Large-Deviations and Applications for Learning Tree-Structured Graphical Models

Vincent Tan

Stochastic Systems Group,
Lab of Information and Decision Systems,
Massachusetts Institute of Technology

Thesis Defense (Nov 16, 2010)

Acknowledgements

The following is joint work with:

- Alan Willsky (MIT)
- Lang Tong (Cornell)
- Animashree Anandkumar (UC Irvine)
- John Fisher (MIT)
- Sujay Sanghavi (UT Austin)
- Matt Johnson (MIT)

Outline

Outline

1. Motivation, Background and Main Contributions
2. Learning Discrete Trees Models: Error Exponent Analysis

Outline

1. Motivation, Background and Main Contributions
2. Learning Discrete Trees Models: Error Exponent Analysis
3. Learning Gaussian Trees Models: Extremal Structures

Outline

1. Motivation, Background and Main Contributions
2. Learning Discrete Trees Models: Error Exponent Analysis
3. Learning Gaussian Trees Models: Extremal Structures
4. Learning High-Dimensional Forest-Structured Models

Outline

Outline

Motivation: A Real-Life Example

- Manchester Asthma and Allergy Study (MAAS)
- More than $n \approx 1000$ children
- Number of variables $d \approx 10^6$
  - Environmental, Physiological and Genetic (SNP)

Motivation: A Real-Life Example

- Manchester Asthma and Allergy Study (MAAS)
- More than $n \approx 1000$ children
- Number of variables $d \approx 10^6$
  - Environmental, Physiological and Genetic (SNP)



www.maas.org.uk



Motivation: Modeling Large Datasets I

- How do we model such data to make useful inferences?

Simpson*, **VYFT*** et al. "Beyond Atopy: Multiple Patterns of Sensitization in Relation to Asthma in a Birth Cohort Study", Am. J. Respir. Crit. Care Med. Feb 2010.

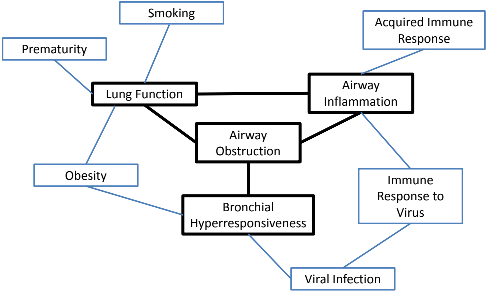
Motivation: Modeling Large Datasets I

- How do we model such data to make useful inferences?
- Model the relationships between variables by a sparse graph
- Reduce the number of interdependencies between the variables

Simpson*, **VYFT*** et al. "Beyond Atopy: Multiple Patterns of Sensitization in Relation to Asthma in a Birth Cohort Study", Am. J. Respir. Crit. Care Med. Feb 2010.

Motivation: Modeling Large Datasets I

- How do we model such data to make useful inferences?
- Model the relationships between variables by a sparse graph
- Reduce the number of interdependencies between the variables



Simpson*, **VYFT*** et al. "Beyond Atopy: Multiple Patterns of Sensitization in Relation to Asthma in a Birth Cohort Study", Am. J. Respir. Crit. Care Med. Feb 2010.

Motivation: Modeling Large Datasets II

- Reduce the dimensionality of the covariates (features) for predicting a variable for interest (e.g., asthma)
- Information-theoretic limits[†]?

[†] **VYFT**, Johnson and Willsky, "Necessary and Sufficient Conditions for Salient Subset Recovery," Intl. Symp. on Info. Theory, Jul 2010.

[‡] **VYFT**, Sanghavi, Fisher and Willsky, "Learning Graphical Models for Hypothesis Testing and Classification," IEEE Trans. on Signal Processing, Nov 2010.

Motivation: Modeling Large Datasets II

- Reduce the dimensionality of the covariates (features) for predicting a variable for interest (e.g., asthma)
- Information-theoretic limits†?
- Learning graphical models tailored specifically for hypothesis testing
- Can we learn better models in the finite-sample setting‡?

† **VYFT**, Johnson and Willsky, "Necessary and Sufficient Conditions for Salient Subset Recovery," Intl. Symp. on Info. Theory, Jul 2010.

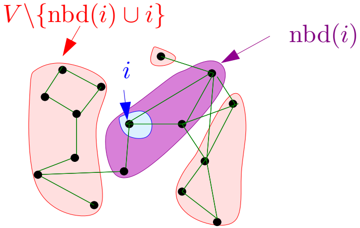
‡ **VYFT**, Sanghavi, Fisher and Willsky, "Learning Graphical Models for Hypothesis Testing and Classification," IEEE Trans. on Signal Processing, Nov 2010.

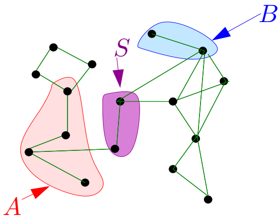
Graphical Models: Introduction

- Graph structure $G = (V, E)$ represents a multivariate distribution of a random vector $\mathbf{X} = (X_1, \ldots, X_d)$ indexed by $V = \{1, \ldots, d\}$
- Node $i \in V$ corresponds to random variable X_i
- Edge set E corresponds to conditional independencies

Graphical Models: Introduction

- Graph structure $G = (V, E)$ represents a multivariate distribution of a random vector $\mathbf{X} = (X_1, \ldots, X_d)$ indexed by $V = \{1, \ldots, d\}$
- Node $i \in V$ corresponds to random variable X_i
- Edge set E corresponds to conditional independencies





$X_i \perp\!\!\!\perp \mathbf{X}_{V\setminus\{\mathrm{nbd}(i)\cup i\}} | \mathbf{X}_{\mathrm{nbd}(i)}$

$\mathbf{X}_A \perp\!\!\!\perp \mathbf{X}_B | \mathbf{X}_S$

From Conditional Independence to Gibbs Distribution

Hammersley-Clifford Theorem (1971)

Let P be the joint pmf of graphical model Markov on $G = (V, E)$:

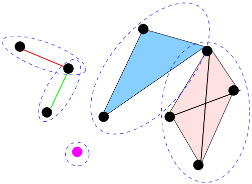
$$P(\mathbf{x}) = \frac{1}{Z} \exp\left[\sum_{c \in \mathcal{C}} \Psi_c(\mathbf{x}_c)\right]$$

From Conditional Independence to Gibbs Distribution

Hammersley-Clifford Theorem (1971)

Let P be the joint pmf of graphical model Markov on $G = (V, E)$:

$$P(\mathbf{x}) = \frac{1}{Z} \exp \left[\sum_{c \in \mathcal{C}} \Psi_c(\mathbf{x}_c) \right]$$

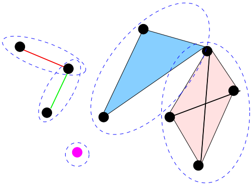


From Conditional Independence to Gibbs Distribution

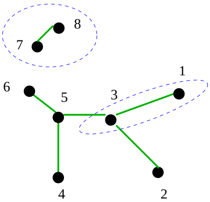
Hammersley-Clifford Theorem (1971)

Let P be the joint pmf of graphical model Markov on $G = (V, E)$:

$$P(\mathbf{x}) = \frac{1}{Z} \exp \left[\sum_{c \in \mathcal{C}} \Psi_c(\mathbf{x}_c) \right]$$



Gaussian Graphical Models

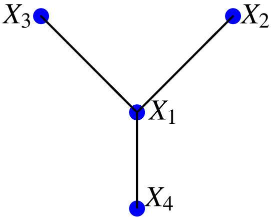


$$\begin{array}{c|cccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \hline 1 & \times & & \times & & & & & \\ 2 & & \times & \times & & \times & & & \\ 3 & \times & \times & \times & & & & & \\ 4 & & & & \times & & & & \\ 5 & & & \times & & \times & \times & & \\ 6 & & & & & \times & \times & & \\ 7 & & & & & & & \times & \times \\ 8 & & & & & & & \times & \times \end{array}$$

Dependency Graph

Inverse Covariance Matrix

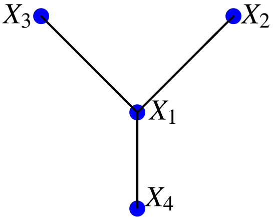
Tree-Structured Graphical Models



$$P(\mathbf{x}) = \prod_{i \in V} P_i(x_i) \prod_{(i,j) \in E} \frac{P_{i,j}(x_i, x_j)}{P_i(x_i) P_j(x_j)}$$

$$= P_1(x_1) \frac{P_{1,2}(x_1, x_2)}{P_1(x_1)} \frac{P_{1,3}(x_1, x_3)}{P_1(x_1)} \frac{P_{1,4}(x_1, x_4)}{P_1(x_1)}$$

Tree-Structured Graphical Models



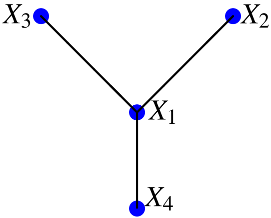
$$P(\mathbf{x}) = \prod_{i\in V} P_i(x_i) \prod_{(i,j)\in E} \frac{P_{i,j}(x_i,x_j)}{P_i(x_i)P_j(x_j)}$$

$$= P_1(x_1)\frac{P_{1,2}(x_1,x_2)}{P_1(x_1)}\frac{P_{1,3}(x_1,x_3)}{P_1(x_1)}\frac{P_{1,4}(x_1,x_4)}{P_1(x_1)}$$

Tree-structured Graphical Models: Tractable Learning and Inference

- Maximum-Likelihood learning of tree structure is tractable
  - Chow-Liu Algorithm (1968)

Tree-Structured Graphical Models



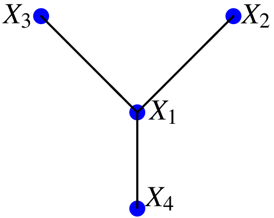
$$P(\mathbf{x}) = \prod_{i\in V} P_i(x_i) \prod_{(i,j)\in E} \frac{P_{i,j}(x_i, x_j)}{P_i(x_i)P_j(x_j)}$$

$$= P_1(x_1)\frac{P_{1,2}(x_1,x_2)}{P_1(x_1)}\frac{P_{1,3}(x_1,x_3)}{P_1(x_1)}\frac{P_{1,4}(x_1,x_4)}{P_1(x_1)}$$

Tree-structured Graphical Models: Tractable Learning and Inference

- Maximum-Likelihood learning of tree structure is tractable
  - Chow-Liu Algorithm (1968)
- Inference on Trees is tractable
  - Sum-Product Algorithm

Tree-Structured Graphical Models



$$P(\mathbf{x}) = \prod_{i \in V} P_i(x_i) \prod_{(i,j) \in E} \frac{P_{i,j}(x_i, x_j)}{P_i(x_i) P_j(x_j)}$$

$$= P_1(x_1) \frac{P_{1,2}(x_1, x_2)}{P_1(x_1)} \frac{P_{1,3}(x_1, x_3)}{P_1(x_1)} \frac{P_{1,4}(x_1, x_4)}{P_1(x_1)}$$

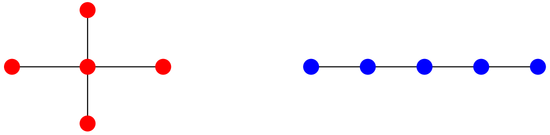
Tree-structured Graphical Models: Tractable Learning and Inference

- Maximum-Likelihood learning of tree structure is tractable
  - Chow-Liu Algorithm (1968)
- Inference on Trees is tractable
  - Sum-Product Algorithm

Which other classes of graphical models are tractable for learning?

