Graphical Models for Sequential Data Modeling and Forecasting

Padhraic Smyth
Information and Computer Science
University of California, Irvine
www.datalab.uci.edu

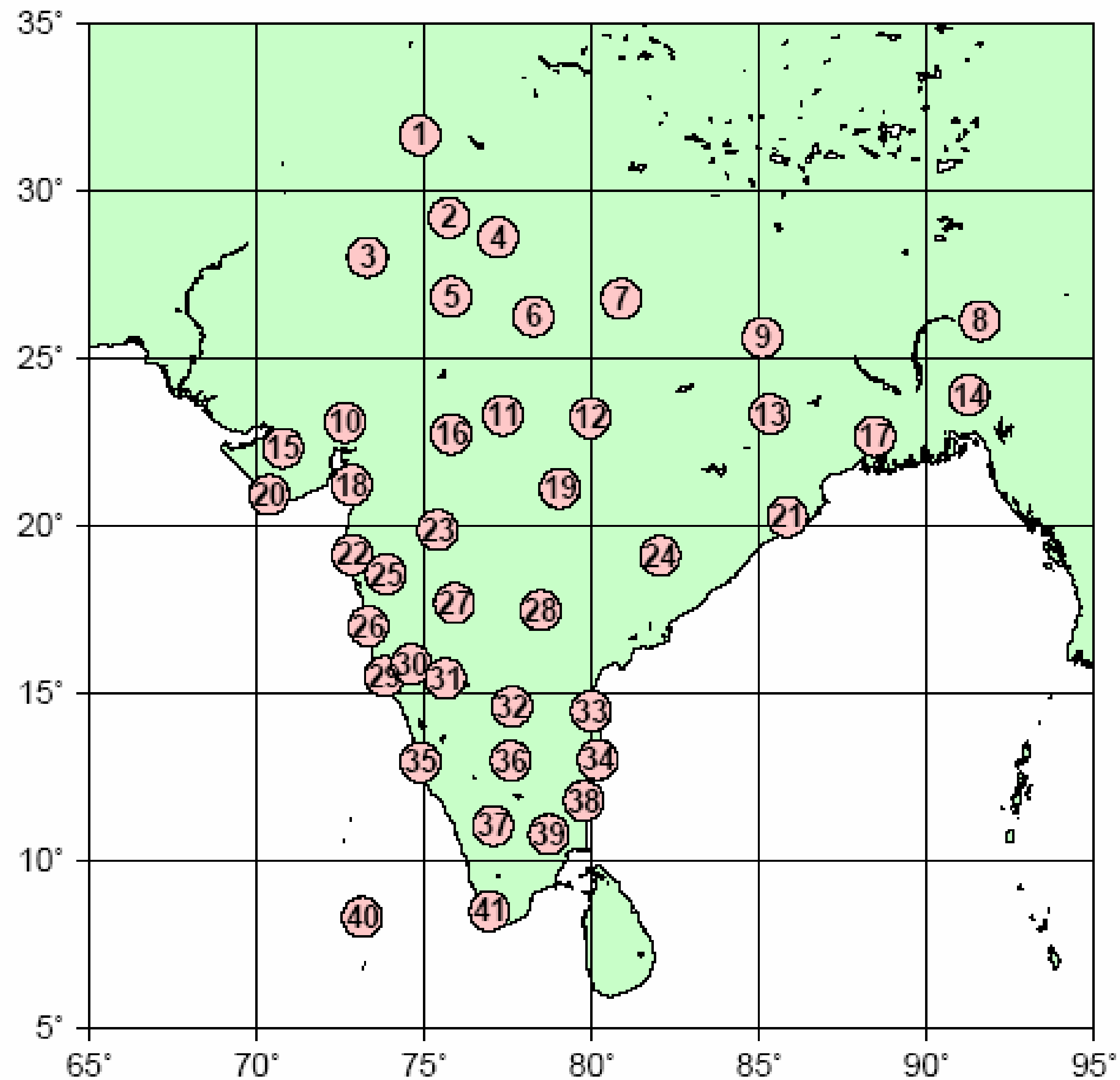
Collaborators

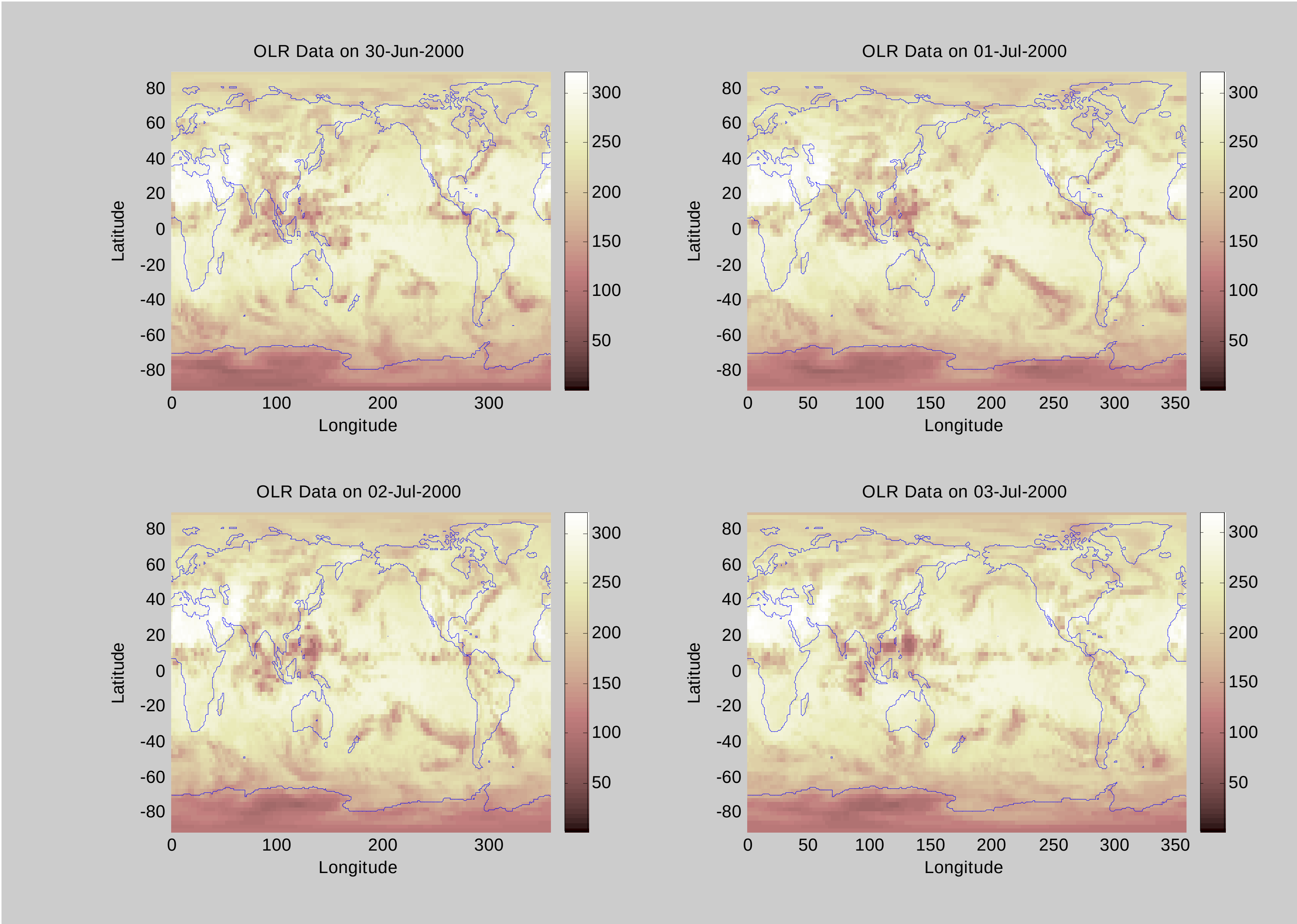
- UC Irvine, computer science
  - Scott Gaffney, Sergey Kirshner
- Atmospheric science
  - Andy Robertson, Suzana Camargo, Michael Ghil

Outline

- Graphical models
  - a framework for working with sets of random variables
  - Modeling sequential data
  - Estimating graphical models from data
- Examples
  - Cyclone clustering
  - Precipitation modeling with hidden Markov models
- Research problems, future directions

India 1973-03 NCDC GSOD Rainfall Stations




OLR Data on 30-Jun-2000
OLR Data on 01-Jul-2000
OLR Data on 02-Jul-2000
OLR Data on 03-Jul-2000
Latitude
Longitude
80
60
40
20
0
-20
-40
-60
-80
0
50
100
150
200
250
300
350

Prediction and Uncertainty

- Uncertainty is ever-present in climate science
  - Model uncertainty
    - which model is more likely given observed data?
  - Forecasting and prediction
    - Distributions over future outcomes
  - Modeling unobserved phenomena
  - Measurement error

- Probability is the language of uncertainty
  - Graphical models are a systematic framework for handling large numbers of random variables

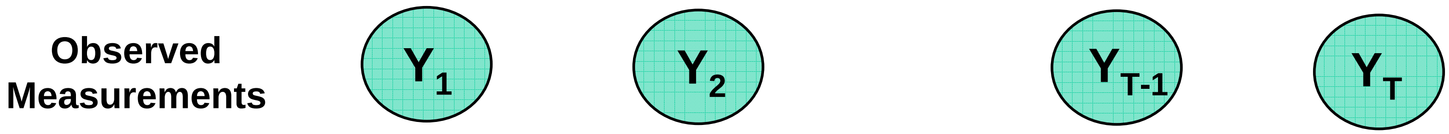
Preliminaries

- Variables
    - Y = y : observed variable
    - S = s : unobserved state variable
    - P(S = s | Y = y) = P(s|y)

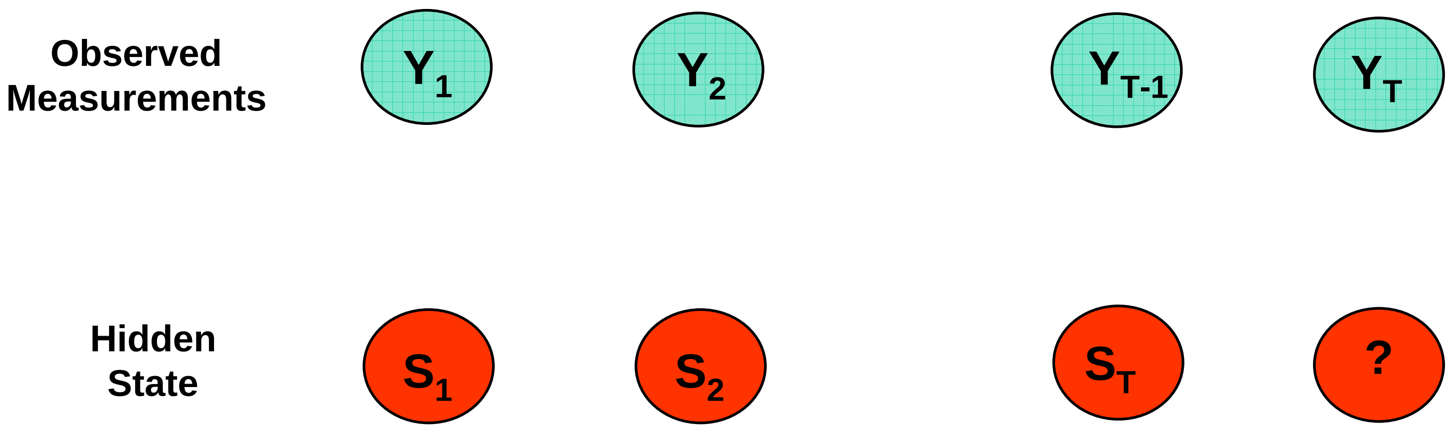
- Joint probability densities or distributions
    - e.g., p(**S**) = $p(S_1, S_2, \ldots\ldots S_T)$
    - If we know the joint density, we can compute any quantity of interest
        - …. But working with the joint density is hard

- Examples
  - **S** discrete: P(**S**) is a table containing $O(K^T)$ numbers
  - **S** continuous: P(**S**) is a function over a T-dimensional space

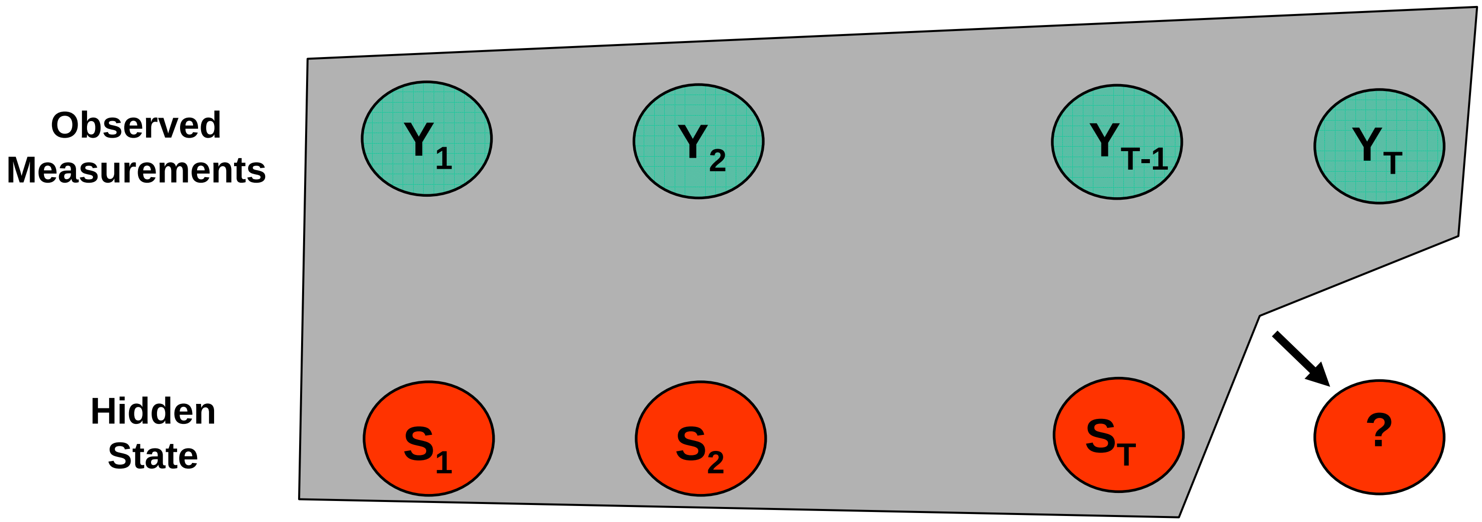
Sequential Data



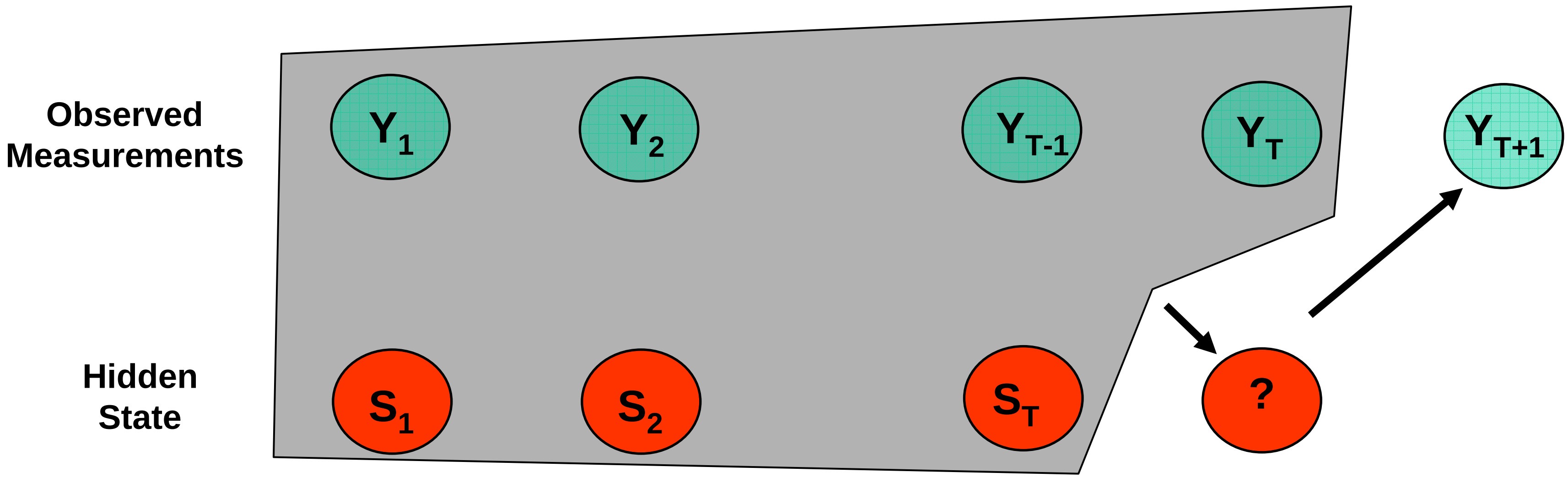
Sequential Data



Sequential Data



Sequential Data



Conditional Probabilities

- Many problems of interest involve computing conditional probabilities, densities, or expectation

  – Prediction

  $$E\ [\ y_{T+1}\ |\ y_T, \ldots\ldots\ y_1\ ]$$

  – State Estimation

  $$\textit{arg max}\ \{\ P(s_1, \ldots\ldots\ s_T\ |\ y_1, \ldots\ldots\ y_T)\ \}$$

  – Parameter Estimation

  $$P(\ \theta\ \ |\ y_1, \ldots\ldots\ y_T)$$

- Note:
  – Computing $P(S_{T+1}\ |\ \ S_1\ = s)$ has time complexity $O(K^T)$

Two Problems

- Problem 1: Computational Complexity
  - computations scale as $O(K^N)$
- Problem 2: Model Specification
  - To specify p(U) we need a table of K^N numbers
  - Where do these numbers come from?

Two Key Ideas

- Problem 1: Computational Complexity
  - Idea:
    - Represent dependency structure as a graph and exploit sparseness in computation

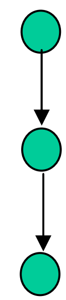
- Problem 2: Model Specification
  - Idea:
    - learn models from data using statistical learning principles

"...probability theory is more fundamentally concerned with the <u>structure</u> of reasoning and causation than with numbers."

Glenn Shafer and Judea Pearl
***Introduction to Readings in Uncertain Reasoning*, **
Morgan Kaufmann, 1990

Graphical Models

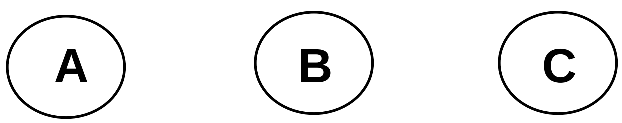
- Dependency structure encoded by an acyclic directed graph
  - Node <-> random variable
  - Edges encode dependencies
    - Absence of edge -> conditional independence
  - Directed and undirected versions



- Why is this useful?
  - A language for communication
  - A language for computation

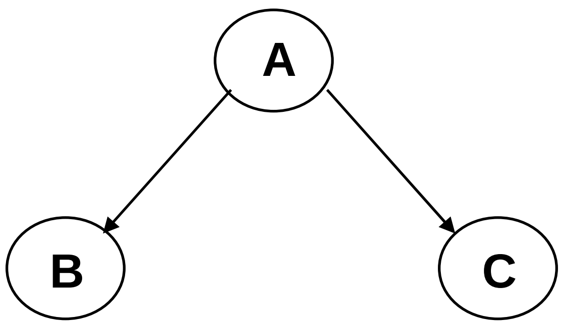
- Origins:
  - Wright 1920's
  - 1988
    - Spiegelhalter and Lauritzen in statistics
    - Pearl in computer science
  - Aka Bayesian networks, belief networks, causal networks, etc

Examples of 3-way Graphical Models



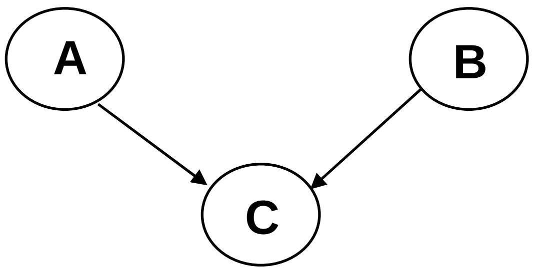
Marginal Independence:
p(A,B,C) = p(A) p(B) p(C)

Examples of 3-way Graphical Models



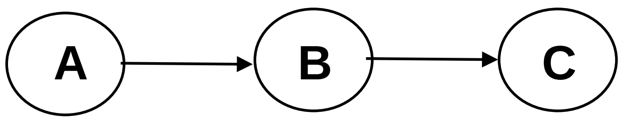
Conditionally independent effects:
p(A,B,C) = p(B|A)p(C|A)p(A)

Examples of 3-way Graphical Models



Independent Causes:
p(A,B,C) = p(C|A,B)p(A)p(B)

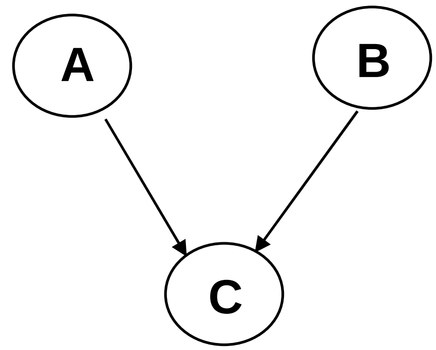
Examples of 3-way Graphical Models



Markov dependence:
p(A,B,C) = p(C|B) p(B|A)p(A)

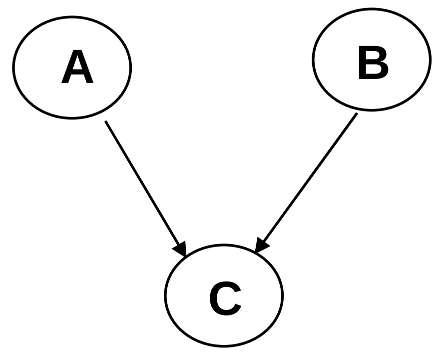
Directed Graphical Models

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



Directed Graphical Models

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

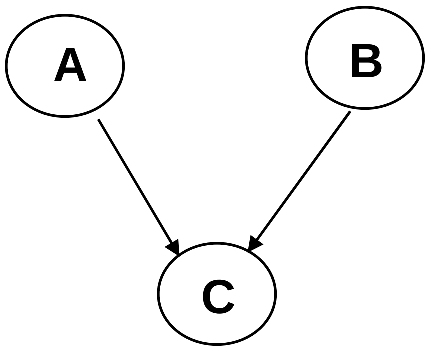


In general,

$p(X_1, X_2, \ldots. X_N) = \prod p(X_i \mid parents(X_i))$

Directed Graphical Models

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

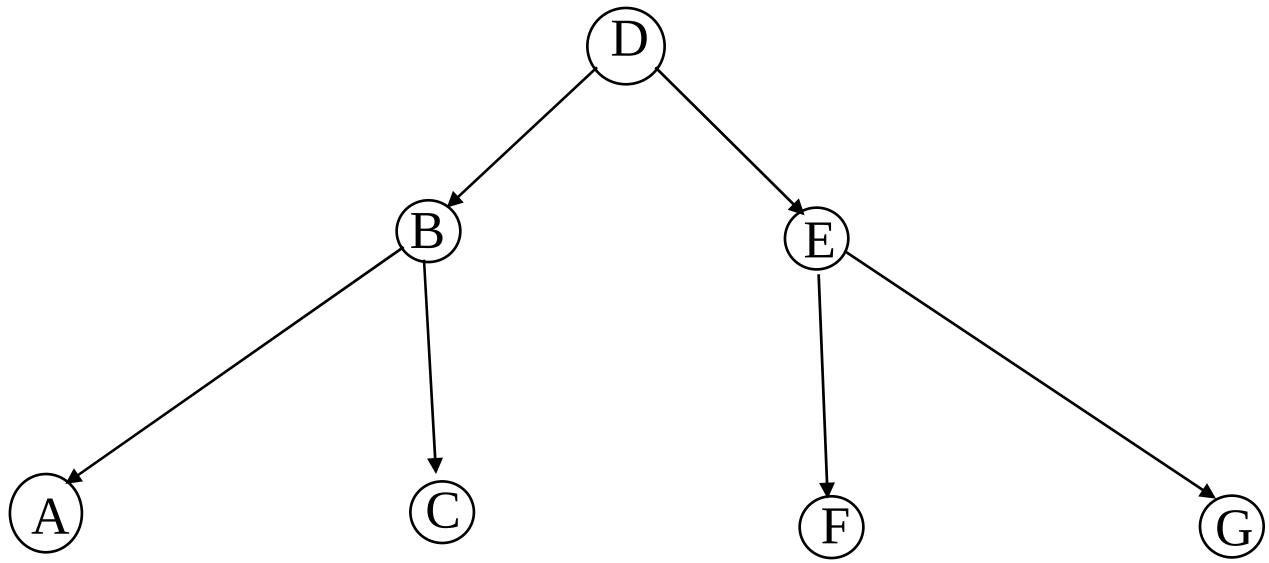


In general,

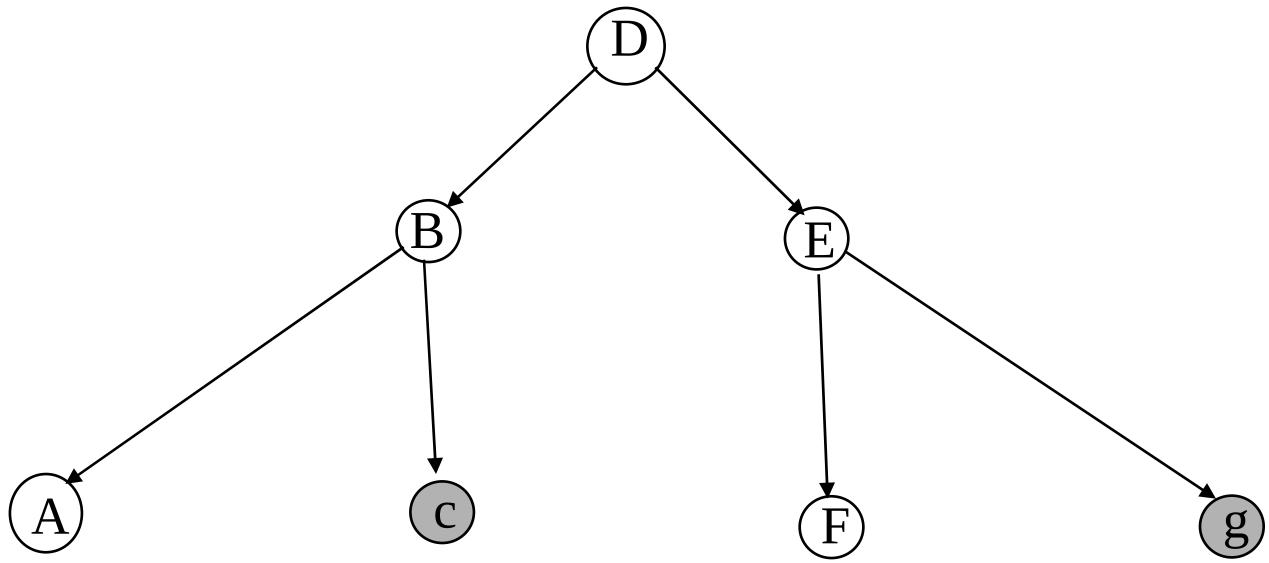
$$p(X_1, X_2,\ldots X_N) = \Pi\, p(X_i \mid \text{parents}(X_i))$$

- Probability model has simple factored form
- Directed edges => direct dependence
- Absence of an edge => conditional independence
- Also known as belief networks, Bayesian networks, causal networks

Example

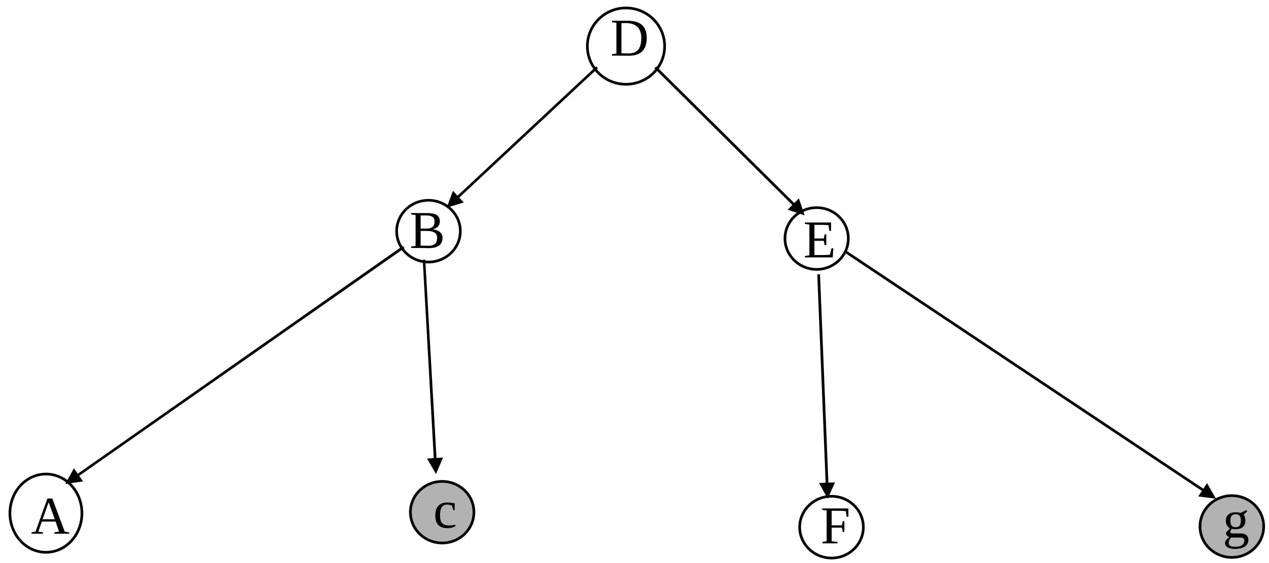


Example



Say we want to compute p(a | c, g)

