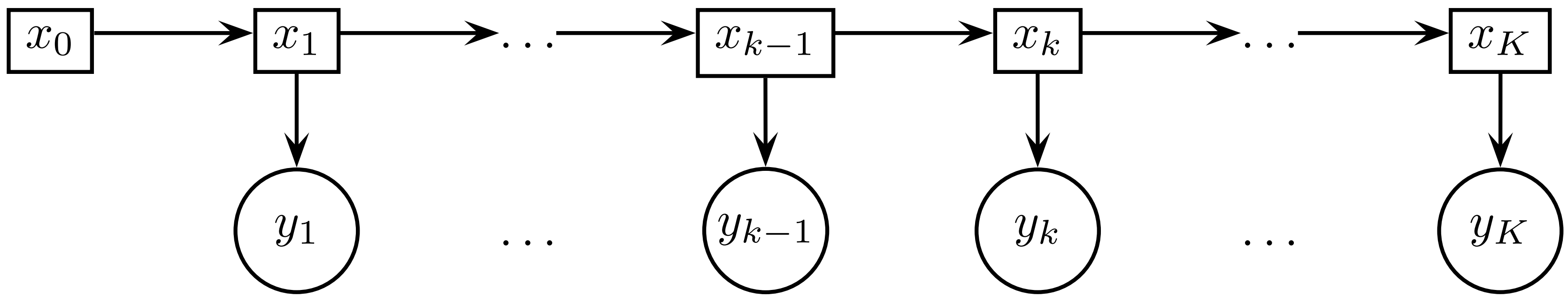


- **Interpolation** $p(y_k, x_k|y_{1:k-1}, y_{k+1:K})$
  fill in lost observations given past and future



Hidden Markov Model [?]

- Mixture model evolving in time



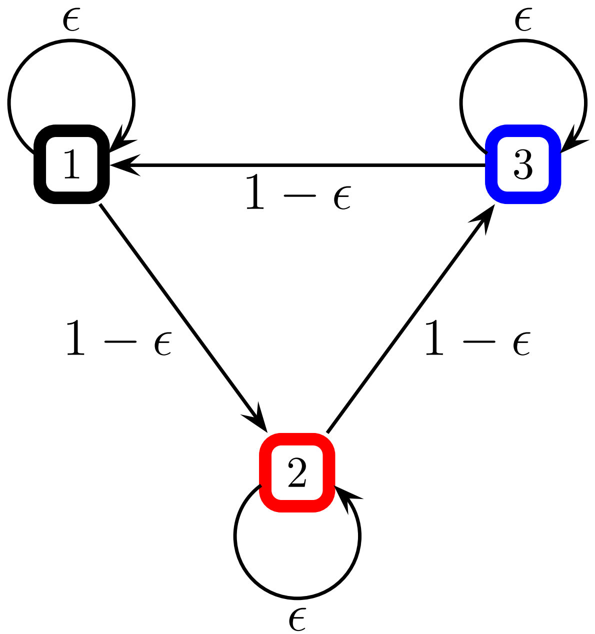
- Observations y_k are continuous or discrete

- Latent variables x_k are discrete
  - Represents the fading memory of the process

- Exact inference possible if x_k has a “small” number of states

Example: Hidden Markov Model

- State transition model (a N by N matrix)

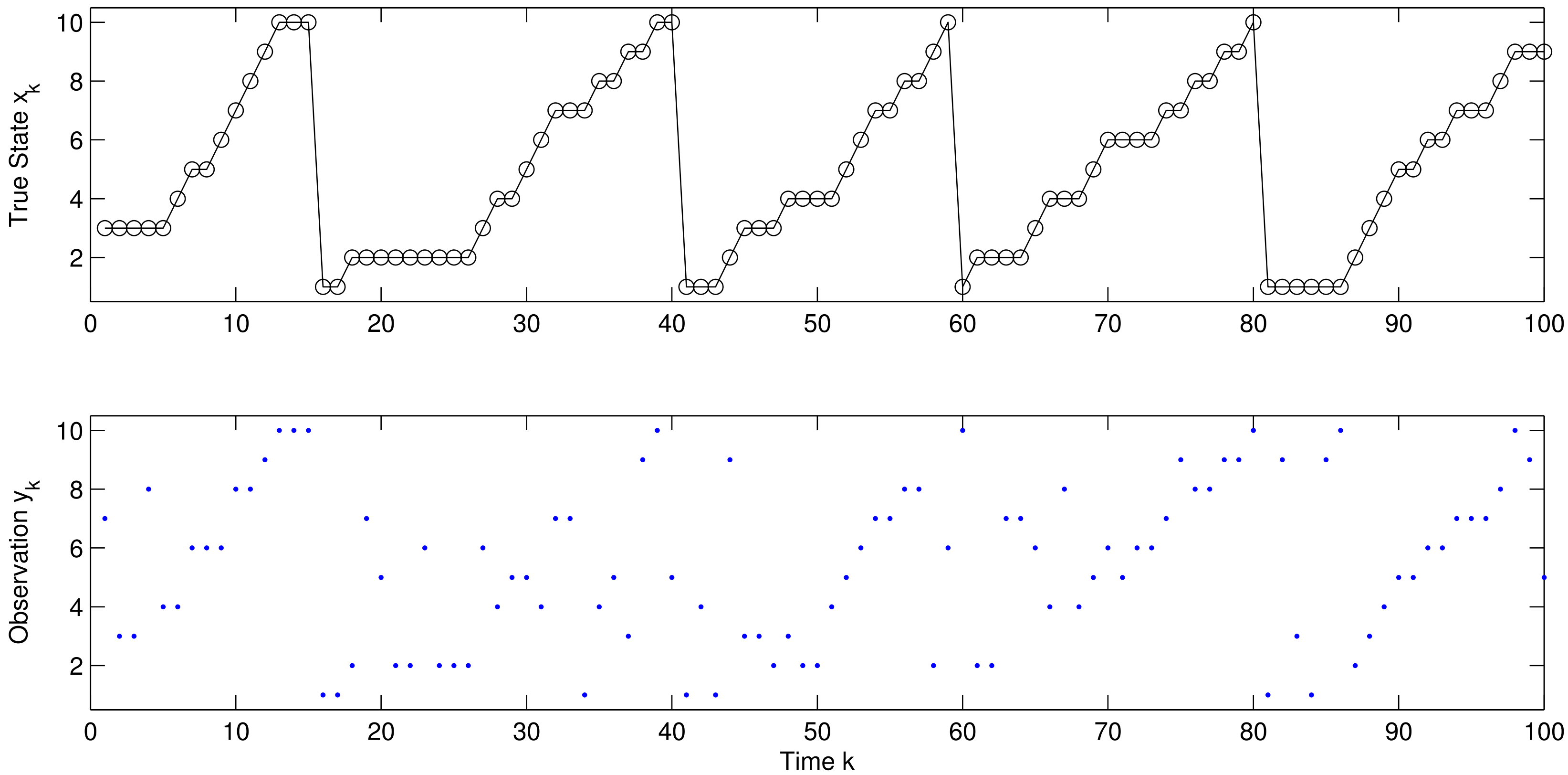


$$(1-\epsilon)\begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} + \epsilon \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

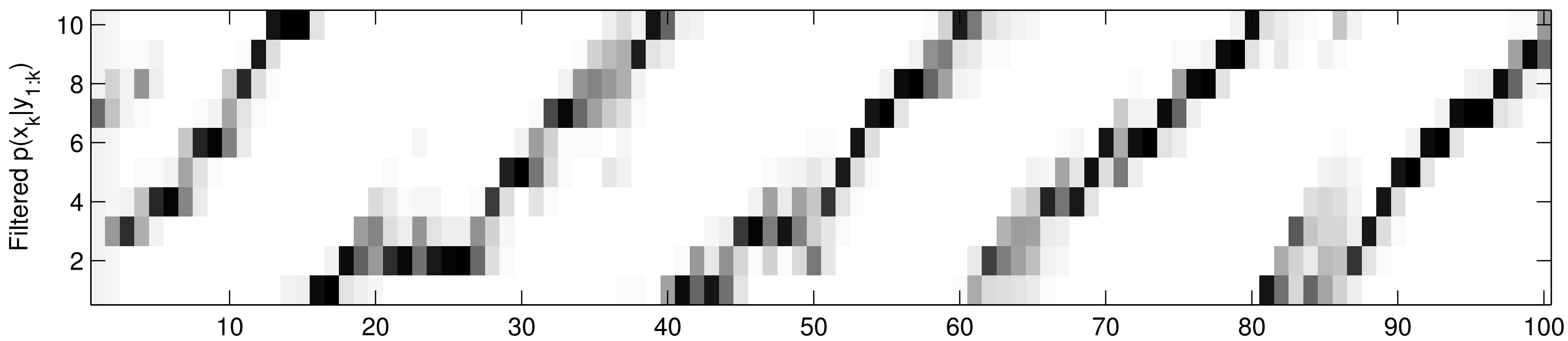
- Observation model $p(y_k|x_k)$

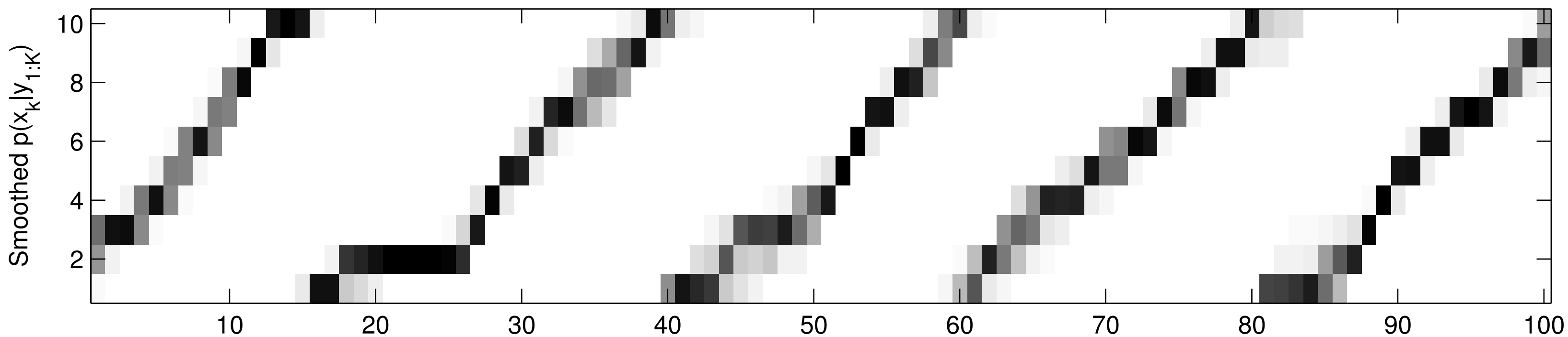
$$y_k \sim w\delta(y_k - x_k) + (1-w)u(1, N)$$