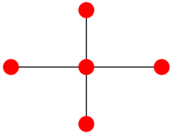
Main Contributions in Thesis: I

Error Exponent Analysis of Tree Structure Learning (Ch. 3 and 4)

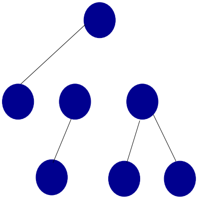


Main Contributions in Thesis: I

Error Exponent Analysis of Tree Structure Learning (Ch. 3 and 4)



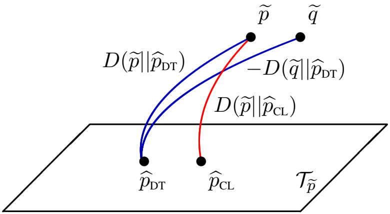
High-Dimensional Structure Learning for Forest Models (Ch. 5)



Main Contributions in Thesis: II

Learning Graphical Models for Hypothesis Testing (Ch. 6)

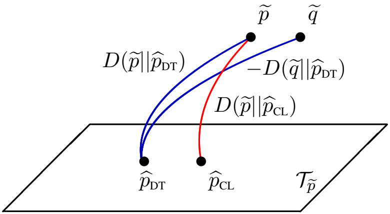
- Devised algorithms for learning trees for hypothesis testing



Main Contributions in Thesis: II

Learning Graphical Models for Hypothesis Testing (Ch. 6)

- Devised algorithms for learning trees for hypothesis testing



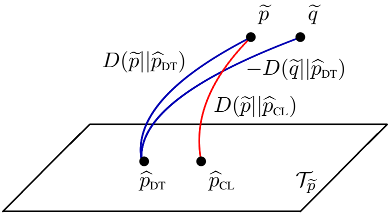
Information-Theoretic Limits for Salient Subset Recovery (Ch. 7)

- Devised necessary and sufficient conditions for estimating of salient set of features

Main Contributions in Thesis: II

Learning Graphical Models for Hypothesis Testing (Ch. 6)

- Devised algorithms for learning trees for hypothesis testing



Information-Theoretic Limits for Salient Subset Recovery (Ch. 7)

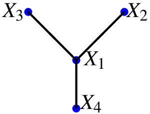
- Devised necessary and sufficient conditions for estimating of salient set of features

We will focus on Chapters 3 - 5 here. See thesis for Chapters 6 and 7.

Outline

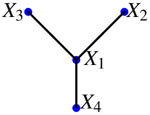
Motivation

ML learning of tree structure given i.i.d. $\mathcal{X}^d$-valued samples



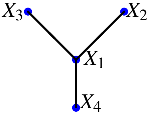
Motivation

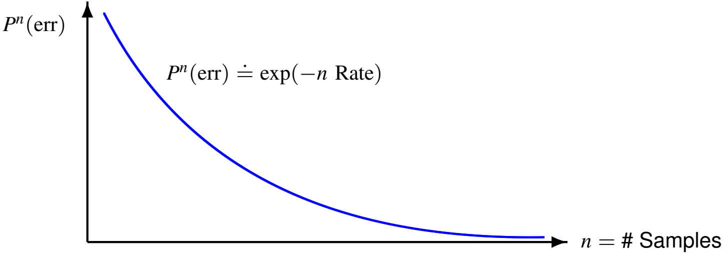
ML learning of tree structure given i.i.d. $\mathcal{X}^d$-valued samples



Motivation

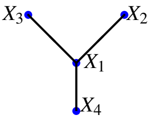
ML learning of tree structure given i.i.d. $\mathcal{X}^d$-valued samples

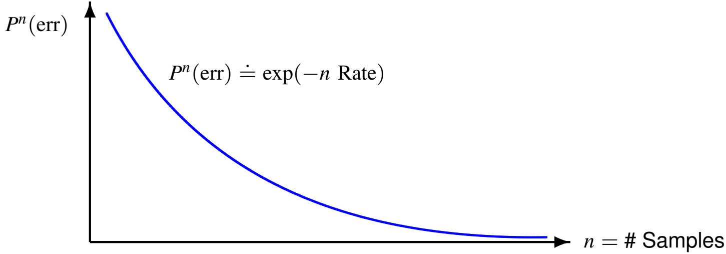




Motivation

ML learning of tree structure given i.i.d. $\mathcal{X}^d$-valued samples

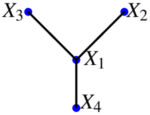


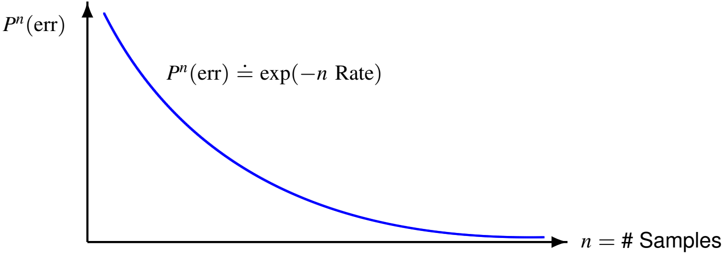


- When does the error probability decay exponentially?

Motivation

ML learning of tree structure given i.i.d. $\mathcal{X}^d$-valued samples

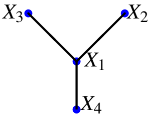


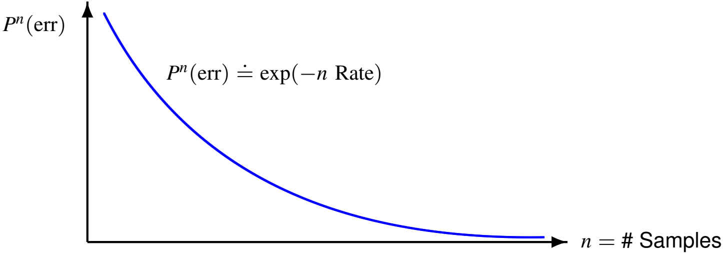


- When does the error probability decay exponentially?
- What is the exact rate of decay of the probability of error?

Motivation

ML learning of tree structure given i.i.d. $\mathcal{X}^d$-valued samples





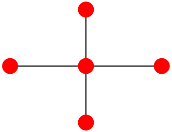
- When does the error probability decay exponentially?
- What is the exact rate of decay of the probability of error?
- How does the error exponent depend on the parameters and structure of the true distribution?

Main Contributions

- Discrete case:
  - Provide the exact rate of decay for a given P
  - Rate of decay $\approx$ SNR for learning

Main Contributions

- Discrete case:
  - Provide the exact rate of decay for a given P
  - Rate of decay $\approx$ SNR for learning
- Gaussian case:
  - Extremal structures: Star (worst) and chain (best) for learning

Related Work in Structure Learning

- ML for trees: Max-weight spanning tree with mutual information edge weights (Chow & Liu 1968)
- Causal dependence trees: directed mutual information (Quinn, Coleman & Kiyavash 2010)
- Convex relaxation methods: ℓ_1 regularization
  - Gaussian graphical models (Meinshausen and Buehlmann 2006)
  - Logistic regression for Ising models (Ravikumar et al. 2010)
- Learning thin junction trees through conditional mutual information tests (Chechetka et al. 2007)
- Conditional independence tests for bounded degree graphs (Bresler et al. 2008)

Related Work in Structure Learning

- ML for trees: Max-weight spanning tree with mutual information edge weights (Chow & Liu 1968)
- Causal dependence trees: directed mutual information (Quinn, Coleman & Kiyavash 2010)
- Convex relaxation methods: ℓ_1 regularization
  - Gaussian graphical models (Meinshausen and Buehlmann 2006)
  - Logistic regression for Ising models (Ravikumar et al. 2010)
- Learning thin junction trees through conditional mutual information tests (Chechetka et al. 2007)
- Conditional independence tests for bounded degree graphs (Bresler et al. 2008)

We obtain and analyze error exponents for the ML learning of trees (and extensions to forests)

ML Learning of Trees (Chow-Liu) I

Samples $\mathbf{x}^n = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ drawn i.i.d. from $P \in \mathcal{P}(\mathcal{X}^d)$, $\mathcal{X}$ is finite

ML Learning of Trees (Chow-Liu) I

Samples $\mathbf{x}^n = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ drawn i.i.d. from $P \in \mathcal{P}(\mathcal{X}^d)$, $\mathcal{X}$ is finite

- Solve the ML problem given the data $\mathbf{x}^n$

$$P_{\mathrm{ML}} \triangleq \operatorname*{argmax}_{Q \in \mathrm{Trees}} \frac{1}{n} \sum_{k=1}^{n} \log Q(\mathbf{x}_k)$$

Samples $\mathbf{x}^n = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ drawn i.i.d. from $P \in \mathcal{P}(\mathcal{X}^d)$, $\mathcal{X}$ is finite

- Solve the ML problem given the data $\mathbf{x}^n$

$$P_{\mathrm{ML}} \triangleq \operatorname*{argmax}_{Q \in \mathrm{Trees}} \frac{1}{n} \sum_{k=1}^{n} \log Q(\mathbf{x}_k)$$

- Denote $\widehat{P}(\mathbf{a}) = \widehat{P}(\mathbf{a}; \mathbf{x}^n)$ as the empirical distribution of $\mathbf{x}^n$

ML Learning of Trees (Chow-Liu) I

Samples $\mathbf{x}^n = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ drawn i.i.d. from $P \in \mathcal{P}(\mathcal{X}^d)$, $\mathcal{X}$ is finite

- Solve the ML problem given the data $\mathbf{x}^n$

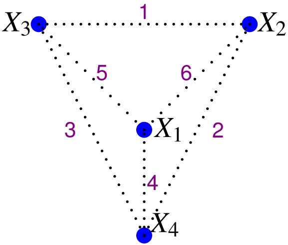
$$P_{\mathrm{ML}} \triangleq \operatorname*{argmax}_{Q \in \mathrm{Trees}} \frac{1}{n} \sum_{k=1}^{n} \log Q(\mathbf{x}_k)$$

- Denote $\widehat{P}(\mathbf{a}) = \widehat{P}(\mathbf{a}; \mathbf{x}^n)$ as the empirical distribution of $\mathbf{x}^n$
- Reduces to a max-weight spanning tree problem (Chow-Liu 1968)

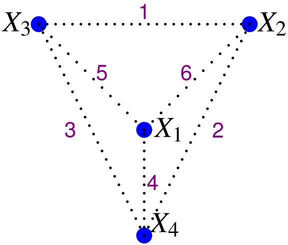
$$E_{\mathrm{ML}} = \operatorname*{argmax}_{E_Q \in \mathrm{Trees}} \sum_{e \in E_Q} I(\widehat{P}_e)$$

  - $\widehat{P}_e$ is the marginal of the empirical on $e = (i, j)$
  - $I(\widehat{P}_e)$ is the mutual information of the empirical $\widehat{P}_e$