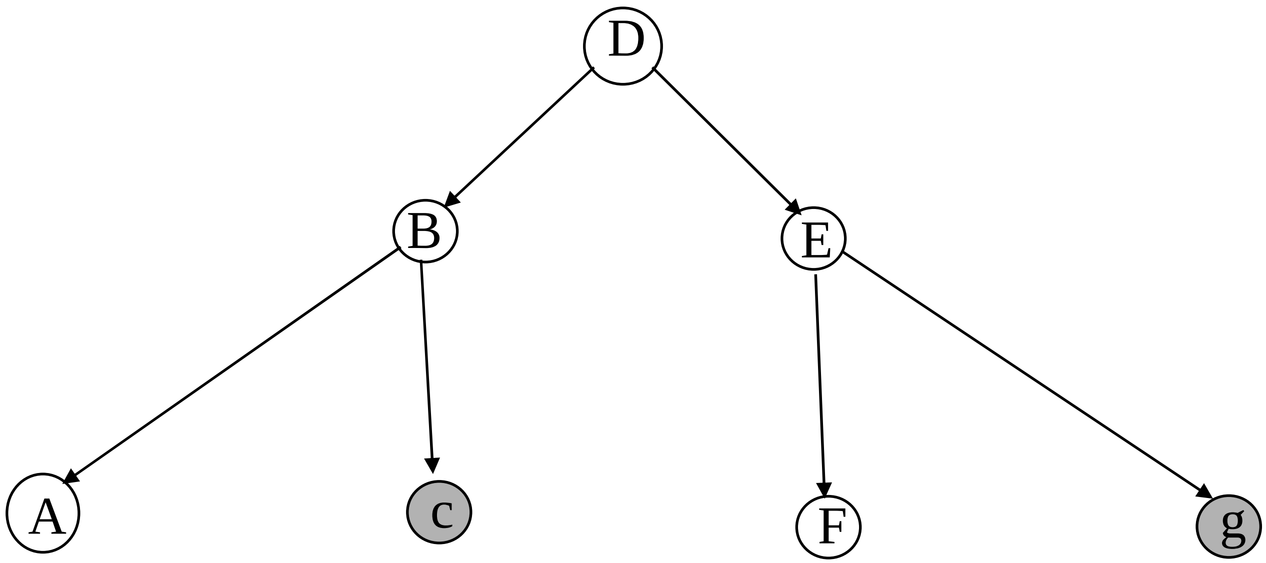
Example



Direct calculation: $p(a|c,g) = \Sigma_{bdef}\, p(a,b,d,e,f \mid c,g)$

Complexity of the sum is $O(K^4)$

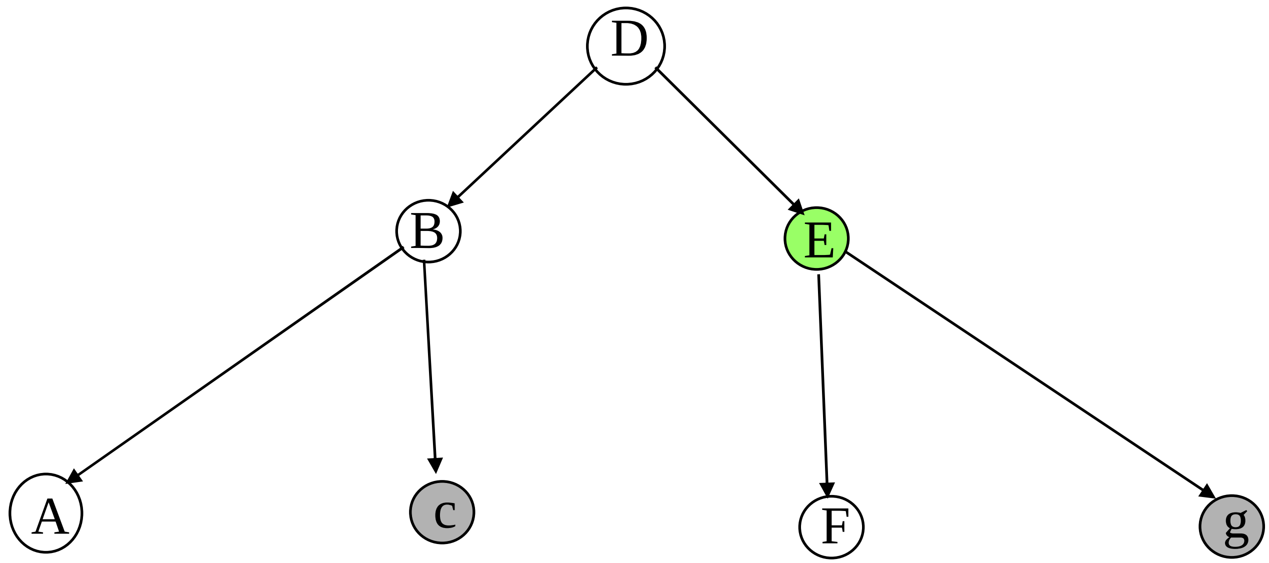
Example



Reordering:

$$\Sigma_d\, p(a|b)\, \Sigma_d\, p(b|d,c)\, \Sigma_e\, p(d|e)\, \Sigma_f\, p(e,f\,|g)$$

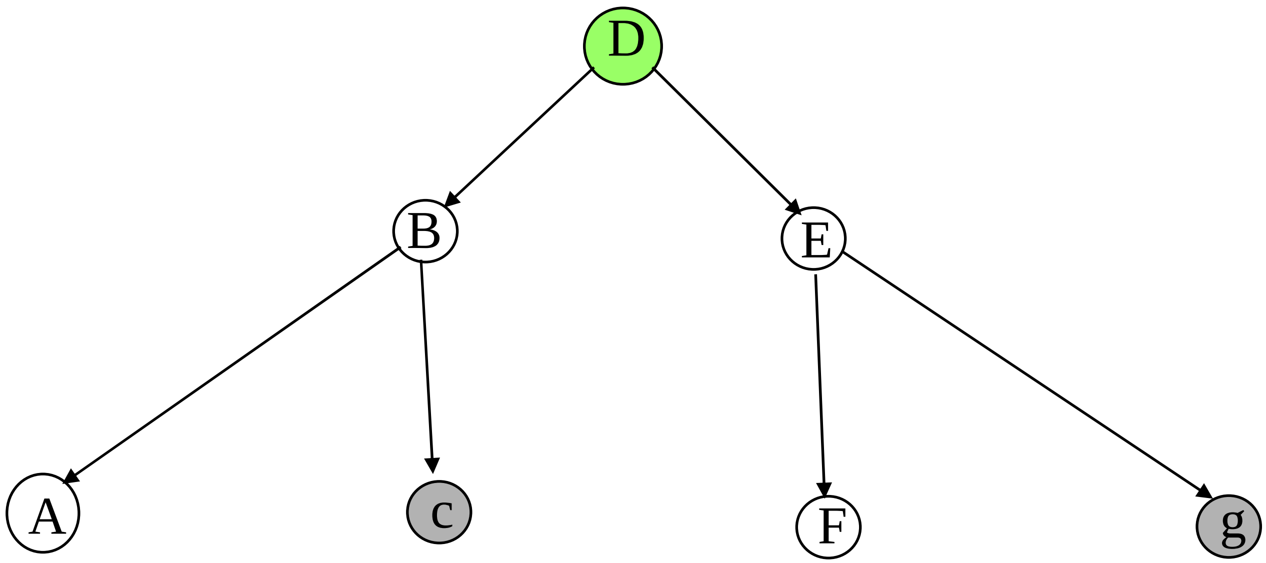
Example



Reordering:

$$\Sigma_b\ p(a|b)\ \Sigma_d\ p(b|d,c)\ \Sigma_e\ p(d|e)\ \underbrace{\Sigma_f\ p(e,f\ |g)}_{p(e|g)}$$

Example

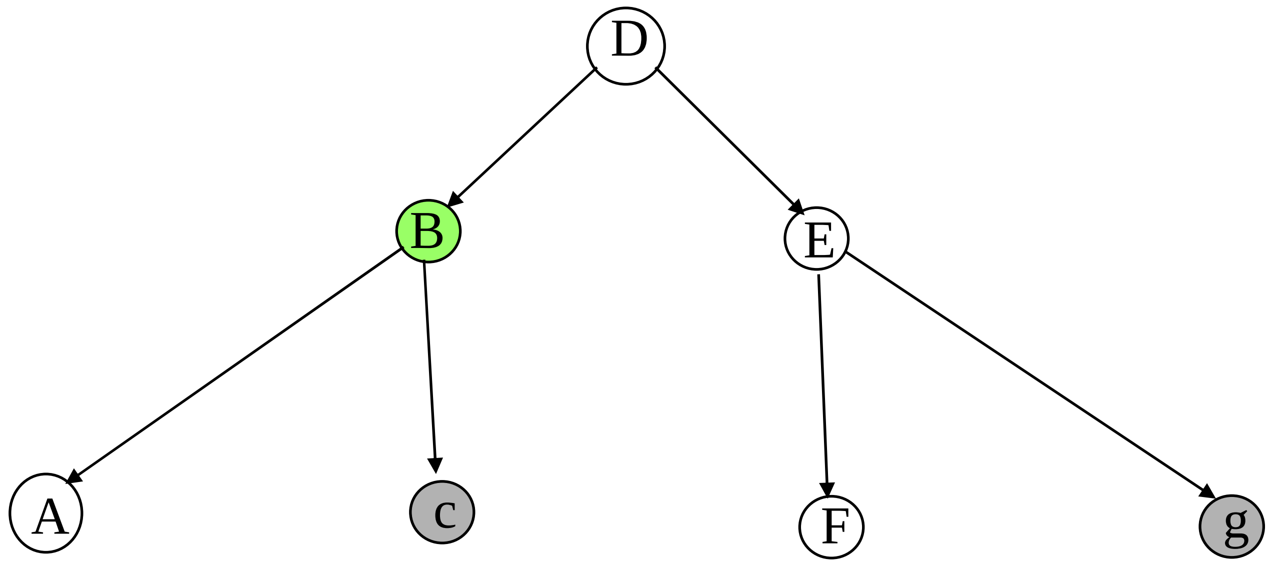


Reordering:

$$\Sigma_b \; p(a|b) \; \Sigma_d \; p(b|d,c) \; \Sigma_e \; p(d|e) \; p(e|g)$$

p(d|g)

Example

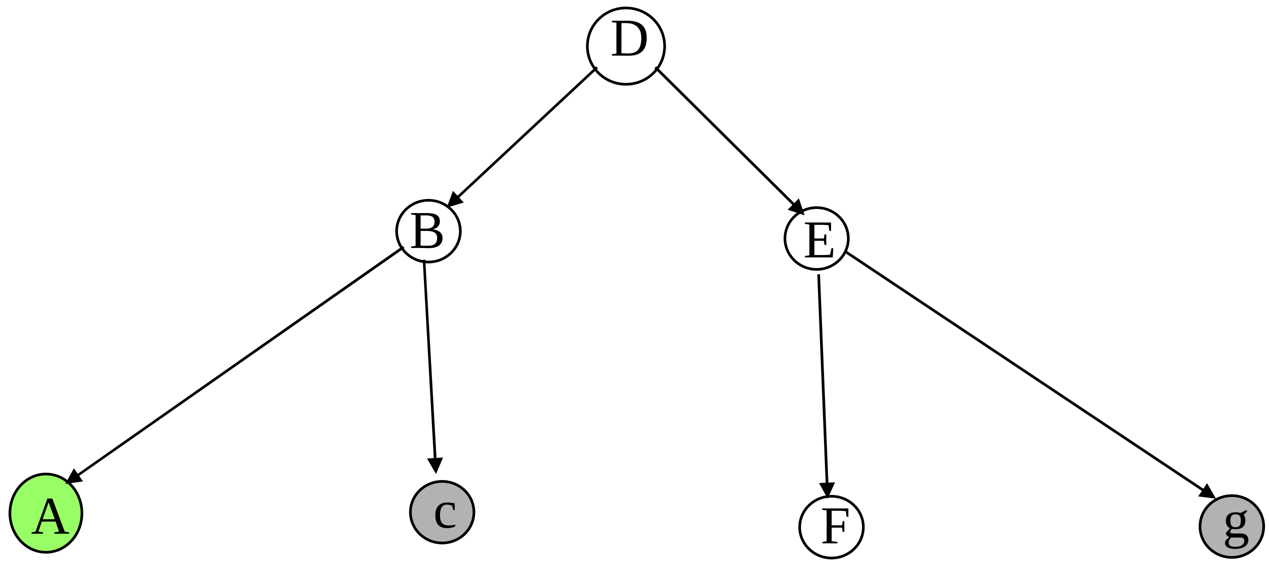


Reordering:

$$\Sigma_b \; p(a|b) \; \Sigma_d \; p(b|d,c) \; p(d|g)$$

p(b|c,g)

Example



Reordering:

$\sum_b p(a|b)\, p(b|c,g)$

$p(a|c,g)$

Complexity is $O(K)$, compared to $O(K^4)$

Probability Calculations on Graphs

- Structure of the graph -> reveals order in which variables can be eliminated

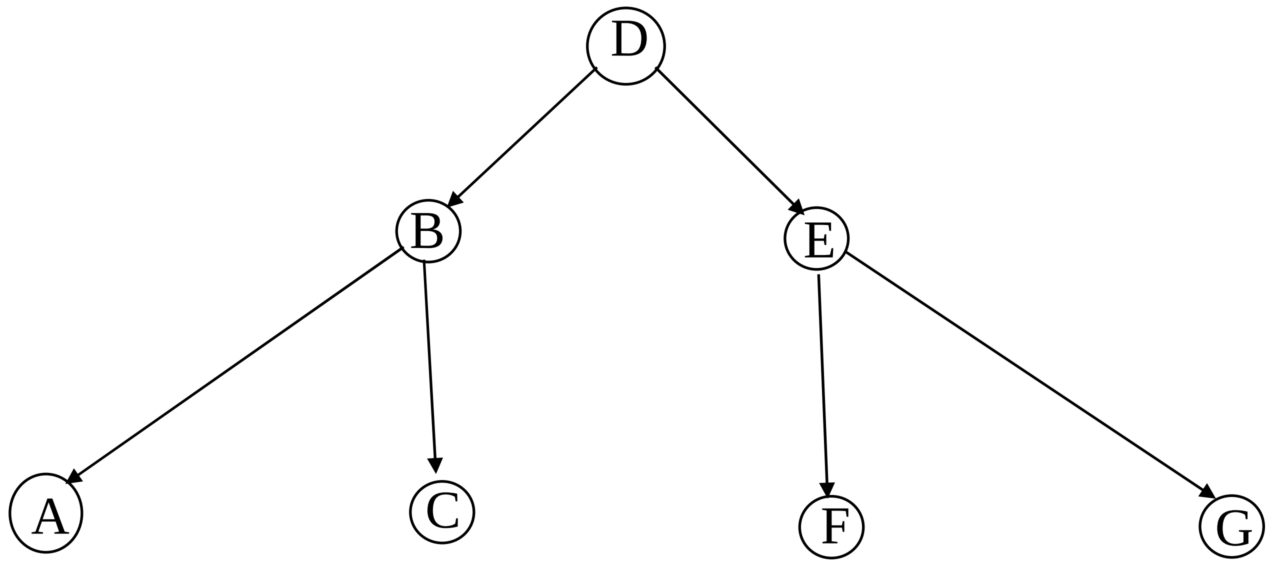
- Complexity is typically $O(K^{\text{max(number of parents)}})$
  - If single parents (e.g., tree), -> O(K)
  - The sparser the graph the lower the complexity

- Technique can be “automated”
  - i.e., a fully general algorithm for arbitrary graphs
  - For continuous variables:
    - replace sum with integral
  - For identification of most likely values
    - Replace sum with max operator

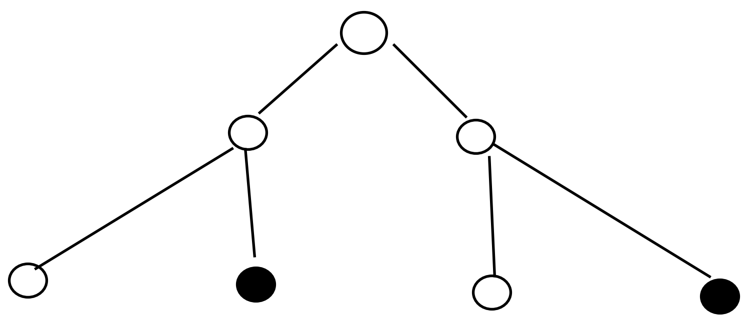
Inference in Graphical Models

- "Inference" = calculating p(one variable | values of others)
- Assume the graph has no loops after arrows are "dropped"
- Message Passing (MP) Algorithm
  - Pearl, 1988; Lauritzen and Spiegelhalter, 1988
  - Declare 1 node (any node) to be a root
  - Schedule two phases of message-passing
    - nodes pass messages up to the root
    - messages are distributed back to the leaves
  - In time O(N), we can compute P(....)

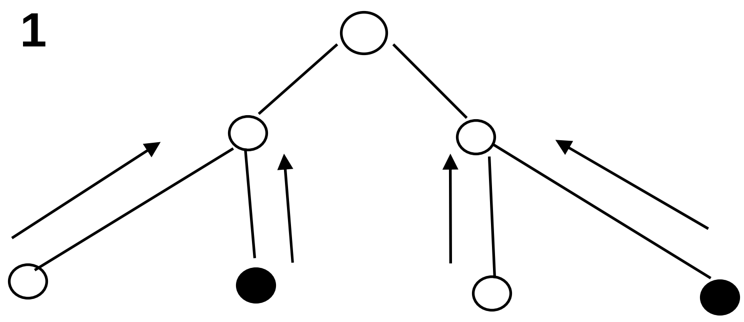
Example



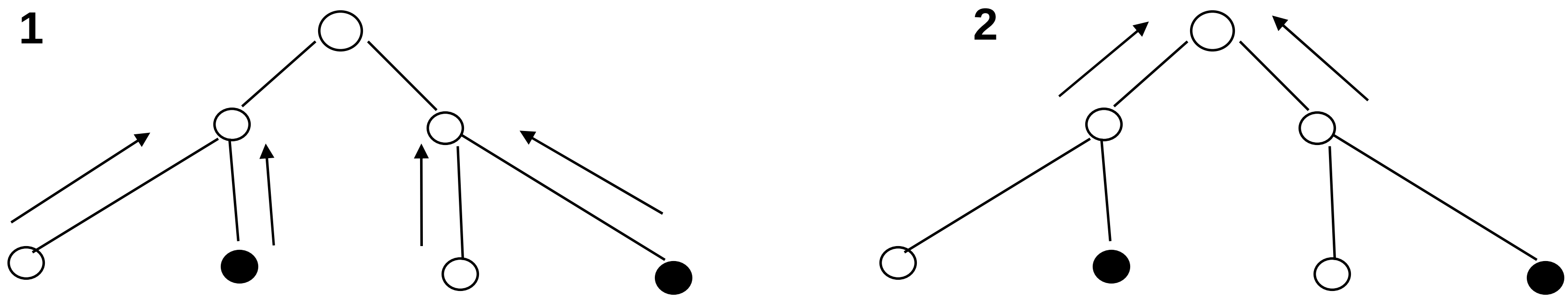
Sketch of the MP algorithm in action



Sketch of the MP algorithm in action

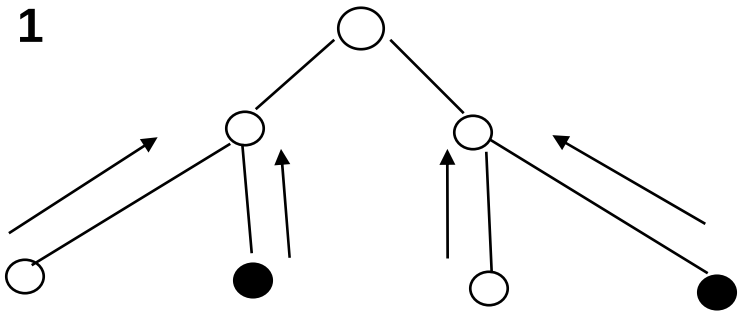


Sketch of the MP algorithm in action

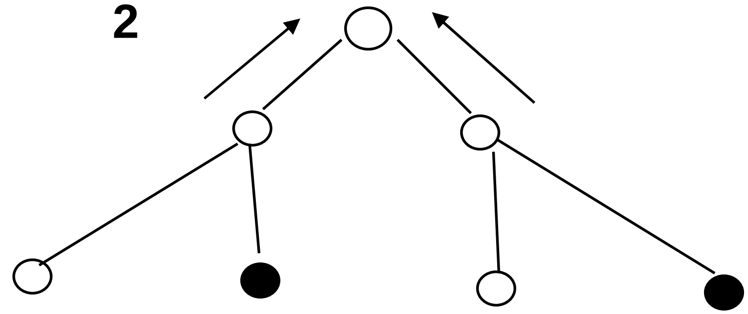


Sketch of the MP algorithm in action

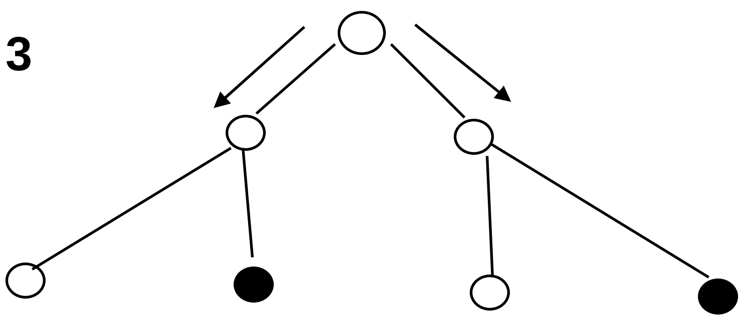
1



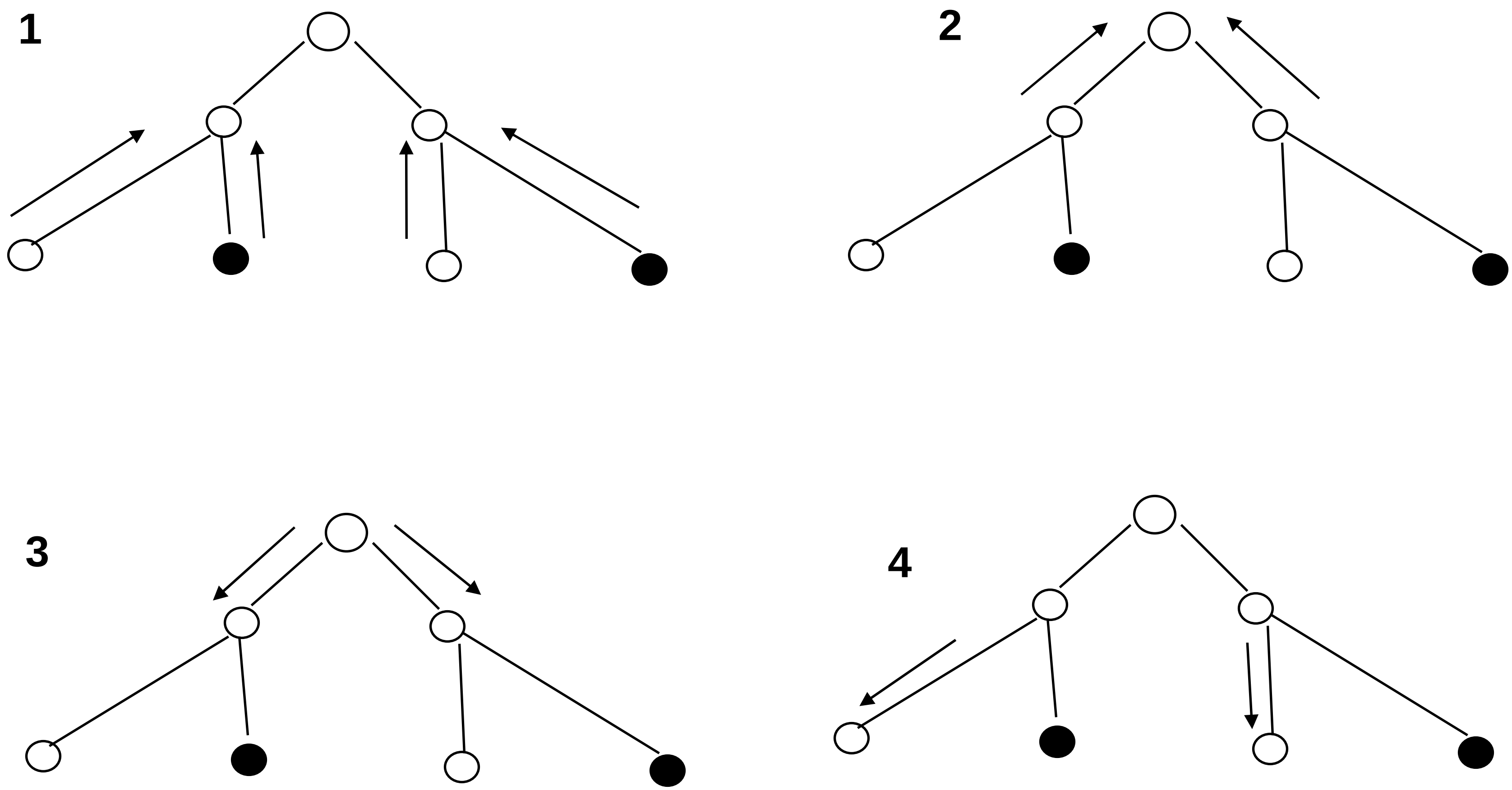
2



3



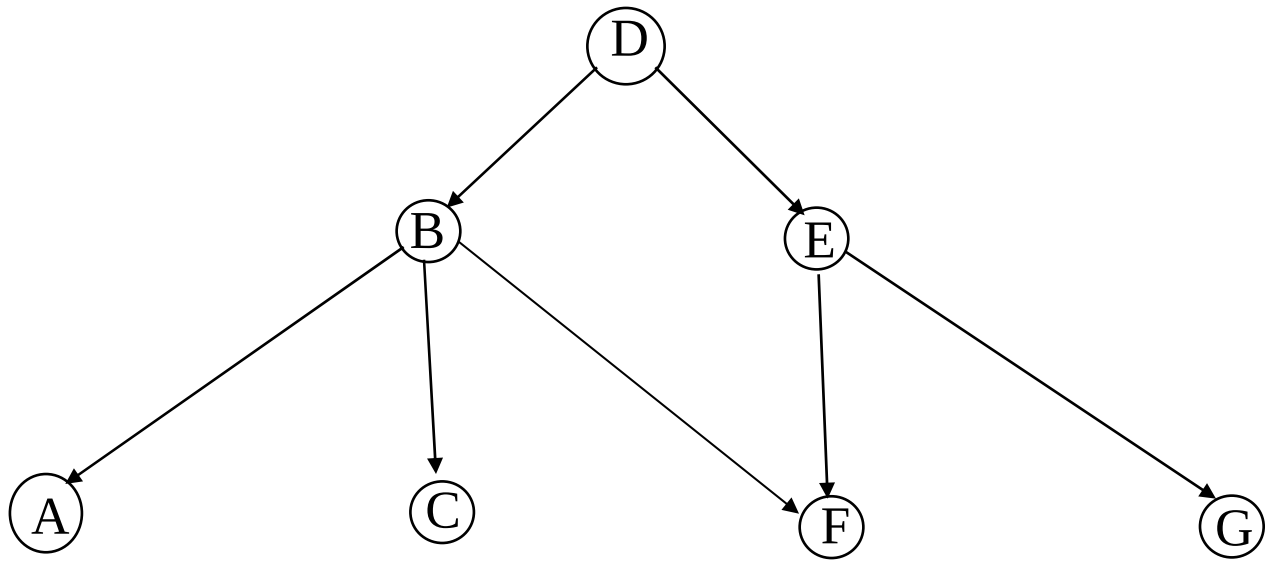
Sketch of the MP algorithm in action



Complexity of the MP Algorithm

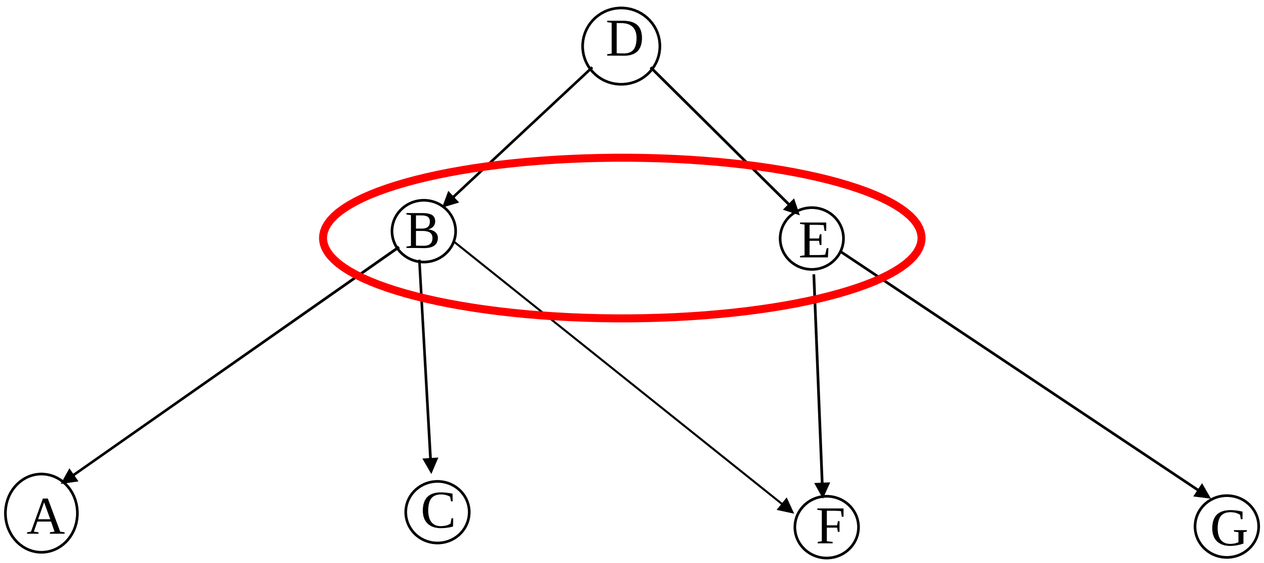
- Efficient
  - Complexity scales as $O(N K^{m})$
    - N = number of variables
    - K = arity of variables
    - m = maximum number of parents for any node
  - Compare to $O(K^N)$ for brute-force method

Graphs with “loops”



