Def. *Intervention* forces one or several random variables

$$X_I = \{X_i\}_{i \in I}$$

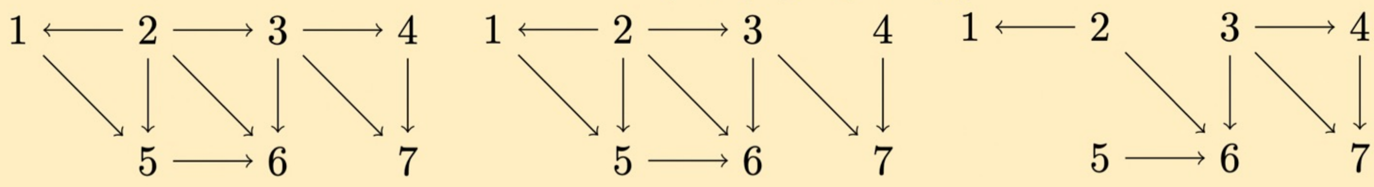
called intervention target to be the value of <u>independent</u> random variables $U_I \in \mathcal{X}^{|I|}$.

Intervention density of X under such intervention is

$$P_G(x \mid do(X_I = U_I)) \triangleq \prod_{i \in I} P_{U_i}(x_i) \cdot \prod_{i \notin I} P_i(x_i \mid x_{\pi_i})$$

This density has the Markov property of the *intervention graph* $G^{(I)}$, which is defined from G but removing all edges pointing to nodes in I.

Ex: Perfect interventions $\mathcal{I} = \{\emptyset, \{4\}, \{3, 5\}\}$



(a) $G^{\emptyset}$ (b) $G^{(\{4\})}$ (c) $G^{(\{3,5\})}$

$\{\phi, \{4\}, \{3,5\}\}$: set of targets.

Def. $\mathbb{I}$-Markov Equivalence Class : two DAGs G_1 and G_2 are $\mathbb{I}$-Markov Equivalent if

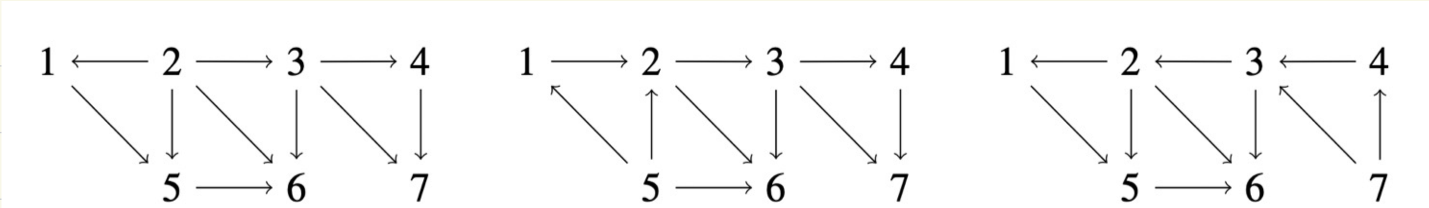
1. $G_1 \sim G_2$, i.e., both in the same Markov Equivalent class
2. $G_1^{(I)}$ and $G_2^{(I)}$ have the same skeleton for all $I \in \mathbb{I}$.

Claim: We can identify a DAG up to its $\mathbb{I}$-Markov Equivalence Class with interventional data on the targets $\mathbb{I}$, as $N \to \infty$.

$\Rightarrow$ We can orient any edges that are
- orientable from observational data
- adjacent to an intervened node.

example> $P(x) = P_1(x_1)\, P_2(x_2|x_1)\, P_3(x_3|x_2)$

	ϕ-MEC	$\{3\}$-MEC	$\{2\},\{3\}$-MEC
Ground Truth:	$1 \to 2 \to 3$	$1 \to 2 \to 3$	$1 \to 2 \to 3$
	$1 \leftarrow 2 \to 3$	$1 \leftarrow 2 \to 3$	
	$1 \leftarrow 2 \leftarrow 3$		



(a) D (b) D_1 (c) D_2

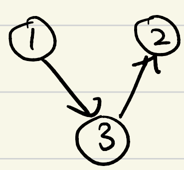
Q. Are they MEC, Markov Equivalent class?

