CMPE 547
Bayesian Statistics and Machine Learning

A. Taylan Cemgil

Dept. of Computer Engineering
Boğaziçi University

A simple problem

Die 1: $\lambda \in \{\epsdice{1}, \epsdice{2}, \epsdice{3}, \epsdice{4}, \epsdice{5}, \epsdice{6}\}$

Die 2: $y \in \{\epsdice[black]{1}, \epsdice[black]{2}, \epsdice[black]{3}, \epsdice[black]{4}, \epsdice[black]{5}, \epsdice[black]{6}\}$

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

What is λ when $\mathcal{D} = 9$?

A simple problem

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

$\mathcal{D} = \lambda + y$	$y =$ ⚀	$y =$ ⚁	$y =$ ⚂	$y =$ ⚃	$y =$ ⚄	$y =$ ⚅
~~$\lambda =$ ⚀~~	2	3	4	5	6	7
~~$\lambda =$ ⚁~~	3	4	5	6	7	8
$\lambda =$ ⚂	4	5	6	7	8	**9**
$\lambda =$ ⚃	5	6	7	8	**9**	10
$\lambda =$ ⚄	6	7	8	**9**	10	11
$\lambda =$ ⚅	7	8	**9**	10	11	12

Bayes' Theorem

Thomas Bayes (1702-1761)

What you know about a parameter $\boldsymbol{\lambda}$ after the data $\mathcal{D}$ arrive is what you knew before about $\boldsymbol{\lambda}$ and what the data $\mathcal{D}$ told you.

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

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}}$$

“Bureaucratical” derivation

Formally we write

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

$$
\begin{aligned}
p(\lambda, y|\mathcal{D}) &= \frac{1}{p(\mathcal{D})} \times p(\mathcal{D}|\lambda, y) \times p(y)p(\lambda) \\
\text{Posterior} &= \frac{1}{\text{Evidence}} \times \text{Likelihood} \times \text{Prior}
\end{aligned}
$$

Kronecker delta function denoting a degenerate (deterministic) distribution $\delta(x) = \begin{cases} 1 & x = 0 \\ 0 & x \neq 0 \end{cases}$

Prior

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

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

- A table with indicies λ and y
- Each cell denotes the probability $p(\lambda, y)$

Likelihood

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

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

- A table with indicies λ and y
- The likelihood is **not** a probability distribution, but a positive function.

Likelihood $\times$ Prior

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

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

Evidence (= Marginal Likelihood)

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

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

Posterior

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

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

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

Marginal Posterior

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

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

The "proportional to" notation

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

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

Another application of Bayes' Theorem: "Model Selection"

Given an unknown number of fair dice with outcomes $\lambda_1, \lambda_2, \dots, \lambda_n$,

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

How many dice are there when $\mathcal{D} = 9$?

Assume that any number n is equally likely *a-priori*

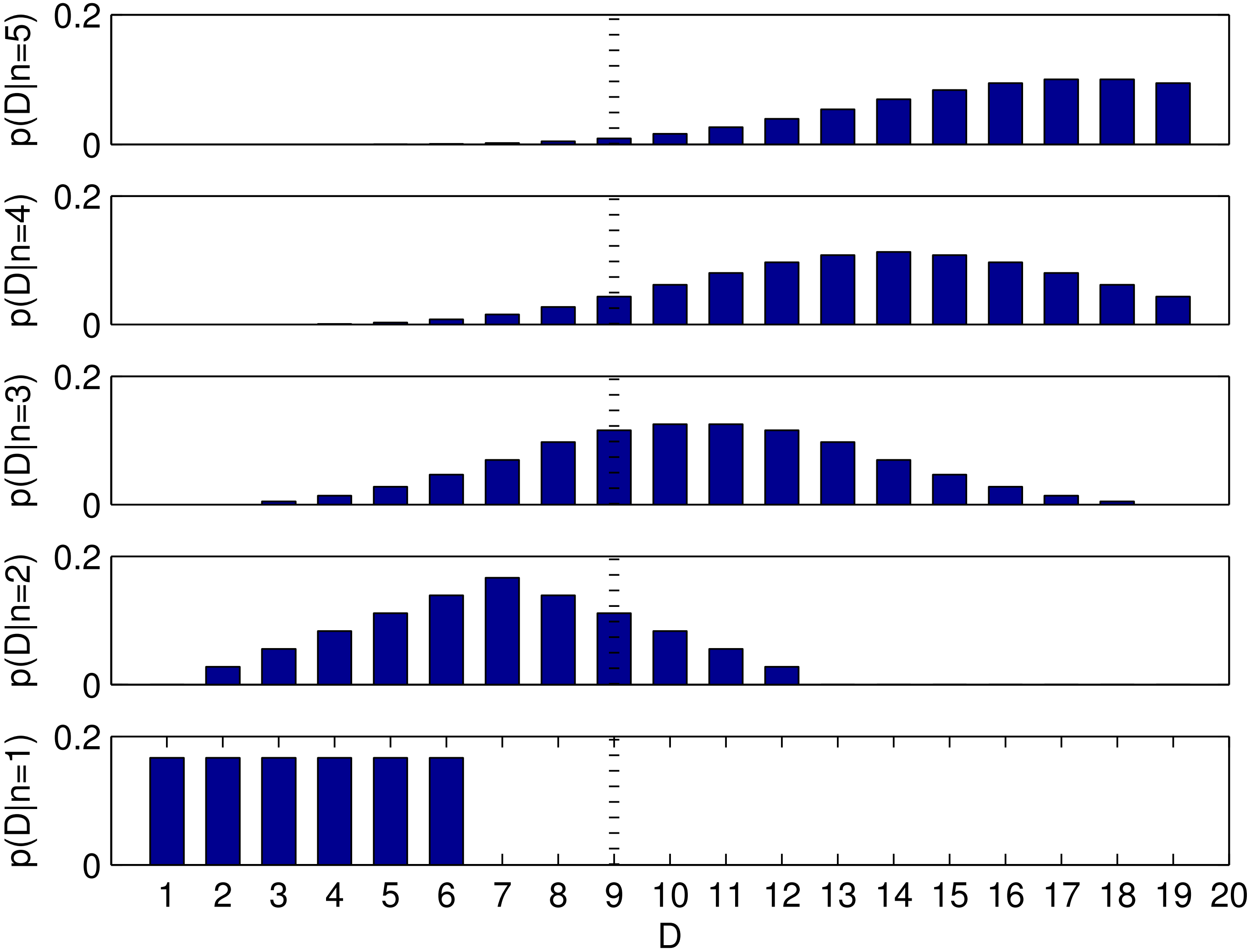
Another application of Bayes' Theorem: "Model Selection"

Given all n are equally likely (i.e., $p(n)$ is flat), we calculate (formally)

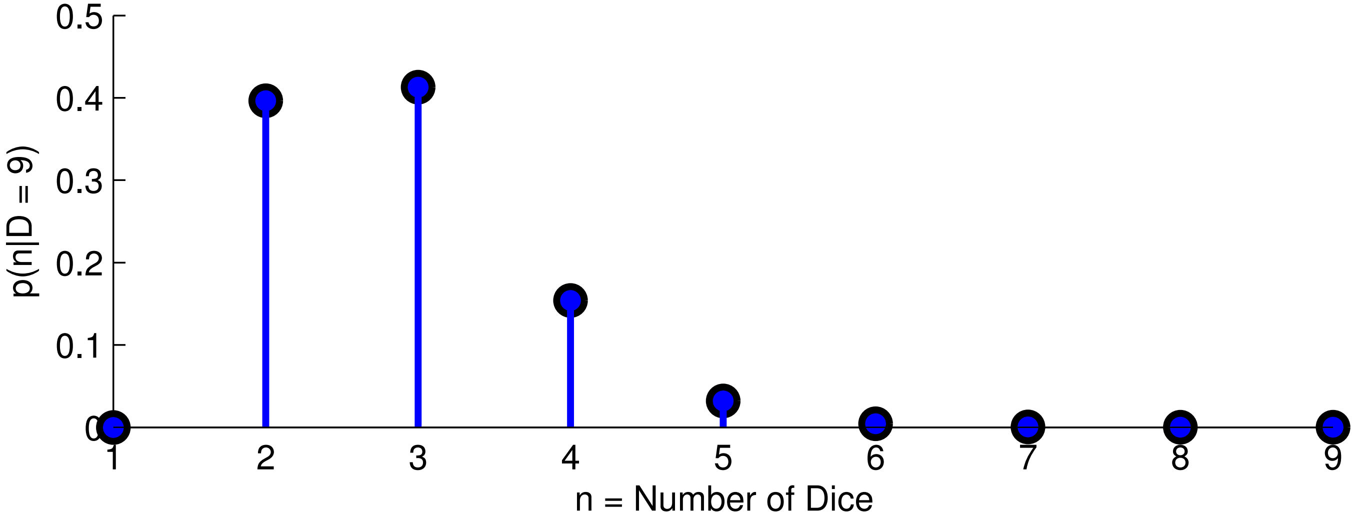
$$p(n|\mathcal{D} = 9) = \frac{p(\mathcal{D} = 9|n)p(n)}{p(\mathcal{D})} \propto p(\mathcal{D} = 9|n)$$

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

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



Another application of Bayes' Theorem: "Model Selection"



- Complex models are more flexible but they spread their probability mass
- Bayesian inference inherently prefers "simpler models" – Occam's razor
- Computational burden: We need to sum over all parameters λ

Probabilistic Inference

A huge spectrum of applications – all boil down to computation of

- **expectations** of functions under probability distributions: **Integration**

$$\langle f(x) \rangle = \int_{\mathcal{X}} dx p(x) f(x) \qquad \langle f(x) \rangle = \sum_{x \in \mathcal{X}} p(x) f(x)$$

- **modes** of functions under probability distributions: **Optimization**

$$x^* = \operatorname*{argmax}_{x \in \mathcal{X}} p(x) f(x)$$

- any "mix" of the above: e.g.,

$$x^* = \operatorname*{argmax}_{x \in \mathcal{X}} p(x) = \operatorname*{argmax}_{x \in \mathcal{X}} \int_{\mathcal{Z}} dz p(z) p(x|z)$$

Divide and Conquer

Probabilistic modelling provides a methodology that puts a clear division between

- What to solve : Model Construction
  - Both an Art and Science
  - Highly domain specific
- How to solve : Inference Algorithm
  - Mechanical (In theory! not in practice)
  - Generic

Probability Theory

- Axiomatic development by Kolmogorov during 30'.
- Modern rigorous treatment as a branch of measure theory.
- A huge spectrum of theoretical and practical applications.
- "Probabilist" versus "Statistician"

The meaning of probability

- **Frequentist view**: Frequencies of outcomes in random experiments,
  - restrict probabilities to refer only to frequencies of outcomes in repeatable random experiments
- **Bayesian view**: Describe degrees of belief
  - Use probabilities to describe inferences.
  - Tomorrow, it will rain with probability 0.95.
- The **Frequentist versus Bayesian debate**,
  - Similar questions but require different emphasis in their answer.
    - Is this drug useful for that disease?
    - Is this webpage relevant for that query?
    - Is there a cow in this image?
    - What is the tempo of this piece of music?

Bayesian interpretation: Degrees of Belief

- **Subjective** interpretation of probability
- Using Bayes rule does not make one a Bayesian, using it always does.
- Cox' axioms
  – Degrees of belief *can* be mapped onto probabilities if they satisfy simple consistency rules.
- The rules of probability ensure **consistency**. Same assumptions and same data will lead to identical conclusions.
- Objective (good) versus Subjective (bad) ?
  – It is not possible to do inference without making assumptions
  – Deductive versus Inductive Reasoning

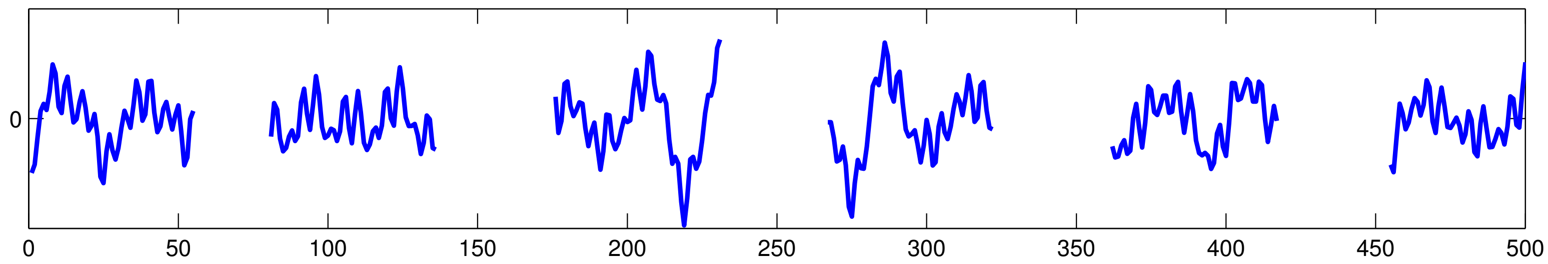
Deductive versus Inductive Reasoning

- Prove that no three positive integers a, b, and c can satisfy the equation

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

  for any integer $n > 2$.

- Infer missing samples given observed ones



Unappropriate Inductive Reasoning

Example from Borovik

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

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

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

Discrete Probability Tables, Univariate

- X : The random variable
- $\mathcal{X} = \{\xi_1, \xi_2, \ldots, \xi_N\}$: Sample space, Domain
- N : Cardinality
- $\pi_i = \Pr\{X = \xi_i\}$: Probabilities
  - $\sum_i \pi_i = \pi_1 + \pi_2 + \cdots + \pi_N = 1$
  - $\pi_i \geq 0$

$p(X)$	
$X = \xi_1$	π_1
$X = \xi_2$	π_2
$X = \xi_3$	π_3
$\vdots$	$\vdots$
$X = \xi_N$	π_N

Discrete Probability Models, Examples

- $\mathcal{X} = \{\text{female}, \text{male}\}$, Gender
- $\mathcal{X} = \{\text{A}, \text{B}, \dots, \text{Z}\}$, First letter of the surname
- $\mathcal{X} = \{1, \dots, e, \dots, N\}$, Height category
- $\mathcal{X} = \{1, \dots, e, \dots, M\}$, Weight category
- Selecting these categories is known as 'feature engineering'

Discrete Probability Tables, Bivariate

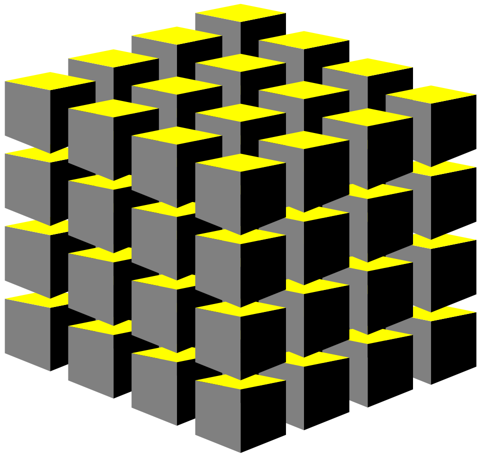
- X, Y : The random variables
- $X \in \mathcal{X} = \{\xi_1, \xi_2, \dots, \xi_{N_x}\}$, $Y \in \mathcal{Y} = \{\eta_1, \eta_2, \dots, \eta_{N_y}\}$
- N_x, N_y : Cardinalities
- $\pi_{i,j} = \Pr\{X = \xi_i, Y = \eta_j\}$: Probabilities
  - $\sum_{i,j} \pi_{i,j} = 1$, $\pi_{i,j} \geq 0$

$p(x,y)$	$y = \eta_1$	$y = \eta_2$	$\dots$	$y = \eta_{N_y}$
$x = \xi_1$	$\pi_{1,1}$	$\pi_{1,2}$	$\dots$	π_{1,N_y}
$x = \xi_2$	$\pi_{2,1}$	$\pi_{2,2}$	$\dots$	π_{2,N_y}
$x = \xi_3$	$\pi_{3,1}$	$\pi_{3,2}$	$\dots$	π_{3,N_y}
$\vdots$	$\vdots$		$\dots$	
$x = \xi_{N_x}$	$\pi_{N_x,1}$	$\pi_{N_x,2}$	$\dots$	π_{N_x,N_y}

Probability Tables

- Joint distribution: A N-dimensional array $p(x_1, x_2, \ldots, x_N)$ where each cell is positive and $\sum_{\mathbf{x}} p(\mathbf{x}) = 1$

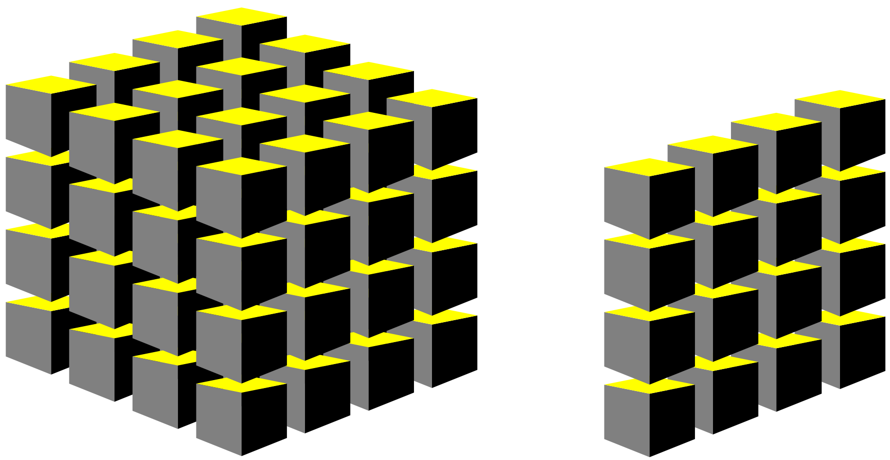
Example: $p(x_1, x_2, x_3)$ with $N_i = 4$



Each cell is a positive number s.t. $\sum_{x_1, x_2, x_3} p(x_1, x_2, x_3) = 1$

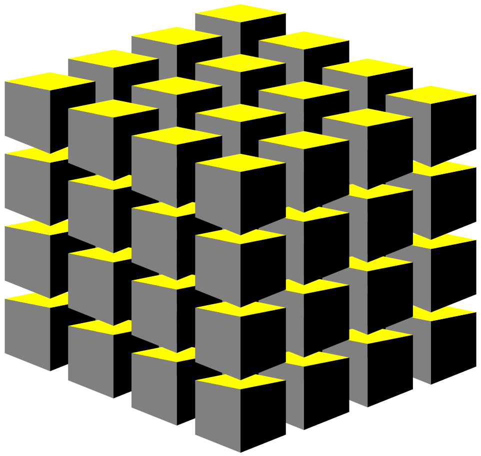
Marginalization == Summing over subsets of variables

$$p(A) = \sum_B p(A, B)$$

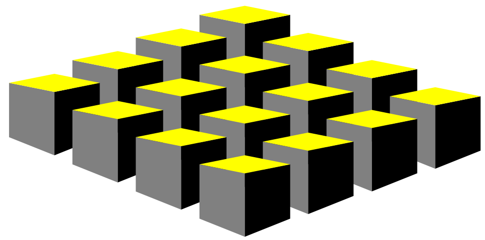


$$\sum_{x_1} p(x_1, x_2, x_3) \quad = \quad p(x_2, x_3)$$

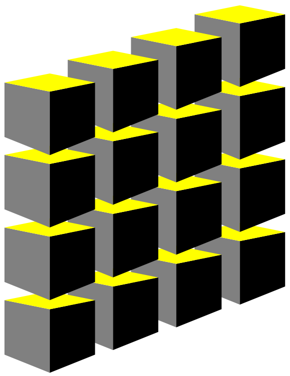
Clamping



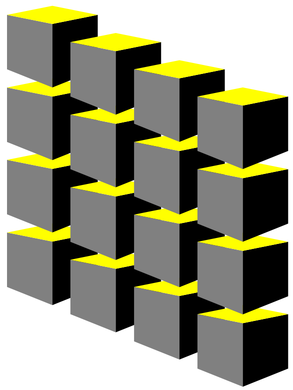
$p(x_1, x_2, x_3)$



$p(x_1, x_2, x_3 = \hat{x}_3)$



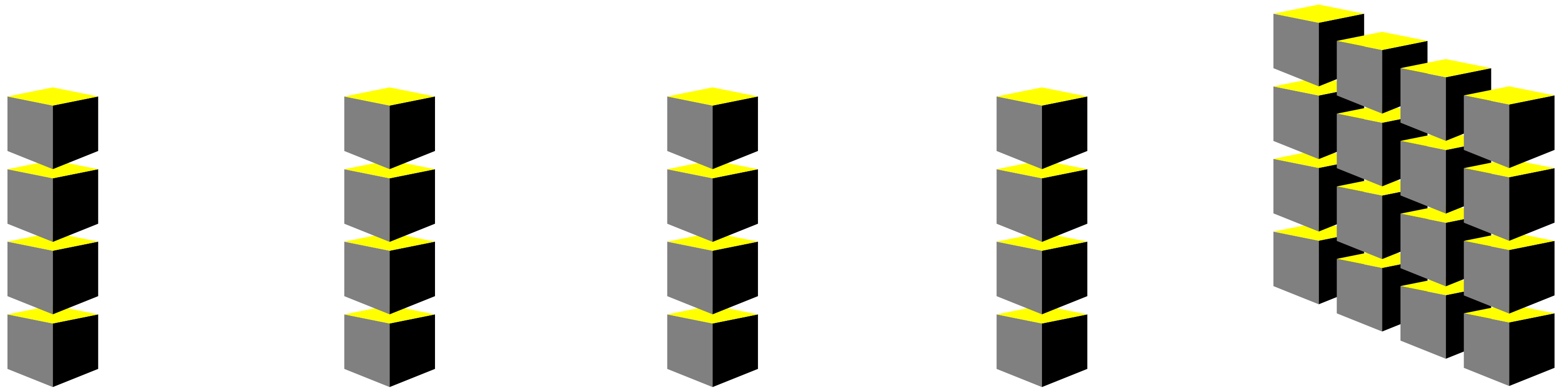
$p(x_1 = \hat{x}_1, x_2, x_3)$



$p(x_1, x_2 = \hat{x}_2, x_3)$

Conditional Probability

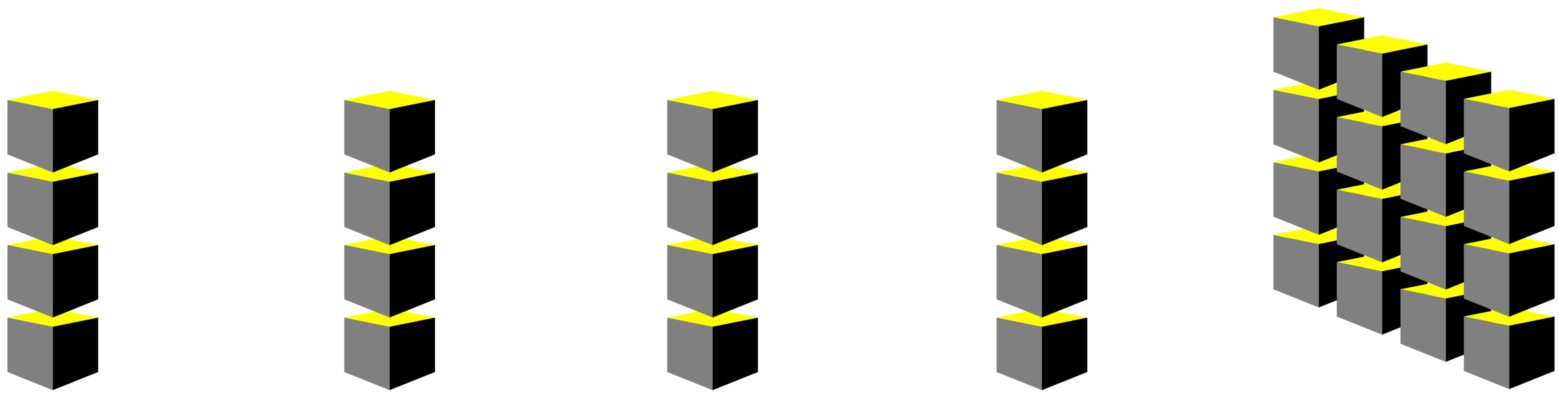
- A **collection** of probability distributions denoted as $p(A|B)$. For each configuration of variables in B we have a probability distribution on variables in A



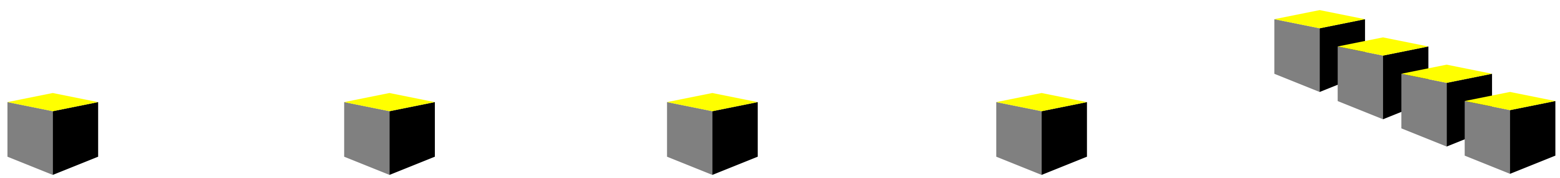
$$\{p(x_3|x_1 = 1), p(x_3|x_1 = 2), p(x_3|x_1 = 3), p(x_3|x_1 = 4)\} = \quad p(x_3|x_1)$$

Conditional Probability (cont)

- We can represent a joint probability distribution as $p(A, B) = p(A|B)p(B)$.

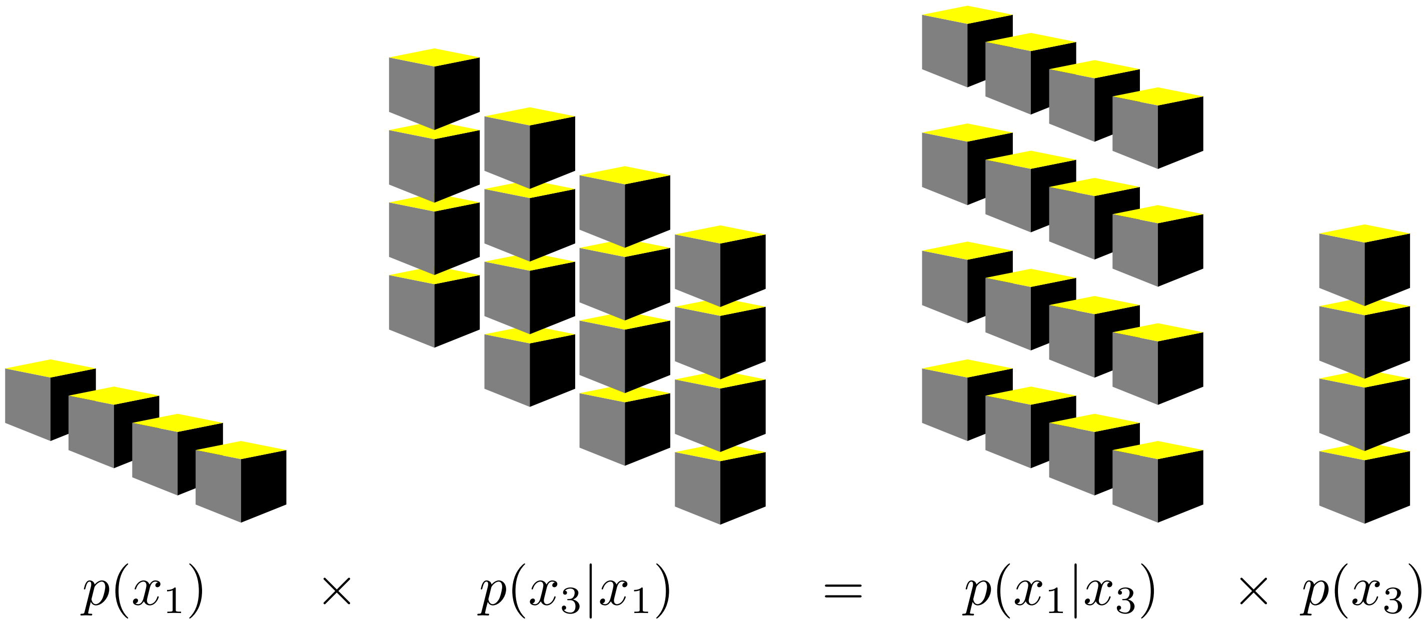


$$
\begin{array}{ccccccc}
\{p(x_3|x_1 = 1), & p(x_3|x_1 = 2), & p(x_3|x_1 = 3), & p(x_3|x_1 = 4)\} & = & p(x_3|x_1) \\
\times & \times & \times & \times & = & \times \\
\{p(x_1 = 1), & p(x_1 = 2), & p(x_1 = 3), & p(x_1 = 4)\} & = & p(x_1)
\end{array}
$$



Properties of Conditional Probabilities

- $p(A, B) = p(B|A)p(A) = p(A|B)p(B)$.



- $p(A) = \sum_B p(A|B)p(B)$

Bayes Theorem Repeated

$$p(B|A) = \frac{p(A|B) \times p(B)}{\sum_B p(A|B)p(B)}$$

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}}$$

- Think of A as an observation and B as its hidden cause.
- Bayes theorem says how to update our prior belief $p(B)$ given a new observation A. This gives a way of "reversing" the conditional probability $p(A|B)$.

Bayes Theorem Repeated

- This rather simple looking formula has surprisingly many applications
  - Medical Diagnosis (Symptoms/Diseases)
  - Speech Recognition (Signal/Phoneme)
  - Music Transcription (Audio/Score)
  - Computer Vision (Image/Object)
  - Robotics (Sensor/Position)
  - Finance (Past Price/Future Price)

- A natural way of combining prior knowledge with data $\Rightarrow$ Learning

Exercise

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

1. Find the following quantities

   - Marginals: $p(x_1)$, $p(x_2)$
   - Conditionals: $p(x_1|x_2)$, $p(x_2|x_1)$
   - Posterior: $p(x_1, x_2 = 2)$, $p(x_1|x_2 = 2)$
   - Evidence: $p(x_2 = 2)$
   - $p(\{\})$
   - Max: $p(x_1^*) = \max_{x_1} p(x_1|x_2 = 1)$
   - Mode: $x_1^* = \arg\max_{x_1} p(x_1|x_2 = 1)$
   - Max-marginal: $\max_{x_1} p(x_1, x_2)$

2. Are x_1 and x_2 independent ? (i.e., Is $p(x_1, x_2) = p(x_1)p(x_2)$?)

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

- 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

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

- Evidence:

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

- Normalisation constant:

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

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

Inference and Learning

- Maximum Likelihood,
- Penalised Likelihood,
- Bayesian Learning

Maximum Likelihood

- 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

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

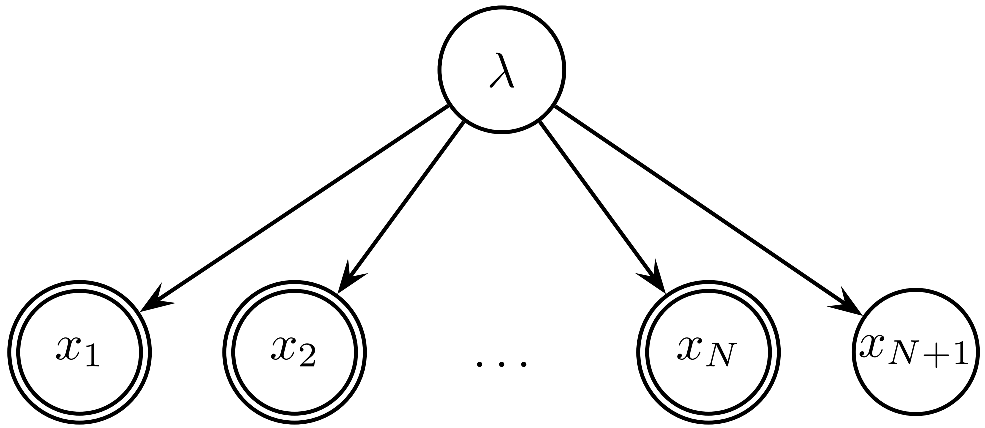
Bayesian Learning

- Treats parameters on the same footing as all other variables
- Integrate over unknown parameters rather than using point estimates
  - ’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 ...

Bayesian Learning, $\lambda = p(x = \textbf{Tail})$

?

Bayesian Learning

T, ?

Bayesian Learning

T, T, ?

Bayesian Learning

T, T, T, ?

Bayesian Learning

T, T, T, T, ?

Bayesian Learning

T, T, T, T, T, ?

Bayesian Learning

T, T, T, T, T, Y, ?

