Learning Restricted Boltzmann Machines

Ankur Moitra (MIT)

joint work with Guy Bresler and Frederic Koehler
GRAPHICAL MODELS

Rich model for defining high-dimensional distributions in terms of their dependence structure
GRAPHICAL MODELS

Rich model for defining high-dimensional distributions in terms of their dependence structure

e.g. an \textbf{Ising model} is a distribution on \( \{\pm 1\}^n \) with

\[
\mathbb{P}[X = x] = \frac{1}{Z} \exp \left( \sum_{i,j} J_{i,j} x_i x_j + \sum_i h_i x_i \right)
\]

where \( Z \) is the partition function

interaction matrix  \hspace{1cm}  external field
GRAPHICAL MODELS

Rich model for defining high-dimensional distributions in terms of their dependence structure

e.g. an Ising model is a distribution on \( \{\pm 1\}^n \) with

\[
\mathbb{P}[X = x] = \frac{1}{Z} \exp \left( \sum_{i,j} J_{i,j} x_i x_j + \sum_i h_i x_i \right)
\]

where \( Z \) is the partition function

**Generalizations:** larger alphabet (Potts model), higher-order interactions (Markov Random Field), directed (Bayesian network)
CONDITIONAL INDEPENDENCE

Often helpful to look at their graph structure:

\[ G = (\{X_1, \cdots, X_n\}, E) \text{ with } E = \{(X_i, X_j) \text{ s.t. } J_{i,j} \neq 0\} \]
CONDITIONAL INDEPENDENCE

Often helpful to look at their graph structure:

\[ G = (\{X_1, \cdots, X_n\}, E) \text{ with } E = \{(X_i, X_j) \text{ s.t. } J_{i,j} \neq 0\} \]

**Key Property:** Two nodes are independent when conditioned on a separator
CONDITIONAL INDEPENDENCE

Often helpful to look at their graph structure:

\[ G = (\{X_1, \cdots, X_n\}, E) \text{ with } E = \{(X_i, X_j) \text{ s.t. } J_{i,j} \neq 0\} \]

**Key Property:** Two nodes are independent when conditioned on a separator – i.e.

\[ X_i \perp X_j | X_U \]

provided that all paths from \( X_i \) to \( X_j \) pass through \( X_U \)
CONDITIONAL INDEPENDENCE

Often helpful to look at their graph structure:

\[ G = (\{X_1, \cdots, X_n\}, E) \quad \text{with} \quad E = \{(X_i, X_j) \quad \text{s.t.} \quad J_{i,j} \neq 0\} \]

**Key Property:** Two nodes are independent when conditioned on a separator – i.e.

\[ X_i \perp X_j \mid X_U \]

provided that all paths from \( X_i \) to \( X_j \) pass through \( X_U \)

Can we learn graphical models from random samples?
HISTORY

Classes of graphical models that can be efficiently learned:

[Chow, Liu ‘68]: Polynomial time algorithm on trees
Classes of graphical models that can be efficiently learned:

[Chow, Liu ‘68]: Polynomial time algorithm on trees

[Karger, Srebro ‘01]: Polynomial time algorithm on graphs of bounded treewidth
HISTORY

Classes of graphical models that can be efficiently learned:

[Chow, Liu ‘68]: Polynomial time algorithm on trees

[Karger, Srebro ‘01]: Polynomial time algorithm on graphs of bounded treewidth

[Bresler ‘15]: Polynomial time algorithm on graphs of bounded degree (doubly-exponential dependence on max degree)
HISTORY

Classes of graphical models that can be efficiently learned:

[Chow, Liu ‘68]: Polynomial time algorithm on trees

[Karger, Srebro ‘01]: Polynomial time algorithm on graphs of bounded treewidth

[Bresler ‘15]: Polynomial time algorithm on graphs of bounded degree (doubly-exponential dependence on max degree)

Improved to singly-exponential in [Vuffray et al. ‘16] and [Klivans, Meka ‘17]
HISTORY

Classes of graphical models that can be efficiently learned:

[Chow, Liu ‘68]: Polynomial time algorithm on trees

[Karger, Srebro ‘01]: Polynomial time algorithm on graphs of bounded treewidth

[Bresler ‘15]: Polynomial time algorithm on graphs of bounded degree (doubly-exponential dependence on max degree)

Improved to singly-exponential in [Vuffray et al. ‘16] and [Klivans, Meka ‘17]

[Bresler et al. ‘08], [Ravikumar et al. ‘10]: Better algorithms when there are no long range correlations
OUTLINE

Part I: Introduction

• Learning Ising Models
• Latent Variables and Higher-Order Dependencies
• Our Results

Part II: Learning Ferromagnetic RBMs

• The Discrete Influence Function
• A Greedy Algorithm
• The Griffiths-Hurst-Sherman Inequality
OUTLINE

Part I: Introduction

• Learning Ising Models
• Latent Variables and Higher-Order Dependencies
• Our Results

Part II: Learning Ferromagnetic RBMs

• The Discrete Influence Function
• A Greedy Algorithm
• The Griffiths-Hurst-Sherman Inequality
Main question:

What if there are unobserved/latent variables?
Main question:

What if there are unobserved/latent variables?

Allows variables to influence each other through unobserved mechanisms
Main question:

What if there are unobserved/latent variables?

