Computer Science

CSC535: Probabilistic Graphical Models

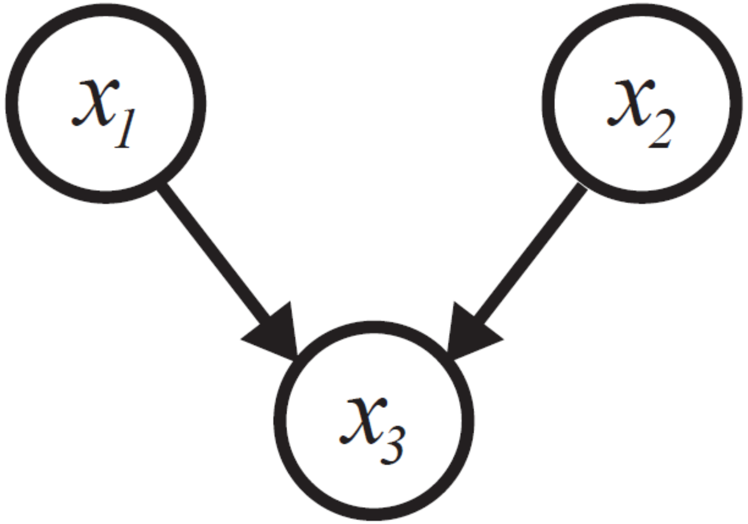
Introduction and Course Overview

Prof. Jason Pacheco

What is a Probabilistic Graphical Model?

*A probabilistic graphical model allows us to pictorially represent a probability distribution**

Graphical Model:



Probability Model:

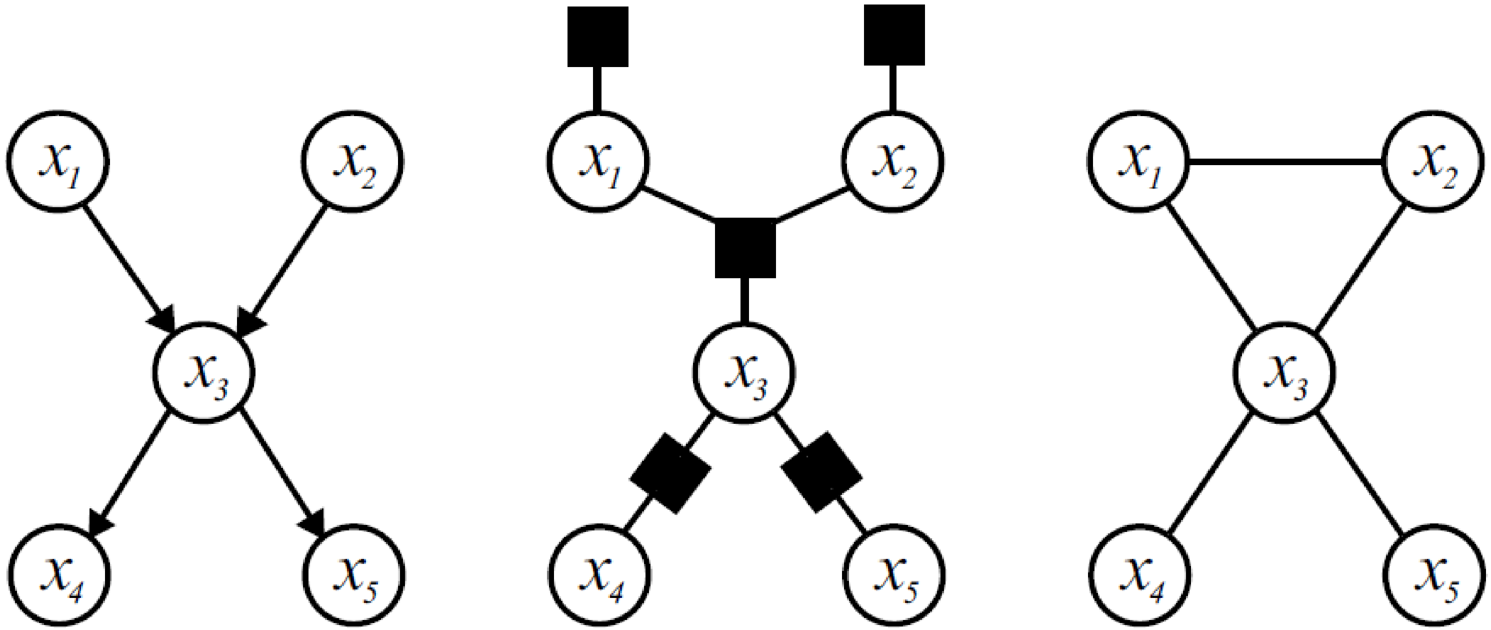
$$p(x_1, x_2, x_3) = p(x_1)p(x_2)p(x_3 \mid x_1, x_2)$$

The graphical model structure *obeys* the factorization of the probability function in a sense we will formalize later

*** We will use the term "distribution" loosely to refer to a CDF / PDF / PMF**

Graphical Models

A variety of graphical models can represent the same probability distribution



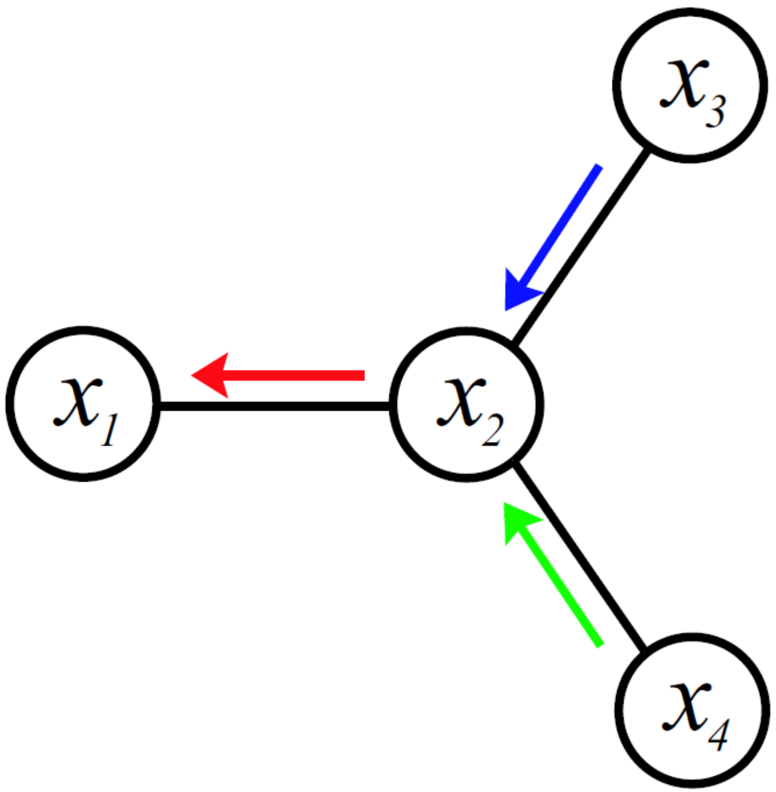
Bayes Network **Factor Graph** **Markov Random Field**

Directed Models **Undirected Models**

Erik Sudderth, PhD Thesis

Why Graphical Models?

Structure simplifies both **representation** and **computation**



Representation

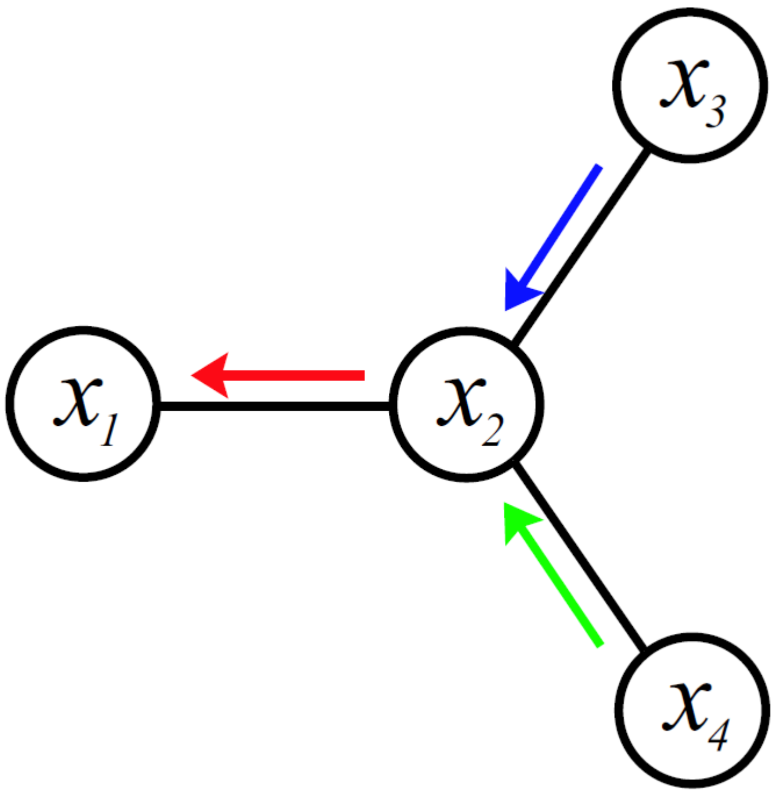
Complex global phenomena arise by simpler-to-specify local interactions

Computation

Inference / estimation depends only on subgraphs (e.g. dynamic programming, belief propagation, Gibbs sampling)

Why Graphical Models?