Bayesian Learning

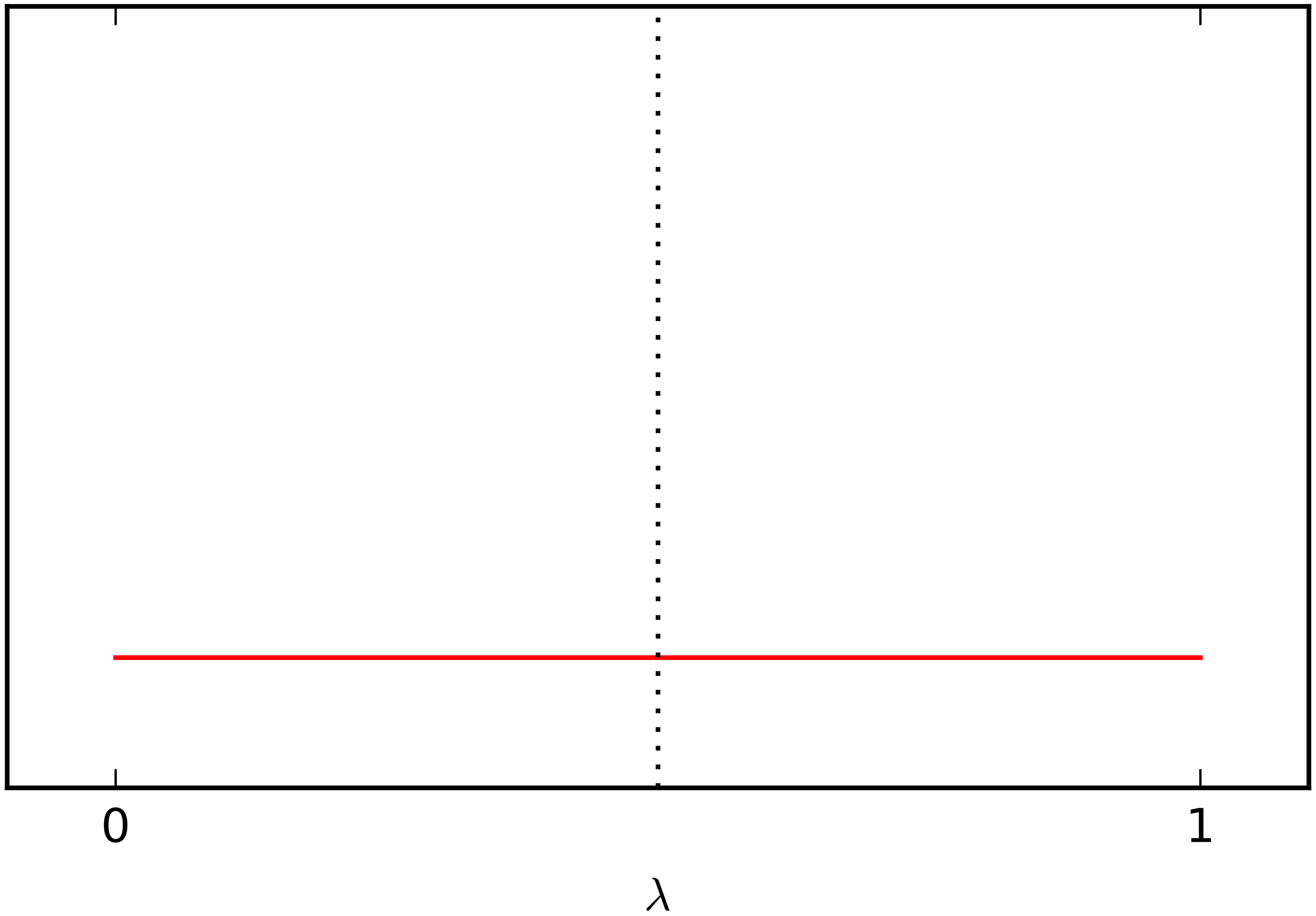
T, T, T, T, T, Y, T, ?

Bayesian Learning

T, T, T, T, T, Y, T, T, ?

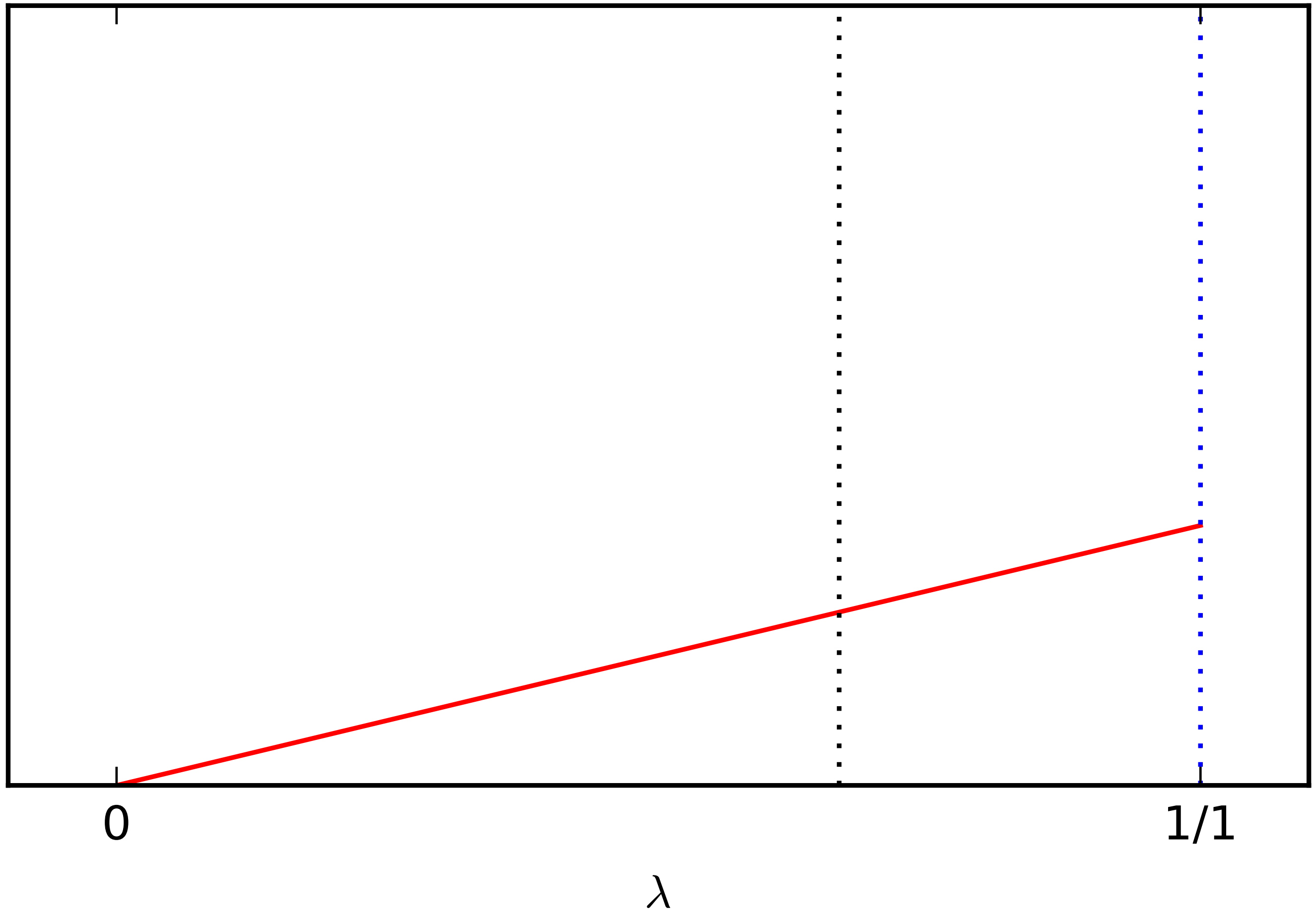
$p(\lambda)$

?



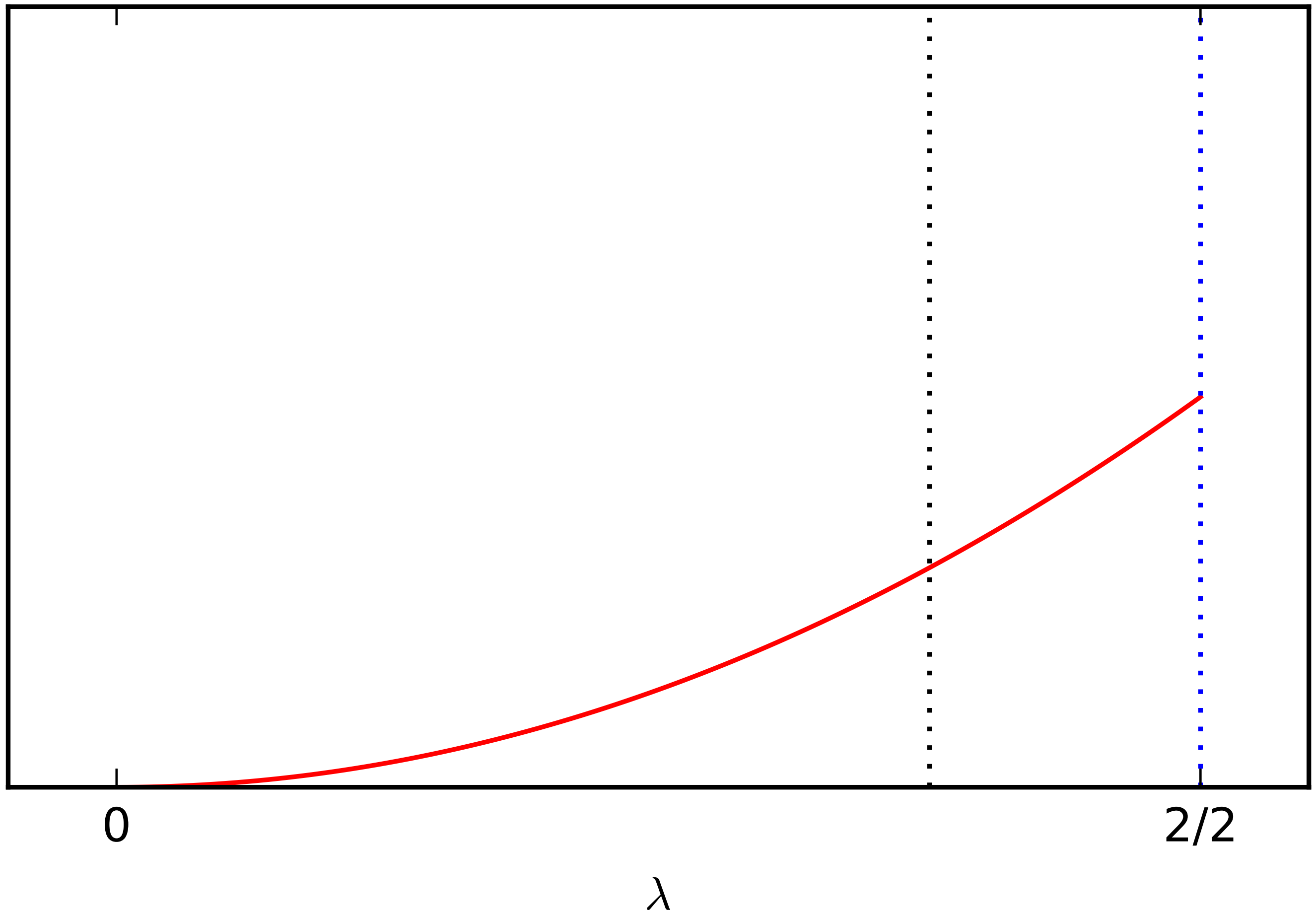
$$p(\lambda|x_1)$$

T, ?



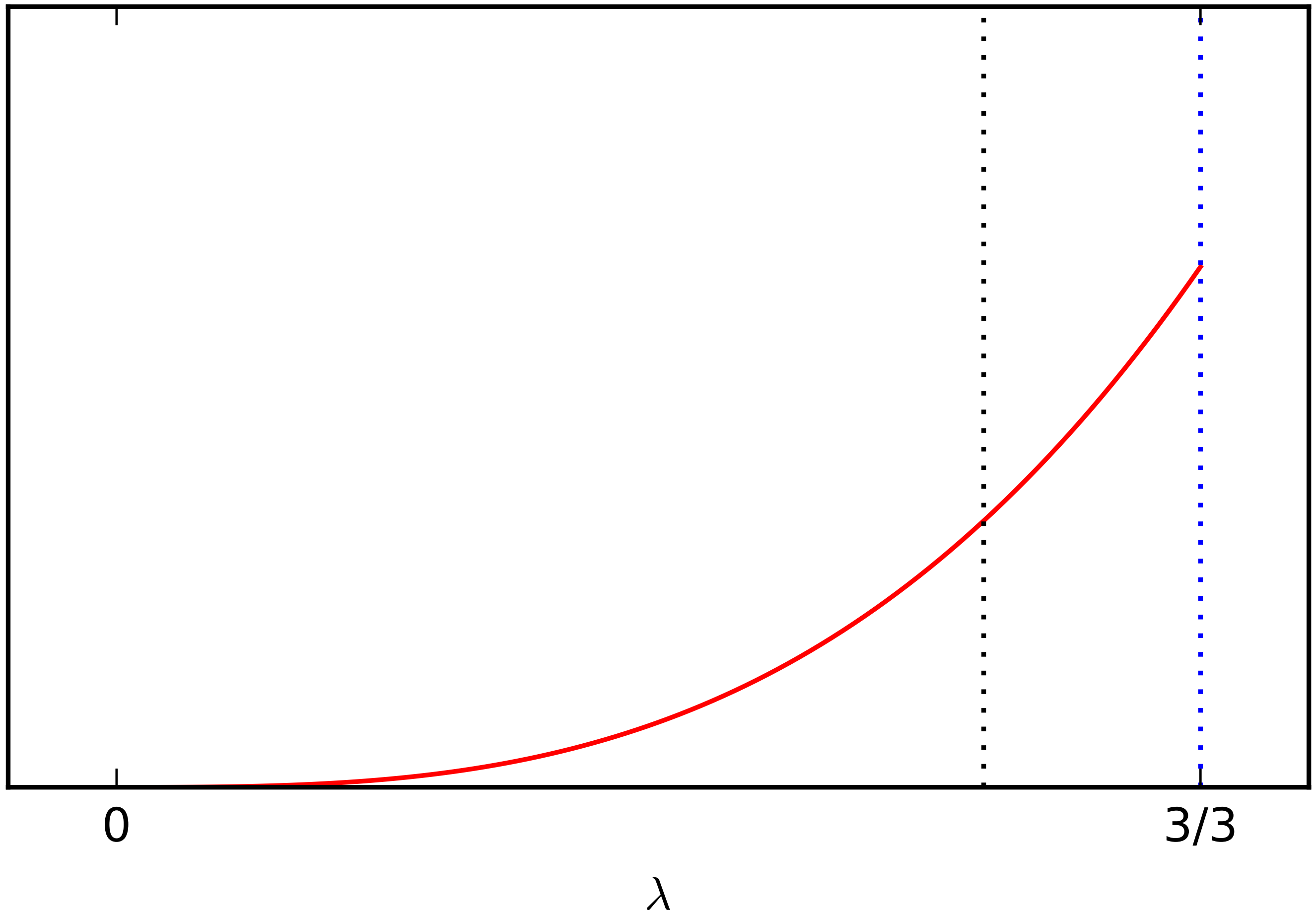
$$p(\lambda|x_1, x_2)$$

T, T, ?



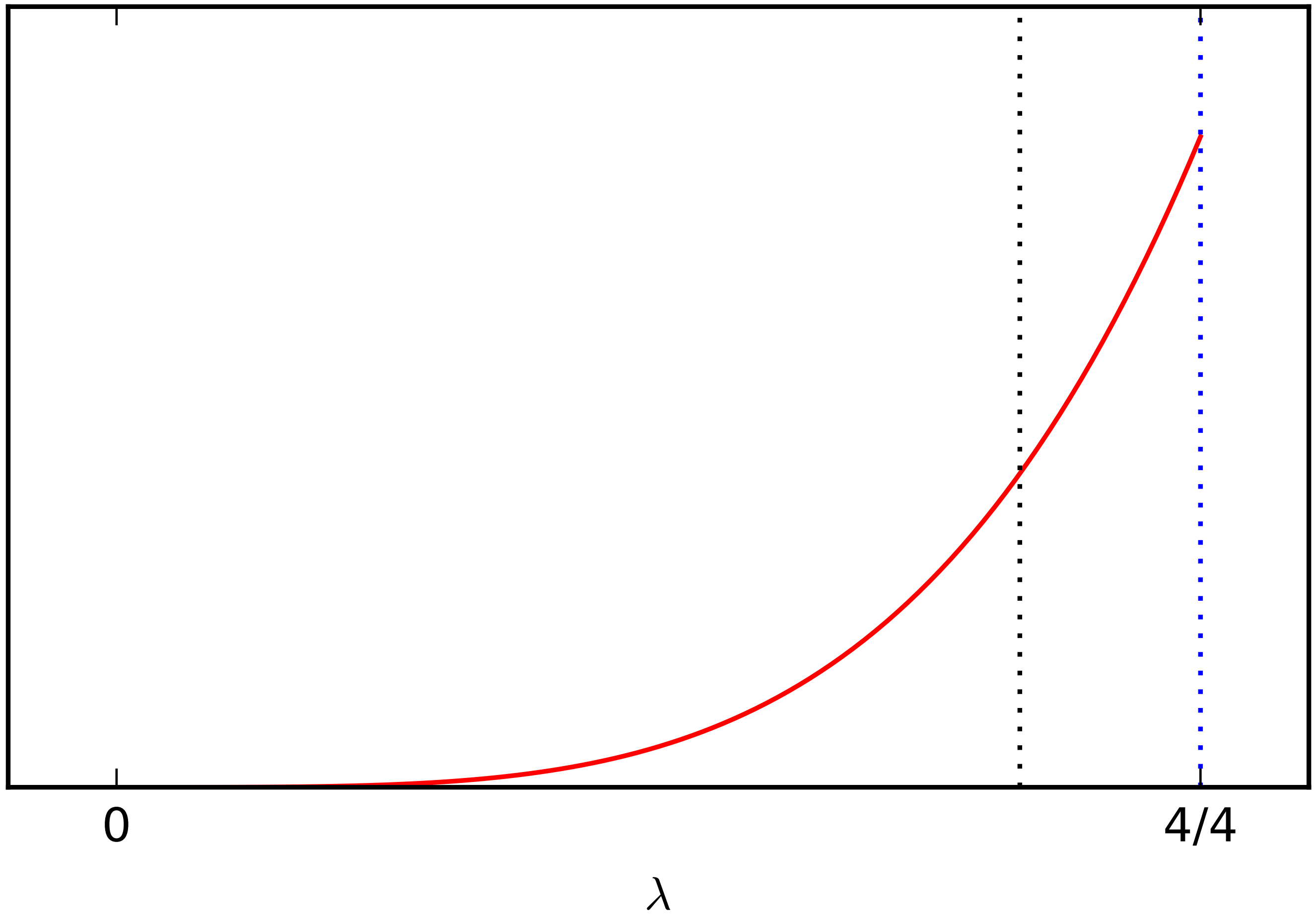
$$p(\lambda|x_{1:3})$$

T, T, T, ?



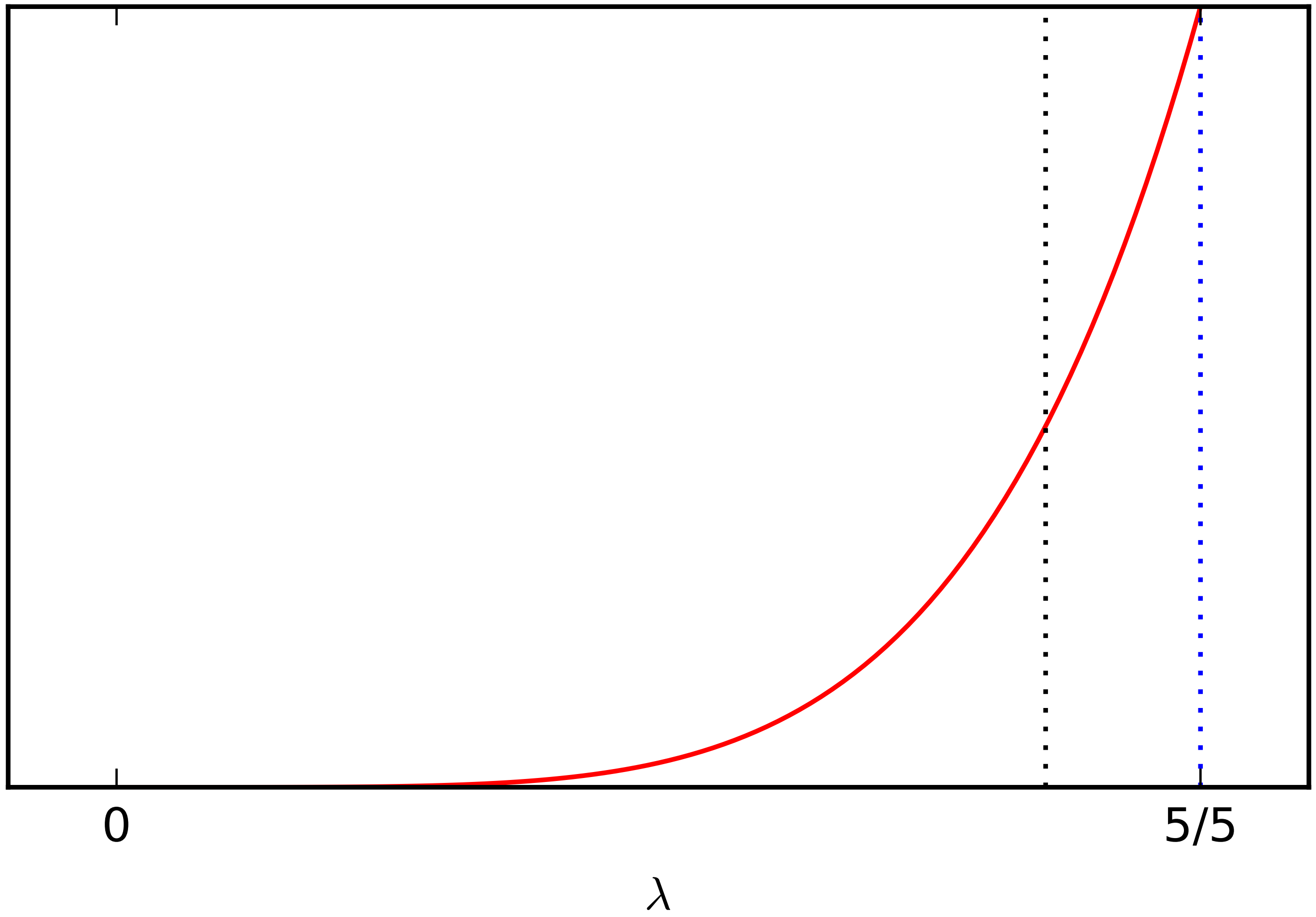
$$p(\lambda|x_{1:4})$$

T, T, T, T, ?



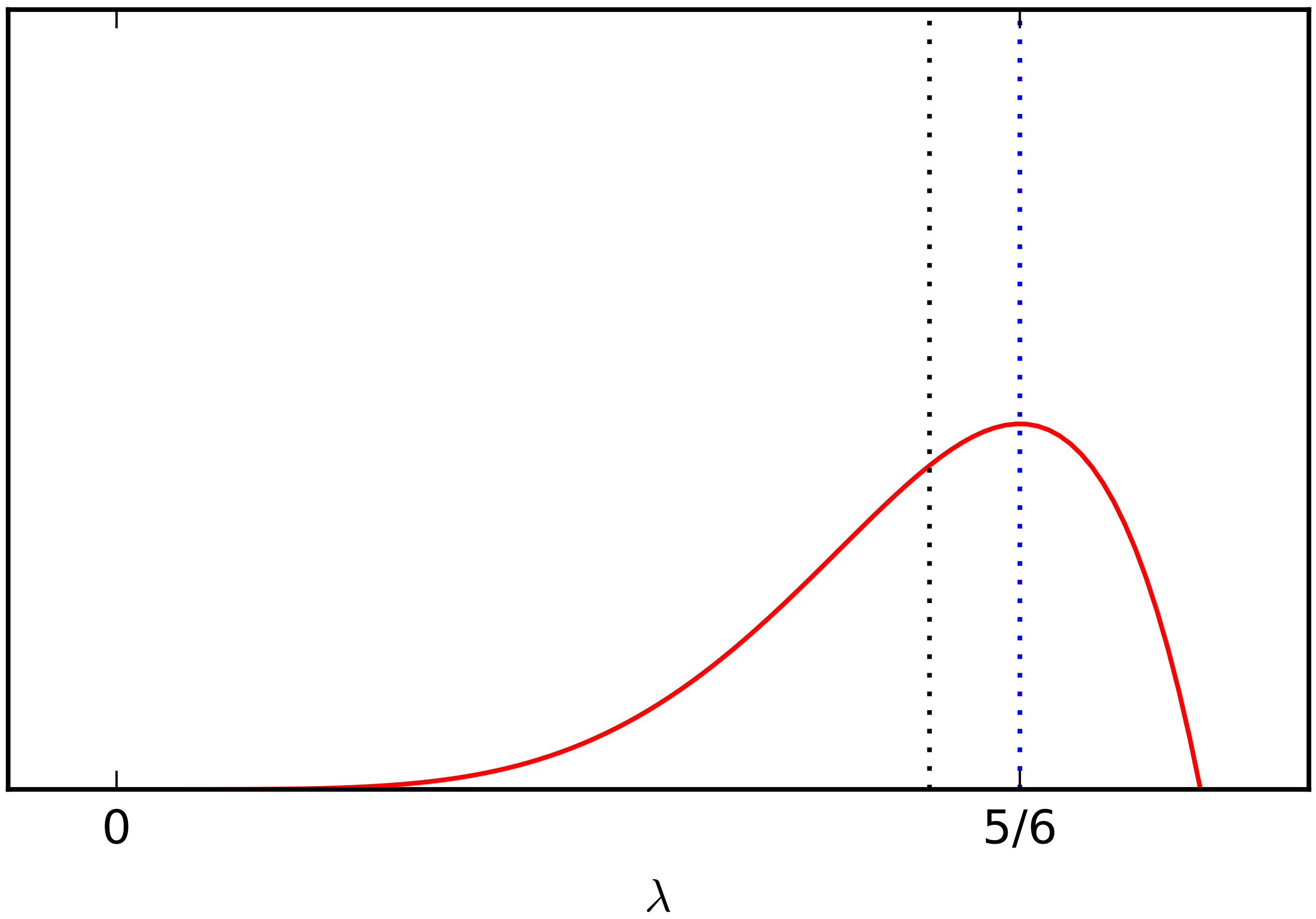
$$p(\lambda|x_{1:5})$$

T, T, T, T, T, ?



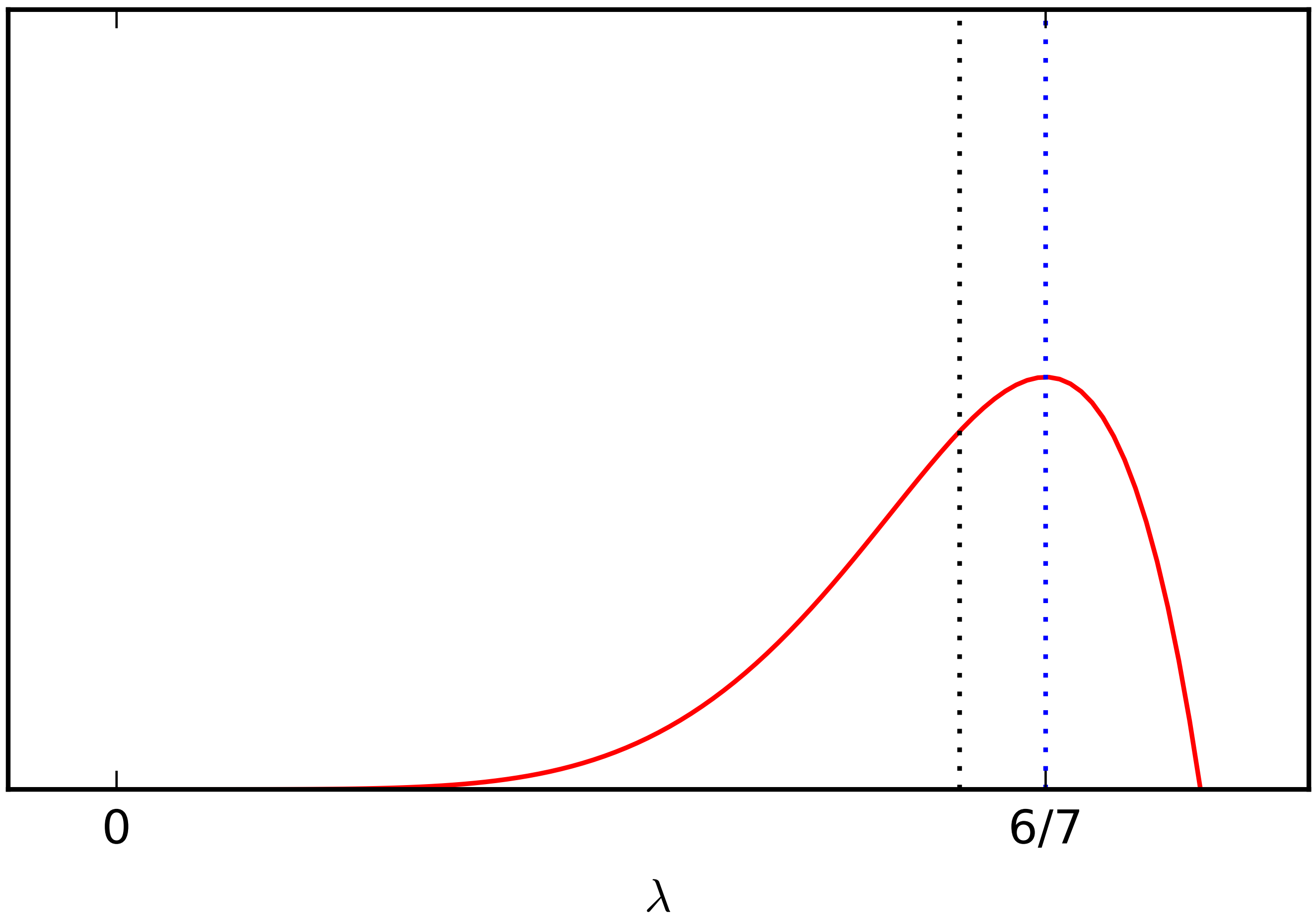
$$p(\lambda | x_{1:6})$$

T, T, T, T, T, Y, ?



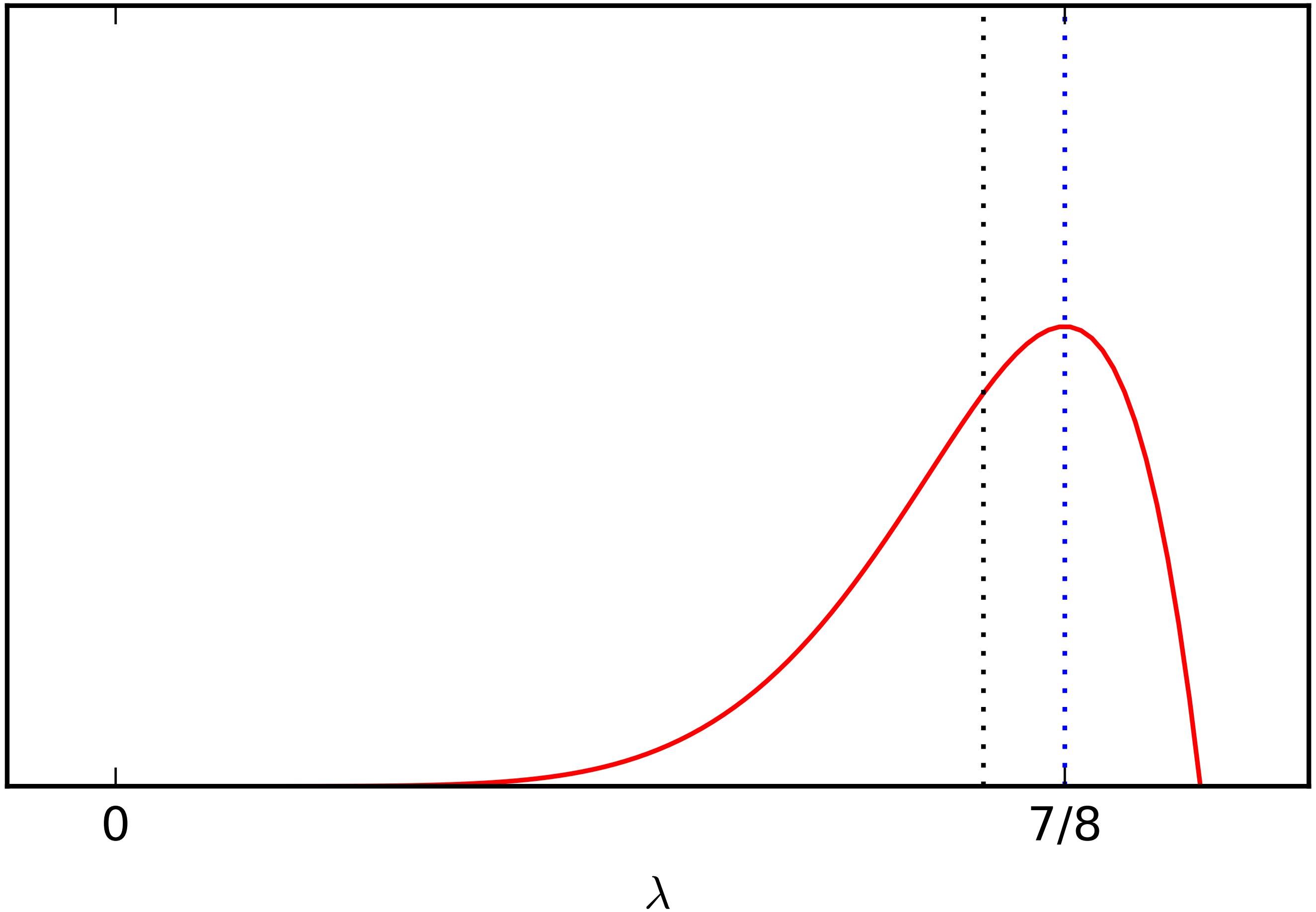
$$p(\lambda|x_{1:7})$$

T, T, T, T, T, Y, T, ?



$$p(\lambda|x_{1:8})$$

T, T, T, T, T, Y, T, T, ?