Example: Hidden Markov Model

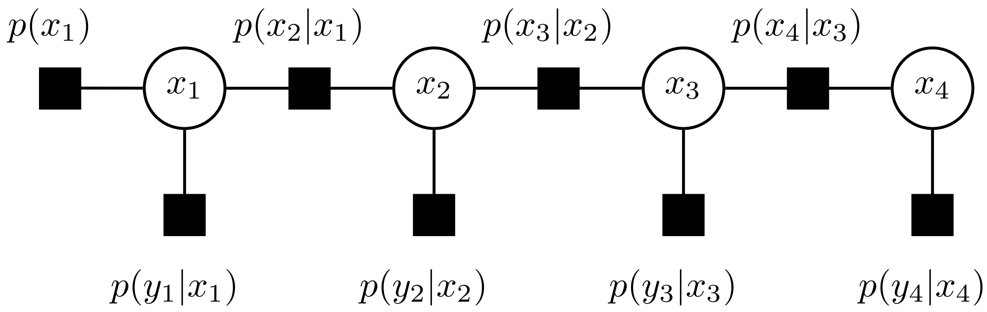


Example: Hidden Markov Model





Exact Inference in HMM, Forward/Backward Algorithm



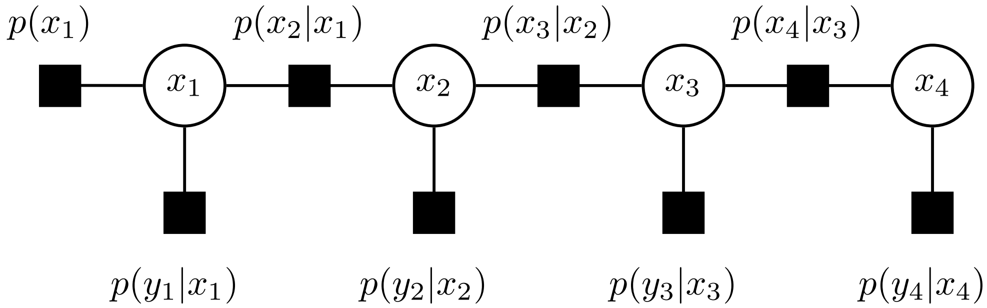
- Forward Pass

$$
\begin{aligned}
p(y_{1:K}) &= \sum_{x_{1:K}} p(y_{1:K}|x_{1:K})p(x_{1:K}) \\
&= \underbrace{\sum_{x_K} p(y_K|x_K) \sum_{x_{K-1}} p(x_K|x_{K-1})}_{\alpha_K} \cdots \sum_{x_2} p(x_3|x_2) \underbrace{p(y_2|x_2) \overbrace{\sum_{x_1} p(x_2|x_1)}^{\alpha_{2|1}}}_{\alpha_2} \underbrace{p(y_1|x_1) \overbrace{p(x_1)}^{\alpha_{1|0}}}_{\alpha_1}
\end{aligned}
$$

- Backward Pass

$$
p(y_{1:K}) = \sum_{x_1} p(x_1)p(y_1|x_1) \ldots \underbrace{\sum_{x_{K-1}} p(x_{K-1}|x_{K-2})p(y_{K-1}|x_{K-1})}_{\beta_{K-2}} \underbrace{\sum_{x_K} p(x_K|x_{K-1})p(y_K|x_K)}_{\beta_{K-1}} \underbrace{\mathbf{1}}_{\beta_K}
$$

Exact Inference in HMM, Viterbi Algorithm



- Merely replace sum by max, equivalent to dynamic programming
- Forward Pass

$$\begin{aligned}
p(y_{1:K}|x^*_{1:K}) &= \max_{x_{1:K}} p(y_{1:K}|x_{1:K})p(x_{1:K}) \\
&= \underbrace{\max_{x_K} p(y_T|x_K) \max_{x_{K-1}} p(x_K|x_{K-1})}_{\alpha_K} \dots \max_{x_2} p(x_3|x_2) \underbrace{p(y_2|x_2) \overbrace{\max_{x_1} p(x_2|x_1)}^{\alpha_{2|1}}}_{\alpha_2} \underbrace{p(y_1|x_1) \overbrace{p(x_1)}^{\alpha_{1|0}}}_{\alpha_1}
\end{aligned}$$

- Backward Pass

$$p(y_{1:K}|x^*_{1:K}) = \max_{x_1} p(x_1)p(y_1|x_1) \dots \underbrace{\max_{x_{K-1}} p(x_{K-1}|x_{K-2})p(y_{K-1}|x_{K-1})}_{\beta_{K-2}} \underbrace{\max_{x_K} p(x_K|x_{K-1})p(y_K|x_K)}_{\beta_{K-1}} \underbrace{\mathbf{1}}_{\beta_K}$$

Implementation of Forward-Backward

1. Setup a parameter structure

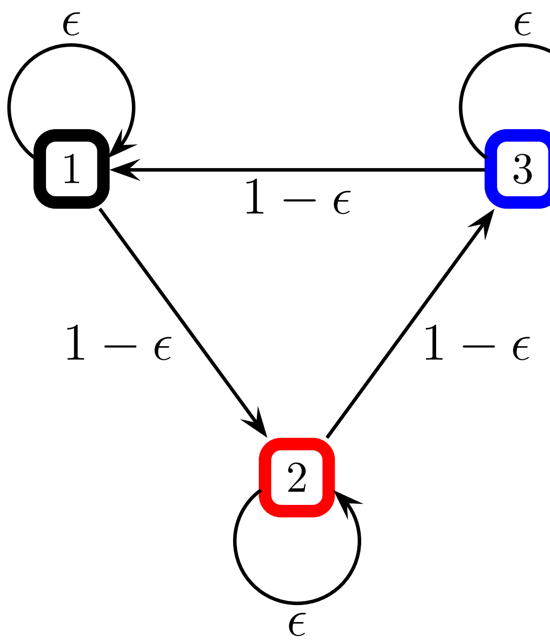
2. Generate data from the true model

3. Inference given true model parameters

4. Test and Visualisation

Example: Hidden Markov Model

- State transition model (a N by N matrix)



$$(1-\epsilon)\begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} + \epsilon \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

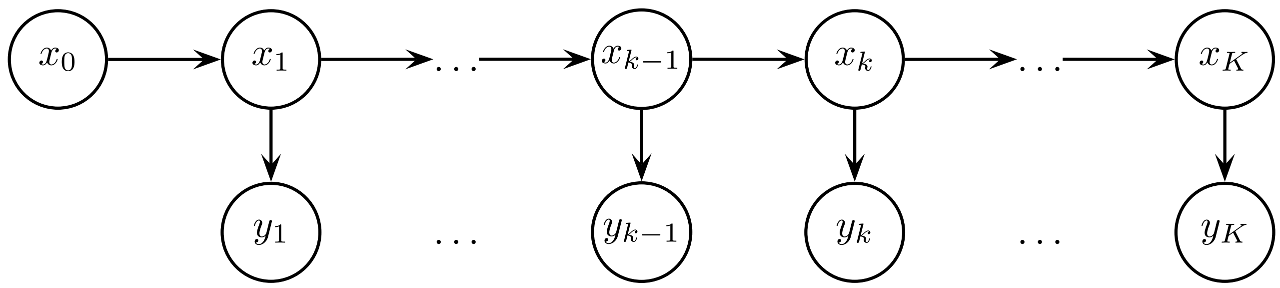
- Observation model $p(y_k|x_k)$

$$y_k \sim w\delta(y_k - x_k) + (1-w)u(1, N)$$

1. Setup a parameter structure

```
N = 50;     % Number of states

% Transition model;
ep = 0.5;  % Probability of not-moving
E = eye(N);
A = ep*E + (1-ep)*E(:, [2:N 1]); % Transition Matrix

% Observation model
w = 0.3; % Probability of observing true state
C = w*E + (1-w)*ones(N)/N; % Observation matrix

% Prior p(x_1)
pri = ones(N, 1)/N;

% Create a parameter structure
hm = struct('A', A, 'C', C, 'p_x1', pri);
```