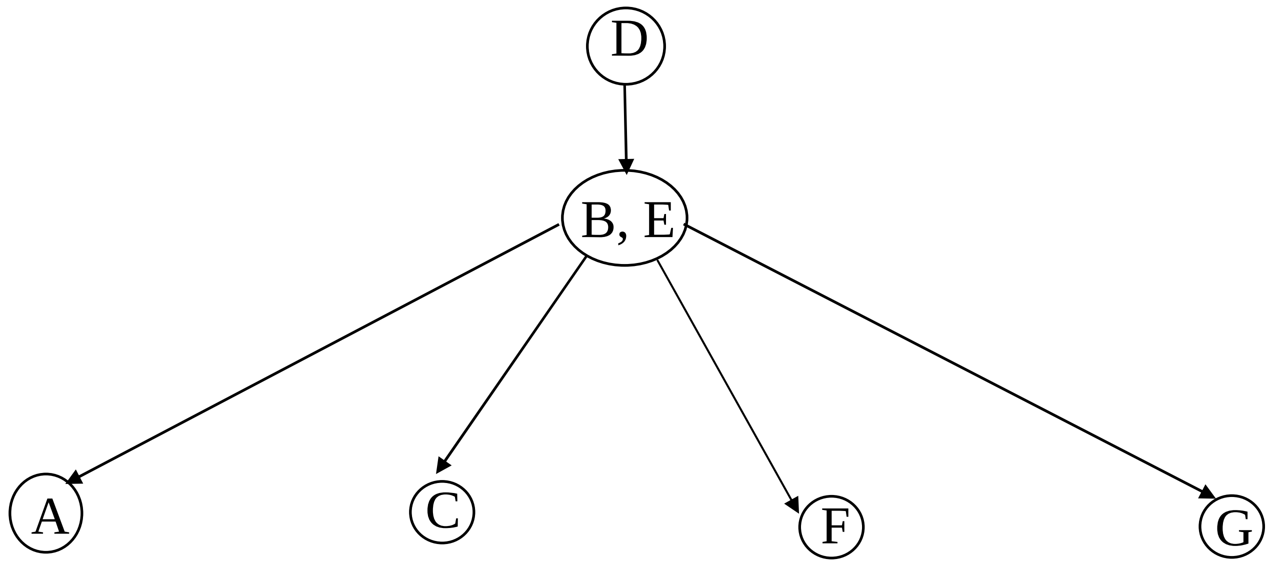
Message passing algorithm does not work when there are multiple paths between 2 nodes

Graphs with “loops”

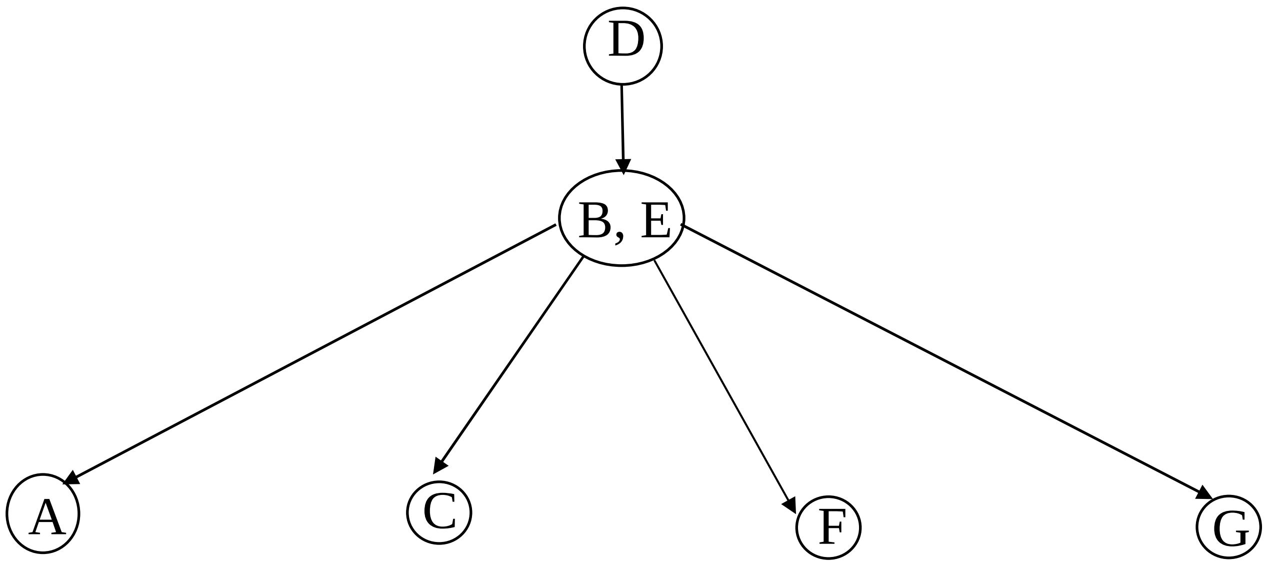


General approach: “cluster” variables together to convert graph to a tree

Junction Tree



Junction Tree



Good news: can perform MP algorithm on this tree

Bad news: complexity is now $O(K^2)$

Additional Topics

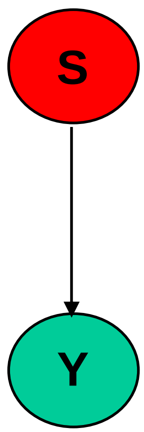
- Continuous-valued variables
  - Gaussian models
    - Tractable closed-form updating equations
  - Non-parametric models (kernel density)
    - Efficient algorithms exist for sparse graphs

- Undirected graphs:
  - Similar representation and semantics
  - Special case: Markov random field (Ising model)
    - Inference in general is intractable

Hidden Variable Models

Mixture Models

$$p(Y) = \Sigma_k \, p(Y \mid S=k) \, p(S=k)$$



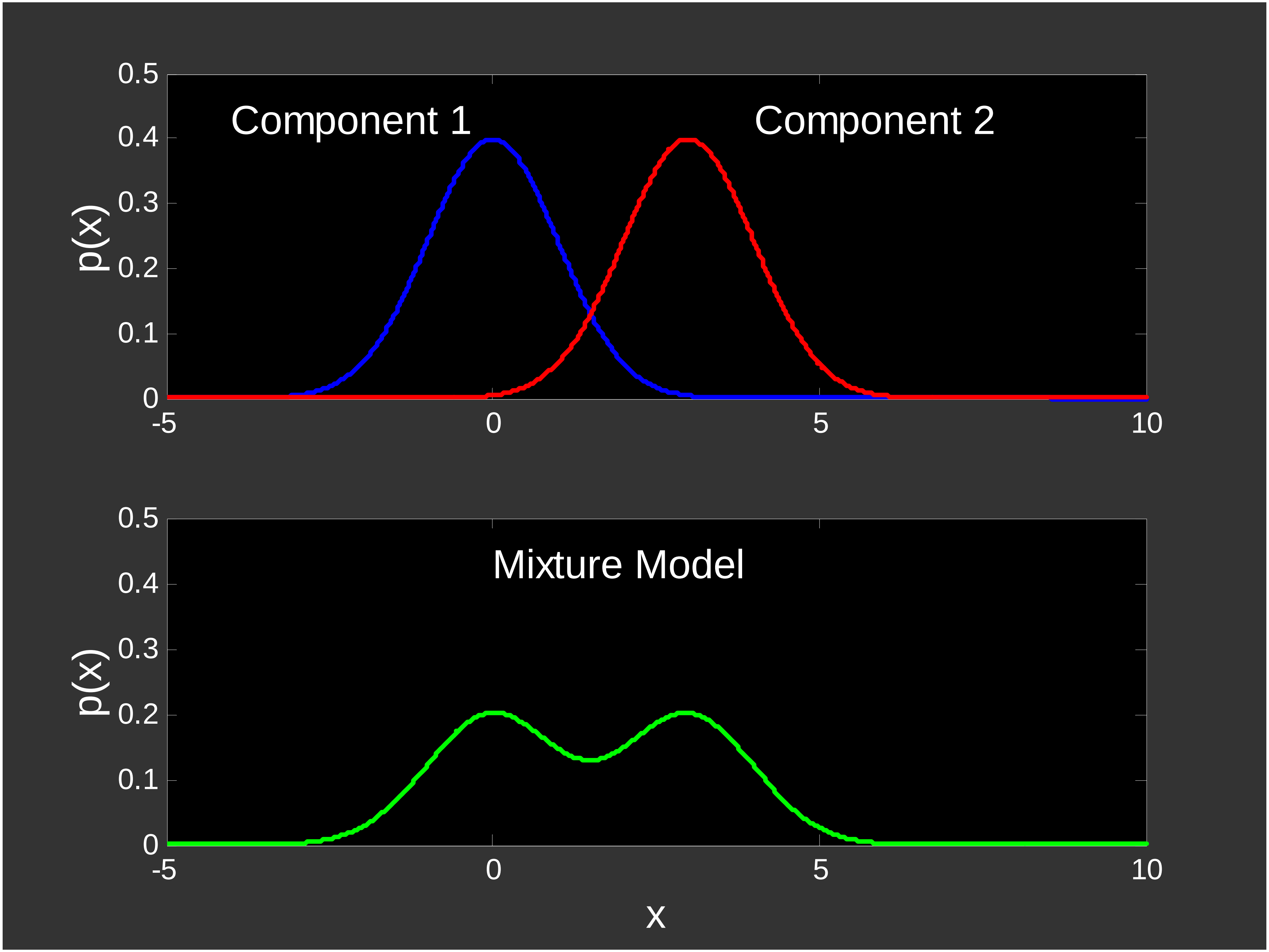
Hidden discrete variable

Observed variable(s)

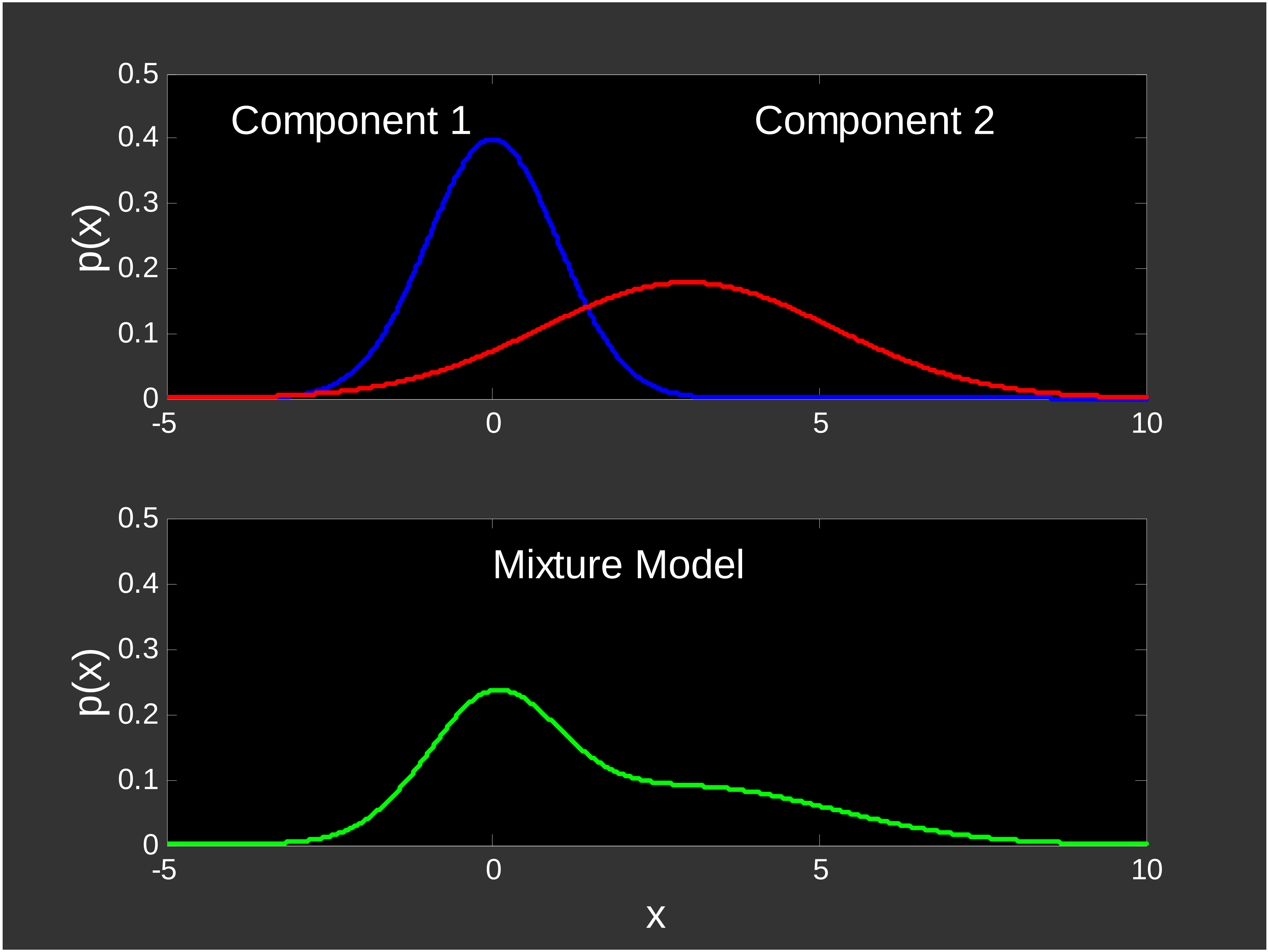
Motivation:

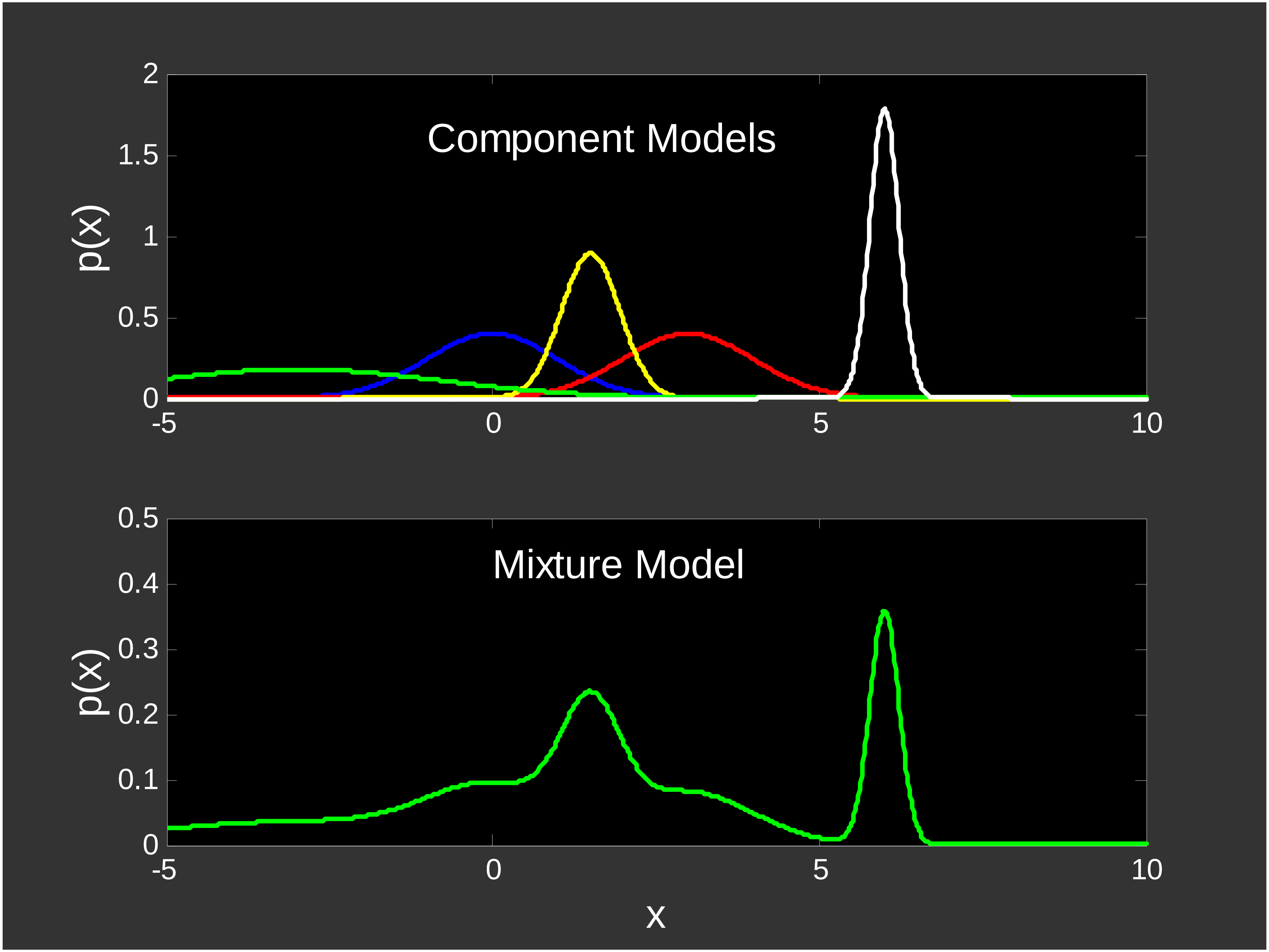
1. models a true process (e.g., fish example)

2. approximate state-based representation
(e.g., regimes in climate data)

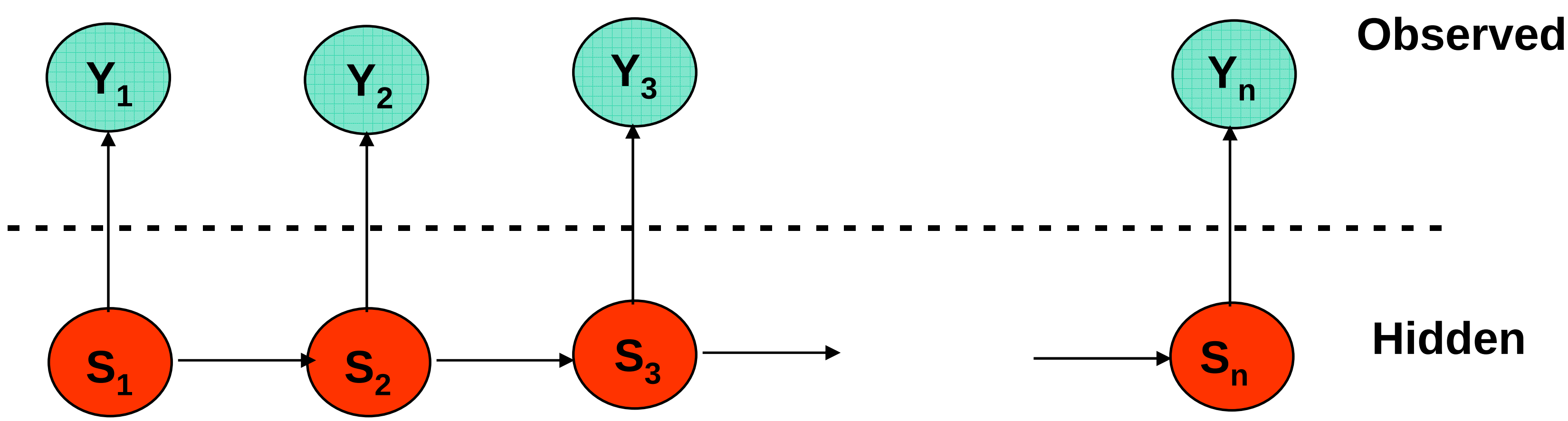


Component 1
Component 2
Mixture Model
p(x)
p(x)
x
0.5
0.4
0.3
0.2
0.1
0
-5
0
5
10

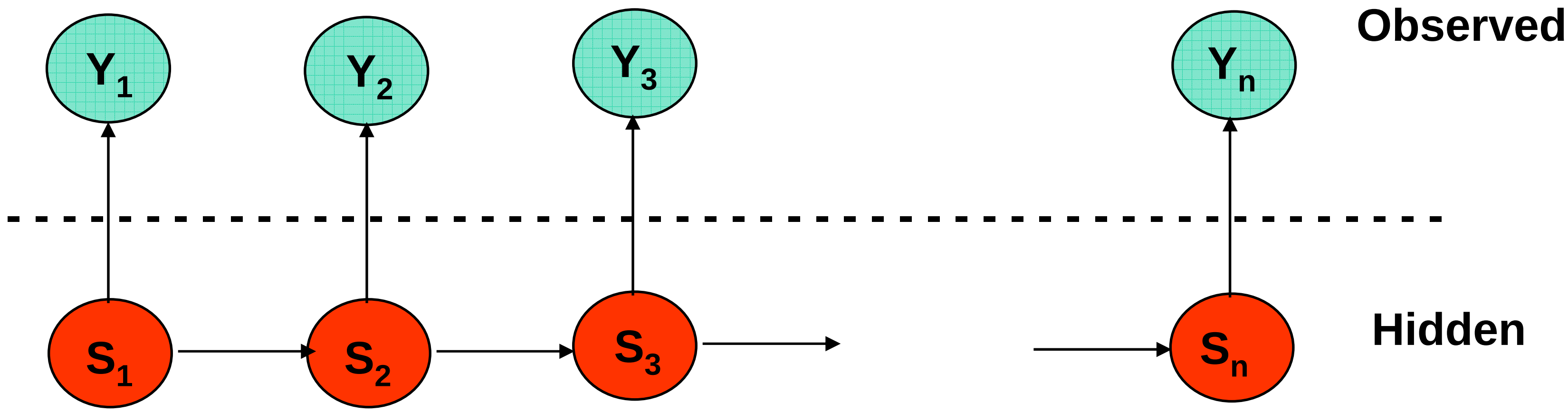

Component 1
Component 2
Mixture Model
p(x)
p(x)
x
0
0.1
0.2
0.3
0.4
0.5
-5
0
5
10


Component Models
p(x)
2
1.5
1
0.5
0
-5
0
5
10
Mixture Model
p(x)
0.5
0.4
0.3
0.2
0.1
0
-5
0
5
10
x

Hidden Markov Model (HMM)



Hidden Markov Model (HMM)



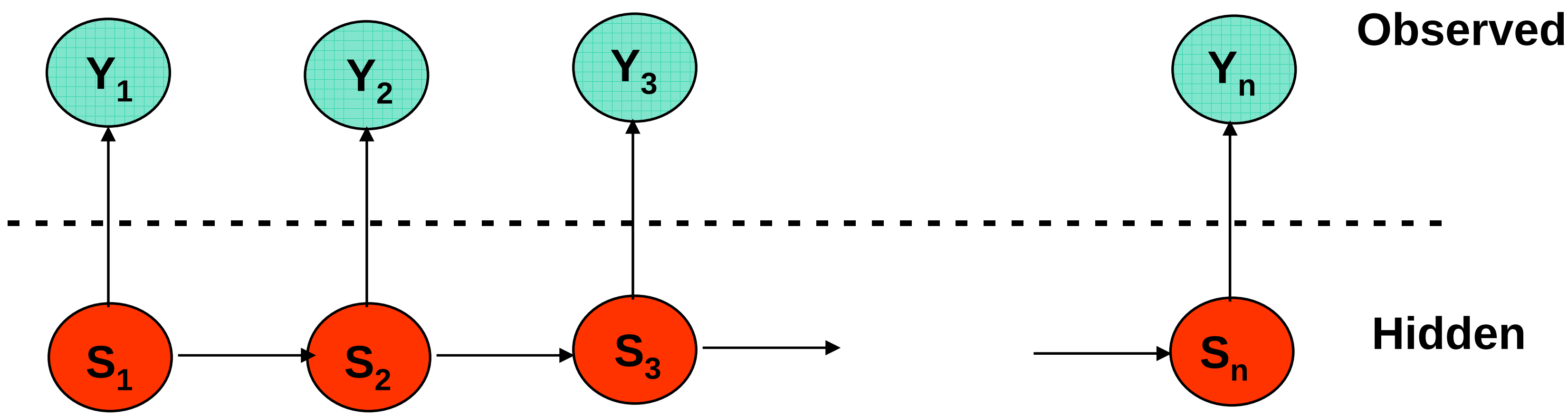
Two key independence assumptions

$$P(s_1, \ldots s_n, y_1, \ldots y_n) = \Pi \, p(s_t \mid s_{t-1}) \, p(y_i \mid s_i)$$

State dynamics

Observation model

Comments on HMMs



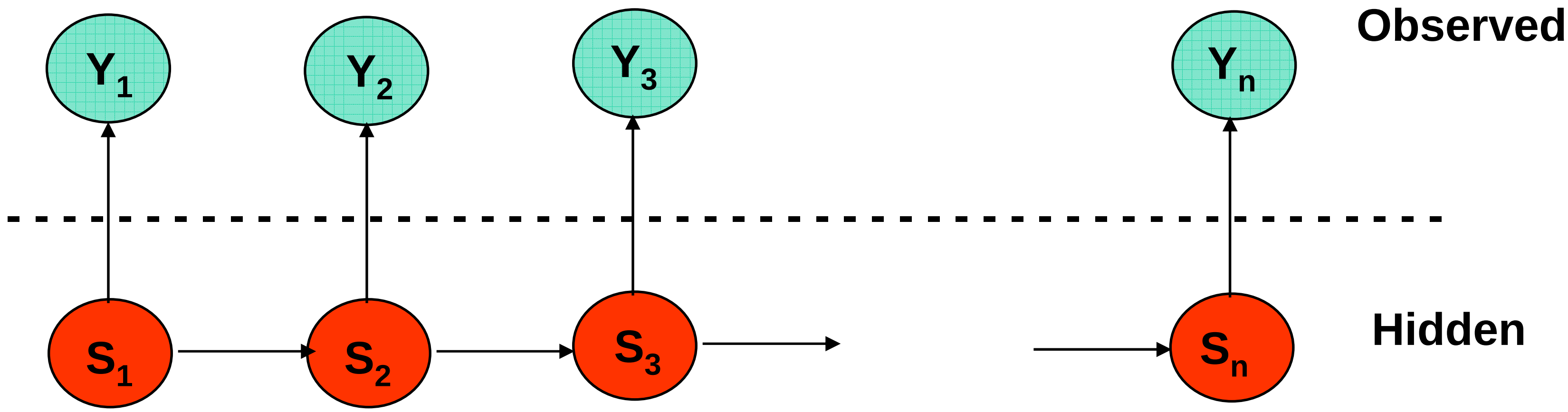
Motivation?