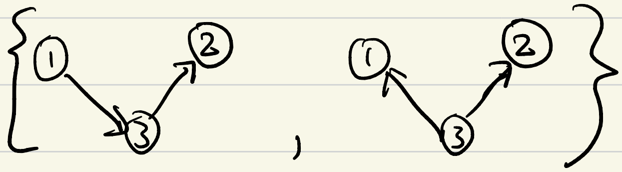
Q. Are they $\mathcal{I}$-MEC, interventional Markov Equivalent class, with $\mathcal{I} = \{\emptyset, \{4\}\}$?

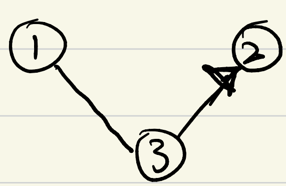
Def. $\mathcal{I}$-essential graph of a DAG G and targets $\mathcal{I}$ is defined as

$$\mathcal{E}_{\mathcal{I}}(G) = \bigcup_{G' \sim_{\mathcal{I}} G} G'$$

the union of Graphs (meaning union of the edges) that are $\mathcal{I}$-Markov Equivalent to G.

$G =$  , $\mathcal{I} = \{\emptyset, 2\}$

$\mathcal{I}$-MEC(G) : $\{$  $\}$

$\mathcal{E}_{\mathcal{I}}(G) =$ 

* Greedy Interventional Equivalence Search [GIES]

Adopt GES algorithm (forward-backward algorithm) for finding MEC under observational Data to interventional Data.

Definition. we want to define a score function over DAG G, target set $\mathcal{I}$, and interventional data $D = \bigcup_{I \in \mathcal{I}} D^{(I)}$.

$D^{(I)} = \{X_i\}_{i=1}^{N_I}$

$\mathrm{SCORE}(G, \mathcal{I}, D)$, that is decomposable

- A score function is decomposable if

$$\mathrm{SCORE}(G, \mathcal{I}, D) = \sum_{i=1}^{n} h(i, \pi_i, \mathcal{I}, D)$$

For example, we can use BIC score:

$$\mathrm{SCORE}(G, \mathcal{I}, D) = \max_{\{P_i(X_i | X_{\pi_i})\}} \sum_{I \in \mathcal{I}} \sum_{l=1}^{N_I} P(X^{(l)} | X_I^{(l)}) U(X_I^{(l)}) - \frac{\log \sum N_I}{2} \cdot \dim(G).$$

[GIES Algorithm]

Input: intervention targets $\mathcal{I}$, interventional data D

Output: $\mathcal{I}$-essential graph.

① initialize $G = (V, \emptyset)$

② forward phase

$$G^{(t+1)} \longleftarrow \underset{G' \in \text{Neighborhood}^{+}(G^{(t)})}{\arg\max} \ \text{SCORE}(G', \mathcal{I}, D).$$

where Neighborhood^{+} of a DAG $G^{(t)}$ is

$$\{ \text{DAG } G' = (V, E') \mid E' = E \cup (\overrightarrow{i,j}) \text{ for some } G(V,E) \sim G^{(t)} \}$$

③ backward phase

$$G^{(t+1)} \longleftarrow \underset{G' \in \text{Neighborhood}^{-}(G^{(t)})}{\arg\max} \ \text{SCORE}(G', \mathcal{I}, D)$$

where Neighborhood^{-} of a DAG $G^{(t)}$ is

$$\{ \text{DAG } G' = (V, E') \mid E' = E \setminus (\overrightarrow{i,j}) \text{ for some } G(V,E) \sim G^{(t)} \}$$

*Course Overview / Going forward.

Part 1. Graphical Models & Conditional Independence structures.