2. Generate data from the true model

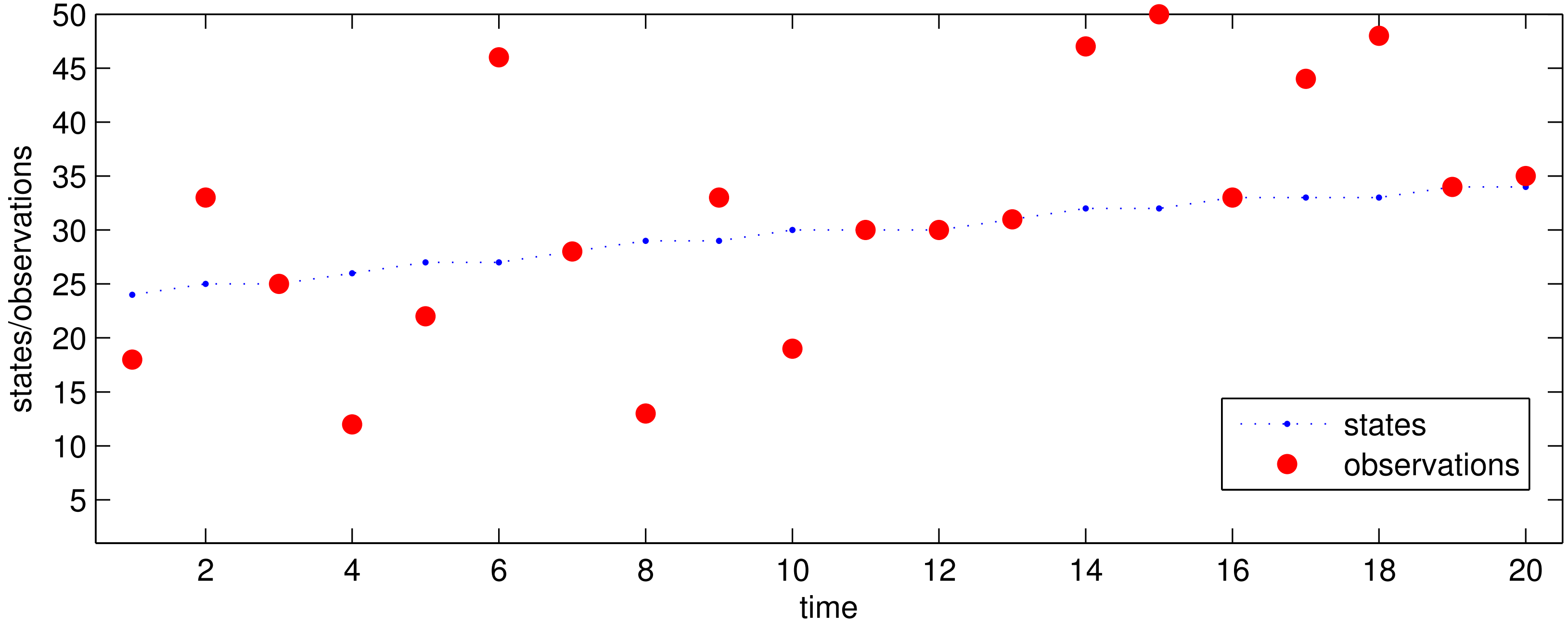


$$
\begin{aligned}
x_k | x_{k-1} &\sim p(x_k | x_{k-1}) \\
y_k | x_k &\sim p(y_k | x_k)
\end{aligned}
$$

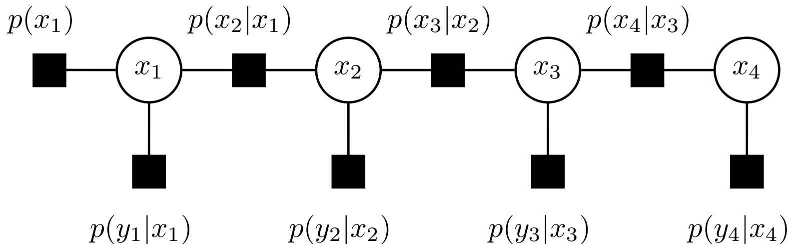
2. Generate data from the true model

```
function [obs, state] = hmm_generate_data(hm, K)
% Inputs :
%          hm : A HMM parameter structure
%           K : Number of time slices to simulate
% Outputs :
%           obs, state : Observations and the state trajectory

state = zeros(1, K);
obs = zeros(1, K);
for k=1:K,
    if k==1,
        state(k) = randgen(hm.p_x1);
    else
        state(k) = randgen(hm.A(:, state(k-1)));
    end;
    obs(k) = randgen(hm.C(:, state(k)));
end;
```

2. Generate data from the true model



3. Inference. Forward pass



- Predict

$$\begin{aligned}
\alpha_{k|k-1}(x_k) &= p(y_{1:k-1}, x_k) = \sum_{x_{k-1}} p(x_k|x_{k-1}) p(y_{1:k-1}, x_{k-1}) \\
&= \sum_{x_{k-1}} p(x_k|x_{k-1}) \alpha_{k-1|k-1}(x_{k-1})
\end{aligned}$$

- Update

$$\begin{aligned}
\alpha_{k|k}(x_k) &= p(y_{1:k}, x_k) = p(y_k|x_k) p(y_{1:k-1}, x_k) \\
&= p(y_k|x_k) \alpha_{k|k-1}(x_k)
\end{aligned}$$

$$
\begin{aligned}
p(y_{1:K}) &= \sum_{x_{1:K}} p(y_{1:K}|x_{1:K})p(x_{1:K}) \\
&= \sum_{x_K} p(y_K|x_K) \sum_{x_{K-1}} p(x_K|x_{K-1}) \cdots \sum_{x_2} p(x_3|x_2)p(y_2|x_2) \sum_{x_1} p(x_2|x_1) \underbrace{p(y_1|x_1) \overbrace{p(x_1)}^{\alpha_{1|0}}}_{\alpha_{1|1}} \\
&= \sum_{x_K} p(y_K|x_K) \sum_{x_{K-1}} p(x_K|x_{K-1}) \cdots \sum_{x_2} p(x_3|x_2)p(y_2|x_2) \sum_{x_1} p(x_2|x_1) \alpha_{1|1}(x_1) \\
&= \sum_{x_K} p(y_K|x_K) \sum_{x_{K-1}} p(x_K|x_{K-1}) \cdots \sum_{x_2} p(x_3|x_2)p(y_2|x_2) \alpha_{2|1}(x_2) \\
&= \sum_{x_K} p(y_K|x_K) \sum_{x_{K-1}} p(x_K|x_{K-1}) \cdots \sum_{x_2} p(x_3|x_2) \alpha_{2|2}(x_2) \\
&= \sum_{x_K} p(y_K|x_K) \sum_{x_{K-1}} p(x_K|x_{K-1}) \ldots \alpha_{3|2}(x_3)
\end{aligned}
$$

3. Inference: Forward pass

```
log_alpha = zeros(N, K);
log_alpha_predict = zeros(N, K);
for k=1:K,
    if k==1,
        log_alpha_predict(:,k) = log(hm.p_x1);
    else
        log_alpha_predict(:,k) ...
            = state_predict(hm.A, log_alpha(:, k-1));
    end;
    log_alpha(:, k) ...
          = state_update(hm.C(y(k), :), log_alpha_predict(:,k));
end;
```

3. Inference. Predict

```
function [lpp] = state_predict(A, log_p)
% STATE_PREDICT Computes A*p in log domain
%
%  [lpp] = state_predict(A, log_p)
%
% Inputs :
%  A : State transition matrix
%   log_p : log p(x_{k-1}, y_{1:k-1}) Filtered potential
%
% Outputs :
%   lpp : log p(x_{k}, y_{1:k-1});  Predicted potential

mx = max(log_p(:));  % Stable computation
p = exp(log_p - mx);
lpp = log(A*p) + mx;
```

Numerically Stable computation of $\log(\sum_i \exp(l_i)))$

- Derivation

$$
\begin{aligned}
L &= \log(\sum_i \exp(l_i)) \\
&= \log(\sum_i \exp(l_i)\frac{\exp(l^*)}{\exp(l^*)}) \\
&= \log(\exp(l^*) \sum_i \exp(l_i - l^*)) \\
&= l^* + \log(\sum_i \exp(l_i - l^*))
\end{aligned}
$$

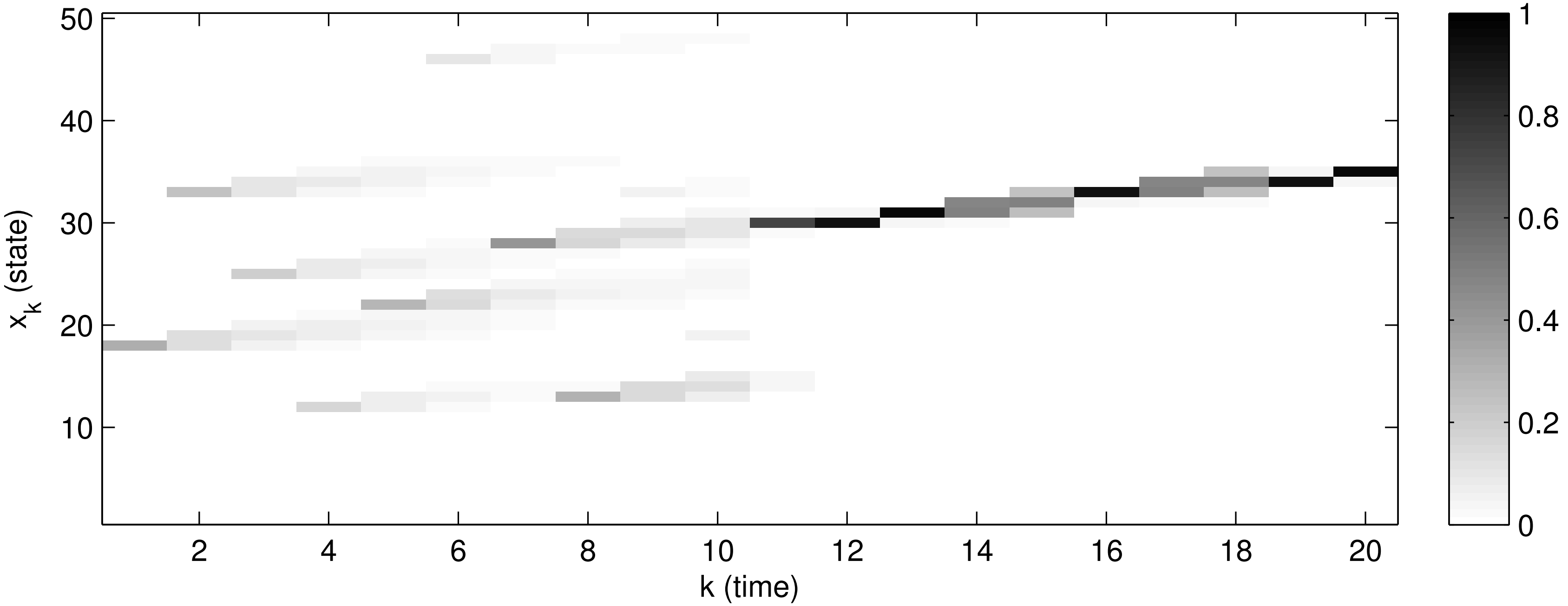
- We take l^* as the maximum $l^* = \max_i l_i$
- Assignment: Implement above as a function `logsumexp(l)`