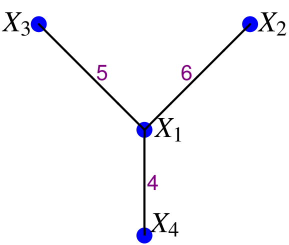
ML Learning of Trees (Chow-Liu) II



ML Learning of Trees (Chow-Liu) II

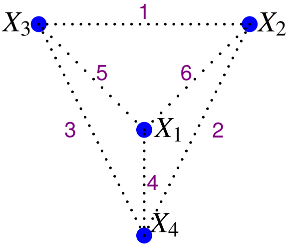


True MI $\{I(P_e)\}$

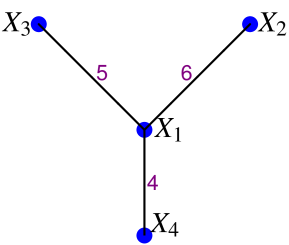


Max-weight spanning tree E_P

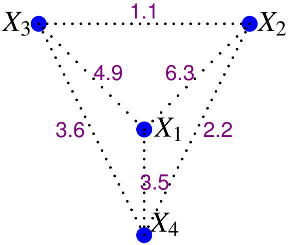
ML Learning of Trees (Chow-Liu) II



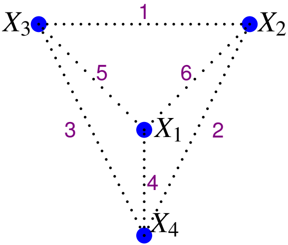
True MI $\{I(P_e)\}$



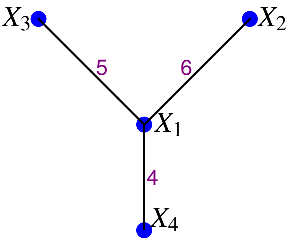
Max-weight spanning tree E_P



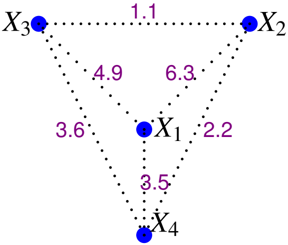
ML Learning of Trees (Chow-Liu) II



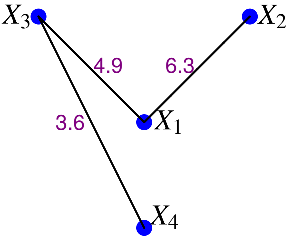
True MI $\{I(P_e)\}$



Max-weight spanning tree E_P



Empirical MI $\{I(\widehat{P}_e)\}$ from $\mathbf{x}^n$



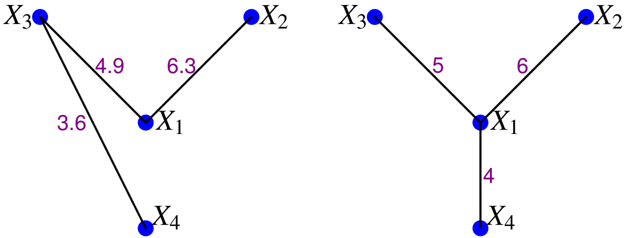
Max-weight spanning tree $E_{\text{ML}} \neq E_P$

Problem Statement

- Define P_{ML} to be ML tree-structured distribution with edge set E_{ML} and the error event is $\{E_{\text{ML}} \neq E_P\}$

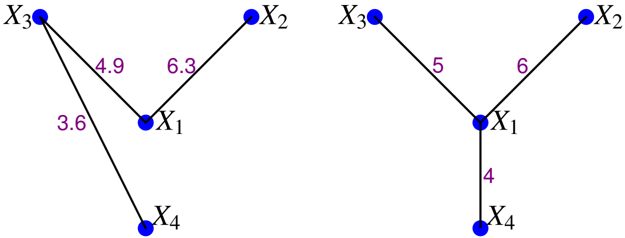
Problem Statement

- Define P_{ML} to be ML tree-structured distribution with edge set E_{ML} and the error event is $\{E_{\text{ML}} \neq E_P\}$



Problem Statement

- Define P_{ML} to be ML tree-structured distribution with edge set E_{ML} and the error event is $\{E_{\mathrm{ML}} \neq E_P\}$

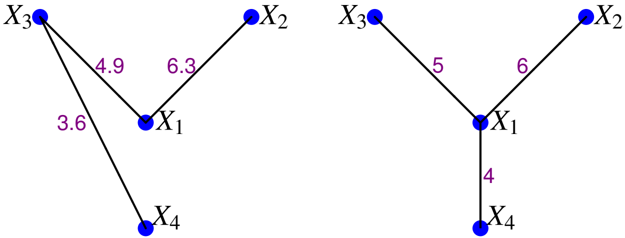


- Find the error exponent K_P:

$$K_P \triangleq \lim_{n\to\infty} -\frac{1}{n} \log P^n \left(E_{\mathrm{ML}} \neq E_P\right)$$

Problem Statement

- Define P_{ML} to be ML tree-structured distribution with edge set E_{ML} and the error event is $\{E_{\mathrm{ML}} \neq E_P\}$

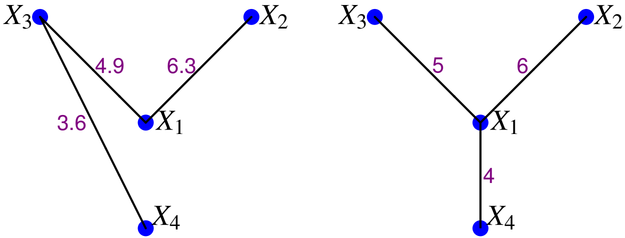


- Find the error exponent K_P:

$$K_P \triangleq \lim_{n\to\infty} -\frac{1}{n} \log P^n\left(E_{\mathrm{ML}} \neq E_P\right) \qquad P^n\left(E_{\mathrm{ML}} \neq E_P\right) \doteq \exp(-nK_P)$$

Problem Statement

- Define P_{ML} to be ML tree-structured distribution with edge set E_{ML} and the error event is $\{E_{\text{ML}} \neq E_P\}$



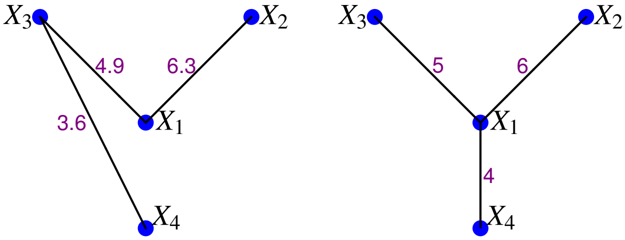
- Find the error exponent K_P:

$$K_P \triangleq \lim_{n\to\infty} -\frac{1}{n} \log P^n\left(E_{\text{ML}} \neq E_P\right) \qquad P^n\left(E_{\text{ML}} \neq E_P\right) \doteq \exp(-nK_P)$$

- Naïvely, what could we do to compute K_P?

Problem Statement

- Define P_{ML} to be ML tree-structured distribution with edge set E_{ML} and the error event is $\{E_{\text{ML}} \neq E_P\}$



- Find the error exponent K_P:

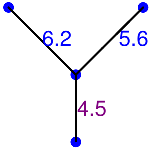
$$K_P \triangleq \lim_{n\to\infty} -\frac{1}{n} \log P^n\left(E_{\text{ML}} \neq E_P\right) \qquad P^n\left(E_{\text{ML}} \neq E_P\right) \doteq \exp(-nK_P)$$

- Naïvely, what could we do to compute K_P?
  I-projections onto all trees?

The Crossover Rate I

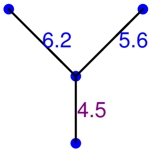
Correct Structure

True MI $I(P_e)$	6	5	4	3	2	1
Emp MI $I(\widehat{P}_e)$	6.2	5.6	4.5	2.8	2.2	1.1



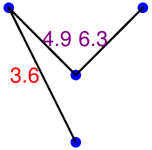
Correct Structure

True MI $I(P_e)$	6	5	4	3	2	1
Emp MI $I(\widehat{P}_e)$	6.2	5.6	4.5	2.8	2.2	1.1



Incorrect Structure!

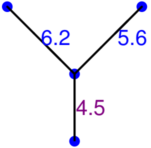
True MI $I(P_e)$	6	5	4	3	2	1
Emp MI $I(\widehat{P}_e)$	6.3	4.9	3.5	3.6	2.2	1.1



The Crossover Rate I

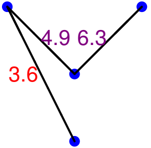
Correct Structure

True MI $I(P_e)$	6	5	4	3	2	1
Emp MI $I(\widehat{P}_e)$	6.2	5.6	4.5	2.8	2.2	1.1



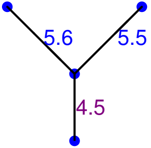
Incorrect Structure!

True MI $I(P_e)$	6	5	4	3	2	1
Emp MI $I(\widehat{P}_e)$	6.3	4.9	3.5	3.6	2.2	1.1

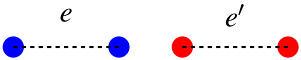


Structure Unaffected

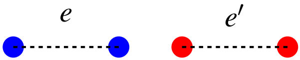
True MI $I(P_e)$	6	5	4	3	2	1
Emp MI $I(\widehat{P}_e)$	5.5	5.6	4.5	3.0	2.2	1.1



The Crossover Rate I



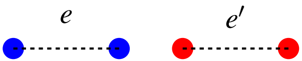
The Crossover Rate I



Given two node pairs $e, e' \in \binom{V}{2}$ with joint distribution $P_{e,e'} \in \mathcal{P}(\mathcal{X}^4)$, s.t.

$$I(P_e) > I(P_{e'}).$$

The Crossover Rate I



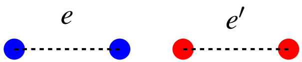
Given two node pairs $e, e' \in \binom{V}{2}$ with joint distribution $P_{e,e'} \in \mathcal{P}(\mathcal{X}^4)$, s.t.

$$I(P_e) > I(P_{e'}).$$

Consider the crossover event of the empirical MI

$$\{I(\widehat{P}_e) \leq I(\widehat{P}_{e'})\}$$

The Crossover Rate I



Given two node pairs $e, e' \in \binom{V}{2}$ with joint distribution $P_{e,e'} \in \mathcal{P}(\mathcal{X}^4)$, s.t.

$$I(P_e) > I(P_{e'}).$$

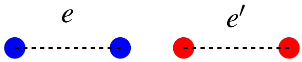
Consider the crossover event of the empirical MI

$$\{I(\widehat{P}_e) \leq I(\widehat{P}_{e'})\}$$

Def: Crossover Rate

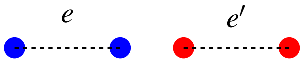
$$J_{e,e'} \triangleq \lim_{n\to\infty} -\frac{1}{n} \log P^n \left(I(\widehat{P}_e) \leq I(\widehat{P}_{e'}) \right)$$

The Crossover Rate II



$I(P_e) > I(P_{e'})$ $\qquad$ $\{I(\widehat{P}_e) \le I(\widehat{P}_{e'})\}$

The Crossover Rate II



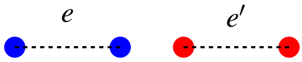
$$I(P_e) > I(P_{e'}) \qquad \{I(\widehat{P}_e) \le I(\widehat{P}_{e'})\}$$

Proposition

The crossover rate for empirical mutual informations is

$$J_{e,e'} = \min_{Q \in \mathcal{P}(\mathcal{X}^4)} \left\{ D(Q \,||\, P_{e,e'}) : I(Q_{e'}) = I(Q_e) \right\}$$

The Crossover Rate II



$I(P_e) > I(P_{e'}) \qquad \{I(\widehat{P}_e) \le I(\widehat{P}_{e'})\}$

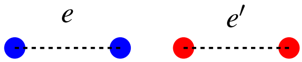
Proposition

The crossover rate *for empirical mutual informations is*

$$J_{e,e'} = \min_{Q \in \mathcal{P}(\mathcal{X}^4)} \Big\{ D(Q \,||\, P_{e,e'}) : I(Q_{e'}) = I(Q_e) \Big\}$$

$\mathcal{P}(\mathcal{X}^4)$

The Crossover Rate II



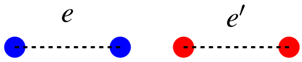
$I(P_e) > I(P_{e'})$ $\qquad$ $\{I(\widehat{P}_e) \le I(\widehat{P}_{e'})\}$

Proposition

The crossover rate for empirical mutual informations is

$$J_{e,e'} = \min_{Q \in \mathcal{P}(\mathcal{X}^4)} \left\{ D(Q \,||\, P_{e,e'}) : I(Q_{e'}) = I(Q_e) \right\}$$

