Uprooting and Rerooting Graphical Models

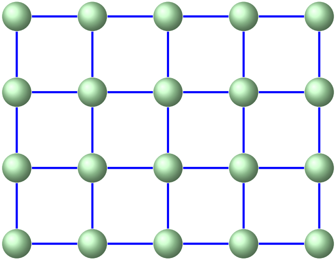
Adrian Weller
University of Cambridge

ICML, New York, NY
June 21, 2016

For more information, see
http://mlg.eng.cam.ac.uk/adrian

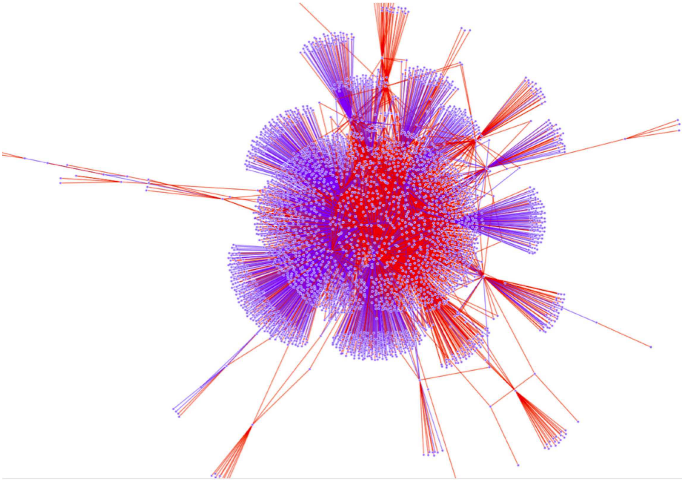
Motivation: *undirected graphical models*

- Powerful way to represent relationships across variables
- Many applications including: computer vision, social network analysis, deep belief networks, protein folding...
- In this talk, focus on binary pairwise (Ising) models



Example: Grid for computer vision (attractive)

Motivation: *undirected graphical models*



Example: Part of epinions social network

Figure courtesy of N. Ruozzi

Fundamental problems of inference

1. MAP inference: find a global configuration of all variables with highest probability
2. Marginal inference: estimate marginal probability distribution of one variable

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

3. Computing the partition function, requires summing over configurations of all variables

All are computationally intractable (NP-hard)

Fundamental problems of inference

1. MAP inference: find a global configuration of all variables with highest probability
2. Marginal inference: estimate marginal probability distribution of one variable

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

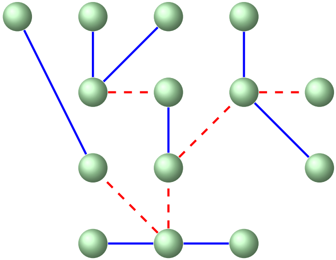
3. Computing the partition function, requires summing over configurations of all variables

All are computationally intractable (NP-hard)

But inference is easier for some models than others

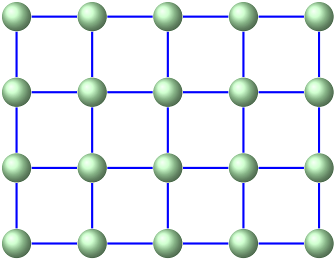
When is inference (relatively) easy?

Tree



STRUCTURE

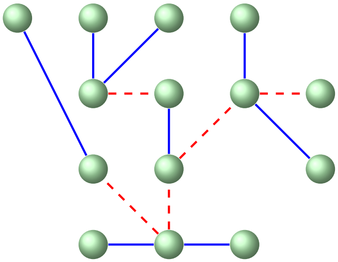
Attractive model



POTENTIALS

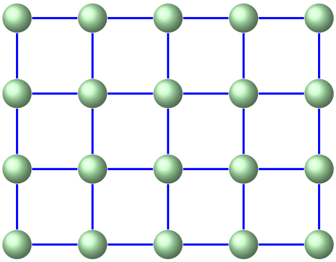
When is inference (relatively) easy?

Tree



STRUCTURE

Attractive model



POTENTIALS

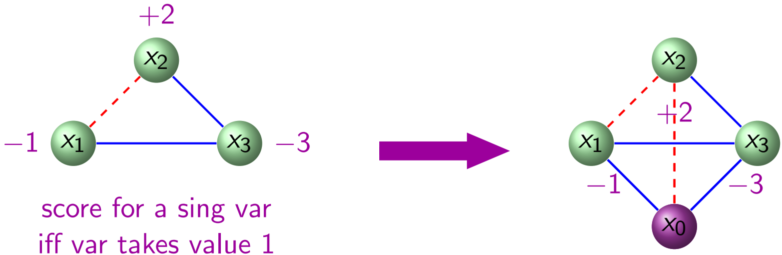
No mention of singleton potentials!