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

Maximum Entropy Principle

What is the least informative distribution that has the given expectations?

$$H[p] = -\int_{\mathcal{X}} p(x) \log(p(x)) dx$$

maximize $H[p]$

subject to

$$\int_{\mathcal{X}} p(x) dx = 1 \qquad \text{Normalizasyon}$$

$$\int_{\mathcal{X}} \psi(x) p(x) dx = s \qquad \text{Moment Eşleme}$$

Lagrange Functional

$$\Lambda(p; \lambda, \theta) = -\int_{\mathcal{X}} p(x) \log(p(x))dx + \lambda(1 - \int_{\mathcal{X}} p(x)dx) + \theta(s - \int_{\mathcal{X}} \psi(x)p(x)dx)$$

$$\begin{aligned} \frac{\delta}{\delta p}\Lambda[p, \lambda, \theta] &= -\log(p(x)) - 1 + \lambda + \theta\phi(x) = 0 \\ p(x) &= \exp(\theta\psi(x)) \exp(\lambda - 1) \end{aligned}$$

Normalization constraint

$$\begin{aligned} \int_{\mathcal{X}} p(x)dx &= 1 = \exp(\lambda - 1) \int_{\mathcal{X}} \exp(\theta\psi(x))dx \\ \exp(\lambda - 1) &= \frac{1}{\int \exp(\theta\psi(x))dx} \end{aligned}$$

get rid of λ

$$A(\theta) \equiv \log \int_{\mathcal{X}} \exp(\theta\psi(x))dx$$

Solution: The exponential family (Gibbs distribution)

$$p(x) = \exp(\theta\psi(x) - A(\theta)) \quad (1)$$

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

Bernoulli $c = \{0, 1\}$ with success probability w

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

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

Is Bernoulli a exponential family?

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

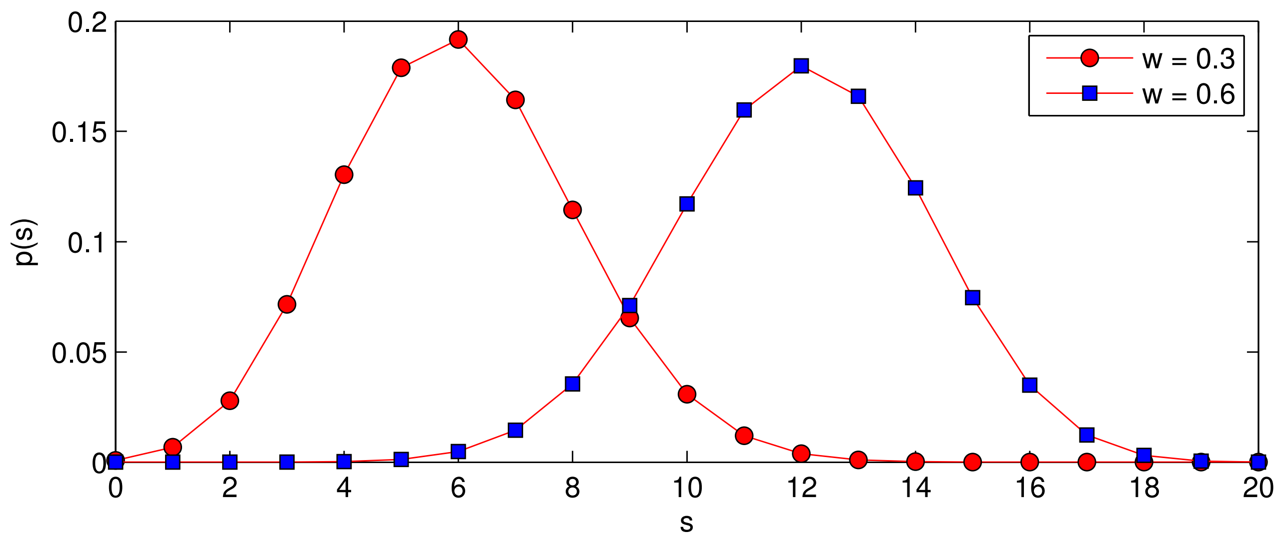
$$p(c|\theta) \quad = \quad h(c)\exp\{\theta^\top \psi(c) - A(\theta)\}$$

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

Binomial Distribution. $\mathcal{BI}(s; N, w)$

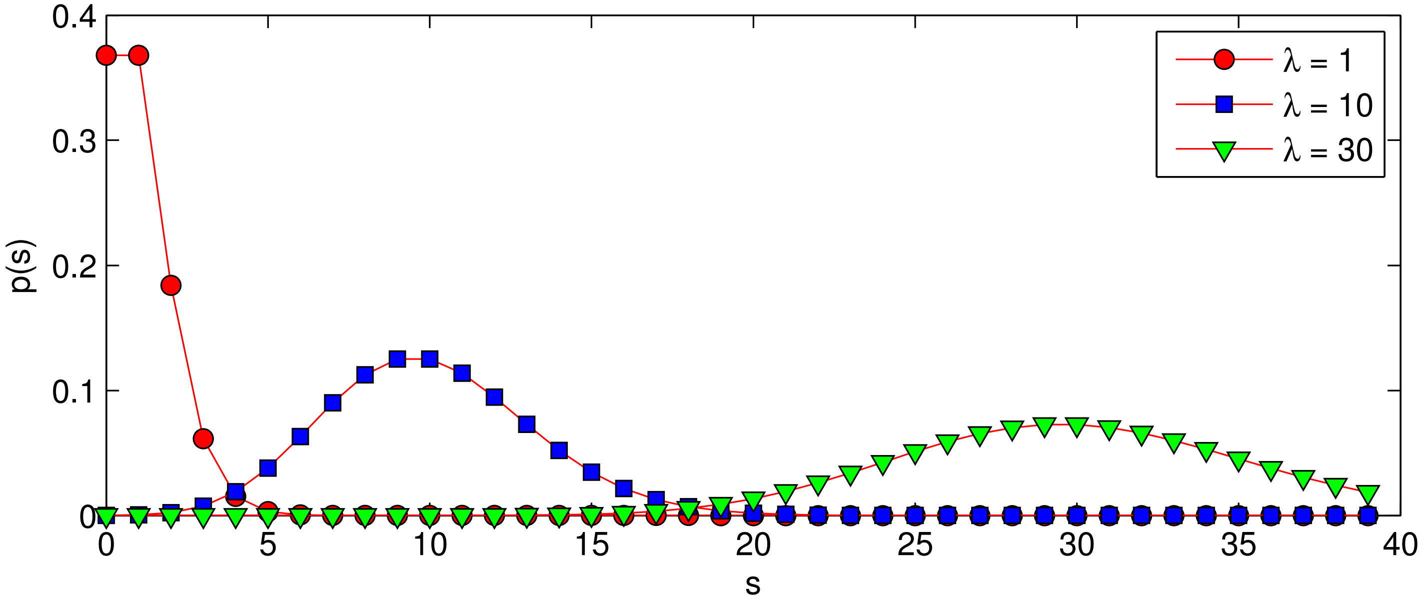
s is the number of successful outcomes in N independent Bernoulli trials with success probability w

$$
\begin{aligned}
\mathcal{BI}(s; N, w) &= \binom{N}{s} w^s (1-w)^{N-s} \\
&= \frac{N!}{s!(N-s)!} \exp(s \log w + (N-s)\log(1-w))
\end{aligned}
$$



Poisson Distribution. $\mathcal{PO}(s;\lambda)$

$$\mathcal{PO}(s;\lambda) = \frac{e^{-\lambda}}{s!}\lambda^s = \exp(s\log\lambda - \lambda - \log(s!))$$



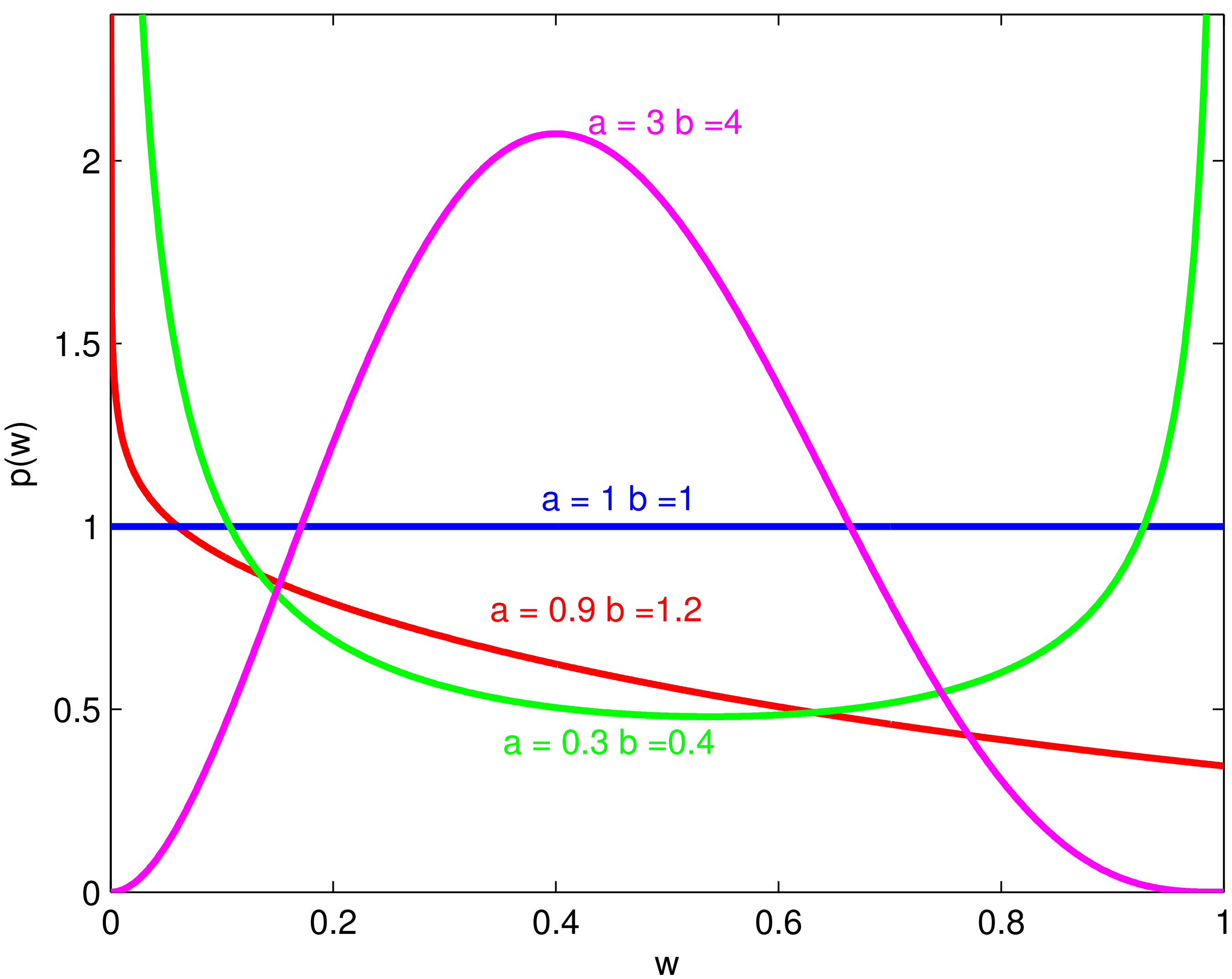
Beta. $\mathcal{B}(w; a, b)$

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

Mean :

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

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



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

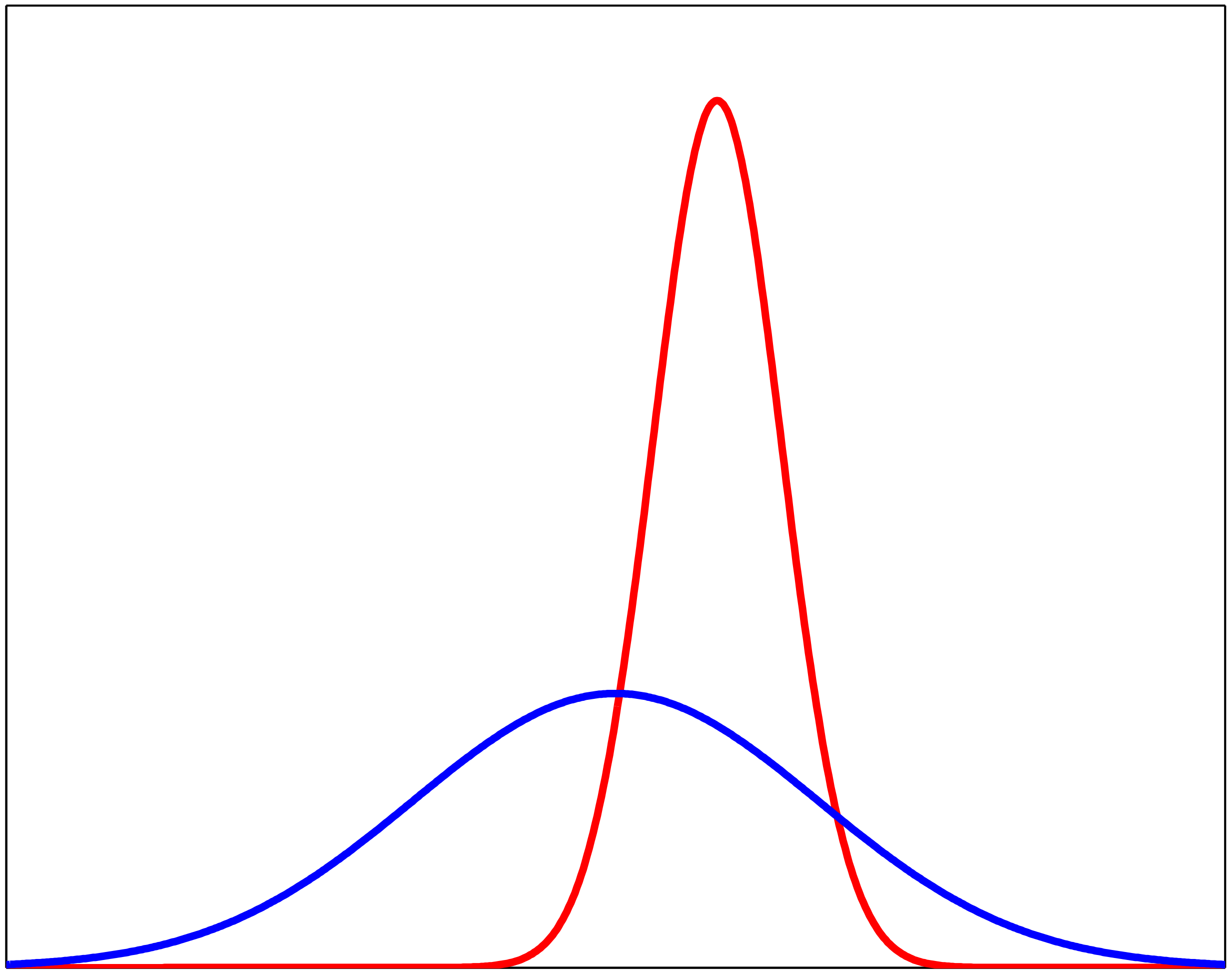
Gauss mean m and variance S

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

Coefficient matching

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

Gaussian.



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

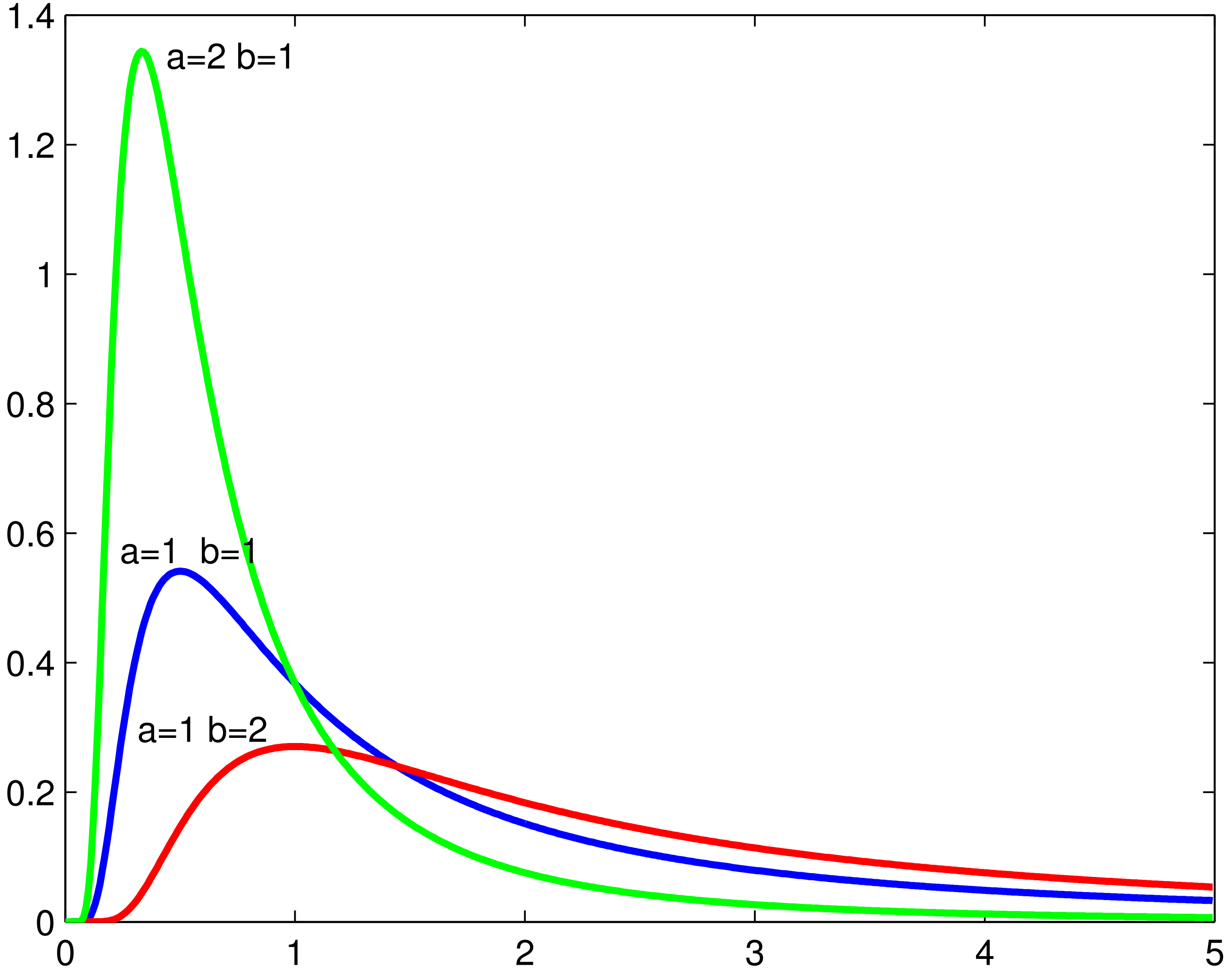
The inverse Gamma distribution with shape a and scale b

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

Match coefficients

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

Inverse Gamma



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

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

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

Hence by matching coefficients, we have

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

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.

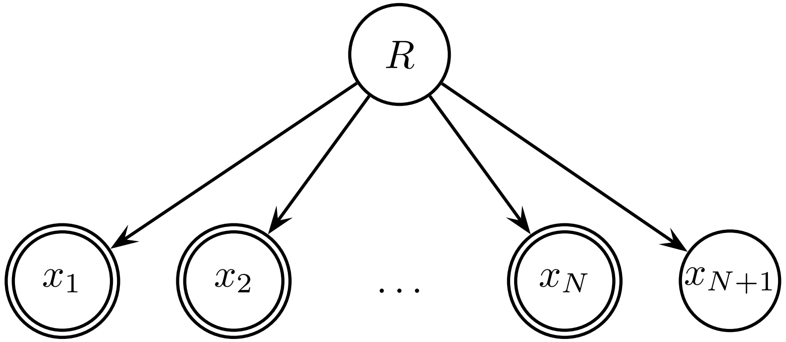
$$
\begin{aligned}
p(x|R) &= \mathcal{N}(x; 0, R) \\
p(R) &= \mathcal{IG}(R; a, b)
\end{aligned}
$$

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

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

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

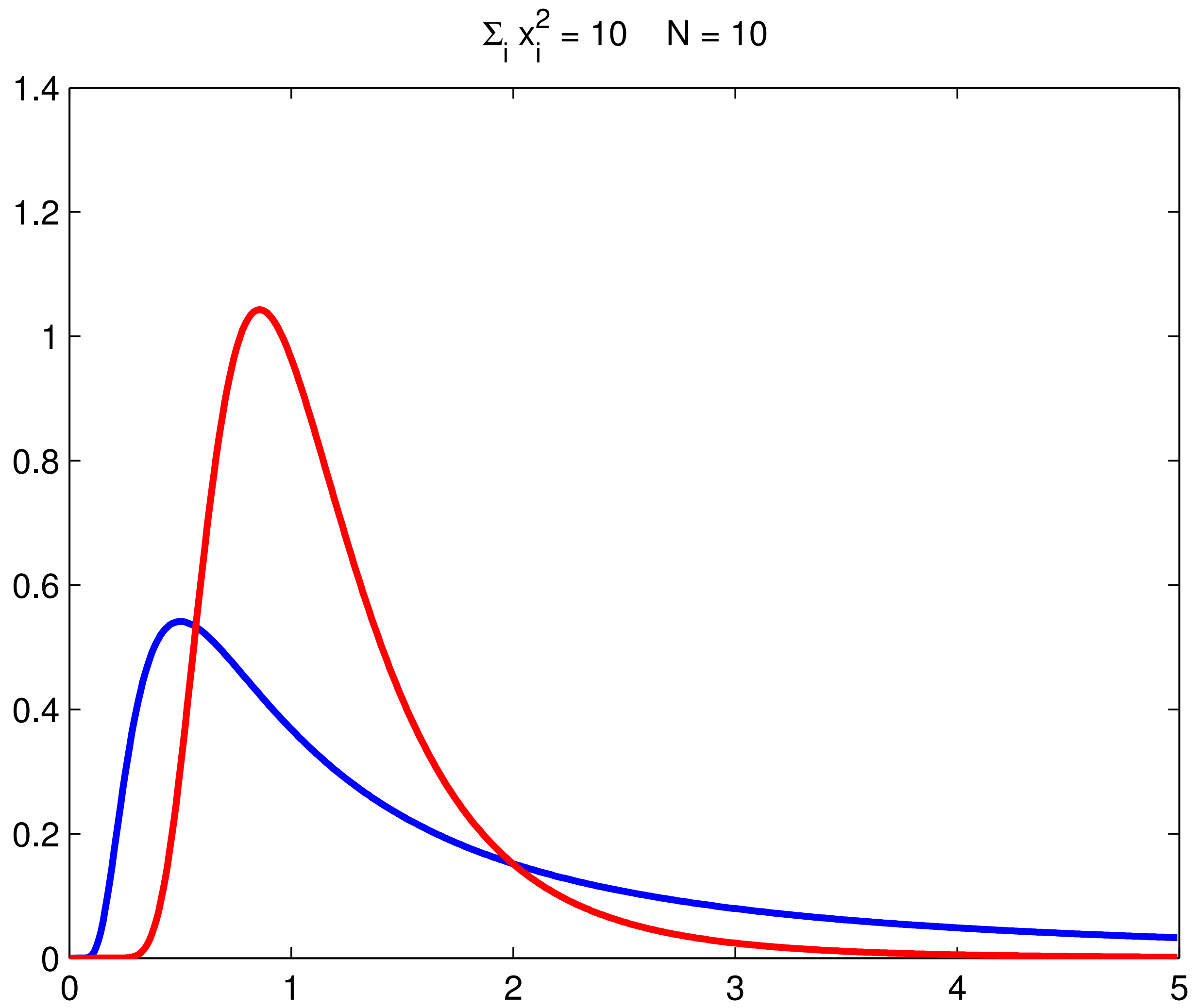
Example: posterior inference of variance R from $x_1, \dots, x_N$.



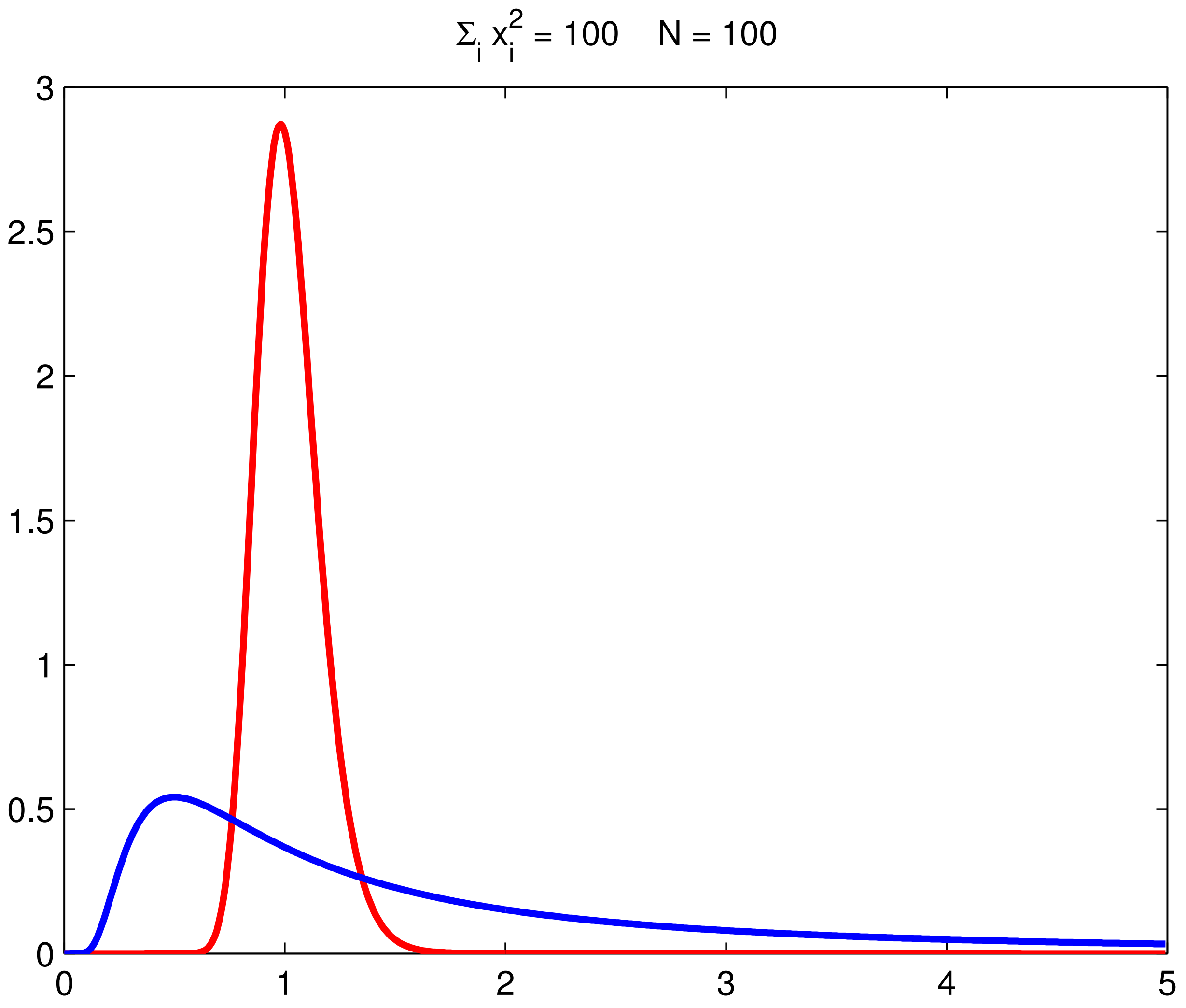
$$
\begin{aligned}
p(R|x) &\propto p(R)\prod_{i=1}^{N} p(x_i|R) \\
&\propto \exp\left(-(a+1)\log R - b\frac{1}{R}\right)\exp\left(-\left(\frac{1}{2}\sum_i x_i^2\right)\frac{1}{R} - \frac{N}{2}\log R\right) \\
&= \exp\left(\left(\begin{array}{c} -(a+1+\frac{N}{2}) \\ -(b+\frac{1}{2}\sum_i x_i^2) \end{array}\right)^\top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

Sufficient statistics are **additive**

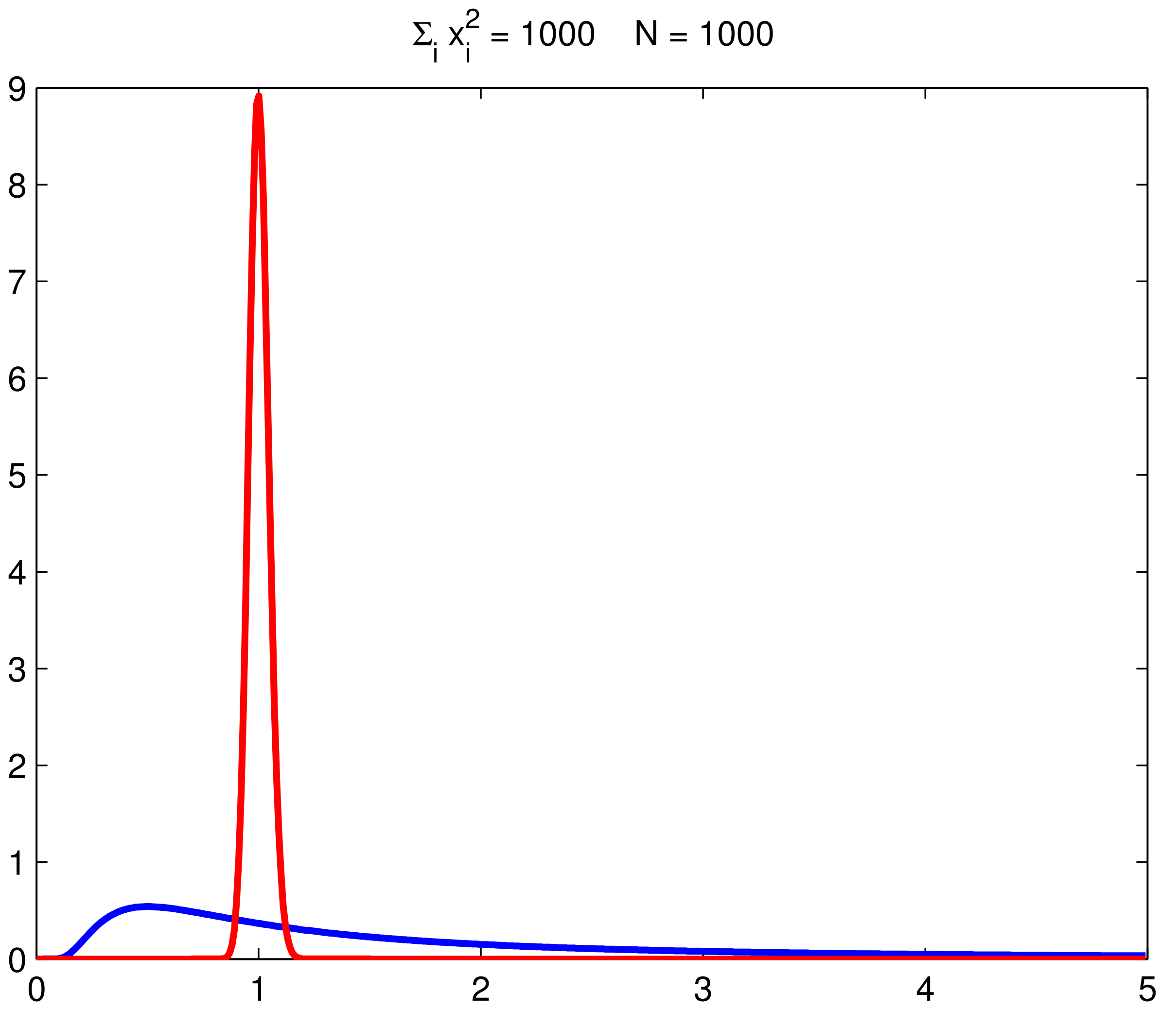
Inverse Gamma, $\sum_i x_i^2 = 10 \quad N = 10$



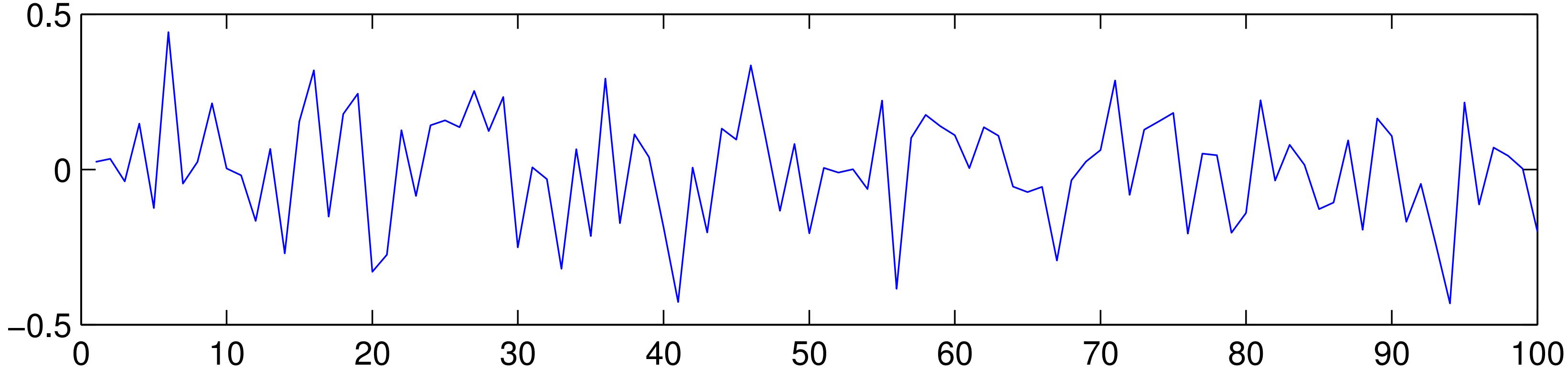
Inverse Gamma, $\sum_i x_i^2 = 100 \quad N = 100$



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



Example: AR(1) model



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

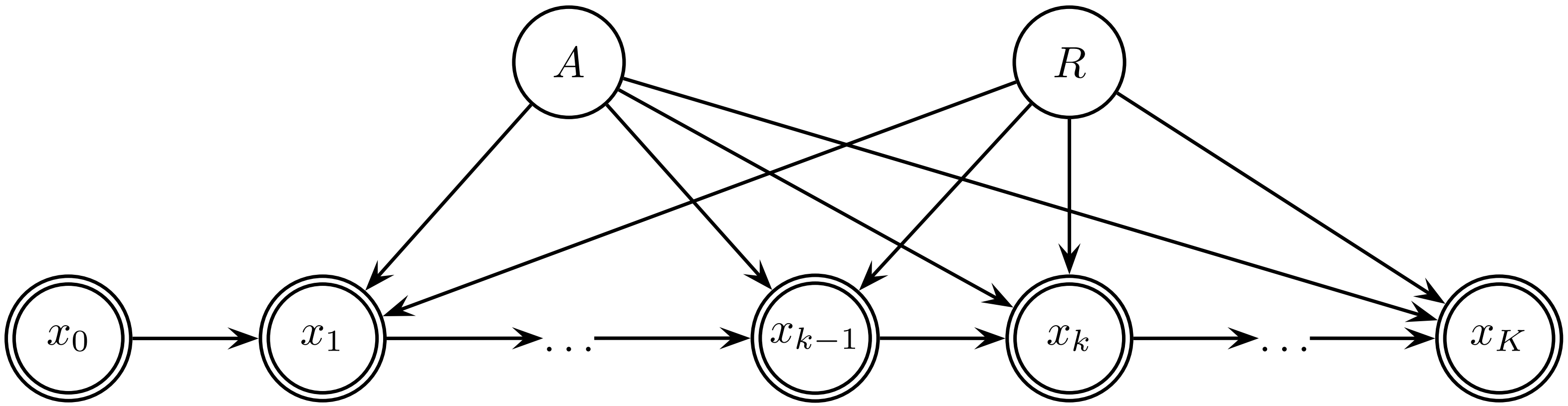
ϵ_k is i.i.d., zero mean and normal with variance R.