Structure simplifies both **representation** and **computation**



Representation

Complex global phenomena arise by simpler-to-specify local interactions

Computation

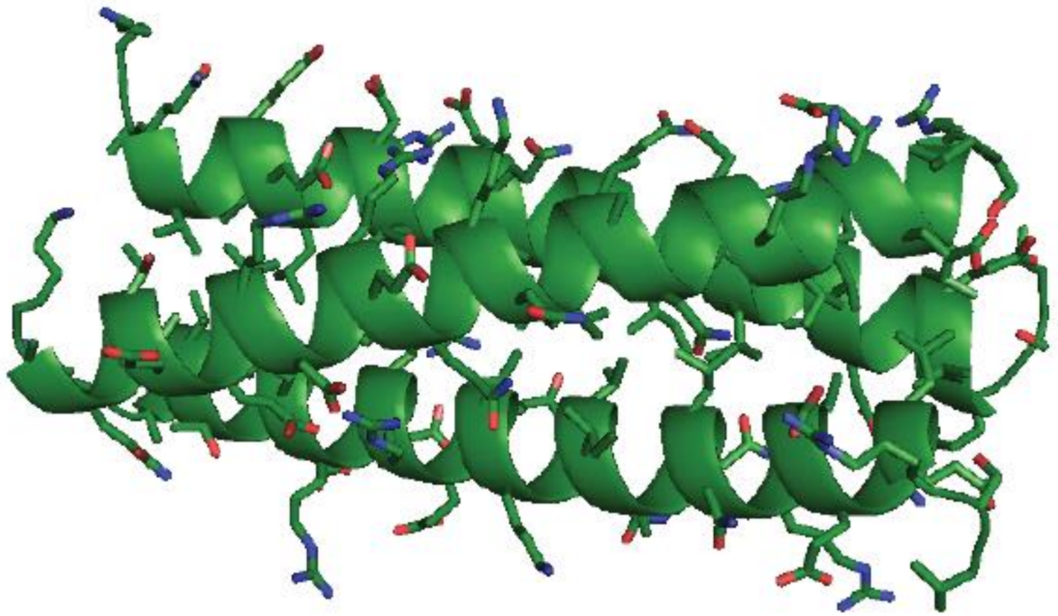
Inference / estimation depends only on subgraphs (e.g. dynamic programming, belief propagation, Gibbs sampling)

We will discuss inference later, but let's focus on representation…

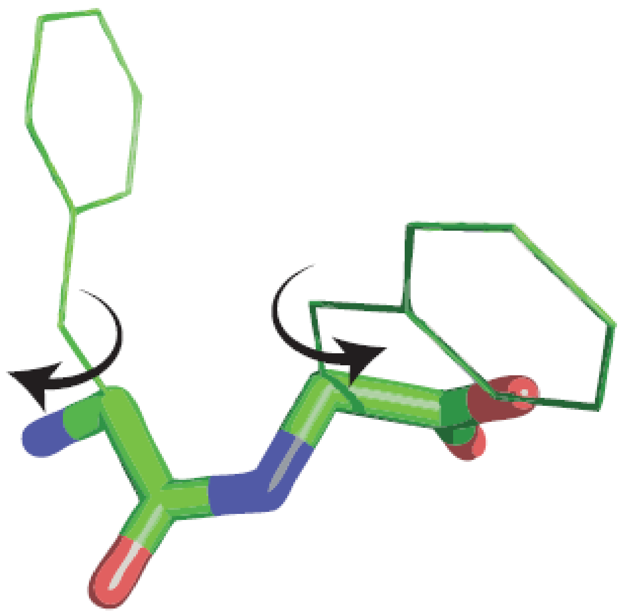
Protein Side Chain Prediction

Problem: Given 3D protein backbone structure, estimate orientation of every side chain molecule.

Backbone + Side Chains



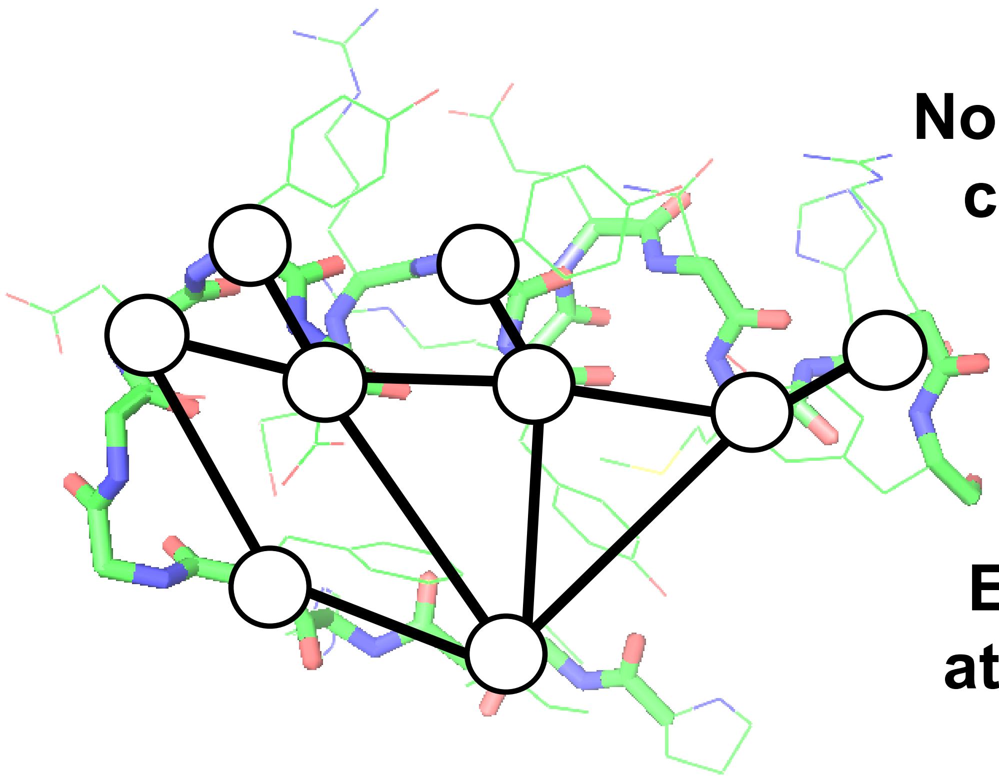
Side Chain Rotation



Solution: Just physics of atomic interaction. Easy, right!?

Protein Side Chain Prediction

Graphical Model

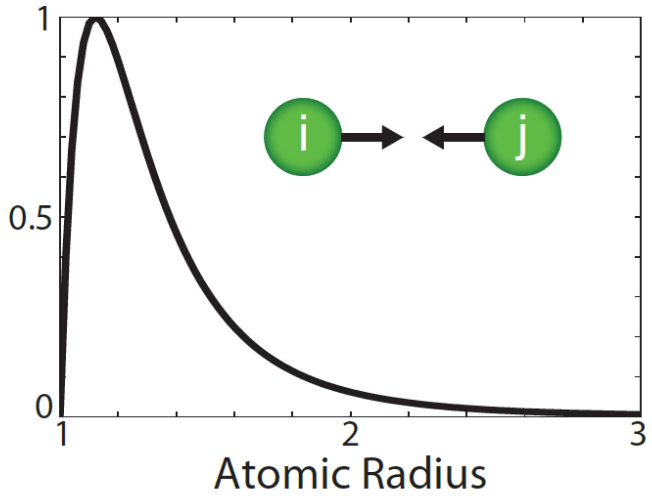


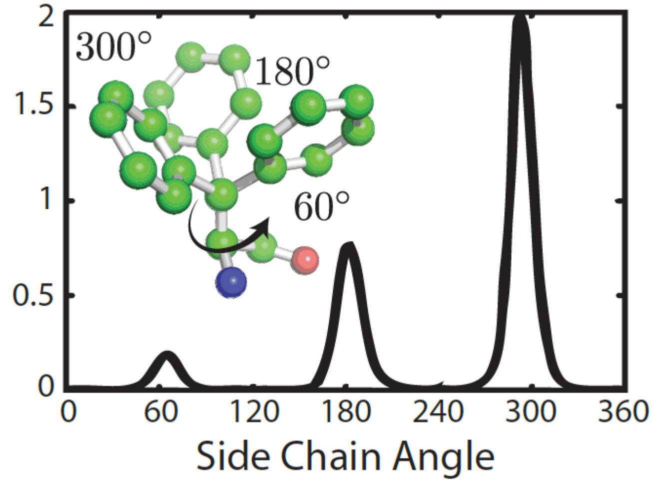
Nodes represent side chain orientations

Edges represent atomic interaction

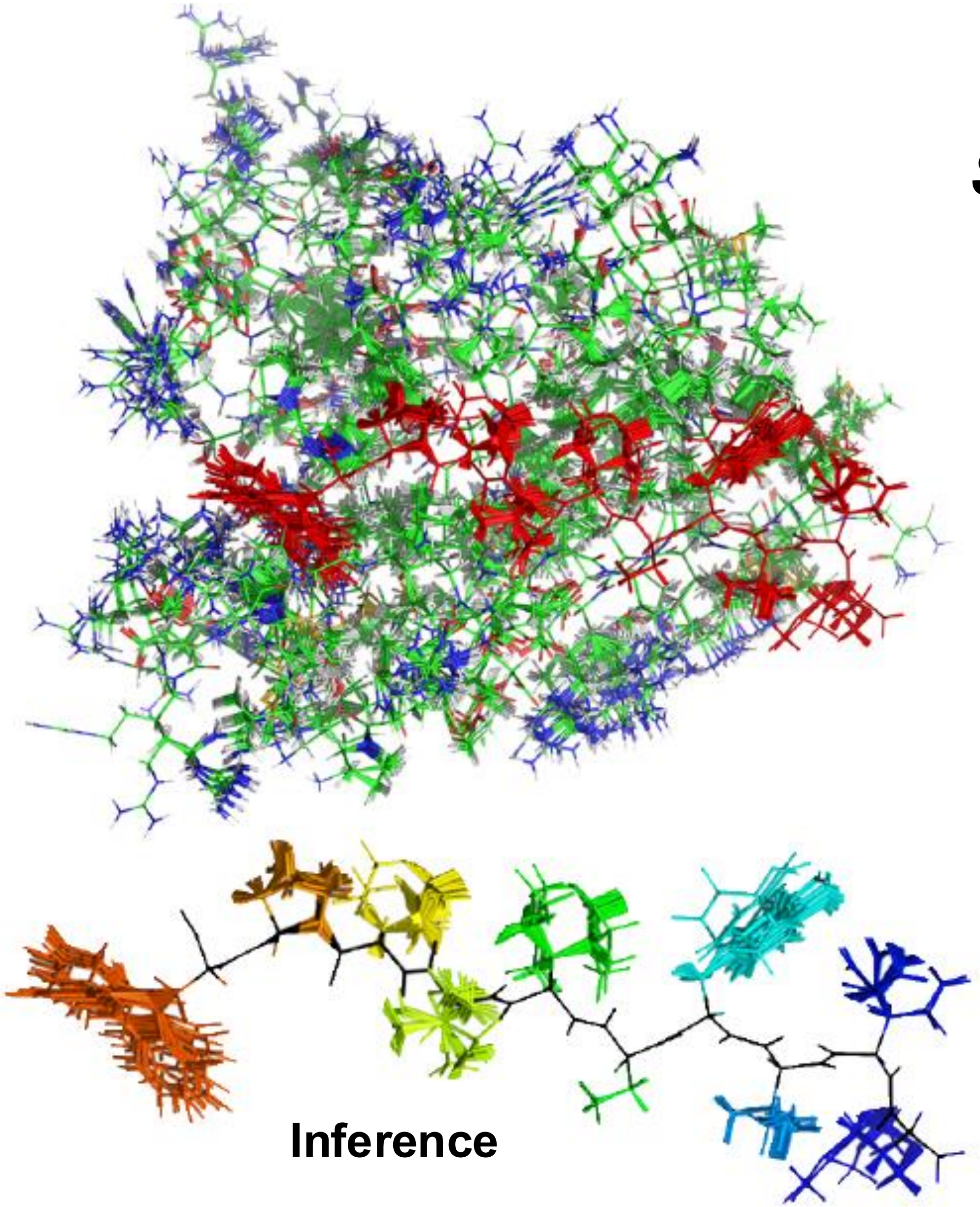
Complex phenomena specified by simpler atomic interactions