- S discrete:
  - -> can provide non-linear switching
  - -> can encode low-dim time-dependence for high-dim Y

- S is continuous, Gaussian dependencies, we have a Kalman filter

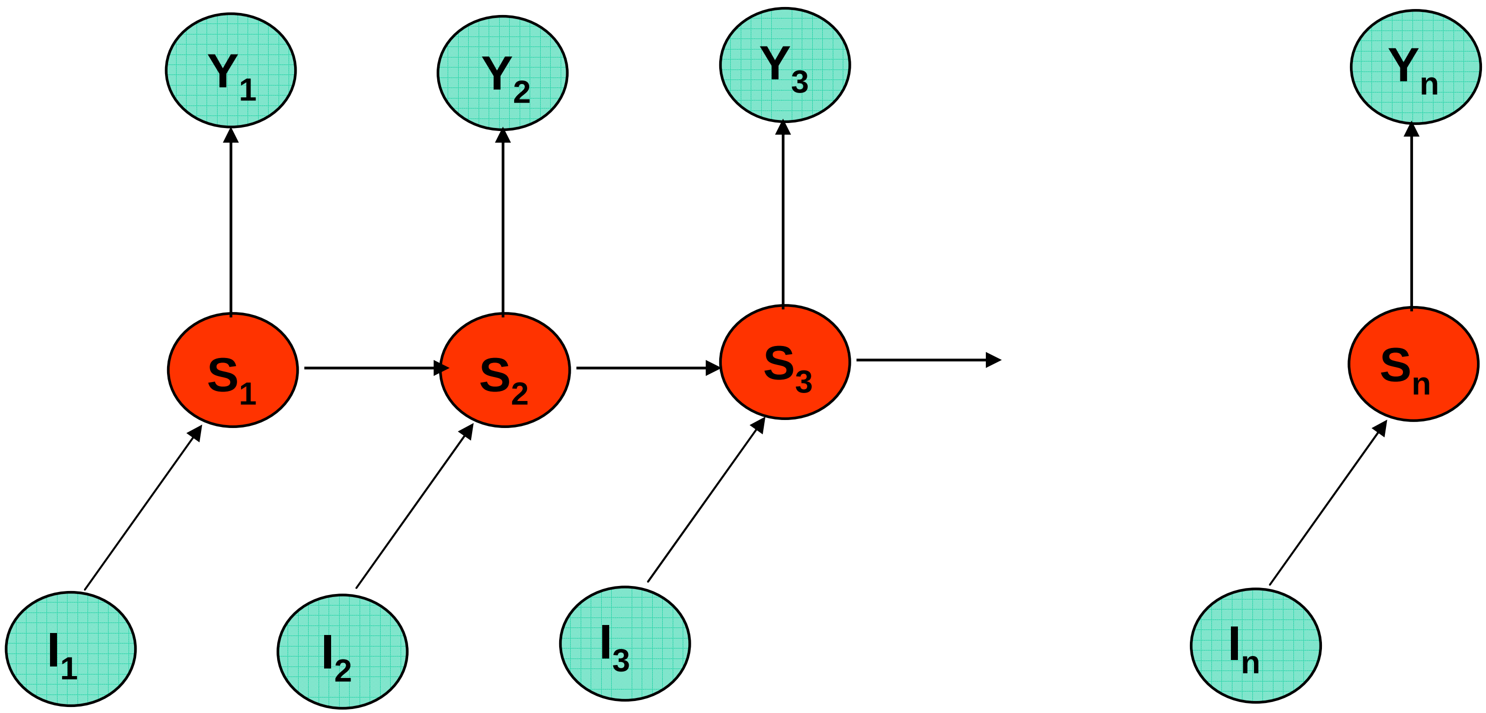
Widely used in speech recognition, protein sequence modeling, …

Probability Computation



- Computing $p(S_n \mid y_1, \ldots y_n)$
  - Recursively compute
    - $p(S_1 \mid y_1)$
    - $p(S_2 \mid y_2, S_1)$ weighted by $p(S_1 \mid y_1)$
    - and so on..
  - This is the MP algorithm, with S_1 as the root node

Generalizing HMMs



Inputs I provide context to influence switching, e.g., downscaling

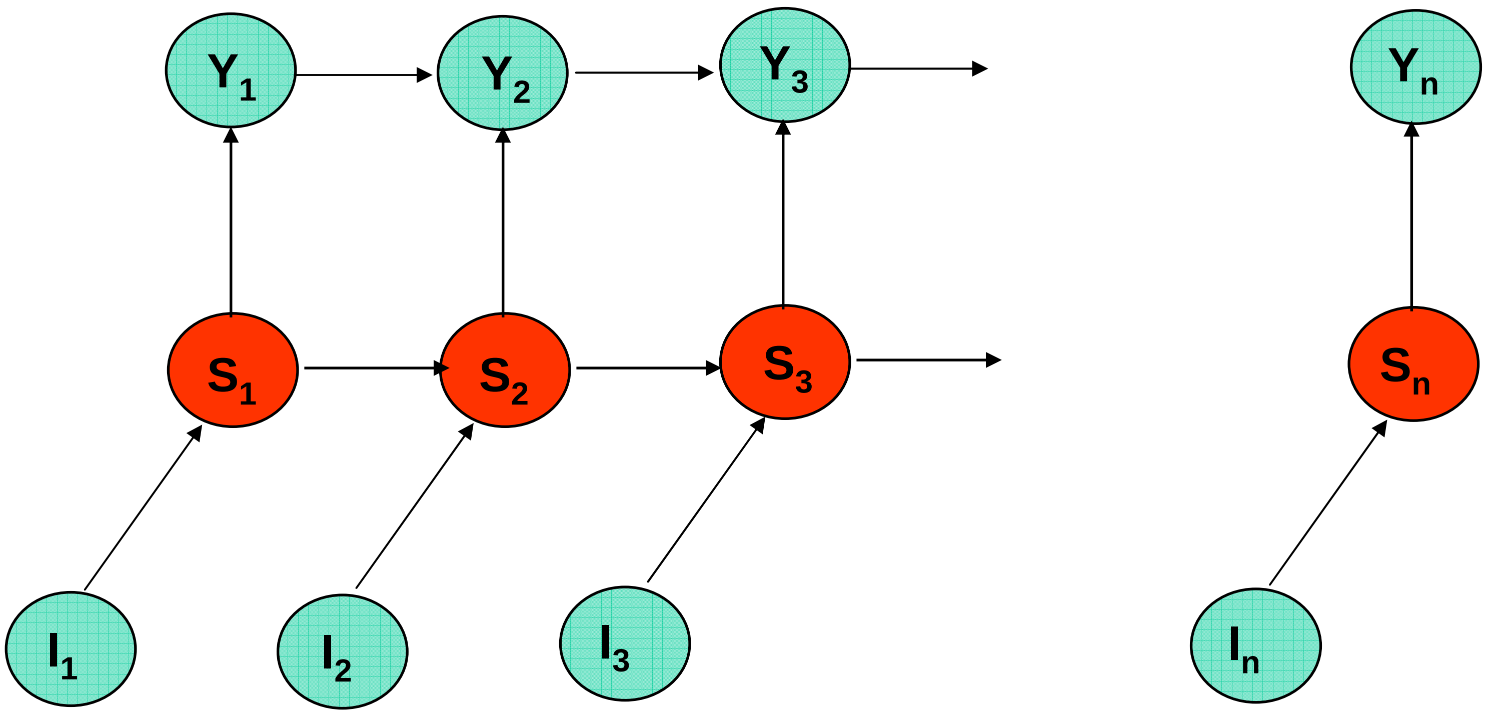
I = observed atmospheric measurements

S = "weather regimes"

Y = observed rainfall (Guttorp and Charles, 1994)

Model is still a tree -> inference is still linear

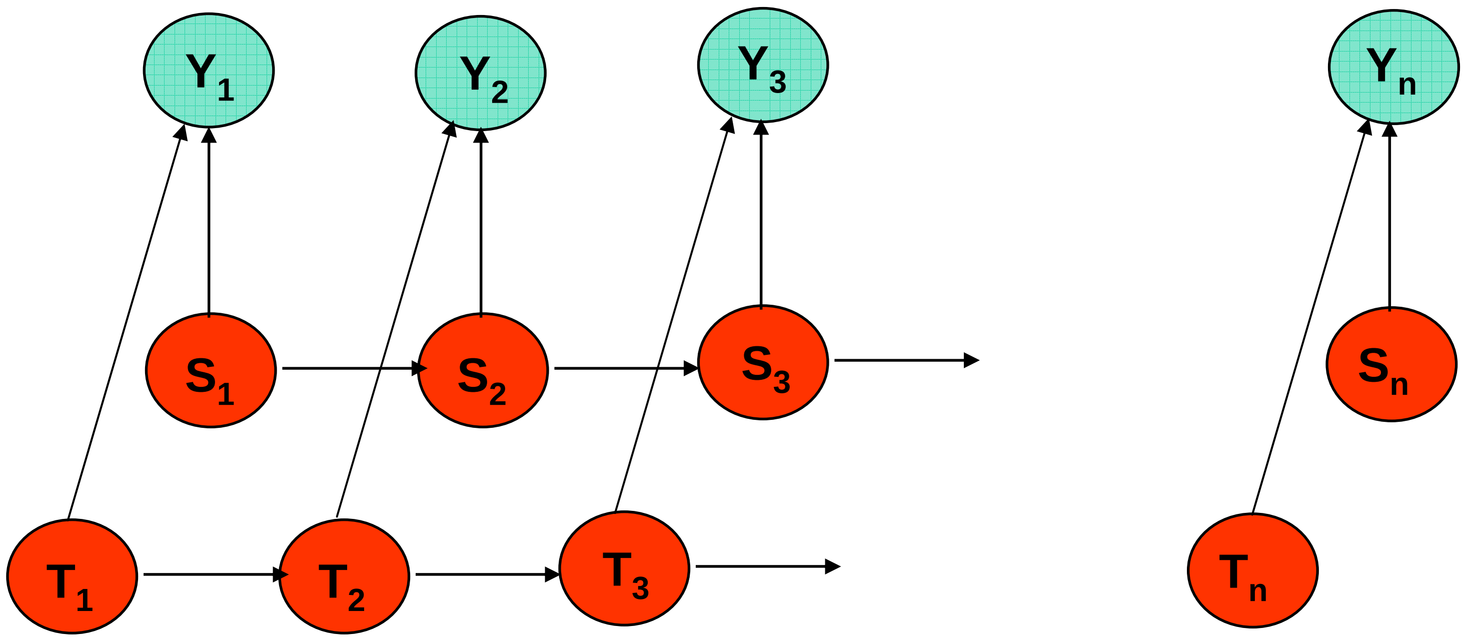
Generalizing HMMs



Add direct dependence between Y's to better model persistence

Can merge each S_t and Y_t to construct a junction tree

Generalizing HMMs



Two independent state variables,
e.g., two processes evolving at different time-scales

Comments on HMMs

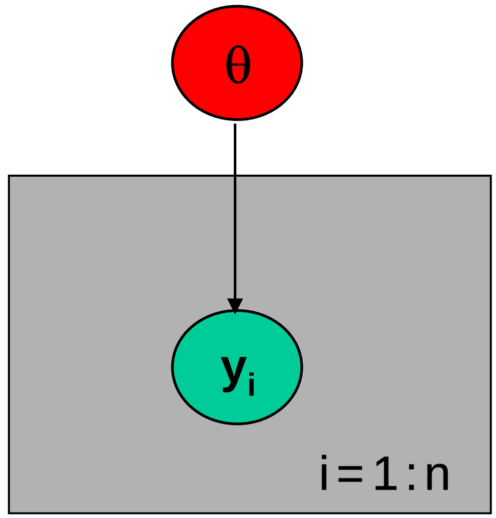
- Non-Gaussian state-space models
  - Non-linear dynamical model for $p(s_t \mid s_{t-1})$
  - Complicates probability calculations and estimation

- Integrating different measurements
  - y variables can include, e.g., remote-sensing, station data,
  - Conditional independence for $p(y_i \mid s_i)$

- Handling missing data
  - e.g., missing measurements y (station data)
  - average over missing data, conditioned on observed data – calculations are straightforward

Learning Model Parameters from Data

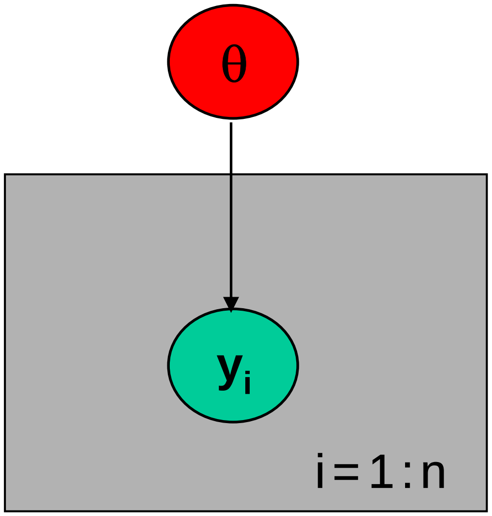
Data and Plates



Model parameters

Data = $\{y_1, \ldots y_n\}$

Maximum Likelihood



Model parameters

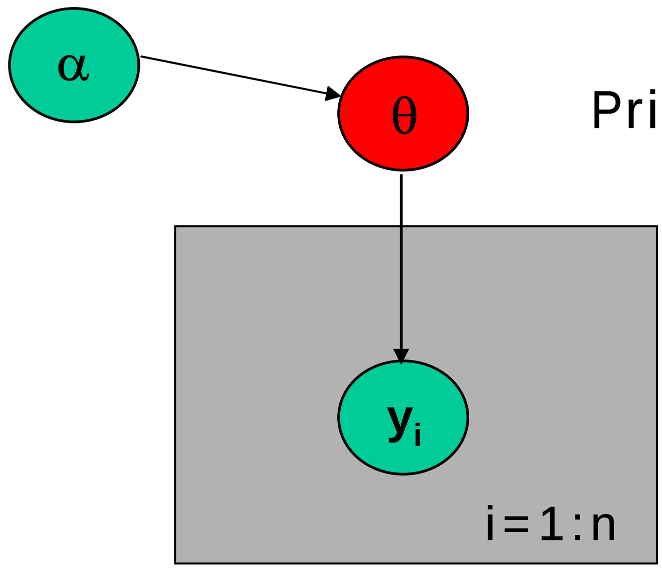
Data = $\{y_1, \ldots y_n\}$

Likelihood(θ) = p(Data | θ) = Π p(y_i | θ)

Maximum Likelihood:

$$\theta_{ML} = \arg\max\{ \text{Likelihood}(\theta) \}$$

Bayesian Estimation



Prior(θ) = p(θ | α)

Maximum A Posteriori:

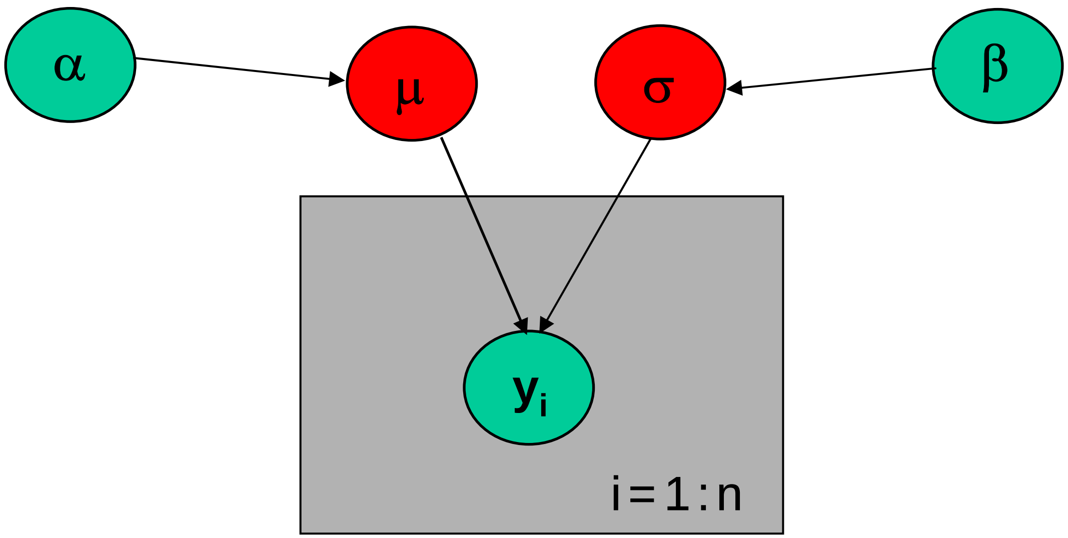
$$\theta_{MAP} = \arg\max\{ \text{Likelihood}(\theta) \times \text{Prior}(\theta) \}$$

Fully Bayesian:

$$p(\theta \mid \text{Data}) = p(\text{Data} \mid \theta)\, p(\theta) \,/\, p(\text{Data})$$

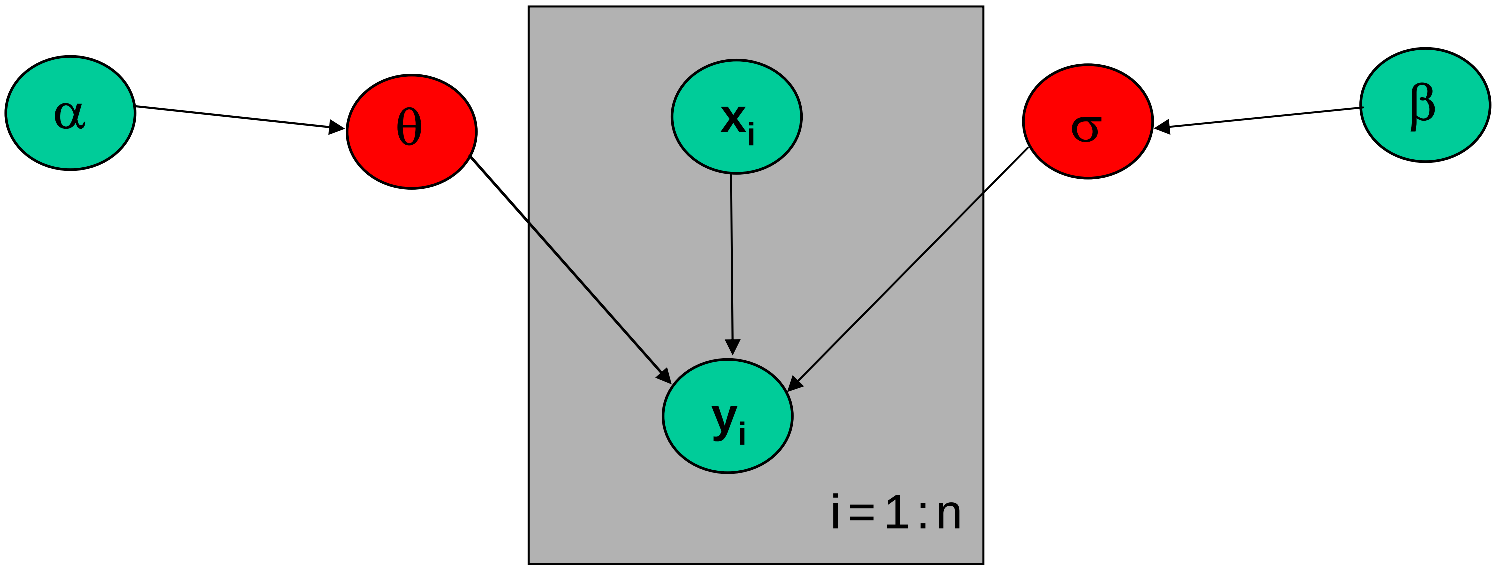
Note: “learning” <-> inference in a graphical model

Example: Gaussian Model



Note: priors and parameters are assumed independent here

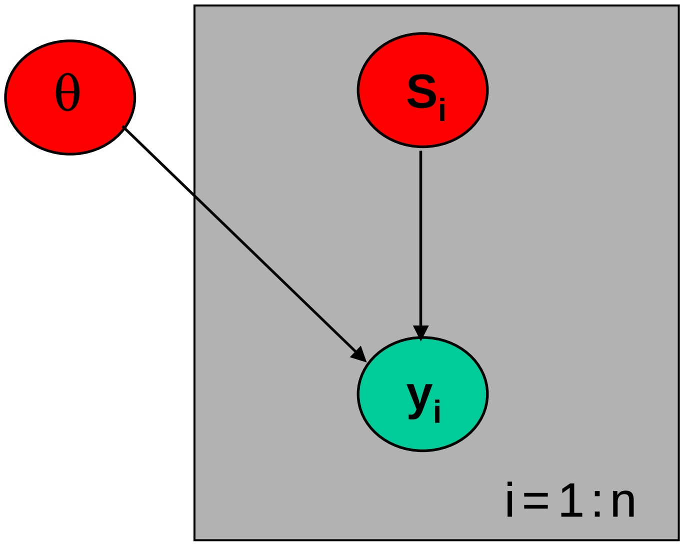
Example: Bayesian Regression



Model: $y_i = f\,[x_i;\theta] + e, \quad e \sim N(0, \sigma^2)$

$$p(y_i \mid x_i) = N\left(f[x_i;\theta]\ ,\ \sigma^2 \right)$$

Mixture Model



$$\text{Likelihood}(\theta)\ p(\theta) = p(\text{Data} \mid \theta)\ p(\theta)$$

$$= p(\theta) \prod_i p(y_i \mid \theta)$$

$$= p(\theta) \prod_i \left[\Sigma_k\, p(y_i \mid s_i = k, \theta)\ p(s_i = k) \right]$$

Estimation with Missing Data

Dempster, Laird, Rubin, 1977

- Guess at some initial parameters θ^0

- E-step
  - For each case, and each unknown variable compute

    $p(S \mid \text{known data}, \theta^0)$

- M-step:
  - Maximize $p(\theta \mid \text{data})$ using $p(S \mid \ldots..)$
  - This yields new parameter estimates θ^1

- This is the EM algorithm:
  - converges to a (local) maximum of $p(\theta \mid \text{data})$

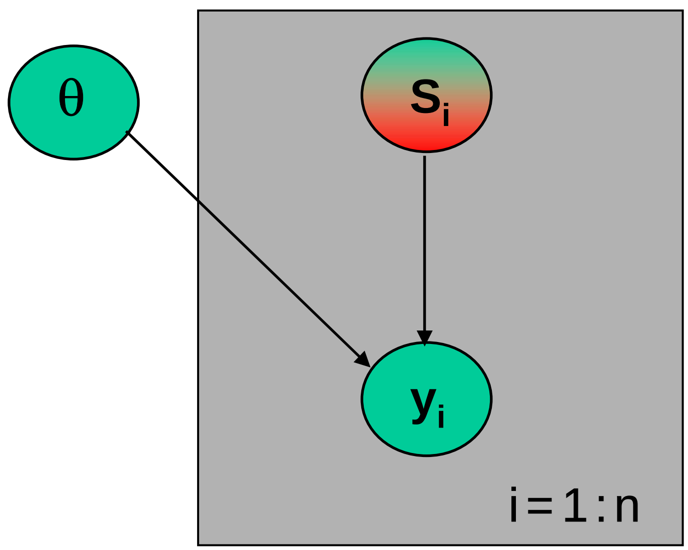
Estimation with Missing Data

Dempster, Laird, Rubin, 1977

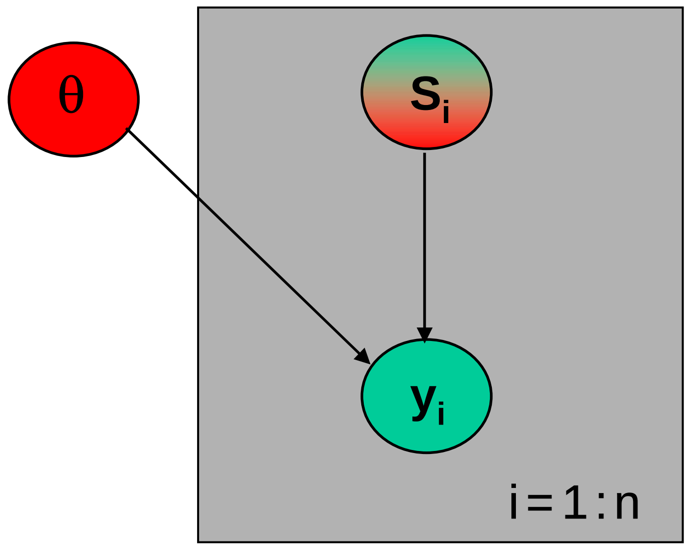
- Guess at some initial parameters θ^0
- E-step (Computation in a graph)
  - For each case, and each unknown variable compute

    $p(S \mid \text{known data}, \theta^0)$
- M-step: (Multivariate optimization)
  - Maximize $p(\theta \mid \text{data})$ using $p(S \mid \ldots..)$
  - This yields new parameter estimates θ^1
- This is the EM algorithm:
  - converges to a (local) maximum of $p(\theta \mid \text{data})$