3. Inference. Update

```
function [lup] = state_update(obs, log_p)
% STATE_UPDATE State update in log domain
%
%  [lup] = state_update(obs, log_p)
%
% Inputs :
%          obs : p(y_k| x_k)
%          log_p : log p(x_k, y_{1, k-1})
%
% Outputs :
% lup : log p(x_k, y_{1, k-1})  p(y_k| x_k)

lup = log(obs(:)) + log_p;
```

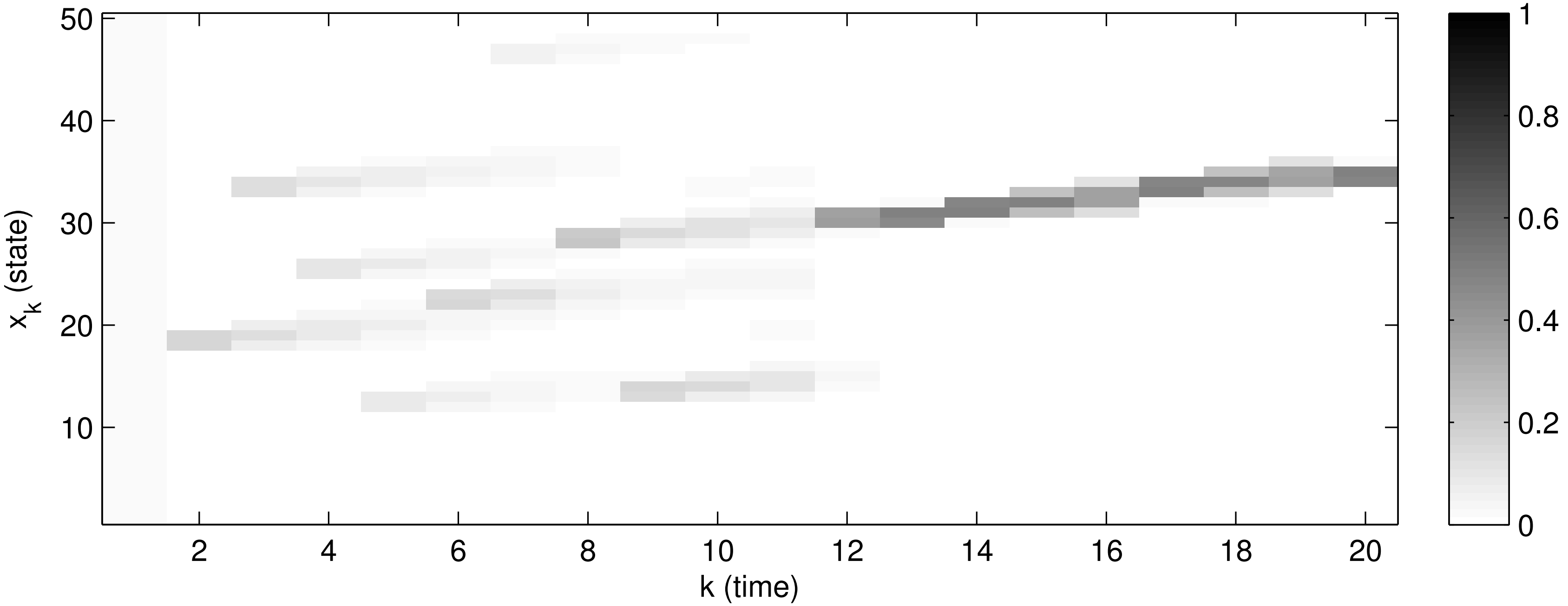
3. Inference. Forward pass.

$$\alpha_{k|k} \equiv p(y_{1:k}, x_k)$$

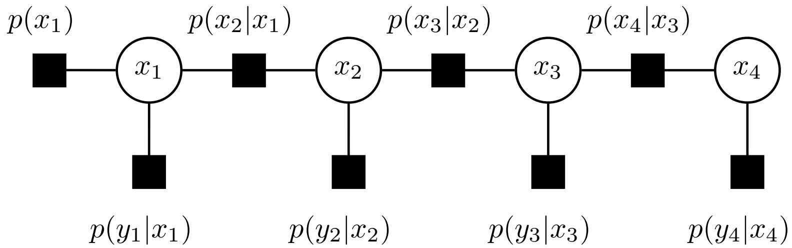


3. Inference. Forward pass

$$\alpha_{k|k-1} \equiv p(y_{1:k-1}, x_k)$$



3. Inference. Backward pass



- "Postdict"

$$
\begin{aligned}
{\color{blue}\beta_{k|k+1}(x_k)} &= p(y_{k+1:K}|x_k) = \sum_{x_{k+1}} p(x_{k+1}|x_k)p(y_{k+1:K}|x_{k+1}) \\
&= \sum_{x_{k+1}} p(x_{k+1}|x_k){\color{red}\beta_{k+1|k+1}(x_{k+1})}
\end{aligned}
$$

- Update

$$
\begin{aligned}
{\color{red}\beta_{k|k}(x_k)} &= p(y_{k:K}|x_k) = p(y_k|x_k)p(y_{k+1:K}|x_k) \\
&= p(y_k|x_k){\color{blue}\beta_{k|k+1}(x_k)}
\end{aligned}
$$

$$
\begin{aligned}
p(y_{1:K}) &= \sum_{x_1} p(x_1)p(y_1|x_1)\dots\sum_{x_{K-1}} p(x_{K-1}|x_{K-2})p(y_{K-1}|x_{K-1})\sum_{x_K} p(x_K|x_{K-1})p(y_K|x_K)\underbrace{\mathbf{1}}_{\beta_{K|K+1}} \\
&= \sum_{x_1} p(x_1)p(y_1|x_1)\cdots\sum_{x_{K-1}} p(x_{K-1}|x_{K-2})p(y_{K-1}|x_{K-1})\sum_{x_K} p(x_K|x_{K-1})\beta_{K|K} \\
&= \sum_{x_1} p(x_1)p(y_1|x_1)\cdots\sum_{x_{K-1}} p(x_{K-1}|x_{K-2})p(y_{K-1}|x_{K-1})\beta_{K-1|K} \\
&= \sum_{x_1} p(x_1)p(y_1|x_1)\cdots\sum_{x_{K-1}} p(x_{K-1}|x_{K-2})\beta_{K-1|K-1} \\
&= \sum_{x_1} p(x_1)p(y_1|x_1)\dots\beta_{K-2|K-1}
\end{aligned}
$$

3. Inference. Backward pass

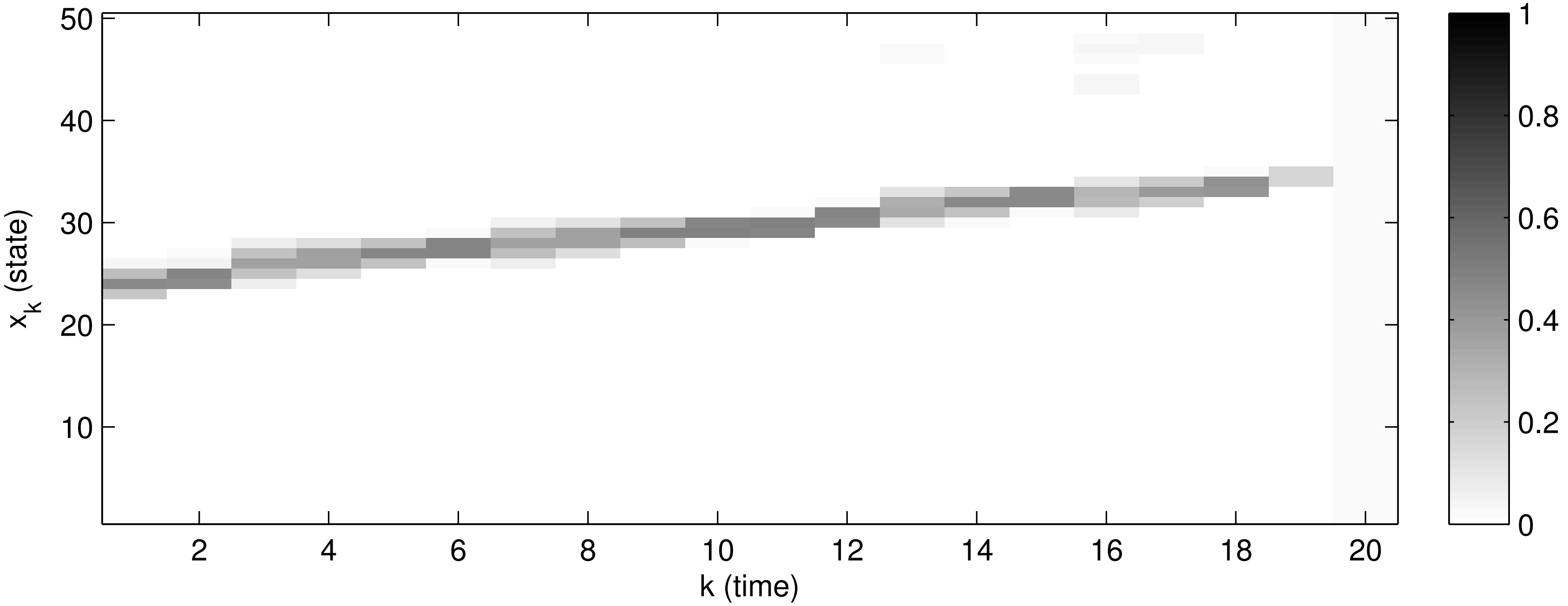
```
log_beta = zeros(N, T);
log_beta_postdict = zeros(N, T);
for t=T:-1:1,
    if t==T,
        log_beta_postdict(:,t) = zeros(N,1);
    else
        log_beta_postdict(:,t) ...
               = state_postdict(hm.A, log_beta(:, t+1));
    end;
    log_beta(:, t) ...
         = state_update(hm.C(y(t), :), log_beta_postdict(:,t));
end;
```