Allows variables to influence each other through unobserved mechanisms

Scientific theories that explain data in a more parsimonious way can be learned/tested
Popular model following Hinton: **Restricted Boltzmann Machines**

**observed variables:** $X_1, \cdots, X_n$

**latent variables:** $Y_1, \cdots, Y_m$
Popular model following Hinton: **Restricted Boltzmann Machines**

**observed variables:** $X_1, \cdots, X_n$

**latent variables:** $Y_1, \cdots, Y_m$

with joint distribution on $\{\pm 1\}^n \times \{\pm 1\}^m$ given by

$$
P[X = x, Y = y] = \frac{1}{Z} \exp \left( \sum_{i,j} J_{i,j} x_i y_j + h^{(1)}(x) + h^{(2)}(y) \right)$$

*external fields*
Popular model following Hinton: **Restricted Boltzmann Machines**

**observed variables:** $X_1, \cdots, X_n$

**latent variables:** $Y_1, \cdots, Y_m$

with joint distribution on $\{\pm 1\}^n \times \{\pm 1\}^m$ given by

$$
\mathbb{P}[X = x, Y = y] = \frac{1}{Z} \exp \left( \sum_{i,j} J_{i,j} x_i y_j + h^{(1)}(x) + h^{(2)}(y) \right)
$$

Used in **feature extraction**, **collaborative filtering** and are the building block of **deep belief networks**
Popular model following Hinton: **Restricted Boltzmann Machines**

- **observed variables**: $X_1, \cdots, X_n$
- **latent variables**: $Y_1, \cdots, Y_m$

with joint distribution on $\{\pm 1\}^n \times \{\pm 1\}^m$ given by

$$P[X = x, Y = y] = \frac{1}{Z} \exp \left( \sum_{i,j} J_{i,j} x_i y_j + h^{(1)}(x) + h^{(2)}(y) \right)$$

Used in **feature extraction, collaborative filtering** and are the building block of **deep belief networks**

Are there efficient algorithms for learning RBMs?
Main Challenge: When you marginalize out a node it creates a higher-order dependence among its neighbors
Main Challenge: When you marginalize out a node it creates a higher-order dependence among its neighbors.
Main Challenge: When you marginalize out a node it creates a higher-order dependence among its neighbors.

In particular, the joint distribution is usually not an Ising model!
Main Challenge: When you marginalize out a node it creates a higher-order dependence among its neighbors.

In particular, the joint distribution is usually not an Ising model!

So what type of distribution is it?
A Markov random field of order $r$ is a distribution on $\{\pm 1\}^n$ with a degree $r$ polynomial given by:

$$P[X = x] = \frac{1}{Z} \exp(f(x))$$
A Markov random field of order $r$ is a distribution on $\{\pm 1\}^n$ with
degree $r$ polynomial

$$\mathbb{P}[X = x] = \frac{1}{Z} \exp\left(f(x)\right)$$

**Folklore Fact:** The marginal distribution on $X$ in an RBM where latent nodes have degree at most $r$ is an order $r$ MRF
A Markov random field of order $r$ is a distribution on $\{\pm 1\}^n$ with binary variables.

$$\mathbb{P}[X = x] = \frac{1}{Z} \exp\left(f(x)\right)$$

degree $r$ polynomial

**Folklore Fact:** The marginal distribution on $X$ in an RBM where latent nodes have degree at most $r$ is an order $r$ MRF.

Can we learn RBMs by learning the joint distribution on observed nodes as an MRF?
A Markov random field of order $r$ is a distribution on $\{\pm 1\}^n$ with binary variables.

$$
\mathbb{P}[X = x] = \frac{1}{Z} \exp\left(f(x)\right)
$$

Folklore Fact: The marginal distribution on $X$ in an RBM where latent nodes have degree at most $r$ is an order $r$ MRF.

Can we learn RBMs by learning the joint distribution on observed nodes as an MRF?

Are there efficient algorithms for learning MRFs?
[Klivans, Meka ‘17], [Hamilton et al. ‘17]: There are $n^{O(r)}$ time algorithms for learning order $r$ MRFs on $n$ variables with bounded degree.
LEARNING MARKOV RANDOM FIELDS

[Klivans, Meka ‘17], [Hamilton et al. ‘17]: There are $n^{O(r)}$ time algorithms for learning order $r$ MRFs on $n$ variables with bounded degree.

Unfortunately:

[Bresler et al. ‘14], [Klivans, Meka ‘17]: Under standard hardness assumptions, learning an order $r$ MRF on $n$ variables takes $n^{\Omega(r)}$ time.
LEARNING MARKOV RANDOM FIELDS

[Klivans, Meka ‘17], [Hamilton et al. ‘17]: There are $n^{O(r)}$ time algorithms for learning order $r$ MRFs on $n$ variables with bounded degree.

Unfortunately:

[Bresler et al. ‘14], [Klivans, Meka ‘17]: Under standard hardness assumptions, learning an order $r$ MRF on $n$ variables takes $n^{\Omega(r)}$ time. Learning a $t$-sparse parity with noise on $n$ variables takes time $n^{\Omega(t)}$. 
LEARNING MARKOV RANDOM FIELDS

[Klivans, Meka ‘17], [Hamilton et al. ‘17]: There are $n^{O(r)}$ time algorithms for learning order $r$ MRFs on $n$ variables with bounded degree.