$\mathcal{P}(\mathcal{X}^4)$ $\qquad$ $\bullet\ P_{e,e'}$

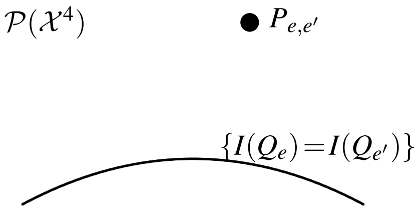


$I(P_e) > I(P_{e'})$ $\{I(\widehat{P}_e) \le I(\widehat{P}_{e'})\}$

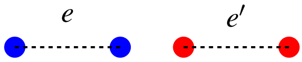
Proposition

The crossover rate for empirical mutual informations is

$$J_{e,e'} = \min_{Q \in \mathcal{P}(\mathcal{X}^4)} \left\{ D(Q \,||\, P_{e,e'}) : I(Q_{e'}) = I(Q_e) \right\}$$



The Crossover Rate II

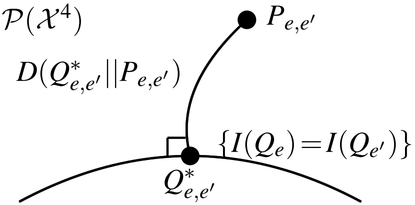


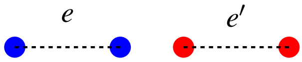
$I(P_e) > I(P_{e'})$ $\quad$ $\{I(\widehat{P}_e) \le I(\widehat{P}_{e'})\}$

Proposition

The crossover rate for empirical mutual informations is

$$J_{e,e'} = \min_{Q \in \mathcal{P}(\mathcal{X}^4)} \Big\{ D(Q \,||\, P_{e,e'}) : I(Q_{e'}) = I(Q_e) \Big\}$$



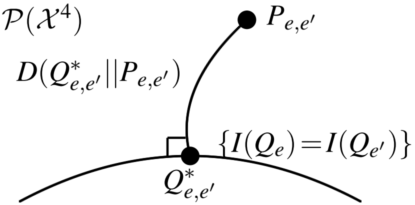


$I(P_e) > I(P_{e'})$ $\quad\quad$ $\{I(\widehat{P}_e) \le I(\widehat{P}_{e'})\}$

Proposition

The crossover rate for empirical mutual informations is

$$J_{e,e'} = \min_{Q \in \mathcal{P}(\mathcal{X}^4)} \left\{ D(Q \,||\, P_{e,e'}) : I(Q_{e'}) = I(Q_e) \right\}$$



- I-projection (Csiszár)
- Sanov's Theorem
- Exact but not intuitive
- Non-Convex

Error Exponent for Structure Learning I

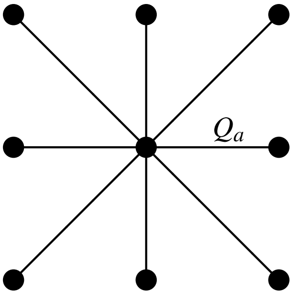
How to calculate the error exponent K_P with the crossover rates $J_{e,e'}$?

Error Exponent for Structure Learning I

How to calculate the error exponent K_P with the crossover rates $J_{e,e'}$?

Easy only in some very special cases

- "Star" graph with $I(Q_a) > I(Q_b) > 0$
- There is a unique crossover rate
- The unique crossover rate is the error exponent

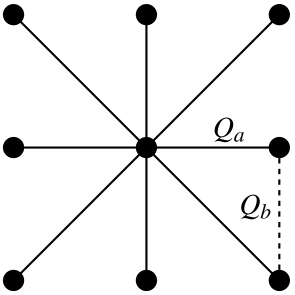


Error Exponent for Structure Learning I

How to calculate the error exponent K_P with the crossover rates $J_{e,e'}$?

Easy only in some very special cases

- "Star" graph with $I(Q_a) > I(Q_b) > 0$
- There is a unique crossover rate
- The unique crossover rate is the error exponent

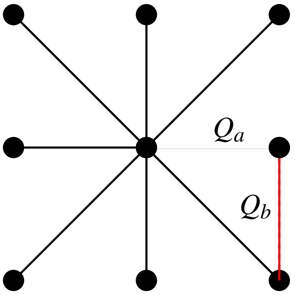


Error Exponent for Structure Learning I

How to calculate the error exponent K_P with the crossover rates $J_{e,e'}$?

Easy only in some very special cases

- "Star" graph with $I(Q_a) > I(Q_b) > 0$
- There is a unique crossover rate
- The unique crossover rate is the error exponent

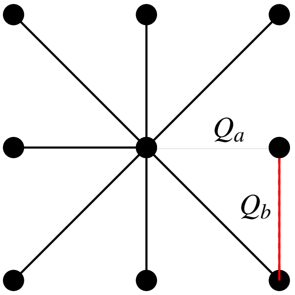


Error Exponent for Structure Learning I

How to calculate the error exponent K_P with the crossover rates $J_{e,e'}$?

Easy only in some very special cases

- "Star" graph with $I(Q_a) > I(Q_b) > 0$
- There is a unique crossover rate
- The unique crossover rate is the error exponent



$$K_P = \min_{R \in \mathcal{P}(\mathcal{X}^4)} \left\{ D(R \,||\, Q_{a,b}) : I(R_e) = I(R_{e'}) \right\}$$

Error Exponent for Structure Learning II

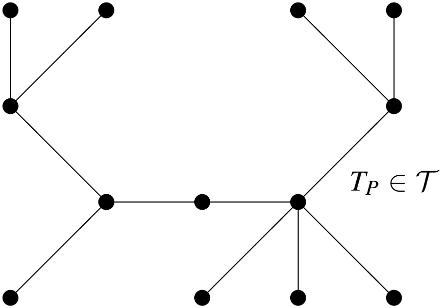
A large deviation is done in the least unlikely of all unlikely ways.

– “Large deviations” by F. Den Hollander

Error Exponent for Structure Learning II

A large deviation is done in the least unlikely of all unlikely ways.

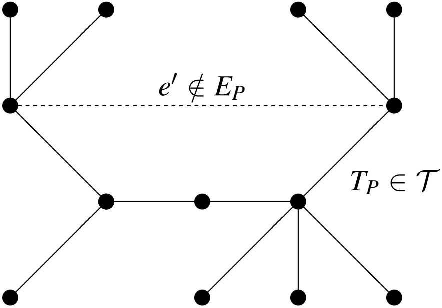
– "Large deviations" by F. Den Hollander



Error Exponent for Structure Learning II

A large deviation is done in the least unlikely of all unlikely ways.

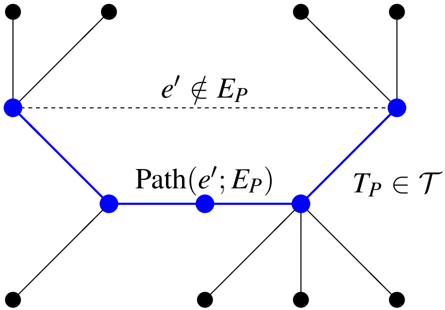
– "Large deviations" by F. Den Hollander



Error Exponent for Structure Learning II

A large deviation is done in the least unlikely of all unlikely ways.

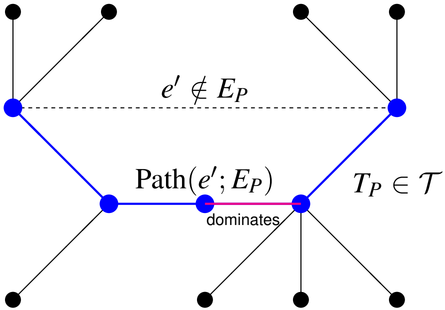
– "Large deviations" by F. Den Hollander



Error Exponent for Structure Learning II

A large deviation is done in the least unlikely of all unlikely ways.

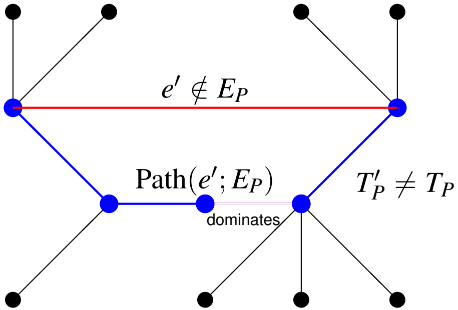
– "Large deviations" by F. Den Hollander



Error Exponent for Structure Learning II

A large deviation is done in the least unlikely of all unlikely ways.

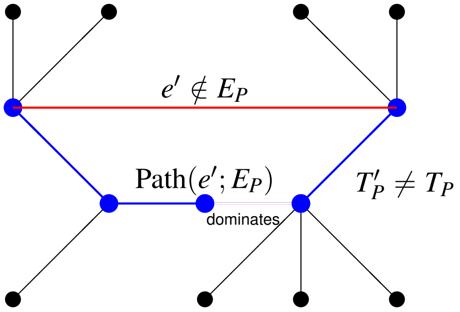
– "Large deviations" by F. Den Hollander



Error Exponent for Structure Learning II

A large deviation is done in the least unlikely of all unlikely ways.

– "Large deviations" by F. Den Hollander



Theorem (Error Exponent)

$$K_P = \min_{e' \notin E_P} \left(\min_{e \in \mathrm{Path}(e'; E_P)} J_{e,e'} \right)$$

Error Exponent for Structure Learning III

$$P^n\left(E_{\mathrm{ML}} \neq E_P\right) \doteq \exp\left[-n \min_{e' \notin E_P}\left(\min_{e \in \mathrm{Path}(e';E_P)} J_{e,e'}\right)\right]$$

Error Exponent for Structure Learning III

$$P^n\left(E_{\mathrm{ML}} \neq E_P\right) \doteq \exp\left[-n \min_{e' \notin E_P} \left(\min_{e \in \mathrm{Path}(e';E_P)} J_{e,e'} \right)\right]$$

We have a finite-sample result too! See thesis

Error Exponent for Structure Learning III

$$P^n\left(E_{\mathrm{ML}} \neq E_P\right) \doteq \exp\left[-n \min_{e' \notin E_P}\left(\min_{e \in \mathrm{Path}(e';E_P)} J_{e,e'}\right)\right]$$

We have a finite-sample result too! See thesis

Proposition

The following statements are equivalent:

(a) *The error probability decays exponentially, i.e.,* $K_P > 0$

(b) T_P *is a connected tree, i.e., not a proper forest*

Error Exponent for Structure Learning III

$$P^n\left(E_{\text{ML}} \neq E_P\right) \doteq \exp\left[-n \min_{e' \notin E_P} \left(\min_{e \in \text{Path}(e';E_P)} J_{e,e'} \right)\right]$$

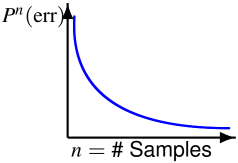
We have a finite-sample result too! See thesis

Proposition

The following statements are equivalent:

(a) *The error probability decays exponentially, i.e.,* $K_P > 0$

(b) T_P *is a connected tree, i.e., not a proper forest*



Error Exponent for Structure Learning III

$$P^n\left(E_{\mathrm{ML}} \neq E_P\right) \doteq \exp\left[-n \min_{e' \notin E_P}\left(\min_{e \in \mathrm{Path}(e';E_P)} J_{e,e'}\right)\right]$$

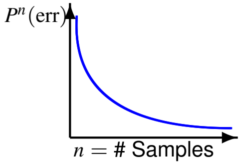
We have a finite-sample result too! See thesis

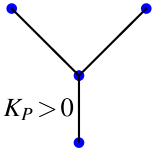
Proposition

The following statements are equivalent:

(a) *The error probability decays exponentially, i.e., $K_P > 0$*

(b) *T_P is a connected tree, i.e., not a proper forest*





Error Exponent for Structure Learning III

$$P^n\left(E_{\mathsf{ML}} \neq E_P\right) \doteq \exp\left[-n \min_{e' \notin E_P}\left(\min_{e \in \mathrm{Path}(e';E_P)} J_{e,e'}\right)\right]$$

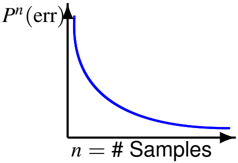
We have a finite-sample result too! See thesis