Estimation problem:

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

AR(1) model, Generative Model notation

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



Observed variables are shown with double circles

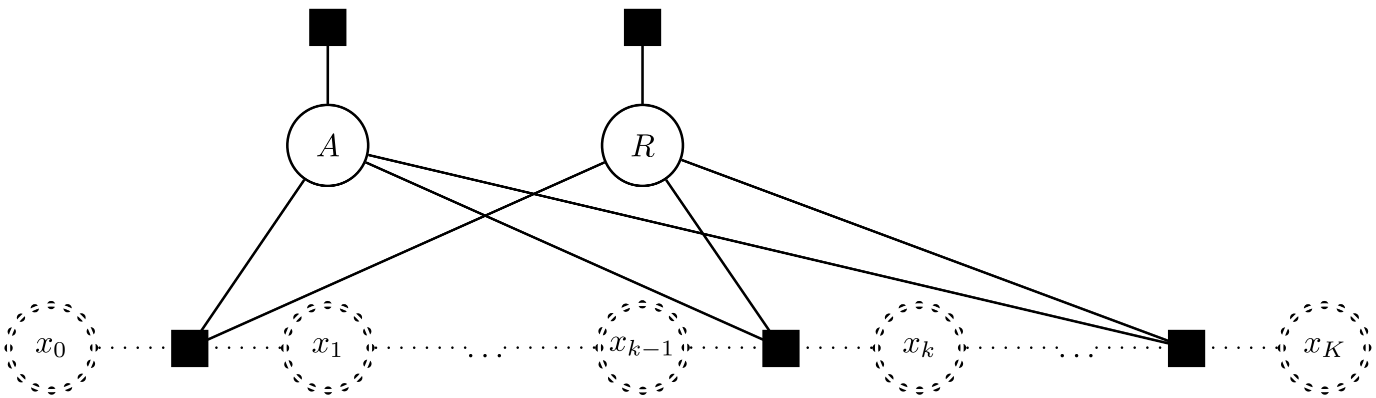
AR(1) Model. Bayesian Posterior Inference

$$p(A, R|x_0, x_1, \dots, x_K) \quad \propto \quad p(x_1, \dots, x_K|x_0, A, R)p(A, R)$$

$$\text{Posterior} \quad \propto \quad \text{Likelihood} \times \text{Prior}$$

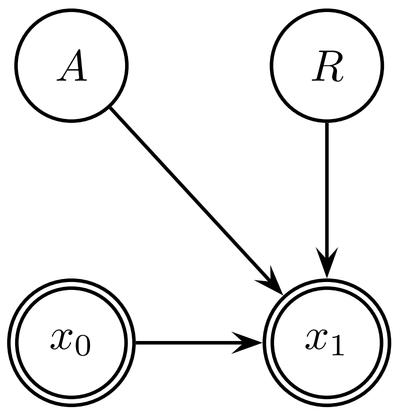
Using the Markovian (conditional independence) structure we have

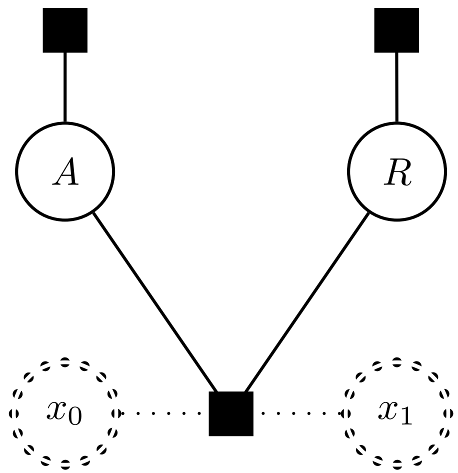
$$p(A, R|x_0, x_1, \dots, x_K) \quad \propto \quad \left(\prod_{k=1}^{K} p(x_k|x_{k-1}, A, R) \right) p(A)p(R)$$



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

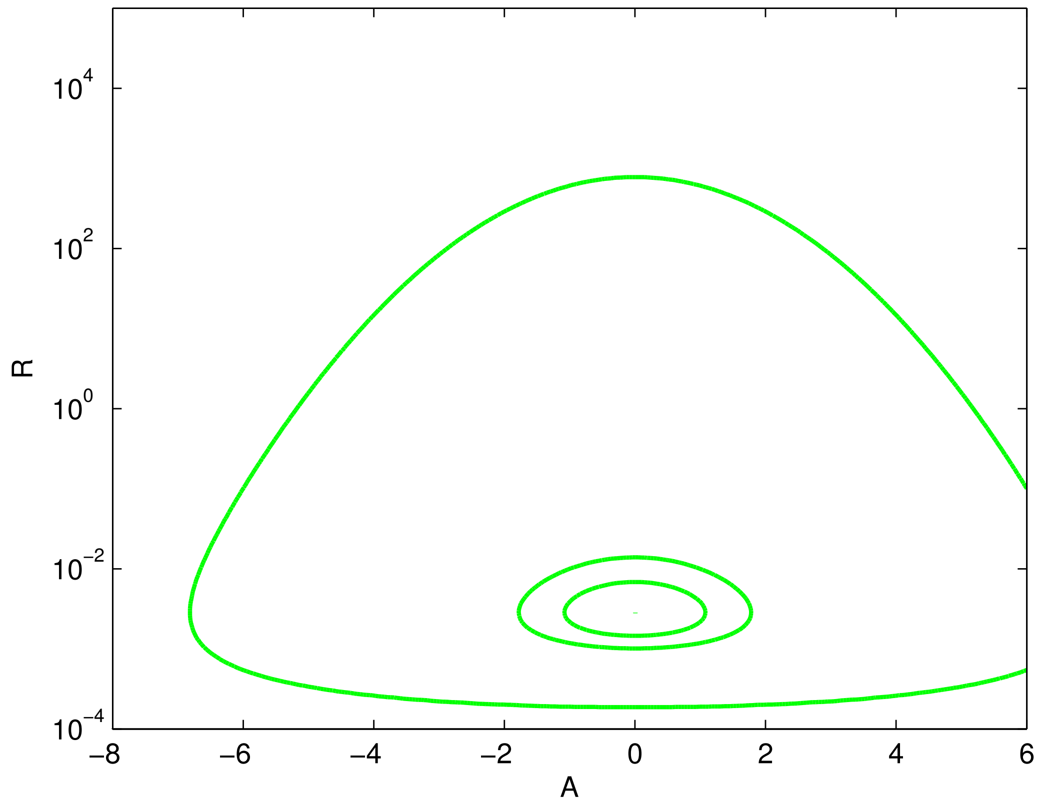
$$
\begin{aligned}
p(A, R|x_0, x_1) &\propto p(x_1|x_0, A, R)p(A)p(R) \\
&= \mathcal{N}(x_1; Ax_0, R)\mathcal{N}(A; 0, P)\mathcal{IG}(R; \nu, \beta/\nu) \\
&\propto \exp\left(-\frac{1}{2}\frac{x_1^2}{R} + x_0x_1\frac{A}{R} - \frac{1}{2}\frac{x_0^2A^2}{R} - \frac{1}{2}\log 2\pi R\right) \\
&\quad \exp\left(-\frac{1}{2}\frac{A^2}{P}\right)\exp\left(-(\nu+1)\log R - \frac{\nu}{\beta}\frac{1}{R}\right)
\end{aligned}
$$

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 $p(A, R)$

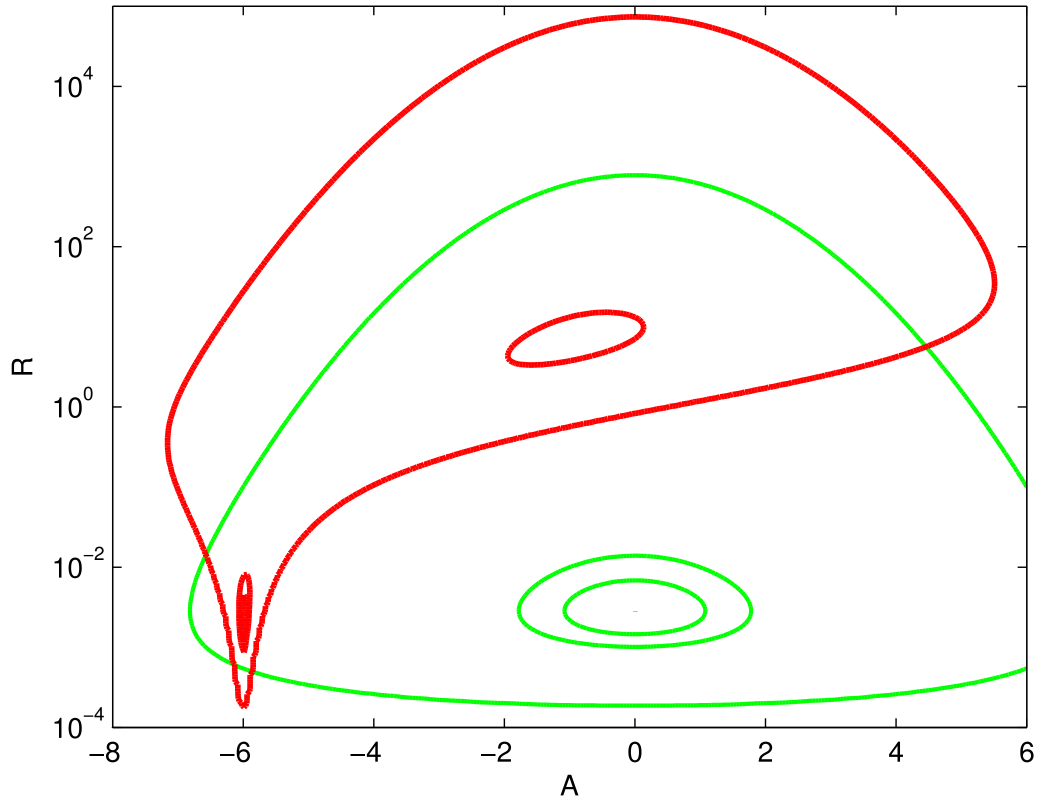
Equiprobability contour of $p(A)p(R)$



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

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

Graphical Models

- formal languages for specification of probability models and associated inference algorithms

- historically, introduced in probabilistic expert systems (Pearl 1988) as a visual guide for representing expert knowledge

- today, a standard tool in machine learning, statistics and signal processing

Graphical Models

- provide graph based algorithms for derivations and computation
- pedagogical insight/motivation for model/algorithm construction
  - Statistics:
    "Kalman filter models and hidden Markov models (HMM) are equivalent upto parametrisation"
  - Signal processing:
    "Fast Fourier transform is an instance of sum-product algorithm on a factor graph"
  - Computer Science:
    "Backtracking in Prolog is equivalent to inference in Bayesian networks with deterministic tables"
- Automated tools for code generation start to emerge, making the design/implement/test cycle shorter

Important types of Graphical Models

- Useful for Model Construction
  - **Directed Acyclic Graphs (DAG), Bayesian Networks**
  - **Undirected Graphs, Markov Networks, Random Fields**
  - Influence diagrams
  - ...
- Useful for Inference
  - **Factor Graphs**
  - Junction/Clique graphs
  - Region graphs
  - ...

Directed Graphical models (DAG)

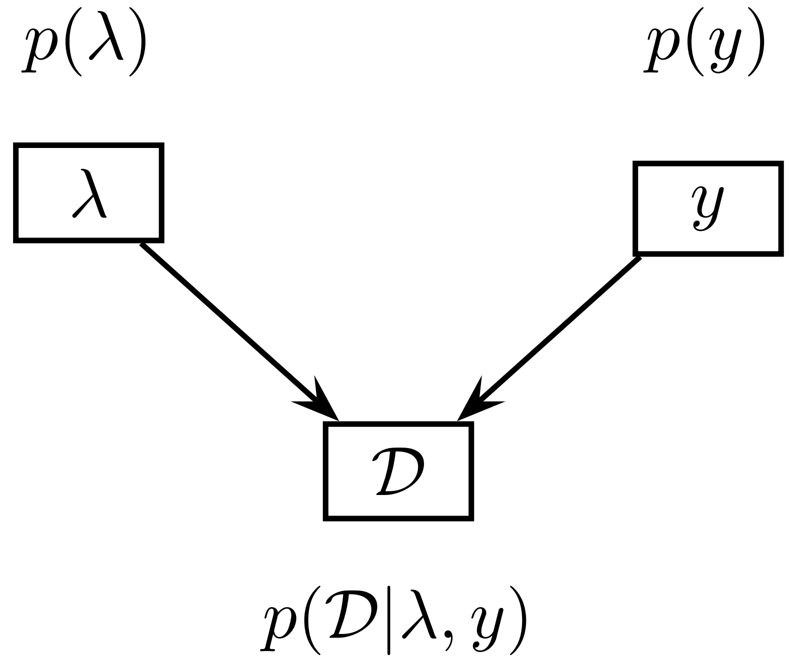
Directed Graphical models

- Each random variable is associated with a node in the graph,
- We draw an arrow from $A \to B$ if $p(B|\dots, A, \dots)$ ($A \in$ parent(B)),
- The edges tell us *qualitatively* about the factorization of the joint probability
- For N random variables $x_1, \dots, x_N$, the distribution admits

$$p(x_1, \dots, x_N) = \prod_{i=1}^{N} p(x_i|\text{parent}(x_i))$$

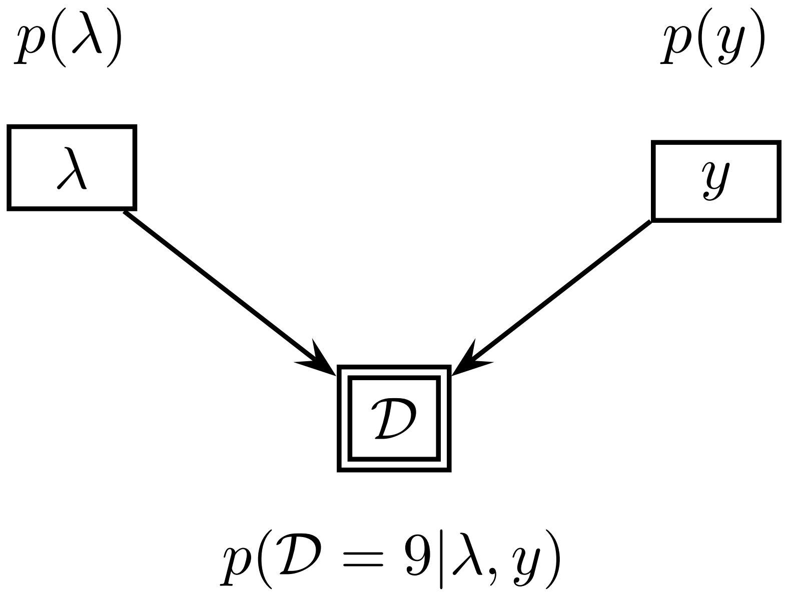
- Describes in a compact way an algorithm to “generate” the data – “Generative models”

DAG Example: Two dice



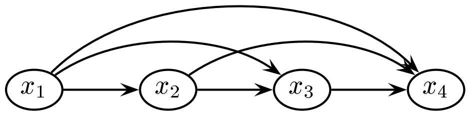
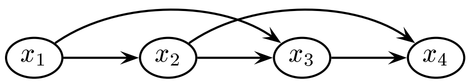
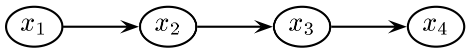
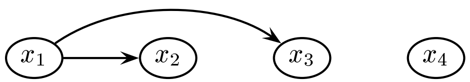
$$p(\mathcal{D}, \lambda, y) = p(\mathcal{D}|\lambda, y)p(\lambda)p(y)$$

DAG with observations



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

Examples

Model	Structure	factorization
Full		$p(x_1)p(x_2\|x_1)p(x_3\|x_1,x_2)p(x_4\|x_1,x_2,x_3)$
Markov(2)		$p(x_1)p(x_2\|x_1)p(x_3\|x_1,x_2)p(x_4\|x_2,x_3)$
Markov(1)		$p(x_1)p(x_2\|x_1)p(x_3\|x_2)p(x_4\|x_3)$
		$p(x_1)p(x_2\|x_1)p(x_3\|x_1)p(x_4)$
Factorized		$p(x_1)p(x_2)p(x_3)p(x_4)$

Removing edges eliminates a term from the conditional probability factors.

Undirected Graphical Models

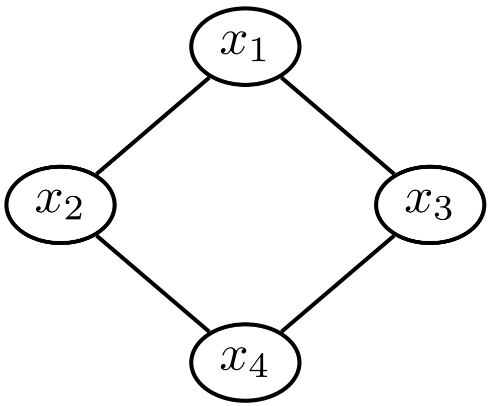
Undirected Graphical Models

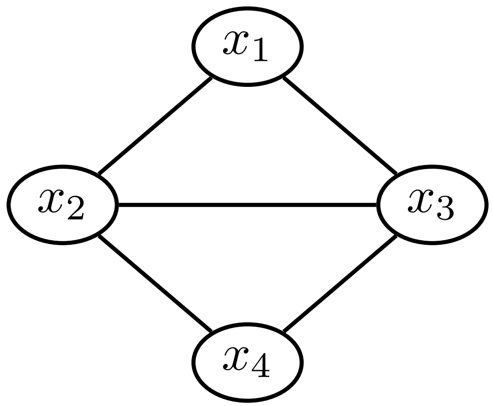
- Define a distribution by non-negative *local compatibility functions* $\phi(x_\alpha)$

$$p(\mathbf{x}) = \frac{1}{Z}\prod_\alpha \phi(x_\alpha)$$

where α runs over **cliques** : fully connected subsets

- Examples



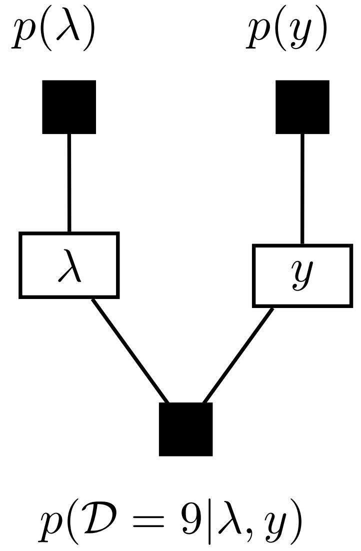


$p(\mathbf{x}) = \frac{1}{Z}\phi(x_1, x_2)\phi(x_1, x_3)\phi(x_2, x_4)\phi(x_3, x_4)$ $\qquad$ $p(\mathbf{x}) = \frac{1}{Z}\phi(x_1, x_2, x_3)\phi(x_2, x_3, x_4)$

Factor graphs

Factor graphs [?]

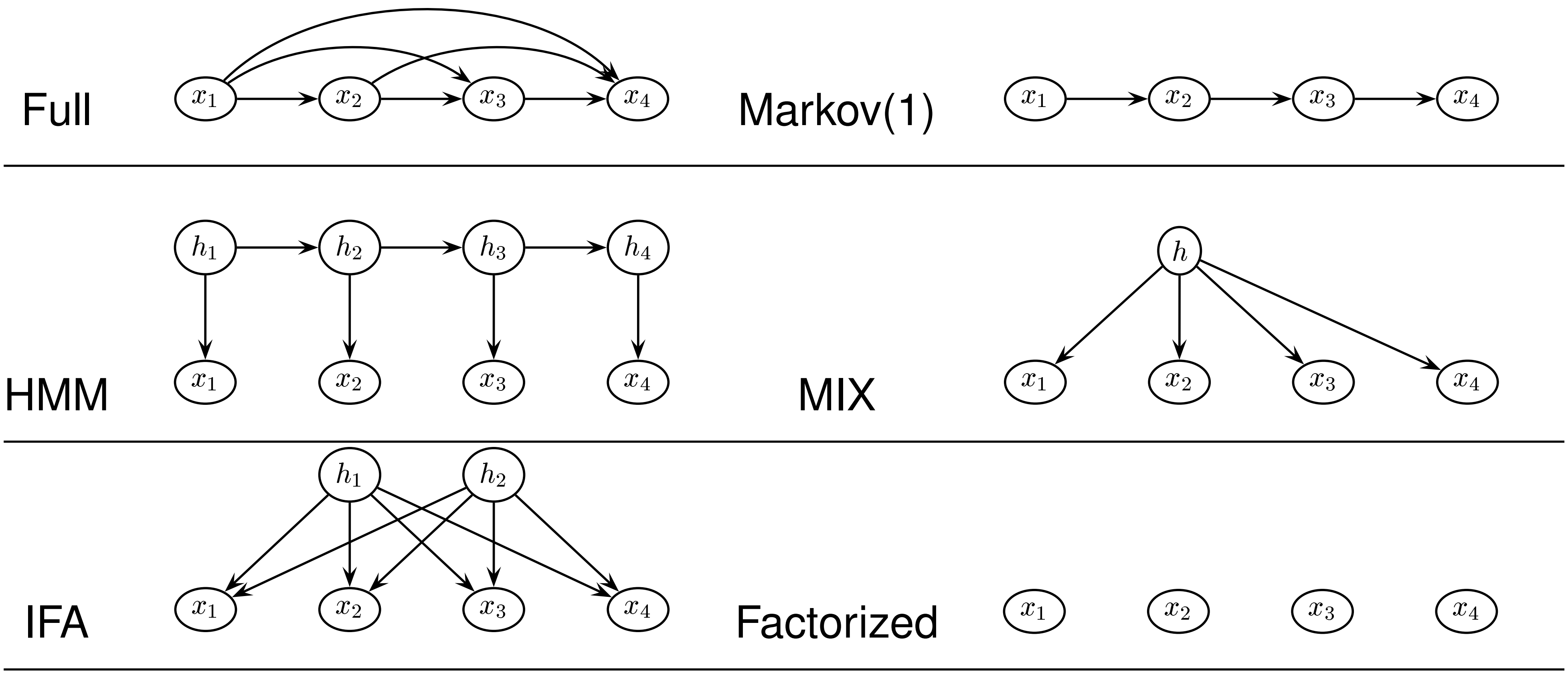
- A bipartite graph. A powerful graphical representation of the inference problem
  - **Factor nodes**: Black squares. Factor potentials (local functions) defining the posterior.
  - **Variable nodes**: White Nodes. Define collections of random variables
  - **Edges**: denote membership. A variable node is connected to a factor node if a member variable is an argument of the local function.



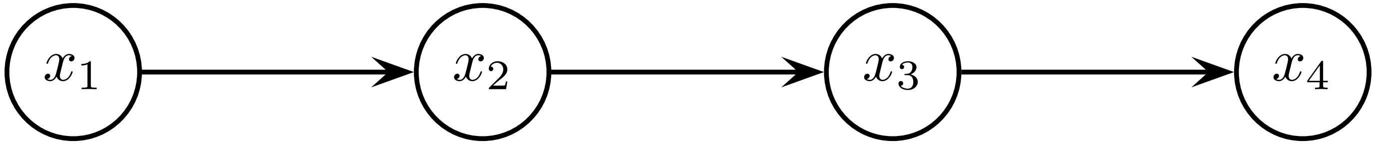
$$\phi_{\mathcal{D}}(\lambda, y) \quad = \quad p(\mathcal{D} = 9|\lambda, y)p(\lambda)p(y) = \phi_1(\lambda, y)\phi_2(\lambda)\phi_3(y)$$

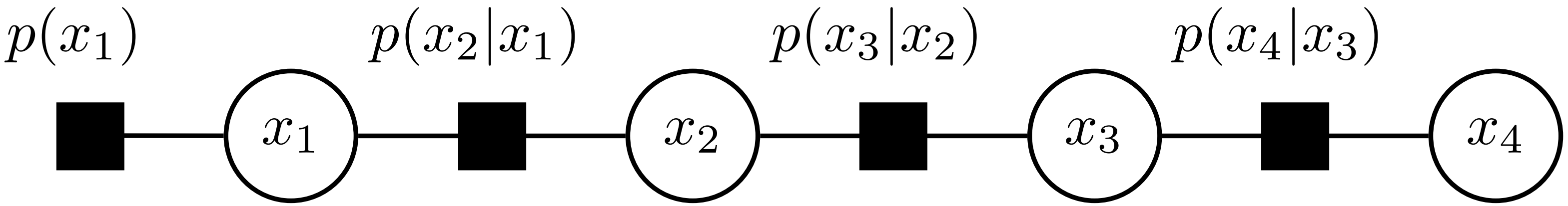
Exercise

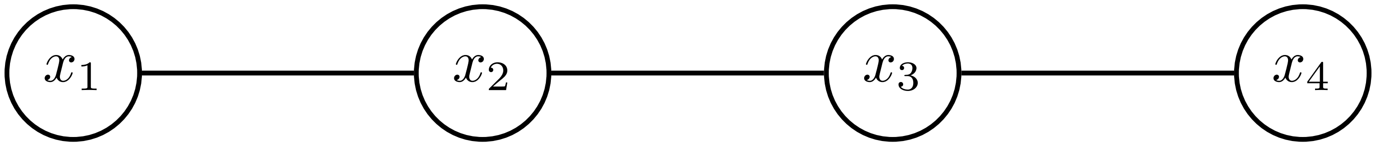
- For the following Graphical models, write down the factors of the joint distribution and plot an equivalent factor graph and an undirected graph.



Answer (Markov(1))

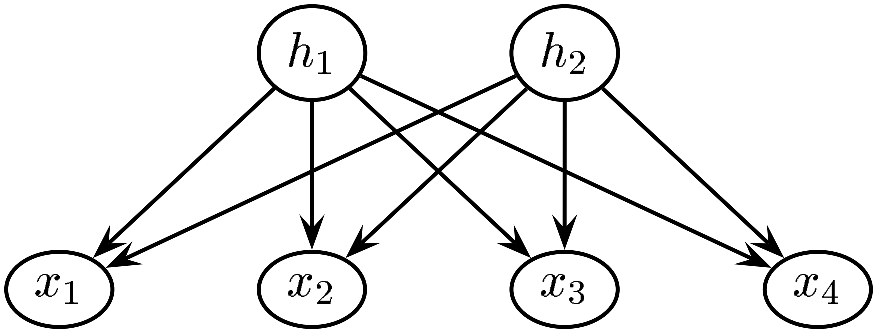




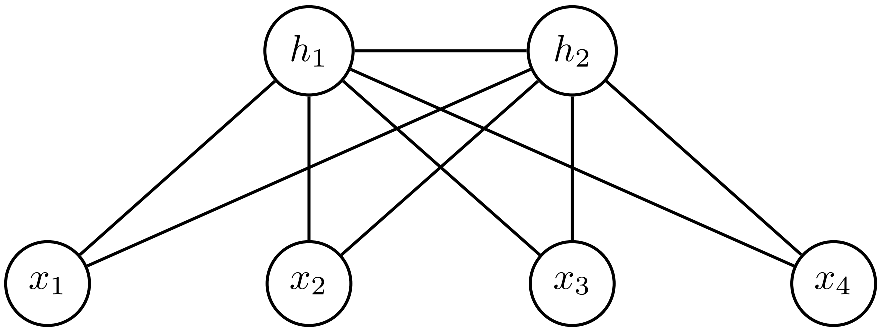


$$\underbrace{p(x_1)p(x_2|x_1)}_{\phi(x_1,x_2)}\underbrace{p(x_3|x_2)}_{\phi(x_2,x_3)}\underbrace{p(x_4|x_3)}_{\phi(x_3,x_4)}$$

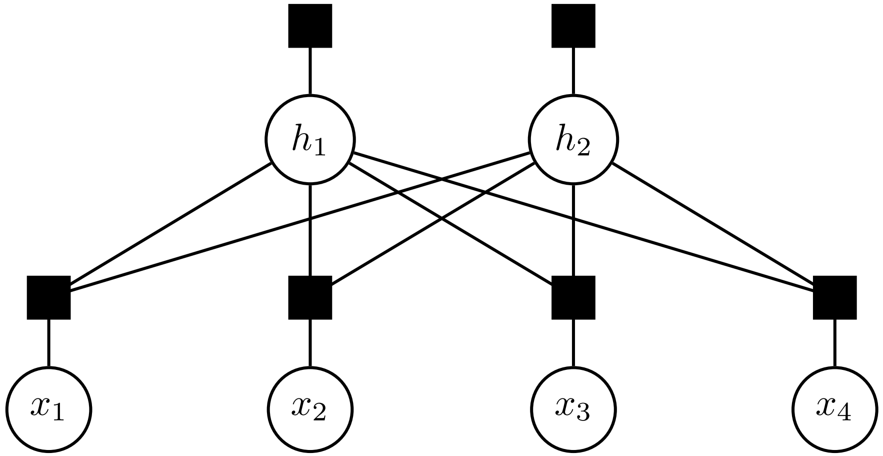
Answer (IFA – Factorial)



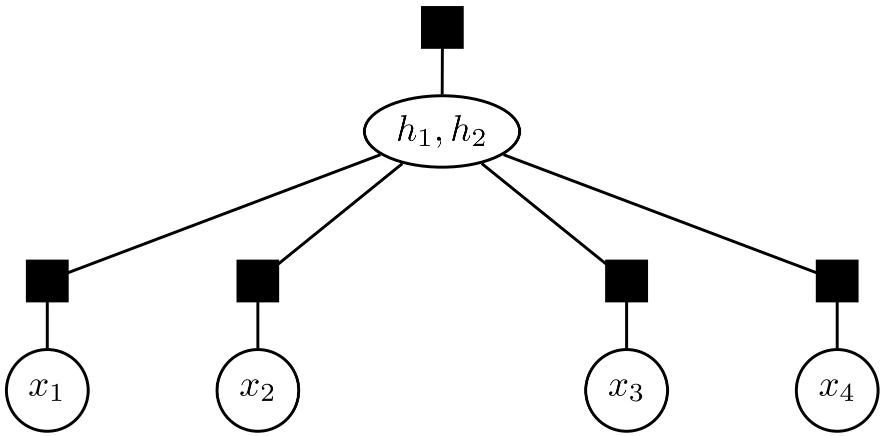
$$p(h_1)p(h_2)\prod_{i=1}^{4}p(x_i|h_1,h_2)$$



Answer (IFA – Factorial)



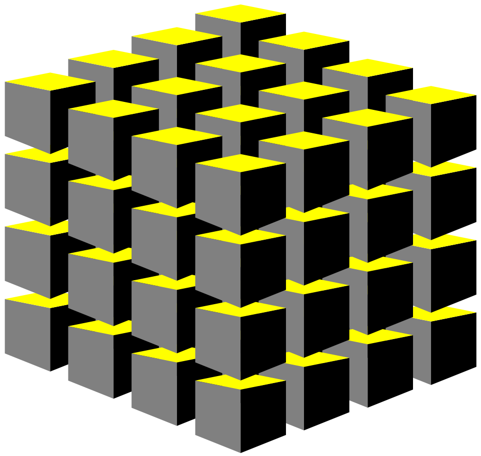
- We can also cluster nodes together



Probability Tables

- Assume all x_i are discrete with $|x_i| = k$. If N is large, a naive table representation is HUGE: k^N entries

Example: $p(x_1, x_2, x_3)$ with $|x_i| = 4$

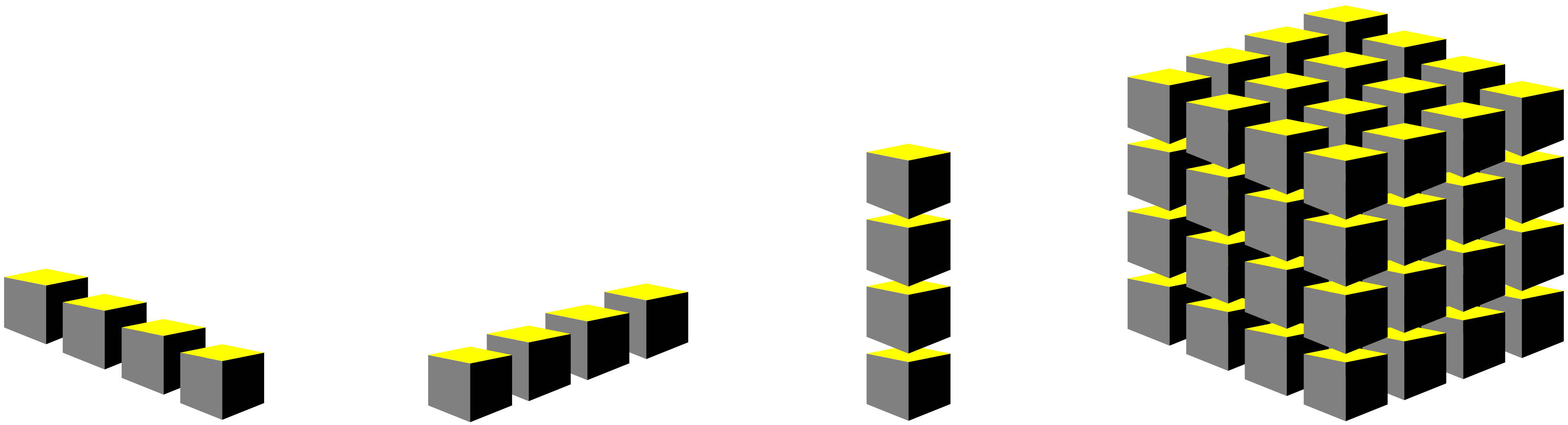


Each cell is a positive number s.t. $\sum_{x_1,x_2,x_3} p(x_1, x_2, x_3) = 1$

- We need efficient data structures to represent joint distributions $p(x_1, x_2, \ldots, x_N)$

Independence Assumption == Complete Factorization

- Assume $p(x_1, x_2, \ldots, x_N) = \prod_k p(x_k)$.