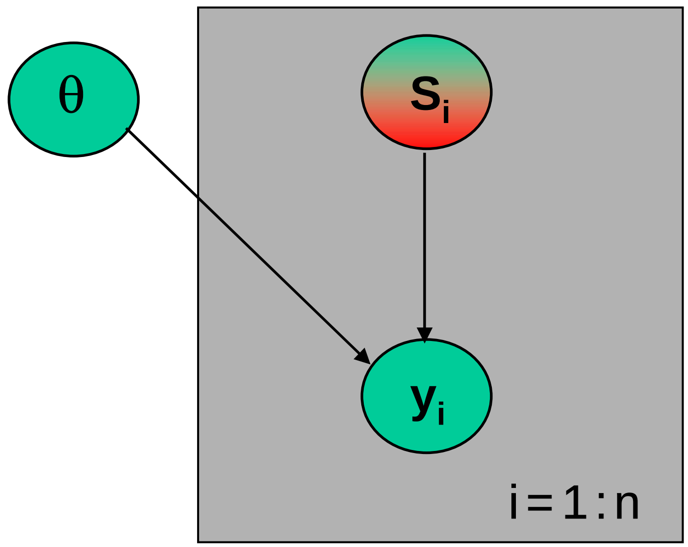
E-Step

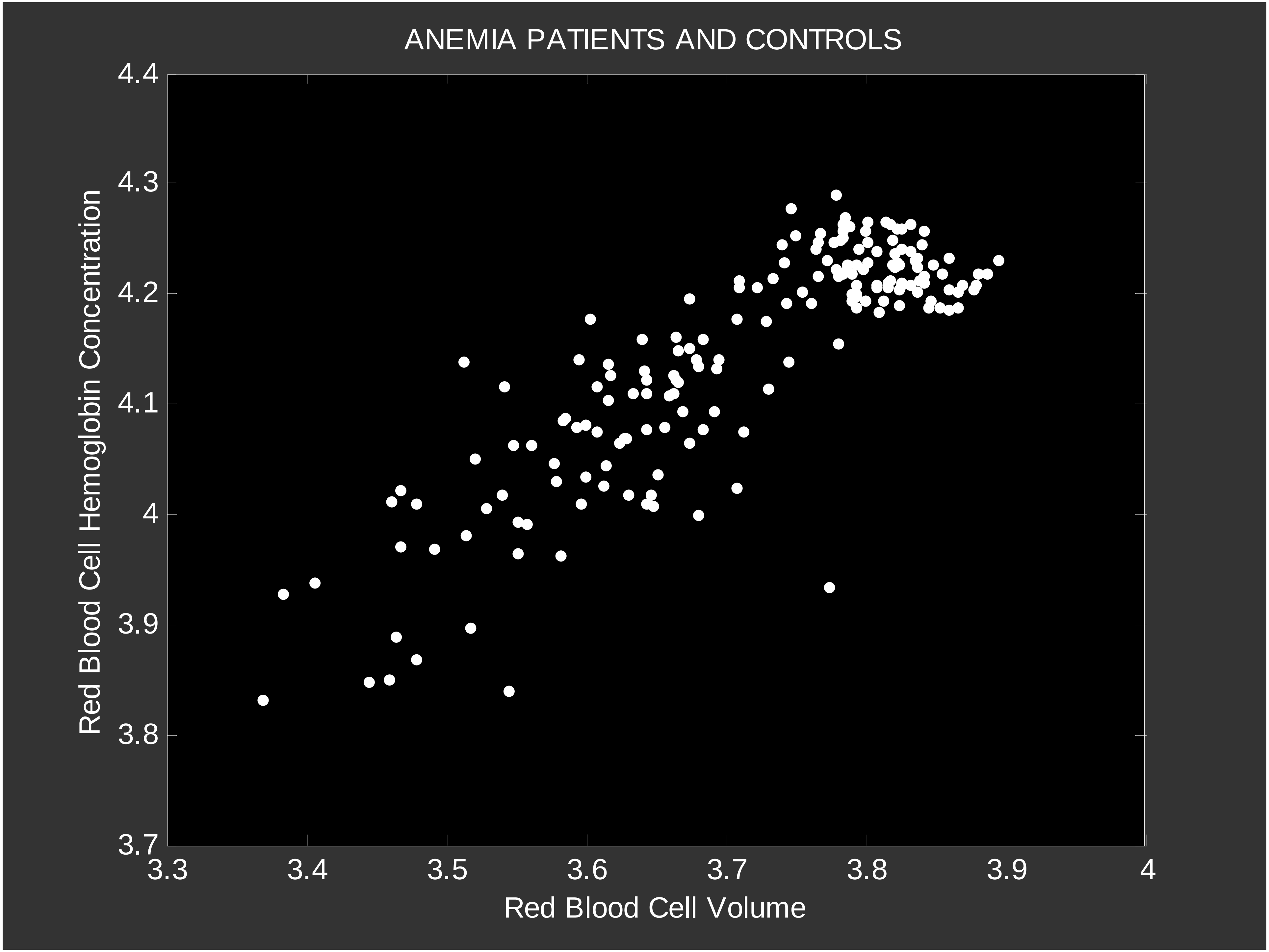


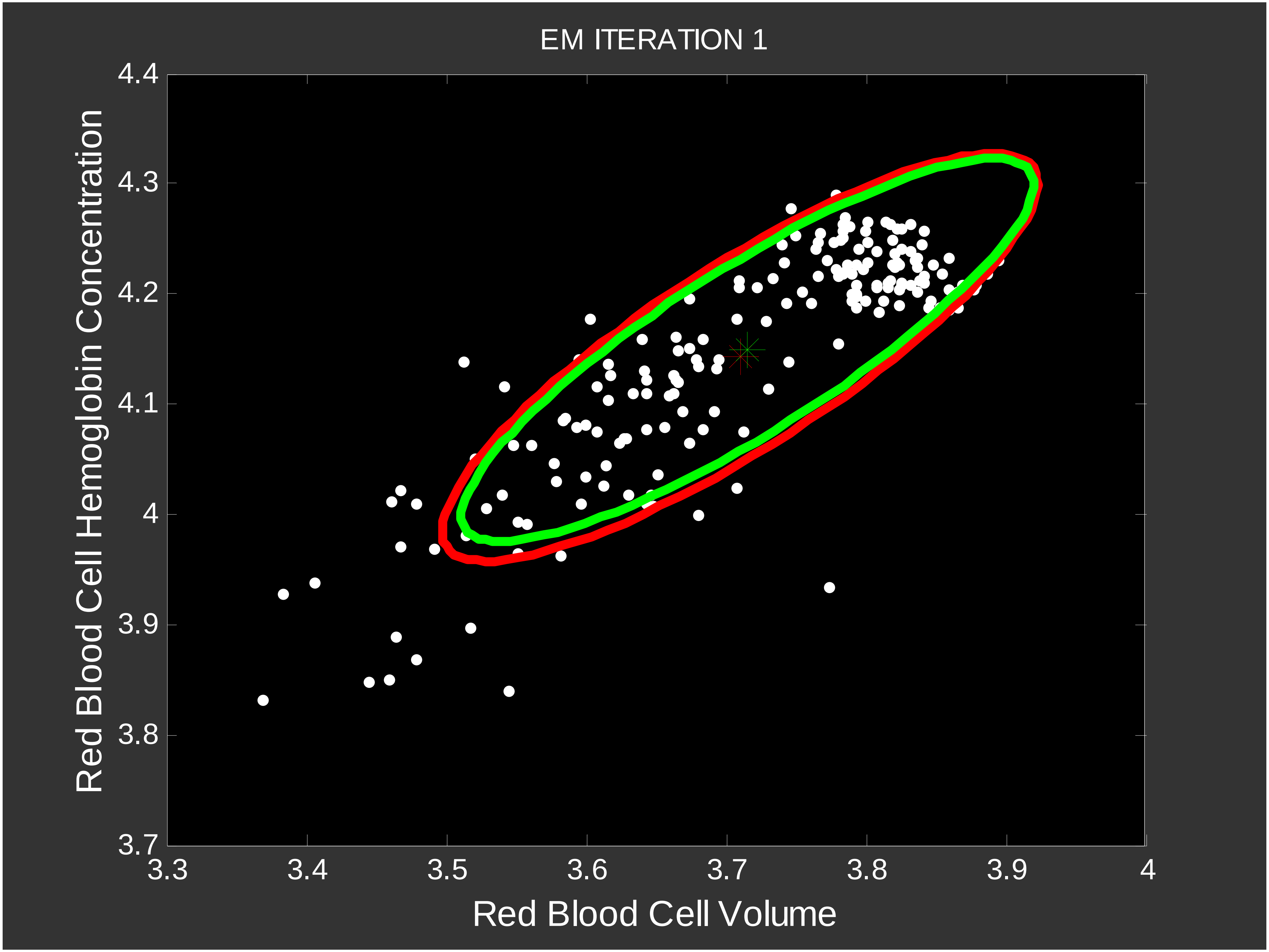
M-Step



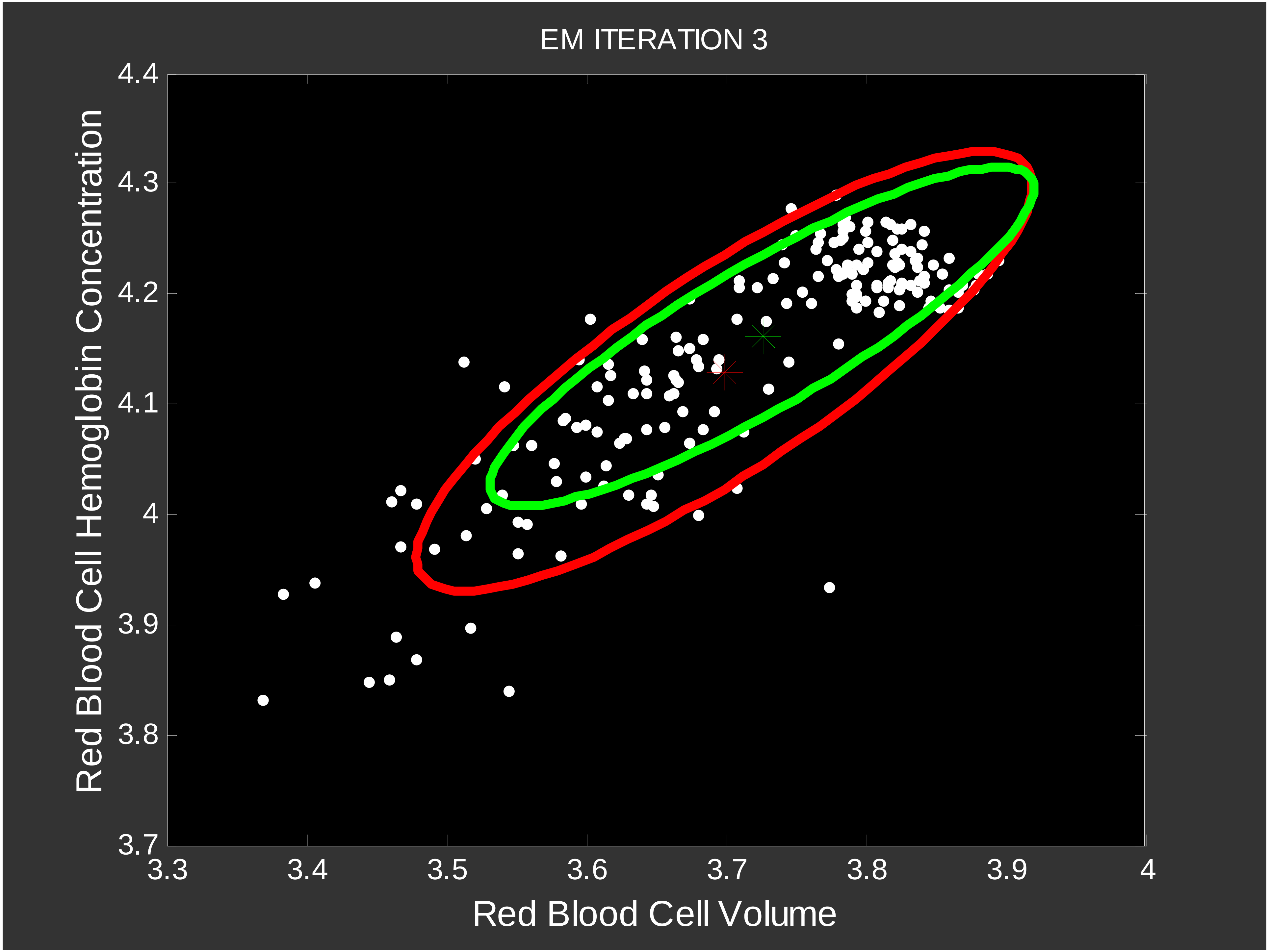
E-Step

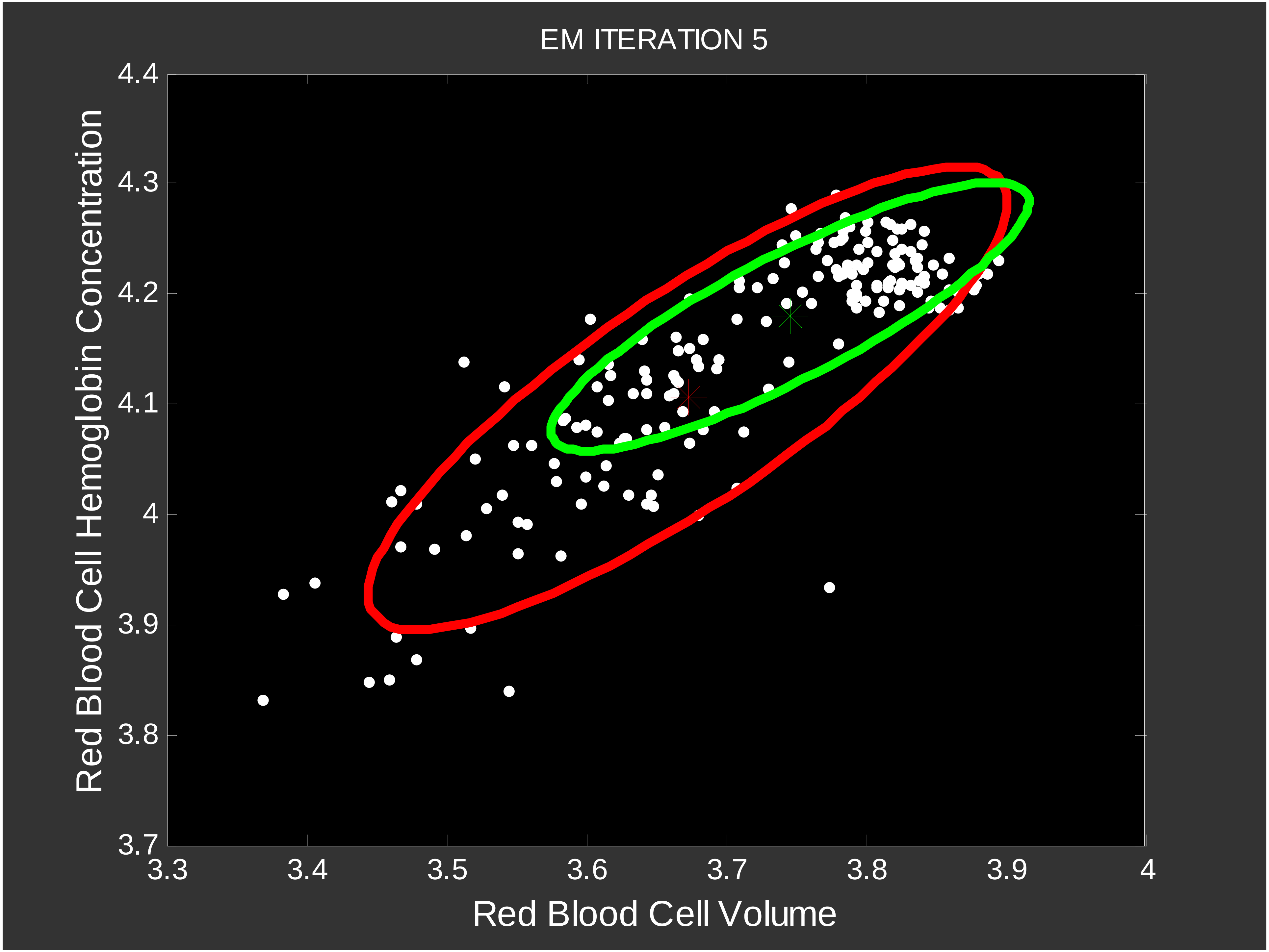



ANEMIA PATIENTS AND CONTROLS
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume

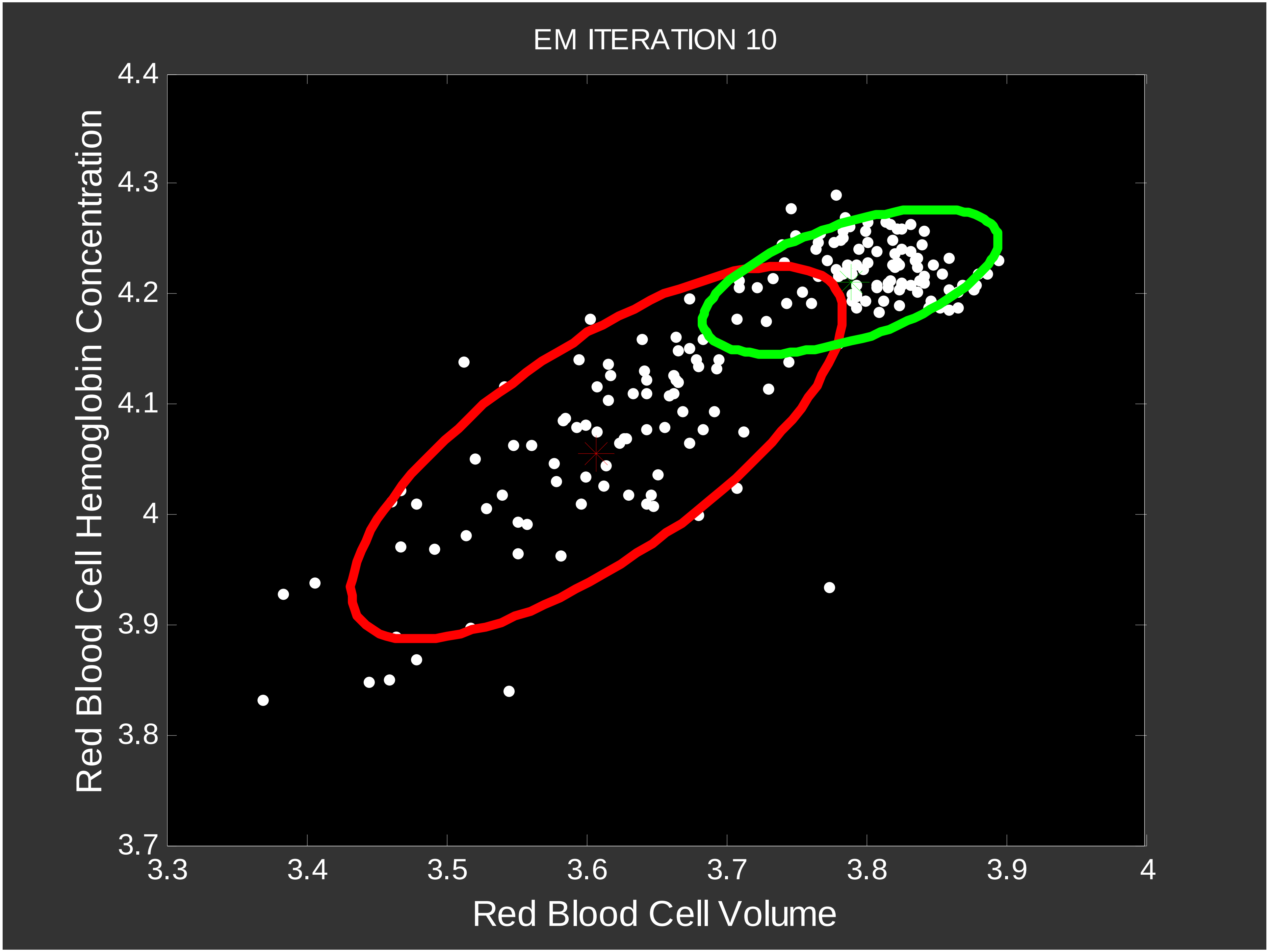


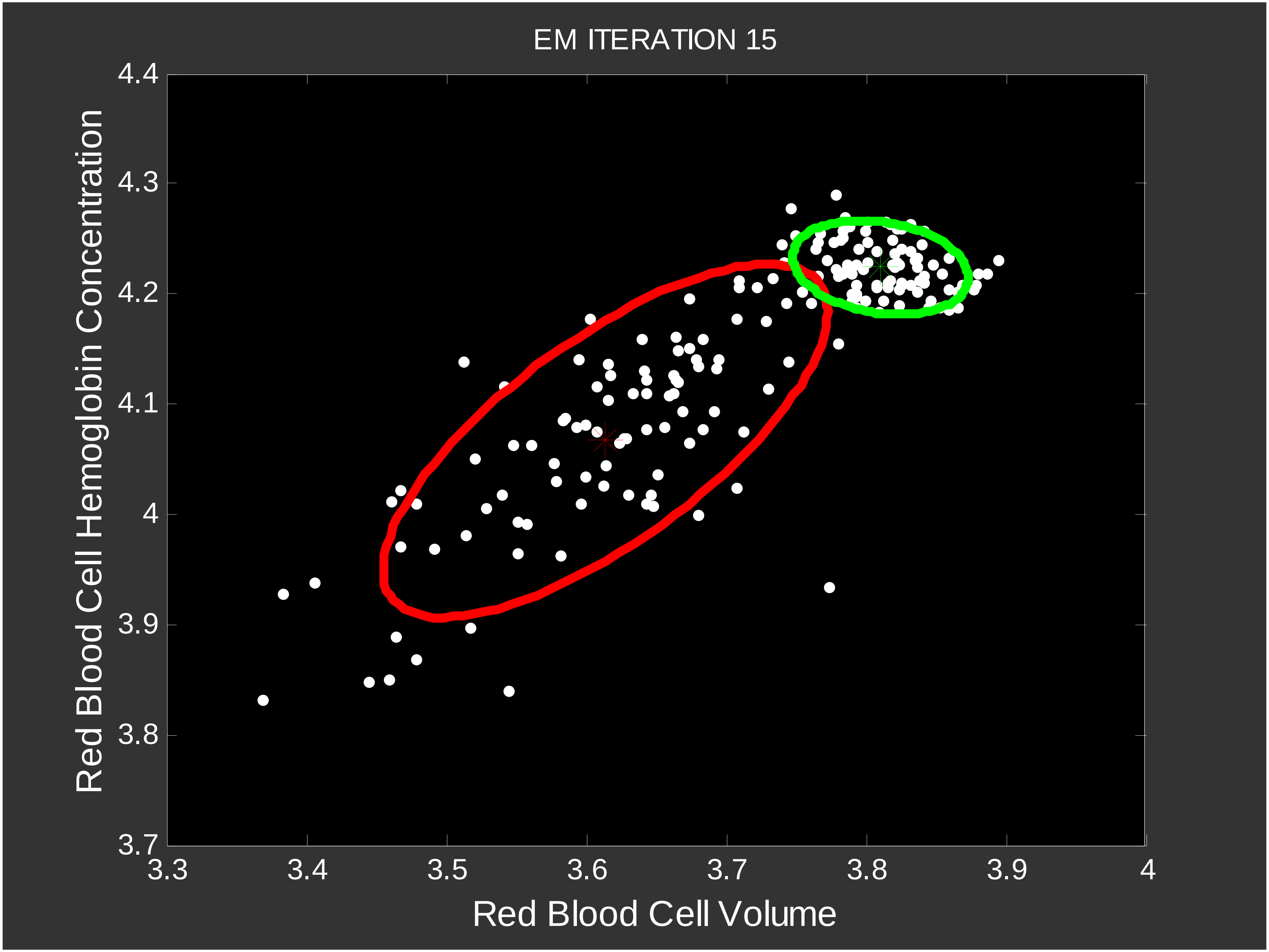
EM ITERATION 1
Red Blood Cell Hemoglobin Concentration
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
Red Blood Cell Volume

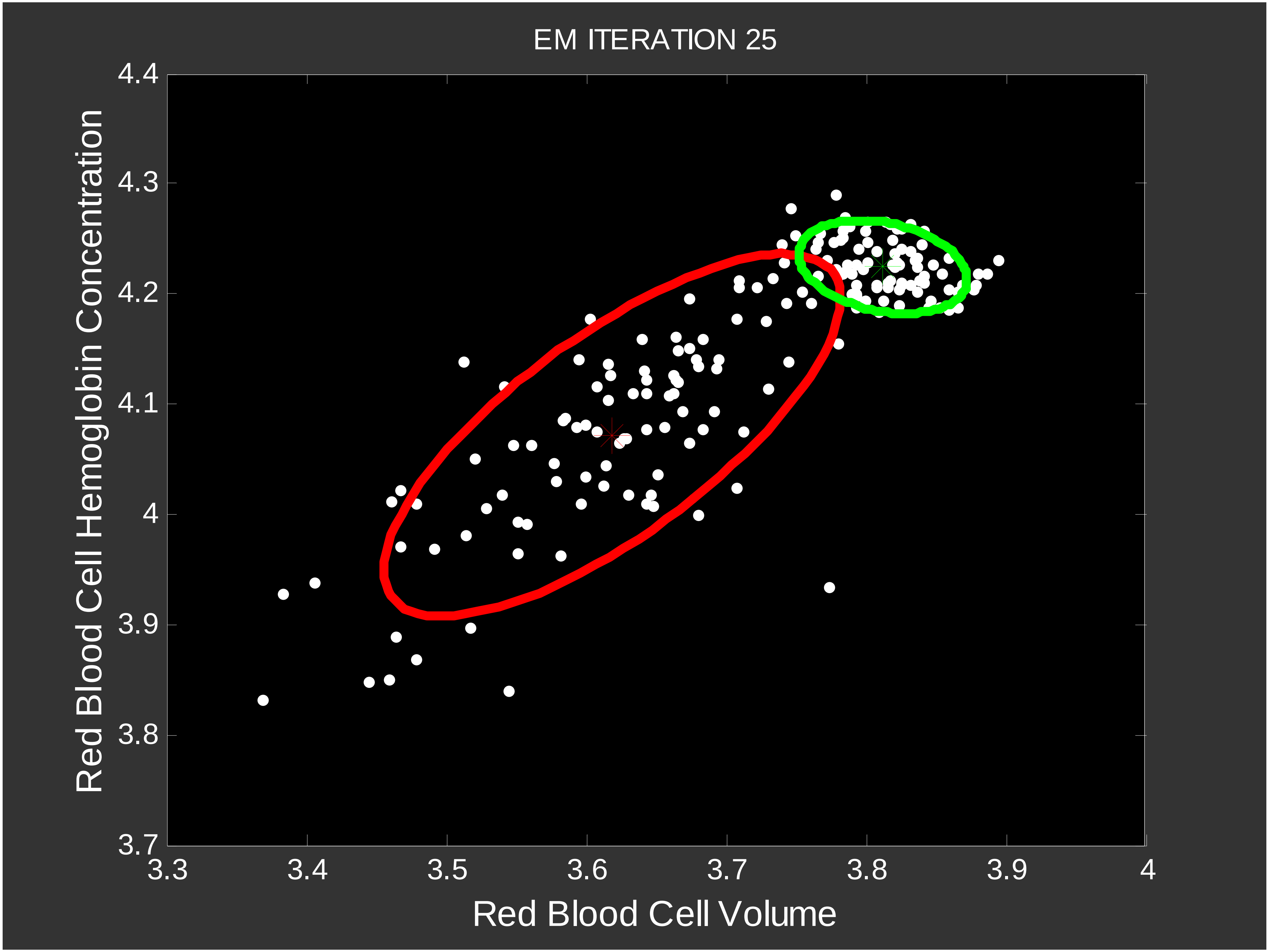

EM ITERATION 3
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4

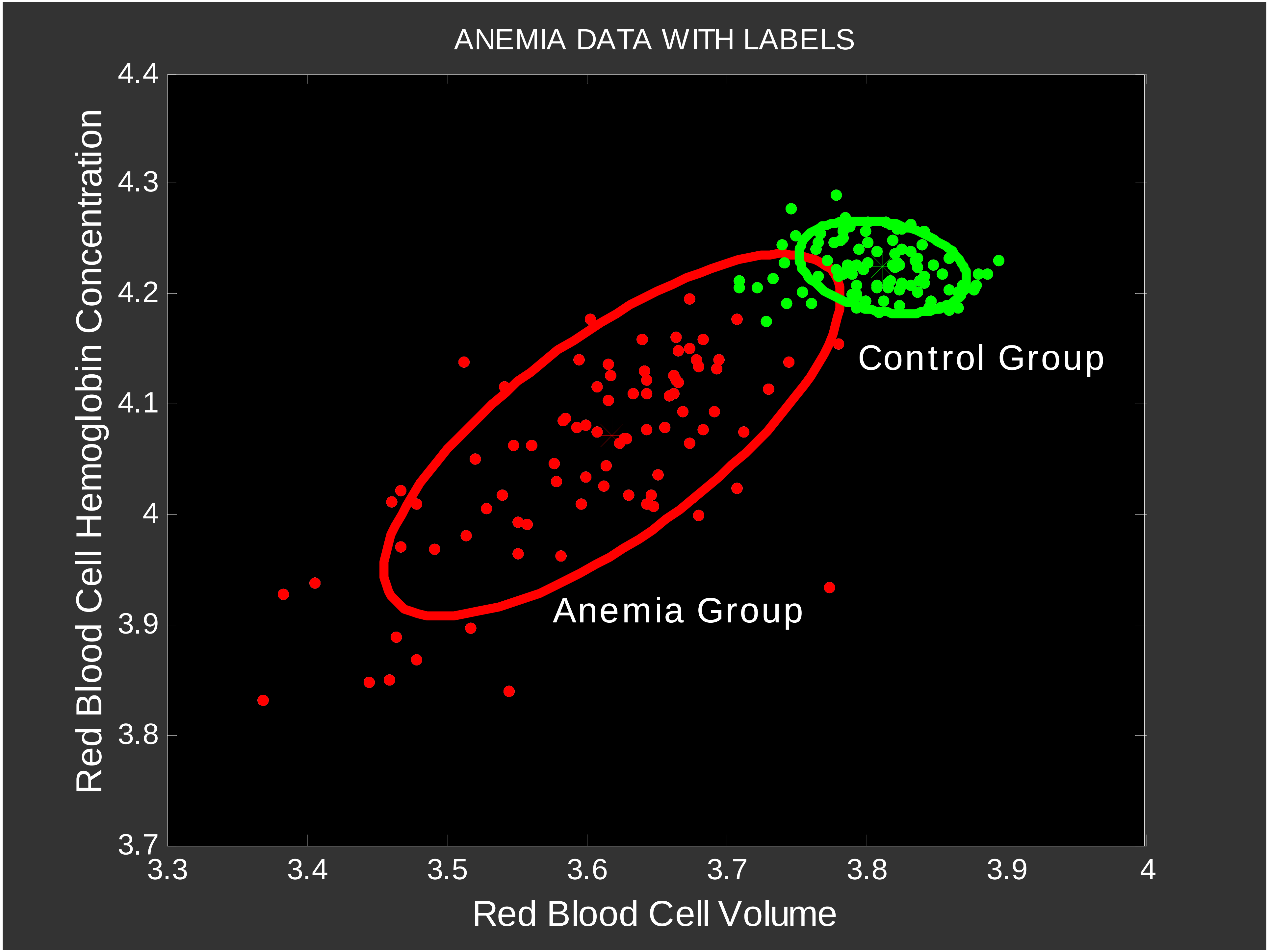


EM ITERATION 5
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4

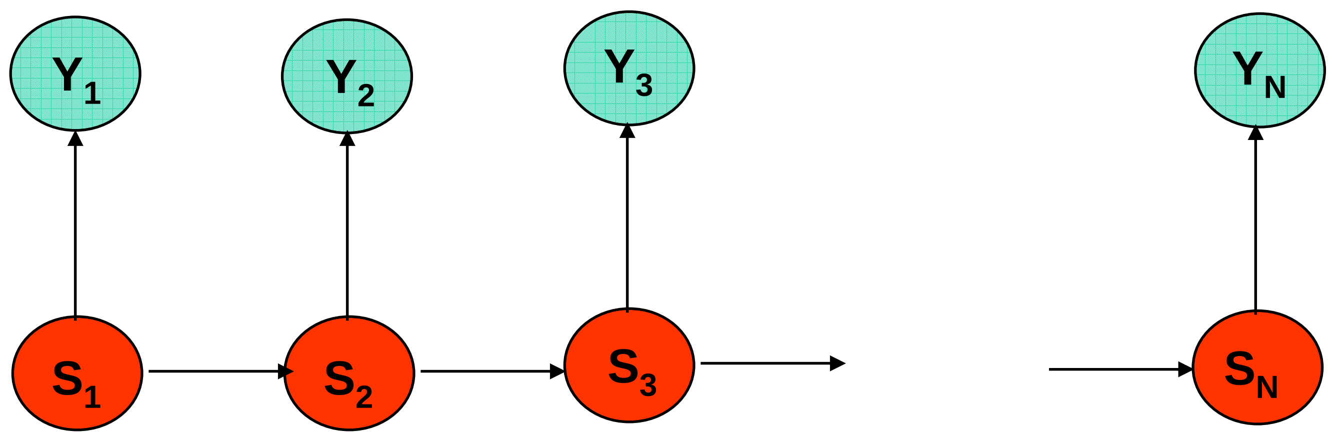

EM ITERATION 10
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4

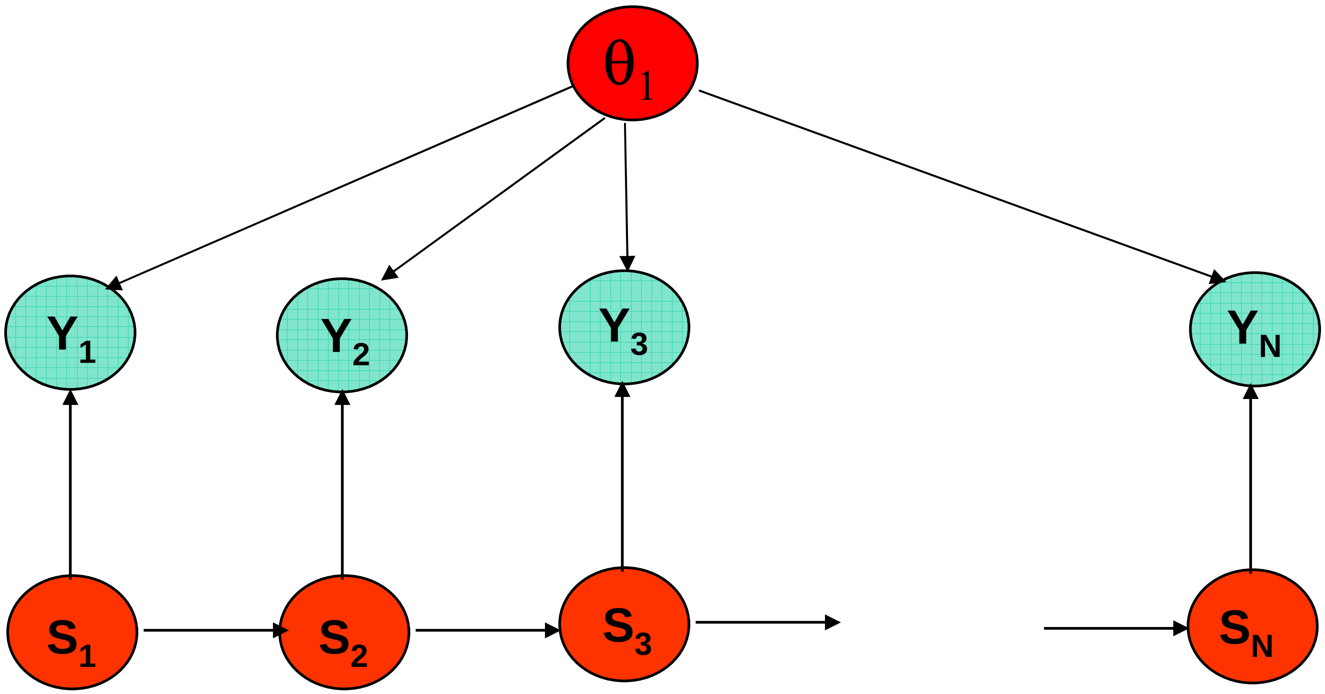

EM ITERATION 15
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4


EM ITERATION 25
Red Blood Cell Hemoglobin Concentration
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
Red Blood Cell Volume

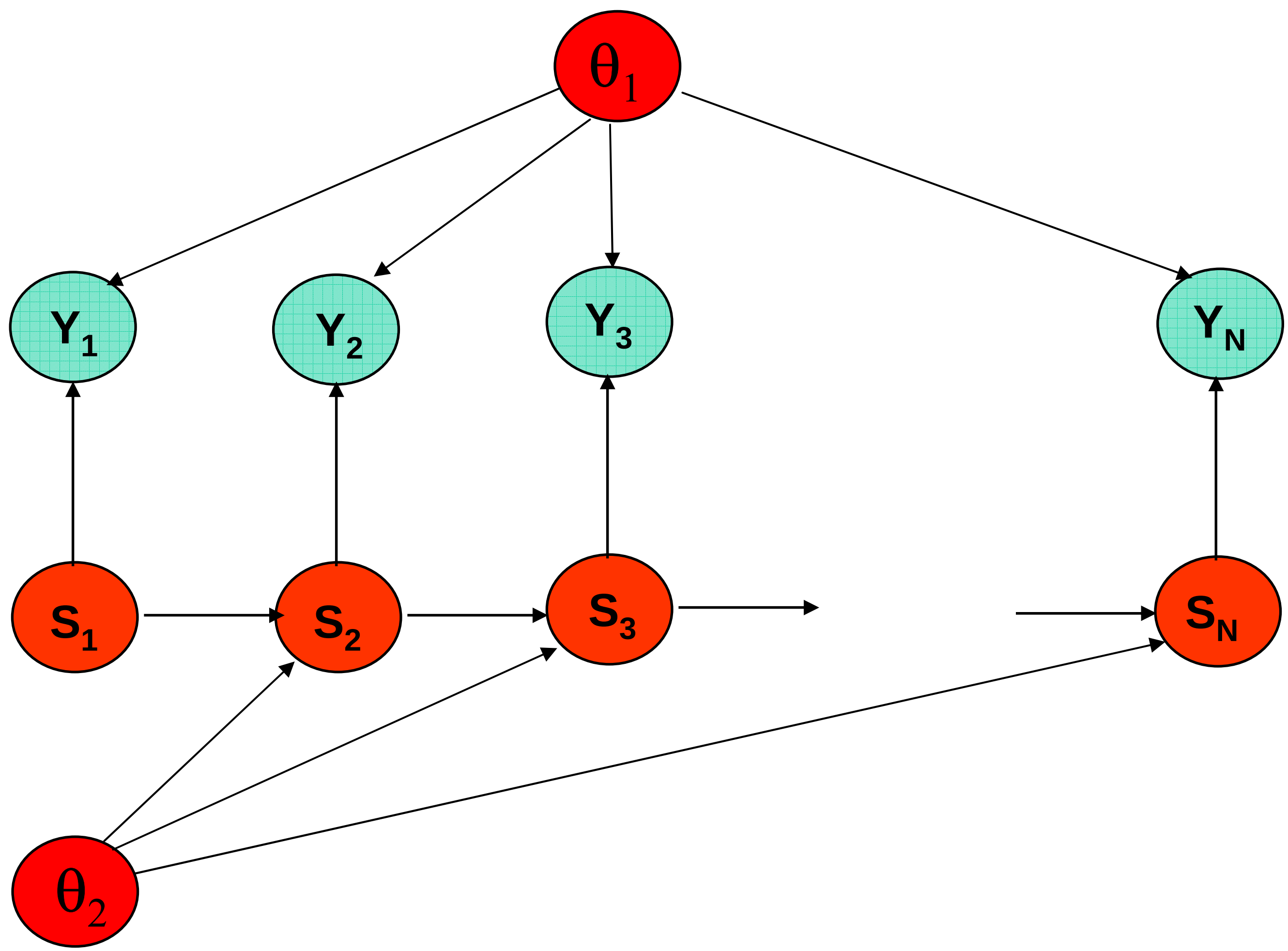

ANEMIA DATA WITH LABELS
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
Control Group
Anemia Group