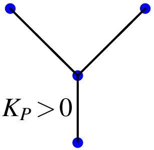
Proposition

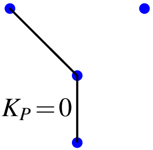
The following statements are equivalent:

(a) *The error probability decays exponentially, i.e.,* $K_P > 0$

(b) T_P *is a connected tree, i.e., not a proper forest*







Approximating The Crossover Rate I

- Def: Very-noisy learning condition on $P_{e,e'}$

$$P_e \approx P_{e'}$$

Approximating The Crossover Rate I

- Def: Very-noisy learning condition on $P_{e,e'}$

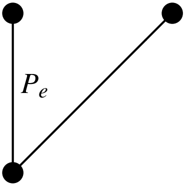
$$P_e \approx P_{e'}$$

$$I(P_e) \approx I(P_{e'})$$

Approximating The Crossover Rate I

- Def: Very-noisy learning condition on $P_{e,e'}$

$$P_e \approx P_{e'}$$



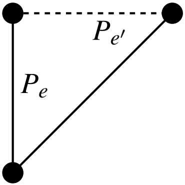
$$I(P_e) \approx I(P_{e'})$$

Approximating The Crossover Rate I

- Def: Very-noisy learning condition on $P_{e,e'}$

$$P_e \approx P_{e'}$$

$$I(P_e) \approx I(P_{e'})$$

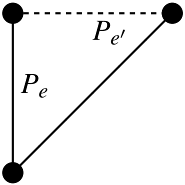


Approximating The Crossover Rate I

- Def: Very-noisy learning condition on $P_{e,e'}$

$$P_e \approx P_{e'}$$

$$I(P_e) \approx I(P_{e'})$$



- Euclidean Information Theory [Borade & Zheng '08]:

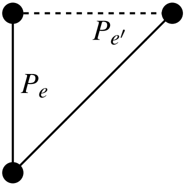
$$P \approx Q \quad \Rightarrow \quad D(P\,||\,Q) \approx \frac{1}{2}\sum_a \frac{(P(a)-Q(a))^2}{P(a)}$$

Approximating The Crossover Rate I

- Def: Very-noisy learning condition on $P_{e,e'}$

$P_e \approx P_{e'}$

$I(P_e) \approx I(P_{e'})$



- Euclidean Information Theory [Borade & Zheng '08]:

$$P \approx Q \quad \Rightarrow \quad D(P\,||\,Q) \approx \frac{1}{2}\sum_a \frac{(P(a)-Q(a))^2}{P(a)}$$

- Def: Given a $P_e = P_{i,j}$ the information density is

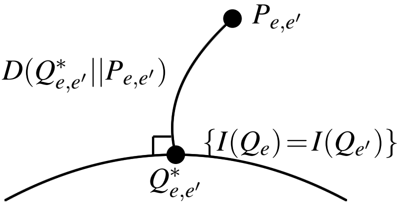
$$S_e(X_i;X_j) \triangleq \log \frac{P_{i,j}(X_i,X_j)}{P_i(X_i)P_j(X_j)}, \qquad \mathbb{E}[S_e] = I(P_e).$$

Approximating The Crossover Rate II

Convexifying the optimization problem by linearizing constraints

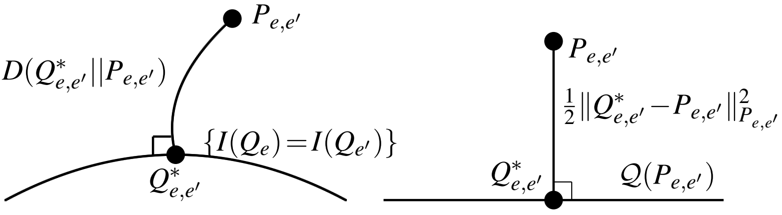
Approximating The Crossover Rate II

Convexifying the optimization problem by linearizing constraints



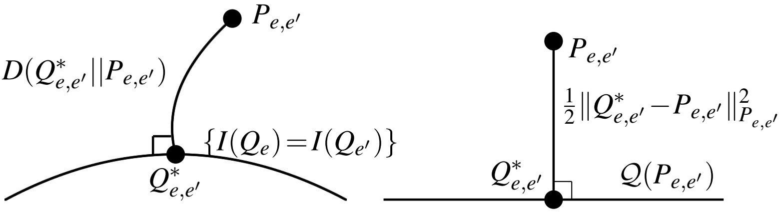
Approximating The Crossover Rate II

Convexifying the optimization problem by linearizing constraints



Approximating The Crossover Rate II

Convexifying the optimization problem by linearizing constraints

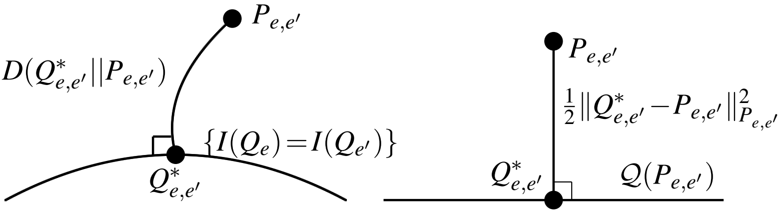


Theorem (Euclidean Approximation of Crossover Rate)

$$\widetilde{J}_{e,e'} = \frac{(I(P_{e'}) - I(P_e))^2}{2\,\mathrm{Var}(S_{e'} - S_e)}$$

Approximating The Crossover Rate II

Convexifying the optimization problem by linearizing constraints

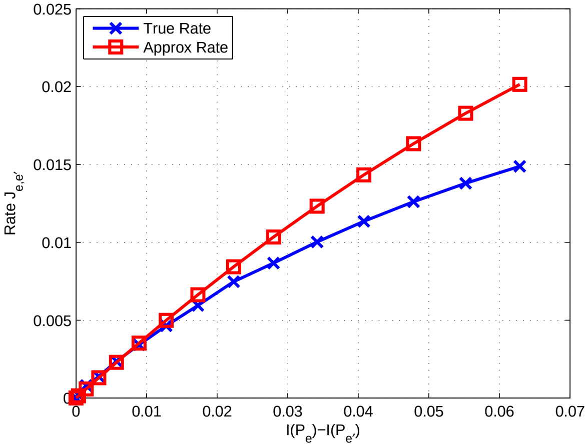


Theorem (Euclidean Approximation of Crossover Rate)

$$\widetilde{J}_{e,e'} = \frac{(I(P_{e'}) - I(P_e))^2}{2\,\mathrm{Var}(S_{e'} - S_e)} = \frac{(\mathbb{E}[S_{e'} - S_e])^2}{2\,\mathrm{Var}(S_{e'} - S_e)} = \frac{1}{2}\text{SNR}$$

The Crossover Rate

How good is the approximation? We consider a binary model



Remarks for Learning Discrete Trees

- Characterized precisely the error exponent for structure learning

$$P^n\left(E_{\mathrm{ML}} \neq E_P\right) \doteq \exp(-nK_P)$$

VYFT, A. Anandkumar, L. Tong, A. S. Willsky "A Large-Deviation Analysis of the Maximum-Likelihood Learning of Markov Tree Structures," ISIT 2009, submitted to IEEE Trans. on Information Theory, revised in Oct 2010.

Remarks for Learning Discrete Trees

- Characterized precisely the error exponent for structure learning

$$P^n\left(E_{\mathrm{ML}} \neq E_P\right) \doteq \exp(-nK_P)$$

- Analysis tools include the method of types (large-deviations) and simple properties of trees

VYFT, A. Anandkumar, L. Tong, A. S. Willsky "A Large-Deviation Analysis of the Maximum-Likelihood Learning of Markov Tree Structures," ISIT 2009, submitted to IEEE Trans. on Information Theory, revised in Oct 2010.

Remarks for Learning Discrete Trees

- Characterized precisely the error exponent for structure learning

$$P^n\left(E_{\mathrm{ML}} \neq E_P\right) \doteq \exp(-nK_P)$$

- Analysis tools include the method of types (large-deviations) and simple properties of trees
- Analyzed the very-noisy learning regime (Euclidean Information Theory) where learning is error-prone
- Extensions to learning the tree projection for non-trees have also been studied.

VYFT, A. Anandkumar, L. Tong, A. S. Willsky "A Large-Deviation Analysis of the Maximum-Likelihood Learning of Markov Tree Structures," ISIT 2009, submitted to IEEE Trans. on Information Theory, revised in Oct 2010.

Outline

Setup

- Jointly Gaussian distribution in very-noisy learning regime

$$p(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T\mathbf{\Sigma}^{-1}\mathbf{x}\right), \qquad \mathbf{x} \in \mathbb{R}^d.$$

- Zero-mean, unit variances

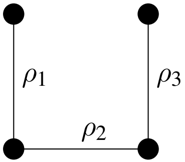
Setup

- Jointly Gaussian distribution in very-noisy learning regime

$$p(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T\boldsymbol{\Sigma}^{-1}\mathbf{x}\right), \qquad \mathbf{x} \in \mathbb{R}^d.$$

- Zero-mean, unit variances
- Keep correlations coefficients on edges fixed – specifies the Gaussian graphical model by Markovianity

ρ_i is the correlation coefficient on edge e_i for $i = 1, \ldots, d-1$



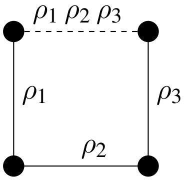
Setup

- Jointly Gaussian distribution in very-noisy learning regime

$$p(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T\mathbf{\Sigma}^{-1}\mathbf{x}\right), \qquad \mathbf{x} \in \mathbb{R}^d.$$

- Zero-mean, unit variances
- Keep correlations coefficients on edges fixed – specifies the Gaussian graphical model by Markovianity

ρ_i is the correlation coefficient on edge e_i for $i = 1, \dots, d-1$



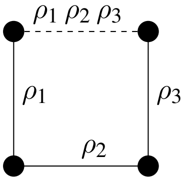
Setup

- Jointly Gaussian distribution in very-noisy learning regime

$$p(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T \mathbf{\Sigma}^{-1}\mathbf{x}\right), \qquad \mathbf{x} \in \mathbb{R}^d.$$

- Zero-mean, unit variances
- Keep correlations coefficients on edges fixed – specifies the Gaussian graphical model by Markovianity

ρ_i is the correlation coefficient on edge e_i for $i = 1, \ldots, d-1$



- Compare the error exponent associated to different structures

The Gaussian Case: Extremal Tree Structures

Theorem (Extremal Structures)

Under the very-noisy assumption,

- *Star graphs are hardest to learn (smallest approx error exponent)*

The Gaussian Case: Extremal Tree Structures

Theorem (Extremal Structures)

Under the very-noisy assumption,

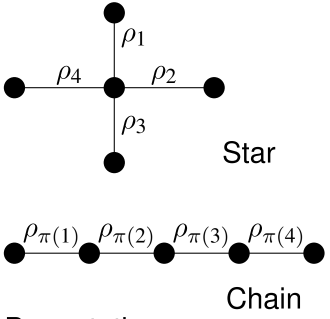
- *Star graphs are hardest to learn (smallest approx error exponent)*
- *Markov chains are easiest to learn (largest approx error exponent)*

The Gaussian Case: Extremal Tree Structures

Theorem (Extremal Structures)

Under the very-noisy assumption,

- *Star graphs are hardest to learn (smallest approx error exponent)*
- *Markov chains are easiest to learn (largest approx error exponent)*



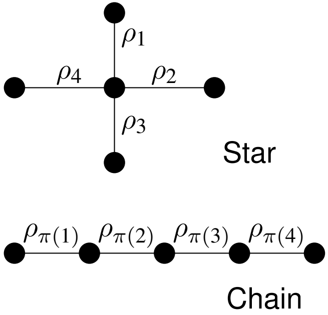
π: Permutation

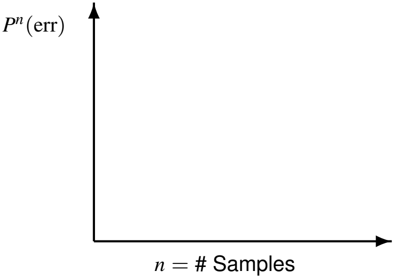
The Gaussian Case: Extremal Tree Structures