Part 2. Inference Problems
- $\arg\max_x P(X|Y)$
- marginal $P(X_i|Y)$
- compute Z.

- Belief Propagation
  - Discrete
  - Gaussian BP
- Variational Methods
- Gibbs Sampling.

Part 3. Learning Graphical Models
- structure Learning
- Causal structure Discovery.

If you liked this course:

CSE 546. Machine Learning

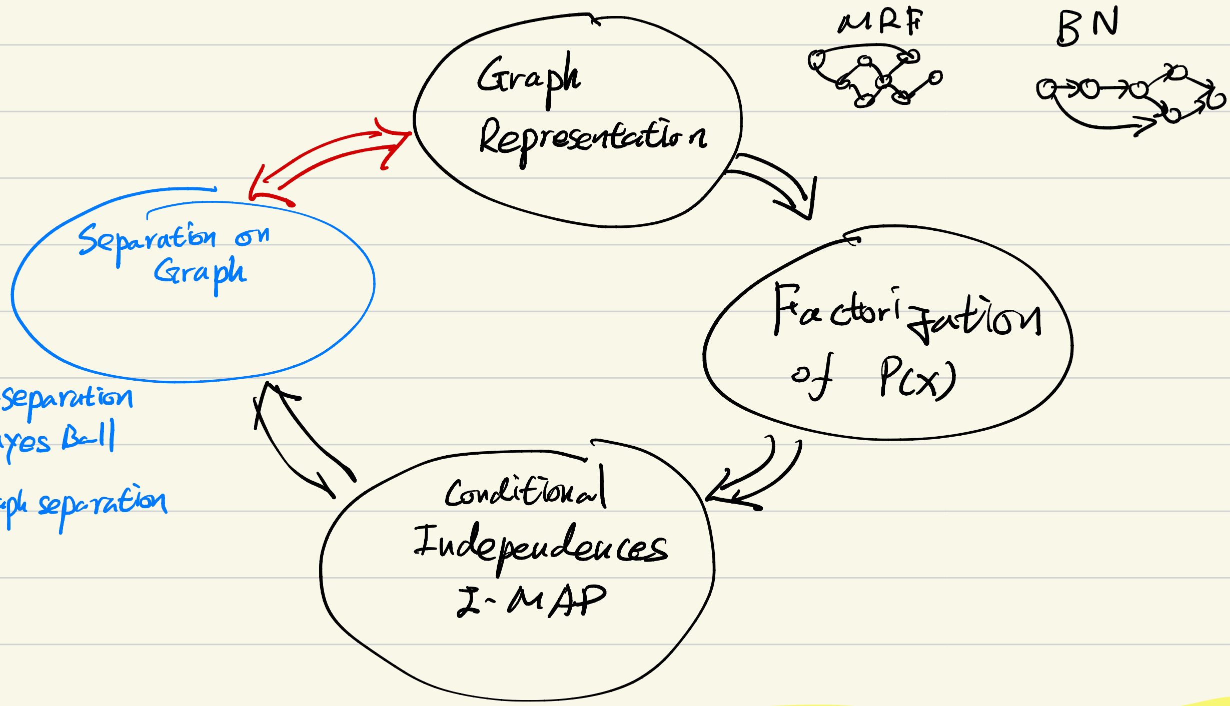
CSE 547. Machine Learning for Big Data

STAT 535. Statistical Learning: Modeling, Prediction, and computing.

STAT 567. Statistical Analysis of Socian Networks.

STAT 566. Causal Modeling.