Unfortunately:

[Bresler et al. ‘14], [Klivans, Meka ‘17]: Under standard hardness assumptions, learning an order $r$ MRF on $n$ variables takes $n^{\Omega(r)}$ time.

Even worse, the reduction produces bounded degree MRFs.

learning a $t$-sparse parity with noise on $n$ variables takes time $n^{\Omega(t)}$. 
Main question (revised): Let $d$ be the maximum degree

Can we learn RBMs in faster than $n^d$ time?
Main question (revised): Let d be the maximum degree

Can we learn RBMs in faster than $n^d$ time?

These algorithms are close to trivial, because we can always brute-force search for the two-hop neighbors of a node in $n^{d^2}$ time.
OUTLINE

Part I: Introduction

• Learning Ising Models
• Latent Variables and Higher-Order Dependencies
• Our Results

Part II: Learning Ferromagnetic RBMs

• The Discrete Influence Function
• A Greedy Algorithm
• The Griffiths-Hurst-Sherman Inequality
OUTLINE

**Part I: Introduction**

- Learning Ising Models
- Latent Variables and Higher-Order Dependencies
- **Our Results**

**Part II: Learning Ferromagnetic RBMs**

- The Discrete Influence Function
- A Greedy Algorithm
- The Griffiths-Hurst-Sherman Inequality
OUR RESULTS: REPRESENTATIONAL POWER

Surprisingly, marginalizing out nodes can produce any higher-order interaction among their neighbors:
OUR RESULTS: REPRESENTATIONAL POWER

Surprisingly, marginalizing out nodes can produce any higher-order interaction among their neighbors:

**Theorem:** Every binary Markov random field of order $t$ can be realized as the distribution on observed nodes of an RBM where the maximum degree of any hidden node is at most $t$
OUR RESULTS: REPRESENTATIONAL POWER

Surprisingly, marginalizing out nodes can produce any higher-order interaction among their neighbors:

**Theorem:** Every binary Markov random field of order $t$ can be realized as the distribution on observed nodes of an RBM where the maximum degree of any hidden node is at most $t$

This precisely characterizes the representational power of bounded degree RBMs
OUR RESULTS: REPRESENTATIONAL POWER

Surprisingly, marginalizing out nodes can produce any higher-order interaction among their neighbors:

**Theorem:** Every binary Markov random field of order $t$ can be realized as the distribution on observed nodes of an RBM where the maximum degree of any hidden node is at most $t$

This precisely characterizes the representational power of bounded degree RBMs

Earlier work of [Martens et al. ‘13] showed that dense RBMs can represent parity (more generally, any predicate depending on # 1s)
OUR RESULTS: HARDNESS

As a result, we obtain hardness for improper learning:
OUR RESULTS: HARDNESS

As a result, we obtain hardness for improper learning:

**Corollary:** Under the sparse parity assumption, it is hard to learn any representation of the distribution on observed nodes within total variation distance $1/3$ in $n^{o(d)}$ time
OUR RESULTS: HARDNESS

As a result, we obtain hardness for improper learning:

**Corollary:** Under the sparse parity assumption, it is hard to learn any representation of the distribution on observed nodes within total variation distance 1/3 in $n^{o(d)}$ time.

Here we allow the algorithm to output any unnormalized function that can be efficiently computed.
OUR RESULTS: HARDNESS

As a result, we obtain hardness for improper learning:

**Corollary:** Under the sparse parity assumption, it is hard to learn any representation of the distribution on observed nodes within total variation distance $1/3$ in $n^{o(d)}$ time.

Here we allow the algorithm to output any unnormalized function that can be efficiently computed.

Our reduction produces an RBM with a constant number of latent nodes—e.g. for $d$-sparse parity $2^d$ hidden nodes of degree $d$. 
OUR RESULTS: HARDNESS

As a result, we obtain hardness for improper learning:

**Corollary:** Under the sparse parity assumption, it is hard to learn any representation of the distribution on observed nodes within total variation distance $1/3$ in $n^{o(d)}$ time.

Here we allow the algorithm to output any unnormalized function that can be efficiently computed.

Our reduction produces an RBM with a constant number of latent nodes — e.g. for $d$-sparse parity $2^d$ hidden nodes of degree $d$.

Earlier work of [Bogdanov et al. ‘08] required a large number of latent variables, one for each gate in a given circuit.
Are there any natural and well-motivated families of RBMs that can be efficiently learned?
Are there any natural and well-motivated families of RBMs that can be efficiently learned?

Yes, if they are ferromagnetic – i.e. $J, h \geq 0$
Are there any natural and well-motivated families of RBMs that can be efficiently learned?

Yes, if they are ferromagnetic – i.e. \( J, h \geq 0 \)

**Historical Note:** Ferromagneticity plays a key role in many classic results in statistical physics and TCS
Are there any natural and well-motivated families of RBMs that can be efficiently learned?

Yes, if they are ferromagnetic – i.e. $J, h \geq 0$

**Historical Note:** Ferromagneticivity plays a key role in many classic results in statistical physics and TCS

(1) [Lee, Yang ‘52] complex zeros of the partition function of a ferromagnetic Ising model lie on the imaginary axis
Are there any natural and well-motivated families of RBMs that can be efficiently learned?