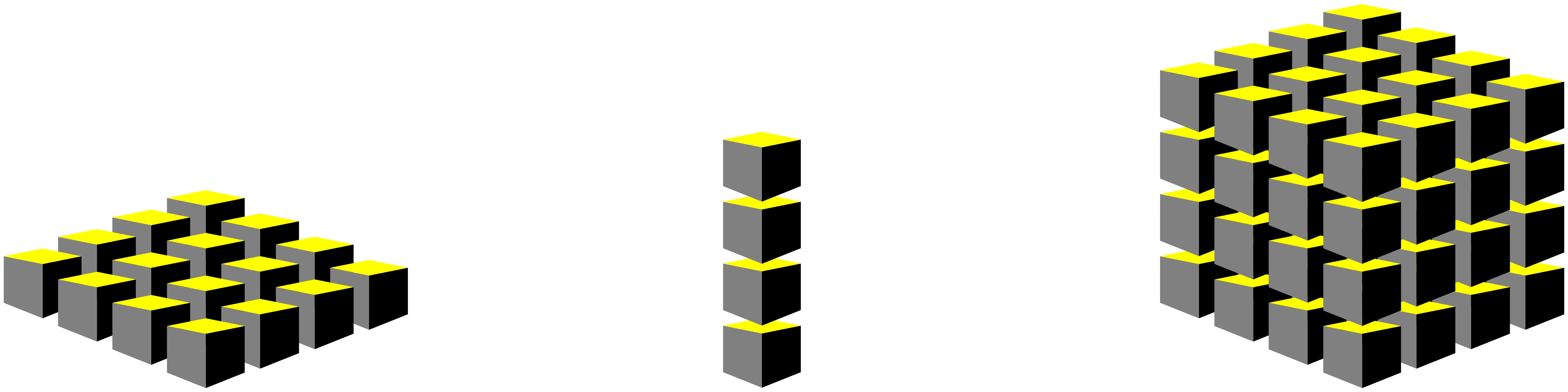


$p(x_1) \quad \times \quad p(x_2) \quad \times \quad p(x_3) \quad = \quad p(x_1, x_2, x_3)$

We need to store 4×3 numbers instead of 4^3 !

- However, complete independence is too restrictive and not very useful.

An alternative Factorization



$$p(x_1, x_2) \quad \times \quad p(x_3) \quad = \quad p(x_1, x_2, x_3)$$

We need to store $4^2 + 4$ numbers instead of 4^3.

- Still some variables are independent from rest. We will make conditional independence assumptions instead.

Conditional Independence

- Two disjoint sets of variables A and B are conditionally independent given a third disjoint set C if

$$p(A, B|C) \quad = \quad p(A|C)p(B|C)$$

- This is equivalent to

$$p(A|BC) \quad = \quad p(A|C)$$

- We denote this relationship with ($\perp\!\!\!\perp$)

$$A \perp\!\!\!\perp B|C$$

Conditional Independence

- Conditional Independence is a key concept in probabilistic models
- Conceptual and Computational simplifications
  - Understanding key factors in a domain
  - Reducing computational burden for inference

Conditional Independence Properties

- Directed Graphical Models
  - d-separation
- Markov Random Fields (MRF's : Undirected Graphical Models)
  - Path Blocking
- Testing for conditional independence in MRF is simpler

d-Separation

- Three disjoint sets of variables A, B and C

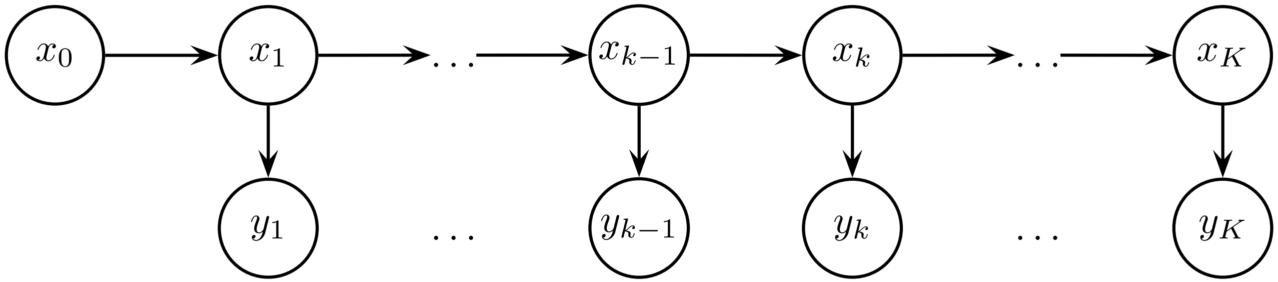
$$A \perp\!\!\!\perp B | C$$

- A path from A to B is blocked by C if

  a the arrows on the path meet either head-to-tail or tail-to-tail at the node, and the node is in the set C, or
  b the arrows meet head-to-head at the node, and neither the node, nor any of its descendants, is in the set C.

Sequential Data: Models, Inference, Terminology

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



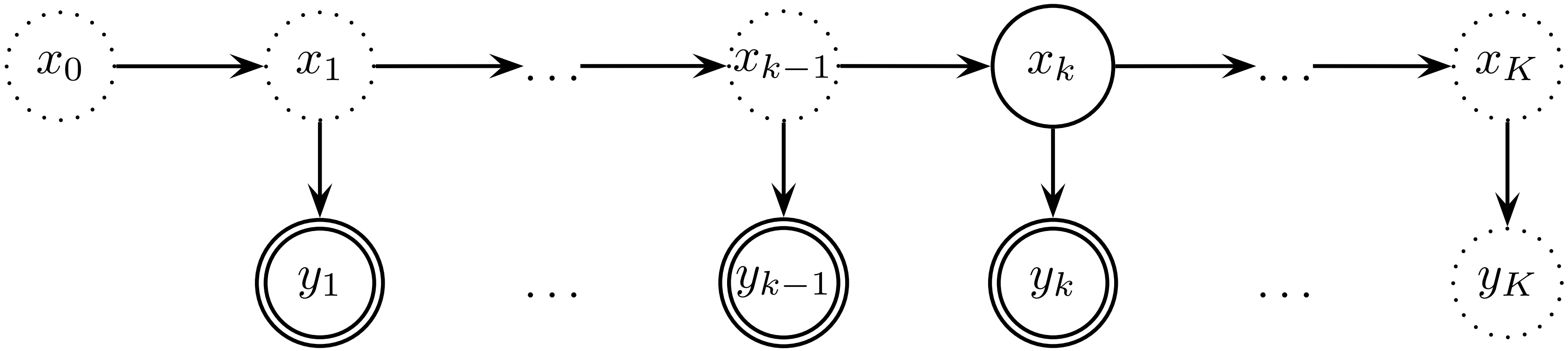
$$x_k \sim p(x_k|x_{k-1}) \quad \text{Transition Model}$$

$$y_k \sim p(y_k|x_k) \quad \text{Observation Model}$$

- 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

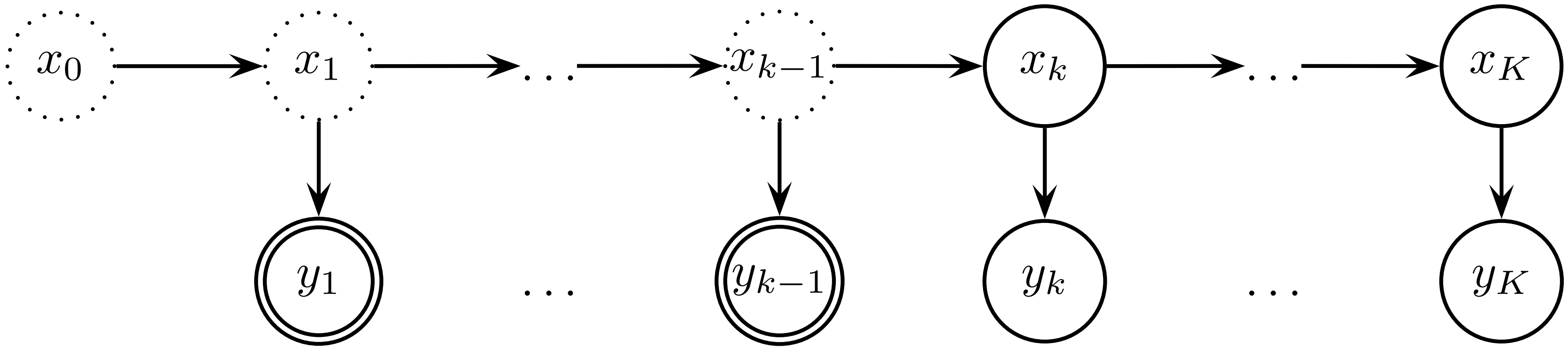
- **Filtering:** $p(x_k|y_{1:k})$
  - Distribution of current state given all past information
  - Realtime/Online/Sequential Processing



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

Online Inference, Terminology

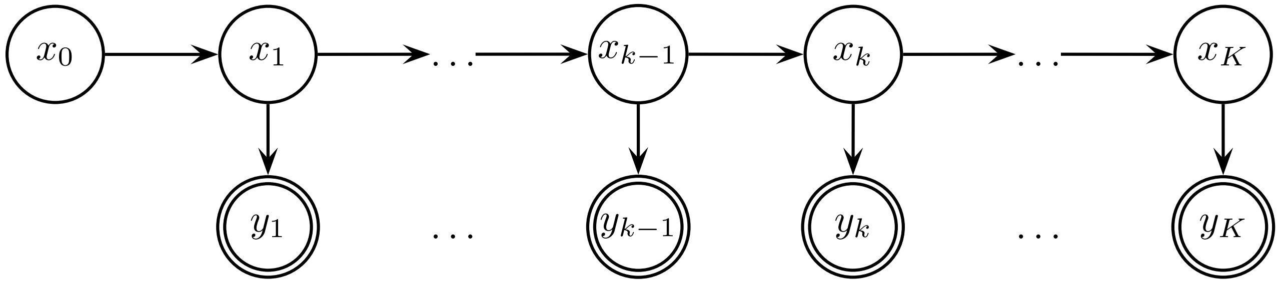
- **Prediction** $p(y_{k:K}, x_{k:K} | y_{1:k-1})$
  - evaluation of possible future outcomes; like filtering without observations



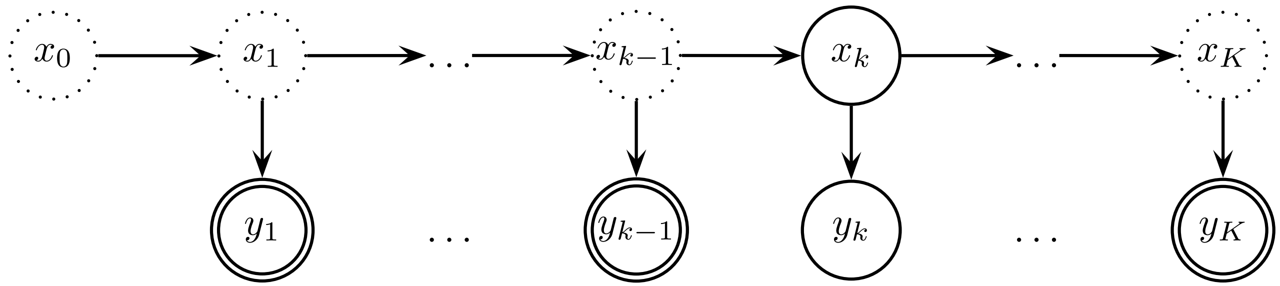
- Accompaniment, Tracking, Restoration

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

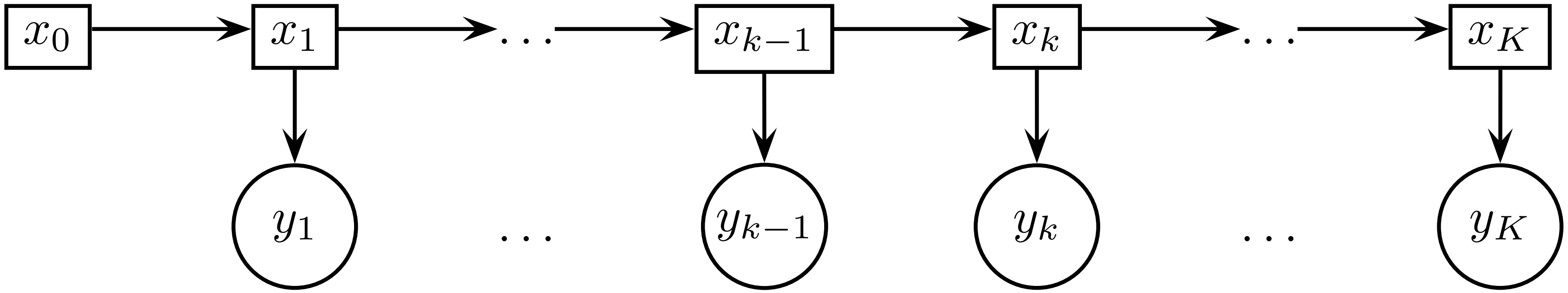


- **Interpolation** $p(y_k, x_k|y_{1:k-1}, y_{k+1:K})$
  fill in lost observations given past and future



Hidden Markov Model [?]

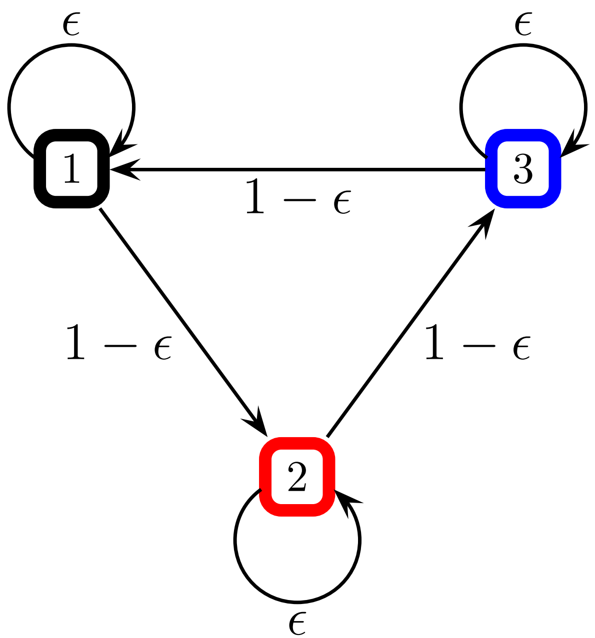
- Mixture model evolving in time



- 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)

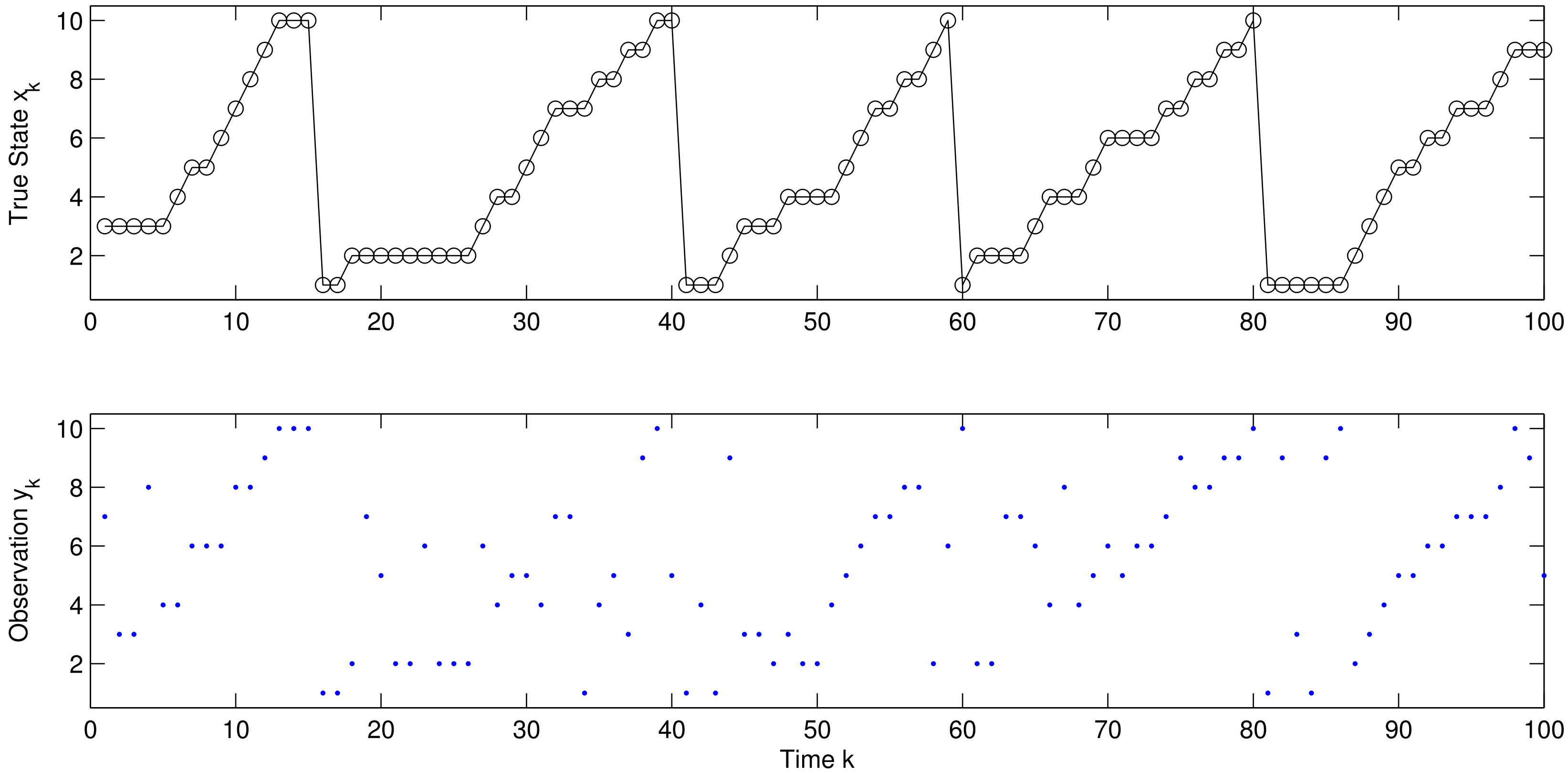


$$(1-\epsilon)\begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} + \epsilon \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

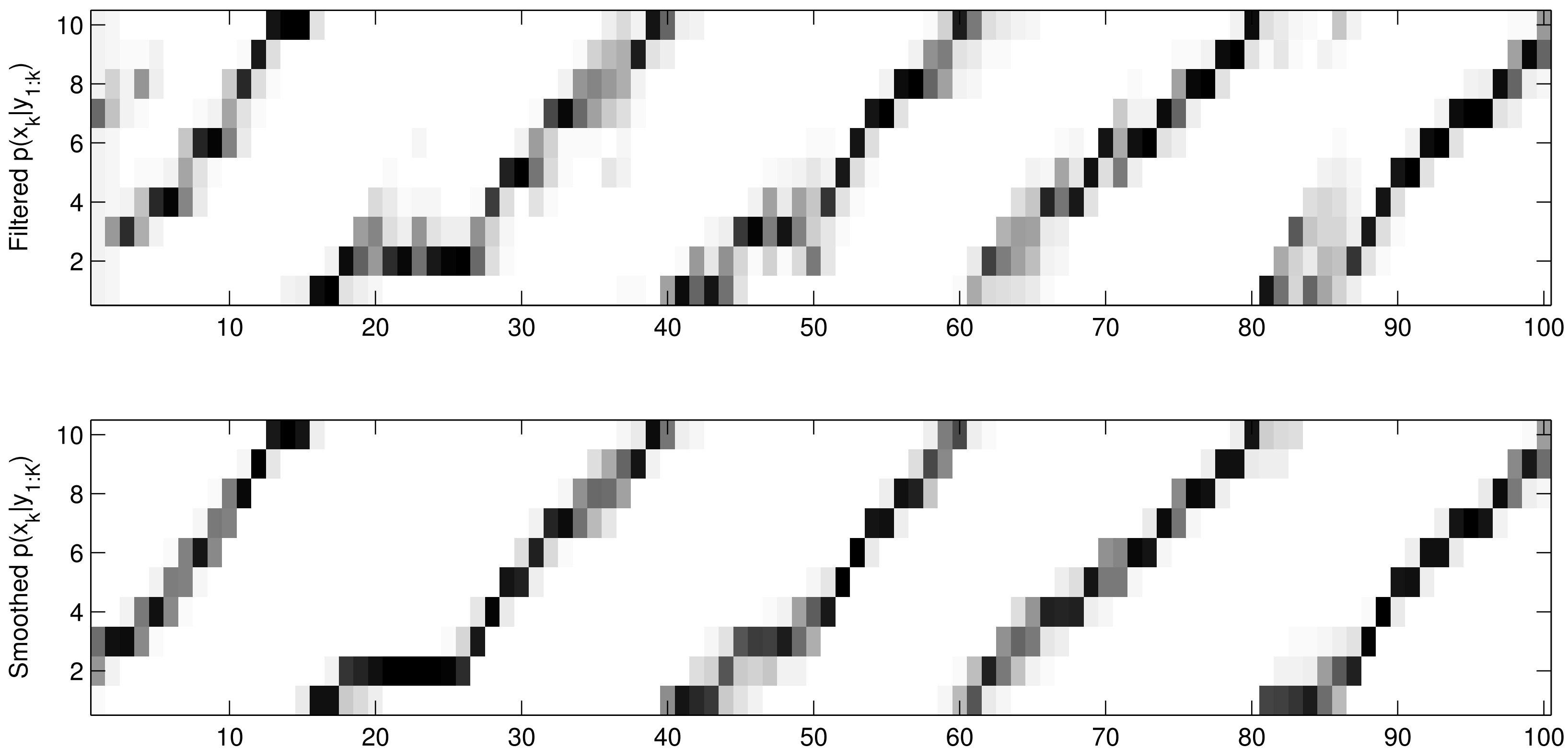
- Observation model $p(y_k|x_k)$

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

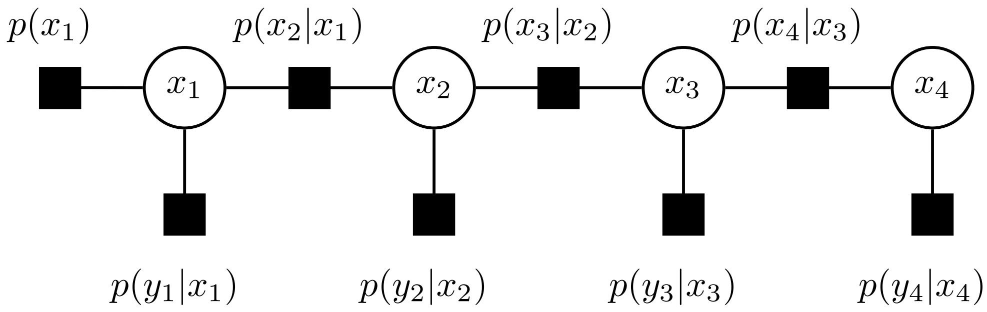
Example: Hidden Markov Model



Example: Hidden Markov Model



Exact Inference in HMM, Forward/Backward Algorithm



- Forward Pass

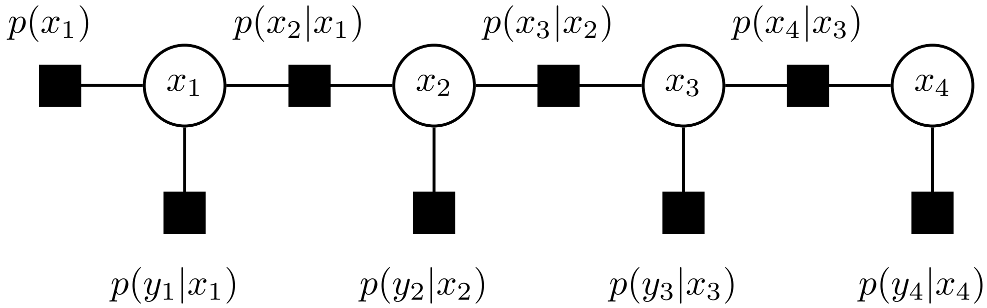
$$p(y_{1:K}) = \sum_{x_{1:K}} p(y_{1:K}|x_{1:K})p(x_{1:K})$$

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

- 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

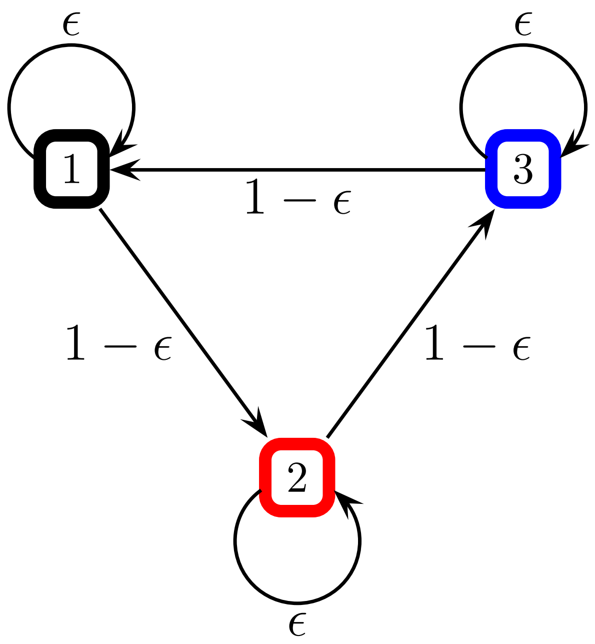
2. Generate data from the true model

3. Inference given true model parameters

4. Test and Visualisation

Example: Hidden Markov Model

- State transition model (a N by N matrix)



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

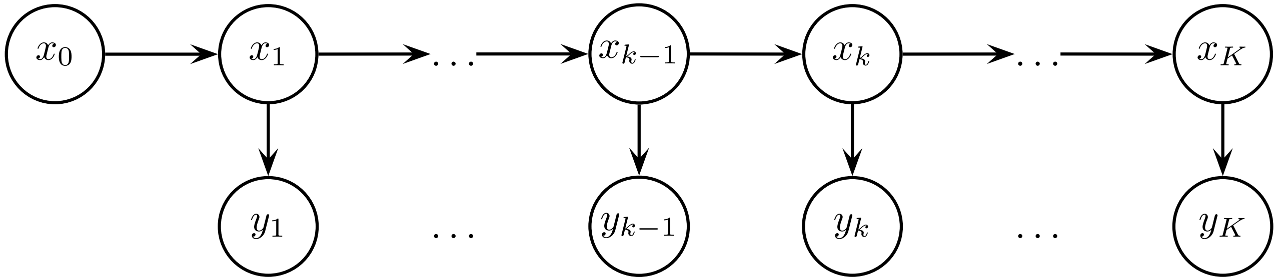
- Observation model $p(y_k|x_k)$

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

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

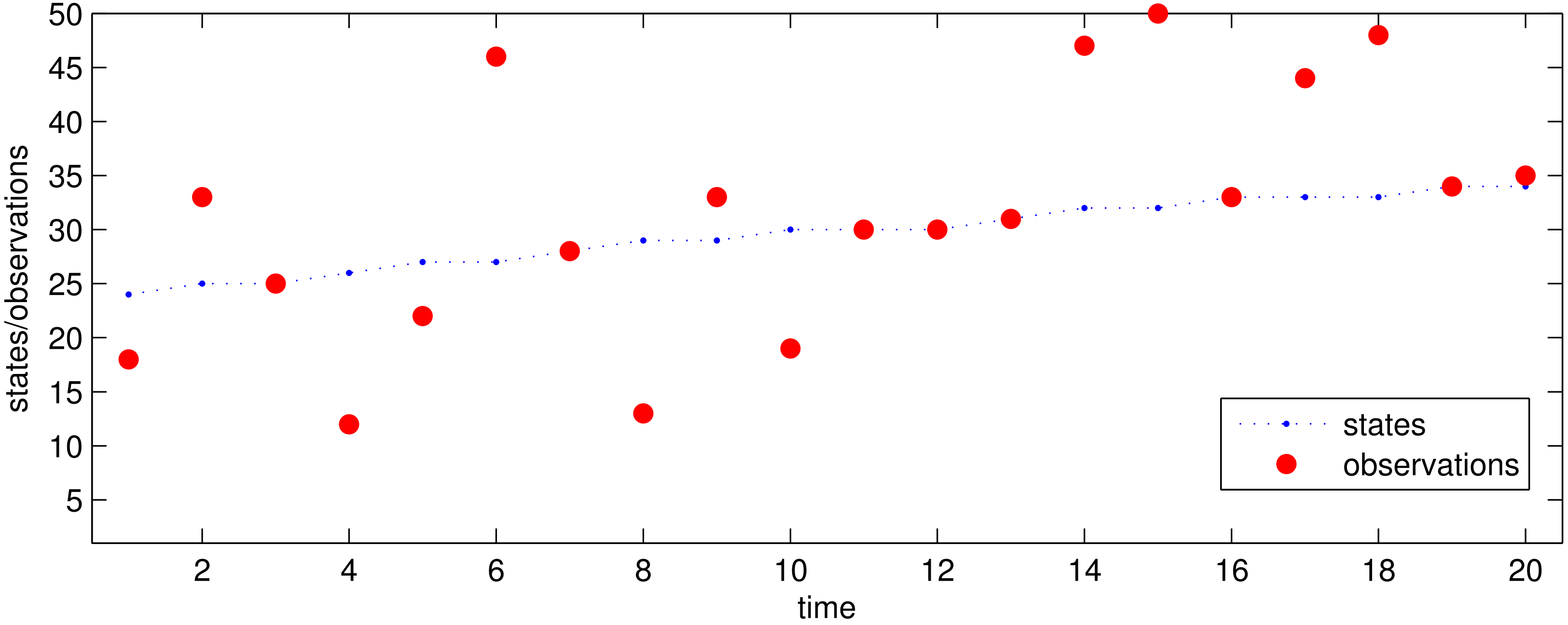


$$
\begin{aligned}
x_k | x_{k-1} &\sim p(x_k | x_{k-1}) \\
y_k | x_k &\sim p(y_k | x_k)
\end{aligned}
$$