Part 1. Probabilistic Graphical Models & Markov Properties



BN: d-separation
Bayes Ball

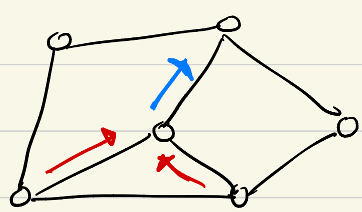
MRF: graph separation

*Beautiful marriage between graph theory and probability theory
(separation) (independence)
Representation

Part 2.1. Inference Problems & Belief Propagation.

equivalently difficult/easy
- Marginalize $P(X_i \mid Y)$
- MAP $\arg\max P(X \mid Y)$
- Partition function Z
- Sampling

Belief Propagation

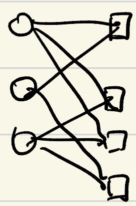


$$m_{i\to j}(x_j) \propto \sum_{x_i} f_{ij}(x_i,x_j) \prod_{k\in\partial i \setminus j} m_{k\to i}(x_i)$$

*Beautiful marriage between graph theory and probability theory
(message passing on Graph) (approximate computation of posteriori distribution)
Efficient computation

* Belief Propagation & Graph Neural Networks.

BP for factor graphs



$$m_{i\to a}(x_i) = \prod_{b\in\partial i\setminus a} \tilde{m}_{b\to i}(x_i)\cdot f_i(x_i;Y_i)$$

$$\tilde{m}_{a\to i}(x_i) = \sum_{X_{\partial a\setminus i}} f_a(X_{\partial a})\prod_{j\in\partial a\setminus i} m_{j\to a}(x_j)$$

in many applications we do not know this factor.
and learning is too complex.

A Graphical Model perspective on Graph Neural Networks.

$$m_{i\to a} = h_w(\{\tilde{m}_{b\to i}\}_{b\in\partial i\setminus a}, Y_i)$$

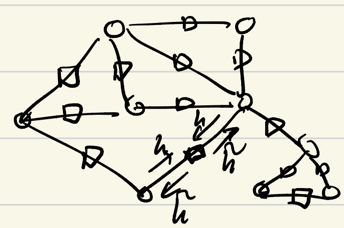
Neural Network – DeepSet
RNN, LSTM

$$\tilde{m}_{a\to i} = \tilde{h}_{\tilde{w}}(\{m_{j\to a}\}_{j\in\partial a\setminus i})$$

and train the weights $w, \tilde{w}$ on some loss. ??

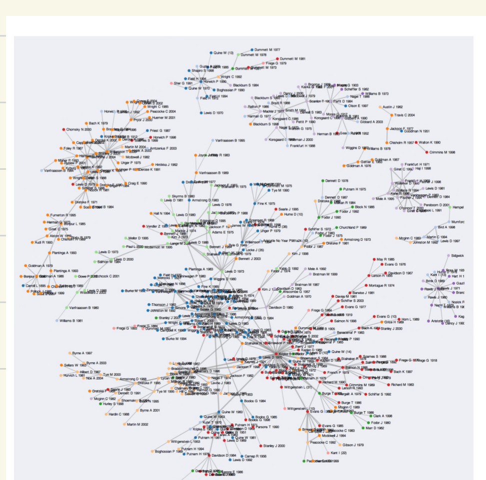
think of the whole thing

as a Deep Neural Architecture.



examples of GNN >.

① Semi-supervised classification of citation network.



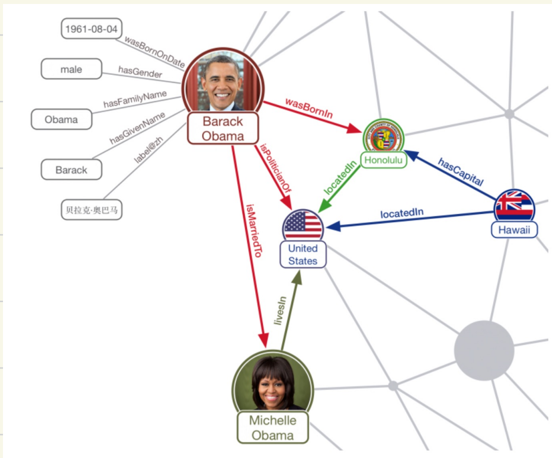
Y_i : feature of document i
e.g. venue, year, BoW.

loss: for a subset of nodes, we have label $Z_i \in \{$NLP, machine learning, computer vision$\}$

$$\sum_{i\in S} \ell(Z_i, f_w(x_i))$$

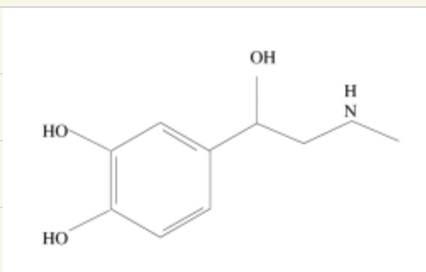
$\{0,1\}^3$: one-hot encoding.