Configuration Likelihoods

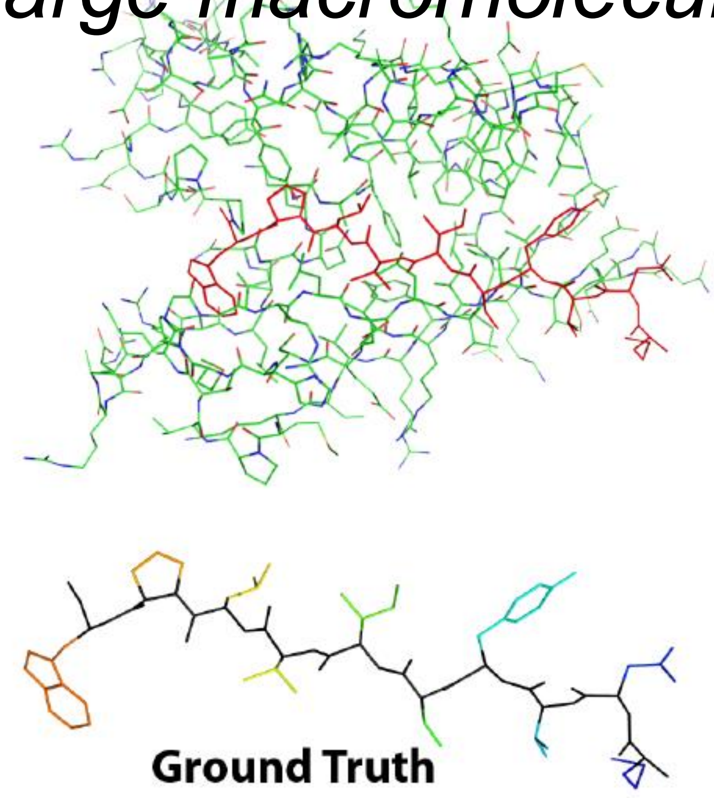




Protein Side Chain Prediction



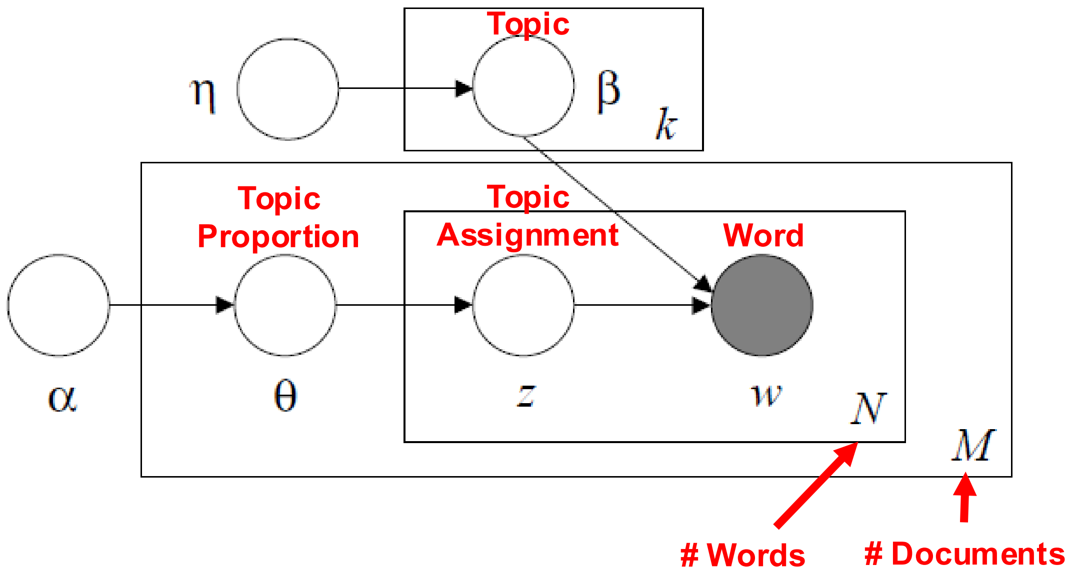
By exploiting graphical model structure we can scale computation to large macromolecules



Pacheco and Sudderth, ICML 2015

Topic Models

Latent Dirichlet Allocation (LDA)



Allows *unsupervised learning* of document corpus via mixture modeling

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

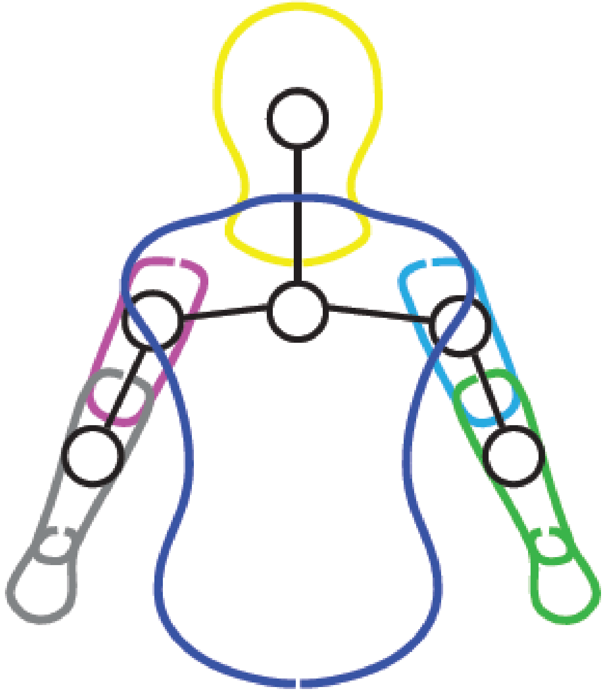
The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

Blei, D. et al., JMLR 2003

Pose Estimation

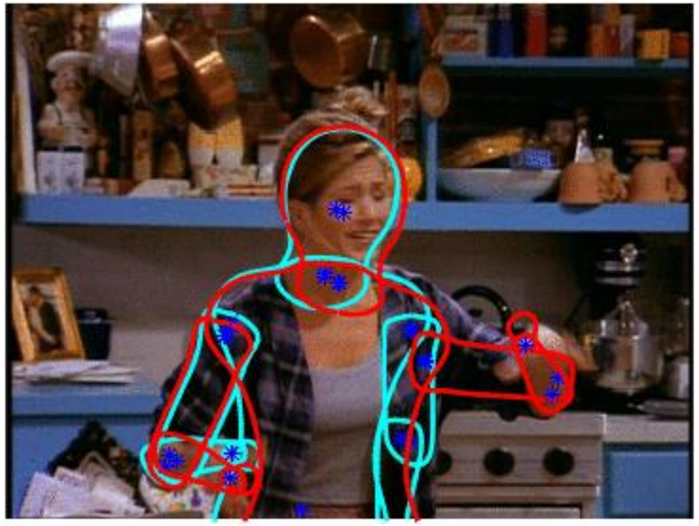
Estimate orientation / shape / pose of human figure from an image