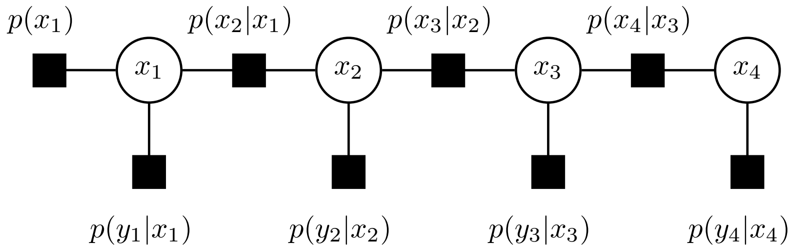
2. Generate data from the true model

```
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

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

$$= \sum_{x_{k-1}} p(x_k|x_{k-1}) \alpha_{k-1|k-1}(x_{k-1})$$

- Update

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

$$= p(y_k|x_k) \alpha_{k|k-1}(x_k)$$

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

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 $l^* = \max_i l_i$

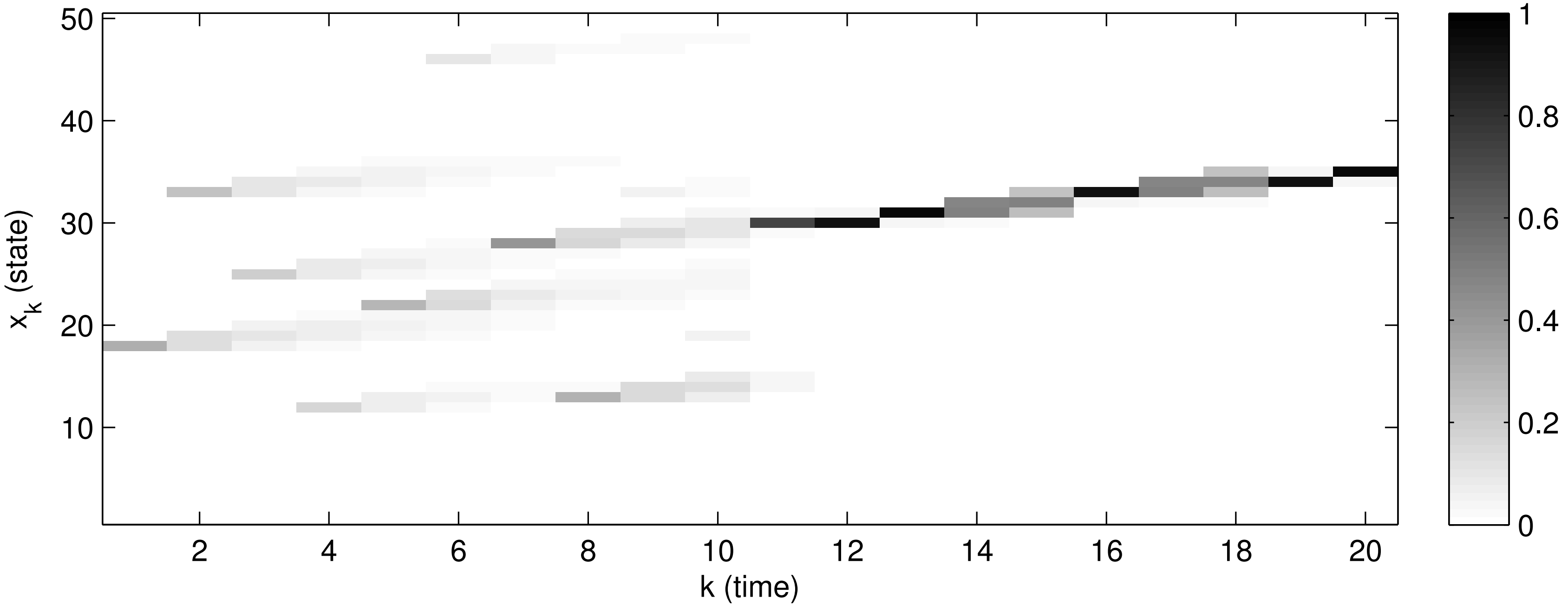
- Assignment: Implement above as a function `logsumexp(l)`

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;
```

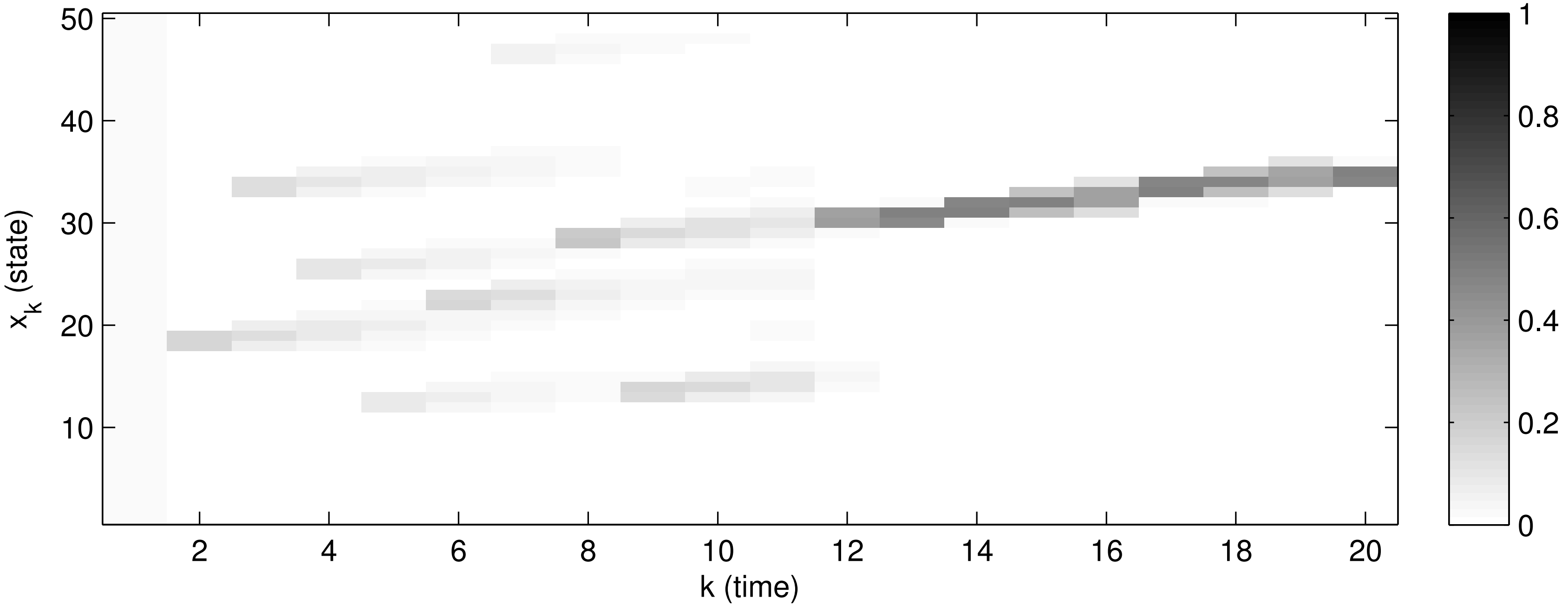
3. Inference. Forward pass.

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

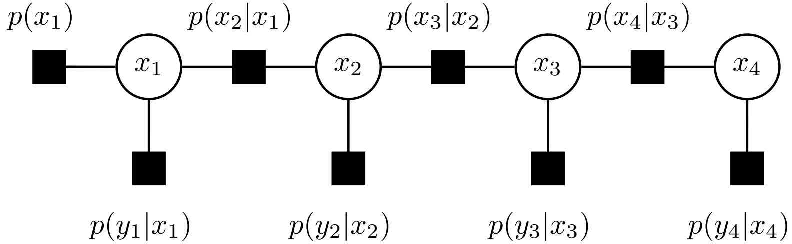


3. Inference. Forward pass

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



3. Inference. Backward pass



- "Postdict"

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

- Update

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

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

3. Inference. Backward pass

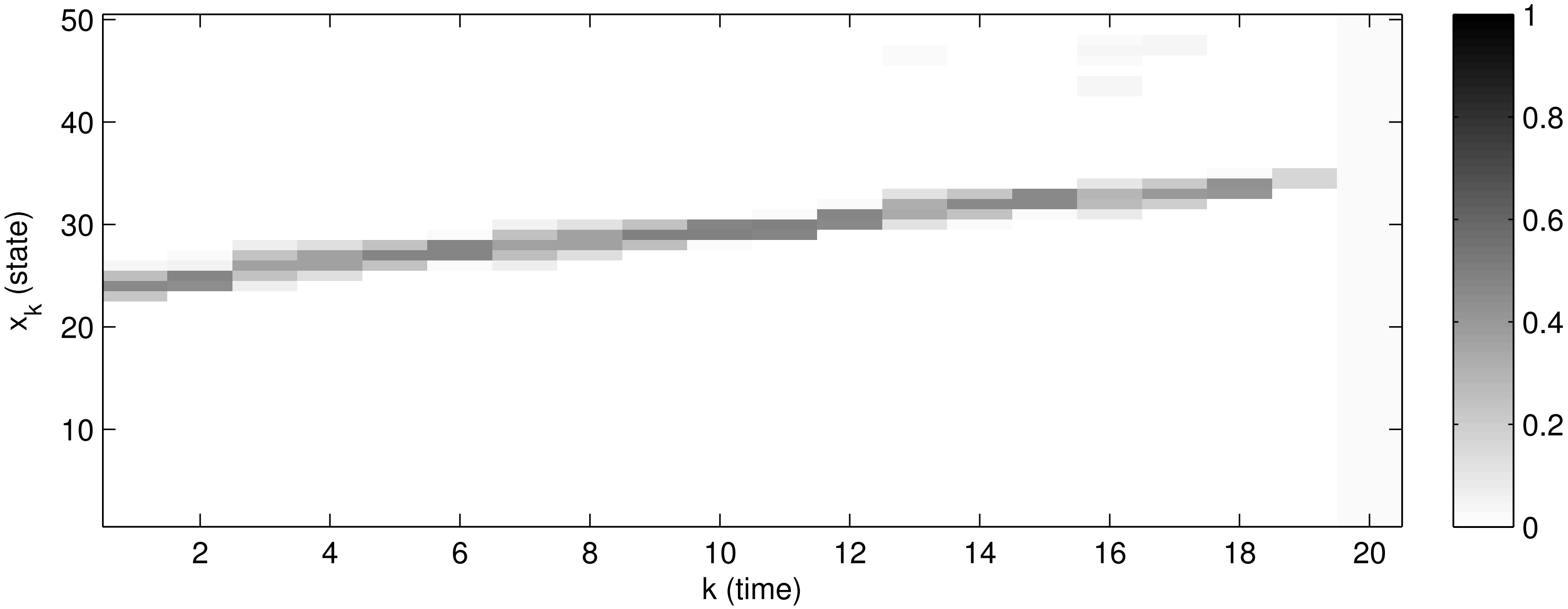
```
log_beta = zeros(N, T);
log_beta_postdict = zeros(N, T);
for t=T:-1:1,
    if t==T,
        log_beta_postdict(:,t) = zeros(N,1);
    else
        log_beta_postdict(:,t) ...
                = state_postdict(hm.A, log_beta(:, t+1));
    end;
    log_beta(:, t) ...
         = state_update(hm.C(y(t),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

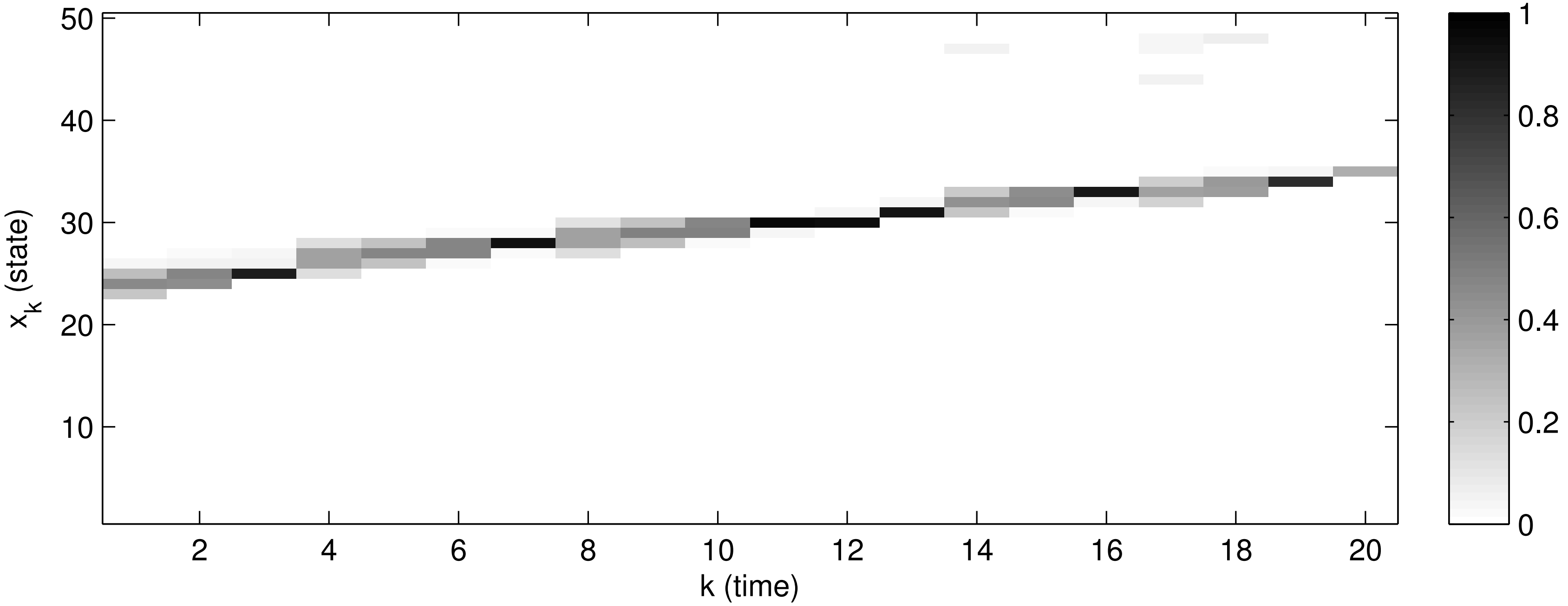
$$\beta_{k|k+1}(x_k) = p(y_{k+1:K}|x_k)$$



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

3. Inference. Backward pass

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



3. Inference. Smoothing.

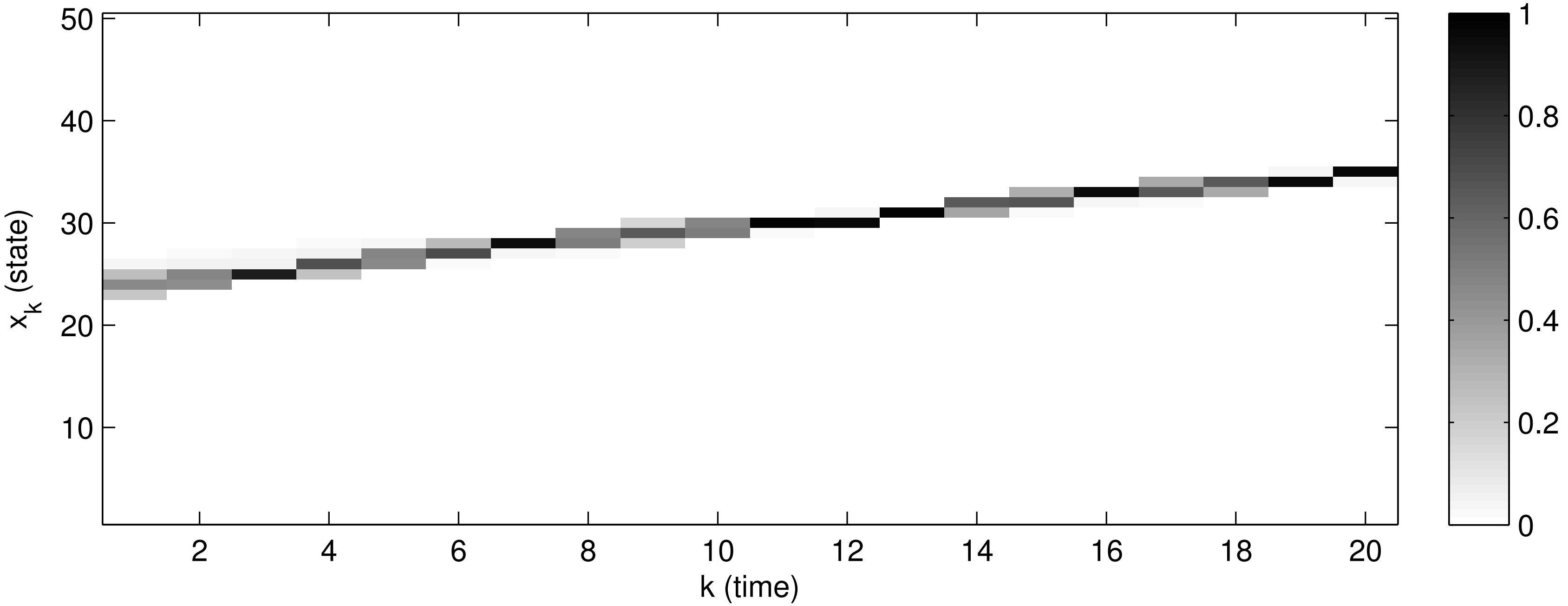
$$
\begin{aligned}
p(y_{1:K}, x_k) &= p(y_{1:k}, x_k) p(y_{k+1:K} | x_k) \\
&= \alpha_{k|k}(x_k) \beta_{k|k+1}(x_k) \\
&\equiv \gamma_k(x_k)
\end{aligned}
$$

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.

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



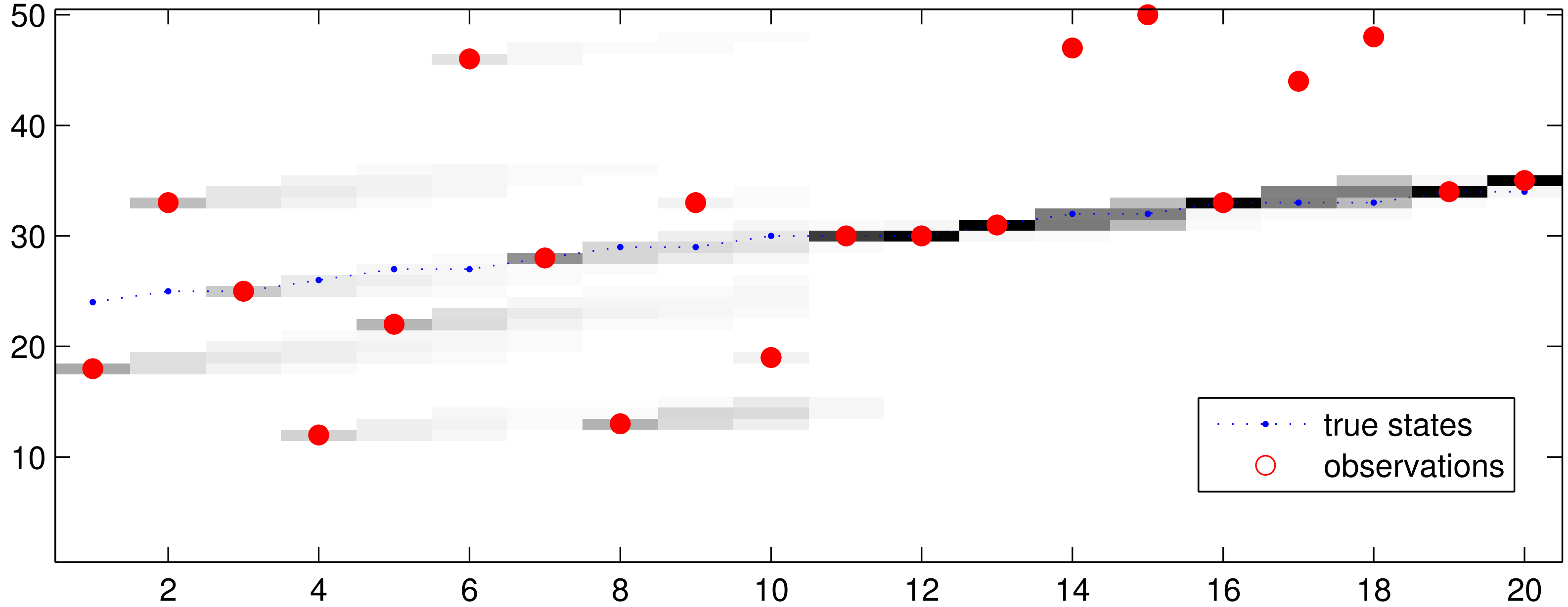
3. Inference. Smoothing.

```
log_gamma = log_alpha + log_beta_postdict
```

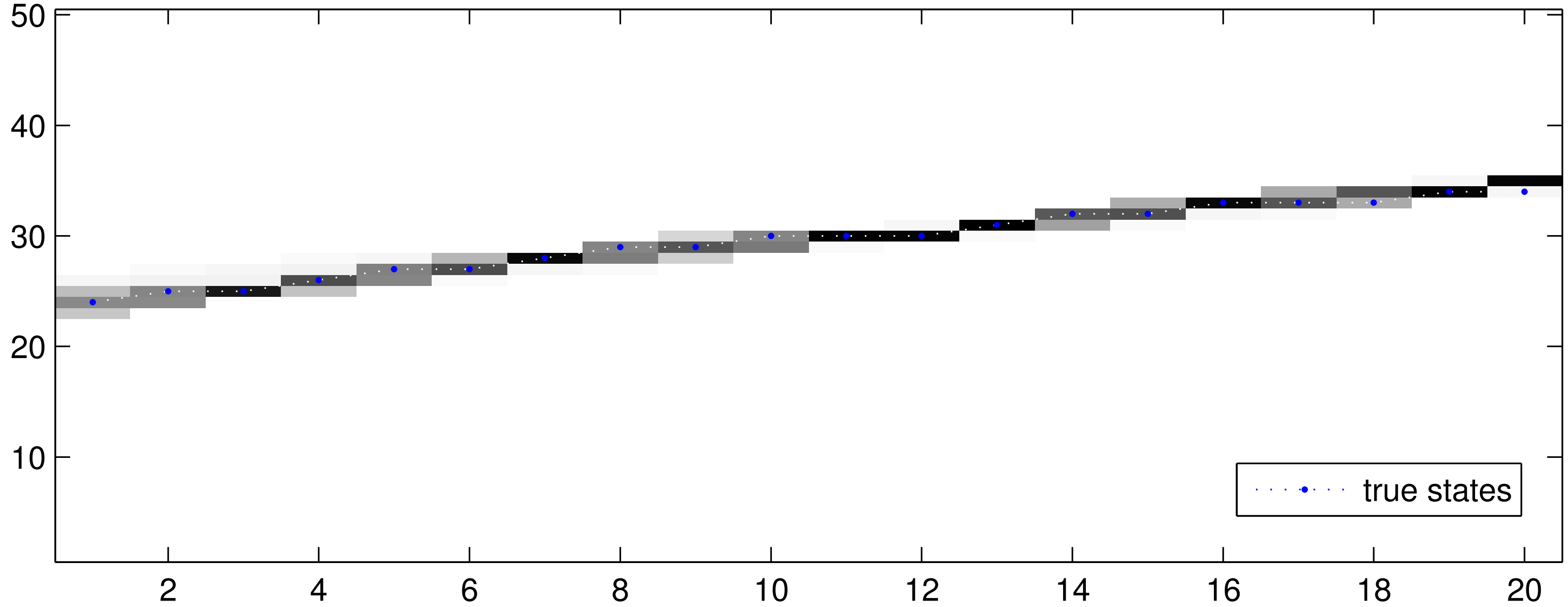
4. Test and Visualisation

```
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.



The Multivariate Gaussian Distribution

The Multivariate Gaussian Distribution. $\mathcal{N}(s; \mu, P)$

μ is the mean and P is the covariance:

$$
\begin{aligned}
\mathcal{N}(s; \mu, P) &= |2\pi P|^{-1/2} \exp\left(-\frac{1}{2}(s-\mu)^\top P^{-1}(s-\mu)\right) \\
&= \exp\left(-\frac{1}{2}s^\top P^{-1} s + \mu^\top P^{-1} s - \frac{1}{2}\mu^\top P^{-1}\mu - \frac{1}{2}|2\pi P|\right) \\
\log \mathcal{N}(s; \mu, P) &= -\frac{1}{2}s^\top P^{-1} s + \mu^\top P^{-1} s + \text{ const} \\
&= -\frac{1}{2}\,\mathbf{Tr}\, P^{-1} s s^\top + \mu^\top P^{-1} s + \text{ const} \\
&=^+ -\frac{1}{2}\,\mathbf{Tr}\, P^{-1} s s^\top + \mu^\top P^{-1} s
\end{aligned}
$$

Notation: $\log f(x) =^+ g(x) \iff f(x) \propto \exp(g(x)) \iff \exists c \in \mathbb{R} : f(x) = c\exp(g(x))$

Gaussian potentials

Consider a Gaussian potential with mean μ and covariance Σ on x.

$$\phi(x) = \alpha \mathcal{N}(\mu, \Sigma) \quad (2)$$

$$= \alpha |2\pi\Sigma|^{-\frac{1}{2}} \exp(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)) \quad (3)$$

where $\int dx \phi(x) = \alpha$ and $|2\pi\Sigma|$ is a short notation for $(2\pi)^d \det \Sigma$, where Σ is $d \times d$..

- If $\alpha = 1$ the potential is normalized.
- A general Gaussian potential ϕ need not to be normalized so α is in fact an arbitrary positive constant.
- The exponent is just a quadratic form.

Canonical Form

$$\begin{aligned}\phi(x) &= \exp(\{\log\alpha - \frac{1}{2}\log|2\pi\Sigma| - \frac{1}{2}\mu^T\Sigma^{-1}\mu\} + \mu^T\Sigma^{-1}x - \frac{1}{2}x^T\Sigma^{-1}x) \\ &= \exp(g + h^T x - \frac{1}{2}x^T K x)\end{aligned}$$

- Alternative to the conventional and intuitive moment form.
- Here we represent the potential by the polynomial coefficients h and K.
- Coefficients h and K as natural parameters.

Canonical and Moment parametrisations

The moment parameters and canonical parameters are related by

$$
\begin{aligned}
K &= \Sigma^{-1} \\
h &= \Sigma^{-1}\mu \\
g &= \log\alpha - \frac{1}{2}\log|2\pi\Sigma| - \frac{1}{2}\mu^T\Sigma^{-1}\Sigma\Sigma^{-1}\mu \\
&= \log\alpha + \frac{1}{2}\log|\frac{K}{2\pi}| - \frac{1}{2}h^T K^{-1} h
\end{aligned}
$$

Jointly Gaussian Vectors

- Moment form

$$\phi(x_1, x_2) \quad = \quad \alpha\mathcal{N}(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix})$$

$$\phi = \alpha|2\pi\Sigma|^{-\frac{1}{2}} \exp(-\frac{1}{2} \begin{pmatrix} x_1 - \mu_1 & x_2 - \mu_2 \end{pmatrix} \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}^{-1} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix})$$

- Canonical form

$$\phi(x_1, x_2) = \exp(g + \begin{pmatrix} h_1 & h_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \frac{1}{2} \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix})$$

- need to find a parametric representation of $K = \Sigma^{-1}$ in terms of the partitions $\Sigma_{11}, \Sigma_{12}, \Sigma_{21}, \Sigma_{22}$.

Partitioned Matrix Inverse

- Strategy: We will find two matrices X and Z such that W becomes block diagonal.

$$
\begin{aligned}
L\Sigma R &= W \\
\Sigma &= L^{-1} W R^{-1} \\
\Sigma^{-1} &= R W^{-1} L = K
\end{aligned}
$$

Gauss Transformations

- Add a multiple of row s to row t
- Premultiply Σ with $L(s,t)$ where

$$L_{i,j}(s,t) = \begin{cases} 1, & i = j \\ \gamma, & i = s \text{ and } j = t \\ 0, & \text{o/w} \end{cases}$$

- Example: $s = 2$, $t = 1$

$$\begin{pmatrix} 1 & \gamma \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} a + \gamma c & b + \gamma d \\ c & d \end{pmatrix}$$

- The inverse just subtracts what is added

$$\begin{pmatrix} 1 & \gamma \\ 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & -\gamma \\ 0 & 1 \end{pmatrix}$$

Gauss Transformations

- Given Σ, add a multiple of column s to column t
- Postmultiply Σ with $R(s,t)$ where

$$R_{i,j}(s,t) = \begin{cases} 1, & i = j \\ \gamma, & j = s \text{ and } i = t \\ 0, & \text{o/w} \end{cases}$$

- Example: $s = 2$, $t = 1$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \gamma & 1 \end{pmatrix} = \begin{pmatrix} a + \gamma b & b \\ c + \gamma d & d \end{pmatrix}$$

Scalar example

$$
\begin{aligned}
\Sigma &= \begin{pmatrix} a & b \\ c & d \end{pmatrix} \\
L\Sigma &= \begin{pmatrix} 1 & -bd^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} a - bd^{-1}c & \cancel{b - bd^{-1}d} \\ c & d \end{pmatrix} \\
L\Sigma R &= \begin{pmatrix} 1 & -bd^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -d^{-1}c & 1 \end{pmatrix} \\
&= \begin{pmatrix} a - bd^{-1}c & 0 \\ c & d \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -d^{-1}c & 1 \end{pmatrix} \\
&= \begin{pmatrix} a - bd^{-1}c & 0 \\ \cancel{c - dd^{-1}c} & d \end{pmatrix} = \begin{pmatrix} a - bd^{-1}c & 0 \\ 0 & d \end{pmatrix} = W
\end{aligned}
$$

Scalar example (cont)

$$
\begin{aligned}
\Sigma &= L^{-1} W R^{-1} \\
\Sigma^{-1} &= R W^{-1} L \\
&= \begin{pmatrix} 1 & 0 \\ -d^{-1}c & 1 \end{pmatrix} \begin{pmatrix} (a - bd^{-1}c)^{-1} & 0 \\ 0 & d^{-1} \end{pmatrix} \begin{pmatrix} 1 & -bd^{-1} \\ 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} (a - bd^{-1}c)^{-1} & 0 \\ -d^{-1}c(a - bd^{-1}c)^{-1} & d^{-1} \end{pmatrix} \begin{pmatrix} 1 & -bd^{-1} \\ 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} (a - bd^{-1}c)^{-1} & -(a - bd^{-1}c)^{-1}bd^{-1} \\ -d^{-1}c(a - bd^{-1}c)^{-1} & d^{-1} + d^{-1}c(a - bd^{-1}c)^{-1}bd^{-1} \end{pmatrix}
\end{aligned}
$$

Scalar example

We could also use

$$
\begin{aligned}
\Sigma &= \begin{pmatrix} a & b \\ c & d \end{pmatrix} \\
L\Sigma &= \begin{pmatrix} 1 & 0 \\ -ca^{-1} & 1 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} \\
L\Sigma R &= \begin{pmatrix} 1 & 0 \\ -ca^{-1} & 1 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} 1 & -a^{-1}b \\ 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} a & 0 \\ 0 & d - ca^{-1}b \end{pmatrix} = W \\
RW^{-1}L &= \begin{pmatrix} a^{-1} + a^{-1}b\left(d - ca^{-1}b\right)^{-1} ca^{-1} & -a^{-1}b\left(d - ca^{-1}b\right)^{-1} \\ -\left(d - ca^{-1}b\right)^{-1} ca^{-1} & \left(d - ca^{-1}b\right)^{-1} \end{pmatrix}
\end{aligned}
$$

Partitioned Matrix Inverse

In matrix case, this leads to following dual factorizations of Σ as

$$\begin{aligned}
\Sigma &= \begin{pmatrix} I & -\Sigma_{12}\Sigma_{22}^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} \Sigma_{11} - \Sigma_{12}(\Sigma_{22})^{-1}\Sigma_{21} & 0 \\ 0 & \Sigma_{22} \end{pmatrix} \begin{pmatrix} I & 0 \\ -\Sigma_{22}^{-1}\Sigma_{21} & I \end{pmatrix} \\
&= \begin{pmatrix} I & 0 \\ -\Sigma_{21}\Sigma_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \Sigma_{11} & 0 \\ 0 & \Sigma_{22} - \Sigma_{21}(\Sigma_{11})^{-1}\Sigma_{12} \end{pmatrix} \begin{pmatrix} I & -\Sigma_{11}^{-1}\Sigma_{12} \\ 0 & I \end{pmatrix}
\end{aligned}$$

The Schur Complement

We will introduce the notation

$$\begin{aligned}
\Sigma/\Sigma_{22} &= \Sigma_{11} - \Sigma_{12}(\Sigma_{22})^{-1}\Sigma_{21} \\
\Sigma/\Sigma_{11} &= \Sigma_{22} - \Sigma_{21}(\Sigma_{11})^{-1}\Sigma_{12}
\end{aligned}$$

Determinant

$$|\Sigma| = |\Sigma/\Sigma_{11}||\Sigma_{11}| = |\Sigma/\Sigma_{22}||\Sigma_{22}|$$

$$\begin{aligned}
\Sigma^{-1} &= \begin{pmatrix} I & 0 \\ -\Sigma_{22}^{-1}\Sigma_{21} & I \end{pmatrix} \begin{pmatrix} (\Sigma/\Sigma_{22})^{-1} & 0 \\ 0 & \Sigma_{22}^{-1} \end{pmatrix} \begin{pmatrix} I & -\Sigma_{12}\Sigma_{22}^{-1} \\ 0 & I \end{pmatrix} \\
&= \begin{pmatrix} I & -\Sigma_{11}^{-1}\Sigma_{12} \\ 0 & I \end{pmatrix} \begin{pmatrix} \Sigma_{11}^{-1} & 0 \\ 0 & (\Sigma/\Sigma_{11})^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -\Sigma_{21}\Sigma_{11}^{-1} & I \end{pmatrix}
\end{aligned}$$

Partitioned Matrix Inverse

$$
\begin{aligned}
\begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} &= \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}^{-1} \\
&= \begin{pmatrix} (\Sigma/\Sigma_{22})^{-1} & -(\Sigma/\Sigma_{22})^{-1}\Sigma_{12}\Sigma_{22}^{-1} \\ -\Sigma_{22}^{-1}\Sigma_{21}(\Sigma/\Sigma_{22})^{-1} & \Sigma_{22}^{-1} + \Sigma_{22}^{-1}\Sigma_{21}(\Sigma/\Sigma_{22})^{-1}\Sigma_{12}\Sigma_{22}^{-1} \end{pmatrix} \\
&= \begin{pmatrix} \Sigma_{11}^{-1} + \Sigma_{11}^{-1}\Sigma_{12}(\Sigma/\Sigma_{11})^{-1}\Sigma_{21}\Sigma_{11}^{-1} & -\Sigma_{11}^{-1}\Sigma_{12}(\Sigma/\Sigma_{11})^{-1} \\ -(\Sigma/\Sigma_{11})^{-1}\Sigma_{21}\Sigma_{11}^{-1} & (\Sigma/\Sigma_{11})^{-1} \end{pmatrix}
\end{aligned}
$$

- Quite complicated looking formulas, but straightforward to implement
- **Caution:** $\Sigma_{11}^{-1} \neq K_{11}$ in general!