HMMs



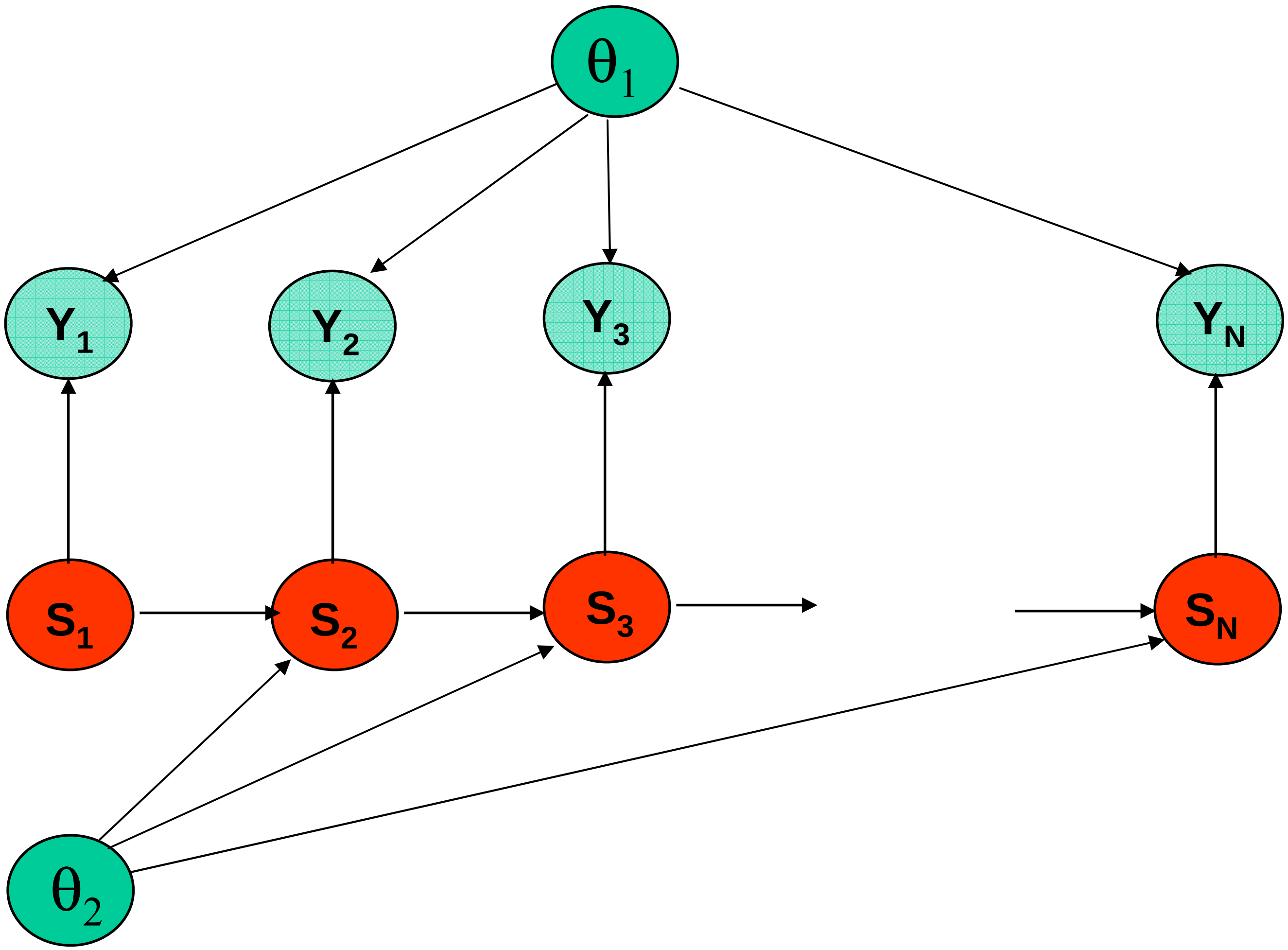


θ_1
Y_1
Y_2
Y_3
Y_N
S_1
S_2
S_3
S_N


θ1
Y1
Y2
Y3
YN
S1
S2
S3
SN
θ2

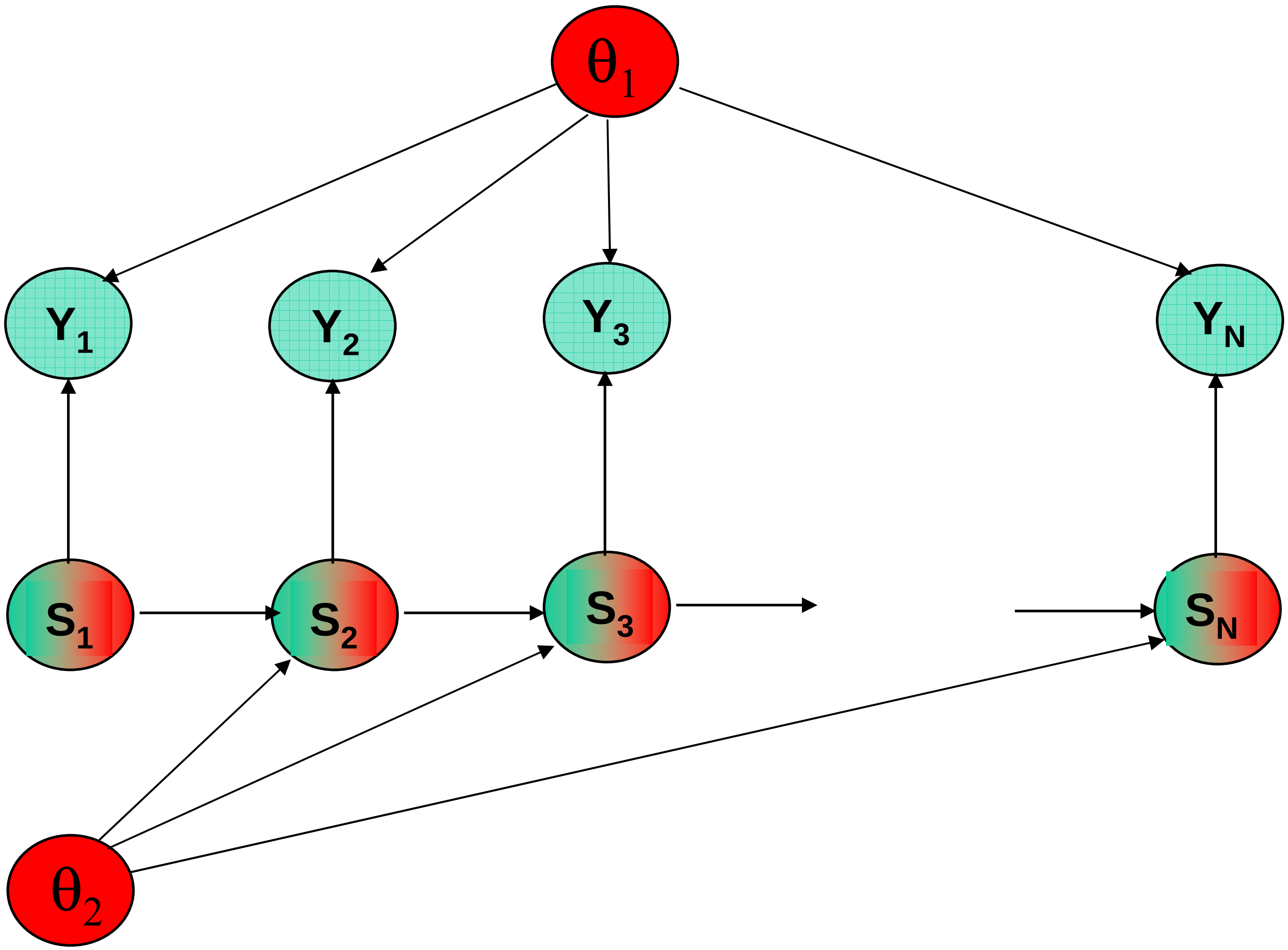
E-Step

Compute p(**s** | θ, **y**), linear time



M-Step

Find θ that maximizes $p(\mathbf{y}|\theta)p(\theta) = \Sigma\, p(\mathbf{y},\mathbf{s}|\theta)\, p(\theta)$



Example 1:

Simulating and Forecasting Seasonal Rainfall Data

Joint work with:

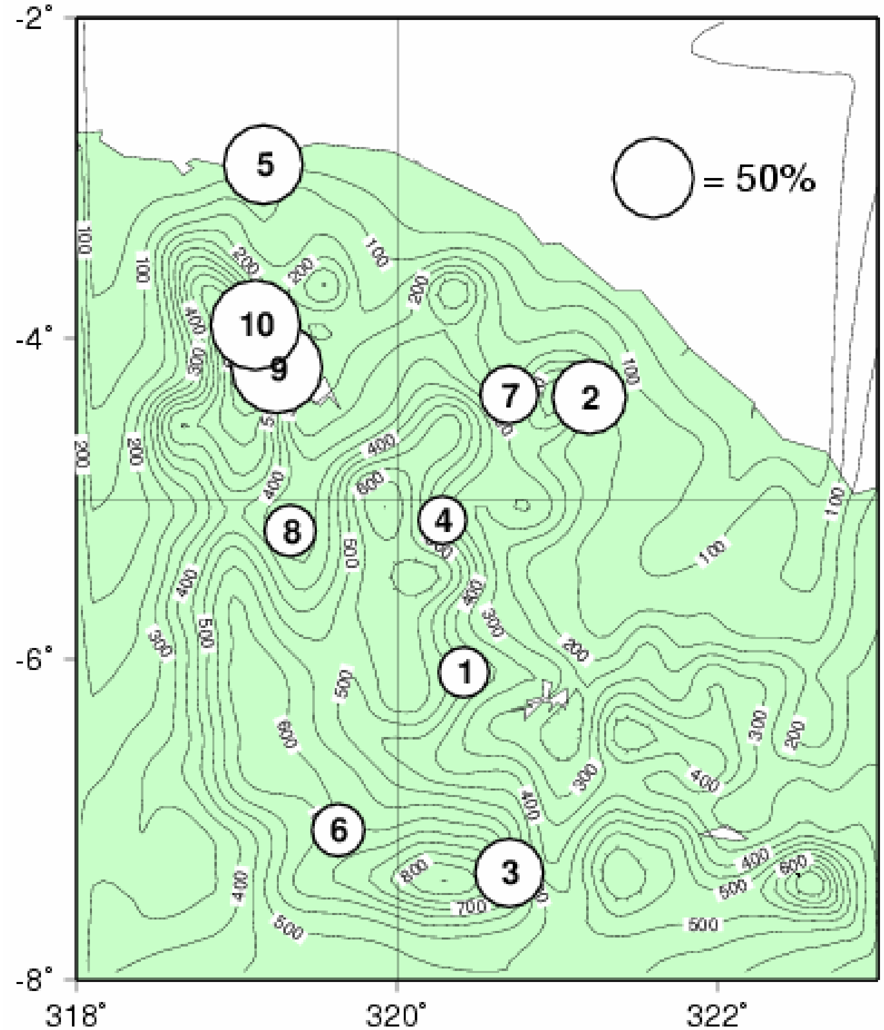
Andy Robertson, International Research Institute for Climate Prediction
Sergey Kirshner, Department of Computer Science, UC Irvine

Robertson, Kirshner, Smyth, Hidden Markov models for modeling daily rainfall occurrence over Brazil, *Journal of Climate*, 17(22):4407-4424, November 2004.

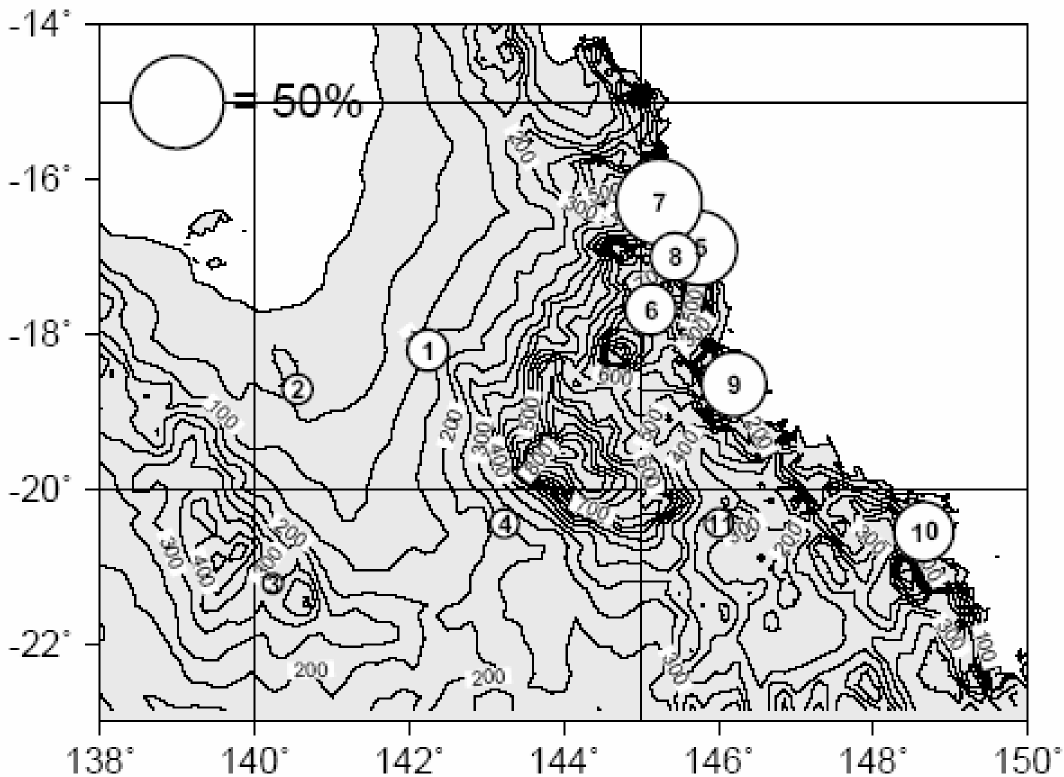
Spatio-Temporal Rainfall Data

Northeast Brazil 1975-2002

90-day time series
24 years
10 stations

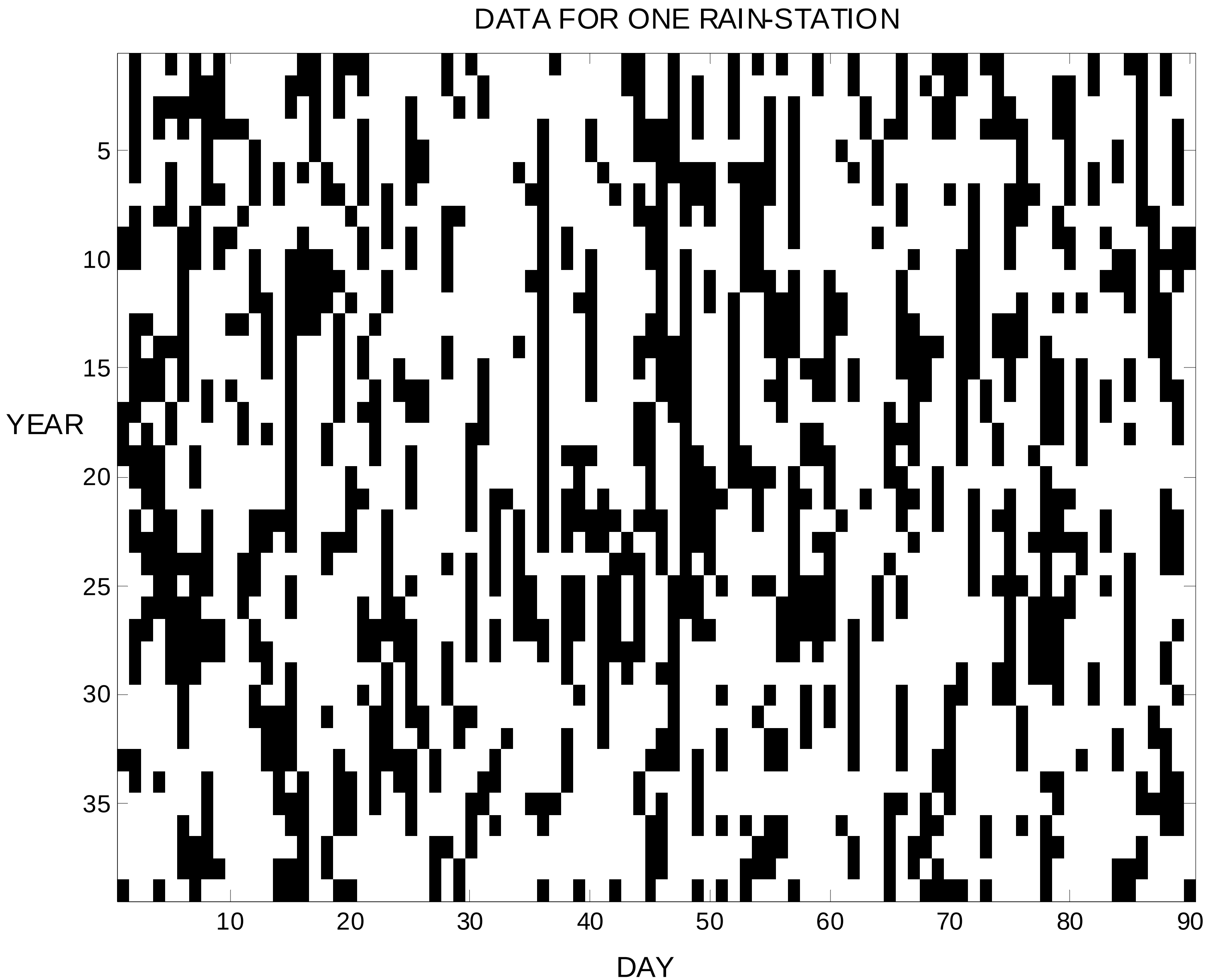


N. Queensland Rainfall Station Oct-Apr Climatology

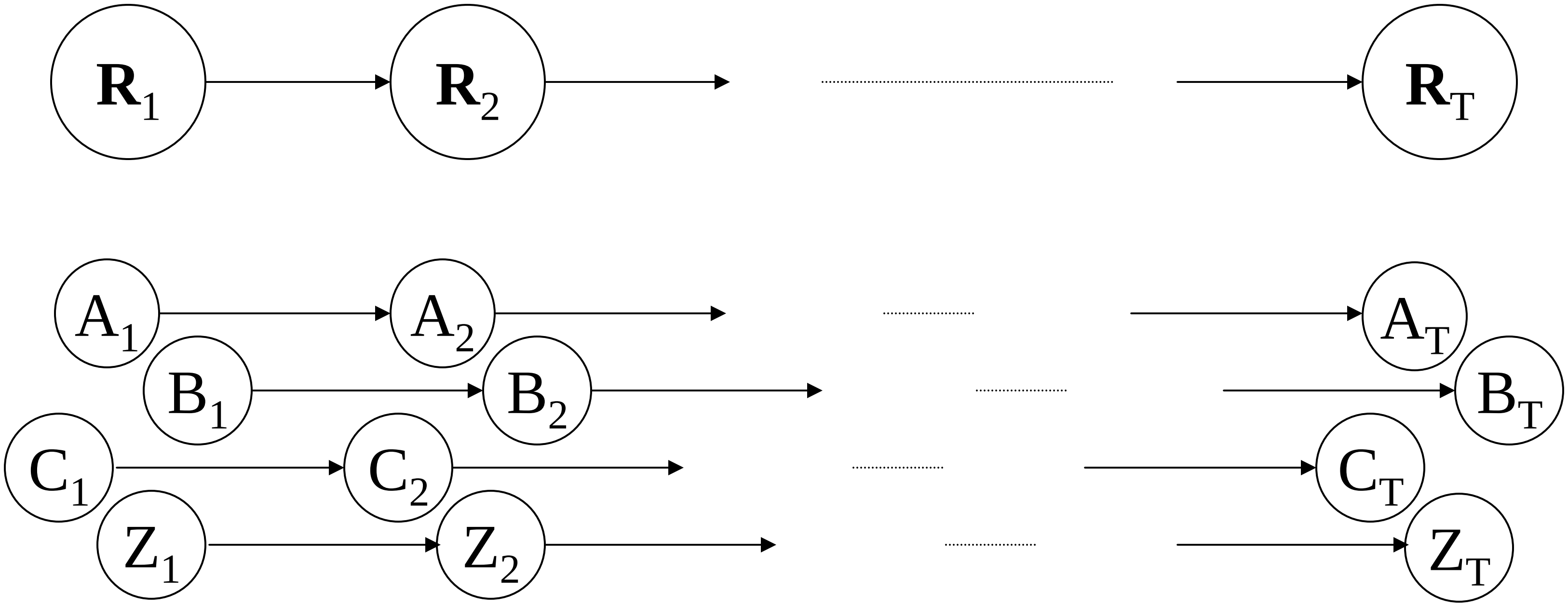


Modeling Goals

- “Downscaling”
  - From GCM output to daily local time-series for crop yield models
- Prediction
  - e.g., “hindcasting” of missing data
- Understanding
  - Relation of precip interannual variability to climate change


DATA FOR ONE RAIN-STATION
YEAR
5
10
15
20
25
30
35
DAY
10
20
30
40
50
60
70
80
90

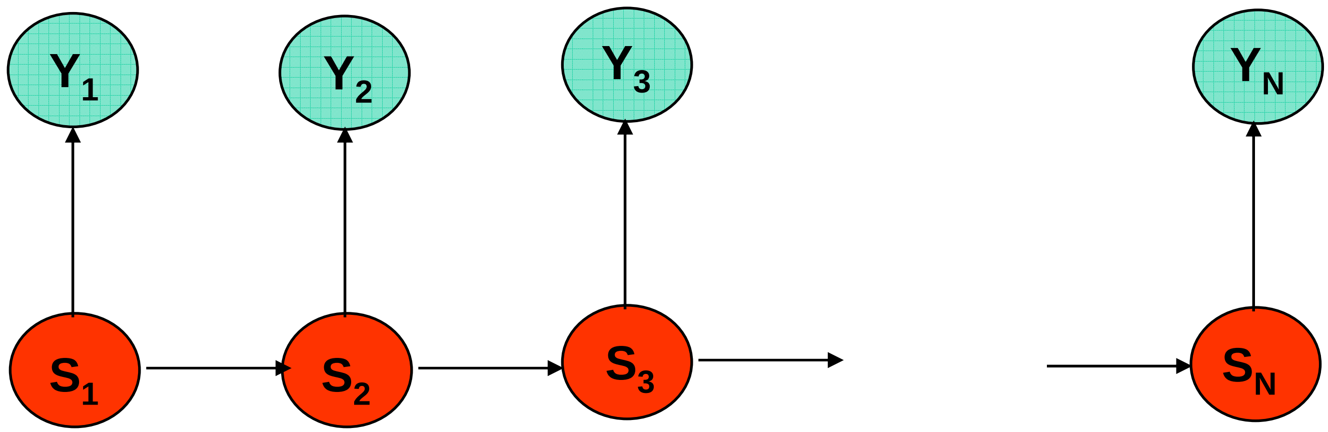
Weather Generator



$$P(\mathbf{R}_{1:T}) = P(\mathbf{R}_1)\prod_{t=2}^{T} P(\mathbf{R}_t \mid \mathbf{R}_{t-1}) = \prod_{c\in\{\mathrm{A},..,\mathrm{Z}\}} \left(P(c_1)\prod_{t=2}^{T} P(c_t \mid c_{t-1}) \right)$$

- Does not take spatial correlation into account

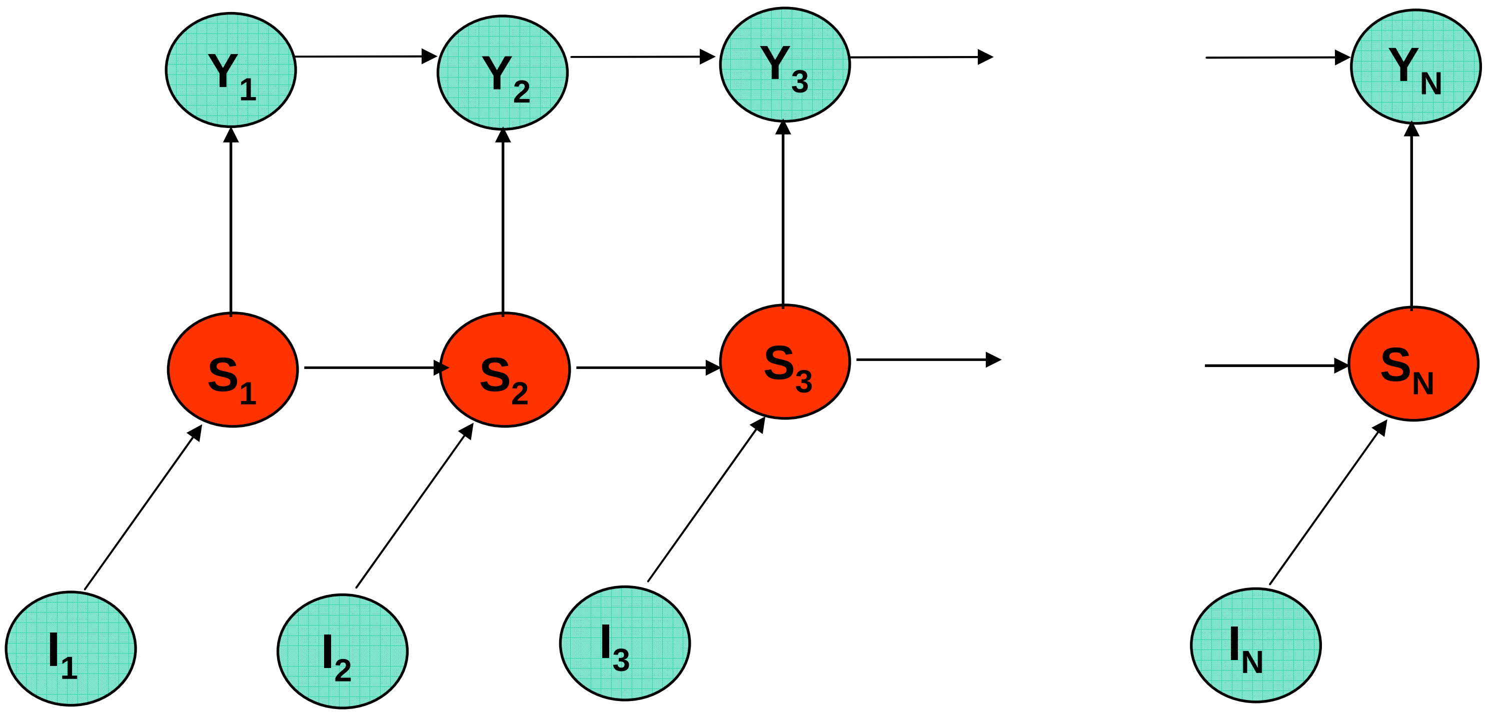
HMMs for Rainfall Modeling



- S = unobserved weather state

  Y = spatial rainfall pattern (“outputs”)

HMMs for Rainfall Modeling



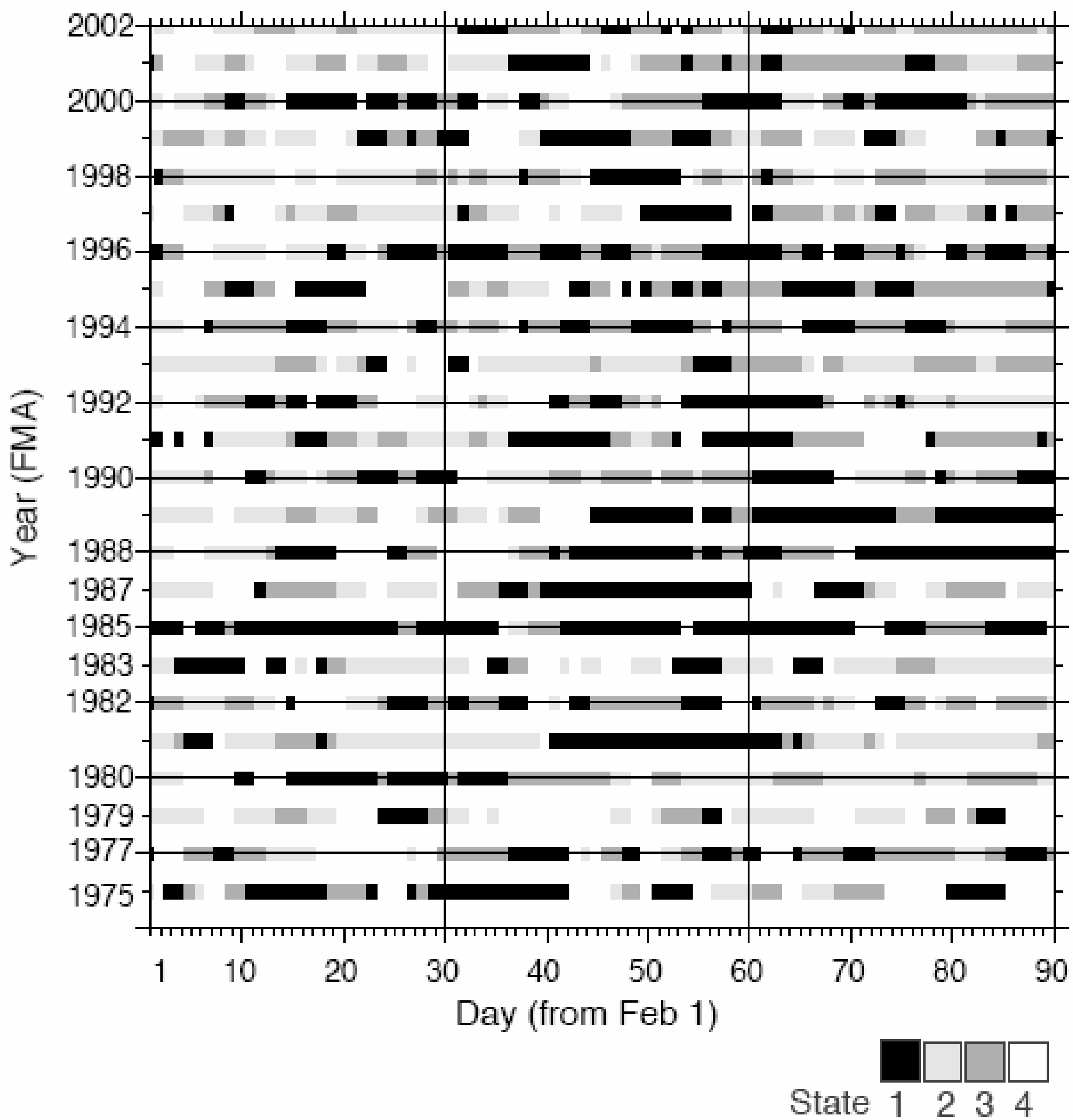
- S = unobserved weather state
  Y = spatial rainfall pattern (“outputs”)
  I = atmospheric variables (“inputs”)