3. Inference. Postdict.

```
function [lpp] = state_postdict(A, log_p)
% STATE_POSTDICT Computes A'*p in log domain
%
%  [lpp] = state_postdict(A, log_p)
%
% Inputs :
% A : State transition matrix
%          log_p : log p(y_{k+1:K}|x_{k+1})    Updated potential
%
% Outputs :
% lpp : log p(y_{k+1:K}| x_k)    Postdicted potential

mx = max(log_p(:));  % Stable computation
p = exp(log_p - mx);
lpp = log(A'*p) + mx;
```

3. Inference. Backward pass

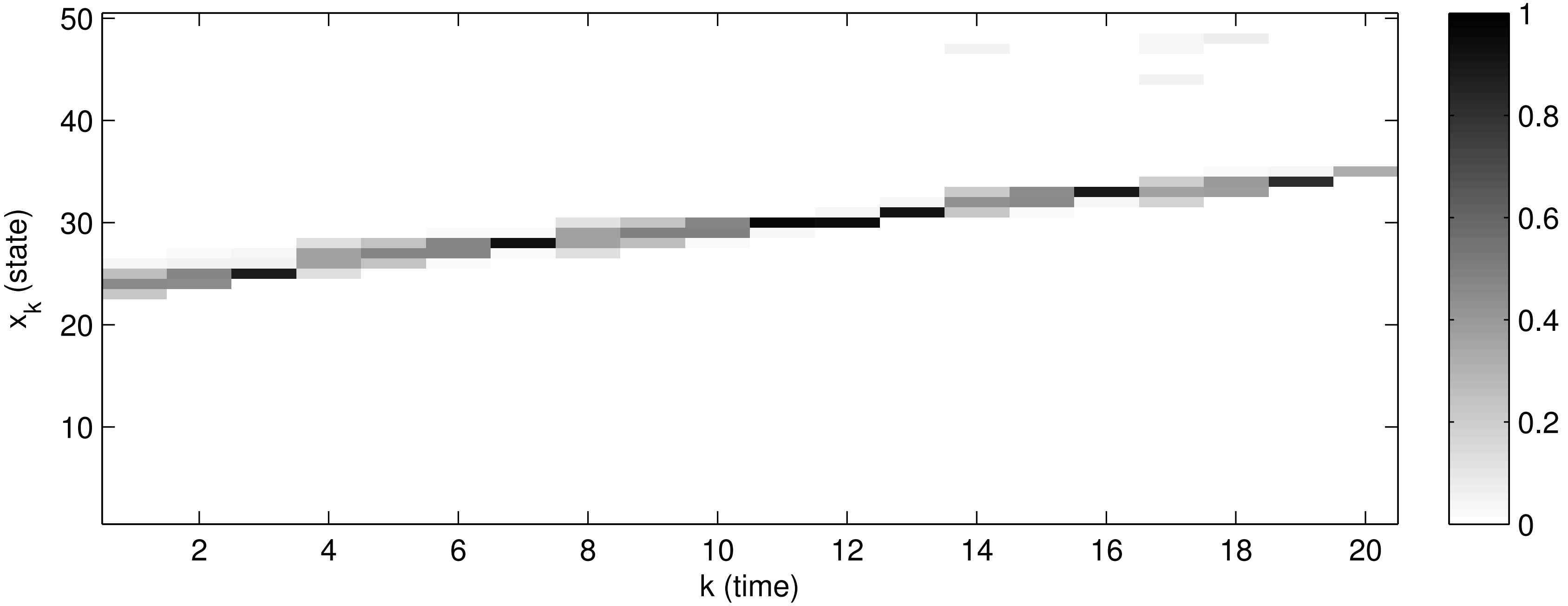
$$\beta_{k|k+1}(x_k) \quad = \quad p(y_{k+1:K}|x_k)$$



We visualise $\hat{\beta} \propto \beta_{k|k+1}(x_k)u(x_k)$

3. Inference. Backward pass

$$\color{red}{\beta_{k|k}(x_k)} \quad = \quad p(y_{k:K}|x_k)$$



3. Inference. Smoothing.

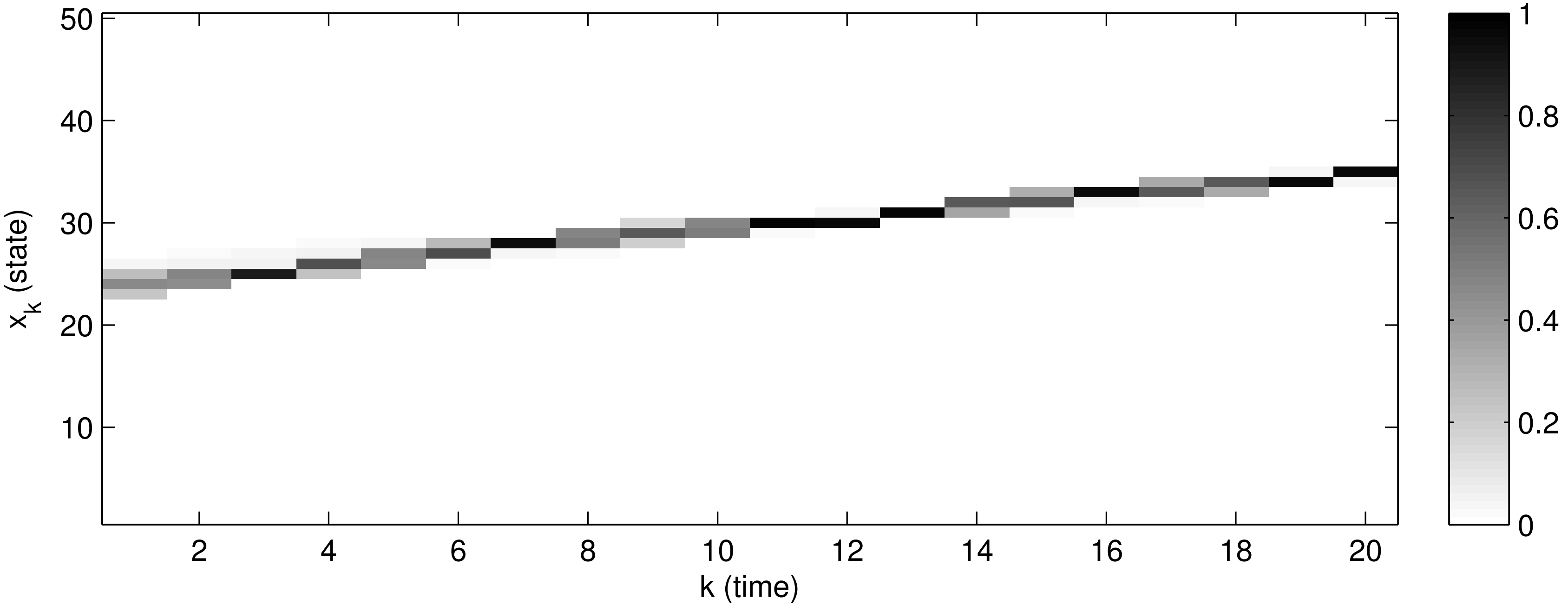
$$
\begin{aligned}
p(y_{1:K}, x_k) &= p(y_{1:k}, x_k)p(y_{k+1:K}|x_k) \\
&= \alpha_{k|k}(x_k)\beta_{k|k+1}(x_k) \\
&\equiv \gamma_k(x_k)
\end{aligned}
$$

Alternatives

$$
\begin{aligned}
\gamma_k(x_k) &= \alpha_{k|k-1}(x_k)\beta_{k|k}(x_k) \\
&= \alpha_{k|k-1}(x_k)p(y_k|x_k)\beta_{k|k+1}(x_k)
\end{aligned}
$$

3. Inference. Smoothing.

$$p(x_k|y_{1:K}) \quad \propto \quad p(y_{1:K}, x_k) = {\color{red}\alpha_{k|k}(x_k)}{\color{blue}\beta_{k|k+1}(x_k)} \equiv \gamma_k(x_k)$$