Matrix Inversion Lemma

- Read the diagonal entries

$$
\begin{aligned}
(\Sigma_{11} - \Sigma_{12}(\Sigma_{22})^{-1}\Sigma_{21})^{-1} &= \Sigma_{11}^{-1} + \Sigma_{11}^{-1}\Sigma_{12}(\Sigma/\Sigma_{11})^{-1}\Sigma_{21}\Sigma_{11}^{-1} \\
\left(A - BC^{-1}D\right)^{-1} &= A^{-1} + A^{-1}B\left(C - DA^{-1}B\right)^{-1}DA^{-1}
\end{aligned}
$$

Factorisation of Multivariate Gaussians

Consider the joint distribution over the variable

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

where the joint distribution is Gaussian $p(x) = \mathcal{N}(x; \mu, \Sigma)$ with

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} \Sigma_1 & \Sigma_{12} \\ \Sigma_{12}^\top & \Sigma_2 \end{pmatrix}$$

Factorisation of Multivariate Gaussians

Find the following

1. Conditionals

   (a) $p(x_1|x_2)$
   (b) $p(x_2|x_1)$

2. Marginals

   (a) $p(x_1)$
   (b) $p(x_2)$

Factorisation of Multivariate Gaussians

Using the partitioned inverse equations, we rearrange

$$p(x_1, x_2) \quad \propto \quad \exp\left(-\frac{1}{2} \left(\begin{array}{c} x_1 - \mu_1 \\ x_2 - \mu_2 \end{array} \right)^\top \left(\begin{array}{cc} \Sigma_1 & \Sigma_{12} \\ \Sigma_{12}^\top & \Sigma_2 \end{array} \right)^{-1} \left(\begin{array}{c} x_1 - \mu_1 \\ x_2 - \mu_2 \end{array} \right) \right)$$

bring the expression in form of $p(x_1)p(x_2|x_1)$ (or $p(x_2)p(x_1|x_2)$) where the marginal and conditional can be easily identified. (See also Bishop, section 2.3.)

Factorisation of Multivariate Gaussians

We have the two decompositions

$$
\begin{aligned}
\Sigma^{-1} &= \left(\begin{array}{cc} \Sigma_1 & \Sigma_{12} \\ \Sigma_{12}^\top & \Sigma_2 \end{array}\right)^{-1} \\
&= \left(\begin{array}{cc} \left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1} & -\left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1}\Sigma_{12}\Sigma_2^{-1} \\ -\Sigma_2^{-1}\Sigma_{12}^\top\left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1} & \Sigma_2^{-1} + \Sigma_2^{-1}\Sigma_{12}^\top\left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1}\Sigma_{12}\Sigma_2^{-1} \end{array}\right) \\
&= \left(\begin{array}{cc} \Sigma_1^{-1} + \Sigma_1^{-1}\Sigma_{12}\left(\Sigma_2 - \Sigma_{12}^\top\Sigma_1^{-1}\Sigma_{12}\right)^{-1}\Sigma_{12}^\top\Sigma_1^{-1} & -\Sigma_1^{-1}\Sigma_{12}\left(\Sigma_2 - \Sigma_{12}^\top\Sigma_1^{-1}\Sigma_{12}\right)^{-1} \\ -\left(\Sigma_2 - \Sigma_{12}^\top\Sigma_1^{-1}\Sigma_{12}\right)^{-1}\Sigma_{12}^\top\Sigma_1^{-1} & \left(\Sigma_2 - \Sigma_{12}^\top\Sigma_1^{-1}\Sigma_{12}\right)^{-1} \end{array}\right)
\end{aligned}
$$

We let $s_i = x_i - \mu_i$ and use the first decomposition.

$$
p(s_1, s_2) \propto \exp\left(-\frac{1}{2}\left(\begin{array}{c} s_1 \\ s_2 \end{array}\right)^\top \left(\begin{array}{cc} \Sigma_1 & \Sigma_{12} \\ \Sigma_{12}^\top & \Sigma_2 \end{array}\right)^{-1} \left(\begin{array}{c} s_1 \\ s_2 \end{array}\right)\right)
$$

$$= \quad \exp\left(-\frac{1}{2}\left(\begin{array}{c} s_1 \\ s_2 \end{array}\right)^\top \left(\begin{array}{cc} \left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1} & -\left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1}\Sigma_{12}\Sigma_2^{-1} \\ -\Sigma_2^{-1}\Sigma_{12}^\top\left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1} & \Sigma_2^{-1} + \Sigma_2^{-1}\Sigma_{12}^\top\left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1}\Sigma_{12}\Sigma_2^{-1} \end{array}\right)\left(\begin{array}{c} s_1 \\ s_2 \end{array}\right.$$

$$\begin{aligned}
= \quad & \exp(-\frac{1}{2}s_1^\top \left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1} s_1 \\
& s_2^\top \Sigma_2^{-1}\Sigma_{12}^\top \left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1} s_1 \\
& -\frac{1}{2}s_2^\top \Sigma_2^{-1}\Sigma_{12}^\top \left(\Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top\right)^{-1} \Sigma_{12}\Sigma_2^{-1} s_2 \\
& -\frac{1}{2}s_2^\top \Sigma_2^{-1} s_2) \\
\propto \quad & \mathcal{N}(s_1; \Sigma_{12}\Sigma_2^{-1}s_2, \Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top)\mathcal{N}(s_2; 0, \Sigma_2) \\
= \quad & \mathcal{N}(x_1; \mu_1 + \Sigma_{12}\Sigma_2^{-1}(x_2 - \mu_2), \Sigma_1 - \Sigma_{12}\Sigma_2^{-1}\Sigma_{12}^\top)\mathcal{N}(x_2; \mu_2, \Sigma_2)
\end{aligned}$$

This leads to a factorisation of form $p(x_2)p(x_1|x_2)$. The second decomposition will lead to the other factorisation $p(x_1)p(x_2|x_1)$.

Approximate Inference

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

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

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: $KL(q||p)$

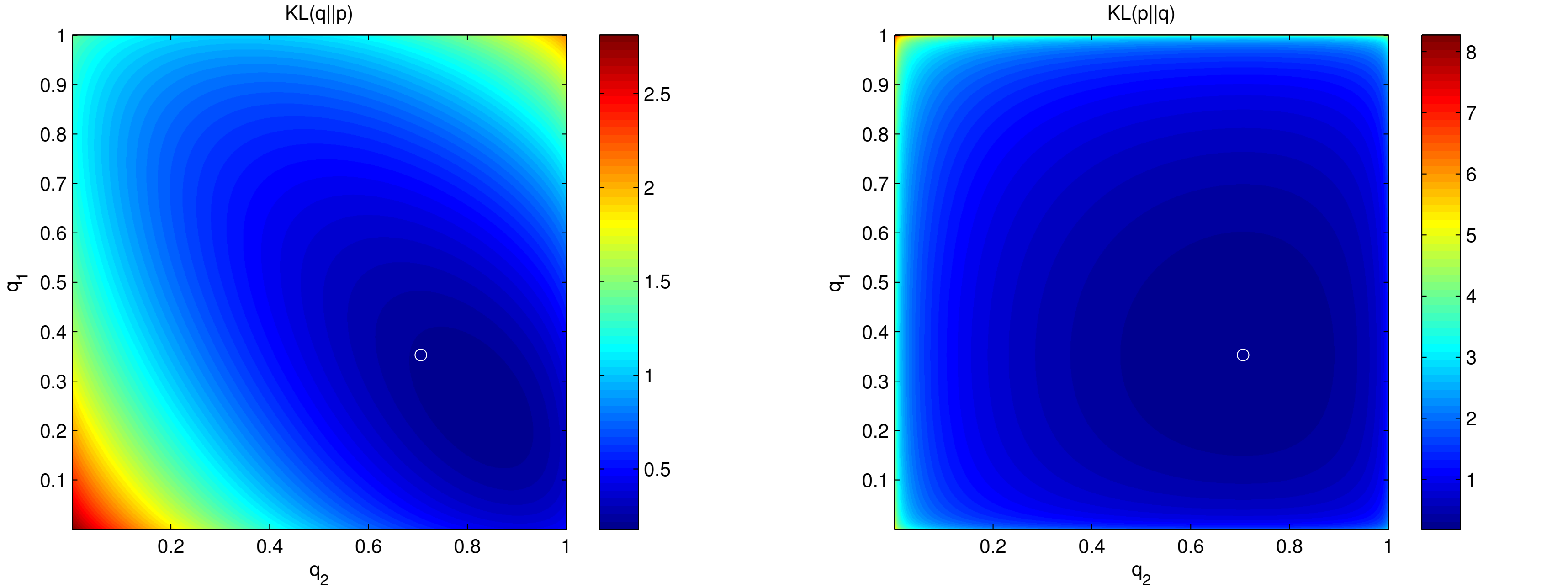
$$\begin{aligned}\frac{\partial KL(q||p)}{\partial q_1} &= (-\log(1-q_1) + \log \pi_{0,0} + \log q_1 - \log \pi_{1,0}) \\ & \quad q_2\,(-\log \pi_{0,0} + \log \pi_{1,0} + \log \pi_{0,1} - \log \pi_{1,1})\end{aligned}$$

The “other” one: $KL(q||p)$

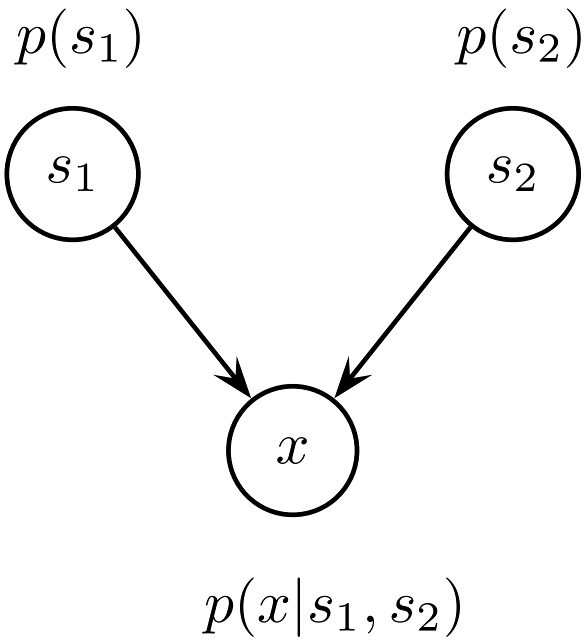
$$
\begin{aligned}
\mathcal{Q}_1 &= \begin{pmatrix} 1-q_1 \\ q_1 \end{pmatrix} = \frac{1}{Z_1} \begin{pmatrix} \pi_{0,0}^{(1-q_2)} \pi_{0,1}^{q_2} \\ \pi_{1,0}^{(1-q_2)} \pi_{1,1}^{q_2} \end{pmatrix} \\
&\propto \begin{pmatrix} \exp((1-q_2)\log\pi_{0,0} + q_2 \log \pi_{0,1}) \\ \exp((1-q_2)\log\pi_{1,0}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

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

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



Toy Model : "One sample source separation (OSSS)"



This graph encodes the joint: $p(x, s_1, s_2) = p(x|s_1, s_2)p(s_1)p(s_2)$

$$
\begin{aligned}
s_1 &\sim p(s_1) = \mathcal{N}(s_1; \mu_1, P_1) \\
s_2 &\sim p(s_2) = \mathcal{N}(s_2; \mu_2, P_2) \\
x|s_1, s_2 &\sim p(x|s_1, s_2) = \mathcal{N}(x; s_1 + s_2, R)
\end{aligned}
$$

The Gaussian Distribution

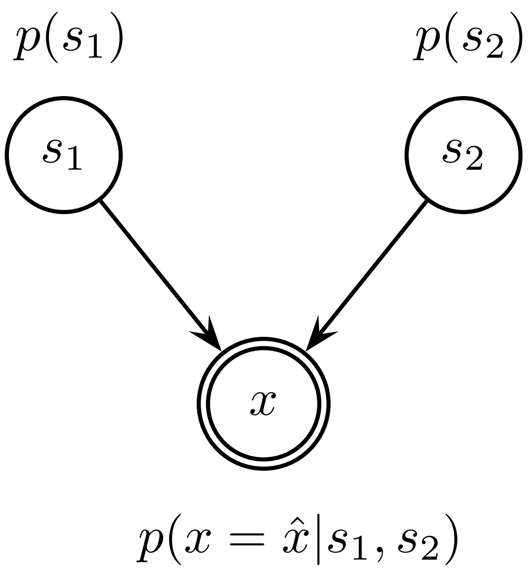
μ is the mean and P is the covariance:

$$
\begin{aligned}
\mathcal{N}(s;\mu,P) &= |2\pi P|^{-1/2} \exp\left(-\frac{1}{2}(s-\mu)^T P^{-1}(s-\mu)\right) \\
&= \exp\left(-\frac{1}{2}s^T P^{-1} s + \mu^T P^{-1} s - \frac{1}{2}\mu^T P^{-1}\mu - \frac{1}{2}|2\pi P|\right) \\
\log \mathcal{N}(s;\mu,P) &= -\frac{1}{2}s^T P^{-1} s + \mu^T P^{-1} s + \text{ const} \\
&= -\frac{1}{2}\mathbf{Tr}\, P^{-1} s s^T + \mu^T P^{-1} s + \text{ const} \\
&=^+ -\frac{1}{2}\mathbf{Tr}\, P^{-1} s s^T + \mu^T P^{-1} s
\end{aligned}
$$

Notation: $\log f(x) =^+ g(x) \iff f(x) \propto \exp(g(x)) \iff \exists c \in \mathbb{R} : f(x) = c \exp(g(x))$

OSSS example

Suppose, we observe $x = \hat{x}$.



- By Bayes' theorem, the posterior is given by:

$$\mathcal{P} \equiv p(s_1, s_2|x = \hat{x}) = \frac{1}{Z_{\hat{x}}} p(x = \hat{x}|s_1, s_2)p(s_1)p(s_2) \equiv \frac{1}{Z_{\hat{x}}} \phi(s_1, s_2)$$

- The function $\phi(s_1, s_2)$ is proportional to the exact posterior. ($Z_{\hat{x}} \equiv p(x = \hat{x})$)

OSSS example, cont.

$$\begin{aligned}
\log p(s_1) &= \mu_1^T P_1^{-1} s_1 - \frac{1}{2} s_1^T P_1^{-1} s_1 + \text{const} \\
\log p(s_2) &= \mu_2^T P_2^{-1} s_2 - \frac{1}{2} s_2^T P_2^{-1} s_2 + \text{const} \\
\log p(x|s_1, s_2) &= \hat{x}^T R^{-1}(s_1 + s_2) - \frac{1}{2}(s_1 + s_2)^T R^{-1}(s_1 + s_2) + \text{const}
\end{aligned}$$

$$\begin{aligned}
\log \phi(s_1, s_2) &= \log p(x = \hat{x}|s_1, s_2) \; + \log p(s_1) \; + \log p(s_2) \\
&=^+ \left(\mu_1^T P_1^{-1} + \hat{x}^T R^{-1}\right) s_1 + \left(\mu_2^T P_2^{-1} + \hat{x}^T R^{-1}\right) s_2 \\
&\quad -\frac{1}{2} \mathbf{Tr}\left(P_1^{-1} + R^{-1}\right) s_1 s_1^T - \underbrace{s_1^T R^{-1} s_2}_{(*)} - \frac{1}{2} \mathbf{Tr}\left(P_2^{-1} + R^{-1}\right) s_2 s_2^T
\end{aligned}$$

- The (*) term is the cross correlation term that makes s_1 and s_2 a-posteriori dependent.

OSSS example, cont.

Completing the square

$$\log \phi(s_1, s_2) \quad =^+ \quad \begin{pmatrix} P_1^{-1}\mu_1 + R^{-1}\hat{x} \\ P_2^{-1}\mu_2 + R^{-1}\hat{x} \end{pmatrix}^\top \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$

$$-\frac{1}{2}\begin{pmatrix} s_1 \\ s_2 \end{pmatrix}^\top \begin{pmatrix} P_1^{-1} + R^{-1} & R^{-1} \\ R^{-1} & P_2^{-1} + R^{-1} \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$

Remember: $\log \mathcal{N}(s; m, \Sigma) \quad =^+ \quad (\Sigma^{-1}m)^\top s - \frac{1}{2}s^\top \Sigma^{-1} s$

$$\Sigma \quad = \quad \begin{pmatrix} P_1^{-1} + R^{-1} & R^{-1} \\ R^{-1} & P_2^{-1} + R^{-1} \end{pmatrix}^{-1} \qquad m = \Sigma \quad \begin{pmatrix} P_1^{-1}\mu_1 + R^{-1}\hat{x} \\ P_2^{-1}\mu_2 + R^{-1}\hat{x} \end{pmatrix}$$

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

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

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)

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

OSSS example, cont.

Let the approximating distribution be factorized as

$$\mathcal{Q} = q(s_1)q(s_2)$$

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

The m_i and S_j are the *variational* parameters to be optimized to minimize

$$KL(\mathcal{Q}||\mathcal{P}) = \langle \log \mathcal{Q} \rangle_{\mathcal{Q}} - \left\langle \log \underbrace{\frac{1}{Z_x}\phi(s_1, s_2)}_{=\mathcal{P}} \right\rangle_{\mathcal{Q}} \tag{4}$$

The form of the mean field solution

$$
\begin{aligned}
0 &\leq \langle \log q(s_1)q(s_2) \rangle_{q(s_1)q(s_2)} + \log Z_x - \langle \log \phi(s_1, s_2) \rangle_{q(s_1)q(s_2)} \\
\log Z_x &\geq \langle \log \phi(s_1, s_2) \rangle_{q(s_1)q(s_2)} - \langle \log q(s_1)q(s_2) \rangle_{q(s_1)q(s_2)} \\
&\equiv -F(p; q) + H(q) \qquad (5)
\end{aligned}
$$

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

$$\begin{aligned}\Lambda &= \int ds_1 q(s_1)\log q(s_1) + \int ds_2 q(s_2)\log q(s_2) + \log Z_x - \int ds_1 ds_2 q(s_1)q(s_2)\log\phi(s_1,s_2) \\ &\quad +\lambda_1(1-\int ds_1 q(s_1)) + \lambda_2(1-\int ds_2 q(s_2)) \qquad (6)\end{aligned}$$

- 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 $q(s_1)$,

$$\begin{aligned}\log q(s_1) &= \lambda_1 - 1 + \langle \log\phi(s_1,s_2)\rangle_{q(s_2)} \\ q(s_1) &= \exp(\lambda_1 - 1)\exp(\langle \log\phi(s_1,s_2)\rangle_{q(s_2)}) \qquad (7)\end{aligned}$$

- Use the fact that

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

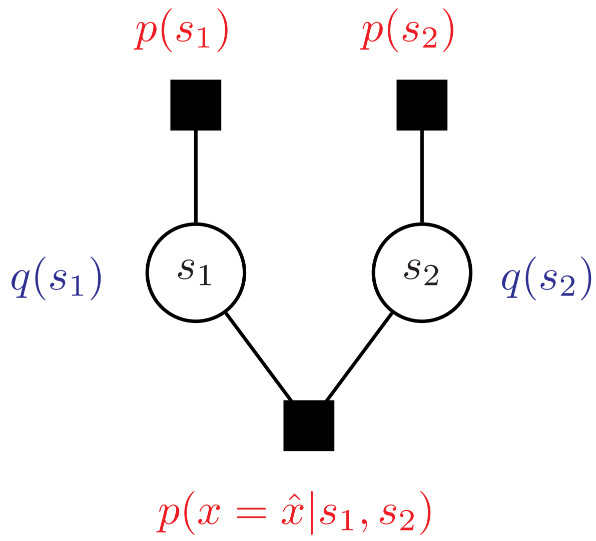
The form of the solution

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

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

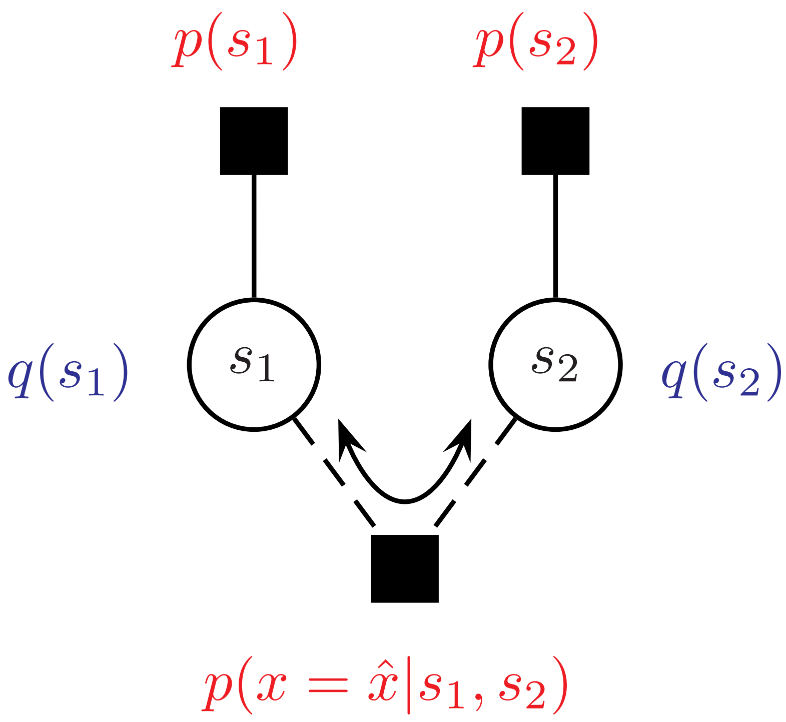
Note the nice symmetry

OSSS: Factor Graph



- A graphical representation of the inference problem
  - **Factor nodes**: Black squares. Factor potentials (local functions) defining the posterior $\mathcal{P}$.
  - **Variable nodes**: Circles. Think of them as "factors" of the approximating distribution $\mathcal{Q}$. (Caution – non standard interpretation!)
  - **Edges**: denote membership. A variable is connected to a factor if it is a variable of the local function.

Fixed Point Iteration for OSSS



$$\log q(s_1) \quad \leftarrow \quad \log p(s_1) + \langle \log p(x = \hat{x}|s_1, s_2) \rangle_{q(s_2)}$$

$$\log q(s_2) \quad \leftarrow \quad \log p(s_2) + \langle \log p(x = \hat{x}|s_1, s_2) \rangle_{q(s_1)}$$

Fixed Point Iteration for the Gaussian Case

$$\log q(s_1) \leftarrow -\frac{1}{2}\mathbf{Tr}\left(P_1^{-1} + R^{-1}\right) s_1 s_1^\top - s_1^\top R^{-1} \underbrace{\langle s_2 \rangle_{q(s_2)}}_{=m_2} + \left(\mu_1^\top P_1^{-1} + \hat{x}^\top R^{-1}\right) s_1$$

$$\log q(s_2) \leftarrow -\underbrace{\langle s_1 \rangle_{q(s_1)}^\top}_{=m_1^\top} R^{-1} s_2 - \frac{1}{2}\mathbf{Tr}\left(P_2^{-1} + R^{-1}\right) s_2 s_2^\top + \left(\mu_2^\top P_2^{-1} + \hat{x}^\top R^{-1}\right) s_2$$

Remember $q(s) = \mathcal{N}(s; m, S)$

$$\log q(s) =^+ -\frac{1}{2}\mathbf{Tr}\, K s s^\top + h^\top s$$

$$\Downarrow$$

$$S = K^{-1} \qquad m = K^{-1} h$$

Fixed Point Equations for the Gaussian Case

- Covariances are obtained directly

$$S_1 = \left(P_1^{-1} + R^{-1}\right)^{-1} \qquad S_2 = \left(P_2^{-1} + R^{-1}\right)^{-1}$$

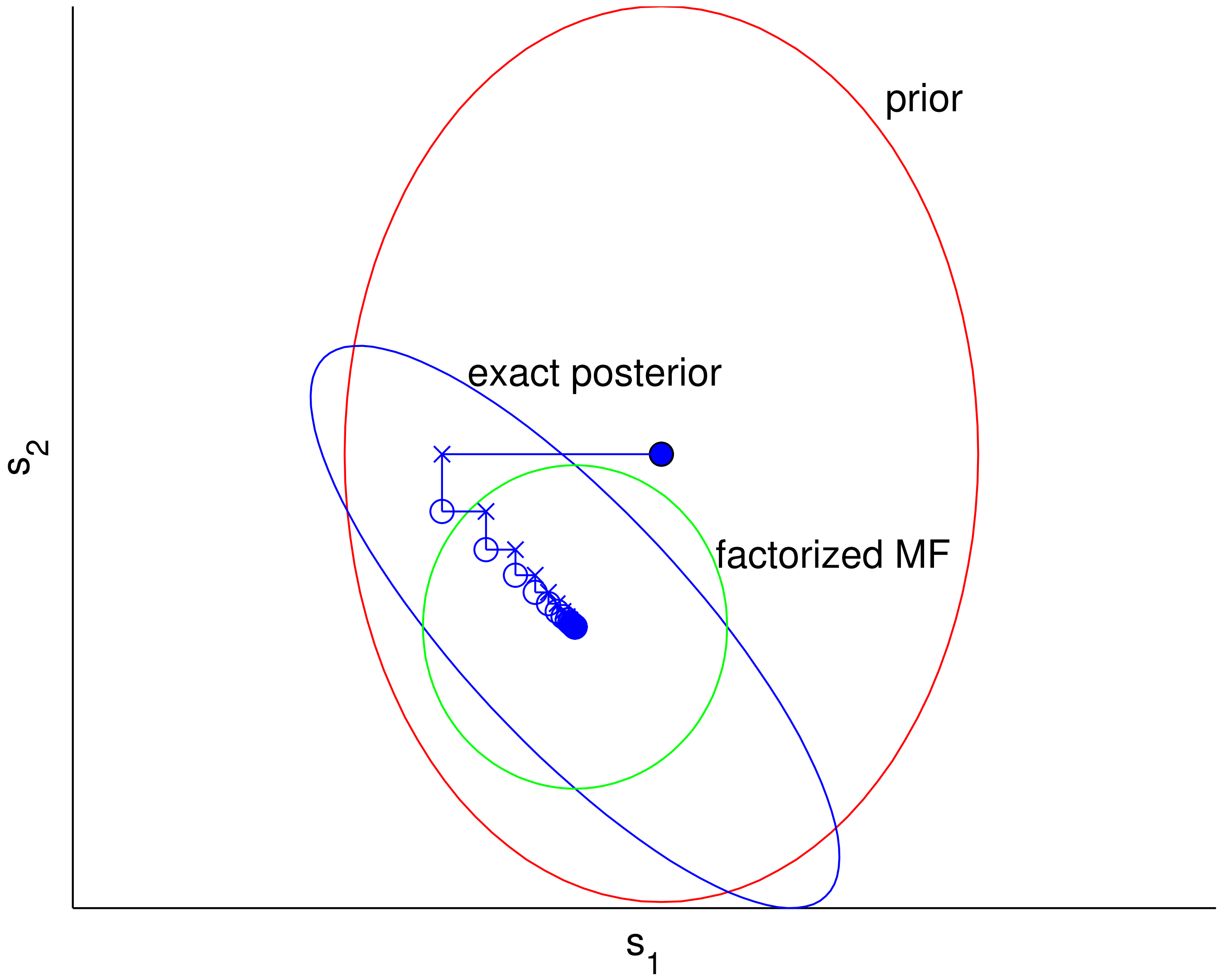
- To compute the means, we should iterate:

$$\begin{aligned} m_1 &= S_1 \left(P_1^{-1}\mu_1 + R^{-1}(\hat{x} - m_2)\right) \\ m_2 &= S_2 \left(P_2^{-1}\mu_2 + R^{-1}(\hat{x} - m_1)\right) \end{aligned}$$

- Intuitive algorithm:
  - Substract from the observation $\hat{x}$ the prediction of the other factors of $\mathcal{Q}$.
  - Compute a fit to this residual (e.g. “fit” m_2 to $\hat{x} - m_1$).

- Equivalent to Gauss-Seidel, an iterative method for solving linear systems of equations.

OSSS example, cont.



Direct Link to Expectation-Maximisation (EM) [?]

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

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

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

$$\tilde{m} \quad = \quad \underset{\xi}{\operatorname{argmin}} \, KL(\delta(s_2 - \xi) || q(s_2))$$

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.