Graphical Model

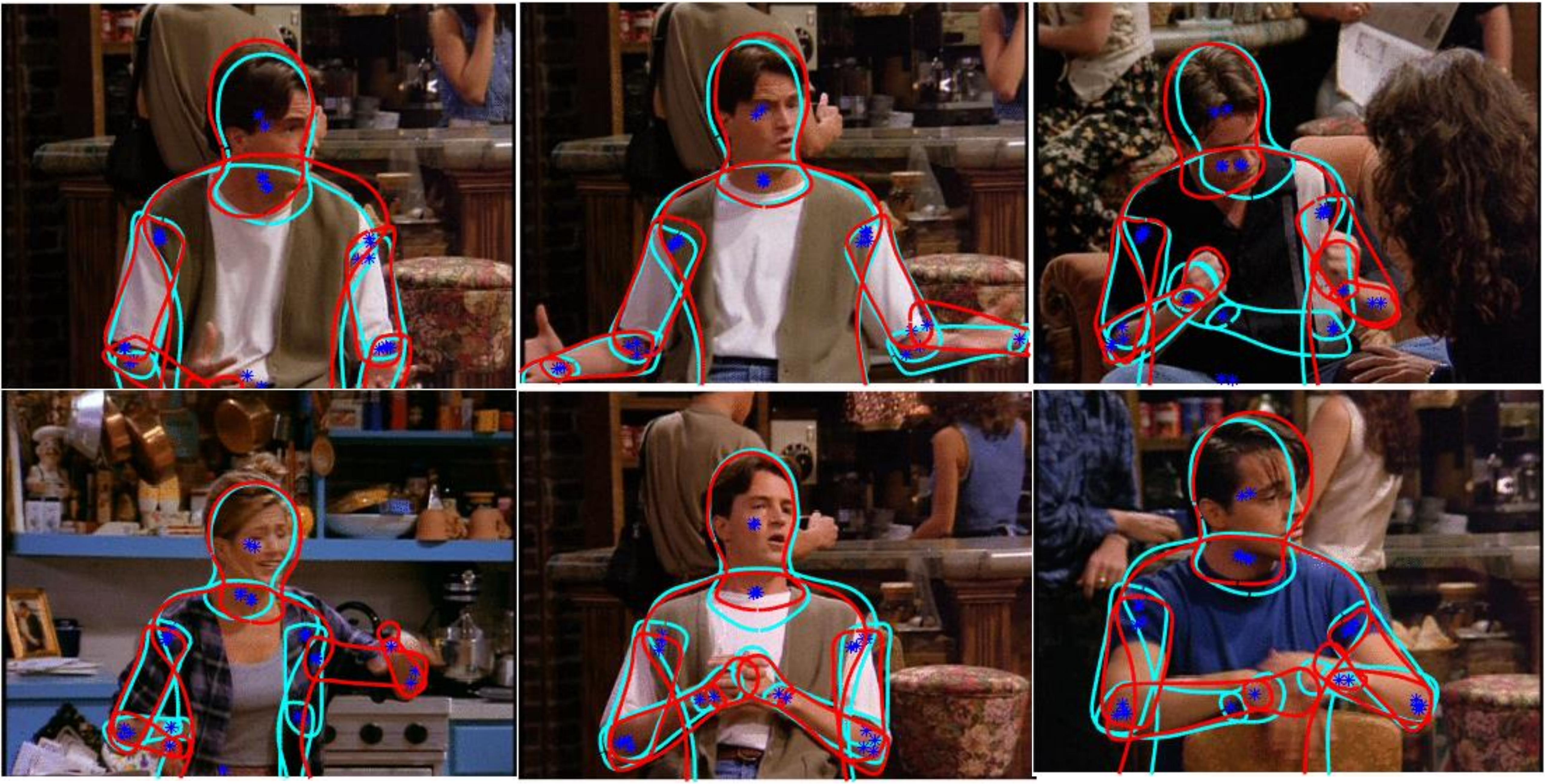


Data

Estimates

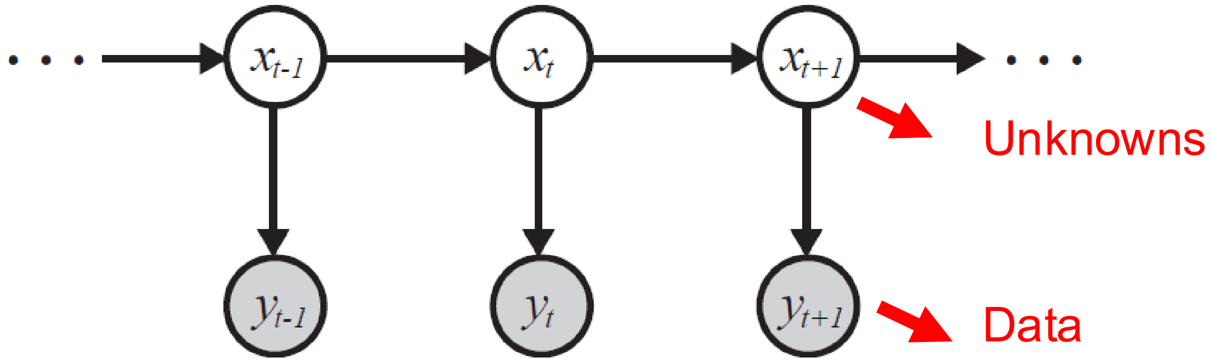


Pacheco, et al., ICML 2014



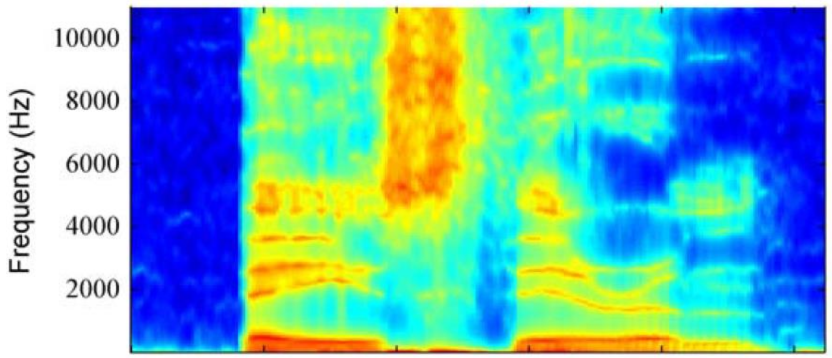
Hidden Markov Models

Sequential models of *discrete quantities of interest*



Example: Speech Recognition

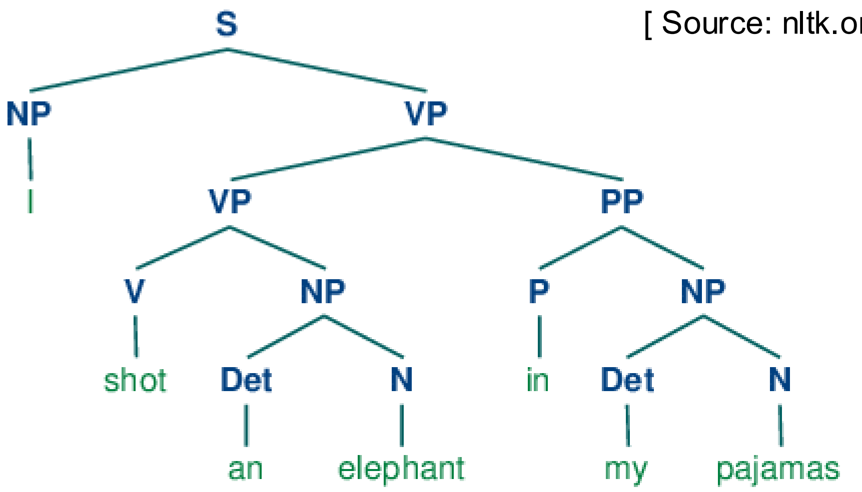
$y =$



x = b-ey-z-th-ih-er-em → Bayes' Theorem

[Source: Bishop, PRML]

[Source: nltk.org]



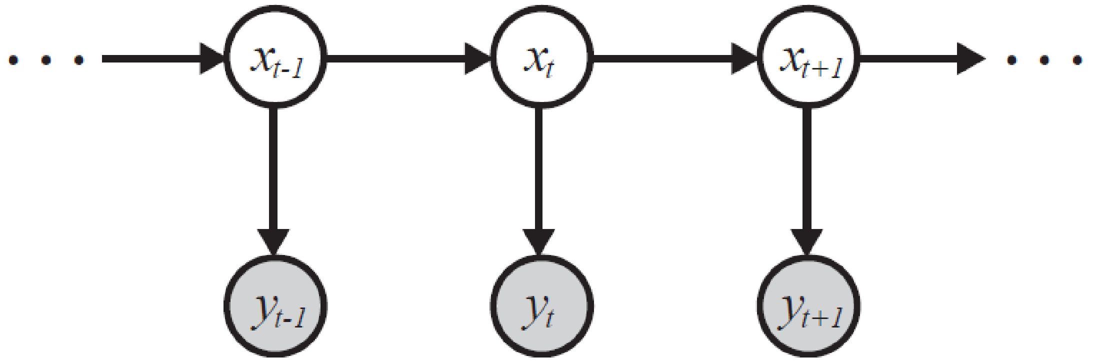
Example: Part-of-speech Tagging:

y = "I shot an elephant in my pajamas."

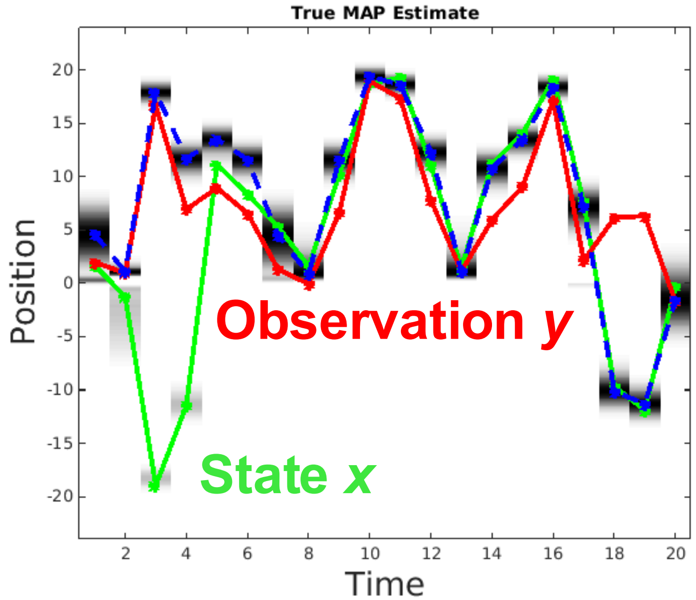
x = NP-V-Det-N-P-Det-N

Dynamical Models

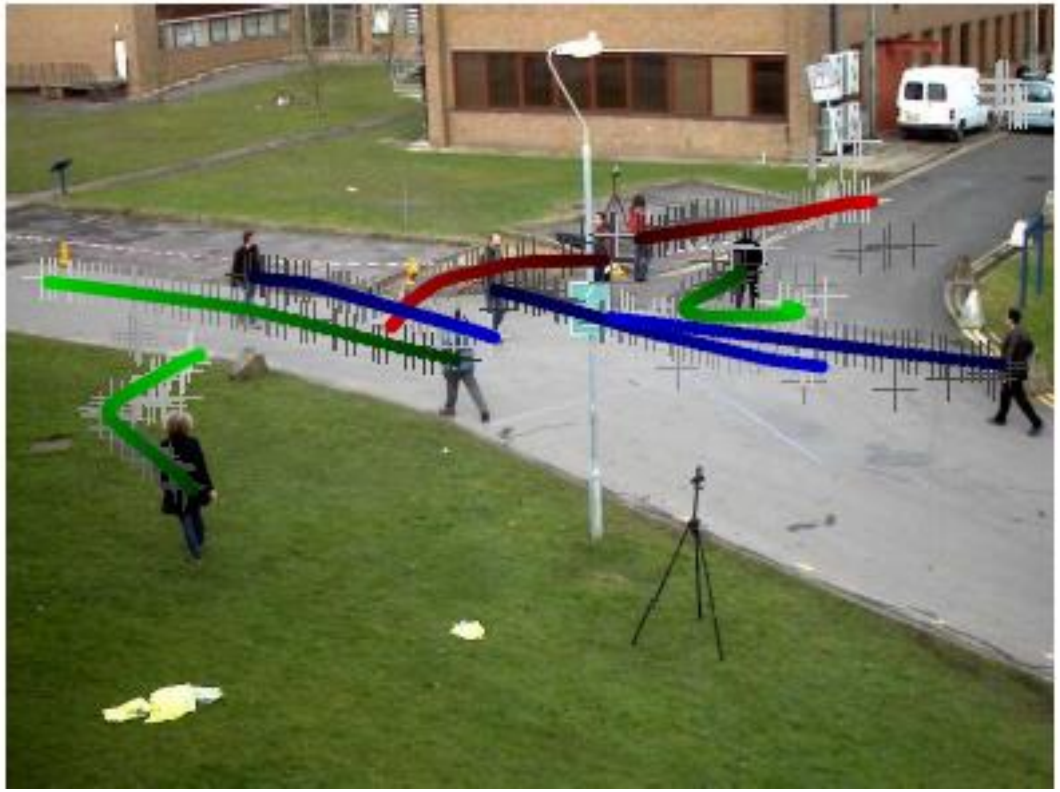
Sequential models of <u>*continuous*</u> *quantities of interest*



Example: Nonlinear Time Series

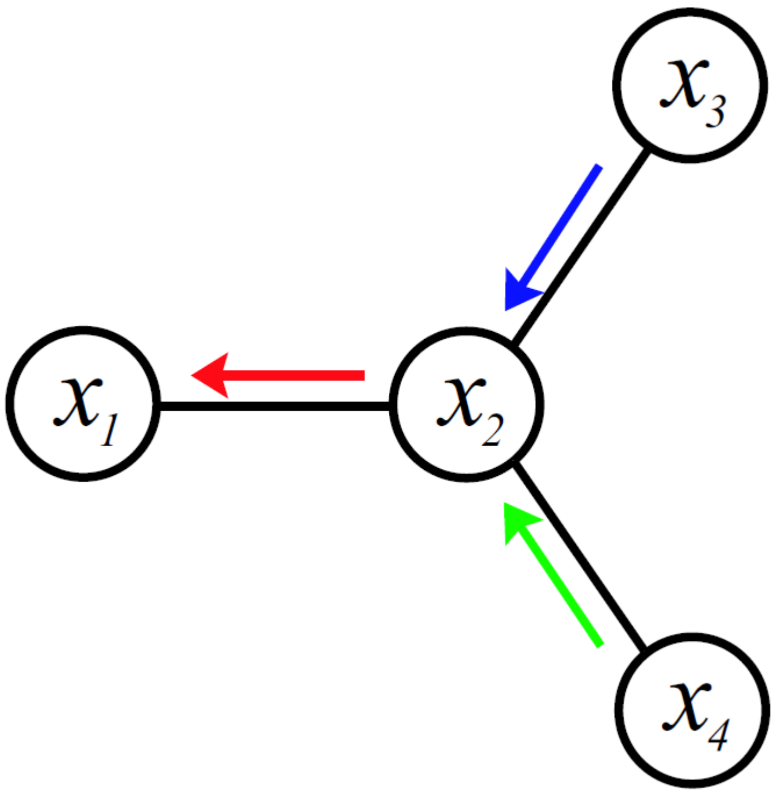


Example: Multitarget Tracking



Why Graphical Models?