Theorem (Extremal Structures)

Under the very-noisy assumption,

- *Star graphs are hardest to learn (smallest approx error exponent)*
- *Markov chains are easiest to learn (largest approx error exponent)*





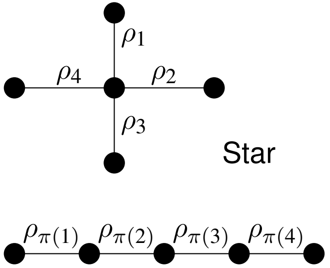
π: Permutation

The Gaussian Case: Extremal Tree Structures

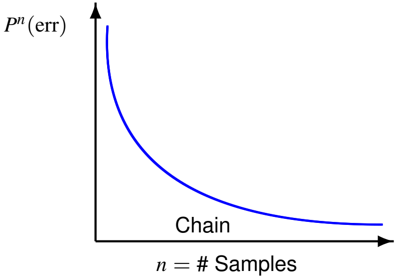
Theorem (Extremal Structures)

Under the very-noisy assumption,

- *Star graphs are hardest to learn (smallest approx error exponent)*
- *Markov chains are easiest to learn (largest approx error exponent)*



π: Permutation

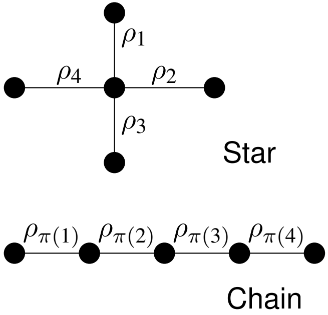


The Gaussian Case: Extremal Tree Structures

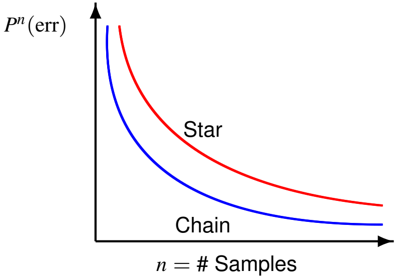
Theorem (Extremal Structures)

Under the very-noisy assumption,

- *Star graphs are hardest to learn (smallest approx error exponent)*
- *Markov chains are easiest to learn (largest approx error exponent)*



π: Permutation

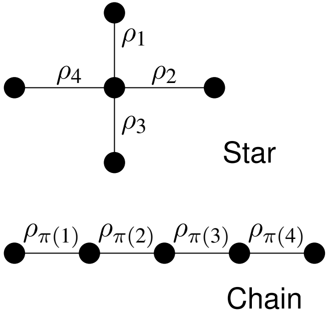


The Gaussian Case: Extremal Tree Structures

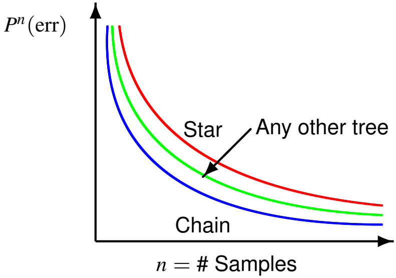
Theorem (Extremal Structures)

Under the very-noisy assumption,

- *Star graphs are hardest to learn (smallest approx error exponent)*
- *Markov chains are easiest to learn (largest approx error exponent)*

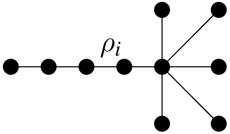


π: Permutation



Numerical Simulations

Chain, Star and Hybrid for $d = 10$

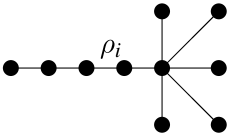


$\rho_i = 0.1 \times i \qquad i \in [1:9]$

Numerical Simulations

Chain, Star and Hybrid for $d = 10$

$\rho_i = 0.1 \times i \qquad i \in [1:9]$



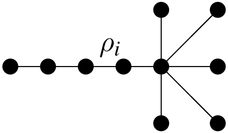
$\mathbb{P}(\text{error})$

$-\frac{1}{n} \log \mathbb{P}(\text{error})$

Numerical Simulations

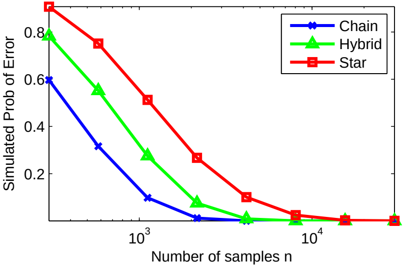
Chain, Star and Hybrid for $d = 10$

$$\rho_i = 0.1 \times i \qquad i \in [1:9]$$



$\mathbb{P}(\text{error})$

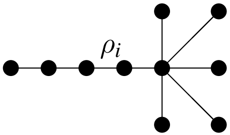
$-\frac{1}{n} \log \mathbb{P}(\text{error})$

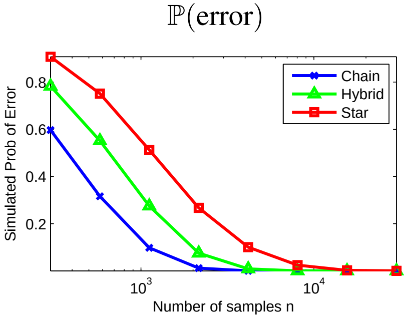


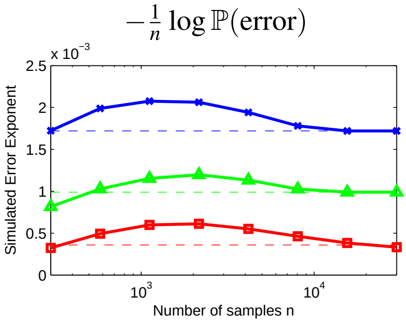
Numerical Simulations

Chain, Star and Hybrid for $d = 10$

$$\rho_i = 0.1 \times i \qquad i \in [1:9]$$

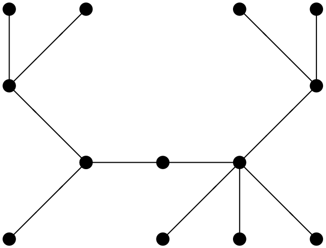






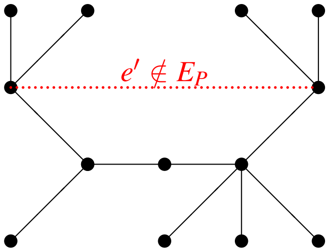
Proof Idea and Intuition

- Correlation decay



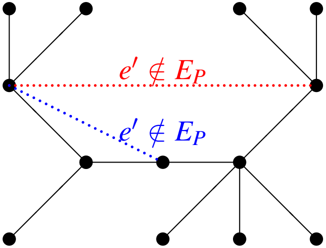
Proof Idea and Intuition

- Correlation decay



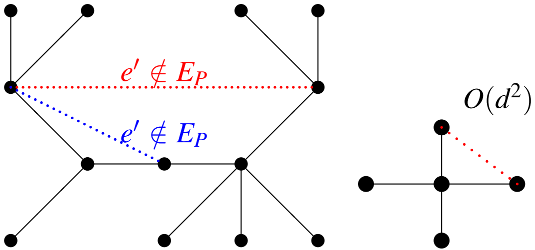
Proof Idea and Intuition

- Correlation decay



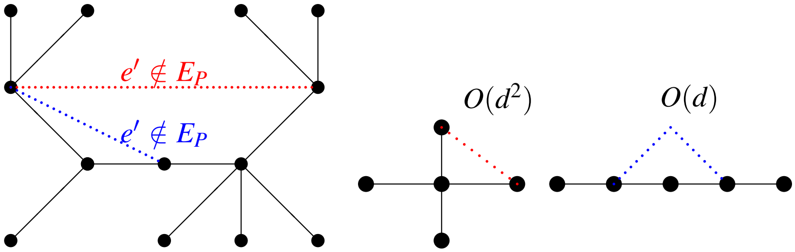
Proof Idea and Intuition

- Correlation decay



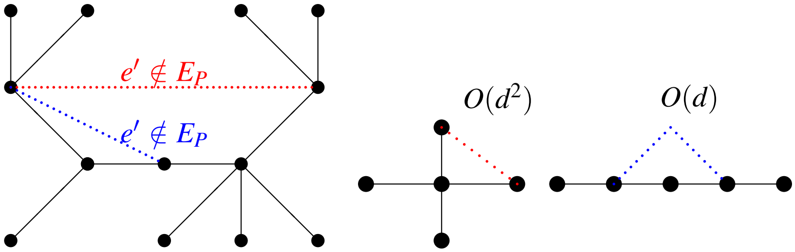
Proof Idea and Intuition

- Correlation decay



Proof Idea and Intuition

- Correlation decay



- Number of distance-two node pairs in:
  - Star is $O(d^2)$
  - Markov chain is $O(d)$

Concluding Remarks for Learning Gaussian Trees

- Gaussianity allows us to perform further analysis to find the extremal structures for learning

---

VYFT, A. Anandkumar, A. S. Willsky "Learning Gaussian Tree Models: Analysis of Error Exponents and Extremal Structures", Allerton 2009, IEEE Trans. on Signal Processing, May 2010.

Concluding Remarks for Learning Gaussian Trees

- Gaussianity allows us to perform further analysis to find the extremal structures for learning
- Allows to derive a data-processing inequality for crossover rates

VYFT, A. Anandkumar, A. S. Willsky "Learning Gaussian Tree Models: Analysis of Error Exponents and Extremal Structures", Allerton 2009, IEEE Trans. on Signal Processing, May 2010.

Concluding Remarks for Learning Gaussian Trees

- Gaussianity allows us to perform further analysis to find the extremal structures for learning
- Allows to derive a data-processing inequality for crossover rates
- Universal result – not (strongly) dependent on choice of correlations

$$\boldsymbol{\rho} = \{\rho_1, \ldots, \rho_{d-1}\}$$

VYFT, A. Anandkumar, A. S. Willsky "Learning Gaussian Tree Models: Analysis of Error Exponents and Extremal Structures", Allerton 2009, IEEE Trans. on Signal Processing, May 2010.