Yes, if they are ferromagnetic – i.e. \( J, h \geq 0 \)

**Historical Note:** Ferromagneticity plays a key role in many classic results in statistical physics and TCS

1. [Lee, Yang ‘52] complex zeros of the partition function of a ferromagnetic Ising model lie on the imaginary axis

2. Seminal work of [Jerrum and Sinclair ‘90] gives an efficient algorithm for sampling from ferromagnetic Ising models
Are there any natural and well-motivated families of RBMs that can be efficiently learned?

Yes, if they are ferromagnetic – i.e. \( J, h \geq 0 \)

**Historical Note:** Ferromagneticity plays a key role in many classic results in statistical physics and TCS

(1) [Lee, Yang ‘52] complex zeros of the partition function of a ferromagnetic Ising model lie on the imaginary axis

(2) Seminal work of [Jerrum and Sinclair ‘90] gives an efficient algorithm for sampling from ferromagnetic Ising models

In our context, it prevents hidden nodes from cancelling out each other’s lower-order interactions
OUR RESULTS: ALGORITHMS

Our main result:

**Theorem:** There is a greedy algorithm with running time $f(d) \cdot n^2$ and sample complexity $f(d) \cdot \log n$ for learning ferromagnetic RBMs, with upper and lower bounds on the interaction strength.
Our main result:

**Theorem:** There is a greedy algorithm with running time $f(d) n^2$ and sample complexity $f(d) \log n$ for learning ferromagnetic RBMs, with upper and lower bounds on the interaction strength.

In particular, we output a description of the joint distribution on observed nodes as an MRF.
OUR RESULTS: ALGORITHMS

Our main result:

**Theorem:** There is a greedy algorithm with running time $f(d) \cdot n^2$ and sample complexity $f(d) \cdot \log n$ for learning ferromagnetic RBMs, with upper and lower bounds on the interaction strength.

In particular, we output a description of the joint distribution on observed nodes as an MRF.

Using results [Liu et al. ‘17] and the Lee-Yang Property, can also perform inference on the learned model.
OUR RESULTS: ALGORITHMS

Our main result:

**Theorem:** There is a greedy algorithm with running time $f(d) n^2$ and sample complexity $f(d) \log n$ for learning ferromagnetic RBMs, with upper and lower bounds on the interaction strength.

In particular, we output a description of the joint distribution on observed nodes as an MRF.

Using results [Liu et al. ‘17] and the Lee-Yang Property, can also perform inference on the learned model.

i.e. a PTAS for estimating the likelihood of any particular output.
OUR RESULTS: ALGORITHMS

Our main result:

**Theorem:** There is a greedy algorithm with running time $f(d) n^2$ and sample complexity $f(d) \log n$ for learning ferromagnetic RBMs, with upper and lower bounds on the interaction strength.

In particular, we output a description of the joint distribution on observed nodes as an MRF.

Using results [Liu et al. ‘17] and the Lee-Yang Property, can also perform inference on the learned model i.e. a PTAS for estimating the likelihood of any particular output.

Everything generalizes to ferromagnetic Ising models with latent variables, under conditions on diameter of latent nodes.
AN APPLICATION

Natural scenario where interactions are positive: Modeling the adoption of technology/spread of an epidemic on a network
AN APPLICATION

Natural scenario where interactions are positive: Modeling the adoption of technology/spread of an epidemic on a network

[Montanari, Saberi ‘10] model the process as Glauber dynamics on an Ising model (see also [Kempe et al. ‘03])
AN APPLICATION

Natural scenario where interactions are positive: **Modeling the adoption of technology/spread of an epidemic on a network**

![Diagram of network with nodes labeled iPhone and Android]

[Montanari, Saberi ‘10] model the process as Glauber dynamics on an Ising model (see also [Kempe et al. ‘03])

When you know the network/interactions, natural questions like: **What are the most influential nodes? How quickly does it spread?**
AN APPLICATION

Natural scenario where interactions are positive: **Modeling the adoption of technology/spread of an epidemic on a network**

But how can you learn the network from observations?
AN APPLICATION

Natural scenario where interactions are positive: **Modeling the adoption of technology/spread of an epidemic on a network**

But how can you learn the network from observations?

This is the problem of learning an Ising model
AN APPLICATION

Natural scenario where interactions are positive: Modeling the adoption of technology/spread of an epidemic on a network

But how can you learn the network from observations?

This is the problem of learning an Ising model

And what if you don’t observe all the nodes in the network?
AN APPLICATION

Natural scenario where interactions are positive: **Modeling the adoption of technology/spread of an epidemic on a network**

But how can you learn the network from observations?

This is the problem of learning an Ising model

**And what if you don’t observe all the nodes in the network?**

This is the problem of learning an Ising model with latent variables
AN APPLICATION

Natural scenario where interactions are positive: **Modeling the adoption of technology/spread of an epidemic on a network**

But how can you learn the network from observations?