Structure simplifies both **representation** and **computation**

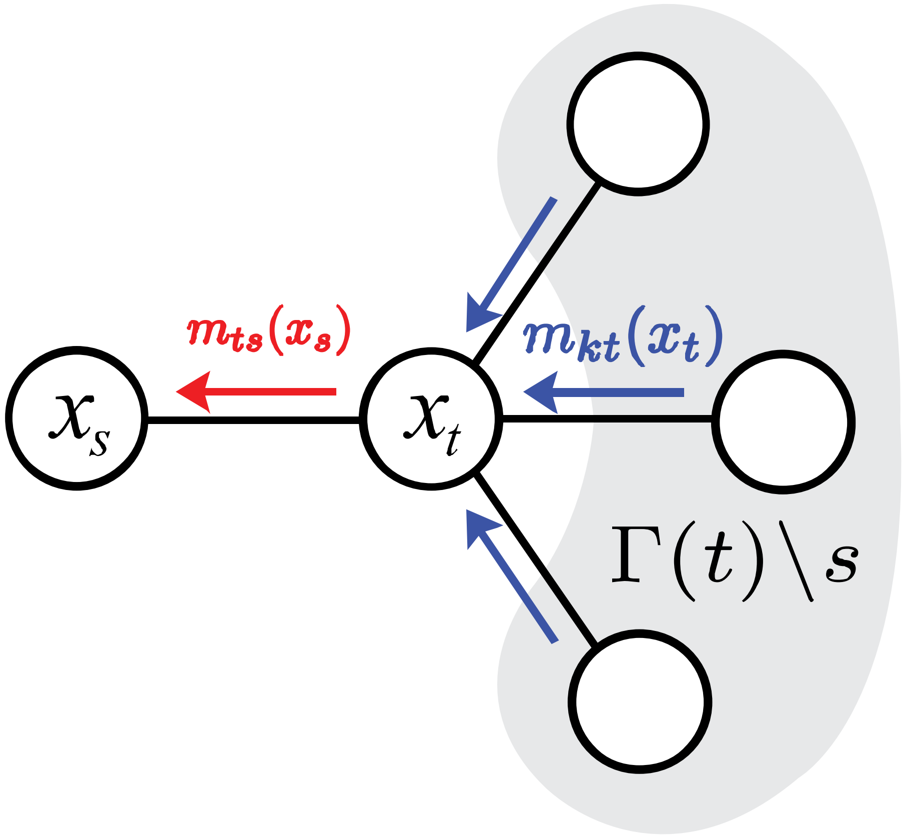


Representation
Complex global phenomena arise by simpler-to-specify local interactions

Computation
Inference / estimation depends only on subgraphs (e.g. dynamic programming, belief propagation, Gibbs sampling)

Dynamic Programming (DP)

Breaks difficult global computations into simpler local updates



Many algorithms use some form of DP

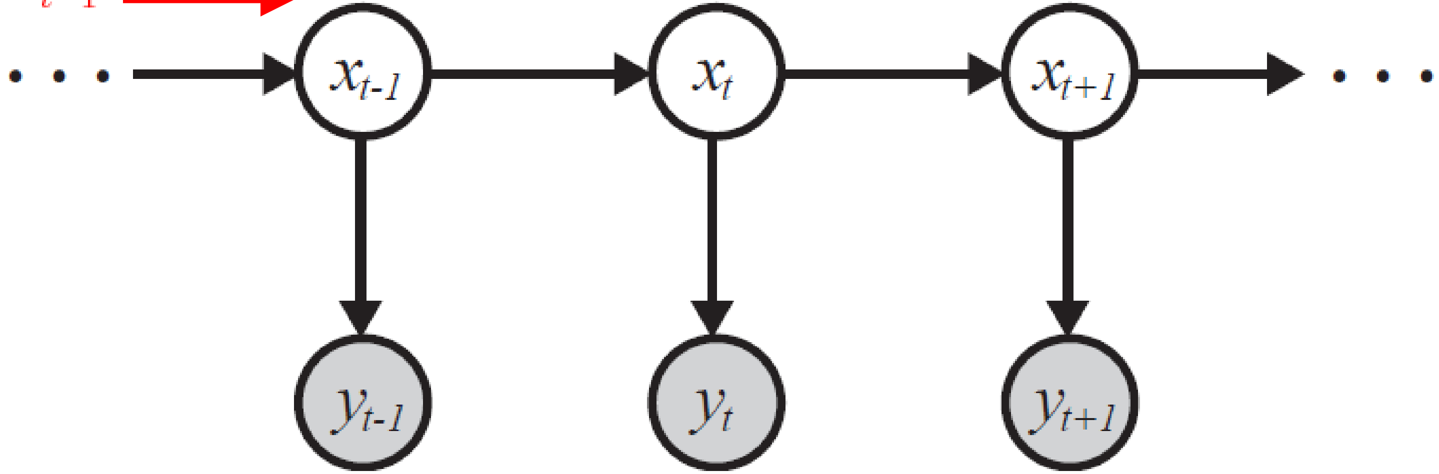
- Belief propagation
- Gibbs sampling
- Particle filtering
- Viterbi decoder for HMMs
- Kalman filter (marginal inference)

Key Idea: Local computations only depend on the statistics of the current node and neighboring interactions

Viterbi Decoder

Summary of

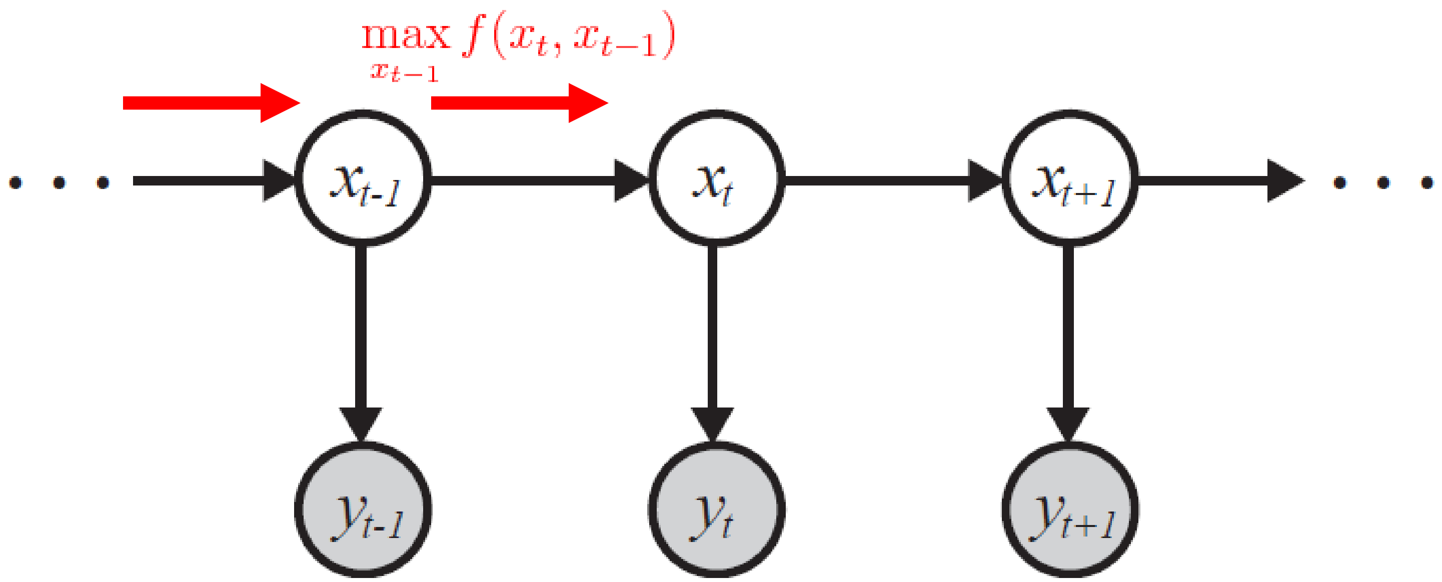
$x_1, \ldots, x_{t-1}$



$$x^* = \operatorname*{argmax}_{x} p(x \mid y)$$

Efficiently computes MAP estimate for state-space model by *passing messages* forward and backward along chain.

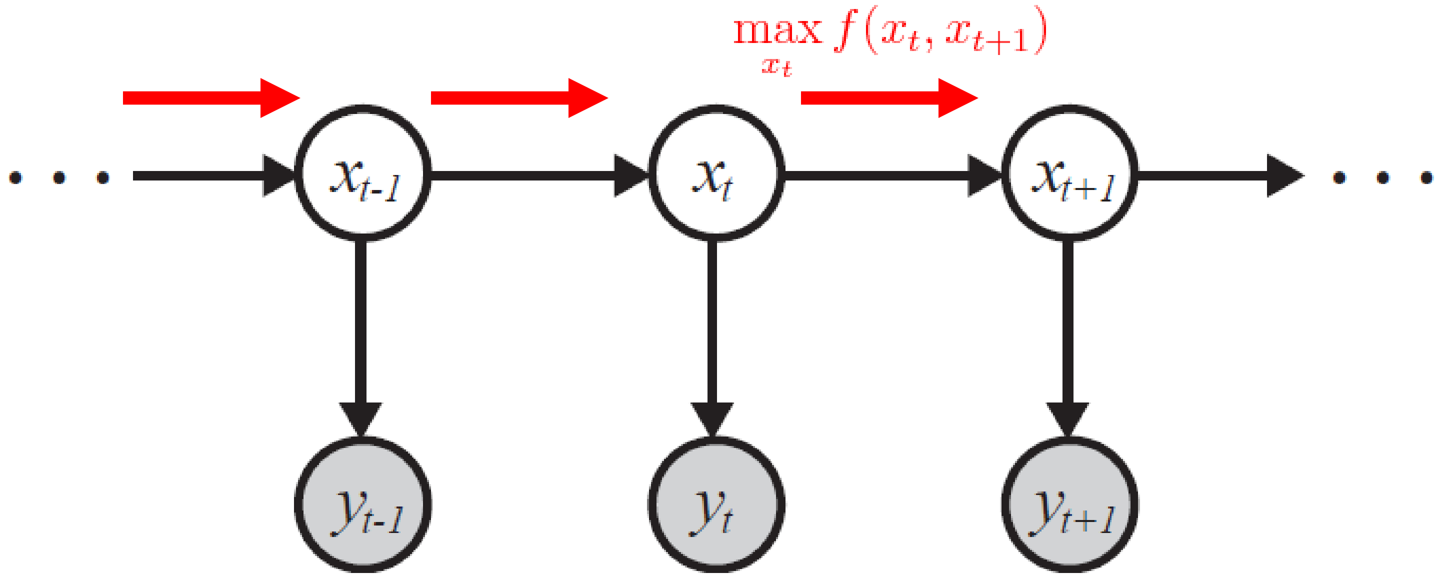
Viterbi Decoder



$$x^* = \operatorname*{argmax}_x p(x \mid y)$$

Efficiently computes MAP estimate for state-space model by *passing messages* forward and backward along chain.