② Predicting links in knowledge graph.



loss: the current edges in G.

③ classifying molecular Network for drug discovery

(

, $\{0,1\}$) $\Rightarrow$ Dataset of paired samples.

feature   label

Part 2.2 Variational Methods

$$\max_{b} G_T(b) = \log Z_G - D_{KL}(b \| p)$$

1st order statistics   2nd order statistics

Naive Mean Field — Bethe free energy — Gibbs free energy — Tree reweighted BP

$\Downarrow$

Recover B.P. $\Rightarrow$ opens new doors other Graph-based Algorithms.

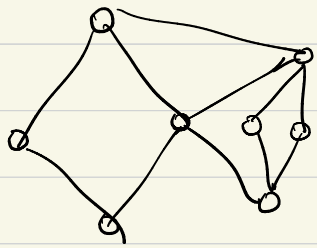
*Beautiful marriage between graph theory and probability theory

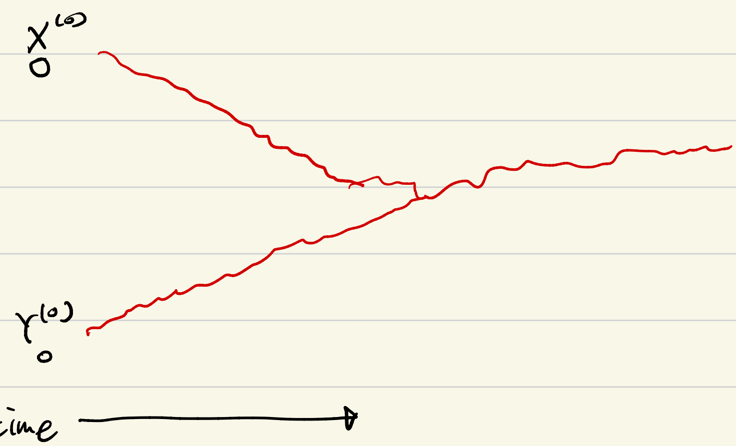
feasible set   Density Estimation

↳ Computational Efficiency.

Part 2.3 Sampling.

Gibbs Sampling.





*Beautiful marriage between graph theory and probability theory

Local structure of a graph. — Convergence of Markov Chain

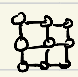
$\underbrace{\hspace{6em}}_{\text{Design fast mixing M.C.}}$

* Inference and Deep Generative Priors.

ex> super resolution

Graphical Models — done by Human.

step 1: construct a G.M. over pixels.



step 2: find $\arg\max_{X} P(X|Y)$

Y ↑ low-resolution.

Deep Generative Priors.

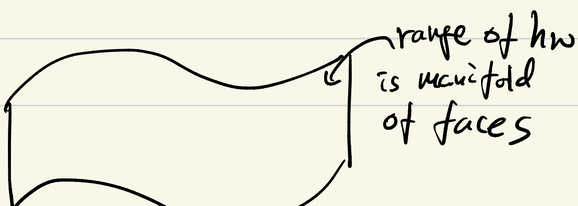
step1. Learn a deep Generative Model.

face $\sim h_w(z)$

z ↑ Gaussian.

step2. find $z : \sum_{ij} (h_w(z)_{ij} - Y_{ij})^2$

Y_{ij} ↑ low-resolution



Part 3. Learning Graphical Models.

- Structure Learning
- Causal Structure Discovery.

Samples → Conditional Independence
→ SCORE

far from practical, and many open questions.