Can we do better by also examining their properties?

Idea: Uprooting (not new)

- Add a new variable X_0
- Transform singleton potentials $\longrightarrow$ edge potentials to X_0

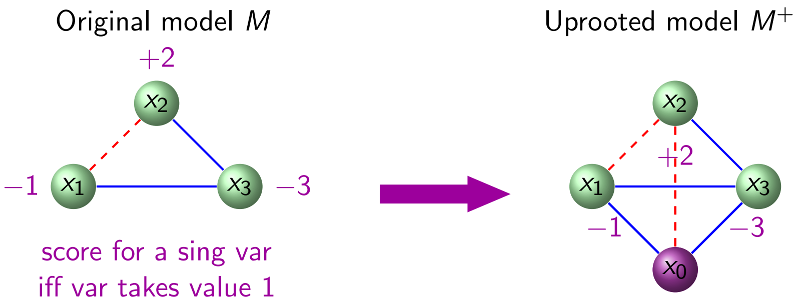
Original model M — Uprooted model M^+



score for a sing var
iff var takes value 1

score for an edge iff its end variables are different

Uprooting



score for an edge iff its end variables are different

- M is M^+ with X_0 *clamped* to 0, write $M = M_0$
- If we don't clamp, each config of $M \to 2$ configs of M^+ with the same score

  e.g. $(x_1, x_2, x_3) = (1, 0, 1) \to (x_0, x_1, x_2, x_3) = \begin{cases} (0, 1, 0, 1) \\ (1, 0, 1, 0) \end{cases}$

Uprooted model M^+, fully symmetric

M^+ config				edges: score ✓ if ends different					
x_0	x_1	x_2	x_3	e_{01}	e_{02}	e_{03}	e_{12}	e_{13}	e_{23}
0	0	0	0						
0	0	0	1			✓		✓	✓
0	0	1	0		✓		✓		✓
0	0	1	1		✓	✓	✓	✓	
0	1	0	0	✓			✓	✓	
0	1	0	1	✓		✓	✓		✓
0	1	1	0	✓	✓			✓	✓
0	1	1	1	✓	✓	✓			
1	0	0	0	✓	✓	✓			
1	0	0	1	✓	✓			✓	✓
1	0	1	0	✓		✓	✓		✓
1	0	1	1	✓			✓	✓	
1	1	0	0		✓	✓	✓	✓	
1	1	0	1		✓		✓		✓
1	1	1	0			✓		✓	✓
1	1	1	1						

Uprooted model M^+, fully symmetric

M^+ config				edges: score ✓ if ends different					
x_0	x_1	x_2	x_3	e_{01}	e_{02}	e_{03}	e_{12}	e_{13}	e_{23}
0	0	0	0						
0	0	0	1			✓		✓	✓
0	0	1	0		✓		✓		✓
0	0	1	1		✓	✓	✓	✓	
0	1	0	0	✓			✓	✓	
0	1	0	1	✓		✓	✓		✓
0	1	1	0	✓	✓			✓	✓
0	1	1	1	✓	✓	✓			
1	0	0	0	✓	✓	✓			
1	0	0	1	✓	✓			✓	✓
1	0	1	0	✓		✓	✓		✓
1	0	1	1	✓			✓	✓	
1	1	0	0		✓	✓	✓	✓	
1	1	0	1		✓		✓		✓
1	1	1	0			✓		✓	✓
1	1	1	1						

Original model $M = M_0$ is M^+ 'rooted at' $x_0 = 0$

M^+ config				edges: score ✓ if ends different					
x_0	x_1	x_2	x_3	e_{01}	e_{02}	e_{03}	e_{12}	e_{13}	e_{23}
0	0	0	0						
0	0	0	1			✓		✓	✓
0	0	1	0		✓		✓		✓
0	0	1	1		✓	✓	✓	✓	
0	1	0	0	✓			✓	✓	
0	1	0	1	✓		✓	✓		✓
0	1	1	0	✓	✓			✓	✓
0	1	1	1	✓	✓	✓			
1	0	0	0	✓	✓	✓			
1	0	0	1	✓	✓			✓	✓
1	0	1	0	✓		✓	✓		✓
1	0	1	1	✓			✓	✓	
1	1	0	0		✓	✓	✓	✓	
1	1	0	1		✓		✓		✓
1	1	1	0			✓		✓	✓
1	1	1	1						