This is the problem of learning an Ising model

**And what if you don’t observe all the nodes in the network?**

This is the problem of learning an Ising model with latent variables

Many natural extensions to consider: Potts models, arbitrary external field
OUTLINE

Part I: Introduction

• Learning Ising Models
• Latent Variables and Higher-Order Dependencies
• Our Results

Part II: Learning Ferromagnetic RBMs

• The Discrete Influence Function
• A Greedy Algorithm
• The Griffiths-Hurst-Sherman Inequality
OUTLINE

Part I: Introduction

• Learning Ising Models
• Latent Variables and Higher-Order Dependencies
• Our Results

Part II: Learning Ferromagnetic RBMs

• The Discrete Influence Function
• A Greedy Algorithm
• The Griffiths-Hurst-Sherman Inequality
MAIN STRUCTURAL RESULT

Key Definition: The discrete influence function at node $i$ is

$$I_i(S) \triangleq \mathbb{E}[X_i | X_S = \{+1\}^S]$$
MAIN STRUCTURAL RESULT

**Key Definition:** The discrete influence function at node $i$ is

$$I_i(S) \triangleq \mathbb{E}[X_i | X_S = \{+1\}^S]$$

i.e. it is a function from subsets $S \subseteq [n] \setminus \{i\}$ to the reals that measures the induced bias
MAIN STRUCTURAL RESULT

Key Definition: The discrete influence function at node $i$ is

$$I_i(S) \triangleq \mathbb{E}[X_i | X_S = \{+1\}^S]$$

i.e. it is a function from subsets $S \subseteq [n] \setminus \{i\}$ to the reals that measures the induced bias.

Now submodularity comes to the rescue:
Key Definition: The **discrete influence function** at node $i$ is

$$I_i(S) \triangleq \mathbb{E}[X_i | X_S = \{+1\}^S]$$

i.e. it is a function from subsets $S \subseteq [n] \setminus \{i\}$ to the reals that measures the induced bias.

Now **submodularity** comes to the rescue:

**Theorem:** Fix a ferromagnetic Ising model. Then for every $i$, the discrete influence function is **monotone** and **submodular**.
MAIN STRUCTURAL RESULT

Key Definition: The discrete influence function at node $i$ is

$$I_i(S) \triangleq \mathbb{E}[X_i | X_S = \{+1\}^S]$$

i.e. it is a function from subsets $S \subseteq [n] \setminus \{i\}$ to the reals that measures the induced bias.

Now submodularity comes to the rescue:

**Theorem:** Fix a ferromagnetic Ising model. Then for every $i$, the discrete influence function is **monotone** and **submodular**

It turns out that the concavity of magnetization is analogous to properties of the multilinear extension.
A HINT AT THE CONNECTION

Definition: The average magnetization is

\[ M = \frac{\mathbb{E}[X_1 + \cdots + X_n]}{n} \]
A HINT AT THE CONNECTION

Definition: The average magnetization is

\[ M = \frac{\mathbb{E}[X_1 + \cdots + X_n]}{n} \]

Suppose \( J \geq 0 \) the external field is \( H \) everywhere, then some intuitive/classic results are known
A HINT AT THE CONNECTION

**Definition:** The average magnetization is

\[ M = \frac{\mathbb{E}[X_1 + \cdots + X_n]}{n} \]

Suppose \( J \geq 0 \) the external field is \( H \) everywhere, then some intuitive/classic results are known

\[ (1) \quad \frac{\partial M}{\partial H} \geq 0 \]
A HINT AT THE CONNECTION

Definition: The average magnetization is

\[ M = \frac{\mathbb{E}[X_1 + \cdots + X_n]}{n} \]

Suppose J \geq 0 the external field is H everywhere, then some intuitive/classic results are known

(1) \( \frac{\partial M}{\partial H} \geq 0 \)  \hspace{1cm} (2) \( \frac{\partial^2 M}{\partial H^2} \leq 0 \) \hspace{1cm} \text{for all } H \geq 0
A HINT AT THE CONNECTION

Definition: The average magnetization is

\[ M = \frac{\mathbb{E}[X_1 + \cdots + X_n]}{n} \]

Suppose \( J \geq 0 \) the external field is \( H \) everywhere, then some intuitive/classic results are known

\[ (1) \quad \frac{\partial M}{\partial H} \geq 0 \quad (2) \quad \frac{\partial^2 M}{\partial H^2} \leq 0 \quad \text{for all } H \geq 0 \]

(2) is called concavity of magnetization, and follows from the famous Griffiths-Hurst-Sherman inequality and captures diminishing returns.
OUTLINE

Part I: Introduction

- Learning Ising Models
- Latent Variables and Higher-Order Dependencies
- Our Results

Part II: Learning Ferromagnetic RBMs

- The Discrete Influence Function
- A Greedy Algorithm
- The Griffiths-Hurst-Sherman Inequality
OUTLINE

Part I: Introduction

• Learning Ising Models
• Latent Variables and Higher-Order Dependencies
• Our Results

Part II: Learning Ferromagnetic RBMs

• The Discrete Influence Function
• A Greedy Algorithm
• The Griffiths-Hurst-Sherman Inequality
KEY IDEAS

Idea #1: Restricting to only the observed nodes, the discrete influence function is still monotone and submodular
KEY IDEAS

Idea #1: Restricting to only the observed nodes, the discrete influence function is still monotone and submodular

Idea #2: The maximizer ought to be the two hop neighbors of node i (or any set containing them)
KEY IDEAS

Idea #1: Restricting to only the observed nodes, the discrete influence function is still monotone and submodular

Idea #2: The maximizer ought to be the two hop neighbors of node i (or any set containing them)
KEY IDEAS

Idea #1: Restricting to only the observed nodes, the discrete influence function is still monotone and submodular.