Outline

Motivation: Prevent Overfitting

- Chow-Liu algorithm tells us how to learn trees
- Suppose we are in the high-dimensional setting where

$$\text{Samples } n \ll \text{Variables } d$$

learning forest-structured graphical models may reduce overfitting vis-à-vis trees [Liu, Lafferty and Wasserman, 2010]

Motivation: Prevent Overfitting

- Chow-Liu algorithm tells us how to learn trees
- Suppose we are in the high-dimensional setting where

$$\text{Samples } n \ll \text{Variables } d$$

  learning forest-structured graphical models may reduce overfitting vis-à-vis trees [Liu, Lafferty and Wasserman, 2010]
- Extend Liu et al.'s work for discrete models and improve convergence results

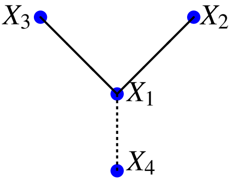
Motivation: Prevent Overfitting

- Chow-Liu algorithm tells us how to learn trees
- Suppose we are in the high-dimensional setting where

$$\text{Samples } n \ll \text{Variables } d$$

  learning forest-structured graphical models may reduce overfitting vis-à-vis trees [Liu, Lafferty and Wasserman, 2010]

- Extend Liu et al.'s work for discrete models and improve convergence results
- Strategy: Remove "weak" edges



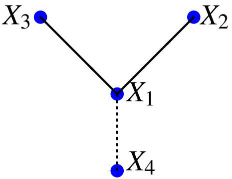
Motivation: Prevent Overfitting

- Chow-Liu algorithm tells us how to learn trees
- Suppose we are in the high-dimensional setting where

$$\text{Samples } n \ll \text{Variables } d$$

  learning forest-structured graphical models may reduce overfitting vis-à-vis trees [Liu, Lafferty and Wasserman, 2010]

- Extend Liu et al.'s work for discrete models and improve convergence results
- Strategy: Remove "weak" edges



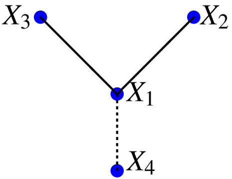
$\Rightarrow$ Reduce Num Params $\Rightarrow$

Motivation: Prevent Overfitting

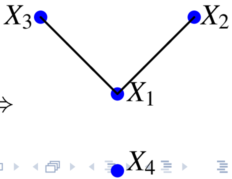
- Chow-Liu algorithm tells us how to learn trees
- Suppose we are in the high-dimensional setting where

$$\text{Samples } n \ll \text{Variables } d$$

  learning forest-structured graphical models may reduce overfitting vis-à-vis trees [Liu, Lafferty and Wasserman, 2010]
- Extend Liu et al.'s work for discrete models and improve convergence results
- Strategy: Remove "weak" edges



$\Rightarrow$ Reduce Num Params $\Rightarrow$



Main Contributions

- Propose CLThres, a thresholding algorithm, for consistently learning forest-structured models

Main Contributions

- Propose CLThres, a thresholding algorithm, for consistently learning forest-structured models
- Prove convergence rates ("moderate deviations") for a fixed discrete graphical model $P \in \mathcal{P}(\mathcal{X}^d)$

Main Contributions

- Propose CLThres, a thresholding algorithm, for consistently learning forest-structured models
- Prove convergence rates ("moderate deviations") for a fixed discrete graphical model $P \in \mathcal{P}(\mathcal{X}^d)$
- Prove achievable scaling laws on (n, d, k) (k is the num edges) for consistent recovery in high-dimensions. Roughly speaking,

$$n \gtrsim \log^{1+\delta}(d-k)$$

is achievable

Main Difficulty

- Unknown minimum mutual information $I_{\min}$ in the forest model

Main Difficulty

- Unknown minimum mutual information $I_{\min}$ in the forest model
- Markov order estimation [Merhav, Gutman, Ziv 1989]

Main Difficulty

- Unknown minimum mutual information $I_{\min}$ in the forest model
- Markov order estimation [Merhav, Gutman, Ziv 1989]
- If known, can easily use a threshold, i.e,

$$\text{if} \qquad I(\widehat{P}_{i,j}) < I_{\min}, \qquad \text{remove} \quad (i,j)$$

Main Difficulty

- Unknown minimum mutual information $I_{\min}$ in the forest model
- Markov order estimation [Merhav, Gutman, Ziv 1989]
- If known, can easily use a threshold, i.e,

$$\text{if} \qquad I(\widehat{P}_{i,j}) < I_{\min}, \qquad \text{remove} \quad (i,j)$$

- How to deal with classic tradeoff between over- and underestimation errors?

The CLThres Algorithm

- Compute the set of empirical mutual information $I(\widehat{P}_{i,j})$ for all $(i,j) \in V \times V$

The CLThres Algorithm

- Compute the set of empirical mutual information $I(\widehat{P}_{i,j})$ for all $(i,j) \in V \times V$
- Max-weight spanning tree

$$\widehat{E}_{d-1} := \underset{E:\text{Tree}}{\operatorname{argmax}} \sum_{(i,j)\in E} I(\widehat{P}_{i,j})$$

The CLThres Algorithm

- Compute the set of empirical mutual information $I(\widehat{P}_{i,j})$ for all $(i,j) \in V \times V$
- Max-weight spanning tree

$$\widehat{E}_{d-1} := \operatorname*{argmax}_{E:\text{Tree}} \sum_{(i,j)\in E} I(\widehat{P}_{i,j})$$

- Estimate number of edges given threshold ϵ_n

$$\widehat{k}_n := \left| \left\{ (i,j) \in \widehat{E}_{d-1} : I(\widehat{P}_{i,j}) \geq \epsilon_n \right\} \right|$$

The CLThres Algorithm

- Compute the set of empirical mutual information $I(\widehat{P}_{i,j})$ for all $(i,j) \in V \times V$
- Max-weight spanning tree

$$\widehat{E}_{d-1} := \underset{E:\text{Tree}}{\operatorname{argmax}} \sum_{(i,j)\in E} I(\widehat{P}_{i,j})$$

- Estimate number of edges given threshold ϵ_n

$$\widehat{k}_n := \left| \left\{ (i,j) \in \widehat{E}_{d-1} : I(\widehat{P}_{i,j}) \geq \epsilon_n \right\} \right|$$

- Output the forest with the top $\widehat{k}_n$ edges
- Computational Complexity = $O((n + \log d)d^2)$

A Convergence Result for CLThres

Assume that $P \in \mathcal{P}(\mathcal{X}^d)$ is a fixed forest-structured graphical model

d does not grow with n

A Convergence Result for CLThres

Assume that $P \in \mathcal{P}(\mathcal{X}^d)$ is a fixed forest-structured graphical model

d does not grow with n

Theorem ("Moderate Deviations")

Assume that the sequence $\{\epsilon_n\}_{n=1}^{\infty}$ satisfies

$$\lim_{n\to\infty} \epsilon_n = 0, \qquad \lim_{n\to\infty} \frac{n\epsilon_n}{\log n} = \infty, \qquad (\epsilon_n := n^{-1/2} \text{ works})$$

A Convergence Result for CLThres

Assume that $P \in \mathcal{P}(\mathcal{X}^d)$ is a fixed forest-structured graphical model

d does not grow with n

Theorem ("Moderate Deviations")

Assume that the sequence $\{\epsilon_n\}_{n=1}^{\infty}$ *satisfies*

$$\lim_{n\to\infty} \epsilon_n = 0, \qquad \lim_{n\to\infty} \frac{n\epsilon_n}{\log n} = \infty, \qquad (\epsilon_n := n^{-1/2} \text{ works})$$

Then

$$\limsup_{n\to\infty} \frac{1}{n\epsilon_n} \log \mathbb{P}(\widehat{E}_{\widehat{k}_n} \neq E_P) \leq -1, \qquad \Rightarrow$$

A Convergence Result for CLThres

Assume that $P \in \mathcal{P}(\mathcal{X}^d)$ is a fixed forest-structured graphical model

d does not grow with n

Theorem ("Moderate Deviations")

Assume that the sequence $\{\epsilon_n\}_{n=1}^{\infty}$ *satisfies*

$$\lim_{n\to\infty} \epsilon_n = 0, \qquad \lim_{n\to\infty} \frac{n\epsilon_n}{\log n} = \infty, \qquad (\epsilon_n := n^{-1/2} \textit{ works})$$

Then

$$\limsup_{n\to\infty} \frac{1}{n\epsilon_n} \log \mathbb{P}(\widehat{E}_{\widehat{k}_n} \neq E_P) \leq -1, \qquad \Rightarrow \qquad \mathbb{P}(\widehat{E}_{\widehat{k}_n} \neq E_P) \approx \exp(-n\epsilon_n)$$

A Convergence Result for CLThres

Assume that $P \in \mathcal{P}(\mathcal{X}^d)$ is a fixed forest-structured graphical model

d does not grow with n

Theorem ("Moderate Deviations")

Assume that the sequence $\{\epsilon_n\}_{n=1}^{\infty}$ *satisfies*

$$\lim_{n\to\infty} \epsilon_n = 0, \qquad \lim_{n\to\infty} \frac{n\epsilon_n}{\log n} = \infty, \qquad (\epsilon_n := n^{-1/2} \text{ works})$$

Then

$$\limsup_{n\to\infty} \frac{1}{n\epsilon_n} \log \mathbb{P}(\widehat{E}_{\widehat{k}_n} \neq E_P) \le -1, \qquad \Rightarrow \qquad \mathbb{P}(\widehat{E}_{\widehat{k}_n} \neq E_P) \approx \exp(-n\epsilon_n)$$

Also have a "liminf" lower bound

Remarks: A Convergence Result for CLThres

- The Chow-Liu phase is consistent with exponential rate of convergence

Remarks: A Convergence Result for CLThres

- The Chow-Liu phase is consistent with exponential rate of convergence
- The sequence can be taken to be $\epsilon_n := n^{-\beta}$ for $\beta \in (0, 1)$

Remarks: A Convergence Result for CLThres

- The Chow-Liu phase is consistent with exponential rate of convergence
- The sequence can be taken to be $\epsilon_n := n^{-\beta}$ for $\beta \in (0, 1)$
- For all n sufficiently large,

$$\epsilon_n < I_{\min}$$

implies no underestimation asymptotically

Remarks: A Convergence Result for CLThres

- The Chow-Liu phase is consistent with exponential rate of convergence
- The sequence can be taken to be $\epsilon_n := n^{-\beta}$ for $\beta \in (0,1)$
- For all n sufficiently large,

$$\epsilon_n < I_{\min}$$

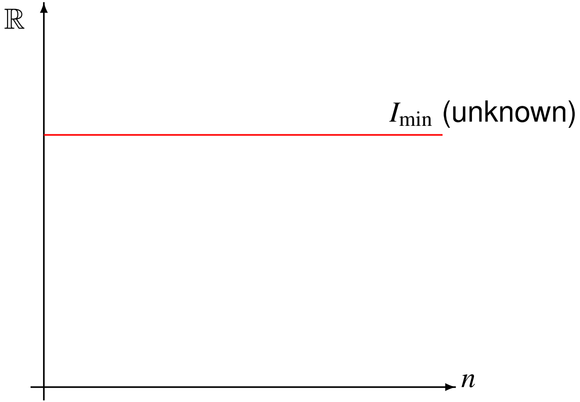
implies no underestimation asymptotically

- Note that for two independent random variables X_i and X_j with product pmf $Q_i \times Q_j$,

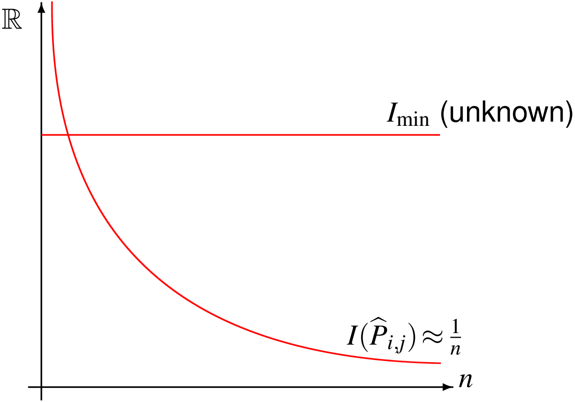
$$\mathrm{std}(I(\widehat{P}_{i,j})) = \Theta(1/n)$$

Since the sequence $\epsilon_n = \omega(\log n / n)$ decays slower than $\mathrm{std}(I(\widehat{P}_{i,j}))$, no overestimation asymptotically

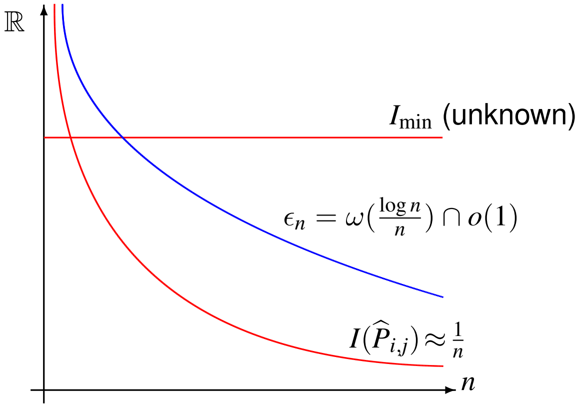
Pruning Away Weak Empirical Mutual Informations