Idea: Reroot to form M_1 as M^+ 'rooted at' $x_1 = 0$

M^+ config				edges: score ✓ if ends different					
x_0	x_1	x_2	x_3	e_{01}	e_{02}	e_{03}	e_{12}	e_{13}	e_{23}
0	**0**	0	0						
0	**0**	0	1			✓		✓	✓
0	**0**	1	0		✓		✓		✓
0	**0**	1	1		✓	✓	✓	✓	
0	1	0	0	✓			✓	✓	
0	1	0	1	✓		✓	✓		✓
0	1	1	0	✓	✓			✓	✓
0	1	1	1	✓	✓	✓			
1	**0**	0	0	✓	✓	✓			
1	**0**	0	1	✓	✓			✓	✓
1	**0**	1	0	✓		✓	✓		✓
1	**0**	1	1	✓			✓	✓	
1	1	0	0		✓	✓	✓	✓	
1	1	0	1		✓		✓		✓
1	1	1	0			✓		✓	✓
1	1	1	1						

Idea: Reroot to form M_2 as M^+ 'rooted at' $x_2 = 0$

M^+ config				edges: score ✓ if ends different					
x_0	x_1	x_2	x_3	e_{01}	e_{02}	e_{03}	e_{12}	e_{13}	e_{23}
0	0	**0**	0						
0	0	**0**	1			✓		✓	✓
0	0	1	0		✓		✓		✓
0	0	1	1		✓	✓	✓	✓	
0	1	**0**	0	✓			✓	✓	
0	1	**0**	1	✓		✓	✓		✓
0	1	1	0	✓	✓			✓	✓
0	1	1	1	✓	✓	✓			
1	0	**0**	0	✓	✓	✓			
1	0	**0**	1	✓	✓			✓	✓
1	0	1	0	✓		✓	✓		✓
1	0	1	1	✓			✓	✓	
1	1	**0**	0		✓	✓	✓	✓	
1	1	**0**	1		✓		✓		✓
1	1	1	0			✓		✓	✓
1	1	1	1						

Idea: Reroot to form M_3 as M^+ 'rooted at' $x_3 = 0$

M^+ config				edges: score ✓ if ends different					
x_0	x_1	x_2	x_3	e_{01}	e_{02}	e_{03}	e_{12}	e_{13}	e_{23}
0	0	0	**0**						
0	0	0	1			✓		✓	✓
0	0	1	**0**		✓		✓		✓
0	0	1	1		✓	✓	✓	✓	
0	1	0	**0**	✓			✓	✓	
0	1	0	1	✓		✓	✓		✓
0	1	1	**0**	✓	✓			✓	✓
0	1	1	1	✓	✓	✓			
1	0	0	**0**	✓	✓	✓			
1	0	0	1	✓	✓			✓	✓
1	0	1	**0**	✓		✓	✓		✓
1	0	1	1	✓			✓	✓	
1	1	0	**0**		✓	✓	✓	✓	
1	1	0	1		✓		✓		✓
1	1	1	**0**			✓		✓	✓
1	1	1	1						

Rerooting observations

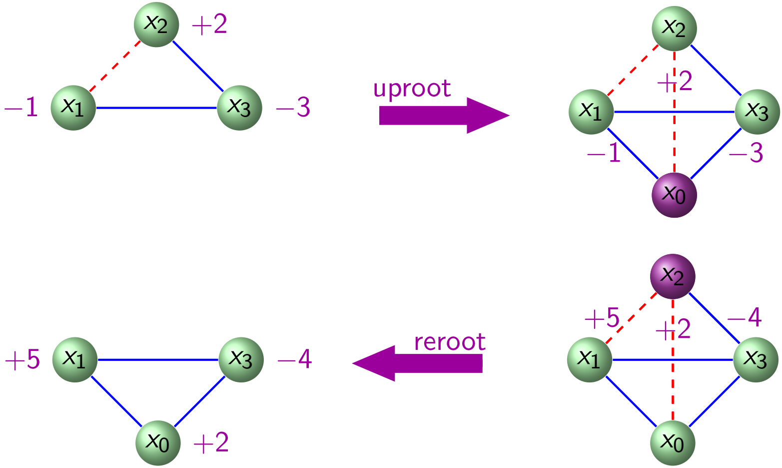
- Rerooted models $\{M_i\}$ form an equivalence class, each has the same 'parent' uprooted model M^+
- Simple score-preserving 1-1 correspondence

  $$\forall i, \text{ configs of } M_i \leftrightarrow \text{ configs of } M_0$$

- Inference (exact or approx) on *any* $M_i \rightarrow$ recover info for M_0
  - Each M_i has the same partition function as M_0
  - MAP config of $M_i \rightarrow$ recover MAP config of M_0
  - Marginals of M_i $\rightarrow$ recover marginals of M_0
- Inference may be much faster / more accurate on some M_i
- Singleton and edge potentials are essentially the same, only appear different due to choice of rooting

Rerooting example

Original model $M = M_0$ Uprooted model M^+



Rerooted model M_2

attractive model

Rerooting: How to pick a good root variable?