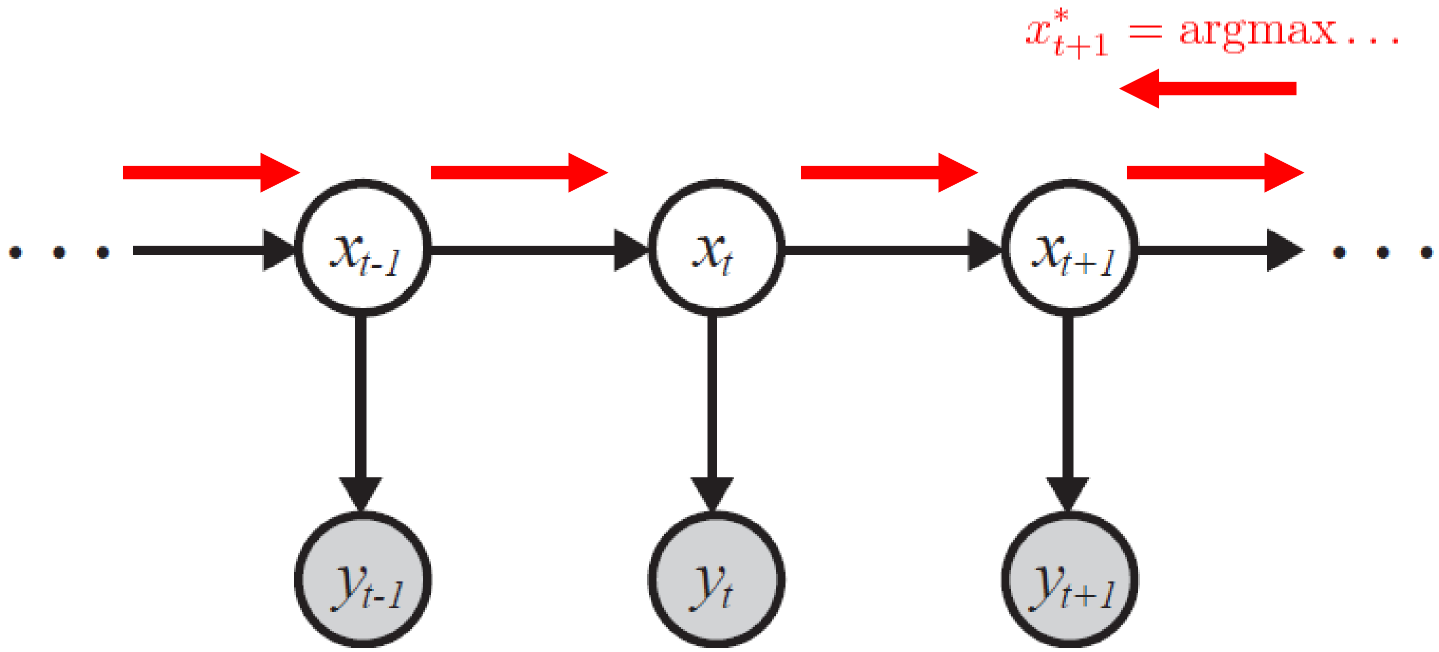
Viterbi Decoder



$$x^* = \operatorname*{argmax}_x p(x \mid y)$$

Efficiently computes MAP estimate for state-space model by *passing messages* forward and backward along chain.

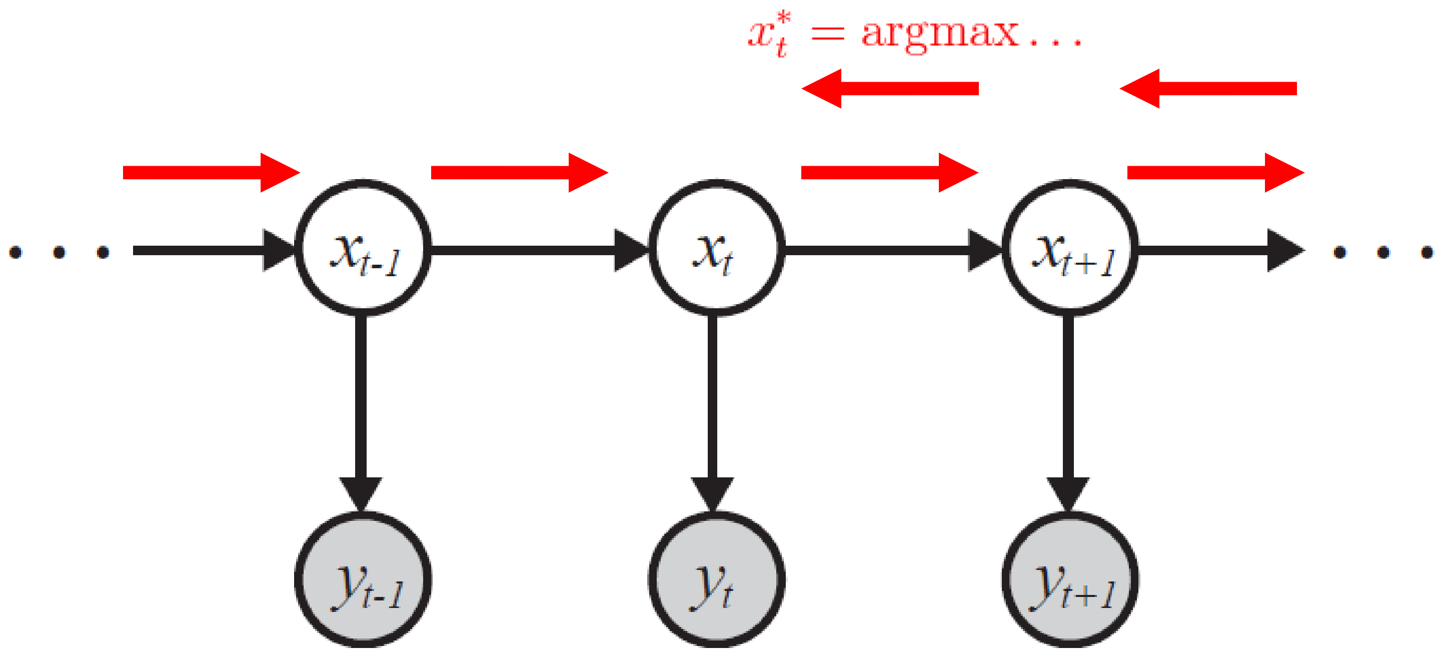
Viterbi Decoder



$$x^* = \underset{x}{\operatorname{argmax}}\, p(x \mid y)$$

Efficiently computes MAP estimate for state-space model by *passing messages* forward and backward along chain.

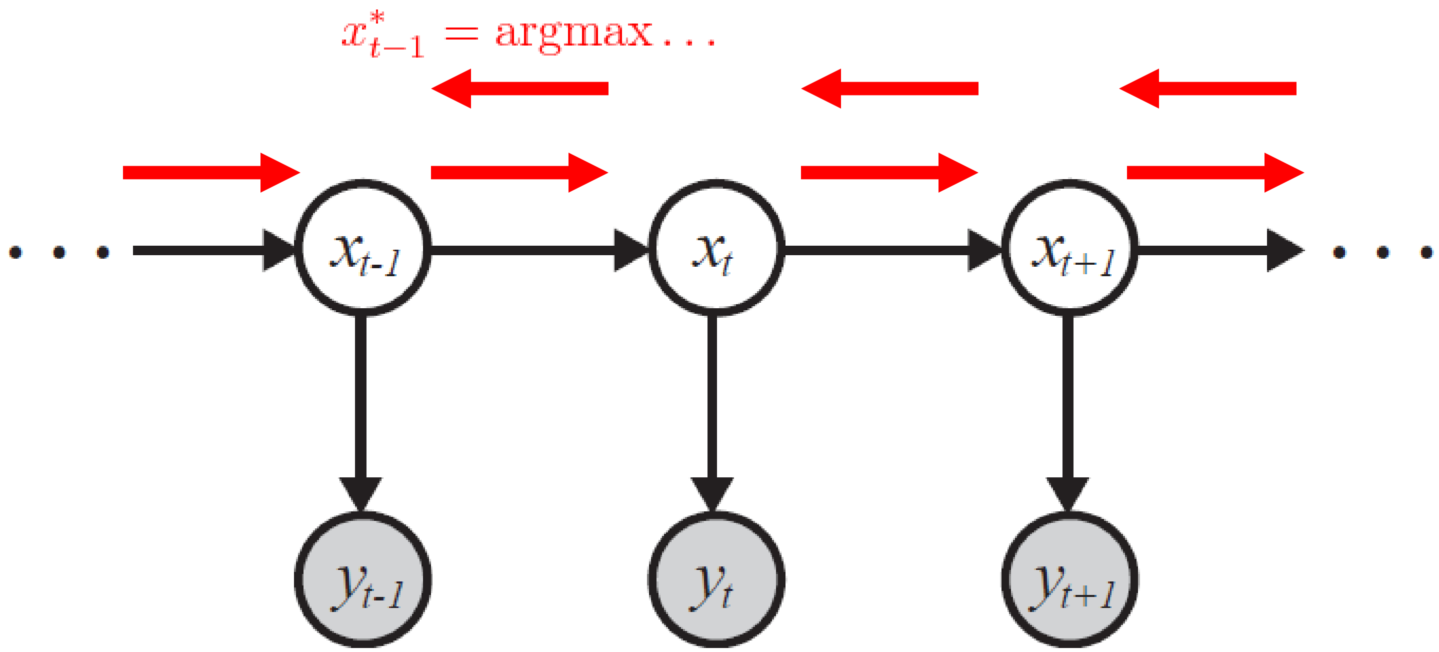
Viterbi Decoder



$$x^* = \underset{x}{\text{argmax}}\, p(x \mid y)$$

Efficiently computes MAP estimate for state-space model by *passing messages* forward and backward along chain.