Modeling and Estimation

- Model
  - Transitions $p(s_t \mid s_{t-1})$ are now $p(s_t \mid s_{t-1}, i_{t-1})$
  - Parametrized by a logistic function

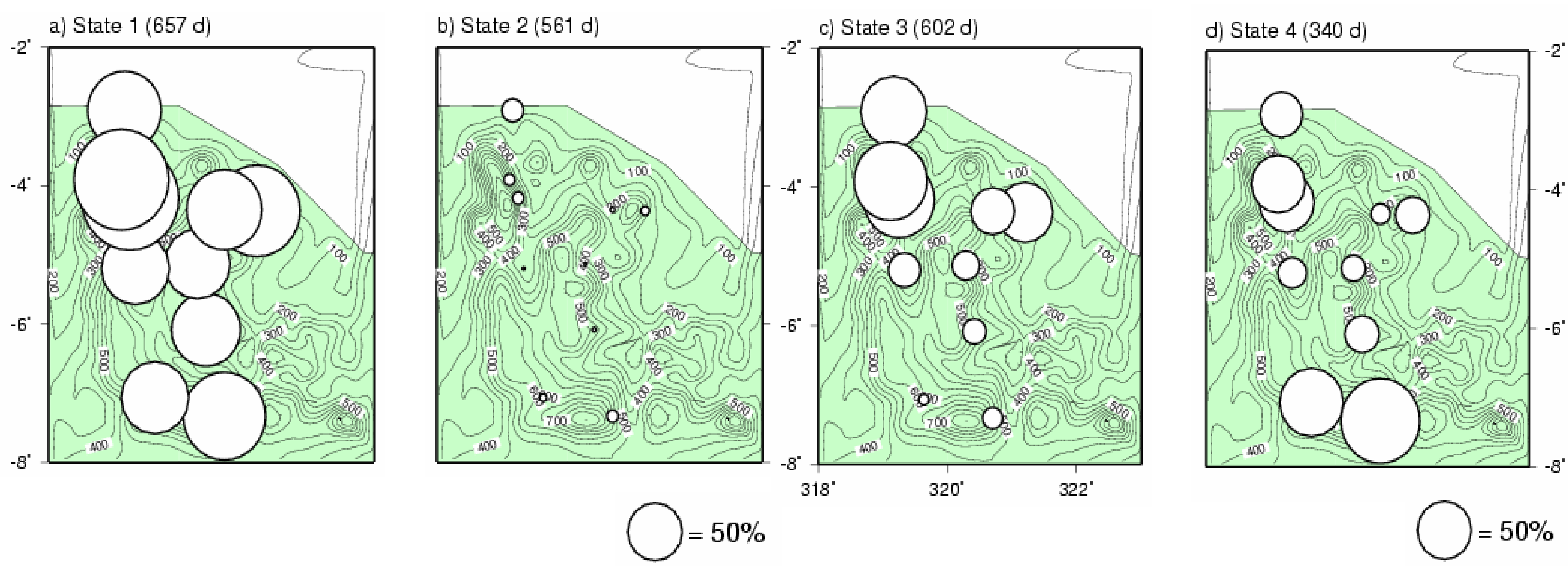
- Parameter estimation
  - EM algorithm can be derived from general principles
  - E-step:
    - linear in length of sequence
  - M-step:
    - No closed form solution with logistic function
    - Solve a numerical optimization problem at each M-step

- "Parsing"
  - Given a model, can estimate most likely state sequence in historical data
  - Assigns each day to its most likely state

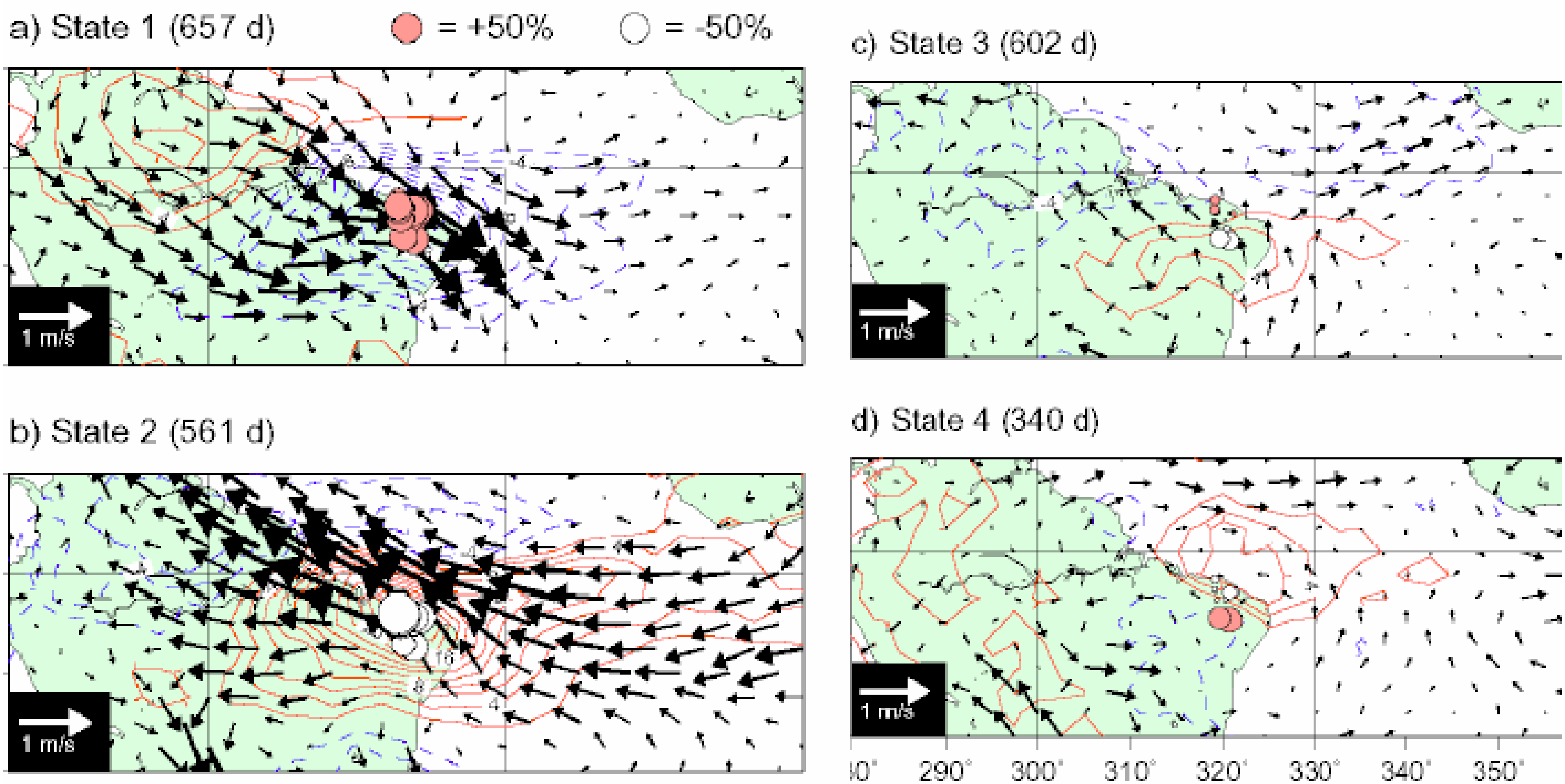


2002
2000
1998
1996
1994
1992
1990
1988
1987
1985
1983
1982
1980
1979
1977
1975
Year (FMA)
1
10
20
30
40
50
60
70
80
90
Day (from Feb 1)
State 1 2 3 4

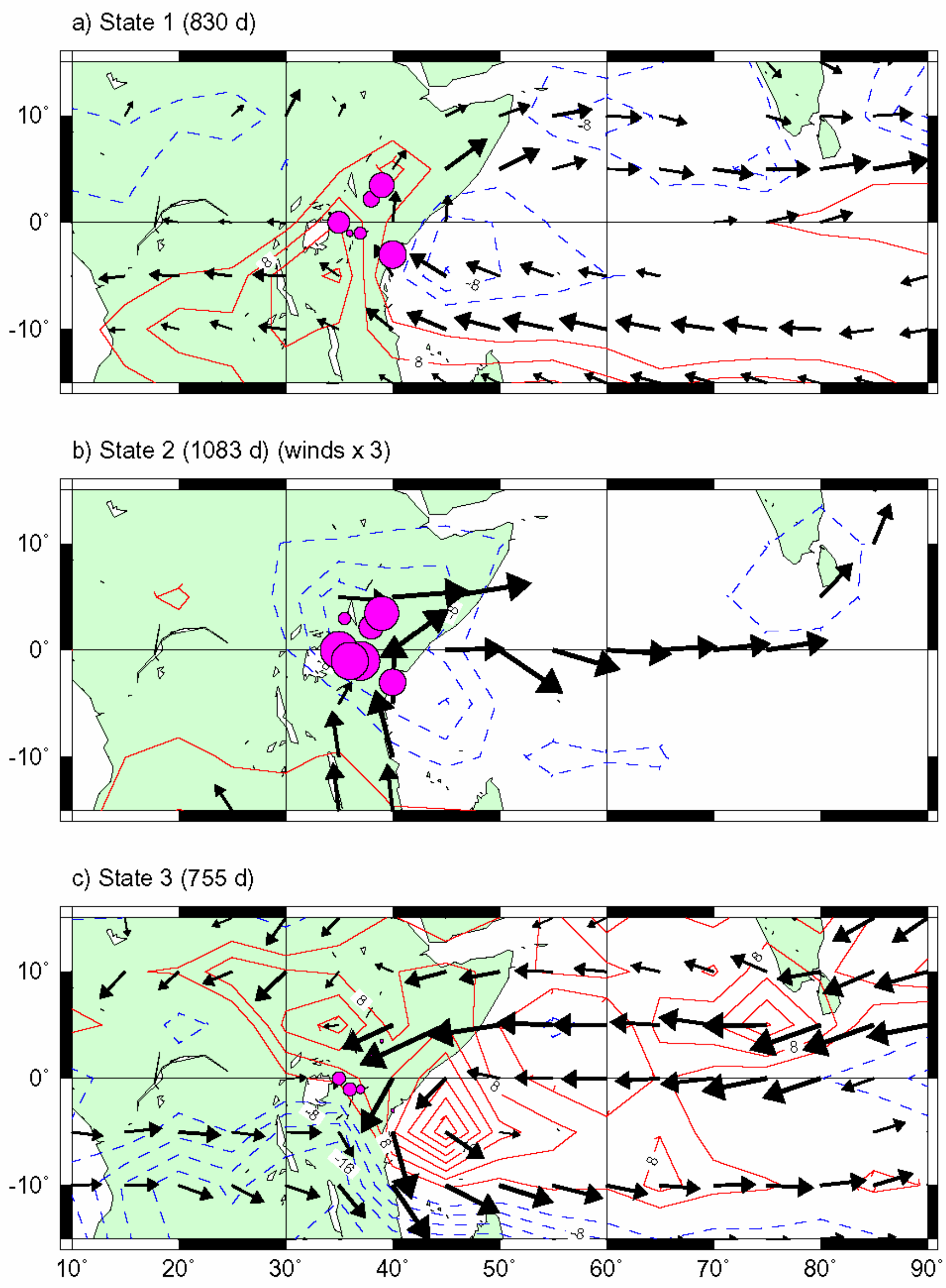
Resulting Weather States



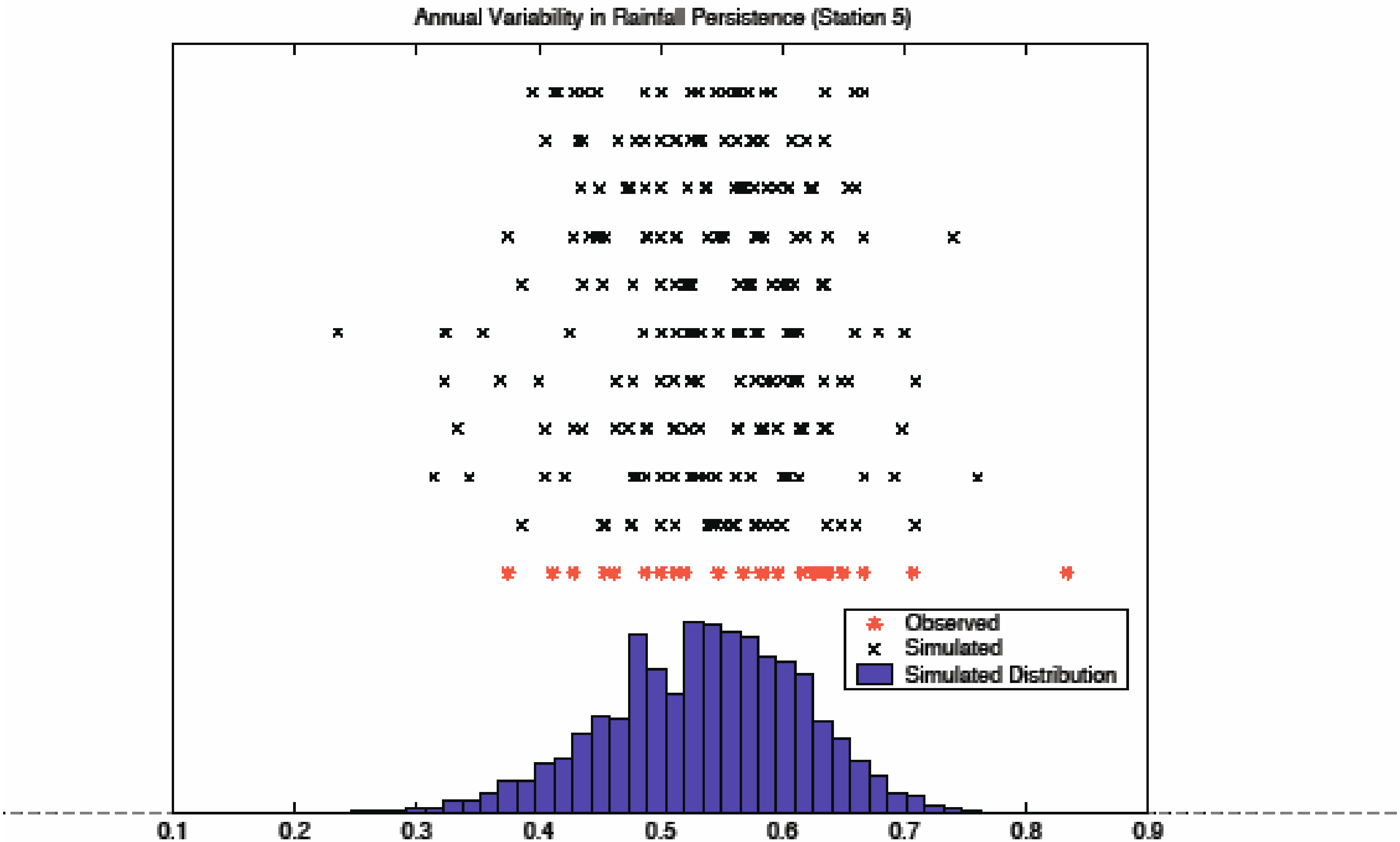
States provide an interpretable “view” of spatio-temporal relationships in the data



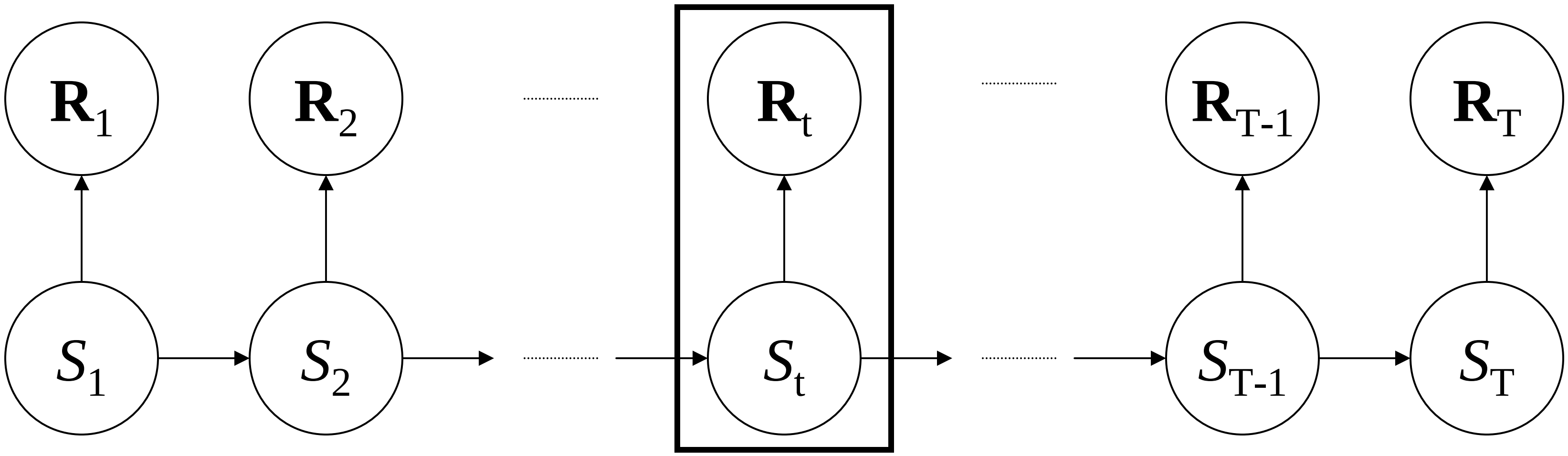
a) State 1 (657 d)
= +50%
= -50%
1 m/s
b) State 2 (561 d)
1 m/s
c) State 3 (602 d)
1 m/s
d) State 4 (340 d)
1 m/s
290°
300°
310°
320°
330°
340°
350°

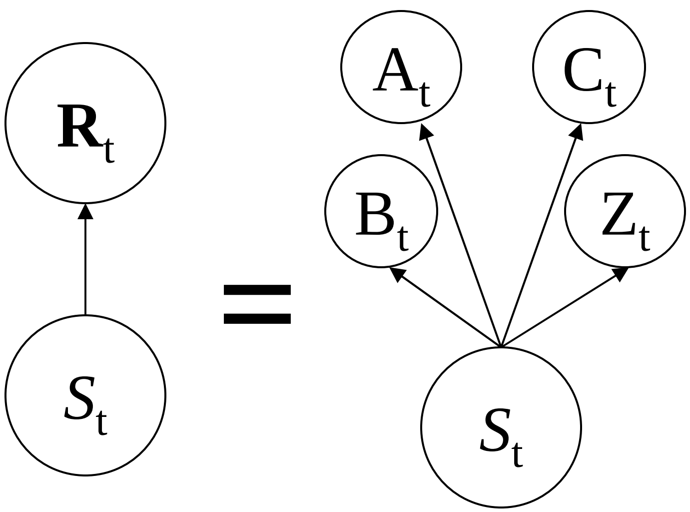


Weather States for Kenya

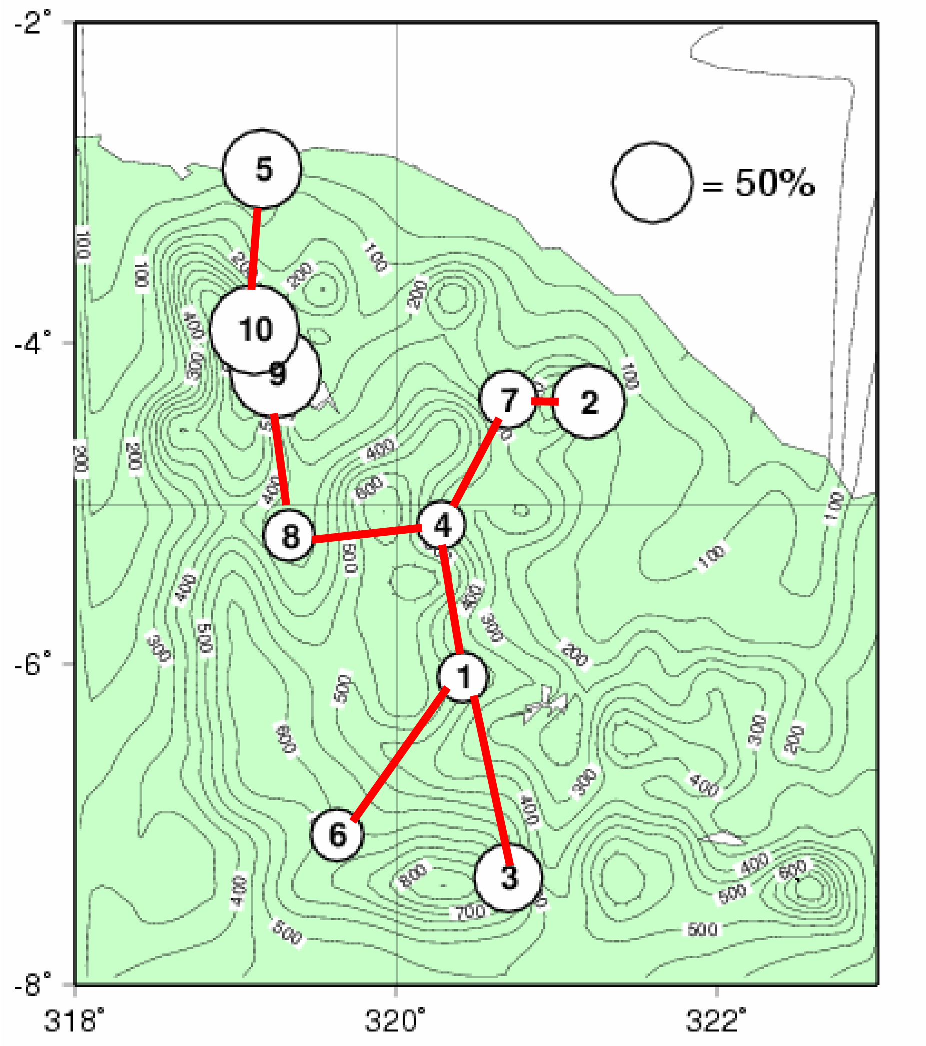

Annual Variability in Rainfall Persistence (Station 5)
Observed
Simulated
Simulated Distribution
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

HMM-Conditional-Independence





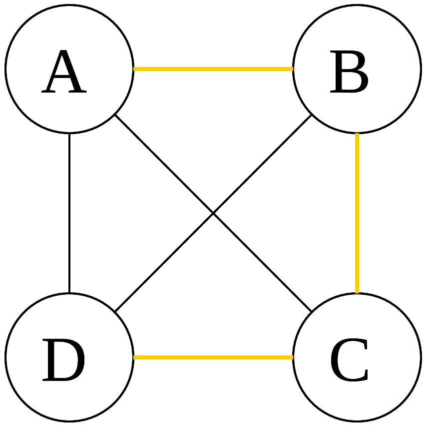
$$
\begin{aligned}
P(\mathbf{R}_t \mid S_t) &= P(\mathrm{A}_t, \ldots, \mathrm{Z}_t \mid S_t) \\
&= \prod_{c \in \{\mathrm{A}, \ldots, \mathrm{Z}\}} P(c_t \mid S_t)
\end{aligned}
$$

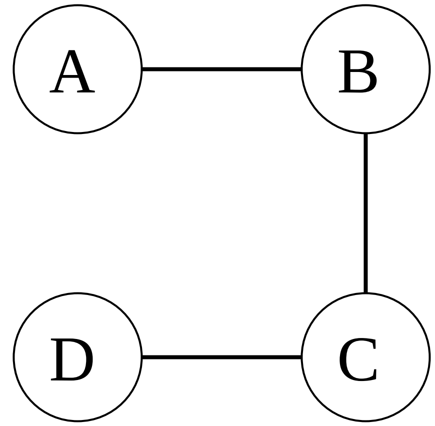


Spatial Chow-Liu Trees

- Spatial distribution given a state is a tree structure
- Useful intermediate between full pair-wise model and conditional independence
- Topology learned from data
- Can use priors based on distance, topography
- Tree-structure over time also

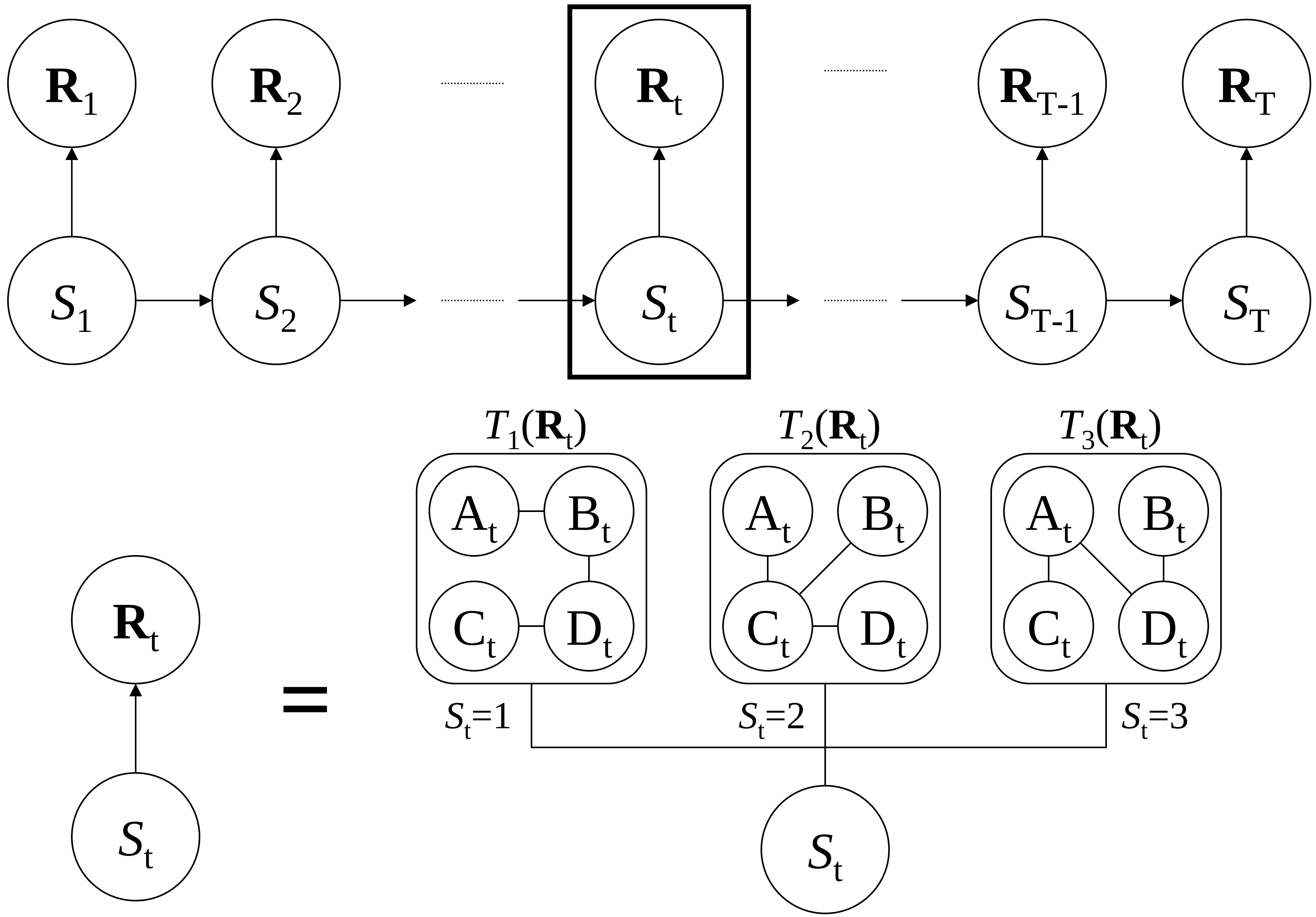
Illustration of CL-Tree Learning



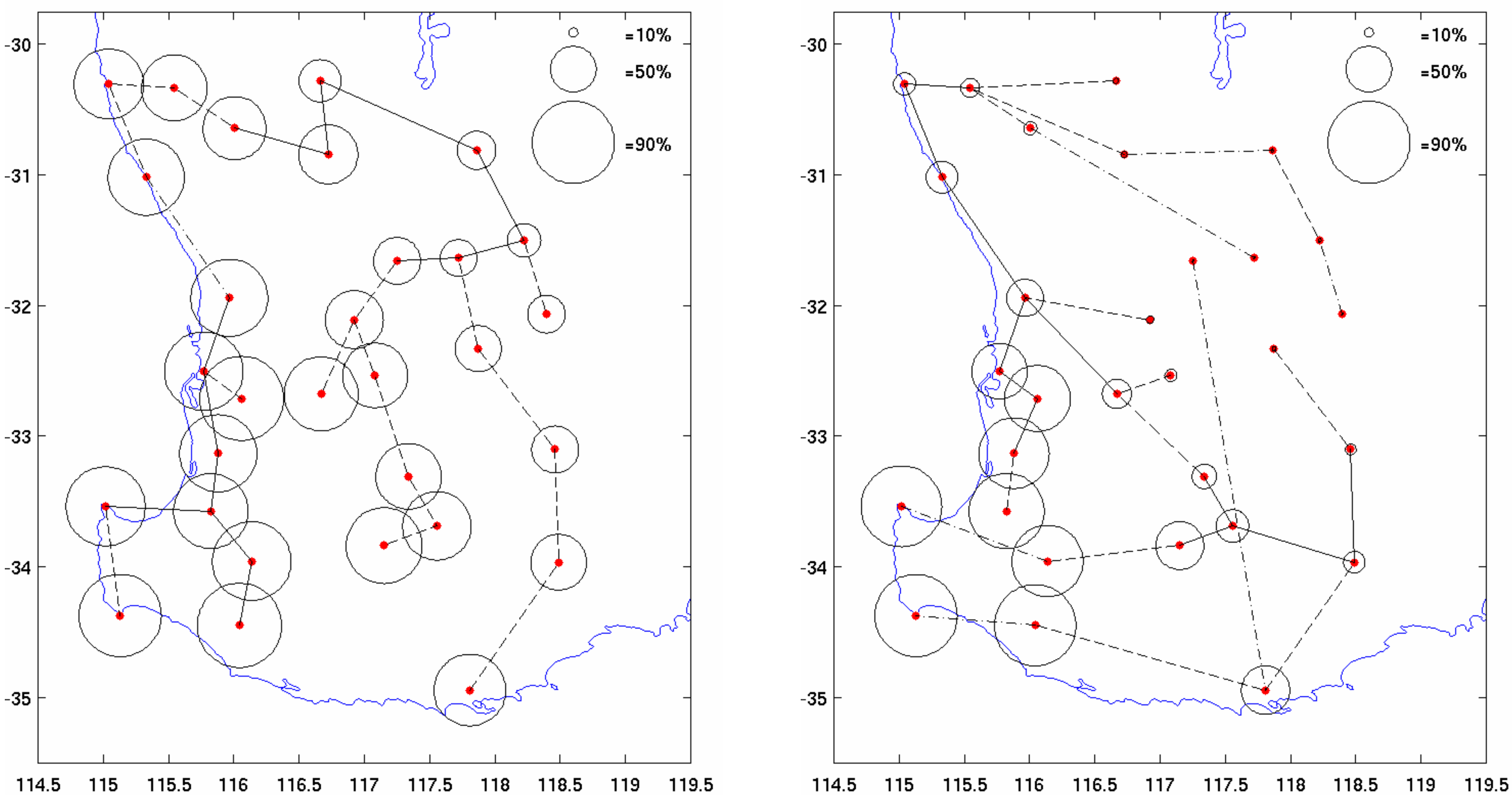


AB	(0.56, 0.11, 0.02, 0.31)	0.3126
AC	(0.51, 0.17, 0.17, 0.15)	0.0229
AD	(0.53, 0.15, 0.19, 0.13)	0.0172
BC	(0.44, 0.14, 0.23, 0.19)	0.0230
BD	(0.46, 0.12, 0.26, 0.16)	0.0183
CD	(0.64, 0.04, 0.08, 0.24)	0.2603