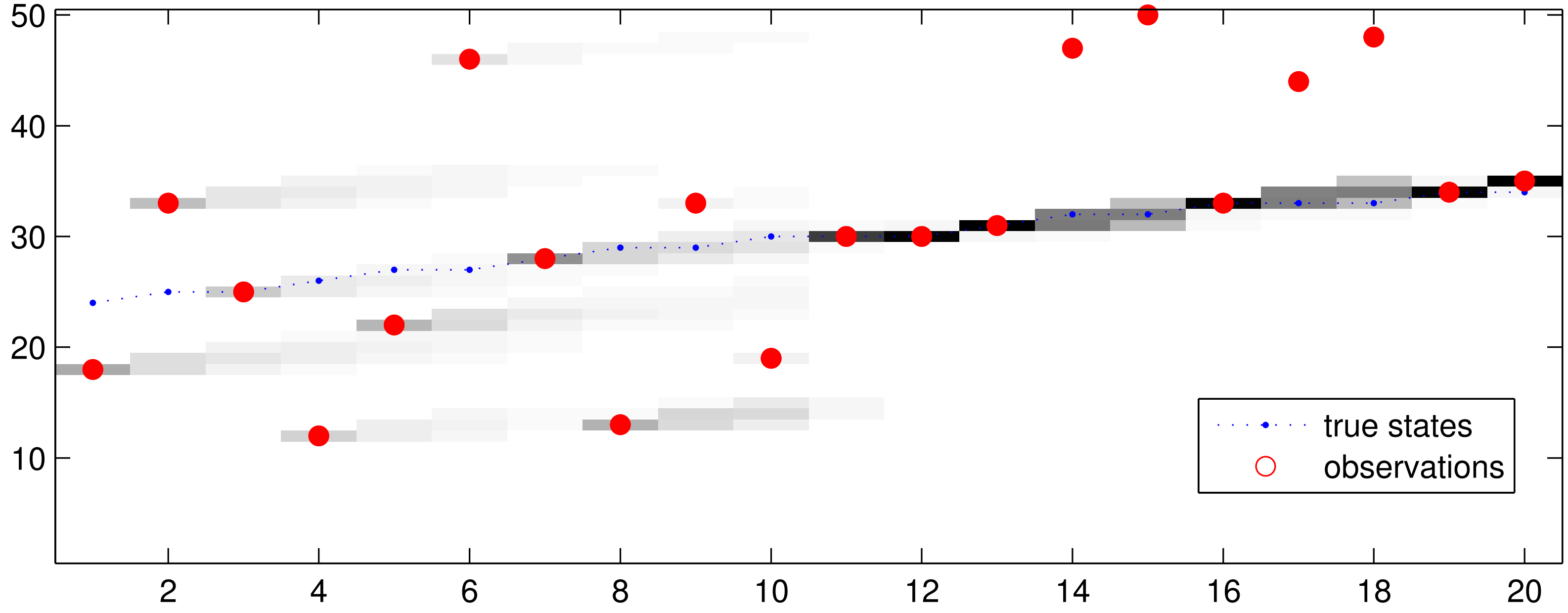
3. Inference. Smoothing.

```
log_gamma = log_alpha + log_beta_postdict
```

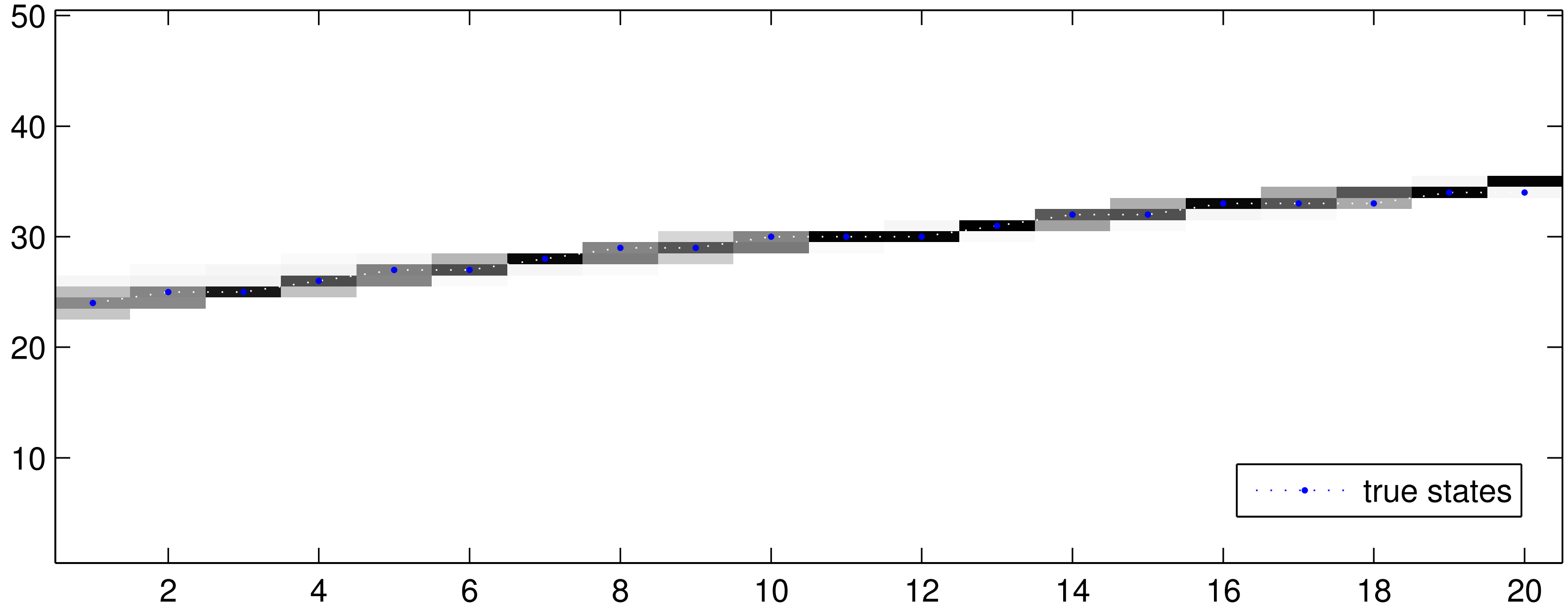
4. Test and Visualisation

```
imagesc(normalize_exp(log_gamma, 1));
set(gca, 'ydir', 'n');
colormap(flipud(gray));
xlabel('k (time)'); ylabel('x_k (state)');
caxis([0 1]);
colorbar

% This has to be constant !! (why)
plot(log_sum_exp(log_gamma, 1));
```

4. Test and Visualise. Filter.



4. Test and Visualise. Smoother.



Outline

- Bayesian Inference Review

- Mean Field, Variational Bayes

Variational Formulation

A simple but very powerful idea:

- Represent the solution of a problem as the minimum of some cost function

- Example: Solving a system of linear equations $p \in \mathcal{X}$

$$Ap = b$$

- Variational formulation

$$p = \arg\min_q \underbrace{\left\{ \frac{1}{2}(b - Aq)^\top (b - Aq) \right\}}_{\mathcal{F}(q)}$$

Variational Formulation

- We can also find approximate solutions
- Suppose we constrain q to a subset

$$q \in \mathcal{X}_q \subset \mathcal{X}$$

- We trivially have

$$\mathcal{F}(p) \;\; = \;\; \min_{q \in \mathcal{X}} \{\mathcal{F}(q)\} \leq \min_{q \in \mathcal{X}_q} \{\mathcal{F}(q)\}$$

Example: Computing Marginals

- Consider a joint distribution $i, j \in \{0, 1\}$

$$p(x_1 = i, x_2 = j) \quad = \quad \pi_{i,j}$$

$p(x_1, x_2)$	$x_2 = 0$	$x_2 = 1$
$x_1 = 0$	$\pi_{0,0}$	$\pi_{0,1}$
$x_1 = 1$	$\pi_{1,0}$	$\pi_{1,1}$

- Marginals

$p(x_1)$	
$x_1 = 0$	$\pi_{0,0} + \pi_{0,1}$
$x_1 = 1$	$\pi_{1,0} + \pi_{1,1}$

$p(x_2)$	$x_2 = 0$	$x_2 = 1$
	$\pi_{0,0} + \pi_{1,0}$	$\pi_{0,1} + \pi_{1,1}$

- How can we express the marginals of a density variationally ?

Example: Computing Marginals

- Take a factorised Distribution

$$
\begin{aligned}
q(x_1 = i, x_2 = j) &= q(x_1 = i)q(x_2, = j) \\
q(x_1 = 1) &= q_1 \\
q(x_2 = 1) &= q_2
\end{aligned}
$$

$q(x_1, x_2)$	$x_2 = 0$	$x_2 = 1$
$x_1 = 0$	$(1 - q_1)(1 - q_2)$	$(1 - q_1)q_2$
$x_1 = 1$	$q_1(1 - q_2)$	$q_1 q_2$

- Compute the “distance” between p and q via **Kullback-Leibler (KL) Divergence**

Kullback-Leibler (KL) Divergence

- A “quasi-distance” between two distributions $\mathcal{P} = p(x)$ and $\mathcal{Q} = q(x)$.

$$KL(\mathcal{P}||\mathcal{Q}) \equiv \int_{\mathcal{X}} dx p(x) \log \frac{p(x)}{q(x)} = \langle \log \mathcal{P} \rangle_{\mathcal{P}} - \langle \log \mathcal{Q} \rangle_{\mathcal{P}}$$

- Unlike a metric, (in general) it is not symmetric,

$$KL(\mathcal{P}||\mathcal{Q}) \neq KL(\mathcal{Q}||\mathcal{P})$$

- But it is non-negative (by Jensen’s Inequality)

$$\begin{aligned} KL(\mathcal{P}||\mathcal{Q}) &= -\int_{\mathcal{X}} dx p(x) \log \frac{q(x)}{p(x)} \\ &\geq -\log \int_{\mathcal{X}} dx p(x) \frac{q(x)}{p(x)} = -\log \int_{\mathcal{X}} dx q(x) = -\log 1 = 0 \end{aligned}$$