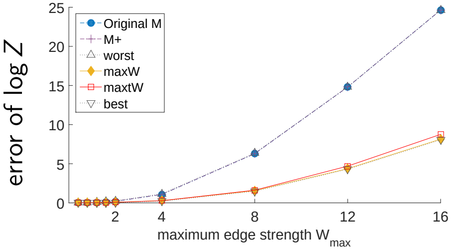
- Same as choosing a good variable to clamp in M^+
- Rerooting substitutes an implicit initial clamp choice for a well chosen one ‘for free’

- Several existing good methods, including maxW
- Idea: break heavy cycles
- Will lead to picking a root to form high singleton potentials
- We introduce maxtW: strength of an edge weight saturates, works well in our context

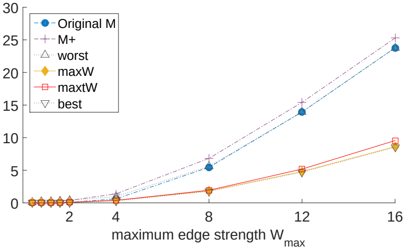
Implications of Rerooting

- Rerooting allows us to generalize or improve many results
- e.g. max flow / min cut can now be used for models where $\exists i$ s.t. M_i is attractive $\Leftrightarrow$ M^+ is almost attractive
  (balanced) (almost balanced)
- e.g. bounds (on marginals, partition function) can be improved by considering different rerootings
- Reveals intriguing perspective: TRI is universally rooted (TRI is the triplet-consistent polytope)

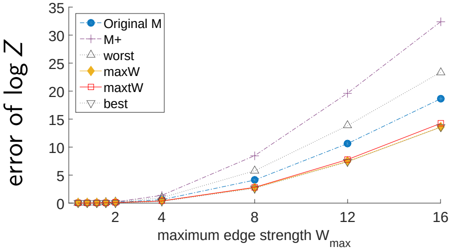
Experiments (Bethe): complete graph on 10 variables



low $\theta_i \sim [-0.1, 0.1]$



medium $\theta_i \sim [-2, 2]$



θ_i & W_{ij} scale together

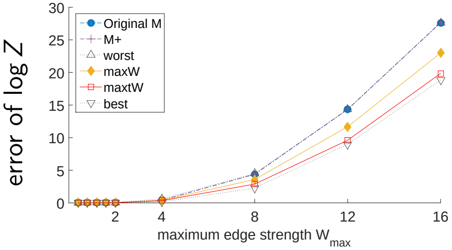
Observe

Rerooting is very effective

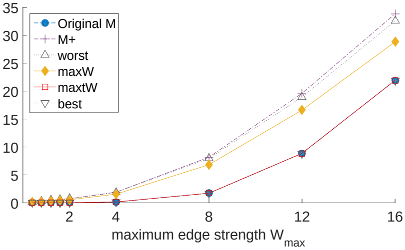
Better for low θ_i

maxW and *maxtW* both perform well

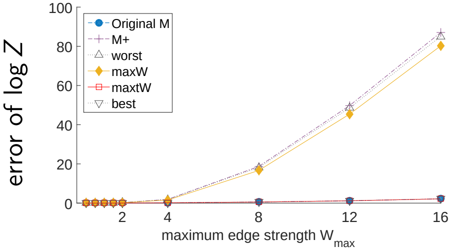
Experiments (Bethe): 9×9 grid



low $\theta_i \sim [-0.1, 0.1]$



medium $\theta_i \sim [-2, 2]$



θ_i & W_{ij} scale together

Observe

Rerooting is less effective

Still helpful for low θ_i

maxtW performs much better than *maxW*

Conclusion

- We can uproot and then reroot any binary pairwise graphical model
- Obtain an equivalence class of models
- Generalizes earlier theoretical results
- Useful in practice, particularly for dense models with strong edges and weak singleton potentials

- Comparison to clamping in M_0 -
  - Clamping requires performing one inference run for each value of the variable clamped
  - Here we get a clamping 'for free'
- Rerooting reveals intriguing perspectives such as TRI is universally rooted

Thank you

Poster #49 tomorrow 10am-1pm

http://mlg.eng.cam.ac.uk/adrian