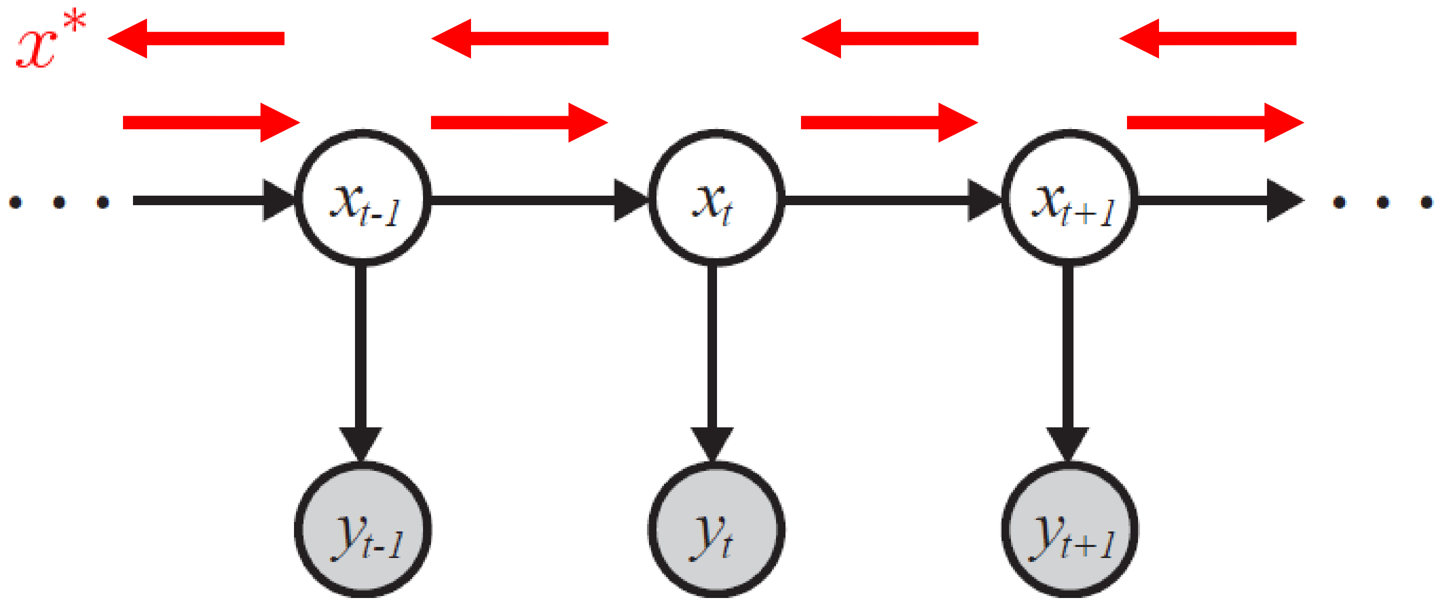
Viterbi Decoder



$$x^* = \underset{x}{\text{argmax}}\, p(x \mid y)$$

Efficiently computes MAP estimate for state-space model by *passing messages* forward and backward along chain.

Viterbi Decoder



$$x^* = \underset{x}{\operatorname{argmax}}\, p(x \mid y)$$

Efficiently computes MAP estimate for state-space model by *passing messages* forward and backward along chain.

Course Objectives

Along with basic familiarity of PGMs, we will develop the following basic skills…

- Create directed / undirected graphical models of stochastic processes
- Identify conditional independencies in graphical models
- Apply exact inference to compute marginal probabilities and maximally probable configurations given a model (elimination, sum-product, and max-sum algorithms)
- Apply approximate inference to learn model parameters using expectation maximization (EM), variational inference, and various Monte Carlo methods

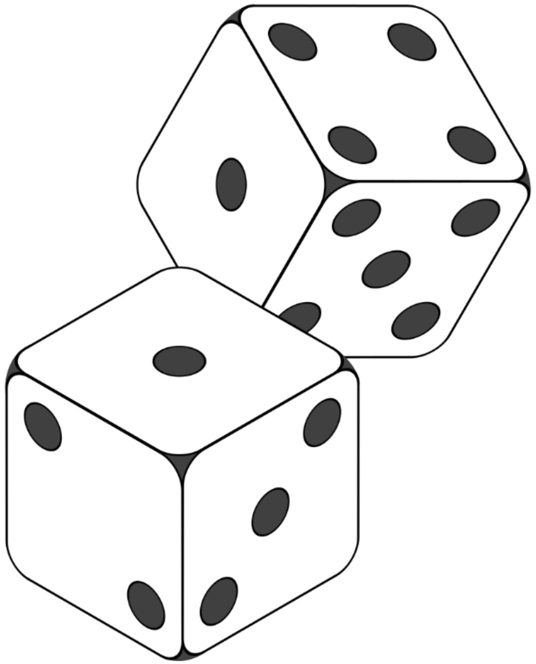
Course Overview

Course is broken down into **six** primary topics…

Probability and Statistics	Message Passing Algorithms	Monte Carlo Methods	Sequence Models and Dynamical Systems	Variational Inference	Bayesian Deep Learning
Probability primer, Bayesian statistics, PGMs, Exponential families	Elimination, Junction tree, Sum-product / max-product, Belief propagation, Viterbi decoding	Rejection sampling, Importance sampling, Metropolis-Hastings, Gibbs	Linear and switching state-space models, Kalman filter, Particle filter	Mean field, Stochastic variational, Bethe energy methods	Combining probability and deep learning models

Probability and Statistics

Suppose we roll <u>two fair dice</u>…



- What are the possible outcomes?
- What is the *probability* of rolling **even** numbers?

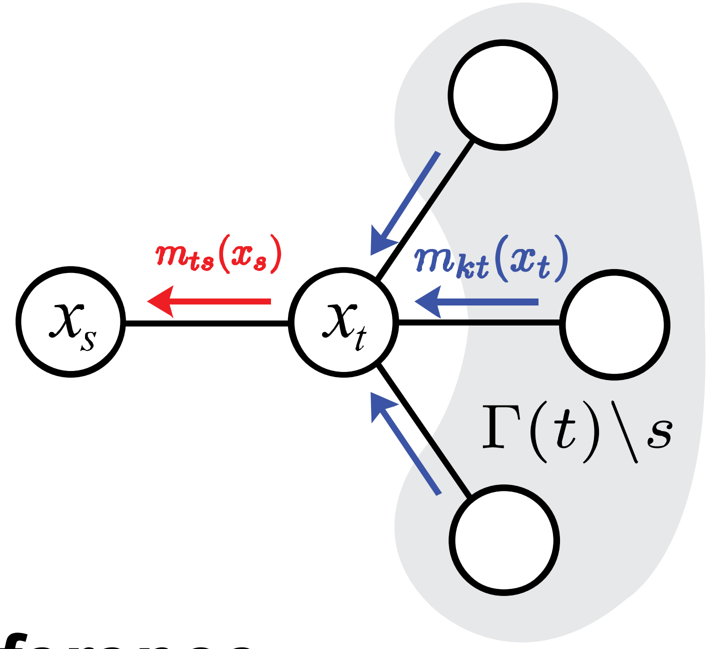
*… this is an **experiment** or **random process**.*

We will learn how to…

- Mathematically formulate outcomes and their probabilities?
- Describe characteristics of random processes
- Estimate unknown quantities (e.g. are the dice actually fair?)
- Characterize the uncertainty in random outcomes
- Identify and measure dependence among random quantities

Message Passing Algorithms

Encompasses a family of dynamic programming algorithms for performing exact / approximate inference

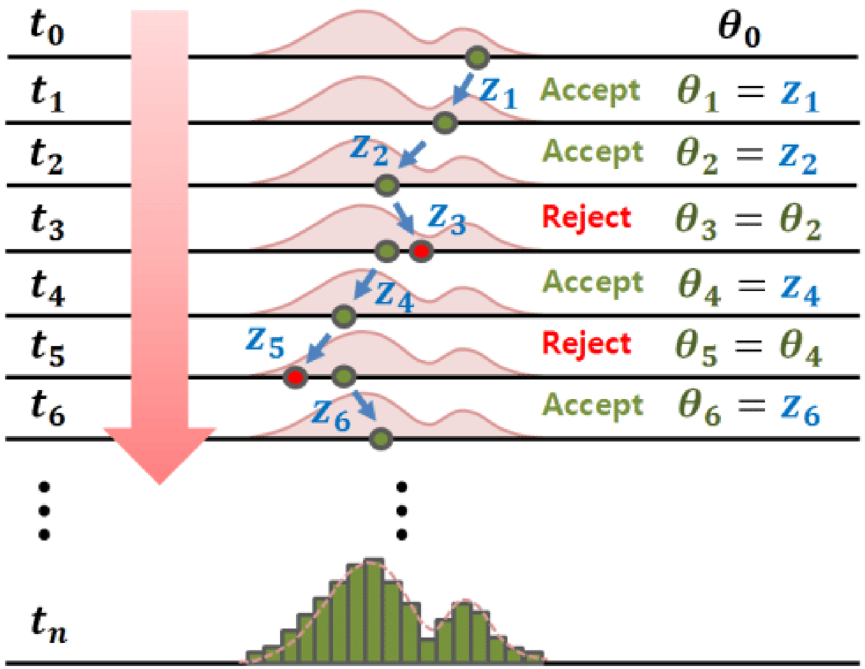


We will learn the following message passing inference

- Variable elimination
- Junction tree
- Sum-Product and Max-Product Belief Propagation
- Loopy Belief Propagation (approximate inference)

Monte Carlo Methods

Sample-based methods that simulate realizations from the model to perform inference