Kullback-Leibler (KL) Divergence

$p(x_1, x_2)$	$x_2 = 0$	$x_2 = 1$
$x_1 = 0$	$\pi_{0,0}$	$\pi_{0,1}$
$x_1 = 1$	$\pi_{1,0}$	$\pi_{1,1}$

$q(x_1, x_2)$	$x_2 = 0$	$x_2 = 1$
$x_1 = 0$	$(1 - q_1)(1 - q_2)$	$(1 - q_1)q_2$
$x_1 = 1$	$q_1(1 - q_2)$	$q_1 q_2$

$$
\begin{aligned}
KL(p||q) &= \sum_{x_1} \sum_{x_2} p(x_1, x_2) \log \left(\frac{p(x_1, x_2)}{q(x_1, x_2)} \right) \\
&= \sum_{i} \sum_{j} \pi_{i,j} \log \left(\frac{\pi_{i,j}}{q(x_1 = i, x_2 = j)} \right) \\
&= \pi_{0,0} \log \left(\frac{\pi_{0,0}}{(1 - q_1)(1 - q_2)} \right) + \pi_{1,0} \log \left(\frac{\pi_{1,0}}{q_1(1 - q_2)} \right) \\
&\quad + \pi_{0,1} \log \left(\frac{\pi_{0,1}}{(1 - q_1)q_2} \right) + \pi_{1,1} \log \left(\frac{\pi_{1,1}}{q_1 q_2} \right)
\end{aligned}
$$

Kullback-Leibler (KL) Divergence

- Let us minimise the KL divergence w.r.t. q_1

$$
\begin{aligned}
KL(p||q) &= -\pi_{0,0}(\log(1-q_1) + \log(1-q_2)) - \pi_{1,0}(\log q_1 + \log(1-q_2)) \\
&\quad -\pi_{0,1}(\log(1-q_1) + \log q_2) - \pi_{1,1}(\log q_1 + \log q_2) \\
&\quad + \sum_i \sum_j \pi_{i,j} \log \pi_{i,j}
\end{aligned}
$$

- We take the derivative and set to zero

$$
\frac{\partial KL(p||q)}{\partial q_1} = \frac{\partial}{\partial q_1}(-\pi_{0,0}\log(1-q_1) - \pi_{1,0}\log q_1 - \pi_{0,1}\log(1-q_1) - \pi_{1,1}\log q_1)
$$

The marginal is the minimiser of $KL(p||q)$

$$
\begin{aligned}
0 &= \pi_{0,0}\frac{1}{(1-q_1)} - \pi_{1,0}\frac{1}{q_1} + \pi_{0,1}\frac{1}{(1-q_1)} - \pi_{1,1}\frac{1}{q_1} \\
&= (\pi_{0,0} + \pi_{0,1})\frac{1}{(1-q_1)} - (\pi_{1,0} + \pi_{1,1})\frac{1}{q_1}
\end{aligned}
$$

$$
\begin{aligned}
q_1 &= \frac{(\pi_{1,0} + \pi_{1,1})}{(\pi_{0,0} + \pi_{0,1} + \pi_{1,0} + \pi_{1,1})} = \pi_{1,0} + \pi_{1,1} = p(x_1 = 1) \\
1 - q_1 &= 1 - (\pi_{1,0} + \pi_{1,1}) = \pi_{0,0} + \pi_{0,1} = 1 - q_1 = p(x_1 = 0)
\end{aligned}
$$

The derivation for q_2 is identical.

The "other" one: $KL(q||p)$

$$
\begin{aligned}
KL(q||p) &= \sum_{x_1}\sum_{x_2} q(x_1, x_2) \log\left(\frac{q(x_1, x_2)}{p(x_1, x_2)}\right) \\
&= \sum_{i}\sum_{j} q(x_1 = i, x_2 = j) \log\left(\frac{q(x_1 = i, x_2 = j)}{\pi_{i,j}}\right) \\
&= (1 - q_1)(1 - q_2) \log\left(\frac{(1 - q_1)(1 - q_2)}{\pi_{0,0}}\right) + q_1(1 - q_2) \log\left(\frac{q_1(1 - q_2)}{\pi_{1,0}}\right) \\
&\quad + (1 - q_1)q_2 \log\left(\frac{(1 - q_1)q_2}{\pi_{0,1}}\right) + q_1 q_2 \log\left(\frac{q_1 q_2}{\pi_{1,1}}\right)
\end{aligned}
$$

The “other” one: $KL(q||p)$

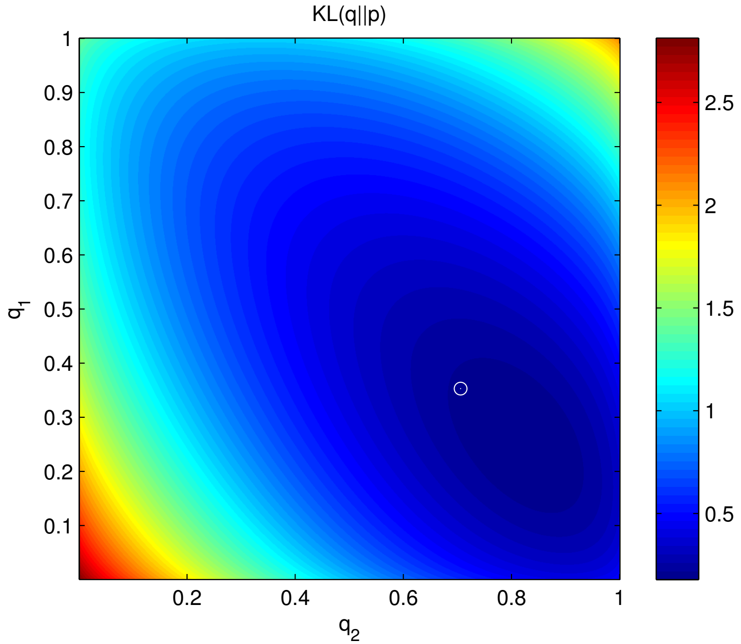
$$
\begin{aligned}
\frac{\partial KL(q||p)}{\partial q_1} &= (-\log(1-q_1) + \log \pi_{0,0} + \log q_1 - \log \pi_{1,0}) \\
& q_2 \left(-\log \pi_{0,0} + \log \pi_{1,0} + \log \pi_{0,1} - \log \pi_{1,1}\right)
\end{aligned}
$$

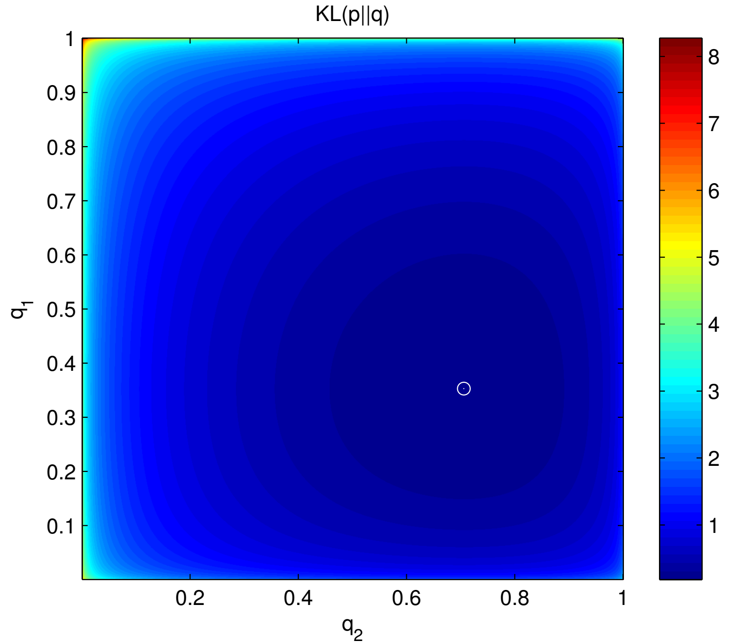
The “other” one: $KL(q||p)$

$$
\begin{aligned}
\mathcal{Q}_1 &= \begin{pmatrix} 1-q_1 \\ q_1 \end{pmatrix} = \frac{1}{Z_1} \begin{pmatrix} \pi_{0,0}^{(1-q_2)} \pi_{0,1}^{q_2} \\ \pi_{1,0}^{(1-q_2)} \pi_{1,1}^{q_2} \end{pmatrix} \\
&\propto \begin{pmatrix} \exp((1-q_2)\log \pi_{0,0} + q_2 \log \pi_{0,1}) \\ \exp((1-q_2)\log \pi_{1,0} + q_2 \log \pi_{1,1}) \end{pmatrix} \\
&= \begin{pmatrix} \exp((1-q_2)\log \pi_{0,0} + q_2 \log \pi_{0,1}) \\ \exp((1-q_2)\log \pi_{1,0} + q_2 \log \pi_{1,1}) \end{pmatrix} \\
&\equiv \exp(\langle \log \pi \rangle_{\mathcal{Q}_2})
\end{aligned}
$$

$$
\mathcal{Q}_2 \quad \propto \quad \exp(\langle \log \pi \rangle_{\mathcal{Q}_1})
$$

$KL(q||p)$ **versus** $KL(p||q)$





Variational Bayes (VB), mean field

We will approximate the posterior $\mathcal{P}$ with a simpler distribution $\mathcal{Q}$.

$$\begin{aligned}
\mathcal{P} &= \frac{1}{Z_x} p(x = \hat{x}|s_1, s_2)p(s_1)p(s_2) \\
\mathcal{Q} &= q(s_1)q(s_2)
\end{aligned}$$

Here, we choose

$$q(s_1) = \mathcal{N}(s_1; m_1, S_1) \qquad\qquad q(s_2) = \mathcal{N}(s_2; m_2, S_2)$$

A “measure of fit” between distributions is the KL divergence

Kullback-Leibler (KL) Divergence

- A “quasi-distance” between two distributions $\mathcal{P} = p(x)$ and $\mathcal{Q} = q(x)$.