Idea #2: The maximizer ought to be the two hop neighbors of node \( i \) (or any set containing them).

Because the two-hop neighbors separate \( i \) from all the other observed nodes.
QUANTITATIVE BOUNDS

We say that an Ising model is \((\alpha, \beta)\)-nondegenerate if

\[
(1) \quad J_{i,j} \neq 0 \Rightarrow |J_{i,j}| \geq \alpha
\]

\[
(2) \quad \sum_j |J_{i,j}| + |h_i| \leq \beta \quad \text{for all } i
\]
QUANTITATIVE BOUNDS

We say that an Ising model is \((\alpha, \beta)\)-nondegenerate if

\[
(1) \quad J_{i,j} \neq 0 \Rightarrow |J_{i,j}| \geq \alpha
\]

\[
(2) \quad \sum_j |J_{i,j}| + |h_i| \leq \beta \quad \text{for all } i
\]

We need these conditions to ensure the graph structure is identifiable.
QUANTITATIVE BOUNDS

We say that an Ising model is \((\alpha, \beta)\)-nondegenerate if

\begin{align*}
(1) \quad J_{i,j} \neq 0 \Rightarrow |J_{i,j}| & \geq \alpha \\
(2) \quad \sum_j |J_{i,j}| + |h_i| & \leq \beta \quad \text{for all } i
\end{align*}

We need these conditions to ensure the graph structure is identifiable

**Key Lemma:** If \(S\) does not contain the two-hop neighbors of \(i\), then there is a node \(j\) such that

\[I_i(S \cup \{j\}) - I_i(S) \geq \left(\frac{2\alpha^2}{1 + e^{2\beta}}\right)(1 - \tanh(\beta))^2\]
KEY IDEAS, CONTINUED

Classic result in approximation algorithms:

**Theorem [Nemhauser et al. ‘78]:** The greedy algorithm achieves a \( 1 - \frac{1}{e} \) factor approximation for maximizing a monotone submodular function subject to a cardinality constraint.
KEY IDEAS, CONTINUED

Now, how can we maximize the discrete influence function?

Theorem [Nemhauser et al. ‘78]: The greedy algorithm achieves a $1 - 1/e$ factor approximation for maximizing a monotone submodular function subject to a cardinality constraint.

Their analysis shows how fast gap to optimum value decreases, also gives a bicriteria approximation algorithm.
Now, how can we maximize the discrete influence function?

**Theorem [Nemhauser et al. ‘78]:** The greedy algorithm achieves a $1 - 1/e$ factor approximation for maximizing a monotone submodular function subject to a cardinality constraint.

Their analysis shows how fast gap to optimum value decreases, also gives a **bicriteria approximation algorithm**

i.e. as we allow the algorithm to output larger size sets, the approximation factor converges to 1
KEY IDEAS, CONTINUED

Now, how can we maximize the discrete influence function?

Theorem [Nemhauser et al. ‘78]: The greedy algorithm achieves a $1 - 1/e$ factor approximation for maximizing a monotone submodular function subject to a cardinality constraint.

Their analysis shows how fast gap to optimum value decreases, also gives a bicriteria approximation algorithm.

i.e. as we allow the algorithm to output larger size sets, the approximation factor converges to 1.

Idea #3: Run the greedy algorithm to learn a small superset of the two-hop neighbors.
KEY IDEAS, CONTINUED

Finally when we have a small superset of the two-hop neighbors, we can learn the induced MRF
KEY IDEAS, CONTINUED

Finally when we have a small superset of the two-hop neighbors, we can learn the induced MRF

The key is, each node no longer participates in $n^d$ possible order $d$ interactions, but rather at most $f(d)$
OUTLINE

Part I: Introduction

- Learning Ising Models
- Latent Variables and Higher-Order Dependencies
- Our Results

Part II: Learning Ferromagnetic RBMs

- The Discrete Influence Function
- A Greedy Algorithm
- The Griffiths-Hurst-Sherman Inequality
OUTLINE

Part I: Introduction

• Learning Ising Models
• Latent Variables and Higher-Order Dependencies
• Our Results

Part II: Learning Ferromagnetic RBMs

• The Discrete Influence Function
• A Greedy Algorithm
• The Griffiths-Hurst-Sherman Inequality
THE SMOOTH INFLUENCE FUNCTION

Definition: The smooth influence function at node $i$ is

$$I_i(h) \triangleq \mathbb{E}[X_i]$$

where the expectation is taken when we set the external field to $h$. 
Definition: The smooth influence function at node $i$ is

$$\mathcal{I}_i(h) \triangleq \mathbb{E}[X_i]$$

where the expectation is taken when we set the external field to $h$

In particular $I_i(S) = \mathcal{I}_i(h')$ where $h'$ comes from setting the coordinates in $S$ to $+\infty$ in $h$
**THE SMOOTH INFLUENCE FUNCTION**

**Definition:** The smooth influence function at node $i$ is

$$\mathcal{I}_i(h) \triangleq \mathbb{E}[X_i]$$

where the expectation is taken when we set the external field to $h$

In particular $I_i(S) = \mathcal{I}_i(h')$ where $h'$ comes from setting the coordinates in $S$ to $+\infty$ in $h$