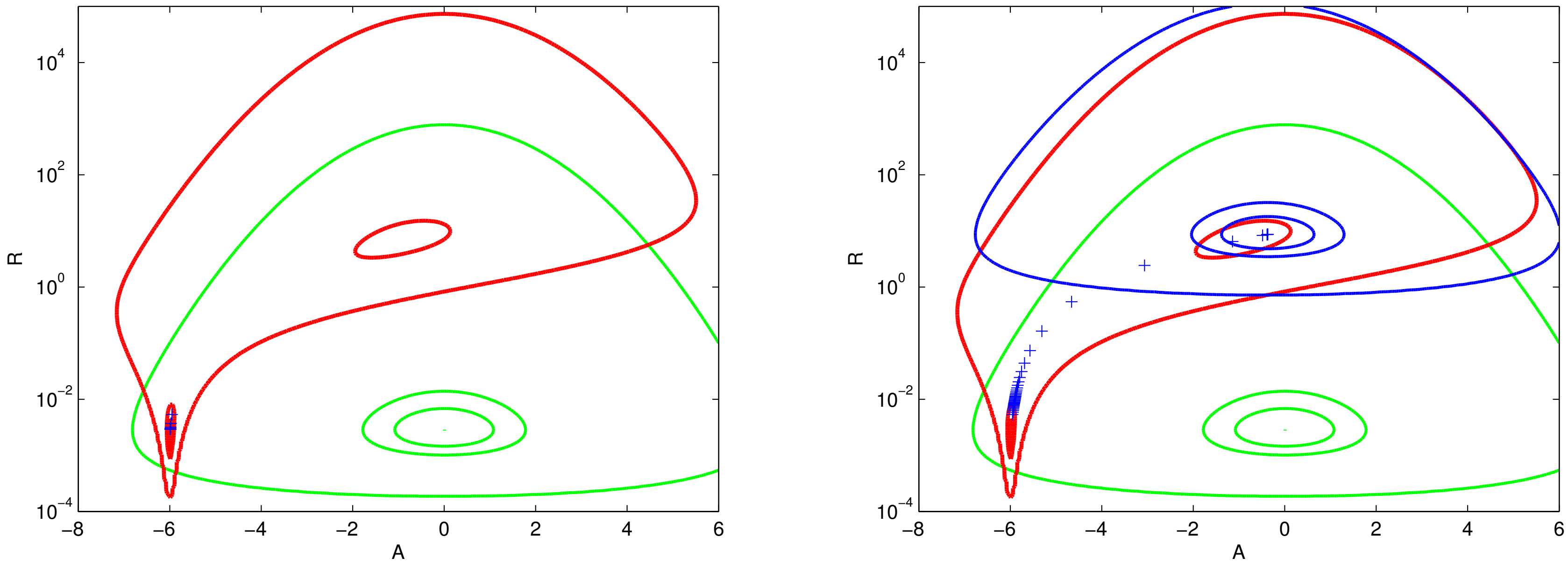
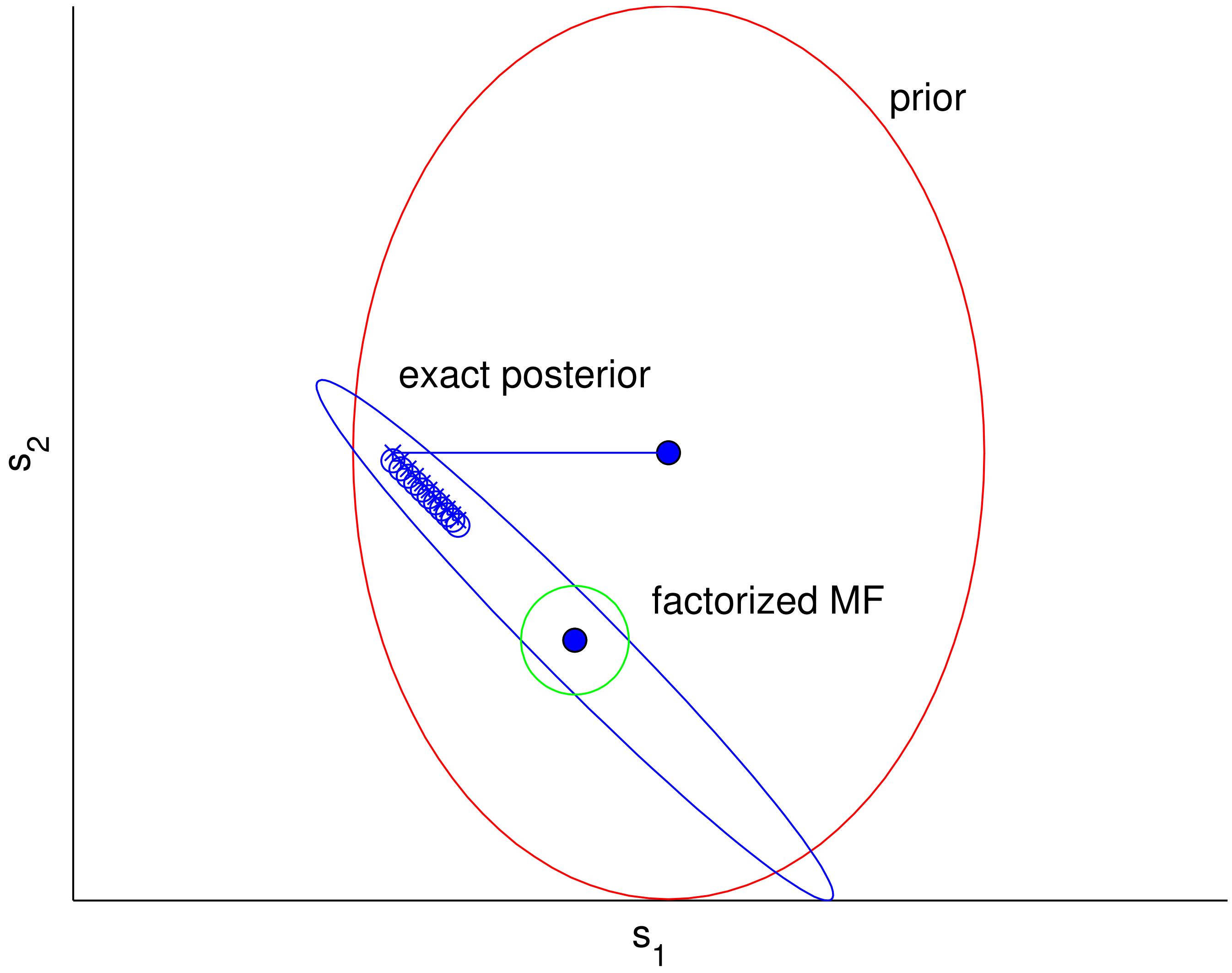


Figure 1: Left, ICM, Right VB. EM is similar to ICM in this AR(1) example.

Convergence Issues

OSSS example, Slow Convergence



Annealing, Bridging, Relaxation, Tempering

Main idea:

- If the original target $\mathcal{P}$ is too complex, relax it.
- First solve a simple version $\mathcal{P}_{\tau_1}$. Call the solution m_{τ_1}
- Make the problem little bit harder $\mathcal{P}_{\tau_1} \to \mathcal{P}_{\tau_2}$, and improve the solution $m_{\tau_1} \to m_{\tau_2}$.
- While $\mathcal{P}_{\tau_1} \to \mathcal{P}_{\tau_2}, \dots, \to \mathcal{P}_T = \mathcal{P}$, we hope to get better and better solutions.

The sequence $\tau_1, \tau_2, \dots, \tau_T$ is called annealing schedule if

$$\mathcal{P}_{\tau_i} \quad \propto \quad \mathcal{P}^{\tau_i}$$

OSSS example: Annealing, Bridging, ...

- Remember the cross term $(*)$ of the posterior:

$$\cdots - \underbrace{s_1^\top R^{-1} s_2}_{(*)} \cdots$$

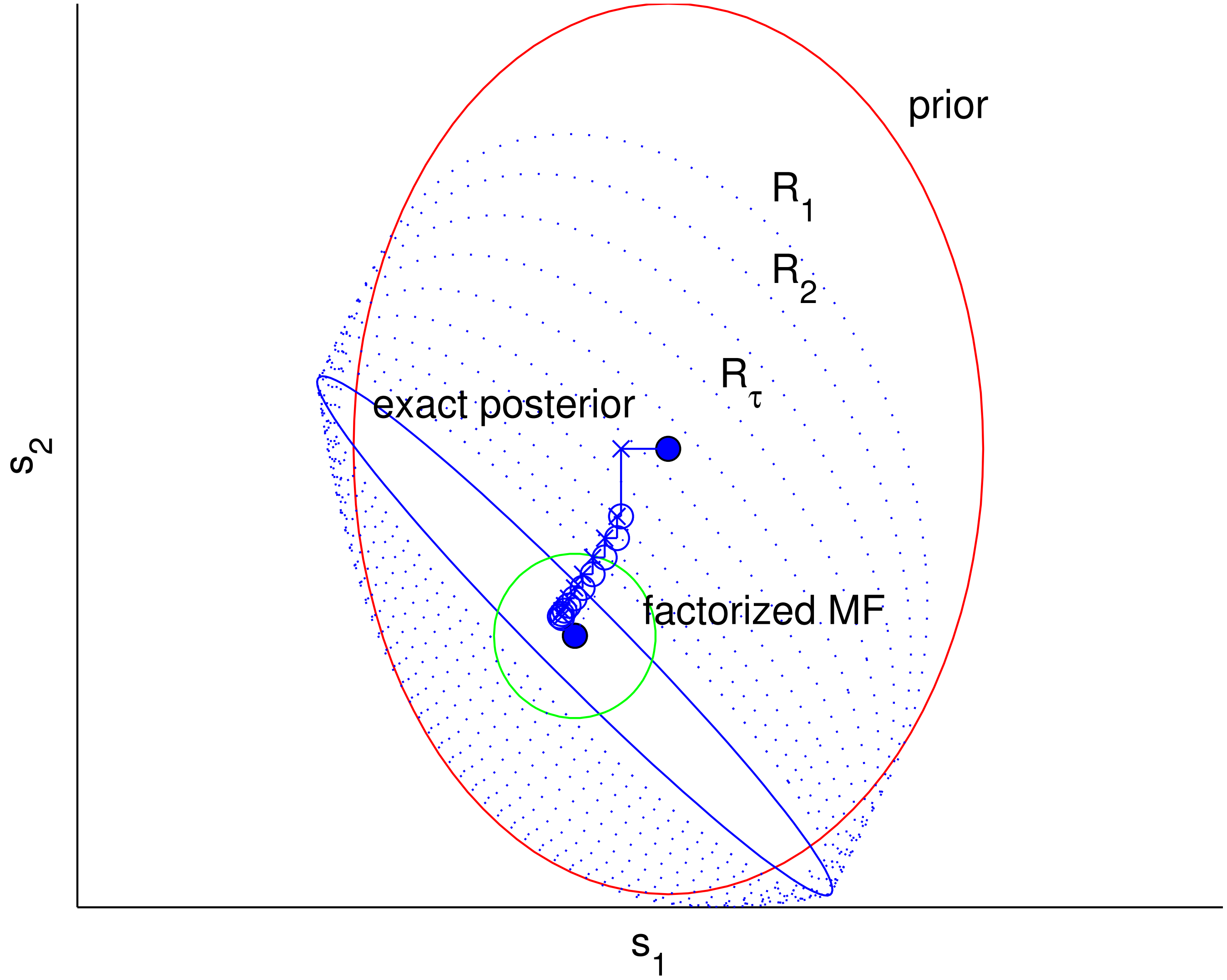
- When the noise variance is low, the coupling is strong.

- If we choose a decreasing sequence of noise covariances

$$R_{\tau_1} > R_{\tau_2} > \cdots > R_{\tau_T} = R$$

  we increase correlations gradually.

OSSS example: Annealing, Bridging, ...



Fixed Point Iterations

Let θ denote the parameter vector of $\mathcal{Q}$.

- Given the fixed point equation F and an initial parameter $\theta^{(0)}$, the inference algorithm is simply

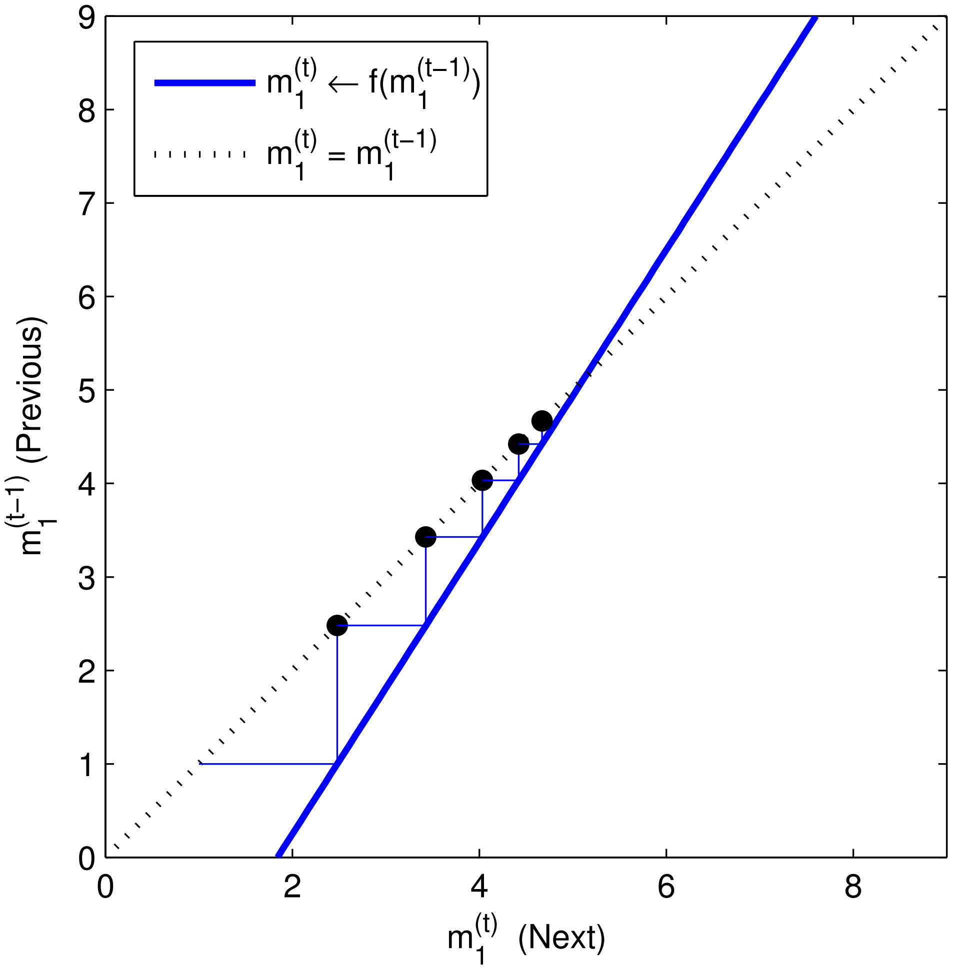
$$\theta^{(t+1)} \quad \leftarrow \quad F(\theta^{(t)})$$

For OSSS $\theta = (m_1, m_2)^\top$ (S_1, S_2 were constant, so we exclude them). The update equations were

$$\begin{aligned} m_1^{(t+1)} \quad &\leftarrow \quad F_1(m_2^{(t)}) \\ m_2^{(t+1)} \quad &\leftarrow \quad F_2(m_1^{(t+1)}) \end{aligned}$$

This is a deterministic dynamical system in the parameter space.

OSSS: Fixed Point iteration for m_1



Derivation of Variational Bayes

Derivation of a Variational Bayes algorithm

1. Write down the log of the full joint (unnormalised) posterior $\log \phi(v_1, \dots, v_N)$

2. Decide the individual factors of the approximating distribution, i.e., find a set of mutually exclusive clusters

$$\{v_1, \dots, v_N\} = \bigcup_\alpha \mathcal{C}_\alpha$$

   (Mean field is $\{v_1, \dots, v_N\} = \{v_1\} \cup \{v_2\} \cup \dots \cup \{v_N\}$)

3. Draw the factor graph and assign each term of $\log \phi$ to individual factors

4. Derive the factors of Q_α the approximating distribution $Q = \prod_\alpha Q_\alpha$ as a function of the sufficient statistics of $\{Q_{-\alpha}\}$

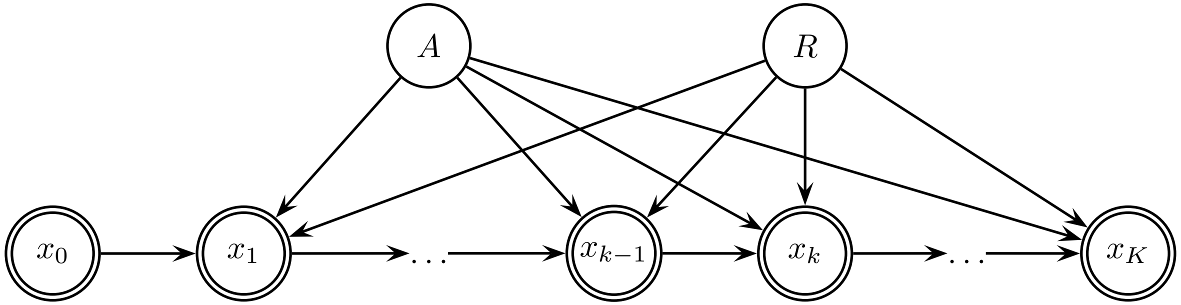
Variational Bayes

5. Initialise the (variational parameters of the) factors of Q to reasonable values

6. Visit each factor of Q_α and update it as a function of $\{Q_{-\alpha}\}$ until convergence

$$Q_\alpha \quad \propto \quad \exp\left(\langle \log \phi \rangle_{Q_{-\alpha}}\right)$$

AR(1) Model



$$
\begin{aligned}
A &\sim \mathcal{N}(A; 0, P) \\
R &\sim \mathcal{IG}(R; \nu, \nu/\beta) \\
x_k | x_{k-1}, A, R &\sim \mathcal{N}(x_k; A x_{k-1}, R) \\
x_0 &= 1 \qquad x_1 = -6
\end{aligned}
$$

Caution: (Wikipedia compatible definition of $\mathcal{IG}$)

$$
\mathcal{IG}(R; a, b) = \exp\left(-(a+1)\log R - \frac{b}{R} - \log\Gamma(a) + a\log b \right)
$$

Step 1: Write down the log of the full joint (unnormalised) posterior $\log \phi(A, R, x_1 = \hat{x}_1 | x_0 = \hat{x}_0)$

$$
\begin{aligned}
\phi &= p(A, R, x_1 = \hat{x}_1 | x_0 = \hat{x}_0) \propto p(x_1 | x_0, A, R) p(A) p(R) \\
&= \mathcal{N}(x_1; A x_0, R) \mathcal{N}(A; 0, P) \mathcal{IG}(R; \nu, \nu/\beta) \\
&\propto \exp\left(-\frac{1}{2}\frac{x_1^2}{R} + x_0 x_1 \frac{A}{R} - \frac{1}{2}\frac{x_0^2 A^2}{R} - \frac{1}{2}\log 2\pi R \right) \\
&\quad \exp\left(-\frac{1}{2}\frac{A^2}{P} - \frac{1}{2}\log |2\pi P| \right) \\
&\quad \exp\left(-(\nu + 1)\log R - \frac{\nu}{\beta}\frac{1}{R} - \log \Gamma(\nu) + \nu \log(\nu/\beta) \right)
\end{aligned}
$$

Step 2. Choose the individual factors of Q

$$
\begin{aligned}
Q &= q(A)q(R) \\
q(A) &= \mathcal{N}(A; m, \Sigma) \\
q(R) &= \mathcal{IG}(R; a, b)
\end{aligned}
$$

Clusters

$$
\mathcal{C} = \{A\} \cup \{R\}
$$

Step 2. Choose the individual factors of Q

Sufficient statistics and modes

- $q(A) = \mathcal{N}(A; m, \Sigma)$

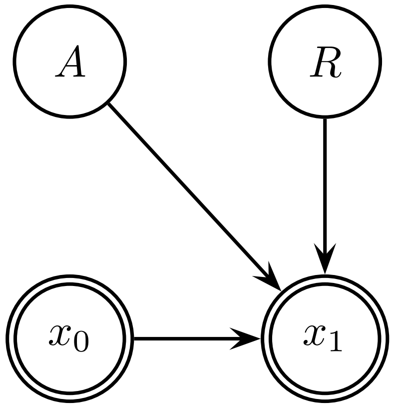
$$\langle A \rangle = m \qquad \left\langle A^2 \right\rangle = \Sigma + m^2 \qquad A^* = m$$

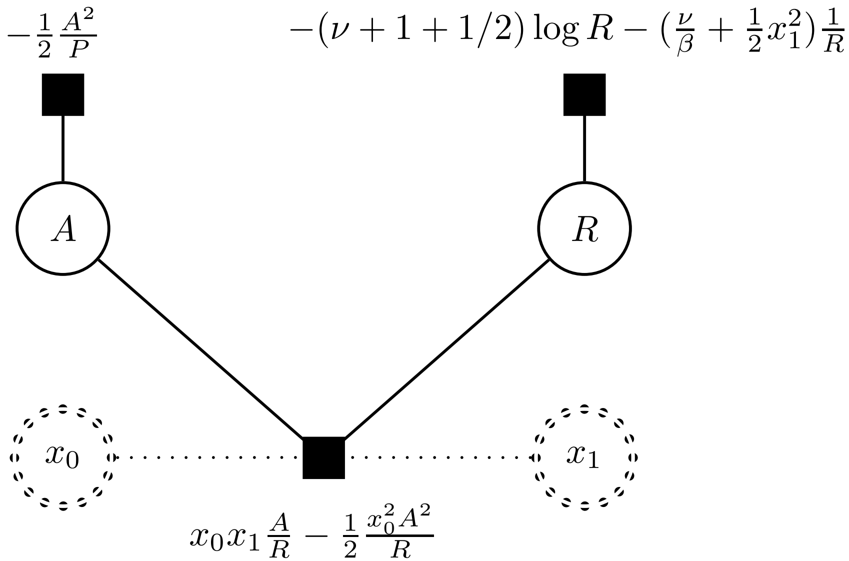
- $q(R) = \mathcal{IG}(R; a, b)$

$$\langle 1/R \rangle = a/b \qquad \langle \log R \rangle = \log(b) - \Psi(a)$$

$$R^* = b/(a+1)$$

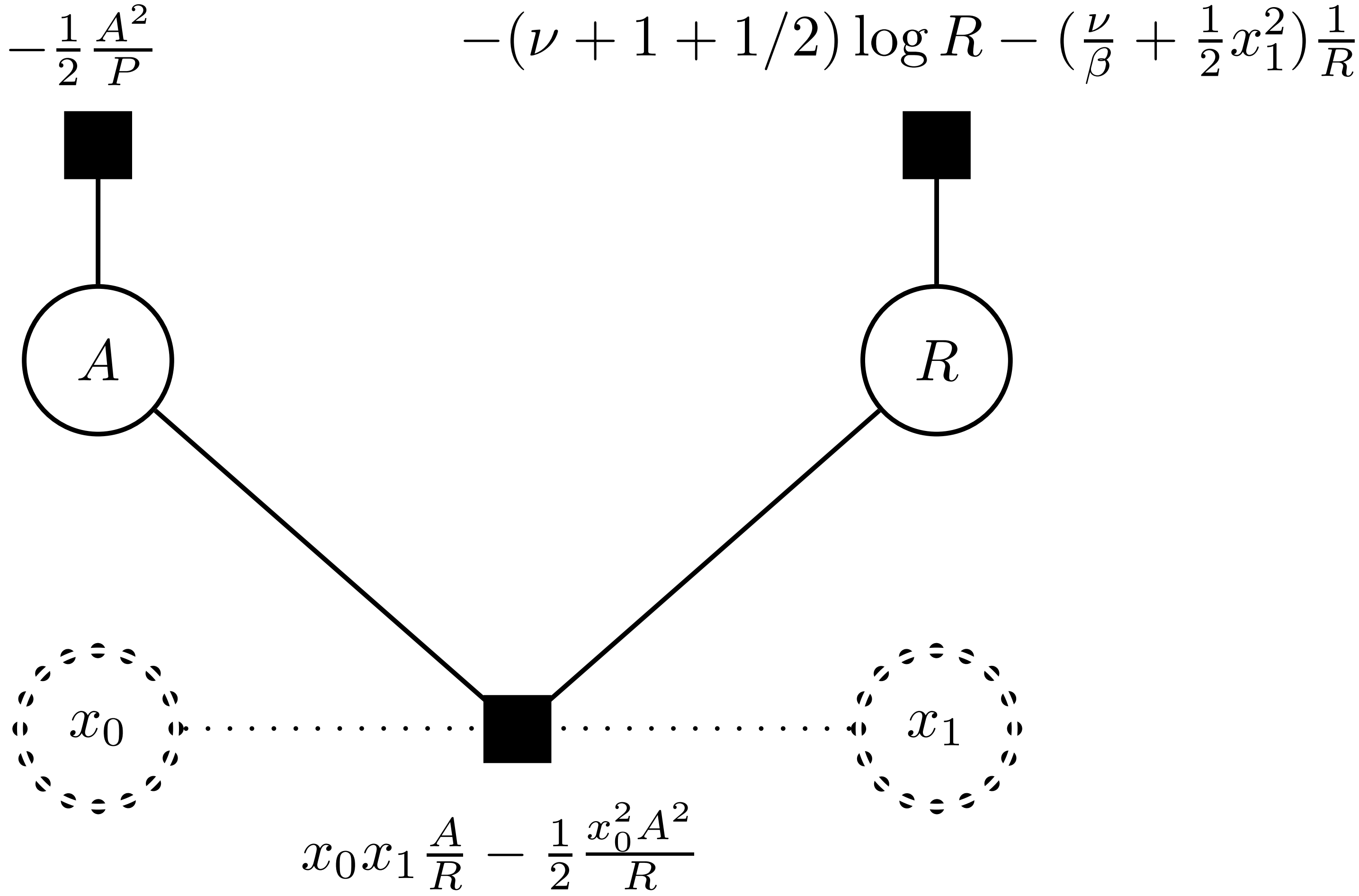
Step 3. Draw the factor graph and assign each term of $\log \phi$ to individual factors





$$
\begin{aligned}
\log \phi &= -\frac{1}{2}\frac{x_1^2}{R} + x_0 x_1 \frac{A}{R} - \frac{1}{2}\frac{x_0^2 A^2}{R} - \frac{1}{2}\log 2\pi R - \frac{1}{2}\frac{A^2}{P} - \frac{1}{2}\log|2\pi P| \\
&\quad -(\nu+1)\log R - \frac{\nu}{\beta}\frac{1}{R} - \log\Gamma(\nu) + \nu\log(\nu/\beta) \\
&=^{+} -\frac{1}{2}\frac{x_1^2}{R} + x_0 x_1 \frac{A}{R} - \frac{1}{2}\frac{x_0^2 A^2}{R} - \frac{1}{2}\log R - \frac{1}{2}\frac{A^2}{P} - (\nu+1)\log R - \frac{\nu}{\beta}\frac{1}{R}
\end{aligned}
$$

Step 4. Derive the factors of Q the approximating distribution as a function of the sufficient statistics of $\{Q_{-\alpha}\}$



Step 4. Derive the factors Q_α

- $q(A) = \mathcal{N}(A; m, \Sigma)$

$$
\begin{aligned}
q(A) &\propto \exp(\langle \log \phi(A, R) \rangle_{q(R)}) \\
&= \exp\left(-\frac{1}{2}\frac{A^2}{P} + \left\langle x_0 x_1 \frac{1}{R} A - \frac{1}{2} x_0^2 \frac{1}{R} A^2 \right\rangle_{q(R)} \right) \\
&= \exp\left(-\frac{1}{2}\left(\frac{1}{P} + x_0^2 \left\langle \frac{1}{R} \right\rangle_{q(R)} \right) A^2 + x_0 x_1 \left\langle \frac{1}{R} \right\rangle_{q(R)} A \right)
\end{aligned}
$$

$$
\begin{aligned}
\Sigma &= \left(\frac{1}{P} + x_0^2 \left\langle \frac{1}{R} \right\rangle_{q(R)} \right)^{-1} = \left(\frac{1}{P} + x_0^2 \frac{a}{b} \right)^{-1} \\
m &= \Sigma x_0 x_1 \left\langle \frac{1}{R} \right\rangle_{q(R)} = \Sigma x_0 x_1 \frac{a}{b}
\end{aligned}
$$

Step 4. Derive the factors of Q

- $q(R) = \mathcal{IG}(R; a, b)$

$$
\begin{aligned}
q(R) &\propto \exp(\langle \log \phi(A, R) \rangle_{q(A)}) \\
&= \exp(-(\nu + 1 + 1/2) \log R - (\frac{\nu}{\beta} + \frac{1}{2} x_1^2 + \left\langle -x_0 x_1 A + \frac{1}{2} x_0^2 A^2 \right\rangle_{q(A)}) \frac{1}{R}) \\
&= \exp(-(\nu + 1 + 1/2) \log R - (\frac{\nu}{\beta} + \frac{1}{2} x_1^2 - x_0 x_1 \langle A \rangle_{q(A)} + \frac{1}{2} x_0^2 \langle A^2 \rangle_{q(A)}) \frac{1}{R})
\end{aligned}
$$

$$
\begin{aligned}
a &= \nu + 1/2 \\
b &= \frac{\nu}{\beta} + \frac{1}{2} x_1^2 - x_0 x_1 \langle A \rangle_{q(A)} + \frac{1}{2} x_0^2 \langle A^2 \rangle_{q(A)} \\
&= \frac{\nu}{\beta} + \frac{1}{2} x_1^2 - x_0 x_1 m + \frac{1}{2} x_0^2 (m^2 + \Sigma)
\end{aligned}
$$

Variational Bayes

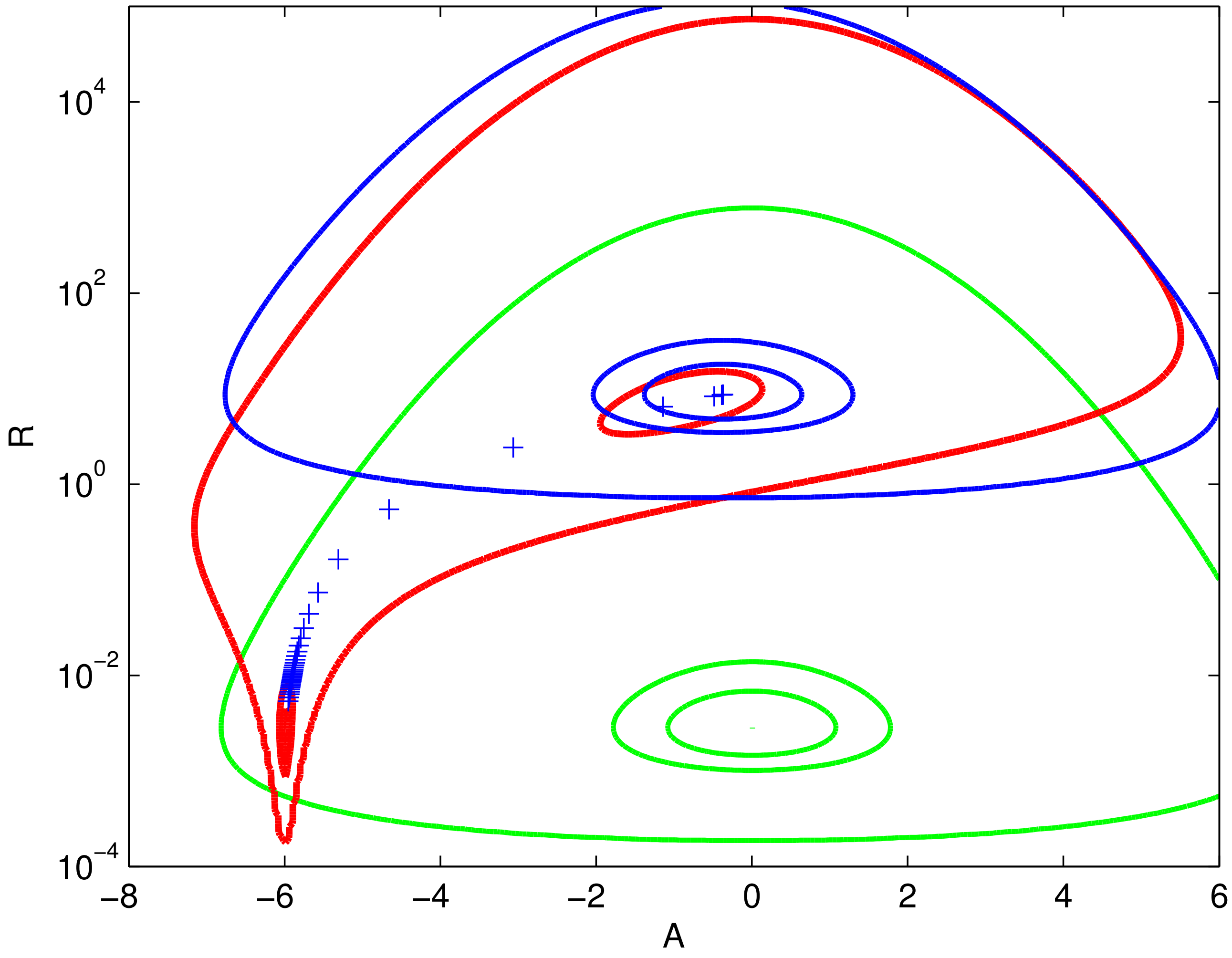
For $\tau = 1, 2, \dots$

$$
\begin{aligned}
q(A)^{(\tau)} &= \exp(\langle \log \phi(A, R) \rangle_{q(R)^{(\tau-1)}}) \\
q(R)^{(\tau)} &= \exp(\langle \log \phi(A, R) \rangle_{q(A)^{(\tau)}})
\end{aligned}
$$

Variational Bayes (Implementation)

```
nu = 0.4; beta = 100; nu_beta = nu/beta;
P = 1.2; x_0 = 1; x_1 = -6;
T = 300; % Number of iterations
E_A = -6; E_A2 = E_A^2;
E_invR = 1/0.00001; % Initial Sufficient stats

for t=2:T,
    % Update q(A)
    Sig = 1/(1/P + x_0^2*E_invR);
    mu = Sig*x_0*x_1*E_invR;

    E_A = mu;        E_A2 = mu.^2 + Sig;

    % Update q(R)
    a = nu+0.5;
    b = 0.5*(x_1.^2 - 2*x_1*x_0*E_A + x_0.^2*E_A2) + nu_beta;

    E_invR = a/b;
end;
```

Variational Bayes



EM - Expectation Maximisation algorithm

- Variational Bayes and Gibbs are for full Bayesian learning
- EM :Maximum likelihood (ML) or Maximum a-posteriori parameter estimation

EM, Case 1

Maximise over the variance R

$$
\begin{aligned}
q(A)^{(\tau)} &= \exp(\log \phi(A, R = R^{(\tau-1)})) = p(A|R^{(\tau-1)}) \\
R^{(\tau)} &= \arg\max \left\langle \log \phi(A, R) \right\rangle_{q(A)^{(\tau)}}
\end{aligned}
$$

EM, Case 2

Maximise over regression coefficient A

$$
\begin{aligned}
A^{(\tau)} &= \arg\max \langle \log \phi(A, R) \rangle_{q(R)^{(\tau-1)}} \\
q(R)^{(\tau)} &= \exp(\log \phi(A = A^{(\tau)}, R)) = p(R|A^{(\tau)})
\end{aligned}
$$

Iterative Conditional Modes

Maximise over the variance R and the regression coefficient A

$$
\begin{aligned}
A^{(\tau)} &= \arg\max \langle \log \phi(A, R) \rangle_{q(R)^{(\tau-1)}} \\
&= \arg\max \log \phi(A, R = R^{(\tau-1)}) \\
R^{(\tau)} &= \arg\max \langle \log \phi(A, R) \rangle_{q(A)^{(\tau)}} \\
&= \arg\max \log \phi(A = A^{(\tau)}, R)
\end{aligned}
$$

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

References

Bayesci Zaman Serileri, 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.

References

M. J. Wainwright and M. I. Jordan, 2008, Graphical Models, Exponential Families, and Variational Inference, Foundations and Trends in Machine Learning, DOI: 10.1561/2200000001

References

Recent Trends

- Z. Ghahramani, Probabilistic machine learning and artificial intelligence, Nature, 2015, doi:10.1038/nature14541
  - Olasılıksal programlama,
  - Bayesci eniyileme,
  - Veri sıkıştırma
  - Otomatik model keşfetme