Pruning Away Weak Empirical Mutual Informations



Pruning Away Weak Empirical Mutual Informations



Asymptotically, ϵ_n will be smaller than $I_{\min}$ and larger than $I(\widehat{P}_{i,j})$ with high probability

Proof Idea

Based fully on the method of types

Proof Idea

Based fully on the method of types

- Estimate Chow-Liu learning error

Proof Idea

Based fully on the method of types

- Estimate Chow-Liu learning error
- Estimate underestimation error

$$\mathbb{P}(\widehat{k}_n < k) \doteq \exp(-nL_P)$$

Proof Idea

Based fully on the method of types

- Estimate Chow-Liu learning error
- Estimate underestimation error

$$\mathbb{P}(\widehat{k}_n < k) \doteq \exp(-nL_P)$$

- Estimate overestimation error

  Decays subexponentially but faster than any polynomial:

$$\mathbb{P}(\widehat{k}_n > k) \approx \exp(-n\epsilon_n)$$

  Upper bound has no dependence on P (there exists a duality gap)

Proof Idea

Based fully on the method of types

- Estimate Chow-Liu learning error
- Estimate underestimation error

$$\mathbb{P}(\widehat{k}_n < k) \doteq \exp(-nL_P)$$

- Estimate overestimation error

  Decays subexponentially but faster than any polynomial:

$$\mathbb{P}(\widehat{k}_n > k) \approx \exp(-n\epsilon_n)$$

  Upper bound has no dependence on P (there exists a duality gap)

  Additional Technique: Euclidean Information Theory

High-Dimensional Learning

- Consider a sequence of structure learning problems indexed by number of samples n

High-Dimensional Learning

- Consider a sequence of structure learning problems indexed by number of samples n
- For each particular problem, we have data $\mathbf{x}^n = \{\mathbf{x}_i\}_{i=1}^n$

High-Dimensional Learning

- Consider a sequence of structure learning problems indexed by number of samples n
- For each particular problem, we have data $\mathbf{x}^n = \{\mathbf{x}_i\}_{i=1}^n$
- Each sample $\mathbf{x}_i \in \mathcal{X}^d$ is drawn independently from a forest-structured model with d nodes and k edges

High-Dimensional Learning

- Consider a sequence of structure learning problems indexed by number of samples n
- For each particular problem, we have data $\mathbf{x}^n = \{\mathbf{x}_i\}_{i=1}^n$
- Each sample $\mathbf{x}_i \in \mathcal{X}^d$ is drawn independently from a forest-structured model with d nodes and k edges
- Sequence of tuples $\{(n, d_n, k_n)\}_{n=1}^{\infty}$

High-Dimensional Learning

- Consider a sequence of structure learning problems indexed by number of samples n
- For each particular problem, we have data $\mathbf{x}^n = \{\mathbf{x}_i\}_{i=1}^n$
- Each sample $\mathbf{x}_i \in \mathcal{X}^d$ is drawn independently from a forest-structured model with d nodes and k edges
- Sequence of tuples $\{(n, d_n, k_n)\}_{n=1}^\infty$

Assumptions

$$\text{(A1)} \qquad I_{\inf} := \inf_{d \in \mathbb{N}} \min_{(i,j) \in E_P} I(P_{i,j}) > 0$$

$$\text{(A2)} \qquad \kappa := \inf_{d \in \mathbb{N}} \min_{(x_i,x_j) \in \mathcal{X}^2} P_{i,j}(x_i, x_j) > 0$$

An Achievable Scaling Law for CLThres

Theorem (Sufficient Conditions)

Assume (A1) and (A2). Fix $\delta > 0$*. There exists constants* $C_1, C_2 > 0$ *such that if*

$$n > \max\left\{C_1 \log d, C_2 \log k,\right.$$

An Achievable Scaling Law for CLThres

Theorem (Sufficient Conditions)

Assume (A1) and (A2). Fix $\delta > 0$. There exists constants $C_1, C_2 > 0$ such that if

$$n > \max\left\{C_1 \log d, C_2 \log k, (2\log(d-k))^{1+\delta}\right\}$$

An Achievable Scaling Law for CLThres

Theorem (Sufficient Conditions)

Assume (A1) and (A2). Fix $\delta > 0$*. There exists constants* $C_1, C_2 > 0$ *such that if*

$$n > \max\left\{C_1 \log d, C_2 \log k, (2\log(d-k))^{1+\delta}\right\}$$

the error probability of structure learning

$$\mathbb{P}(\textit{error}) \to 0$$

as $(n, d_n, k_n) \to \infty$

Remarks on the Achievable Scaling Law for CLThres

- If the model parameters (n, d, k) grow with n but if

$$\begin{aligned} d &\quad \text{subexponential} \\ k &\quad \text{subexponential} \\ d-k &\quad \text{subexponential} \end{aligned}$$

structure recovery is asymptotically possible

Remarks on the Achievable Scaling Law for CLThres

- If the model parameters (n, d, k) grow with n but if

$$\begin{aligned} d &\quad \text{subexponential} \\ k &\quad \text{subexponential} \\ d - k &\quad \text{subexponential} \end{aligned}$$

  structure recovery is asymptotically possible

- d can grow much faster than n

Remarks on the Achievable Scaling Law for CLThres

- If the model parameters (n, d, k) grow with n but if

$$\begin{aligned} d &\quad \text{subexponential} \\ k &\quad \text{subexponential} \\ d-k &\quad \text{subexponential} \end{aligned}$$

  structure recovery is asymptotically possible

- d can grow much faster than n

- Proof uses:
  1. Previous fixed d result
  2. Exponents in the limsup upper bound do not vanish with increasing problem size as $(n, d_n, k_n) \to \infty$

A Simple Strong Converse Result

Proposition (A Necessary Condition)

Assume forests with d nodes are chosen uniformly at random. Fix $\eta > 0$. Then if

$$n < \frac{(1-\eta)\log d}{\log|\mathcal{X}|}$$

the error probability of structure learning

$$\mathbb{P}(\textit{error}) \to 1$$

as $(n, d_n) \to \infty$ (independent of k_n)

A Simple Strong Converse Result

Proposition (A Necessary Condition)

Assume forests with d nodes are chosen uniformly at random. Fix $\eta > 0$. Then if

$$n < \frac{(1-\eta)\log d}{\log|\mathcal{X}|}$$

the error probability of structure learning

$$\mathbb{P}(\mathit{error}) \to 1$$

as $(n, d_n) \to \infty$ (independent of k_n)

- $\Omega(\log d)$ is necessary for successful recovery
- This lower bound is independent of parameters
- The dependence on num of edges k_n can be made more explicit
- Close to the sufficient condition

Concluding Remarks for Learning Forests

- Proposed a simple extension of Chow-Liu's MWST algorithm to learn forests consistently

VYFT, A. Anandkumar and A. S. Willsky "Learning High-Dimensional Markov Forest Distributions: Analysis of Error Rates", Allerton 10, Submitted to JMLR.

Concluding Remarks for Learning Forests

- Proposed a simple extension of Chow-Liu's MWST algorithm to learn forests consistently
- Error rates in the form of a "moderate deviations" result

VYFT, A. Anandkumar and A. S. Willsky "Learning High-Dimensional Markov Forest Distributions: Analysis of Error Rates", Allerton 10, Submitted to JMLR.

Concluding Remarks for Learning Forests

- Proposed a simple extension of Chow-Liu's MWST algorithm to learn forests consistently
- Error rates in the form of a "moderate deviations" result
- Scaling laws on (n, d, k) for structural consistency in high dimensions

VYFT, A. Anandkumar and A. S. Willsky "Learning High-Dimensional Markov Forest Distributions: Analysis of Error Rates", Allerton 10, Submitted to JMLR.

Concluding Remarks for Learning Forests

- Proposed a simple extension of Chow-Liu's MWST algorithm to learn forests consistently
- Error rates in the form of a "moderate deviations" result
- Scaling laws on (n, d, k) for structural consistency in high dimensions

Extensions:

VYFT, A. Anandkumar and A. S. Willsky "Learning High-Dimensional Markov Forest Distributions: Analysis of Error Rates", Allerton 10, Submitted to JMLR.

Concluding Remarks for Learning Forests

- Proposed a simple extension of Chow-Liu's MWST algorithm to learn forests consistently
- Error rates in the form of a "moderate deviations" result
- Scaling laws on (n, d, k) for structural consistency in high dimensions

Extensions:

- Risk consistency has also been analyzed (See thesis for details)

$$R(P^*) = O_p\left(\frac{d \log d}{n^{1-\gamma}}\right)$$

VYFT, A. Anandkumar and A. S. Willsky "Learning High-Dimensional Markov Forest Distributions: Analysis of Error Rates", Allerton 10, Submitted to JMLR.

Concluding Remarks for Learning Forests

- Proposed a simple extension of Chow-Liu's MWST algorithm to learn forests consistently
- Error rates in the form of a "moderate deviations" result
- Scaling laws on (n, d, k) for structural consistency in high dimensions

Extensions:

- Risk consistency has also been analyzed (See thesis for details)

$$R(P^*) = O_p\left(\frac{d \log d}{n^{1-\gamma}}\right)$$

- Need to find the right balance between over- and underestimation for the finite sample case

VYFT, A. Anandkumar and A. S. Willsky "Learning High-Dimensional Markov Forest Distributions: Analysis of Error Rates", Allerton 10, Submitted to JMLR.

Outline

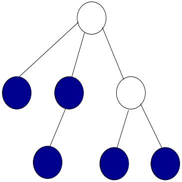
Structure Learning in Graphical Models Beyond Trees

Techniques extend to learning other classes of graphical models

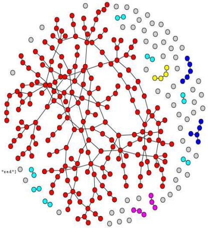
Structure Learning in Graphical Models Beyond Trees

Techniques extend to learning other classes of graphical models

Latent Trees



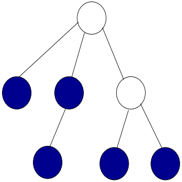
Random Graphs



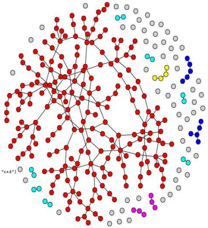
Beyond Trees

Structure Learning in Graphical Models Beyond Trees

Latent Trees



Random Graphs



Techniques extend to learning other classes of graphical models

- Learn latent trees, where only a subset of nodes are observed
- If the original graph is drawn from the Erdős-Rényi ensemble $\mathcal{G}(n, \frac{c}{n})$, we can also provide guarantees for structure learning
- Utilize the fact that the model is locally tree-like

Conclusions

- Graphical models provide a powerful and parsimonious representation of high-dimensional data
- (Ch. 3) Provided large-deviation analysis of ML learning of tree-structured distributions
  - (Ch. 4) Identified extremal structures for tree-structured Gaussian graphical models

Conclusions

- Graphical models provide a powerful and parsimonious representation of high-dimensional data
- (Ch. 3) Provided large-deviation analysis of ML learning of tree-structured distributions
  - (Ch. 4) Identified extremal structures for tree-structured Gaussian graphical models
- (Ch. 5) Extended analysis to forest-structured graphical models
  - Derived scaling laws on num variables, num edges and num samples for consistent learning in high-dimensions

Conclusions

- Graphical models provide a powerful and parsimonious representation of high-dimensional data
- (Ch. 3) Provided large-deviation analysis of ML learning of tree-structured distributions
  - (Ch. 4) Identified extremal structures for tree-structured Gaussian graphical models
- (Ch. 5) Extended analysis to forest-structured graphical models
  - Derived scaling laws on num variables, num edges and num samples for consistent learning in high-dimensions
- (Ch. 6) Also proposed algorithms for learning tree models for hypothesis testing