$$KL(\mathcal{P}||\mathcal{Q}) \equiv \int_{\mathcal{X}} dx p(x) \log \frac{p(x)}{q(x)} = \langle \log \mathcal{P} \rangle_{\mathcal{P}} - \langle \log \mathcal{Q} \rangle_{\mathcal{P}}$$

- Unlike a metric, (in general) it is not symmetric,

$$KL(\mathcal{P}||\mathcal{Q}) \neq KL(\mathcal{Q}||\mathcal{P})$$

- But it is non-negative (by Jensen’s Inequality)

$$\begin{aligned} KL(\mathcal{P}||\mathcal{Q}) &= -\int_{\mathcal{X}} dx p(x) \log \frac{q(x)}{p(x)} \\ &\geq -\log \int_{\mathcal{X}} dx p(x) \frac{q(x)}{p(x)} = -\log \int_{\mathcal{X}} dx q(x) = -\log 1 = 0 \end{aligned}$$

The form of the mean field solution

$$
\begin{aligned}
0 &\leq \langle \log q(s_1)q(s_2) \rangle_{q(s_1)q(s_2)} + \log Z_x - \langle \log \phi(s_1, s_2) \rangle_{q(s_1)q(s_2)} \\
\log Z_x &\geq \langle \log \phi(s_1, s_2) \rangle_{q(s_1)q(s_2)} - \langle \log q(s_1)q(s_2) \rangle_{q(s_1)q(s_2)} \\
&\equiv -F(p; q) + H(q) \qquad (1)
\end{aligned}
$$

Here, F is the *energy* and H is the *entropy*. We need to maximize the right hand side.

$$\text{Evidence} \geq -\text{Energy} + \text{Entropy}$$

Note r.h.s. is a **lower bound** [**?**]. The mean field equations **monotonically** increase this bound. Good for assessing convergence and debugging computer code.

Details of derivation

- Define the Lagrangian

$$\begin{aligned}\Lambda &= \int ds_1 q(s_1)\log q(s_1) + \int ds_2 q(s_2)\log q(s_2) + \log Z_x - \int ds_1 ds_2 q(s_1)q(s_2)\log\phi(s_1,s_2) \\ &\quad +\lambda_1(1-\int ds_1 q(s_1)) + \lambda_2(1-\int ds_2 q(s_2)) \qquad (2)\end{aligned}$$

- Calculate the functional derivatives w.r.t. $q(s_1)$ and set to zero

$$\frac{\delta}{\delta q(s_1)}\Lambda = \log q(s_1) + 1 - \langle \log\phi(s_1,s_2)\rangle_{q(s_2)} - \lambda_1$$

- Solve for $q(s_1)$,

$$\begin{aligned}\log q(s_1) &= \lambda_1 - 1 + \langle \log\phi(s_1,s_2)\rangle_{q(s_2)} \\ q(s_1) &= \exp(\lambda_1 - 1)\exp(\langle \log\phi(s_1,s_2)\rangle_{q(s_2)}) \qquad (3)\end{aligned}$$

- Use the fact that

$$\begin{aligned}1 &= \int ds_1 q(s_1) = \exp(\lambda_1 - 1)\int ds_1 \exp(\langle \log\phi(s_1,s_2)\rangle_{q(s_2)}) \\ \lambda_1 &= 1 - \log\int ds_1 \exp(\langle \log\phi(s_1,s_2)\rangle_{q(s_2)})\end{aligned}$$

The form of the solution

- No direct analytical solution
- We obtain fixed point equations in closed form

$$
\begin{aligned}
q(s_1) &\propto \exp(\langle \log \phi(s_1, s_2) \rangle_{q(s_2)}) \\
q(s_2) &\propto \exp(\langle \log \phi(s_1, s_2) \rangle_{q(s_1)})
\end{aligned}
$$

Note the nice symmetry

Direct Link to Expectation-Maximisation (EM)

Suppose we choose one of the distributions degenerate, i.e.

$$\tilde{q}(s_2) = \delta(s_2 - \tilde{m})$$

where $\tilde{m}$ corresponds to the “location parameter” of $\tilde{q}(s_2)$. We need to find the closest degenerate distribution to the actual mean field solution $q(s_2)$, hence we take one more KL and minimize

$$\tilde{m} = \operatorname*{argmin}_{\xi} KL(\delta(s_2 - \xi)||q(s_2))$$

It can be shown that this leads exactly to the EM fixed point iterations.

Iterated Conditional Modes (ICM)

If we choose both distributions degenerate, i.e.

$$
\begin{aligned}
\tilde{q}(s_1) &= \delta(s_1 - \tilde{m}_1) \\
\tilde{q}(s_2) &= \delta(s_2 - \tilde{m}_2)
\end{aligned}
$$

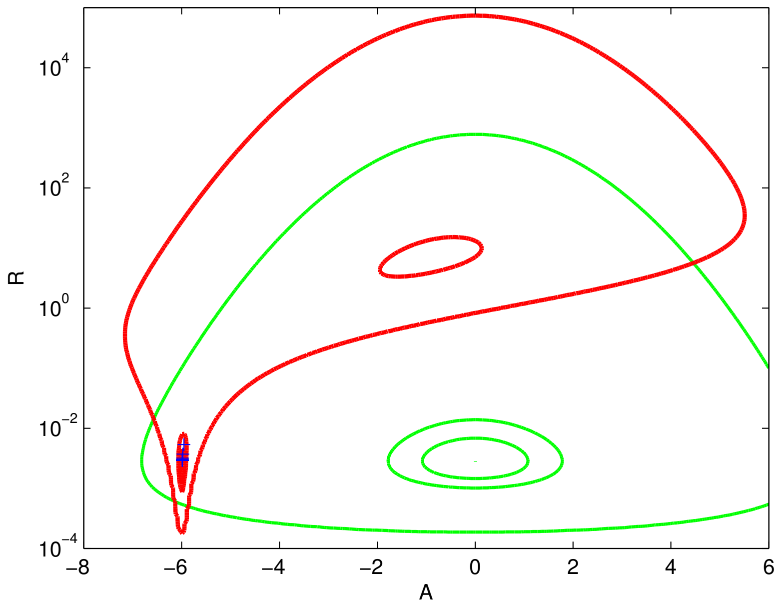
It can be shown that this leads exactly to the ICM fixed point iterations. This algorithm is equivalent to coordinate ascent in the original posterior surface $\phi(s_1, s_2)$.

$$
\begin{aligned}
\tilde{m}_1 &= \underset{s_1}{\operatorname{argmax}}\, \phi(s_1, s_2 = \tilde{m}_2) \\
\tilde{m}_2 &= \underset{s_2}{\operatorname{argmax}}\, \phi(s_1 = \tilde{m}_1, s_2)
\end{aligned}
$$

ICM, EM, VB ...

For OSSS, all algorithms are identical. This is in general not true.

While algorithmic details are very similar, there can be big qualitative differences in terms of fixed points.



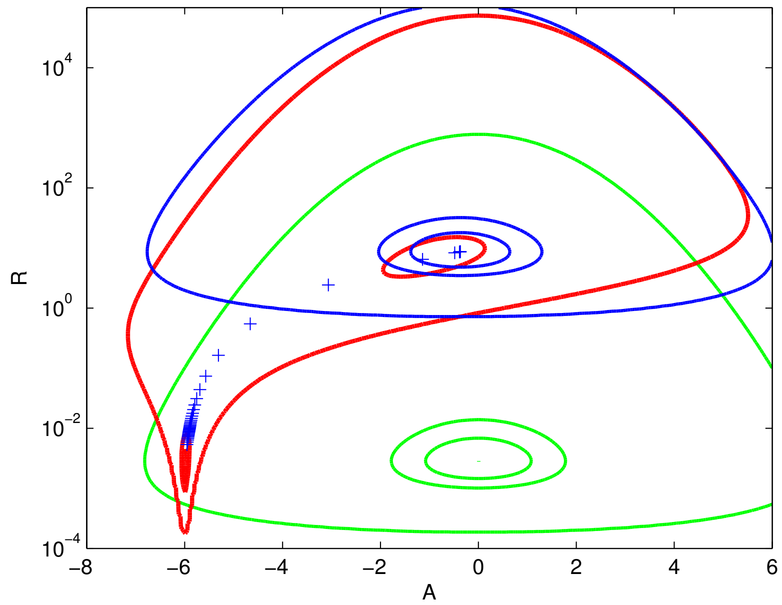


Figure 1: Left, ICM, Right VB. EM is similar to ICM in this AR(1) example.

Some References

Text Books:

- Bayesian Reasoning and Machine Learning, David Barber, 2012, CUP Online
- Pattern Recognition and Machine Learning, Christopher Bishop, 2006 Springer
- Machine Learning, A Probabilistic Perspective, Kevin P. Murphy, 2012 MIT Press

Some References

Bayesian Time Series, Monte Carlo

- A. T. Cemgil, A Tutorial Introduction to Monte Carlo methods, Markov Chain Monte Carlo and Particle Filtering, 2012. (https://dl.dropboxusercontent.com/u/9787379/cmpe58n/cmpe58n-lecture-notes.pdf)
- D. Barber, A. T. Cemgil and S. Chiappa, Bayesian Time Series Models. Cambridge University Press, 2011.
- D Barber and A. T. Cemgil, Graphical Models for Time Series, IEEE Signal Processing Magazine, Special issue on graphical models, vol. 27, no. 6, pp. 18-28, October 2010.

Some References

Recent Trends

- Z. Ghahramani, Probabilistic machine learning and artificial intelligence, Nature, 2015, doi:10.1038/nature14541
  - probabilistic programming,
  - Bayesian optimization,
  - data compression
  - automatic model discovery