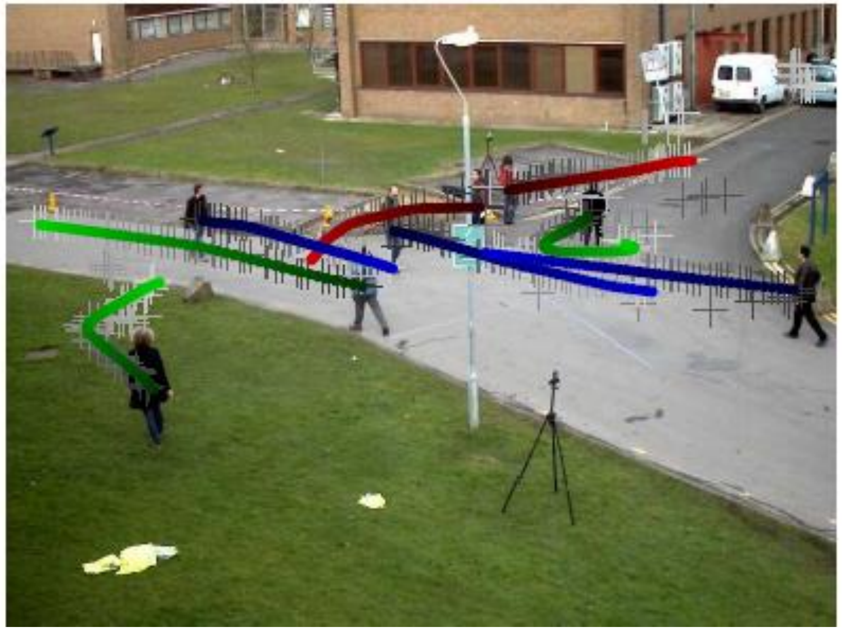


We will learn how to perform sample-based inference using:

- Rejection sampling
- Importance sampling
- Sequential importance sampling (particle filter)
- Markov chain Monte Carlo (MCMC) : Metropolis-Hastings
- MCMC : Gibbs Sampling

[Image: Jin, Seung-Seop et al. (2019)]

Sequence Models / Dynamical Systems



Data follow an explicit ordering or sequence…

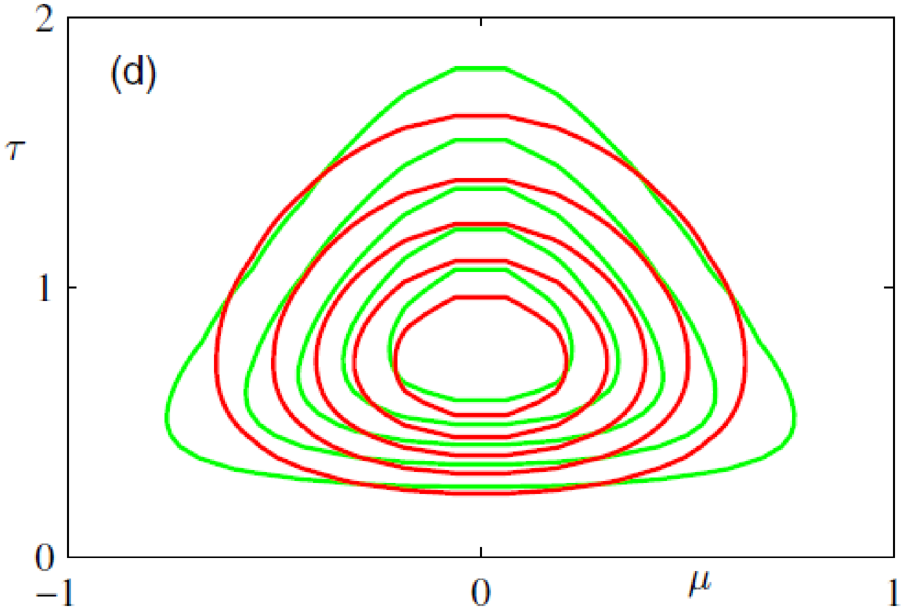
- *Hidden Markov Model*
- *Forward Backward Algorithm*
- *Baum-Welch Algorithm*

State-space models describe time-ordered data

- Target tracking
- Linear Dynamical System
- Kalman Filter
- Switching State-Space Models
- Nonlinear Dynamical Systems

Variational Inference

Recasts statistical inference as the solution to an optimization problem



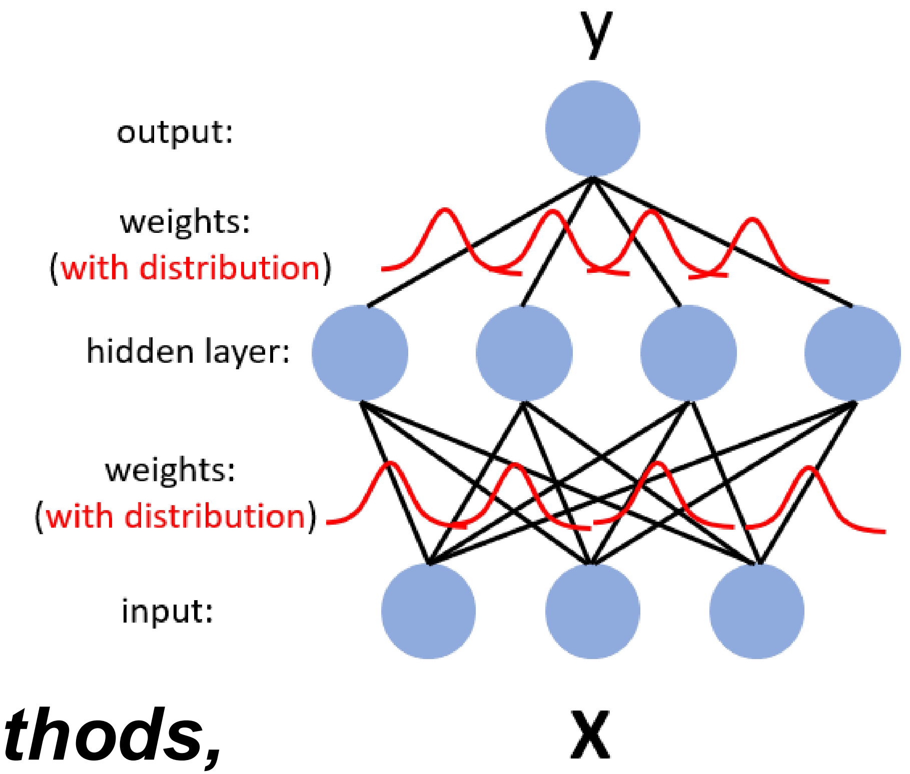
$$min_q \, \mathrm{KL}(q(x) \| p(x))$$

We will learn how to conduct inference via,

- *Mean field and variational Bayes*
- *Stochastic variational*
- *Bethe free energy methods (Belief Propagation, Expectation Propagation)*

Bayesian Deep Learning

Combine probabilistic reasoning with Deep Learning models



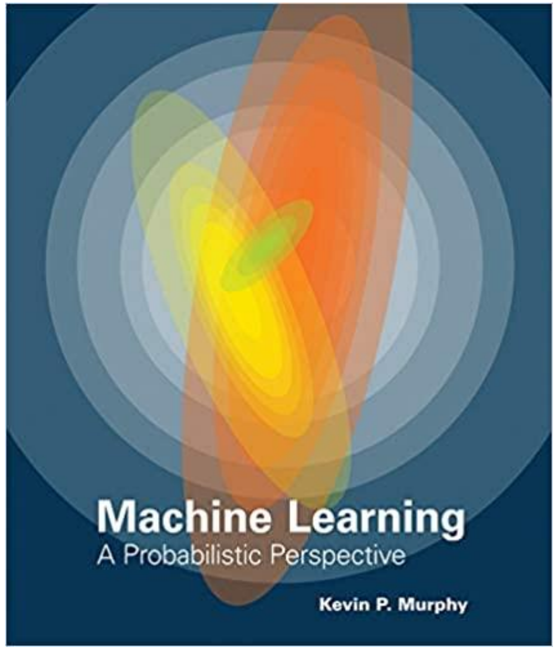
We will learn the following models and methods,

- *Variational autoencoder*
- *Bayesian Neural Network*
- *Structured Variational Autoencoders*
- *Dropout predictions*

[Image: Lin, Xi et al. (2018)]

Textbook

Readings assigned for each lecture



Murphy, K. "Machine Learning: A Probabilistic Perspective." MIT press, 2012

[(UA Library)]

There is a newer version floating around on the internet – we are using the 2012 copy

All readings and additional material will be posted on the course webpage

Online Resources

All material (lectures / HWs / readings) are available on the **course webpage**:

http://pachecoj.com/courses/csc535_fall25

We will use **D2L** for Zoom links, submitting assignments, grades:

https://d2l.arizona.edu/d2l/home/1651352

We will use **Piazza** for discussion:

https://piazza.com/arizona/fall2025/csc535

Grading

6 Homeworks + Midterm + Final Exam

Grading Breakdown

- Homework: 65%
- Midterm: 15%
- Final: 20%

Assignments will be a mix of math and coding…you will generally have 2 or more weeks (depending on difficulty)

Programming Language:

- Some homeworks will provide **Python** template code
- You may use Matlab / R / etc. but will need to reimplement any handout code

Grading Questions

- I will announce in class and/or Piazza when grading of each item is complete
- Officially, you have **1 week** to raise any grading concerns (from the completion of grading)
- If you don’t receive a grade, but should have, you must tell me **within 1 week**

Required Background

Undergrad calculus

- You should be able to compute derivatives
- You should understand the fundamentals of integration

Basic understanding of optimization

- Nonlinear vs. linear programming
- Gradient ascent
- Dynamic programming (we will cover the basics)

Basic Linear Algebra

- Basic vector / Matrix algebra
- Matrix inversion / rank / condition

Basic understanding of probability and statistics

- Marginal and conditional probability distributions
- Expected values / Variance
- Maximum likelihood estimation

Late Policy

Late submissions impact other students, delay grading, and delay solutions

But sometimes we need a little extra time…

- **No more than 1** late assignment **no more than 2 days late** without penalty
- Subsequent late assignments will be deducted a full grade point per day late
- I will not accept an assignment more than 2 days late
- I will consider special circumstances on a case-by-case basis (let me know ahead of time)
- D2L will accept late assignments but they will be flagged
- I will not accept late midterm / final exams

Academic Integrity Rules

- You may discuss assignments with other students
- You **may not** discuss or share assignment solutions
- You may consult any online or textbook resources
- You **may not** directly copy from external resources
- You **may not** upload solution material publicly accessible web
- You **may not** discuss exams with students in any capacity

Good Rule Cite any external resource you use that may be considered plagiarism without citation.

Lectures and Attendance

Lecture Attendance

- You should attend lecture regularly
- This will encourage in-class Q&A and discussion to clarify points
- Class sizes are small and you will be missed if absent

Lecture Recordings

- Lectures will not be recorded this semester

Office Hours

Use scheduled office hours for

- ➢ Specific homework questions
- ➢ Clarification on lecture / reading topics
- ➢ General course-related questions

Details

- ➢ 2 hours per week
- ➢ Office hours will be held on Zoom
- ➢ Message me on Piazza if you have a conflict with hours and I will try to schedule something for you
- ➢ Friday 3-5pm (beginning next week)

Piazza

- Use Piazza for **all course communication**
- If you email me directly I may not see it (I get a lot of email)
- You can ask / answer questions related to the course
- Also post course-related material (e.g. if you find something on the internet that is interesting / useful to the course)

Questions? Comments? Thoughts?