HMM-Chow-Liu


$\mathbf{R}_1$
$\mathbf{R}_2$
$\mathbf{R}_t$
$\mathbf{R}_{T-1}$
$\mathbf{R}_T$
S_1
S_2
S_t
S_{T-1}
S_T
$T_1(\mathbf{R}_t)$
$T_2(\mathbf{R}_t)$
$T_3(\mathbf{R}_t)$
A_t
B_t
C_t
D_t
$\mathbf{R}_t$
=
S_t
S_t=1
S_t=2
S_t=3
S_t

Tree-Structured Weather States



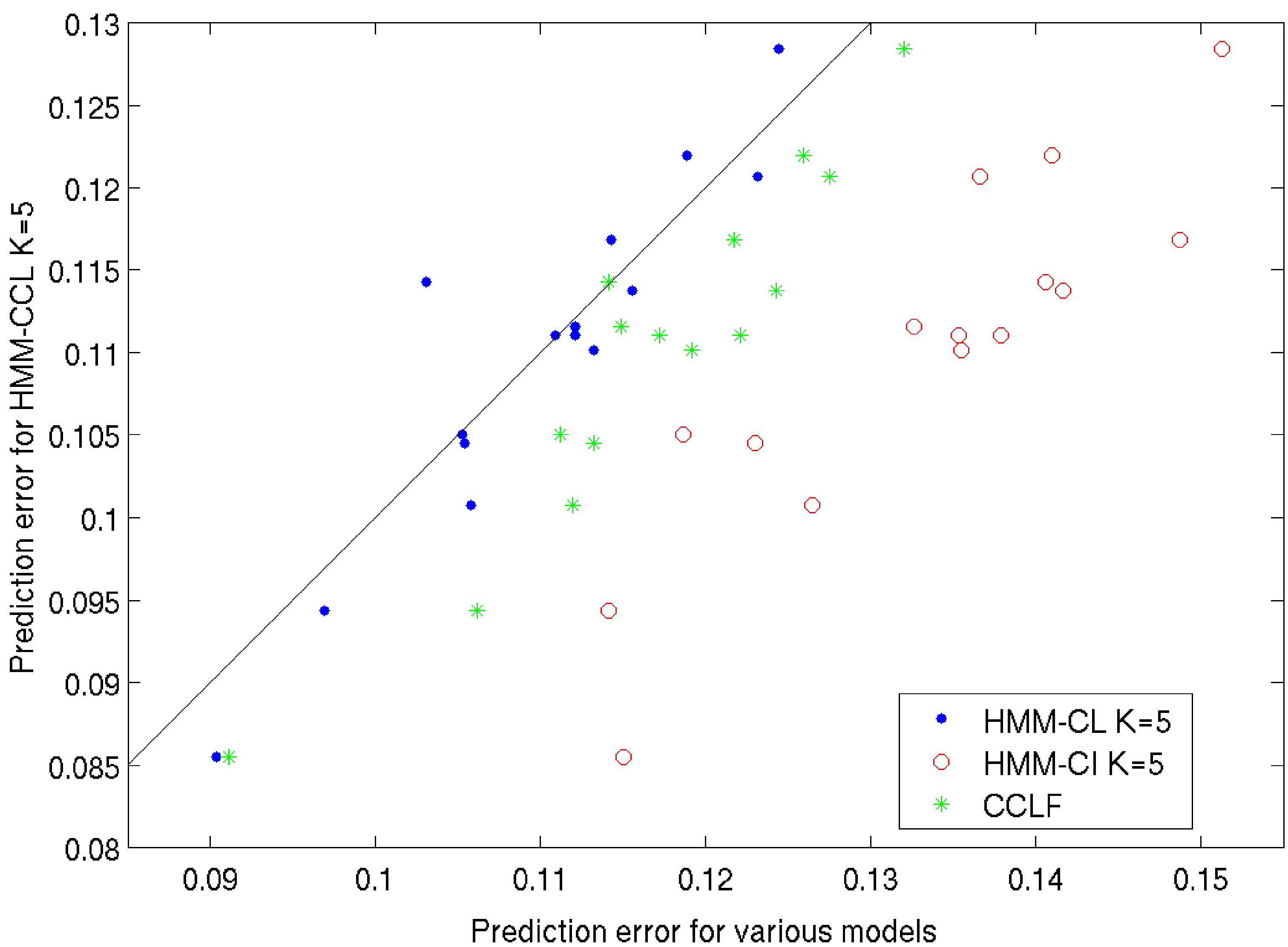
Evaluation

- HMM models with tree structures learned from historical precipitation data using EM
  - Brazil, Kenya, Senegal, Australia, Western US, ...

- Cross-validation to evaluate predictive power
  - Train on N-K seasons, predict data from other K seasons
  - Leave out single (1-day) station measurement and predict
  - Repeat, and look at average prediction accuracy

- Results
  - First-order Markov chains capture no spatial dependence
  - HMM with conditional-independence -> quite good
  - HMMs with tree-structures are most accurate

Australia (predictive error)



Example 2:

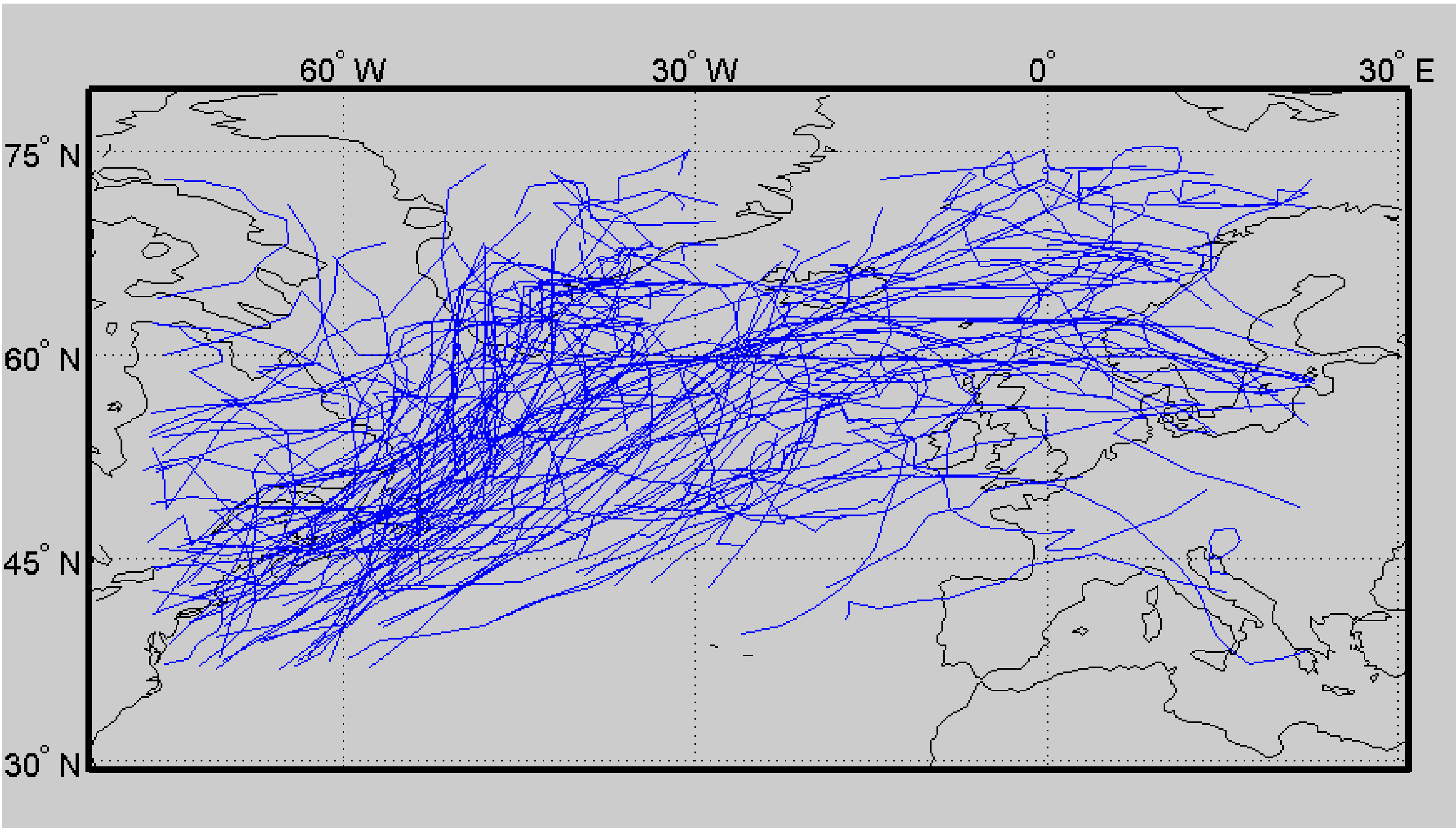
Clustering Cyclone Trajectories

Joint work with:

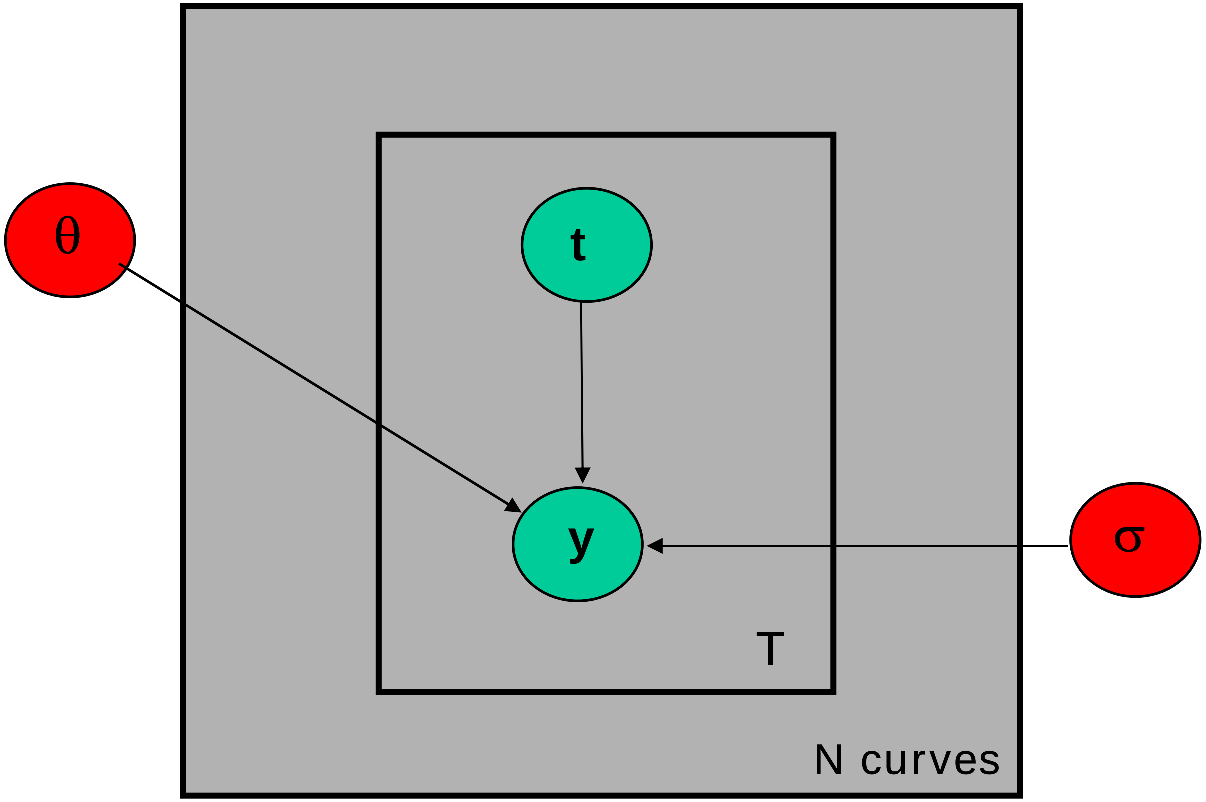
Suzana Camargo, Andy Robertson, International Research Institute for Climate Prediction

Scott Gaffney, Department of Computer Science, UC Irvine

Storm Trajectories

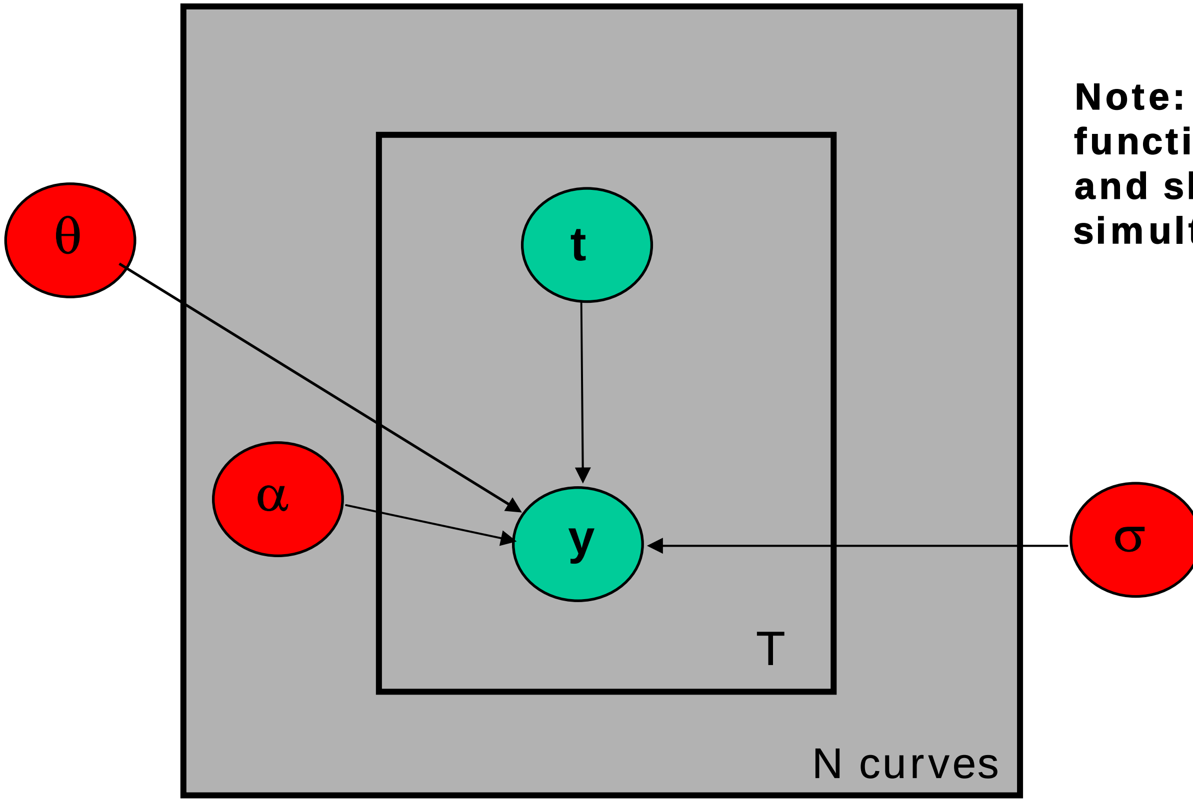


Graphical Models for Sets of Trajectories



Each curve: $P(\mathbf{y}_i \mid \mathbf{t}_i, \theta) =$ product of Gaussians

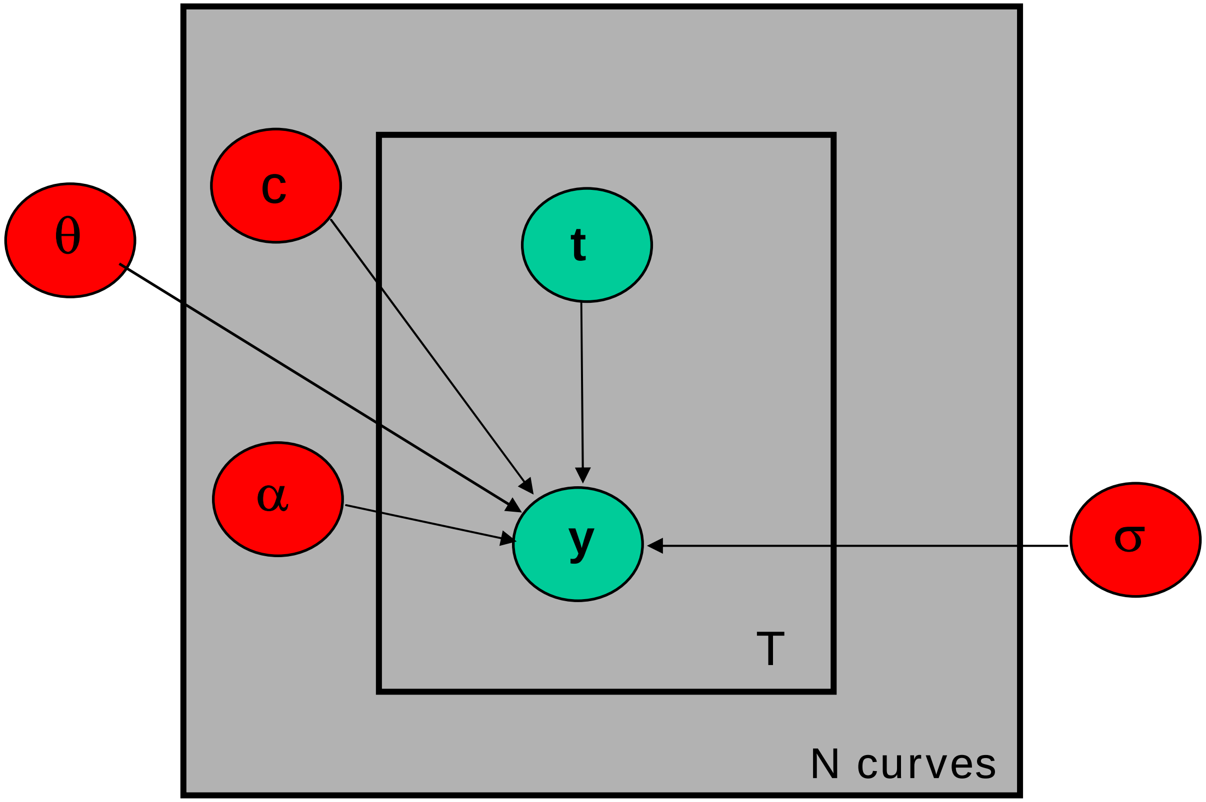
Curve-Specific Transformations



Note: we can learn function parameters and shifts simultaneously with EM

e.g., $y_i = at^2 + bt + c + \alpha_i$, $\theta = \{a, b, c, \alpha_1, \ldots \alpha_N\}$

Clustering: Mixtures of Trajectories

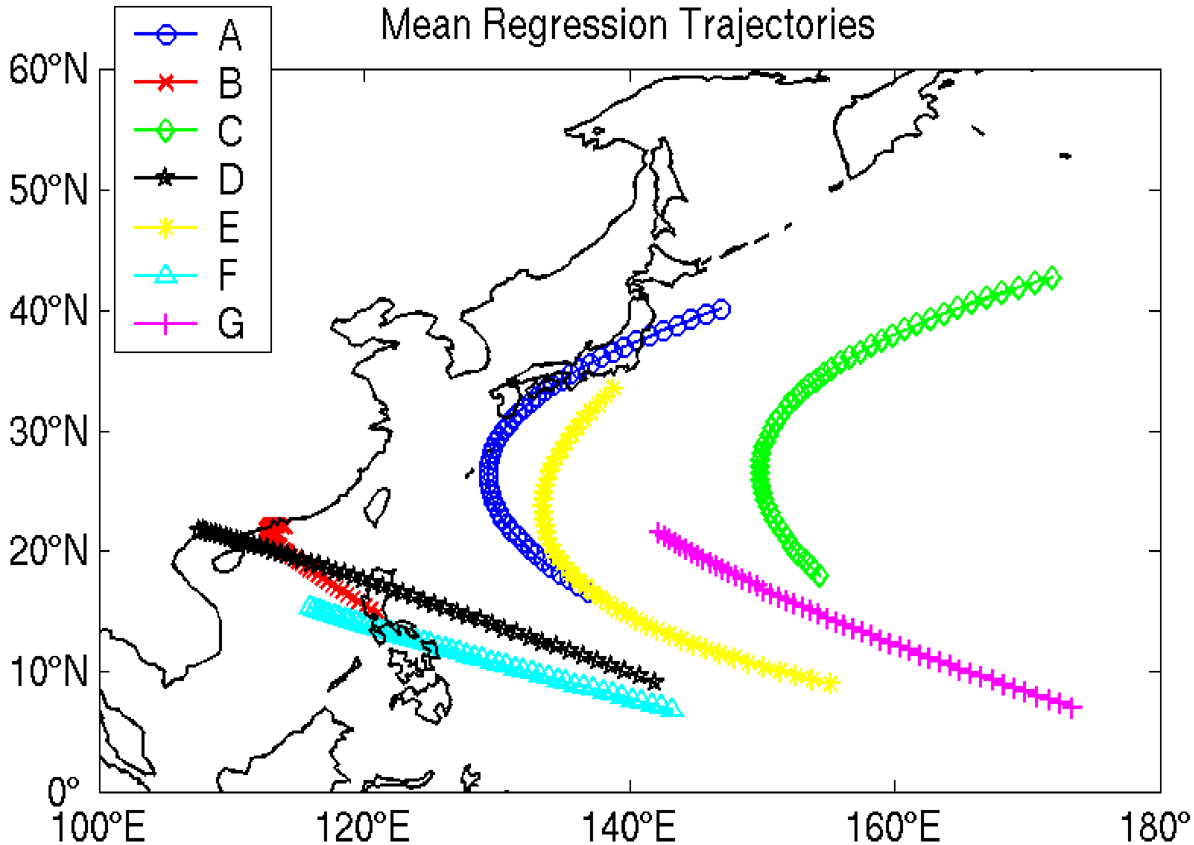


Each set of trajectory points comes from 1 of K models

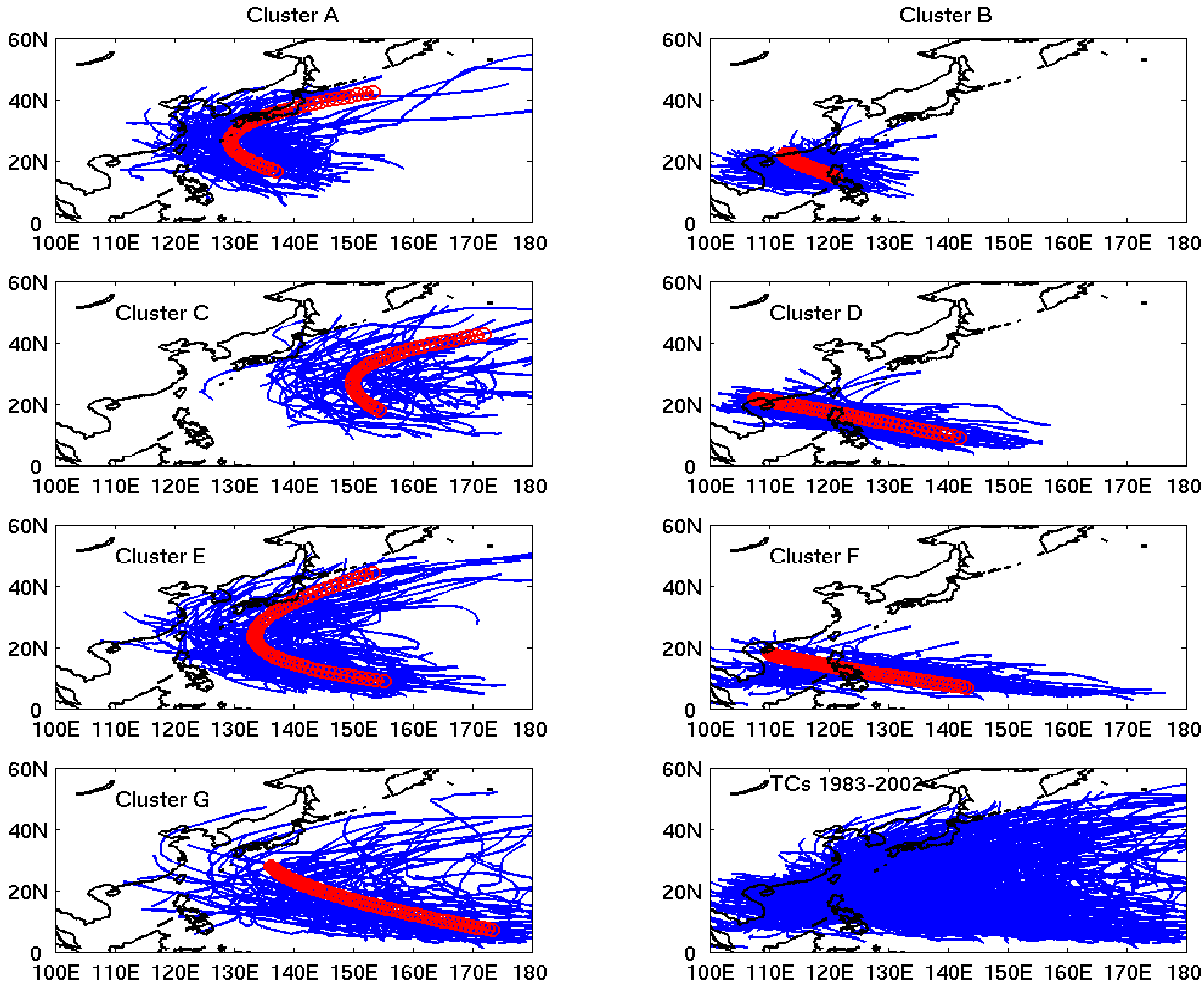
Model for group k is a Gaussian curve model

Marginal probability for a trajectory = mixture model

Cluster Shapes for Pacific Cyclones



TROPICAL CYCLONES Western North Pacific 1983-2002



Topics not discussed....

- Learning model structure from data
  - Without hidden variables -> doable
  - With hidden variables -> difficult

- Non-Gaussian models for continuous data
  - Relatively little work

- Monte-Carlo sampling techniques
  - for probability calculation and forecasting
  - E.g., sequential importance sampling (“particle filtering”)

- Prediction using model-averaging
  - Bayesian approach:
    - Estimate model-combining weights using Bayesian estimation methods
  - Empirical approach:
    - Estimate model-combining weights that lead to the best prediction

Looking to the future…

- Integration of different data sources for climate modeling
  - Temperature, precipitation, ground-cover, etc
  - Integrating satellite data with traditional data
    - e.g. MODIS data

- Leads to "large-scale structured stochastic models"
  - Multiple temporal scales, spatial scales, diffferent variables
  - Issues
    - missing data
    - data on different time-scales/spatial grids
    - Variable selection, model selection….
    - Parameter/model/forecast uncertainty

- Graphical models provide a useful framework for
  - Thinking about model structure
  - General-purpose algorithms for estimation and prediction
  - Efficient computation
  - General "language" for Bayesian modeling (e.g., BUGS)

References

- Papers from my Web page:
  - Rainfall modeling with HMMs
    - Robertson, Kirshner, Smyth, Hidden Markov models for modeling daily rainfall occurrence over Brazil, *Journal of Climate*, 17(22):4407-4424, November 2004.
  - Graphical models and HMMs
    - Smyth, Heckerman, Jordan, 1997, *Neural Computation*
- Other sources
  - Kevin Murphy:
    - Dynamic Bayesian Networks: Representation, Inference, and Learning, Phd thesis, EECS Department, UC Berkeley, 2002
    - Dynamic Bayesian Networks, draft book chapter.