In retrospect, it is the **multilinear extension** of $I_i$
THE GHS INEQUALITY

The Griffith-Hurst-Sherman inequality states

\[
\mathbb{E}[X_i X_j X_k X_\ell] - \mathbb{E}[X_i X_j] \mathbb{E}[X_k X_\ell] \\
- \mathbb{E}[X_i X_k] \mathbb{E}[X_j X_\ell] - \mathbb{E}[X_i X_\ell] \mathbb{E}[X_j X_k] \\
+ 2 \cdot \mathbb{E}[X_i X_\ell] \mathbb{E}[X_j X_\ell] \mathbb{E}[X_k X_\ell] \leq 0
\]
THE GHS INEQUALITY

The Griffith-Hurst-Sherman inequality states

\[ \mathbb{E}[X_iX_jX_kX_\ell] - \mathbb{E}[X_iX_j]\mathbb{E}[X_kX_\ell] \\
- \mathbb{E}[X_iX_k]\mathbb{E}[X_jX_\ell] - \mathbb{E}[X_iX_\ell]\mathbb{E}[X_jX_k] \\
+ 2 \cdot \mathbb{E}[X_iX_\ell]\mathbb{E}[X_jX_\ell]\mathbb{E}[X_kX_\ell] \leq 0 \]

Their paper introduced a classic technique called the random current method.
THE GHS INEQUALITY

The Griffith-Hurst-Sherman inequality states

\[ \mathbb{E}[X_i X_j X_k X_\ell] - \mathbb{E}[X_i X_j] \mathbb{E}[X_k X_\ell] \\
- \mathbb{E}[X_i X_k] \mathbb{E}[X_j X_\ell] - \mathbb{E}[X_i X_\ell] \mathbb{E}[X_j X_k] \\
+ 2 \cdot \mathbb{E}[X_i X_\ell] \mathbb{E}[X_j X_\ell] \mathbb{E}[X_k X_\ell] \leq 0 \]

Their paper introduced a classic technique called the random current method.

Each of these terms arises as a partial derivative of the log partition function, and so does the smooth influence function.
SOME IMPLICATIONS

It turns out this inequality implies

GHS inequality

concavity of magnetization
SOME IMPLICATIONS

It turns out this inequality implies

\[ \frac{\partial^2 I_i}{\partial h_j \partial h_k} \leq 0 \]

concavity of magnetization
SOME IMPLICATIONS

Also Griffith's inequality, which states

\[ \text{Cov}(X_i, X_j) \geq 0 \]
SOME IMPLICATIONS

Also Griffith’s inequality, which states

\[ \text{Cov}(X_i, X_j) \geq 0 \]

in turn implies

\[ \frac{\partial I_i}{\partial h_j} \geq 0 \]
How do these properties imply the discrete influence function is monotone and submodular?
How do these properties imply the discrete influence function is monotone and submodular?

Essentially, by integrating
How do these properties imply the discrete influence function is monotone and submodular?

Essentially, by integrating

**Proof:** Fix $S$ and let $h' = h + \infty \cdot 1_S$
How do these properties imply the discrete influence function is monotone and submodular?

Essentially, by integrating

**Proof:** Fix $S$ and let $h' = h + \infty \cdot 1_S$

Then we can compute

$$I_i(S \cup \{j\}) - I_i(S) = \int_{t=0}^{\infty} \frac{\partial I_i(h' + te_j)}{\partial h_j} dt$$
How do these properties imply the discrete influence function is monotone and submodular?

Essentially, by integrating

**Proof:** Fix $S$ and let $h' = h + \infty \cdot 1_S$

Then we can compute

$$I_i(S \cup \{j\}) - I_i(S) = \int_{t=0}^{\infty} \frac{\partial I_i(h' + te_j)}{\partial h_j} dt$$

which is nonnegative because $\frac{\partial I_i}{\partial h_j} \geq 0$
How do these properties imply the discrete influence function is monotone and submodular?

Essentially, by integrating

**Proof:** Fix $S$ and let $h' = h + \infty \cdot 1_S$

Then we can compute

$$I_i(S \cup \{j\}) - I_i(S) = \int_{t=0}^{\infty} \frac{\partial I_i(h' + te_j)}{\partial h_j} \, dt$$

which is nonnegative because $\frac{\partial I_i}{\partial h_j} \geq 0$ (monotonicity)
Proof: Fix $S \subset T$, let $h' = h + \infty \cdot 1_S$, $h'' = h + \infty \cdot 1_T$
**Proof:** Fix $S \subset T$, let $h' = h + \infty \cdot 1_S$, $h'' = h + \infty \cdot 1_T$

Then we can compute

$$I_i(S \cup \{j\}) - I_i(S) = \int_{t=0}^{\infty} \frac{\partial I_i(h' + te_j)}{\partial h_j} dt$$
Proof: Fix $S \subset T$, let $h' = h + \infty \cdot 1_S \cdot h'' = h + \infty \cdot 1_T$

Then we can compute

$$I_i(S \cup \{j\}) - I_i(S) = \int_{t=0}^{\infty} \frac{\partial I_i(h' + te_j)}{\partial h_j} dt$$

$$\geq \int_{t=0}^{\infty} \frac{\partial I_i(h'' + te_j)}{\partial h_j} dt$$
**Proof:** Fix $S \subset T$, let $h' = h + \infty \cdot 1_S$, $h'' = h + \infty \cdot 1_T$

Then we can compute

$$I_i(S \cup \{j\}) - I_i(S) = \int_{t=0}^{\infty} \frac{\partial I_i(h' + te_j)}{\partial h_j} dt$$

$$\geq \int_{t=0}^{\infty} \frac{\partial I_i(h'' + te_j)}{\partial h_j} dt$$

\[
\begin{aligned}
\left( \text{because } h' \leq h'' \text{ and } \frac{\partial^2 I_i}{\partial h_j \partial h_k} \leq 0 \right)
\end{aligned}
\]
Proof: Fix $S \subset T$, let $h' = h + \infty \cdot 1_S, h'' = h + \infty \cdot 1_T$.

Then we can compute

$$I_i(S \cup \{j\}) - I_i(S) = \int_{t=0}^{\infty} \frac{\partial I_i(h' + te_j)}{\partial h_j} dt$$

$$\geq \int_{t=0}^{\infty} \frac{\partial I_i(h'' + te_j)}{\partial h_j} dt$$

because $h' \leq h''$ and $\frac{\partial^2 I_i}{\partial h_j \partial h_k} \leq 0$

Finally the right hand side is $= I_i(T \cup \{j\}) - I_i(T)$.
Proof: Fix $S \subset T$, let $h' = h + \infty \cdot 1_S, h'' = h + \infty \cdot 1_T$

Then we can compute

$$I_i(S \cup \{j\}) - I_i(S) = \int_{t=0}^{\infty} \frac{\partial I_i(h' + te_j)}{\partial h_j} dt$$

$$\geq \int_{t=0}^{\infty} \frac{\partial I_i(h'' + te_j)}{\partial h_j} dt$$

because $h' \leq h''$ and $\frac{\partial^2 I_i}{\partial h_j \partial h_k} \leq 0$

Finally the right hand side is $= I_i(T \cup \{j\}) - I_i(T)$
DISCUSSION

In general, we need more avenues for circumventing hardness – i.e. **beyond worst-case analysis**
DISCUSSION

In general, we need more avenues for circumventing hardness – i.e. **beyond worst-case analysis**

Even for graphical models, ferromagneticity is just the beginning
In general, we need more avenues for circumventing hardness – i.e. **beyond worst-case analysis**

Even for graphical models, ferromagneticity is just the beginning

**Fact [Folklore]:** Best known algorithms for learning a $d$-junta on $n$ variables run in time $n^{cd}$, but if you perturb the function can learn in time $f(d)\text{poly}(n)$
DISCUSSION

In general, we need more avenues for circumventing hardness – i.e. **beyond worst-case analysis**

Even for graphical models, ferromagneticity is just the beginning

**Fact [Folklore]:** Best known algorithms for learning a d-junta on n variables run in time $n^{cd}$, but if you perturb the function can learn in time $f(d)\text{poly}(n)$

**What if you perturb the parameters of an RBM?**
DISCUSSION

In general, we need more avenues for circumventing hardness – i.e. **beyond worst-case analysis**

Even for graphical models, ferromagneticity is just the beginning

**Fact [Folklore]:** Best known algorithms for learning a $d$-junta on $n$ variables run in time $n^{cd}$, but if you perturb the function can learn in time $f(d)\text{poly}(n)$

What if you perturb the parameters of an RBM?

Are there algorithms that learn the graph structure in $f(d)\text{poly}(n)$ time, even without ferromagneticity?
DISCUSSION

Our hard instances have $2^d$ hidden variables of degree $d$ ...
DISCUSSION

Our hard instances have $2^d$ hidden variables of degree $d$ ... so that the distribution on observed nodes is $(d-1)$-wise indep.
DISCUSSION

Our hard instances have $2^d$ hidden variables of degree $d$ ...

so that the distribution on observed nodes is $(d-1)$-wise indep.

Besides ferromagneticity, are there other ways (e.g. expansion) that preclude sparse parity with noise?
DISCUSSION

Our hard instances have $2^d$ hidden variables of degree $d$ ... so that the distribution on observed nodes is $(d-1)$-wise indep.

Besides ferromagneticity, are there other ways (e.g. expansion) that preclude sparse parity with noise?

And can these conditions lead to new algorithms with provable guarantees?
DISCUSSION

This is not all that different from pseudorandomness, where we use what a model can’t do to fool it.
DISCUSSION

This is not all that different from pseudorandomness, where we use what a model can’t do to fool it.

**In theoretical machine learning, I think we need algorithms and complexity insights even to find the right models.**
DISCUSSION

This is not all that different from pseudorandomness, where we use what a model can’t do to fool it

In theoretical machine learning, I think we need algorithms and complexity insights even to find the right models

We are searching for models just as much (if not more) as we are searching for algorithms
Summary:

• Precise characterization of distributions that can be represented as bounded degree RBMs
• Lower bounds for learning RBMs with constant number of latent variables
• Greedy algorithm for learning ferromagnetic RBMs based on submodularity
Summary:

• Precise characterization of distributions that can be represented as bounded degree RBMs
• Lower bounds for learning RBMs with constant number of latent variables
• Greedy algorithm for learning ferromagnetic RBMs based on submodularity

Thanks! Any Questions?